instruments

MDPI

Article

Assessing the Performance of Deep Learning Predictions for Dynamic Aperture of a Hadron Circular Particle Accelerator

Davide Di Croce [1,*], Massimo Giovannozzi [2], Carlo Emilio Montanari [3], Tatiana Pieloni [1], Stefano Redaelli [2] and Frederik F. Van der Veken [2]

[1] Particle Accelerator Physics Laboratory, Ecole Polytechnique Fédérale de Lausanne, Rte de la Sorge, 1015 Lausanne, Switzerland; tatiana.pieloni@epfl.ch
[2] Beams Department, CERN, Esplanade des Particules 1, 1211 Geneva, Switzerland; massimo.giovannozzi@cern.ch (M.G.); stefano.redaelli@cern.ch (S.R.); frederik.van.der.veken@cern.ch (F.F.V.d.V.)
[3] Department of Physics and Astronomy, University of Manchester, Oxford Rd, Manchester M13 9PL, UK; carlo.emilio.montanari@cern.ch
* Correspondence: davide.di.croce@cern.ch

Abstract: Understanding the concept of dynamic aperture provides essential insights into nonlinear beam dynamics, beam losses, and the beam lifetime in circular particle accelerators. This comprehension is crucial for the functioning of modern hadron synchrotrons like the CERN Large Hadron Collider and the planning of future ones such as the Future Circular Collider. The dynamic aperture defines the extent of the region in phase space where the trajectories of charged particles are bounded over numerous revolutions, the actual number being defined by the physical application. Traditional methods for calculating the dynamic aperture depend on computationally demanding numerical simulations, which require tracking over multiple turns of numerous initial conditions appropriately distributed in phase space. Prior research has shown the efficiency of a multilayer perceptron network in forecasting the dynamic aperture of the CERN Large Hadron Collider ring, achieving a remarkable speed-up of up to 200-fold compared to standard numerical tracking tools. Building on recent advancements, we conducted a comparative study of various deep learning networks based on BERT, DenseNet, ResNet and VGG architectures. The results demonstrate substantial enhancements in the prediction of the dynamic aperture, marking a significant advancement in the development of more precise and efficient surrogate models of beam dynamics.

Keywords: machine learning; deep learning; circular particle accelerators; single-particle nonlinear beam dynamics

Citation: Di Croce, D.; Giovannozzi, M.; Montanari, C.E.; Pieloni, T.; Redaelli, S.; Van der Veken, F.F. Assessing the Performance of Deep Learning Predictions for Dynamic Aperture of a Hadron Circular Particle Accelerator. *Instruments* **2024**, *8*, 50. https://doi.org/10.3390/instruments8040050

Academic Editor: Nicolas Delerue

Received: 30 September 2024
Revised: 29 October 2024
Accepted: 4 November 2024
Published: 19 November 2024

1. Introduction

The advent of modern hadron circular accelerators that utilise superconducting magnets has significantly enhanced performance, especially regarding energy reach. However, a major drawback is that the inevitable addition of field errors inherent to superconducting magnets, which generate the field by means of current distributions (the magnetic field produced by normal-conducting magnets is controlled through the accurate shaping of their poles, which allows for finer control compared to the current distribution in superconducting magnets), renders the beam dynamics strongly nonlinear. This nonlinearity can potentially excite resonances, leading to beam losses and an increase in transverse emittances. These negative phenomena adversely affect the performance of accelerators, necessitating the implementation of mitigation strategies during both the design phase and later during the operation of the machine. In this context, it is vital to identify an indicator to measure the impact of magnetic errors on beam dynamics. Parenthetically, we should note that nonlinear effects also play a role in the dynamics of circular lepton accelerators.

The interested readers may refer to the selected references specified [1–7] including references therein. Among several possible options, the most effective is the dynamic aperture (DA), which is defined as the extent of the connected phase-space region within which single-particle dynamics remain bounded (see, e.g., [8] and references therein). A key aspect of this definition is specifying the time interval over which the dynamics should stay bounded, which is a parameter determined by physical considerations, particularly the length of the accelerator's operational cycle. The DA offers essential insights into the nonlinear beam dynamics of non-interacting particles as well as the resonance mechanisms that provoke beam losses and reduce beam lifetime. Despite its somewhat abstract nature, a direct link can be made to the beam losses resulting from nonlinear beam dynamics [9], which is a relationship that underpins the recent method for measuring DA in circular rings [10].

Mastering DA is crucial for optimising the functionality of current circular particle accelerators like the CERN Large Hadron Collider (LHC) [11] and its luminosity upgrade (HL-LHC) [12], and it represents a fundamental figure of merit for the conception of future accelerators, such as the CERN Future Circular Collider (FCC), in its hadron–hadron version (FCC-hh) [13] (refer to Ref. [14] for an overview of the most recent baseline layout).

The numerical determination of the DA involves monitoring multiple initial conditions in the phase space over numerous revolutions, the precise value of which depends on the application or specific physical processes involved. For example, in lepton rings, energy damping introduces a natural time scale that is relatively short, i.e., 1×10^2 turns to 1×10^4 turns, for DA calculations. In contrast, in hadron rings, where energy damping is minimal or absent, the time scale is defined by the application, and for the LHC or HL-LHC, it is usual to consider 1×10^5 (or 1×10^6) turns. Therefore, the methods presented in this paper primarily focus on the DA concerning hadron rings, as the absence of significant damping effects renders the long-term dynamics especially crucial, making DA calculation a highly CPU-demanding activity. It should be noted that the standard number of turns used in LHC DA calculations is equal to only about 9 s (or 90 s) of storage time, compared to 10 to 15 h of a typical fill duration: This suggests that there is a significant discrepancy between what can be computed and what is needed to model the actual LHC performance and that its treatment is crucial to the advancement of the field. The computation of DA is very demanding, particularly for large hadron accelerator rings like the LHC. The computational challenge is two-fold. Firstly, it involves the large number of initial conditions needed to accurately explore the phase space; secondly, it requires a substantial number of turns to assess the stability of the dynamics. The first challenge can be mitigated, as the initial conditions are non-interacting, by parallelising the DA computation algorithms [15] or exploiting the performance advantages provided by GPUs. The latter challenge lacks a straightforward solution; the most effective method has been the derivation of scaling laws for the DA versus the turn number. This method has seen success through the application of advanced theorems from dynamical systems theory, which yields the long-sought scaling laws [16,17].

Conducting precise numerical calculations of the DA requires significant CPU resources. Nevertheless, a single DA value is rarely practically valuable in the design and analysis of circular accelerators. Meaningful outcomes require examining several machine configurations. There are at least two reasons for this. First, the accelerator model may be known with limited precision. In such cases, a Monte Carlo method is typically employed, in which certain parameters describing the accelerator model are varied across a high-dimensional space. For the LHC, this technique applies to the errors that affect the superconducting magnets. Although an extensive measurement campaign was conducted to qualify all produced magnets, only a small subset was tested under cold conditions, which are the conditions of normal operation. Magnetic errors measured under these conditions can be directly used in DA simulations. For most magnets, errors are known only under warm conditions, and a transformation is required to deduce errors under cold conditions [18–21]. It is customary to neglect the impact on the field quality experienced by

the charged beams of the misalignment errors of the various magnets. This methodology introduces some degree of uncertainty regarding the actual errors in these magnets. As a result, it is typical to produce realisations of the magnetic errors measured under warm conditions based on the uncertainty in the measurements. For the LHC, it is standard to consider sixty realisations (also known as seeds) of the ring lattice, each incorporating different sets of magnetic errors: overall, the true DA value should fall within the range of DA values for the collection of realisations. The second reason for assessing multiple DA values is that optimising the ring configuration is a crucial aspect of the circular accelerator design and operation. This is usually achieved by evaluating numerous machine configurations and examining the hyper-surface that represents the DA as a function of the machine parameters, identifying regions that maximise the DA and produce minimal DA variation close to the peak (see, e.g., Refs. [22,23]).

Observing the current trend in accelerator physics, it is clear that the pursuit of constructing surrogate DA models has grown into a highly appealing research area. Clearly, in this field, modern computational technologies, such as machine learning (ML), offer an attractive solution to this demand. Recently, ML methodologies have been investigated to create efficient surrogate models for DA (see, e.g., [24]) or to better reconstruct its time dependence [25]. Furthermore, deep learning (DL) techniques have been applied to quickly and precisely forecast DA for unfamiliar machine configurations [26]. By training a multilayer perceptron (MLP) on a comprehensive data set of simulated initial conditions, this method effectively captures the intricate relationships between accelerator parameters and the corresponding DA values. Furthermore, this model has shown the ability to accurately represent even new optical configurations of the accelerator with minimal new data due to its ability to extract universal physical features from common accelerator variables that influence beam dynamics. This feature holds significant promise for the design and optimisation of future accelerators, underscoring the crucial role of surrogate models.

Expanding on our initial MLP, we conducted additional research, evaluations, and comparisons involving various cutting-edge DL architectures. Each architecture has been tested on the same data set to assess its potential in improving the accuracy of DA forecasts. It is important to note that while numerical accuracy is crucial, it should be weighed alongside the inference time, which needs to be sufficiently brief to make these advanced architectures a viable substitute for traditional numerical DA computations. These points are thoroughly examined and discussed throughout this paper.

The layout of this paper is as follows: Section 2 describes the computation of the DA through direct numerical simulations and addresses the data preparation to implement the DL models. Section 3 examines the proposed DA regressor, providing a comprehensive review of the various architectures developed and evaluated in this research. The analysis of the developed DL architectures is divided into three sections: Section 4 focuses on the training and accuracy of the different architectures; Section 5 discusses the inference performance; and Section 6 looks at the surrogate model's performance when DA is considered a function of the number of turns. Lastly, conclusions are presented in Section 7, and the appendices provide details on some aspects of this study.

2. Simulated Samples for Deep Learning Models of the DA

2.1. Simulating the DA

The definition of DA requires some care, and the detail is given in Appendix A. The stability of the orbits should be checked by performing an appropriate sampling of the phase space, which is a very time-consuming task. For this reason, some simplifications can be applied so that only a subspace of the entire phase space is carried out. It is customary to consider a scan over the initial conditions of the form $(x, 0, y, 0)$, which reduces the computation of the DA to a 2D scan. Furthermore, in the rest of the paper, the concept of angular DA is used: It is simply defined as $\mathrm{DA}(\alpha_k, N) = r_s(\alpha_k, N)$ using the same notation as in Equation (A14).

An essential point to cover is the criterion for defining bounded orbits, which necessitates setting a threshold amplitude. Orbits with amplitudes below this threshold throughout the tracking simulation are classified as bounded (or stable). If an orbit exceeds this threshold, it is deemed unbounded (or unstable), and the number of turns reached is recorded to define the orbit's stability time. This assessment is performed turn by turn with a default threshold value of 1 m. In particular, in the LHC, a collimation system [11] is used to safeguard the cold aperture by absorbing particles at higher amplitudes. The jaws of this system serve as absorbers and set a physical limit on the amplitudes of the orbit. Therefore, in our DA simulations, we considered defining the threshold amplitude based on the aperture of the collimator jaws.

From a computational perspective, we evaluated DA by tracking initial conditions in phase space using XSuite [27,28]. These initial conditions are set in polar coordinates, allowing us to establish the stability limit for each angle, which is known as the angular DA. To speed up the tracking process, an initial coarse scan was performed using evenly spaced initial conditions over eight polar angles within the range of $[0, \pi/2]$ and 33 radial amplitude values spanning $[0, 20\sigma]$. (It is customary to express amplitudes in terms of the transverse rms beam size, which is computed starting from the values of the normalised emittance ϵ^*. In the case of the LHC [11], the nominal value of ϵ^* equals 3.5 µm for both x and y planes.) This approach identifies the angular DA value for each angle and pinpoints the smallest amplitude where the initial conditions do not remain bounded beyond 1×10^3 turns, defining the boundary of the fast-loss region. Subsequently, a finer scan is performed within the amplitude range from 2σ inside the stable region to 2σ outside the fast-loss boundary and tracking for 1×10^5 turns. This finer scan employs radial increments of 0.06σ and examines 44 polar angles within $[0, \pi/2]$. Overall, this method provides a detailed phase space scan, focusing on areas near the stability boundary. This technique not only improves the accuracy of the angular DA computation but also triples the tracking efficiency by limiting the scanned space, as empirically shown [29].

An example of the results of these calculations, performed in the $x - y$ space, is shown in Figure 1 for one of the LHC configurations that are part of the data set used to construct the DL surrogate models.

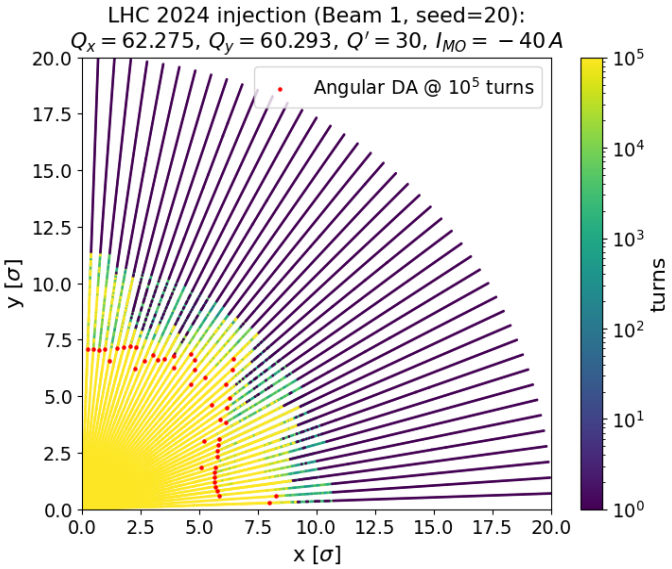


Figure 1. Stability time for a distribution of initial conditions in $x - y$ space for one of the LHC configurations that are part of the data set used for constructing the DL surrogate models. The reduction in the extent of the stable region for an increasing number of turns is clearly visible. This information is then used to determine the value of the angular DA. As an example, the angular DA for 1×10^5 turns is shown in red.

We note that the horizontal and vertical axes represent in reality the value of the horizontal and vertical invariants. Therefore, for the case of the hadron accelerators, for which the dynamics is symplectic, only positive values of the initial coordinates are considered.

2.2. Generating Accelerator Configurations and DA Samples

The data set used to build the DL models is made up of accelerator configurations that are generated using MAD-X [30–33]. The accelerator considered is the LHC, which is in its configuration used during the 2024 physics run at injection energy (450 GeV). This base configuration has been used to generate many variants labelled by the values of some key accelerator parameters, namely the betatron tunes Q_x, Q_y, the linear chromaticities Q'_x, Q'_y, the strength of the Landau octupole magnets (these magnets are used in operations to passively mitigate collective instabilities. The strength of the magnets is the key parameter and, in our studies, it has been labelled by the current I_{MO} that flows in the octupoles), and the realisations of the magnetic field errors (also called seeds) that have been assigned to the various magnet families. Furthermore, in these studies, the ring configurations for Beam 1 (clockwise beam) and Beam 2 (counter-clockwise beam) have been independently considered. Note that the two magnetic channels of Beam 1 and Beam 2 only have a relatively small number of single-bore magnets in common, namely 36 superconducting magnets and 30 normal-conducting magnets, while the two-bore magnets are several thousands. For the two-bore magnets, the magnetic field errors are different for the two apertures, thus justifying the approach of considering both beams as independent configurations.

An initial data set comprising 5×10^3 LHC configurations was generated by performing a random uniform grid search of the following parameters: $Q_x \in [62.1, 62.5]$ and $Q_y \in [60.1, 60.5]$, with steps of size 5×10^{-3}; $Q' \in [0, 30]$ (high chromaticity values were incorporated into the scan since they may actually be employed to mitigate collective instabilities, and it is important to integrate this aspect into our DA model; also, note that the same value Q' is used for Q'_x and Q'_y), with steps of size 2; $I_{\mathrm{MO}} \in [-40\,\mathrm{A}, 40\,\mathrm{A}]$ with steps of size 5 A. For each of the 1000 configurations, five different realisations of magnetic errors (seeds), randomly selected among the 60 available, have been considered for both beams, resulting in a total of 5×10^3 configurations.

The control of the ring chromaticity is performed by means of two families of sextupole magnets that are installed in regular arc cells, close to the focusing and defocusing quadrupoles, as is customary for a FODO lattice. The Landau octupoles are also located close to focusing and defocusing cell quadrupoles, but only in a subset of the regular cells in the ring arcs.

Moreover, 5655 additional accelerator configurations (which gives a data set with a total of 10.655×10^3 configurations) were generated using the active learning (AL) framework developed by our team [29]. This framework allows the smart sampling of accelerator configurations through a clever selection of configurations with higher error estimates. As demonstrated in our previous studies, by prioritising configurations with larger magnitude error, the AL framework enables a more efficient exploration of machine configurations where the surrogate model has not yet fully captured the underlying physics features.

We also considered six values of the threshold amplitude to identify bounded orbits, namely four cases using different collimator apertures ($5\sigma, 5.7\sigma, 6.7\sigma, 11\sigma$) and one using the default aperture of 1 m.

The relevance of the evolution over time of DA has already been mentioned above. Therefore, the possibility of reconstructing a model for this dependence has been explored by monitoring the stability time of the initial conditions. For this reason, the stability times have been binned using 19 distinct intervals whose boundaries are given by $1, 5, 10l_1$, $100 + 50l_2$, $1 \leq l_1 \leq 10$, $1 \leq l_2 \leq 8$ thousand turns.

Concerning the number of turns performed in the tracking simulations, 3805 LHC configurations were tracked for 5×10^5 turns, while in the remaining 6850 configurations, tracking was limited to 1×10^5 turns. This approach was used to assess the capacity of the

model to predict the angular DA over a larger number of turns even with a smaller sample size. Further discussion of this methodology is provided in Section 6.

The quantity of samples employed in our model is determined by the product of three factors: the quantity of accelerator configurations in the data set, the 44 angles used to examine the phase space, and the 19 bins used to observe the evolution of the stability time. This calculation results in a considerable 836-fold increase in the size of the data set of the accelerator configurations, substantially improving the chances of model training convergence while allowing us to track the contours of the stability limits and their development.

As beam–beam effects are not included, the value of the beam emittance represents a simple scale factor. Therefore, it is possible to augment the angular DA data set, computed for the nominal value, by using different values of the beam normalised emittances ϵ'_x and ϵ'_y. This strategy does not require any additional CPU-intensive angular DA computation but rather only a simple rescaling of existing results. In fact, the value of the angular $\mathrm{DA}'(\alpha_k, N)$ using ϵ'_x and ϵ'_y is given by

$$\mathrm{DA}'(\alpha_k, N) = \mathrm{DA}(\alpha_k, N)\sqrt{\frac{\epsilon^* \sqrt{\epsilon_x'^2 \sin^2 \alpha_k + \epsilon_y'^2 \cos^2 \alpha_k}}{\epsilon'_x \epsilon'_y}}\,, \tag{1}$$

where ϵ^* is the nominal normalised emittance value used to compute $\mathrm{DA}(\alpha_k, N)$. In our studies, we considered the following emittance values:

$$\begin{aligned} \epsilon'_x &\in \{0.25, 0.5, 0.75, 1.0, 1.5, 2.0, 3.5, 5.0, 7.0, 10.0, 20.0, 50.0\}\ \mu\mathrm{m} \\ \epsilon'_y &= \epsilon'_x(1+\xi)\,, \end{aligned} \tag{2}$$

where ξ is a Gaussian-distributed random variable with zero mean and sigma equal to 0.1, which corresponds to having always almost equal horizontal and vertical emittances. This approach is designed to achieve two objectives: it enables the surrogate model to grasp an initial understanding of how DA is influenced by the beam emittance value and, more significantly, to introduce a method to even out the distribution of angular DA. This is essential to provide an unbiased training set for the DA regressor, which is achieved by randomly sampling the inverse distribution of the angular DA values after augmentation. The distribution of angular DA is illustrated in Figure 2 both before (red) and after (blue) the augmentation and unbiasing steps. Initially, the higher DA values were not sufficiently represented; however, following these adjustments, the distribution becomes more uniform, resulting in a better-balanced representation of DA values.

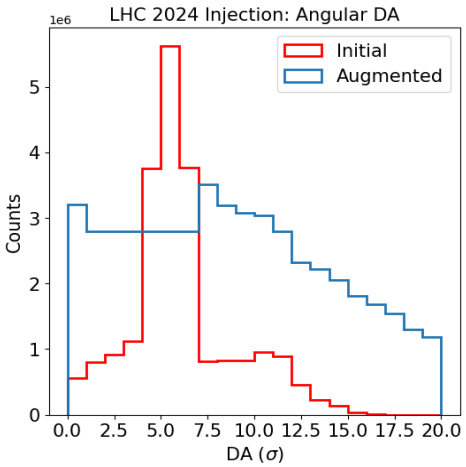


Figure 2. Distribution of the angular DA before (blue) and after (red) the augmentation and unbiasing pre-processing.

The total number of samples after the unbiasing step is about 50 million. From this sample size, 10% of the samples were used for validation and 10% were used to test the performance of the model, while the rest was used for training.

2.3. Pre-Processing of DA Data Set

Labels used to denote machine configurations might be inadequate for classifying the beam dynamics. Consequently, to provide a more comprehensive description of the phenomena that affect DA, we incorporated several variables calculated using MAD-X and PTC [34–36]. For a better characterisation of the LHC ring's optical configuration, we included the maximum values of the Twiss parameters $\alpha_{x,y}$ and $\beta_{x,y}$, along with the phase advance $\mu_{x,y}$, between the high-luminosity insertions of the ATLAS and CMS experiments, which are located at IP1 and IP5, respectively. This approach aims to better capture the interrelationship between the optical parameters and the angular DA. Additionally, to encapsulate observables related to nonlinear beam dynamics, we accounted for seven anharmonicity coefficients, specifically the amplitude detuning terms up to the second order [37].

A crucial step is the standardisation of input variables that take real values. However, note that the beam and seed variables take discrete values and are excluded from this step. This process entails normalising the distributions by using their mean and standard deviation. By ensuring a uniform scaling of all input features, this step helps to achieve quicker model convergence and improves model stability [38]. Consequently, this improves the performance and interpretability of the model.

In contrast to [26], we did not cap the DA values, as our augmentation and unbiasing steps result in a balanced distribution of DA values, allowing the model to learn from the full range of DA values without introducing bias.

3. DA Regressor

The DA regressor is a neural net designed to assess accelerator parameters, the polar angle, and the number of tracked turns to predict the angular DA value. We explored several network architectures using the TensorFlow library [39] and used the rectified linear unit (ReLU) activation function [40] to enhance nonlinear learning capabilities. Furthermore, the hyperparameters were fine-tuned using random search with the Keras Tuner framework [41] to improve model performance. The subsequent subsections offer a comprehensive description of each tested network.

The network architectures discussed in this document showcase the top performing setups discovered during our investigation, encompassing various layer configurations and types. Our experiments revealed that attention mechanisms and residual connections were especially effective in representing the numerical variables in this data set. Although these configurations delivered optimal results, it is possible that they are tailored to this specific data set, suggesting that exploring a wider range of hyperparameters might provide additional insight. Nonetheless, we consider the effectiveness of attention and residual mechanisms in this scenario to be noteworthy, potentially pointing to their broader applicability in similar data sets.

3.1. Multilayer Perceptron

The MLP consists of a fully connected neural network, and its advantage over more complex architectures lies in its simplicity, requiring fewer computational resources and fewer data for training. The implemented network structure closely follows that implemented in our previous study [29]. This includes four hidden layers with 2048, 1024, 512, and 128 nodes, respectively. Batch normalisation is applied to each hidden layer to stabilise and speed up training by normalising layer outputs. To avoid overfitting, three dropout layers between the hidden dense layers with a rate of 1% were applied in [29]. However, after performing hyperparameter tuning for this new data set, the optimal dropout rate was found to be 50%. The structure of the MLP network used is shown in Figure 3.

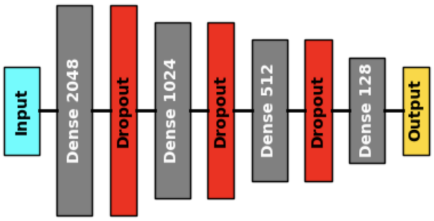


Figure 3. Architecture of the designed MLP for the DA regressor, featuring a fully connected structure.

3.2. Bidirectional Encoder Representations from Transformers

The Bidirectional Encoder Representations from Transformers (BERT) neural network [42] was initially designed to improve natural language understanding tasks due to its unique architecture. BERT is characterised by its use of self-attention mechanisms in multiple Transformer blocks [43], which allows it to capture complex, bidirectional dependencies within input data. The embedding in the network converts the input features into dense vectors, which effectively represent the patterns in the data. These vectors are then processed through self-attention layers, where each token in the input sequence is able to consider and weigh its relationship to all the other tokens in the sequence. This dynamic attention mechanism adjusts focus based on the relevance of tokens, allowing the model to capture subtle contextual relationships across the entire sequence, enhancing the model's ability to learn intricate relationships and improve predictive accuracy.

Our BERT-based network is designed for numerical data processing and consists of 12 Transformer encoders. Each Transformer encoder comprises a multi-head self-attention layer with eight attention heads, allowing the model to learn from different representation subspaces simultaneously. Following this, a single feed-forward neural layer (FFN) is employed with a hidden layer size of size 512. Layer normalisation and dropout layers with a rate of 0.5 are applied before and after each FFN to stabilise the training and prevent overfitting. A global average pooling layer is included after the Transformer blocks to reduce the sequence dimension and aggregate the information into a fixed-size vector, ensuring efficient downstream processing. The structure of the BERT network used is shown in Figure 4.

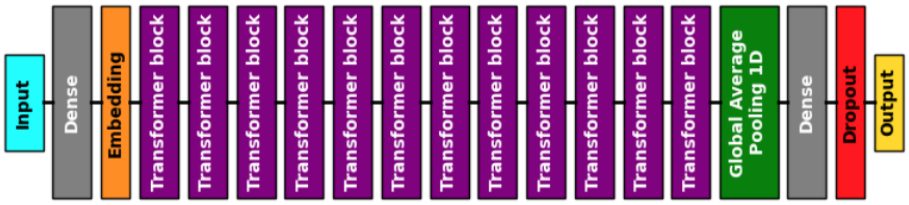


Figure 4. Architecture of the designed BERT for the DA regressor.

3.3. Densely Connected Convolutional Networks

The Densely Connected Convolutional Networks (DenseNet) network introduces a novel connectivity pattern in which each layer is directly connected to every other layer in a feed-forward fashion [44]. This dense connectivity results in a significant reduction in the number of parameters compared to other deep network configurations of similar depth, leading to more compact models with improved computational efficiency. The structure of the DenseNet network used is shown in Figure 5.

Our implementation of DenseNet consists of 121 layers, including dense blocks, transition layers, and fully connected layers. The network begins with an initial convolutional layer, which is followed by maximum pooling to reduce spatial dimensions. Then, a series of four dense blocks, each separated by a transition layer, are applied. Within each dense block, multiple layers are connected directly to every other layer, promoting gradient flow and alleviating the vanishing gradient problem. A global average pooling layer is used

before the fully connected layers to aggregate features, and dropout with a rate of 0.5 is used to prevent overfitting, enhancing the generalisability of the model.

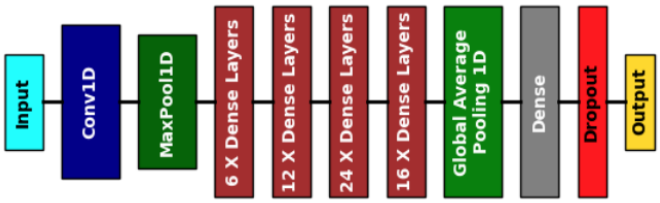


Figure 5. Architecture of the designed DenseNet-121 for the DA regressor.

3.4. Residual Networks

Residual Network (ResNet) is a deep convolutional neural network (CNN) known to address the challenges of training very deep networks through the introduction of residual learning [45]. It leverages residual connections, or *skip connections*, that allow the network to bypass one or more layers. Instead of learning the full output transformation at each layer, the network learns the residual (or difference) between the input and output of the layer. This mechanism helps mitigate issues such as the vanishing gradient problem and enables the construction of much deeper networks without performance degradation. The structure of the ResNet network used is shown in Figure 6.

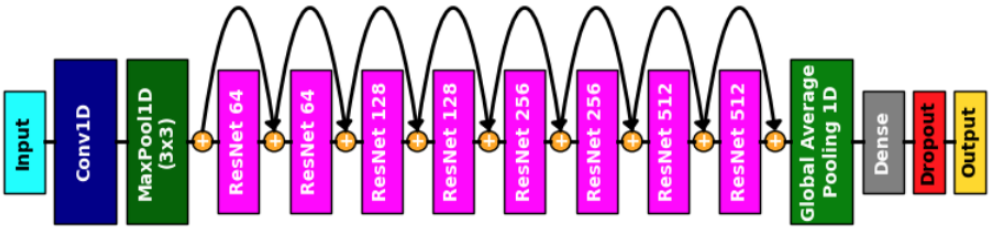


Figure 6. Architecture of the designed ResNet-18 for the DA regressor.

We used a relatively shallow variant of this topology, based on ResNet-18, making it computationally efficient and suitable for tasks where faster training and inference are needed without sacrificing performance. It starts with an initial 1D convolutional layer followed by eight residual blocks, each containing two convolutional layers with Batch Normalisation. The network ends with a global average pooling layer to aggregate features, a dense layer with 1024 units, and a dropout layer with a rate of 0.5 to mitigate overfitting.

3.5. Visual Geometry Group

The Visual Geometry Group (VGG)-16 model developed for our application features a deep architecture with a series of convolutional blocks designed for effective feature extraction [46]. Each VGG block consists of multiple 1D convolutional layers followed by max-pooling, which reduces the spatial dimensions of the feature maps by selecting the maximum value in each region of the feature map, allowing the network to focus on the most important information while downsampling. The structure of the VGG-16 network used is shown in Figure 7.

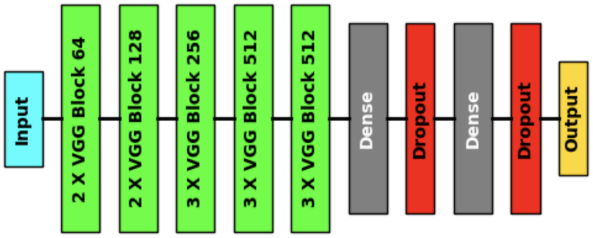


Figure 7. Architecture of the designed VGG-16 for the DA regressor.

After the convolutional layers, the features are flattened and passed through two fully connected dense layers with 4096 units each, including a dropout layer with a rate of 0.5 for regularisation.

3.6. Hybrid

Finally, we developed a hybrid network that integrates three key components: a Transformer encoder, a residual block, and a dense network. The input features are converted into dense vectors through an embedding layer. Then, the Transformer encoder, featuring a multi-head attention layer with a feed-forward layer, captures complex dependencies and contextual information. This is followed by a residual block that enhances feature learning through skip connections. A dense block builds upon the residual connections, incorporating dropout and concatenation to improve robustness and performance. The dense network processes the extracted features with dense layers and an additional dropout layer (with a 0.5 rate) for regularisation. Combining these elements ensures that the network maintains high performance while being computationally efficient compared to traditional DenseNet and BERT models. The structure of the Hybrid network used is shown in Figure 8.

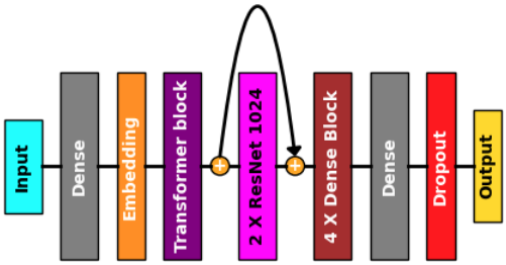


Figure 8. Architecture of the designed Hybrid network for the DA regressor.

4. Training and Precision of DA Models

For training of all the networks tested, we used about 1.1×10^4 accelerator configurations of the LHC 2024 optics case at injection energy, comprising data tracked for 1×10^5 turns and 5×10^5 turns, totalling approximately 9.1 million samples. The batch size used for training is 1024 samples. The loss function used for the regressor is the mean absolute error (MAE) function that is trained with the NADAM optimiser [47]. The initial learning rate is 5×10^{-5} and is halved every five sequential epochs if the validation loss is not improved to enhance the model accuracy. Early stopping was used with a patience of 20 epochs to prevent overfitting. Figure 9 illustrates the performance improvements throughout the training process and the adjustments to the learning rate.

The MAE, the mean absolute percentage error (MAPE), and the root mean squared error (RMSE) for the test (train) data set are presented in Table 1, while Table 2 shows the training time spent. The BERT model achieves the lowest MAE in the test (0.342σ). However, it required 37 h for training, reflecting a high computational cost. DenseNet-121 exhibits a reasonable MAE in the test (0.365σ) but required the longest training time (138 h), indicating a significant computational cost to achieve its performance. ResNet-18 shows a balance between prediction accuracy (0.376σ) and training time, taking only 39 min per epoch. In contrast, the MLP (baseline), with moderate error metrics (0.406σ), completed the training only in 3 h. Moreover, there are no signs of overfitting, as evidenced by the similar error metrics between the training and test sets in Table 1. This is further supported by the training progress graph, such as the one shown in Figure 9, which demonstrates consistent performance across both data sets.

Figure 10 illustrates the 2D histogram of the predicted versus computed angular DA for all evaluated networks. This visualisation shows that most models perform well for the majority of data points, with a tight cluster along the diagonal line, indicative of accurate predictions. Additionally, the Pearson correlation coefficient is presented, demonstrating values approaching unity across all instances albeit marginally lower for the Hybrid and VGG-16 architectures.

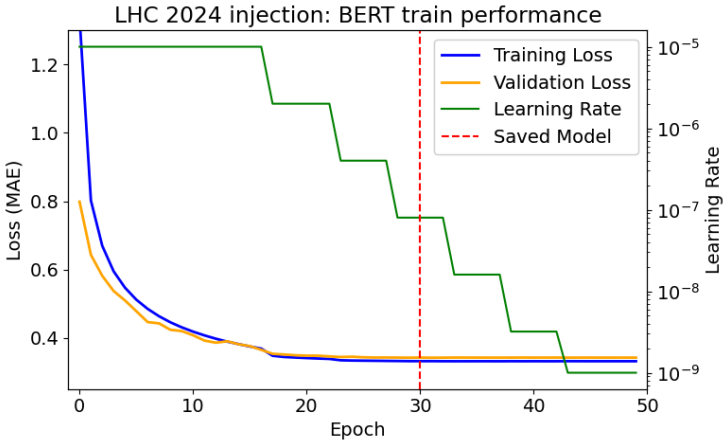


Figure 9. Training and validation performance over epochs. The blue line represents the training loss (MAE), while the orange line shows the validation loss. The green line tracks the learning rate adjustments throughout the training. The vertical dashed line indicates the epoch at which the model was saved.

Table 1. Prediction performance of the network architectures on the train and test data sets.

Network	MAE [σ]		MAPE [%]		RMSE [σ]	
	Train	Test	Train	Test	Train	Test
BERT	0.340	0.342	14.36	14.37	0.536	0.536
DenseNet-121	0.356	0.365	12.81	13.11	0.546	0.560
ResNet-18	0.373	0.376	14.42	14.42	0.578	0.578
MLP (baseline)	0.403	0.406	18.35	18.57	0.618	0.621
Hybrid	0.546	0.549	28.78	28.87	0.799	0.801
VGG-16	0.642	0.645	62.90	62.91	0.899	0.899

Table 2. Time performance of the network architectures during training.

Network	Epochs	Time Per Epoch [min]	Total Time [h]
BERT	30	74.3	37
DenseNet-121	76	107.7	138
ResNet-18	74	38.8	48
MLP (baseline)	39	5.1	3
Hybrid	49	38.1	31
VGG-16	57	22.9	22

Figure 11 presents the comparison of the distribution between the computed and predicted angular DA, demonstrating their compatibility and further validating the predictive accuracy of the models. In addition to the significant 16% enhancement in MAE achieved by the BERT architecture compared to our former MLP baseline model, BERT also excels in angular DA coverage. As illustrated in Figure 11, all models typically underestimate the final angular DA bin. This is a familiar problem in regression assignments, as regressors generally exhibit increased errors near boundary values. However, it should be noted that the BERT model delivers markedly more precise predictions, particularly at higher angular DA values where the MLP falters. This is crucial, since predicting values close to the extremities of a distribution can be challenging. The attention mechanism of BERT enables it to focus on the most pertinent input features, thus improving its accuracy at the boundaries, whereas the uniform weighting of MLP often decreases performance in these areas. This underscores the superior predictive power and robustness of BERT. In

closing, it is noteworthy that the rise in DA prediction associated with the 6σ bin remains unexplained, as it is absent in the numerical data.

In addition, we provided an assessment of the models using the Kolmogorov–Smirnov (KS) test, which measures the maximum distance between the empirical distribution functions of the predicted and computed angular DA. The KS test values further highlight the effectiveness of BERT, which achieves the best KS test value. It is followed by DenseNet, ResNet-18, and the MLP baseline. Models such as Hybrid and VGG-16 exhibit poorer performance with KS values. In this regard, we note that the KS values might be high because these models tend to prioritise the reduction of errors in the central parts of the distribution, where most of the data are concentrated. This occurs at the cost of precision in the tails. The KS test is sensitive to differences at the extremes of the distribution, which can result in a less precise performance of the model in these areas. These results support the superior distributional alignment achieved by BERT, underlining its robustness at varying values of angular DA.

Although the Hybrid network developed for the regression of the angular DA exhibited faster inference times than DenseNet, BERT, and ResNet, its prediction accuracy lagged behind. This could be attributed to the network's insufficient depth, preventing it from fully utilising the benefits of residual connections, attention mechanisms, and dense blocks.

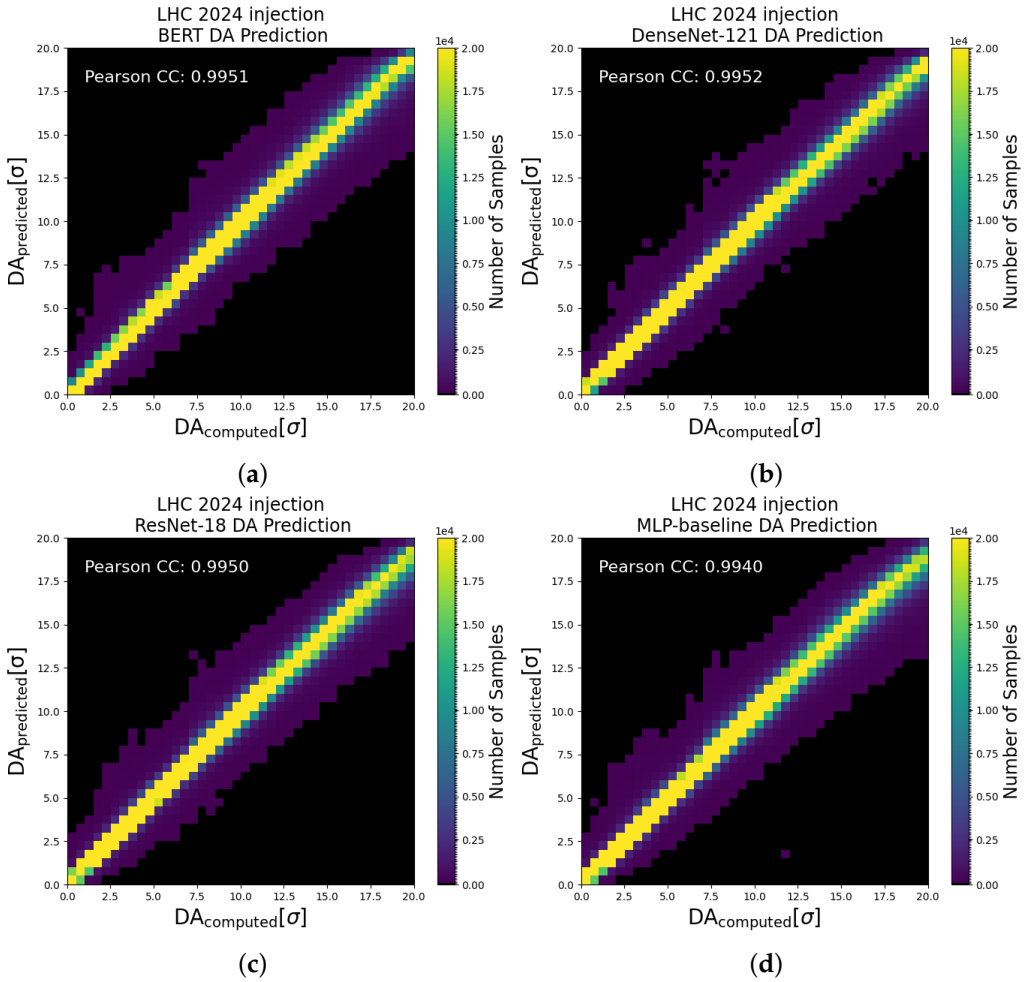
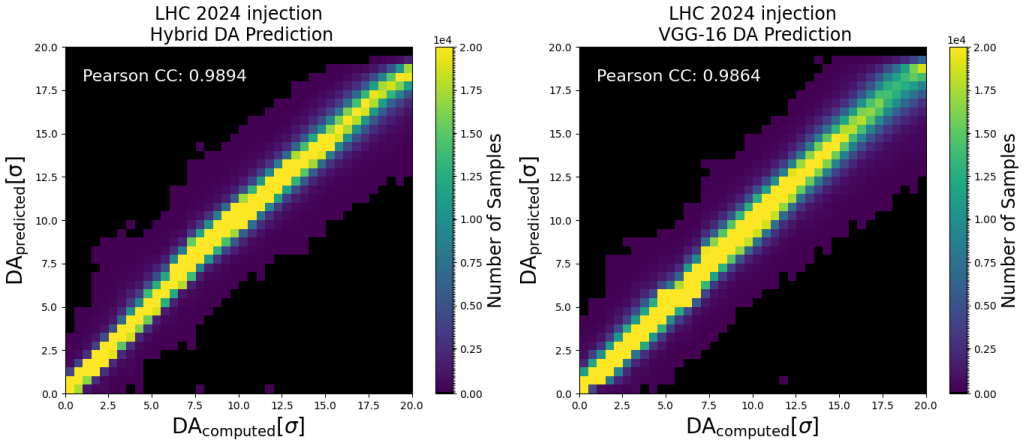
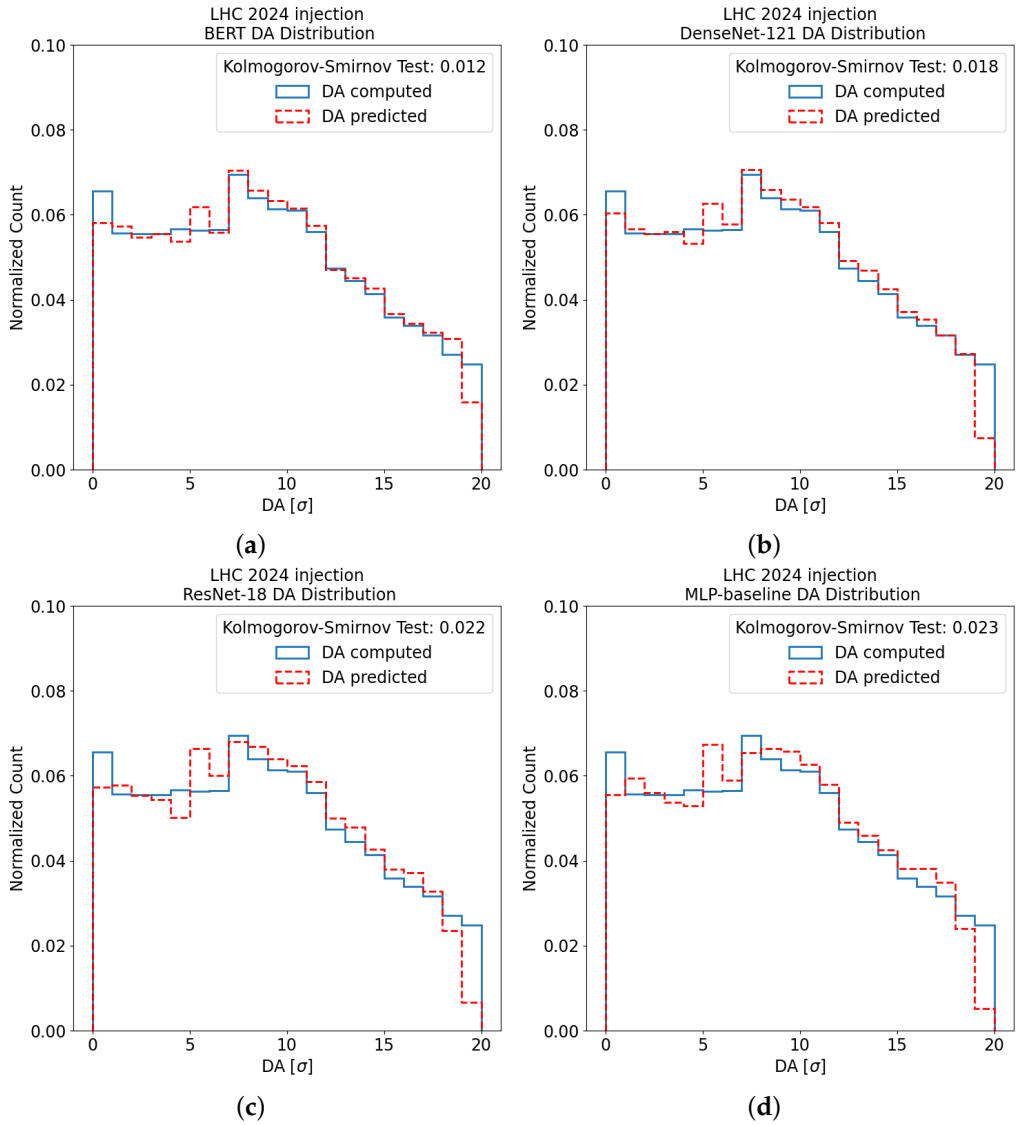


Figure 10. *Cont.*



(**e**) (**f**)

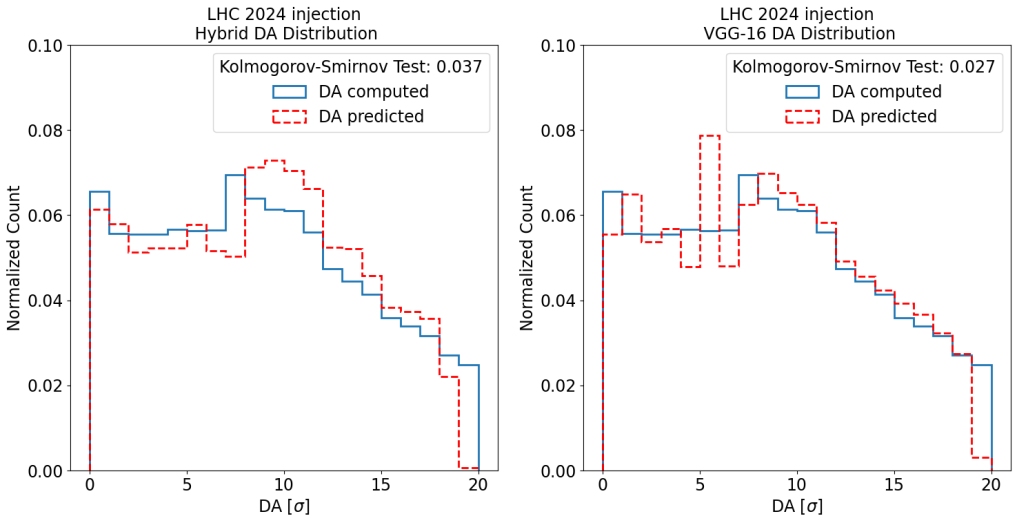
Figure 10. Predicted angular DA as a function of the computed angular DA values for the test data set for: (**a**) BERT, (**b**) DenseNet-121, (**c**) ResNet-18, (**d**) MLP (baseline), (**e**) Hybrid and (**f**) VGG-16 architectures. The Pearson correlation coefficient is also shown.



(**a**) (**b**)

(**c**) (**d**)

Figure 11. *Cont.*



(**e**) (**f**)

Figure 11. Computed (blue) and predicted (red) angular DA distribution for the test data set for: (**a**) BERT, (**b**) DenseNet-121, (**c**) ResNet-18, (**d**) MLP (baseline), (**e**) Hybrid and (**f**) VGG-16 architectures. The outcome of the Kolmogorov–Smirnov test, used to compare the two distributions, is also reported.

5. Inference Performance of DA Models

The inference time of DL networks is influenced by several aspects, including the number of samples and the parameters of the model. These elements can lead to a significant increase in computational costs for training the network, affecting both the duration of training and the use of memory. To address this issue, we assessed the performance of two hardware architectures designed for accelerated operation with our DA regressor: an Apple M3 MAX [48] and a NVIDIA V100 [49]. A brief overview of the hardware setup is provided in Appendix B.

Table 3 displays the timing performance of various network architectures when predicting a single angular DA and a complete accelerator configuration in two hardware platforms. Examples of complete machine configurations are illustrated in Figure 12. The findings indicate that the NVIDIA V100 generally surpasses the M3 MAX in inference speed for most architectures even though the latter features quicker I/O speeds. In addition, architectures with deeper layers and more intricate operations, such as BERT, exhibit longer inference durations due to higher computational demands. In contrast, simpler models or those with fewer parameters, such as ResNet-18 and MLP (baseline), generally exhibit faster inference times.

Table 3. Time performance of the considered network architectures during inference.

Network	Single Angular DA		Machine Configuration	
	M3 MAX [ns]	**V100 [ns]**	**M3 MAX [ms]**	**V100 [ms]**
BERT	189	61	158	51
DenseNet-121	158	41	132	34
ResNet-18	21	12	18	10
MLP (baseline)	10	4	8	3
Hybrid	116	45	97	38
VGG-16	66	27	55	23
Data Loading	0.9	3	0.8	2

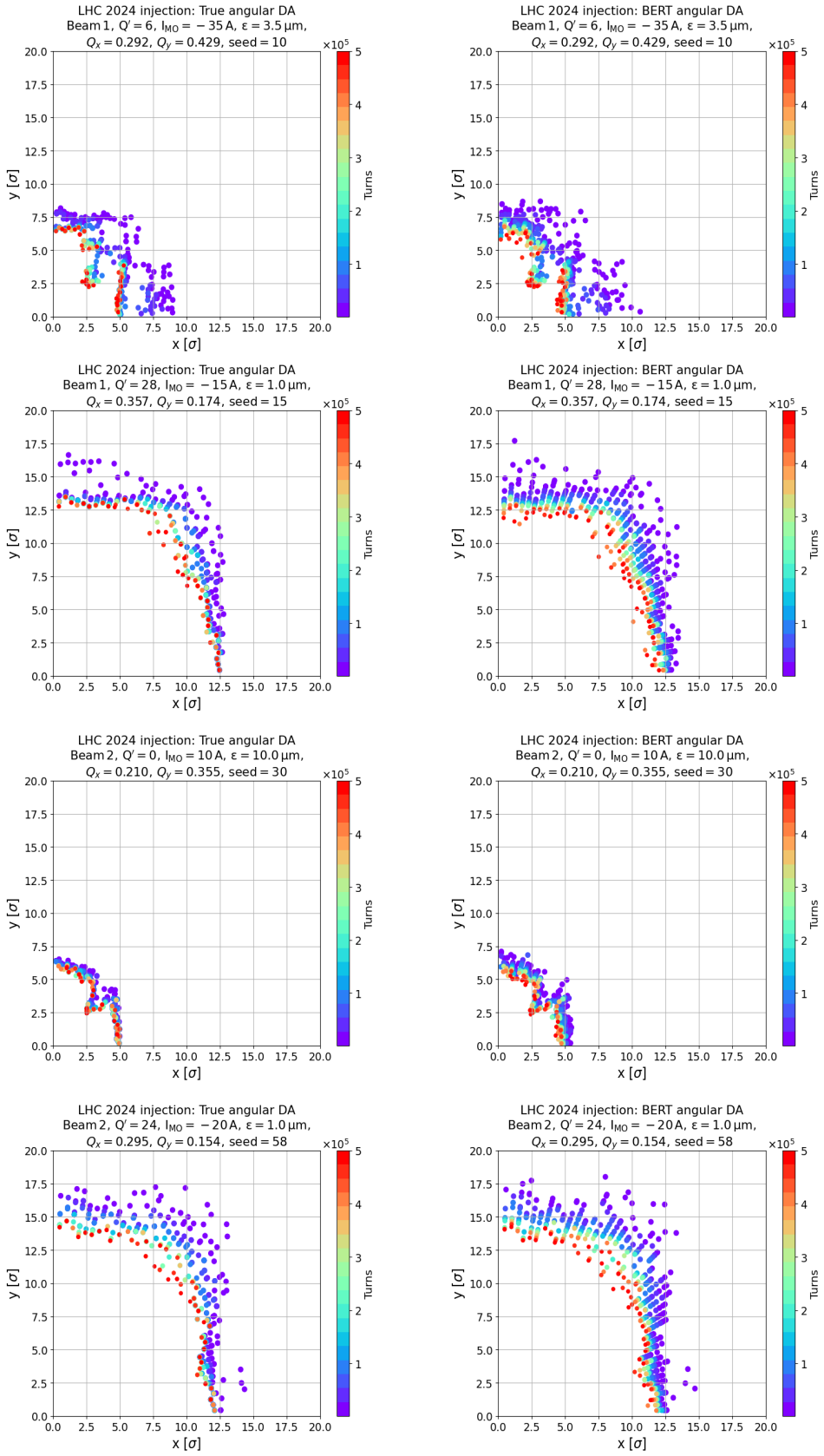


Figure 12. The true angular DA (**left**) and the BERT prediction (**right**) for four different machine configurations present on the test data set. The colours indicate the stability time in turns.

While the conventional numerical tracking approach using X-Suite and the HT-Condor batch system [50] can manage around 1×10^3 jobs simultaneously and takes approximately 30 h to track particles across 1×10^3 different configurations, equating to about 107 s per accelerator configuration, BERT, even as the most intricate network, delivers a significant performance boost. On the M3 MAX, BERT achieves a speed-up factor of roughly 675, and on the NVIDIA V100, it delivers an even higher speed-up factor of 2×10^3. Naturally, it is important to emphasise that achieving such a significant speed-up is only feasible after constructing an adequately large data set, which allows for the computation of DL surrogate models. This data set is created through conventional numerical tracking simulations, which are computationally very expensive. Consequently, the benefit of the surrogate model lies in the context of exploring a wide array of ring configurations, such as for design studies or optimisation purposes.

6. Prediction of DA for Higher Number of Turns

In previous studies [26,29], we limited the tracking of particles to 1×10^5 turns. Certainly, it is valuable to predict DA for a larger number of turns; but expanding the tracking is computationally demanding. For this reason, as mentioned in Section 2, only one third of the ring configurations in our data set were tracked up to 5×10^5. This introduces the possibility of testing the ability to capture the evolution of the DA when tracking particles for an even longer duration even if only a fraction of the samples contain extended tracking information.

Figure 13 shows the absolute error distribution for each of the bins used to distribute the stability time when the BERT network is used. Performance in the test data set demonstrates that the model accurately predicts the angular DA, even with limited information on extended turn counts. This indicates that the model capability captures the evolution of DA as a function of turn, even if the training of the surrogate model has been performed including only a reduced sample size that contains simulation results up to 5×10^5 turns. It should be noted that there were slight discrepancies in the MAE values between machine configurations tracked up to 1×10^5 turns and those tracked up to 5×10^5 turns, as indicated in Figure 13, which may be due to differences in the statistical characteristics of the training data. In particular, the input variables may exhibit varying distributions between the samples tracked up to 1×10^5 turns and those tracked up to 5×10^5 turns, potentially impacting the model's predictive accuracy across these subsets. These distribution shifts could lead to the observed performance discrepancies.

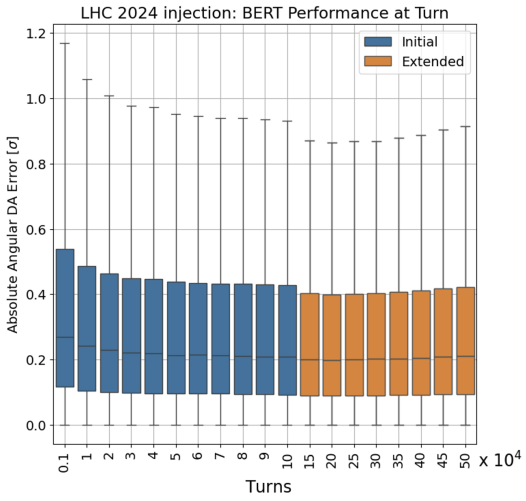


Figure 13. Box plot of the MAE as a function of the number of turns for the machine configurations tracked up to 5×10^5 turns in the test data set. The box limits indicate the range of the central 50% of the data with a central line marking the mean value.

7. Conclusions

This research investigates several advanced deep learning architectures to improve DA prediction in circular particle accelerators, using 2024 LHC optics at injection as a benchmark. We experimented with models including BERT, DenseNet, and ResNet, finding substantial enhancements in prediction accuracy compared to our previous MLP model. Specifically, the BERT model achieved the highest precision, although it required higher computational resources. However, ResNet-18 provided a balanced trade-off between performance and computational efficiency with respect to the inference time; the training time is only 30% grater than BERT. All evaluated networks showed remarkable speed increases in inference times over traditional tracking methods, achieving predictions that were at least 675 times faster on advanced hardware. Furthermore, these models effectively predicted DA over a greater number of turns, demonstrating their robustness even with limited training data on long-duration tracking. This progress presents new opportunities for optimising and designing future accelerators by providing fast and accurate DA predictions, reducing the dependency on time-consuming simulations. Implementing these deep learning models in beam dynamics research represents a substantial advancement in the development of efficient surrogate models.

Future studies could extend this work by comparing the performance of the models with measured beam data from the LHC to further validate their effectiveness. Furthermore, incorporating variables that describe the collision process, such as beam separation and crossing angle, could extend the ability of the models to predict angular DA under more complex operational conditions. An alternative research direction involves crafting physics-informed models that incorporate the DA scaling law as a function of the turn number. This approach could significantly improve our ability to predict accelerator performance over time durations that are relevant for standard operations.

Author Contributions: Conceptualization, D.D.C., M.G., T.P. and F.F.V.d.V.; Methodology, D.D.C., M.G. and F.F.V.d.V.; Software, D.D.C. and F.F.V.d.V.; Formal analysis, D.D.C.; Investigation, D.D.C.; Data curation, D.D.C.; Writing—original draft, D.D.C. and M.G.; Writing— review and editing, M.G., C.E.M., T.P., S.R. and F.F.V.d.V.; Visualization, C.E.M.; Supervision, T.P., S.R. and F.F.V.d.V.; Project administration, T.P.; Funding acquisition, T.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by the Swiss Data Science Center project grant C20-10.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Errors in the Numerical Evaluation of the DA

Following [8], let us consider the phase-space volume of the initial conditions that are bounded after N turns:

$$\int\int\int\int \chi(x_1, p_{x_1}, x_2, p_{x_2})\, dx_1\, dp_{x_1}\, dx_2\, dp_{x_2}\,, \tag{A1}$$

where $\chi(x_1, p_{x_1}, x_2, p_{x_2})$ is the generalisation of the characteristic function to the 4D case; i.e., it is equal to one if the orbit starting at $(x_1, p_{x_1}, x_2, p_{x_2})$ is bounded and zero if it is not.

To exclude the disconnected part of the stability domain in the integral (A1), we have to choose a suitable coordinate transformation. Since linear motion is the direct product of rotations, the natural choice is to use the polar variables (r_i, ϑ_i), where r_1 and r_2 are linear invariants. The nonlinear part of the equations of motion adds a coupling between

the two planes; hence, it is natural to replace r_1 and r_2 with the polar variables $r\cos\alpha$ and $r\sin\alpha$, respectively

$$\begin{cases} x_1 &= r\cos\alpha\cos\vartheta_1 \\ p_{x_1} &= r\cos\alpha\sin\vartheta_1 \\ x_2 &= r\sin\alpha\cos\vartheta_2 \\ p_{x_2} &= r\sin\alpha\sin\vartheta_2\,. \end{cases} \qquad \begin{array}{l} r\in[0,+\infty] \\ \alpha\in[0,\pi/2] \\ \theta_i\in[0,2\pi] \end{array} \qquad i=1,2 \tag{A2}$$

Substituting in Equation (A1), we obtain

$$\int_0^{2\pi}\int_0^{2\pi}\int_0^{\pi/2}\int_0^{\infty}\chi(r,\alpha,\vartheta_1,\vartheta_2)\,r^3\sin\alpha\cos\alpha\;d\Omega_4\,, \tag{A3}$$

where $d\Omega_4$ represents the volume element

$$d\Omega_4 = dr\,d\alpha\,d\vartheta_1\,d\vartheta_2\,. \tag{A4}$$

Having fixed α and ϑ_1,ϑ_2, let $r(\alpha,\vartheta_1,\vartheta_2,N)$ be the last value of r whose orbit is bounded after N turns. Then, the volume of a connected stability domain is

$$A_{\alpha,\vartheta_1,\vartheta_2,N} = \frac{1}{8}\int_0^{2\pi}\int_0^{2\pi}\int_0^{\pi/2}[r_s(\alpha,\vartheta_1,\vartheta_2,N)]^4\sin 2\alpha\;d\Omega_3\,, \tag{A5}$$

where

$$d\Omega_3 = d\alpha\,d\vartheta_1\,d\vartheta_2\,. \tag{A6}$$

Note that r_s represents the amplitude of the last bounded orbit in the direction given by the angles α, ϑ_1, and ϑ_2. To ensure that stable islands disconnected from the main stable domain are excluded by this approach, the amplitude of the first unbounded orbit should be determined, and r_s is the previous stable amplitude.

We define the DA as the radius of the hyper-sphere that has the same volume as the stability domain

$$\mathrm{DA}(N) = r_{\alpha,\boldsymbol{\vartheta},N} = \left(\frac{2A_{\alpha,\boldsymbol{\vartheta},N}}{\pi^2}\right)^{1/4}. \tag{A7}$$

On a computer code, Equation (A5) is calculated considering K steps in the angle α and L steps in the angles ϑ_i and it reads

$$\mathrm{DA}(N) = \left[\frac{\pi}{2\,KL^2}\sum_{k=1}^{K}\sum_{l_1,l_2=1}^{L}[r_s(\alpha_k,\vartheta_1,\vartheta_2,N)]^4\sin 2\alpha_k\right]^{1/4}. \tag{A8}$$

The discretisation of the integral (A5) introduces numerical errors that are given by the following contributions:

- The discretisation in the angles ϑ_i gives a relative error proportional to L^{-1} corresponding to a trapezoidal integration rule [51]. A better estimate of the error (L^{-2}) requires some regularity for the derivative of the function $r_s(\alpha,\vartheta_1,\vartheta_2)$. This is probably not the case at the border of the stability domain.
- The discretisation of the angle α gives a relative error proportional to K^{-1}.
- The discretisation in radius r gives a relative error proportional to J^{-1}, where J is the number of amplitude steps.

The integration steps should produce comparable errors, i.e., $J\propto K\propto L$. In this way, one can obtain a relative error of $1/(4J)$ by evaluating J^4 orbits, i.e., NJ^4 iterates. The factor 4 in the error estimate is due to the dimensionality of the phase space. The fourth power in the number of orbits comes from the dimensionality of phase space and makes a precise estimate of the dynamic aperture very CPU time consuming.

A refined approach to improve the numerical accuracy of the triple integral implies considering the following integral

$$A_{\alpha,\boldsymbol{\vartheta},N} = \frac{1}{8}\int_0^{\pi/2} \sin 2\alpha \int_0^{2\pi}\int_0^{2\pi} [r_s(\alpha,\boldsymbol{\vartheta},N)]^4\, d\vartheta_1\, d\vartheta_2\, d\alpha\,, \tag{A9}$$

using for each angle an equally spaced sampling by means of (2^n+1) elements. In this way, we can apply iteratively the Romberg integration scheme three times, which is quite powerful for sufficiently smooth (e.g., analytic) integrands, which are integrated over intervals that contain no singularities and where the endpoints are also non-singular [52].

It is possible to reduce the size of the scanning procedure, and hence the CPU time needed, by setting the angles $\boldsymbol{\vartheta}$ to a constant value, e.g., zero, thus performing only a 2D scan over r and α. This is what is typically performed in standard numerical simulations of DA for complex accelerator lattices and is also the approach used for the numerical studies discussed in this paper. In this case, the transformation (A2) reads

$$\begin{cases} x_1 &= r\cos\alpha & \\ p_{x_1} &= 0 & r\in[0,+\infty] \\ x_2 &= r\sin\alpha & \alpha\in[0,\pi/2] \\ p_{x_2} &= 0\,, & \end{cases} \tag{A10}$$

and the original integral is transformed to

$$\int_0^{\pi/2}\int_0^{\infty} r\, dr\, d\alpha\,. \tag{A11}$$

Having fixed α, let $r(\alpha,N)$ be the last value of r whose orbit is bounded after N iterations. Then, the volume of a connected stability domain is

$$A_{\alpha,N} = \frac{1}{2}\int_0^{\pi/2} [r_s(\alpha,N)]^2\, d\alpha\,. \tag{A12}$$

In this case, we define the DA as the radius of the sphere that has the same volume as the stability domain, namely

$$\mathrm{DA}(N) = r_{\alpha,N} = \left(\frac{4A_{\alpha,N}}{\pi}\right)^{1/2}. \tag{A13}$$

When Equation (A12) is implemented in a computer code, one considers K steps in the angle α, and the dynamic aperture reads

$$\mathrm{DA}(N) = \left[\frac{1}{K}\sum_{k=1}^{K}[r_s(\alpha_k,N)]^2\right]^{1/2} \tag{A14}$$

In the case of the 2D scan, the numerical error in the computation of the integral (A12) is given by the following contributions:

- The discretisation of the angle α gives a relative error proportional to K^{-1}.
- The discretisation in the radius r gives a relative error proportional to J^{-1}, where J is the number of amplitude steps.

The integration steps should produce comparable errors, i.e., $J \propto K$. In this way, one can obtain a relative error of $1/(2J)$ by evaluating J^2 orbits, i.e., NJ^2 iterates (the factor 2 in the error estimate is due to the dimensionality of the phase space).

Appendix B. Characteristics of the Hardware Used in Our Studies

The M3 Max is a system-on-chip designed by Apple. Featuring 14 CPU cores and 30 GPU cores, along with a dedicated neural engine for machine learning tasks, the M3

Max can provide a peak performance of up to 120 teraflops in GPU computations. In our study, we evaluated this chip using an Apple MacBook Pro equipped with 96 GB of RAM. We utilised the TensorFlow-Metal plugin to leverage Metal's acceleration capabilities, optimising GPU performance for faster processing.

The V100 is a high-end GPU designed by NVIDIA for scientific computing and deep learning. It has 5120 CUDA cores, 16 GB of high-bandwidth memory, and 640 Tensor Cores for accelerating matrix computations, providing a peak of performance of 130 teraflops in GPU computations. This GPU is complemented by an Intel Xeon CPU E5-2630 v4, which provides 40 CPU cores running at 2.20 GHz. The system is also supported by 125.64 GB of RAM, enabling the efficient handling of large data sets and demanding computations.

References

1. Huang, X.; Safranek, J. Nonlinear dynamics optimization with particle swarm and genetic algorithms for SPEAR3 emittance upgrade. *Nucl. Instrum. Methods Phys. Res. Sect. A* **2014**, *757*, 48–53. [CrossRef]
2. Bengtsson, J.; Martin, I.P.S.; Rowland, J.H.; Bartolini, R. On-line control of the nonlinear dynamics for synchrotrons. *Phys. Rev. ST Accel. Beams* **2015**, *18*, 074002. [CrossRef]
3. Soutome, K.; Tanaka, H. Higher-order formulas of amplitude-dependent tune shift caused by a sextupole magnetic field distribution. *Phys. Rev. Accel. Beams* **2017**, *20*, 064001. [CrossRef]
4. Li, Y.; Cheng, W.; Yu, L.H.; Rainer, R. Genetic algorithm enhanced by machine learning in dynamic aperture optimization. *Phys. Rev. Accel. Beams* **2018**, *21*, 054601. [CrossRef]
5. Sanchez, E.A.; Flores, A.; Hernandez-Cobos, J.; Moreno, M.; Antillón, A. A novel approach using nonlinear surfaces for dynamic aperture optimization in MBA synchrotron light sources. *Sci. Rep.* **2023**, *13*, 23007. [CrossRef]
6. Sánchez, E.A.; Flores, A.; Hernández-Cobos, J.; Moreno, M.; Antillón, A. Increasing beam stability zone in synchrotron light sources using polynomial quasi-invariants. *Sci. Rep.* **2023**, *13*, 1335. [CrossRef]
7. Carmignani, N.; Carver, L.; Hoummi, L.; Liuzzo, S.; Perron, T.; White, S. Nonlinear Dynamics Measurements at the EBS Storage Ring. In Proceedings of the 67th ICFA Advanced Beam Dynamics Workshop on Future Light Sources, Lucerne, Switzerland, 27 August–1 September 2023; Volume FLS2023, p. TU4P18. [CrossRef]
8. Todesco, E.; Giovannozzi, M. Dynamic aperture estimates and phase-space distortions in nonlinear betatron motion. *Phys. Rev. E* **1996**, *53*, 4067–4076. [CrossRef]
9. Giovannozzi, M. A proposed scaling law for intensity evolution in hadron storage rings based on dynamic aperture variation with time. *Phys. Rev. Spec. Top. Accel. Beams* **2012**, *15*, 024001. [CrossRef]
10. Maclean, E.; Giovannozzi, M.; Appleby, R. Innovative method to measure the extent of the stable phase-space region of proton synchrotrons. *Phys. Rev. Accel. Beams* **2019**, *22*, 034002. [CrossRef]
11. Brüning, O.S.; Collier, P.; Lebrun, P.; Myers, S.; Ostojic, R.; Poole, J.; Proudlock, P. *LHC Design Report*; CERN Yellow Rep. Monogr; CERN: Geneva, Switzerland, 2004. [CrossRef]
12. Apollinari, G.; Béjar Alonso, I.; Brüning, O.; Fessia, P.; Lamont, M.; Rossi, L.; Tavian, L. *High-Luminosity Large Hadron Collider (HL–LHC)*; CERN Yellow Rep. Monogr.; CERN: Geneva, Switzerland, 2017; Volume 4. [CrossRef]
13. Abada, A.; Abbrescia, M.; AbdusSalam, S.S.; Abdyukhanov, I.; Abelleira Fernandez, J.; Abramov, A.; Aburaia, M.; Acar, A.O.; Adzic, P.R.; Agrawal, P.; et al. FCC–hh: The Hadron Collider: Future Circular Collider Conceptual Design Report Volume 3. Future Circular Collider. *Eur. Phys. J. Spec. Top.* **2019**, *228*, 755–1107. [CrossRef]
14. Abramov, A.; Bartmann, W.; Benedikt, M.; Bruce, R.; Giovannozzi, M.; Perez-Segurana, G.; Risselada, T.; Zimmermann, F. A new baseline layout for the FCC-hh ring. In Proceedings of the Proceedings IPAC'24, International Particle Accelerator Conference, Nashville, TN, USA, 19–24 May 2024; JACoW Publishing: Geneva, Switzerland, 2024; pp. 75–78. [CrossRef]
15. Giovannozzi, M.; Mcintosh, E. Development of Parallel Codes for the Study of Nonlinear Beam Dynamics. *Int. J. Mod. Phys. C* **1997**, *08*, 155–170. [CrossRef]
16. Giovannozzi, M.; Scandale, W.; Todesco, E. Dynamic aperture extrapolation in presence of tune modulation. *Phys. Rev.* **1998**, *E57*, 3432. [CrossRef]
17. Bazzani, A.; Giovannozzi, M.; Maclean, E.H.; Montanari, C.E.; Van der Veken, F.F.; Van Goethem, W. Advances on the modeling of the time evolution of dynamic aperture of hadron circular accelerators. *Phys. Rev. Accel. Beams* **2019**, *22*, 104003. [CrossRef]
18. Bottura, L.; Buzio, M.; Deniau, L.; Granata, V.; Li, L.; Pieloni, T.; Sanfilippo, S.; Scandale, W.; Todesco, E.; Völlinger, C.; et al. *Warm-Cold Magnetic Field Correlation in the LHC Main Dipoles*; Technical Report; CERN: Geneva, Switzerland, 2003.
19. Bottura, L.; Fartoukh, S.D.; Granata, V.; Todesco, E. A Strategy for Sampling of the Field Quality of the LHC Dipoles. In Proceedings of the Proceedings EPAC'04, European Particle Accelerator Conference, Lucerne, Switzerland, 5–9 July 2004; JACoW Publishing: Geneva, Switzerland, 2004; pp. 1606–1608.
20. Todesco, E. Field Quality and Warm-Cold Correlations in the LHC Main Dipoles. Technical Report. 2004. Available online: https://cds.cern.ch/record/725442 (accessed on 25 October 2024).
21. Sammut, N.; Bottura, L.; Micallef, J. Mathematical formulation to predict the harmonics of the superconducting Large Hadron Collider magnets. *Phys. Rev. ST Accel. Beams* **2006**, *9*, 012402. [CrossRef]

22. Barranco Garcia, J.; Pieloni, T. Global Compensation of Long-Range Beam-Beam Effects with Octupole Magnets: Dynamic Aperture Simulations for the HL-LHC Case and Possible Usage in LHC and FCC. Technical Report CERN-ACC-NOTE-2017-0036, CERN. 2017. Available online: https://cds.cern.ch/record/2263347 (accessed on 25 September 2024).
23. Skoufaris, K.; Fartoukh, S.; Papaphilippou, Y.; Poyet, A.; Rossi, A.; Sterbini, G.; Kaltchev, D. Numerical optimization of dc wire parameters for mitigation of the long range beam-beam interactions in High Luminosity Large Hadron Collider. *Phys. Rev. Accel. Beams* **2021**, *24*, 074001. [CrossRef]
24. Schenk, M.; Coyle, L.; Pieloni, T.; Obozinski, G.; Giovannozzi, M.; Mereghetti, A.; Krymova, E. Modeling Particle Stability Plots for Accelerator Optimization Using Adaptive Sampling. In Proceedings of the Proceedings IPAC'21, Campinas, Brazil, 24–28 May 2021; JACoW Publishing: Geneva, Switzerland, 2021; pp. 1923–1926. [CrossRef]
25. Casanova, M.; Dalena, B.; Bonaventura, L.; Giovannozzi, M. Ensemble reservoir computing for dynamical systems: Prediction of phase-space stable region for hadron storage rings. *Eur. Phys. J. Plus* **2023**, *138*, 559. [CrossRef]
26. Di Croce, D.; Giovannozzi, M.; Pieloni, T.; Seidel, M.; Van der Veken, F.F. Accelerating dynamic aperture evaluation using deep neural networks. In Proceedings of the 14th International Particle Accelerator Conference (IPAC'23), Venice, Italy, 7–12 May 2023; JACoW Publishing: Geneva, Switzerland, 2023; pp. 2870–2873. [CrossRef]
27. Xsuite Documentation. 2023. Available online: http://xsuite.web.cern.ch (accessed on 25 September 2024).
28. Iadarola, G.; Maria, R.D.; Lopaciuk, S.; Abramov, A.; Buffat, X.; Demetriadou, D.; Deniau, L.; Hermes, P.; Kicsiny, P.; Kruyt, P.; et al. Xsuite: An integrated beam physics simulation framework. In Proceedings of the 68th ICFA ABDW on High-Intensity and High-Brightness Hadron Beams, Geneva, Switzerland, 9–13 October 2023.
29. Di Croce, D.; Giovannozzi, M.; Krymova, E.; Pieloni, T.; Redaelli, S.; Seidel, M.; Tomás, R.; Van der Veken, F.F. Optimizing dynamic aperture studies with active learning. *J. Instrum.* **2024**, *19*, P04004. [CrossRef]
30. Grote, H.; Schmidt, F. MAD-X – An Upgrade from MAD8. In Proceedings of the Proceedings PAC'03, Portland, OR, USA, 12–16 May 2003; JACoW Publishing: Geneva, Switzerland, 2003; pp. 3497–3499. [CrossRef]
31. Deniau, L.; Burkhardt, H.; De Maria, R.; Giovannozzi, M.; Jowett, J.M.; Latina, A.; Persson, T.; Schmidt, F.; Shreyber, I.S.; Skowroński, P.K.; et al. Upgrade of MAD-X for HL-LHC Project and FCC Studies. In Proceedings of the Proceedings ICAP'18, Key West, FL, USA, 20–24 October 2018; JACoW Publishing: Geneva, Switzerland, 2019; pp. 165–171. [CrossRef]
32. De Maria, R.; Latina, A.; Schmidt, F.; Dilly, J.; Deniau, L.; Skowronski, P. Status of MAD-X V5.09. In Proceedings of the Proceedings IPAC'23, International Particle Accelerator Conference, Venice, Italy, 7–12 May 2023; JACoW Publishing: Geneva, Switzerland, 2023; pp. 3340–3343. [CrossRef]
33. MAD—Methodical Accelerator Design. Available online: https://mad.web.cern.ch/mad/ (accessed on 25 September 2024).
34. Schmidt, F.; Forest, E.; McIntosh, E. *Introduction to the Polymorphic Tracking Code: Fibre Bundles, Polymorphic Taylor Types and "Exact Tracking"*; Technical Report; CERN: Geneva, Switzerland, 2002.
35. Schmidt, F. MAD-X PTC Integration. In Proceedings of the Proceedings PAC'05, Particle Accelerator Conference, Knoxville, TN, USA, 16–20 May 2005; JACoW Publishing: Geneva, Switzerland, 2005; pp. 1272–1274.
36. MAD-X-PTC Documentation. 2006. Available online: https://madx.web.cern.ch/doc/PTC_reference_manual.pdf (accessed on 25 October 2024).
37. Bazzani, A.; Servizi, G.; Todesco, E.; Turchetti, G. *A Normal form Approach to the Theory of Nonlinear Betatronic Motion*; CERN Yellow Reports Monographs; CERN: Geneva, Switzerland, 1994. [CrossRef]
38. LeCun, Y.A.; Bottou, L.; Orr, G.B.; Müller, K.R. Efficient BackProp. In *Neural Networks: Tricks of the Trade*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 9–48. [CrossRef]
39. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv* **2016**. [CrossRef]
40. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines Vinod Nair. In Proceedings of the ICML, Haifa, Israel, 21–24 June 2010; Volume 27, pp. 807–814.
41. Keras Tuner. Available online: https://github.com/keras-team/keras-tuner (accessed on 25 September 2024).
42. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arXiv:1810.04805.
43. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2023**, arXiv:1706.03762.
44. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [CrossRef]
45. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
46. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
47. Dozat, T. Incorporating Nesterov momentum into Adam. In Proceedings of the 4th International Conference on Learning Representations (ICLR'16), San Juan, Puerto Rico, 2–4 May 2016.
48. Apple Inc. Apple M3 MAX Chip. 2021. Available online: https://www.apple.com/newsroom/2023/10/apple-unveils-m3-m3-pro-and-m3-max-the-most-advanced-chips-for-a-personal-computer/ (accessed on 25 September 2024).

49. NVIDIA Corporation. NVIDIA V100. 2020. Available online: https://www.nvidia.com/en-us/data-center/v100/ (accessed on 25 September 2024).
50. CERN HT-Condor Documentation. 2023. Available online: https://batchdocs.web.cern.ch/local/submit.html (accessed on 25 September 2024).
51. Stoer, J. *Introduction to Numerical Analysis*; Springer: New York, NY, USA, 1980.
52. Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. *Numerical Recipes: The Art of Scientific Computing*; Cambridge University Press: Cambridge, UK, 2007.