Reservoir Computing on Non-Linear Schrödinger Equation

A. Hasmi[1]

[1]PhD Student of Mathematics
Khalifa University

4 February 2022

Outline

1. Theory
2. Result
   - Verification with Lorenz-96
   - Non-Linear Schrödinger

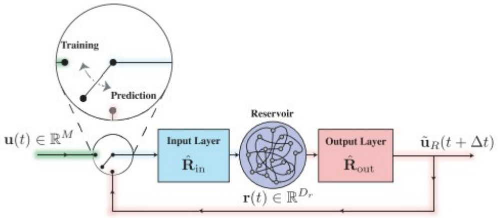
Data Driven Method for Partial Differential Equation

Frame subtitles are optional. Use upper- or lowercase letters.

- First Principle Methods: Discretize PDEs using Numerical Schemes: Finite Difference, Finite Element, Finite Volume, Spectral Methods, etc
- Data Driven Method: Incorporate Observation/Data to simulate PDEs: Physics Informed Neural Network,
- Pros/Reasoning Using DDM:
  - Learning Hidden Physics from Data (Turbulence, Wall Model)
  - Reducing Computational Cost (Reduced Order Method,)
  - Data Assimilation with observation data[4]

Reservoir Computing

- Recurrent Neural Network but only train the output layer [2]
- Fast training as only output parameters need to optimized [1, 2]
- Does not suffer from vanishing gradient problem



Reservoir Computing for Spatiotemporal Problem

- By locality of interaction between grids, we can split the spatial domain into several reservoirs and train them independently
- internal states updates:

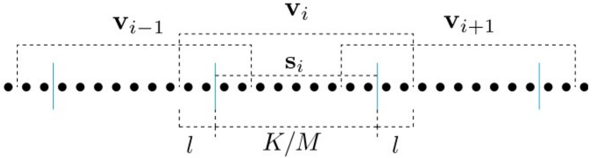
$$r_{i+1}^{(k)} = \tanh\left(W^{(k)} r_i^{(k)} + W_{in}^{(k)} v_i^{(k)}\right)$$

- output readout

$$y_i^{(k)} = W_{out}^{(k)} \hat{r}_i^{(k)} \qquad \hat{r}_{ij}^{(k)} = \begin{cases} r_{ij}^{(k)} & j \text{ is odd} \\ \left(r_{ij}^{(k)}\right)^2 & j \text{ is even} \end{cases}$$

- Training Objectives

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



Prediction Steps

1. Compute $y_i^{(k)} = W_{out}^k \hat{r}_i^{(k)}$
2. Construct $\tilde{v}_i^{(k)}$ based on $y_i^{(k-1)}, y_i^{(k)} y_i^{(k+1)}$
3. Update the reservoir states: $r_{i+1}^{(k)} = \tanh\left(W^{(k)} r_i^{(k)} + W_{in}^{(k)} \tilde{v}_i^{(k)}\right)$

Outline

Lorenz 96

- Periodic and Chaotic ODE
- Dummy equation for atmospheric model
- Equations given by:

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

- Training data generated by solving equation using RK orde 4 method

Result

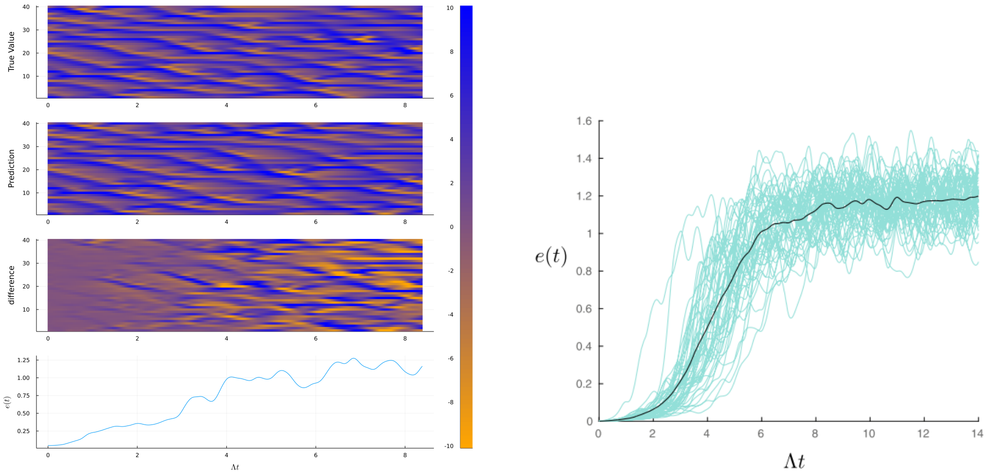


Figure: (Left) Simulation result: True Value based RK4 solution, Prediction based on Reservoir Computing, Difference: (True Value - Prediction), e: Root Mean Square (Right) Reference RMS based

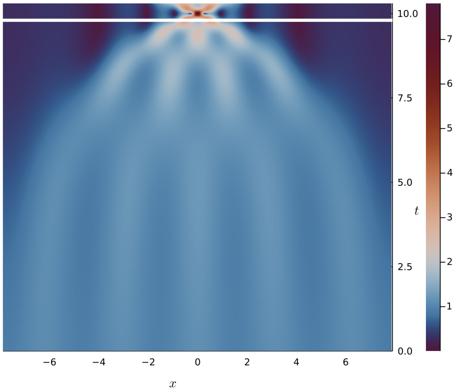
Outline

Non-Linear Schrödinger

- Equation given by:

$$i\frac{\partial\psi}{\partial\tau}+\frac{1}{2}\frac{\partial^2\psi}{\partial\xi^2}+|\psi|^2\,\psi=0$$

- Can be used to modeled rogue wave
- The figure shows training and testing data for NLS. The limit of training data is given by whiteline



Simulation Hyperparameter

Parameter	Explanation	Value
ρ	Spectral Radius of W	0.6
σ	maximum elementwise magnitude of W_{in}	0.1
β	Regularization Parameter	10^{-4}
M	Number of reservoirs	32
l	Number of overlap grid	6

NLT1

$$\hat{r}_{ij}^{(k)} = \begin{cases} r_{ij}^{(k)} & j \text{ is odd} \\ \left(r_{ij}^{(k)}\right)^2 & j \text{ is even} \end{cases}$$

NLT2

$$\hat{r}_{ij}^{(k)} = \begin{cases} r_{ij}^{(k)} & j \text{ is odd} \\ \left|r_{ij}^{(k)}\right|^2 r_{ij}^{(k)} & j \text{ is even} \end{cases}$$

Result: NLT1

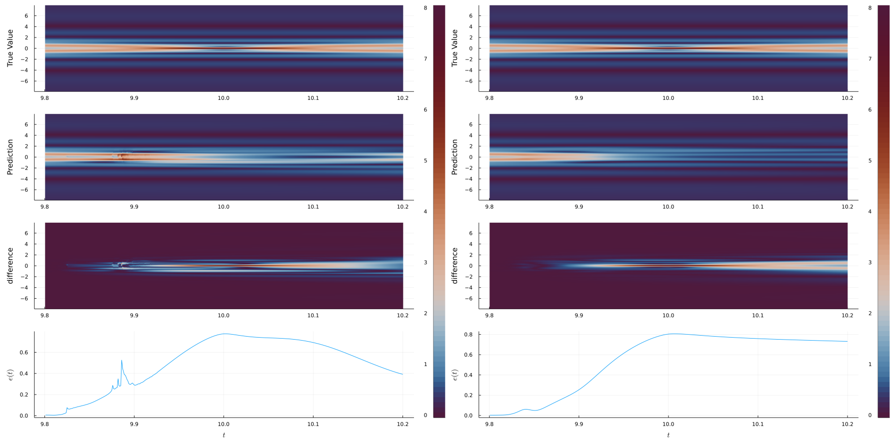


Figure: Simulation using two different time discretization in training data: 0.01 (9800 Nt) in the left figure, 0.005 (19600Nt) in the right figure. Both of using 4000 reservoir nodes

Result: NLT2

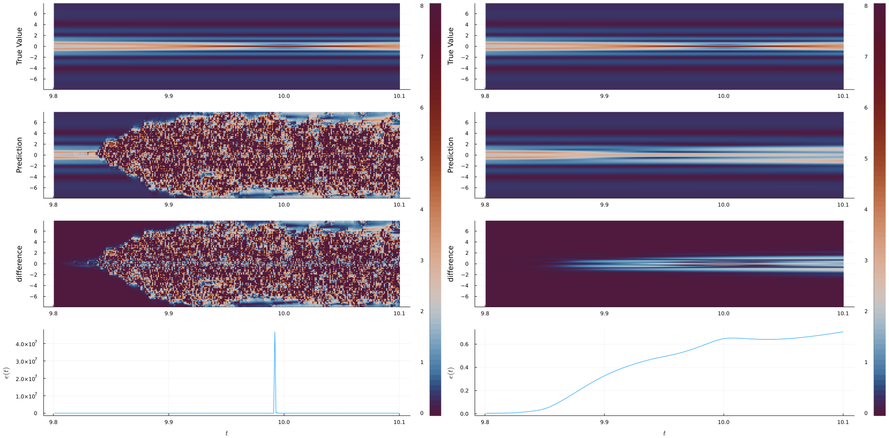


Figure: Result using different number of node in reservoir $\Delta t = 0.01$: left (4000), right (800)

Result: NLT2

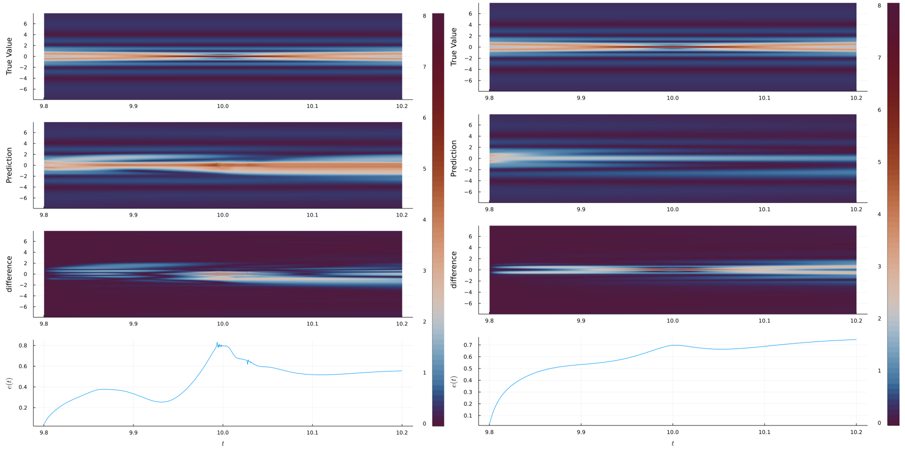


Figure: Result using different number of node in reservoir using $\Delta t = 0.005$: left (800), right (1600)

Summary

- Enhancing the RC with NLT promote the expressibility of the RC
- Too much node reservoir can lead to completely wrong prediction (overfitting?)
- Finding the suitable hyperparameter for RC is not trivial
- Outlook
  - Experiment with different hyperparameter?
  - ??

For Further Reading I

- Ashesh Chattopadhyay, Pedram Hassanzadeh, and Devika 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, 27(3):373–389, 2020.
- Mantas Lukoševičius and Herbert Jaeger.
  Reservoir computing approaches to recurrent neural network training.
  Computer Science Review, 3(3):127–149, 2009.
- Jaideep Pathak and Edward Ott.
  Reservoir Computing for Forecasting Large Spatiotemporal Dynamical Systems.
  In *Reservoir Computing*, pages 117–138. Springer Nature Singapore, 2021.

For Further Reading II

Jaideep Pathak, Alexander Wikner, Rebeckah Fussell, Sarthak Chandra, Brian R. Hunt, Michelle Girvan, and Edward Ott.
Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model.
Chaos, 28(4), 2018.