MACHINE LEARNING FOR ANALYSIS OF HIGH-DIMENSIONAL CHAOTIC SPATIOTEMPORAL DYNAMICAL SYSTEMS

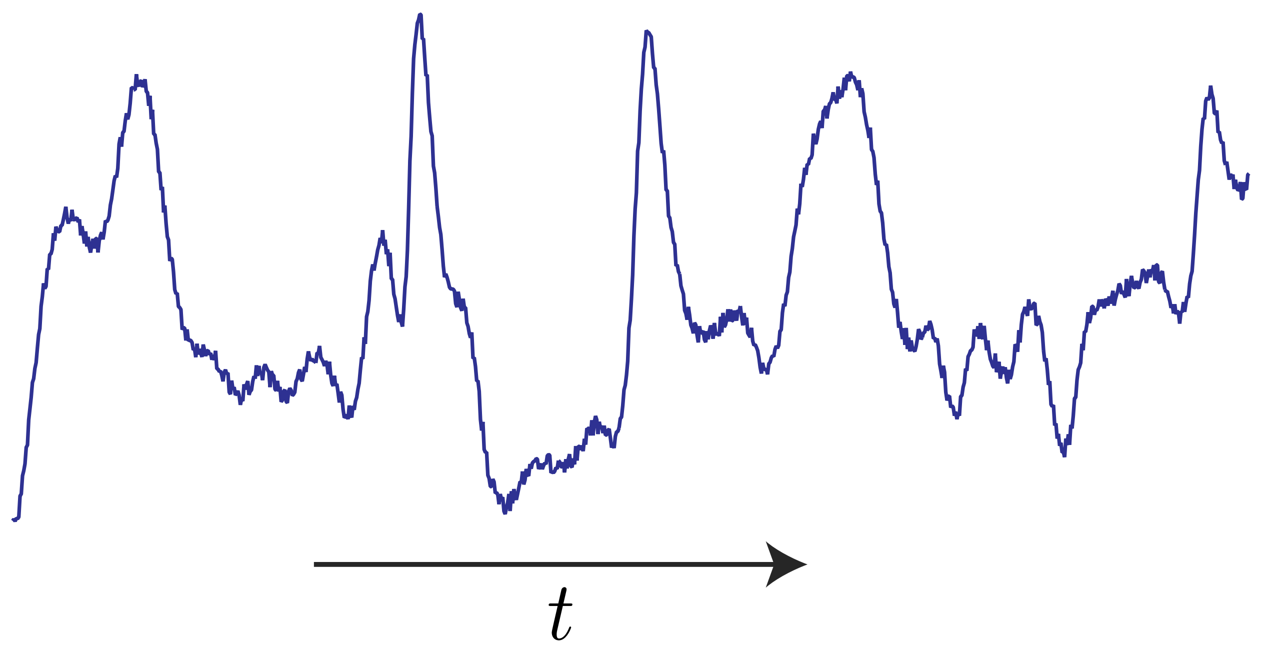
6th December 2018
Princeton Plasma Physics Laboratory Theory Seminar

Jaideep Pathak
Zhixin Lu, Alex Wikner, Rebeckah Fussell, Brian Hunt, Michelle Girvan, Edward Ott
University of Maryland, College Park

Given some data from a dynamical system, what can we say about the dynamical process that generated this data?

?

$$\boxed{\frac{d\mathbf{x}}{dt} = \mathbf{F}(\mathbf{x}, t)}$$



OUTLINE

Short-term forecasting

Prediction

Given a limited time series of past measurements can we predict the future state of the dynamical system, at least in the short term?

Long-term dynamics

Reconstructing the attractor

Can we learn something about the long-term dynamics of the system? Specifically, can we use our setup to understand the ergodic properties of a high-dimensional dynamical system?

Large systems

Scalability

Can we use our setup for very high-dimensional attractors?

INTRODUCTION

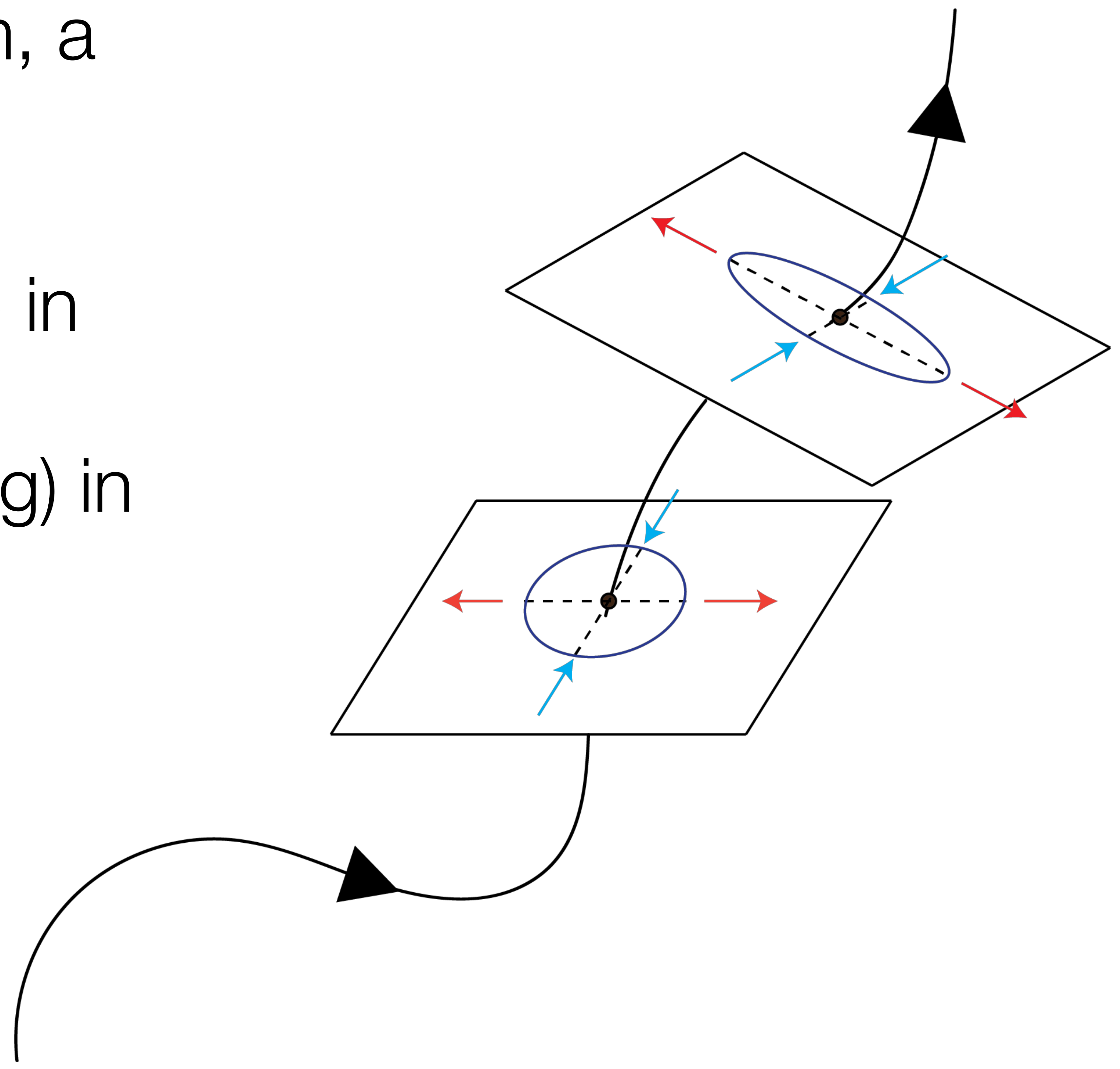
Operationally, we say a dynamical system is chaotic if, in a bounded phase space, two nearby trajectories diverge exponentially.

$$\mathbf{x}_1(t) - \mathbf{x}_2(t) = \delta\mathbf{x}(t)$$

$$\|\delta\mathbf{x}(t)\| \sim \|\delta\mathbf{x}(0)\| \exp(\Lambda t)$$

INTRODUCTION

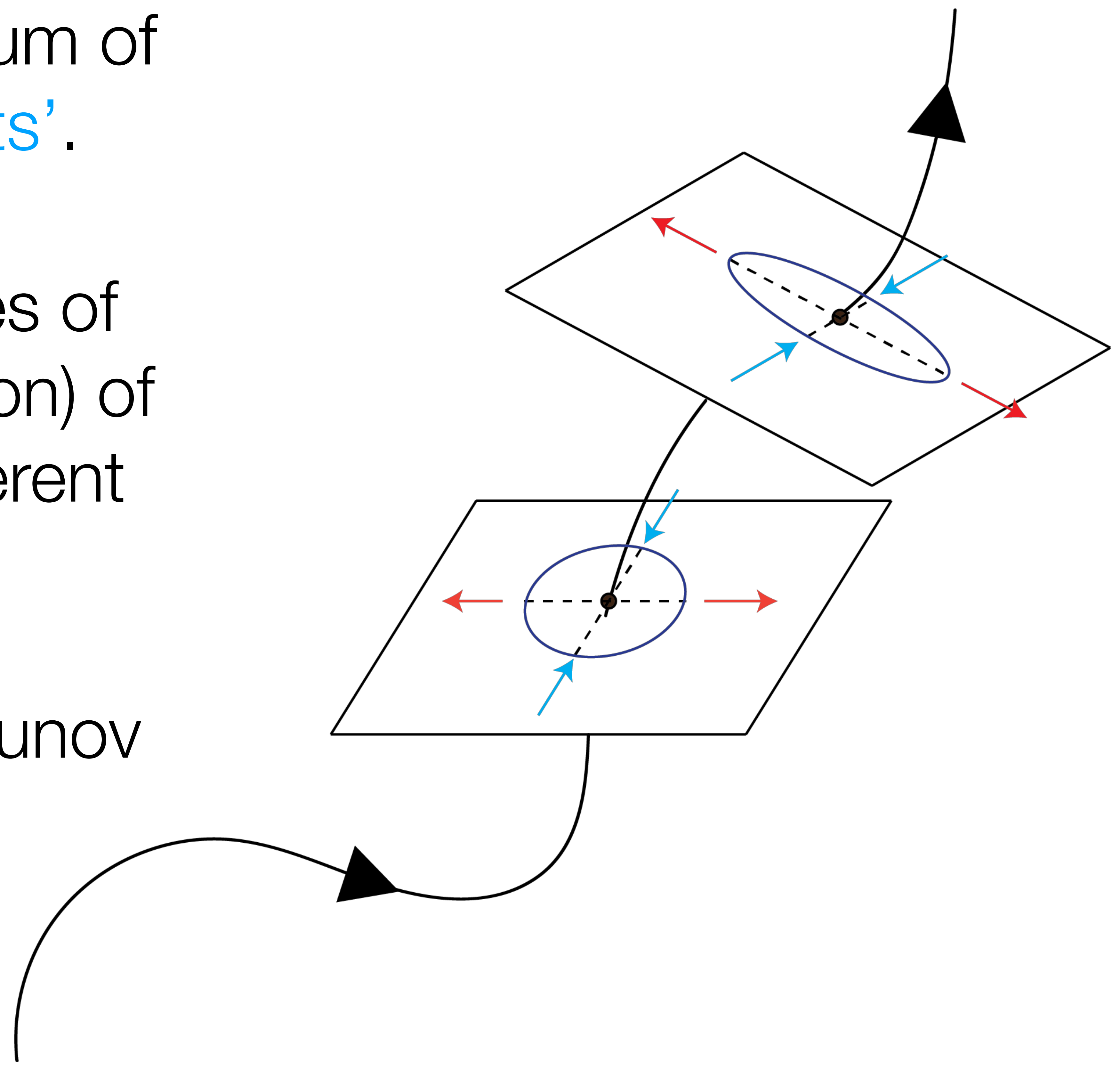
In a chaotic system, a perturbation to the trajectory may be stable (contracting) in some directions, unstable (expanding) in some directions.



INTRODUCTION

This leads to the concept of a spectrum of ‘Lyapunov exponents’.

The exponential rates of growth (or contraction) of perturbations in different directions are characterized by corresponding Lyapunov exponents.



INTRODUCTION

The Lyapunov exponents of a dynamical system are an important characteristic and provide us with a lot of useful information.

Is a system chaotic?

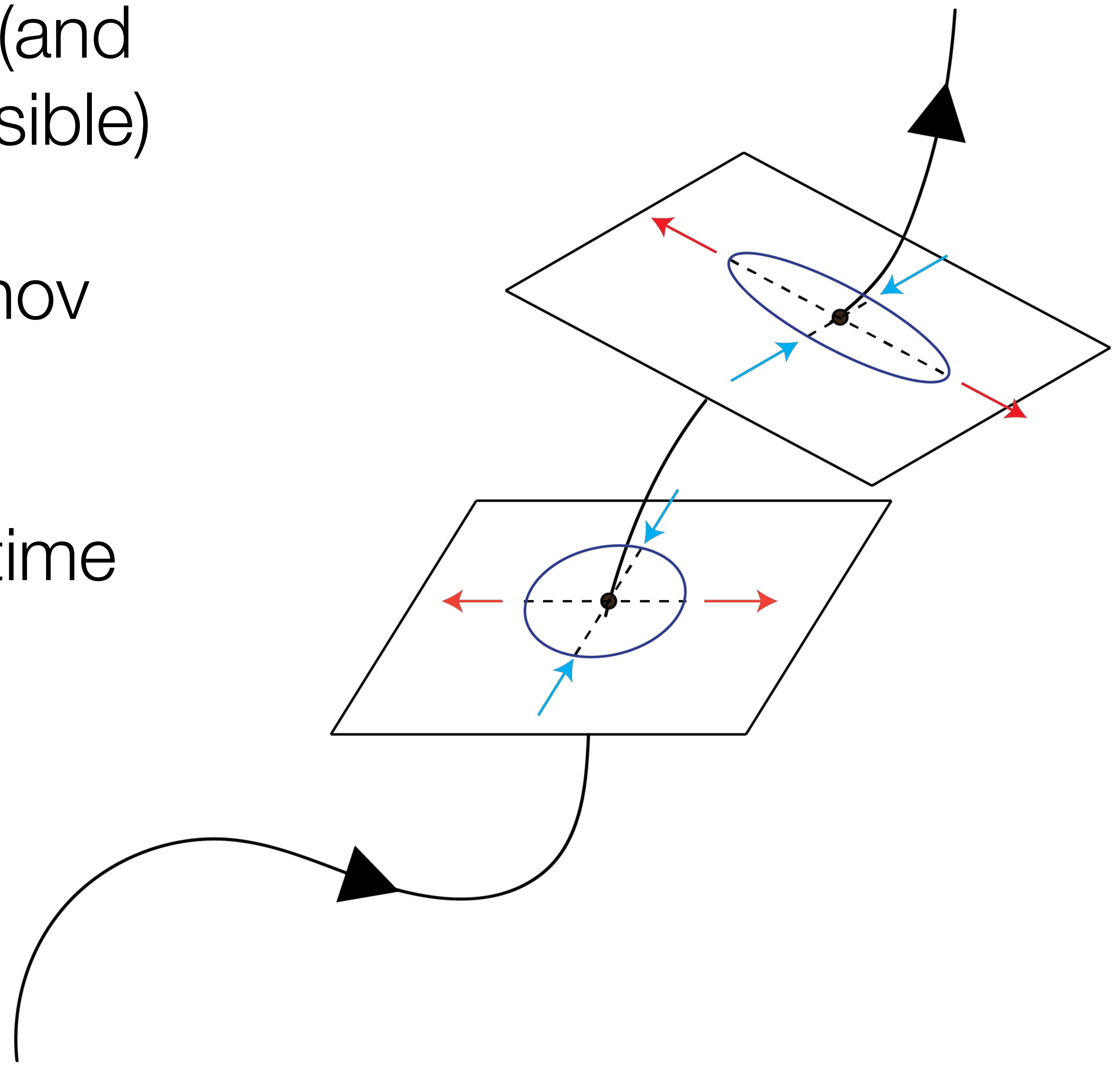
What is the ‘complexity’? of the system

What is the effective dimension of the chaotic attractor of the system (via the ‘Kaplan-Yorke conjecture’)?

What is the possible prediction horizon?

INTRODUCTION

In general, in the past it has proven difficult (and sometimes not possible) to estimate the spectrum of Lyapunov exponents of a dynamical system purely from limited time series data.



OUTLINE

Prediction

Given a limited time series of past measurements can we predict the future state of the dynamical system, at least in the short term?

Reconstructing the attractor

Can we learn something about the long-term dynamics of the system? Specifically, can we use our setup to find the Lyapunov exponents of a high-dimensional dynamical system?

Scalability

Can we use our setup for very high-dimensional attractors?

MACHINE LEARNING TECHNIQUE: RESERVOIR COMPUTING

Provides a way to train Recurrent Neural Networks.

Introduced by Jaeger (2001), and Maass et al. (2002).

Can be interpreted as a very high-dimensional system, called the reservoir (not to be confused with the measured dynamical system) which provides a rich repository of dynamics.

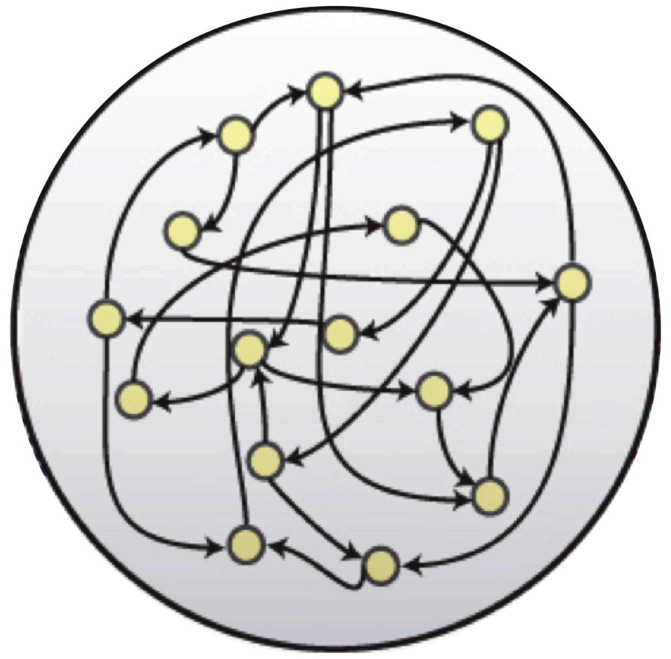
RESERVOIR COMPUTING

The reservoir in our setup is a network of D_r nodes.

Each node i has multiple inputs and outputs and a scalar state denoted by $r_i(t)$.

The weighted connections between the nodes can be represented by an adjacency matrix $\mathbf{A}$.

$\mathbf{r}(t)$



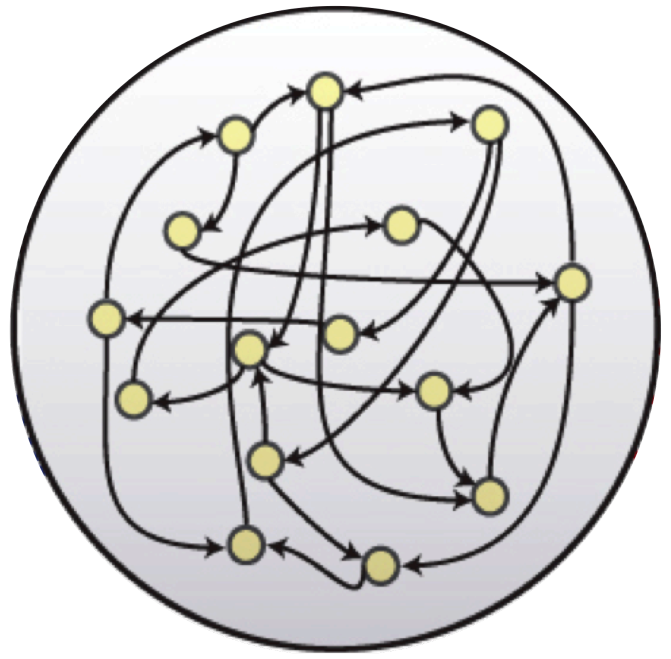
$\mathbf{A} : D_r \times D_r$

RESERVOIR COMPUTING

The reservoir in our setup is a network of D_r nodes.

Each node i has multiple inputs and outputs and a scalar state denoted by $r_i(t)$.

The weighted connections between the nodes can be represented by an adjacency matrix $\mathbf{A}$.



$\mathbf{A} : D_r \times D_r$

- Sparse.
- Randomly Generated.
- Fixed.

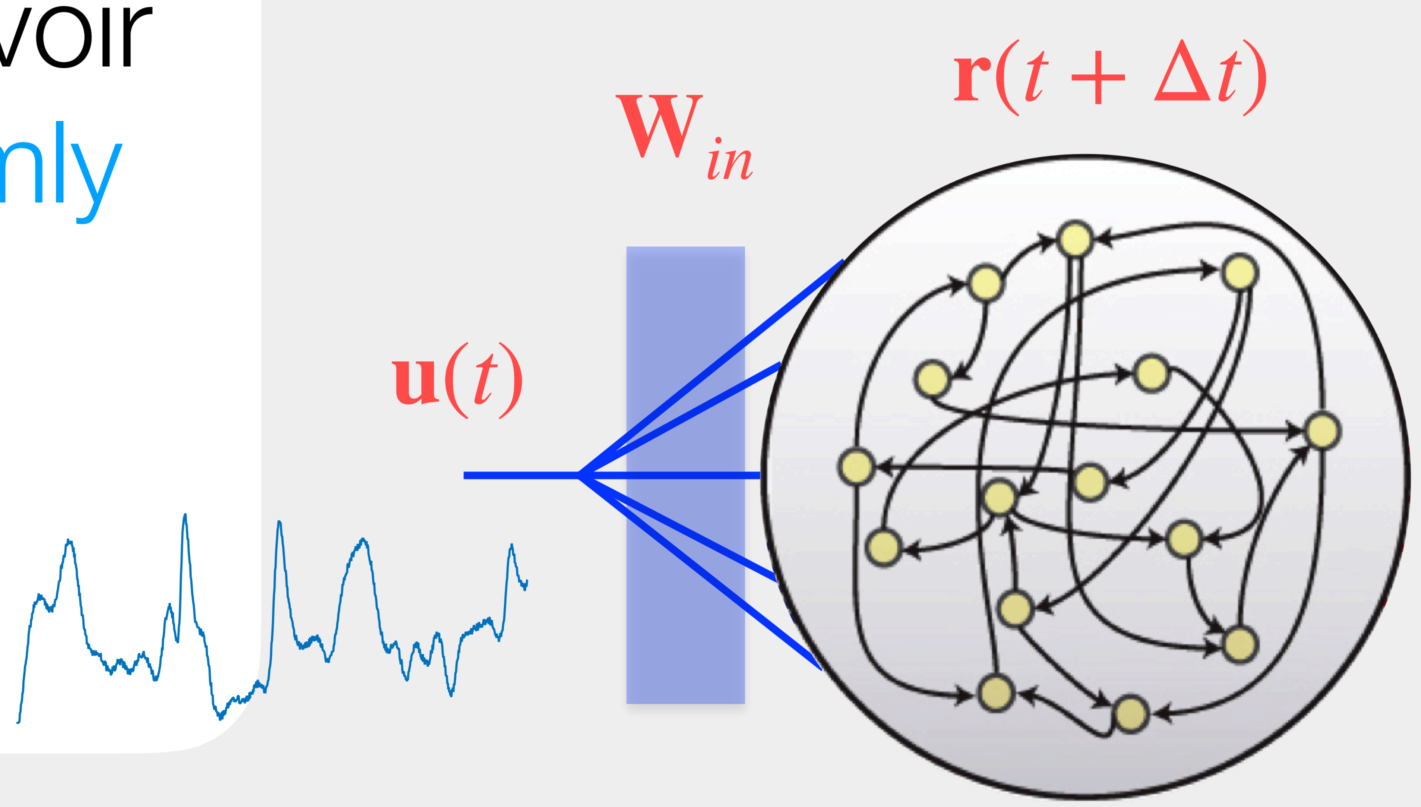
RESERVOIR NEURAL NETWORK IMPLEMENTATION

FEED DATA

An input is coupled to the reservoir network through a fixed, randomly generated input matrix.

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{u}(t)\right)$$

$$-T \leq t \leq 0$$



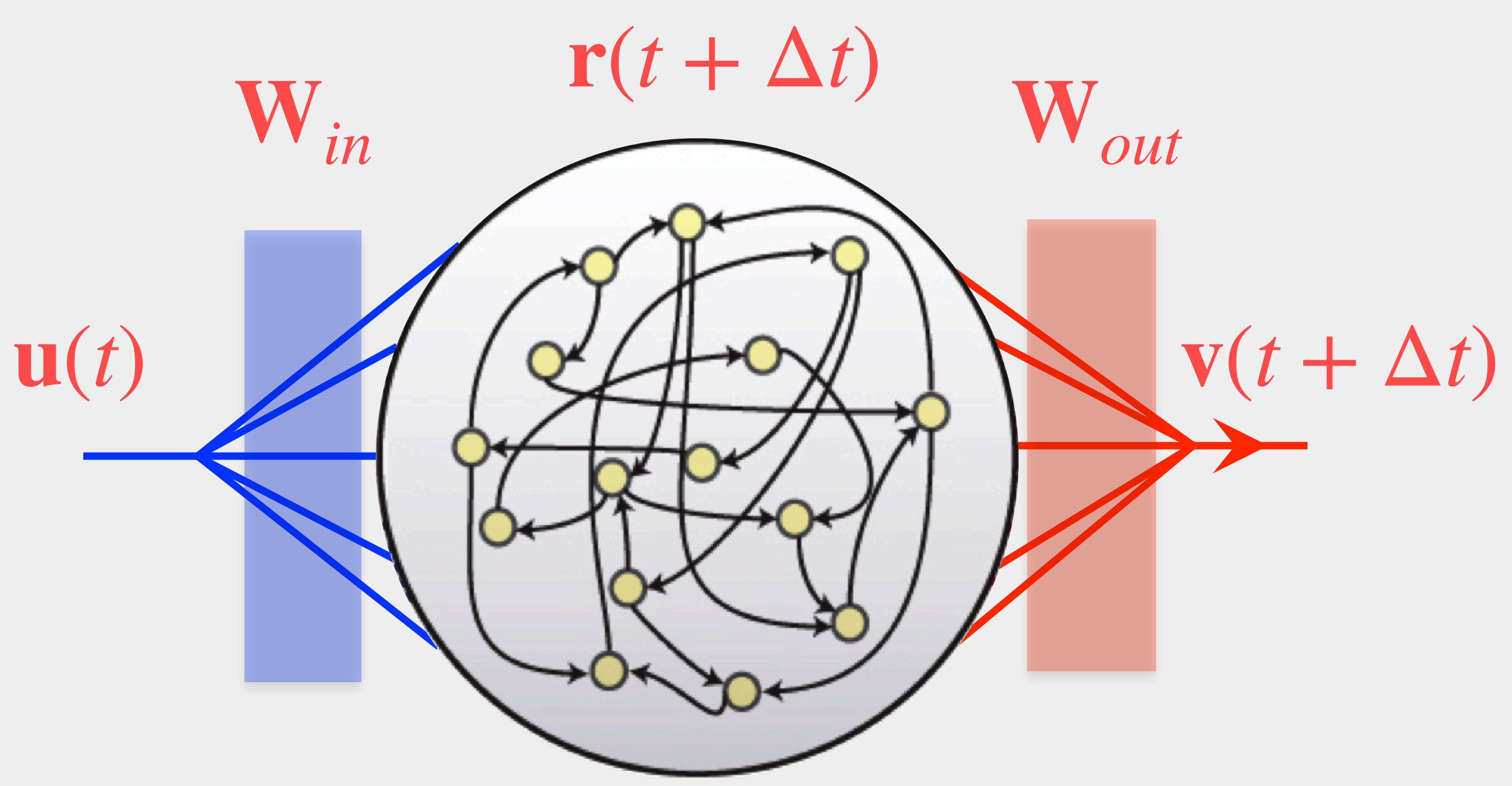
RESERVOIR NEURAL NETWORK IMPLEMENTATION

FEED DATA

An input is coupled to the reservoir network through a fixed, randomly generated input matrix.

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{u}(t)\right)$$

$$-T \leq t \leq 0$$



LINEAR FIT

Find the output weight matrix that minimizes the following loss function

$$\mathscr{L}(\mathbf{W}_{out}) = \sum_{t=-T}^{0} \|\mathbf{W}_{out}\mathbf{r}(t) - \mathbf{u}(t)\|^2$$

$$\mathbf{v}(t) = \mathbf{W}_{out}\mathbf{r}(t)$$
$$\mathbf{v}(t) \simeq \mathbf{u}(t)$$

RESERVOIR NEURAL NETW
IMPLEMENTATION

FEED DATA

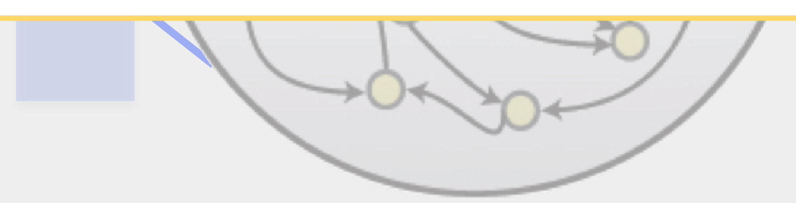
An input is coupled to the reservoir network through a fixed, randomly generated input matrix.

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{u}(t)\right)$$

$$-T \leq t \leq 0$$

TRAINING IS A SIMPLE LINEAR REGRESSION PROBLEM.

INSTEAD OF GRADIENT DESCENT, WE HAVE A SIMPLER PROBLEM OF MATRIX INVERSION.



LINEAR FIT

Find the output weight matrix that minimizes the following loss function

$$\mathscr{L}(\mathbf{W}_{out}) = \sum_{t=-T}^{0} \|\mathbf{W}_{out}\mathbf{r}(t) - \mathbf{u}(t)\|^2$$

$$\mathbf{v}(t) = \mathbf{W}_{out}\mathbf{r}(t)$$

$$\mathbf{v}(t) \simeq \mathbf{u}(t)$$

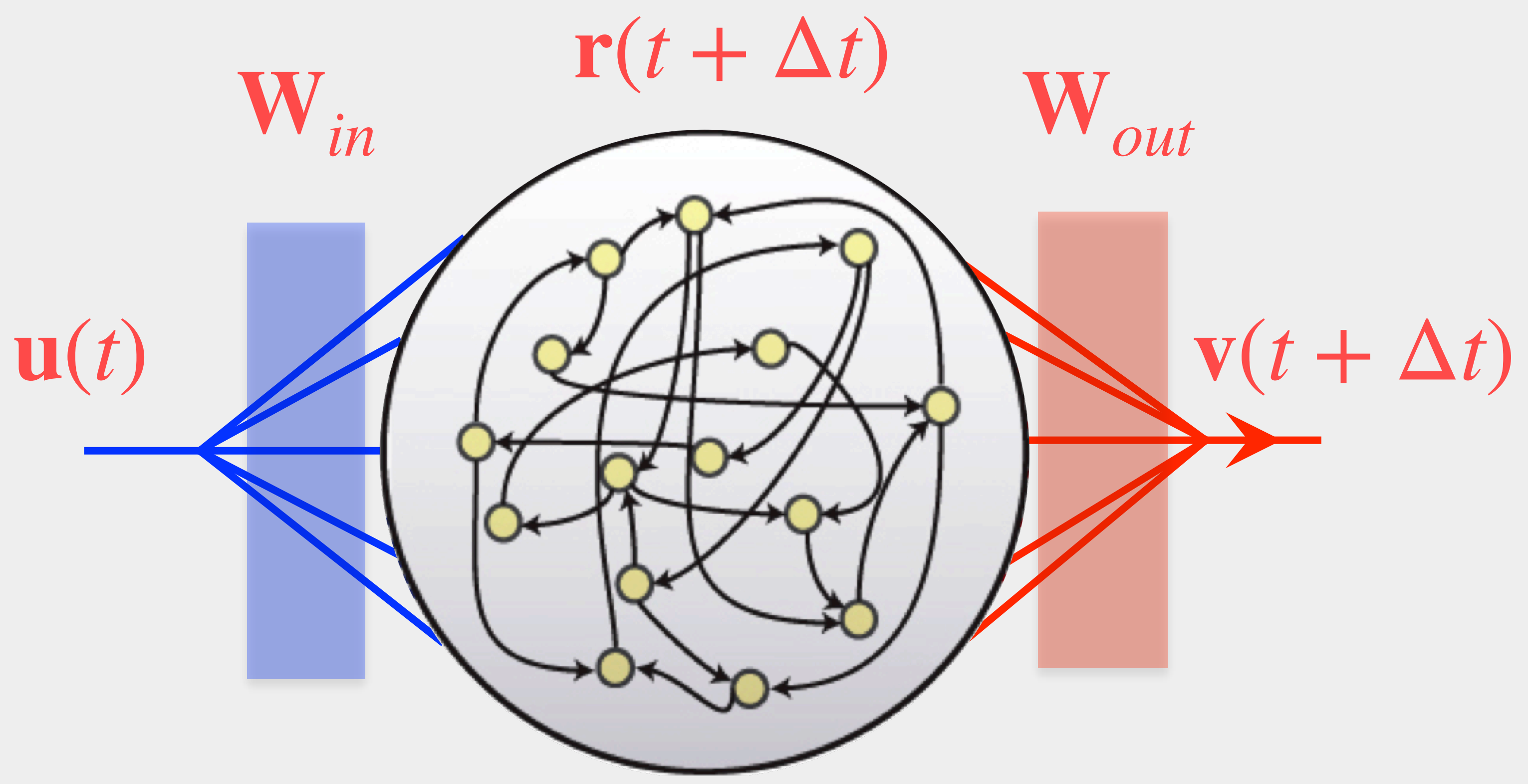
RESERVOIR NEURAL NETWORK IMPLEMENTATION

FEED DATA

An input is coupled to the reservoir network through a fixed, randomly generated input matrix.

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{u}(t)\right)$$

$$-T \leq t \leq 0$$



LINEAR FIT

Find the output weight matrix that minimizes the following loss function

$$\mathscr{L}(\mathbf{W}_{out}) = \sum_{t=-T}^{0} \|\mathbf{W}_{out}\mathbf{r}(t) - \mathbf{u}(t)\|^2$$

$$\mathbf{v}(t) = \mathbf{W}_{out}\mathbf{r}(t)$$
$$\mathbf{v}(t) \simeq \mathbf{u}(t)$$

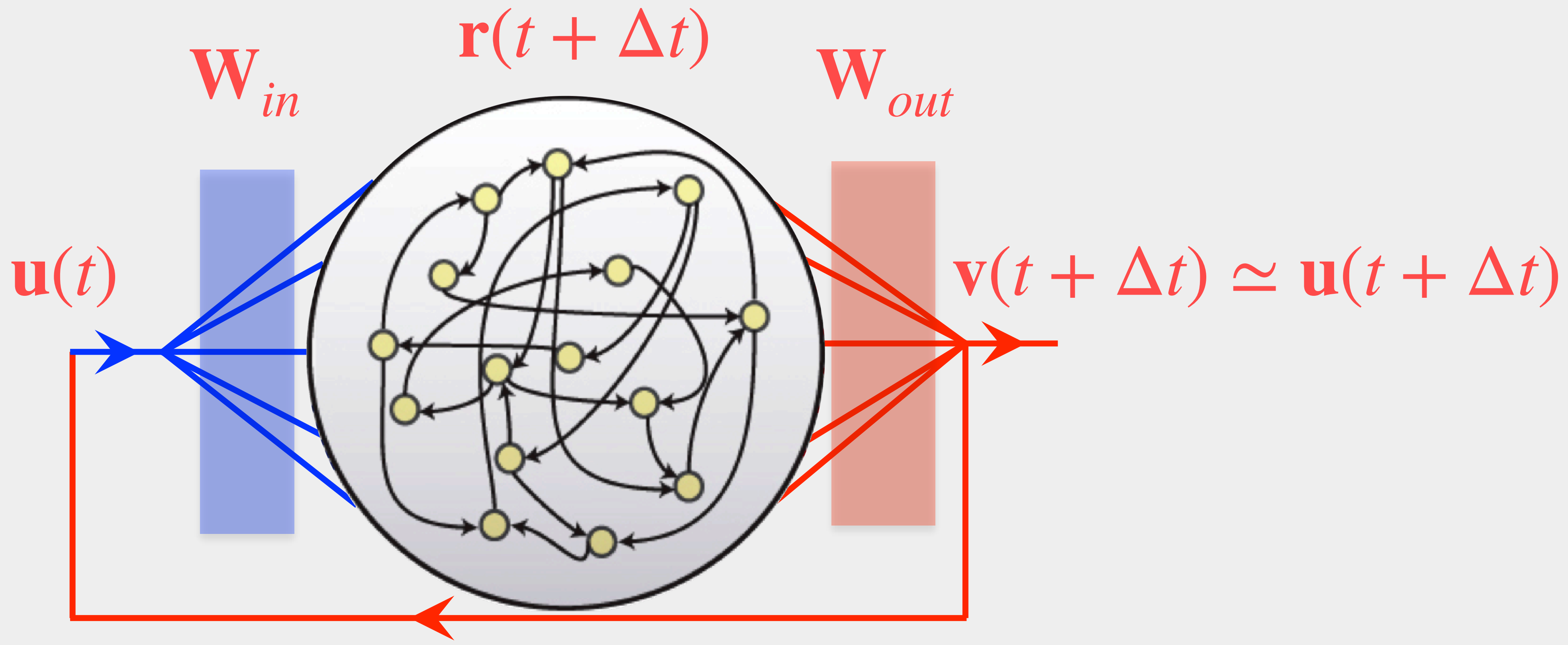
RESERVOIR NEURAL NETWORK IMPLEMENTATION

FEED DATA

An input is coupled to the reservoir network through a fixed, randomly generated input matrix.

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{u}(t)\right)$$

$$-T \leq t \leq 0$$



LINEAR FIT

Find the output weight matrix that minimizes the following loss function

$$\mathscr{L}(\mathbf{W}_{out}) = \sum_{t=-T}^{0} \|\mathbf{W}_{out}\mathbf{r}(t) - \mathbf{u}(t)\|^2$$

$$\mathbf{v}(t) = \mathbf{W}_{out}\mathbf{r}(t)$$

$$\mathbf{v}(t) \simeq \mathbf{u}(t)$$

PREDICTION

$$t > 0$$

$$\mathbf{v}(t) = \mathbf{W}_{out}\mathbf{r}(t)$$

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{v}(t)\right)$$

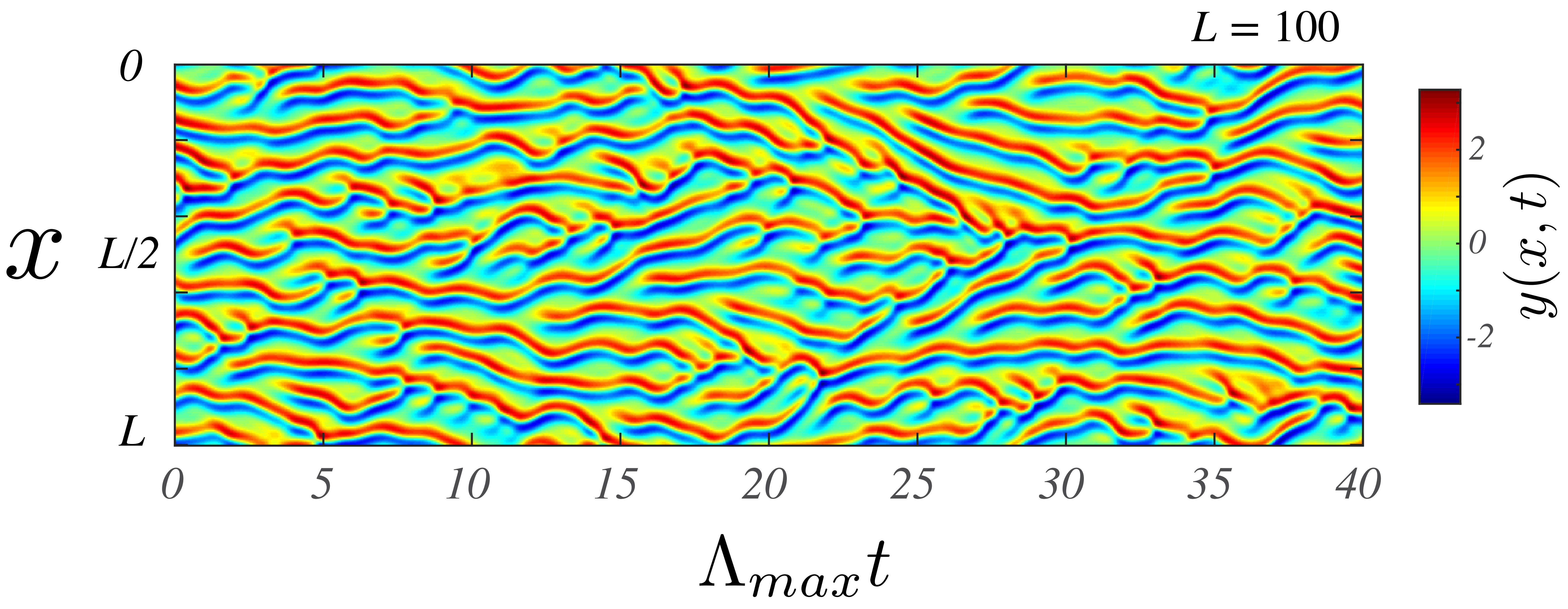
THE KURAMOTO-SIVASHINSKY (KS) SYSTEM

A nonlinear, spatiotemporal chaotic PDE

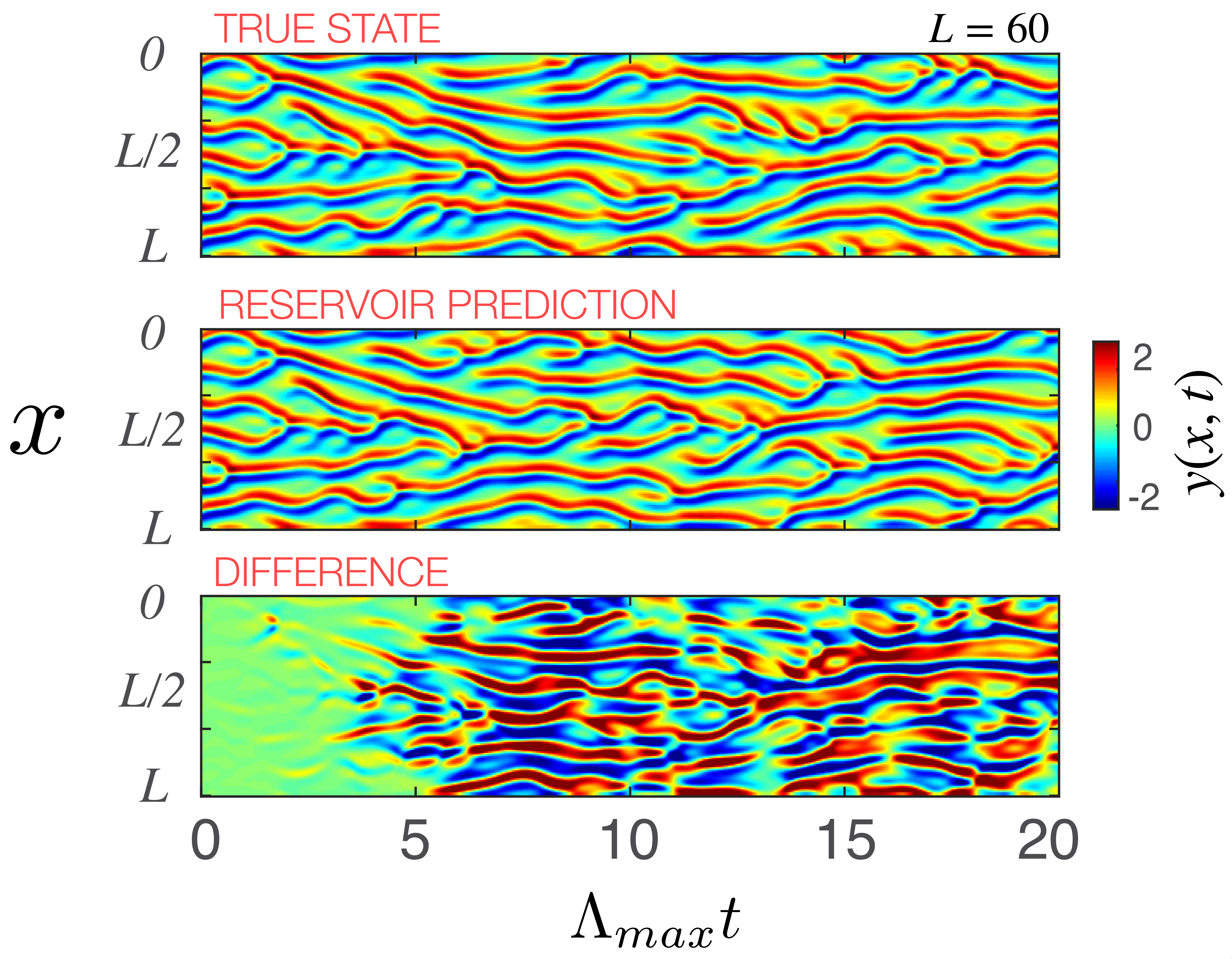
$y_t = -yy_x - y_{xx} - y_{xxxx}$

$x \in [0,L)$

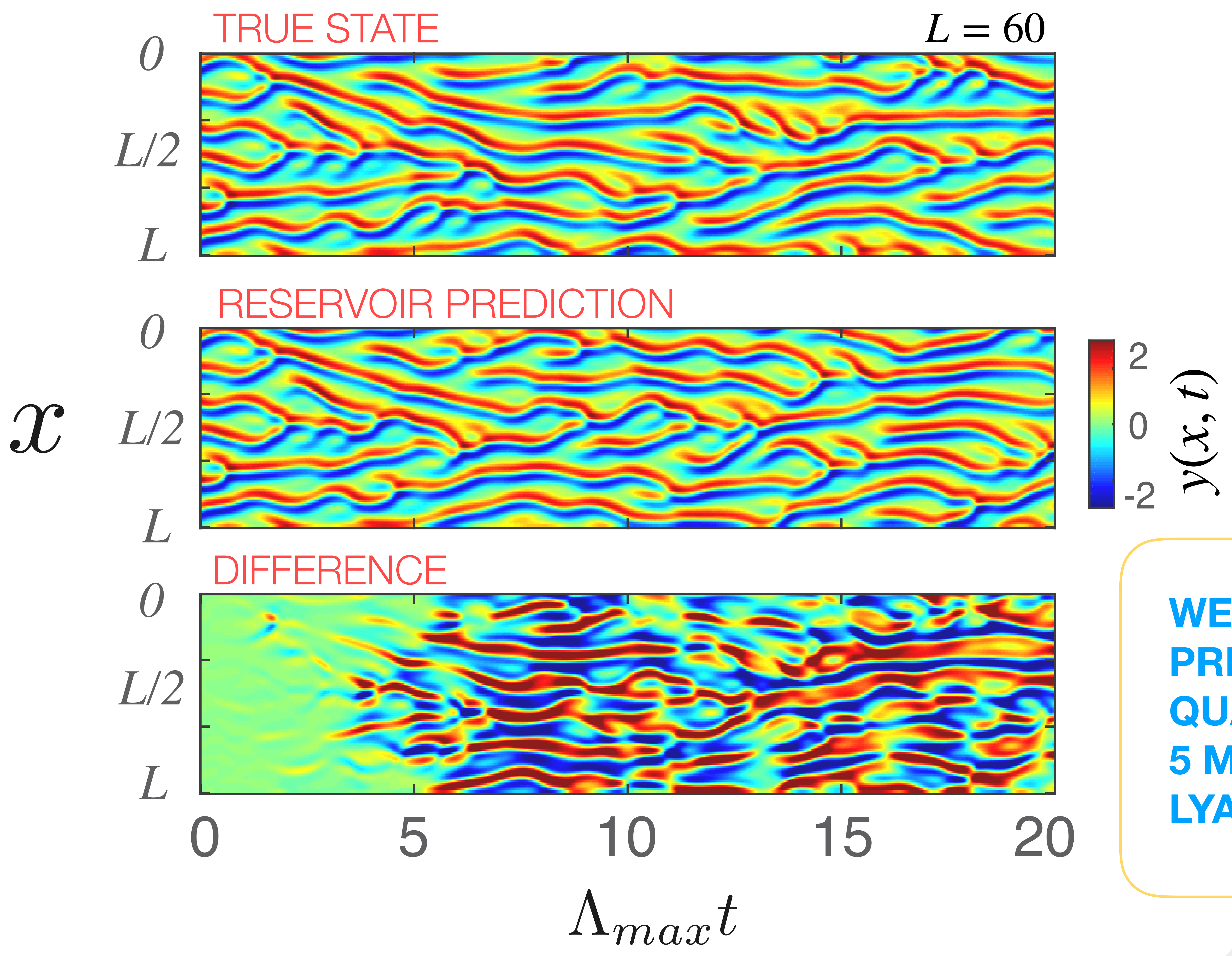
$y(x+L) = y(x)$



SHORT-TERM FORECASTING OF CHAOS: KURAMOTO-SIVASHINSKY EQUATION



SHORT-TERM FORECASTING OF CHAOS: KURAMOTO-SIVASHINSKY EQUATION



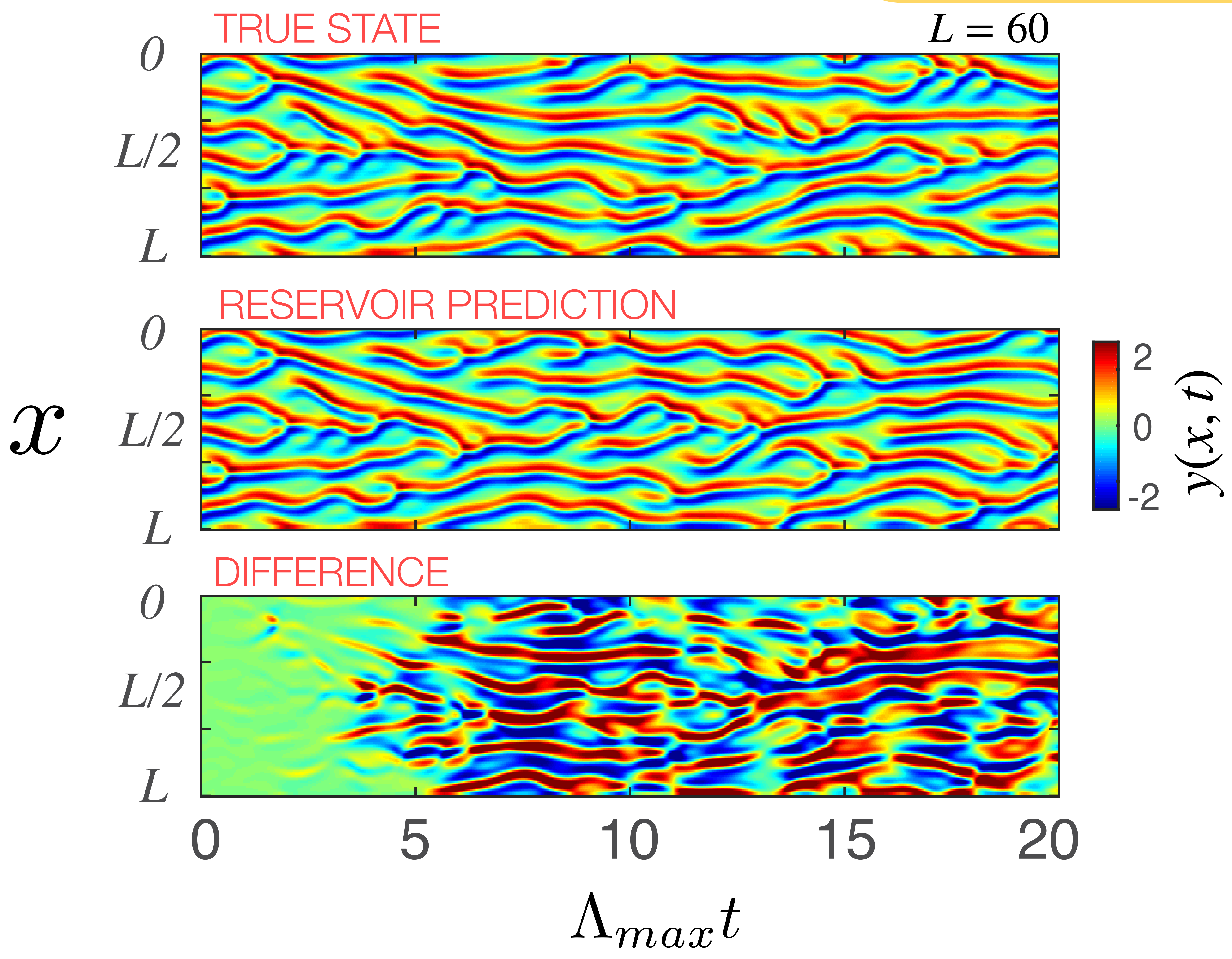
WE OBTAIN GOOD PREDICTION QUALITY FOR ABOUT 5 MULTIPLES OF THE LYAPUNOV TIME

SHORT-TERM FORECASTING OF CHAOS: KURAMOTO-SIVASHINSKY EQUATION

The reservoir computer is really good at learning the dynamics from data alone.

It is capable of making high quality short term predictions when the system dynamics is unknown.

NOTICE THAT THE DYNAMICS LOOKS ‘KS LIKE’ EVEN WHEN THE PREDICTION HAS DIVERGED FROM THE TRUE STATE



‘CLIMATE’ OF THE RESERVOIR DYNAMICS

Has the reservoir truly learned the dynamical behavior of the Kuramoto-Sivashinsky system?

If it has, the ergodic properties of the autonomous reservoir dynamical system should resemble those of the true system.

Is there a way to verify this?

OUTLINE

Prediction
Given a limited time series of past measurements can we predict the future state of the dynamical system, at least in the short term?

Reconstructing the attractor
Can we learn something about the long-term dynamics of the system? Specifically, can we use our setup to find the Lyapunov exponents of a high-dimensional dynamical system?

Scalability
Can we use our setup for very high-dimensional attractors?

LYAPUNOV EXPONENTS FROM DATA

We know the evolution equation of the post-training autonomous reservoir system.

$$\mathbf{r}(t + \Delta t) = \mathbf{F}\left[\mathbf{r}(t)\right]$$

LYAPUNOV EXPONENTS FROM DATA

We know the evolution equation of the post-training autonomous reservoir system.

$$\mathbf{r}(t+\Delta t) = \mathbf{F}\left[\mathbf{r}(t)\right]$$

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{W}_{out}\mathbf{r}(t)\right)$$

LYAPUNOV EXPONENTS FROM DATA

We know the evolution equation of the post-training autonomous reservoir system.

$$\mathbf{r}(t+\Delta t) = \mathbf{F}\left[\mathbf{r}(t)\right]$$

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{W}_{out}\mathbf{r}(t)\right)$$

We can calculate the equation of the tangent map evolution.

$$\delta\mathbf{r}(t+\Delta t) = D\mathbf{F}\left[\mathbf{r}(t)\right]\delta\mathbf{r}(t)$$

LYAPUNOV EXPONENTS FROM DATA

We know the evolution equation of the post-training autonomous reservoir system.

$$\mathbf{r}(t+\Delta t) = \mathbf{F}\left[\mathbf{r}(t)\right]$$

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{W}_{out}\mathbf{r}(t)\right)$$

We can calculate the equation of the tangent map evolution.

$$\delta\mathbf{r}(t+\Delta t) = D\mathbf{F}\left[\mathbf{r}(t)\right]\delta\mathbf{r}(t)$$

Use these to compute the Lyapunov exponents of the reservoir dynamical system.

LYAPUNOV EXPONENTS FROM DATA

We know the evolution equation of the post-training autonomous reservoir system.

$$\mathbf{r}(t+\Delta t) = \mathbf{F}\left[\mathbf{r}(t)\right]$$

$$\mathbf{r}(t+\Delta t) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{W}_{out}\mathbf{r}(t)\right)$$

We can calculate the equation of the tangent map evolution.

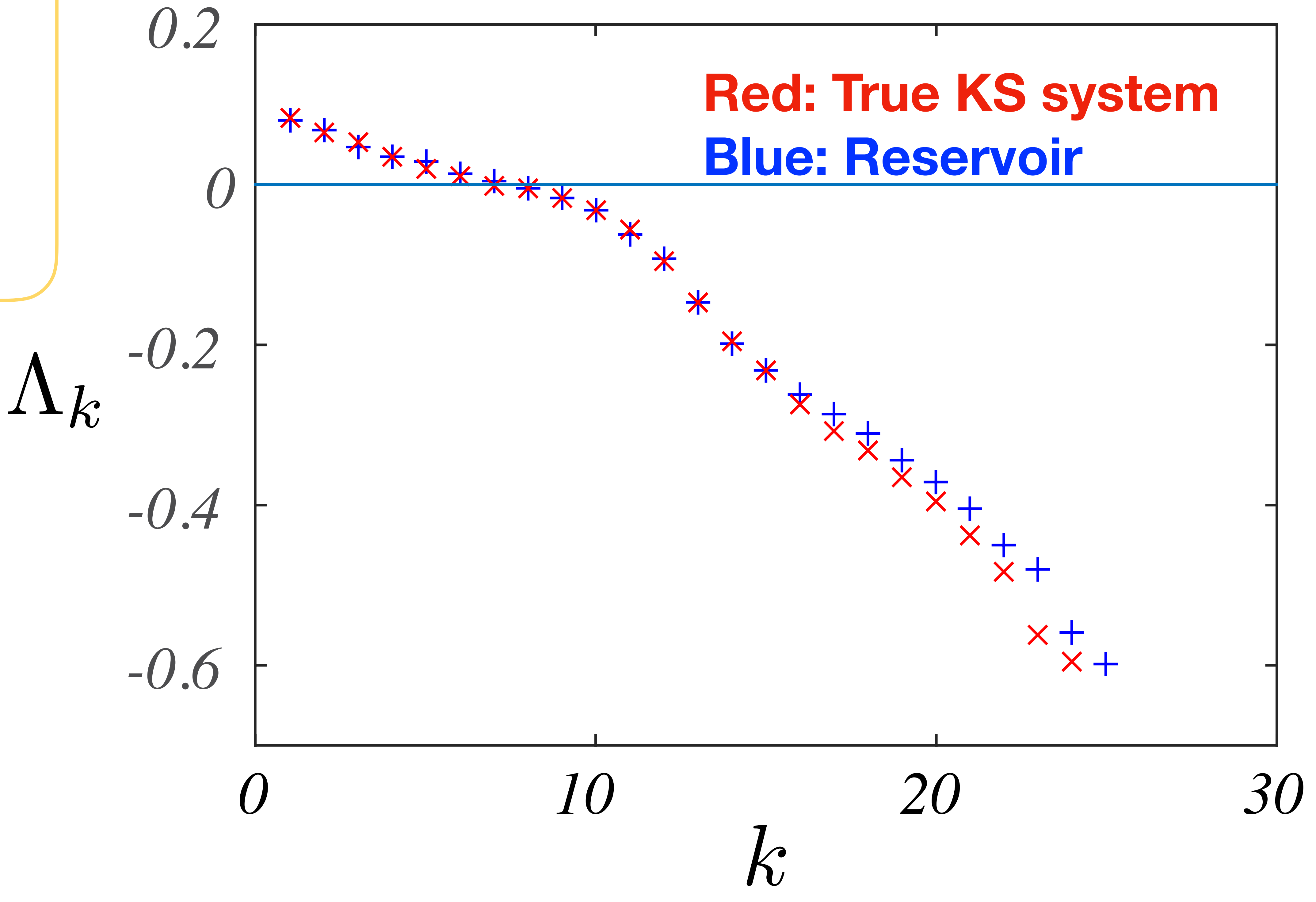
$$\delta\mathbf{r}(t+\Delta t) = D\mathbf{F}\left[\mathbf{r}(t)\right]\delta\mathbf{r}(t)$$

Use these to compute the Lyapunov exponents of the reservoir dynamical system.

Are the Lyapunov exponents of the reservoir same as those of the data generating system?

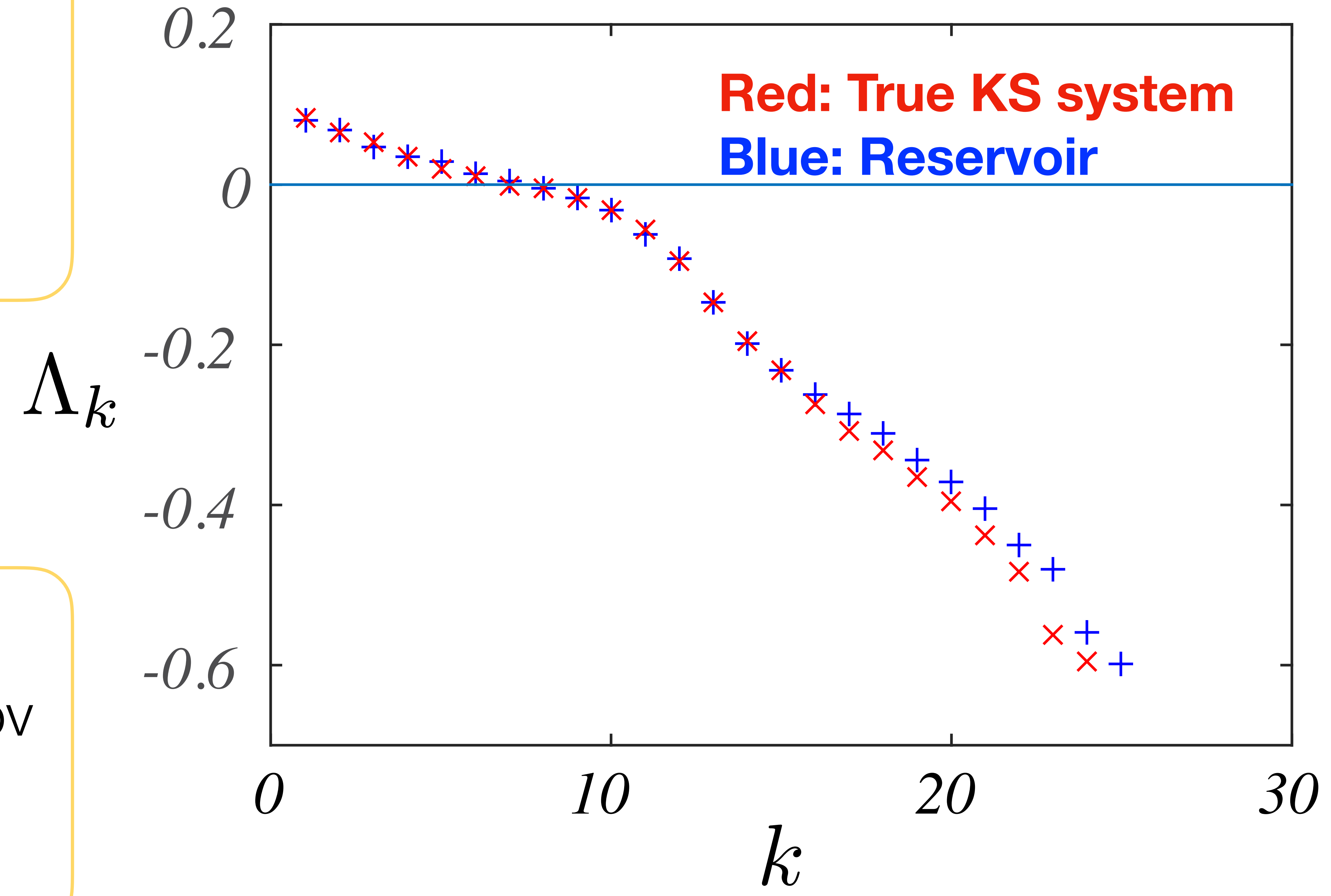
LYAPUNOV EXPONENTS FROM DATA

The Lyapunov exponents of the KS system are indeed accurately reproduced by the reservoir!



LYAPUNOV EXPONENTS FROM DATA

The Lyapunov exponents of the KS system are indeed accurately reproduced by the reservoir!



The reservoir fails at obtaining symmetry-related zero Lyapunov exponents. This aspect is discussed further in our paper.

OUTLINE

Prediction
Given a limited time series of past measurements can we predict the future state of the dynamical system, at least in the short term?

Reconstructing the attractor
Can we learn something about the long-term dynamics of the system? Specifically, can we use our setup to find the Lyapunov exponents of a high-dimensional dynamical system?

Scalability
Can we use our setup for very high-dimensional attractors?

VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS

Can machine learning be useful for studying very high dimensional dynamical systems? Interesting dynamical systems such as atmospheric general circulation models are very high-dimensional.

VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS

Can machine learning be useful for studying very high dimensional dynamical systems? Interesting dynamical systems such as atmospheric general circulation models are very high-dimensional.

Why is this a difficult problem?

VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS

Can machine learning be useful for studying very high dimensional dynamical systems? Interesting dynamical systems such as atmospheric general circulation models are very high-dimensional.

Why is this a difficult problem?

As you increase the size/complexity of the dynamical system, you need a bigger neural network.

But if we keep increasing the nodes, the training problem becomes intractable.

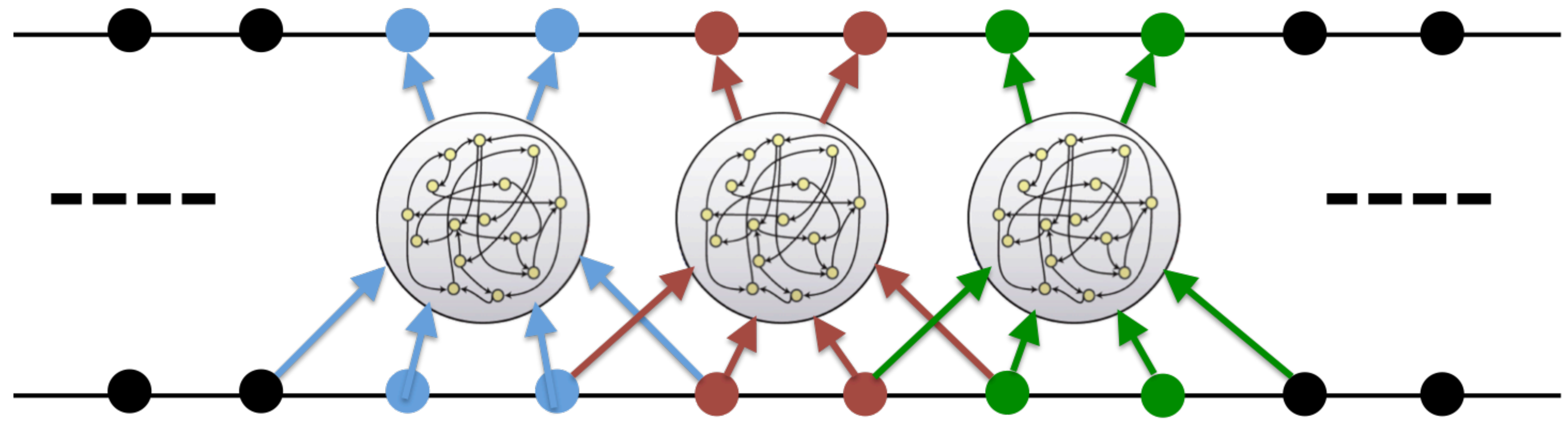
VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS

We use some knowledge of the physics of spatiotemporal systems: our system has short-range spatial correlations.

The training and prediction of spatiotemporal chaos can be very effectively parallelized

VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS

Training



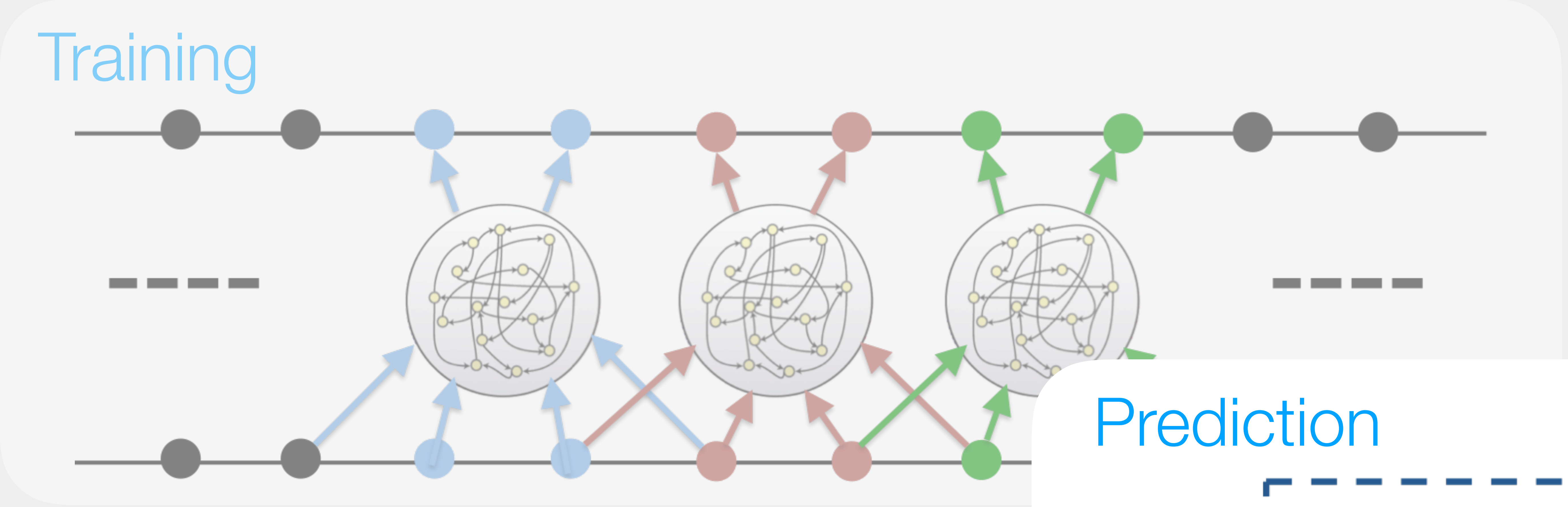
Multiple reservoirs, parallel computing on a cluster.

Each reservoir is assigned to a local neighborhood on the spatial grid and is trained to predict a subset of its inputs.

The local neighborhood consists of the grid points to be predicted plus buffer zones on each side.

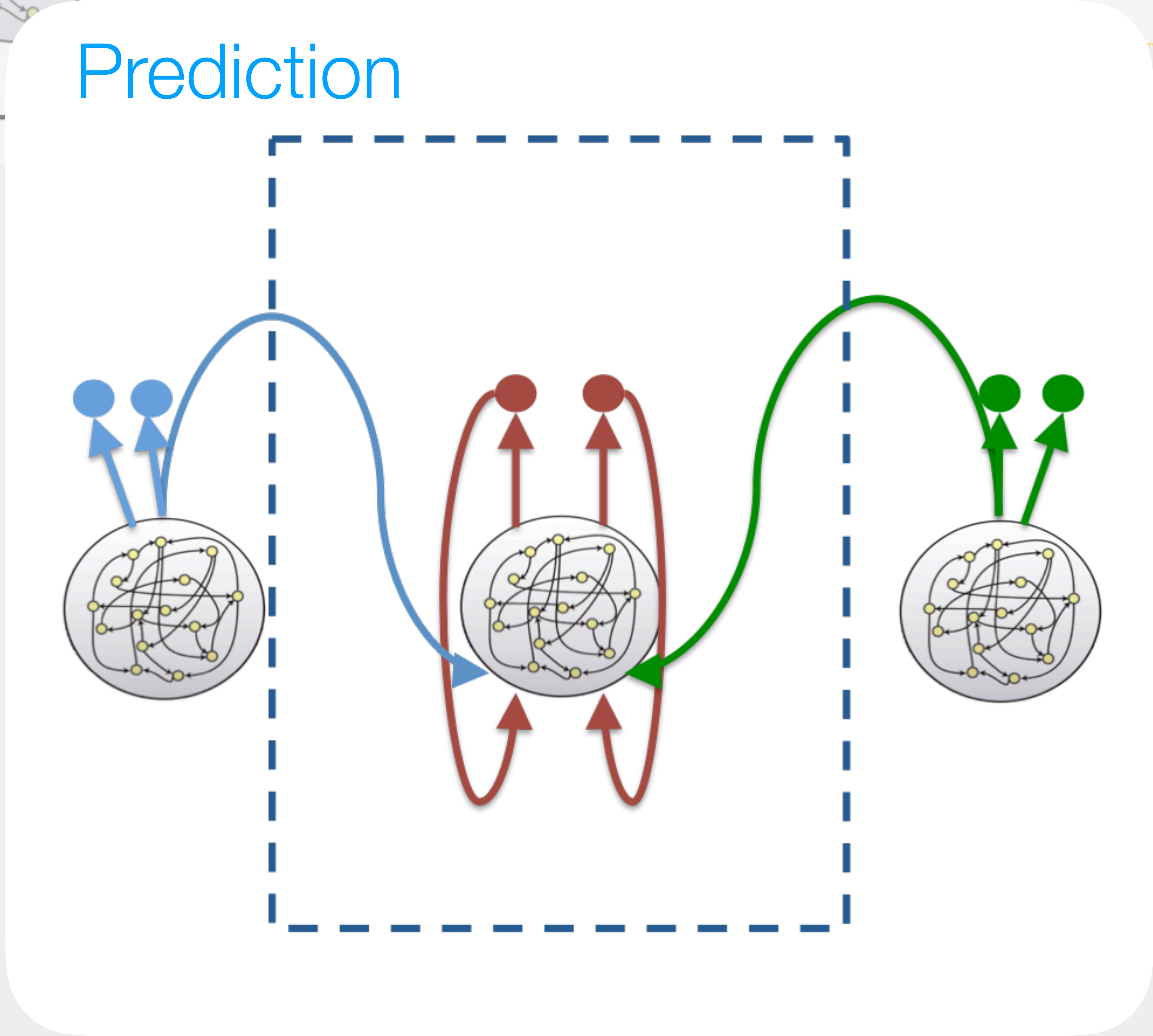
Each of the relatively small parallel reservoirs has its own relatively small independently trained output matrix.

VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS



Multiple reservoirs, parallel computing on a cluster.

During the prediction phase, a given reservoir has feedbacks from its own outputs and from the outputs of neighboring reservoirs on each side to account for the buffer zones/ overlaps.



VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS

We use a KS type system with periodic boundary conditions $y(x + L, t) = y(x, t)$.

The attractor dimension of this system increases linearly with the periodicity length L.

L	D_{KY}
100	23
200	43
400	85
800	167
1600	338

D_{KY} : Kaplan-Yorke Dimension of the Chaotic Attractor (Measure of Complexity)

VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS: KURAMOTO-SIVASHINSKY

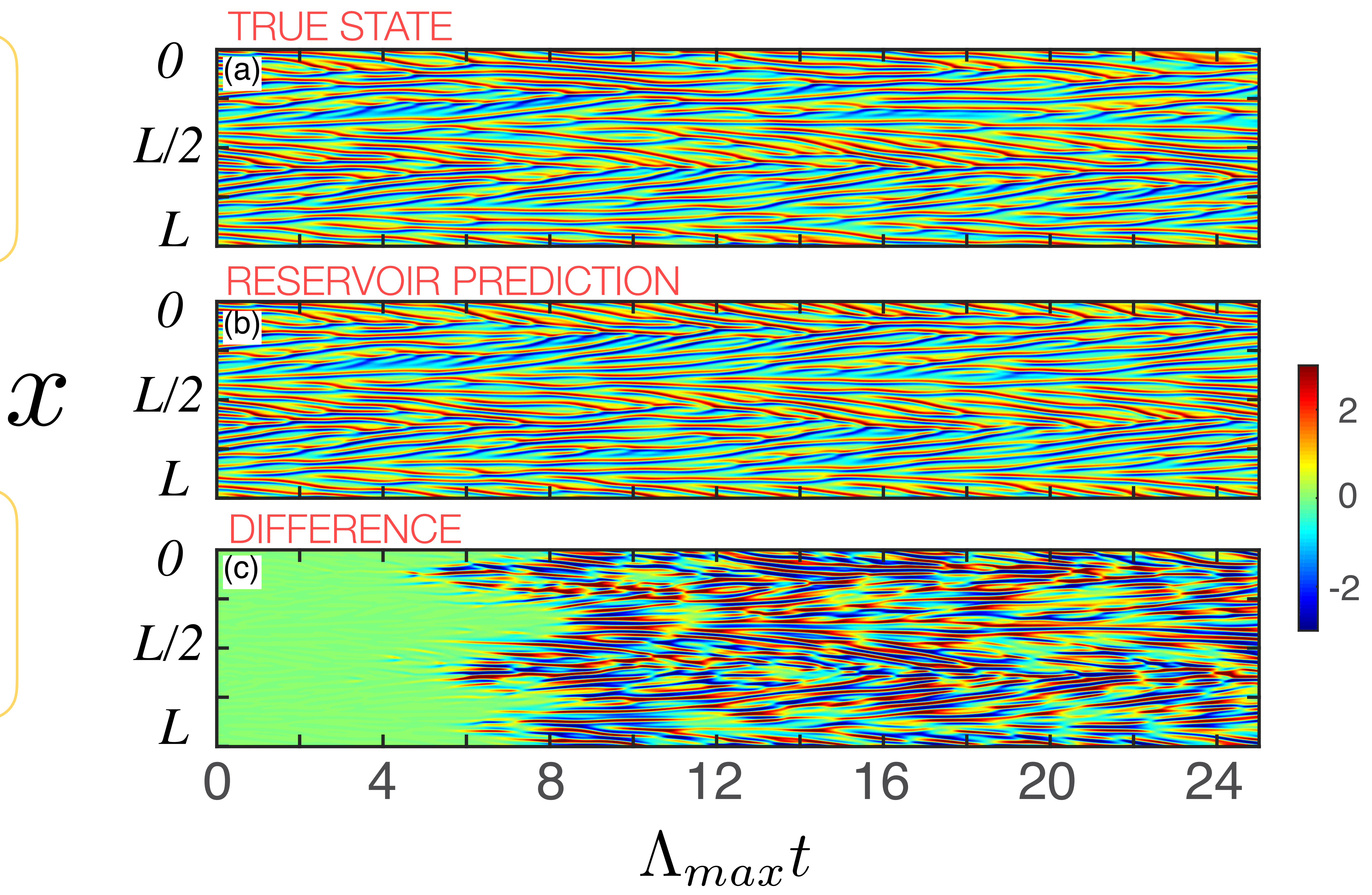
Model Parameters:

$L = 200$

$D_{KY} = 43$

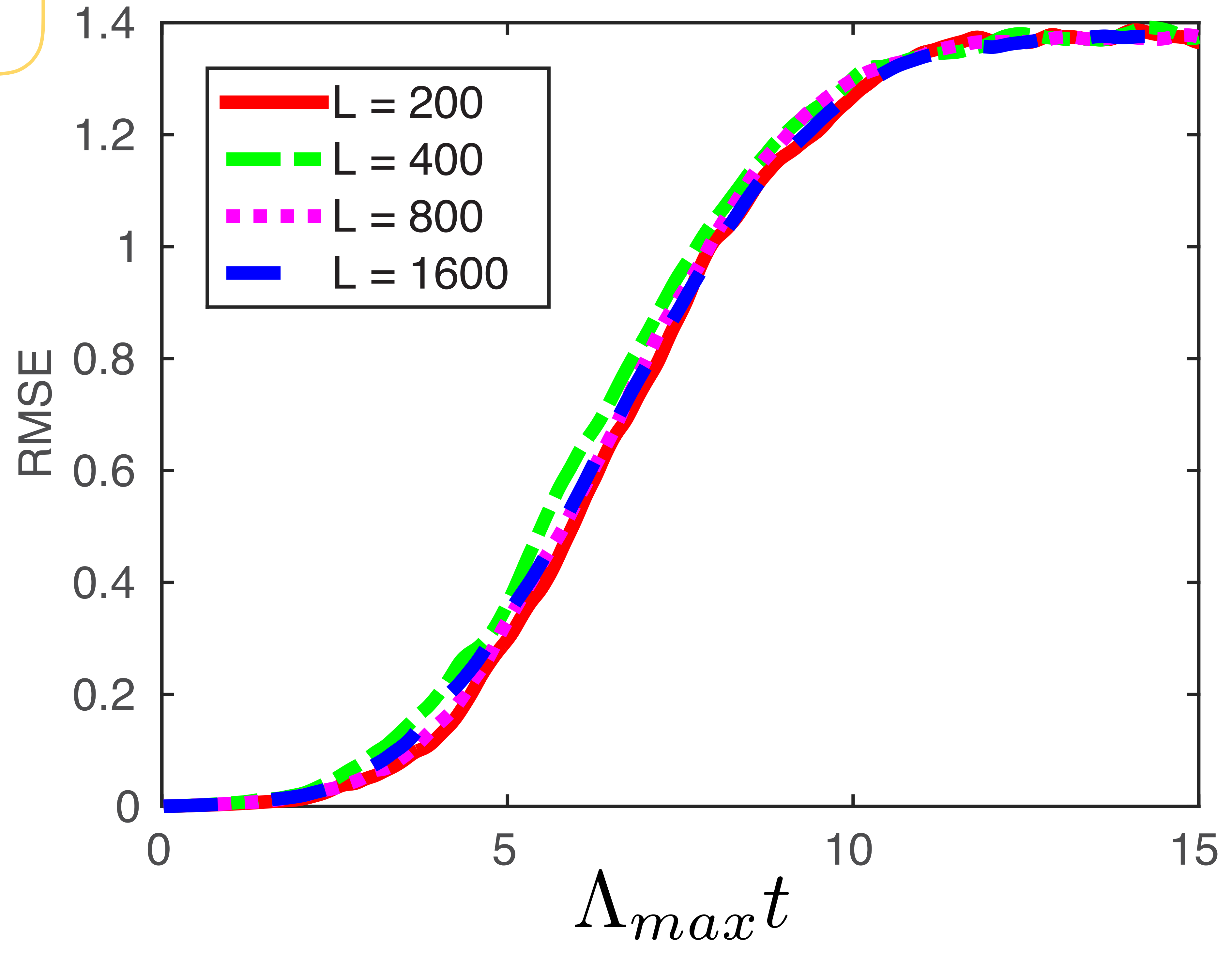
Reservoir:

$(D_r = 5000) \times 64$



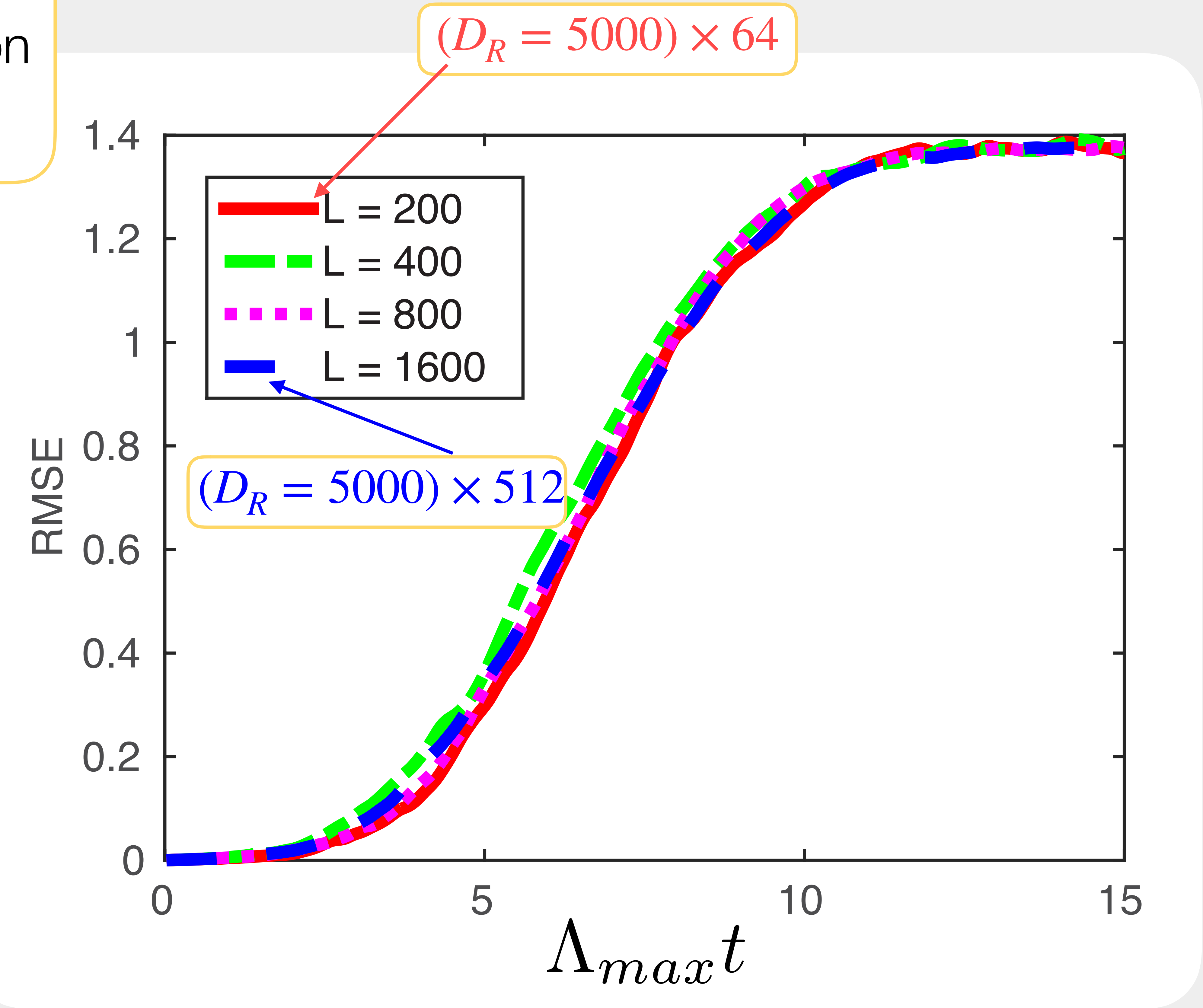
VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS: KURAMOTO-SIVASHINSKY

In this plot we show the RMS error as a function of time in the prediction phase (averaged over many trials).



VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS: KURAMOTO-SIVASHINSKY

In this plot we show the RMS error as a function of time in the prediction phase (averaged over many trials).

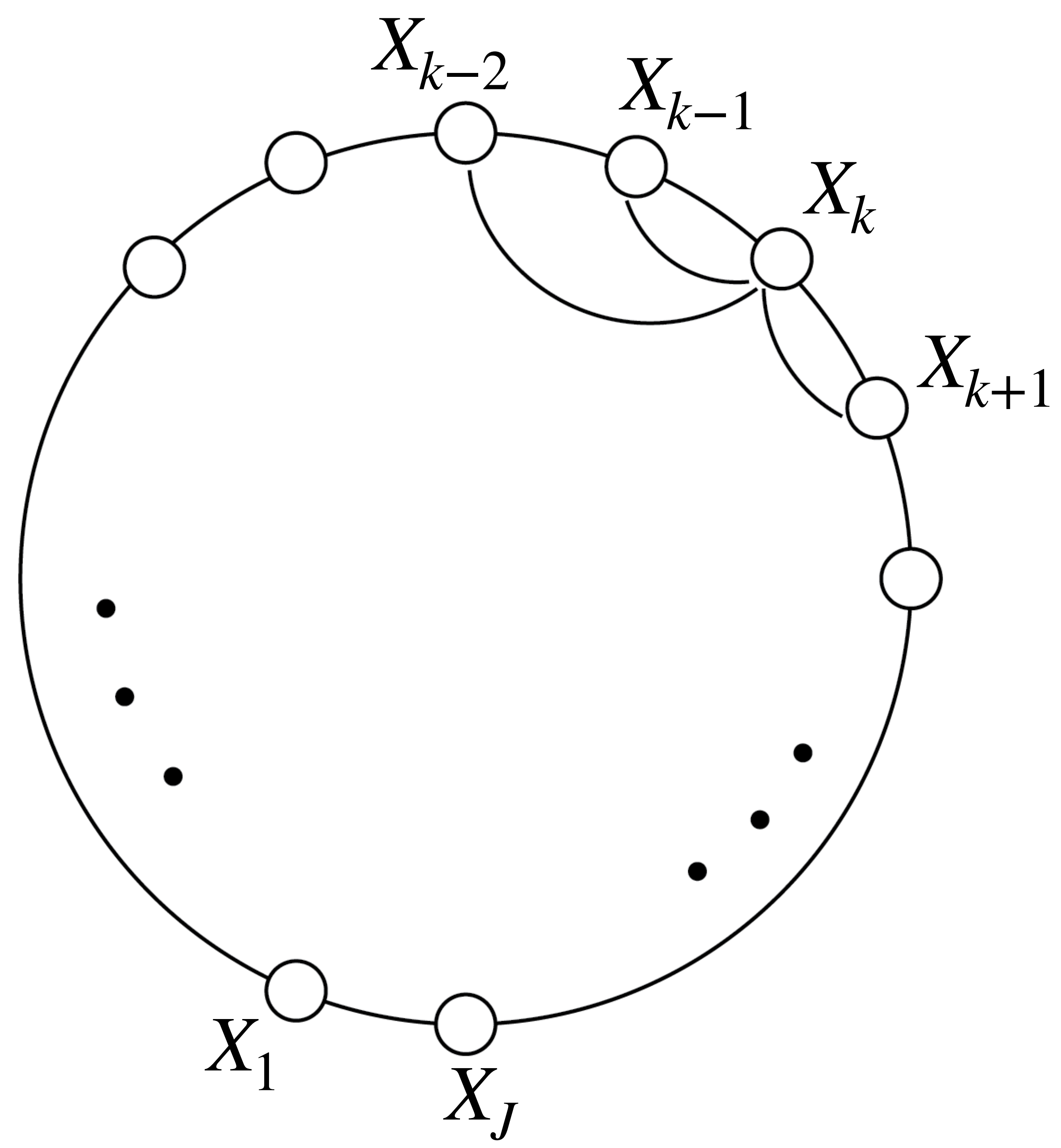


VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS: THE LORENZ 1996 TOY WEATHER MODEL

As another example, we consider the Lorenz 1996 model of atmospheric dynamics often used for testing techniques in weather prediction.

$$\frac{dX_k}{dt} = -X_k + \left(X_{k+1} - X_{k-2}\right) X_{k-1} + F$$

$k = 1,2,\ldots J$



VERY HIGH-DIMENSIONAL DYNAMICAL SYSTEMS: THE LORENZ 1996 TOY WEATHER MODEL

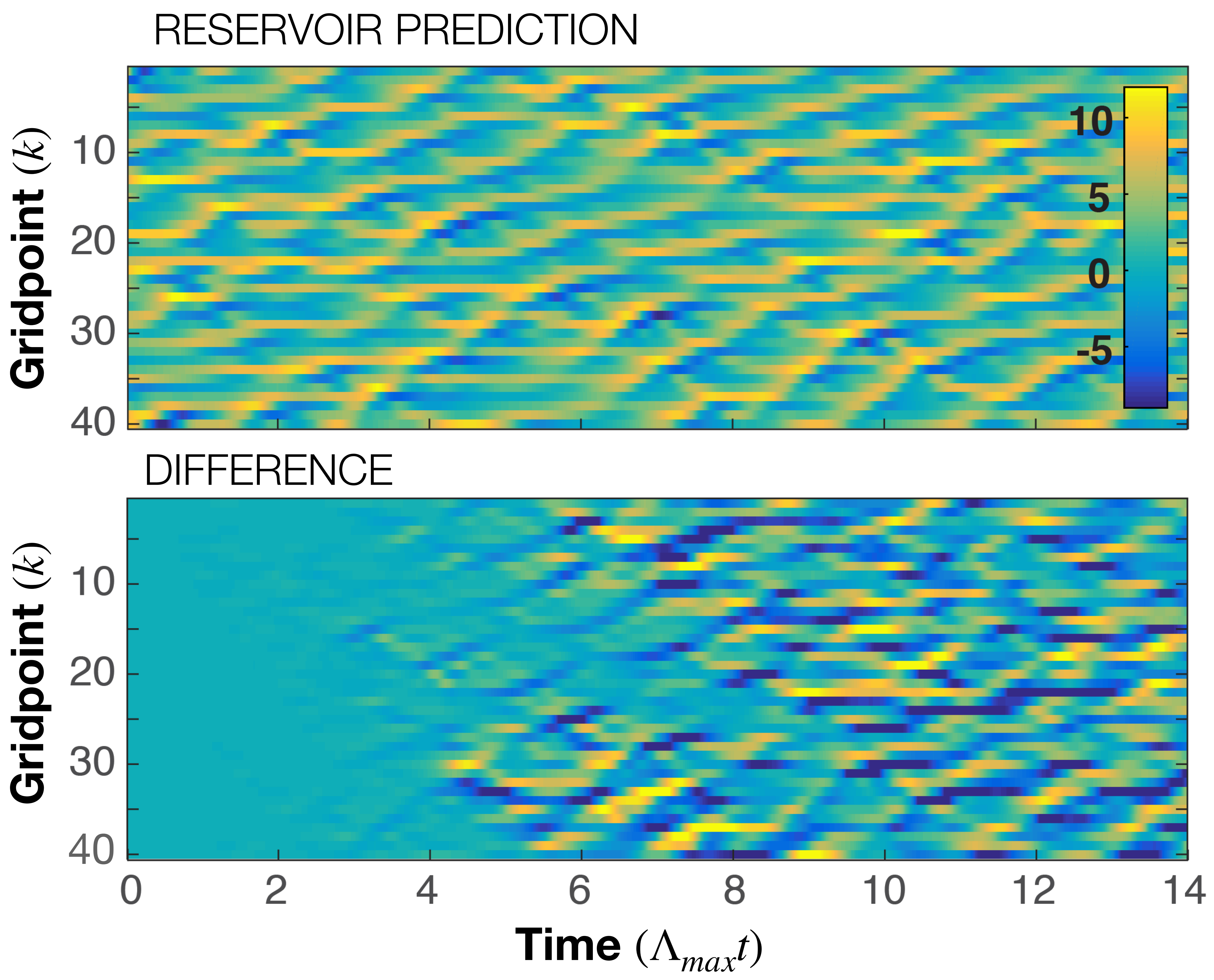
Model Parameters:

$J = 40$

$F = 8$

Reservoir:

$(D_r = 5000) \times 20$



OUTLINE

Prediction

Given a limited time series of past measurements can we predict the future state of the dynamical system, at least in the short term?

Reconstructing the attractor

Can we learn something about the long-term dynamics of the system? Specifically, can we use our setup to find the Lyapunov exponents of a high dimensional dynamical system?

Scalability

Can we use our setup for very high dimensional attractors?

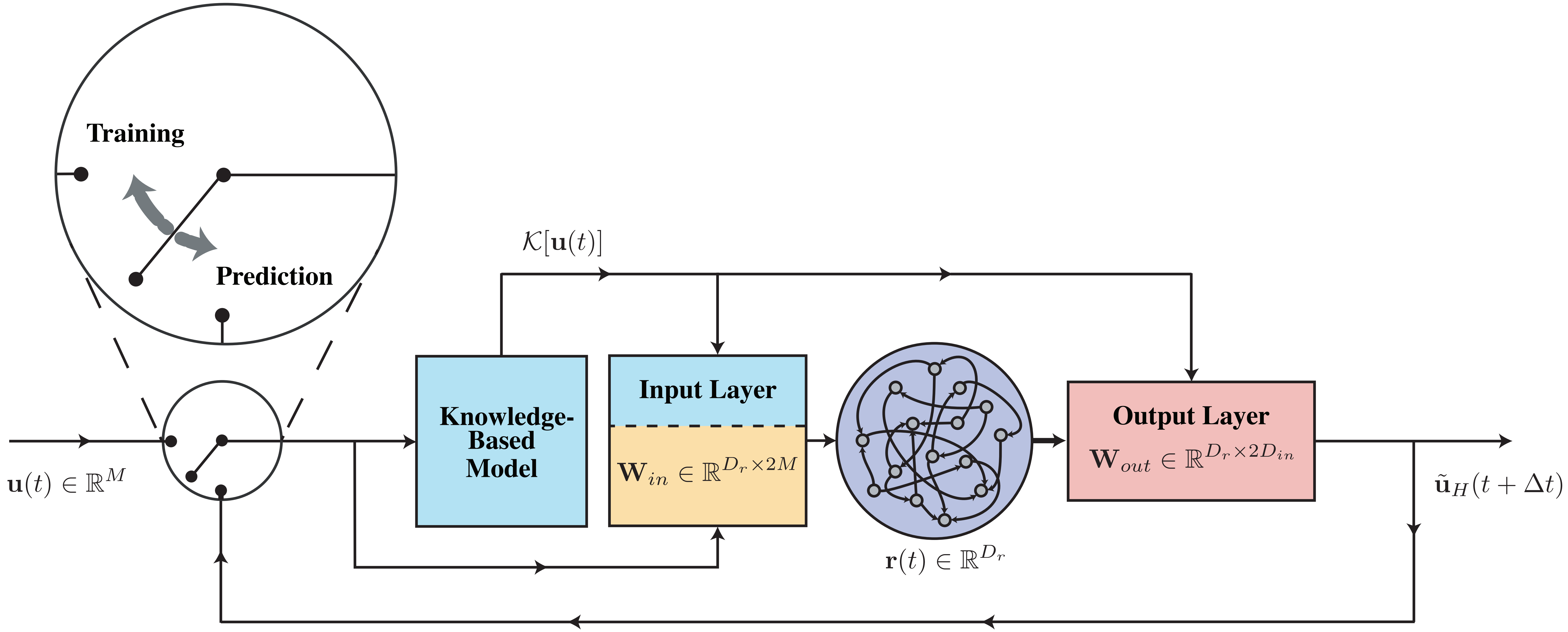
DATA-ASSISTED MODELS: COMBINING KNOWLEDGE AND DATA

Even with parallelization, if the system is very complex, a purely data-driven approach may require a prohibitive amount of training data or computational resources.

However, in many cases, we may have partial knowledge of the dynamics which can give us an imperfect physics-based model.

This motivates the question: Can we build a hybrid data-assisted model that combines machine learning with an imperfect knowledge-based model?

DATA-ASSISTED MODEL



DATA-ASSISTED MODEL

Ground Truth:

$$y_t = -yy_x - y_{xx} - y_{xxxx}$$

Imperfect Model:

$$y_t = -yy_x - (1+\epsilon)y_{xx} - y_{xxxx}$$

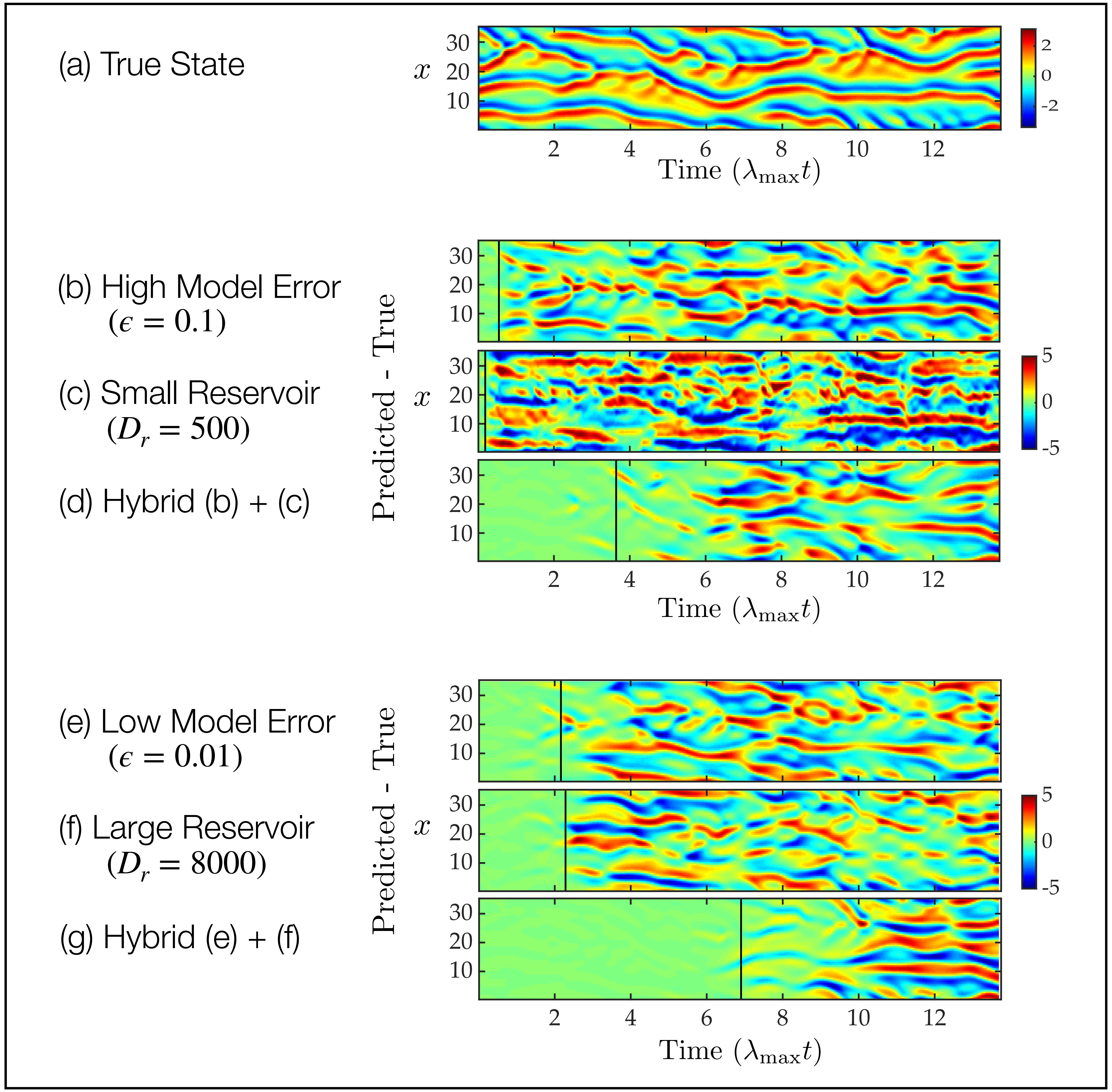
DATA-ASSISTED MODEL

Ground Truth:

$$y_t = -yy_x - y_{xx} - y_{xxxx}$$

Imperfect Model:

$$y_t = -yy_x - (1+\epsilon)y_{xx} - y_{xxxx}$$



RESERVOIR COMPUTING AND OTHER RNN ARCHITECTURES

Reservoir computers/ Echo state networks offer very quick and cheap training as compared to Recurrent Neural Networks trained using gradient descent and backpropagation (e.g. deep layered models like LSTM networks). We are currently performing systematic comparisons.

Training time for the examples shown today is approximately ~10 min on our cluster of CPUs. Although, this is at the cost of using hundreds of cores for large spatiotemporal systems.

OUR NEXT STEPS (IN PROGRESS)

Extend the parallelized hybrid machine learning framework to a simplified but full scale atmospheric general circulation model (SPEEDY developed and maintained by the ICTP).

OUR NEXT STEPS (IN PROGRESS)

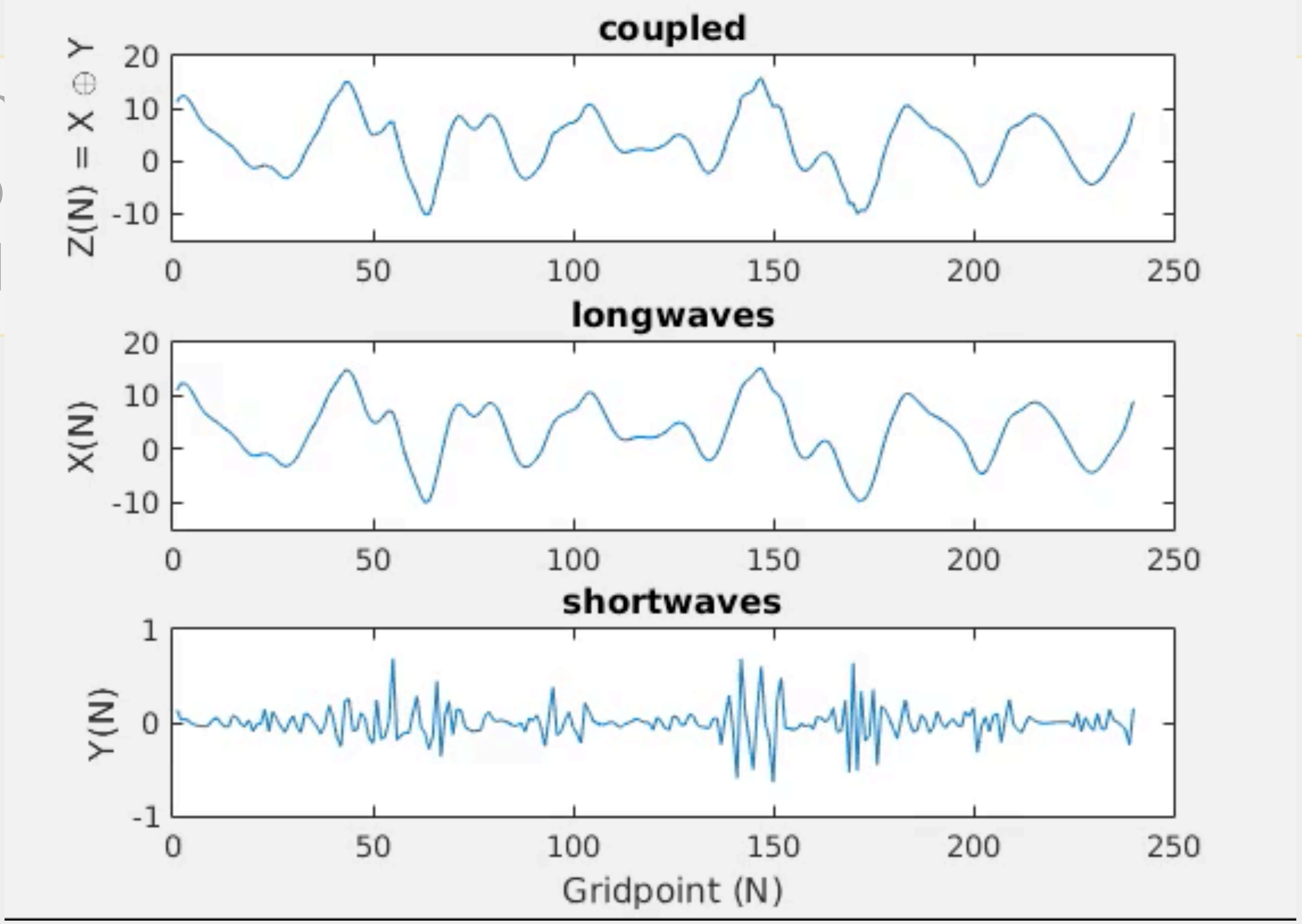
Extend the parallelized hybrid machine learning framework to a simplified but full scale atmospheric general circulation model (SPEEDY developed and maintained by the ICTP).

Consider a problem with coupled dynamics on multiple length scales. Use machine learning to deal with the subgrid-scale dynamics that could be difficult to model from first principles.

OUR NEXT STEPS (IN PROGRESS)

Extend the parallelized hybrid machine learning framework to a simplified but full scale atmospheric general circulation model (SPEEDY developed and maintained by the ICTP).

Consider a problem with
Use machine learning to
could be difficult to mod



OUR NEXT STEPS (IN PROGRESS)

Extend the parallelized hybrid machine learning framework to a simplified but full scale atmospheric general circulation model (SPEEDY developed and maintained by the ICTP).

Consider a problem with coupled dynamics on multiple length scales. Use machine learning to deal with the effects of subgrid-scale dynamics that could be difficult to model from first principles.

Use data assimilation methods such as an ensemble Kalman filter to incorporate sparse and noisy measurements into the machine learning framework.

CONCLUSIONS

Machine Learning can be a very effective tool for studying high dimensional chaos, including problems which are difficult to tackle with traditional tools.

Potential applications include weather prediction, understanding neural activity, control of chaos, improved simulation techniques for PDEs.

REFERENCES

Jaeger et al. "**Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication.**" Science 304.5667 (2004): 78-80.
A paper that introduced reservoir computing.

Pathak et al. "**Model-free prediction of large spatiotemporally chaotic systems from data: a reservoir computing approach.**" Physical Review Letters 120.2 (2018): 024102.
Prediction of high dimensional attractors.

Pathak et al. "**Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data.**" Chaos 27.12 (2017): 121102.
Estimating Lyapunov exponents.

Pathak et al. "**Hybrid forecasting of chaotic processes: using machine learning in conjunction with a knowledge-based model.**" Chaos 28.4 (2018): 041101.
Machine learning in conjunction with incomplete knowledge-based models.

Lu et al. "**Attractor reconstruction by machine learning.**" Chaos 28.6 (2018): 061104.
An attempt to understand attractor reconstruction through the lens of generalized synchronization.

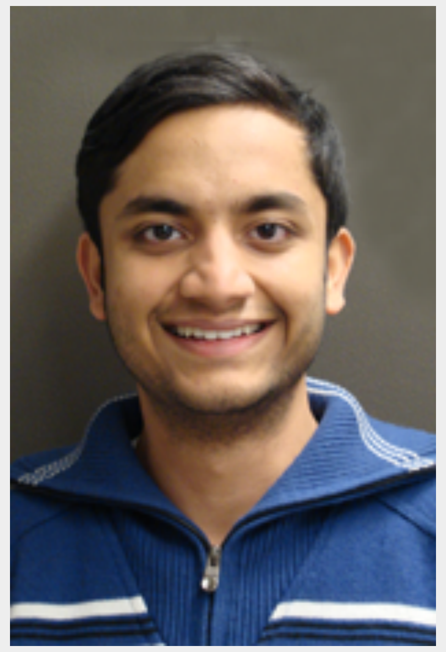
THANK YOU!

My Collaborators:

Prof. Michelle Girvan
Prof. Brian Hunt
Prof. Edward Ott
(University of Maryland, College Park)

Alexander Wikner
Rebeckah Fussell

Dr. Zhixin Lu
(University of Pennsylvania)



Sarthak Chandra
(University of Maryland)

Supported by:

U.S. Army Research Office.

DARPA.

Laboratory of Telecommunication Sciences.