Hysteresis Modeling in Iron-Dominated Magnets Based on a Multi-Layered NARX Neural Network Approach

Vincenzo Di Capua – IMMW22

Date: 26-30/09/2022

Outline

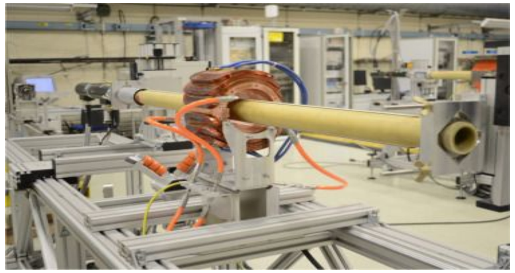
- Problem
- Machine learning to predict magnetic field
- Case study
- Proposal
- System identification
- Results
- Summary
- Future plans

Problem

Magnet Design → Magnet Manufacture → Machine Operation

Offline magnetic measurements and simulations

B-Train systems



Role of measurements and simulations in magnetic design, manufacture and operations

Machine learning to predict magnetic field

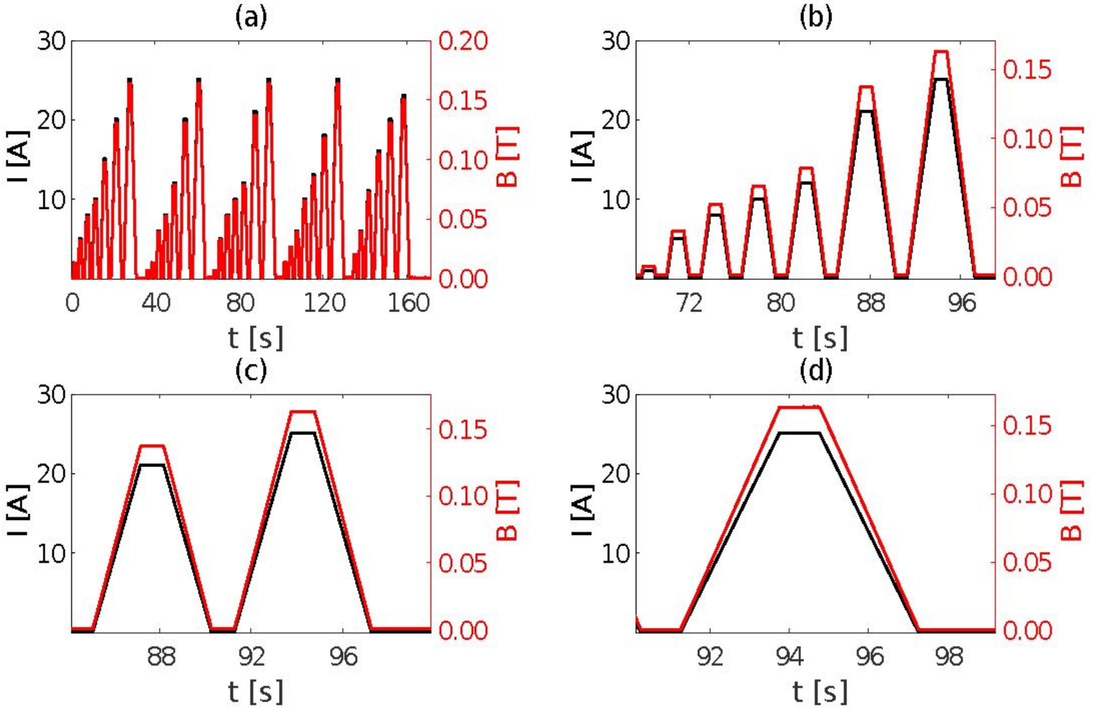
But why machine learning ?

- The operation of synchrotrons requires knowledge of the magnetic field within a typical **tolerance of 0.01%** at any given time during a magnetic cycle. At CERN, the field of the bending dipoles of the Large Hadron Collider (LHC) is predicted by the **field description for the LHC (FiDeL)** semi-empirical mathematical model [1].

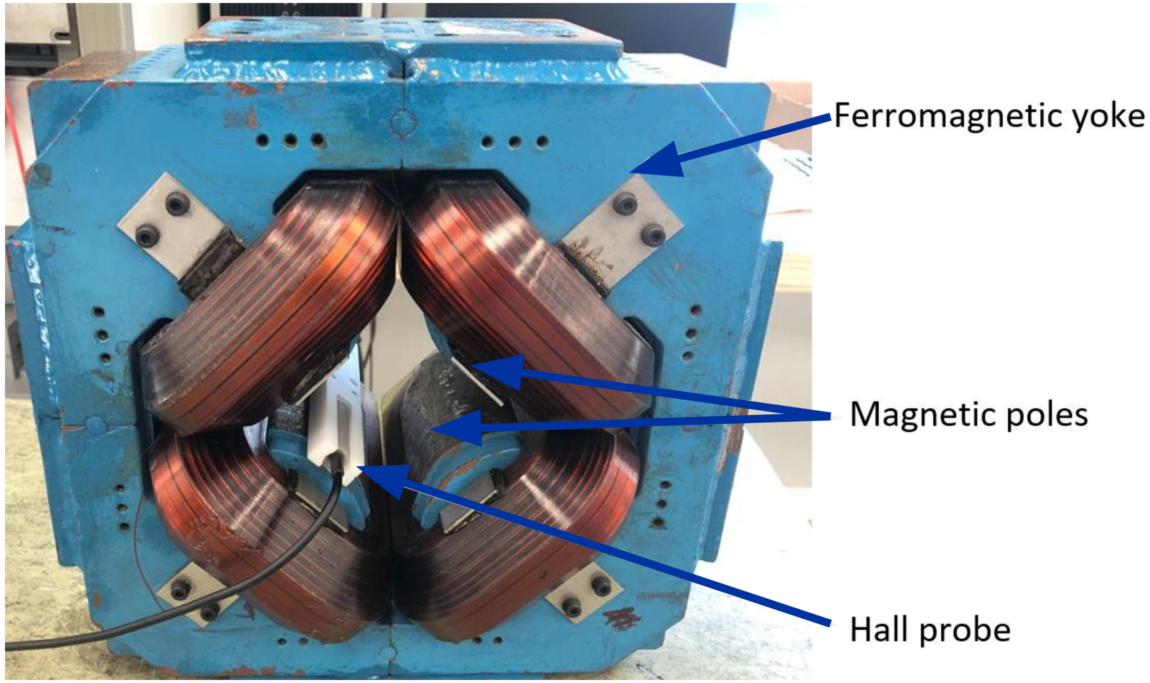
- In the **LHC injectors**, on the other hand, the **current-to-field characteristic** of the magnets is **dominated by the iron core**, which gives rise to non-linear effects such as eddy currents, saturation and hysteresis.

- Recent attempts at mathematical prediction using **analytical expressions** [2] or **Preisach models** [3] could not attain better than percent-level accuracy, which is inadequate for normal operation. Other classes of methods, such as **Jiles-Atherton differential models**, were considered but ultimately deemed unsuitable, with particular respect to their difficulties in handling minor hysteresis loops.

Case study

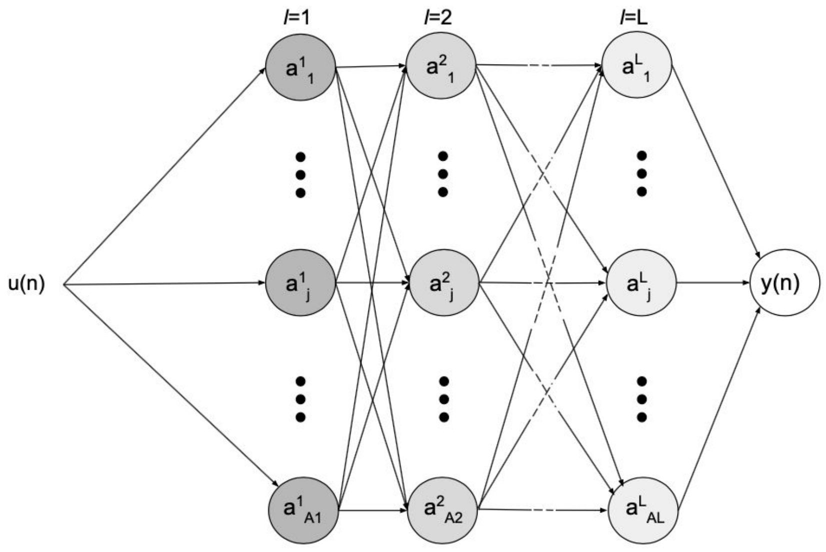


Training dataset



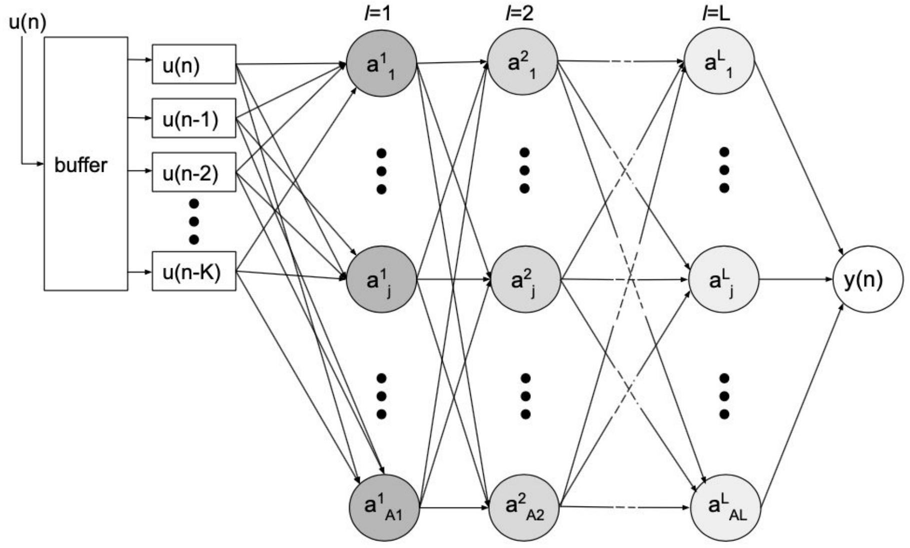
Correction quadrupole used as a case study

Proposal



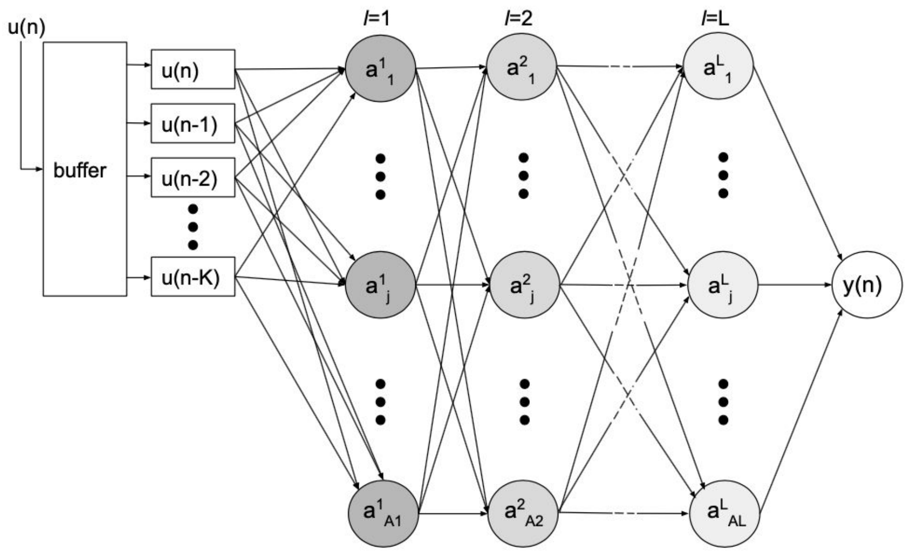
Multi layer Perceptron (MLP)

- MLP retains no time information
- How can we introduce this information ?



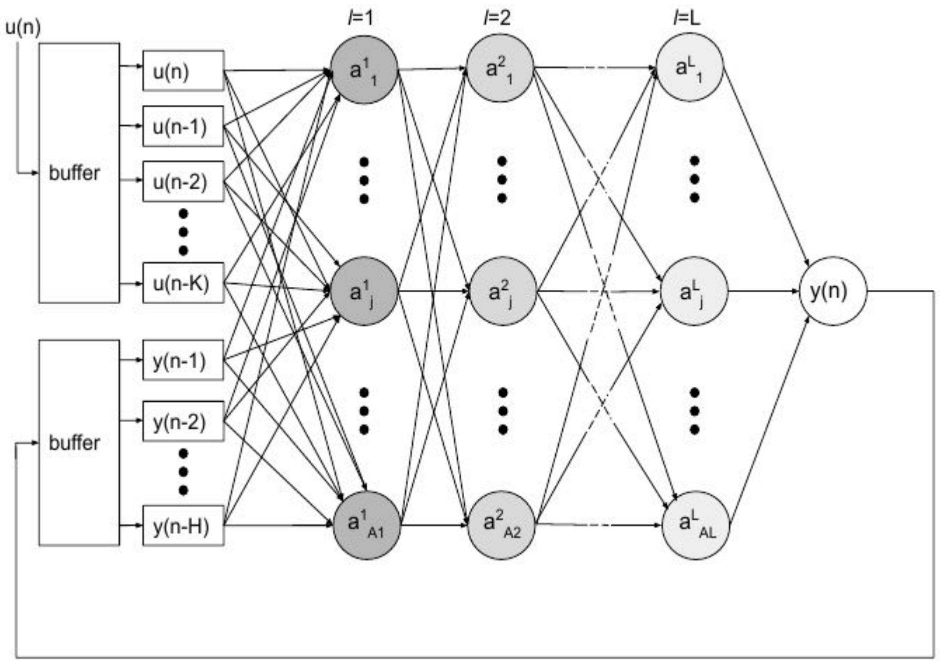
Time Delayed Neural Network

Proposal



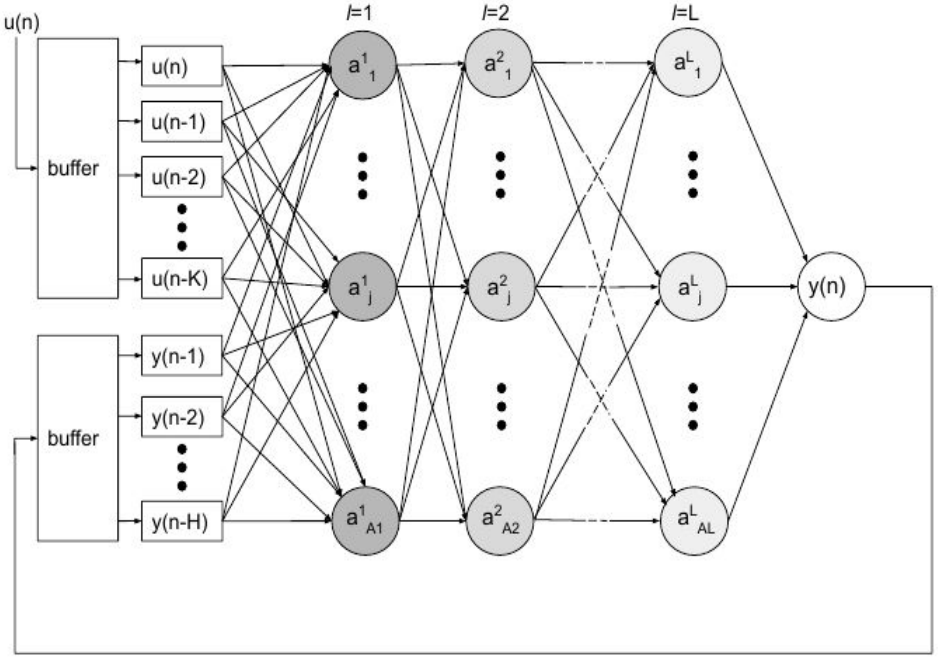
Time Delayed Neural Network

- Still no information about the previous outputs of the network
- How can we introduce this information ?



Non Linear Autoregressive Exogenous Input (NARX) Neural Network

Proposal



Non Linear Autoregressive Exogenous Input (NARX) Neural Network

Mathematical description

$$a_j^1(n) = \phi^{A_1}\left(\sum_{h=1}^{H} W_{jh}^{O} y(n-h) + \sum_{k=0}^{K} W_{jk}^{I} u(n-k)\right)$$

$$a_j^l(n) = \phi^{A_l}\left(\sum_{i=0}^{A_l} W_{ji}^{l} a_i^{l-1}(n)\right)$$

$$y(n) = \sum_{i=0}^{A_{L-1}} W_i^E a_i^L(n)$$

- φ^{A1} is the activation function of the neurons of the layer (*tanh* in this case)
- W^l_{ji} are the weights of the interconnections between neurons.

System identification

Hyperparameters definition for the model selection

Neural Network Model	Multilayer Perception	Time Delay	Autoregressive Exogenous
Hyperparameter	θ_{MLP}	θ_{TDNN}	θ_{NARX}
L	$\{1,\ldots,15\}$	$\tilde{L}_{MLP}$	$\tilde{L}_{MLP}$
$\mathbf{A}$	$\{1,\ldots,10\}^L$	$\tilde{\mathbf{A}}_{MLP}$	$\tilde{\mathbf{A}}_{MLP}$
K	0	$\{1,\ldots,35\}$	$\tilde{K}_{TDNN}$
H	0	0	$\{1,\ldots,35\}$

Bridge Criterion (BC) for the model selection

$$RMSE(y_i, D^{\text{test}}) = \sqrt{\frac{\sum_{n\in N}(y_i(n) - B^{\text{test}}(n))^2}{|N|}}$$

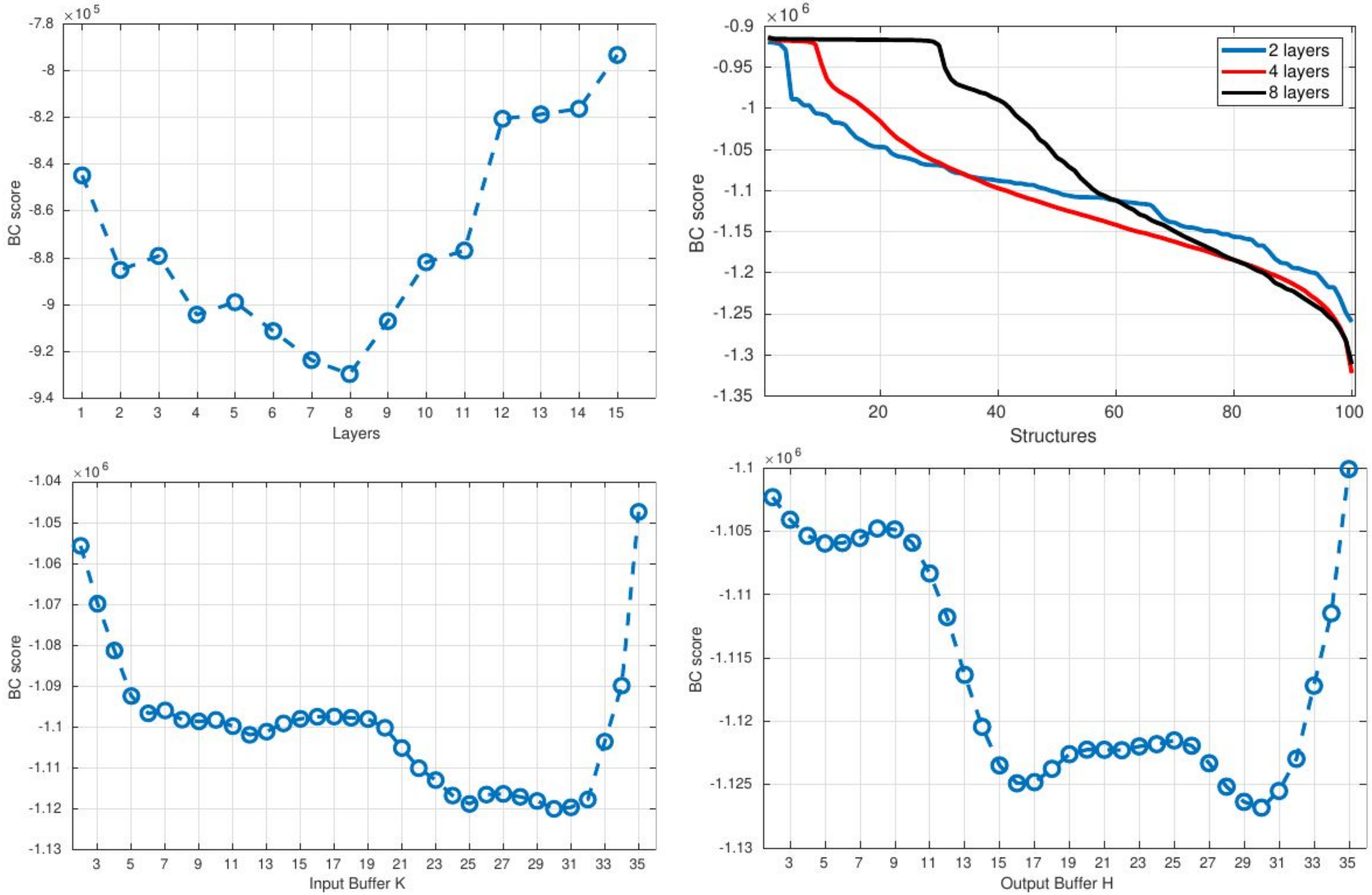
$$Score(j) = |N| \ln\left(\frac{1}{R}\sum_{i=1}^{R} Error^2(i)\right) + |N|^{2/3} \cdot (1 + 1/2 + \ldots + 1/|W|)$$

- The **minimization of the BC** term corresponds to the **maximization of the evidence** of the j^{th} model, ensuring a **balanced model fit**.

System identification

Four steps model selection

compromise between performance and complexity



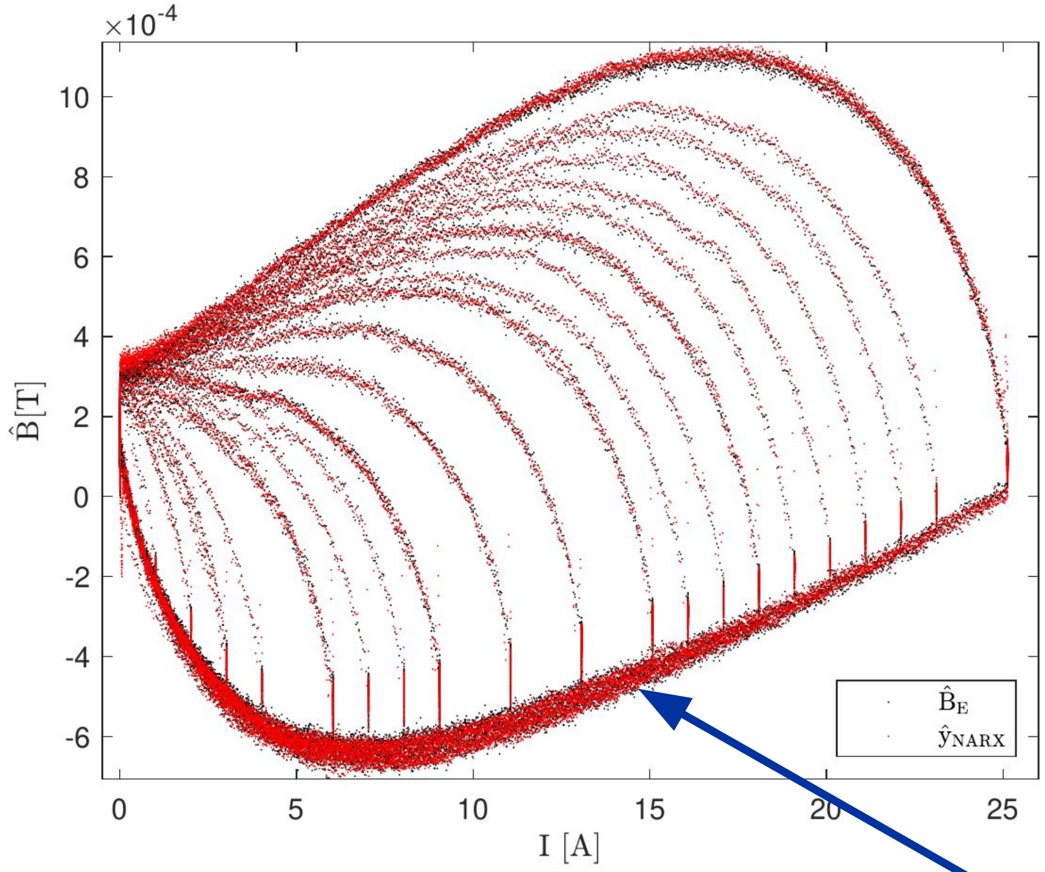
System identification

Four steps model selection outcomes

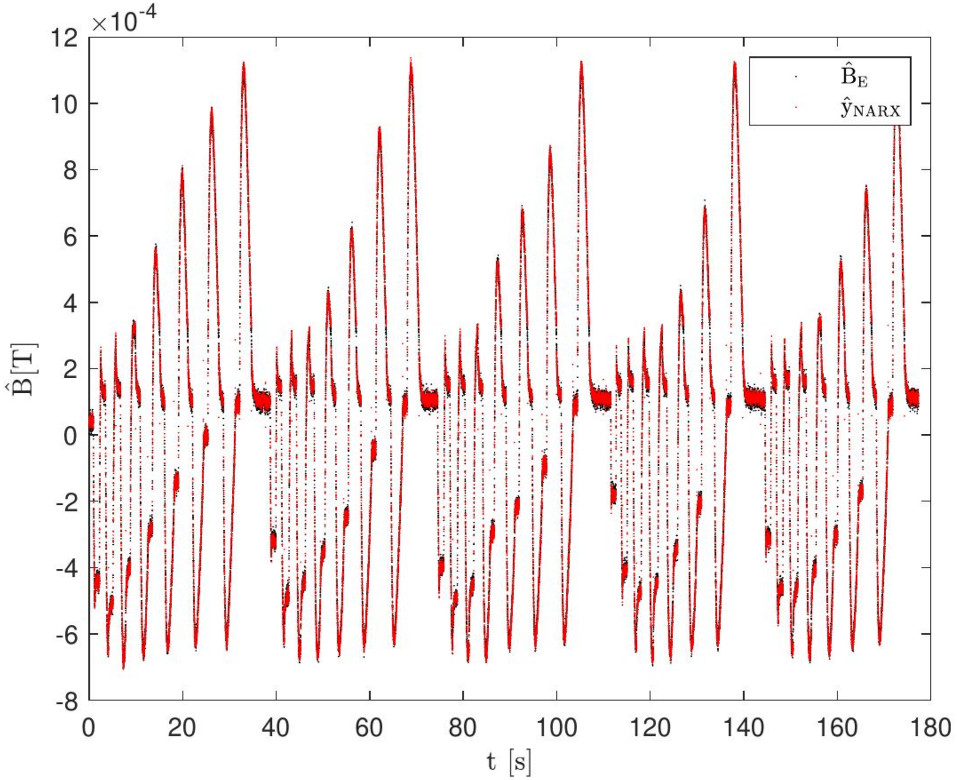
compromise between performance and complexity

Hyperparameters	L	$\mathbf{A}$	K	H	$\|W\|$
$\tilde{\theta}_{MLP1}$	2	$(10, 9)$	0	0	129
$\tilde{\theta}_{MLP2}$	4	$(1, 1, 1, 10)$	0	0	37
$\tilde{\theta}_{TDNN1}$	2	$(10, 9)$	26	0	379
$\tilde{\theta}_{TDNN2}$	4	$(1, 1, 1, 10)$	31	0	67
$\tilde{\theta}_{NARX1}$	2	$(10, 9)$	26	31	689
$\tilde{\theta}_{NARX2}$	2	$(7, 8)$	26	17	381
$\tilde{\theta}_{NARX3}$	4	$(1, 1, 1, 10)$	31	31	98
$\tilde{\theta}_{NARX4}$	4	$(1, 8, 5, 4)$	31	31	153

Results



Hysteresis comparison measured vs predicted (by NARX4)



Measured vs predicted magnetic field of the non-linear component of the field

20 ppm prediction accuracy

Results

Some more performance indicators to facilitate the comparison with the requirements

Normalized RMSE

$$NRMSE(y, D_{\mathrm{E}}) = \frac{RMSE(y, D_{\mathrm{E}})}{B_{\max}} \cdot 100$$

Maximum Absolute Error

$$MAE(y, D_{\mathrm{E}}) = \max\left\{|y(n) - B_{\mathrm{E}}(n)|\right\}_{n \in N}$$

Maximum Percentage Error

$$MPE(y, D_{\mathrm{E}}) = \frac{MAE(y, D_{\mathrm{E}})}{B_{\max}} \cdot 100$$

Results

Performance comparison among the different architectures, trained and computed on the test dataset at the decimated data rate of 250 S/s.

Architecture	Test Dataset	Hyper- -parameters	RMSE [T]	NMRSE (%)	MAE [T]	MPE (%)
Linear Regression		G, B_0	$9.07 \cdot 10^{-04}$	$5.67 \cdot 10^{-01}$	$2.60 \cdot 10^{-03}$	1.61
MLP1	D_{E}	$\tilde{\theta}_{MLP1}$	$8.70 \cdot 10^{-04}$	$5.44 \cdot 10^{-01}$	$2.60 \cdot 10^{-03}$	1.64
MLP2	D_{E}	$\tilde{\theta}_{MLP2}$	$8.83 \cdot 10^{-04}$	$5.52 \cdot 10^{-01}$	$2.50 \cdot 10^{-03}$	1.55
TDNN1	D_{E}	$\tilde{\theta}_{TDNN1}$	$7.95 \cdot 10^{-04}$	$4.97 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.21
TDNN2	D_{E}	$\tilde{\theta}_{TDNN2}$	$8.05 \cdot 10^{-04}$	$5.03 \cdot 10^{-01}$	$2.20 \cdot 10^{-03}$	1.39
NARX1	D_{E}	$\tilde{\theta}_{NARX1}$	$2.12 \cdot 10^{-05}$	$1.32 \cdot 10^{-02}$	$3.36 \cdot 10^{-04}$	$2.10 \cdot 10^{-01}$
NARX2	D_{E}	$\tilde{\theta}_{NARX2}$	$2.13 \cdot 10^{-05}$	$1.33 \cdot 10^{-02}$	$3.17 \cdot 10^{-04}$	$1.98 \cdot 10^{-01}$
NARX3	D_{E}	$\tilde{\theta}_{NARX3}$	$2.05 \cdot 10^{-05}$	$1.28 \cdot 10^{-02}$	$3.11 \cdot 10^{-04}$	$1.95 \cdot 10^{-01}$
NARX4	D_{E}	$\tilde{\theta}_{NARX4}$	$2.05 \cdot 10^{-05}$	$1.28 \cdot 10^{-02}$	$3.12 \cdot 10^{-04}$	$1.95 \cdot 10^{-01}$
LR+MLP1	D_{E}	$G, B_0, \tilde{\theta}_{MLP1}$	$8.00 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.16
LR+MLP2	D_{E}	$G, B_0, \tilde{\theta}_{MLP2}$	$8.00 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.17
LR+TDNN1	D_{E}	$G, B_0, \tilde{\theta}_{TDNN1}$	$8.05 \cdot 10^{-04}$	$5.03 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.21
LR+TDNN2	D_{E}	$G, B_0, \tilde{\theta}_{TDNN2}$	$7.97 \cdot 10^{-04}$	$4.98 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.20
LR+NARX1	D_{E}	$G, B_0, \tilde{\theta}_{NARX1}$	$7.99 \cdot 10^{-04}$	$4.99 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.19
LR+NARX3	D_{E}	$G, B_0, \tilde{\theta}_{NARX2}$	$7.99 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.19
LR+NARX3	D_{E}	$G, B_0, \tilde{\theta}_{NARX3}$	$8.00 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.20
LR+NARX4	D_{E}	$G, B_0, \tilde{\theta}_{NARX4}$	$8.01 \cdot 10^{-04}$	$5.01 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.19

Machine learning to predict magnetic field

Performance comparison among the different architectures, trained on the test dataset at the decimated data rate of 250 S/s. and computed at the full data rate of 2.5 kS/s

Architecture	Test Dataset	Hyper- -parameters	RMSE [T]	NMRSE (%)	MAE [T]	MPE (%)
Linear Regression		G, B_0	$9.07 \cdot 10^{-04}$	$5.67 \cdot 10^{-01}$	$2.60 \cdot 10^{-03}$	1.61
MLP1	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{MLP1}$	$8.70 \cdot 10^{-04}$	$5.44 \cdot 10^{-01}$	$2.60 \cdot 10^{-03}$	1.64
MLP2	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{MLP2}$	$8.83 \cdot 10^{-04}$	$5.52 \cdot 10^{-01}$	$2.50 \cdot 10^{-03}$	1.55
TDNN1	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{TDNN1}$	$1.10 \cdot 10^{-03}$	$6.66 \cdot 10^{-01}$	$2.70 \cdot 10^{-03}$	1.67
TDNN2	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{TDNN2}$	$8.52 \cdot 10^{-04}$	$5.32 \cdot 10^{-01}$	$2.50 \cdot 10^{-03}$	1.56
NARX1	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{NARX1}$	$9.92 \cdot 10^{-06}$	$6.20 \cdot 10^{-03}$	$4.63 \cdot 10^{-05}$	$2.89 \cdot 10^{-02}$
NARX2	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{NARX2}$	$1.27 \cdot 10^{-05}$	$8.00 \cdot 10^{-03}$	$6.90 \cdot 10^{-05}$	$4.31 \cdot 10^{-02}$
NARX3	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{NARX3}$	$9.22 \cdot 10^{-06}$	$5.80 \cdot 10^{-03}$	$3.98 \cdot 10^{-05}$	$2.49 \cdot 10^{-02}$
NARX4	$\bar{D}_\mathrm{E}$	$\tilde{\theta}_{NARX4}$	$9.28 \cdot 10^{-06}$	$5.80 \cdot 10^{-03}$	$4.06 \cdot 10^{-05}$	$2.54 \cdot 10^{-02}$
LR+MLP1	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{MLP1}$	$8.00 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.16
LR+MLP2	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{MLP2}$	$8.00 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.17
LR+TDNN1	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{TDNN1}$	$8.05 \cdot 10^{-04}$	$5.03 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.18
LR+TDNN2	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{TDNN2}$	$7.97 \cdot 10^{-04}$	$4.98 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.18
LR+NARX1	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{NARX1}$	$7.98 \cdot 10^{-04}$	$4.99 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.17
LR+NARX2	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{NARX2}$	$7.98 \cdot 10^{-04}$	$4.99 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.17
LR+NARX3	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{NARX3}$	$7.99 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.17
LR+NARX4	$\bar{D}_\mathrm{E}$	$G, B_0, \tilde{\theta}_{NARX4}$	$8.00 \cdot 10^{-04}$	$5.00 \cdot 10^{-01}$	$1.90 \cdot 10^{-03}$	1.17

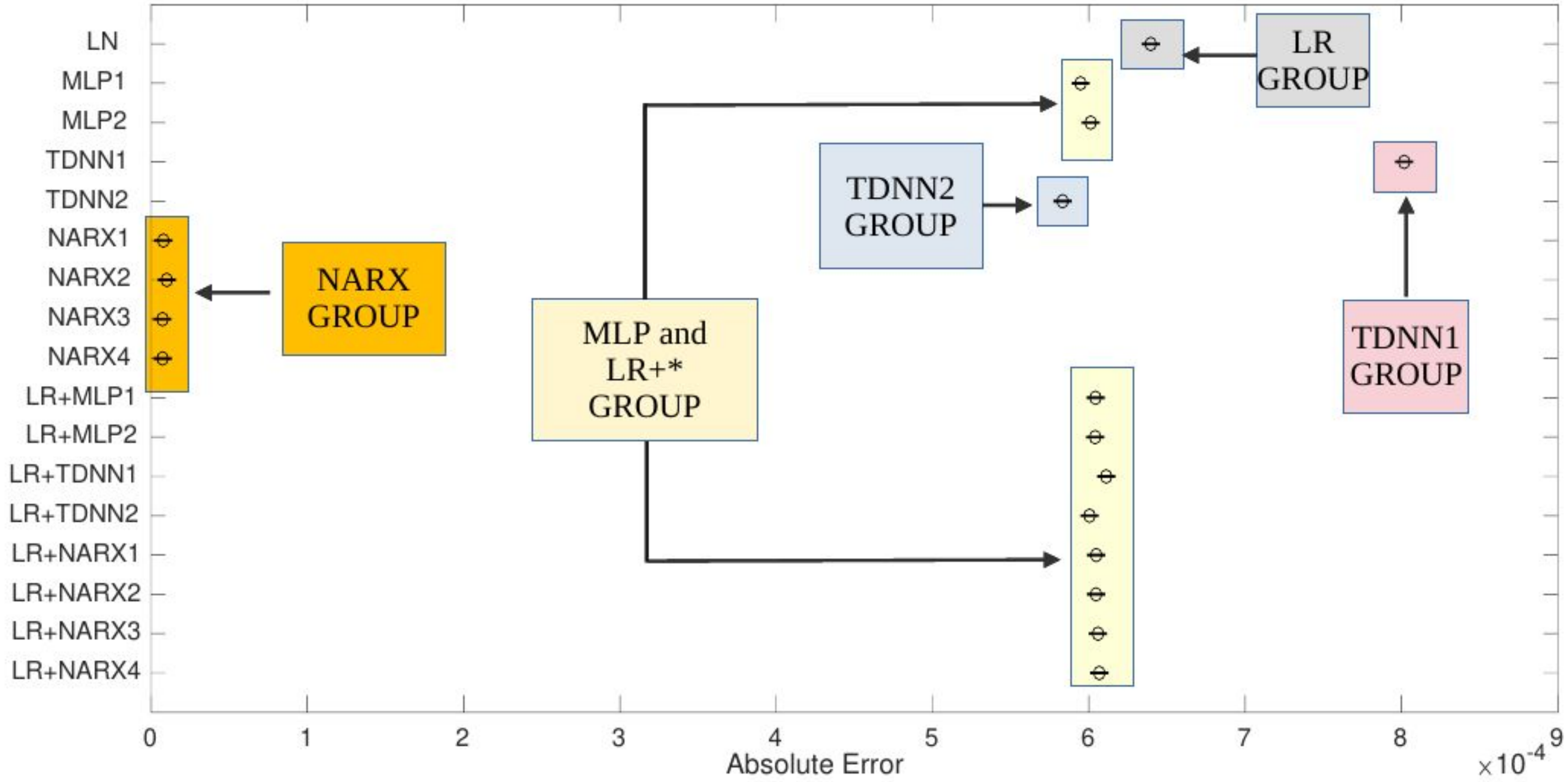
Results

Literature comparison

https://www.worldscientific.com/doi/pdf/10.1142/S0129065721500337

Architecture	Metric	Value
Deep Neural Network (MLP with two hidden layers), Ref. [61]	Root Mean Square Error	0.13 %
Preisach + Feed-forward neural network (one hidden layer), Ref. [59]	Maximum Absolute Error	13 %
Preisach, Ref. [15]	Relative Error	0.2 %
Preisach + Recurrent Neural Network, Ref. [72]	Normalized Root Mean Square Error	0.7 %
Neural Network, Ref. [73]	Relative Error	< 8 %
Genetic Algorithm + Neural Network, Ref. [76]	Mean Square Error	< 5 %
Proposed architecture NARX4	Normalized Root Mean Square Error	$5.80 \cdot 10^{-3}$ %

Results



ANOVA results on Absolute Errors computed for the competing models on Dataset DE . The horizontal lines are the 95 % confidence interval for each model. Five groups are highlighted: LR group (gray), MLP group and LR+* group (yellow), TDNN1 (pink), TDNN2 (in blue) and the NARX group (orange)

Summary

- The operation of synchrotrons requires knowledge of the magnetic field within a typical tolerance of 0.01%;
- A novel neural network based algorithm to predict the magnetic field and its non-linearities starting from the input current have been developed;
- Experimentally tested in conditions representative of those found in particle accelerators and similar, pulsed-mode machines;
- NARX networks achieve in general the required level of performance i.e. an NRMSE better than 0.01%;
- Prediction accuracy generally improves when the network is trained on low data rate (250 S/s) signals and tested at higher data rate (2.5 kS/s).

Future plans

- Increase the variety of training and test waveforms;
- Extend the range of excitation currents;
- Test the presented approach on other kinds of magnets;
- Real time implementation of the NN approach to integrate it in the control system of LHC injectors;
- Exploit physics informed neural network approaches to further increase the reliability of the architecture.

Thanks for the attention

Any questions?

home.cern