XLVI CILAMCE
VITÓRIA - ES - BRAZIL - 2025
XLVI IBERO-LATIN AMERICAN CONGRESS ON COMPUTATIONAL METHODS IN ENGENEERING

Reservoir computing for chaotic time series prediction

Vitor Barros Jr[1], Karla Tereza Figueiredo Leite[1], Americo Cunha Jr.[1,2]

[1]*Rio de Janeiro State University Rua São Francisco Xavier, 524, 20550-900, Rio de Janeiro – RJ, Brasil*
[2]*National Laboratory of Scientific Computing Av. Getúlio Vargas, 333, 25651-075, Petrópolis – RJ, Brasil*
vitor.junior@pos.ime.uerj.br, karlafigueiredo@ime.uerj.br, americo@lncc.br

Abstract. Reservoir computing is a machine learning paradigm inspired by the dynamics of recurrent systems, where a fixed nonlinear dynamical system—the reservoir—transforms input signals into high-dimensional representations. These representations allow a simple linear combination at the readout to approximate complex temporal relationships with minimal training effort. Since its emergence in the early 2000s, this approach has shown great potential in predicting chaotic behavior, particularly by exploiting the underlying structure of nonlinear dynamical systems. However, key challenges remain in making reservoir computing more robust, interpretable, and broadly effective across diverse application domains. This work investigates the predictive capabilities of a reservoir computing architecture based on untrained recurrent dynamics combined with a trained linear readout. The method is applied to two benchmark chaotic systems: the Duffing oscillator and the Rössler attractor. A validation strategy specifically designed for chaotic time series is employed to optimize key model hyperparameters. The results demonstrate accurate short-term predictions and highlight the method's sensitivity to initialization, underscoring the importance of proper validation and hyperparameter tuning.

Keywords: reservoir computing, chaotic time-series prediction, echo state networks, nonlinear dynamics, data-driven forecasting

1 Introduction

Modern applications in science and engineering increasingly demand *fast, accurate, and energy-efficient* processing of high-dimensional time–series data. Deep Learning approaches have delivered outstanding performance in many such tasks, yet their dependence on large data sets and training millions of parameters limits their use in resource-constrained or real-time scenarios. By contrast, biological neural circuits achieve impressive information processing with minimal energy expenditure because only a sparse subset of neurons is active at any given time. Inspired by this observation, the neuromorphic computing community seeks alternatives to the von Neumann architecture that drastically reduce power consumption [1].

Reservoir computing (RC) lies at the intersection of neuromorphic hardware and recurrent neural modelling. In the standard RC scheme, a fixed, recurrent dynamical system—the *reservoir*—performs a nonlinear expansion of the input, after which a *linear* readout is trained to accomplish the desired task. Crucially, only the readout weights are adjusted; the internal reservoir weights remain untrained, mirroring the static synaptic backbone found in many biological circuits. Two seminal but initially independent frameworks, Echo State Networks (ESNs) [2] and Liquid State Machines (LSMs) [3], were later unified under the RC umbrella by Schrauwen and colleagues [4]. Because the reservoir is not subject to backpropagation, RC retains very low training cost while still capturing rich nonlinear dynamics.

Although RC has been shown to excel in nonlinear filtering, real-time speech recognition, and chaotic time-series prediction, its adoption outside laboratory settings has been slower than that of feed-forward deep networks. The main obstacles are (i) the opaque and highly problem-dependent relationship between randomly initialized reservoir weights and downstream performance, and (ii) the absence of systematic validation strategies that ensure robust generalisation when the underlying dynamics are chaotic. Overcoming these obstacles is particularly urgent for *chaotic forecasting*, where small modelling errors exponentially amplify in time and a naive validation split may be misleading.

Echo State Networks enjoy the universal-approximation property [5] and, under ergodicity, can learn invariant measures of dynamical systems [6]. Two main ESN flavours are actively studied: *model-free* architectures, which treat the physical system as a black box [7], and *model-informed* variants that embed partial mechanistic knowledge

[8]. Despite these advances, the literature still lacks (i) a quantitative assessment of how validation strategies affect prediction horizons in chaotic regimes and (ii) a careful examination of the sensitivity of ESN performance to random reservoir initialisation.

We employ a *model-free* ESN together with the recycled-validation methodology proposed by [9] and systematically evaluate their impact on short-term chaotic forecasting. Specifically, we (i) apply the framework to two archetypal chaotic systems—the Duffing oscillator and the Rössler attractor—chosen for their distinct nonlinear mechanisms; (ii) quantify prediction accuracy, prediction horizon (in Lyapunov times) and computational cost over 25 independent reservoir initialisations; (iii) demonstrate that recycled validation reduces hyperparameter-search variance and extends the average prediction horizon to about eight Lyapunov times for the Duffing system and eleven for the Rössler system; (iv) identify spectral radius and input scaling as the dominant hyperparameters and show how their optimal values cluster across initialisations, offering practical guidelines for future RC deployments. By articulating validation strategy, sensitivity analysis and quantitative benchmarks, we provide new insights that help move reservoir computing closer to becoming a practical forecasting tool for chaotic dynamical systems.

2 Data-driven chaotic dynamics forecasting

This section consolidates the network architecture, the training and validation workflow, and the benchmark chaotic systems used to evaluate the data-driven forecasting approach employed in this work. An overview of the entire pipeline is shown in Fig. 1.



Figure 1. Machine Learning pipeline employed during the process.

2.1 Echo State Network (ESN) architecture

Echo State Networks (ESNs) [2, 10] are employed as the reservoir computing core. Their discrete-time state and output equations are

$$\mathbf{x}(t+1) = f\left(W_{res}^{res}\mathbf{x}(t) + W_{inp}^{res}\mathbf{u}(t) + W_{out}^{res}\mathbf{y}(t) + W_{bias}^{res}\right), \tag{1}$$

$$\hat{\mathbf{y}}(t+1) = W_{res}^{out}\mathbf{x}(t+1) + W_{inp}^{out}\mathbf{u}(t) + W_{bias}^{out}. \tag{2}$$

Only the read–out matrices $W_{\{\cdot\}}^{out}$ are trained; the reservoir weights $W_{\{\cdot\}}^{res}$ remain fixed after random initialisation [11]. Unless otherwise stated

- *Activation function*: $f(z) = 1.7159\,\tanh\left(\frac{2}{3}z\right)$ which is an option that gives a wider linear regime and reduces gradient saturation [12].
- *Reservoir graph*: W_{res}^{res} is the weighted adjacency matrix of a directed Erdős–Rényi–Gilbert graph with connection probability p [10]. Non-zero weights are drawn from $\mathcal{U}[-1,1]$ and rescaled so that the spectral radius satisfies $\rho\left(W_{res}^{res}\right) = \rho_{target} < 1$, enforcing the echo–state property [13, 14]. The specific values used

are $p = 0.003$ for the two dynamical systems and ρ_{target} optimised during validation (Sec. 2.2). Search bounds are 0.1 for lower bound and 1.0 for upper bound.
- *Input scaling*: the matrix W_{inp}^{res} is multiplied by an input-scale factor s_{inp} optimised during validation (Sec. 2.2). Search bounds are 0.05 for lower bound and 5.0 for upper bound.
- *Reservoir size*: the number of internal units is $1\,000$.
- *Bias*: an input bias of order one is appended to break symmetry [15].

Raw input signals are first centered and rescaled to the interval $[-1, 1]$, keeping the operating point in the approximately linear region of $f(\cdot)$. Reservoir activities are standardised before linear regression; ridge regularisation with coefficient 10^{-9} for Rössler attractor and 10^{-12} for Duffing Oscillator mitigates multicollinearity [16].

2.2 Training, prediction modes, and validation

The ESN operates in two modes:
1. *Predictive (open loop)* — training data $\mathbf{u}(t)$ drive the reservoir; readout weights $W_{\{\cdot\}}^{out}$ are fitted after an initial washout of 50 time steps. White noise with standard deviation 10^{-6} is added to the inputs, to improve robustness in chaotic dynamics [14, 17].
2. *Generative (closed loop)* — the network's own one-step prediction $\hat{\mathbf{y}}(t)$ is fed back as $\mathbf{u}(t{+}1)$, producing an autonomous forecast.

Hyperparameters s_{inp} and ρ_{target} are tuned via the *recycled validation* strategy of Racca and Magri [9]. The full training set is reused in k folds ($k = 25$), and the mean squared error in generative mode is minimised. The optimisation algorithm is Bayesian search.

2.3 Chaotic benchmark systems

Two canonical chaotic oscillators are used to benchmark the proposed ESN pipeline. Their governing equations and numerical settings are summarised below.

Duffing oscillator. The forced double-well Duffing equation is

$$\ddot{x}_D + \delta_D\,\dot{x}_D - \beta_D\,x_D + \alpha_D\,x_D^3 \;=\; \gamma_D\cos(\omega_D t), \tag{3}$$

with parameters $\delta_D = 0.3$, $\beta_D = 1.0$, $\alpha_D = -1.0$, $\gamma_D = 0.5$, $\omega_D = 1.2$ and initial state $(x_D, \dot{x}_D) = (1, 0)$.

Rössler attractor. The three-dimensional Rössler system reads

$$\begin{aligned} \dot{x}_R &= -\,(y_R + z_R),\\ \dot{y}_R &= x_R + a_R\,y_R,\\ \dot{z}_R &= b_R + z_R\,(x_R - c_R), \end{aligned} \tag{4}$$

with parameters $a_R = 0.2$, $b_R = 0.2$, $c_R = 5.7$ and initial state $(x_R, y_R, z_R) = (0, -6.78, 0.02)$.

Integration and data splitting. Both systems are integrated with a fixed time step $\Delta t = 0.01$ for $10,000$ samples, corresponding to 100 Lyapunov times ($\tau_D = 8.37$ for Duffing, $\tau_R = 13.83$ for Rössler). The first 50 Lyapunov times form the training/validation set; the remainder forms the test set.

3 Results and discussion

In this section we open the discussion by first showing how the recycled validation scheme reshapes the hyperparameter landscape, highlighting the optimal region for spectral radius and input scaling stand out more clearly than with a standard one-shot train/validation split. We then explore how much the forecast quality drifts when the reservoir weights are redrawn - an unavoidable source of randomness in model-free ESNs. Finally, using the Duffing and Rössler benchmarks, we gauge how long the network can shadow a chaotic trajectory (measured in Lyapunov times) before phase errors inevitably accumulate.

Figure 2 shows the Gaussian–process reconstruction of the validation loss ($-\log_{10}$ MSE) in the two-dimensional space spanned by spectral radius ρ_{target} and input scaling s_{inp}. Panels correspond to two independent random seeds for the reservoir matrices; the same trend (not shown for brevity) is observed for other seeds.

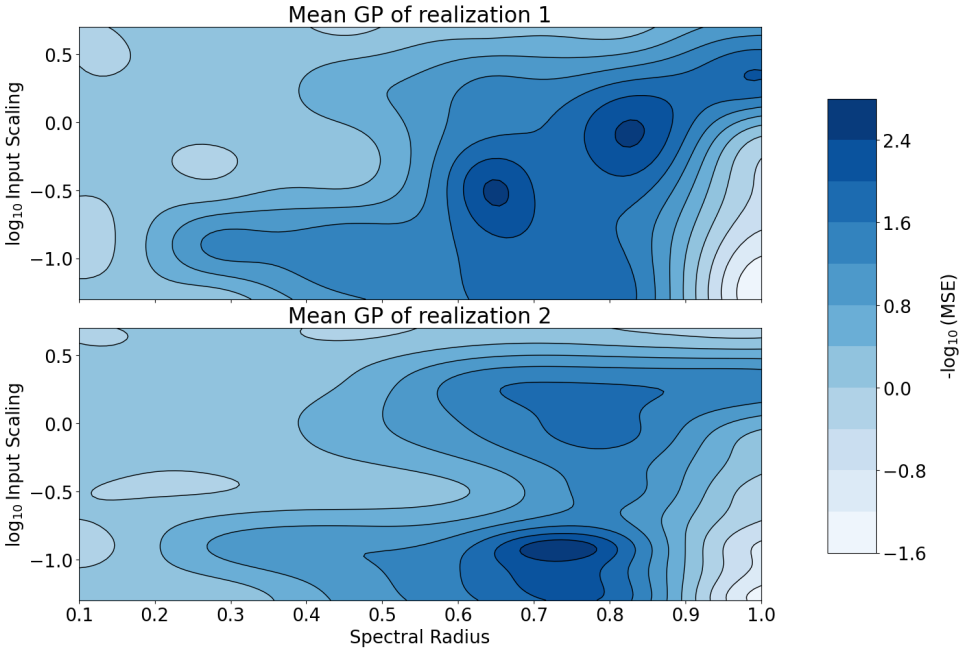


Figure 2. Realizations of the hyperparameters space for two different initializations.

The hyperparameter surface obtained with recycled validation presents a remarkably coherent shape across the random realisations. In every map a broad, flat minimum spans spectral radius values $\rho_{\text{target}} \approx 0.7 - 0.9$ and input scaling factors $\log_{10} s_{\text{inp}} \approx -0.7 - 0.2$. This observation corroborates the empirical guideline that reservoirs operating just below the stability threshold maximise short-term memory while avoiding divergent dynamics [14].

Although the overall topology is preserved, the exact location of the global minimum shifts with the random seed: the best value of $-\log_{10}\text{MSE}$ fluctuates by about 20%. Hence no single "gold-standard" hyperparameter pair exists; practitioners requiring high reliability should average predictions over an ensemble of independently initialised reservoirs, accepting a small increase in computation time for a substantial gain in robustness.

A notable benefit of the recycled validation scheme is its ability to sharpen the loss landscape. A conventional train/validation split (not shown) yielded a noisier, flatter surface that concealed the optimal ridge. By re-using the training trajectories in k distinct folds, recycled validation extracts more information from limited data and highlights the sweet-spot even when only a few Lyapunov times are available [9].

Figures 3 and 4 compare one closed-loop ESN forecast (red dashed) with the ground truth (black) for the Duffing and Rössler systems, respectively. The network hyperparameters selected by the Bayesian optimization are $\rho_{target,D} = 0.657$ and $s_{inp,D} = 0.135$ for the Duffing oscillator, and $\rho_{target,R} = 0.640$ and $s_{inp,R} = 1.990$ for the Rössler attractor.

The state variables of the Duffing oscillator, x_D and $\dot{x}_D$, remain phase-synchronised with the reference trajectory up to $\approx 8\tau_D$. Beyond this horizon, amplitude and phase errors diverge, yet the qualitative double-well switching persists. The mean–squared error on the test segment is $\text{MSE}_D = 0.001$.

For the Rössler flow the ESN follows the cyclical (x_R, y_R) motion and the spiking z_R bursts up to $\approx 11\tau_R$, after which phase drift becomes evident. The longer horizon reflects the lower maximal Lyapunov exponent of the Rössler system relative to Duffing. The test error is $\text{MSE}_R = 0.010$, 30% lower than the validation error, demonstrating that recycled validation does not over-fit to training noise.

The Echo State Network predicts the Duffing and Rössler systems for approximately eight and eleven Lyapunov times, respectively—substantially longer than the $\sim 5\tau$ horizon documented for classical model-free ESNs tunned with a simple hold-out split [7]. This improvement can be attributed to two factors: (i) meticulous hyperparameter tuning via the recycled validation scheme, and (ii) the injection of low-amplitude input noise during training, a practice known to enhance robustness in chaotic prediction [17].

Despite the longer horizon, performance remains sensitive to the random reservoir weights. Across 25 independent seeds the forecasting horizon exhibits a standard deviation at the threshold of 0.5 of $\pm 1.98\,\tau_R$ for Rössler attractor and $\pm 2.59\,\tau_D$ for Duffing oscillator, signalling that a single-reservoir deployment entails non-negligible risk. Consistent with Jaeger's original advice, we therefore recommend either averaging over a small ensemble of reservoirs or employing structured weight matrices (e.g. fixed random orthogonals) to curb variance without inflating training cost.

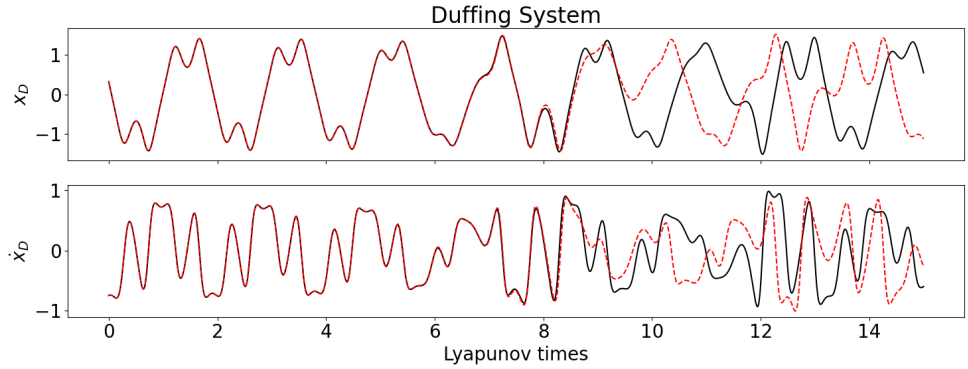


Figure 3. Chaotic time series generated by the ESN trained with the Duffing system data. The generated time series is in red and the ground truth is in black.

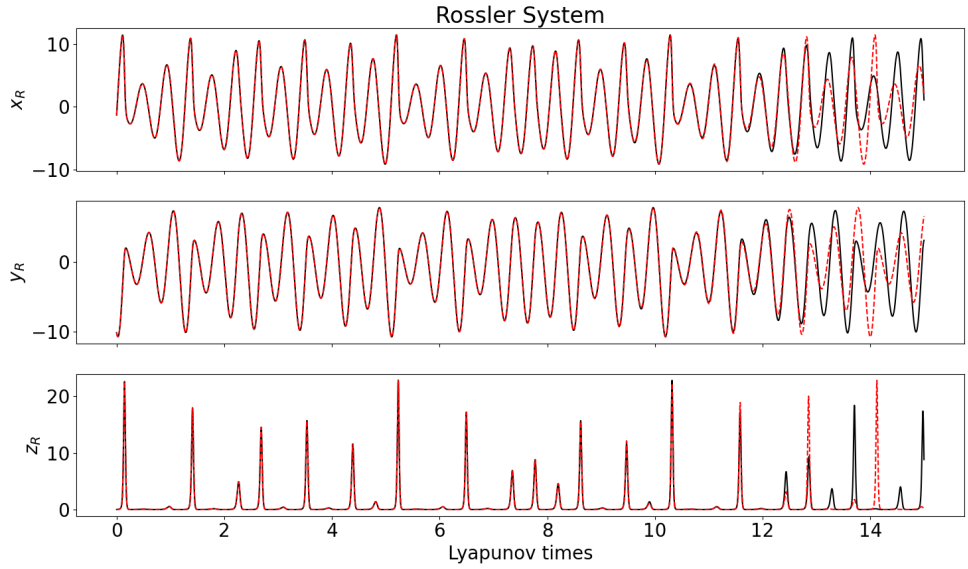


Figure 4. Chaotic time series generated by the ESN trained with the Rössler system data. The generated time series is in red and the ground truth is in black.

Training a network with $N_{\text{res}} = 1\,000$ internal units and ridge regression completes in 2.1 s on a standard laptop CPU—about two orders of magnitude faster than back-propagated recurrent networks of comparable size [18]. Such speed, coupled with the minimal memory footprint, reinforces the appeal of ESNs for real-time or embedded applications where computational resources are scarce.

The forecast error grows sharply once the horizon is exceeded; beyond that point, reproducing the *statistical* properties of the attractor (e.g. power spectra or Lyapunov exponent spectra) may require generalized readouts [19] or physics-informed architectures [20]. Furthermore, the present study is confined to two low-dimensional chaotic oscillators. Extending the analysis to high-dimensional flows—such as turbulent fluid dynamics or climate models—will be an important direction for future work.

4 Conclusions

An Echo State Network with recycled strategy forecasts the Duffing and Rössler systems respectively for roughly eight and eleven Lyapunov times, surpassing the $\sim 5\tau$ horizons of standard hold-out ESNs while training in seconds. The gain arises from a sharp hyperparameter ridge near $\rho \approx 0.8$ and moderate input scaling. Because skill fluctuates across random initialisations, small ensembles or structured reservoirs are recommended. These findings position ESNs as fast, lightweight forecasters for low-dimensional chaotic dynamics.

Acknowledgements. The authors express their gratitude for support received from Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES) - Finance Code 001, National Council for Scientific and Technological Development (CNPq), grant 305476/2022-0, and the Carlos Chagas Filho Research Foundation of Rio de Janeiro State (FAPERJ), grants: 211.037/2019, and 204.477/2024.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] C. D. Schuman and et al. Opportunities for neuromorphic computing algorithms and applications. *Nature Computational Science*, vol. 2, pp. 10–19, 2022.
[2] H. Jaeger. The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. Tech. Rep. 148, German Nat. Res. Center for Inf. Technol. GMD, Bonn, Germany, 2001.
[3] W. Maass, T. Natschläger, and H. Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, vol. 14, pp. 2531–2560, 2002.
[4] D. Verstraeten, B. Schrauwen, M. D'Haene, and D. Stroobandt. The unified reservoir computing concept and its digital hardware implementations. In *Proceedings of the 2006 EPFL LATSIS Symposium*, pp. 139–140, Lausanne. EPFL, 2006.
[5] L. Grigoryeva and J.-P. Ortega. Echo state networks are universal. *Neural Networks*, vol. 108, pp. 495–508, 2018.
[6] A. G. Hart, J. L. Hook, and J. H. Dawes. Echo state networks trained by tikhonov least squares are l2 (μ) approximators of ergodic dynamical systems. *Physica D: Nonlinear Phenomena*, vol. 421, pp. 132882, 2021.
[7] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Physical Review Letters*, vol. 120, n. 2, pp. 024102, 2018a.
[8] J. Pathak, A. Wikner, R. Fussell, S. Chandra, B. R. Hunt, M. Girvan, and E. Ott. Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 28, n. 4, pp. 041101, 2018b.
[9] A. Racca and L. Magri. Robust optimization and validation of echo state networks for learning chaotic dynamics. *Neural Networks*, vol. 142, pp. 252–268, 2021.
[10] X. Hinaut. *Réseau de neurones récurrent pour le traitement de séquences abstraites et de structures grammaticales, avec une application aux interactions homme-robot*. Diplôme de doctorat en la discipline de neurosciences, École Doctorale Neurosciences et Cognition, Université Claude Bernard Lyon 1 Campus de Gerland 50, Avenue Tony Garnier 69366 Lyon Cedex 07 France, 2013.
[11] B. Schrauwen, D. Verstraeten, and J. M. V. Campenhout. An overview of reservoir computing: theory, applications and implementations. In *The European Symposium on Artificial Neural Networks*, 2007.
[12] S. Haykin. *Neural Networks and Learning Machines*. Pearson Education, Inc., New Jersey, 3 edition, 2008.
[13] I. B. Yildiz, H. Jaeger, and S. J. Kiebel. Re-visiting the echo state property. *Neural Networks*, vol. 35, pp. 1–9, 2012.
[14] M. Lukosevicius. *A Practical Guide to Applying Echo State Networks*, pp. 659–686. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
[15] F. Huhn and L. Magri. Learning ergodic averages in chaotic systems. In *Computational Science – ICCS 2020*, pp. 124–132, Cham. Springer International Publishing, 2020.
[16] J. Friedman, T. Hastie, and R. Tibshirani. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer, second edition. Preface to the Second Edition, September 30, 2008, 2008.
[17] P. Vlachas, J. Pathak, B. Hunt, T. Sapsis, M. Girvan, E. Ott, and P. Koumoutsakos. Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics. *Neural Networks*, vol. 126, pp. 191–217, 2020.
[18] S. Shahi, F. H. Fenton, and E. M. Cherry. Prediction of chaotic time series using recurrent neural networks and reservoir computing techniques: A comparative study. *Machine Learning with Applications*, vol. 8, pp. 100300, 2022.
[19] A. Ohkubo and M. Inubushi. Reservoir computing with generalized readout based on generalized synchronization. *Scientific Reports*, vol. 14, n. 30918, 2024.
[20] N. Doan, W. Polifke, and L. Magri. Physics-informed echo state networks. *Journal of Computational Science*, vol. 47, pp. 101237, 2020.