Università degli Studi di Palermo

AREA QUALITÀ, PROGRAMMAZIONE E SUPPORTO STRATEGICO
SETTORE STRATEGIA PER LA RICERCA
U. O. DOTTORATI

Dottorato di Ricerca in Scienze Fisiche e Chimiche

Dipartimento di Fisica e Chimica - Emilio Segrè

Settore Scientifico Disciplinare PHYS-04/A

Noise-Resilient Quantum Extreme Learning Machines for Astrophysical Spectral Analysis

DOTTORE

Marco Vetrano

COORDINATORE

Prof. Marco Cannas

TUTOR

Prof. Massimo Palma

CO-TUTOR

Prof. Salvatore Lorenzo

CICLO XXXVIII

ANNO CONSEGUIMENTO TITOLO 2026

Finanziato dall'Unione europea NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA

To my family,
which have always supported me.
To my fellow office mates,
who made every day worth coming to the office.
To Sarah, my soul mate,
who became part of my life at the beginning of this journey
and always found new ways to make me smile.

ABSTRACT

In this thesis we investigate the information processing capabilities of Quantum Extreme Learning Machines and discuss their real world applications in the field of astrophysics, particularly for atmospheric retrieval of exoplanets and supernovae spectral analysis, designing noise resilient models for Noise Intermediate Scale Quantum (NISQ) implementations.

Extreme Learning Machines (ELMs) offer rapid training and structural efficiency by utilizing a randomized hidden layer—the reservoir—to map input data into a high-dimensional feature space, effectively linearizing complex learning problems. While traditionally implemented via randomized classical neural networks, the mathematical properties of these reservoirs naturally extend to complex physical systems. This universality has driven the development of Quantum Extreme Learning Machines (QELMs), which utilize quantum systems as the computational reservoir. In the field of physical extreme learning machines, information is processed through the internal dynamics of the system and recollected by measuring the system. In the context of QELMs, this consists in selecting a set of observables or a POVM to be measured on the reservoir, whose outcomes is then used to train the post-processing layer.

In this work, we will first review the foundations of Neural Networs, ELMs and Reservoir Computers, highlighting specifically on similarities and fundamental differences between these models and develop the theory behind the quantum counterparts. Then, we demonstrate that QELMs can efficiently process and estimate quantum states well beyond the reservoir's scrambling time, beyond which the information should be irretrievable through local measurements. Furthermore, we show that within this regime, specific classes of Hamiltonians

achieve reconstruction accuracies matching those of global random unitary dynamics. Following these theoretical aspects, we design noise-resilient QELM architectures tailored for the spectral analysis of complex astrophysical phenomena. We apply these models to exoplanetary atmospheric retrieval and supernova remnant datasets, particularly deploying the algorithm for the exoplanetary context on real quantum hardware (IBM Fez). By evaluating various pre-processing and encoding strategies—including Dense Angle Encoding, IQP embedding, and Data Re-uploading—we demonstrate that these noise-resilient QELMs are highly adaptable to different classes of astrophysical objects and capable of extracting critical spectral parameters also on NISQ devices, paving the way for real world applications of Quantum Computing in the astrophysical field.

Contents

Introduction

Recent advances in artificial intelligence have led to the an exponential growth of interest in **Machine Learning** (ML) [1, 2] from both a scientific and an industrial standpoint. Such growth is driven by the its versatility and its accuracy in modeling patterns within complex and non-linear phenomena.

Of particular importance is **Deep Learning** (DL), a sub-field of ML based on **Neural Networks** (NNs) [3–8]. NNs are a class of - predominantly - supervised learning models that employ complex non-linear systems to approximate a target function based on a set of examples. The learning process involves adjusting the network's internal parameters via an optimization procedure [9, 10] . As will be demonstrated in the first part of this thesis, NNs are characterized by high generalizability and have subsequently found a wide range of applications.

As task complexity increases, so does the need for more complex architectures with a higher number of trainable parameters. This growth in degrees of freedom can become computationally expensive in terms of both memory requirements and processing time.

To address these scalability challenges, this work focuses on a specific class of models: **Reservoir Computing** (RC) and **Extreme Learning Machines** (ELM) [11–18]. While these terms are occasionally used interchangeably in the

literature, they represent distinct but related frameworks. These models share the structural properties of classical NNs but offer a specific and distinct feature: they require training only for the final readout layer, while the internal hidden layers (or "reservoir") stay fixed and random.

This approach significantly speeds up the training process compared to fully trainable NNs, despite a potential trade-off in accuracy for certain tasks. The architecture of these models allows the "reservoir" to be realized not only via software-simulated networks but also through physical substrates acting as processing units. [19–26]

The main purpose of this thesis is to investigate the features of a specific class of physical implementations for these models: **Quantum Reservoir Computing** (QRC) and **Quantum Extreme Learning Machines** (QELM) [27–54]. These paradigms employ the high-dimensional internal dynamics of a quantum system as a processing unit. As we shall observe, utilizing a quantum substrate in this context introduces both novel computational perspectives and distinct physical challenges.

This thesis is organized as follows: chapter 1 reviews the theoretical foundations of NNs, contrasting them with the RC and ELM frameworks; chapter 2 defines quantum channels, POVMs and different encoding strategies for the implementation of QELMs. This grants the foundations for the second part, which focuses on the fundamental properties of quantum implementations. Specifically, in chapter 3 and chapter 4 we analyze the performance of QELMs through the lens of **Quantum Information Scrambling** (QIS) [55–65]. In the third and final part, we address the practical utility of these architectures, proposing in chapter 5 a **noise-resilient QELM framework** designed to infer physical and chemical parameters from synthetic transmission spectra of exoplanets and for supernovae

explosion spectral analysis [66–80]. This section moves beyond mere simulation to true experimental validation, reporting the results obtained by deploying our model on actual quantum hardware, demonstrating its efficacy in a real-world quantum processing environment. Finally, in chapter 6, we further develop this model, applying it to a spectral dataset of supernova remnants and improving it with different pre-processing and encoding strategies.

Part I

Theoretical Background

Chapter 1

Neural Networks and Reservoir Computing

Neural Networks (NNs) [3–5] represent the cornerstone of modern Artificial Intelligence, demonstrating unprecedented capabilities in tasks ranging from computer vision to natural language processing [6–8]. However, the predictive power of these models often comes at a significant computational cost. To understand the motivation behind Extreme Learning Machines (ELM) [11,12] and Reservoir Computing (RC) [13–18], we must first examine the standard training protocols for classical networks and their inherent limitations, particularly when processing temporal data.

1.1 Fundamentals on Neural Networks

1.1.1 The Structure

Certain computational tasks involve state spaces of such a high dimensionality that explicitly designing algorithms to account for every variable is analytically impossible. Artificial Neural Networks (ANNs) were conceived to address this challenge, providing a framework to learn algorithms from data samples and generalize to previously unseen inputs. A NN is designed as a **universal function approximator** [81–84], presenting a complex, architecture governed by a set of internal parameters θ (weights and biases), which must be optimized to model accurately a determined target.

Formally, given a dataset $\mathcal{D} = \{(\mathbf{x}_k, \mathbf{y}_k)\}$ where input vectors $\mathbf{x}_k \in \mathbb{R}^n$ and targets $\mathbf{y}_k \in \mathbb{R}^m$, and assuming the existence of a target function $f^* : \mathbb{R}^n \to \mathbb{R}^m$ such that $\mathbf{y}_k = f^*(\mathbf{x}_k)$, the goal of a NN is to approximate this relationship via a parametric function $\mathbf{y}_k \approx f(\mathbf{x}_k; \theta)$ [2, 5].

The building units of this architecture is the neuron (or perceptron). A neuron computes the application of a non-linear activation function, $\sigma(\cdot)$ on the inputs. The non-linearity is a crucial element for the augmentation of variance and distinguishability of the input data. Activation functions all present step-like properties or abruptly varying gradients, such as the widely employed Rectified Linear Unity (ReLU) [2, 3, 5, 85]:

$$\sigma(z) = \max(0, z) = \begin{cases} z & \text{if } z > 0, \\ 0 & \text{if } z \leq 0. \end{cases} \tag{1.1}$$

Nonetheless, while ReLU is very often a standard choice, it not always convenient since it can lead to training issues such as dead neurons.

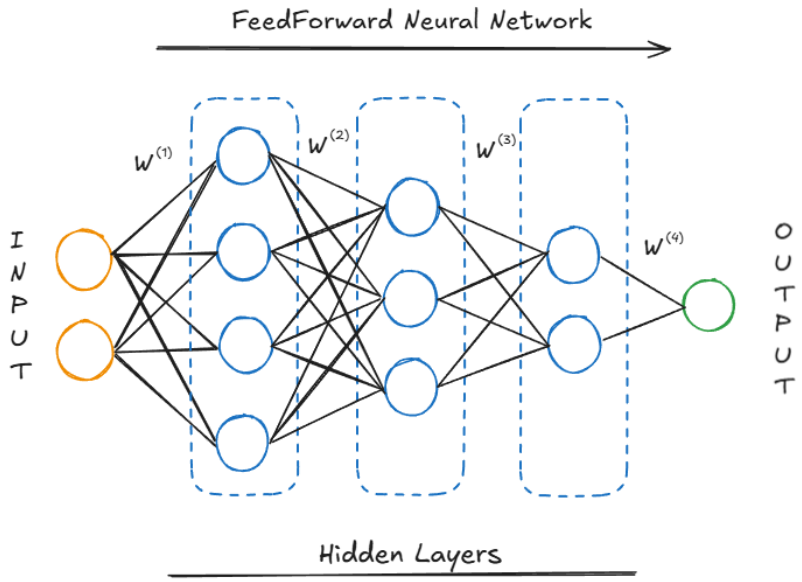


Figure 1.1: **Structure of a FeedForward Neural Network** - The information about the input is transmitted unidirectionally through the network with the output of each layer being linearly transformed using weights and biases $W^{(l)}$ before entering the next set of neurons. The weight matrices are fine-tuned during the training phase through optimization algorithms.

Neurons are organized into interconnected layers. The action of each layer consists in projecting input data into new feature spaces through linear transformations, followed by the application of the non-linear activation functions. The network learns to rotate and scale input features (via weights $\mathbf{W}$ and biases $\mathbf{b}$) to find a reference frame where the relevant features exhibit maximum variance or separability, while regulating neuron's activations. Mathematically, the neurons of the first layer transform the input $\mathbf{x}$ as:

$$\mathbf{h}^{(1)} = \sigma(\mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)}). \tag{1.2}$$

The output $\mathbf{h}^{(1)}$ is then propagated to subsequent layers - as shown in fig. 1.1 - applying similar transformations until the final output is generated.

1.1.2 Recurrent Neural Networks

Up to this point, our discussion has focused on **Feedforward Neural Networks** (FNNs), a class of **memoryless** NNs where information propagates unidirectionally from input to output. These models are generally employed for static tasks, such as image classification, where the output depends only on the current input $\mathbf{x}_k$ and not on the history of previous inputs.

However, many real-world applications—such as time-series forecasting, speech recognition, and natural language processing—involve temporal dependencies on past inputs. These tasks are generally addressed through a different subclass of models known as **Recurrent Neural Networks**. [86, 87]

To provide the network with the capacity of processing time sequences, the model must be supplied with memory. This implies that the internal state of

the network at any given time t must depend on its state at previous time steps. To achieve this, the standard layer equation is modified to include a recurrent connection:

$$\mathbf{h}_t^{(l)} = \sigma(\mathbf{W}^{(l)}\mathbf{x}_t + \mathbf{W}_{rec}^{(l)}\mathbf{h}_{t-1} + \mathbf{b}^{(l)}) \tag{1.3}$$

Here, l denotes the layer index, and $\mathbf{x}_t$ is the input at time t. The addition of the

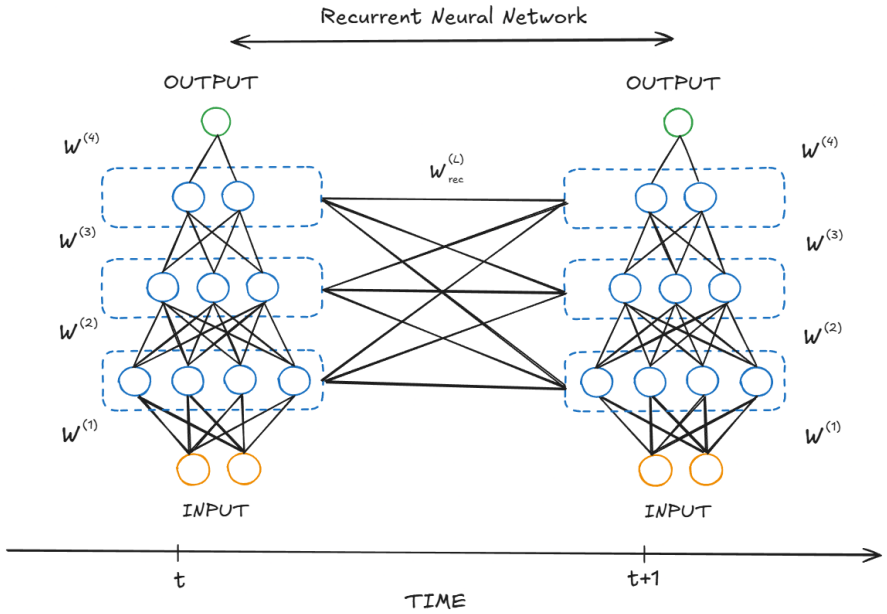


Figure 1.2: **Structure of a Recurrent Neural Network** - The information about the input sequence propagates through the network which maintains memory of previous inputs through the weight matrices W_{rec}^{l}.

term $\mathbf{W}_{rec}^{(l)}\mathbf{h}_{t-1}^{(l)}$ provides the network with an internal feedback loop, which relates its internal state with past inputs, allowing it to frame the temporal context of the sequence. In fig. 1.2 we present a scheme of the model.

1.1.3 The Training Phase

The representational capacity of a NN is determined by its architecture, with the **complexity** of the target function dictating the requirements for the correct functioning of the model. This architecture, in turn, defines the total number of learnable parameters (weights and biases, denoted as θ) which must be optimized during the **training phase**.

The training process is an iterative process consisting of a **forward propagation** followed by **backpropagation**. At the beginning, the network parameters θ are set to a random initial condition [5]. The user defines a **loss function** $\mathcal{L}(\theta)$ (e.g. Mean Squared Error (MSE)) to quantify the discrepancy between the network's prediction and the ground truth.

During the forward pass, the ansatz model is used to generate predictions accordingly to the training input data, while computing the value of the loss function. In the backward pass, the internal parameters are modified to minimize the loss function. The parameters are updated using **optimization routines** [9,10,88,89], which employ different strategies to reach the global minimum of the loss function.

The most fundamental approach is Stochastic Gradient Descent [89] (SGD), defined by the update rule:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L}(\theta_t), \tag{1.4}$$

where η is the **learning rate**, a hyperparameter controlling the step size and $\nabla_\theta \mathcal{L}(\theta_t)$ represents the gradient of the loss function with respect to the internal parameters of the network.

1.1.4 Optimization Challenges

With a high number of parameters, the convergence of algorithms like SGD can be slow, due to the insurgence of local minima, saddle point which act as bottlenecks for the algorithm. To mitigate this effect, modern optimization routines incorporate **momentum** terms, which utilize the history of past gradients to speed up convergence, e.g. **Nesterov Accelerated Gradient** and **Adaptive Moment Estimation optimizer** (Adam) [9, 10].

A much more complex problem is that of vanishing or diverging gradients [90, 91], which is a prevalent issue in deep NNs and even more critical in RNNs. Indeed, depending on the spectral properties of the network structure, even small changes in the parameters can lead to vanishing gradients, causing learning to stop prematurely, or explode, leading to numerical instability and non-converging solutions. This phenomenon is particularly present in RNNs due to the internal feedback connections, which cause the gradient calculation to depend on a long history of time steps, effectively deepening the computational graph and compounding the instability.

1.2 Extreme Learning Machines and Reservoir Computing

The computational challenges associated with training deep and recurrent networks motivated the development of alternative training paradigms. The core insight driving these approaches is that full training of the network parameters is not strictly necessary to achieve universal approximation. By fixing the internal structure of the network to a random initialization and training only the final

readout layer, one can achieve comparable performance with orders of magnitude faster training times. This paradigm is key in two architectures: Extreme Learning Machines for static data, and Reservoir Computing for time sequences.

1.2.1 Extreme Learning Machines

An Extreme Learning Machine is a simple supervised learning model with a structure similar to that of classical FNN but different for its specific training methodology. Unlike traditional networks where all layers are tunable, an ELM comprises a single hidden layer—referred to in this context as **reservoir**—whose internal weights and biases are initialized randomly and fixed. This reservoir acts as a non-linear map $G : \mathbb{R}^n \to \mathbb{R}^{N_{out}}$, projecting the input data into a high-dimensional feature space. The fundamental intuition is that by projecting input vectors into a sufficiently large space via random non-linear transformations, the data becomes linearly separable, significantly simplifying the learning process [11, 12]. Mathematically, the ELM function can be expressed as the composition of the reservoir map and a linear readout layer:

$$f_{model}(\mathbf{x}) = \mathbf{W}_{out} \cdot G(\mathbf{x}), \tag{1.5}$$

where $\mathbf{W}_{out} : \mathbb{R}^{N_{out}} \to \mathbb{R}^m$ represents the trainable weights of the output layer. A scheme of the model is presented in fig. 1.3.

Because the reservoir parameters are fixed, the training process reduces to a linear least-squares problem. Given a training dataset $\{\mathbf{x}_k, \mathbf{y}_k\}_{k=1}^{N_{train}}$, we define the matrix of reservoir activations $G(\mathbf{x}) = \mathbf{R} \in \mathbb{R}^{N_{out} \times N_{train}}$ and the target matrix $\mathbf{Y} \in \mathbb{R}^{m \times N_{train}}$. The optimal readout weights $\mathbf{W}_{out}$ are obtained by minimizing

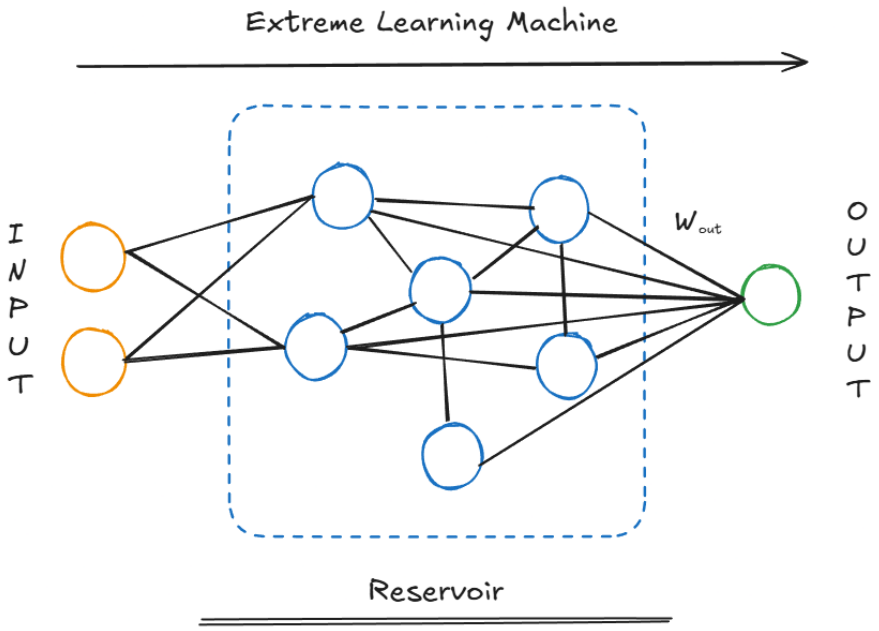


Figure 1.3: **Structure of an Extreme Learning Machine** - The information of the input pass through the reservoir unidirectionally. In this context all the neurons are randomly connected and the hidden layer (reservoir) is unique. The output is post-processed using a weight matrix W_{out} obtained through a Moore-Penrose pseudo-inverse.

the squared error

$$\|\mathbf{W}_{out}\mathbf{R} - \mathbf{Y}\|^2, \tag{1.6}$$

which yields the analytical solution:

$$\mathbf{W}_{out} = \mathbf{Y}\mathbf{R}^{+} \tag{1.7}$$

so that

$$y_k = W_{out} r_k \qquad \forall y_k, r_k \in \mathbf{Y}, \mathbf{R}. \tag{1.8}$$

Here, $\mathbf{R}^{+}$ denotes the Moore-Penrose pseudo-inverse of the reservoir output matrix $\mathbf{R}$. This computation relies on the Singular Value Decomposition (SVD)

$\mathbf{R} = \mathbf{U}\Sigma\mathbf{V}^\dagger$, where U, V are two isometries and Σ is a square positive matrix with the singular values of $\mathbf{R}$ in its diagonal. Then it is possible to compute $\mathbf{R}^+ = \mathbf{V}\Sigma^+\mathbf{U}^\dagger$ by inverting the previous equation [92]. Computing $\mathbf{R}^+$ is essential because $\mathbf{R}$ is generally non-square (since $N_{out} \neq N_{train}$); nonetheless, in the case in which $N_{out} = N_{train}$, then $\mathbf{R}^+ \equiv \mathbf{R}^{-1}$.

Despite the simplicity of the model, the solution to eq. (1.7) is not always numerically stable. This numerical stability is generally quantified through the condition number of the output matrix of the reservoir [93]. Given the reservoir matrix $\mathbf{R}$, the condition number is defined as the ratio:

$$\kappa(\mathbf{R}) = \frac{\lambda_{max}(\mathbf{R})}{\lambda_{min}(\mathbf{R})} \tag{1.9}$$

where λ_{max} and λ_{min} represent the largest and smallest singular values of $\mathbf{R}$, respectively. A high condition number ($\kappa \gg 1$) indicates an ill-conditioned linear system, meaning the solution is highly sensitive to small perturbations or noise. To mitigate this, a regularization term is often added (Tikhonov Regularization) [94], modifying the solution to $\mathbf{W}_{out} = \mathbf{Y}\mathbf{R}^T(\mathbf{R}\mathbf{R}^T + \alpha\mathbf{I})^{-1}$, where α is a regularization parameter.

ELMs are designed for static tasks where each input vector is independent of the others. In this framework, the output depends solely on the instantaneous input, with no dependence on the history of previous interactions. Notably, a memory provided reservoir can still be used for static tasks by resetting its state to a standard initial condition, effectively erasing any memory and ensuring the independence of consecutive inputs.

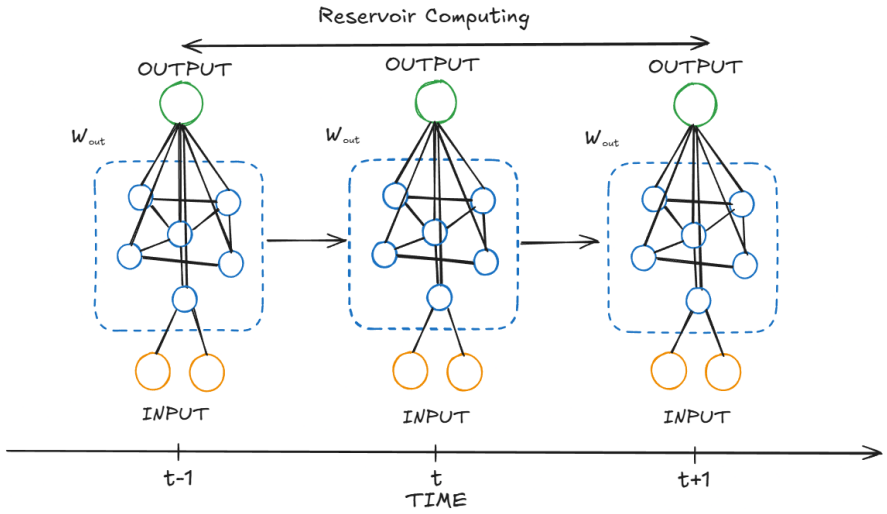


Figure 1.4: **Structure of a Reservoir Computer** - As in the case of ELMs, reservoir computing models present a random internal structure and a unique hidden layer referred to as reservoir. In this context the reservoir retains memory of each input through randomly initialized internal connections with past states (Echo State Network), or by simply not re-initializing it if the reservoir is physical.

1.3 Reservoir Computing

While Reservoir Computing and ELM share the random structure, they differ for its internal recurrence, which provides the model with memory and therefore with the capacity to process time-dependent tasks. Unlike the static structure of an ELM, as shown in fig. 1.4, a Reservoir Computer is a dynamical system driven by an external input signal, whose state depends also on previous inputs in order to solve temporal tasks, in analogy with RNNs. [13–18] To work effectively as a computational platform, a RC model must satisfy three critical properties:

- **The Echo State Property (ESP)** [95]: the state of the reservoir should be uniquely determined by the history of the input, asymptotically forgetting its initial conditions. Formally, consider a reservoir whose state $\mathbf{r}(t)$ evolves

according to a differential equation driven by an input $\mathbf{x}(t)$:

$$\dot{\mathbf{r}} = h(\mathbf{r}, \mathbf{x}). \tag{1.10}$$

The system exhibits the ESP if, for any input sequence and distinct initial conditions $\mathbf{r}(0)$ and $\mathbf{r}'(0)$, the trajectories converge asymptotically:

$$\lim_{t\to\infty} |\mathbf{r}(t) - \mathbf{r}'(t)| = 0. \tag{1.11}$$

This implies that after a sufficient "washout" period, the reservoir state forgets its initial condition, synchronizing with the input signal.

- **Fading Memory** [96]: Closely related to the ESP, fading memory ensures that the influence of past inputs on the current state decays over time. While the reservoir must retain information to process temporal dependencies, this memory must not be infinite; the state $\mathbf{r}(t)$ should depend primarily on the recent history of inputs rather than the distant past. This guarantees stability and prevents the system from becoming chaotic or from divergency. The fading memory property can be assessed by solving some benchmark tasks such as the Short Term Memory task (STM) [97, 98], which quantifies the linear memory, or other task quantifying the non-linear memory such as the Parity Check (PC) [99] and the Nonlinear Auto-Regressive Moving Average (NARMA) task [87].

- **Separability (Approximation Property)** [100]: The reservoir dynamics must map distinct input sequences into distinct state trajectories. If two different input sequences $\{\mathbf{x}_0, \ldots, \mathbf{x}_t\}$ and $\{\mathbf{x}'_0, \ldots, \mathbf{x}'_t\}$ drive the system to the exact same state $\mathbf{r}(t)$, the readout layer will be unable to distinguish them. Thus, the reservoir must act as a high-dimensional, non-linear projection

that expands the separability of the input classes in the state space.

From a mathematical perspective, while eq. (1.8) holds for ELMs, the mapping applied by the reservoir is no longer $G(x_k) = r_k$ but its state depends also on the previous history of inputs, thus given a training dataset $\{\mathbf{x}_k, \mathbf{y}_k\}_{k=1}^{N_{train}}$, the function applied by the RC model is

$$f_{model} = W_{out} G(x_k, r_{k-1}) = W_{out} r_k \tag{1.12}$$

and in the training phase we aim to find the W_{out} which minimizes eq. (1.6).

1.4 Information Processing Substrates

The versatility of the ELM and RC frameworks lies in their mathematical generality. While historically ELM and RC models have been implemented using random NNs (**Echo State Networks**), the Universal Approximation Theorem [82] guarantees that a wide class of non-linear functions $G(\mathbf{x})$ can serve as the basis for approximating any continuous target function, provided the dimensionality of the feature space is sufficiently large. This theorem does not imply that $G(\mathbf{x})$ should necessarily be implemented through random neural networks, but rather require the reservoir to generate a sufficiently high dimensional and non-linear projection of the input. This suggests a paradigm shift, in which the reservoir is not software simulated, but it is instead a physical substrate exhibiting a rich dynamical behavior. Therefore, the field has expanded to include Physical Reservoir Computing, utilizing substrates such as coupled non-linear electronic oscillators, photonic circuits, and soft robotic materials [19–26].

In the last decade, this paradigm has gained significant attention within the quantum physics community. Quantum systems are naturally suited for this role due to their generally complex dynamics and the scaling of the dimension of Hilbert spaces.

The implementation of ELMs employing quantum physical systems as information processing units is at the basis of Quantum Extreme Learning Machines (QELM) and Quantum Reservoir Computing (QRC) [27–54], which will be the central focus of this thesis.

Chapter 2

Quantum Extreme Learning Machines and Quantum Reservoir Computing

In this chapter, we extend the paradigms of Extreme Learning Machines and Reservoir Computing into the quantum domain. This goes beyond a mere change of hardware; it represents a fundamental shift in the computational resources available for information processing. By replacing classical non-linear nodes with the dynamics of a quantum system, we gain access to an exponentially large Hilbert space, offering a potentially vast feature space for data representation.

The use of quantum substrates as reservoirs has catalyzed a new wave of research, bridging the gap between quantum information theory and machine learning opening two distinct but complementary lines of research [30]:

- **Quantum for Quantum tasks**, in which the model is tasked with processing intrinsically quantum information. Recent studies have demonstrated

the efficacy of QELMs and QRCs in tasks such as entanglement detection, quantum state tomography, and the characterization of quantum phase transitions [27–54];

- **Quantum for Classical tasks**, in which the models can be deployed to solve classical computational problems. In this second case, classical data need to be encoded in quantum states [30, 48, 49, 101, 102].

This chapter provides the theoretical framework for both approaches, defining the mathematical formalism of QELM and QRC that will underpin the experimental and numerical results presented in Part II of this thesis.

2.1 Open Quantum Systems and the Density Matrix Formalism

In this section, we review the theory of open quantum systems and the density matrix formalism [103, 104].

In standard quantum mechanics, isolated systems evolve unitarily according to the Schrödinger equation and are completely described by pure state vectors $|\psi\rangle$ in a Hilbert space $\mathcal{H}$. However, this description is often insufficient for realistic physical systems, which are rarely strictly isolated and instead interact with a larger environment, or bath, E. To mathematically describe these open systems, pure states are inadequate. We require a statistical tool that simultaneously accounts for intrinsic quantum uncertainty and classical ignorance regarding the system's exact preparation state. This necessitates the density matrix formalism. If a system is prepared in an ensemble of pure states $\{|\psi_i\rangle\}_{i=1}^{N}$, each with a classical

probability p_i of occurring, the state is represented by the density operator:

$$\rho = \sum_{i=1}^{N} p_i |\psi_i\rangle\langle\psi_i|$$

. A valid density matrix must satisfy three physical constraints: hermiticity ($\rho = \rho^\dagger$) to guarantee real observable expectation values , positivity ($\rho \geq 0$) to ensure non-negative probabilities , and unit trace ($\mathrm{Tr}(\rho) = 1$) to ensure the conservation of total probability. A system whose preparation is perfectly known is pure and thus its state is known; on the other hand a system whose preparation has classical uncertainties such in the case of the interaction with an unknown external environment, is mixed and is described by an ensemble of states $\{p_i, |\psi_i\rangle\}$.

When our system S interacts with the environment E, the composite system $S+E$ generally evolves unitarily, and could technically be described by a joint pure state ρ_{SE}. However, in practice, we generally do not have access to the evolution of the full state of the composite system $S+E$, thus only having partial information on the system S and its preparation. The state ρ_S can be mathematically described departing from the total state of the composite system, using the partial trace

$$\rho_S = \mathrm{Tr}_E\{\rho_{SE}\}. \tag{2.1}$$

The partial trace operation mathematically corresponds to throwing away any information about the state of the environment and, since the interaction generates entanglement between the system and the bath, this localized information loss dictates that ρ_S will evolve into a mixed state. This distinction between unitary and dissipative evolution is the most important difference between QELMs and QRCs. As with their classical counterparts, Extreme Learning Machines do not require memory to function; the state of the reservoir is reset after processing each input.

Consequently, QELMs can utilize closed, unitary quantum dynamics. Quantum Reservoir Computing, however, relies strictly on the Echo State Property—a fading memory of past inputs. To prevent the reservoir from indefinitely retaining information (which would happen under unitary evolution), a QRC model strictly requires the non-unitary, dissipative dynamics of an open quantum system to trace out past inputs, leaving only remnants in the current state.

To formalize these concepts for our models, the next two sections introduce Quantum Channels—the unified mathematical framework for describing both unitary and dissipative evolutions—and Positive Operator-Valued Measures (POVMs), which govern the information extraction at the readout layer.

2.2 Quantum Channels

The most general description of quantum state evolution is provided by the concept of a quantum channel. This formalism provides a unified framework to describe the dynamics of both closed and open quantum systems, proving highly effective in the description of QELM and QRC models. From the perspective of quantum information theory, a quantum channel can be viewed as a communication medium—a "wire" through which quantum information is transmitted across space or time. An ideal, lossless wire corresponds to unitary evolution, preserving quantum information perfectly. A realistic wire, however, introduces quantum noise: the system interacts with an external environment, inducing information leakage and transforming the initial pure state into a statistical mixture.Mathematically, a quantum channel is a linear map $\mathcal{E} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ that transforms density operators on a Hilbert space $\mathcal{H}_A$ into density operators on $\mathcal{H}_B$ [103–105]. To ensure that this map represents a physically realizable process, it must strictly satisfy three conditions:

it must be Linear, Trace-Preserving, and Completely Positive (CPTP).Given an ensemble of quantum states $\{p_i, \rho_i\}$ forming a mixed state $\rho_{mix} = \sum_i p_i \rho_i$, the linearity condition ensures that the channel acts independently on the probabilistic components of the mixture:

$$\mathcal{E}\left(\sum_i p_i \rho_i\right) = \sum_i p_i \mathcal{E}(\rho_i). \tag{2.2}$$

The map must be trace-preserving to conserve total probability. For any input density matrix ρ where $\mathrm{Tr}\{\rho\} = 1$, the output must satisfy $\mathrm{Tr}\{\mathcal{E}(\rho)\} = 1$.Finally, the Complete Positivity (CP) condition ensures that the map yields valid physical states even when acting on subsystems of larger, entangled systems. A map is positive if $\mathcal{E}(\rho) \geq 0$, maintaining the probabilistic interpretation of the density matrix's eigenvalues. However, complete positivity requires that for any arbitrary auxiliary system of dimension n, the trivial extension of the map $\mathcal{E} \otimes \mathbb{I}_n$ remains positive.To demonstrate why mere positivity is insufficient, consider the transposition map T, defined by $T(\rho) = \rho^T$. The transpose map preserves the trace and is positive, as it leaves the eigenvalues of ρ unchanged. However, it fails the criterion of complete positivity. Consider a bipartite system consisting of two subsystems, A and B, prepared in a Bell state:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

A Bell state is a maximally entangled pure state. Physically, this means the two subsystems are perfectly correlated. Measuring subsystem A instantly determines the state of subsystem B with absolute certainty. The global density matrix for this entirely inseparable system is $\rho_{AB} = |\Phi^+\rangle\langle\Phi^+|$.If we apply the transpose map locally only to subsystem A—an operation known as the Partial Transpose, denoted

as $T \otimes \mathbb{I}$—the resulting matrix $(T \otimes \mathbb{I})(\rho_{AB})$ yields the following eigenspectrum:

$$\lambda = \left\{ \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, -\frac{1}{2} \right\}$$

The emergence of a negative eigenvalue $(-1/2)$ implies a negative probability of measuring the system in a specific state. Because a valid local operation cannot result in a globally unphysical state, the transpose map cannot represent a physical process.

Any open quantum system can be thought as a sub-system of a bigger closed composite system, whose evolution is perfectly unitary and reversible. The dilation theorem [106] states exactly that the dynamics of any quantum system evolving according to a non-unitary quantum channel $\mathcal{E}(\rho)$

$$\mathcal{E}(\rho_S) = \mathrm{Tr}_E \left[U_{SE} \left(\rho_S \otimes |0\rangle\langle 0|_E \right) U_{SE}^\dagger \right] \tag{2.3}$$

Of course from the computational point of view, computing the dynamics of the full system is unfeasible, but we can evaluate the partial trace in Equation eq. (2.3) over an orthonormal basis $\{|e_k\rangle\}$ of the environment. This projects the global unitary dynamics back down strictly into the local space of the system. This projection yields the Operator-Sum Representation, most commonly known in quantum information as the Kraus representation [104] :

$$\mathcal{E}(\rho) = \sum_k K_k \rho K_k^\dagger \tag{2.4}$$

where the operators $K_k = \langle e_k | U_{SE} | 0 \rangle_E$ are the Kraus operators. The Kraus operators act strictly on the Hilbert space of the principal system $(\mathcal{H}_S \to \mathcal{H}_S)$, but they are still able to carry the information about the inaccessible environment. Because the channel must preserve total probability, the trace-preserving condition imposes

a strict algebraic constraint on the Kraus operators, known as the completeness relation:

$$\sum_k K_k^\dagger K_k = \mathbb{I}_S \tag{2.5}$$

2.3 Measurements and POVMs

As in the classical framework, a critical component of Quantum Reservoir Computing is the readout layer—the mechanism through which the high-dimensional internal state of the reservoir is mapped to a classical output [103]. While in Echo State Networks this corresponds to reading neuron activations, in the quantum domain, extracting classical information from a quantum state ρ requires a quantum measurement.The simplest approach is the projective measurement of an observable $\mathbf{O}$. By spectral decomposition, the observable can be written in terms of orthogonal projectors Π_j onto its eigenspaces:

$$\mathbf{O} = \sum_j m_j \Pi_j$$

where m_j are the real eigenvalues representing the possible measurement outcomes. The probability of obtaining a specific outcome m_j from a system in state ρ is given by Born's rule:

$$p(m_j) = \mathrm{Tr}(\Pi_j \rho)$$

However, standard projective measurements are overly restrictive for general open quantum systems or complex readout architectures. In many physical implementations, we are solely interested in the probabilities of the measurement outcomes and not in the specific post-measurement state of the system. This requires the most general formulation of a quantum measurement: the Positive Operator-Valued

Measure (POVM).A POVM is defined by a set of generalized measurement operators $\{\mathbf{M}_j\}$. The probability of outcome j is determined by:

$$p(j) = \mathrm{Tr}(\mathbf{M}_j^\dagger \mathbf{M}_j \rho)$$

We can define a new set of operators $\mathbf{E}_j = \mathbf{M}_j^\dagger \mathbf{M}_j$. Because $\mathbf{M}_j^\dagger \mathbf{M}_j$ is positive semi-definite, each $\mathbf{E}_j \geq 0$. To ensure that the sum of all probabilities equals unity for any arbitrary state ρ, these operators must satisfy the completeness relation:

$$\sum_j \mathbf{E}_j = \mathbb{I}$$

The complete set of operators $\{\mathbf{E}_j\}$ constitutes the POVM. Each element $\mathbf{E}_j$ contains all the necessary information to determine the probability of a given measurement outcome, completely characterizing the readout mechanism.

2.4 Processing data with QELMs and QRCs

Having established the necessary theoretical tools, we can now define the Quantum Extreme Learning Machine framework.

The workflow for a quantum reservoir depends on the nature of the input data. In the case of intrinsically quantum tasks, the input is already a quantum state ρ_{in}. For classical tasks, a pre-processing step is required to encode classical vectors into quantum states (a topic we are going to review in the next section). For the present discussion, let us assume the input is provided as a set of labeled quantum states $\{\rho_{in}^k, \mathbf{y}_k\}_{k=1}^{N_{train}}$.

In the QELM framework, the classical non-linear map $G(\mathbf{x})$ is replaced by a

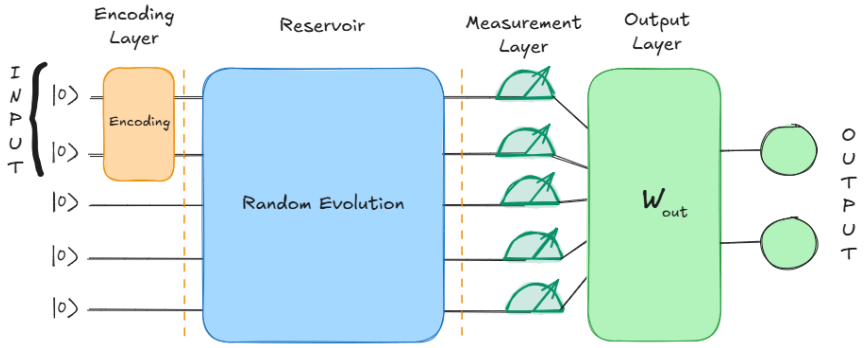


Figure 2.1: **QELM Implementation on quantum circuit** - Data are encoded through specific operation on the initial state of a subset of qubits in the circuit. The reservoir in this context consists of a set of random unitary operation on all the qubits of the circuit. They are then measured through projective measurements, whose outcomes corresponds to the probabilities of measuring a specific state. The probabilities are then used to train W_{out} to approximate the target function $\mathbf{Y} = f^*(\mathbf{X})$.

quantum process. This consists of a quantum channel $\mathcal{E}$ (representing the reservoir dynamics) followed by a measurement process. The measurement is defined by a set of observables $\{O_j\}_{j=1}^{N_{out}}$ or POVM elements $\{E_j\}$. The feature vector for the k-th input is thus given by the expectation values (or probabilities) [29, 30]:

$$P_{j,k} = \mathrm{Tr}[O_j \mathcal{E}(\rho_{in}^k \otimes \rho_{res})]. \tag{2.6}$$

Here, $P_{j,k}$ represents the j-th feature of the k-th training sample. Note that if $\{O_j\}$ defines a POVM, these values correspond strictly to probabilities; if $\{O_j\}$ are arbitrary observables (e.g., Pauli matrices), they correspond to expectation values.

The function of the quantum model $f_{model}^q : \mathcal{H}^{in} \to \mathcal{H}^{out}$, mapping the input state to the target output, is then constructed as a linear combination of these quantum features:

$$y_k = f^q_{model}(\rho^k_{in}) = \sum_{j=0}^{N_{out}} W_{i,j} Q_{j,k} \tag{2.7}$$

where $\mathbf{W}_{out}$ is the trainable weight matrix. A pictorial representation of the model is shown in fig. 2.1.

As in the classical context, QELM does not have any memory requirement and thus the quantum channel can be either unitary or non-unitary, with the condition that - after each interaction with an input data - the state of the reservoir is reset to a standard initial condition; on the other hand, for QRC implementations, the reservoir must satisfy the conditions listed in section 1.3, which in this framework correspond to requiring a non-unitary quantum channel, as unitary dynamics preserve information indefinitely (infinite memory). [30, 107, 108] Therefore eq. (2.6) is modified considering also the state update - refer to fig. 2.2 - of the reservoir which tracks back previous inputs retaining memory of them

$$Q_{j,k} = \mathrm{Tr}\{O_j \mathcal{E}(\rho^k_{in} \otimes \rho^{k-1}_{res})\}, \tag{2.8}$$

where $\mathcal{E}$ must be strictly a non-unitary quantum channel. As in the classical context, we can ensure the QRC model possesses the properties listed in section 1.3 by solving the memory tasks for the fading memory; on the other hand, to check whether the reservoir possesses the ESP, in this context we have to consider two copies of the reservoir in two different initial state ρ_0, ρ'_0 and quantify the trace distance between these two states after the interaction with the same input sequence [32, 95]. The reservoir has the ESP if and only if

$$\lim_{t\to\infty} \frac{1}{2} \mathrm{Tr}\{\sqrt{(\rho_t - \rho'_t)^\dagger (\rho_t - \rho'_t)}\} = 0 \tag{2.9}$$

2.5 Encoding Strategies for Classical Data

The workflow of QELM and QRC applied to classical tasks begins with the crucial step of data encoding. We must map a classical input vector $\mathbf{x} \in \mathbb{R}^n$ into a quantum state $\rho_{\mathbf{x}}$ living in the Hilbert space of the reservoir. This mapping, often referred to as the quantum feature map, is non-trivial; it determines both the expressivity of the model (the richness of the feature space) and its feasibility on physical hardware.

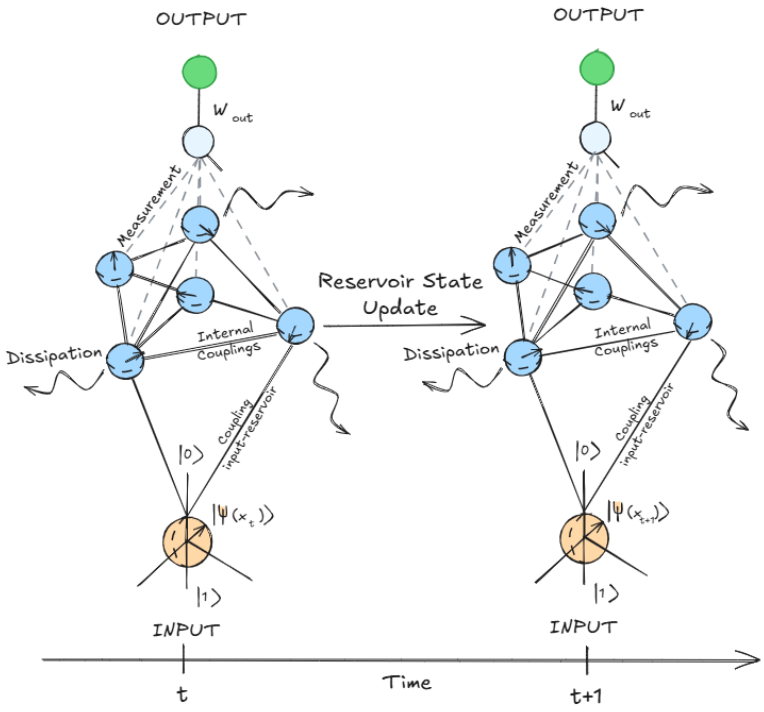


Figure 2.2: **Quantum Reservoir Computing** - The data is encoded in a input quantum state which interacts with a open quantum reservoir. A set of measurements is performed on the reservoir, whose outcome is then used to train the linear output layer. External dissipation provide the reservoir with fading memory. In this framework the state of the reservoir is not reset, thus is state depends on the previous input history.

2.5.1 The NISQ Constraint

Current quantum computations is only implemented on **Noisy Intermediate-Scale Quantum** (NISQ) devices [109, 110]. These are characterized by a limited number of qubits (typically $50-150$) and with error correction consisting in encoding one logical qubit into several physical qubits [111–113], thus reducing the effect of noise on the resulting state. Each quantum device comes with a set of basic operations which is possible to perform on each qubit. Indeed, while it is technically possible to implement any quantum channel on a quantum device, it is also true that from an experimental point of view, the final operation gets decomposed in a number of more basic operations referred to as quantum gates. These are not perfect and generally each quantum processor carries among its specifics both the coherence times for each qubit and the errors generated from each gate operation. This imposes a strict constraint on circuit depth: if a circuit takes too long to run, environmental noise (decoherence) will degrade the quantum information before the computation is complete, rendering the result useless.

This dictates a strict requirement for encoding strategies: they must be resource-efficient. An ideal encoding scheme for QELMs should map classical data into a high-dimensional feature space using a minimal number of qubits and a shallow state preparation circuit. The choice of encoding thus represents a trade-off between information compression and circuit complexity.

2.5.2 Common Encoding Schemes

For gate-based quantum circuits, the choice of encoding depends on the data type and the available resources. The three most common strategies are:

- **Basis Encoding:** Primarily used for binary or integer data. A binary input string $\mathbf{b} = (b_1, \ldots, b_n)$ is mapped to the computational basis state $|b_1 \ldots b_n\rangle$ by applying Pauli-X gates to qubits where $b_i = 1$. While conceptually simple, this method scales linearly with the input dimension $O(n)$ and is inefficient for continuous data, which requires discretization [30].

- **Angle (Rotation) Encoding:** This is the standard strategy for continuous data in NISQ applications. The components of the input vector $\mathbf{x} = (x_1, \ldots, x_n)$ serve as rotation angles for parameterized quantum gates. For example, a layer of R_x rotations transforms the initial state as:

$$|\psi_{\mathbf{x}}\rangle = \bigotimes_{i=1}^{n} R_x(x_i)|0\rangle_i = \bigotimes_{i=1}^{n} \left(\cos(x_i/2)|0\rangle - i\sin(x_i/2)|1\rangle\right), \tag{2.10}$$

  where $R_x(\theta) = e^{-i\frac{\theta}{2}\sigma_x}$. A more expressive variant, the **Dense Angle Encoding**, utilizes two rotations per qubit (e.g., R_x and R_z) to encode two features x_i, x_j into a single qubit's relative phase and amplitude:

$$|\psi_{\mathbf{x}}\rangle = \bigotimes_{k=1}^{n/2} R_z(x_{2k})R_x(x_{2k-1})|0\rangle_k. \tag{2.11}$$

  This method is **hardware efficient**: the state preparation requires a constant depth circuit ($O(1)$), making it highly robust to noise. It naturally introduces non-linearity via the cosine and sine functions. While it scales linearly in qubits (n or $n/2$). [48, 49, 101].

- **IQP Embedding**: In contrast to earlier encoding strategies, the Instantaneous Quantum Polynomial (IQP) encoding [114] introduces non-linear feature mapping by interleaving single-qubit rotations with two-qubit entangling gates. The procedure begins by preparing an initial superposition state

via a layer of Hadamard gates applied to all N_q qubits, followed by local Z-rotations parameterized by the individual classical features x_i. This initial step prepares the state:

$$|\psi(\mathbf{x})\rangle = \bigotimes_{i=1}^{N_q} R_z(x_i) H |0\rangle_i. \tag{2.12}$$

To capture correlations between features, the encoding is further enriched by applying two-qubit ZZ rotations, defined as $ZZ(x_i, x_j) = e^{-ix_i x_j \sigma_i^z \otimes \sigma_j^z}$, typically across adjacent qubit pairs. This operation effectively embeds quadratic, non-linear products of the input data into the quantum state. To increase the expressivity of the feature map, this entire foundational block can be repeated d times. While repeating these encoding layers successfully maps the classical data into a higher-dimensional, more complex Hilbert space, it concomitantly increases the overall depth of the quantum circuit, rendering it more susceptible to hardware noise.

- **Amplitude Encoding:** This method aims to maximize information density by encoding $2^n - 1$ features into the 2^n amplitudes of an n-qubit state:

$$|\psi_{\mathbf{x}}\rangle = \sum_{i=0}^{2^n-1} x_i |i\rangle, \quad \text{with } \sum |x_i|^2 = 1. \tag{2.13}$$

While this technique permits to encode a very high number of parameters compared to the others, encoding a quantum state with such a high number of features requires a very deep quantum circuit making it generally inefficient for the current NISQ devices [102];

- **Data Re-uploading**: Finally, the data re-uploading [115] strategy circumvents the expressivity limitations of a single encoding layer by repeatedly injecting the classical data into the quantum system. This technique con-

sists of the alternating application of a data-encoding unitary, $U_{enc}(\mathbf{x})$, and a universal quantum processing block—in this context, the quantum reservoir, U_{res}. By repeating this sequence r times, the system is able to exploit any correlation created by the reservoir to enrich the expressivity of the encoding, thus introducing higher order features in the quantum state. The final encoded state is thus given by:

$$|\psi(\mathbf{x})\rangle = \prod_{k=1}^{r} U_{res}^{(k)} U_{enc}^{(k)}(\mathbf{x})|0\rangle^{\otimes N_q} \tag{2.14}$$

This sequential re-introduction of data allows the circuit to approximate arbitrary continuous functions, effectively utilizing the reservoir's evolution to map the classical input into a highly expressive quantum feature space.

2.5.3 Encoding for QRC

In the context of QRC there are some more specific encoding strategies consisting in encoding the input directly in the hamiltonian governing the dynamics of the system. The most utilized framework is that of the Ising Hamiltonian:

$$H_0 = \sum_{i,j} J_{i,j}\sigma_i^x\sigma_j^x + \sum_i h_i\sigma_i^z. \tag{2.15}$$

In this framework, the general approach involves encoding the input sequence directly into the Hamiltonian by inserting a secondary local term [32]. This modification yields the total Hamiltonian:

$$H_{enc}(t) = h(1 + x_t)\sum_i \sigma_i^x \tag{2.16}$$

therefore modifying the full hamiltonian as follows

$$H(t) = H_0 + H_{enc}(t). \tag{2.17}$$

Subsequently, external decoherence is introduced to induce memory within the system; an alternative approach consists in considering a collisional approach in which each input of the sequence is encoded through an amplitude encoding in a single qubit, [30, 107]

$$\rho_{in}^t = x_t|0\rangle\langle 0| + (1 - x_t)|1\rangle\langle 1| \tag{2.18}$$

then it evolves unitarily with the reservoir, following the hamiltonian of eq. (2.15), and the state of the reservoir is updated by tracing out the qubit encoding the input

$$\rho_{res}^{t+1} = \mathrm{Tr}\{U\rho_{in}^t \otimes \rho_{res}^t U^\dagger\}. \tag{2.19}$$

With this method, the memory is preserved by the reservoir, by tracing out the input after the interaction with the reservoir [116].

2.6 Exponential Concentration

As previously discussed, the training phase consists solely of optimizing classical linear regression weights on the measured observables, bypassing any optimization on the hidden layers. However, this guaranteed trainability does not inherently guarantee scalability.As the size of the quantum system increases, the expectation values of the output observables can exponentially concentrate around fixed, input-independent values. This phenomenon is driven by multiple distinct mechanisms [43, 117]. First, the choice of measurement can induce concentration. When an observable acts non-trivially on all N_q qubits—such as projecting onto a spe-

cific computational basis state $O = |m\rangle\langle m|$—the expectation value exponentially vanishes due to the vastness of the 2^{N_q}-dimensional Hilbert space. Obtaining significant statistics to distinguish this signal from zero requires an exponentially scaling number of shots. Furthermore, concentration can also be induced from scrambling dynamics, which we are going to review more in detail in the next chapter. These dynamics delocalize information across the degrees of freedom of a quantum system, making it inaccessible through local measurements. If the unitary evolution U_R of the reservoir is chaotic enough to approximate a Haar random unitary, the expectation value of any observable will converge to an input independent value. In [43] it has been proved that the probability of the expectation value deviating from this mean by more than δ is bounded by:

$$Pr_{U_R}[|\langle O\rangle_x - Tr[O]/d| \geq \delta] \leq O(d^{-1})$$

where $d = 2^{N_q}$ and $\langle O\rangle_x = \mathrm{Tr}\{OU_R\rho(x)U_R^\dagger\}$. Rather than yielding information, this scrambling obliterates the input dependency. Furthermore, concentration can happen also when measuring local observables. For instance, in cases in which the measured sub-system is too entangled to the unmeasured portion of the reservoir, tracing out the hidden part of the reservoir takes the system into a maximally mixed state. Finally, even if local observables are measured, concentration occurs if the measured subsystem is highly entangled with the unmeasured "hidden" reservoir. In this case, tracing out the hidden qubits leaves the accessible subsystem exponentially close to a maximally mixed state. Hardware Noise: Finally, executing these models on near-term hardware introduces local errors, such as Pauli noise, at each layer of the reservoir evolution. This acts as a contractive dissipative channel, forcefully driving the density matrix of the full system toward a maximally mixed state. As demonstrated by [117], restricting the reservoir dy-

namics via Hamiltonian symmetries limits the state's ergodic exploration of the Hilbert space, effectively preventing this exponential concentration and preserving the distinguishability of the observables.

2.7 Properties of the Reservoir

In this chapter, we have established the formal framework for Quantum Extreme Learning Machines and Quantum Reservoir Computing. We have defined the necessary components: the input encoding strategies to map classical data into the Hilbert space, the open quantum system formalism to describe reservoir dynamics, and the measurement protocols required for information extraction.

In the next part of this work, we will assess the properties of the reservoir, specifically in the context of QELM. While in an idealized scenario, a random unitary matrix coupled with linear post-processing (and non-linear encoding) acts as a universal function approximator, we aim to investigate which specific properties render a reservoir capable of efficiently processing information.

For this reason, the next part of this thesis begins with a review of the Quantum Information Scrambling framework [55–65]. We will then show the first new results of these thesis, by employing this framework to analyze the dynamical properties that a quantum system must possess to be classified as an effective reservoir [28].

Part II

The Quantum Information Scrambling Perspective on QELMs

Chapter 3

Quantum Information Scrambling

Quantum Information Scrambling (QIS) [55–65] is a branch of information theory that studies the delocalization of information within a quantum system governed by unitary dynamics. Unlike decoherence, where information is lost to an external environment, scrambling describes the spreading of local information across the system's many-body degrees of freedom, effectively encoding it into global entanglements.

The concept of scrambling was originally formulated in the context of the **Black Hole Information Paradox** [118, 119] to explain how quantum information falling into a black hole is processed and subsequently reflected as Hawking radiation.

As discussed in the previous chapter, the high dimensionality of the Hilbert space is a fundamental asset for processing information. However, for a quantum system to work effectively as a reservoir, it is not sufficient to merely access a large Hilbert space; the system must be endowed with *rich internal dynamics*. In this context, scrambling provides significant insights into the complexity of these

dynamics, while simultaneously offering rigorous methods to quantify the amount of locally retrievable information within the system.

In this chapter, we review the theoretical framework of QIS and define its two most commonly used quantifiers—specifically the **Out-of-Time-Order Correlator (OTOC)** and **Holevo Information**. These definitions establish the necessary background for the next chapter, where we present the results obtained using scrambling metrics to analyze the performance of Quantum Extreme Learning Machines (QELM) in state estimation tasks.

3.1 Out-of-Time-Order Correlators

Given two quantum systems A and B interacting via a specific quantum dynamics, the information initially localized in A is said to be **scrambled** when it is no longer possible to retrieve it through local measurements on the joined system $A + B$.

The most widely employed metric to quantify this phenomenon is the **Out-of-Time-Order Correlator** (OTOC) [55–58, 120–126]. Conceptually, given two initially uncorrelated systems A and B, OTOCs estimate the degree of non-commutativity between two initially commuting local operators as the two systems interact.

Formally, consider a Hilbert space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$. Let O_A and O_B be two local operators acting on subsystems A and B, respectively. Prior to interaction ($t = 0$), these operators act on disjoint subspaces and therefore commute:

$$[O_A, O_B] = 0. \tag{3.1}$$

When the two systems interact under a quantum channel Φ, they become more and more correlated, so that in the Heisenberg picture formalism, the two operators

will not commute anymore:

$$[O_A(t), O_B] \neq 0. \tag{3.2}$$

We quantify such correlation growth using the expectation value of the squared commutator. In the infinite temperature limit (where the state of the system is the maximally mixed state $\rho = \mathbb{I}/d$) - for finite dimensional systems - this is defined as:

$$C(t) = \langle |[O_A(t), O_B]|^2 \rangle_{\beta=0} = \frac{1}{2d} \mathrm{Tr}\{\big([O_A(t), O_B]^\dagger [O_A(t), O_B]\big)\}, \tag{3.3}$$

where d is the dimension of the Hilbert space. This quantity $C(t)$ serves as the scrambling quantifier, measuring the extent to which the operator $O_A(t)$ becomes correlated with the operator O_B.

In the specific case where O_A and O_B are both Hermitian and unitary (e.g., Pauli operators), eq. (3.3) simplifies. Expanding the commutator, we obtain [57, 127]:

$$C(t) = 1 - \frac{1}{d} \mathrm{Re}(\mathrm{Tr}\{O_A(t) O_B O_A(t) O_B\}). \tag{3.4}$$

The term inside the trace in eq. (3.4), $F(t) = \frac{1}{d}\mathrm{Re}(\mathrm{Tr}\{O_A(t)O_BO_A(t)O_B\})$, is what is referred to as the **OTOC**. As the system scrambles, $C(t)$ grows from 0 to 1, while the OTOC $F(t)$ decays from 1 to 0.

The time scale in which the OTOC $F(t) \to 0$ is what is generally referred to as **scrambling time**.

3.2 Holevo Information

While OTOCs characterize the dynamical degree of non-commutativity of local operators, a complementary perspective is offered by quantum information theory.

Specifically, we focus on the **Holevo information** [128].

Given two observers, namely Alice and Bob, suppose Alice wants to send a message X to Bob. Suppose Alice encodes the message into an ensemble of quantum states $\eta = \{p_i, \rho_i\}_{i=1}^{M}$. In order to retrieve the message, Bob will perform a POVM on the system $\mathbf{\Pi} = \{\Pi_k\}$.

The amount of information about the original message - encoded by Alice in each state i of the mixture and retrieved by Bob through a measurement k - corresponds to the mutual information $I(X:Y)$ of the joint probability distribution:

$$p(i,k) = p_i \mathrm{Tr}\{(\Pi_k \rho_i)\}. \tag{3.5}$$

The **accessible information** [103,128,129] corresponds to the maximal of this communication protocol corresponds to the maximal mutual information $I(X:Y)$ which Bob can access by optimizing over all the possible measurement protocols:

$$I_{\mathrm{acc}}(\eta) = \max_{\mathbf{\Pi}} I(X:Y). \tag{3.6}$$

Computing this quantity is generally computationally expensive. Nonetheless, Holevo Theorem poses a more accessible upper bound to this quantity, which we are going to refer to as **Holevo Information** [128]

$$I_{\mathrm{acc}}(\eta) \leq \chi(\eta) \equiv S\left(\sum_i p_i \rho_i\right) - \sum_i p_i S(\rho_i), \tag{3.7}$$

where $S(\rho) = -\mathrm{Tr}\{\rho \log \rho\}$ is the Von Neumann entropy.

3.3 Holevo Information as a Scrambling Quantifier

Recently, the Holevo Information has been employed to characterize scrambling [59,60,64]. The operational intuition is distinct from OTOCs: if a system scrambles information effectively, local measurements should fail to distinguish the input states.

Let us consider a scenario where Alice encodes information using a set of states that evolve under a quantum channel Φ, generating the ensemble $\eta = \{p_i, \Phi(\rho_i)\}$. To quantify scrambling, we analyze the information recoverable from a specific subsystem j of the Q-partite reservoir. The state of this subsystem is obtained by tracing out the environment (the remaining $Q-1$ parts):

$$\rho_i^{(j)} = \mathrm{Tr}_{\bar{j}}[\Phi(\rho_i)], \tag{3.8}$$

where $\mathrm{Tr}_{\bar{j}}$ denotes the partial trace over the complement of j. The local Holevo information is thus given by:

$$\chi^{(j)} = S\left(\sum_i p_i \rho_i^{(j)}\right) - \sum_i p_i S(\rho_i^{(j)}). \tag{3.9}$$

This quantity, $\chi^{(j)}$, bounds the maximal accessible information about the input retrievable by an observer with restricted access to the subsystem j only. High values of $\chi^{(j)}$ mean that the input states $\rho_i^{(j)}$ remain distinguishable locally. As $\chi^{(j)} \to 0$, the local states become indistinguishable (typically converging to the maximally mixed state). This implies that the information has been delocalized into global correlations, accessible only via joint measurements on the entire system.

3.4 The Scrambling Perspective on QELM

As discussed, QIS describes the delocalization of information into global correlations, effectively hiding it from local observation. This phenomenon poses a potential conflict for QELM, especially in contexts where the readout layer employs local measurements to reconstruct the output. Consequently, one might guess that scrambling dynamics are inefficient for learning, as the relevant information would be inaccessible to the model.

In the next chapter, we will analyze the performance of QELMs in state estimation tasks, examining how reconstruction accuracy evolves versus the scrambling time. By correlating the model's predictive power with the OTOC and the Holevo information defined here, we aim to determine if effective learning can persist beyond the scrambling time.

Chapter 4

State Estimation with QELM beyond the Scrambling Time

In this chapter, we present our results obtained in [28], investigating the interplay between the predictive capabilities of QELMs and the dynamics of information spreading. As anticipated in the previous chapter, we address a state estimation task using a reservoir subject to unitary evolution, interpreting the model's performance through the lens of the QIS framework.

While in general when working with quantum hardware [130], the readout layer is trained through the entire outcome probabilities of the state of the reservoir, in this context we restrict ourselves to local measurements. In the context of QIS, this framework should be characterized by fundamental limitations, since the information on the inputs should be hidden in non-local correlations by definition [55–65]. Consequently, standard intuition suggests that once the system evolves beyond the scrambling time, the reconstruction of the input state should become inefficient or impossible. Contrary to this expectation, our findings demonstrate that the

information retrieved via local measurements remains sufficient for accurate state estimation well into the scrambling regime.

Furthermore, we analyze how the internal structure of the reservoir influences this behavior. We show that while the interaction topology significantly affects accuracy during the initial transient phase of the evolution, these structural differences vanish at longer timescales, matching the performances of a random unitary evolution.

4.1 The Task

In this work, we address the problem of single-qubit state estimation. Formally, this entails reconstructing the **Bloch Vector B** of an input state $\rho_{in} = \frac{1}{2}(\mathbb{I}+\mathbf{B}\cdot\boldsymbol{\sigma})$. This task is equivalent to estimating the expectation values of the Pauli matrices $\boldsymbol{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$ given the input state:

$$\mathbf{y} = \mathrm{Tr}\{\boldsymbol{\sigma}\rho_{in}\}. \tag{4.1}$$

The reservoir comprises N_q qubits, upon which we perform local projective measurements of the σ_z operator following the unitary interaction with the input. Our objective is to analyze the time dependence of the reconstruction accuracy by measuring the reservoir at distinct, longer and longer, interaction times. Consequently, the training phase requires the optimization of a specific readout layer, $\mathbf{W}^{(t)}$, specialized for processing the measurement outcomes at each specific time step t. Given a labeled training dataset $\mathcal{D} = \{(\mathbf{y}_k, \rho_{in}^k)\}_{k=1}^{D_{train}}$, the output of the

QELM for the k-th sample is defined as:

$$y_k^j(t) = \sum_{i=1}^{N_q} W_{i,j}^{(t)} \mathrm{Tr}\{\Phi_{(t)}^{\dagger}(\sigma_z^i)\rho_{in}^k \otimes \rho_{res}\} = \sum_{i=1}^{N_q} W_{i,j}^{(t)} \langle \sigma_z^i \rangle_k^{(t)}. \tag{4.2}$$

Here, $\langle \sigma_z^i \rangle_k^{(t)}$ denotes the expectation value of σ_z on the i-th reservoir qubit at time t for the k-th input. Following the ELM training protocol described in section 1.2.1, if we define $\mathbf{Y}$ as the matrix of target vectors and $\langle \boldsymbol{\sigma}_z \rangle^{(t)}$ as the matrix collecting the reservoir measurements at time t, the optimal weights are computed via the Moore-Penrose pseudoinverse:

$$\mathbf{W}^{(t)} = \mathbf{Y}\boldsymbol{\sigma}_z^{(t)+}. \tag{4.3}$$

To quantify the accuracy of the reconstructions we employ the Mean Squared Error, averaged over the whole testing set

$$\mathrm{MSE}(\mathbf{Y}, \mathbf{Y}^{target}) = \frac{1}{D_{test}} \sum_{k=1}^{D_{test}} |\mathbf{y}_k - \mathbf{y}_k^{target}|^2 \tag{4.4}$$

4.2 Reservoir and Topologies

As briefly discussed in the previous section, the total system consists of $N_q + 1$ qubits, evolving under a unitary, time-independent Hamiltonian. The first N_q qubits constitute the reservoir, upon which we perform local measurements for the state estimation, while the (N_q+1)-th qubit constitutes the input node, initialized in a generally unknown state ρ_{in}.

The total Hamiltonian governing the system's evolution is decomposed into two factors: $H = H_{res} + H_{inj}$. Here, H_{res} describes the internal dynamics of the

reservoir, while H_{inj} governs the coupling between the input qubit and the reservoir. The reservoir Hamiltonian includes local driving terms acting on individual qubits as well as arbitrary two-body interactions:

$$H_{res} = \sum_{i<j}^{N_q} \sum_{\alpha,\beta}^{x,y,z} J_{i,j}^{\alpha,\beta} \sigma_i^\alpha \otimes \sigma_j^\beta + \sum_{i=1}^{N_q} \sum_{\alpha}^{x,y,z} h_i^\alpha \sigma_i^\alpha, \tag{4.5}$$

where the indices i, j run over the reservoir qubits, and $\alpha, \beta \in \{x, y, z\}$ denote the Pauli operators.

On the other hand, the injection Hamiltonian H_{inj} consists only of two-body interaction terms connecting the input qubit (index in) to the reservoir:

$$H_{inj} = \sum_{i=1}^{N_q} \sum_{\alpha,\beta}^{x,y,z} J_{i,in}^{\alpha,\beta} \sigma_i^\alpha \otimes \sigma_{in}^\beta. \tag{4.6}$$

To ensure the generality of the results, the coupling constants are treated as random variables. The local fields h_i^α and the interaction strengths $J^{\alpha,\beta}$ are drawn uniformly from the intervals $[-0.1, 0.1]$ and $[-1, 1]$, respectively. All numerical results presented in this chapter are obtained by averaging over an ensemble of 500 distinct Hamiltonian realizations. Furthermore, we consider a reservoir composed of $N_q = 7$ and a sampling statistics fixed to 10^6 measurements. The sampling statistics varies the maximal reconstruction accuracy, which has been proved in [29] to have a dependence which scale as $\frac{1}{N}$ with the measurements statistics.

In this work, we investigate the impact of connectivity of the reservoir into both the accuracy and the scrambling dynamics. In particular, we consider three different reservoir topologies scaling from lower to higher connectivity and also in expected scrambling capabilities: open Chain (C), a Ring (R) with periodic

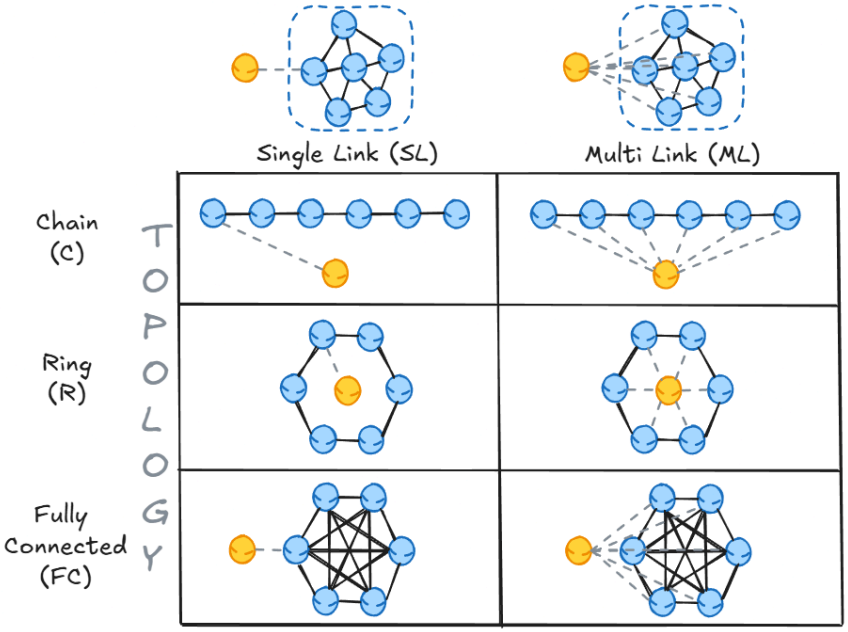


Figure 4.1: **Reservoir Topologies and Injection Schemes** - Different topologies and injection schemes used to couple the qubits in the system. Each different reservoir topology, respectively Chain (C), Ring (R) and Fully Connected (FC) corresponds to a different hamiltonian H_{res}. Each injection scheme, namely Single Link (SL) and Multi Link (ML) corresponds to a different number of interaction terms between the input and the reservoir in H_{inj}. In particular, in the first case we have a one-to-one interaction, while in the second case a one-to-all interaction.

boundary conditions, and a Fully Connected (FC) graph where all qubits interact with one another.

Furthermore, we employ two distinct injection strategies: the Single Link (SL) scheme, where the input qubit couples to a single node of the reservoir and the Multi Link (ML) scheme, where the input interacts directly with every qubit in the reservoir.

4.3 Scrambling Quantifiers

4.3.1 Non-Commutativity

Having defined the state estimation protocol, we now discuss the connection between the QELM performance and the scrambling properties of the reservoir. As discussed in chapter 3, information scrambling is operationally defined by the growth of non-commutativity between initially commuting local operators.

To adapt this framework to our workflow, we measure the OTOC between the evolved operators $\{\sigma_i^z(t)\}_{i=1}^{N_q}$, and the local operators acting on the input qubit at $t = 0$, denoted as $\{\sigma_{in}^\alpha\}$ where $\alpha \in \{x, y, z\}$. The specific OTOC component $F_i^\alpha(t)$ is defined as [57, 127]:

$$F_i^\alpha(t) = \frac{1}{2^{N_q}} \mathrm{Tr}\{\sigma_i^z(t)\sigma_{in}^\alpha \sigma_i^z(t)\sigma_{in}^\alpha\}. \tag{4.7}$$

To obtain a global quantifier of the information scrambling in the system, we compute the average non-commutativity $\bar{C}(t)$. This is obtained by averaging the decay of $F_i^\alpha(t)$ over all reservoir qubits i and all input Pauli bases α:

$$\bar{C}(t) = 1 - \frac{1}{3N_q} \sum_{i=1}^{N_q} \sum_{\alpha}^{x,y,z} F_i^\alpha(t). \tag{4.8}$$

As the system scrambles, $F(t) \to 0$ and $\bar{C}(t) \to 1$, meaning that the local information of the input operator has been delocalized into global correlations.

4.3.2 Holevo Information

Following the reasoning of section 3.3, in this framework the variable X encoded by Alice is the initial unknown state of the qubit, which is distributed among the qubits in the reservoir through a set of quantum channels Φ_k^t. Defining the full state of the system $\rho = \rho_{in} \otimes \rho_{res}$, due to the unitarity of the quantum channel governing the full dynamics of ρ, it is possible to describe the encoding quantum channels of Alice Φ_k^t as follows:

$$\rho_k(t) = \Phi_k^t(\rho) = \mathrm{Tr}_{\bar{k}}\{e^{-iHt}\rho e^{iHt}\} \tag{4.9}$$

where $\mathrm{Tr}_{\bar{k}}\{\cdot\}$ represents the trace over the qubits complementary to k.

Therefore, for each qubit of the reservoir, Alice has at her disposal three ensembles of quantum states $\eta_k^\alpha = \{p_k^\alpha, \Phi_k^t(\rho^\alpha)\}_{k=1}^d$, where k runs over the qubit in the reservoir, and α is the encoding basis of the input. Indeed, in order to ensure that the Holevo information does not depend on the basis of states Alice is using to encode the message, we consider a set of balances probabilities $p_k^\alpha = \frac{1}{2}$ and compute the **local Holevo information** for each qubit in the reservoir considering all the possible encoding eigenbasis for the input, namely σ_x, σ_y and σ_z.

We thus define the local Holevo information as [59, 60, 64]:

$$\chi_k^\alpha(t) = S\left(\frac{1}{2}\sum_{i=1}^{2}\rho_{i,k}^\alpha(t)\right) - \frac{1}{2}\sum_{i=1}^{2}S(\rho_{i,k}^\alpha(t)), \tag{4.10}$$

where i runs over the number of eigenvectors of each eigenbasis α and k is the index of the corresponding qubit in the reservoir.

As in the non-commutativity framework we compute the average Holevo information in the reservoir by averaging over all the qubits in the reservoir and the

possible encoding eigenbasis used by Alice:

$$\bar{\chi}(t) = \frac{1}{3N_q} \sum_{k=1}^{N_q} \sum_{\alpha}^{x,y,z} \chi_k^{\alpha}(t) \tag{4.11}$$

The remarkable feature of the definition of local Holevo information in eq. (4.10) is that it is able to quantify the distinguishability of the state of each qubit in the reservoir after the interaction with orthogonal input state though the quantum channel $\Phi_k^t(\rho^\alpha)$. If after the scrambling time $\chi_k^\alpha \to 0$ this would imply that Bob should not be able to distinguish completely different input states in the reservoir.

4.4 Results

In this section we report the results obtained in our work [28]: while in previous works the accuracy of QELMs was proven to depend on the reservoir dynamics and in particular on the topology of the reservoir [29,36,52], in our work we prove how this dependence on the topology disappears beyond the scrambling time, with all the reservoirs presenting the same accuracy, saturating to a minimal value. This implies the possibility of implementing a QELM without the need of temporal fine-tuning.

However, while the accuracy does not depend on the topology after the scrambling time, we highlight the presence of better performances in the transient regime leading to the full scrambling. Remarkably, this optimal performance shows correlation with the local Holevo information, which shows a higher distinguishability of the outcome states of the qubits in the reservoir, especially in the less connected topologies. On the other hand, this same correlation is not seen in the behavior

of the non-commutativity that, while successfully addressing the scrambling time, does not provide any information on the accuracy of the model, highlighting the differences in the QIS features they measure.

4.4.1 Reservoir Dimension

In fig. 4.2, we analyze the reconstruction Mean Squared Error (MSE) as a function of the reservoir dimension N_q, varying the size from 2 to 7 qubits. To isolate the effects of the internal reservoir structure, we employ the SL injection strategy. This ensures that the complexity of the input-coupling remains constant regardless of the reservoir size. We compare two distinct dynamical regimes: a short interaction time ($t = 0.25$, panel **a**) and a long interaction time ($t = 5$, panel **b**).

Note that with $N_q < 4$ qubits in the reservoir there is not enough information to retrieve the initial state of the input, obtaining an higher MSE for those points. The data show that, while for short interaction time with the reservoir it is possible to observe differences in the behavior of each topology, as reported in [29], for longer interactions these differences disappear. This suggests that as the interaction lasts in time, the information is spread by the evolution among the nodes of the reservoir in the same way for all the topologies, regardless any initial difference in the structure of interactions.

A highly interesting result is obtained by replacing the random Hamiltonian with a Haar random unitary evolution. In this framework the outcome state of the system slows strong internal correlations, but interestingly the resulting accuracy is the same as that for the different Hamiltonians at long evolution time. This result not only further confirms our hypothesis on the disappearing differences between the various evolutions, but also remarks how the long-time behavior of

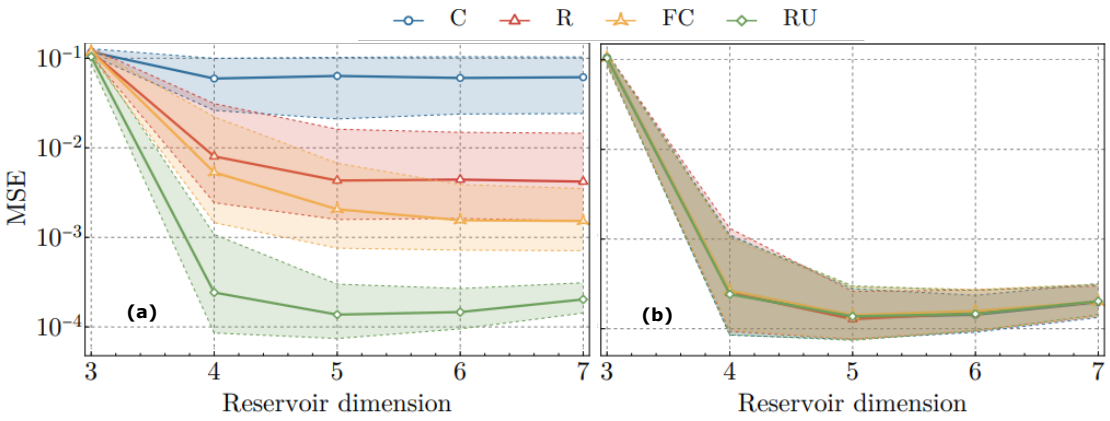


Figure 4.2: **Accuracy with varying reservoir dimension** - Reconstruction MSE vs number of qubits in the reservoir, with SL input coupling and different reservoir topologies. The evolution time is **(a)** $t = 0.25$ and **(b)** $t = 5$. Each point represents the median of the MSEs computed on an ensemble of 500 random Hamiltonians, with sampling statistics of 10^6 for both training and testing in each case. Training and testing sets each consisted of 50 random states. All the data of the simulation lie in the colored area, marking the boundaries of the first and third quartiles. In addition to the standard topologies C, R, FC defined in fig. 4.1, we also report the MSE obtained letting the whole input and reservoir system evolve unitarily with a Haar random unitary (RU).

a generic class of Hamiltonians approaches that given by Haar random unitaries, which are known to be optimal for observable estimation tasks [29, 131, 132]

4.4.2 Estimation accuracy and QIS

In this subsection we report the results on the time-dependence of the MSE for the coupling and injection schemes shown in fig. 4.1, relating it to the non-commutativity and local Holevo information in each system.

As shown in the previous section, for evolution times longer than the scrambling timescale, defined by the non-commutativity saturation, all the schemes are equivalent, achieving to the same accuracy. This data also shows that studying the

average non-commutativity is not sufficient to predict the reconstruction MSE, as it monotonically increases until saturation. On the hand, the Holevo information follows more consistently the MSE behavior, as it is crucial to quantify the amount of correlation between the input and the local output qubits.

In all topologies, the MSE eventually saturates to an asymptotic value of $(1.8 \pm 0.5) \times 10^{-4}$, the averaged χ to $(2.5 \pm 1.2) \times 10^{-3}$, and the OTOC to $C = 0.997 \pm 0.003$. We also note how increasing the number of links within the full system reduces the scrambling time, leading the system to faster convergence. Indeed, each link corresponds to additional interaction terms, allowing the information to flow faster through the reservoir.

In fig. 4.4 we report the time-dependence of the condition number κ for all different topologies. Recalling our discussion regarding the meaning of this quantity in section 1.2.1, the condition number quantifies the numerical stability of the solution provided by the QELM. In particular, it shows how stochastic errors in the training phase can propagate and are amplified in the testing stage, therefore being an effective performance estimate of the model.

A phenomenon characteristic of ML-coupled systems seen in fig. 4.3 is a transient regime where reconstruction accuracy is higher than its asymptotic value, which is also reflected in higher values of the average χ corresponding to the same evolution times. This is due to the direct couplings in ML systems allowing information to initially spread and become locally retrievable from all reservoir qubits, to then become partially lost to the correlations at longer times, as can be seen from fig. 4.5 **(d)-(e)-(f)**.

Furthermore, for both injection schemes, right before the saturation, C and R topologies achieve better accuracy than the FC one. This feature can be related to

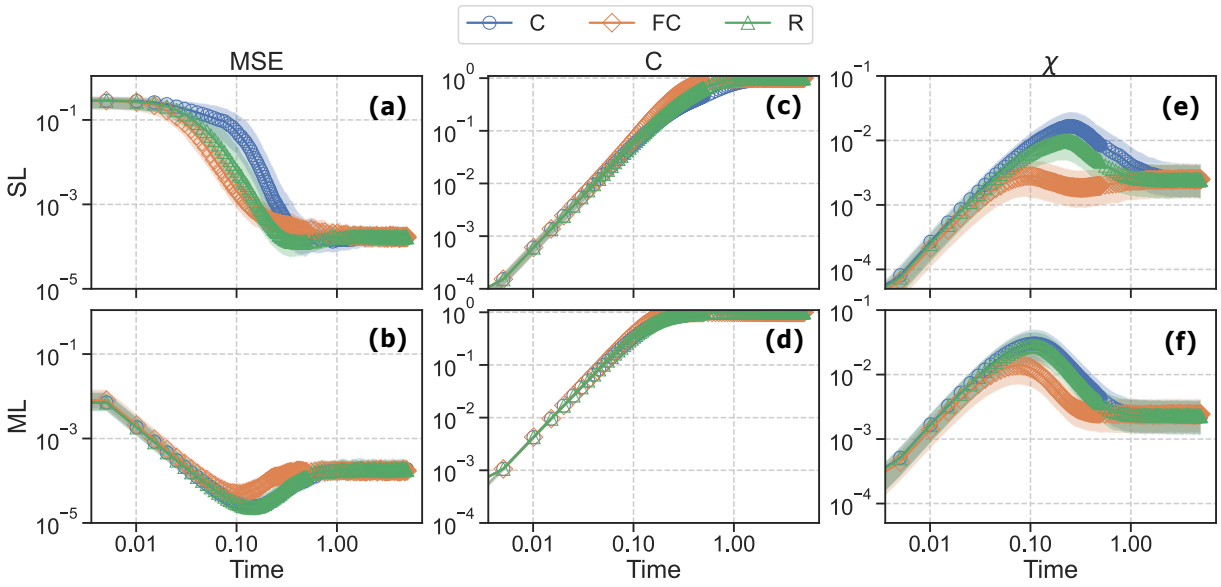


Figure 4.3: **Average Results** - Reconstruction MSE **(a)-(b)**, two-point correlation function C **(c)-(d)**, and local Holevo information χ **(e)-(f)**, as a function of time, for different interaction topologies, as a function of time in the interval $t \in [0, 5]$. In each plot we present the data corresponding to the three interaction topologies, FC (orange diamonds), C (blue circles), R (green triangles), outlined in fig. 4.1. The left realizations **(a)-(c)-(e)** refer to the SL scheme, while the right realizations **(b)-(d)-(f)** the ML scheme. In each case, we present the median results over 500 realizations of random Hamiltonians with the corresponding topology, with a reservoir of $N = 7$ qubits plus a single input qubit. The error bars show the first and third quartiles around the median. For the MSE we used sampling statistics of 10^6 in both training and test, both of which were performed with training and testing dataset comprised of 50 random states each.

the higher efficiency of the FC scheme to hide information in non-local correlations due to the higher number of interaction terms.

Remarkably, fig. 4.3 shows in some cases contradicting results when relating the average χ and the MSE. This is particularly evident with the FC topology in the SL coupling where, in the transient regime, the results show a lower averaged χ and at the same time a lower MSE. This observation is directly related to the average, which hides the real information distribution in the network. Indeed, the

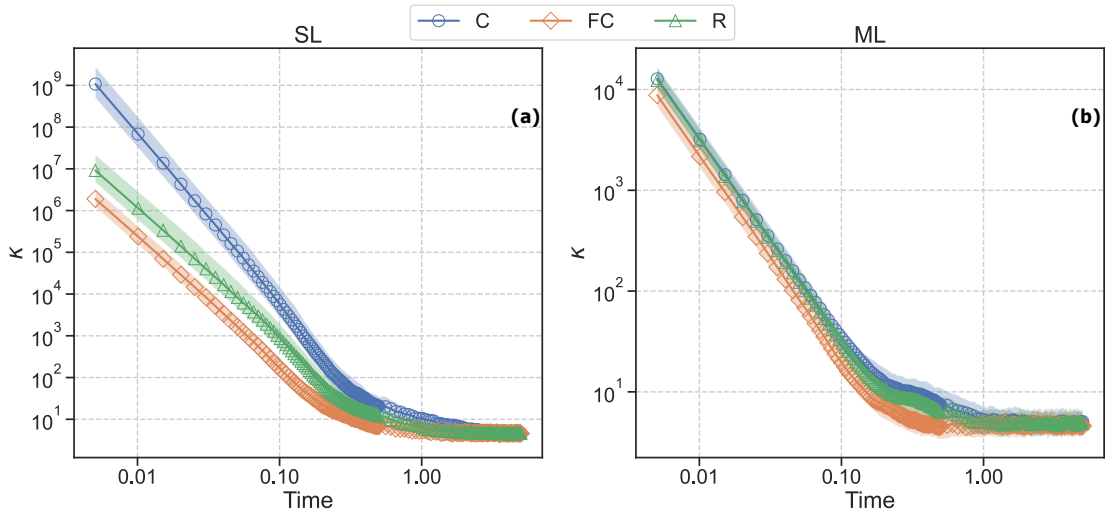


Figure 4.4: **Condition Number** - Condition number of $\mathbf{P}^{\text{train}}$ for the topologies of fig. 4.1 with SL (**a**) and ML (**b**) coupling schemes, for different evolution times $t \in [0, 5]$. In each topology and coupling scheme, each point gives to the condition number averaged over 500 random Hamiltonians, a reservoir with $N = 7$ qubits, and sampling statistics of 10^6 for both training and testing. The error bars represent the first and third quartiles. Training and testing were conducted with 50 states each. The long-time condition numbers are $\kappa = 4.6 \pm 1.1$ and $\kappa = 5.1 \pm 1.2$ for SL and ML schemes, respectively.

results in fig. 4.5 show a symmetric information distribution among the qubits in the reservoir for the FC topology; on the other hand, in the other two reservoirs the only informationally relevant nodes are those closer to the input state, showing a completely asymmetric distribution of the information. This reflects on a higher average χ for the C and R topologies, since FC is faster in spreading it both locally and non-locally, nonetheless almost all their outcome measurements are uncorrelated to the input state.

A different situation is observed in the ML-coupled systems where information spreads much more symmetrically, and thus the average χ is inversely correlated with the MSE, as expected. Beyond the scrambling time every single χ saturates to the same value of $\chi \simeq (2.5 \pm 1.2) \times 10^{-3}$, as expected from fig. 4.3.

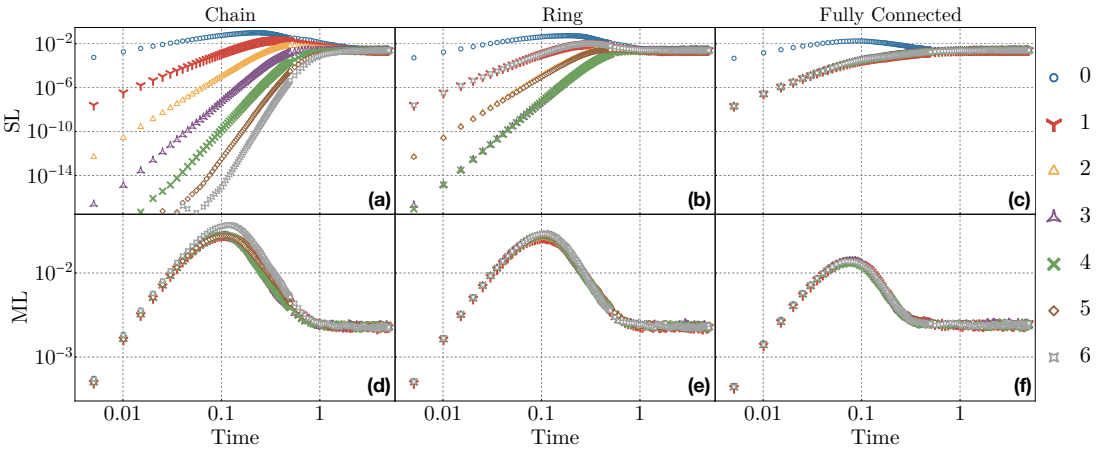


Figure 4.5: **Holevo information Distribution** - Distribution of the Holevo information among the nodes of the reservoir for SL (**a,b,c**) and ML (**d,e,f**) input coupling, and for chain (**a,d**), ring (**b,e**), and fully connected (**c,f**) interaction topologies. Each figure reports the Holevo χ corresponding to each of the $N = 7$ reservoir qubits, here labelled from 0 to 6, for different evolution times $t \in [0, 5]$. Each data point is the median of the dataset obtained over an ensemble of 500 random Hamiltonians.

Overall, all the reported data shows a robust convergence to a regime where state estimation is possible with the same performances granted by random unitary dynamics, without the need to fine-tune the interaction times. The initial differences between interaction topologies - due to different rates of information spreading - gradually converge to a uniform behaviour over longer timescales, once information has spread evenly across the network. Furthermore, if such fine-tuning is feasible, higher performances are possible for some types of interactions in the transient regime before the scrambling time.

4.5 Discussion

This chapter provides strong evidence that QELM-based accurate reconstruction [29, 51, 52] with local measurements is possible well beyond the scrambling time for several types of dynamics. Indeed, we showed the equivalence - in terms of accuracy - of such evolution with that obtained through Haar-random unitaries, which results in the maximum amount of distributed correlation across the system. These findings are interesting for both their experimental implications, and for the insights they provide into the relations between QELMs and QIS.

From an experimental perspective, our findings mean that for many types of dynamics there is no need to fine-tune the evolution time for the purpose of reconstructing properties of input states via QELMs. As long as the system is left to evolve long enough for the information to spread uniformly, accurate reconstruction is always possible. The evolution time must still remain fixed across different measurements, but its precise value does not need to be known to the experimenter.

At a more fundamental level, our findings offer a novel perspective into the nature of QIS and its relations to QELMs and state reconstruction tasks. Even though a scrambling system is considered to be one where information is hidden in the correlation and is locally irretrievable, our results highlight that - at least for relatively small systems - the opposite is true from a state estimation perspective: when information is allowed to spread uniformly throughout the reservoir qubits, the residual local information is still sufficient to retrieve arbitrary properties of input states.

In summary, our results pave the way for robust experimental state reconstruction schemes that do not rely on accurate knowledge of the underlying dynamic or

fine-tuning of the experimental apparatus, and furthermore suggest that the way information spreads locally for scrambling system might be a useful resource for quantum state estimation purposes.

In the next chapter, we will discuss another application of QELM, this time to a classical task. In particular, we will present a noise resilient QELM framework, which we are going to employ for the atmospheric retrieval of exoplanets, implementing the model on a real quantum device.

Part III

Quantum Extreme Learning for Astrophysical Data Analysis

Chapter 5

Exoplanetary Atmospheres Retrieval with a QELM

The analysis of exoplanetary atmospheres via absorption and emission spectra relies heavily on mathematical modeling [66–73]. Forward models simulate expected spectra based on atmospheric composition and physical conditions; in atmospheric retrievals, these are compared with observations to estimate parameters [74–80]. However, balancing computational cost with model accuracy remains challenging. Simple models allow for fast analysis but risk oversimplification and bias [133,134], while complex physical simulations capture intricate phenomena at the cost of significant computational resources [135–138].

Machine learning offers a solution, enabling rapid and accurate spectral characterization [139–142]. These algorithms significantly improve computational efficiency over classic retrieval methods and can be designed to output posterior distributions, synthetic spectra, or direct atmospheric parameters.

However, reliable ML algorithms depend on high-quality training datasets. As

most models struggle to extrapolate outside their training parameters, generating robust datasets that reproduce features observable by space- and ground-based instruments is crucial. This scalability is vital for exploiting low-resolution spectroscopy from missions like JWST [143] and Ariel [144]. As model complexity increases to interpret these rich datasets, ML tools must evolve to meet new challenges in precision and efficiency.

Inspired by the architecture used in [145], in this work, we construct a QELM to extract information from synthetic spectra generated by TauREx [146, 147], focusing on minimizing quantum resources while maintaining resilience to noise in near-term implementations. In this chapter, based on [130], we describe the first QELM applied to an astrophysical analysis problem. To test QELM's robustness, the algorithm is evaluated on datasets within the JWST spectral range, both with and without added artificial shot noise, to assess its response to realistic data imperfections. Additionally, robustness is proved by running the JWST dataset on the IBM Fez quantum hardware, confirming the performances and so the potential of QELM for NISQ applications.

5.1 Dataset: Generation, Pre-Processing and Encoding Pipeline

In this section, we present the structure of the training dataset, its pre-processing and the QELM workflow

5.1.1 Dataset Generation

In this chapter we will focus on a classical application of QELM models, in particular by training one capable to extract atmospheric features from a dataset of spectra of exoplanets. Specifically, we are going to process datasets of synthetic spectra generated by TauREx [146,147]. TauREx is an open-source Bayesian code for exoplanetary atmospheric modeling and retrieval which allows for the generation and retrieval of transmission, emission and reflection spectra over a wide range of compositions and physical parameters. It computes radiative transfer on exoplanetary atmospheres taking into account different chemical and physical conditions to produce the resulting spectrum. Its object-oriented structure allows the inclusion of many different atmospheric scenarios and it is widely used in the exoplanetary community.

The dataset employed in this study is the same described and processed in [140]. The atmospheric model relies on seven parameters distributed among four molecular volume mixing ratios (CH_4, CO_2, CO, H_2O) and three physical parameters representing planetary mass (M), radius (R), and equilibrium temperature (T). We assume constant vertical abundance profiles and an isothermal temperature-pressure profile. The parameter space is defined by specific lower and upper bounds, with each interval discretized into 10 values (see table 5.1). Spectra were generated in the $[0.3, 50]\,\mu$m range, divided into 515 spectral bins, by combining these parameter values. This yields a total pool of 10^7 spectra, from which we sampled a subset of 10^4 spectra for the training and testing stages. This baseline dataset is hereafter referred to as **TauREx**.

To evaluate the QELM in a realistic observational scenario, we generated a second dataset, denoted as JWST. We adapted the baseline TauREx spectra to match the instrumental resolution and spectral range ($[0.6, 2.8]\,\mu$m) of the HAT-

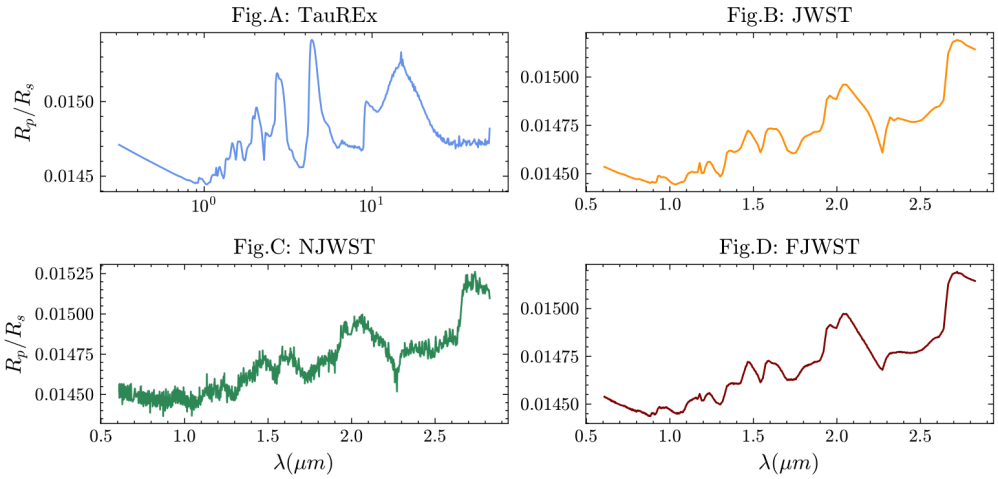


Figure 5.1: **Spectra** - Example of a simulated spectrum produced by TauREx (**Fig.A**), interpolated on JWST spectral range (**Fig.B**), with the addition of shot noise (**Fig.C**) and with the addition of a principal component filter taking 10 components in account (**Fig.D**). As can be seen in the spectra, adding the filter slightly modifies the shape of the spectrum.

P-18b observations presented in [148].Subsequently, we introduced artificial shot noise to the JWST spectra to create the NJWST (Noisy JWST) dataset. This process assumes a photon source temperature of $T_\star = 6460\,\mathrm{K}$ and follows the noise model detailed by [149]. Finally, to assess noise mitigation strategies, we generated the FJWST (Filtered JWST) dataset by applying the pre-processing steps described in the following subsection. Representative examples of all four datasets are displayed in fig. 5.1.

5.1.2 Pre-Processing

As already discussed in section 2.5.1 and section 2.5, the primary challenge in Quantum Machine Learning in general, lies in efficiently encoding high-dimensional data into a Hilbert space of limited dimension and limit the influence of decoherence noise into the quantum device.

Variable	Lower Bound	Upper Bound
CH_4	10^{-8}	10^{-1}
CO_2	10^{-8}	10^{-1}
CO	10^{-8}	10^{-1}
H_2O	10^{-8}	10^{-1}
M	$0.8M_J$	$2.0M_J$
R	$0.8R_J$	$1.5R_J$
T	$1000K$	$2000K$

Table 5.1: **Spectral parameters bounds** - The spectra considered in this work have been produced considering atmospheric parameters in the intervals above. Each interval has been divided in 10 values. The spectra are produced considering all the 10^7 possible combination of these values.

In this context, the use of a compressing strategy is not only suggested but mandatory. This is indeed a crucial element in efficiently encoding the most relevant features of a high-dimensional data, limiting both the information loss due to the compression strategy and the quantum noise.

Therefore, the common approach consists in reducing the dimension of data employing either convolutional neural networks, auto-encoders or **principal component analysis** (PCA) [48,150–152] to extract the most relevant and discernible features of a dataset. For instance, PCA is a linear transformation which consists in finding a different representation of the dataset, searching for a reference frame in

with augmented variance and separability.

However, in generative contexts like Quantum Generative Adversarial Networks (qGANs) [145], reducing the qubit count presents a significant trade-off. Since the generator's output resolution scales with the number of qubits, heavy compression can degrade the quality of generated data. To address this, previous works have proposed "task factorization"—training multiple generators in parallel to produce

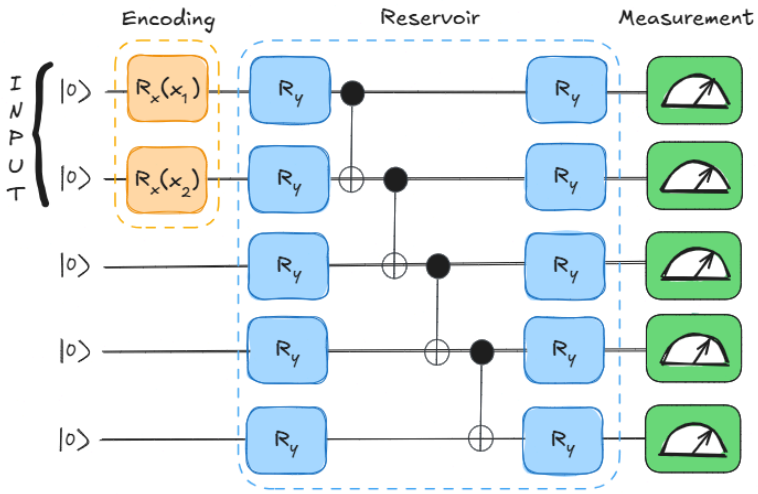


Figure 5.2: **Pictorial representation of a single reservoir** - The **encoding** of the classical data is performed by means of RX gates whose rotation angles depend on the principal component extracted by each patch; the **reservoir layer** consists of random RY rotations, a set of CNOT and other random RY rotations. Finally the outcomes are collected performing a projective measurement in the computational basis

smaller sub-sections of the data [145].

In this work we employ a hybrid approach which combines both the dimensionality reduction of the PCA and the task factorization. In particular, we divided the spectral range according to the major water bands in the IR while also taking into account the instrument pass-bands of JWST/NIRISS, NIRcam, MIRI and Hubble/WFC3, for a total of $P_{\mathbf{TauREx}}$=14 spectral bands, which from now on we are going to address as "*patches*".
The number of patches has been reduced to $P_{\mathbf{JWST}}$=8 when considering only the JWST/NIRISS spectral range. Beyond the need to factorize the problem, this operation has also been performed in order to normalize the patches one by one and amplify their spectral features.

As outlined in section 5.1.1, we also tested the model considering artificial shot

noise added to the **JWST** dataset. The shot noise is computed as in [149].

We assumed that measurement noise is dominated by quantum detection noise following Poisson statistics with equal mean and variance between ∼0.6 and 2.8 μm. We estimated the mean number of collected photoelectrons to be:

$$N_{\rm ph} = \frac{\pi\tau\Delta t}{hc}\left(\frac{R_\star D}{2d}\right)^2 \int_{\lambda_1}^{\lambda_2} B(\lambda, T_\star)\lambda d\lambda, \tag{5.1}$$

where λ_1 and λ_2 are the limiting wavelengths of the spectral bin, d is the distance of the star (we took here 270 pc by default), $R_\star = 1.46 R_{Sun}$, $T_\star = 6460K$, d=270pc and $\Delta t = 21340s$, respectively. The parameters $D = 16$, $\tau = 0.4$, and Δt are the diameter of the telescope, the performance of the system, and the integration time, respectively, whose values were fixed for JWST according to [149]. Since systematics may prevent us from reaching a 10 ppm precision with the JWST, we assumed a floor noise of 30 ppm throughout the whole spectral domain with a normal distribution [153] from 0.6 to $\sim 2.8\,\mu$m, wherever the noise was lower than 30 ppm.

For the FJWST dataset, we apply a PCA-based filtering step to mitigate noise before processing. We reconstruct the spectra using only the first 10 principal components. This threshold was selected by analyzing the cumulative explained variance, ensuring that the majority of signal information is retained while high-frequency noise is discarded. This filtering is critical, as standard normalization would otherwise amplify the noise, hiding genuine spectral features.

The final pre-processing step involves extracting the principal components (PCs) from each spectral patch to further reduce dimensionality. Normalization ensures these components are of comparable magnitude. Each patch is thus

mapped to a vector $\mathbf{x}^i$ of dimension PC. To preserve global context lost during patching, we append a supplementary vector containing the global maximum, minimum, and average value of the full spectrum. The final input matrix $\mathbf{X}$ is organized as follows:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^1 \\ \mathbf{x}^2 \\ \cdots \\ \mathbf{x}^{P+1} \end{bmatrix} \quad \text{with } P = \begin{cases} P_{\textbf{TauREx}}=14 \\ P_{\textbf{JWST}}=8 \end{cases} \tag{5.2}$$

5.1.3 Encoding Scheme and Reservoir Structure

To limit the number of qubits and the depth of the circuit, with a similar reasoning of [145, 154] we consider a factorized reservoir where the first sub-system encodes the global features of the spectra, while, the remaining P sub-reservoirs, are dedicated to processing the information from each individual spectral band.

Each sub-reservoir is initialized as a quantum register with $N_q \geq PC$ qubits in the state $|\mathbf{0}\rangle$. The processing pipeline for the i-th patch $\mathbf{x}^i$ consists of three distinct stages (visualized in fig. 5.2):

- **Encoding Layer**: The PC principal components of the patch are mapped to the qubit states via Angle Encoding. We apply a layer of R_x rotations, where the rotation angle of the k-th qubit corresponds to the k-th component of the input vector:
$$U^i_{Enc} = \bigotimes_{k=1}^{PC} R_x(x^i_k). \tag{5.3}$$
- **Reservoir Layer**: To induce entanglement and spread information, we employ a hardware-efficient ansatz. This consists of an initial layer of random

R_y rotations, followed by a linear chain of CNOT gates acting on nearest neighbors $(k, k+1)$, and capped by a final layer of random R_y rotations:

$$U_{Res} = \left(\bigotimes_{k=1}^{N_q} R_y(\beta_k) \right) \left(\prod_{k=1}^{N_q - 1} \mathrm{CNOT}_{k,k+1} \right) \left(\bigotimes_{k=1}^{N_q} R_y(\alpha_k) \right). \tag{5.4}$$

The parameters α_k, β_k are randomly initialized and remain fixed for all inputs. Crucially, this gate set was selected to match the native gate set of the target hardware, minimizing compilation overhead and circuit depth.

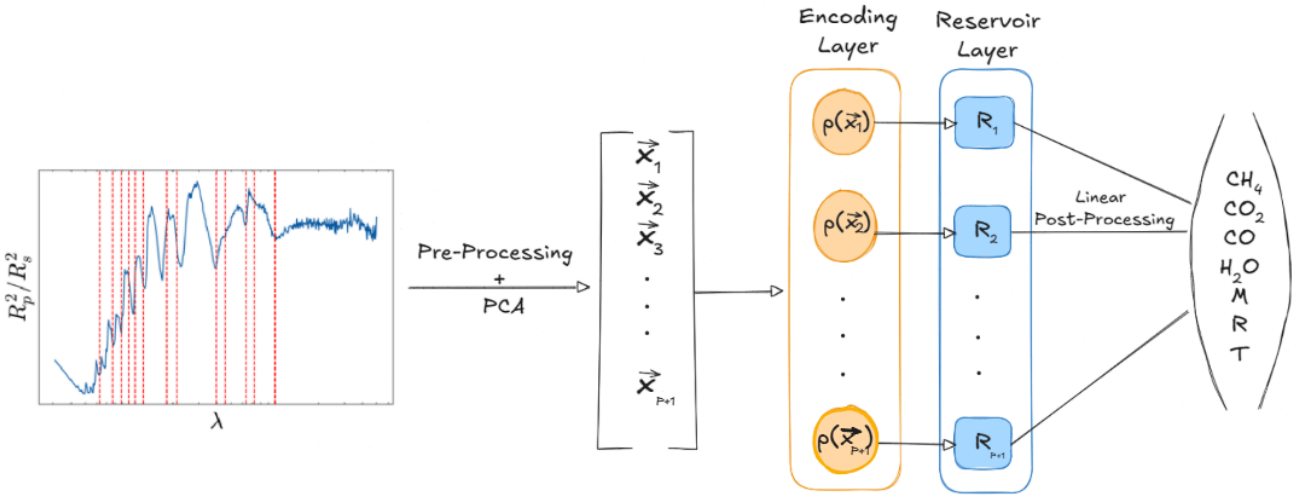


Figure 5.3: **Resuming scheme of the whole processing** - Spectra are patched according to [140] and pre-processed by the processing layer consisting of an interpolator (for JWST spectra), a normalization layer and a principal component analysis layer acting on each of the patches. The components are encoded in a reservoir as shown in Figure 5.2 and the outcome probabilities are than post-processed via the output layer to obtain the atmospheric parameters.

- **Measurement Layer**: The final stage consists of projective measurements in the computational basis. For each sub-reservoir, we obtain a probability

distribution $\mathbf{p}^i$ over the 2^{N_q} basis states:

$$p_m^i = |\langle m|U_{Res}U_{Enc}^i|\mathbf{0}\rangle|^2. \tag{5.5}$$

These local probability vectors are then concatenated into a global feature matrix $\mathbf{Q}$ (as shown below), which serves as the input for the classical linear readout training:

$$\mathbf{Q} = \left[\mathbf{p}^1, \mathbf{p}^2, \ldots, \mathbf{p}^{P+1}\right]^T. \tag{5.6}$$

Finally we train the output linear layer as already seen in the previous section in order to retrieve the atmospheric parameters listed in table 5.1:

$$\mathbf{W} = \mathbf{Y}\mathbf{Q}^+. \tag{5.7}$$

The pictorial representation of the model in its entirety is shown in fig. 5.3

We assess the model's generalization using the Mean Squared Relative Error (MSRE) calculated over the testing dataset (D_{test}):

$$\epsilon = \frac{1}{D_{test}} \sum_k^{D_{test}} \frac{(y_k^{test} - y_k^{pred})^2}{(y_k^{test})^2}. \tag{5.8}$$

Beyond this continuous error metric, we also define a binary accuracy score, $A(\epsilon^*)$, which counts the percentage of retrievals where the individual error ϵ_k falls below a tolerance threshold ϵ^*:

$$A(\epsilon^*) = \frac{100}{D_{\text{test}}} \sum_k^{D_{\text{test}}} \delta(\epsilon_k; \epsilon^*) \tag{5.9}$$

$$\delta(\epsilon_k, \epsilon^*) = \begin{cases} 1, & \text{if } \epsilon_k \leq \epsilon^*, \\ 0, & \text{otherwise.} \end{cases} \tag{5.10}$$

Having observed a linear relationship between accuracy and tolerance, we adopt a fixed threshold of $\epsilon^* = 5\%$ for all subsequent analyses.

The expressivity of the present QELM is constrained by the data encoding. Following the Fourier analysis of QELMs [43], the encoding determines the set of achievable Fourier frequencies, while the reservoir dynamics and the measurement determine the corresponding coefficients and accessible feature functions. In our case, each spectral patch is represented by M PCA coordinates and encoded through local Pauli rotations Equation (5.3). Therefore, for each coordinate x_k^i, the accessible frequencies are restricted to the differences of the eigenvalues of the Pauli generator. For the hardware experiment, $M = 5$ the Fourier support of each block is small and implies that the demonstrated model should not be interpreted as a generic highly expressive quantum model or as evidence of quantum advantage. The factorized architecture further implies that the complete feature map is a direct sum of patchwise feature maps,

$$\Phi_{\text{QELM}}(x) = \bigoplus_{i=0}^{N_p} \Phi_i(\mathbf{x}^i),$$

rather than a single global feature map over all spectral coordinates. We emphasize that the novelty of our approach is the structured use of this primitive for high-dimensional continuous inverse regression. This architecture provides a concrete route for applying QELMs to high-dimensional scientific data while respecting near-term hardware constraints. Increasing the input resolution does not require a single reservoir acting on all spectral bins; it can instead be handled by increasing the number of structured spectral blocks. The present implementation does not

create global quantum correlations between different patches. Cross-patch correlations are learned only through the final classical linear regression layer. Thus the framework should be understood as a modular, hardware-compatible QELM architecture for structured inverse problems, rather than as a claim of a new universal QELM model or of quantum advantage.

5.2 Results - Hardware Implementation

5.2.1 JWST Benchmark Simulations

To prepare for implementation on a physical quantum hardware, we first carried on a series of benchmarks to optimize the key hyperparameters, namely the number of principal components PCs and the size of the training dataset. These tests were performed on the **JWST** dataset, which represents a good compromise between the most complete data generated by **TauREx** and the noisy, observational **NJWST** data (see fig. 5.1 **(a-c)**).

These preliminary benchmarks were performed under idealized, noiseless conditions, assuming infinite sampling precision, to isolate the performance from statistical measurement noise. The simulations employed 8-qubit reservoirs, with results summarized in Figure 5.4.

Panel **(a)** displays MSRE [see Equation (5.8)] for each atmospheric parameter as a function of the extracted principal components PC, using a fixed training set of 8000 spectra. Panel (**b**) shows the convergence of the MSRE as a function of training set size, with PC fixed at 6. From this exploratory analysis, we determined that extracting a minimum of PC=5 principal components is sufficient for robust performance, though marginal improvements in retrieving H_2O

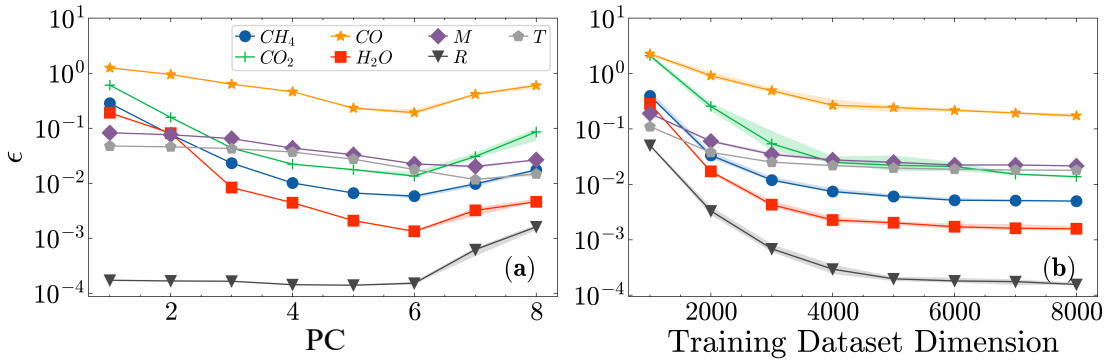


Figure 5.4: **Benchmark simulations using JWST dataset** - Benchmark simulations at infinite statistics to determine the minimal number of principal components to extract from each spectral patch (Fig. **a**) and the minimal number of training data. Both benchmarks were performed using reservoirs of N_q=8 qubits. The results in the upper plot are obtained considering a training dataset of 8000 spectra, while those in the lower plot considering a PC=6 principal components. For each point in the plot we made the training 10 times with different training datasets, while testing always on the same set

and CO_2 concentrations are observed with PC=6. Furthermore, panel **(b)** points to a minimal requirement of 4000 spectra for the algorithm to converge.

The retrieval accuracy for CO saturates at an MSRE of approximately 10^{-1}, regardless of the number of training samples or principal components retained. This behavior is consistent with results reported in spectral reconstruction using DC-GANs [140]. It arises because the CO molecule lacks prominent spectral features within the considered bandpass, making its abundance inherently more difficult to predict than that of the other three target molecules.

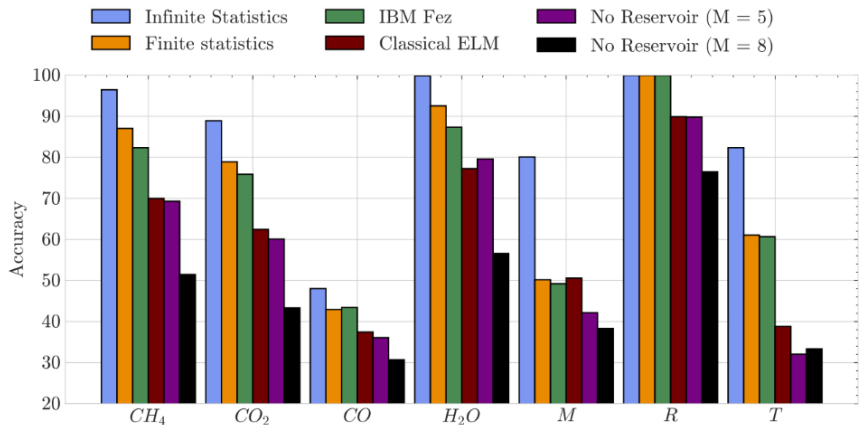
5.2.2 Hardware Implementation

We deployed the algorithm on IBM Fez, a superconducting quantum processor featuring 156 qubits. As detailed in section 5.1, processing the JWST dataset needs 9 distinct sub-reservoirs: 8 dedicated to the principal components of specific

spectral patches plus an additional circuit encoding the global features (mean, maximum, minimum). To parallelize execution on the QPU, we allocated nine subsets of five qubits each. Within each sub-reservoir, qubits were encoded with distinct principal components following the scheme in Figure 5.2.

Based on our benchmark simulations, we employed a dataset of 4080 spectra (split 75%/25% for training and testing) and extracted $PC = 5$ principal components per patch. Under the present experimental conditions, the model is classically simulable because it is composed of independent five-qubit reservoirs. Nonetheless, while performing the real experiment on QPU requires a computational time which only depends on the depth of the quantum circuit, from a simulational point of view the computational time strongly depends on the considered number of qubits per reservoir. However, from both perspectives this in general strongly depends on the number of qubits on the QPU for the quantum case and on the specifics of the classical computer from the simulation standpoint. We therefore do not interpret the hardware experiment as evidence of quantum advantage. The relevant scalability property of the proposed architecture is instead modularity: spectra depending on a higher number of parameters could generally require a finer patching, increasing the number of associated sub-reservoirs, while keeping each reservoir below the regime in which concentration effects make the measured observables input-insensitive. This patchwise scaling differs from the scaling of a single unstructured many-qubit reservoir. In a monolithic reservoir, concentration effects may cause observable readouts to approach an input-independent value exponentially with the number of qubits, requiring an exponentially increasing number of shots to resolve the input dependence.

Figure 5.5 presents the retrieval accuracies defined in eq. (5.9). We compare the results obtained doing infinite-statistics simulations (blue bars), finite sam-



Accuracy (%)	CH_4	CO_2	CO	H_2O	M	R	T
Infinite Stat	96.5	88.9	48.0	99.9	80.1	100.0	82.4
Finite Stat	87.1	78.9	42.9	92.5	50.2	100.0	61.1
IBM Fez	82.4	75.9	43.4	87.4	49.2	100.0	60.7
Classical ELM	70.0	62.5	37.5	77.3	50.6	89.9	38.8
No Reservoir (M = 5)	69.3	60.1	36.1	79.6	42.2	89.8	32.1
No Reservoir (M = 8)	51.5	43.3	30.7	56.6	38.3	76.5	33.3

Figure 5.5: **Hardware Results with JWST dataset** - Results obtained by processing the dataset using IBM Fez. The results have been obtained using reservoirs of 5 qubits each and M=5. In this context we considered a dataset of D=4080 spectra, with the 75% used for training and the rest for testing. Both the results obtained on hardware and the finite statistics simulation were obtained with 20000 shots.The algorithm was also tested considering a classical ELM (brown bars), and considering the QELM without any reservoir - using only the encoding layers with M=5 (purple bars) and M=8 (black bars) - as a baseline to show how the reservoir is actually improving the predictions.

pling statistics with 20,000 shots (orange bars) and experimental execution with the same statistics (green bars). As expected, accuracy slightly decreases at finite sampling statistics and again when implementing the model on hardware. These deviation arise from shot noise, readout errors and gate imperfections given by the device. However, the strong consistency across all three cases validates the robustness of our QELM architecture. Since the QELM receives features already extracted by classical patching, normalization and PCA, the retrieval accuracy cannot be attributed solely to the quantum reservoir. We therefore compare

against linear regression on the same PCA features and against a classical ELM with a matched random-feature dimension. These comparisons are necessary to identify the contribution of the quantum feature map relative to the information already present in the preprocessed classical representation. In Figure 5.5 (black and purple bars) we show the accuracies obtained performing the linear regression directly on the preprocessed data. The purple bars for $M = 5$, corresponds to the same number of PCA components used in the hardware QELM experiment, and the black bars for $M = 8$, which tests whether simply retaining more PCA components can reproduce the QELM performance. The results show that the classical preprocessing is indeed informative. In particular, the radius is already well captured by the preprocessed representation, consistently with the fact that its information is largely contained in global spectral quantities. However, the no-reservoir baselines do not reproduce the QELM accuracy for most retrieved parameters. For example, at $M = 5$ the direct linear baseline is substantially below the IBM Fez QELM results for CH_4, CO_2, CO, H_2O, M, R, and T. Increasing the number of PCA components to $M = 8$ does not close the gap and, in several cases, performs worse. Thus the observed performance cannot be explained simply as a consequence of applying linear regression to a low-dimensional PCA representation. For the implementation of the classical ELM we considered a similar setup as in the quantum case, considering 9 random neural networks of 32 neurons each. In this context, the input data are multiplied by a random matrix of dimension 32×5, followed by a sigmoid function. The choice of the number of neurons for each reservoir has been done accordingly to the dimension of the Hilbert space in the quantum case for a consistent comparison. As before, we collected the outputs of all the reservoirs in a unique training matrix, in order to tune the output layer parameters. Figure 5.5 shows that with this setup the resulting accuracies are much lower than in the quantum counterpart.

The figure also shows the predictive capabilities of the PCA without the reservoir are not powerful enough, with the model achieving only the 90% of accuracy for the radius which should be the easiest parameter to be retrieved, and decreasing accuracy also for the other parameters. Interestingly, increasing the number of principal components is detrimental for the model, as already anticipated by fig. 5.4.

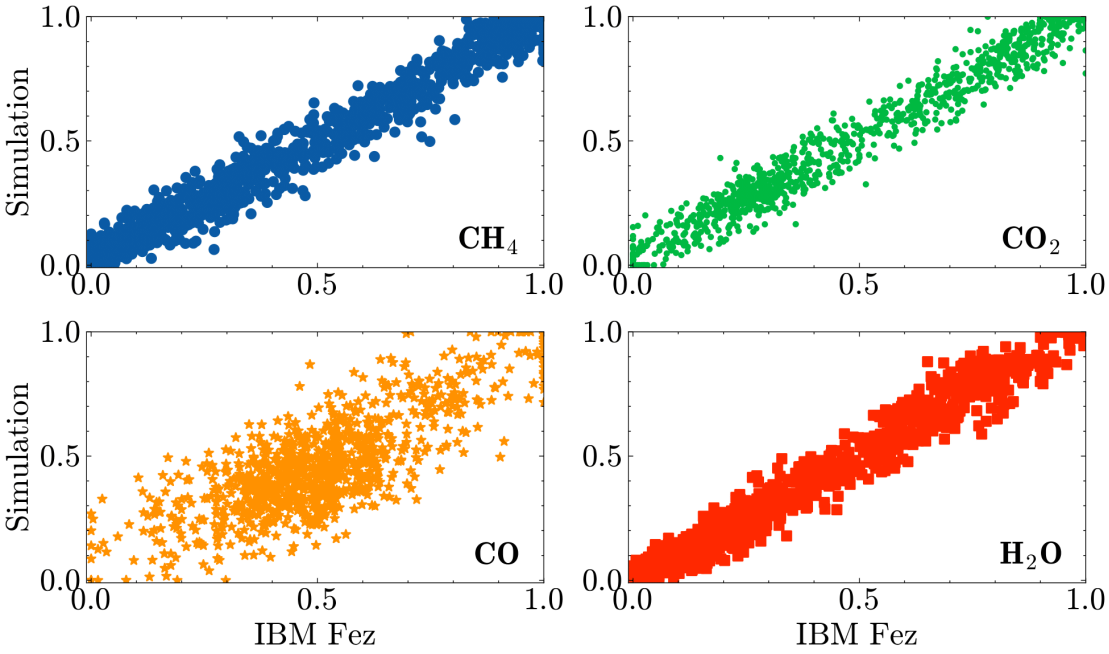


Figure 5.6: **Compared predictions using simulator and hardware with JWST dataset** - Parameters estimated by IBM Fez and by finite statistics simulations for each parameter.

The numerical results in Figure 5.5 show that, with the current dataset and set of principal components, the model is able—at infinite statistics—to retrieve the planetary radius for all samples with a relative error below 5%. Similarly, the concentrations of CH_4 and H_2O are successfully reconstructed for nearly the entire dataset within the same 5% threshold. The concentration of CO_2 is accurately retrieved for about 86% of the dataset, while the remaining atmospheric components

are reconstructed with progressively lower accuracy.

Most importantly, the discrepancy between the finite-statistics simulation and the experimental hardware results is negligible. The agreement between the finite-shot simulation and the IBM Fez, shown in Figure 5.6, results should be understood as readout-level adaptation to the effective noisy feature map. In the QELM protocol, the output weights are trained directly on the probability matrix produced by the reservoir. On hardware, this matrix is not the ideal noiseless matrix P_{ideal}, but an experimentally realized matrix $P_{\text{hw}} = P_{\text{ideal}} + \Delta P_{\text{hw}}$, where ΔP_{hw} includes the effects of finite sampling, readout errors, gate imperfections, crosstalk, and other device-specific noise sources. Since training and testing are performed using the same hardware-generated feature map, the final linear readout learns the regression map associated with the experimentally realized features. This mechanism has clear limitations. It can only compensate distortions that are sufficiently stable between training and testing and that do not erase the input-dependent information contained in the feature matrix. In the simple case where the dominant effect of hardware noise is an approximate attenuation, $P_{\text{hw}} \simeq \lambda P_{\text{ideal}}$, the Moore-Penrose pseudo-inverse can absorb this factor through $(\lambda P_{\text{ideal}})^{+} = \lambda^{-1} P_{\text{ideal}}^{+}$. This compensation is algebraic and occurs in the classical readout layer; it should not be interpreted as a quantum error-mitigation protocol. Moreover, it is not cost-free: the corresponding increase in the norm of the learned weights can amplify finite-shot fluctuations in the measured probabilities. Therefore, the hardware results demonstrate robustness under the stable noise conditions and shallow circuits used in the IBM Fez experiment.

5.3 Results - Simulative Benchmarks on Different Datasets

In this section, we present the analysis of the results obtained across the four datasets introduced in fig. 5.1. In particular, section 5.3.1 extends the benchmark analysis —previously described for the **JWST** case — to the remaining datasets to identify their respective optimal configurations. After having established the optimal parameters, in section 5.3.2 we report the final retrieval performance achieved by processing the four datasets in a simulative context.

5.3.1 Training and Feature Test

Following the methodology already developed for the **JWST** dataset, we extended the benchmark analysis to all datasets illustrated in fig. 5.1. As in the previous case, we determined the optimal number of principal components (PC) by fixing the training size to $D_{\text{train}} = 8000$ and evaluating performance on a test set of $D_{\text{test}} = 2000$ under idealized, noiseless conditions.

The results, displayed in fig. 5.7, show distinct behaviors across the datasets. In the **TauREx** and **JWST** cases (fig. 5.7 **(a)** and **(b)**) we observe that the MSRE in the retrieval of all the parameters, reaches a minimum $PC = 5{-}6$ components. Consistent with previous findings, CO retrieval shows a plateau above 0.1 regardless of the number of components, confirming the difficulty of assessing the information about its concentration.

On the other hand, for the two noisy and filtered datasets (**NJWST** and **FJWST**) increasing the number of principal components beyond $PC = 5$ does

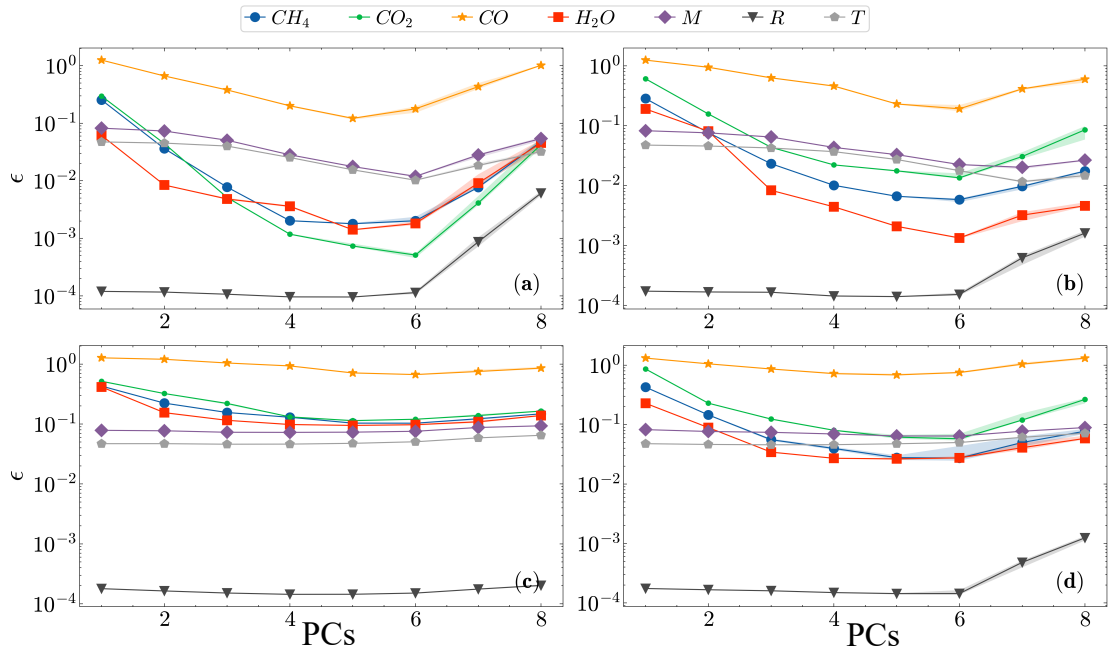


Figure 5.7: **Benchmark simulations on all datasets with variation of principal components** - Mean squared relative error of the algorithm as a function of the number of encoded principal components. The simulations were performed on a dataset of 10^4 spectra, with the 80% used for training and the remaining 20% for testing. Panel**(a)** shows the results for the **TauREx** dataset, **(b)** for the interpolated dataset within **JWST** spectral range, in **(c)** for the **NJWST** dataset and **(d)** for the **FJWST** dataset. Each point has been obtained by doing 10 different training instances, considering different training sets, while keeping fixed the testing set.

not grant any advantage in terms of accuracy. This suggest that shot noise and the filtering procedure both destroy most of the information captured through PCA.

Interestingly, the retrieval accuracy for planetary radius R, mass M, and temperature T remains independent of the number of principal components. This implies that information regarding mass and temperature is concentrated in the first principal component. Conversely, initial tests using only the minimum, maximum and average of the spectra to train the model, suggested that the robustness of the radius retrieval is primarily granted by these global features.

Finally, we assessed the training requirements by fixing PC=6 and varying D_{train} (see Figure 5.8). Here we observe that the **TauREx** dataset requires approximately $D_{\text{train}} = 5000$ samples for convergence, while the others reach their plateau earlier, typically between 3000 and 4000 samples.

5.3.2 Simulations Results

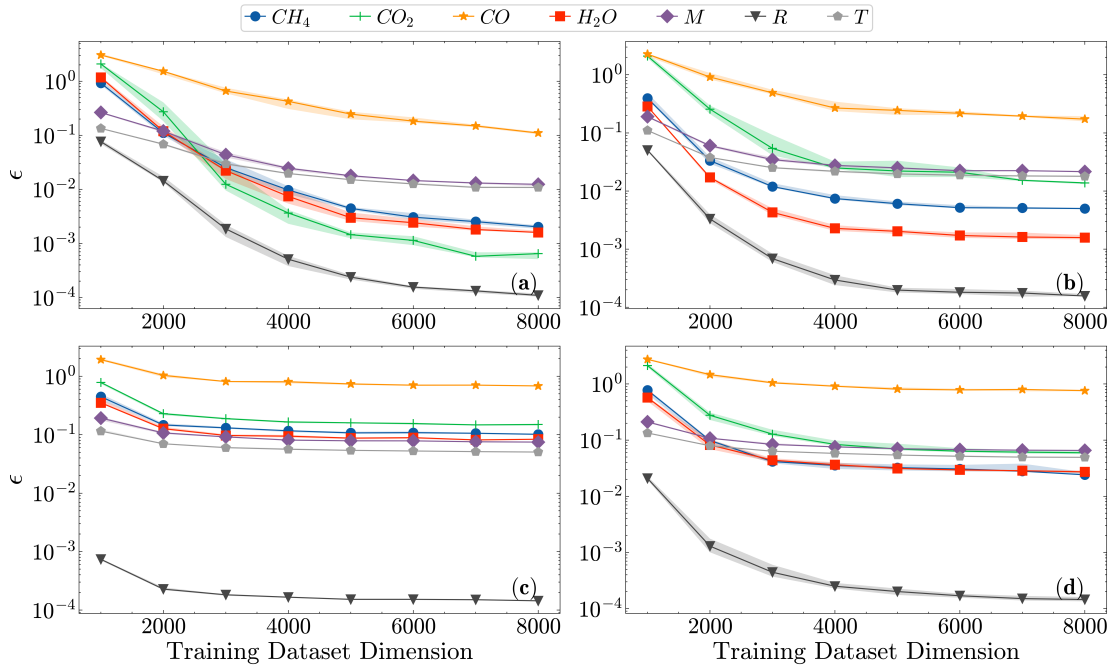


Figure 5.8: **Benchmark simulation with all datasets as a function of the number of training data** - Accuracy of the algorithm as a function of the number of training states and considering a test of 2000 spectra. In the panel **(a)** shows the performances using the **TauREx** dataset, in **(b)** shows those using the interpolated dataset in the spectral range of **JWST**, in **(c)** those with the dataset of **NJWST** and in **(d)** those with the dataset of **FJWST**. The number of components used for each patch is PC=6. Each point has been obtained by doing 10 different training instances, considering different training sets, while keeping fixed the testing set.

The results obtained Section 5.3.1 allow us to test the algorithm at its full potential.

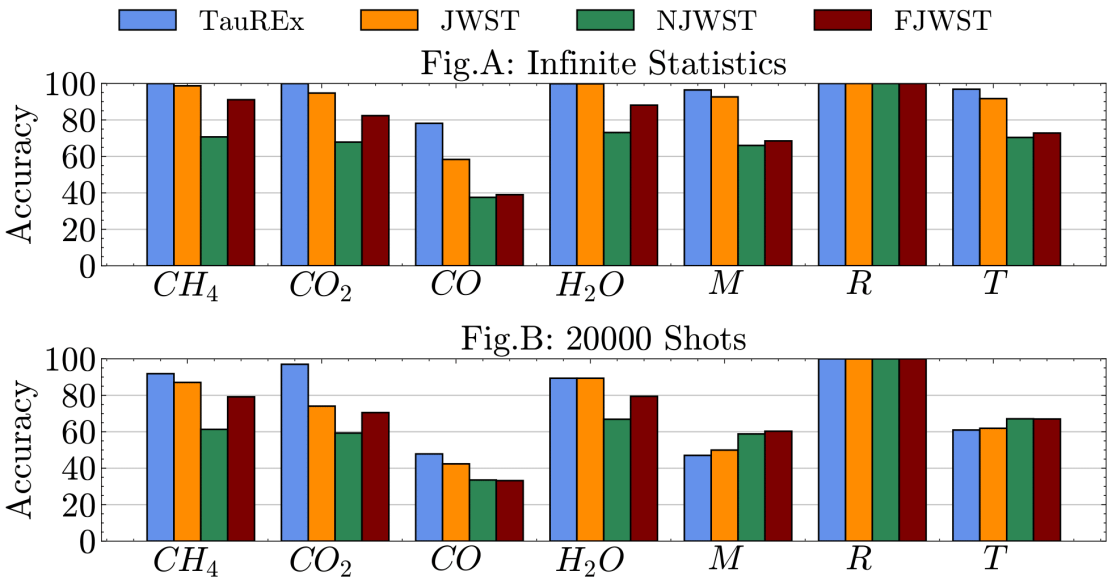


Figure 5.9: **Simulation results in the best conditions** - Accuracy of the QELM in retrieving each atmospheric parameter using the datasets shown in fig. 5.1. In this context we used a dataset of 10000 spectra split in $80-20\%$ respectively for training and testing and used $PC = 6$ principal components for each patch. In **Fig. A** are reported the results at infinite statistics, while in **Fig. B** the results with a statistics of 20000 shots.

Figure 5.9 presents the atmospheric retrieval results across the four dataset categories. For the **TauREx** dataset, the plots show that nearly all parameters are retrieved with $99-100\%$ accuracy in the limit of infinite statistics. The exceptions are planetary mass and temperature, which achieve a slightly lower success rate of 96%, and CO, which remains the most challenging parameter with an accuracy of approximately 80%. In the finite statistics regime (see Figure 5.9 **(B)**), we observe that while CH_4, CO_2, H_2O, and (R) maintain success rates exceeding 90%, the retrieval of mass and temperature worsens, highlighting a higher sensitivity to statistical noise.

For the **JWST** dataset, the QELM exhibits a comparable level of accuracy to that observed in the **TauREx** case, with only a slight degradation in the re-

trievability of CO_2 and CO. This suggest that part of their spectral features lies within the spectral range that was excluded.

Adding shot noise to this dataset and processing it without first filtering the noise using the principal component analysis results in a retrieval accuracy below the 70% for all parameters, except for the radius. The latter shows robustness against noise, as previously seen in Figure 5.7, confirming its information is encoded in the average values of the spectra and is therefore less affected by the presence of shot noise. These performances can be enhanced by applying a filtering procedure during pre-processing. Indeed, without such filtering, the normalization of each spectral patch amplifies not only the spectral features but also the shot noise, invalidating the principal components extracted before the encoding. As shown in Figure 5.9, the principal component filtering significantly improves retrieval performance, Althogh the success rate remains lower than those obtained for the **JWST** dataset, they still outperform the **NJWST**, reaching $70 - 80\%$ of accuracy at finite statistics for CH_4, CO_2 and H_2O. Naturally, all the results at finite statistics can be improved with a higher number of shots approaching the infinite statistics results.

Chapter 6

Supernovae Spectral Analysis with QELM

This final chapter reviews our recent results in applying Quantum Extreme Learning Machines (QELMs) to astrophysical data. Building upon the framework established in the preceding chapter, we aim to extend the model's capabilities by evaluating it on an entirely distinct dataset: X-ray emission spectra from supernova remnants.

Consistent with the methodology used for the exoplanetary analysis, this dataset is entirely synthetic. It was generated by sampling a comprehensive grid of physical and chemical parameters to simulate realistic emission spectra. To further optimize the QELM architecture, we introduce and benchmark several model enhancements in this chapter. These include testing different pre-processing strategies such as Partial Least Squared Regression (PLSR), alternative encoding strategies (such as Dense Angle and IQP encoding) and data re-uploading protocols.

6.1 The Dataset

The dataset utilized in this work consists of 170,000 synthetic X-ray spectra generated using the XSPEC (version 12.15.1) software package [155]. Specifically, we modeled the thermal emission from an optically thin plasma in non-equilibrium ionization (NEI) by adopting the VNEI model within XSPEC. The ionization fractions and subsequent spectral calculations were derived using ATOMDB 3.1.3 (http://www.atomdb.org/).

We explored different values of the plasma temperature, of the ionization parameter τ, defined as the integral over time of the plasma density calculated since its impulsive heating) and of the Si, S, Fe and Ni abundances (referred to the corresponding solar values reported in [156]). The values of each parameter are intrinsic in the bounds reported in table 6.1. We also incorporated the effects of photoelectric absorption by the interstellar medium using the TBABS model within XSPEC, exploring different values for the absorbing hydrogen column density (N_H). The sampling bounds for each parameter are detailed in the table below.

Finally, we passed our synthetic spectra through the instrumental response of an active X-ray telescope to simulate realistic observational conditions. We simulated an XMM-Newton observation utilizing the EPIC-pn camera [157] equipped with a medium filter. The required Auxiliary Response Files (ARF) and Redistribution Matrix Files (RMF) were generated using the ARFGEN and RMFGEN tasks of the Science Analysis System (SAS version 22.1), assuming a circular extraction region with an angular radius of 20 arcseconds.

Consistent with the methodology outlined in the previous chapter, all target parameters were normalized to the interval $[0, 1]$ prior to training.To optimize the

Variable	Units	Lower Bound	Upper Bound
N_H	$10^{22} \times cm^{-2}$	0.08	0.2
kT	keV	0.8	1.75
S	C_m/C_H	1.0	4.0
Si	C_m/C_H	1.0	4.0
Fe	C_m/C_H	1.0	4.0
Ni	C_m/C_H	1.0	4.0
τ	$10^{10} \times s/cm^3$	3.31	65.2

Table 6.1: **Spectral parameters bounds** - The spectra considered in this work have been produced considering physical and chemical parameters distributed non-uniformly in the intervals reported above. The complete dataset comprises 170000 spectra.

extraction of physical features and facilitate the factorized quantum reservoir architecture, the simulated spectra were once again divided into distinct sub-bands, or "patches." Whereas the patch boundaries in the exoplanetary analysis were dictated by major water absorption bands in the infrared, the boundaries for the supernova remnants were specifically engineered to isolate key physical and chemical features in the X-ray regime. The spectra were segmented into the following distinct spectral bands: It is important to note a specific physical constraint em-

Table 6.2: X-Ray Spectral Bands and Key Diagnostics

Energy Band (keV)	**Dominant Feature**	**Key Physical Parameter**
0.5 – 1.5	Soft X-ray emission (ISM absorption)	Hydrogen column density (N_H)
0.7 – 1.2	Primary spectral features	Iron (Fe) abundance
1.6 – 2.1	Emission lines (Si & S complex)	Si and (S) abundances
4.4 – 6.2	X-ray continuum	Plasma temperature (kT)
6.3 – 6.9	Iron K-shell complex	Ionization parameter (τ), Fe abundance

bedded within the simulated dataset: the spectra were generated such that the abundances of Fe and Ni are perfectly coupled (i.e., they share the same abundance values relative to solar), as are the abundances of S and Si. Consequently, while the selected spectral bands do not contain distinct, isolated emission fea-

tures for Nickel, the QELM can implicitly infer its concentration by mapping the correlated Iron features. In the following sections, we will demonstrate that the model successfully capitalizes on these correlations to retrieve the correct values for all coupled elements.

6.2 Upgrading the QELM Pipeline

While the baseline model is the same as the previous chapter, in this work we further developed it in order to enhance the performances and solve some issues intrinsic to this particular dataset.

In this section we are going to review all the upgrades performed on the architecture described in section 5.1.3, describing alternative pre-processing strategies, encodings.

6.2.1 Data Compression

While the key data pre-processing and normalization techniques remain consistent with our previous work on exoplanetary atmospheres [140], this chapter introduces a comparative benchmark of data compression techniques. In particular, we evaluate the performance of PLSR against the previously utilized Principal Component Analysis PCA.

While PCA is an unsupervised technique which consists in applying orthogonal transformation to the input data matrix, retaining the components with highest variance, PLSR is a supervised dimensionality reduction method. Specifically, rather than simply focusing on the input matrix alone, PLSR seeks to maximize

the covariance between the input features and the target labels - namely each spectral patch with our spectral parameters. This ensures that the extracted latent variables retain the most relevant information for the specific predictive task.

Mathematically, PCA performs a standardization of the input data matrix $\mathbf{X}$, centering the mean and standardizing the variance to 1, defining the matrix $\tilde{\mathbf{X}}$ and computes its Singular Value Decomposition:

$$\tilde{\mathbf{X}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \tag{6.1}$$

where we remind $\mathbf{U}, \mathbf{V}$ being two orthogonal matrices containing the left and right singular vectors of $\tilde{\mathbf{X}}$ and $\boldsymbol{\Sigma}$ being a square matrix containing its the singular values in the diagonal. The columns of $\mathbf{V}$ represent the principal axes of the data, while the singular values in $\boldsymbol{\Sigma}$ are proportional to the variance captured along each corresponding axis. The singular vectors are sorted in descending order of their corresponding singular values, meaning the first column captures the maximum possible variance. To achieve dimensionality reduction from D to M dimensions (where $M < D$), the data is projected onto the subspace spanned by the most significant principal components. This compressed representation, $\mathbf{X}_M \in \mathbb{R}^{N \times M}$, is obtained by multiplying the centered data by $\mathbf{V}_M$, a matrix containing only the first M columns of $\mathbf{V}$:

$$\mathbf{X}_M = \tilde{\mathbf{X}}\mathbf{V}_M$$

Unlike the single-step projection of PCA, PLSR compression is generally performed iteratively. The algorithm decomposes both the input matrix $\mathbf{X}$ and the target matrix $\mathbf{Y}$ by finding projection weight matrices $\mathbf{S}$ and $\mathbf{C}$ that maximize the covariance between their respective score matrices. This optimization problem can be formulated as:

$$\mathbf{S}, \mathbf{C} = \arg\max(\mathbf{S}^T\mathbf{X}^T\mathbf{Y}\mathbf{C}) \tag{6.2}$$

subject to normalization constraints on the weight vectors. These weights are subsequently used to project $\mathbf{X}$ and $\mathbf{Y}$ onto the latent score matrices $\mathbf{T}$ and $\mathbf{U}$:

$$\mathbf{T} = \mathbf{XS}, \quad \mathbf{U} = \mathbf{YC}$$

The first M columns of the score matrix $\mathbf{T}$ represent the PLSR-compressed data.

6.2.2 Encoding Strategies

As previously discussed in section 2.5.2, the quantum feature map—the mechanism by which classical data is embedded into a quantum state—is a critical determinant of both the model's expressivity and its practical feasibility on near-term hardware. In our prior investigation of exoplanetary atmospheres, we employed

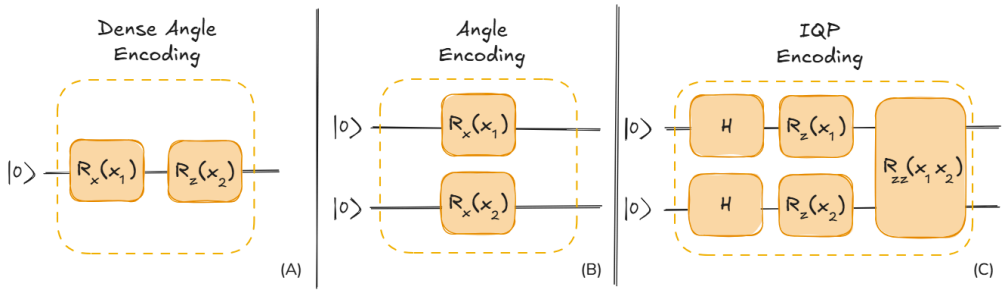


Figure 6.1: **Encoding Strategies** - Schematic representation of the encoding strategies evaluated to enhance the baseline model from chapter 5. (A) Dense Angle Encoding embeds two features per qubit, exploring a larger region of the Bloch sphere and halving the required qubit count compared to standard Angle Encoding. (B) Standard Angle Encoding. (C) IQP Encoding utilizes diagonal entangling gates to natively embed higher-order correlations into the initial quantum state.

a standard angle encoding strategy. Despite its relative simplicity, this method yielded highly accurate parameter retrievals. In this chapter, we improve our architectural framework by benchmarking this baseline against dense angle encod-

ing and IQP embedding (illustrated in fig. 6.1).Dense angle encoding maximizes resource utility by utilizing two distinct rotational gates to embed two classical features into the relative phase and amplitude of a single qubit. While this adds a marginal increase to the initial circuit depth, it reduces the total qubit requirement to $N_q/2$ compared to standard angle encoding. Conversely, IQP encoding introduces entangling operations directly into the feature map, potentially enabling the quantum state to natively capture higher-order, non-linear correlations present within the input data. Finally, we evaluate the impact of data re-uploading

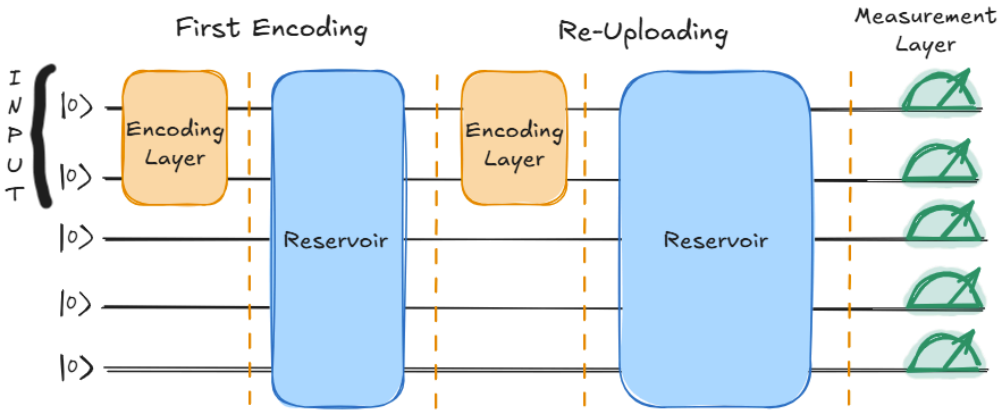


Figure 6.2: **Data Re-Uploading** - The classical input data is iteratively reinjected into the quantum system, alternating between encoding layers and reservoir processing layers. This iterative process increases the representational capacity of the model by exploring a more complex subspace of the multi-qubit Hilbert space.

(fig. 6.2) across all three encoding schemes. This technique iteratively injects the classical input into the quantum circuit, interleaving encoding blocks with the reservoir's processing layers. In an idealized, noiseless simulation, this systematic repetition increases the representational capacity and non-linearity of the quantum model. However, it also causes the circuit depth to scale linearly with the number of re-uploading layers. In an experimental context on physical hardware, this increased depth can quickly become detrimental, as it introduces a higher ac-

cumulation of gate errors and decoherence noise.Therefore, a primary objective of this benchmarking analysis is to identify the optimal operational trade-off: finding an encoding scheme and re-uploading depth that maximizes predictive accuracy while remaining sufficiently shallow to mitigate the destructive effects of quantum noise.

6.3 Results and Discussion

This section presents the empirical validation of the enhanced QELM architecture. To systematically evaluate the model's capabilities and the impact of our proposed upgrades, the discussion is structured into three sections: at first we establish the baseline hardware requirements by determining the minimal number of qubits and latent components—extracted via both PCA and PLSR—necessary to carry out the dataset analysis; next, we evaluate the predictive performance across the three quantum encoding strategies illustrated in fig. 6.1. This stage simultaneously tests the efficacy of data re-uploading at various circuit depths and directly compares the feature extraction capabilities of PCA versus PLSR; finally, we subject the optimized model to a strict generalization test using a newly generated validation dataset. While the physical parameters of this dataset remain confined within the bounds established in table 6.1, they are drawn from a completely different statistical distribution. This critical analysis proves whether the algorithm has genuinely learned the underlying physics of the X-ray emission models, or if it has merely overfitted and memorized the parameter grid of the primary training set.

In this context, in addiction to the principal components, instead of the global average of the spectrum, we encoded the **hardness ratio** and the average of the

soft band of the spectrum. These are defined in the following way:

$$HR = \frac{H - S}{H + S} \tag{6.3}$$

where H is the hard X band of the spectrum $(0.5 - 1.5)keV$, while S is the soft band of the spectrum $(1.5 - 7.5)keV$.

In order quantify the retrieval accuracy across all experiments, we employ the Normalized Mean Squared Error (NMSE). This metric evaluates the deviation between the ground-truth test parameters and the algorithm's predictions, with both sets rescaled to the $[0, 1]$ interval to ensure equal weighting across disparate physical units:

$$\text{NMSE}(\bar{\mathbf{Y}}_{test}, \bar{\mathbf{Y}}_{pred}) = \frac{1}{N} \sum_{i=1}^{N} (\bar{y}_i^{test} - \bar{y}_i^{pred})^2$$

where N represents the total number of test samples, and $\bar{y}$ denotes the normalized parameter values.

All the results were obtained considering a total of $N_p = 5$ patches, for a total of 6 reservoirs, 5 for the spectral bands and one encoding the hardness ratio.

6.3.1 Resource Benchmarks

Consistent with our previous methodology, adapting the QELM to an entirely new dataset requires a preliminary evaluation of its fundamental resource requirements. Before performing the full analysis, we benchmarked the model's performance by

varying the number of encoded latent components and the total number of qubits. As a first step, we fixed the reservoir size at $N_q = 10$ qubits and varied the number of encoded components. To directly compare data compression techniques,

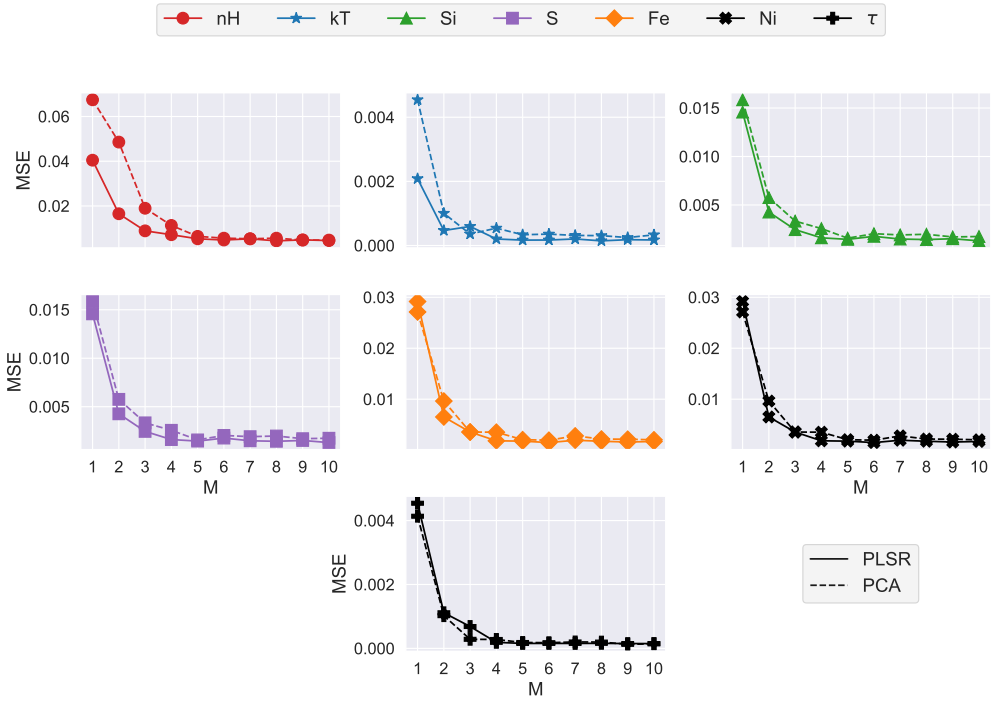


Figure 6.3: **Benchmark for the number of principal components** - Normalized Mean Squared Error (NMSE) evaluated on the test set as a function of the number of extracted components used for data compression. Simulations were conducted at infinite statistics using 10-qubit reservoirs. Retrieval accuracy reaches a stable plateau after encoding 6 components per sub-reservoir.

these components were extracted using both PCA and PLSR. For this baseline assessment, we employed the standard Angle Encoding. All benchmark simulations in this phase were performed in the infinite statistics limit to isolate the model's theoretical capacity from finite-sampling shot noise.

Figure 6.3 illustrates the NMSE for each target parameter as a function of the number of extracted components, M. The retrieval accuracy plateaus for all parameters at $M = 5$ to 6 components. While the asymptotic performance of PLSR is only marginally superior to PCA, PLSR achieves better performances, especially for N_H and kT, already with $M = 1$, approaching the same plateau as the PCA

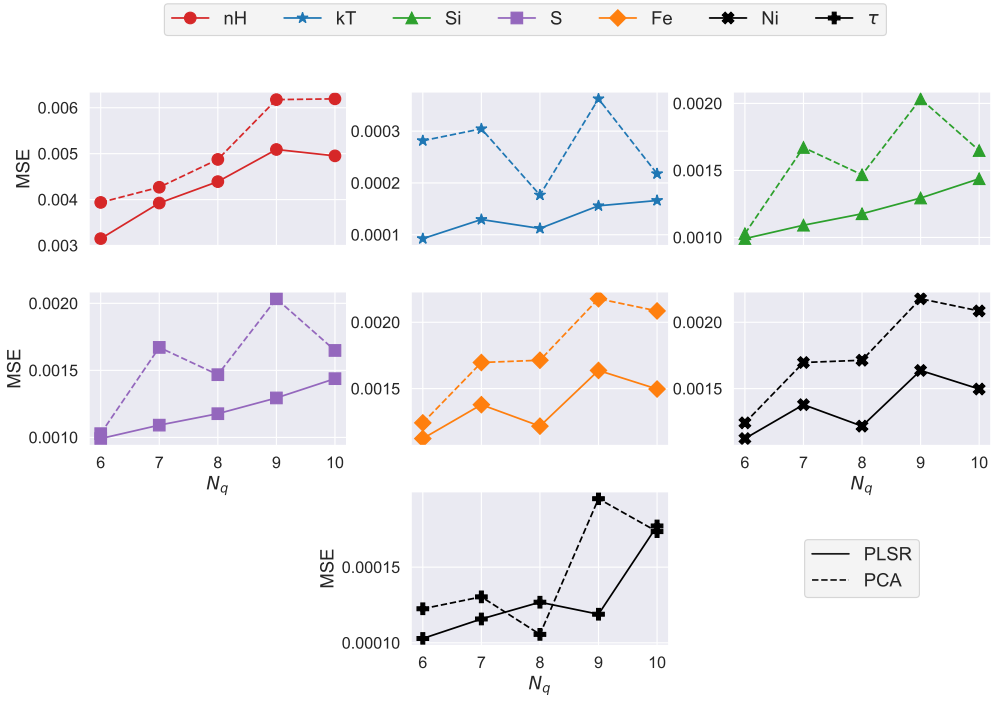


Figure 6.4: **Benchmark for the Number of Qubits with Angle Encoding** - Infinite-statistics simulations evaluating the NMSE as a function of reservoir qubit count (N_q). The number of encoded components is fixed at $M = 6$, utilizing standard Angle Encoding. The results indicate that accuracy is largely independent of N_q once the minimum component threshold is met, with minor variations stemming from distinct random reservoir initializations.

with the increase of encoded latent components.

In fig. 6.4 we fix the number of extracted principal components to $M = 6$ and the evaluate the impact of changing the number of qubits N_q on the performance of the model. The figure shows only minor fluctuations, which can be addressed to the intrinsic random nature of the reservoir's initial parameters, rather than an increasing or decreasing representational capacity.

Introducing Dense Angle Encoding into these benchmarks yields substantial resource advantages. A fundamental hardware constraint of standard Angle Encoding is that the number of qubits must be strictly greater than or equal to the

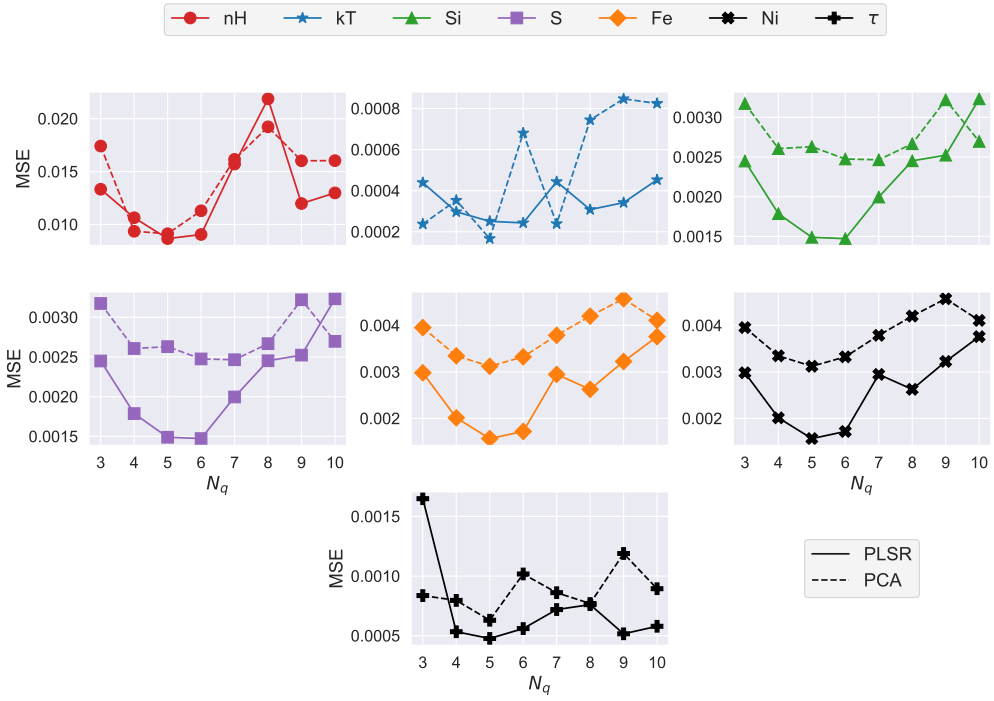


Figure 6.5: **Benchmark for the Number of Qubits with Dense Angle Encoding** - Infinite-statistics simulations evaluating the NMSE as a function of reservoir qubit count, utilizing Dense Angle Encoding to embed $M = 6$ principal components. The data reveals a weak dependence on qubit count, demonstrating that Dense Angle Encoding achieves comparable accuracy to standard Angle Encoding (fig. 6.4) using exactly half the physical qubits.

number of encoded features $N_q \geq M$. Because Dense Angle Encoding embeds two features per qubit, it effectively relaxes this condition to $N_q = M/2$.

Figure 6.5 presents the results of this optimized approach, fixing the latent components at $M = 6$ while employing Dense Angle Encoding. Notably, the model achieves the same order of magnitude in accuracy as the previous baseline using only $N_q = 3$ qubits. While the overall accuracy exhibits a weak dependence on adding additional qubits, the retrieval of the critical ionization parameter (τ) shows a distinct improvement when the reservoir size is marginally increased to $N_q = 4$.

6.3.2 Benchmarking the Encoding Strategies

In the previous section, we proved how Dense Angle Encoding provides an advantage in reducing the number of qubits required to effectively solve the regression task. Here, we carry on our study to evaluate the impact of data re-uploading when combined with the three distinct encoding strategies introduced in fig. 6.1.

Figures fig. 6.6, fig. 6.7, and fig. 6.8 show the model's NMSE as a function of the number of data re-uploads for Angle, Dense, and IQP encoding, respectively. These simulations were performed with reservoirs of $N_q = 6$ qubits and $M = 6$ extracted components. For both standard Angle and Dense Angle encoding, iterative re-uploading yields a substantial improvement in predictive performance, with the NMSE decreasing by nearly a full order of magnitude before plateauing. On the other hand, this same trend is not followed in the IQP encoding, where data re-uploading decreases the model's accuracy. This phenomenon is likely caused by the R_{ZZ} entangling gates inherent to the IQP feature map which, rather than improving the expressivity of the model, hides the information inside non-measured correlations.

Consistently with what we observed in the previous section, PLSR marginally outperforms PCA across all re-uploading depth, though advantage is not significant enough to prefer a compression technique to the other. Finally, while re-uploading is a powerful tool for increasing the expressivity, its utility depends on the employed quantum feature map used to encode the classical data and, more importantly, come with a intrinsic trade-off of increasing the circuit depth.

To better frame the differences between these methods, fig. 6.9 provides a direct performance comparison across the three quantum feature map. While Dense Angle Encoding obtains the greatest benefits from data re-uploading, achieving

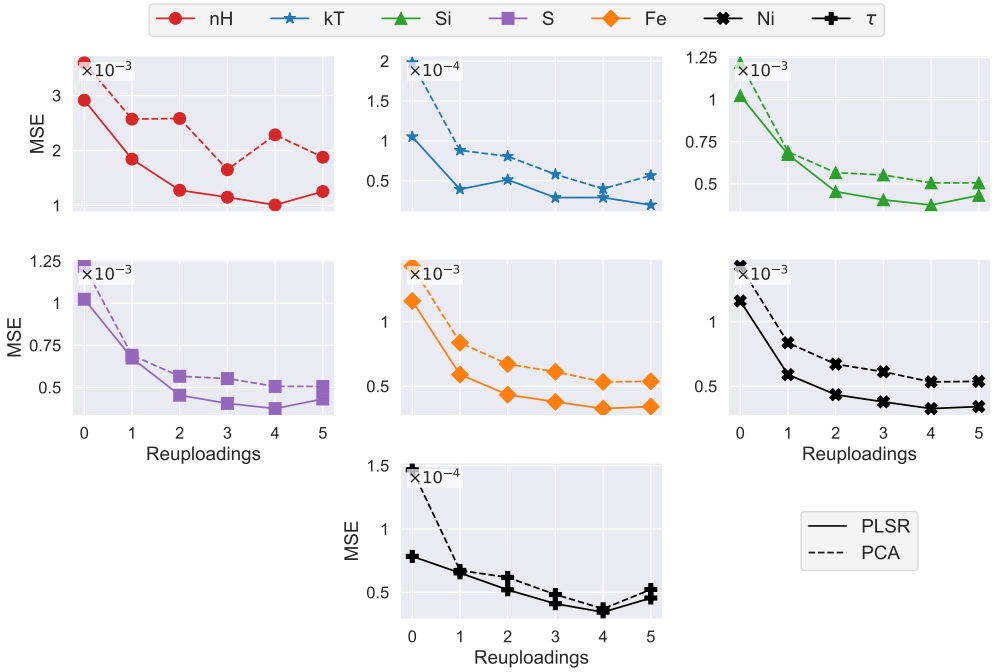


Figure 6.6: **Data Re-Uploading with Angle Encoding** - This figure shows the NMSE of the model as a function of the number of input data re-uploads using a standard Angle Encoding strategy. Pre-processing was performed using both PLSR and PCA. Simulations were conducted at infinite statistics, utilizing $M = 6$ components and reservoirs of $N_q = 6$ qubits. The accuracy monotonically increases (NMSE decreases) with additional re-uploads until reaching a plateau.

the highest accuracy among all the methods with 4 uploads, IQP is able to reach almost the same performance without the need of any prior re-uploading. This presents a critical optimization opportunity: while Dense Encoding combined with re-uploading is the best performing feature map, it requires 45 layers (fig. 6.10) to be implemented (35 considering only $N_q = 4$). On the other hand, a single-upload IQP circuit only requires 10 layer to be implemented, effectively presenting an advantage in case of a real hardware implementation. This also points out that the predictive accuracy is not strictly bound to the raw circuit depth or the number of reservoir layers, but rather by the mathematical structure of the implemented encoding strategy.

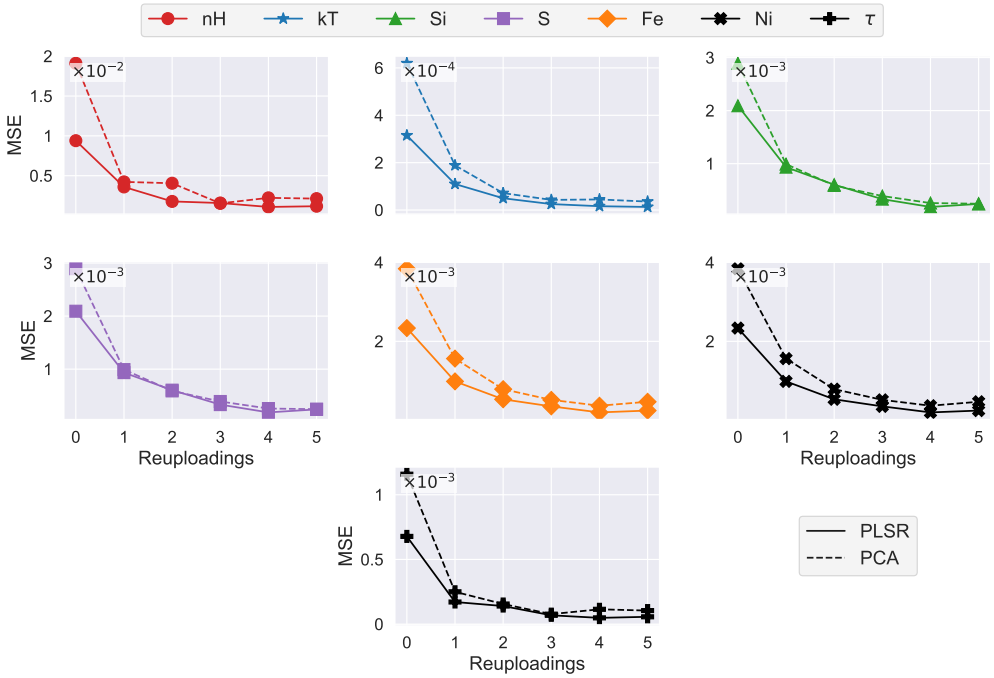


Figure 6.7: **Data Re-Uploading with Dense Encoding** - This figure displays the NMSE of the model as the number of re-uploads increases under a Dense Angle Encoding strategy. Simulations were performed at infinite statistics with $M = 6$ and $N_q = 6$. Similar to the trend in fig. 6.6, the error monotonically decreases before plateauing.

6.3.3 Testing with Parameters from Unknown Distribution

While our results are already very promising, we need to test whether the model is actually understanding the physics of supernova remnants, thus fitting the actual relation between the input parameter and the outcome spectra, or if it is memorizing the distribution connecting the training and testing parameters to the dataset.

In order to infer this property, we generated two other different datasets, whose parameter values are always confined within the bounds reported in table 6.1, but

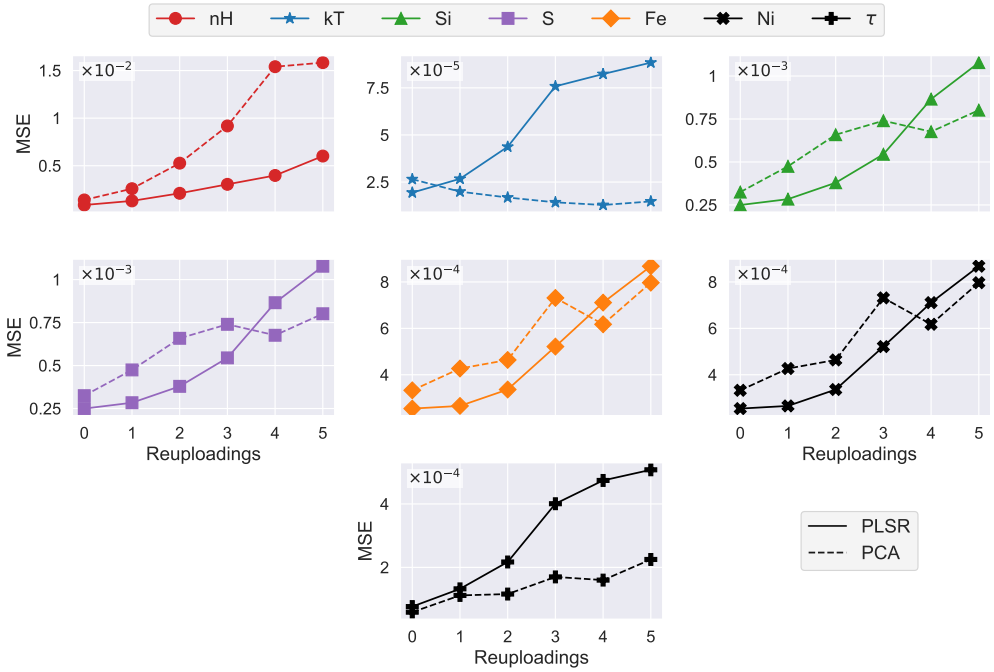


Figure 6.8: **Data Re-Uploading with IQP Encoding** - This figure illustrates the NMSE of the model under an IQP Encoding strategy. In stark contrast to the previous methods, the error strictly grows as the number of re-uploads increases, indicating that a single data injection is optimal for this highly entangled feature map.

with different distribution. One of the two datasets is used as a testing set; while, the second dataset is used to mix it inside the current training set in order to help the model to generalize correctly. The parameters of the two distributions are sampled in such a way that their values are close to those in the first dataset. The idea is to make the parameters' distribution as smooth as possible.

Figure 6.11 shows the NMSE as a function of the mixing of the training dataset between data from different distribution, obtained by a model composed of reservoirs of $N_q = 6$ with $M = 6$ and employing the IQP encoding with single injection. The NMSE is computed on the retrieved parameters from the data of a third distribution. The results show that without this initial mixing in the training phase,

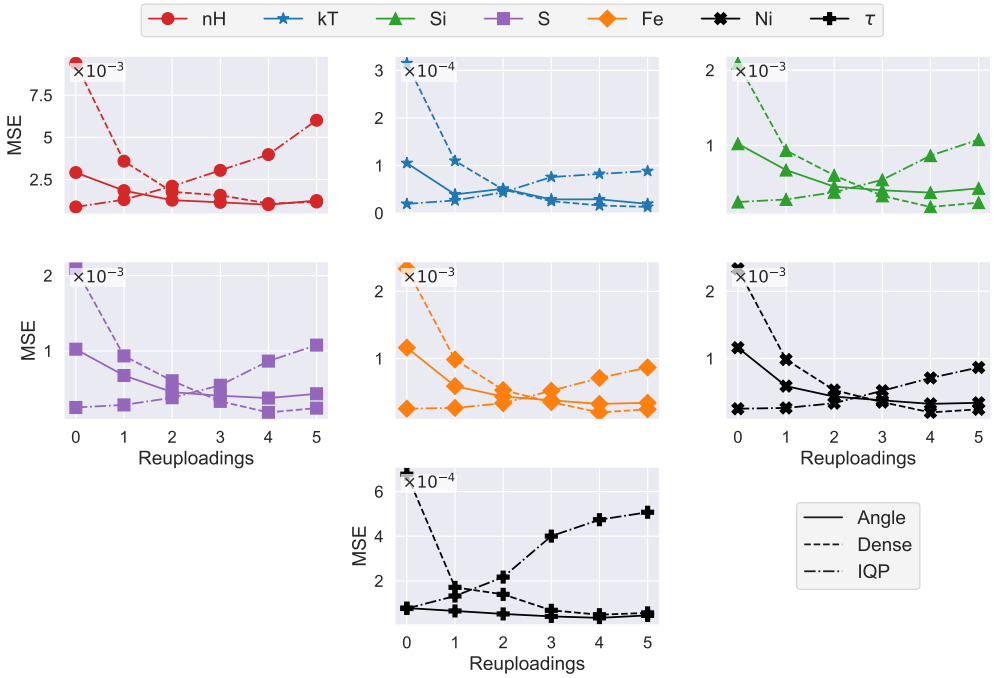


Figure 6.9: **Accuracy Comparison Between Encoding Strategies** - This figure compares the NMSE of the model as a function of the number of uploads, at infinite statistics, considering $N_q = 6$ reservoirs and PLSR compression. Dense encoding achieves the highest peak accuracy at four uploads. While IQP encoding suffers from re-uploading, its single-upload iteration remains highly competitive, presenting high accuracy with low circuit depth.

the model goes in a sort of grid overfitting, where it learns to approximate the step function instead of the correct function modeling the physics of the system. As the mixing increases the system learns to generalize to different distribution, obtaining the correct solution of the problem. Comparing this figure with fig. 6.9 we observe that, while the model is at a certain point capable of retrieving the parameters for this test dataset, it is not as accurate as in the case of the spectra from the initial distribution.

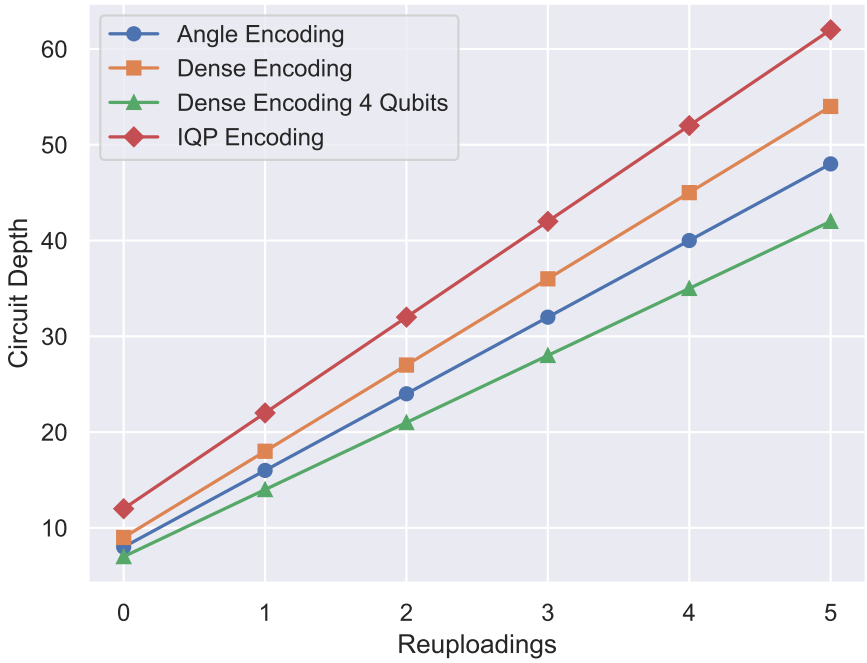


Figure 6.10: **Circuit depth dependence on number of uploads** - This figure shows the dependence of the circuit depth from the number of re-uploads in the different encoding strategies, including also the number reservoir layers. These refer to reservoirs of $N_q = 6$, except for the green line, representing a reservoir of $N_q = 4$ qubits with Dense encoding.

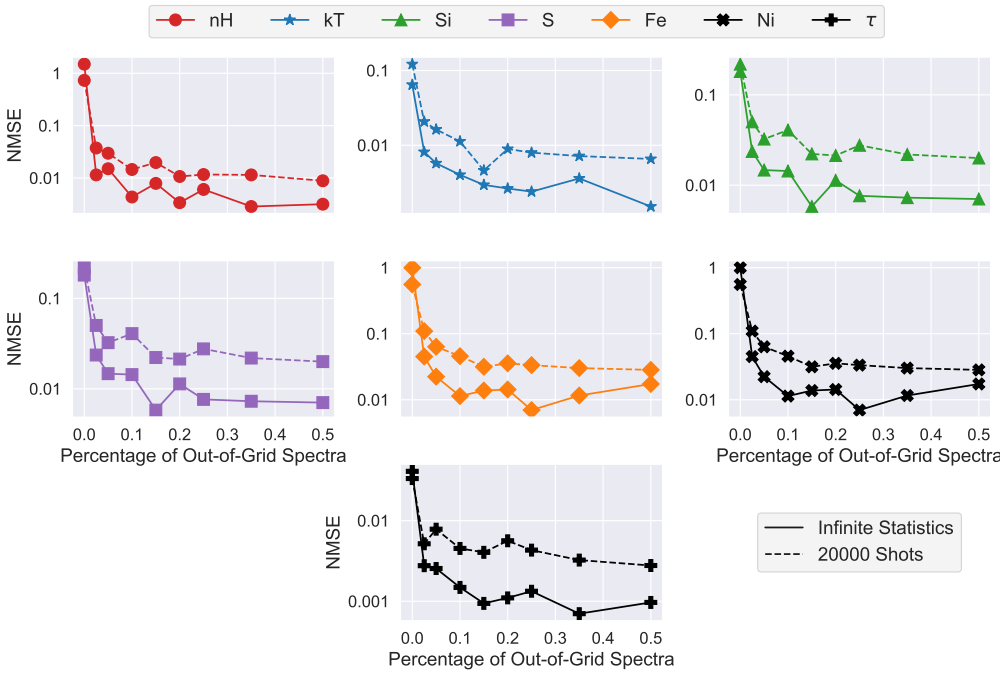


Figure 6.11: **Retrieval accuracy on out-of-grid dataset** - This figure show the retrieval NMSE on a dataset with parameters sampled out of the grid, using the IQP encoding and reservoirs of 6 qubits, using both infinite statistics and a statistics of $N_{shots} = 20000$. In the training phase we have used a dataset with total size of 20000 spectra, which have been gradually mixed with a second composed of spectra with different sampling distribution. The test has been performed on 1000 spectra with a third distribution.

Conclusion

This thesis proves the theoretical viability and practical power of Quantum Extreme Learning Machines for near-term astrophysical applications, showing their potential and limitation, especially in the field of physical implementations.

The conventional framework of quantum information scrambling states that once a system evolves beyond the scrambling time, local information is hidden within non-local correlations and becomes locally irretrievable. However, the findings presented in our work contradict this expectation for state estimation tasks. We demonstrated that the information retrieved via local measurements remains sufficient for accurate state reconstruction well into the scrambling regime. Furthermore, while the interaction topology of the reservoir significantly impacts the estimation accuracy during the initial transient phase of the evolution, these structural differences entirely vanish at longer timescales. In this last regime, the accuracy of all tested topologies converges and matches the performance of a Haar-random unitary evolution, implying that robust experimental state reconstruction does not require precise fine-tuning of the evolution time, nor does it rely on perfect knowledge of the underlying Hamiltonian dynamics.

QELMs represent a highly pragmatic alternative in the landscape of near-term quantum machine learning. In this work, we address the computational bot-

tlenecks of classical exoplanetary retrieval methods and propose the first noise-resilient QELM framework for exoplanetary spectral analysis. The model was evaluated on synthetic datasets dependent on specific chemical and physical parameters—namely CH_4, CO_2, H_2O, CO, M, R, and T. We evaluated variations of this data by restricting the observed wavelengths - comparing the TauREx dataset, comprising wavelengths measured by multiple detectors, with a JWST-specific dataset - and introducing simulated shot noise. The QELM proved exceptionally effective when processing the complete TauREx dataset, achieving an accuracy of over 90% for most parameters (CH_4, CO_2, H_2O, R) at both infinite and finite statistics. Comparable accuracies were maintained when restricting the data to the JWST spectral band. Testing this JWST framework on the 156-qubit IBM Fez hardware not only proved the model's inherent fault tolerance, but also that the QELM outperforms a classical extreme learning machine of comparable feature dimensions. Finally, we have shown that while the model is not able to retrieve the atmospheric parameters when adding shot noise to the dataset, it is possible to improve its accuracy adding a filter employing a principal component analysis to the pre-processing pipeline.

Last but not least, we have also proved that our algorithm can adapt to many different spectral dataset, testing its efficiency also on a spectral dataset of supernova remnant. In this context, we have improved our model, by testing different compression techniques in the pre-processing of the dataset, and different encoding strategies, testing the previously employed angle encoding against dense, IQP encoding and data re-uploading. We have shown that the pre-processing strategy does not provide any kind of advantage while restricting to linear methods as PCA and PLSR. Furthermore, we have proved that while data re-uploading improves the accuracy with both angle and dense encoding, in the case of IQP encoding it becomes detrimental, gradually decreasing the accuracy. Finally, the combination

of dense angle encoding and data re-uploading seems to be generally the more expressive strategy, leading to the overall highest accuracy. Nonetheless, our results show that the IQP encoding achieves similar accuracy without re-uploading the data, therefore maintaining low depth for the circuit.

This thesis establishes the foundational framework for Quantum Machine Learning in astrophysics, driving the transition toward a new generation of fault-tolerant spectral retrieval models.

Bibliography

[1] Alpaydin, E. *Machine learning* (MIT press, 2021).

[2] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *nature* **521**, 436–444 (2015).

[3] Rusk, N. Deep learning. *Nature Methods* **13**, 35–35 (2016).

[4] Amodei, D. *et al.* Deep speech 2: End-to-end speech recognition in english and mandarin. In *International conference on machine learning*, 173–182 (PMLR, 2016).

[5] Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (2016), mit press edn. URL http://www.deeplearningbook.org.

[6] Liu, A. *et al.* Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437* (2024).

[7] Team, G. *et al.* Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530* (2024).

[8] Achiam, J. *et al.* Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

[9] Kingma, D. P. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[10] Nesterov, Y. A method for unconstrained convex minimization problem with the rate of convergence o (1/k2). In *Dokl. Akad. Nauk. SSSR*, vol. 269, 543 (1983).

[11] Huang, G.-B., Wang, D. H. & Lan, Y. Extreme learning machines: a survey. *Int. J. Mach. Learn. & Cyber.* **2**, 107–122 (2011). URL https://doi.org/10.1007/s13042-011-0019-y.

[12] Huang, G.-B., Zhu, Q.-Y. & Siew, C.-K. Extreme learning machine: a new learning scheme of feedforward neural networks. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, vol. 2, 985–990 vol.2 (2004). ISSN: 1098-7576.

[13] Dale, M., Miller, J. F. & Stepney, S. Reservoir Computing as a Model for In-Materio Computing. In Adamatzky, A. (ed.) *Advances in Unconventional Computing*, vol. 22, 533–571 (Springer International Publishing, Cham, 2017). URL http://link.springer.com/10.1007/978-3-319-33924-5_22. Series Title: Emergence, Complexity and Computation.

[14] Budhiraja, R., Kumar, M., Das, M. K., Bafila, A. S. & Singh, S. A reservoir computing approach for forecasting and regenerating both dynamical and time-delay controlled financial system behavior. *PLoS One* **16**, e0246737 (2021).

[15] Lukoševičius, M. & Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Science Review* **3**, 127–149 (2009). URL https://www.sciencedirect.com/science/article/pii/S1574013709000173.

[16] Lukoševičius, M. A Practical Guide to Applying Echo State Networks. In Montavon, G., Orr, G. B. & Müller, K.-R. (eds.) *Neural Networks: Tricks of the Trade: Second Edition*, Lecture Notes in Computer Science, 659–686 (Springer, Berlin, Heidelberg, 2012). URL https://doi.org/10.1007/978-3-642-35289-8_36.

[17] Konkoli, Z. On Reservoir Computing: From Mathematical Foundations to Unconventional Applications. In Adamatzky, A. (ed.) *Advances in Unconventional Computing: Volume 1: Theory*, Emergence, Complexity and Computation, 573–607 (Springer International Publishing, Cham, 2017). URL https://doi.org/10.1007/978-3-319-33924-5_23.

[18] Grigoryeva, L. & Ortega, J.-P. Echo state networks are universal. *Neural Networks* **108**, 495–508 (2018).

[19] Tanaka, G. *et al.* Recent advances in physical reservoir computing: A review. *Neural Networks* **115**, 100–123 (2019). URL https://www.sciencedirect.com/science/article/pii/S0893608019300784.

[20] Lilak, S. *et al.* Spoken Digit Classification by In-Materio Reservoir Computing With Neuromorphic Atomic Switch Networks. *Frontiers in Nanotechnology* **3** (2021). URL https://www.frontiersin.org/articles/10.3389/fnano.2021.675792.

[21] Mastoi, Q.-u.-a., Wah, T. Y. & Gopal Raj, R. Reservoir Computing Based Echo State Networks for Ventricular Heart Beat Classification. *Applied Sciences* **9**, 702 (2019). URL https://www.mdpi.com/2076-3417/9/4/702. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute.

[22] Bhovad, P. & Li, S. Physical reservoir computing with origami and its application to robotic crawling. *Sci Rep* **11**, 13002 (2021). URL https://

www.nature.com/articles/s41598-021-92257-1. Number: 1 Publisher: Nature Publishing Group.

[23] Goudarzi, A., Lakin, M. R. & Stefanovic, D. DNA Reservoir Computing: A Novel Molecular Computing Approach. In Soloveichik, D. & Yurke, B. (eds.) *DNA Computing and Molecular Programming*, Lecture Notes in Computer Science, 76–89 (Springer International Publishing, Cham, 2013).

[24] Nokkala, J., Martínez-Peña, R., Zambrini, R. & Soriano, M. C. High-Performance Reservoir Computing With Fluctuations in Linear Networks. *IEEE Transactions on Neural Networks and Learning Systems* **33**, 2664–2675 (2022). URL https://ieeexplore.ieee.org/abstract/document/9525045. Conference Name: IEEE Transactions on Neural Networks and Learning Systems.

[25] Borrageiro, G., Firoozye, N. & Barucca, P. The Recurrent Reinforcement Learning Crypto Agent. *IEEE Access* **10**, 38590–38599 (2022). Conference Name: IEEE Access.

[26] Jaeger, H. & Haas, H. Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication. *Science* **304**, 78–80 (2004).

[27] Nakajima, K., Fujii, K., Negoro, M., Mitarai, K. & Kitagawa, M. Boosting computational power through spatial multiplexing in quantum reservoir computing. *Physical Review Applied* **11**, 034021 (2019).

[28] Vetrano, M., Lo Monaco, G., Innocenti, L., Lorenzo, S. & Palma, G. M. State estimation with quantum extreme learning machines beyond the scrambling time. *npj Quantum Information* **11**, 20 (2025).

[29] Innocenti, L. *et al.* Potential and limitations of quantum extreme learning machines. *Communications Physics* **6**, 118 (2023).

[30] Mujal, P. *et al.* Opportunities in quantum reservoir computing and extreme learning machines. *Advanced Quantum Technologies* **4**, 2100027 (2021).

[31] Mujal, P., Martínez-Peña, R., Giorgi, G. L., Soriano, M. C. & Zambrini, R. Time-series quantum reservoir computing with weak and projective measurements. *npj Quantum Inf* **9**, 1–10 (2023). URL https://www.nature.com/articles/s41534-023-00682-z. Number: 1 Publisher: Nature Publishing Group.

[32] Sannia, A., Martínez-Peña, R., Soriano, M. C., Giorgi, G. L. & Zambrini, R. Dissipation as a resource for quantum reservoir computing. *Quantum* **8**, 1291 (2024).

[33] Nokkala, J., Giorgi, G. L. & Zambrini, R. Retrieving past quantum features with deep hybrid classical-quantum reservoir computing. *Machine Learning: Science and Technology* **5**, 035022 (2024).

[34] Martínez-Peña, R. & Ortega, J.-P. Quantum reservoir computing in finite dimensions. *Physical Review E* **107**, 035306 (2023).

[35] Fujii, K. & Nakajima, K. Harnessing Disordered-Ensemble Quantum Dynamics for Machine Learning. *Phys. Rev. Appl.* **8**, 024030 (2017). URL https://link.aps.org/doi/10.1103/PhysRevApplied.8.024030. Publisher: American Physical Society.

[36] Ghosh, S., Krisnanda, T., Paterek, T. & Liew, T. C. H. Realising and compressing quantum circuits with quantum reservoir computing. *Commun Phys* **4**, 1–7 (2021). URL https://www.nature.com/articles/s42005-021-00606-3. Number: 1 Publisher: Nature Publishing Group.

[37] Martínez-Peña, R., Nokkala, J., Giorgi, G. L., Zambrini, R. & Soriano, M. C. Information Processing Capacity of Spin-Based Quantum Reservoir Com-

puting Systems. *Cogn Comput* (2020). URL https://doi.org/10.1007/s12559-020-09772-y.

[38] Tran, Q. H. & Nakajima, K. Higher-Order Quantum Reservoir Computing (2020). URL http://arxiv.org/abs/2006.08999. ArXiv:2006.08999 [nlin, physics:quant-ph].

[39] Nokkala, J. *et al.* Gaussian states of continuous-variable quantum systems provide universal and versatile reservoir computing. *Commun Phys* **4**, 1–11 (2021). URL https://www.nature.com/articles/s42005-021-00556-w. Number: 1 Publisher: Nature Publishing Group.

[40] Martínez-Peña, R. & Ortega, J.-P. Quantum reservoir computing in finite dimensions. *Phys. Rev. E* **107**, 035306 (2023). URL https://link.aps.org/doi/10.1103/PhysRevE.107.035306. Publisher: American Physical Society.

[41] Domingo, L., Carlo, G. & Borondo, F. Taking advantage of noise in quantum reservoir computing. *Sci Rep* **13**, 8790 (2023). URL https://www.nature.com/articles/s41598-023-35461-5. Number: 1 Publisher: Nature Publishing Group.

[42] Govia, L., Ribeill, G., Rowlands, G., Krovi, H. & Ohki, T. Quantum reservoir computing with a single nonlinear oscillator. *Physical Review Research* **3**, 013077 (2021). URL https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.3.013077.

[43] Xiong, W. *et al.* On fundamental aspects of quantum extreme learning machines (2023). URL http://arxiv.org/abs/2312.15124. ArXiv:2312.15124 [quant-ph, stat].

[44] Čindrak, S., Donvil, B., Lüdge, K. & Jaurigue, L. Enhancing the performance of quantum reservoir computing and solving the time-complexity problem by artificial memory restriction. *Physical Review Research* **6**, 013051 (2024).

[45] Dudas, J. *et al.* Quantum reservoir computing implementation on coherently coupled quantum oscillators. *npj Quantum Information* **9**, 64 (2023). URL https://www.nature.com/articles/s41534-023-00734-4.

[46] García-Beni, J., Giorgi, G. L., Soriano, M. C. & Zambrini, R. Scalable Photonic Platform for Real-Time Quantum Reservoir Computing. *Phys. Rev. Appl.* **20**, 014051 (2023). URL https://link.aps.org/doi/10.1103/PhysRevApplied.20.014051. Publisher: American Physical Society.

[47] García-Beni, J., Giorgi, G. L., Soriano, M. C. & Zambrini, R. Squeezing as a resource for time series processing in quantum reservoir computing. *Optics Express* **32**, 6733–6747 (2024).

[48] De Lorenzis, A. *et al.* Harnessing quantum extreme learning machines for image classification. *arXiv preprint arXiv:2409.00998* (2024).

[49] De Lorenzis, A. *et al.* Behind the scenes of the quantum extreme learning machines. *arXiv preprint arXiv:2509.06873* (2025).

[50] Lo Monaco, G., Bertini, M., Lorenzo, S. & Massimo Palma, G. Quantum extreme learning of molecular potential energy surfaces and force fields. *Machine Learning: Science and Technology* **5**, 35014 (2024). 2406.14607.

[51] Zia, D. *et al.* Quantum extreme learning machines for photonic entanglement witnessing (2025). URL http://arxiv.org/abs/2502.18361. 2502.18361.

[52] Suprano, A. *et al.* Experimental property reconstruction in a photonic quantum extreme learning machine. *Physical Review Letters* **132**, 160802 (2024).

[53] Čindrak, S., Jaurigue, L. & Lüdge, K. Krylov expressivity in quantum reservoir computing and quantum extreme learning. *arXiv preprint arXiv:2409.12079* (2024).

[54] Wang, X., Ali, S., Arrieta, A., Arcaini, P. & Arratibel, M. Application of Quantum Extreme Learning Machines for QoS Prediction of Elevators' Software in an Industrial Context (2024). URL http://arxiv.org/abs/2402.12777. ArXiv:2402.12777 [cs].

[55] Swingle, B., Bentsen, G., Schleier-Smith, M. & Hayden, P. Measuring the scrambling of quantum information. *Phys. Rev. A* **94**, 040302 (2016). URL https://link.aps.org/doi/10.1103/PhysRevA.94.040302. Publisher: American Physical Society.

[56] Hosur, P., Qi, X.-L., Roberts, D. A. & Yoshida, B. Chaos in quantum channels. *J. High Energ. Phys.* **2016**, 4 (2016). URL https://doi.org/10.1007/JHEP02(2016)004.

[57] Zanardi, P. & Anand, N. Information scrambling and chaos in open quantum systems. *Physical Review A* **103**, 062214 (2021).

[58] Hashimoto, K., Murata, K. & Yoshii, R. Out-of-time-order correlators in quantum mechanics. *Journal of High Energy Physics* **2017**, 1–31 (2017). URL https://link.springer.com/article/10.1007/JHEP10(2017)138.

[59] Monaco, G. L. *et al.* Quantum scrambling via accessible tripartite information. *Quantum Science and Technology* **8**, 035006 (2023). URL https://iopscience.iop.org/article/10.1088/2058-9565/accd92/meta.

[60] Lo Monaco, G. *et al.* An operational definition of quantum information scrambling (2023). 2312.11619.

[61] Schnaack, O. *et al.* Tripartite information, scrambling, and the role of Hilbert space partitioning in quantum lattice models. *Phys. Rev. B* **100**, 224302 (2019). URL https://link.aps.org/doi/10.1103/PhysRevB.100.224302. Publisher: American Physical Society.

[62] Touil, A. & Deffner, S. Quantum scrambling and the growth of mutual information. *Quantum Science and Technology* **5**, 035005 (2020).

[63] Yan, B., Cincio, L. & Zurek, W. H. Information scrambling and loschmidt echo. *Physical review letters* **124**, 160603 (2020). URL https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.124.160603.

[64] Yuan, D., Zhang, S.-Y., Wang, Y., Duan, L.-M. & Deng, D.-L. Quantum information scrambling in quantum many-body scarred systems. *Phys. Rev. Res.* **4**, 023095 (2022). URL https://link.aps.org/doi/10.1103/PhysRevResearch.4.023095. Publisher: American Physical Society.

[65] Zhuang, J.-Z., Wu, Y.-K. & Duan, L.-M. Phase-transition-like behavior in information retrieval of a quantum scrambled random circuit system. *Physical Review B* **106**, 144308 (2022).

[66] Kreidberg, L. *et al.* Global Climate and Atmospheric Composition of the Ultra-hot Jupiter WASP-103b from HST and Spitzer Phase Curve Observations. *The Astronomical Journal* **156**, 17 (2018). 1805.00029.

[67] Tsiaras, A. *et al.* A Population Study of Gaseous Exoplanets. *The Astronomical Journal* **155**, 156 (2018). 1704.05413.

[68] Bruno, G. *et al.* A Comparative Study of WASP-67 b and HAT-P-38 b from WFC3 Data. *The Astronomical Journal* **155**, 55 (2018). 1712.03384.

[69] Mansfield, M. *et al.* An HST/WFC3 Thermal Emission Spectrum of the Hot Jupiter HAT-P-7b. *The Astronomical Journal* **156**, 10 (2018). 1805.00424.

[70] Spake, J. J. *et al.* Helium in the eroding atmosphere of an exoplanet. *Nature* **557**, 68–70 (2018).

[71] Sheppard, K. B. *et al.* Evidence for a Dayside Thermal Inversion and High Metallicity for the Hot Jupiter WASP-18b. *The Astrophysical Journal Letters* **850**, L32 (2017). 1711.10491.

[72] Barstow, J. K., Aigrain, S., Irwin, P. G. J. & Sing, D. K. A CONSISTENT RETRIEVAL ANALYSIS OF 10 HOT JUPITERS OBSERVED IN TRANSMISSION. *The Astrophysical Journal* **834**, 50 (2016). 1610.01841.

[73] Rocchetto, M., Waldmann, I. P., Venot, O., Lagage, P.-O. & Tinetti, G. EXPLORING BIASES OF ATMOSPHERIC RETRIEVALS IN SIMULATED JWST TRANSMISSION SPECTRA OF HOT JUPITERS. *The Astrophysical Journal* **833**, 120 (2016). 1610.02848.

[74] Irwin, P. G. *et al.* The NEMESIS planetary atmosphere radiative transfer and retrieval tool. *Journal of Quantitative Spectroscopy and Radiative Transfer* **109**, 1136–1150 (2008).

[75] Madhusudhan, N. & Seager, S. A temperature and abundance retrieval method for exoplanet atmospheres. *Astrophysical Journal* **707**, 24–39 (2009). 0910.1347.

[76] Line, M. R. *et al.* A systematic retrieval analysis of secondary eclipse spectra. I. A comparison of atmospheric retrieval techniques. *Astrophysical Journal* **775**, 137 (2013). 1304.5561.

[77] Benneke, B. & Seager, S. How to distinguish between cloudy mini-neptunes and water/volatile- dominated super-earths. *Astrophysical Journal* **778**, 153 (2013). 1306.6325.

[78] Cubillos, P. *et al.* BART: Bayesian Atmospheric Radiative Transfer code. *Astrophysics Source Code Library* ascl—-1608 (2020).

[79] Gandhi, S. & Madhusudhan, N. Retrieval of exoplanet emission spectra with HyDRA. *Monthly Notices of the Royal Astronomical Society* **474**, 271–278 (2018). 1710.06433.

[80] Lavie, B. *et al.* HELIOS–RETRIEVAL: An Open-source, Nested Sampling Atmospheric Retrieval Code; Application to the HR 8799 Exoplanets and Inferred Constraints for Planet Formation. *The Astronomical Journal* **154**, 91 (2017). 1610.03216.

[81] Wang, J., Lu, S., Wang, S.-H. & Zhang, Y.-D. A review on extreme learning machine. *Multimedia Tools and Applications* **81**, 41611–41660 (2022). URL https://doi.org/10.1007/s11042-021-11007-7.

[82] Huang, G.-B., Chen, L. & Siew, C. Universal Approximation Using Incremental Constructive Feedforward Networks With Random Hidden Nodes **17**, 879–92 (2006).

[83] Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems* **2**, 303–314 (1989).

[84] Hornik, K. Approximation capabilities of multilayer feedforward networks. *Neural networks* **4**, 251–257 (1991).

[85] Browne, M. & Ghidary, S. S. Convolutional Neural Networks for Image Processing: An Application in Robot Vision. In Gedeon, T. T. D. & Fung, L. C. C. (eds.) *AI 2003: Advances in Artificial Intelligence*, Lecture Notes in Computer Science, 641–652 (Springer, Berlin, Heidelberg, 2003).

[86] Vaswani, A. *et al.* Attention is all you need. *Advances in neural information processing systems* **30** (2017).

[87] Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural computation* **9**, 1735–1780 (1997).

[88] Ruder, S. An overview of gradient descent optimization algorithms (2017). URL http://arxiv.org/abs/1609.04747. ArXiv:1609.04747 [cs].

[89] Amari, S.-i. Backpropagation and stochastic gradient descent method. *Neurocomputing* **5**, 185–196 (1993).

[90] Bengio, Y., Simard, P. & Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks* **5**, 157–166 (1994).

[91] Pascanu, R., Mikolov, T. & Bengio, Y. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, 1310–1318 (Pmlr, 2013).

[92] Serre, D. Matrices with entries in a principal ideal domain; jordan reduction. In *Matrices: Theory and Applications*, 163–181 (Springer, 2010).

[93] Belsley, D. A., Kuh, E. & Welsch, R. E. *Regression diagnostics: Identifying influential data and sources of collinearity* (John Wiley & Sons, 2005).

[94] Tikhonov, A. N. Solution of incorrectly formulated problems and the regularization method. *Sov Dok* **4**, 1035–1038 (1963).

[95] Yildiz, I. B., Jaeger, H. & Kiebel, S. J. Re-visiting the echo state property. *Neural networks* **35**, 1–9 (2012).

[96] Del Papa, B., Priesemann, V. & Triesch, J. Fading memory, plasticity, and criticality in recurrent networks. In *The functional role of critical dynamics in neural systems*, 95–115 (Springer, 2019).

[97] Boyd, S. & Chua, L. Fading memory and the problem of approximating nonlinear operators with volterra series. *IEEE Transactions on circuits and systems* **32**, 1150–1161 (2003).

[98] Fette, G. & Eggert, J. Short term memory and pattern matching with simple echo state networks. In *International Conference on Artificial Neural Networks*, 13–18 (Springer, 2005).

[99] Lintern, G. & Kugler, P. N. Self-organization in connectionist models: Associative memory, dissipative structures, and thermodynamic law. *Human Movement Science* **10**, 447–483 (1991).

[100] Krishnagopal, S., Girvan, M., Ott, E. & Hunt, B. R. Separation of chaotic signals by reservoir computing. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **30** (2020).

[101] Munikote, N. Comparing quantum encoding techniques. *arXiv preprint arXiv:2410.09121* (2024).

[102] Li, H., Ni, H. & Ying, L. On efficient quantum block encoding of pseudo-differential operators. *Quantum* **7**, 1031 (2023).

[103] Nielsen, A., M., Chuang & L., I. *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, 2010).

[104] Breuer, H.-P., Petruccione, F., Breuer, H.-P. & Petruccione, F. *The Theory of Open Quantum Systems* (Oxford University Press, Oxford, New York, 2007).

[105] Caruso, F., Giovannetti, V., Lupo, C. & Mancini, S. Quantum channels and memory effects. *Reviews of Modern Physics* **86**, 1203–1259 (2014).

[106] Stinespring, W. F. Positive functions on c*-algebras. *Proceedings of the american mathematical society* **6**, 211–216 (1955).

[107] Fujii, K. & Nakajima, K. Harnessing disordered-ensemble quantum dynamics for machine learning. *Physical Review Applied* **8**, 024030 (2017).

[108] Chen, J. & Nurdin, H. I. Correction to: Learning nonlinear input–output maps with dissipative quantum systems. *Quantum Information Processing* **18**, 354 (2019).

[109] Lau, J. W. Z., Lim, K. H., Shrotriya, H. & Kwek, L. C. NISQ computing: where are we and where do we go? *AAPPS bulletin* **32**, 27 (2022).

[110] Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018). 1801.00862.

[111] Paetznick, A. *et al.* Demonstration of logical qubits and repeated error correction with better-than-physical error rates. *arXiv preprint arXiv:2404.02280* (2024).

[112] Suppressing quantum errors by scaling a surface code logical qubit. *Nature* **614**, 676–681 (2023).

[113] Cai, W. *et al.* Protecting entanglement between logical qubits via quantum error correction. *Nature Physics* **20**, 1022–1026 (2024).

[114] Havlíček, V. *et al.* Supervised learning with quantum-enhanced feature spaces. *Nature* **567**, 209–212 (2019).

[115] Pérez-Salinas, A., Cervera-Lierta, A., Gil-Fuster, E. & Latorre, J. I. Data re-uploading for a universal quantum classifier. *Quantum* **4**, 226 (2020).

[116] Ciccarello, F., Lorenzo, S., Giovannetti, V. & Palma, G. M. Quantum collision models: Open system dynamics from repeated interactions. *Physics Reports* **954**, 1–70 (2022).

[117] Sannia, A., Giorgi, G. L. & Zambrini, R. Exponential concentration and symmetries in quantum reservoir computing. *arXiv preprint arXiv:2505.10062* (2025).

[118] Sekino, Y. & Susskind, L. Fast scramblers. *Journal of High Energy Physics* **2008**, 065 (2008).

[119] Hayden, P. & Preskill, J. Black holes as mirrors: quantum information in random subsystems. *J. High Energy Phys.* **2007**, 120 (2007). URL https://dx.doi.org/10.1088/1126-6708/2007/09/120.

[120] Swingle, B. Unscrambling the physics of out-of-time-order correlators. *Nature Physics* **14**, 988–990 (2018).

[121] Xu, S. & Swingle, B. Scrambling dynamics and out-of-time-ordered correlators in quantum many-body systems. *PRX quantum* **5**, 010201 (2024). URL https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.5.010201.

[122] Gherardini, S. & De Chiara, G. Quasiprobabilities in quantum thermodynamics and many-body systems. *PRX Quantum* **5**, 030201 (2024). URL https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.5.030201.

[123] Green, A. M. *et al.* Experimental measurement of out-of-time-ordered correlators at finite temperature. *Physical Review Letters* **128**, 140601 (2022). URL https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.128.140601.

[124] Li, J. *et al.* Measuring Out-of-Time-Order Correlators on a Nuclear Magnetic Resonance Quantum Simulator. *Phys. Rev. X* **7**, 031011 (2017). URL https://link.aps.org/doi/10.1103/PhysRevX.7.031011. Publisher: American Physical Society.

[125] Nie, X. *et al.* Experimental observation of equilibrium and dynamical quantum phase transitions via out-of-time-ordered correlators. *Physical Review Letters* **124**, 250601 (2020).

[126] Halpern, N. Y., Swingle, B. & Dressel, J. Quasiprobability behind the out-of-time-ordered correlator. *Physical Review A* **97**, 042105 (2018).

[127] Jalabert, R. A., García-Mata, I. & Wisniacki, D. A. Semiclassical theory of out-of-time-order correlators for low-dimensional classically chaotic systems. *Phys. Rev. E* **98**, 062218 (2018). URL https://link.aps.org/doi/10.1103/PhysRevE.98.062218.

[128] Holevo, A. S. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii* **9**, 3–11 (1973).

[129] Qi, X.-L., Shangnan, Z. & Yang, Z. Holevo information and ensemble theory of gravity. *Journal of High Energy Physics* **2022**, 1–24 (2022).

[130] Vetrano, M., Zingales, T., Palma, G. M. & Lorenzo, S. Exoplanetary atmospheres retrieval via a quantum extreme learning machine. *arXiv preprint arXiv:2509.03617* (2025).

[131] Innocenti, L. *et al.* Shadow tomography on general measurement frames. *PRX Quantum* **4**, 040328 (2023).

[132] Huang, H.-Y., Kueng, R. & Preskill, J. Predicting many properties of a quantum system from very few measurements. *Nature Physics* **16**, 1050–1057 (2020).

[133] Pluriel, W., Zingales, T., Leconte, J. & Parmentier, V. Strong biases in retrieved atmospheric composition caused by day-night chemical heterogeneities. *Astronomy and Astrophysics* **636**, A66 (2020). URL https://doi.org/10.1051/0004-6361/202037678. 2003.05943.

[134] Pluriel, W. *et al.* Toward a multidimensional analysis of transmission spectroscopy: II. Day-night-induced biases in retrievals from hot to ultrahot Jupiters. *Astronomy and Astrophysics* **658**, A42 (2022). 2110.09080.

[135] Feroz, F. & Hobson, M. P. Multimodal nested sampling: An efficient and robust alternative to Markov Chain Monte Carlo methods for astronomical data analyses. *Monthly Notices of the Royal Astronomical Society* **384**, 449–463 (2008).

[136] Skilling, J. Nested sampling. *Bayesian inference and maximum entropy methods in science and engineering* **735**, 395–405 (2004).

[137] Feroz, F., Gair, J. R., Hobson, M. P. & Porter, E. K. Use of the MULTINEST algorithm for gravitational wave data analysis. *Classical and Quantum Gravity* **26**, 215003 (2009). 0904.1544.

[138] Gregory, P. C. Bayesian exoplanet tests of a new method for MCMC sampling in highly correlated model parameter spaces. *Monthly Notices of the Royal Astronomical Society* **410**, 94–110 (2011).

[139] Waldmann, I. P. DREAMING OF ATMOSPHERES. *The Astrophysical Journal* **820**, 107 (2016). 1511.08339.

[140] Zingales, T. & Waldmann, I. P. ExoGAN: Retrieving Exoplanetary Atmospheres Using Deep Convolutional Generative Adversarial Networks. *The Astronomical Journal* **156**, 268 (2018). 1806.02906.

[141] Márquez-Neila, P., Fisher, C., Sznitman, R. & Heng, K. Supervised machine learning for analysing spectra of exoplanetary atmospheres. *Nature Astronomy* **2**, 719–724 (2018). 1806.03944.

[142] Cobb, A. D. *et al.* An Ensemble of Bayesian Neural Networks for Exoplanetary Atmospheric Retrieval. *The Astronomical Journal* **158**, 33 (2019). 1905.10659.

[143] Gardner, J. P. The james webb space telescope. *Proceedings of the Eighteenth International Symposium on Space Terahertz Technology 2007, ISSTT 2007* **123**, 237 (2007).

[144] Tinetti, G. *et al.* The science of ARIEL (Atmospheric Remote-sensing Infrared Exoplanet Large-survey). In *Space Telescopes and Instrumentation 2016: Optical, Infrared, and Millimeter Wave*, vol. 9904, 99041X (SPIE, 2016).

[145] Huang, H. L. *et al.* Experimental Quantum Generative Adversarial Networks for Image Generation. *Physical Review Applied* **16**, 24051 (2021). 2010.06201.

[146] Al-Refaie, A. F., Changeat, Q., Waldmann, I. P. & Tinetti, G. TauREx 3: A Fast, Dynamic, and Extendable Framework for Retrievals. *The Astrophysical Journal* **917**, 37 (2021). 1912.07759.

[147] Waldmann, I. P. *et al.* Tau-REx I: A next generation retrieval code for exoplanetary atmospheres. *Astrophysical Journal* **802**, 107 (2015). 1409.2312.

[148] Fu, G. *et al.* Water and an Escaping Helium Tail Detected in the Hazy and Methane-depleted Atmosphere of HAT-P-18b from JWST NIRISS/SOSS. *The Astrophysical Journal Letters* **940**, L35 (2022). 2211.13761.

[149] Cowan, N. B. *et al.* Characterizing transiting planet atmospheres through 2025. *Publications of the Astronomical Society of the Pacific* **127**, 311 (2015).

[150] Huang, H. Y. *et al.* Power of data in quantum machine learning. *Nature Communications* **12**, 2631 (2021). 2011.01938.

[151] Rebentrost, P., Mohseni, M. & Lloyd, S. Quantum support vector machine for big data classification. *Physical Review Letters* **113**, 130503 (2014). 1307.0471.

[152] Senokosov, A., Sedykh, A., Sagingalieva, A., Kyriacou, B. & Melnikov, A. Quantum machine learning for image classification. *Machine Learning: Science and Technology* **5**, 15040 (2024).

[153] Greene, T. P. *et al.* CHARACTERIZING TRANSITING EXOPLANET ATMOSPHERES WITHJWST. *The Astrophysical Journal* **817**, 17 (2016). URL https://doi.org/10.3847%2F0004-637x%2F817%2F1%2F17.

[154] Tsang, S. L., West, M. T., Erfani, S. M. & Usman, M. Hybrid Quantum-Classical Generative Adversarial Network for High-Resolution Image Generation. *IEEE Transactions on Quantum Engineering* **4** (2023). 2212.11614.

[155] Arnaud, K. Astronomical data analysis software and systems v. In *ASP Conf.*, vol. 17 (1996).

[156] Anders, E. & Grevesse, N. Abundances of the elements: Meteoritic and solar. *Geochimica et Cosmochimica acta* **53**, 197–214 (1989).

[157] Strüder, L. *et al.* The european photon imaging camera on xmm-newton: the pn-ccd camera. *Astronomy & Astrophysics* **365**, L18–L26 (2001).