Centrality

Students
FRANCISCO REYES
ADRIAN LARA TLAXCALA

Supervisors
DR. VADIM KOLESNIKOV
DR. IVONNE MALDONADO
DR. VIKTAR KIREYEU

Cuts

Cuts used
eta in [-1,2]
DCA <= 2
Nhits >= 20

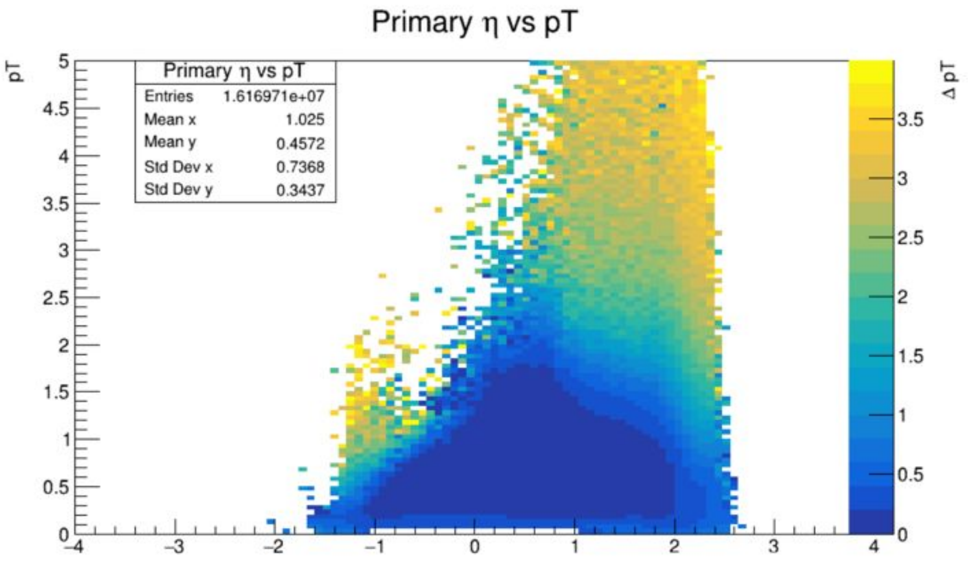
Number of events: 384,800

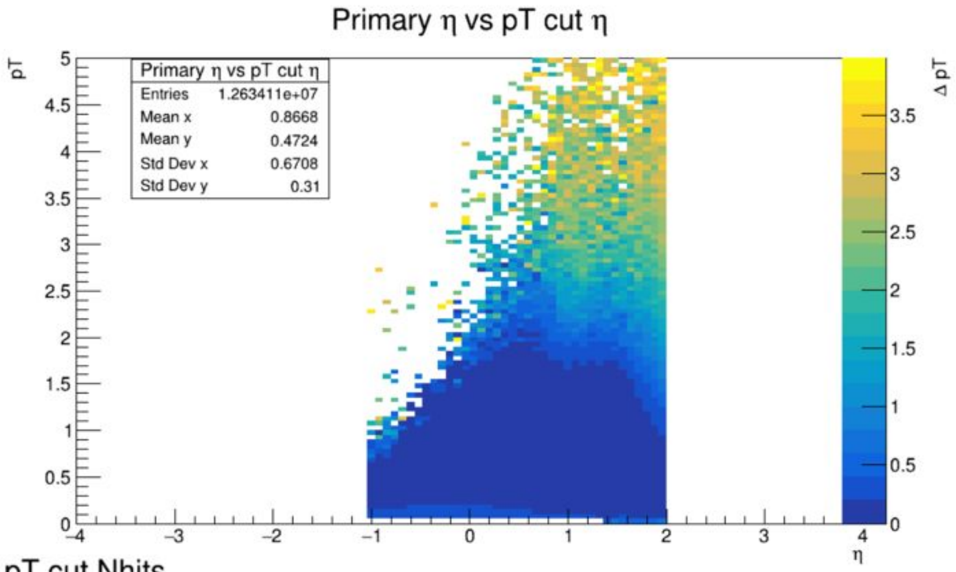
Reconstructed data

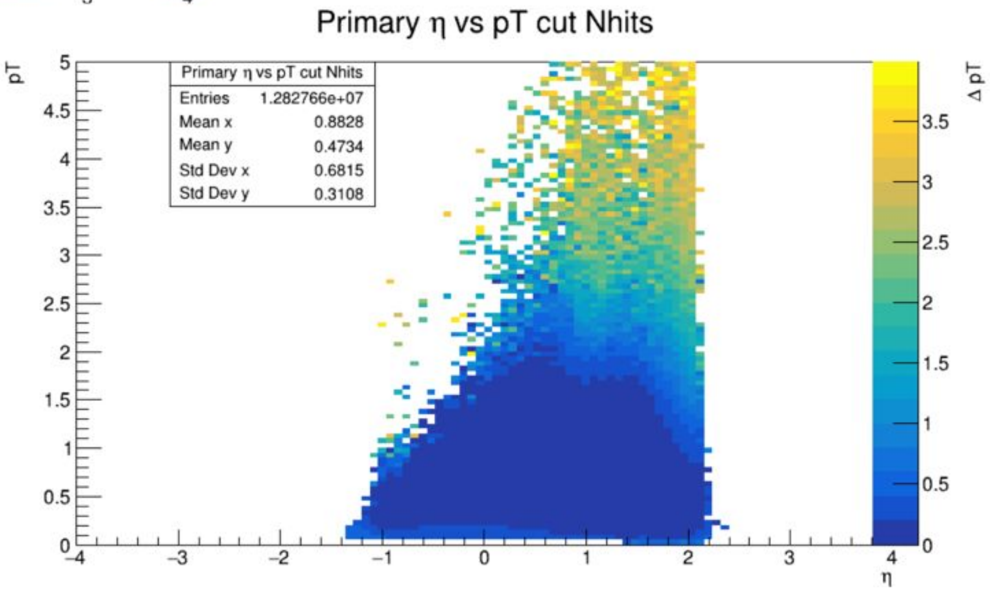
ncx cluster route:
/scratch1/maldonado/FXT/SIM_85_XeW_2.5GeV/

For all work

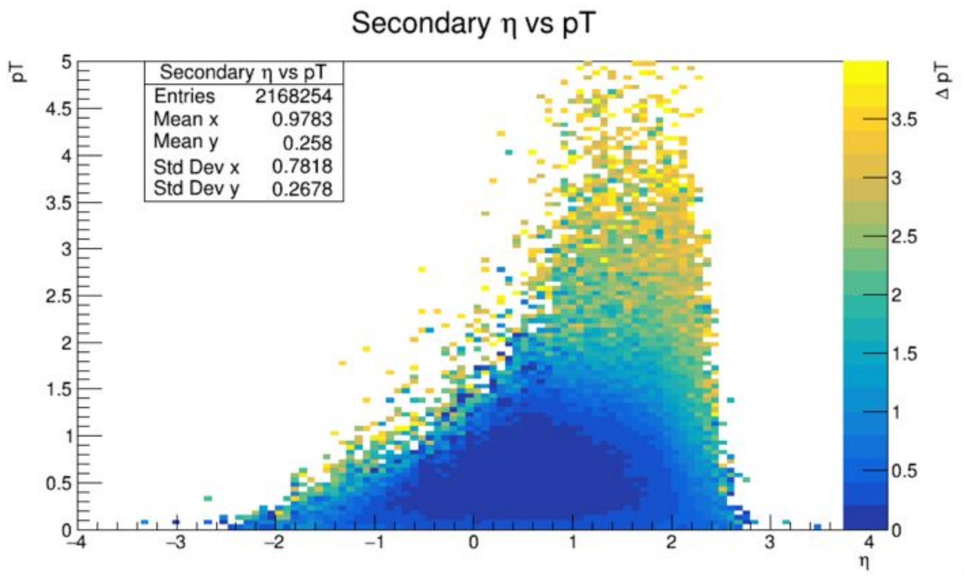
Primary eta vs pT distributions

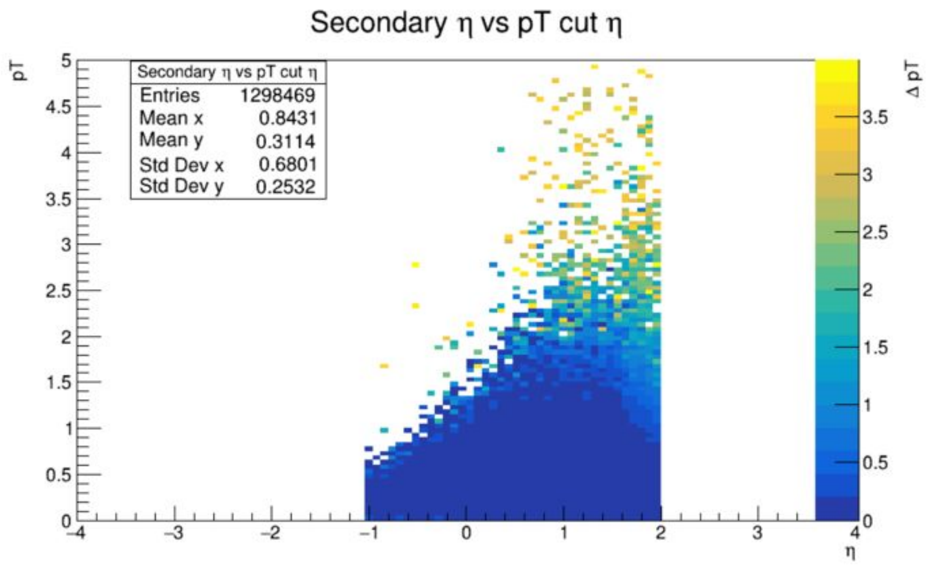


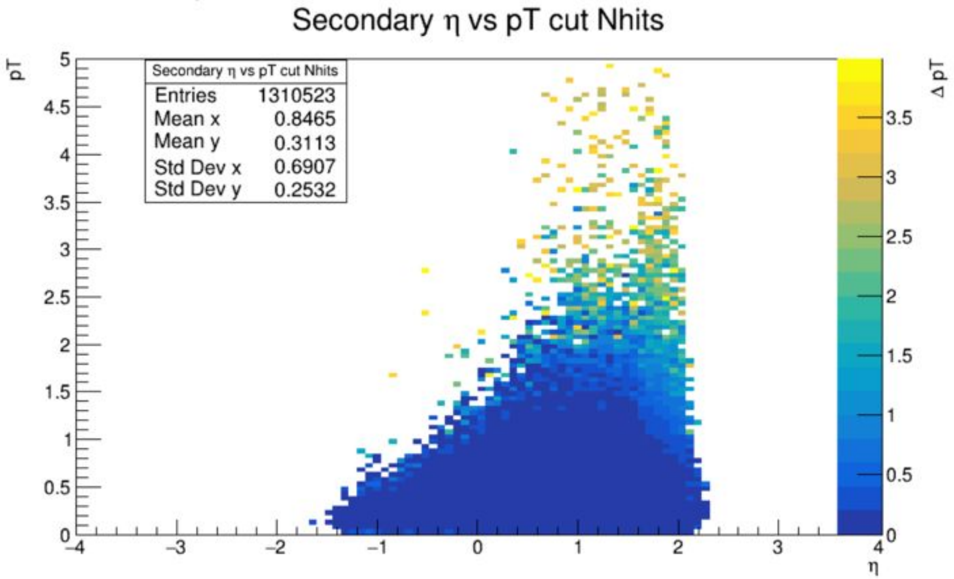




Secondary eta vs pT distributions







Gamma fit

IS A METHOD FOR DETERMINING CENTRALITY FROM THE MULTIPLICITY BY CREATING AN ADJUSTMENT WITH DIFFERENT PARAMETERS

Centrality from the point of view of multiplicity

$$c_{N_{ch}} = \frac{1}{\sigma_{inel}} \int_{N_{ch}}^{\infty} \frac{d\sigma}{dN_{ch}} dN_{ch}$$

σ_{inel} Inelastic cross-section

N_{ch} Multiplicity

$$\frac{1}{N}\frac{dN}{dN_{ch}} = n_{pp}\left[(1-x)\frac{N_{part}}{2} + xN_{coll}\right]$$

n_{pp} Average multiplicity per unit of pseudorapidity

N_{part} Number of participants

N_{coll} Number of colliders

$c_{N_{ch}}$ Centrality

Determination of the centrality

$$P(N_{ch}|b) = \frac{1}{\Gamma(k)\theta^k} N_{ch}^{k-1} e^{-N_{ch}/\theta}$$

$k\theta$ Define the shape of the multiplicity distribution and can be attributed to the mean and standard deviation

b Impact Parameter

$$\Gamma(k) \equiv \int_0^\infty x^{k-1} e^{-x} dx \qquad \langle N_{ch} \rangle = k\theta, \ \sigma_{N_{ch}} = \sqrt{k}\theta.$$

$P(N_{ch}|b)$ Probability distribution of the charge particle multiplicity

$$\frac{1}{M} M_{\Gamma-fit} \equiv P(N_{ch}) = \int_0^\infty P(N_{ch}|b) P(b) db, \ P(b) = \frac{2\pi b}{\sigma_{inel}} P_{inel}(b).$$

Determination of the centrality

$$c_b = \int_0^b P(b')db'.$$

c_b Cumulative probability distribution

$$P(N_{ch}) = \int_0^1 P(N_{ch}|c_b)dc_b.$$

$$k(b) = k_0 \cdot exp\left[-\sum_{i=1}^{3} a_i \left(c_b\right)^i\right]$$

Results

From the simulated data for the experiment, the MpdDstReader.C macro is used to have a multiplicity histogram, later the Framework software with the GammaFit method was used

For a sample of Xenon 124 and Wolframio 184 at an energy of 2.94 at the center of mass

Fixed target experiment

https://github.com/FlowNICA/CentralityFramework/tree/master

https://indico.jinr.ru/event/3206/contributions/17259/attachments/12935/21631/QAreq25.pdf

Configuration the GammaFit2.C

If the effective section is known, the parameters must be configured

You can select whether to make an automatic adjustment or if you want to insert the effective section and start there to make the adjustment:
True performs the auto adjustment
False if effective section is known

```
bool GetSigma = true;
Float_t sigma = 571.2;
Float_t pi = TMath::Pi(), bmax = 18;
Color_t color[10] = {kRed + 2, kBlue + 1, 14, kGreen + 3, kMagenta + 3, kGreen + 1, kYellow + 2, 46, kBlue - 9, kViolet + 8};
//  Float_t teta = 1.367447, n_knee = 140.194, a1 = -2.54682, a2 = 0.993259, a3 = -2.94017, chi2, NDF, chi2_NDF;
 //Float_t teta = 1, n_knee = 70.194, a1 = -4, a2 = 2.4, a3 = -3.94017, chi2, NDF, chi2_NDF;
// Au+Au
// Float_t teta = 0.6, n_knee = 137, a1 = -2.4, a2 = 0.58, a3 = -2.2, chi2, NDF, chi2_NDF; // 2.4, 2.7 GeV
Float_t teta = 1.1, n_knee = 220, a1 = -5., a2 = 2., a3 = -5., chi2, NDF, chi2_NDF;
// Float_t teta = 1.1, n_knee = 900, a1 = -4., a2 = -4., a3 = 5., chi2, NDF, chi2_NDF;
```

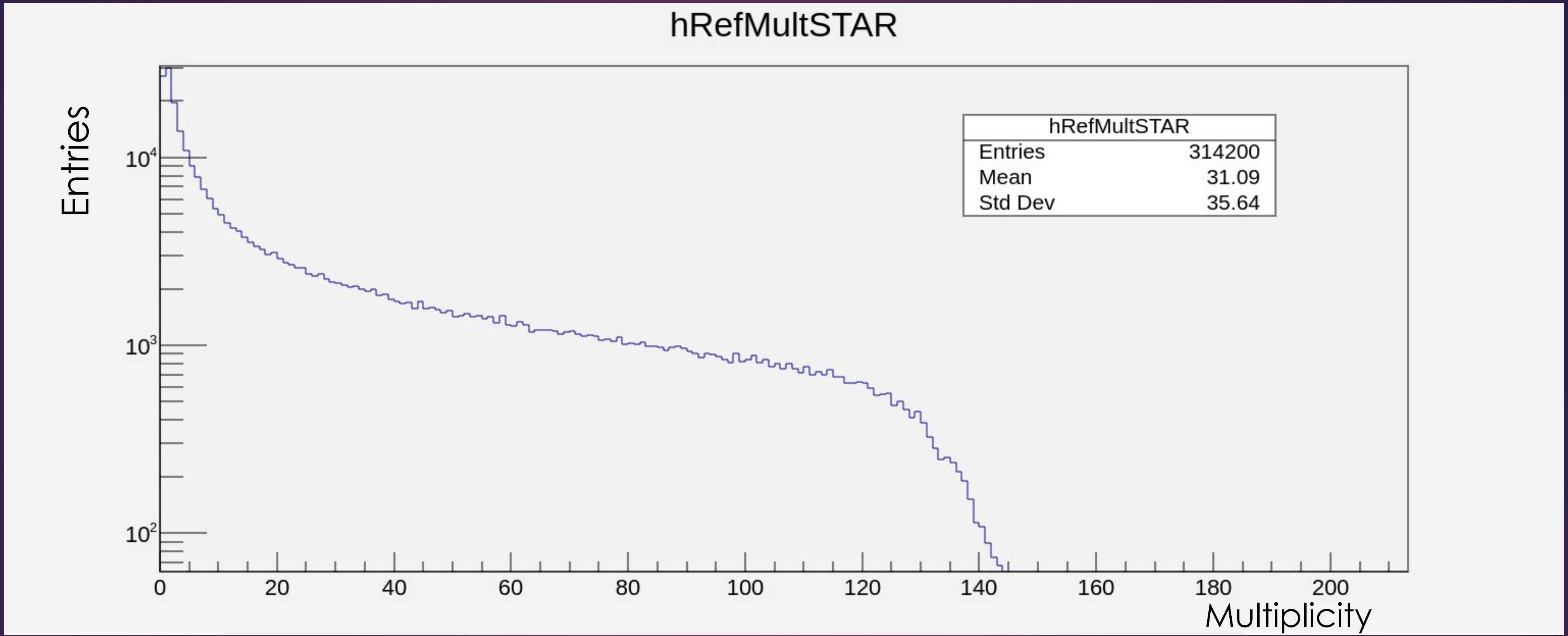
run GammaFit2.C and printFinal.C

Environment variables of a new version different from those of the github documentation are loaded

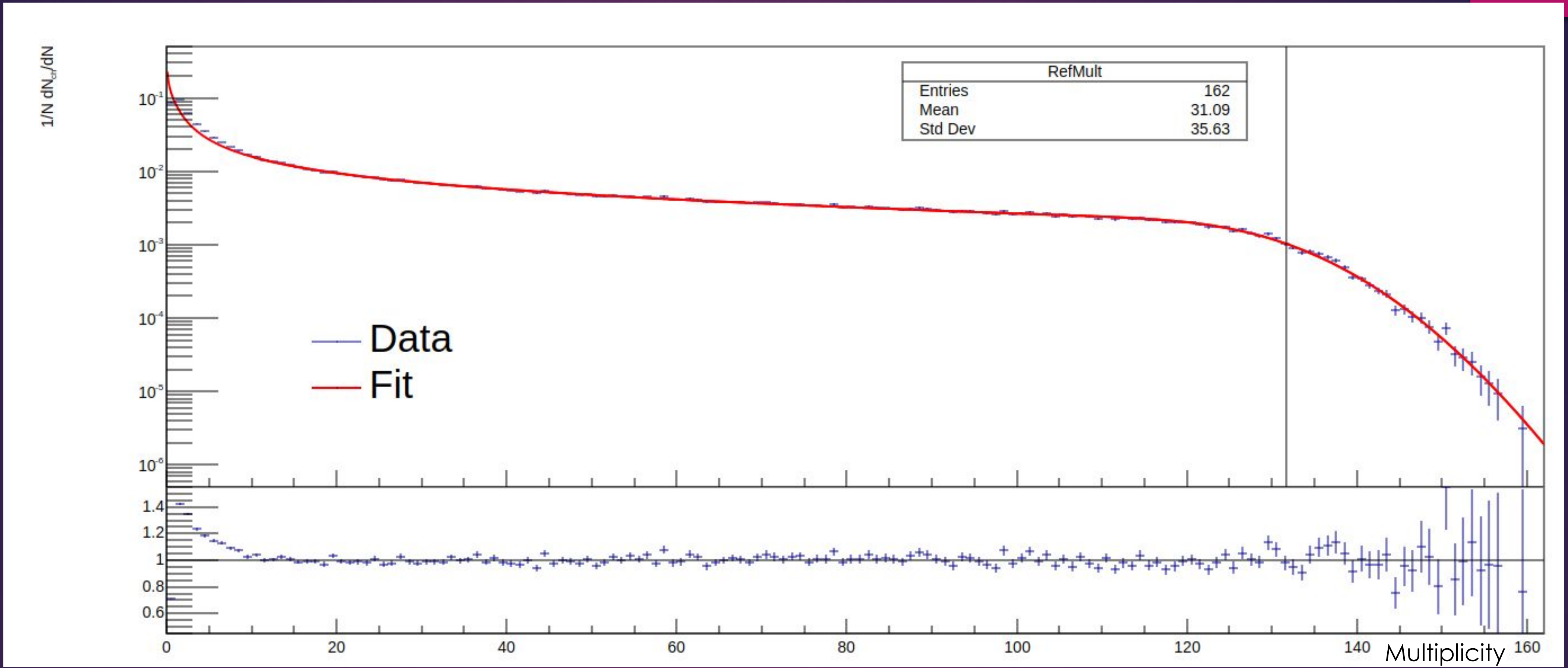
```
source /cvmfs/nica.jinr.ru/sw/os/login.sh
module add ROOT/v6.32.02-1
module add OpenSSL/v3.2.1-1
module add CMake/v3.28.3-1
```

The two macros are then executed in the same order as shown

```
root GammaFit2.C++'("./mult.root","hRefMultSTAR","./out.root",10,"hBvsRefMult")'
root printFinal.C'("./out.root","out.C")'
```



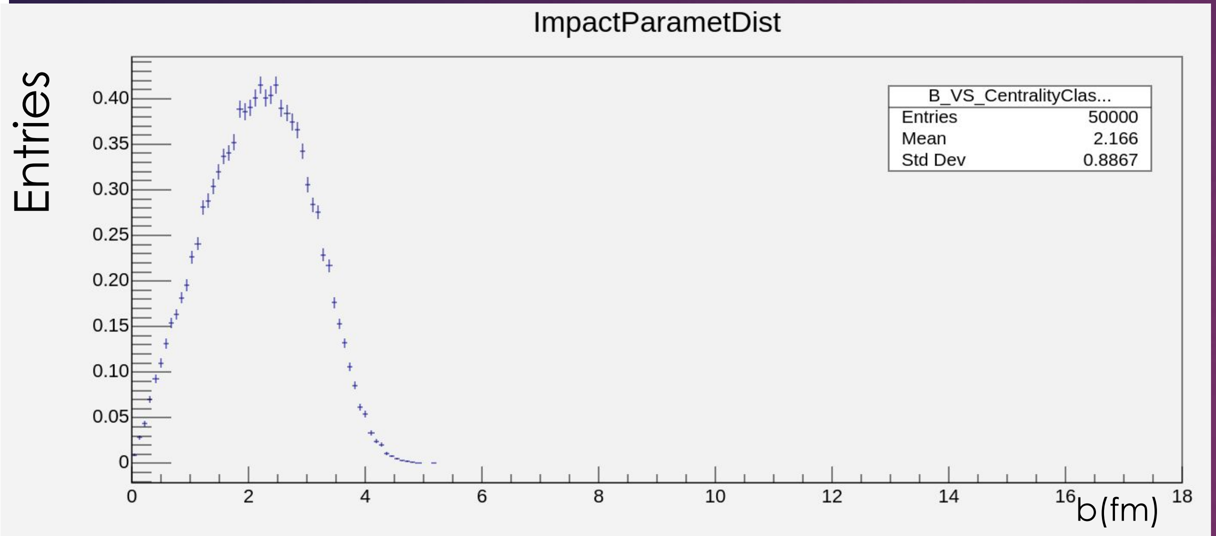
MULTIPLICITY



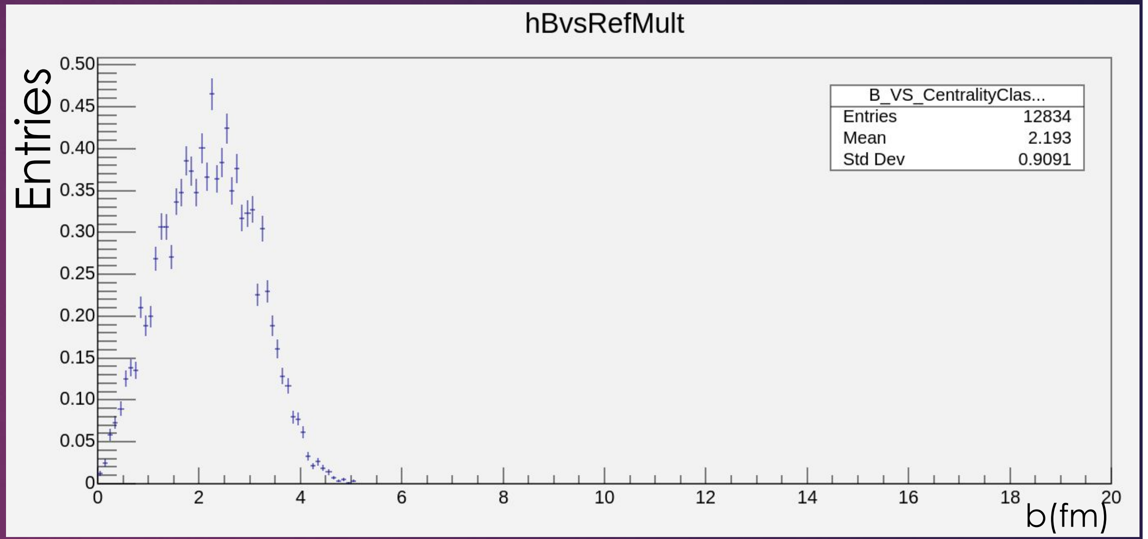
RefMult
Entries 162
Mean 31.09
Std Dev 35.63
Data
Fit
1/N dN_ch/dN
Multiplicity

FIT

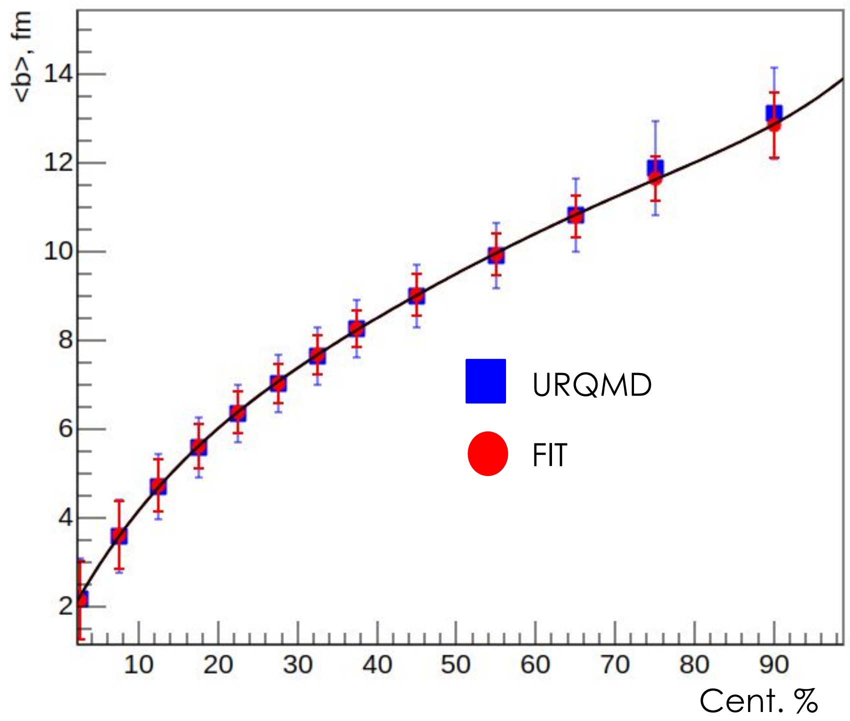
Impact parameter the URQMD

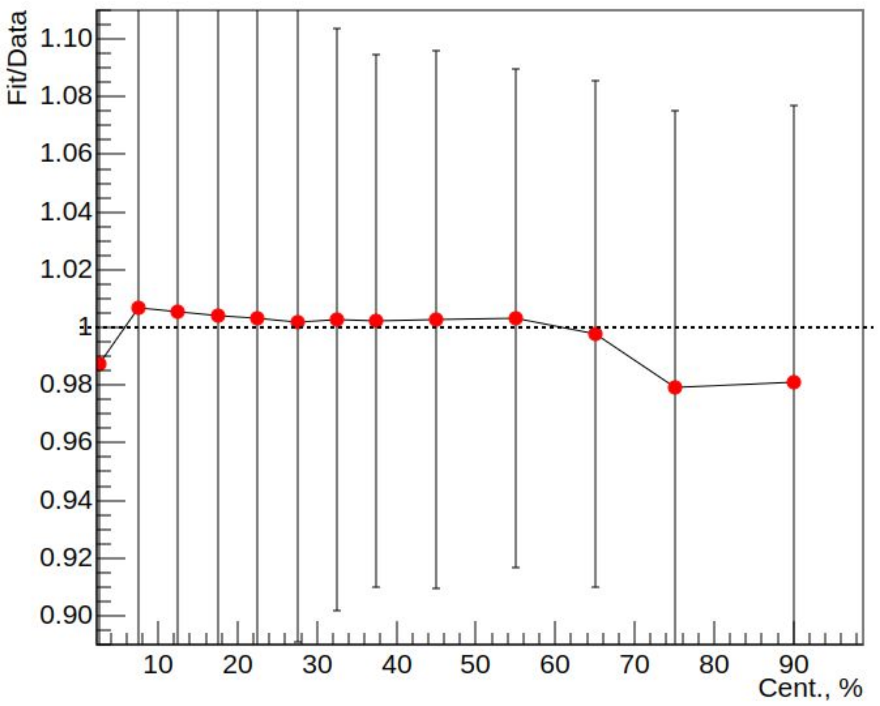


Impact parameter Fit



ADJUSTMENT BY EXTRAPOLATION OF THE IMPACT PARAMETER for 0-10% centrality





IMPACT PARAMETER AS MULTIPLICITY FUNCTION AND COMPARISON OF THE QUALITY OF METHOD ADJUSTMENT

The method gives us different parameters adjusted from the multiplicity

Parameters	Values
NDF	143
a1	-3.73
a2	0.164
a3	-2.84
chi^2	163.5
n_knee	131.7
sigma	679.9
teta	0.647
chi^2/NDF	1.14311

Parameters

Centralidad %	Multiplicity	Impact parameter (fm)
0-10	162-96	2.94-5.18
10-20	96-70	5.18-6.75
20-30	70-50	6.75-7.39
30-40	50-34	7.39-7.97
40-50	34-23	7.97-9.51
50-60	23-15	9.51-10.42
60-70	15-9	10.42-11.24
70-80	9-5	11.24-12.01
80-100	5-1	12.01-14.10

FIT

running the printFinal.C macro we can have centrality information

root printFinal.C'("./out.root","out.C")'

```
GNU nano 2.3.1                         File: printFinal.C

    std::cout << "File: " << inFileName.Data() << "." << std::endl;
    std::cout << std::endl;
    std::cout << "Cent, %   | Mult_min | Mult_max | <b>, fm |   RMS   | bmin, fm | bmax, fm |" << std::endl;
    std::cout << "----------|----------|----------|---------|---------|----------|----------|" << std::endl;
    for (int i = 0; i < NreasonableClasses; i++)
    {
        std::cout << Form("%3.0f - %3.0f | %8i | %8i | %7.2f | %7.2f | %8.2f | %8.2f |",
                          vCent.at(i).first, vCent.at(i).second,
                          vBorders.at(i).first, vBorders.at(i).second,
                          vBavg.at(i), vBavgRMS.at(i), vBimp.at(i).first, vBimp.at(i).second)
                  << std::endl;
    }
    std::cout << "----------------------------------------------------------------------------" << std::endl;
    std::cout << std::endl;
    std::cout << "  teta  | n_knee |   a1   |   a2   |   a3   |  chi2  |   NDF   | minNch |" << std::endl;
    std::cout << "--------|--------|--------|--------|--------|--------|---------|--------|" << std::endl;

    std::cout << Form(" %6.2f | %6.2f | %6.2f | %6.2f | %6.2f | %6.2f | %7.2f | %6i |", teta, n_knee, a1, a2, a3, chi2, NDF, minNch) << std::endl
              << std::endl;

    std::cout << std::endl;
    std::cout << "const float BinN[Nb+1]={";
    for (int i = 0; i < NreasonableClasses; i++)
    {
        if(i==NreasonableClasses-1)
        std::cout << vBorders.at(i).first << "";
        else
        std::cout << vBorders.at(i).first << ",";
    }
    std::cout << "};"<< std::endl;

    std::cout << "const float Binb[Nb+1]={";
    for (int i = 0; i < NreasonableClasses; i++)
    {
        if(i==NreasonableClasses-1)
        std::cout << Form("%4.2f",vBimp.at(i).second )<< "";
        else
        std::cout << Form("%4.2f",vBimp.at(i).second ) << ",";
    }
    std::cout << "};"<< std::endl;;
}
```

```
Centrality classes in multiplicity
0% - 5%, 162 - 113 ,centrality 0 - 0.0485608
5% - 10%, 113 - 96 ,centrality 0.0485608 - 0.0997122
10% - 15%, 96 - 82 ,centrality 0.0997122 - 0.148673
15% - 20%, 82 - 70 ,centrality 0.148673 - 0.196858
20% - 25%, 70 - 59 ,centrality 0.196858 - 0.247524
25% - 30%, 59 - 50 ,centrality 0.247524 - 0.294965
30% - 35%, 50 - 41 ,centrality 0.294965 - 0.349558
35% - 40%, 41 - 34 ,centrality 0.349558 - 0.398681
40% - 50%, 34 - 23 ,centrality 0.398681 - 0.493871
50% - 60%, 23 - 15 ,centrality 0.493871 - 0.587373
60% - 70%, 15 - 9 ,centrality 0.587373 - 0.68661
70% - 80%, 9 - 5 ,centrality 0.68661 - 0.786968
80% - 100%, 5 - 1 ,centrality 0.786968 - 1

Centrality classes in impact parameter
Centr. class, bmin-bmax , bmean
0% - 5%, 0 - 2.94487 fm, 2.21777 fm
5% - 10%, 2.94487 - 4.18066 fm, 3.59577 fm
10% - 15%, 4.18066 - 5.18802 fm, 4.70868 fm
15% - 20%, 5.18802 - 6.02882 fm, 5.62594 fm
20% - 25%, 6.02882 - 6.75104 fm, 6.40225 fm
25% - 30%, 6.75104 - 7.39056 fm, 7.07932 fm
30% - 35%, 7.39056 - 7.97294 fm, 7.68765 fm
35% - 40%, 7.97294 - 8.51516 fm, 8.24829 fm
40% - 50%, 8.51516 - 9.51486 fm, 9.02741 fm
50% - 60%, 9.51486 - 10.4214 fm, 9.9794 fm
60% - 70%, 10.4214 - 11.2427 fm, 10.8416 fm
70% - 80%, 11.2427 - 12.0192 fm, 11.6312 fm
80% - 100%, 12.0192 - 14.1021 fm, 12.8805 fm
```

FIT

MC Glauber

How it works

MC Glauber Package

Fermi nuclear distribution

$$\rho(r) = \rho_0 \frac{1 + w\left(\frac{r}{R}\right)^2}{1 + \exp\frac{r-R}{a}},$$

a is the skin thickness

R is radius of the nucleus

ρ_0 is density in the center of the nucleus

Collision parameter

$$d < \sqrt{\frac{\sigma_{NN}^{inel}}{\pi}}.$$

Inelastic cross section Nucleon-Nucleon : σ_{NN}^{inel}

How it works

Centrality Framework

Negative Binomial Distribution

$$P_{\mu,k}(n) = \frac{\Gamma(n+k)}{\Gamma(n+1)\Gamma(k)} \cdot \frac{(\mu/k)^n}{(\mu/k+1)^{n+k}}.$$

Number of ancestors

$$N_a(f) = fN_{part} + (1-f)N_{coll}$$

$$\frac{d\sigma}{dN_{ch}} \equiv F_{fit}(f,\mu,k) = N_a(f) \times P_{\mu,k}(N_{ch})$$

$\frac{d\sigma}{dN_{ch}}$ - Probability of an inelastic collision at a given N_{ch}

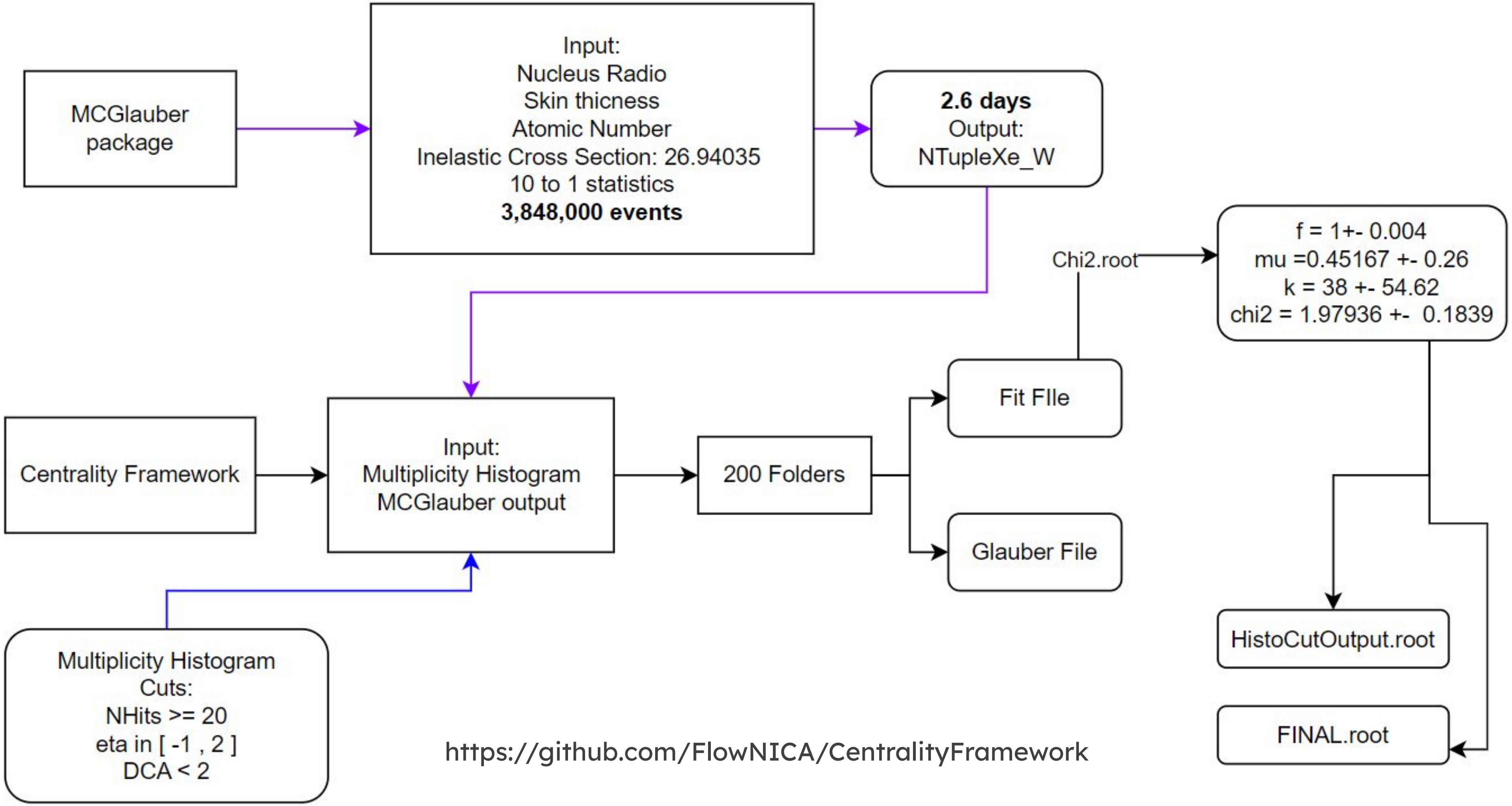
f - Fraction of production from soft component
μ - Mean multiplicity value
k - Width of multiplicity distribution

How it works

Formula for centrality

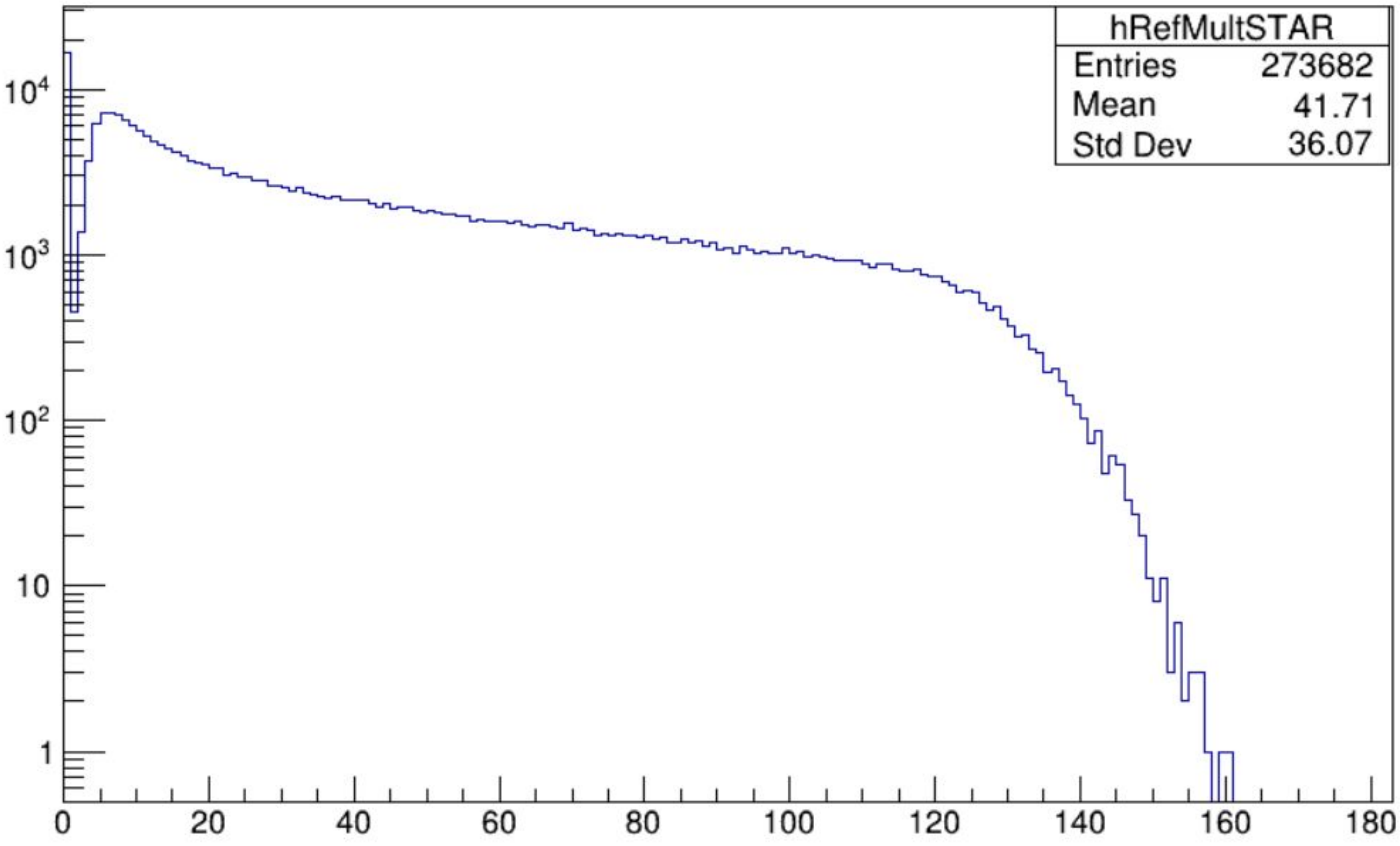
$$c_{N_{ch}} = \frac{1}{\sigma_{inel}} \int_{N_{ch}}^{\infty} \frac{d\sigma}{dN_{ch}} dN_{ch},$$

How it's implemented



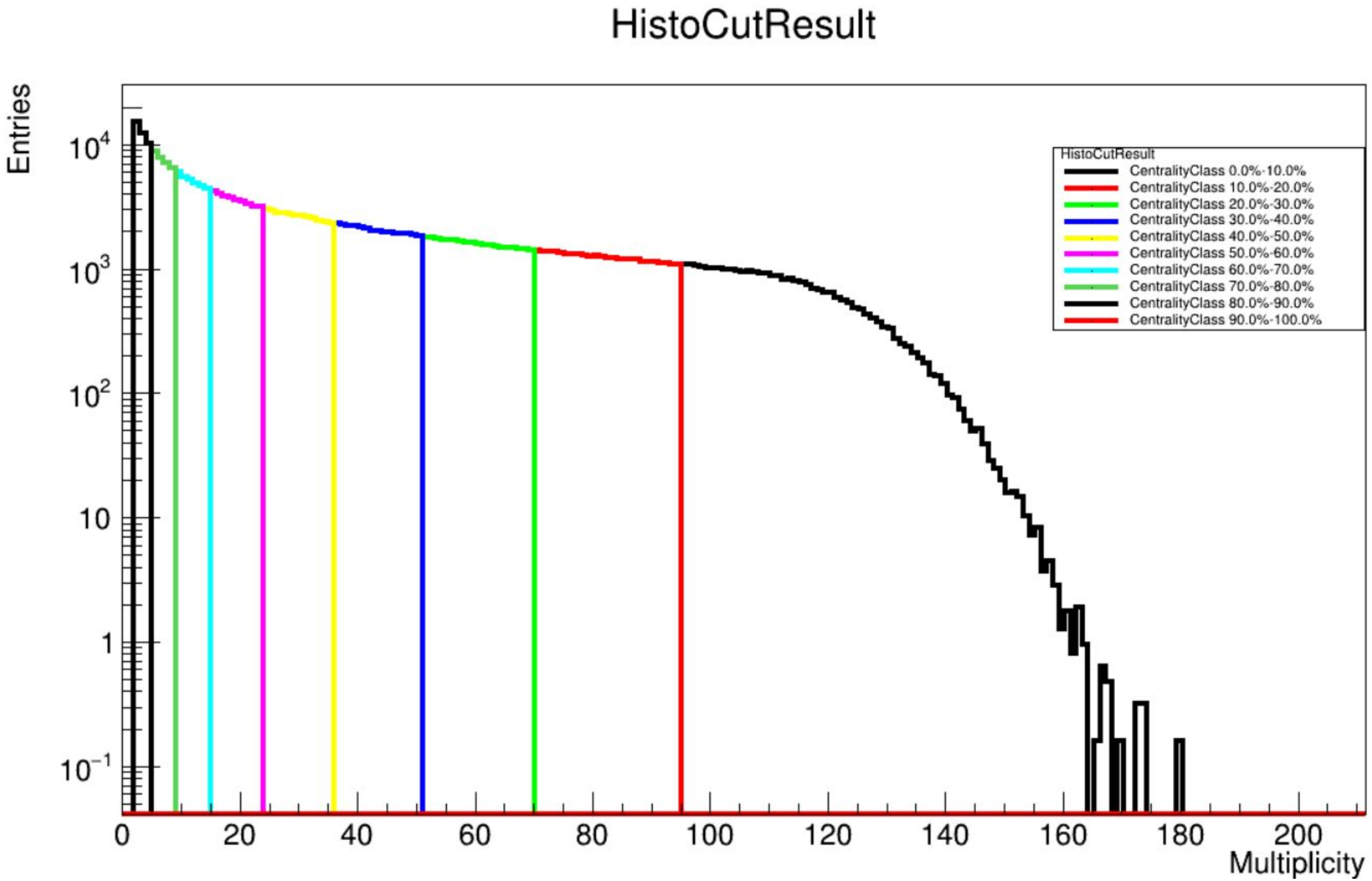
https://github.com/FlowNICA/CentralityFramework

Multiplicity



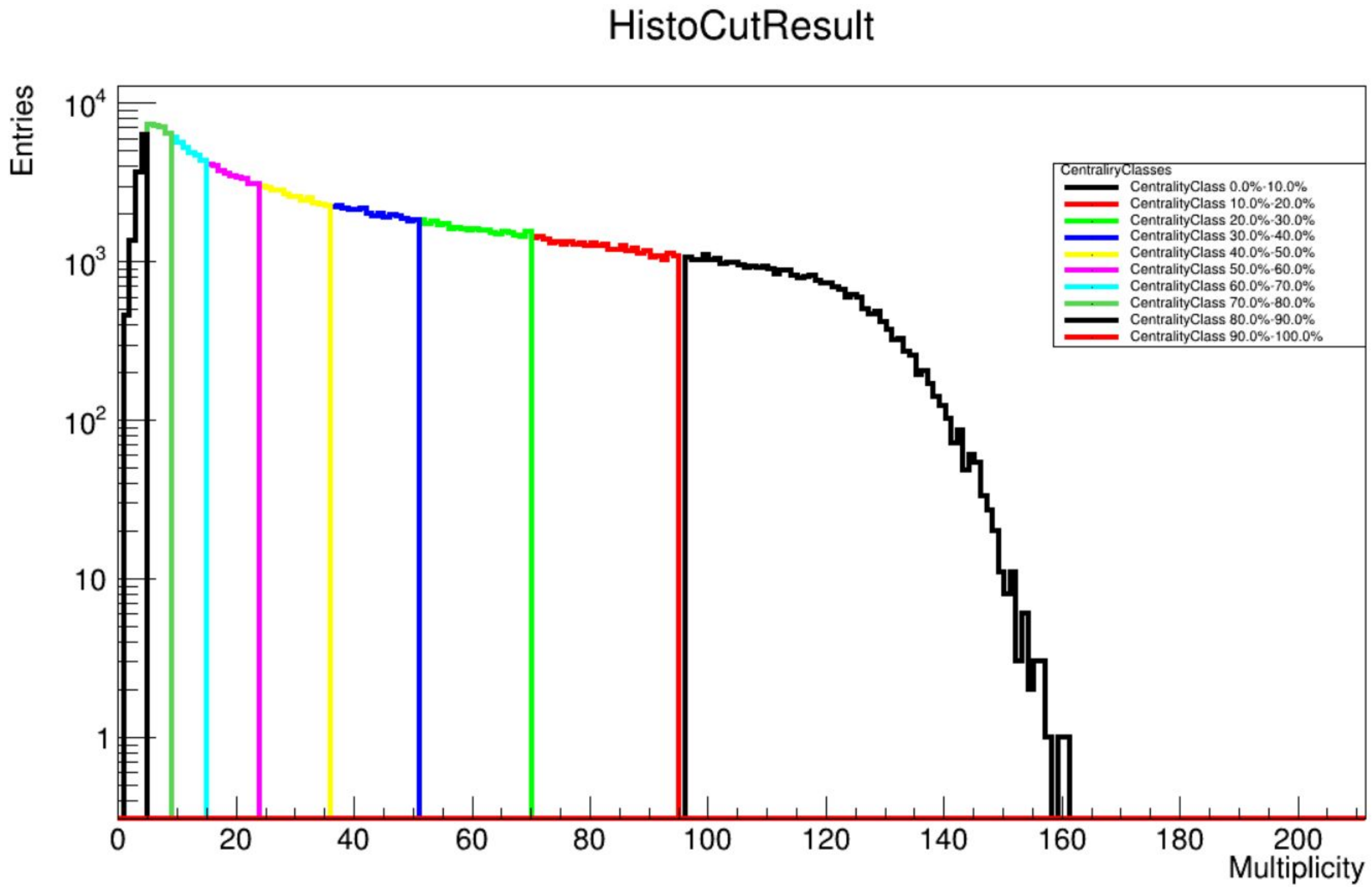
Cuts:

NHits >= 20
eta in [-1 , 2]
DCA < 2



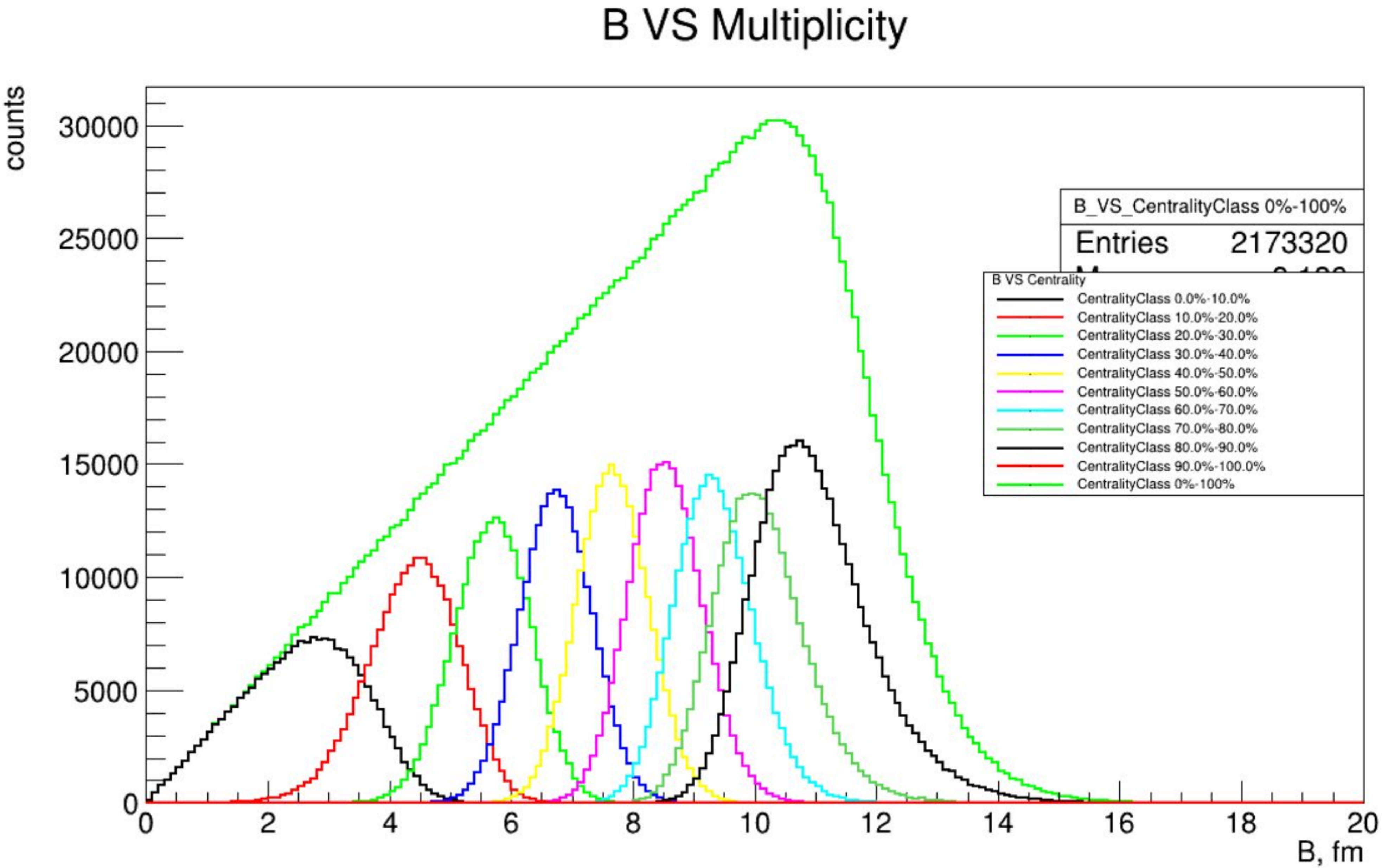
Cuts:

NHits >= 20
eta in [-1 , 2]
DCA < 2



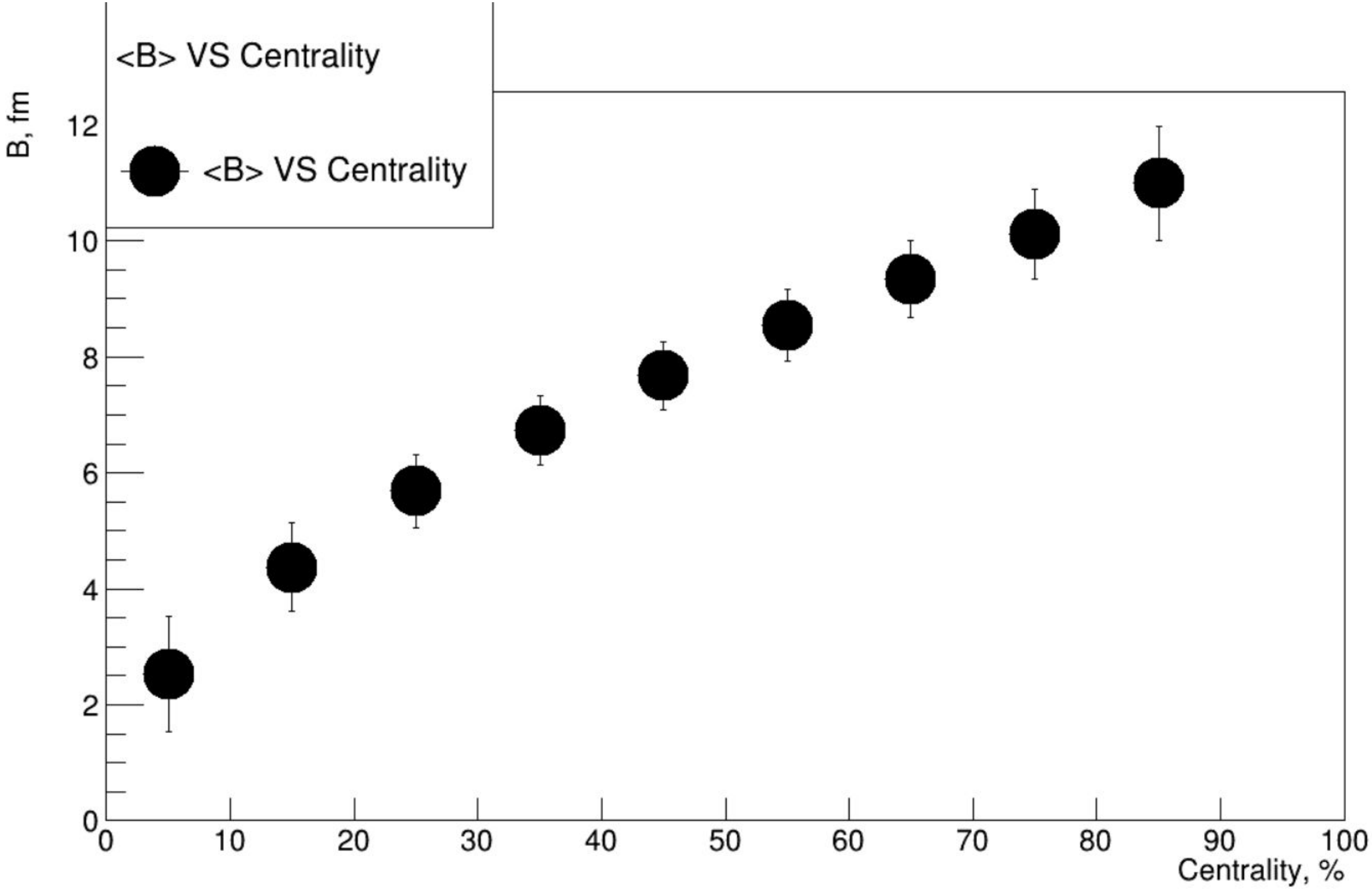
Cuts:

NHits >= 20
eta in [-1 , 2]
DCA < 2

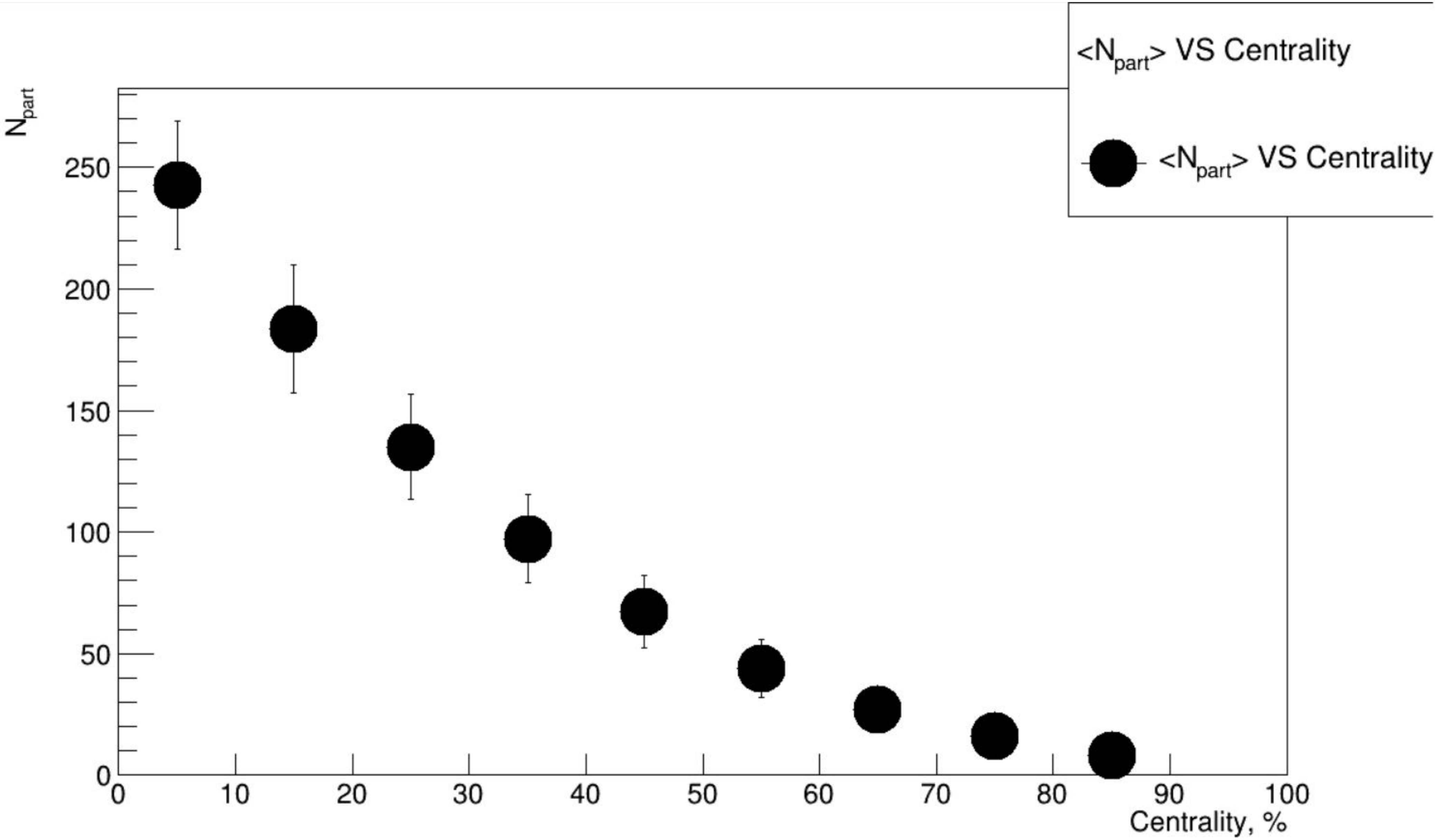


Cuts:

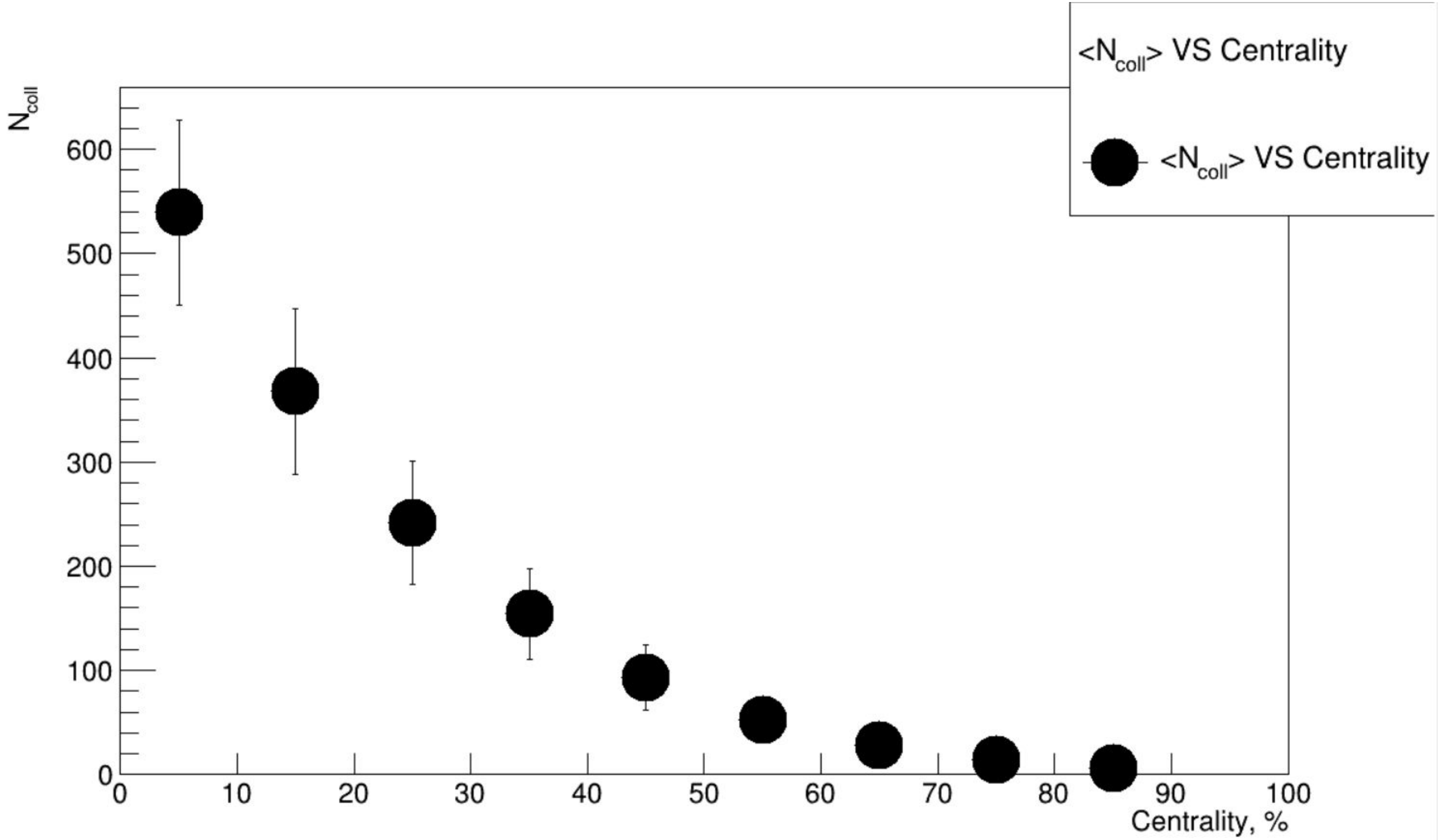
NHits >= 20
eta in [-1 , 2]
DCA < 2



<B> VS Centrality
<B> VS Centrality
B, fm
0
2
4
6
8
10
12
0
10
20
30
40
50
60
70
80
90
100
Centrality, %



<N_part> VS Centrality
<N_part> VS Centrality
N_part
250
200
150
100
50
0
0
10
20
30
40
50
60
70
80
90
100
Centrality, %



<N_coll> VS Centrality
<N_coll> VS Centrality
N_coll
0
100
200
300
400
500
600
0
10
20
30
40
50
60
70
80
90
100
Centrality, %

MC Glauber Fit

Centrality, %	N_{ch}^{min}	N_{ch}^{max}	$\langle b \rangle$, fm	RMS	$\langle N_{part} \rangle$	RMS	$\langle N_{coll} \rangle$	RMS
0 - 10	95	164	2.54	1.00	242.75	26.44	539.73	88.76
10 - 20	70	95	4.37	0.76	183.50	26.19	367.84	79.38
20 - 30	51	70	5.69	0.63	134.96	21.54	241.37	59.12
30 - 40	36	51	6.74	0.59	97.33	17.96	154.42	43.75
40 - 50	24	36	7.68	0.59	67.38	14.84	93.66	31.25
50 - 60	15	24	8.55	0.61	43.96	11.82	52.81	20.86
60 - 70	9	15	9.35	0.66	27.27	9.01	28.33	13.12
70 - 80	5	9	10.11	0.78	16.04	6.73	14.59	8.12
80 - 90	2	5	10.99	0.99	8.23	4.74	6.60	4.77

MCGlauber vs Gamma Fit

MCGlauber **Gamma Fit**

Centrality, %	N_{ch}^{min}	N_{ch}^{max}	N_{ch}^{min}	N_{ch}^{max}	ΔN_{ch}^{min}	ΔN_{ch}^{max}
0 - 10	95	164	96	162	1	2
10 - 20	70	95	70	96	0	1
20 - 30	51	70	50	70	1	0
30 - 40	36	51	34	50	2	1
40 - 50	24	36	33	34	9	2
50 - 60	15	24	15	23	0	1
60 - 70	9	15	9	15	0	0
70 - 80	5	9	5	9	0	0
80 - 90	2	5	1	5	1	0

MCGlauber vs Gamma Fit

	MCGlauber	**Gamma Fit**
Centrality %	<b>	<b>
0 - 10	2.54	2.21
10 - 20	4.37	4.7
20 - 30	5.69	5.62
30 - 40	6.74	6.4
40 - 50	7.68	7.68
50 - 60	8.55	8.24
60 - 70	9.35	9.02
70 - 80	10.11	10.84
80 - 90	10.99	11.63

Resume and Future Work

Two methods for extraction of centrality from multiplicity are exposed:

GammaFit method with chi^2/NDF = 1.14

MCGlauber method with a chi^2/NDF = 1.97

GammaFit is apparently faster and with better settings but the MCGlauber gives more information.

Future work:

Use 4 centrality bins for an analysis of reconstruction efficiency for pt and resolutions by particle species