



Machine Protection & Electrical Integrity  
TE-MPE

September 2023  
**Technical note 2023-03**  
johannes.sadler@warwick.ac.uk  
EDMS Nr: 2965246

## TECHNICAL NOTE

# Investigation of UFO dynamics in the LHC using displaced bunches

Johannes Sadler

Supervisor: Christoph Wiesner

CERN, CH-1211 Geneva, Switzerland

Keywords: Machine Protection, LHC, UFO, diamond Beam Loss Monitors, Displaced Bunches

---

### Summary

Dust particles with typically tens of micrometers radius are believed to be the cause of millisecond beam loss spikes observed at the LHC. These dust particles have become known as unidentified falling objects (UFOs). They are detrimental to machine availability, since they can cause beam dumps and even magnet quenches. The mechanism by which UFOs are released into the beam is unknown, which prevents the prediction of future UFO event rates. Mapping the trajectory of a UFO could lead to an improved understanding of the release mechanism. The position of a UFO can be calculated from bunch-by-bunch loss signals of bunches that are displaced in the transverse plane. These bunch-by-bunch loss signals are obtained from diamond beam loss monitors (dBLMs). This report presents both a code to calculate the UFO position and an analysis of dBLM data from 2023. A Python library was created to process the dBLM data, and a filtering algorithm was used to identify UFO events from the dataset. The work was performed within the Summer Student project of Johannes Sadler in 2023.

The developed scripts can be found on the following GitLab repository:  
<https://gitlab.cern.ch/machine-protection/ufo-studies>

---

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Displaced bunches</b>	<b>4</b>
2.1	Analytic description . . . . .	4
2.2	Geometric interpretation . . . . .	5
2.3	Calculation of UFO position from N bunches . . . . .	8
2.3.1	Algorithm and implementation . . . . .	8
2.3.2	Testing and results . . . . .	9
2.4	Discussion . . . . .	9
<b>3</b>	<b>Diamond BLM data analysis</b>	<b>11</b>
3.1	A typical UFO signal . . . . .	11
3.2	Data processing on Apache Spark . . . . .	14
3.3	Filtering: search for UFOs . . . . .	15
<b>4</b>	<b>Conclusion</b>	<b>16</b>
<b>5</b>	<b>Acknowledgments</b>	<b>16</b>
<b>A</b>	<b>UFO event plots</b>	<b>19</b>

# 1 Introduction

Sporadic millisecond beam loss spikes observed at the LHC are thought to be caused by interactions between the circulating proton beam and dust particles. These dust particles have come to be known as unidentified falling objects (UFOs). Transient beam loss events, thought to be caused by UFOs, have been under study since the start of the LHC due to their relevance for machine protection and their impact on availability. In Run II alone (2015-2018), 139 beam dumps and 12 magnet quenches were attributed to UFOs [1]. Each beam dump results in 1-2 hours of lost beam time, whilst each magnet quench typically requires a recovery time of 10-12 hours. The origin of the dust particles and their release mechanism are both unknown. Their better understanding is vital to estimate the criticality of UFOs for future machines.

UFO events occur all around the LHC circumference, last a maximum of 1-2 ms, and can have either positively or negatively skewed asymmetric Gaussian loss profiles [1]. Studies on beam-dust interactions in the LHC have reported that the dust particles are initially negatively charged [2, 3]. The prevailing hypothesis is that the dust particle is attracted by the electric field of the circulating proton beam, or falls into the beam through gravity. Electrons are ejected from the dust particle when it interacts with the beam protons, resulting in a positive charge build-up and subsequent repulsion from the beam.

During the beam-dust interaction, a small fraction of the beam protons undergo an inelastic nuclear collision, resulting in local particle showers that can be detected by the ionisation chamber beam loss monitors (ICBLMs). These are the main beam loss monitoring system of the LHC, with a minimum integration time of 40  $\mu s$ , positioned all around the 26.7 km ring. We do not consider the ICBLMs further in this project, because they cannot distinguish bunch-by-bunch losses. Other beam protons are inelastically scattered and continue to travel around the ring until they hit the primary collimators in the betatron collimation region, located in Insertion Region 7 (IR7) of the LHC. Diamond beam loss monitors (dBLMs) are installed in the collimation region to detect the resulting particle showers. There are 6 dBLM devices, 3 per beam, and they have a time resolution of 1.54 ns (sampling rate 650 MHz) [4]. They can distinguish losses from consecutive proton bunches, and are the focus of the second half of this report.

During LHC Run II, a method to study the dynamics of a UFO was established, based on using bunches of varying transverse sizes, and measuring bunch-by-bunch losses with the dBLMs [1]. The method allowed the UFO position to be estimated on a turn-by-turn basis, subject to a four-fold symmetry which is inherent to the method, i.e. it is not possible to tell which quadrant the UFO is in. It was proposed to remove this four-fold symmetry by using bunches displaced in the transverse plane [5]. This report starts by presenting the analytical framework for the method of displaced bunches, and demonstrates a code for calculating the UFO position. We then analyse the dBLM data from January to August 2023, with the aim of re-establishing the reliable capture of UFO events.

## 2 Displaced bunches

### 2.1 Analytic description

We begin by assuming a Gaussian proton density profile of the bunches in the transverse planes, and that losses are proportional to the proton flux through the dust particle. Therefore the measured losses for a bunch  $i$  are given by

$$\text{signal} \propto \frac{N_i}{\sigma_{xi}\sigma_{yi}\sqrt{2\pi}} \exp -\frac{1}{2}\left(\frac{x_i^2}{\sigma_{xi}^2} + \frac{y_i^2}{\sigma_{yi}^2}\right)[5], \quad (1)$$

where  $N_i$  is the bunch intensity,  $\sigma_{xi}$  and  $\sigma_{yi}$  are the transverse bunch sizes, and  $(x_i, y_i)$  is the position of the dust particle measured from the centre of the bunch. We define  $m_i$  to be the loss signal normalised by the bunch intensity, and take the ratio of normalised losses between bunch  $i$  and bunch  $j$ :

$$\begin{aligned} \frac{m_i}{m_j} &= \frac{\sigma_{xj}\sigma_{yj} \exp -\frac{1}{2}\left(\frac{x_i^2}{\sigma_{xi}^2} + \frac{y_i^2}{\sigma_{yi}^2}\right)}{\sigma_{xi}\sigma_{yi} \exp -\frac{1}{2}\left(\frac{x_j^2}{\sigma_{xj}^2} + \frac{y_j^2}{\sigma_{yj}^2}\right)} \\ &= \frac{\sigma_{xj}\sigma_{yj}}{\sigma_{xi}\sigma_{yi}} \exp \frac{1}{2}\left(\frac{x_j^2}{\sigma_{xj}^2} - \frac{x_i^2}{\sigma_{xi}^2} + \frac{y_j^2}{\sigma_{yj}^2} - \frac{y_i^2}{\sigma_{yi}^2}\right)[5]. \end{aligned} \quad (2)$$

We now assume that the bunches have the same transverse size, i.e.  $\sigma_{xj} = \sigma_{xi}$  and  $\sigma_{yj} = \sigma_{yi}$ . This is based on observations from the blown-up bunches experiment, where it was found that the variation in bunch sizes in the non-blown-up bunches was less than 3% [1]. Introducing  $\Delta x = x_j - x_i$  and  $\Delta y = y_j - y_i$ , the horizontal and vertical displacements of bunch  $i$  from bunch  $j$ , we obtain:

$$\frac{m_i}{m_j} = \exp \left( \frac{x_j \Delta x}{\sigma_x^2} + \frac{y_j \Delta y}{\sigma_y^2} - \frac{\Delta x^2}{2\sigma_x^2} - \frac{\Delta y^2}{2\sigma_y^2} \right). \quad (3)$$

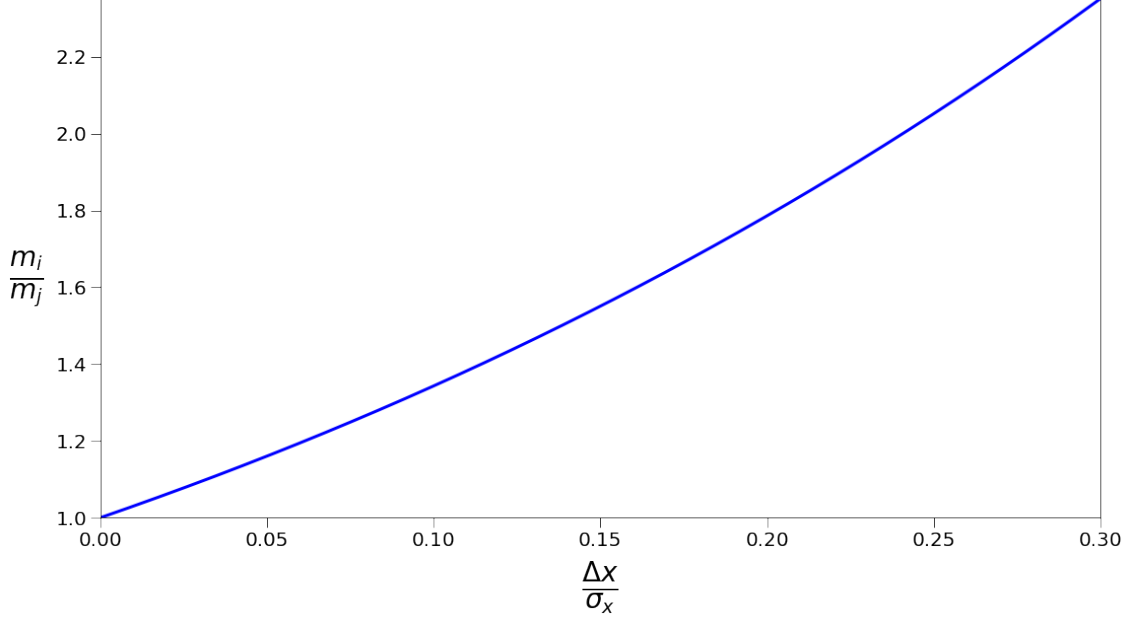
It is useful to non-dimensionalise Eqn. 3 using the scaling  $\tilde{x}_j = \frac{x_j}{\sigma_x}$ ,  $\tilde{y}_j = \frac{y_j}{\sigma_y}$ ,  $\Delta\tilde{x} = \frac{\Delta x}{\sigma_x}$ ,  $\Delta\tilde{y} = \frac{\Delta y}{\sigma_y}$ :

$$\frac{m_i}{m_j} = \exp \left( \tilde{x}_j \Delta\tilde{x} + \tilde{y}_j \Delta\tilde{y} - \frac{\Delta\tilde{x}^2}{2} - \frac{\Delta\tilde{y}^2}{2} \right). \quad (4)$$

Hence we express bunch sizes and displacements in terms of  $\sigma_x$  and  $\sigma_y$ .

To estimate the loss ratio that might be observed, consider a bunch that is displaced in  $x$  only, i.e.  $\Delta y = 0$ . Fixing the UFO position at  $x_j = 3\sigma_x$ , we plot the ratio of normalised losses against the bunch displacement in Fig. 1. The choice of  $x_j = 3\sigma_x$  is based on experimental and numerical studies in Run II, which found a maximum UFO penetration depth of around  $\sim 3\sigma$  from the beam center [6]. It is expected that bunch displacements of the order of  $\sim 0.1\sigma$  are realistic, either caused by different beam-beam kicks or by applying a recurring dipolar kick each turn. Reading off Fig. 1,  $\Delta x = 0.1\sigma_x$  would result in a ratio of losses of 1.34.

To explore a wider parameter range, we plot the normalised loss ratio for varying UFO position and bunch displacement in Fig. 2. We see that a greater loss ratio is expected



**Figure 1:** Ratio of normalised losses against bunch displacement, for a UFO at position  $x_j = 3\sigma_x$

for increased bunch displacement, and UFO positions further from the beam center. The absolute losses, however, will be less for UFOs further from the beam, so it may be more difficult to accurately measure the losses for such UFOs.

Taking the natural logarithm of Eqn. 4 and dropping the tilde notation for convenience yields:

$$\log\left(\frac{m_i}{m_j}\right) = x_j \Delta x + y_j \Delta y - \frac{\Delta x^2}{2} - \frac{\Delta y^2}{2}. \quad (5)$$

Recall that  $(x_j, y_j)$  is the position of the UFO measured from the centre of bunch  $j$ . In order to solve for the UFO position, we fix the origin and assign coordinates to the position of the UFO and the position of each bunch. Let  $(\chi, \psi)$  be the coordinates of the UFO, and  $(X_j, Y_j)$  be the coordinates of bunch  $j$ . Making the substitution  $x_j = \chi - X_j$ ,  $y_j = \psi - Y_j$  and rearranging yields:

$$\chi \Delta x + \psi \Delta y = \log\left(\frac{m_i}{m_j}\right) + X_j \Delta x + Y_j \Delta y + \frac{\Delta x^2}{2} + \frac{\Delta y^2}{2}. \quad (6)$$

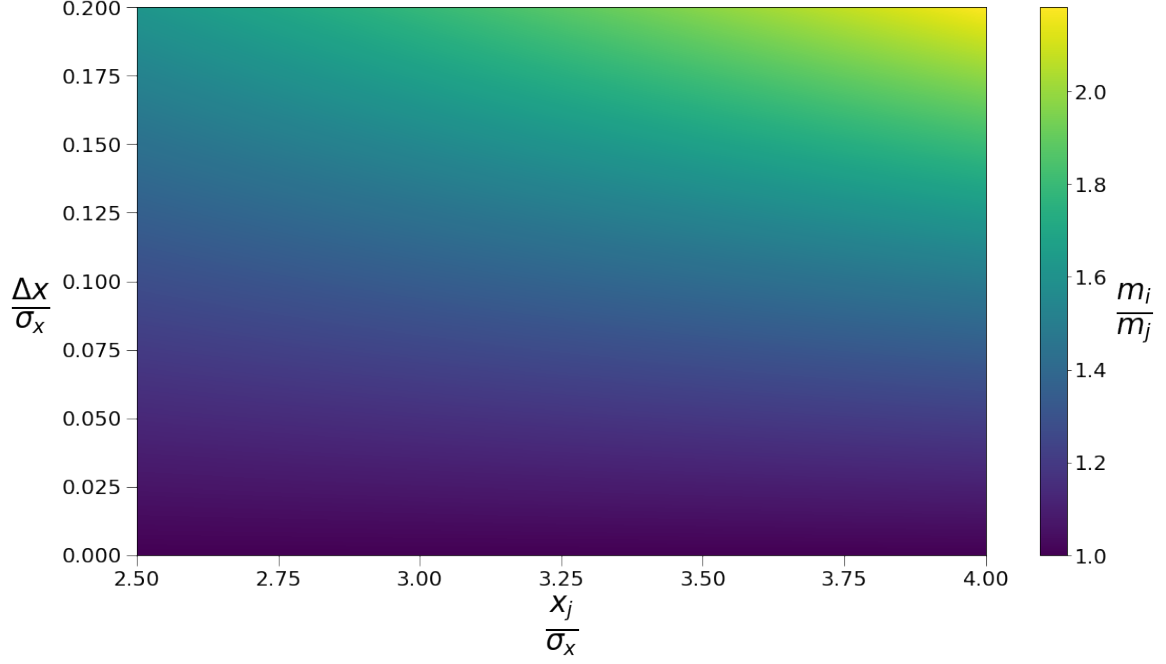
Now  $\Delta x = x_j - x_i = X_i - X_j$ , and similarly  $\Delta y = Y_i - Y_j$ , which yields

$$\chi(X_i - X_j) + \psi(Y_i - Y_j) = \log\left(\frac{m_i}{m_j}\right) + \frac{1}{2}(X_i^2 + Y_i^2 - X_j^2 - Y_j^2). \quad (7)$$

Notice that Eqn. 7 is symmetric under exchange of the bunch labels.

## 2.2 Geometric interpretation

In this section, we give a geometric interpretation of the displaced bunches calculation, that can help visualise the information about the UFO position.



**Figure 2:** Ratio of normalised losses for varying UFO position and bunch displacement. A higher ratio is expected for greater bunch displacement and UFO positions further from the center of the beam.

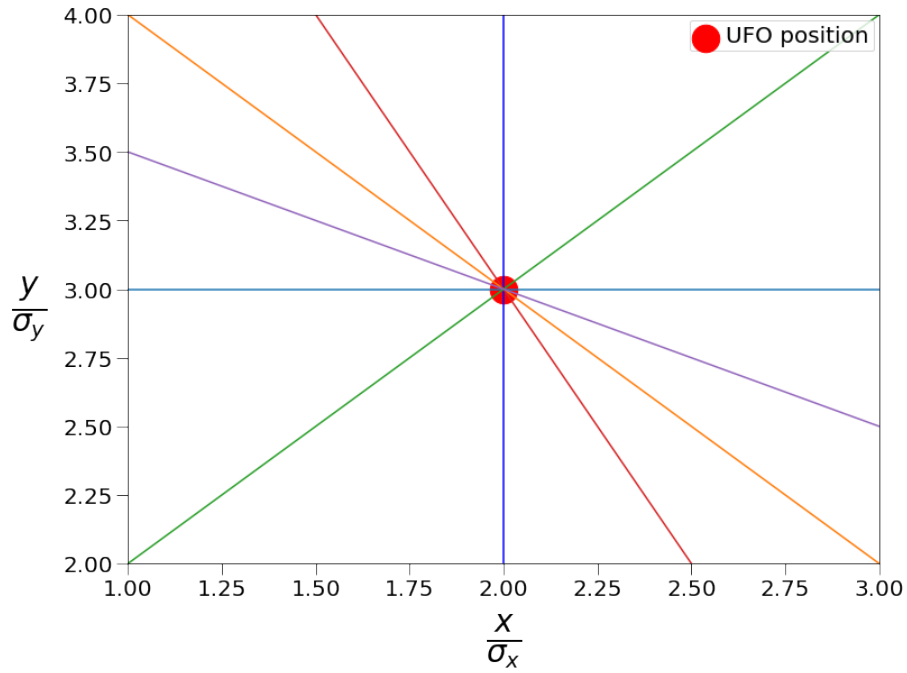
The unknowns in Eqn. 7 are  $\chi$  and  $\psi$ , so we have a straight line of possible UFO positions. Each pair of 2 bunches gives such a line. It can be shown that Eqn. 7 is perpendicular to the line passing through bunches  $i$  and  $j$ . For example, if the bunches are displaced horizontally, so  $\Delta y = 0$ , we obtain a vertical line.

For  $N$  bunches, we obtain  $\binom{N}{2}$  lines. In the ideal case, these lines would all intersect exactly at the position of the dust particle. This is shown in Fig. 3 for the case of a UFO at  $(2, 3)$ , and 4 bunches, resulting in  $\binom{4}{2} = 6$  lines. Here the dust position and bunch positions were chosen, and the model used to calculate the resulting loss ratios. The bunches were positioned at  $(0, 0)$ ,  $(0, 0.1)$ ,  $(0.1, 0)$ ,  $(0.2, 0.2)$ . All  $x$  and  $y$  coordinates are given in units of  $\sigma_x$  and  $\sigma_y$  respectively.

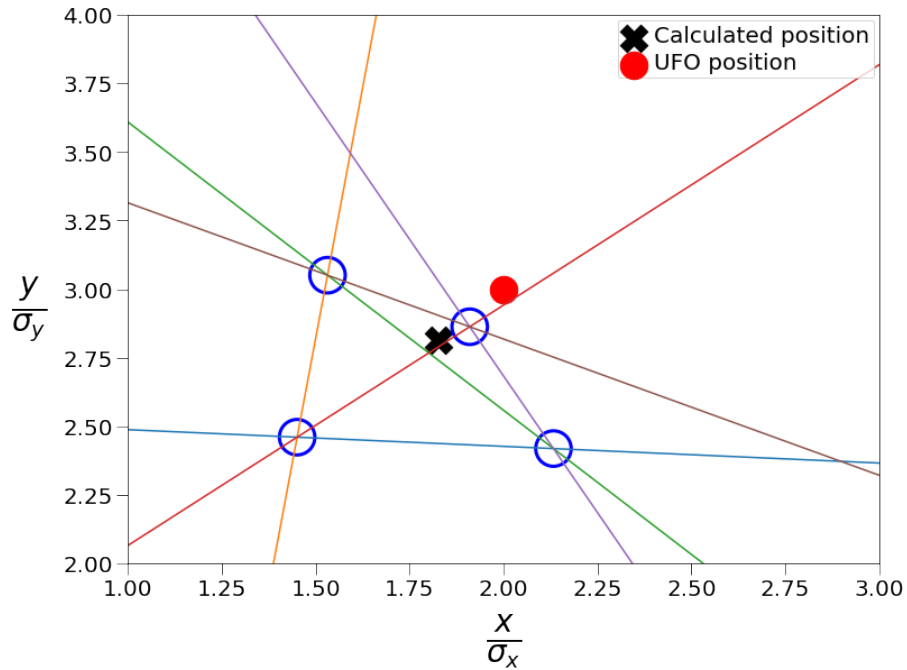
In practice, due to noise in the measurements, the lines will not all intersect at the same point. This is shown in Fig. 4, where 5% Gaussian noise was added to the position and loss signals of each bunch from Fig. 3.

### ***An aside: "triple-intersection" points***

*We know that 2 bunches give a line of possible UFO positions. So for 3 bunches, we have  $\binom{3}{2} = 3$  lines. It can be shown that these 3 lines must all intersect at a single point, which we call a "triple-intersection" point. Thus, for  $N$  bunches, we expect  $\binom{N}{3}$  "triple-intersection" points. In Fig. 4, the  $\binom{4}{3} = 4$  "triple-intersection" points are circled in blue.*



**Figure 3:** Lines of possible UFO position obtained from 4 bunches, positioned at (0,0), (0,0.1), (0.1,0), (0.2,0.2). The UFO position is (2,3). Since we have 4 bunches, we have  $\binom{4}{2} = 6$  lines. The ideal loss signals are calculated from the model, so the lines all intersect at the exact UFO position. All x and y coordinates are given in units of  $\sigma_x$  and  $\sigma_y$  respectively.



**Figure 4:** Lines of possible UFO position obtained from 4 bunches, positioned at (0,0), (0,0.1), (0.1,0), (0.2,0.2). The UFO position is (2,3). We add Gaussian noise to the bunch coordinates and ideal loss signals, so the lines do not all intersect at one point. Since we have 4 bunches, we have  $\binom{4}{3} = 4$  "triple-intersection" points, circled in blue. The UFO position calculated by the code presented in section 2.3 is shown by the black cross. All x and y coordinates are given in units of  $\sigma_x$  and  $\sigma_y$  respectively.

## 2.3 Calculation of UFO position from N bunches

In this section, we consider how to calculate the UFO position when we have position and loss signals of N bunches, and present a Python code to solve the problem. We proceed by looking for an approximate solution to the system of  $\binom{N}{2}$  lines described in section 2.2.

### 2.3.1 Algorithm and implementation

We are considering N bunches with arbitrary transverse positions, so some bunches may have positions in the transverse plane that are close together. Lines from 2 bunches that are close together in the transverse plane can be easily skewed by noise and provide little information about the UFO position. Bunches that are further apart in the plane provide more information. When accounting for the  $\binom{N}{2}$  lines described in section 2.2, we therefore include a weighting according to the transverse separation between the bunches.

We define

$$d_{ij} = \left| \chi(X_i - X_j) + \psi(Y_i - Y_j) - \log\left(\frac{m_i}{m_j}\right) - \frac{1}{2}(X_i^2 + Y_i^2 - X_j^2 - Y_j^2) \right|. \quad (8)$$

For a line in the plane given by  $ax + by + c = 0$ , where  $a$  and  $b$  are not both zero, the distance from the line to a point  $(x_0, y_0)$  is given by

$$\frac{|ax_0 + by_0 + c|}{\sqrt{a^2 + b^2}}. \quad (9)$$

By comparing Eqns. 7, 8 and 9, one can see that  $d_{ij}$  is equal to the distance between the point  $(\chi, \psi)$  and the line given by Eqn. 7, multiplied by the distance between bunches  $i$  and  $j$  in the transverse plane. We now define the "cost function":

$$f(\chi, \psi) = \sum_{(i,j)} d_{ij}^2, \quad (10)$$

where  $(i,j)$  denotes the combination of  $i$  and  $j$ , and the sum is taken over all possible combinations of  $i$  and  $j$ . We minimise the value of the cost function to solve for the UFO position. This is achieved in the code by the *numpy.linalg.lstsq* method [7], which minimises the Euclidean 2-norm  $\|Ax - b\|$ , where  $Ax = b$  is the matrix problem. Thus our calculation of the UFO position involves minimising the sum of the squared distances from the position to each line, weighted by the squared transverse separation between the two bunches.

A Python script was developed to implement the following algorithm:

1. Find the  $\binom{N}{2}$  pairs of bunches from the N bunches given,
2. Use Eqn. 7 to set up the matrix problem,
3. Calculate UFO position by minimising the cost function  $f(\chi, \psi)$  as given in Eqn. 10.

The input parameters for the code are:

1. N - the number of bunches,

2.  $X_s$  - a list of x-coordinates of the bunches,
3.  $Y_s$  - a list of y-coordinates of the bunches,
4.  $m_s$  - a list of loss signals normalised by the bunch intensity.

$X_s$ ,  $Y_s$  and  $m_s$  should all have length  $N$ .  $X_s$  and  $Y_s$  are given in units of  $\sigma_x$  and  $\sigma_y$  respectively. The UFO position calculated from the bunch positions and loss signals in Fig. 4 is shown by the black cross.

### 2.3.2 Testing and results

The code is validated by generating  $N$  bunch positions with  $x$  and  $y$  coordinates from a uniform random distribution in the range  $[-0.05, 0.05]$ , based on the idea that we have  $\sim 0.1\sigma$  displacement. The UFO position is fixed at  $(3,0)$ , the resulting loss ratios are calculated based on the model, and Gaussian noise is added to each position and loss signal. The standard deviation of the noise added to the bunch coordinates is 0.005. Based on the analysis of the dBLM data presented in section 3, noise with a standard deviation of 10% is added to the loss signals. The process of generating  $N$  bunch positions, calculating loss signals, adding noise, and calculating back the UFO position is repeated for  $3 \leq N \leq 48$  and 1000 times for each  $N$ . Here we use the assumption that between 3 and 48 bunches are displaced.

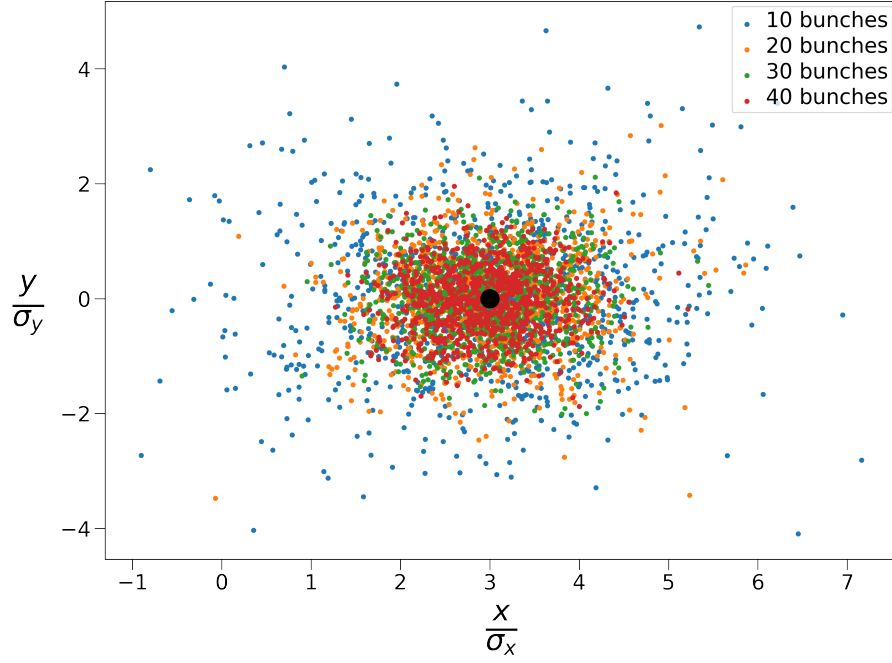
The results for selected  $N$  are shown in Fig. 5. We see that the calculated positions are centered on the true UFO position  $(3,0)$ , and that using more bunches gives a more accurate result. In the case  $\sigma_x = \sigma_y$ , the error (distance from true position  $(3,0)$ ) in units of  $\sigma$  can be calculated for each measurement. Figure 6 depicts the average error against  $N$ . The data for  $N=3$  and  $N=4$  is omitted because the average errors were much higher,  $19.3\sigma$  and  $4.1\sigma$  respectively, which would obscure the rest of the plot. We see that the accuracy of the calculation begins to plateau as  $N$  is increased.

Refinements in the data analysis could reduce the fluctuations in normalised losses from consecutive bunches, so the code was run several times to study different levels of noise. Figure 6 shows the average error in position measurement for Gaussian noise of 5%, 10% and 15%. We see that reducing the noise in the normalised loss signals is key to improving the measurement. For example, with 30 bunches, the average errors for 5%, 10% and 15% noise are  $0.47\sigma$ ,  $0.82\sigma$ ,  $1.25\sigma$  respectively. It was also found that increasing the bunch displacement can greatly improve the UFO position calculation, but here we work with the assumption that  $\sim 0.1\sigma$  displacement is realistic.

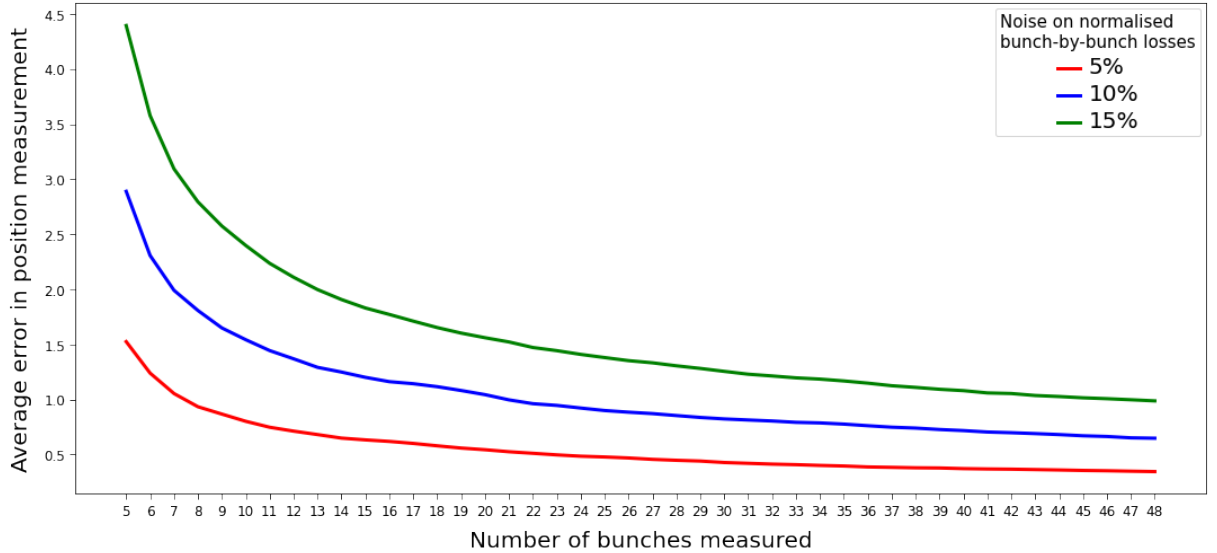
## 2.4 Discussion

We have demonstrated that  $N$  bunches with arbitrary positions can be used to calculate the UFO position. The random uniform distribution of bunch positions is not realistic, but demonstrates the generality of the approach.

At present, there is no reason to believe that there would be significant variation in the transverse sizes of the bunches. If required, however, it would be straight forward to modify this work to account for bunches of varying transverse size. Rather than many straight



**Figure 5:** Calculated UFO positions with Gaussian noise added to the bunch positions and loss signals. The true UFO position is (3,0), shown by the black dot. The calculation was done 1000 times each for 10, 20, 30 and 40 bunches.



**Figure 6:** Average error (distance from true UFO position in units of  $\sigma$ ) after 1000 hypothetical measurements against number of bunches measured, for different levels of Gaussian noise on the normalised loss signals.

lines, one would obtain a system of ellipses, hyperbolas and parabolas. One appropriate choice would be to use the *scipy.optimize.least\_squares* method, which solves nonlinear least-squares problems. Note that one can then no longer express coordinates in units of  $\sigma_x$  and  $\sigma_y$ , since they vary between bunches.

### 3 Diamond BLM data analysis

Diamond BLMs enable losses from consecutive bunches to be distinguished, which facilitates the displaced bunches calculation. During LHC Run II, a real-time UFO detection method ("auto-trigger") was established to trigger read-out during beam operation. The dBLMs also trigger a readout at every injection and beam dump (a "timing event"). After a hardware change during the long shutdown in 2019, the auto-trigger was reactivated in January 2023 to continue UFO studies.

At the time of this project, the information of whether the dBLM read-out was caused by a timing event or the UFO auto-trigger was not stored. To avoid looking at triggers on timing events, triggers occurring after injection and before dump were plotted and checked. Six UFO events were found by this manual approach, but it was observed that the majority of dBLM triggers clearly did not show UFO-like loss behaviour. This motivated a systematic search of the existing data from 2023. In section 3.1 we present a typical UFO event, one of the six found by the method described above. We identify a feature believed to be characteristic of UFO events, which is later used to filter for UFO candidates.

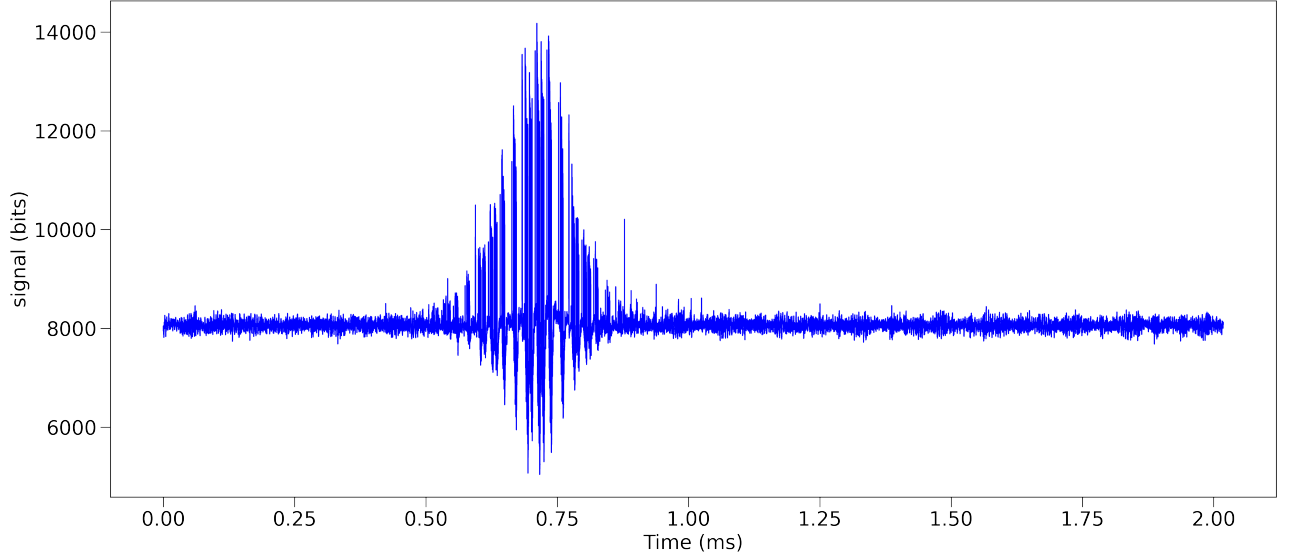
Between the six devices, the number of events from January to August 2023 is 236,847. Retrieving the data for this many events from the logging system is extremely time-consuming. In section 3.2 we present a Python library that was created to process large batches of dBLM events and save them to files, which can then be rapidly read and filtered. In section 3.3 we filter the events to find UFO candidates. This work could help to re-establish the reliable capture of UFO events with the dBLMs.

#### 3.1 A typical UFO signal

Figure 7 shows a typical UFO signal. The event lasted around 0.4 ms, and the loss profile is approximately a symmetric Gaussian. Asymmetric Gaussian UFO events with either a longer rise time or longer fall time have been observed previously [1].

The read-out buffer from the diamond BLMs is given by an array of around 1.3 million values, where each value is a sample from the detector (sample rate is 650 MHz). In Fig. 7, the x-axis has been converted to milliseconds for readability. The other information we get from the dBLM is an array of 'flags', which has the same length as the 'signal' array. The 'flags' tell us which samples correspond to the start of a new 'bunch slot', and which correspond to a new turn. 'Bunch slot' refers to one of the 3564 possible bunch slots of the LHC. With the current 25 ns bunch spacing, one bunch slot corresponds to 10 RF buckets. A subset of the bunch slots are filled with protons during an accelerator cycle. Nominally, 2808 of the 3564 bunch slots are filled.

We combine the 'signal' array and 'flags' array into one data structure: a nested array, where the signal is split into turns, and each turn is split into bunches [8]. We call this data



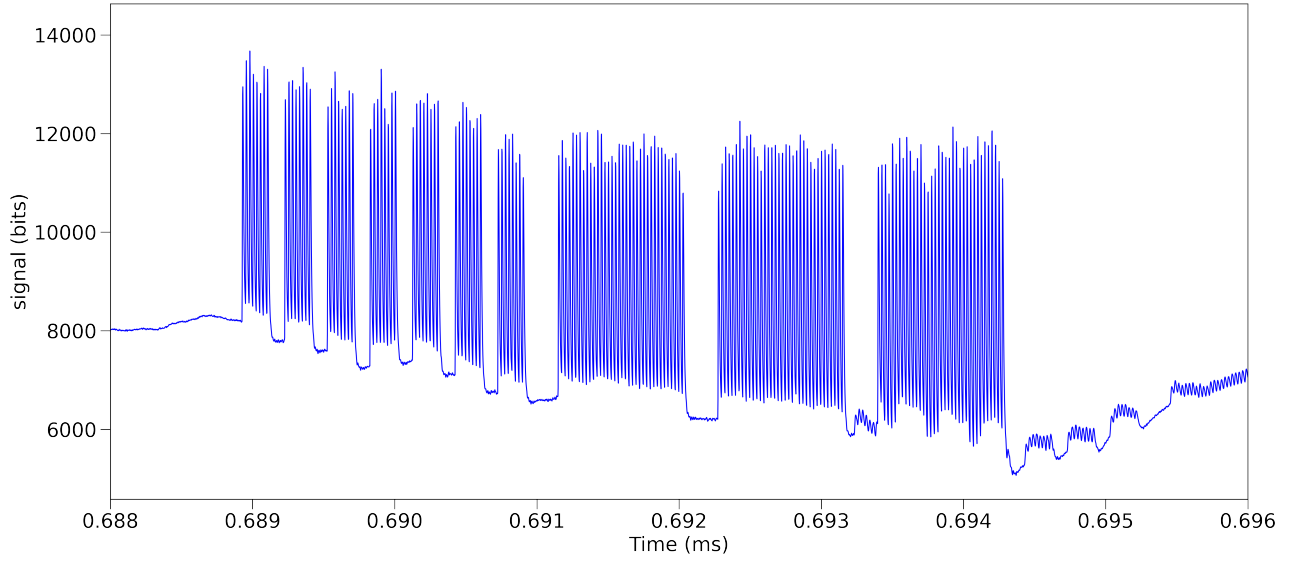
**Figure 7:** A typical UFO signal, recorded on the 05/05/2023 at 20:30:23 by device 'HC.TZ76.BLMDIAMOND3.3'. The event took place at an energy of 5139 GeV, during the 'ramp' phase of the accelerator cycle, with a beam intensity of  $1.48 \times 10^{14}$  protons.

structure 'matrix'. Thus, if we want the loss signal on the 3rd turn from the 1500th bunch slot, we access it by `matrix[2][1499]`. The transformation from 'signal' and 'flags' into matrix allows us to look at the behaviour of specific bunches over time. One challenge associated with this transformation is that the number of samples per bunch is not uniform: some have 16, and others 17, a consequence of the 650 MHz sample rate and 25 ns bunch slots. We use Awkward Arrays to overcome this - a Python library for nested, variable sized data [9].

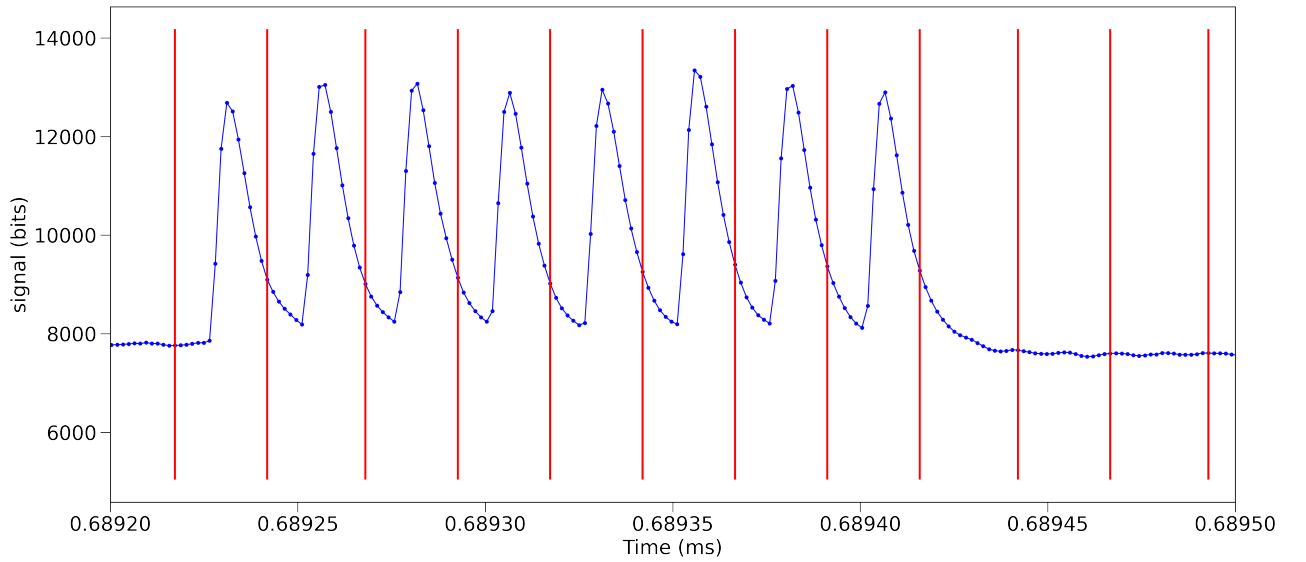
Notice that in Fig. 7 the baseline of the signal drops after the onset of the UFO event. This is designed to increase the dynamic range of the BLM in the case of continuous losses. Figure 8 zooms in on losses from 10 bunch trains during the UFO event, and we clearly see the signal drop below the baseline when consecutive bunch trains incur losses.

In order to quantify the losses for a specific bunch, we have to account for the baseline drop. Figure 9 shows an even closer look at the signal, focusing on a train of 8 bunches. Here the 'flags' information is included: the red vertical lines denote the start of a new bunch slot. An LHC bunch slot is 25 ns, but the protons are concentrated inside one RF bucket with a typical 4-sigma bunch length of about 1 ns. Losses from the proton bunch cause charge to accumulate in the detector, which then exponentially decays. There is an overlap of the exponential decay signal into the next bunch. We choose to take the range (maximum value minus minimum value) of the samples belonging to a bunch as a first approximation for the bunch-by-bunch losses. The more rigorous approach, involving fitting an exponential decay, extrapolating and numerically integrating, would be more challenging for processing a large number of events.

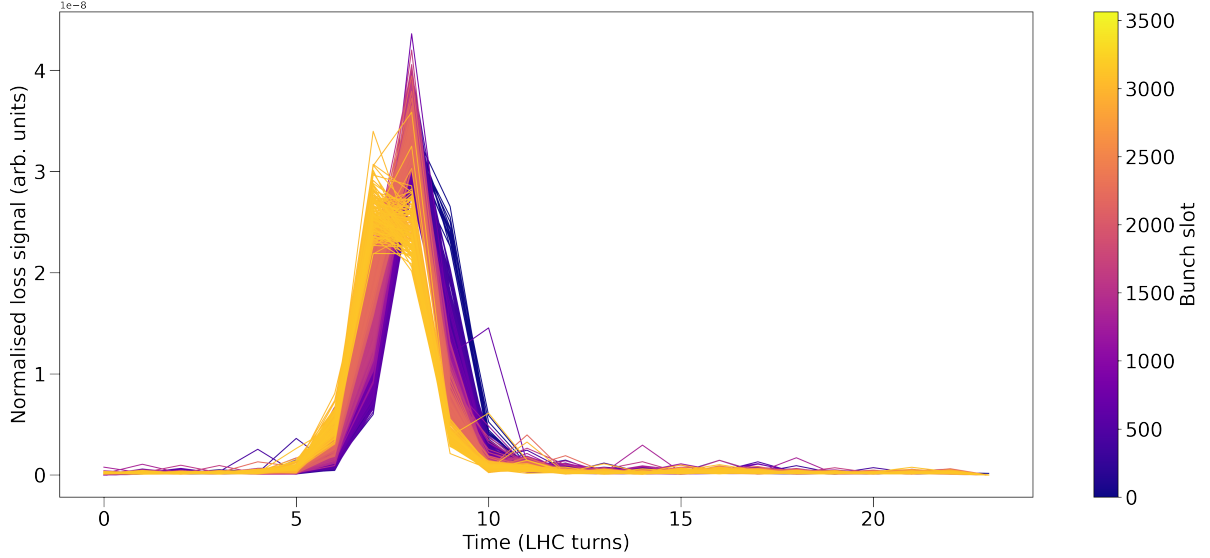
Guided by the displaced bunches calculation in section 2, we would like to normalise our bunch-by-bunch losses by the bunch intensities. We therefore query the filling pattern and bunch intensities, which come from other LHC devices. They record data at 1 Hz, so the timestamps have to be aligned as best as possible with the dBLM event timestamp. We use these data to carry out the normalisation: The dBLM signals of the filled bunch slots



**Figure 8:** Zooming in on losses from 10 bunch trains from the UFO event in Fig. 7. The first 7 trains consist of 8 bunches, and the last 3 trains consist of 36 bunches. The trains of 8 are separated by 4 empty bunch slots, whilst the trains of 36 are preceded by 9 empty bunch slots. The baseline drop is clearly visible.



**Figure 9:** A closer look at the second bunch train from Fig. 8, which consists of 8 bunches. The blue points are the samples from the detector, which have been joined together in a line plot. The vertical red lines denote the start of a new bunch slot, and come from the 'flags' data of the diamond BLMs. We see that the signal from a bunch decays exponentially, and that there is an overlap into the next bunch.



**Figure 10:** Normalised losses of filled bunch slots plotted over LHC turns, for the typical UFO signal from Fig. 7. The bunch slot number is shown by the colour bar. We see that consecutive bunches see similar loss profiles. For turns 7, 8 and 9, all bunches see losses from the UFO.

are divided by their corresponding intensity, and unfilled bunch slots are set to zero [8]. We call the resulting array 'peak\_n'. It contains the normalised losses for each filled bunch slot, turn-by-turn.

Using 'peak\_n', we plot the normalised losses for filled bunch slots over LHC turns, shown in Fig. 10. Consistent with expectations, consecutive bunches see similar normalised loss profiles. There are turns where all bunches incur losses. This is characteristic of UFO events, and the filtering algorithm uses this feature.

### 3.2 Data processing on Apache Spark

A Python library, which we call pydust [8], was created to process large batches of dBLM trigger events. Pydust uses Apache Spark, a distributed computing framework designed for large volumes of data [10]. For each of the 236,847 events, the transformation of 'signal' and 'flags' into 'matrix' is carried out as described in section 3.1. The bunch intensities are queried and used to carry out the normalisation to obtain 'peak\_n'. Other important parameters are also queried: beam mode, energy and total intensity. The information is stored in a Spark dataframe.

During the development of Pydust, it was discovered that the dBLM trigger events were often clustered on certain days. For example, on April 21st 2023, device 'HC.TZ76.BLMDIA MOND3.3' alone had 38,403 events. We know that the number of injections, beam dumps, and UFO events in a single day should be orders of magnitude less than this, so the UFO auto-trigger was triggering too often. It was also discovered that some events had 'missing bunches', i.e. we had less than the expected 3564 bunch slots in a turn. The events with 'missing bunches' were predominantly on days when we had a high number of triggers, which suggests the 'missing bunches' are linked with triggering too often. For example, of the 38,403 triggers of device 'HC.TZ76.BLMDIAMOND3.3' on April 21st, 38,389 have

'missing bunches'. We keep track of events with the 'missing bunches' issue by adding a column called 'bunch\_count\_inconsistencies' to the dataframe.

The resulting dataframe has 236,847 rows, one row for each dBLM trigger. The columns include: 'device', 'year', 'month', 'day', 'hour', 'minute', 'second', 'matrix', 'peak\_n', 'bunch\_intensities', 'total\_intensity', 'energy' and 'bunch\_count\_inconsistencies'. The dataframe is saved to a file in parquet format ( $\sim 400\text{GB}$ ). This file can be read back into a Spark dataframe for further processing and filtering.

### 3.3 Filtering: search for UFOs

Guided by the UFO characteristic discussed in section 3.1 and the 'missing bunches' issue discussed in section 3.2, we filter the processed data as follows:

1. Remove all the events with 'bunch\_count\_inconsistencies' equal to 'True'.
2. Choose events with losses for 1-20 turns. This is to remove events with continuous losses across all the turns. Having losses on a given turn is defined as follows: 80% of the filled bunch slots should have normalised losses over  $0.1 \times 10^{-8}$  (arb. units).

Step 2 is achieved by using a Spark user-defined function (UDF) to add a column called 'lossturns' to the dataframe, and then choosing events with  $1 \leq \text{lossturns} \leq 20$ . We expect 100% of the filled bunch slots to see losses, but we use 80% to be lenient. The initial 6 UFO events that were found had losses an order of magnitude greater than  $0.1 \times 10^{-8}$  (arb. units), however this low threshold was already sufficient to remove most of the 236,847 events.

The number of events by device through each stage of the filtering is shown in table 1. Filtering step 1 reduces the number of events from 236,847 to 107,960. Then, filtering step 2 brings the number down to 71. These 71 events were checked manually and given a 'manual\_UFO\_label': 'True' for a UFO, 'False' for not a UFO, and 'None' for ambiguous cases. Of the 71 events picked out by the filter, 26 were labelled as UFOs, 38 as not UFOs, and 7 were ambiguous. We see that two of the devices were particularly sensitive, with devices 3.3 and 3.5 seeing 23 of the 26 events labelled as UFOs. Devices 3.3 and 3.5 are positioned in the corresponding location for beams 1 and 2 respectively, suggesting that this location is best suited for detecting UFOs. The results also revealed that 25 of the 26 events labelled as UFOs took place during the 'ramp' beam mode, with the one other event during stable beams. This is surprising, since the time spent in the ramp is a fraction of that spent during stable beams.

The filtered dataframe of 71 events, including a column with the 'manual\_UFO\_label', was saved to a file in parquet format for further study. Additionally, a pdf document was generated, which contains the following for each of the 71 UFO candidates: the plot of the raw signal, the plot of filled bunch slots over turns, and information about the event (total intensity, beam mode, etc.). The plots of filled bunch slots over turns for the 26 events labelled as UFOs can be found in Appendix A.

**Table 1:** Results of filtering applied to the 236,847 dBLM events from January to August 2023. The number of events by device is given at each stage of the filtering. Filter 1 and filter 2 correspond to the filtering steps described in section 3.3. Devices ending in '.3' belong to beam 1, whilst devices ending in '.5' belong to beam 2. 71 events were picked out by the filtering, 26 of which were labelled manually as UFOs.

Device	All events	Filter 1	Filter 2	Labelled UFOs
HC.TZ76.BLMDIAMOND3.3	62,745	9,294	27	13
HC.TZ76.BLMDIAMOND3.5	67,420	12,958	30	10
HC.TZ76.BLMDIAMOND2.3	10,050	7,405	4	1
HC.TZ76.BLMDIAMOND2.5	17,870	1,207	2	1
HC.TZ76.BLMDIAMOND.3	16,186	15,021	4	0
HC.TZ76.BLMDIAMOND.5	62,576	62,075	4	1
<b>Total</b>	<b>236,847</b>	<b>107,960</b>	<b>71</b>	<b>26</b>

## 4 Conclusion

A code was created to calculate the UFO position from measurements of  $N$  bunches with arbitrary transverse positions, without being skewed by bunches with a small transverse separation. To carry out the displaced bunches experiment in practice, we need to reliably capture UFO events on the diamond BLMs. A systematic search of the dBLM data from January to August 2023 found 71 UFO candidates from a total of 236,847 events. The 71 UFO candidates were saved to a file in parquet format for further study, and 26 of the 71 events were manually identified to be UFOs. Complex queries can be rapidly applied to the processed dBLM data from this project, which may be useful to re-establish the reliable capture of UFO events.

## 5 Acknowledgments

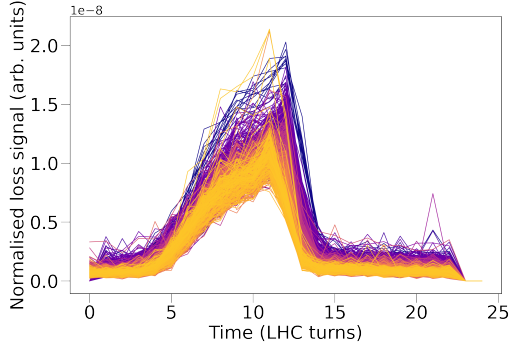
I would like to express my gratitude to Christoph Wiesner for his invaluable guidance during this project, and for providing many insightful comments and suggestions. I would like to thank Cedric Hernalsteens for his work developing the pydust library and post-processing the dBLM data. Thanks also to Eva Calvo and Manuel Gonzalez Berges for fruitful discussions on the diamond Beam Loss Monitors and to Bjorn Hans Filip Lindstrom for sharing his experience and knowledge of UFO studies. Finally, thank you to Daniel Wollmann for enabling this summer project in his section.

## References

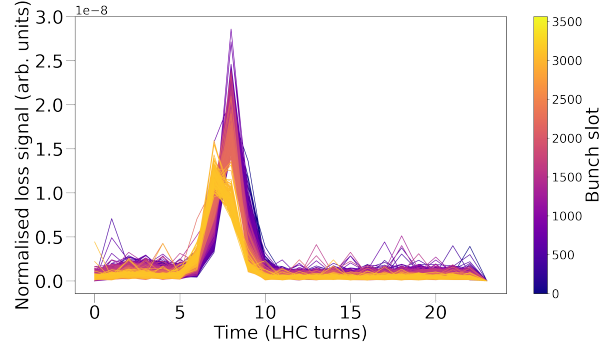
- [1] B. Lindstrom et al. “Dynamics of the interaction of dust particles with the LHC beam”. In: *Phys. Rev. Accel. Beams* 23 (12 Dec. 2020), p. 124501. DOI: 10.1103/PhysRevAccelBeams.23.124501.
- [2] P. Bélanger et al. “Charging mechanisms and orbital dynamics of charged dust grains in the LHC”. In: *Phys. Rev. Accel. Beams* 25 (10 Oct. 2022), p. 101001. DOI: 10.1103/PhysRevAccelBeams.25.101001.
- [3] A. Lechner et al. “Dust-induced beam losses in the cryogenic arcs of the CERN Large Hadron Collider. Study of dust-induced beam losses in the cryogenic arcs of the CERN Large Hadron Collider”. In: *Phys. Rev. Accel. Beams* 25.4 (2022), p. 041001. DOI: 10.1103/PhysRevAccelBeams.25.041001.
- [4] *SPS & LHC BLM Diamonds System*. Accessed on November 5, 2023. URL: <https://wikis.cern.ch/pages/viewpage.action?pageId=94539715>.
- [5] Bjorn Hans Filip Lindstrom. “Criticality of fast failures in the High Luminosity Large Hadron Collider”. Presented 27 Jan 2021. PhD thesis. 2021. URL: <https://cds.cern.ch/record/2762088>.
- [6] B Lindstrom et al. “Results of UFO dynamics studies with beam in the LHC”. In: *Journal of Physics: Conference Series* 1067.2 (Sept. 2018), p. 022001. DOI: 10.1088/1742-6596/1067/2/022001.
- [7] *NumPy documentation*. Accessed on November 5, 2023. URL: <https://numpy.org/doc/stable/reference/generated/numpy.linalg.lstsq.html>.
- [8] C. Hernalsteens. *Query and Postprocessing of dBLM data for UFO studies*. Private Communication. Aug. 2023.
- [9] *Awkward Array documentation*. Accessed on November 5, 2023. URL: <https://awkward-array.org/doc/main/>.
- [10] *PySpark Overview*. Accessed on November 5, 2023. URL: <https://spark.apache.org/docs/latest/api/python/index.html>.



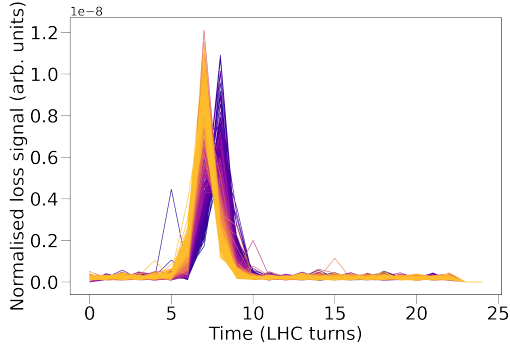
## A UFO event plots



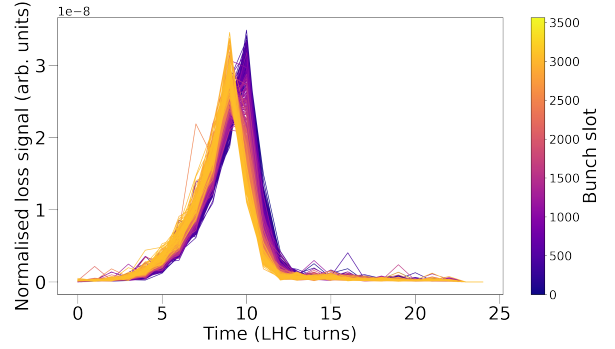
(a) Device 'HC.TZ76.BLM DIAMOND3.3', 01/05/2023 at 10:54:43, Beam mode: RAMP, Energy: 6526 GeV, Total intensity:  $1.25 \times 10^{14}$  protons.



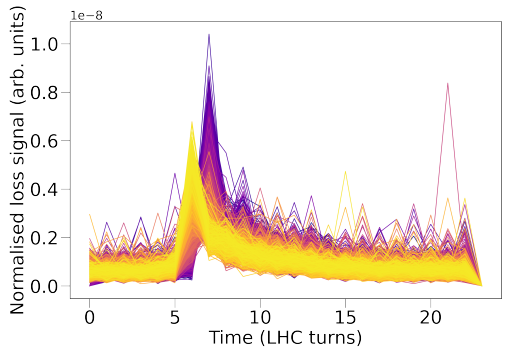
(b) Device 'HC.TZ76.BLM DIAMOND3.3', 06/05/2023 at 03:24:50, Beam mode: RAMP, Energy: 4317 GeV, Total intensity:  $1.44 \times 10^{14}$  protons.



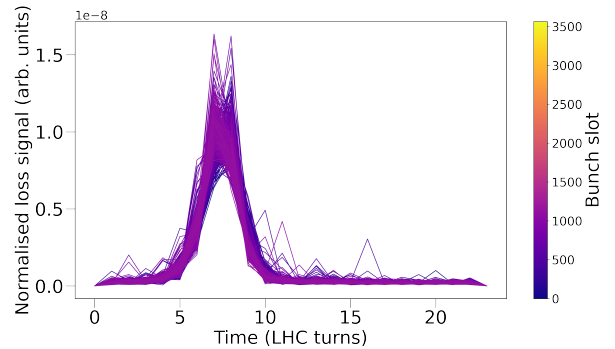
(c) Device 'HC.TZ76.BLM DIAMOND3.3', 27/04/2023 at 21:07:43, Beam mode: RAMP, Energy: 1211 GeV, Total intensity:  $5.75 \times 10^{13}$  protons.



(d) Device 'HC.TZ76.BLM DIAMOND3.3', 27/04/2023 at 18:17:09, Beam mode: RAMP, Energy: 6005 GeV, Total intensity:  $6.00 \times 10^{13}$  protons.

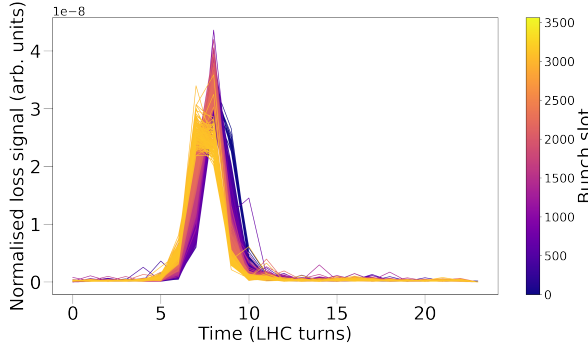


(e) Device 'HC.TZ76.BLM DIAMOND3.3', 08/06/2023 at 00:08:18, Beam mode: RAMP, Energy: 6593 GeV, Total intensity:  $3.63 \times 10^{14}$  protons.

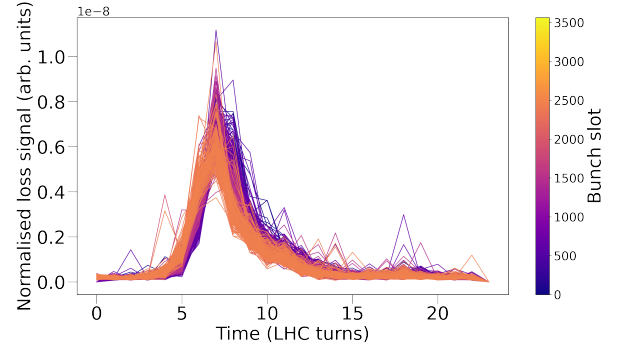


(f) Device 'HC.TZ76.BLM DIAMOND3.3', 30/05/2023 at 16:40:29, Beam mode: RAMP, Energy: 1064 GeV, Total intensity:  $9.61 \times 10^{13}$  protons.

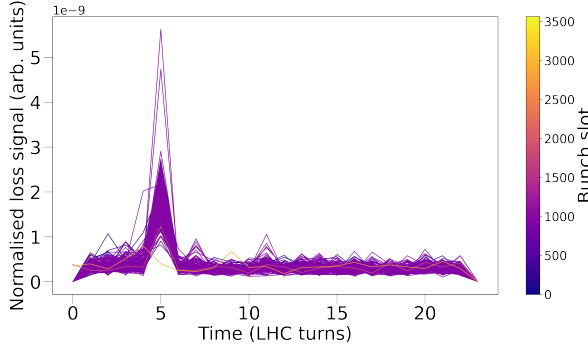
**Figure 11:** dBLM trigger events identified by the filtering and manually labelled as UFOs. Normalised losses are plotted over LHC turns for each filled bunch slot. The bunch slot number is shown by the colour bar. (Part 1)



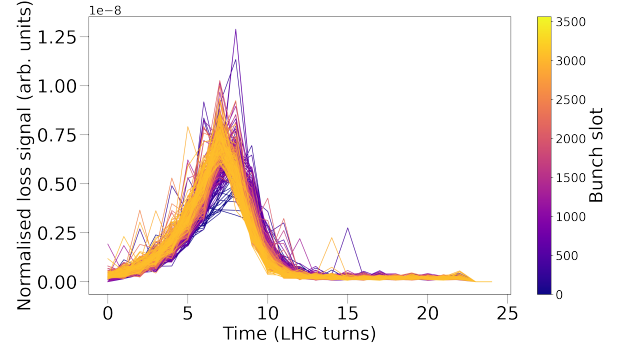
(g) Device 'HC.TZ76.BLMDIAMOND3.3', 05/05/2023 at 20:30:23, Beam mode: RAMP, Energy: 5139 GeV, Total intensity:  $1.48 \times 10^{14}$  protons.



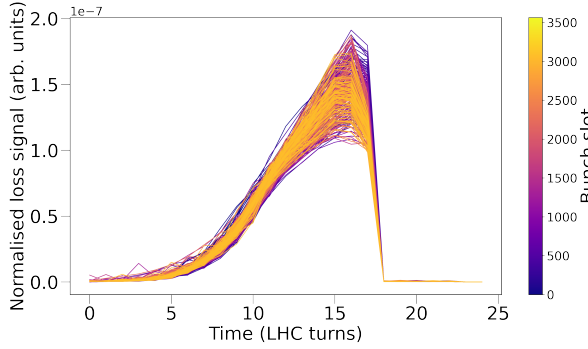
(h) Device 'HC.TZ76.BLMDIAMOND3.3', 05/05/2023 at 11:09:18, Beam mode: RAMP, Energy: 6416 GeV, Total intensity:  $5.85 \times 10^{13}$  protons.



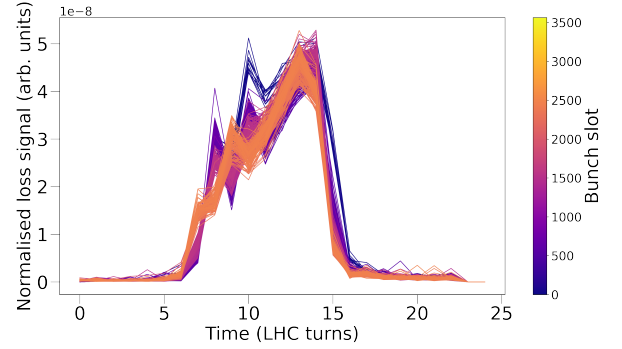
(i) Device 'HC.TZ76.BLMDIAMOND3.3', 01/07/2023 at 01:09:50, Beam mode: RAMP, Energy: 5214 GeV, Total intensity:  $1.27 \times 10^{13}$  protons.



(j) Device 'HC.TZ76.BLMDIAMOND3.3', 28/04/2023 at 17:45:43, Beam mode: RAMP, Energy: 1108 GeV, Total intensity:  $6.15 \times 10^{13}$  protons.

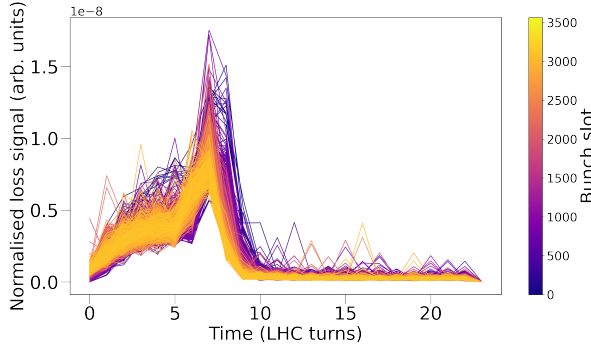


(k) Device 'HC.TZ76.BLMDIAMOND3.3', 28/04/2023 at 23:14:06, Beam mode: RAMP, Energy: 4920 GeV, Total intensity:  $2.59 \times 10^{13}$  protons.

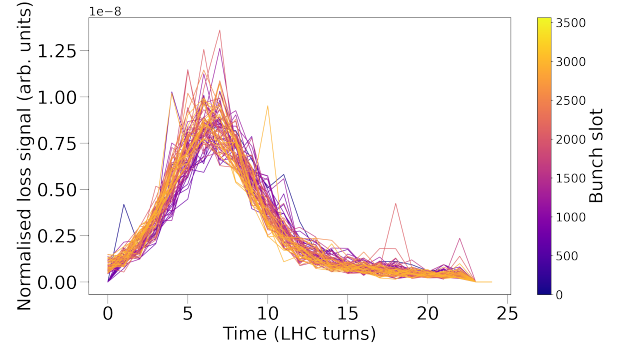


(l) Device 'HC.TZ76.BLMDIAMOND3.3', 29/04/2023 at 17:51:43, Beam mode: RAMP, Energy: 5367 GeV, Total intensity:  $5.55 \times 10^{13}$  protons.

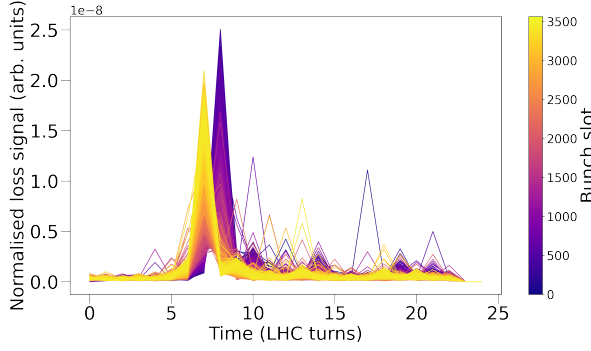
**Figure 11:** dBLM trigger events identified by the filtering and manually labelled as UFOs. Normalised losses are plotted over LHC turns for each filled bunch slot. The bunch slot number is shown by the colour bar. (Part 2)



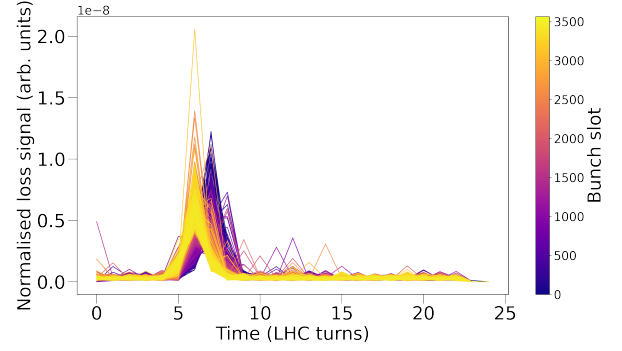
(m) Device 'HC.TZ76.BLMDIAMOND3.3', 07/05/2023 at 20:22:06, Beam mode: RAMP, Energy: 6242 GeV, Total intensity:  $2.59 \times 10^{14}$  protons.



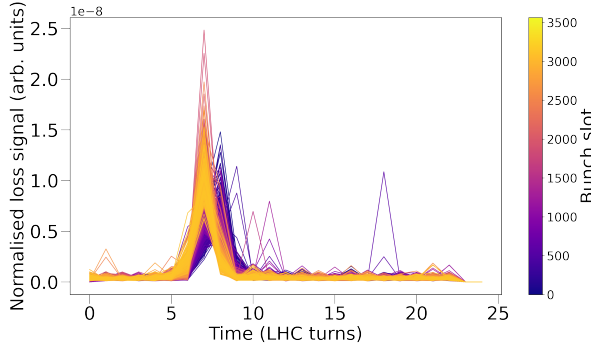
(n) Device 'HC.TZ76.BLMDIAMOND3.5', 24/04/2023 at 23:02:14, Beam mode: RAMP, Energy: 2057 GeV, Total intensity:  $1.13 \times 10^{13}$  protons.



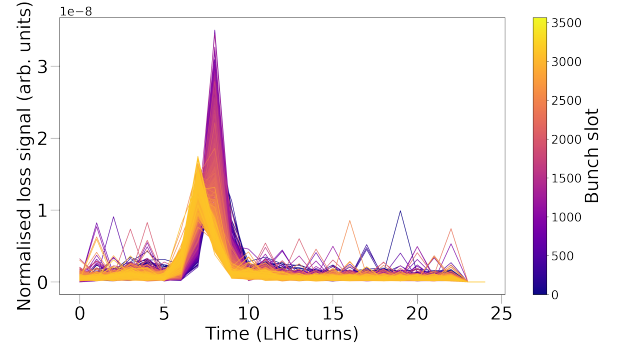
(o) Device 'HC.TZ76.BLMDIAMOND3.5', 08/06/2023 at 00:05:47, Beam mode: RAMP, Energy: 5731 GeV, Total intensity:  $3.60 \times 10^{14}$  protons.



(p) Device 'HC.TZ76.BLMDIAMOND3.5', 12/07/2023 at 12:05:37, Beam mode: RAMP, Energy: 5596 GeV, Total intensity:  $3.96 \times 10^{14}$  protons.

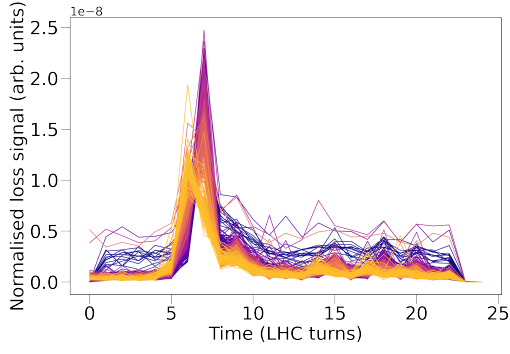


(q) Device 'HC.TZ76.BLMDIAMOND3.5', 10/05/2023 at 03:13:55, Beam mode: RAMP, Energy: 5957 GeV, Total intensity:  $2.48 \times 10^{14}$  protons.

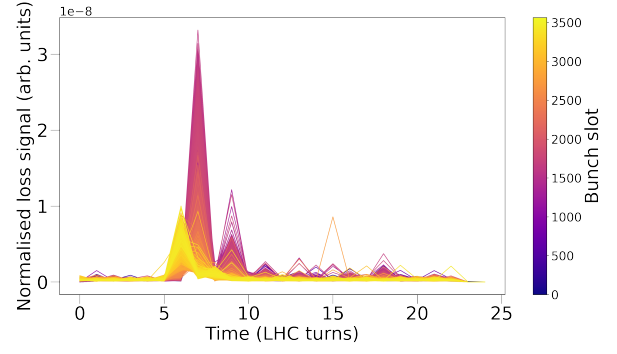


(r) Device 'HC.TZ76.BLMDIAMOND3.5', 10/05/2023 at 03:15:59, Beam mode: RAMP, Energy: 6665 GeV, Total intensity:  $2.48 \times 10^{14}$  protons.

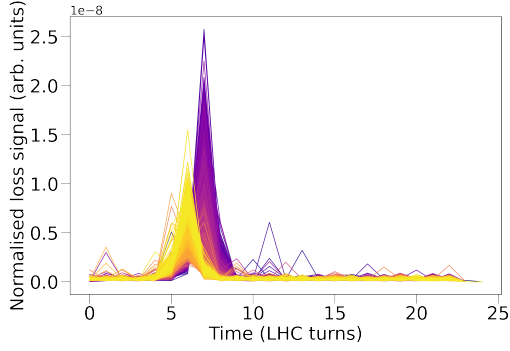
**Figure 11:** dBLM trigger events identified by the filtering and manually labelled as UFOs. Normalised losses are plotted over LHC turns for each filled bunch. The bunch slot number is shown by the colour bar. (Part 3)



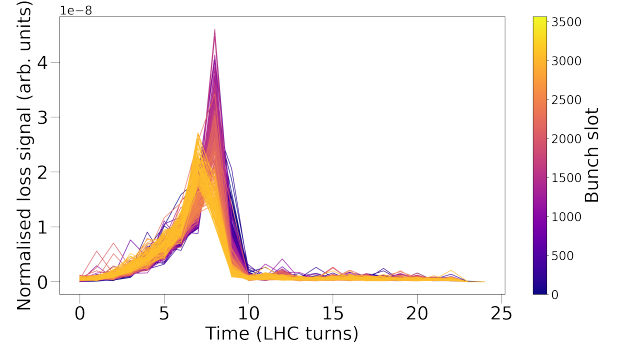
(s) Device 'HC.TZ76.BLMDIAMOND3.5', 28/04/2023 at 17:55:42, Beam mode: RAMP, Energy: 4515 GeV, Total intensity:  $6.06 \times 10^{13}$  protons.



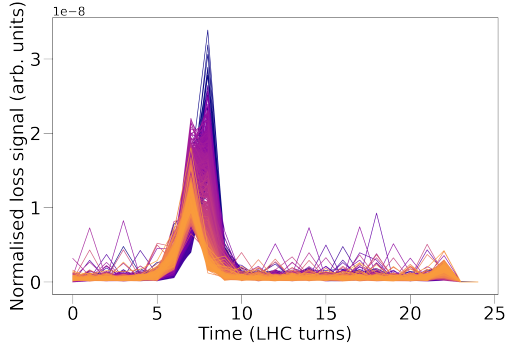
(t) Device 'HC.TZ76.BLMDIAMOND3.5', 11/07/2023 at 06:55:48, Beam mode: RAMP, Energy: 5914 GeV, Total intensity:  $3.82 \times 10^{14}$  protons.



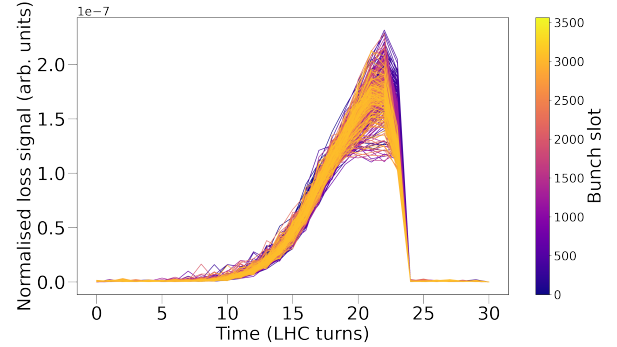
(u) Device 'HC.TZ76.BLMDIAMOND3.5', 11/07/2023 at 02:55:22, Beam mode: RAMP, Energy: 5421 GeV, Total intensity:  $3.89 \times 10^{14}$  protons.



(v) Device 'HC.TZ76.BLMDIAMOND3.5', 27/04/2023 at 22:46:01, Beam mode: STABLE, Energy: 6800 GeV, Total intensity:  $5.18 \times 10^{13}$  protons.

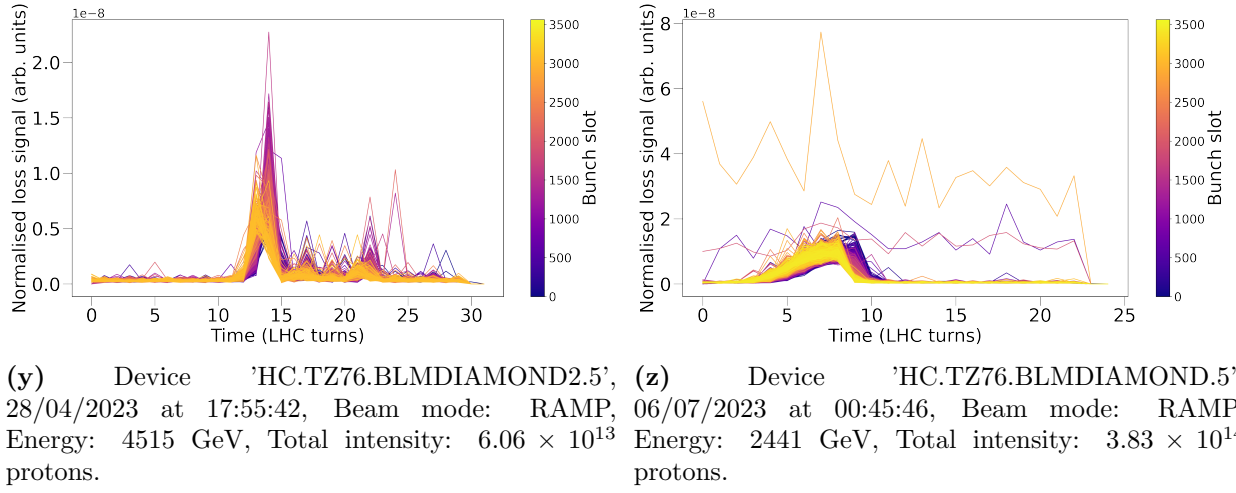


(w) Device 'HC.TZ76.BLMDIAMOND3.5', 03/06/2023 at 23:24:25, Beam mode: RAMP, Energy: 6662 GeV, Total intensity:  $2.81 \times 10^{14}$  protons.



(x) Device 'HC.TZ76.BLMDIAMOND2.3', 28/04/2023 at 23:14:06, Beam mode: RAMP, Energy: 4920 GeV, Total intensity:  $2.59 \times 10^{13}$  protons.

**Figure 11:** dBLM trigger events identified by the filtering and manually labelled as UFOs. Normalised losses are plotted over LHC turns for each filled bunch slot. The bunch slot number is shown by the colour bar. (Part 4)



**Figure 11:** dBLM trigger events identified by the filtering and manually labelled as UFOs. Normalised losses are plotted over LHC turns for each filled bunch slot. The bunch slot number is shown by the colour bar. (Part 5)