Beam Loss Threshold Selection for Automatic LHC Collimator Alignment

G. Azzopardi [1,2], G. Valentino [2], B. Salvachua [1], S. Redaelli [1], A. Muscat [2]

[1] CERN, Geneva Switzerland, [2] University of Malta, Malta

Introduction

The Large Hadron Collider (LHC) at CERN relies on **100 collimators** to protect its sensitive equipment from unavoidable beam losses. Each collimator is made up of two jaws (*left* and *right*).

Collimator positions are determined following a **beam-based alignment (BBA)**:

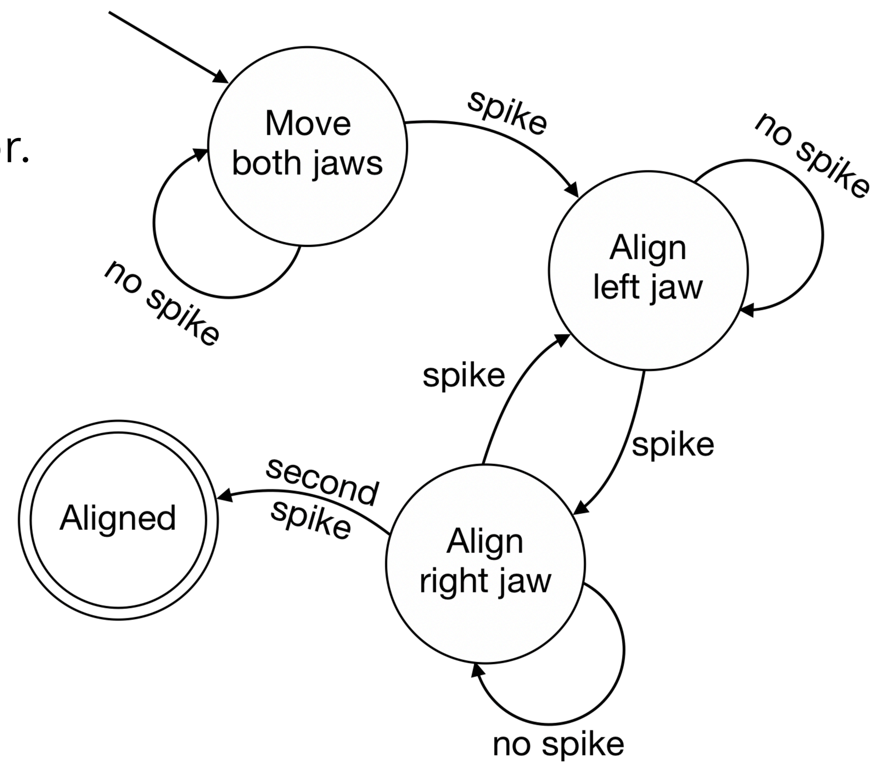
- Each collimator is assigned a **Beam Loss Monitoring (BLM)** detector.
- The alignment is done with a step precision of 5 μm.
- Calculate beam position and beam size at each collimator.

Beam-based alignment procedure:

- **Jaws aligned separately** to monitor the BLM signal.
- A collimator is considered aligned when a signature spike pattern is detected in the BLM losses [1].

Threshold selection:

- BLM losses change across collimators therefore suitable thresholds must be selected in real-time.
- A **selection is made before starting the alignment/ movement** of any jaw(s).



State machine of the jaw movements performed to align a single collimator. A threshold must be selected before each state.

The threshold must be selected **high enough such that any noise spikes are ignored**, thus avoiding any unnecessary pauses during the jaws movement, and **low enough to immediately stop the jaws** and minimise losses.

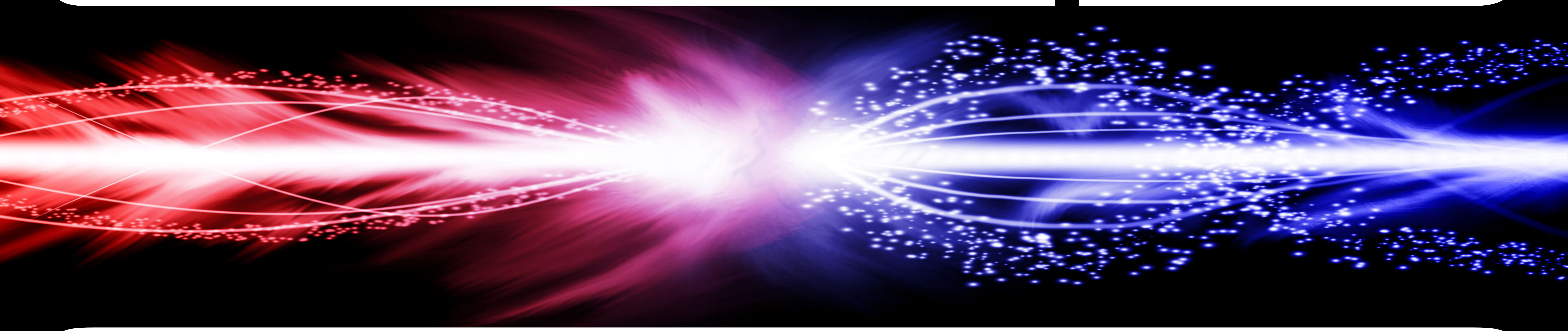
Threshold Selection Algorithm

An algorithm was designed to automate the selection based on the Exponentially Weighted Moving RMS:

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

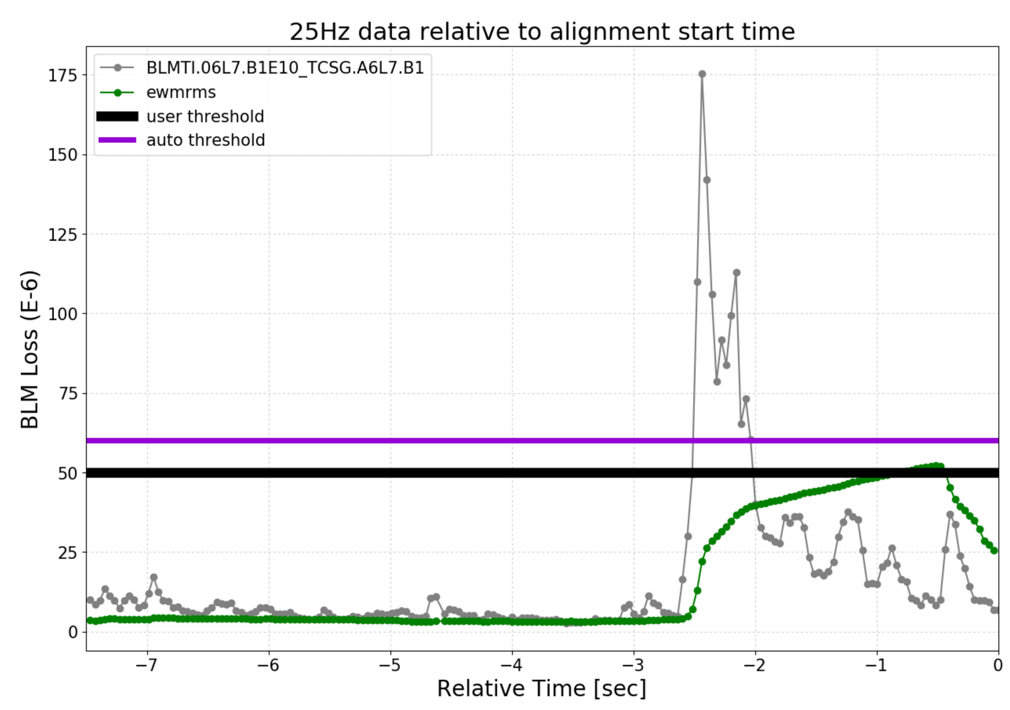
```
int window_size = 50;
double T[] = threshold options;
double weights[] = power exponentials;

//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;
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<=t_prev then
    auto_threshold = T[prev_t+1];
end
```

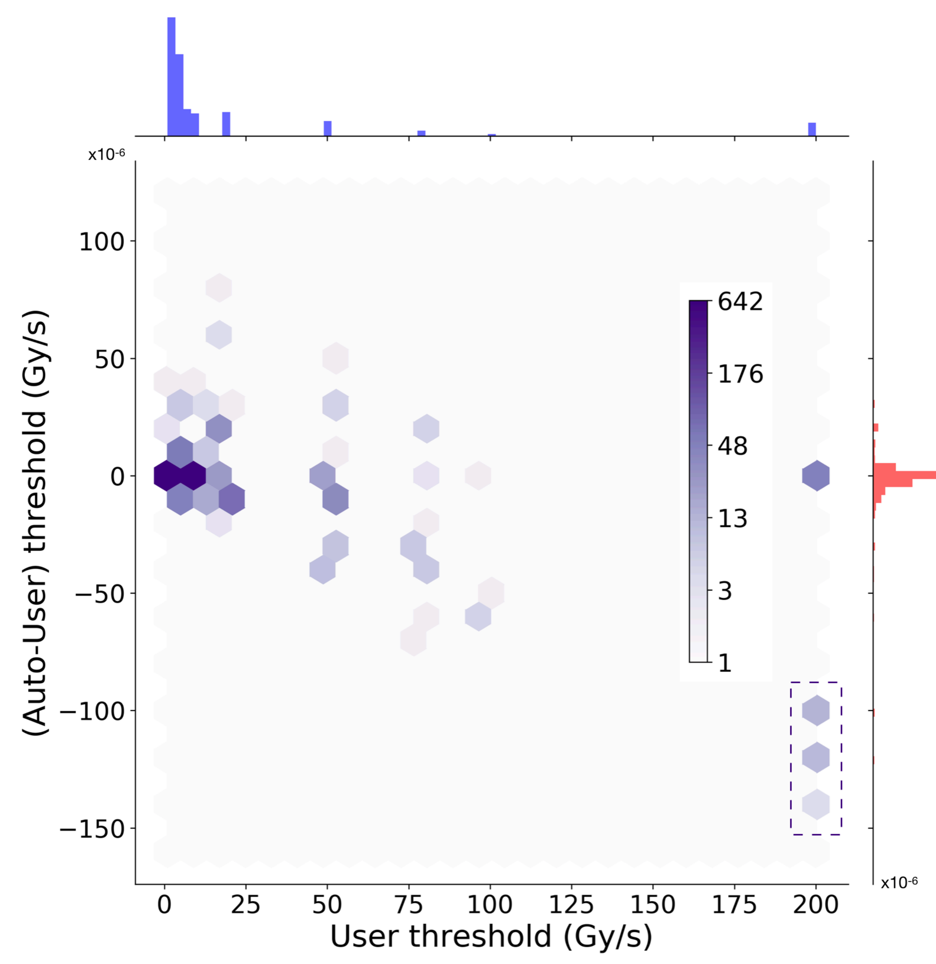


Results Overview

- A data set of **1778 samples** was collated from alignments in 2016, to analyse and test the new automatic algorithm.
- The **automatically selected thresholds were compared to the thresholds selected by the users** in 2016.
- The difference between the two sets of selections was negligible for 90% of the cases.
- The large differences occur when dealing with larger thresholds at 2 x10^{-4} Gy/s, thus also making them suitable selections.
- **ALL thresholds selected automatically were suitable selections**.



Example of 25 Hz BLM signal with the thresholds selected by the user and new automatic algorithm.



The difference between the thresholds selected by the algorithm and the user, against the thresholds selected by the user.

Conclusion

The automatic threshold selection algorithm was incorporated into the beam-based alignment software to fully-automate the alignment. This was successfully used during commissioning 2018 [2], and all other alignment campaigns required throughout the year. As a result, the positive results of this new software validate the new algorithm designed for automatically selecting the threshold.

References

[1] G. Azzopardi et al., "Automatic Spike Detection in Beam Loss Signals for LHC Collimator Alignment," in *NIM-A*, 2019.

[2] G. Azzopardi et al., "Operational Results of LHC Collimator Alignment using Machine Learning," in *Proceedings of IPAC'19*.

gabriella.azzopardi@cern.ch