Improved track reconstruction in the testbeam-analysis framework

Mihaela Pehlivanova

August 2023

Supervisors: Nazar Bartosik, Simone Gennai

Contents

1 Introduction

The Large Hadron Collider (LHC) is currently undergoing preparations for a significant upgrade known as the High Luminosity LHC (HL-LHC). This upgrade aims to achieve higher instantaneous luminosity levels, specifically in the range of $5-7\times10^{34}cm^{-2}s^{-1}$. By the end of 2037, the HL-LHC project targets an integrated luminosity of 3000 fb^{-1}. This increase in luminosity will result in more frequent and intense collisions within the particle accelerator.

With the upgraded luminosity, the LHC will be able to accommodate up to 200 simultaneous collisions occurring every 25 ns. These collisions will lead to the production of approximately 10,000 charged particles per bunch crossing. As a consequence of these enhanced collision conditions, the detectors such as the Compact Muon Solenoid (CMS) detector, will require comprehensive upgrades to ensure their capability to handle the increased radiation levels and data rates.

The Phase-2 upgrade for the CMS detector will necessitate the complete replacement of its tracker system. This tracker system is responsible for precisely tracing the paths of charged particles produced in the collisions, as well as accurately identifying the locations of collision vertices. This upgrade is required to maintain and improve the tracking and vertexing capabilities of the CMS detector under the demanding conditions of the HL-LHC.

2 CMS Inner Tracker system

The Inner Tracker System, located closest to the interaction point within the detector, is planned for an upgrade in preparation for the higher radiation levels resulting from the LHC's increased luminosity[1]. The system will encounter radiation levels leading to an accumulated total ionizing dose of up to 1.2 Grad. Consequently, the entire tracker system must be enhanced for radiation hardness to withstand these conditions. Furthermore, the upgraded tracker must be equipped to manage the elevated data rates generated by the intensified collisions.

One crucial feature of the upgraded system is its extended coverage in terms of pseudo-rapidity (η), reaching up to $\eta = 4$. This expansion in tracking volume enables a more comprehensive understanding of particle behavior across a wider range of angles and energies.

To ensure the readiness of the new detector, a series of test beam measurements have been conducted. These measurements employ beam telescopes in various facilities to simulate the irradiation conditions that the detector will encounter during its final deployment. The measured pixels are of dimensions $25 \times 100\ \mu m^2$ and $50 \times 50\ \mu m^2$ (shown on fig. 1). These prototypes are subjected to similar conditions as those expected in the operational environment, enabling accurate testing, qualification, and characterization in terms of efficiency, spatial resolution, and timing performance.

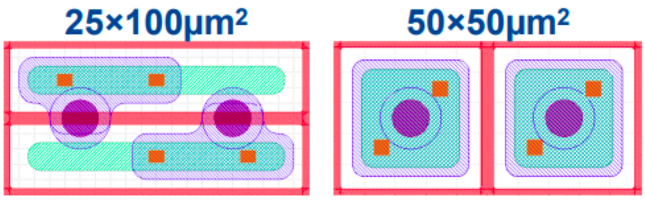


Figure 1: Internal Tracker sensor cells.The p-stop inter-pixel isolation areas are shown in orange, the n-doped implants in green, the metal layer in light purple, the bump pad opening in purple. The orange squares indicate the contact openings. [1]

By subjecting the prototypes to test beam measurements and evaluating their efficiency, spatial resolution, and timing performance, researchers can ascertain the detector's functionality and reliability under conditions similar to the actual deployment scenario. The testing and characterization process is imperative for ensuring that the upgraded Inner Tracker System of the CMS detector is well-equipped to handle the demanding conditions of the High Luminosity LHC and to maintain the precision and accuracy necessary for particle tracking and vertexing.

3 Test beam telescope

In the setup (fig.2), we have a series of detector planes arranged along the z-direction, which aligns with the path of the particle beam. Due to practical limitations, these planes might be slightly misaligned. As a particle moves through the material, it generates both electrons and electron vacancies called 'holes.' However, due to the bigger drift velocity of electrons compared to holes, the signal primarily consists of electrons. [4]

Regardless of where the particle traveled through the pixel, the track is typically reconstructed near the pixel's center. If multiple pixels were activated, a more accurate position can be determined by considering the charge distribution across them. Analyzing the tracks on each plane, we ideally expect a straight line, but real-world misalignments cause the track to exhibit a zig-zag pattern.

The concept of alignment aims to reduce the residuals observed on each plane by analyzing a significant number of tracks. This involves iteratively adjusting the position and rotation of each plane. This process is then repeated multiple times, gradually tightening the track-quality criteria, with the goal of ultimately achieving precise alignment (fig.3). Large residuals can undermine the accuracy

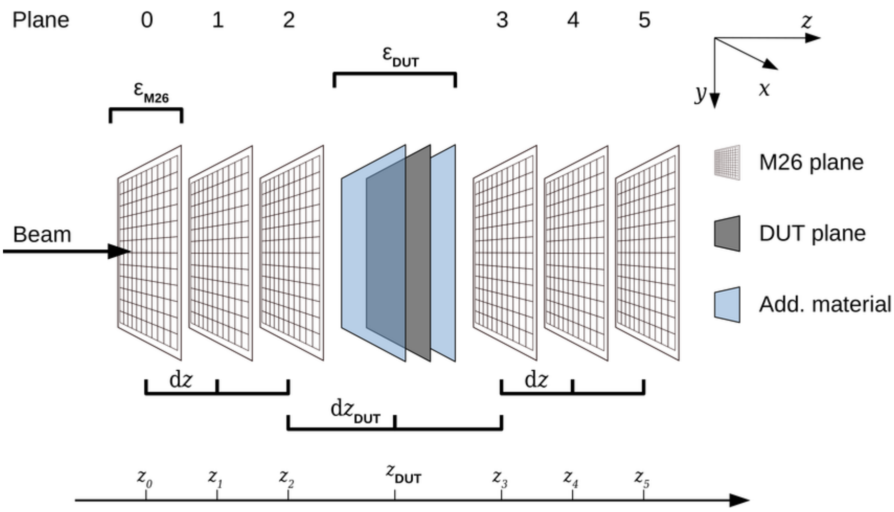


Figure 2: Set up of MIMOSA Telescope. [2]

of the reconstruction. By carefully adjusting the positions of the planes, we can significantly improve the quality of the reconstructed track.

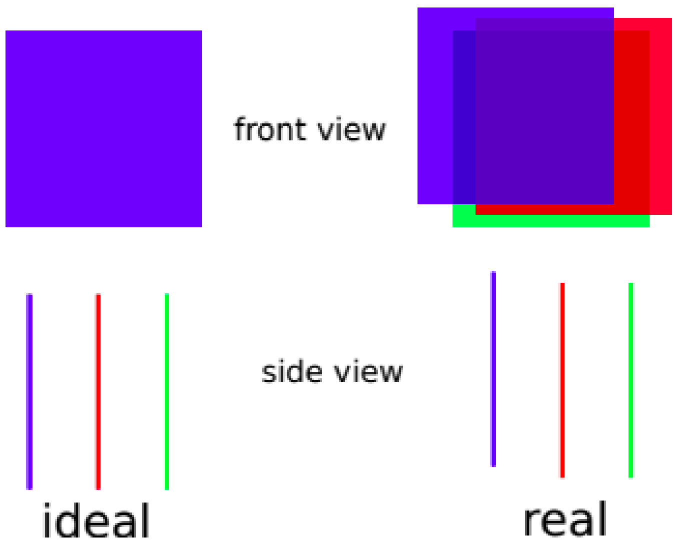


Figure 3: Real and ideal misalignment of the sensor planes (exaggerated). Different telescope planes are indicated in different colours. [4]

Depending on the angle (fig.4) at which the particle crosses and the direction of charge drift in the Silicon material there can be different clusters of pixels. However, as a general trend, a particle crossing at the center is more likely to result in a 1-pixel cluster, while a crossing closer to the edge is likely to generate a 2-pixel cluster. Employing a 'cluster size' of 2, where two adjacent pixels are considered, can lead to improved hit reconstruction. Combining hits from each plane in these neighboring pixels, we can more accurately pinpoint the particle's path by weighing the signal which was left in the two pixels.

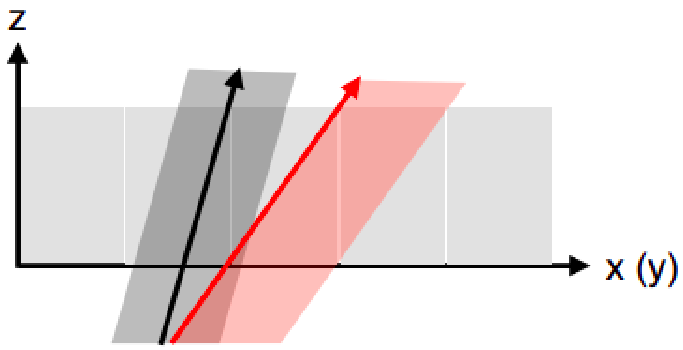


Figure 4: In a pixel detector's side view, arrows represent potential MIP (Minimum Ionizing Particle) paths. The red and black shaded areas depict different scenarios. The black arrow's angle allows it to cross one or two pixels. When particle enters to the left of the black arrow, it is more likely to fire one pixel, if it traverse on the right - it is more likely to fire two pixels. On the other hand, the red arrow always crosses two pixels. [5]

4 Corryvreckan

Corryvreckan is framework written in modern C++ programming. Its structure is like building blocks that you can easily rearrange. Specialized sections handle specific jobs within the reconstruction process. To share information, these sections pass data back and forth using a central storage space called a clipboard. [3] In the configuration files, you can add different modules for each specific task that needs to be done during the analysis. These modules are independent from the core framework, which means you can easily add or remove them to create the analysis sequence you want. You can see a common example of this sequence in fig.5.

4.1 Event Building

Corryvreckan uses a set of modules called EventLoaders [3] to prepare information about pixel or strip hits from various detectors. These modules decode raw

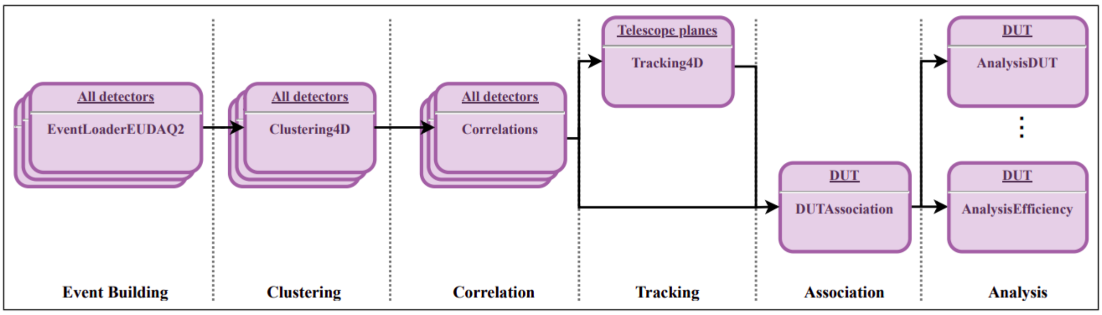


Figure 5: A typical Corryvreckan reconstruction chain with a beam telescope setup and a single DUT. The dots in the Analysis stage indicate the possible inclusion of further analysis modules. [3]

data or apply calibrations to determine hit positions. There are detector-specific EventLoader modules, as well as EventLoaderEUDAQ and EventLoaderEUDAQ2 for reading data from detectors.

The first module in the reconstruction sequence sets the range of data being processed, defined by start and end timestamps or associated trigger numbers. All following modules must work within this event definition, providing data that matches it. Another key module, FileReader, lets you read intermediate results from a previous reconstruction or simulation run.

4.2 Clustering

Corryvreckan has two clustering methods. One method searches for nearby hits to create clusters, while the other considers hit timestamps to group hits accurately even in high-rate scenarios. Once clusters are formed, hit positions are calculated. If charge information is available, a charge-weighted center-of-gravity algorithm is used. Otherwise, for binary hit information, the hit position is calculated as the average.

4.3 Track Finding and Fitting

Corryvreckan splits the track reconstruction into two steps: finding track candidates and fitting particle tracks. The track finding process is managed by Tracking modules, which use different track models for fitting. In this process, a track model is chosen based on the configuration, and a track candidate is assembled using hits from relevant detectors.

Different Tracking modules are available. The Tracking4D [3] module initially constructs a track candidate using two telescope planes and later includes more clusters from additional planes within a defined spatial and time window.

Corryvreckan supports three track models: a simple straight-line track, defined by a reference position and a specific direction, multiplet track model, which comprises two tracklets joined by a single kink and General Broken Line (GBL). This model accommodates kinks at every detector and takes into account multiple Coulomb scattering across all detector planes and even the surrounding medium. In the frame of this project we used the straight-line track model.

4.4 Association

A module associates clusters from the device under test (DUT) with reference tracks from telescope planes.This allows for the independent handling of multiple detectors as DUTs, each with unique criteria and settings. Multiple clusters from a single DUT can be linked to a track, and the closest cluster in space can be chosen for analysis.

The DUTAssociation [3] module offers two ways to associate DUT clusters with tracks. One method involves comparing the cluster center position with the track position at the DUT, using a distance-based rule. The second option decides based on the nearest distance between the track and any of the pixel hits within the cluster. This approach is helpful for recovering large clusters, including those caused by delta rays that might pull the cluster center far from the track incidence point.

In this reconstruction phase, timing information can be used to minimize discrepancies between the particle track and the DUT cluster, further enhancing accuracy.

4.5 Alignment

Corryvreckan offers a two-step alignment process. After each step, an updated geometry file which holds information as position of the specific plane and reference if the plane is considered a DUT or not is generated. This file can be used as input for subsequent alignment steps and the final analysis. The different steps of alignment are:

Prealignment: this step focuses on getting the detector components into a rough initial alignment state. It's necessary when the misalignment is significant and aims to make the detectors usable for further analysis. A prealignment is performed by calculating correlations among all detectors and a reference plane. The obtained residuals are then centered around zero by shifting individual detector positions in the global x and y coordinates. This initial prealignment, with loose criteria, enables tracking.

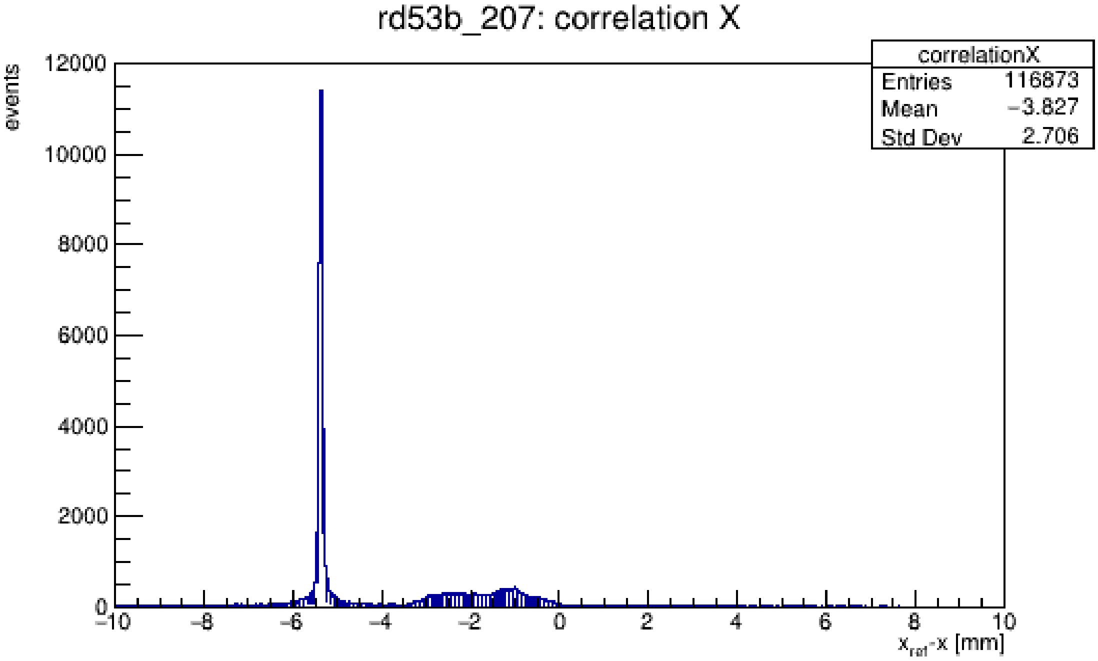


Figure 6: Correlation plot for X axis of DUT plane rd53b_20 with initial input geometry.

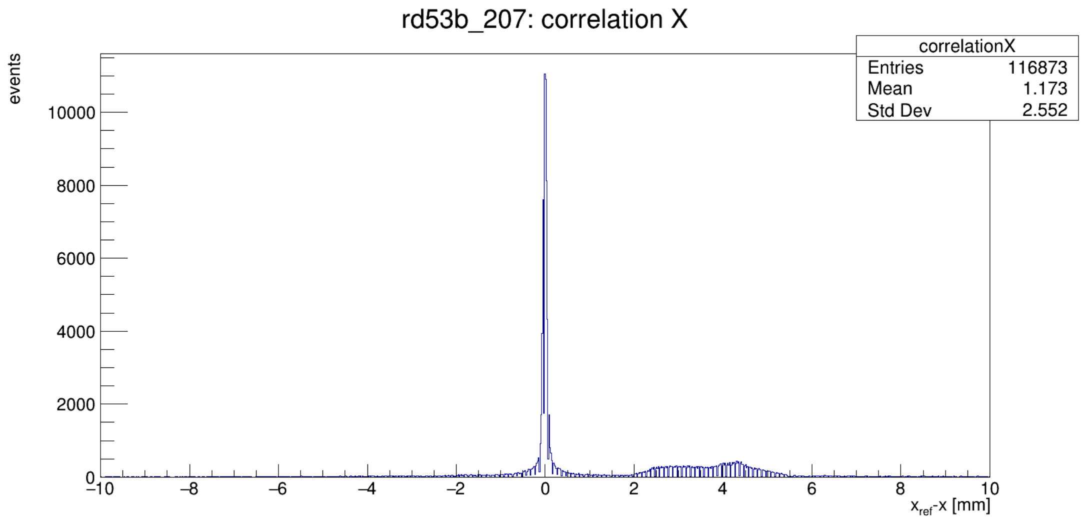


Figure 7: Correlation plot for X axis of DUT plane rd53b_207 after prealignment step. We can see that the peak is better centered around 0.

Alignment of the Telescope: here, the primary goal is to achieve precise alignment of the tracking detector planes within the telescope. It ensures that all telescope detector planes are correctly aligned with each other and parallel to the particle beam. This alignment is based on well-calibrated reference tracks.

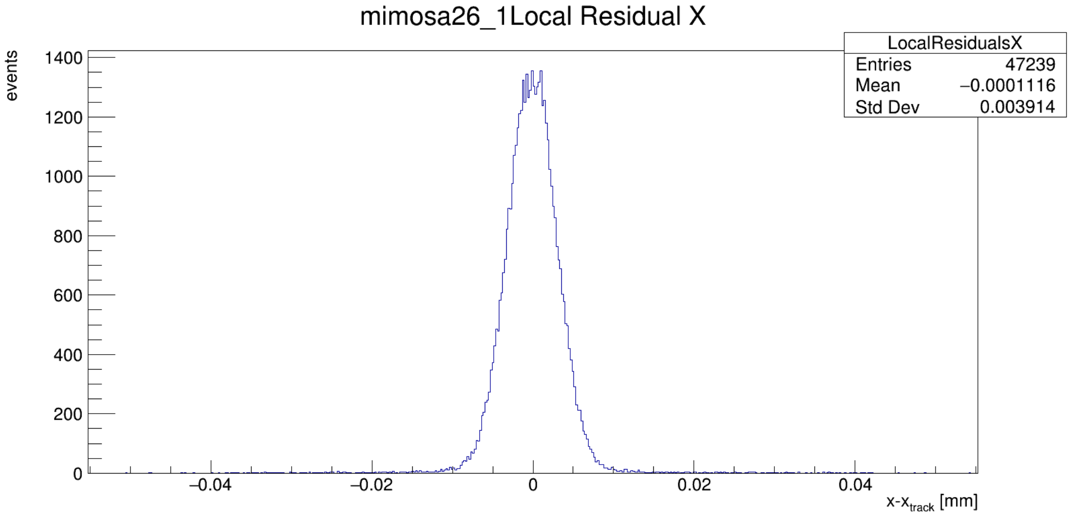


Figure 8: Plot for local X axis of mimosa26_1 plane before tightening the criteria.

Parameters we used for the first step of the alignment of the telescope in the modules `Tracking4D` and `AlignmentTrackChi2`:

```
[Tracking4D]
log_level = INFO
track_model = "straightline"
momentum = 75.0GeV
min_hits_on_track = 5
spatial_cut_abs = 500um, 500um
time_cut_abs = 200ns
exclude_dut = true

[AlignmentTrackChi2]
log_level = INFO
align_position = true
align_orientation = true
prune_tracks = true
max_associated_clusters = 1
max_track_chi2ndof = 200
```

Tightened parameters for the same modules of the last step of alignment of the telescope. We are decreasing the `spatial_cut_abs` and `max_track_chi2ndof`.

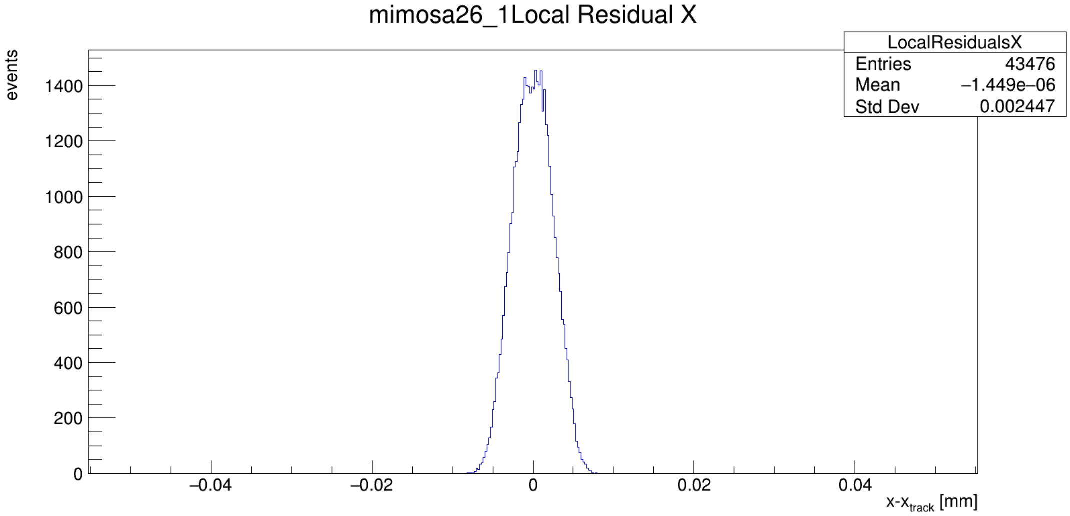


Figure 9: Residuals plot for local X axis of mimosa26_1 plane after tightening the criteria.

```
[Tracking4D]
track_model = "straightline"
momentum = 75GeV
min_hits_on_track = 5
spatial_cut_abs = 10um, 10um
time_cut_abs = 200ns
exclude_dut = true

[AlignmentTrackChi2]
align_position = true
align_orientation = true
prune_tracks = true
max_associated_clusters = 1
max_track_chi2ndof = 5
```

Alignment of the DUT (Device Under Test): DUT alignment optimizes the alignment of the DUT detector planes to achieve high measurement accuracy specifically for the DUT. It assumes that the telescope is already correctly aligned and focuses on minimizing the differences (residuals) between measured particle trajectories through the DUT and reference tracks from the aligned telescope. We talk more about DUT alignment in section 5.1.2.

4.6 Analysis

The toolkit supports modular analysis for evaluating detector performance. Modules like `AnalysisEfficiency`, `AnalysisSensorEdge`, and `AnalysisDUT` [3] assess efficiency, spatial resolution, and more. Users can also contribute custom analysis modules.

5 Improving track reconstruction in the testbeam-analysis framework

The measured sensors are with size 25 x 100 μm^2. Taking the fact that pixels are uniformly distributed the resolution for X and Y axes is:

$$\sigma_x = \frac{l_x}{\sqrt{12}} \qquad \sigma_y = \frac{l_y}{\sqrt{12}} \tag{1}$$

Their asymmetry leads to a significant difference, by a factor of 4, in the local resolutions along the X and Y axes.In the axis with the higher resolution (here Y), even slight changes in particle position can be accurately discerned. This implies that the reconstructed position of the particle in the Y direction would be quite precise.

However, on the axis with the lower resolution (in this case, X), the detector is less capable of distinguishing small shifts in position. As a result, the reconstructed position in the X direction may be more prone to errors and uncertainties.

Now, when it comes to reconstructing the overall trajectory of the particle, which essentially involves combining its X and Y positions, the differing resolutions play a critical role. The precision of the Y position measurement could potentially outweigh the accuracy of the X position measurement due to the resolution disparity. This can lead to a skewed or biased trajectory reconstruction that is more sensitive to changes along the higher-resolution axis.

5.1 Main idea

Our focus lies in selecting tracks that exhibit a larger cluster sizes, meaning those comprising more than just one pixel. Also the focus lies in local X direction, where the resolution is worse because of the bigger pitch size.

To elaborate, when we examine the scenario with the smallest cluster size - consisting of only two pixels - we encounter a situation where both pixels are jointly registering the charge deposited by a particle. Subsequently, the charge-weighted center-of-gravity algorithm comes into play to determine the precise particle position. The resultant reconstructed hit position will exhibit a bias towards the pixel with the greater charge accumulation.

We implemented that idea for the module *AlignmentDUTResidual* introducing new variables in the source code for which the track selection to be made for. We modified the source code of the *AlignmentDUTResidual* module to include additional variables that play a role in determining which tracks are selected for further processing or analysis. By introducing these new variables, the module can filter and manipulate tracks based on specific criteria - minimum rows, columns and a specific predetermined plane on which the filtering operation is focused. Part of the filtering we can see below:

```
<std::vector<Cluster*> cluster_v;
if(std::find(clusterFilter_detectors.begin(),
                clusterFilter_detectors.end(), m_detector->
                getName()) != clusterFilter_detectors.end()){
    for(auto cluster : associated_clusters){
        if(cluster->rowWidth() < clusterFilter_min_rows ||
                cluster->columnWidth() < clusterFilter_min_columns){
                continue;
        }
    }
}
```

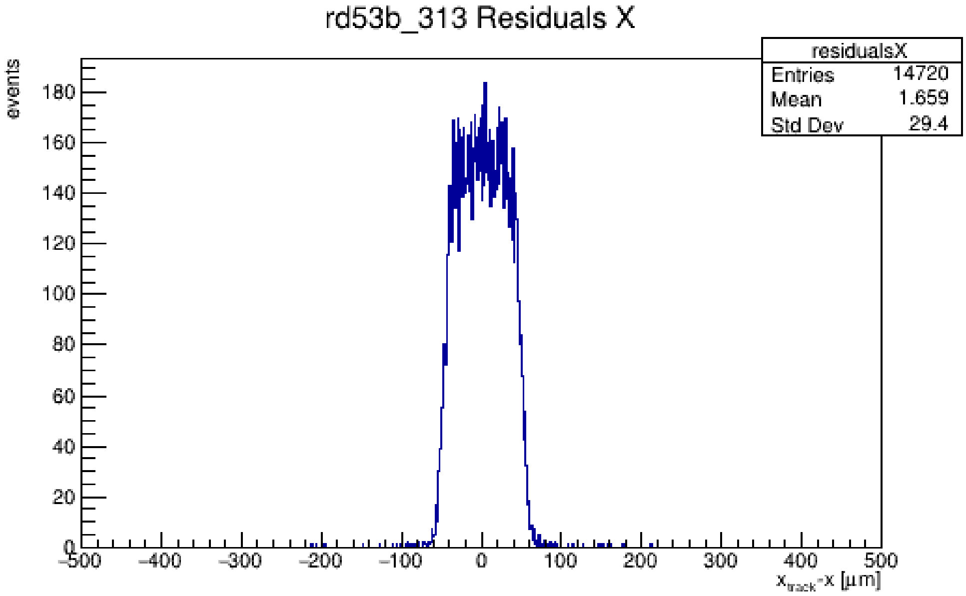
And in the configuration file for the alignment of DUT:

```
clusterFilter_min_rows = 1
clusterFilter_min_columns = 2
clusterFilter_detectors = "rd53b_313"
```

5.1.1 Encountered issues in the framework

While attempting to implement cluster filtering, we identified and fixed a bug within the code of the *EventLoaderEUDAQ2* module. Specifically, the issue resides in the `get_pixel_data` function, where data pertaining to a pixel was being duplicated in the same vector. This unintended duplication resulted in inflated data and consequently skewed statistical metrics, rendering the statistics unreliable. We fixed that, but there were more issues.

5.1.2 DUT alignment



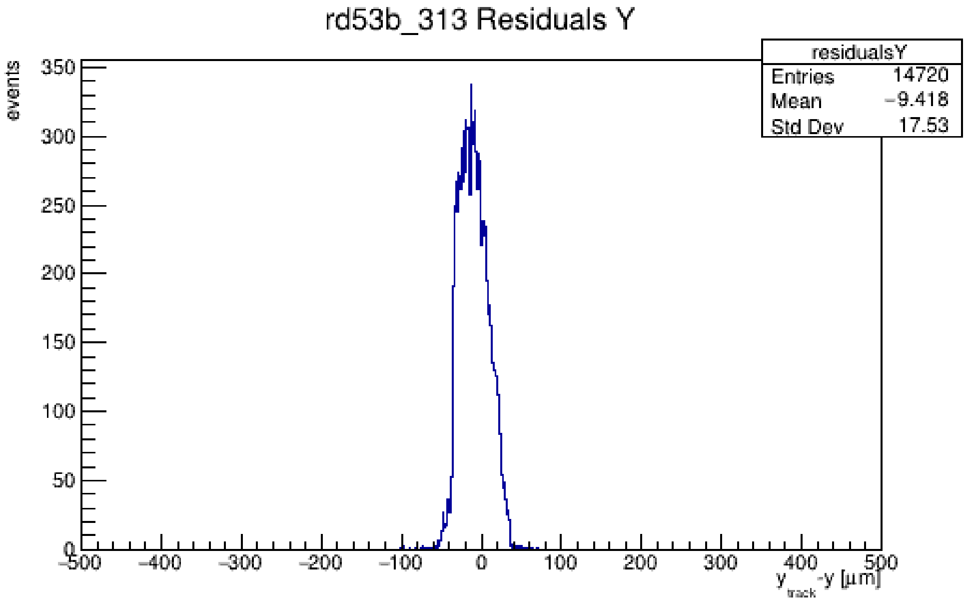


Figure 10: Residuals for detector rd53b without filtering. The upper histogram shows the residuals for X axis and the bottom one - for Y axis.

In Figure 10, it's evident that the peak is not centered around zero, indicating a misalignment of the Device Under Test (DUT) plane. For reference how the plot should look like we can point to fig.9 where the telescope is centered around zero.

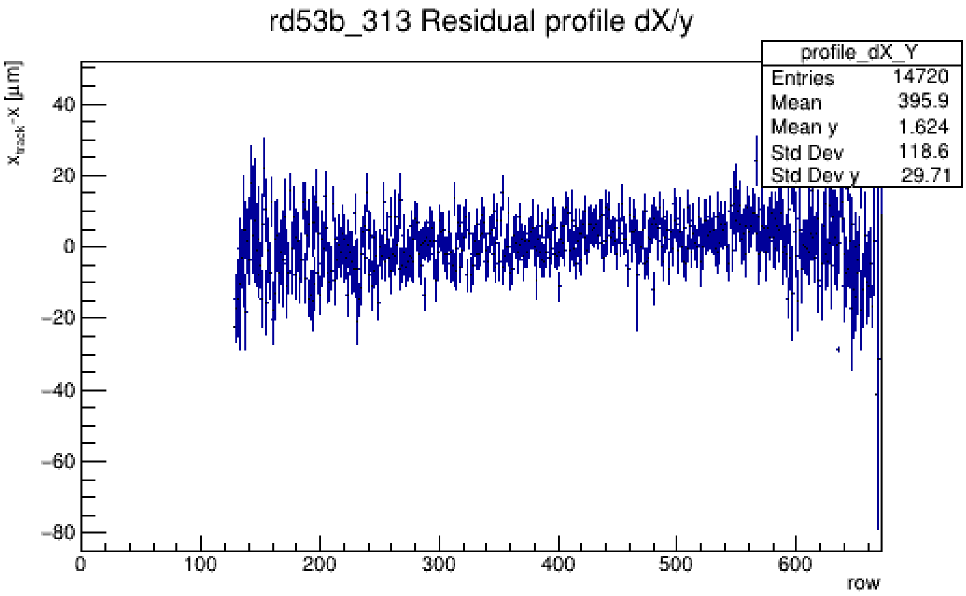


Figure 11: Residual profile dX/Y.

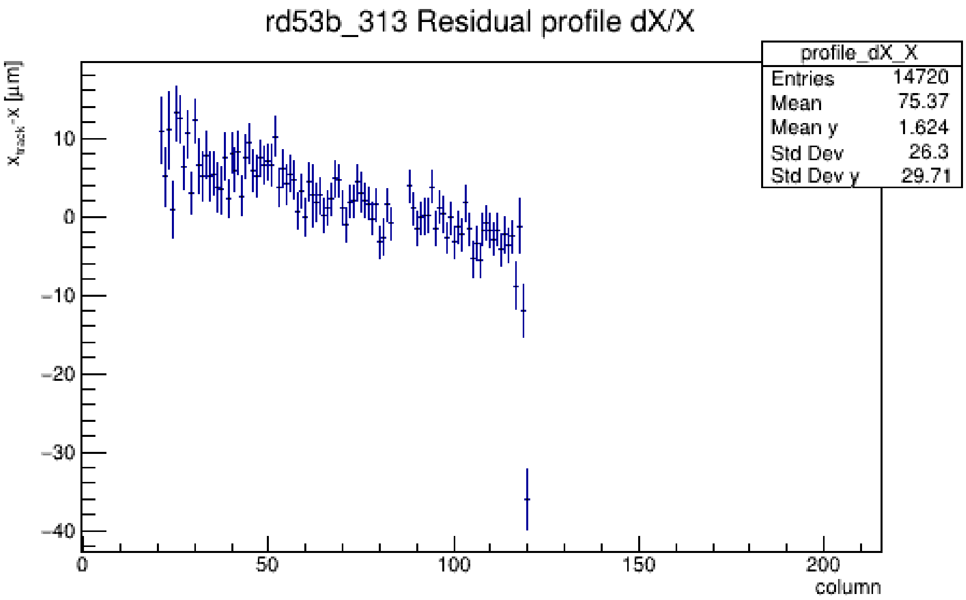


Figure 12: Residual profile dX/X.

During the alignment process, the goal is to make adjustments to the DUT's position or orientation in order to minimize any deviations in its output signal. The residual profile provides insights into how well the alignment procedure was

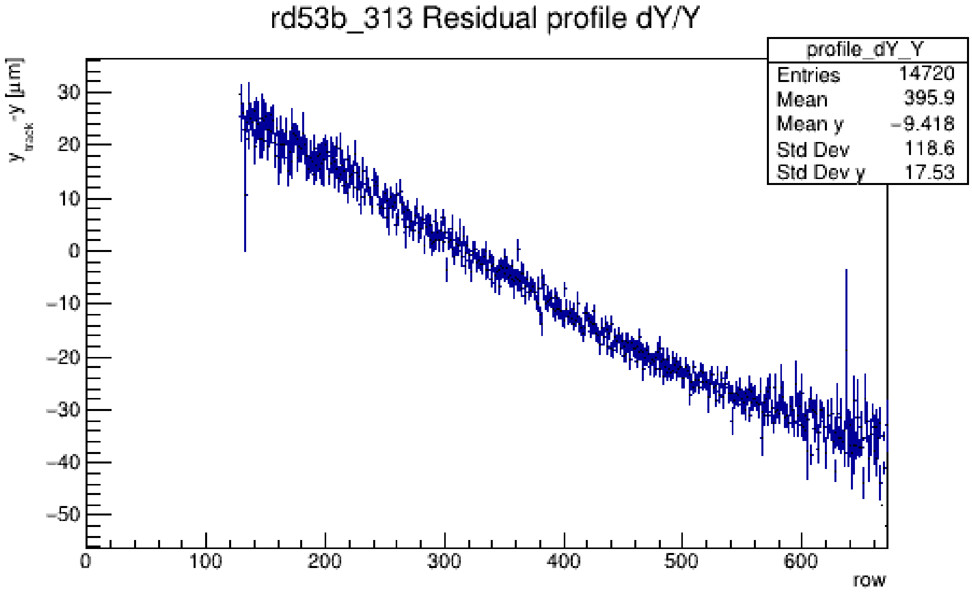


Figure 13: Residual profile dY/Y.

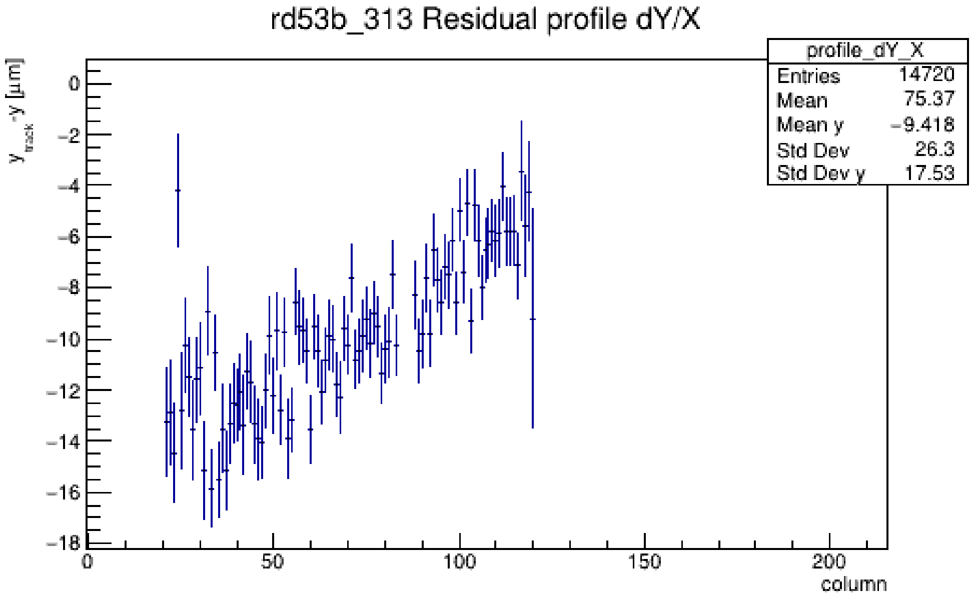


Figure 14: Residual profile dY/X.

performed. A value close to zero indicates a successful alignment, as the observed output signal is nearly identical to the expected signal. On the other hand, larger values away from zero indicate misalignment or deviations from the expected performance. If the residual profile is non-zero, it implies that there is still a misalignment issue present.

The histograms shown in Figure 12, Figure 13, and Figure 14 consistently exhibit a noticeable tilt, which strongly suggests the presence of a misalignment within the Device Under Test (DUT). This tilt remains persistent even after applying the alignment procedure.

In Figure 11, the plost is not sensitive to the kind of misalignment we have in the other plots. In summary, the alignment procedure of DUT planes is unuccessful and a comprehensive investigation is necessary to pinpoint the exact source of the misalignment and determine the appropriate corrective measures.

6 Acknowledgments

I would like to express my sincere gratitude to Nazar Bartosik and Simone Gennai for their invaluable mentorship during my time at CERN. Despite the challenges encountered in the course of this project, their guidance and support have been unwavering and greatly appreciated.

References

[1] Panja-Riina Luukka on behalf of the CMS Tracker Group, CMS Inner Tracker Upgrade, Proceedings of Science, 2020, vol. Vertex2019, p. 008, DOI: https://doi.org/10.22323/1.373.0008

[2] arXiv:1603.09669 [physics.ins-det]

[3] D. Dannheim et al 2021 JINST 16 P03008

[4] https://www.physi.uni-heidelberg.de/Einrichtungen/FP/anleitungen/F96.pdf

[5] arXiv:1711.00590 [physics.ins-det]