AFRL

UNDERSTANDING AND LEVERAGING SYNCHRONIZATION IN RESERVOIR COMPUTING

ADRIAN S. WONG
JACOBS TECHNOLOGY GROUP,
IN-SPACE PROPULSION BRANCH (RQRS)

WORK WITH: DANIEL ECKHARDT

Roadmap:

- Problem and Motivation
  - High-dimensional system
  - Low-dimensional surrogate model
  - Reservoir Computing (RC)
- RC and Synchronization
  - “How” RC works
- Synchronization and Observers (Inference)
  - Inferring missing data
- Conclusions

System and Measurements

$\dot{x} = f(x; p)$ [ODE or PDE]

x : "state" of the system

p : parameters

$y = h(x)$+noise

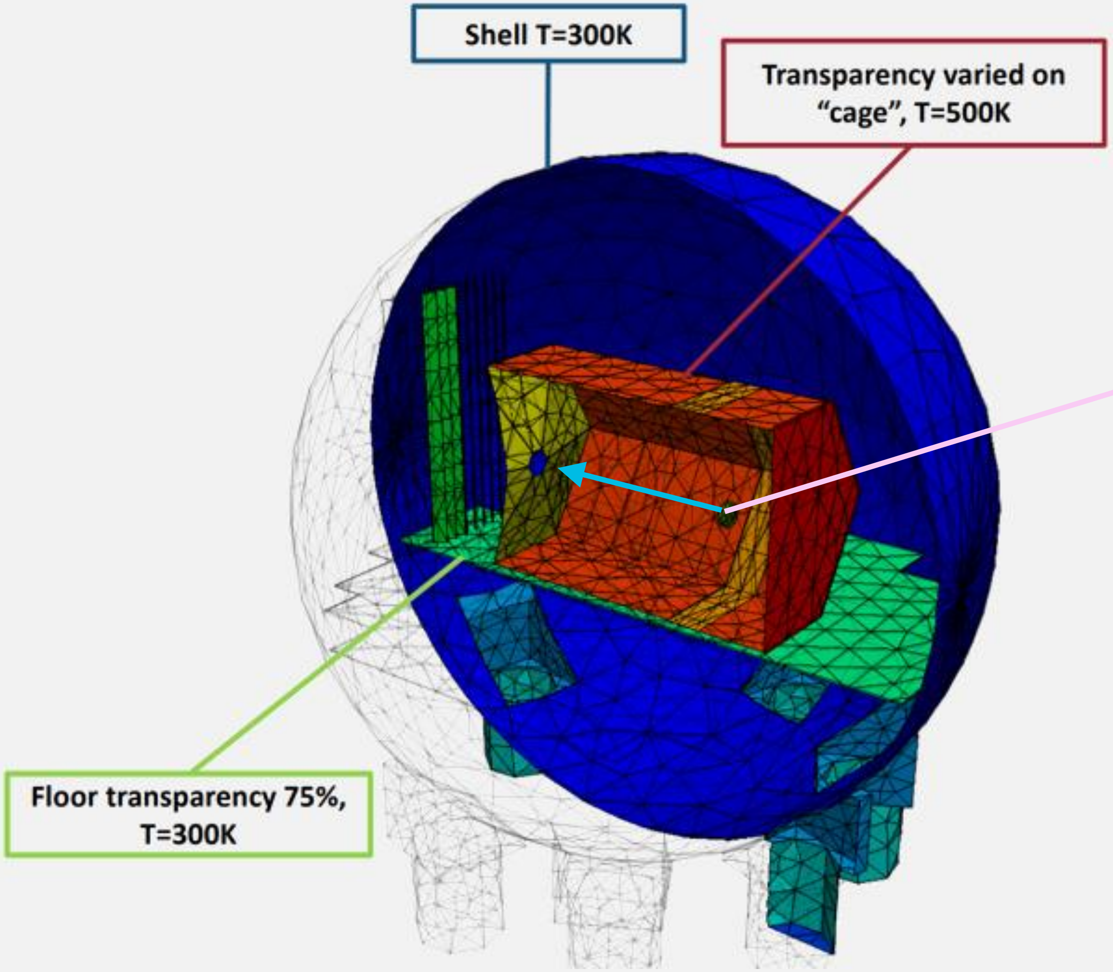
y : measurements

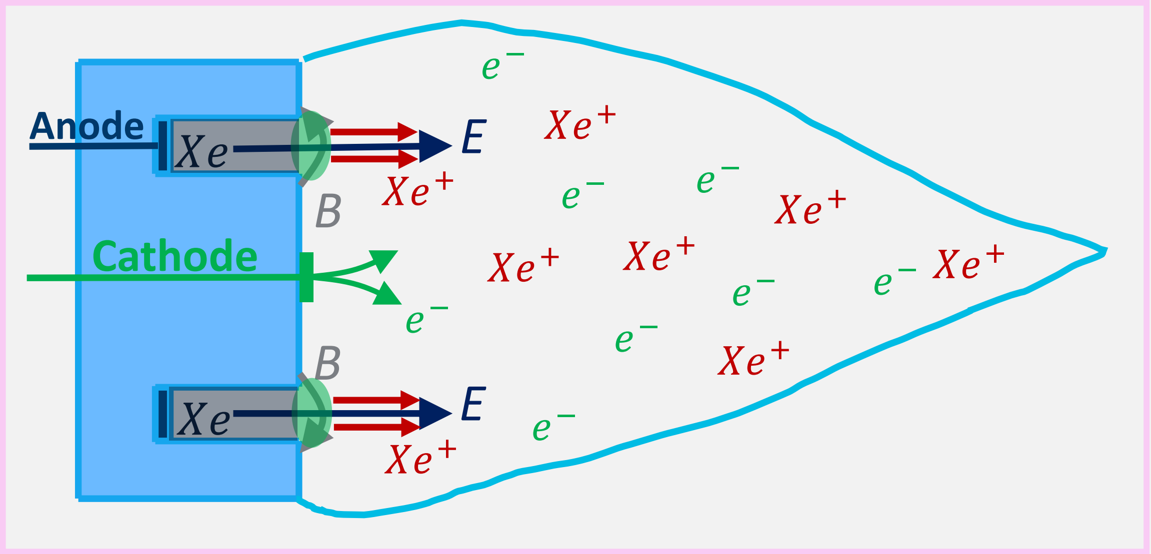
"State"
- Kinetic
- Fluid
- …
- ROMs
- Stat. Fit

Dimension/
Fidelity

Experimental Constraint: $\dim y \ll \dim x$

Hall-Effect Thruster Interacting with Cage



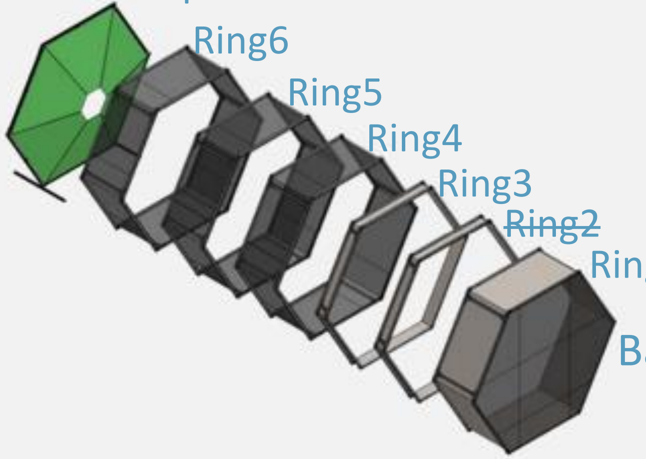


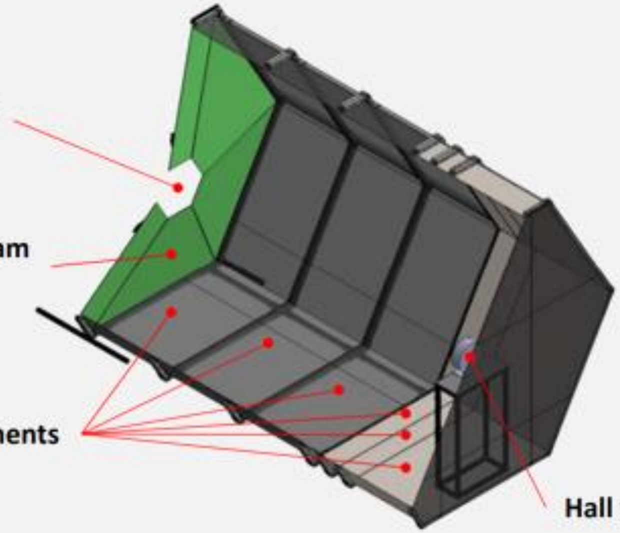
The facility and thruster are one <u>huge</u> dynamical system (plasma).

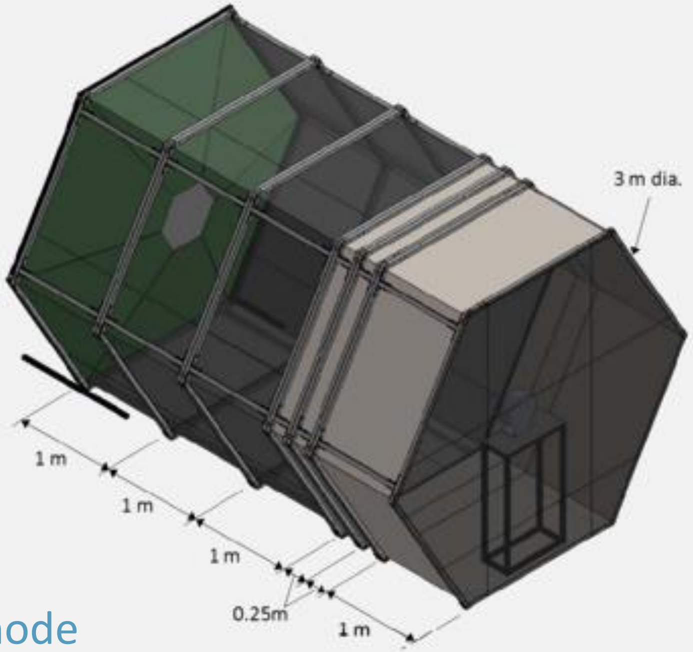
MacDonald, Natalia A. "Electric propulsion test and evaluation methodologies for plasma in the environments of space and testing (EP TEMPEST)." *AFOSR T&E Program Review* (2016).

Measurements

- "High Dimensional" Plasma
  - Complex system, multiscale, multiphysics
  - Difficult to model computationally
- 9 current measurements: y







MacDonald, Natalia A. "Electric propulsion test and evaluation methodologies for plasma in the environments of space and testing (EP TEMPEST)." *AFOSR T&E Program Review* (2016).

Learning Dynamics

Unknown

True Dynamics $\dot{x} = f(x; p)$

Measurement $y = h(x)$+noise

$$\left.\begin{array}{l} \dot{x} = f(x; p) \\ y = h(x) + \text{noise} \end{array}\right\} \quad y_{n+1} = F(y_n)$$

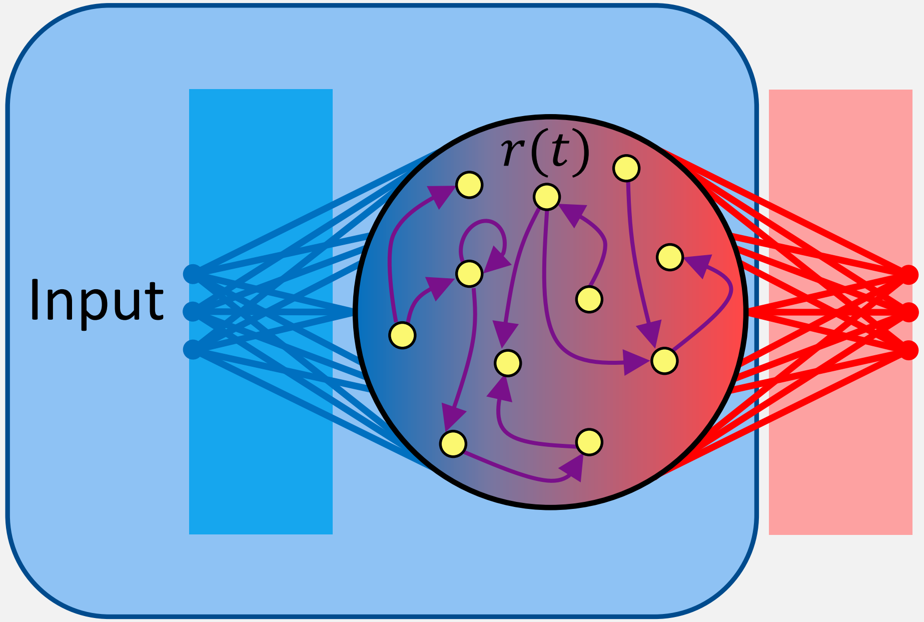
Effective Dynamics

(learn)

Given data $\{y\}$, find F such that $y_{n+1} = F(y_n)$ beyond measurement window.

"Can we predict future <u>measurements</u> from data alone?"

Reservoir Computing: Listening Phase (1/3)



Input Data: y

Reservoir State: r

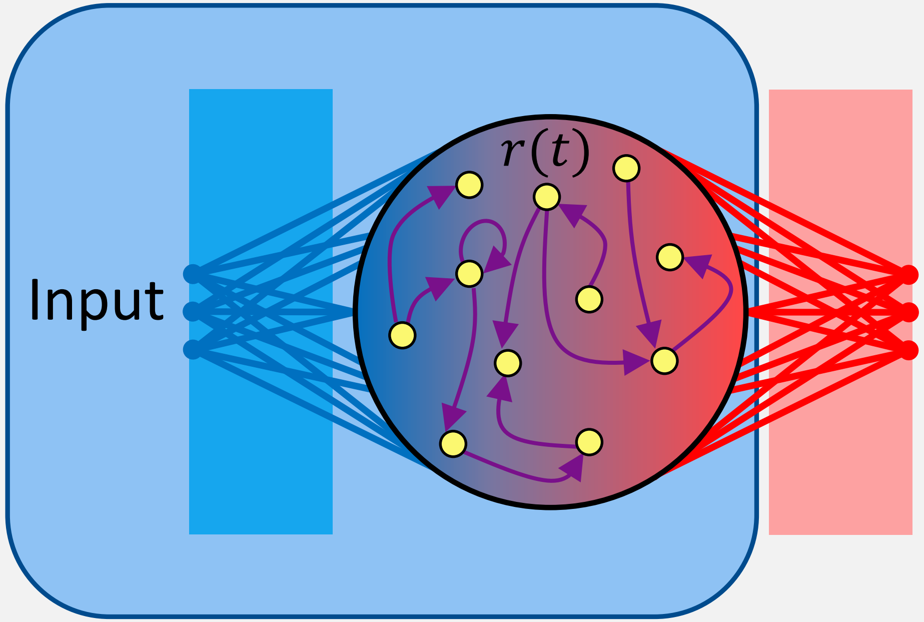
Driven Reservoir: $\dot{r} = -r + \tanh(Ar + By)$

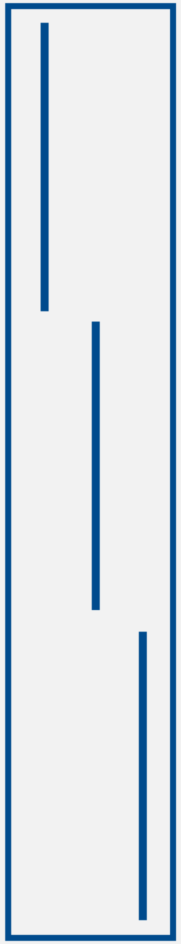
Integrate ($RK2$): $\{r(t_1), r(t_2), \dots, r(t_M)\}$

Set $r(t_1) = 0$. Stochastic Initialization* for A, B.

*A sparse (2%) and scaled so $\rho(A) \sim 1$

Reservoir Computing: Listening Phase (1/3)



$B =$ 

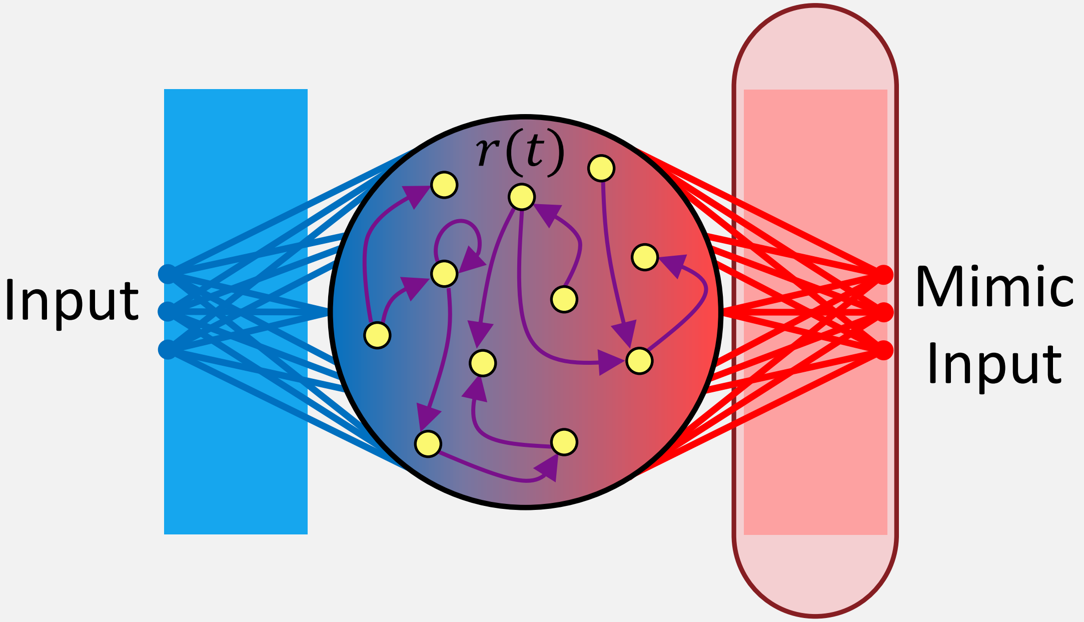
Input Data: y

Reservoir State: r

Driven Reservoir: $\dot{r} = -r + \tanh(Ar + By)$

Integrate ($RK2$): $\{r(t_1), r(t_2), \ldots, r(t_M)\}$

Reservoir Computing: Training Phase (2/3)

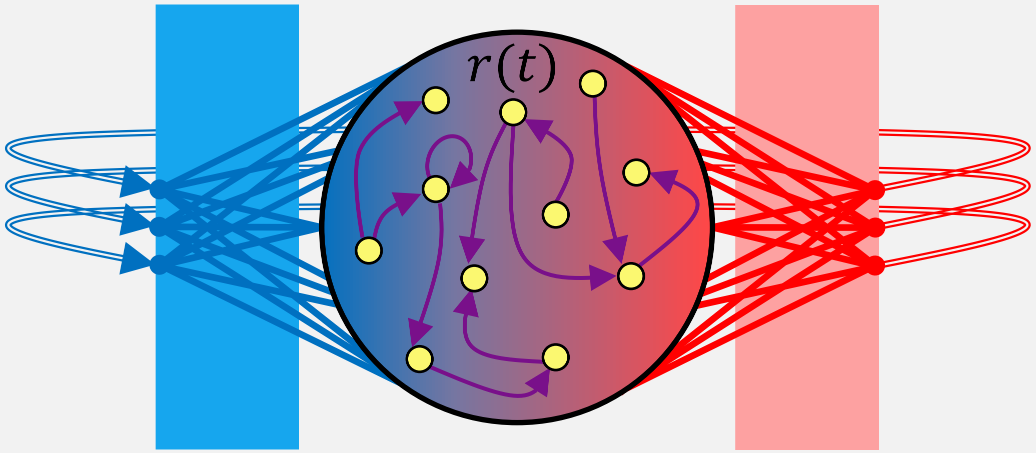


Input Data: y

Estimate: $\hat{y}$

Solve for W: $y \simeq \hat{y} \stackrel{\text{def}}{=} Wr \Longrightarrow \boxed{\underset{W}{\operatorname{argmin}} \|y - Wr\|^2}$

Reservoir Computing: Predicting Phase (3/3)



$\hat{y} = Wr$

Driven Reservoir: $\dot{r} = -r + \tanh(Ar + By)$

Predicting Reservoir: $\dot{r} = -r + \tanh([A + BW]r)$

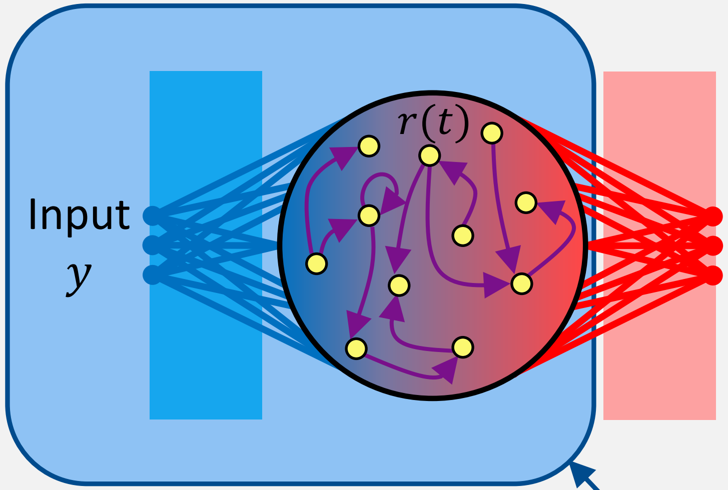
Integrate ($RK2$): $\{r(t_{M+1}), r(t_{M+2}), \dots\}$

Readout: $\hat{y} = Wr$

Compare $\hat{y}$ and y.

Reservoir Computing Phases

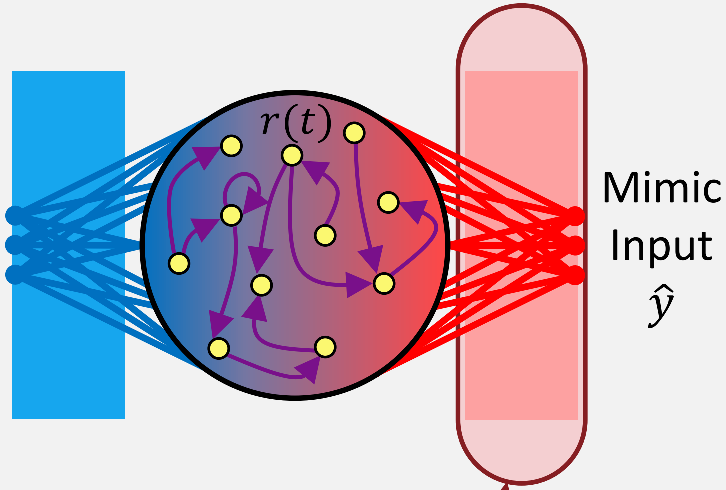
Listen



- Drive fixed system
- $\boldsymbol{A}$ and $\boldsymbol{B}$ arbitrary
- $\dot{r} = -r + \tanh(\boldsymbol{A}r + \boldsymbol{B}y)$

State-Contracting $\Longrightarrow$
Generalized Synchronization

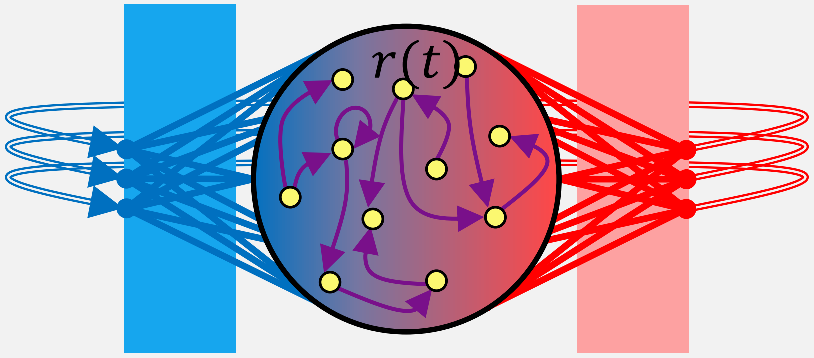
Train



Solve for W

- $y \simeq \hat{y} = Wr$
- Linear, fast

Predict

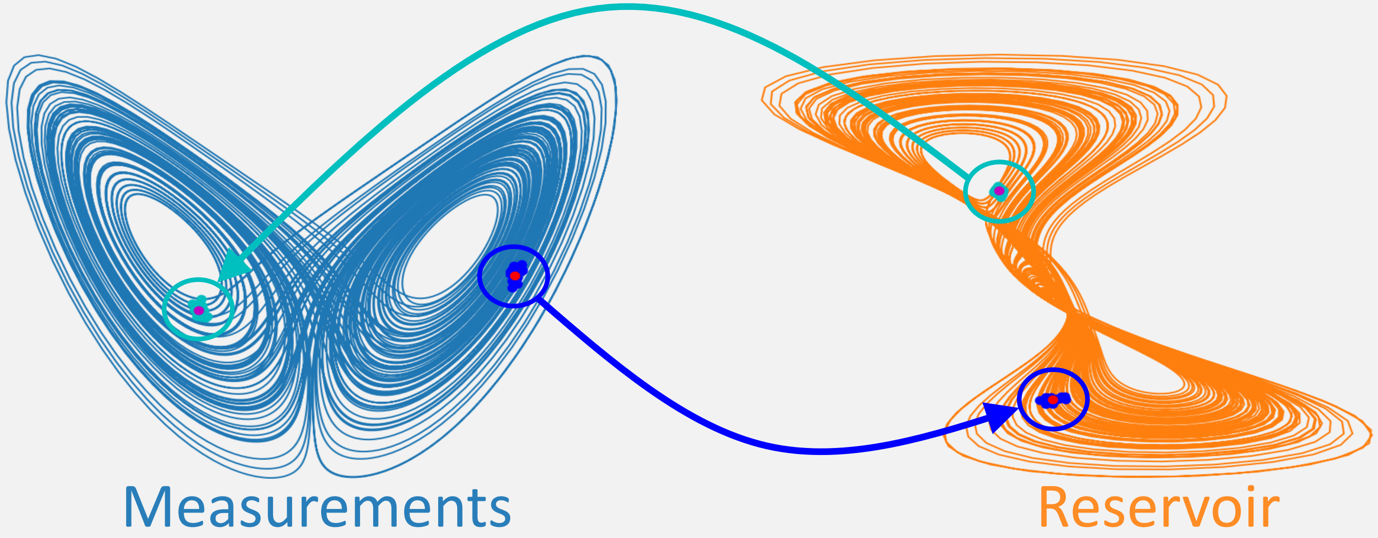


Close the feedback loop

- $\dot{r} = -r + \tanh(Ar + By)$
  $\simeq -r + \tanh(Ar + B\hat{y})$
  $= -r + \tanh([A + BW]r)$

Universal Approx.

Unique Continuous Map across Attractors



Generalized Synchronization

$$\Psi(\mathcal{A}_y) = \mathcal{A}_r$$
$$\text{as } t \to \infty$$

Conjecture:

If $\dim r > 2\dim y$, then Ψ is almost-surely an embedding.

Hart, A., Hook, J., & Dawes, J. (2020). Embedding and approximation theorems for echo state networks. Neural Networks, 128, 234-247.

Unique Continuous Map across Attractors

Measurements

Reservoir

Generalized Synchronization

$$\Psi(\mathcal{A}_y) = \mathcal{A}_r$$
$$\text{as } t \to \infty$$

<u>Conjecture:</u>

If $\dim r > 2 \dim y$, then Ψ is almost-surely an embedding.

Hart, A., Hook, J., & Dawes, J. (2020). Embedding and approximation theorems for echo state networks. Neural Networks, 128, 234-247.

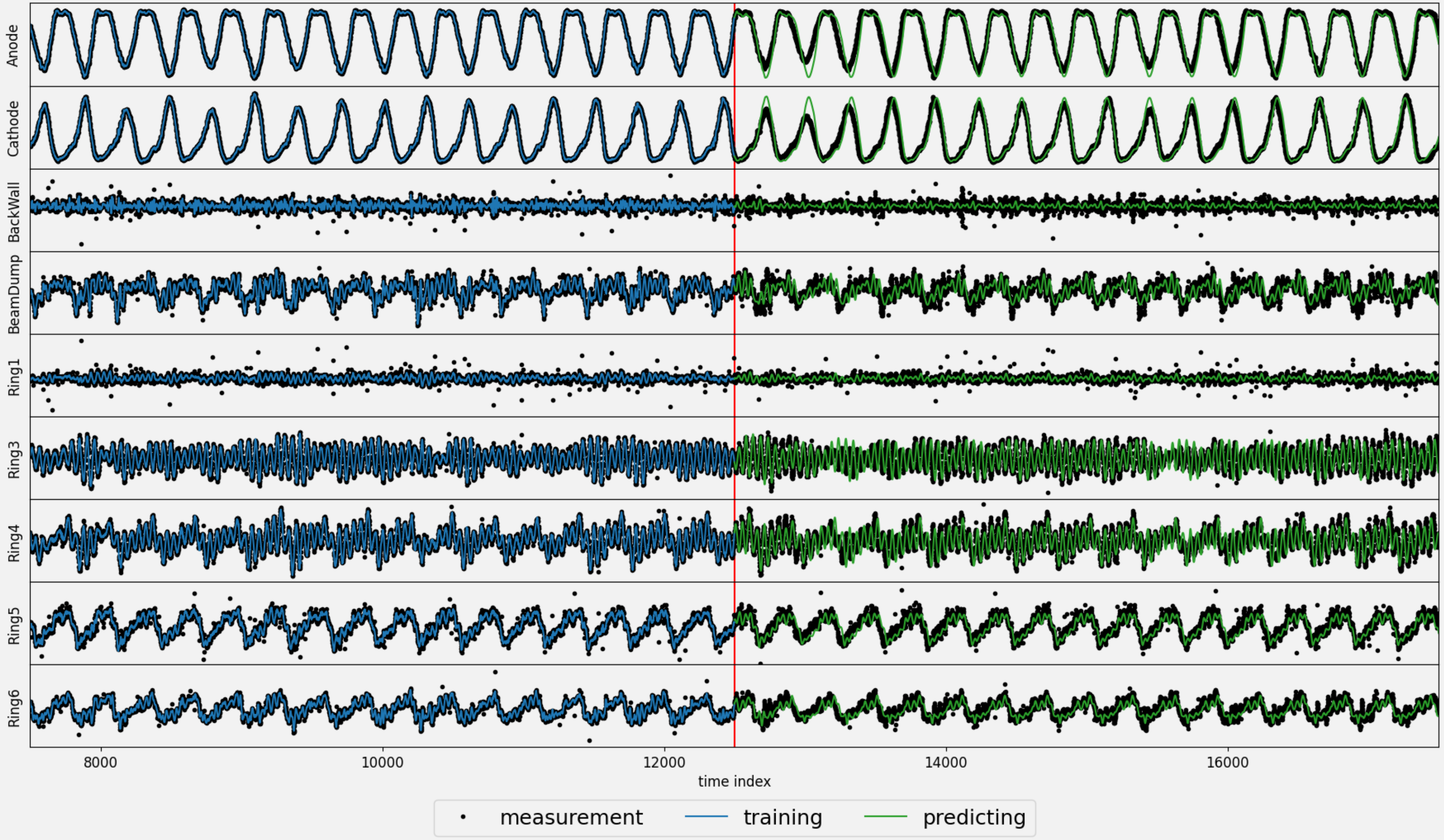
Role of Each Phase

$$
\begin{tikzcd}
\mathcal{A}_y \arrow[r, "\Phi"] \arrow[d, "\Psi"'] & \mathcal{A}_y & & \mathcal{A}_y \arrow[r, "\Phi"] \arrow[d, "\Psi"'] & \mathcal{A}_y \\
\mathcal{A}_r \arrow[r, "\Gamma"] & \mathcal{A}_r \arrow[u, "\Psi^{-1}"'] \arrow[r, phantom, "\simeq"] & & \mathcal{A}_r \arrow[r, Rightarrow, "\bar{\sigma}_W"] & \mathcal{A}_r \arrow[u, Rightarrow, "W"']
\end{tikzcd}
$$

1. Setup: Arbitrary reservoir implies arbitrary map Ψ.
2. Listen: Driven system samples pairs $\{y(t), r(t)\}$ such that $\Psi\big(y(t)\big) = r(t)$.
3. Train: Find (linear) **approx**. W s.t. $y(t) = \Psi^{-1}\big(r(t)\big) \simeq Wr(t)$.
4. Predict: Get $\bar{\sigma}_W$ “for free” from 2. and 3.

RC as Surrogate Model

- 9 measurements
- 12,500 examples
- ~2.6 sec run time
  - Python, (old) desktop
- $7209 \simeq 85^2$ weights
  - But 400 nodes/neurons
- Disclaimer:
  - Predictions “decay” to periodic
  - Tune-by-hand necessary
  - RK2: $\Delta t = 0.22$



Leveraging Synchronization of RC

Identical Synchronization

VOLUME 64, NUMBER 8 PHYSICAL REVIEW LETTERS 19 FEBRUARY 1990

Synchronization in Chaotic Systems

Louis M. Pecora and Thomas L. Carroll
Code 6341, Naval Research Laboratory, Washington, D.C. 20375
(Received 20 December 1989)

- Certain nearly-identical systems can synchronize
- (Identical) Synchronization:
  - Two (nearly identical, possibly chaotic) systems with different I.C. driven by the same input approach the same solution
- Robust/stable in the presence of:
  - Measurement noise
  - Parameter mismatch

Experimental data

Applicable to surrogate models!

“Complete Replacement”

Fundamentals of synchronization in chaotic systems, concepts, and applications

Louis M. Pecora, Thomas L. Carroll, Gregg A. Johnson, and Douglas J. Mar
Code 6343, U.S. Naval Research Laboratory, Washington, District of Columbia 20375

James F. Heagy
Institutes for Defense Analysis, Science and Technology Division, Alexandria, Virginia 22311-1772

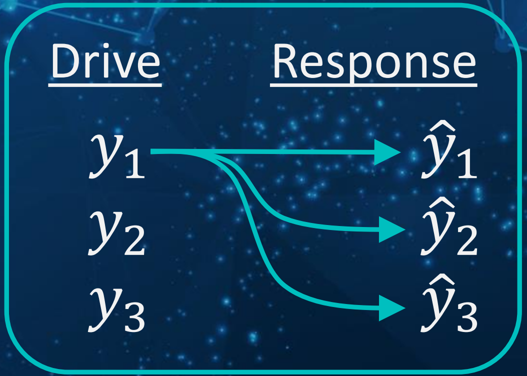
(Received 29 April 1997; accepted for publication 29 September 1997)

Drive System y

$$\dot{y}_1 = -\sigma(y_2 - y_1)$$
$$\dot{y}_2 = y_1(\rho - y_3) - y_2$$
$$\dot{y}_3 = y_1 y_2 - \beta y_3$$

Response System y

$$\dot{\hat{y}}_1 = -\sigma(\hat{y}_2 - y_1)$$
$$\dot{\hat{y}}_2 = y_1(\rho' - \hat{y}_3) - \hat{y}_2$$
$$\dot{\hat{y}}_3 = y_1 \hat{y}_2 - \beta' \hat{y}_3$$



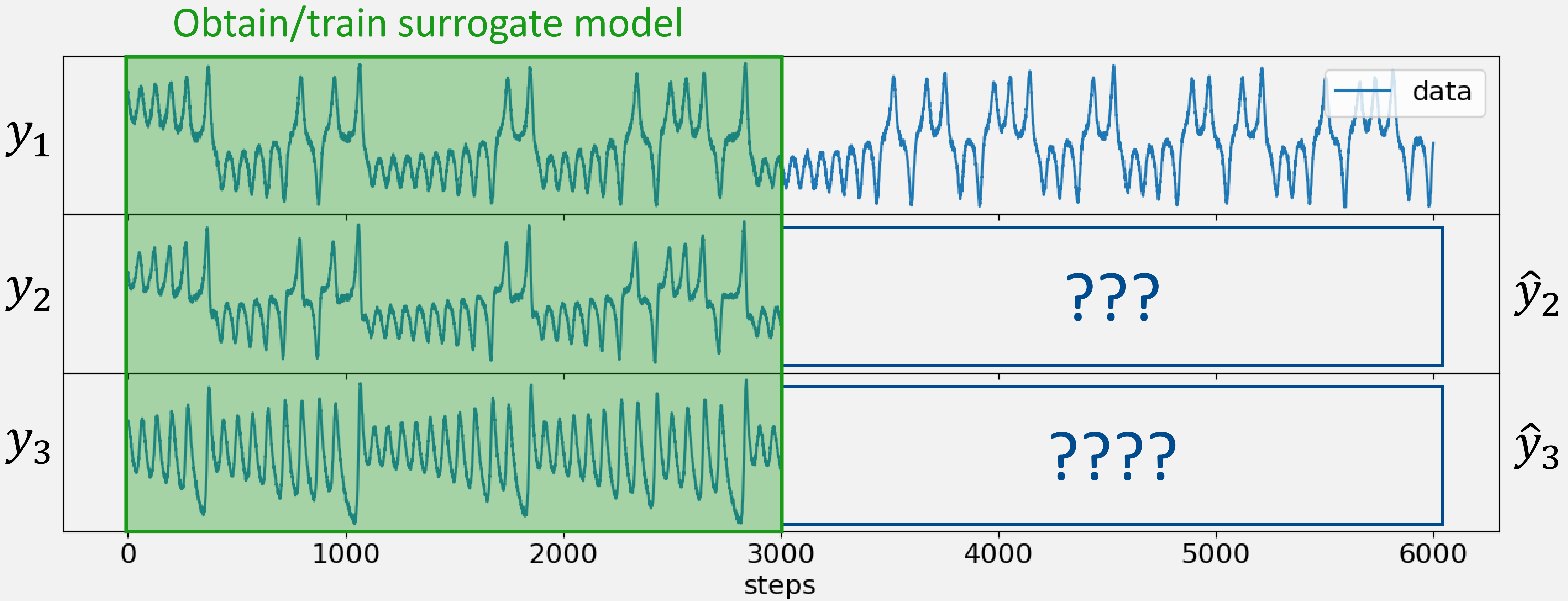
Noiseless and Identical:

$$\lim_{t\to\infty} y_{2,3} - \hat{y}_{2,3} = 0$$

Noisy and/or Param. Mismatch:

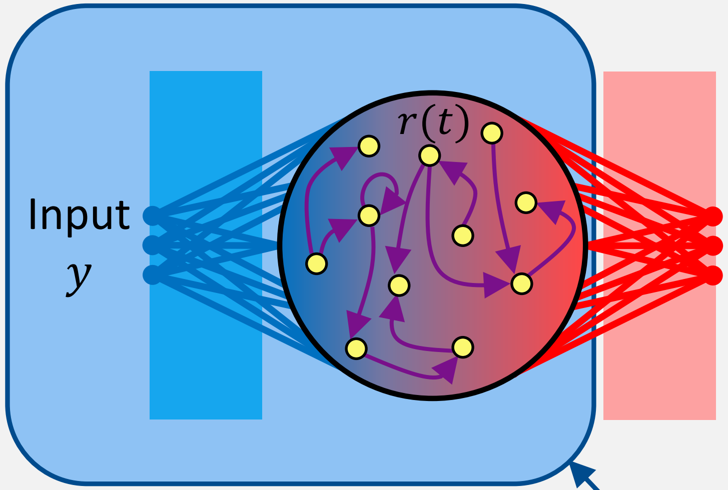
$$\lim_{t\to\infty} \left\| y_{2,3} - \hat{y}_{2,3} \right\| \le \epsilon$$

Synchronization for Observers (Inference)



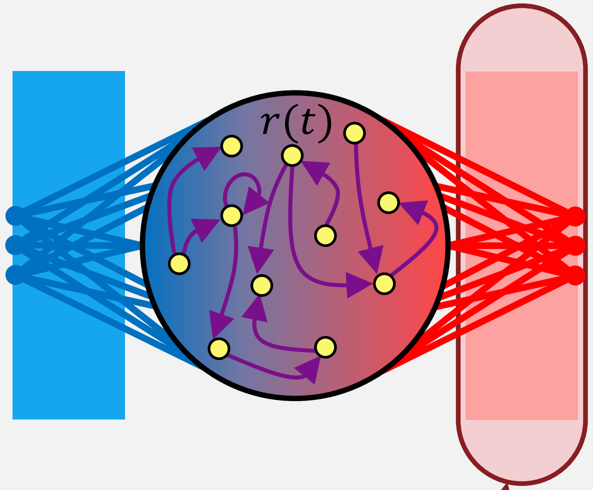
Reservoir Computing Phases

Listen



- Drive fixed system
- $\boldsymbol{A}$ and $\boldsymbol{B}$ arbitrary
- $\dot{r} = -r + \tanh(\boldsymbol{A}r + \boldsymbol{B}y)$

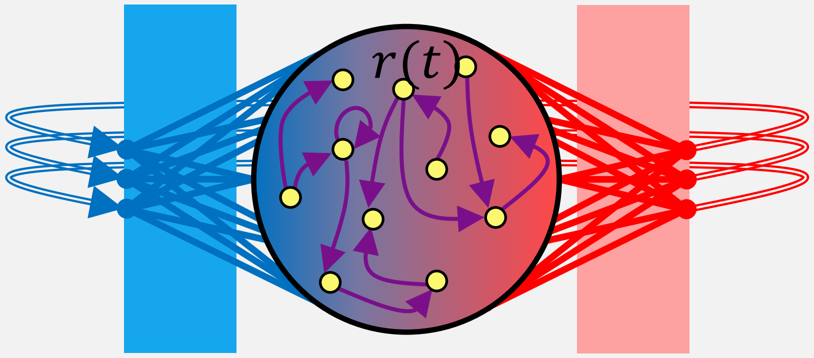
Train



Solve for W

- $y \simeq \hat{y} = Wr$
- Linear, fast

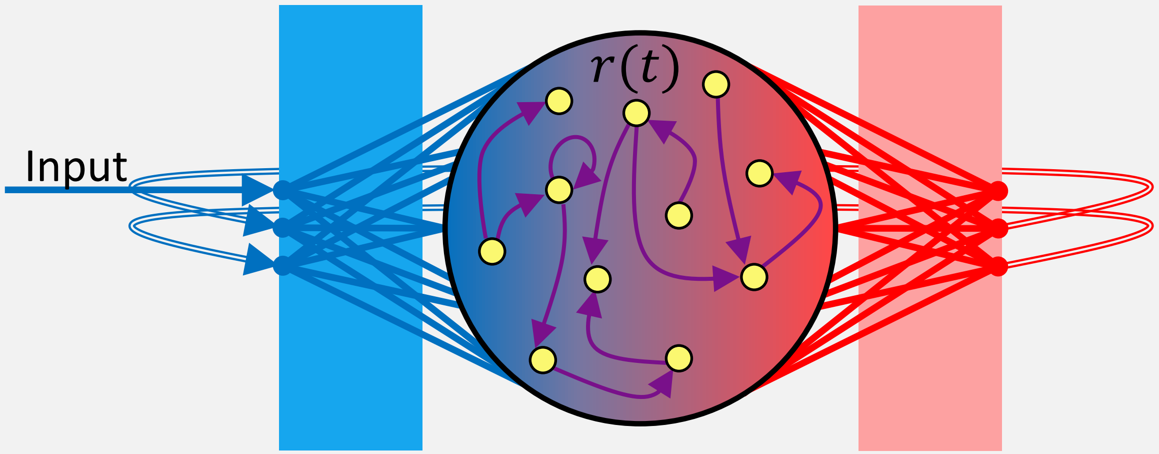
Predict



Close the feedback loop

- $$\begin{aligned} \dot{r} &= -r + \tanh(Ar + \boxed{By}) \\ &\simeq -r + \tanh(Ar + \boxed{B\hat{y}}) \\ &= -r + \tanh([A + BW]r) \end{aligned}$$

Reservoir Computing: Inferring Phase



$z = [y_1, \hat{y}_2, \hat{y}_3]^{\mathsf{T}}$

Driven Reservoir: $\dot{r} = -r + \tanh(Ar + By)$

Inferring Reservoir: $\dot{r} = -r + \tanh(Ar + Bz)$

Integrate ($RK2$): $\{r(t_{M+1}), r(t_{M+2}), \ldots\}$

Readout: $\hat{y} = Wr$

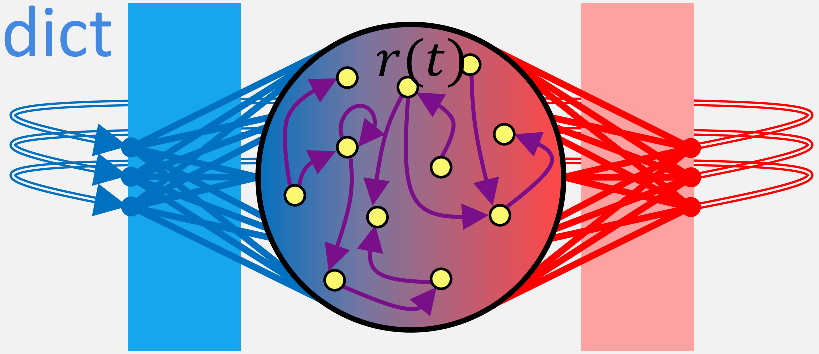
Compare z and y.

Synchronization for Observers (Inference)

- Integrate with $[y_1, \hat{y}_2, \hat{y}_3]$
- Approaches $[y_1, y_2, y_3]$
  - Even when $\hat{y}_2, \hat{y}_3$ have moderate noise
  - Even with slight system mismatch

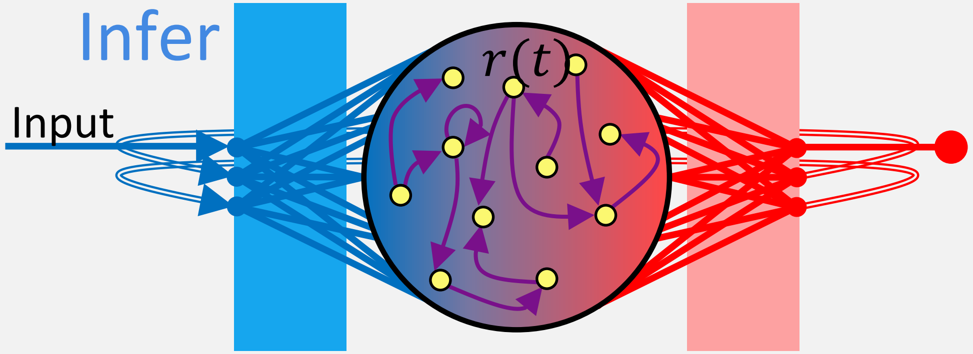
Lu, Z., Pathak, J., Hunt, B., Girvan, M., Brockett, R., & Ott, E. (2017). Reservoir observers: Model-free inference of unmeasured variables in chaotic systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *27*(4).

Predict



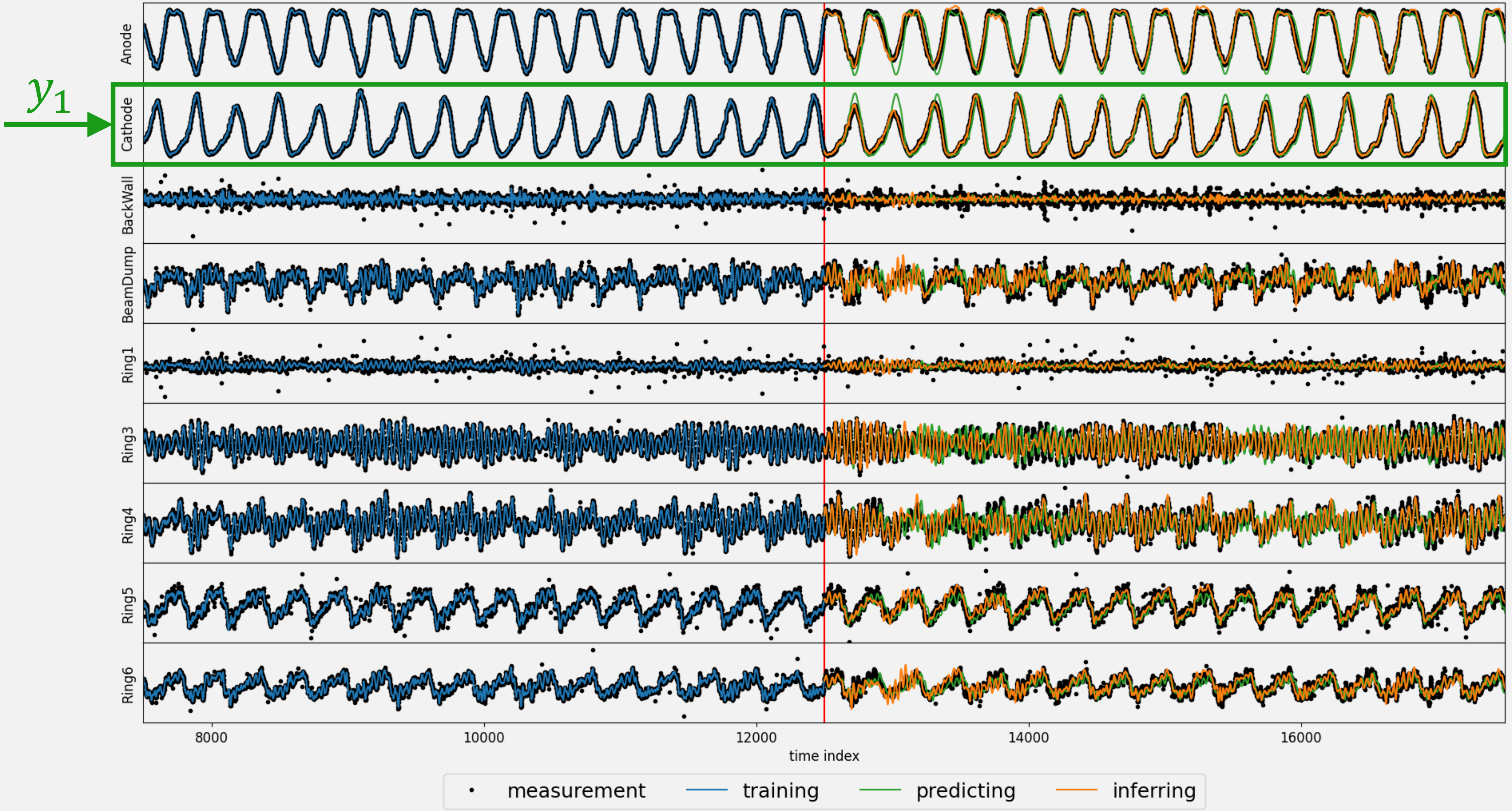
$$\dot{r} = -r + \tanh(Ar + B\hat{y})$$

Infer



$$z = [y_1, \hat{y}_2, \hat{y}_3]^{\top}$$
$$\dot{r} = -r + \tanh(Ar + Bz)$$

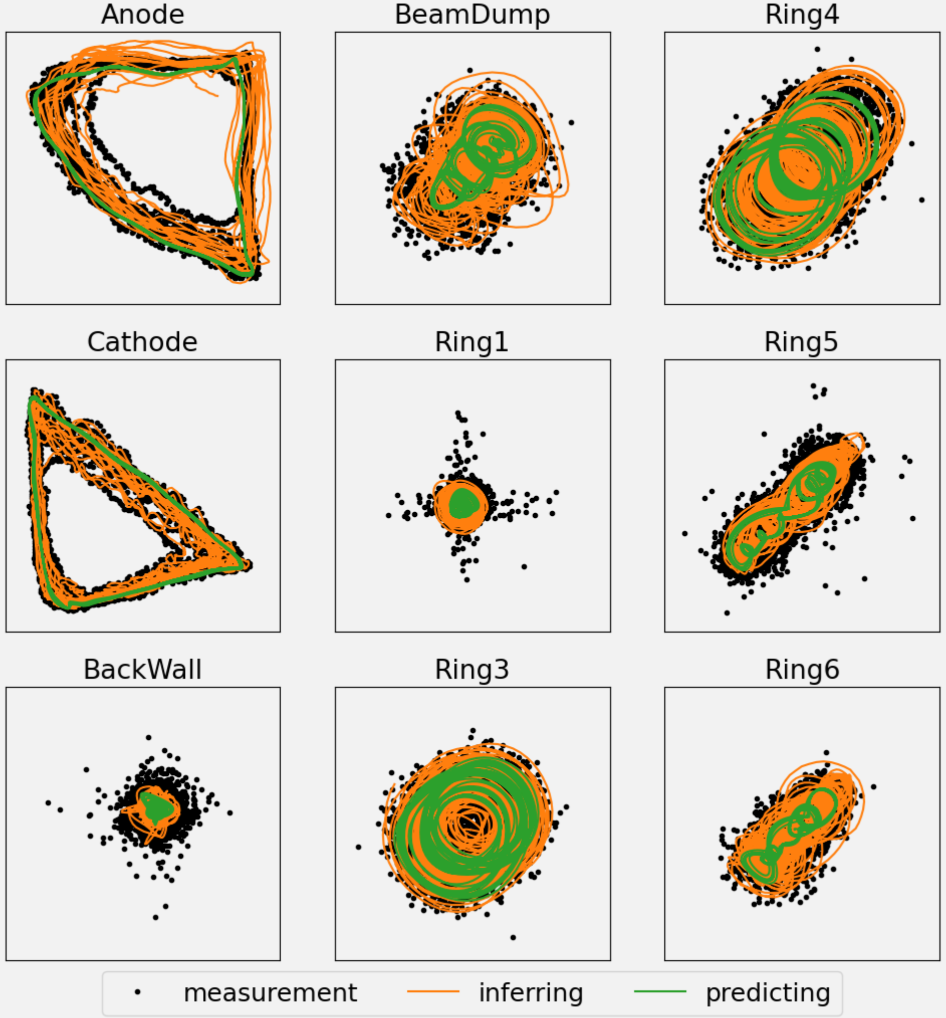
Predicting v. Inferring

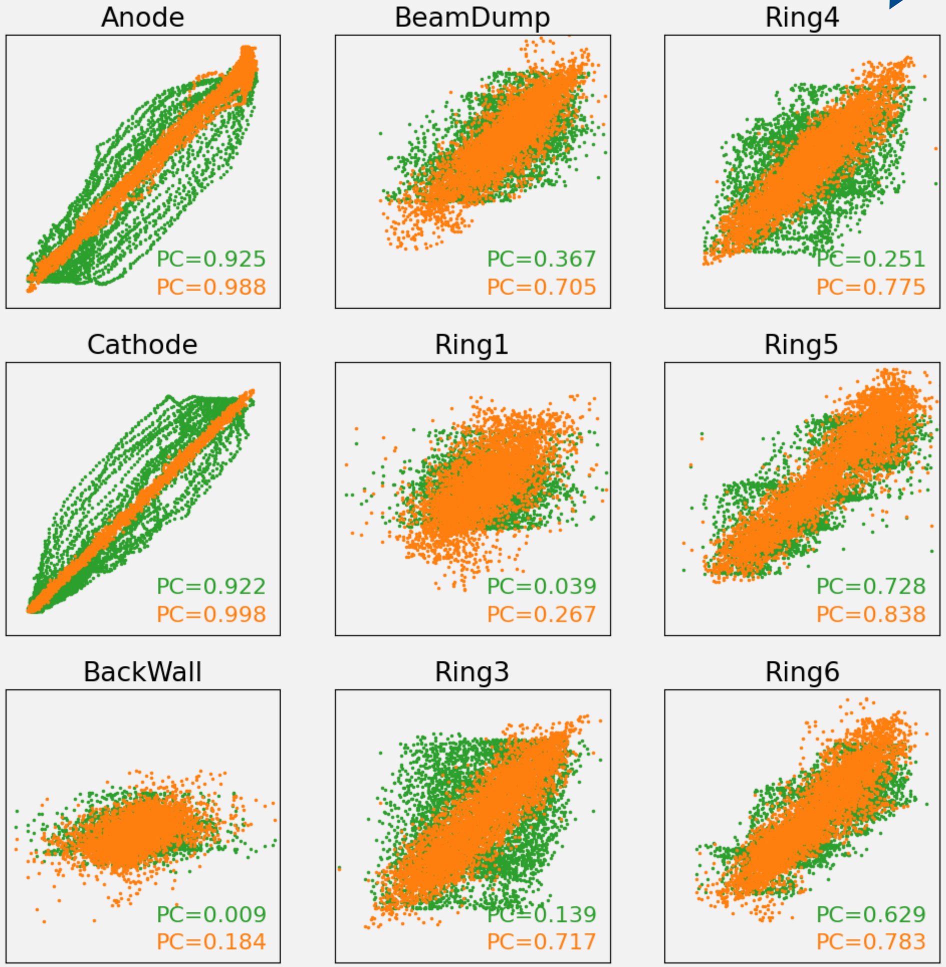


Predicting v. Inferring

"Synchronization" Plots

Time delay attractor





Conclusions:

- Thruster+Cage as a dynamical system
  - Sparsely measured
  - Find effective dynamical system
- Role of sync. in RC
  - Unique, continuous → generates samples
  - Learn inverse with samples
  - Close-loop → “self-consistent” prediction
- Sync. of surrogate models
  - Maintains sync. with noisy data
  - Allows for use as observer
  - Mechanism for low-fidelity Digital Twin (?)
  - Small footprint → Neuromorphic (RI?)

- Acknowledgments: Robert Martin, Justin Koo

Ongoing/Potential work:

- Extension to PDE data
- Pairing with Hi-Fi models
- Pruning of dynamics
- Time delay “preconditioner”
- RC for Controls
- Quantum RC ☺