Differences of local coordinate residuals for hits in overlapping modules of a same VXD layer

Filippo Dattola

Diego Tonelli

Overview

The goal of this study is to **develop a novel approach to supplement the vertex detector alignment procedures currently in use.**

Use simulated $e^+e^- \to \mu^+\mu^-$ events, reconstructed through various characteristic misaligned-detector configurations, to:

- identify sensor-based variables sensitive to the specific features of each configuration;

- develop procedures aimed at using them swiftly for early identification of realistic misalignments;

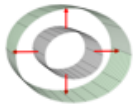
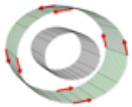
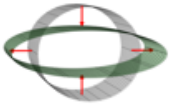
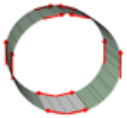
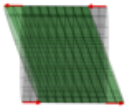
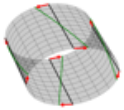
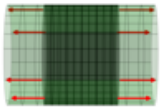
- complement and supplement the standard Belle II approaches to VXD alignment.

All results illustrated in the following slides are obtained using Belle II software tools of the 02-00-01 release

Weak modes

An insidious class of misalignments exists, to which the standard alignment procedure is poorly sensitive: **the weak modes.**

They are coherent, common-mode movements of individual VXD sensors poorly affecting the overall χ^2

	Δr	$r\Delta\phi$	Δz
r	**Radial expansion** $\Delta r = c_{scale} \cdot r$ 	**Curl** $r\Delta\phi = c_{scale} \cdot r + c_0$ 	**Telescope** $\Delta z = c_{scale} \cdot r$
ϕ	**Elliptical expansion** $\Delta r = c_{scale} \cdot \cos(2\phi) \cdot r$ 	**Clamshell** $\Delta\phi = c_{scale} \cdot \cos(\phi)$ 	**Skew** $\Delta z = c_{scale} \cdot \cos(\phi)$ 
z	**Bowing** $\Delta r = c_{scale} \cdot \lvert z \rvert$	**Twist** $r\Delta\phi = c_{scale} \cdot z$ 	**Z expansion** $\Delta z = c_{scale} \cdot z$ 

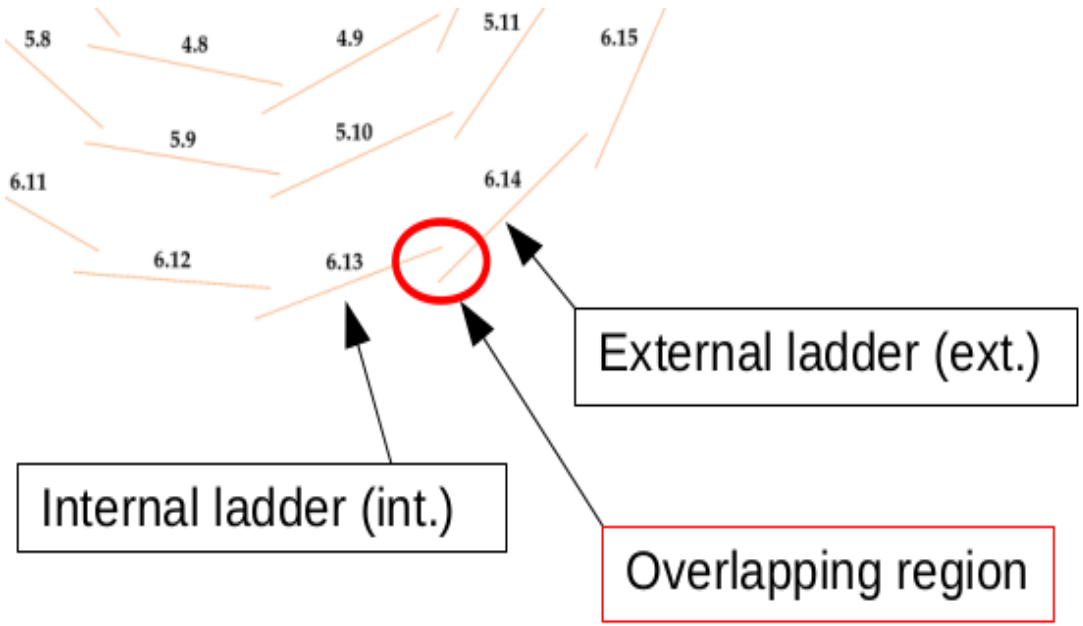
The standard procedure could in principle identify weak modes but with a **significant increase in converging time.**

Look for a faster procedure to spot weak modes (possibly) during on-line reconstruction, by using hits in overlapping modules of a same VXD layer.

A novel approach

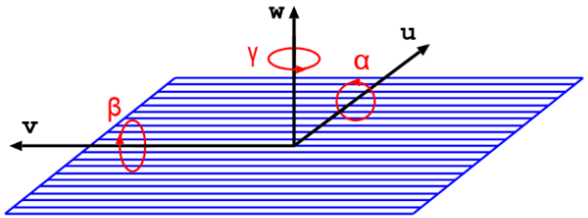
Consider $e^+e^- \to \mu^+\mu^-$ events (easily reconstructed) and use **hits in azimuthally adjacent modules, overlapping within a same radial layer of the inner tracker, to identify possible weak-mode misalignments.**

Needs larger amounts of data (only a small fraction of tracks traverse the overlapping sensors) but: marginally sensitive to multiple Coulomb scattering (little radial distance between overlapping sensors) and can use any kind of tracks (signal and background).



Detailed description of the algorithm to compute residuals is given at the end of this document in "Residuals algorithm".

Define test variables, using local coordinates *u, v*:



$$res_{(u)} = \Delta u_{ext.} - \Delta u_{int.}$$

$$res_{(v)} = \Delta v_{ext.} - \Delta v_{int.}$$

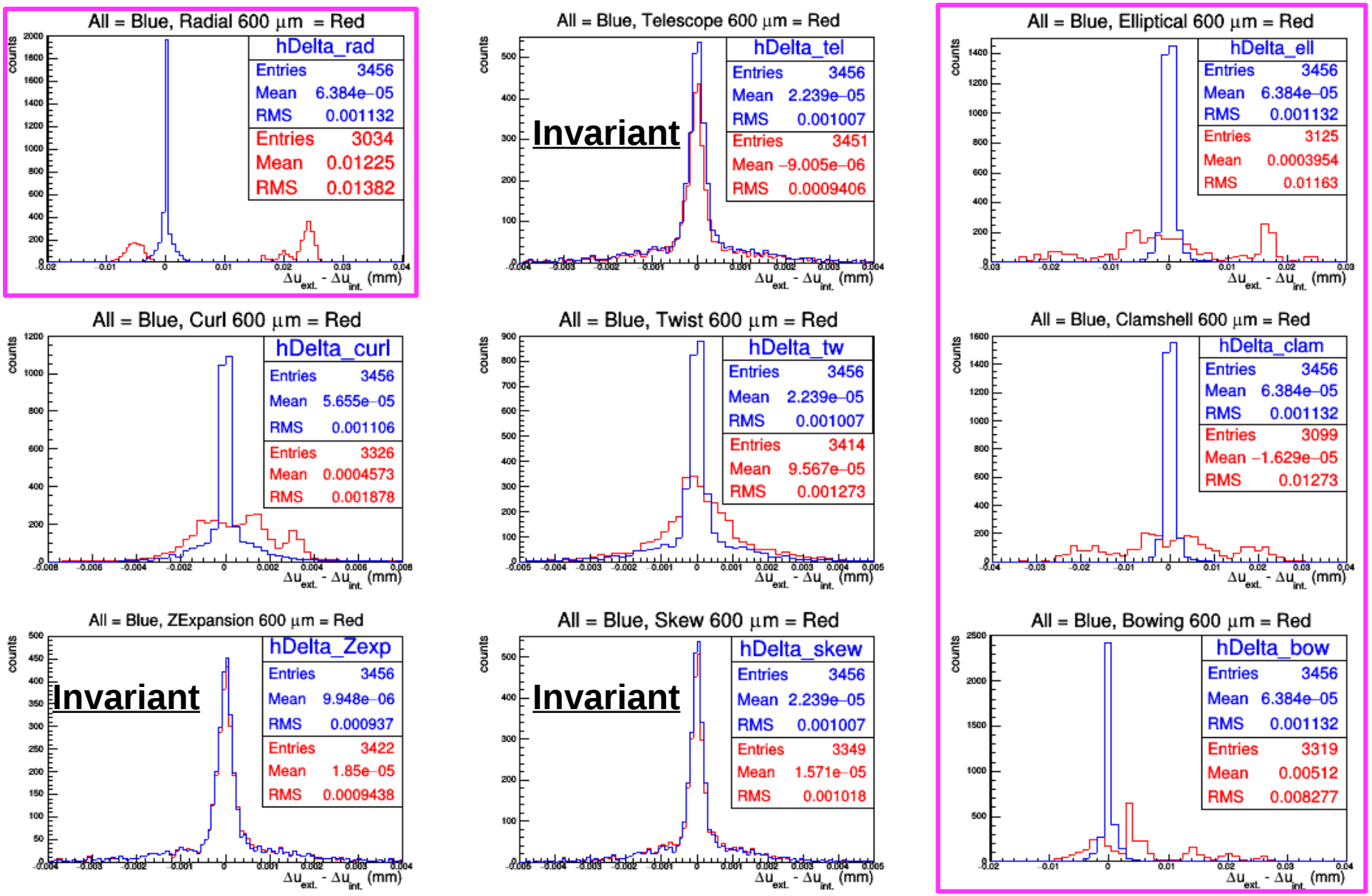
Where $\Delta u = u^{meas.} - u^{pred.}$

$$\Delta v = v^{meas.} - v^{pred.}$$

And u^{pred}, v^{pred} come from the track state vector.

Benchmark misalignments: 600 µm, u-residuals.

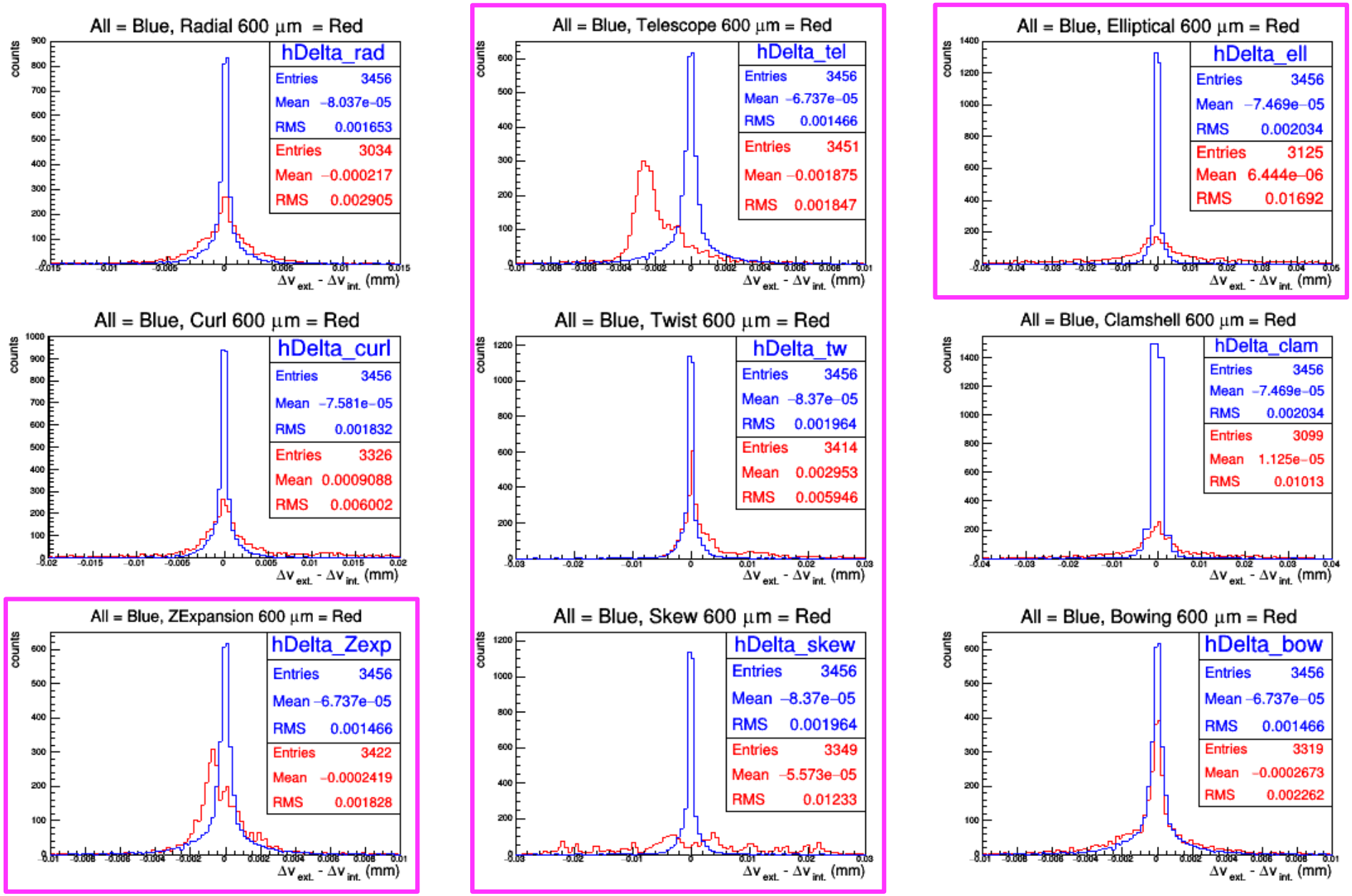
$\Delta u_{ext.} - \Delta u_{int.}$ comparison for aligned or misaligned VXD (by up to 600 µm) according to each weak mode.



Asymmetric and/or multi-modal distributions for <u>radial expansion, elliptical expansion, clamshell and bowing.</u>

Benchmark misalignments: 600 μm, v-residuals.

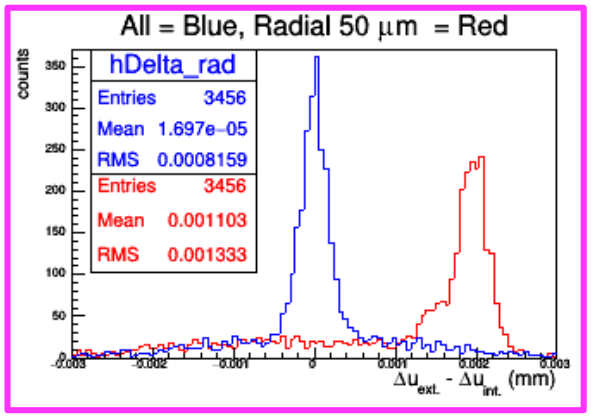
$\Delta v_{ext.} - \Delta v_{int.}$ for aligned or misaligned VXD (by up to **600 μm**) according to each weak mode.

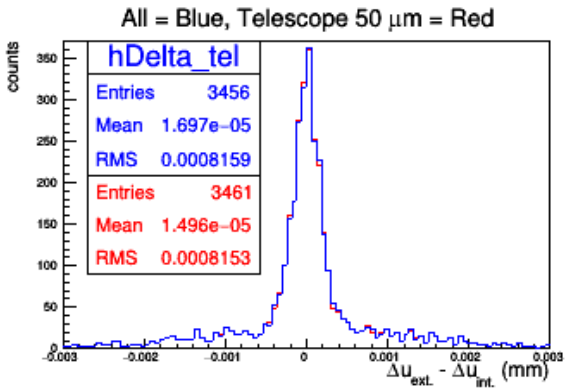


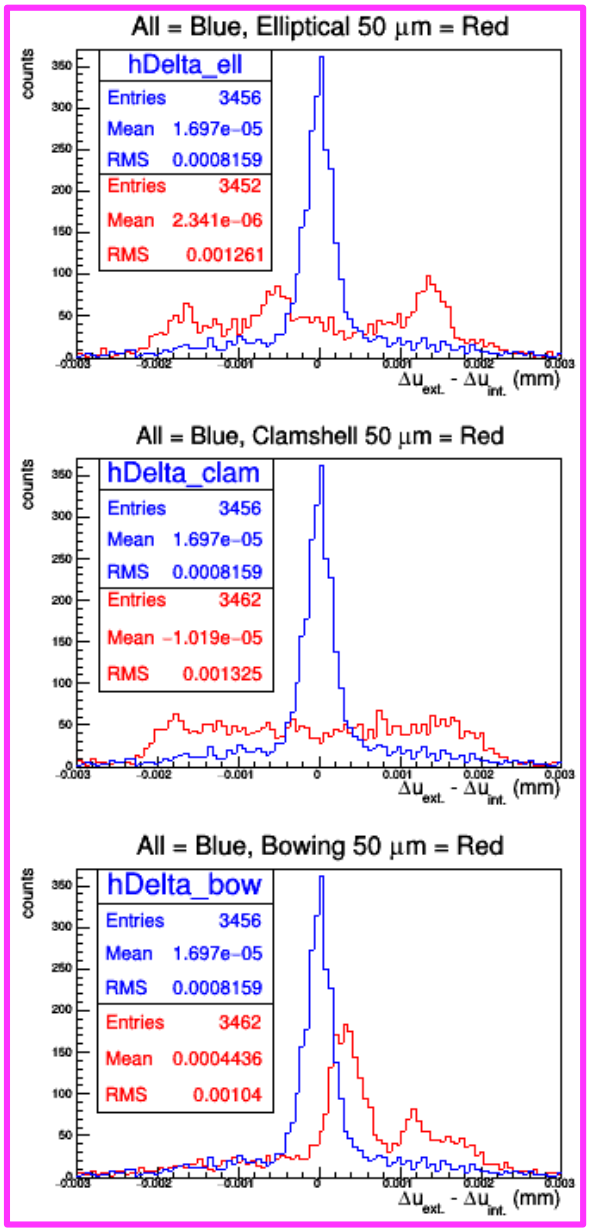
Shifts (<u>z-expansion and telescope</u>) and large broadenings (<u>twist, skew, elliptical expansion</u>) observed.

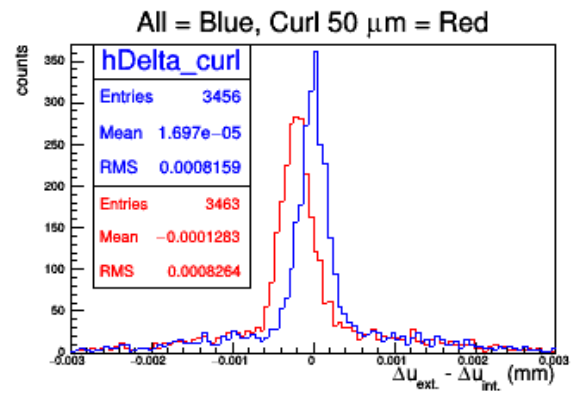
Realistic misalignments: 50 µm, u-residuals.

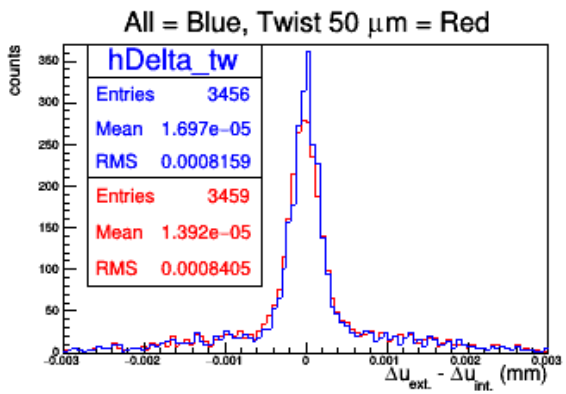
$\Delta u_{ext.} - \Delta u_{int.}$ comparison for aligned or misaligned VXD (by up to 50 µm) according to each weak mode.

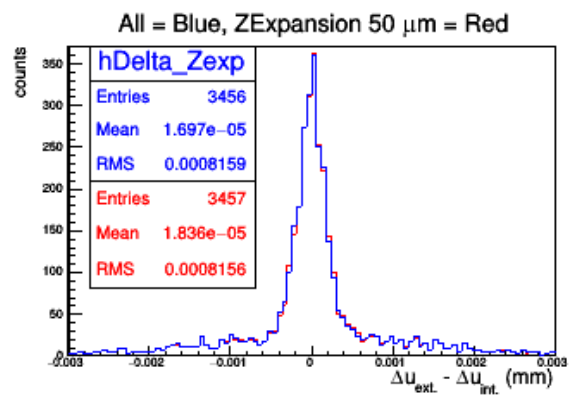


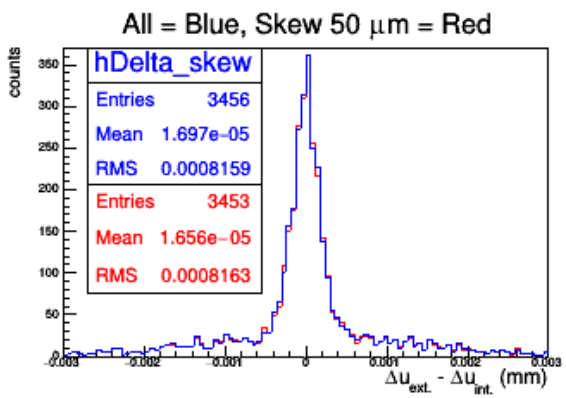








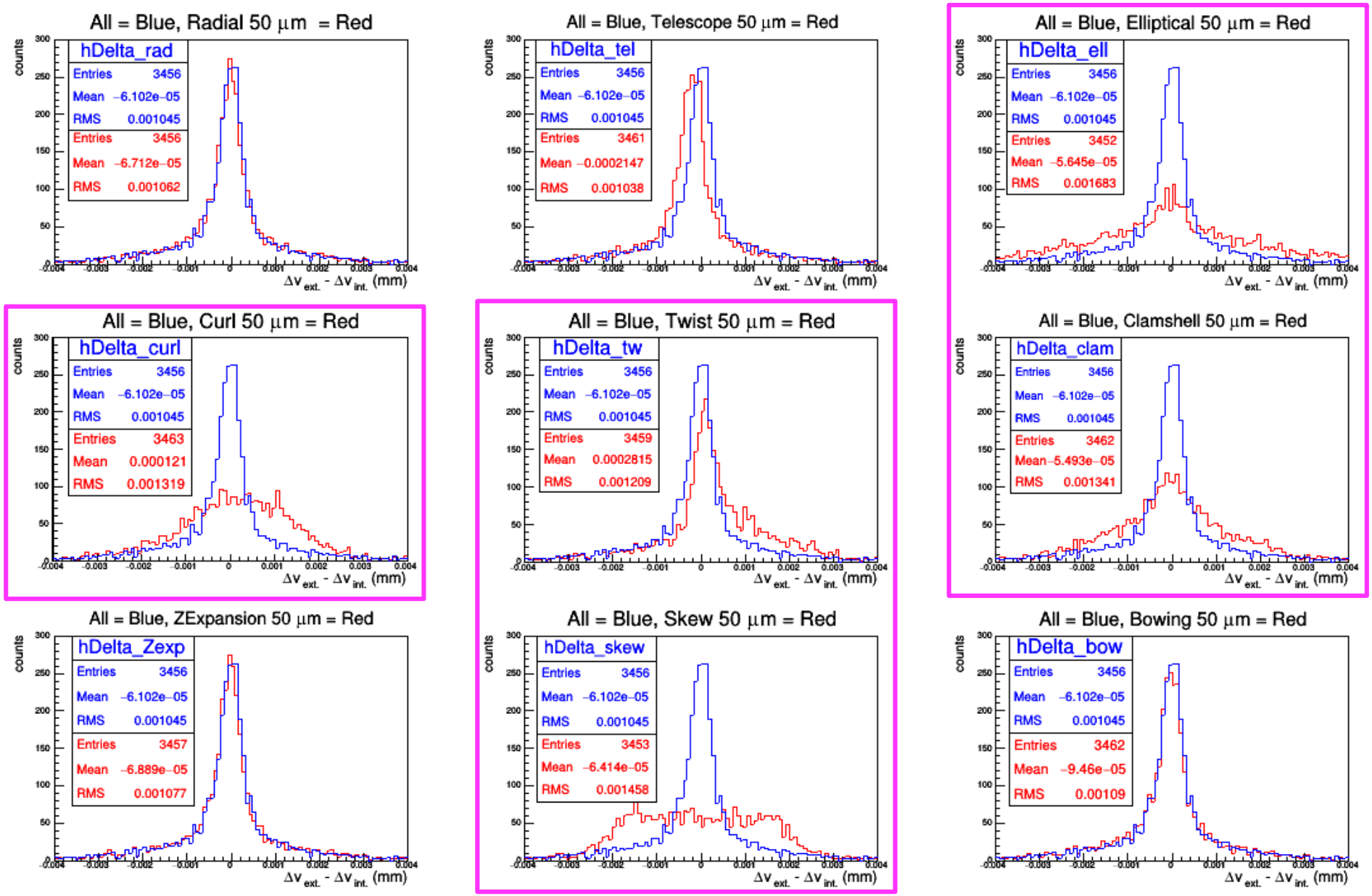




Distinctive features for <u>radial expansion, elliptical expansion, clamshell, bowing.</u>

Realistic misalignments: 50 μm, v-residuals.

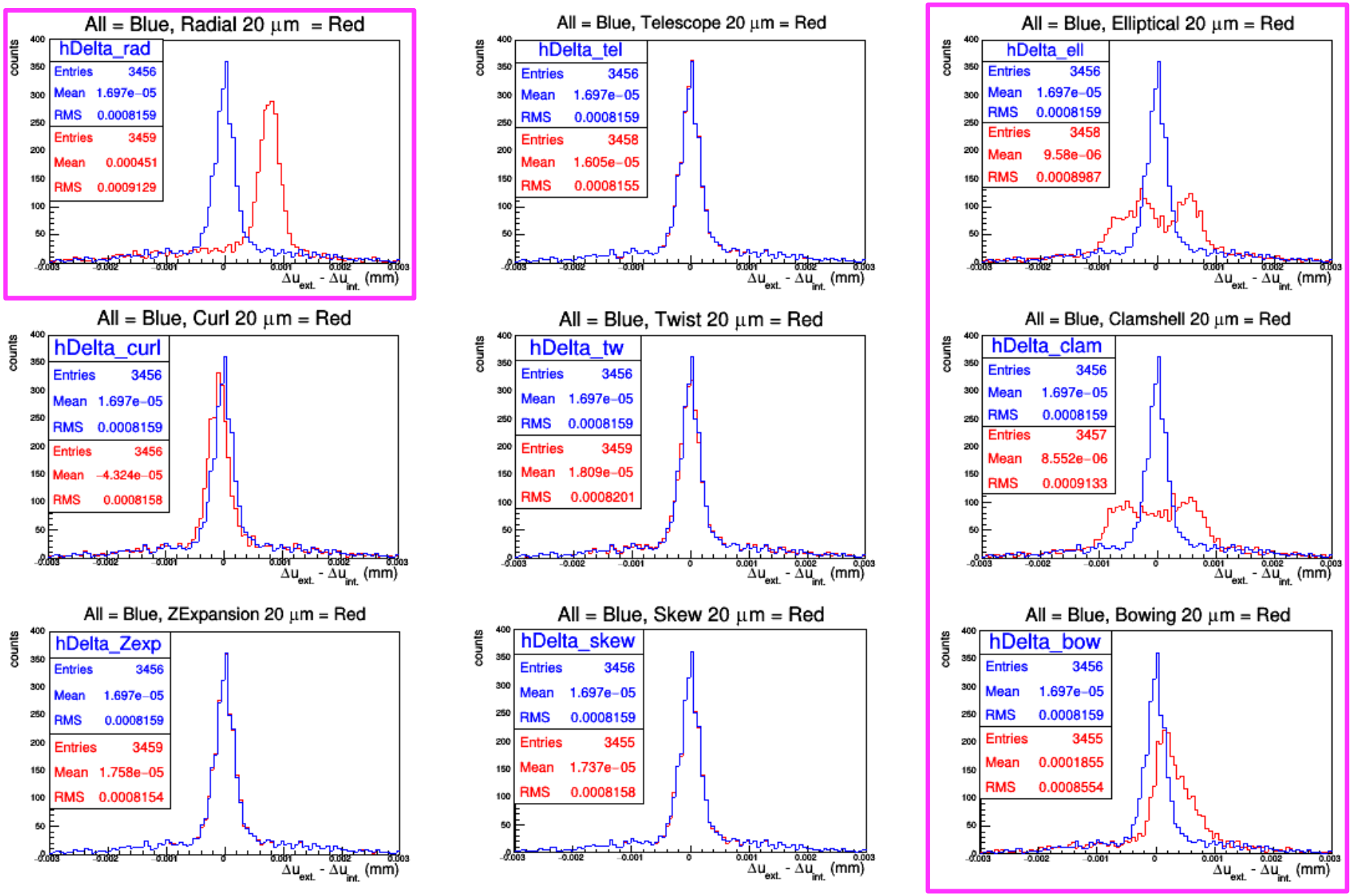
$\Delta v_{ext.} - \Delta v_{int.}$ for aligned or misaligned VXD (by up to **50 μm**) according to each weak mode.



Distinctive features for <u>curl, twist, skew, clamshell and elliptical expansion.</u>

Realistic misalignments: 20 μm, u-residuals.

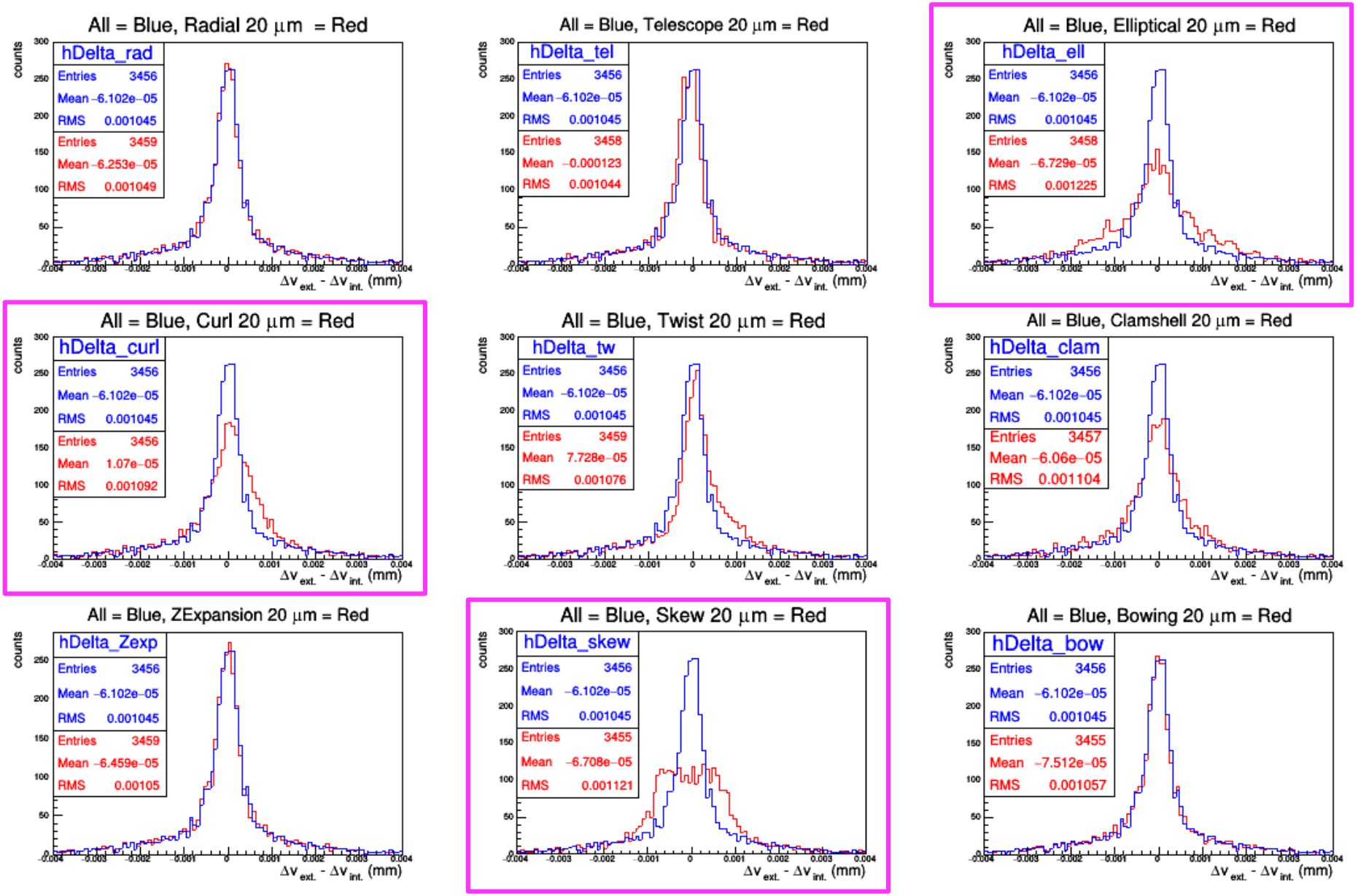
$\Delta u_{ext.} - \Delta u_{int.}$ comparison for aligned or misaligned VXD (by up to 20 μm) according to each weak mode.



Distinctive features for <u>radial expansion, elliptical expansion, clamshell, bowing.</u>

Realistic misalignments: 20 μm, v-residuals.

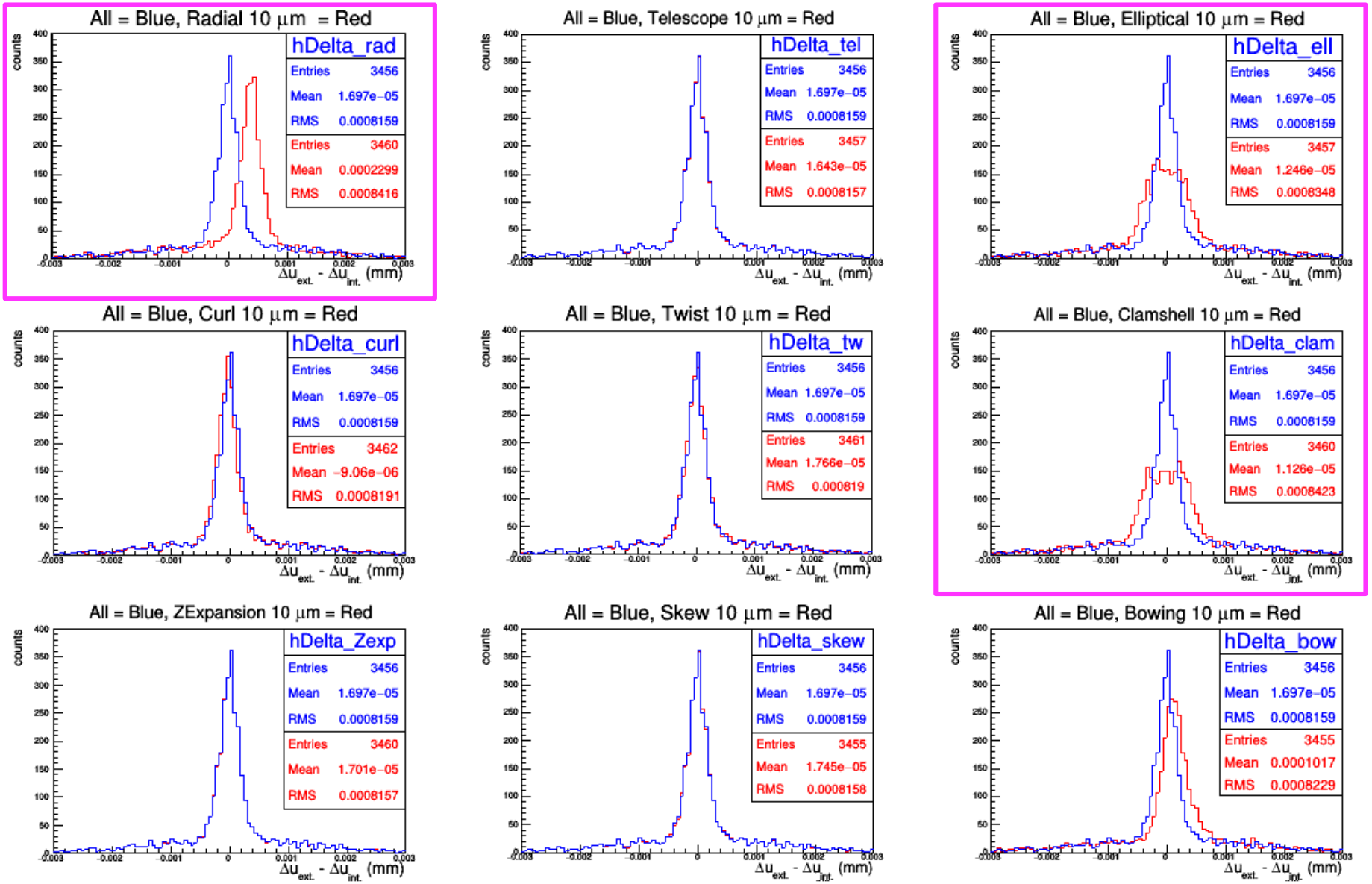
$\Delta v_{ext.} - \Delta v_{int.}$ for aligned or misaligned VXD (by up to **20 μm**) according to each weak mode.



Distinctive features for <u>elliptical expansion, curl and skew.</u>

Realistic misalignments: 10 µm, u-residuals.

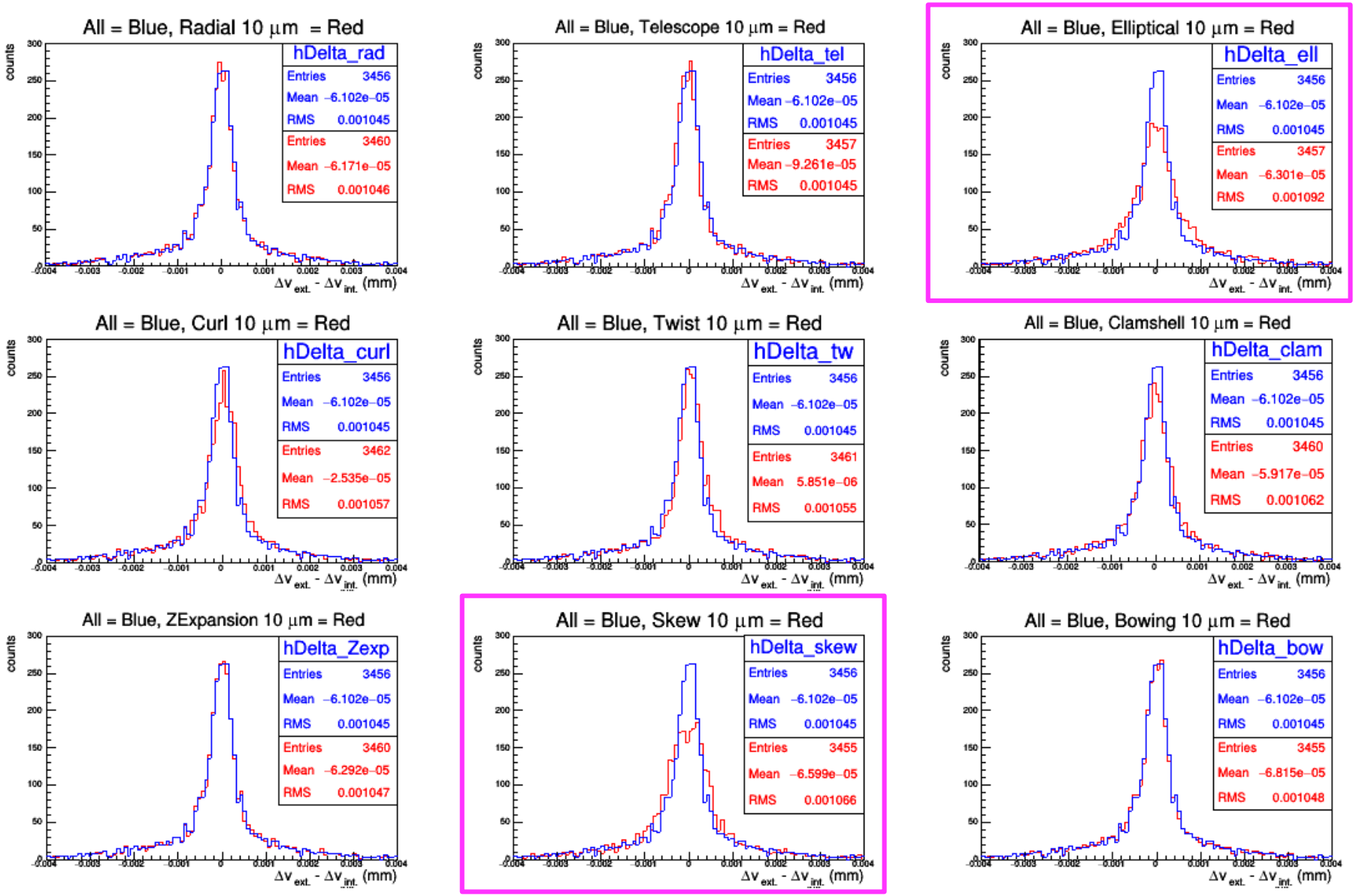
$\Delta u_{ext.} - \Delta u_{int.}$ comparison for aligned or misaligned VXD (by up to 10 µm) according to each weak mode.



Distinctive features for <u>radial expansion, elliptical expansion and clamshell.</u>

Realistic misalignments: 10 μm, v-residuals.

$\Delta v_{ext.} - \Delta v_{int.}$ for aligned or misaligned VXD (by up to **10 μm**) according to each weak mode.



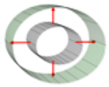
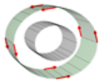
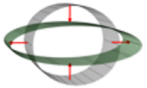
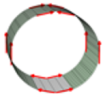
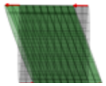
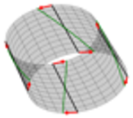
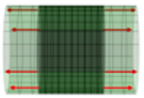
Distinctive features observed <u>only in case of skew and elliptical expansion.</u>

Global picture

✓ : highly sensitive to the weak-mode.

✓ : poorly sensitive to the weak-mode.

✗ : invariant to the weak-mode.

	Radial expansion $\Delta r = c_{scale} \cdot r$ 	Curl $r\Delta\phi = c_{scale} \cdot r + c_0$ 	Telescope $\Delta z = c_{scale} \cdot r$	Elliptical expansion $\Delta r = c_{scale} \cdot \cos(2\phi) \cdot r$ 	Clamshell $\Delta\phi = c_{scale} \cdot \cos(\phi)$ 	Skew $\Delta z = c_{scale} \cdot \cos(\phi)$ 	Bowing $\Delta r = c_{scale} \cdot \lvert z \rvert$	Twist $r\Delta\phi = c_{scale} \cdot z$ 	Z expansion $\Delta z = c_{scale} \cdot z$ 
10 μm									
$res_{(u)}$	✓	✓	✗	✓	✓	✗	✓	✗	✗
$res_{(v)}$	✗	✗	✗	✓	✗	✓	✗	✗	✗
20 μm									
$res_{(u)}$	✓	✓	✗	✓	✓	✗	✓	✗	✗
$res_{(v)}$	✗	✓	✗	✓	✓	✓	✗	✓	✗
50 μm									
$res_{(u)}$	✓	✓	✗	✓	✓	✗	✓	✓	✗
$res_{(v)}$	✗	✓	✓	✓	✓	✓	✗	✓	✗

Summary

I developed a **procedure for early identification of realistic VXD weak-mode misalignments.**

This procedure may supplement the standard alignment algorithm of Belle II and be used during on-line reconstruction.

Looking at track hits in overlapping sensors, for realistic 10 – 50 μm weak mode misalignments, **I can spot eight characteristic misalignments out of nine**, using either *u* or *v* residual differences.

The only weak mode that cannot be identified is z-expansion, due to the fact that it shifts coherently all sensors in a same VXD layer and keeps the alignment almost unmodified in the region where sensors overlap.

Residual algorithm

The goal is to compute residual differences $\Delta u_{ext.} - \Delta u_{int.}$ and $\Delta v_{ext.} - \Delta v_{int.}$, for two consecutive track hits in two overlapping modules of a same VXD layer.

To do so I developed a C++ module, **MuPullStudiesModule**, to be added in the python script performing reconstruction, that works as follow:

- Restrict to events where only 2 tracks are reconstructed;
- Loop on the reconstructed tracks;
- Skip to next track in loop if fit on a track is not successful;
- For each track in the loop build:
  - the vector of PXD clusters;
  - the vector of SVD clusters
  - the vector of informations on reconstructed hits;
- Loop on the PXD clusters;
- For each PXD cluster get the information (getRecoHitInformation) on the corresponding reconstructed hit;
- If no information is available, skip to the next PXD cluster;

Residual algorithm

- Get the track point corresponding to the information;
- Get fit result (getFitterInfo) for the track point;
- If no fit result is available, skip to the next PXD cluster;
- Get fit vector on plane linked to information on reconstructed PXD cluster (getMeasuredStateOnPlaneFromRecoHit);
- Get u and v coordinates from fit vector (getState)
- Get u and v measured coordinates of PXD cluster and identify the measuring sensor (getSensorID);
- Get sensor layer (getLayerNumber), ladder (getLadderNumber) and number (getSensorNumber);
- Fill a tree for PXD hits with u, v measured and fitted coordinates and sensor number, layer, ladder;
- Loop on SVD clusters, repeat same steps of loop on PXD, and fill a tree for SVD hits;

Residual algorithm

- Read PXD tree: get all information for a hit (int-hit) and the next one (ext-hit);

- If int-layer = ext-layer, |int-ladder – ext-ladder| = 1, |int-u – ext-u| < 0.5 mm and |int-v – ext-v| < 0.5 mm (hits in overlapping modules): compute the difference of residuals $\Delta u(v)_{ext.} - \Delta u(v)_{int.}$ where $\Delta u(v) = u(v)_{meas} - u(v)_{fit}$.

- Compute residual differences also for PXD special cases:
  - PXD layer = 1, int-ladder = 1 and ext-ladder = 8;
  - PXD layer = 2, int-ladder = 1 and ext-ladder = 12.

- Read SVD tree and compute residual differences if int-layer = ext-layer, |int-ladder – ext-ladder| = 1, |int-u – ext-u| < 1 mm and |int-v – ext-v| < 1 mm.

- Compute the difference of residuals also for SVD special cases:
  -SVD layer = 3, int-ladder = 7 and ext-ladder = 1;
  -SVD layer = 4, int-ladder = 10 and ext-ladder = 1;
  -SVD layer = 5, int-ladder = 12 and ext-ladder = 1;
  -SVD layer = 6, int-ladder = 16 and ext-ladder = 1;

- Fill histogram of residual differences.

Generation, simulation and reconstruction procedure

- Generate 10^4 $e^+e^- \to \mu^+\mu^-$ events using KKMC generator and simulate each event.

- **For the aligned configuration:** reconstruct each event using alignment information from the default database.

- **For weak modes:** create 9 directories (each corresponding to one weak mode); in each directory run the python code misalignVXDSystematic.py, and create a local database (localdb) containing misaligned VXD geometry; reconstruct events in misaligned VXD geometry.

- Change the magnitude of misalignments using an input parameter for misalignVXDSystematic.py.

- Run in parallel 9 reconstructions (one for each mode) on a same simulated sample, for a specific maximum misalignment, using KEKCC batch system.