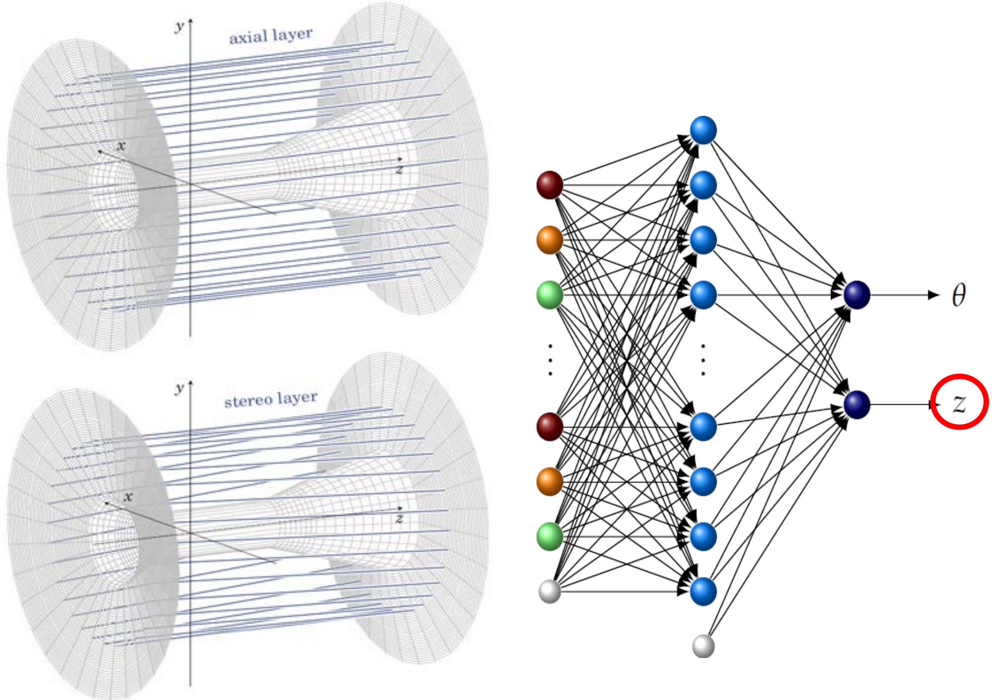
The Neural Network First-Level Hardware Track Trigger of the Belle II Experiment

Christian Kiesling
Max-Planck-Institute for Physics

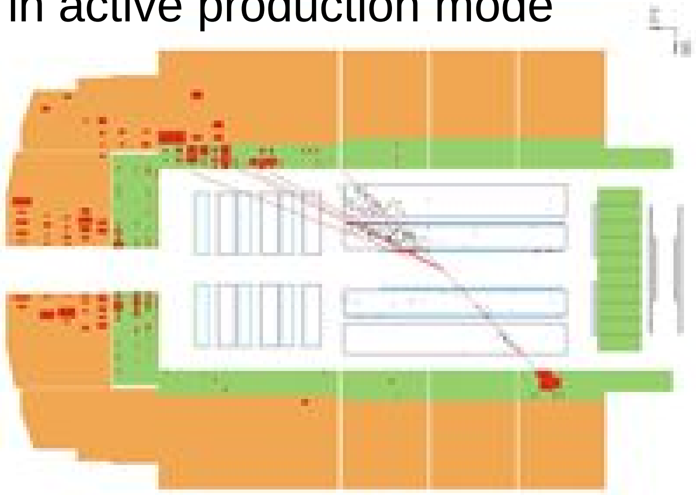
Overview:

- SuperKEKB & Belle II's Track Trigger
- Principles of the Neural Approach to Track Triggers
- Physics-motivated Preprocessing of Input Variables
- Performance of the Neural Track Trigger,
  -> Launch of a Minimum Bias Single Track Trigger (STT)
- Problems and Solutions -> Upgrade program
- Summary and Conclusions



Legacy from AINHEP 1999, Heraklion

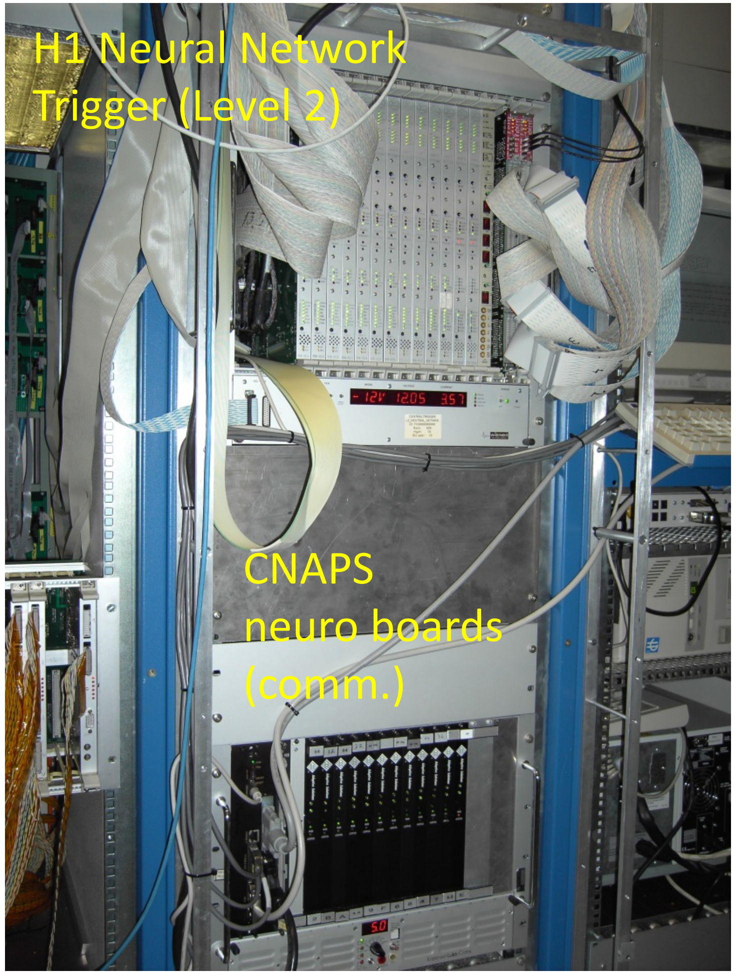
H1 @HERA ep Collider:
First Neural Trigger in HEP
in active production mode



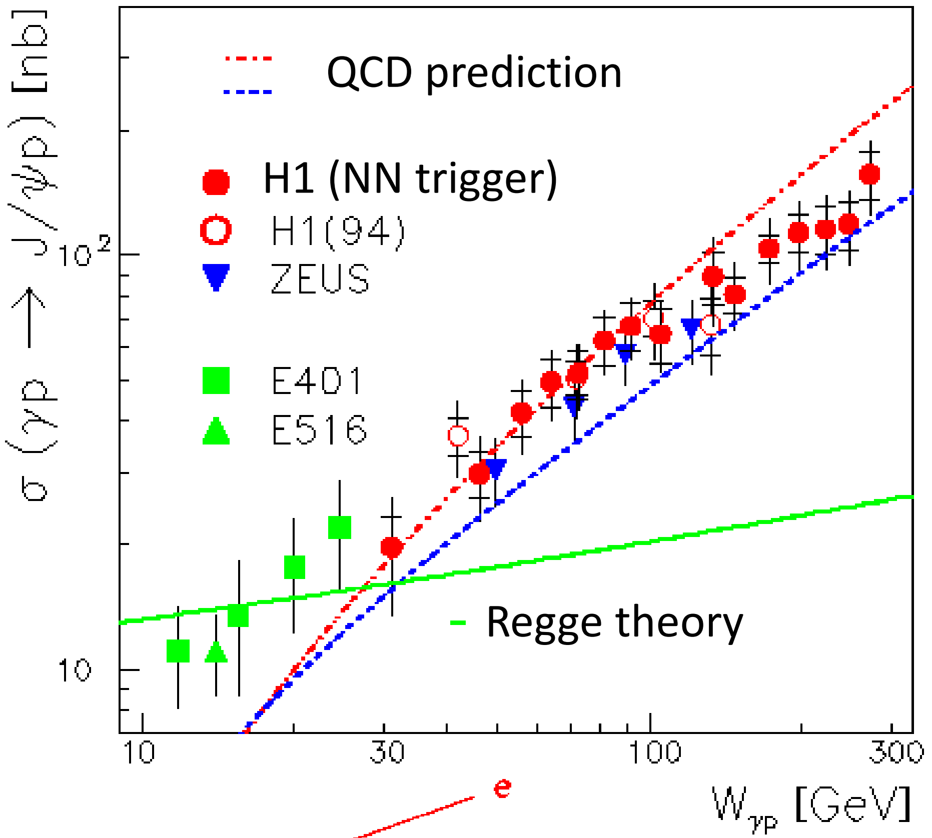
12 networks running in parallel, each trained for specific physics

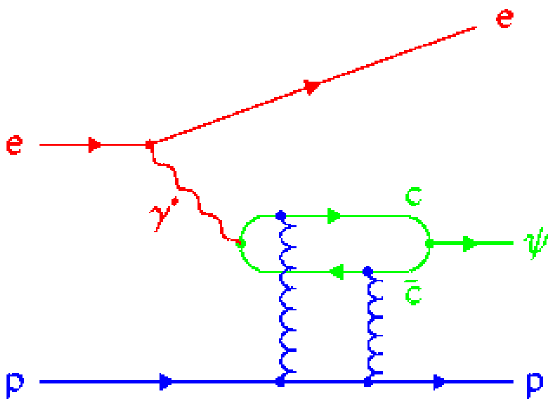
Principle: „open“ L1, „clean“ via L2

J. K. Köhne, C. Kiesling *et al.*, Nucl. Instrum. Meth. A **389** (1997) 128.



Preprocessor: all subdetectors
Networks: 64 x 64 x 1
Latency: 20 µs





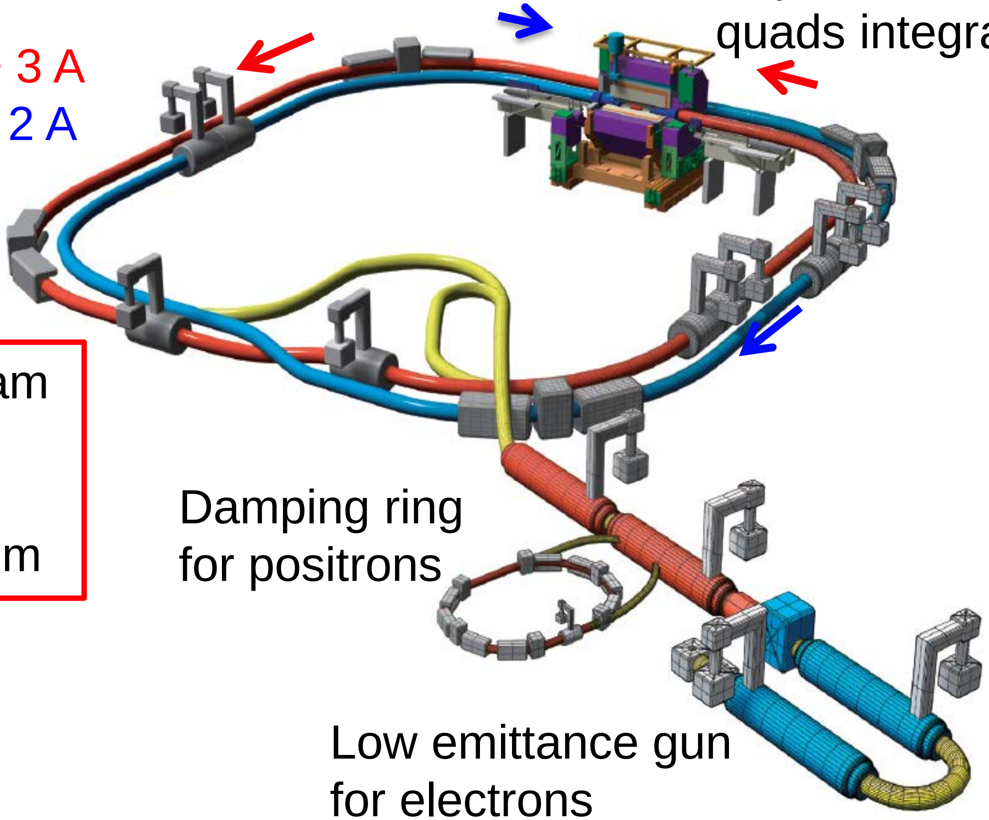
C. Adloff et al., Phys. Lett. B**483** (2000) 23

SuperKEKB & Belle II

e^+ 4GeV ~ 3 A
e^- 7GeV ~ 2 A

Nano-beam scheme:

σ_y ~ 50 nm

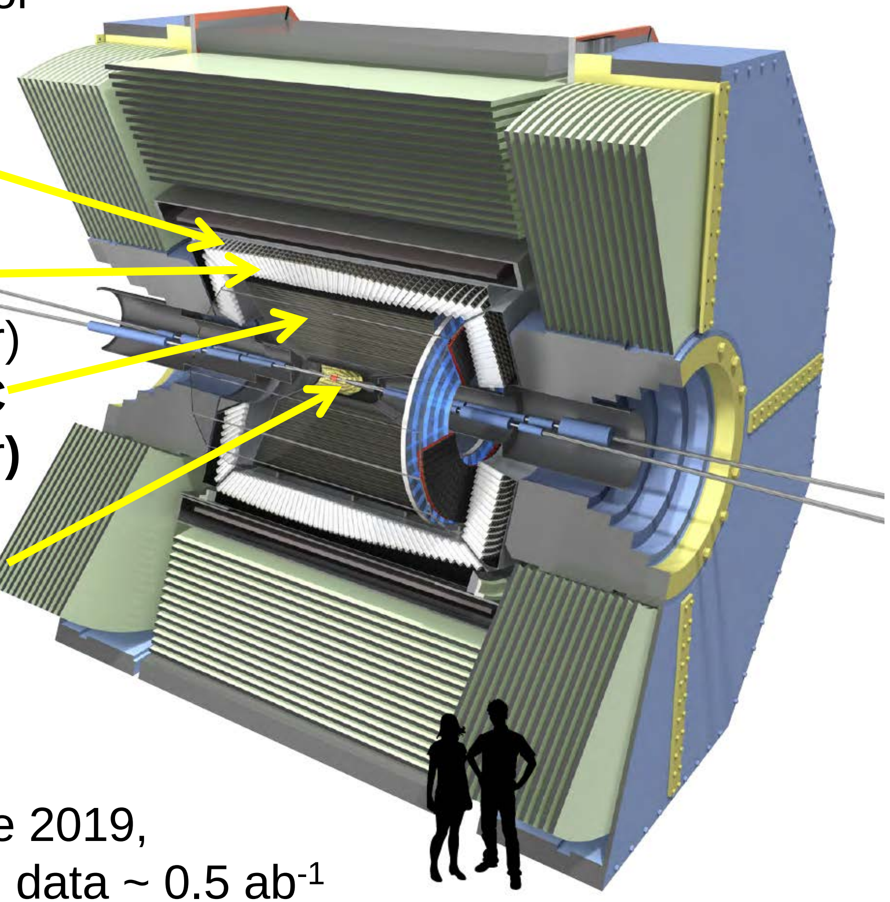


Superconducting final focusing quads integrated into detector

Damping ring for positrons

Low emittance gun for electrons

target $\mathcal{L}$ = 6x10^{35}/cm^2/s



new PID

ECL
(-> trigger)
new CDC
(-> trigger)

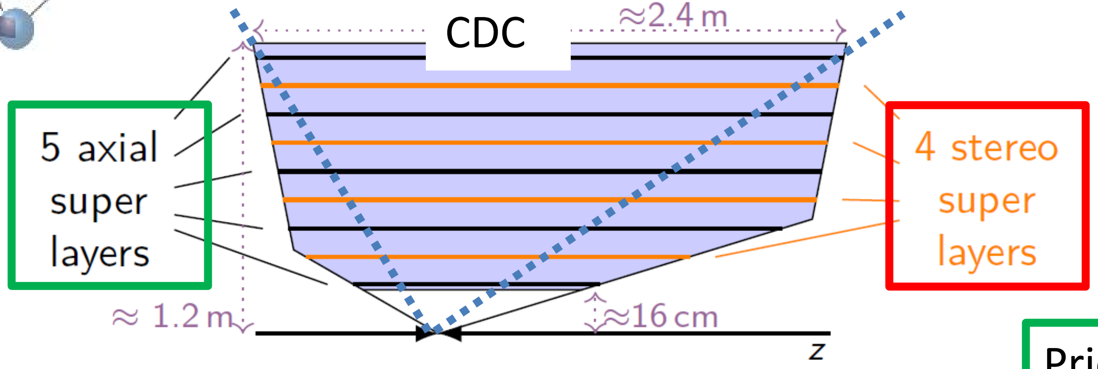
new VXD

running since 2019,
accumulated data ~ 0.5 ab^{-1}

peak luminosity $\mathcal{L}$ = 4.7 x 10^{34} /cm^2/s
I(e^+ / e^-) = (1.4/1.2 A) , β^* = 1 mm

located @ KEK, Tsukuba, Japan

The „Conventional“ Belle II L1 Track Trigger („2D“)

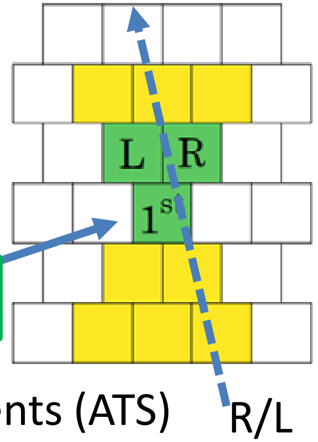


- 56 layers combined to 9 super layers (SL)
- 2336 track segments (TS) in 9 SL

Track Segments: Hit patterns compatible with traversing track (LUT)

Priority wire = „hit“

5 axial SLs



Axial track segments (ATS)

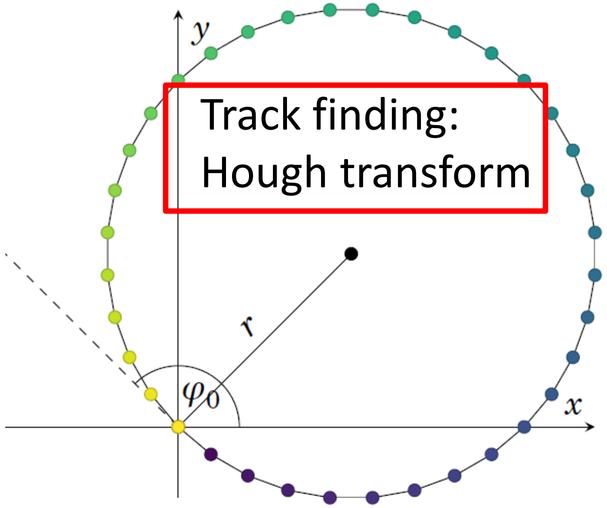
L1 Pipeline

CDC front end → Track Segment Finder (TSF) → HoughTransform + peak finding → Global Decision Logic (GDL)

5µs

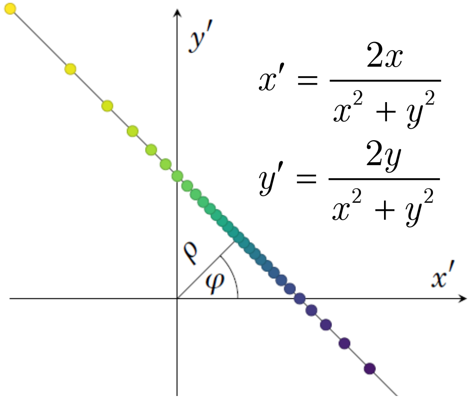
algos on FPGA boards „UT(3)“ Virtex 6 XC6VHX380/565T

geometrical space → conformal space → parameter space



Track finding: Hough transform

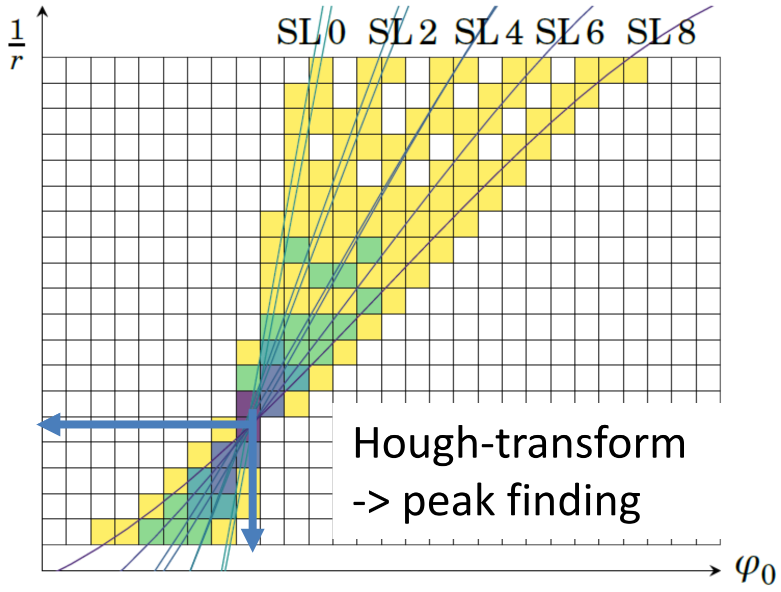
Tracks from IP by definition



$$x' = \frac{2x}{x^2 + y^2}$$

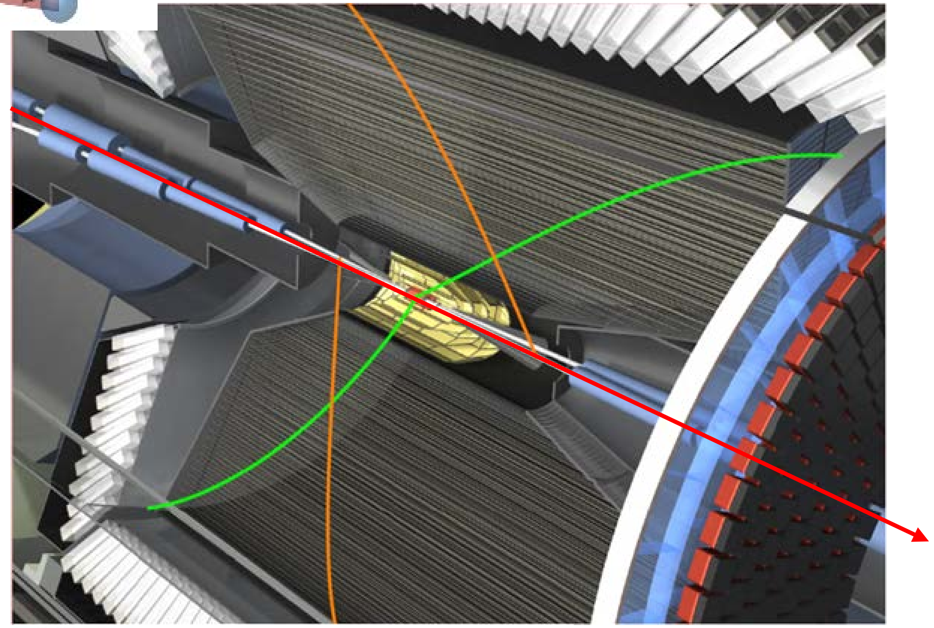
$$y' = \frac{2y}{x^2 + y^2}$$

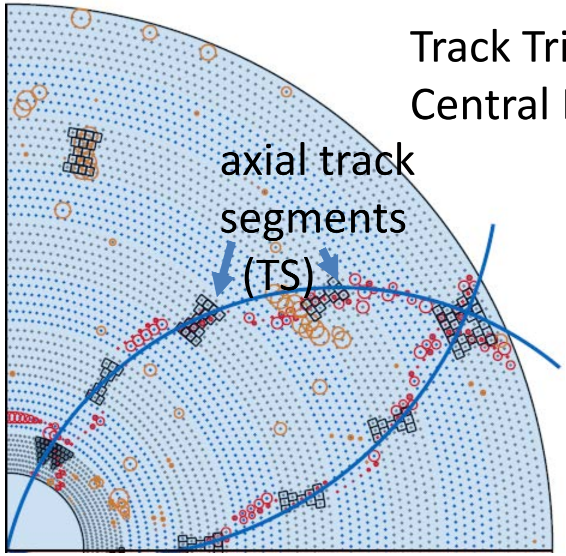
Each {hit + IP} produces a set of [1/r, φ] points, basically on a straight line



Hough-transform -> peak finding

Challenge of the Conventional Track Trigger



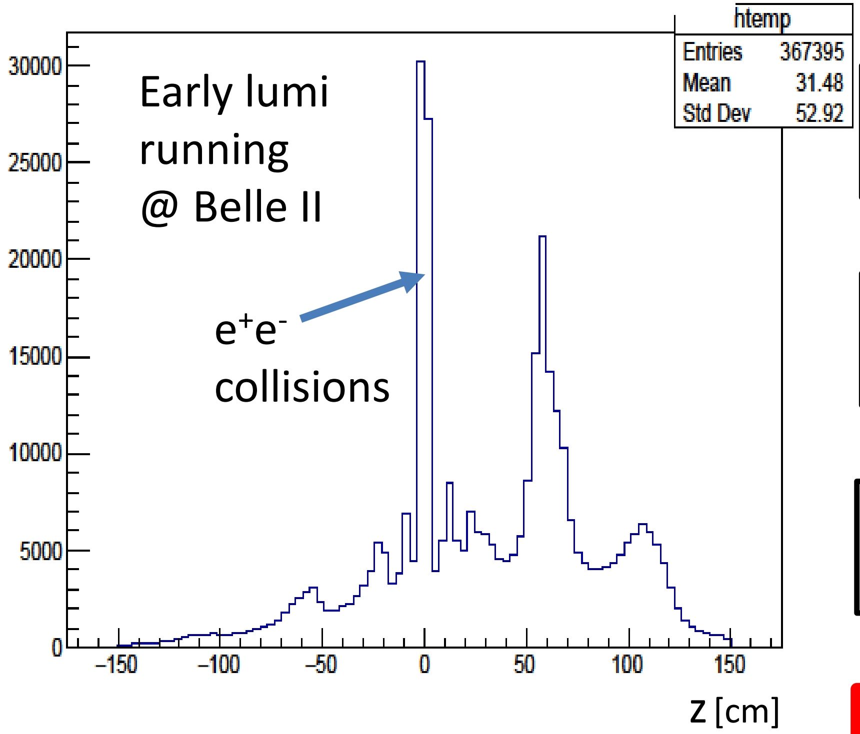


Track Trigger derived from Central Drift Chamber:

Belle II:
Initially, track trigger only in 2D, using Hough transforms

2D tracks >= 2

z-vertex distribution (offline) :



Majority of tracks from „obstacles" outside of the interaction region (IP) (|z| >> 1 cm): only ~10% from IP

→ „z-vertex" trigger mandatory

L1 Pipeline

- CDC front end
- Track Segment Finder (TSF)
- HoughTransform + peak finding
- 3D track fit: iterative process
  - Latency ??
  - precision ??

  -> Machine Learning

5μs

AI Trigger Group at Belle II

KIT ITIV

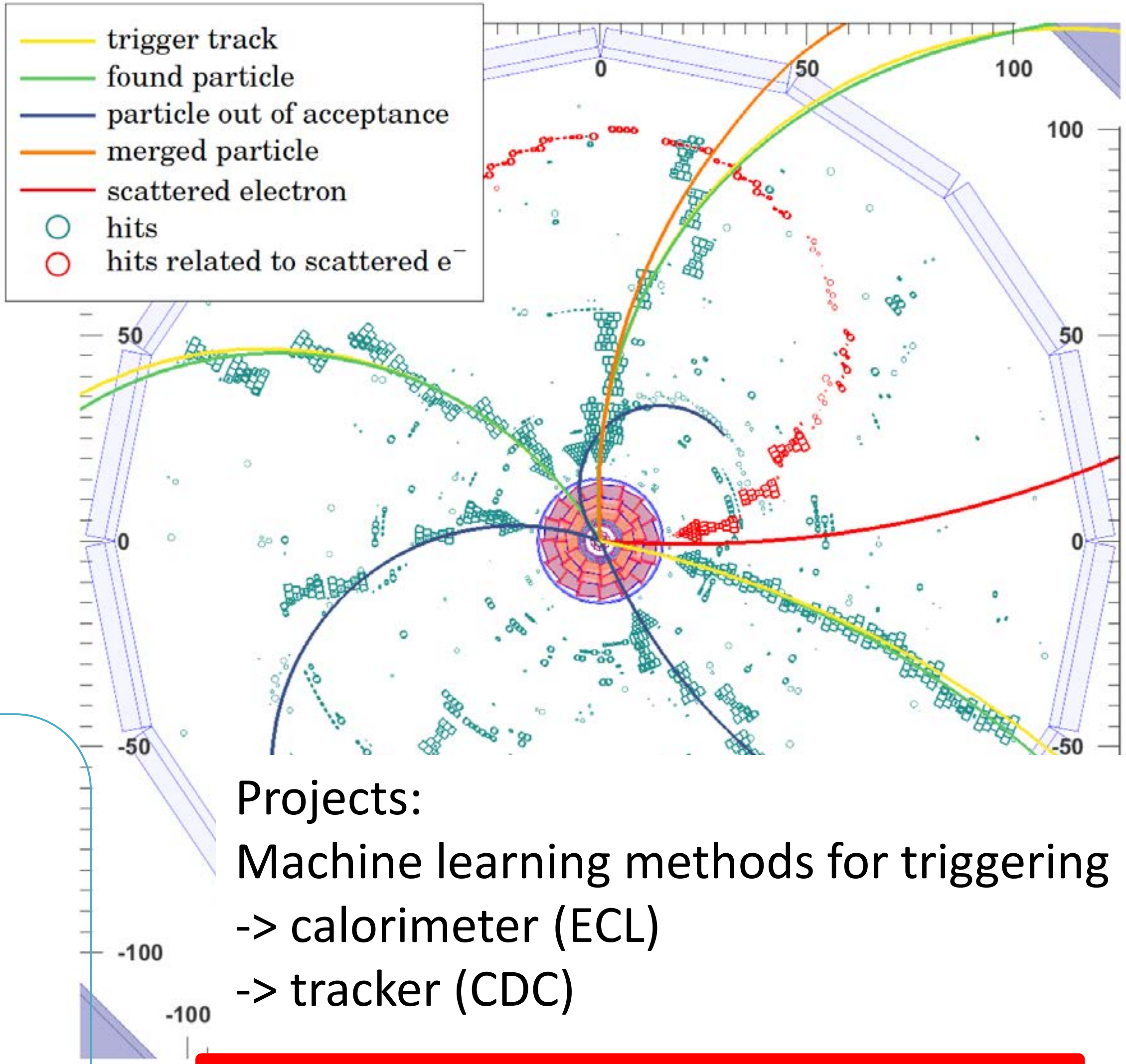
- Marc Neu
- Kai Unger
- **Jürgen Becker**

KIT ETP

- Lea Reuter
- Greta Heine
- Slavomira Stefkova
- **Torben Ferber**

MPI / LMU / TUM

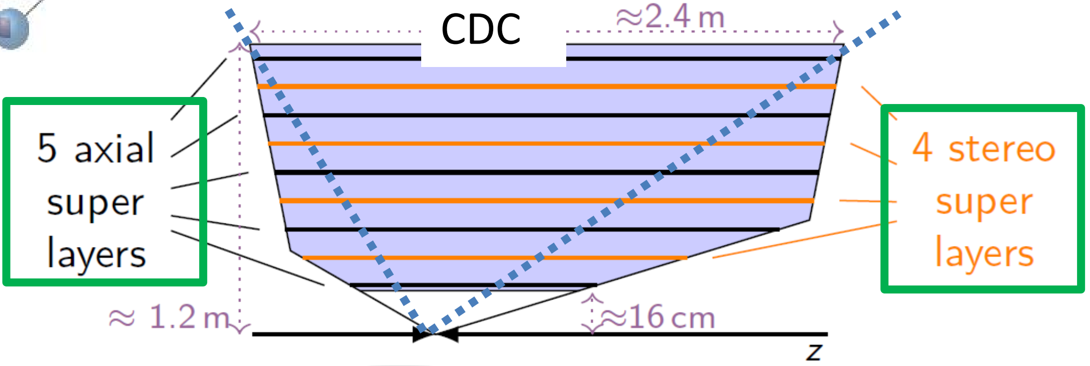
- Felix Meggendorfer
- Simon Hiesl
- Timo Forsthofer
- **Christian Kiesling**
- Alois Knoll

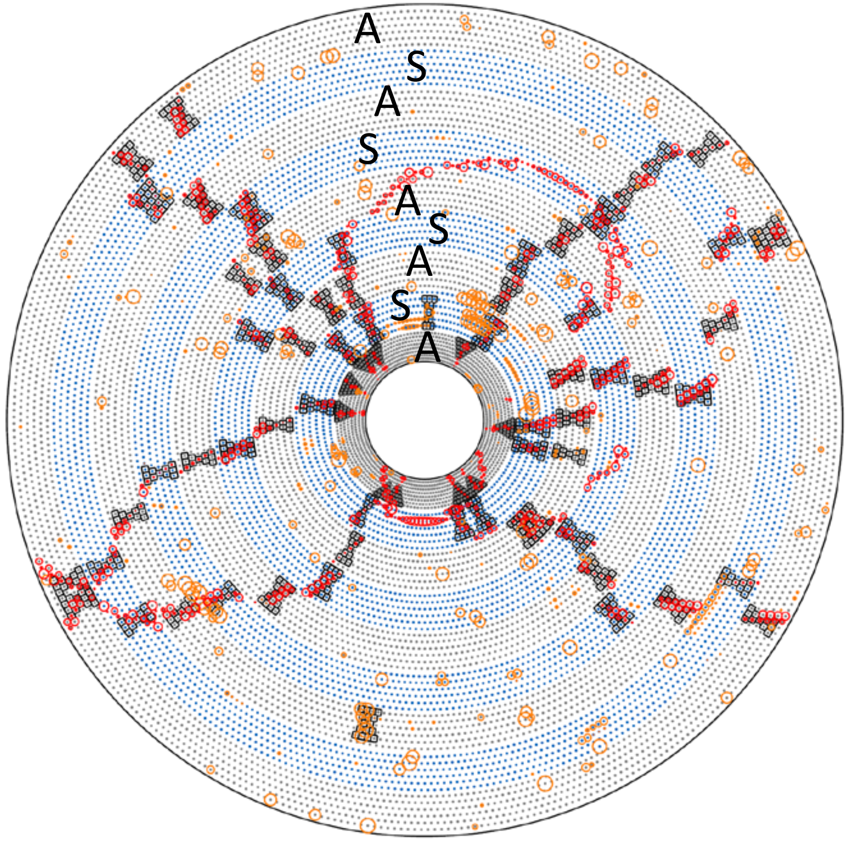


Projects:
Machine learning methods for triggering
-> calorimeter (ECL)
-> tracker (CDC)

here: Neural Network „z“ Trigger @L1

Principle of the Neural L1 Track Trigger





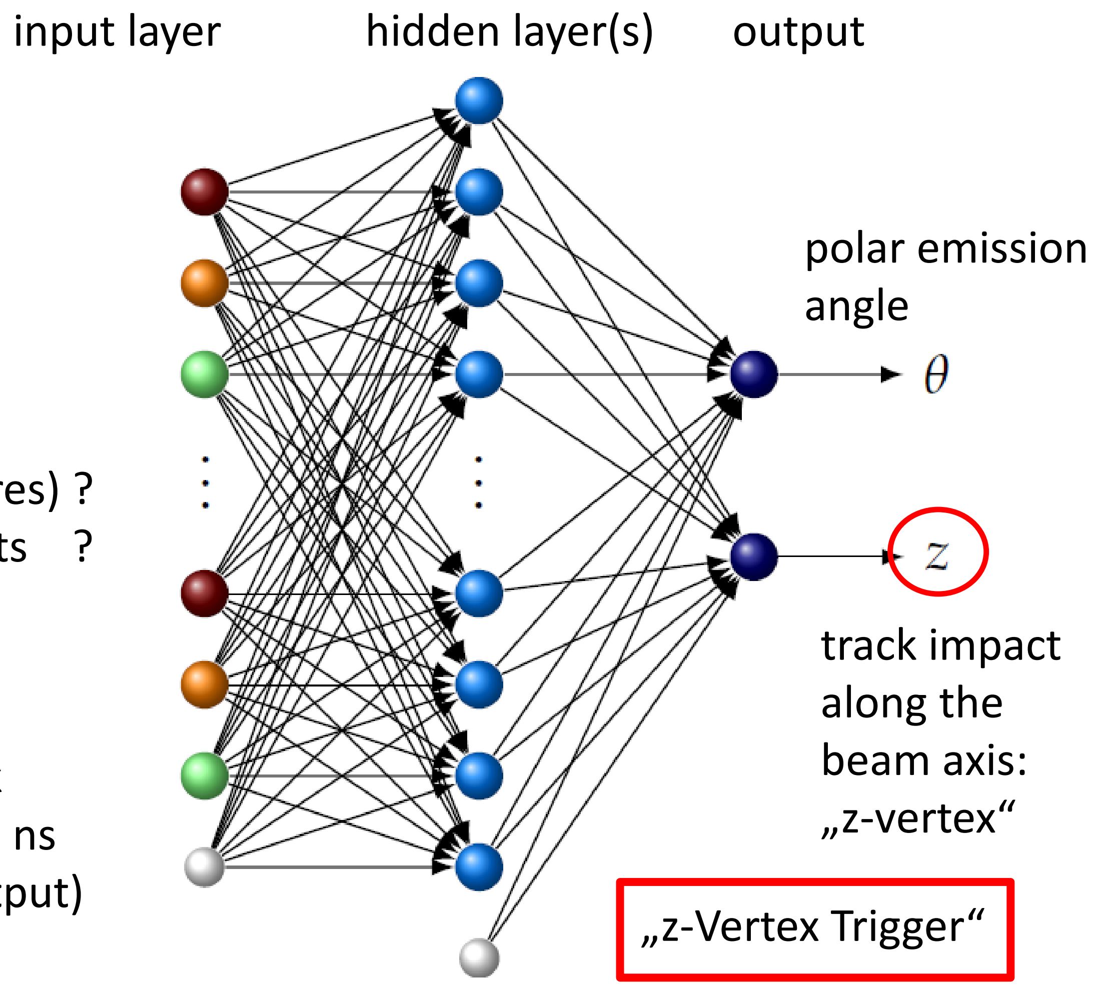
Central question:

What are the input variables:

- entire „picture“ (wires) ?
- set of track segments ?
- ?

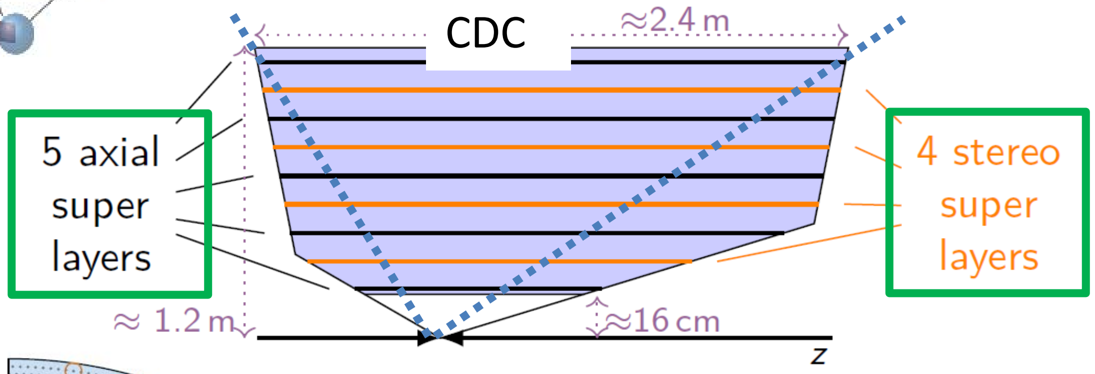
Note:
total latency for track reconstruction ~ 700 ns (starting with TSF output)

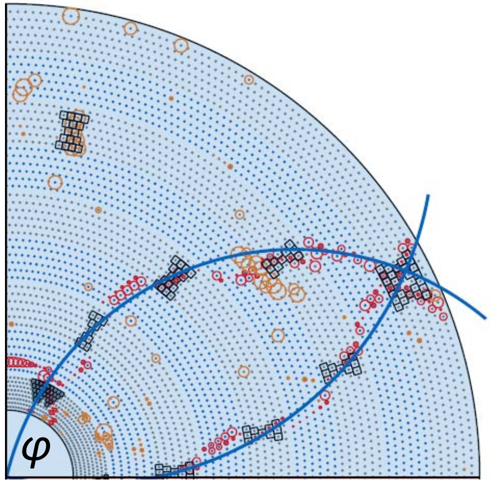
Architecture for each track candidate (networks to solve a regression task)



„z-Vertex Trigger“

Input Preprocessing & Neural Networks



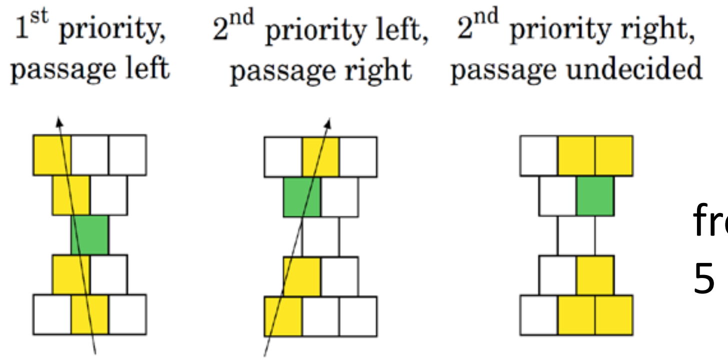


Target: tracks = well-known geometrical objects

patterns in known B-field:
- helices in space
- circles in transverse plane

„Natural input": 2D track candidates in each of 4 quadrants from Hough transforms (-> azimuth φ and $1/R = 1/p_T$)

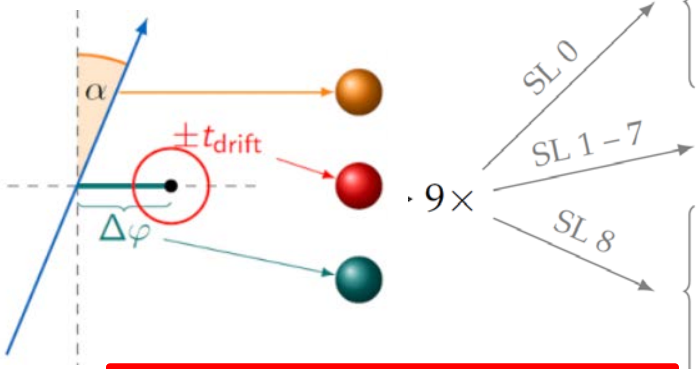
- calculate crossing angle α through TS
- determine „sign" of drifttime (from wire pattern in TS)

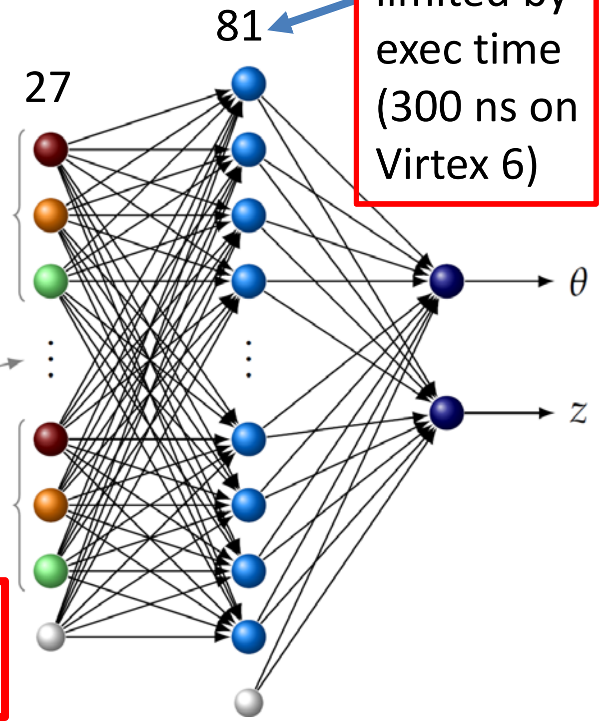


LUT

from all
5 aSLs / 4 sSLs

3 preprocessed inputs per TS in each of the 9 SLs:
- crossing angle α (calculation)
- signed dirfttime (LUT)
- stereo wires selected from predef. range Δϕ (LUT)

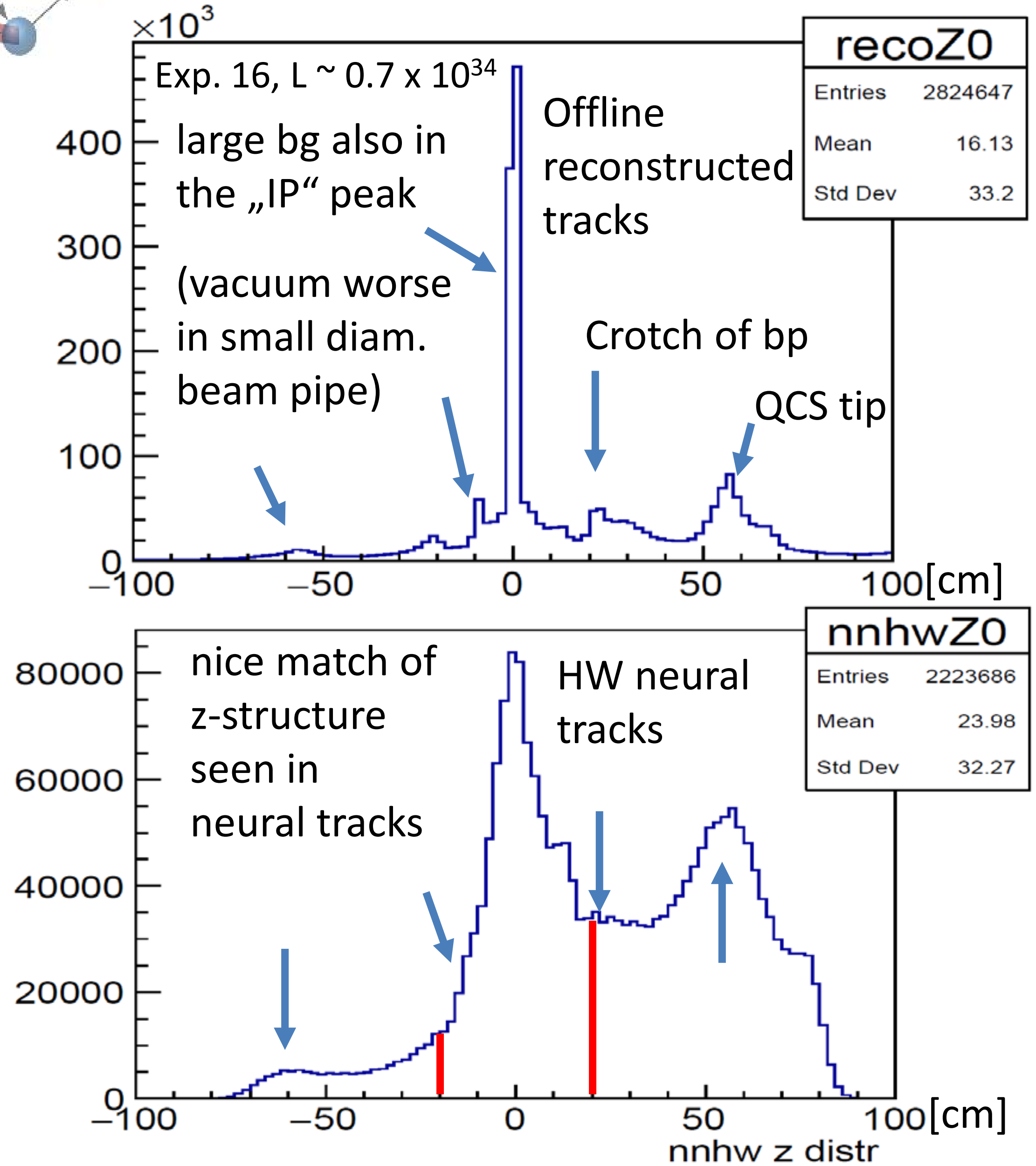




limited by exec time (300 ns on Virtex 6)

Networks trained with fully reconstucted offline tracks

Commissioning the Neural z-Trigger in 2020



Fall 2020 running

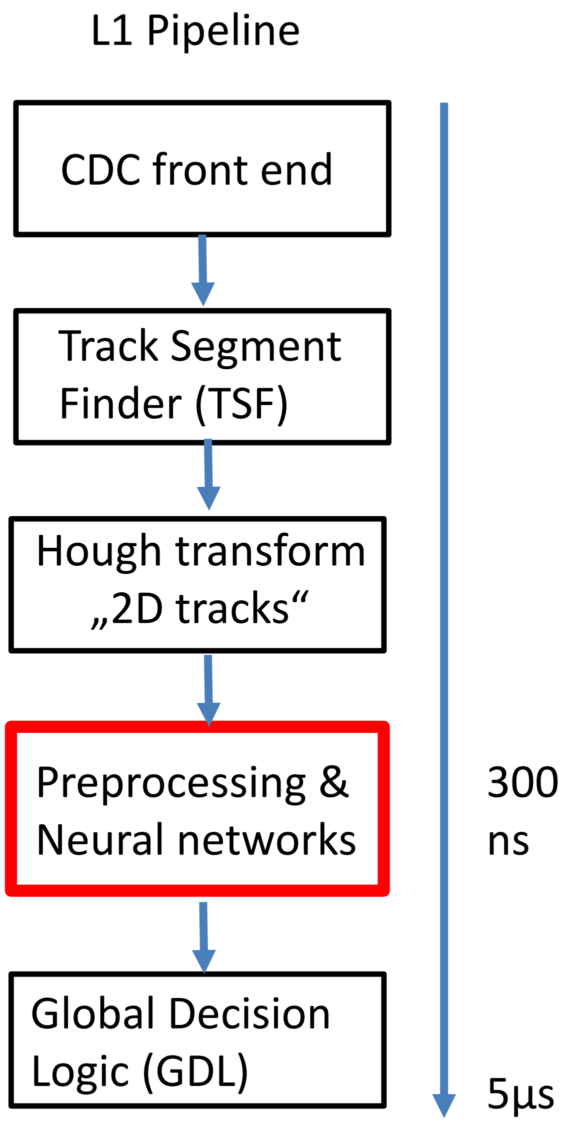
Reco tracks:
z-distribution after full off-line reconstruction,
including VXD space points

Networks trained with real data from May-June 2020

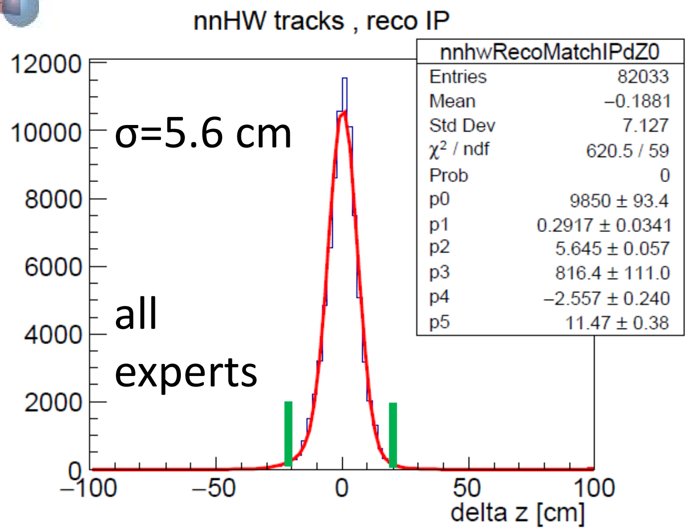
The „Expert Networks“:

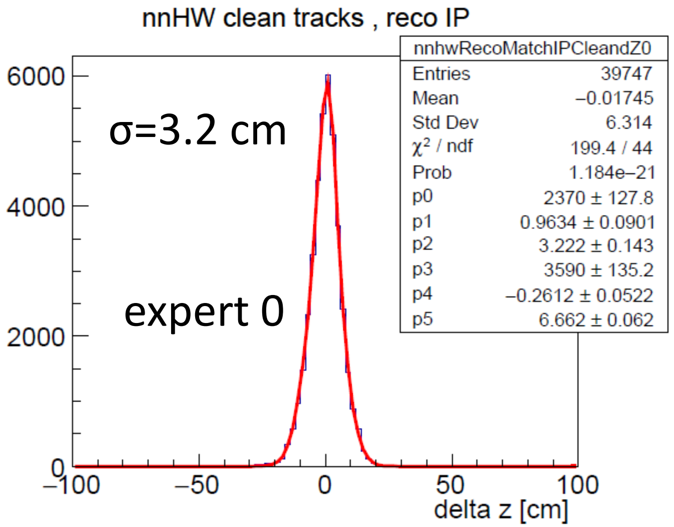
5 different networks trained, depending on the number of available stereo TS

Expert 0: all 4 stereo TS
Expert 1-4: one of the stereo TS missing



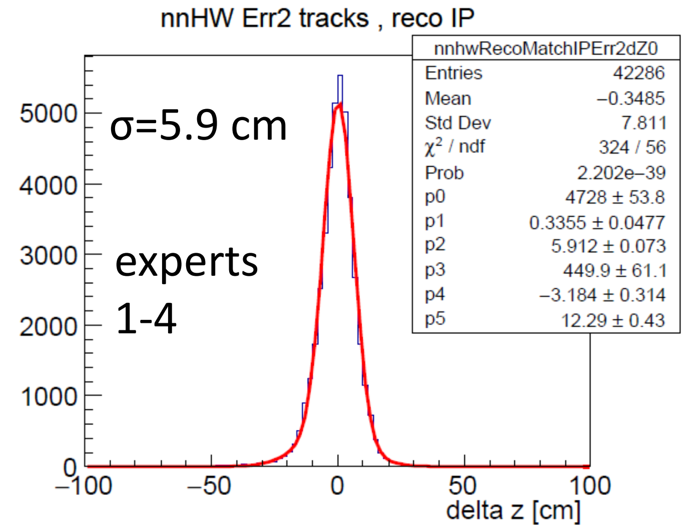
Performance of the Neural z-Trigger (I)





Belle II Track Trigger 2021 running

Large 2D trigger rate in 2021 ->
„y“ bit: ≥ 1 track, |z| < 20 [cm],
require ≥ 2 (2D) tracks && y=1

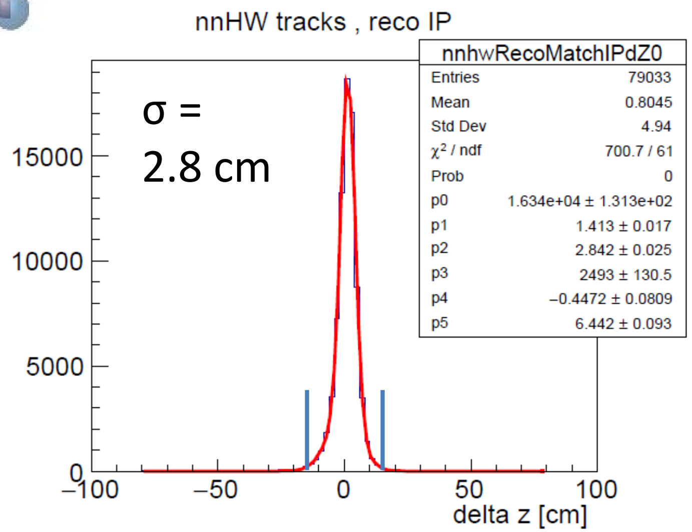


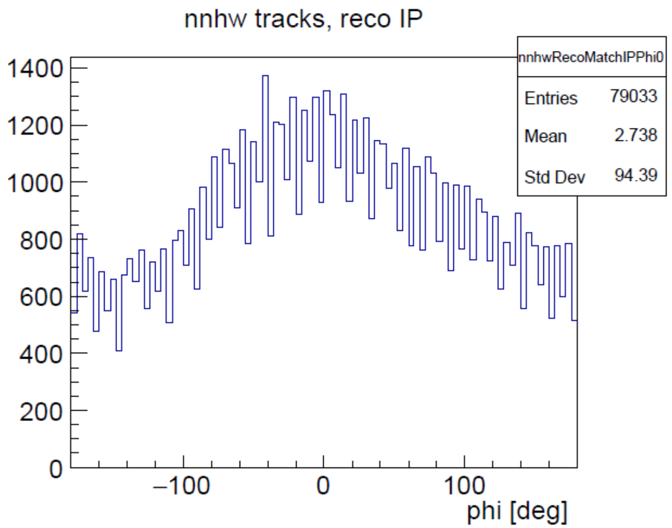
Instantaneous lumi = (3.8×10^{34})
end of 2021,
background rising with luminosity
in 2021

NN resolution of IP tracks very stable, proving robustness of the neural network technique against changing conditons

Fundamental change at Belle II wrt Track Triggers:
due to overwhelming BG,
all 2-Track Triggers require
at least one neural track: „y=1“

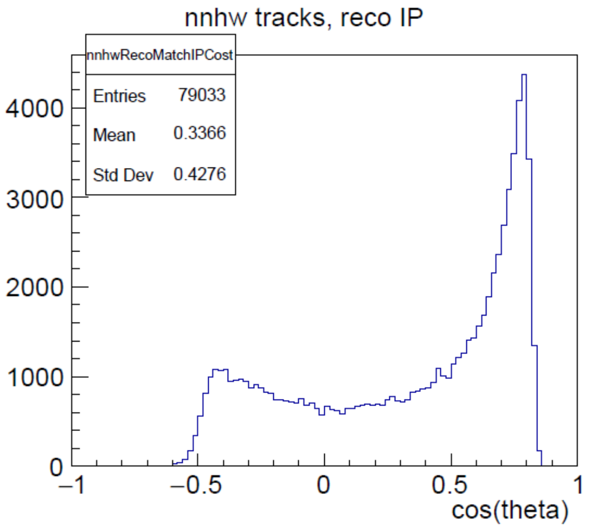
Performance of the Neural z-Trigger (II)





Retrainig of neural networks with data from the end of 2021 (high background data)

Use modern training library **PyTorch** (previously used FANN, integrated into Belle II software library)



Exp 24, runs 790 - 890, New FW

Results from improved training

Gaussian fits to neuro tracks associated with reco tracks from IP (|z|<1 cm, d < 1.5cm)

Central Gauss: σ = 2.8 cm

2nd Gauss: σ = 6.4 cm (13.2 %)

2020 training:
central Gauss σ = 5.6 cm
2nd Gauss σ = 11.5 cm

factor 2 improvement !!

Minimum Bias Single Track Trigger in Belle II : STT

Can we launch a track trigger requiring only one track?

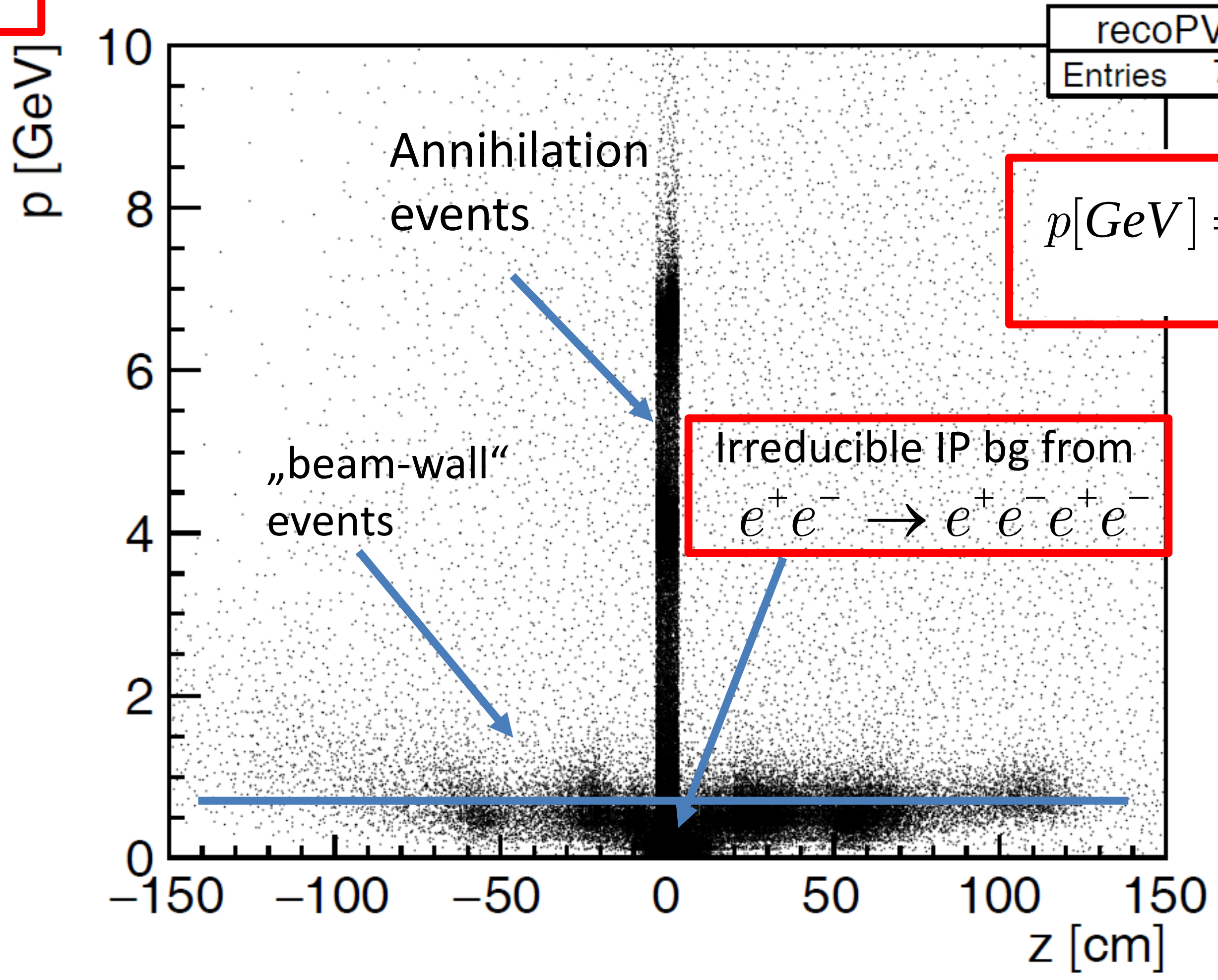
Sources of Background:

Collisions of electrons/positrons with elements of the beam guide system, mostly producing protons from nuclear spallation

(momentum of particles outside of IP mostly below 1 GeV !!)

from IP (!!): QED events

Distribution of Reco Track momentum vs z

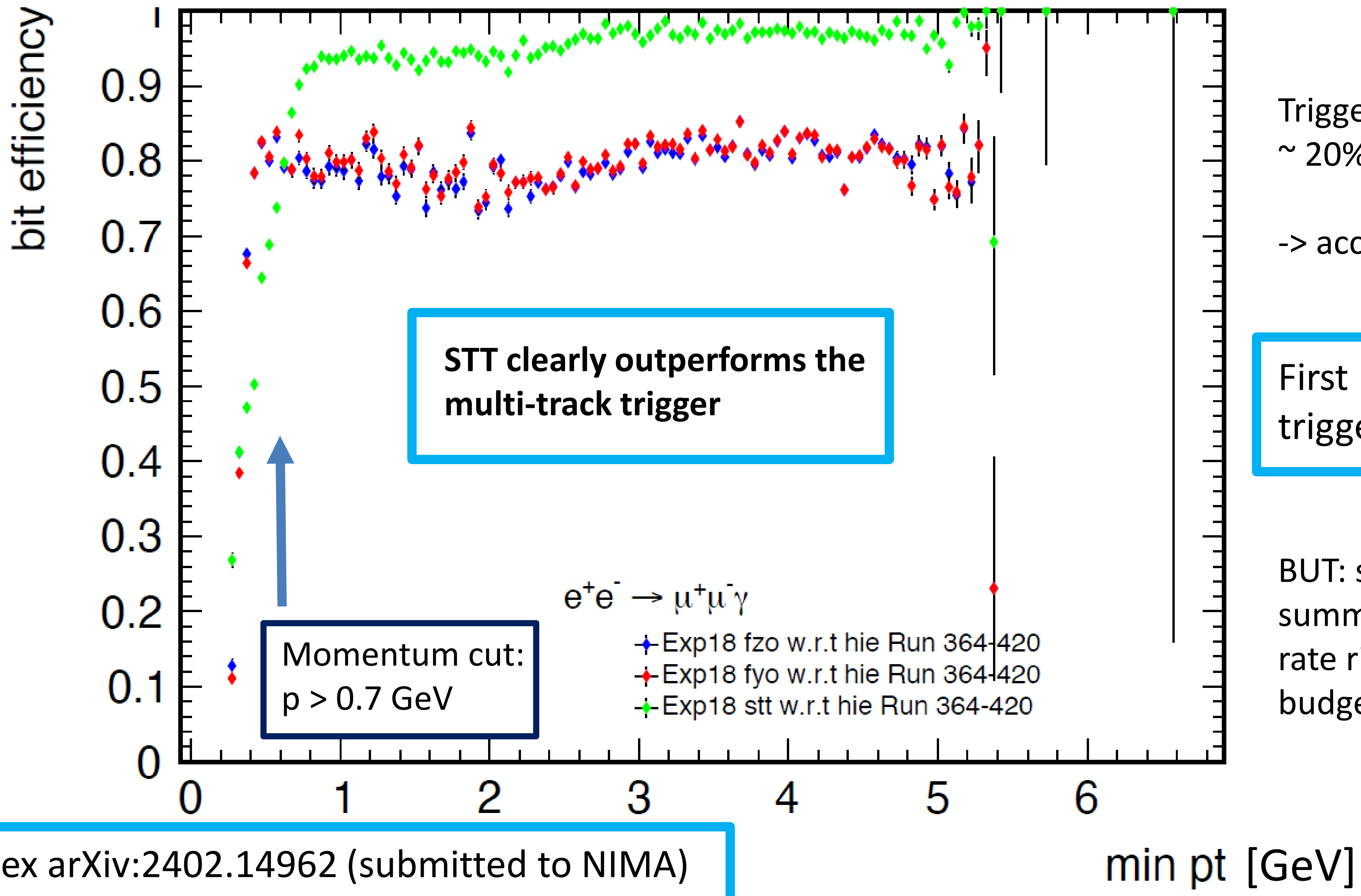


$$p[GeV] = \frac{1}{|\omega|[1/m]\sin(\theta)} 0.3B[T]$$

use the second output („θ“) of the networks

Cut on p > 0.7 GeV

STT: Superior Efficiency



Trigger rate of the STT
~ 20% -25% of total rate budget

-> acceptable

First minimum bias track trigger in HEP

BUT: some problems during summer 2022 running:
rate rising to ~50% of total budget -> ??

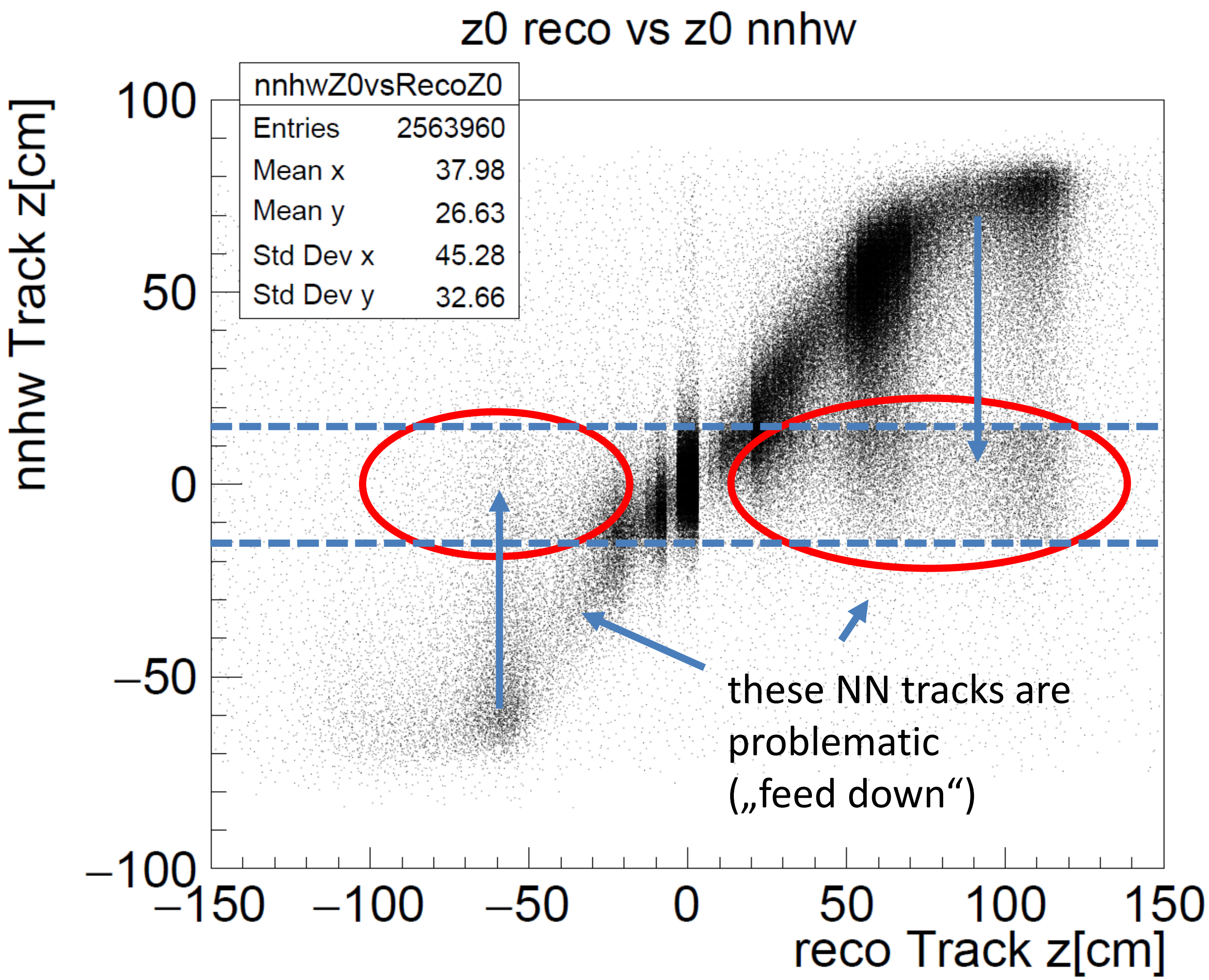
hep-ex arXiv:2402.14962 (submitted to NIMA)

Problems of the STT (I) : „Feed-Down Effect“

Increase of machine-induced background
with rising luminosity

-> increase of STT trigger rate
observed
(but efficiency stable !!)

Excess rate may saturate DAQ
and increase deadtime



Band at |z|<15cm:
acceptance for a
valid neural track

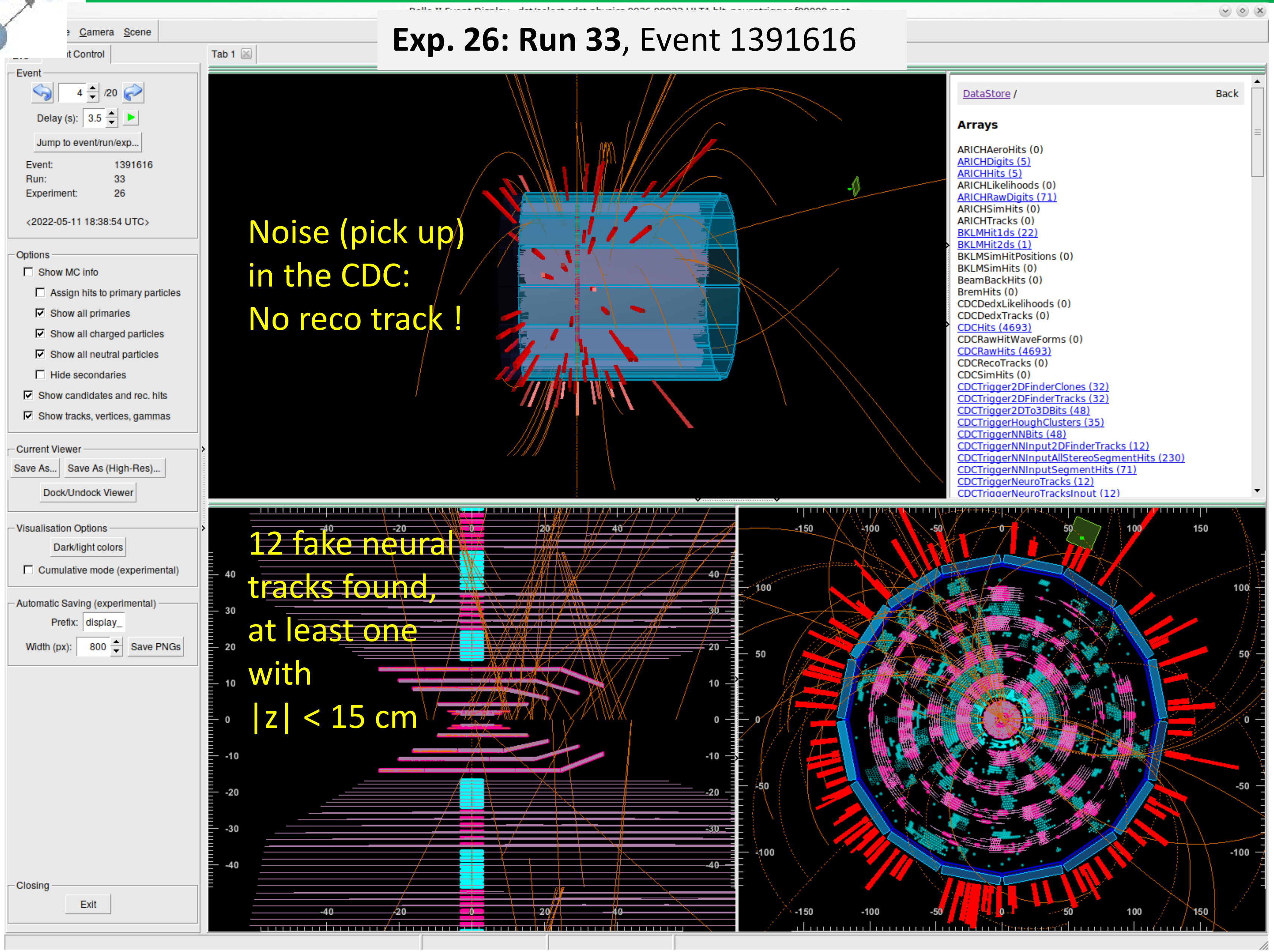
Large |z|:
a certain fraction of
tracks shifted into
IP region
-> increase of rate

Why are tracks
predicted around
IP while coming
from large |z| ?

feed-down especially strong for expert 4 (inner stereo SL missing)

Problems of the STT (II) : „Fake Tracks“

Exp. 26: Run 33, Event 1391616



Feed-Down and Fake Tracks have the same source:

Large number of fake 2D track candidates (require 4 out of 5 SLs), formed by „random“ noise in the CDC, mostly synchrotron radiation photons and electronic cross talk

Neural tracks formed by: noisy 2D track candidates & noise in the stereo layers

Upgrade Program for the STT

-> keep efficiency of STT & low dead time with rising luminosity (BG)

Physics goals: low charged multiplicity, e.g. τ 1-prong decays (- > τ EDM, LFV),

- $e^+e^- \to \pi^+\pi^-(\gamma)$ for g-2 (hadronic vaccum polarization) etc.
- quite generally: determination of lepton ID, tracking efficiency for the „other track“
- **STT is a minimum bias single track trigger**

New FPGA Hardware now available: „UT4 Board” with Virtex Ultrascale 160/190

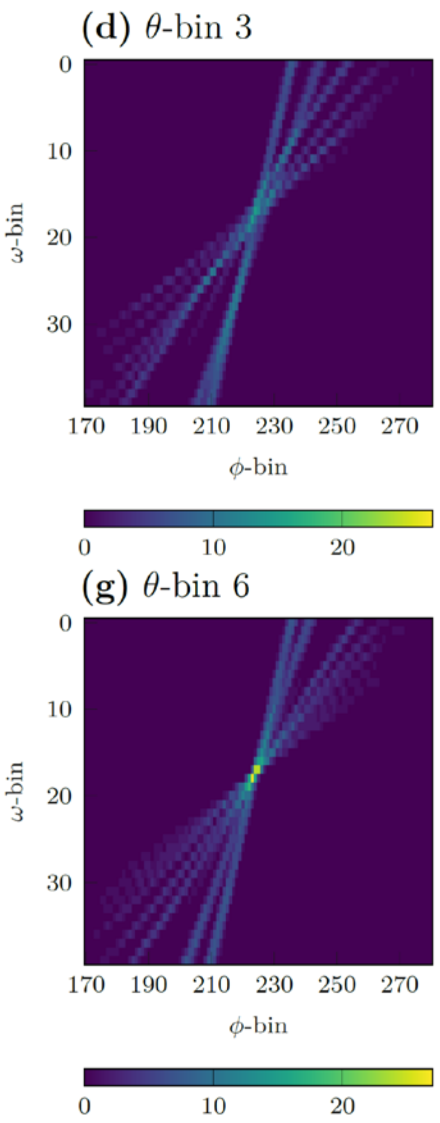
Improved track model for neural input / training algorithms:

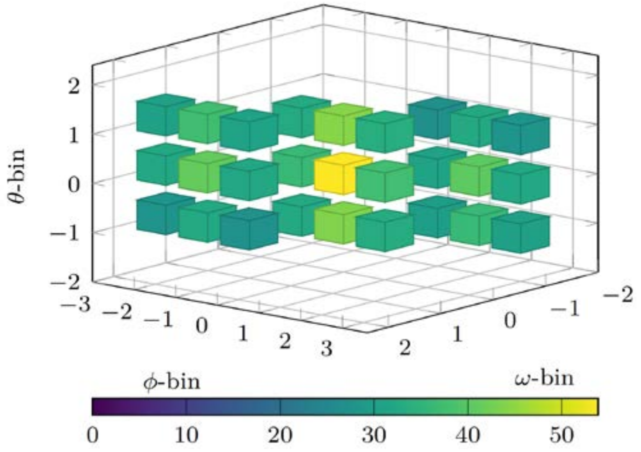
- **track finding in 3D Hough space** -> this is **really new** (S. Skambraks, S. Hiesl)
- network architecture: „deep-learning“ + additional inputs (T. Forsthofer)
- -> improve resolutions @ IP and for larger |z|
- -> reduce feed-down & fake tracks

FPGA Implementation:

- new algorithms on new UT4-Boards using hls4ml
- optimize latency: e.g. move STT decision to NN

3D track model





Summary and Conclusions

First Level 1 Neural Network Track Trigger, realized for the Belle II Detector, operational since Jan. 2021

- At least one Neural Track required to assert a track trigger

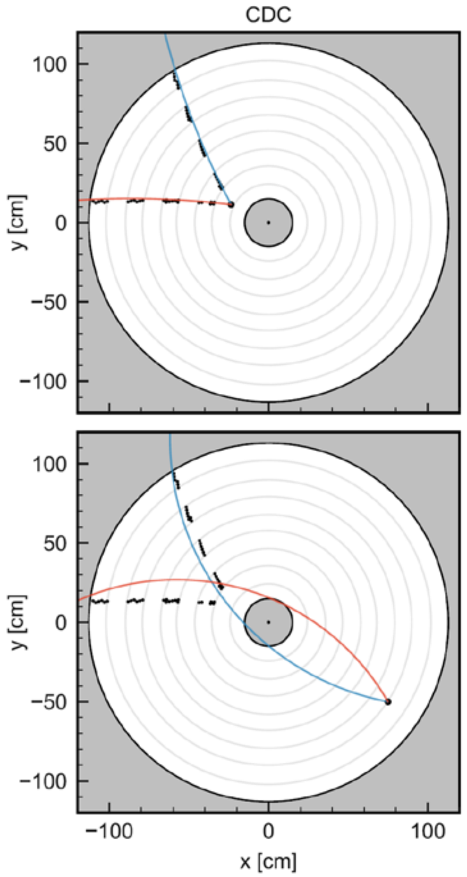
  -> Neural Nets: „working horses“ for Belle II track trigger system

- Minimum Bias Single Track Trigger (STT) shows excellent performance, even under severe background conditions
  However, “Feed-down” and “Fakes” need attention

- Upgrade: More powerful FPGA boards now available: Virtex UltraScale 7 XCVU160
  - track finding via 3D Hough cluster algorithm (novel method!)
  - additional inputs from all wires with the TSs
    (drifttime + coarse analog thresholds for CDC signals)
  - deep-learning neural networks for improved performance

- Commissioning by summer 2024, launch planned for the fall 2024 data taking

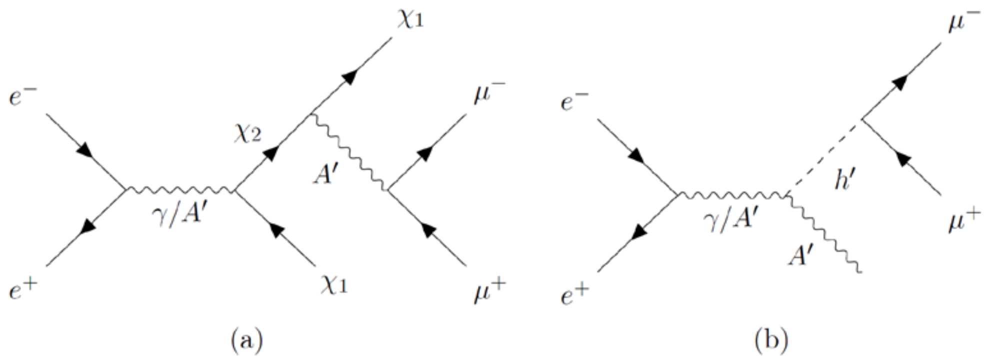
- New: Displaced Vertex Trigger on the horizon, commissioning planned end of 2024

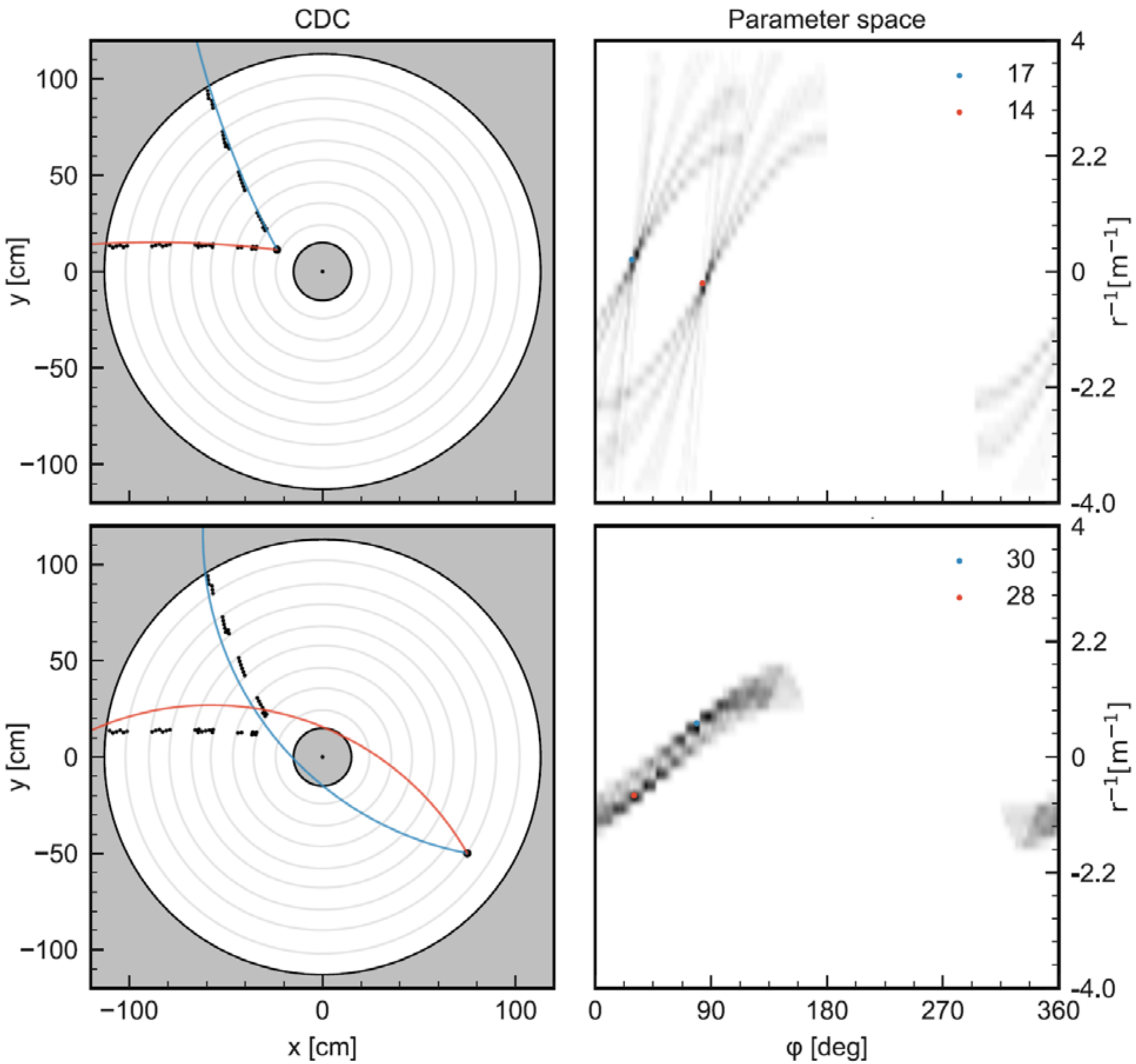


BACKUP

How to Trigger on feebly Interacting Neutral Particles

Example: Inelastic Dark Matter production
DM particles expected to be quite long-lived





Basic idea:

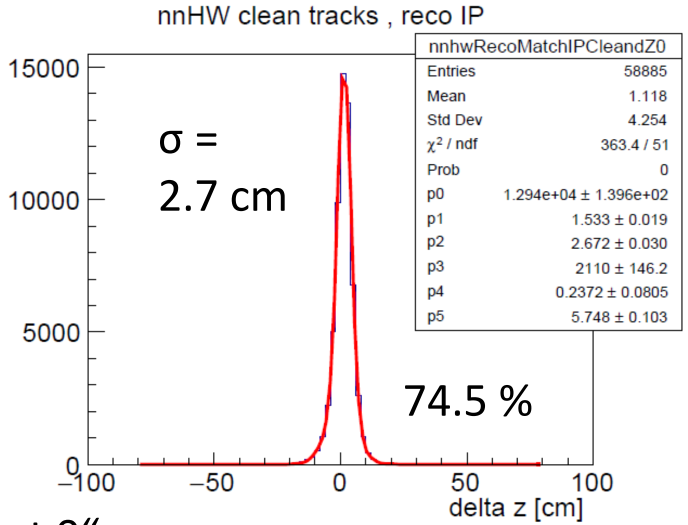
Divide the CDC axial wire planes into a set of „Macro Cells", serving as origins for the Hough transforms

FPGA:

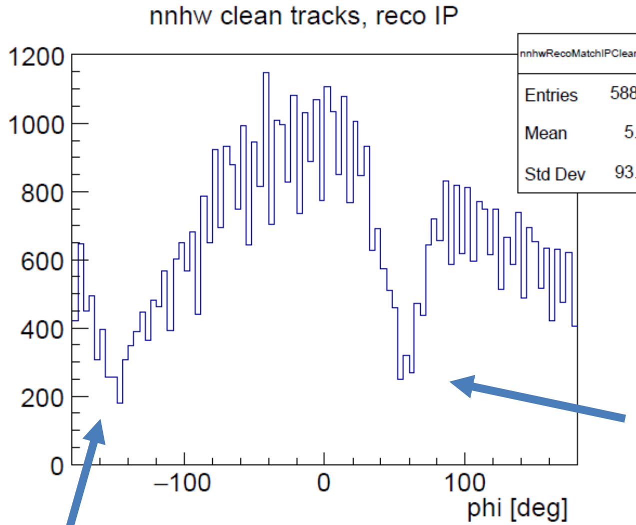
-> execute all Hough transforms with origins in each of the Mcro cells in parallel (typically of O(100))

-> use neural networks to determine „correct" vertex

z-Resolution for „Clean“ IP Tracks („Expert 0“)

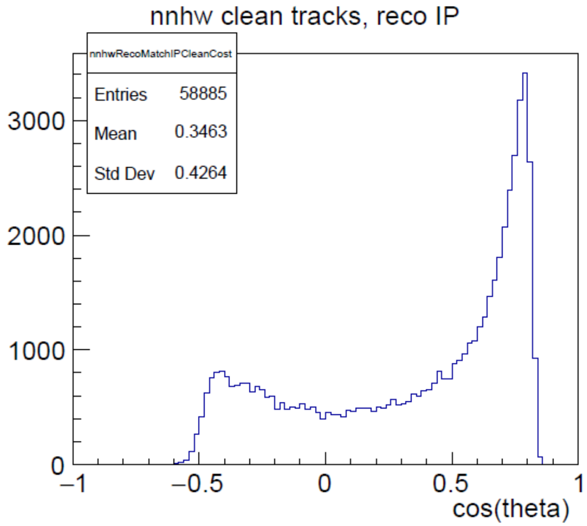


„expert 0“



known problem

new CDC inefficiency



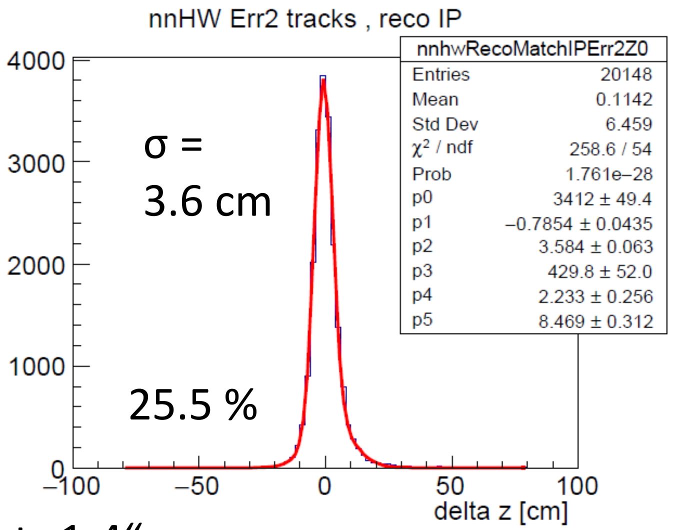
Gaussian fits to neuro tracks associated with reco tracks from IP (|z|<1 cm, d < 1.5cm)

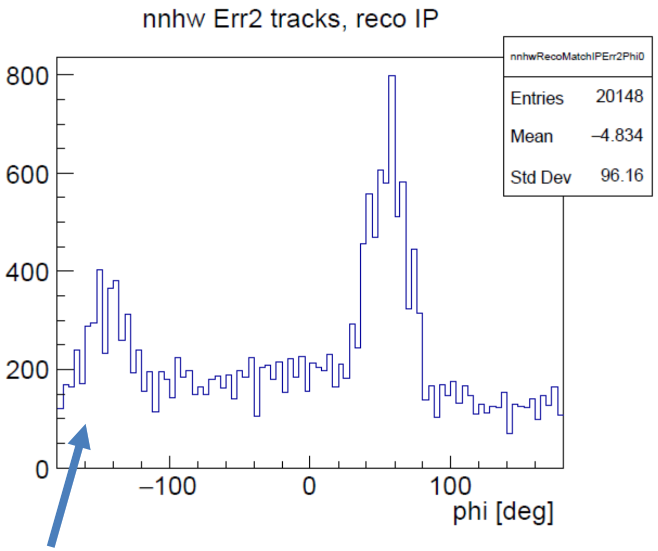
Central Gauss: σ = 2.7 cm

2nd Gauss: σ = 5.7 cm (14.1 %)

Exp 24, runs 790 - 890, New FW STT active, zcutTrig = 20 cm 19/03/2022 50

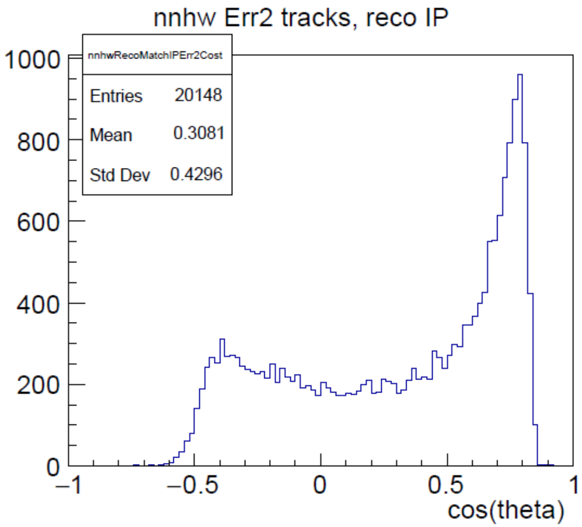
z-Resolution for IP Tracks („Experts 1-4“)





„experts 1-4“

new CDC inefficiency



Gaussian fits to neuro tracks associated with reco tracks from IP (|z|<1 cm, d < 1.5cm)

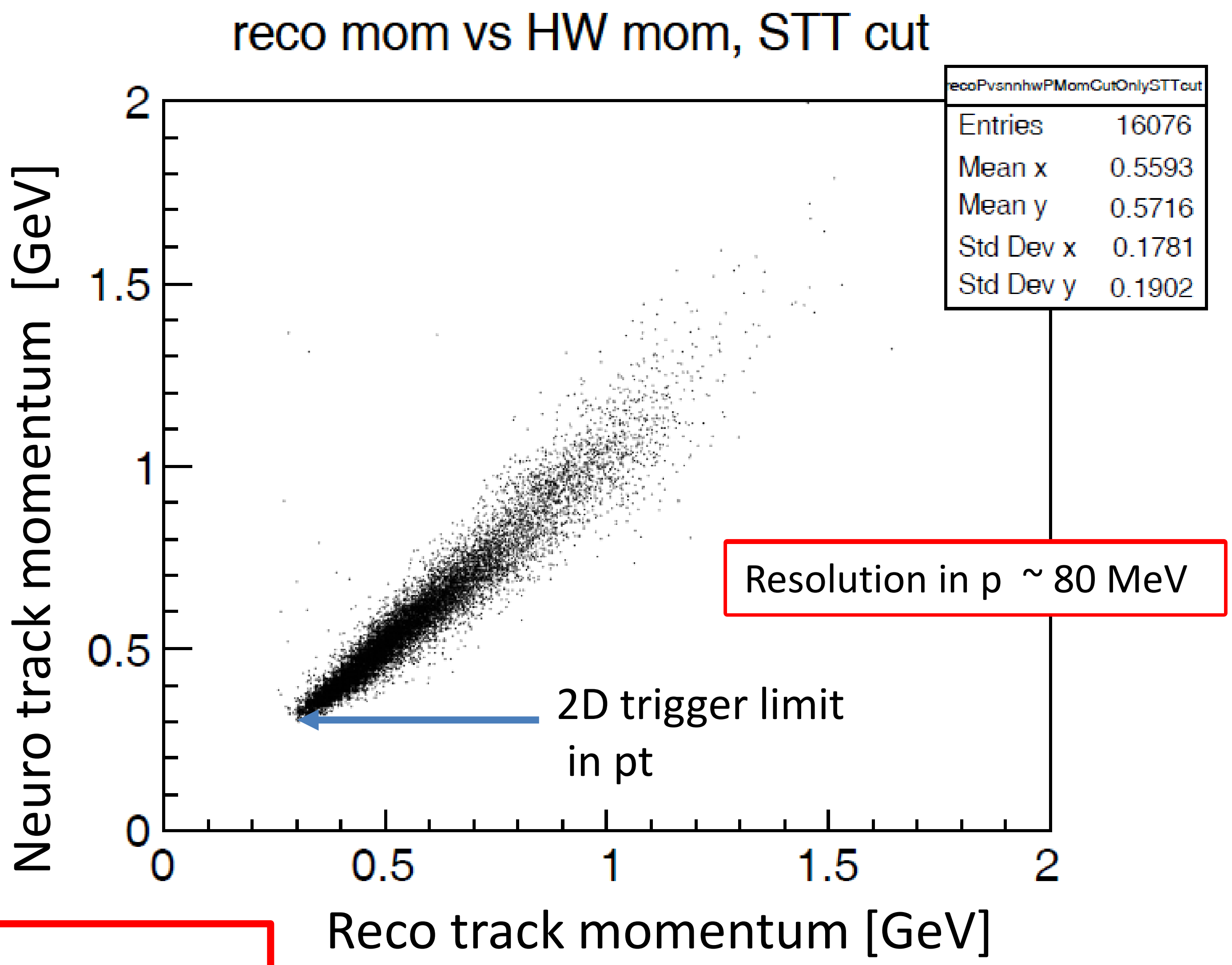
Central Gauss: σ = 3.6 cm

2nd Gauss: σ = 8.5 cm (11.2 %)

Exp 24, runs 790 - 890, New FW   STT active, zcutTrig = 20 cm   19/03/2022 51

Reducing STT Trigger Rate: Neuro Track p

momentum correlation of neuro tracks and reco tracks

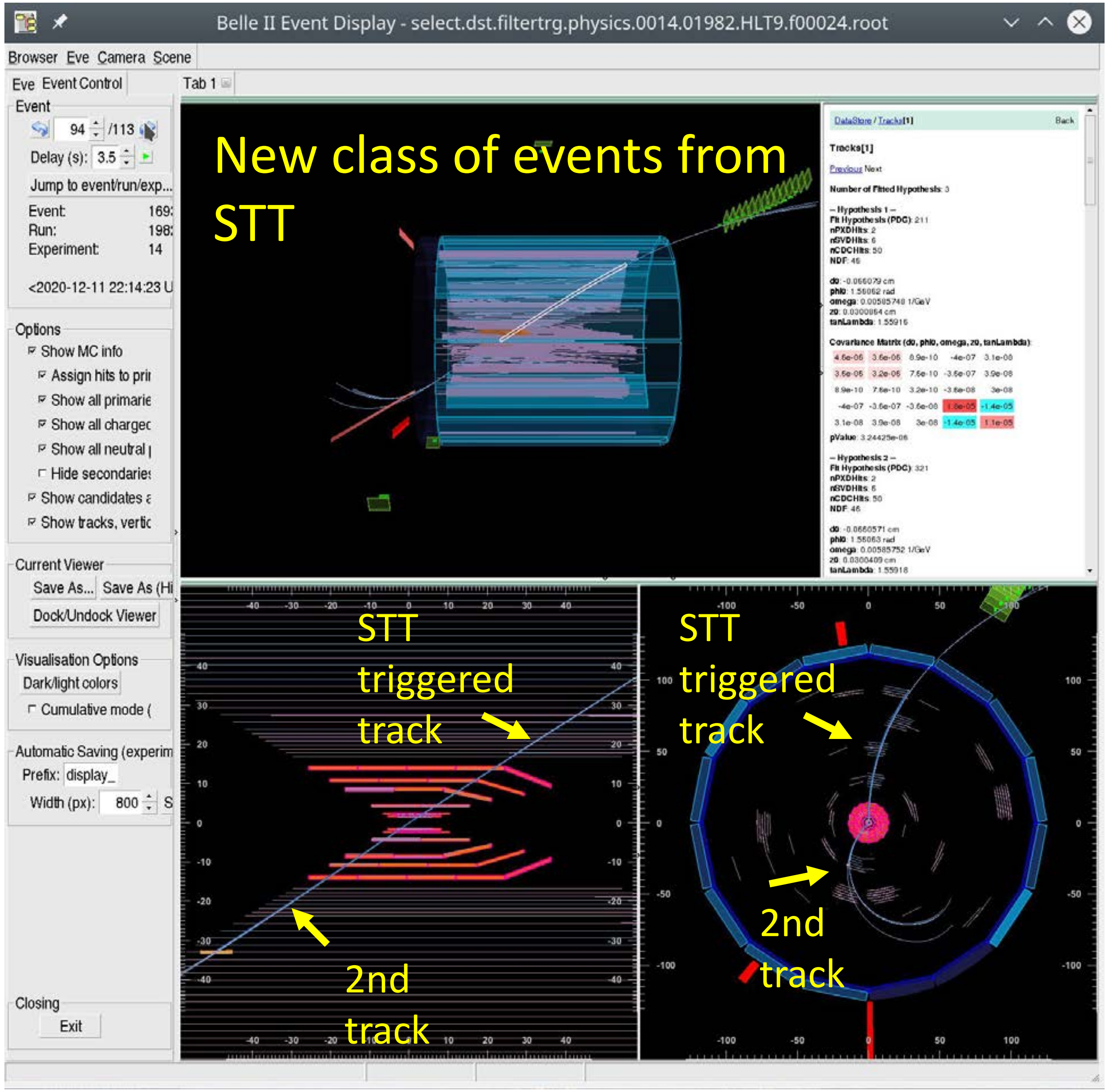


calculate the neuro track momentum:

$$p[GeV] = \frac{1}{|\omega|[1/m]\sin(\theta)} 0.3B[T]$$

ω ~ transverse momentum (from 2D track)

STT Triggers ONLY



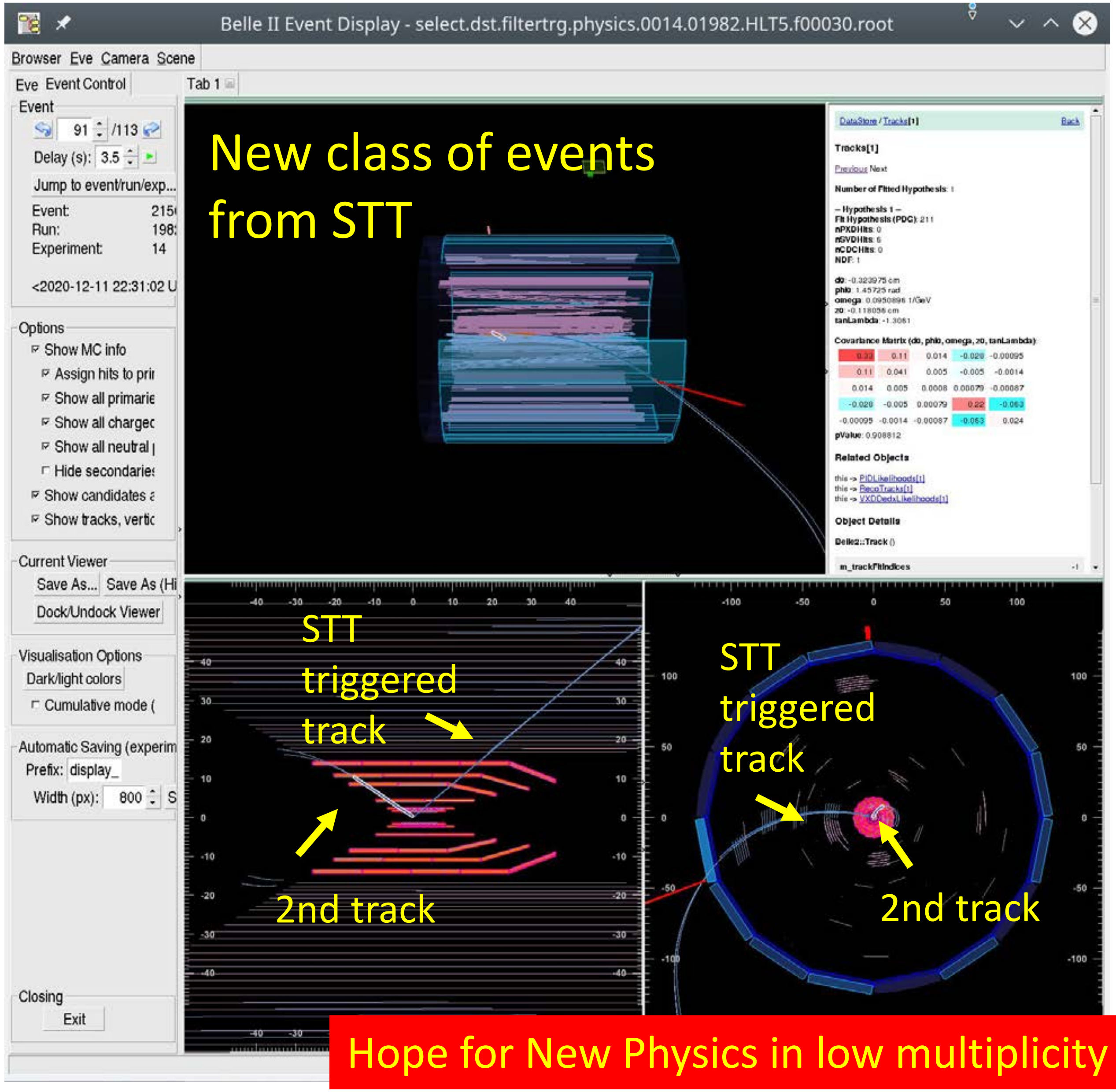
Event display shows the reco tracks

2nd track at shallow ϴ cannot be seen by CDC trigger

Note:
2nd track is unbiased (can be anywhere in the detector)

Event class only triggered by STT (~12% of STT events)

STT Triggers ONLY



2nd track reconstructed only in PXD/SVD

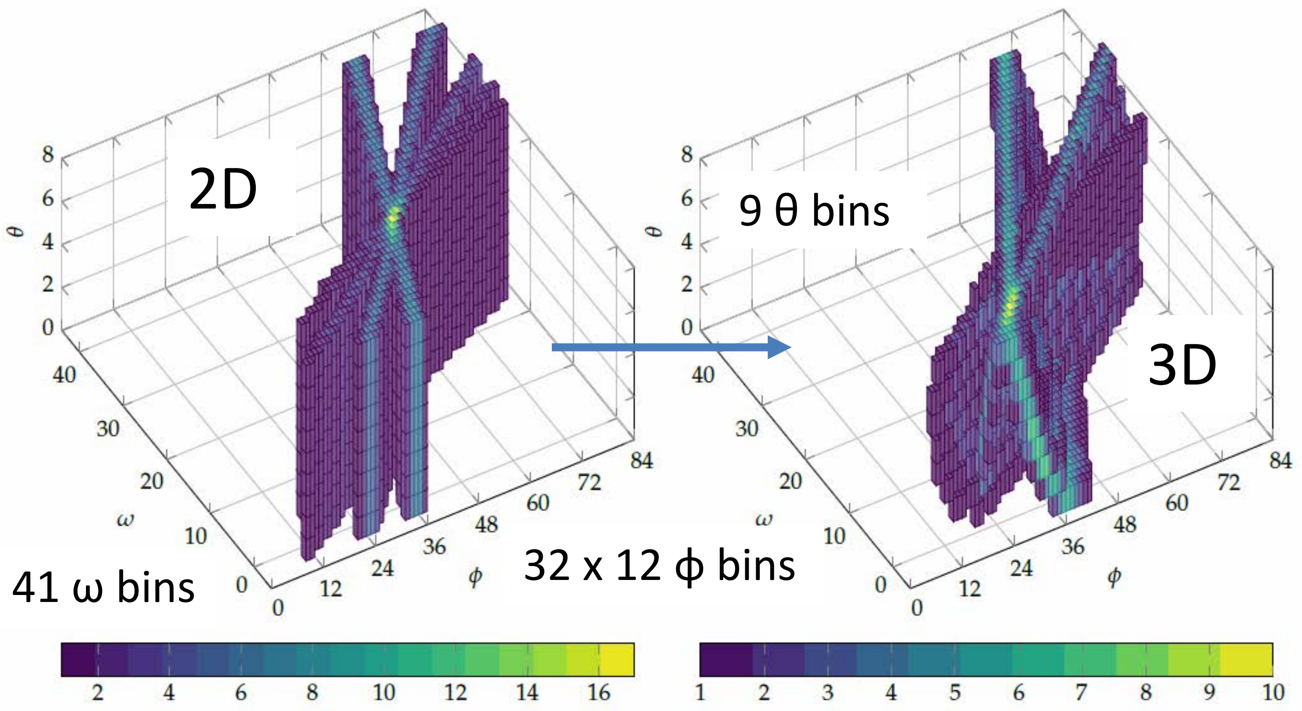
Caution:

efficiency of STT not easy to calculate from data since no other orthogonal trigger (e.g. ECL) available

Hope for New Physics in low multiplicity final states ?

3D Hough Track Finding

- Extend traditional 2D ($\omega=1/p_T$,ϕ=azimuth angle) Hough space by a third dimension, the (binned) polar angle θ
- For track finding use axial and stereo track segments (->3D)
- Peak finding in 3D Hough space



Main advantages:

- more TS (9 vs 5)
  -> suppress fakes
- No need to choose STS by min drift time
  -> find „correct“ STS
- Force track model to originate from IP
  -> suppress candiates far from IP
- 3D track candidates come with θ estimate,
  -> improve z resolution

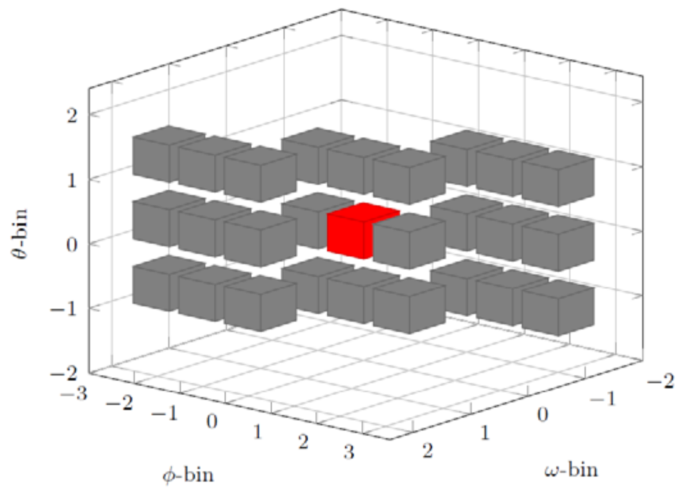
Clustering Algorithm in 3 Dimensions

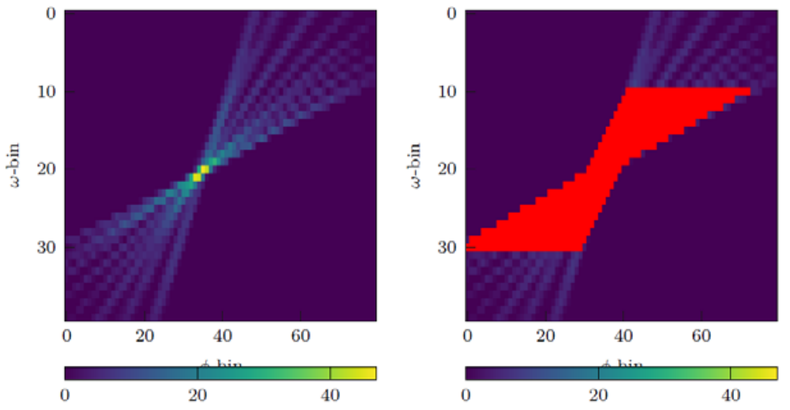
Original algorithm: DBSCAN $\rightarrow$ Difficult to implement on an FPGA (non-deterministic length $\implies$ latency not fixed)

Update: **Fixed Clustering**
Three steps, repeated `iterations` times:

- Step 1: Global maximum search on Hough space
- Step 2: A fixed shape is put around the maximum
  - The weights in this shape are added up (total weight)
  - If total weight $\geq$ `mintotalweight` and peak weight $\geq$ `minpeakweight` the cluster is saved
  - All hits (TS) are extracted and have to pass two TS cuts
- Step 3: Cells around the global maximum are set to zero ("Butterfly-Shape" cutout)

Fixed shape:





(a) Complete Cluster (b) Cutout

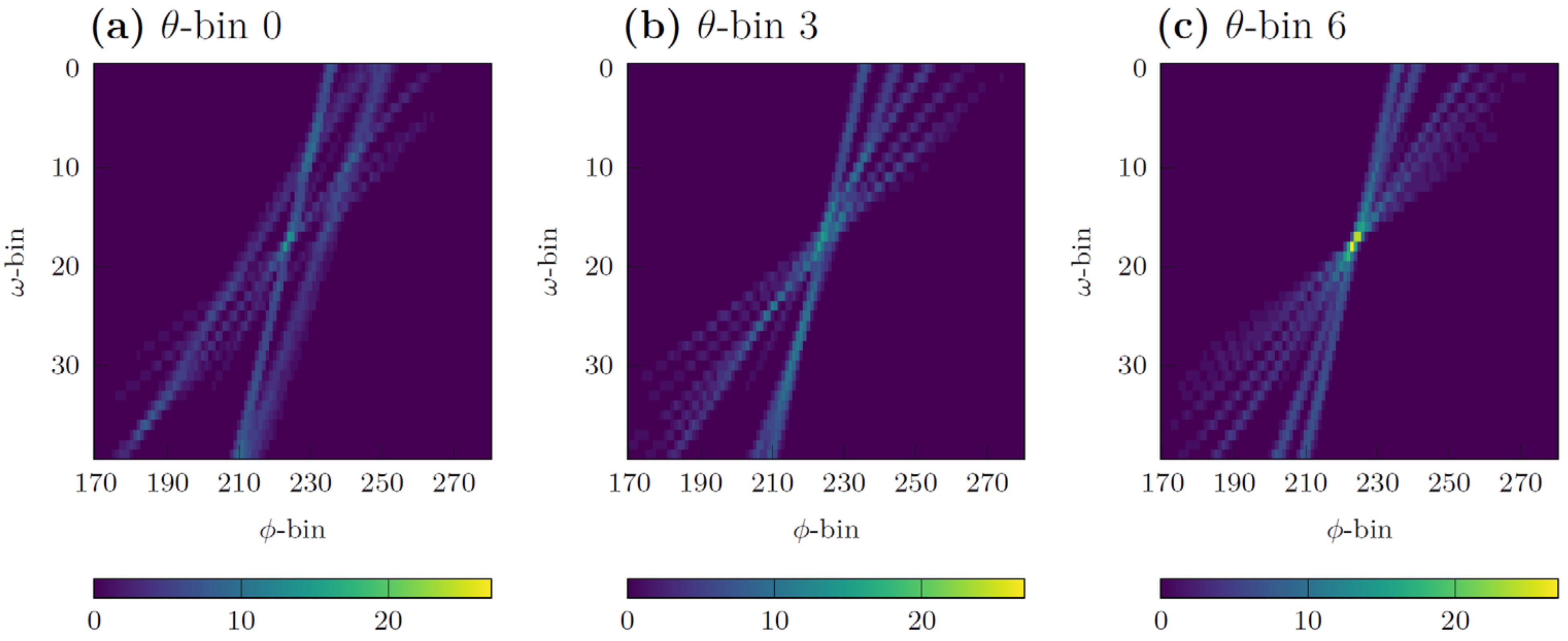
Simon Hiesl (MPI & LMU)

Extension to 3D: The NDFinder

New curve parameter: Polar angle $\theta \implies$ 3D-Hough space

- 9 bins in $\theta \in [19, 140]^\circ$, 384 bins in $\phi \in [0, 360]^\circ$, 40 bins in $\omega \propto q \cdot p_T^{-1}$, $p_T \in [0.25, 10]$ GeV/c

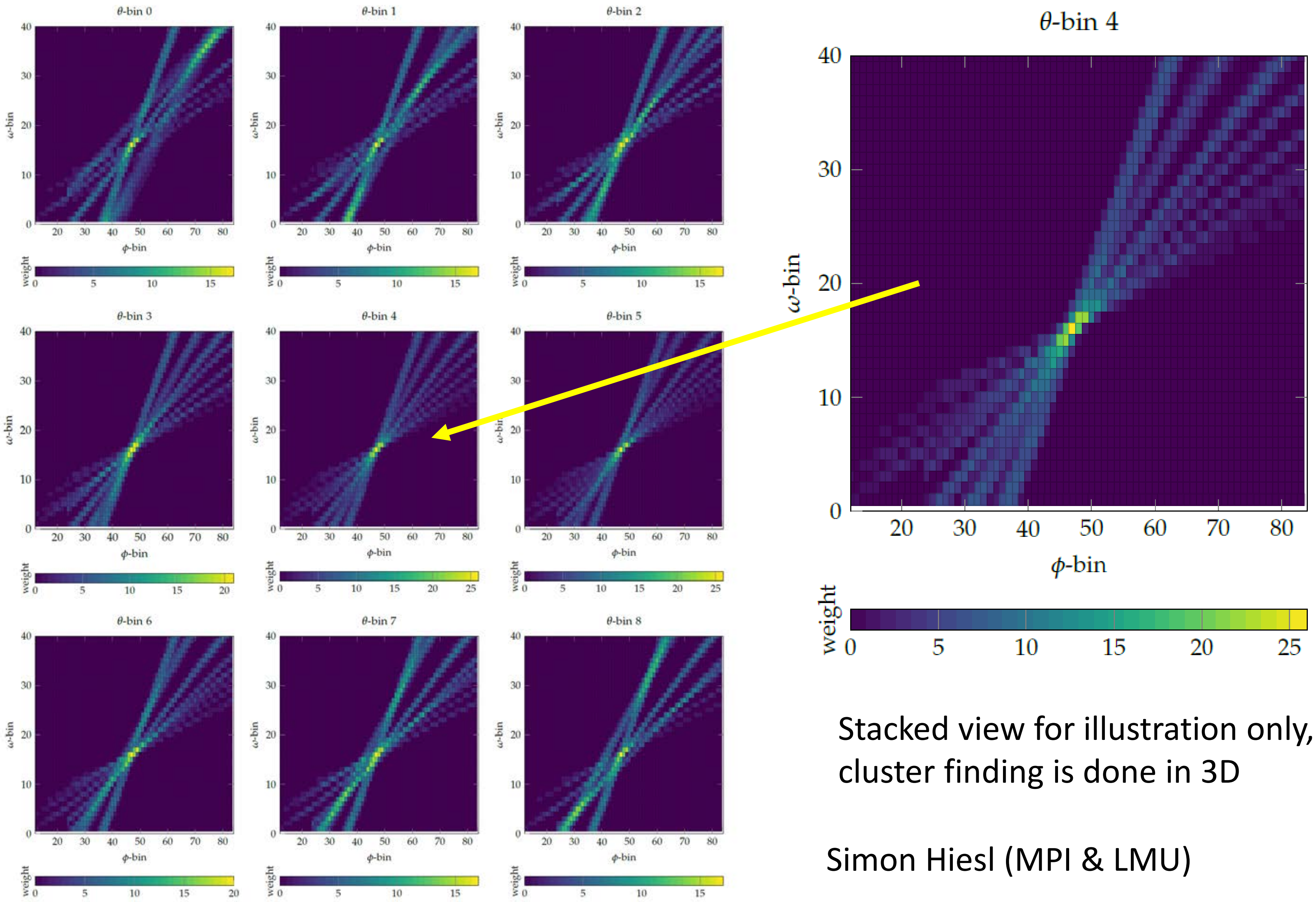
Vertex assumption: The track originates from $(x, y, z) = (0, 0, 0)$ (IP)



$\implies$ Intersection point yields ω, ϕ and θ

Simon Hiesl (MPI & LMU)

Example of 3D Hough Map

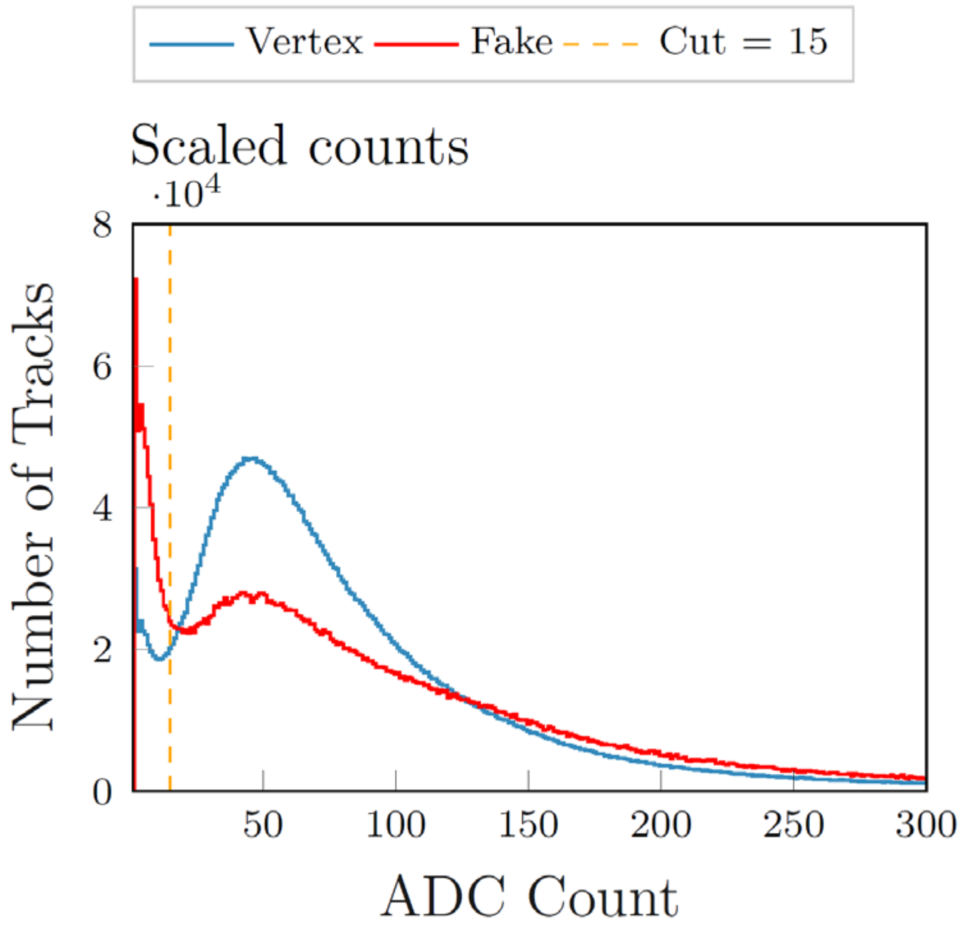


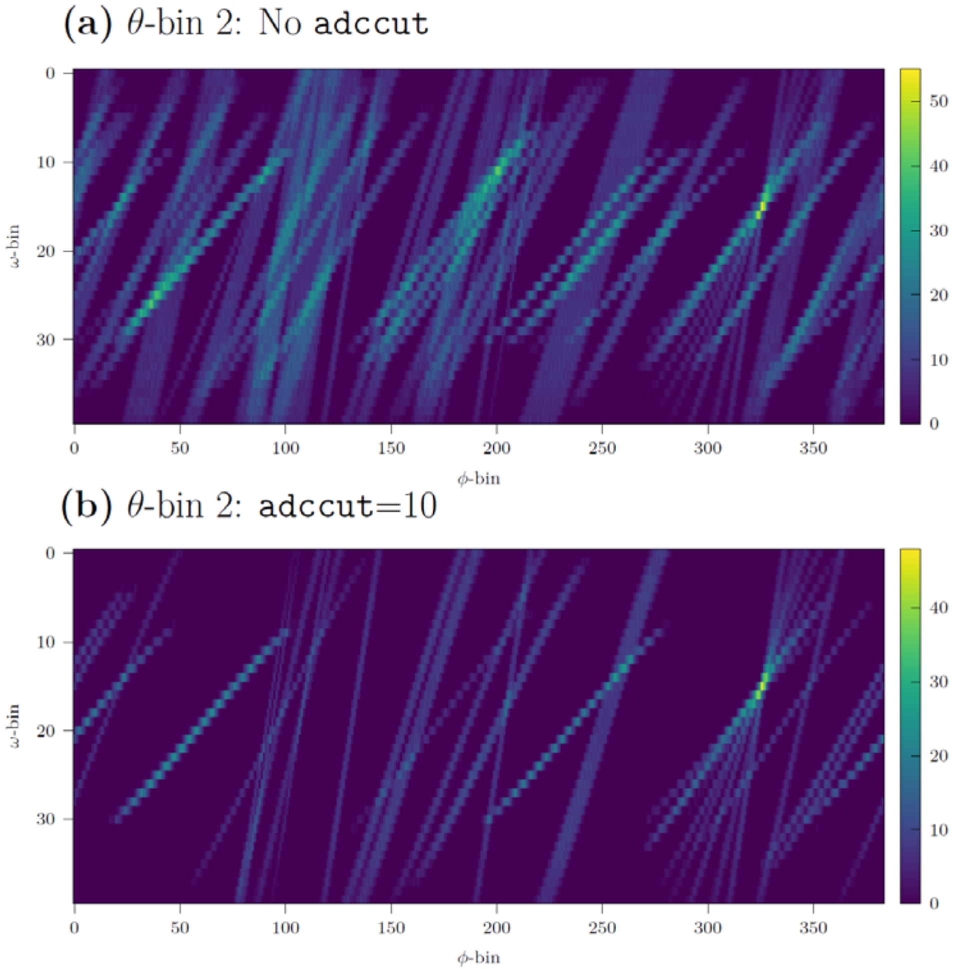
Stacked view for illustration only, cluster finding is done in 3D

Simon Hiesl (MPI & LMU)

Real Data Analysis

- Very high backgrounds were observed in the last experiment (due to high luminosity)
- The Hough spaces contain a lot of background track segments





Simon Hiesl (MPI & LMU)

$\Longrightarrow$ Reduction of noise using a cut on the ADC count

Efficiency on Real Single Track Events

Simon Hiesl (MPI & LMU)

- Hit to cluster relation:
  - All hits in a cluster are considered
  - The largest weight distribution for each SL is used
- Cut on the number of axial and stereo SL hits (for background reduction)

Efficiency for single track events: Cut at $\pm 10\,\mathrm{cm}$

`adccut`	Efficiency 3D	Efficiency 2D
No Count	94.1%	94.0%
10 Counts	96.3%	95.3%

Fake-Rate for all found tracks:

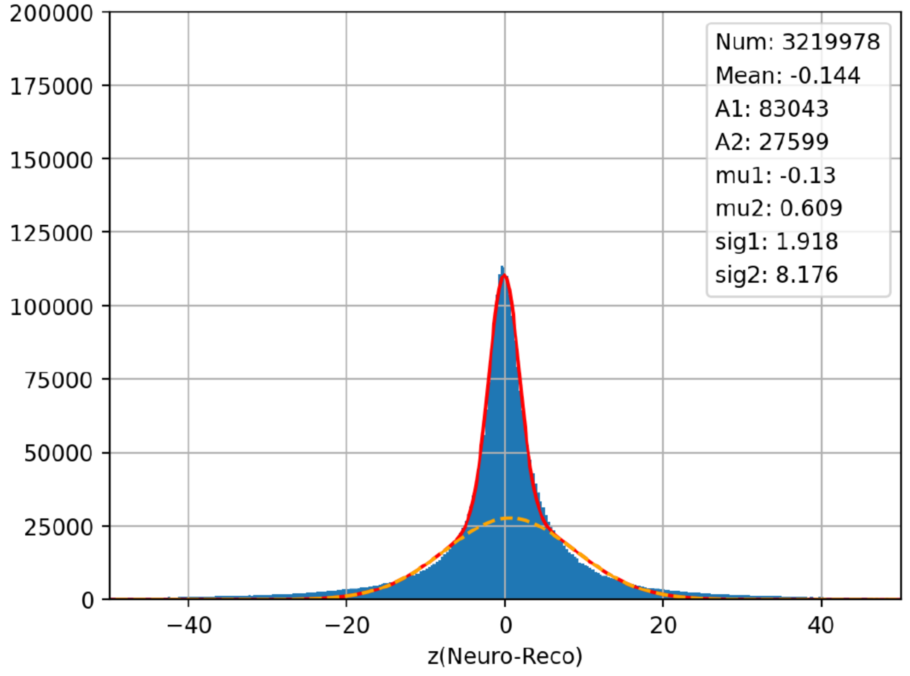
`adccut`	Fake-Rate 3D	Fake-Rate 2D
No Count	13.1%	31.6%
10 Counts	5.8%	13.5%

But: Neural network not trained for 3D candidates at the moment (see presentation by Timo Forsthofer)

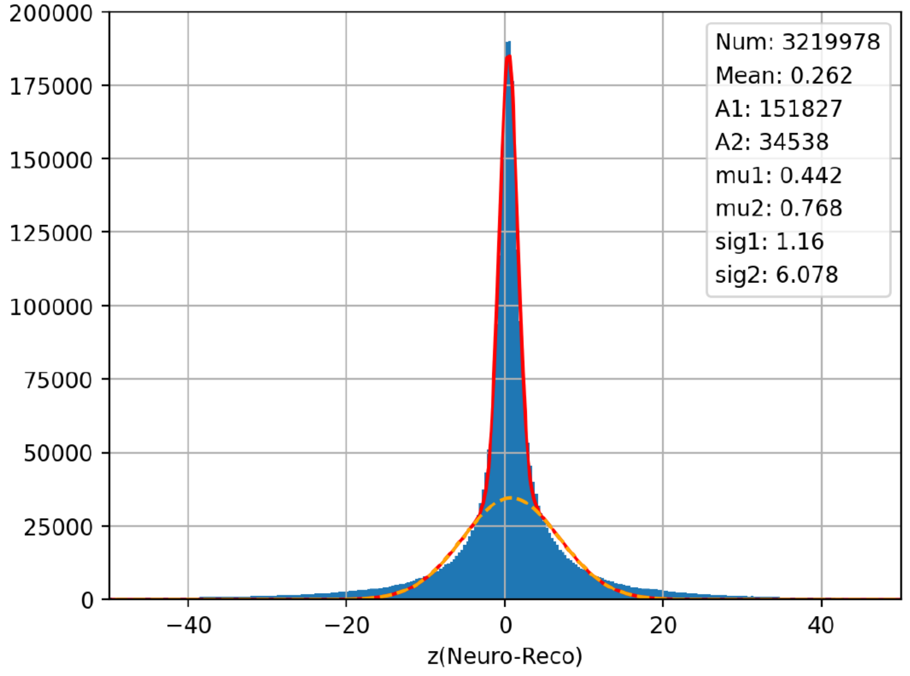
Deep Learning Architectures

- New, more powerful FPGAs allow for bigger networks
- Three or four hidden layers beneficial for resolution
- More hidden layers better than more nodes per layer

Timo Forsthofer
(MPP & LMU)



1HL with 81 Nodes

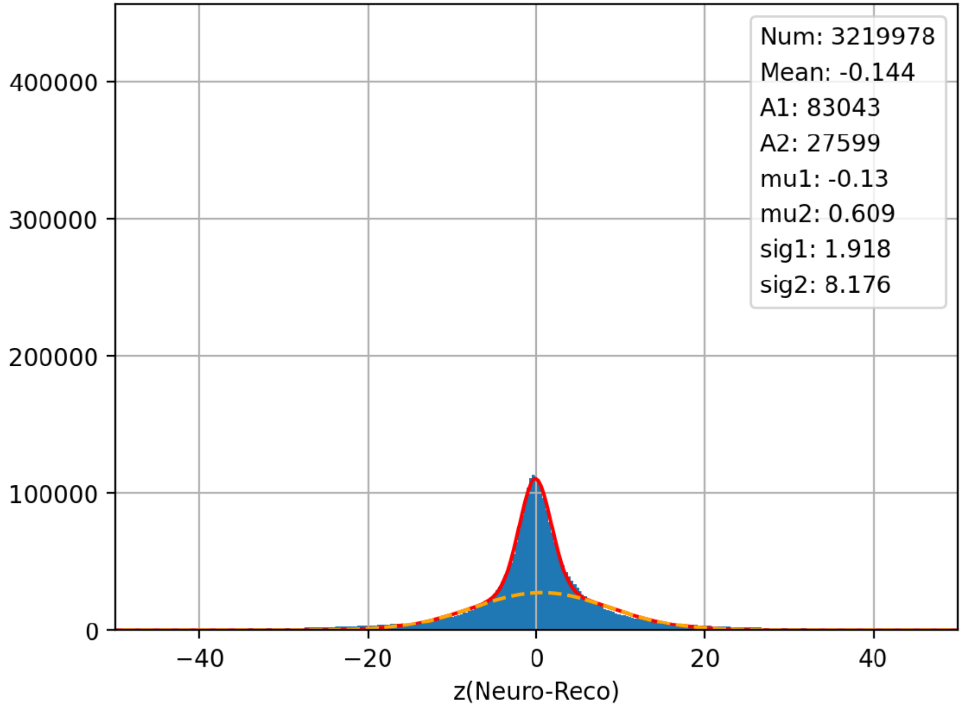


4HL with 100 Nodes per HL

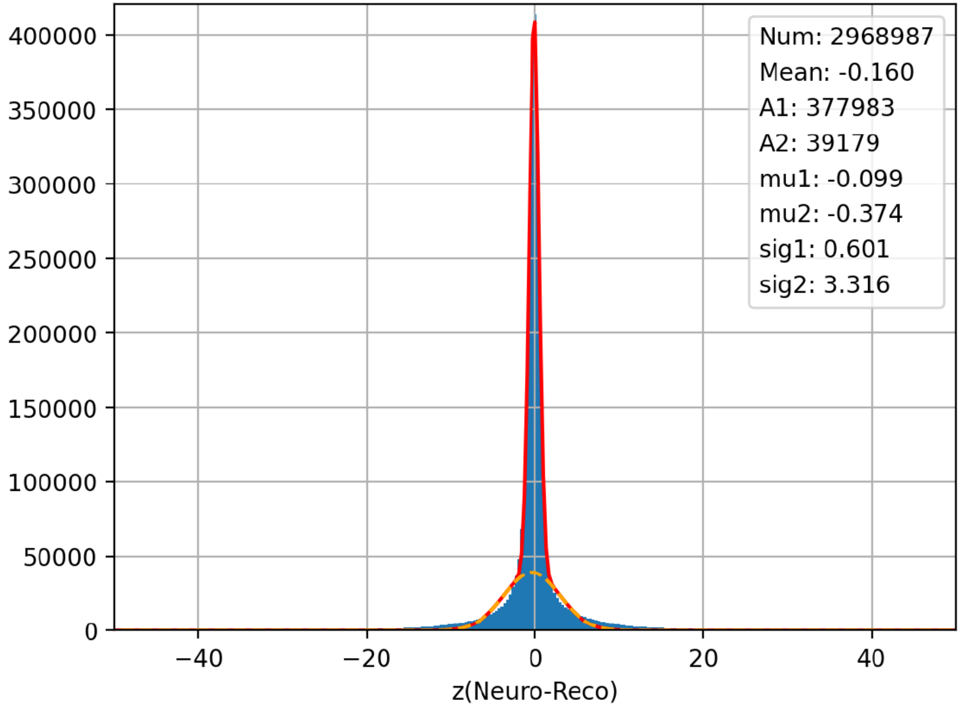
Final Performance Evaluation

Timo Forsthofer
(MPP & LMU)

- Combination of all advances leads to increase in accuracy by almost a factor of three
- z-Cut can be reduced from 15cm to under 10cm



Present Network Architecture

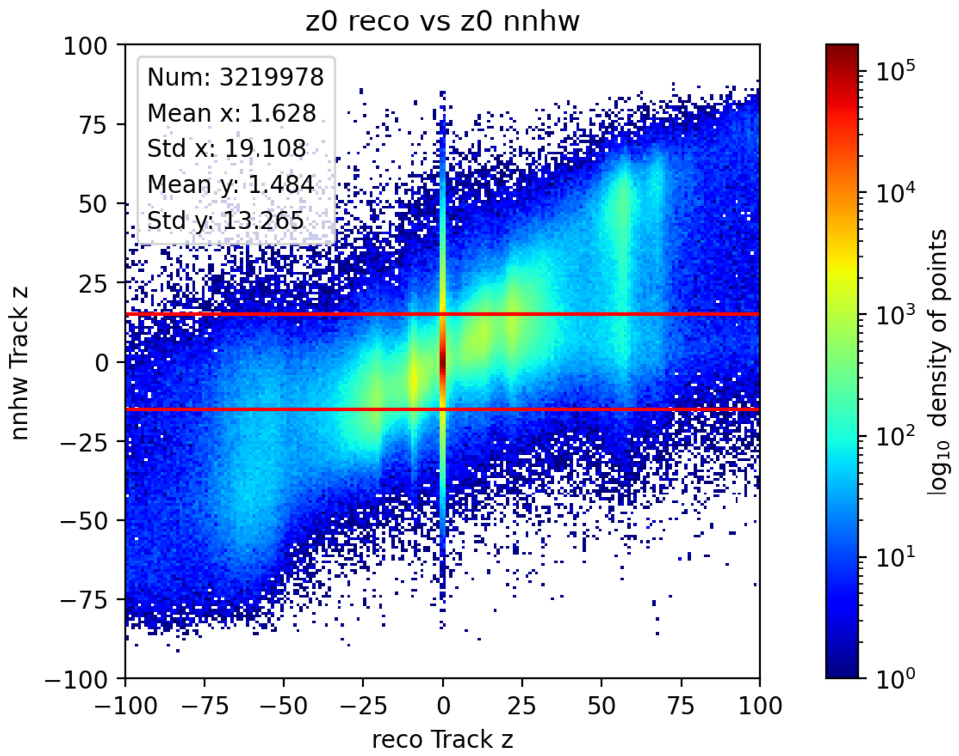


Deep Neural Network with Extended Input, ADC-cut and 3D-Input

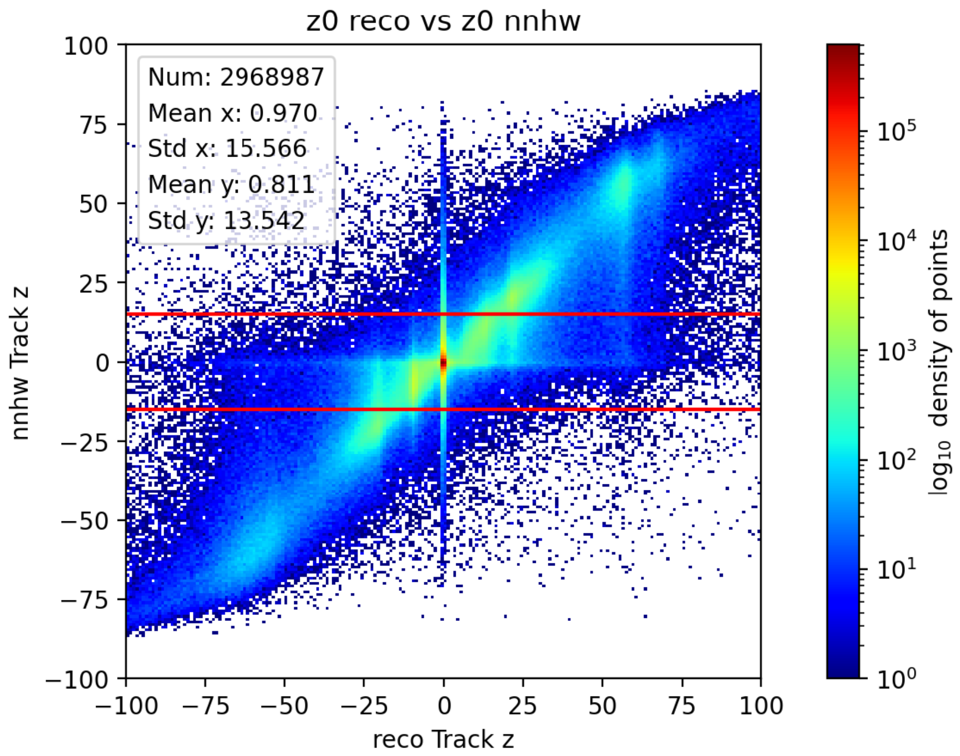
Final Performance Evaluation

Timo Forsthofer
(MPP & LMU)

- Especially extended input helpful in reducing Feed-Up and Feed-Down



Present Network Architecture



Deep Neural Network with Extended Input, ADC-cut and 3D-Input

Combining ADC-Cut and 3D-Hough-Finder

Timo Forsthofer
(MPP & LMU)

- ADC-Cut works well with 3D-Hough Finder (see presentation by Simon Hiesl)
- 3D-Hough Finder already rejects a lot of background and fake tracks, so the performance is underrepresented here