View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by OAR@UM

AUTOMATIC BEAM LOSS THRESHOLD SELECTION FOR LHC COLLIMATOR ALIGNMENT

G. Azzopardi*[1,2], G. Valentino[1], B. Salvachua[2], S. Redaelli[2], A. Muscat[1]
[1] University of Malta, Msida, Malta
[2] CERN, Geneva, Switzerland

Abstract

The collimation system used in the Large Hadron Collider at CERN is positioned around the beam with a hierarchy that protects sensitive equipment from unavoidable beam losses. The collimator settings are determined using a beam-based alignment technique, where collimator jaws are moved towards the beam until the beam losses exceed a predefined threshold. This threshold needs to be updated dynamically, corresponding to the changes in the beam losses. The current method for aligning collimators is semi-automated requiring a collimation expert to monitor the loss signals and continuously select and update the threshold accordingly. The human element in this procedure is a major bottleneck for speeding up the alignment. This paper therefore proposes a method to fully automate this threshold selection. A data set was formed from previous alignment campaigns and analysed to define an algorithm that produced results consistent with the user selections. In over 90% of the cases the difference between the two was negligible and the algorithm presented in this study was used for collimator alignments throughout 2018.

INTRODUCTION

The CERN Large Hadron Collider (LHC) is the world's largest particle accelerator. It accelerates and collides two counter-rotating beams, each having a nominal energy of 6.5 TeV during Run 2 [1]. The LHC is susceptible to beam losses from normal and abnormal conditions [2, 3]. Such beam losses are handled by a robust collimation system to safely dispose of the losses in the collimation regions.

The collimation system makes use of 100 collimators in the LHC, able to provide a cleaning efficiency of 99.998% of all halo particles [4]. A collimator is made up of two parallel absorbing blocks, referred to as *left* and *right* jaws, which are positioned symmetrically around the beam.

The position of each collimator's left and right jaw respect a hierarchy, with the settings determined following a beam-based alignment (BBA). This procedure moves collimator jaws separately towards the beam halo, whilst monitoring the measured beam loss signal. A collimator is said to be aligned when both jaws are centred around the beam after touching the beam halo. At present, collimator jaws automatically move towards the beam until the losses exceed a threshold selected by the collimation expert. Once the jaws stop moving, the expert must determine whether the collimator is aligned or not, and update the threshold accordingly. This provides a semi-automatic approach which requires collimation experts to oversee and control the entire alignment campaign.

Collimators are aligned each year during commissioning, to ensure the correct setup for the LHC to achieve nominal operation. They are aligned for different machine states; at injection (450 GeV) 79 collimators are aligned, and at flat top (6.5 TeV) 75 collimators are aligned. The collimator settings are monitored along the year as the beam orbit may shift over time [5], thus potentially requiring the collimators to be realigned. Moreover, different collimator setups are required when machine parameters are changed.

The frequency of collimator alignment campaigns motivated the development of an automatic method, to allow for collimator alignments to be performed more efficiently and at regular intervals. Automating the alignment procedure requires replacing each of the user tasks with dedicated algorithms. This paper proposes to automate one of these tasks by automatically selecting the threshold for stopping the movement of the jaws based on real-time beam loss data. This paper is structured by first presenting research on threshold usage in time-series data and looking at the initial attempt at automatic threshold selection in LHC collimation. This is followed by introducing the newly designed algorithm and finally comparing the results.

BACKGROUND

A Beam Loss Monitoring (BLM) device is associated with each collimator to detect the beam losses generated when halo particles impact the collimator jaws. This BLM detector is positioned outside the beam vacuum, immediately downstream, as shown in Figure 1. Such particle losses are proportional to the amount of beam intercepted by the collimator jaws, which are in units of Gy/s.

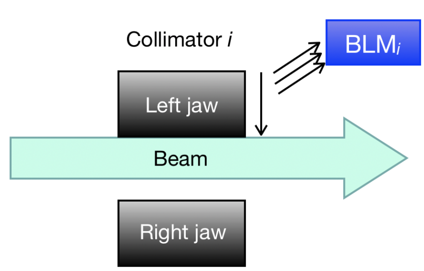


Figure 1: The jaws of collimator *i* around the beam, with its left jaw scraping the beam halo and the showers are detected by the corresponding BLM detector downstream, from [6].

* gabriella.azzopardi@cern.ch

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

Semi-Automatic BBA Threshold

Collimators are aligned using a beam-based alignment procedure established in [7], which has been used in the LHC since the start-up in 2010 [8]. A collimator is considered aligned when a jaw movement towards the beam produces a clear loss spike in the assigned BLM detector.

The beam-based alignment is currently performed semi-automatically whereby a collimation expert must make decisions as the alignment progresses. This requires the user to select a collimator (i) and select a threshold based on the current BLM signal (S_i^{Thres}), such that the collimator jaw(s) will automatically move towards the beam until the BLM losses exceed the selected threshold. The thresholds are selected as low as possible to avoid cutting into the core of the beam and to perform all collimator alignments in the same fill.

Aligning a collimator starts off by moving both jaws towards the beam simultaneously, to save time in cases where the jaws are far out. Following this, the collimator jaws must be aligned one at a time to be able to associate the BLM losses with the particular jaw. Therefore the left jaw is first aligned until an alignment spike is observed, followed by the right jaw. This is repeated for each jaw to obtain a second alignment spike, to ensure that the collimator has indeed touched the beam. This process is depicted in the state machine in Figure 2, whereby the threshold selection process must be applied before each state.

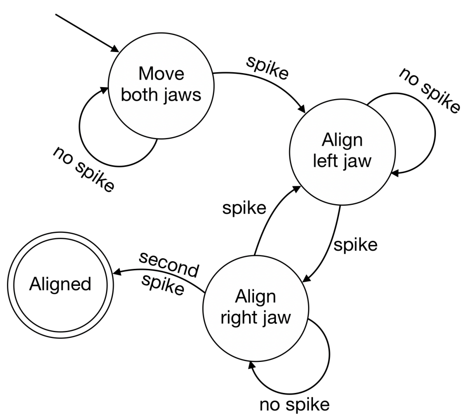


Figure 2: State machine of the jaw movements performed to align a single collimator.

The BLM losses differ across collimators as well as between fills, therefore the threshold must be selected accordingly in real-time. In general, the ideal threshold must be:

- High enough to ignore any noise spikes, to directly touch the beam without having any interruptions during the movement towards the beam.
- Low enough to immediately stop the jaws and generate minimal losses when the collimator actually touches the beam.

Fully-automating the BBA, requires automatically selecting the BLM threshold, by finding a compromise between the two requirements.

RELATED WORK

Automating the selection of the threshold for the LHC collimators was previously studied in [9]. This was explored alongside electromyography (EMG), as this commonly uses thresholding to provide feedback on the recorded signal.

Thresholding in Electromyography

Electromyography is a technique used to evaluate and record electrical activity produced by skeletal muscles. When a muscle contracts, electrical activity is generated within the muscle and this is recorded using electrodes placed on the surface of the skin [10]. An important step when working with EMG signals is preprocessing to extract the relevant information. One of the most common signal attributes extracted from the time-domain is the amplitude. This is done by calculating the Root Mean Square (RMS), as can be seen in a number of papers [11–13]. Thresholding also plays vital roles in EMG studies:

- Subject Training - Threshold markers can be used in a number of ways to accurately perceive how tense or active a patient's muscle really is. This is required when patients have lost touch with whether a muscle is tensed or not due to problems with muscle-oriented pain and tension. In such cases a threshold is required as a target level during an EMG activity, whereby the patient tries to reach various thresholds multiple times [10].
- Seizure Detection - The unpredictability of seizure occurrence is distressing therefore it is suggested to wear seizure detection EMG devices to prevent unexpected deaths. Research presented in [14] implements an algorithm into a wearable device whereby if the number of zero-crossings exceed a predefined threshold, then the alarm is triggered. The device is able to give real-time seizure alarms with a sensitivity of 93.8%.

Similar to EMG data, BLM signals are noisy time series data. Therefore the analysis in this paper includes averaging using RMS, as this methodology is commonly adopted in time-series data analysis.

Initial Automatic Threshold Selection

When selecting the threshold at time t = 0s the most recent values ($\mathrm{BLM}_{t=-5s}$ to $\mathrm{BLM}_{t=-0s}$) are given the most importance as these would indicate the latest level of losses. At the same time, the previous values till $\mathrm{BLM}_{t=-19s}$ must also be considered as these indicate the previous situation of the losses, in case the previous spike would still be decaying to its steady-state value. This can be expressed mathematically using an exponentially weighted moving average (EWMA) over a 20 second window:

$$\mathrm{EWMA}_{\mathrm{BLM}} = \frac{\sum_{i=1}^{20} e^{i} \times \mathrm{BLM}_{t=i-20}}{\sum_{i=1}^{20} e^{i}} \quad (1)$$

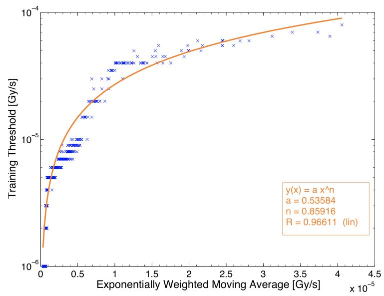
In total 475 samples were gathered, each consisting of a 20 second window of 1 Hz BLM data, and the corresponding

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

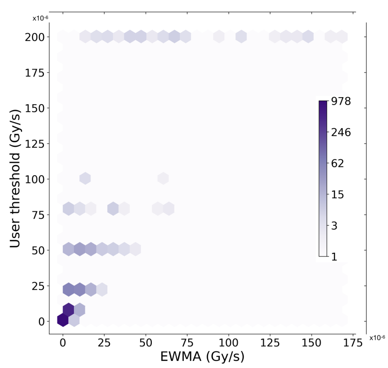
threshold selected by the user. A power function was fitted to the user thresholds as a function of the EWMA, as shown in Figure 3a, generating a correlation coefficient of 0.96611. Based on this fit, the threshold was automatically calculated using:

$$S_i^{\mathrm{Thres}} = 0.54 \times (\mathrm{EWMA}_{\mathrm{BLM}})^{0.86} \tag{2}$$

The data used was based on semi-automatic alignments performed in 2011 at 3.5 TeV. Since then, the energy has doubled and the data is now acquired and logged at a frequency of 100 Hz. As a result, the EWMA and user thresholds are no longer correlated, as shown in Figure 3b, therefore this algorithm must be revised.



(a) 2011 user selected threshold against EWMA, from [9]. A power fit can be applied to the data.



(b) 2016 user selected threshold against EWMA.

Figure 3: Loss thresholds applied before the start of jaw movements in a) 2011 and b) 2016, as a function of the exponentially weighted moving average of the BLM signal.

Moreover, the waiting time between one alignment and another was fixed at ~10 seconds, ideally this should be decreased as much as possible to increase efficiency. Finally, this threshold selection algorithm was implemented at the user application level, whereas moving it to the server level would allow for faster execution.

IMPLEMENTATION

The automatic threshold selection algorithm must attempt to mimic the BLM signal analysis made by the collimation expert when selecting the threshold, namely:

- Give higher importance to the most recent BLM values
- Partially factor out high spikes
- Assume long decays will eventually reach the steady state level
- Introduce a gap above the steady state high enough to allow for alignment spikes

The application used for the semi-automatic BBA was designed to provide a discrete list of thresholds. These thresholds were developed over time following experience from manual alignments, and are available for the collimation expert to select the one which corresponds best to the current BLM signal. These thresholds are kept also for the automation proposed in this paper, to better compare the results and validate the tool.

Data Analysis

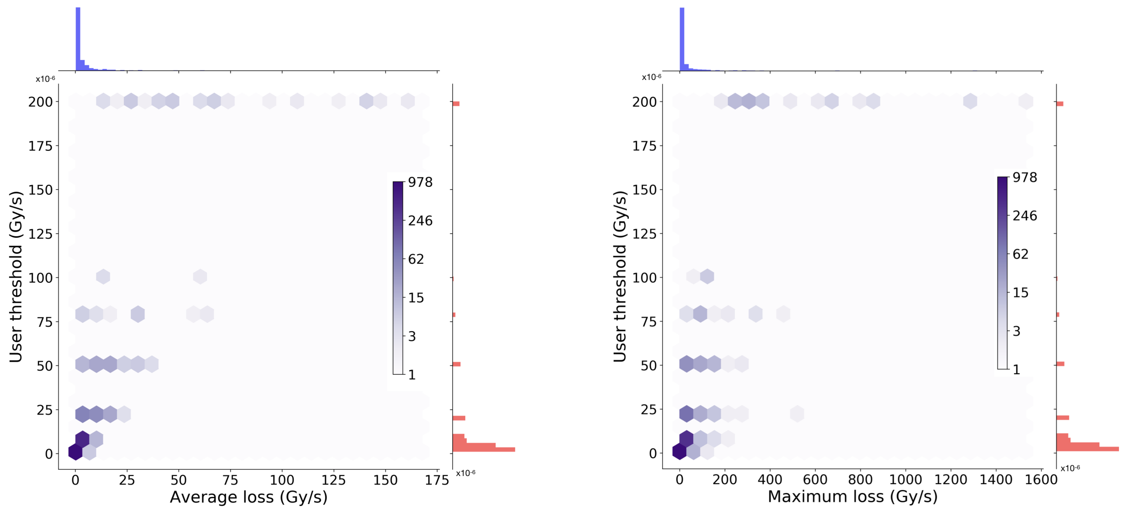
Alignments in 2016 were done using the semi-automatic alignment tool requiring experts to manually select the thresholds at the start of each jaw movement towards the beam. This provided a data set of 1778 samples at injection and flat top, which were studied to find an automatic technique for selecting a reasonable threshold. The samples consist of 7.5 second windows of BLM signals, as the maximum decay time is 6 seconds at flat top [15], and each sample was studied at 25 Hz, as this is what is available at the time of threshold selection. In addition, the thresholds selected by the user at the time were also extracted.

To determine whether there is a pattern between the BLM losses and the user selected thresholds, the thresholds are analysed against the maximum and average losses in Figure 4. The majority of the samples required thresholds below 2.5×10^{-5} Gy/s, and as expected, the larger thresholds are selected both when the average loss is larger as well as the maximum loss. Therefore RMS-smoothing will be applied to be immune to outliers, being a verified approach in time-series data. In order to assign different priorities to the data depending on their occurrence in time, exponential weights can be combined with the RMS by applying EWMRMS.

New Automatic Threshold Selection Algorithm

The threshold selection algorithm was implemented at the server level [16] and is shown in Algorithm 1. This algorithm is provided with the latest BLM data of length 188 (7.5 s at 25 Hz) to apply an EWMRMS on windows of fixed length (*length = 50*). This is used to automatically select a reasonable threshold (*auto threshold*) from a discrete list. The first step is to create an array of length 188, of evenly spaced numbers between -1 and 0 and use them as power coefficients to the exponential function (*weights[size=188]*). A window for each BLM data element i is taken with the element itself and the 49 elements (or less) which precede it (*window[i-50, i]*). The RMS is then applied to the window and is multiplied by the exponential factor corresponding to the data element (*weights[i]*), using Equation 3.

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.



(a) User selected threshold against the average BLM losses.

(b) User selected threshold against the maximum BLM losses.

Figure 4: The threshold selected by the user against the (a) average BLM loss, (b) maximum BLM loss, 7.5 seconds before the user selected the threshold. Marginal histograms are included to show the distribution of each measurement.

$$\text{EWMRMS} = \text{weights[i]} \sqrt{\frac{\sum_{j=1}^{\text{length}} \text{window[j]}^2}{|\text{window}|}} \tag{3}$$

Once the EWMRMS is calculated for all data elements, the maximum EWMRMS value is used to select the first threshold higher. If the threshold selected is also the first threshold above the mean value of the data, then the next highest threshold is selected. Finally, if the selected threshold is less than or equal to the threshold selected for the preceding movement of the same collimator, and if that movement did not generate an alignment spike, then the next highest threshold is selected.

RESULTS AND VALIDATION

The threshold algorithm was tested on the data set gathered. Figure 5 shows two BLM signals and the thresholds selected by the user (*user threshold*) and the algorithm (*auto threshold*), based on the BLM signal, for the next alignment step. Figure 5a, displays an alignment spike 7 seconds before the start of the alignment, and one can see that the spike completely decayed and the signal reached steady state towards the end of the signal. Therefore in this case the spike is not important and one would simply aim for a suitable gap between the steady state and the threshold. This is the approach taken by both the user and the algorithm, such that the same threshold was selected. On the other hand, Figure 5b shows an alignment spike towards the end of the signal. In this case the spike must be considered and the steady state before the spike is ignored as now (and possibly in the future) the steady-state losses are higher. Therefore the aim would be to create a suitable gap between the new steady state and the threshold, and partially factor in the spike. Once again both the user and the algorithm selected suitable thresholds, such that they are high enough to allow for clear spikes, whilst at the same time conservative of future losses.

```
int window_size = 50;
double T[] = threshold options;
double weights[] = power exponentials;

//INPUT: previous_threshold (T[prev_t])
mean = data_sum / data_length;
for i = 0; i <data_length; i++ do
    start_position = i - window_size;
    if start_position <0 then
        start_position = 0;
    end
    window = data[start_position, i];
    ewmrms = Calculate EWMRMS using
      Equation 3;
end
auto_threshold = T[t] >max(ewmrms);
if auto_threshold >mean and T[t-1] <mean then
    auto_threshold = T[t+1];
end
if t <= prev_t then
    auto_threshold = T[prev_t+1];
end
```

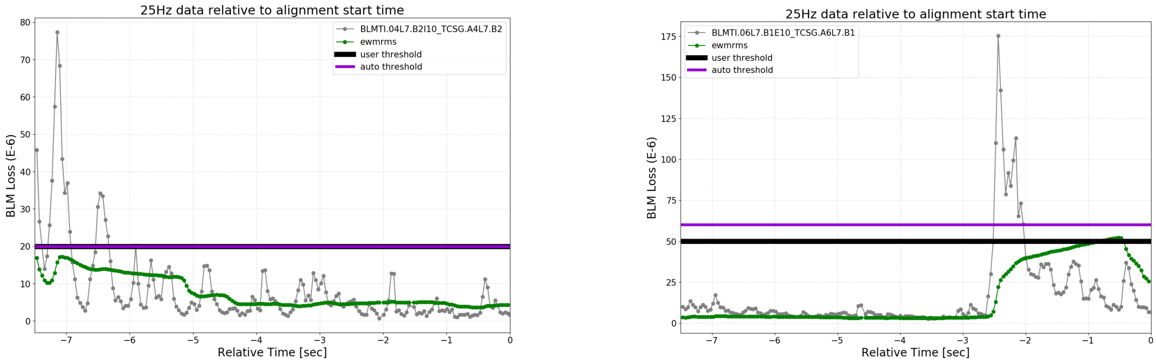
Algorithm 1: Automatic threshold selection algorithm.

The overall performance of the algorithm is determined by comparing the selected thresholds to the ones selected by the user. Figure 6a shows the difference between the two thresholds against the thresholds selected by the user. The difference is negligible for 90% of the cases, and the large differences occur when dealing with larger thresholds at 2×10^{-4} Gy/s, thus also making them suitable selections.

Finally, the thresholds selected by the new algorithm were compared to those selected by the old algorithm in Figure 6b, by comparing them to the user selected thresholds. These

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

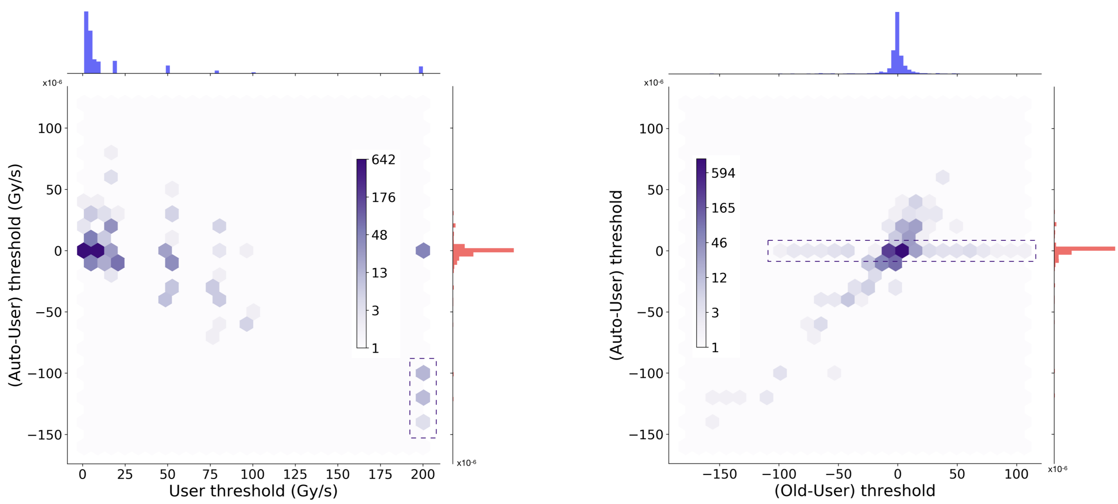
Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.



(a) BLM signal starting with a spike

(b) BLM signal ending with a spike

Figure 5: Two examples of 25 Hz BLM signals with the thresholds selected by the user and algorithm.



(a) The difference between the thresholds selected by the new automatic algorithm and the user, against the threshold selected by the user. The dotted rectangle highlights that the larger differences occur only when a high threshold is necessary.

(b) The new automatic algorithm selections against the old algorithm selections, both compared to the user selections. The dotted rectangle highlights that the new algorithm corresponds better to the user.

Figure 6: The performance of the new automatic algorithm compared to (a) the user threshold, (b) the old automatic algorithm, including marginal histograms to show the distribution of each measure.

results indicate that the new algorithm selected thresholds that corresponded better to those selected by the user.

The threshold algorithm was incorporated into the beam-based alignment software together with automatic spike detection [17], to fully-automate the alignment. This new software was successfully used during commissioning 2018 [18], and all other alignment campaigns required throughout the year [19]. As a result, the positive results of this new software validate the new algorithm designed for automatically selecting the threshold.

CONCLUSION

The sensitive equipment in the LHC is protected by the collimation system, which consists of 100 collimators precisely aligned around the beam. Collimator alignments are currently performed semi-automatically whereby collimators automatically move towards the beam until the BLM losses exceed the threshold predefined by the expert. Once the threshold is exceeded, the expert must analyse the losses to update the threshold as required, until the collimator is aligned. This paper proposes to automate the process of threshold selection, as a necessary step to fully-automate the entire alignment procedure.

Previous work to automate the BBA threshold and thresholding in time-series data were studied to define a new algorithm based on more recent LHC data. The presented algorithm is based on data from alignments performed in 2016, and the results obtained were compared to the user selected thresholds at the time. Overall, the results indicate that the thresholds selected automatically were consistent with the thresholds selected by the users, and show an improvement when compared to the thresholds selected using the automatic algorithm presented in previous work. These promising results endorse using this algorithm to fully-automate the alignment, and in fact the full-automation was successfully used throughout 2018, thus fully validating this new algorithm.

REFERENCES

[1] L. Evans, "The Large Hadron Collider", in *New J. Phys.*, vol. 9, no. 9, art. no. 335 (22 pages), 2007. doi:10.1088/1367-2630/9/9/335

[2] R. W. Assmann *et al.*, "Requirements for the LHC Collimation System", in *Proc. 8th European Particle Accelerator Conf. (EPAC'02)*, Paris, France, Jun. 2002, paper TUAGB001, pp. 197–199.

[3] C. Bracco, "Commissioning Scenarios and Tests for the LHC Collimation system", Ph.D. thesis, Ecole Polytechnique Federale de Lausanne, 2009.

[4] G. Valentino, "Fast Automatic Beam-Based Alignment of the LHC Collimator Jaws", Ph.D. thesis, University of Malta, Malta, 2013.

[5] R. J. Steinhagen, "LHC Beam Stability and Feedback Control-Orbit and Energy", Ph.D. thesis, RWTH Aachen U., 2007.

[6] G. Azzopardi, *et al.*, "Data-driven cross-talk modeling of beam losses in LHC collimators", *Physical Review Accelerators and Beams*, vol. 22 no. 8, p. 083002, 2019. doi:10.1103/PhysRevAccelBeams.22.083002

[7] R. W. Assmann *et al.*, "Expected Performance and Beam-based Optimization of the LHC Collimation System", in *Proc. 9th European Particle Accelerator Conf. (EPAC'04)*, Lucerne, Switzerland, Jul. 2004, paper WEPLT006, pp. 1825–1827.

[8] D. Wollmann *et al.*, "First Cleaning with LHC Collimators", in *Proc. 1st Int. Particle Accelerator Conf. (IPAC'10)*, Kyoto, Japan, May 2010, paper TUOAMH01, pp. 1237–1239.

[9] G. Valentino, *et al.*, "Automatic threshold selection for BLM signals during LHC collimator beam-based alignment", *2012 Sixth UKSim/AMSS European Symposium on Computer Modeling and Simulation*, IEEE, 2012. doi:10.1109/EMS.2012.16

[10] R. Merletti, P. A. Parker, and P. J. Parker, "Electromyography: physiology, engineering, and non-invasive applications", vol. 11, John Wiley & Sons, 2004.

[11] M. Point, *et al.*, "Cryotherapy induces an increase in muscle stiffness", *Scandinavian journal of medicine & science in sports*, vol. 28, no. 1, pp. 260–266, 2018. doi:10.1111/sms.12872

[12] R. M. Erskine, *et al.*, "The individual and combined effects of obesity- and ageing- induced systemic inflammation on human skeletal muscle properties", *International Journal of Obesity*, vol. 41, no. 1, pp. 102–111, 2017. doi:10.1038/ijo.2016.151

[13] C. M. Smith, *et al.*, "Differences between the Time Course of Changes in Neuromuscular Responses and Pretest versus Posttest Measurements for the Examination of Fatigue", *Journal of Nature and Science (JNSCI)*, vol. 3, no. 10, 2017.

[14] S. Beniczky, "Automated real-time detection of tonic-clonic seizures using a wearable EMG device", *Neurology,* vol. 90, no. 5, pp. 428–434, 2018. doi:10.1212/WNL.0000000000004893

[15] G. Valentino, *et al.*, "Beam diffusion measurements using collimator scans in the LHC", *Physical Review Special Topics-Accelerators and Beams*, vol. 16, no. 2, p. 021003, 2013. doi:10.1103/PhysRevSTAB.16.021003

[16] G. Azzopardi, *et al.*, "Software Architecture for Automatic LHC Collimator Alignment using Machine Learning", presented at ICALEPCS'19, New York, USA, Oct. 2019, paper MOCPL04, this conference.

[17] G. Azzopardi, *et al.*, "Automatic spike detection in beam loss signals for LHC collimator alignment", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment,* vol. 934, pp. 10-18, 2019. doi:10.1016/j.nima.2019.04.057

[18] G. Azzopardi, A. Muscat, S. Redaelli, B. Salvachua, and G. Valentino, "Operational Results of LHC Collimator Alignment Using Machine Learning", in *Proc. 10th Int. Particle Accelerator Conf. (IPAC'19)*, Melbourne, Australia, May 2019, pp. 1208–1211. doi:10.18429/JACoW-IPAC2019-TUZZPLM1

[19] G. Azzopardi, *et al.*, "Operational Results on the Fully-Automatic LHC Collimator Alignment", in *Physical Review Special Topics-Accelerators and Beams*, vol. 22, no. 9, p. 093001, 2019. doi:10.1103/PhysRevAccelBeams.22.093001

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.