

SIMDized cluster transformation for ALICE HLT

Sergey Gorbunov



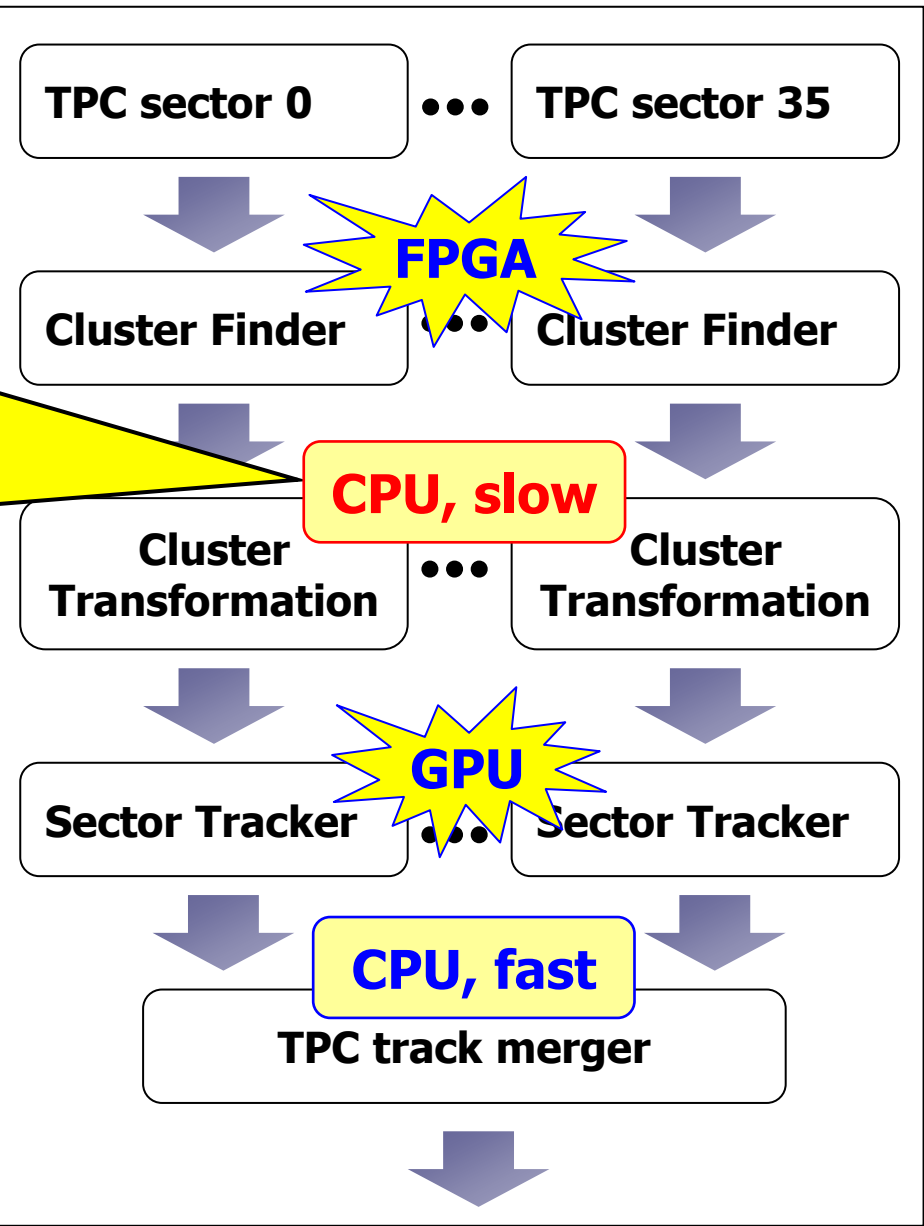
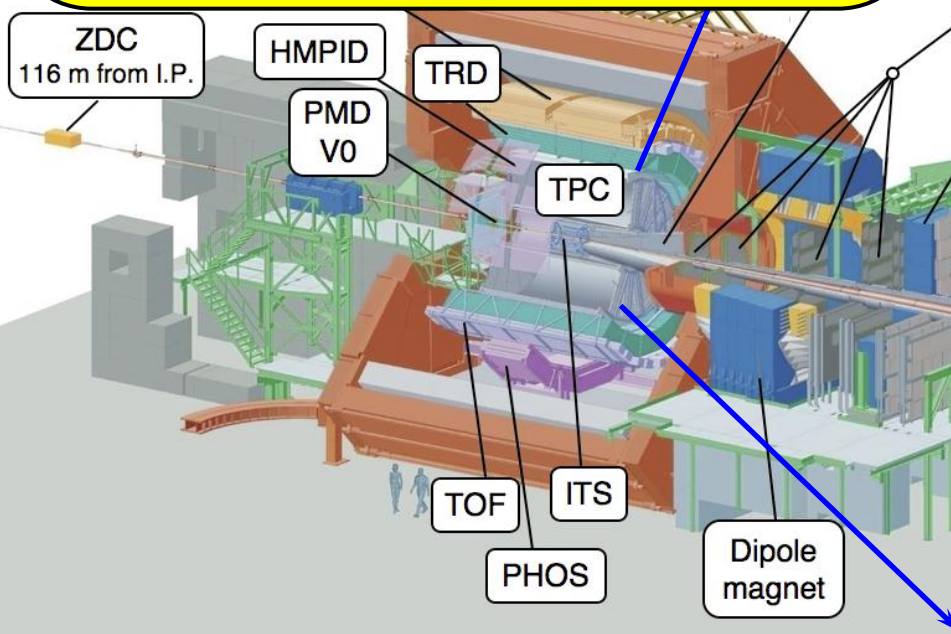
FIAS Frankfurt Institute
for Advanced Studies



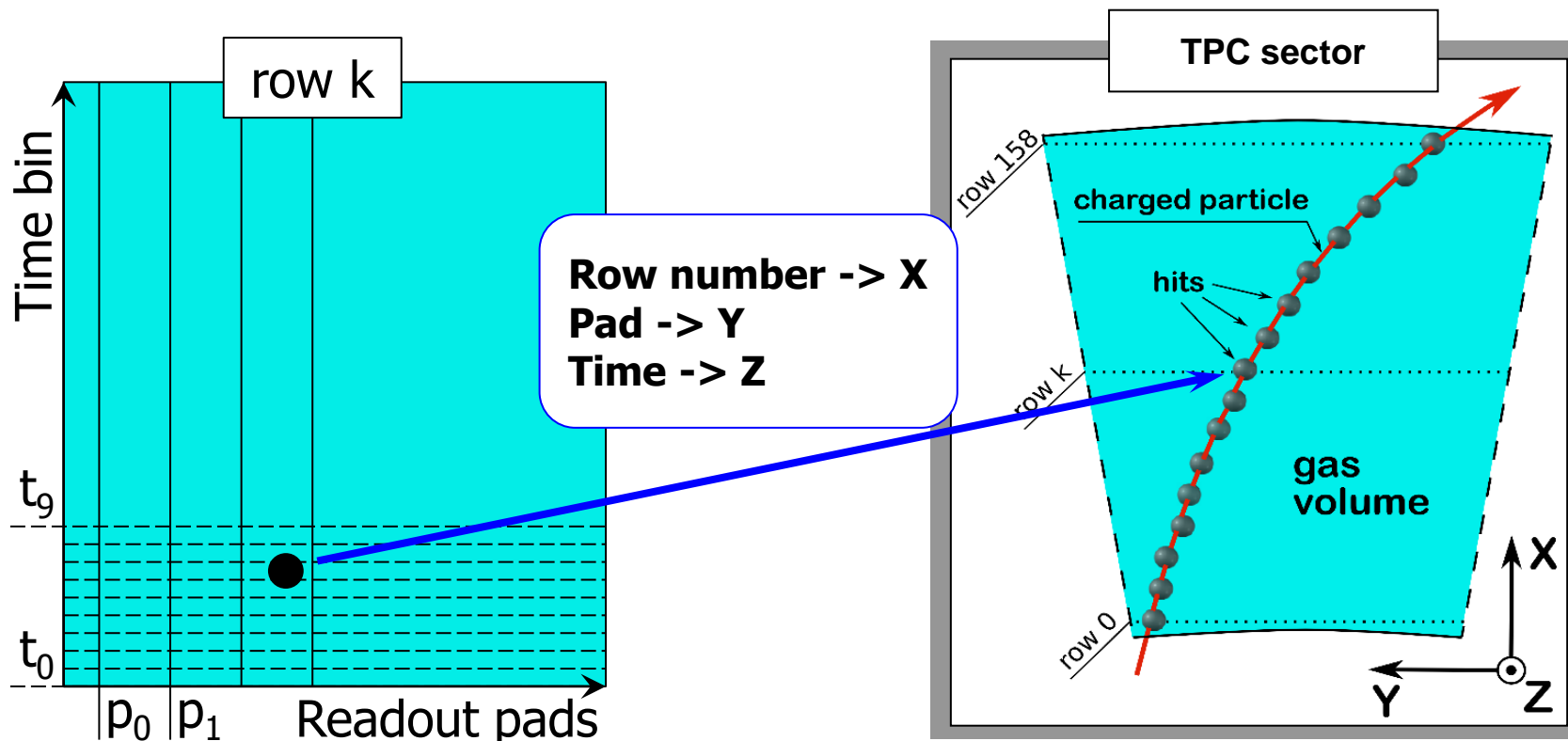
ALICE HLT: Reconstruction in TPC tracker

The **Cluster Transformation** applies calibration and alignment to the TPC clusters. It was the show stopper for HLT:

- off-line code under development, speed is out of the HLT control
- getting more and more complicated
- suddenly became 10 times slower



TPC cluster transformation: (Pad, Time) \rightarrow (X,Y,Z)



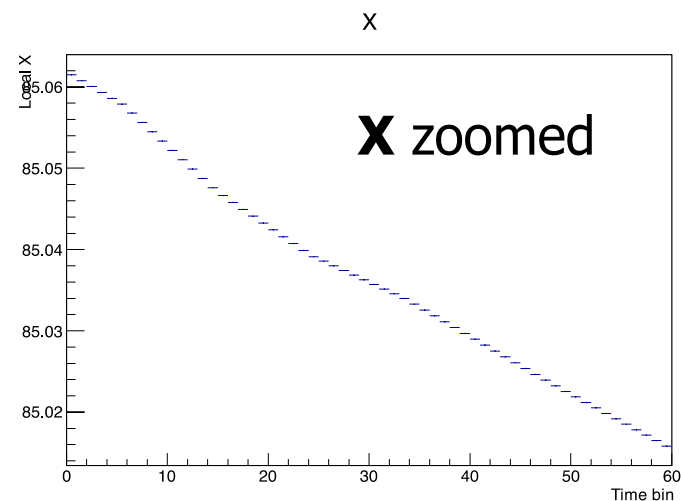
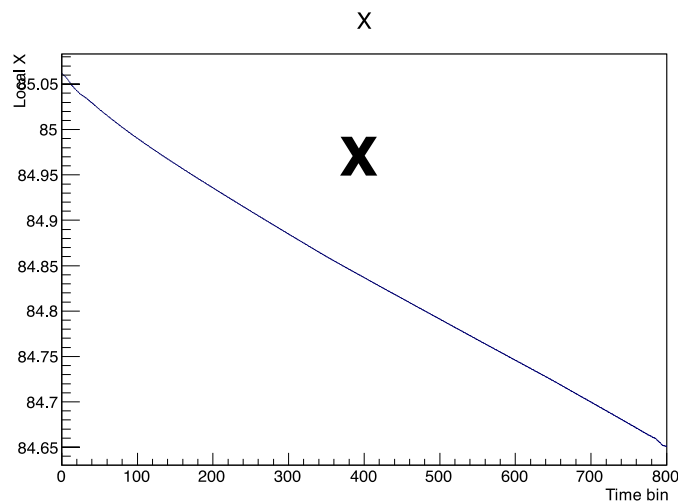
Composed TPC transformation:

1. TPCExBShape
2. alignGlobal
3. alignLocal
4. alignQuadrant
5. FCVoltError3D
6. FitBoundary
7. FitExBTwist
8. FitAlignTPC
9. FitRocAlignZSum
- ... there will be more

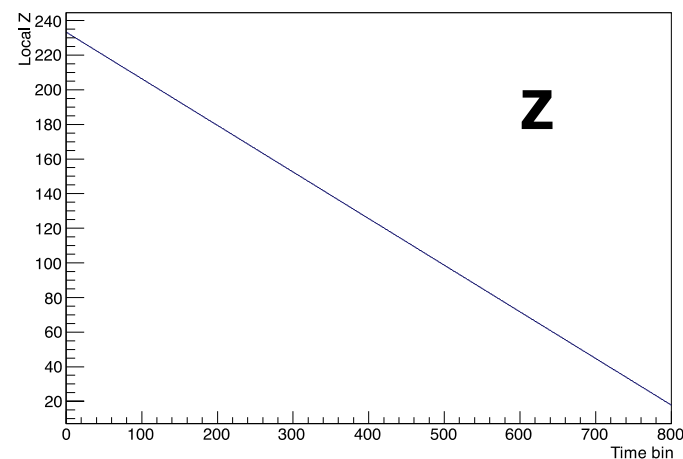
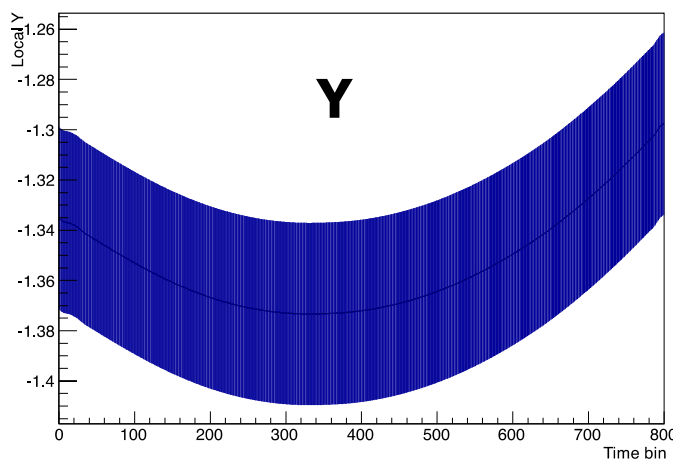
Speed: 7.9 ms/cluster = 6 s/event (PbPb)

Too many different transformations – no hope to optimize the code ...

Final transformation: It does not look that complicated:



γ Sec=0, Row=0, pad coordinate in $[z_2, 31]$



Idea: Fit the original Transformation with polynoms/splines:

Initialization phase: Create temporary splines by calling TPCTransform() :

- no matter which corrections are applied,
- no matter how slow it is

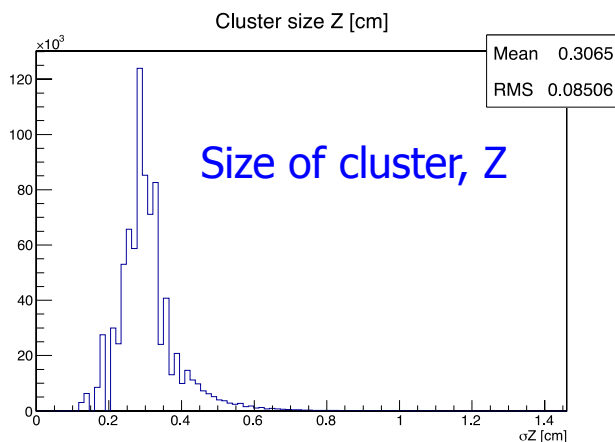
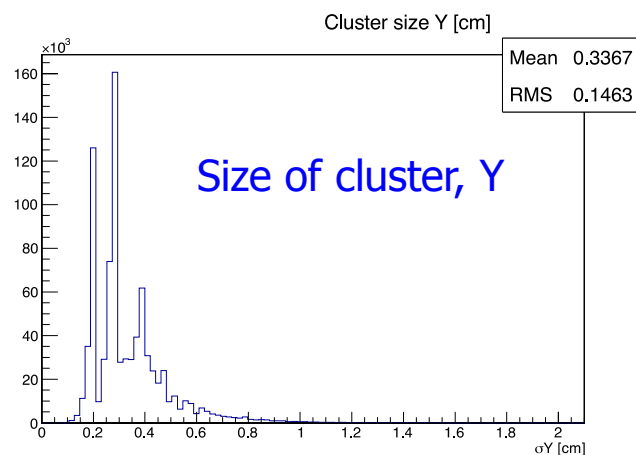
Running phase: Use of the splines:

- few arithmetic operators --- very fast
- transformation time is constant: no more dependence on other code and data base objects

Idea: Fit the original Transformation with polynoms/splines:

Requirements:

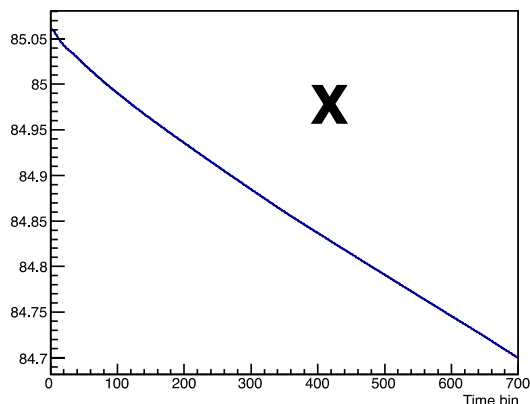
- Accuracy: 100 μm
- Speed: 10 times faster
- Transformation for 33 rows should fit to CPU cache
- Use of SIMD vectors (Vc package) for faster calculations



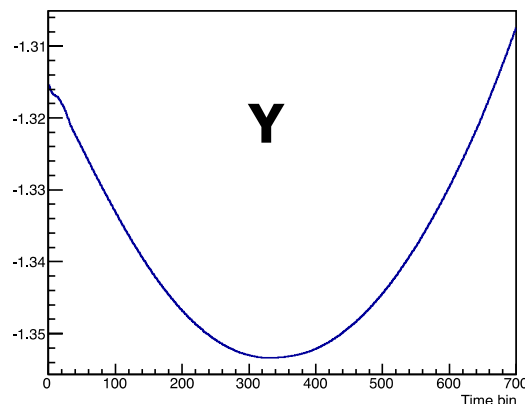
Fit with polynoms: Many local features- does not work well:

Difference with polynom of 0 order [cm]:

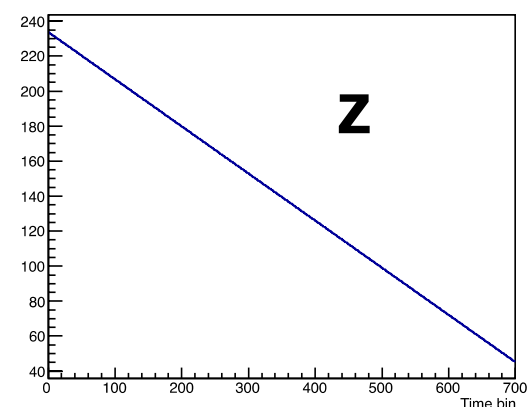
X



Y

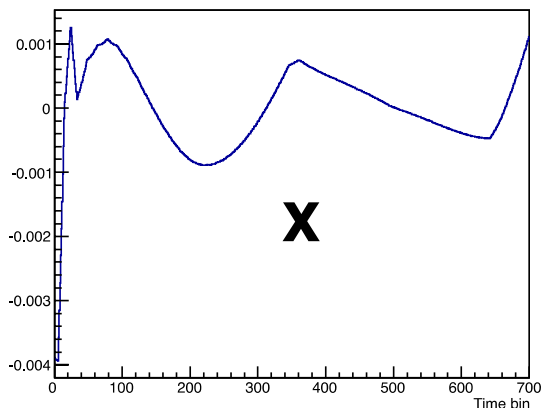


Z

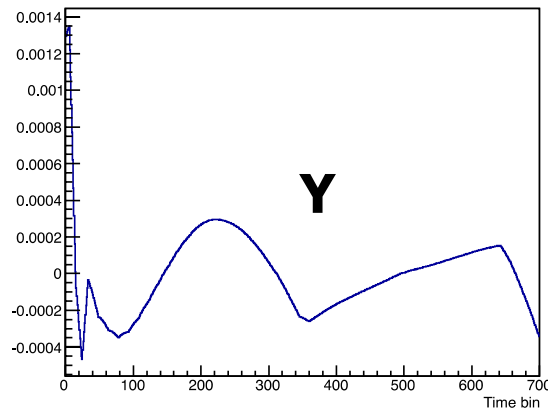


Difference with 5-th order polynom [cm]:

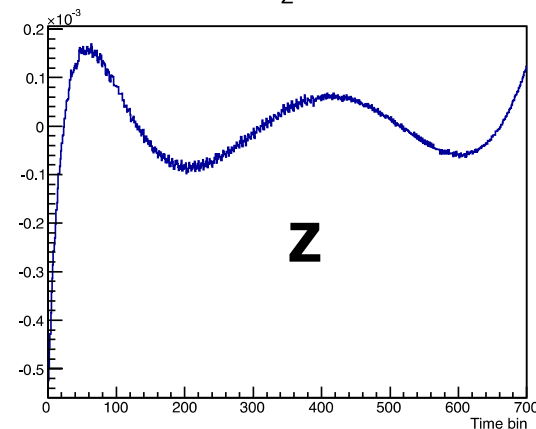
X



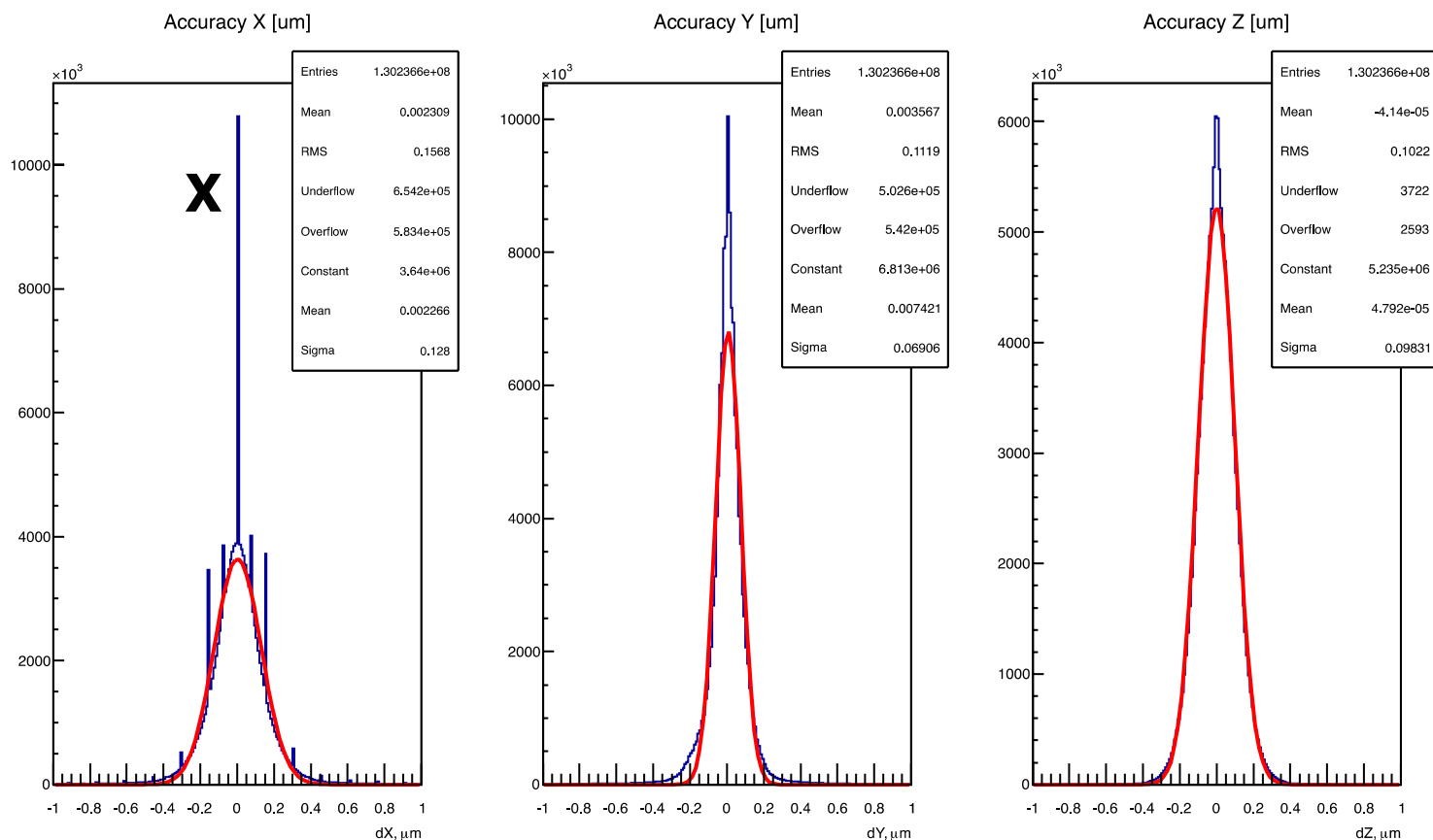
Y



Z

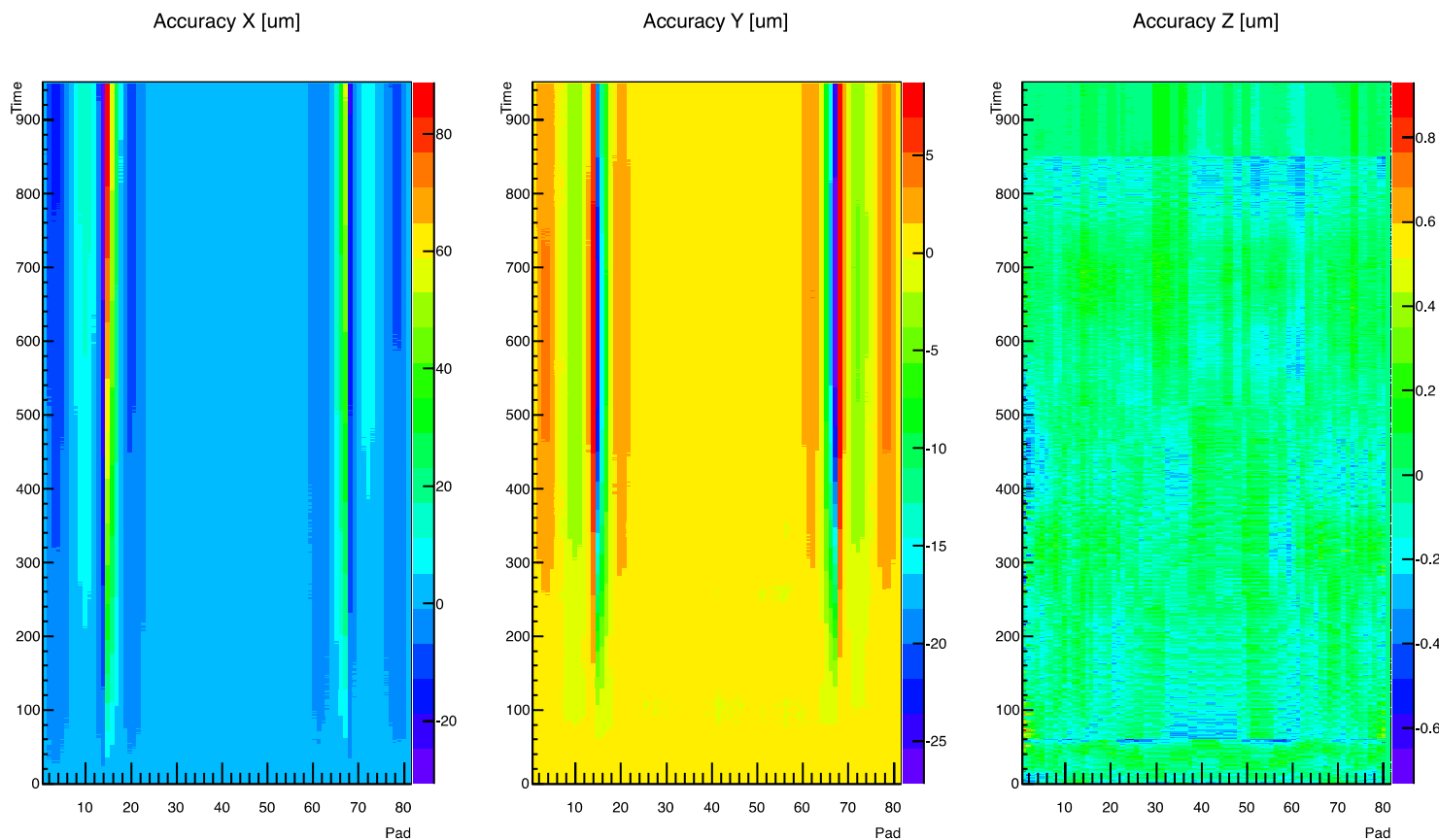


Fit with splines: Accuracy about 0.1 μm for X,Y,Z



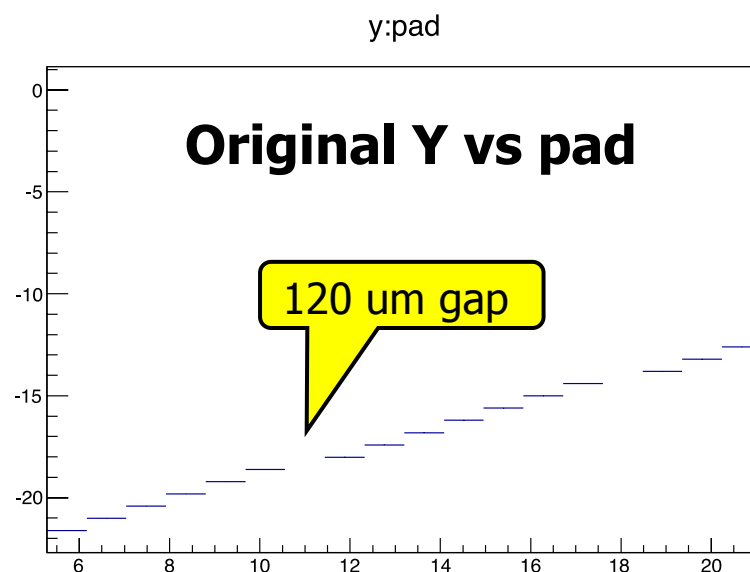
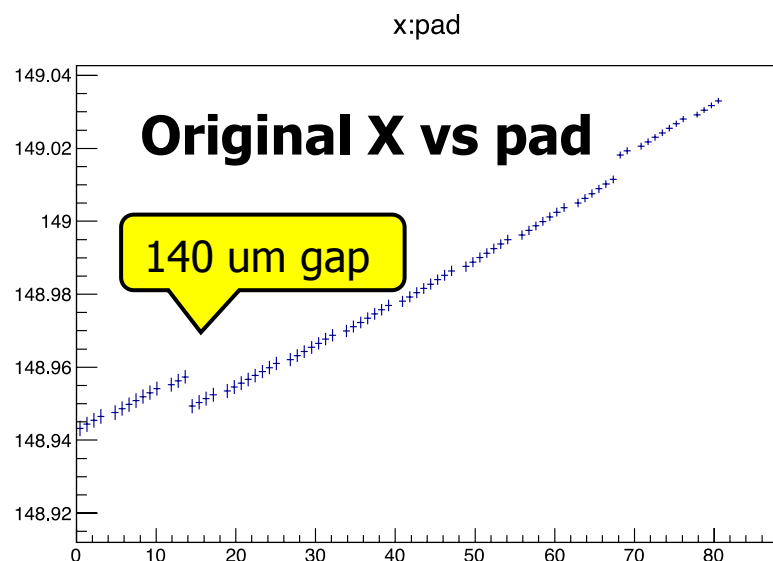
Worst deviation in X: 88 μm , in Y: 81 μm , in Z: 3.8 μm

Fast HLT transformation: TPC row with the worst deviations



(Orig. - Fast transformation)[μm] vs Pad and Time

Fast HLT transformation: TPC row with the worst deviations



**This particular TPC row seems to be miscalibrated to 140um
-> accuracy of the Fast Transformation [80um] here is not worse
than the accuracy of the original transformation [140um].**

Fast HLT Transformation: Results

- Accuracy: 0.1 μm
- Speed-up in scalar: 58.1 times
- Speed-up with SIMD vectors: **83.6 times**
(SIMD give factor 1.44 extra speed-up)
- Transformation size for 33 rows: 170kB (full TPC: 29MB)

Use of SIMD vectors:

- **Vc** package from M.Kretz – very useful tool
- Spline points are stored as 4-vectors: $(x, y, z, *)$
 - This scheme allows one to avoid packing of data to vectors, but it only works with SIMD vectors of size 4.

Parallelization in ALICE HLT:

- All time-consuming HLT components are running now on parallel hardware:
 - cluster finder: **FPGA**
 - cluster transformation: **SIMD** (via Vc package)
 - sector tracker: **GPU**
- Fast transformation gives factor of **83.6** times speed-up
- Vc package installed and can be used offline as well



Pb+Pb @ $\sqrt{s} = 2.76$ ATeV

2010-11-08 11:30:46

Fill : 1482

Run : 137124

Event : 0x00000000D3BBE693