POLITECNICO DI MILANO

Master's Degree in Mathematical Engineering

Master's Degree Thesis

Echo State Networks for Dynamic Aperture Prediction

Supervisors

Prof. Luca BONAVENTURA

Dr. Barbara DALENA

Candidate

Maxime CASANOVA

July 2022

Summary

In this thesis, we investigate the ability of an ensemble reservoir computing approach based on an Echo State Network to predict the Dynamic Aperture, which is a measure of the region of stable motion of a particle after a certain number of turns in a circular collider. The approach has been tested first using Dynamic Aperture data generated by a realistic model of the High-Luminosity LHC up to 10^5 particle revolutions. Then, to pursue the analysis further, artificial Dynamic Aperture data have been generated to a larger number of turns (i.e, 10^7) using the 4D Hénon Map, a simplified model of a circular accelerator. The optimization of the Echo State Network hyperparameters and the importance of the validation data have been discussed. We show that the prediction errors of the proposed approach, when supplied with relevant validation data, is in average lower than those obtained by using analytical models based on the Nekhoroshev theorem.

Acknowledgements

I would like to first thanks Dr. Barbara Dalena and Prof. Luca Bonaventura for all their explanations regarding colliders, dynamic aperture and ensemble approaches. I am also very grateful for their involvement in the writing of this thesis with their many feedbacks. Thanks to them, i learned a lot both on a scientific point of view (through new concept in physics, statistics, machine learning) but also on a personal point of view. Eventually, I am also grateful to Massimo Giovannozzi for our several discussions on scaling laws and Hénon Map.

Table of Contents

List of Tables

List of Figures

Acronyms

CERN

European Organization for Nuclear Research

DA

Dynamic Aperture

ESN

Echo State Network

ESP

Echo State Property

HL-LHC

High Luminosity Large Hadron Collider

LHC

Large Hadron Collider

MSE

Mean Square Error

RRMSE

Relative Root Mean Square Error

1 Introduction

In 1919, Ernest Rutherford discovered that nitrogen atoms could be split by bombarding them with alpha particles (i.e, particles composed of two protons and neutrons bound together) emitted by radioactive sources. This discovery may be considered as the starting point of the development of more powerful machines used to propel particles at higher intensity to study the atomic structure at smaller scales. In 1929, Ernest Lawrence developed the first cyclotron used to accelerate particles at higher energies than those produced by radioactive sources. In a cyclotron, particles move along a spiral path guided by a magnetic field produced by the magnets of the accelerators. Derived from the cyclotrons, the first synchrotrons were introduced a few years later, with the main improvement that the magnetic field evolved with time during the accelerating process of the particles [1]. Nowadays, the most powerful circular collider is a synchrotron, the Large Hadron Collider (LHC). In order to build such machines, design studies through tracking simulations from realistic models of colliders are required [2]. For instance, tracking simulations are performed to compute the Dynamic Aperture (DA) in order to study to effect of the magnetic fields which can be responsible of the unstable motion of a particle [3].

The DA represents a measure of the region of stable motion of a particle after a certain number of turns in a circular accelerator. The possible sources of unstable motions are magnetic fields and imperfections in the placement of elements imperfections [3]. Typically, DA is used to define tolerances on magnetic field quality and non-linear corrections schemes in the design phase of the accelerator, as well as to verify the effect of corrections in existing machines. Currently, DA estimation for high-energy hadron colliders such as the LHC is performed through computer simulations, which are rather computationally intensive, in particular for large number of revolutions in the accelerator (i.e, $\geq 10^6$ turns) [4].

In the last decade, the use of neural networks has greatly increased in a large number of research areas. For instance, neural networks are used for speech recognition [5] or for the forecasting of wind power [6]. Among neural network techniques, the most common architectures are feedforward [7], convolutional [8] and recurrent [9] neural networks. Feedforward neural networks are composed by neurons linked by connections to other neurons only. They provide only input-output relationships and can approximate very large classes of functions. Instead, recurrent neural networks are composed by neurons linked by connections to themselves and to other neurons. They preserve an internal state that is a nonlinear transform of the input signal and can therefore be considered as dynamical systems.

Echo State Networks (ESN) are one of the classes of recurrent neural network using the reservoir computing approach. This approach has the main advantage of reducing significantly the computational time required by the training process, which is performed to find the optimal parameters (called weights) of a neural network. Indeed, the peculiarity of the ESN is that the training is performed usually by linear regression [10] to compute the weights used to project the reservoir state onto the output state, so that no back-propagation is needed. Back-propagation [11] refers to the numerical procedure (usually based on the stochastic gradient method) employed for the training of feedforward networks, which is responsible of a major share of its computational cost. ESN has also been proved to be an universal approximant of dynamical systems [12]. Thus, ESN is a natural candidate for the prediction of DA for a large number of turns.

The thesis is organized as follows. In Chapter 2 we present some general concepts about circular colliders and we identify the non linear imperfections that may cause a reduction of the region of stable motion of particles. We introduce a definition of the DA and the main equations used to compute it in the case of a 4D phase space. We also recall the 4D Hénon Map used as an idealized model of accelerator on which to test our prediction approach. Some analytical models taken from the literature and used to extrapolate the DA are also introduced. Then, in Chapter 3, we introduce the framework of the continuous time leaky ESN that will be used in the thesis. We also discuss the Echo State Property (ESP) and review a sufficient condition which can be applied in practice to satisfy it. Chapter 4 is devoted both to the presentation of the different sets of DA data used to train and test the proposed ensemble approach based on ESN and to the description of the simulation setup employed in this thesis. In particular, we introduce the ensemble validation procedure used to tune the model hyperparameters as well as the algorithm used to perform the analysis. Finally, in Chapter 5, we present the results regarding the ability of the proposed approach to predict the DA and discuss possible future developments.

2 Non Linear Imperfections in Circular Colliders and Dynamic Aperture

We start this Chapter by introducing some considerations about non linear imperfections in circular colliders, which can be responsible of a reduction of the region of stable motion of particles [3]. The size of this region is taken to be equal to the DA. We then recall a DA definition proposed in Ref. [13] and introduce the general equations used to estimate it, in the case of a 4D phase space. In particular, we show how one can simplify the problem to considering a 2D case under some assumptions. Finally, we introduce the 4D Hénon map used to generate cheaply artificial DA data at a large number of turns, as well as scaling laws proposed in Ref. [14] to forecast the time evolution of the DA.

2.1 Generalities

In this Section, the basic concepts concerning circular colliders are introduced, as well as the magnetic field errors of accelerator magnets, which are the main sources of non linear imperfections for high energy proton colliders [3]. The stability of the colliders with respect to these imperfections must be addressed in order to both assess the performance of the current colliders and to design the future ones.

2.1.1 Basic Concepts on Colliders

Particle accelerators are machines which propel charged particles to very high speeds and energies using electromagnetic fields. In particular, a stream of particles called beam travels through the accelerator, moving with a velocity close to the light speed. Colliders are particle accelerators in which the collision of two opposing particle beams occurs. The most famous collider is the Large Hadron Collider (LHC), which was built by the European Organization for Nuclear Research (CERN) [15] between 1998 and 2008 . Hadrons are subatomic particles such as the protons and neutrons. The LHC is currently the largest (27 km) and highest energy circular collider. An upgrade of the LHC, the High Luminosity LHC (HL-LHC), is a challenging project scheduled for around 2025, whose main goal is to provide more accurate measurements of new particles and observe new processes occurring below the current sensitivity level. Other smaller colliders exist, such as the Proton Synchrotron (PS) and the Super Proton Synchrotron (SPS), which are currently used to inject high intensity proton beams to the LHC. Eventually, the Future Circular Collider (FCC) with a circumference of 100 km is a proposed post-LHC collider whose goal is to push the energy and intensity frontiers of particle colliders to reach collision energies of 100 Tera electronvolt (TeV). For comparison, the LHC is currently able to reach collision energies of 14 TeV. Figure 2.1 shows a comparison of the size of the different colliders.

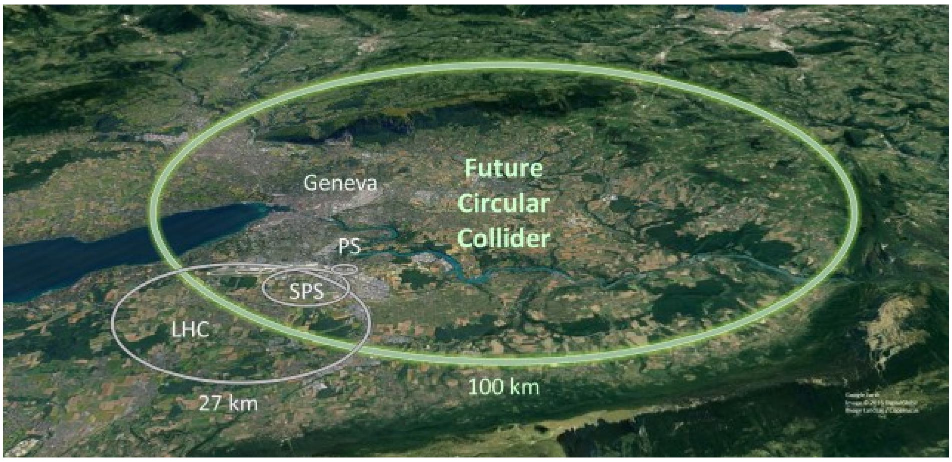


Figure 2.1: Picture of the different colliders (PS, SPS, LHC and FCC).

2.1.2 Magnetic Field Errors

In circular colliders, the sources of non linear imperfections can be many, in particular related to the presence of magnetic field errors in accelerator magnets. The magnetic field created by the magnets can be described in terms of an expansion of the following kind [16], [17]:

$$B_y + iB_x = B_{ref} \sum_{n=1}^{\infty} (b_n + ia_n) \Big(\frac{x+iy}{R_{ref}} \Big)^{n-1} \tag{2.1}$$

where B_y, B_x and B_{ref} are the transverse magnetic field components and the reference field, respectively and b_n and a_n are the normal and the skew harmonics component. The subscript $n = 1$ refers to dipole, $n = 2$ to a quadrupole and so on.

In fact, Eq.(2.1) is one possible representation of the magnetic field of the accelerator element, which satisfies the Maxwell equations

$$\mathrm{div}(B) = 0 \quad \mathrm{rot}(B) = 0.$$

. In particular, the choice of using complex number is based on some computational considerations.
In all dipole or quadrupole magnets, a certain number of higher harmonic components ($n > 2$, i.e, sextupole, octupole...) are present. These harmonics are called non linear imperfections [3]. The estimation of these imperfections is crucial, since they can have a direct impact on the performance of the collider. Indeed, they can be responsible of a reduction of the region of stable motion of particles in the collider. Based on the experience of LHC, the multipole harmonics of the main magnetic fields can be modelled as a sum of three contributions [18].

$$b_n = b_{nS} + \xi_U 1.5 b_{nU} + \xi_R b_{nR} \tag{2.2}$$

$$a_n = a_{nS} + \xi_U 1.5 a_{nU} + \xi_R a_{nR} \tag{2.3}$$

where ξ_U and ξ_R denote pseudo-random numbers with Gaussian distribution truncated at 1.5 and 3 σ (i.e, standard deviation). The first one is the systematic magnetic field error (b_{nS} and a_{nS}), which depends on the design geometry, the second one is the magnetic field error (b_{nU} and a_{nU}) coming from the magnet assembly and the last one is the random magnetic field error (b_{nR} and a_{nR}), which depends on the magnet.

2.2 Dynamic Aperture in a 4D Phase Space

We introduce the definition and estimation of the DA for a 4D phase space [13], and how such estimation can be simplified in the case of a 2D scanning [19]. We work in the 4D phase space because we are interested in defining the area of stability in the plane transverse to the direction of motion. In fact, the effect of the coupling between the longitudinal and transverse plane can be neglected [18] because static magnetic fields have a small effect along the longitudinal axis. The computation of the DA is useful since it can be seen both as a figure of merit to design future colliders and as powerful indicator to assess the performance of the current colliders. Indeed, the evaluation of DA allows to estimate an upper bound for the unwanted non linear imperfections defined in Section 2.1.2.

2.2.1 Definition

We recall a definition proposed in [13] of the DA in the case of a 4D phase space $(x_1, p_{x_1}, x_2, p_{x_2})$, where p_{x_1} and p_{x_2} denote the momenta associated with x_1 and x_2 in the plane transverse to the longitudinal direction. The phase space is defined as the space where all the possible states of a system (in position and momentum) are represented. The estimation of the DA is associated to the computation of the volume in phase space of the set of initial conditions that generates bounded orbits after N turns, or revolutions, in the collider. The volume of this set of initial conditions in phase space whose associated orbits are bounded after N turns is defined as:

$$\iiint \chi(x_1, p_{x_1}, x_2, p_{x_2})\, dx_1\, dp_{x_1}\, dx_2\, dp_{x_2} \tag{2.4}$$

where $\chi(x_1, p_{x_1}, x_2, p_{x_2})$ denotes the characteristic function equal to 1 if the orbit starting at (x_1, x_2) with momenta p_{x_1}, p_{x_2} is bounded and 0 if not.
We define the DA as the radius of the circle whose area is equal to the area of the stability domain. The stability domain represents the area of the connected region of initial conditions whose associated orbits are bounded after a given number of turns N.

2.2.2 DA Estimation

We first consider the polar variables $(r_1, \theta_1, r_2, \theta_2)$, where r_1 and r_2 are the radii of polar coordinates in a four dimensional space, seen as a product of two two dimensional spaces. We then replace r_1 and r_2 by the two dimensional polar variables $r\ cos(\alpha)$ and $r\ sin(\alpha)$, so that we rewrite Eq. (2.4) as:

$$\int_0^{2\pi}\int_0^{2\pi}\int_0^{\pi/2}\int_0^{\infty} \chi(r, \alpha, \theta_1, \theta_2) r^3 sin(\alpha) cos(\alpha)\, dr\, d\alpha\, d\theta_1\, d\theta_2 \tag{2.5}$$

where $r \in [0, \infty[$, $\theta_1, \theta_2 \in [0, 2\pi[$ and $\alpha \in [0, \pi/2]$.

The volume of the connected stability domain is defined as:

$$A_{\alpha,\theta,N} = \frac{1}{8}\int_0^{2\pi}\int_0^{2\pi}\int_0^{\pi/2} r(\alpha, \theta, N)^4 sin(2\alpha)\, d\alpha\, d\theta_1\, d\theta_2 \tag{2.6}$$

where $\theta = (\theta_1, \theta_2)$ and $r(\alpha, \theta, N)$ denotes the largest value of r such that an orbit starting at distance r from the axis is bounded after N turns.

Using Eq. (2.6), we can estimate the DA as function of the number of turns N:

$$D_{\alpha,\theta,N} = \left(\frac{2A_{\alpha,\theta,N}}{\pi^2}\right)^{1/4}. \tag{2.7}$$

2.2.3 Dimensionality Reduction

It is possible to reduce the CPU time of the simulation by setting $\theta = 0$, so that the scanning of the initial particle amplitudes is only performed over r and α. This is typically what can be done by using the SixTrack code, developed at

CERN [20], which is optimized for long term tracking simulations of high energy hadron accelerators. If this simplification is introduced, the volume of a connected stability domain is redefined as:

$$A_{\alpha,N} = \frac{1}{2}\int_0^{\pi/2} r(\alpha, N)^2 \, d\alpha, \tag{2.8}$$

and the DA can be expressed as a function of the number of turns N:

$$D_{\alpha,N} = \left(\frac{4A_{\alpha,\theta,N}}{\pi}\right)^{1/2}. \tag{2.9}$$

However, in practice, taking the average of the stable radii is found to be a better indicator, as reported for example in Ref. [21], so Eq. (2.9) can be simplified as follow:

$$D_{\alpha,N} = \frac{2}{\pi}\int_0^{\pi/2} r(\alpha, N) \, d\alpha \tag{2.10}$$

The approximate computation of Eq. (2.10) is straightforward and can be performed for example by considering K steps in the angle α. Due to the normalization factor in Eq. (2.10), this yields

$$D_{\alpha,N} = \frac{1}{K}\sum_{k=1}^{K} r(\alpha_k, N). \tag{2.11}$$

Finally, a simplified formula for the numerical error of the estimation of DA after N turns can can be obtained in Ref. [21]:

$$\Delta D_{\alpha,N} \approx \frac{\Delta r}{2} \tag{2.12}$$

where Δr is the step size in r.

2.3 4D Hénon Map

The 4D Hénon Map is a well-known dynamical system that displays a rich dynamical behaviour as presented in Ref. [14]. The model is defined as,

$$\begin{pmatrix} x_1^{(n+1)} \\ p_{x_1}^{(n+1)} \\ x_2^{(n+1)} \\ p_{x_2}^{(n+1)} \end{pmatrix} = L \begin{pmatrix} x_1^{(n)} \\ p_{x_1}^{(n)} + \left(x_1^{(n)}\right)^2 - \left(x_2^{(n)}\right)^2 + \mu\left(\left(x_1^{(n)}\right)^3 - 3\left(x_2^{(n)}\right)^2 x_1^{(n)}\right) \\ x_2^{(n)} \\ p_{x_2}^{(n)} - 2x_1^{(n)}x_2^{(n)} + \mu\left(\left(x_2^{(n)}\right)^3 - 3\left(x_1^{(n)}\right)^2 x_2^{(n)}\right) \end{pmatrix} \tag{2.13}$$

where the upper script $^{(n)}$ denotes the discrete time and L is a matrix given by the direct product of two 2D rotations R,

$$L = \begin{pmatrix} R(w_{x_1}^{(n)}) & 0 \\ 0 & R(w_{x_2}^{(n)}). \end{pmatrix} \tag{2.14}$$

Here the linear frequencies vary with the discrete time according to

$$w_{x_1}^{(n)} = w_{x10}\Big(1 + \epsilon \sum_{k=1}^{m} \epsilon_k cos(\Omega_k n)\Big) \tag{2.15}$$

$$w_{x_2}^{(n)} = w_{x20}\Big(1 + \epsilon \sum_{k=1}^{m} \epsilon_k cos(\Omega_k n)\Big) \tag{2.16}$$

where ϵ denotes the tune modulation and ϵ_k and Ω_k are fixed according previous studies on the SPS Ref. [14].

The 4D Hénon Map represents a simplified model of a circular accelerator. In particular, it models the effects of a sextupole and octupole magnet on the particle motion at each turn through the quadtratic and cubic non linear terms. Sextupole and octupole magnets are composed respectively of six and eight magnetic poles arranged in a configuration of alternating north and south poles around an axis. Setting $\mu = 0$ is equivalent to consider the sextupole effects only. The 4D Hénon Map will be used here to generate cheaply DA data for large numbers of turns (i.e, 10^7 turns).

2.4 Scaling Laws

The estimation of the DA for a large number of turns can have a significant computational cost. Thus, the development of cheaper models allowing to forecast accurately the DA up to a large number of turns is a field of strong interest. We present here some analytical models proposed in the literature, which are based on the Nekhoroshev theorem, as introduced in Ref. [14]. In particular, these analytical models have been benchmarked using numerical simulation of the 4D Hénon map introduced in Section 2.3 as well as a realistic representation of the beam dynamics in the LHC obtained using the SixTrack code.

A first analytical model for the description of the time evolution of the DA has been proposed in the form of:

$$\textbf{SL}: \quad D_N = \rho_* \Big(\frac{\kappa}{2e}\Big)^{\kappa} \frac{1}{ln(N)^{\kappa}} \tag{2.17}$$

where ρ_* and κ are fitting parameters.

Another version of the same model can be developed based on the Lambert function W_{-1} as discussed in Ref. [14]. In this case, the proposed analytical model reads:

$$\textbf{SL2}: \quad D_N = \rho_* \frac{1}{\Big(-2e\lambda W_{-1}\big(-\frac{1}{2e\lambda}\big(\frac{\rho_*}{6}\big)^{1/\kappa}\big(\frac{8}{7}N\big)^{-1/(\lambda\kappa)}\big)\Big)^{\kappa}} \tag{2.18}$$

where the fitting parameters are ρ_*, κ and λ. However, in Ref. [14], it is suggested to fix $\lambda = 1/2$.

For practical applications, the series expansion of W_{-1} can be used:

$$W_{-1}(x) = ln(-x) - ln(-ln(-x)) + \sum_{l=0}^{\infty}\sum_{m=1}^{\infty} c_{lm} ln(-x)^{-l-m} ln(-ln(-x))^m \tag{2.19}$$

where

$$c_{lm} = \frac{(-1)^l}{m!}\begin{bmatrix} l+m \\ l+1 \end{bmatrix} \tag{2.20}$$

and the symbol in square brackets represents a Stirling cycle number [22]. The series expansion can be truncated at a certain finite order to retain only the lowest order terms.

3 Continuous Time Leaky Echo State Network

In this Chapter, we present first some general concepts about ESN. Then, we introduce the mathematical framework of the continuous time leaky ESN applied for supervised learning tasks. Finally, we recall the definition of the Echo State Property (ESP) and a sufficient condition to guarantee this property that can be used in practice for applications of ESN.

3.1 Generalities

ESN are a type of Recurrent Neural Network using the Reservoir Computing approach. In this type of neural networks, the data input is fed into connected to a random and non trainable network, called the reservoir. This reservoir is eventually connected via trainable weights to the ESN output. The use of ESN for time series prediction has become widespread due to its cheap training process and its remarkable performance in dynamical system modeling [23]. Contrary to feedforward neural networks, ESN do not suffer from vanishing or exploding gradients (caused by the fact that the neural networks parameters remain almost constant or lead to numerical instabilities) that induce bad performance of the training algorithm [24].

3.2 ESN definition

We introduce the definition of the continuous time leaky ESN as in Ref. [25]. We consider the case of networks with continuous time t, K inputs, N_r reservoir neurons and L outputs. We define by $u = u(t) \in \mathbb{R}^K$ and $x^{target} = x^{target}(t) \in \mathbb{R}^L$ the training input and target output of the ESN, respectively. The ESN output is denoted by $x^{out} = x^{out}(t) \in \mathbb{R}^L$, while the internal reservoir activation state is given by $x = x(t) \in \mathbb{R}^{N_r}$, the input weight matrix $W^{in} \in \mathcal{M}_{N_r \times K}(\mathbb{R})$, the reservoir weight matrix $W \in \mathcal{M}_{N_r \times N_r}(\mathbb{R})$ and the output weight matrix $W^{out} \in \mathcal{M}_{L \times (N_r+K)}(\mathbb{R})$. In this way, the continuous time dynamics of a leaky ESN is given by:

$$\begin{aligned} \frac{dx}{dt} &= \frac{1}{c}(-ax + f(W^{in}u + Wx)) && (3.1) \\ x^{out} &= g(W^{out}[x;u]) && (3.2) \end{aligned}$$

where c is a global time constant, a the leaking rate, f a sigmoid function, g the output activation function and [.;.] denotes vector concatenation.

Eq. (3.1) can be discretized in time e.g. by the explicit Euler method using k_{train} time steps of size Δt. In this way, k_{train} corresponds to the number of training data. This yields the following discretized network update equation:

$$\begin{aligned} x_k = F(x_{k-1}, u_k) &= \left(1 - a\frac{\Delta t}{c}\right) x_{k-1} + \frac{\Delta t}{c} f(W^{in}u_k + Wx_{k-1}) && (3.3) \\ x_k^{out} &= g(W^{out}[x_k; u_k]) && (3.4) \end{aligned}$$

where x_k denotes the update of the reservoir activation state at the discrete time k.

Remark 3.1 *The leaking rate a of the reservoir nodes can be regarded as the speed of the reservoir update dynamics.*

In the case of a linear readout (i.e, g is the identity function), we can rewrite Eq. 3.3 in matrix notation as:

$$X^{out} = W^{out}X \qquad (3.5)$$

where $X^{out} \in \mathbb{R}^{L \times k_{train}}$ contains the L ESN outputs at every time step $k = 1, ..., k_{train}$ and where $X \in \mathcal{M}_{(N_r+K)\times k_{train}}(\mathbb{R})$ contains the concatenation of the training input and reservoir activation state at every $k = 1, ..., k_{train}$:

$$X = \begin{pmatrix} u_1 & \cdots & u_{k_{train}} \\ x_2 & \cdots & x_{k_{train}+1} \end{pmatrix} \qquad (3.6)$$

Finding the optimal output weight matrix W^{out} that minimizes the square error between x^{out} and x^{target} is done by solving the following minimization problem:

$$W^{out} = \operatorname{argmin} J(W^{out}) = \operatorname{argmin} \frac{1}{L}\sum_{i=1}^{L}\left(\sum_{k=1}^{T}(x_{ik}^{out} - x_{ik}^{target})^2 + \beta\|w_i^{out}\|^2\right) \qquad (3.7)$$

where J denotes the cost function that we want to minimize and $\|w_i^{out}\|$ the Euclidean norm of the ith row of W^{out}.

The solution of the minimization problem stated in Eq.(3.7) can be found efficiently using linear regression with Tikhonov (or ridge) regularization [26]:

$$W^{out} = X^{target} X^T (XX^T + \beta I)^{-1} \tag{3.8}$$

where $.^T$ denotes the transpose, $I \in \mathcal{M}_{(N_r+K)\times(N_r+K)}(\mathbb{R})$ the identity matrix and $X^{target} \in \mathcal{M}_{L\times k_{train}}(\mathbb{R})$ the output target matrix which contains the L ESN target outputs at every time step k.

Remark 3.2 *Using the regularization parameter β allows to get a compromise between having a small training error and small output weights* [27]. *In other words, it prevents from overfitting. Also, it may be used to prevent numerical instabilities occurring in the inversion of XX^T.*

The learning is performed on the so called *training set*, which contains k_{train} training data. A sketch of the training phase of the ESN is provided in Figure 3.1.

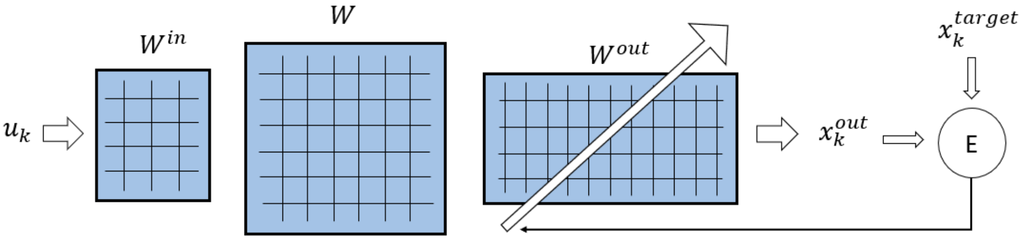


Figure 3.1: Sketch of the training procedure for the leaky ESN. The size of the matrices have been selected arbitrarily. E denotes the square error between x_k^{out} and x_k^{target}, $k = 1, ..., k_{train}$.

After the training, the ESN is usually validated using k_{val} new data and tested using k_{test} data. The validation and test procedures are detailed in Chapter 4. Also, as stated in Eq. (3.7), only the output weight matrix W^{out} is trained, while the input and reservoir matrices W^{in} and W are generated randomly, as explained in detail in Chapter 4.

3.3 The Echo State Property

An important prerequisite for the output-only training described in the previous section is the so called Echo State Property (ESP), which guarantees that initial conditions have an effect that vanishes over time. We use the work in Ref. [28] to recall the definition of the ESP and a new sufficient condition that can be used in practice. Satisfying the ESP allows to guarantee that the reservoir activation state

x_{k-1} is uniquely determined by any left-infinite input sequence $..., u_{k-2}, u_{k-1}$. In order to define the ESP, we require the *compactness condition*, that is, we assume that the states and the inputs belong to compact sets $X \in \mathbb{R}^{N_r}, U \in \mathbb{R}^K$ and that $F(x_{k-1}, u_k) \in X$ and $u_{k-1} \in U$, $\forall k \in \mathbb{Z}$.

Remark 3.3 *In practice, the ESN inputs will always be bounded so that the compactness of U is guaranteed. Also, in the case of bounded sigmoid functions f such as* tanh*, the state space X is compact too.*

We now define $U^{-\infty} := \{u^{-\infty} = (..., u_{-1}, u_0) \,|\, u_k \in U \ \forall k \in \mathbb{Z}\}$ and $X^{-\infty} := \{x^{-\infty} = (..., x_{-1}, x_0) \,|\, x_k \in X \ \forall k \in \mathbb{Z}\}$ which are the sets of infinite left input and reservoir activation state sequences.

Definition 3.1 (ESP). A network $F : X \times U \to \mathrm{X}$ (with the *compactness condition*) has the Echo State Property with respect to U if for any left input sequence $u^{-\infty} \in U^{-\infty}$ and any two state sequences $x^{-\infty}, y^{-\infty} \in X^{-\infty}$ compatible with $u^{-\infty}$ (i.e, $x_k = F(x_{k-1}, u_k), \forall k \leq 0$), it holds that for all $k \geq 0, \|x_k - y_k\| \leq \delta_k$, where δ_k denotes a small value.

For practical applications, Definition 3.1 is not really useful. Thus, we introduce the following Theorem 3.1 that should be used in practice. It provides a sufficient condition for satisfying the ESP in the case of the leaky ESN:

Theorem 3.1 (Sufficient condition of the ESP). If the spectral radius of the matrix

$$\tilde{W} = \frac{\Delta t}{c}|W| + \left(1 - a\frac{\Delta t}{c}\right) I$$

is smaller than 1, i.e. $\rho(\tilde{W}) < 1$, then the leaky ESN with $f = \tanh$ has the ESP for all inputs.

This condition is only sufficient but not necessary. In other words, setting $\rho(\tilde{W}) \geq 1$ does not necessarily lead to bad performance of the leaky ESN.

4 Dynamic Aperture Data and Simulation Setup

This Chapter is dedicated to the introduction of the different sets of Dynamic Aperture data used in the thesis, as well as to the validation and testing procedure used for our proposed predictive approach based on ESN.

4.1 The Dynamic Aperture Data

This first Section is dedicated to the presentation of the two datasets used to test our predictive approach. The first consists of data obtained from a full numerical simulation of the HL-LHC, while the second is a more extended dataset, generated using the 4D Hénon map introduced in Section 2.3. In particular, we present how we split our two DA datasets into a *training set*, a *validation set* and a *test set*.

4.1.1 The DA Datasets

HL-LHC dataset The first dataset of DA values has been generated using a realistic model of the HL-LHC [3]. The DA is estimated using Eq. (2.11) based on the output of the SixTrack code [20]. Also, the associated DA discretization error $\Delta D_{\alpha,N}$ is estimated using Eq. (2.12). These data are grouped into 60 datasets (called seeds). Each seed corresponds to a proper machine configuration, which is defined by different randomly distributed magnetic field errors b_{nR} and a_{nR} as presented in Eq. (2.2). The tracking has been performed up to $N = 10^5$ turns, where K_r=14 radii r and K_α=11 angles α have been scanned. The DA data corresponding to all the 60 seeds are plotted in Figure 4.1.
Since the discrete time ESN defined in Eq.(3.3) uses a constant time step, the data are considered as the values of a piecewise constant function of time, so as to allow for use of the data at all the discrete time levels used by the ESN. The graphs of the 60 piecewise constant functions defined by the seeds are shown in Figure 4.2 and contain 10^3 data points.

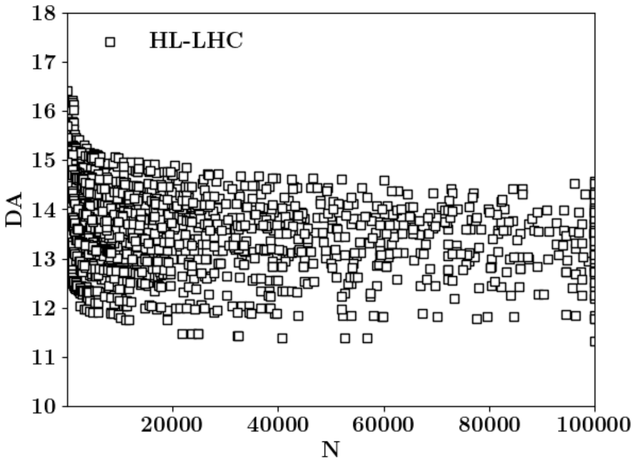


Figure 4.1: Graphs of the 60 seeds containing the original DA data of the HL-LHC realistic model. The size of the white black squares corresponds to the associated DA error $\Delta D_{\alpha,N}$.

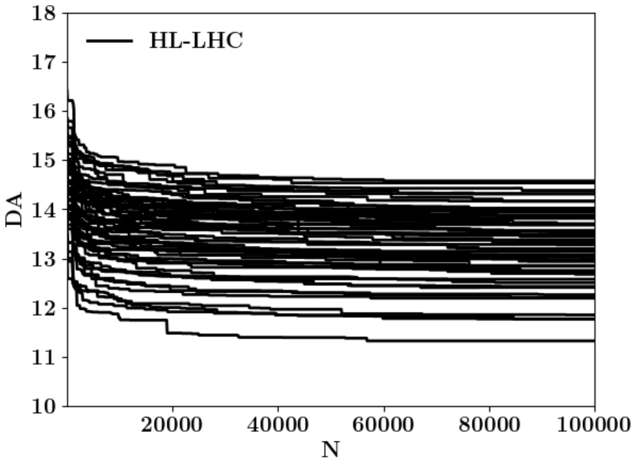


Figure 4.2: Graphs of the 60 piecewise constant functions defined by the seeds of the original DA data of the HL-LHC realistic model.

4D Hénon Map dataset The second dataset is obtained by a simulation of the 4D Hénon Map system introduced in Section 2.3. The tracking has been performed up to $N = 10^7$ turns for 20 different values of the modulation amplitude ϵ and 3 different values of μ to generate a total of 60 different cases. We varied ϵ between

5.10^{-4} and 1.10^{-2} for $\mu = 0$, $\mu = 0.2$ and $\mu =$ - 0.2. The DA is computed using the same equations introduced in Section 2.2. Also, we decided to use the same grid for the scan of the radii and angles to the HL-LHC case, so that we have scanned through K_α=11 angles and K_r=14 radii. As previously, we built 60 piecewise constant functions based on 10^3 data points. In Appendix B, we show the DA where the scan has been performed for larger number of angles K_α and radii K_r.

4.1.2 Training, Validation and Test data

The *training set* is used to train the ESN and find the optimal output weight matrix W_{out} using the ridge regression procedure presented in Eq.(3.8). The *validation set* is used to find the best hyperparameters, according to an ensemble procedure that will be explained in greater detail in Section 4.2. Finally, the *test set* is used to demonstrate the predictive capability of the ESN approach on data that was not used previously in the training or validation processes.

HL-LHC dataset For each seed, we use the first $k_{train} = 400$ data points (i.e, from 10 to 4.10^4 turns) to train the ESN. Then, we use the next $k_{val} = 100$ data points (i.e, from 4.10^4 to 5.10^4 turns) for the validation. The remaining 500 data points (i.e, from 5.10^4 to 1.10^5 turns) are used to test our predictive approach. In other words, the number of test data is $k_{test} = 500$. A picture of the DA data used to train and test can be see in Figure 4.3.

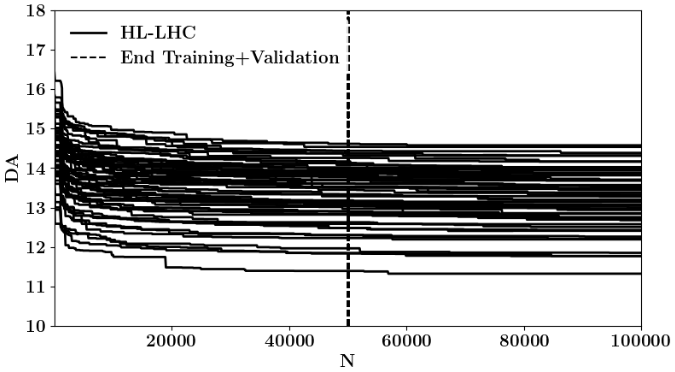


Figure 4.3: Splitting of the 60 seeds into a training, validation and test set.

4D Hénon Map dataset As for the HL-LHC case, we use the DA data from 10 to 5.10^4 turns to train and validate our approach. However, since we generated DA until a longer number of turns, we decide to test our predictive approach from

5.10^4 to 1.10^7 turns. A picture of the DA data used to train and test can be seen in Figure 4.4.

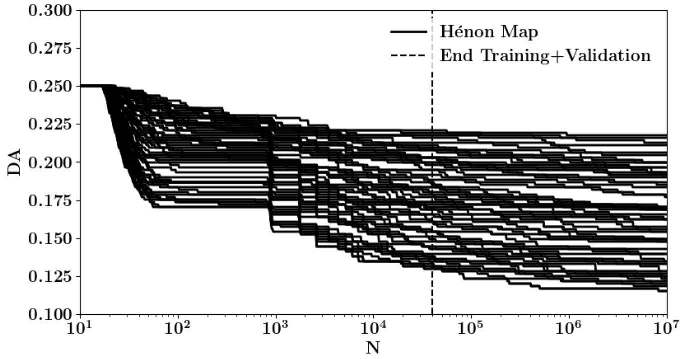


Figure 4.4: Splitting of the 60 cases into a training and test set.

4.2 Simulation Setup

We present the simulation setup used to train and test an ensemble predictive approach based on ESN inspired by [6] and [29]. We first describe the validation procedure used to tune the hyperparameters required for the complete definition of the ESN. Then, we introduce the test procedure performed to assess the predictive ability of the proposed approach.

4.2.1 Validation Procedure

The proposed validation procedure used to tune the hyperparameters of our ESN model can be described as follows. After defining what is an hyperparameter, we perform a short sensitivity analysis using the HL-LHC dataset in order to study the ESN sensitivity with respect to its hyperparameters. In other words, we try to understand which hyperparamters may be fixed and which must be tuned. Based on this sensitivity analysis, we finally present the ensemble validation procedure used to tune only the most sensitive hyperparameters.

4.2.1.1 Generalities

Hyperparameters are parameters that we need to fix to complete the definition of a predictive model. The performance of the model depends strongly on the choice of their values. In Chapter 3, we have already introduced several hyperparameters such as the leaking rate a, the regularization parameter β, the reservoir size N_r

and the spectral radius ρ of $\tilde{W}$. In particular, we show how ρ can can be fixed using Algorithm 1. Other hyperparameters can be introduced in the current model, such as the sparsity ratio s of W (i.e, the fraction of 0 elements in W), the choice of the sigmoid function f or the burn-in BI as in Ref. [30] which correspond to the number of time step of the input data we want to discard. In the case of models with a large number of hyperparameters, such as the ESN, it is essential to study the sensitivity of the results with respect to these hyperparameters, in order to understand if we may fix their values or if they have to be tuned. If this happens, the computational cost of the validation procedure can be significantly reduced.

4.2.1.2 Sensitivity Analysis

As explained previously, we want to fix as much as possible the hyperparameters required by the ESN approach in order to save some computational time during the ensemble validation procedure that will be detailed in the next section. Some of them can already be fixed for mathematical consideration. Indeed, to satisfy the sufficient condition for the ESP introduced in Section 3.3, we fix the spectral radius $\rho(\tilde{W}) = 0.99$. Furthermore, we fix the sigmoid function $f = \tanh$, the leaking rate $a = 1$,and $\frac{\Delta t}{c} = 0.01$.

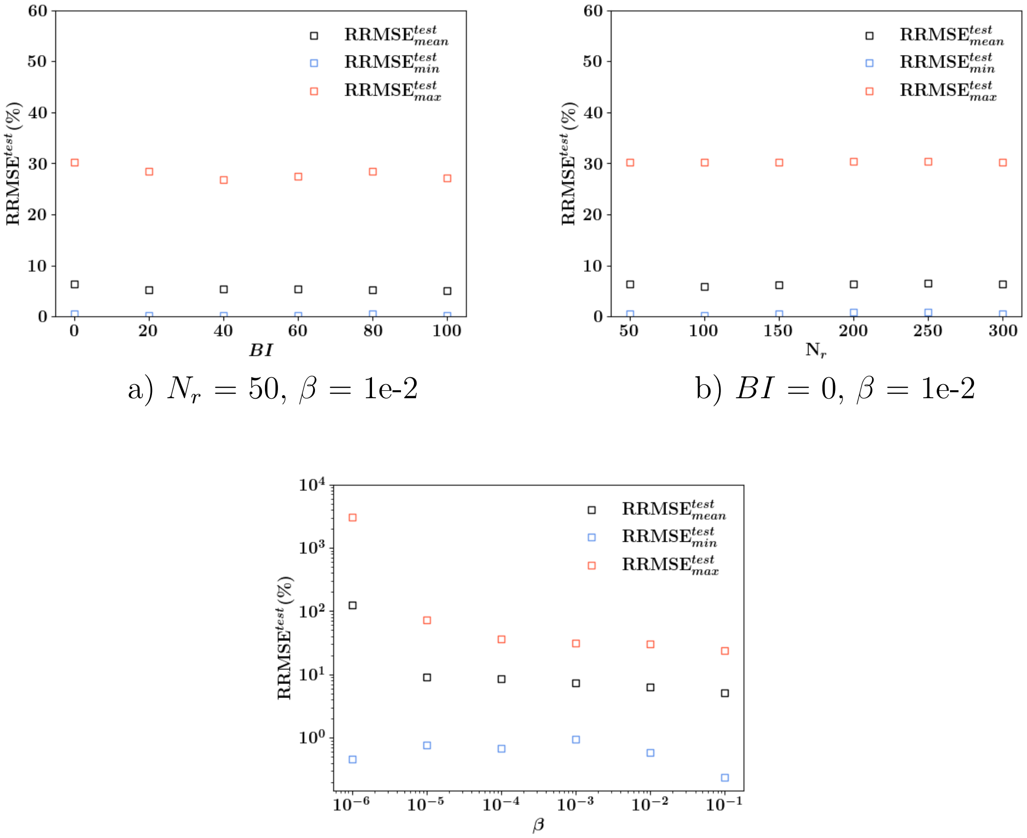
For the other hyperparameters (i.e, reservoir size N_r, burn-in BI, regularization parameter β), we perform a sensitivity analysis to study their effects on the results of the ESN predictions. We compute the mean, minimum and maximum of the Relative Root Mean Square Error (RRMSE) in the *test set* RRMSE^{test} over the 60 seeds of the HL-LHC realistic model and for different values of N_r, BI and β. The RRMSE in % between the output of our approach x^{out} and the target $x^{test\ target}$ in the *test set* is defined as [31]:

$$RRMSE^{test} = 100\sqrt{\frac{\sum_{k=1}^{k_{test}}(x_k^{out} - x_k^{test\ target})^2}{\sum_{k=1}^{k_{test}}(x_k^{test\ target})^2}} \tag{4.1}$$

where k_{test} is the number of test data.

The plots are shown in Figure 4.5. Clearly, we observe that the mean RRMSE$^{test}_{mean}$ of the RRMSE^{test} over the 60 seeds vary the most with respect to the regularization β. Instead, the mean of the RRMSE^{test} remains almost constant with respect to the burn-in BI and reservoir size N_r. In other words, our ESN model is not very sensitive with respect to these hyperparameters, so that we may fix them without a significant impact on the performance of the ESN. Thus, we decide to fix $BI = 0$ and $N_r = 50$ for any input. For such a small reservoir, we set the sparsity ratio $s = 0$, so that all the elements of W are non null.

Remark 4.1 *Sparsity is usually required for large reservoirs ($N_r > 10^3$) to reduce* *the computational cost of the training and simulation phase. In our case, since we use a small reservoir of size $N_r = 50$, there is no strong motivation to use sparse matrices. However, it is important to notice that the proposed approach is implemented so as to deal also with sparse matrices if required.*



a) $N_r = 50$, β = 1e-2

b) $BI = 0$, β = 1e-2

$BI = 0$, $N_r = 50$

Figure 4.5: Mean, Minimum and Maximum of the RRMSE in the *test set* over the 60 seeds of the HL-LHC realistic model for different N_r, BI and β

To sum up, the only hyperparameter that must be tuned for any input is the regularisation parameter β. The tuning is performed using the ensemble validation approach proposed in the next section, inspired by [6] and [29].

In Table 4.1, we summarise the values of all the hyperparameters fixed for any ESN input.

N	s	ρ	α	BI	f	$\frac{\Delta t}{c}$
50	0	0.99	0.01	0	tanh	0.01

Table 4.1: Fixed Hyperparameters

Another important element that we still have not considered is the random generation of the input and internal weight matrices W^{in} and W. This is done by sampling their elements from a uniform pseudo-random distribution on (0,1) and scaling them into the interval (-0.5,0.5) to have negative elements as well. In addition, the spectral radius of $\tilde{W}$ is fixed to 0.99 for all the generations, in order to satisfy the ESP. The generation procedure of W^{in} and W is detailed in Algorithm 1.

Algorithm 1 Generation

```
Input: $\rho$ spectral radius of $\tilde{W}$ (fixed to 0.99), $K$ input size, $N_r$ reservoir size
Output: $W^{in}$ input weight matrix, $W$ reservoir weight matrix
Step 1: Random intialization
$W^{in}_{i,j} \sim \mathcal{U}(0,1) - 0.5$ , $i = 1, ..., N_r$, $j = 1, ..., K$
$W_{i,j} \sim \mathcal{U}(0,1) - 0.5$ , $i, j = 1, ..., N_r$
Step 2: Rescaling
$\rho_{rand}$ = Spectral radius of $\frac{\Delta t}{c}|W| + (1-a\frac{\Delta t}{c})I$
W = W $\rho/\rho_{rand}$
```

Because the generation is random, an ensemble validation approach is then necessary.

4.2.1.3 An Ensemble Validation Approach

We present the validation procedure based on an ensemble approach used to deal with the randomness in the initialisation of the input and reservoir weight matrices W^{in} and W. Algorithm 2 presents the pseudo code of the validation procedure. In the validation approach, we compute the Mean Square Error (MSE) in the *validation set* MSE^{val} of our ESN based predictive approach for N_W different pairs of (W^{in},W) and N_β different regularization parameters β. Then, we compute β^{val} the β which minimizes in average (i.e, over the N_W pairs of (W^{in},W)) the MSE^{val}. Finally, β^{val} is used to evaluate our approach for new data.

The target validation data $x^{val\ target}$ which contain the k_{val} validation data can be obtained either from the actual data, by using the piecewise functions introduced previously, or by fitting the data with the scaling law introduced in Section 2.4. We denote by **ESN** the validation approach using only the piecewise constant input functions built from the original data and by **ESN-SL** the validation approach using the output of the fitted scaling law presented in Eq.(2.17) (**SL**) as validation data. The other scaling law presented in Eq.(2.18) (**SL2**) gave comparable results to the **SL**, so for simplicity of the formula, we decided to use only the **SL**. The comparison of the two scaling laws **SL** and **SL2** is presented in the Appendix A.

Algorithm 2 Validation

Input: β^{vector} vector of regularization parameters β of length N_β, N_W random different pairs of (W^{in},W), $x^{train\ target}$ training target data, $x^{val\ target}$ validation target data, u input data, H set of all the fixed hyperparameters

Output: $\beta^{val} \in \beta^{vector}$ which minimize in average the MSE in the *validation set* MSE^{val}

Step 1: Store the MSE in the validation set in $M \in \mathcal{M}_{N_\beta \times N_W}(\mathbb{R})$

```
for $i = 1$ to $N_\beta$ do
    for $j = 1$ to $N_W$ do
        $W^{out}$ = Training($u$, $x^{train\ target}$, $H$, $(W^{in}, W)^j$, $\beta^i$, where = training set)
        $x^{out}$ = Prediction($H$, $(W^{in}, W)^j$, $W^{out}$, where = validation set)
        $\mathrm{M}_{i,j} = MSE(\mathrm{x}^{out}, x^{val\ target}) = \frac{1}{n}\sum_{i=1}^{k_{val}}(x_i^{out} - x_i^{val\ target})^2$
    end
end
```

where W^{out} is the output weight matrix found by ridge regression and x^{out} is the predicted output of our approach

Step 2: Compute vector of mean m^{mean} **over the rows of** M

$$m_{mean} = [\ m_{mean}^{\beta^1}\ ,\ ...,\ m_{mean}^{\beta^{N_\beta}}]^T$$

where $m_{mean}^{\beta^i}$ is the mean of the $i-th$ row of M.

Step 3: Compute the minimum and select β^{val}

$$\beta^{val} = \underset{\beta^i}{\mathrm{argmin}}\ m_{mean}^{\beta^i}$$

Remark 4.2 *We did not detail the functions* Training() *and* Prediction() *since they use exactly the same equations presented in* Section 3.2

We tried out our validation method with $\beta^{vector} = [1e-4, 2e-4, 3e-4, 1e-3, 2e-3, 3e-3, 1e-2, 2e-2, 3e-2]$ vector of regularization parameters β of length $N_\beta = 9$

4.2.2 Test Procedure

We end this chapter with the description of the testing procedure presented in Algorithm 3, also based on an ensemble method, used to assess the predictive capabilities of our two approaches (i.e, **ESN** and **ESN-SL**).

```
Algorithm 3 Test

Set of Fixed Hyperparameters $H$: $N_r = 50$, $a = 0.01$, $\rho(\tilde{W}) = 0.99$, $s = 0$, $f = tanh$, $BI = 0$

Input: $\beta^{val}$ = Validation output, $N_W = 100$ number of different pairs of $(W^{in}$,$W)$,
$x^{train\ target}$ training target data of size $k_{train}$, $x^{test\ target}$ test target data of size $k_{test}$,
$u$ input data

Output: $x^{out}_{mean}$ mean of the predicted outputs, $RRMSE^{test}$ between $x^{out}_{mean}$ and $x^{test\ target}$

Step 1: Generation of $(W^{in}$,$W)$
$(W^{in}$,$W)^i$ = Generation($\rho(\tilde{W})$, $N_r$), $i = 1, ..., N_W$

Step 2: Predict in the test set and store in $p \in \mathbb{R}^{N_W}$
for $j = 1$ to $N_W$ do
    $W^{out}$ = Training($u$, $x^{train\ target}$, $H$, $(W^{in}, W)^j$, $\beta^{val}$, where = training set)
    $x^{out}$ = Prediction($H$, $(W^{in}, W)^j$, $W^{out}$, where = test set)
    $p_j = x^{out}$
end

Step 3: Average over the predictions $p$
$x^{out}_{mean}$ = mean($p$)

Step 4: RRMSE
RRMSE$^{test}$ = RRMSE($x^{out}_{mean}$, $x^{test\ target}$)
```

5 Results and Discussions

In this last Chapter, we present the DA predictions obtained with the two approaches denoted previously by **ESN** and **ESN-SL**. In particular, we compare these approaches with the fitted scaling law **SL** presented in Eq.(2.17) and used in [14]. We recall that **ESN** denotes the mean prediction x_{mean}^{out} over $N_W = 100$ predictions using piecewise constant functions built from a given dataset as validation data, whereas **ESN-SL** denotes the mean prediction x_{mean}^{out} over $N_W = 100$ predictions using the fitted scaling law **SL** as validation data. The training data of the **ESN** and **ESN-SL** are the same, only the validation data differ. The hyperparameters used, the validation and testing methods are those previously introduced in Section 4.2. We test the proposed approaches both with the HL-LHC and 4D Hénon Map datasets presented in Section 4.1.1

5.1 DA Predictions of the HL-LHC dataset

We start this first Section by presenting the results regarding the abilities of the **ESN-SL**, **ESN** and **SL** to forecast the DA of the HL-LHC realistic model. First, we plot the values of β^{val} found in our validation procedure for all seeds. Then, we show for each approach the best and worst seed predictions. Finally, we plot the predictions, with their RRMSE in the *test set*, over the 60 seeds.

5.1.1 Validation output

Before testing our approaches, it is necessary to apply our validation procedure, described in Algorithm 2, in order to find the β^{val} minimizing the MSE on the *validation set*. Thus, in Figure 5.1, we plot the distribution of the β^{val} found in the validation process. As we can see, due to the different validation datasets, the values of β^{val} found by **ESN** and **ESN-SL** differ slightly. Once the β^{val} value has been computed for each of the 60 seeds, we can test our approaches using Algorithm 3 and plot the predicted DA.

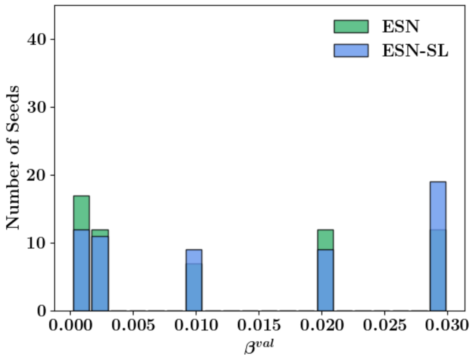
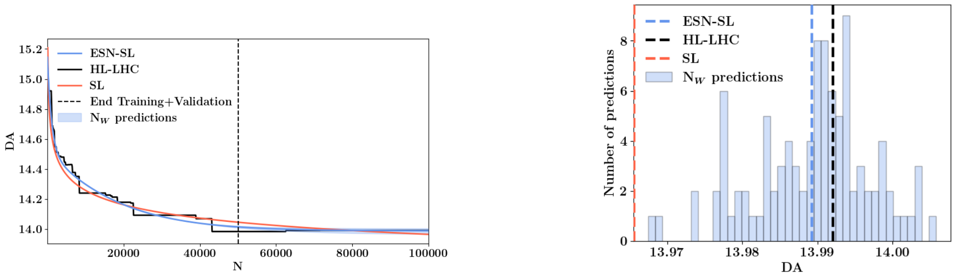


Figure 5.1: Distribution of the β^{val} over the 60 seeds for the **ESN-SL** and **ESN**.

5.1.2 Best and worst seed predictions

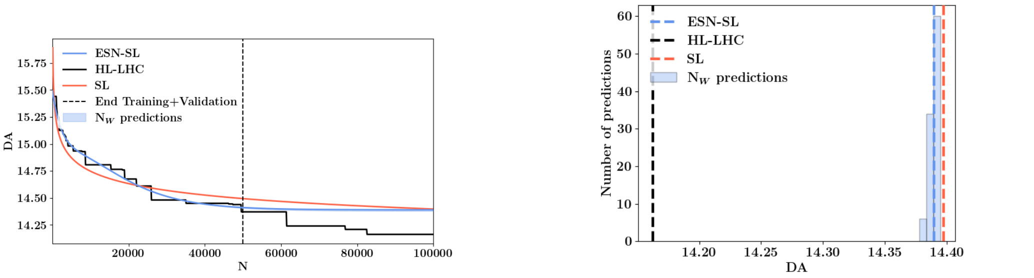
In this section, we focus only on the best and worst seed predictions of the **ESN-SL** and **ESN**. In Figures 5.2-5.4, we plot the x^{out}_{mean} of the best and worst seed predictions of the **ESN-SL** and **ESN** and compare them with the **SL** prediction. We also show the distribution of the $N_W = 100$ predictions p_j associated to the N_W pairs of $(W^{in}, W)^j$, $j = 1, ..., N_W$ at the end of the *test set* (i.e, $N = 10^5$ turns). As we can observe, the best and worst seed predictions differ for the **ESN-SL** and **ESN** approaches. Also, the worst predictions of the **ESN-SL** and **ESN** are better than the **SL** prediction.



a) **ESN-SL** x^{out}_{mean}

b) Empirical distribution of N_W predictions at $N = 10^5$

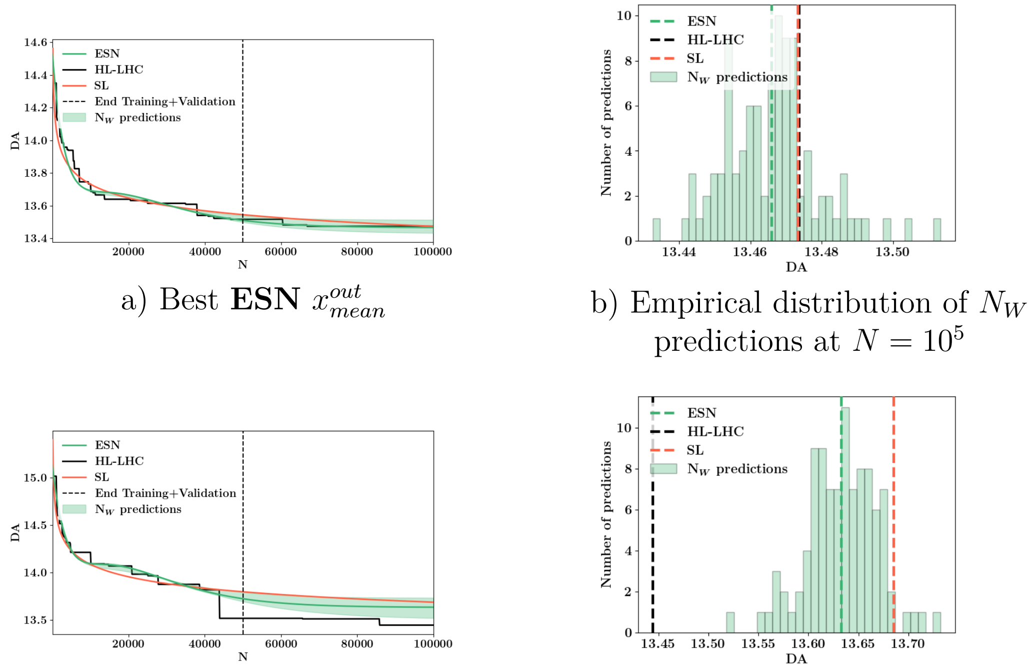
Figure 5.2: x^{out}_{mean} of the **ESN-SL** prediction for the best (53th) seed with the distribution of its $N_W = 100$ predictions at $N = 10^5$ turns and comparison with **SL**.



a) **ESN-SL** x^{out}_{mean}

b) Empirical distribution of N_W predictions at $N = 10^5$

Figure 5.3: x^{out}_{mean} **ESN-SL** prediction of the worst (24th) seed with the distribution of its $N_W = 100$ predictions at $N = 10^5$ turns and comparison with **SL**.



a) Best **ESN** x^{out}_{mean}

b) Empirical distribution of N_W predictions at $N = 10^5$

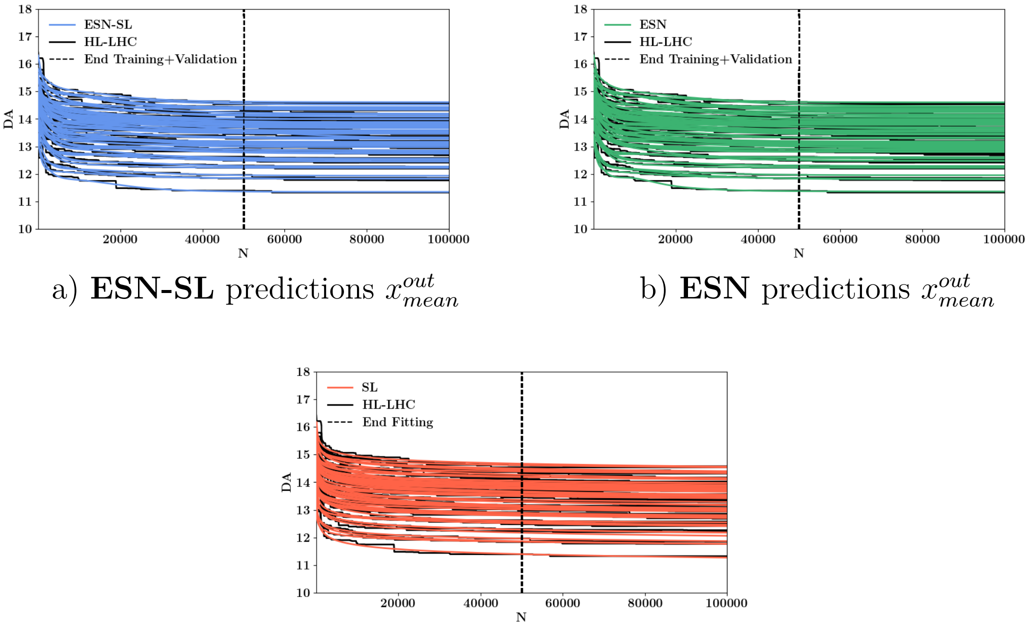
a) Worst **ESN** x^{out}_{mean}

b) Empirical distribution of N_W predictions at $N = 10^5$

Figure 5.4: x^{out}_{mean} of the **ESN** prediction for the best (25th) and worst (35th) seed with the distribution of its $N_W = 100$ predictions at $N = 10^5$ turns and comparison with **SL**.

5.1.3 60 seed predictions

We show in Figure 5.5 the overlapped predictions of the 60 seeds, in order to show that all the predictions fit well in general with the test data.



a) **ESN-SL** predictions x_{mean}^{out}

b) **ESN** predictions x_{mean}^{out}

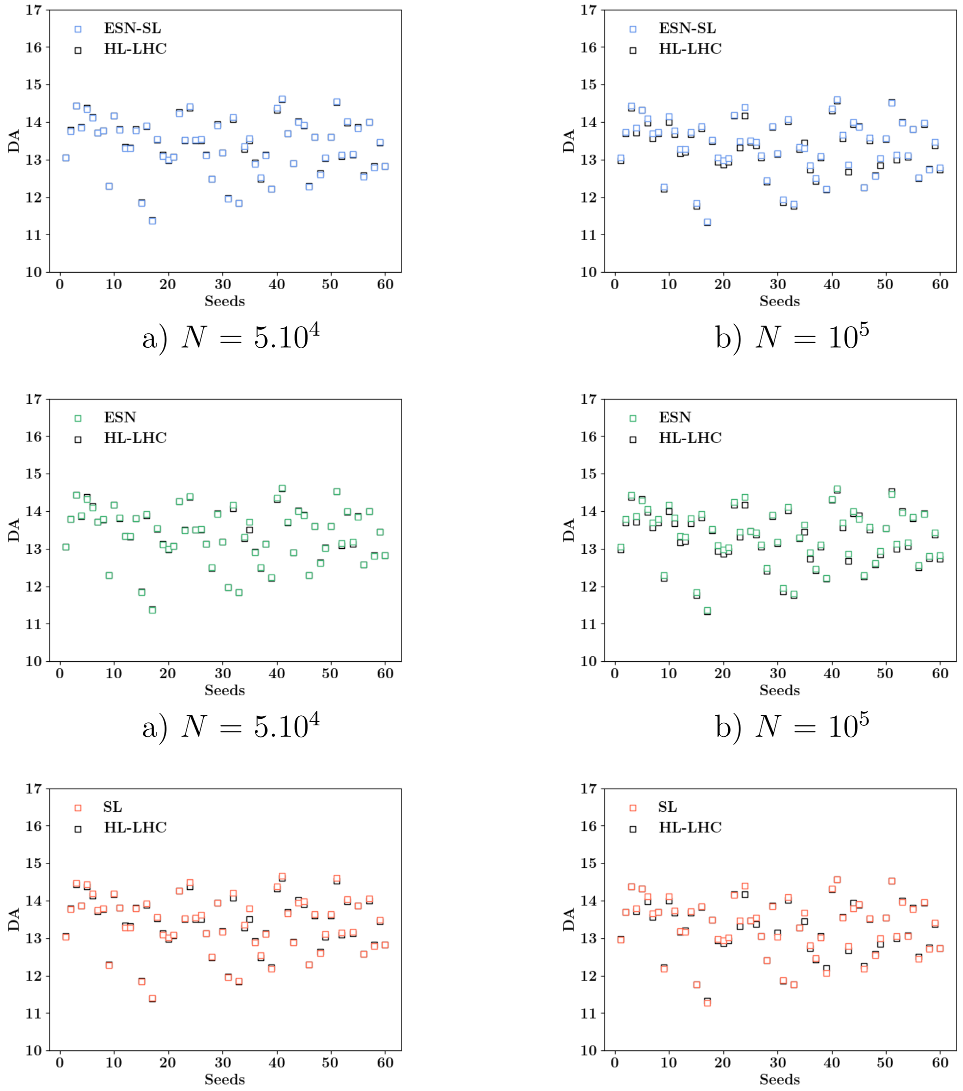
b) **SL** predictions

Figure 5.5: ESN-SL, **ESN** and **SL** predictions for the 60 seeds.

Also, in Figure 5.6, we plot the predicted DA values of our three approaches at the beginning ($N = 5.10^4$ turns) and end ($N = 10^5$ turns) of the *test set* and compare them with the HL-LHC data. We observe that the DA data predicted at $N = 5.10^4$ and $N = 1.10^5$ with the **ESN-SL** fit globally slightly better than the ones obtained with the **ESN** or **SL**.

For a more detailed analysis, we plot the distribution of the RRMSE in the *test set* over the 60 seeds in Figure 5.7. We can clearly see that the maximal RRMSE is lower with the **ESN-SL** and **ESN** approaches than with the **SL**. Also, we notice that the **ESN-SL** approach predict more seeds with lower RRMSE than the **ESN**. In Table 5.1, we summarise the maximum and mean RRMSE over the 60 seeds for the three approaches.

On average, the RRMSE of the **ESN-SL** is 10% better than the **ESN** and 5% better than **SL** with a maximal RRMSE 27% lower than the one of the **SL**. It seems that using validation data generated by the fitted scaling law allows to improve the prediction by finding more relevant β^{val} with respect to the piecewise constant inputs built based on the original data.



a) $N = 5.10^4$

b) $N = 10^5$

a) $N = 5.10^4$

b) $N = 10^5$

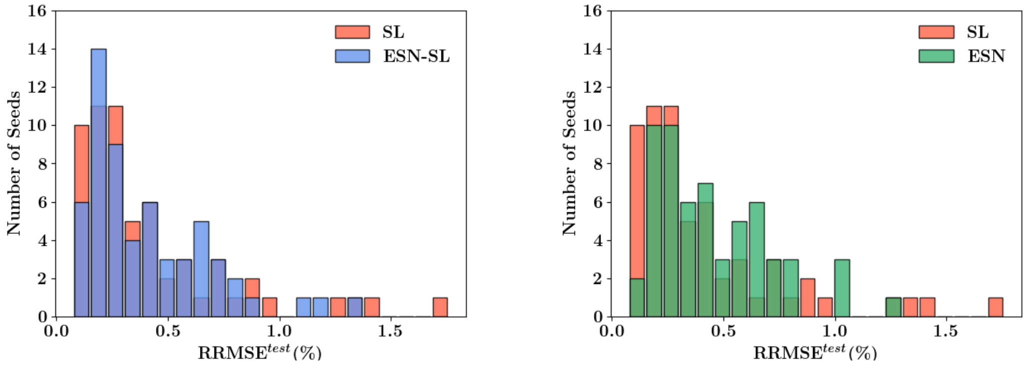
a) $N = 5.10^4$

b) $N = 10^5$

Figure 5.6: Predictions x^{out}_{mean} for the 60 seeds at $N = 5.10^4$ turns and $N = 10^5$ turns for the **ESN-SL**, **ESN** and **SL**.

In Table 5.2, we compare the CPU time required by the three approaches. It is clear that the scaling law is much less time consuming. However, the training time and test time of the **ESN-SL** and **ESN** remain very low and the CPU time required by the ESN approaches is mainly due to the ensemble validation approach, which requires the scan over the $N_W = 100$ different (W^{in}, W) in order to produce

100 differents predictions for each of the 60 seeds. On one hand, validating over a single reservoir realization, as common in earlier literature on RC approaches, could allow to reduce substantially the CPU time required training, validation and test of the **ESN-SL** and **ESN**, while providing a prediction that would be on average still superior to that of the **SL** approach. On the other hand, however, the ensemble validation approaches can be easily parallelized. Furthermore, ensemble approaches provide the complete probability distribution of DA values over the whole ensemble, thus allowing for a more robust quantification of the uncertainties involved in these estimates.



a) Comparison **ESN-SL** and **SL**

b) Comparison **ESN** and **SL**

Figure 5.7: Distribution of the RRMSE in the *test set* RRMSE^{test} over the 60 seeds for the **ESN-SL**, **ESN** and **SL**.

	RRMSE^{test} max (%)	RRMSE^{test} mean (%)
ESN-SL	1.30	0.40
ESN	1.22	0.44
SL	1.76	0.42

Table 5.1: Performance of the **ESN-SL** and **SL** for the prediction of DA over the 60 seeds.

	Training time	Validation time	Test time
ESN-SL	3	55	4
ESN	3	55	4
SL	2.10^{-3}	X	6.10^{-5}

Table 5.2: CPU time (s) of the **ESN-SL**, **ESN-SL** and **SL** approaches

5.2 DA Predictions of the 4D Hénon Map dataset

This Section is dedicated to the presentation of the results regarding the forecasting of the DA for the dataset produced with the 4D Hénon Map. In particular, our aim is to reproduce results similar to those found with the HL-LHC dataset using a longer dataset. As in the previous section, we start by plotting the β^{val} found by our validation procedure for all cases. Then, we show the best and worst case predictions for each approach and we plot the predictions, with their RRMSE in the *test set*, over the 60 cases.

5.2.1 Validation output

The prediction in the *test set* requires the knowledge of the β^{val}. Thus, in Figure 5.8, we plot the distribution of the β^{val} found in the validation process. The values of the β^{val} found by **ESN** and **ESN-SL** differ from each other much more than with the HL-LHC dataset. As a consequence, we can expect much more significant differences between the accuracy of the **ESN** and **ESN-SL** predictions.

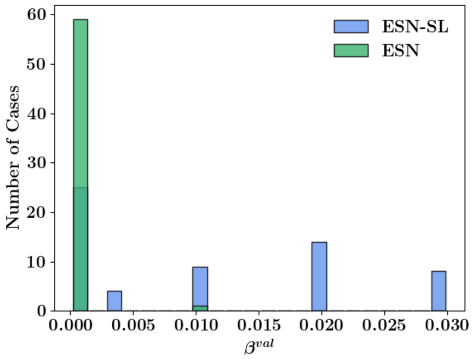
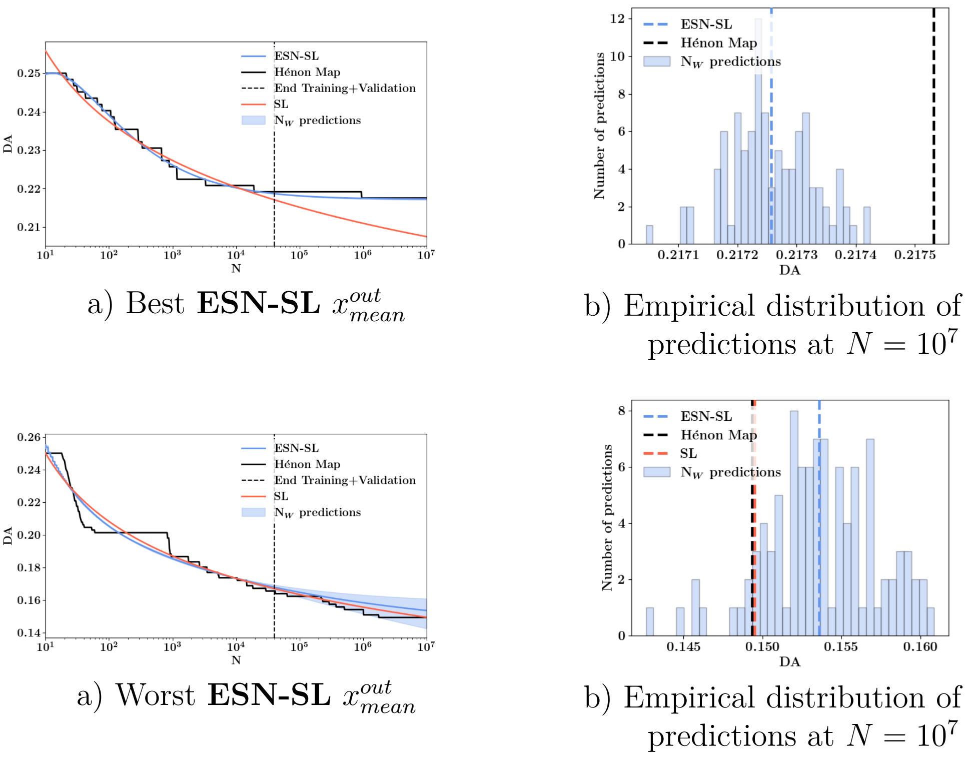


Figure 5.8: Distribution of the β^{val} over the 60 cases for the **ESN-SL** and **ESN**.

5.2.2 Best and worst case predictions

We also start by looking only at the best and worst case predictions of the **ESN-SL** and **ESN**. In Figures 5.9-5.11, we plot the x^{out}_{mean} of the best and worst case predictions of **ESN-SL** and compare with that obtained by the **SL**. For **ESN-SL** and **ESN**, we also show the distribution of the $N_W = 100$ predictions at $N = 10^7$ turns (end of the *test set*). Here, we notice that the best case (41th) prediction is identical for the two approaches. However, the worst case prediction obtained by the **ESN-SL** is much better than the one of the **ESN**.



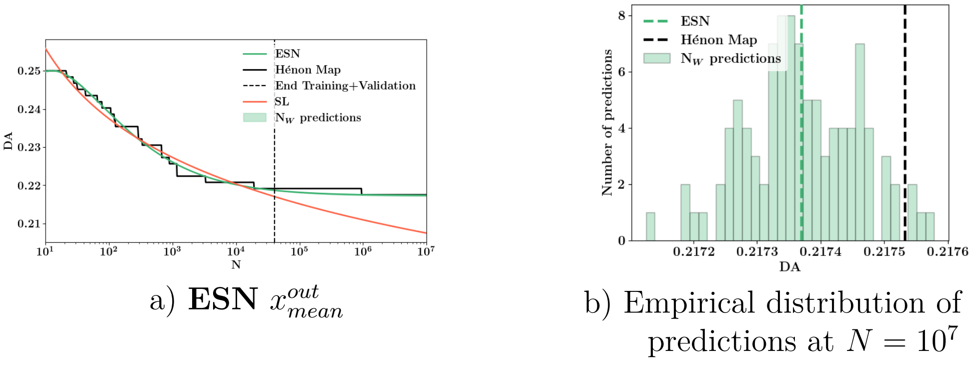
a) Best **ESN-SL** x_{mean}^{out}

b) Empirical distribution of N_W predictions at $N = 10^7$

a) Worst **ESN-SL** x_{mean}^{out}

b) Empirical distribution of N_W predictions at $N = 10^7$

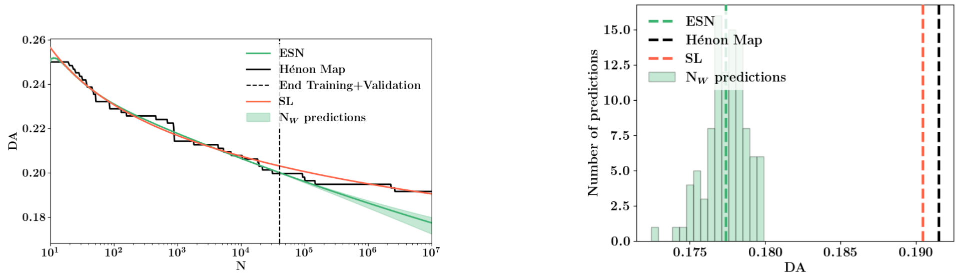
Figure 5.9: x_{mean}^{out} of the best (41th) and worst (31th) case prediction of the **ESN-SL** with the distribution of its $N_W = 100$ predictions at $N = 10^7$ turns and comparison with **SL**.



a) **ESN** x_{mean}^{out}

b) Empirical distribution of N_W predictions at $N = 10^7$

Figure 5.10: x_{mean}^{out} of the **ESN** best (41-th) case prediction with the distribution of its $N_W = 100$ predictions at $N = 10^7$ turns and comparison with **SL**.



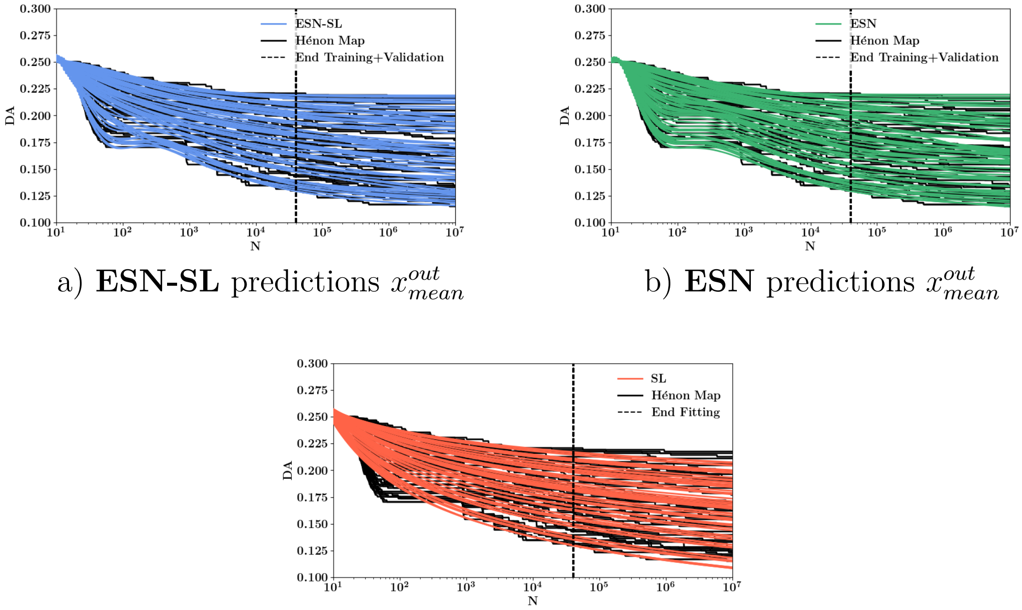
a) **ESN** x^{out}_{mean}

b) Empirical distribution of N_W predictions at $N = 10^5$

Figure 5.11: x^{out}_{mean} of the **ESN** worst (47-th) case prediction with the distribution of its $N_W = 100$ predictions at $N = 10^7$ turns and comparison with **SL**.

5.2.3 60 cases predictions

We plot in Figure 5.12 the predictions obtained by the three approaches for all the 60 cases.



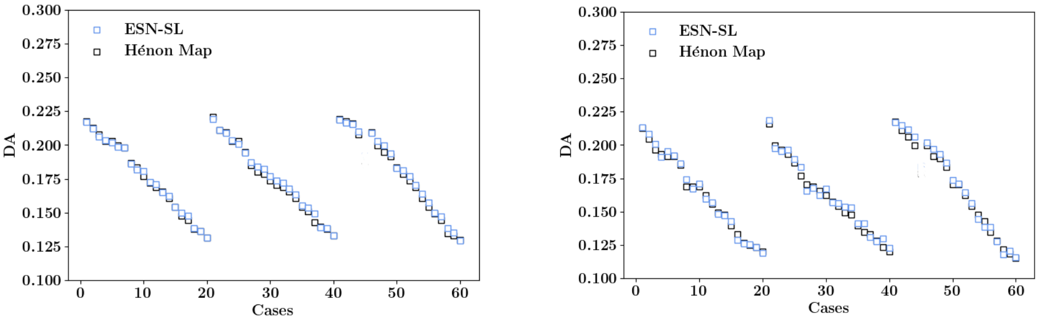
a) **ESN-SL** predictions x^{out}_{mean}

b) **ESN** predictions x^{out}_{mean}

b) **SL** predictions

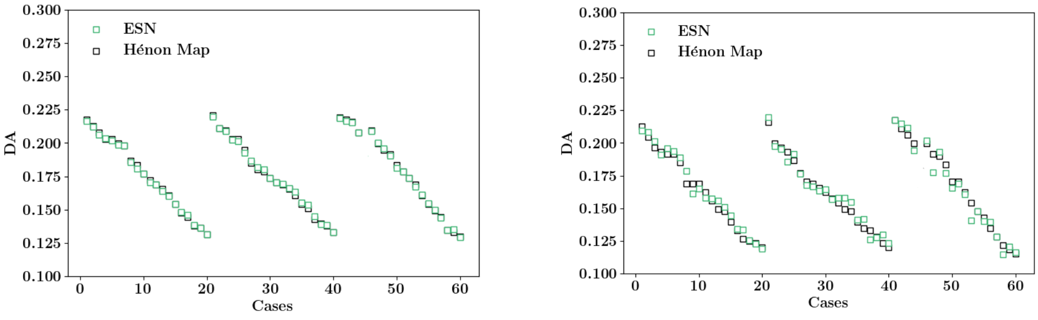
Figure 5.12: Predictions of the 60 cases for the **ESN-SL**, **ESN** and **SL**

Also, in Figures 5.13-5.15, we plot the predicted DA values of our three approaches at the beginning ($N = 5.10^4$ turns) and end ($N = 10^7$ turns) of the *test set* and compare them with the Hénon map data. We observe again that the DA prediction with the **ESN-SL** at $N = 5.10^4$ and $N = 1.10^7$ fit globally better than those obtained with the **ESN** or **SL**.



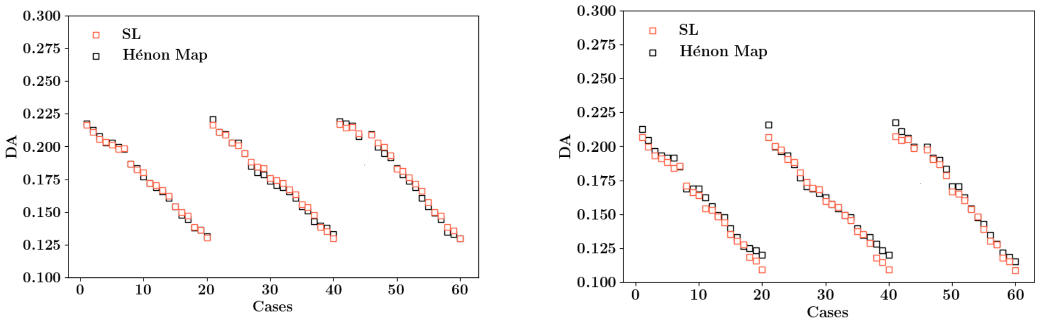
a) $N = 5.10^4$ b) $N = 10^7$

Figure 5.13: Predictions x^{out}_{mean} for the 60 cases at $N = 5.10^4$ turns and $N = 10^7$ turns for the **ESN-SL**.



a) $N = 5.10^4$ b) $N = 10^7$

Figure 5.14: **ESN** predictions x^{out}_{mean} for the 60 cases at $N = 5.10^4$ turns and $N = 10^7$ turns .

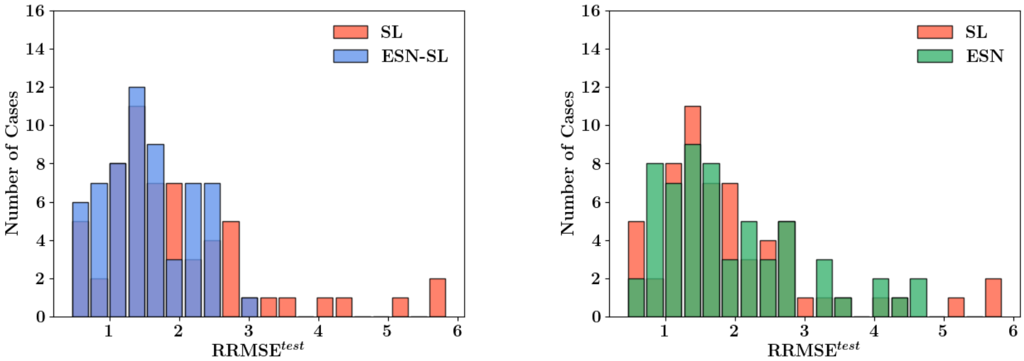


a) $N = 5.10^4$

b) $N = 10^7$

Figure 5.15: **SL** predictions for the 60 cases at $N = 5.10^4$ turns and $N = 10^7$ turns.

To compare more in detail the three approaches, we plot the distribution of the RRMSE in the *test set* over the 60 cases in Figure 5.16. We can clearly see that the RRMSE obtained are larger than in the HL-LHC case for all the three approaches. This is mainly due to the fact that we predict until a larger number of turns. In Table 5.3, we summarise the maximum and mean RRMSE over the 60 cases for the three approaches.



a) Comparison **ESN-SL** and **SL**

b) Comparison **ESN** and **SL**

Figure 5.16: Distribution of the RRMSE in the *test set* RRMSE^{test} over the 60 cases for the **ESN-SL**, **ESN** and **SL**

On average, the RRMSE of the **ESN-SL** prediction is 23% better than the **SL** and 21% better than the **ESN**. Also, its maximal RRMSE is almost 50% lower than the one of the **SL**. In conclusion, the results on the longer Hénon map dataset seem to be coherent with those obtained on the HL-LHC data, since also in this case, the **ESN-SL** performs in average better than the **ESN** and **SL**. In other words, using validation data generated by the fitted scaling law seems to be beneficial for the quality of the prediction.

	RRMSE^{test} max	RRMSE^{test} mean
ESN-SL	3.01	1.53
ESN	4.74	1.93
SL	5.85	1.98

Table 5.3: Performance of the **ESN-SL** and **SL** for the prediction of DA over the 60 cases.

Since each of the 60 cases contain the same number of DA data points to the 60 seeds of the HL-LHC, the CPU time of the three approaches is the same to the one reported in Table 5.2.

6 Conclusion

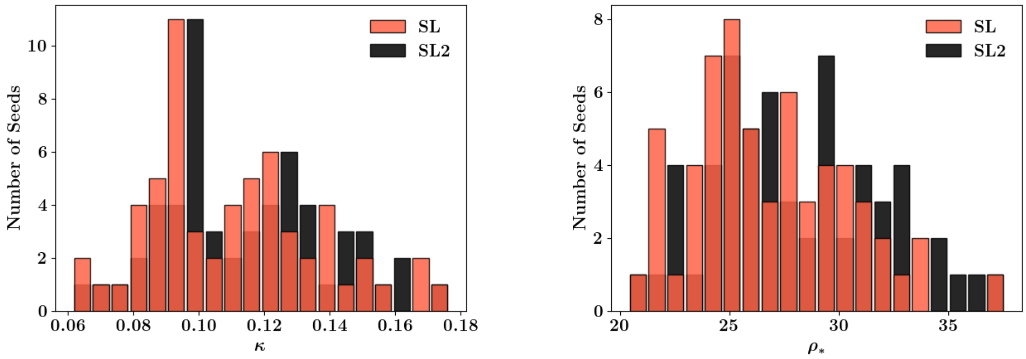
In this thesis, we have developed an ensemble prediction approach for the DA of a circular collider based on ESN. In particular, we have implemented the **ESN-SL** model using the fitted scaling law as validation data. We showed that the model allowed to improve on average the accuracy of the predictions given by the scaling law **SL** both for the HL-LHC realistic and the 4D Hénon Map simplified model. Also, for both applications, the maximal RRMSE of the **ESN-SL** was lower than the one of the **SL** showing a coherent results for the two different datasets. Through a sensitivity analysis and for some mathematical considerations mainly related to the ESP, we showed that most of the hyperparameters of our approach could be fixed so that only the β regularization hyperparameter had to be tuned. The computational cost of the proposed process is much higher than that of the scaling law fitting, mainly because of the validation procedure. However, the ensemble validation approaches can be easily parallelized and ensemble approaches also provide an estimate of the complete probability distribution of DA values over the whole ensemble. This allows for a more robust quantification of the uncertainties involved in these estimates. Eventually, even if the improvement of the predictions given by the **ESN-SL** with respect to the **SL** is not so significant, further works can be developed in order to improve the results. For instance, implementing a multiple reservoirs ESN could help to increase to accuracy of the predictions.

A Scaling Laws Comparisons

This first appendix is dedicated to the comparison of the scaling laws **SL** presented in Eq.(2.17) and **SL2** presented in Eq.(2.18) in their abilities to forecast the DA.

A.1 HL-LHC dataset

We start the comparison the the HL-LHC dataset presented in detail in Chapter 4. In Figure A.1, we compare the distribution of the fitting parameter κ and ρ_* found by least square method using the training and validation data. In other words, the fitting is performed from 10 to 5.10^4 turns.



a) κ

b) ρ_*

Figure A.1: Distribution of the fitting parameter κ and ρ_* over the 60 seeds for **SL** and **SL2**

Eventually, in Figure A.2, we compare the RRMSE in the *test set* over the 60 seeds. The *test set* contains the remaining DA data from 5.10^4 to 10^5 turns. As we can observe, the two scaling laws perform similarly for the extrapolation of DA.

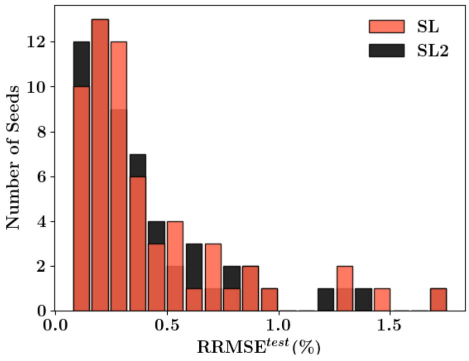


Figure A.2: Distribution of the RRMSE in the *test set* for **SL** and **SL2** over the 60 seeds

A.2 4D Hénon Map dataset

We use now the 4D Hénon Map dataset presented in detail in Chapter 4. In Figure A.3, we compare the distribution of the fitting parameter κ and ρ_* found by least square method using the training and validation data. In other words, the fitting is performed from 10 to 5.10^4 turns.

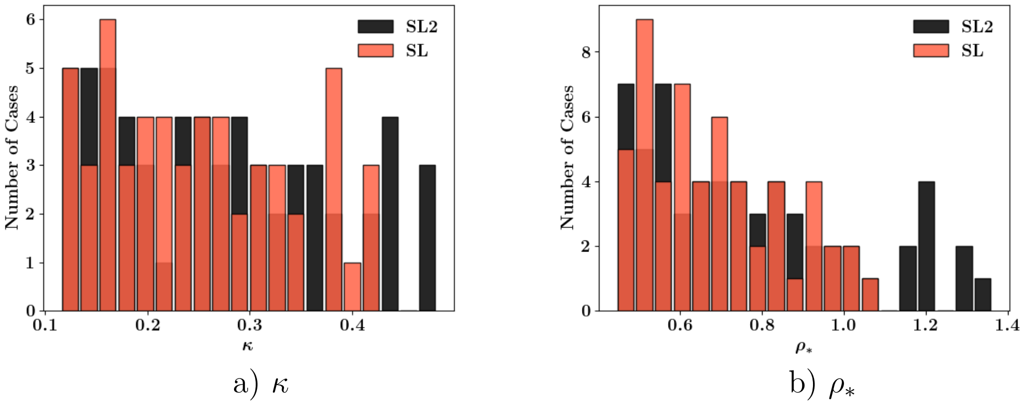


Figure A.3: Distribution of the fitting parameter κ and ρ_* over the 60 cases for **SL** and **SL2**

Eventually, in Figure A.4, we compare the RRMSE in the *test set* over the 60 cases. The *test set* contains the remaining DA data points from 5.10^4 to 10^7 turns. As previously, we find that the two scaling laws perform similarly for the extrapolation of DA.

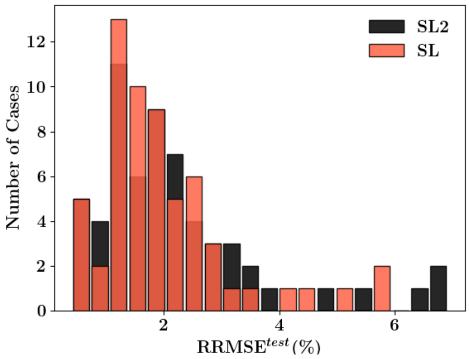
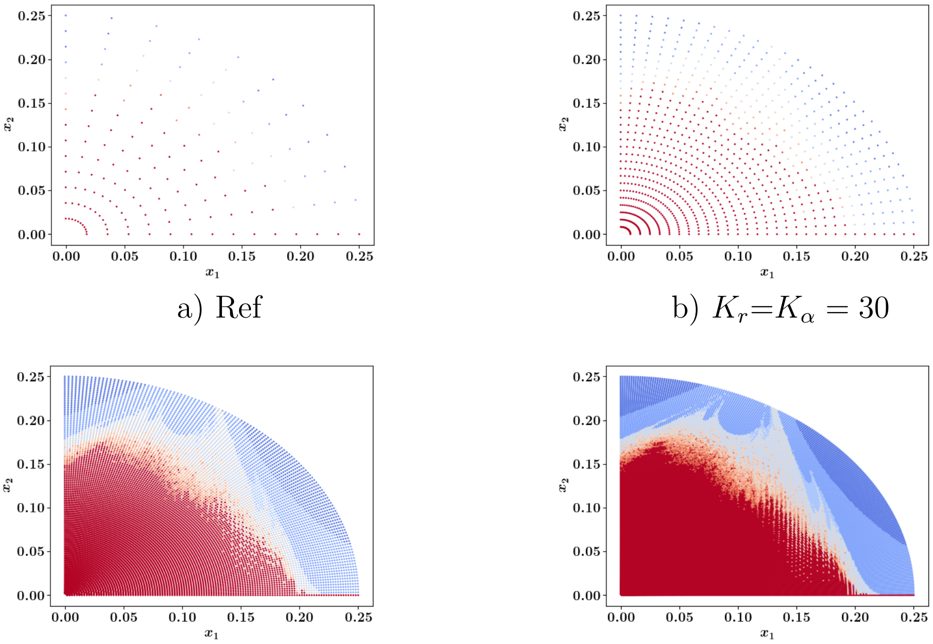


Figure A.4: Distribution of the RRMSE in the *test set* for **SL** and **SL2** over the 60 cases

B Rough test of convergence of the 4D Hénon Map

In this second appendix, we perform a rough test of convergence for one arbitrary case ($\epsilon = 30$, $\mu = 0.2$) of the 4D Hénon Map. In this way, the estimation of the DA is done for different number of angle K_α and radius K_r until N $= 10^7$ turns. We compare the results with our Ref values (i.e, $K_\alpha = 11$ and $K_r = 14$) used in the report. In Figure B.1, we plot the assoaciated stability domains whereas in Figure B.2, the corresponding estimation of the DA.



a) Ref

b) K_r=K_α = 30

b) K_r=K_α = 120

b) K_r=K_α = 210

Figure B.1: Stability domain of the 4D Hénon Map for different number of angle K_α and radius K_r.

As we can observe, the DA, estimated using the Ref values did not fully converge. However, we want to mention that the choice of taking K_α=11 and K_r=14 was only motivated to scan the angles and radii on the same grid to the one of the HL-LHC realistic model.

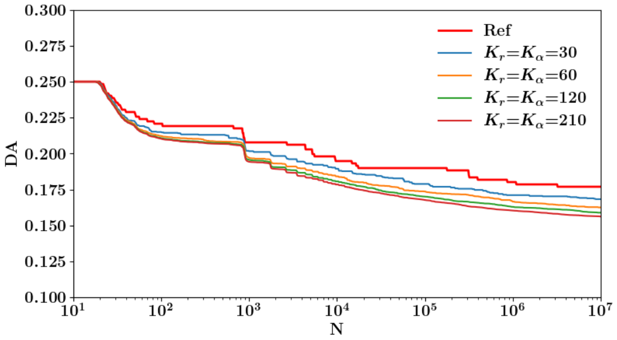


Figure B.2: DA evaluated for different number of angle K_α and radius K_r.

Bibliography

[1] J. K. Koga and T. Tajima. «Particle diffusion from the beam-beam interaction in synchrotron colliders». *Phys. Rev. Lett.* 72 (13 Mar. 1994), pp. 2025–2028.

[2] R. Bruce, R. W. Assmann, and Boccone. «Simulations and measurements of beam loss patterns at the CERN Large Hadron Collider». *Phys. Rev. ST Accel. Beams* 17 (Aug. 2014).

[3] Barbara Dalena. «Performance optimizations at present and future High Energy Colliders» (2021).

[4] F.F. Van der Veken and M. Giovannozzi. «Bridging mathematics and physics: models of the evolution of dynamic aperture in hadron colliders and applications to LHC». *European Physical Society Conference on High Energy Physics* (2019).

[5] Wouter Gevaert, Georgi Tsenov, and Valeri Mladenov. «Neural networks used for speech recognition». *Journal of Automatic Control* 20 (2010).

[6] Stefano Castruccio Huang Huang and Marc G. Genton. «Forecasting High-Frequency Spatio-Temporal Wind Power with Dimensionally Reduced Echo State Networks». *Journal of the Royal Statistical Society* (2019).

[7] Daniel Svozil, Vladimír Kvasnicka, and Jirí Pospichal. «Introduction to multi-layer feed-forward neural networks». *Chemometrics and Intelligent Laboratory Systems* 39.1 (1997), pp. 43–62.

[8] Keiron O'Shea and Ryan Nash. «An Introduction to Convolutional Neural Networks» (2015).

[9] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. «Recurrent Neural Network Regularization». *CoRR* (2014).

[10] Ali Rodan and Peter Tino. «Minimum Complexity Echo State Network». *IEEE Transactions on Neural Networks* 22.1 (2011), pp. 131–144.

[11] Robert Hecht-Nielsen. «III.3 - Theory of the Backpropagation Neural Network Based on "nonindent" by Robert Hecht-Nielsen, which appeared in Proceedings of the International Joint Conference on Neural Networks 1, 593–611, June 1989. © 1989 IEEE.» (1992). Ed. by Harry Wechsler, pp. 65–93.

[12] Lyudmila Grigoryeva and Juan-Pablo Ortega. «Echo state networks are universal». *Neural Networks* 108 (2018), pp. 495–508.

[13] E. Todesco and M. Giovannozzi. «Dynamic aperture estimates and phase-space distorsions in nonlinear betatron motion». *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics* 53 (1996), pp. 4067–4076.

[14] A. Bazzani, M. Giovannozzi, E. H. Maclean, C. E. Montanari, F. 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* 22 (Oct. 2019).

[15] *CERN*. URL: `https://home.cern/fr/science/accelerators/large-hadron-collider`.

[16] F. Ruggiero and F. Zimmermann. «Luminosity optimization near the beam-beam limit by increasing the bunch length or crossing angle» (2002).

[17] Oliver Bruning and Stephane Fartoukh. «BField Quality Specification for the LHC Main Dipole Magnets. Tech» (2001).

[18] Thomas Pugnat. «3D non-linear beam dynamics for the LHC upgrades» (2021).

[19] M. Titze M. Giovannozzi C.E. Montanari. «Dynamic aperture estimates for 4D and 6D non-linear motion and beam intensity evolution models» (2022).

[20] *SixTrack*. URL: `https://github.com/SixTrack/SixTrack`.

[21] W. Scandale M. Giovannozzi and E. Todesco. «Dynamic aperture extrapolation in the presence of tune modulation». *Phys. Rev. E.* 57 (1997).

[22] D.E. Knuth R.L. Graham and O. Patashnik. «Concrete Mathematics» (1994).

[23] Decai Li, Min Han, and Jun Wang. «Chaotic Time Series Prediction Based on a Novel Robust Echo State Network». *IEEE Transactions on Neural Networks and Learning Systems* 23.5 (2012), pp. 787–799.

[24] Boris Hanin. «Which Neural Net Architectures Give Rise to Exploding and Vanishing Gradients?» 31 (2018). Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett.

[25] Herbert Jaeger, Mantas Lukoševičius, Dan Popovici, and Udo Siewert. «Optimization and applications of echo state networks with leaky- integrator neurons». *Neural Networks* 20.3 (2007), pp. 335–352.

[26] Tor A. Johansen. «On Tikhonov regularization, bias and variance in nonlinear system identification». *Automatica* 33.3 (1997), pp. 441–446.

[27] Mantas Lukoševičius and Herbert Jaeger. «Reservoir computing approaches to recurrent neural network training». *Computer Science Review* 3.3 (2009), pp. 127–149.

[28] Izzet B Yildiz, Herbert Jaeger, and Stefan J Kiebel. «Re-visiting the echo state property». *Neural Networks* 35 (Nov. 2012), pp. 1–9.

[29] P.L. McDermott and C.K. Wikle. «An ensemble quadratic echo state network for nonlinear spatio-temporal forecasting». *Stat* (2017).

[30] Yuji Kawai, Jihoon Park, and Minoru Asada. «A small-world topology enhances the echo state property and signal propagation in reservoir computing». *Neural Networks* 112 (2019), pp. 15–23.

[31] Milan Despotovic and Vladimir Nedic. «Evaluation of empirical models for predicting monthly mean horizontal diffuse solar radiation». *Renewable and Sustainable Energy Reviews* 56 (2016), pp. 246–260.