AI techniques to improve optics measurements based on the Turn-by-turn Beam Position Monitors

Q. Bruant, Y. Nasr, L. Vitileia, J. Piscart, H. Le Corre,

A. Gomez, C. Ndungu-Ndegwa,

F. Bugiotti, B. Dalena,

J. Keintzel, Y. Ohnishi,

V. Gautard

Thanks to: M. Le Garrec, H. Sugimoto

Outline

- Motivation and context
- Current state of SuperKEKB turn by turn BPM analysis
- Anomaly detection
- Signal denoising
- Overview & Perspectives

CONTEXT & MOTIVATION

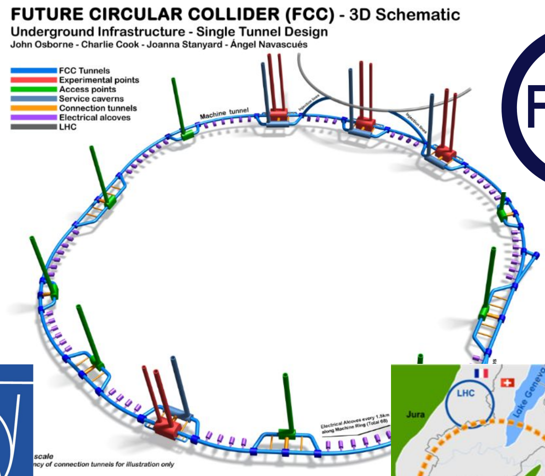
Context

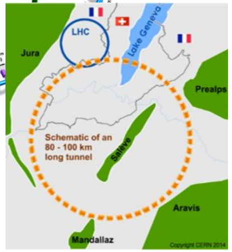
International FCC collaboration (CERN as host lab) to study:

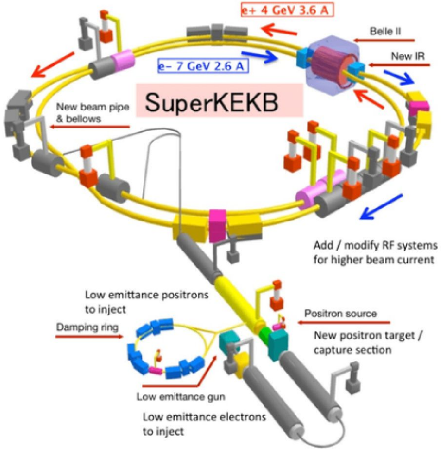
- 91 km tunnel infrastructure in Geneva area, site specific
- e^+e^- collider (*FCC-ee*), as first step

Feasibility study published in March

[Future Circular Collider Feasibility Study Report Volume 2 - CERN Document Server]





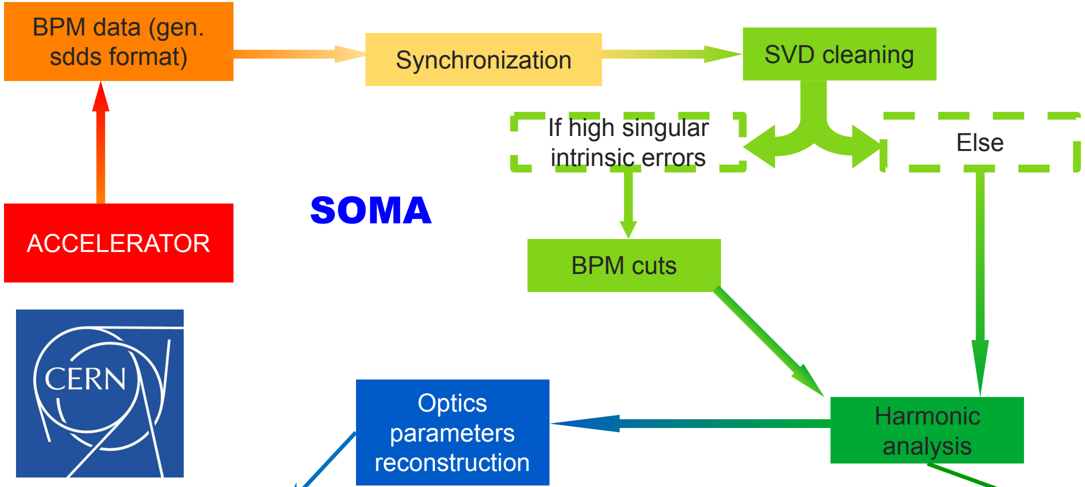


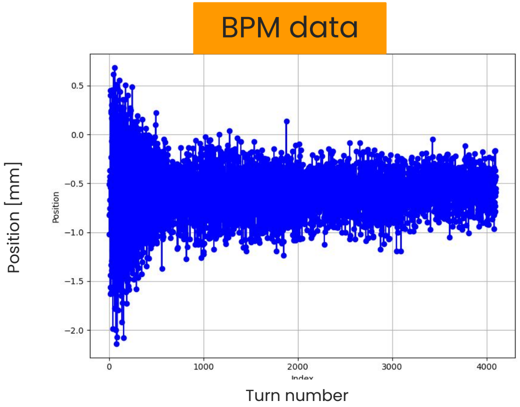
<- Use of SuperKEKB as a test bench

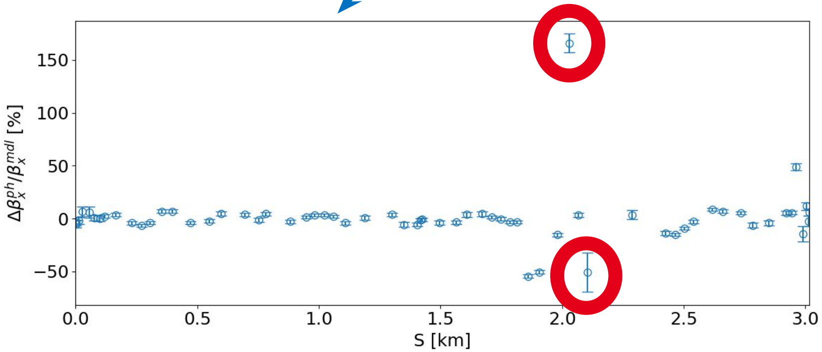
Motivation and challenges

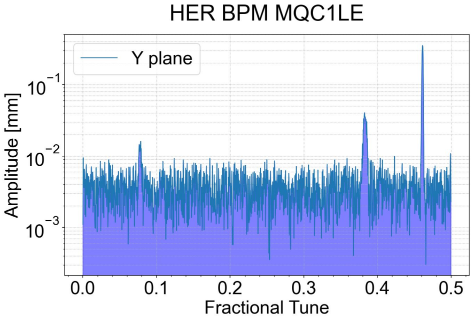
Challenge	1000s of BPM in the FCC accelerators -> noisy, sometime faulty
Impact	Faulty BPMs -> unreliable measurements of optical functions
Goal	Use of ML methods to detect faulty BPMs and denoise BPMs signal
Opportunity	Reduce probability of error-induced beam loss -> allow for more continuous operation at FCC scale

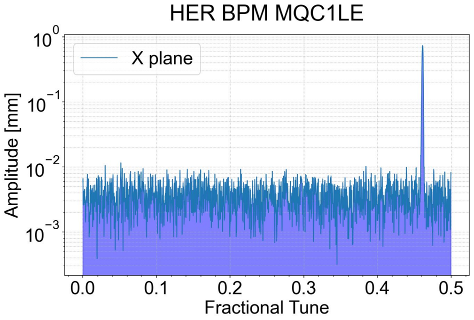
Current TbT BPM data pipeline







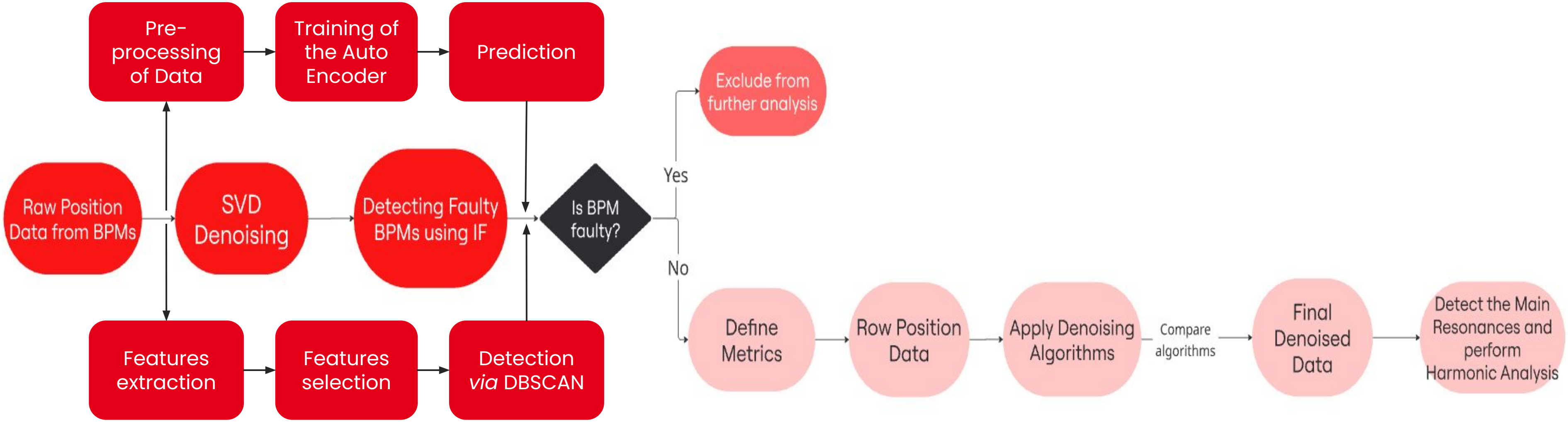




Improved TbT BPMs data pipeline

Automatically detect faulty BPMs

Reduce noise on good BPMs



ANOMALY DETECTION

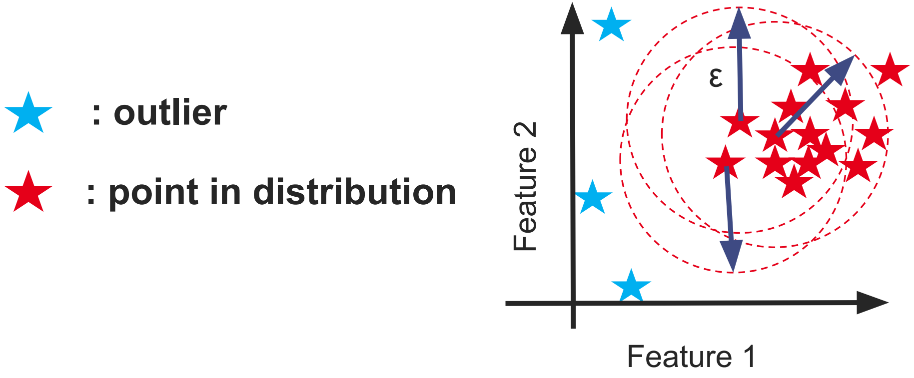
ANOMALY DETECTION

ALGORITHMS

DBSCAN / Isolation Forest (IF)

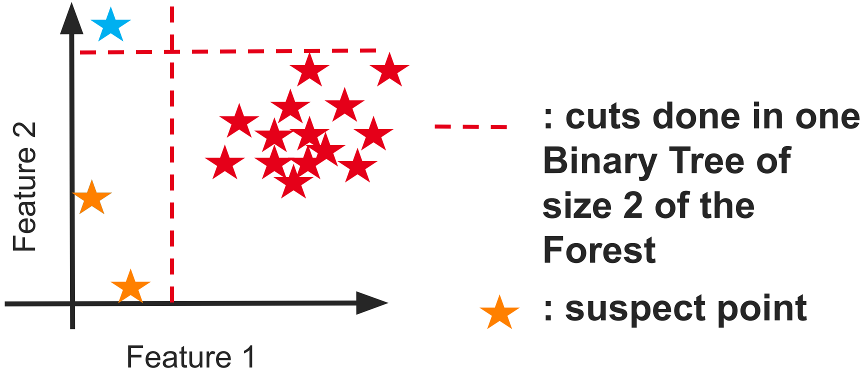
DBSCAN:

- Clustering algorithm in 1vAll config.
  - Hyperspace of features extracted from Multivariate Time Series (MTS)
- Distance-based approach

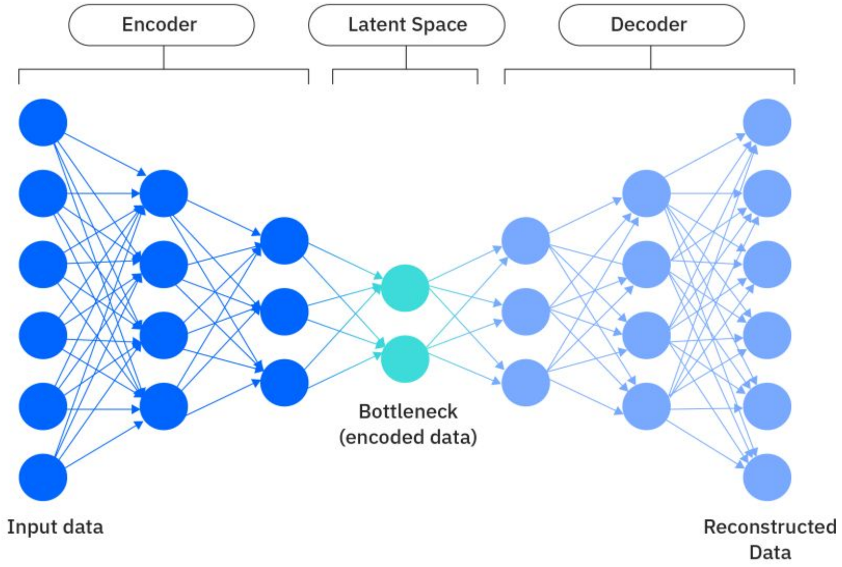


IF:

- Random-based Binary Forest:
  - Hyperspace of features extracted from Multivariate Time Series (MTS)
- Isolation via random cuts in features



AutoEncoder



Schematic of an Autoencoder: Encoder → Latent → Decoder

- Learns to reconstruct input via a compressed bottleneck.
- No labels needed — fully unsupervised.
- Used to learn dominant structures in the full dataset.

Definition:

$$\text{Encoder: } z = f_{\text{enc}}(x)$$
$$\text{Decoder: } \hat{x} = f_{\text{dec}}(z)$$
$$\text{Autoencoder: } \hat{x} = f_{\text{dec}}(f_{\text{enc}}(x))$$

Loss (MSE):

$$\mathcal{L}_{\text{MSE}} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2$$

Anomaly Score:

$$\text{Score}(x) = \frac{1}{HW}\sum_{j=1}^{H}\sum_{k=1}^{W}(x_{jk} - \hat{x}_{jk})^2$$

Thresholding:

$$\text{Anomaly if Score}(x) > \tau$$

ANOMALY DETECTION

PROCESS & RESULTS

Followed process (IF)

We designed a full pipeline to process the data, clean it, extract features and detect outliers

Dataset: several (~few 10s) measurements done in February 2024

DATA PARSING

- Extracting data from experiment files
- Separation HER/LER

BUILD BPMs MATRICES

- Synchronisation and combination of X and Y planes signals across all BPMs
- First selection

SIGNAL DENOISING & FEATURE EXTRACTION

- Window filtering over each BPM signal
- SVD denoising
- Extraction of features (Variance, Skewness, Kurtosis,...)

ANOMALY DETECTION

- Apply Isolation Forest (IF) to detect abnormal behavior

CLASSIFICATION

- Anomaly count for each BPM.
- Classification in 3 classes: Faulty, Suspect, Good

Results (IF)

LER_2024_02_06

HER_2024_02_06

4/6 from the list of the expected faulty BPM in HER

4/6 from the list of the expected faulty BPM in LER

Several detection in HER that are not in the list

Faulty BPM HER (IF)	Expected faulty BPM HER	Faulty BPM LER (IF)	Expected faulty BPM LER
MQEAE35	MQEAE35	MQD3P23	MQD3P23
MQD3E18	MQD3E18	MQEAP29	MQEAP29
MQD3E23	MQD3E23	MQEAP35	MQEAP35
MQR2ORE	MQR2ORE	MQW2ORP	MQW2ORP
	MQD3E8		MQEAP10
	MQEAE20		MQD3P8
MQLB1LE			
MQC2RE			

Followed process (DBSCAN)

Features extraction by Time2Feat library

Features selection by Time2Feat based on a PCA decomposition w/ explainability variance as figure of merit

Features = statistical parameters (mean, variance, skewness, autocorrelation, FFT coefficients, ….)

➢ **1 file, all BPMs**

For HER:

Dataset: HER_2024_02_06_IK_H_Vinjkick/HER_2024_02_06_16_22_57.data

Total number of features kept: 37/201

For LER:

Dataset: LER_2024_02_06_IK_H_Vinjkick/LER_2024_02_06_17_02_14.data

Total number of features kept: 31/207

Results (DBSCAN)

LER_2024_02_06_17_02_14.data

HER_2024_02_06_16_22_57.data

Faulty BPM HER (DBSCAN)	Expected faulty BPM HER	Faulty BPM LER (DBSCAN)	Expected faulty BPM LER
MQEAE35	MQEAE35	MQEAP35	MQEAP35
MQD3E18	MQD3E18	MQW2ORP	MQW2ORP
MQEAE20	MQEAE20	MQEAP29	MQEAP29
MQD3E8	MQD3E8		MQD3P8
MQR2ORE	MQR2ORE		MQEAP10
	MQD3E23		MQD3P23
MQEAE25		MQEAP32	
MQEAE33		MQEAP33	
MQD3E29		MQI6P	
		MQEAP38	
		MQEAP44	
		MQD3P29	

HER:
5/6 of expected faulty BPM are detected

LER:
3/6 of expected faulty BPM are detected

Variability for detected BPMs in LER

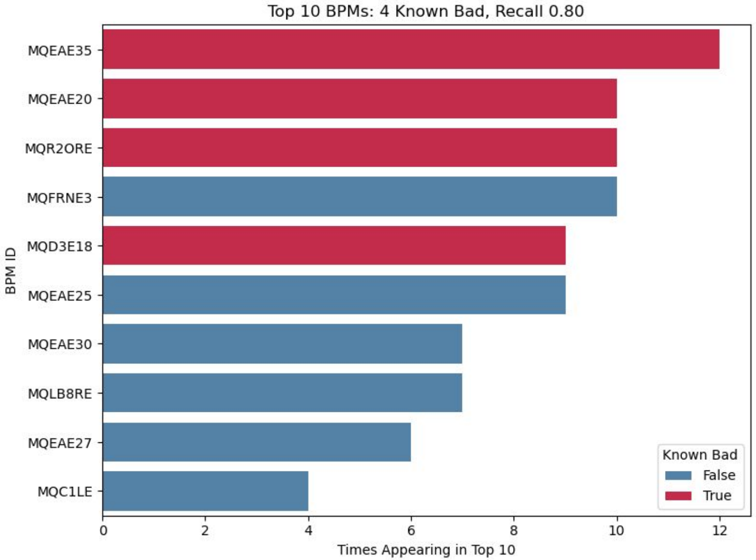
Followed process (AutoEncoder)

Unsupervised 1D CNN Autoencoder

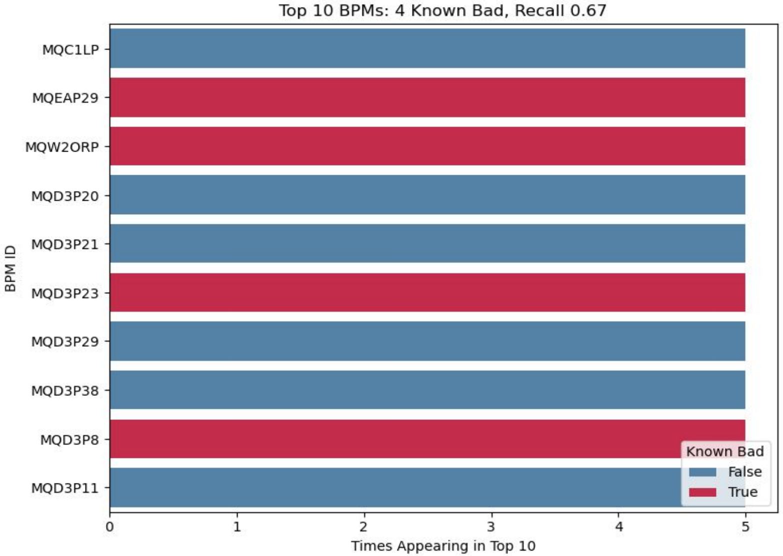
- **Preprocessing:** Turn trimming (5 turns) & Z-score normalization
- **HER Model:** Trained on 63 (2024_06), inferred on 12 (3995 turns) tracks
- **LER Model:** Trained on 9 (2024_02_06), inferred on 5 (995 turns) tracks
- **Outlier Detection:** Top 10 BPMs by reconstruction error, aggregated by frequency
- **Validation:** Compared to known faulty BPMs
- **Output:** BPM outlier frequency plots

Results (AutoEncoder)

HER/2024_06



LER/2024_02_06



- **HER Model:** Higher recall (identifies more known bad BPMs)
  - **Diagnostic Value:** Varied outlier frequencies highlight *persistently* problematic BPMs
- **LER Model:** Lower recall; uniform outlier frequencies
  - **Limitation:** Likely due to significantly smaller training data
- **Common Finding:** Both models identify additional outlier BPMs not on known-faulty lists – critical for further investigation

Overview

HER				LER			
IF	DBSCAN	AutoEncoder	Expected	IF	DBSCAN	AutoEncoder	Expected
MQEAE35	MQEAE35	MQEAE35	MQEAE35	MQEAP35	MQEAP35		MQEAP35
MQD3E18	MQD3E18	MQD3E18	MQD3E18	MQW2ORP	MQW2ORP	MQW2ORP	MQW2ORP
	MQEAE20	MQEAE20	MQEAE20	MQEAP29	MQEAP29		MQEAP29
MQR2ORE	MQR2ORE	MQR2ORE	MQR2ORE			MQD3P8	MQD3P8
	MQD3E8		MQD3E8				MQEAP10
MQD3E23			MQD3E23	MQD3P23			MQD3P23
MQLB1LE		MQFRNE3			MQEAP32	MQC1LP	
MQC2RE		MQEAE30			MQEAP33	MQEAP29	
	MQD3E29	MQLB8RE			MQI6P	MQD3P20	
	MQEAE33	MQEAE27			MQEAP38	MQD3P21	
	MQEAE25	MQEAE25			MQEAP44	MQD3P23	
					MQD3P29	MQD3P29	
						MQD3P38	
						MQD3P11	

■ DENOISING

Evaluation Metrics for Denoising

Peak Retention Ratio

$$\mathrm{PRR} = \frac{\text{amplitude}(\text{max peak denoised})}{\text{amplitude}(\text{max peak original})}$$

How well the main peak is preserved

Noise Reduction Ratio

$$\mathrm{NRR} = 1 - \frac{\sum \text{noise\_denoised}}{\sum \text{noise\_original}}$$

How much noise is removed

SNR Improvement (Signal/Noise Ratio)

$$\mathrm{SNR} = 10 \log_{10} \left(\frac{\sum \text{signal power}}{\sum \text{noise power}} \right)$$

$$\Delta\mathrm{SNR} = \mathrm{SNR}_{\text{denoised}} - \mathrm{SNR}_{\text{original}}$$

How much signal quality is improved

Frequency Shift

$$\text{Frequency Shift} = |f_{\text{max\_original}} - f_{\text{max\_denoised}}|$$

How much the main frequency changed

Algorithms

SVD

Separates signal from noise by decomposing data into orthogonal components, and keeping the largest ones

LSTM

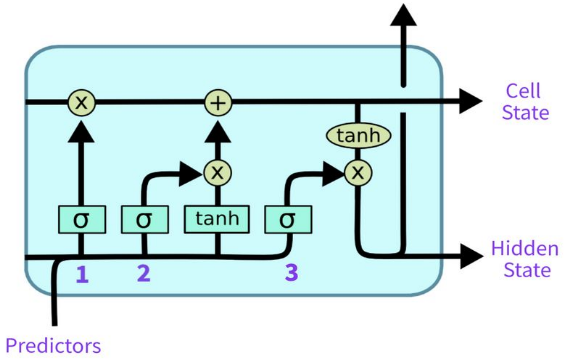
Uses a recursive neural network (RNN) to predict the true positions, without the random noise

Overall Purpose: Compare fast linear filters (SVD) with a more powerful but slower learning-based method (LSTM)

What is an LSTM

LSTMs work in the same way as RNNs, but each cell is more complex:

- LSTMs maintain a "**cell state**" in addition to the hidden state allowing it to keep longer-term information
- This helps to **avoid the vanishing gradient problem**



We combine these cells into 3 layers of 50 LSTM cells

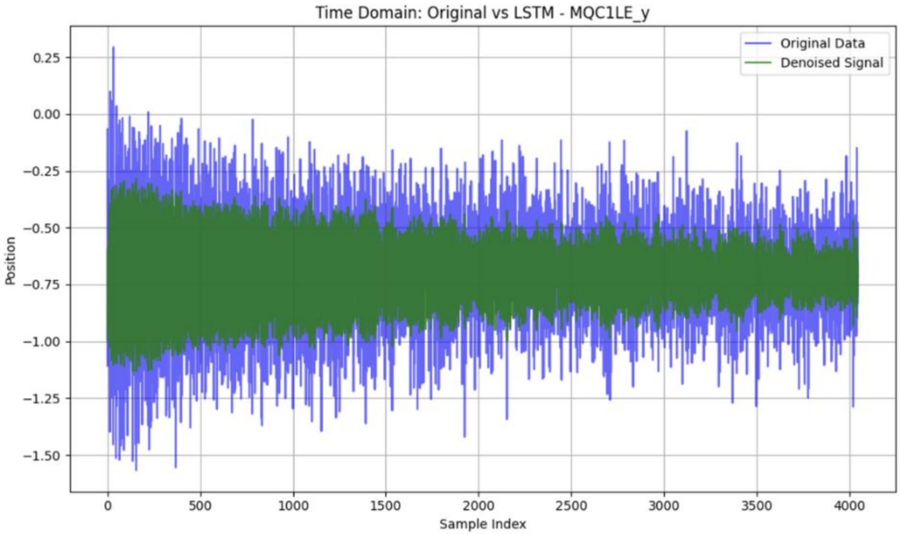
1. Having a deep model allows it to learn complex relationships in the data
2. We add BatchNorm and Dropout for normalization

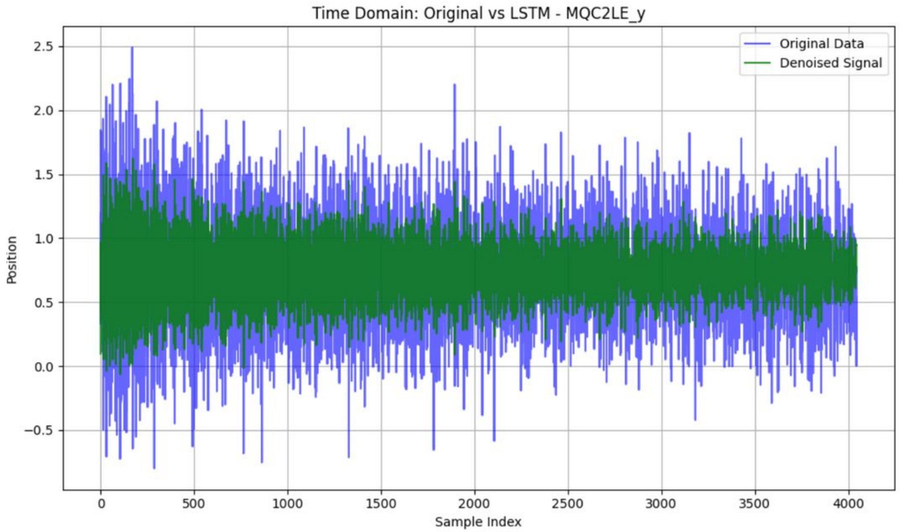
We train the model on a series of noisy positions

1. The model learns the underlying patterns in the data
2. We then use the model's predictions as our estimate of the denoised signal

Example BPM positions

File used: HER_2024_06_25_23_52_21.data



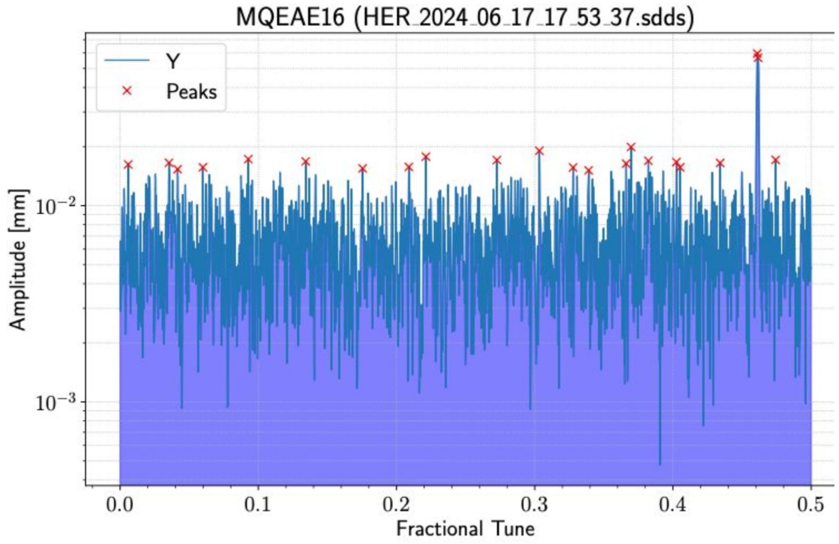


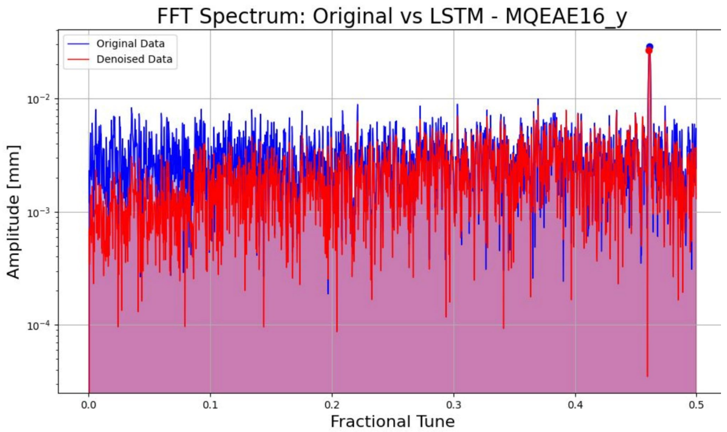
Results

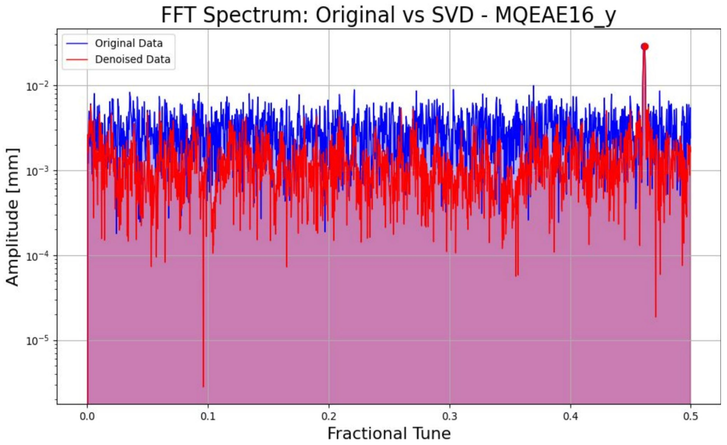
Metrics	SVD	LSTM
Peak Retention Ratio	0.99	0,95
Noise Reduction Ratio	0.45	0.57
SNR	2.99	8,4
Frequency Shift	0.0035	0
Time to denoise	< 1 min	~ 1h

Models comparison with SOMA

Frequency dependant denoising ->







Full Overview

- ➢ Comparison of 3 ML algorithms (Isolation Forest (IF) vs Density Based Spatial Clustering Application w/ Noise (DBSCAN) vs AutoEncoder)
- ➢ Detection of ~66% of previously identified faulty BPMs in both rings
  - IF appear to have the greater purity, DBSCAN the greater efficiency, AutoEncoder seems to be the most robust
- ➢ More powerful denoising with LSTM vs current SVD at expense of getting a frequency-dependant denoising and longer computation time

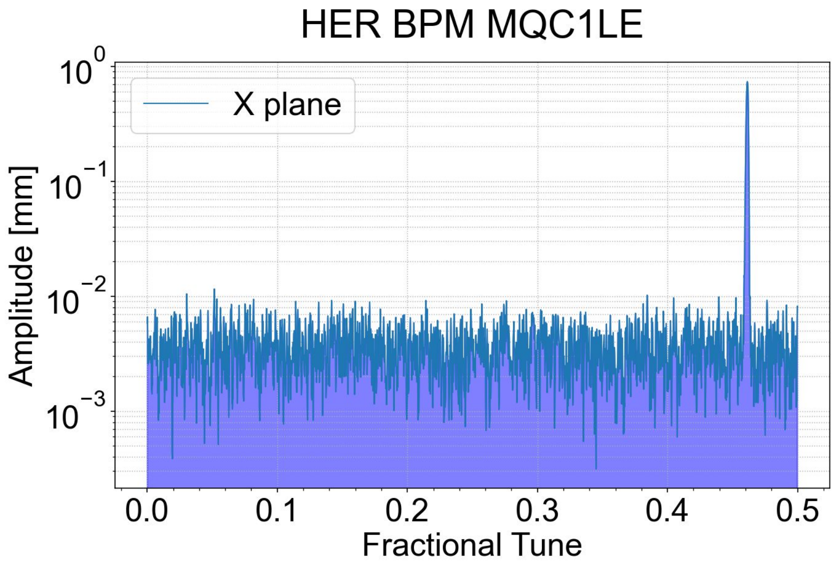
Perspectives

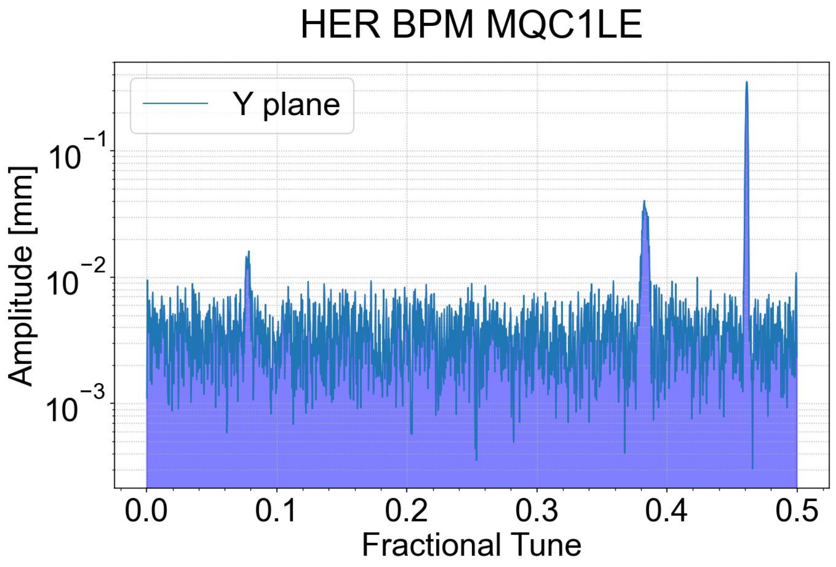
- ➢ Characterize precisely the algorithms on simulated BPMs signals both for SuperKEKB and FCC-ee High Energy Booster
- ➢ Further understanding and refinement of the algorithms
- ➢ Reconstruct the Optics functions with SOMA on denoised data
- ➢ Look at other AI techniques (ESN, DeepNN, etc,...)

Thanks for your attention

Example of superKEKB TbT BPMs analysis

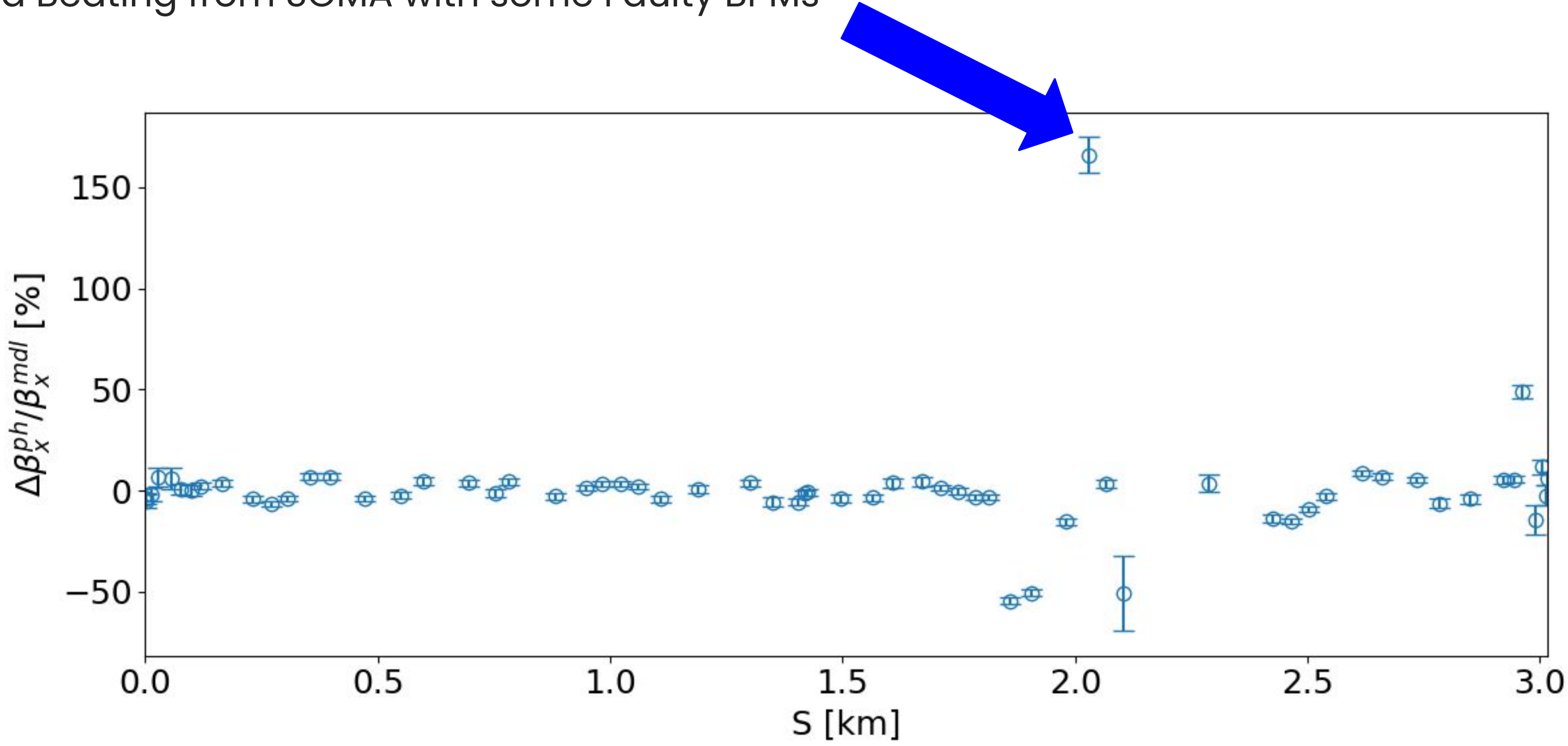
Spectra of June 17 - 2024 measurements employing Harpy (from SOMA)



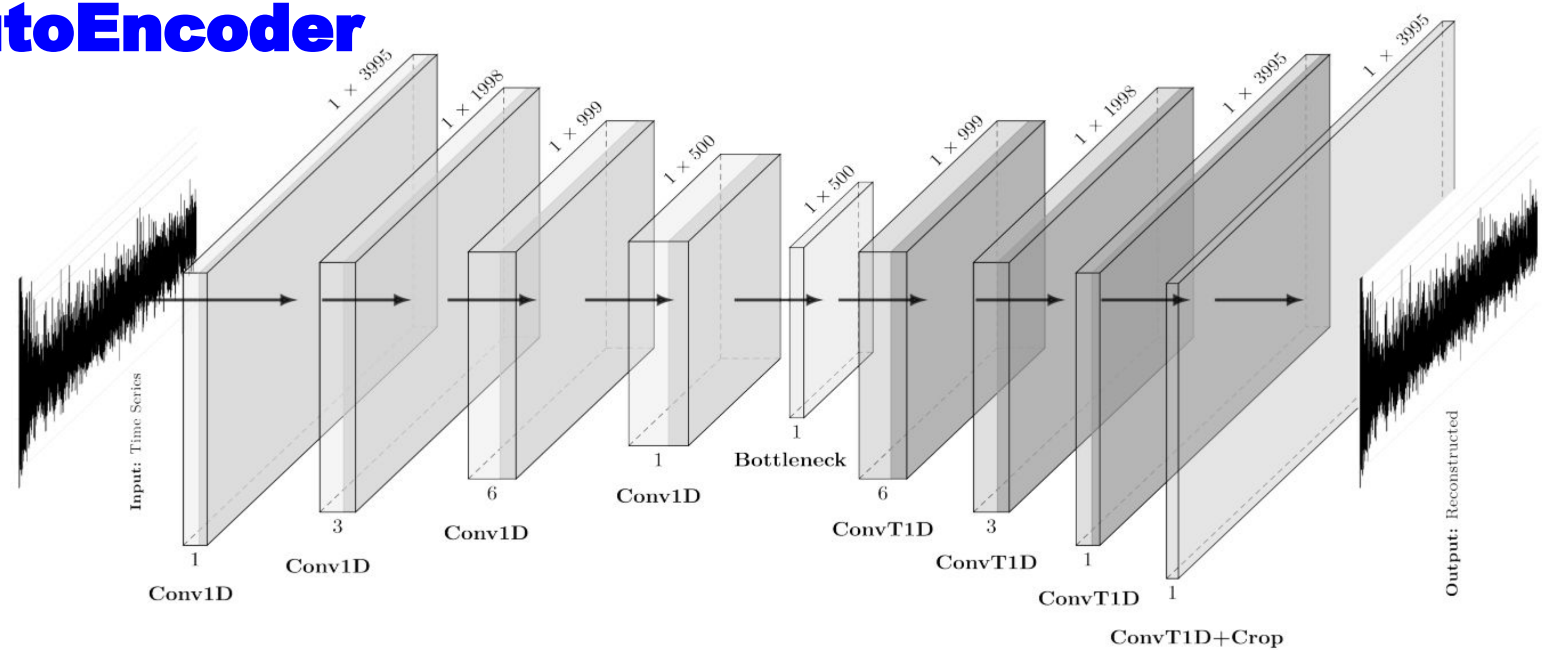


Example of superKEKB TbT BPMs analysis

Beta Beating from SOMA with some Faulty BPMs



AutoEncoder



Encoder

- 4 Conv1D + ReLU
- Kernel sizes: [7, 7, 5, 5]
- Stride = 2
- Channels: 1 → 16 → 32 → 64 → 128

Decoder

- 4 ConvTranspose1D + ReLU
- Kernel sizes: [5, 5, 7, 7]
- Stride = 2, output padding = 1
- Final crop: match 3995-point input

Data & Training

- Input: 3995-point BPM traces
- Normalization: mean/std per trace
- Optimizer: Adam (10^{-3}), Batch size: 16, Epochs: 50, Loss: MSE

Examples of features