Sensor alignment by tracks

A. Heikkinen, V. Karimäki, T. Lampén and T. Lindén
Helsinki Institute of Physics, University of Helsinki

CHEP03, March 24-28, 2003
La Jolla, California

Outline:

- **Introduction**
- **Description of the alignment algorithm**
- **Performance with real data**
- **Performance studies with simulated pixel detector**
- **Summary**

Introduction - tracking sensor alignment

Characteristics of typical inner tracking detectors being built:

- large dimensions, from a few dm up to a few meters
- high sensor hit precision, from a few microns to tens of microns
- built out of hundreds or thousands of sensors

It follows that precise determination of sensor positions is a big challenge for:

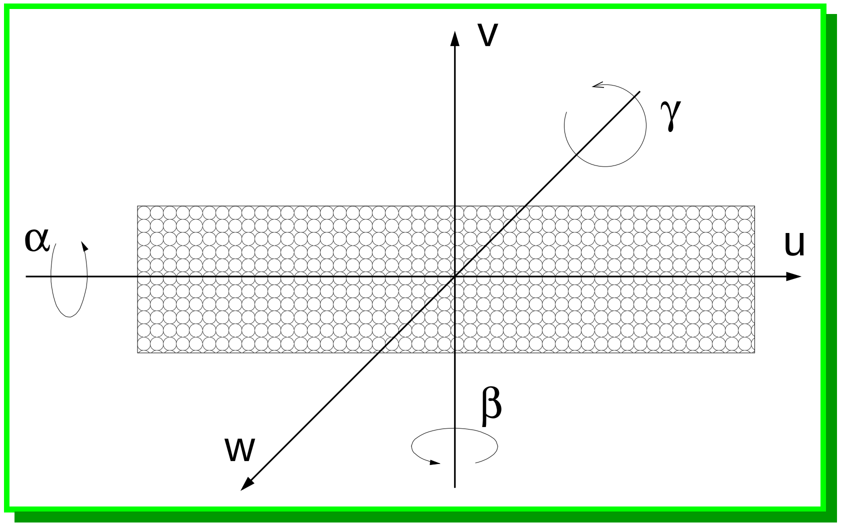
- assembly precision, typically a few hundred μm
- survey measurements, precision of the order 100 μm or better
- monitoring devices to monitor sensor movements at run time
- final position calibration i.e. detector alignment to be made by reconstructed tracks

In this work we introduce an efficient algorithm to determine corrections to sensor locations and orientations by tracks.

- translation plus tilt correction for sensors (up to 6 parameters per sensor)
- straightforward mathematical formalism
- no big matrix inversions
- applies to detector setups with very large number of sensors

Working in local coordinate system

Local (sensor) coordinate system:



Alignment corrections $\Delta u, \Delta v, \Delta w$:
correction for sensor origin

Alignment corrections $\Delta\alpha, \Delta\beta, \Delta\gamma$:
correction for rotation local $\Leftrightarrow$ global

Benefits of working in local coordinate system:

- **hit positions are invariant**
- **only trajectory impact point moves as a function of sensor translation and tilt**

V. Karimäki/HIP
Helsinki Institute of Physics, University of Helsinki

Description of the algorithm

- the task is to correct the coordinate transformations:

$$\bar{\mathbf{q}} = \mathbf{R}(\bar{\mathbf{r}} - \bar{\mathbf{r}}_0)$$

  from global system to sensor local system

- corrected transformation reads:

$$\bar{\mathbf{q}}' = \Delta\mathbf{R}\,\mathbf{R}(\bar{\mathbf{r}} - \bar{\mathbf{r}}_0) - \Delta\bar{\mathbf{q}} \qquad \text{where:}$$

$$\Delta\bar{\mathbf{q}} = (\Delta u, \Delta v, \Delta w) \qquad \text{(sensor shift)}$$

$$\Delta\mathbf{R} = \Delta\mathbf{R}(\Delta\alpha, \Delta\beta, \Delta\gamma) \qquad \text{(sensor tilt)}$$

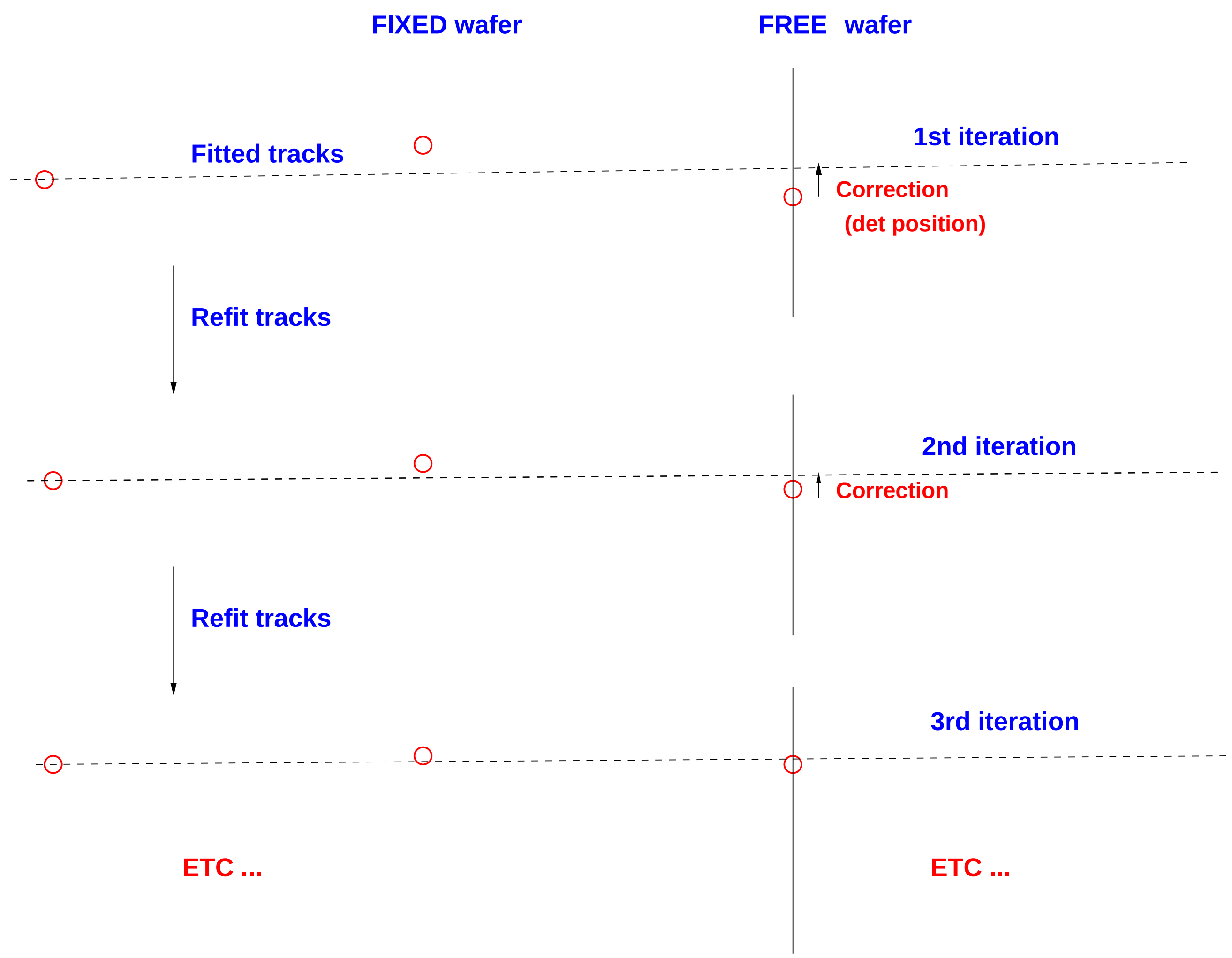
- how does the trajectory impact point $\bar{\mathbf{q}}_x$ transform under a sensor shift and tilt correction: $\bar{\mathbf{q}}_x \rightarrow \bar{\mathbf{q}}'_x$?
- depends also on the trajectory direction $\hat{\mathbf{s}}$ at the impact point:

$$\bar{\mathbf{q}}'_x = \mathbf{R}'(\bar{\mathbf{r}}_x - \bar{\mathbf{r}}_0) + \frac{[\Delta\bar{\mathbf{q}} - \mathbf{R}'(\bar{\mathbf{r}}_x - \bar{\mathbf{r}}_0)] \cdot \hat{\mathbf{w}}}{\mathbf{R}'\hat{\mathbf{s}} \cdot \hat{\mathbf{w}}}\mathbf{R}'\hat{\mathbf{s}} - \Delta\bar{\mathbf{q}}$$

  where $\mathrm{R}' = \Delta\mathrm{R}\,\mathrm{R}$

- this is the key formula of the algorithm
- residuals $\bar{\varepsilon}$ = hit position minus impact position
- separation of the χ^2 function: $\boldsymbol{\chi}^2 = \sum_{sensors} \boldsymbol{\chi}^2_{sensor}$
- minimizing separately each χ^2_{sensor} and iterating
- one iteration cycle: tracks refitting followed by minimization of all χ^2_{sensor}

Simplified illustration - how iteration works



Iteration which involves tracks refitting and χ^2 minimization for alignment parameters of all alignable sensors is repeated until convergence is observed.

V. Karimäki/HIP
Helsinki Institute of Physics, University of Helsinki

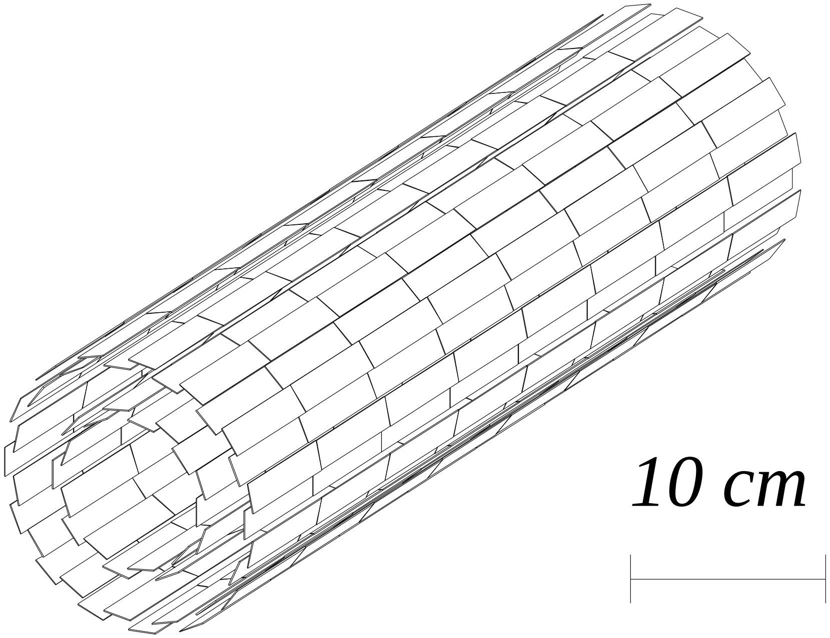
Algorithm performance studies

We have tested the algorithm both with real data and by simulation:

- test beam data from Helsinki Si Beam Telescope (SiBT) at the CERN H2 beam
- pixel vertex detector barrel model (CMS-like) with two layers

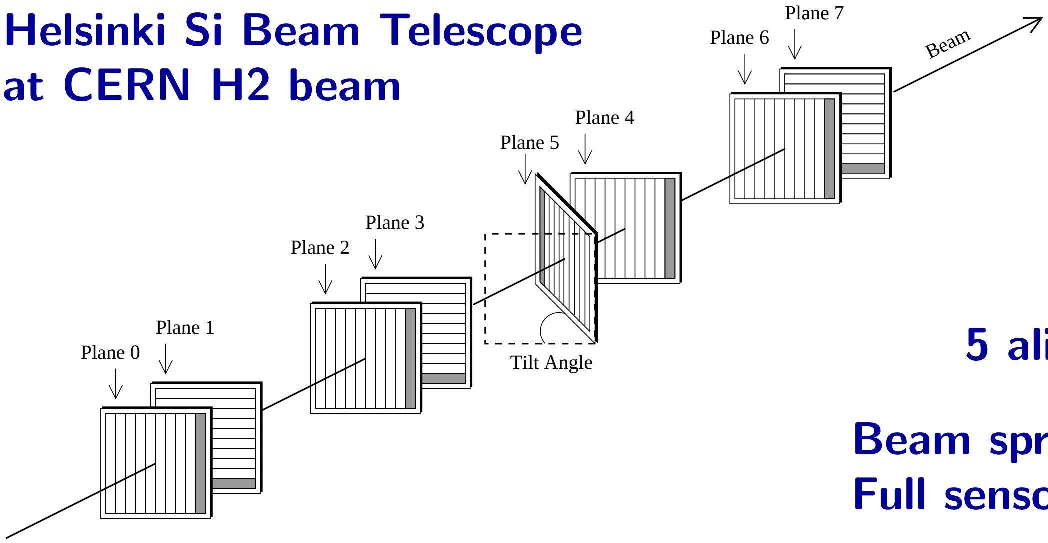
1. **SiBT:**
   - Eight silicon strip sensors, 5×5 cm^2, total length of array 55 cm
   - Data taken with no field and 300 GeV muons

2. **Vertex detector (pixel barrel):**
   - sensors $16\,$mm $\times$ $64\,$mm
   - layers at 4.4 cm and 7.3 cm, $L = 50\,$cm
   - 144 sensors in layer 1, 240 in layer 2
   - sensor overlaps in $r\varphi$ about 4 %
   - no overlaps in z
   - used tracks with $p_T >$0.8 GeV
   - beam-line and vertex z-constraint
   - Gaussian smeared hits plus m.s.



Algorithm performance with test beam data

Helsinki Si Beam Telescope at CERN H2 beam



pitch = $50\,\mu$m

Strip detectors:
5 alignment parameters per sensor

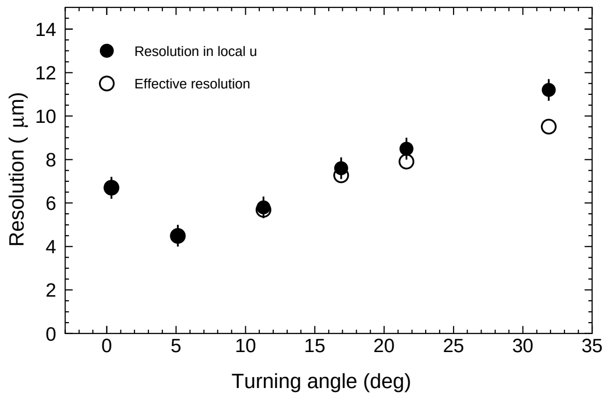
Beam spread = ± 10 mrad (stereo effect)
Full sensor area coverage by beam

Used 3000 tracks for alignment of the tilted sensor (5)

Obtained alignment parameters:

Parameter	At 0 degrees	At 30 degrees
$\Delta u(\mu$m)	186.0±0.1	-264.7±0.1
$\Delta w(\mu$m)	200±20	-131±6
$\Delta\alpha$(mrad)	5.6±0.7	12.9±0.9
$\Delta\beta$(mrad)	5.8±0.9	32.59±0.04
$\Delta\gamma$(mrad)	-14.12±0.01	-15.86±0.01

Obtained resolutions:



Pixel simulation: layer 2 fixed, layer 1 mis-aligned (I)

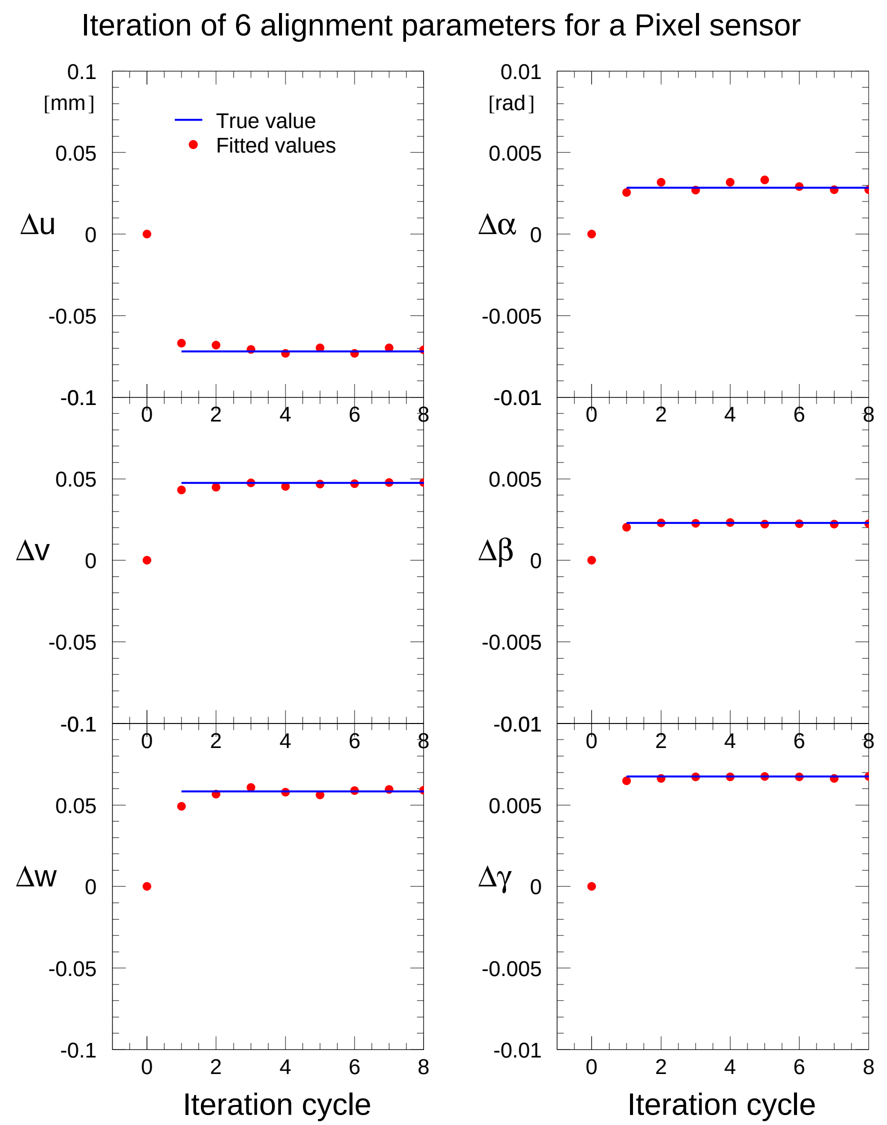
- **Layer 2 fixed**
- **Layer 1 sensors mis-aligned at random: flat within $\pm 100\,\mu$m and ± 10 mrad**
- $6 \times 144 =$ **864 parameters to be fitted**
- ≈ 500 **tracks per sensor in the mean**

Figure:

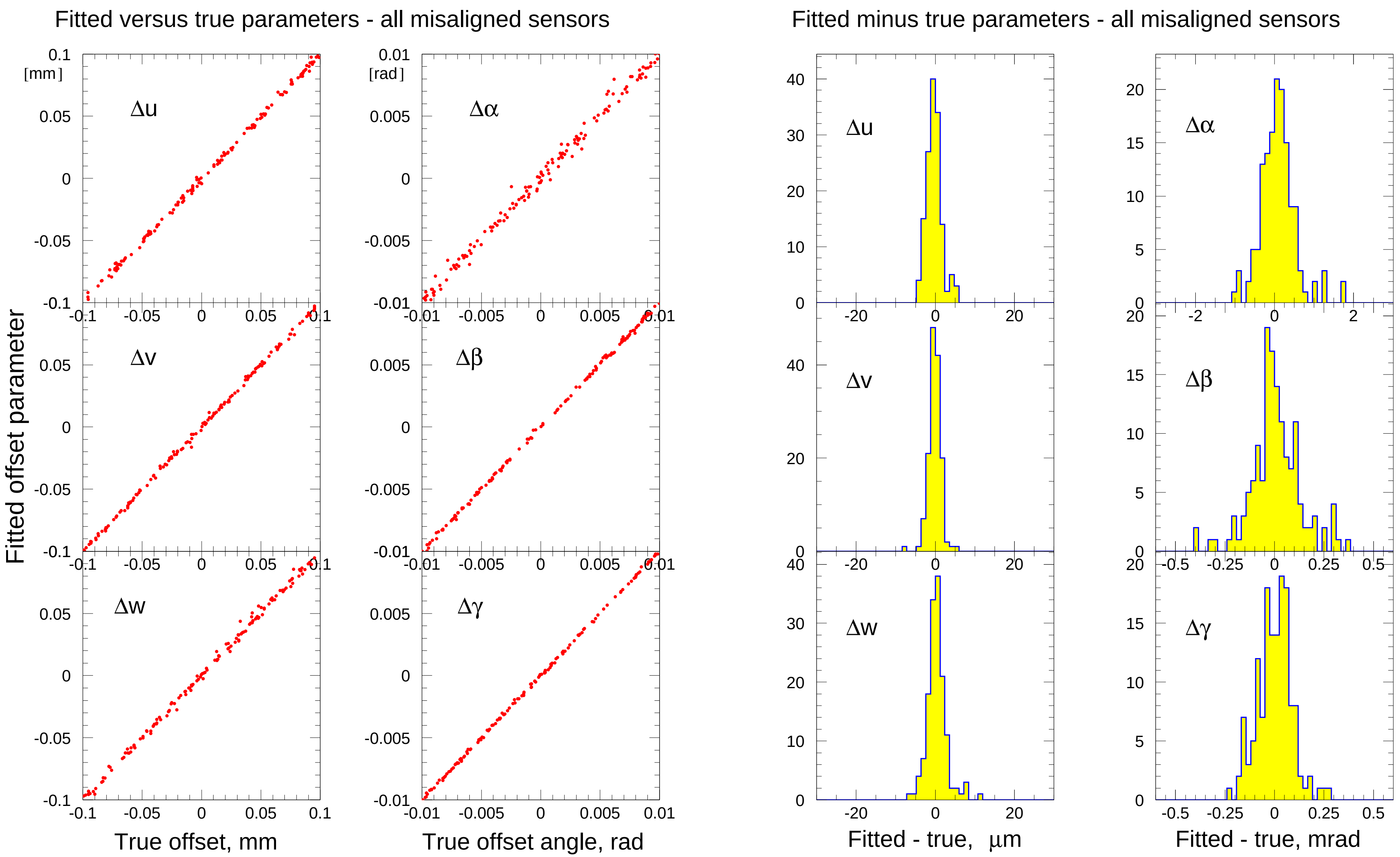
Convergence of 6 parameters for one sensor

Same behaviour for all the other 143 sensors

In this case 3–4 iteration cycles are sufficient

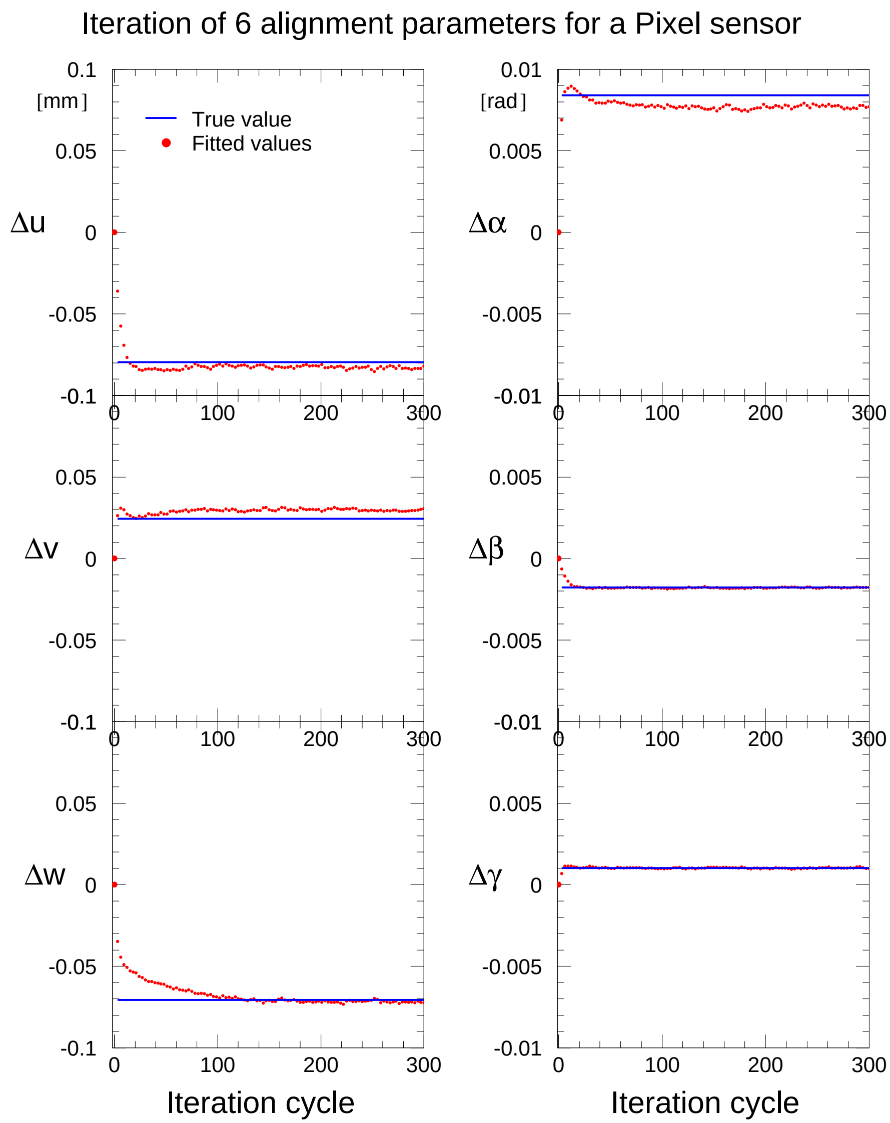


Pixel simulation: layer 2 fixed, layer 1 mis-aligned (II)

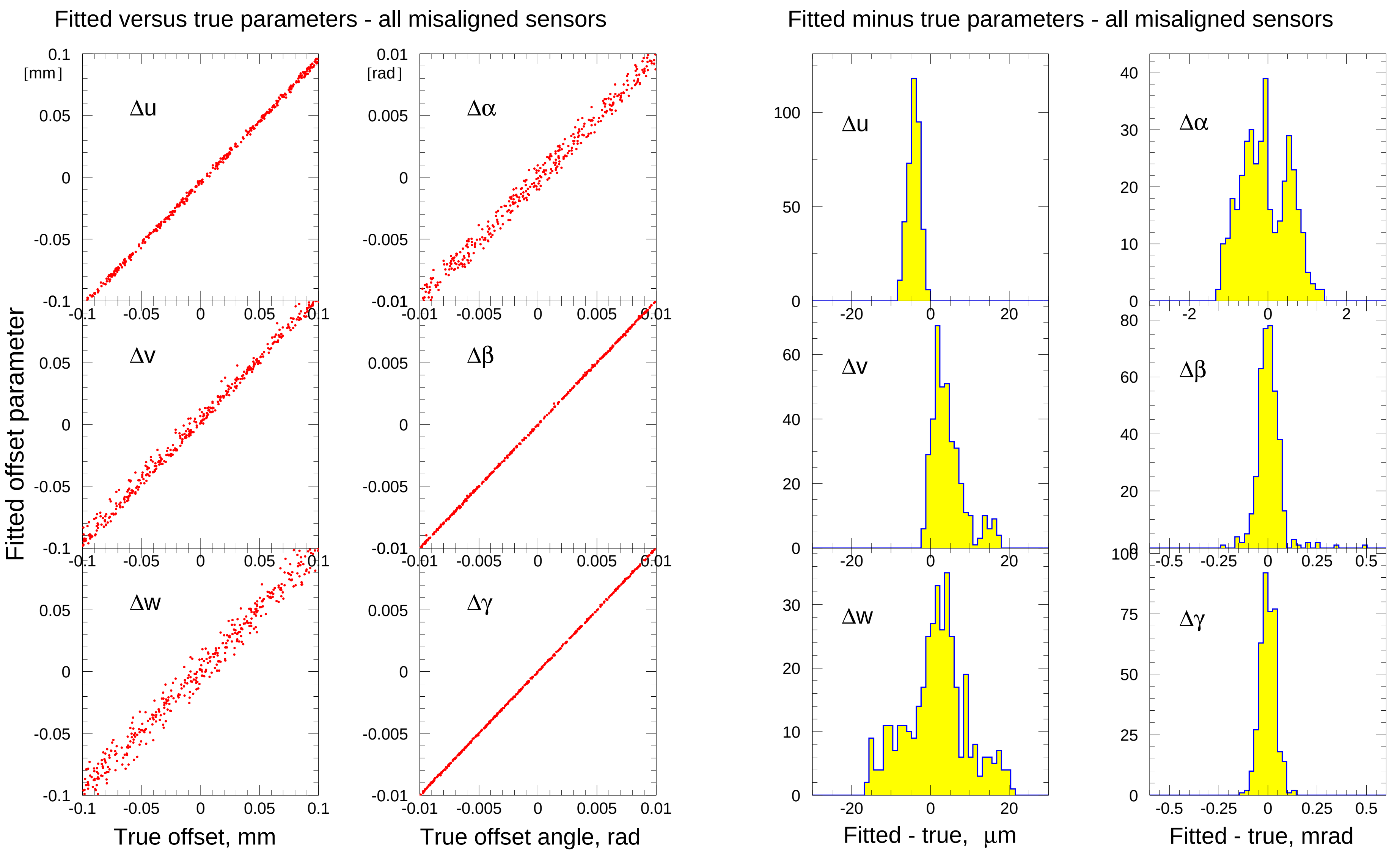


Extreme case: Only one sensor fixed, 383 mis-aligned (I)

- **Fixed sensor: in layer 2, near z=0, $\varphi = 0$**
- **The rest to be aligned**
- $6 \times 383 = 2298$ **parameters fitted**
- 2×10^5 **tracks used per iteration cycle**
- **Up to 300 iterations**
- **In figure: parameter convergence for a sensor farthest away from the reference sensor**
- **Faster convergence for sensors closer to the reference sensor**



Extreme case: Only one sensor fixed, 383 mis-aligned (II)



Summary

Effective sensor alignment algorithm with the following basic features:

- formulation in local coordinate system
- trajectory impact point coordinates as a function of offsets and tilts
- straightforward mathematics
- separation of the χ^2 function in terms of sensors
- small matrix formalism (max 6×6)
- minimization by iterative steps:

1. minimization of all χ^2_{sensor} separately
2. refitting tracks, back to 1.

Demonstration of the performance

- test beam data with Si strip telescope, precise 5 parameters alignment
- 2-layer pixel barrel alignment simulation, two options:

1. layer 2 fixed, layer 1 sensors to be aligned
   - all 864 parameters obtained with good precision
2. only one sensor fixed, the rest to be aligned
   - all 2298 parameters converge with fairly good precision