ISBN: 978-3-95450-253-0 ISSN: 2673-5571 doi:10.18429/JACoW-HB2023-THAFP09

OPTIMIZING BEAM DYNAMICS IN LHC WITH ACTIVE DEEP LEARNING

D. Di Croce[1,*], M. Giovannozzi[2], E. Krymova[3], T. Pieloni[1], M. Seidel[1,4], F. F. Van der Veken[2]
[1]École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
[2]CERN, Geneva, Switzerland
[3]The Swiss Data Science Center, Zurich, Switzerland
[4]Paul Scherrer Institut, 5232 Villigen PSI, Switzerland

Abstract

The Dynamic Aperture (DA) is an important concept for the study of non-linear beam dynamics in a circular accelerator. It refers to the region in phase space where a particle's motion remains bounded over a given number of turns. Understanding the features of DA is crucial for operating circular accelerators, like the CERN Large Hadron Collider, as it provides insights on non-linear beam dynamics and the phenomena affecting beam lifetime. The standard approach to calculate the DA requires accurate numerical simulations to perform tracking of initial conditions distributed in phase space over a sufficient number of turns in a circular machine to understand the beam dynamics. This process is very computationally intensive. In our study, we aim at determining the evolution of beam stability for a set of machine parameters, like betatron tune, chromaticity, and Landau octupole strengths, i.e., the values that maximise the DA, using a Deep Neural Network (DNN) model. To enhance its performance, we integrated the DNN model into an innovative Active Learning (AL) framework. This framework not only enables the retraining and updating of the DNN model but also facilitates efficient data generation through smart sampling.

INTRODUCTION

The study of dynamic aperture (DA), defined as the extent of the connected phase-space region in which the single-particle dynamic is bounded, offers valuable insight into the non-linear beam dynamics of single particles and the underlying mechanisms contributing to beam losses [1]. The numerical calculation of the DA involves tracking a large number of initial conditions in phase space for many turns. This method is computationally demanding, especially for large accelerators such as the CERN Large Hadron Collider (LHC) [2, 3], and for this reason, analytical scaling laws have been studied for several years [4, 5].

In recent years, we have developed a Machine Learning (ML) model to quickly and accurately predict DA for unknown machine configurations [6]. To achieve this, we trained a Deep Neural Network (DNN) on a substantial dataset of simulated initial conditions, enabling it to capture the intricate relationship between the initial conditions and the resulting DA.

In this study, we integrate machine learning techniques into an Active Learning (AL) framework. Additionally, we introduce an error estimator alongside the DA model to gauge the uncertainty in the predictions. This enables the AL algorithm to perform smart sampling, i.e., it can determine which new machine configuration to simulate first, based on the size of the predicted DA error. This approach aims to facilitate the rapid estimation of DA and its associated error for new machine parameters. At the same time, it aims to expand the initial dataset, eventually improving the performance of the ML model in an efficient manner.

SIMULATED SAMPLES

To train the DNN, we simulated several accelerator configurations using MAD-X [7] and the 2023 LHC lattice at the injection configuration at 450 GeV. We varied six accelerator parameters, namely the betatron tunes Q_x, Q_y, chromaticities Q'_x, Q'_y, strength of the Landau octupoles (using the current, I_{MO}, powering them) and the realisations (also called seeds) of the magnetic field errors assigned to the various magnet families. Furthermore, both Beam 1 and Beam 2 have been considered in these studies. We performed a random uniform grid search of the following parameters: $Q_x \in [62.1, 62.5]$ and $Q_y \in [60.1, 60.5]$ both with steps of size 5×10^{-3}, 15 Q' values in $[0, 30]$, 17 I_{MO} in $[-40, 40]$, and 60 random realisations of the magnetic errors for both Beam 1 and Beam 2. The final dataset is made up of 10459 machine configurations.

The phase space was probed by tracking with XSuite [8, 9] for 10^5 turns. To speed up the tracking, we perform an initial scan of the initial conditions uniformly distributed in 8 polar angles in $[0, \pi/2]$ and 33 radial amplitudes in $[0.0, 20\sigma]$, to identify the value of the last stable amplitude for that angle (limit of the stable zone), as well as the first amplitude where the particle does not survive more than 10^3 turns (limit of the fast-loss zone). Then a finer scan of the initial conditions between the stable zone -2σ and the fast-loss zone $+2\sigma$ was performed, along with 44 polar angles in $[0, \pi/2]$ and 330 radial amplitudes in $[0.0, 20\sigma]$. An example of the results of these computations in the $x - y$ space is shown in Fig. 1 for a specific accelerator configuration.

The target of the ML regressor is the last stable amplitude for every angle (angular DA). To prevent extreme angular DA values from affecting the regressor [10], we cap values above 18 σ, as they are outliers in a distribution ranging from 0 σ to 20 σ. To gain insight into the evolution of beam stability,

* davide.dicroce@epfl.ch

ISBN: 978-3-95450-253-0 ISSN: 2673-5571 doi:10.18429/JACoW-HB2023-THAFP09

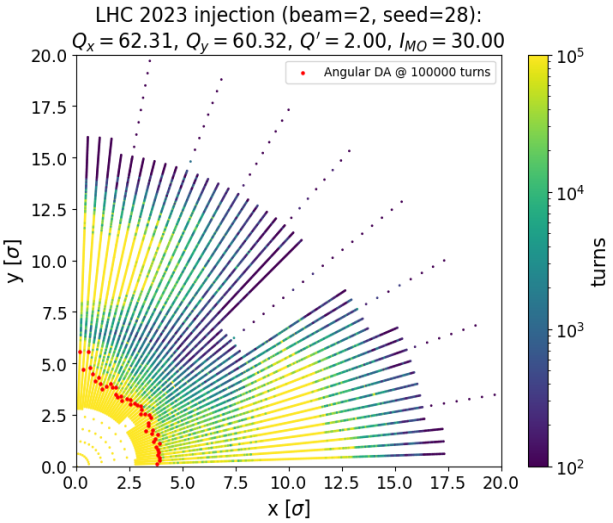

Figure 1: Stability time for a distribution of intial conditions used to compute the DA for a specific accelerator configuration. The angular DA at 10^5 turns is shown in red.

we analysed the angular DA as if we were tracking it for 12 different stability time limits in $[10^3, 10^5]$ turns.From the initials size of 10459 different machine configurations, by taking into account the variations in angles and turns, the dataset size has now expanded to 5.5 million samples. From this number, 10% of the samples were used for validation and 10% to test the performance of the model.

Moreover, other accelerator parameters, in addition to those from the machine grid scan, were added to the ML model: seven anharmonicities up to second order from PTC [11], the α and β maximum, and the phase advance ($\mu_{x,y}$) at IP5.

NETWORK ARCHITECTURE AND TRAINING

Similar to our previous study [6], we use a simple deep neural network with a concatenation layer to add the information of the discrete variables (beam and seeds). The network was developed using the TensorFlow library [12]. Architecture and hyperparameters were optimised by random search with the Keras Tuner framework [13]. The best model consists of four hidden layers with 2048, 1024, 512, and 128 nodes, respectively. Dropout (1%) was added between hidden layers to improve performance and avoid overfitting. The loss used for the regressor is the Mean Absolute Error (MAE) function and trained with the NADAM optimiser [14]. The initial learning rate is 5×10^{-5} and is halved every 10 sequential epochs if the validation loss is not improved. We found training for 364 epochs to be sufficient for validation loss convergence.

DA AND ERROR ESTIMATION

The MAE of the angular DA regressor is 0.351(0.351) beam σ for the test (train) dataset, and the Mean Absolute Percentage Error (MAPE) is 11.91(11.50)% for the test (train) dataset. These results indicates that the regressor is making relatively accurate predictions, with errors that are generally small compared to the range of angular DA values. In addition, analysis of the angular DA scatter histogram in Fig. 2 reveals that the model performs well for most of the data points, with a tight cluster around the diagonal line, indicating accurate predictions.

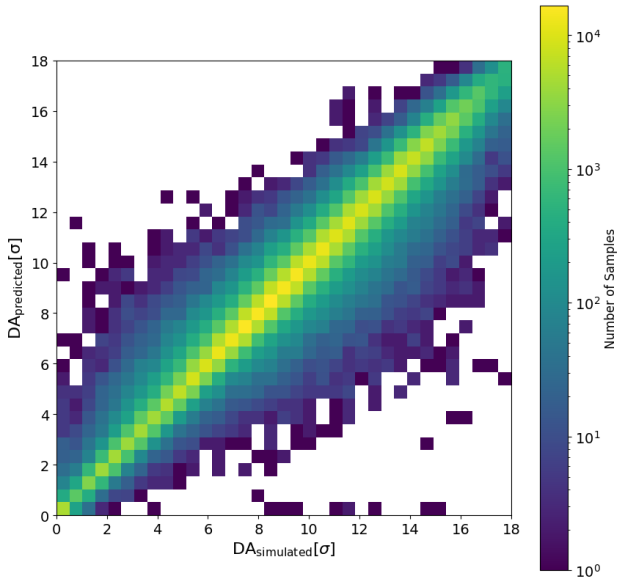

Figure 2: Angular DA predicted as a function of the expected angular DA values for the test data set.

During inference, we can introduce dropout, where some of the network's hidden units are randomly set to zero during forward propagation. This introduces variability in the predictions, which can be used to estimate uncertainty. This technique is called Monte Carlo (MC) dropout. Specifically, we applied a 1% dropout rate between the first two hidden layers during inference and conducted 128 inferences for each angular DA prediction. We then calculated the standard deviation of these predictions to determine uncertainty. Figure 3 illustrates the Percentage Absolute Errors (APE) distributions for both the true error (the actual angular DA prediction error) and the error predicted using the MC dropout.

The timing performance to predict the angular DA and its error of a batch of 1024 samples is 140 s (1.41 ms per inference) using a Titan V GPU [15], utilising 48 AMD Ryzen Threadripper 2970WX CPU cores for data loading. Taking into account a single machine configuration (12 survival turns × 44 angles), this results in an inference of 0.75 s/machine configuration.

ACTIVE LEARNING FRAMEWORK

By integrating the angular DA regressor and the error estimator, the AL framework is able to make intelligent sampling decisions based on the magnitude of the associated error. Specifically, if the error is small, the framework will utilise the DA predicted by the ML model. Examples of

ISBN: 978-3-95450-253-0 ISSN: 2673-5571 doi:10.18429/JACoW-HB2023-THAFP09

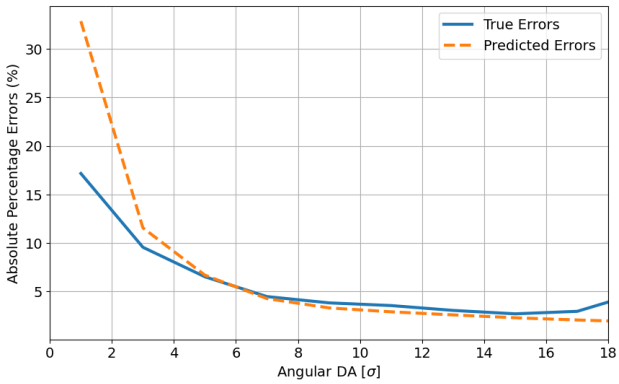

Figure 3: True (blue) and predicted (orange) APE in function of the angular DA.

angular DA reconstruction carried out by the AL framework for four accelerator configurations (available only in the test data set) with predicted APEs lower than 10% are shown in Fig. 4.

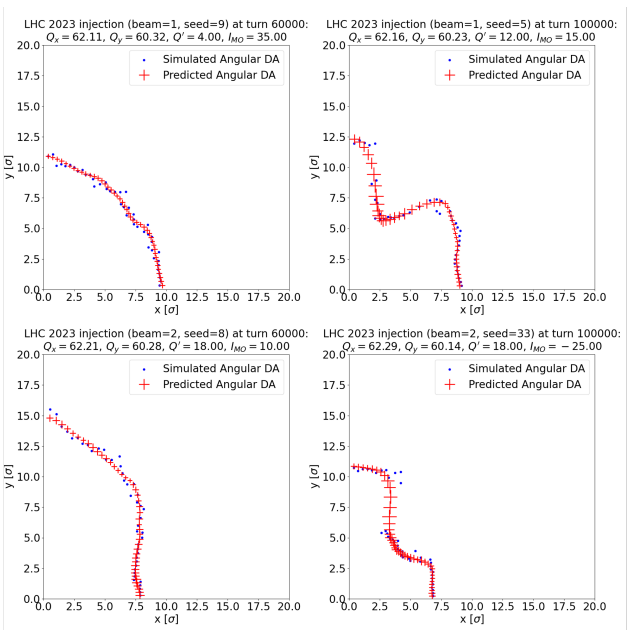

Figure 4: Simulated and predicted angular DA four configurations of the test data set (predicted APE lower than 10%).

Conversely, if the error is significant, it will opt for running the full simulation/tracking process to generate new data allowing the ML model to learn this specific feature. By prioritising predictions with higher errors, the framework efficiently determines the sequence in which to simulate different machine configurations (smart sampling). The pipeline of the AL framework is shown in Fig. 5.

Furthermore, leveraging the machine configurations estimated with small errors (APE lower than 10 %), we created a synthetic dataset comprising 1000 machine configurations. This synthetic dataset was then used to retrain the ML model. As a result of this update, the model's MAPE improved to 9.57% (9.61%) for the test (training) dataset.

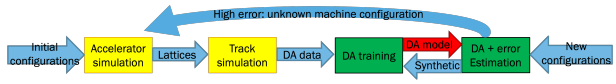

Figure 5: Pipeline of the AL framework. In yellow the full simulation/tracking algorithms and in green the ML algorithms.

Notable is the potential of deep learning to streamline DA evaluation, speeding up machine parameter optimisation. Furthermore, the AL framework intelligently incorporates traditional tracking methods, particularly when significant prediction errors are detected. In such cases, the framework initiates full simulations to ensure accuracy.

DISCUSSION

In this study, we achieved significantly better performance with a Mean Absolute Error (MAE) of 0.351 for the test dataset, compared to our previous study's MAE of 0.64 [6]. This improvement can be attributed to the use of a richer set of input variables, with the current model incorporating 21 machine variables, as opposed to the previous study's use of only 7. This boost in performance is thanks to our current model's incorporation of 21 machine variables, compared to the previous study's meager 7 [6], showcasing how expanding the input features significantly enhanced predictive accuracy.

The computational efficiency of our approach is a notable achievement. While the full simulation using MAD-X and XSuite, combined with the HTCondor system [16], took approximately 13 days to generate a dataset containing 10459 configurations (equivalent to 107 s/machine configuration). The AL framework, once trained, is approximately 140 times faster.

In future investigations, our aim is to create two distinct datasets: one generated using the AL smart sampler and another employing grid search. These datasets will serve as valuable resources for retraining the model and for comparative analysis, allowing us to assess the respective performance of these sampling methods.

CONCLUSIONS

Our study used machine learning, specifically leveraging the AL framework with DNN to accurately predict angular DA and its associated error in circular accelerators. Importantly, it demonstrated the capability to intelligently sample new machine configurations, enriching the dataset and thereby improving its performance. Additionally, it demonstrated exceptional speed in computation, establishing it as a powerful tool for optimising beam dynamics.

ACKNOWLEDGEMENTS

This work was carried out under the auspices and with the support of the Swiss Accelerator Research and Technology programme [17] and the Swiss Data Science Centre (SDSC).

ISBN: 978-3-95450-253-0 ISSN: 2673-5571 doi:10.18429/JACoW-HB2023-THAFP09

REFERENCES

[1] M. Giovannozzi, "A proposed scaling law for intensity evolution in hadron storage rings based on dynamic aperture variation with time," *Phys. Rev. Spec. Top. Accel. Beams*, vol. 15, p. 024 001, 2012. doi:10.1103/PhysRevSTAB.15.024001

[2] O. S. Brüning *et al.*, *LHC Design Report*. CERN, 2004. doi:10.5170/CERN-2004-003-V-1

[3] G. Apollinari *et al.*, *High-Luminosity Large Hadron Collider (HL–LHC)*. CERN, 2017, vol. 4. doi:10.23731/CYRM-2017-004

[4] M. Giovannozzi, W. Scandale, and E. Todesco, "Dynamic aperture extrapolation in the presence of tune modulation," *Phys. Rev. E*, vol. 57, pp. 3432–3443, 3 1998. doi:10.1103/PhysRevE.57.3432

[5] A. Bazzani, M. Giovannozzi, E. Maclean, C. Montanari, F. Van der Veken, and W. Van Goethem, "Advances on the modeling of the time evolution of dynamic aperture of hadron circular accelerators," *Phys. Rev. Accel. Beams*, vol. 22, no. 10, p. 104 003, 2019. doi:10.1103/PhysRevAccelBeams.22.104003

[6] D. Di Croce, M. Giovannozzi, T. Pieloni, M. Seidel, and F. F. Van der Veken, "Accelerating dynamic aperture evaluation using deep neural networks," in *Proc. 14th Int. Particle Accelerator Conf. (IPAC'23)*, Venice, Italy, May 2023, pp. 2870–2873. doi:10.18429/jacow-ipac2023-wepa097

[7] *MAD - Methodical Accelerator Design*, https://mad.web.cern.ch/mad/.

[8] *Xsuite documentation, 2023*. http://xsuite.web.cern.ch

[9] G. Iadarola *et al.*, "Xsuite: An integrated beam physics simulation framework," presented at the 68th ICFA ABDW on High-Intensity and High-Brightness Hadron Beams (HB'23), Geneva, Switzerland, Oct. 2023, paper TUA2I1, these proceedings.

[10] R. D. Cook and S. Weisberg, "Influential observations, high leverage points, and outliers in linear regression," *J. Am. Stat. Assoc.*, vol. 74, no. 365, pp. 169–174, 1982.

[11] *MADX-PTC documentation, 2006*. http://mad.web.cern.ch/mad/madX/doc/usrguide/ptc_general/ptc_general.html

[12] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous systems," Mar. 2016. doi:10.48550/arXiv.1603.04467

[13] A. O. Gomes *et al.*, *Keras tuner*, GitHub repository. https://github.com/keras-team/keras-tuner

[14] T. Dozat, "Incorporating Nesterov momentum into Adam," in *Proc. 4th Int. Conf. on Learning Representations (ICLR'16)*, 2016.

[15] NVIDIA Corporation, *NVIDIA TITAN V*, 2017. https://www.nvidia.com/en-us/titan/titan-v/

[16] "CERN HT-Condor documentation." (2023), https://batchdocs.web.cern.ch/local/submit.html

[17] *Accelerator Research and Technology programme - CHART*, http://www.chart.ch.