



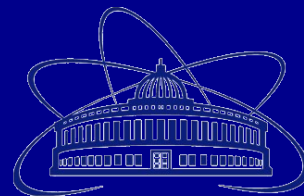
4th Computing/Analysis Workshop of the MexNICA Collaboration

Λ Production at NICA Energies

I. A. Maldonado C.

ivonne.alicia.maldonado@gmail.com

June 16th, 2022

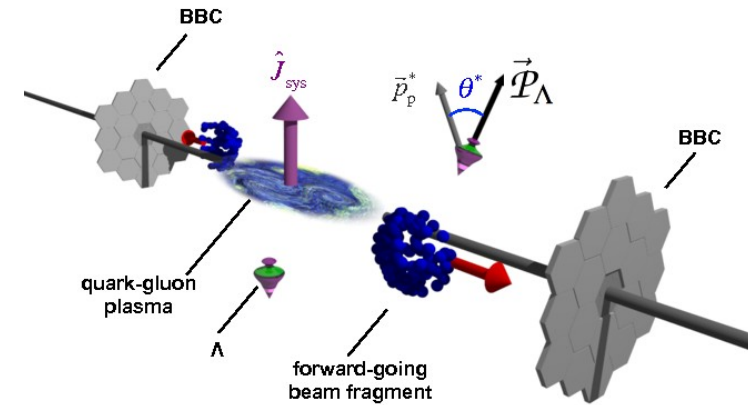


Content

- Introducción
- Reconstrucción del decaimiento
- Diagrama de Armenteros-Podolanski
- Variables angulares
- Resumen

Introduction

- El estudio de partículas con extrañeza es de gran interés en HIC, en particular su polarización.
- La polarización está ligada a las propiedades del medio producidas en HIC.
- Para colisiones semicentrales, el momento angular se puede cuantificar en términos de la vorticidad térmica
- La polarización puede medirse a través del decaimiento cargado de Lambda

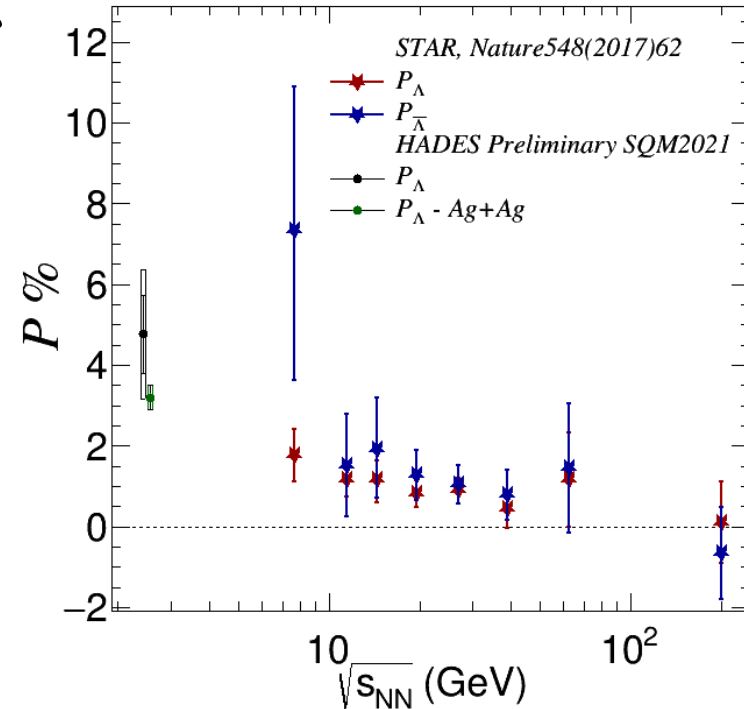


El fluido tiene una estructura giratoria orientada en la dirección del momento angular total.

[Nature 548,62-65(2017)]

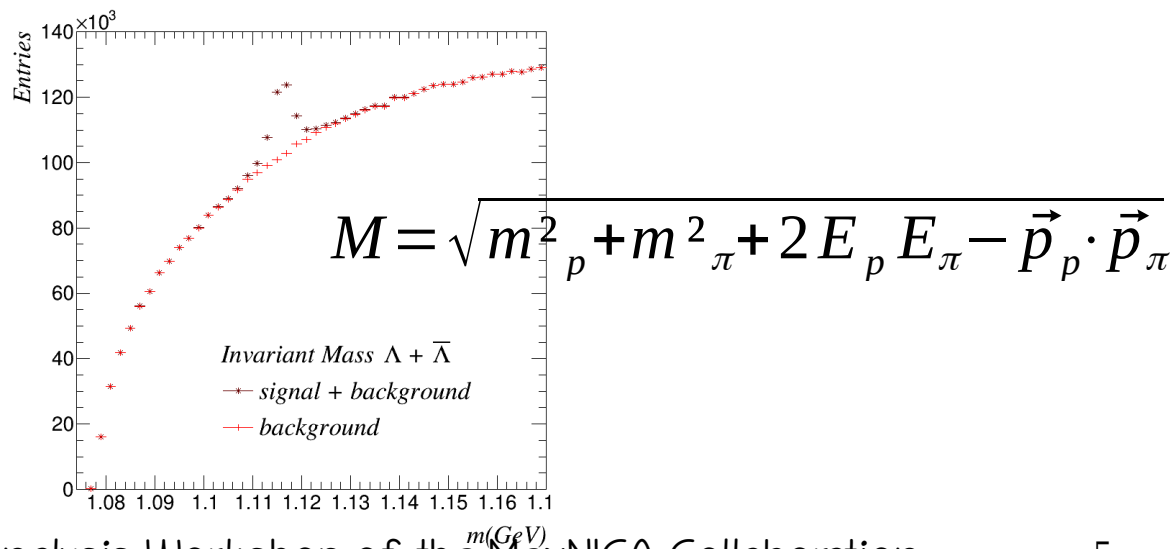
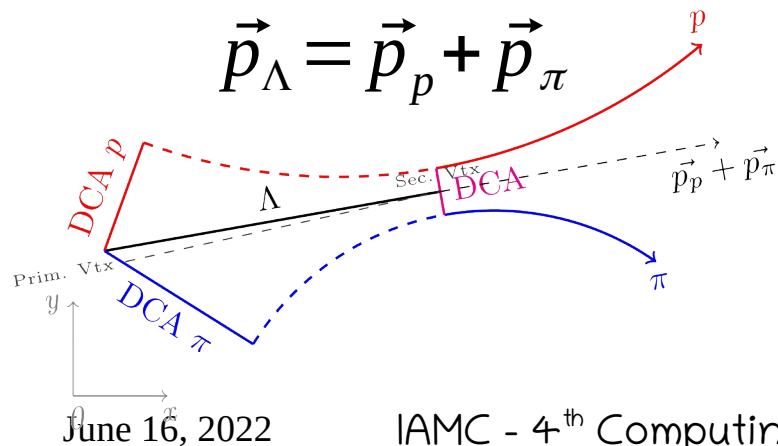
Introduction

- La polarización está ligada a las propiedades del medio producidas en HIC.
- Crece conforme la energía de la colisión decrece
- AntiLambda mayor polarización que Lambda
- Experimentos STAR BES-II + FXT (3-19 GeV), HADES (2-3 GeV) y NICA con el MPD (4-11 GeV) buscan medir la polarización a bajas energías



Reconstrucción de Λ

- A través de su decaimiento cargado. $\Lambda \rightarrow p^+ + \pi^-$
- Combinación de pares de trazas de carga opuesta



Código para reconstruir Λ

- Procedimiento para MCTracks
- Caso de MpdTracks es similar

```
Int_t ntracksmc = MCTracks->GetEntriesFast();
for (int iTr=0; iTr<ntracksmc; iTr++)
{
    auto mctrack = (MpdMCTrack*) MCTracks->UncheckedAt(iTr);

    Double_t ptmc=mctrack->GetPt();
    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));

    Int_t abspdg=TMath::Abs(mctrack->GetPdgCode());
    Int_t fpgd=mctrack->GetPdgCode();

    //loop to choose only secondaries
    if(mctrack->GetMotherId()!=-1){
        if((fpgd==2212) || (fpgd == 211)){ // candidate to positive Lambda's daughter
            auto mctrackmom = (MpdMCTrack*) MCTracks->UncheckedAt(mctrack->GetMotherId()); // mother particle

            Int_t mompdg = mctrackmom->GetPdgCode();

            if(TMath::Abs(mompdg)==3122)hmassL->Fill(mctrackmom->GetMass());

            if(TMath::Abs(mompdg)!=3122)continue;
        }
    }
    for(Int_t iTrn=0; iTrn<ntracksmc; iTrn++){ // candidate to negative Lambda's daughter
        if(iTrn == iTr)continue;
        auto mctrackn = (MpdMCTrack*) MCTracks->UncheckedAt(iTrn);

        if (mctrackn->GetMotherId()!= mctrack->GetMotherId())continue;
        Int_t npdg=mctrackn->GetPdgCode();

        if((npdg== -2212) || (npdg == -211)){

            // positive daughter variables
            Double_t masspos=mctrack->GetMass();
            Double_t Enpos=mctrack->GetEnergy();
            // negative daughter variables
            TVector3 Pn(mctrackn->GetPx(),mctrackn->GetPy(),mctrackn->GetPz());
            Double_t massneg=mctrackn->GetMass();
            Double_t Enneg=mctrackn->GetEnergy();

            // lambda variables
            TVector3 PLa(mctrackmom->GetPx(),mctrackmom->GetPy(),mctrackmom->GetPz());
```

Código para reconstruir Λ

- Procedimiento para MpdMCTracks
- Contaminación por los canales de decaimiento que incluyen más de 1 pión

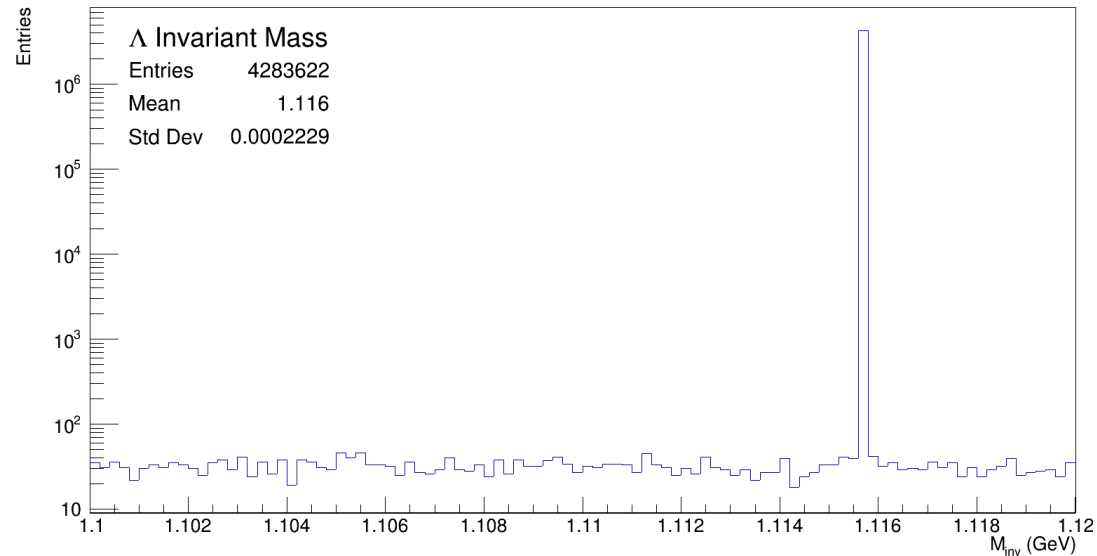
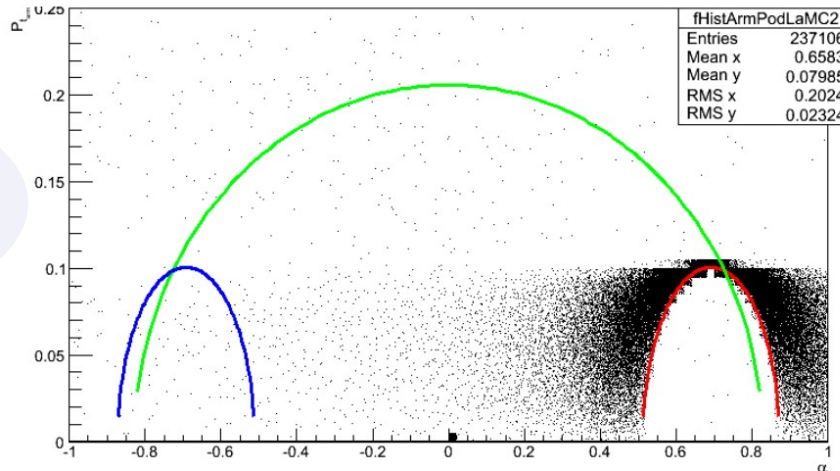


Diagrama de Armenteros-Podolanski

- Gráfica del momento transverso de las partículas hijas vs asimetría del momento longitudinal con respecto a la dirección de vuelo de Λ .

$$\alpha = \frac{p_L^p - p_L^\pi}{p_L^p + p_L^\pi} \quad \text{vs} \quad p_T$$

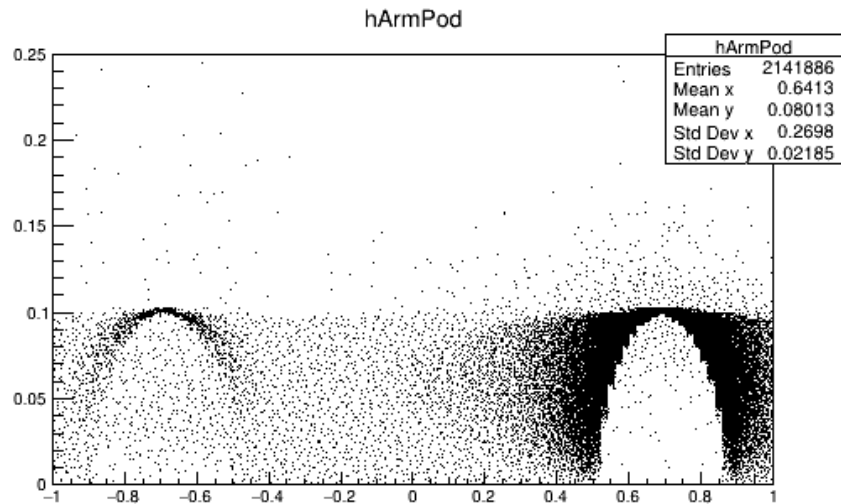


Separar Lambda de antiLambda

Corte en masa invariante, limpia la hemi-elipse

Código para extraer α y p_T .

Corte en α permite
diferenciar Λ de $\bar{\Lambda}$.



```
// positive daughter variables
Double_t masspos=mctrack->GetMass();
Double_t Enpos=mctrack->GetEnergy();
// negative daughter variables
TVector3 Pn(mctrackn->GetPx(),mctrackn->GetPy(),mctrackn->GetPz());
Double_t massneg=mctrackn->GetMass();
Double_t Enneg=mctrackn->GetEnergy();

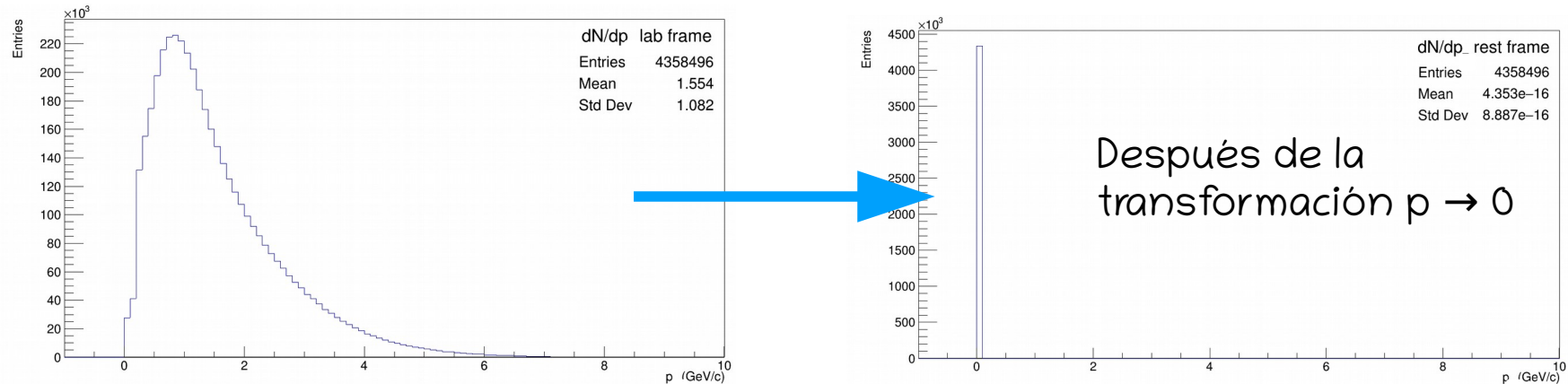
// lambda variables
TVector3 PLa(mctrackmom->GetPx(),mctrackmom->GetPy(),mctrackmom->GetPz());
Double_t EnLa=mctrackmom->GetEnergy();

//ArmPod variables
Float_t      lQlNeg = Pn.Dot(PLa)/PLa.Mag();
Float_t      lQlPos = P.Dot(PLa)/PLa.Mag();
Float_t      alpha = (lQlPos - lQlNeg)/(lQlPos + lQlNeg);
Float_t      ptarm = Pn.Perp(PLa);

hArmPod->Fill(alpha, ptarm);
```

Distribución angular del protón.

- Para medir la distribución angular consideramos el sistema de referencia de la Λ en reposo, lo que requiere una T. de Lorentz



Transformación de Lorentz

- TLorentz también permite hacer la transformación

```
// positive daughter variables
Double_t masspos=mctrack->GetMass();
Double_t Enpos=mctrack->GetEnergy();
// negative daughter variables
TVector3 Pn(mctrackn->GetPx(),mctrackn->GetPy(),mctrackn->GetPz());
Double_t massneg=mctrackn->GetMass();
Double_t Enneg=mctrackn->GetEnergy();

// lambda variables
TVector3 PLa(mctrackmom->GetPx(),mctrackmom->GetPy(),mctrackmom->GetPz());
Double_t EnLa=mctrackmom->GetEnergy();
TVector3 lBeta = (1./EnLa)*PLa;
Double_t MagBeta = lBeta.Mag();
Double_t lgamma = 1.0/TMath::Sqrt(1.0 - (MagBeta*MagBeta));

// Restframe vectors - Lorentz transformations
Double_t ldotPB = (PLa.Dot(P))/EnLa;
TVector3 lPRF = P + ((lgamma-1.0)/:pow(MagBeta,2))*ldotPB*lBeta - lgamma*lBeta*Enpos;

Double_t ldotLL = (PLa.Dot(PLa))/EnLa;
TVector3 lPLaRF = PLa + ((lgamma-1.0)/:pow(MagBeta,2))*ldotLL*lBeta - lgamma*lBeta*EnLa;
```

```
// Variables for Angular distributions
```

```
Double_t PsiEP = MCHheader->GetRotZ();
TVector3 maL(TMATH::Sin(PsiEP),-TMATH::Cos(PsiEP),0);
```

```
// angle shift
```

```
Double_t phip;
if(lPRF.Y() >= 0){
    phip=TMATH::ATan2(lPRF.Y(),lPRF.X());
}else{
    phip=2*TMATH::Pi() + TMATH::ATan2(lPRF.Y(),lPRF.X());
}
```

```
Double_t philab;
```

```
if(P.Y() >= 0){
    philab=TMATH::ATan2(P.Y(),P.X());
}else{
    philab=2*TMATH::Pi() + TMATH::ATan2(P.Y(),P.X());
}
```

```
Double_t deltaphi;// = -99.0;
```

```
if((PsiEP-phi) >= 0){
    deltaphi = PsiEP - phip;
}else{
    deltaphi = 2*TMATH::Pi() + (PsiEP-phi);
}
```

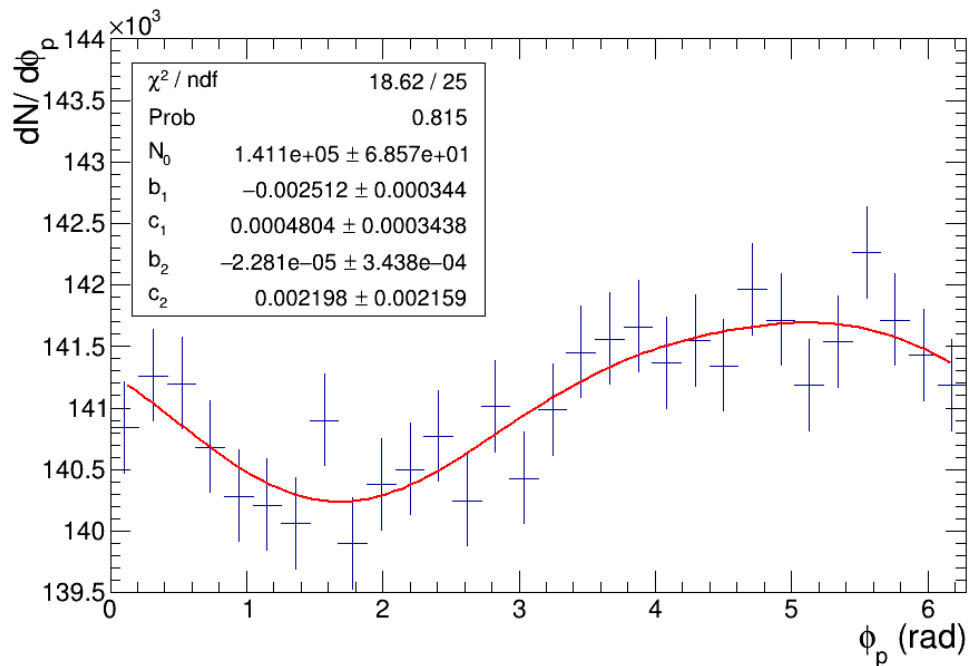
```
hAngleP->Fill(philab);
hAngleEP->Fill(PsiEP);
hDeltap->Fill(deltaphi);
```

Distribuciones angulares

PHSD BiBi - $\sqrt{s_{NN}}=9.0$ GeV

■ $b \in (6,10)$ fm

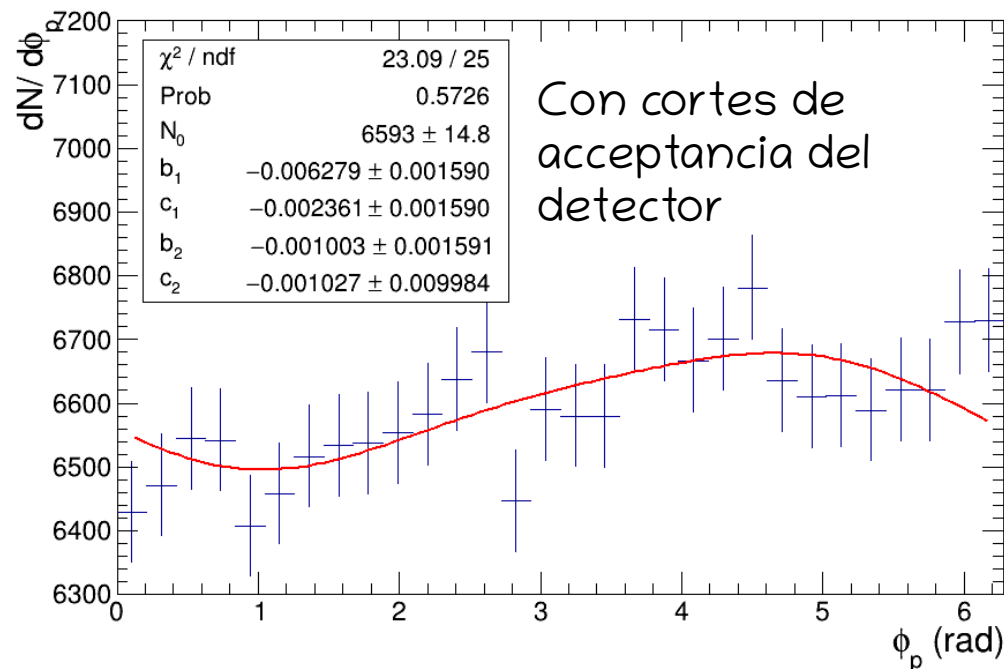
— $N_0(1+2b_1\sin(\Delta\phi_p)+2c_1\cos(\Delta\phi_p)+2b_2\sin(2\Delta\phi_p)+2c_2\cos(2\Delta\phi_p))$



PHSD BiBi - $\sqrt{s_{NN}}=9.0$ GeV

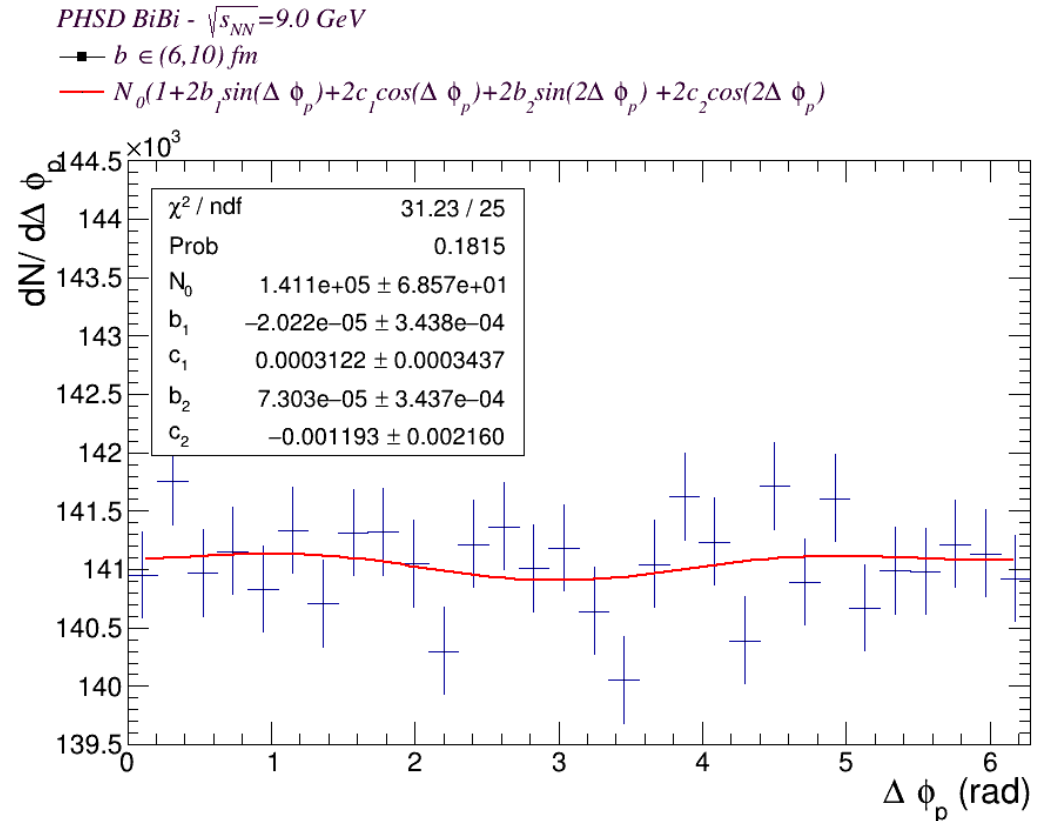
■ $b \in (6,10)$ fm

— $N_0(1+2b_1\sin(\Delta\phi_p)+2c_1\cos(\Delta\phi_p)+2b_2\sin(2\Delta\phi_p)+2c_2\cos(2\Delta\phi_p))$



Distribuciones angulares $\Delta\phi$

- El análisis actual requiere una revisión de la selección del intervalo $\Delta\Lambda$

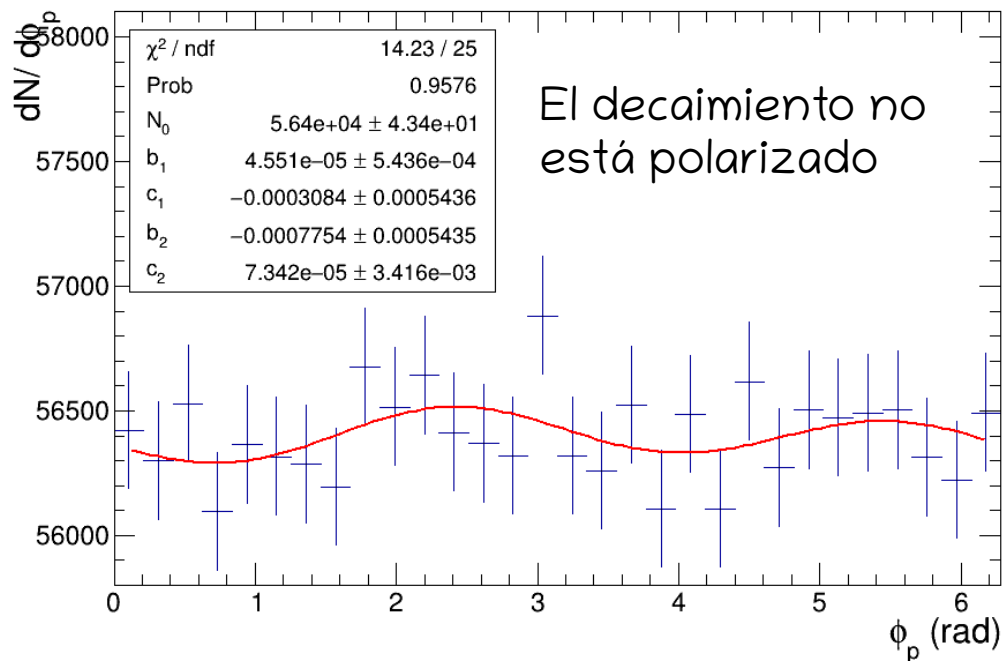


Comparación con UrQMD

UrQMD BiBi - $\sqrt{s_{NN}}=9.2$ GeV

■ $b \in (6,10)$ fm

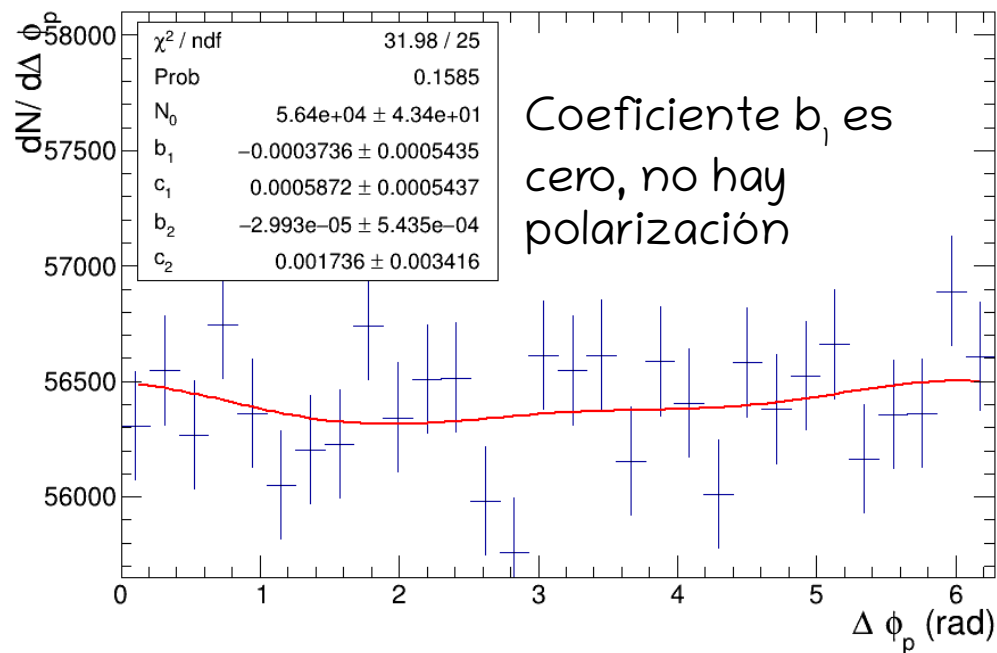
— $N_0(1+2b_1\sin(\Delta\phi_p)+2c_1\cos(\Delta\phi_p)+2b_2\sin(2\Delta\phi_p)+2c_2\cos(2\Delta\phi_p))$



UrQMD BiBi - $\sqrt{s_{NN}}=9.2$ GeV

■ $b \in (6,10)$ fm

— $N_0(1+2b_1\sin(\Delta\phi_p)+2c_1\cos(\Delta\phi_p)+2b_2\sin(2\Delta\phi_p)+2c_2\cos(2\Delta\phi_p))$



Resumen

- Macro de análisis simplificado para reconstrucción de Lambdas a través de su decaimiento cargado, usando MpdMCTracks
- Este procedimiento se puede ampliar a MpdTracks, incluyendo cortes para la correcta identificación de trazas secundarias
- Macros en: [git](https://github.com/ivonnemaldonado/Lambda_reconstruction)
https://github.com/ivonnemaldonado/Lambda_reconstruction

Resumen

- Una versión más desarrollada de los macros será puesta en el Tutorial del MPD

Tutorial

- <https://mpdroot.jinr.ru/analysis-tutorial/>

