Time-of-Flight Estimation using Machine Learning Techniques

Konrad Helms

ILD Analysis/Software Meeting, June 5, 2024

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN IN PUBLICA COMMODA SEIT 1737

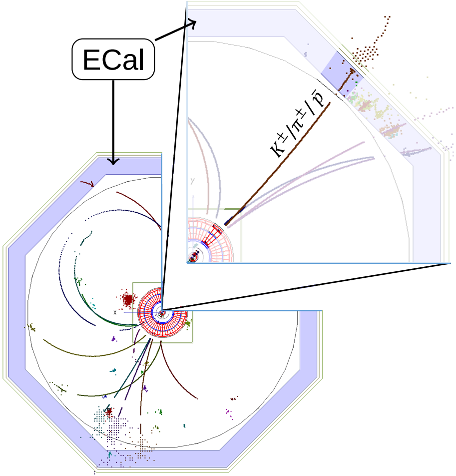
HELMHOLTZ

Time-of-Flight and Particle Identification

- time-of-flight (TOF) estimation for hadron showers $\Rightarrow$ particle identification
- for $p \lesssim \mathcal{O}(10\,\mathrm{GeV})$: $\pi^{\pm}$ vs. $K^{\pm}$ vs. $\overset{(-)}{p}$
- particle identified by electric charge &

$$m_0 = \frac{p \cdot \mathrm{TOF}}{\ell_{\mathrm{track}}} \sqrt{1 - \left(\frac{\ell_{\mathrm{track}}}{(\mathrm{TOF} \cdot c)}\right)^2}$$

- case study: showers in barrel and endcap of the International Large Detector



A Short History of Time-of-Flight Estimation # 1

Old Benchmark

- select hits closest to track extrapolation into ECal in **first 10 layers**
- correct measured hit time by travel time to detection position, assuming velocity c:

$$t_{\text{corrected},i} = t_i - \frac{\text{d}(\text{ECal}, (x, y, z)_i)}{c}$$

$$\text{TOF} = \frac{1}{n_{\text{hits}}} \sum_{i=1}^{n_{\text{hits}}} t_{\text{corrected},i}$$

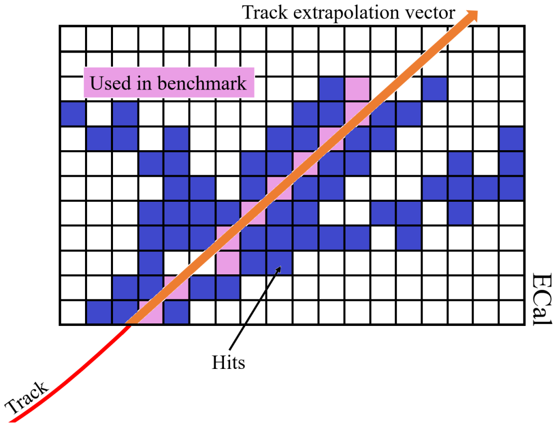
New Benchmark

- only consider the **first 10 layers**
- select hits in a cylinder of radius $R^* = 10.3$ mm around the track extrapolation vector
- correct measured hit time t_i by travel time to detection position d_i, assuming c: $\tilde{t}_i = t_i - \frac{d_i}{c}$
- then, keep only hits i which fulfil: $|\tilde{t}_i - \text{median}(\tilde{t}_i)| \leq 179\,\text{ps}^*$
- use the remaining hits in old benchmark

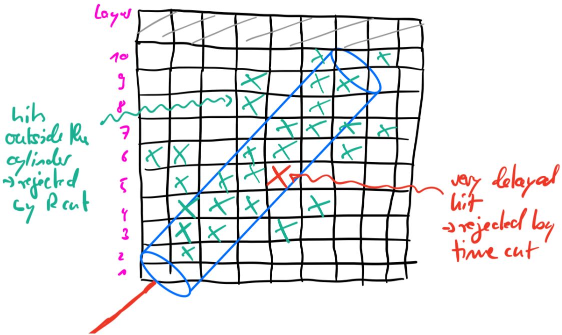
* more on the specific values later

A Short History of Time-of-Flight Estimation # 2

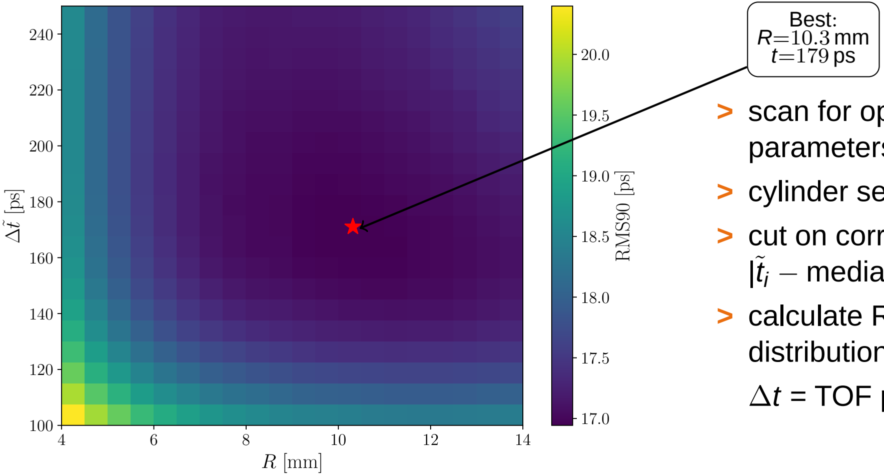
Old Benchmark



New Benchmark



The New Benchmark Parameter Selection



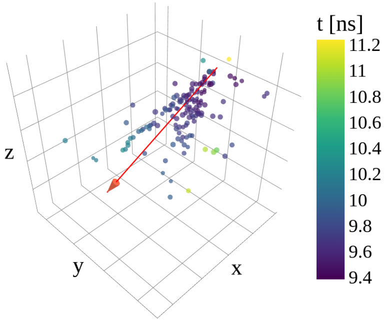
- scan for optimal parameters
- cylinder selection: R
- cut on corrected time: $|\tilde{t}_i - \text{median}(\tilde{t}_i)| \leq \Delta\tilde{t}$
- calculate RMS90 of the distribution:

  Δt = TOF pred. - true TOF

Dataset

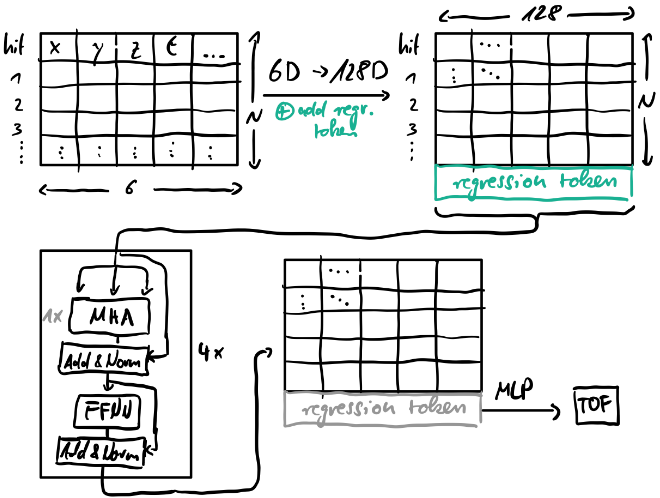
- 6D point cloud $\oplus$ track information:

$$\{ \underbrace{[x, y, z, t, e, \text{hit layer}]_i}_{\text{coordinates, per hit}} \oplus \underbrace{[p_x, p_y, p_z, x_{\text{ECal}}, y_{\text{ECal}}, z_{\text{ECal}}]}_{\text{reconstructed track information, per shower}} \mid i = 1 \dots n_{\text{hits}} \} \times n_{\text{showers}}$$

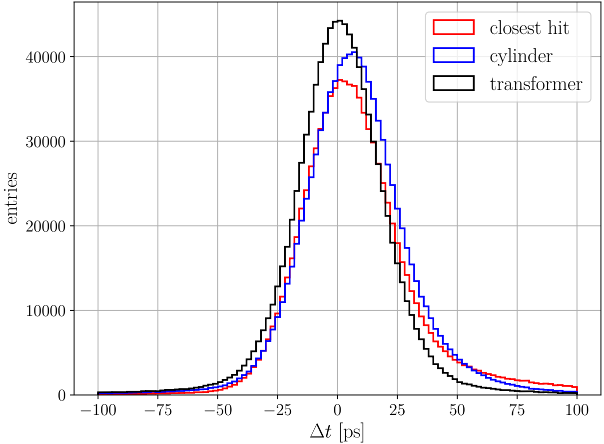


- hit time resolution ± 50 ps in ECal
- rotate showers, such that track extrapolation vectors align (implicitly uses $\vec{p}$)
- centre showers (implicitly uses $x_{\text{ECal}}, y_{\text{ECal}}, z_{\text{ECal}}$)
- network only gets $[x, y, z, t, e, d, d_{\text{T}}, d_{\tilde{t}}, \text{hit layer}]_i$

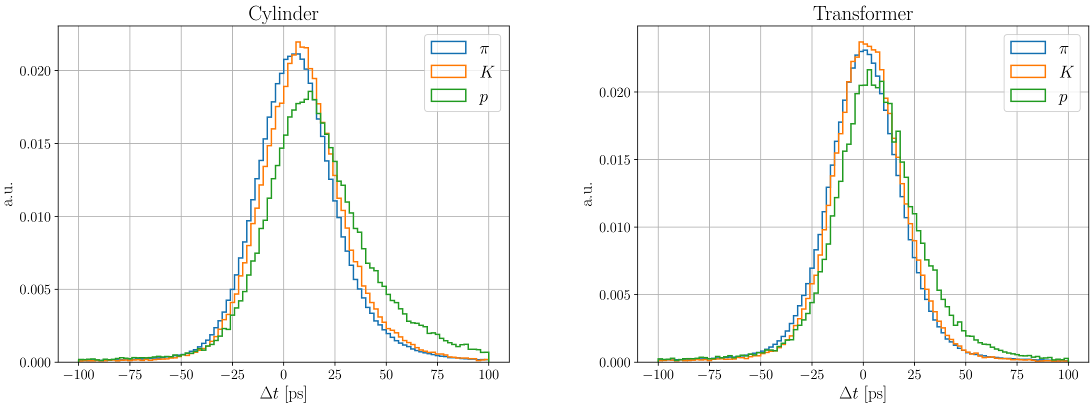
Time-of-Flight Transformer



Results # 1

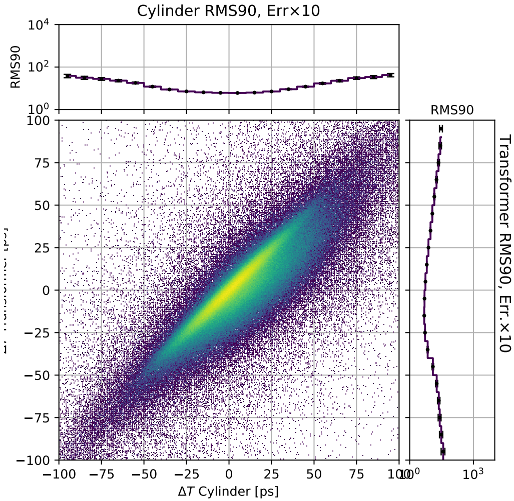


estimator	RMS 90	mean 90
closest hit	25.56 ± 0.02	9.15 ± 0.03
cylinder	17.36 ± 0.01	6.80 ± 0.02
transformer	15.69 ± 0.01	1.13 ± 0.02



estimator	RMS 90: π	RMS 90: K	RMS 90: p
cylinder	17.01 ± 0.01	16.88 ± 0.04	22.42 ± 0.07
transformer	15.60 ± 0.01	14.75 ± 0.03	18.41 ± 0.06

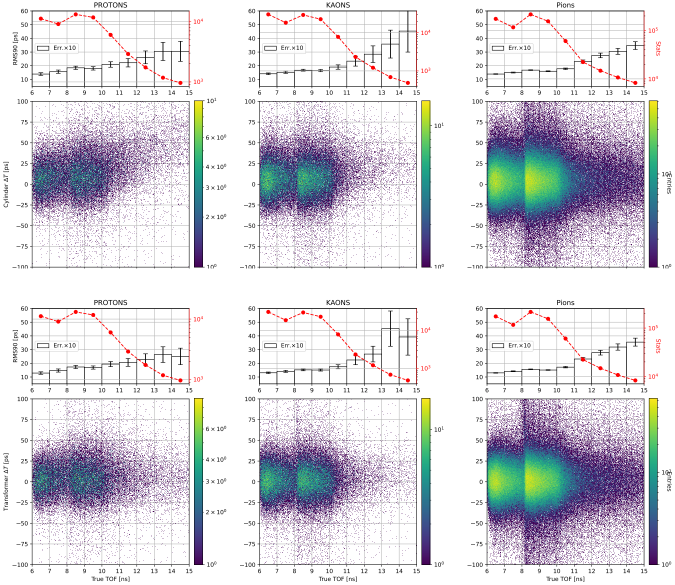
Results # 3



- > transformer vs. cylinder
- > performance similar, except for slight bump to the lower right
- > transformer performs better in these cases

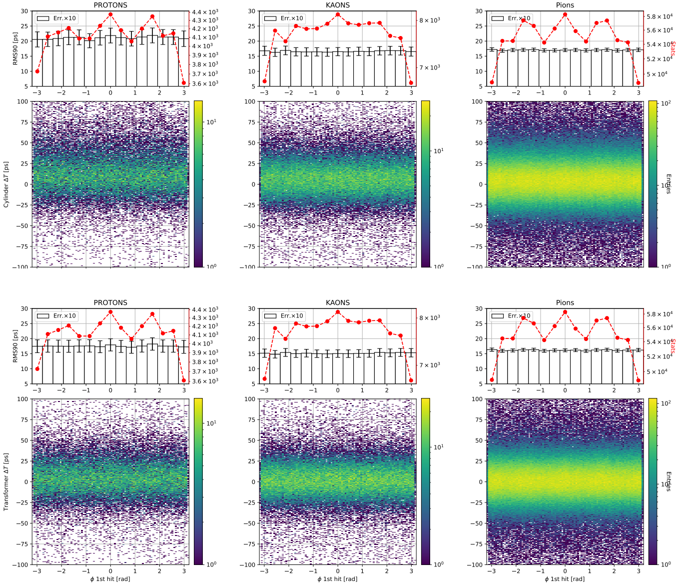
Results # 4

- proton TOF predictions improved
- in high statistics region RMS90 vs. true TOF spectrum flat
- in low statistics region steep rise of RMS90 for both



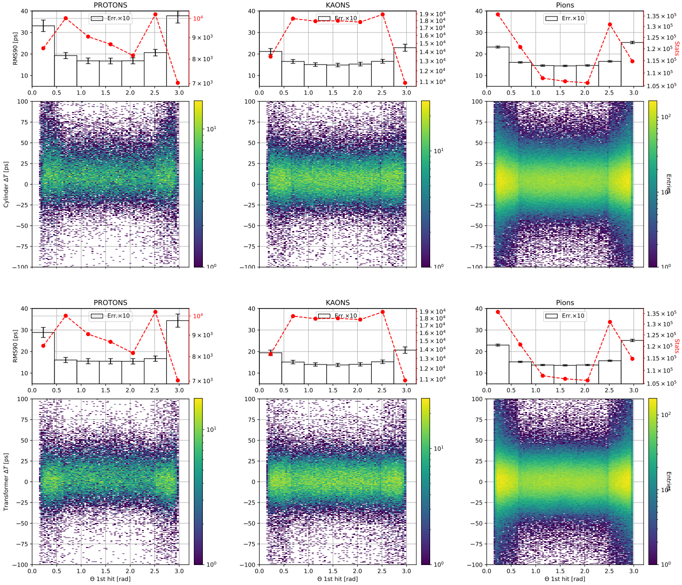
Results # 5

- RMS90 vs ϕ spectrum completely flat for both
- transformer proton TOF predictions significantly better



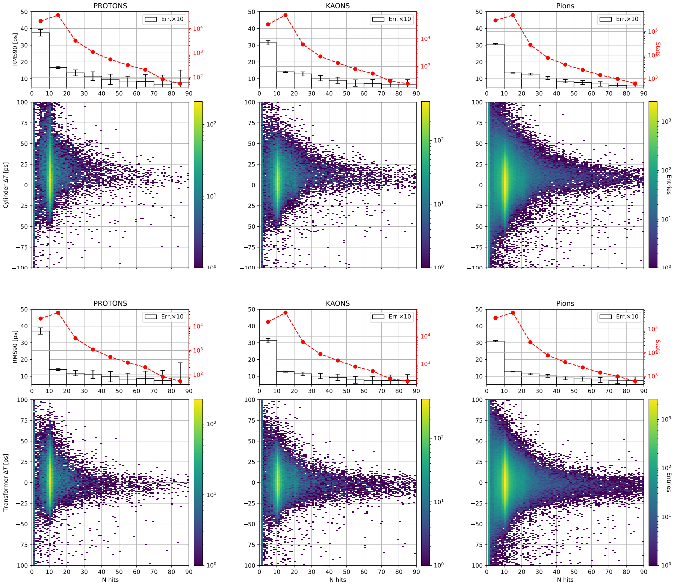
Results # 6

- worse performance of both methods in endcap
- protons suffer the most
- barrel RMS90 vs θ spectrum flat



Results # 7

- 1 hit ’showers’ are outliers, therefore large RMS90
- for minimum ionising particles with $\simeq 10$ hits performance good
- prediction accuracy increases with number of hits in shower equally for both



Summary & Outlook

Summary:

- transformer is slightly better - everywhere
- not a surprise for a model with $\mathcal{O}(100\mathrm{k})$ parameters
- transformer is a lot slower (in Python): ~ 2 h for 1 mio. showers (this can be heavily optimized)

On-going/Outlook:

- predict particle *v*
- focus on removing the bias (in RMS and mean) for the protons completely
- weighted RMS, keep statistics
- more data = more better + train longer!

Thank you!

Contact

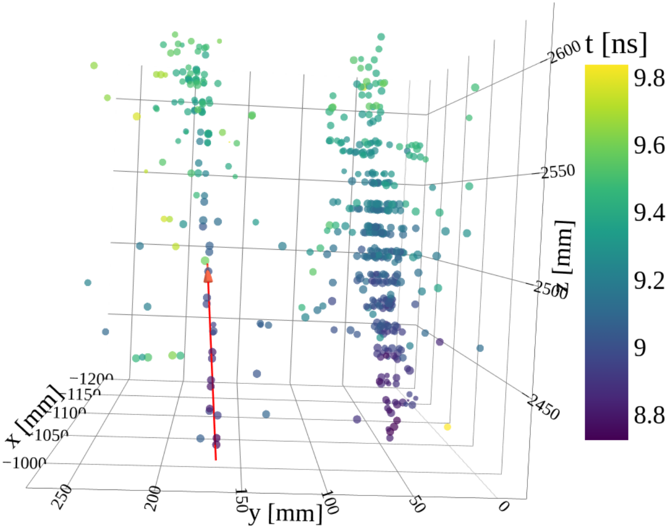
Deutsches Elektronen-
Synchrotron DESY

www.desy.de

Konrad Helms
0009-0008-7200-7670
DESY FTX / Georg-August-Universität Göttingen
konrad.helms@desy.de

Backup

Problems



t [ns]
9.8
9.6
9.4
9.2
9
8.8
2600
2550
2500
2450
z [mm]
−1200
−1150
−1100
−1050
−1000
x [mm]
250
200
150
100
50
0
y [mm]