UPGRADED CONTROL SYSTEM FOR LHC BEAM-BASED COLLIMATOR ALIGNMENT

G. Valentino*, G. Baud, M. Gasior, S. Jackson, L. Jensen, J. Olexa, S. Redaelli, J. Wenninger
CERN, Geneva, Switzerland

Abstract

In the Large Hadron Collider (LHC), over 100 movable collimators are connected to a three-tier control system which moves them to the required settings throughout the operational cycle from injection to collision energy. A dedicated control system was developed to align the collimators to the beam during machine commissioning periods and hence determine operational settings for the active run. During Long Shutdown 1, the control system was upgraded to allow beam-based alignments to be performed using embedded beam position monitors in 18 newly installed collimators as well as beam loss monitors. This paper presents the new collimation controls architecture for LHC Run II along with several modifications in the Java-based application layer.

INTRODUCTION

The Large Hadron Collider (LHC) accelerates two counter-rotating beams to an energy of 7 TeV before colliding them in four points where experiment detectors are located [1]. Uncontrolled beam losses can result in quenches of the super-conducting magnets, which can damage the machine. For this reason, a beam collimation system is installed in the LHC, comprising over 100 collimators [2]. It is also designed to protect the LHC in the event of fast failures, which might damage accelerator components.

Each collimator installed for ring cleaning consists of two jaws, made of graphite, tungsten or copper, which have to be positioned around the beam at all times with an accuracy of less than 50 μm. The operational settings of the collimators are defined in order to establish a four-stage hierarchy. They are determined via a beam-based alignment procedure [3] which uses feedback from a Beam Loss Monitoring (BLM) [4] detector positioned downstream from the collimator. From the aligned jaw positions, the measured beam centers and beam sizes can be calculated.

The primary collimators (TCP) are positioned closest to the beam, followed by the secondary collimators (TCSG), tertiary collimators (TCT) and absorbers (TCLA). Most of the collimators are installed in Insertion Region (IR) 3 and 7 for off-momentum and betatron cleaning respectively. The TCTs are installed upstream of the experiment detectors (IR1, 2, 5 and 8), and a TCSG is installed in the dump region (IR6).

The software architecture for the LHC collimation system respects the standards of the LHC Software Architecture (LSA) [5], which consists of a 3-tier structure. For the collimators, the bottom layer is composed of a double PXI system that is used to control and read out stepping motors, sensors and measurement devices. The collimator jaw positions can be positioned with an accuracy of 5 μm, i.e. less than 2% of the 1σ beam size at the primary collimators at 7 TeV. The maximum jaw movement rate is 2 mm/s.

Application servers that host databases and operational files make up the middle layer, on top of which Graphical User Interface (GUI) console applications run. The collimator positions can be set from a remote location in the CERN Control Centre (CCC) and synchronized with the operation of the LHC cycle. The software applications interact with the hardware via the Common Middleware (CMW) [6] and Front-End Software Architecture (FESA) [7] infrastructures.

During Long Shutdown 1 (LS1), all 16 TCTs and the 2 TCSGs in IR6 were replaced by a new design, in which Beam Position Monitor (BPM) pick-ups are embedded in the upstream and downstream corner of each collimator jaw [8, 9]. The LHC installation followed a successful validation of a prototype with beam in the Super Proton Synchrotron (SPS) [10]. This required an upgrade of the beam-based alignment control system to acquire and use both the BLM and the BPM data for the alignment.

ALIGNMENT PROCEDURES

BLM-based Alignment

A four-stage procedure is used to align the collimators with feedback from the BLMs. Both jaws of a reference collimator are moved towards the beam in steps of 5-20 μm (1) until they reach the beam halo on either side, which is established when an appropriate loss pattern is observed in a downstream BLM. Then, the collimator for which the beam center and beam size need to be measured is aligned (2). The center is given as the average of the two aligned jaw positions. The reference collimator is then re-aligned (3), and the beam size at the previous collimator is given as the ratio of its gap in mm when aligned to the average of the cut in units of beam σ. The final step is to open the collimator to the new operational positions (4). This procedure is performed at very small gaps of $< 4\ \sigma$.

One of the main drawbacks of this procedure is that it causes losses of a fraction of the circulating beam and therefore can only be carried out with safe beam intensities. Collimator alignment can therefore not be performed dur-

* gianluca.valentino@cern.ch

ISBN 978-3-95450-148-9

ing standard operation. Due to the time needed to complete an alignment of the full system (∼4 hours), which needs to be repeated for different subsets of collimators at different points in the machine cycle, the system performance relies on machine and collimator setting stability and reproducibility from fill to fill.

BPM-based Alignment

The beam position between two BPM electrodes can be calculated using a well-known linear technique. In a simple 2D approximation of a BPM arrangement, which consists of a circular beam-pipe and two point-like electrodes located 180^o apart on a horizontal axis, the approximate position of a charged particle, denoted as X_{bpm}, is calculated from the distance between the opposite BPM electrodes B and the induced potential V_1 and V_2 on opposite BPM electrodes:

$$X_{bpm} = \frac{B}{4}\frac{V_1 - V_2}{V_1 + V_2} \quad (1)$$

The objective of the alignment is to minimize X_{bpm}. A successive approximation algorithm was developed to automatically align the collimator jaws around the beam axis from any starting jaw gap and beam offset [11]. The left and right jaws are moved towards and away from the beam respectively, or vice-versa, depending on the sign of the beam offset. Several iterations are needed due to nonlinearities inherent in the BPM geometry. Contrary to the BLM-based method, where the jaws touch a reference halo, this technique does not provide a measurement of the beam size at the collimator. However, the measured beam sizes are only used to calculate the operational settings at injection, as the error introduced by the β-beat tends to be more than the alignment error due to the jaw step size, while at top energy the nominal beam sizes are used as the beam sizes shrink. BPM-based alignment is more useful at top energy in terms of machine commissioning efficiency, as the majority of the machine configuration changes throughout the run take place there.

ALIGNMENT SOFTWARE ARCHITECTURE

LHC Run 1 Architecture

During Run 1, all collimators were aligned using the BLM-based technique as the embedded BPM collimators were not installed. BLM data was transmitted at 12.5 Hz from the crates in the tunnel via UDP to a Linux server, which swallowed or forwarded the packets depending on whether a Java client GUI subscribed to the data. The header of each UDP packet contained the IR and the position within the IR (left, right, center) of the BLM crate. The payload consisted of the data arranged in a 16 × 16 2D array. The data are in integer format, and were then converted to units of Gy/s. The collimator jaw positions were set and read out via a FESA class called *LHCCollimator* [12]. A feedback loop was implemented in the Java GUI, which stopped the jaw movement when the losses exceeded a pre-defined threshold in Gy/s [13].

A prototype collimator with embedded BPMs was installed in the SPS in Run 1 for beam tests. This architecture was re-used for the BPM data acquisition, which is provided by electronics based on compensated diode detectors (called DOROS) [14]. By taking measurements over thousands of turns, sub-micrometer resolution is achieved. The BPM-based alignment algorithm was implemented in the Java GUI to make it easier to modify during the beam tests.

LHC Run 2 Architecture

In LS1, the Run 1 control system was upgraded to allow both BLM-based and BPM-based collimator alignment to be performed in the same software module. The new software architecture is shown in Fig. 1. The functionality previously present in two separate Java applications was moved to a new FESA class (called *LHCCollAlign*) running on a Front-End Computer (FEC). The Run 1 UDP packet format was reused to allow the BLM crates to send the 12.5 Hz data to the FEC. *LHCCollAlign* acts as a BLM concentrator and combines the 12.5 Hz data from all 27 BLM crates for logging purposes.

The DOROS electronics was upgraded from the SPS beam tests with automatic gain control to ensure that the signals remain within a fixed range independently of the BPM aperture and beam intensity in the LHC. In addition, asymmetries between two opposite BPM pick-up channels are corrected online by means of a switching mechanism, in which the signals A and B from two opposite electrodes are connected to respective channels A and B or to channels B and A. An averaged, calibrated beam position measurement is then provided at 1 Hz by a second FESA class (*BPMCOL*), which receives the data from the DOROS boxes via UDP and send back control messages to the DOROS boxes (therefore acting as a DOROS controller). The beam positions (X_{bpm}) at each collimator are calculated based on the electrode signals and the BPM aperture. The aperture is calculated by *LHCCollAlign* from the upstream and downstream jaw positions of the 18 BPM-equipped collimators, and a constant offset which is the retraction of the BPM pick-up button with respect to the jaw surface. This is then sent to BPMCOL at 1 Hz, which is the readout rate from *LHCCollimator*. All collimators can be concurrently aligned using either of the two techniques.

The BLM-based feedback loop is always active, meaning that should high losses occur during the BPM-based alignment, when the collimator is supposed to be far from the beam, the jaw movement is stopped. The software architecture is designed to be easily extensible should further collimators be equipped with BPMs in future LHC runs. Once sufficient experience is gained with the system, the existing collimation hierarchy margins in the TCSPs and TCTPs (placed to account for beam orbit drifts) can be re-

ISBN 978-3-95450-148-9

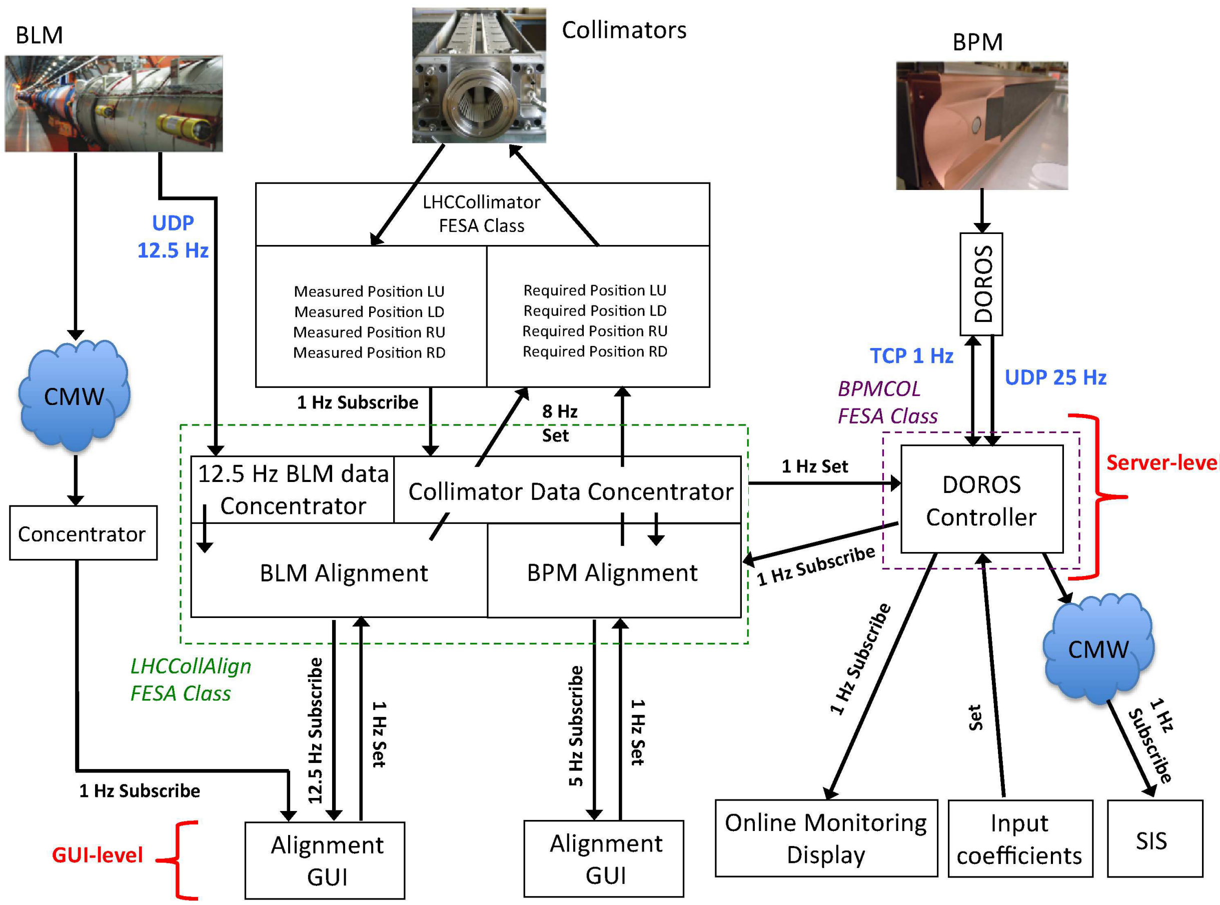

Figure 1: The software architecture for the BPM and BLM data acquisition and collimator alignment.

duced [15]. The beam position measurements would be sent to the LHC Software Interlock System [16] to dump the beam in case of orbit drifts above a certain threshold.

USER INTERFACES

Two dedicated Java GUI applications are used for the BLM-based and BPM-based alignment. The BLM alignment GUI also subscribes to both the standard 1 Hz and 12.5 Hz data for display purposes, and is identical to that used for Run 1 [17]. The user can input 4 parameters, namely the left and right jaw step size, the time interval between each step, and a stopping BLM threshold. On the other hand, the BPM alignment GUI requires a value for X_{bpm} below which the successive approximation algorithm stops, and a time interval between each step. A minimum gap value is also required to ensure that the collimator jaw does not inadvertently move too close to the beam.

An online display was developed to provide monitoring of the beam orbit in all embedded BPM collimators. A screenshot of the GUI for one collimator is shown in Fig. 2. The upstream and downstream beam positions relative to the collimator center are represented by circles. The half-width of the box represents 1 σ, and the intermediate lines indicate an editable threshold in mm, which is then converted to units of σ. The circles turn red if they exceed this

Figure 2: Subview of the monitoring display showing the up and downstream beam positions at one collimator.

limit. The non-linearities inherent in the BPMs can be corrected for by performing a 2D polynomial fit to a series of measured beam positions at different collimator gaps and offsets [11]. The fit coefficients can be sent to *BPMCOL* via a dedicated GUI.

ALIGNMENT RESULTS

The software architecture was tested during a SPS beam test and in the LHC during the beam commissioning period at the start of Run 2. An example of a BPM-based collimator alignment is shown in Fig. 3. A total of 12 iterations were needed to complete the alignment. Initially, both jaw corners are moved in parallel until the upstream electrode signals are equalized. Then, a tilt is gradually introduced in the jaws until the downstream electrode signals are also equalized. All collimators were aligned simultaneously in 15-25 seconds, which is a remarkable speed-up over the ~1.5 hours required to align the same collimators with the BLM-based technique.

ISBN 978-3-95450-148-9

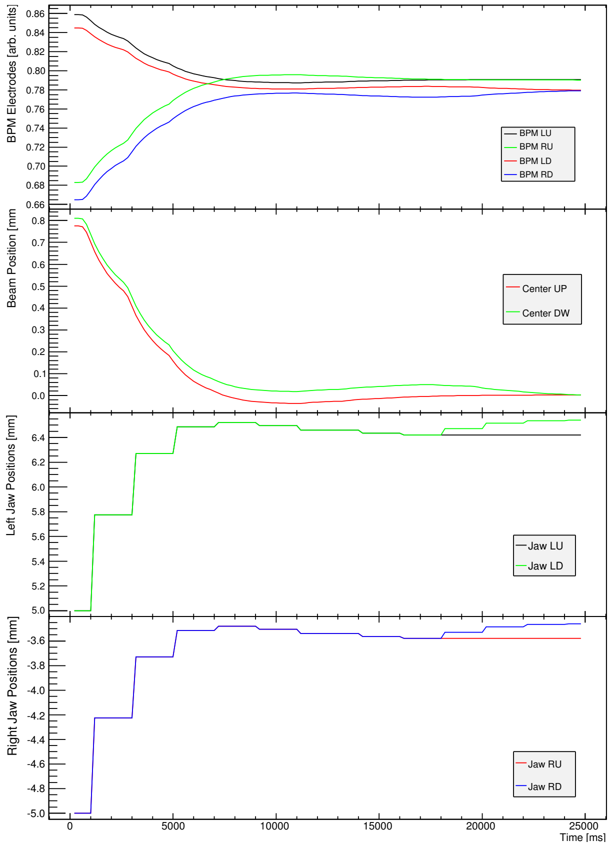

Figure 3: BPM-based collimator alignment.

CONCLUSION

Beam-based alignment of the LHC collimators is used to determine the jaw settings needed for operation. During Run 1, all collimators were aligned with a beam loss feedback algorithm, which ensured that the jaws stopped moving when the beam halo was reached. This was performed using a Java GUI application. The replacement of 20% of the system with embedded BPM collimators required a controls software upgrade. A new middleware layer was developed in FESA to ensure that the BLM and BPM data acquisition could be performed reliably for the given real-time constraints, and the separate alignment techniques could be performed in the same software module. The next steps will involve a thorough fill-to-fill analysis of the BPM data to determine whether the present collimation hierarchy margins to account for orbit drifts can be removed, and orbit interlocks could instead be put in place. This would allow to reduce the β^* and therefore extend the luminosity reach of the LHC.

REFERENCES

[1] Report No. CERN-2004-003-V1, edited by O. S. Brüning, P. Collier, P. Lebrun, S. Myers, R. Ostojic, J. Poole, P. Proudlock, 2004.

[2] R. W. Assmann *et al.*, "Requirements for the LHC collimation system". In Proceedings of EPAC'02, Paris, France, pp. 197-199.

[3] G. Valentino *et al.*, "Semiautomatic beam-based LHC collimator alignment", Phys. Rev. ST Accel. Beams, **15** 015002 (2012).

[4] E. B. Holzer *et al.*, "Beam loss monitoring system for the LHC". In Proceedings of the IEEE Nuclear Science Symposium Conference Record, San Juan, Puerto Rico, 2005, pp. 1052-1056.

[5] G. Kruk, S. Deghaye, M. Lamont, M. Misiowiec, W. Sliwinski, "LHC software architecture (LSA) - evolution towards LHC beam commissioning". In Proceedings of ICALEPCS'07, Knoxville, Tennessee, USA, pp. 307-309.

[6] K. Kostro, V. Baggiolini, F. Calderini, F. Chevrier, S. Jensen, R. Swoboda, N. Trofimov, "Controls middleware - the new generation". In Proceedings of EPAC'02, Paris, France, 2002, pp. 2028-2030.

[7] M. Arruat *et al.*, "Front-end software architecture". In Proceedings of ICALEPCS'07, Knoxville, Tennessee, USA, 2007.

[8] A. Dallocchio *et al.*, "LHC collimators with embedded beam position monitors: a new advanced mechanical design". In Proceedings of IPAC'11, San Sebastian, Spain, 2011.

[9] B. Salvachua *et al.*, "Collimation system post-LS1: status and commissioning". In Proceedings of the 5th LHC Beam Workshop, Evian, France, 2014.

[10] D. Wollmann *et al.*, "Beam feasibility study of a collimator with in-jaw beam position monitors", Nucl. Instr. Meth. Phys. Res. A, Vol. 768, pp. 62-68 (2014).

[11] G. Valentino, A. Nosych, R. Bruce, M. Gasior, D. Mirarchi, S. Redaell, B. Salvachua, D. Wollmann, "Successive approximation algorithm for beam-position-monitor-based LHC collimator alignment", Phys. Rev. ST Accel. Beams, **17** 021005 (2014).

[12] A. Masi, R. Losito, "LHC collimators low level control system". IEEE Trans. Nucl. Sci., vol. 55, no. 1, pp. 333-340, 2008.

[13] G. Valentino, R. W. Assmann, R. Bruce, S. Redaelli, N. Sammut, "Automatic threshold selection for BLM signals during LHC collimator beam-based alignment". In Proceedings of UKSim/AMSS EMS'12, Valletta, Malta, pp. 210-213.

[14] M. Gasior *et al.*, "BPM electronics based on compensated diode detectors - results from development systems". In Proceedings of the 2012 Beam Instrumentation Workshop, Newport News, VA, USA, 2012.

[15] R. Bruce, R. W. Assmann, L. Lari, S. Redaelli, "Collimator hierarchy limits: assumptions and impact on machine protection and performance". In Proceedings of the Machine Protection Workshop, Annecy, France, 2013.

[16] J. Wozniak, V. Baggiolini, D. Garcia Quintas, J. Wenninger, "Software interlocks system". In Proceedings of ICALEPCS'07, Knoxville, Tennessee, USA.

[17] G. Valentino, R. W. Assmann, S. Redaelli, N. Sammut, "LHC collimator alignment operational tool". In Proceedings of ICALEPCS'13, San Francisco, CA, USA, 2013.

ISBN 978-3-95450-148-9