High Performance Computing and Reservoir Computing group

Akshat Verma
Chinchuthakun Worameth
Pongsakorn Chairatanakul
Moderators: Alexander Thomas Magro & Wentao Sun

Chaotic Processes

- A chaotic system or dynamical system are systems which appear random but are governed by complex mathematical equations.
- A famous example is the Lorenz System
- A slight change in initial condition can vastly change the state of system at a later point

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x),$$

$$\frac{\mathrm{d}y}{\mathrm{d}t} = x(\rho - z) - y,$$

$$\frac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z.$$

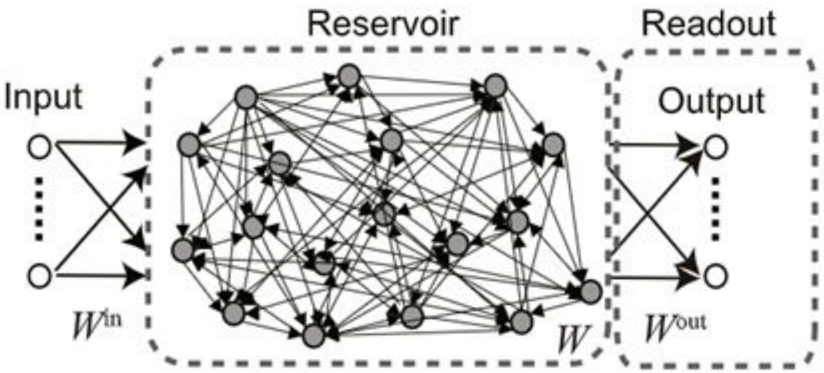
Lorenz System
Example of a chaotic system

Forecasting Chaotic processes

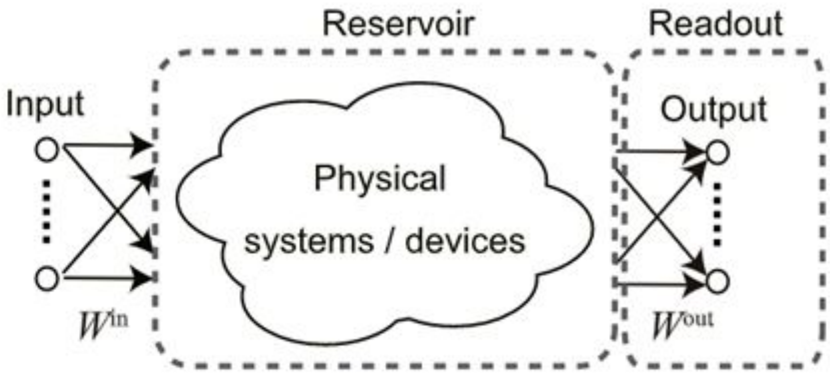
- The state of any chaotic process at a given time can be predicted by two main methods:
  - Using knowledge based model
  - Using machine learning and predicting from past states
- While using a knowledge based model, we use mathematical equations that have underlying knowledge of the mechanics or working of the system and then calculate the current state using the equation.
  - We don’t always know the exact equations/parameters.
- The machine learning method predicts state based on past states.
- As the ML model has no understanding of the system, the data required and computation cost to predict can be quite prohibitive.

What is Reservoir computing

- A state-of-the-art recurrent neural network (RNN) framework
- Different forms of reservoirs and are fixed when training.
- Input layer and reservoir layer are fixed after generated.
- Only train the output weight.



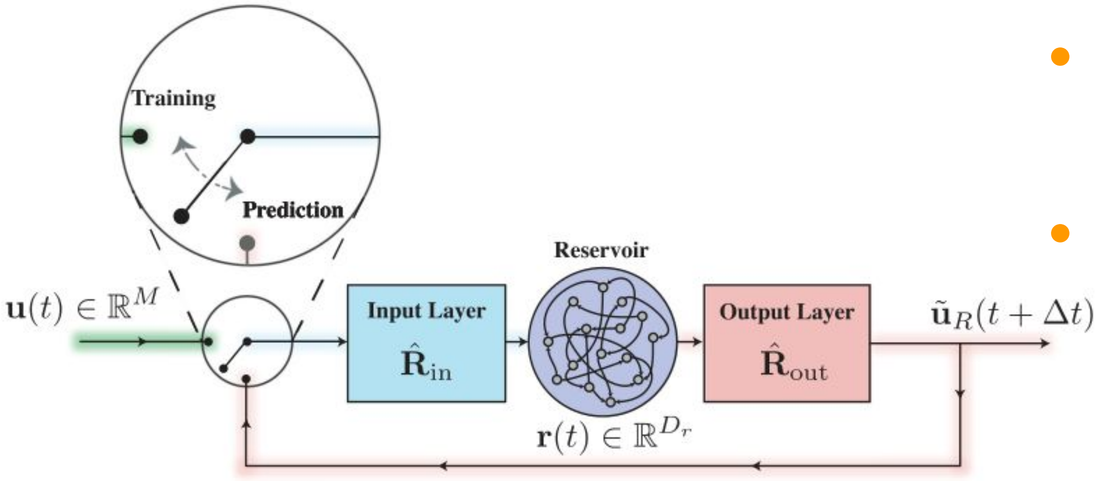
(a) Conventional RC



(b) Physical RC

Reservoir computing for close the loop predicting

- In training phrase, use input u(t) to train the model.
- In prediction phrase, use the prediction output to be the input.



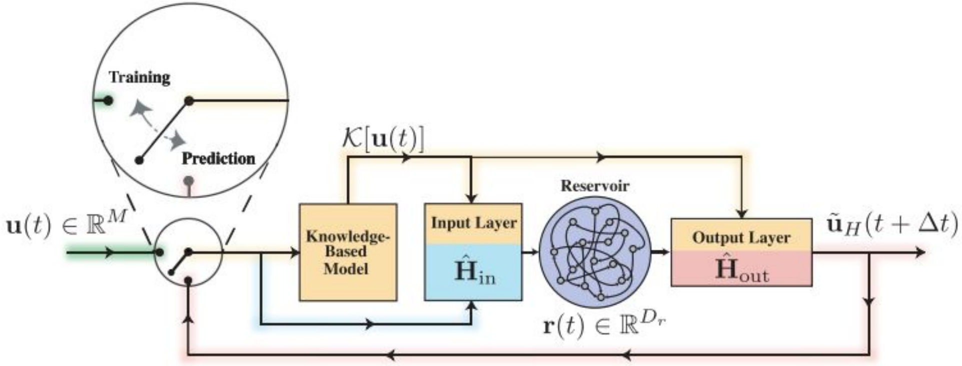
- As our selected task is a temporal data, RNNs can give the best output.
- Moreover, the training time for a regular RNN is very large compared to a Reservoir based RNN, so it makes sense to use reservoir model instead of traditional RNN.

Motivation for selecting this topic

- While using recurrent neural networks, after reaching a specific number of nodes, the accuracy in the output stagnates. This results in increasing the computational cost while the accuracy of model stays similar.
- Physics-informed neural network introducing the physical model in the NN can help with further enhancement.
- To overcome this, we incorporate a knowledge based model to get better result.

Hybrid Model

- In order to predict the state of such a chaotic system more efficiently, a hybrid model can be used using both knowledge based and Machine Learning



The hybrid model is inspired by the work in the paper "Hybrid forecasting of chaotic processes[1]"

In the hybrid model, we add an error parameter in the knowledge based model

[1] Pathak, J., Wikner, A., Fussell, R., Chandra, S., Hunt, B. R., Girvan, M., & Ott, E. (2018). Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *28*(4), 041101. https://doi.org/10.1063/1.5028373

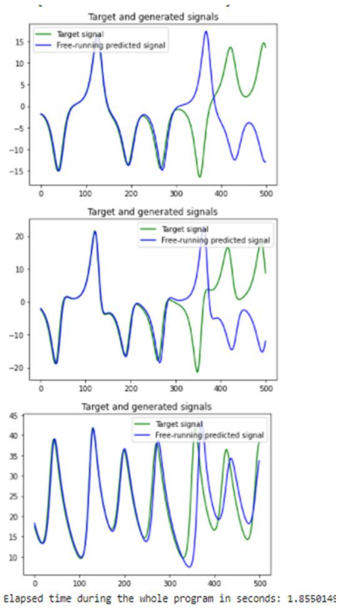
Our plan for the week

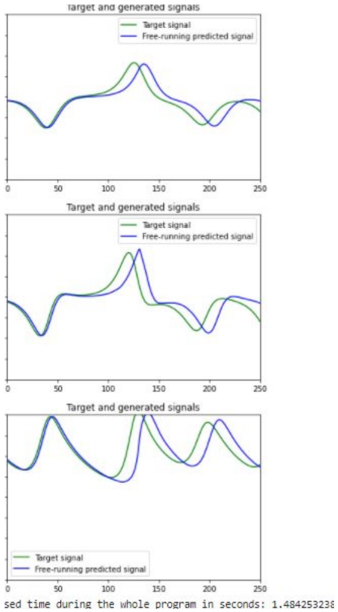
- Implement hybrid model on python
- Try to find a balance between physical and rnn
- Parallelize the computations if possible for both models
- Compare performance for models with different hyperparameters
- Improve on the reservoir model

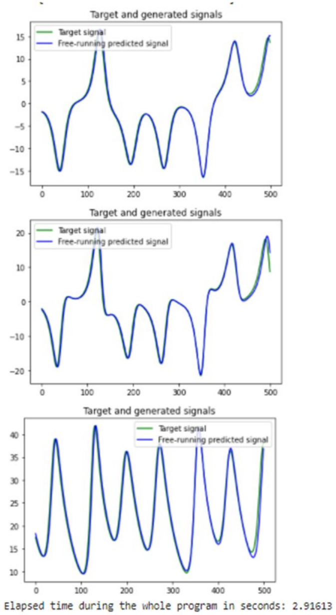
Performance comparison of hybrid model with reservoir only

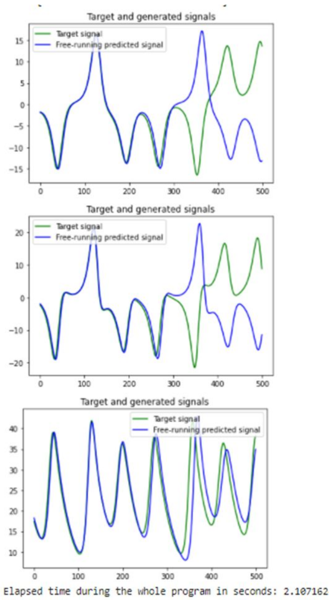
- Performance of both models are highly dependent on reservoir size
- As reservoir model relies only on the reservoir for prediction, we can expect it's performance to be directly related to the size of the reservoir
- For the hybrid model, which relies on both reservoir and knowledge based models, we can expect to see irregularities in the relationship of size of reservoir and performance

Side by side comparison









Hybrid model

Reservoir Model

Reservoir Size = 100

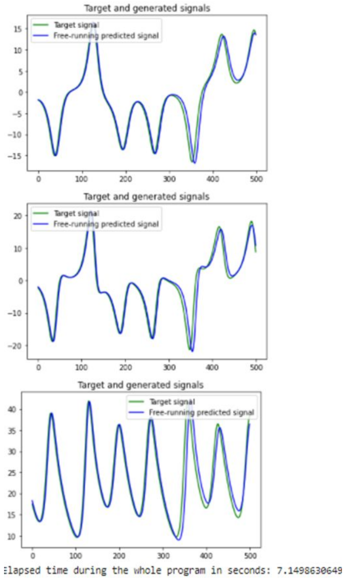
Hybrid model

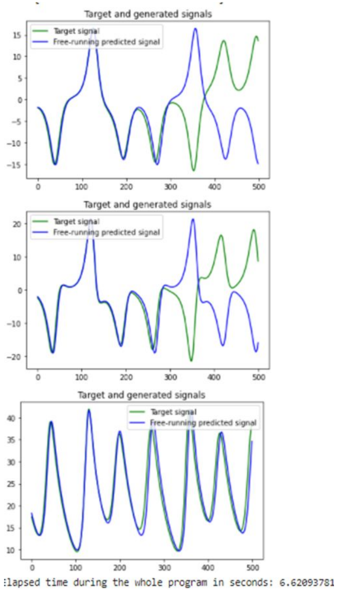
Reservoir Model

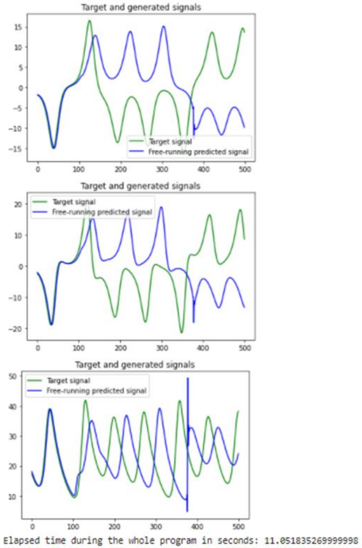
Reservoir Size = 300

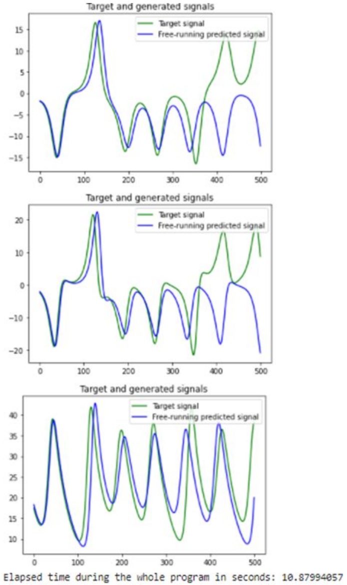
Side by side comparison

Perhaps overfitting here









Hybrid model | Reservoir Model | Hybrid model | Reservoir Model

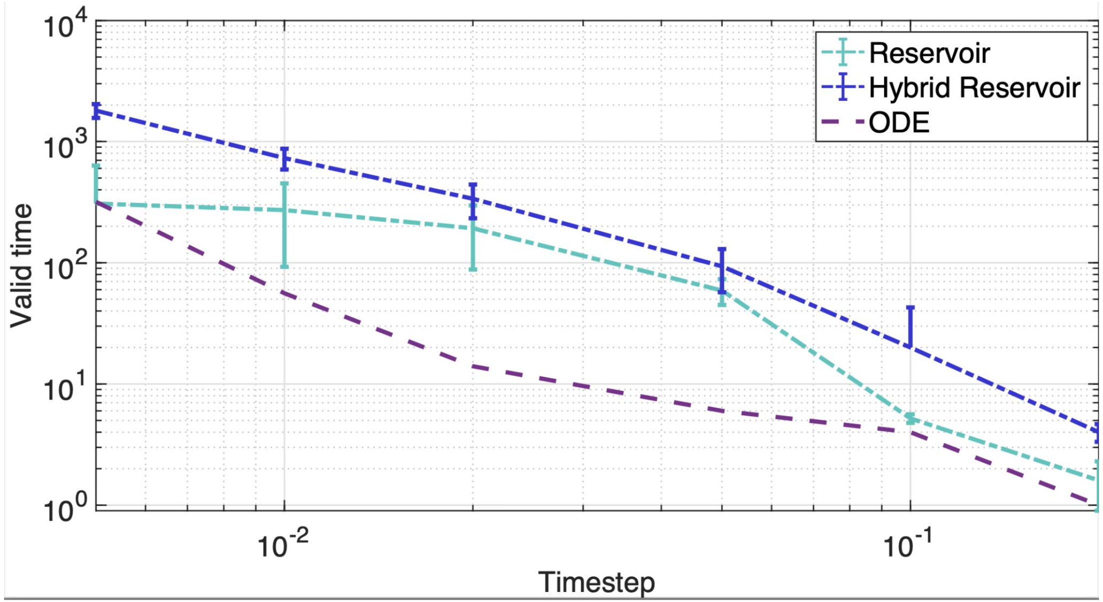
Reservoir Size = 700 | Reservoir Size = 1000

Conclusion from varying reservoir sizes

- The reservoir only model is faster for all cases
- For low number of reservoir nodes, the performance of hybrid model is good while that of reservoir model is mostly incorrect
- The Hybrid model performs best with an intermittent number of reservoir nodes
- For very high number of reservoir nodes, the performance of reservoir model is slightly decreased but that of hybrid model is greatly decreased
- In conclusion, hybrid model works great with an intermittent number of nodes in the reservoir

Quantitative enhancement

- Generate the dataset in a small timestep at 1e-3 (more accurate).
- Run the system in a large timestep (from 5e-3 to 5e-1)

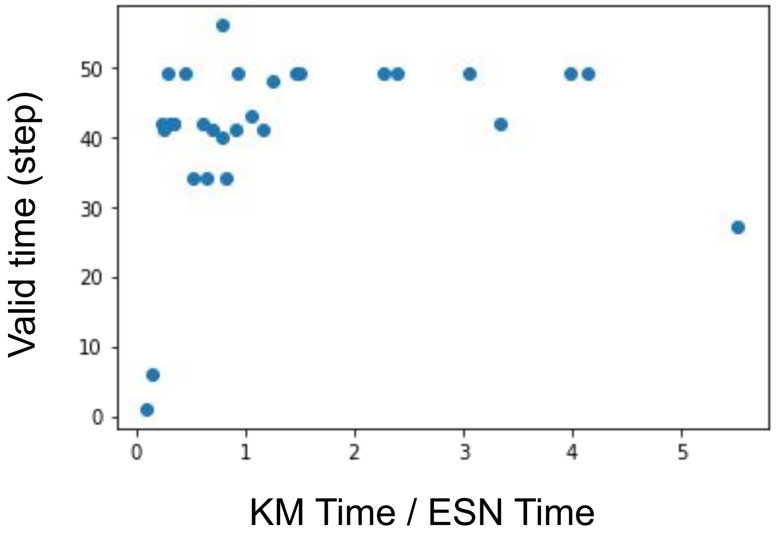


10 replicated experiments for each setting and calculate the mean and SD

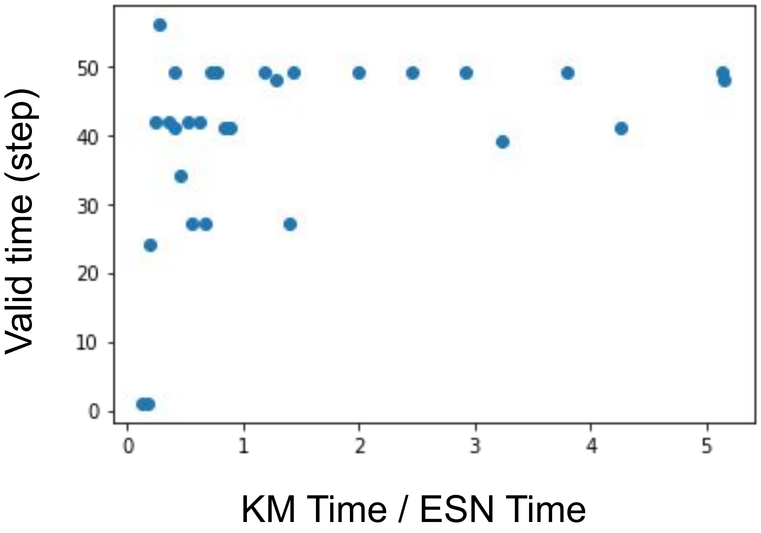
Balance computational resource for knowledge models

- Regardless of the degree of error in knowledge model, it is recommended to allocate more computational resource for ESN.

ε = 1



ε = 0



Other Attempts and future work

- Have implemented knowledge-based model and reservoir model in parallel
  - Communication times outweigh benefits within python
- Attempted to improve computational speed of knowledge based model through parallel algorithms such as **Parareal** and **MGRIT**:
  - Limitations of python and google colab (only 2 processors available)

- **Future works**
  - Could use a sparsely connected reservoir to improve stability to provide more stability.
  - Experiment in a more parallel-friendly language (such as C or MATLAB), and with more processors
  - Investigate on more experiments such as more trials, how to improve accuracy, and identify the influence of random seeds

What we learned

- Reservoir Computing (RC) and its benefits over regular RNN
- Hand-on experience on Echo State Network (ESN), a model in RC
- Basics of parallel computing and a few parallel numerical algorithms e.g. Parareal, MGRIT
- How to approach a scientific paper and collaborate on improving/working on something similar