PCA method for Track Trigger

Loriano Storchi, Luisa Alunni Solestizi, Aniello Spiezia, Livio Fanò,
Gian Mario Bilei. **Atanu Modak**

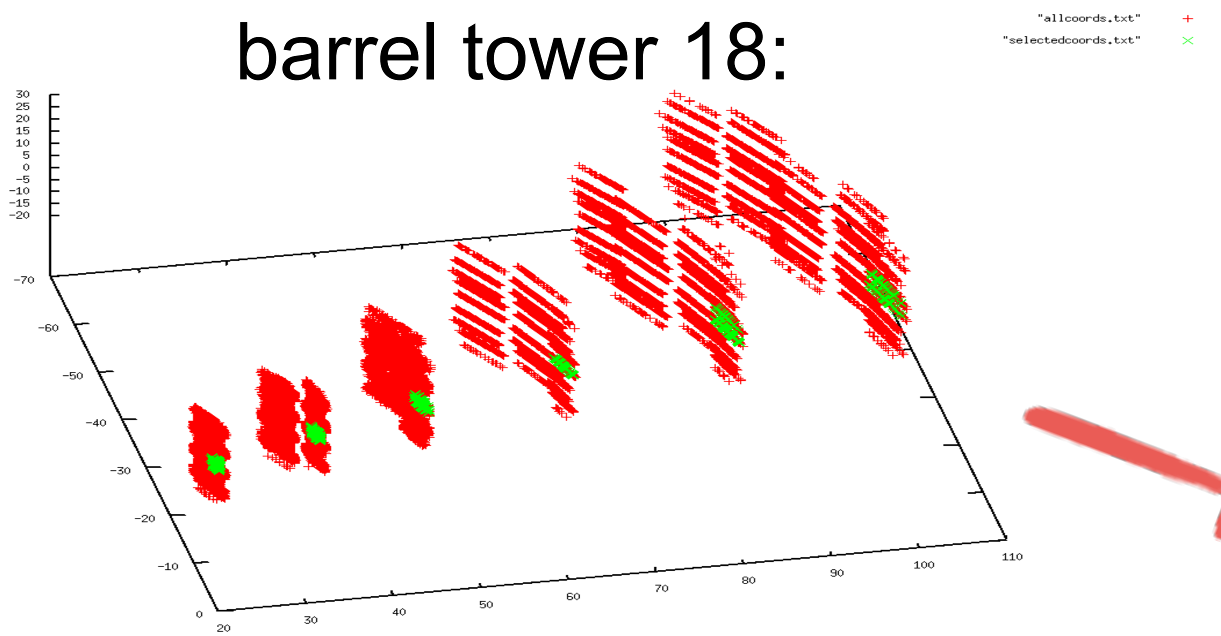
Perugia
INFN
Istituto Nazionale di Fisica Nucleare

Department of Physics
Dipartimento di Fisica

CMS
Compact Muon Solenoid

PCA problem

The red dots are the stubs in the barrel tower 18:



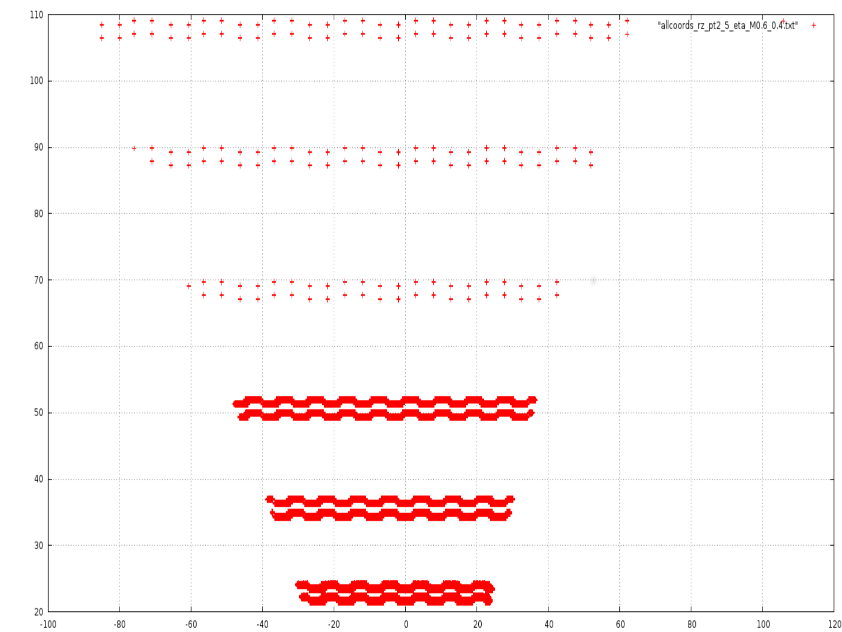
- The problem is divided in two sub-problems:

  plane r-z → from (r_i, z_i), where i is the referred to the stubs, we find the two parameters of the track: **z_0 and η ($\cot(\theta)$);**

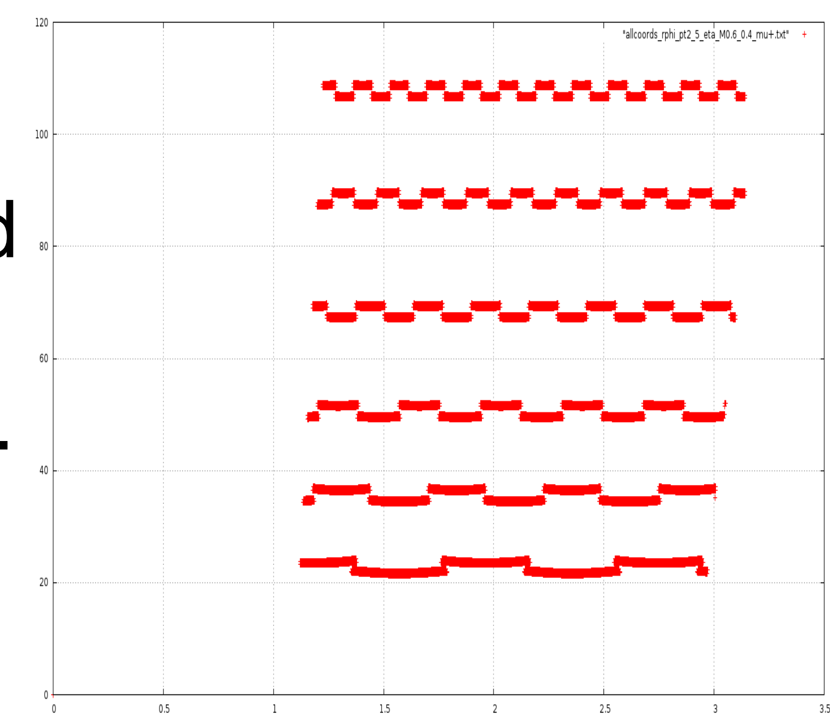
  plane r-ϕ → from (r_i, ϕ_i), where i is the referred to the stubs, we find the two parameters of the track: **c/p_T and ϕ**, where c is the charge.

- For each plane, there are 12 coordinates = 2 coordinates x 6 (3) layers → **12 (6) stub coordinates.**

r-ϕ plane:

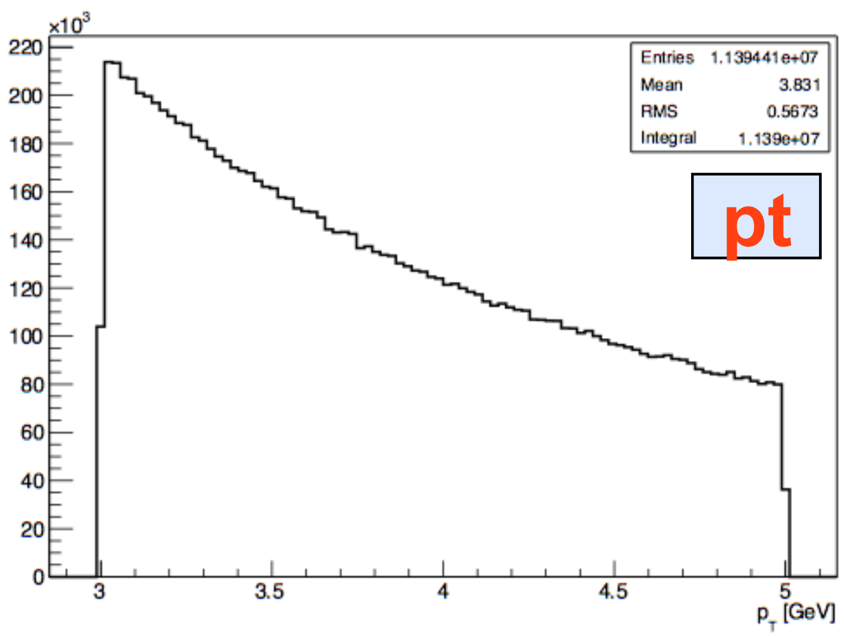


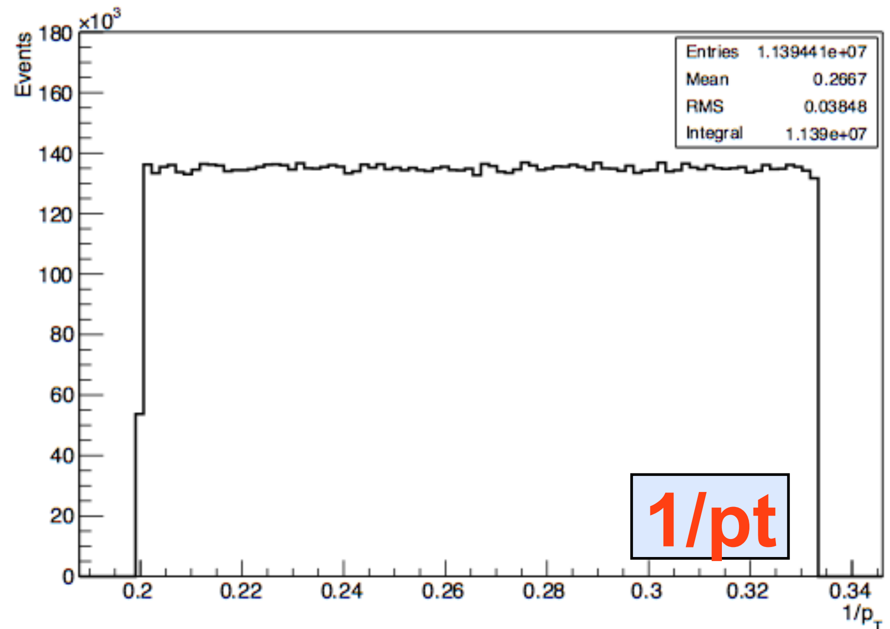
r-z plane:

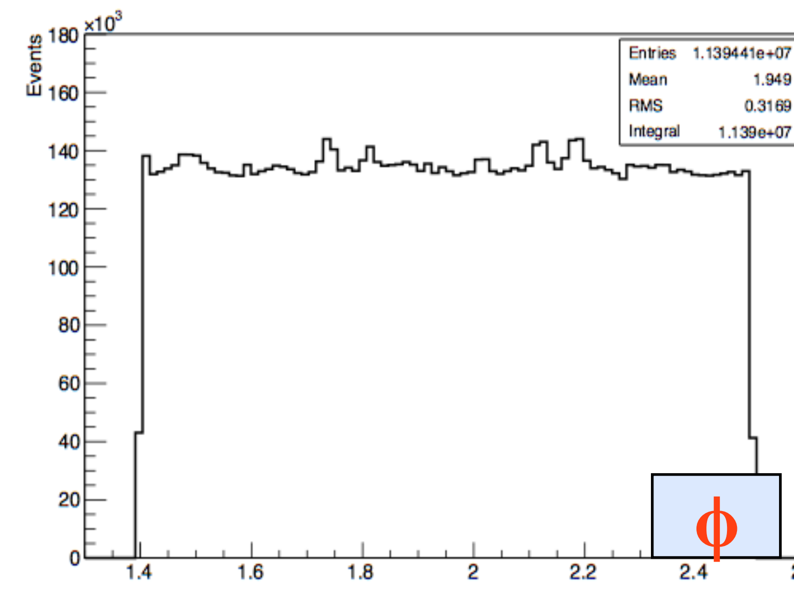


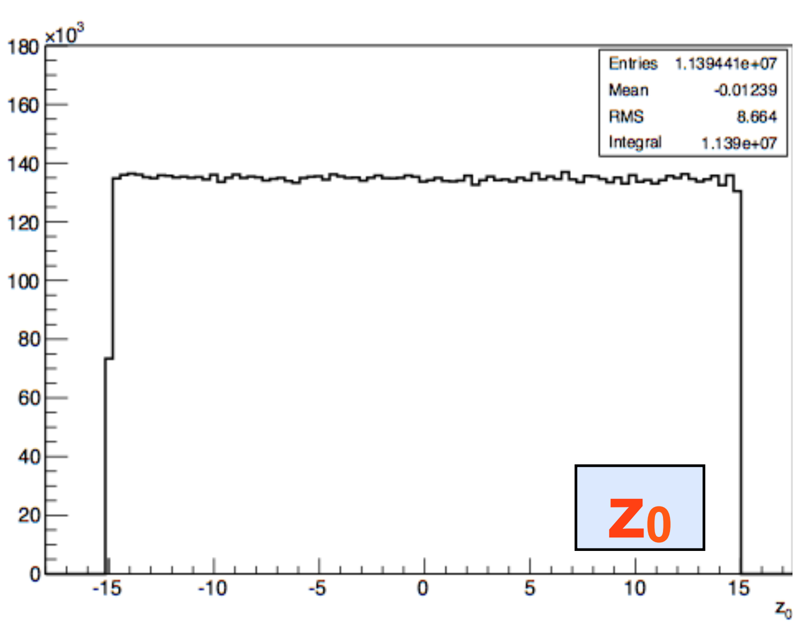
Generation

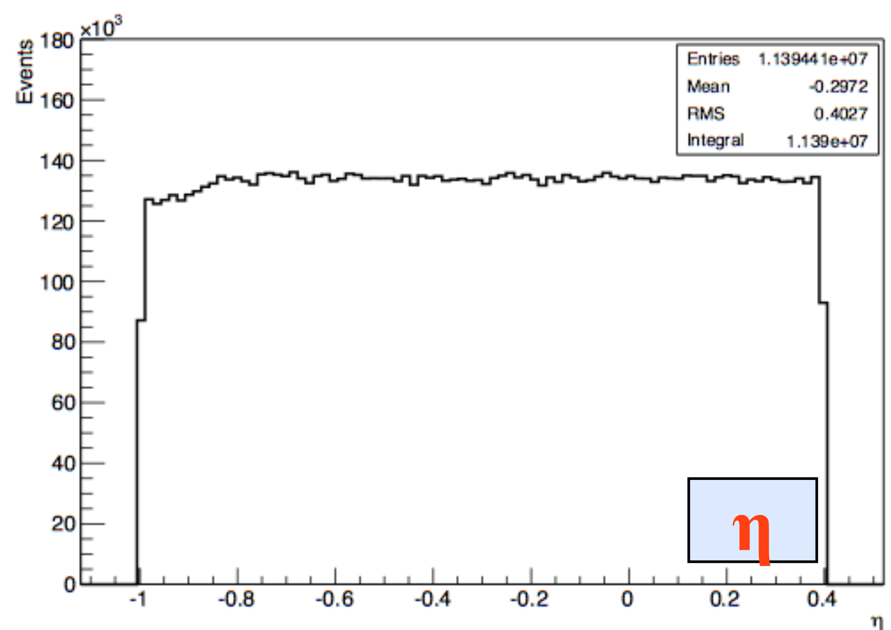
Generated muons samples with these track-parameters distributions:

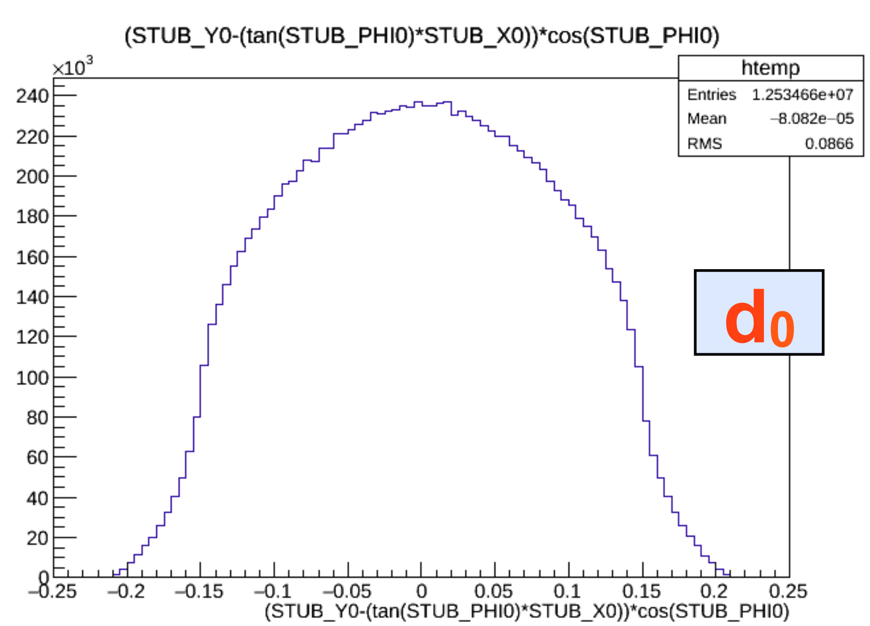












Results Barrel

Generation

- The generation has been performed using the following framework: http://sviret.web.cern.ch/sviret/Welcome.php?n=CMS.HLLHCTuto620
- The **code for the PCA method that we are developing** can be found here: https://bitbucket.org/lstorchi/gf_fit
- The events are generated in the **tower 18**:
  - η is in the range **(-0.6, 0.4)**
  - ϕ is the range **(1.1, 2.9)**

PLANE rz

- 1 datasets have been generated:
  - ~10M of events per sample
  - **pt** → {2, 200} GeV

PLANE rphi

- 9 datasets have been generated in different ranges of pt:
  - ~5M of events per sample
  - **pt bins** → {2, 5, 10, 15, 20, 30, 40, 50, 100, 200} GeV

Results Barrel Plane rz

Plane rz

PLANE rz

- 1 **datasets** have been generated in different ranges of pt:
  - ~10M of events per sample
  - pt ➙ {2, 200} GeV

In the plane rz, the idea is to have:

- **1 bin in pt**: [2, 200] GeV
- **20 bins in eta**: bins of **0.05** (between -0.6 and 0.4 for this tower)

  OR **10 bins in eta**: bins of **0.1** (between -0.6 and 0.4 for this tower)

  OR **5 bins in eta**: bins of **0.2** (between -0.6 and 0.4 for this tower)
- **1 bin in phi**: [1.1, 2.9] for this tower
- the **s-modules are excluded by the computation of the constants**, so that only the first three layers are considered

At the end, we have 20 or 10 or 5 sets of constants for each tower

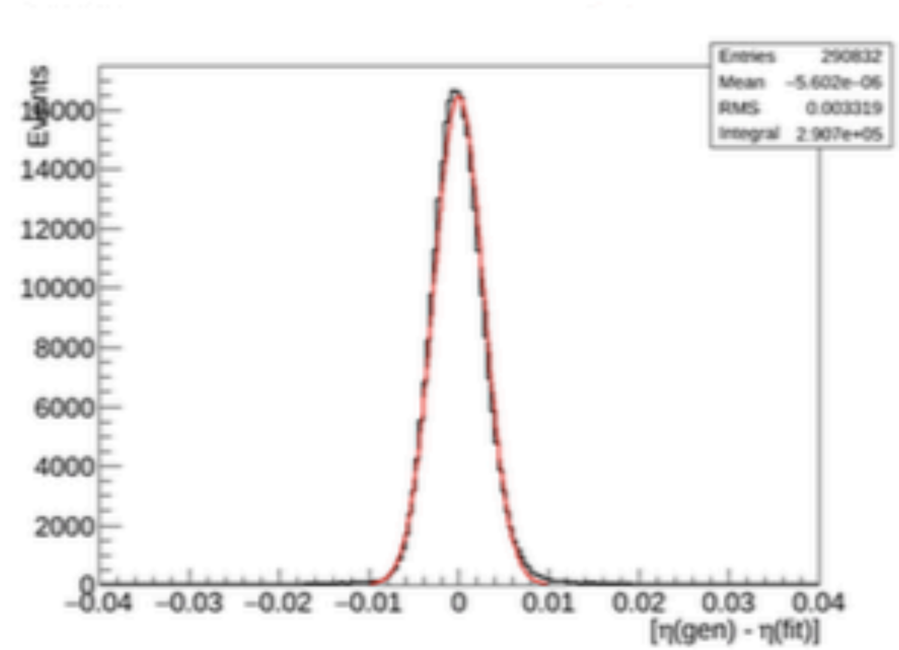
Plane rz – η results

- We plot the following variable:

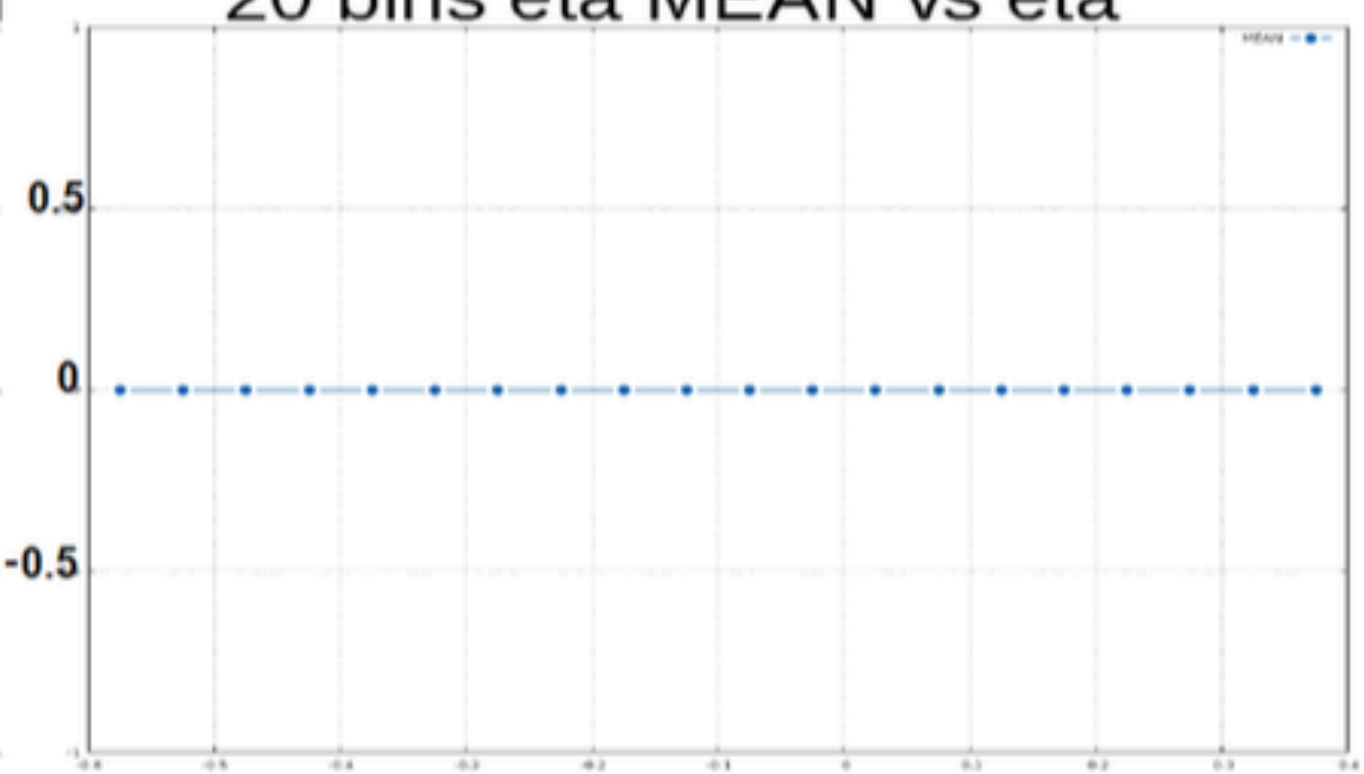
  [η(generated) - η(fit)]

- We plot the **mean** and the **RMS** as a function of η,
- dividing in **20 bins** of **0.05** in η.

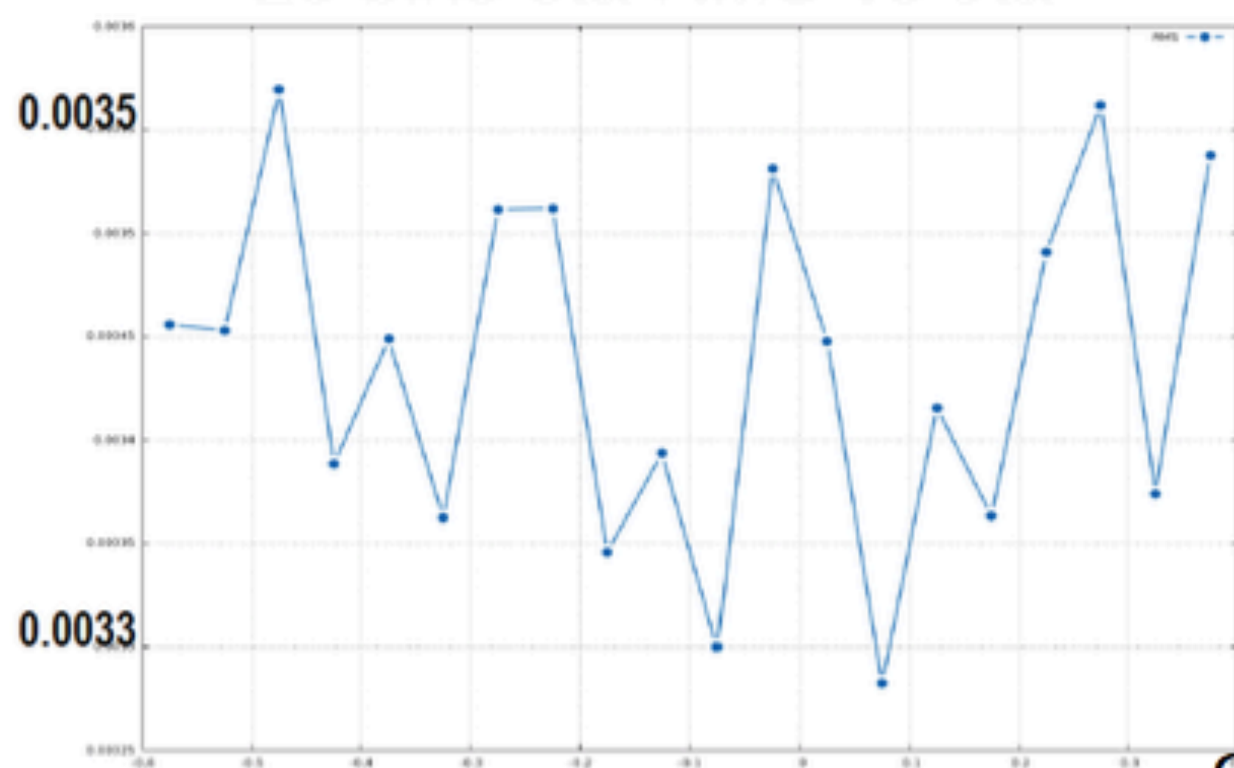
First η bin



20 bins eta MEAN vs eta



20 bins eta RMS vs eta



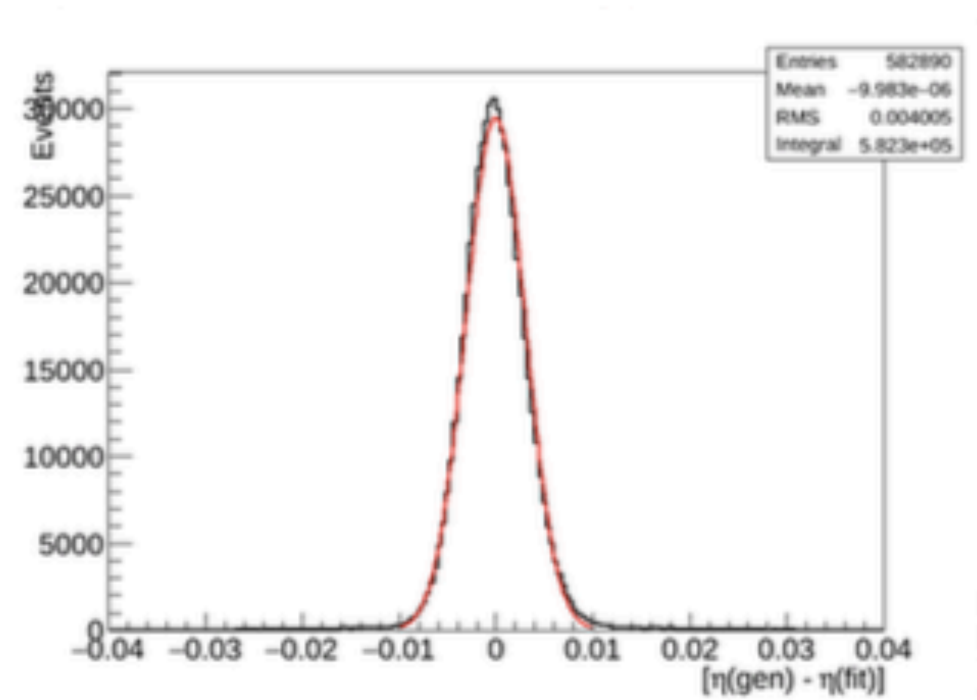
Plane rz – η results

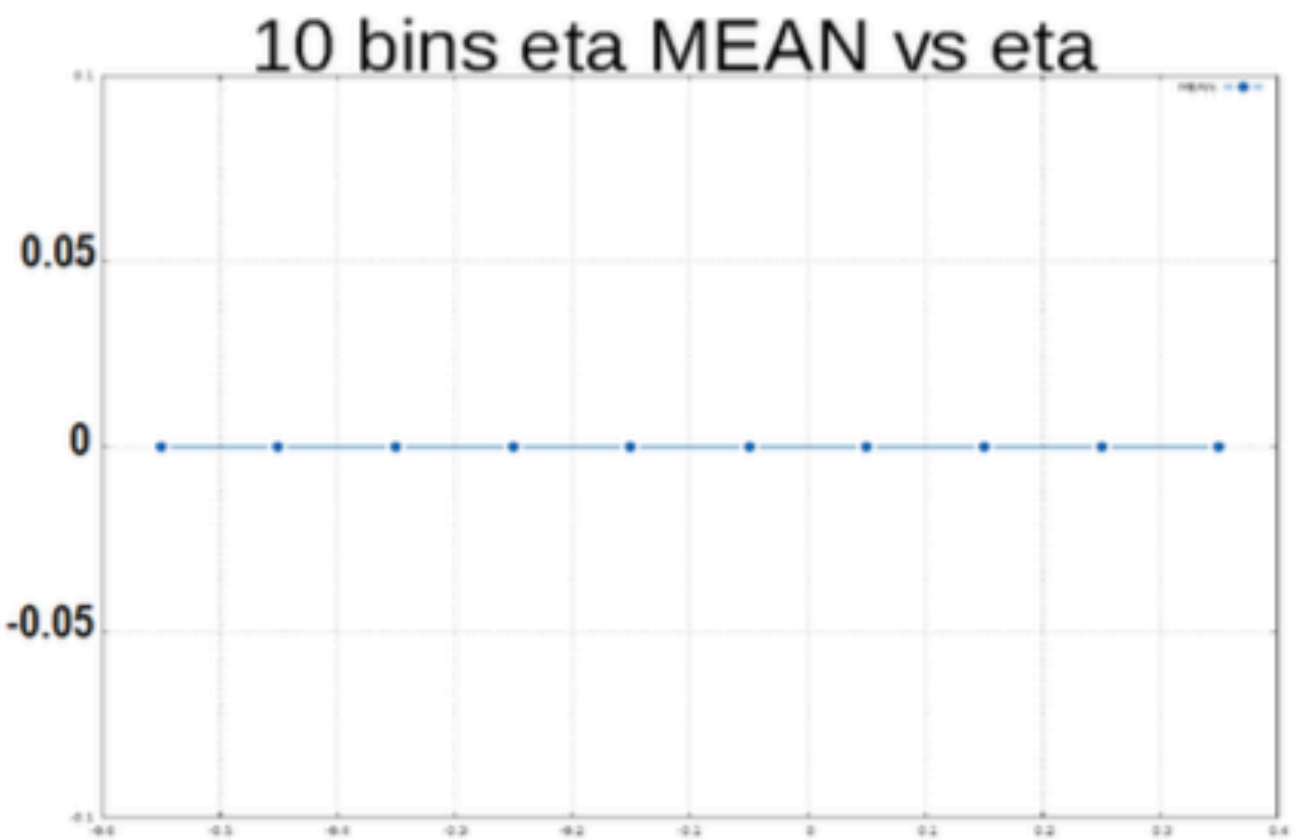
- We plot the following variable:

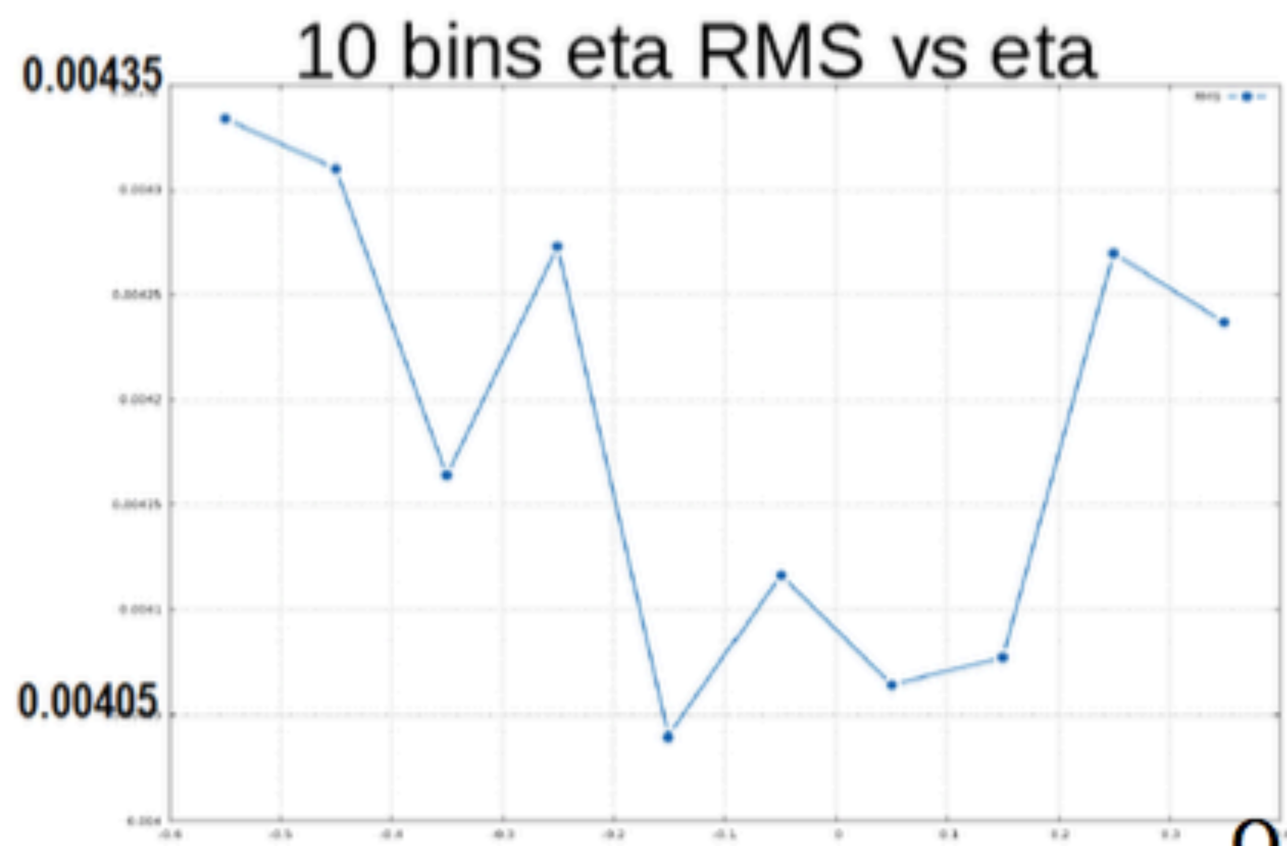
  [η(generated) - η(fit)]

- We plot the **mean** and the **RMS** as a function of η,
- dividing in **10 bins** of **0.1** in η.

First η bin







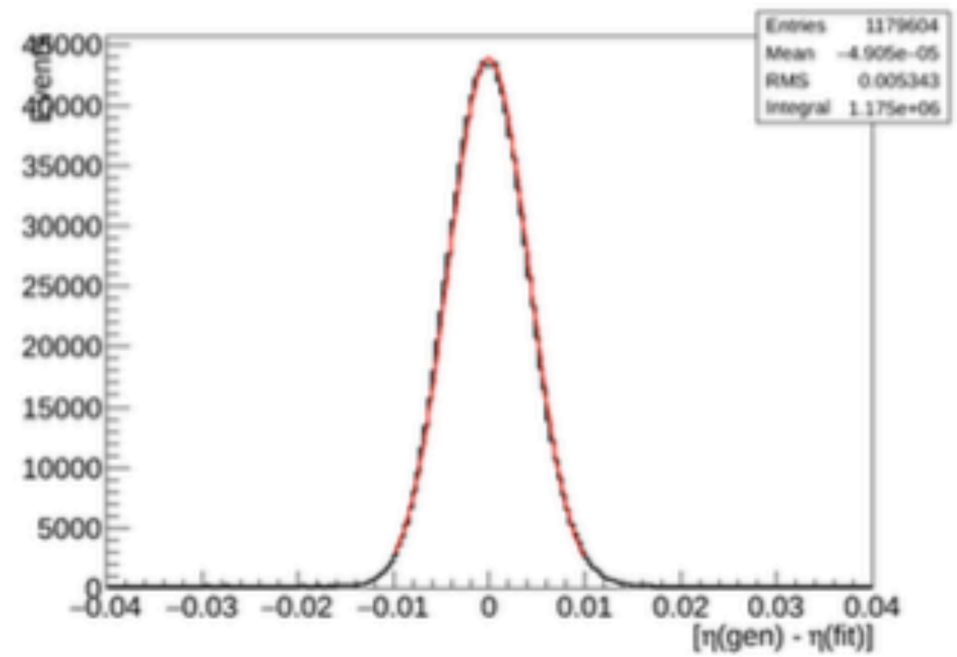
Plane rz – η results

- We plot the following variable:

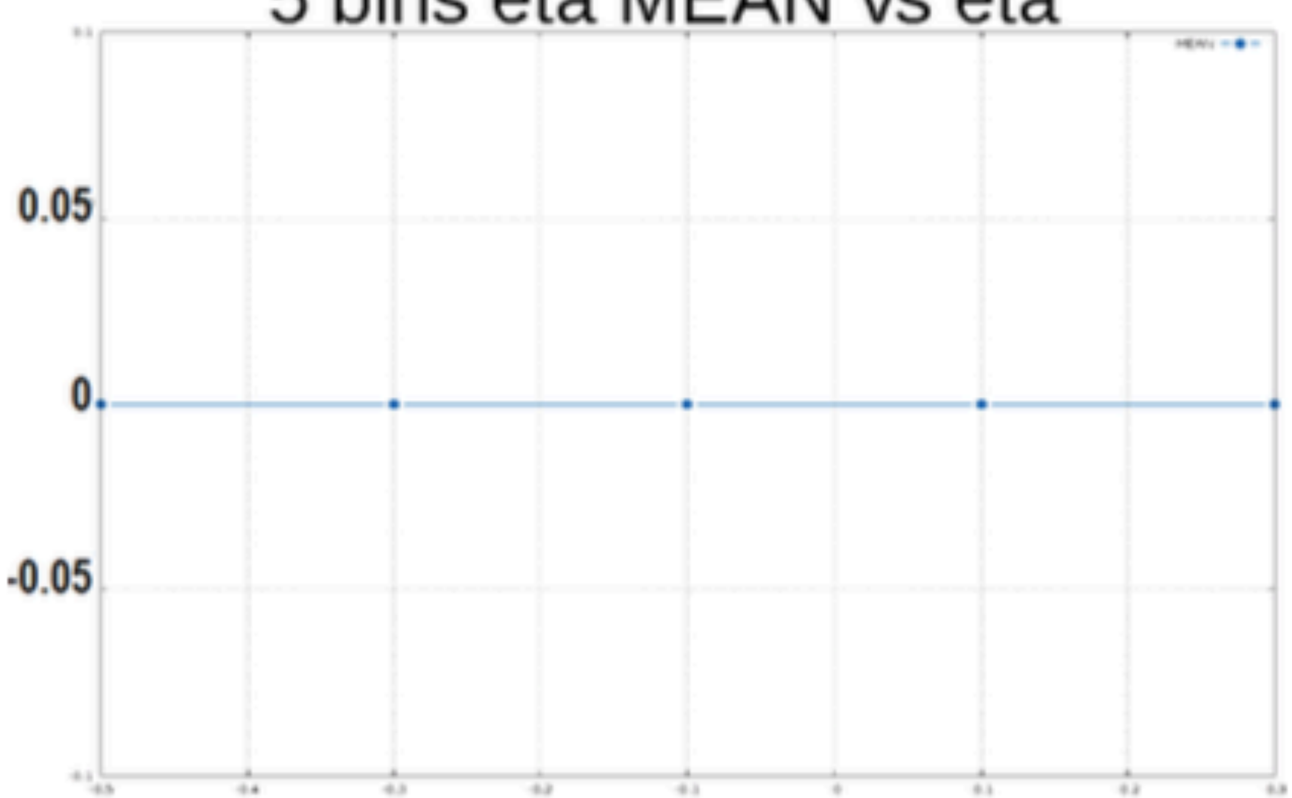
  [η(generated) - η(fit)]

- We plot the **mean** and the **RMS** as a function of η,
- dividing in **5 bins** of **0.2** in η.

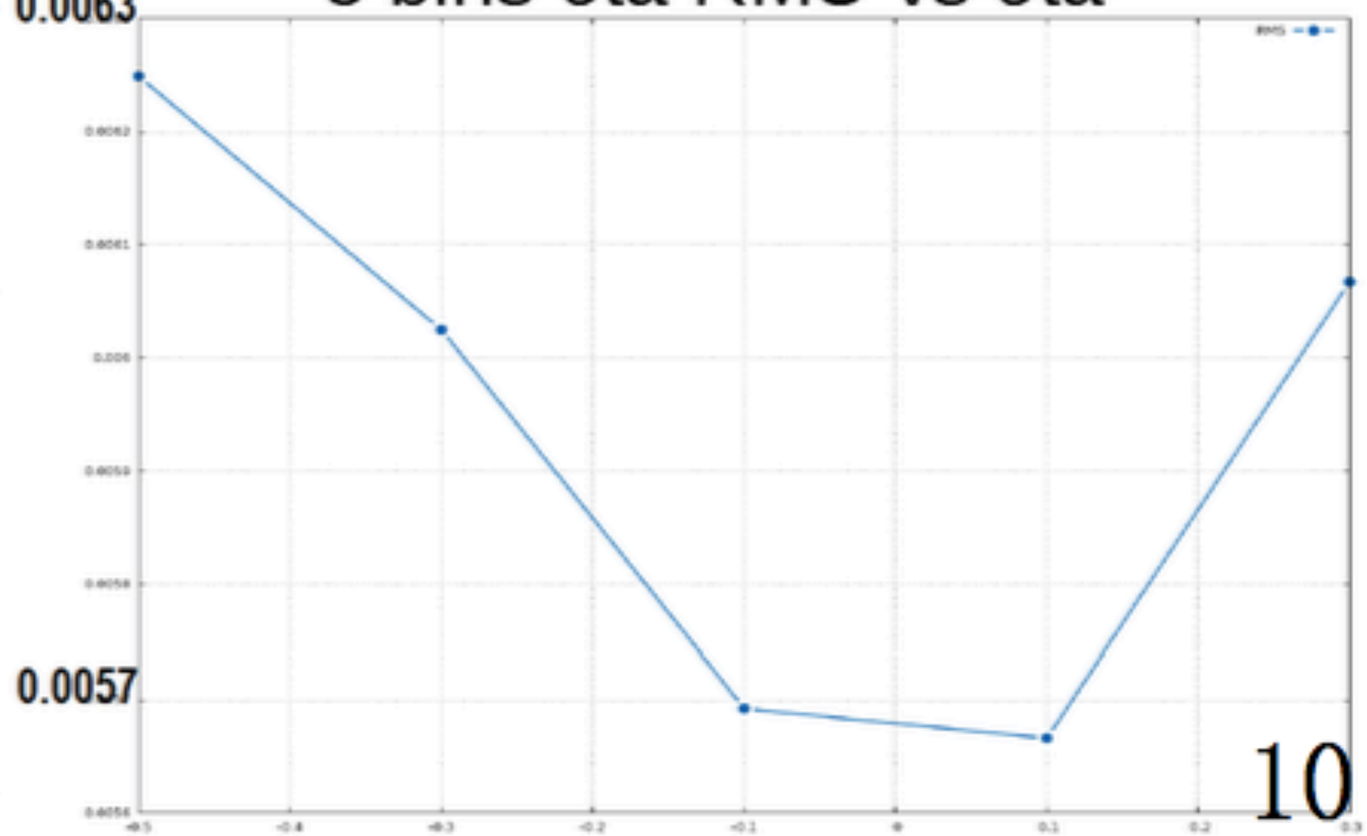
First η bin



5 bins eta MEAN vs eta

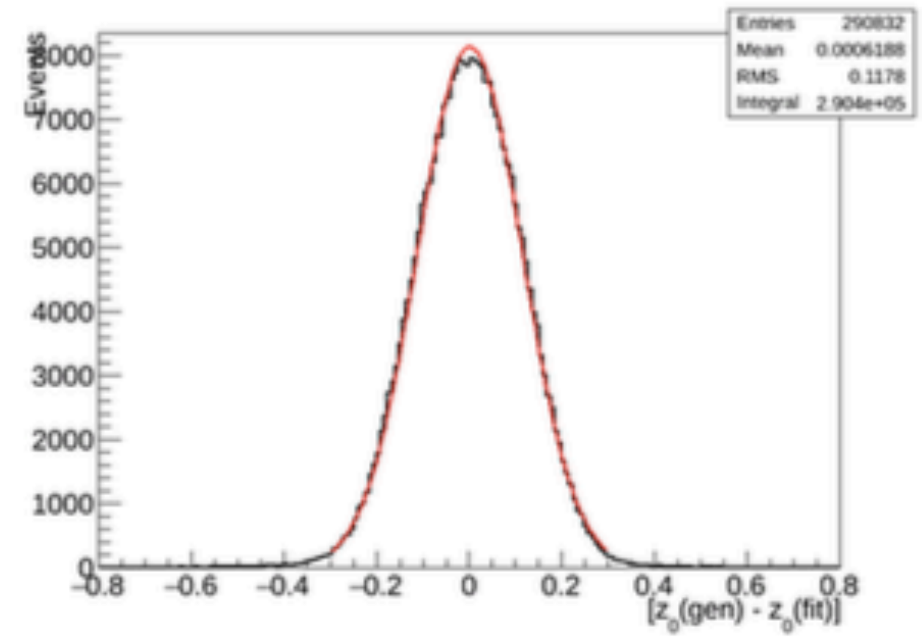


5 bins eta RMS vs eta



Plane rz – z_0 results

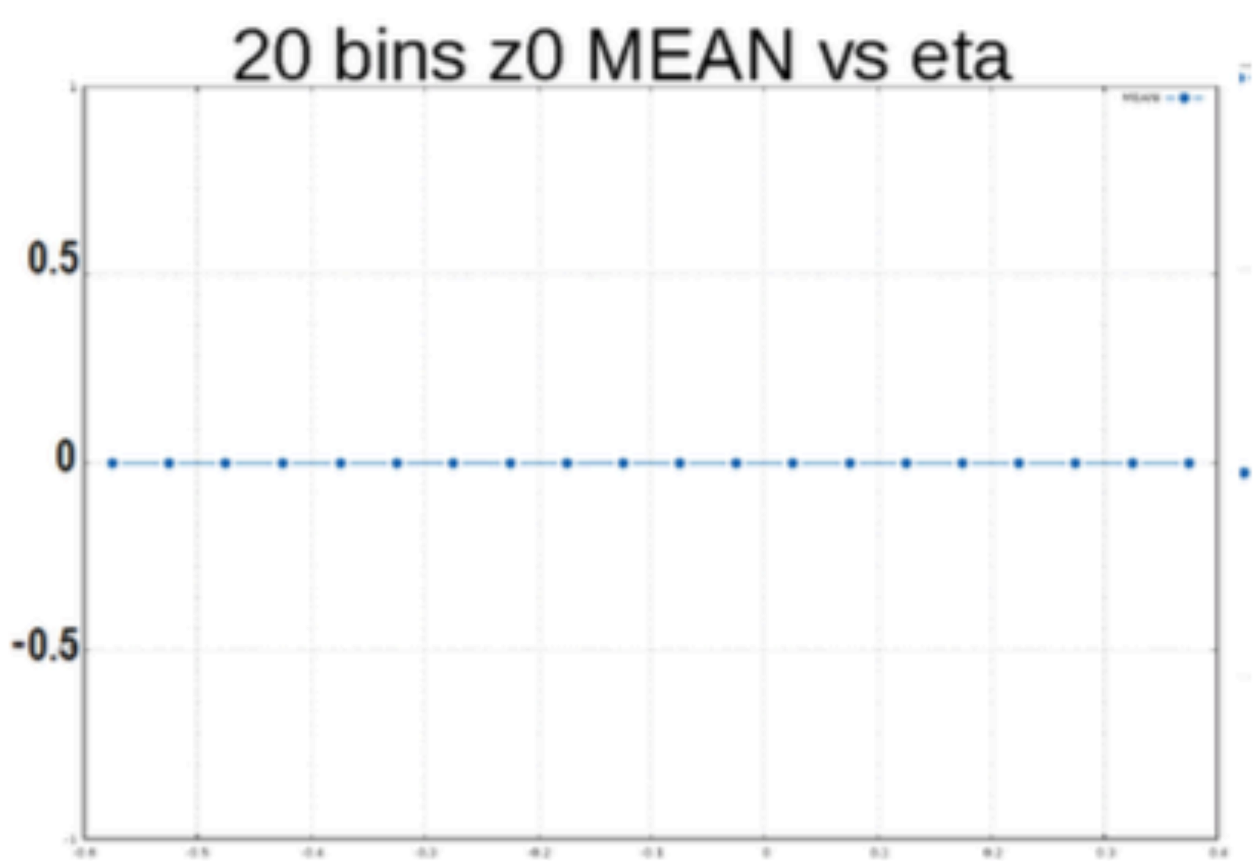
First η bin

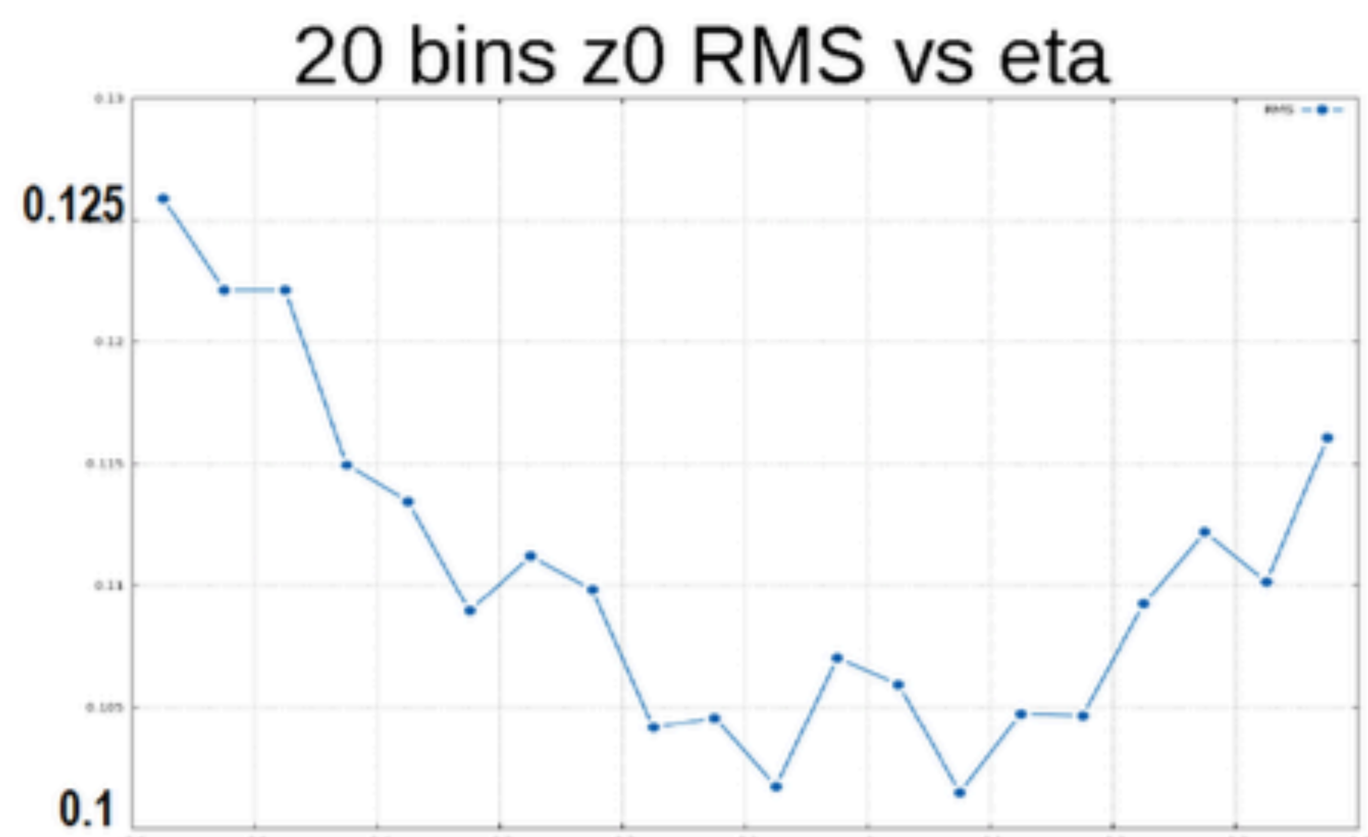


- We plot the following variable:

$$[z_0(\text{generated}) - z_0(\text{fit})]$$

- We plot the **mean** and the **RMS** as a function of η,
- dividing in **20 bins** of **0.05** in η.





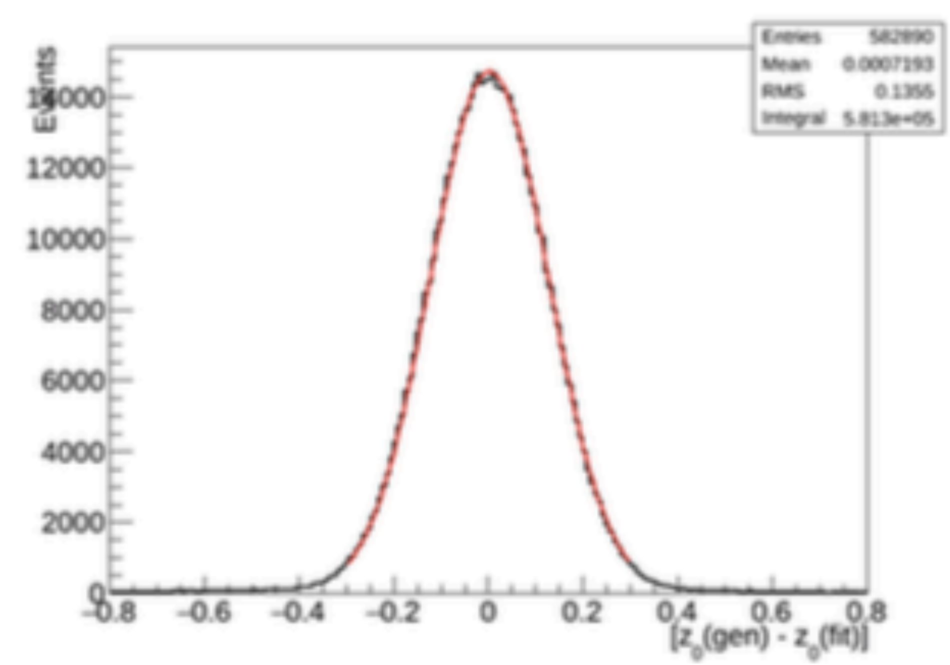
Plane rz – z_0 results

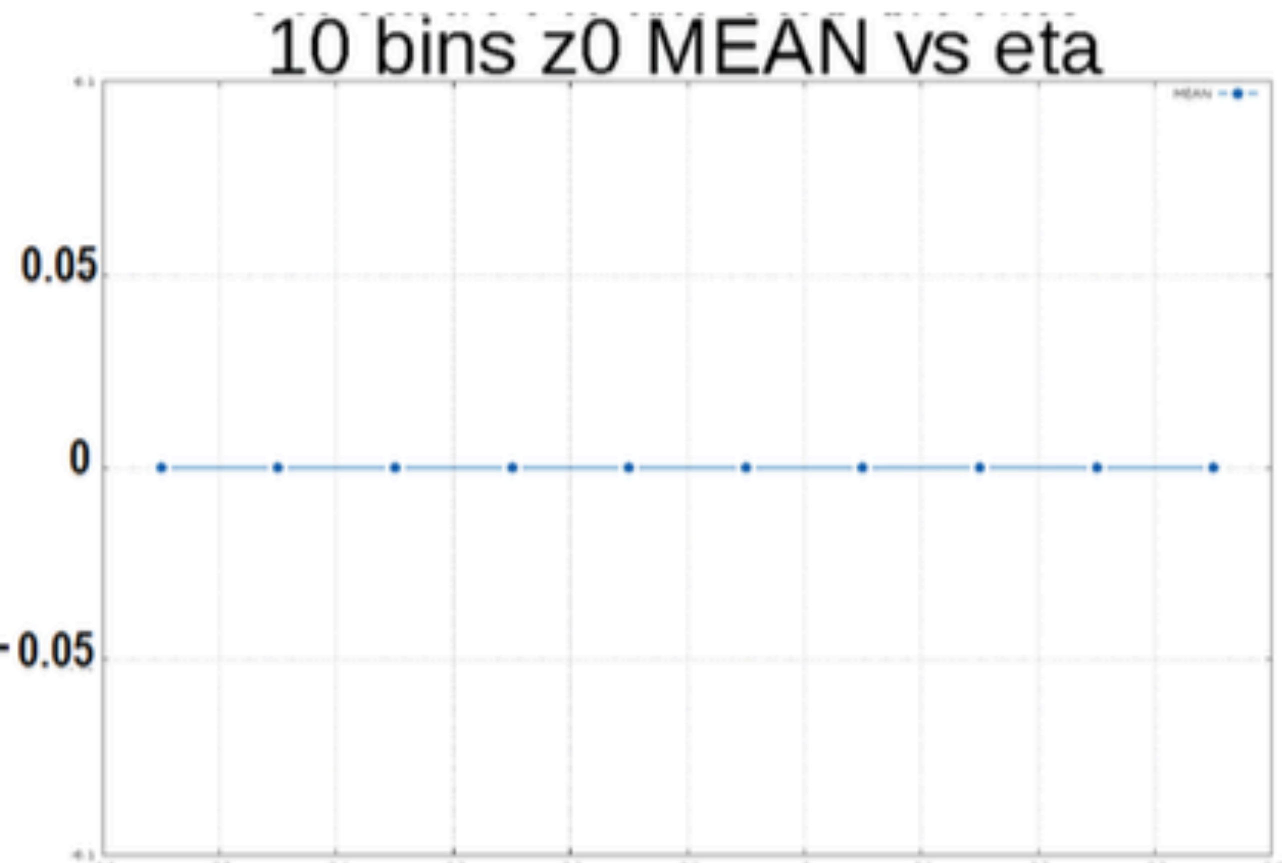
- We plot the following variable:

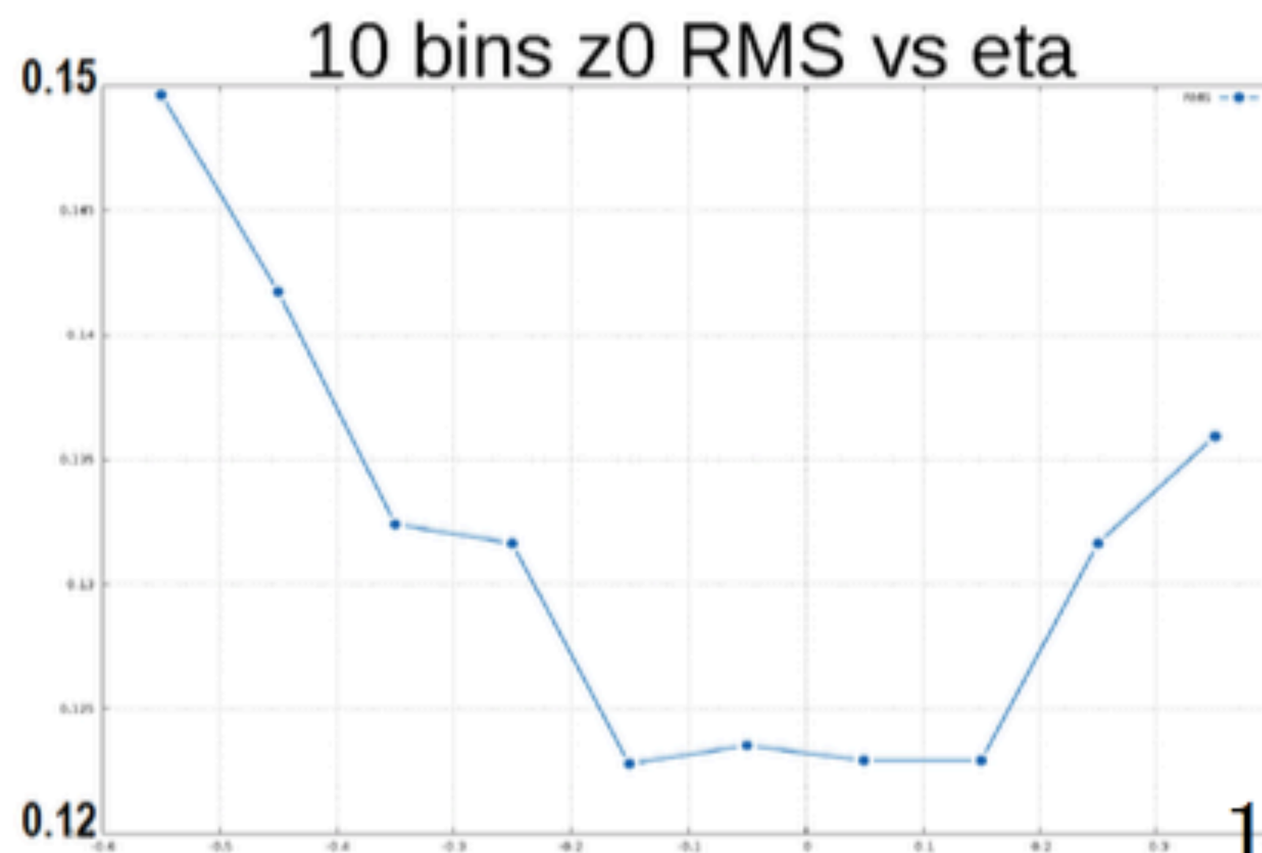
  [z_0(generated) - z_0(fit)]

- We plot the **mean** and the **RMS** as a function of η,
- dividing in **10 bins** of **0.1** in η.

First η bin







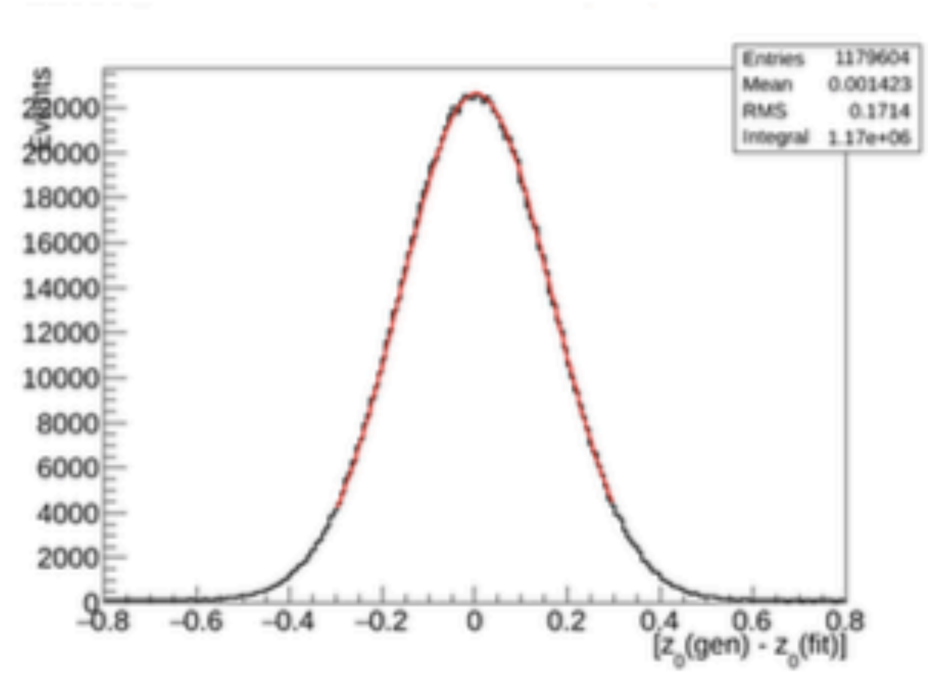
Plane rz – z_0 results

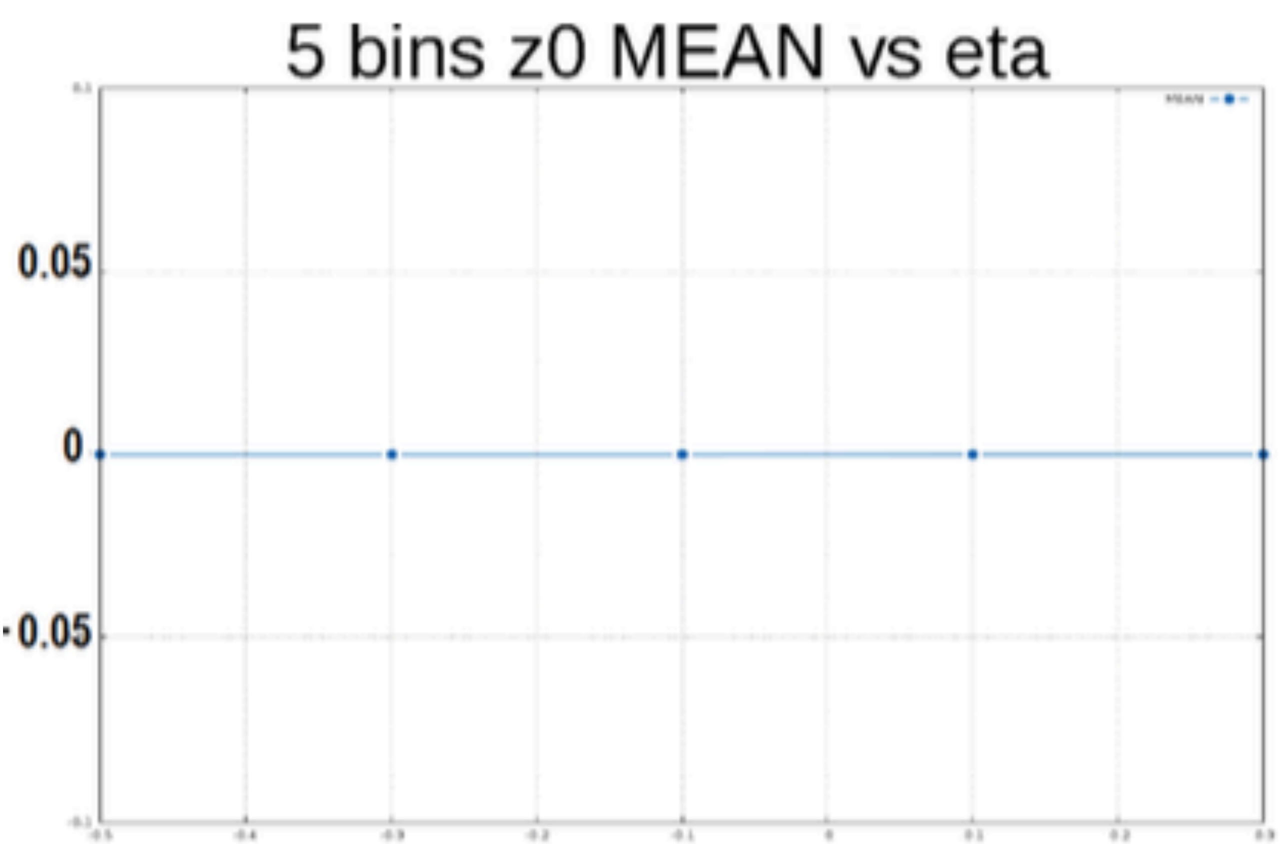
First η bin

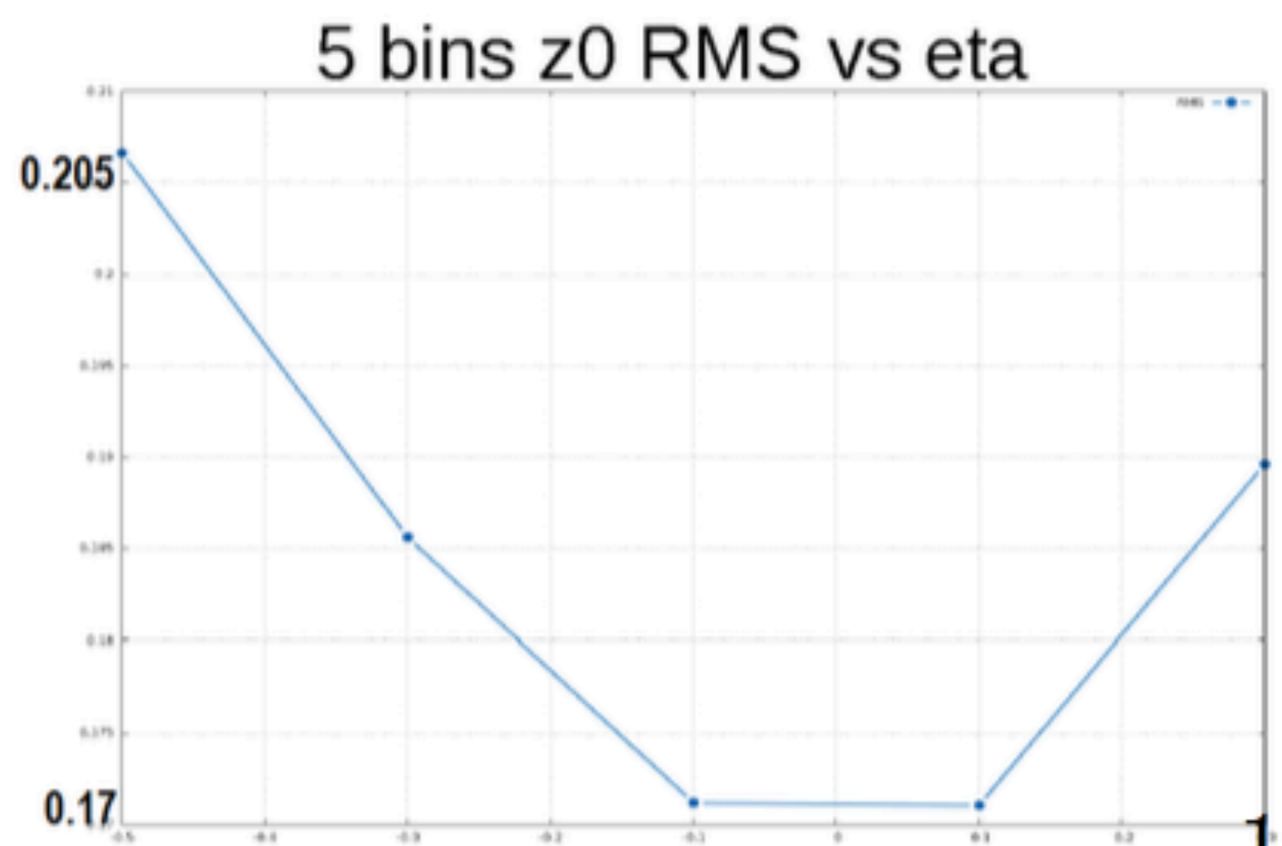
- We plot the following variable:

  [z_0(generated) - z_0(fit)]

- We plot the **mean** and the **RMS** as a function of η,
- dividing in **5 bins** of **0.2** in η.







Results Barrel Plane rϕ

Plane rϕ

PLANE rϕ

- 9 datasets have been generated in different ranges of pt:
  - ~5M of events per sample
  - pt bins → {2, 5, 10, 15, 20, 30, 40, 50, 100, 200} GeV

In the plane rϕ, the idea is to have:

- **9 bins in pt**: {2, 5, 10, 15, 20, 30, 40, 50, 100, 200} GeV

  OR 7 bins in pt: {2, 7, 12, 18, 25, 50, 100, 200} GeV

  OR 6 bins in pt: {2, 8, 15, 30, 50, 100, 200} GeV
- **one bin in eta**: [-0.6, 0.2] for this tower (warning for eta 0.2;04!)
- **one bin in phi**: [1.1, 2.9] for this tower
- the computation of the constants and the fit is done **separately for positive and negative muons**

At the end, we have 18 or 14 or 12 sets of constants for each tower

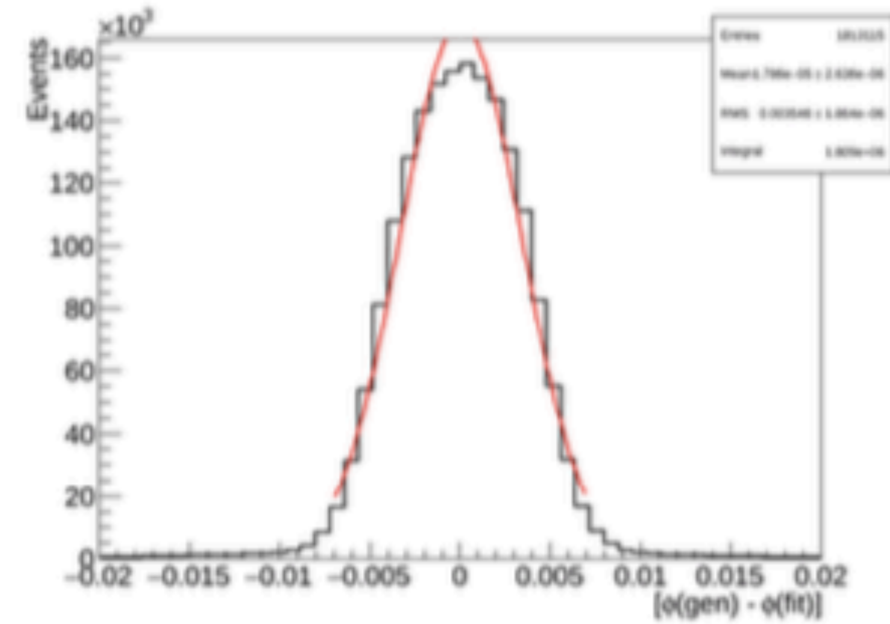
Plane rϕ – ϕ results

- We plot the following variable:

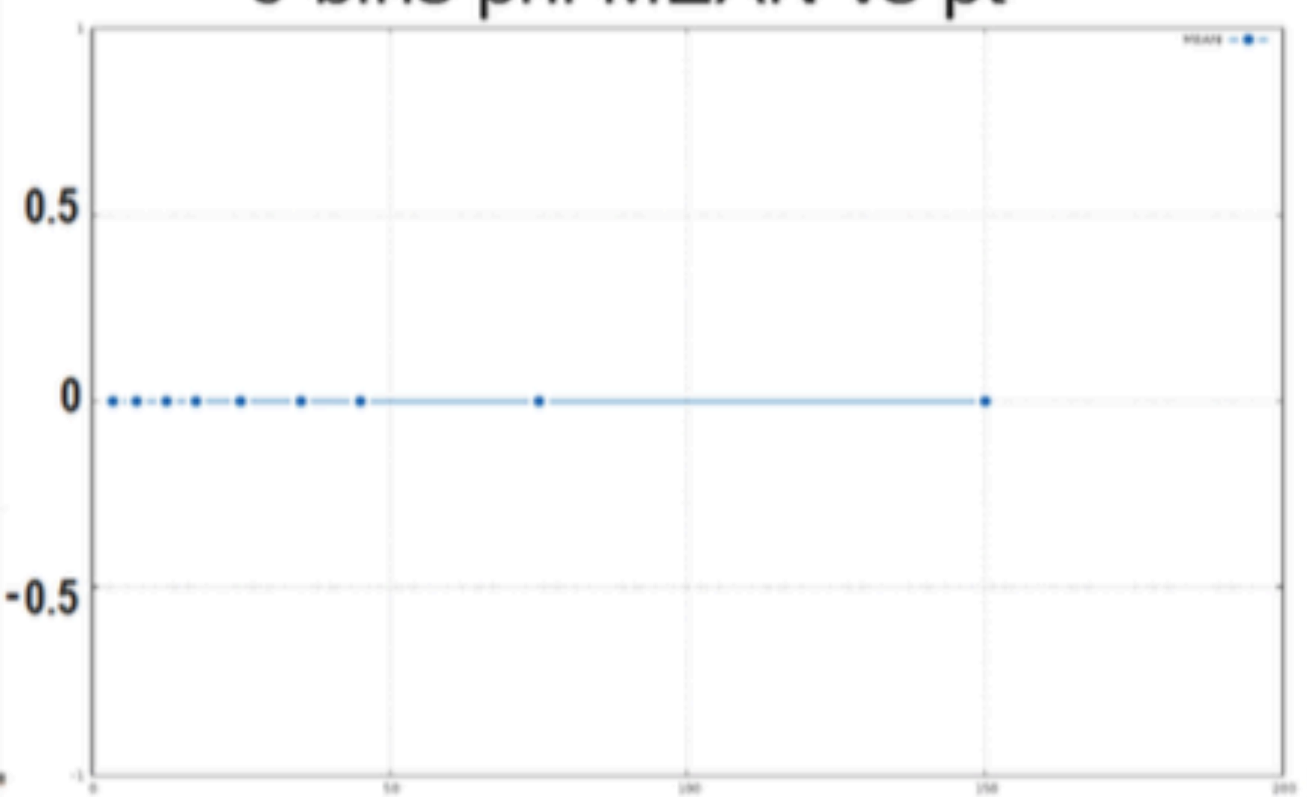
  [ϕ(generated) - ϕ(fit)]

- We have plotted the **mean** and the **RMS** as a function of pT,
- for **9 bins** in pT.

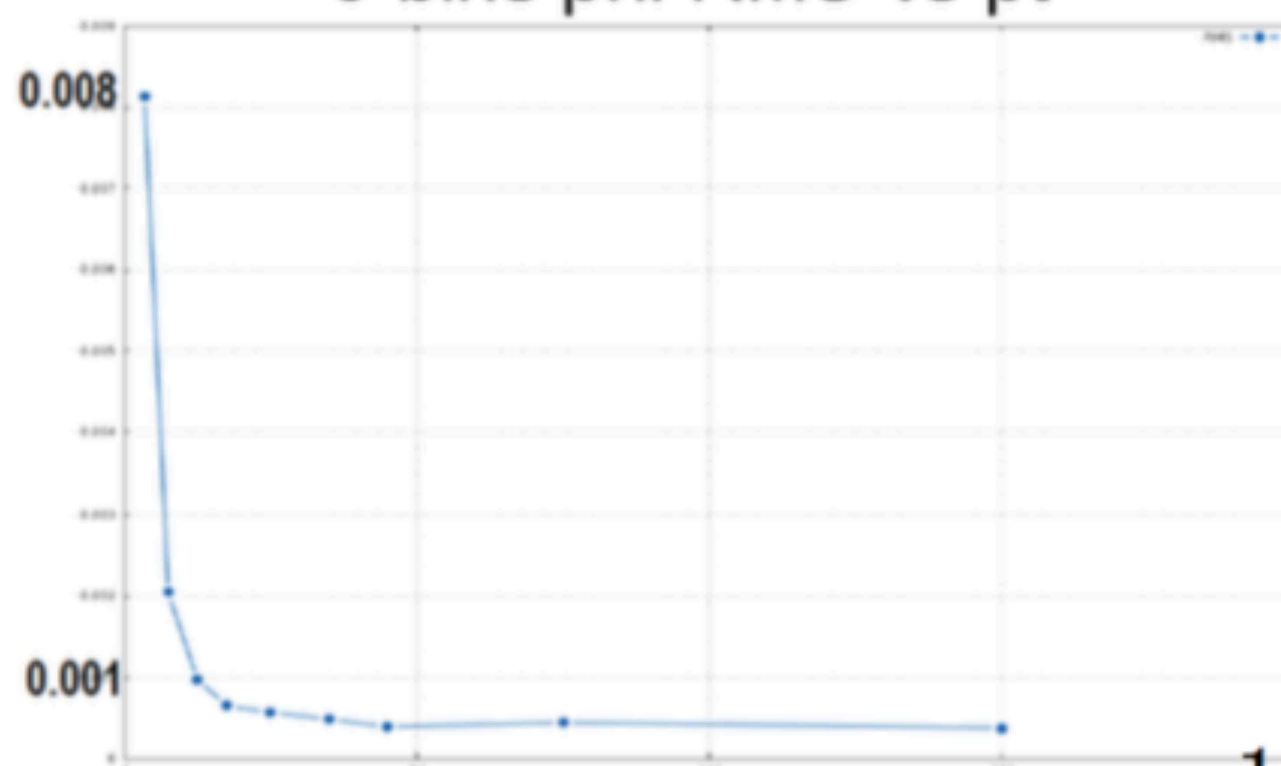
First pT bin



9 bins phi MEAN vs pt



9 bins phi RMS vs pt



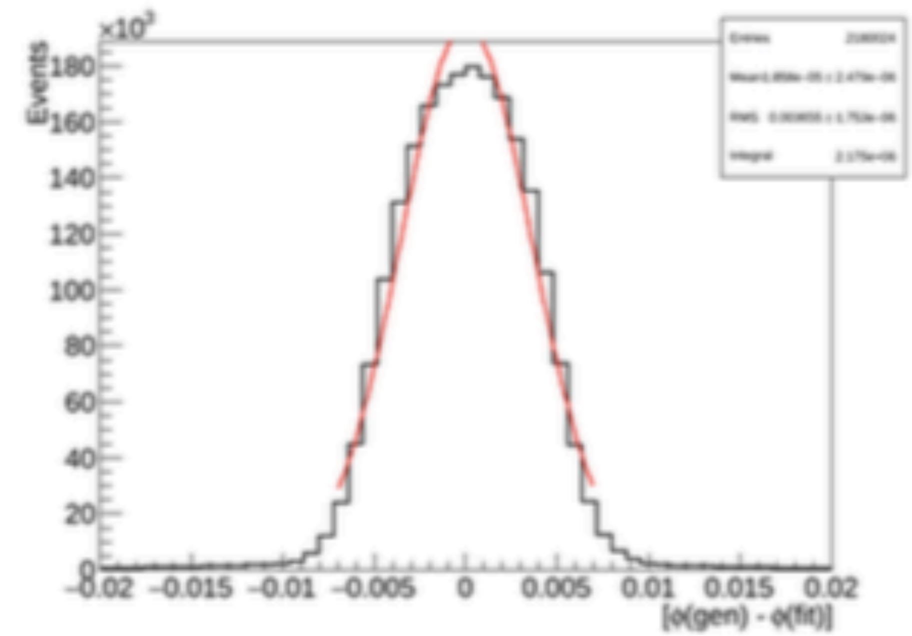
Plane rϕ − ϕ results

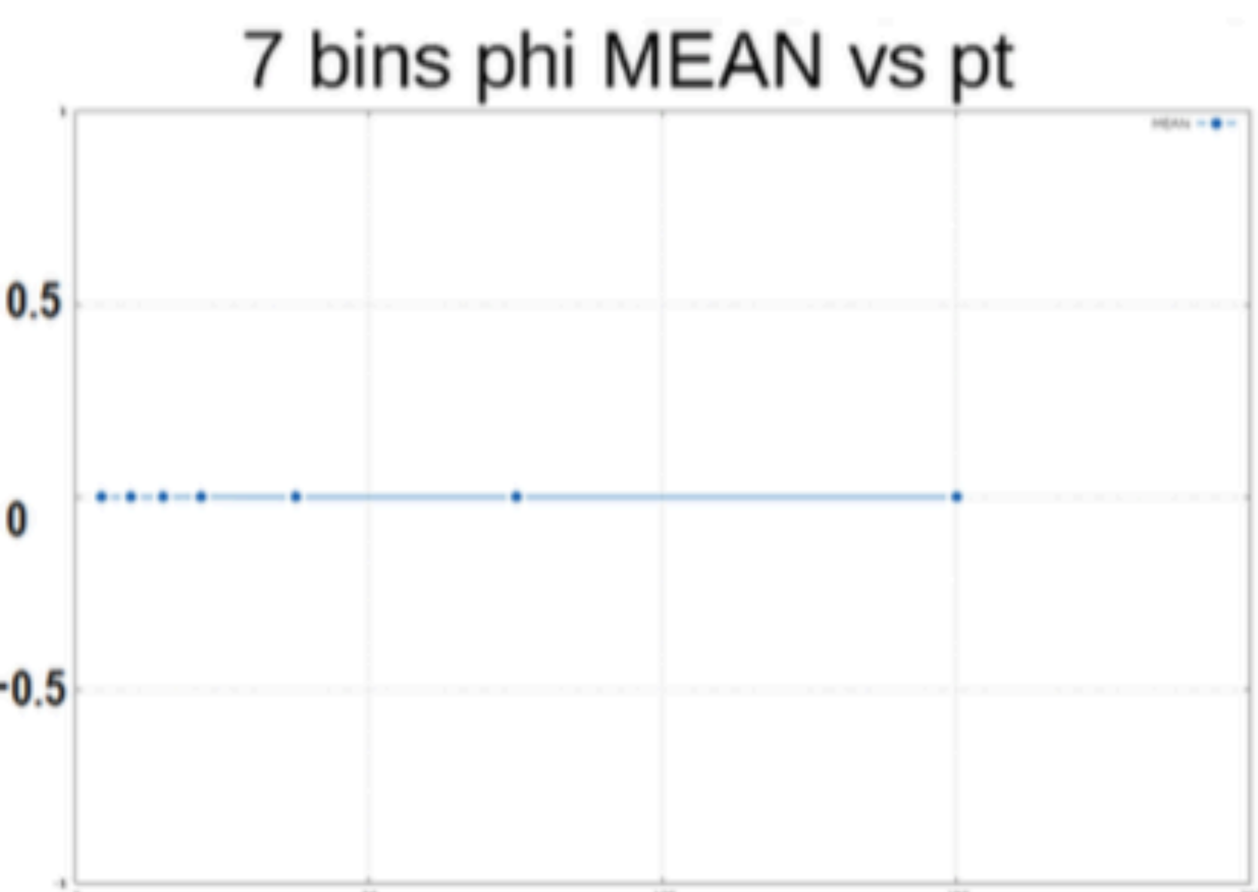
- We plot the following variable:

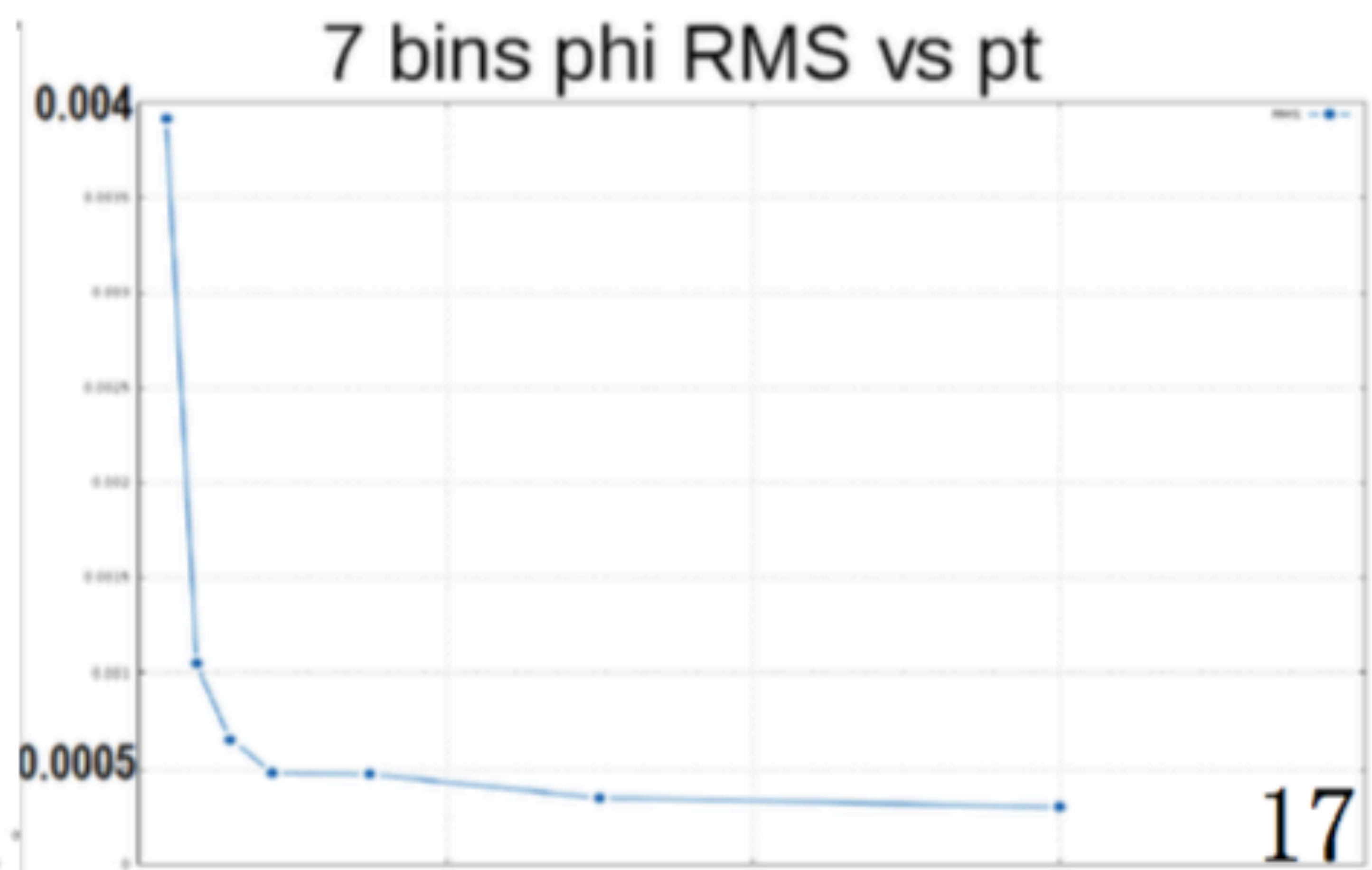
  [ϕ(generated) - ϕ(fit)]

- We have plotted the **mean** and the **RMS** as a function of pT,
- for **7 bins** in pT.

First pT bin







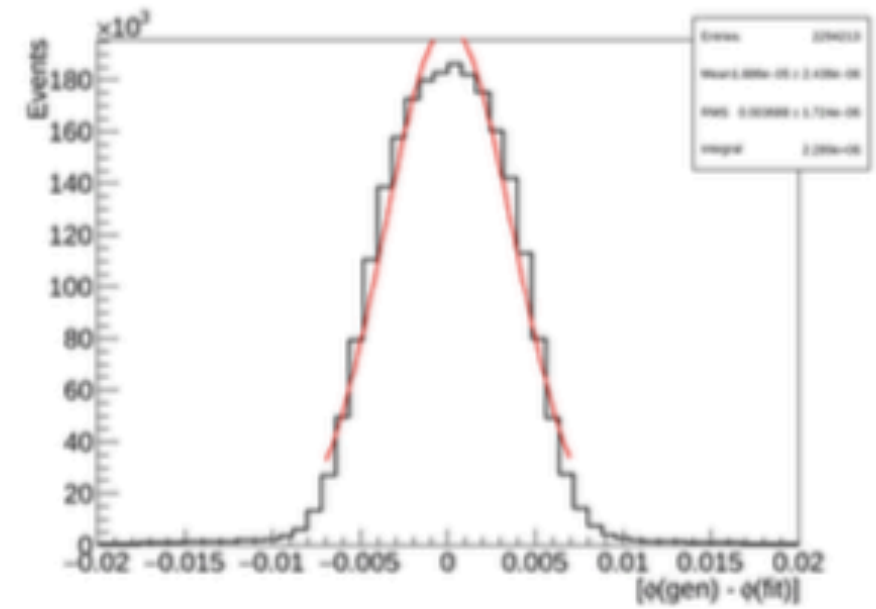
Plane rϕ − ϕ results

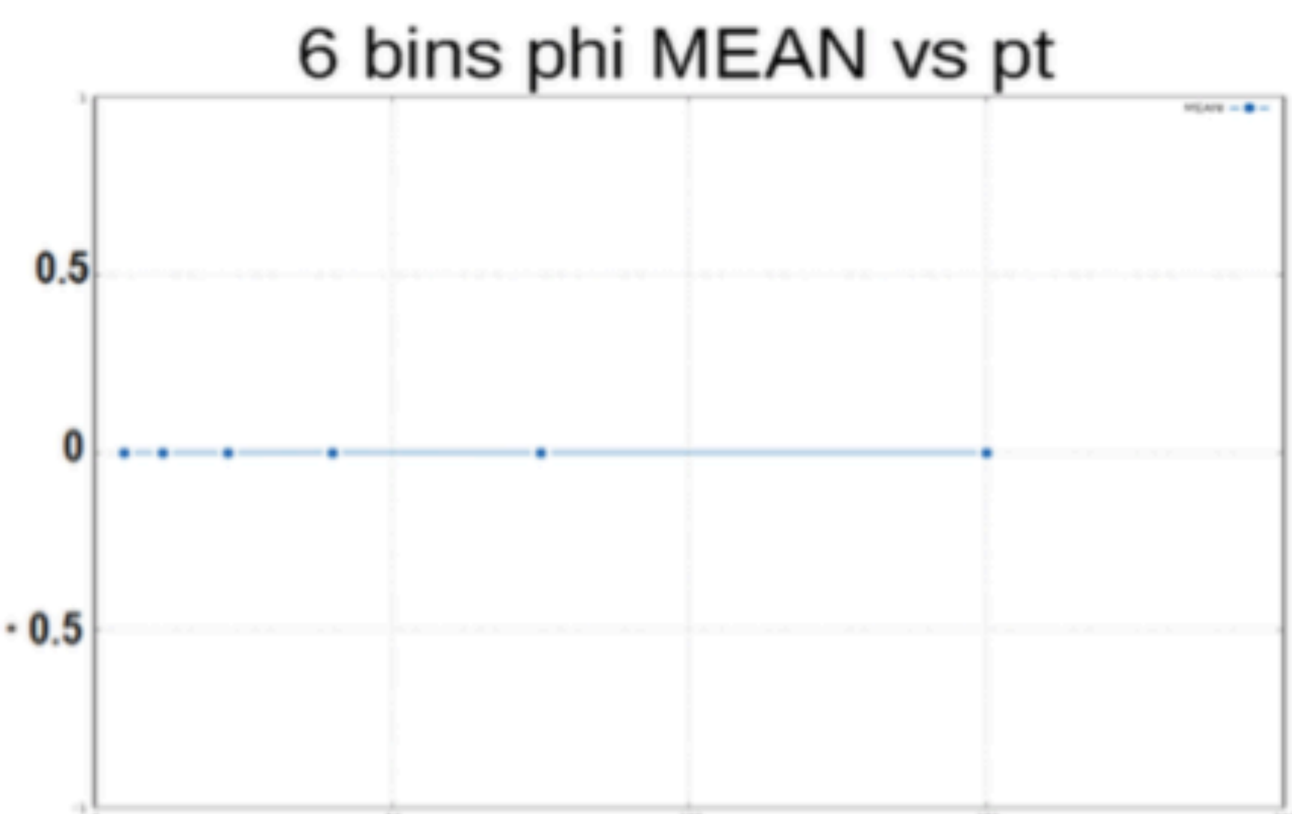
- We plot the following variable:

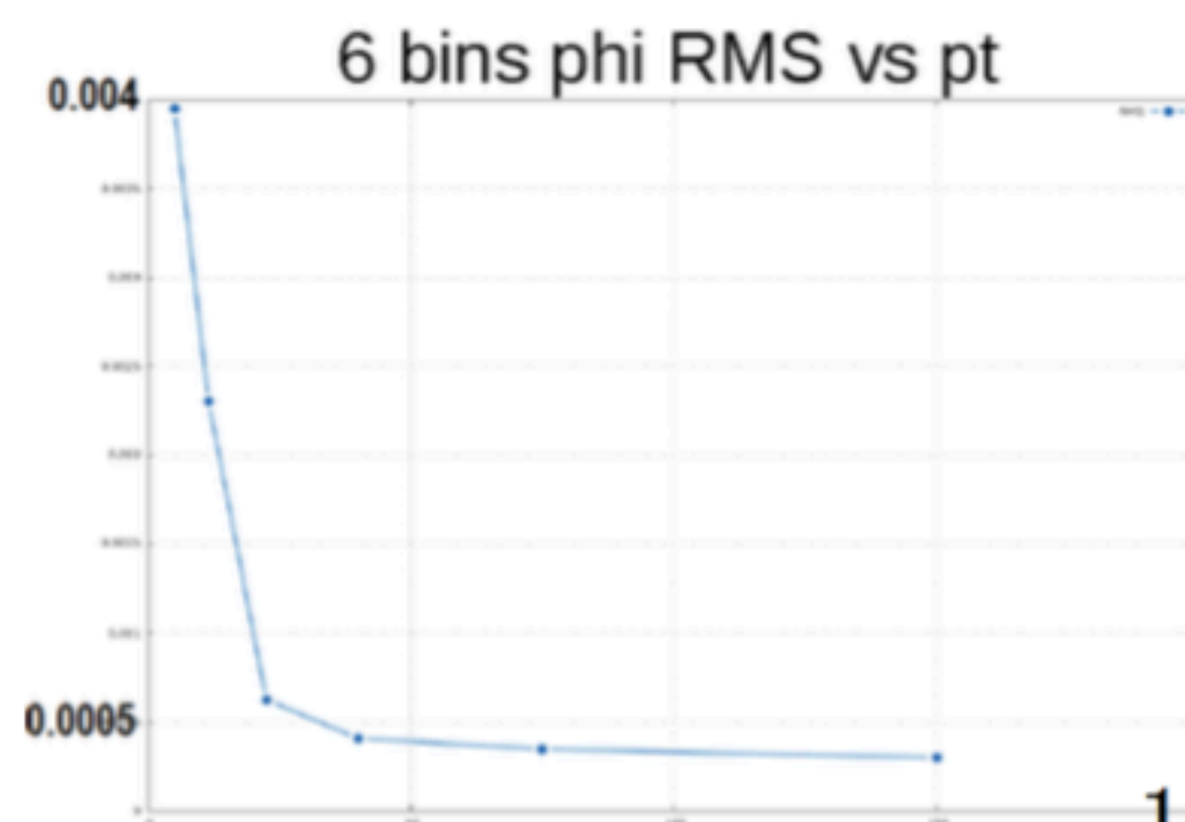
  [ϕ(generated) - ϕ(fit)]

- We have plotted the **mean** and the **RMS** as a function of pT,
- for **6 bins** in pT.

First pT bin







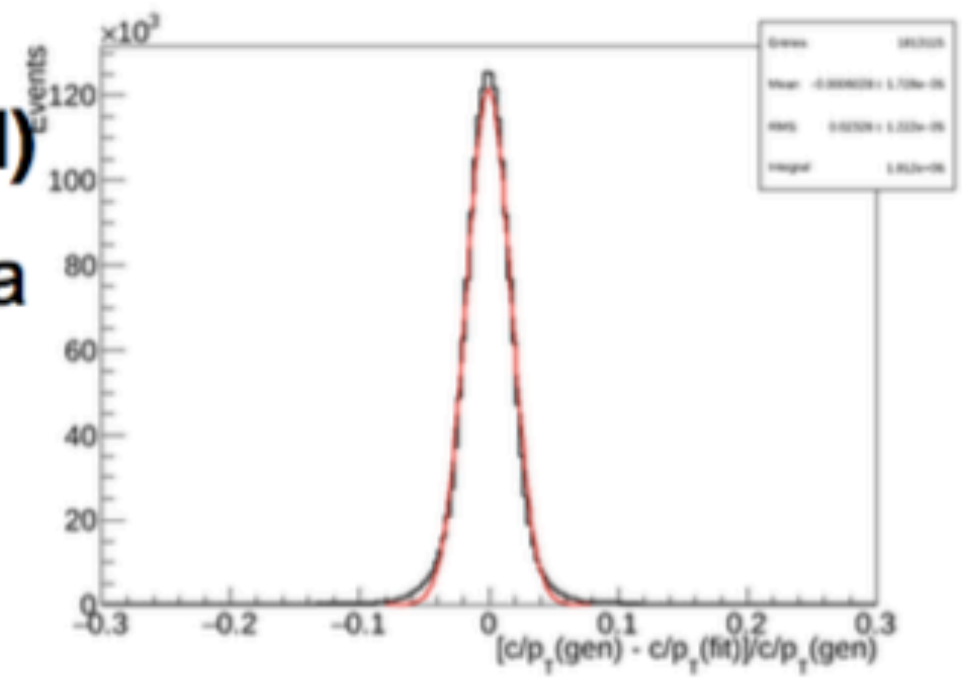
Plane rφ – c/pT results

- We plot the following variable:

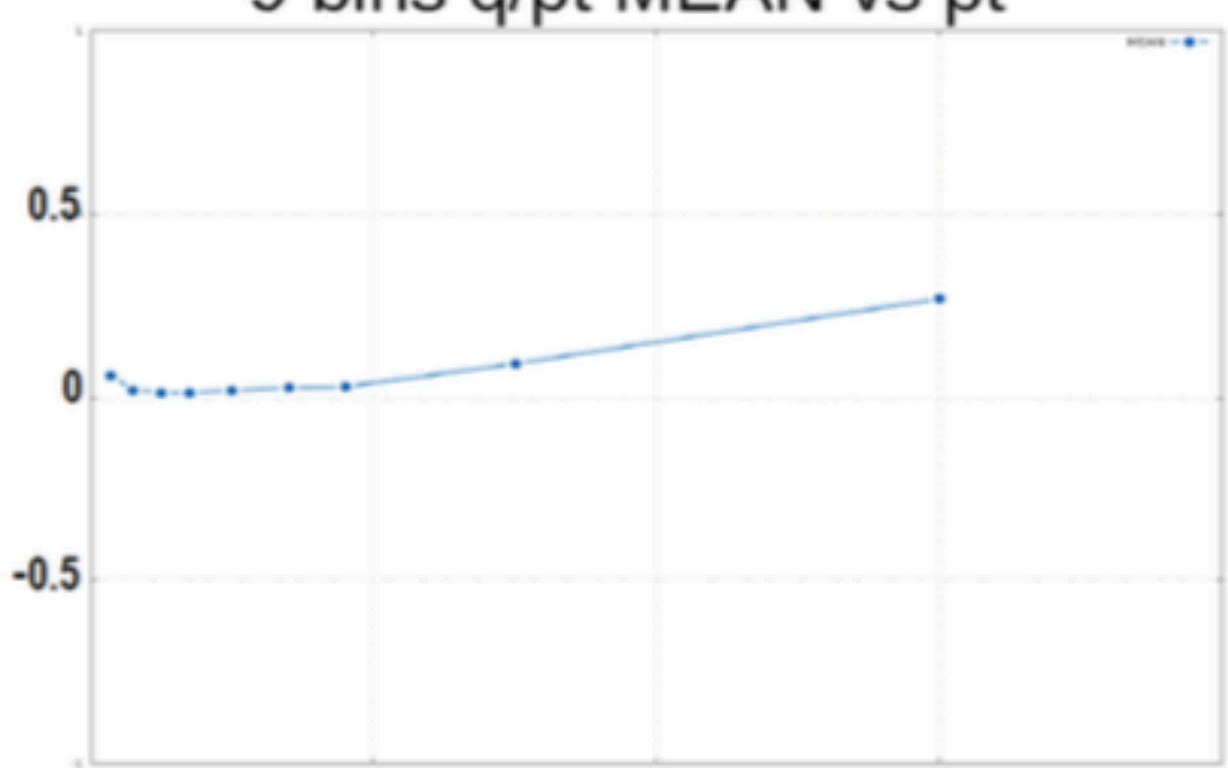
[c/pT(generated) – c/pT(fit)]/c/pT(generated)

- We have plotted the **mean** and the **RMS** as a function of pT,
- for **9 bins** in pT.

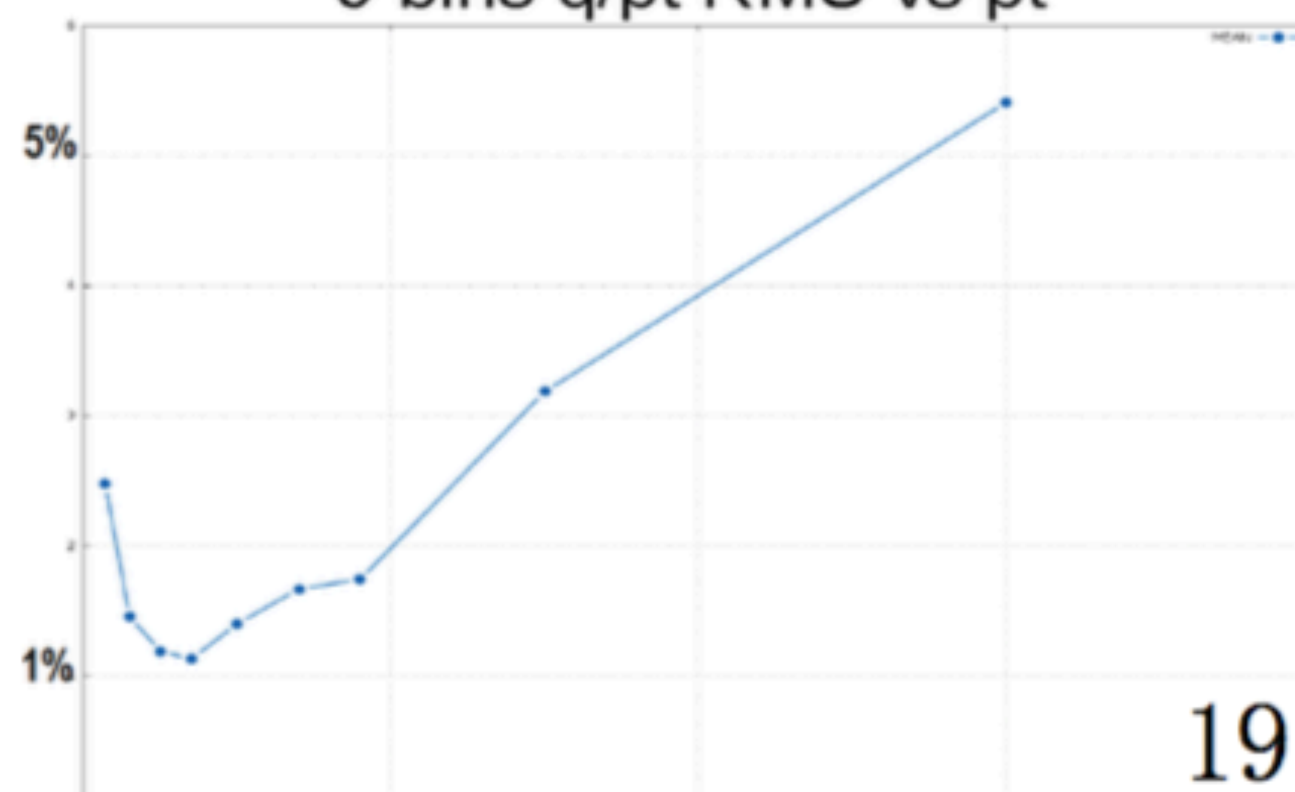
First pT bin



9 bins q/pt MEAN vs pt



9 bins q/pt RMS vs pt



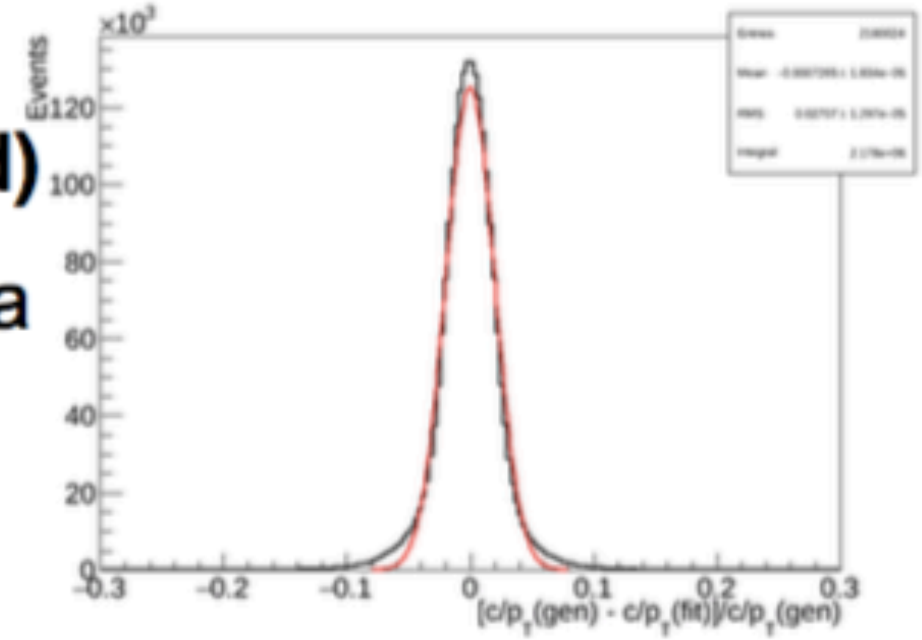
Plane rϕ – c/pT results

First pT bin

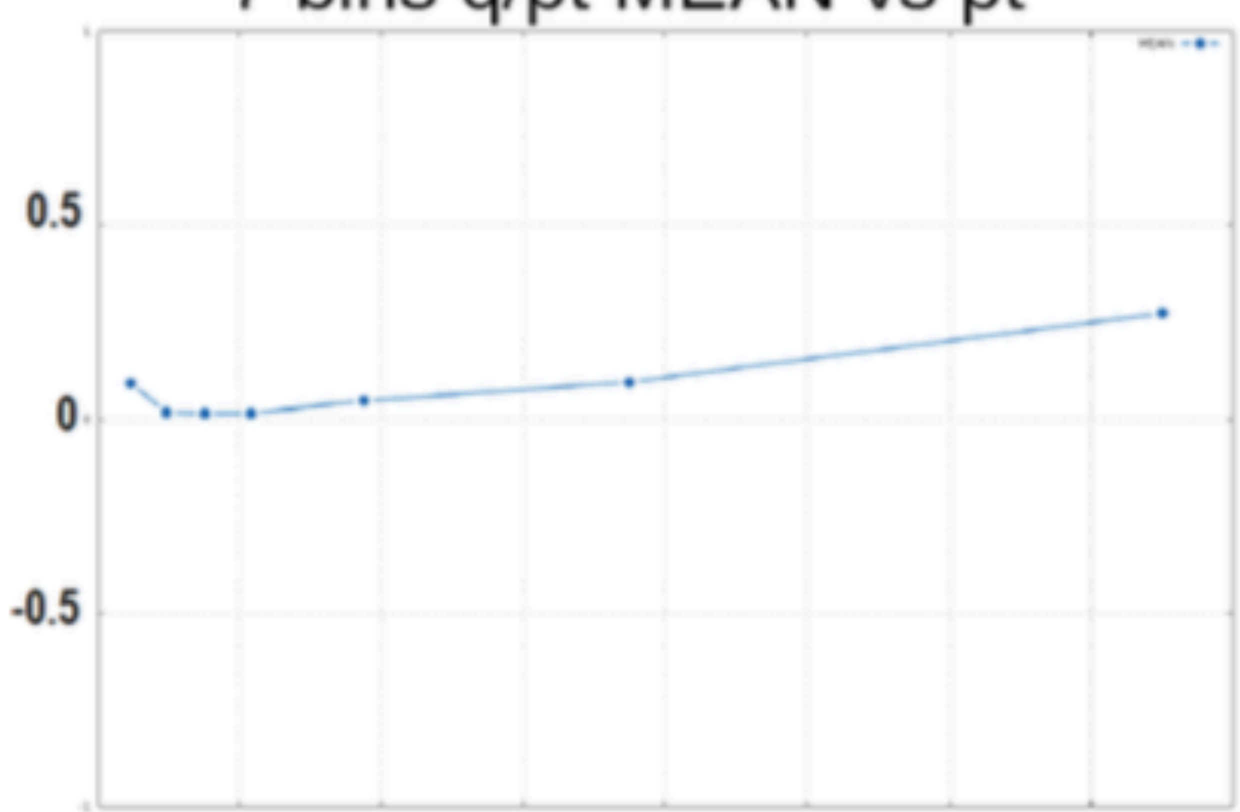
- We plot the following variable:

[c/pT(generated) – c/pT(fit)]/c/pT(generated)

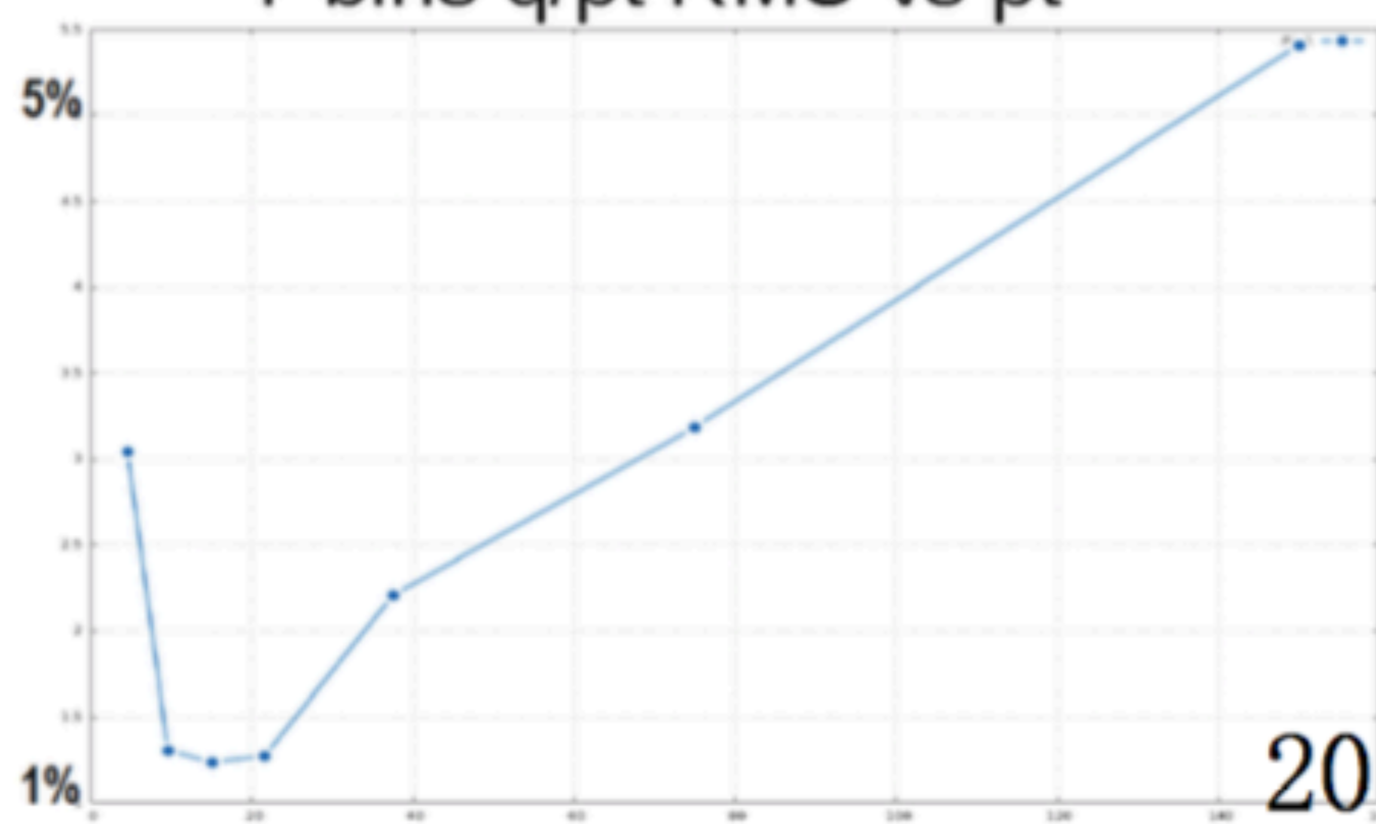
- We have plotted the **mean** and the **RMS** as a function of pT,
- for **7 bin** in pT.



7 bins q/pt MEAN vs pt



7 bins q/pt RMS vs pt



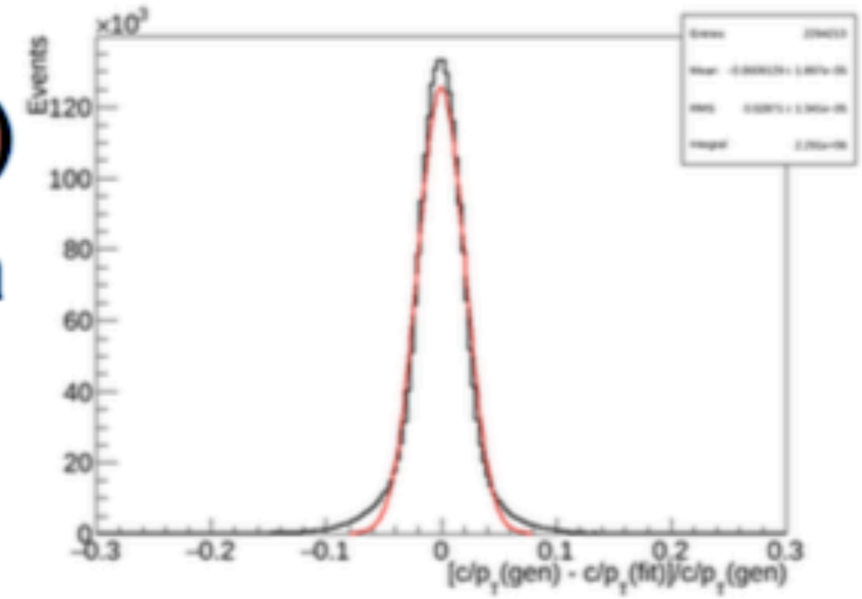
Plane rϕ – c/pT results

- We plot the following variable:

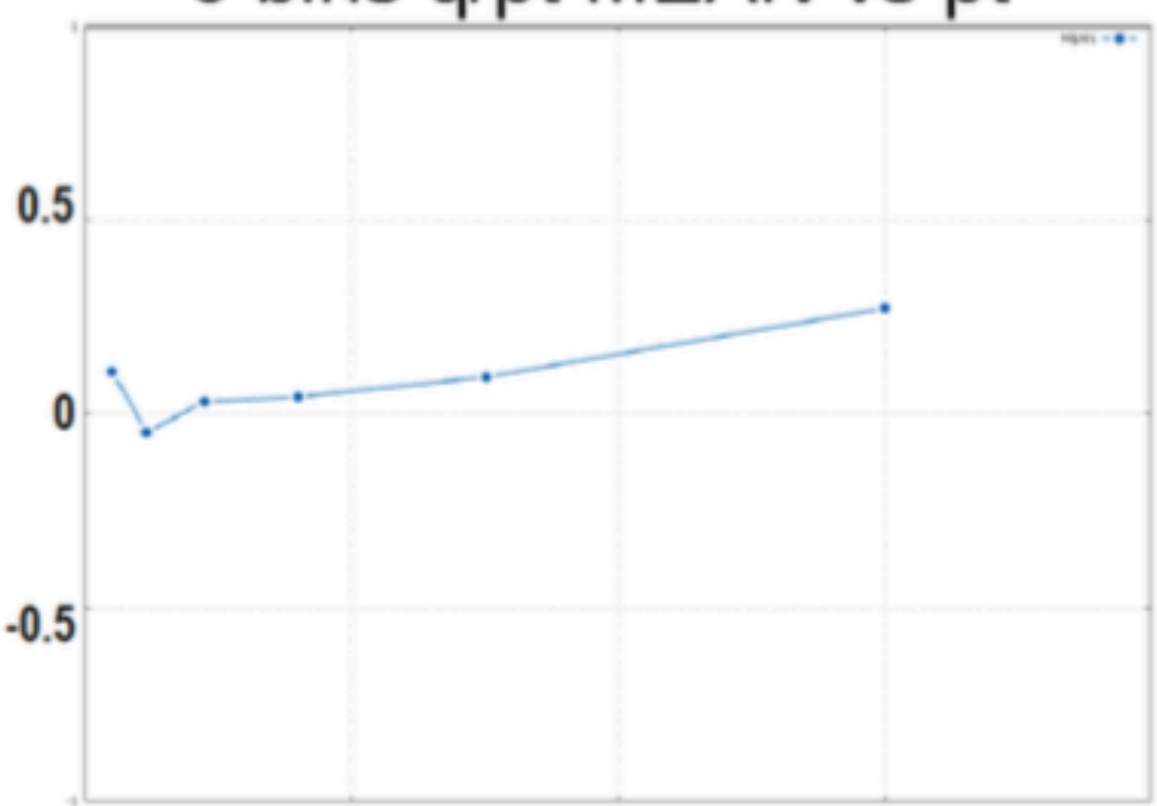
[c/pT(generated) – c/pT(fit)]/c/pT(generated)

- We have plotted the **mean** and the **RMS** as a function of pT,
- for **6 bin** in pT.

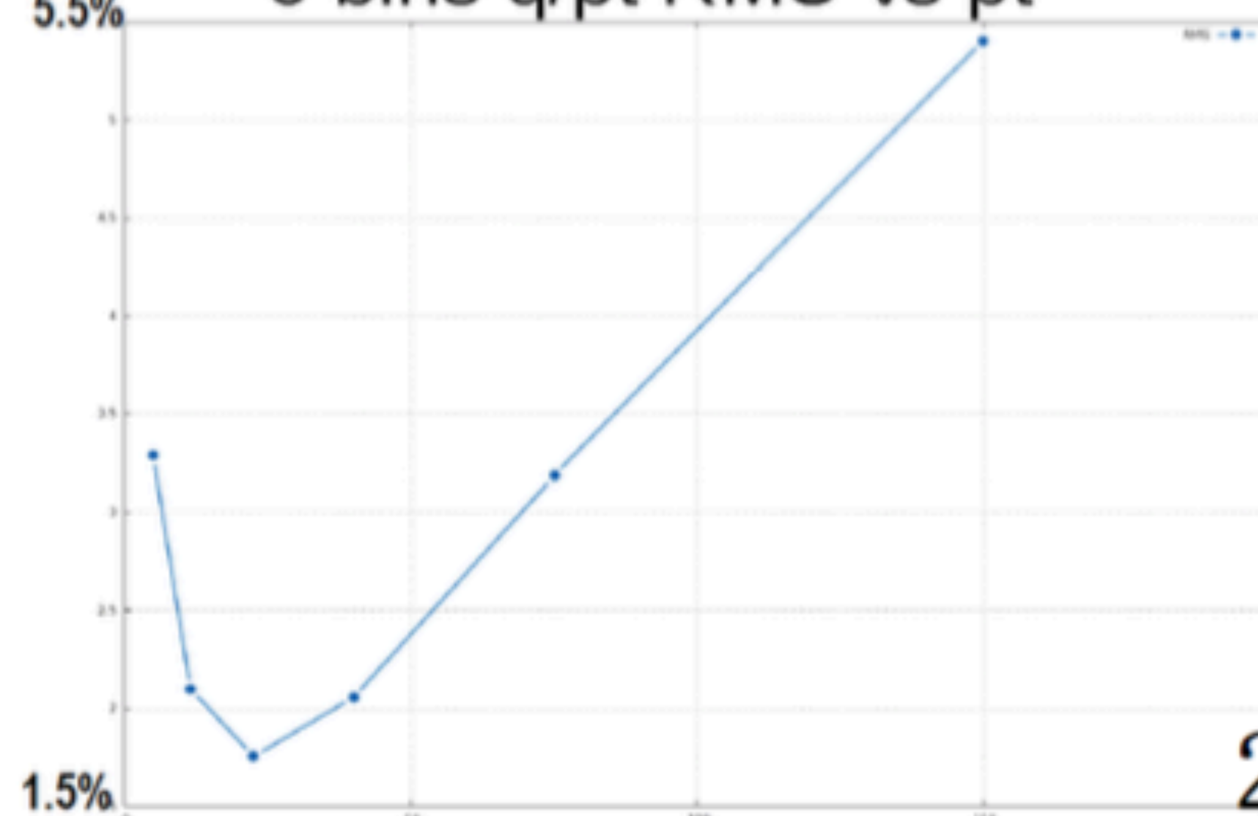
First pT bin



6 bins q/pt MEAN vs pt



6 bins q/pt RMS vs pt



Plane rz using straight-line

Straight-line equation

- rz plane using only layer 1 and 3, two points we get the slope q (i.e. η) and the intercept b (i.e. z_0) of a line.
- Using a simple python code https://bitbucket.org/lstorchi/gf_fit

$$q = \frac{z_3 - z_1}{r_3 - r_1}$$

$$b = z_1 - q \times r_1$$

Straight-line equation

Tower 18	Mean Δη	Standard deviation
-0,6 to 0,4 no bin	0.0043	0.011
-0,6 to -0,4	0.023	0.0081
-0,4 to -0,2	0.0055	0.0038
-0,2 to 0,0	0.00051	0.0026
0,0 to 0,2	-0.00051	0.0026
0,2 to 0,4	-0.0055	0.0038

Tower 18	Mean Δz_0 cm	Standard deviation cm
-0,6 to 0,4 no bin	-0.0037	0.090
-0,6 to -0,4	-0.019	0.096
-0,4 to -0,2	-0.011	0.089
-0,2 to 0,0	-0.0035	0.086
0,0 to 0,2	0.0035	0.086
0,2 to 0,4	0.011	0.089

Conclusion

Conclusion

- The fit is done separately for two planes: rz and rphi
- Fit of four parameters **(z_0, η, c/p_T, ϕ)** is performed
- The resolutions in the barrel are:

rz plane			
	20 bins in η	10 bins in η	5 bins in η
$\Delta\eta$	0.0033 to 0.0036	0.0040 to 0.0043	0.0057 to 0.0062
Δz_0 cm	0.101 to 0.126	0.123 to 0.150	0.171 to 0.207
rϕ plane			
	9 bins in pT	7 bins in pT	6 bins in pT
$\Delta\phi$	0.00038 to 0.0081	0.00030 to 0.0039	0.00030 to 0.0040
$\Delta c/pT$	1.1% to 5.4%	1.2% to 5.4%	1.7 % to 5.4%

The sets of constants needed to obtain the best resolution are 38 for each barrel tower

Conclusion

Open points:

- ✓ d_0 how to fit
- ✓ FPGA porting, first step use 16-bit integer representation
- ✓ Forward region almost ready, do we need to perform some test ?

Integer Representation

- **Master Equation:**

  $[p \text{ (parameter)}]_{2\times1} = [A \text{ (constant)}]_{2\times12} * [X \text{ (stub coordinates)}]_{12\times1} + [q \text{ (constant)}]_{2\times1}$

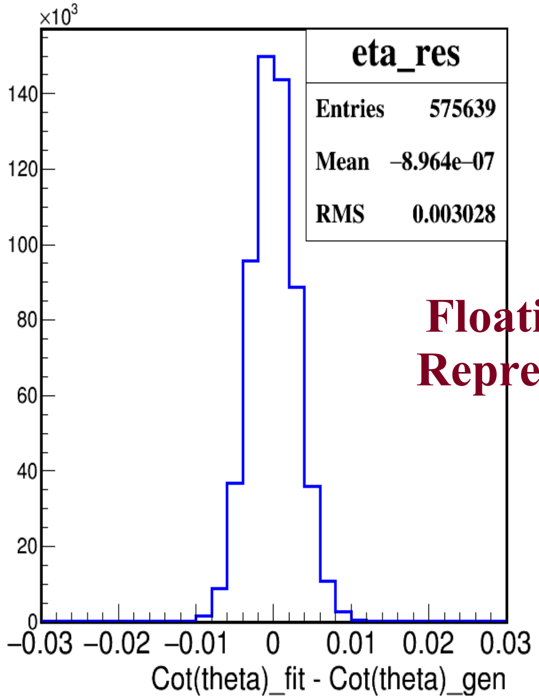
  We already know the set of constants "A" and "q"

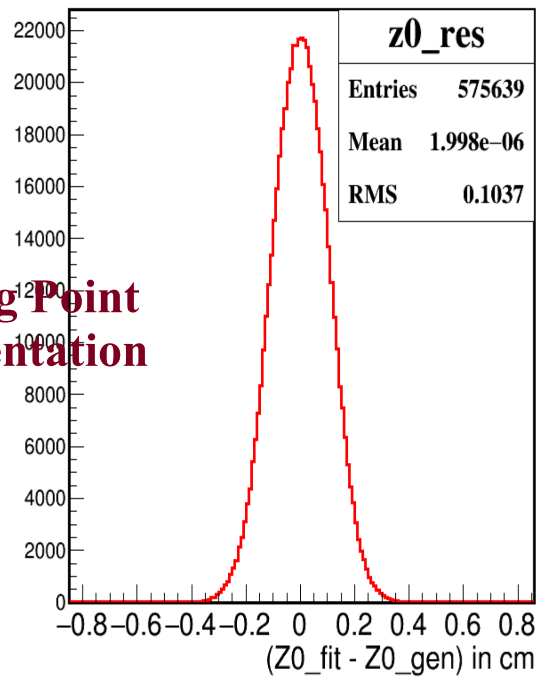
  For a set of Stub Coordinates "X", we want to compute "p"

- Transform the set of constants and stub coordinates to integer (currently using int16_t)

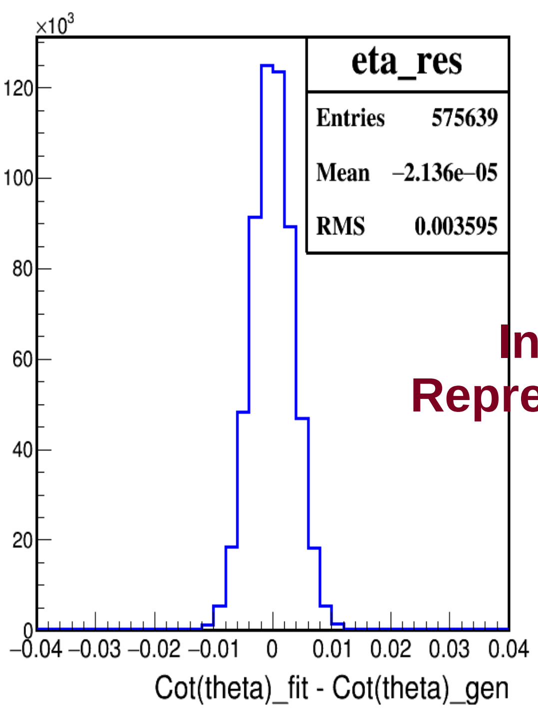
  - since int16_t only covers $\sim \pm 32$k, there is a limlit by which one can scale up a particular parameter during the transformation.

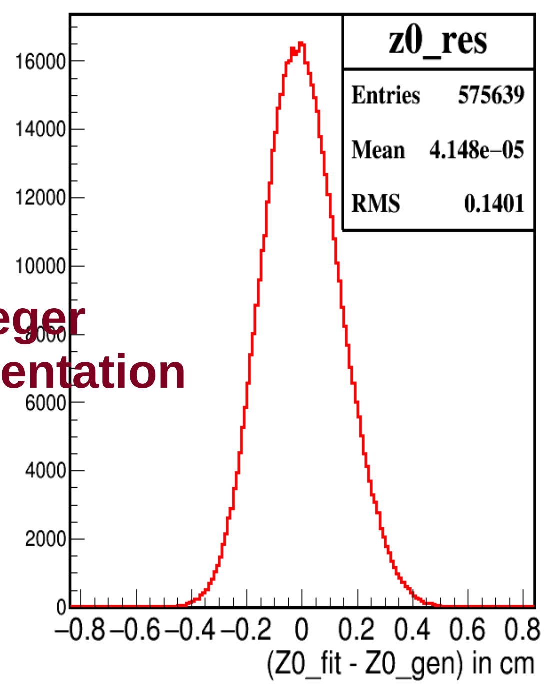
Resolution: Cot(theta)   Resolution: Z0





Floating Point Representation





Integer Representation

Plane: r-z

Bin:

1 Pt Bin 2-200

1 Phi Bin 1.1-2.9

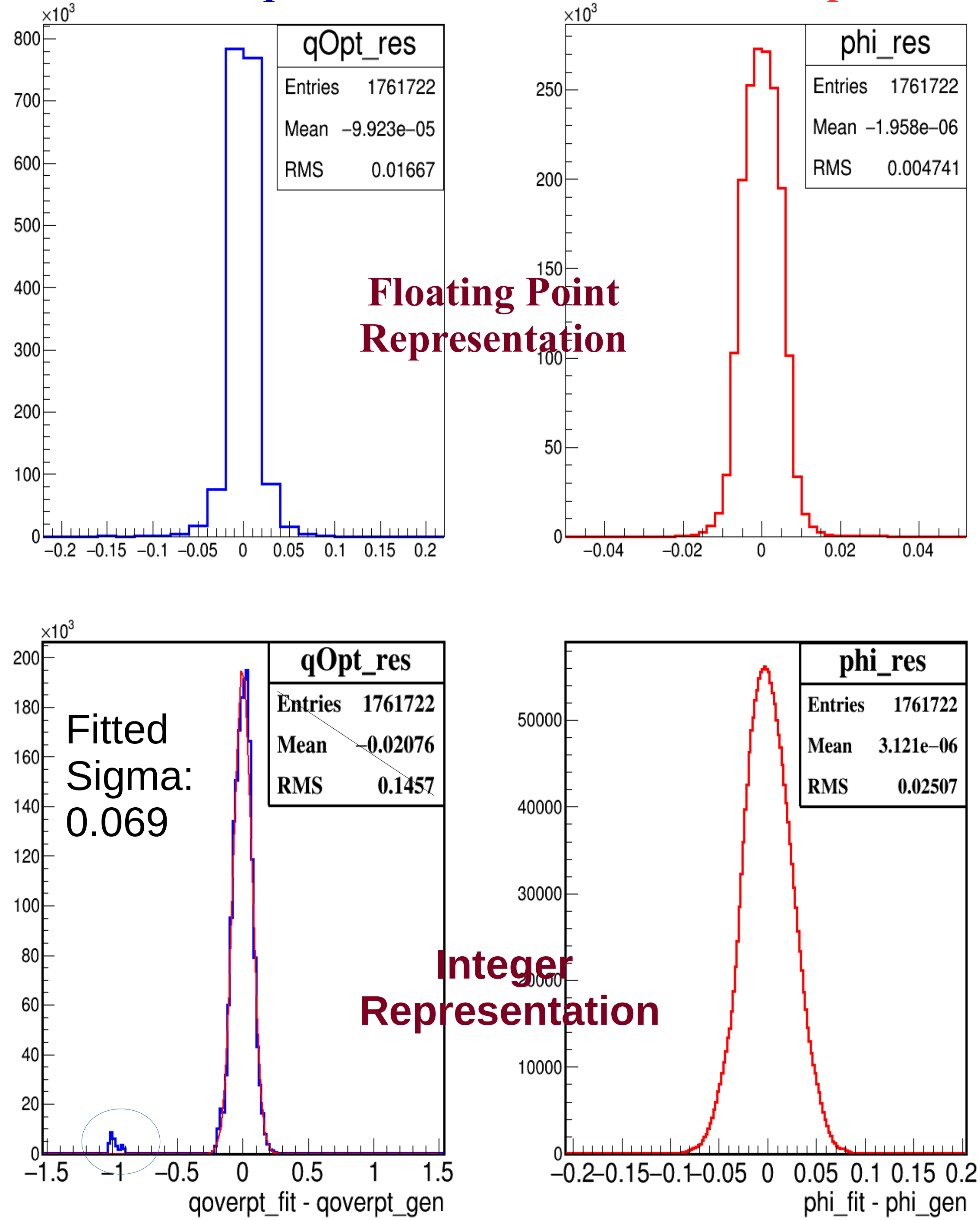
Eta Bin 0.3-0.4 (0ne of 10 bins)

Resolution	Cot(theta)	Z0
Float	0.0030	0.103 cm
Integer(16)	0.0035	0.140* cm

***can be improved using slightly bigger range than int16**

**Mean values of the integer represenation distributions have been adjusted with offset corrections

Resolution: qOverPt **Resolution: phi**



Plane: r-phi

Bin:

Pt Bin 2-5 (One of the 9 bins)
1 Phi Bin 1.1 to 2.9
1 Eta Bin -0.6 to 0.4
Charge +ve

Resolution	qOverPt	phi
Float	0.016	0.0047
Integer(16)	0.069	0.0250

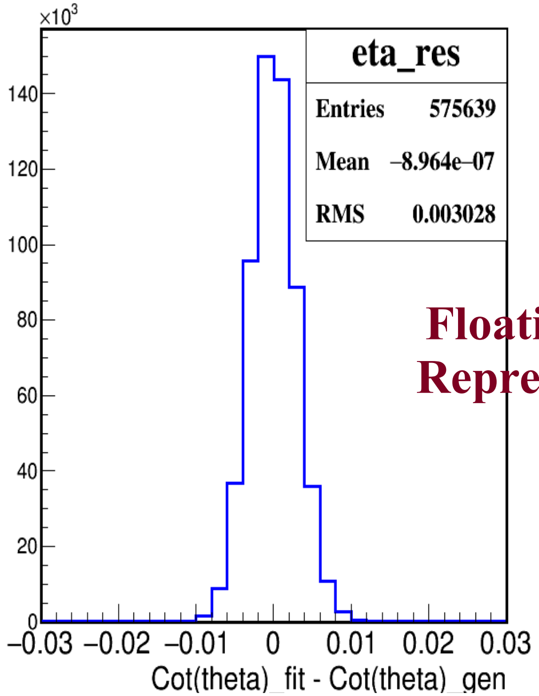
r-phi plane resolution numbers can be improved using better precision on stub coordinates below mm

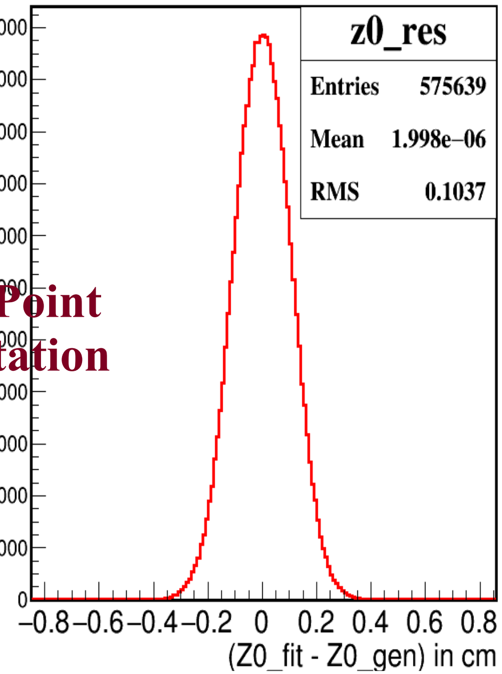
**Mean values of the integer represenation distributions have been adjusted with offset corrections

Back Up

Resolution: Cot(theta)

Resolution: Z0





Floating Point Representation

Plane: r-z

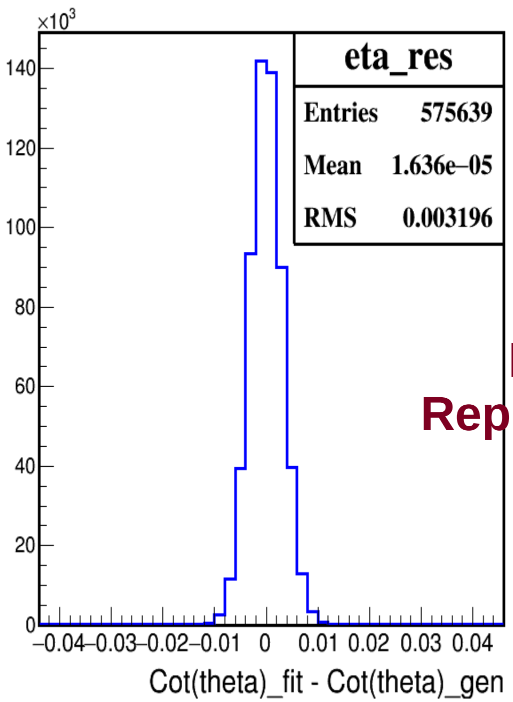
Bin:

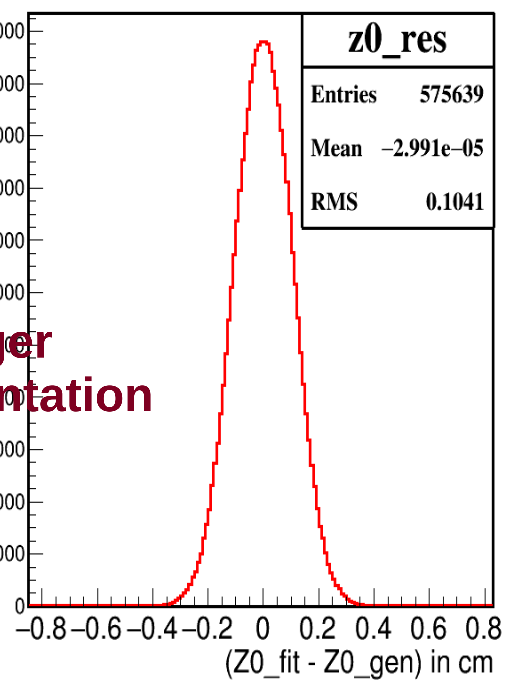
1 Pt Bin 2-200
1 Phi Bin 1.1-2.9
Eta Bin 0.3-0.4 (0ne of 10 bins)

Changes wrt slide 2:

Stub coordinates in 0.1 mm precision (earlier it was 1.0 mm)

Int Type 32, but essentially can be covered by 20 bits



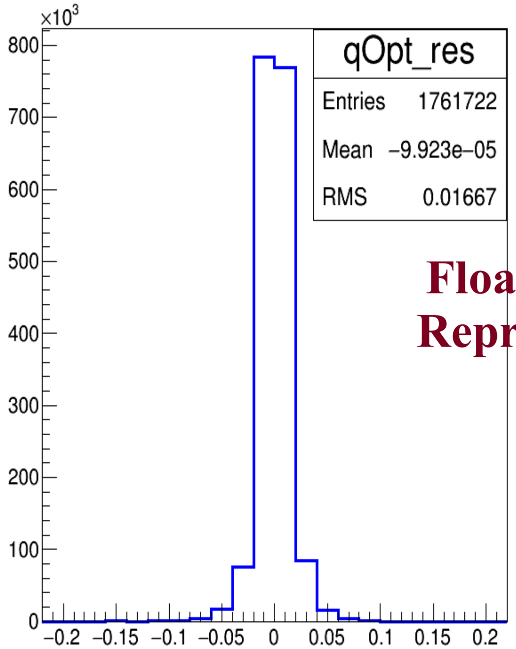


Integer Representation

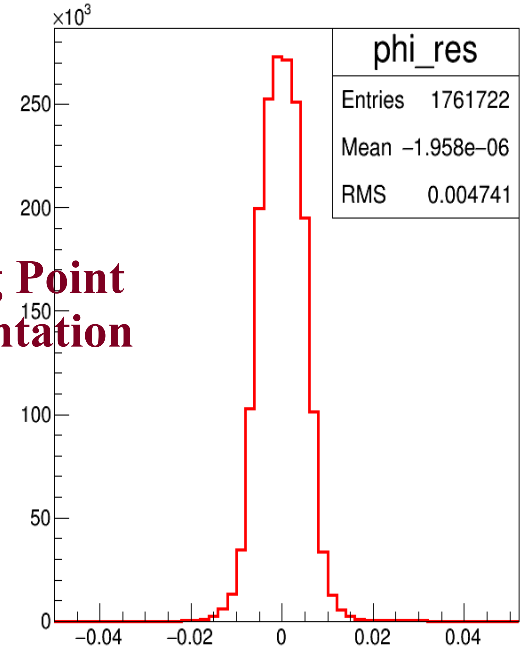
Resolution	Cot(theta)	Z0
Float	0.0030	0.103 cm
Integer(32)	0.0031	0.104 cm

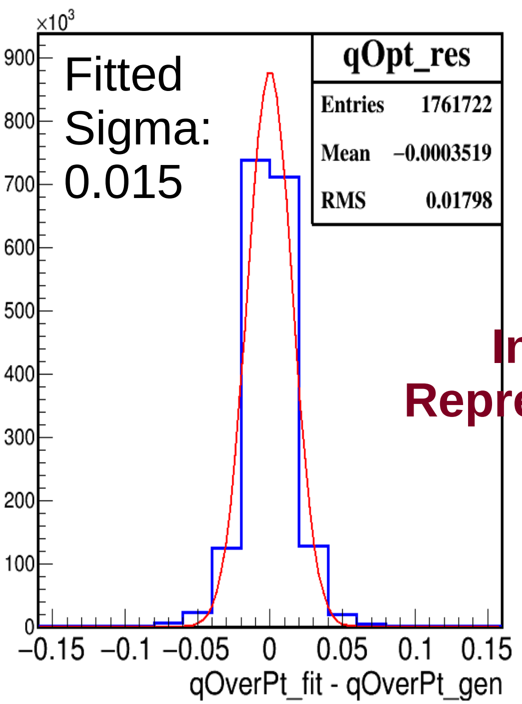
**Mean values of the integer represenation distributions have been adjusted with offset corrections

Resolution: qOverPt **Resolution: phi**

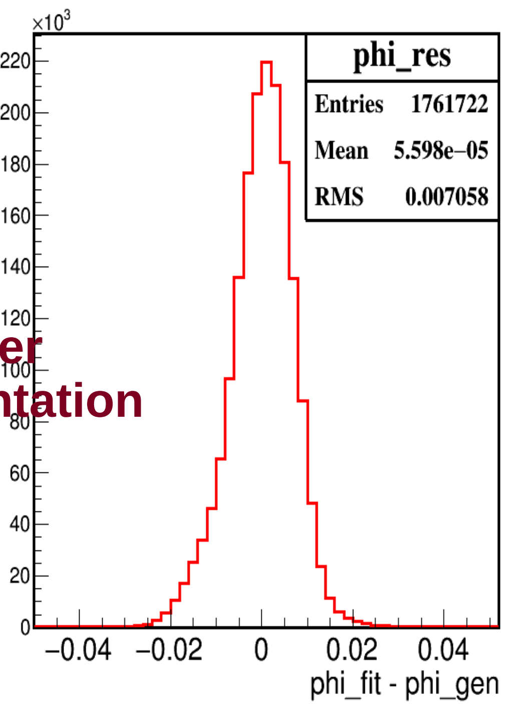


Floating Point Representation





Integer Representation



Plane: r-phi

Bin:

Pt Bin 2-5 (One of the 9 bins)
1 Phi Bin 1.1 to 2.9
1 Eta Bin -0.6 to 0.4
Charge +ve

Changes wrt slide 2:

Stub coordinates in 10 times better precision

Int Type 32, but essentially can be covered by 20 bits

Resolution	qOverPt	phi
Float	0.016	0.0047
Integer(32)	0.017	0.0070

**Mean values of the integer represenation distributions have been adjusted with offset corrections

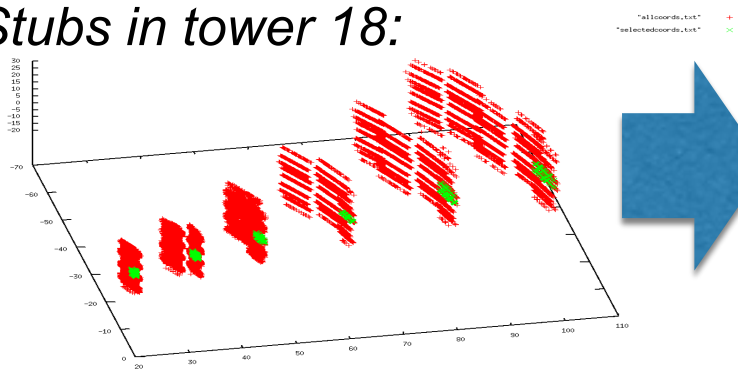
Track Finding

Proposed solution for the CMS track finding in the phase 2 of LHC:

Patter recognition → stubs coordinates of the tracks are passed to the AM chips that compare, in parallel, each stub combination with the patterns stored. Matched stubs sent to the fitting.

Fitting →based on the PCA (Principal Component Analysis) statistical method: a linear relation exists between stub coordinates $\mathbf{x}_i$ and the track parametri $\mathbf{p}_i$ (z_0, d_0, η, c/p_T and ϕ)

Stubs in tower 18:



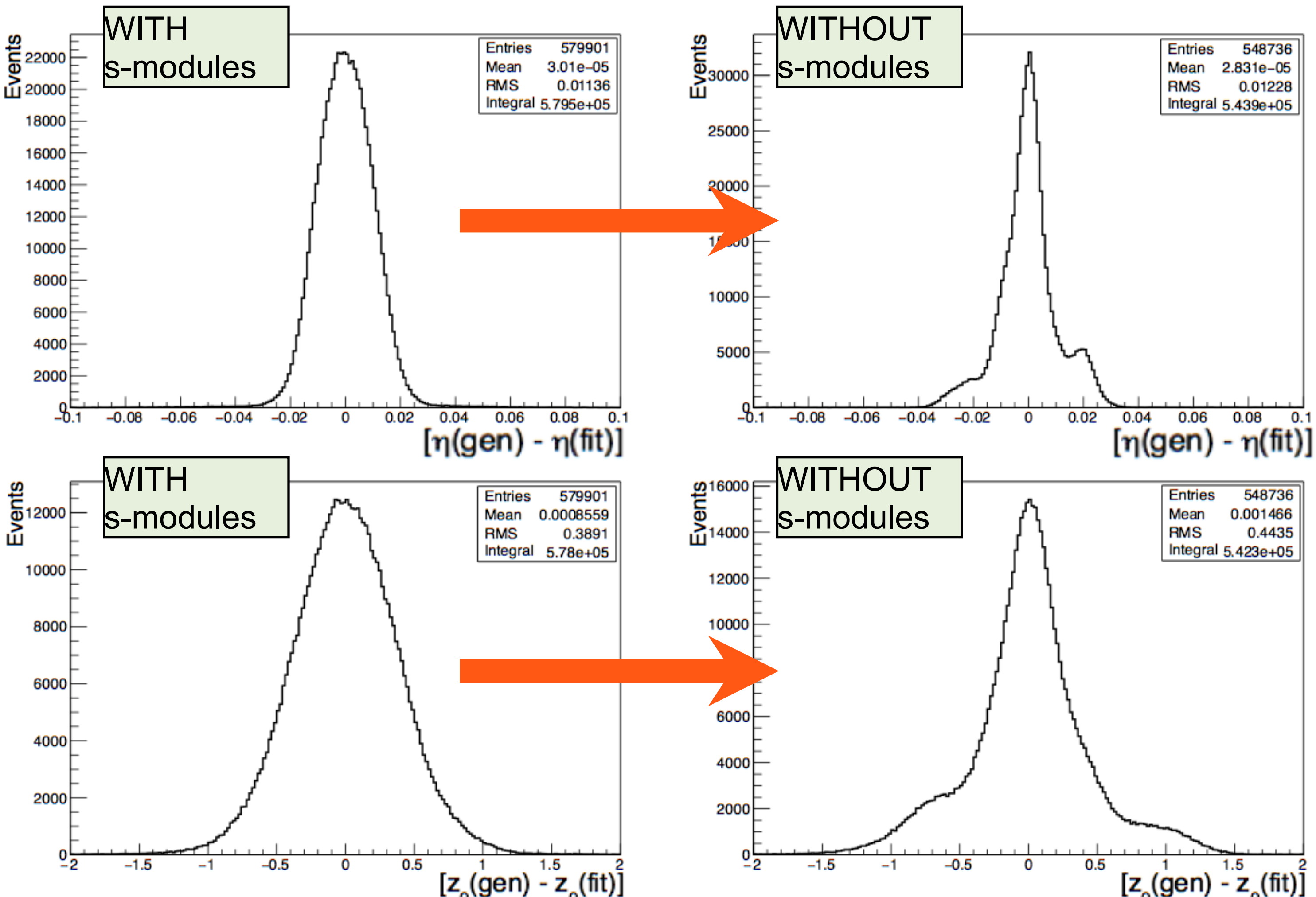
Fitting formula:

$$p_i(PCA) = \sum_j A_{ij} x_j + q_i$$

The PCA constants $\mathbf{A}_{ij}$ and $\mathbf{q}_i$ have to be pre-calculated and stored for each geometrical sector of the tracker.

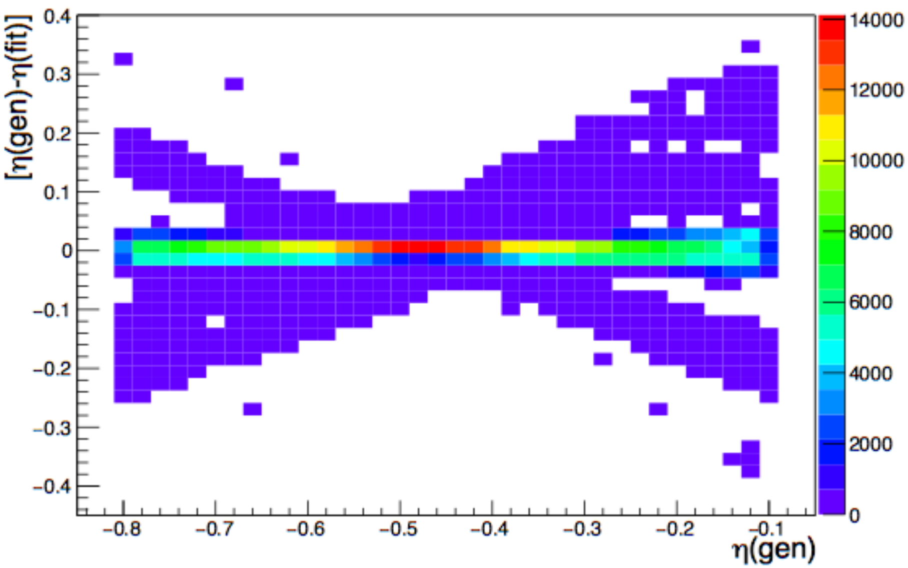
Exclude s-modules - plane rz

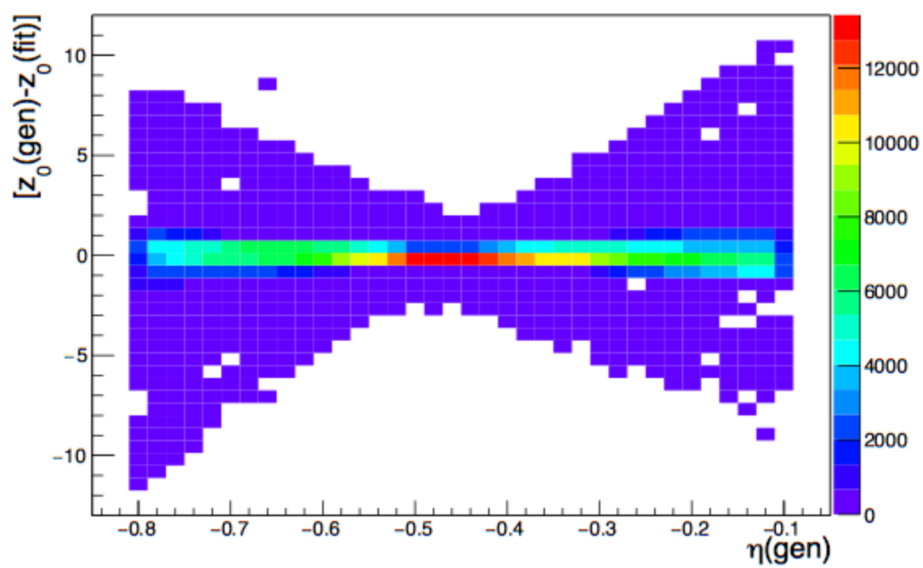
- We can think to reduce the number of costants used in the fit, using only the first three layers for the rz plane, i.e. not using the s-modules
- What happens if we do not consider the s-modules?



Exclude s-modules - plane rz

- We can think to reduce the number of costants used in the fit, using only the first three layers for the rz plane, i.e. not using the s-modules
- What happens if we do not consider the s-modules?





Exclude s-modules - plane rz

- We can think to reduce the number of costants used in the fit, using only the first three layers for the rz plane, i.e. not using the s-modules
- What happens if we do not consider the s-modules?

WITH s-modules ($-0.8<\eta<-0.1$)	WITHOUT s-modules ($-0.8<\eta<-0.1$)	WITHOUT s-modules ($-0.6<\eta<-0.1$)	WITHOUT s-modules ($-0.4<\eta<-0.1$)