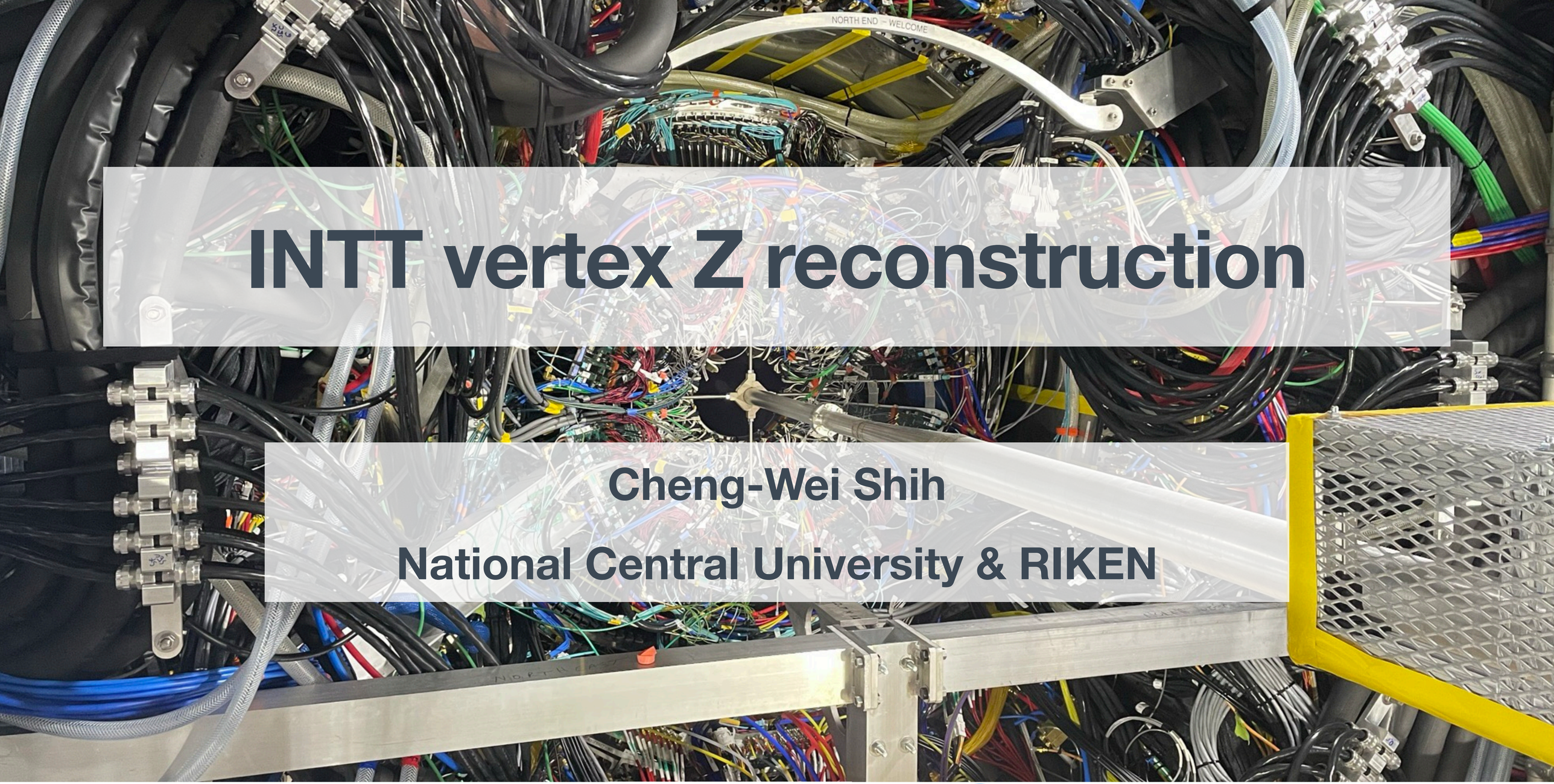


INTT vertex Z reconstruction

Cheng-Wei Shih

National Central University & RIKEN

June 16th, 2024
INTT meeting

sPHENIX INTT sPHENIX 國立中央大學 National Central University RIKEN

Vertex Z - procedures

1. Update the cluster ϕ and radius based on the reconstructed average vertex XY

2. Loop over all the inner and outer cluster pairs, keep the pairs which pass the Δϕ cut and DCA cut (proto-tracklets)

3. Move to the Z-Radius plane. For each proto-tracklets, the full strip lengths of the two clusters (inner and outer) are considered. Project the **possible vertex Z*** to the Z axis (radius = 0)

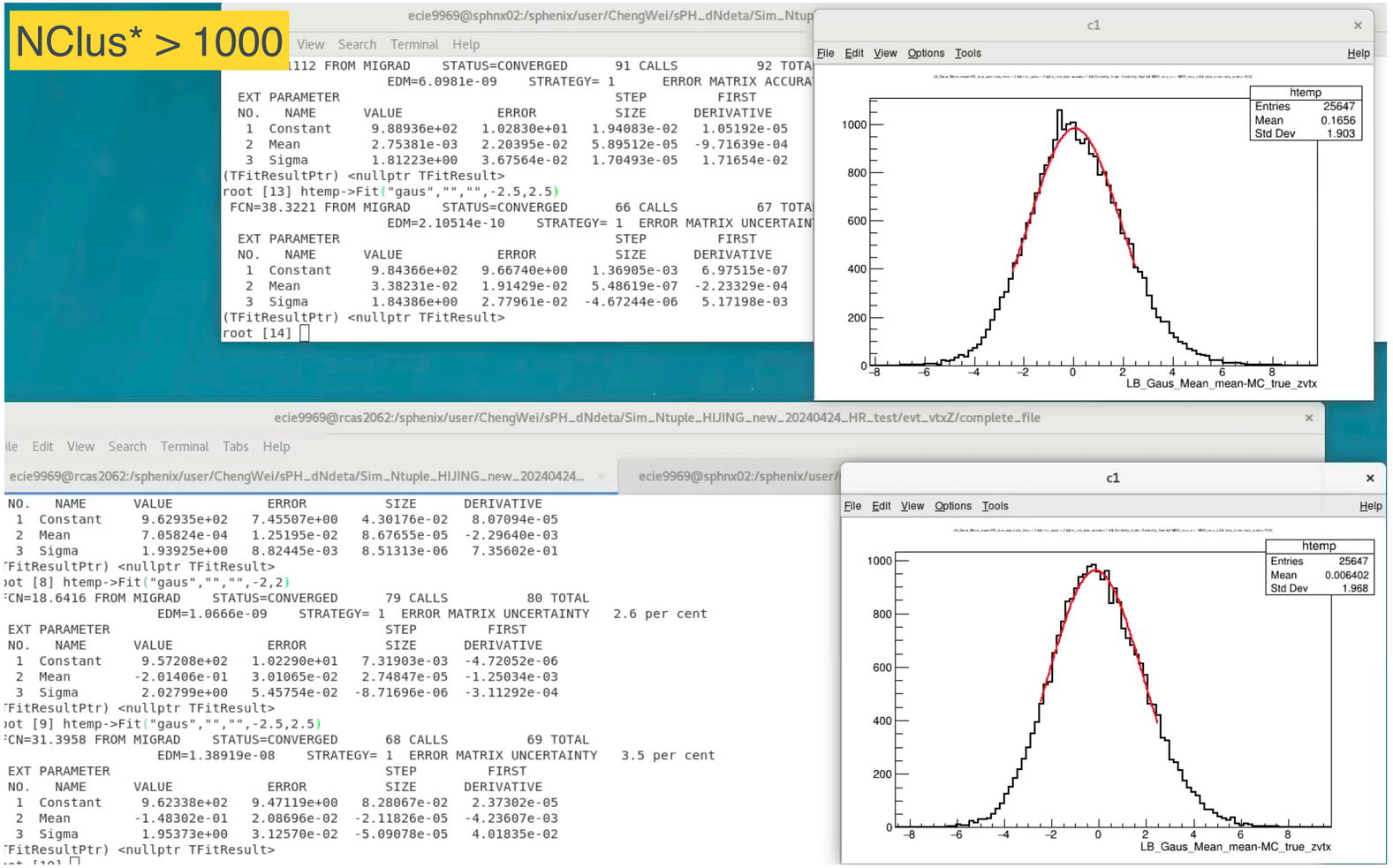
4. Fill all the **possible vertex Zs*** of all the proto-tracklets into the histogram

5. Fit the distribution with **Gaussian function + offset** to determine the reconstructed vertex Z for single event (fit parameter and fit range tunable)

Possible vertex Z*: the shapes are different based on the methods

Vertex Z improvement

- Suggestion given by Akiba san
- Idea: the weight of each combination filled into the 1D histogram should be the same, i.e., **weighting the entry by (1./"possible vertex Z range")**



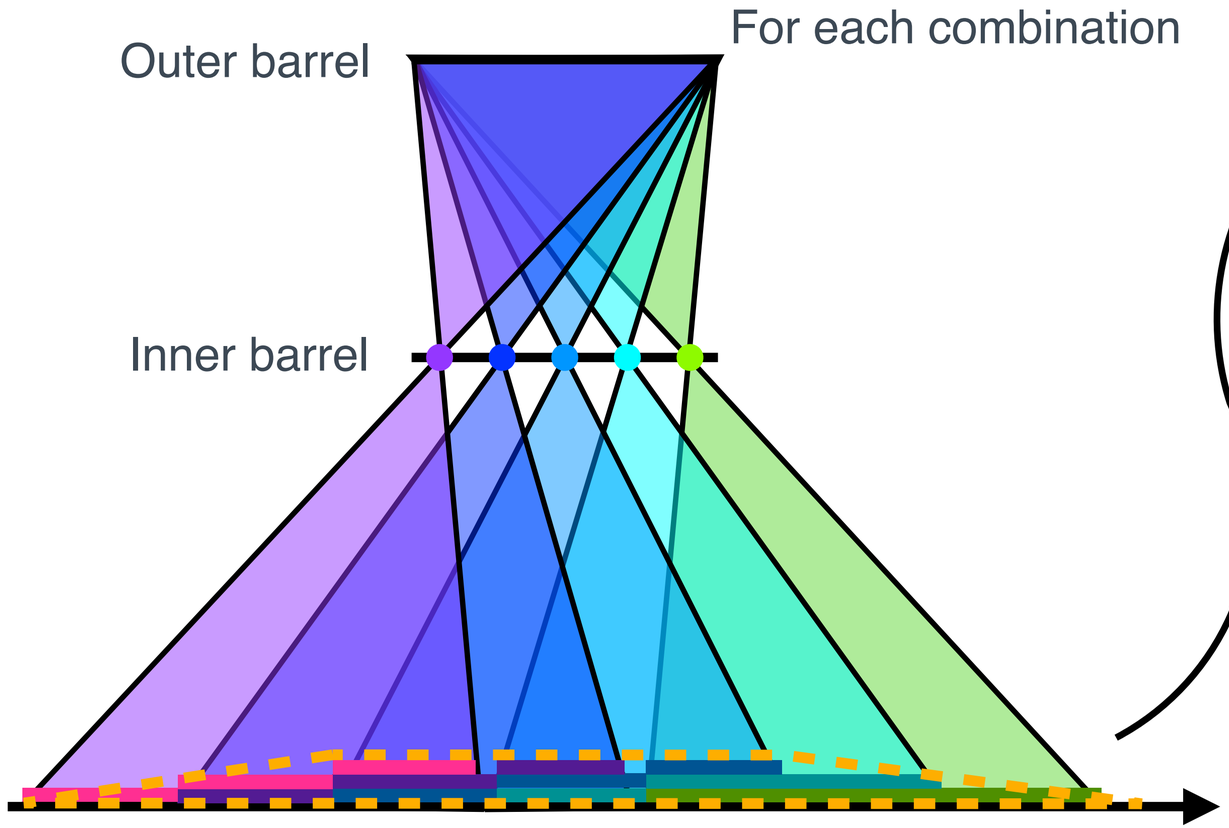
Entries weighted
Fit mean : 3.382e-02
Fit width : 1.8439e+00

Original
Fit mean : -1.483e-01
Fit width : 1.9537e+00

~6% improvements

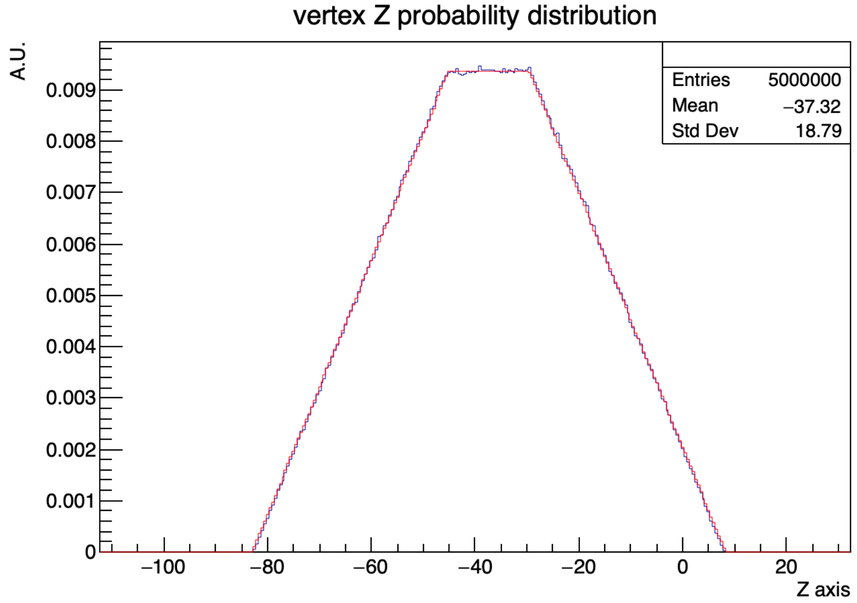
Vertex Z improvement - trapezoidal

- Idea given by Akiba san. For each combination, take into account of the distribution of the possible vertex Z range, and normalize the distribution, and fill into the histogram. (Used to assume the Uniform distribution of the vertex Z)
- Justification of using **possible vertex Z range / sqrt(12)**: The region with higher probability is presumably selected



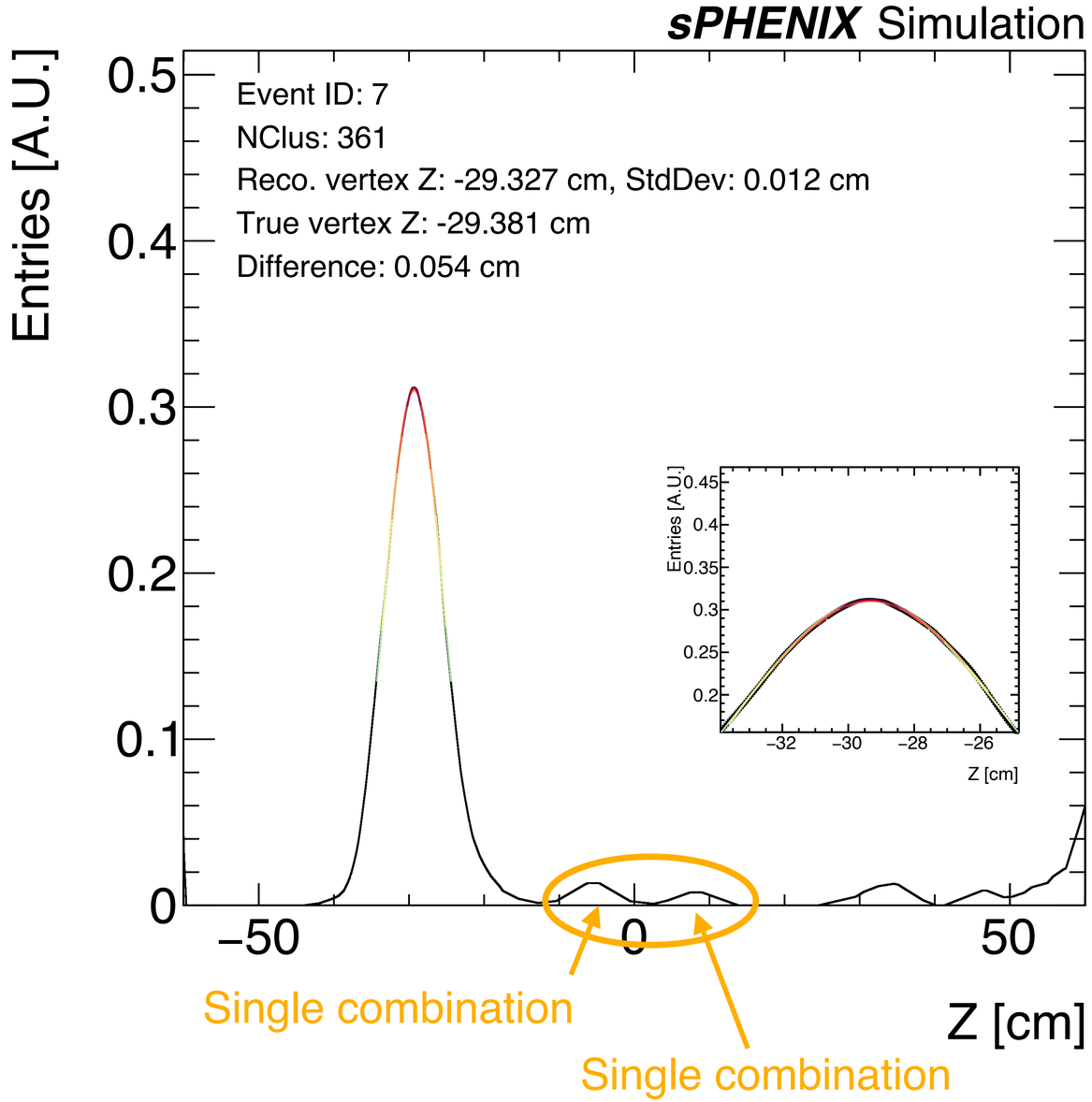
For each combination

Trapezoidal shape for each combination



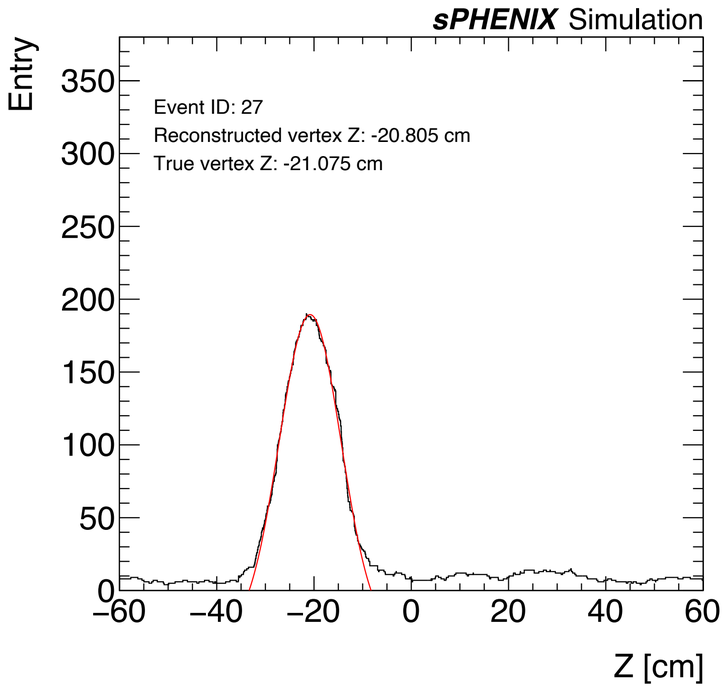
Caveat: for each combination in single event, have to have the shape, and fill that into histogram, not trivial…
Not easy, but still have the way to make it happen

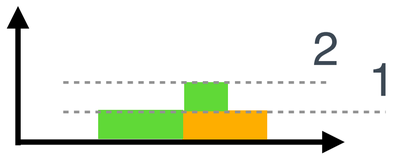
Vertex Z improvement - trapezoidal



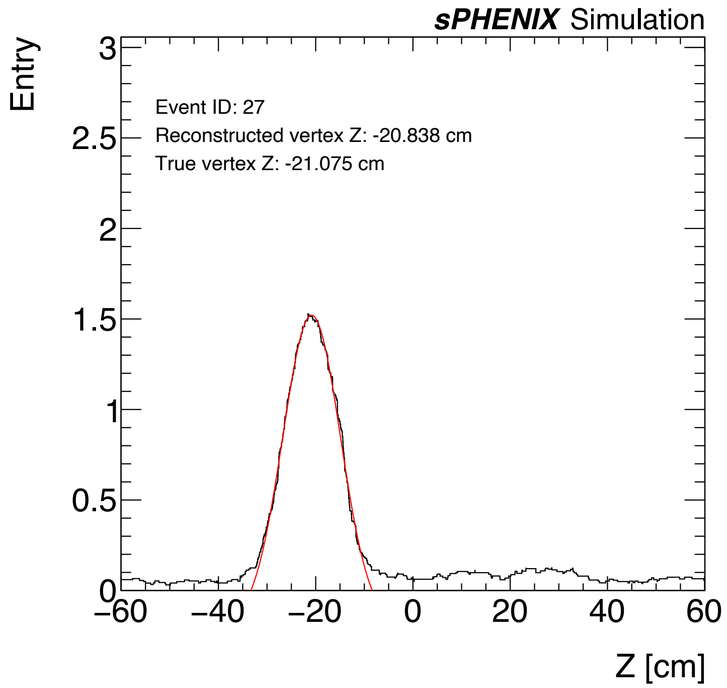
Vertex Z improvement

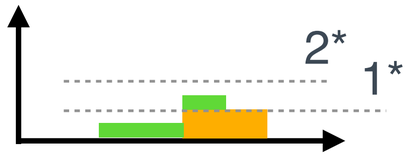
Original





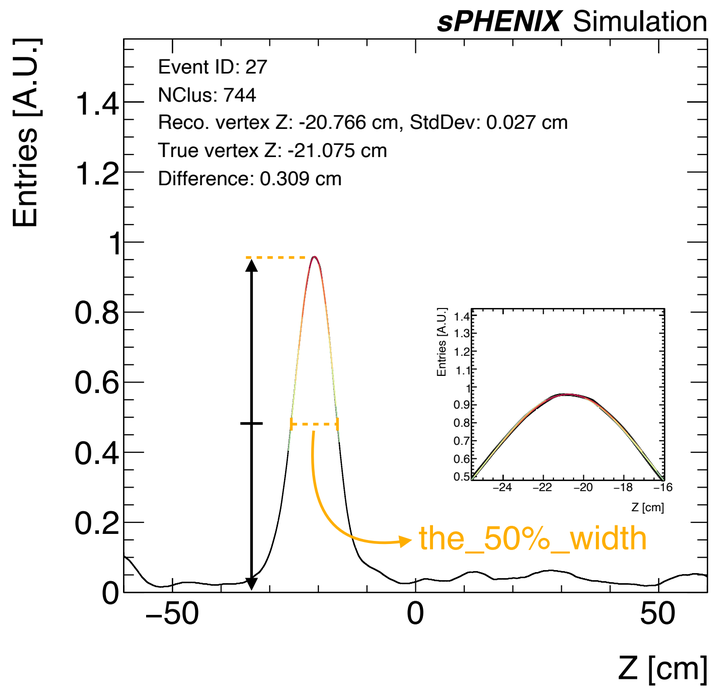
Entries weighted

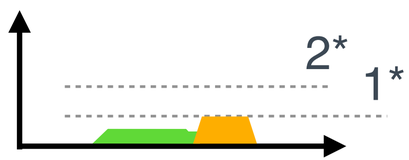




Trapezoidal

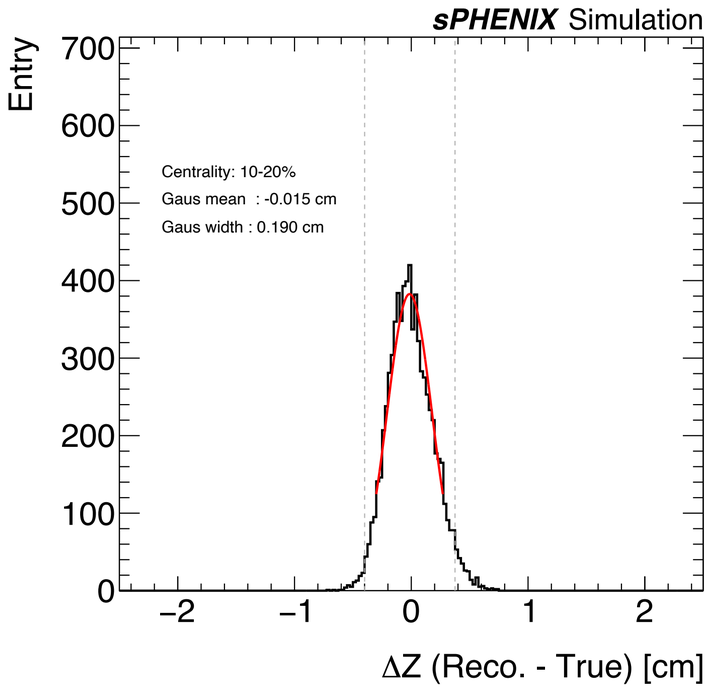
(final vertex Z given by average of 7 gaussian fits with the fit ranges of "mean ±(0.2 + 0.15 x i) x **the_50%_width**")

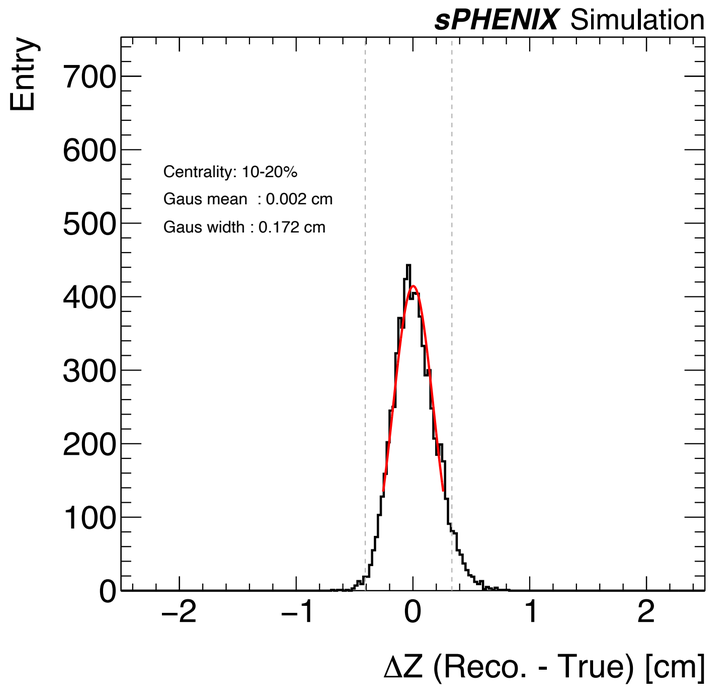




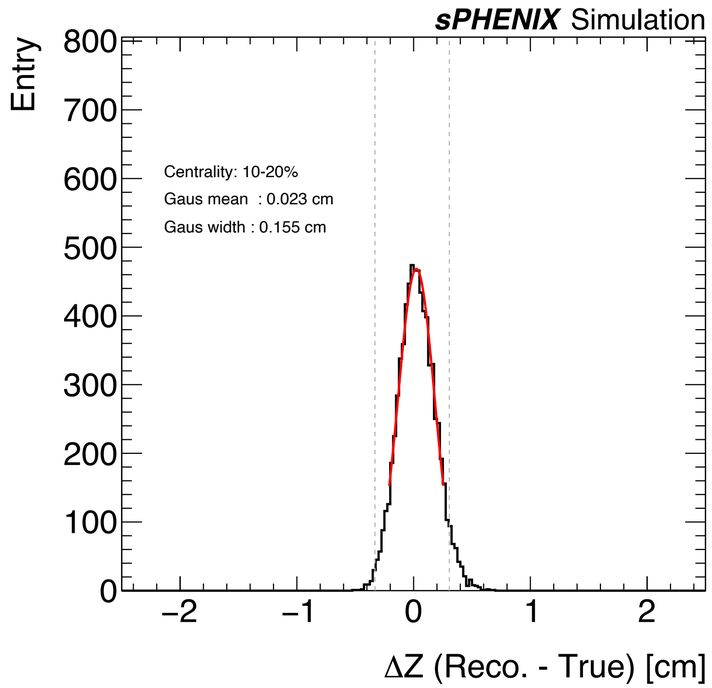
Vertex Z improvement

Very central events





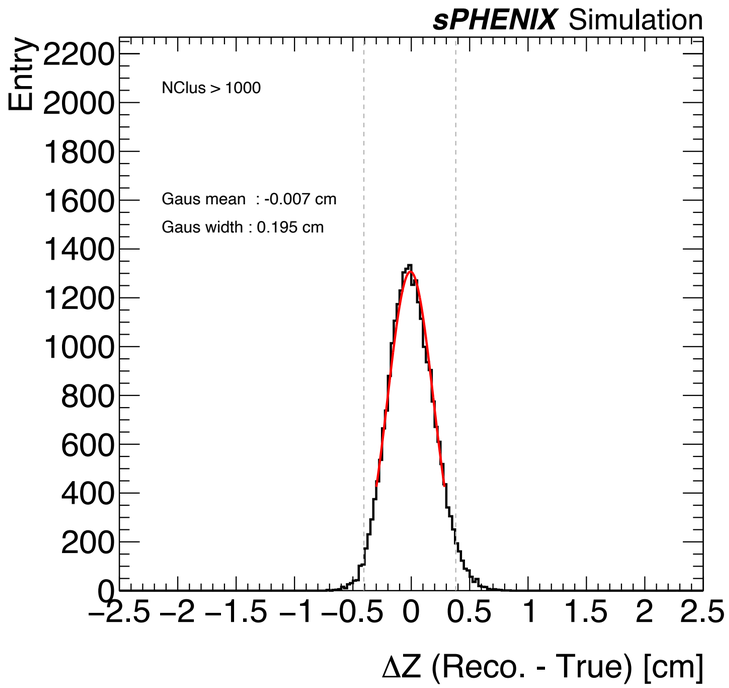
9.5% improvement

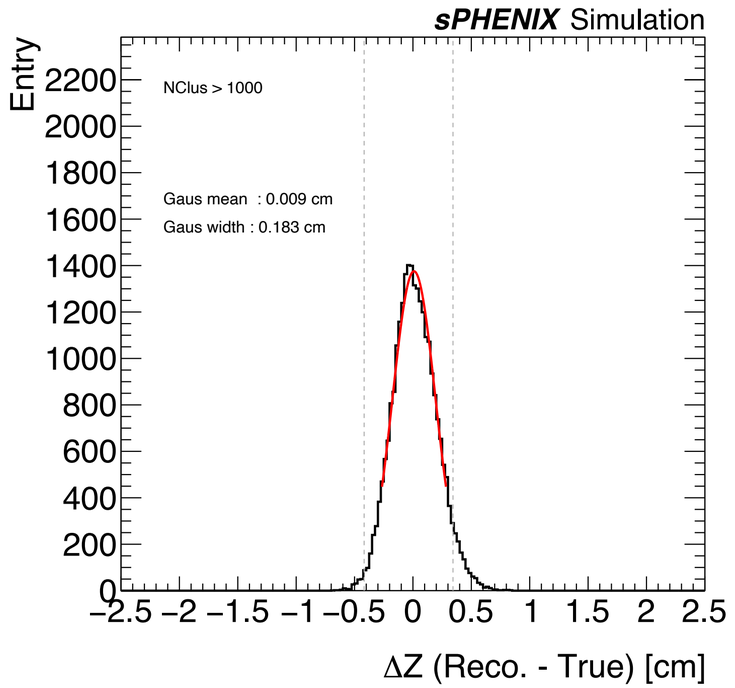


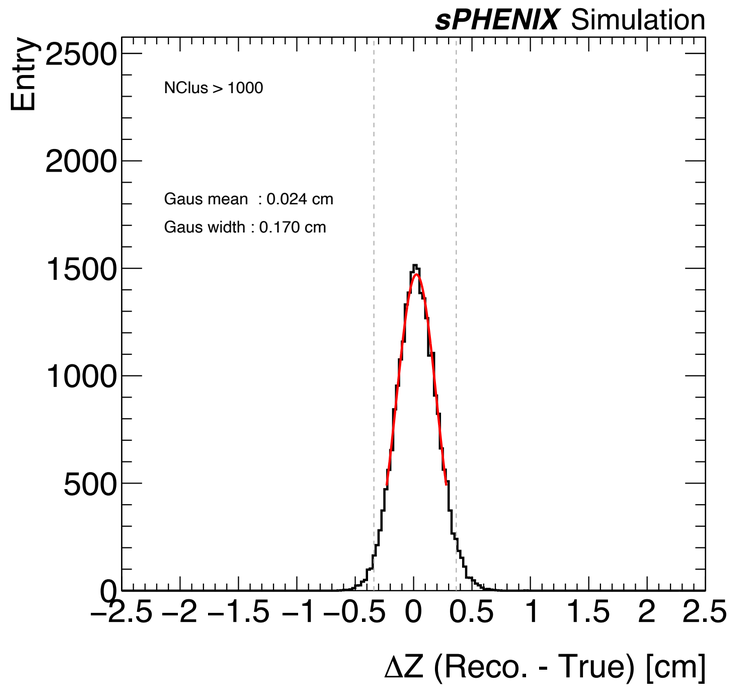
18.4% improvement

Vertex Z improvement

Number of valid clusters > 1000







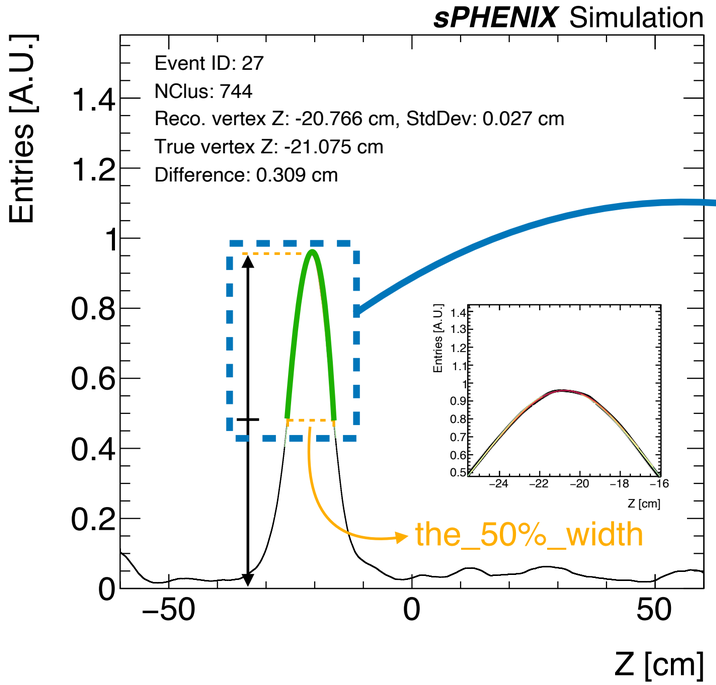
6.2% improvement

12.8% improvement

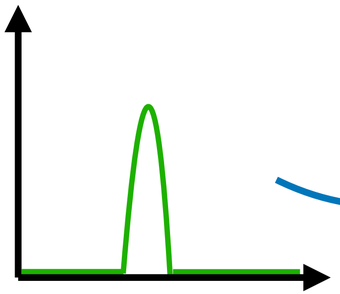
Valid clusters: cluster adc > 35 && cluster phi size < 6

Determining the vertex Z from the histogram

- New trial: after having the histograms made of possible vertex Z ranges, use ML (XGBoost) to do the final vertex Z determination
- Training variables: the content of each bin of the histogram post the 50% entry cut (2401 variables currently, corresponding to the number of bins of histogram)
- Total MC events: 80k (80% training, 20% testing)



Post applying the bkg cut on the histo. by 50% of the highest bin content (the head structure)



0
0
0
0
0
0…

0.01
0.014
0.02
0.026….

0.015
0.013
0.008…

0
0
0
0

Feed to the ML together with the answer (True vertex Z)

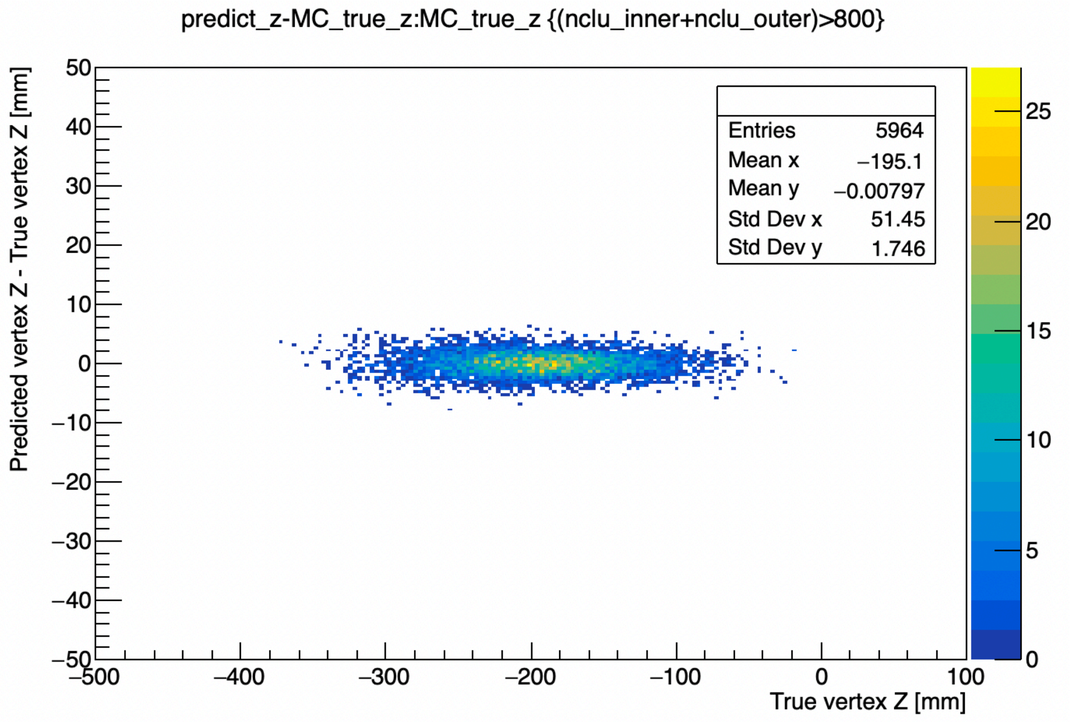
XGBoost

2401 variables for each event

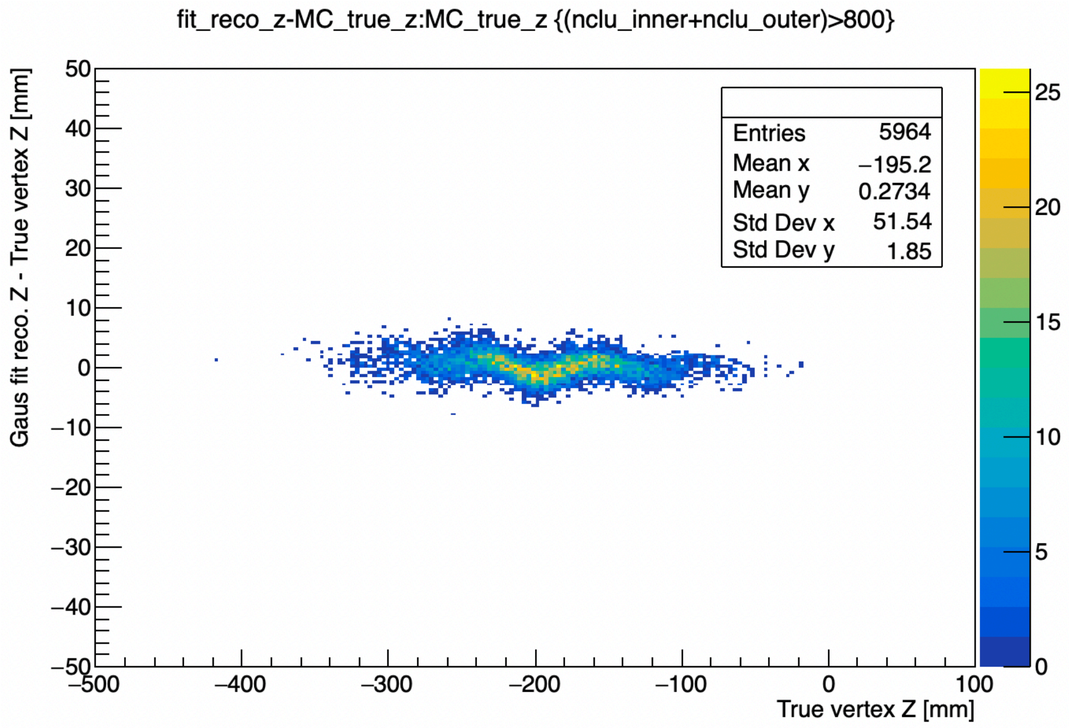
Determining the vertex Z from the histogram

The test sample 20% of the total MC events

Reco. vertex Z predicted by training model



Reco. vertex Z by 7 Gaus fittings



Determining the vertex Z from the histogram

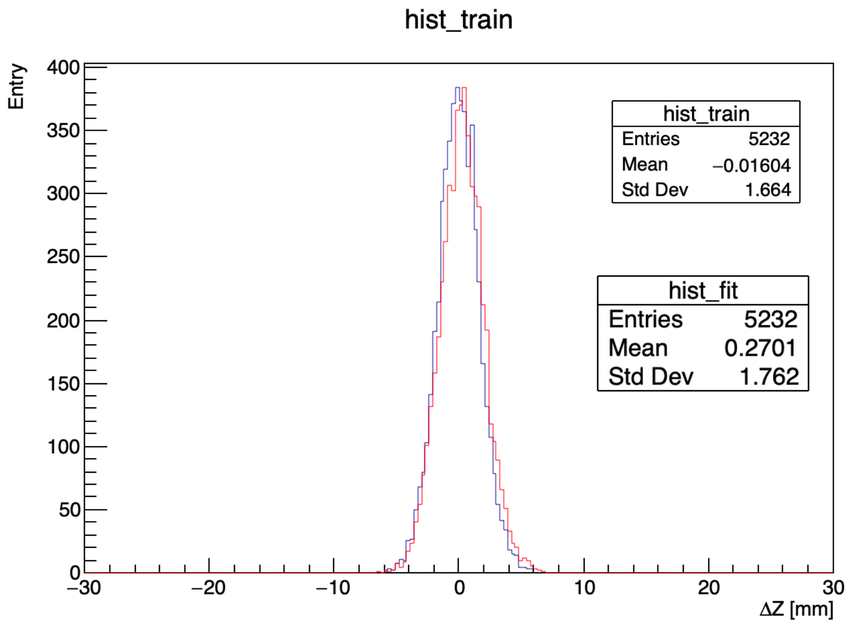
The test sample 20% of the total MC events

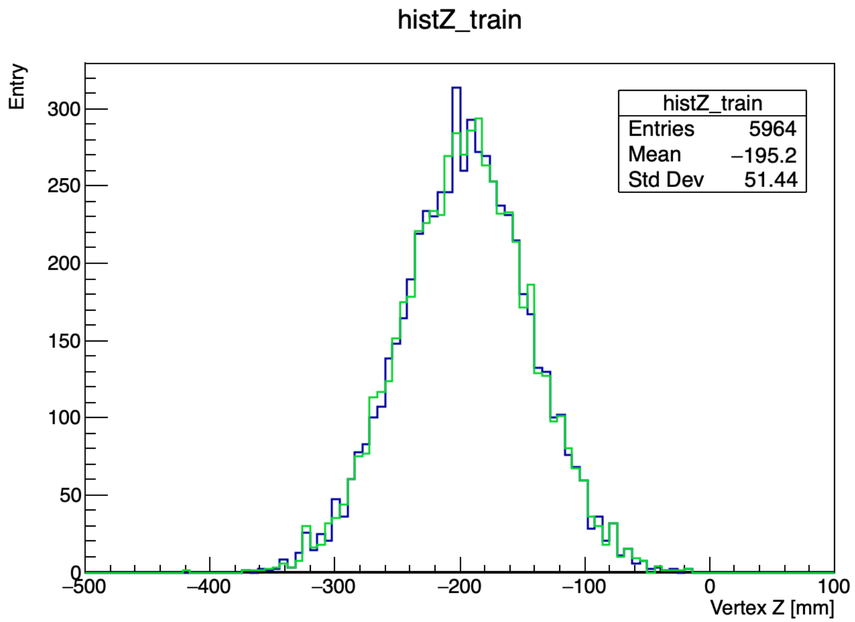
True vertex Z

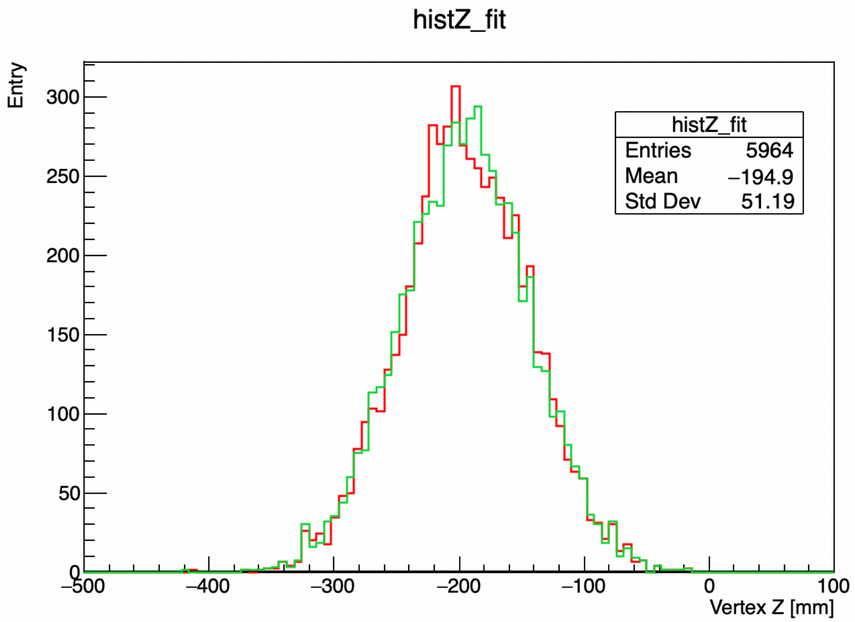
Reco. vertex Z predicted by training model

Reco. vertex Z by 7 Gaus fittings

Number of cluster* > 800







Train StdDev: 1.664

7 Gaus fit StdDev: 1.762

Summary and thoughts

- With the trapezoidal describing the shapes of the possible vertex Z range for each combination and the entry normalization involved, (I think) we are really touching the limit of the vertex Z determination by INTT standalone data, which is good!

- IMHO, the avg_7_Gaus_fit is already very optimal, but nevertheless, just thinking out loud, giving the ML a try

- It seems that the ML does not improve the resolution “that much”, (I think) which indicates that, again, we are touching the limit

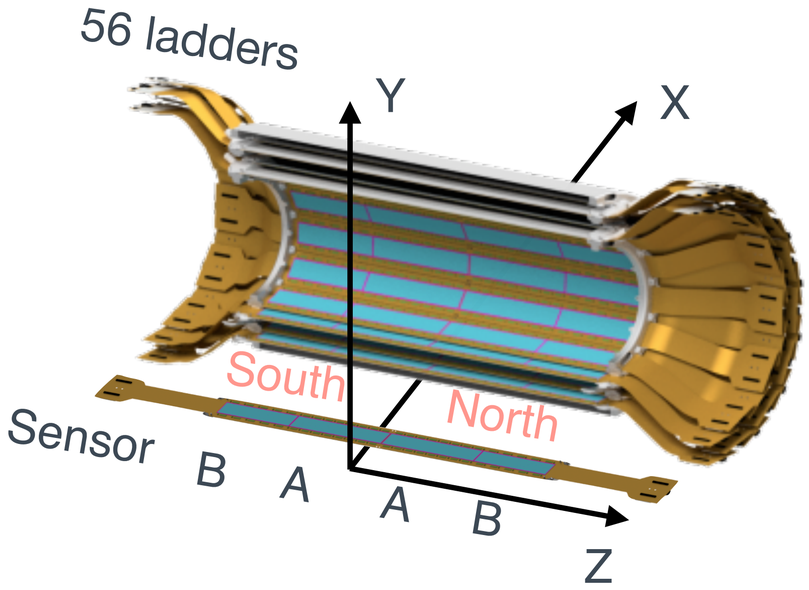
- But ML seems to have the capability to mitigate the wiggling structure at the vertex Z around -20 cm, which makes this approach promising (maybe not the run24 needs to worry about), but more check is needed for sure (retrain with higher statistic, etc)

- Concern: data/MC discrepancy, model overtrain. For data, can possibly (only) be validated by cross check study with MBD vertex Z

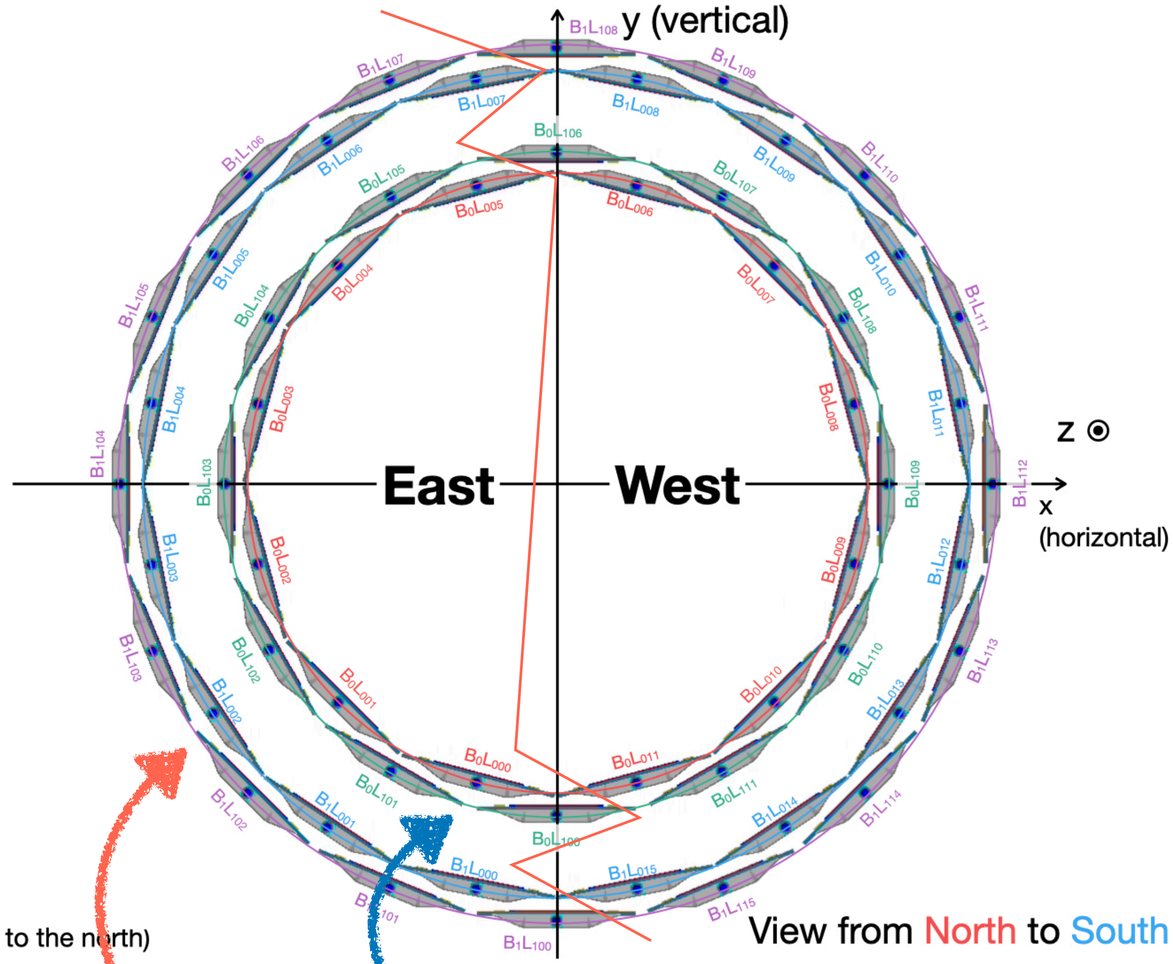
Back up

INTT geometry

INTT: 2 sensors X 2 sides of half-ladders X 56 ladders = 224 sensors



Notation: B_xL_{yzz}
- x: Barrel ID (0 for inner or 1 for outer)
- y: Layer ID (0 for inner or 1 for outer)
- zz: Ladder ID (from 0 to 15)



Axis (Right-haded coordinate)
- x-axis: $\vec{y} \times \vec{z}$
- y-axis: Vertically upward direction
- z-axis: The blue beam direction (pointing to the north)