Update of estimation of Core-Corona contribution to Global Polarization within mpdroot framework

Use of Centrality and evPlane wagon

I.A. Maldonado C.
maldonado@jinr.ru

April 11, 2023

Content

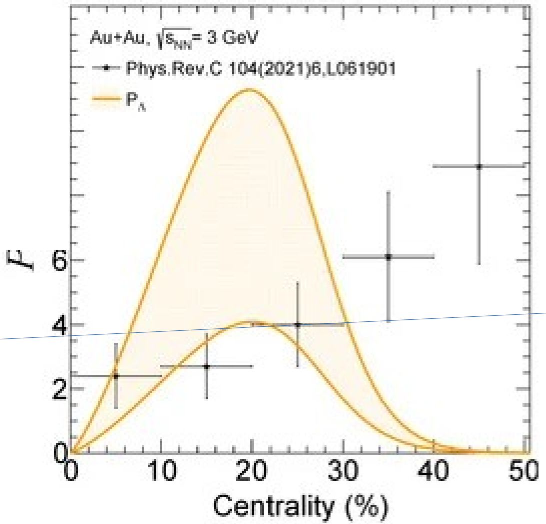
- Motivation
- Implementation
- Results
- Summary

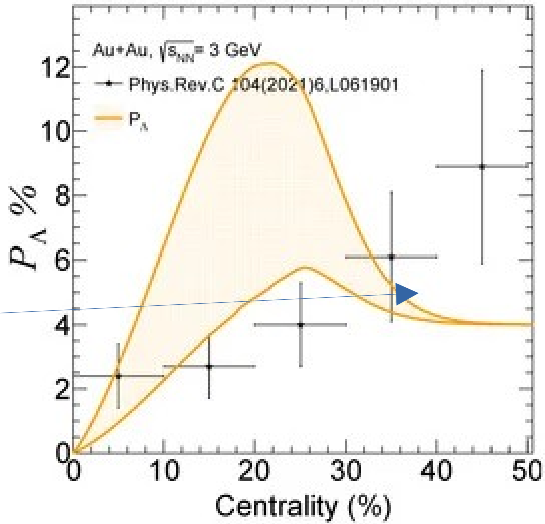
Motivation

- Core-Corona approach to describe main features of Hyperon global polarization but as a function of centrality fails, unless we consider polarization of particles in the corona, where the model suggest that QGP is not formed

$$\mathscr{P}^{\Lambda} = \frac{\mathscr{P}^{\Lambda}_{REC} + z\frac{N_{\Lambda\ QGP}}{N_{\Lambda\ REC}}}{1+\frac{N_{\Lambda\ QGP}}{N_{\Lambda\ REC}}},$$

A constant value increases the global polarization for 40-50% centrality bin





We wonder if polarization in corona produced by another mechanism as in pp collisions can contribute to global polarization.

Model Polarization in the corona

- With UrQMD we generate a small sample with core separation and assign an arbitrary polarization to the particles outside the core ~ 40% along the direction of production plane which corresponds to transverse polarization in pp collisions, and we use polarization transfer method implemented for PHSD data
- We extract transverse polarization with MC data and for comparison we repeat the exercise with request 25 and 30

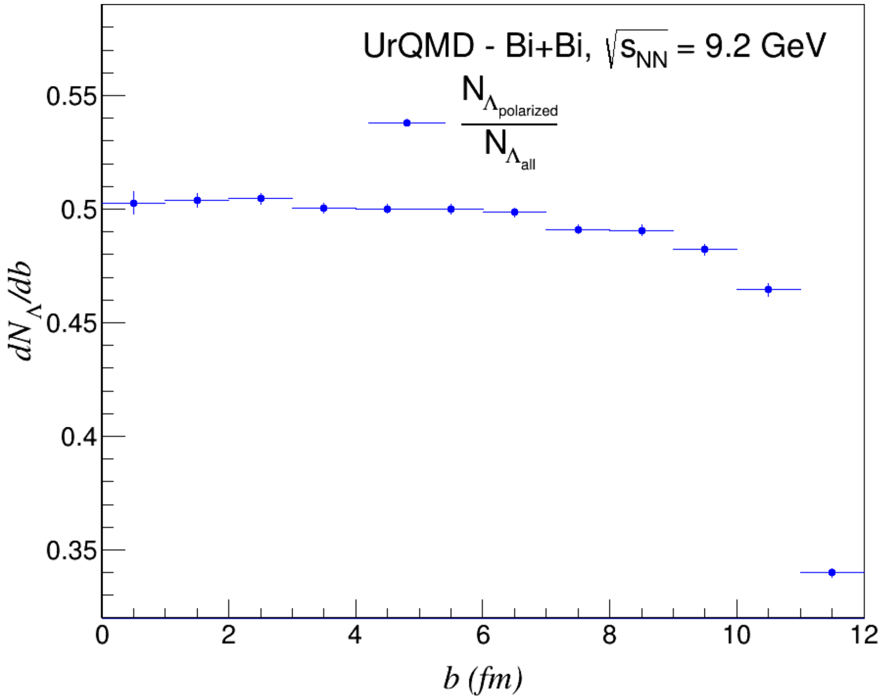
Samples - Used

Bi + Bi collisions at $\sqrt{s_{NN}}$ =9.2 GeV

- Request 25 ~ 2.1M events
- Request 30 ~ 2.8M events
- Core-Corona - ~150000 events with UrQMD and Core-Corona separation.

Corona + Secondary

- Ratio between polarized Λs produced in the corona with respect all Λs, including those secondary produced by decays



Task follows examples of pairKK and photon wagon

- MpdV0AnalysisTask - preliminary analysis done with MCTracks, because the low statistics in core-corona sample
  - Implements Event Plane angle to get polarization with flow method
  - Implements Centrality to get Polarization as a function of centrality

RunAnalises.C macro

- To know time and memory consumption you can add
  - At the begging of macro
    - TStopwatch timer;
    - timer.Start();
    - ProcInfo_t proc;
    - MemInfo_t memory;
  - At the end of macro
    - timer.Stop();
    - Double_t rtime = timer.RealTime(), ctime = timer.CpuTime();
    - printf("RealTime=%f seconds, CpuTime=%f seconds\n", rtime, ctime);
    - cout << "Macro finished successfully." << endl;

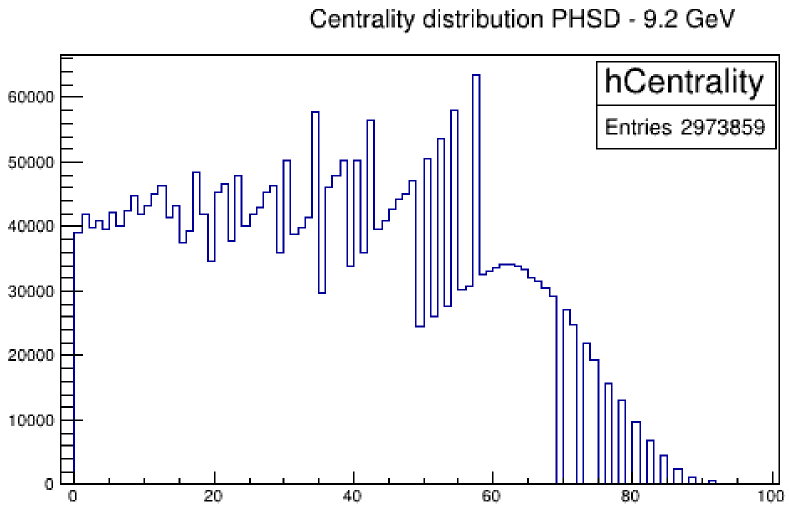
Tasks required for this analysis

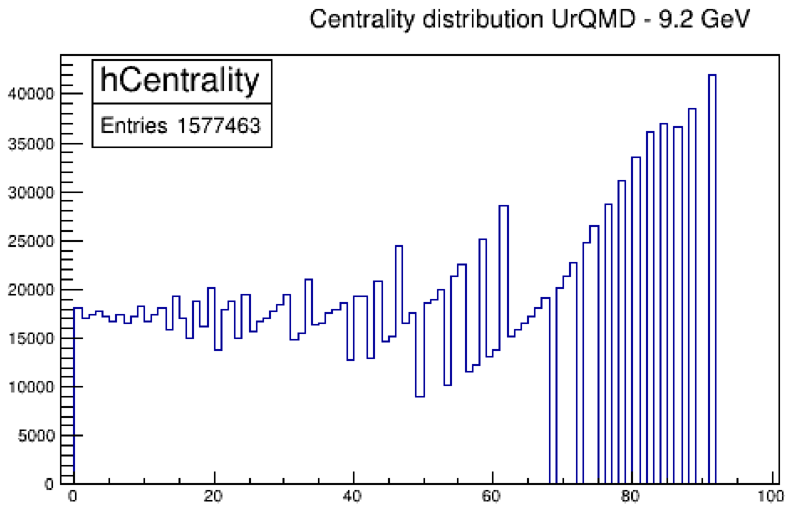
- To get Centrality
  - MpdCentralityAll pCentr("pCentr","pCentr") ;
  - man.AddTask(&pCentr) ;
- To get Event Plane
  - MpdEventPlaneAll pEP("pEP","pEP") ;
  - man.AddTask(&pEP) ;
- To implement the task
  - MpdV0AnalysisTask MpdV0AnalysisTask("V0","V0") ;
  - man.AddTask(&MpdV0AnalysisTask) ;

Lego Train helps to parallelize analysis

- Analysis split in 56 nodes (PHSD) and 38 nodes (UrQMD)
- For each 50000 PHSD events
  - RealTime=16731.950305 seconds, CpuTime=15284.050000 seconds
- For each 54900 UrQMD events
  - RealTime=7696.981067 seconds, CpuTime=7006.290000 seconds

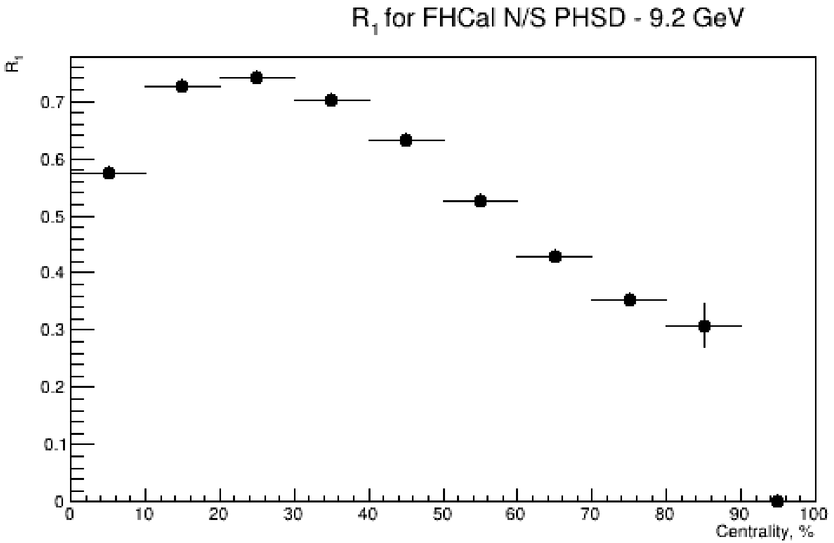
Centrality Distribution obtained

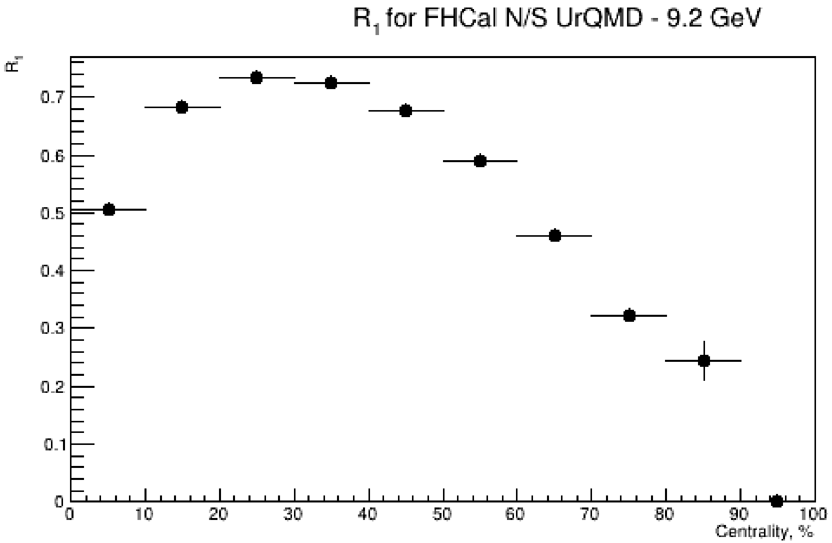




- Histograms from Centrality task - file pCentr.root for PHSD(left) and UrQMD(right) samples

Event Plane Resolution obtained





- Histograms from Event Plane task - file pEP.root for PHSD(left) and UrQMD(right) samples obtained with macro getResolution.C

Call Centrality - as in pairKK

- In the header file MpdV0Analysistask.h
  - float cen;
  - int   anaBin;
- In the implementation file MpdV0Analysistask.cxx
  - In the function selectEvent, get the centrality and select bin with:

```
cen = event.getCentrTPC();
AnaBin = -1;
If (cen >= 0 && cen < 10) anaBin = 0;
if (cen >= 10 && cen < 20) anaBin = 1;
if (cen >= 20 && cen < 30) anaBin = 2;
if (cen >= 30 && cen < 40) anaBin = 3;
if (cen >= 40 && cen < 50) anaBin = 4;
if (cen >= 50 && cen < 60) anaBin = 5;
if (cen >= 60 && cen < 70) anaBin = 6;
if (cen >= 70 && cen < 80) anaBin = 7;
if (cen >= 80 && cen < 90) anaBin = 8;
if (cen >= 90 && cen < 100) anaBin = 9;
```

- To Fill histograms for each centrality bin
  - In the header File

```
TH1F *h[10];
```

  - In the implementation file, describe the histogram in the UserInit function

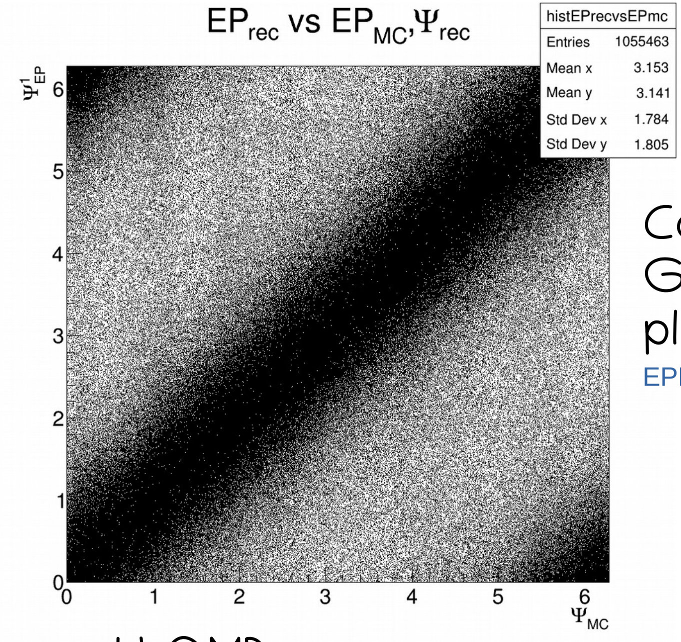
```
for(Int_t k=0;k<10;++k){
h[10]=new TH1F(Form("h_%d",k),Form("Cent. Bin %d",k),100,0,10);
}
```

  - In the ProcessEvent function in the corresponding loop of you analysis fill the histogram for each bin

```
h[anaBin]->Fill(my_variable);
```

Call Event Plane

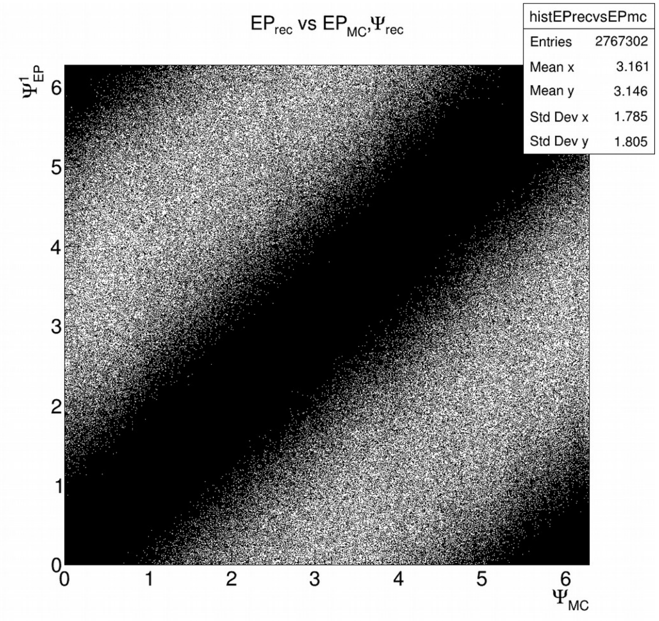
- In the function ProcessEvent, get the angle and write in the interval (0,2π) with:
  - Double_t EPtrain = event.fMpdEP.GetPhiEP_FHCal_F_all() + TMath::Pi();



UrQMD

Comparison with Generated event plane

EPMC = MCHeader->GetRotZ();



PHSD

To check the Task we calculate

- Transverse local polarization with the angular distribution

$$\frac{dN}{d\Omega}=\frac{N}{4\pi}(1+\alpha\cos\theta')$$

With the angle Θ' of decay proton measured with respect the normal to production plane given by

$$\hat{n}\equiv\frac{\vec{p}_{beam}\times\vec{p}_{\Lambda}}{|\vec{p}_{beam}\times\vec{p}_{\Lambda}|}$$

- Global Polarization with the same method as in local polarization but now the angle measured with respect the Total angular Momentum

$$\hat{L}=\hat{b}\times p_{\widehat{beam}}=(\sin\Psi_{EP},-\cos\Psi_{EP},0)$$

- Global Polarization with the Flow Method

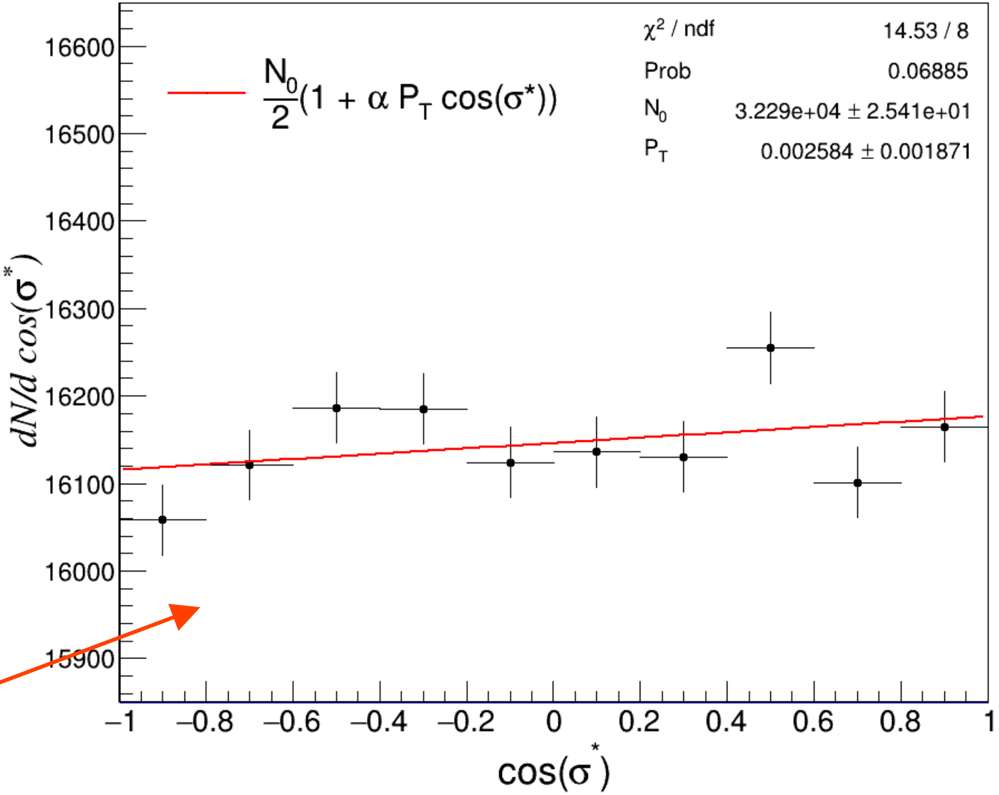
$$\mathcal{P}_{\Lambda}=-\frac{8\langle\sin(\phi_p-\Psi_{RP})\rangle}{\pi\alpha}$$

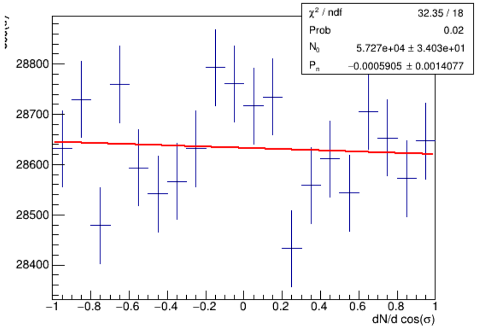
For each centrality bin

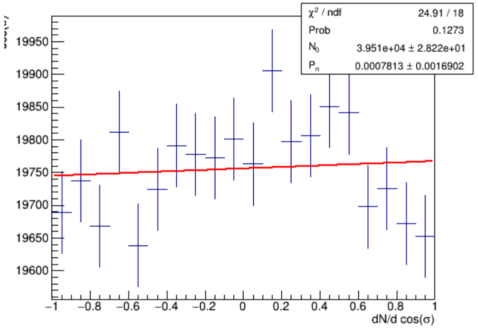
Results with MCTracks
Fits to histograms

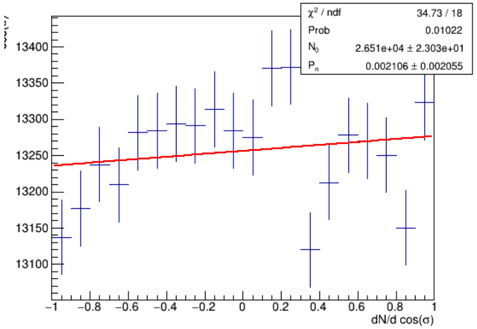
Fit to extract polarization

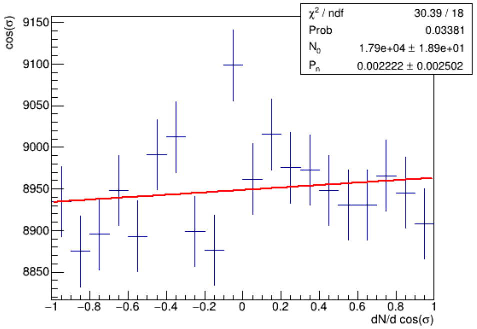
- Distribution of decay proton with respect to:
  - Normal to production plane
    - Transverse polarization
  - Angular momentum
    - Global Polarization
- On the plot w.r.t normal to production plane

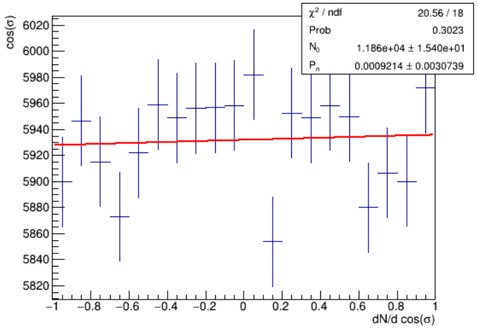


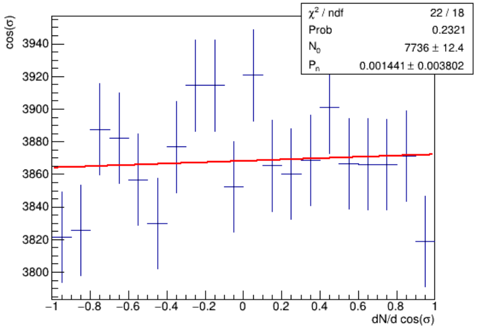


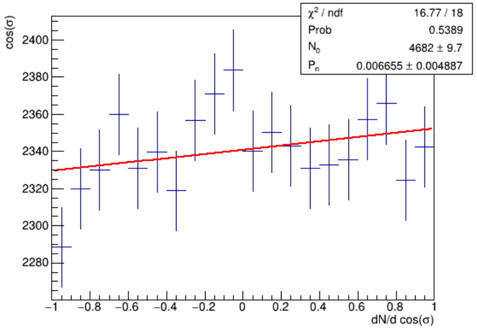


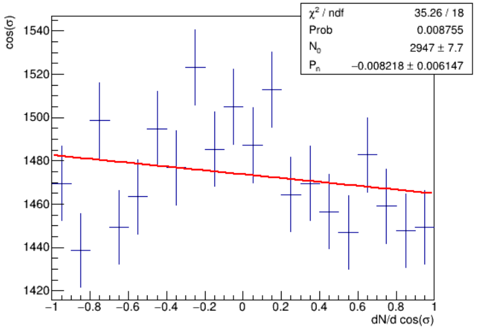


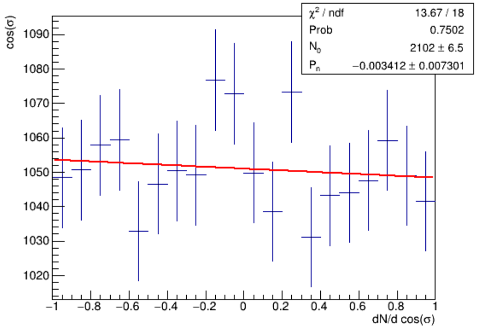


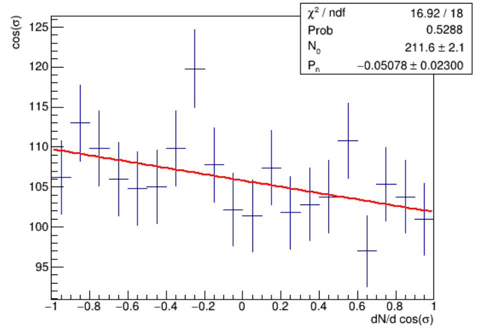




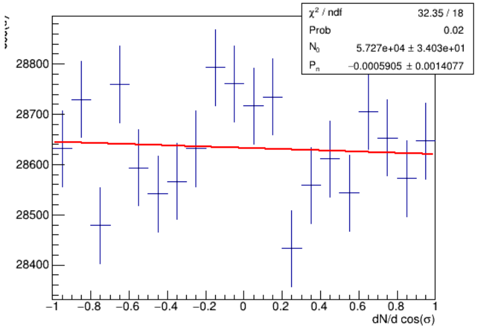


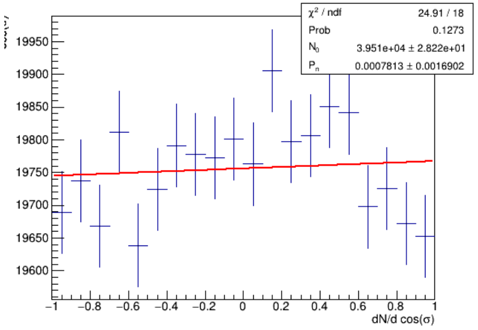


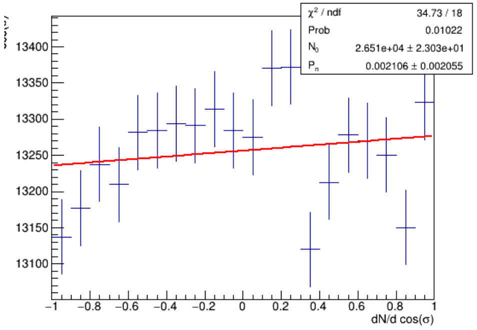


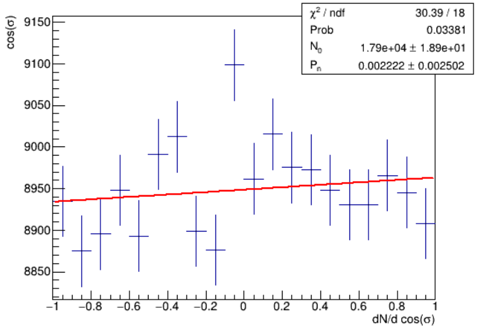


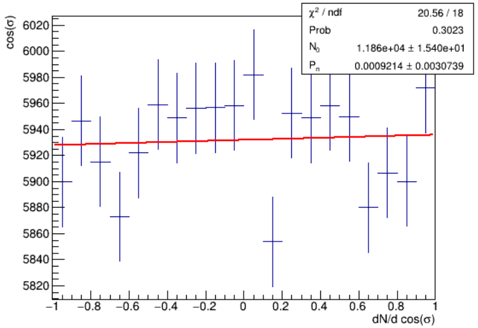
UrQMD Transverse Polarization - Corona Contribution

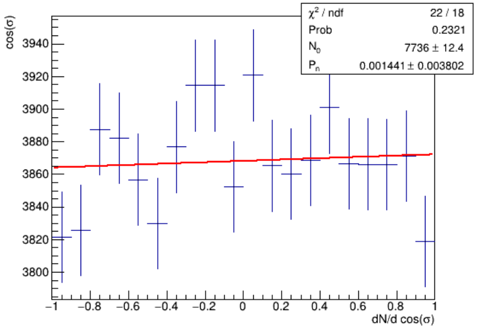


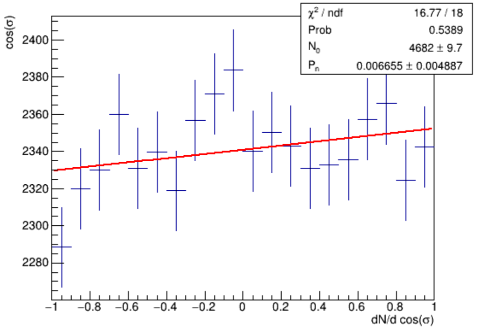


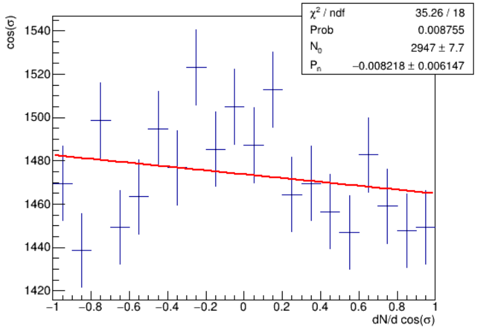


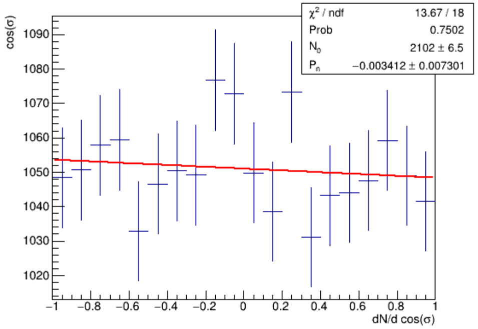


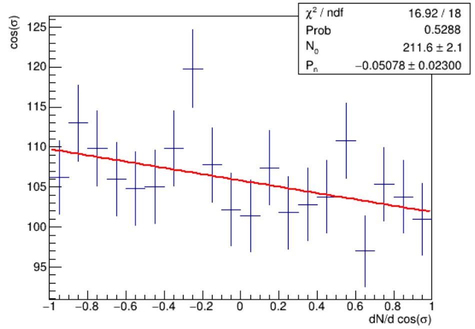












UrQMD Global Polarization - projection

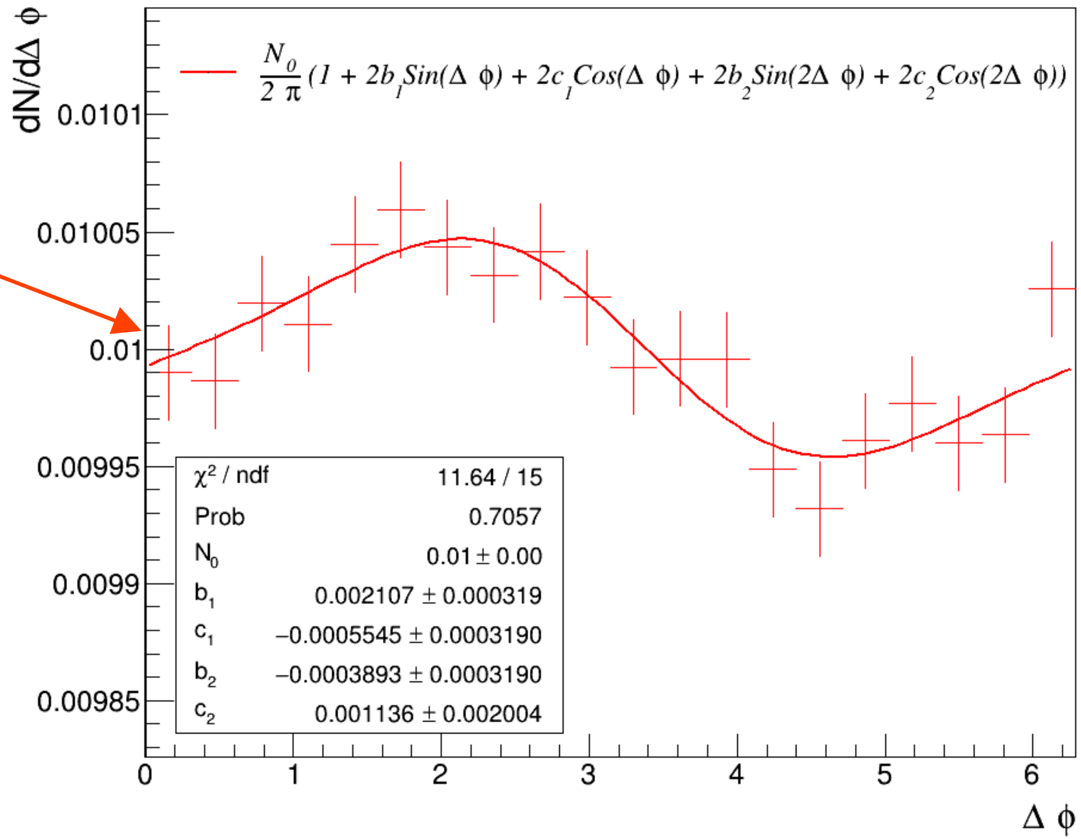
Fit to extract polarization

- Example of the Fit to the angular difference

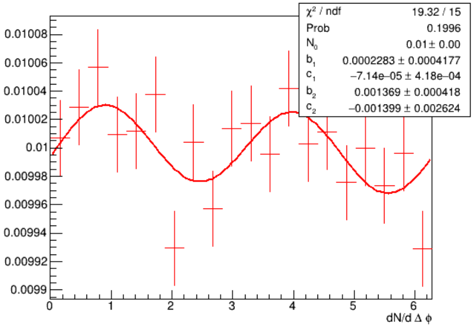
$$\Delta\phi = \Psi_{EP} - \phi$$

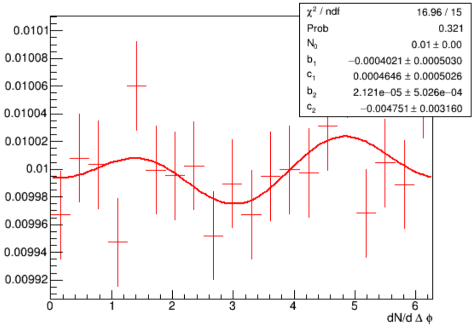
- Polarization

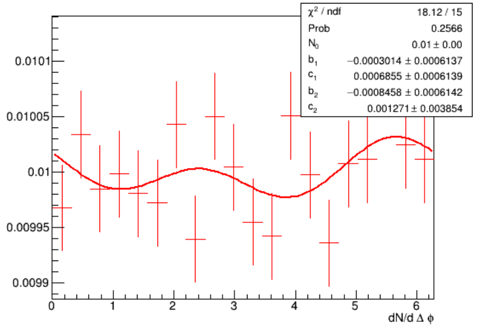
$$P = \frac{8 b_1}{\alpha \pi R^1_{EP}}$$

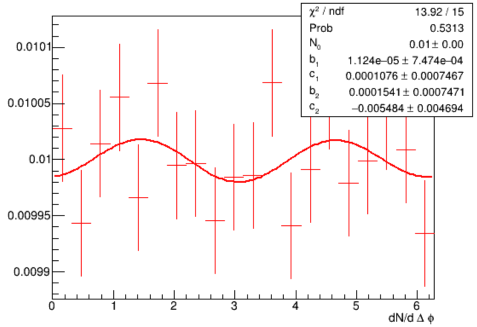


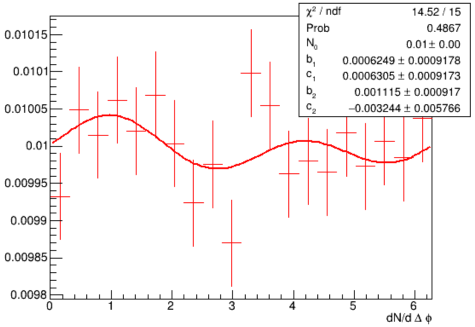
χ^2 / ndf	11.64 / 15
Prob	0.7057
N_0	0.01 ± 0.00
b_1	0.002107 ± 0.000319
c_1	−0.0005545 ± 0.0003190
b_2	−0.0003893 ± 0.0003190
c_2	0.001136 ± 0.002004

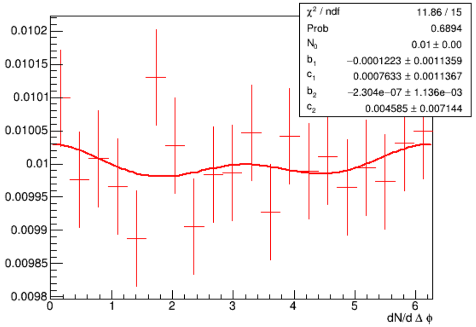


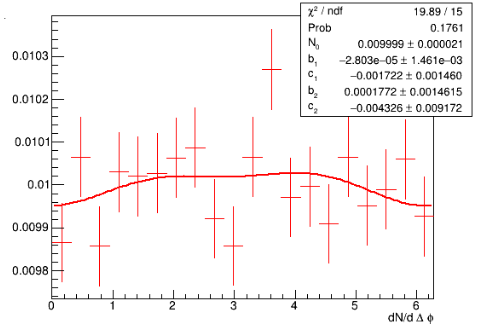


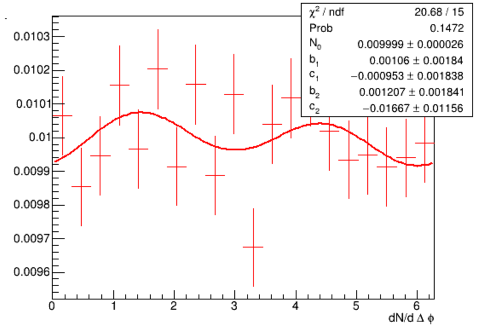


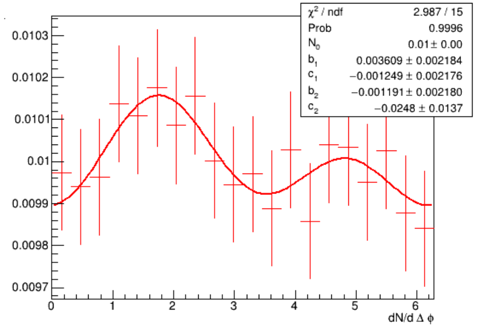


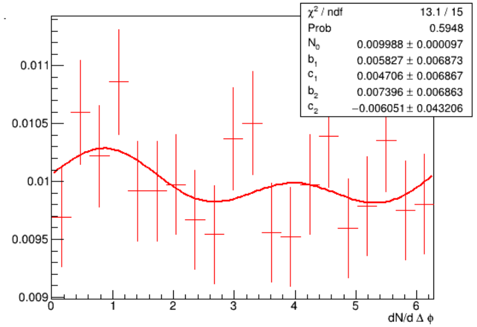






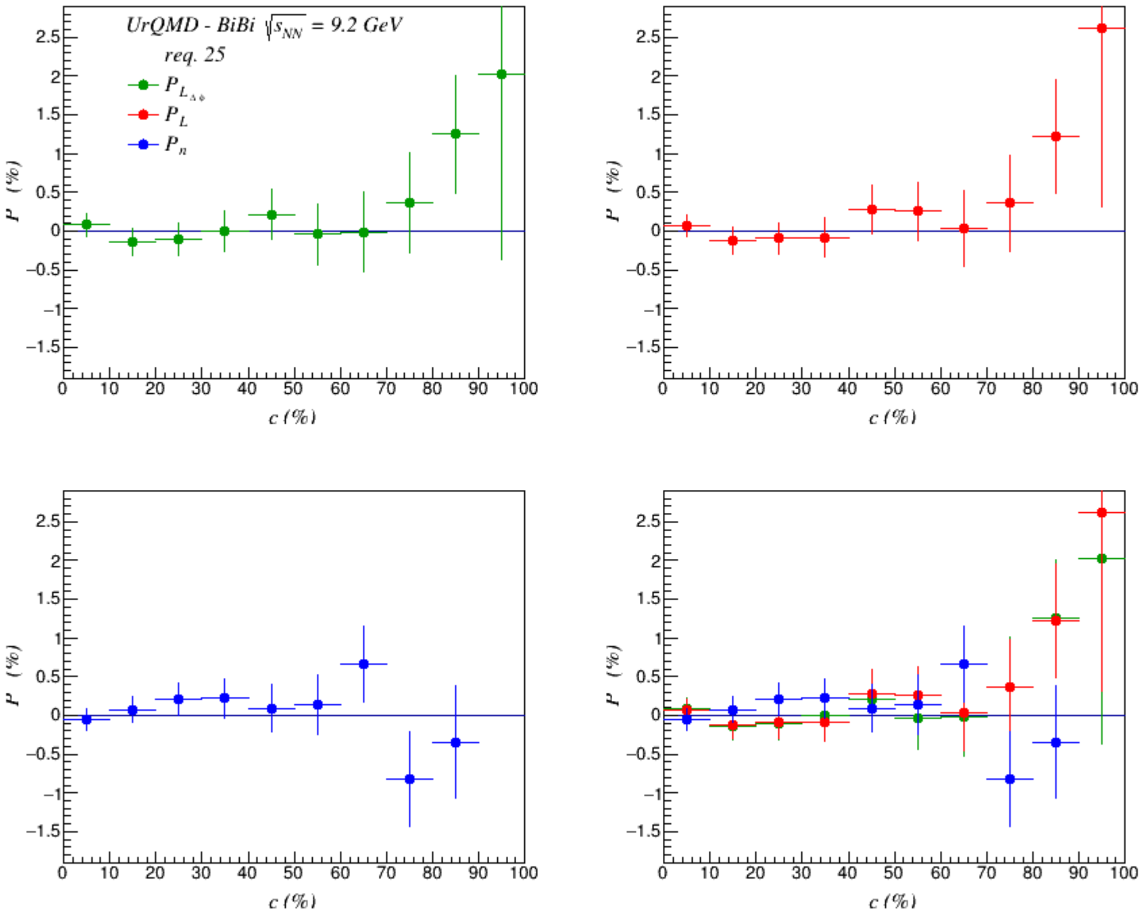






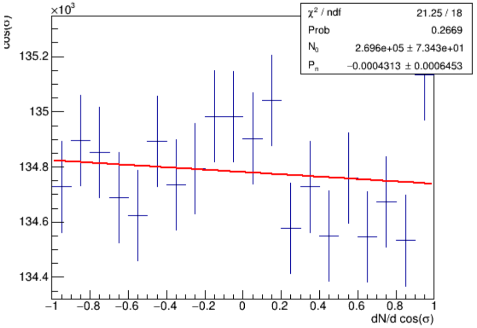
UrQMD Global Polarization - Flow method

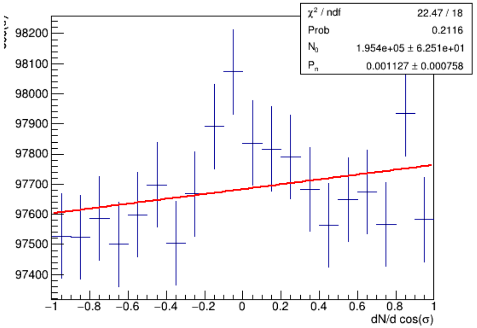
Comparison of Results

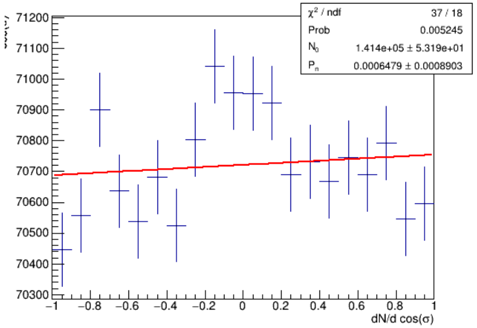


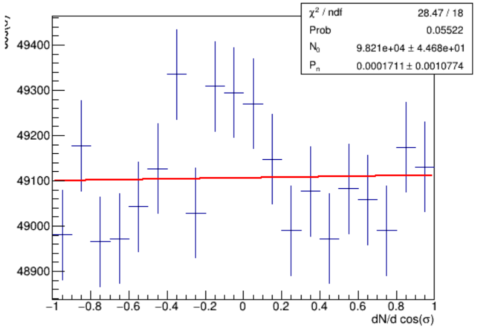
- Most peripheral bins shows a global polarization - even sample was not generated with polarization
- Projection method and Flow method shows similar results
- Transverse polarization is consistent with zero except the most peripheral bins from positive change to negative value 5% in c(90,100)

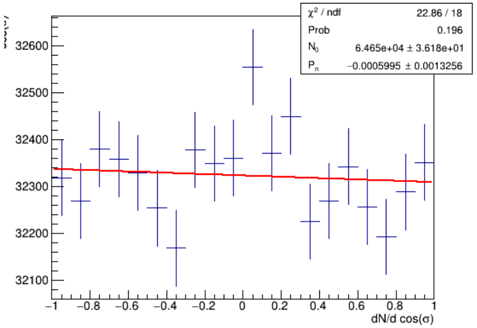
PHSD -Transverse Polarization - Corona Contribution

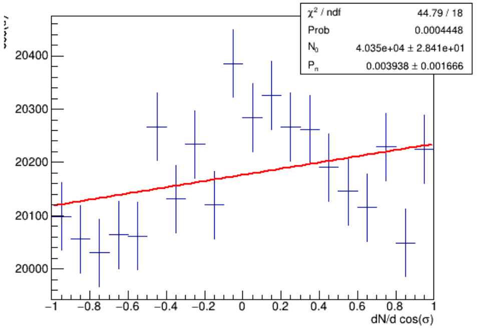


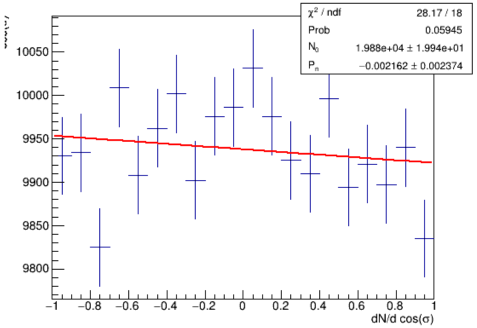


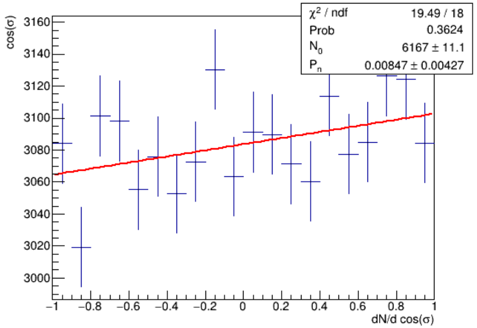


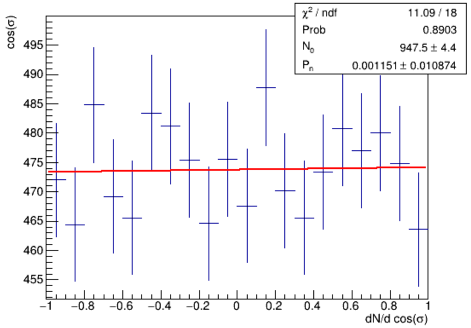


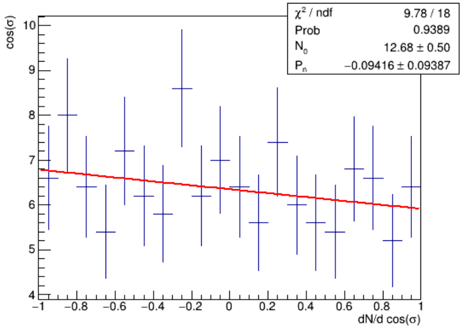


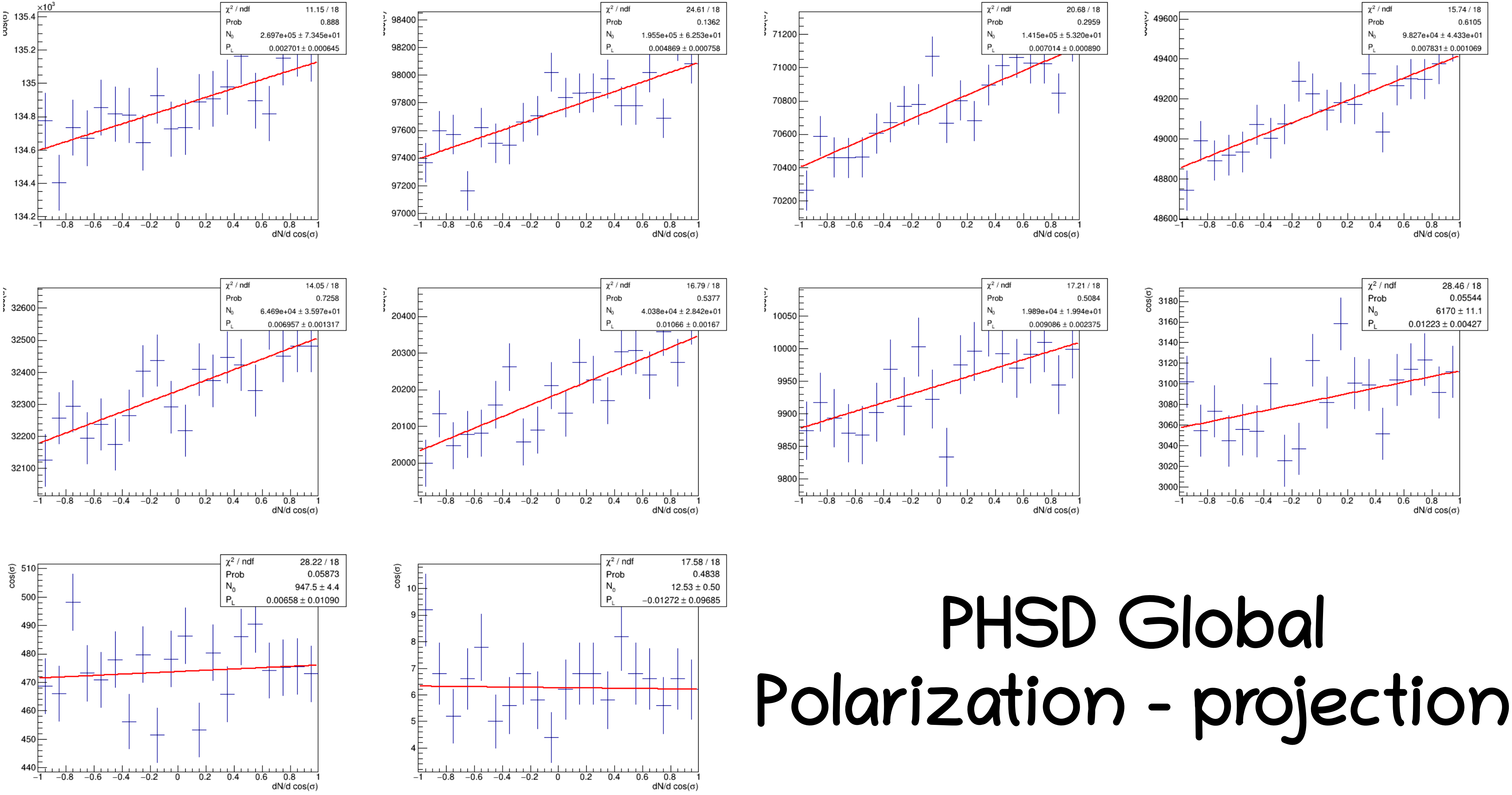




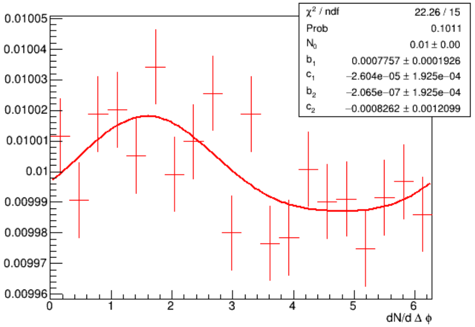


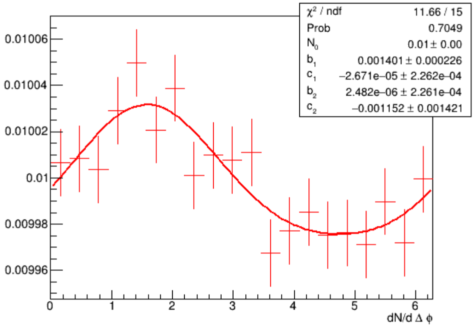


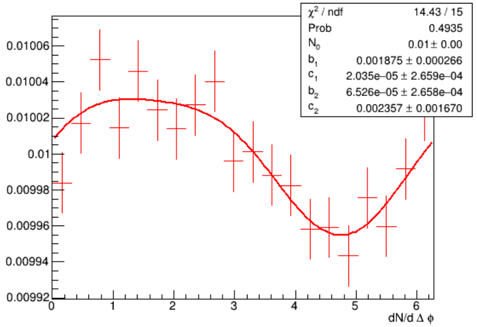


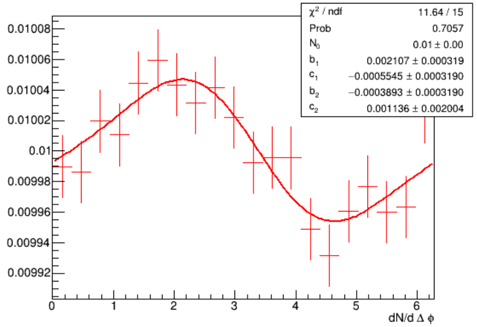


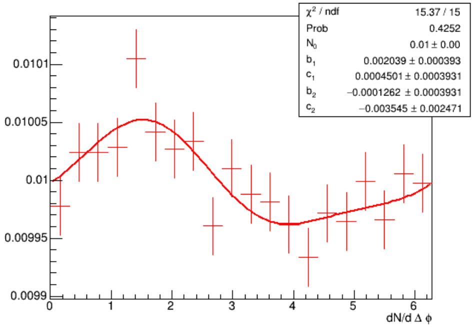
PHSD Global Polarization - projection

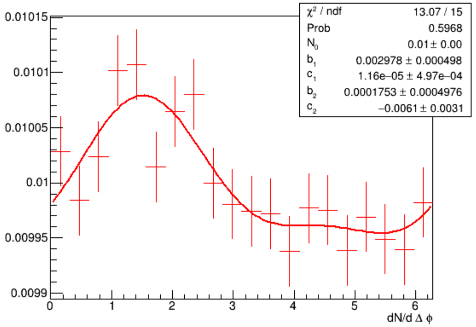


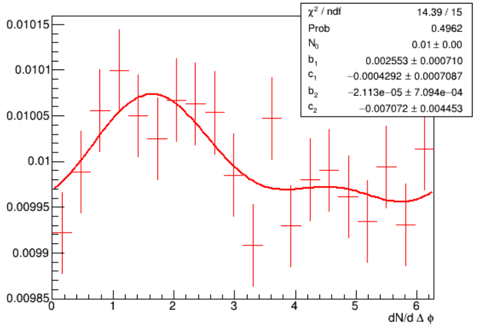


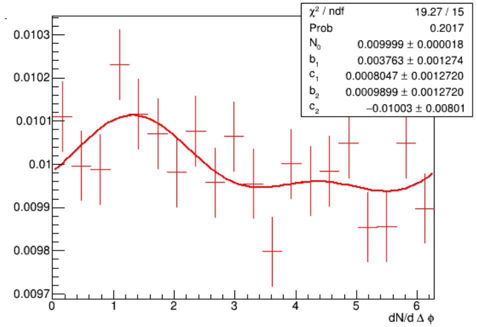


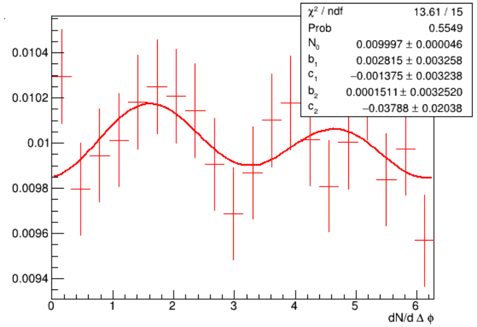


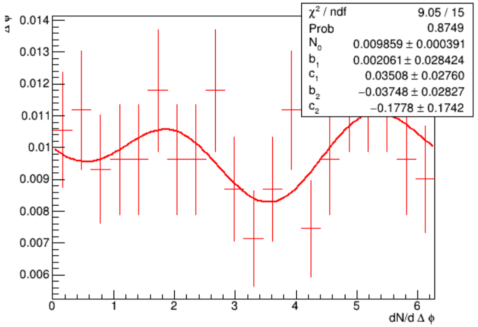






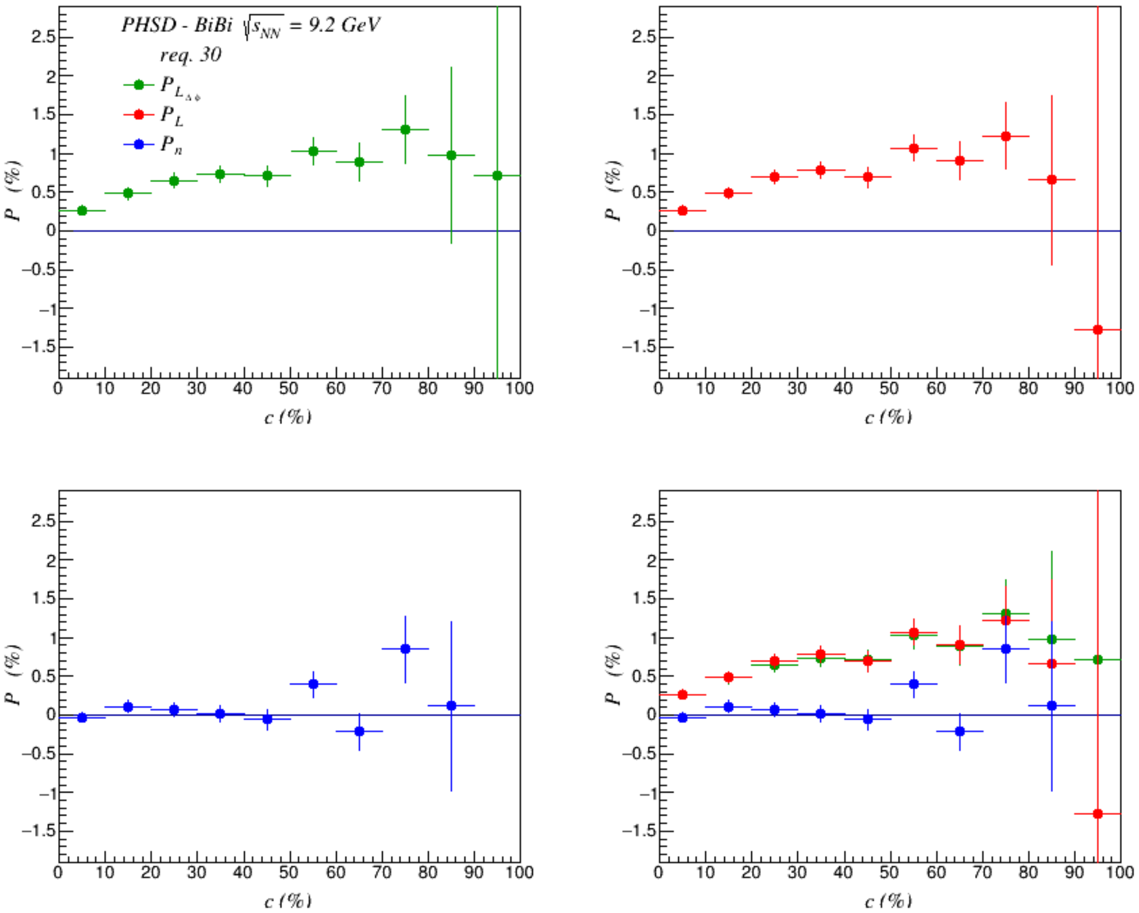






PHSD Global Polarization - Flow method

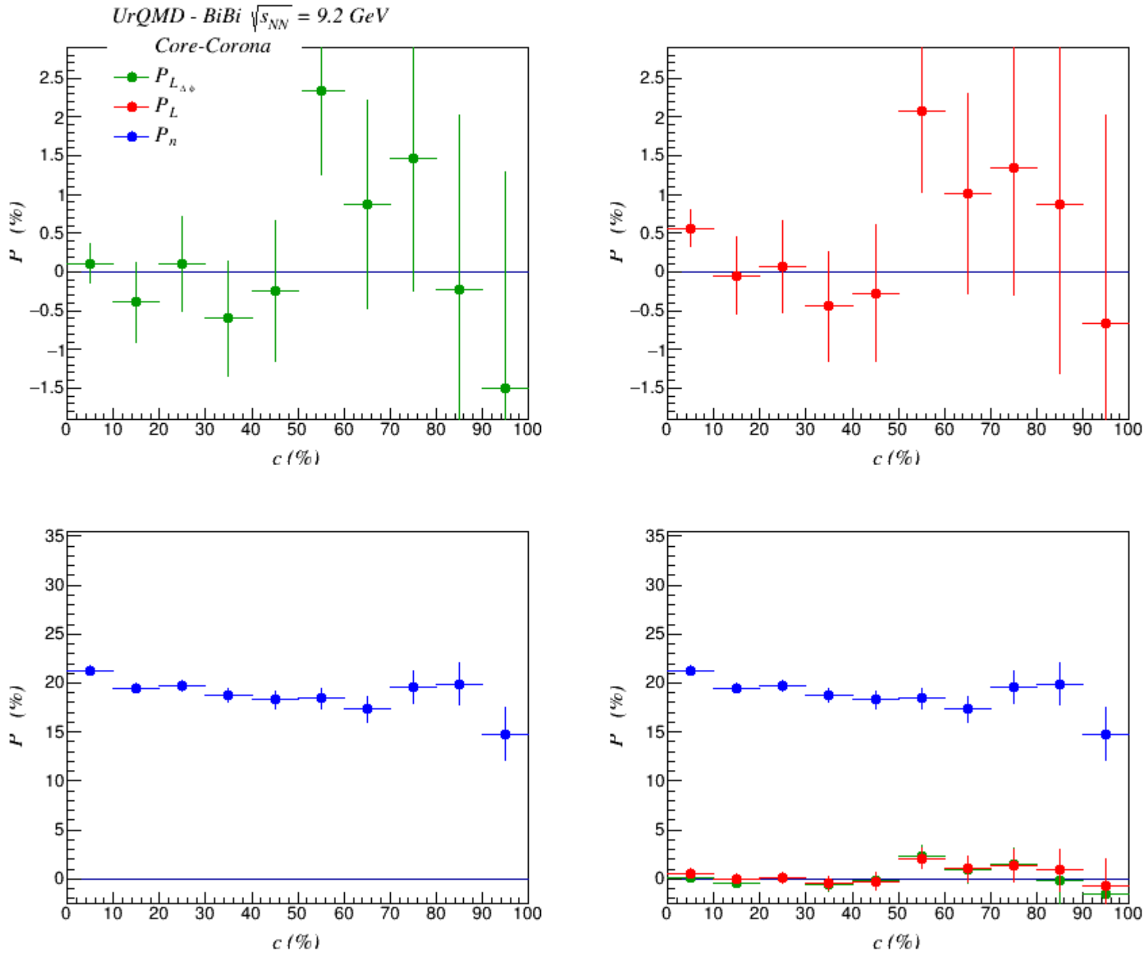
Comparison of Results



- Most peripheral bin require more statistics
- Projection method and Flow method shows similar results (green and red)
- Transverse polarization is consistent with zero except the most peripheral bins from positive change to negative value 9% in c(90,100) higher wrt UrQMD

Similar for Core - Corona

Comparison of Results

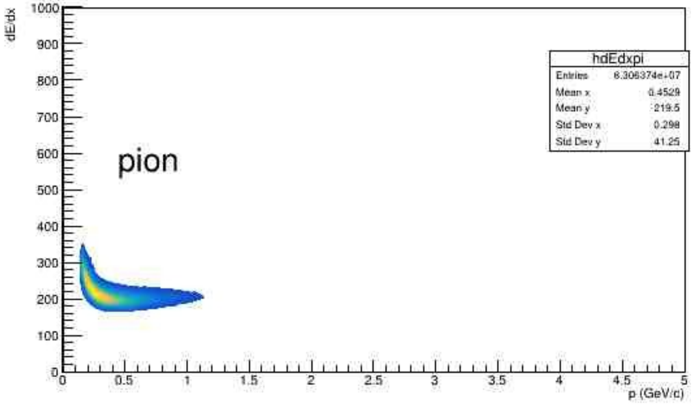


- Poor statistics for global polarization
- Projection method and Flow method shows similar results (green and red) as in previous
- Transverse polarization measured is 20% assuming only ~50% of particles are polarized dilutes original polarization by half

Results with MpdGlobalTracks Requires particle ID and V0 reconstruction

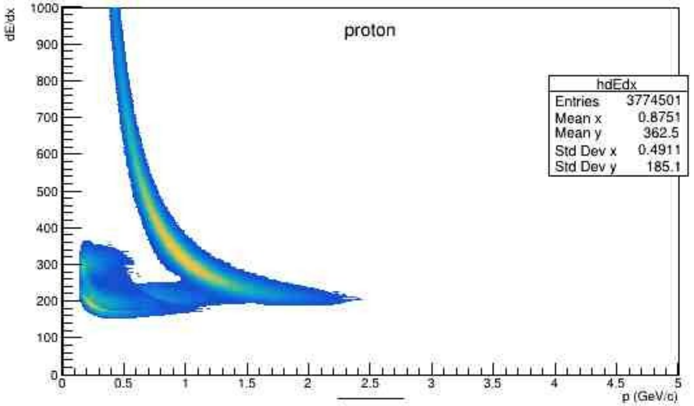
Select protons and pions

- For now only for Lambda
- Follows the function selectTrack as in pairKK and photons
- It select events that pass cuts on |&eta|, pT, nhits and dca and applies PID selection with MpdPID class in https://git.jinr.ru/nica/mpdroot/-/tree/dev/core/mpdPid



- Is implemented with the following lines

```
isGoodPID =
mPID→FillProbs(TMath::Abs(pt)*TMath::CosH(eta),track
→GetdEdXTPC()*6.036e-3,track→GetTofMass2(),-1);
if (isGoodPID && (mPID->GetProbPi() < 0.75)) {
  return false;
 }
```



Example of output histograms

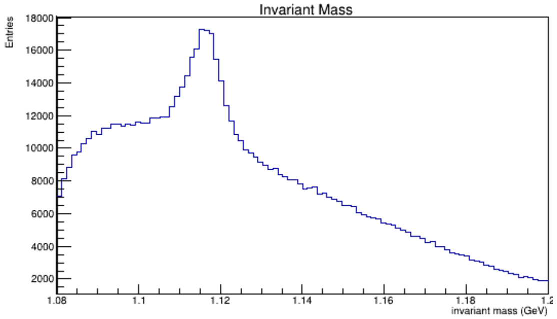
V0 reconstruction

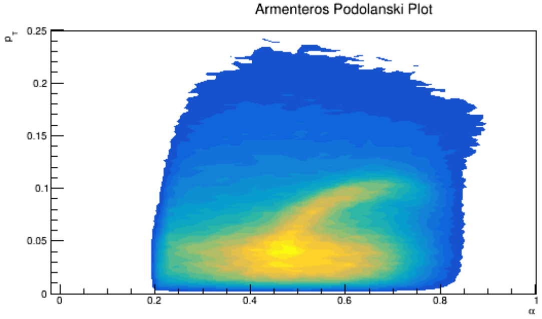
- Follows photons task to build Lambda with trp and trn MpdTpcKalmanTracks

```
MpdTpcKalmanTrack trCorK1(*trp);
MpdHelix helixp = MakeHelix(trCorK1);
MpdParticle      pr(trCorK1, 0);
if(chargep > 0)pr.SetPdg(2212);
if(chargep < 0)pr.SetPdg(-2212);
pr.SetMass(0.938272);

MpdTpcKalmanTrack trCorK2(*trn);
MpdHelix helixn = MakeHelix(trCorK2);
MpdParticle      pi(trCorK2, 0);
if(chargen > 0)pi.SetPdg(211);
if(chargep < 0)pi.SetPdg(-211);
pi.SetMass(0.139570);

mPartV0.clear();
mPartV0.emplace_back(&pr);
mPartV0.emplace_back(&pi);

MpdParticle V0La;
float      chi2La = Tmath::Abs(V0La.BuildMother(mPartV0));
float      ptLa   = V0La.Pt();
...
```

- Clean the signal requires improve the cuts on V0 reconstruction
- This is work in progress

Examples of output

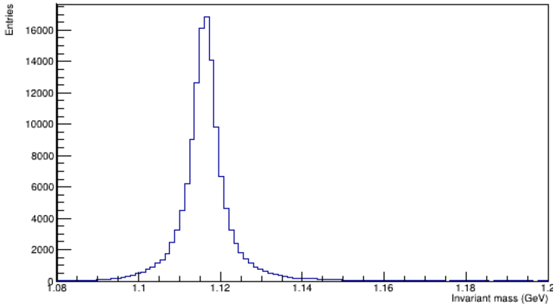


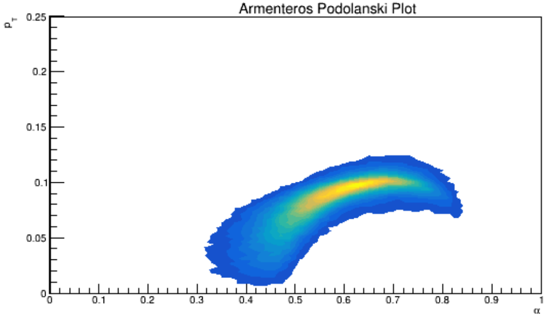


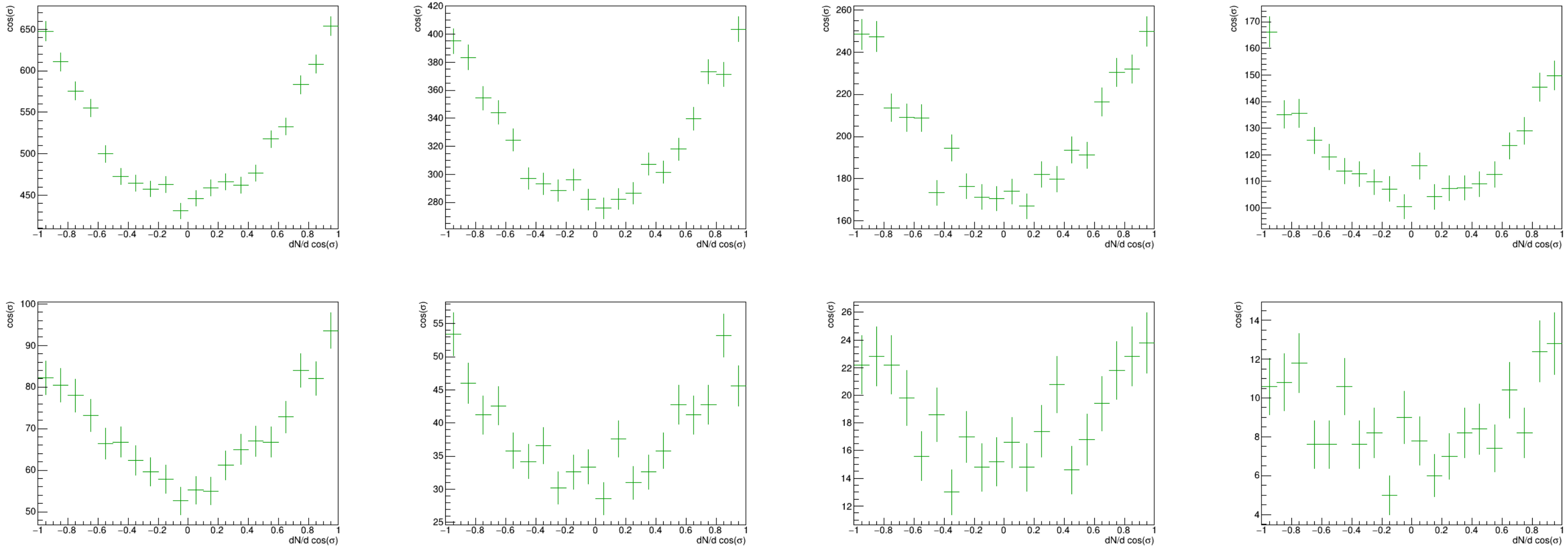
Starting with MC association

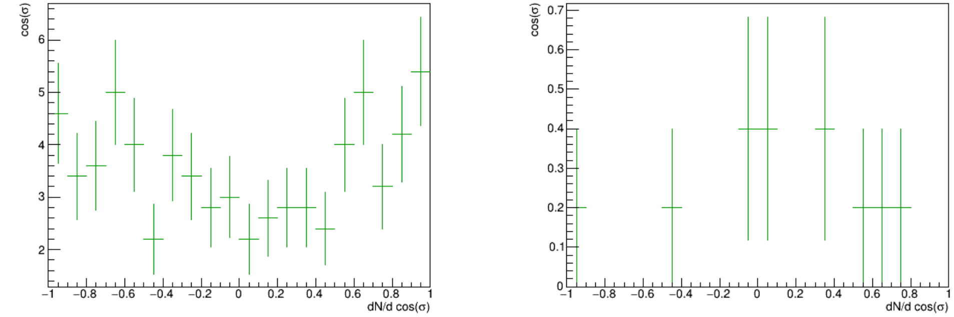
- To test the task, first attempt is with MC association
- Cleans the signal

Examples of output



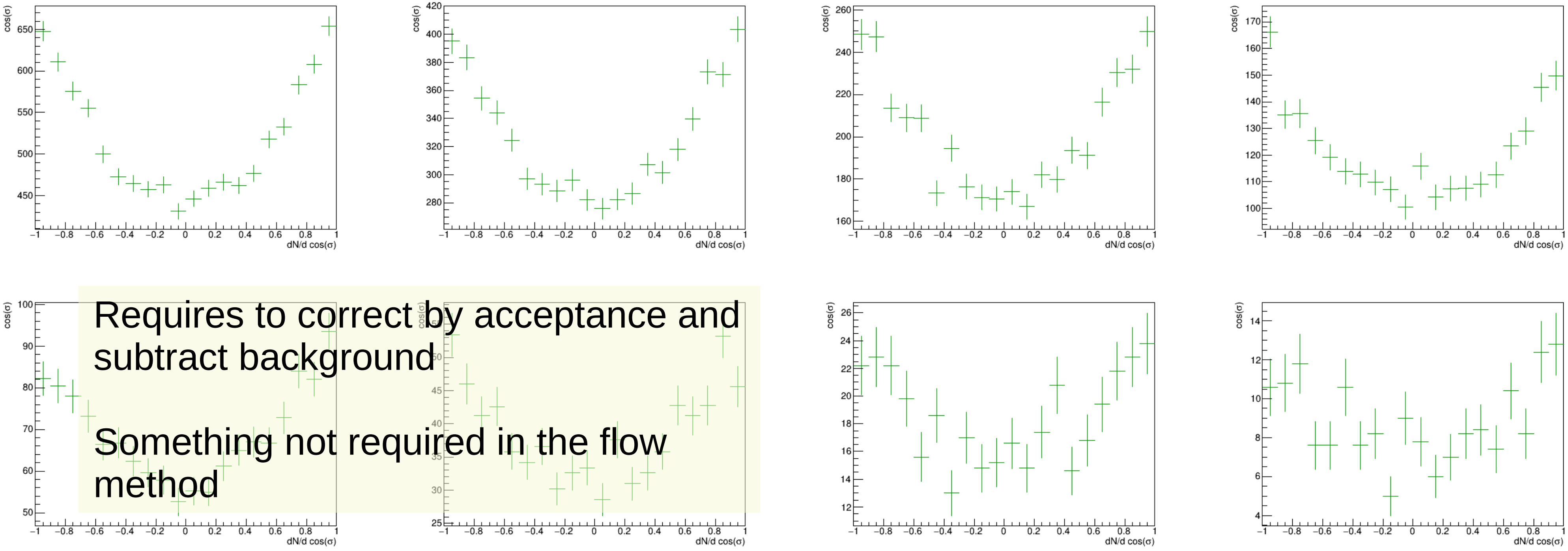


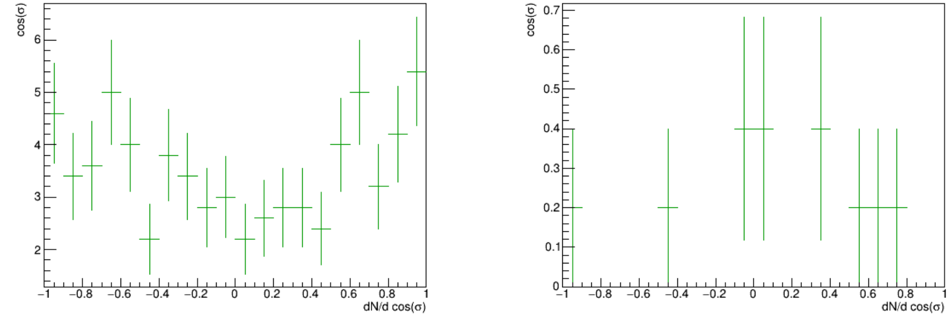




UrQMD Transverse Polarization - projection

UrQMD Transverse Polarization - projection

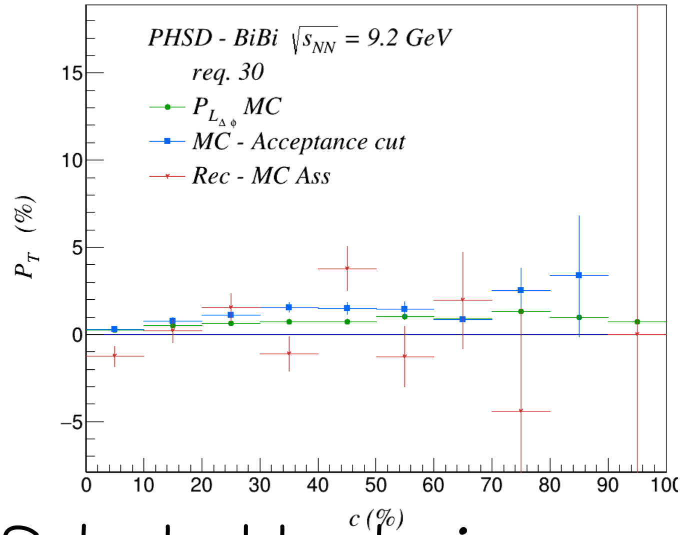


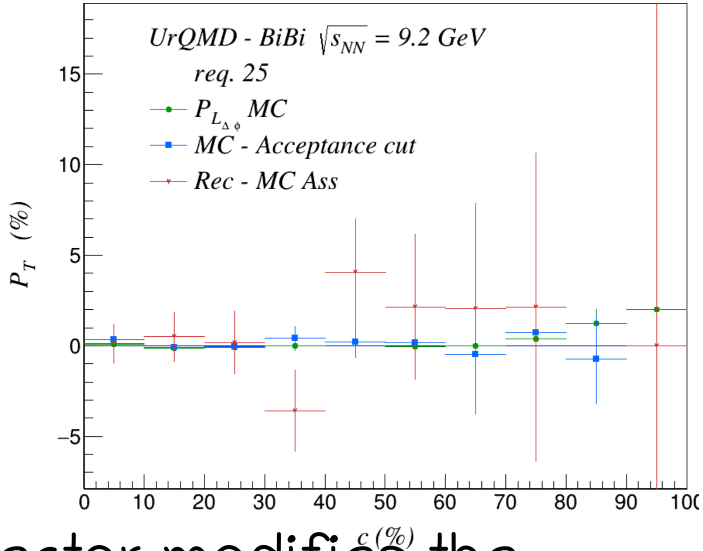


Requires to correct by acceptance and subtract background

Something not required in the flow method

Global Polarization with Flow Method





- Selected tracks in acceptance of detector modifies the distribution, it looks to increase a little bit polarization (blue line)
- Reconstruction with association requires improve the selection on task

Summary

- It has been shown the implementation of analysis train, the task call centrality and event plane determination
- Results of polarization with MC tracks are shown for PHSD, UrQMD and Core-Corona
- This is work in progress it need to be improved to measure polarization contribution from corona
- Preliminary version https://github.com/iamaldonado/CoreCoronaTask