

Cell Center Finder for the NO ν A Far Detector Modules

Vladimir Kravtsov and Thomas Coan

(Dated: November 26, 2013)

I. INTRODUCTION.

This note describes the procedure that locates the cell center positions of empty PVC modules positioned on the block pivoter table as a function of distance along each cells 15.4-meter length using data taken by the laser scanning system at Ash River.

The NO ν A far detector comprises 28 blocks with 32 planes in each block. Each plane is assembled from 12 15.4-meter long modules that themselves contain 32 cells. Overall, each block and plane contains 12,288 and 384 cells, respectively. Successive planes (horizontal and vertical) are oriented perpendicularly to one another when they are stacked and glued on the block pivoter table during block construction until a block of 32 planes is completed. Modules are glued to other modules in the plane beneath them but not to their in-plane neighbors. Horizontal modules are tilted such that when a block is positioned in the hall for eventual data taking, a module's west end is rotated 1.25 mrad upwards with respect to its east end. Vertical modules are positioned on the pivoter table such that centers are offset from their nearest vertical neighbors by 0.625 inches. See P. Lukens' note "NO ν A Far Detector Parameters" in docdb (doc #2066) for useful mechanical details on the far detector blocks and modules.

For eventual data analysis (e.g., track and shower reconstruction) it is desired to know the relative position of cell centers for each of the cells, particularly within a 32-plane block, as a function of distance along a cell. Due to imperfect manufacturing, individual modules are not straight but bend either within their own plane ("bananaing") or bend out-of-plane. The software described in this note measures these two types of bending. The module manufacturing process seems to preclude any detectable shear distortion of modules. We make no attempt to measure it.

Measurement of in-plane and out-of-plane bending was made using a ceiling mounted, continuous wave "phase-based" 3-dimensional laser scanner (Leica HDS6100) that emits amplitude modulated ~ 670 nm laser light in fixed spherical coordinate steps $\Delta\phi = \Delta\theta = \pi/20000 = 0.009$ degrees. Since the scanner is approximately 17 meters above the pivoter table, there is a corresponding step size of 3 mm on the surface of a module being measured. The laser light strikes the modules lying horizontally on the block pivoter table, reflects back toward the scanner and the phase difference between the transmitted and reflected light is a measure of the distance between the scanner and module surface. Since the laser light is emitted uniformly in solid angle while the module surface is at varying distances from the scanner and not completely planar, the resulting three-dimensional "point cloud" of measured points on a module surface is somewhat complicated.

The gist of the cell-center-finding algorithm is to convert the point cloud into a non-uniform mesh of elevation versus position in a reference plane, and search for local maxima that correspond to the flat horizontal portions of a module's top surface. The centers of these flat portions are directly above the corresponding cell centers, the geometrical quantity of interest. Fig. 1 shows the cross-section of a 32-channel module as it lies on the pivoter table and Fig. 2 shows several modules on the table being assembled into a plane. The laser scanner is well above the assembly table, out of view.



FIG. 1: A cross-sectional view of a 32-channel module. The top and bottom surfaces of the extrusions are flat in a region directly above and below a cell center. The eventual neutrino beam direction is in the plane of the paper and parallel to the longer pair of cell walls.

Four metrology targets were placed in fixed positions at the corners of every measured plane. Laser measurement of the target positions defines the position of a plane in the "block" coordinate system. The origin of the right-handed block coordinate system is positioned near the center of the pivoter table. The Z axis direction is perpendicular to the planes and points downwards with the X-axis pointing to the west. See B. Chowdhury's document "Coordinate Systems in Making of Far Detector" (doc #9795), page 9, for an annotated photograph of the block coordinate system. For the next data analysis step, every other plane was rotated by 90 degrees in such way that the Y axis is always along the cell direction and X axis is transverse to the cell direction.

The measurement of plane position in the Z direction was made within a square much larger than the dimensions of the production table. Due to non-ideal reflection from the non-glossy PVC surface of a plane, the raw laser scan

data does not follow exactly the form of a module's surface and has a highly irregular background with many spikes parallel to the Z-axis.

The NOvA Surface Analysis [1] program (NSA) was used for removal of background and those points located outside the plane of modules (e.g., those points that correspond to positions on the floor of the assembly hall). Two output text files from NSA were used for the next analysis step. The first file, LayerLL-reduced.txt, contains the (x,y,z) points in the block coordinate system. The second one, LayerLL-range.txt, lists the range of the full data set for the X and Y axes.



FIG. 2: View looking down on several modules placed on the block pivoter table. The laser scanner is above the table, out of view.

Using the information from these two files the measured Z dependence on X position for all Y positions was built. The distribution in a 1-dimensional histogram looks like the set of peaks with peak maxima near the center of cells. Therefore the position of all 384 cells of a plane can be obtained from the position of the peaks.

II. DATA ANALYSIS.

Data analysis proceeds in three steps: peak finding and sorting of peaks; correction of data for extra or missing peaks; and calculation of shape parameters to be loaded into the alignment database.

A. Peak finding and sorting

The ROOT class TSpectrum was used to find the peaks that correspond to the cell center positions. TSpectrum was developed originally for identification of peaks in multidimensional coincidence gamma-ray spectra. The one-dimensional peak search function Search() has several parameters. The sigma of searched peaks, amplitude threshold, and background parameters of this function were all tuned using laser scan data. Measured X, Y and Z positions were temporarily shifted to positive values since Search() cannot work with a negative ones. When the main analysis was finished, all points were moved back to the block coordinate system.

The simple bubble method was used for the sorting of peak position. The nominal position of a peak is not always fully correct. All peaks were fitted with a Gaussian function using the peak position from TSpectrum as an input parameter. Fig. 3 shows an example of a distribution that was built with the NSA program.

The red triangles are the peak positions determined by TSpectrum. Blue lines are from a Gaussian fit.

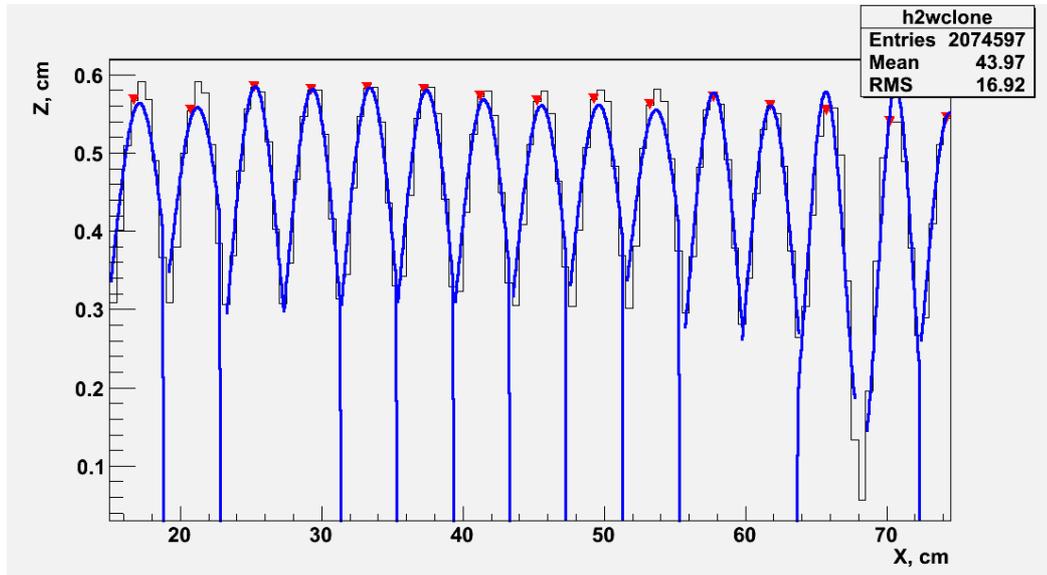


FIG. 3: The distribution of elevation (Z) versus position perpendicular to the cell direction (X) for all positions along the cell length (Y). Only several peaks in a narrow window are shown.

Data points inside modules grooves were removed in subsequent analysis steps. This data has flat peaks of width less than 2 cm in the middle of a cell. Fig. 4 shows this new distribution.

Flat peaks have a significant variation in width and distance between them that can be a problem for the correct calculation of cell position. Since the form of a peak is similar to a rectangle, fitting a Gaussian to such a peak can be problematic or impossible in many cases, too.

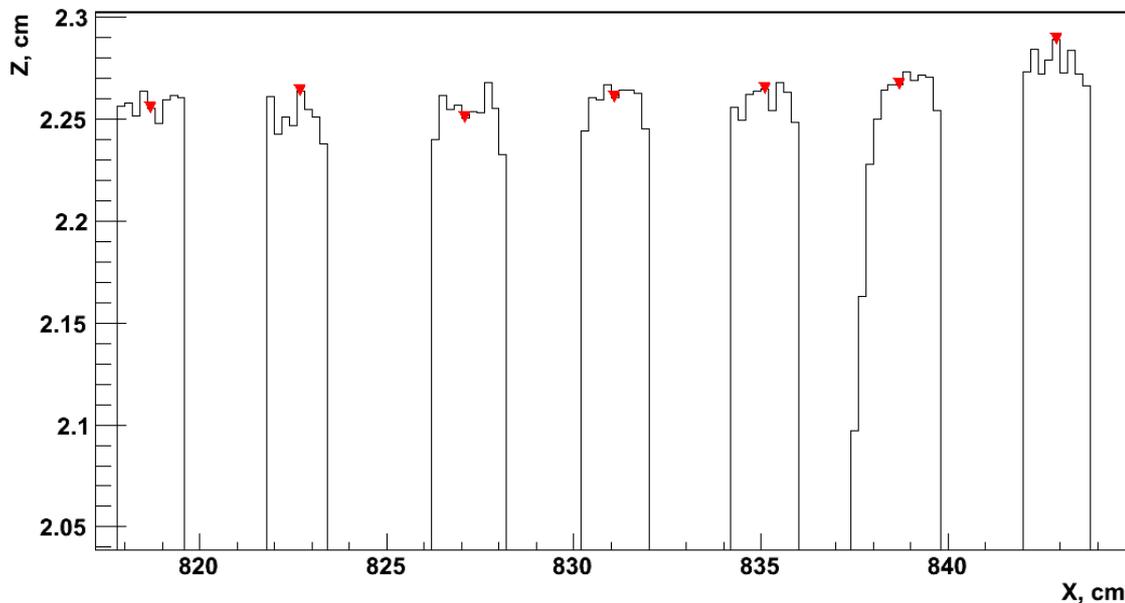


FIG. 4: The dependency Z on X with a last version of the NOvA Surface Analysis program that was used for all next analysis.

Four methods were compared to optimize the accuracy of peak finding:

1. The positions of peaks were found by TSpectrum only;
2. TSpectrum peaks fit with a Gaussian function;
3. Gaussian fit and then adding TSpectrum peaks when the fit fails;
4. Using the mean position of the flat peak.

The last method was used in the next analysis step due to its simplicity and good (~ 1.2 mm) accuracy.

B. Correcting data for extra or missing peaks

Several algorithms were developed for correcting imperfect data to eliminate extra peaks or to insert missing peaks. Fig. 5 shows an example of a spurious peak found by TSpectrum.

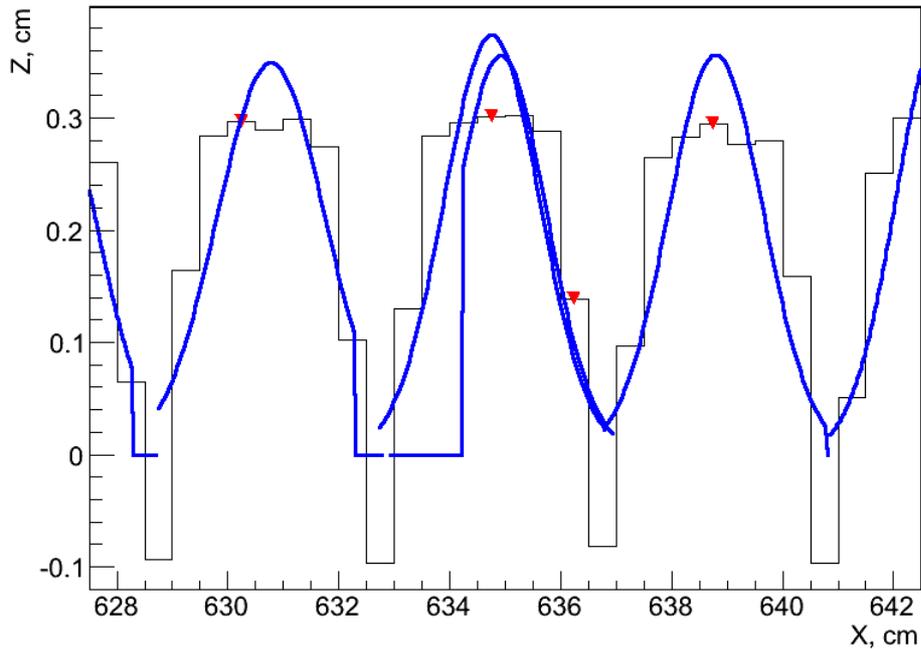


FIG. 5: The distribution of elevation (Z) versus distance perpendicular to cell direction (X). Peaks found by TSpectrum are marked by an inverted red triangle. An incorrect peak is located at 636 cm.

The algorithm for removing or adding peaks is based on the calculated distance between neighboring cells. If the distance is less than half the average cell spacing d_c , as determined by laser scan measurements, the putative peak is removed from the peak array. If the distance is more than $1.8d_c$ a peak is added to the array at distance d_c distance from the correct neighboring peak to the left.

Sometimes the cleaning procedure removes all points at the plane edges and the first or last peaks do not appear in the data. The algorithms for the adding in the first and last lost peaks are based on the calculated average position of these peaks in all cells. Some peaks were broken by the cleaning program into several parts and TSpectrum could then find several peaks instead of one. Each peak found was tested for empty histogram bins to its left and right. Gaps up to 4 empty bins are attached to a peak and narrow peaks between gaps are combined to form one peak. This algorithm ultimately groups all narrow peaks into one peak with the correct width.

The smoothness of plotted Z on Y non-flatness and X on Y in plane non-linearity dependencies along the cell is a final criterion for the data improving. Since both distributions are sufficiently smooth a big jump between nearby measurement points means that the data is not fully correct. (The cell lengths are smooth with no sharp discontinuities.) The fitting of a distribution with a single peaking point can fail. Although the number of similar dependencies is small the alignment parameters for individual cells for this can be lost. If the difference between two

adjacent points is greater than 1 cm, the anomalous point was corrected and its value was replaced with the average of two neighboring points at left and right. When the first or last point jumped, it was replaced with a nearby point. Additional iterative tests were applied for the case when the number of jumped points is more than one.

C. Calculation of shape parameters.

Laser scan data for a given plane were arbitrarily divided into 29 regions along each cell to increase statistics for peak finding so that each region is 51 cm long. The distributions of Z versus Y and X versus Y were separately fit to polynomials. A polynomial of degree 5 was used for the Z versus Y distribution as a compromise between a minimal number of database parameters and fitting accuracy. Fig. 6 shows an example of data fit with two separate polynomials. The green line is a polynomial of degree 9 and the red line is a polynomial of degree 5. The polynomial of degree 5 was finally selected as a compromise between the number of alignment parameters and the accuracy of fitting.

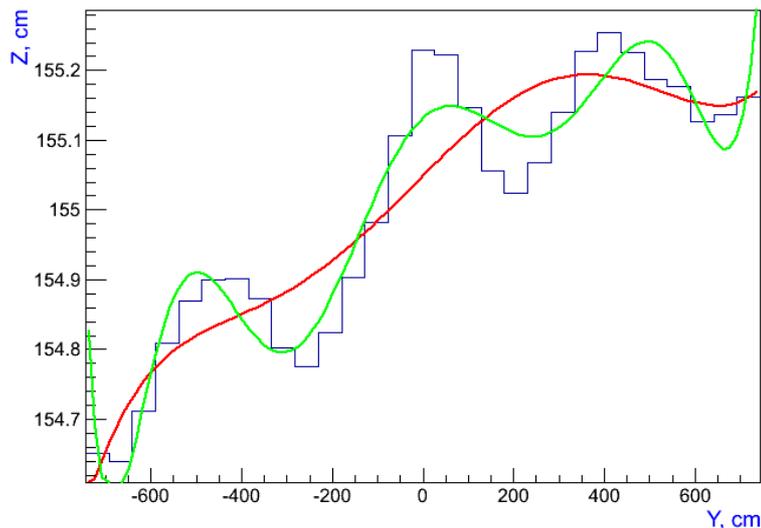


FIG. 6: Z versus Y data fit with a polynomial of degree 5 (red line) and 9 (green line).

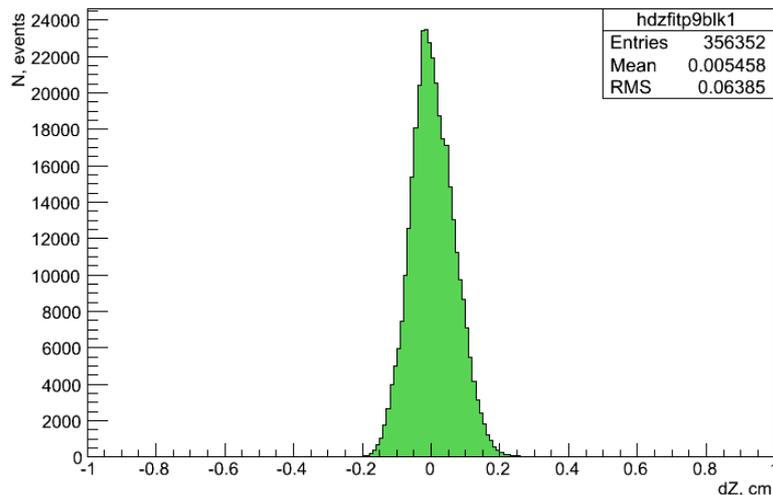


FIG. 7: The difference between Z versus Y data and a 5th degree polynomial fit to the data for block.

The accuracy of fitting in this case for all 12,288 cells of block 03 is 0.6 mm with a maximum spread of ± 2 mm (see Fig. 7).

The distribution of X versus Y, representing in-plane bending, was fit with a polynomial of degree 2. Red line in Fig. 8 is a fitting line. The accuracy of fitting is 1.3 mm (Fig. 9).

The coefficients a_1 and a_2 of the polynomial function $y = a_0 + a_1x + a_2x^2$ for first 15 blocks are plotted in Appendixes A and B.

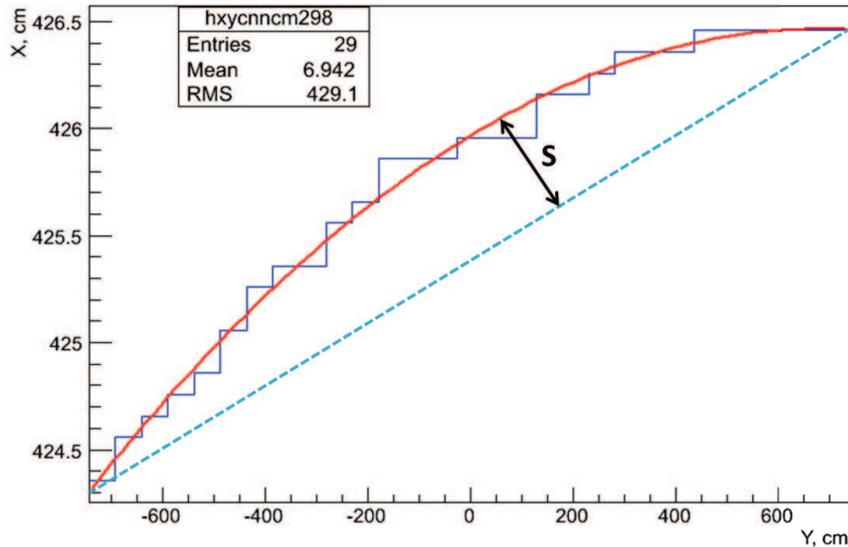


FIG. 8: A fit with a polynomial of degree 2 (red line) to the data distribution of X versus Y (blue histogram). S is the maximum perpendicular distance of a cell center from a straight line (cyan dashed line) joining a cell's two ends.

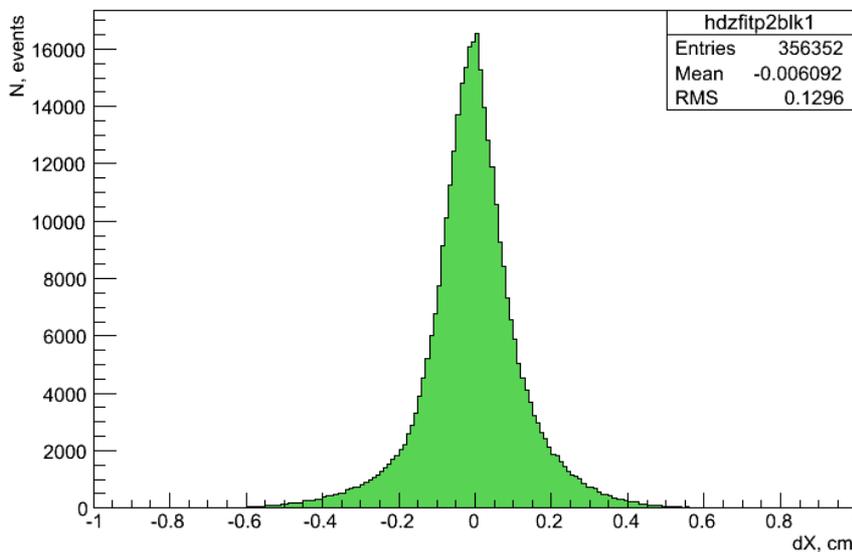


FIG. 9: The difference between X versus Y data and a fit to that data with a polynomial of degree 2. The RMS of the distribution is 1.3 mm.

The sagitta S or maximum perpendicular distance of a cell center from a straight line joining the cell's ends

was calculated for Block 03. The sagitta S of all 12,288 cells in Block 03 are shown in Fig. 10. The mean value $\langle S \rangle = 1.2$ cm with a full spread of 3 cm.

Appendix C show sagitta S for the first 15 blocks.

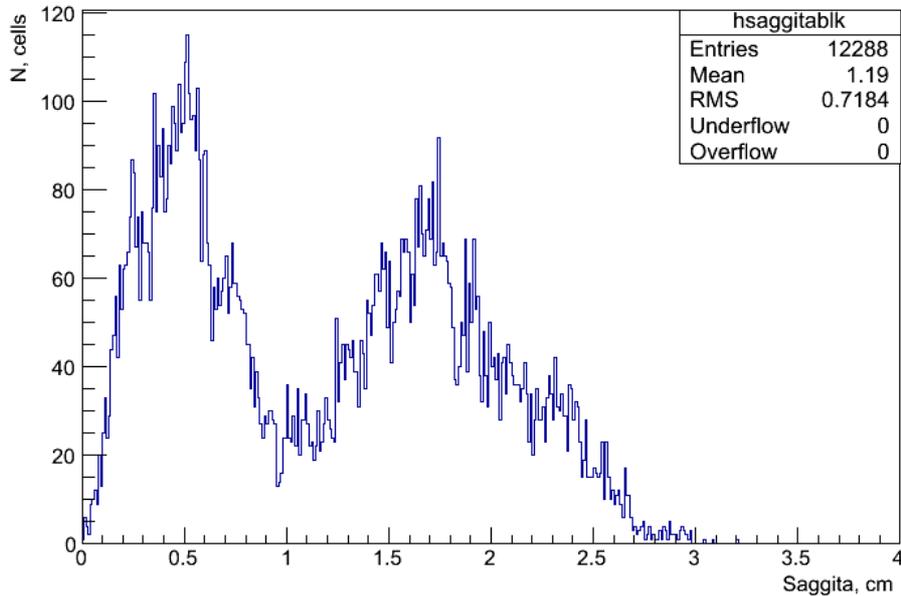


FIG. 10: The sagitta of all 12288 cells for block 03.

Fig. 11 shows the cell spacing of one plane in all 29 regions for block 03. The mean value of Gaussian fit is 3.94 ± 0.01 cm. The small peak at right is a gap between extrusions. The mean value of this gap is $4.56 \pm .003$ cm.

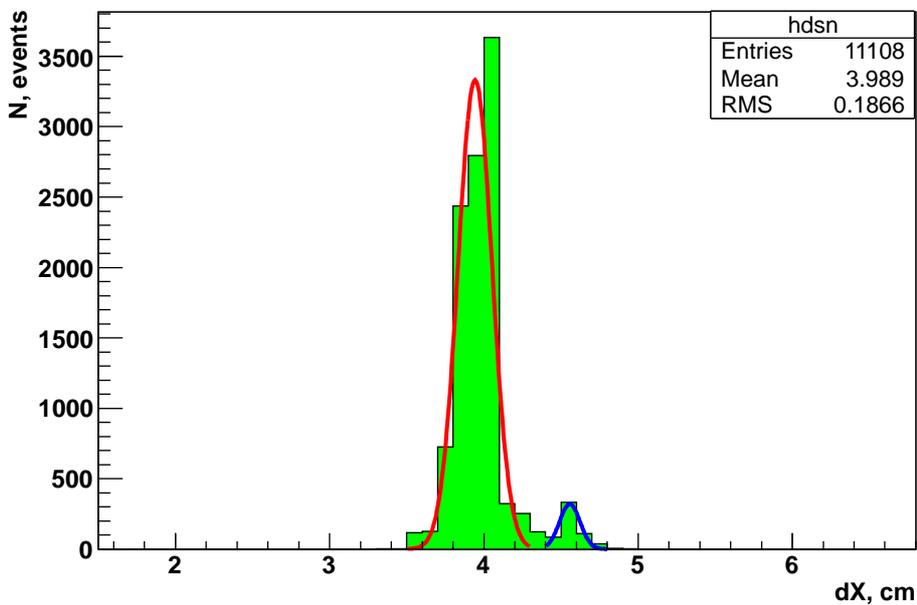


FIG. 11: The cell spacing of one representative plane. The small peak at right is the spacing between neighboring end cells of adjacent extrusions.

Two methods were used for calculating the distance between adjacent planes. Fig. 12 shows the distribution of distances between planes in 29 regions along the cell direction. The mean value of Gaussian fit gives the average plane spacing.

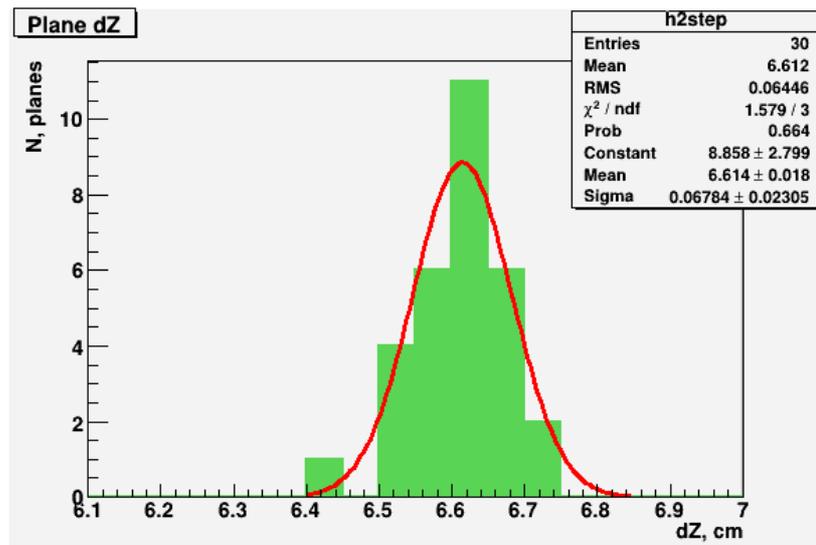


FIG. 12: The distance between planes.

A second way is to fit a straight line to the mean Z position of an entire plane versus plane number N. The distribution and fit are shown in Fig. 13 for block 00. The slope of the fit line gives the plane spacing. Note that the mean value of the Gaussian fit 6.61 ± 0.01 cm equals exactly the slope of the fitted line 6.61 ± 0.01 cm.

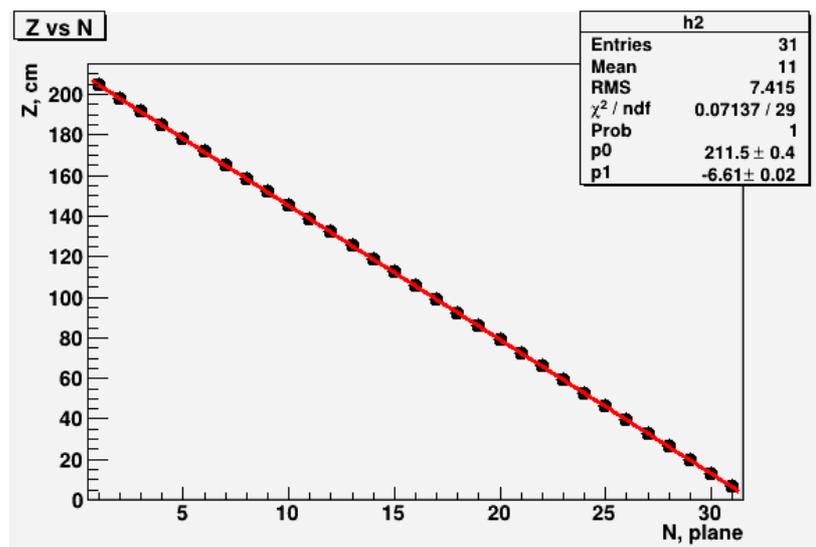


FIG. 13: The mean Z position of a plane versus plane number for block 00. Only 31 of 32 points are plotted since the program was crashed at the peak finding procedure in plane 00. This problem was fixed later and all data are processed now.

III. CELL SHAPE PARAMETERS FOR DATABASE.

Obtaining cell shape parameters for the offline database is the main goal of the laser data scan analysis. The fit parameters for each cell are written to an output file as two lines. The first line contains the plane number in detector coordinate system starting from plane 0, cell number from 0 to 383 and 6 polynomial coefficients from the 5th degree

polynomial fit of the Z vs Y data distribution. The second line has plane number, cell number and 3 polynomial coefficients from the fit of the X vs Y distribution with a quadratic. Switching to the detector coordinate system (x, y, z) familiar to offline workers, the polynomials describe the $x = f(y)$ shape function for odd planes and $y = f(x)$ for even planes. Recall that the first plane in a block is horizontal and measures the y coordinate in the detector coordinate system.

The parameters of fitting were written to the NOvA database table with "laser" name. Each row of table has 11 elements:

```
chn, za0, za1, za2, za3, za4, za5 xa0, xa1, xa2 xr1.
```

Here "chn" stands for channel number and is calculated from block and cell number as $N_{ch} = N_{blk} * 1000 + N_{cell}$, The "za0 - za5" are coefficients of a polynomial function used to fit Z versus Y laser data. The "xa0 - xa2" are coefficients of a quadratic function used to fit in-plane bending for X versus Y laser scan data, and "xr1" is the X position of the cell in the first 50 cm region.

IV. RUNNING CELL CENTER FINDER

The LaserScan package includes two Root scripts. The first script, laserhist.C, builds the Z versus X distributions for 29 regions of a plane. The second script, peaks-fit.C, finds the cell center positions, makes corrections for extra or missing peaks, fits the Z versus Y and X versus Y distributions, and writes the alignment parameters to output files. Both scripts run over a full block of 32 planes. The package is committed to NOvA SVN and can be added to your test area by issuing the command `addpkg_svn LaserScan`

The original outputs of the NOvA Surface Analysis program are contained in a tar file located in `/nova/ana/assembly_ana/BlockNN/LayerLL`. A small script `laseruntarblkNN` located in `/nova/app/users/kravtsov` extracts `LayerLL-reduced.txt` and `LayerLL-range.txt` files from archive and copies them to `/nova/ana/users/kravtsov/Laser/BlockNN`, maintaining the LayerLL directory structure. Both files are stored permanently for possible future reprocessing. For the present analysis the BlockNN directory is copied to `/nova/ana/users/kravtsov/Laser_ana`. Another small script `mvscript00` runs in BlockNN, where script is located, moves the files up to the parent directory, and removes the directory structure. Here NN stands for block number (0-27) and LL for plane number (0-31).

The first script runs in ROOT by commands:

```
.L laserhist.C
laserhist()
```

These two output files from the NSA program are used as input for laserhist.C, which in turn creates two output files. The first output file `blockNNplaneLL.root` contains the Z versus Y histograms of 29 regions for all planes. The second output file, `LayerLL-z29.txt`, lists a mean Z position for each plane in a block.

The second script, `peaks-fit.C`, uses the two output files from laserhist.C and is run in ROOT by the commands:

```
.L peaks-fit.C
fitpeaks()
```

Four output files for each plane will then be created:

1. `blockNNplaneLL_out.root` with plane output histograms;
2. `fullblockNN_hists.root` with block output histograms;
3. `blockNN_planeLL_fit-params.dat` with alignment parameters;
4. `blockNN_planeLL_384cells-region1.dat` with cell positions at first region.

The location of script, input and output files is `/nova/ana/users/kravtsov/Laser_ana/BlockNN` where db directory contains last two files. The db directory should be created before running the script.

[1] B. Mercurio, Surface Analysis, NOVA Document 8990 (2013).

V. APPENDIX A

Distribution of the coefficient a_1 in the quadratic function $X = a_0 + a_1Y + a_2Y^2$ used to fit in-plane bending data X v Y . Y is measured along a cell in the direction of the fiber.

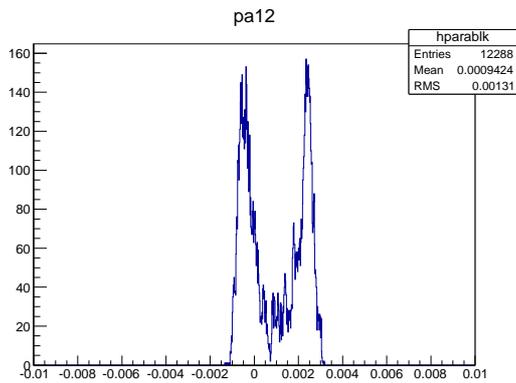


FIG. 14: Block 00

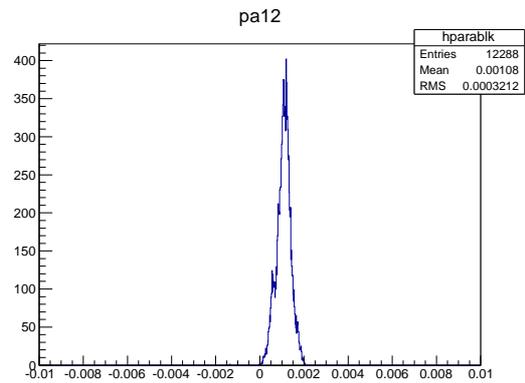


FIG. 15: Block 01

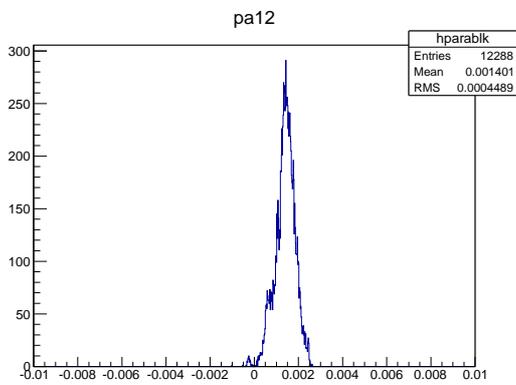


FIG. 16: Block 02

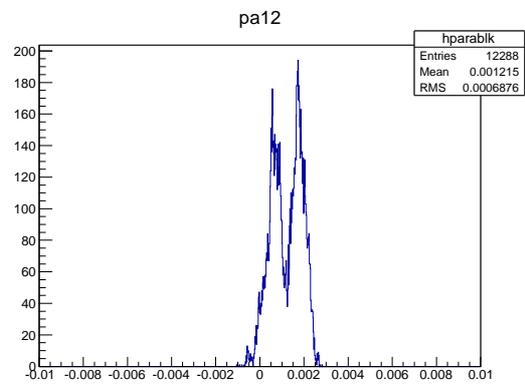


FIG. 17: Block 03

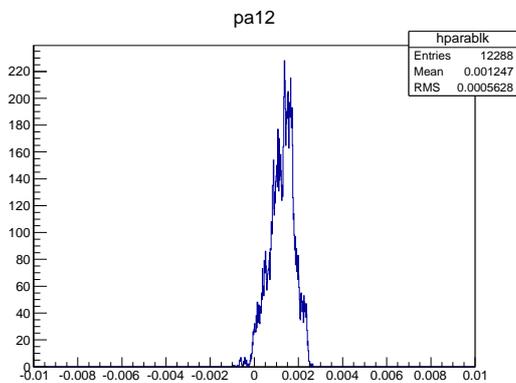


FIG. 18: Block 04

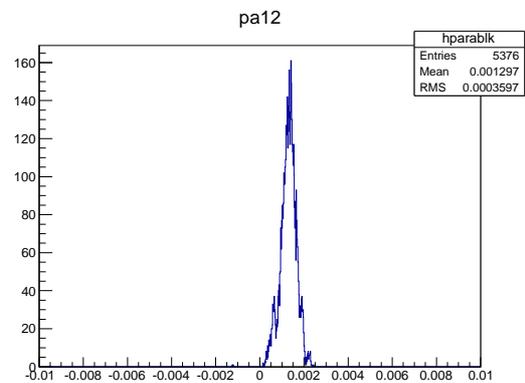


FIG. 19: Block 05

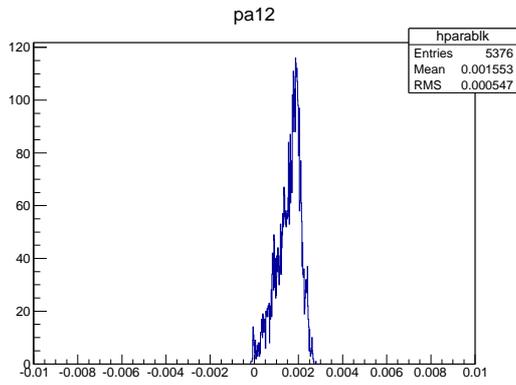


FIG. 20: Block 06

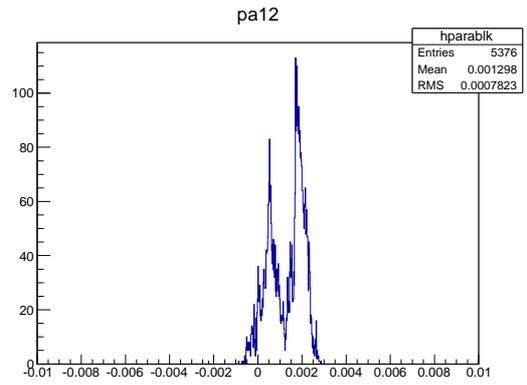


FIG. 21: Block 07

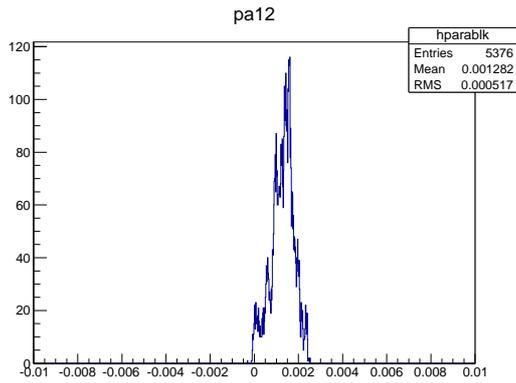


FIG. 22: Block 08

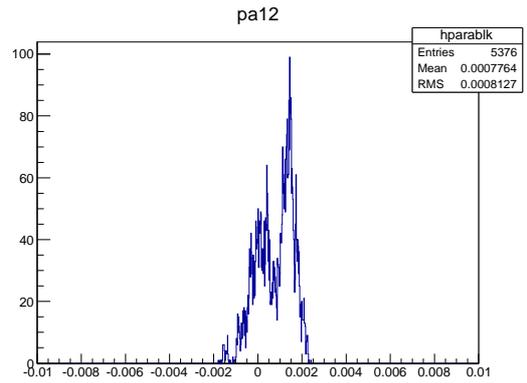


FIG. 23: Block 09

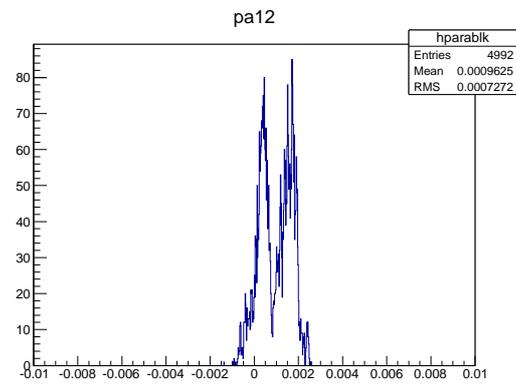


FIG. 24: Block 10

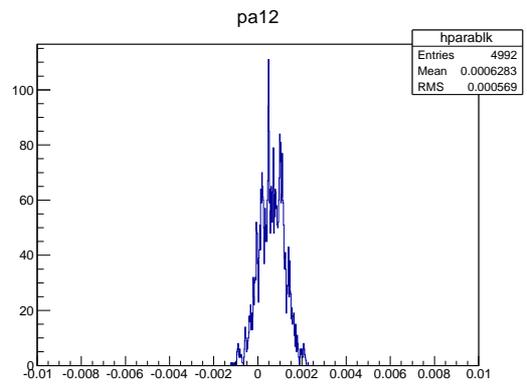


FIG. 25: Block 11

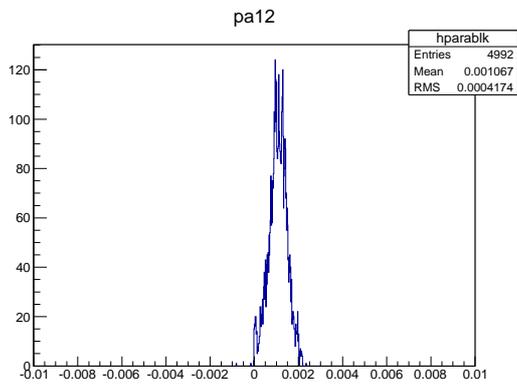


FIG. 26: Block 12

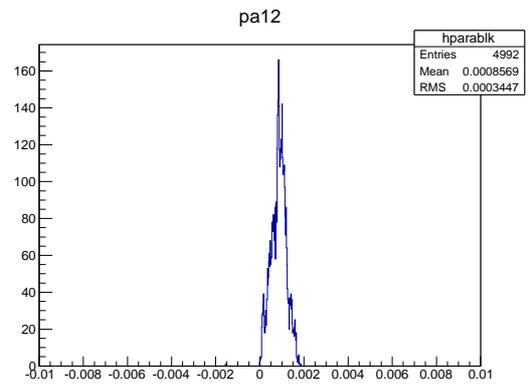


FIG. 27: Block 13

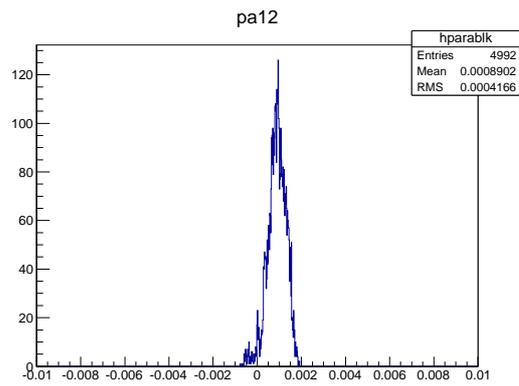


FIG. 28: Block 14

VI. APPENDIX B

The distribution of the coefficient a_2 in the quadratic function $X = a_0 + a_1Y + a_2Y^2$ used to fit in-plane bending data X v Y . Y is measured along a cell in the direction of the fiber. The horizontal axis unit is cm^{-1} .

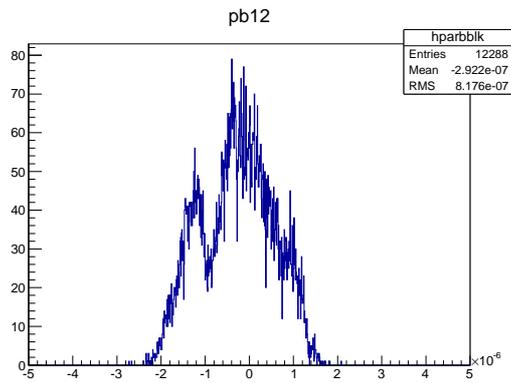


FIG. 29: Block 00

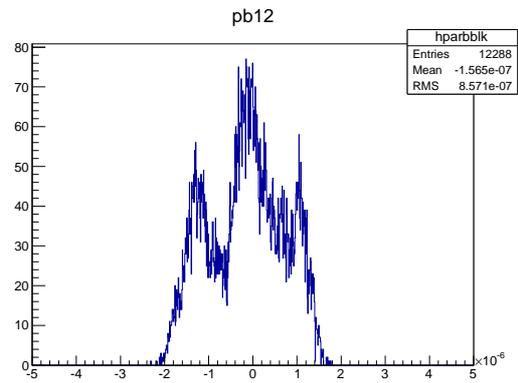


FIG. 30: Block 01

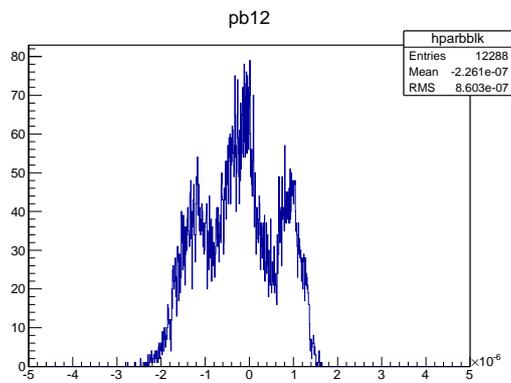


FIG. 31: Block 02

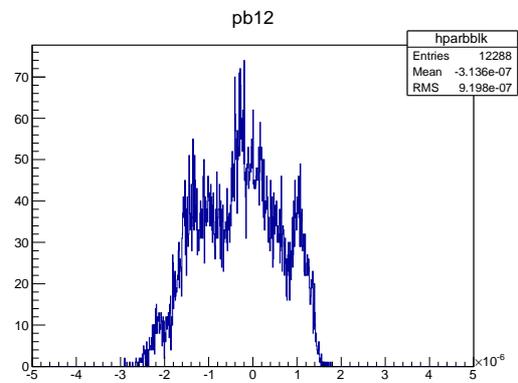


FIG. 32: Block 03

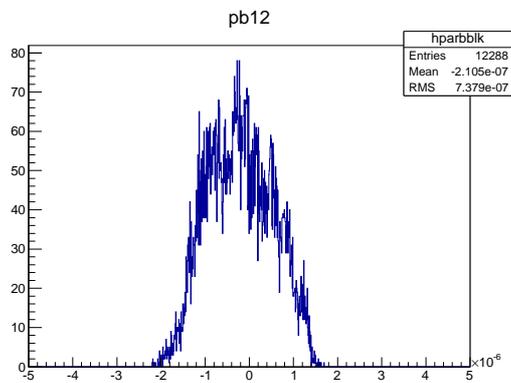


FIG. 33: Block 04

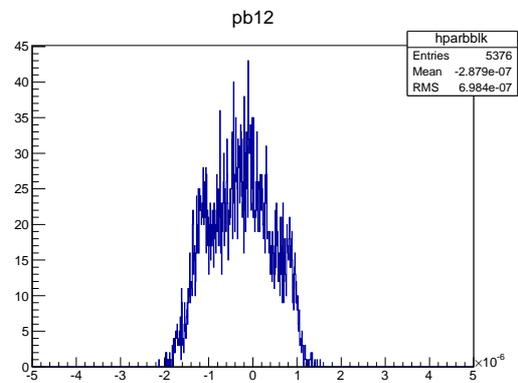


FIG. 34: Block 05

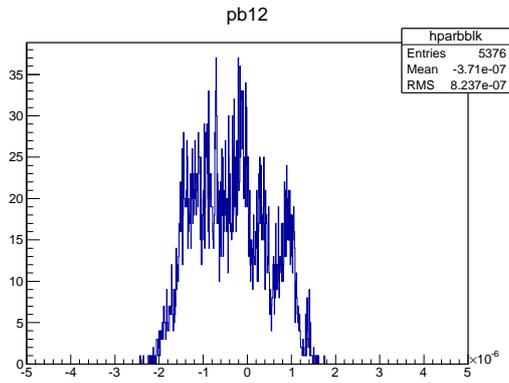


FIG. 35: Block 06

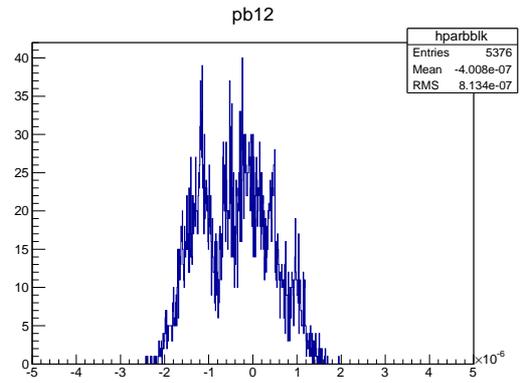


FIG. 36: Block 07

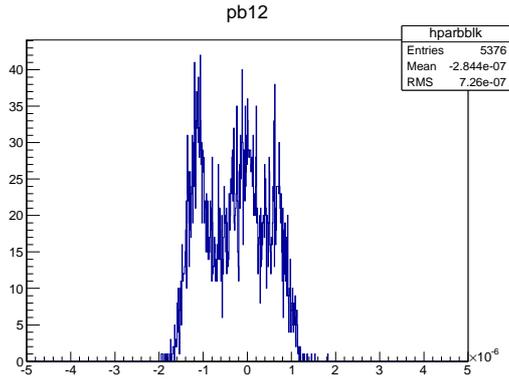


FIG. 37: Block 08

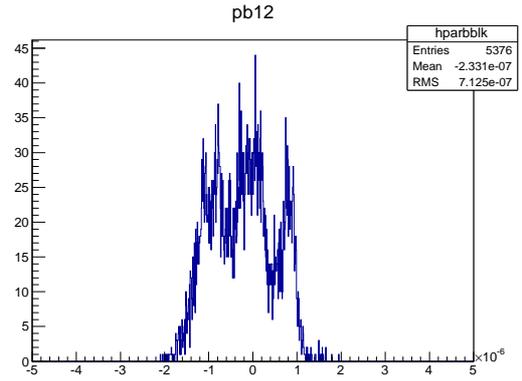


FIG. 38: Block 09

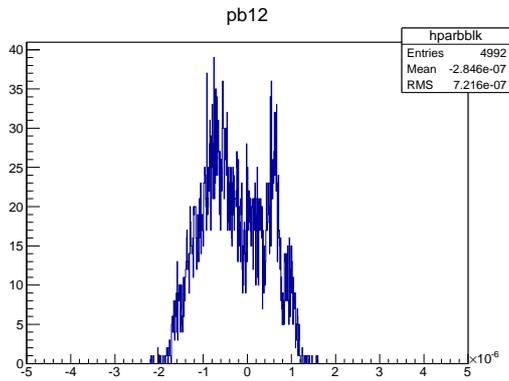


FIG. 39: Block 10

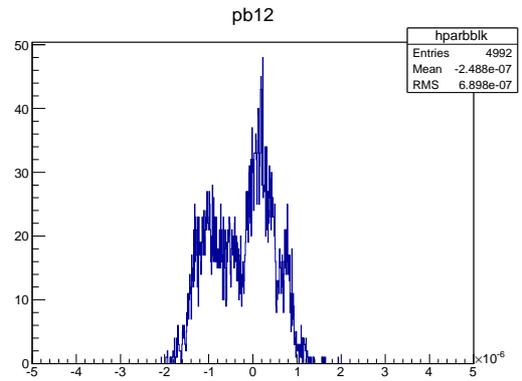


FIG. 40: Block 11

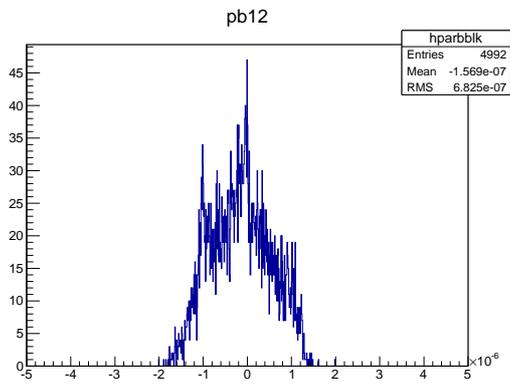


FIG. 41: Block 12

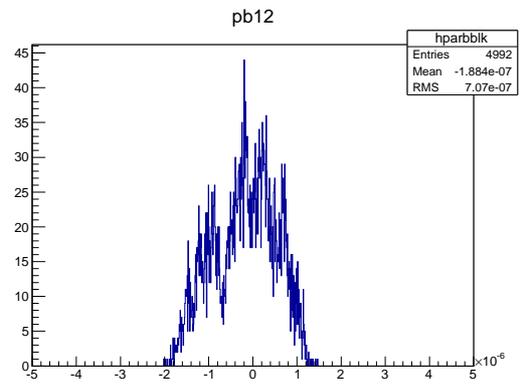


FIG. 42: Block 13

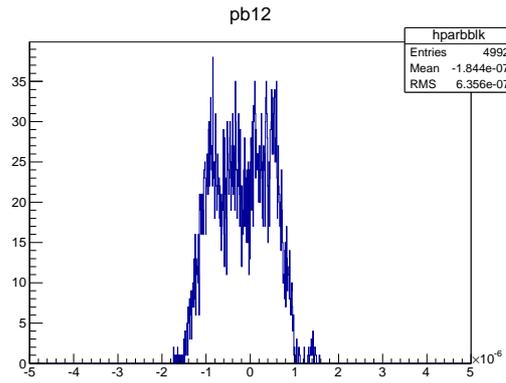


FIG. 43: Block 14

VII. APPENDIX C

The distribution of sagitta S for all cells in complete blocks. The horizontal axis unit is cm.

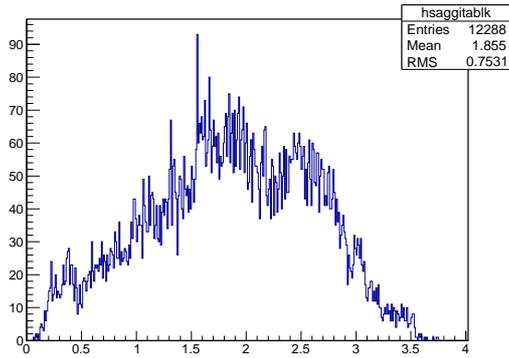


FIG. 44: Block 00

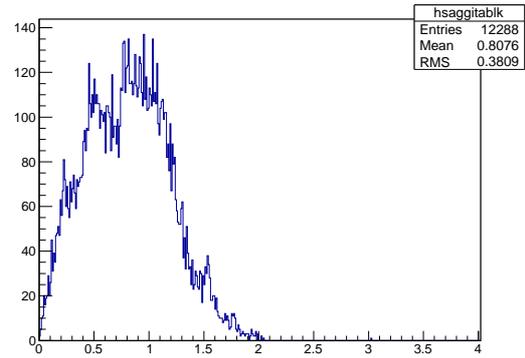


FIG. 45: Block 01

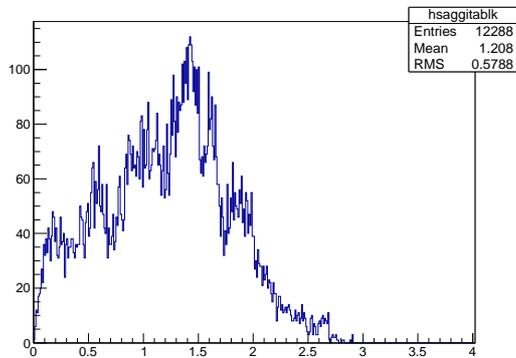


FIG. 46: Block 02

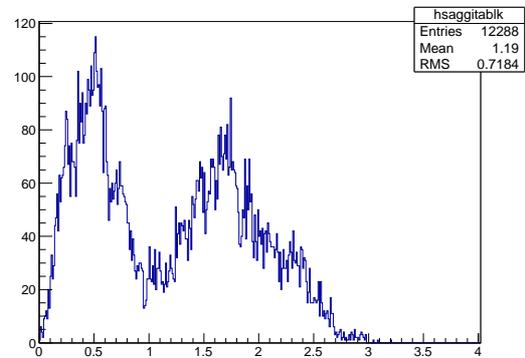


FIG. 47: Block 03

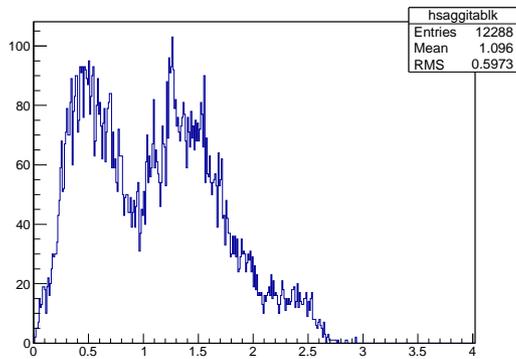


FIG. 48: Block 04

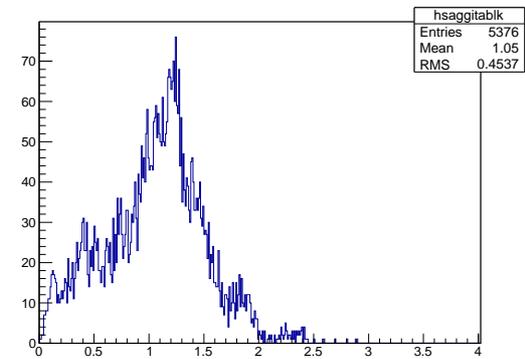


FIG. 49: Block 05

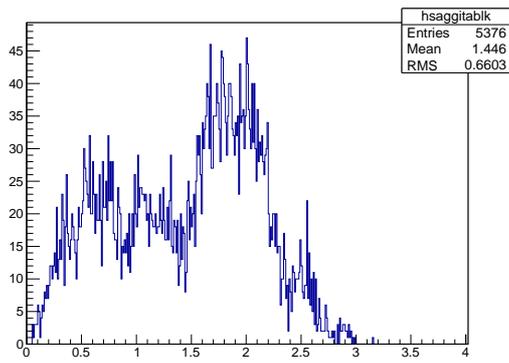


FIG. 50: Block 06

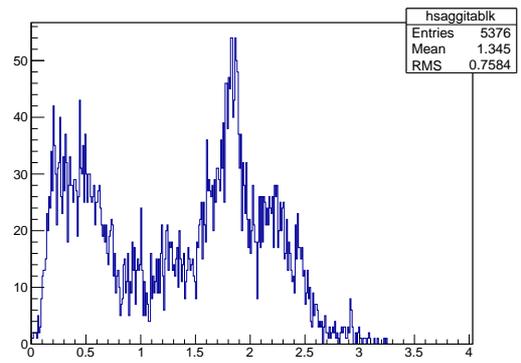


FIG. 51: Block 07

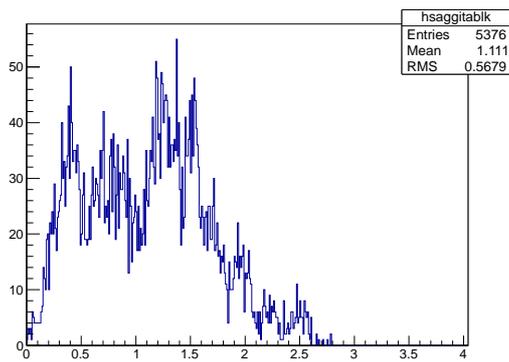


FIG. 52: Block 08

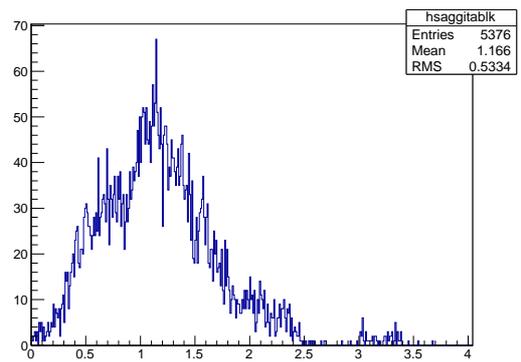


FIG. 53: Block 09

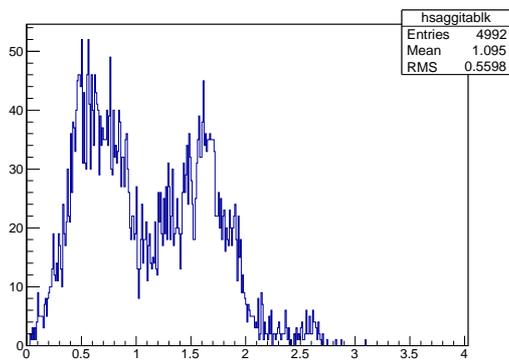


FIG. 54: Block 10

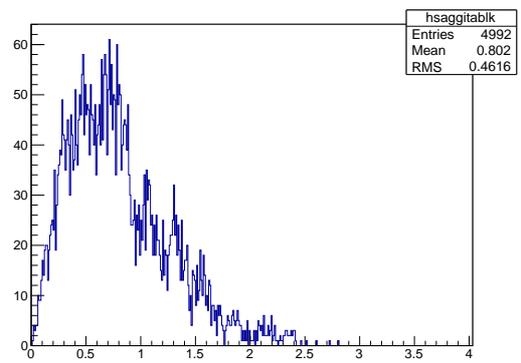


FIG. 55: Block 11

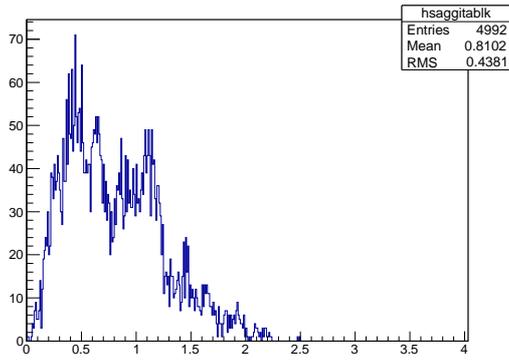


FIG. 56: Block 12

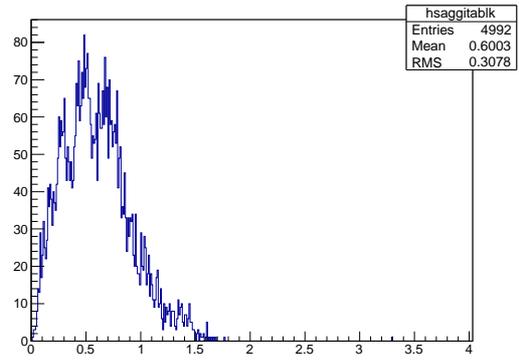


FIG. 57: Block 13

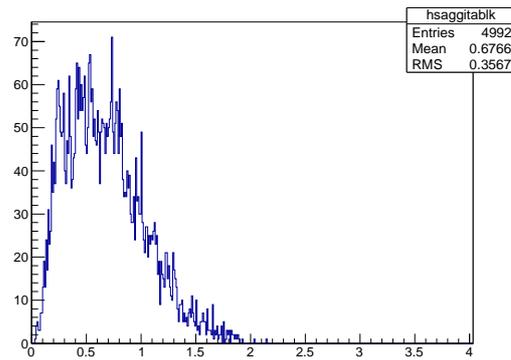


FIG. 58: Block 14