calo position correction and pt function

Jingyu

pt func of Δϕ and η

- Akiba's suggest:

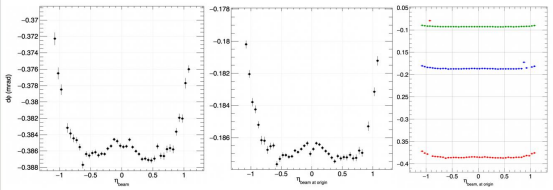
dphi=0.2/pT at eta=0

Dphi=c(¥eta)/pT c(¥eta) = 0.2 at ¥eta=0

c(¥eta)= 0.19? ¥eta=0.5

c(¥eta) = 0.2 + c1*¥eta + c2*¥eta^2

Genki: pt function of angel have eta dependence



[0.5, 1, 2, 10, 15, 25, 50, 100, 200] ->
[0.0961, 0.1922, 0.3844, 1.922, 2.883, 4.805, 9.61, 19.22, 38.44]

Former: pT = 0.1922 / Δϕ
Now: Maybe we need a new function consider the eta dependence and for a fix R

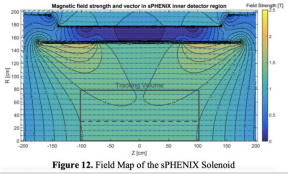
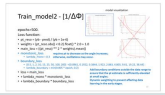


Figure 12. Field Map of the sPHENIX Solenoid

From the discussion on Mattermost:
we can see the dependence of pT on Δϕ, as well as its η dependence.
If there is a better function that describes pT as a function of both Δϕ and η, I could incorporate more accurate boundary conditions, which might lead to improved results.



19

- Assume on func to calculate pt, Δϕ and η are separable,

  F(Δϕ,η) = C(η)*f(Δϕ), and f(Δϕ) = 1/Δϕ.

Then I want to get the C(η) from

- **C(η) = ⟨pT·Δϕ⟩**
- C_value = pT·Δϕ for each electron

$$f(x) = p_0 + p_1\,x + p_2\,x^2 + p_3\,x^3 + p_4\,x^4$$

- Fill the C_value on a TProfile, and fit the TProfile with poly 4.

pT·Δϕ - η distribution

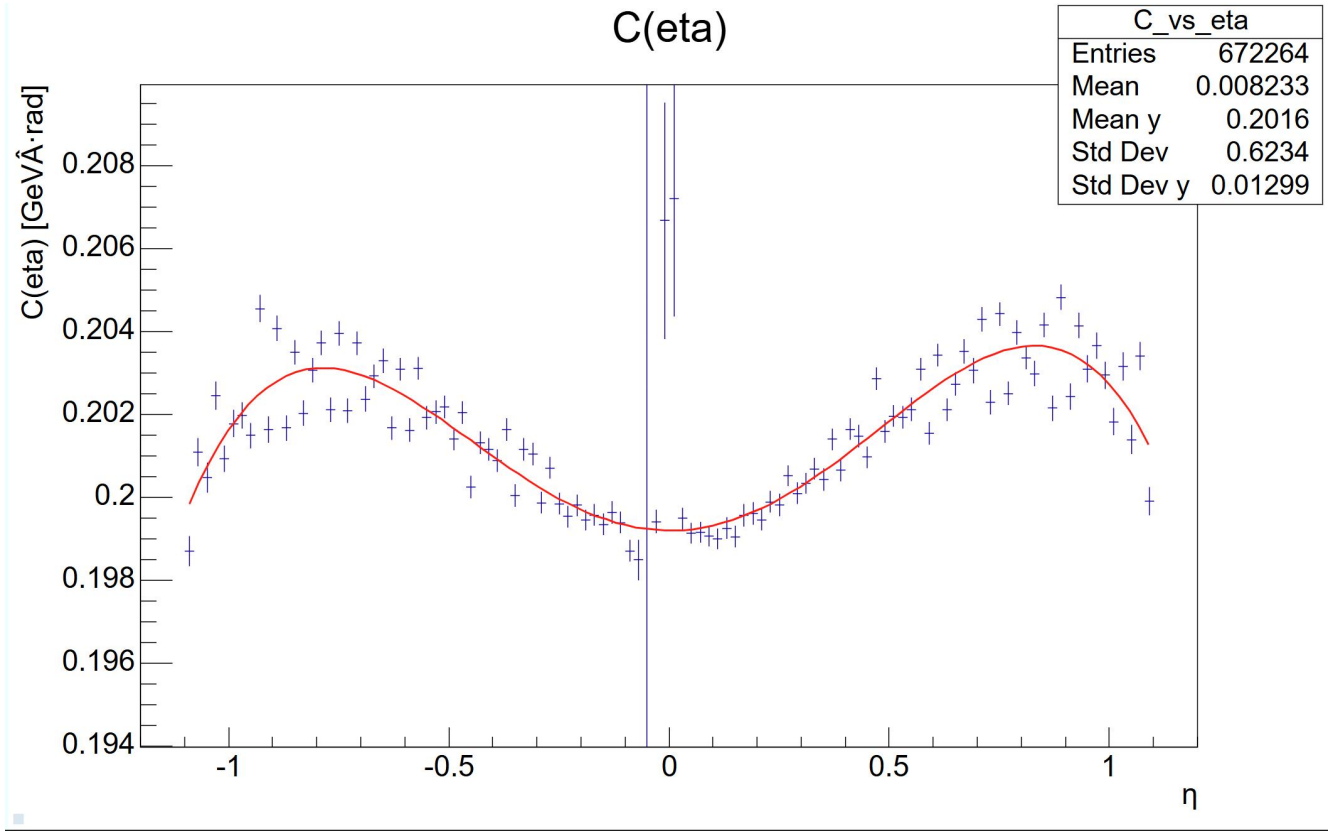
- loop the simulation data, for each electron
  - Get the **truth pT**, Get the primary particle eta **η**
  - Get the **Δϕ** from iINTT, oINTT reco and Primary particle first hit on CEMC position
  - calculate the **C_value = pT·Δϕ**
  - skip the very large and smale C_value, **if (Cval > 0.3 || Cval < 0.15) continue**
  - Fill a TProfile with η and pT·Δϕ**, TProfile *tpr1->Fill(eta, C_value)**
  - Fill a TGraph with η and pT·Δϕ points

$$f(x) = p_0 + p_1\,x + p_2\,x^2 + p_3\,x^3 + p_4\,x^4$$

- Fit the TProfile with Poly4 func (5para. x^0-x^4)
  - loop xbins, if(profC->GetBinContent(ib) > 0.205) profC->SetBinError(ibin, 2e-1)
    adjust the error on odd bin, Minimize its impact on the fit.

- Fit the TGraph with Poly4 func (5para. x^0-x^4)
  - Advantage is that the fit is performed directly on the data points, without any binning.

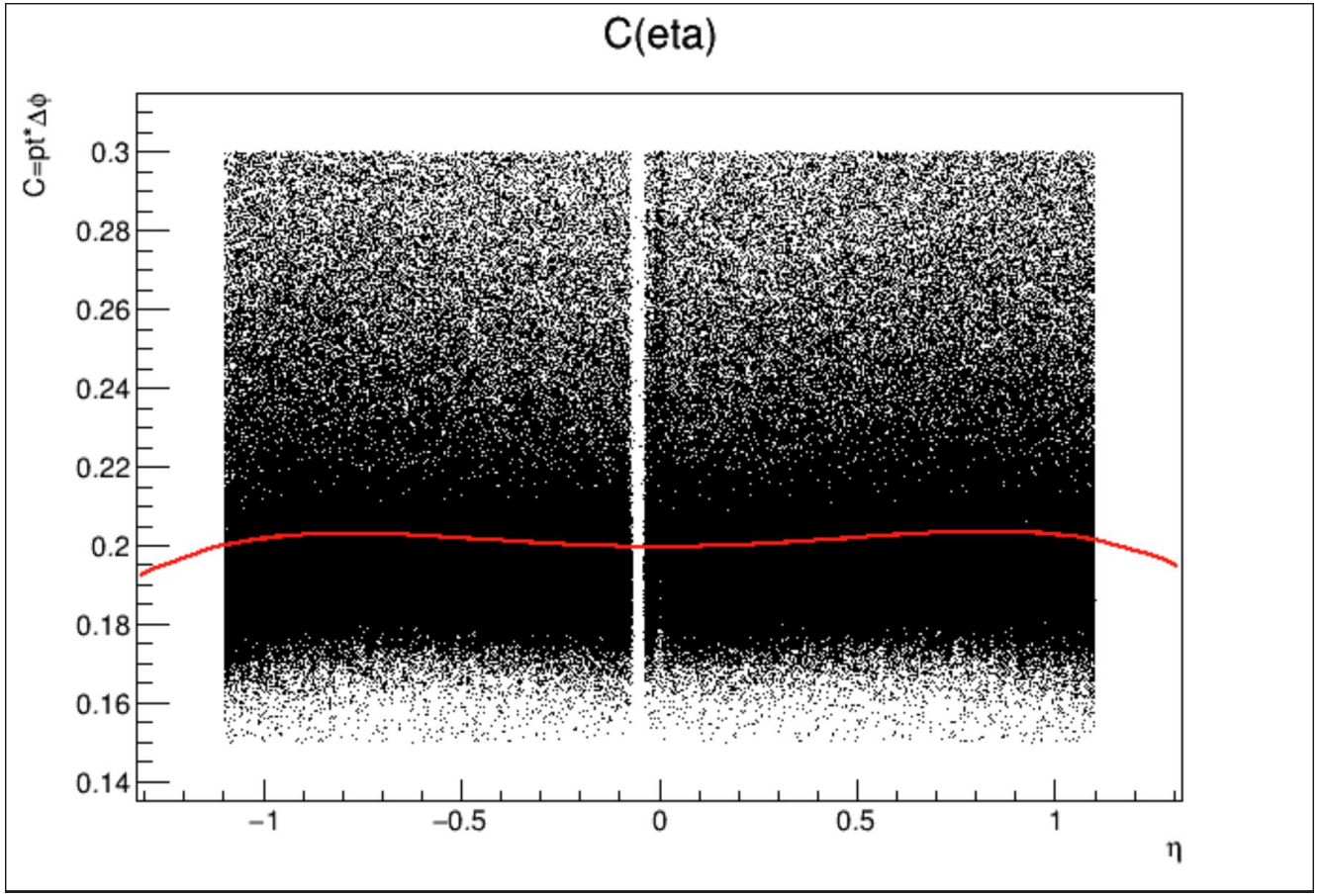
fit func on TProfile



NO.	NAME	VALUE	ERROR
1	p0	1.99198e-01	5.07218e-05
2	p1	-6.38608e-05	1.06560e-04
3	p2	1.28447e-02	2.75958e-04
4	p3	6.06315e-04	1.43027e-04
5	p4	-9.86226e-03	2.63624e-04

$$f(x) = p_0 + p_1\,x + p_2\,x^2 + p_3\,x^3 + p_4\,x^4$$

fit func on TGraph

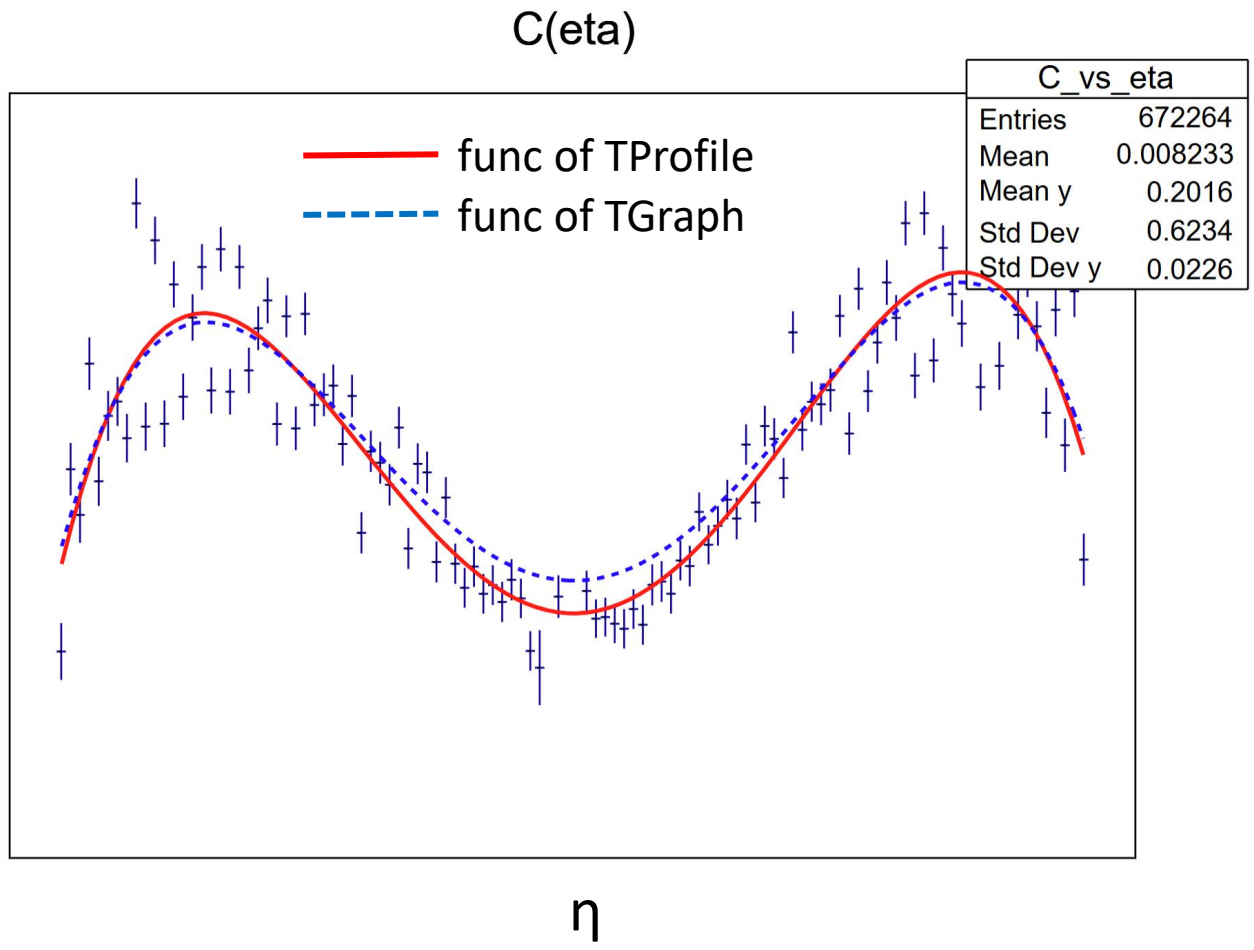


p0	=	0.199626	+/-	5.178e-05
p1	=	-6.95944e-05	+/-	0.000107422
p2	=	0.0111714	+/-	0.00027129
p3	=	0.000603954	+/-	0.000139037
p4	=	-0.00859856	+/-	0.000253666

$$f(x) = p_0 + p_1\,x + p_2\,x^2 + p_3\,x^3 + p_4\,x^4$$

Compare the func.s

$C = pT \cdot \Delta\phi$



func of TGraph compare with func ofTProfile without Give a large error on odd bin, Conside the odd point -> ~0 η range, unc of TGraph larger than unc of TGraph

Thanks

0716

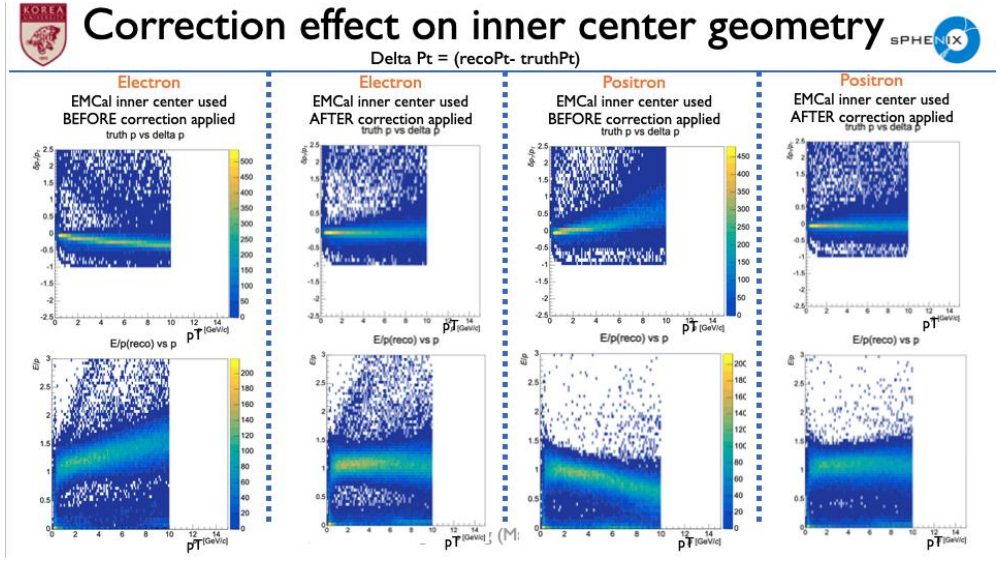
- Jingyu

EMCal position phi reco and correct

- Cluster Position Study Without Magnetic Field Previously
- Other studys are with Magnetic Field
  - combined position and energy reconstruct pt
  - Machine learning reconstruct pt
  - pt func with eta-dependence
- In the earlier scenario where we used the cluster inner-face center, the correction we applied to the position was inaccurate.

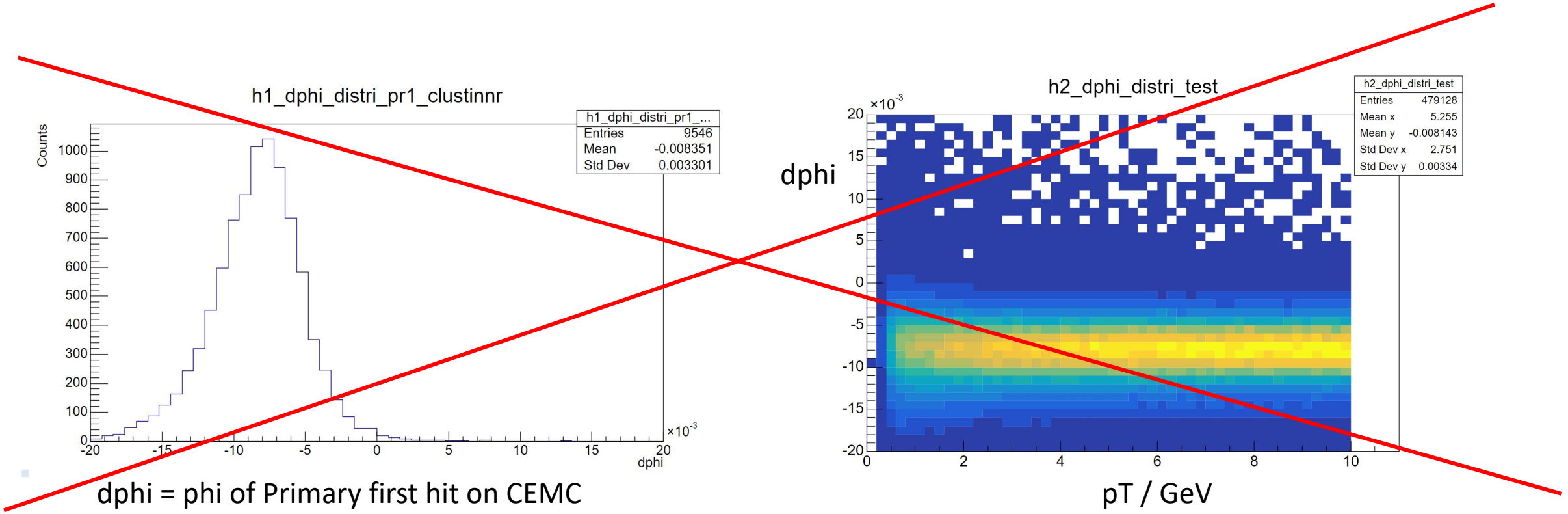
I used an inappropriate configuration(G4Setup_sPHENIX.C file).

I notice there is no magnetic field, use another G4Setup_sPHENIX.C file (now same as Jaein Genki ‘s)



Based on the previous pT reconstruction results, our correction is effective but not sufficiently accurate.
Since the magnetic field alters the incident direction, it affects the EMCal reconstruction of charged particles.
so I need to update the results using the more accurate correction

Cluster Position Study Without a Magnetic Field Previously



dphi = phi of Primary first hit on CEMC
- phi of Cluster reco with tower inner face center

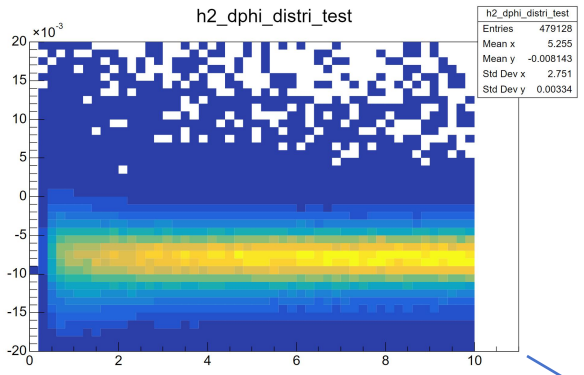
These events used in position correction study with a wrong magnetic field configure:
wrong Magnetic field setting
-> without charge effect
-> wrong correction for cluster reco with tower inner face center

dphi = phi of Primary particle first hit on CEMC

- phi of Cluster reco with tower inner face center

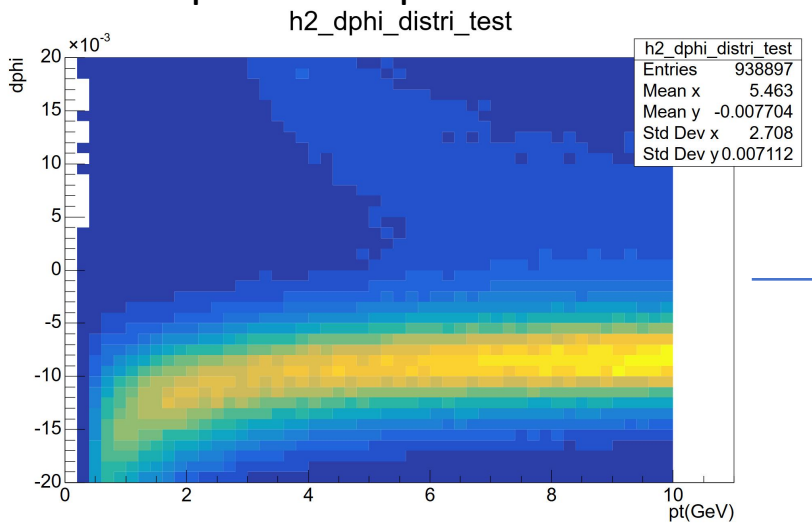
For using in real data, we need correct method base on observable variable

without Magnetic field

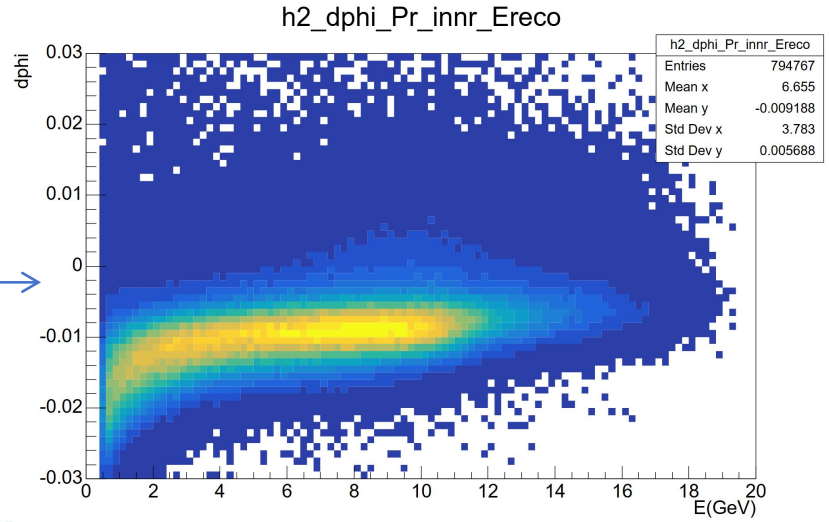


with M field electron:

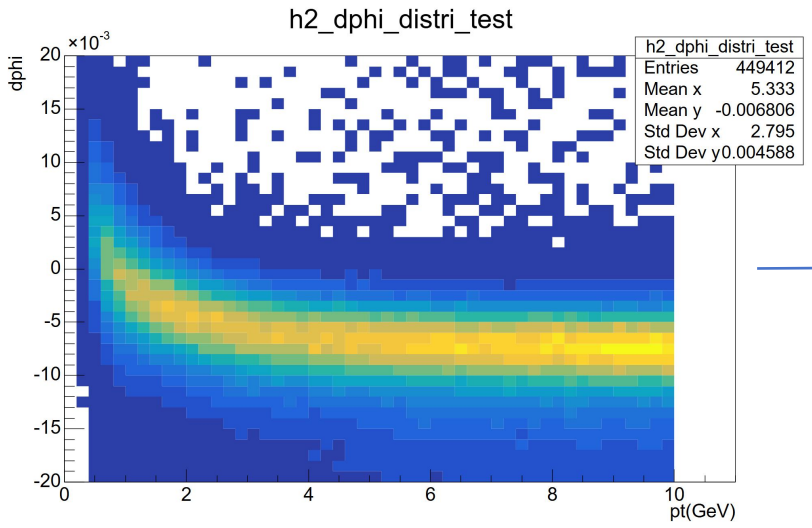
dphi - truth pT

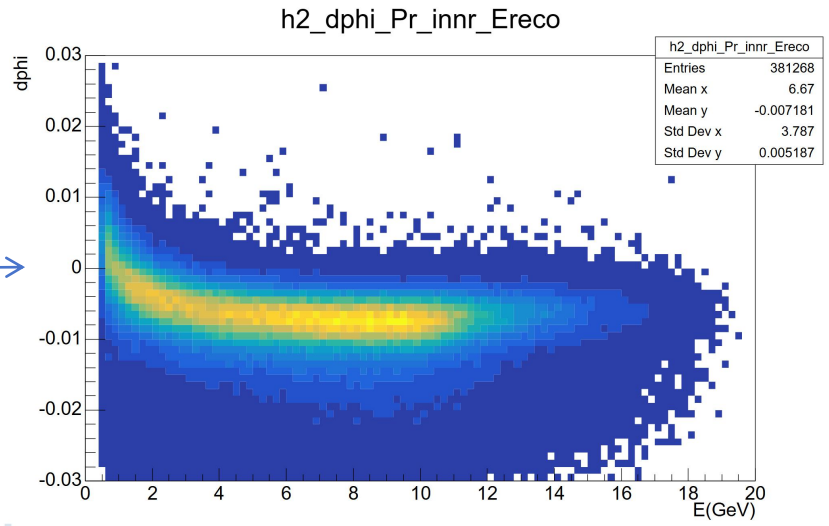


dphi - reconstrution Energy



with M field positron:

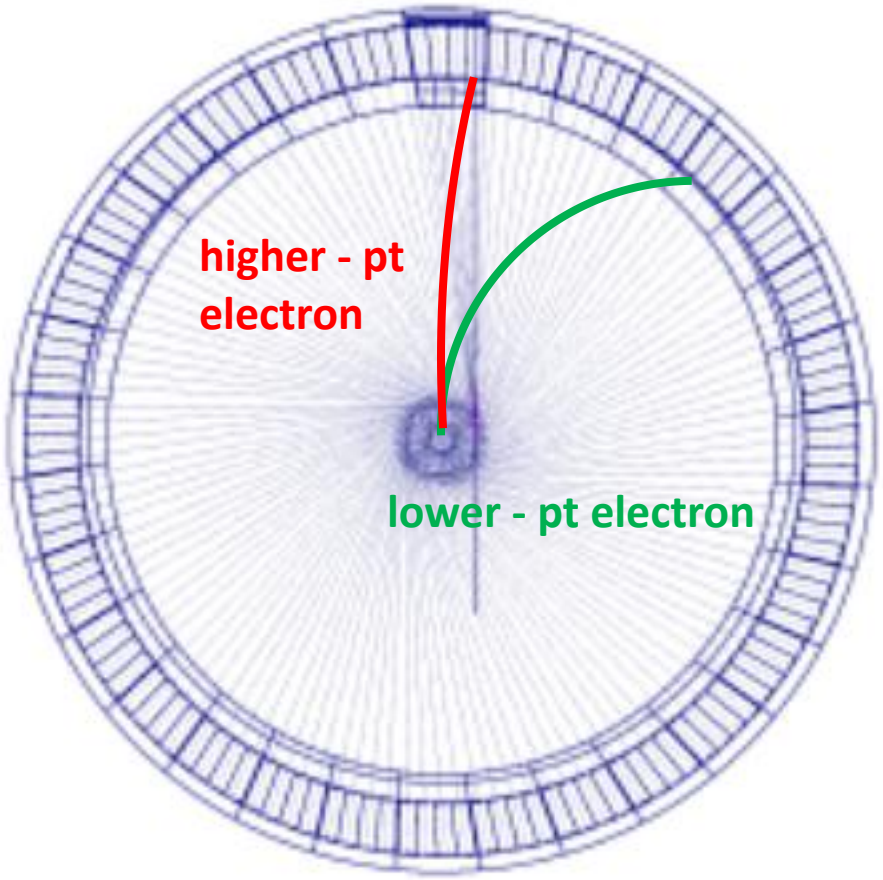


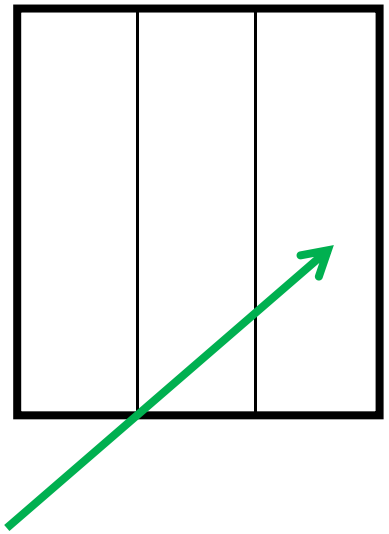


we can see the difference after correct Magnetic field from charge effect.

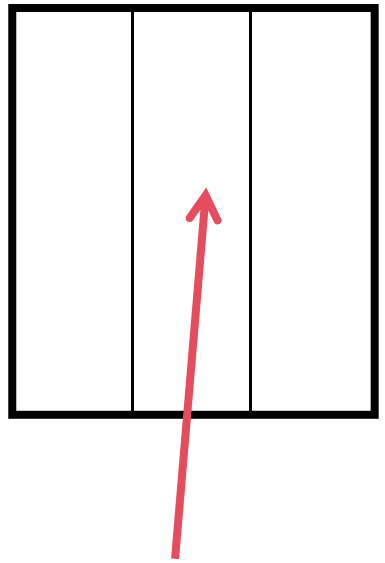
Explanation of Magnetic effect on EMCal response

For low-pt electrons and positrons, the magnetic field significantly change their pt direction, causing them to hit the EMCal at a large angle.
In contrast, high-pt particles experience only a small change in incident direction and are less affected by the magnetic field.

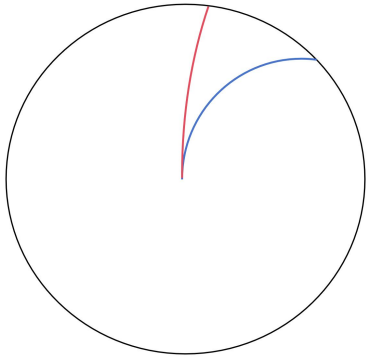




lower - pt electron



higher - pt electron

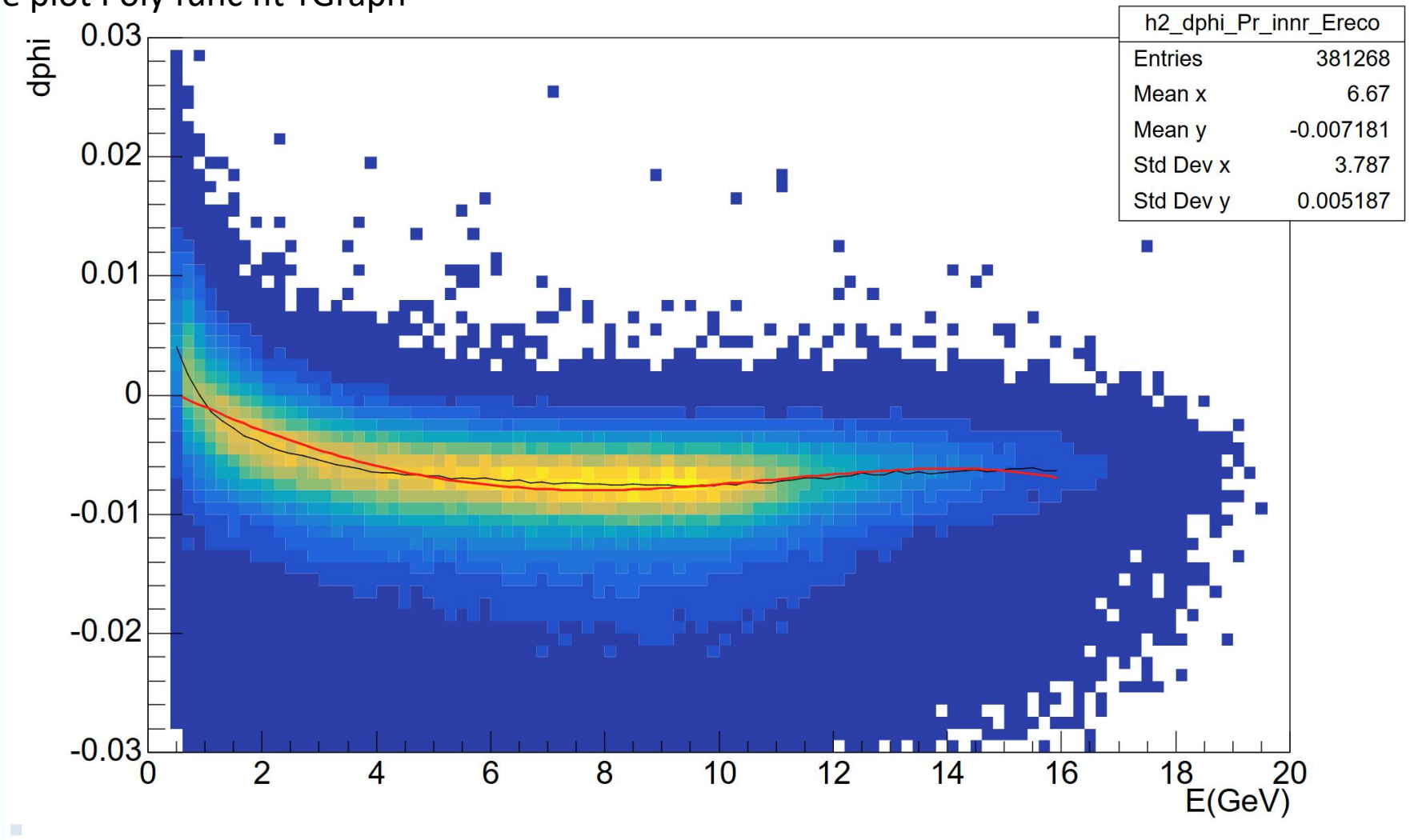


Get correction

compare the two curve, we can see directly using points is better than fitting results:
smooth enough (Donnot worry about interpolation problem), more accurate.

curve plot TGraph of points (Energy, peak value)

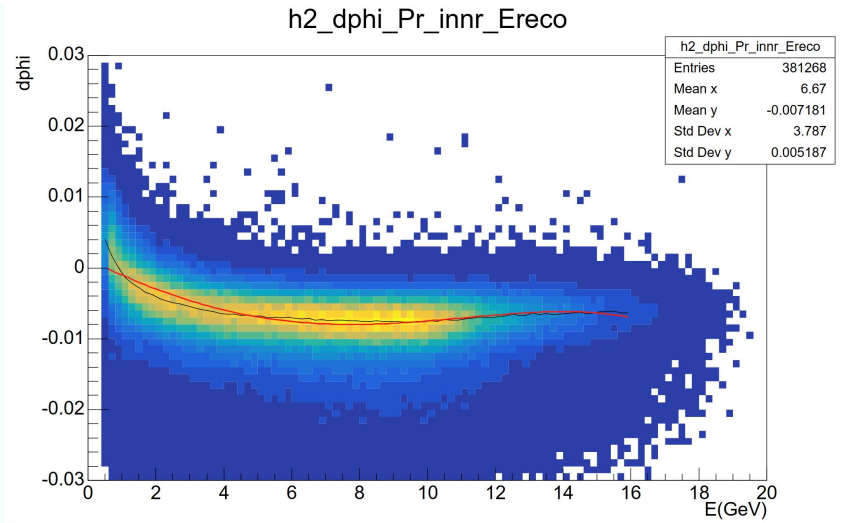
curve plot Poly func fit TGraph



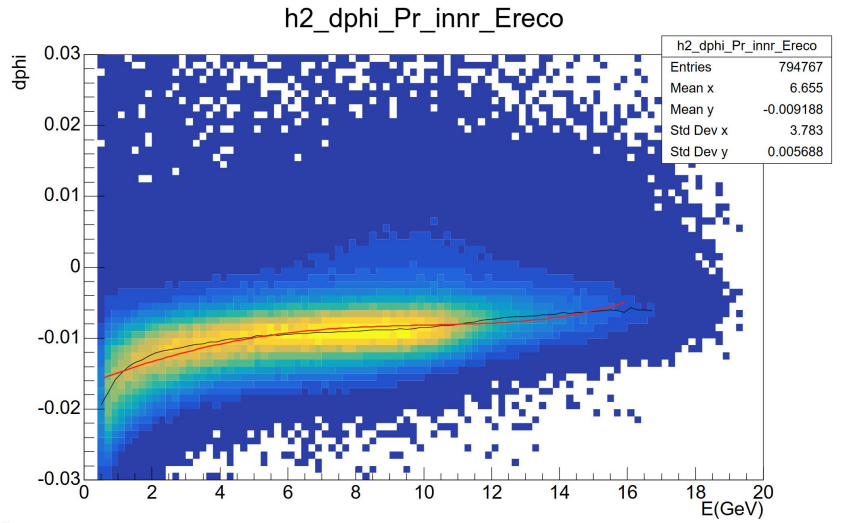
I use TGraph the black line interpolation by reconstruction cluster Energy to correct the phi reconstrution

Correction cluster position reco with tower inner face center

positron:



electron:

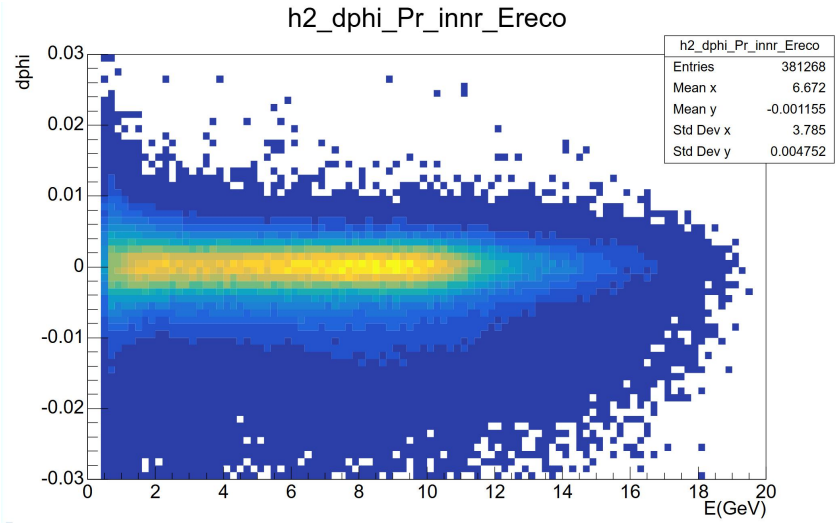


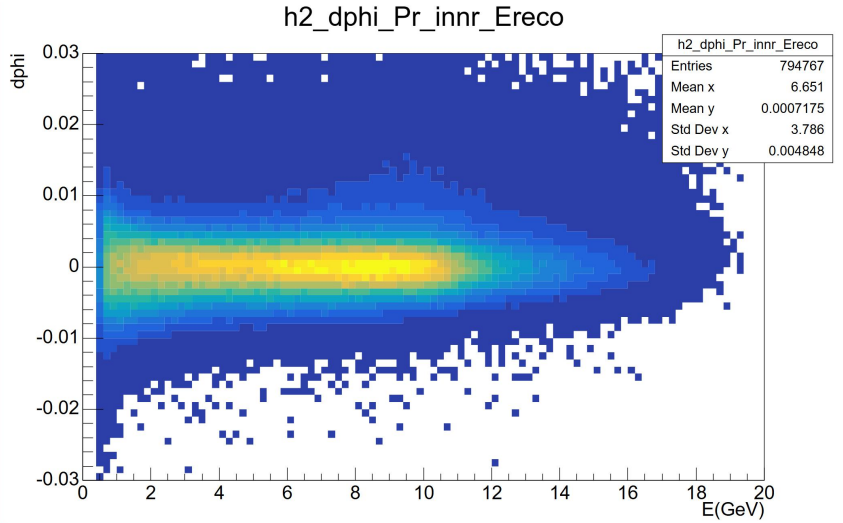
From dphi - reconstrution Energy distribution project each x-bin on y-axis, found the peak value.

Get the points(Energy, dphi peak value), show is black line, then fit the points with poly func to get the correction, the red line

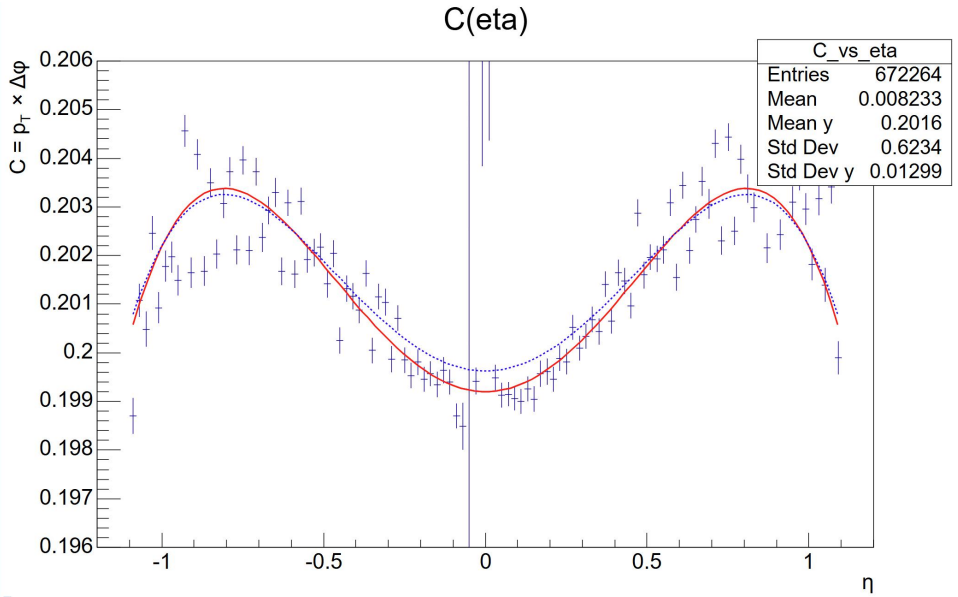
I use TGraph the black line interpolation by reconstruction cluster Energy to correct the phi reconstrution

After correction





1st rough fit with a larger width -- **(0.15 < Cval < 0.3)**



Angle between 2 lines
line connect EMC and oINTT
line connect iINTT oINTT

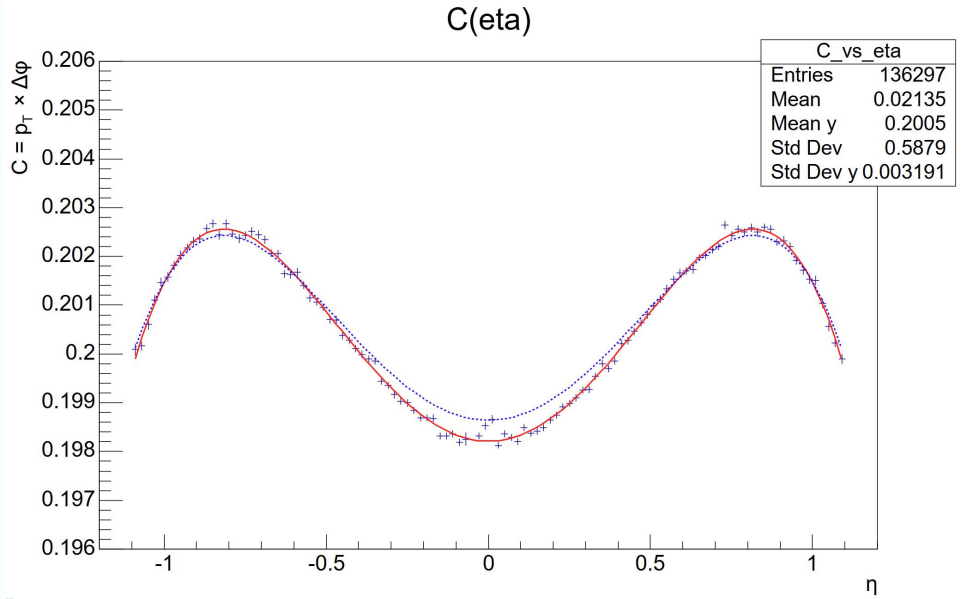
iINTT oINTT reconstruction position
Primary first hit on CEMC

func of TProfile
func of TGraph

$$f(x) = p_0 + p_1\,x^2 + p_2\,x^4$$

Remove asymmetry terms, Only the even terms

2rd fit with a narrow range -- - **(rough func - 0.005 < Cval < rough func + 0.005)**



these 2 func can be used in pt reconstrution

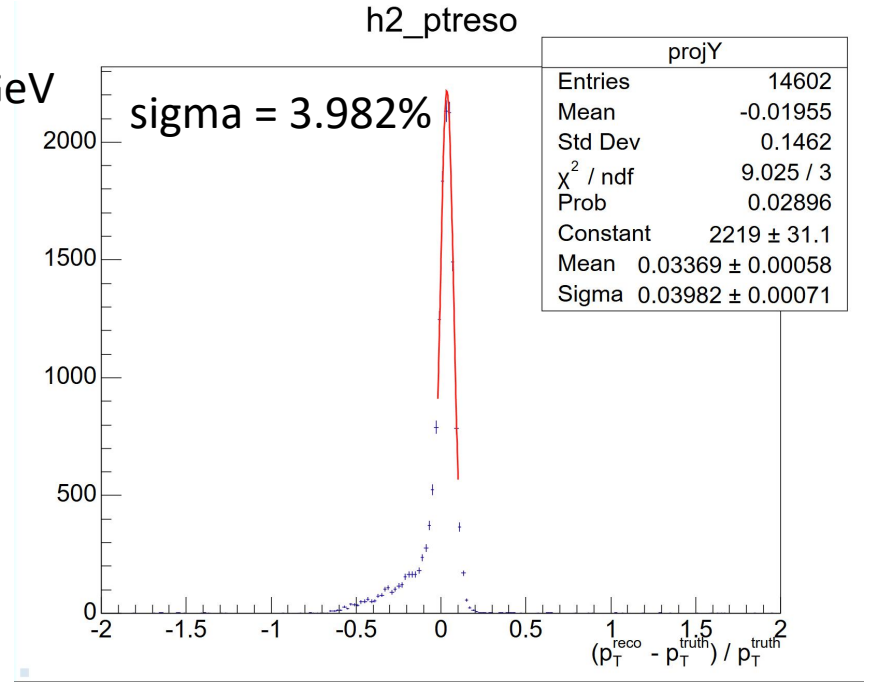
ptFunc1 Performance

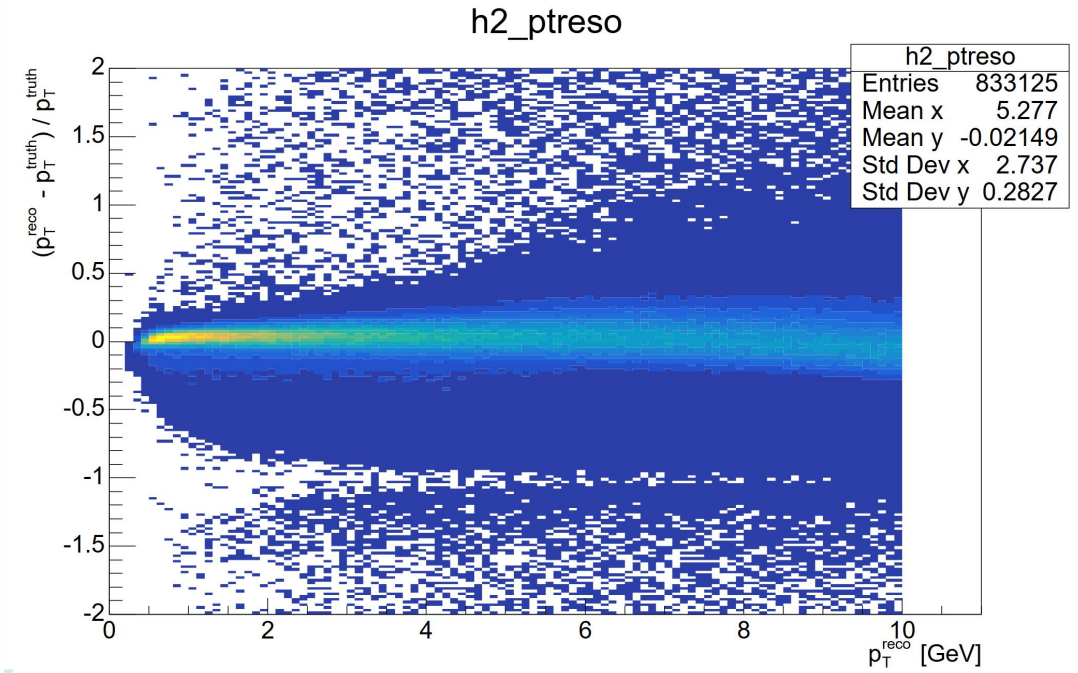
Angle between 2 lines
line connect EMC and oINTT
line connect iINTT oINTT

iINTT oINTT reconstruction position
Cluster reco with tower inner face center after correction

Func from TProfile

0.9-1.1 GeV





~4% pt resolution on 0.9-1.1GeV

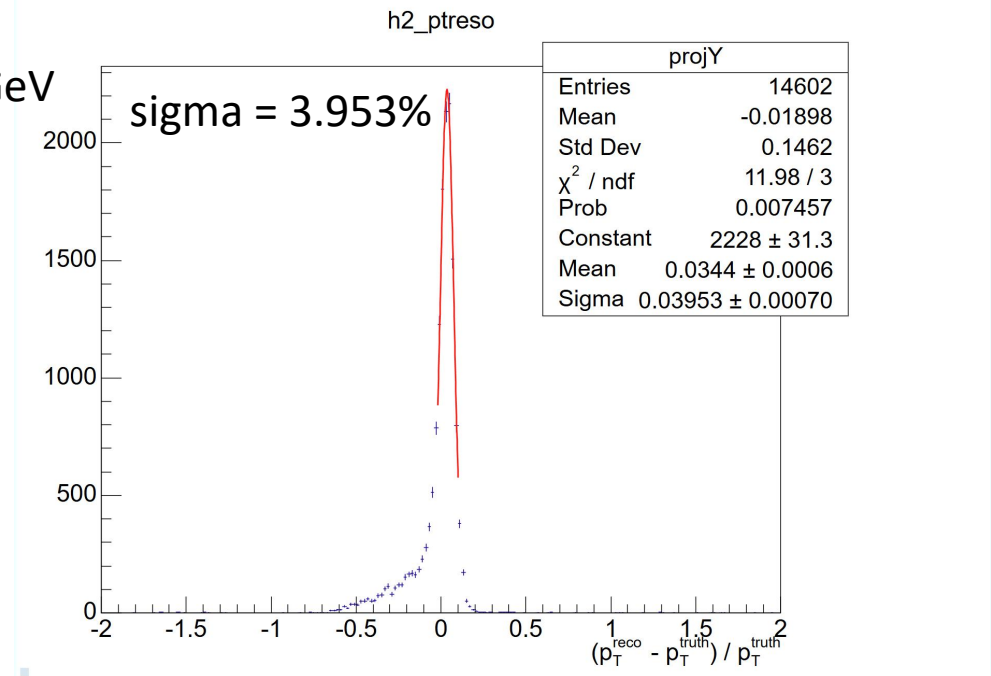
ptFunc2 Performance

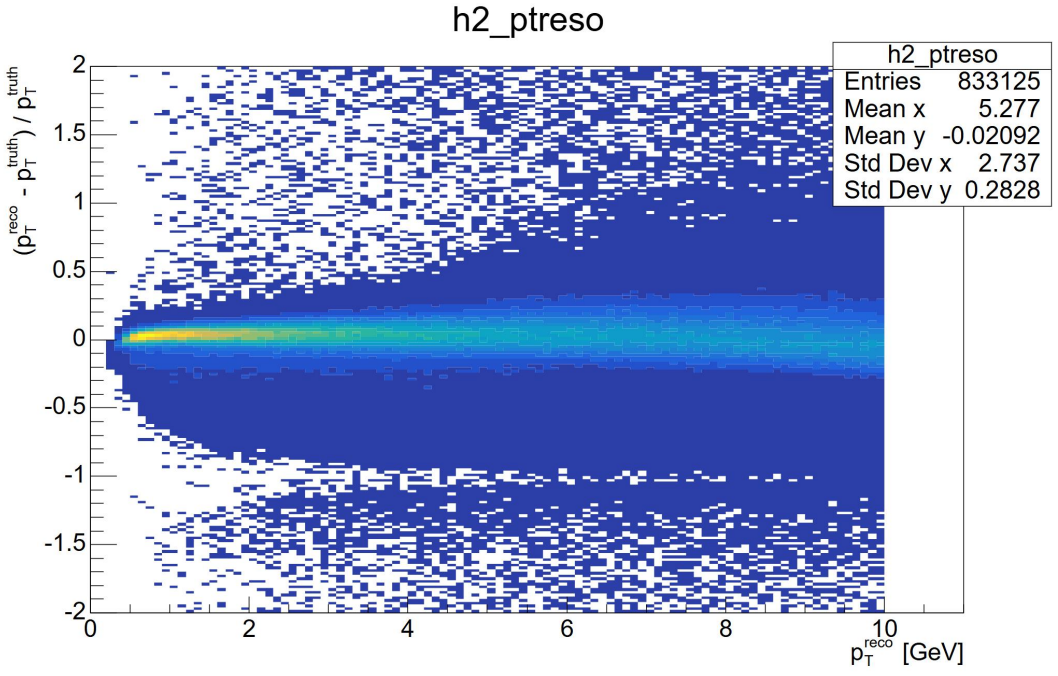
Angle between 2 lines
line connect EMC and oINTT
line connect iINTT oINTT

iINTT oINTT reconstruction position
Cluster reco with tower inner face center after correction

Func from TGraph

0.9-1.1 GeV





~4% pt resolution on 0.9-1.1GeV

Both two func have similar performance,
pt resolution about 4% on 0.9-1.1 GeV

performance summary:

Both two func have similar performance,
pt resolution about 4% on 0.9-1.1 GeV

Thanks for your attention

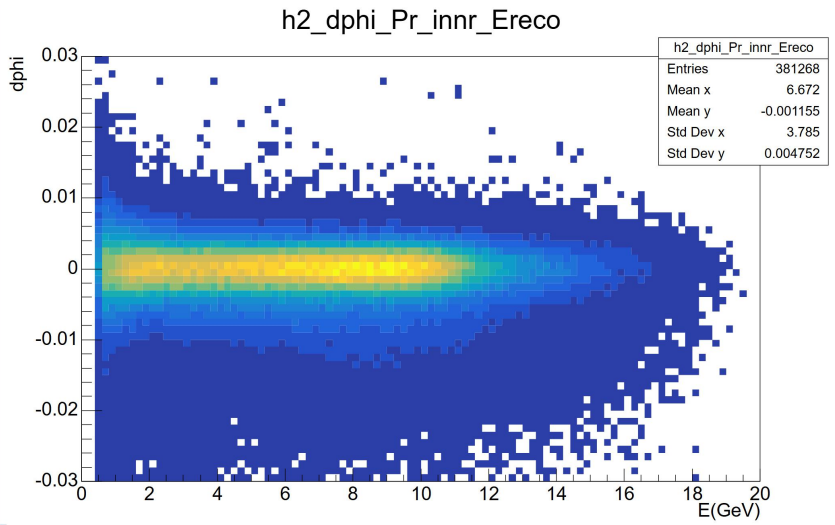
Phi reconstruct bias and resolution - pt

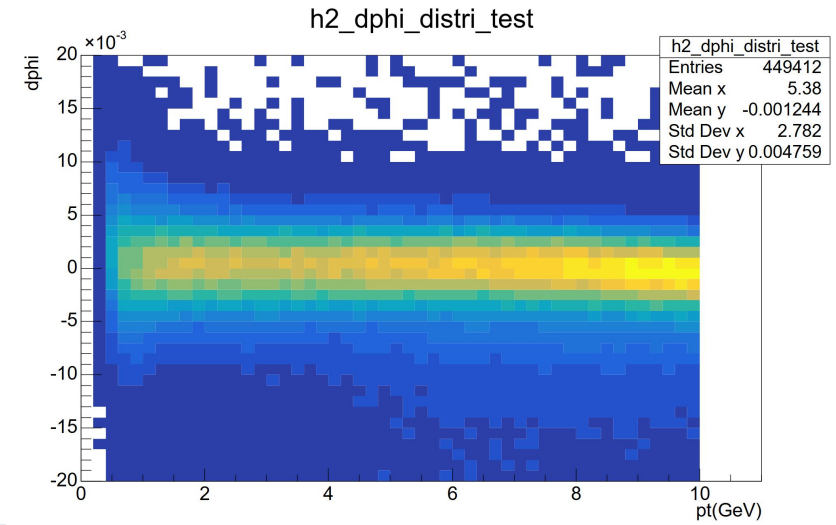
- Jingyu 0723

dphi - E and dphi - pt

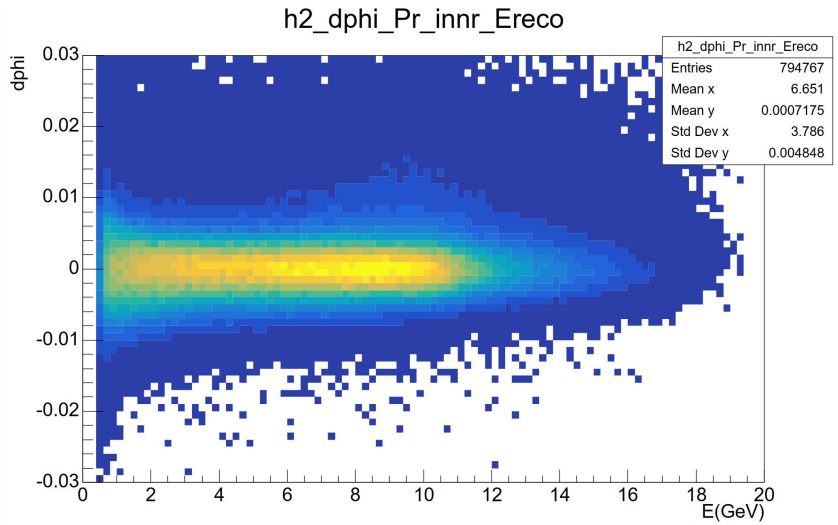
dphi = phi of Primary particle first hit on CEMC
- phi of Cluster reco with tower inner face center

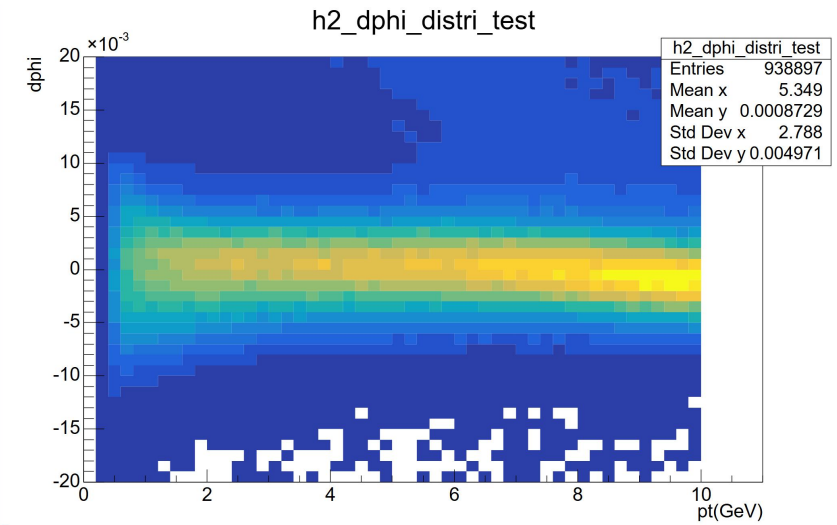
positron:





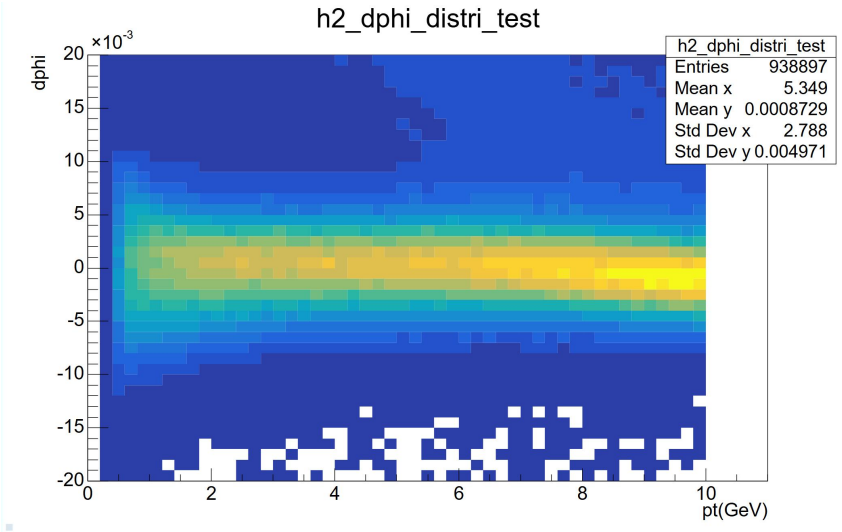
electron:



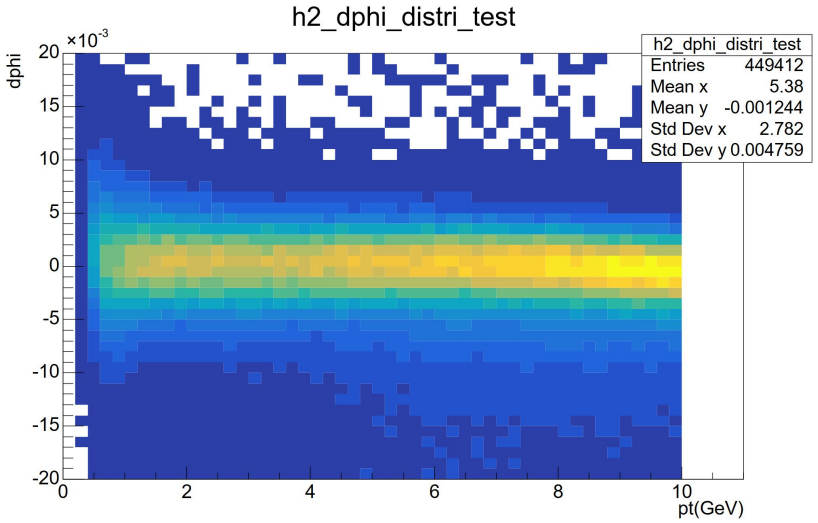


dphi(truth - reco) vs pT

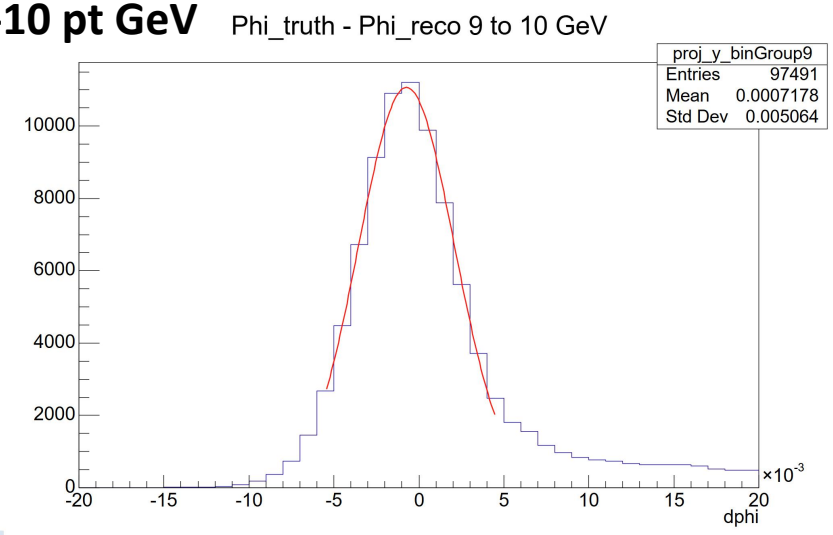
Electron

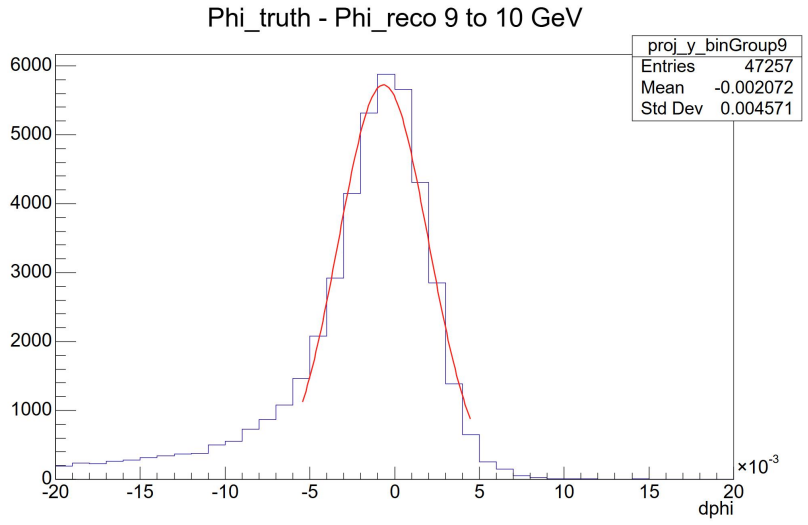


Positron



10 bins for 0-10 pt GeV





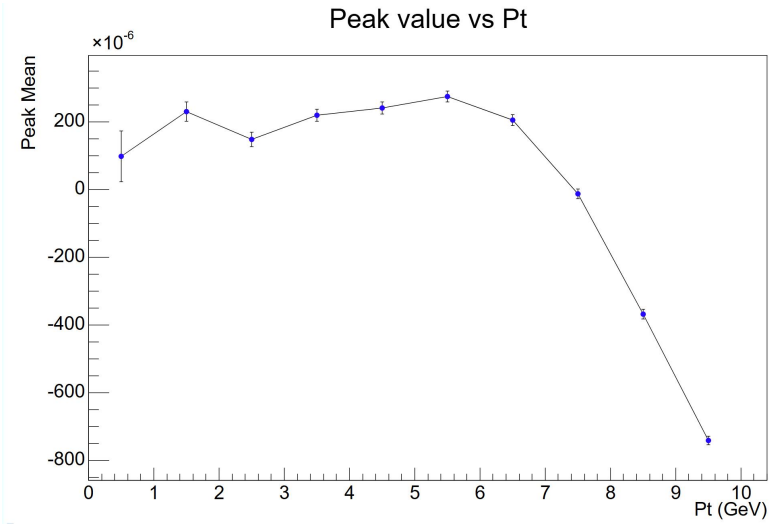
Fit to get peak value (bias) and width (resolution)

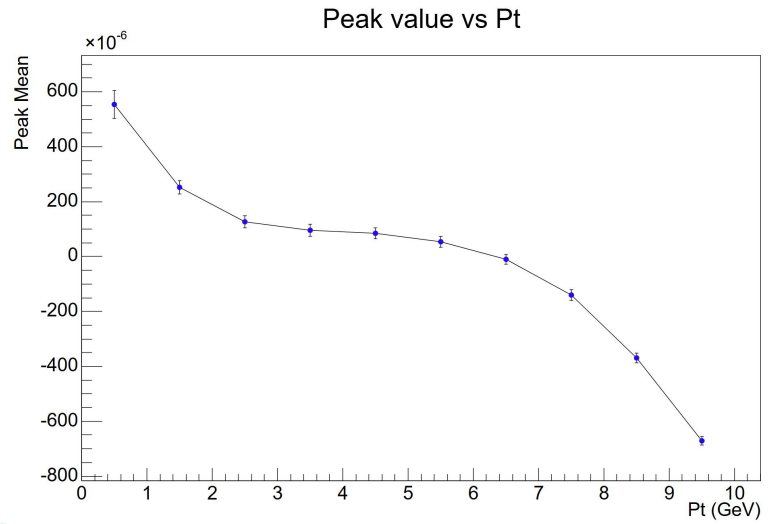
phi bias and resolution - pt

Electron

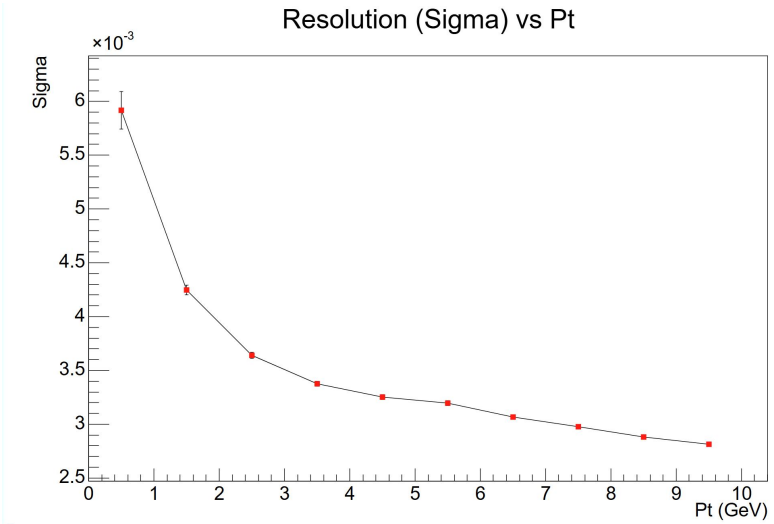
Positron

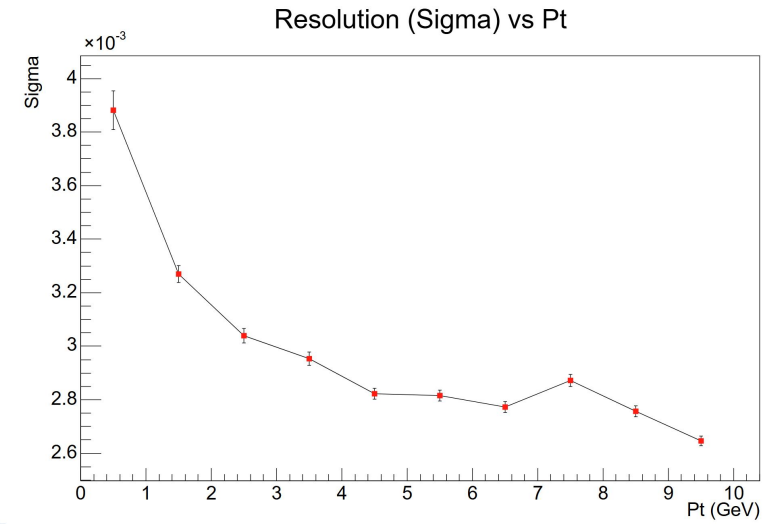
bias





resolution





Pt reconstruction bias and resolution - pt

2-d performance

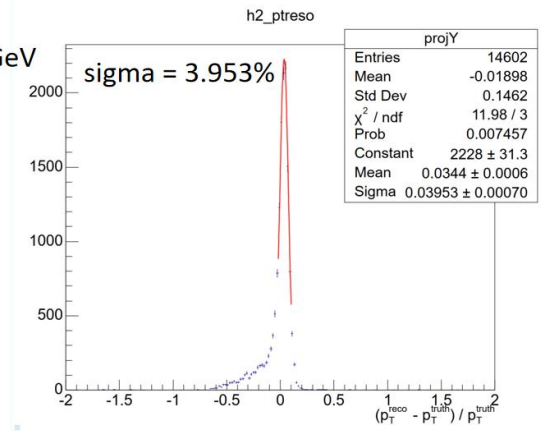
ptFunc2 Performance

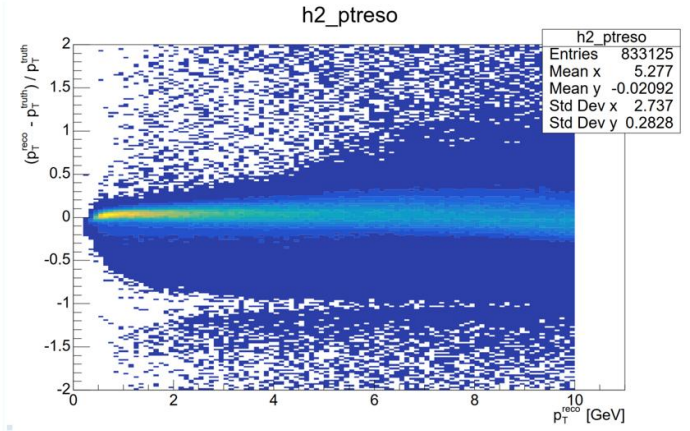
Angle between 2 lines
line connect EMC and oINTT
line connect iINTT oINTT

iINTT oINTT reconstruction position
Cluster reco with tower inner face center after correction

Func from TGraph

0.9-1.1 GeV



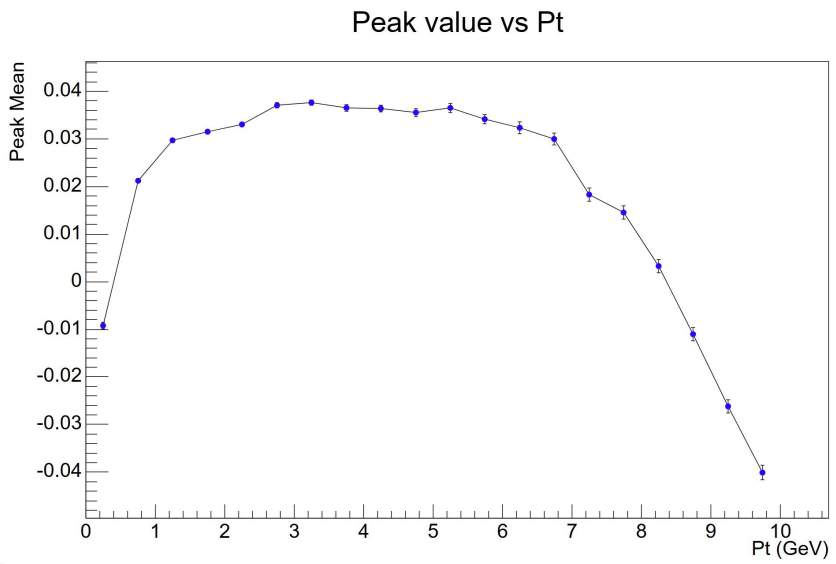


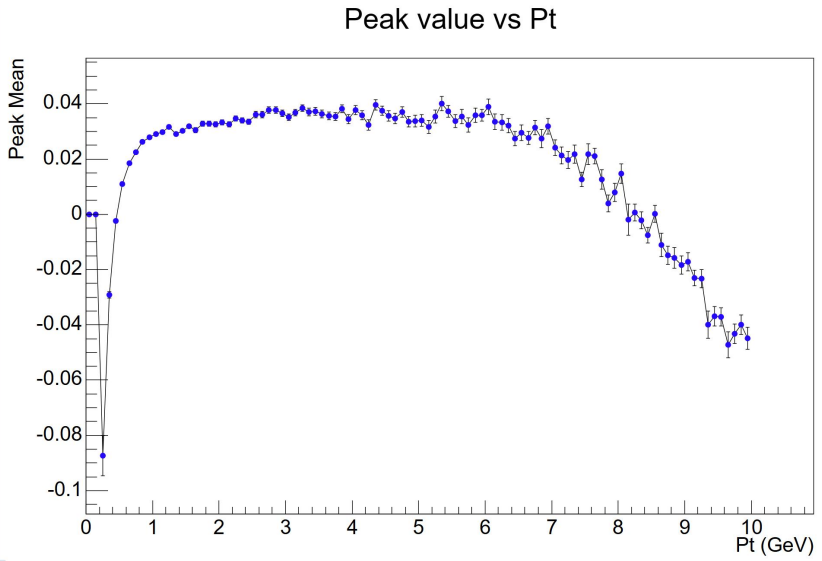
~4% pt resolution on 0.9-1.1GeV

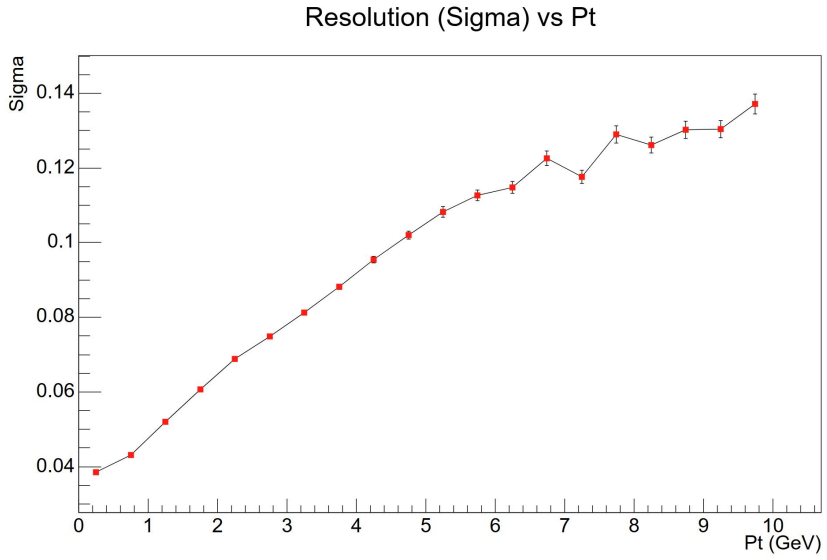
Both two func have similar performance,
pt resolution about 4% on 0.9-1.1 GeV

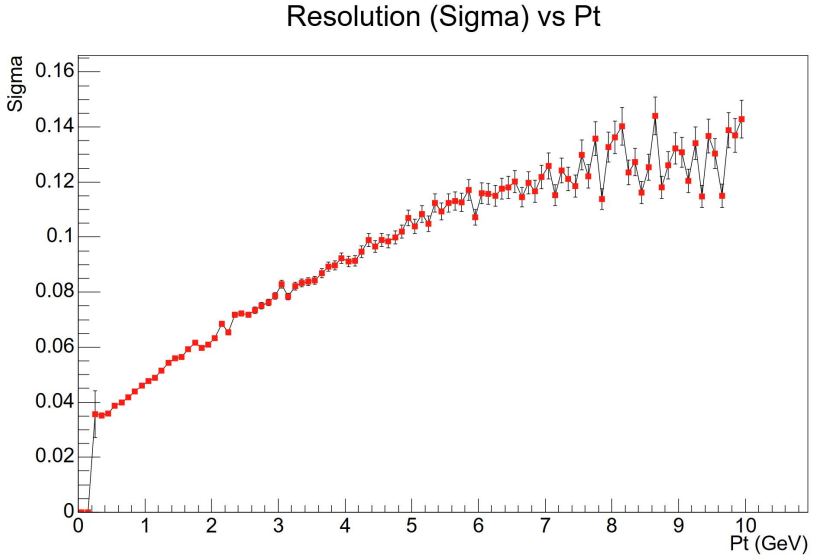
17

electron Pt bias and resolution - pt





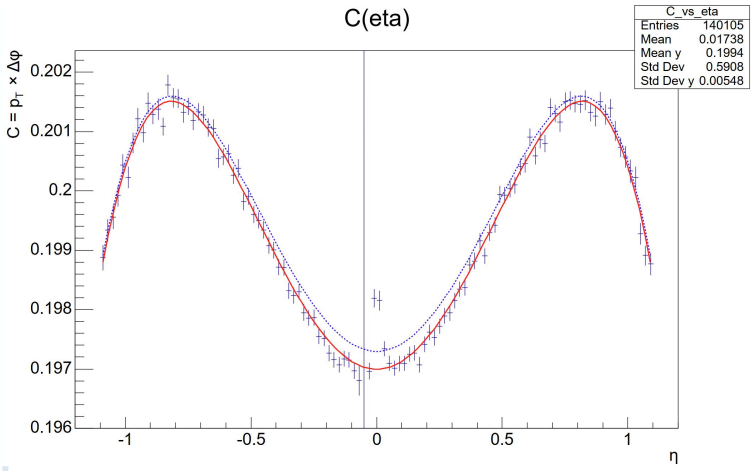


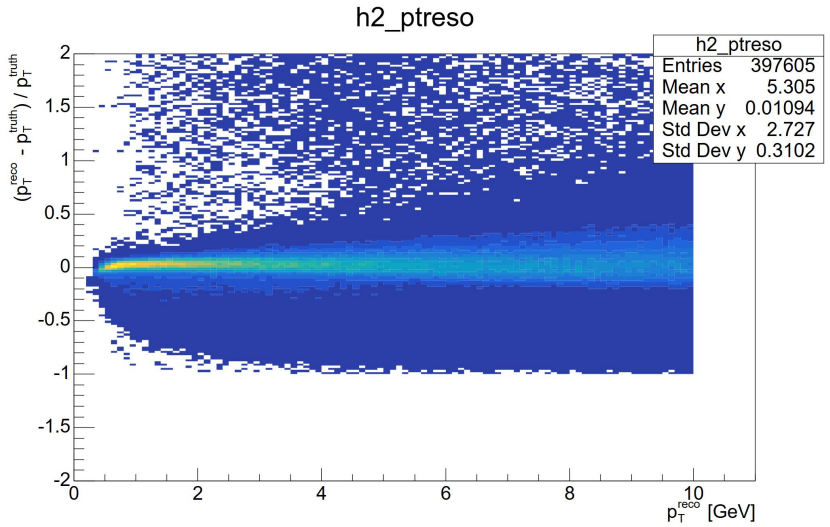


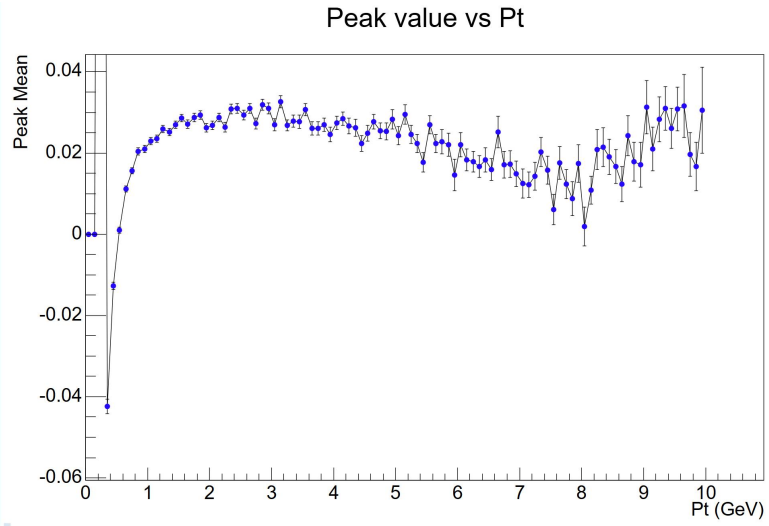
pt resolution increase with larger pt

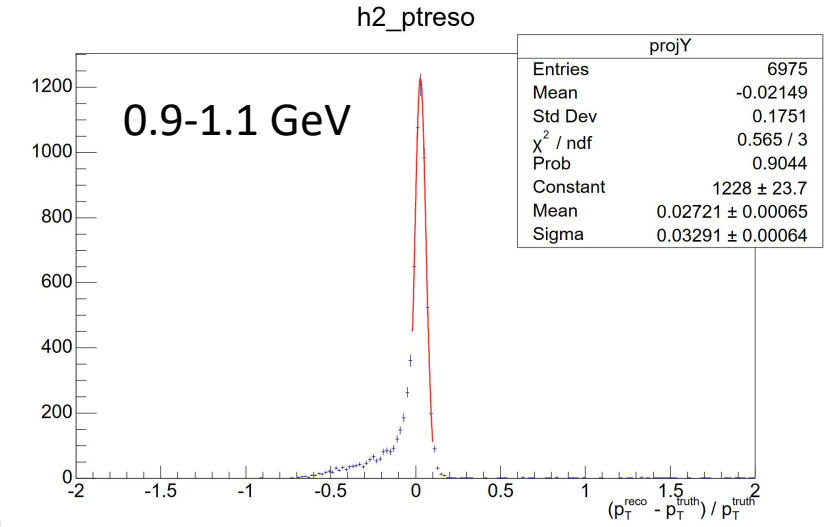
positron Pt bias and resolution - pt

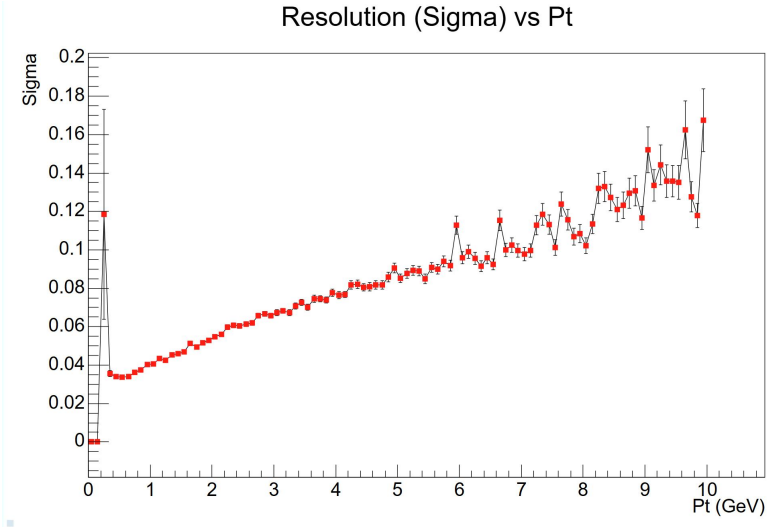
Do it again on positron:
position correction;
Fit to get function C(η) = pT*Δϕ;
Calculate pt = C(η)/Δϕ, plot performance











Thanks