Reservoir Computing on Non-Linear Schrödinger Equation

A. Hasmi[1]

[1]PhD Student of Mathematics
Khalifa University

7 March 2022

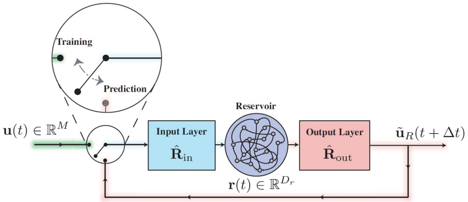
Data Driven Method for Partial Differential Equation

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[5]

Reservoir Computing

- Recurrent Neural Network but only train the output layer [4]
- Fast training as only output parameters need to optimized [1, 4]
- 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

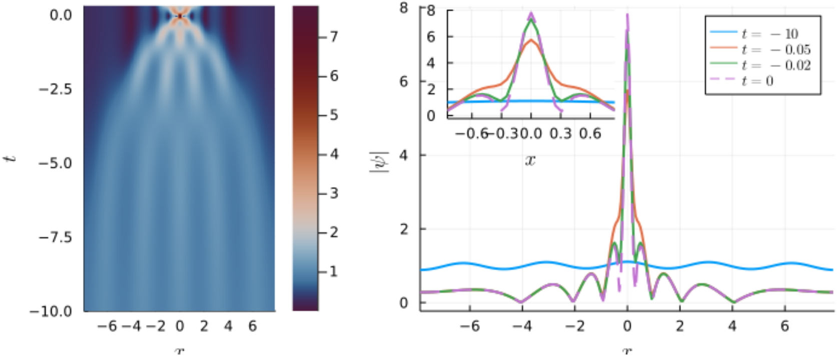
1. Compute $y_i^{(k)} = W_{out}^k \hat{r}_i^{(k)}$
2. Construct $\tilde{v}_i^{(k)}$ based on $y_i^{(k-1)}, y_i^{(k)} y_i^{(k+1)}$
3. Update the reservoir states: $r_{i+1}^{(k)} = \tanh\left(W^{(k)} r_i^{(k)} + W_{in}^{(k)} \tilde{v}_i^{(k)}\right)$

Outline

Training Data

- 5th order maximum intensity breather governed by NLS equation [2]

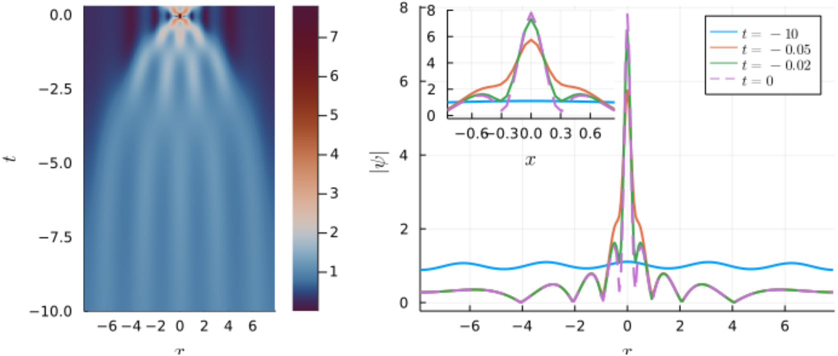
$$i\frac{\partial\psi}{\partial\tau}+\frac{1}{2}\frac{\partial^2\psi}{\partial\xi^2}+|\psi|^2\psi=0$$



Training Data

- 5th order maximum intensity breather governed by NLS equation [2]

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



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

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

NLT3

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

Assessing the Result

To measure the performance, we use H^1 norm (Sobolev space norm). Let $u \in H^1$, then the norm is defined as:

$$\|u\|_{H^1} = \left[\int |u|^2 \, dx\right]^{\frac{1}{2}} + \left[\int |du|^2 \, dx\right]^{\frac{1}{2}}$$

the error is calculated at $t = 10$ (the expected peak), with error defined as:

$$e = \frac{\left\|\hat{\psi} - \psi\right\|_{H^1}}{\|\psi\|_{H^1}}$$

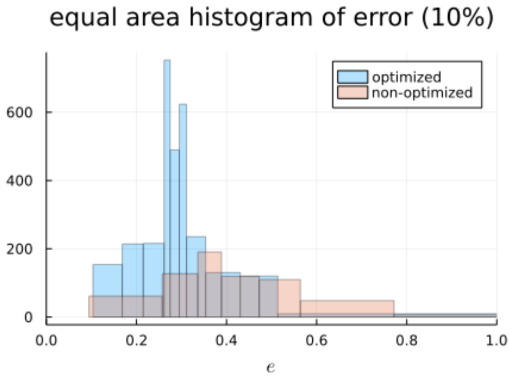
Hyperparameter Optimization

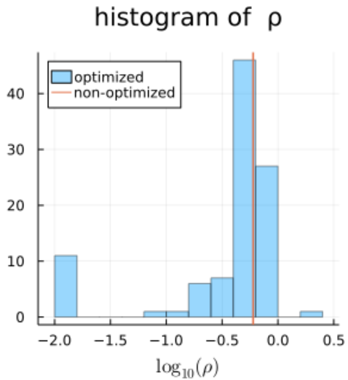
For each realization we train ESN with training data $(\psi)_{t=-10}^{t=-0.05}$, then calculate cross validation error:

$$e_{CV}\left(\hat{\psi};\rho\right)=\int_{-0.05}^{-0.02}\frac{\left\|\hat{\psi}-\psi\right\|_{H^1}}{\left\|\psi\right\|_{H^1}}dt$$

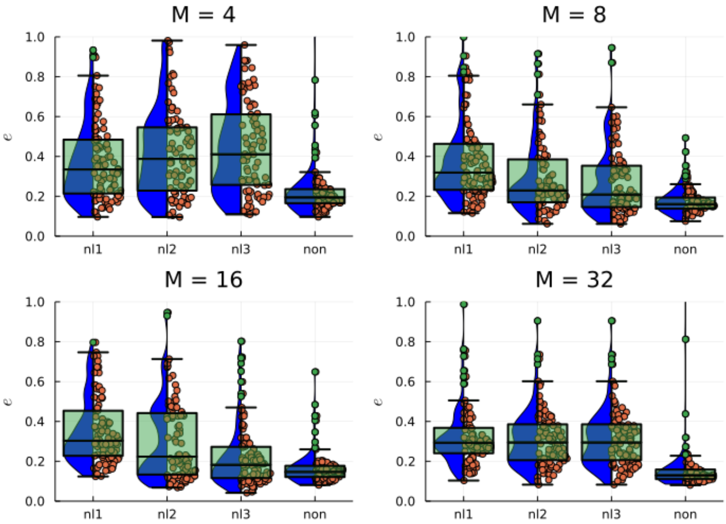
with $\log_{10}\rho\in[-2,0.4]$. Define $\tilde{\rho}=\text{argmax}_{\rho}e_{CV}$. Then train based on $(\psi)_{t=-10}^{t=-0.02}$ and $\tilde{\rho}$ and calculate e at $t=0$

Result: Optimized vs Non-Optimized Cases

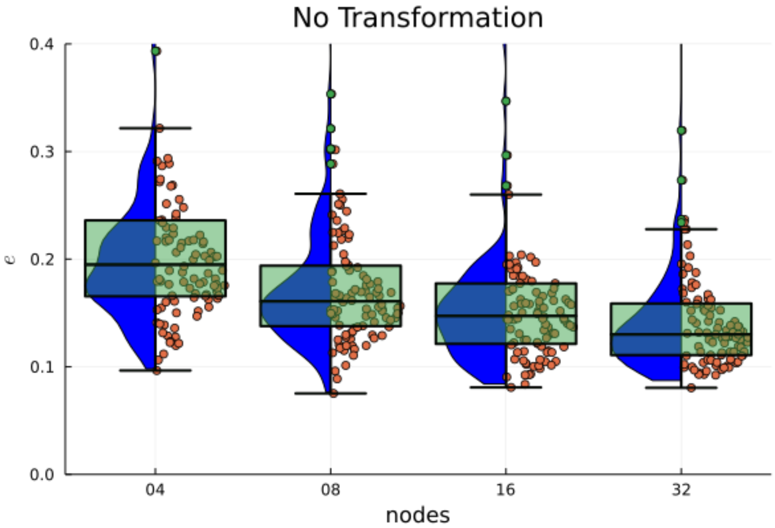




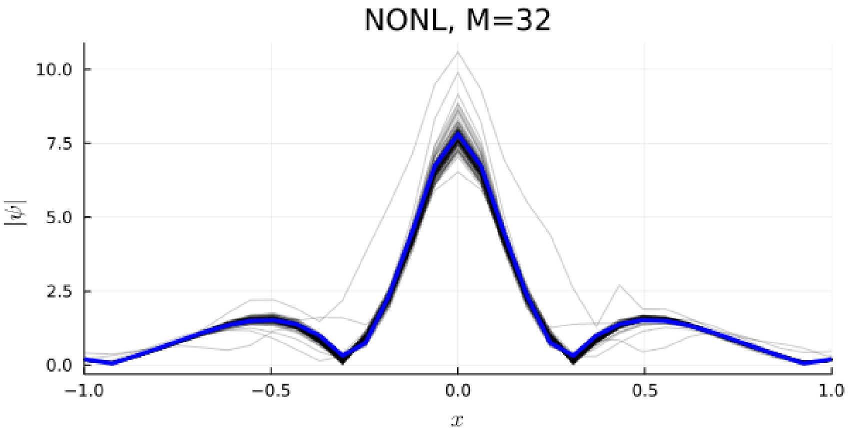
Result:Effect of Reservoir Number and NLT



Result: No Transformation



Result: NoNLT, M=32



Summary

- Spectral Radius Optimization significantly improve RC performances
- Increasing M improves RC result given optimized hyperparameter
- no non-linear transformation gives best result for NLS prediction

- Outlook
  - Can we improve optimization algorithm ? Bayesian Optimization ? [3, 6]
  - Further sensitivity studies: redo for reservoir nodes, training data length

For Further Reading I

[1] 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.

[2] Siu A. Chin, Omar A. Ashour, Stanko N. Nikolić, and Milivoj R. Belić.
Maximal intensity higher-order Akhmediev breathers of the nonlinear Schrödinger equation and their systematic generation.
Physics Letters, Section A: General, Atomic and Solid State Physics, 380(43):3625–3629, 2016.

[3] Francisco Huhn and Luca Magri.
Gradient-free optimization of chaotic acoustics with reservoir computing.
014402, 2021.

For Further Reading II

[4] Mantas Lukoševičius and Herbert Jaeger.
Reservoir computing approaches to recurrent neural network training.
Computer Science Review, 3(3):127–149, 2009.

[5] 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.

[6] Alberto Racca and Luca Magri.
Robust Optimization and Validation of Echo State Networks for learning chaotic dynamics.
Neural Networks, 142:252–268, 2021.