Application of machine learning techniques at the CERN Large Hadron Collider

EPS-HEP 2019, Ghent
Friday July 12, 2019

F.F. Van der Veken\textsuperscript{1,2},
G. Azzopardi\textsuperscript{1,2}, F.H. Blanc\textsuperscript{4}, L.T.D. Coyle\textsuperscript{1,4}, E. Fol\textsuperscript{1,3}, M. Giovannozzi\textsuperscript{1}, T. Pieloni\textsuperscript{1,4}, S. Redaelli\textsuperscript{1}, L. Rivkin\textsuperscript{4,5}, B.M. Salvachua Ferrando\textsuperscript{1}, M. Schenk\textsuperscript{1,4}, R. Tomas Garcia\textsuperscript{1}, G. Valentino\textsuperscript{1,2},

\textsuperscript{1}CERN, \textsuperscript{2}University of Malta, \textsuperscript{3}Johann-Wolfgang-Goethe University, \textsuperscript{4}Ecole Polytechnique Federale Lausanne, \textsuperscript{5}Paul Scherrer Institut
Outline

1. Introduction
2. Collimator alignment
3. Optics measurements and corrections
4. Dynamic aperture studies
5. Beam lifetime optimisation
6. Conclusions and Outlook
Introduction

Machine Learning

- ML has been used extensively in several domains
- Very recently also in accelerator physics
- Also for the LHC at CERN
- In this talk: 4 collaborations working on different topics
Introduction

Machine Learning at the LHC

Concerning **machine operation**:
- Collimator alignment
- Recognition of faulty monitors
- Correction of beam optics variables

Concerning **analysis** of measurements and simulations:
- Anomaly detection in tracking simulations
- Extrapolation of tracking simulations
- Modelling beam lifetime by operational settings
Outline

1. Introduction
2. Collimator alignment
3. Optics measurements and corrections
4. Dynamic aperture studies
5. Beam lifetime optimisation
6. Conclusions and Outlook
LHC Protection System

- The LHC uses a system of **100 collimators** for protection.
- These must be **aligned** around the two beams with a precision better than 50\(\mu\)m.
- Alignments are performed yearly before start of operation.

Applications with Machine Learning

- Alignment is tedious, time-consuming, and repetitive.
- Ideal situation for machine learning.
  \[\Rightarrow \text{supervised learning}\]
Setup

Beam Loss Monitors (BLMs)
- record **losses** as they touch the beam
- experts monitor these losses to deduce collimator **alignment**

Spikes
- data sample taken when collimator stops moving
- spike when threshold in BLM is passed
- goal is to distinguish **real** spikes (beam is hit) from noise
Spike Classification

Approach

- Six ML models for spike classification were compared
  - Logistic Regression, Neural Network, SVM, Decision Tree, Random Forest, Gradient Boost
- Data (8706 samples) split into: 85% training, 15% testing
- Enforce: no false positives
- False negatives are OK (because alignment will continue)
- No retraining needed unless hardware changes
- Analysis of beam crosstalk allows parallel alignments
Results

- Run I:
  - 2010: 20.5 hours
  - 2011: 17.5 hours
  - 2012: 12.5 hours

- Run II:
  - 2015: 5.5
  - 2016: 2.9
  - 2017: 2.83
  - 2018: 1.5
  - 2018 Parallel: 0.83

ML

- 79 collimators in 50 minutes!
Results

- ML can replace human operators for alignments
- More than three times faster!
- ML-based alignment will be **default** from now on
Outline

1. Introduction
2. Collimator alignment
3. Optics measurements and corrections
4. Dynamic aperture studies
5. Beam lifetime optimisation
6. Conclusions and Outlook
Context

Analysis of Beam Optics

- Beam Position Monitors (BPMs) measure excited beam
- Faulty BPMs give unphysical values for optics functions

Applications with Machine Learning

- **Identify** and remove faulty BPMs from data
  → anomaly detection by *unsupervised learning*
- **Correct** \( \beta \)-beating
  → *supervised learning*
Anomaly Detection

Approach

- Past measurements show that \( \sim 10\% \) of BPMs are faulty.
- Non-physical spikes in optics are artefact of bad BPMs.
  - Use ML to identify faulty BPMs from harmonic analysis.
  - To avoid spikes in optics functions.
- Enforce: no false negatives (don’t keep a bad BPM).
- False positives are OK (we have >1000 BPMs...)

- Four ML algorithms are compared:
  - K-means, DBSCAN, Local Outlier Factor, Isolation Forest.
Anomaly Detection

Results

- EPS-HEP 2019 ML @ LHC

- GOETHE
  UNIVERSITÄT
  FRANKFURT AM MAIN

- CERN

- EPS-HEP 2019
  ML @ LHC

- 8/21
Anomaly Detection

Results

- Very good recognition of faulty BPMs
- Now integrated by default into optics measurements at LHC
- Successfully used during commissioning and machine developments
Correction of Beta-Beating

Setup

- $\beta$-function calculated from harmonic analysis of BPMs
- $\beta$-beating is ratio of measured over designed $\beta$-function
- Corrections in the LHC are based on response matrix

Machine Learning (work in progress)

ML to replace response matrix $\Rightarrow$ supervised learning
Correction of Beta-Beating

**Approach**

- Ideal optics
- 190 errors *in quad circuits* (not in a single quadrupole)
- Perturbed optics
- Phase advance measured at 1046 BPMs from ideal optics
- Phase advance measured at 1046 BPMs from perturbed optics

**Correlation!**

- MAD-X
- Difference

ML
Correction of Beta-Beating

Three ML algorithms are compared:

- Convolutional Neural Network, Linear Regression, Ridge
- CNN (Keras with TensorFlow backend):
  - Used for image processing
  - Spatially dependent features: phase advance between neighbouring BPMs
  - Different deep layers look for different features
- Very simple model is applied: no parameter tuning, no optimisation

⇒ Lots of improvements are possible
Correction of Beta-Beating

Results

![Graph showing uncorrected and corrected beta-beating with CNN and RM compared to BPM index]
Correction of Beta-Beating

Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Peak</th>
<th>RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uncorrected</td>
<td>32±10</td>
<td>11±3</td>
</tr>
<tr>
<td>Response Matrix</td>
<td>11±5</td>
<td>3±2</td>
</tr>
<tr>
<td>CNN</td>
<td>11±2</td>
<td>3.2±0.5</td>
</tr>
<tr>
<td>Ridge regression</td>
<td>10±2</td>
<td>2.9±0.8</td>
</tr>
<tr>
<td>Linear regression</td>
<td>9±2</td>
<td>2.6±1.7</td>
</tr>
</tbody>
</table>

- All methods demonstrate similar performance
- Linear Regression ML achieves best correction
Outline

1. Introduction
2. Collimator alignment
3. Optics measurements and corrections
4. Dynamic aperture studies
5. Beam lifetime optimisation
6. Conclusions and Outlook
Context

Dynamic Aperture

- A tool to estimate **beam quality**
- It is the volume of the smallest connected region in phase space that remains **stable** for a certain amount of time
- Its **evolution** over time can be estimated with scaling laws
- DA can describe **beam losses** and **luminosity evolution**

Applications with Machine Learning

- Anomaly detection ⇒ **unsupervised learning**
- DA extrapolation ⇒ **supervised learning**
Anomaly Detection

Setup
- 60 random realisations (‘seeds’) in LHC simulations
- Sometimes one seed gives very bad DA for one angle (because close to resonance, internal cancellations, ...)

Machine Learning
- Use ML to flag these outliers
  → let human decide whether or not to remove
- Investigate anomaly dependence on angles or seeds
Anomaly Detection

Approach

Points are sometimes clustered in several groups

⇒ **DBSCAN** to recognise clusters

(scaled over population, min 3 points in a cluster)

points not in cluster are possible outliers

⇒ **LOF** to quantify outlier strength

⇒ Cut off at **minimum threshold**, and outliers can only exist as minima or maxima (not in between)
Anomaly Detection

Results

Anomaly Detection

Results
Anomaly Detection

Results

[Graph showing anomaly detection results with green checkmark and red question mark.]
Anomaly Detection

Results

DBSCAN

Frequency of anomaly vs Angle

DBSCAN

Frequency of anomaly vs Seed number
Anomaly Detection

Results

![DBSCAN Chart](chart1)

![DBSCAN Chart](chart2)
Anomaly Detection

<table>
<thead>
<tr>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Outlier detection per angle works as expected</td>
</tr>
<tr>
<td>But human verification is indeed needed!</td>
</tr>
<tr>
<td>→ to decide whether or not to remove a particular seed</td>
</tr>
<tr>
<td>(depending on behaviour of nearby angles)</td>
</tr>
<tr>
<td>• $\approx 10 \times$ more outliers at large angles and seeds 1 and 52</td>
</tr>
<tr>
<td>$\Rightarrow$ further investigation needed</td>
</tr>
</tbody>
</table>
Curve Fitting and Extrapolation

Setup
- DA simulations are very CPU-intensive
  - only $10^5 - 10^6$ turns ($\sim$ 1 minute) are achievable
- Realistic timescales are much larger ($\sim$ 10 hours)
  - simulations need to be extrapolated
- Scaling laws exist to describe evolution over time

Machine Learning (work in progress)
- Use ML to improve fitting to scaling laws
- Recurrent Neural Network to make prediction estimates
  (Well-suited to predicted sequential / time-series data)
Curve Fitting and Extrapolation

**Approach**

- Existing scaling laws work well to **describe** the data.
- But not that much to **predict** (sensitivity of fit parameters).

![Graph showing dynamic aperture vs. number of turns with two fitted curves up to different turn counts.](image-url)
Curve Fitting and Extrapolation

Trying with a Neural Network

- Brute-force approach: not including any info from scaling
- Time series analysis (LSTM with Keras)
Curve Fitting and Extrapolation

Trying with a Neural Network

- Results aren’t very impressive; deeper investigation is needed
- Alternative: use a Neural Network to find optimal **weights** to fit to existing scaling laws
Outline

1. Introduction
2. Collimator alignment
3. Optics measurements and corrections
4. Dynamic aperture studies
5. Beam lifetime optimisation
6. Conclusions and Outlook
Context

Beam Lifetime

- Is the time $\tau$ such that intensity $I(\tau) = \frac{1}{e} I_0$
- **Real-life** counterpart of DA, describing **beam quality**
- Strongly influenced by **operational settings**
- Extraction from simulation is difficult (coherent instabilities)

Applications with Machine Learning

- Avoid time- and CPU-consuming tracking simulations
- Model that directly relates **lifetime** to **machine settings**
- Ample data available, focus on **2017 and 2018**
  $\Rightarrow$ **supervised learning**
Beam Lifetime Model

**Approach**

**Input:**
- tunes (H/V, B1/B2)
- sextupole strengths (B1/B2)
- elapsed time
- number of bunches (B1/B2)
- emittances (H/V, B1/B2)
- octupole strength (B1/B2)
- timestamps
- ...

**Output:**
- beam lifetimes (B1/B2, from slope of BCTs)

**Data from Run 2**
Beam Lifetime Model

Result: Prediction of Lifetime (with LightGBM algorithm)

Fill 7056
Beam 2
Beam Lifetime Model

Result: Prediction of Lifetime (with LightGBM algorithm)

Fill 7056
Beam 2
Good agreement!
Multi-Parameter Optimisation

Optimal Settings

- Close to resonances: highest lifetime
- However this also gives emittance blow-up
- Latter is unwanted as it decreases luminosity

⇒ Multi-objective optimisation problem
Multi-Parameter Optimisation

Optimal Settings

Graphs showing various parameters over time.
Multi-Parameter Optimisation

Optimal Settings

Beam 1 recommended settings:

\[ q_x = 0.279 \]
\[ q_y = 0.286 \]

Beam 2 similar
Outline

1. Introduction
2. Collimator alignment
3. Optics measurements and corrections
4. Dynamic aperture studies
5. Beam lifetime optimisation
6. Conclusions and Outlook
Conclusions

- **Collimator Alignment:**
  - ML is now the standard tool for collimator alignments

- **Optics Measurements and Correction:**
  - ML is now the standard tool to find faulty BPMs
  - First steps are made to use ML as an alternative for the response matrix

- **Dynamic Aperture:**
  - Anomaly detection is very efficient

- **Beam Lifetime:**
  - First steps are made towards a model that predicts lifetime in function of the operational parameters
Outlook

- **Collimator Alignment:**
  - Advanced crosstalk analysis → more alignments in parallel

- **Optics Correction:**
  - Larger dataset → more general model
  - Add more sources of errors and non-linearities
  - Reinforcement Learning

- **Dynamic Aperture:**
  - Anomaly detection by centralised supervised learning
  - Improve prediction algorithms using high-precision data
  - Use supervised learning on fitting weights

- **Beam Lifetime:**
  - Larger dataset and more operational parameters
    → more general model
Backup Slides Collimator Alignments
Setup

Spike Parameterisation

5 parameters:

jaw position (1), spike height (1), and decay fit (3)
Backup Slides Beam Lifetimes
Internal Correlations

Correlations

Lifetimes depend on tunes from both beams!

⇒ Need to de-correlate before continuing
Internal Correlations

Correlations

Lifetimes depend on tunes from both beams!

⇒ Need to de-correlate before continuing
Robustness of Model

**Machine Development**

- Use dedicated MD run:
  - to decorrelate tunes between two beams
  - to extend tune range further than only current operational settings
- This allows us to test robustness of model:
  - does the tunes correlation matter?
  - behaviour of other beam parameters when lifetime is large?
Robustness of Model

Machine Development

- random walk over tunes
- different random walk for beam 2 at the same moment
- do this for different operational settings
Robustness of Model

Machine Development

No more correlations between beams!

However, emittance becomes important