MPDROOT

Second part

INTEREST TEAM:
Flores Aguilar Adolfo
Márquez Ramírez Juan Carlos
Reyes Rodríguez José Francisco
San Juan López Alejandro:
alejandrosanjuan59@gmail.com

HISTOGRAMS

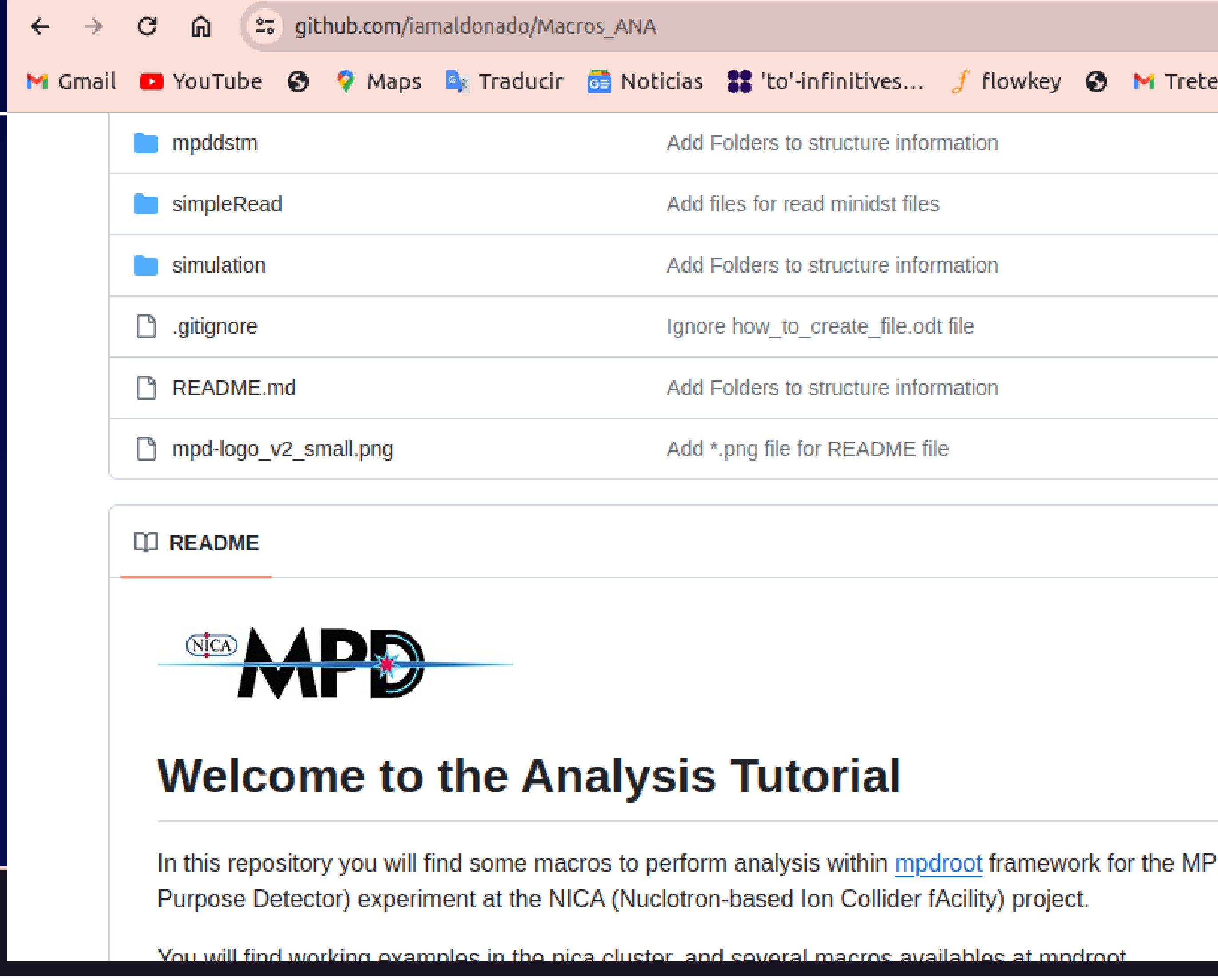
In a similar way to how the first histogram was obtained, others were also obtained using mpdroot and the github repository.

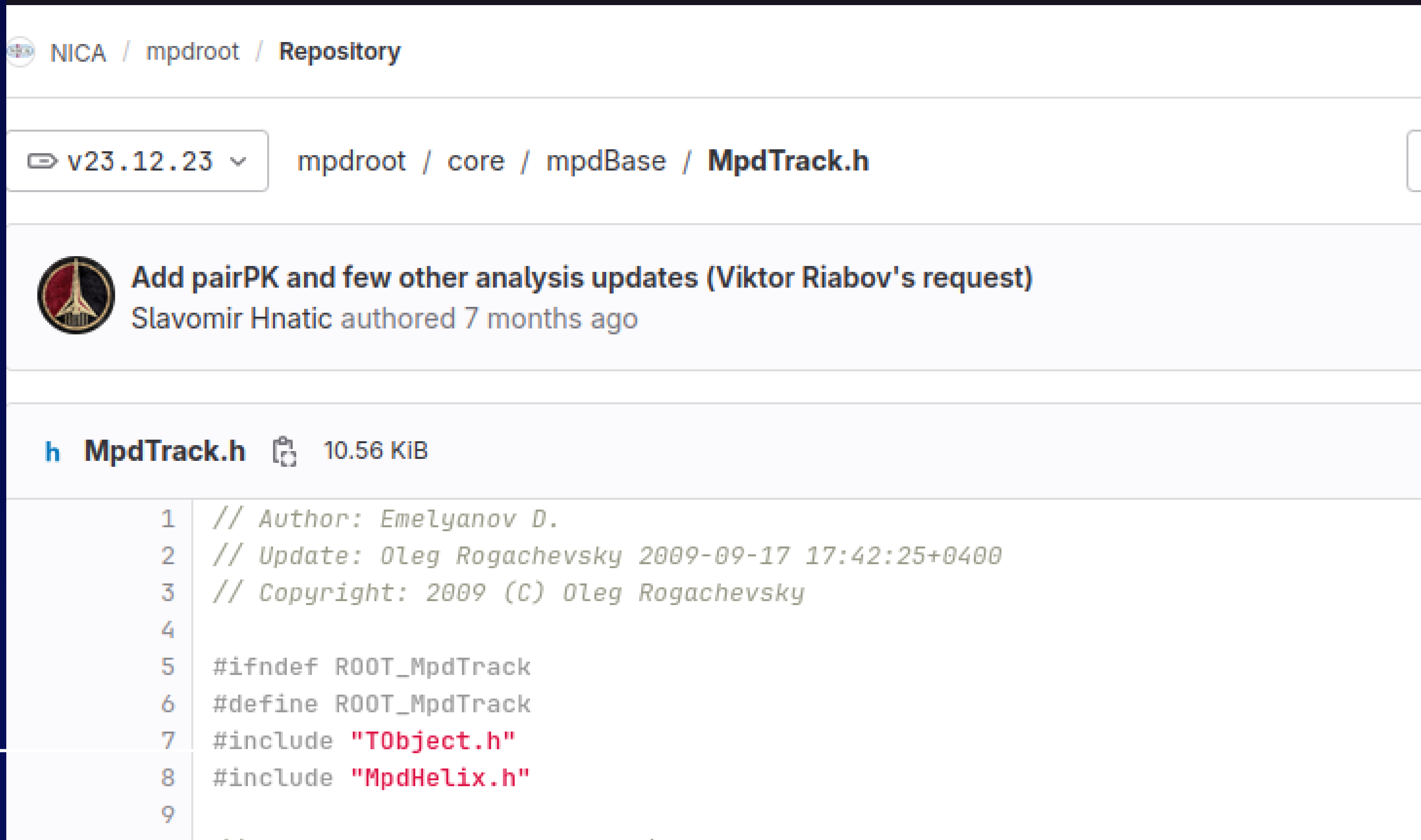
Github repository:

1. core/mpdBase/MpdTrack.h · v23.12.23 · NICA / mpdroot · GitLab. (2023, July 20). GitLab.https://git.jinr.ru/nica/mpdroot/-/blob/v23.12.23/core/mpdBase/MpdTrack.h?.ref_type=tags

2. Iamaldonado. (n.d.). Macros_ANA/simulation/transport/transport.md at main · iamaldonado/Macros_ANA. GitHub. https://github.com/iamaldonado/Macros_ANA/blob/main/simulation/transport/transport.md

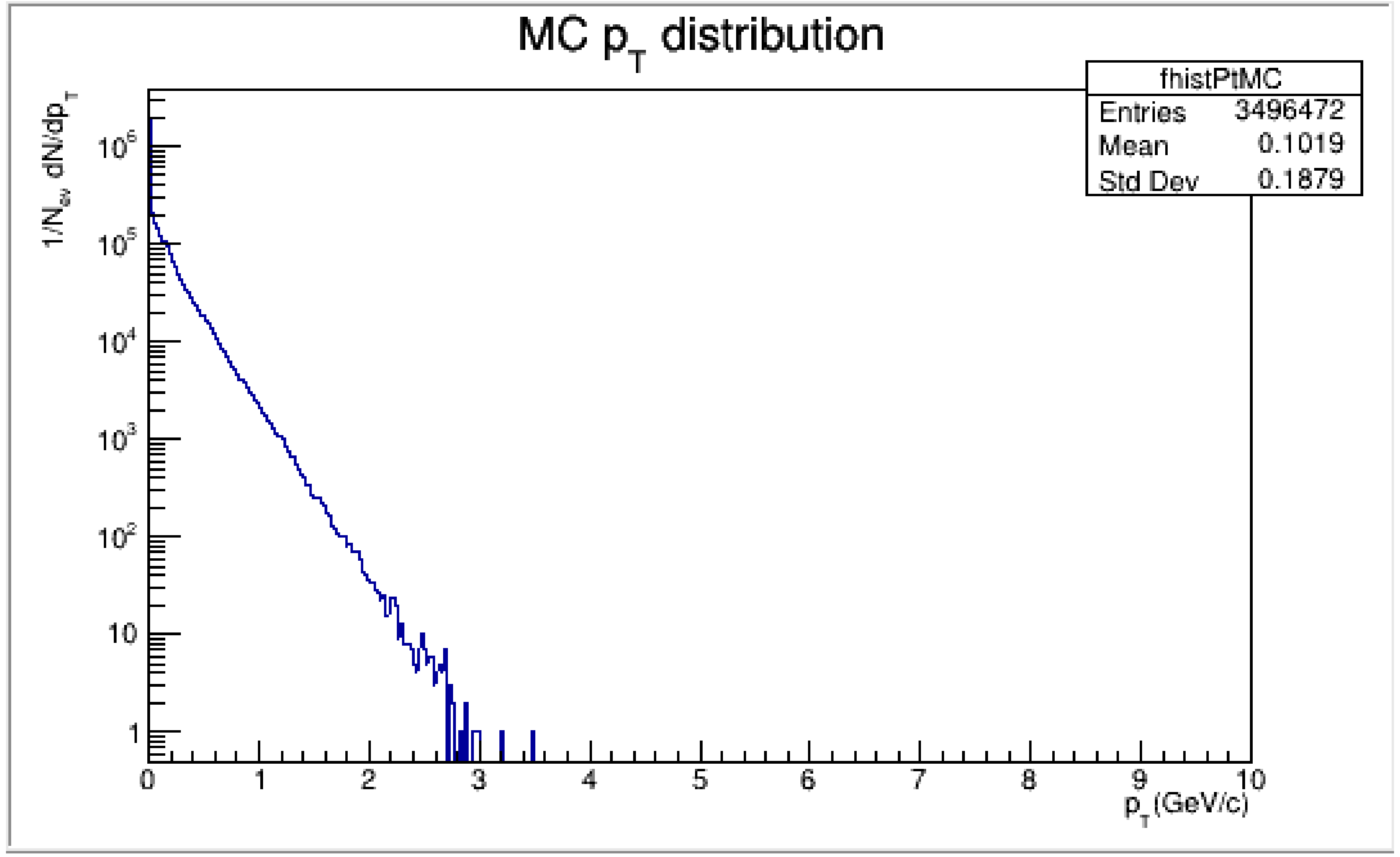
3.



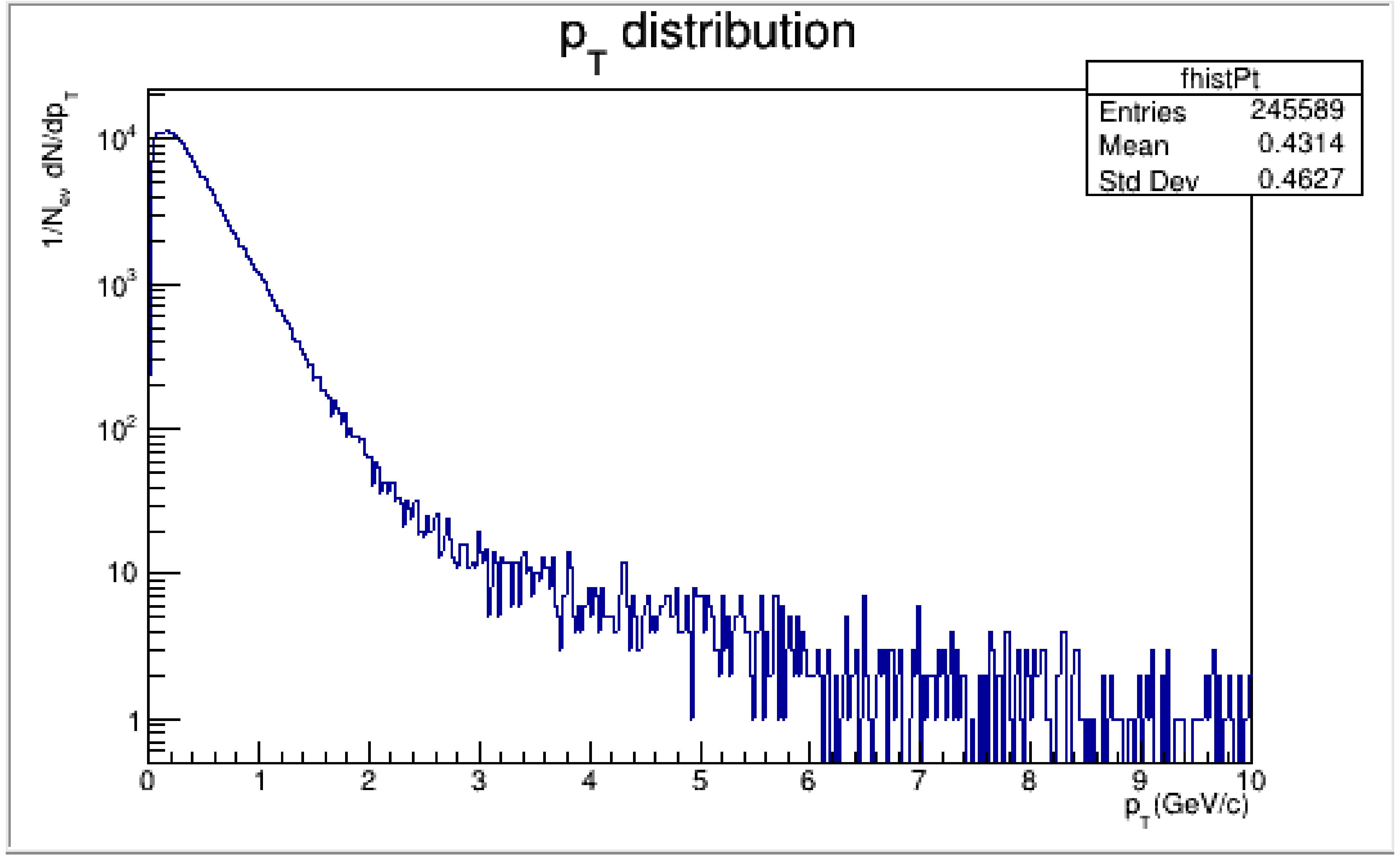


Transverse Moment Distribution (PT)

Generated tracks (MC)

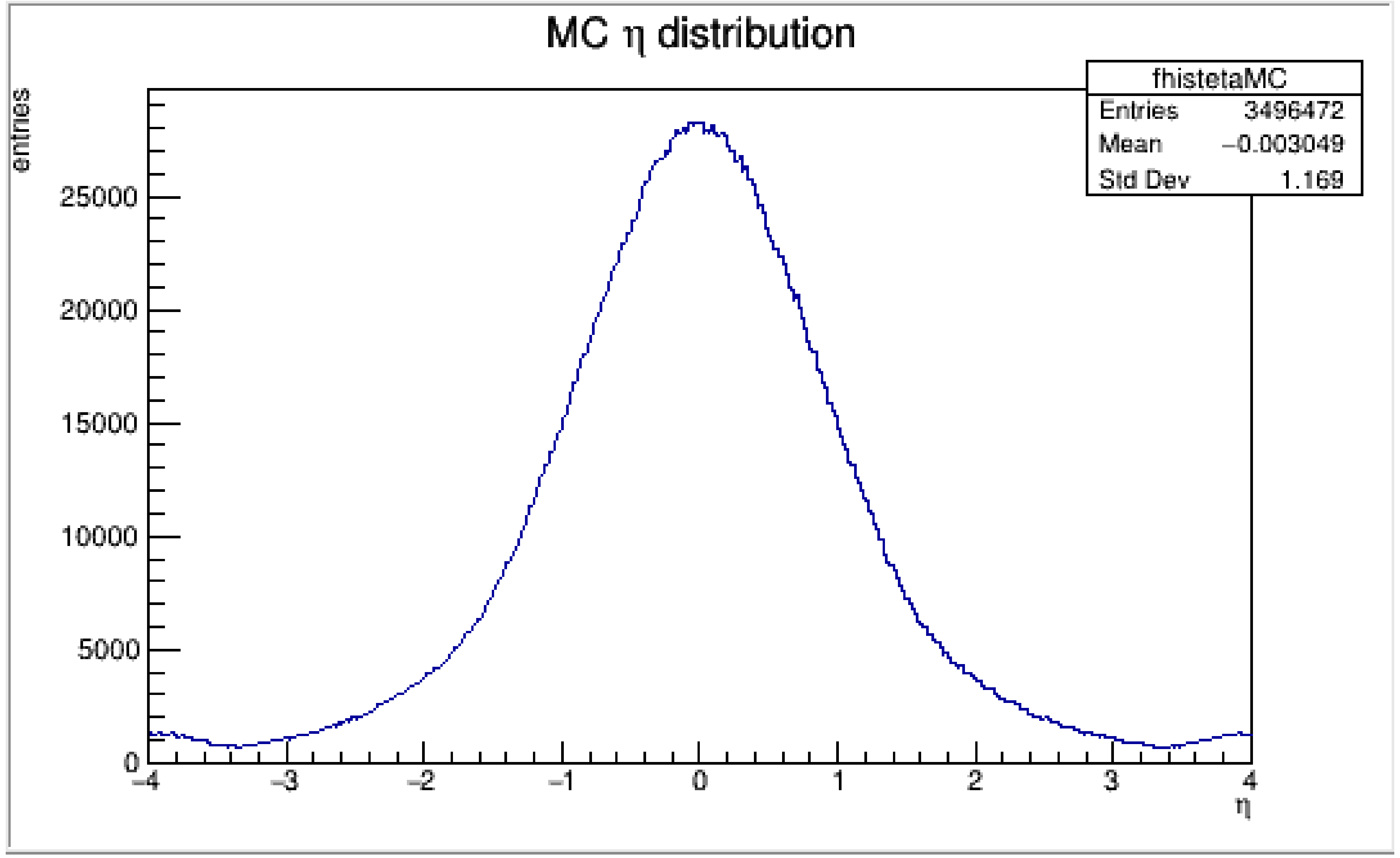


Reconstructed tracks (global tracks)

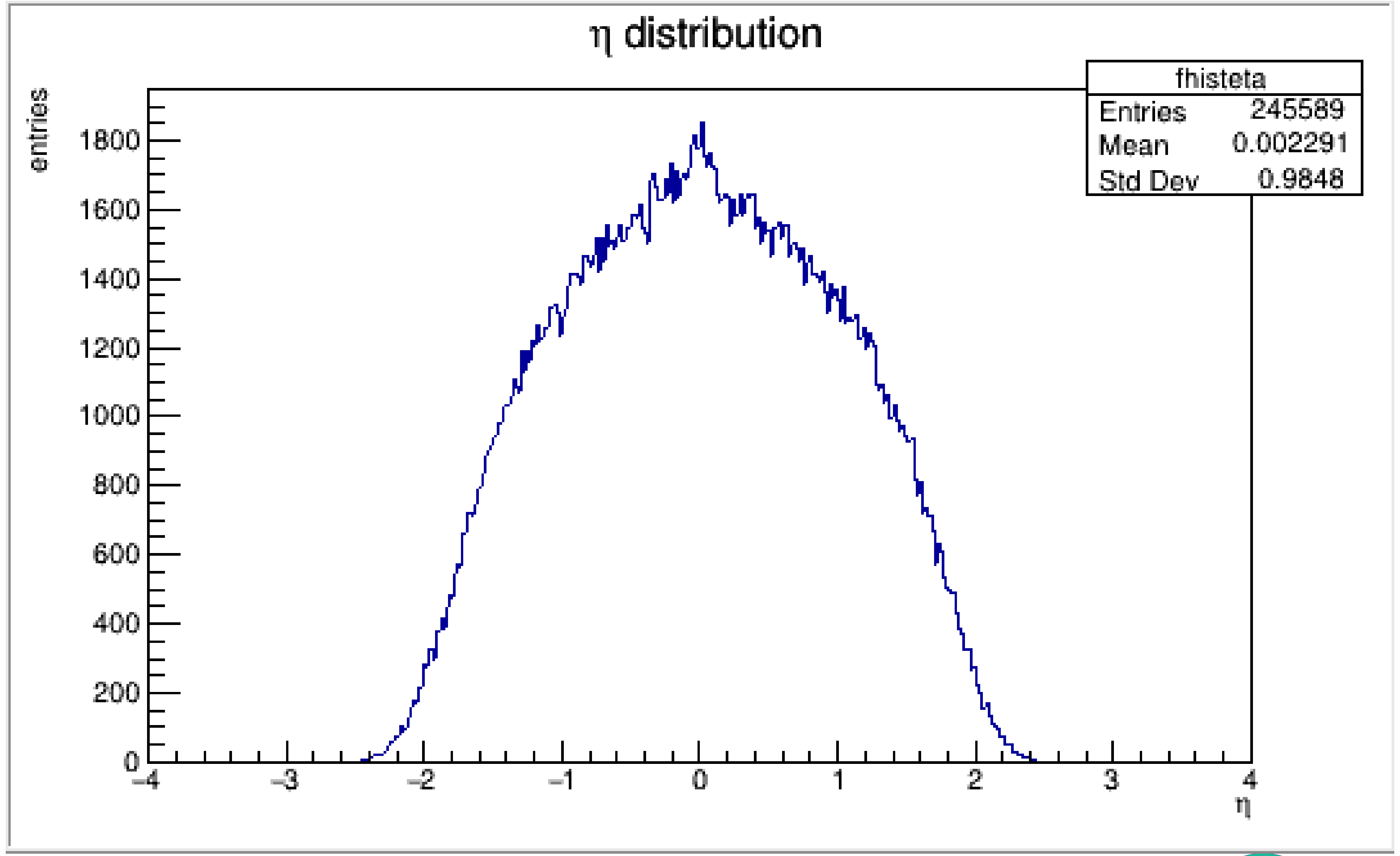


Pseudorapidity distribution (η)

Generated tracks (MC)



Reconstructed tracks (global tracks)



Difference between generated and reconstructed tracks

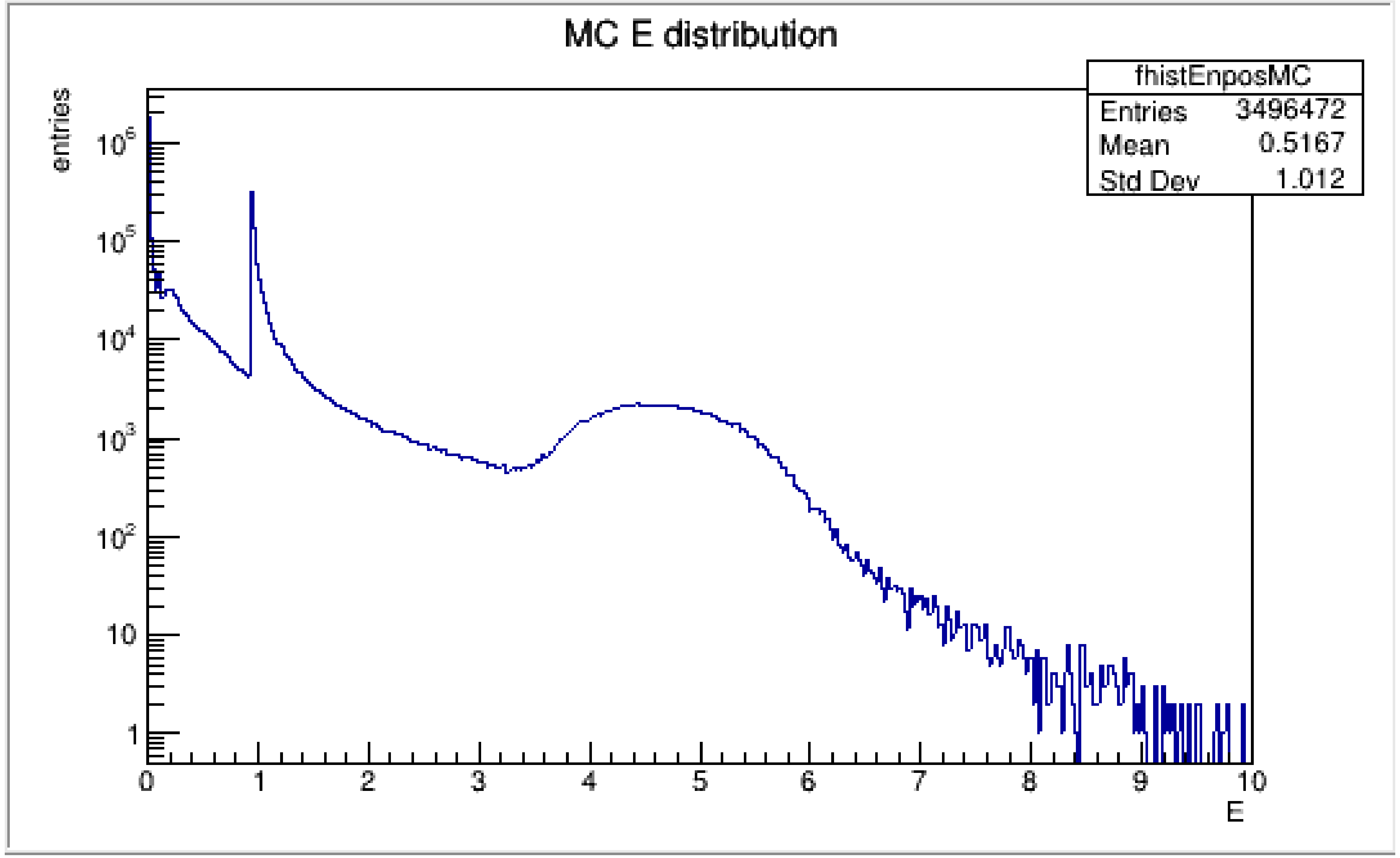
This is due to the acceptance

what is acceptance?
Acceptance is defined as the fraction of events that pass the requirements p_T and η

Acceptance corrections depend on the model and are taken into account in the systematic uncertainty.

Energy Distribution (E)

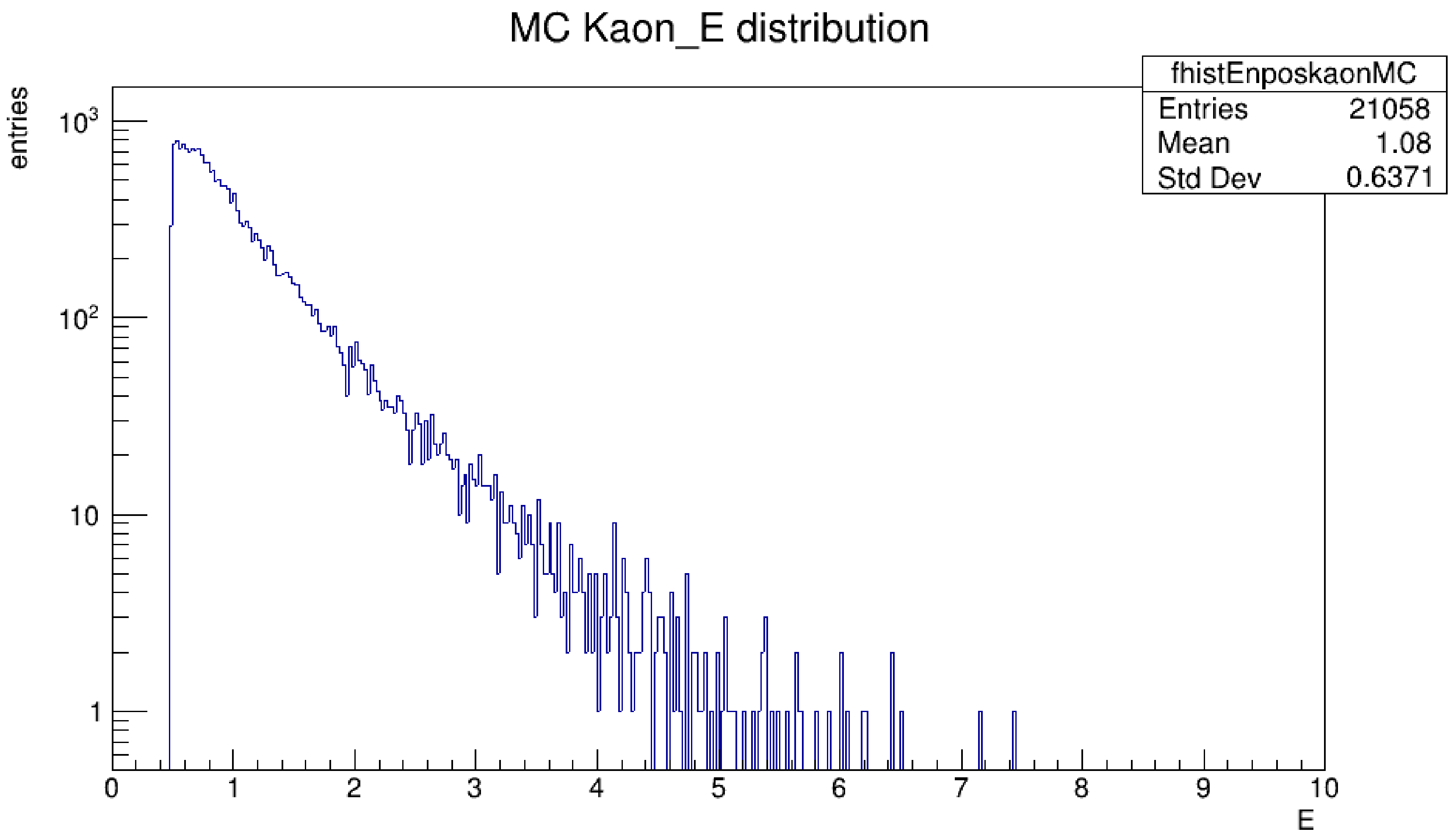
Generated tracks (MC)



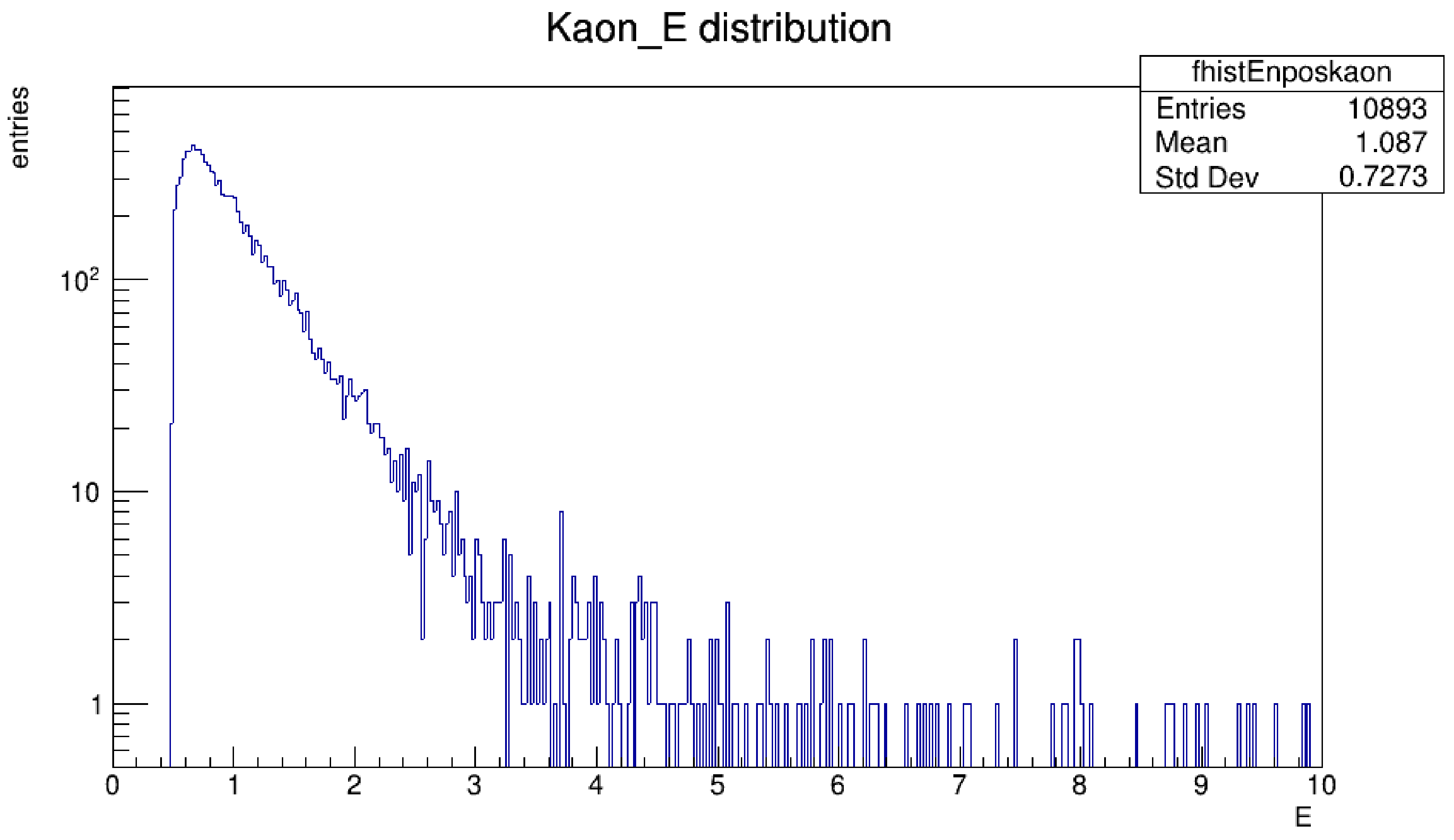
The distribution of energy for each particle after the collision is shown.

Kaon Energy Distribución

Generated tracks (MC)

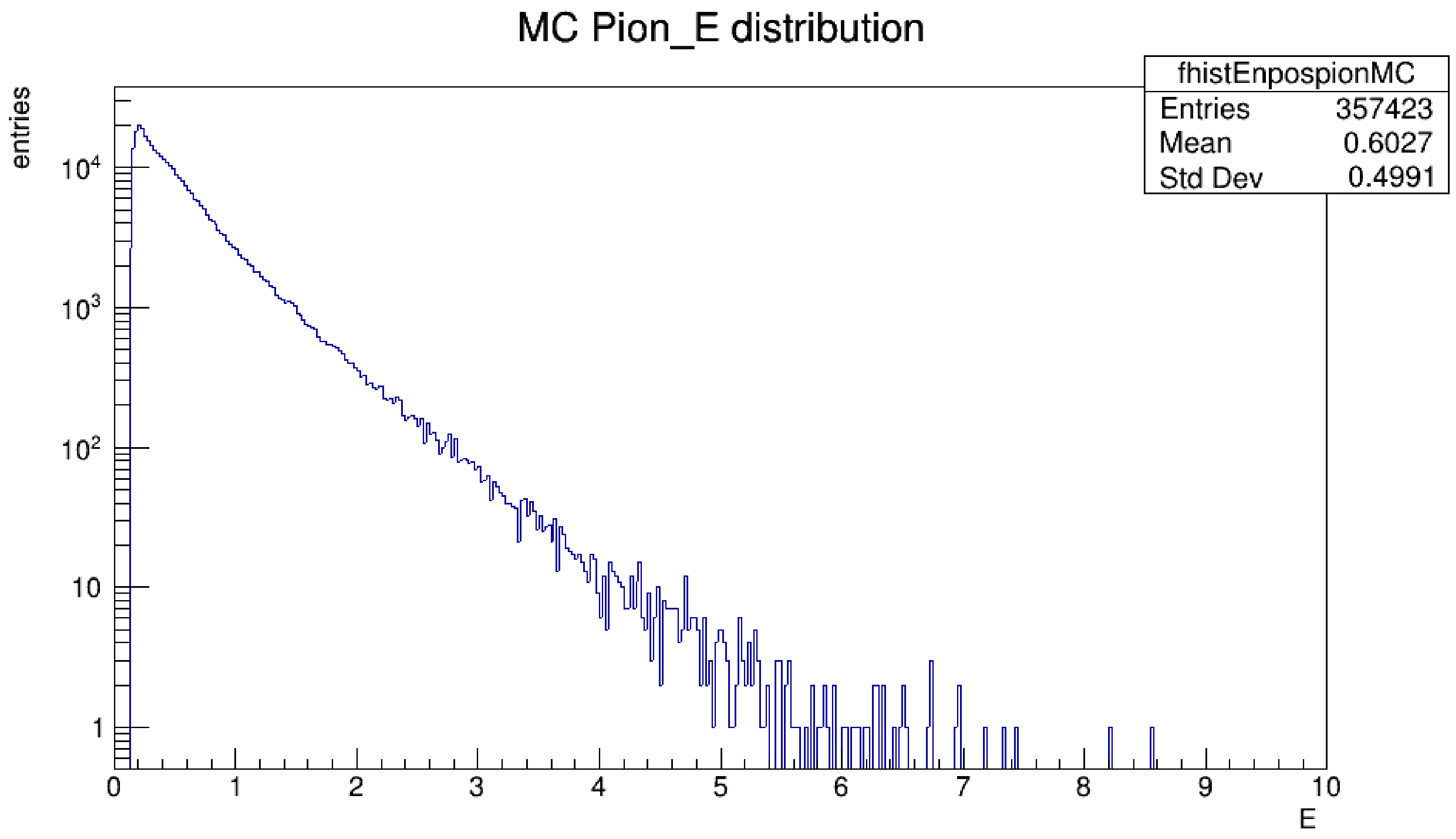


Reconstructed tracks (global tracks)

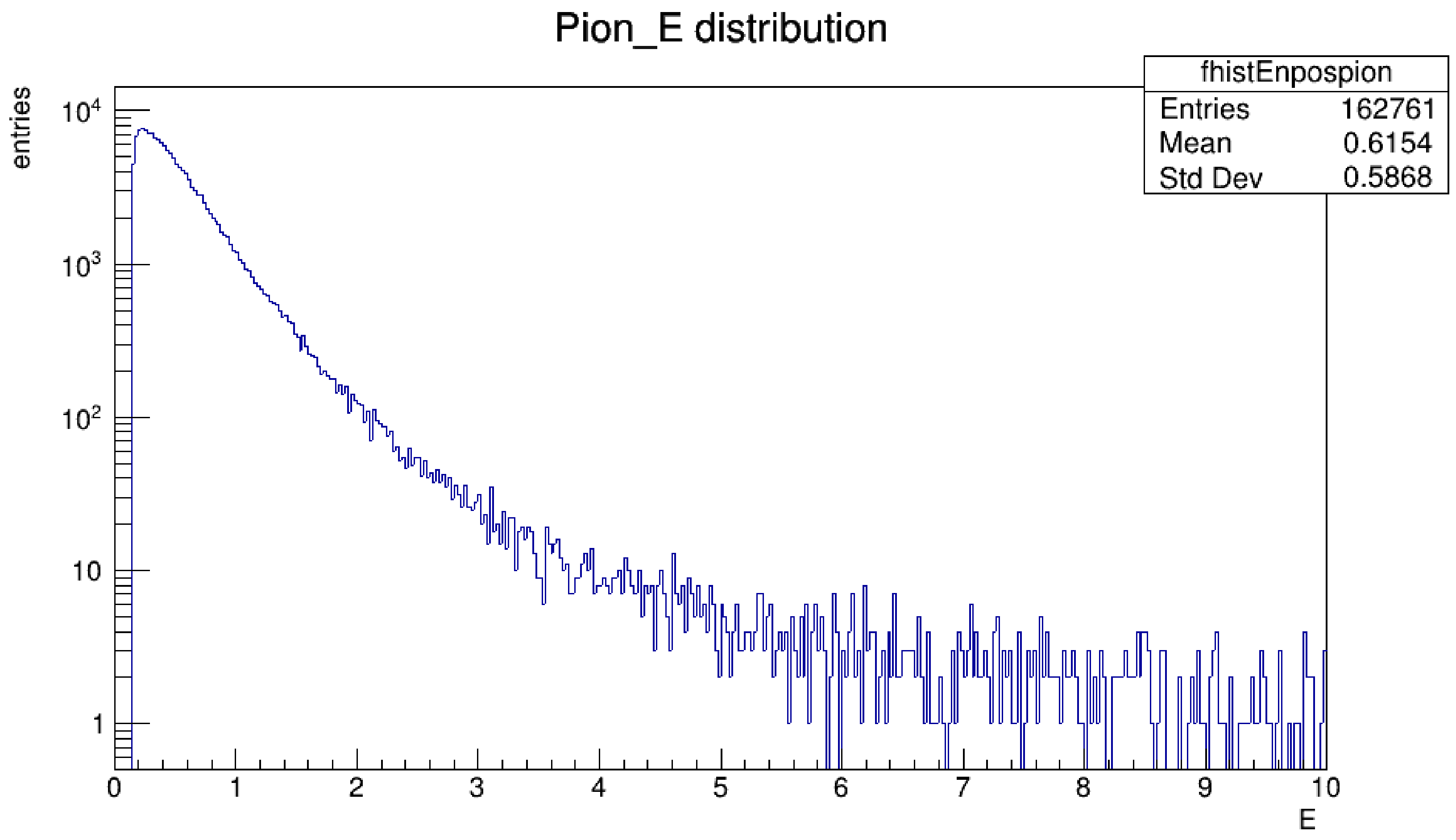


Pion Energy Distribución

Generated tracks (MC)

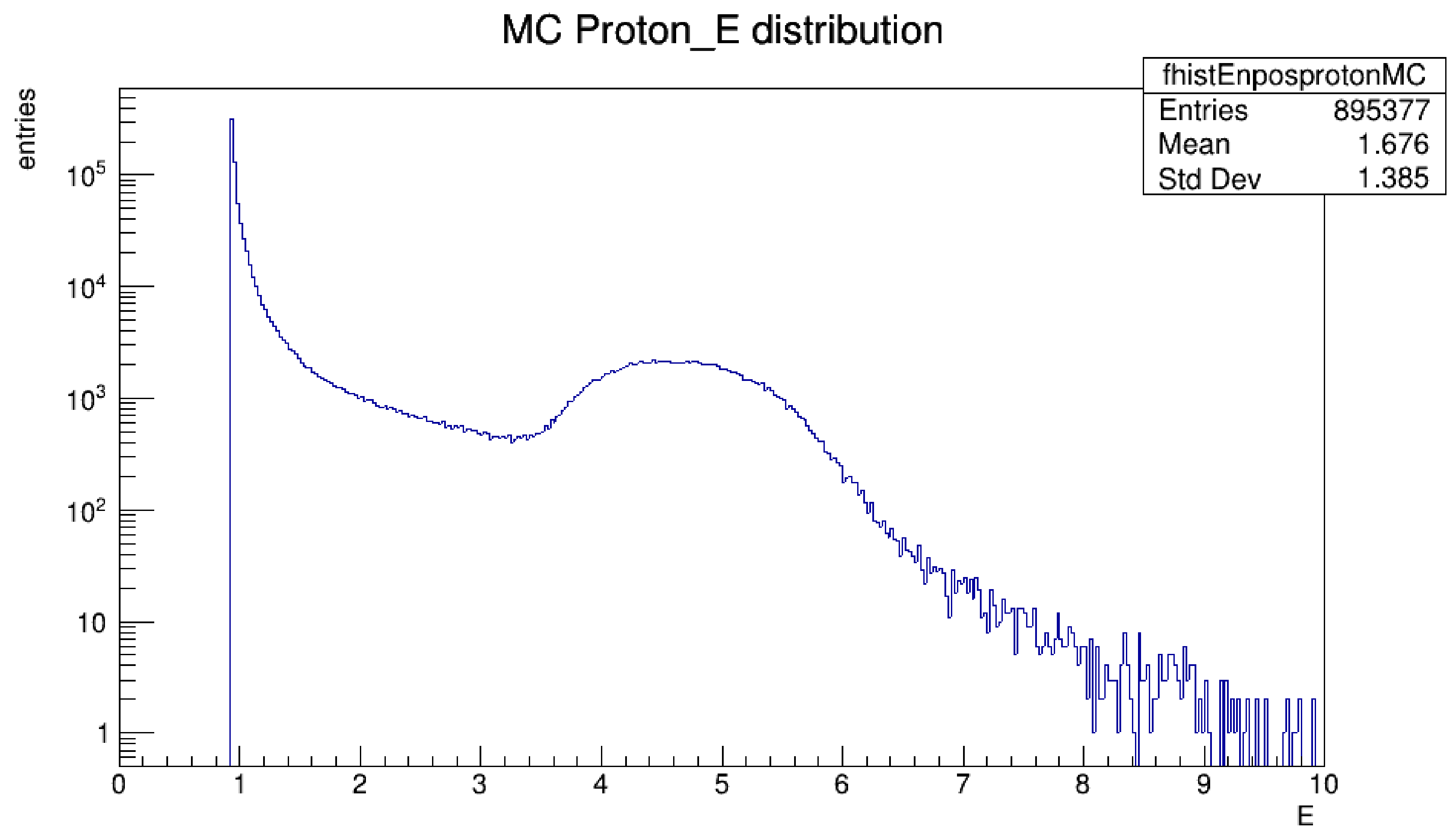


Reconstructed tracks (global tracks)

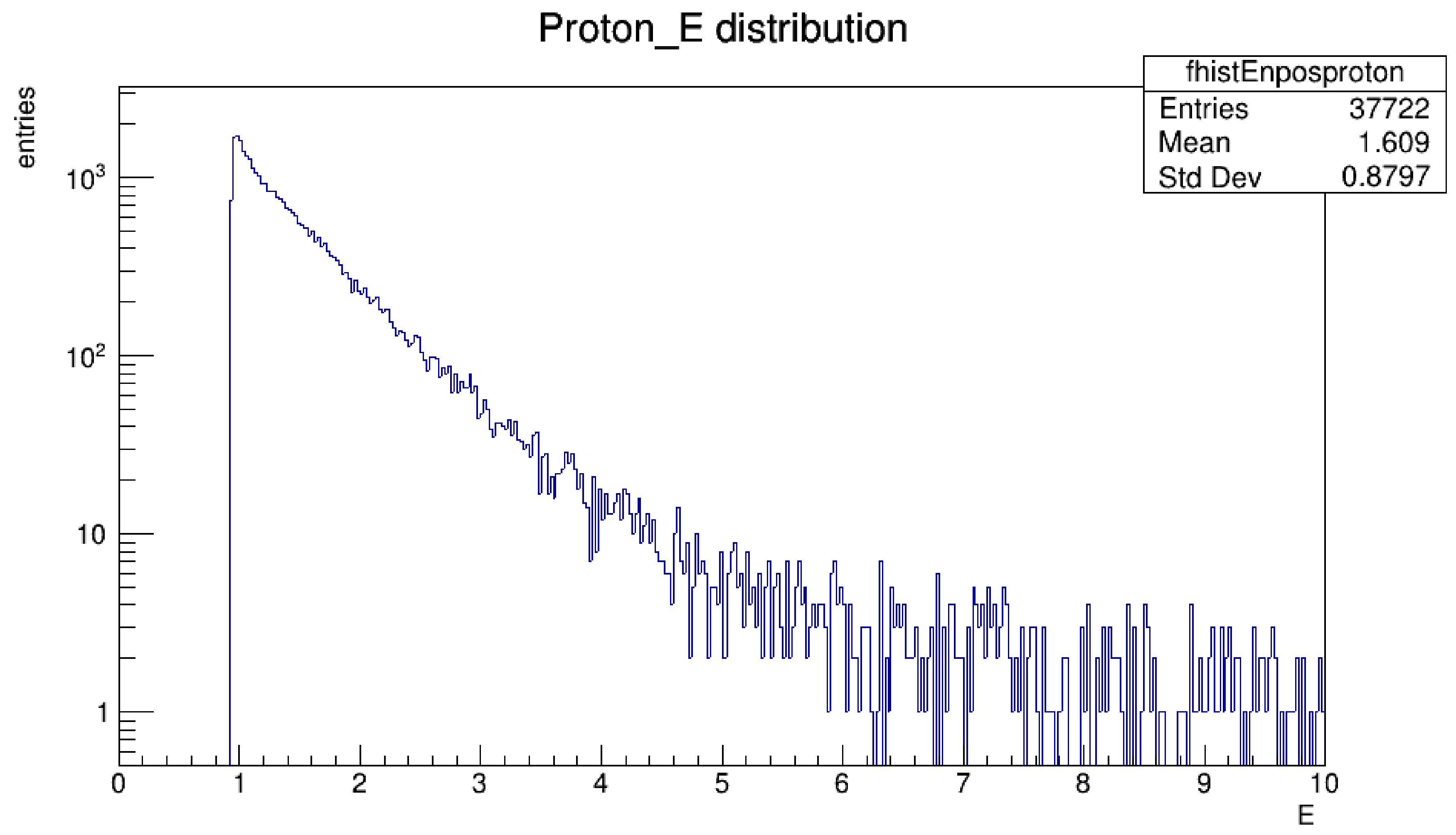


Pion Energy Distribución

Generated tracks (MC)



Reconstructed tracks (global tracks)



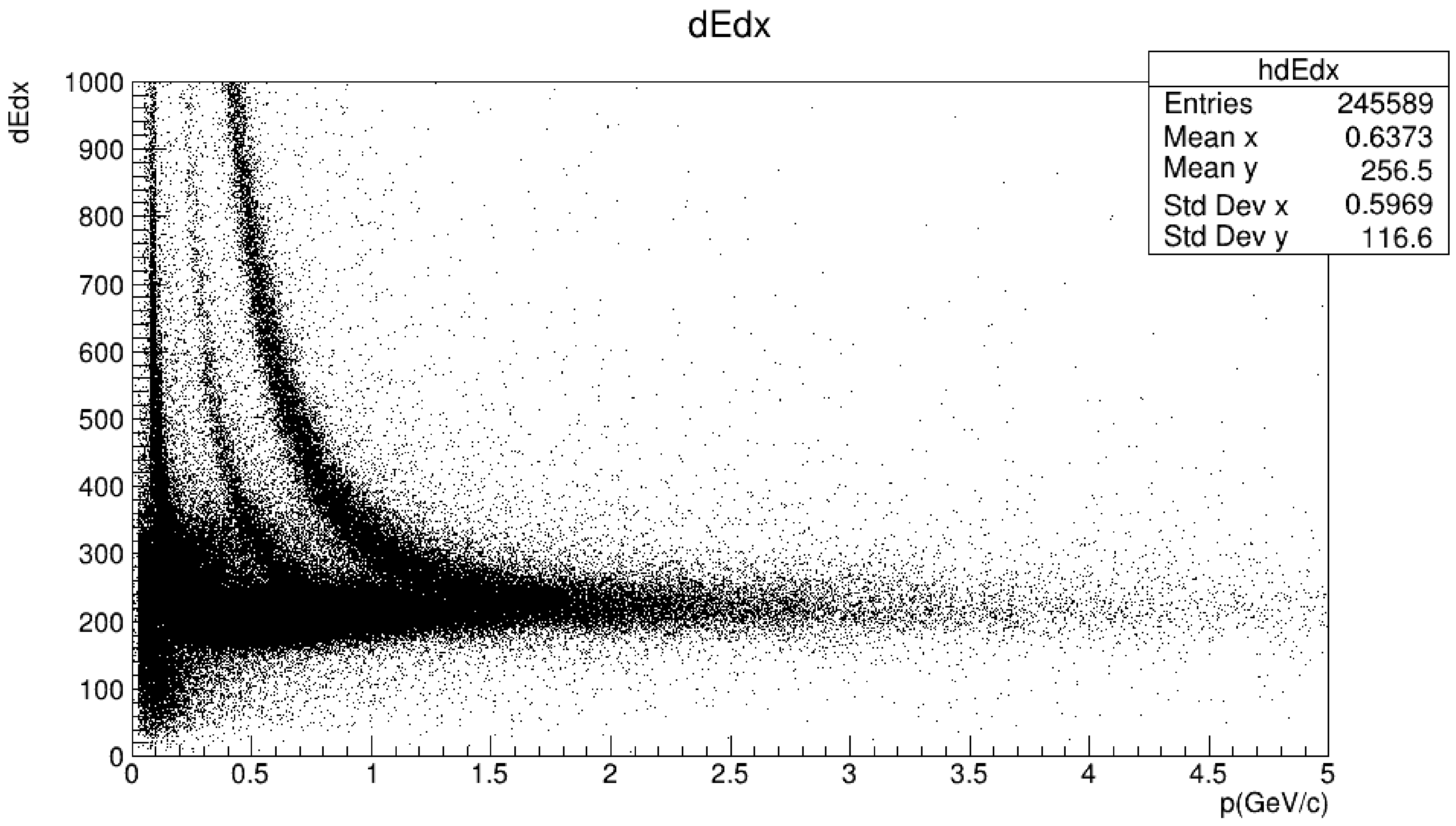
Diferents betewen loss energy MC and loss energy reconstrution

```
fTMCTracks = event.fMCTrack; //branches name defined in MpdAnalysisManager


cout << "N of MC tracks = " << nmctracks << endl;

for (Int_t i = 0; i < nmctracks; i++)
 {
   MpdMCTrack *MCtrack = (MpdMCTrack*) fTMCTracks->UncheckedAt(i);
   Int_t pdg = TMath::Abs(MCtrack->GetPdgCode());
   if (pdg == 211 || pdg == 321 || pdg == 2212 || pdg == 11 || pdg == 13 )
   {
           if(pdg == 321)
           {
                   Double_t Enposk=MCtrack->GetEnergy();
                   fhistEnposkMC->Fill(Enposk);
           }else if(pdg == 211)
           {
                   Double_t Enpospi=MCtrack->GetEnergy();
                   fhistEnpospiMC->Fill(Enpospi);
           }else if(pdg == 2212)
           {
                   Double_t Enpospro=MCtrack->GetEnergy();
                   fhistEnposproMC->Fill(Enpospro);
           }


   Double_t ptmc=MCtrack->GetPt();
   fhistPtMC->Fill(ptmc);

   TVector3 P(MCtrack->GetPx(),MCtrack->GetPy(),MCtrack->GetPz());

   Double_t etamc=0.5*TMath::Log((P.Mag() + MCtrack->GetPz())/(P.Mag() - MCtrack->GetPz()+1.e-13));
   //if(etamc > 1.3) continue;
   fhistetaMC->Fill(etamc);

   Double_t Enpos=MCtrack->GetEnergy();
   fhistEnposMC->Fill(Enpos);
   }
 }//fin del loop de los tracks montecarlo
```

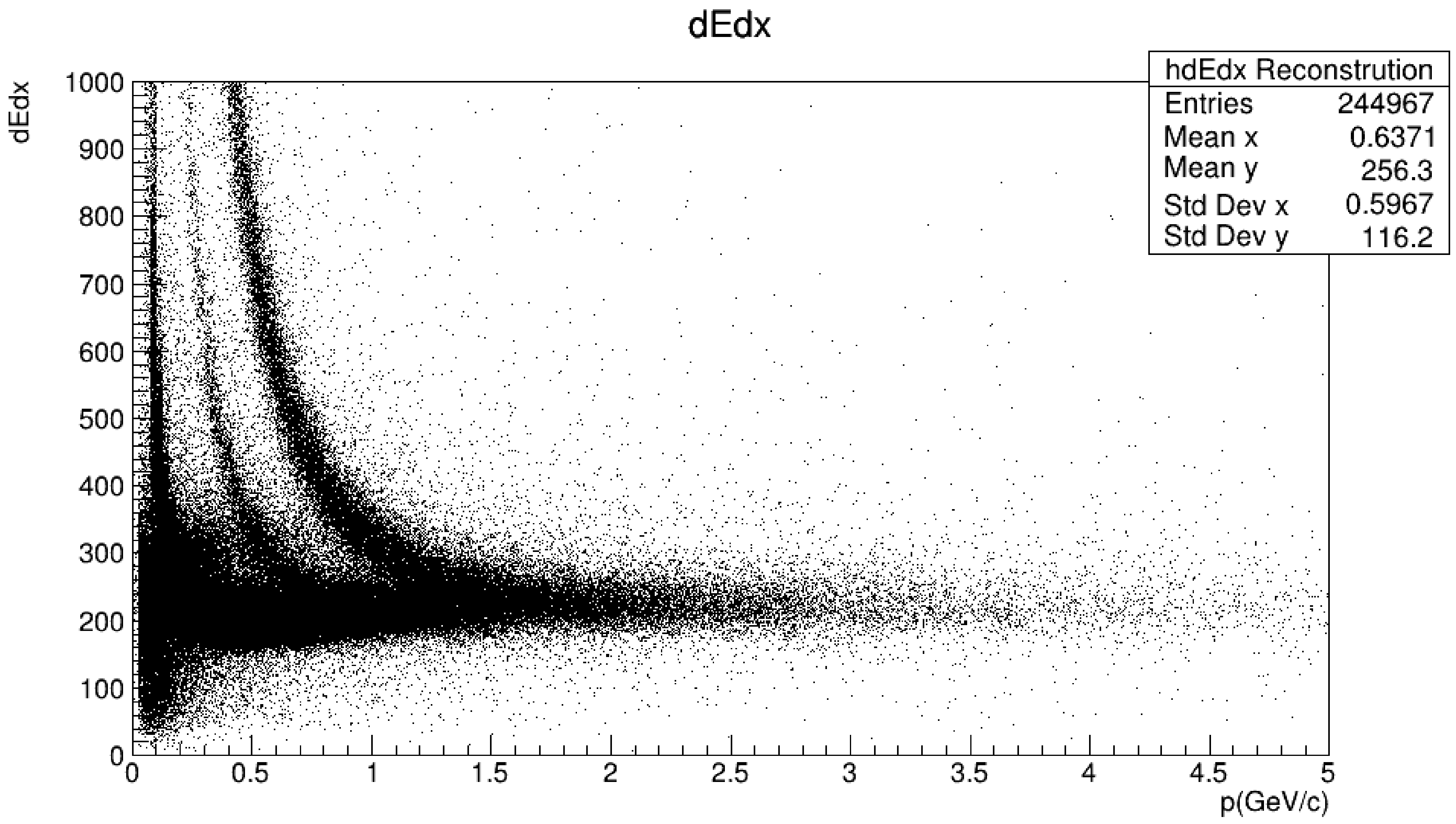
```
 }//fin del loop de los tracks montecarlo
//Reconstructed tracks
fTDstEvent = event.fMPDEvent;
fTMpdGlobalTracks = event.fMPDEvent->GetGlobalTracks();
Int_t ntracks=fTMpdGlobalTracks->GetEntriesFast();
for (Int_t i = 0; i < ntracks; i++)
{
    MpdTrack *track = (MpdTrack*) fTMpdGlobalTracks->UncheckedAt(i);
    Int_t idtrack = track->GetID();
    MpdMCTrack *mcTr = (MpdMCTrack*)fTMCTracks->UncheckedAt(idtrack);
    Int_t pdg = TMath::Abs(mcTr->GetPdgCode());
    TVector3 P(track->GetPx(),track->GetPy(),track->GetPz());


            if(pdg == 321)
            {
                    Double_t massK =0.493;
                    Double_t Enposk=TMath::Sqrt(massK*massK+P.Mag2());
                    fhistEnposk->Fill(Enposk);
            }else if(pdg == 211)
            {
                    Double_t massPi = 0.139;
                    Double_t Enpospi=TMath::Sqrt(massPi*massPi+P.Mag2());
                    fhistEnpospi->Fill(Enpospi);
            }else if(pdg == 2212)
            {
                    Double_t massPro = 0.938;
                    Double_t Enpospro=TMath::Sqrt(massPro*massPro+P.Mag2());
                    fhistEnpospro->Fill(Enpospro);
            }


    Double_t ptmc=track->GetPt();
    fhistPt->Fill(ptmc);


    Double_t etamc=track->GetEta();
    fhisteta->Fill(etamc);

    Double_t Enpos=mcTr->GetEnergy();
    fhistEnpos->Fill(Enpos);

    float dEdx = track->GetdEdXTPC();
    mhdEdx->Fill(TMath::Abs(ptmc)*TMath::CosH(etamc), dEdx);

    if (pdg == 211 || pdg == 321 || pdg == 2212 || pdg == 11 || pdg == 13 )
    {

    mhdEdxa->Fill(TMath::Abs(ptmc)*TMath::CosH(etamc), dEdx);

    }
}//Aquí se cierra el loop de las trazas reconstuidas
}
//___________________________________________________________________________
```

Loss Energy

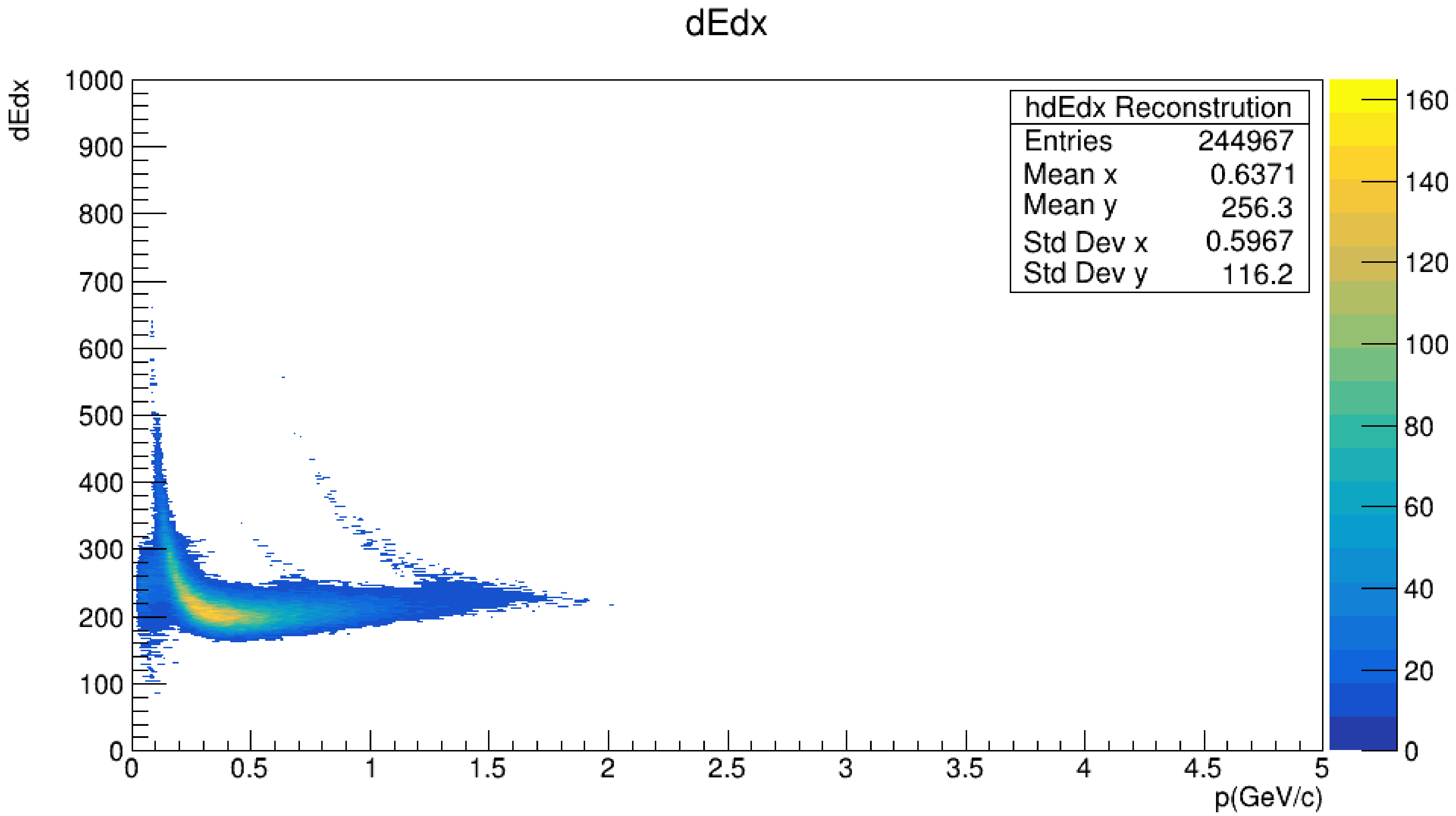
Generated tracks (MC)



Reconstructed tracks (global tracks)



Particles distribution on Loss Energy



Conclusions

For now we can use the Monte Carlo wagon and get several variables. On the other hand, we can also carry out reconstructions of the processes of our interest and compare

Referents

- https://git.jinr.ru/nica/mpdroot/-/tree/dev?ref_type=heads
- https://github.com/iamaldonado/CoreCoronaTask/tree/main
- https://github.com/iamaldonado/Macros_ANA/blob/main/README.md
- https://github.com/iamaldonado/Macros_ANA/tree/main/simpleRead/mpddst