Reservoir Computing with Noise and an Overview of Current Efforts

Francesco Sorrentino
University of New Mexico

Simon Fraser University, 11/25/2022

Current Work on Reservoir Computing

- Performance of Reservoir Computing in the presence of noise
- Decomposing Reservoir Computers into minimal units
- Time-Varying Reservoir Computing
- Limitations and alternative approaches to Reservoir Computing

Collaborators:

Chad Nathe (UNM, ME)
Raju Nayak (UNM, ME)
Pappu Chandra (Union College, NY)
Nicholas Mecholsky (Catholic University of America, DC)
Joe Hart (Naval Research Laboratory, DC)
Thomas Carroll (Naval Research Laboratory, DC)
Lou Pecora (Naval Research Laboratory, DC)
Francesco Caravelli (Los Alamos National Laboratory, NM)
Wai Lim Ku (US National Institute of Health)

Why Reservoir Computing?

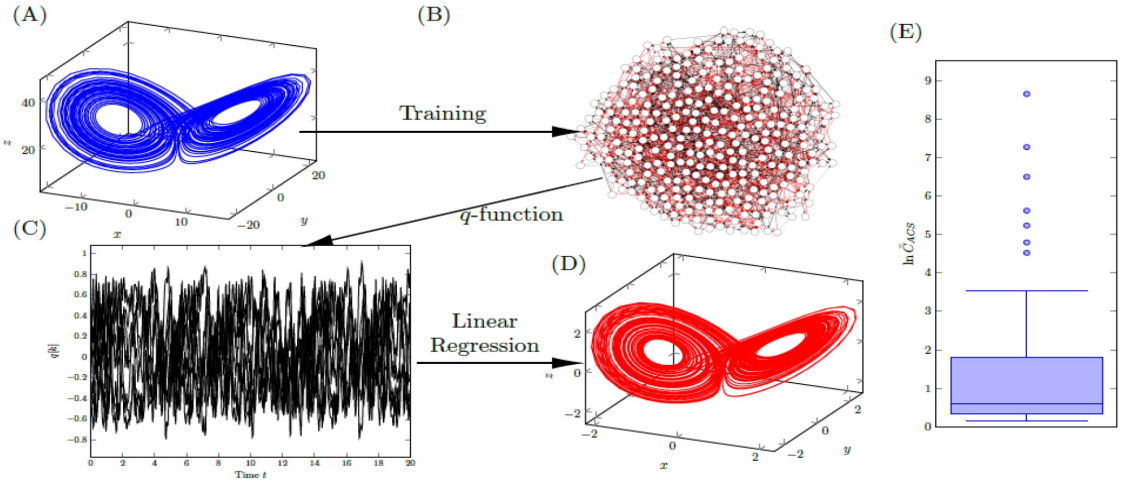
A reservoir computer (RC) is a complex nonlinear dynamical system that is used for processing and analyzing temporal data, modeling of complex dynamical systems, speech recognition, learning of context free and context sensitive languages, the reconstruction and prediction of chaotic attractors, image recognition, control of robotic systems, predicting catastrophic critical transitions and amplitude death in oscillating systems.

What are reservoir computers good at??

They are good at dealing at processing signals, of many different types eg temporal signal but also spatial signals…

Why Chaos?

Chaotic signals are sensitively dependent on the initial conditions, therefore they are are to predict.
If we can predict chaotic systems we will likely be able to predict non-chaotic systems.
An example of a physical chaotic system? The weather.
Low dimensional chaos: Lorenz, Roessler, Hindmarsh-Rose, Chen, Chua, etc…



Why Noise?

Chaotic systems/signals are deterministic

Noise is a non-deterministic “random” process

Noise affects signals in all engineering applications

Biological systems are typically affected by very large amounts of noise

Our ability to predict/observe systems is affected by the presence of noise

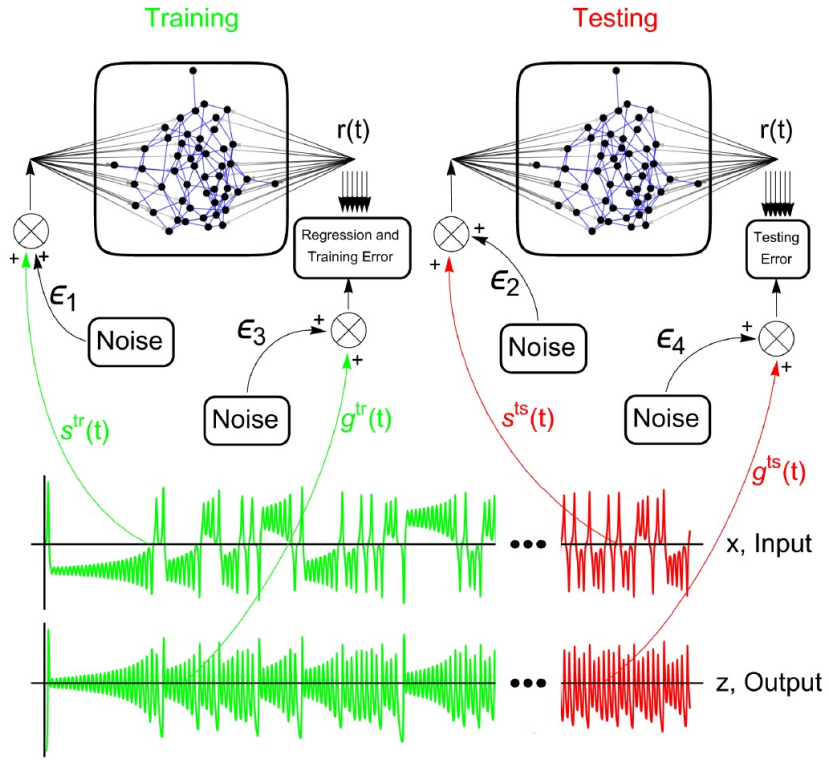
General equations of a RC

$$\dot{\mathbf{r}}(t) = \gamma[-\mathbf{r}(t) + \tanh(A\mathbf{r}(t) + s(t)\mathbf{w})],$$

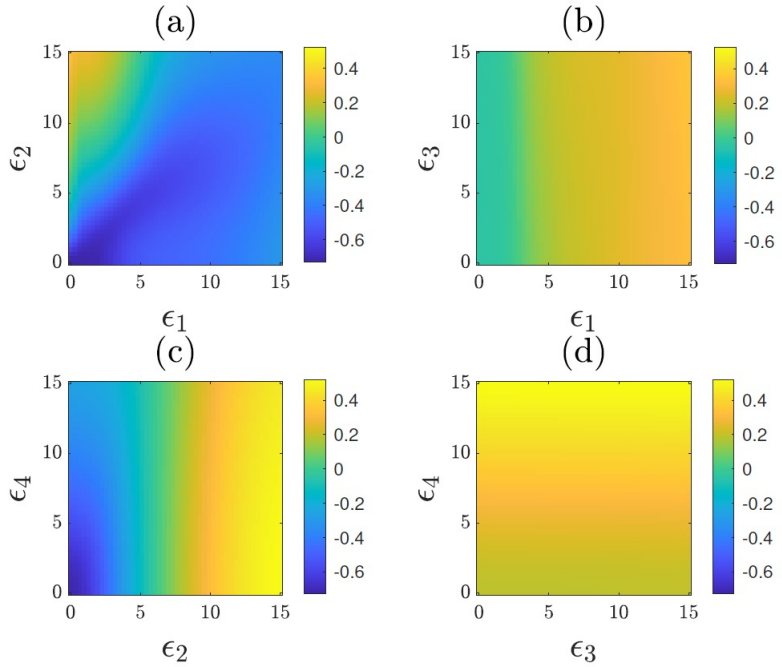
$$\dot{\mathbf{r}}(t) = \gamma[-\mathbf{r}(t) + A\mathbf{r}(t) + s(t)\mathbf{w}],$$

A compromise: Linear time varying equations

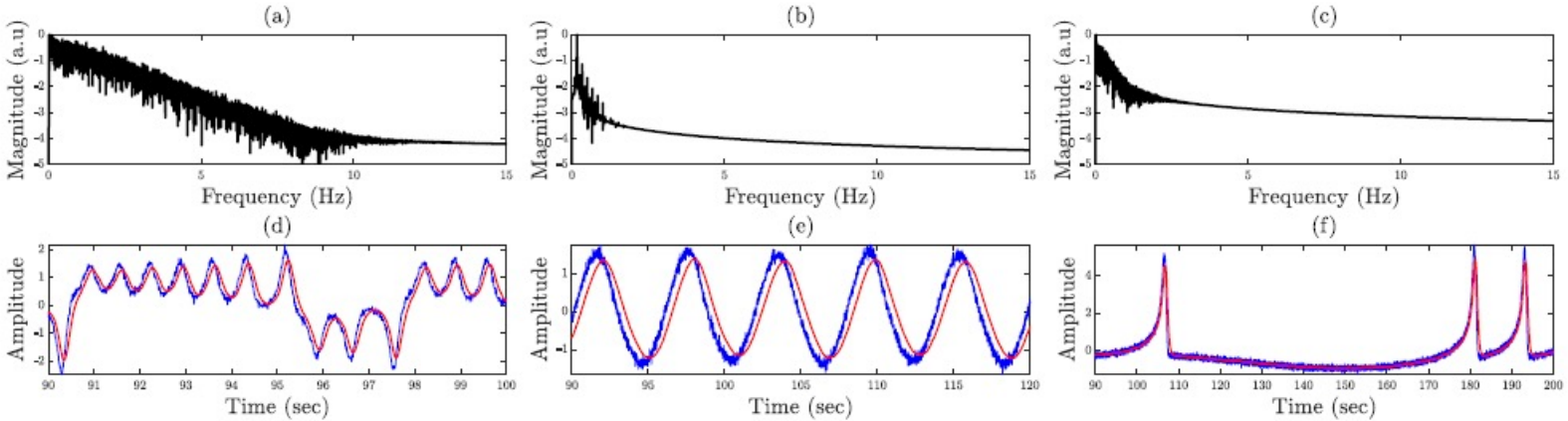
How does noise affect RC?



Performance of RCs in the presence of noise

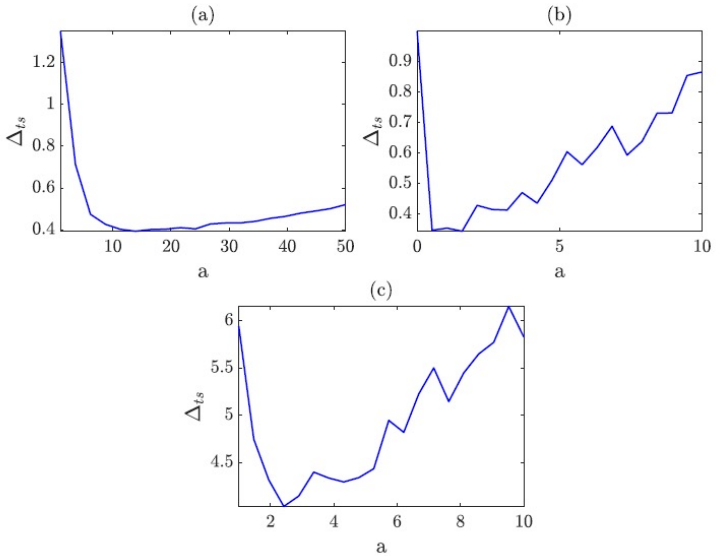


Spectrum of Chaotic Signals And Filtered Chaotic Signals



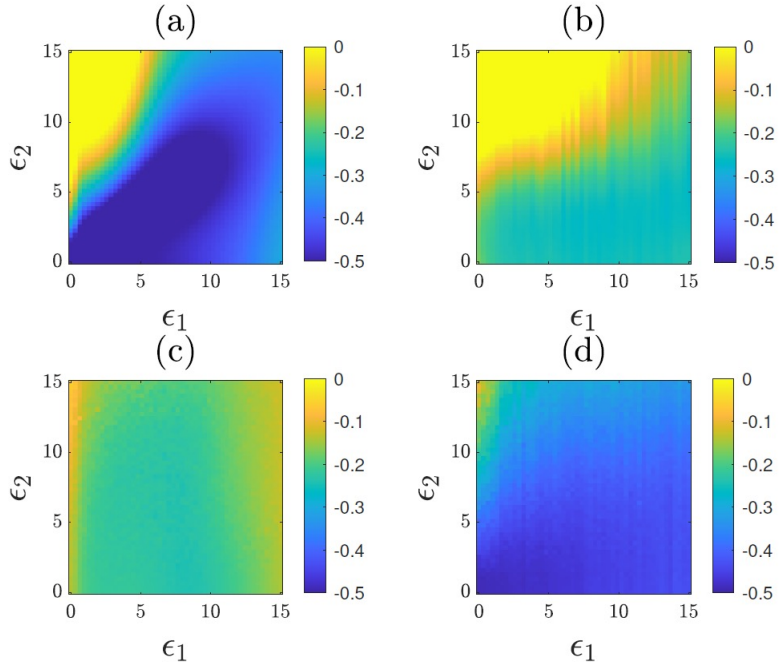
LORENZ ROESSLER HINDMARSH ROSE

Remedy to Noise: LPF

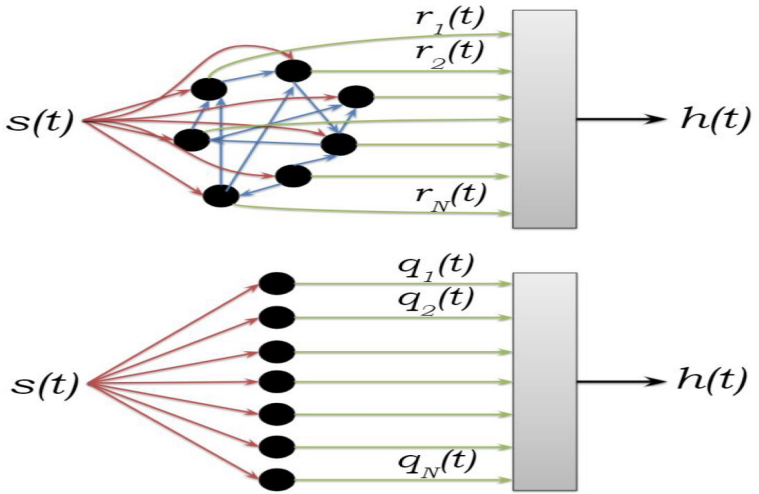


Testing errors for the (a) Lorenz, (b) Roessler and (c) HR systems, plotted as a function of the LPF cutoff frequency a.

Performance of RCs with LPF

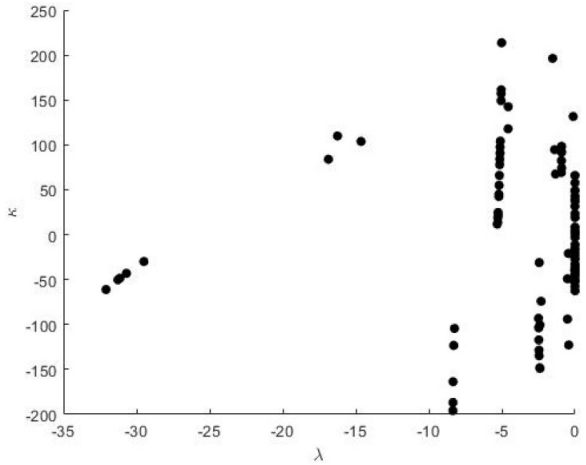


Decomposing Reservoir Computers into minimal units (part II)



This decomposition will not be generically be possible if the Reservoir is nonlinear

Advantages: optimizing the network topology in terms of eigenvalues



Results obtained via Particle Swarm Optimization

Conclusions

We first consider the case that filtering is not applied and find that the best performance is achieved when the noise strength affecting the input signal is about the same in the training and in the testing phases.

We introduce low-pass-filtering applied to the input signals both in the training phase and in the testing phase.

We investigate the performance of the RC as the cutoff frequency of the low pass filter is varied and find the optimal value of the cutoff frequency.

We see a substantial improvement in the testing error, provided that the same type of filtering is applied in the training phase and in the testing phase.

One conclusion that we obtain is that it may be good to filter input and output signals, even when an estimate on the amount of noise that affects these signals is not available.

RCs can be decomposed in independent modes and optimized

Sources

- MATLAB code used throughout derived from code provided by Tom Carroll