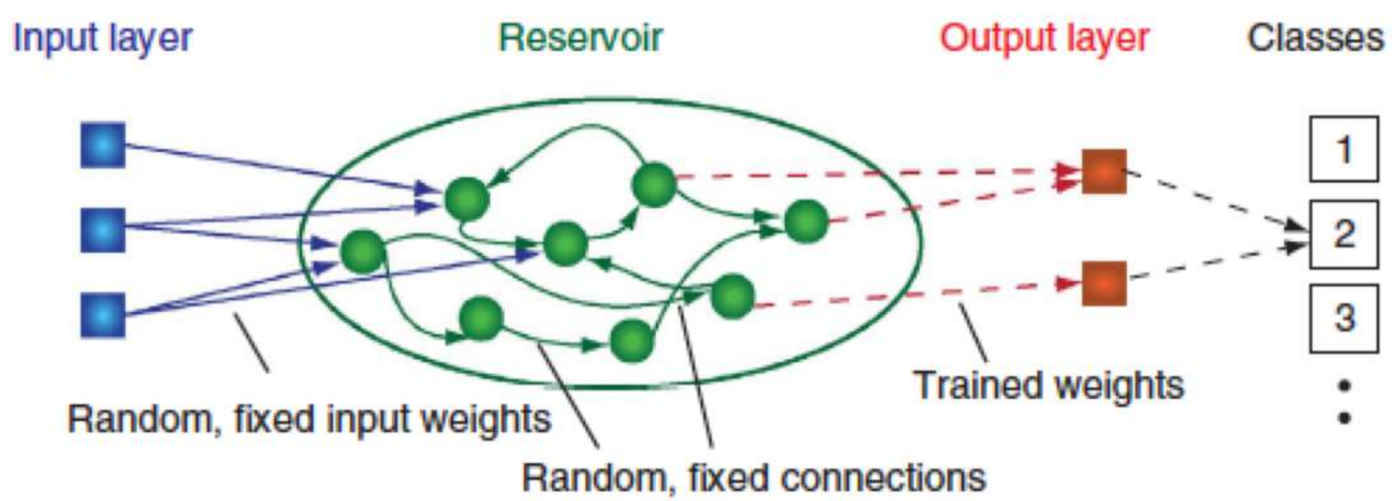
An Introduction to Reservoir Computing – the new frontier in AI and Neural Networks

Lou Pecora and Tom Carroll
6392 Magnetic Materials and Nonlinear Dynamics
6390 Center for Materials Physics and Technology
U.S. Naval Research Laboratory

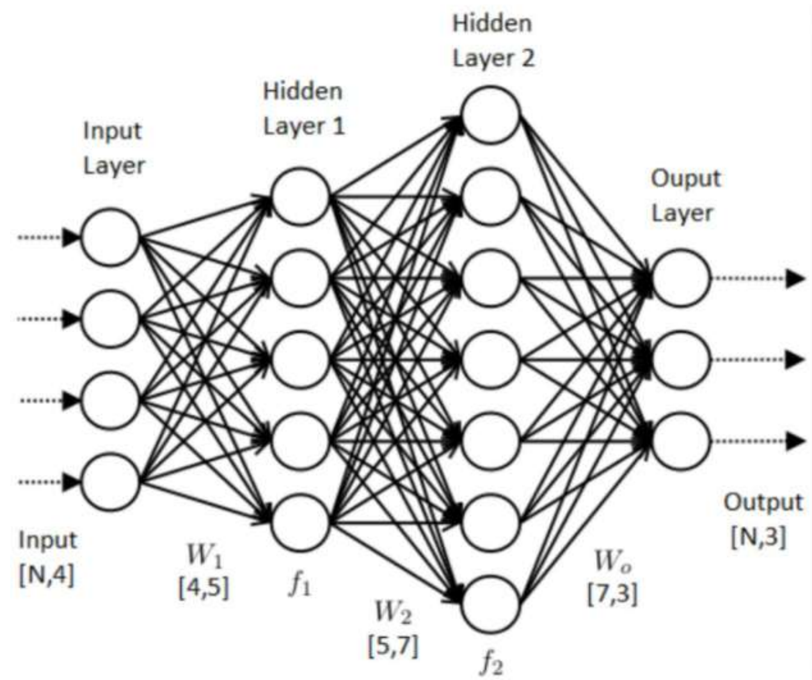


ISINP-2 Como, Italy, Jul 2019

- Examples of Reservoir Computing (RC)
- Some nonlinear dynamics to model RC
- Applications of RC to dynamical system
- Relation to physiological systems?

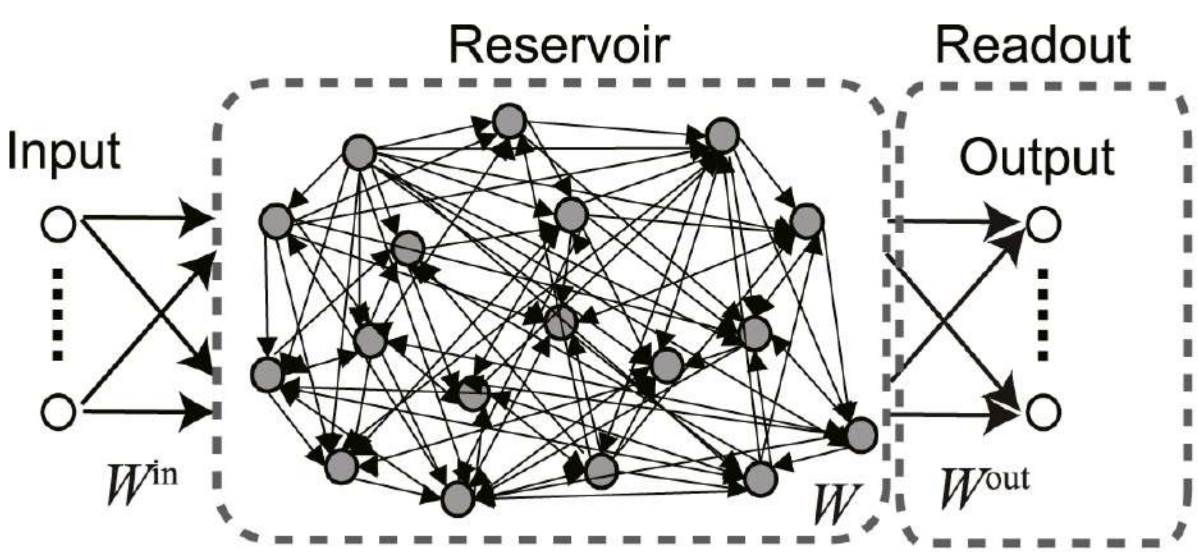
Neural Networks vs. Reservoir Computer

Neural Networks

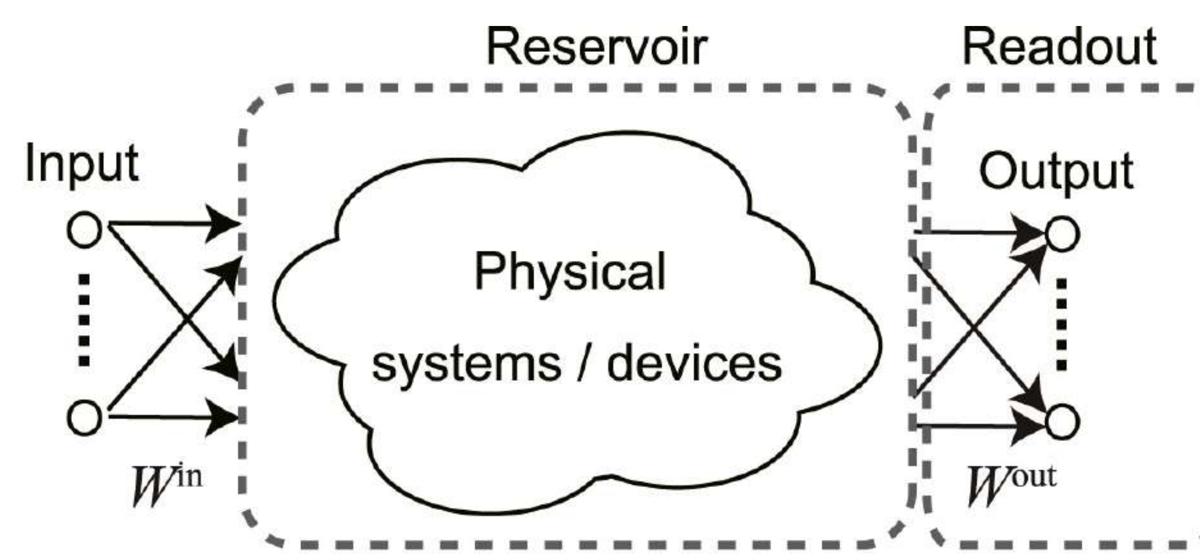


output generates information signals that classify, analyze, make decisions, generate other useful signals, etc.

Reservoir Computer



(a) Conventional RC



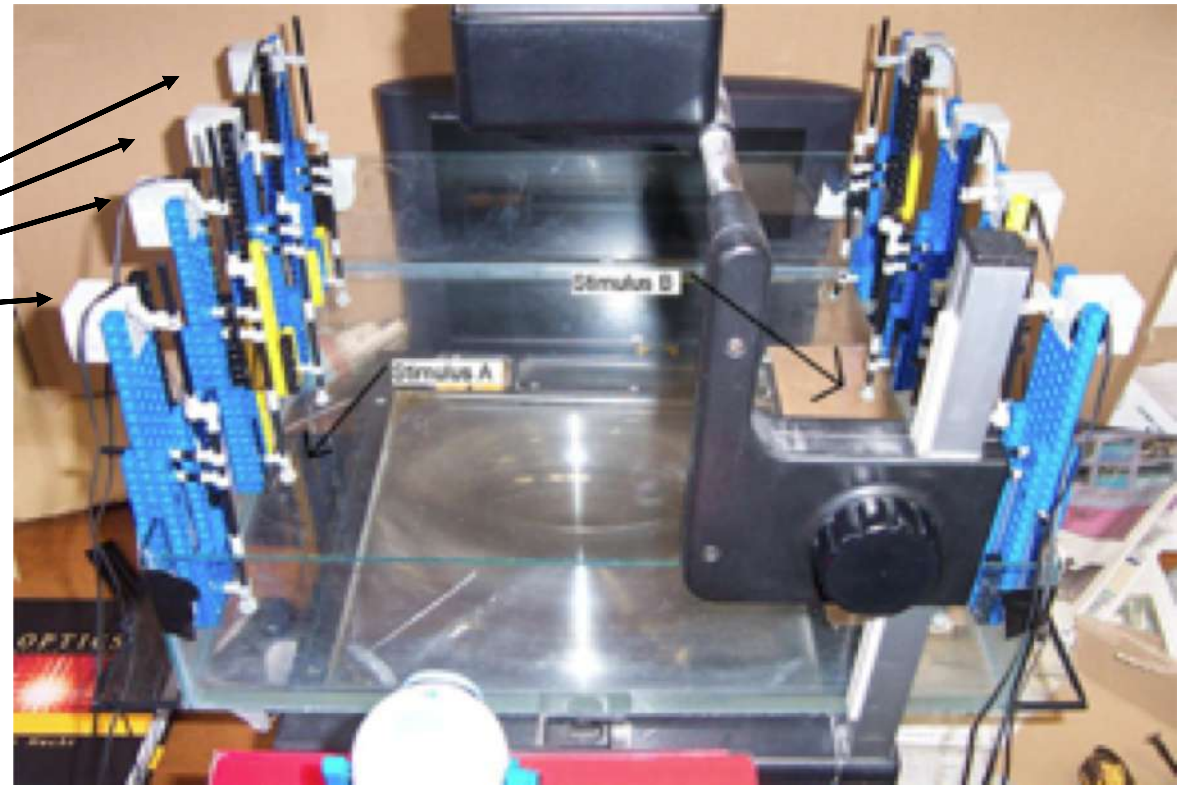
(b) Physical RC

(Tanaka, et al., Recent Advances in Neural Networks, Neural Networks, volume. 115, pp 100-123 (2019))

- **Examples of Reservoir Computing (RC)**

Some Interesting Reservoir Computers

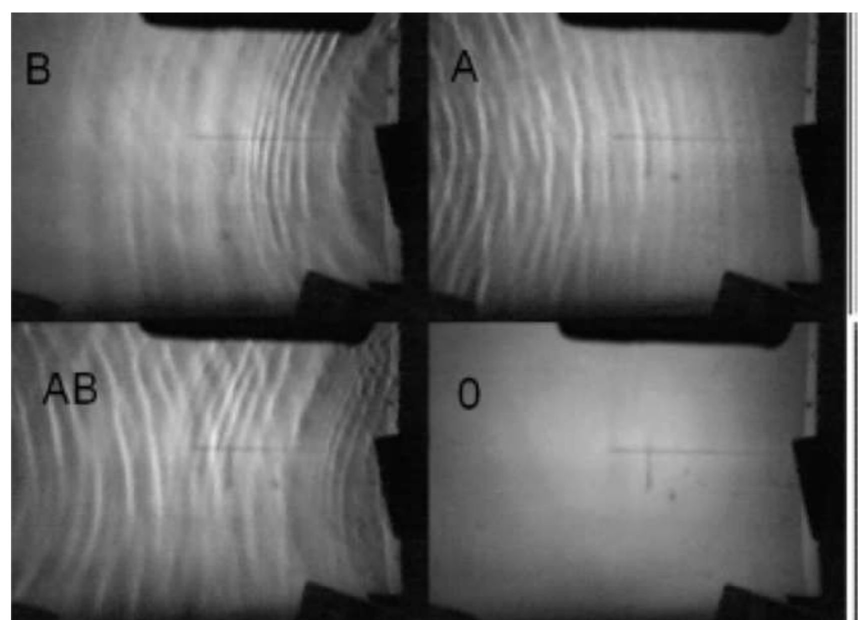
A liquid state machine.



input

Spoken
“one” or “zero”

training with 20
samples of each word

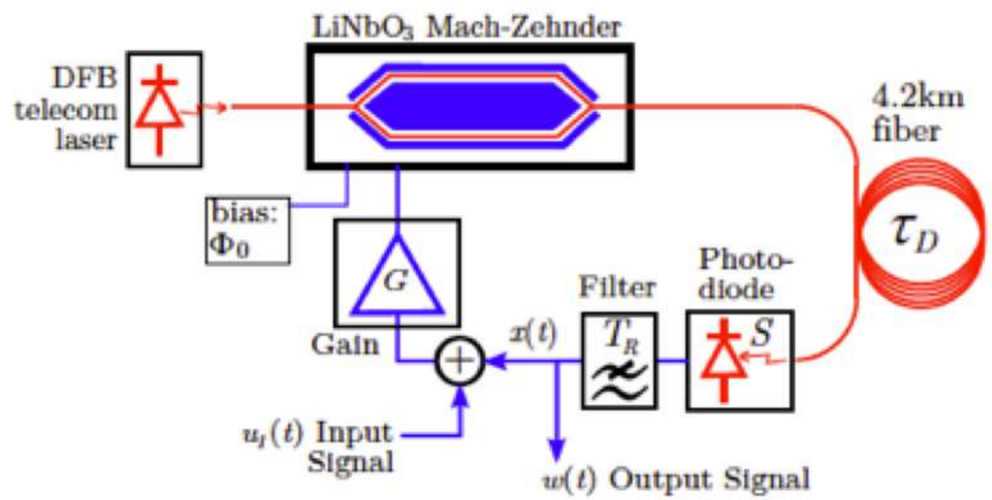


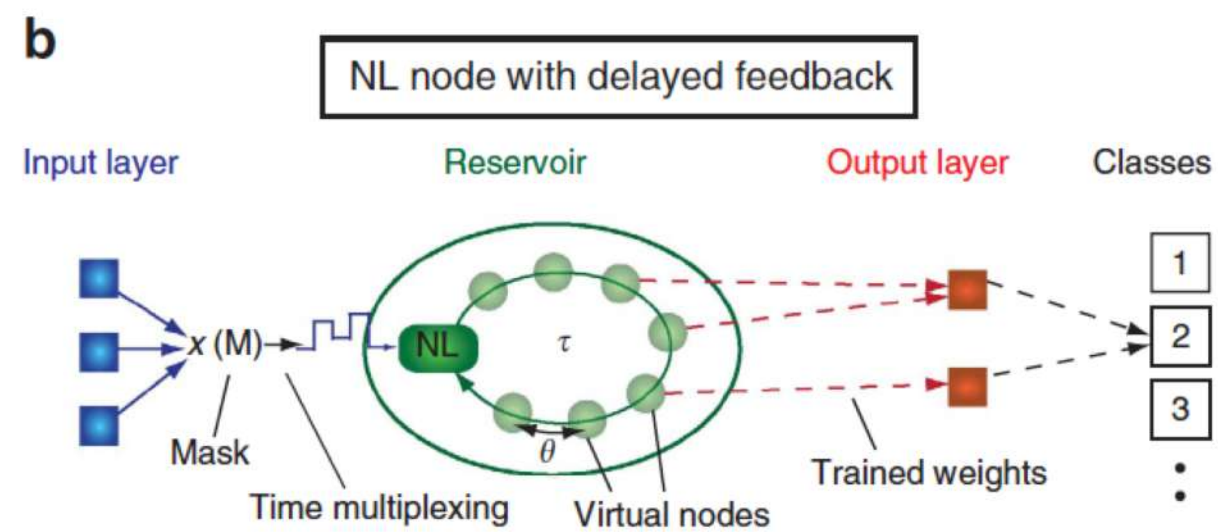
output
processed using
linear methods

“one” or “zero”

Fernando & Sojakka, School of Cognitive and Computer Sciences, U. Sussex, Brighton, UK

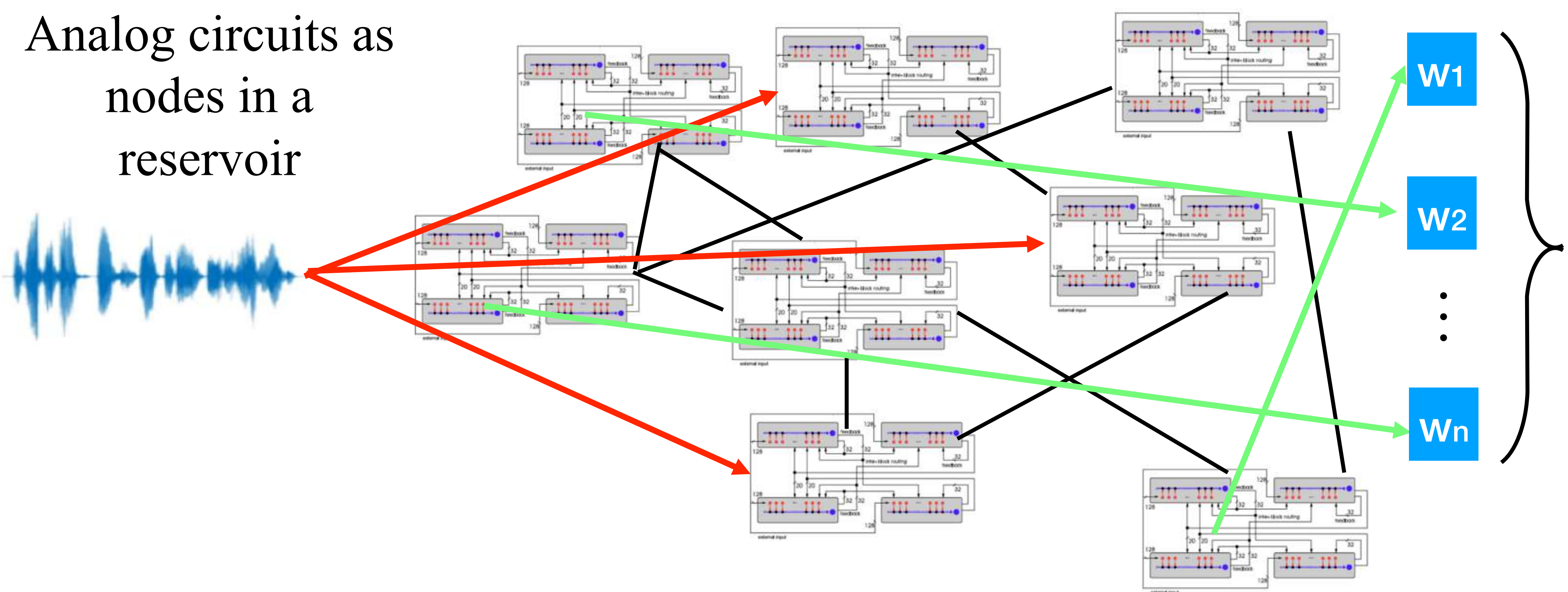
A delayed feedback laser system.





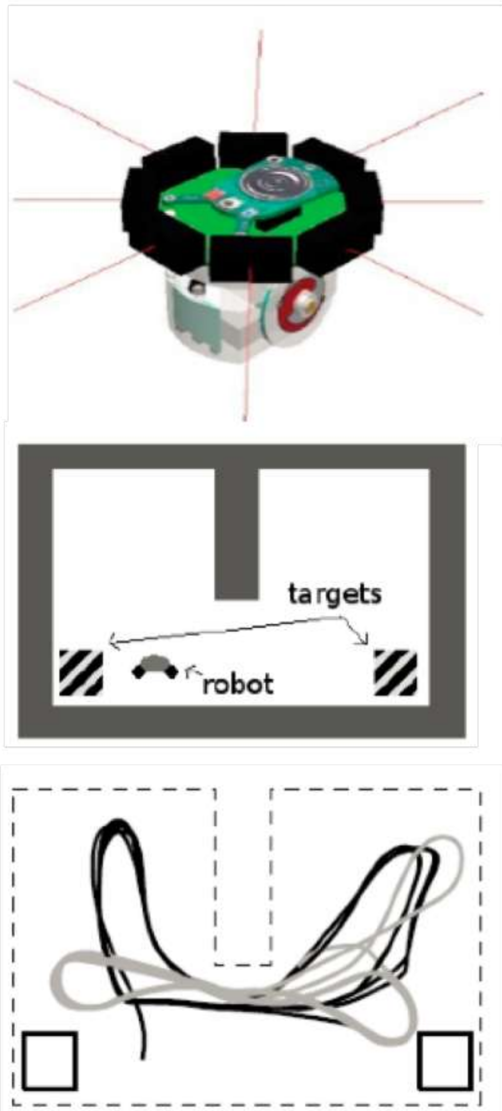
Speech recognition
Numbers: “1”, “2”, etc.

Analog circuits as nodes in a reservoir

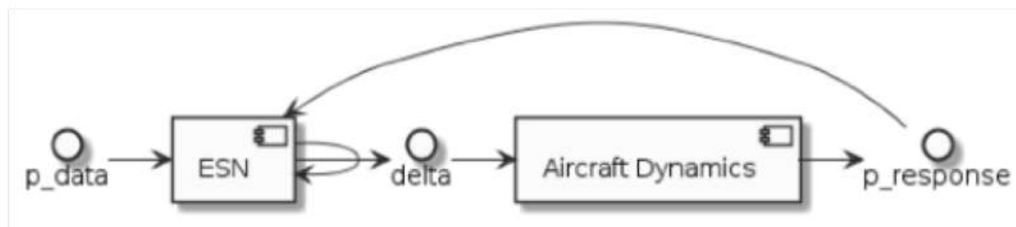


+ = Information signal

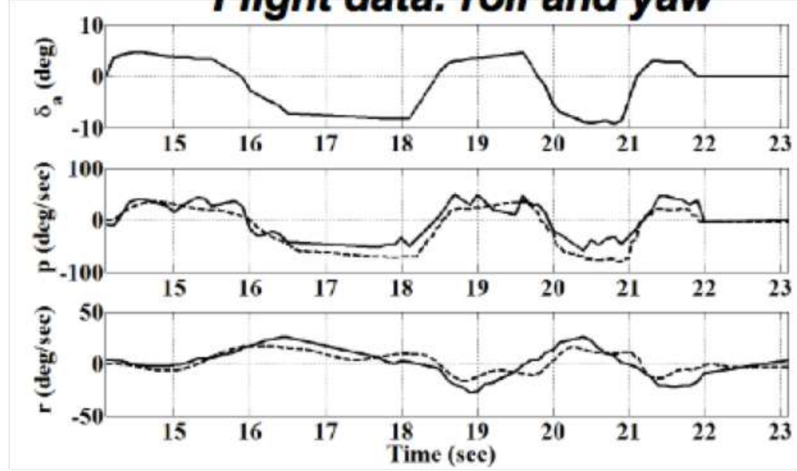
Robot Navigation
(Ghent University, Belgium)



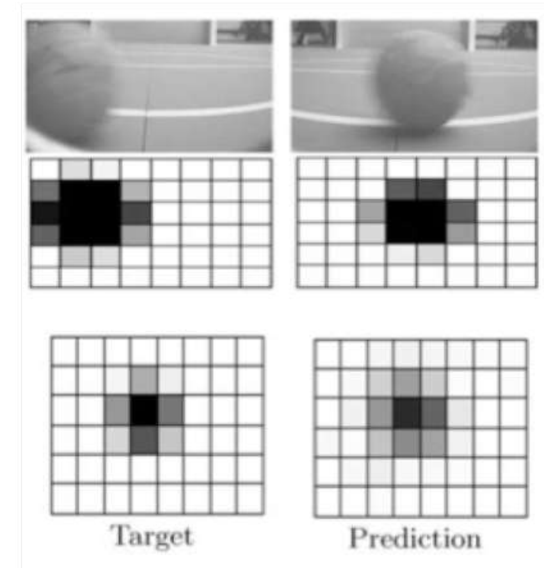
Nonlinear control of UAV
(Cal. State Univ., Pomona)



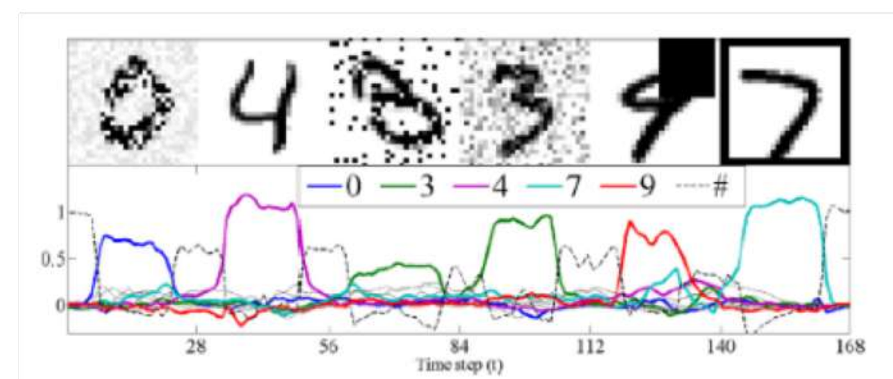
Flight data: roll and yaw



Movement prediction
(Graz University, Austria)



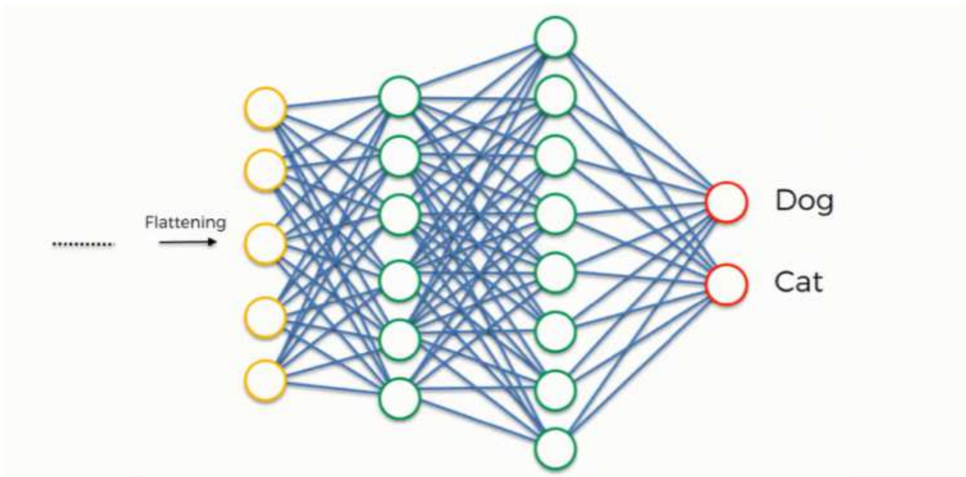
Noisy image recognition
(Univ. Ghent / Korean Inst. S&T)

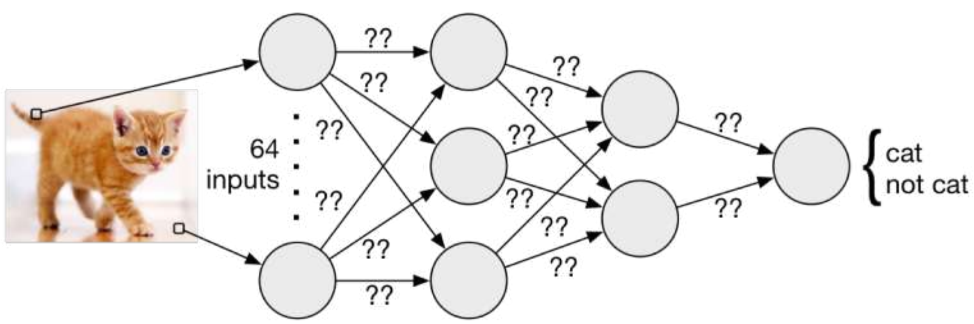


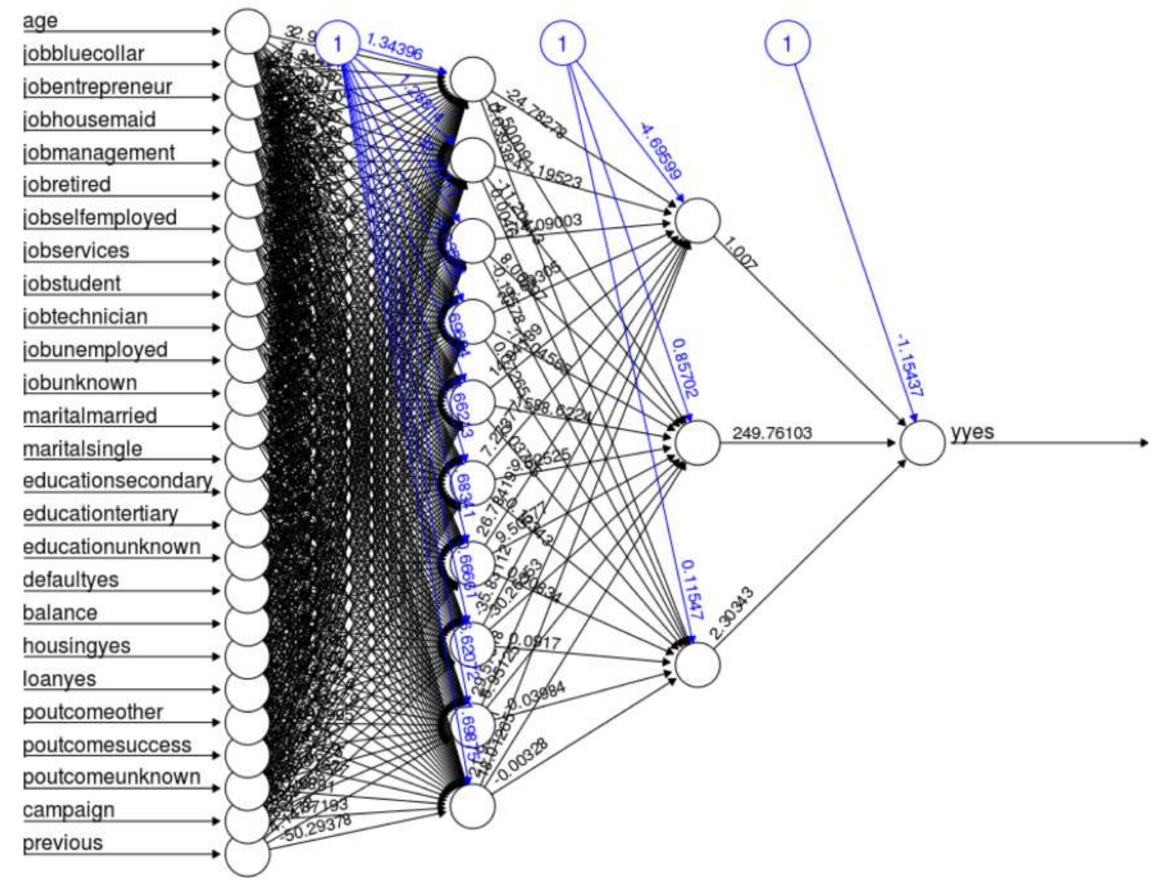
Other test applications

Applications	Benchmark tasks
Pattern classification	Spoken digit recognition (Verstraeten et al., 2005b) Waveform classification (Paquot et al., 2012) Human action recognition (Soh and Demiris, 2012) Handwritten digit image recognition (Jalalvand et al., 2015)
Time series forecasting	Chaotic time series prediction (Jaeger, 2001a) NARMA time series prediction (Jaeger, 2003)
Pattern generation	Sine-wave generation (Jaeger, 2002) Limit cycle generation (Hauser et al., 2012)
Adaptive filtering and control	Channel equalization (Jaeger and Haas, 2004)
System approximation	Temporal XOR task (Bertschinger and Natschläger, 2004) Temporal parity task (Bertschinger and Natschläger, 2004)
Short-term memory	Memory capacity (Jaeger, 2001b)

How is this different from Neural Networks, etc.?







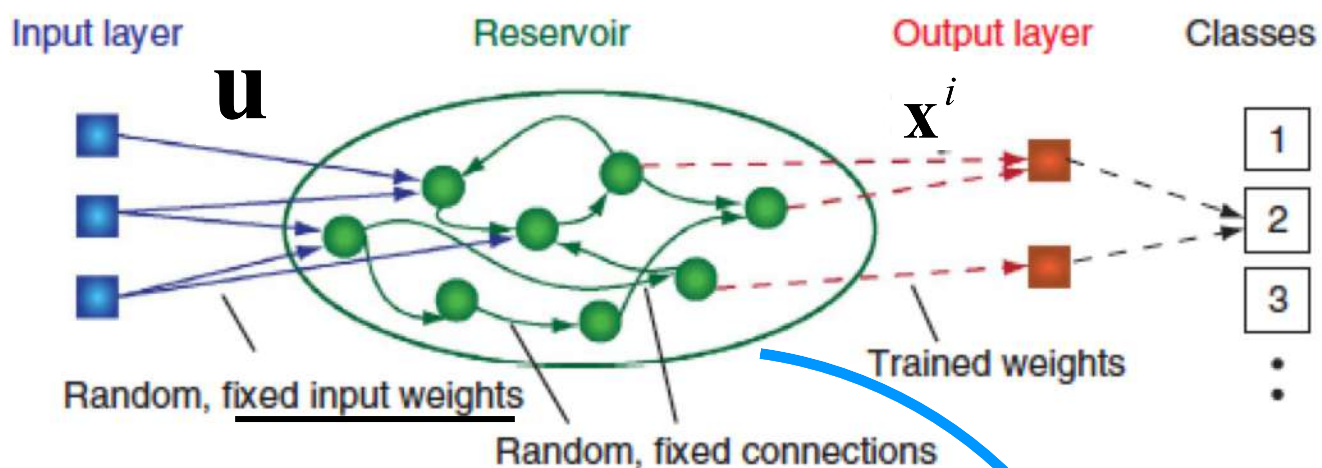
Both NN and RC involve supervised learning.

- NN train the whole network.
- RC only train the output weights. Network stays fixed.

FAST training

- **RC can be physical systems.** ***FAST operation***

What is happening here?



$\sum_i W_i \mathbf{x}_i$ = information out or choice/decision

W_i obtained from training

Each node is a dynamical object

i^{th} node

$$\frac{d\mathbf{x}^i}{dt} = \mathbf{F}\left(\mathbf{x}^i\right)$$

$$\frac{d\mathbf{x}^i}{dt} = \mathbf{F}\left(\mathbf{x}^i\right) + \sigma \sum_{j=1}^{N} C_{ij} \mathbf{H}\left(\mathbf{x}^j\right) + \mathbf{J}\left(\mathbf{u}\right)$$

network

input signals

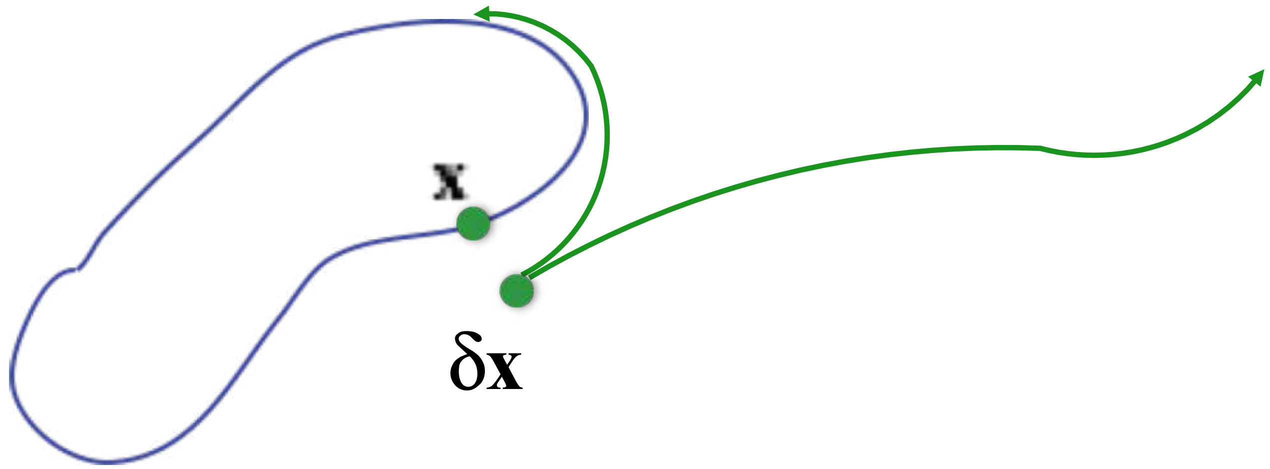
Reservoir = Driven Dynamical System

Some Dynamical properties of a Reservoir.

Some Nonlinear Dynamics Concepts

Determing the stability of the dynamics

$$\frac{d\mathbf{x}^i}{dt} = \mathbf{F}\left(\mathbf{x}^i\right)$$



$$\frac{d(\mathbf{x}+\delta\mathbf{x})}{dt} = \mathbf{F}\left(\mathbf{x}+\delta\mathbf{x}\right) = \mathbf{F}\left(\mathbf{x}\right) + \frac{\partial\mathbf{F}}{\partial\mathbf{x}}\delta\mathbf{x} + \ldots = \mathbf{F}\left(\mathbf{x}\right) + D\mathbf{F}\ \delta\mathbf{x}$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{F}\left(\mathbf{x}\right)$$

$$\frac{d\delta\mathbf{x}}{dt} = D\mathbf{F}\left(\mathbf{x}\right)\delta\mathbf{x}$$ variational equation

$$\delta\mathbf{x} \sim e^{\lambda t}$$

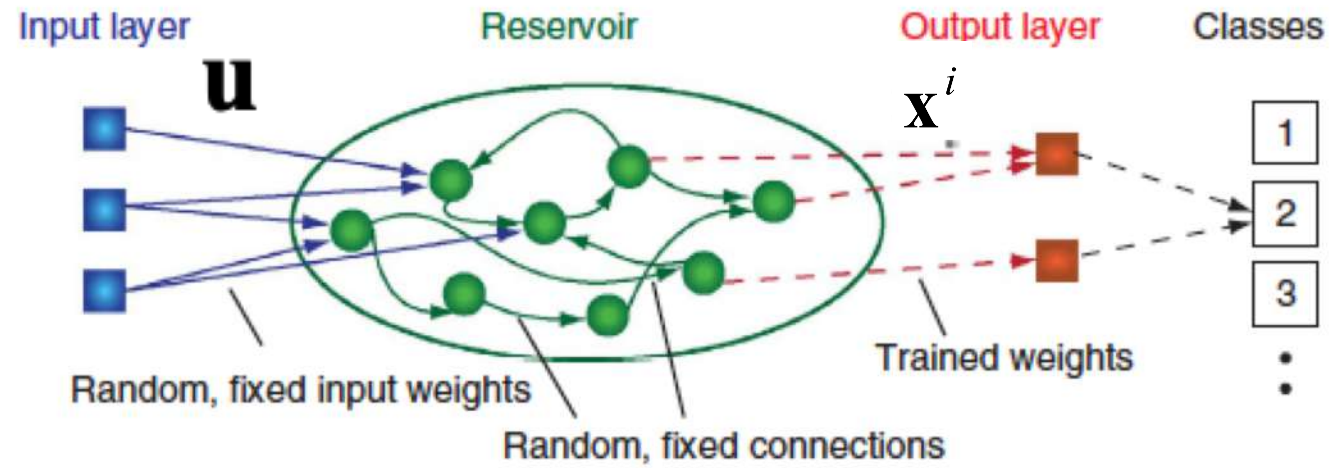
$\lambda < 0$ stable — Fixed point, Periodic

$\lambda = 0$ neutral — Periodic

$\lambda > 0$ unstable — Chaotic

Number of exponents= dimension of the system

Driven Systems



$$\frac{d\mathbf{x}^i}{dt} = \mathbf{F}\left(\mathbf{x}^i\right) + \sigma \sum_{j=1}^{N} C_{ij}\mathbf{H}\left(\mathbf{x}^j\right) + \mathbf{J}\left(\mathbf{u}\right)$$

network

input signals

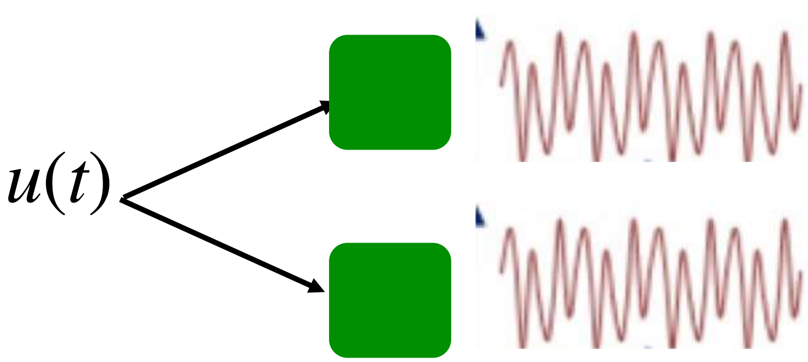
$\boldsymbol{u}(t)$

Periodic
Chaotic
Random
A structured signal (Radar)
Acoustic (speech)
….

$$|\delta\mathbf{x} \sim e^{\lambda t}|$$

$\lambda < 0$ stable
$\cancel{\lambda = 0 \text{ neutral}}$
$\lambda > 0$ unstable

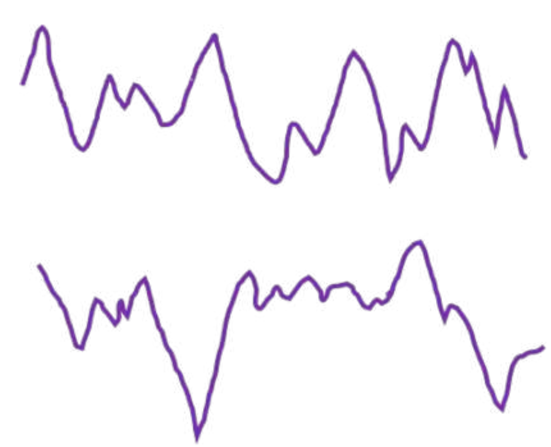
$\lambda < 0$ stable



will generate same answer

Stability or Reproducibility

$\lambda > 0$ unstable

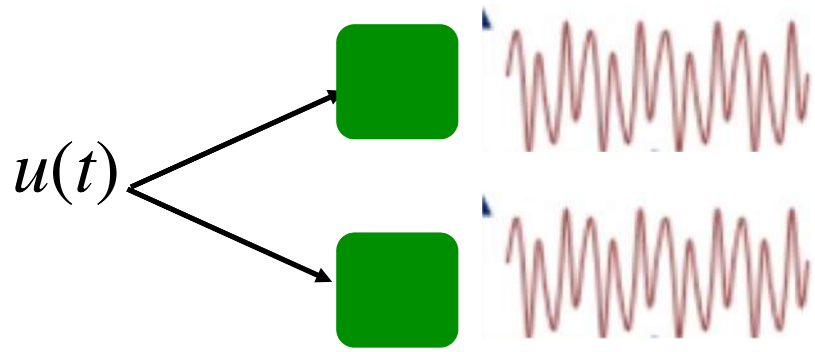


will generate different answers

generalized synchronization (Rulkov, et al.)

Generalized Synchronization

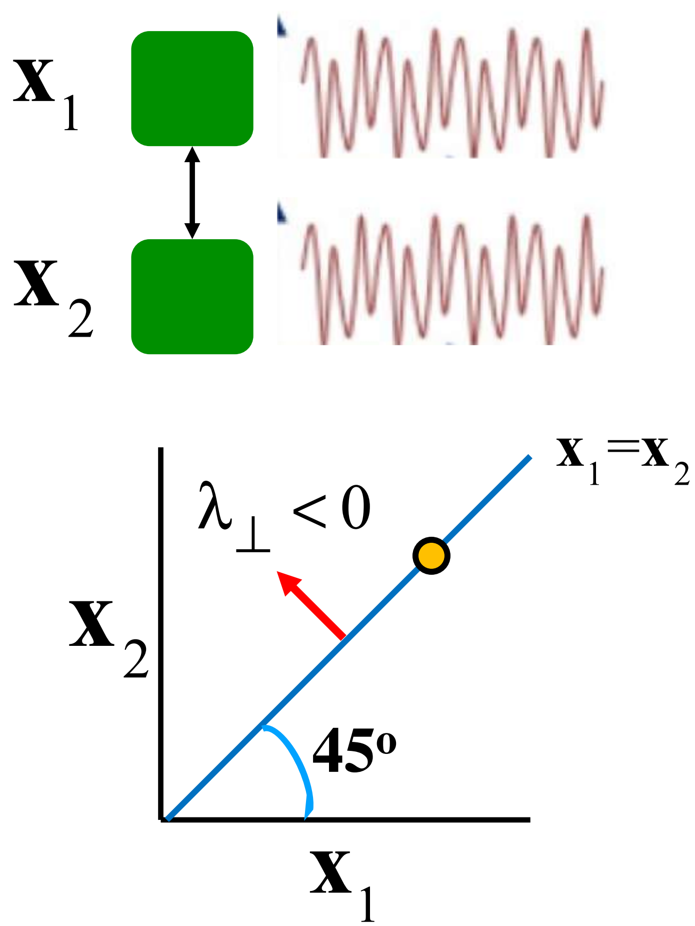
$\lambda < 0$ stable



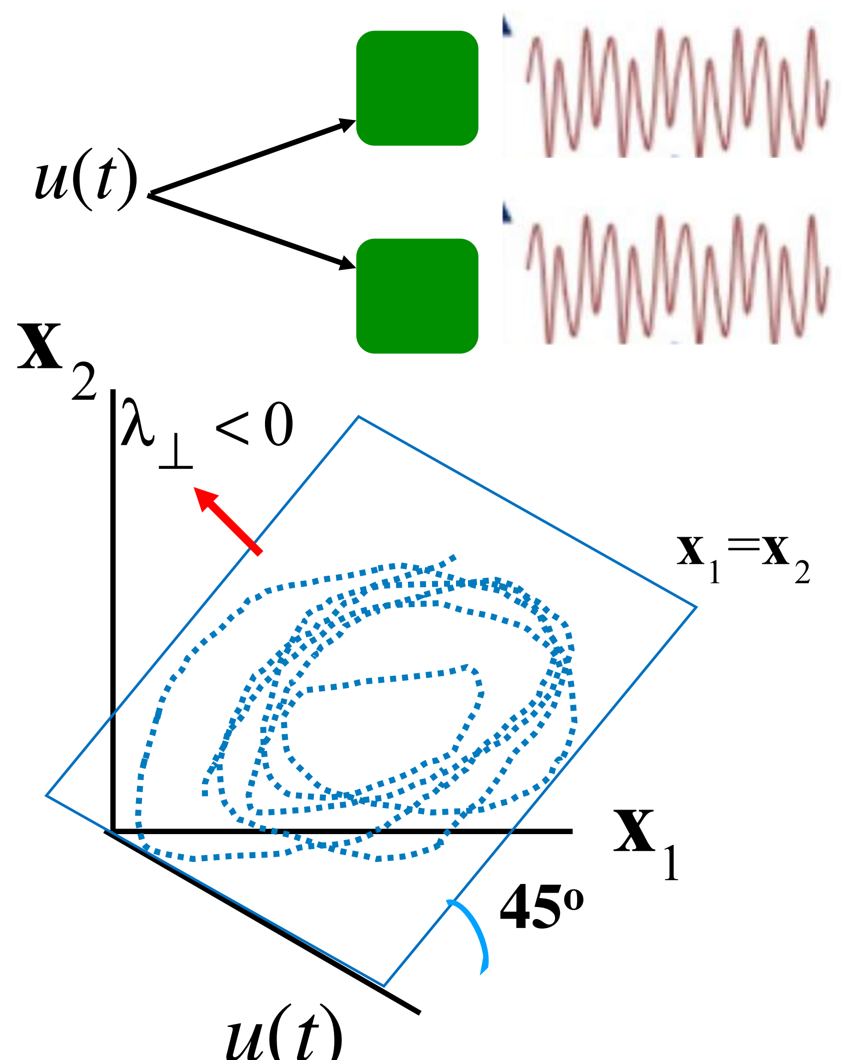
will generate same answer

Stability or Reproducibility

mutual synchronization



generalized synchronization



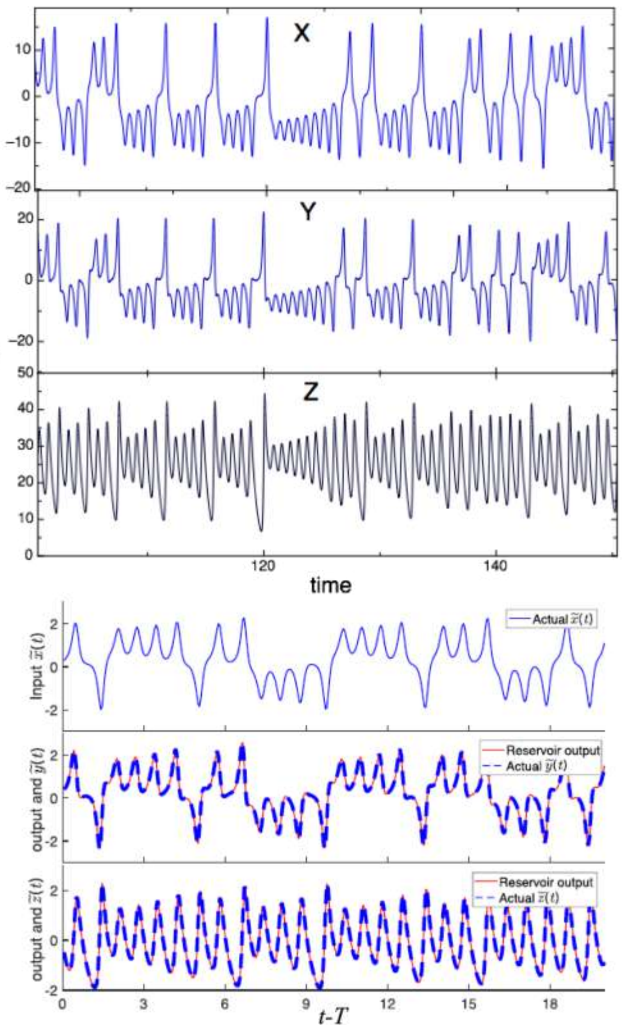
NOTE: $\lambda_\perp$ depends on $u(t)$. So we write $\lambda_\perp(u)$ **conditional** Lyapunov exponent

- **Applications of RC to dynamical system**

Using Reservoir Computing to Reconstruct Dynamical Signals

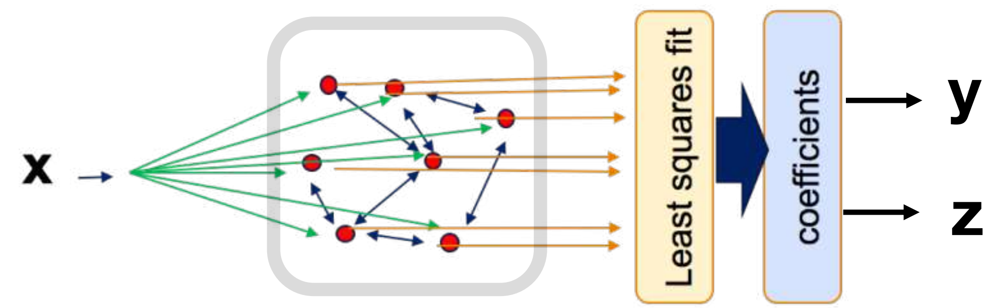
(a) Reproduce "missing signals" from input signal from a dynamical system (Tom Carroll, NRL)

$$\frac{dx}{dt} = \sigma(y - x)$$
$$\frac{dy}{dt} = x(\rho - z) - y$$
$$\frac{dz}{dt} = xy - \beta z$$

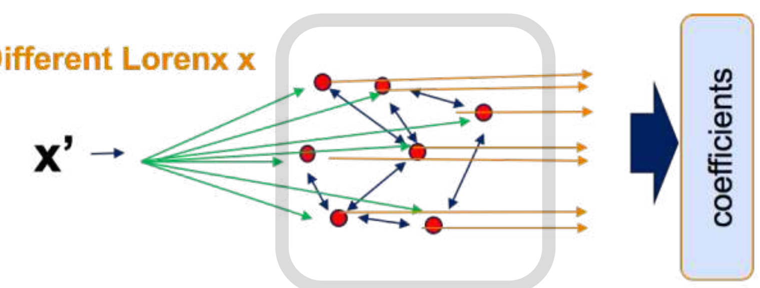


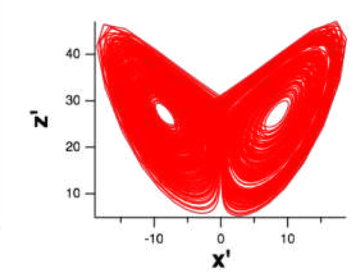
Using a polynomial vector field

$$\frac{d\mathbf{R}}{dt} = \tau\left[b_1\mathbf{R} + b_2\mathbf{R}^2 + b_3\mathbf{R}^3 + \mathbf{A}\mathbf{R} + \mathbf{W}s\right]$$



Training





Application

Calculating properties of the system during the application gives correct answers.

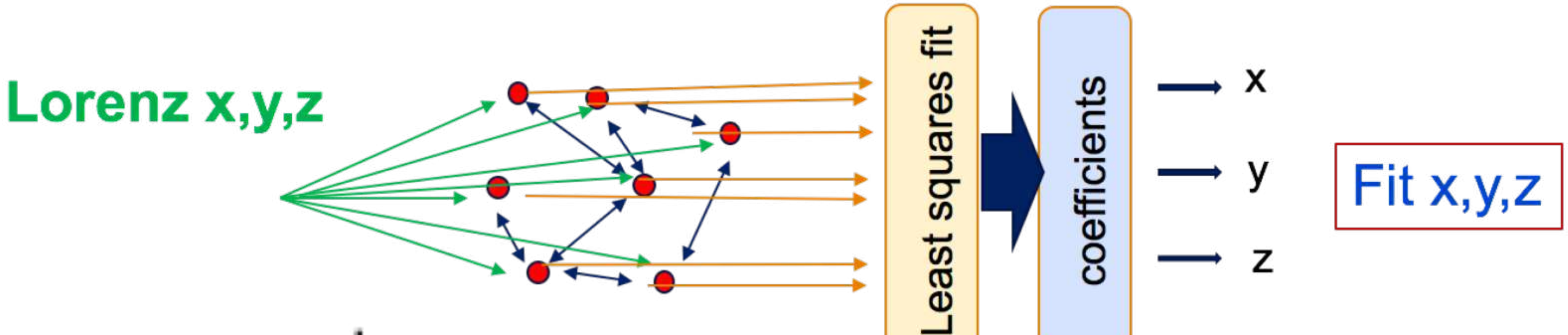
Lyapunov exponents = (-14.5723906, ~ 0.0, +1.64023001)

(b) Preliminary test of reservoir training verses Neural Network. (Tom Carroll, NRL)

- LSTM neural network with 2 layers and 50 hidden nodes in each layer
- 1000 node reservoir with polynomial vector fields and random connections

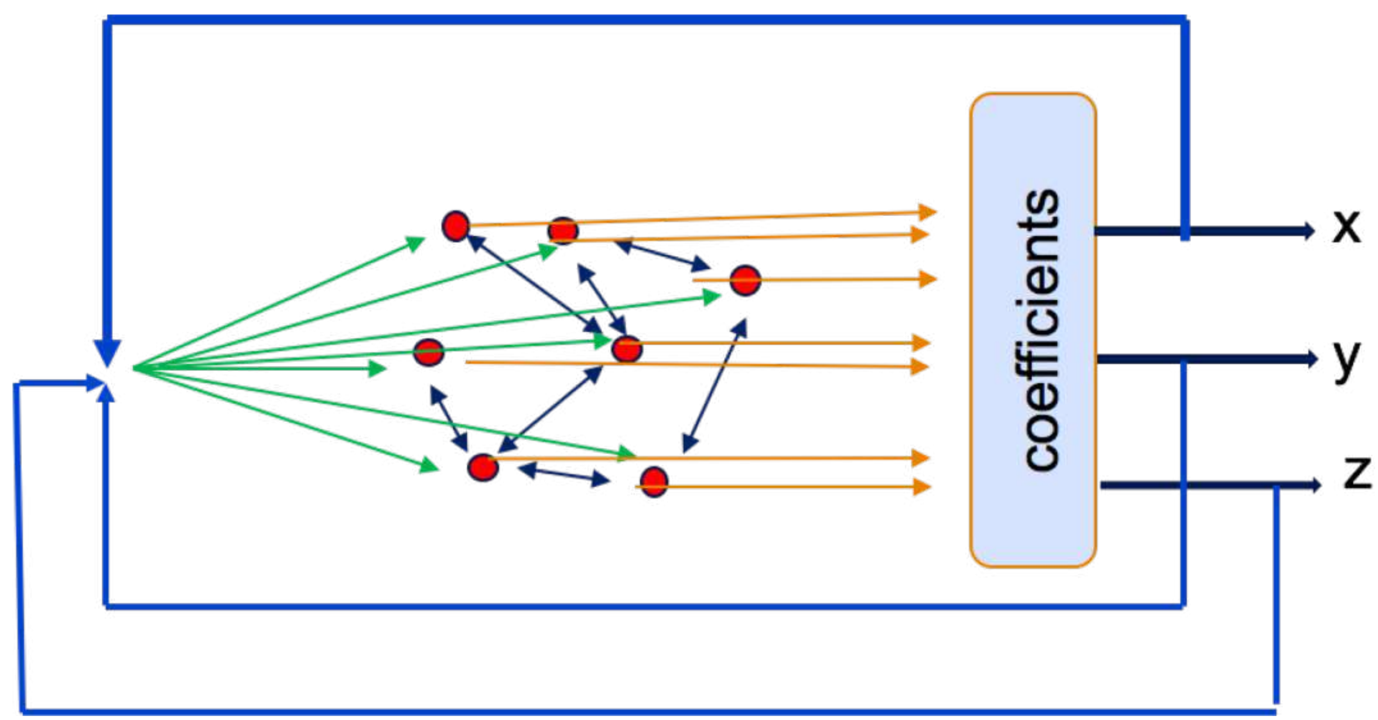
neural network took 1521 seconds to train and the error in fitting the z signal was 0.12. The 1000 reservoirs took a total of 180 seconds and gave an error of 0.0012.

“Learning” the dynamics of a dynamical system

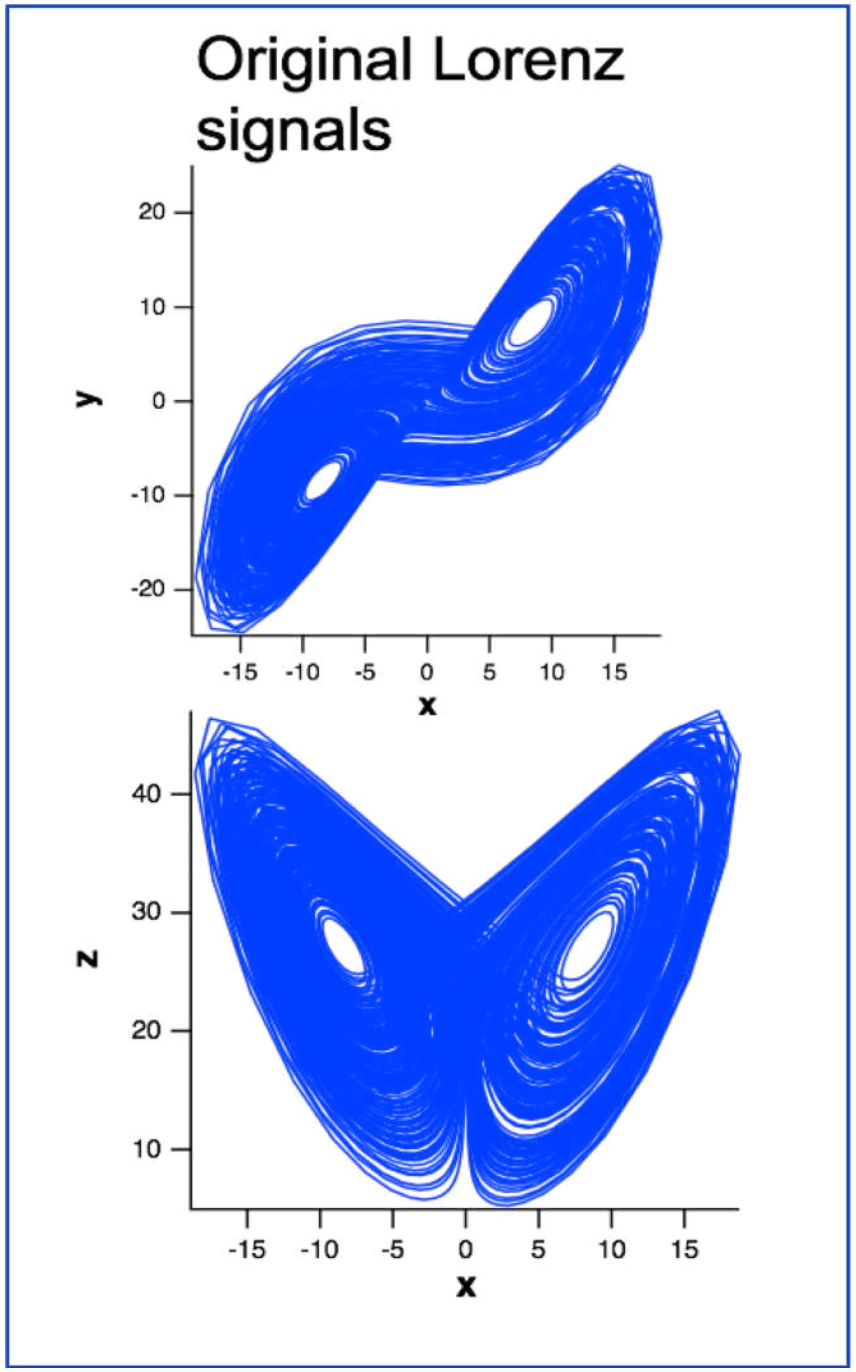


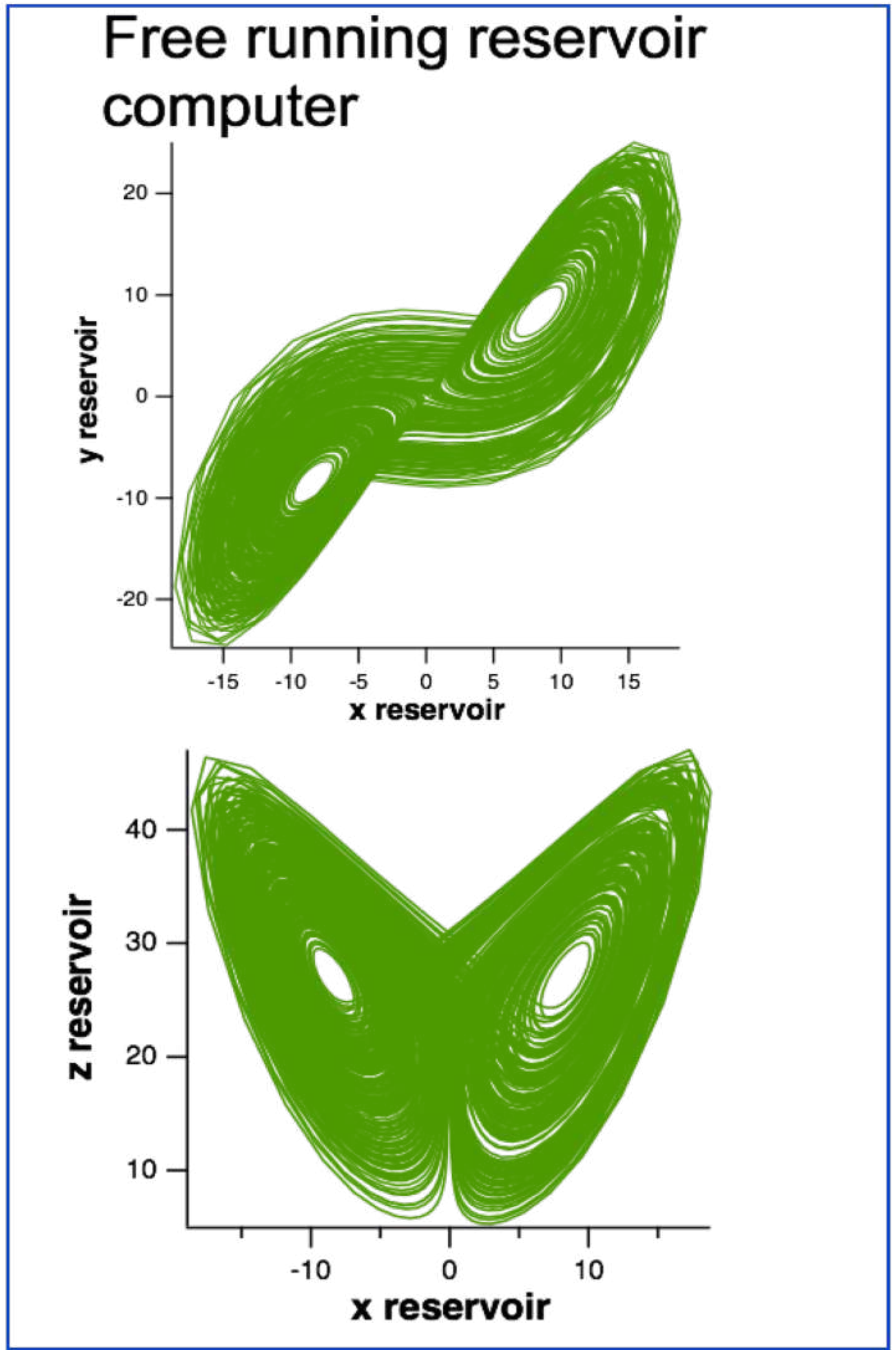
$$\frac{dx}{dt} \sim \tanh(x + Wu)$$

Simulations from University of Maryland



Free running reservoir computer reproduces the Lorenz attractor





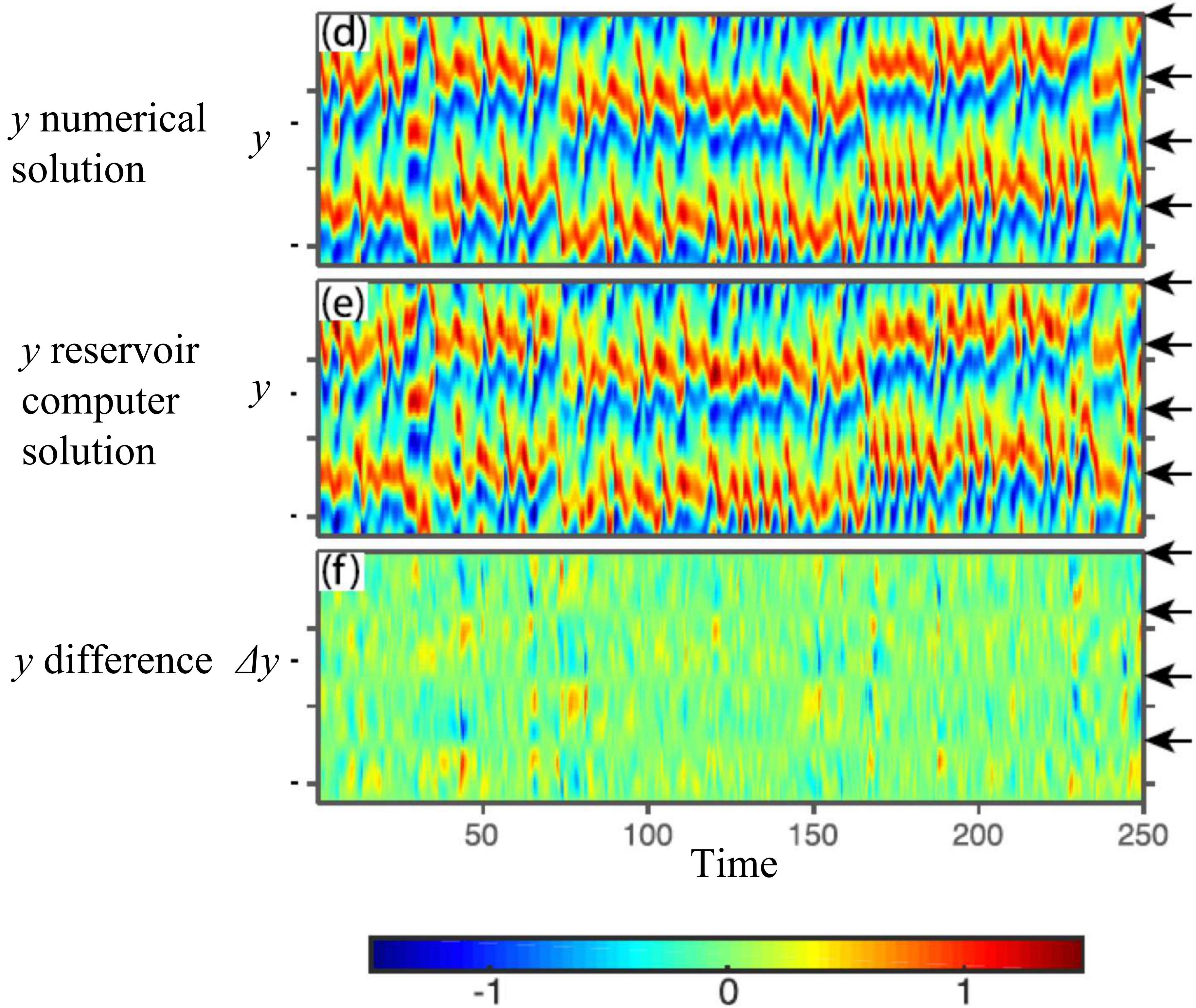
J. Pathak,Z. Lu,1, B.R. Hunt, M. Girvan, E. Ott, Using Machine Learning to Replicate Chaotic Attractors and Calculate Lyapunov Exponents from Data, CHAOS

Attractor reconstruction by machine learning, Z. Lu, B. R. Hunt, and E. Ott

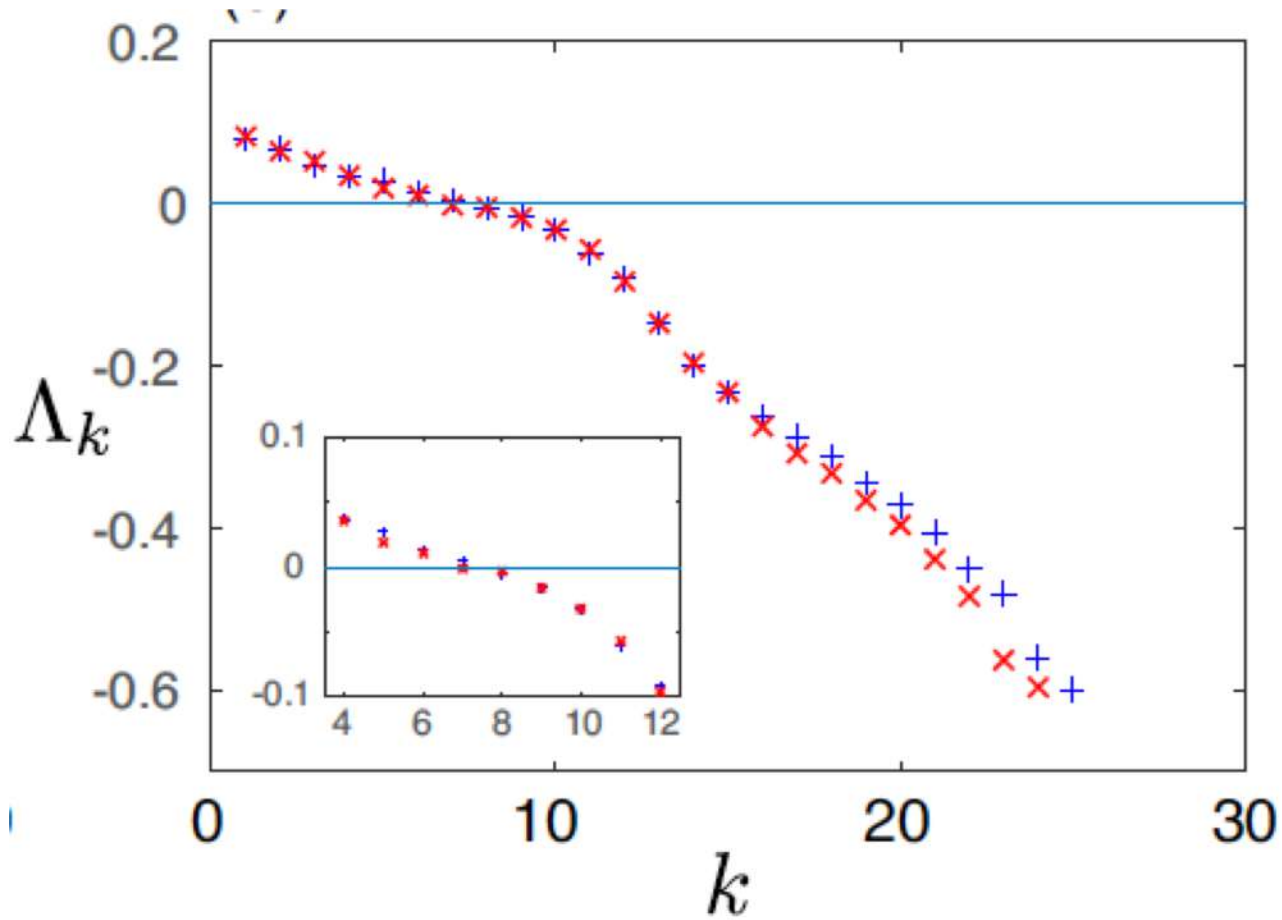
Kuramoto–Sivashinsky equations

$$\frac{\partial y}{\partial t} = -y\frac{\partial y}{\partial x} - \frac{\partial^2 y}{\partial x^2} - \frac{\partial^4 y}{\partial x^4}$$

Circular boundary conditions



Kuramoto–Sivashinsky equations, Lyapunov exponents



Calculating the first 26 Lyapunov exponents using RC

So far: a little bit of the dynamics of a RC (stability)

How does a RC work?

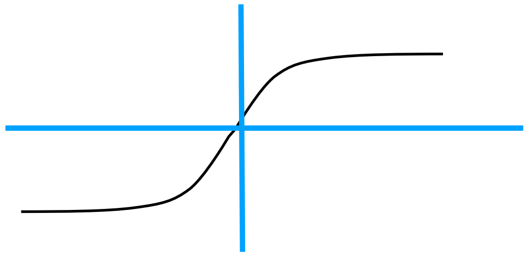
How can a RC emulate a full dynamical system just from time series?

How can a RC identify audio or visual signals ?

Nobody knows.

RC “Folklore”

- **Operate at the edge of chaos**
- **Need to use sigmoid nodes**



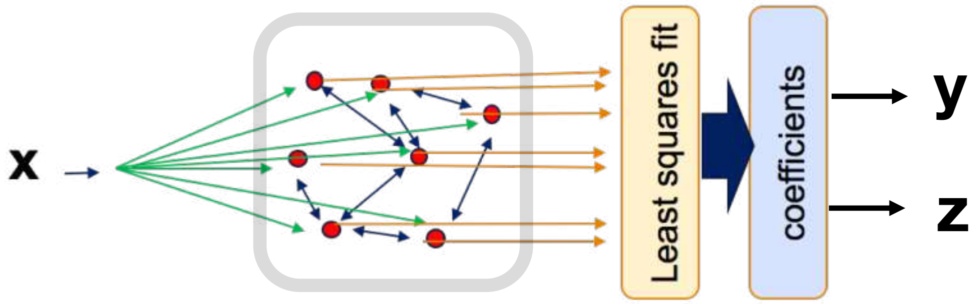
- **Sparse networks for the RC**
- **Test RC stability with iid input**
- **Need fading memory**

  stable, dissipative system = forget initial conditions

  flow is continuous and smooth C^1

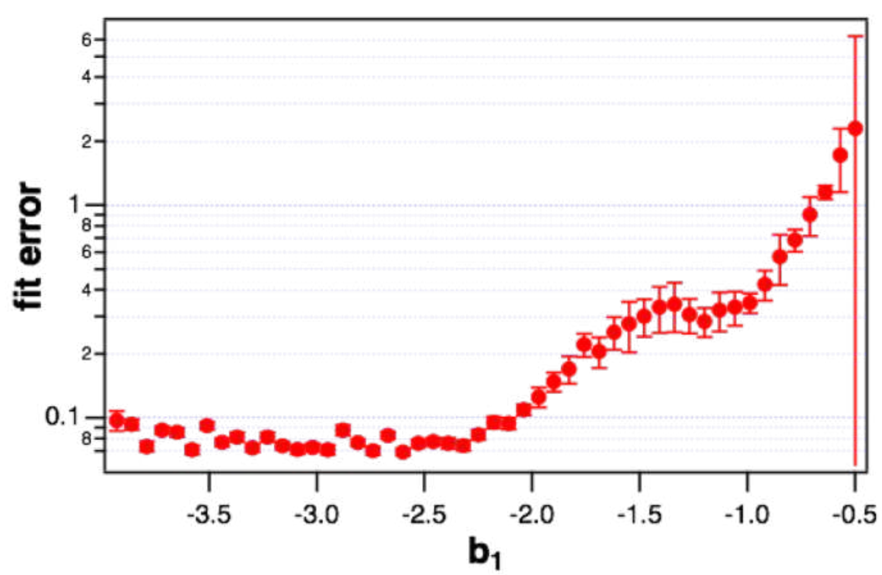
Some tests of various quantities in relation to quality of RC fidelity

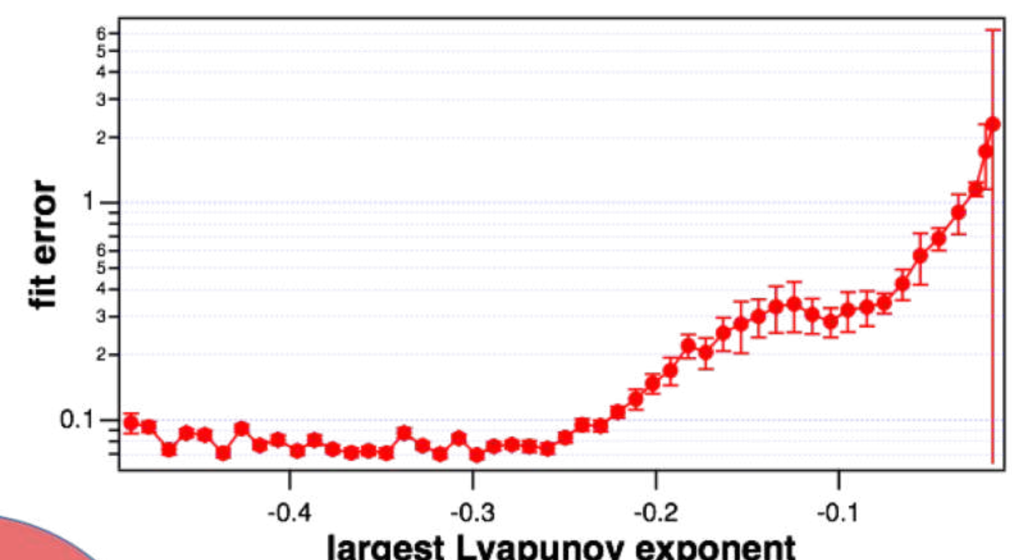
Operating near the “edge of chaos” - Tom Carroll (NRL)



Training

$$\frac{d\mathbf{R}}{dt} = \tau\left[b_1\mathbf{R} + b_2\mathbf{R}^2 + b_3\mathbf{R}^3 + \mathbf{AR} + \mathbf{W}\mathbf{x}\right]$$





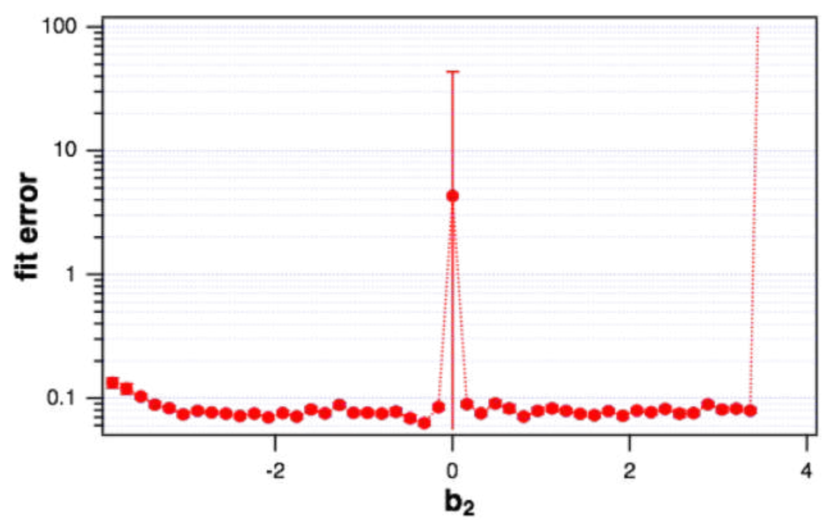
Edge of stability?

Sigmoid vector fields are not necessary

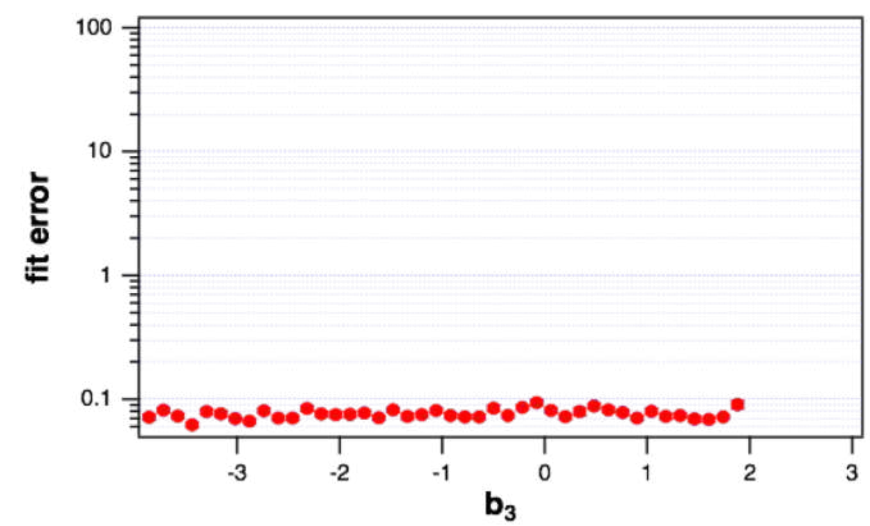
$$\frac{d\mathbf{R}}{dt} = \tau\left[b_1\mathbf{R} + b_2\mathbf{R}^2 + b_3\mathbf{R}^3 + \mathbf{AR} + \mathbf{W}s\right]$$

Effect of nonlinearity

Vary quadratic term



Vary cubic term

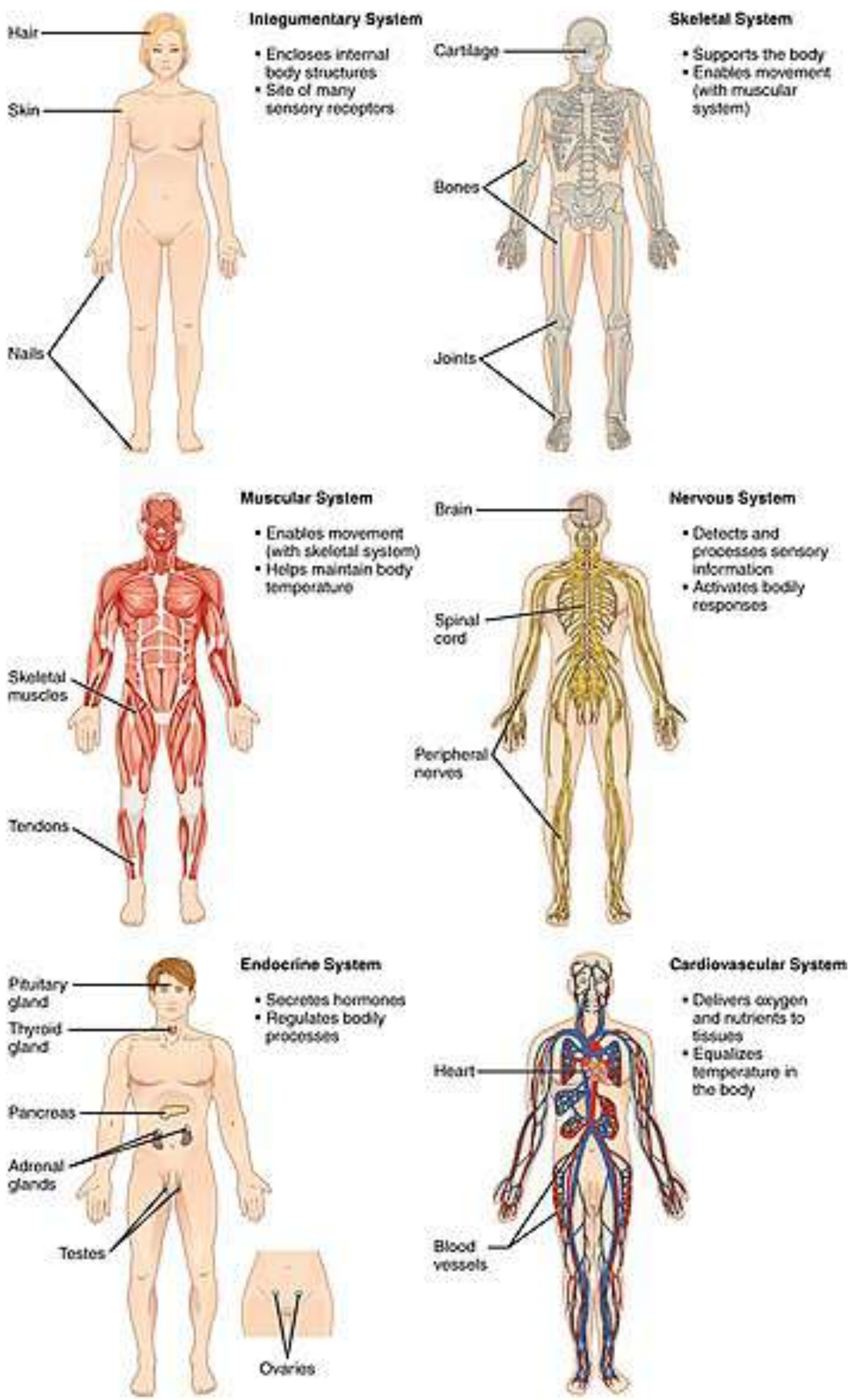


Quadratic nonlinearity necessary, but size not critical
Cubic nonlinearity not necessary

Tom Carroll (Naval Research Laboratory)

Can RC inform us about physiological systems?

physiological systems



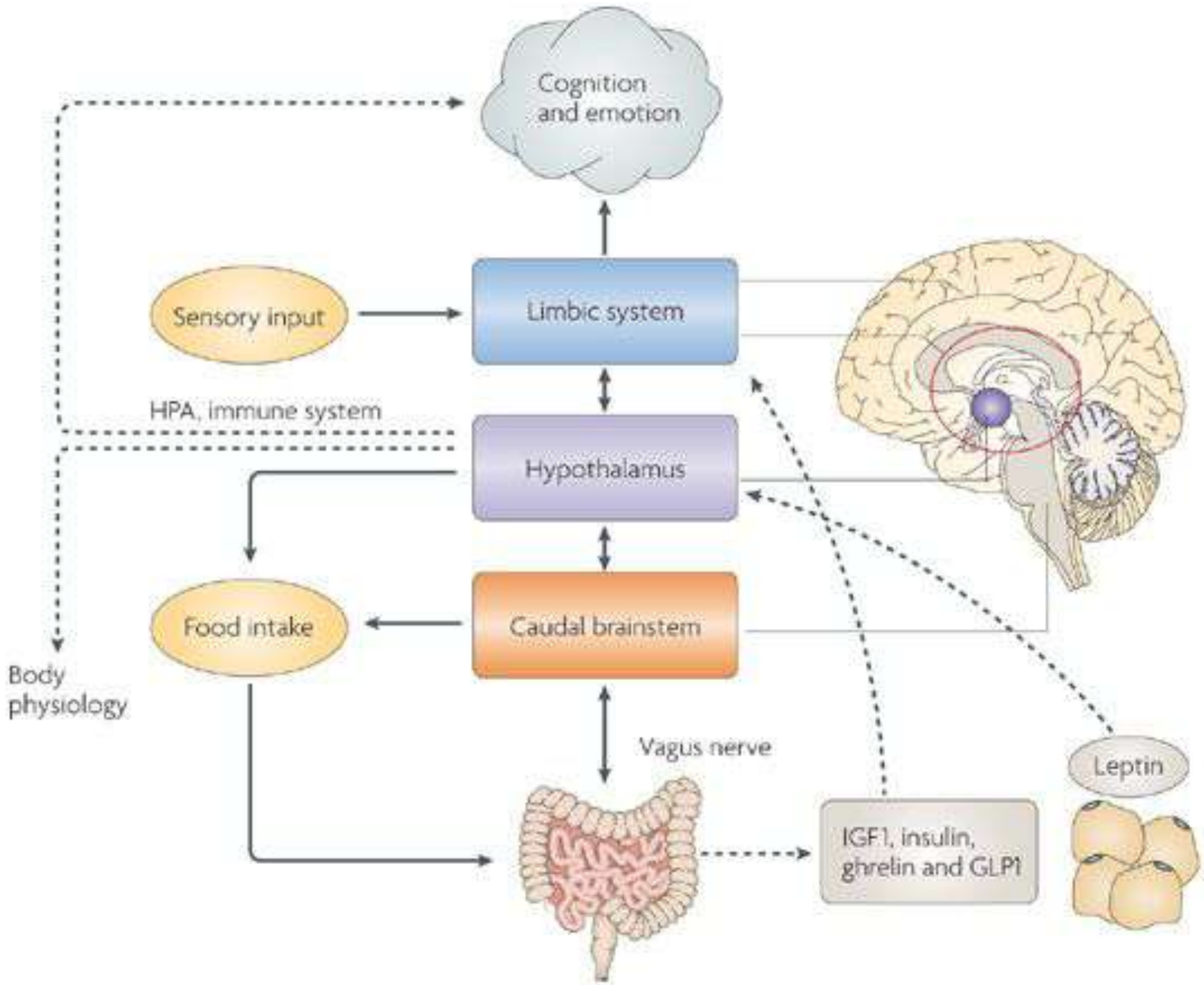
coupled and driven systems

coupled systems behave in a coordinated way [Synchronization]

driven systems behave in a consistent way [Generalized Synchronization]

Sensory systems

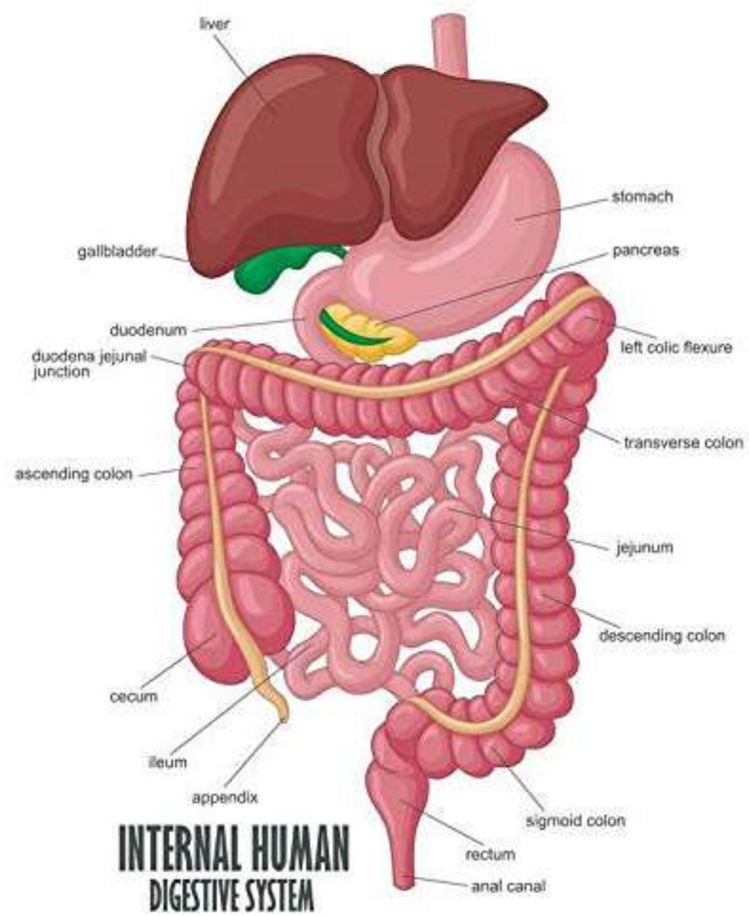
Sensory processing: organization of bodily sensations from the body itself and the environment, making it possible to for the body to operate effectively within the environment.



Physiological Systems

RC shows that a networks of nonlinear dynamical systems can react in consistent and reproducible ways to complex and very different inputs yielding consistent outputs that allow identification, enable accurate classification of inputs, and automatic reactions to the stimuli.

A model or metaphor for physiological systems?



Long ago ...

From L. Pecora and T. Carroll, Synchronization in Chaotic Systems, PRL, volume. 64, No. 8, 821 (1990)

Recent interesting results suggest the possibility of extending the synchronization concept to that of a metaphor for some neural processes. Freeman has suggested that one should view the brain response as an attractor. The process of synchronization can be viewed as a response system that "knows" what state (attractor) to go to when driven (stimulated) by a particular signal. It would be interesting to see whether this dynamical view could supplant the more "fixed-point" view of neural nets.

Questions? Comments?