Lambda b Polarization & Angles

Asher Sabbagh & Mero Elmarassy

Lambda b

Λ_b^0

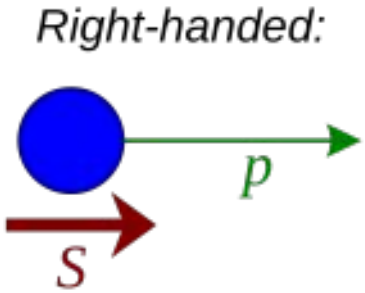
- Particles similar to proton in structure with one u quark replaced with a beauty (b) quark (udb)
- B quarks come from Z decay
- Decays via weak force to lambda 0 (uds) and dilepton:

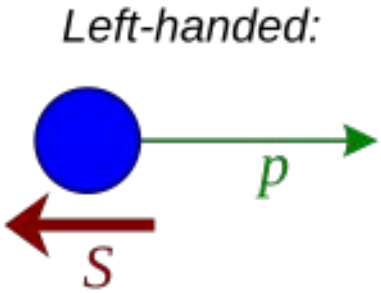
$$\Lambda_b^0 \rightarrow \Lambda^0 + \mu^+ \mu^-$$

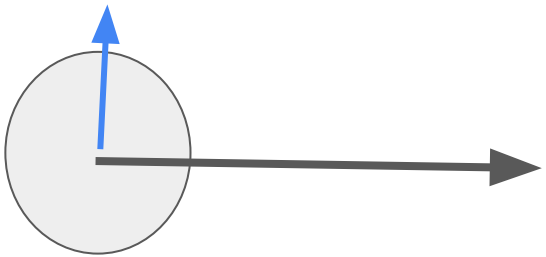
$$\Lambda^0 \rightarrow p^+ + \pi^-$$

Lambda 0 particle

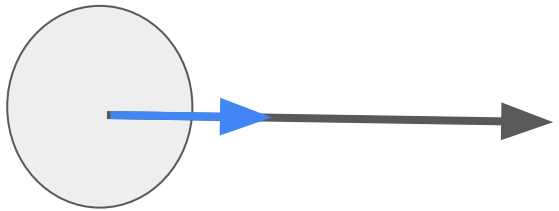
- Since this is a weak force interaction, the spin of the lambda b matters
- Helicity = component of spin in direction of momentum
- Polarization of Lb comes from polarization of Z boson







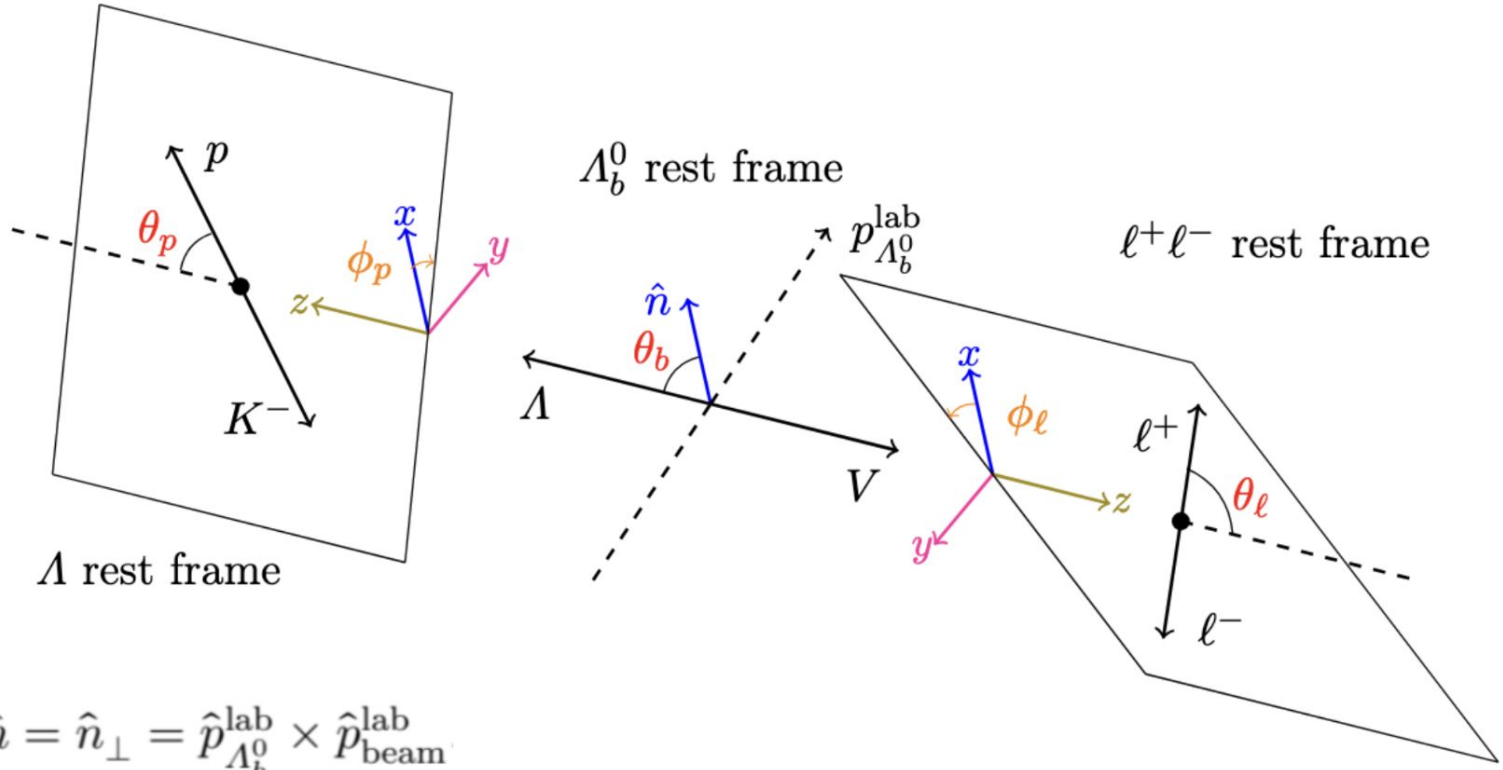
H = 0
(longitudinal)



H = 1
(transverse)

θ: angle between $\vec{p}$ and z-axis

ϕ : polar angle



$$\hat{n} = \hat{n}_\perp = \hat{p}_{\Lambda_b^0}^{\rm lab} \times \hat{p}_{\rm beam}^{\rm lab}$$

$$\hat{n}_\parallel = \hat{p}_{\Lambda_b^0}^{\rm lab}$$

Calculating Angles

- Angles were calculated by boosting into the Lambda b frame, then finding the vector angle between the momenta in this frame

- All angles were calculated for two ROOT files: one with pure physics, and one with simulated detector effects

Detector effects

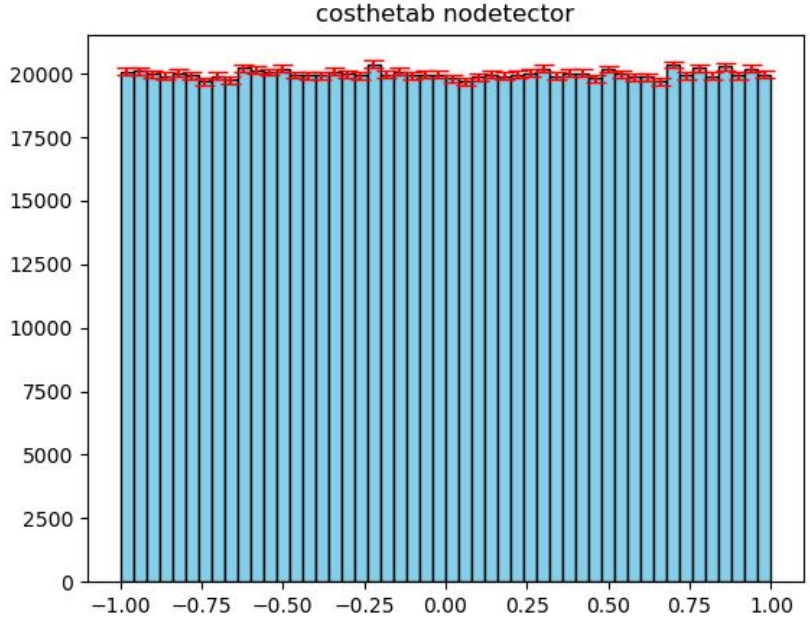
Need to fit the angles to a pdf: (Mero)

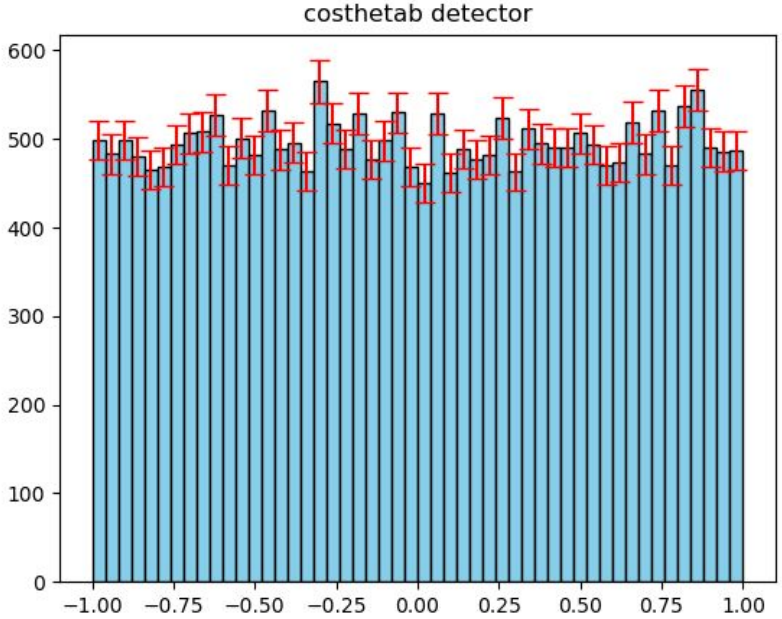
$$\frac{\mathrm{d}^6\Gamma}{\mathrm{d}q^2\,\mathrm{d}\vec{\Omega}} = \frac{3}{32\pi^2}\Big(\sum_{i=0}^{34} K_i(q^2) f_i(\vec{\Omega})\Big)$$

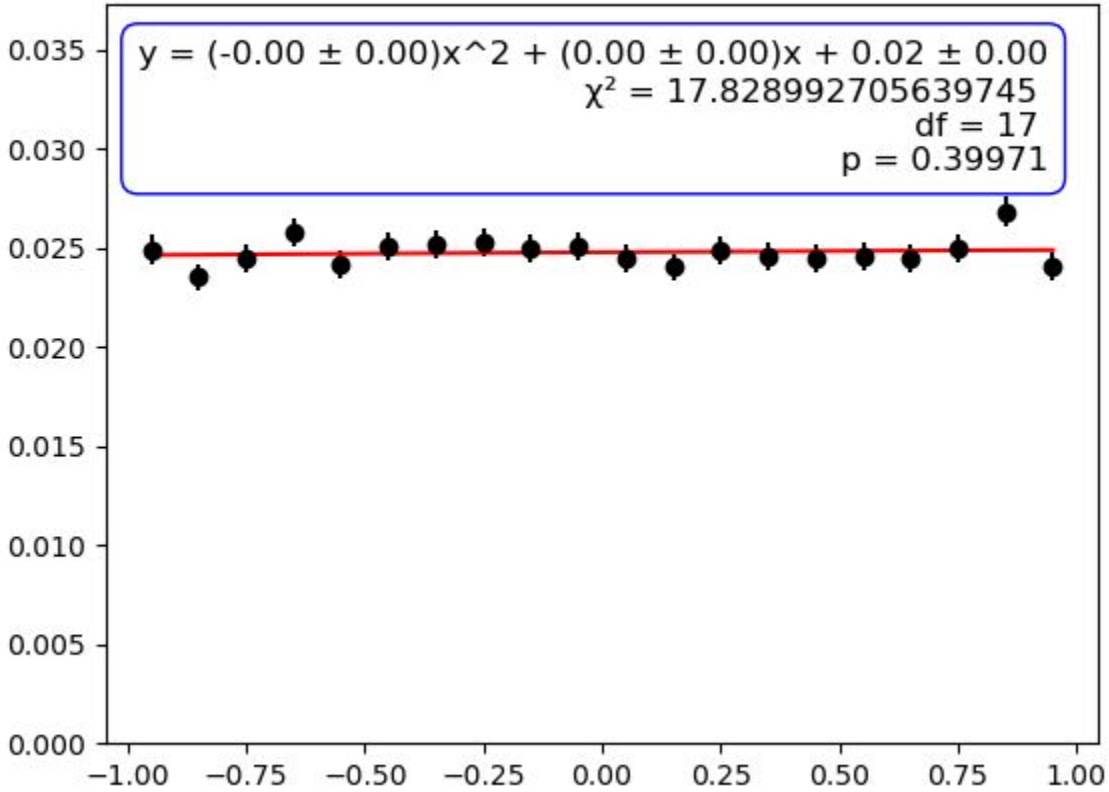
$$\begin{aligned}\frac{\mathrm{d}^6\Gamma}{\mathrm{d}q^2\,\mathrm{d}\vec{\Omega}} = \frac{3}{32\pi^2}\Big(& \left(K_1 \sin^2\theta_l + K_2\cos^2\theta_l + K_3\cos\theta_l\right) + \\ & \left(K_4\sin^2\theta_l + K_5\cos^2\theta_l + K_6\cos\theta_l\right)\cos\theta_b + \\ & \left(K_7\sin\theta_l\cos\theta_l + K_8\sin\theta_l\right)\sin\theta_b\cos\left(\phi_b+\phi_l\right) + \\ & \left(K_9\sin\theta_l\cos\theta_l + K_{10}\sin\theta_l\right)\sin\theta_b\sin\left(\phi_b+\phi_l\right) + \\ & \left(K_{11}\sin^2\theta_l + K_{12}\cos^2\theta_l + K_{13}\cos\theta_l\right)\cos\theta + \\ & \left(K_{14}\sin^2\theta_l + K_{15}\cos^2\theta_l + K_{16}\cos\theta_l\right)\cos\theta_b\cos\theta + \\ & \left(K_{17}\sin\theta_l\cos\theta_l + K_{18}\sin\theta_l\right)\sin\theta_b\cos\left(\phi_b+\phi_l\right)\cos\theta + \\ & \left(K_{19}\sin\theta_l\cos\theta_l + K_{20}\sin\theta_l\right)\sin\theta_b\sin\left(\phi_b+\phi_l\right)\cos\theta + \\ & \left(K_{21}\cos\theta_l\sin\theta_l + K_{22}\sin\theta_l\right)\sin\phi_l\sin\theta + \\ & \left(K_{23}\cos\theta_l\sin\theta_l + K_{24}\sin\theta_l\right)\cos\phi_l\sin\theta + \\ & \left(K_{25}\cos\theta_l\sin\theta_l + K_{26}\sin\theta_l\right)\sin\phi_l\cos\theta_b\sin\theta + \\ & \left(K_{27}\cos\theta_l\sin\theta_l + K_{28}\sin\theta_l\right)\cos\phi_l\cos\theta_b\sin\theta + \\ & \left(K_{29}\cos^2\theta_l + K_{30}\sin^2\theta_l\right)\sin\theta_b\sin\phi_b\sin\theta + \\ & \left(K_{31}\cos^2\theta_l + K_{32}\sin^2\theta_l\right)\sin\theta_b\cos\phi_b\sin\theta + \\ & \left(K_{33}\sin^2\theta_l\right)\sin\theta_b\cos\left(2\phi_l+\phi_b\right)\sin\theta + \\ & \left(K_{34}\sin^2\theta_l\right)\sin\theta_b\sin\left(2\phi_l+\phi_b\right)\sin\theta\Big)\ .\end{aligned}$$

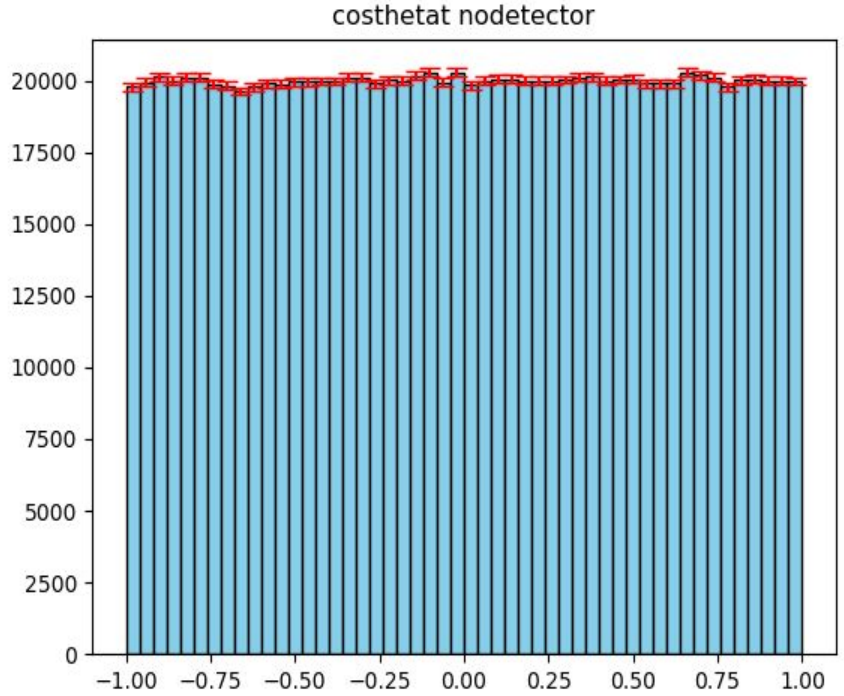
Can use efficiencies to simply calculation:

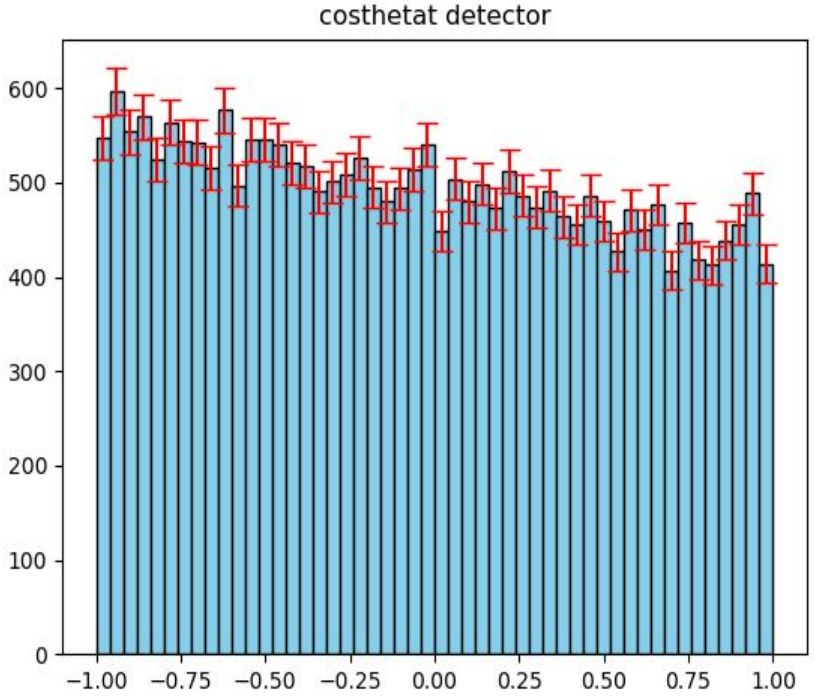
$$\mathrm{efficiency}_i = \frac{n_{i_{\mathrm{det}}}}{n_{i_{\mathrm{nodet}}}}$$

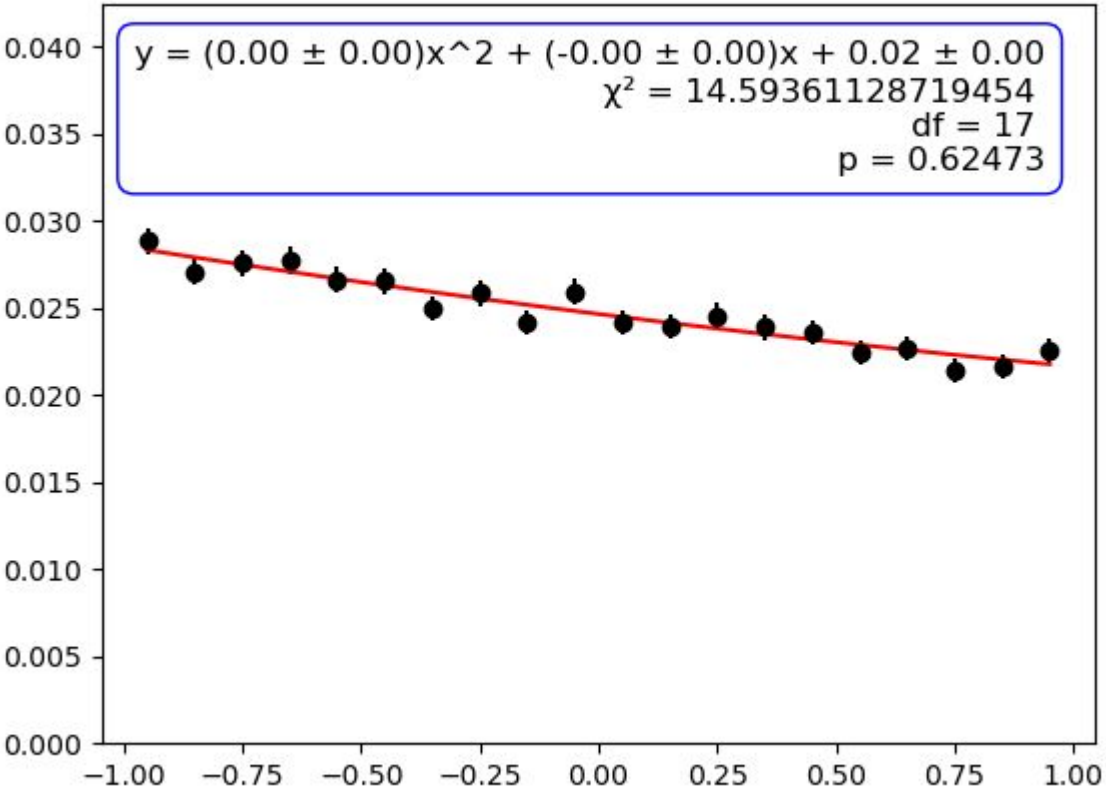

costhetab nodetector
20000
17500
15000
12500
10000
7500
5000
2500
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

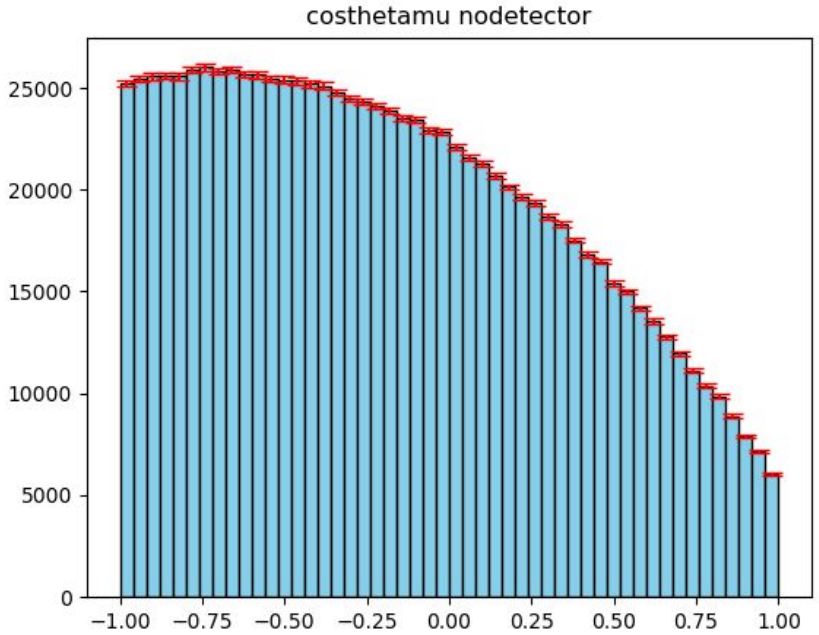

costhetab detector
600
500
400
300
200
100
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

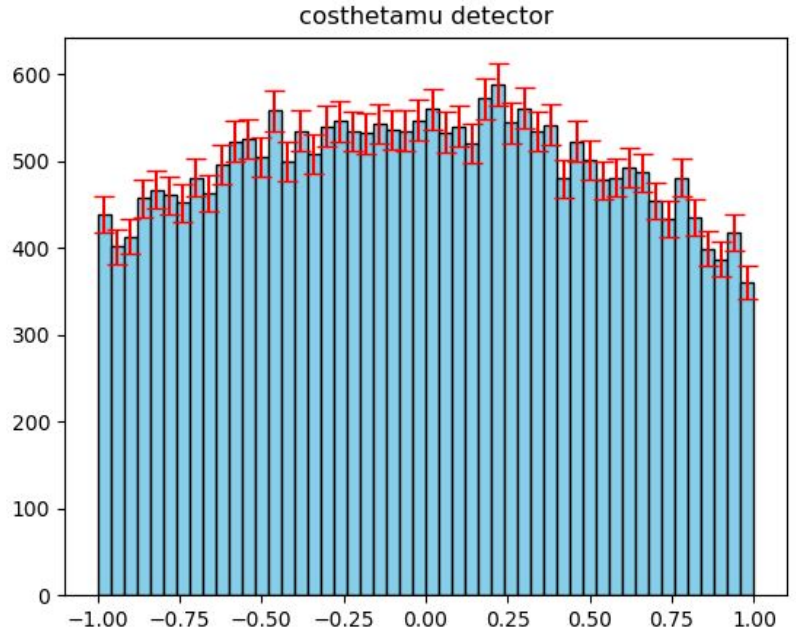

y = (-0.00 ± 0.00)x^2 + (0.00 ± 0.00)x + 0.02 ± 0.00
χ² = 17.828992705639745
df = 17
p = 0.39971
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


costhetat nodetector
20000
17500
15000
12500
10000
7500
5000
2500
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

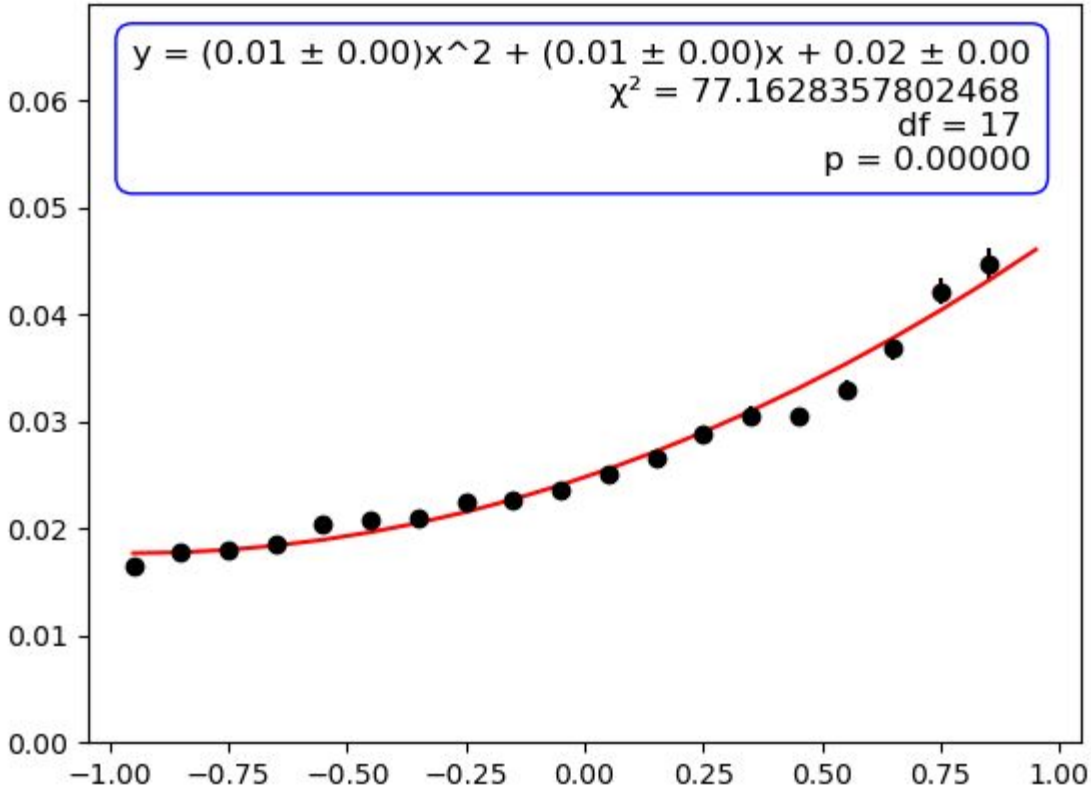

costhetat detector
600
500
400
300
200
100
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

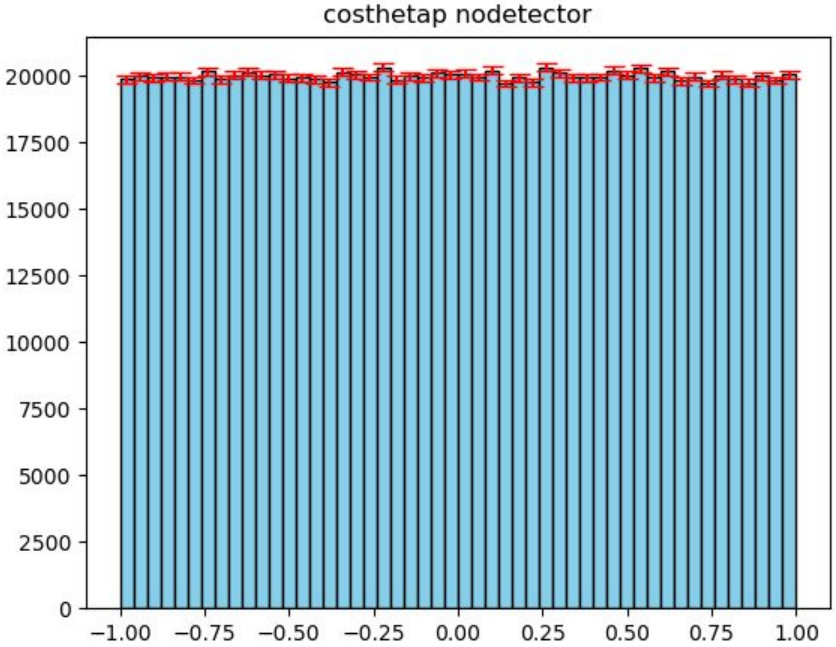

y = (0.00 ± 0.00)x^2 + (-0.00 ± 0.00)x + 0.02 ± 0.00
χ² = 14.59361128719454
df = 17
p = 0.62473
0.040
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

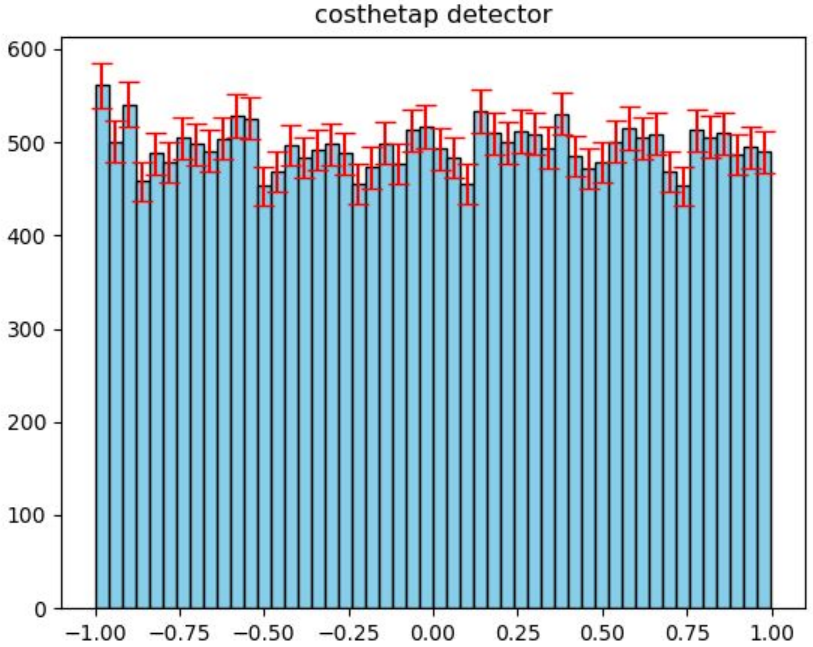


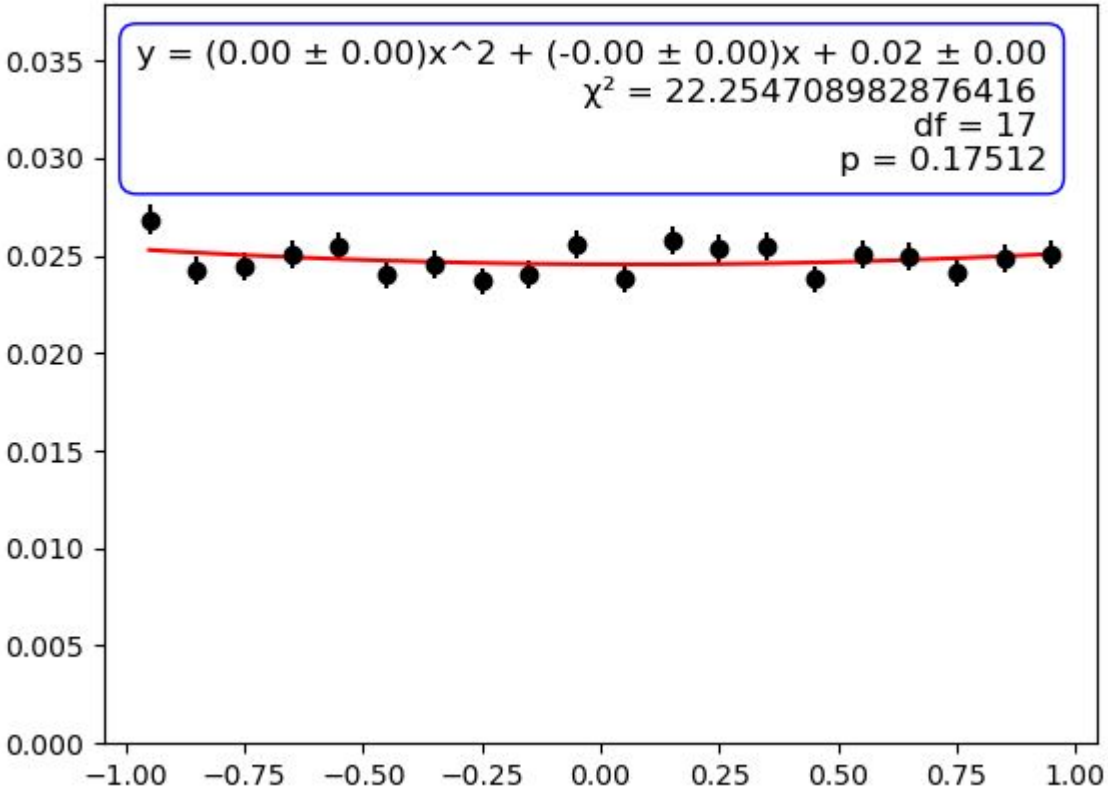


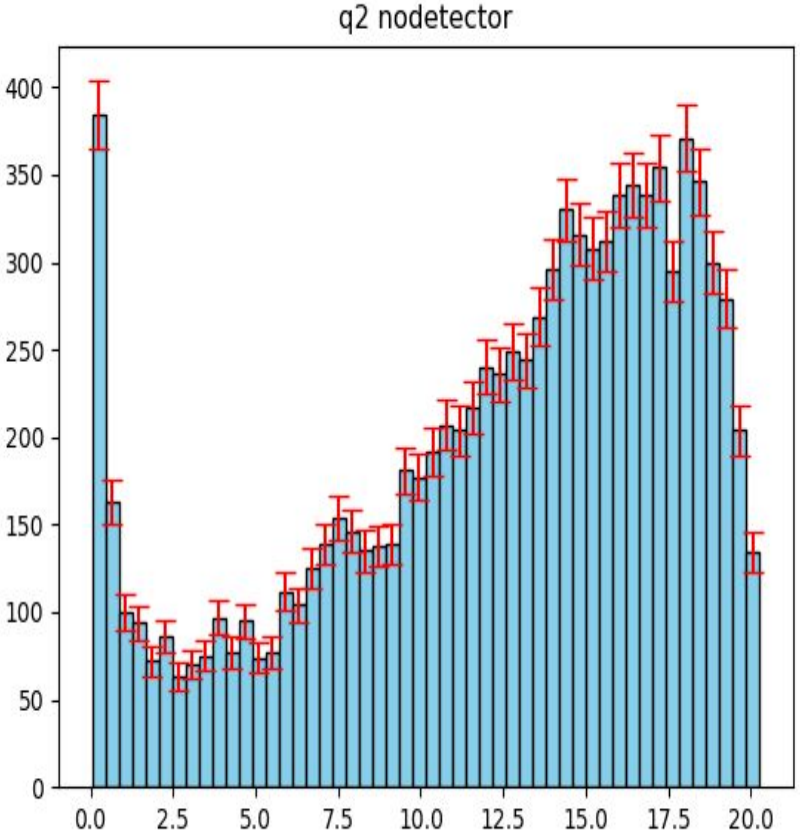
- Asymmetry possibly due to how muons are selected from background in FCC simulator - still needs work

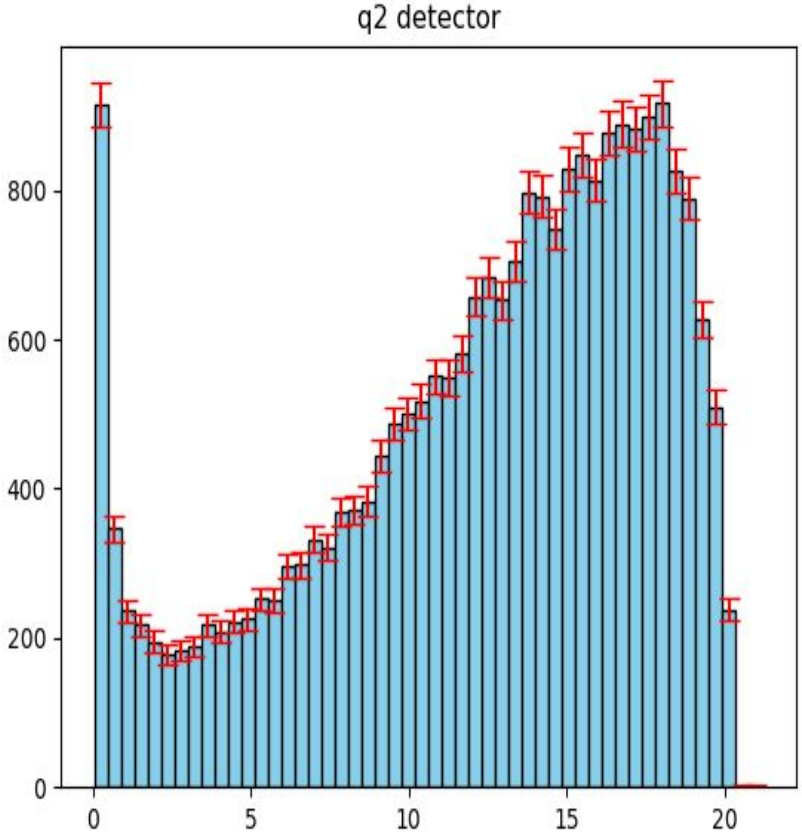

y = (0.01 ± 0.00)x^2 + (0.01 ± 0.00)x + 0.02 ± 0.00
χ² = 77.1628357802468
df = 17
p = 0.00000
0.06
0.05
0.04
0.03
0.02
0.01
0.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


costhetap nodetector
20000
17500
15000
12500
10000
7500
5000
2500
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


costhetap detector
600
500
400
300
200
100
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

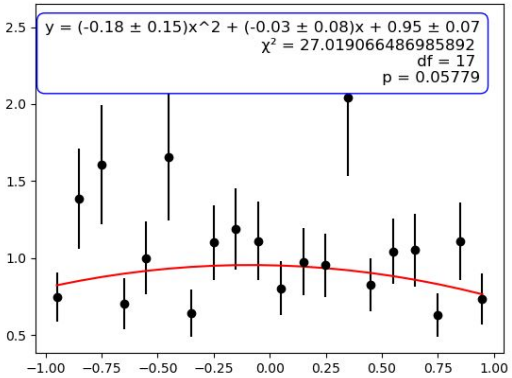

y = (0.00 ± 0.00)x^2 + (-0.00 ± 0.00)x + 0.02 ± 0.00
χ² = 22.254708982876416
df = 17
p = 0.17512
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

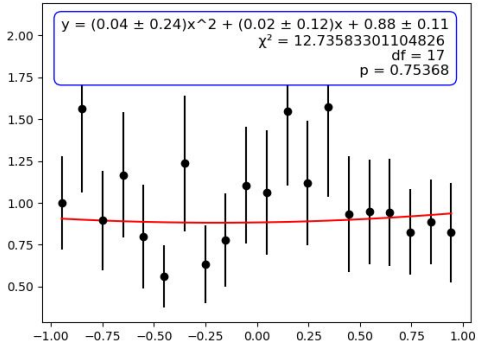


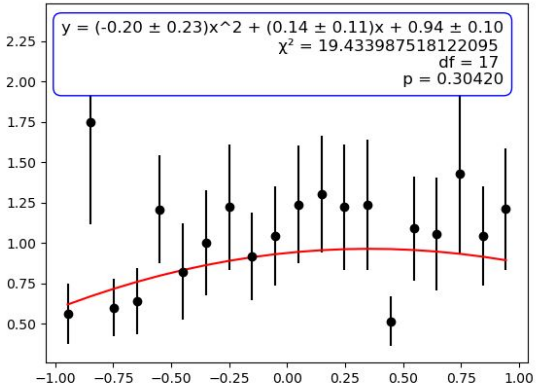


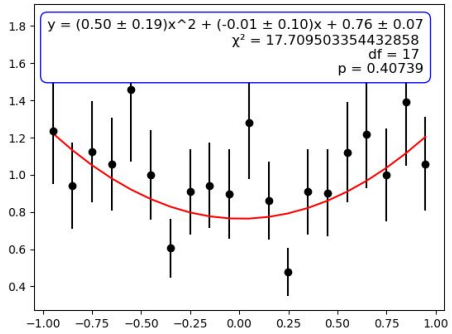
Shape comes from different Feynman diagrams at different q2

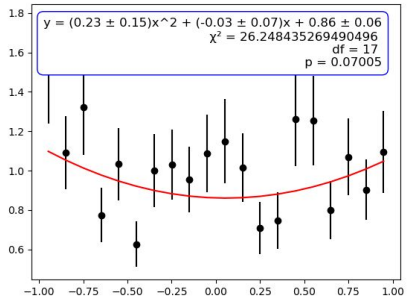
q2 Binning:

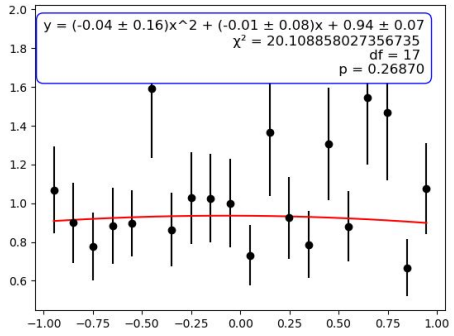

y = (-0.18 ± 0.15)x^2 + (-0.03 ± 0.08)x + 0.95 ± 0.07
χ² = 27.019066486985892
df = 17
p = 0.05779


y = (0.04 ± 0.24)x^2 + (0.02 ± 0.12)x + 0.88 ± 0.11
χ² = 12.73583301104826
df = 17
p = 0.75368

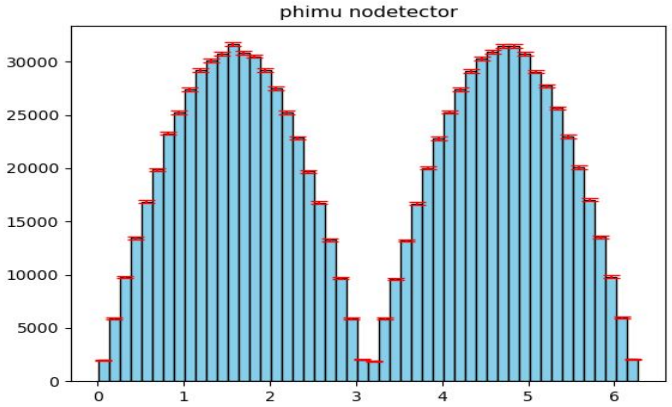

y = (-0.20 ± 0.23)x^2 + (0.14 ± 0.11)x + 0.94 ± 0.10
χ² = 19.433987518122095
df = 17
p = 0.30420

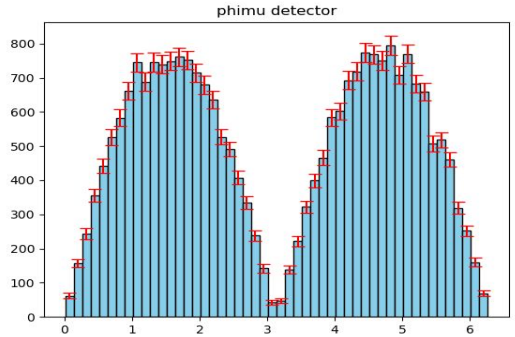

y = (0.50 ± 0.19)x^2 + (-0.01 ± 0.10)x + 0.76 ± 0.07
χ² = 17.709503354432858
df = 17
p = 0.40739

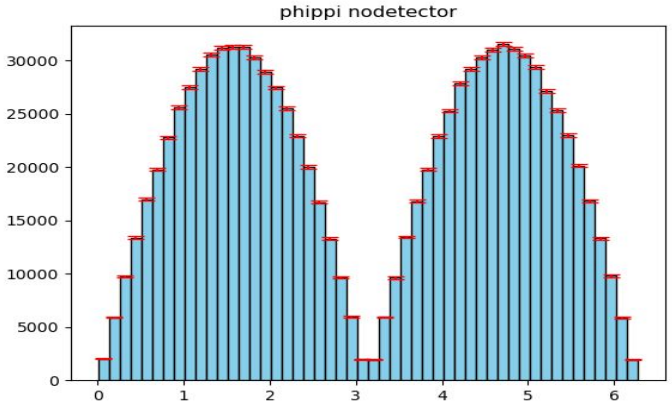

y = (0.23 ± 0.15)x^2 + (-0.03 ± 0.07)x + 0.86 ± 0.06
χ² = 26.248435269490496
df = 17
p = 0.07005

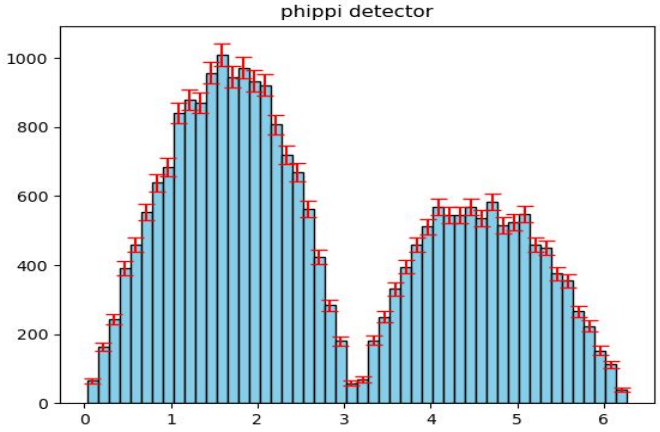

y = (-0.04 ± 0.16)x^2 + (-0.01 ± 0.08)x + 0.94 ± 0.07
χ² = 20.108858027356735
df = 17
p = 0.26870

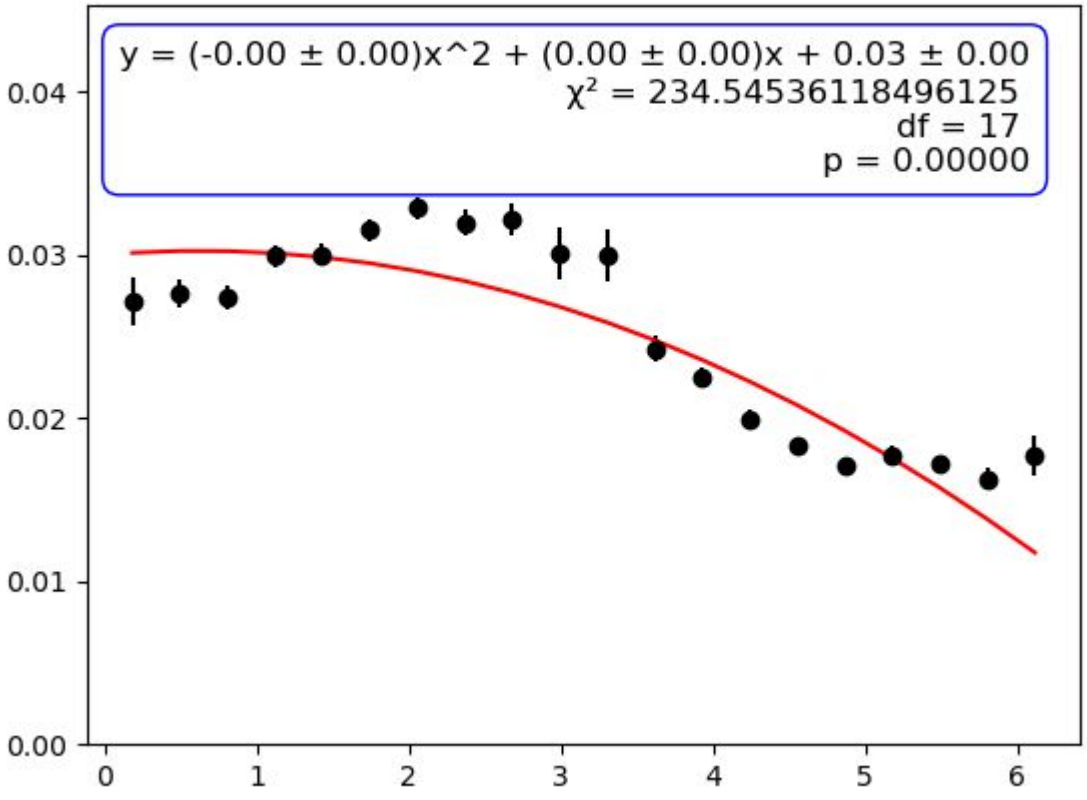
Issue with phi angles:



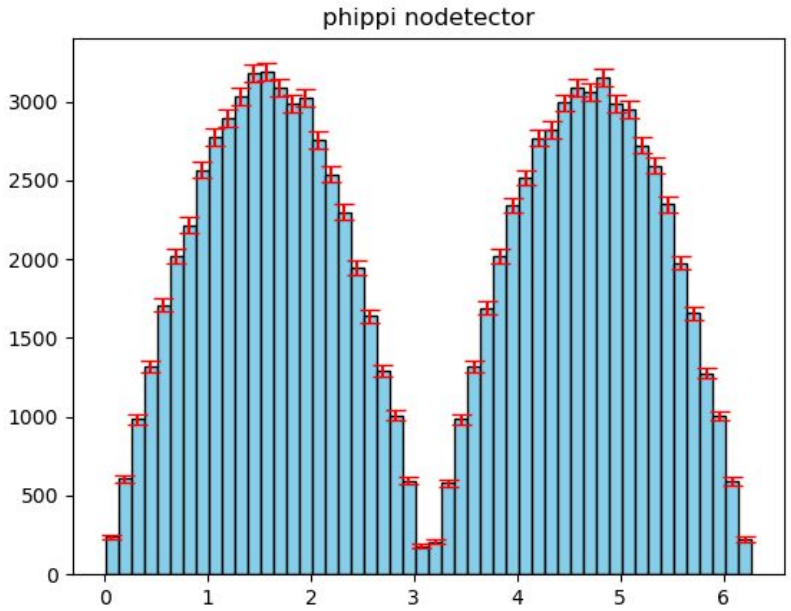


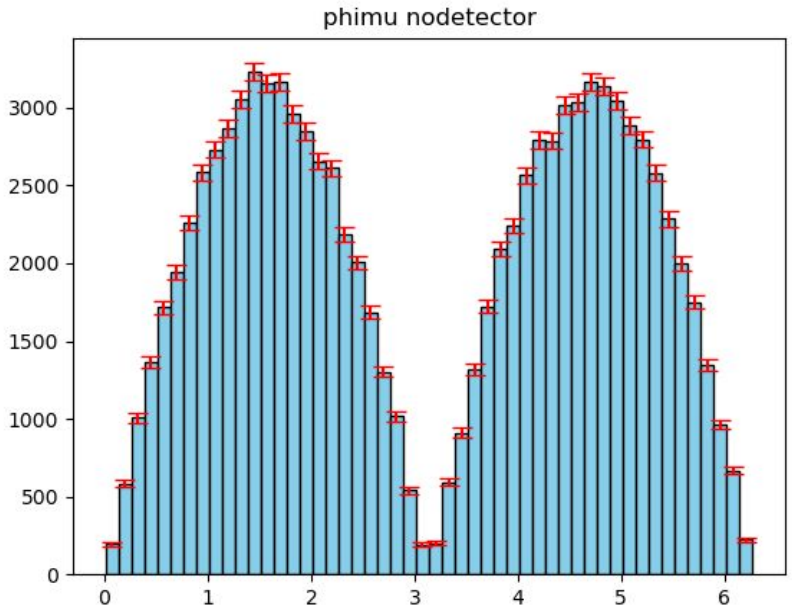




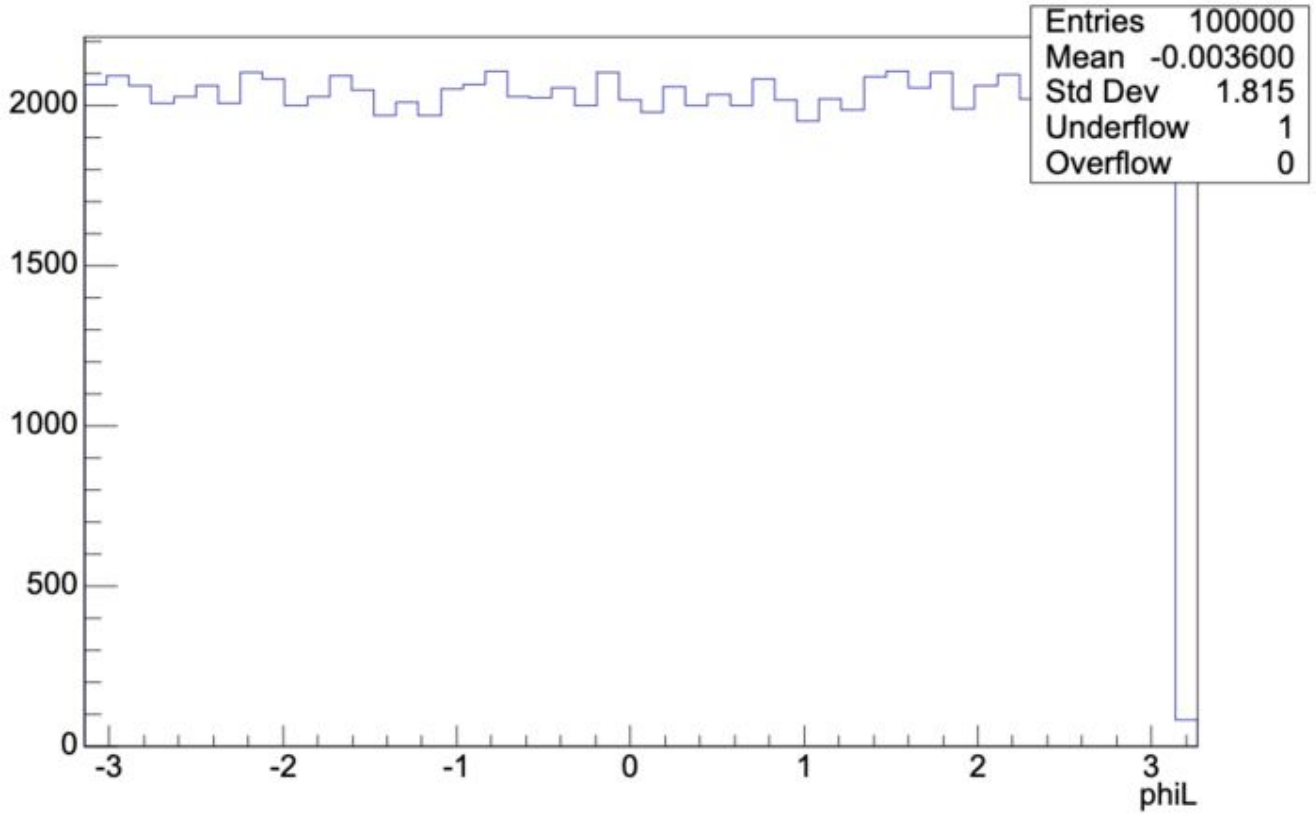

y = (-0.00 ± 0.00)x^2 + (0.00 ± 0.00)x + 0.03 ± 0.00
χ² = 234.54536118496125
df = 17
p = 0.00000
0.04
0.03
0.02
0.01
0.00
0
1
2
3
4
5
6

Ran phi angles over phase space simulation:

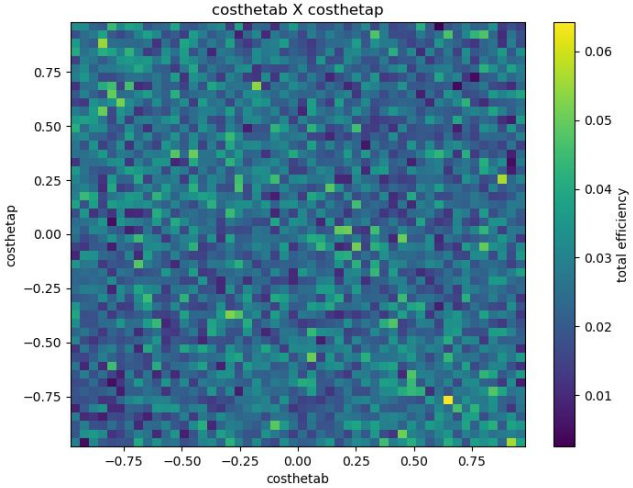


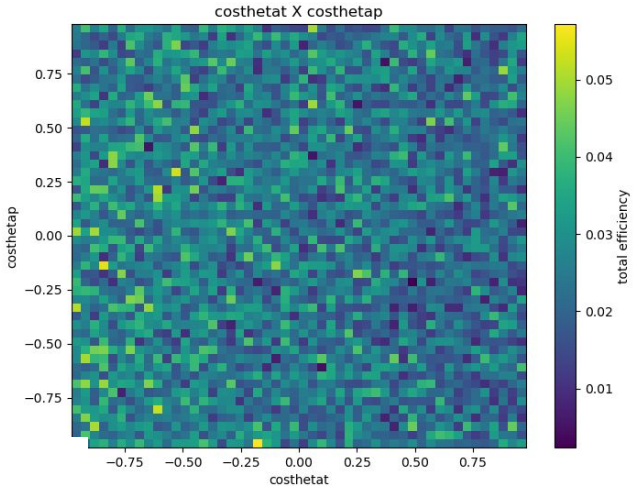


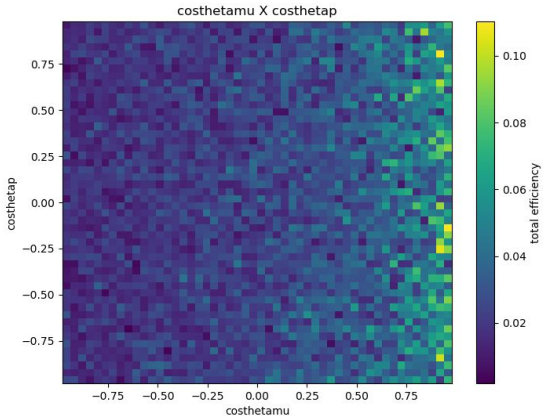
Using C++ script:



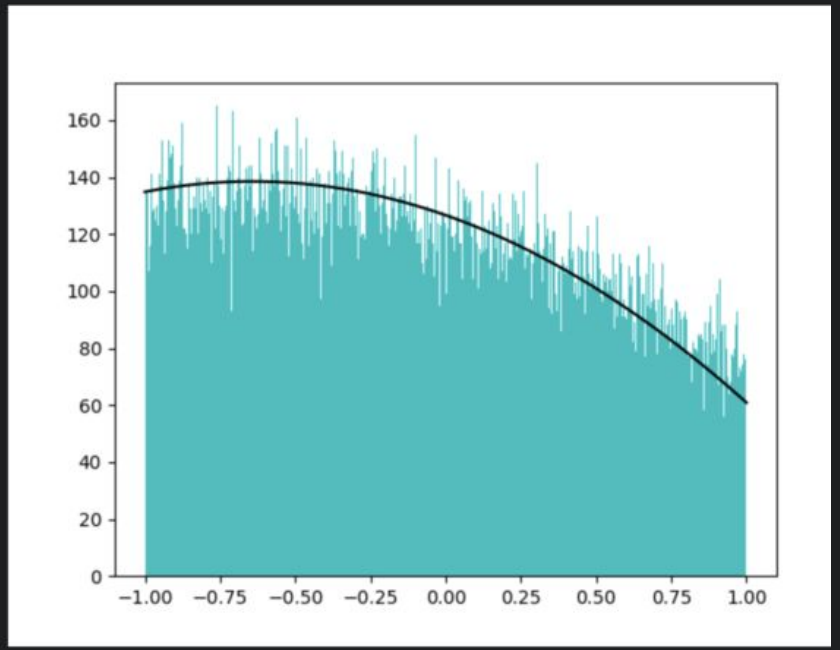
2D efficiencies:

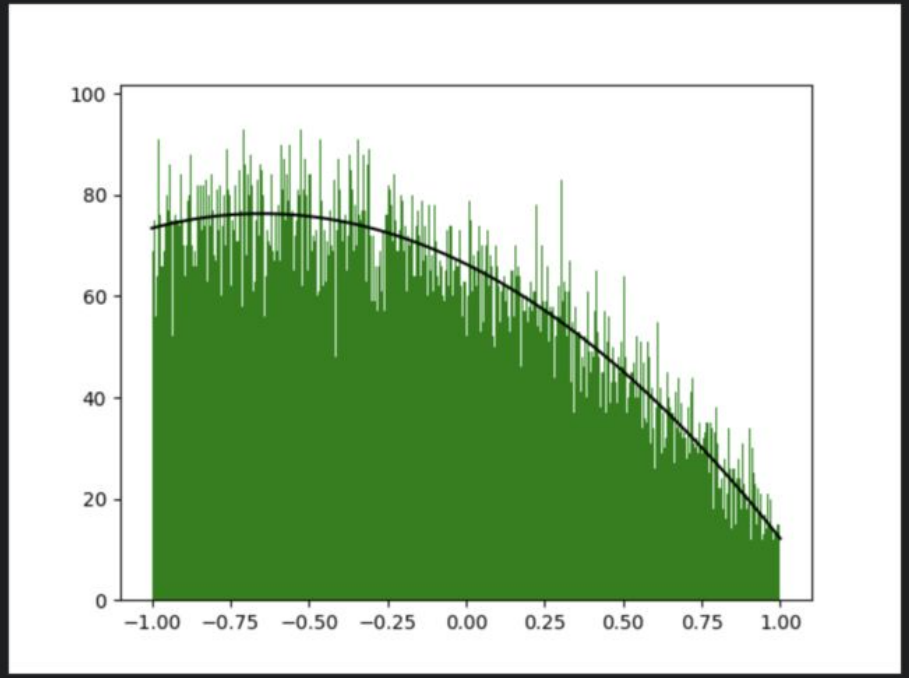




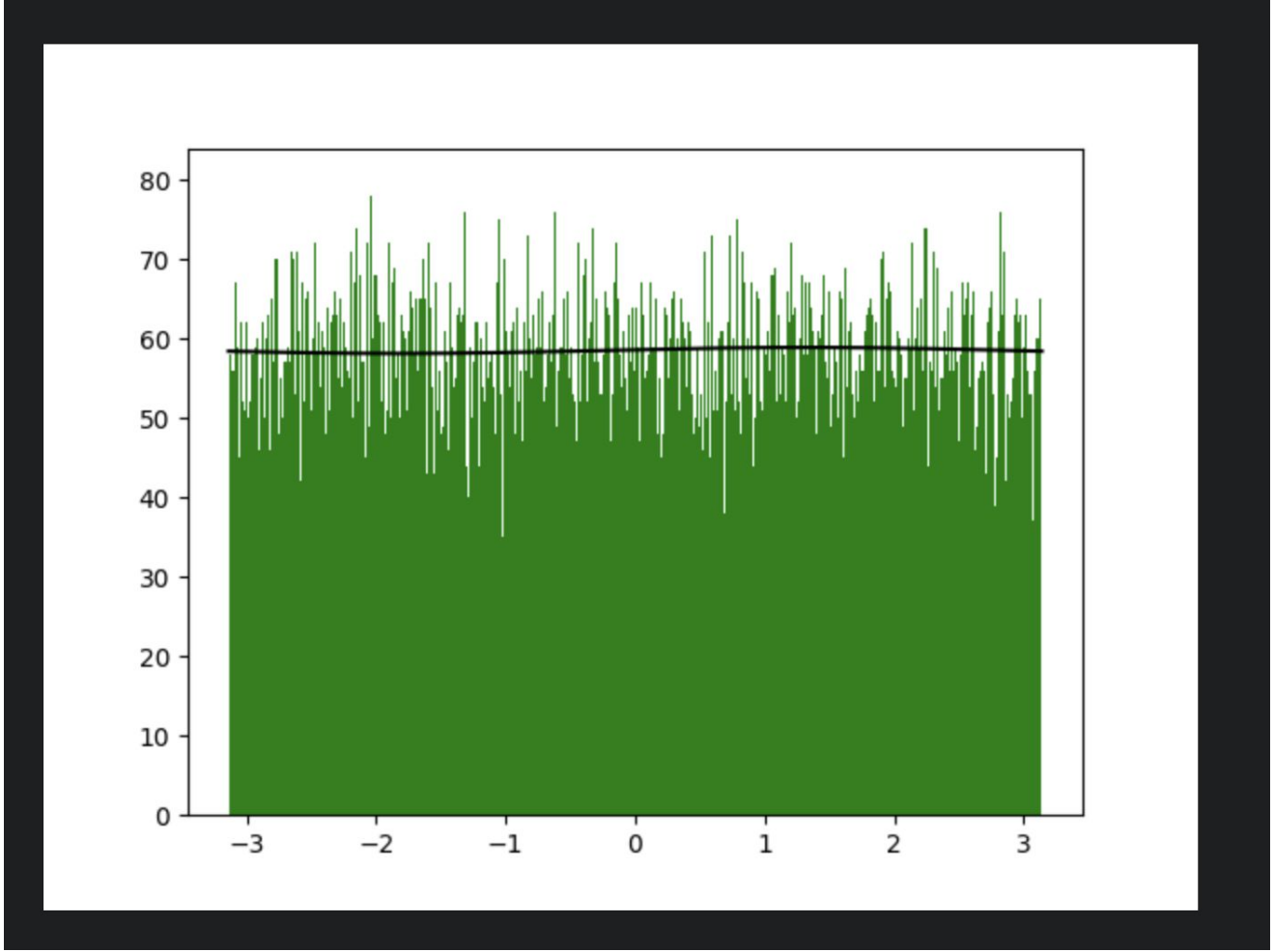


Mero's Work: Fitting angles





cos(thetamu) (ignoring efficiencies)



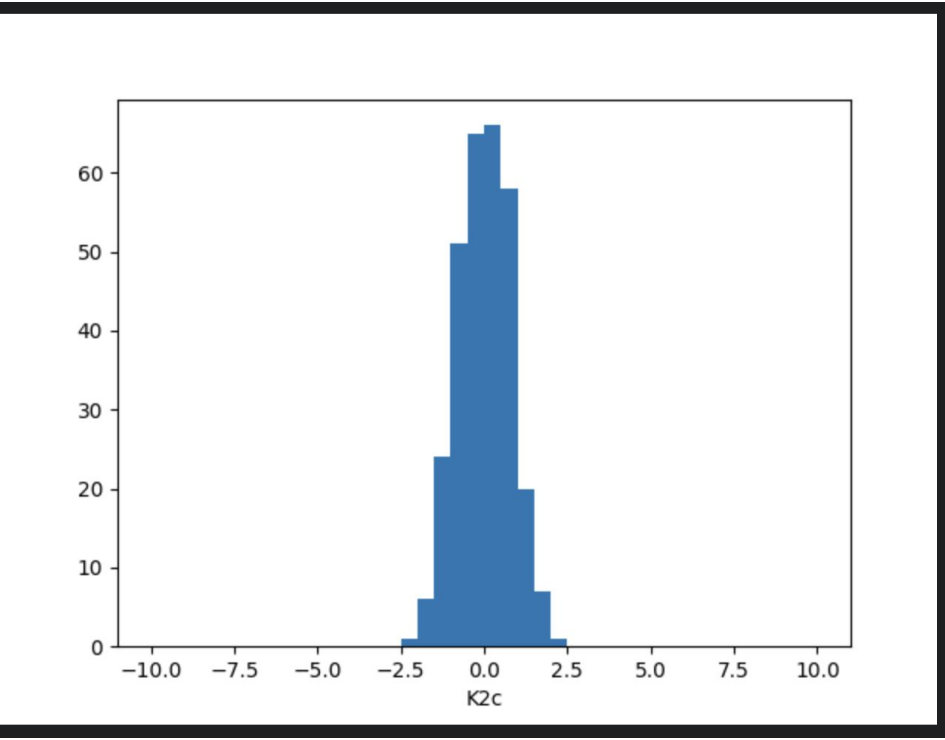
phi

- Fit parameters are related to the Lambda b polarization
- Check validity of PDF by taking pulls for each parameter:

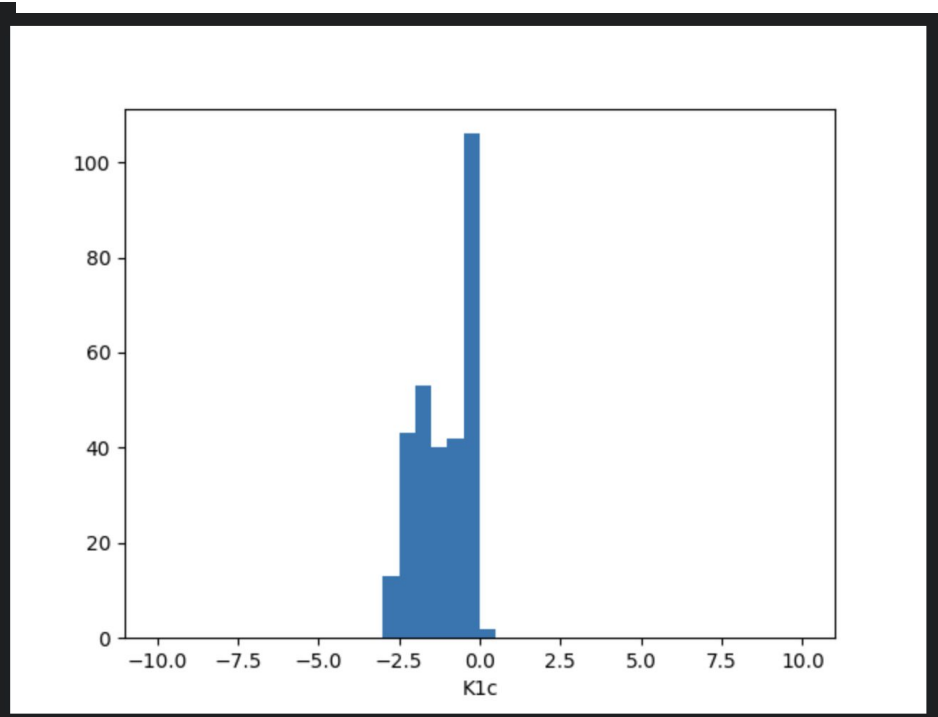
$$\text{pull} = \frac{\mu_{\text{total}} - \mu_{\text{sample}}}{\sigma_{\text{sample}}}$$

- Expect gaussian pulls with mean=0, sigma=1

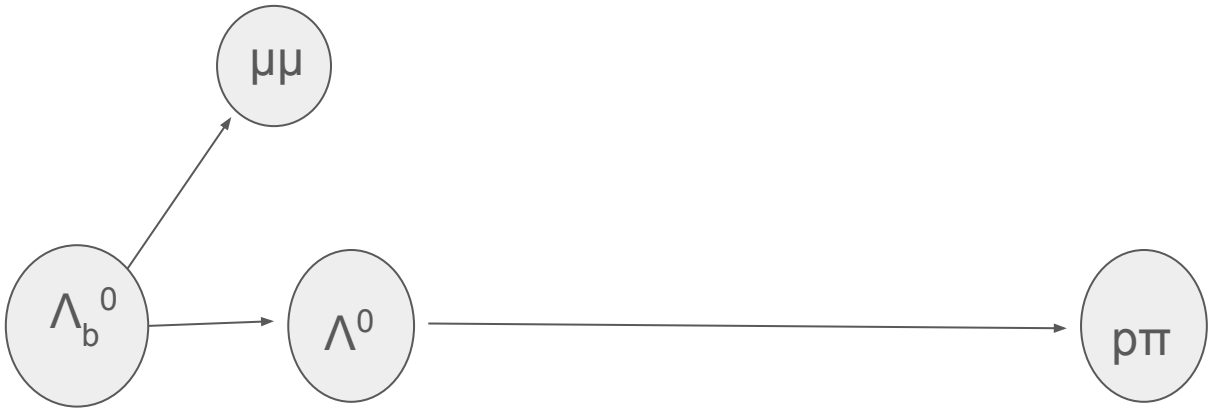
Biased pulls in some parameters:



Good pull example



Bad pull example (maybe not randomizing initial parameter)

Whats next:

- Biased pulls could be a result of how the ROOT files are generated
- Right now the code is using particle ID, but this is unrealistic
- Instead, use geometry of the decay: