PROCEEDINGS OF SCIENCE

AI Techniques to Improve Optics Measurements Using Turn-by-Turn Beam Position Monitors

Quentin Bruant,[a,*] **Charles Ndung'u Ndegwa,**[b,c] **Yasmina Nasr,**[d] **Jonathan Piscart,**[d] **Leonardo Vitileia,**[d] **Barbara Dalena,**[a] **Francesca Bugiotti**[b] **and Valérie Gautard**[a]

[a] *Commissariat à l'énergie atomique - Institut de Recherche sur les lois Fondamentales de l'Univers (CEA/Irfu), Saclay, France*
[b] *CentraleSupélec / LISN / CNRS, Paris-Saclay University, Gif-sur-Yvette, France*
[c] *Department of Physics, University of Nairobi, Kenya*
[d] *CentraleSupélec, Gif-sur-Yvette, France*
E-mail: quentin.bruant@cea.fr, charles.ndungu-ndegwa@universite-paris-saclay.fr, barbara.dalena@cea.fr, francesca.bugiotti@centralesupelec.fr

A high energy collider ring needs a full network of sensors to operate. In particular, beam position monitors (BPMs) provide the transverse position of the beam in the chamber at multiple locations around the ring, and one category of BPMs, the Turn-by-turn BPMs (TbT-BPMs), is used to detect any magnetic defects in the lattice. Several methods exist to reconstruct this lattice and the associated optical functions, but each of them needs to have several functional TbT-BPMs dispersed all along the ring. In the context of the FCC, this involves several thousand BPMs scattered along a 91-km ring, operating in a very adverse environment, especially due to the effects of radiation on the electronics. Moreover, in order to maximize the duty cycle in this very large-scale accelerator, operation may have to occur even when some of its sensors are down. Consequently, the fast detection of faulty TbT-BPMs and the adaptation of the optics functions reconstruction are paramount, as is having sufficient confidence in the actual measurement, which needs to be sensitive and precise enough (i.e., having a high signal-to-noise ratio) to support decisions on the correction of the behavior of the multi-GeV beam. Indeed, the ability to function with very high precision in a very noisy environment is also a challenge that SuperKEKB, the largest e+/e- collider currently in operation, but also future colliders and light sources, need in order to measure rapidly and with high precision the optics functions. We present several machine-learning methods, adapted to the accelerator context, for detecting faulty TbT BPMs and for denoising the tracks in SuperKEKB, with the aim of identifying an approach that is efficient and scalable to the scale of the FCC accelerator.

The European Physical Society Conference on High Energy Physics (EPS-HEP2025)
7-11 July 2025
Marseille, France

[*] Speaker

PoS(EPS-HEP2025)587

1. Introduction

A beam position monitor (BPM) [1] consists of two or four symmetrically arranged electrodes coupled to the electromagnetic field of the circulating beam, thereby generating electrical signals at their output. The combination of the signals from the pickup electrodes allows the transverse position of the beam centroid to be determined as the beam passes the sensor. A special type of BPMs are the Turn-by-Turn (TbT) monitors, which are able to provide the beam centroid information at every turn. In a ring accelerator, a synchronized and distributed TbT BPM system is of particular interest for tracking the beam response to excitations, thereby enabling the reconstruction and analyses of the linear and non-linear optical parameters of the accelerator [2] [3]. However, BPM data quality is often degraded by two major factors: faulty sensors and background noise. The analysis pipeline developed for LHC TbT-BPMs, detailed in Ref. [4], identifies faulty BPMs by applying a manual threshold based on physics-constraint knowledge (chamber transverse size, *etc*. . .). An alternative approach is presented in Ref. [5], which applies an Isolation Forest algorithm (IF) to automatically flag outliers, after an SVD-based white noise reduction of the position time series performed prior to detection, as detailed in Ref. [6].
The main rings of SuperKEKB have 68 TbT-BPMs in the High Energy Ring (HER) and 70 in the Low Energy Ring (LER). These BPMs are not used as the primary system to operate the machine; therefore, their performance is not continuously monitored (*i.e* the list of malfunctioning BPMs provided by the operation team may not be exhaustive). The complexity of the accelerator optics also makes the reconstruction of its parameters with the previously described analysis pipeline very challenging. In the following, we explore the possibility of adding machine-learning-based techniques to this proven pipeline in order to improve the detection of faulty BPMs and to maintain the quality of linear and non-linear optics parameters reconstruction. In this study, we used TbT-BPM data taken during SuperKEKB commissioning and operation in 2024. The data consists of about 500 files containing either 68×2 (HER) or 70×2 (LER) one-dimensional time series corresponding to the horizontal and vertical beam centroid position for 1000 or 4096 turns, according to the measurement.

2. Anomaly Detection

Herein, we follow the procedure shown in Fig 1. We compare three unsupervised methods based on different algorithms: an IF, a Density-Based Spectral Clustering with Noise (DBSCAN), and an AutoEncoder (AE). IF has already been used to detect faulty BPMs in the LHC TbT-BPM system [5]. The other two algorithms have been chosen because they complement the IF although with an increasing level of complexity (Fig. 2). We use experimental datasets acquired during February 2024 and June 2024 on-site displacements, and the list of known hardware-damaged BPMs was provided by the SuperKEKB operations team.

2.1 SVD-Isolation Forest

This fault detection methodology identifies malfunctioning BPMs through a four-stage pipeline: signal pre-processing, SVD-based denoising, windowed feature extraction, and unsupervised anomaly detection through IF. Each stage addresses specific challenges in BPM diagnostics. Pre-processing

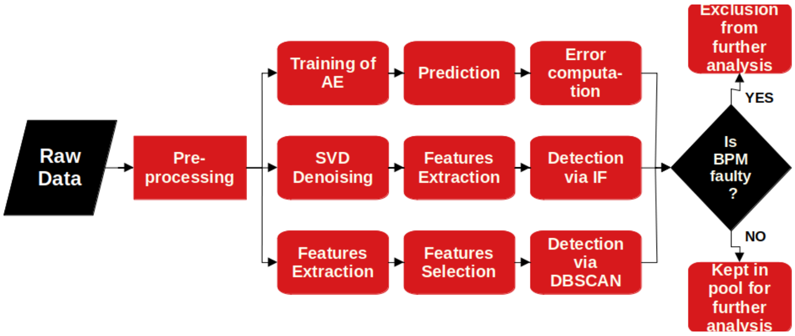


Figure 1: Workflow followed for anomaly detection

normalizes signals and removes systematic offsets. SVD denoising stabilizes statistical features by suppressing high-frequency measurement artifacts. Feature extraction quantifies signal characteristics in overlapping time windows. We segment each channel (x, y) into overlapping windows of length 512 with hop of 256. The analysis is limited to the first 1000 turns for each measurement. For each window, we compute statistical and spectral descriptors independently on x and y, then concatenate across channels: *Central tendency & variability:* mean, standard deviation, range, coefficient of variation; *Distribution shape:* skewness, kurtosis; *Complexity:* Shannon entropy; *Temporal structure:* zero-crossing rate; *Cross-channel coupling:* Pearson correlation between x and y within the window; *Frequency domain:* dominant frequency (via quadratic-interpolated FFT peak) and spectral concentration (ratio of power in the top 5 FFT bins to total in-band power). Features are standardized using a robust scaler (median and interquartile range) to mitigate the influence of extreme outliers. The resulting per-window vectors form a per-BPM feature matrix spanning all windows and runs. Finally, after feature extraction, we applied Isolation Forest (IF). Each window's feature vector is fed into the model without labels. We fit IF with n_estimators=100, contamination=0.06, and we collect results for four random_state. Windows receive labels (-1 anomalous, +1 normal). We assign BPM labels from window-level outputs using the following rules:

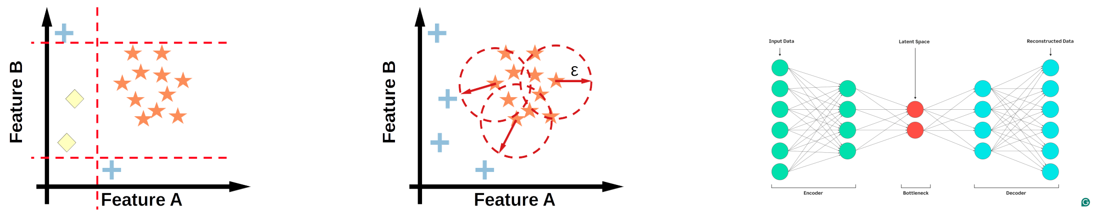
- **Faulty**: anomaly count $\geq number\ of\ window$;
- **Suspected Faulty**: $number\ of\ window/3 \leq$ anomaly count $< number\ of\ window$;
- **Correct**: none of the above conditions hold.

2.2 Density-Based Spectral Clustering with Noise (DBSCAN)

DBSCAN is a clustering algorithm applied to the vectors representing the BPMs in a hyperspace defined by the statistical features of the signal of each BPM. We used it in a 1 vs All approach, where we assume the main distribution of BPM features represents the functioning ones and the detected outliers are the faulty ones. The main asset of this axis is the path chosen to compute these features. Instead of choosing the features that will be computed for all time series of all measurements *a priori*, with this method, they are computed for each run of each measurement and are computed and selected *a posteriori* from the time series *via* a library called Time2feat [7]. This selection is done automatically *via* a Principal Component Analysis (PCA) applied on the variance explainability of the set of features of BPMs time series. The main asset of this method is the ability to detect if a given BPM is indeed a faulty measurement to measurement. This key feature of the algorithm is important because some SuperKEKB TbT-BPMs appear to be problematic only when a given set of measurements is achieved in the machine. After the extraction and the selection of the most interesting features to discriminate the problematic BPMs from the good ones, Time2feat is also designed to apply different clustering methods to the data selected features. We modified the code of the library to insert the DBSCAN algorithm in the clustering area of the library. The full characterization of this method is ongoing using simulated TbT-BPM data obtained by tracking. In particular in order to optimize the hyperparameters such as the ε distance. In the results presented below, the ε distance was put at 7.5 in the hyperspace of features for both rings.

2.3 One-dimensional Convolutional (1D-CNN) autoencoder

A 1D-CNN autoencoder [8] was developed to detect faulty BPM signals by learning the normal temporal structure of TbT data. The input consisted of sequences of 3995 turns for the HER and 995 turns for the LER. Each sequence was pre-processed by trimming a few first turns to remove startup artifacts and applying Z-score normalization to standardize the scale. The encoder applied a series of strided convolutional layers with ReLU activations [9] to compress the signal into a latent representation. The decoder mirrored this process with transposed convolutions to reconstruct the original sequence length. The model was implemented in Python using PyTorch [10] and trained separately for the HER and LER datasets. For the HER, 63 tracks from June 2024 were used for training and 12 for validation. For the LER, 9 tracks from February 2024 were used for training and 5 for validation. Training was carried out for 50 epochs with the Adam optimizer, a learning rate of 10^{-3}, a batch size of 16, and a mean squared error as a loss function. After training, reconstruction errors were computed for each BPM. In each measurement, the ten BPMs with the largest errors were marked as outliers, and their recurrence across measurements was aggregated into frequency plots to identify persistent anomalies.



(a) Isolation Forest (IF)
Stars : Main distribution
Crosses : Outlier
Diamonds : Suspect

(b) Density-Based Spectral Clustering Application w/ Noise (DBSCAN)
Stars : Main distribution
Crosses : Outlier

(c) Autoencoder (AE)
Left : Encoder
Right : Decoder
Center : Latent space

Figure 2: Sketches of the operation of Isolation Forest, DBSCAN, and Autoencoder, respectively

3. Results

The results obtained by the three methods applied are presented in Table 1. For the DBSCAN, the Time2feat library managed to extract 201 and 207 statistical features for the HER and the LER, respectively, out of which 37 and 31 were selected (from a PCA threshold put at 90%). In the HER data (taken in June), five out of the six known-faulty BPMs are detected by the DBSCAN and the Autoencoder algorithms (~ 83%), while the Isolation Forest detects four (~ 67%). Furthermore, all three algorithms also detect several BPMs that are not in the list provided. One can see in Figure 3, that one of the BPM detected which is not in the list provided seems to also be quite problematic which lead to the assumption that the list provided is not complete. In the LER data (taken in February), the performance of the Autoencoder is stable with a recall of two-thirds. The Isolation Forest appears to be better than with the HER, with five BPM detected out of the six communicated (~83%) while the DBSCAN shows a poorer performance (50%). One can also note that the total number of detected BPM is stable (between 8 and 11), which is expected since the total number of BPM in both rings is similar (67 in HER, 70 in LER), and the dynamics, albeit proper to each ring, is similar. This stability, combined with the evolution of the performance to detect the BPM in the communicated list, causes an increase in the detection of BPM not included in the given list. For the DBSCAN, this effect may be due to the fact that the algorithm tested here was tuned on the HER and not the LER. In the Autoencoder, the reconstruction errors were more evenly distributed across BPMs. This broader distribution likely reflects the limited size and diversity of the LER training set, which restricted the model's ability to learn representative beam dynamics. The computation time of the DBSCAN method is around 45 minutes, with most of it dedicated to the sequential computation of the features. It is one of the main drawbacks of the method since, due to some features being highly combinatorial, this time extends rapidly with the size of the dataset. On the contrary, the Isolation Forest in our configuration is fast (in the order of a few seconds) since the features are computed *a priori*, are in a limited number compared to the amount computed in the DBSCAN method, and are limited to a fraction of each BPM track. With its different point of view and behavior with respect to the data, the Autoencoder can infer results in a few seconds once trained. Depending on the size of the representative datasets used for its training, this step takes few minutes to few tens of minutes. This step, being the core of the algorithm performance, is difficult to shrink.

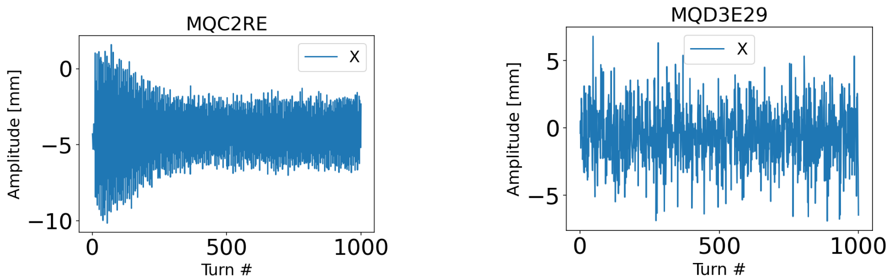


Figure 3: Plot of the transverse position of the beam. *Left:* functioning BPM. *Right:* faulty BPM.

4. Conclusion

The three methods explored in this proceeding offer complementary points of view for the detection of potentially problematic TbT-BPMs, especially for the LER of SuperKEKB, with all known-faulty BPMs detected by combining the three methods. All methods show a preliminary

Table 1: Comparison of machine-learning–based anomaly detection methods applied to HER and LER BPM data. The table reports the known faulty BPMs provided by the SuperKEKB operations team, the BPMs flagged by each method, and the corresponding recall (percentage of known faulty BPMs correctly identified).

Date [2024]	Ring	Known faulty BPMs	Method (Recall [%])	Detected BPMs
Feb	LER	**MQEAP35**, **MQW2ORP**, **MQEAP29**, **MQD3P8**, **MQEAP10**, **MQD3P23**	DBSCAN (50%)	**MQEAP35**, **MQW2ORP**, **MQEAP29**, MQEAP32, MQEAP33, MQI6P, MQEAP38, MQEAP44, MQD3P29
			1D-CNN AE (~67%)	**MQW2ORP**, **MQEAP29**, **MQD3P8**, **MQD3P23**, MQC1LP, MQD3P20, MQD3P21, MQD3P29, MQD3P38, MQD3P11
			IF (~83%)	**MQW2ORP**, **MQEAP29**, **MQD3P23**, **MQEAP35**, **MQEAP10**, MQEAP44, MQEAP32, MQEAP30, MQN6OP, MQEAP33, MQEAP38
Jun	HER	**MQEAE35**, **MQD3E18**, **MQEAE20**, **MQR2ORE**, **MQD3E23**, **MQD3E8**,	DBSCAN (~83%)	**MQEAE35**, **MQD3E18**, **MQEAE20**, **MQR2ORE**, **MQD3E8**, MQEAE25, MQEAE33, MQD3E29
			1D-CNN AE (~83%)	**MQEAE35**, **MQD3E18**, **MQEAE20**, **MQR2ORE**, **MQD3E8** MQFRNE3, MQEAE25, MQEAE30, MQLB8RE, MQEAE27, MQCIILE
			IF (~67%)	**MQEAE35**, **MQD3E18**, **MQEAE20**, **MQR2ORE**, MQFRNE3, MQEAE25, MQEAE30, MQLB8RE, MQEAE27, MQCIILE

efficiency above 50% in retrieving known-faulty BPMs along both rings. With its different approach, the AE complements the feature-based methods presented, thus offering a richer diagnostic perspective.

5. Acknowledgment

This project has received funding from the European Union's Horizon-Europe research and innovation programme under the Marie Skłodowska-Curie Grant Agreement ID: 101086276 (EA-JADE Research and Innovation Staff Exchange action). Furthermore, the authors gratefully acknowledge the support of the French Development Agency and Expertise France through the University of Nairobi Engineering and Science Complex mobility grant. We also extend our gratitude to Y. Ohnishi and the SuperKEKB operation teams for their invaluable insight in their collider and J.Keintzel of CERN, for her support in the collider data processing.

PoS(EPS-HEP2025)587

References

[1] Peter Forck, Dmitry Liakin, and Piotr Kowina. Beam position monitors. 2009.

[2] A. Langner and R. Tomás. Optics measurement algorithms and error analysis for the proton energy frontier. *Phys. Rev. ST Accel. Beams*, 18:031002, Mar 2015.

[3] A. Wegscheider, A. Langner, R. Tomás, and A. Franchi. Analytical *n* beam position monitor method. *Phys. Rev. Accel. Beams*, 20:111002, Nov 2017.

[4] Jacqueline Keintzel. Soma: Superkekb optics measurement analysis, 2024. Version: vX.Y (commit abc123); accessed: 2025-10-27.

[5] E. Fol, R. Tomás, J. Coello de Portugal, and G. Franchetti. Detection of faulty beam position monitors using unsupervised learning. *Phys. Rev. Accel. Beams*, 23:102805, Oct 2020.

[6] L. Malina, J.M. Coello de Portugal, J. Dilly, P.K. Skowroński, R. Tomás, and M.S. Toplis. Performance Optimisation of Turn-by-Turn Beam Position Monitor Data Harmonic Analysis. In *Proc. 9th International Particle Accelerator Conference (IPAC'18), Vancouver, BC, Canada, April 29-May 4, 2018*, number 9 in International Particle Accelerator Conference, pages 3064–3067, Geneva, Switzerland, June 2018. JACoW Publishing. https://doi.org/10.18429/JACoW-IPAC2018-THPAF045.

[7] Angela Bonifati, Francesco Del Buono, Francesco Guerra, and Donato Tiano. Time2feat: learning interpretable representations for multivariate time series clustering. *Proc. VLDB Endow.*, 16(2):193–201, October 2022.

[8] Ayokunle Olalekan Ige and Malusi Sibiya. State-of-the-art in 1d convolutional neural networks: A survey. *IEEE Access*, 2024.

[9] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. *Advances in neural information processing systems*, 30, 2017.

[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

PoS(EPS-HEP2025)587