JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

Bundesministerium für Bildung und Forschung

HIC for FAIR
Helmholtz International Center

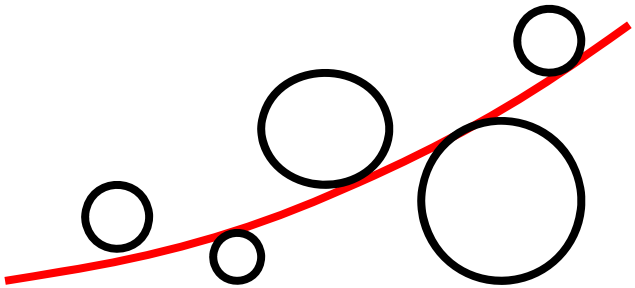
FPGA based online tracking at $\bar{P}$ANDA

Sören Lange

Universität Gießen
FAIR Event Reconstruction Workshop, 29.10.2012, GSI Darmstadt

Introduction

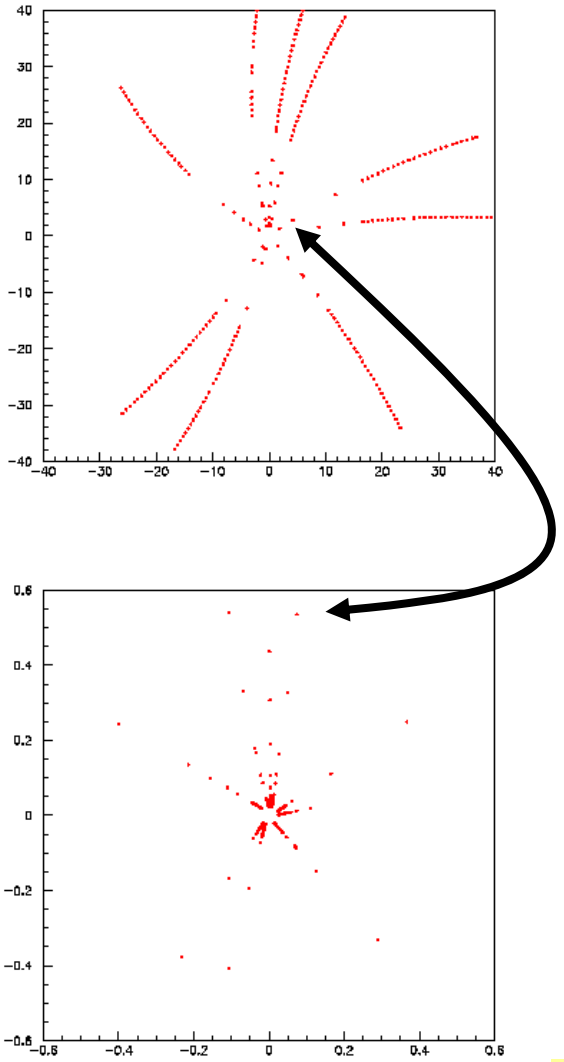
- Requirements:
  - helix in xy plane
  - hits are not points, but drift circles,
    helix is tangent to circles
  - should run on ATCA Compute Node
    (Virtex FPGA, histograms and lookup tables)
- Project started in 2008,
  diploma thesis David Münchow (Giessen)
  indico.uni-giessen.de/indico/getFile.py/access?contribId=5&resId=0&materialId=slides&confId=39
  based upon (x,y) hits only (no drift time)
- at same time (fruitful synergy):
  - conformal map in PandaRoot
    (track finding pre-step for genfit Kalman filter)
  - Belle II PXD
    adaptive Hough transform as ROI finder



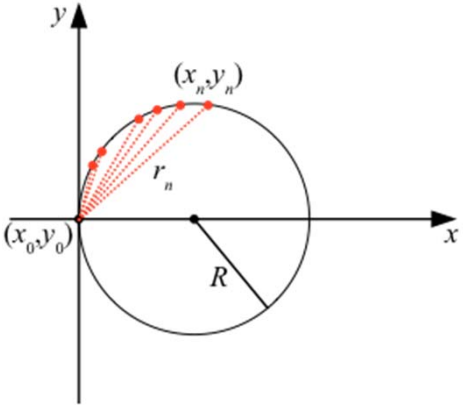
- Development in C++ and VHDL
  C++ „emulator“ for determination of efficiency and resolution (MC truth)
  C++ standalone code was distributed later

Step #1: Conformal Map

- transform circles into straight lines
- inverse space:
  MVD hits provide strong constraint
  (“lever arm”)
- radius in denominator is squared
  (sqrt() not required)



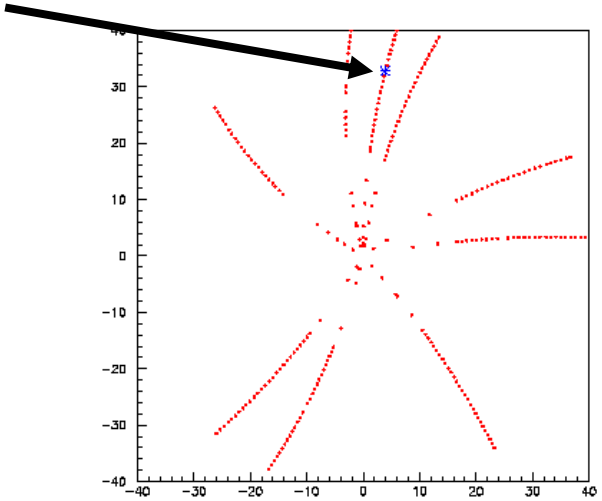
$$x' = \frac{x - x_0}{r^2}$$
$$y' = \frac{y - y_0}{r^2}$$
$$r^2 = (x - x_0)^2 + (y - y_0)^2$$



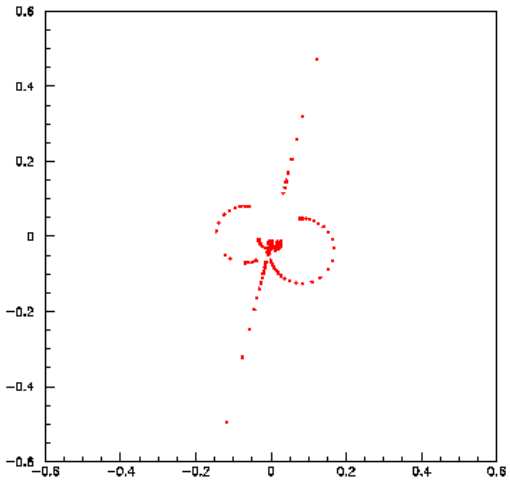
(x_0, y_0) here: (0,0)

Conformal Map: Secondary Vertices

(x_0, y_0)

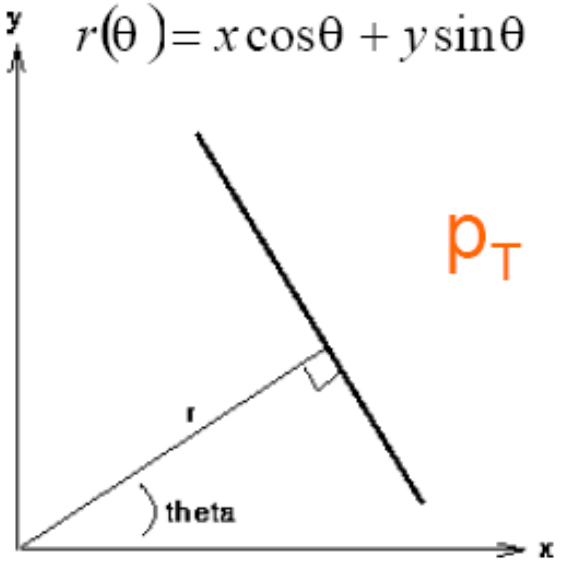


- Arbitrary (x_0,y_0)
- Track containing (x_o,y_o) gives line
- all other tracks give circle
  (Hough transform only finds lines)
- re-do complete procedure
  for all (x_o,y_0)
  (in parallel on FPGA)



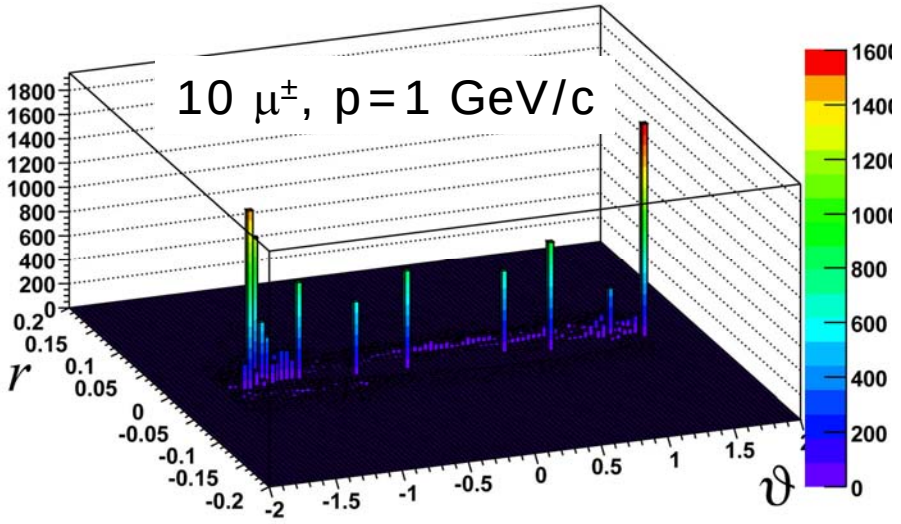
Step #2: Hough Transformation

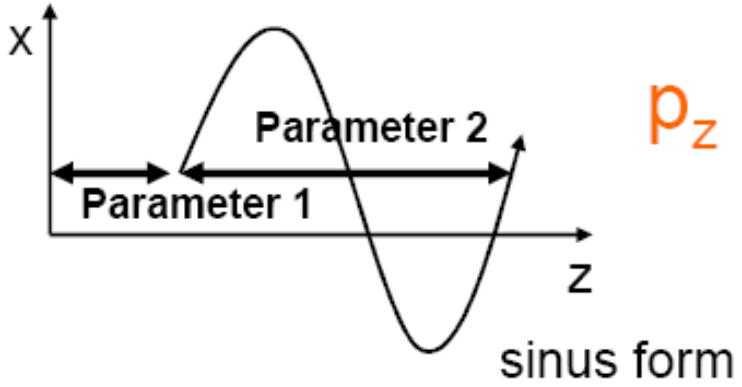
Hough space is filled with conformal coordinates



$r(\theta) = x\cos\theta + y\sin\theta$

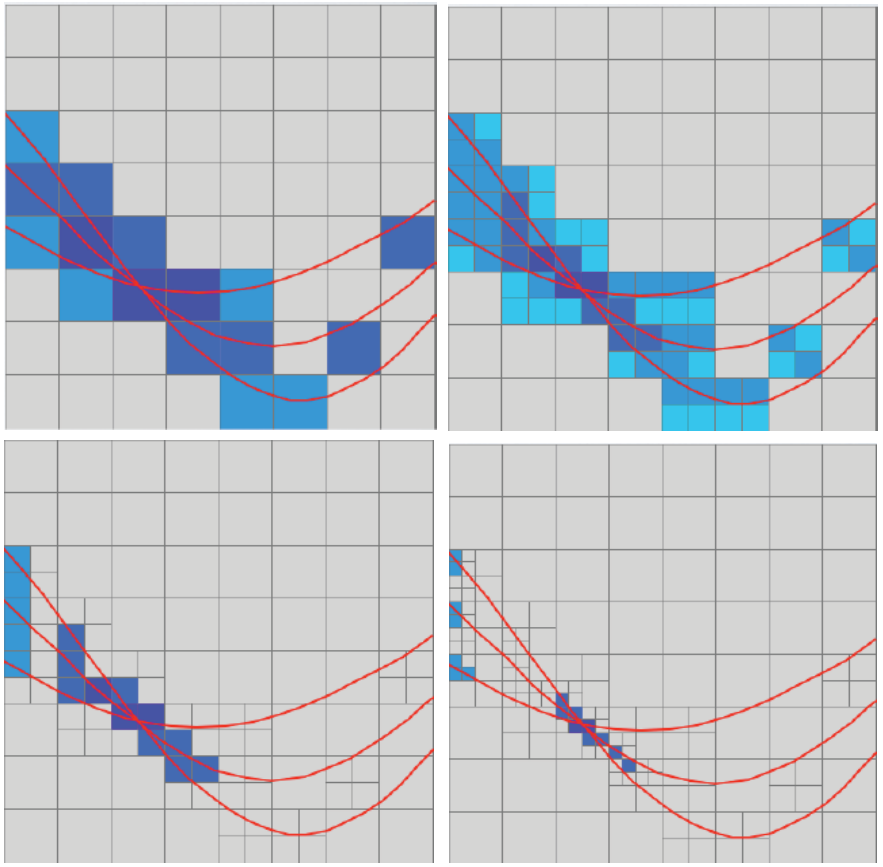
p_T





p_z (not explained today)

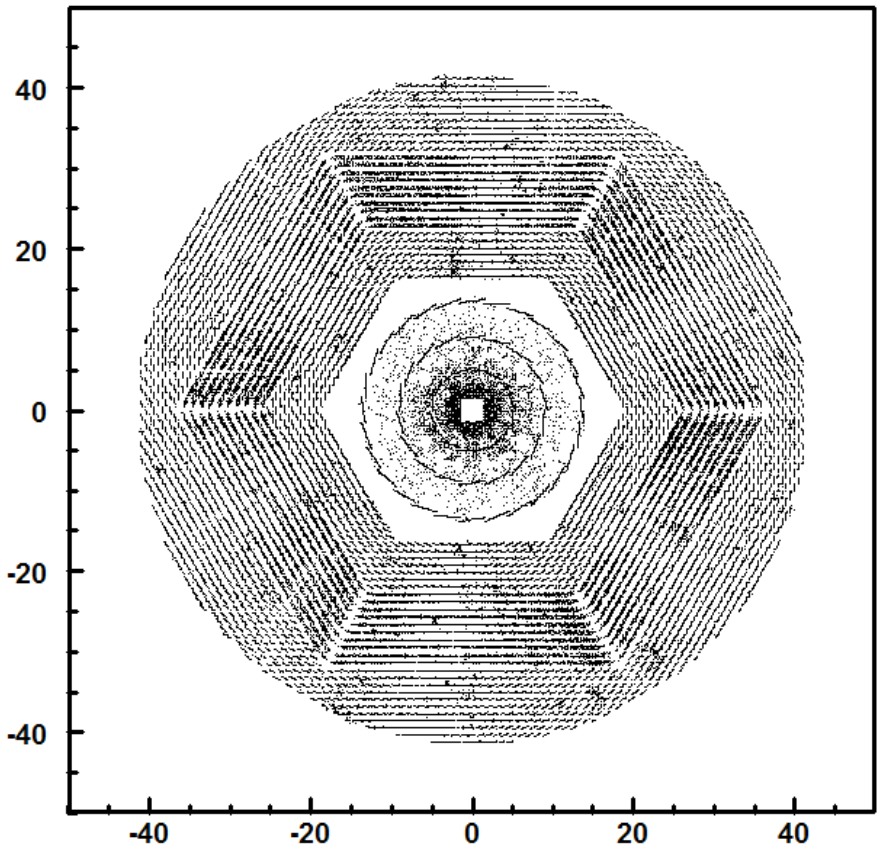
Adaptive Hough Transform



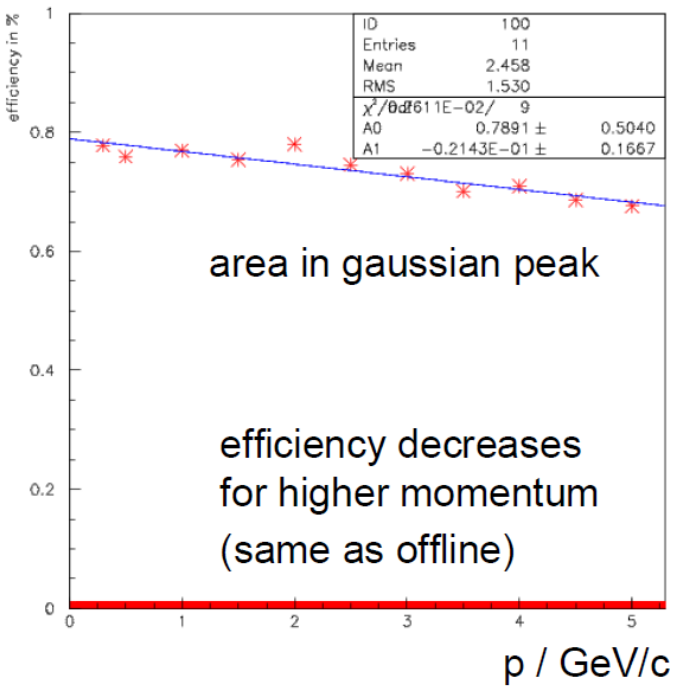
- avoids peak finder
- requires sorting
  (keep the best 8 bins
  for the next step)
  but sorting is unclocked
- achieved speed-up
  factor ~126

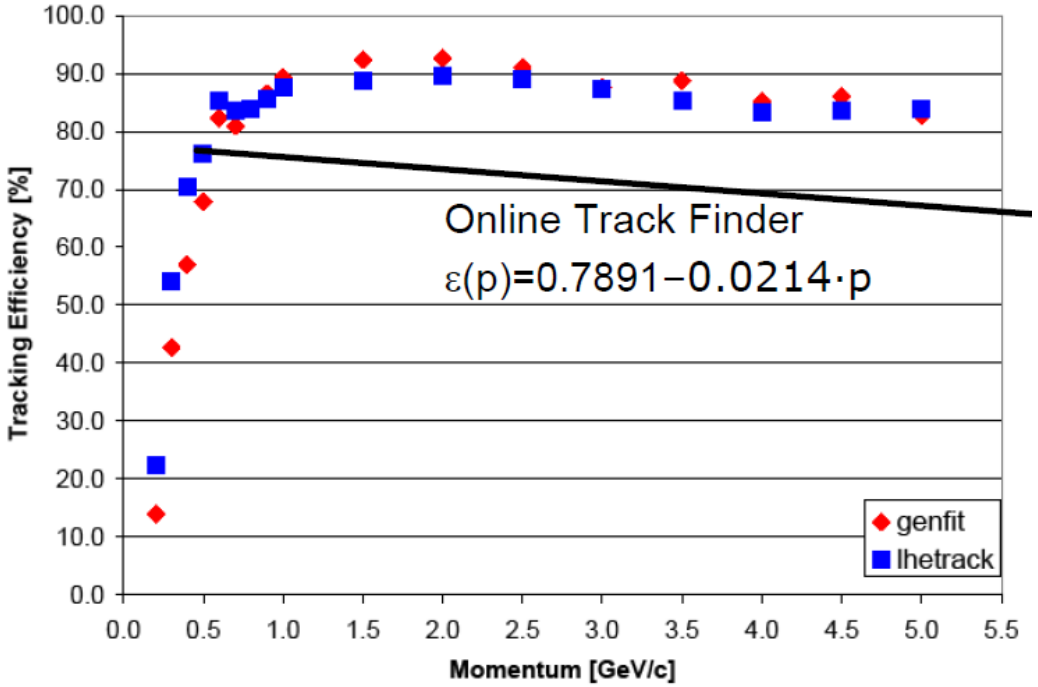
Idea for Belle II PXD
by C. Heller (MPI Munich)

Test data (MC), xy digis, p=1 GeV/c
z coordinate by pre-fitter



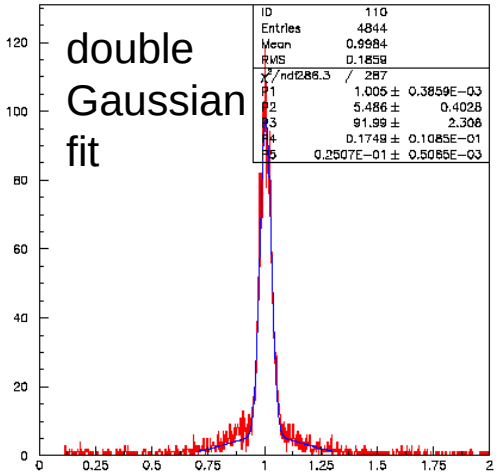
Efficiency

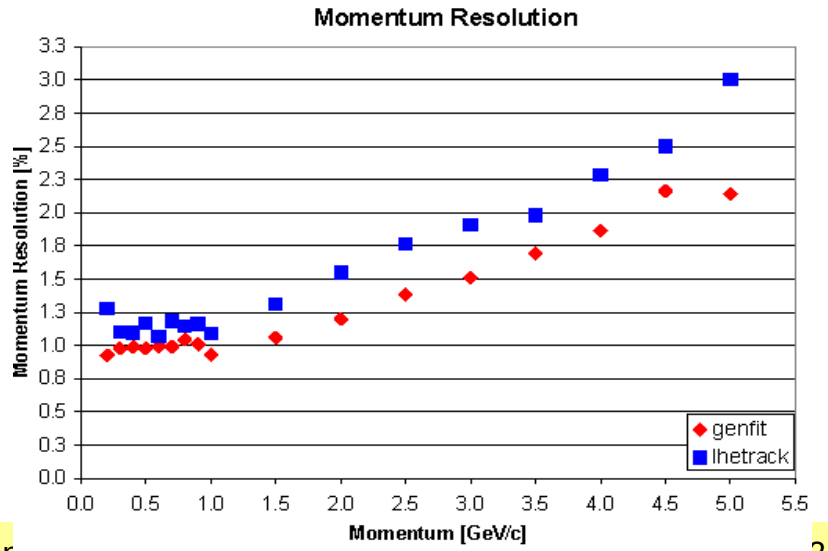

efficiency in %
ID 100
Entries 11
Mean 2.458
RMS 1.530
A0 0.7891 ± 0.5040
A1 −0.2143E−01 ± 0.1667
area in gaussian peak
efficiency decreases
for higher momentum
(same as offline)
p / GeV/c

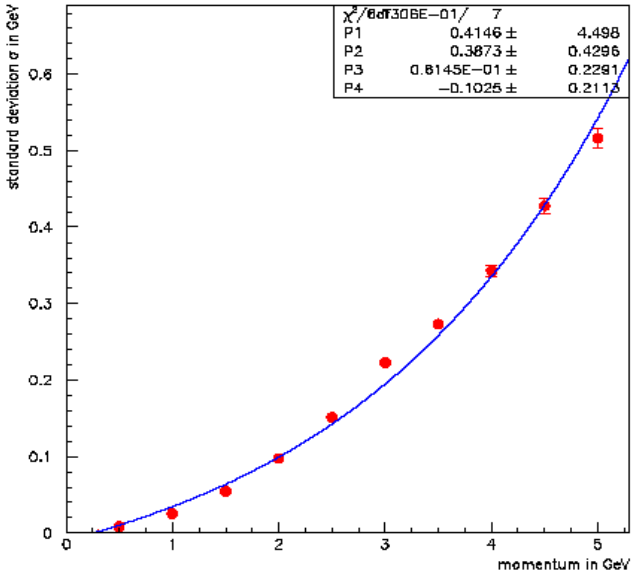

Tracking Efficiency [%]
Online Track Finder
ε(p)=0.7891−0.0214·p
genfit
lhetrack
Momentum [GeV/c]

Momentum Resolution

- Hough space resolution of 800×800 pixel
- online 3% @ 1 GeV, 14% @ 5 GeV
- PandaRoot 1% @ 1 GeV, 3% @ 5 GeV



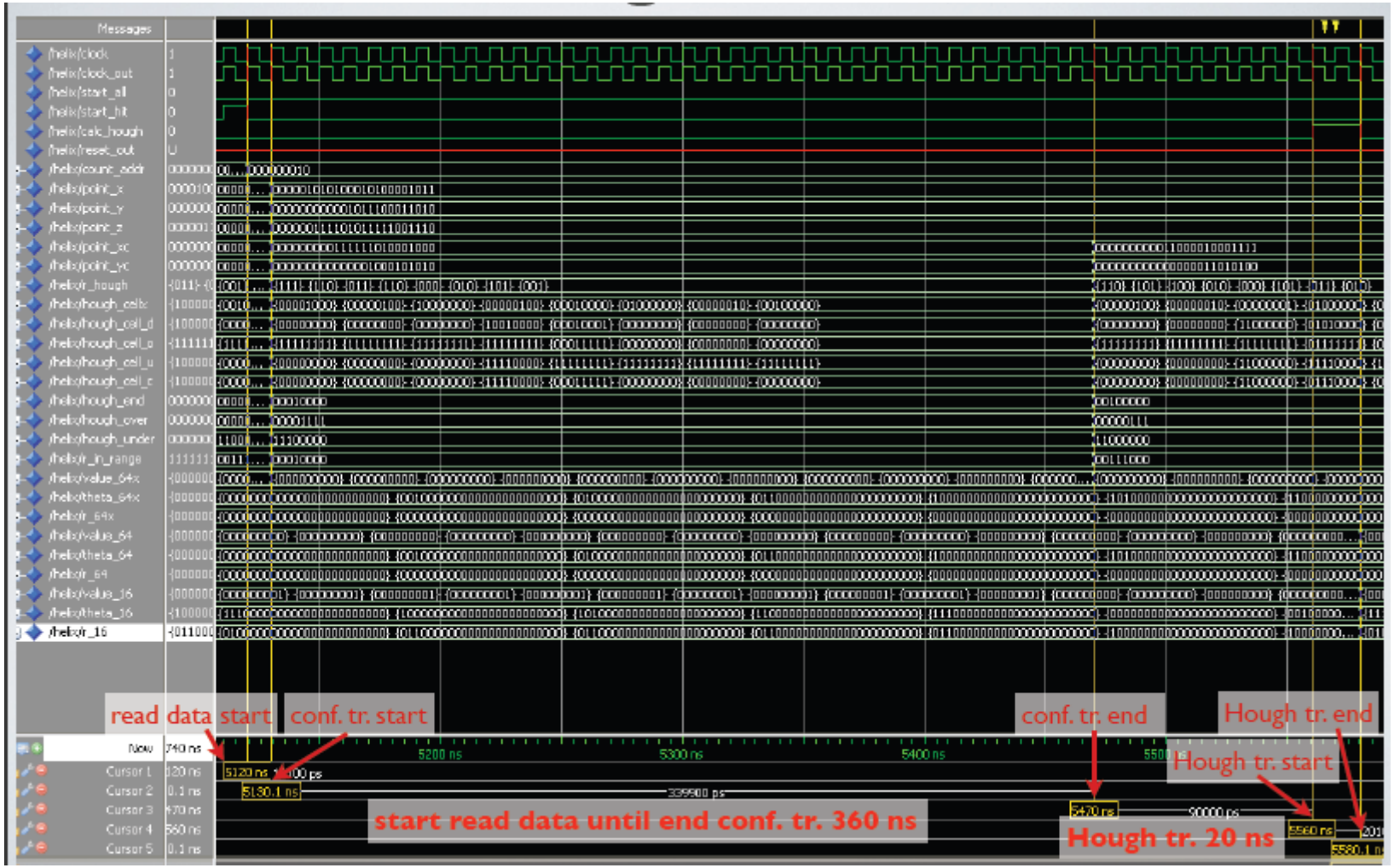




FPGA Implementation

- Conformal Map
  - fixpoint arithmetics
  - conformal coordinate range ±128 1/cm
  - spatial binning 1.5 x 10^{-5}
- Hough Transform
  - Hough Space 128×128 resolution
  - 5 steps (adaptive)
  - parallel in ϑ, serial in r
- Functions
  - sin() only $\rightarrow$ lookup table 128 x 16 bit
    only ϑ=0-90^{o},
    90^{o} – 180^{o} and cos () from phase shifted and inversion
  - parallel division
    (multiple adders, bitwise parallel, unclocked)
  - sqrt() not needed

Adaptive Hough Transfrom on FPGA
64 Hough cells parallel



more than factor 65 faster than 2.4 GHz PC
(incl. sorting algorithm)

Further Developments

- use drift time circles (x,y,r_{drift})
  $\rightarrow$ talk by Yutie Liang
- Try GPU and compare timing to FPGA
  $\rightarrow$ see talk by Mohammad Al-Turany

For discussion session: Cellular Automaton at Belle II PXD

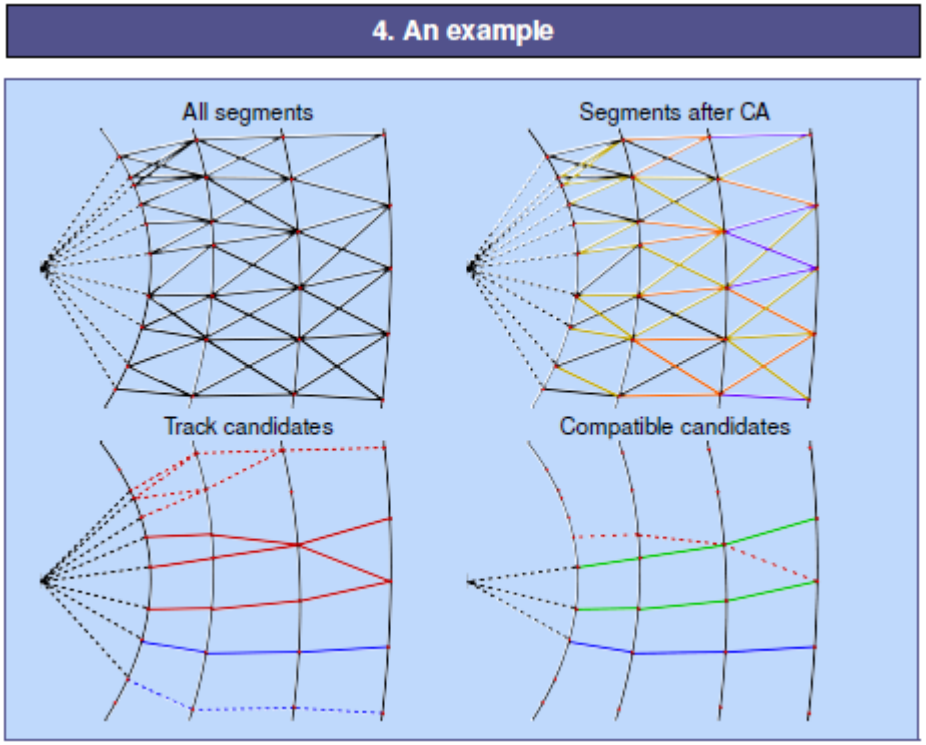
Low-momentum track finding in Belle II

HEPHY Institute of High Energy Physics

J. Lettenbichler, R. Frühwirth, M. Nadler

Institute of High Energy Physics, Austrian Academy of Sciences

ÖAW Austrian Academy of Sciences



To be applied online as „hit recovery“
for low pT<70 MeV
(SVD+PXD tracks, which
never reach the CDC)

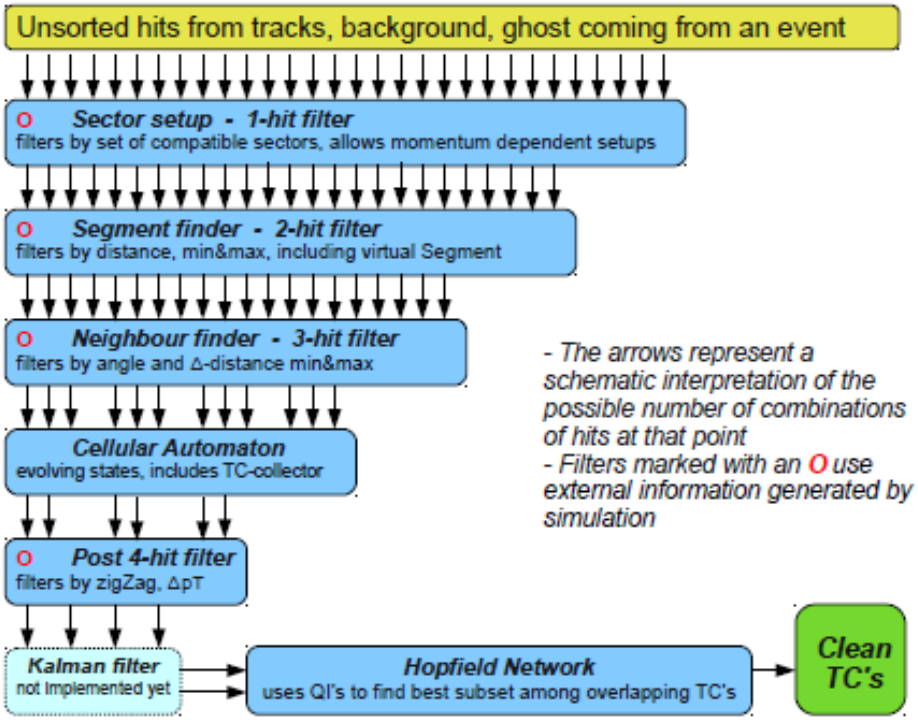
Will be implemented as HLT prototype
in DESY PXD test 11/2013
(ATCA Compute Node will get ROI decision from HLT)

3. Track finding strategy

Global structure

- Stepwise reduction of combinatorics
- Cellular automaton (CA) for finding track candidates
- Kalman filter for computing quality indicators
- Hopfield network for eliminating overlapping candidates

Schematic view of the low momentum track finder in Belle II



Sectors

- Cells of CA are track segments connecting two hits
- Sensors are divided into **sectors**
- This allows for tighter cuts in the segment filters and reduces the number of allowed hit combinations