Nonlinear Transformation in Reservoir Computing for Lorenz-96 Equations

100060615
Khalifa University
Abu Dhabi, United Arab Emirates

***Abstract*—Reservoir computing is state of the art in solving dynamical systems using machine learning. In this paper, we investigate the impact of nonlinear states transformation on the accuracy of short-time prediction of Lorenz-96 equations. A preliminary numerical experiment showed no significant difference between the performance of the nonlinear transformation used in the simulation.**

I. Introduction

The partial differential equations are widely used for modelling physical systems. A classical approach to solve this equations is by discretizing the methods using various discretization schemes such as finite difference method, finite element method, finite volume method, etc. Recently there has been growing interest in solving the partial differential equations based on data-driven approaches (see survey in [1]). Among reasons to consider machine learning to solve physical problems includes ability to incorporate measurement data/knowledge model in the model [2] and speed up computations by modelling computation expensive task and using machine learning approach as closure model to describe missing dynamics [3]. However, applications of machine learning still pose many challenges and skepticism, such as the generalizability of the prediction, interprebility of the model, the incorporating the physical knowledge in the system [4] .

Reservoir computing is one of the machine learning approach that is used to solve partial differential equations involving time and spatial variables. It has shown superiority in simulating dynamical systems compared to other machine learning approaches [5], [6], [7]. Recently, there have been growing interest to tackle problems involving partial differential equations [6], [2], [8]. Nonlinear transformation was introduced in [6], [2] to enrich the nonlinear feature of the reservoir. The paper aims to compare the non-linear transformations proposed in [6] in solving Lorenz-96 equations which is a prototype for a chaotic spatiotemporal system.

II. Method

Reservoir Computing (RC) can be taught as a subset of recurrent neural networks, however the reservoir parameters are fixed, and only the readout parameters are trained. In this research, we follow the adjustment for spatiotemporal models [2], which split the spatial data into overlapping domains and train multiple reservoirs for each data chunk. The reasoning for splitting the data is that the exchange of information between two very far seperated spatial grid is not immediate, thus creating reservoir state modelling a connection between far-away points is not physical and a waste of computation effort. Thus partitioning the grid while maintaining the exchange of information between reservoirs, render the size of reservoir and the amount of training data manageable.

To be precise, let a measurement data consist of spatiotemporal data $\mathbf{U} = (u_{ij})$, with temporal index $1 \leq i \leq N_\tau$, and spatial index $1 \leq j < N_\xi$. The periodic spatial grids are splitted into non-overlapping regions $\mathbf{s}^{(k)} = \left(s_{ij}^{(k)}\right)$, with $1 \leq j \leq n_p, 1 \leq k \leq M$ with n_p is the number of nodes at each partition and M is the number of partition. Remark that $M \times n_p = N_\xi$. Associated with each partition $\mathbf{s}^{(k)}$ an overlapping domain $\mathbf{v}^{(k)}$ with l nodes overlap buffer regions at each most left and right nodes , thus $\mathbf{v}^{(k)}$ has $n_p + 2l$ spatial grid. Furthermore, the data is temporally split into training and testing sets. The overlapping domain serves as input for each reservoir enabling exchange of information between neighbouring reservoir, while the non-overlapping domains will act as the output.

Listening process corresponds to constructing reservoirs and evolve them based on input and previous states. Specifically let $r_i^{(k)} \in \mathbb{R}^M$ with $M \gg n_p + 2l$ denote the state of reservoir k at time index i. The reservoir dynamics is given by: $r_{i+1}^{(k)} = \tanh\left(W^{(k)} r_i^{(k)} + W_{in}^{(k)} v_i^{(k)}\right)$. The matrix $W^{(k)}$ is an adjecency matrix which set to ensure "Echo States Property"[7]. In a simplified term, this condition expresses that the previous state's effect to the future state should vanish gradually. Practically, the echo state property is satisfied by scaling $W^{(k)}$, such that $W^{(k)}$ is contractive i.e. the spectral radius of matrix ρ satisfies $\rho < 1$. Furthermore, the adjecency matrix is set to be sparsely connected and random to ensure that the reservoir is rich enough for learning phase. The input coupling matrix $W_{in}^{(k)}$ is a sparse matrix whose entries magnitude are not more than σ.

The learning phase involves finding an output matrix $W_{out}^{(k)}$ such that the output $\tilde{s}_i^{(k)} = W_{out}^{(k)} \hat{r}_i^{(k)}$ minimize a cost functional. The modified state $\hat{r}_i^{(k)}$ is a nonlinear modification aimed to further enhance the nonlinearity feature of the reservoir[6], [2]. The cost function is regularized using Tikhonov regularization and defined as

$$E^{(k)} = \min_{W_{out}^{k}} \sum_{i=1}^{N_{train}} \left\| W_{out}^{(k)} \hat{r}_i^{(k)} - s_i^{(k)} \right\|_2^2 + \beta \left\| W_{out}^{(k)} \right\|^2 \quad (1)$$

which can be solved by any linear least square algorithm.

The prediction phase updates the reservoir nodes in the similar manner as the training phase except that a feedback of the reservoir output replaces the input vector $v_i^{(k)}$. Noting that $v_i^{(k)}$ consists of overlapping domains, the output should be constructed by combining readout from the neighbouring reservoir $y_i^{(k-1)}, y_i^{(k)} y_i^{(k+1)}$. The output is generated by multiplying the learned output matrix and the updated reservoir state. These processes are repeated until the desired prediction horizon.

III. Numerical Experiment

The Lorentz 96 system is a coupled ordinary differential equation that is used as an initial model for atmospheric dynamics or testing a machine learning algorithm for chaotic spatiotemporal system [8], [6]. The model is defined as coupled ordinary differential equations:

$$\frac{dX_j}{dt} = -X_j + X_{j-1}X_{j+1} - X_{j-1}X_{j-2} + F$$

with the spatial variables X_j, $1 \leq j \leq N$,$X_{j+N} = X_j$.

The training data was generated by using 4th order Runge Kutta time integrator with $\Delta t = 0.01$ and choosing $F = 8$. The initial condition at every grid is a random number between 0 and 1. The largest Lyapunov exponent for this system is $\Lambda = 1.4$. The hyperparameter used in reservoir computing is given in Table I which are quite similar with the hyperparameter in a similar problem used in [8]. The spatially averaged error at time index i is given by: $e_i = \frac{\|\tilde{u}_i - u_i\|}{\|u_i\|}$

Parameter	ρ	σ	β	M	l	D	N	N_{train}
Value	0.6	0.1	10^{-4}	20	2	5000	40	80000

TABLE I
Hyperparameter for Lorenz 96

There are three non-linear transformation considered in this paper. The NLT1 was proposed in [2] and transforms r_{ij} for odd j into r_{ij}^2. The reference [6] considered alternative transformations, which we denote as NLT2 and NLT3 with the odd entries transformed into $r_{i(j-1)}r_{i(j-2)}$ and $r_{i(j-1)}r_{i(j+1)}$,respectively.

The upper graph of Figure 1 shows the time trace of prediction using three non-linear transformations at X_{20}. The picture shows that the three non-linear transformations have comparable short term prediction capabilities. A similar pattern could be observed in the lower figure, which shows the spatial averaged error of the three nonlinear transformations. Even though apparently NLT1 gives double error compared to the rest of non-linear transformation until $\Lambda t = 1$, the error is not very much different at 2 Lyapunov times. Considering that the Lorenz-96 is a chaotic system, a particular realization of the prediction is not enough to conclude which non-linear transformation gives a better result. A plausible approach to get a conclusive pattern is by repeatedly doing the prediction phase by resetting the reservoir states and feed with real data and then switching to prediction phase and take an average over the different realizations of the systems [8].

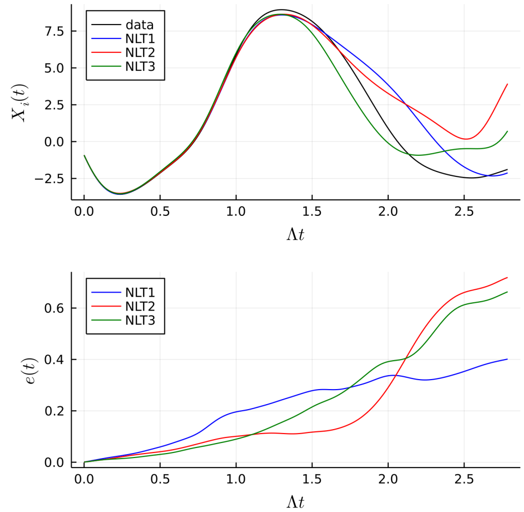

Fig. 1. (upper): comparison of time trace prediction at X_{20}. (lower): spatially average error over the whole spatial grid

IV. Conclusions

Reservoir computing is a data-driven method suitable for solving dynamical systems. The method accurately gives short time prediction over Lorenz-96 equations with RMSE less than 0.2 for one Lyapunov times. The non-linear transformations considered in the study yield similar performance in the prediction phase. However, further research should be done, such as obtaining statistical inferences from the data.

References

[1] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine Learning for Fluid Mechanics," *Annual Review of Fluid Mechanics*, vol. 52, no. 1, pp. 477–508, jan 2020. [Online]. Available: https://doi.org/10.1146/annurev-fluid-010719-060214

[2] 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, no. 2, p. 24102, 2018. [Online]. Available: https://doi.org/10.1103/PhysRevLett.120.024102

[3] R. Vinuesa and S. L. Brunton, "The Potential of Machine Learning to Enhance Computational Fluid Dynamics," pp. 1–13, 2021. [Online]. Available: http://arxiv.org/abs/2110.02085

[4] M. P. Brenner, J. D. Eldredge, and J. B. Freund, "Perspective on machine learning for advancing fluid mechanics," *Physical Review Fluids*, vol. 4, no. 10, p. 100501, 2019. [Online]. Available: https://doi.org/10.1103/PhysRevFluids.4.100501

[5] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Computer Science Review*, vol. 3, no. 3, pp. 127–149, 2009.

[6] A. Chattopadhyay, P. Hassanzadeh, and D. Subramanian, "Data-driven predictions of a multiscale Lorenz 96 chaotic system using machine-learning methods: Reservoir computing, artificial neural network, and long short-term memory network," *Nonlinear Processes in Geophysics*, vol. 27, no. 3, pp. 373–389, 2020.

[7] H. Jaeger and H. Haas, "Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication," *Science*, vol. 304, no. 5667, pp. 78–80, 2004.

[8] J. Pathak and E. Ott, "Reservoir Computing for Forecasting Large Spatiotemporal Dynamical Systems," in *Reservoir Computing*. Springer Nature Singapore, 2021, pp. 117–138.