University of Groningen

Bachelor Thesis

An accurate prediction of LHCb's Vertex Locator pixel noise

***A**uthor:*

M.E. Vos

***E**xaminers:*

Dr. K.A.M. De Bruyn

Dr. Ir. C.J.G. Onderwater

June, 2021

Contents

1 Introduction **2**

2 Theory **4**

2.1 Noise distribution and noise position 4

2.2 The data sets 5

2.3 A fitted function representing the relation between noise position and trim level 6

2.3.1 Quantifying the quality of fit 6

2.4 The uncertainty of the noise position 7

3 The average noise position **8**

3.1 Residual of the linear fit 8

3.2 The improved fit 10

4 Looking at individual pixels **12**

4.1 Residual of the linear fit 12

4.2 A search for the best fit configuration 13

4.2.1 Optimize the fit degree and trim level amount to use 14

4.2.2 Optimize which trim levels to use 14

5 Making predictions **16**

6 Conclusion and outlook **18**

Appendices **19**

A Plots of individual pixels **19**

1 Introduction

You, as a human being, are a curious creature. When you see a stone, you ask yourself what it is made of. If you ask yourself that question often enough, you end up at particle physics. The study of understanding the building blocks of nature.

One way of finding out what things are made of is trying to smash them to pieces. The Large Hadron Collider (LHC) at the European Organization for Nuclear Research (CERN) is doing just that. To do this it uses a 27-kilometer ring, consisting of superconducting magnets. It accelerates two protons, in opposite direction, around the ring for multiple rounds before they are steered towards each other and collide. At several locations around the ring, researchers try to detect particles in the debris of the collision.

One of the experiments at CERN is called LHC-beauty (LHCb). The LHCb experiment is trying to understand the properties of particles that makes up the proton, named quarks. Specifically, it studies the beauty quark, hence the name. To detect this particle a setup is used as shown in Figure 1.

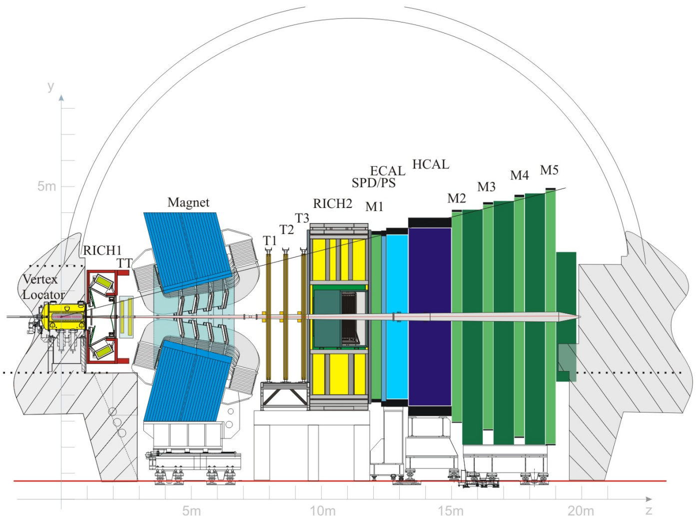

Figure 1: Side view of the LHCb setup [2].

The most prominent detector of the LHCb experiment is called the Vertex Locator (VELO), which surrounds the proton collision region. The VELO is about to be replaced by a new and faster version of itself. The new VELO uses multiple grids of pixels situated in different planes, which track the trajectory of particles that pass through the detector. Figure 2 shows a schematic representation of this. When a particle passes through a pixel, a current starts to flow which is then detected by the VELO.

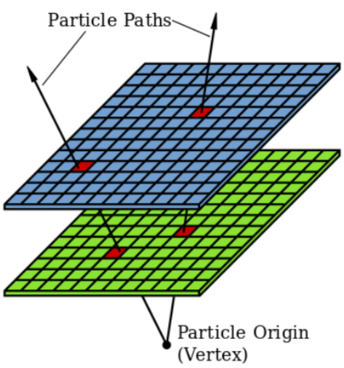

Figure 2: A schematic representation of how the VELO can recontruct the trajectories of particles that pass through the detector [4].

When there is no particle passing through the pixel, a small current can still flow. This current can have multiple origins, for example thermal fluctuations around the pixel. This small current is called noise. A simulation of this small current is shown in Figure 3. In this Figure a particle passing through would appear as a high spike. It is critical to have a clear differentiation between a current coming from noise and a current coming from a passing particle. For this differentiation a critical value can be set. If a measurement is above this value it is counted as a particle and if it stays below this value it's ignored. This critical value is called a threshold.

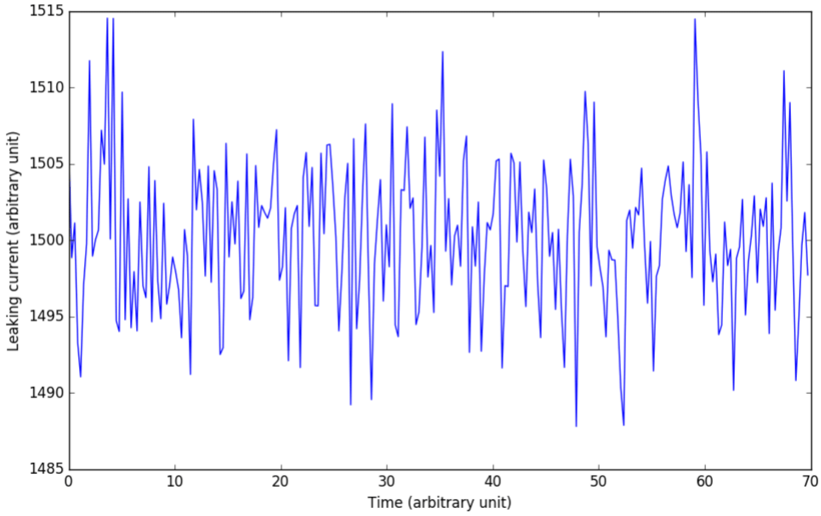

Figure 3: A simulation on the noise of a pixel.

Only a single threshold can be set on an entire grid, which causes a problem. The noise on each pixel is slightly different because the pixels are not identical to each other. The single threshold would therefore be too high for one pixel but too low for another.

The solution for this is called trimming. Trimming can compensate for the differences in noise between the pixels, by adding a starting value to the measured current. There are sixteen different trim levels that each pixel can be set to. The goal is to trim the pixels in such a manner that the noise are at equal position. This process is called noise equalisation.

Before the noise equalisation can be done, one needs to know where the noise is located on each trim level. In an ideal world one would simply measure the noise for each trim level. But this takes a lot of time. Due to the fact that the equalisation process will happen on a daily basis there is not enough time for measuring each trim level.

Instead of measuring the noise on each trim level an approximation can be done. Currently only the noise of the two outermost trim levels are measured and the rest of the trim levels are linearly interpolated between them.

In this thesis, the accuracy by which the interpolation predicts the noise of the unmeasured trim levels is investigated. Furthermore, an improved prediction method is explored. The goal is to predict the noise position on all trim levels based on the measured noise position of a subset of trim levels. The prediction must be as good as possible and the amount of measured trim levels must be as small as possible.

2 Theory

The VELO, shown in Figure 4, has 26 planes. Each plane is composed of 24 grids, which are called VeloPix ASICs (application specified integrated circuit). Each red square in Figure 4 has 3 ASICs. An ASIC has 256 x 256 (65536) pixels, each with a surface area of $55 \times 55 \mu m^2$[3]. Per ASIC a single threshold can be set to ignore noise.

Before the threshold is set, the pixels are equalised. The equalisation is done by giving each pixel a trim level. There are in total sixteen trim levels which are denoted by a hexadecimal value. A hexadecimal value uses letters to denote the numbers ten to sixteen. So the trim levels are denoted by 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F.

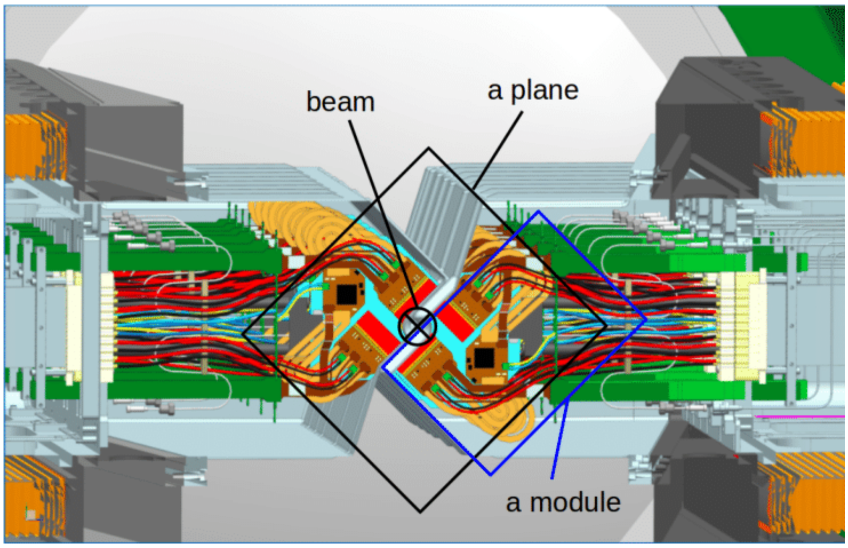

Figure 4: A schematic representation of the VELO [4].

2.1 Noise distribution and noise position

The noise on the VELO is a randomly fluctuating current, which was simulated in Figure 3. The current shown in figure 3 can never be measured exactly. The only thing that can be measured is how often the current crosses a certain level. Doing this for a range of current levels gives rise to a noise distribution, which is shown in Figure 5.

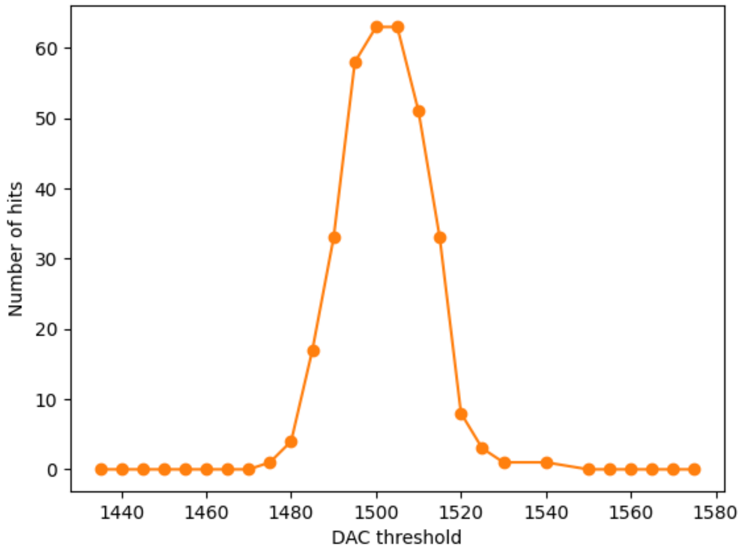

Figure 5: An example of a noise distribution.

The current is converted inside the ASIC to a digital signal and can therefore not be measured explicitly. Therefore the current is stated in an arbitrary unit called DAC. The ASICs pixel can count at most 63 hits per measurement, which is why the noise distribution in Figure 5 is capped. The distribution shows that the noise is located around 1500 DAC, which can also be seen in the simulation.

The goal of the equalisation is to align the mean of the noise distribution of all pixels. Since this thesis focuses on the equalisation process, the mean of this distribution is the only thing of interest. Other characteristics of the distribution, like the standard deviation, are not regarded. The mean of the noise distribution is from now on called the *noise position*.

2.2 The data sets

For this thesis the noise position on each pixel is measured for all sixteen trim levels. This can show how close the predictions are to the measured noise position.

The measurements are done on five different ASICs. This allows for a comparison between ASICs and for finding similar characteristics over different ASICs.

Figure 6 shows the noise positions for each trim level on one of the data sets. It shows that in general the trim level shifts the noise position by some amount to the right.

The goal of the equalisation process is to find the trim levels that collapse the distributions shown in figure 6 into a single line, in an ideal situation.

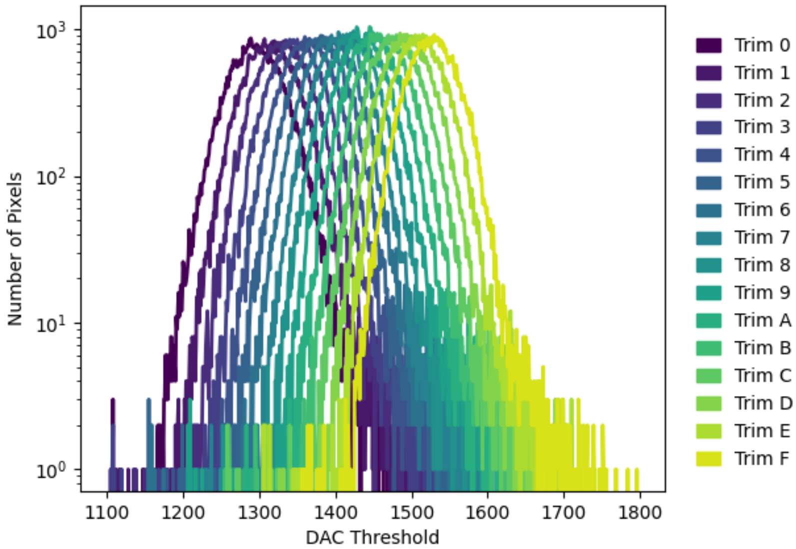

Figure 6: A distribution of noise positions per trim levels for a single data set.

2.3 A fitted function representing the relation between noise position and trim level

Measuring the noise position for each trim level takes too much time. Therefore, an approximation must be done. This approximation can be done by fitting a function through some measured noise positions which would then give the noise position of all the trim levels.

To fit a function two decisions must be made.

1. The function type that is fitted through the measured noise positions.
2. Which noise positions are used for the fit.

These two decisions together are called the *fit configuration*.

Currently the noise positions of all trim levels are linearly interpolated between the two outermost trim levels. The fit configuration of this approximation is a linear function fitted through the trim level 0 and F.

In the following sections the best fit configuration is explored. Multiple function types will be fitted through different sets of trim levels. To compare the fit configurations the quality of the fit must be quantified.

2.3.1 Quantifying the quality of fit

To quantify how good the fitted function is, the chi-squared test is used. In this test the noise position given by the fitted function is compared to the measured noise position.

For ν different trim levels with a measured noise position of x_i, with an uncertainty of σ_i, and an expected noise position of μ_i the chi-square is defined as [1],

$$\chi^2 = \sum_{i=0}^{\nu} \frac{(x_i - \mu_i)^2}{\sigma_i^2} \tag{1}$$

where x_i comes from the data set and is either the noise position, when looking at individual pixels, or the average of 256x256 pixels when looking at the whole ASIC. The uncertainty σ_i will be discussed in the next section. The μ_i is the noise position given by the fitted function.

The degrees of freedom for the chi-squared test is the total amount of trim levels (sixteen) minus the amount used for fitting the function.

From this chi-squared test a p-value is calculated. The p-value is more intuitive then the result from the chi-squared test. The p-value stays between 0 and 1, closer to 1 means a better fit.

2.4 The uncertainty of the noise position

The noise distribution is measured at an interval of 5 DAC. Within each step the data is treated as a uniform distribution. The uncertainty of the noise position is estimated to be the standard deviation of this uniform distribution [5]. This standard deviation is given by,

$$\sigma = \frac{a - b}{\sqrt{12}} \tag{2}$$

Here $a - b$ is the width of the interval, which in this case is 5. This gives the following uncertainty for the noise position,

$$\sigma = \frac{5}{\sqrt{12}} = 1.44$$

Figure 7 shows the noise distribution of four pixels together with the uncertainty region. The figure shows that the uncertainty in the noise position is reasonable.

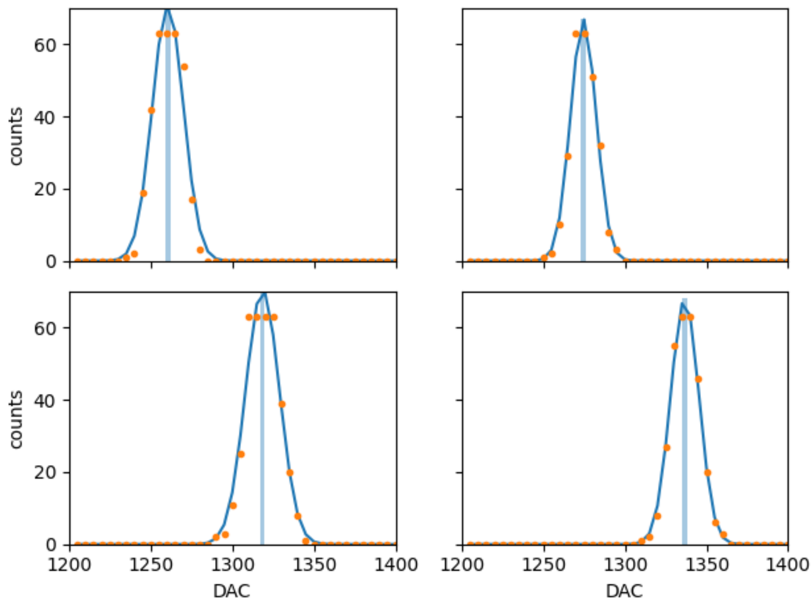

Figure 7: The noise distribution of four pixels with the uncertainty region.

3 The average noise position

To keep things simple, this section starts of by looking at the average noise position. For each trim level the average noise position of an entire ASIC is calculated. This average is the mean of the distributions shown in Figure 6. The average noise position per trim level can give a hint on the general relation between trim level and noise position. In the next section this analysis is extend to individual pixels.

Because this average is just to give an indication for individual pixels the uncertainty is taken to be the same as for individual pixels.

As mentioned earlier, currently the noise position of the two outermost trim levels are measured and the other noise positions are linearly interpolated between them. The fit configuration for this situation together with the average noise position is shown in Figure 8.

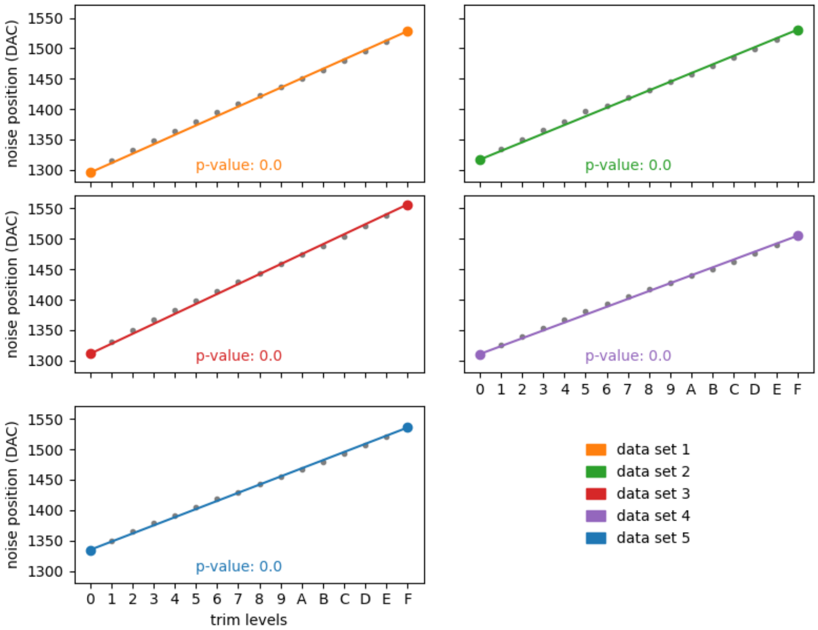

Figure 8: A straight line fit between trim 0 and F for all data sets.

The average noise positions do not follow the straight line, therefore the prediction is not accurate. First the dots are slightly above the line and later they drop below the line. The p-value also shows that the data points are not well represented by the the straight line.

3.1 Residual of the linear fit

To better visualise the deviation of the average noise position from the straight line in Figure 8. One can look at the difference between the fitted function and the measured trim levels. This is called the residual of the fit and is shown in Figure 9.

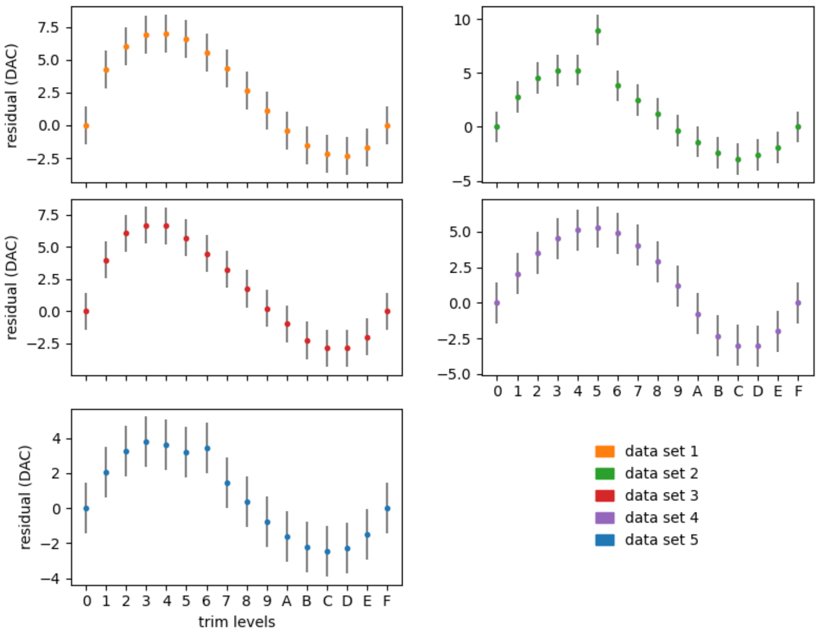

Figure 9: The residual of the straight line fit between 0 and F for all data sets.

For all five data sets there appears a remarkable s-shape. The s-shape is not symmetric around 0, but note that it was a choice to do the linear fit between 0 and F. If the linear fit was done between any other two points, the residual would appear different.

Note that there are two outliers; data set 2, trim 5 and data set 5, trim 6. The noise distributions in these two scans are very wide and of inconsistent shapes. It is clear that something went wrong with the measurements of these data sets. From now on these two scans will be ignored in both the plots and the calculated p-values.

In order to take the s-shape into account one can fit a polynomial through it. It is clear from the s-shape that the polynomial must be at least of a third order. A third degree polynomial needs at least four data points to fit through.

Figure 10 shows a third degree polynomial fitted through the trim levels 0,5,A and F. These trim levels are chosen because they are evenly distributed. Only for data set 2 the trim levels 0,4,A and F are used because of the outlier at trim 5.

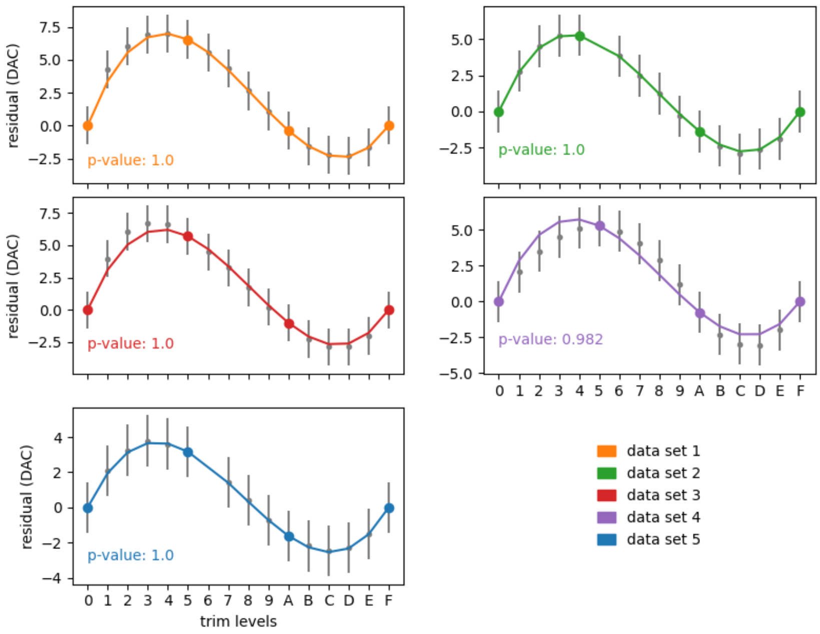

Figure 10: A 3rd degree polynomial fit together with the residual for all data sets.

The fitted function follows the s-shape for each of the four data sets. Four of the five data sets show a p-value of 1. Only for data set 4 the s-shape doesn't seem to represent the relation quit as good as for the rest.

3.2 The improved fit

The residual of the straight line clarified that a third degree polynomial, together with four data points, is needed to accurately fit a function through all the trim levels. In Figure 11 this is applied to the original noise positions.

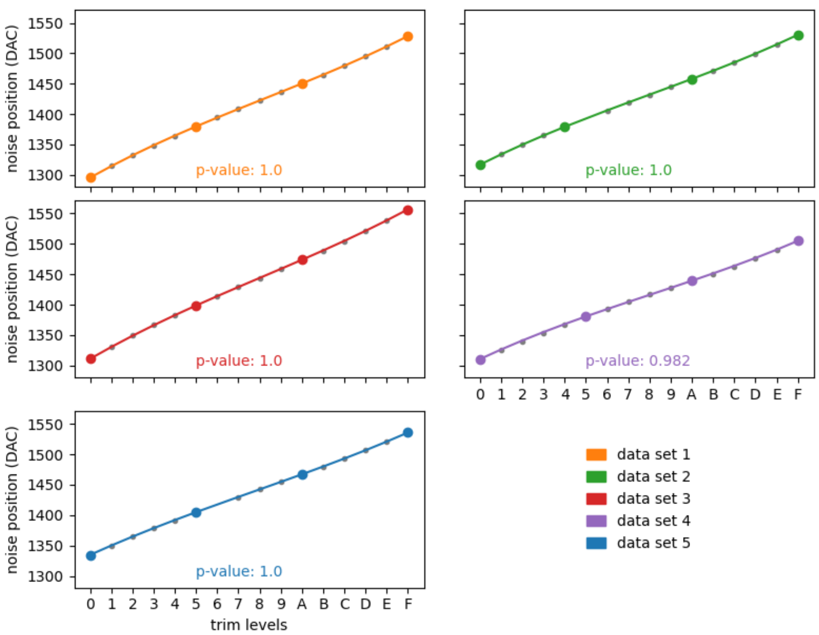

Figure 11: A 3rd degree polynomial fit together with the noise position for all data sets.

The third degree polynomial follows the data points much better than the straigth line. There is no clear deviation from the fitted function. The p-value has risen to a perfect 1.0 for four of the five data sets, which shows that the fit is indeed much better.

4 Looking at individual pixels

Now the analysis is extended to individual pixels. The analysis is slightly different then the one in section 3. This is because not every individual pixel can be analysed by use of a plot. Therefore only a handful of pixels are plotted to give an idea of the situation. The average p-value over all pixels tells something about the quality of the fit configuration.

First the fit configuration that is currently used is analysis. This is a straight line fit between trim 0 and F. Figure 12 the situation for nine individual pixels. The average p-value is calculated for all five data sets.

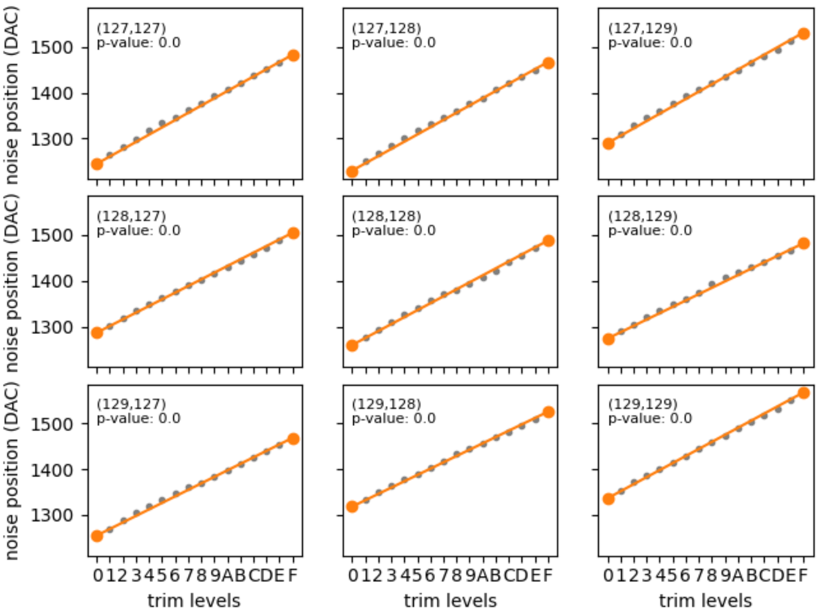

Figure 12: A straight line fit between trim 0 and F for nine individual pixels of data set 1.

The average p-value per data set:

- **data set 1: 0.0**
- **data set 2: 0.0**
- **data set 3: 0.0**
- **data set 4: 0.0**
- **data set 5: 0.0**

The average p-value for all pixels show that the straight line fit is not a good fit configuration.

4.1 Residual of the linear fit

Again the deviation from the straight line is visualised by looking at the residual. This shows the complexity that the straight line is missing.

In the analysis for the average noise position, the residual showed that a third degree polynomial was needed to accurately fit the data points. Therefore this fit configuration is taken as a starting point.

Figure 13 shows the residual for nine individual pixels of data set 1 together with the third degree polynomial fit. The same trim levels are used for the average noise position analysis. Which are trim levels 0,5,A,F for all data sets expect for data set 2, for which 0,4,A,F are used. Again the two outliers found in section 3.1 are ignored when calculating the average p-value.

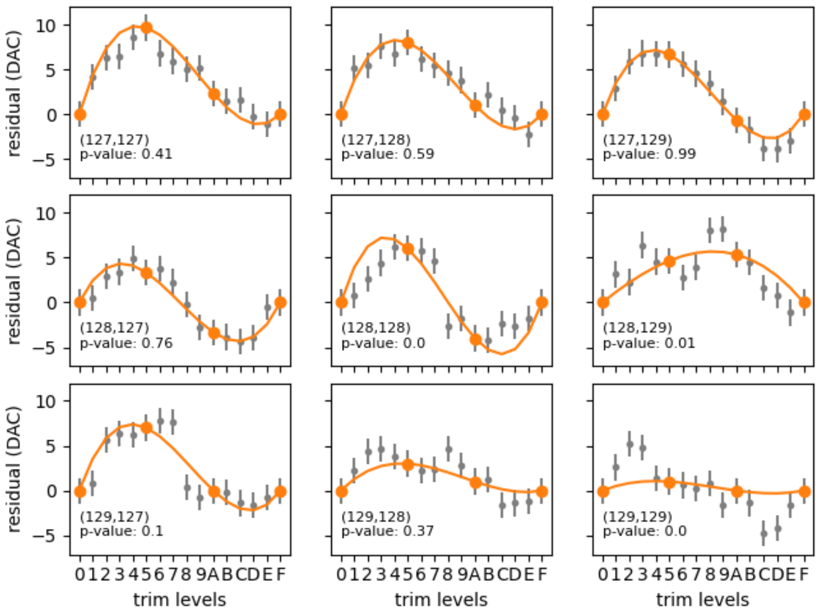

Figure 13: The residual of the straight line fit for nine pixels of data set 1. See Figure 17 in appendix A for the residual of more pixels.

The average p-value:

- **data set 1: 0.48**
- **data set 2: 0.58 (0,4,A,F)**
- **data set 3: 0.49**
- **data set 4: 0.09**
- **data set 5: 0.67**

Again the residual, shown in Figure 13, reveals a remarkable deviation. One similar to that of the average noise position.

There seems to be two different groups. While the pixels with coordinate (127,129), (128,127), (129,129) have the familiar s-shape, (127,127), (127,128), (128,129), (129,127) don't show the same s-shape.

Furthermore, Figure 13 also shows that the choice of trim levels matters a lot. Take for example pixel (129,129) at the bottom left. For this pixel another set of trim levels would have given a much better fit. This can hardly be avoided, since knowing the best trim to fit is impossible when not all trims are measured. This will always remain a guess.

The average p-value for each data set show a major improvement over the straight line fit. The only data set that doesn't seem to be well represented by the third degree polynomial is data set 4, which was already found in the analysis of the average noise position.

4.2 A search for the best fit configuration

For the average noise position, the p-value close to 1 indicated that a third degree polynomial using four data points is a perfect fit configuration for the job. But for individual pixels the p-value show room for even more improvement. In this section many different fit configurations are compared.

4.2.1 Optimize the fit degree and trim level amount to use

Figure 14 shows the average p-value over all pixels of data set 1, for a range of different fit configurations. The degree of the fitted polynomial is varied between 1 and 5. The amount of trims used for fitting is varied from the minimum amount needed for each degree, which is the degree plus 1, until all sixteen trim levels are used.

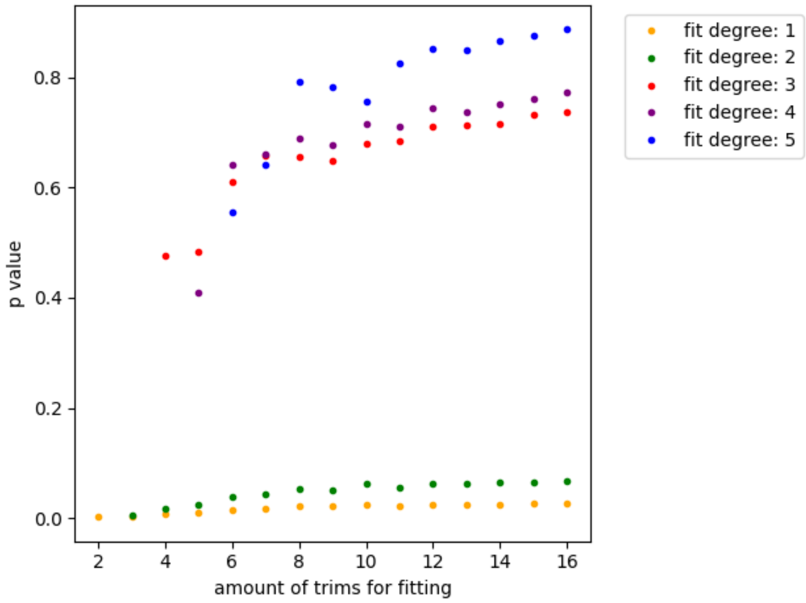

Figure 14: Different configurations for fitting the noise position of individual pixels.

In Figure 14 there is a lot of information. It shows that a first and second order polynomial cannot accurately describe the relation. No matter how much trim levels are used, the p-value stays close to 0.

The third degree polynomial makes a large jump in p-value over the two lower degrees. With just four data points the average p-value rises to about 0.48. Obviously from there the p-value rises when more trim levels are used the fit.

The fourth and fifth degree polynomial are both slightly better then the third degree. But they both need much more data points to make a good fit.

This analysis is expected to be similar for the other data sets.

The third degree polynomial has quite a sweet spot. In the remaining analysis this fit configuration is chosen as the best option.

4.2.2 Optimize which trim levels to use

Until this point the trim levels used for fitting are evenly distributed over the sixteen trim levels. In this section the four trim levels used for the third degree polynomial fit are varied.

This is done by letting a computer calculate the average p-value of all pixels for different sets of trim levels. In total there are $16 * 15 * 14 * 13 = 43680$ possible combinations of four trim levels.

The machine that is used for this research takes 30 seconds to calculate the average p-value for one such fit configuration. Therefore trying out each possibility would take 15 days per data set.

To make the amount of possibilities smaller each of the four trim levels are only given a range of possibilities. The first trim level to use can take trim 0 to 3, the second can take 1 to 7, the third can take 8 to E and the last one can take C to F.

This gave the following result.

Data set	trims to use	average p-value
1	0,4,B,F	0.49
2	0,3,C,F	0.62
3	0,4,B,F	0.51
4	0,4,B,F	0.10
5	0,3,C,F	0.71

Table 1: The best four trims to use for fitting a third degree polynomial for each data set.

Table 1 shows that for 3 data sets the best trims for fitting are 0,4,B and F. The other 2 data sets have 0,3,C and F as best trims for fitting.

It is interesting to see that the two middle trim levels are more to the side. This is probably because the best trim levels to use are the ones where the s-shaped residual has its maximum and minimum. It was already evident in Figure 9 that these extremes are more to the side.

The two data sets for which the best trims to use are 0,3,C and F, are the ones with the outliers.

From this analysis the best trims to use in general are taken to be 0,4,B and F.

5 Making predictions

Until now an attempt has been made to improve the prediction method. It is time to put the improved prediction method to the test. In this section the noise positions are predicted and compared to the measured noise positions. This is only done for data set 1.

In Figure 15 the noise position is predicted by use of the straight line fit between 0 and F. The distributions show how often the prediction differed from the real value by a certain amount.

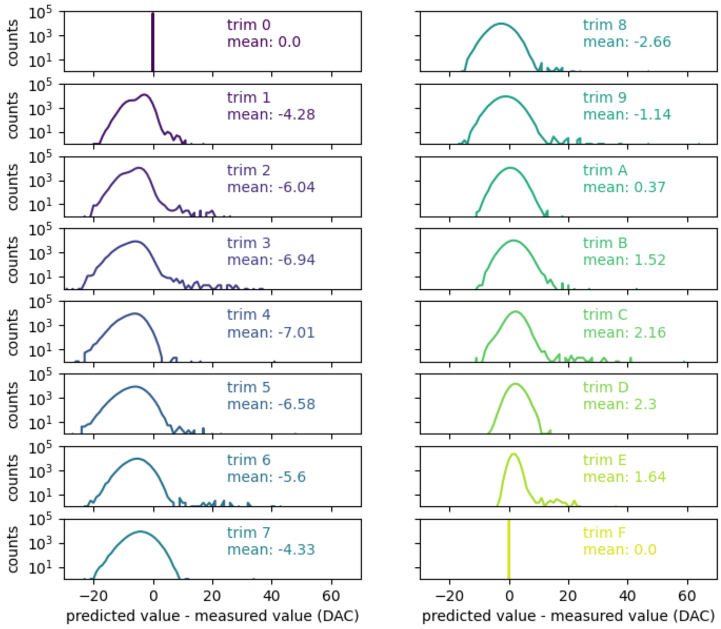

Figure 15: The difference between real value and predicted value for data set 1. Prediction is based on a straight line fit between trim 0 and F.

For trim 0 and F the distributions show a delta function. This is because 0 and F are used to make the straight line so this value can be perfectly predicted.

The rest of the distributions show two important things. First of all, the distributions have a Gaussian like shape. This shows that the predictions are not perfect and that there is a random error.

Second of all, and most importantly, the mean of the distribution is not at zero. Each distribution has its mean at a different position. This deviation is because of the extra shape that is ignored by the straight line fit.

Now the improved prediction function is tested. In Figure 16 the prediction is based on a third degree polynomial fit through trim 0,4,B and F.

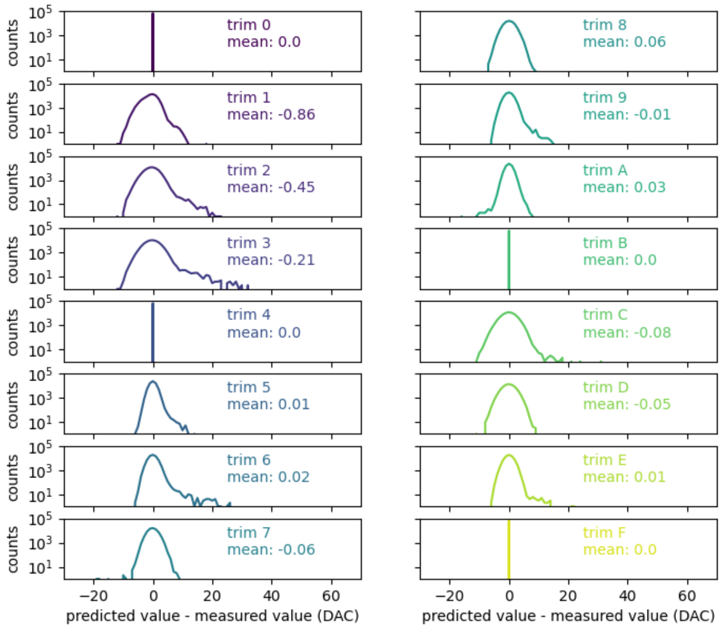

Figure 16: The difference between real value and prediction for data set 1. Prediction is based on a third degree polynomial fit through trim 0,4,B and F.

Again the distribution show a Gaussian like shape. This means that the predictions are not perfect and that there is still an random error. The random error is not systematic so its difficult to make that error smaller. The predictions will never be perfect.

More importantly the Gaussian are now all centered around zero. The deviation of the mean in Figure 15 has disappeared. It shows that it is very important to take the extra s-shaped residual into account when making making predictions.

6 Conclusion and outlook

In this thesis, the equalisation process for the noise of the VELO has been studied. The current assumption of a linear interpolation between the outermost trim levels is questioned and a better prediction method is explored.

This is done by visualizing the noise position on all trim levels and looking for an overall trend. This can clarify how the relation between noise position and trim level is best represented.

The analysis on the average noise positions suggests that there is more to the relation then just a straight line. There is an extra s-shape on top of the straight line that can best be described by a third degree polynomial. To accurately fit this polynomial, four noise positions must be measured instead of the two needed for the straight line.

The analysis on individual pixels shows again that the straight line does not grasp the complexity of the relation. A third degree polynomial fitted through trim level 0,4,B and F is found to be an optimal fit configuration. The average p-value is much improved for four of the five data sets.

One of the data sets could not be well represented by the polynomial fit. This shows that the improved prediction method is not without its flaws. There appears to be ASICs that behave slightly different then the rest.

At last, the new prediction method is put to the test. The predictions of the new method show a clear improvement over that of the straight line fit.

The issue with the equalisation process is the time it has to be done in. If there were no time restrictions all trim levels would simply be measured and no predictions would have to be made. But in reality the equalisation process will be cramped in very short time spans as often as possible.

If the time allows it would be advisable to measure the trim levels 0,4,B and F. These measurements can then be used to fit a third degree polynomial. This fitted function then give an accurate prediction for the noise position on all trim levels.

The equalisation process will happen multiple times on the same pixels. This is because the noise position is expected to chance. After this analysis, I suspect that it is possible to remember the data gathered in an equalisation process, and use this to improve subsequent equalisation processes. I suspect that the shape that was found on top of the linear fit will generally stay the same over time. Therefore such an extra shape can be remembered from previous measurements and added on top of a linear fit. This would allow to have a much better prediction when only measuring two trim levels.

Appendices

A Plots of individual pixels

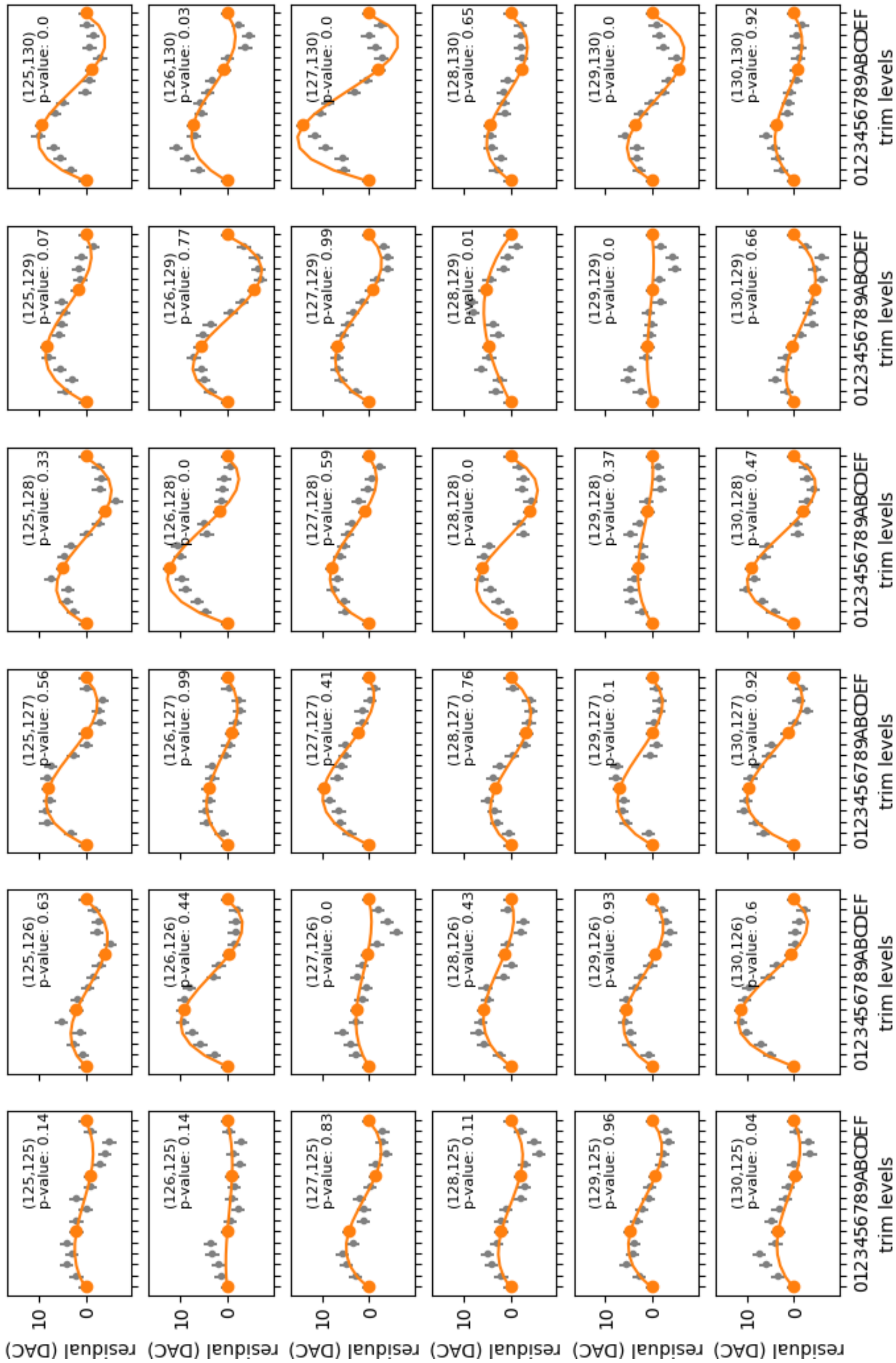

Figure 17: Residual of straight line fit between 0 and F together with a polynomial fit for 36 pixels.

References

[1] Rene Andrae, Tim Schulze-Hartung, and Peter Melchior. "Dos and don'ts of reduced chi-squared". In: (Dec. 2010).

[2] I. Bediaga et al. "Framework TDR for the LHCb Upgrade: Technical Design Report". In: (Apr. 2012).

[3] LHCb Collaboration. *LHCb VELO Upgrade Technical Design Report*. Tech. rep. Nov. 2013. URL: `https://cds.cern.ch/record/1624070`.

[4] Anthony Giraudo. "Study of the Velopix pixels noise". bachelor thesis. 2018.

[5] lumenlearning. *The Uniform Distribution*. URL: `https://courses.lumenlearning.com/odessa-introstats1-1/chapter/the-uniform-distribution/` (visited on 06/04/2021).