ETH zürich

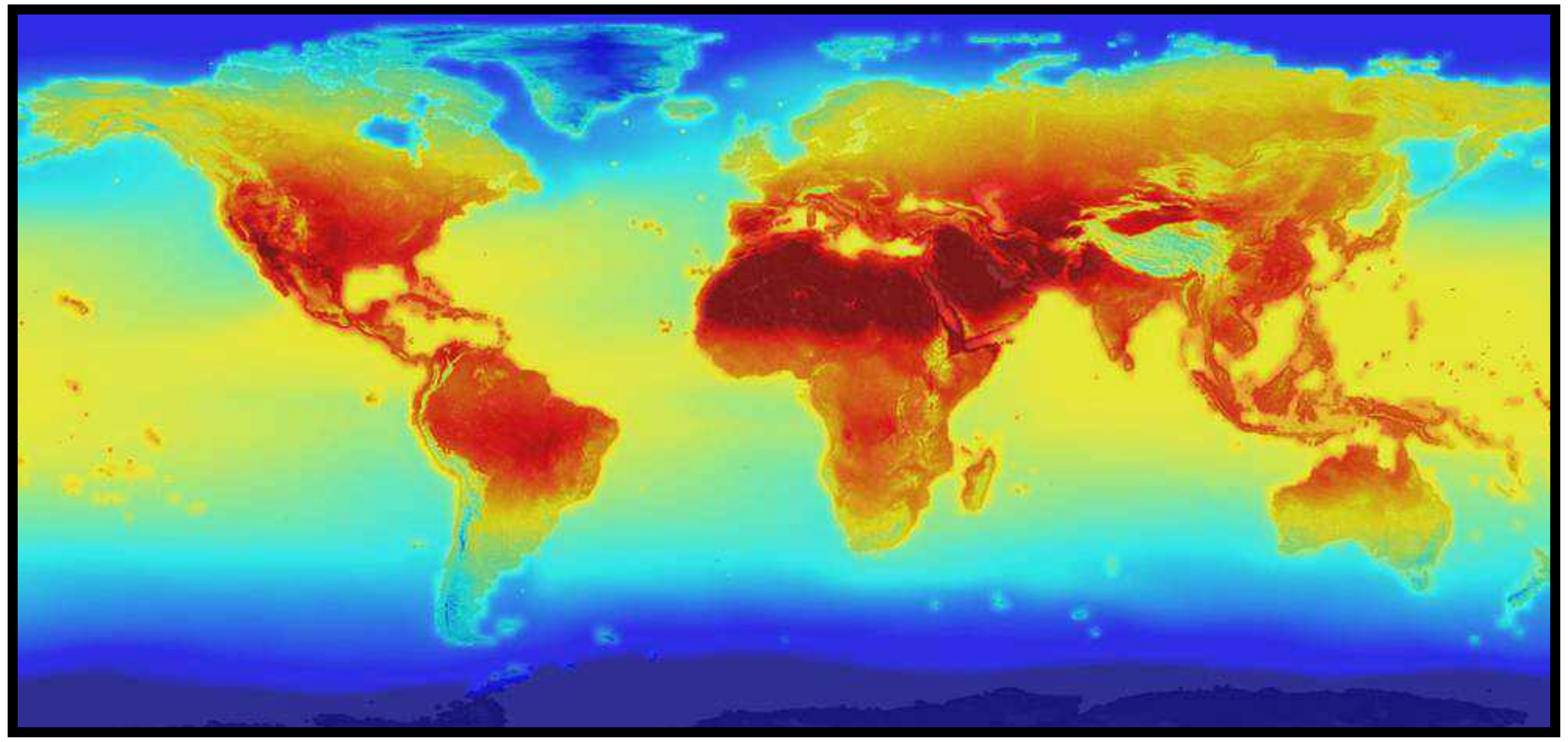
Third Symposium on Machine Learning and Dynamical Systems
September 27, 2022

Learning and Forecasting the Effective Dynamics of Complex Systems across Scales

Pantelis R. Vlachas
AI2C Technologies, CSE-Lab ETH Zurich

Motivation

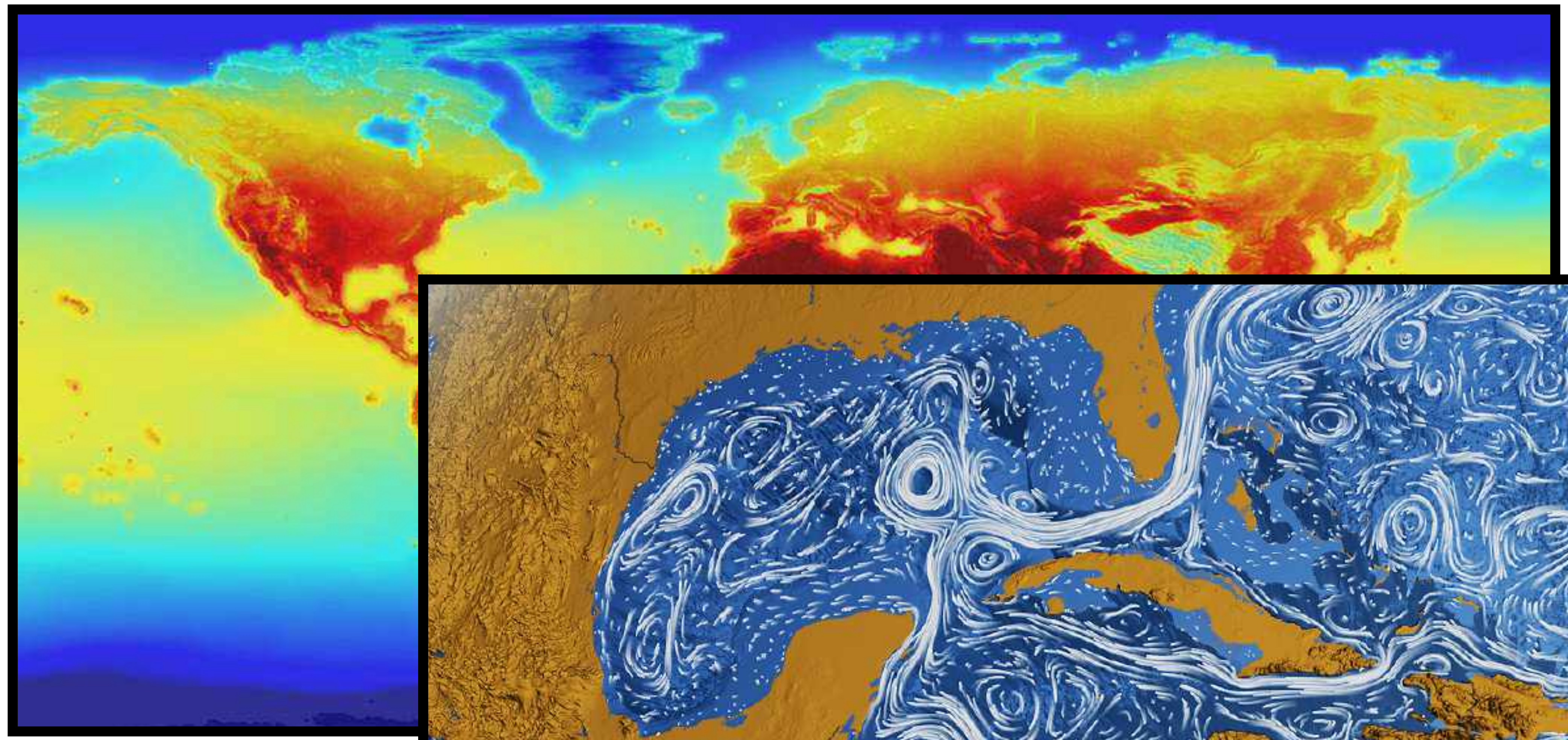
Motivation



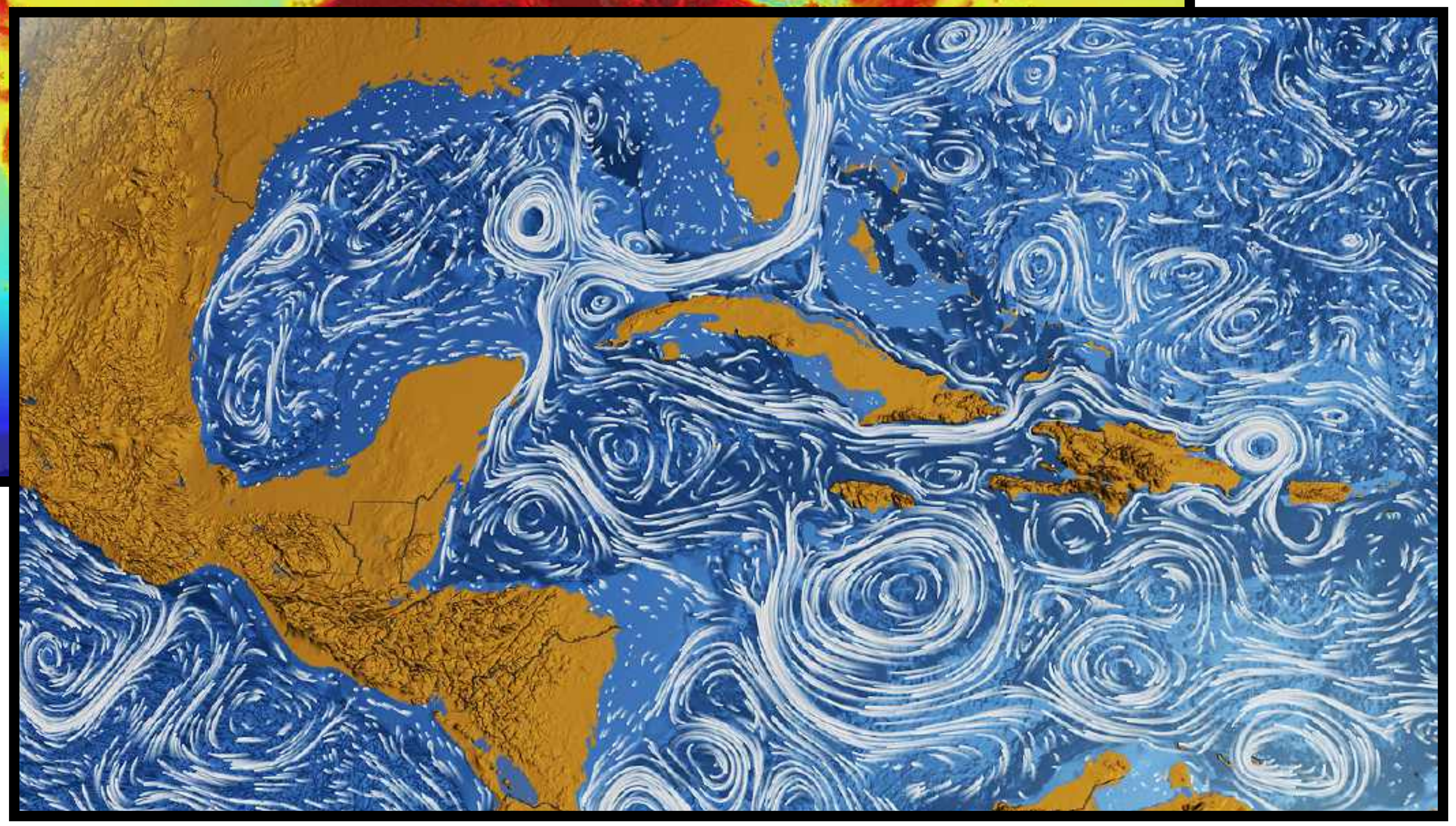
Climate

- Complex multiscale systems
  (deterministic, stochastic, chaotic)

Motivation



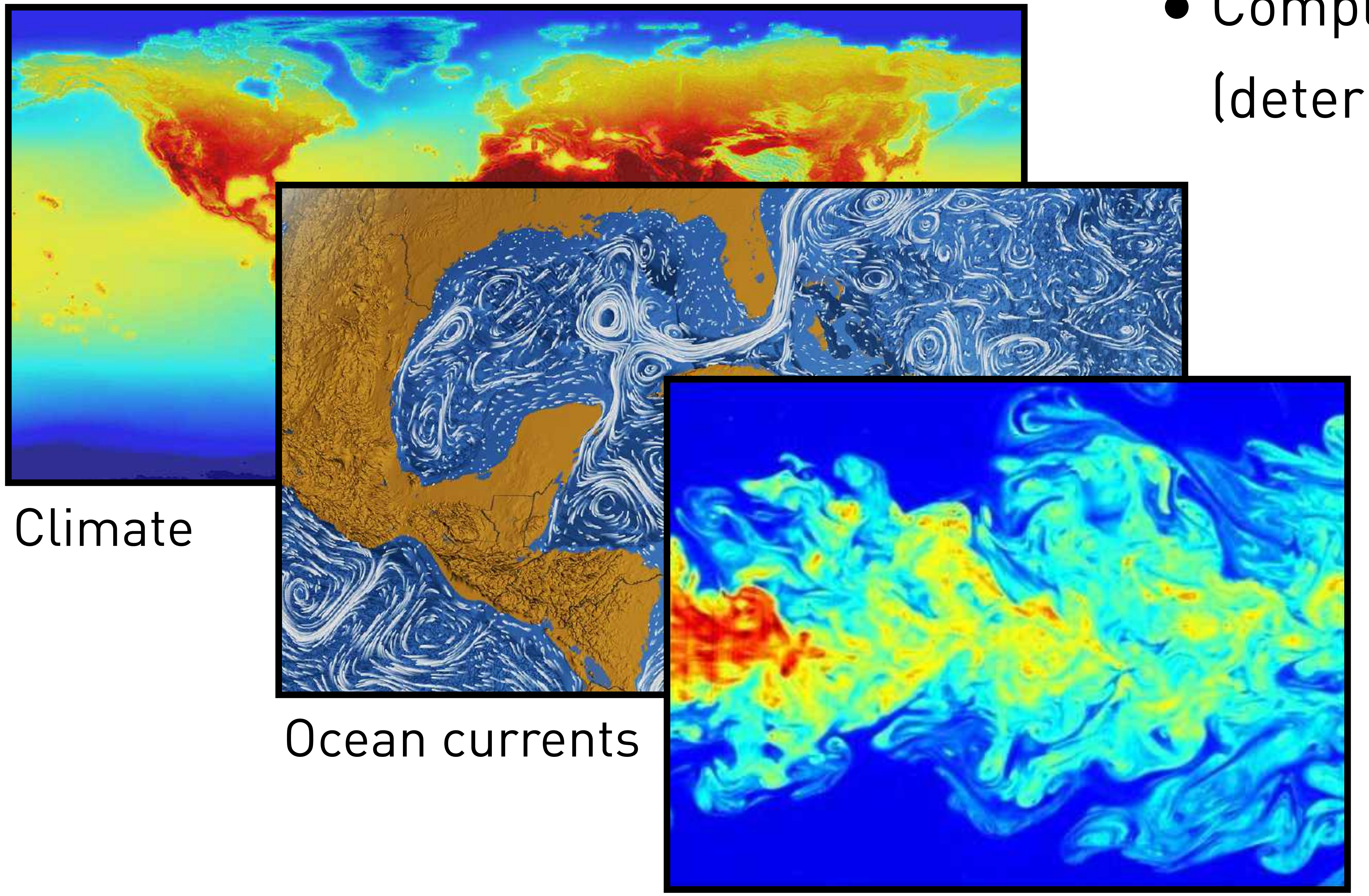
Climate



Ocean currents

- Complex multiscale systems
  (deterministic, stochastic, chaotic)

Motivation



Climate

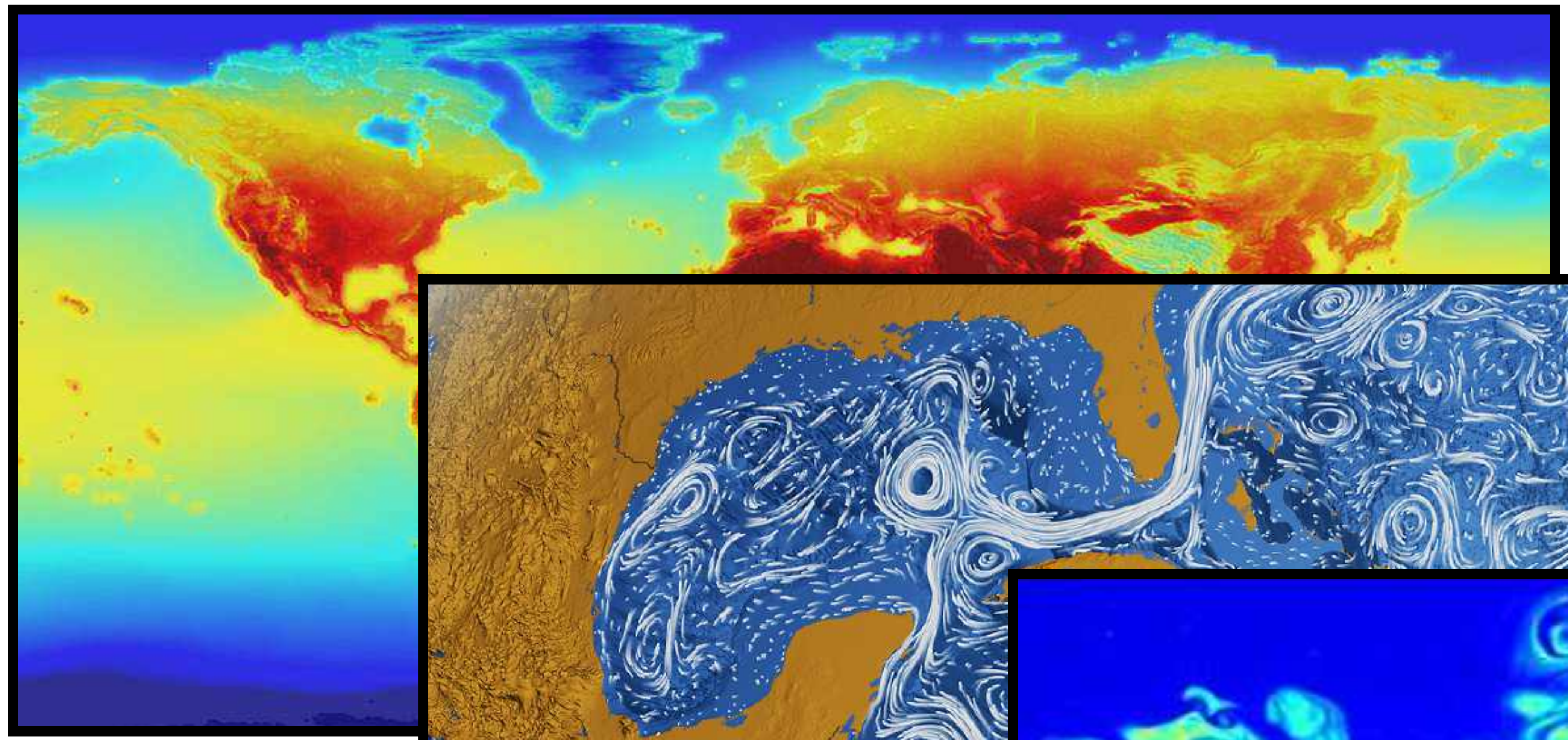
Ocean currents

Turbulence

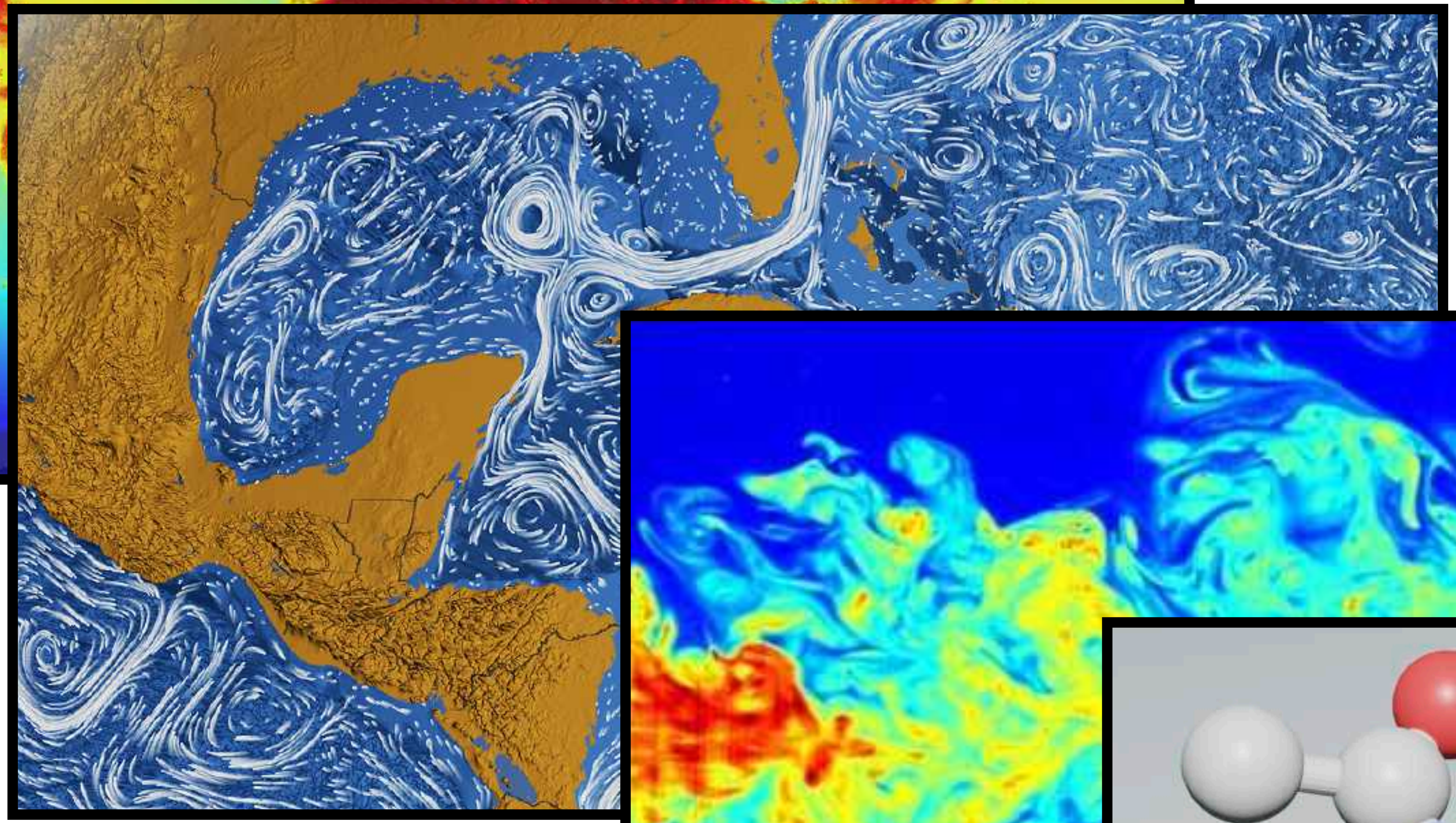
- Complex multiscale systems (deterministic, stochastic, chaotic)

Motivation

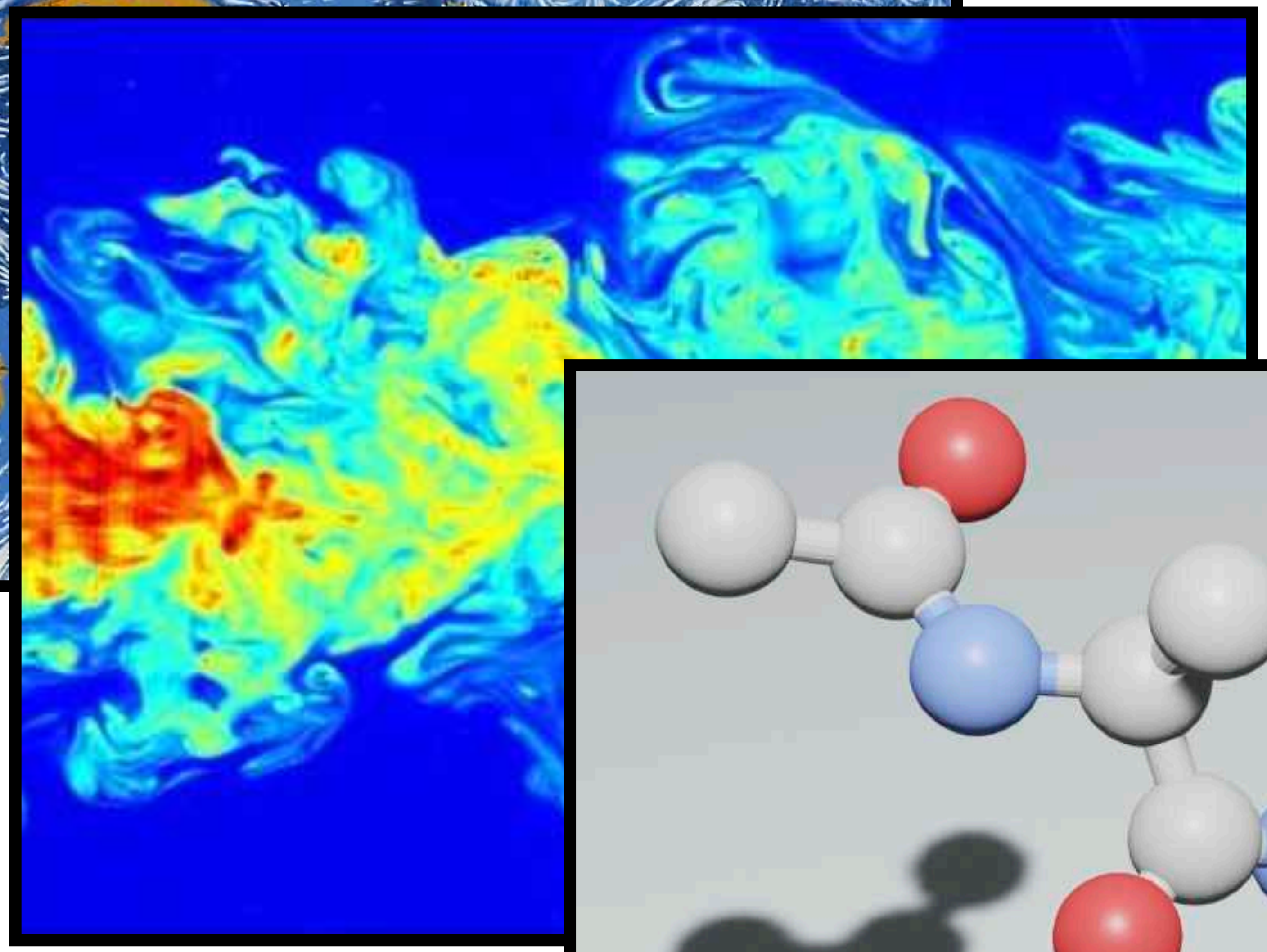
- Complex multiscale systems (deterministic, stochastic, chaotic)



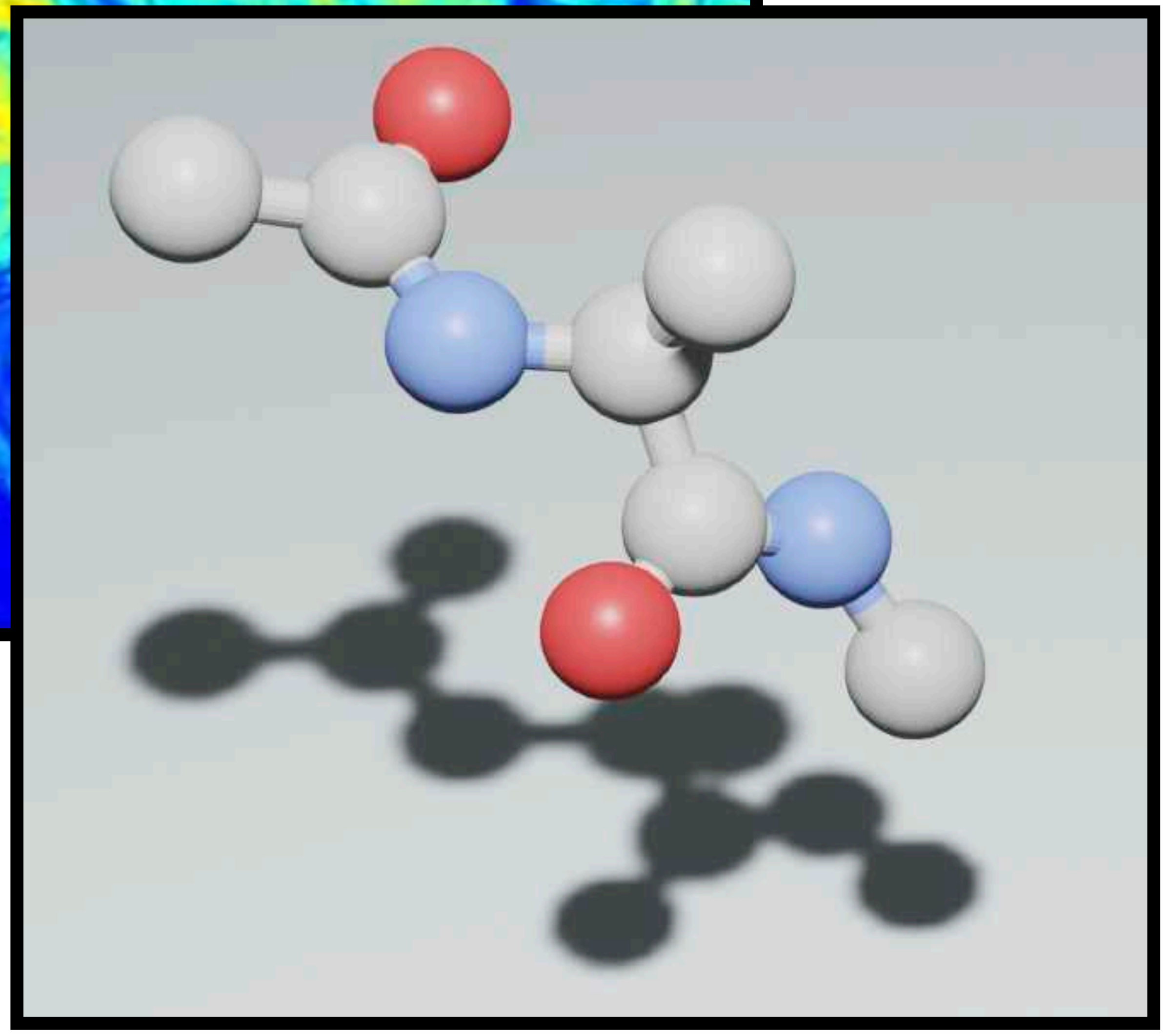
Climate



Ocean currents



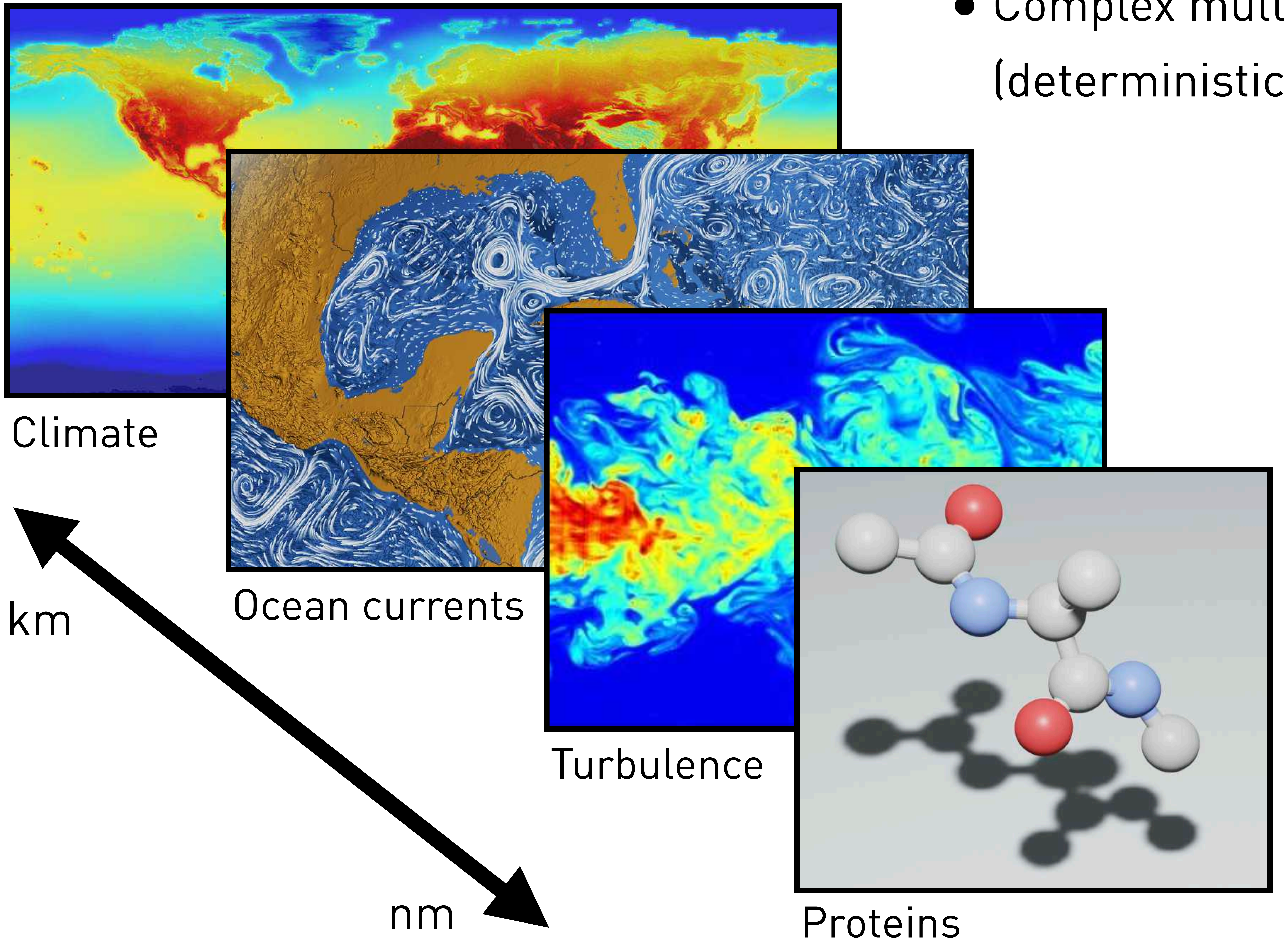
Turbulence



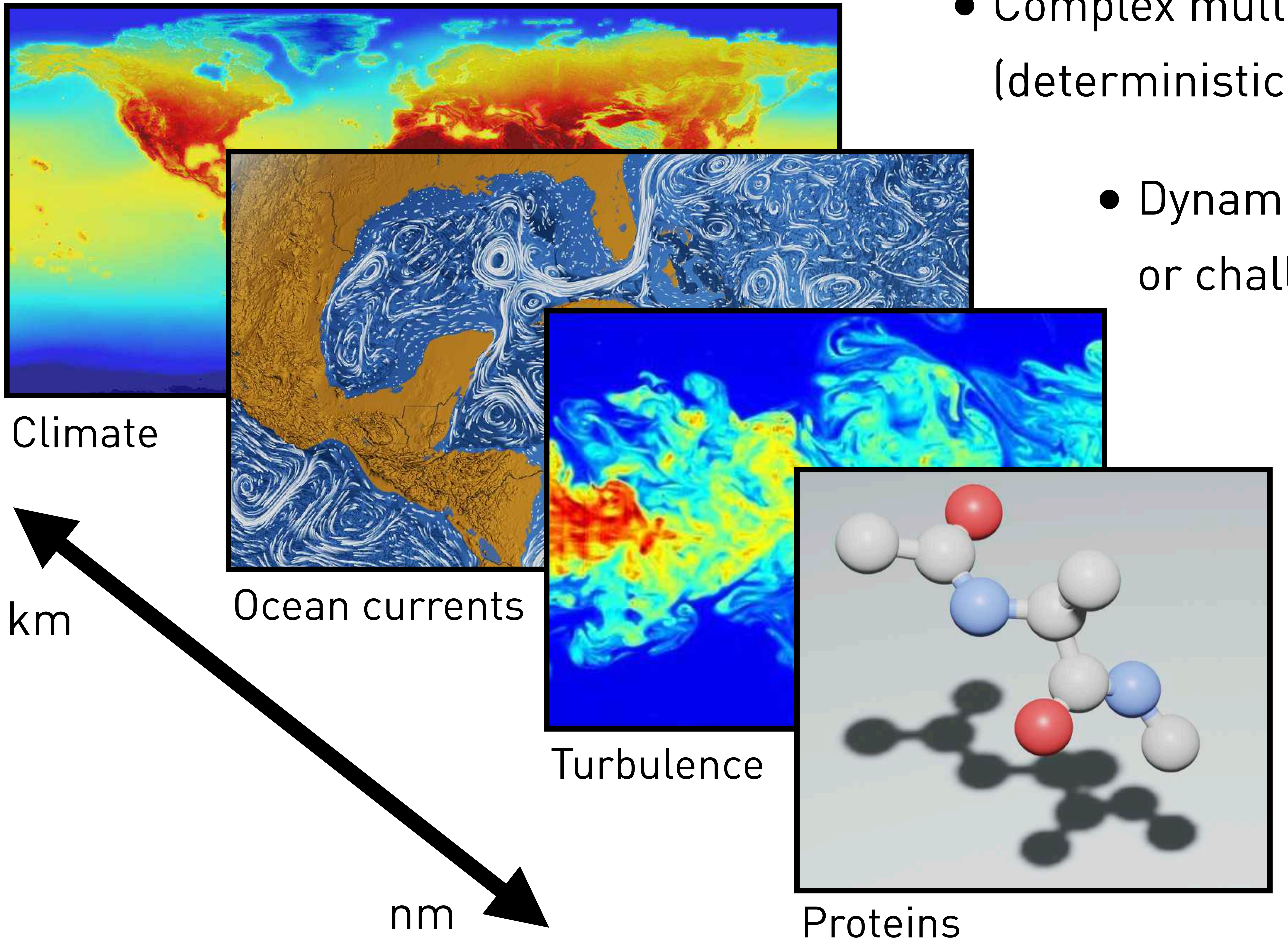
Proteins

Motivation

- Complex multiscale systems (deterministic, stochastic, chaotic)

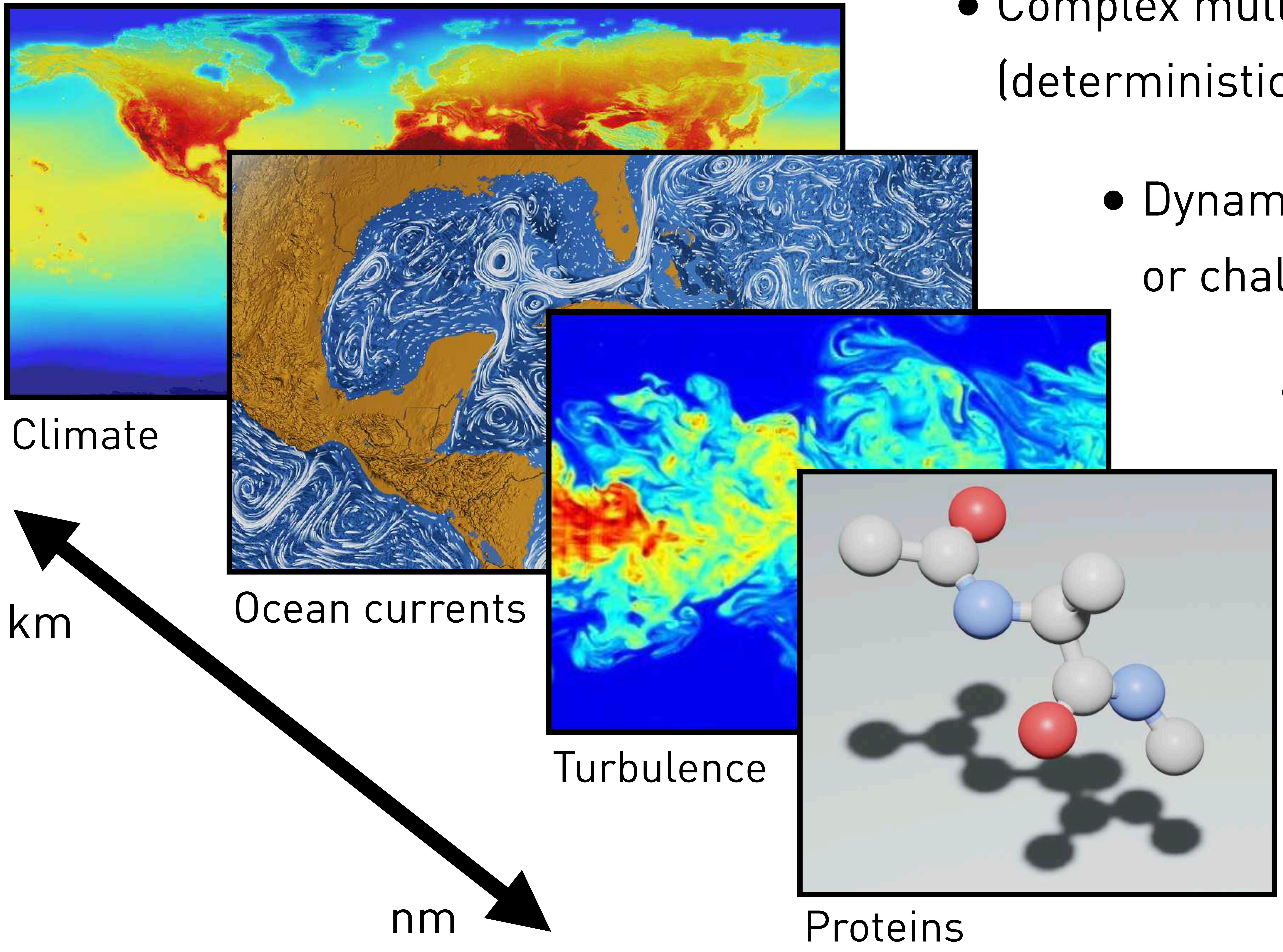


Motivation



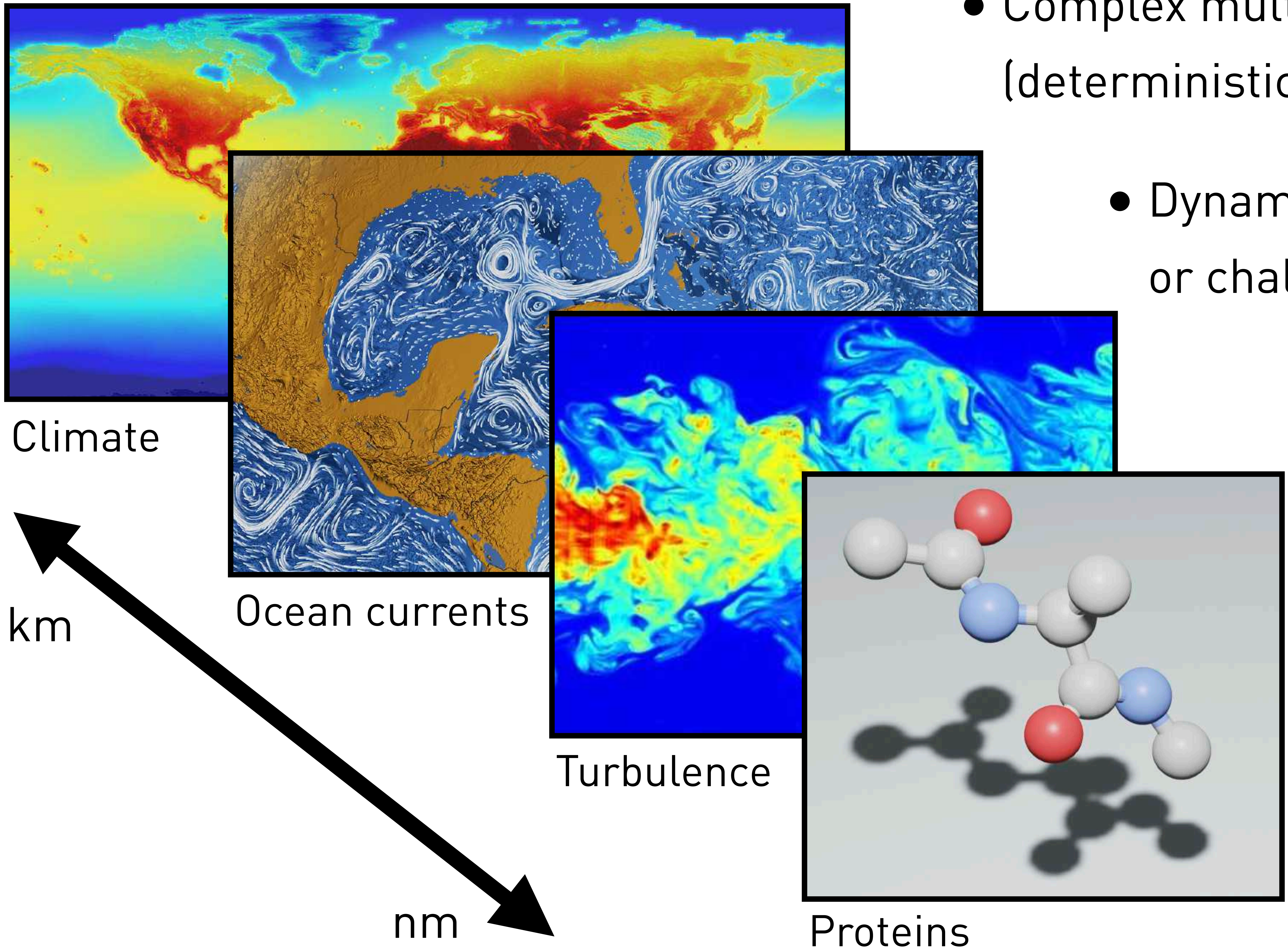
- Complex multiscale systems (deterministic, stochastic, chaotic)
  - Dynamics expensive to simulate and/or challenging to forecast

Motivation



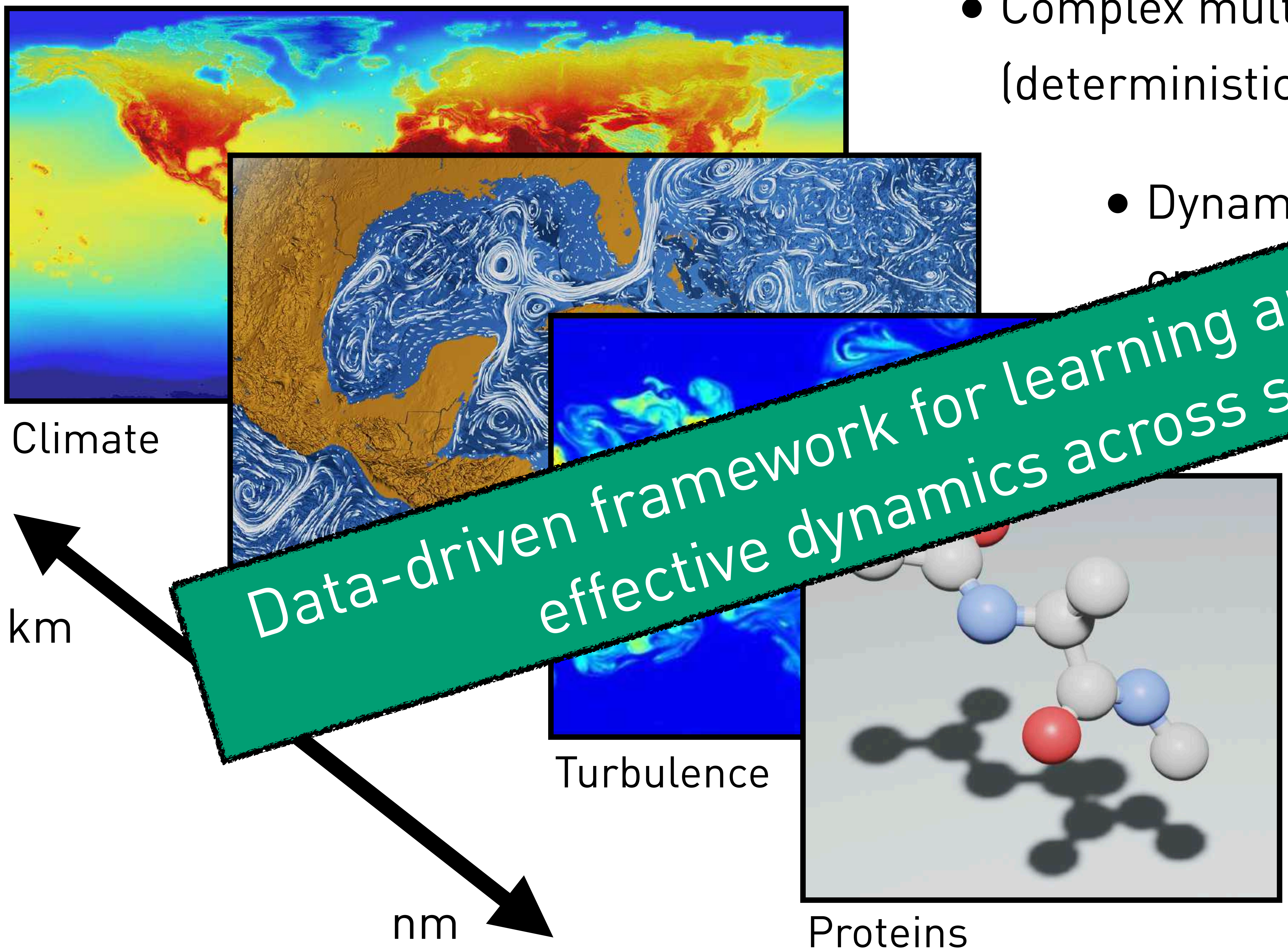
- Complex multiscale systems (deterministic, stochastic, chaotic)
  - Dynamics expensive to simulate and/or challenging to forecast
    - Can we design fast (multiscale) methods that reproduce system dynamics?

Motivation



- Complex multiscale systems (deterministic, stochastic, chaotic)
    - Dynamics expensive to simulate and/or challenging to forecast
        - Can we design fast (multiscale) methods that reproduce system dynamics?
            - Examples:
              LES/RANS
              Surrogate models / DMD
              Coarse graining models of molecular systems

Motivation



- Complex multiscale systems (deterministic, stochastic, chaotic)
  - Dynamics exp… ate and/ o…
    - … we design fast (multiscale) methods that reproduce system dynamics?
      - Examples:

        LES/RANS

        Surrogate models / DMD

        Coarse graining models of molecular systems

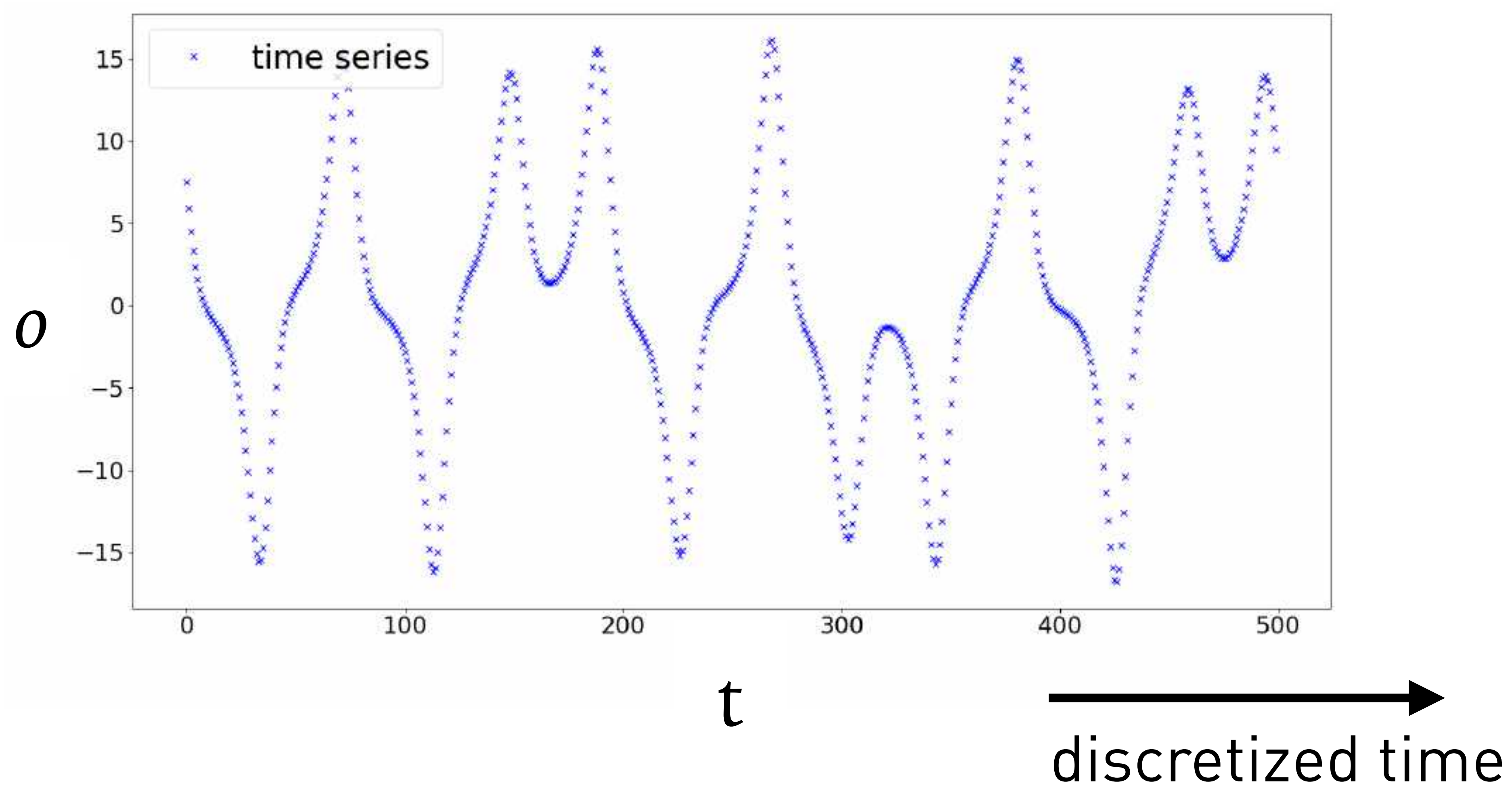
Data-driven framework for learning and forecasting effective dynamics across scales

I

Forecasting Complex Dynamics with Recurrent Neural Networks

Recurrent Neural Networks (RNNs)

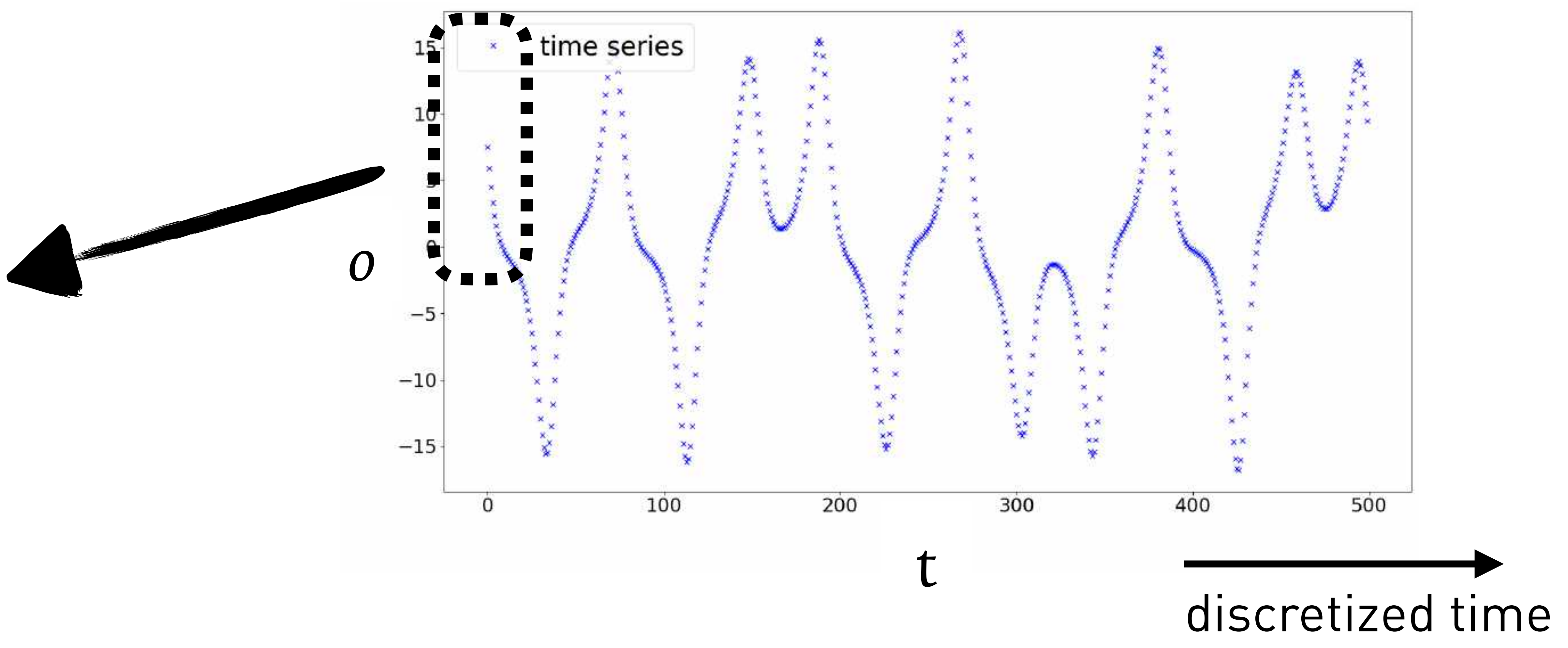
Recurrent Neural Networks (RNNs)



Data from trajectories

- Sensory data / noisy
- Unknown underlying dynamics
- No equations based on first principles (physics)
- Does not describe full system state

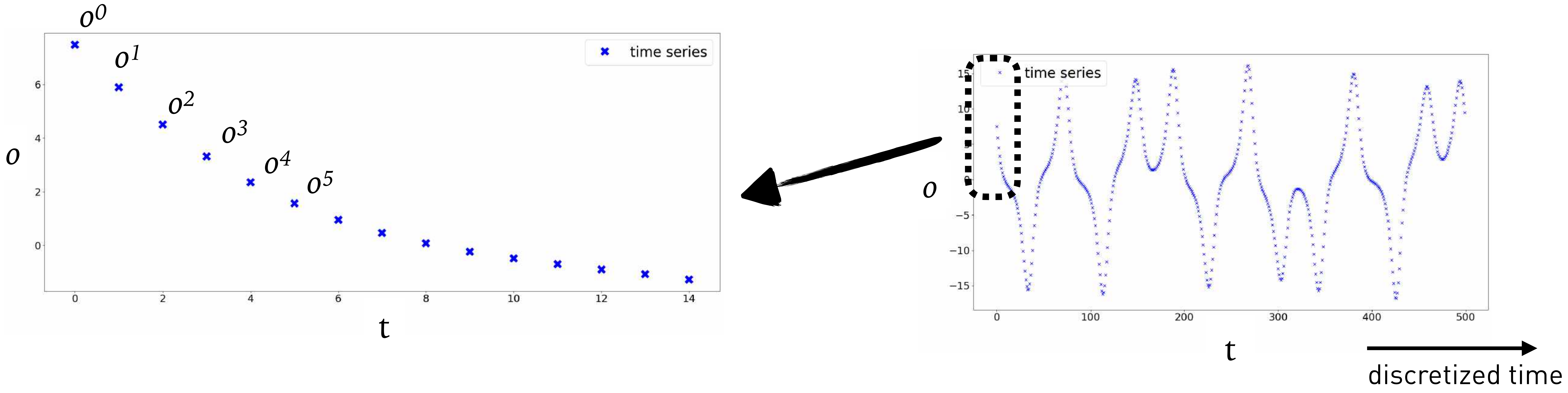
Recurrent Neural Networks (RNNs)



Data from trajectories

- Sensory data / noisy
- Unknown underlying dynamics
- No equations based on first principles (physics)
- Does not describe full system state

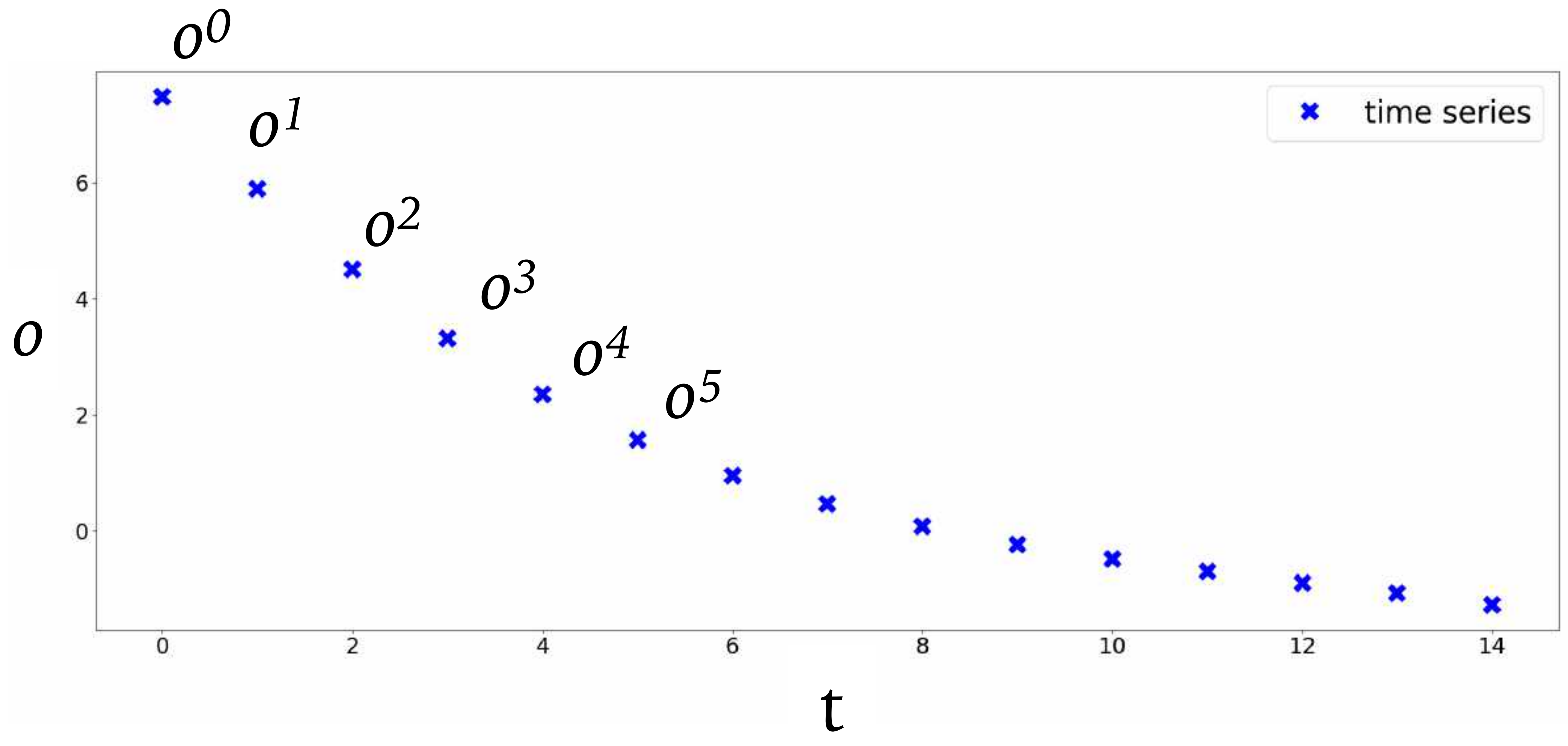
Recurrent Neural Networks (RNNs)



Data from trajectories

- Sensory data / noisy
- Unknown underlying dynamics
- No equations based on first principles (physics)
- Does not describe full system state

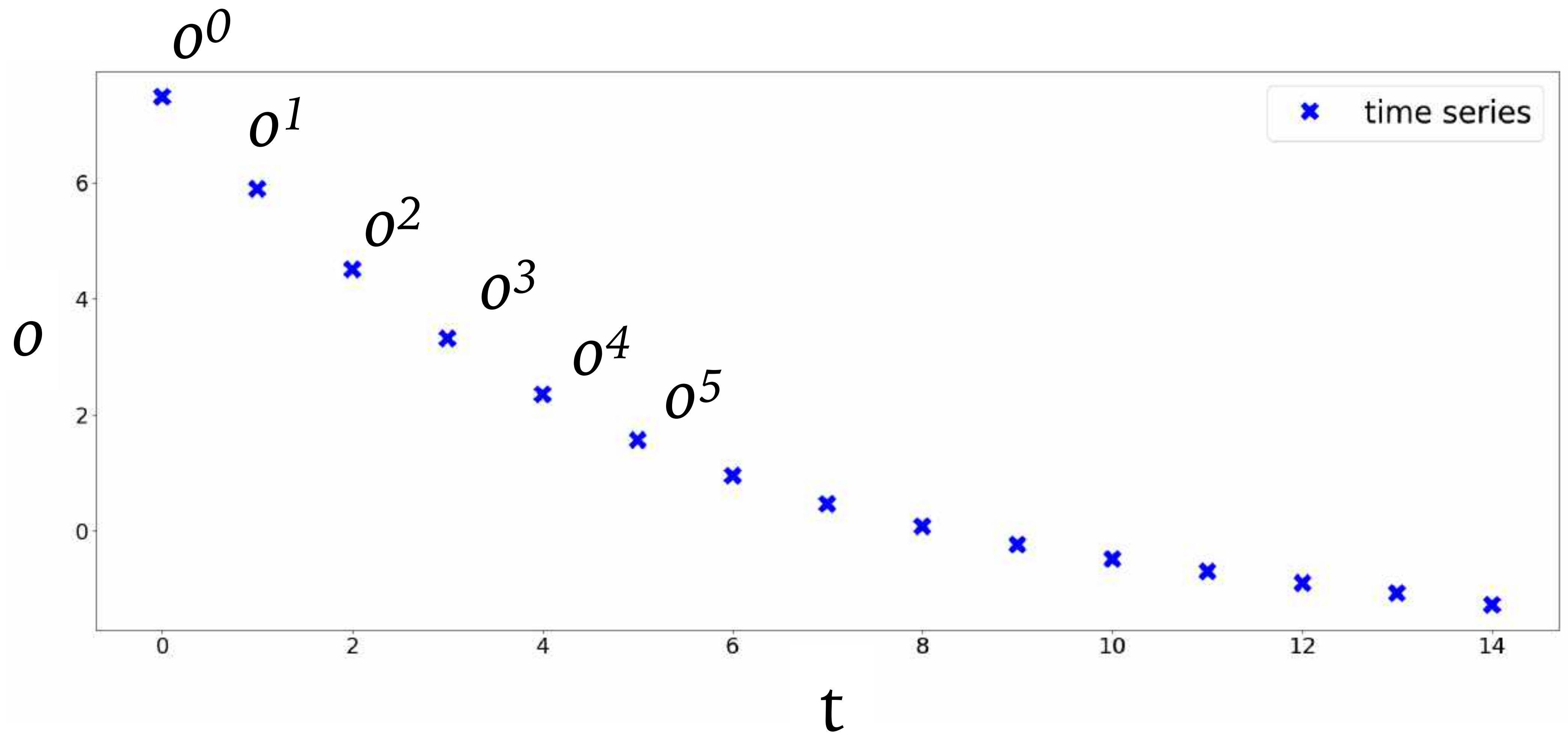
Recurrent Neural Networks (RNNs)



Model $o^{t+1} = o^t + \Delta t\, \dot{o}^t$

with $\dot{o}^t = f(o^t, o^{t-1}, o^{t-2}, \ldots)$

Recurrent Neural Networks (RNNs)

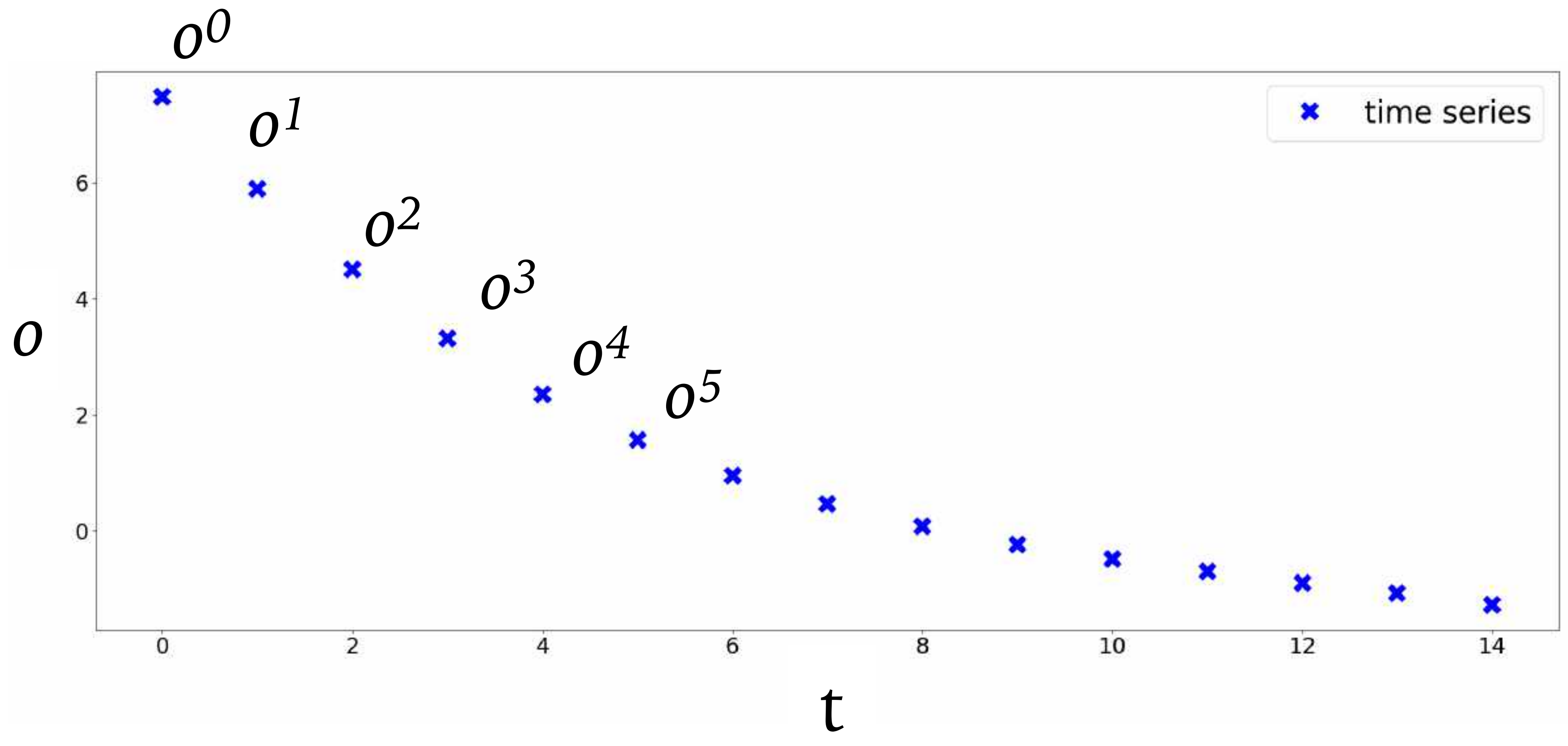


Model $o^{t+1} = o^t + \Delta t\, \dot{o}^t$

with $\dot{o}^t = f(o^t, \underbrace{o^{t-1}, o^{t-2}, \ldots}_{h^{t-1}})$

history encoded in h^{t-1}

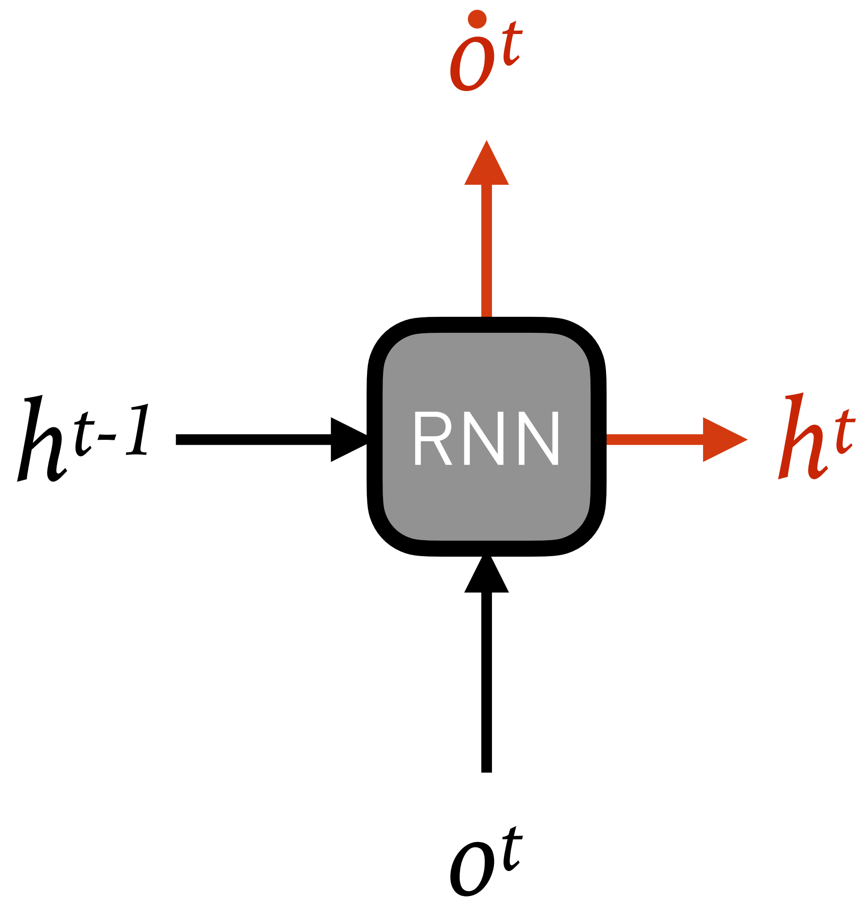
Recurrent Neural Networks (RNNs)



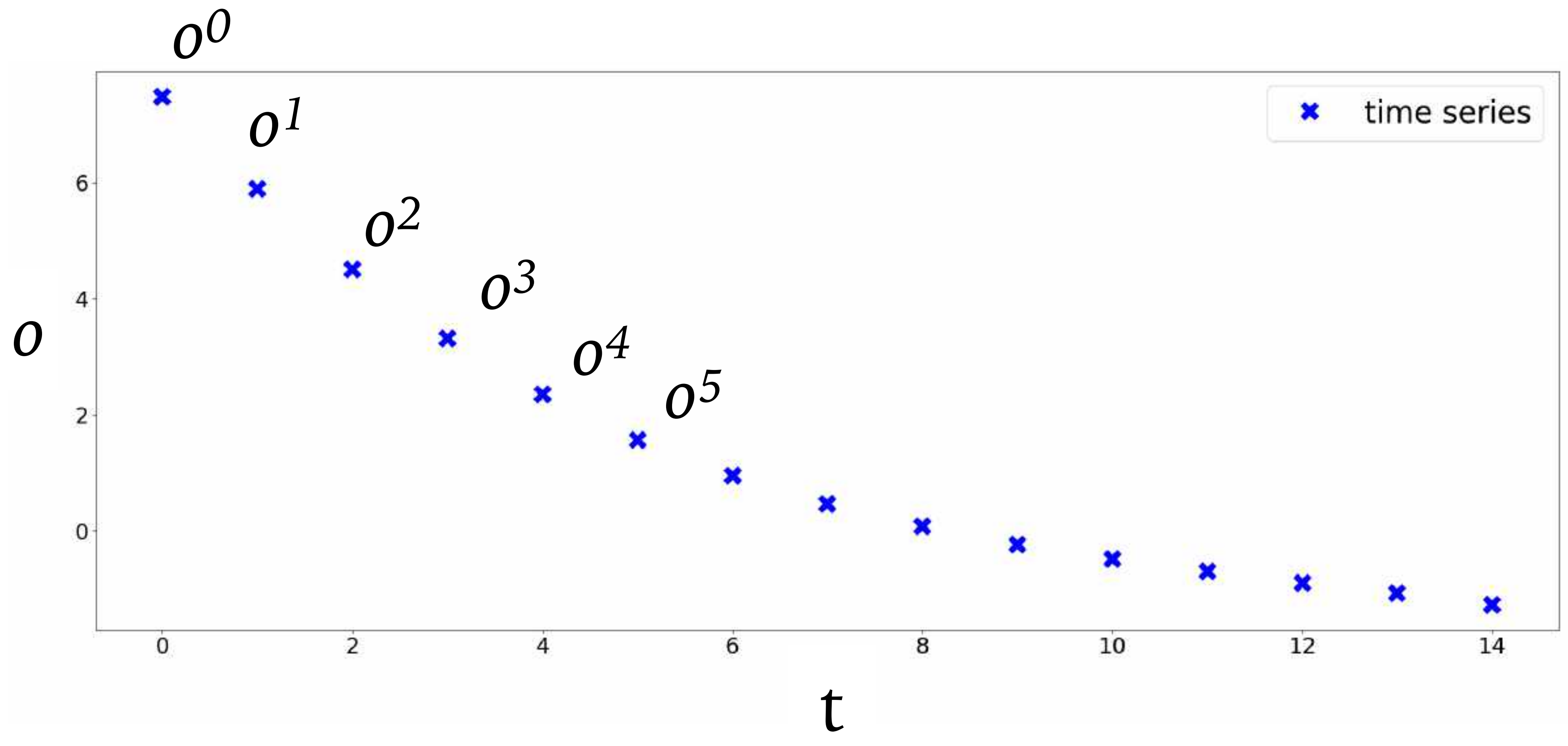
Model $o^{t+1} = o^t + \Delta t \, \dot{o}^t$

with $\dot{o}^t = f(\underbrace{o^t, o^{t-1}, o^{t-2}, \ldots}_{h^{t-1}})$

history
encoded in h^{t-1}



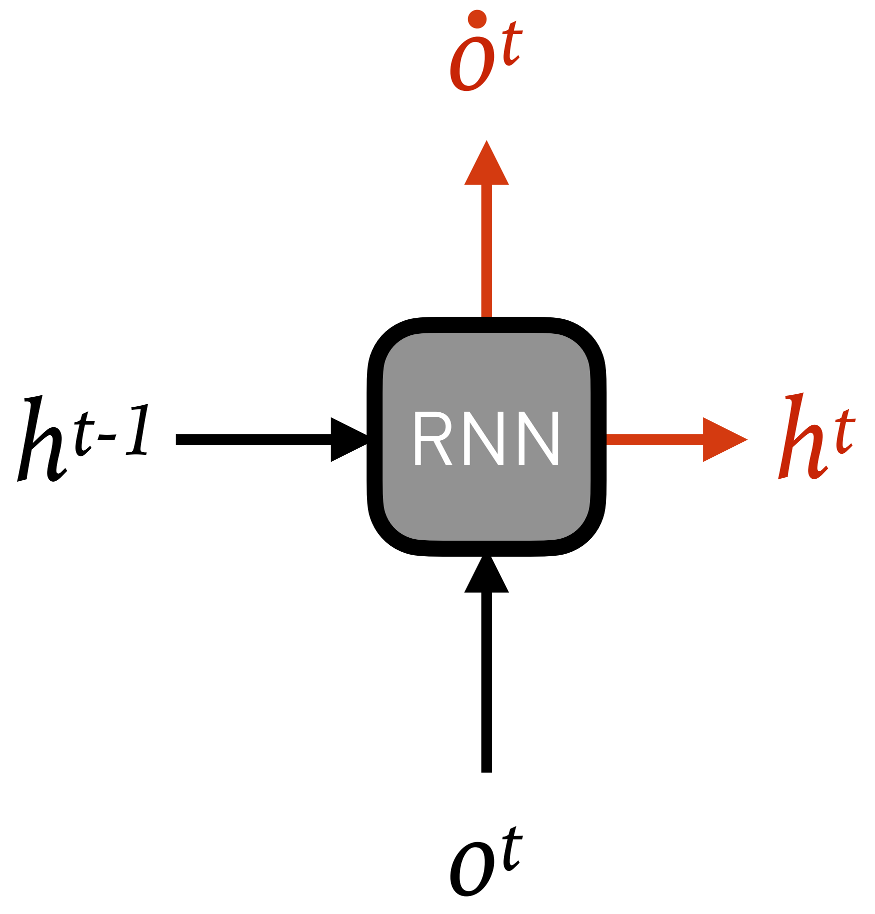
Recurrent Neural Networks (RNNs)



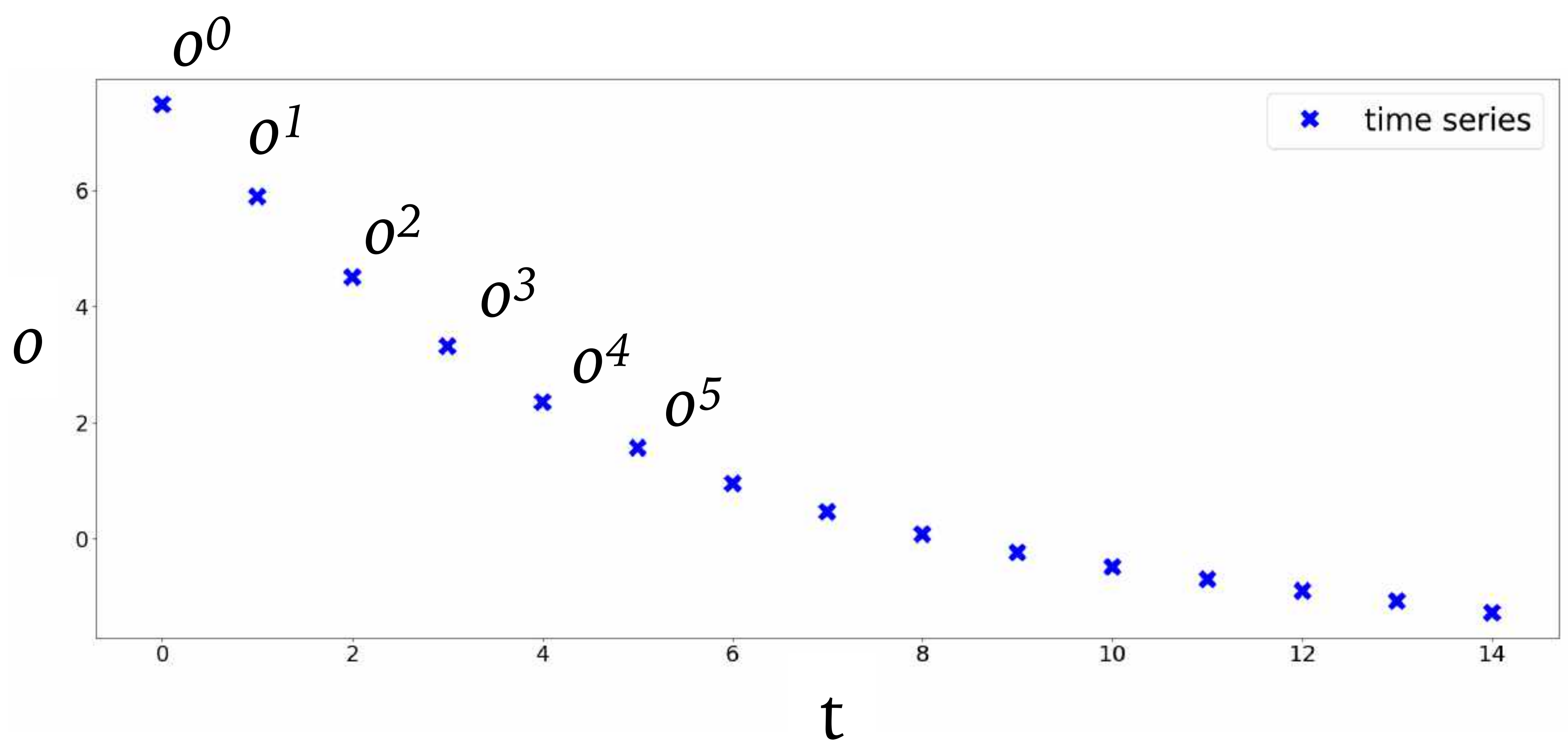
Model $o^{t+1} = o^t + \Delta t\, \dot{o}^t$

with $\dot{o}^t = f(\underbrace{o^t, o^{t-1}, o^{t-2}, \dots}_{h^{t-1}})$

history
encoded in h^{t-1}



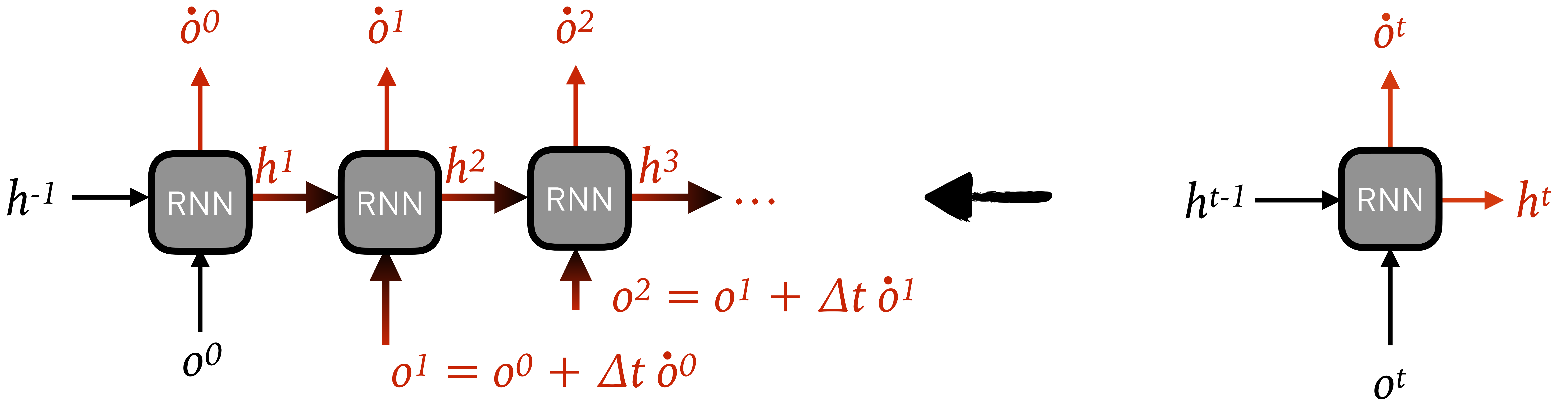
Recurrent Neural Networks (RNNs)



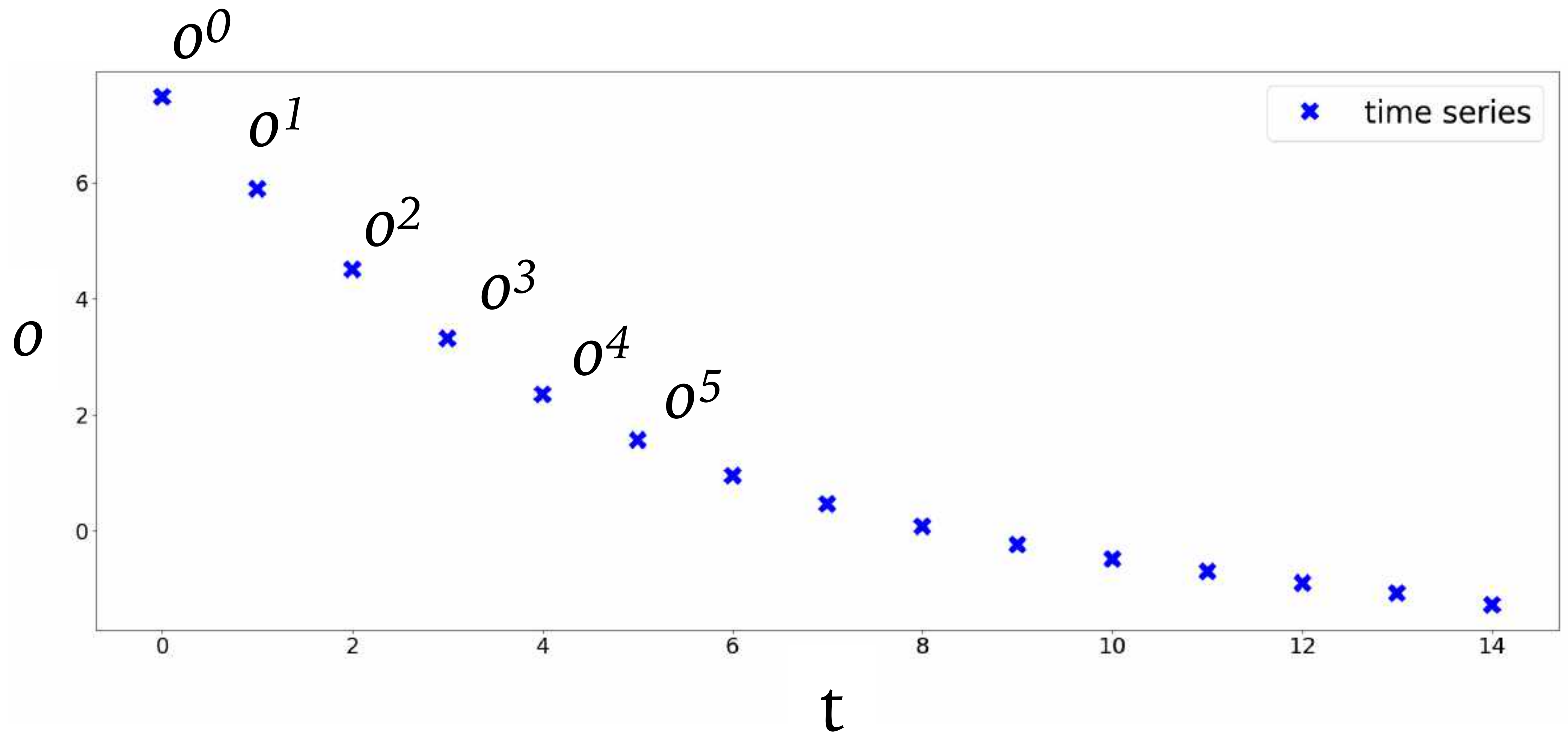
Model $o^{t+1} = o^t + \Delta t\, \dot{o}^t$

with $\dot{o}^t = f(o^t, \underbrace{o^{t-1}, o^{t-2}, \ldots}_{h^{t-1}})$

history encoded in h^{t-1}



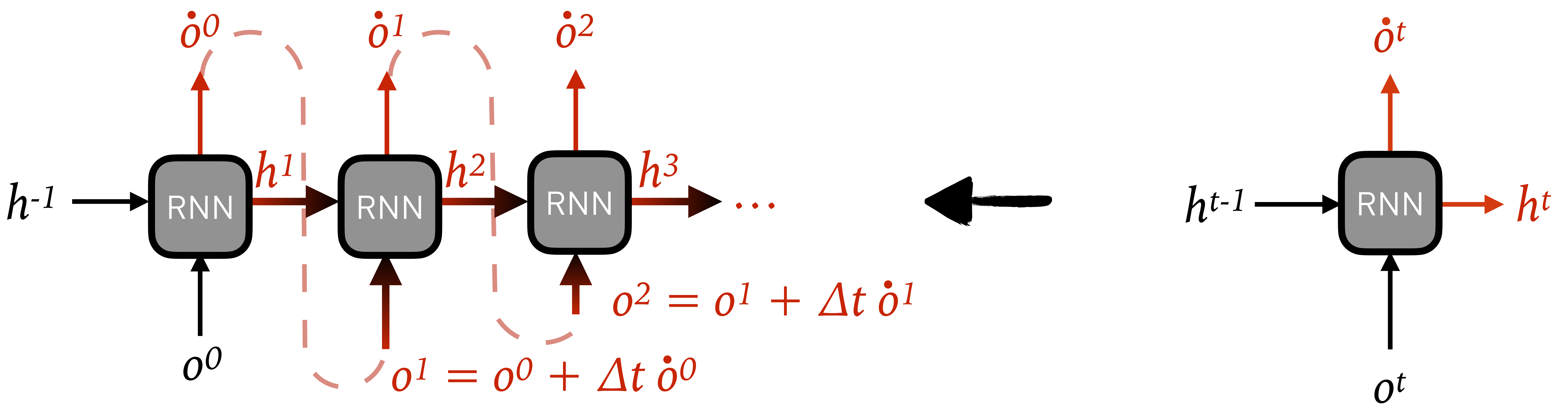
Recurrent Neural Networks (RNNs)



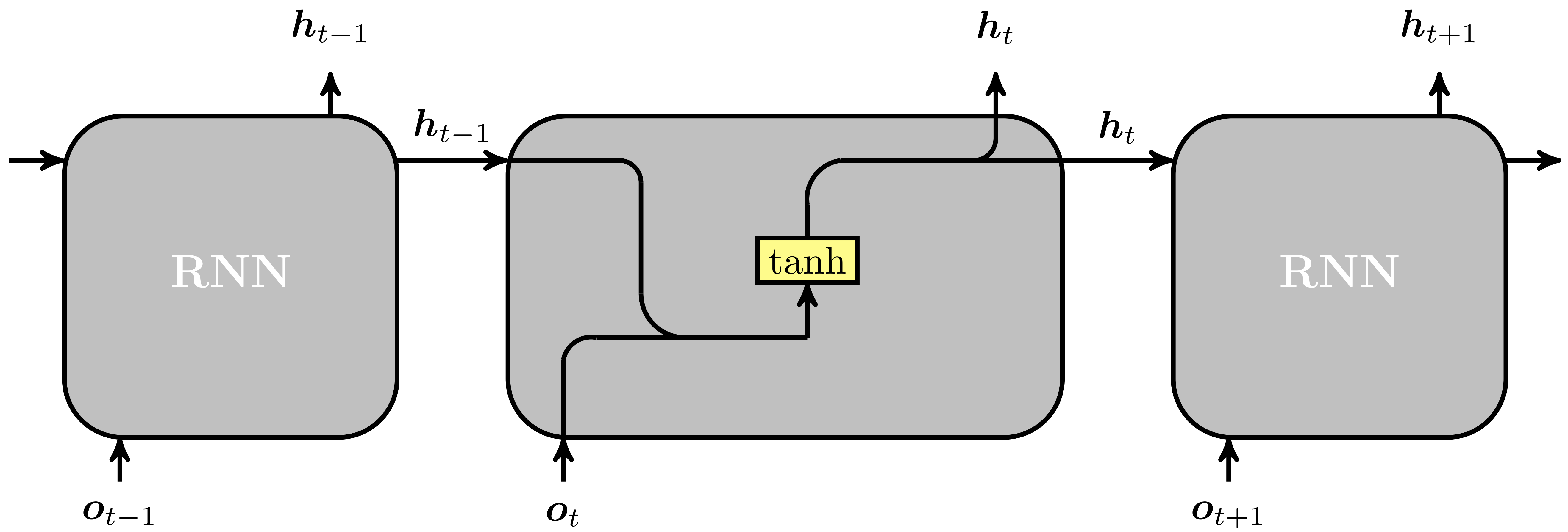
Model $o^{t+1} = o^t + \Delta t\, \dot{o}^t$

with $\dot{o}^t = f(\underbrace{o^t, o^{t-1}, o^{t-2}, \ldots}_{h^{t-1}})$

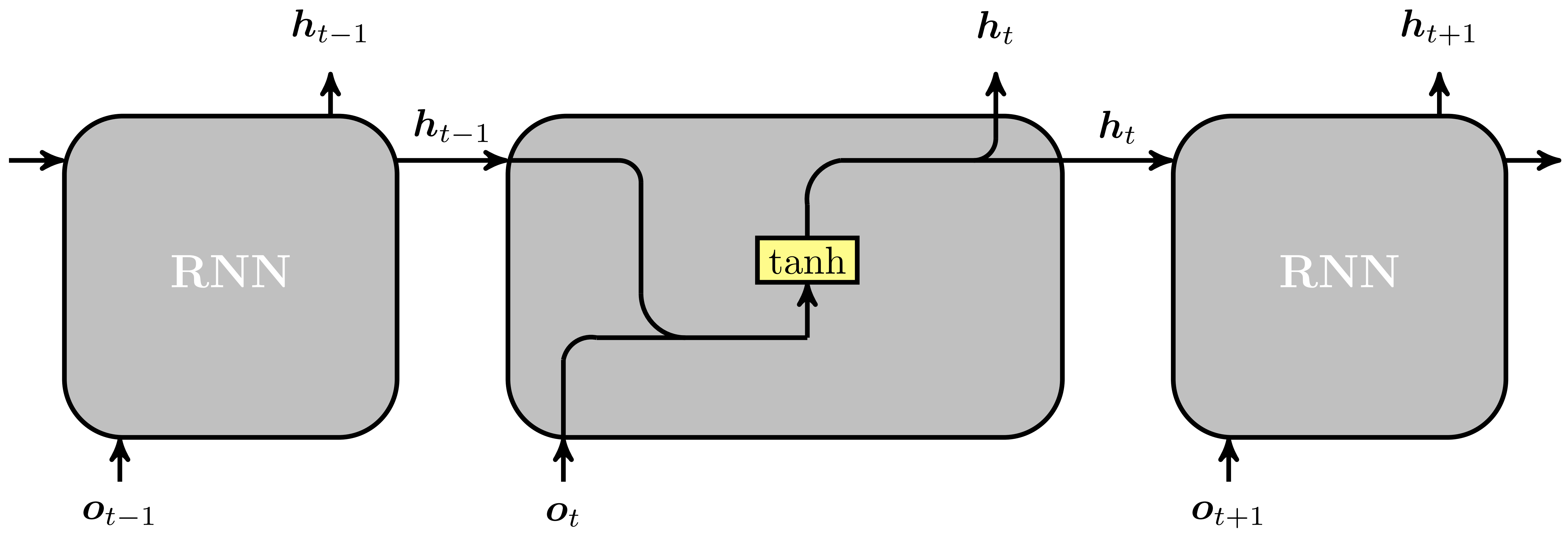
history encoded in h^{t-1}



Elman RNN (1990)



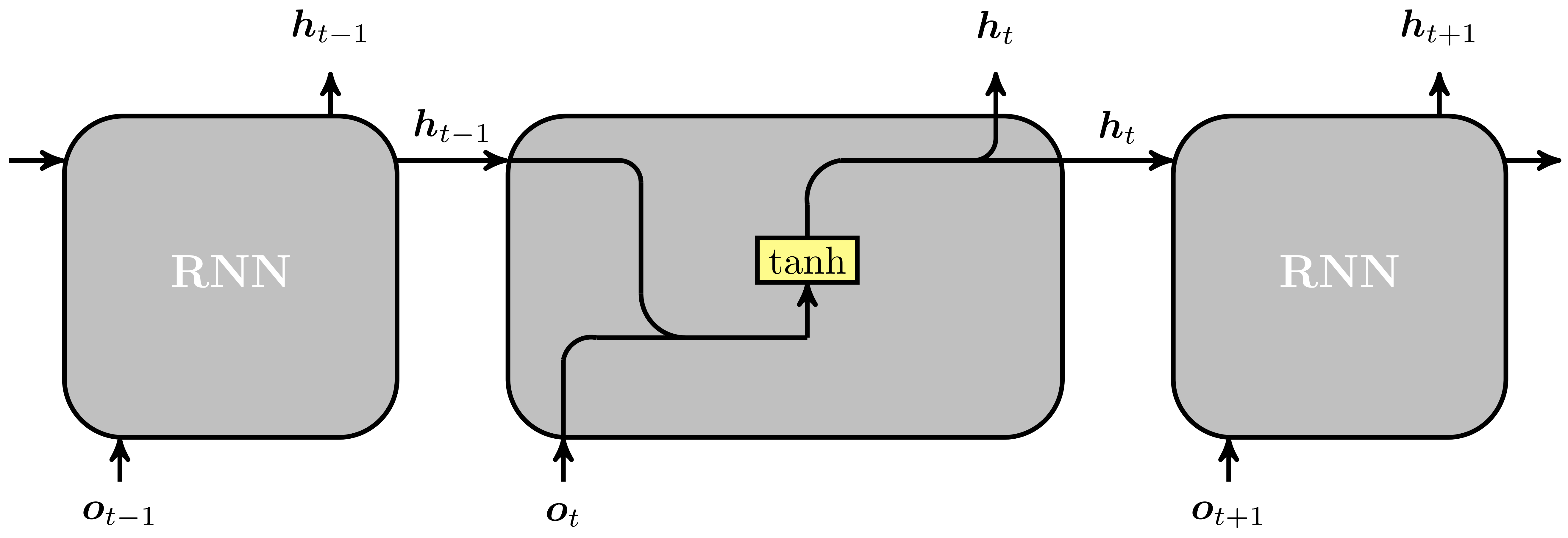
Elman RNN (1990)



Hidden-to-hidden mapping

$$h_t = \tanh\big(W_{ho}\, o_t + W_{hh}\, h_{t-1} + b_h\big)$$

Elman RNN (1990)



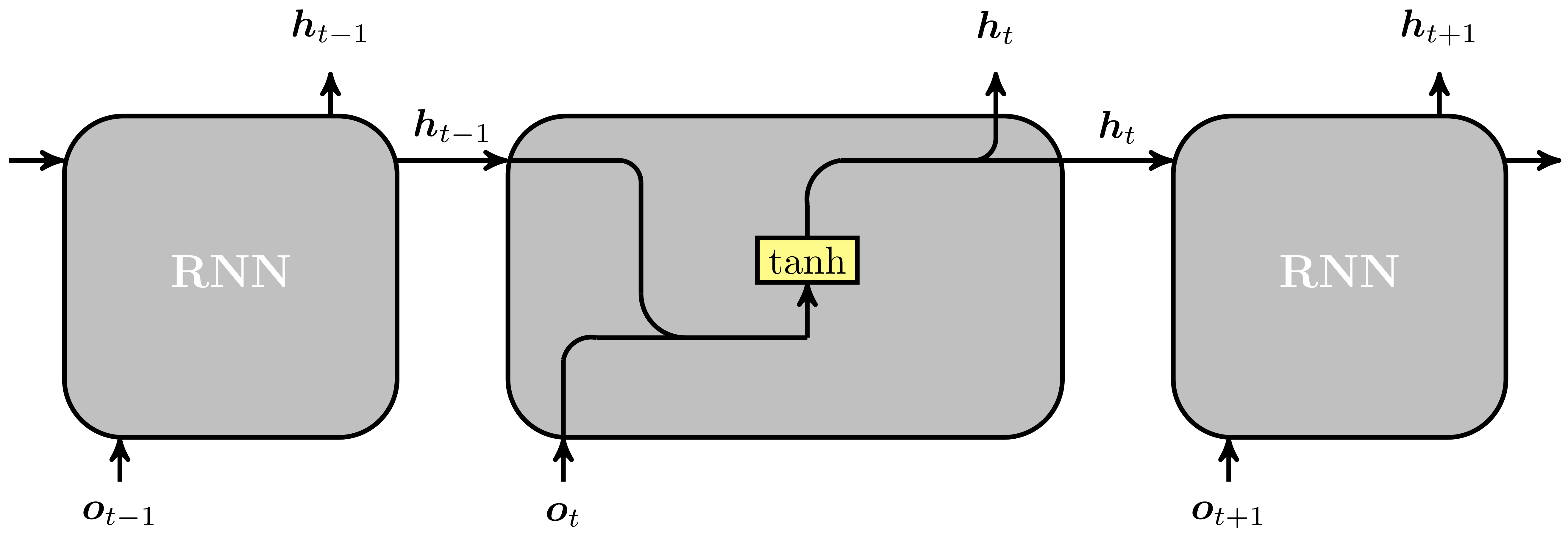
Hidden-to-hidden mapping

$$h_t = \tanh\left(W_{ho}\, o_t + W_{hh}\, h_{t-1} + b_h\right)$$

Hidden-to-output mapping

$$y_t = W_{oh}\, h_t \begin{cases} \hat{=}\ o_{t+1} \\ \textbf{or} \\ \hat{=}\ \dot{o}_t \end{cases}$$

Elman RNN (1990)



Hidden-to-hidden mapping

$$h_t = \tanh\left(W_{ho}\, o_t + W_{hh}\, h_{t-1} + b_h\right)$$

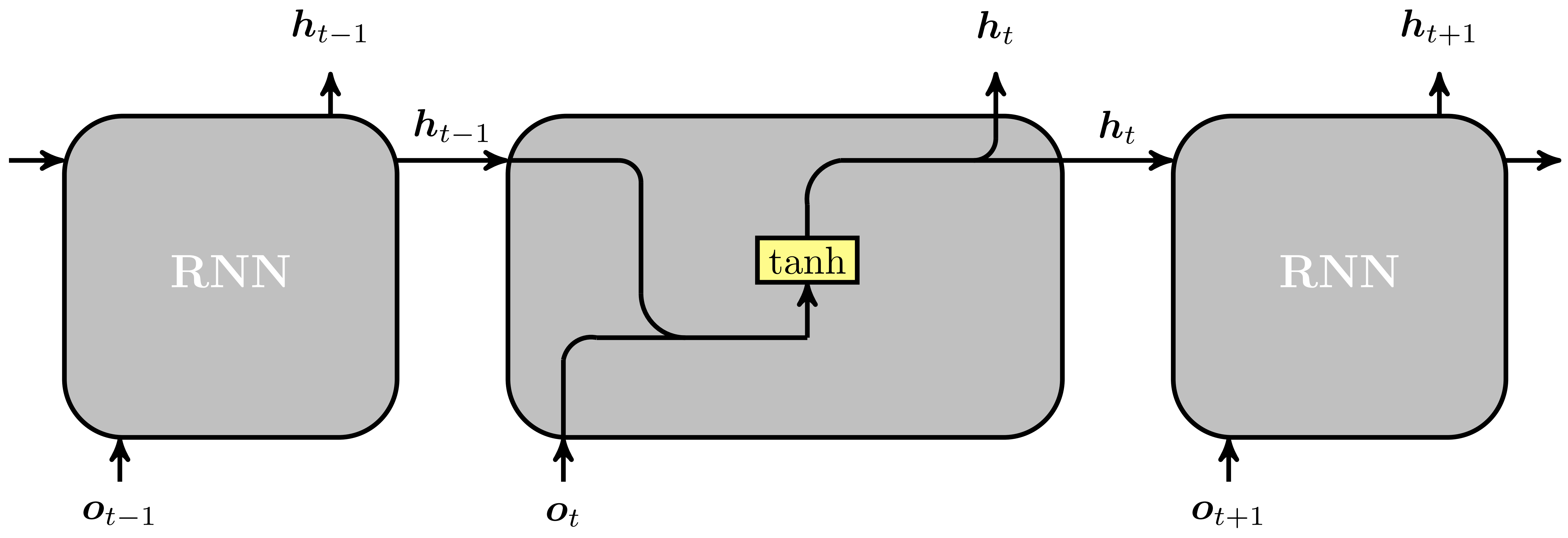
Hidden-to-output mapping

$$y_t = W_{oh}\, h_t \begin{cases} \hat{=}\; o_{t+1} \\ \textbf{or} \\ \hat{=}\; \dot{o}_t \end{cases}$$

Parameters (WEIGHTS) to be learned:

$$W_{ho}\,,\, W_{hh}\,,\, b_h\,,\, W_{oh}$$

Elman RNN (1990)



Training this network (fitting the WEIGHTS to data) is difficult.
(vanishing gradients problem)

Hidden-to-hidden mapping

$$h_t = \tanh\left(W_{ho}\, o_t + W_{hh}\, h_{t-1} + b_h\right)$$

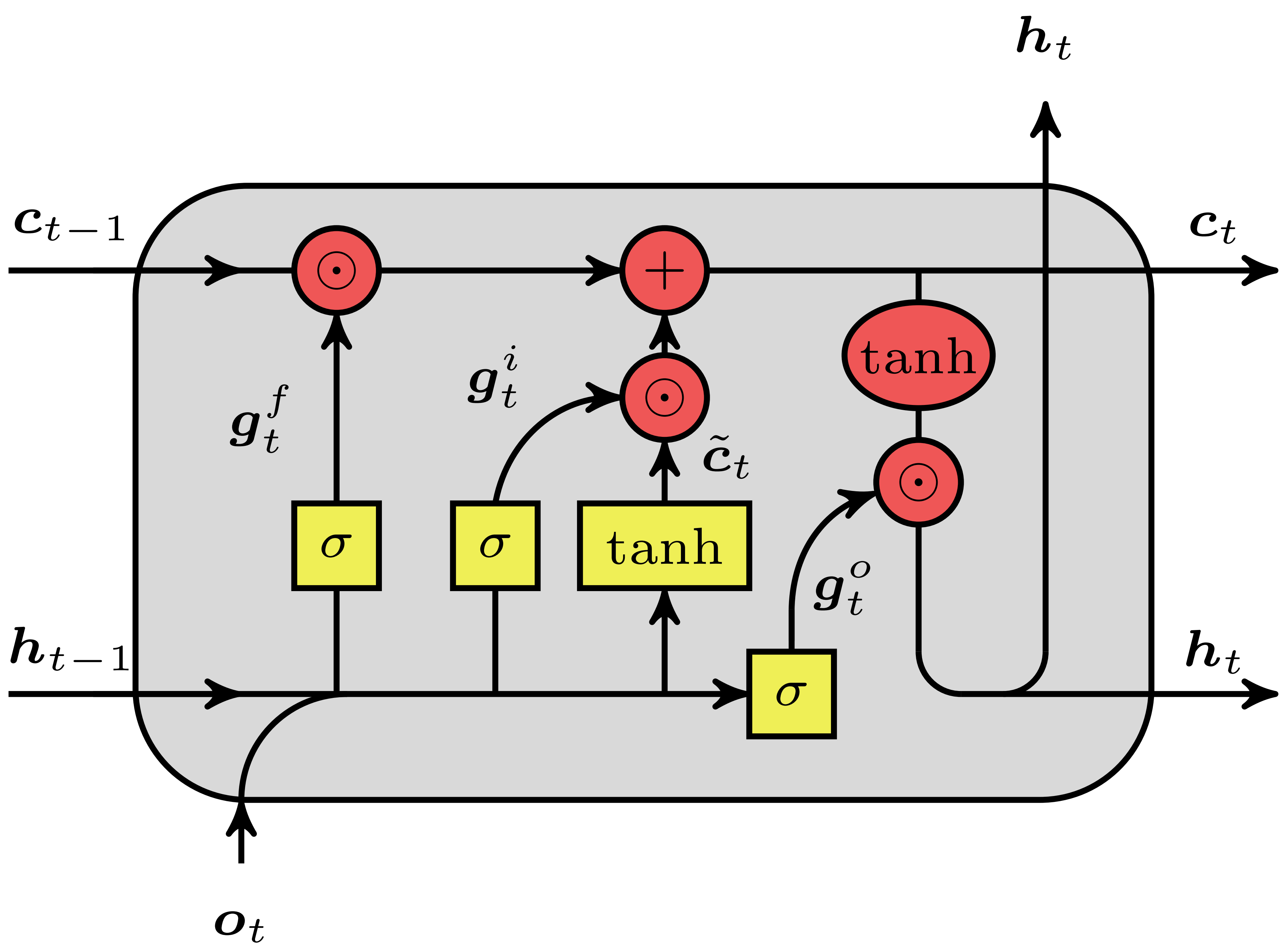
Hidden-to-output mapping

$$y_t = W_{oh}\, h_t \begin{cases} \hat{=}\ o_{t+1} \\ \textbf{or} \\ \hat{=}\ \dot{o}_t \end{cases}$$

Parameters (WEIGHTS) to be learned:

$$W_{ho}\,,\, W_{hh}\,,\, b_h\,,\, W_{oh}$$

Long Short-Term Memory (LSTM) S. Hochreiter and J. Schmidhuber (1997)



Long Short-Term Memory Cell

Kuramoto-Sivashinsky

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

Kuramoto-Sivashinsky

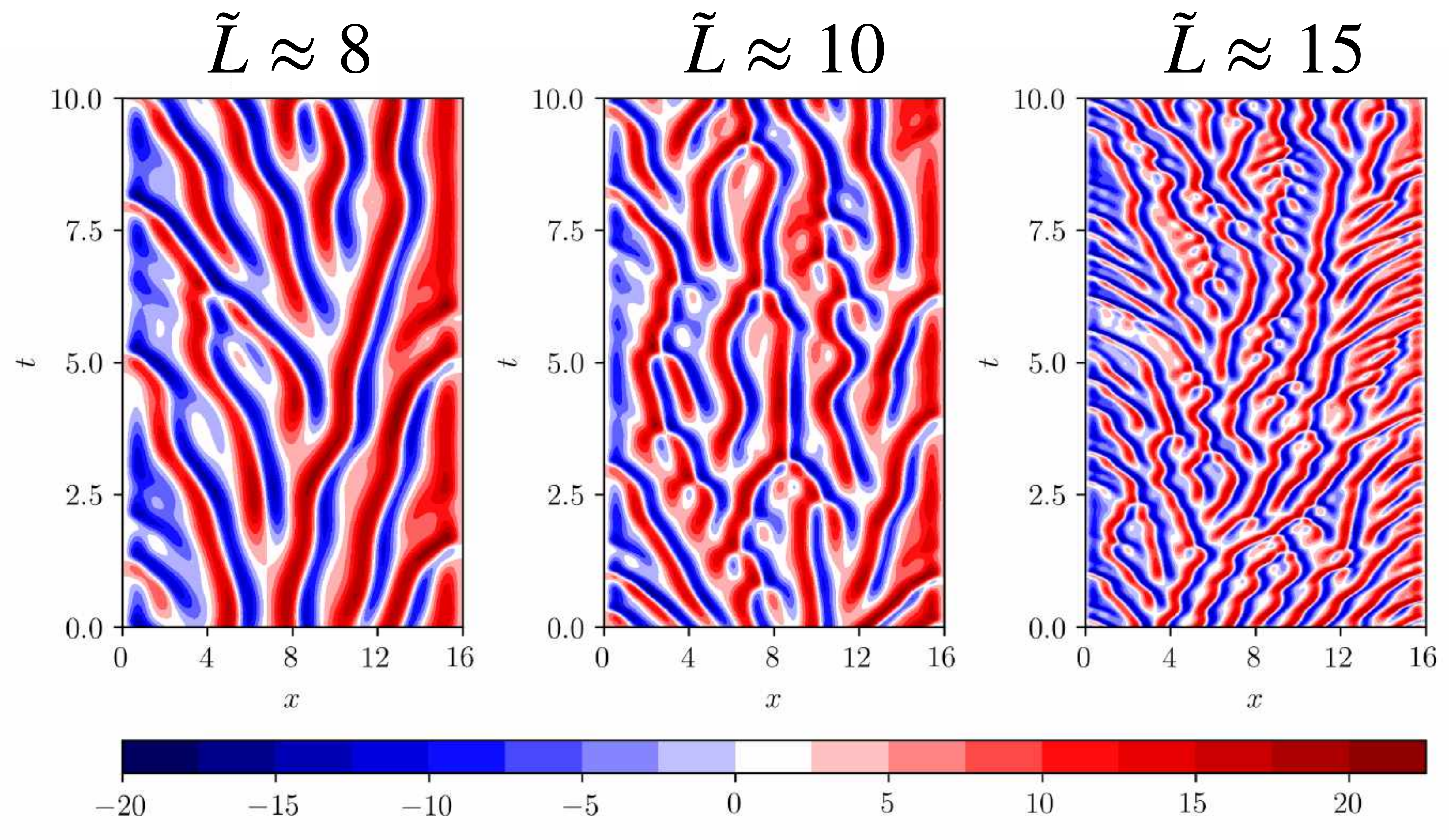
$$\frac{\partial u}{\partial t} = -\nu \frac{\partial^4 u}{\partial x^4} - \frac{\partial^2 u}{\partial x^2} - u \frac{\partial u}{\partial x}$$

- Fourth order PDE, negative viscocity ν
- Dirichlet & second order boundary conditions
- Domain $x \in [0, L]$, $L = 16$
- **Chaoticity scales with bifurcation parameter**

$$\tilde{L} = \frac{L}{2\pi\sqrt{\nu}}$$

Kuramoto-Sivashinsky

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

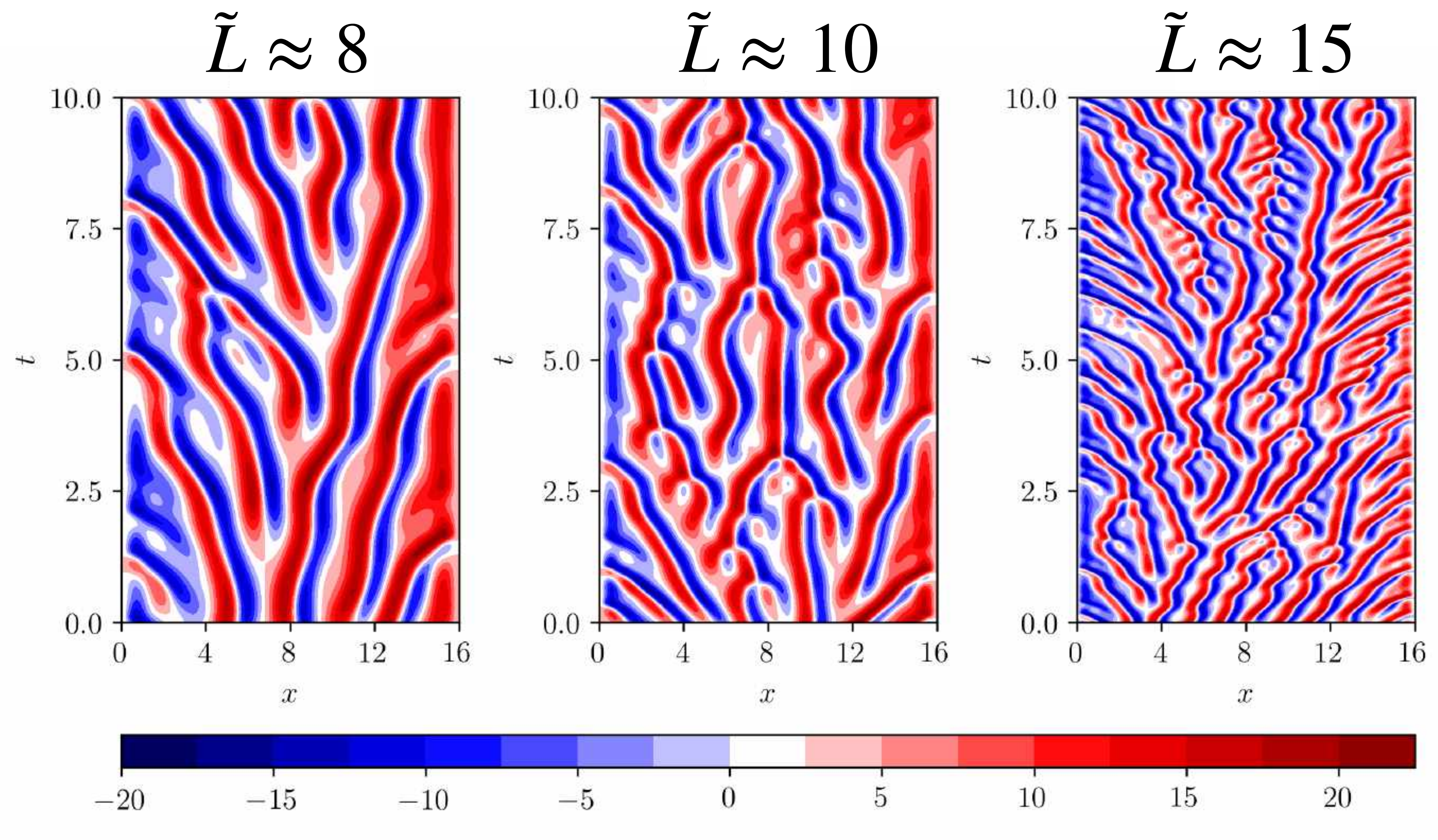


- Fourth order PDE, negative viscocity ν
- Dirichlet & second order boundary conditions
- Domain $x \in [0, L]$, $L = 16$
- **Chaoticity scales with bifurcation parameter**

$$\tilde{L} = \frac{L}{2\pi\sqrt{\nu}}$$

Kuramoto-Sivashinsky

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



- Fourth order PDE, negative viscocity ν
- Dirichlet & second order boundary conditions
- Domain $x \in [0, L]$, $L = 16$
- **Chaoticity scales with bifurcation parameter**

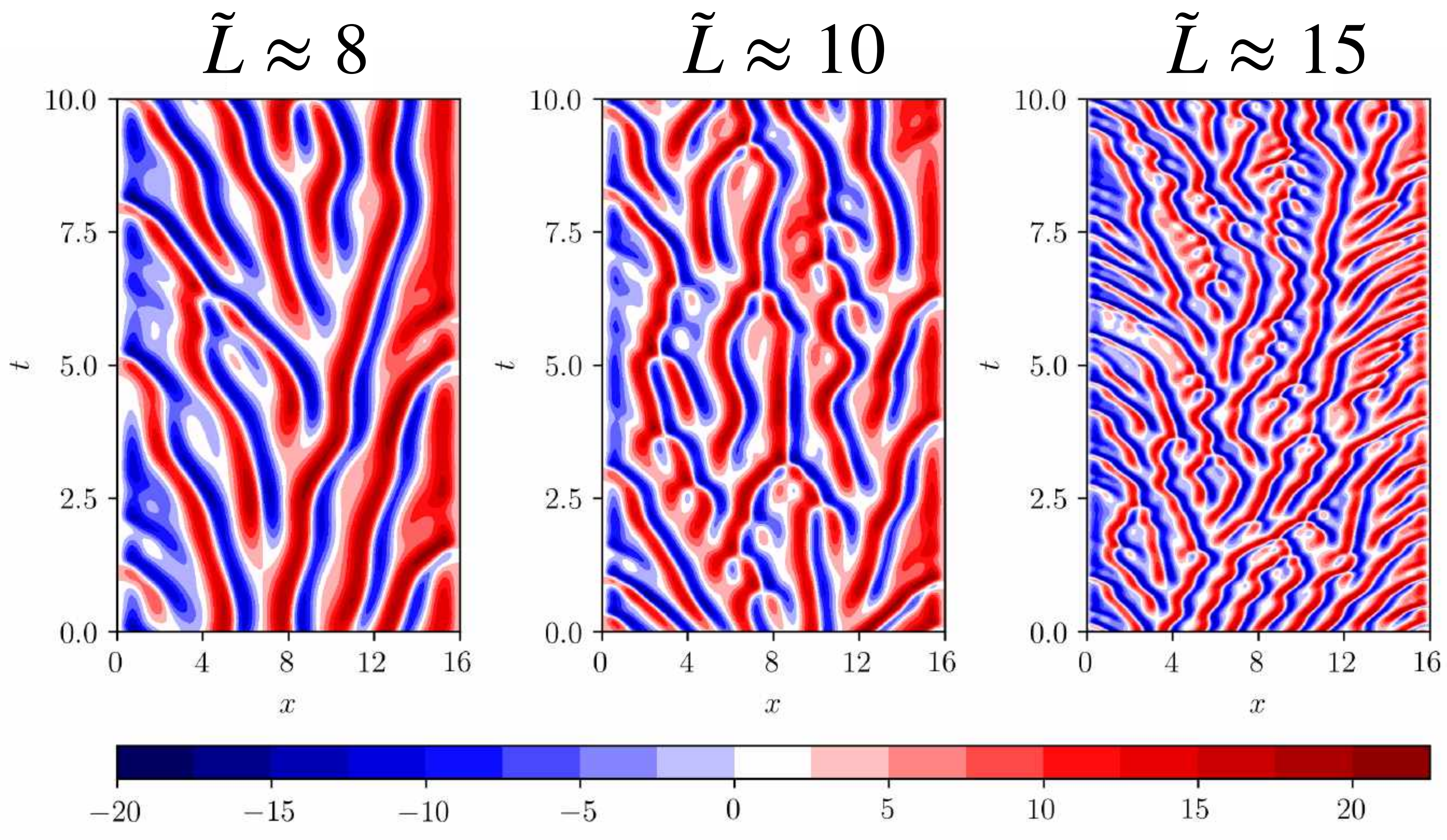
$$\tilde{L} = \frac{L}{2\pi\sqrt{\nu}}$$

Discretization with $d_u = 512$ gridpoints

$$d_u = \frac{L}{\Delta x}$$

Kuramoto-Sivashinsky

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



- Fourth order PDE, negative viscocity ν
- Dirichlet & second order boundary conditions
- Domain $x \in [0, L]$, $L = 16$
- **Chaoticity scales with bifurcation parameter**

$$\tilde{L} = \frac{L}{2\pi\sqrt{\nu}}$$

Discretization with $d_u = 512$ gridpoints

$$d_u = \frac{L}{\Delta x}$$

$$\frac{du_i}{dt} = -\nu \frac{u_{i-2} - 4u_{i-1} + 6u_i - 4u_{i+1} + u_{i+2}}{\Delta x^4} - \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} - \frac{u_{i+1}^2 - u_{i-1}^2}{4\Delta x}$$

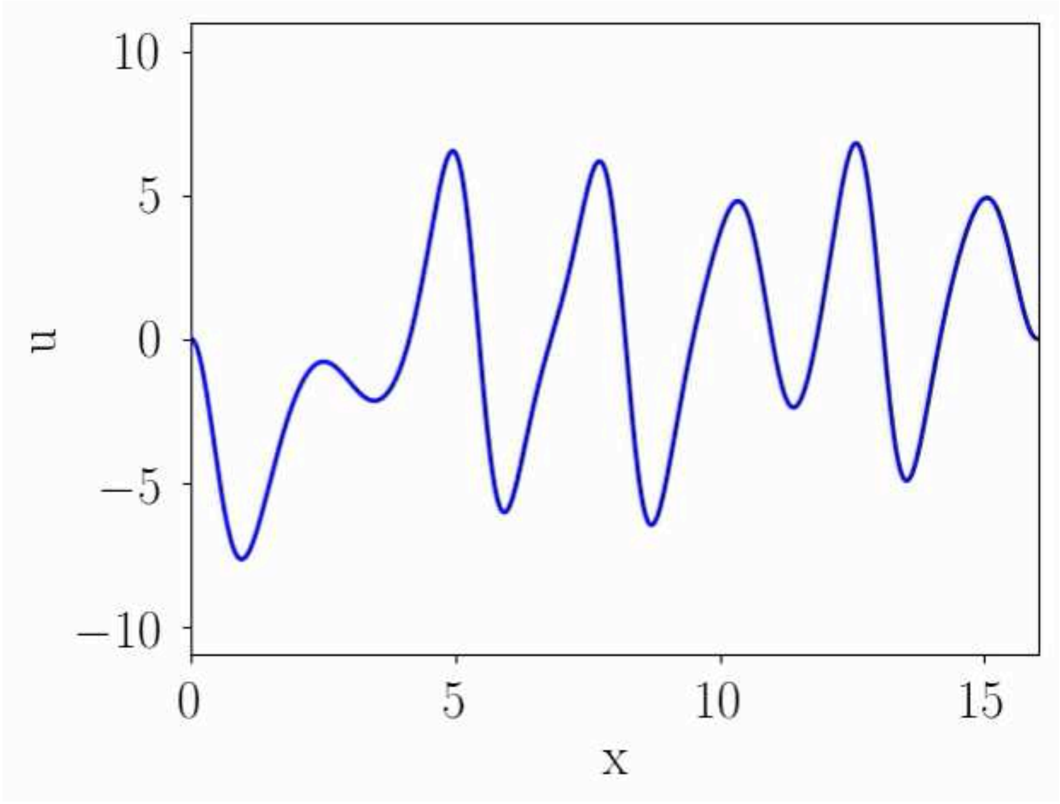
Integration with $dt = 0.02$ up to $T = 10^4$
(after discarding initial transients)
500.000 **samples**

Constructing the observable - training data

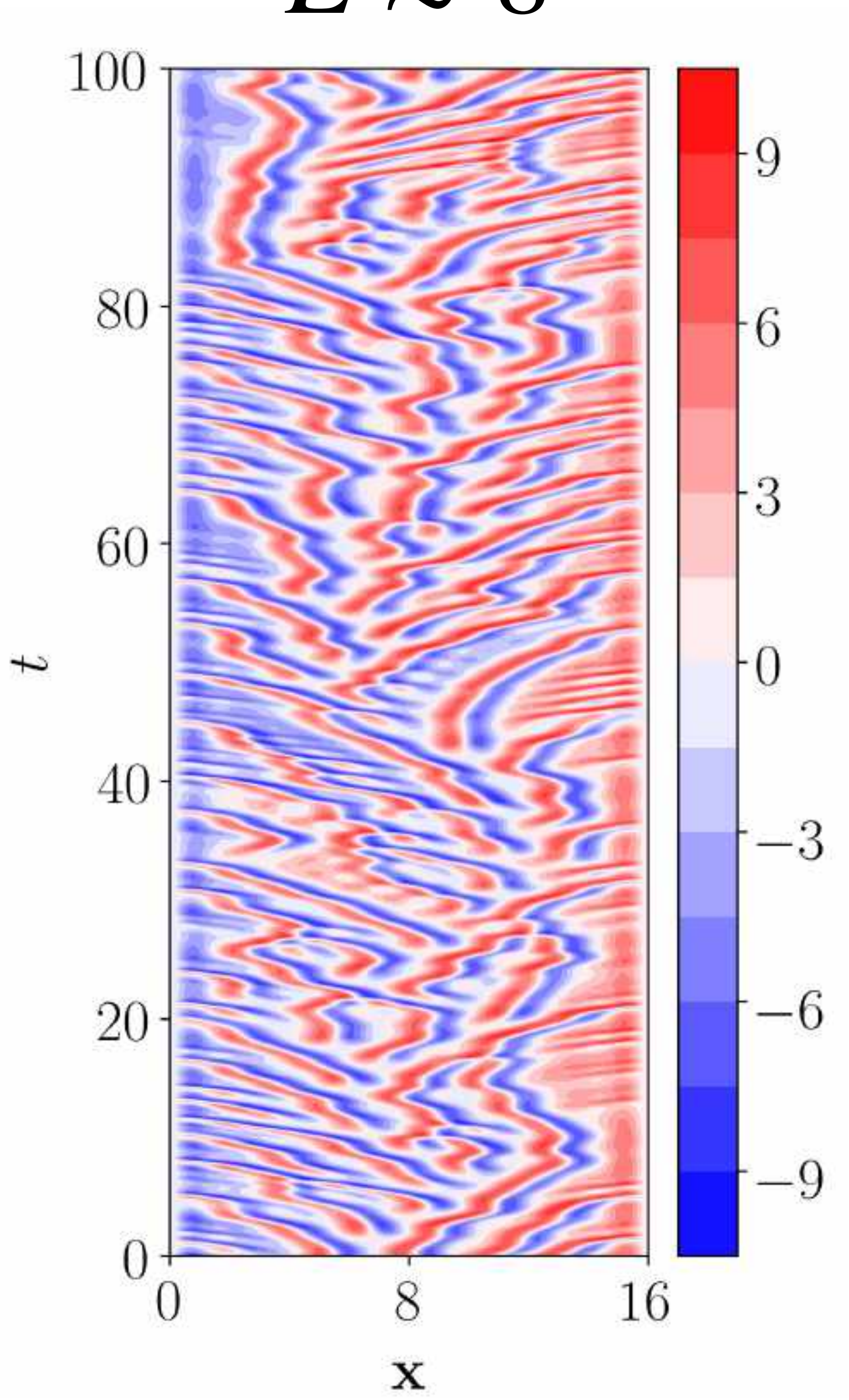
High dimensional

High dimensional simulation data

- $T = 10^4$
- half a million samples
- $u_t \in \mathbb{R}^{512}$



$\tilde{L} \approx 8$

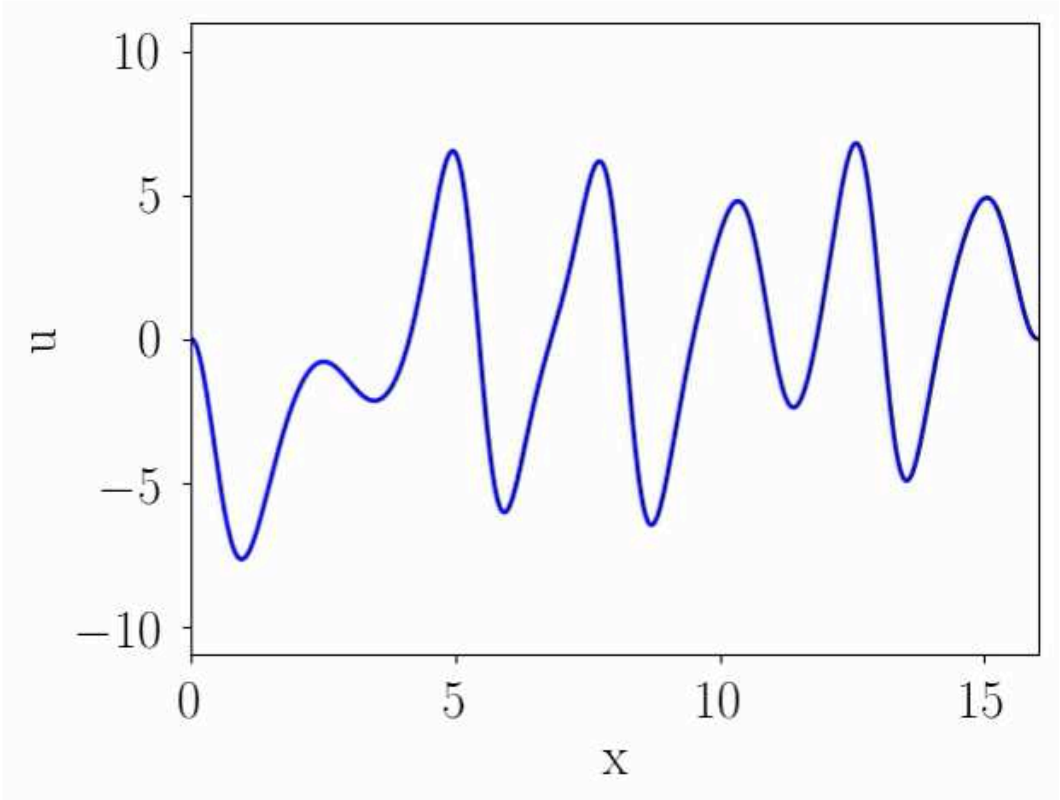


Constructing the observable - training data

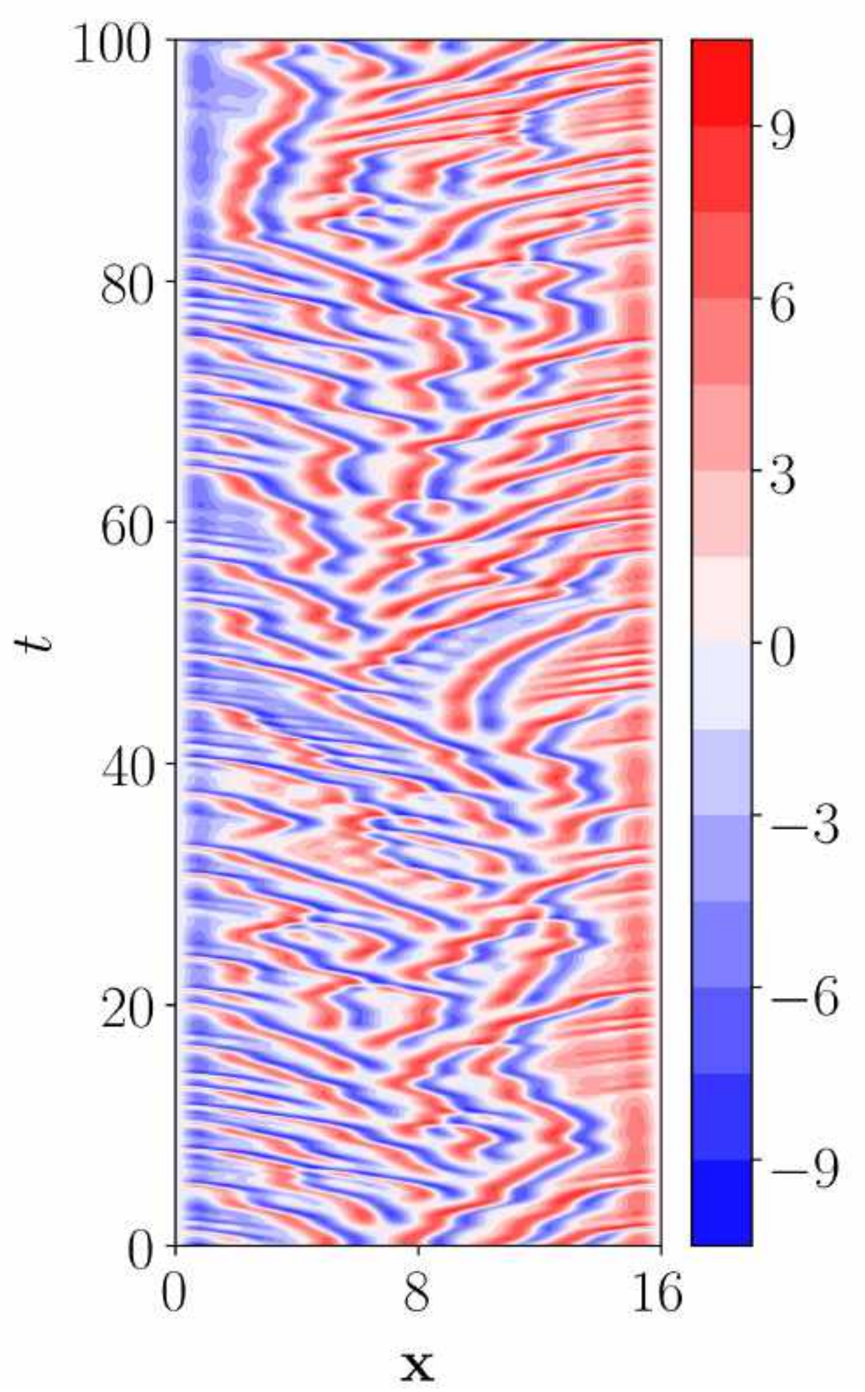
High dimensional

High dimensional simulation data

- $T = 10^4$
- half a million samples
- $u_t \in \mathbb{R}^{512}$



$\tilde{L} \approx 8$

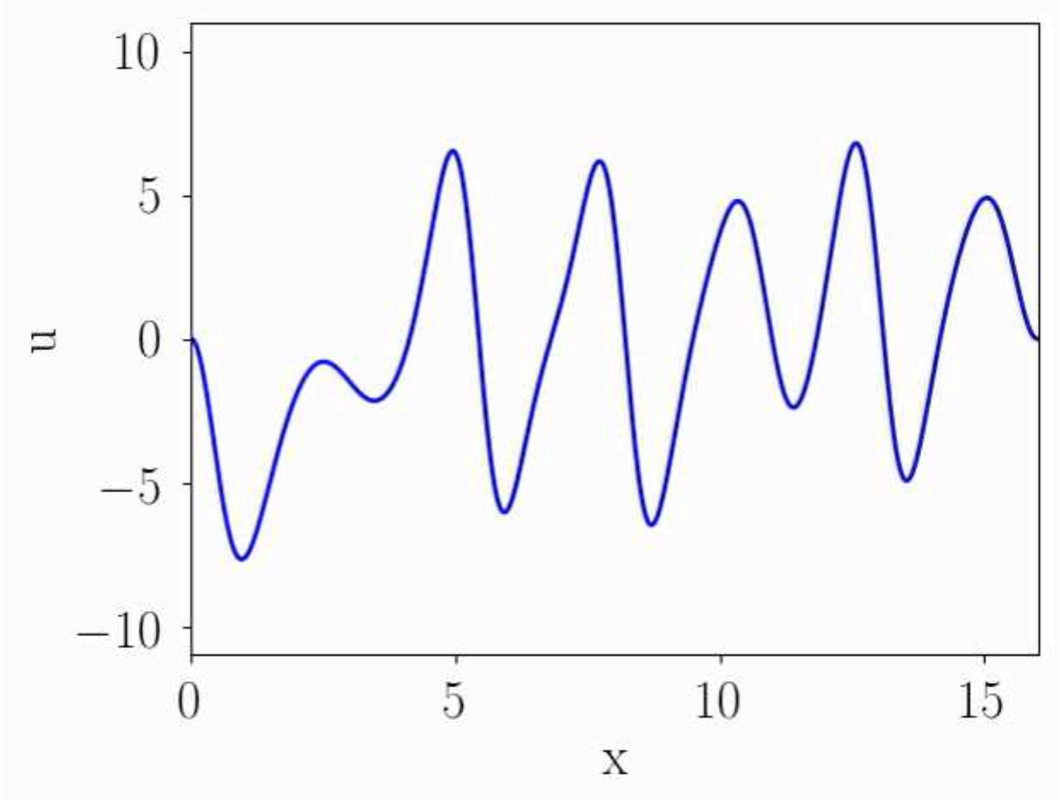


Constructing the observable - training data

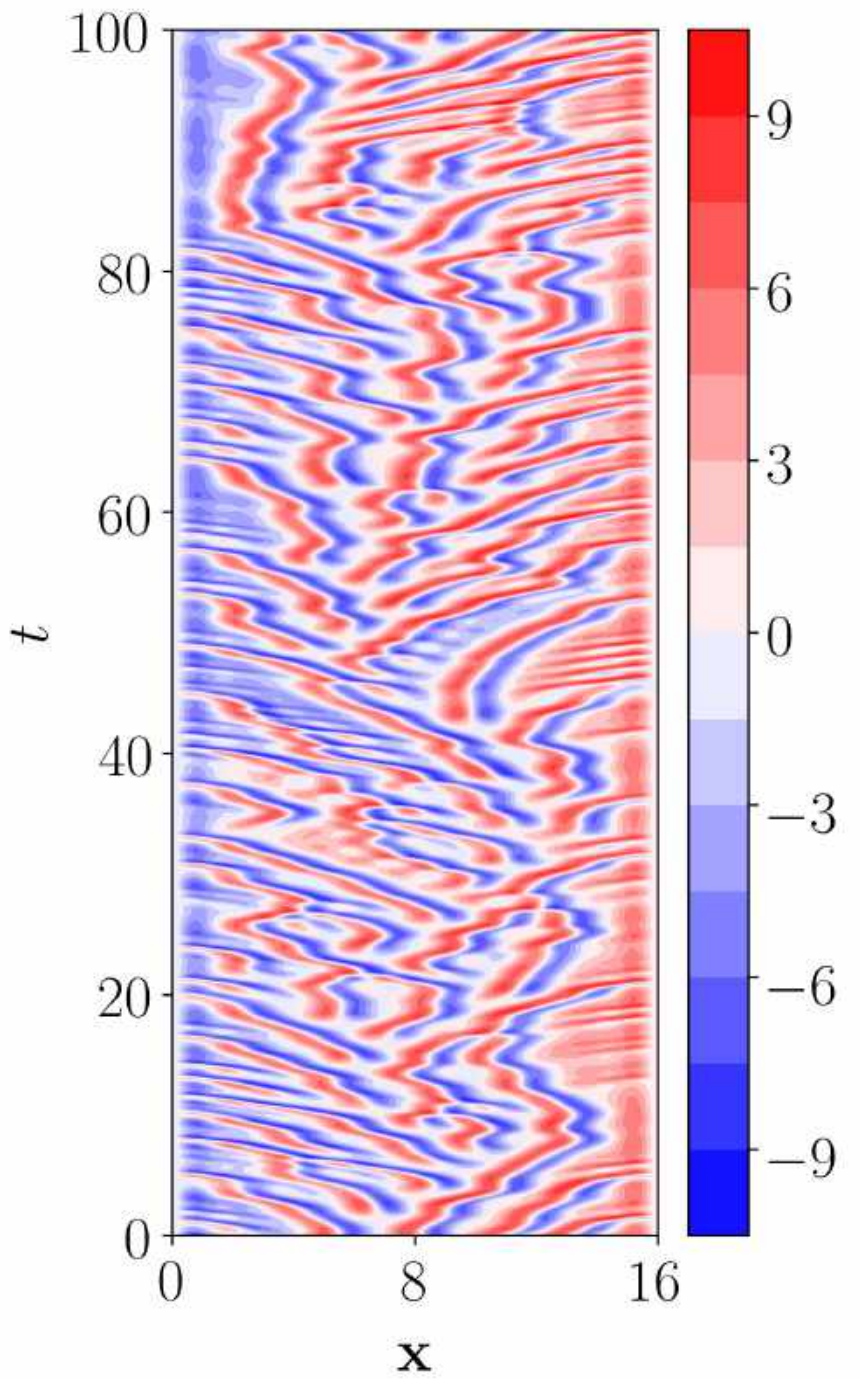
High dimensional

High dimensional simulation data

- $T = 10^4$
- half a million samples
- $u_t \in \mathbb{R}^{512}$

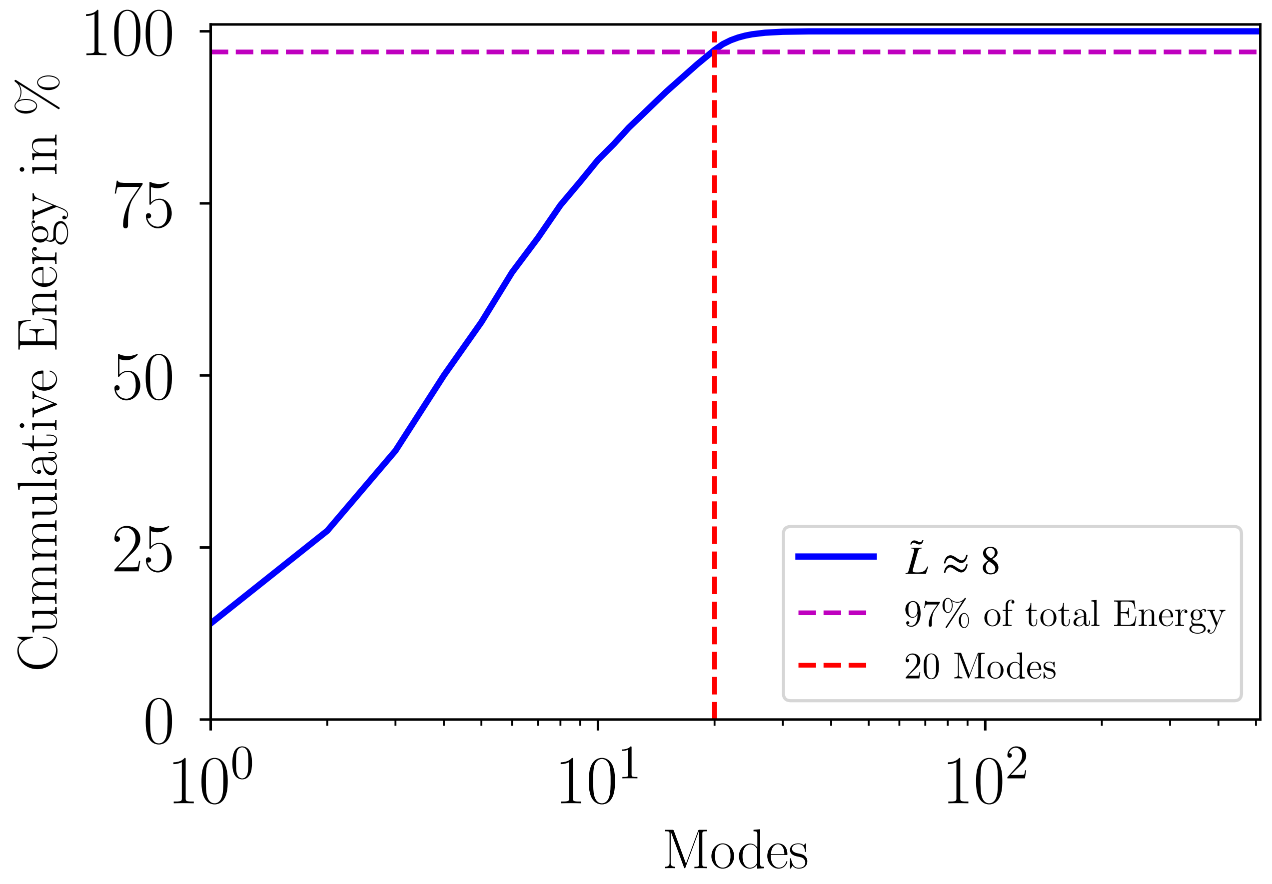


$\tilde{L} \approx 8$



SVD / PCA

Singular Value Decomposition

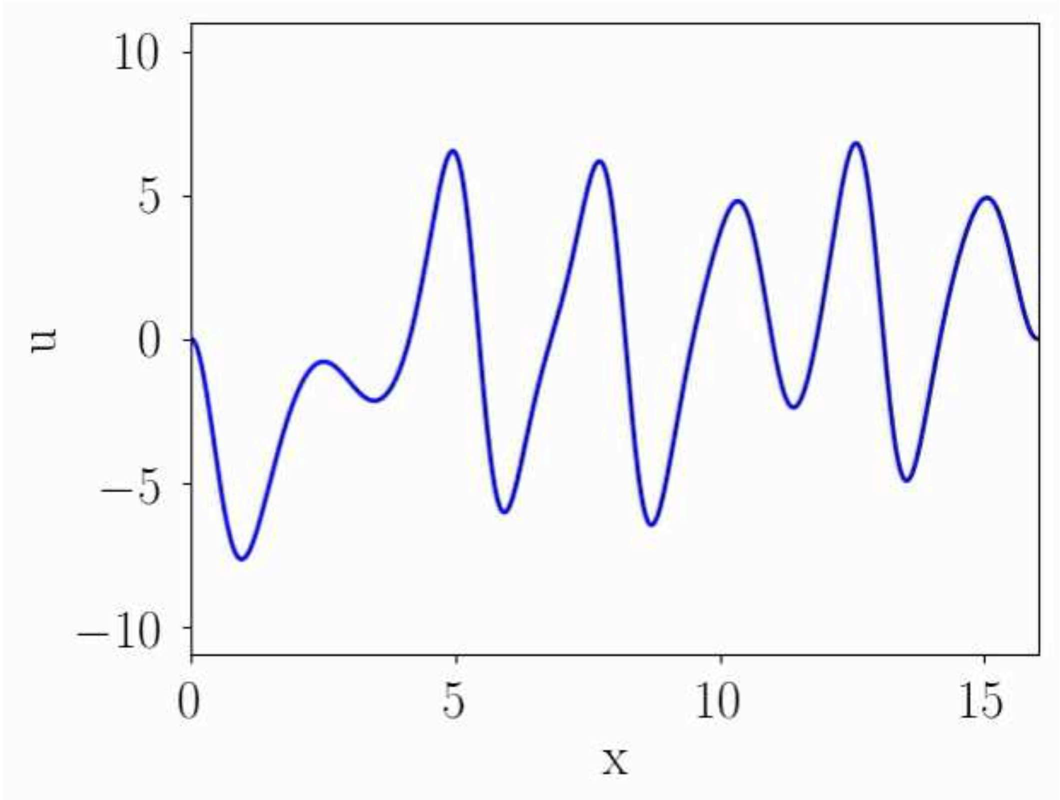


Constructing the observable - training data

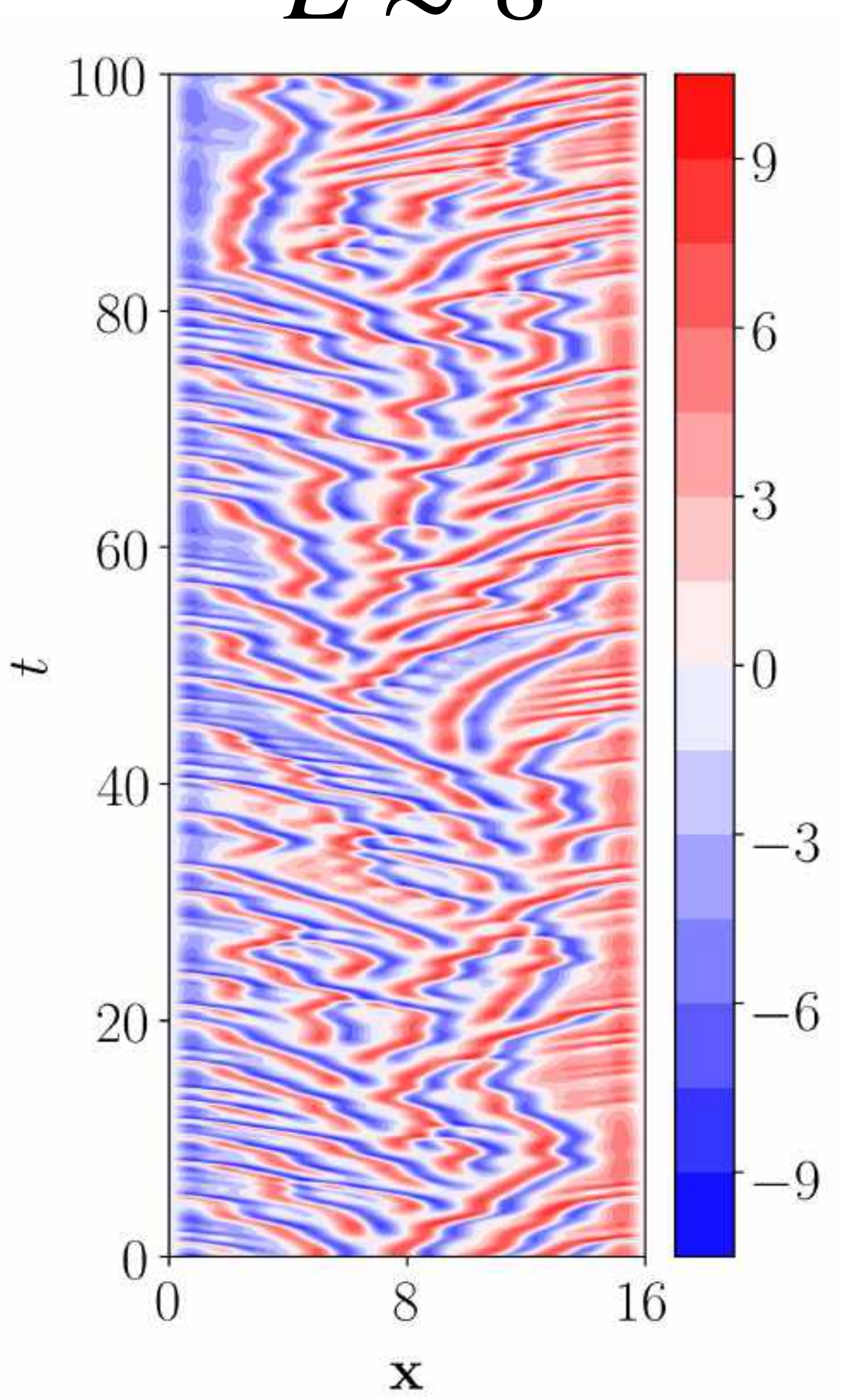
High dimensional

High dimensional simulation data

- $T = 10^4$
- half a million samples
- $u_t \in \mathbb{R}^{512}$



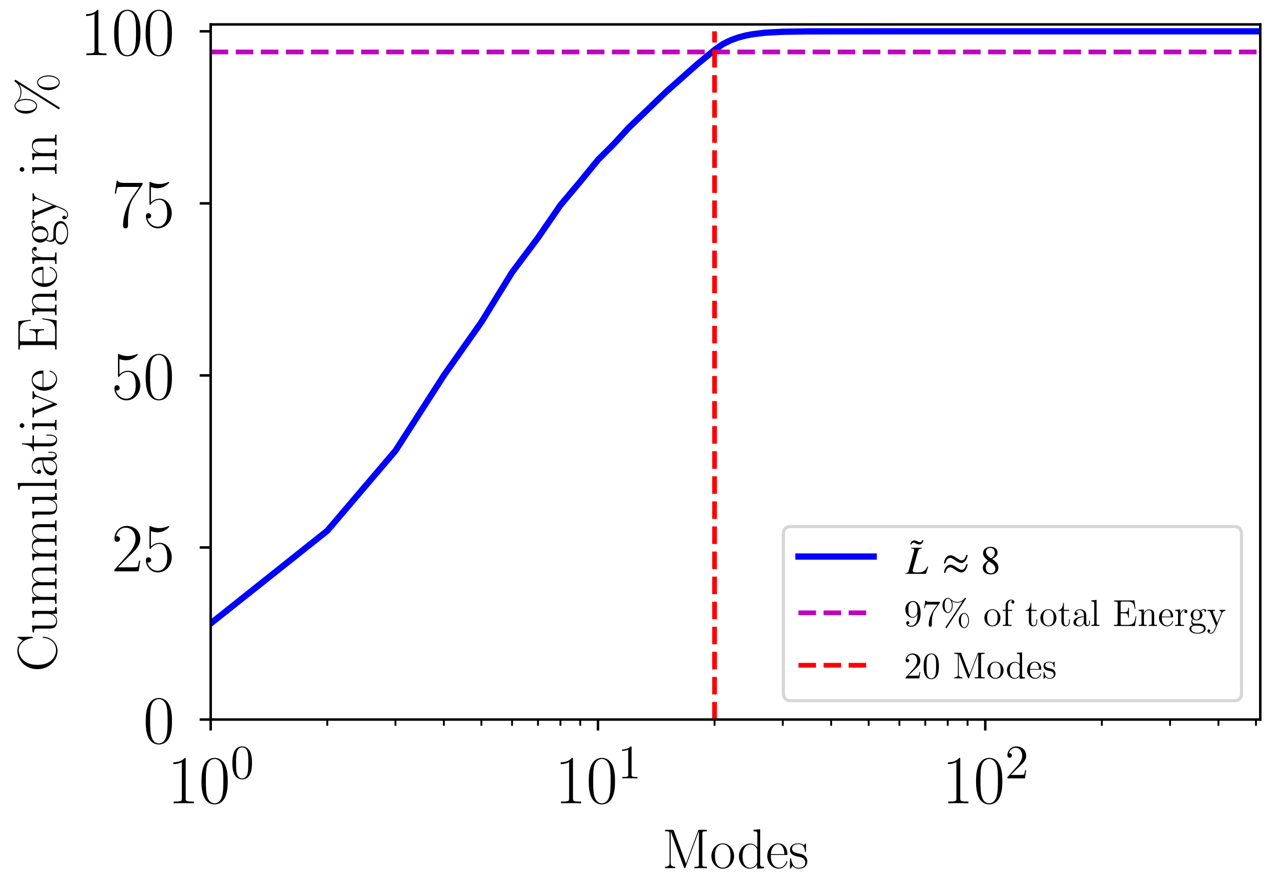
$\tilde{L} \approx 8$



SVD / PCA

Singular Value Decomposition

Throw away modes with low energy

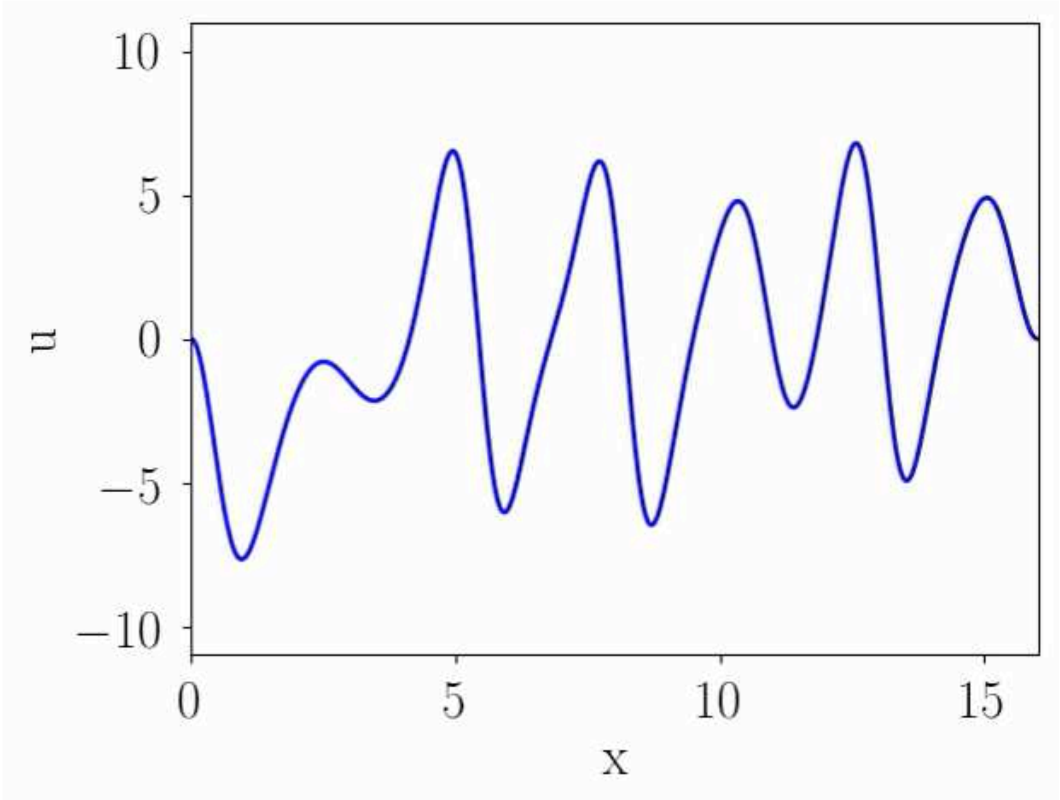


Constructing the observable - training data

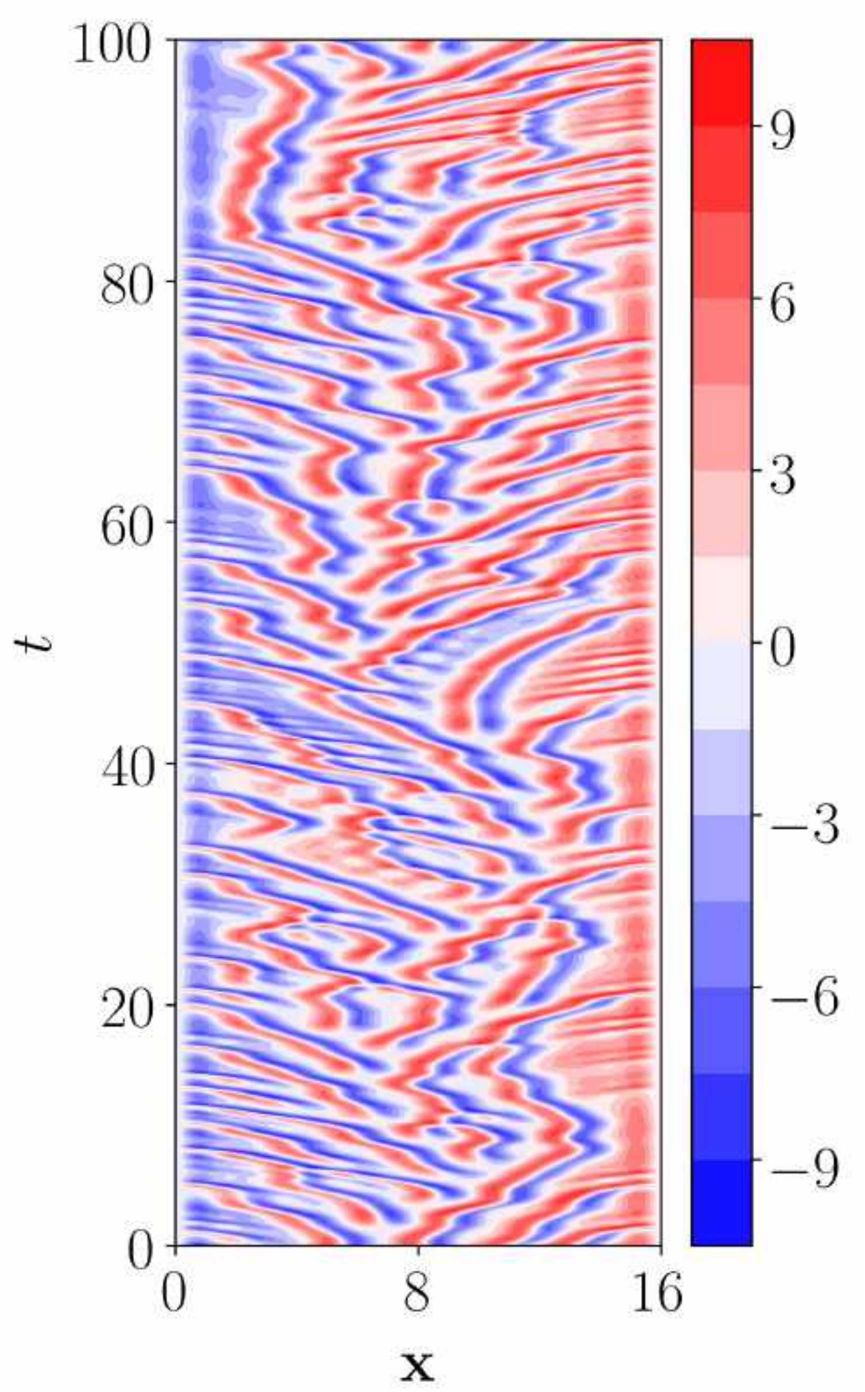
High dimensional

High dimensional simulation data

- $T = 10^4$
- half a million samples
- $u_t \in \mathbb{R}^{512}$



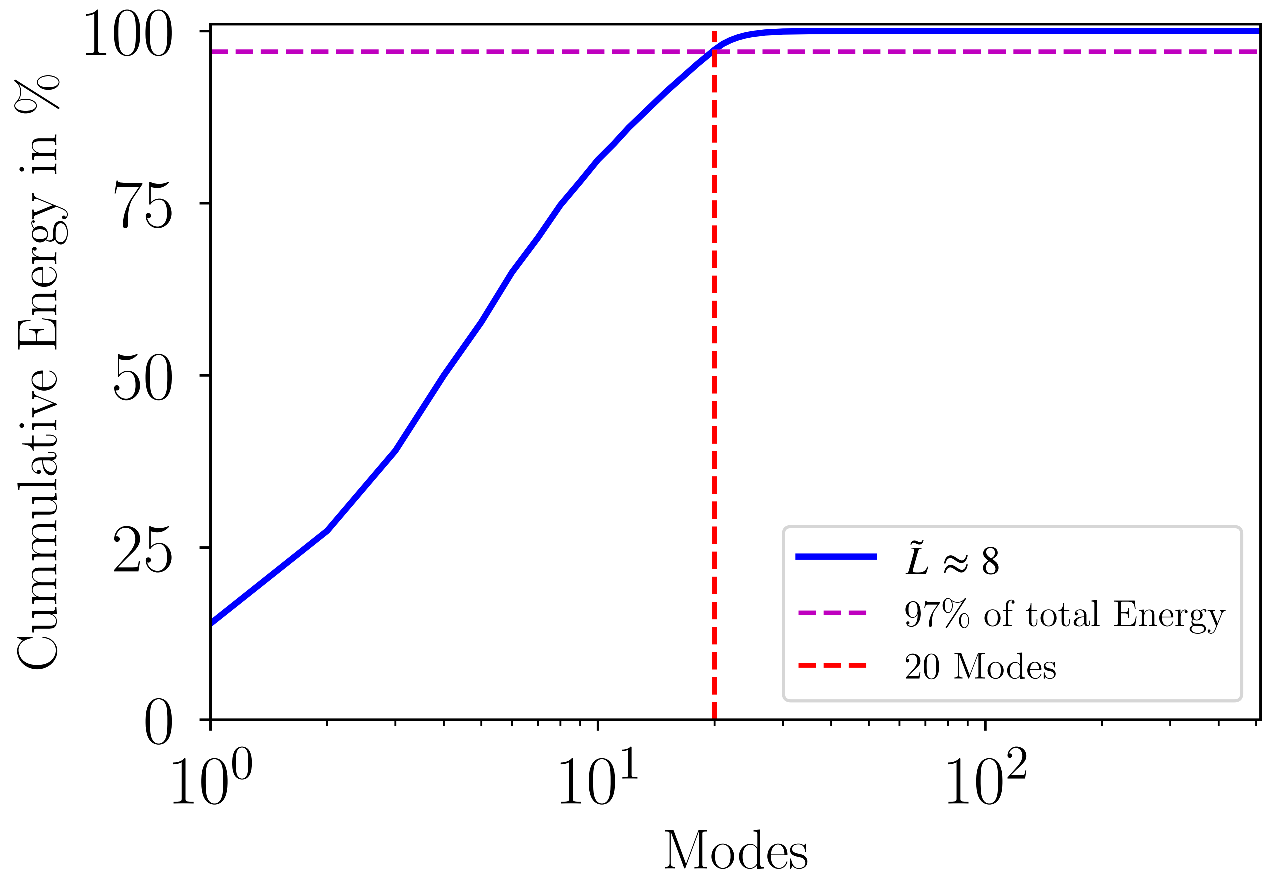
$\tilde{L} \approx 8$

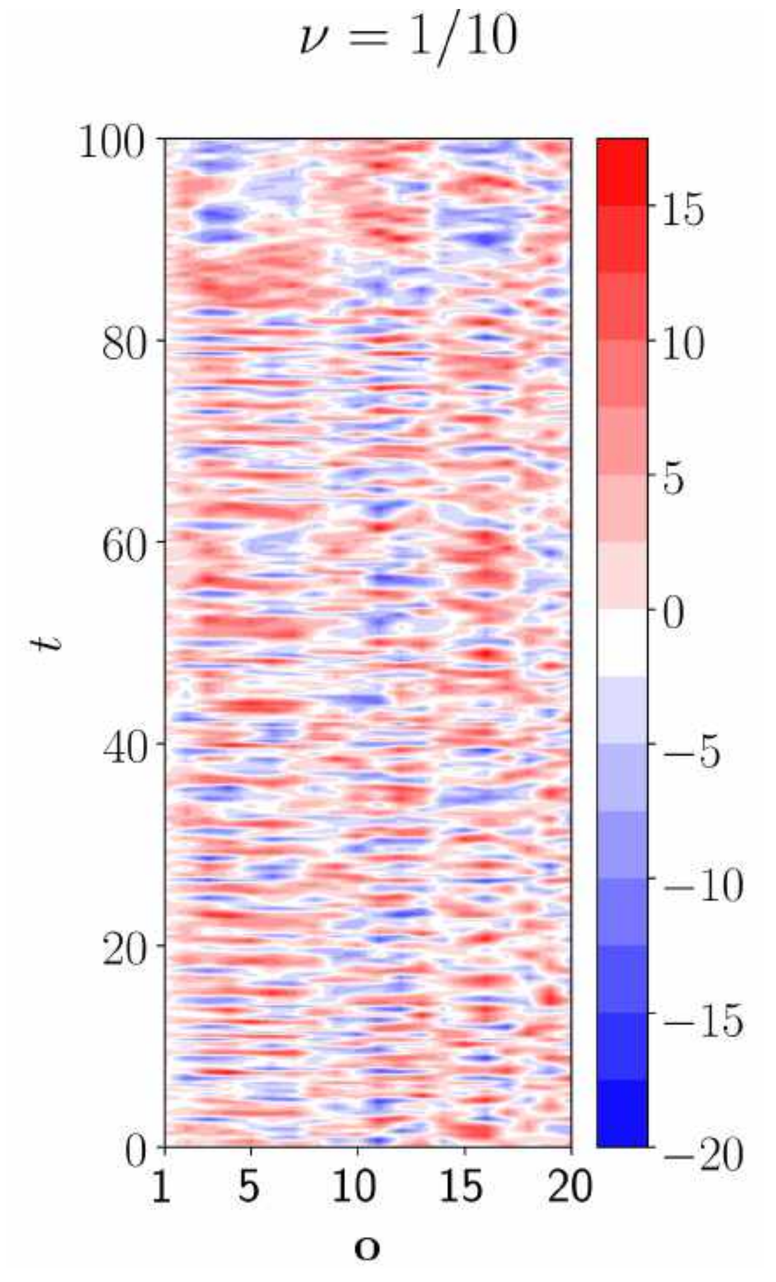


SVD / PCA

Singular Value Decomposition

Throw away modes with low energy





20 Modes (observable)

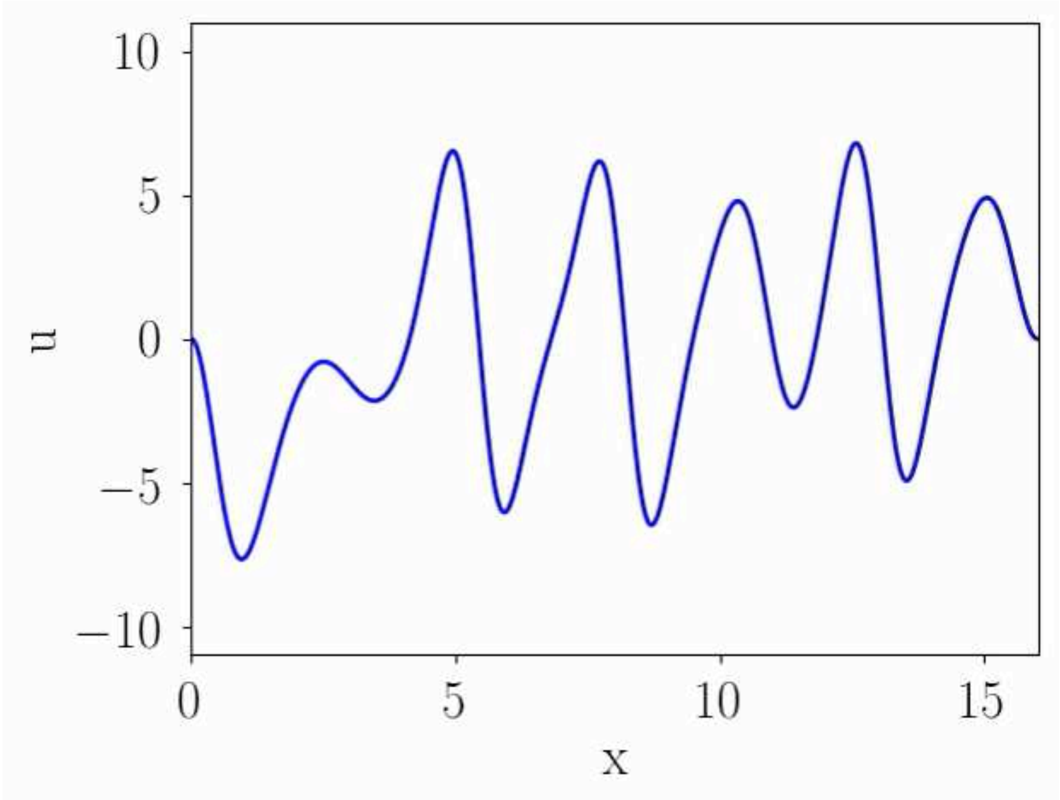
$o_t \in \mathbb{R}^{20}$

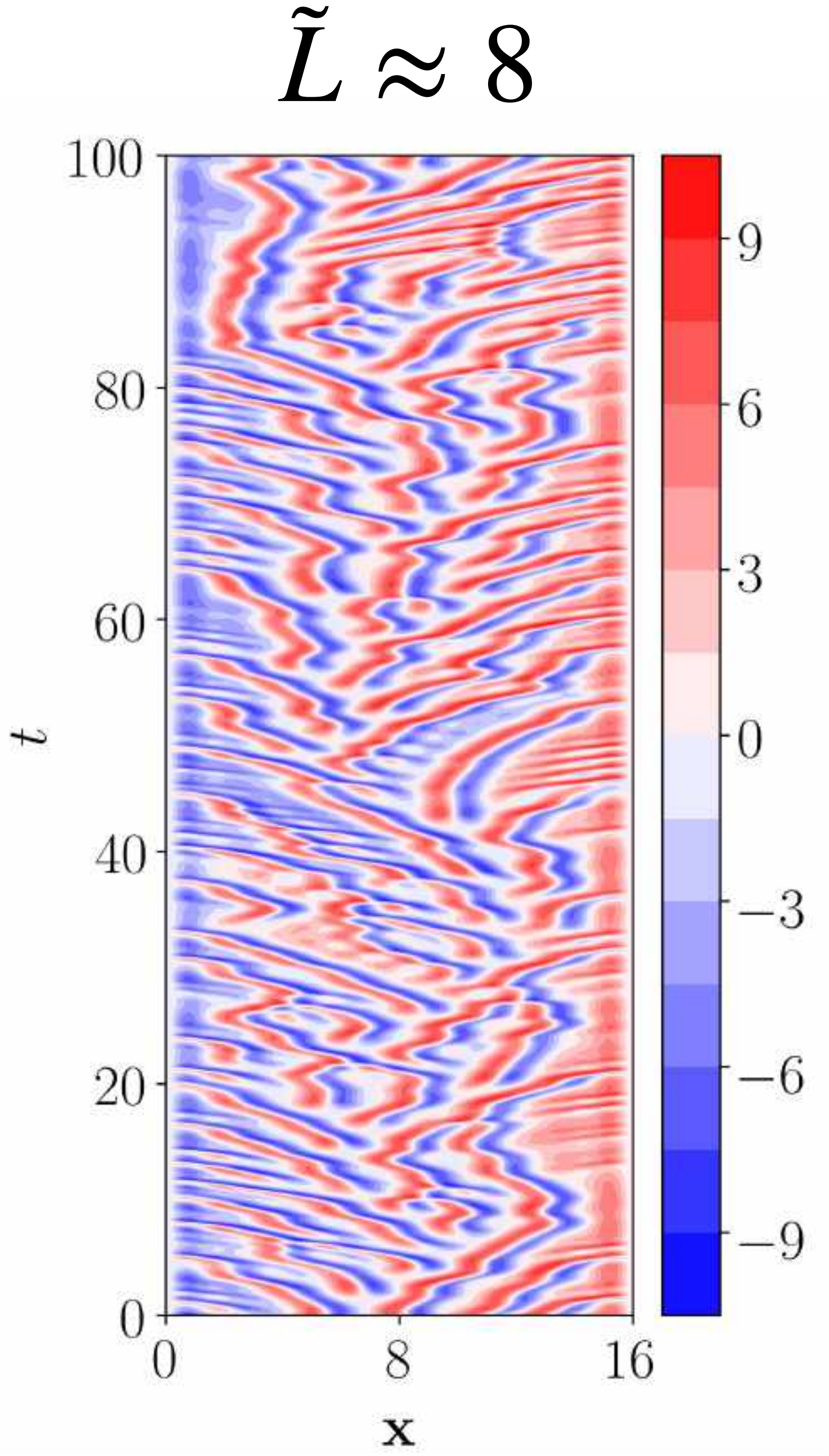
Constructing the observable - training data

High dimensional

High dimensional simulation data

- $T = 10^4$
- half a million samples
- $u_t \in \mathbb{R}^{512}$

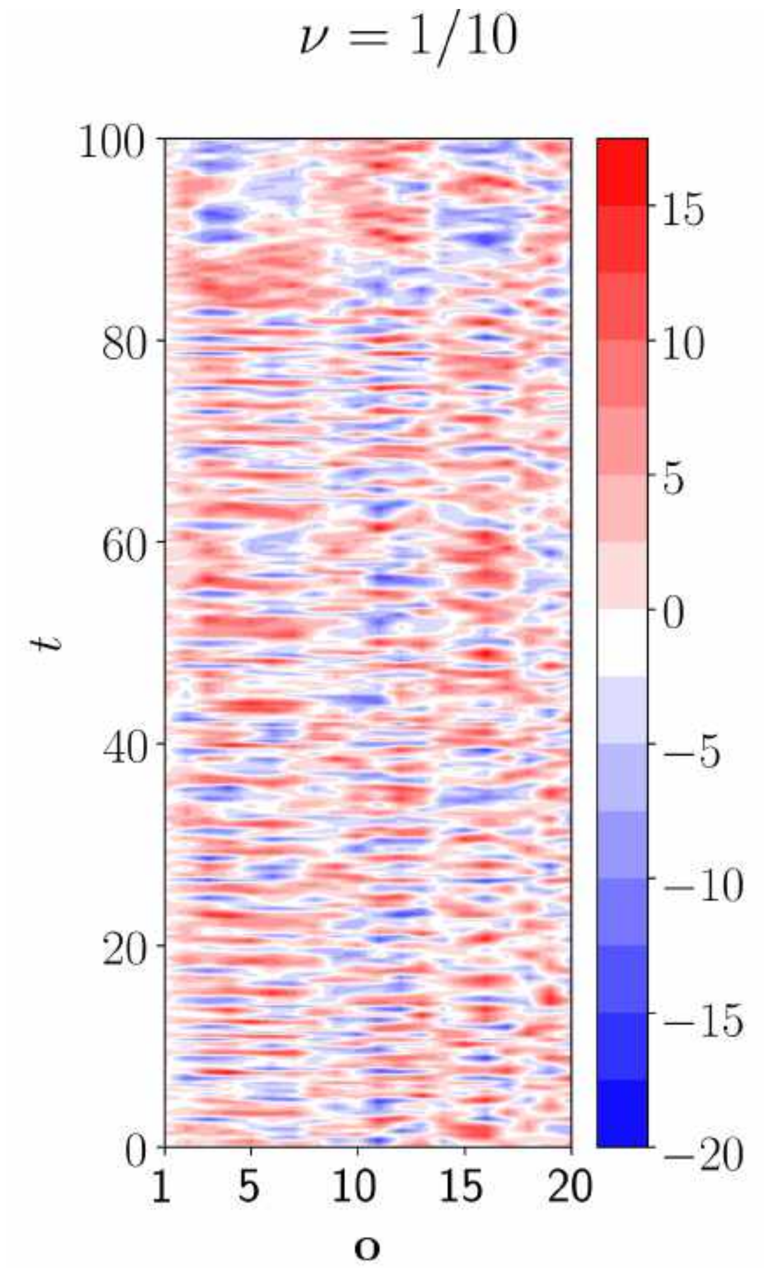




SVD / PCA

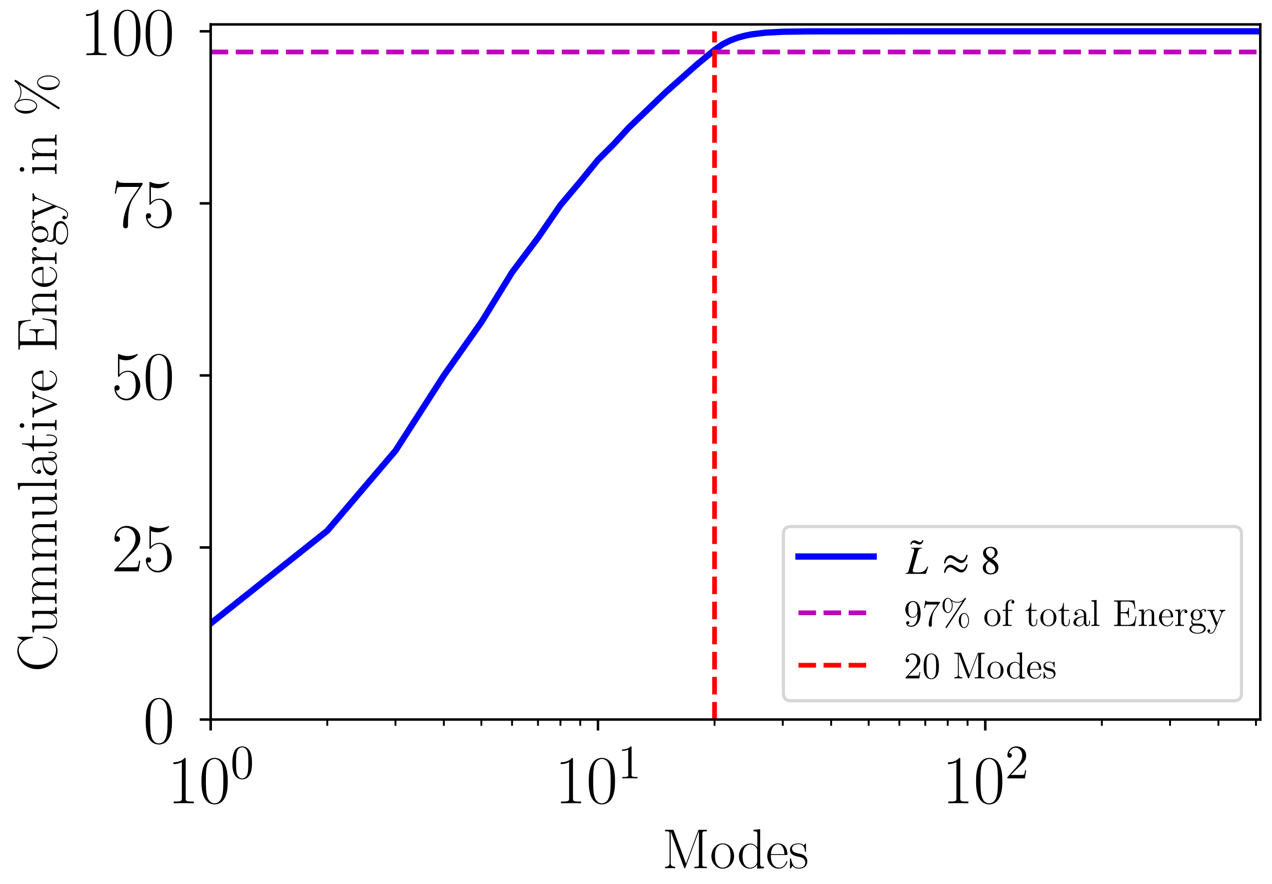
Singular Value Decomposition

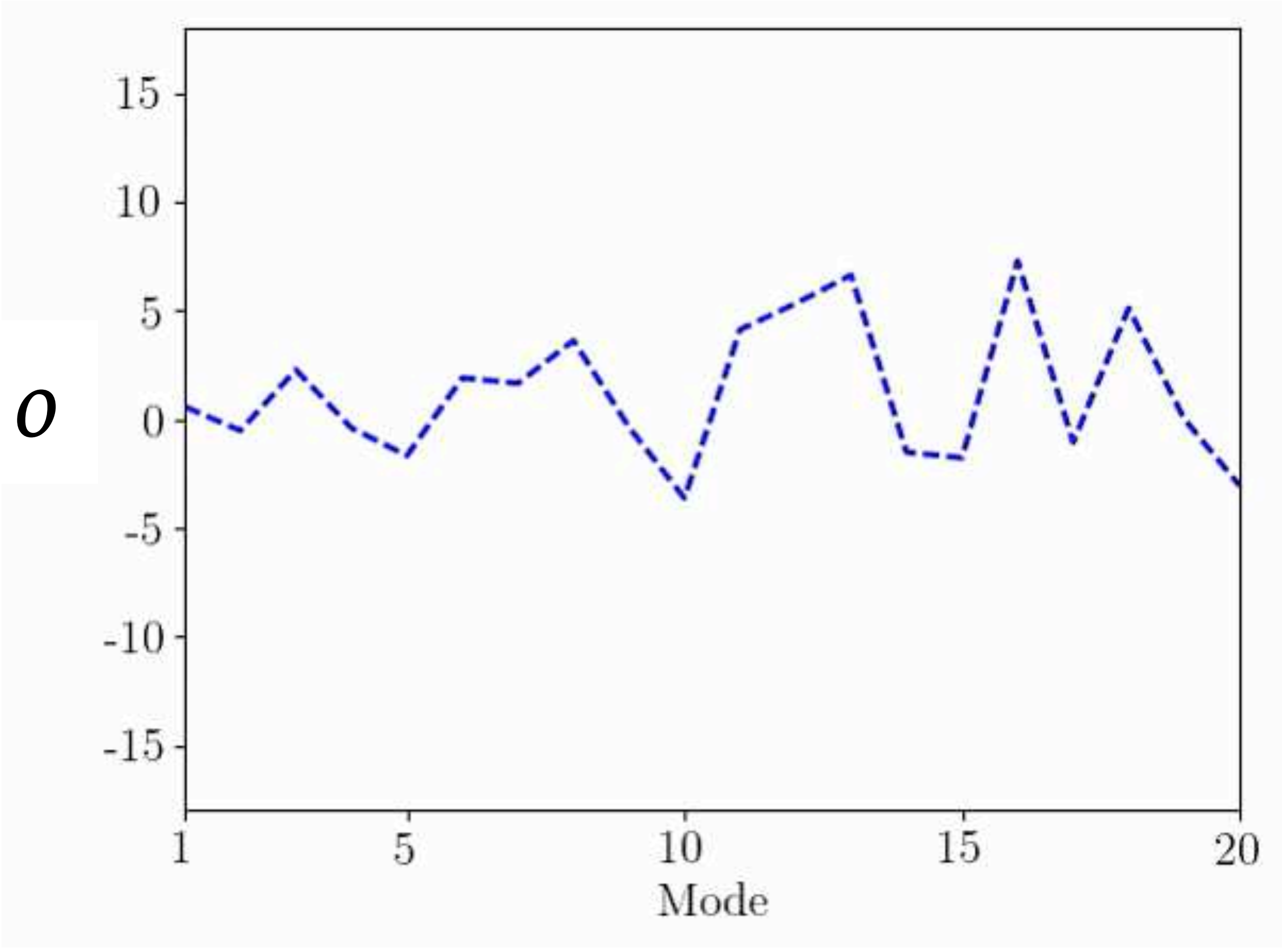
Throw away modes with low energy



20 Modes (observable)

$o_t \in \mathbb{R}^{20}$





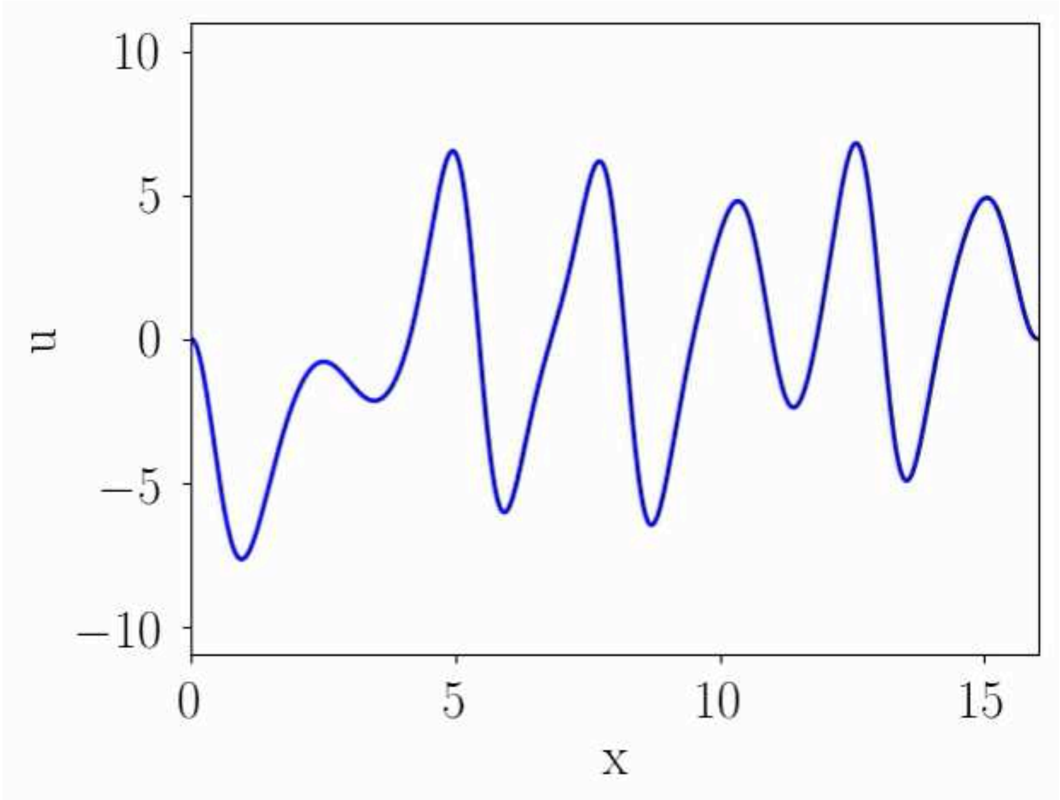
*Low dimensional **reduced-order** state (most energetic modes)*

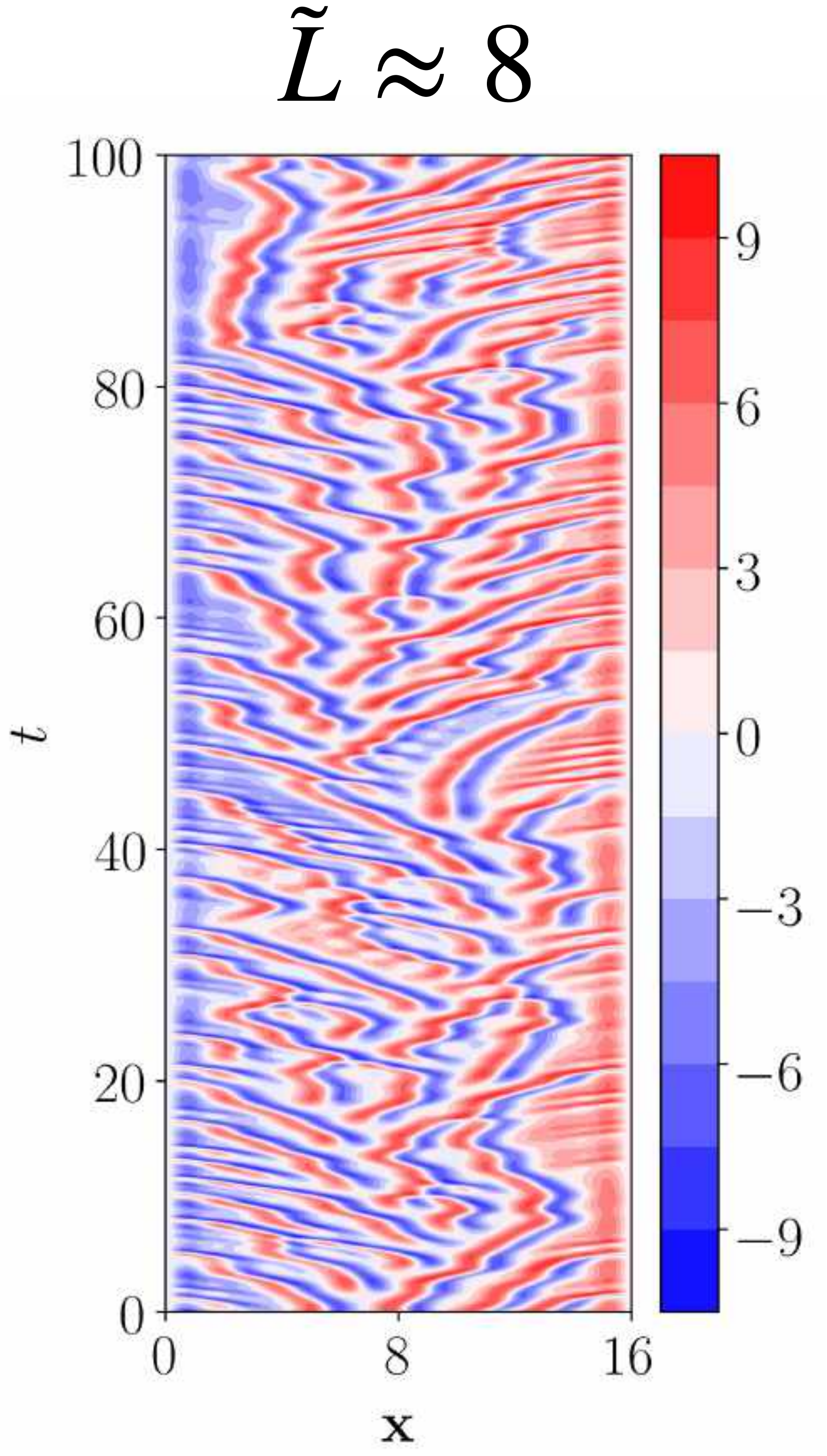
Constructing the observable - training data

High dimensional

High dimensional simulation data

- $T = 10^4$
- half a million samples
- $u_t \in \mathbb{R}^{512}$

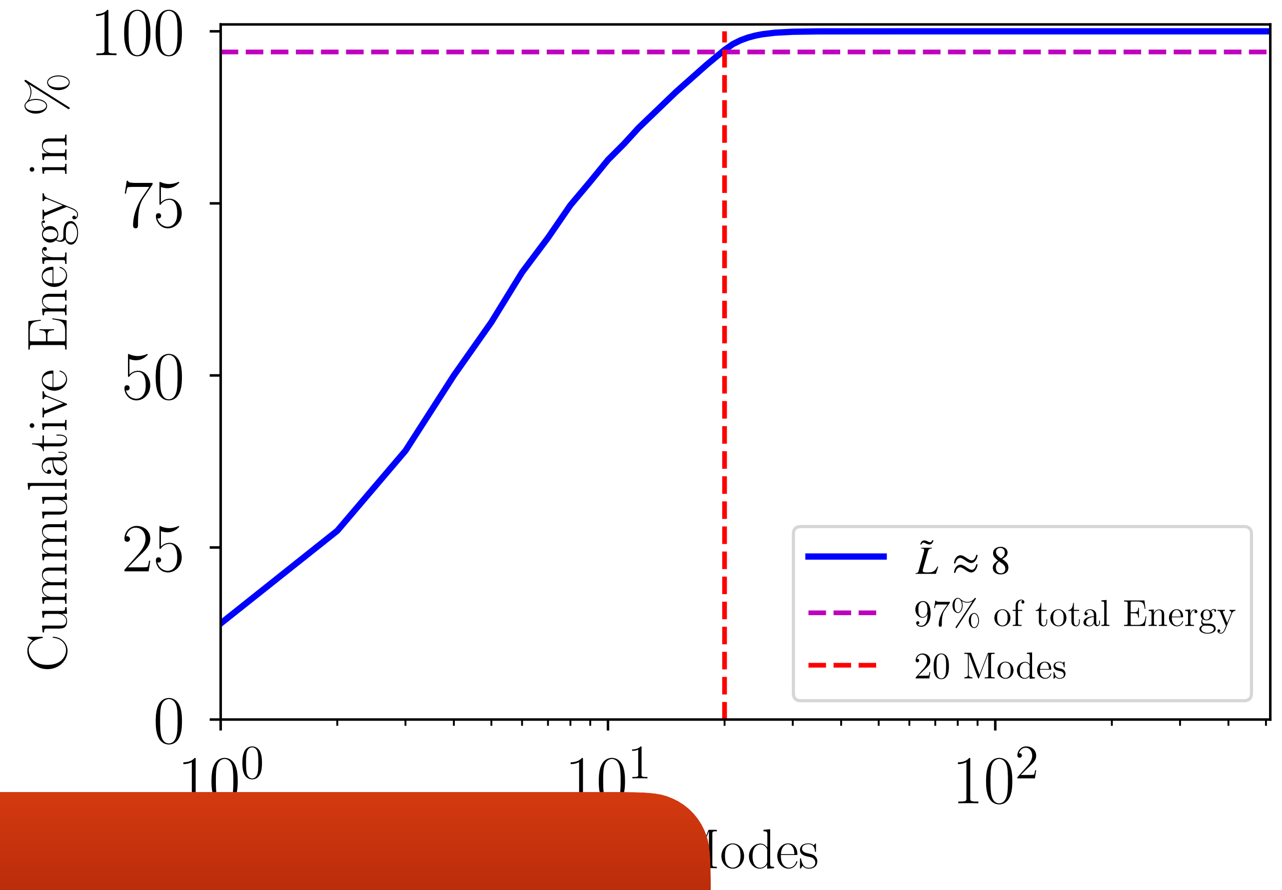


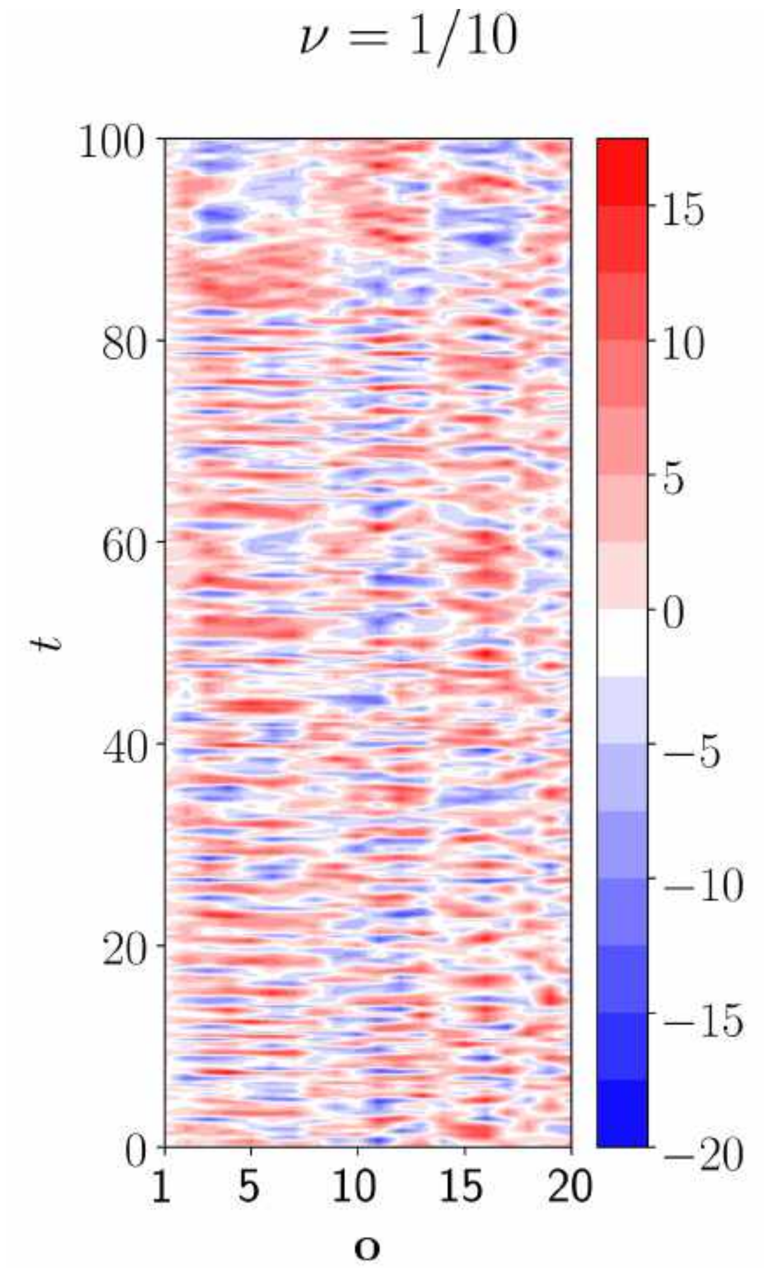


SVD / PCA

Singular Value Decomposition

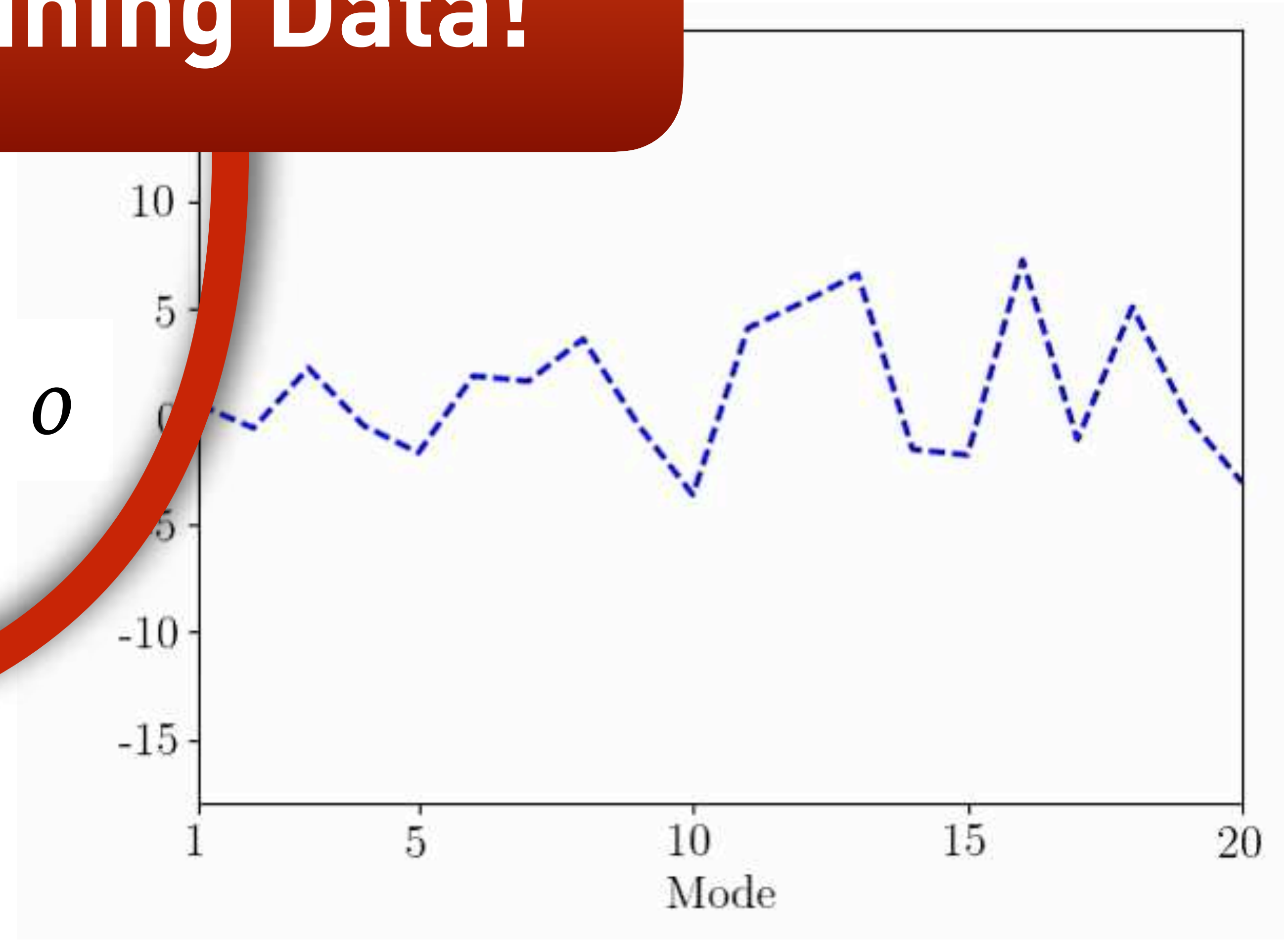
Throw away modes with low energy





20 Modes (observable)
$o_t \in \mathbb{R}^{20}$

Training Data!



*Low dimensional **reduced-order** state (most energetic modes)*

Forecasting on UNSEEN data - Iterative prediction in practice

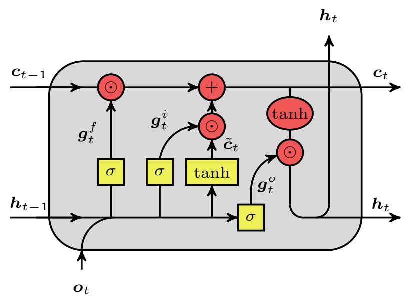
Forecasting on UNSEEN data - Iterative prediction in practice

data

Forecasting on UNSEEN data - Iterative prediction in practice

data

&

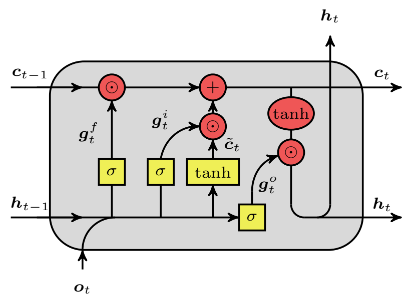


model

Forecasting on UNSEEN data - Iterative prediction in practice

data

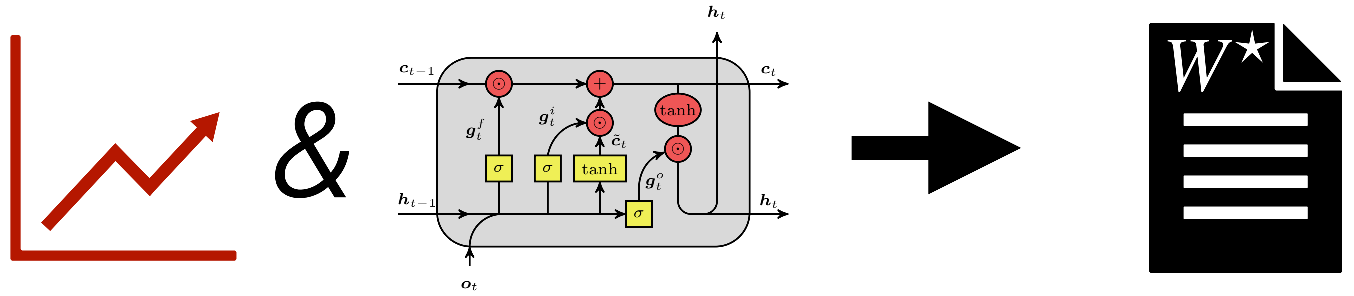
&



model

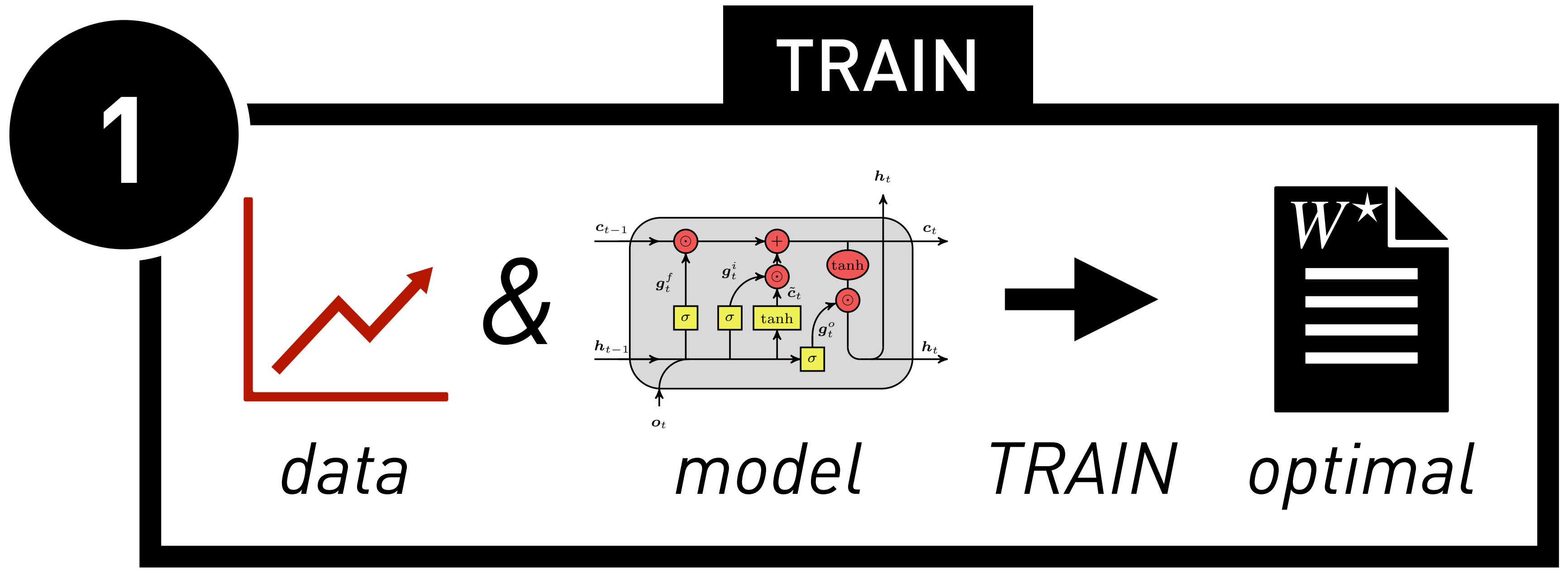
TRAIN

Forecasting on UNSEEN data - Iterative prediction in practice



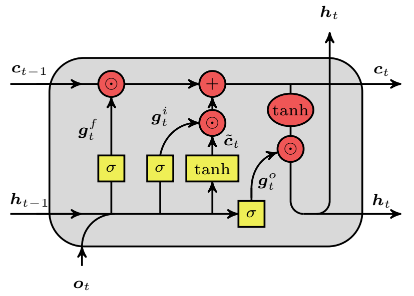
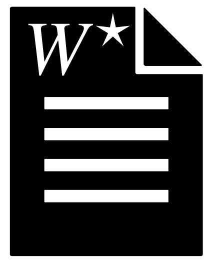
data *model* *TRAIN* *optimal*

Forecasting on UNSEEN data - Iterative prediction in practice



Forecasting on UNSEEN data - Iterative prediction in practice

1 **TRAIN**


&

→


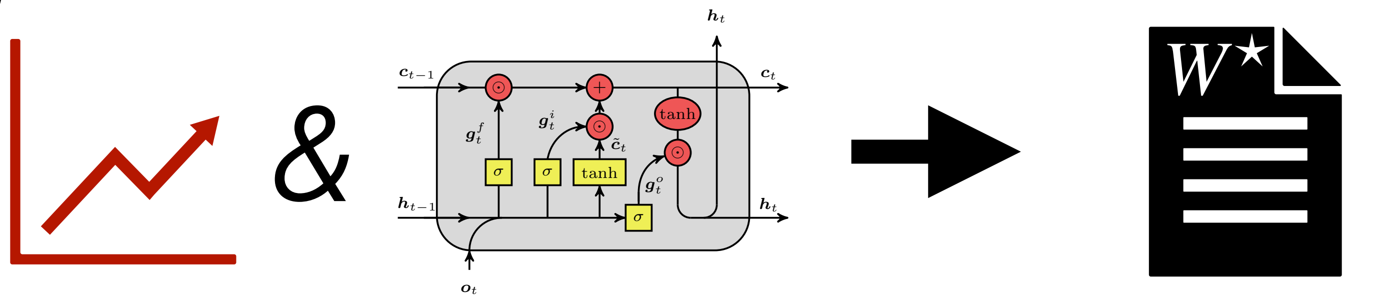
data *model* *TRAIN* *optimal*

2 **TEST**

How to predict the dynamics of TEST (unseen) data?

Forecasting on UNSEEN data - Iterative prediction in practice

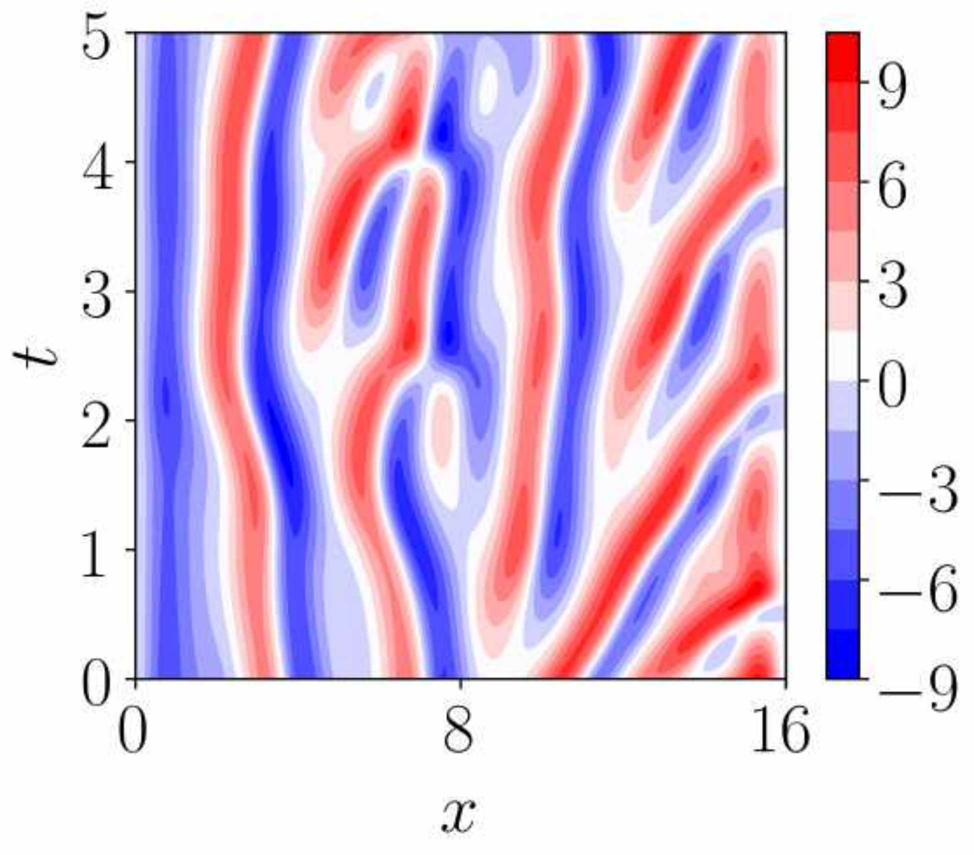
1 TRAIN



2 TEST

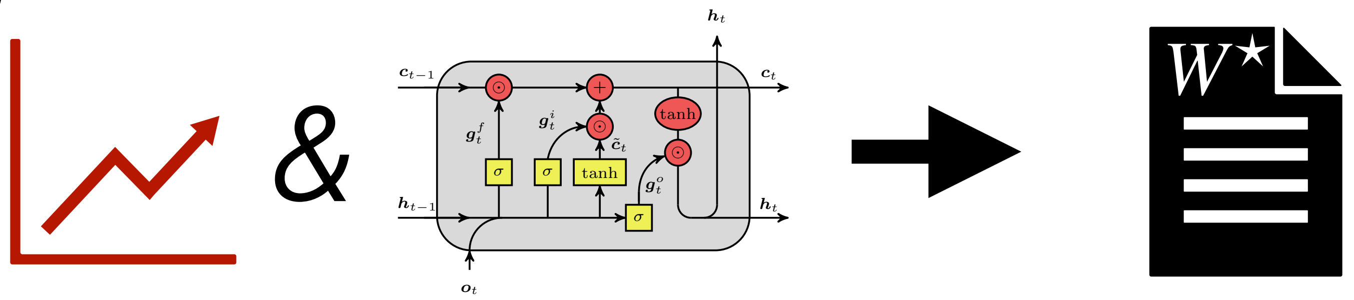
How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* state *dynamics (reference)*



Forecasting on UNSEEN data - Iterative prediction in practice

1 **TRAIN**

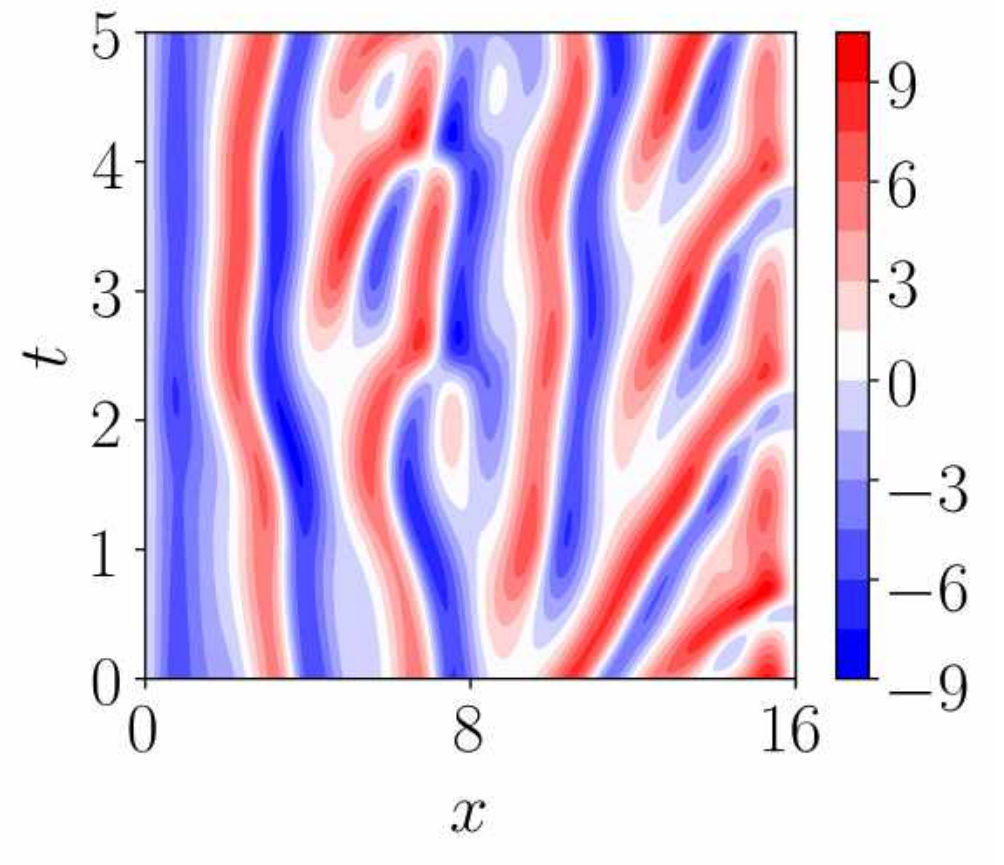


data *model* *TRAIN* *optimal*

2 **TEST**

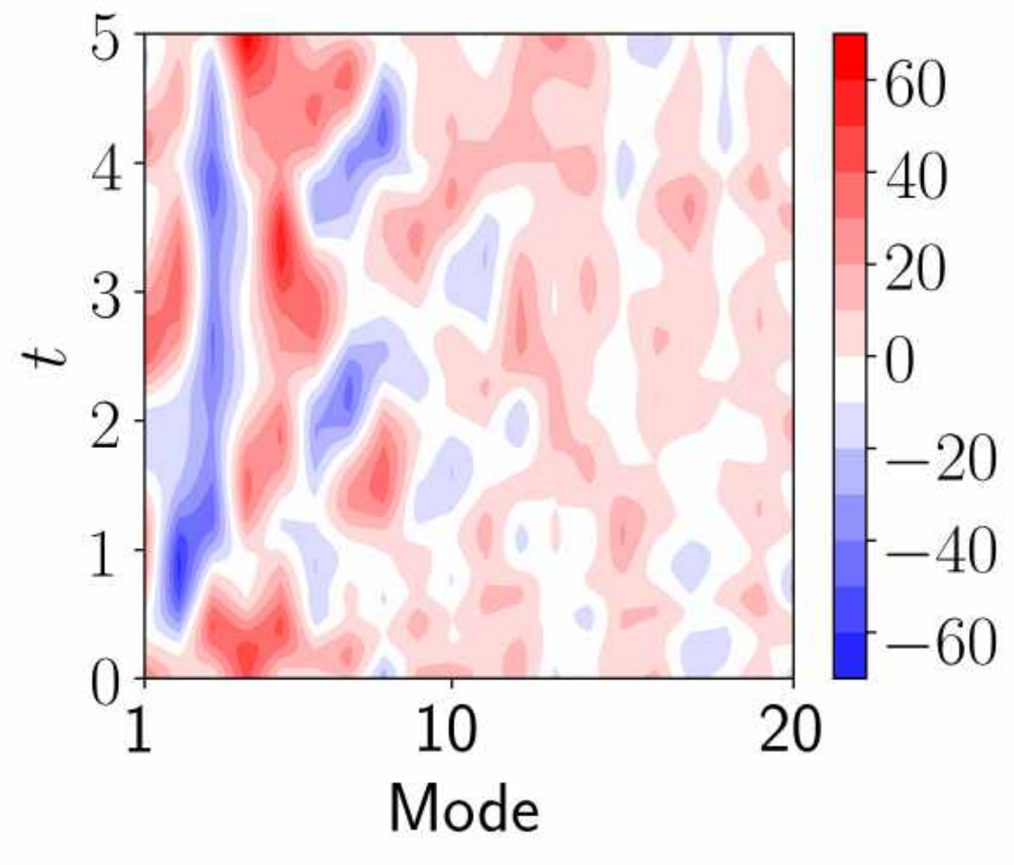
How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* *state dynamics (reference)*



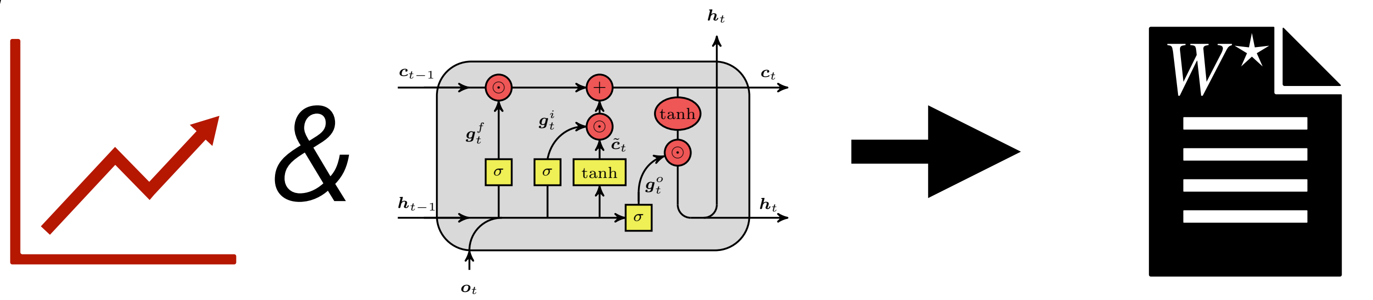
SVD

***SVD Mode* dynamics (reference)**



Forecasting on UNSEEN data - Iterative prediction in practice

1 TRAIN

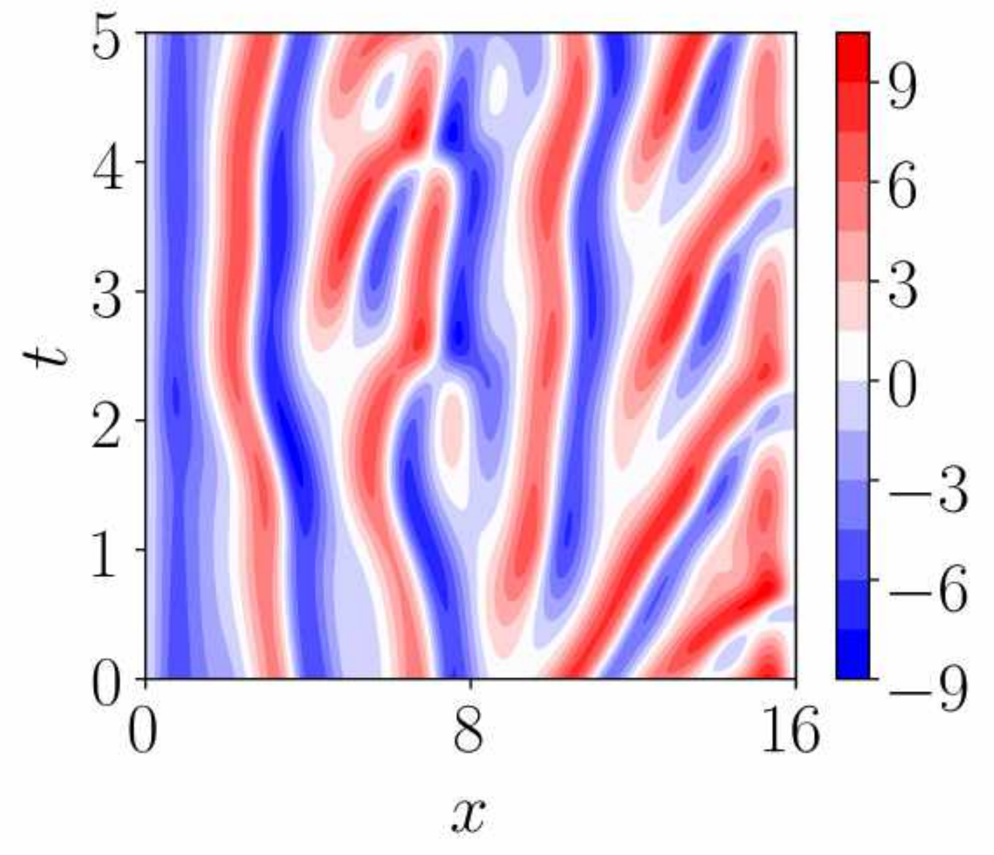


data *model* *TRAIN* *optimal*

2 TEST

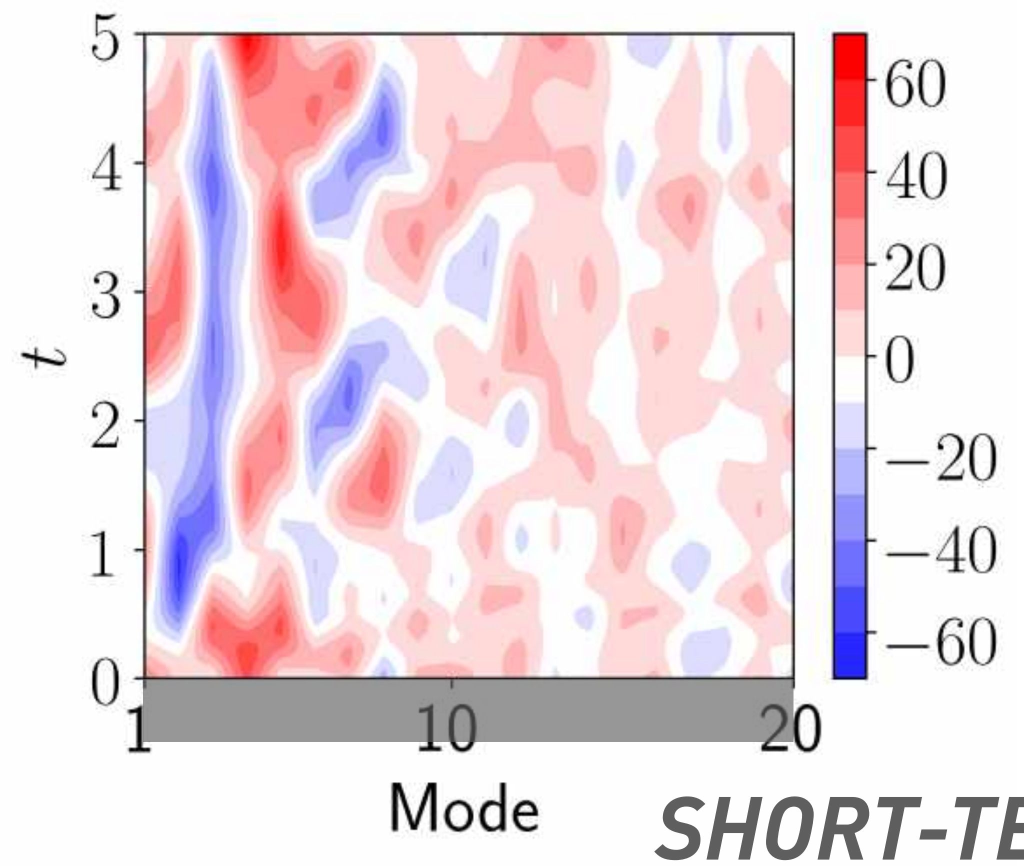
How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* *state dynamics (reference)*



SVD

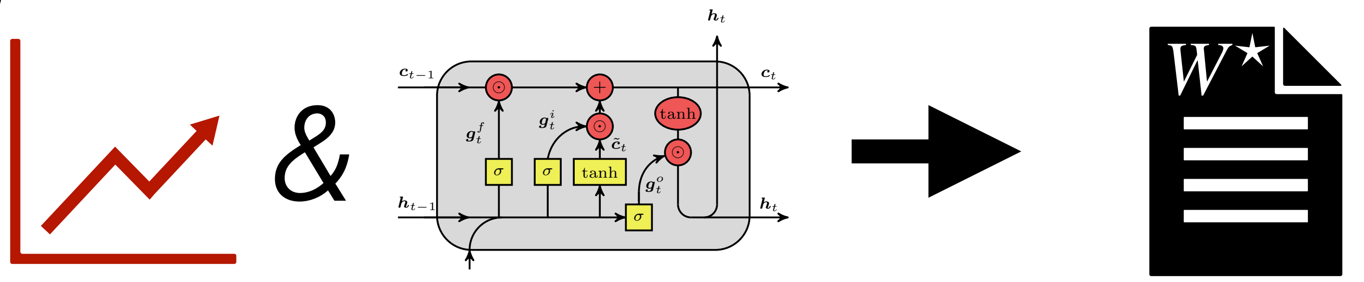
SVD Mode dynamics (reference)



SHORT-TERM HISTORY (known)

Forecasting on UNSEEN data - Iterative prediction in practice

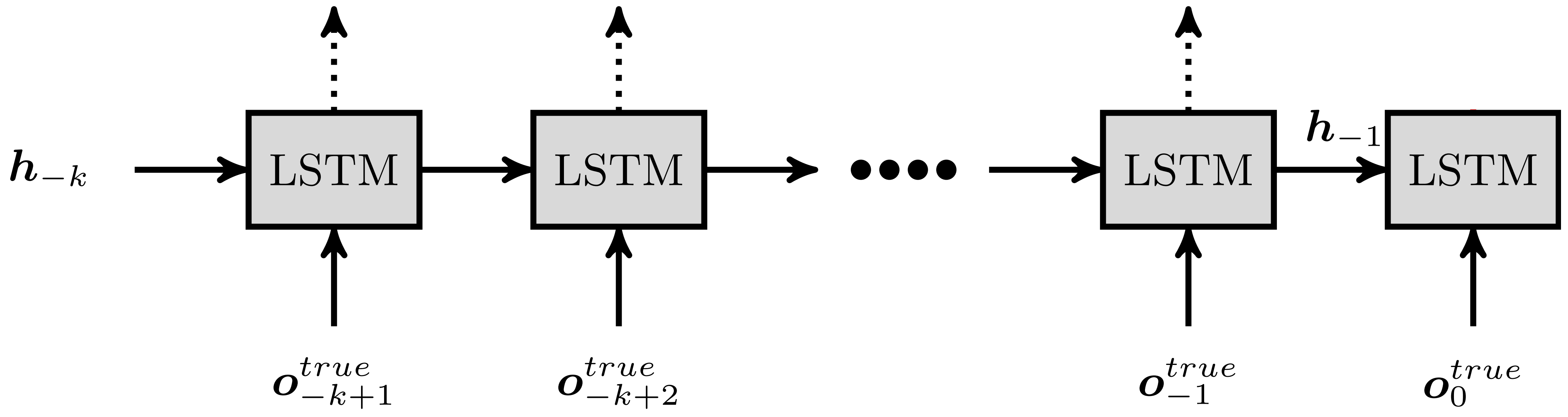
1 TRAIN



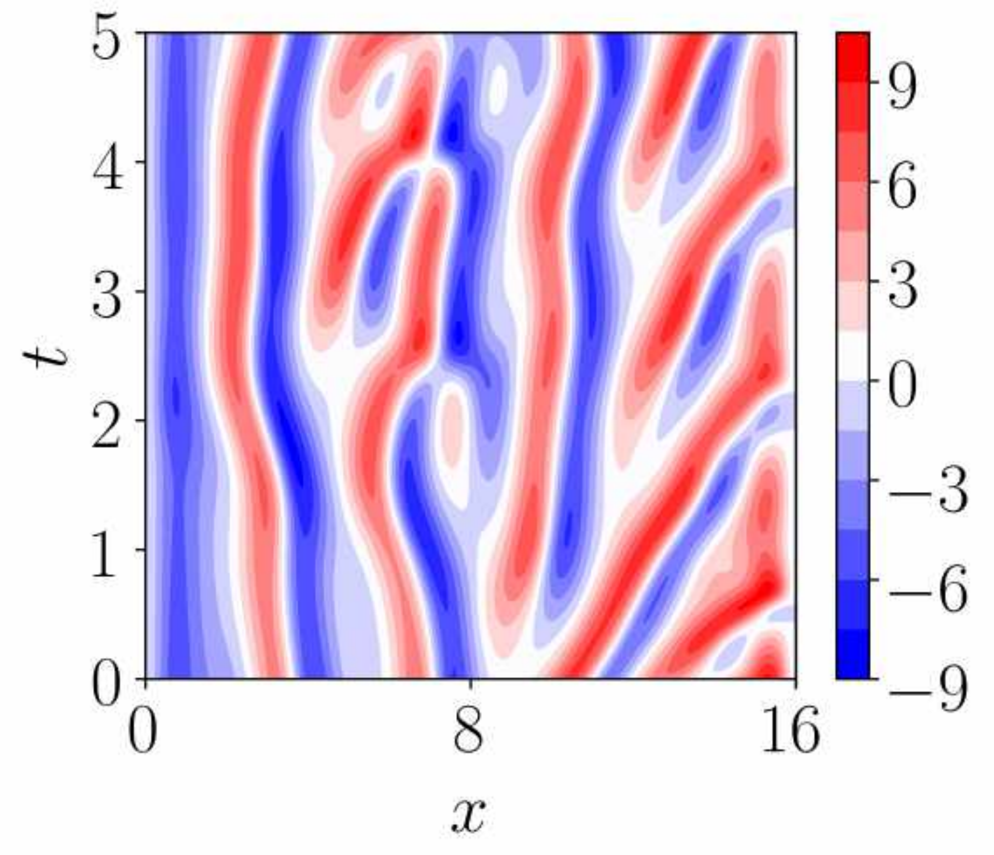
data & *model* *TRAIN* *optimal*

2 TEST

How to predict the dynamics of TEST (unseen) data?

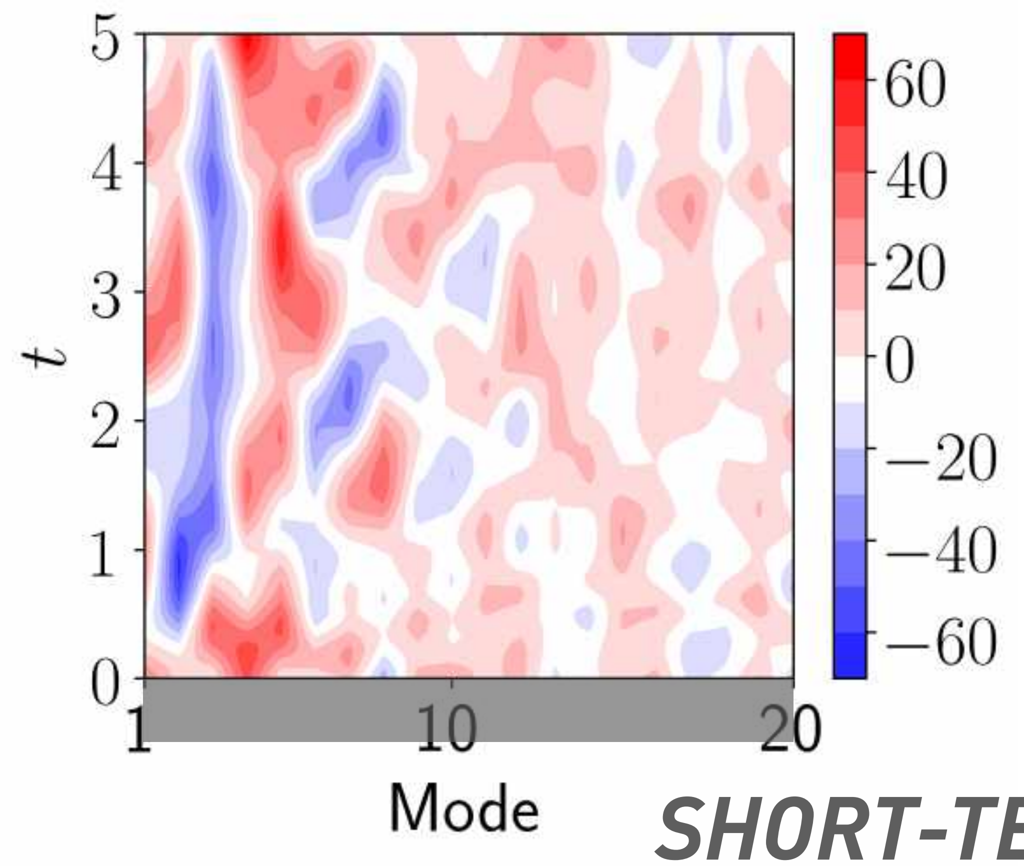


TEST: *UNKNOWN* *state dynamics (reference)*



SVD

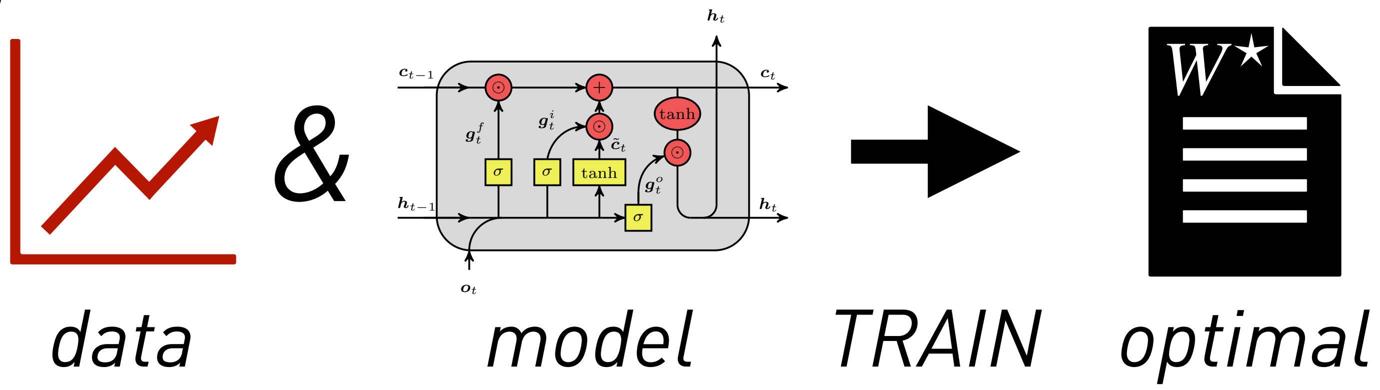
SVD Mode dynamics (reference)



SHORT-TERM HISTORY (known)

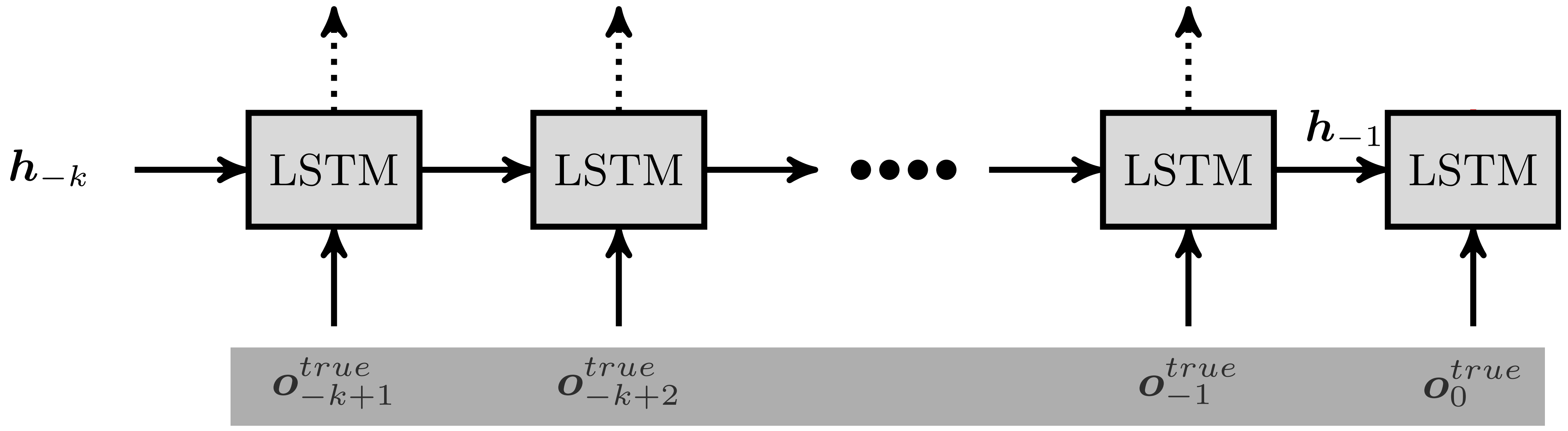
Forecasting on UNSEEN data - Iterative prediction in practice

1 TRAIN



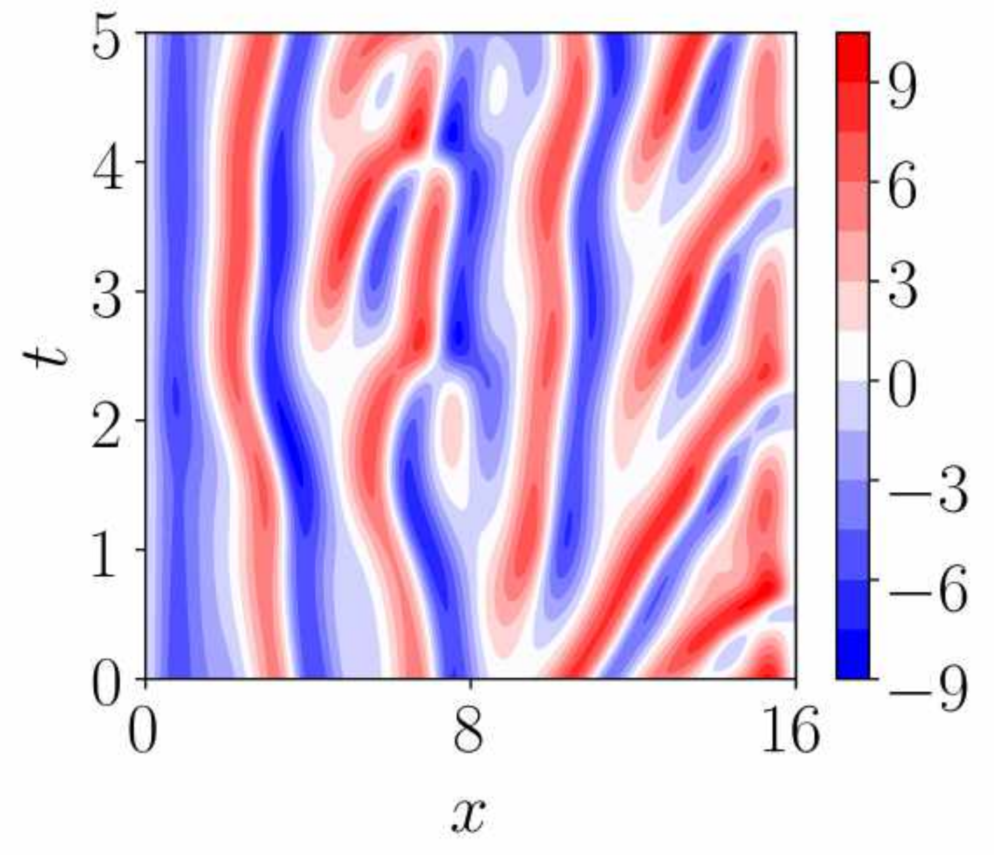
2 TEST

How to predict the dynamics of TEST (unseen) data?



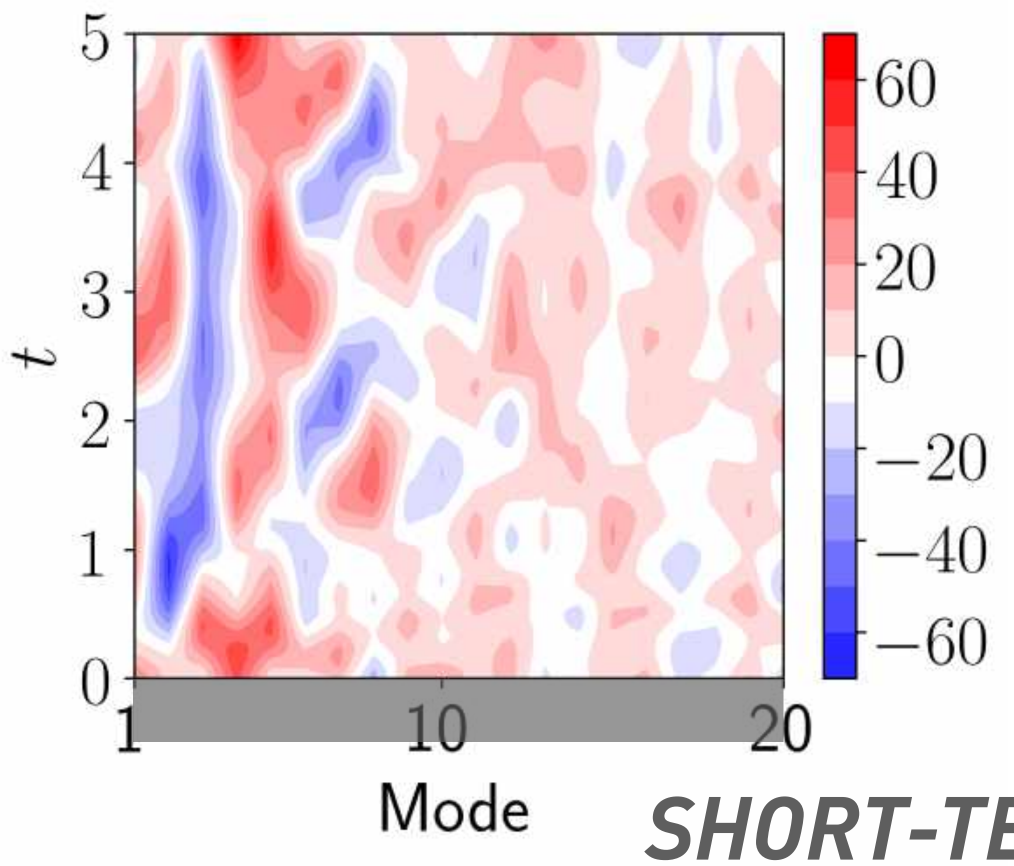
SHORT-TERM HISTORY (known)

TEST: *UNKNOWN* *state dynamics (reference)*



SVD

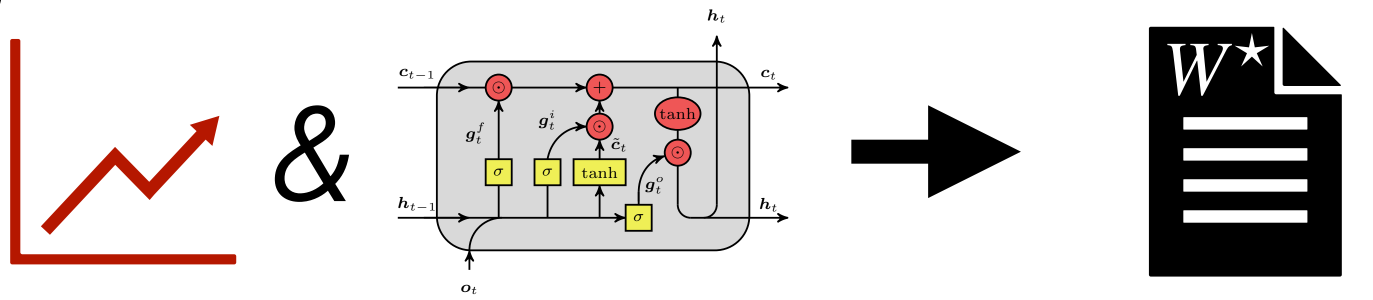
SVD Mode dynamics (reference)



SHORT-TERM HISTORY (known)

Forecasting on UNSEEN data - Iterative prediction in practice

1 **TRAIN**



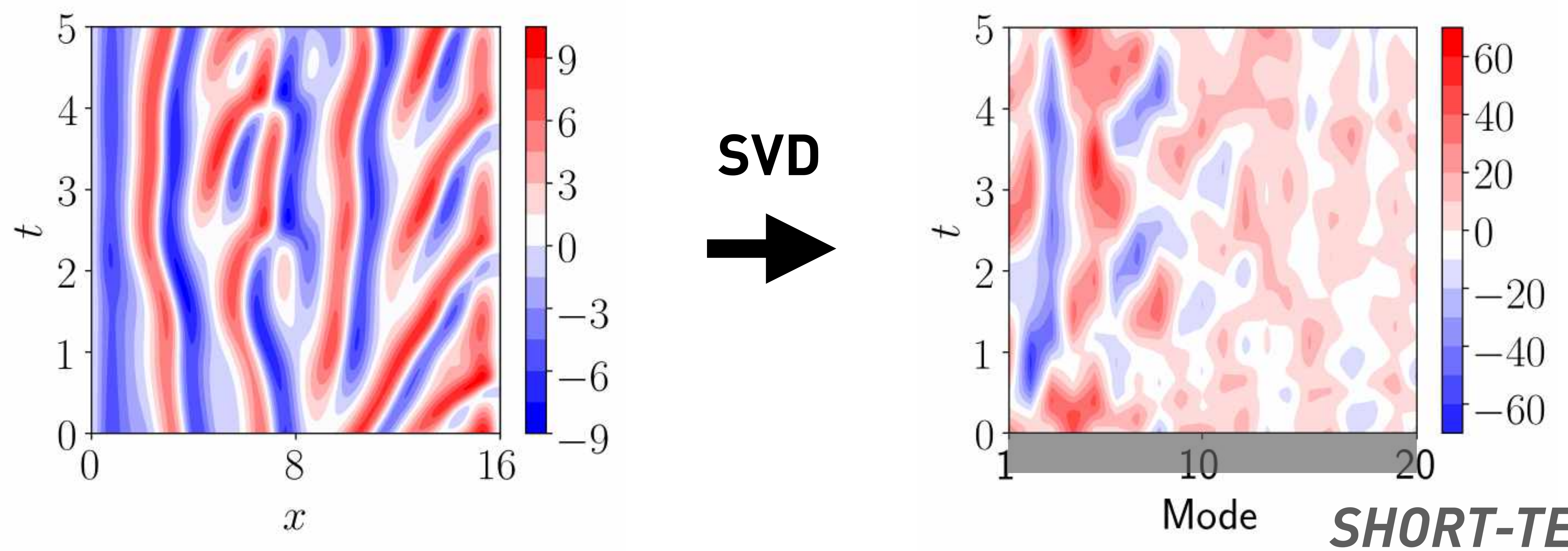
data & *model* → *TRAIN* *optimal*

2 **TEST**

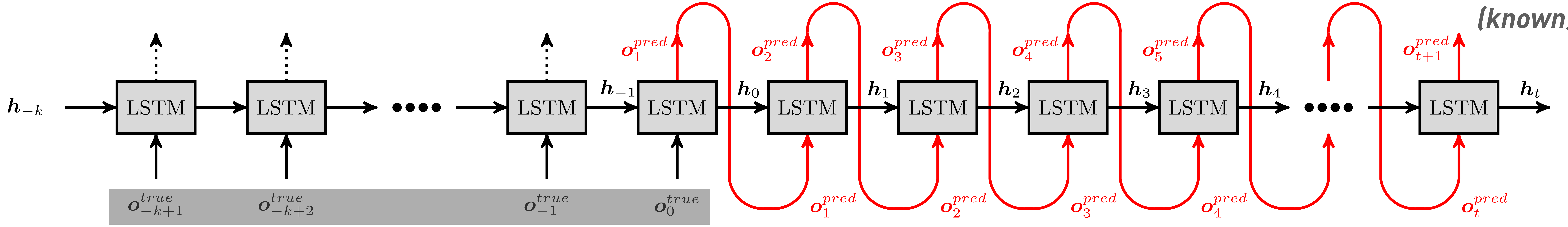
How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* *state dynamics (reference)*

***SVD Mode* dynamics (reference)**



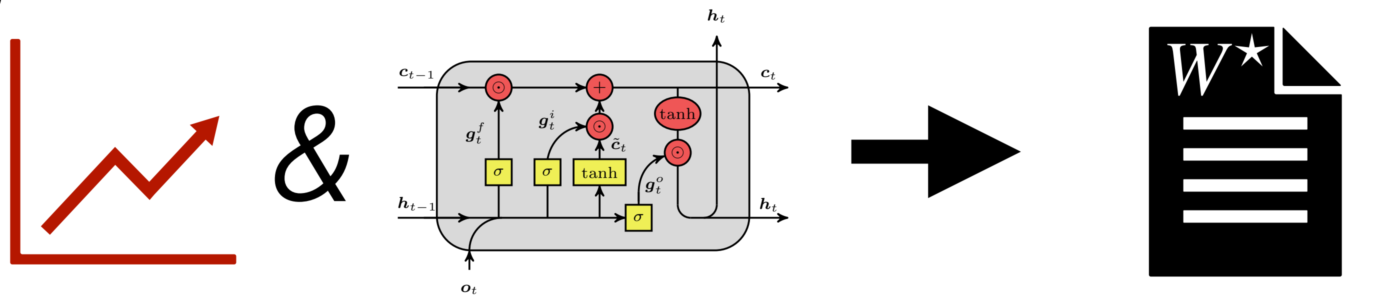
SHORT-TERM HISTORY (known)



SHORT-TERM HISTORY (known)

Forecasting on UNSEEN data - Iterative prediction in practice

1 TRAIN



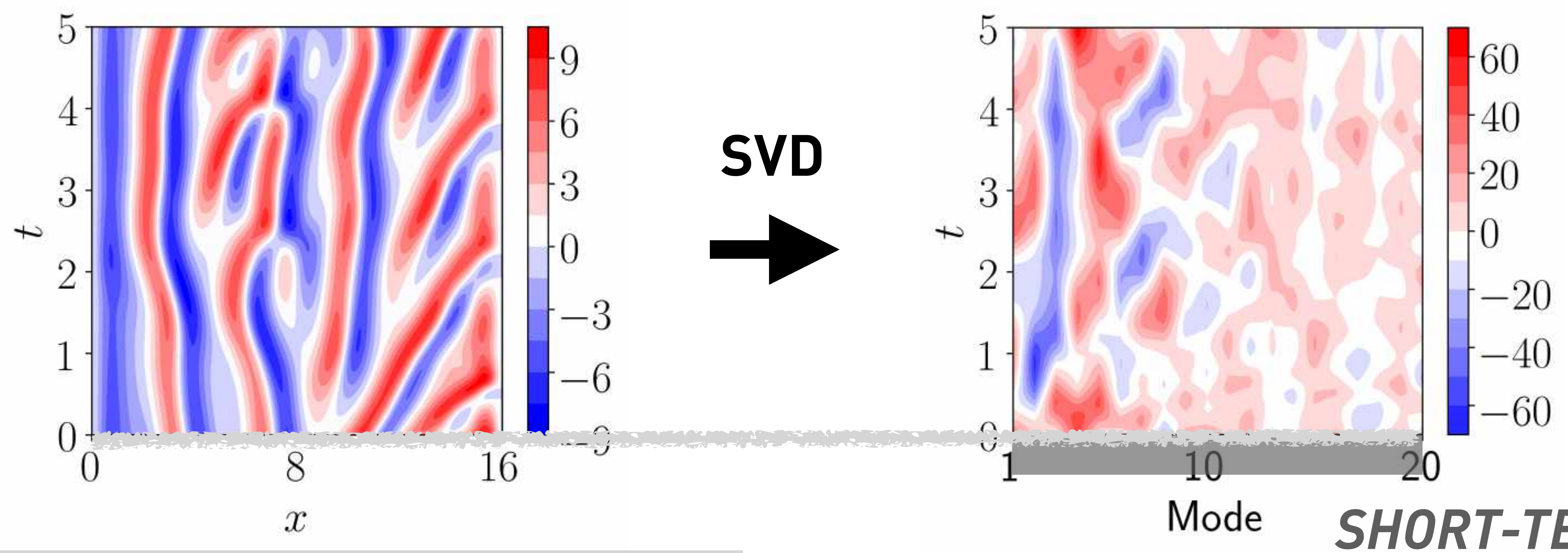
2 TEST

How to predict the dynamics of TEST (unseen) data?

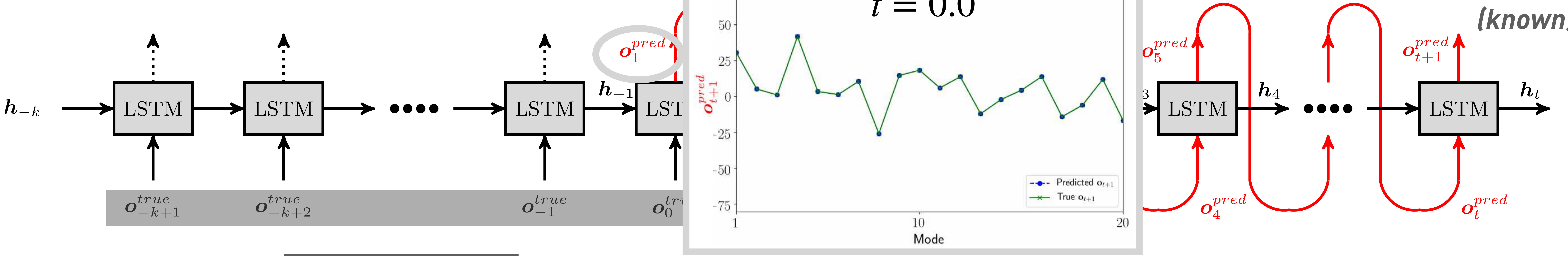
TEST: *UNKNOWN* state dynamics (reference)

SVD

***SVD Mode* dynamics (reference)**



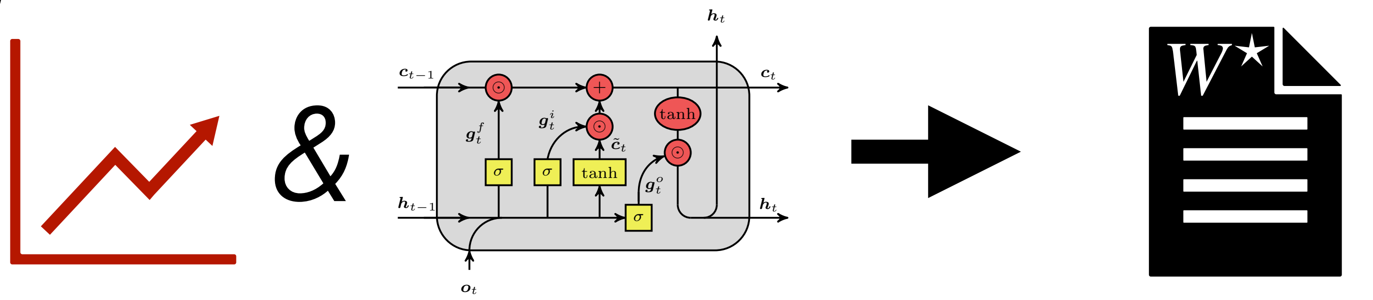
SHORT-TERM HISTORY (known)



SHORT-TERM HISTORY (known)

Forecasting on UNSEEN data - Iterative prediction in practice

1 **TRAIN**



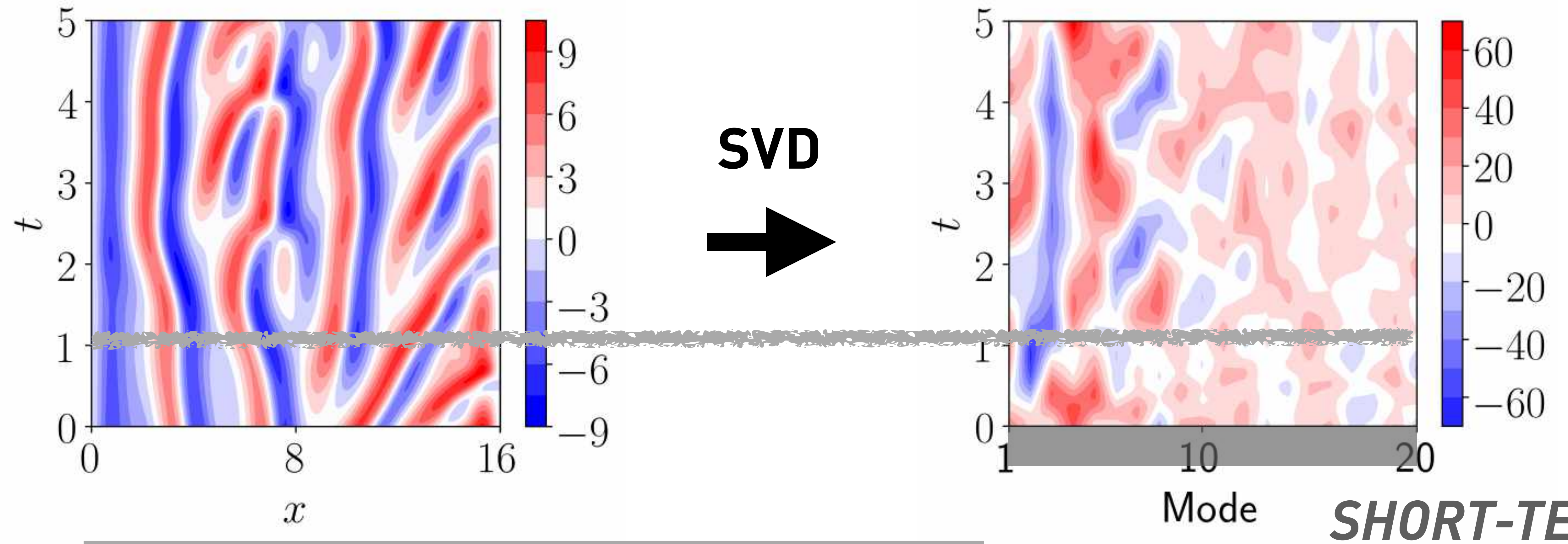
data *model* *TRAIN* *optimal*

2 **TEST**

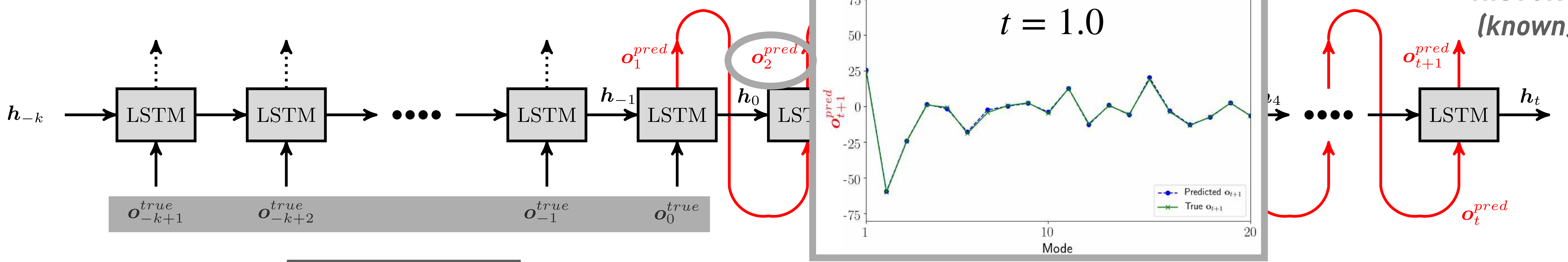
How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* state dynamics (reference)

SVD Mode dynamics (reference)



SHORT-TERM HISTORY (known)

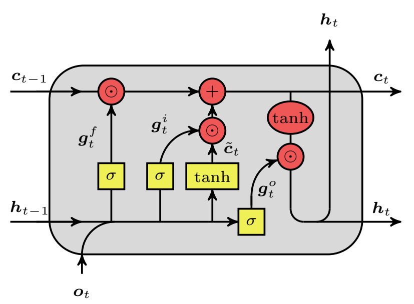


SHORT-TERM HISTORY (known)

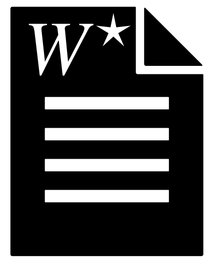
Forecasting on UNSEEN data - Iterative prediction in practice

1 **TRAIN**

&



→ *TRAIN*

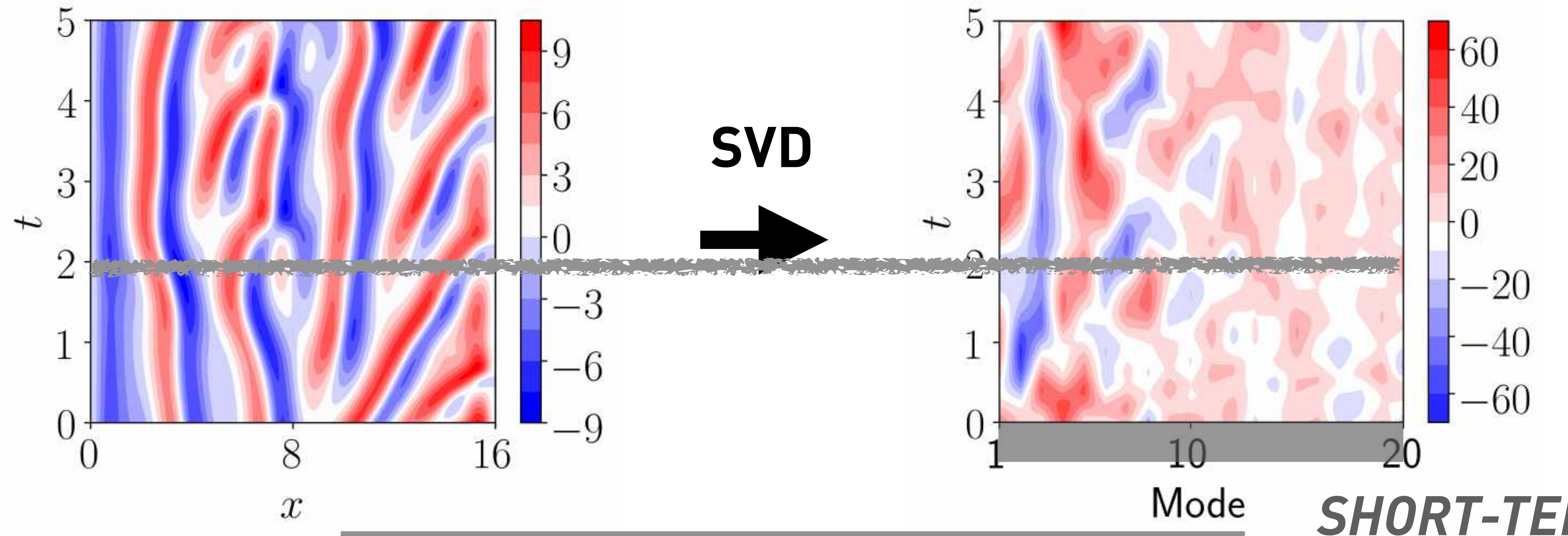


2 **TEST**

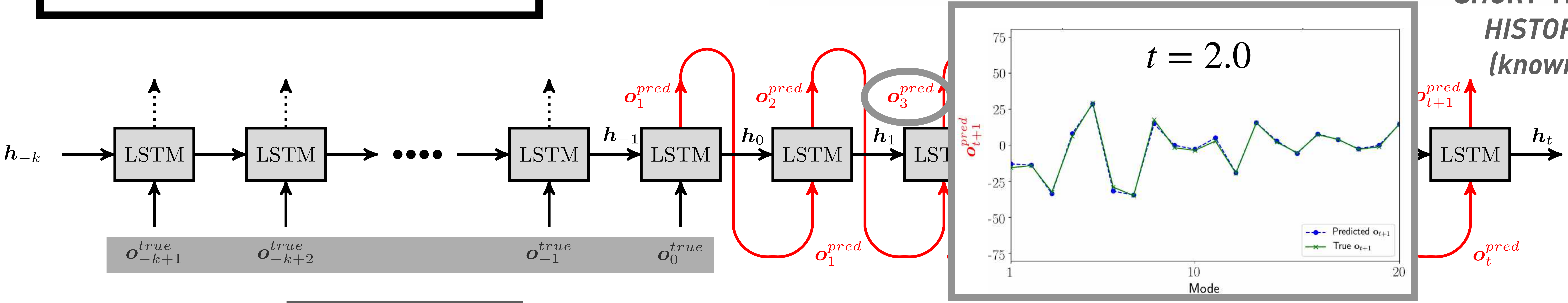
How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* state dynamics (reference)

SVD Mode dynamics (reference)



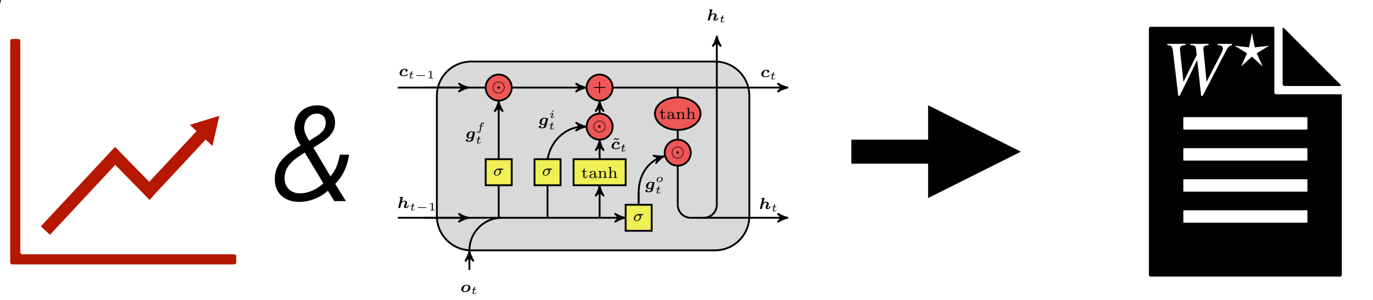
SHORT-TERM HISTORY (known)



SHORT-TERM HISTORY (known)

Forecasting on UNSEEN data - Iterative prediction in practice

1 **TRAIN**



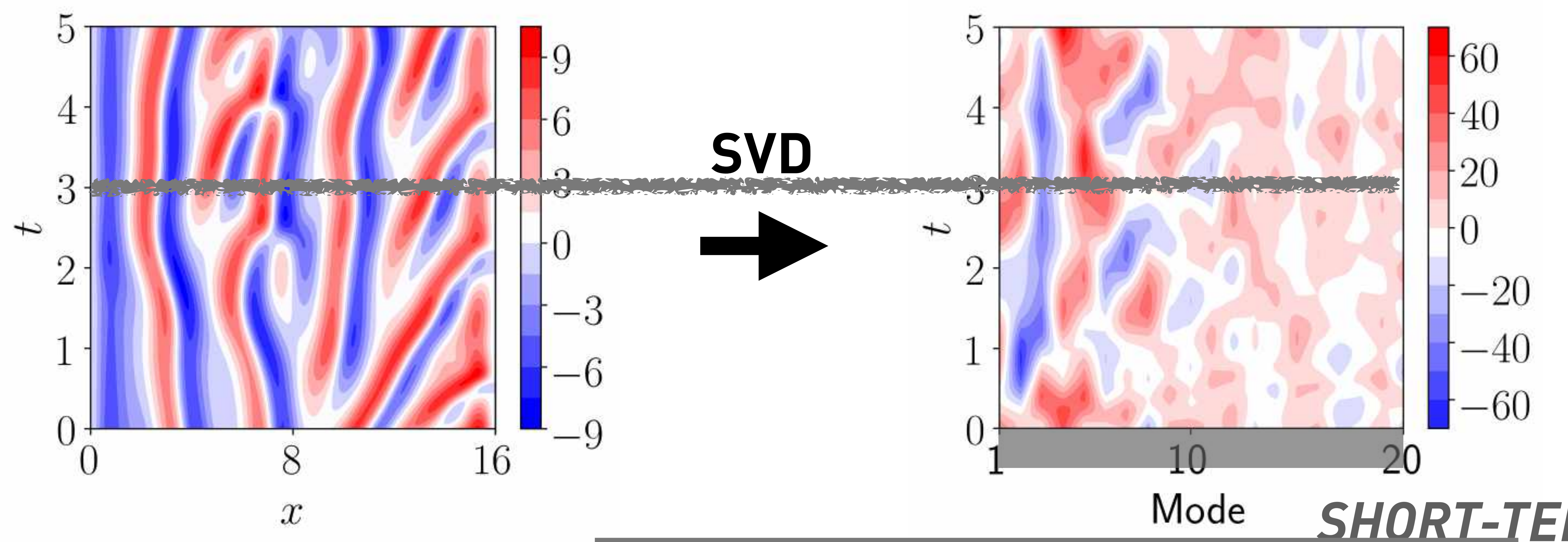
data *model* *TRAIN* *optimal*

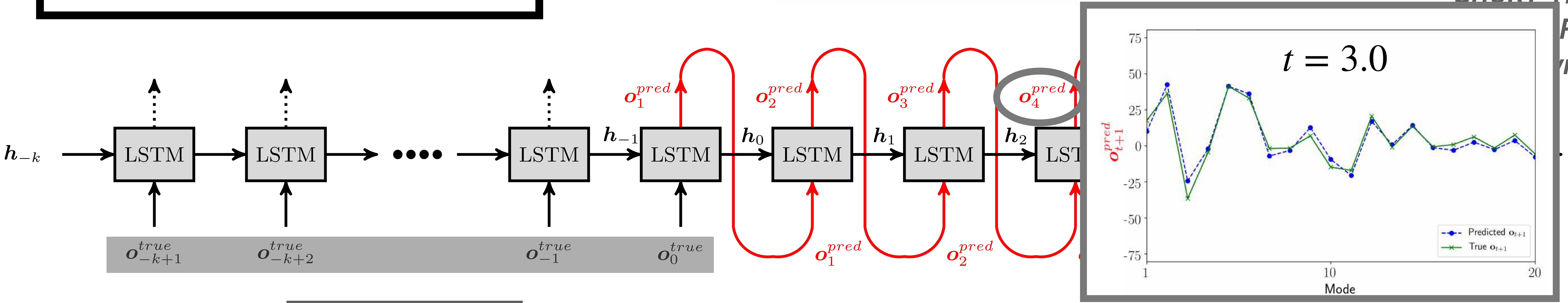
2 **TEST**

How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* state *dynamics (reference)*

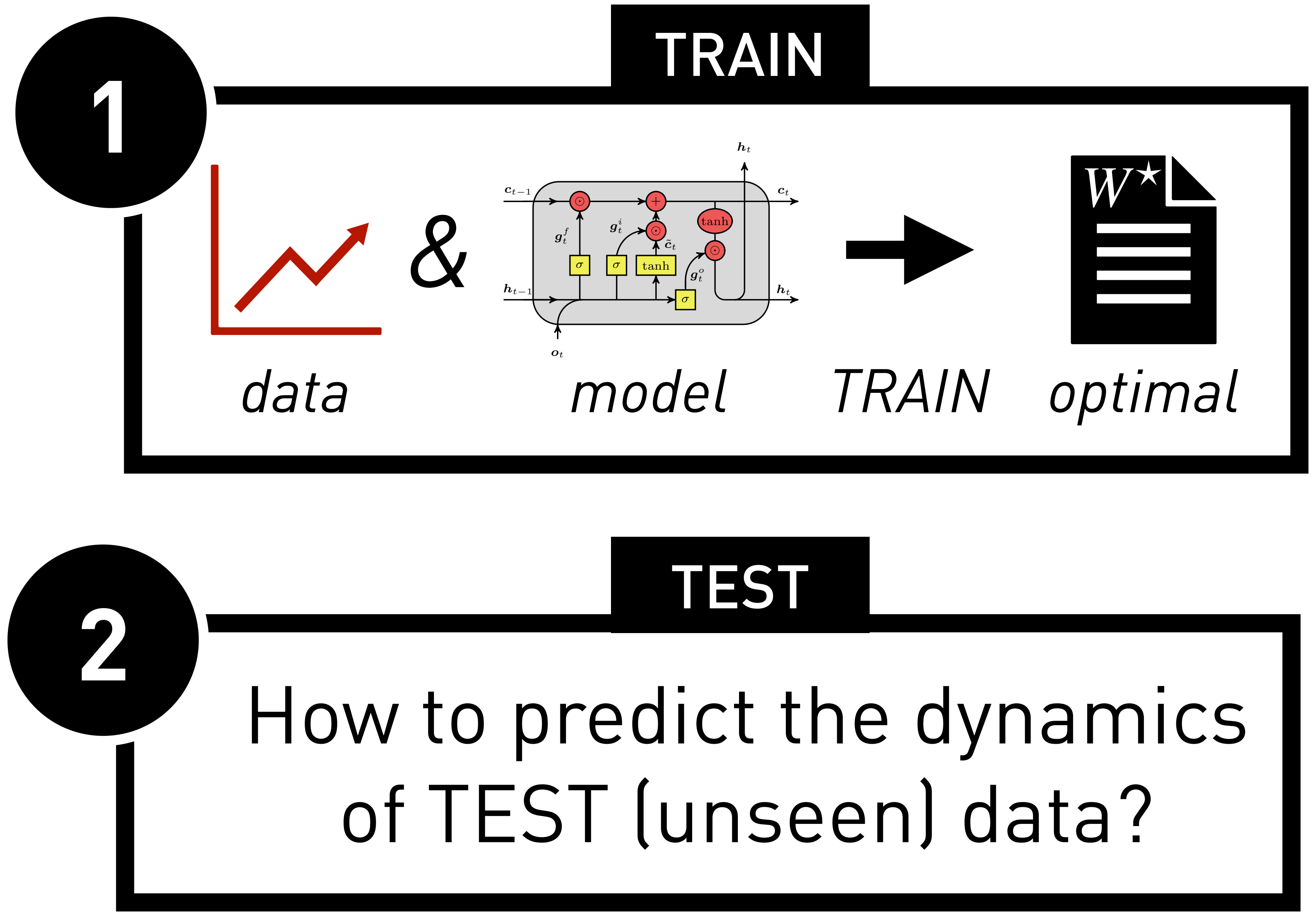
***SVD Mode* dynamics (reference)**

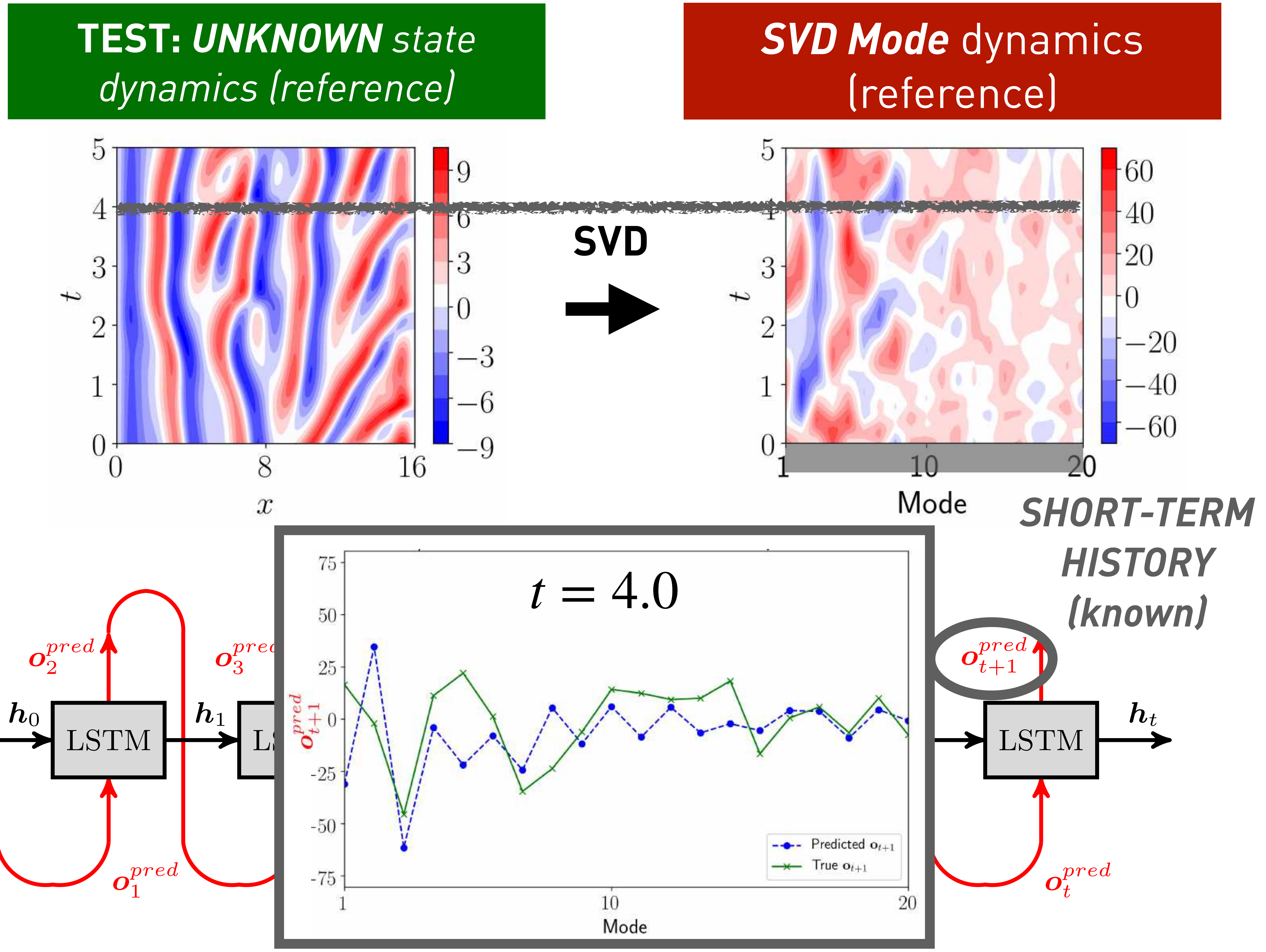


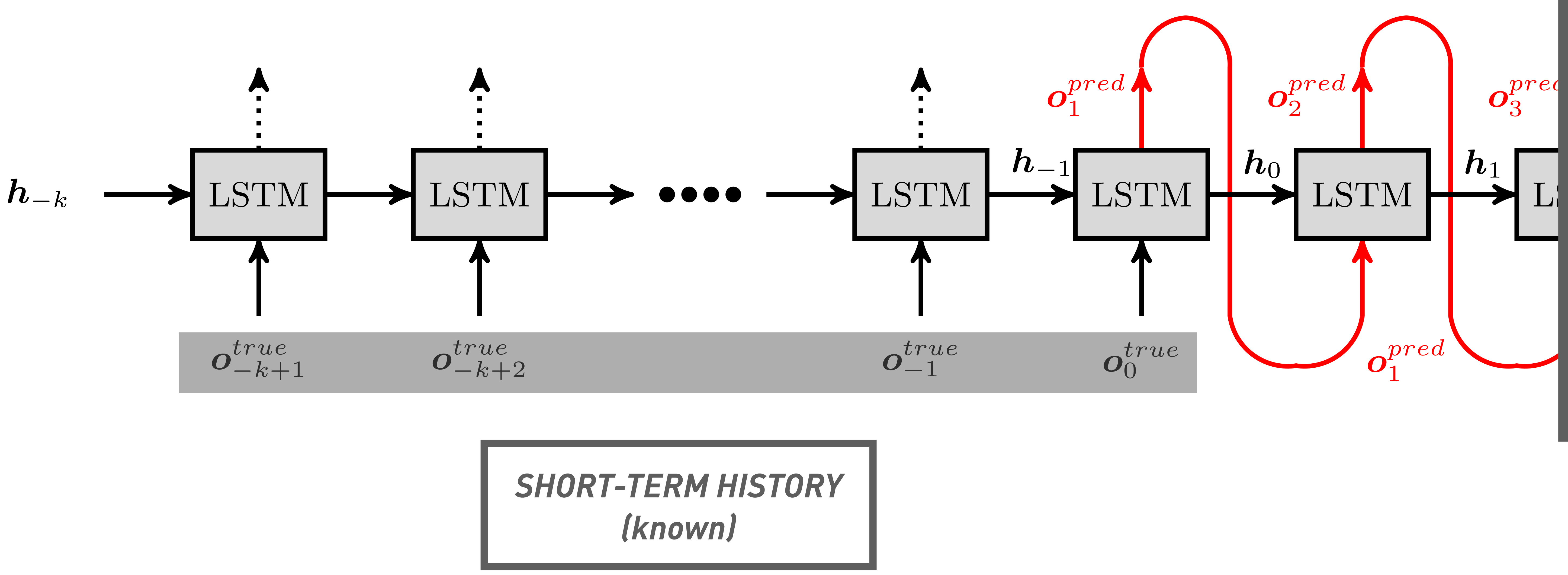


SHORT-TERM HISTORY (known)

Forecasting on UNSEEN data - Iterative prediction in practice

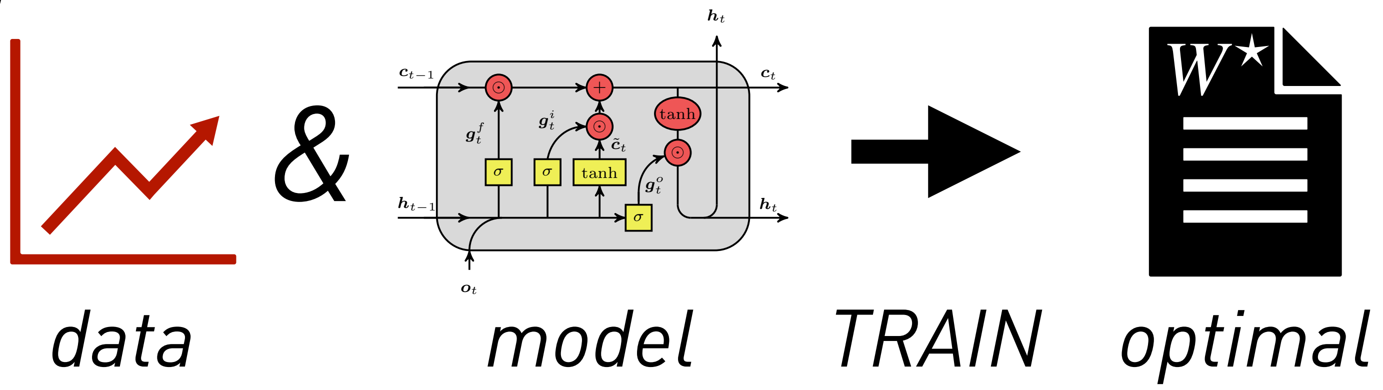






Forecasting on UNSEEN data - Iterative prediction in practice

1 TRAIN

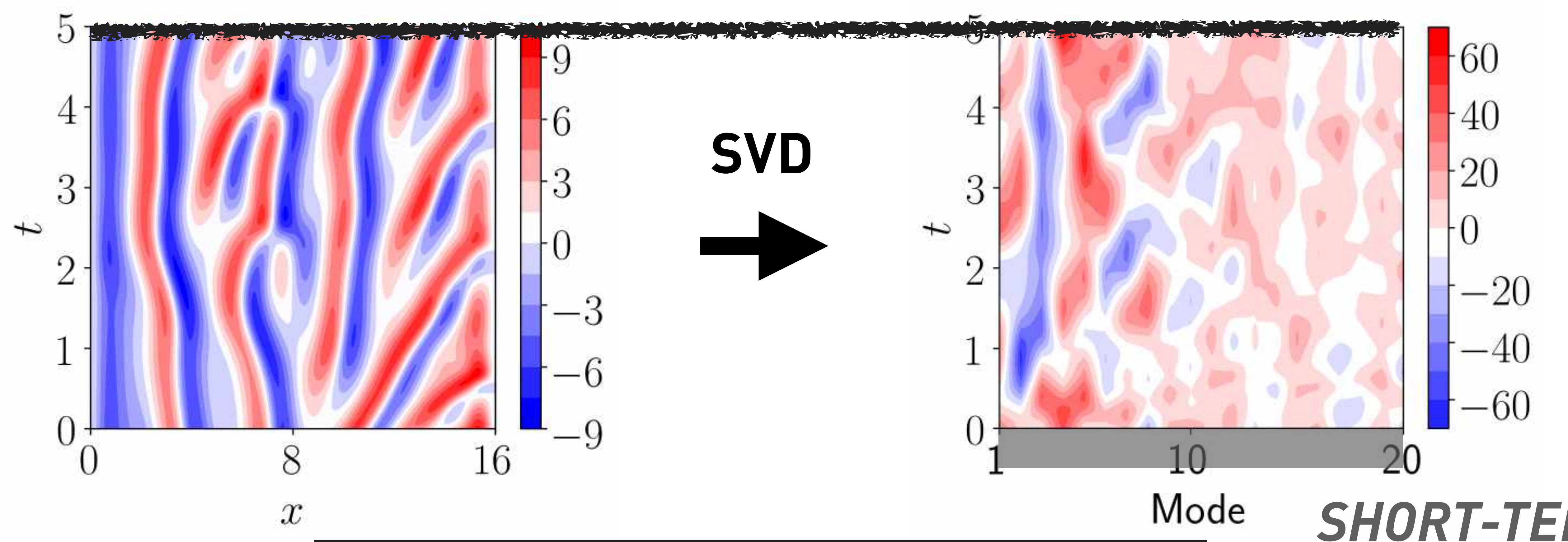


2 TEST

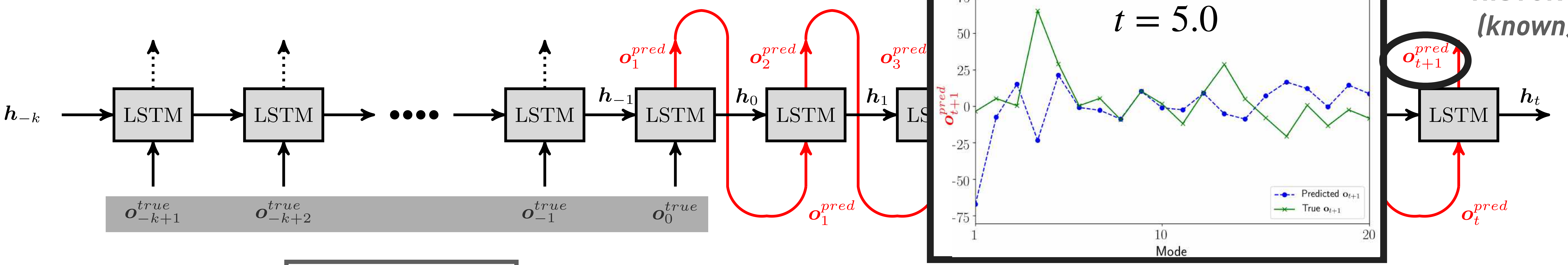
How to predict the dynamics of TEST (unseen) data?

TEST: *UNKNOWN* *state dynamics (reference)*

SVD Mode dynamics (reference)



SHORT-TERM HISTORY (known)



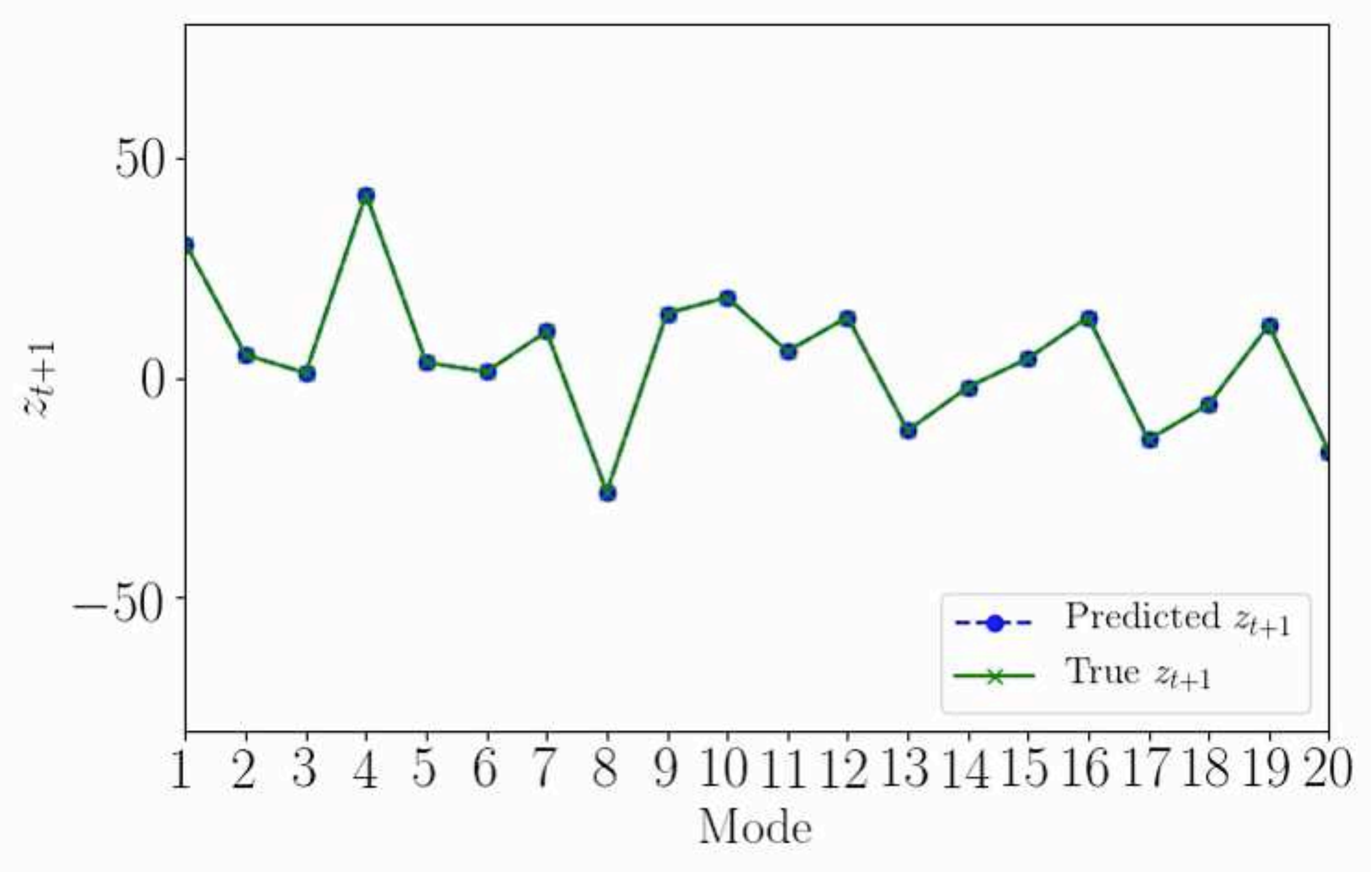
SHORT-TERM HISTORY (known)

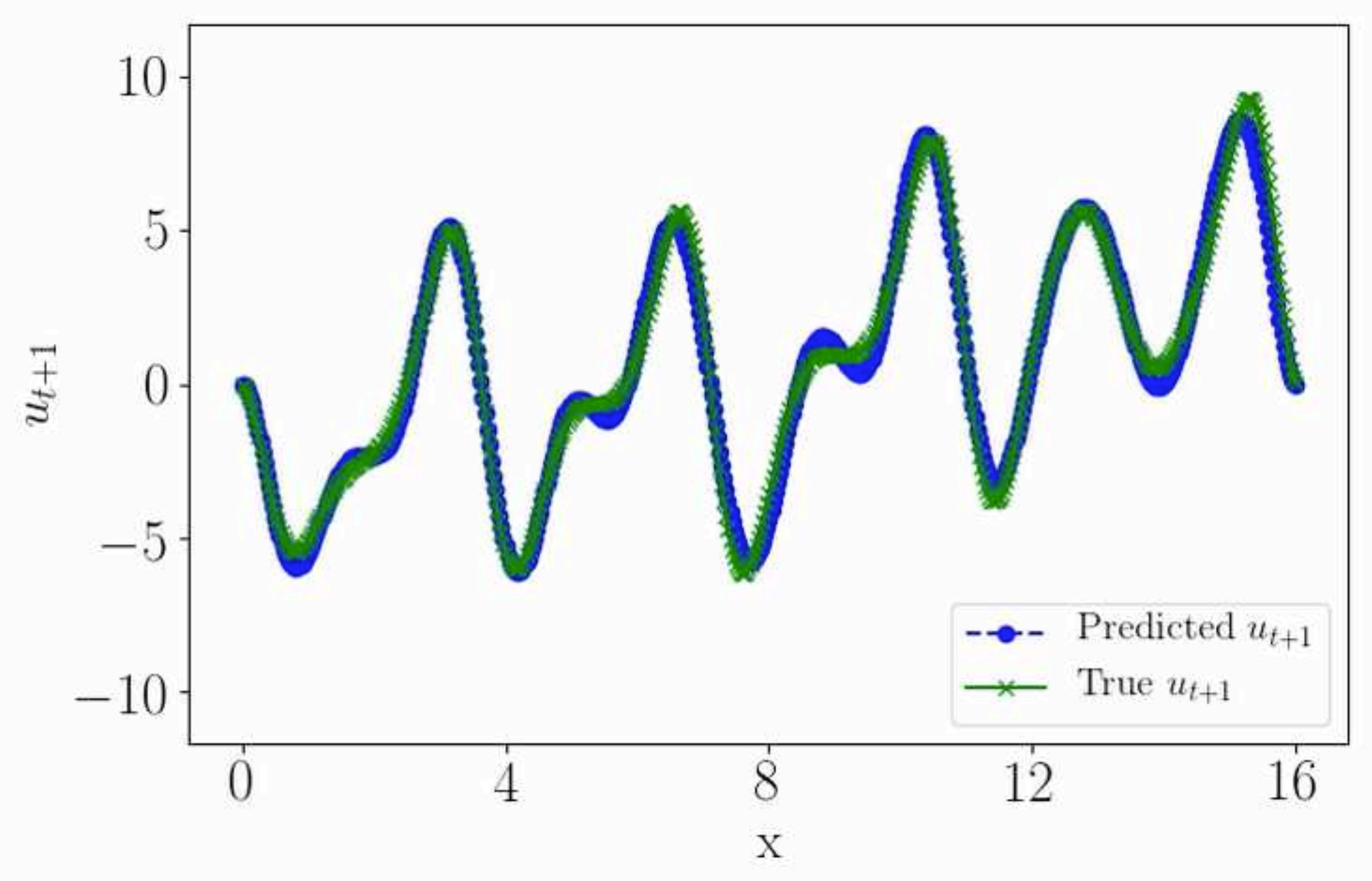
Accumulation of prediction error

prediction in **reduced** space

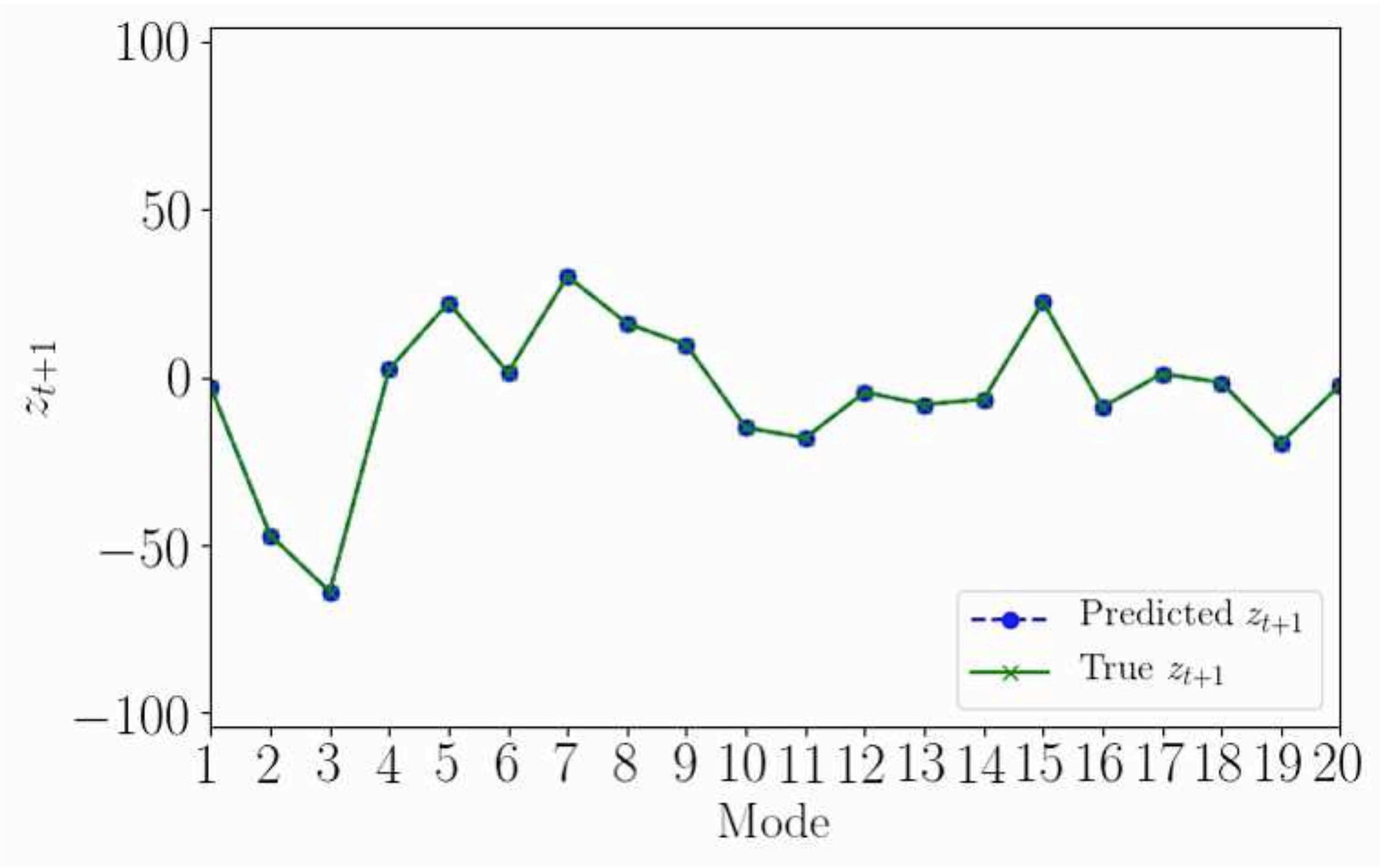
expanded in **high-dimensional** space

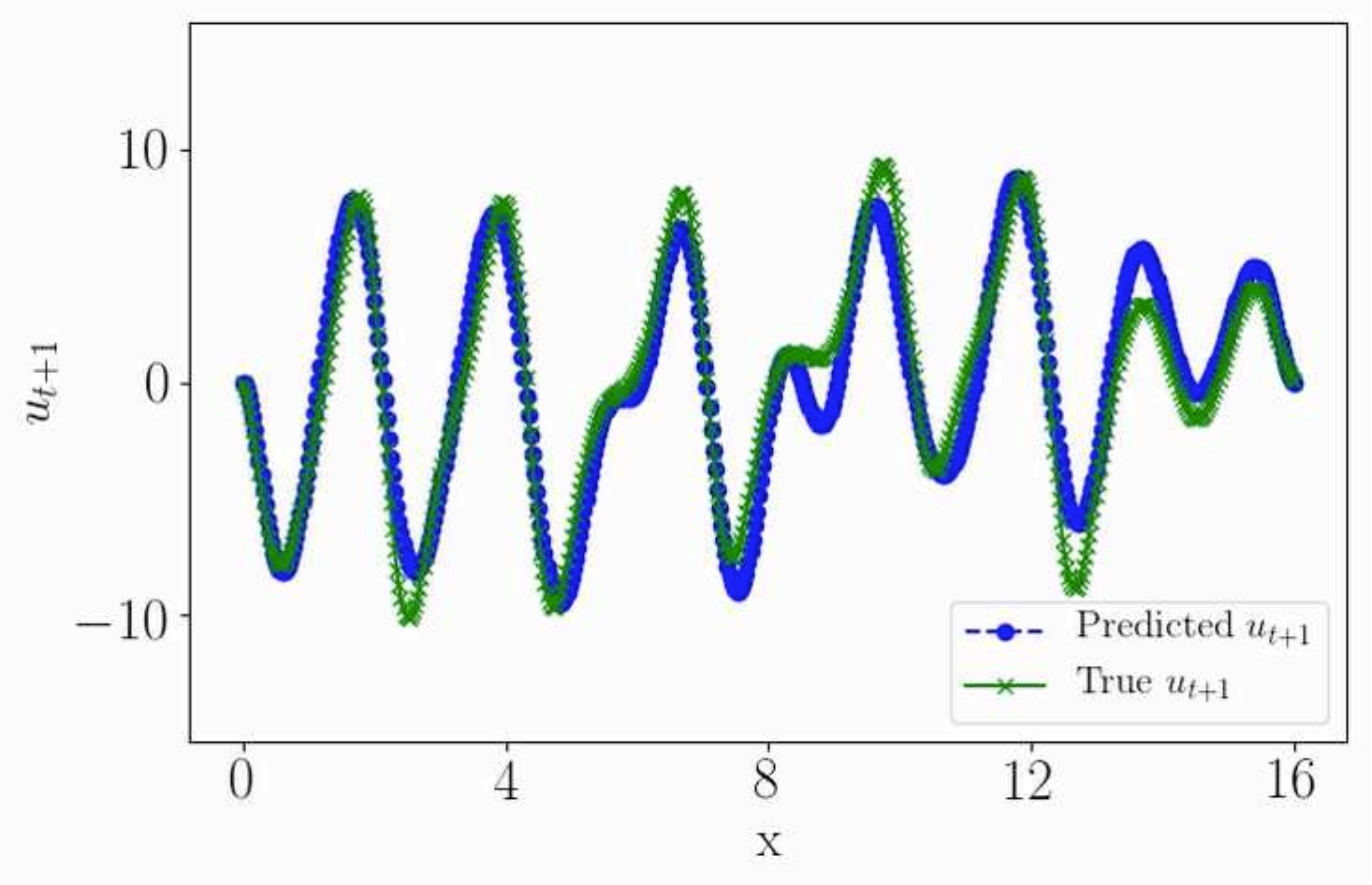
$\tilde{L} \approx 8$





$\tilde{L} \approx 10$



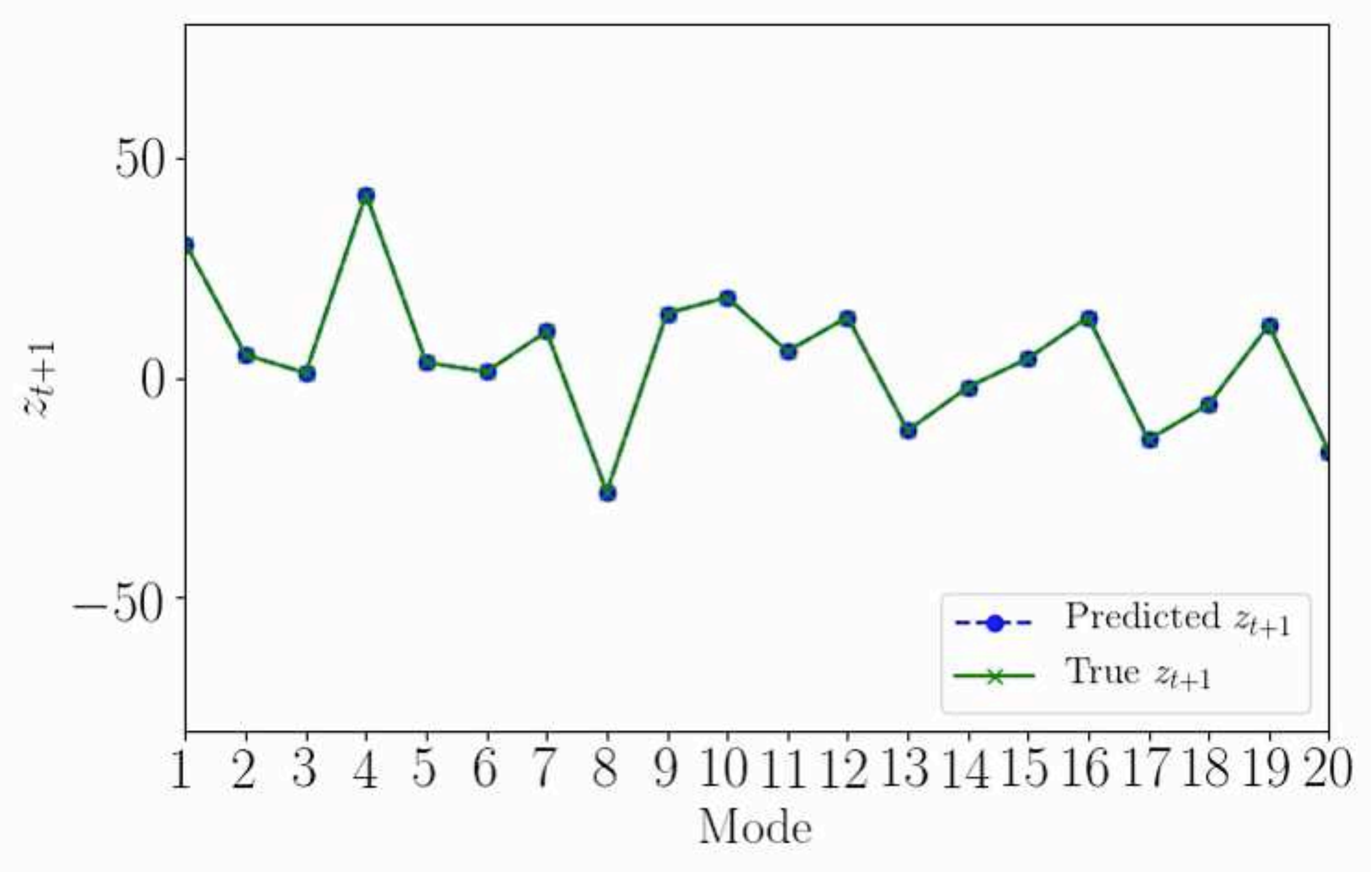


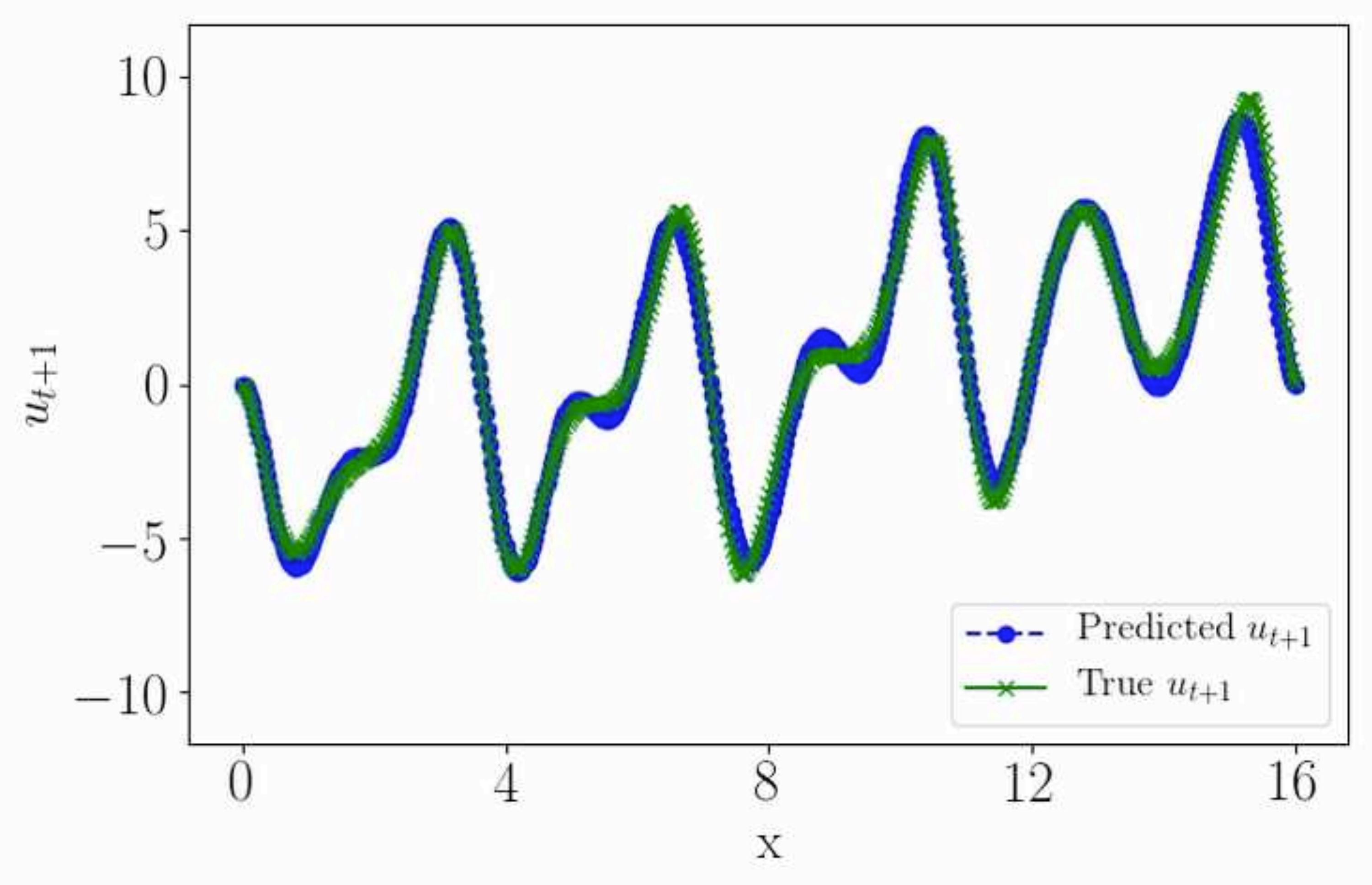
Accumulation of prediction error

prediction in **reduced** space

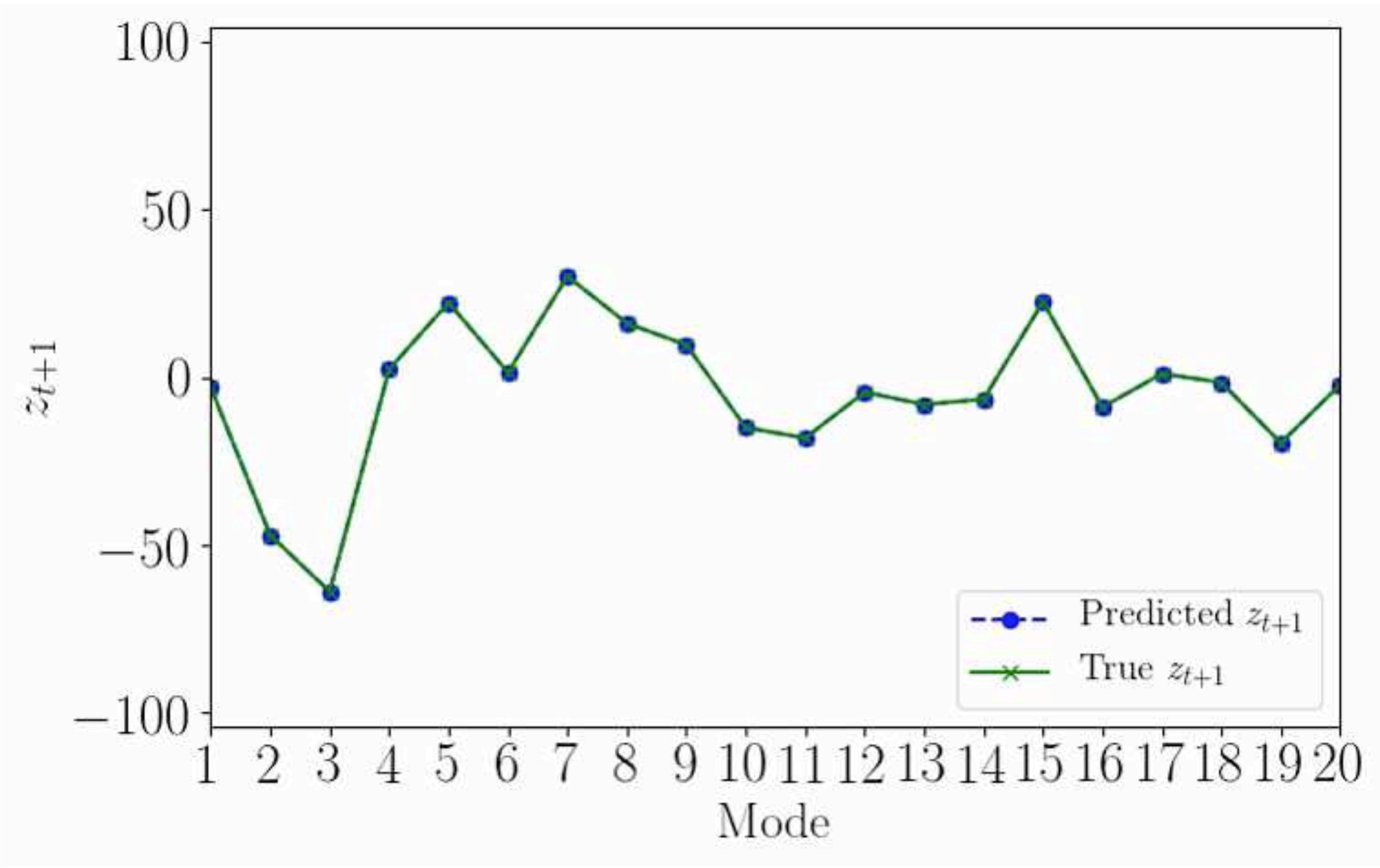
expanded in **high-dimensional** space

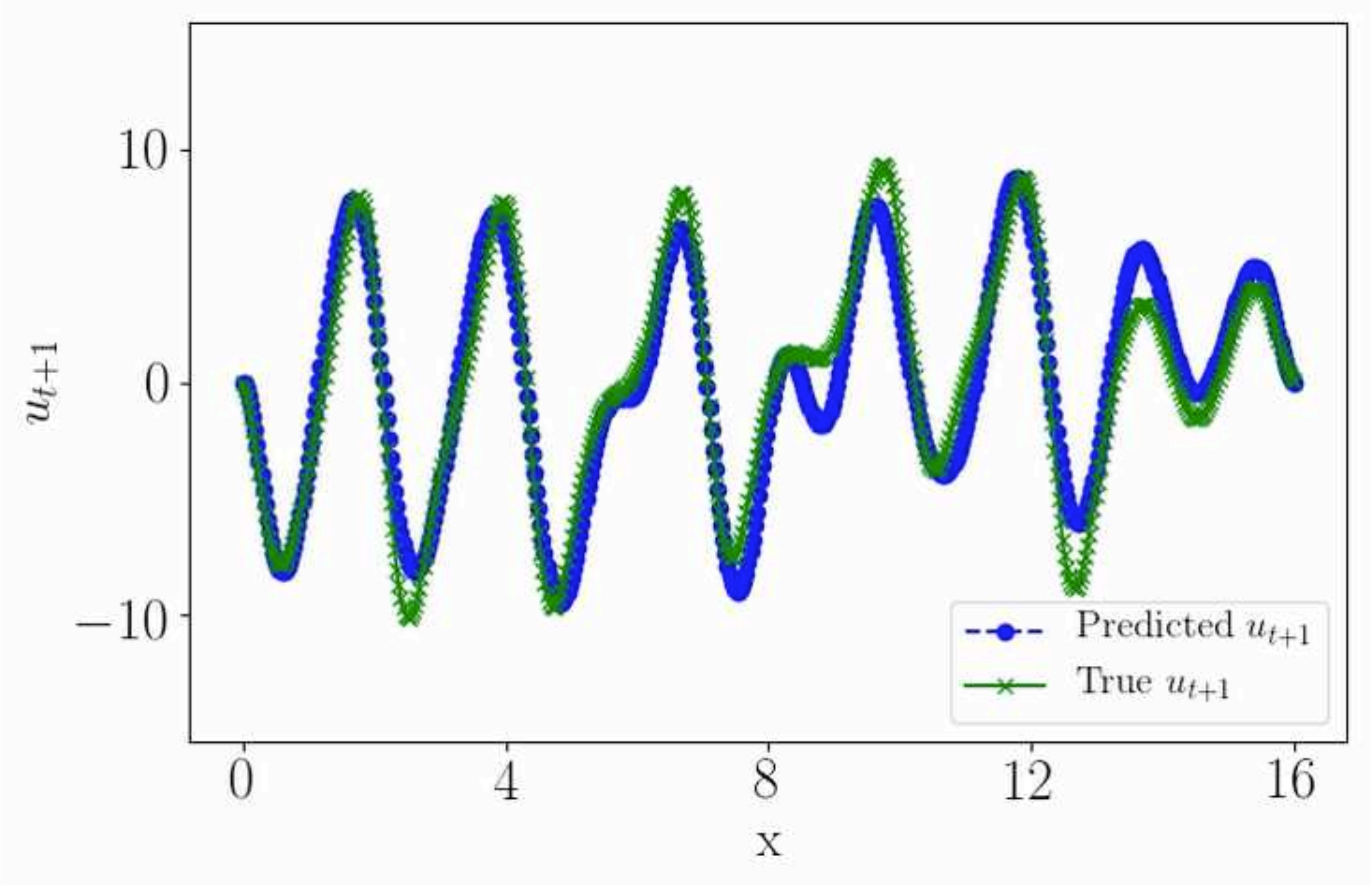
$\tilde{L} \approx 8$





$\tilde{L} \approx 10$





Challenges

Challenges

1. Iterative prediction error **accumulates** leading to unphysical predictions
   - divergence from attractor

Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

- Dynamics underrepresented in training data

Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

- Dynamics underrepresented in training data
- Scarce data in attractor boundaries

Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

- Dynamics underrepresented in training data
- Scarce data in attractor boundaries
- Under-resolved high dimensional dynamics

Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

- Dynamics underrepresented in training data
- Scarce data in attractor boundaries
- Under-resolved high dimensional dynamics
- Models not generalising / distribution shift

Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions

- divergence from attractor

- Dynamics underrepresented in training data
- Scarce data in attractor boundaries
- Under-resolved high dimensional dynamics
- Models not generalising / distribution shift

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

Mean Stochastic Model

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

parameters estimated from **data**

wiener process

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

parameters estimated from **data**

wiener process

$$c = \frac{1}{T}$$

decorrelation time

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

parameters estimated from **data**

wiener process

$$\zeta = \sqrt{-2\, c\, \sigma_z}$$

data standard deviation

$$c = \frac{1}{T}$$

decorrelation time

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap
- Relies on **global attractor statistics**

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

parameters estimated from **data** | wiener process

$$\zeta = \sqrt{-2\, c\, \sigma_z} \qquad c = \frac{1}{T}$$

data standard deviation | **decorrelation time**

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap
- Relies on **global attractor statistics**
- Converges in the **long-term** in **mean statistical behavior**

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

parameters estimated from **data**

wiener process

$$\zeta = \sqrt{-2\, c\, \sigma_z}$$

data standard deviation

$$c = \frac{1}{T}$$

decorrelation time

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap
- Relies on **global attractor statistics**
- Converges in the **long-term** in **mean statistical behavior**
- Efficient & effective in highly chaotic systems [1,2]

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

parameters estimated from **data** | wiener process

$$\zeta = \sqrt{-2\, c\, \sigma_z} \qquad c = \frac{1}{T}$$

data standard deviation | **decorrelation time**

[1] **AJ Majda, J Harlim,** *Filtering complex turbulent systems,* **Cambridge University Press, 2012**
[2] **ZY Wan, TP Sapsis,** *Reduced-space Gaussian Process Regression for data-driven probabilistic forecast of chaotic dynamical systems,* **Physica D: Nonlinear Phenomena, 2017**

Capturing Long-Term Behavior

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

ADD - > MEAN STOCHASTIC MODEL (MSM)

- Ornstein-Uhlenbeck process - computationally cheap
- Relies on **global attractor statistics**
- Converges in the **long-term** in **mean statistical behavior**
- Efficient & effective in highly chaotic systems [1,2]

Hybrid LSTM - MSM

$$\dot{z}_t = \begin{cases} \mathrm{LSTM}^{\mathrm{W}}(z_t, z_{t-1}, z_{t-2}, \dots) & \text{if } p_{train}(z_t) \geq \theta \\ \mathrm{MSM}^{\zeta,c}(z_t) & \text{if } p_{train}(z_t) < \theta \end{cases}$$

*Use MSM in attractor regions **underrepresented** in the training data or near attractor boundaries*

Mean Stochastic Model

$$dz_t = c\, z_t\, dt + \zeta\, dW_t$$

parameters estimated from **data**

wiener process

$$\zeta = \sqrt{-2\, c\, \sigma_z}$$

data standard deviation

$$c = \frac{1}{T}$$

decorrelation time

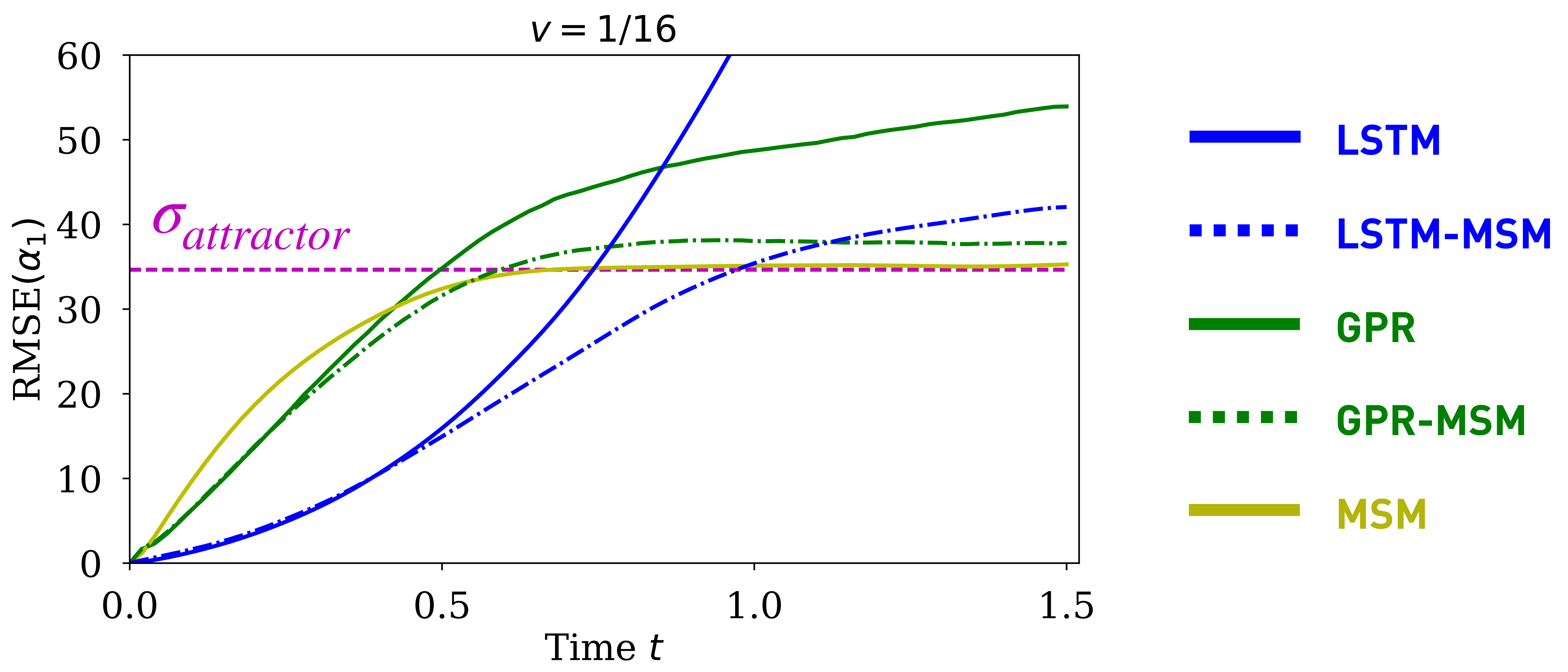
[1] **AJ Majda, J Harlim,** *Filtering complex turbulent systems,* **Cambridge University Press, 2012**
[2] **ZY Wan, TP Sapsis,** *Reduced-space Gaussian Process Regression for data-driven probabilistic forecast of chaotic dynamical systems,* **Physica D: Nonlinear Phenomena, 2017**

Results on KS - *Comparison with Gaussian Process Regression (GPR)*

V	Total number of initial conditions **(IC)**
k	Mode number
i	IC index
z_k^i	**True** state of mode k starting from **IC** i
$\tilde{z}_k^i$	**Predicted** state of mode k starting from **IC** i

Root mean square error:

$$\text{RMSE}(z_k) = \sqrt{\frac{1}{V}\sum_{i=1}^{V}\left(z_k^i - \tilde{z}_k^i\right)^2}$$



RMSE evolution in time of **the most energetic** mode
(averaged over 1000 initial conditions)

PR Vlachas, W Byeon, Z Wan, T Sapsis, P Koumoutsakos, *Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks,* **Proc. Roy. Soc. A , 2018**

Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

- Dynamics underrepresented in training data
- Scarce data in attractor boundaries
- Under-resolved high dimensional dynamics
- Models not generalising / distribution shift

Mitigation? Hybrid LSTM - MSM approach

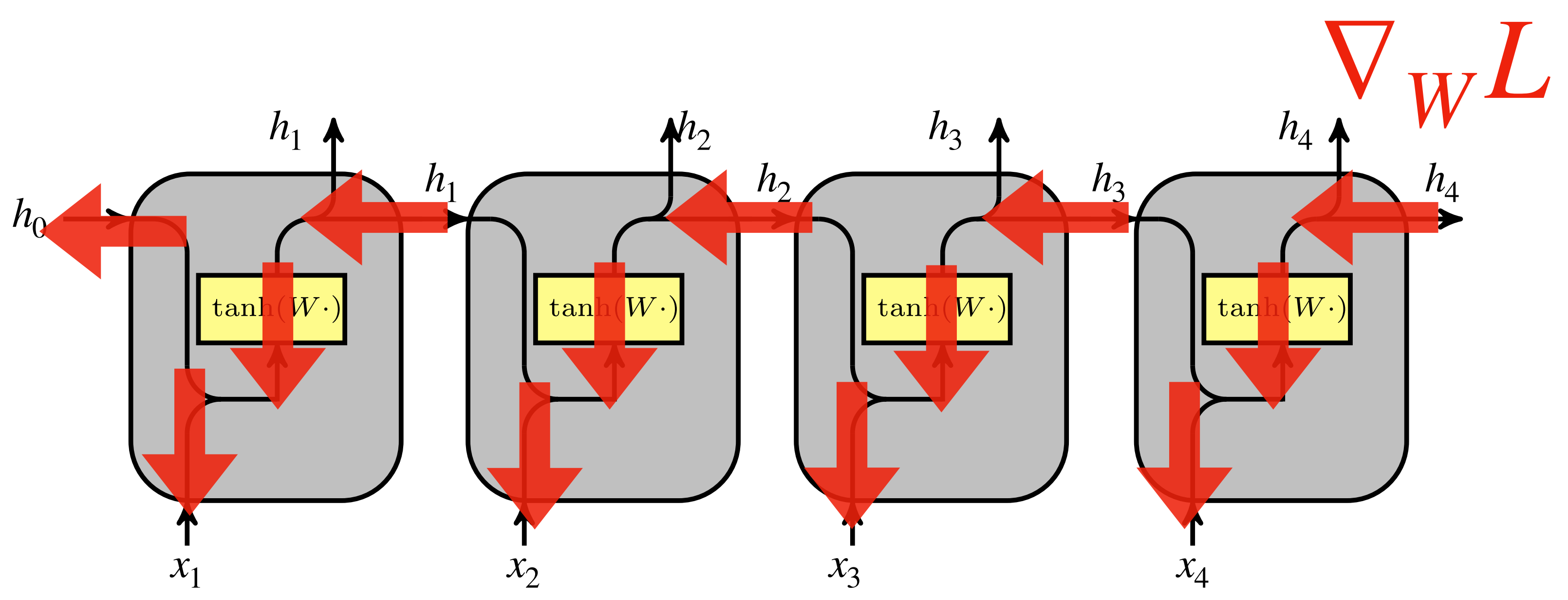
Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

- Dynamics underrepresented in training data
- Scarce data in attractor boundaries
- Under-resolved high dimensional dynamics
- Models not generalising / distribution shift

Mitigation? Hybrid LSTM - MSM approach

2 **Vanishing gradients problem during training:** As the gradient is back-propagated during training of the networks it may **vanish to zero or explode.**



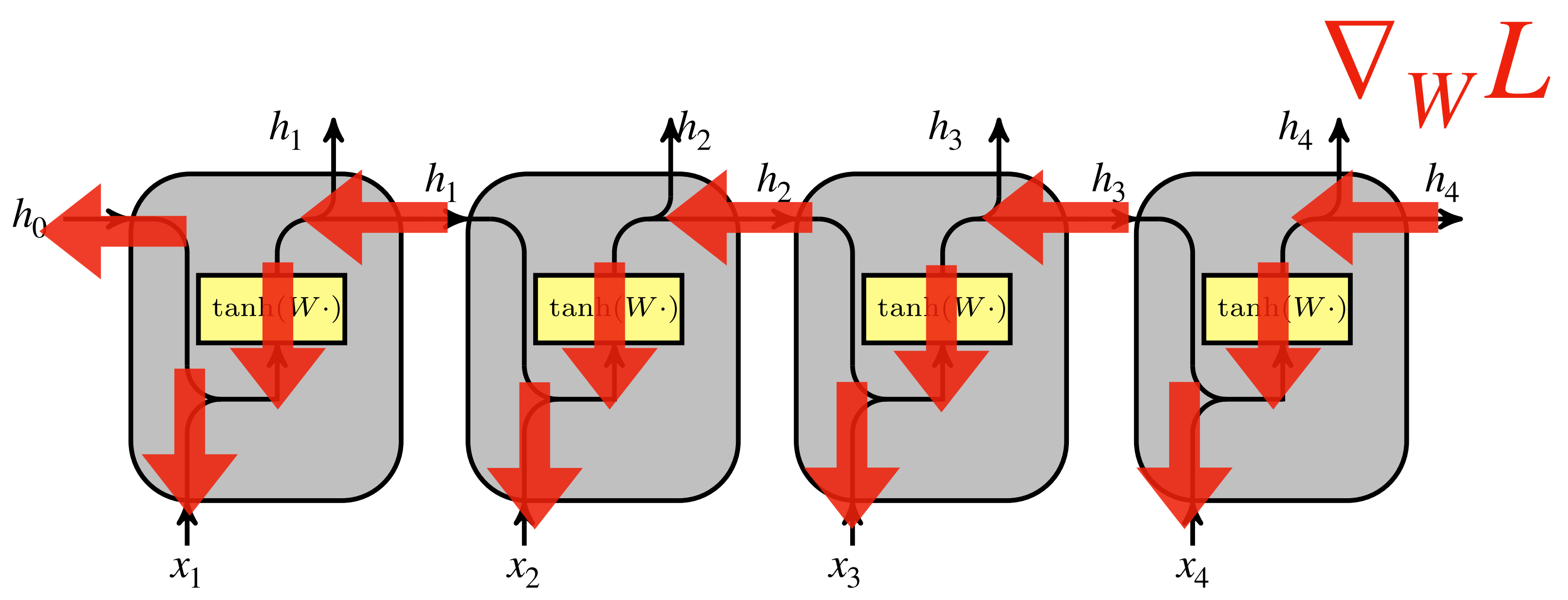
Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

- Dynamics underrepresented in training data
- Scarce data in attractor boundaries
- Under-resolved high dimensional dynamics
- Models not generalising / distribution shift

Mitigation? Hybrid LSTM - MSM approach

2 **Vanishing gradients problem during training:** As the gradient is back-propagated during training of the networks it may **vanish to zero or explode.**



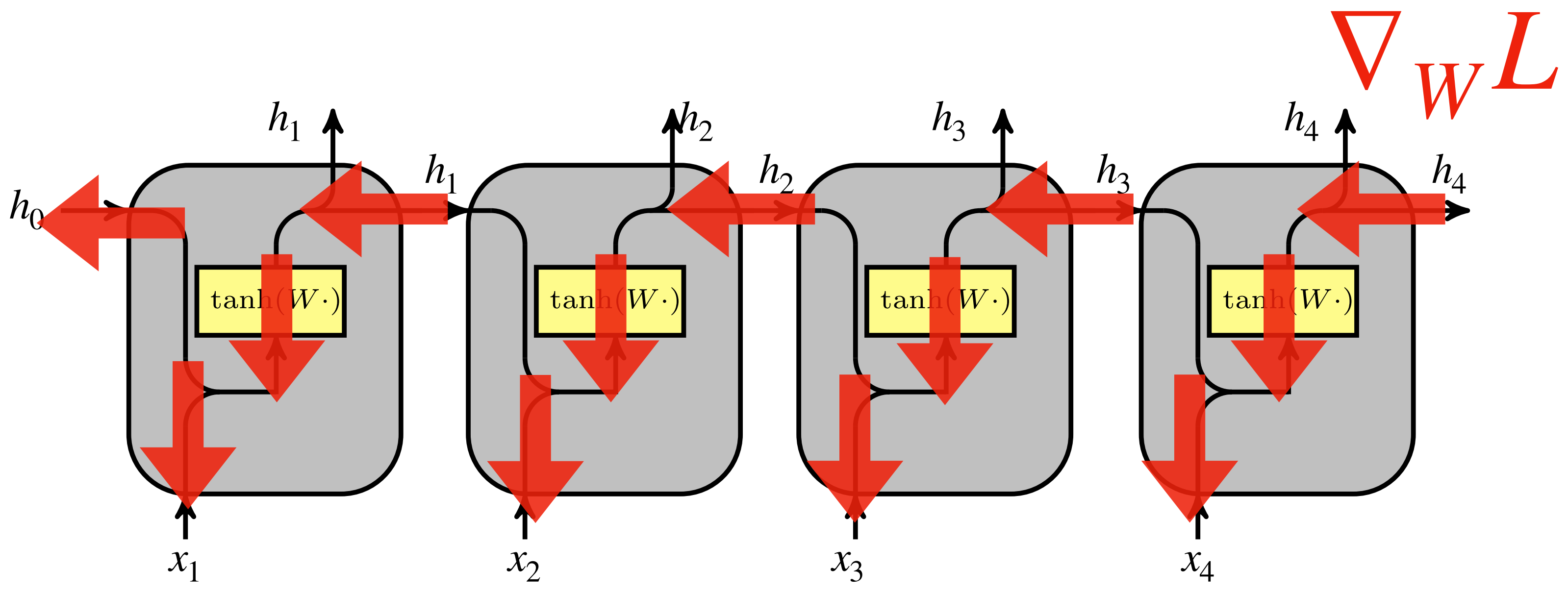
Challenges

1 Iterative prediction error **accumulates** leading to unphysical predictions
- divergence from attractor

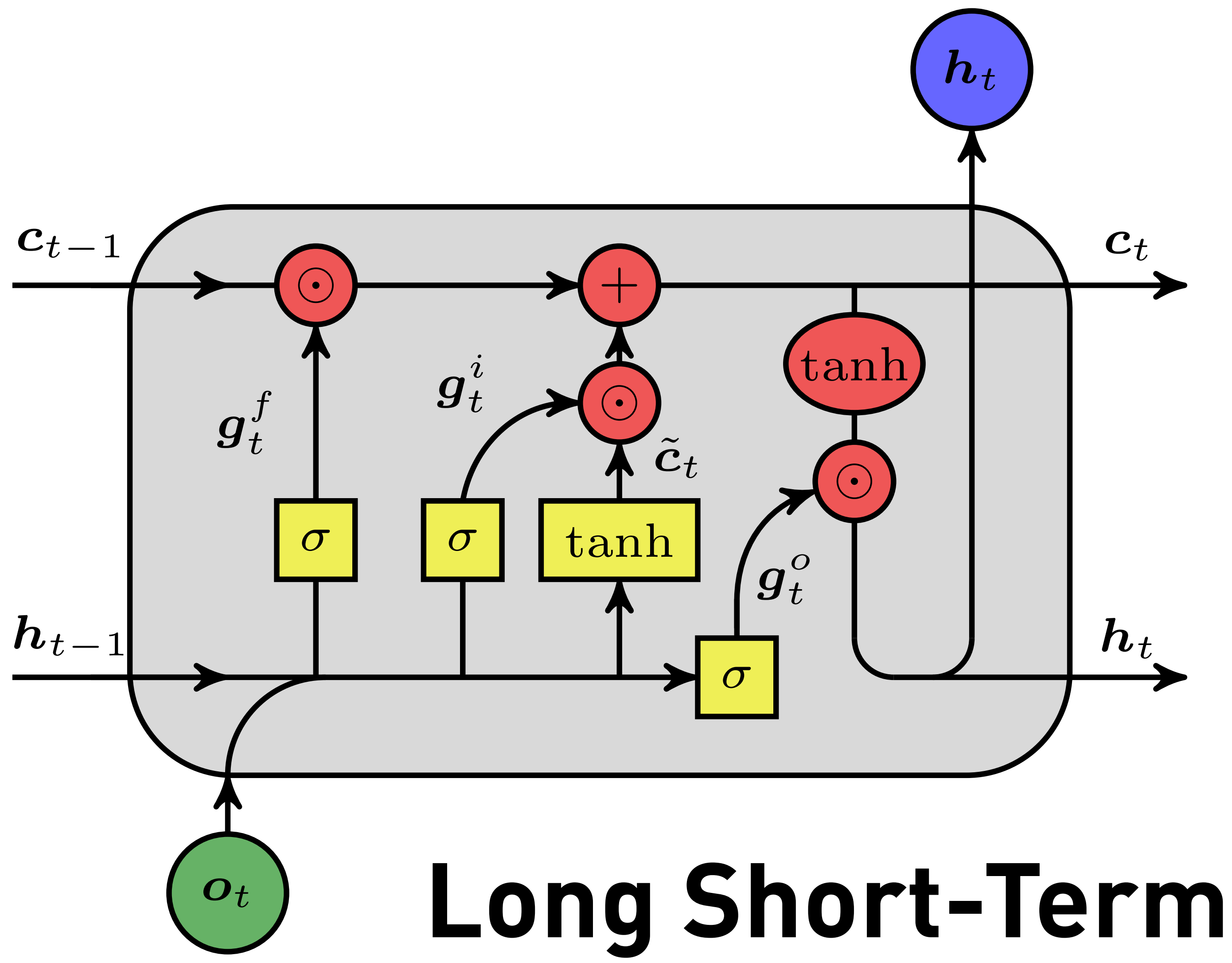
- Dynamics underrepresented in training data
- Scarce data in attractor boundaries
- Under-resolved high dimensional dynamics
- Models not generalising / distribution shift

Mitigation? Hybrid LSTM - MSM approach

2 **Vanishing gradients problem during training:** As the gradient is back-propagated during training of the networks it may **vanish to zero or explode.**

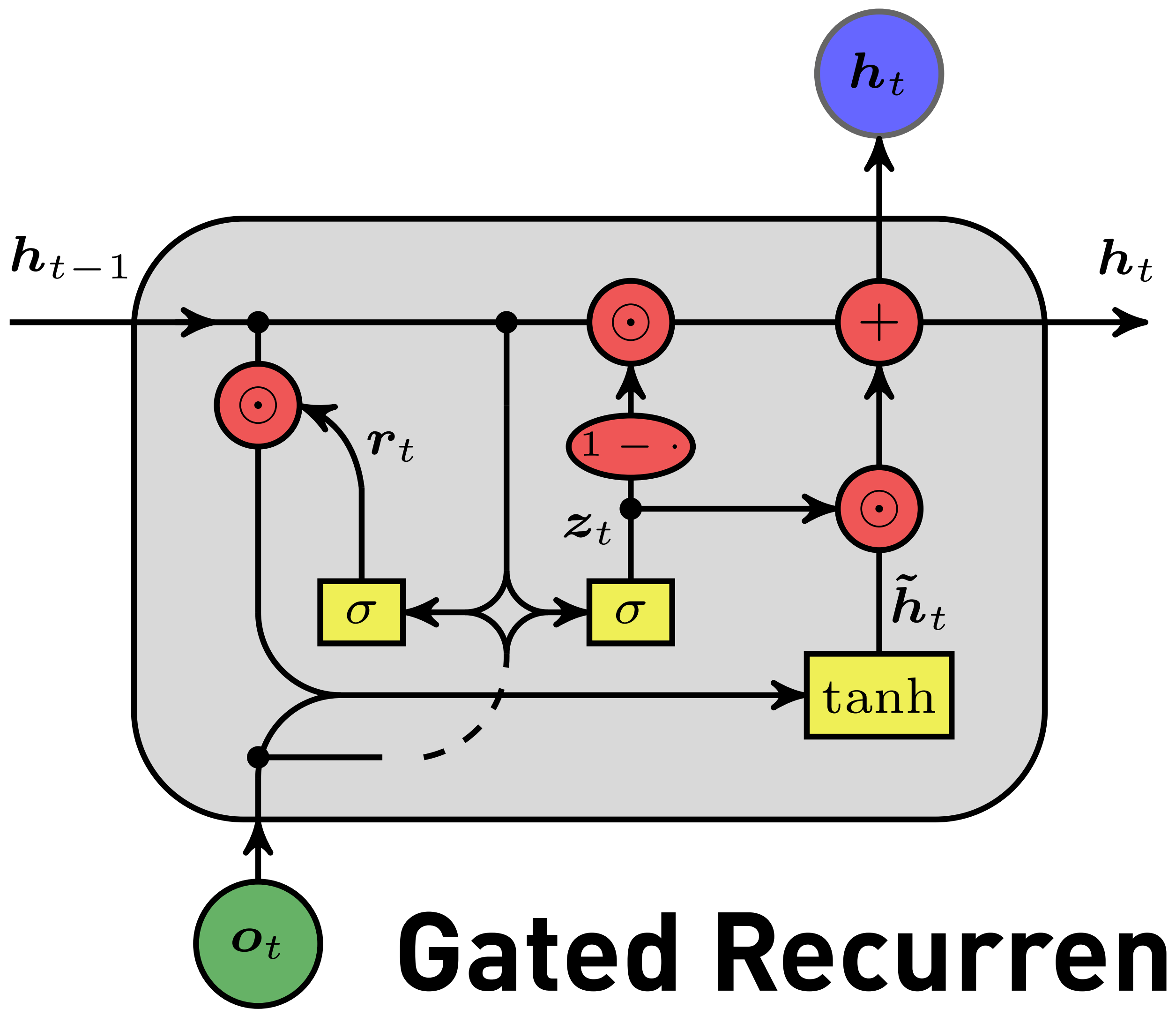


Mitigation? Sophisticated architectures



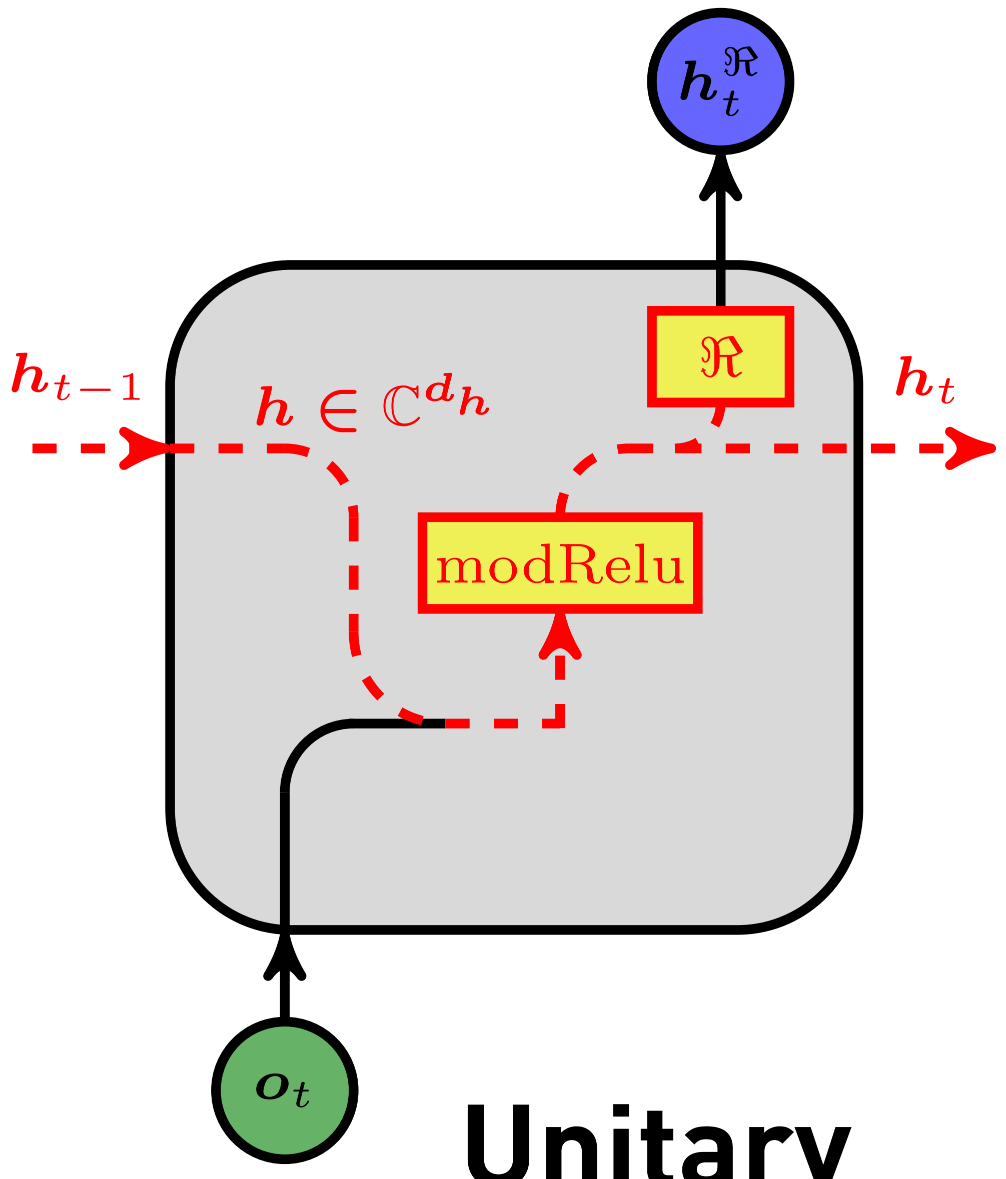
Long Short-Term Memory (LSTM)

- Gating mechanisms
- Proposed by S. Hochreiter and J. Schmidhuber (1997)
- **Trained with Backpropagation through time (BPTT)**



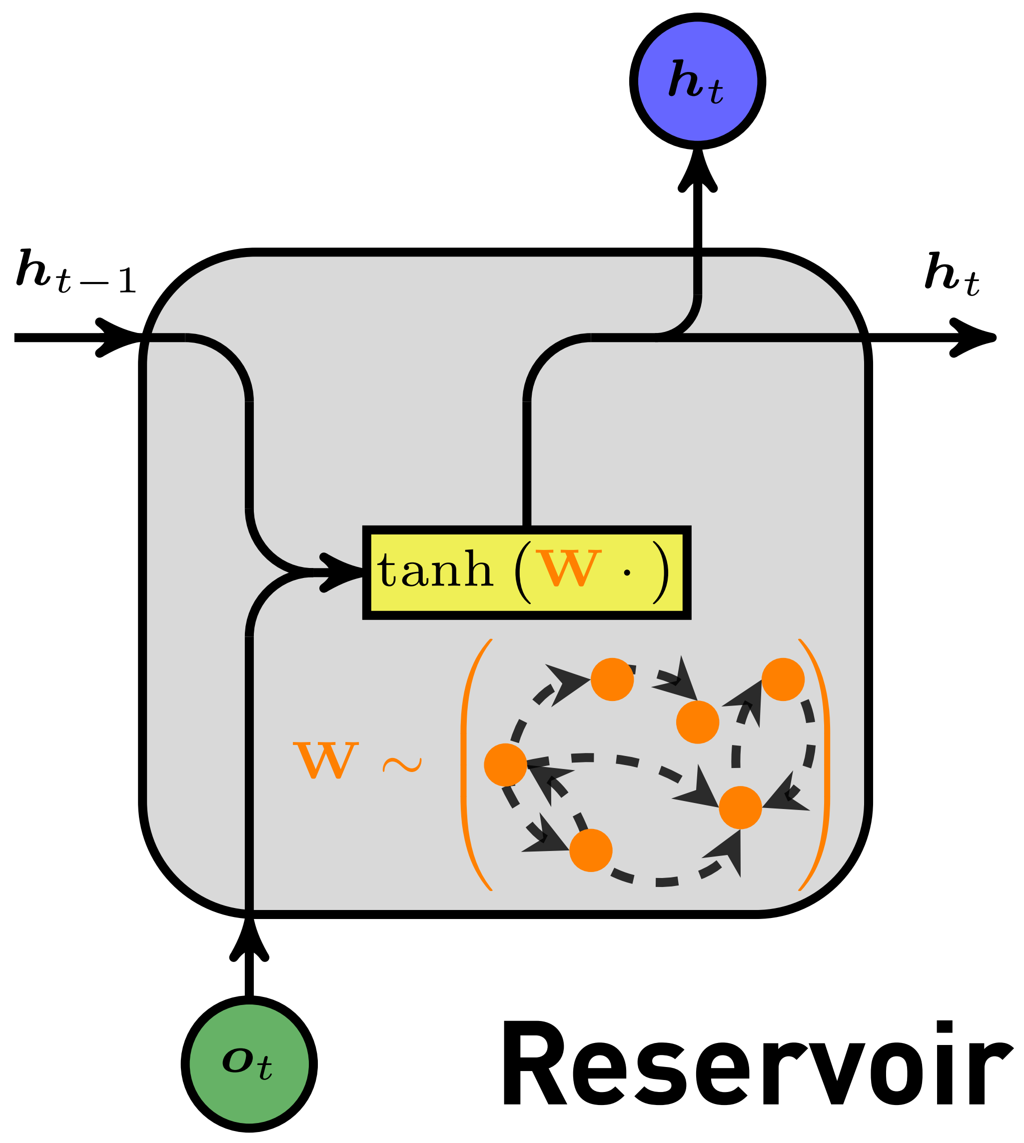
Gated Recurrent Unit (GRU)

- Gating mechanisms
- Alternative to LSTM, less params., Cho et al. (2014)
- Comparable with LSTM polyphonic/speech data, Chung et al. (2014)
- **Trained with BPTT**



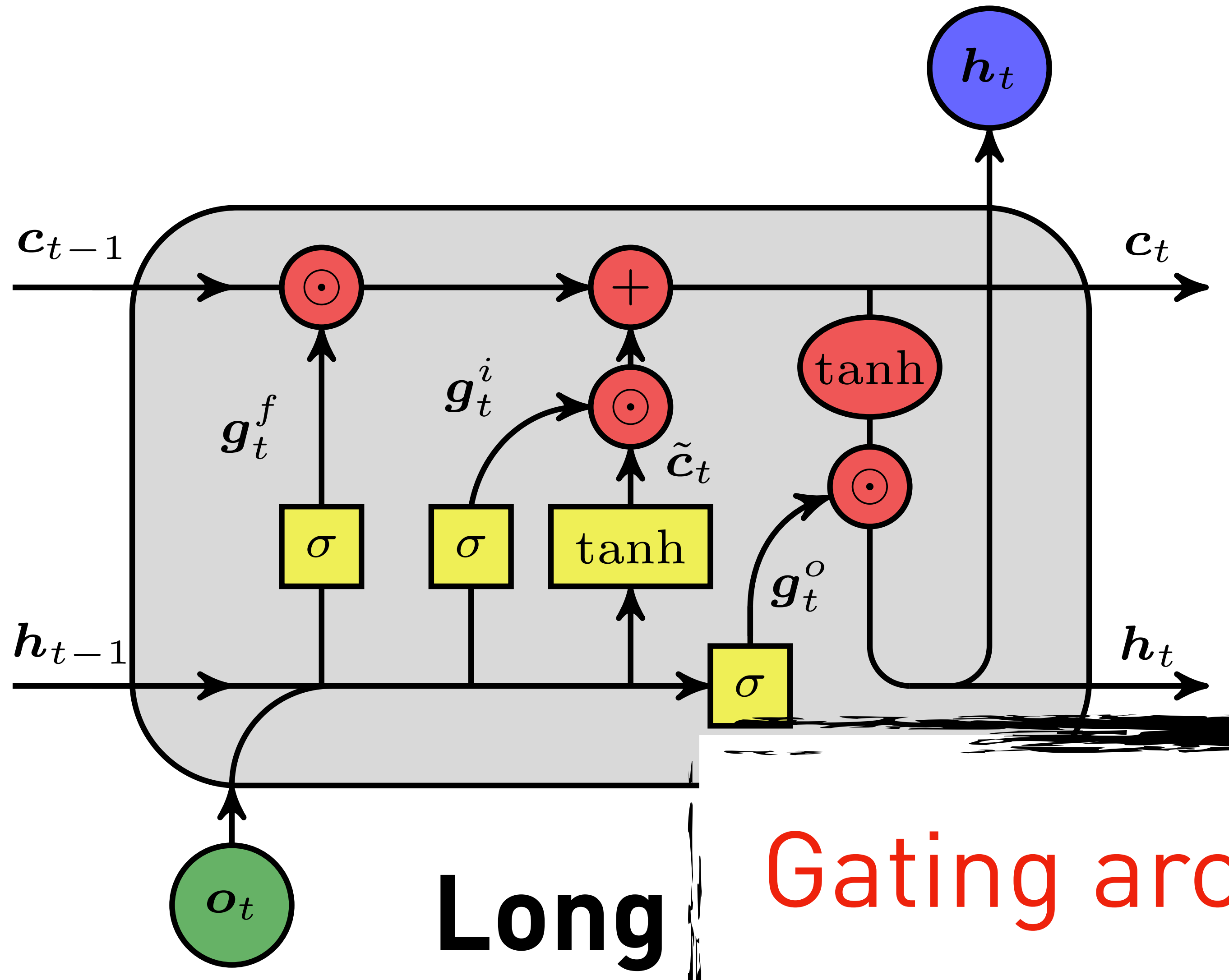
Unitary

- Arjovsky et al. (2016); Jing et al. (2017)
- Recurrent weight matrix is complex unitary with spectral radius one
- Complex generalization of Elman RNNs
- modRelu as a activation
- **Trained with BPTT**



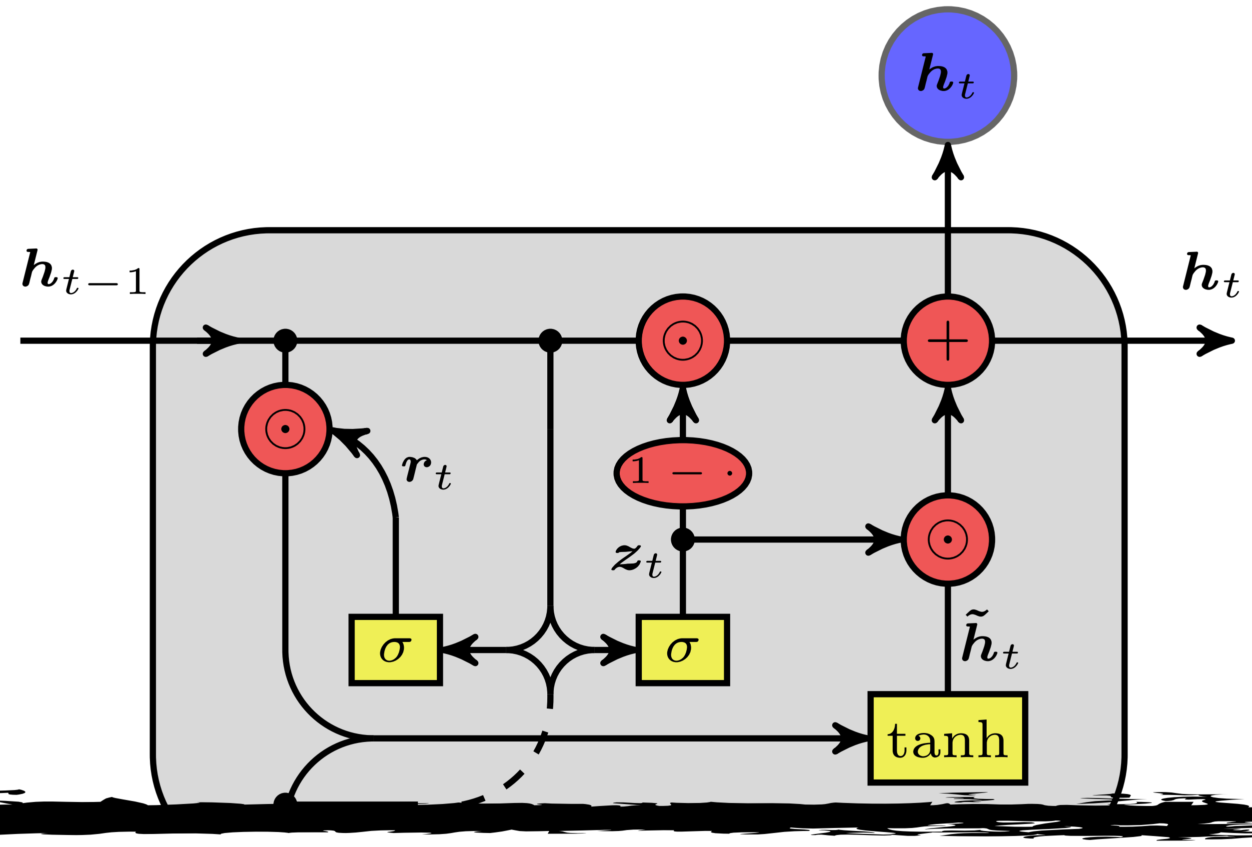
Reservoir Computer (RC)

- Pathak, Ott, et. al. (2017, 2018)
- Echo state networks, Liquid state machines, Maass et. al. (2002), Jaeger et. al. (2007)
- **Random** ***sparse*** recurrent weight matrix with ***spectral radius smaller than one***
- **Train linear output layer with regularised least squares regression**



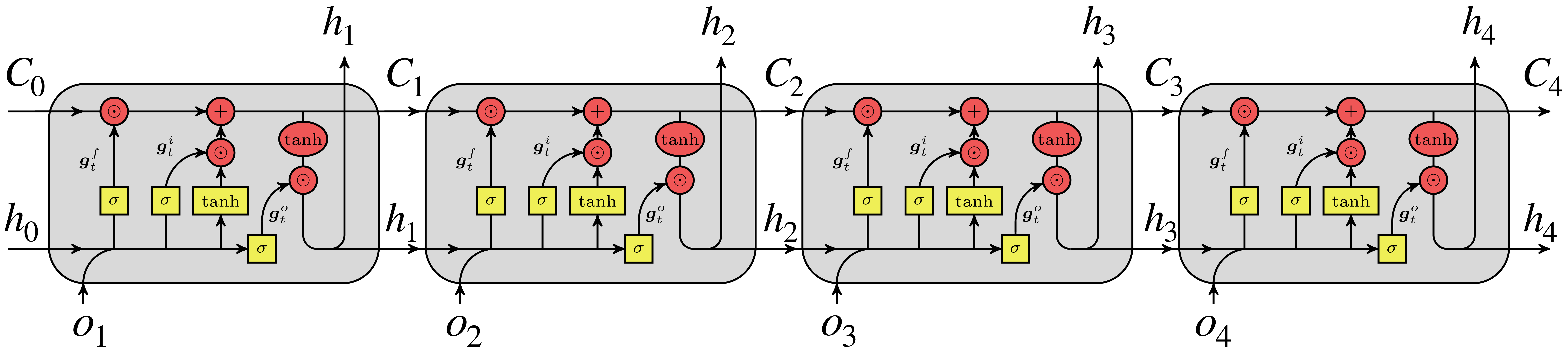
- Gating mechanisms
- Proposed by S. Hochreiter and J. Schmidhuber (1997)
- **Trained with**

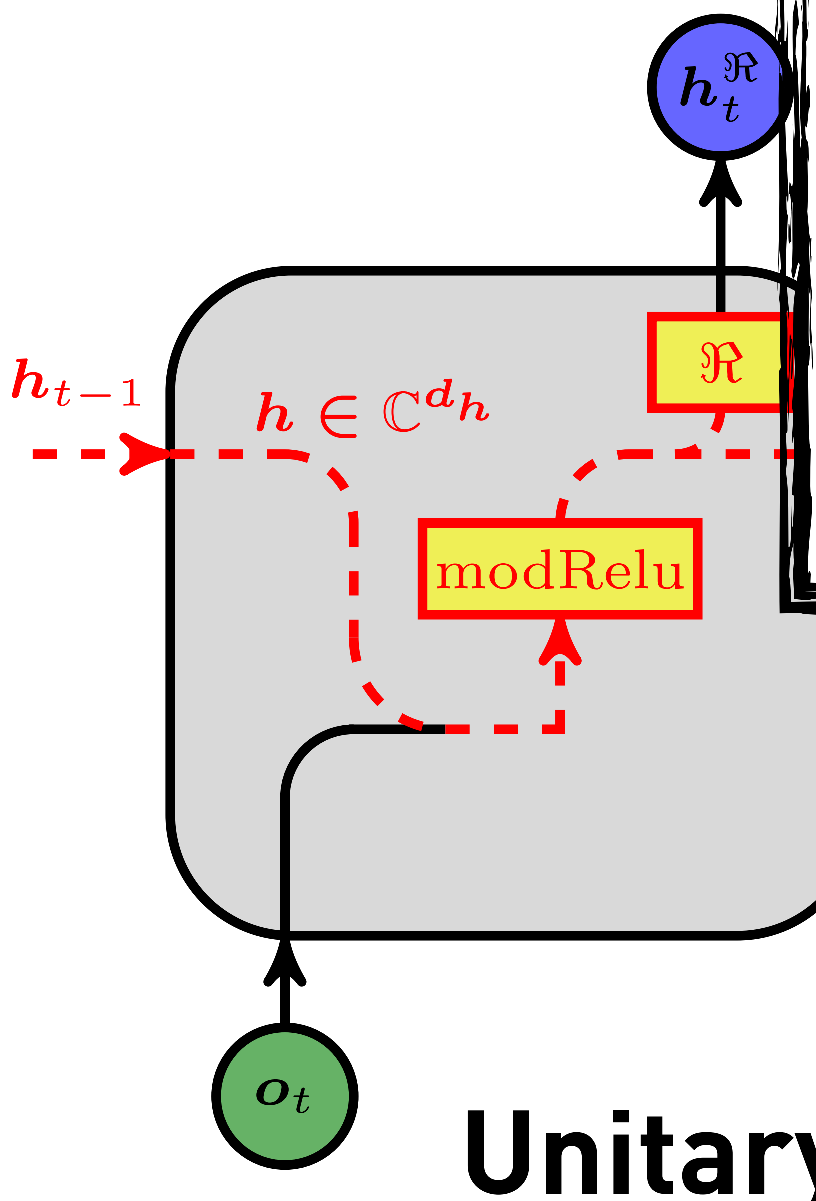
Long
Mem



- Gating mechanisms
- Alternative to LSTM, less params., Cho et al. (2014)
- Comparable with LSTM polyphonic/speech data

Gating architectures: ***Uninterrupted gradient flow !***

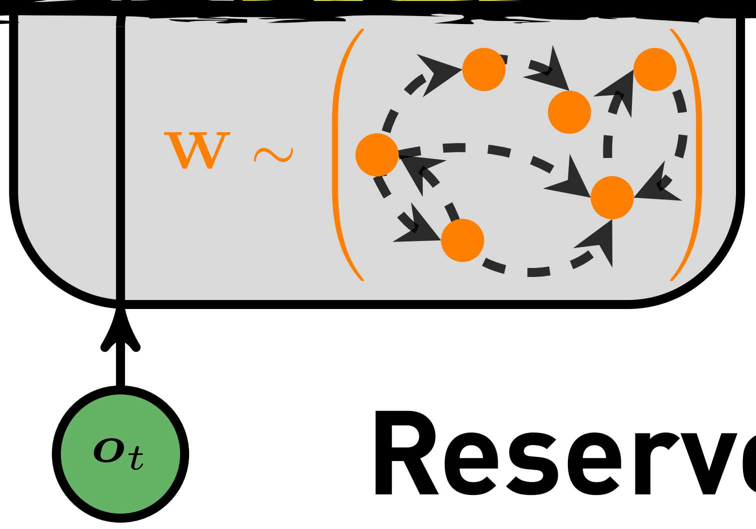




Unitary

spectral radius one

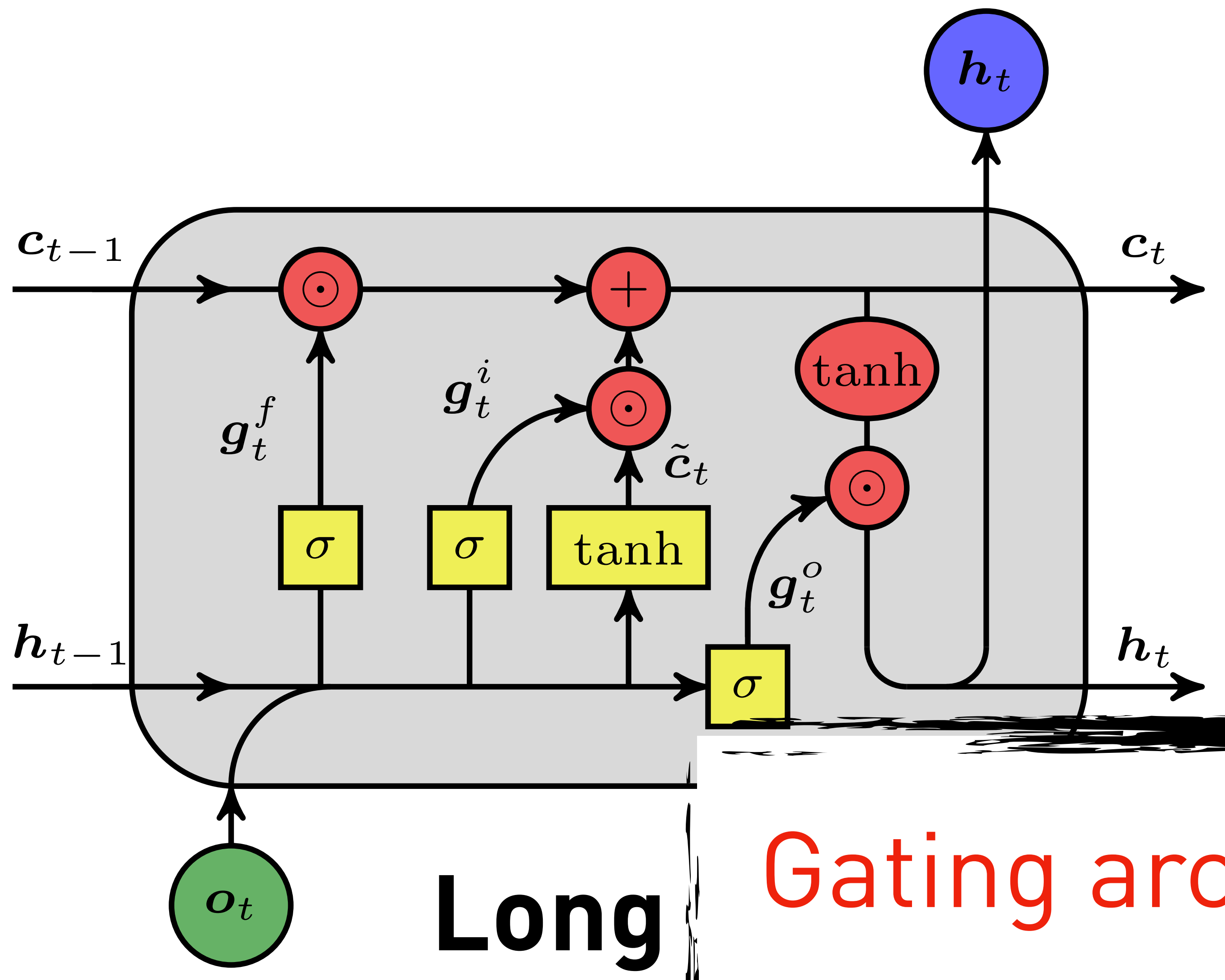
- Complex generalization of Elman RNNs
- modRelu as a activation
- **Trained with BPTT**



Reservoir Computer (RC)

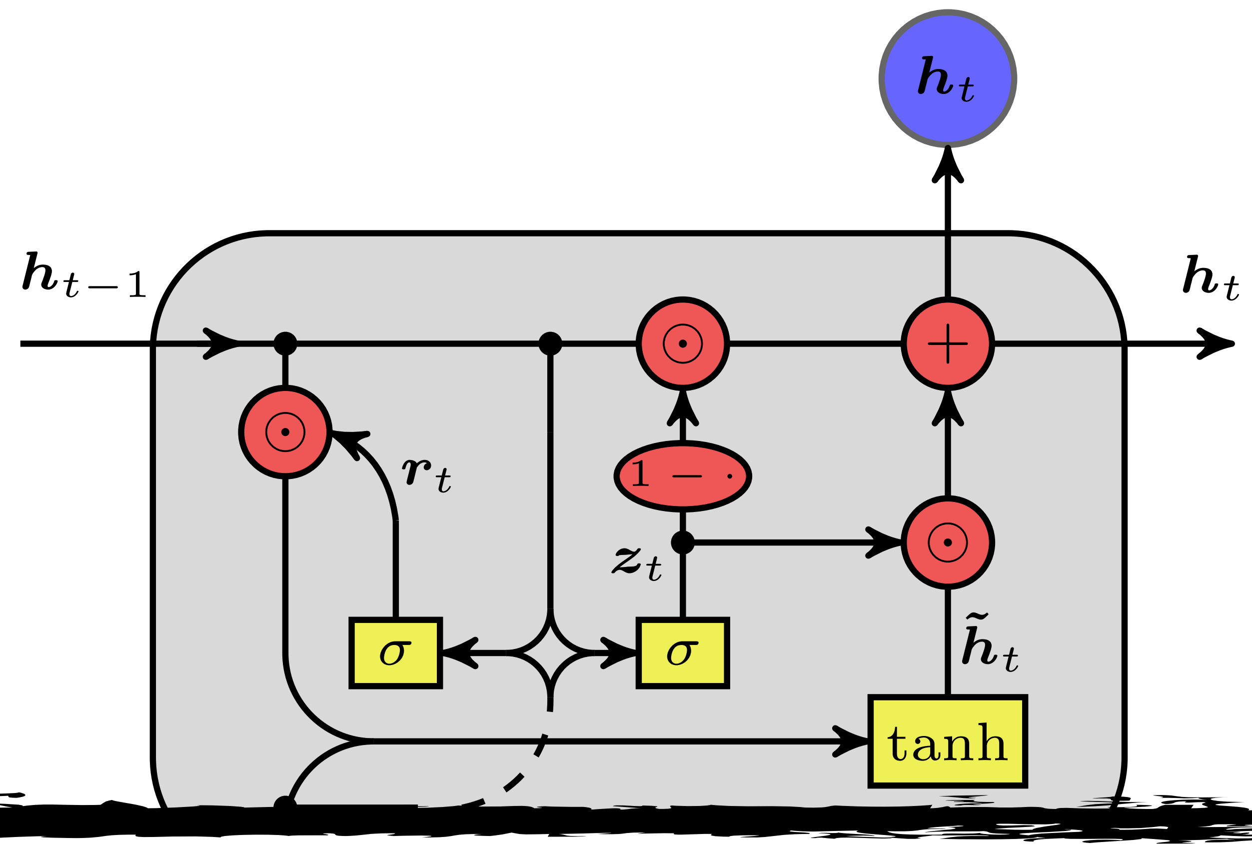
weight matrix with ***spectral radius smaller than one***

- **Train linear output layer with regularised least squares regression**



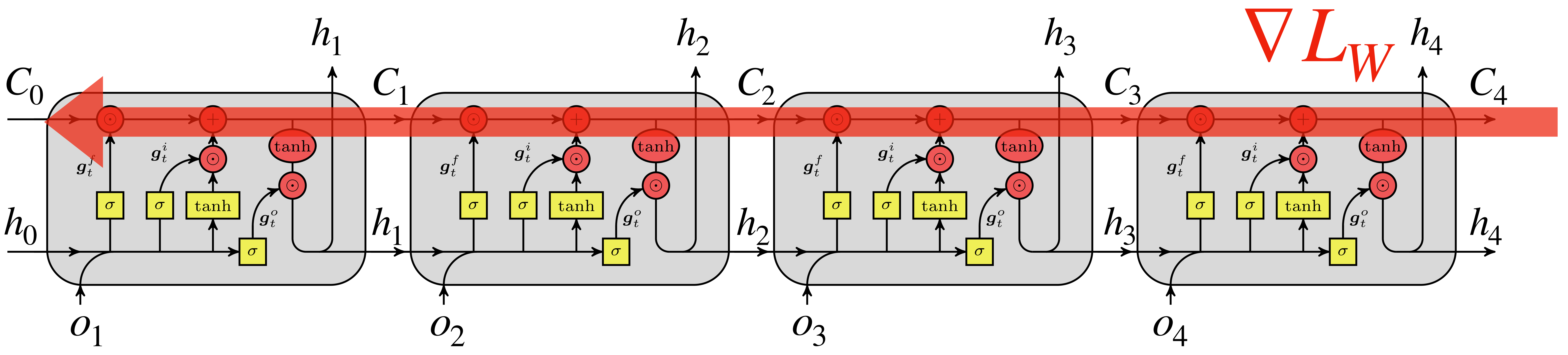
- Gating mechanisms
- Proposed by S. Hochreiter and J. Schmidhuber (1997)
- **Trained with**

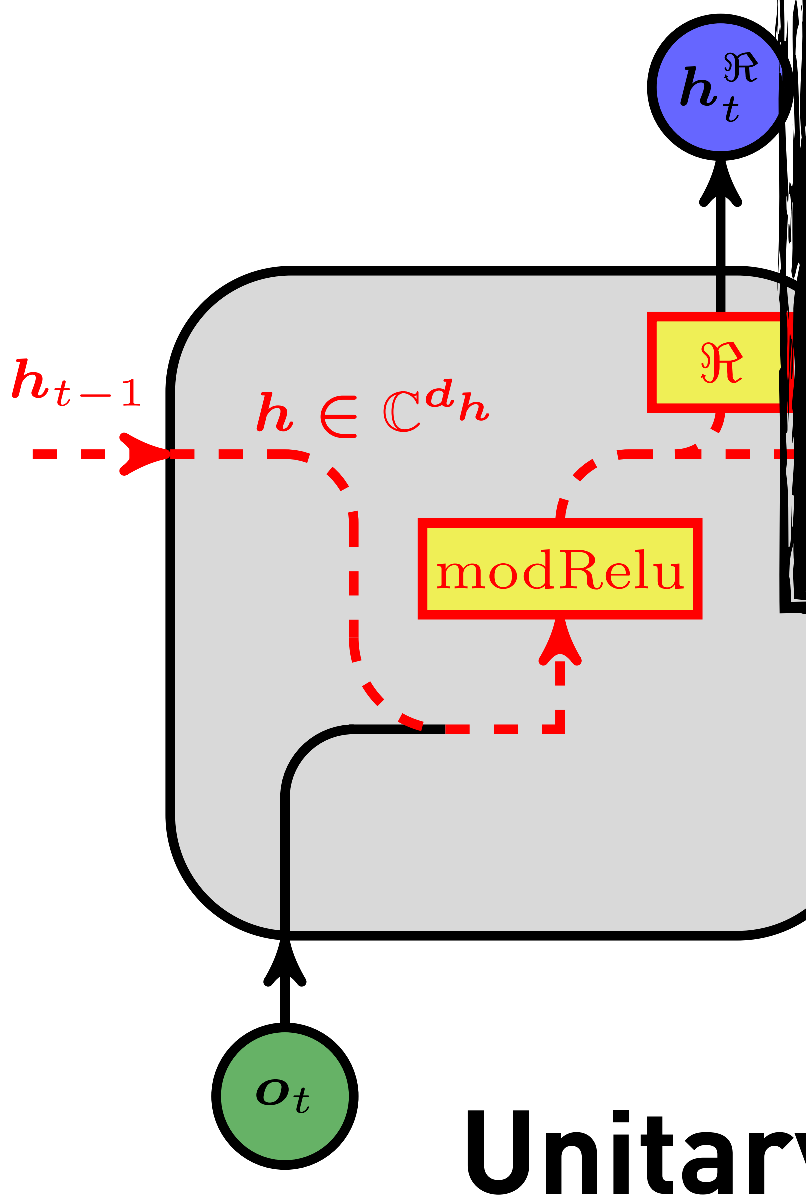
Long
Mem



- Gating mechanisms
- Alternative to LSTM, less params., Cho et al. (2014)
- Comparable with LSTM polyphonic/speech data

Gating architectures: ***Uninterrupted gradient flow !***

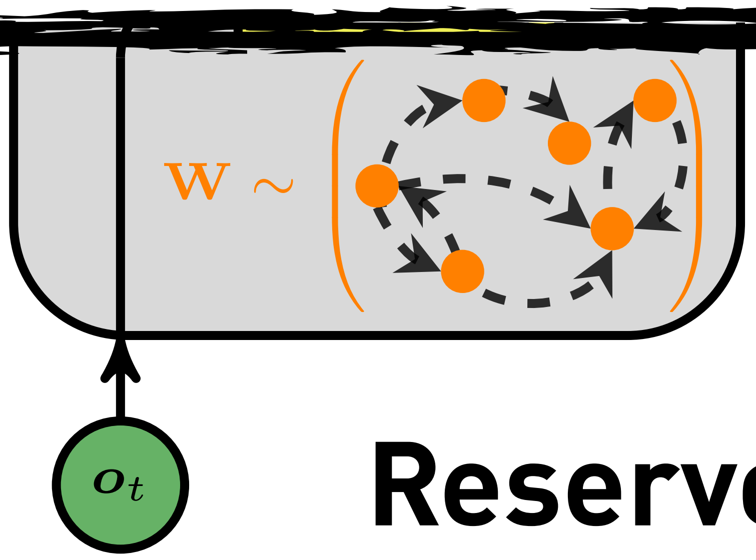




spectral radius one

- Complex generalization of Elman RNNs
- modRelu as a activation
- **Trained with BPTT**

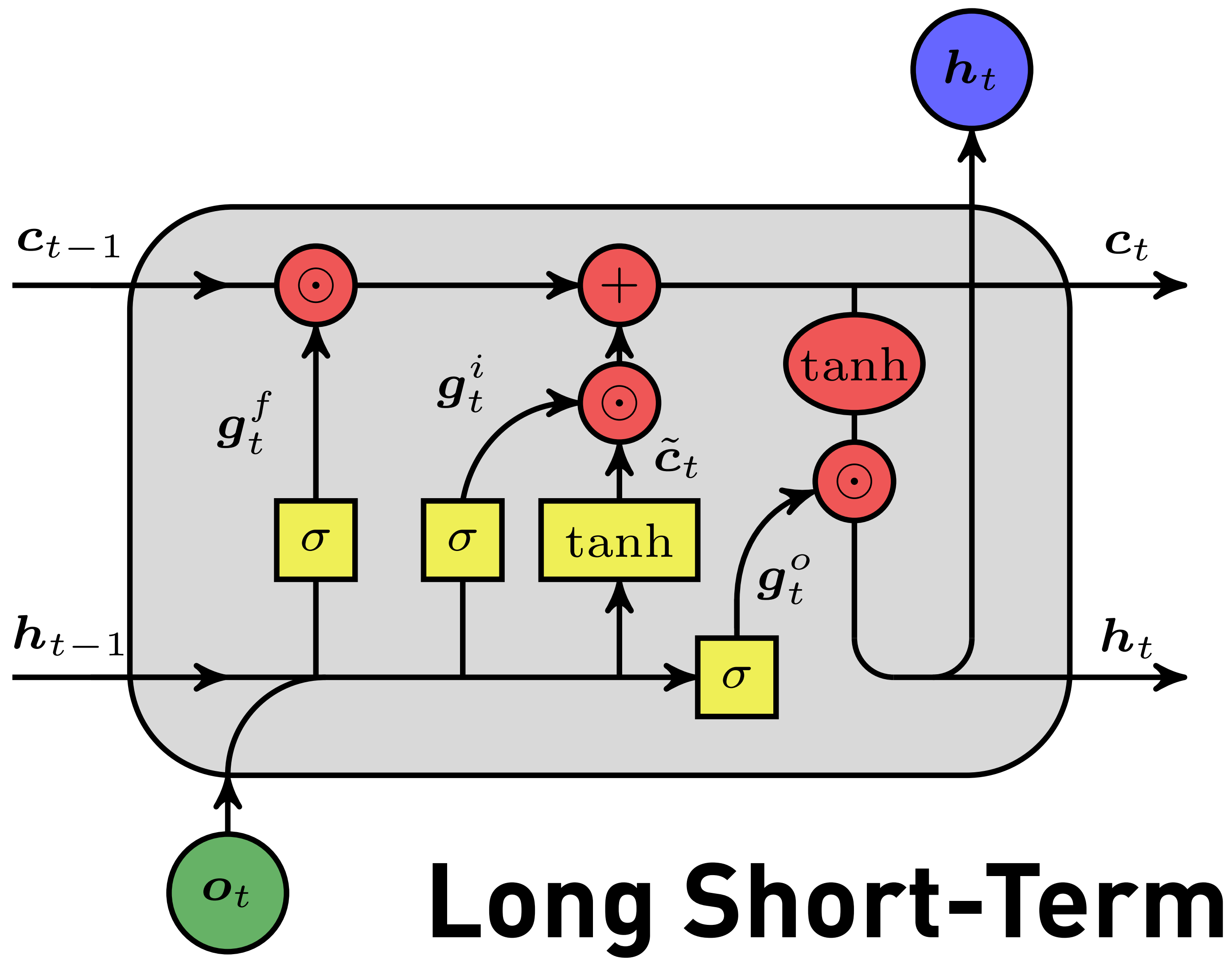
Unitary



weight matrix with ***spectral radius smaller than one***

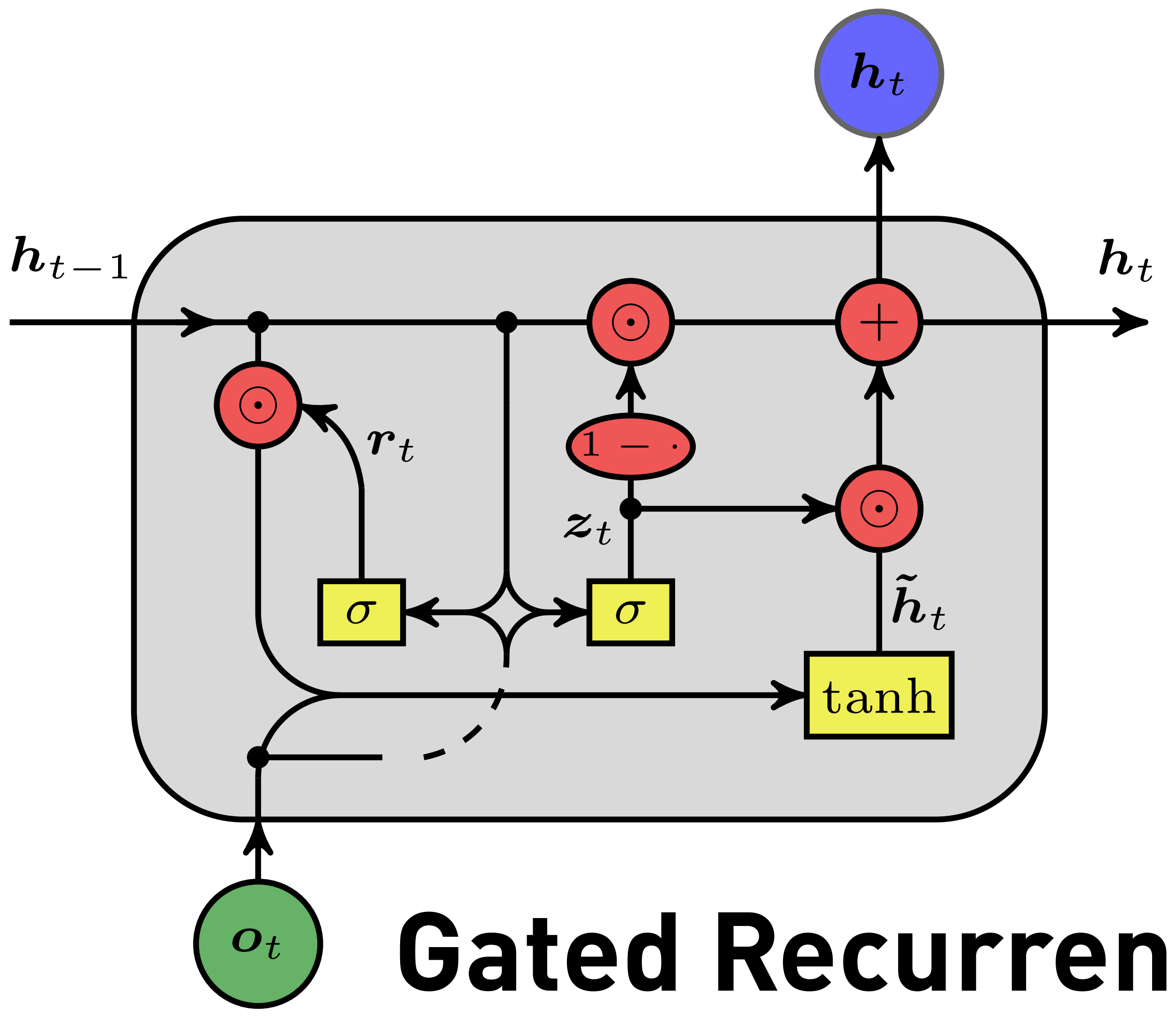
- **Train linear output layer with regularised least squares regression**

Reservoir Computer (RC)



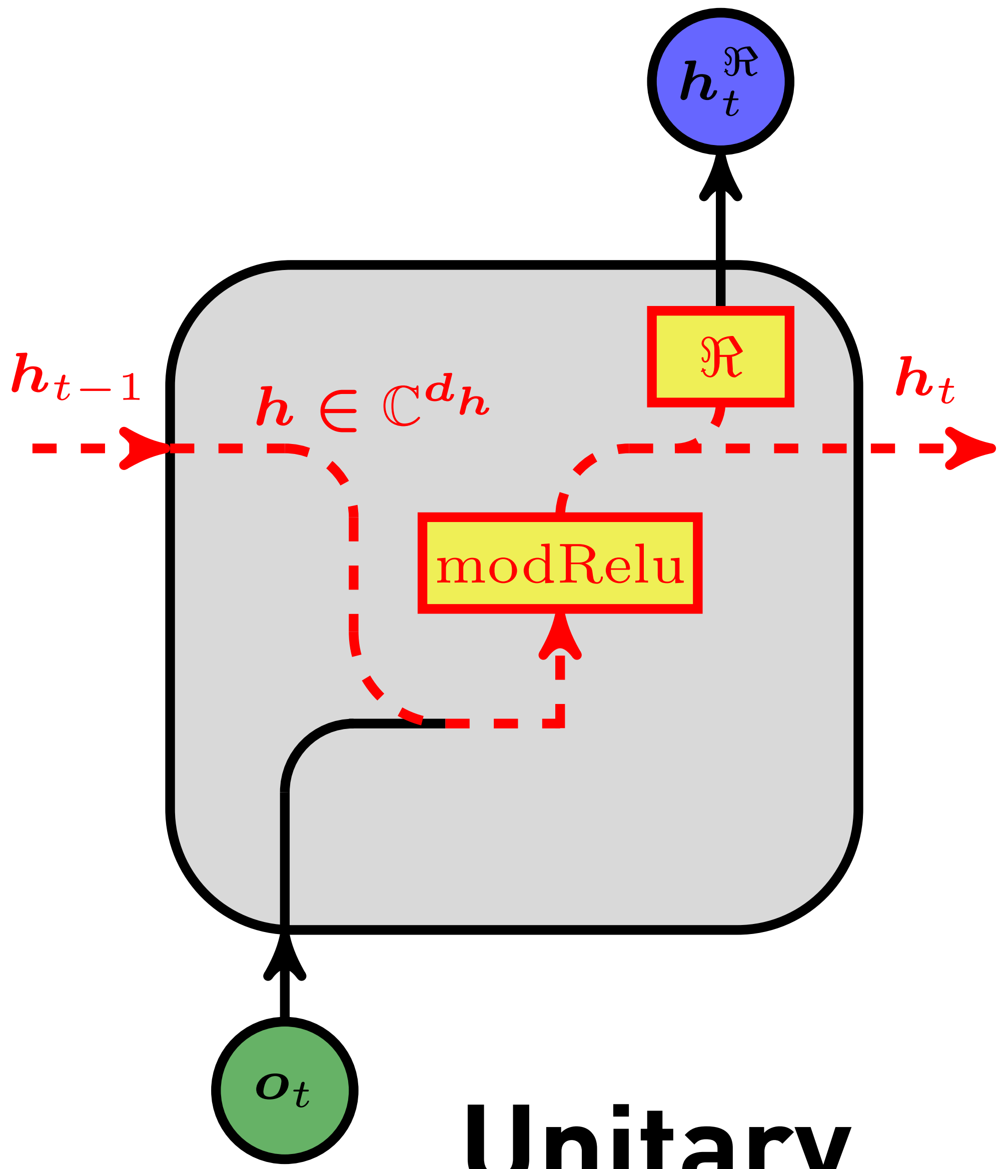
Long Short-Term Memory (LSTM)

- Gating mechanisms
- Proposed by S. Hochreiter and J. Schmidhuber (1997)
- **Trained with Backpropagation through time (BPTT)**



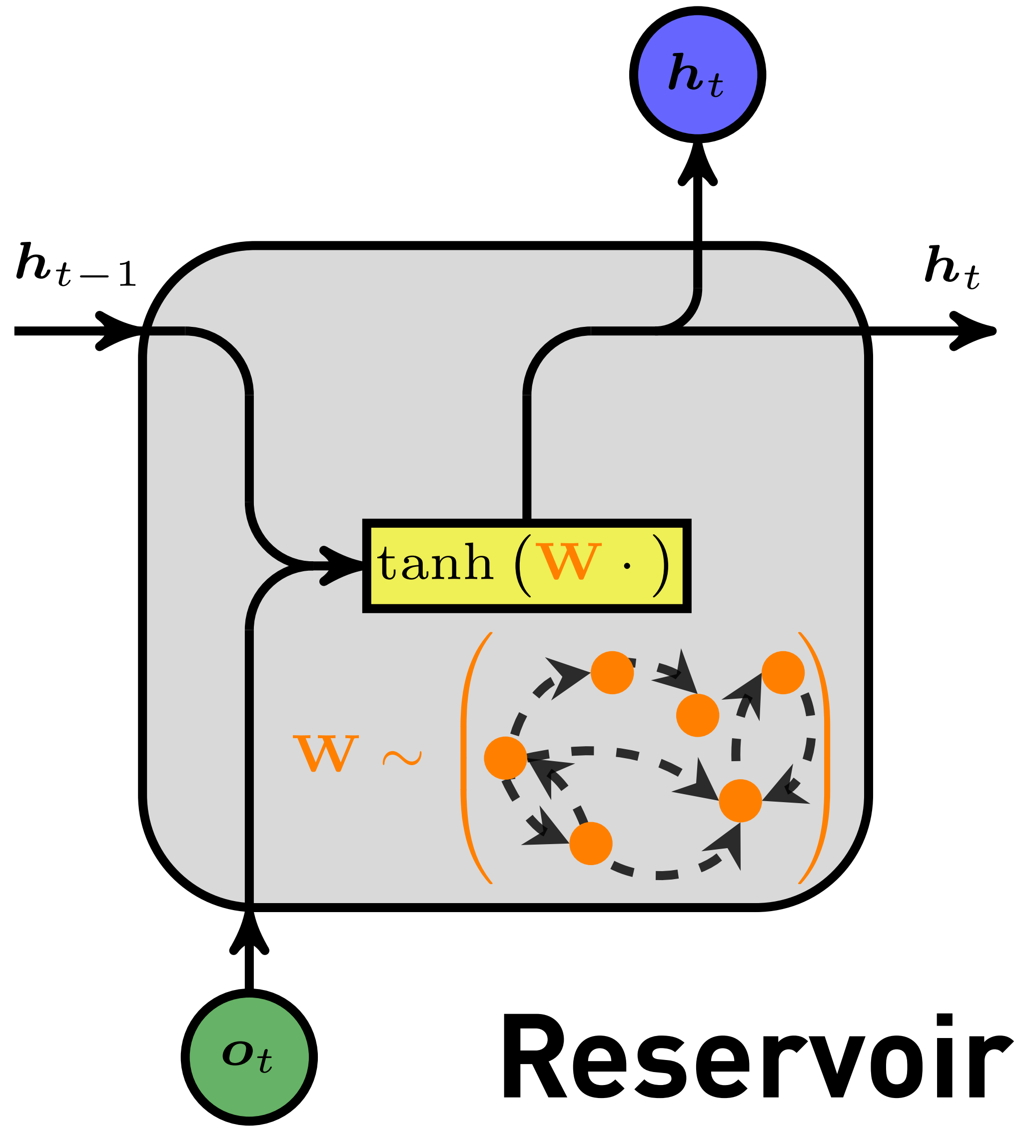
Gated Recurrent Unit (GRU)

- Gating mechanisms
- Alternative to LSTM, less params., Cho et al. (2014)
- Comparable with LSTM polyphonic/speech data, Chung et al. (2014)
- **Trained with BPTT**



Unitary

- Arjovsky et al. (2016); Jing et al. (2017)
- Recurrent weight matrix is complex unitary with spectral radius one
- Complex generalization of Elman RNNs
- modRelu as a activation
- **Trained with BPTT**



Reservoir Computer (RC)

- Pathak, Ott, et. al. (2017, 2018)
- Echo state networks, Liquid state machines, Maass et. al. (2002), Jaeger et. al. (2007)
- **Random** ***sparse*** recurrent weight matrix with ***spectral radius smaller than one***
- **Train linear output layer with regularised least squares regression**

Lorenz 96 - 35 / 40 mode observable

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **JNN, 2020**

Lorenz 96 - 35 / 40 mode observable

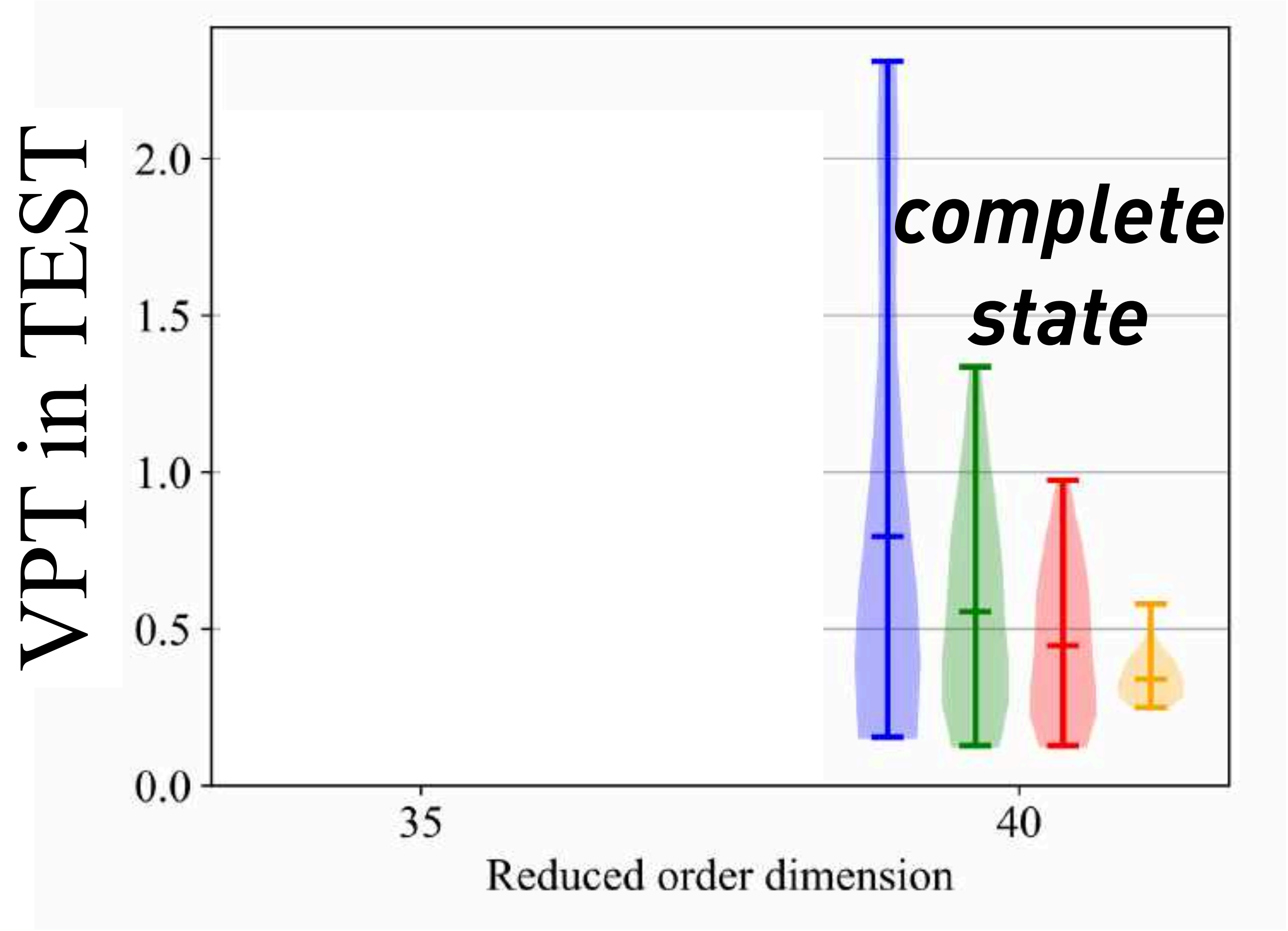
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **JNN, 2020**

- Valid prediction time $\mathrm{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\operatorname{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)

Lorenz 96 - 35 / 40 mode observable

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **JNN, 2020**

- Valid prediction time $\mathrm{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\operatorname{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)

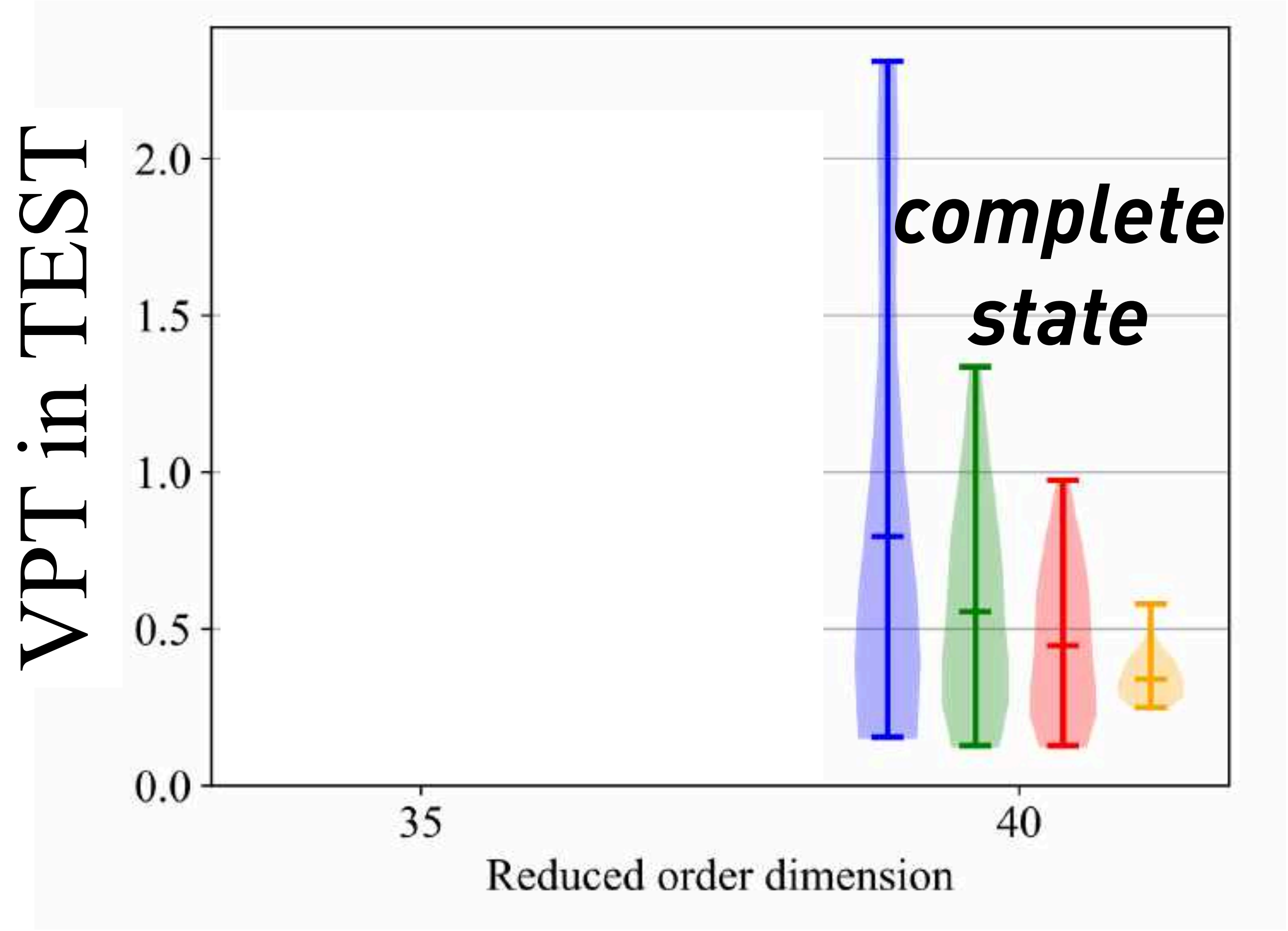


RC (or □) ; GRU (or ✕) ; LSTM (or ○) ; Unit (or ◇) ; Ideal ;

Lorenz 96 - 35 / 40 mode observable

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **JNN, 2020**

- Valid prediction time $\mathrm{VPT} = \dfrac{1}{T_{\Lambda_1}} \underset{t_f}{\operatorname{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)



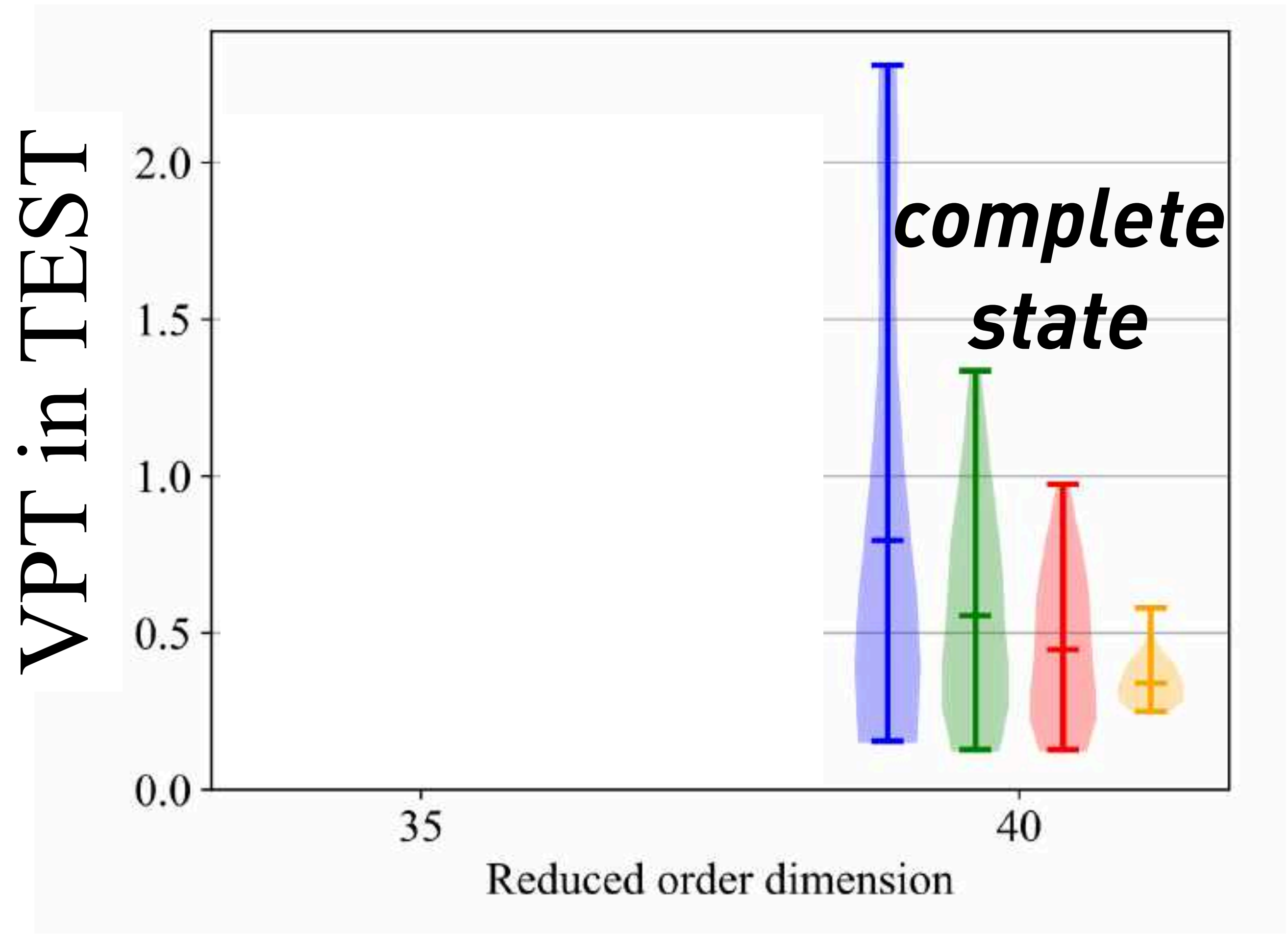
RC (or □) ; GRU (or ✕) ; LSTM (or ○) ; Unit (or ◇) ; Ideal ;

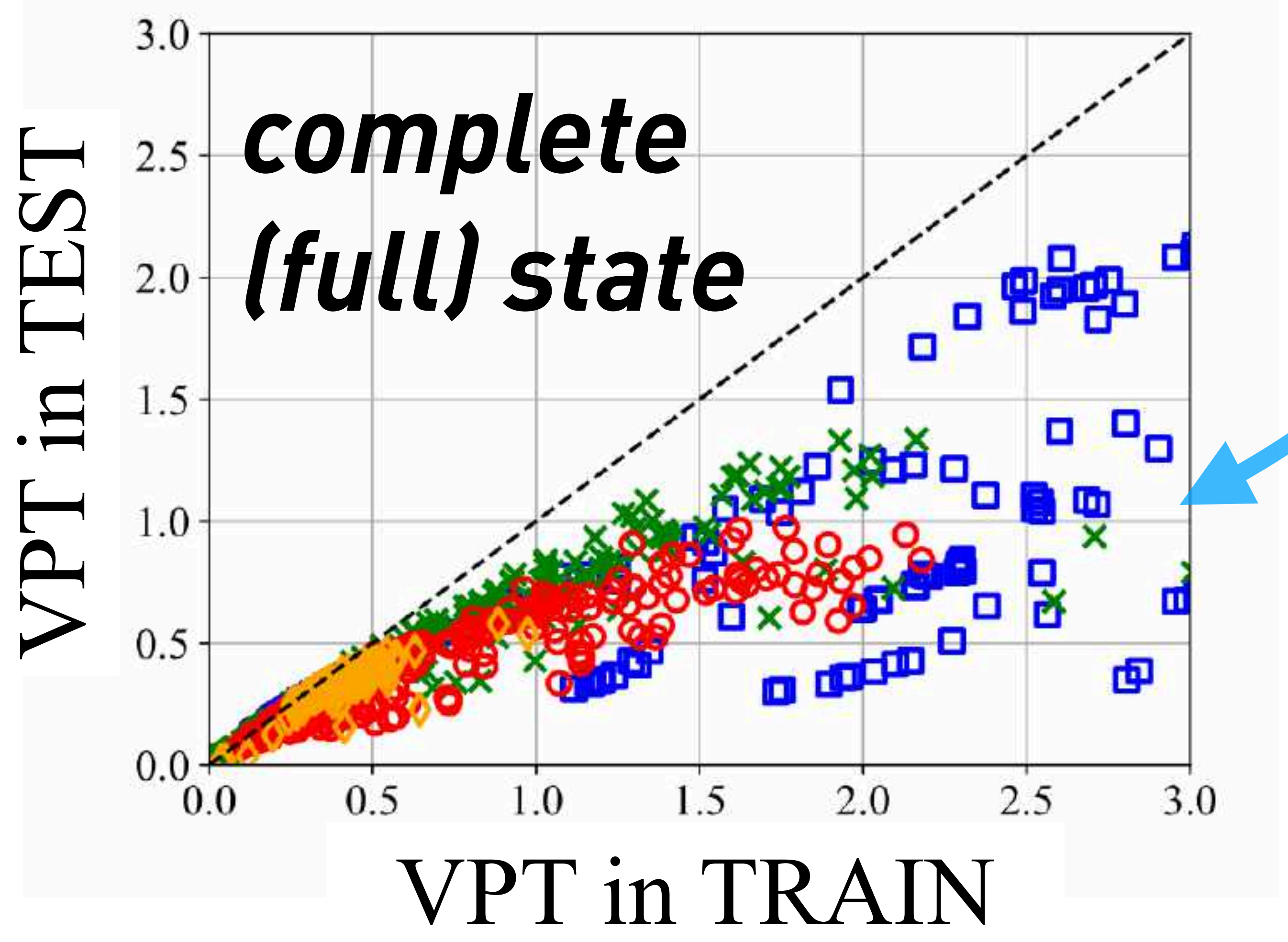
Lorenz 96 - 35 / 40 mode observable

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **JNN, 2020**

- Valid prediction time $\mathrm{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\operatorname{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)

every marker is a trained model !





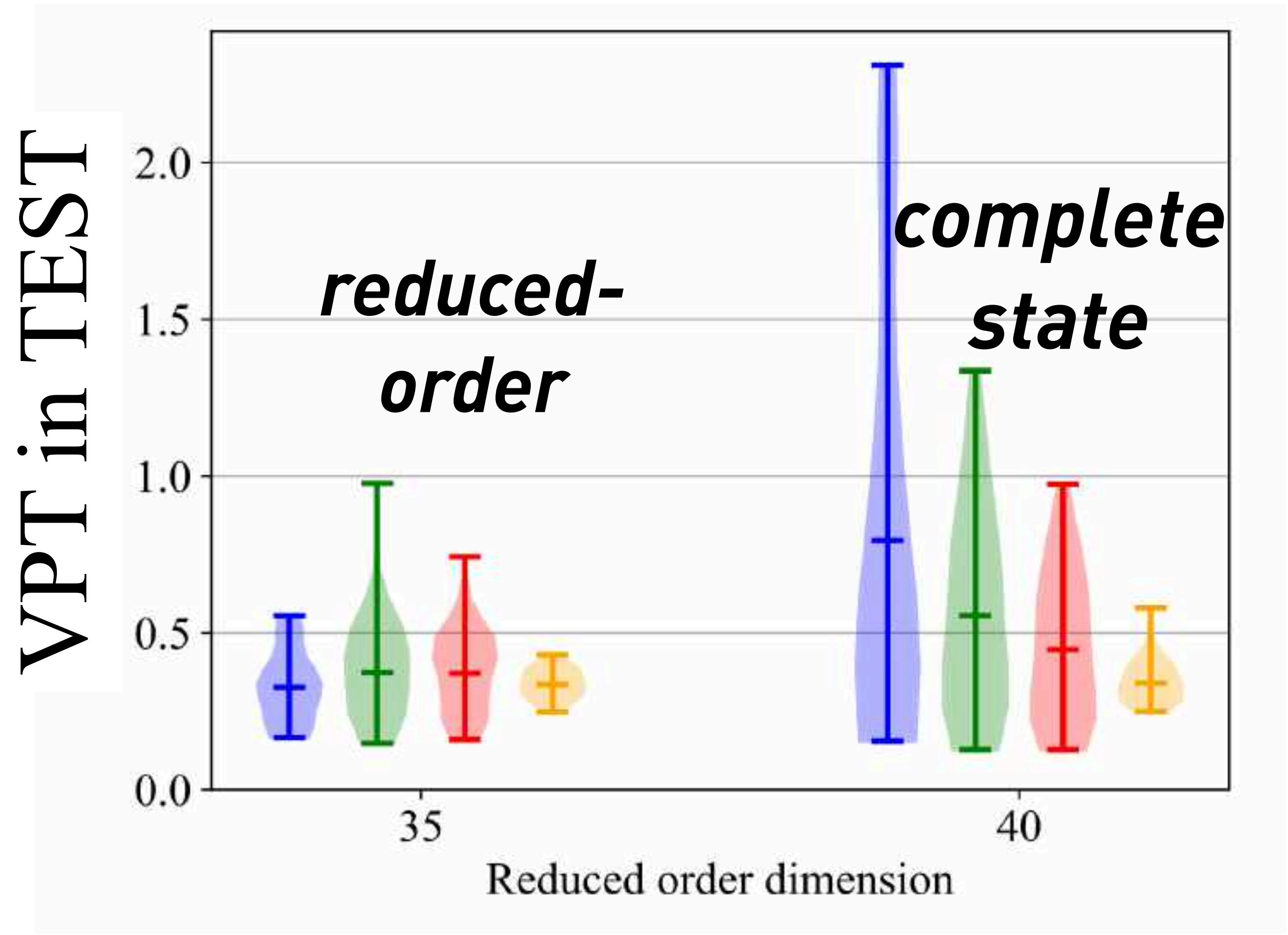
RC (or □) ; GRU (or ✕) ; LSTM (or ○) ; Unit (or ◇) ; Ideal ;

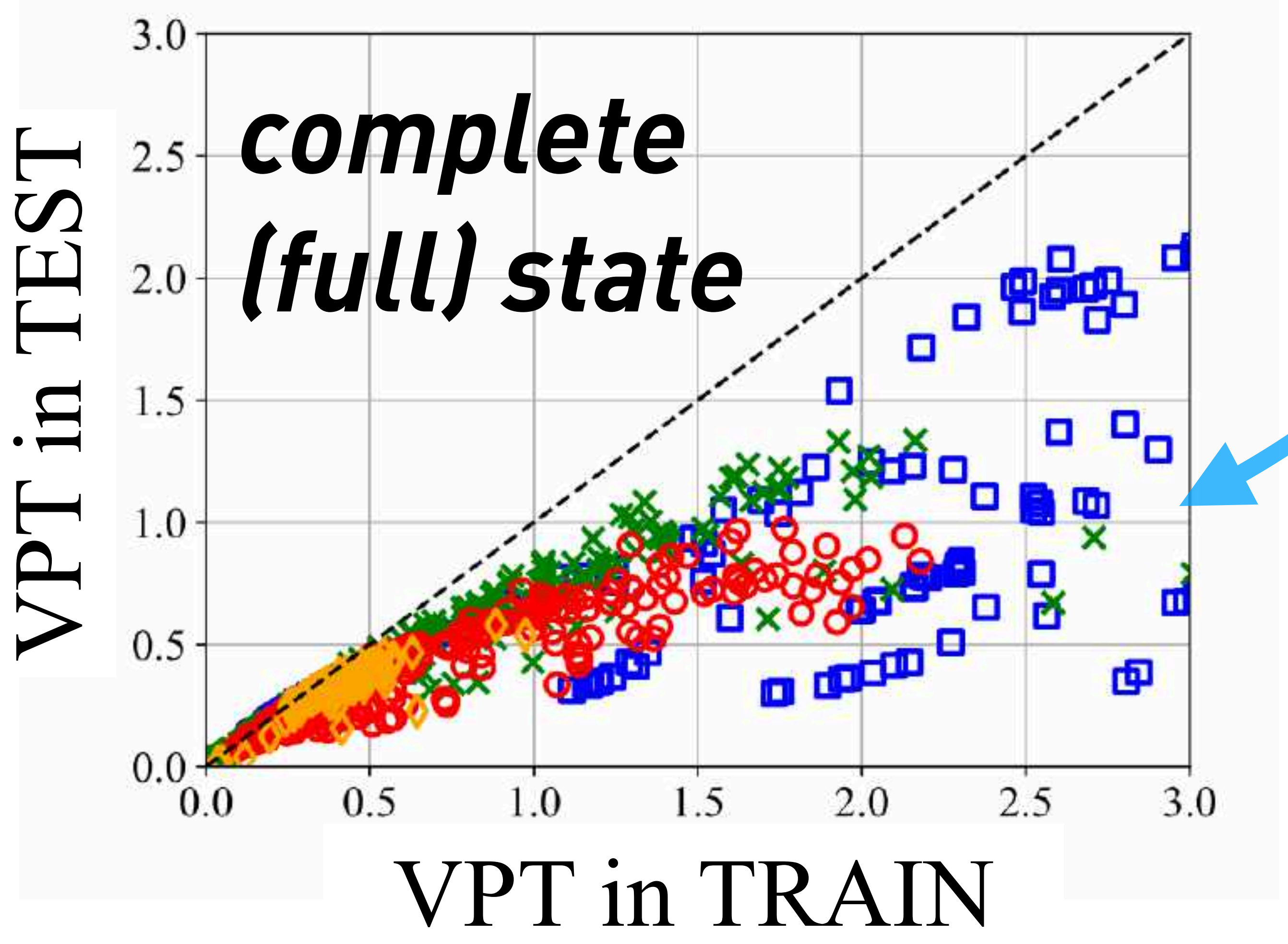
Lorenz 96 - 35 / 40 mode observable

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics*, JNN, 2020

- Valid prediction time $\mathrm{VPT} = \frac{1}{T_{\Lambda_1}} \operatorname*{argmax}_{t_f} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)

every marker is a trained model !





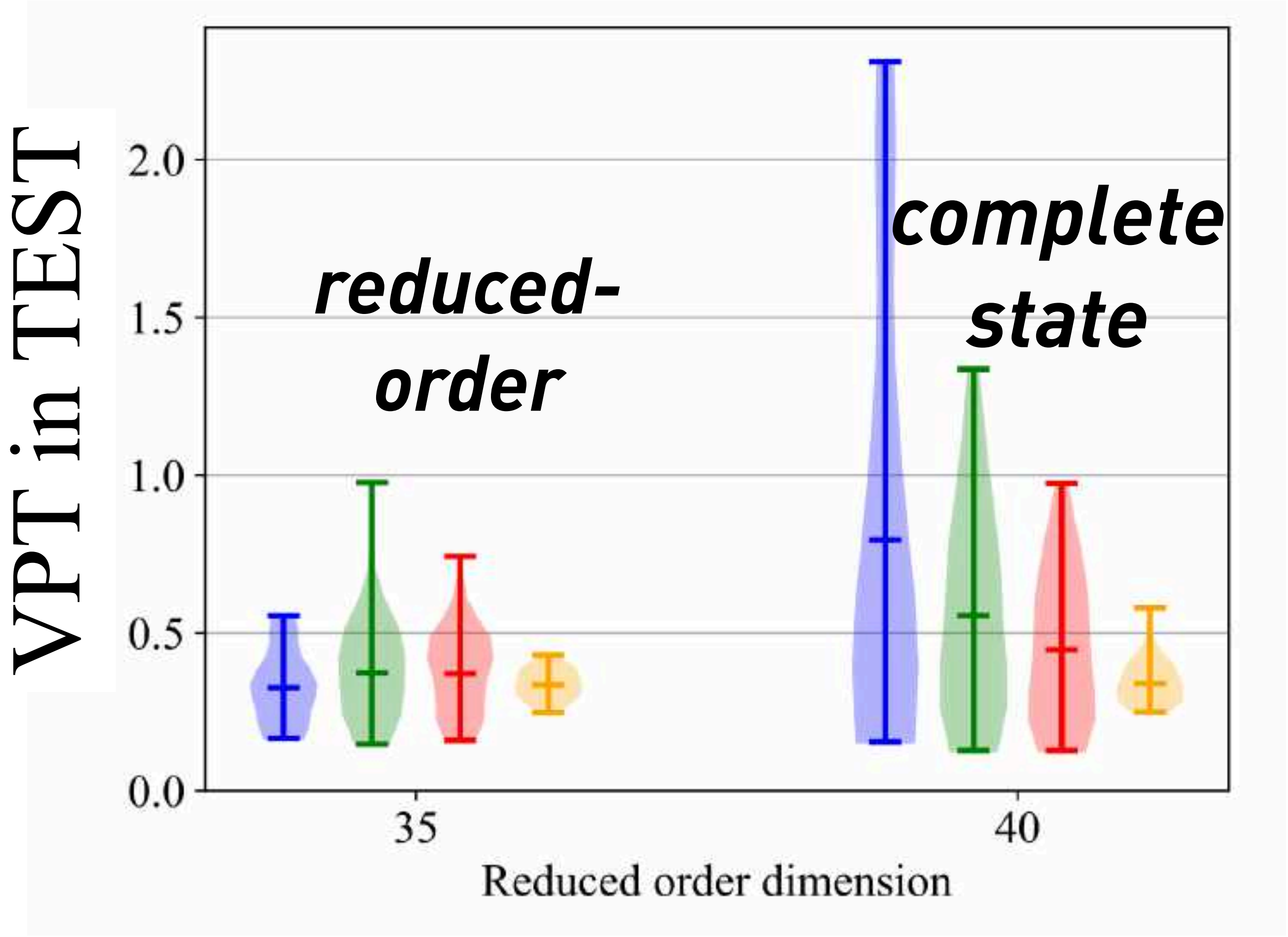
RC (or □) ; GRU (or ✕) ; LSTM (or ○) ; Unit (or ◇) ; Ideal;

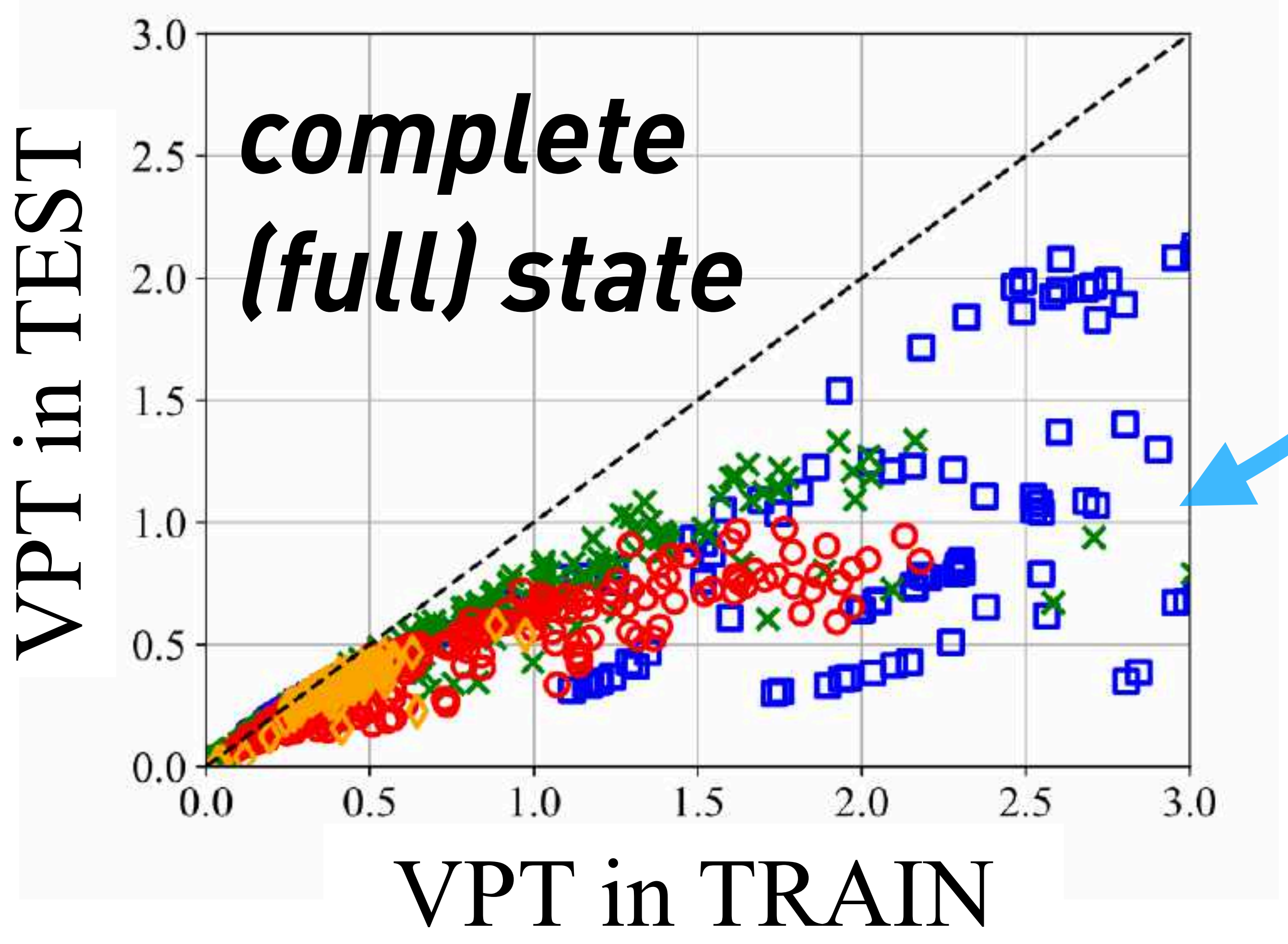
Lorenz 96 - 35 / 40 mode observable

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **JNN, 2020**

- Valid prediction time $\mathrm{VPT} = \dfrac{1}{T_{\Lambda_1}} \underset{t_f}{\mathrm{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)
- ***Gated architectures*** superior in case of ***reduced order state/observable*** (more expensive to train)

every marker is a trained model !





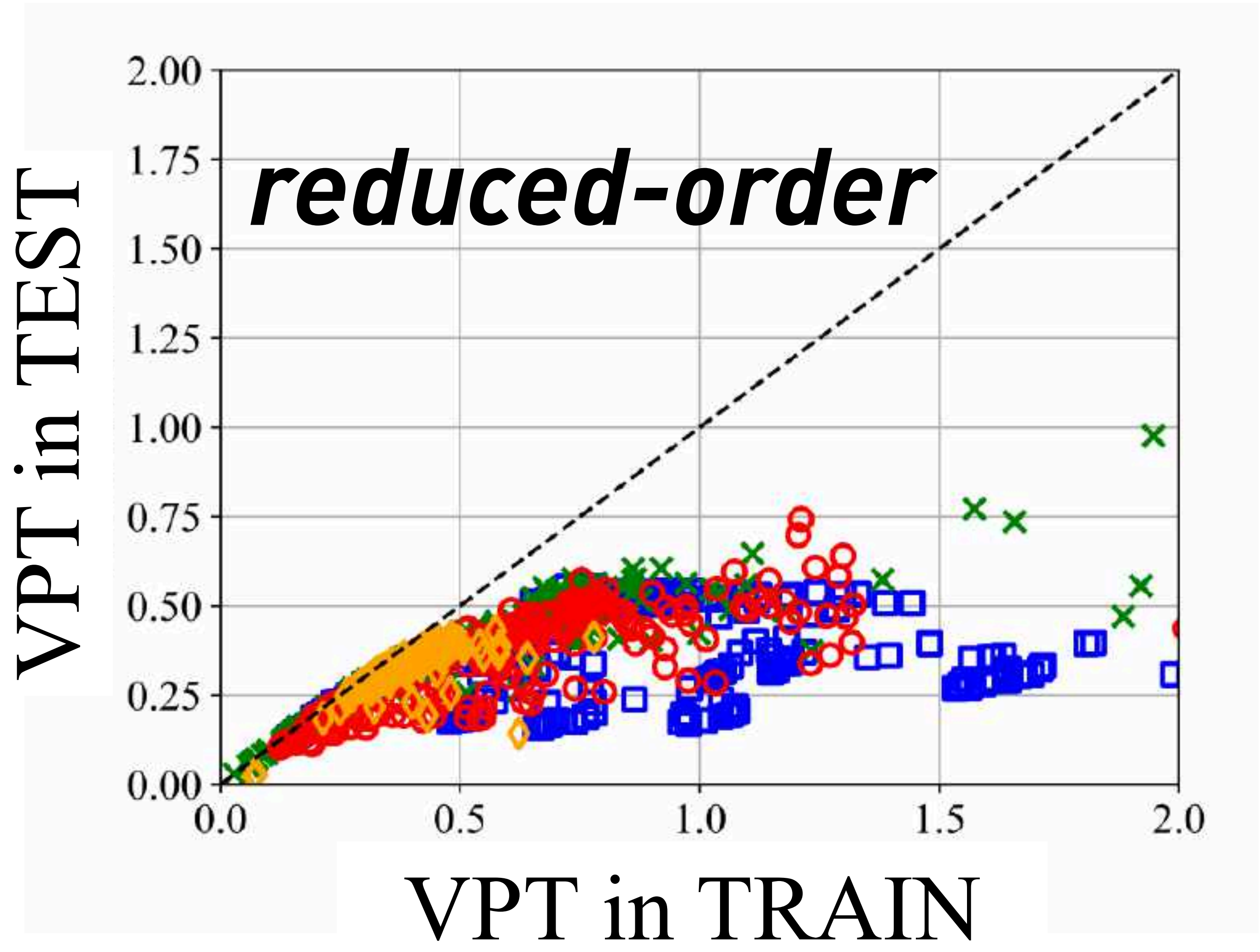
RC (or □) ; GRU (or ✕) ; LSTM (or ○) ; Unit (or ◇) ; Ideal ;

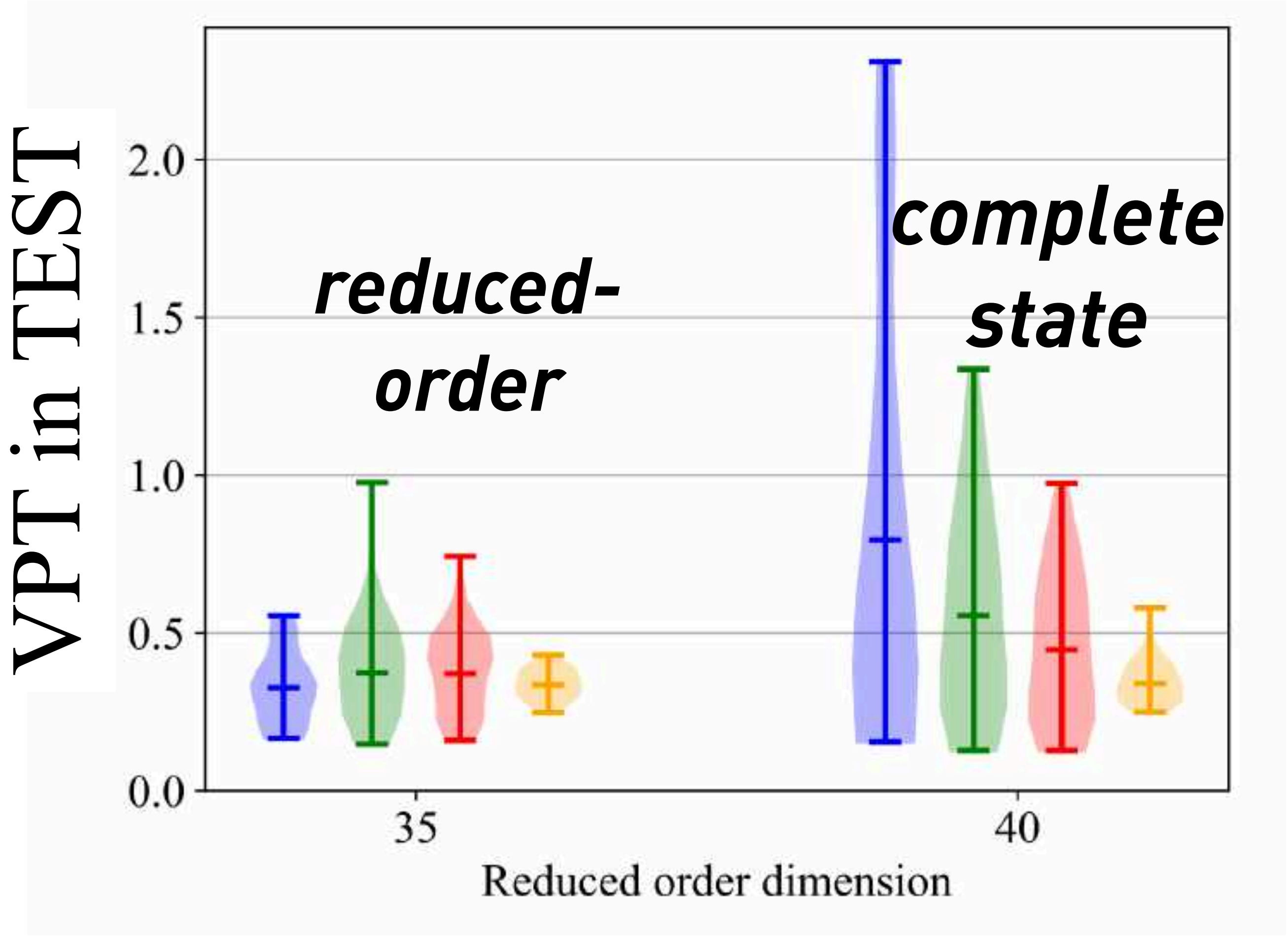
Lorenz 96 - 35 / 40 mode observable

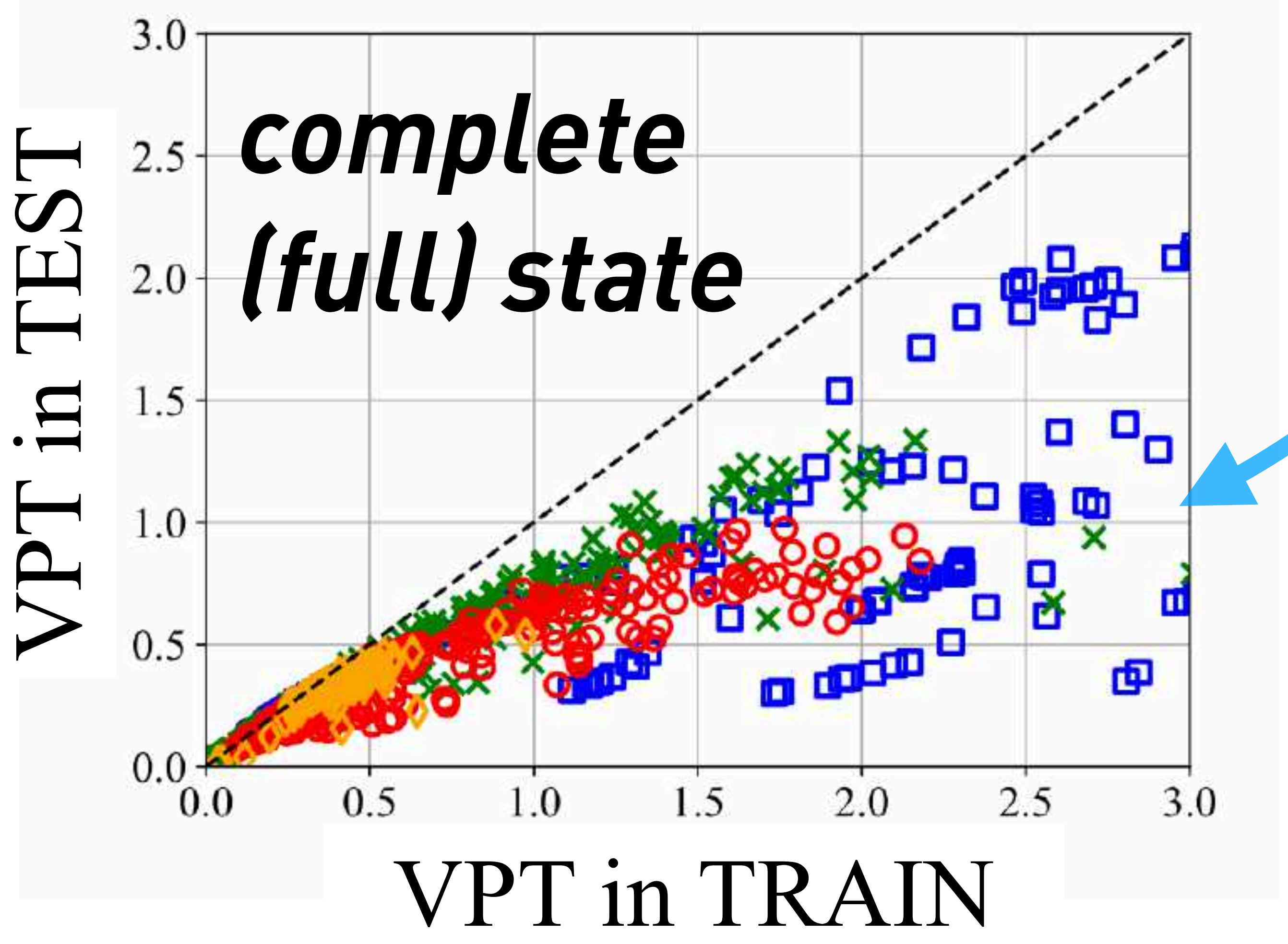
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics*, **JNN, 2020**

- Valid prediction time $\text{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\text{argmax}} \{ t_f \mid \text{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f \}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)
- ***Gated architectures*** superior in case of ***reduced order state/observable*** (more expensive to train)

every marker is a trained model !







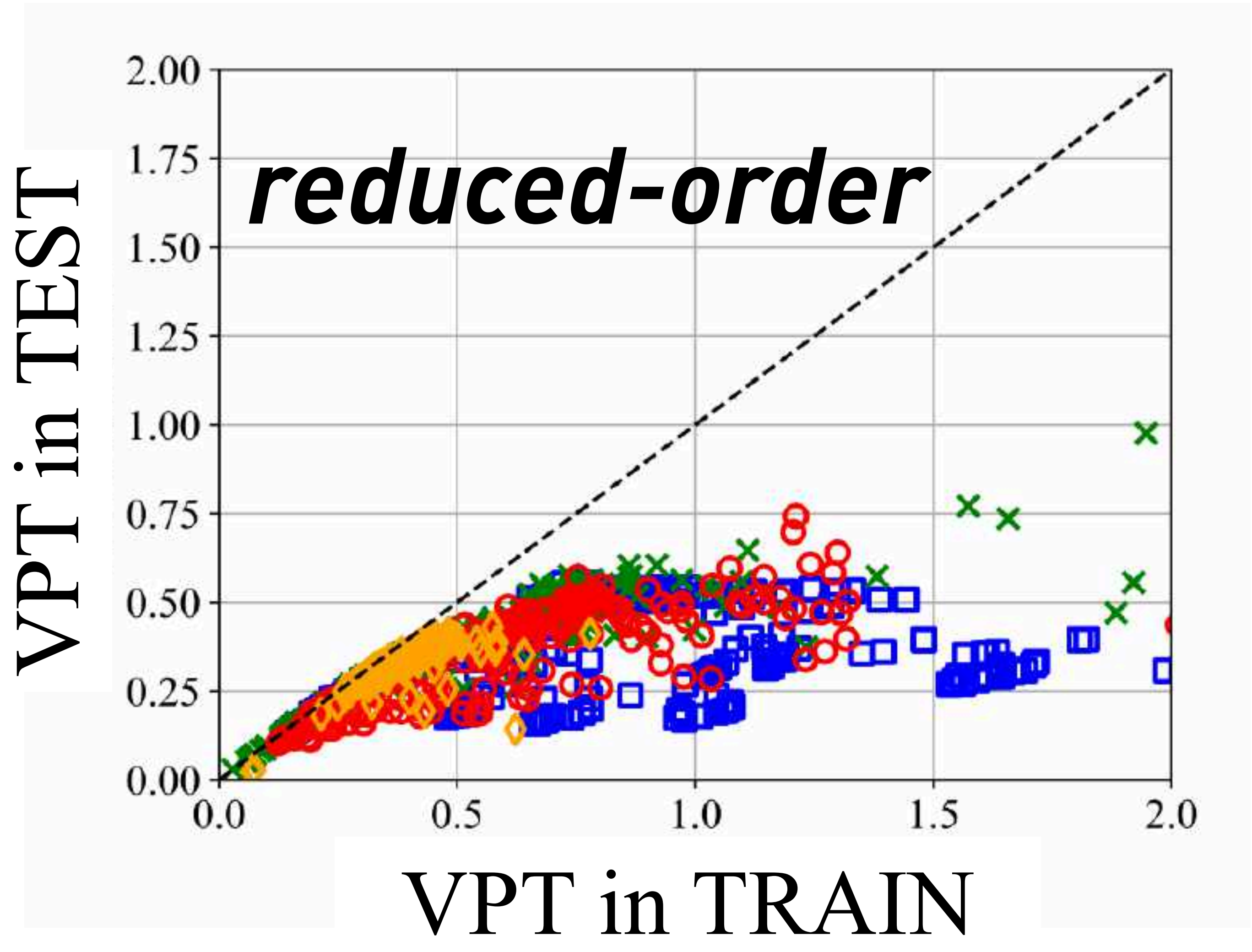
RC (or □) ; GRU (or ✕) ; LSTM (or ○) ; Unit (or ◇) ; Ideal ;

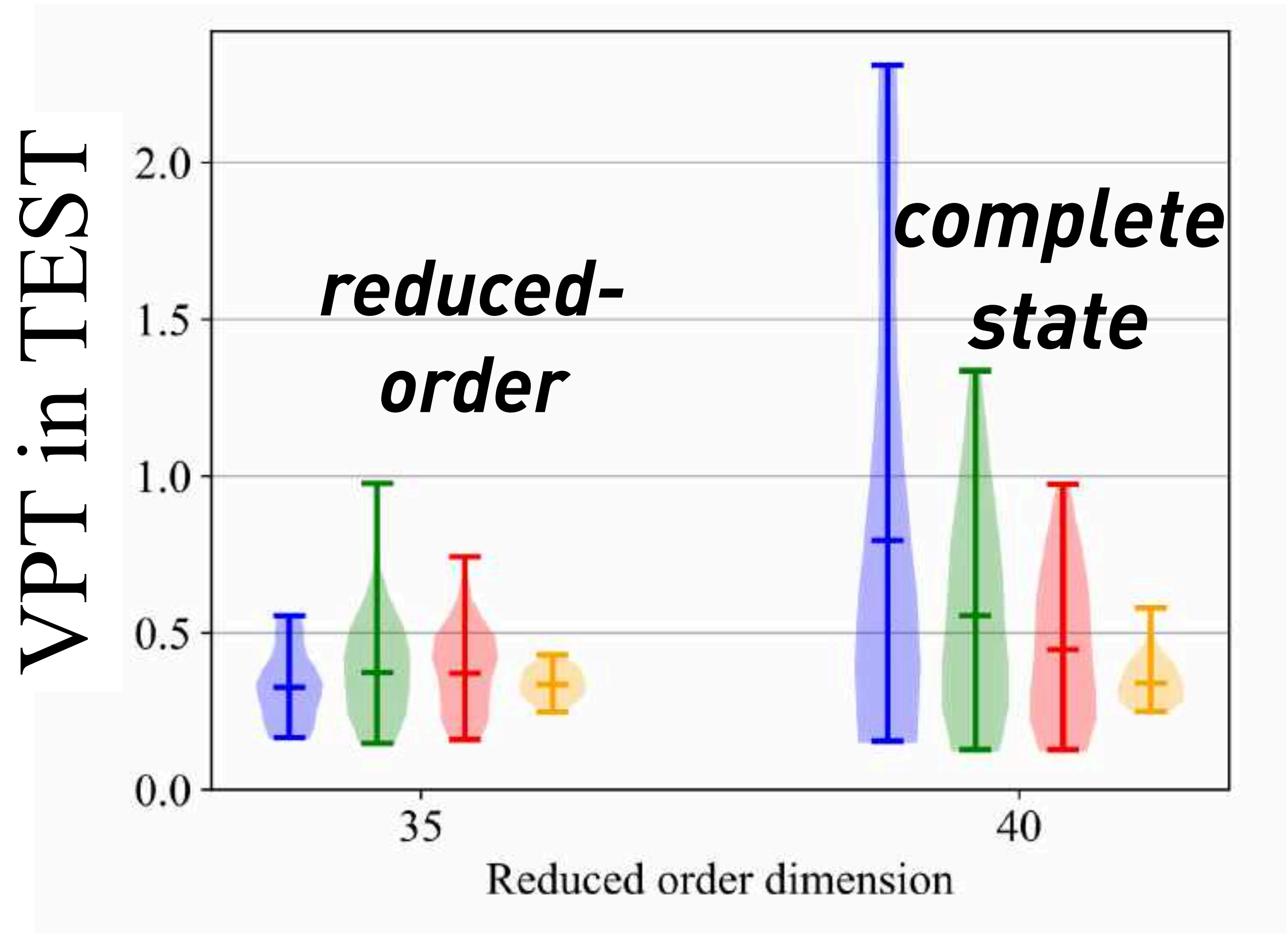
Lorenz 96 - 35 / 40 mode observable

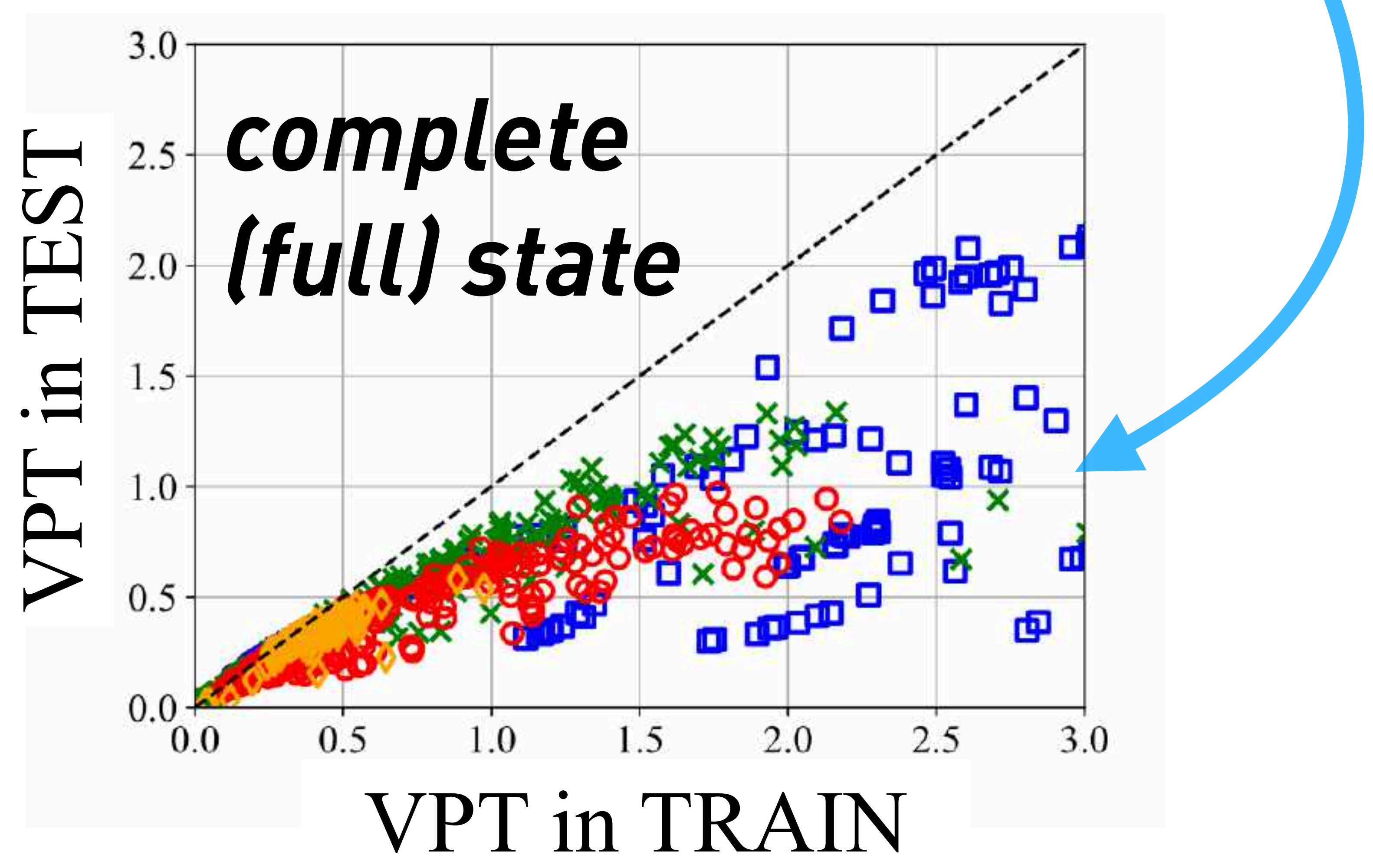
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics*, **JNN, 2020**

- Valid prediction time $\text{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\text{argmax}} \{t_f \mid \text{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)
- ***Gated architectures*** superior in case of ***reduced order state/observable*** (more expensive to train)
- ***RC*** has expressive power but lacks generalisation !

every marker is a trained model !





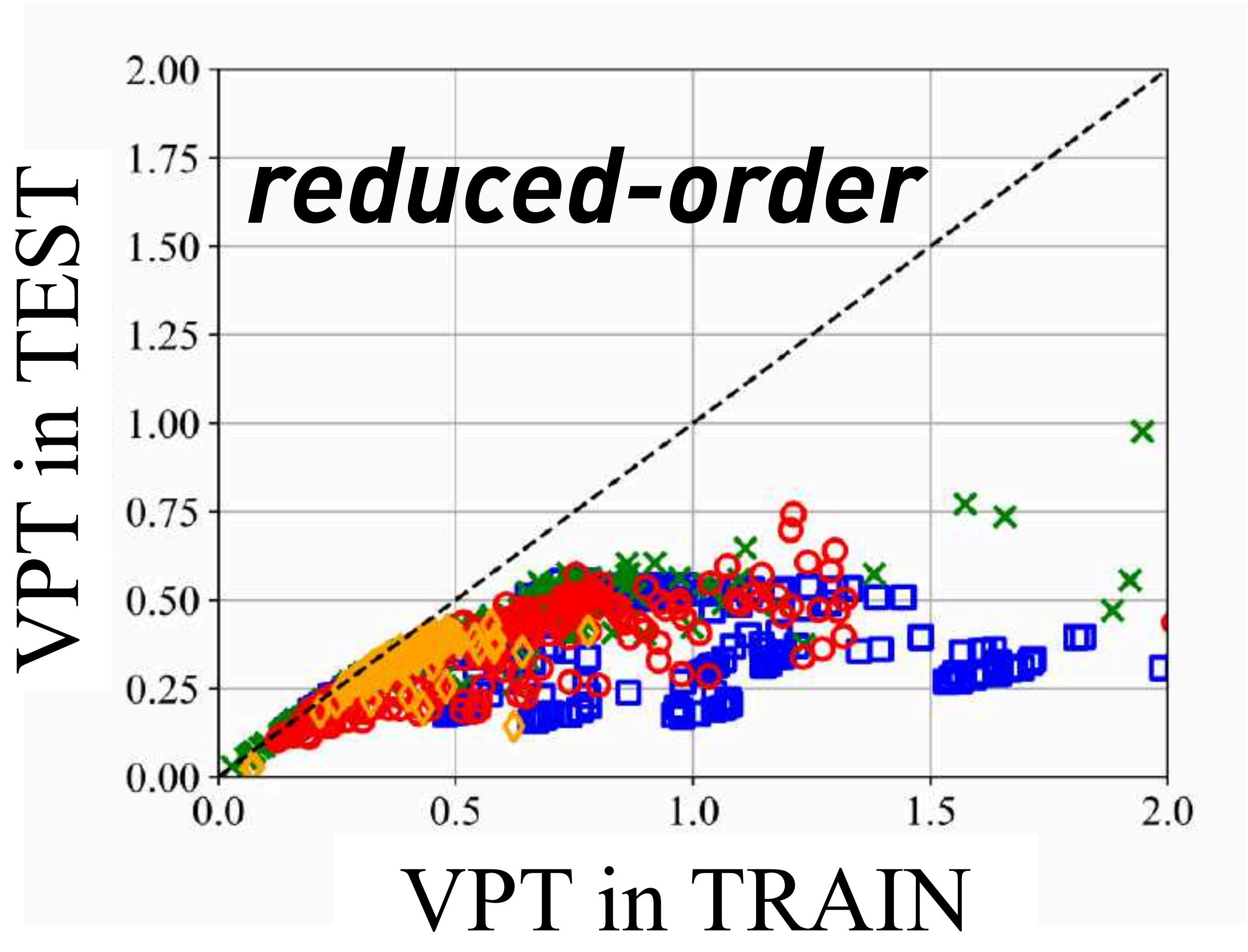


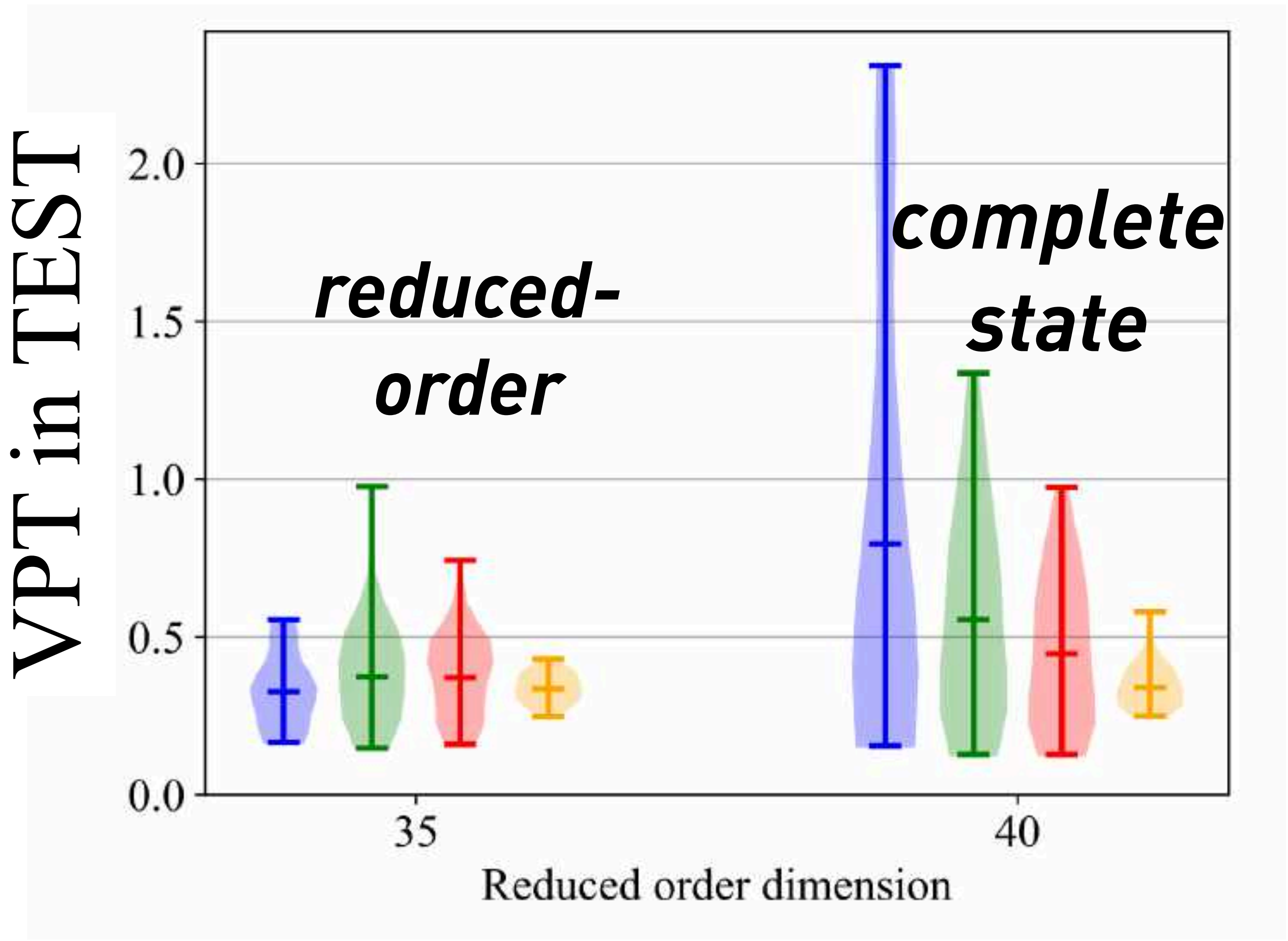
Lorenz 96 - 35 / 40 mode observable

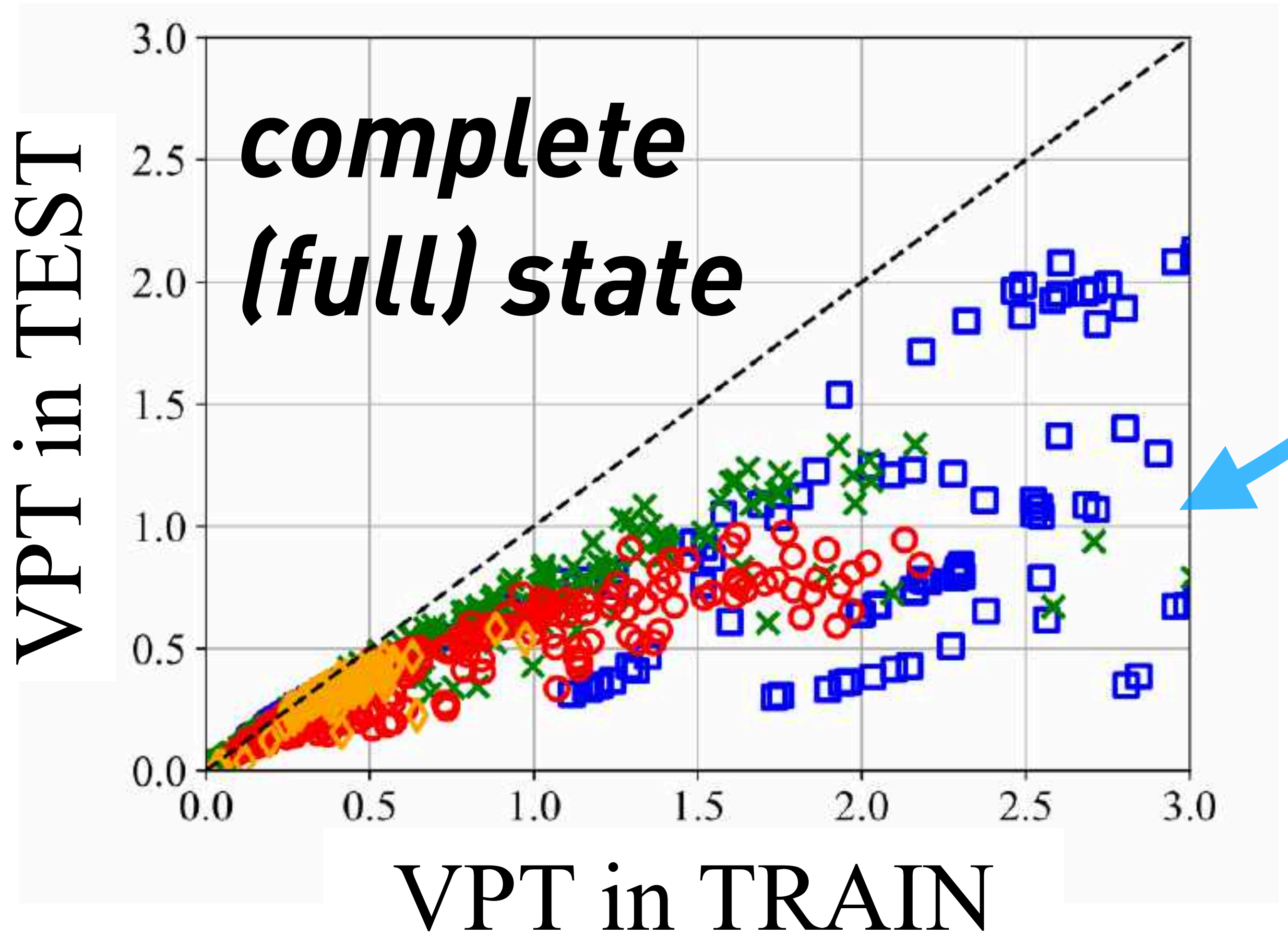
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics*, **JNN, 2020**

- Valid prediction time $\mathrm{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\mathrm{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)
- ***Gated architectures*** superior in case of ***reduced order state/observable*** (more expensive to train)
- ***RC*** has expressive power but lacks generalisation !
- ***Gated architectures*** more robust against overfitting

every marker is a trained model !







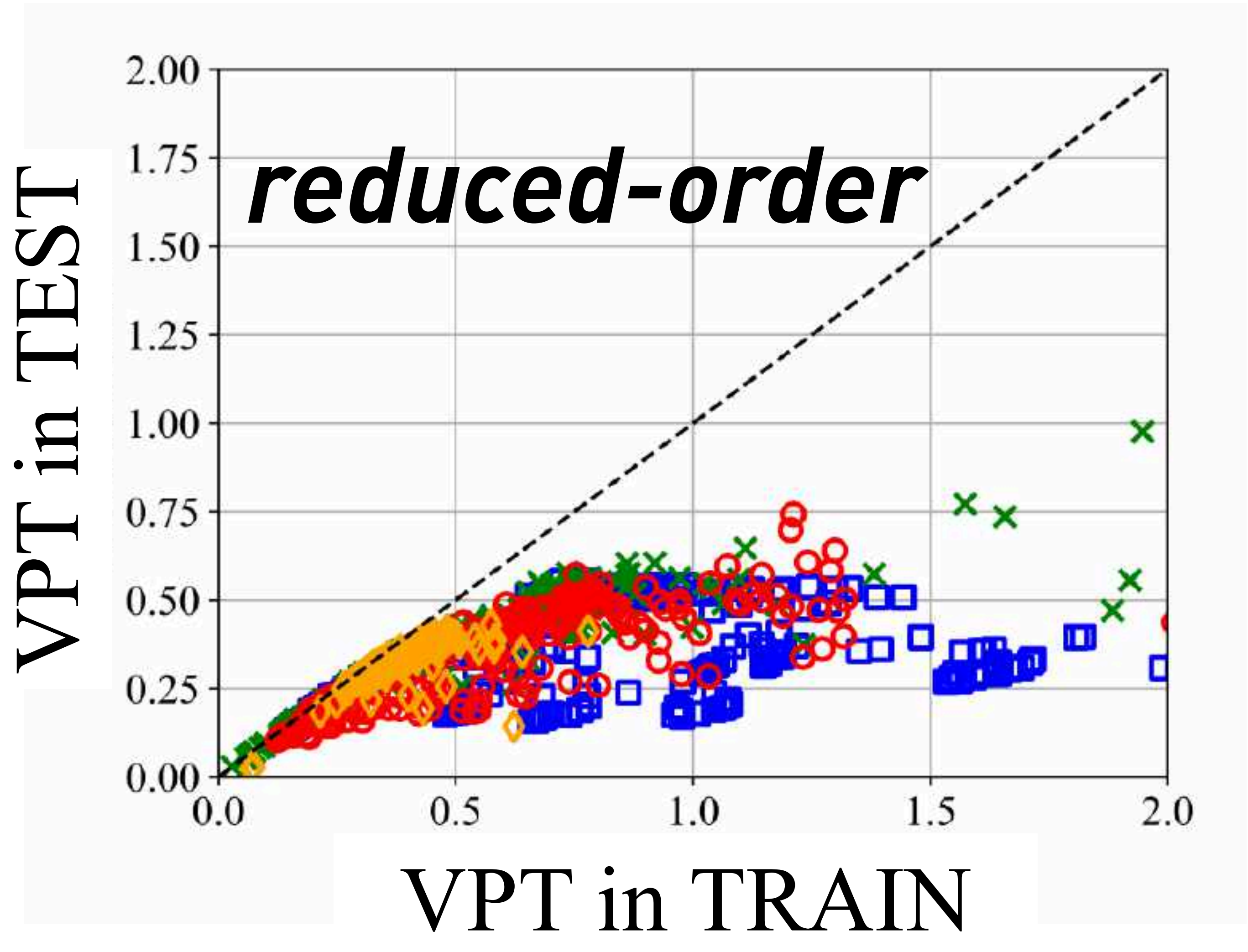
RC (or □) ; GRU (or ✕) ; LSTM (or ○) ; Unit (or ◇) ; Ideal ;

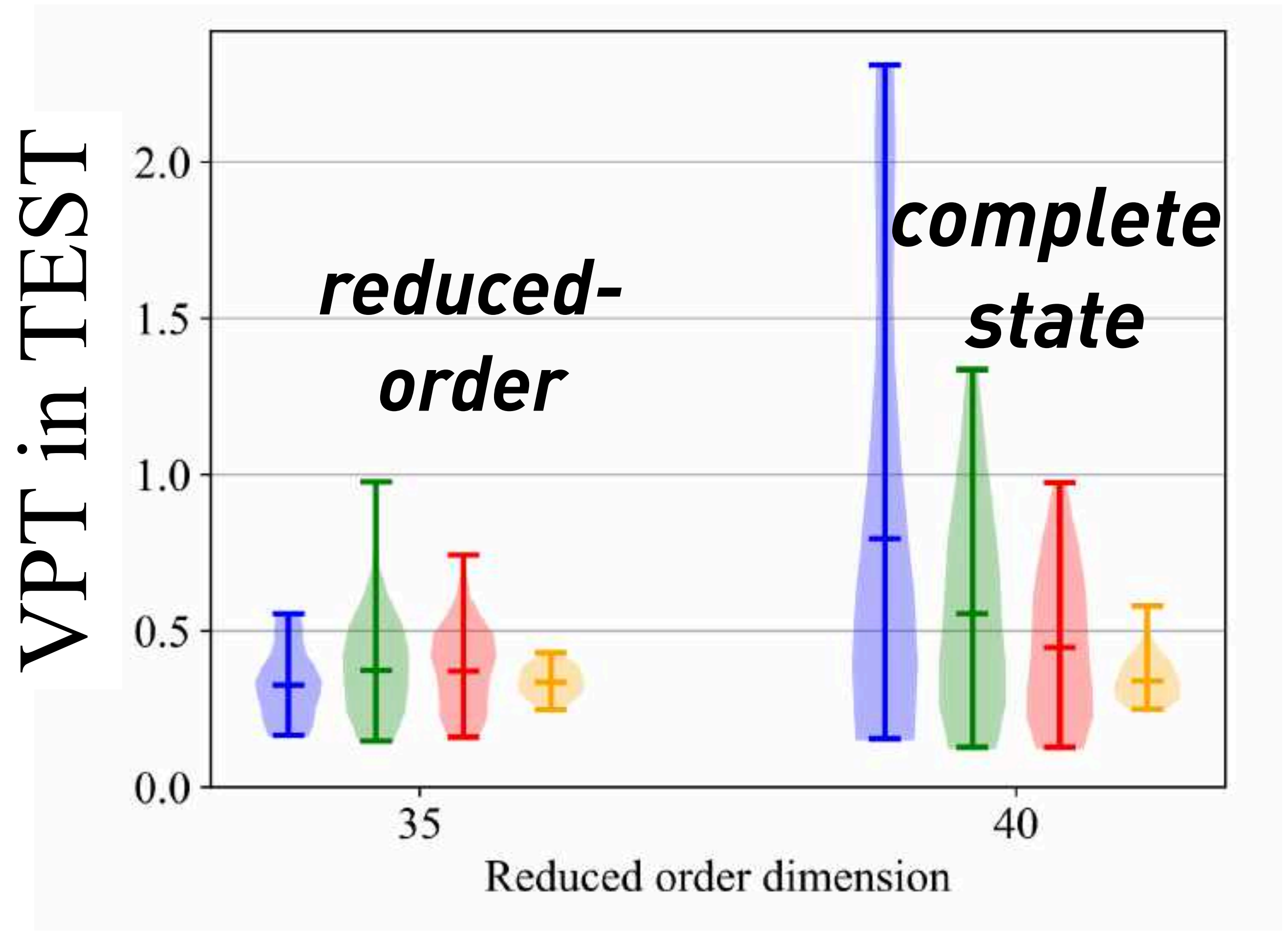
Lorenz 96 - 35 / 40 mode observable

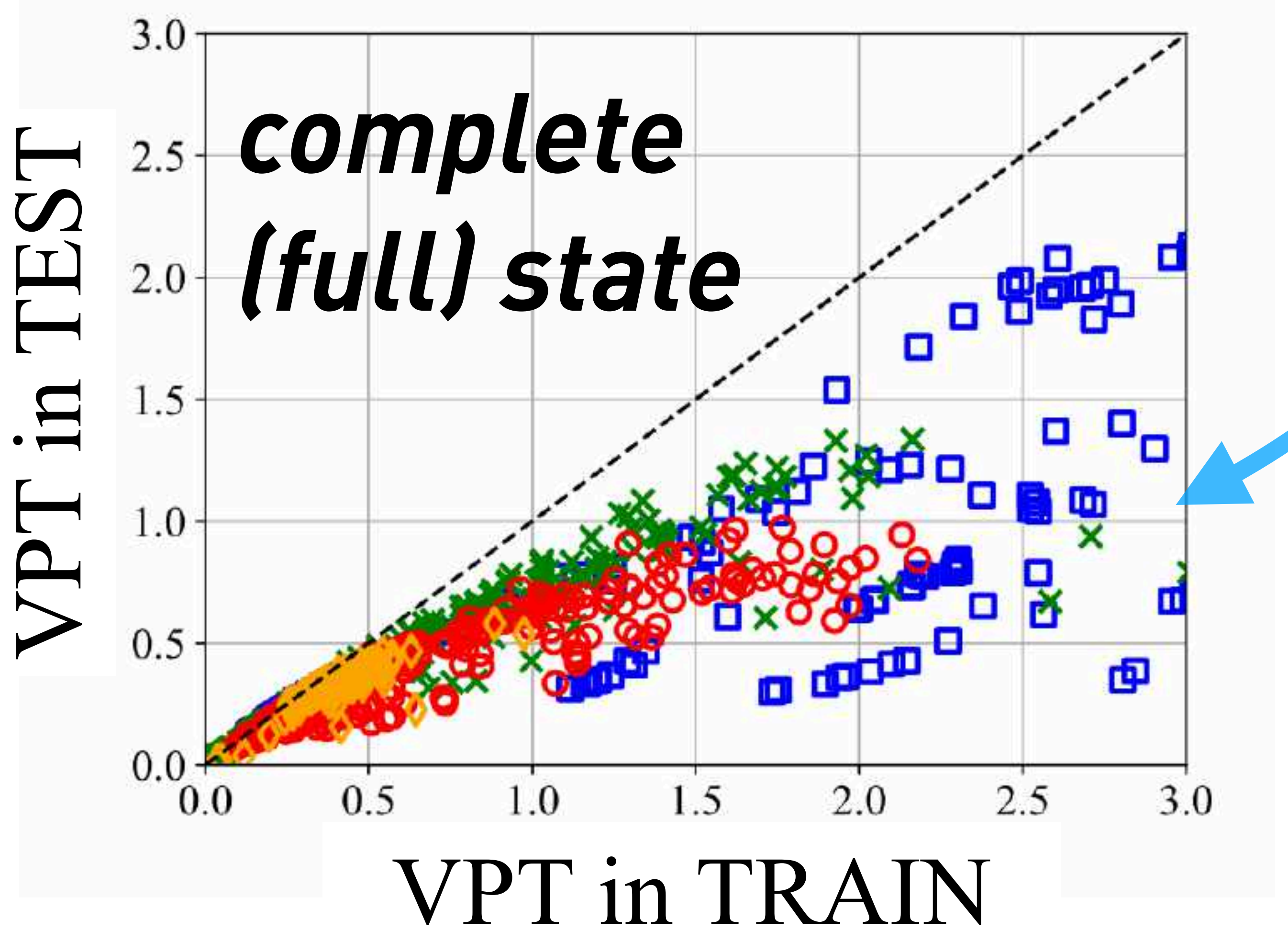
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics*, JNN, 2020

- Valid prediction time $\mathrm{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\mathrm{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)
- ***Gated architectures*** superior in case of ***reduced order state/observable*** (more expensive to train)
- ***RC*** has expressive power but lacks generalisation !
- ***Gated architectures*** more robust against overfitting
- Regularisation procedures utilized in BPTT (zoneout, dropout) are effective

every marker is a trained model !





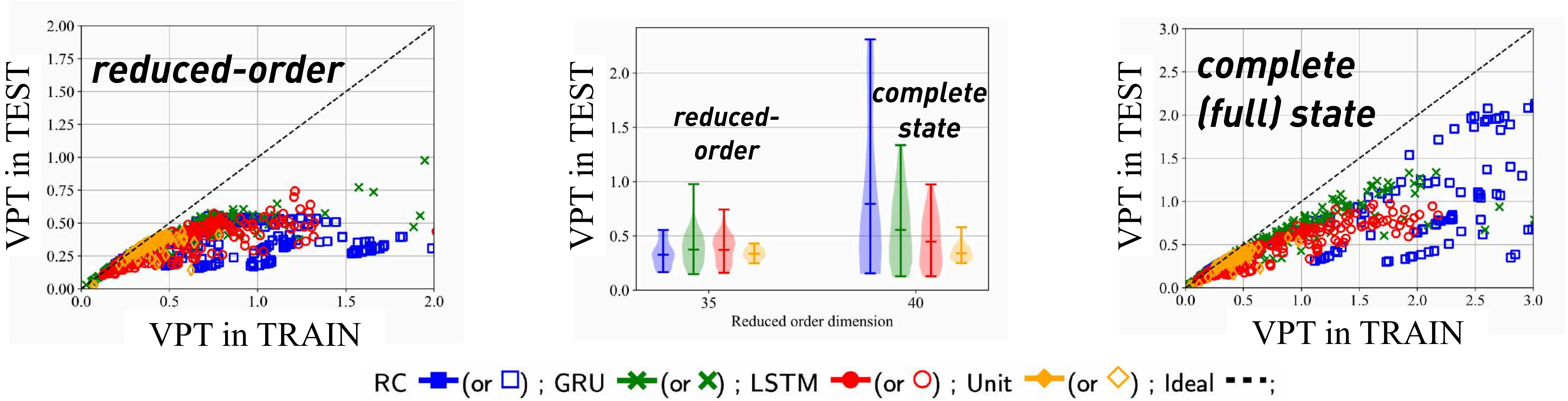


RC (or □) ; GRU (or ×) ; LSTM (or ○) ; Unit (or ◇) ; Ideal ;

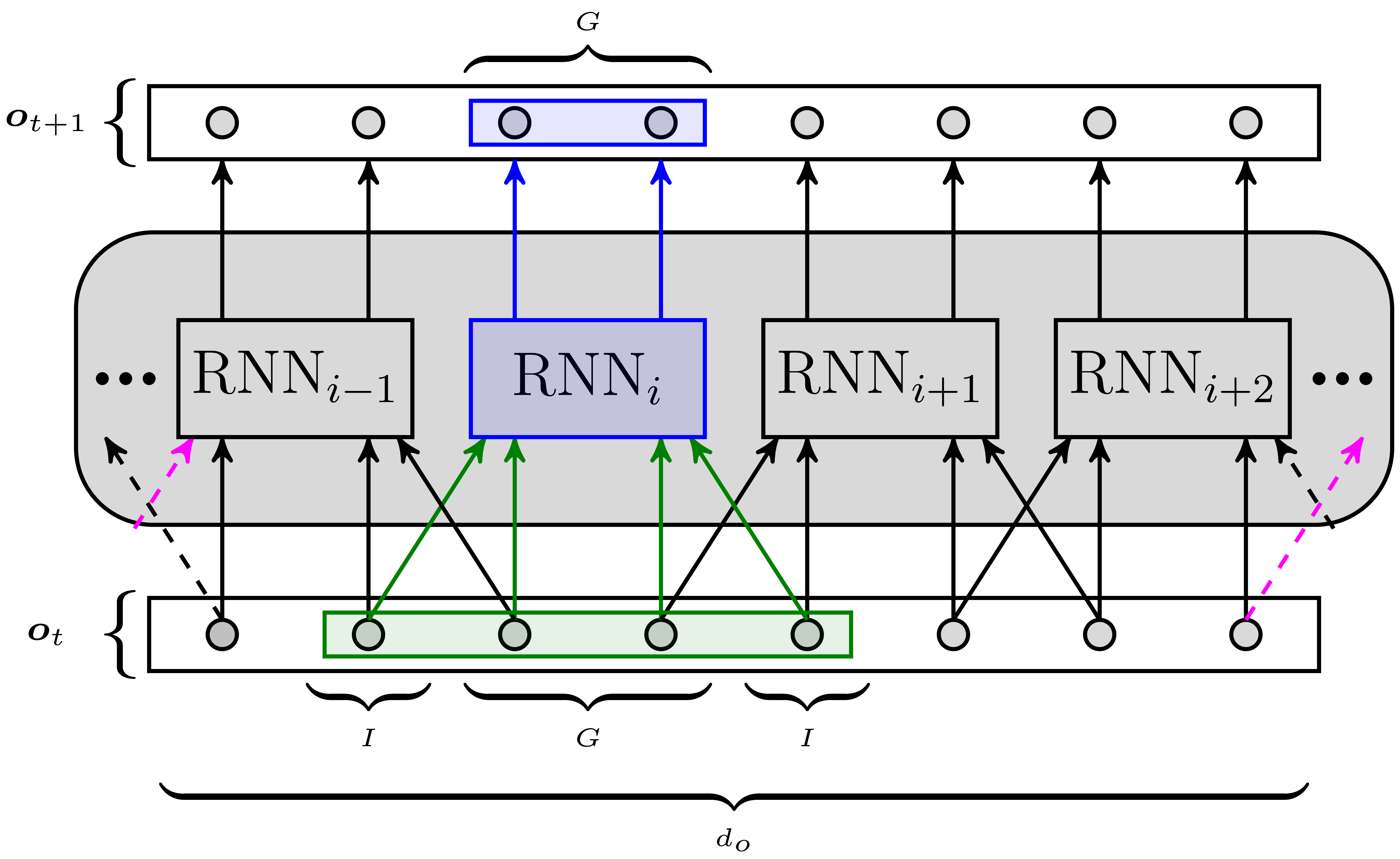
Lorenz 96 - 35 / 40 mode observable

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **JNN, 2020**

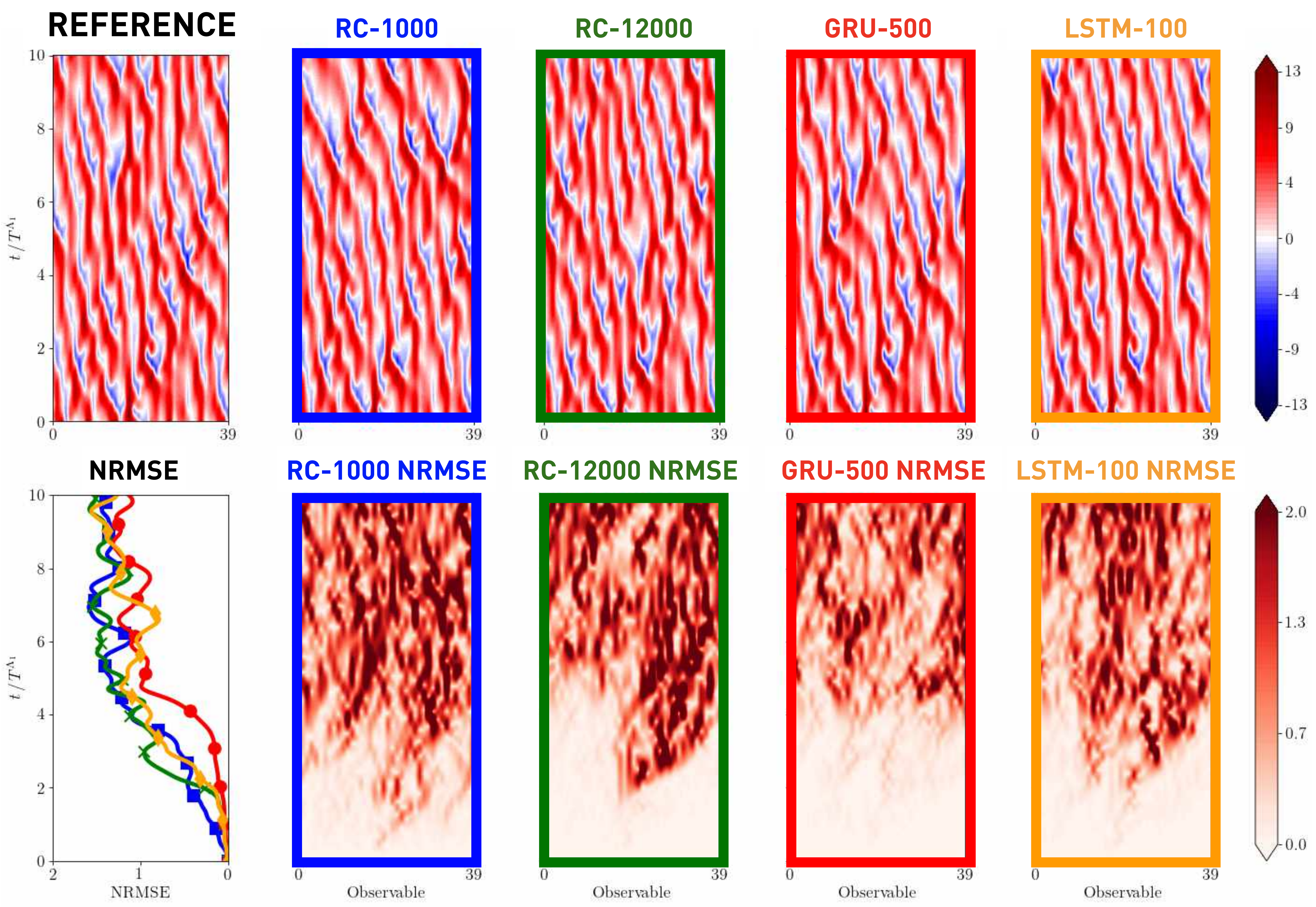
- Valid prediction time $\mathrm{VPT} = \frac{1}{T_{\Lambda_1}} \underset{t_f}{\mathrm{argmax}} \{t_f \mid \mathrm{NRMSE}(\mathbf{o}_t) < \epsilon, \forall t \leq t_f\}$, $\epsilon = 0.5$ here (the higher, the better)
- ***RC*** superior in case of ***complete (full) state*** information (good generalization, cheaper to train)
- ***Gated architectures*** superior in case of ***reduced order state/observable*** (more expensive to train)
- ***RC*** has expressive power but lacks generalisation !
- ***Gated architectures*** more robust against overfitting
- Regularisation procedures utilized in BPTT (zoneout, dropout) are effective



Lorenz 96, $F = 8$, *full state information & parallelism*



Lorenz 96, $F = 8$, *full state information & parallelism*



RC-1000; RC-12000; GRU-500; LSTM-100;

II

Learning Effective Dynamics

Equation-Free Framework (EFF) - Kevrekidis et. al.

Equation-Free Framework (EFF) - Kevrekidis et. al.

- Complex Multiscale systems: ***Micro*** scale ("particles") and ***Macro*** scale ("continuum") dynamics

Equation-Free Framework (EFF) - Kevrekidis et. al.

- Complex Multiscale systems: ***Micro*** scale ("particles") and ***Macro*** scale ("continuum") dynamics
- ***Microscale simulations:*** accurate **but** expensive to evaluate/not available

Equation-Free Framework (EFF) - Kevrekidis et. al.

- Complex Multiscale systems: ***Micro*** scale ("particles") and ***Macro*** scale ("continuum") dynamics
- ***Microscale simulations:*** accurate **but** expensive to evaluate/not available
- ***Macroscale simulations:*** approximate **but** inexpensive to evaluate

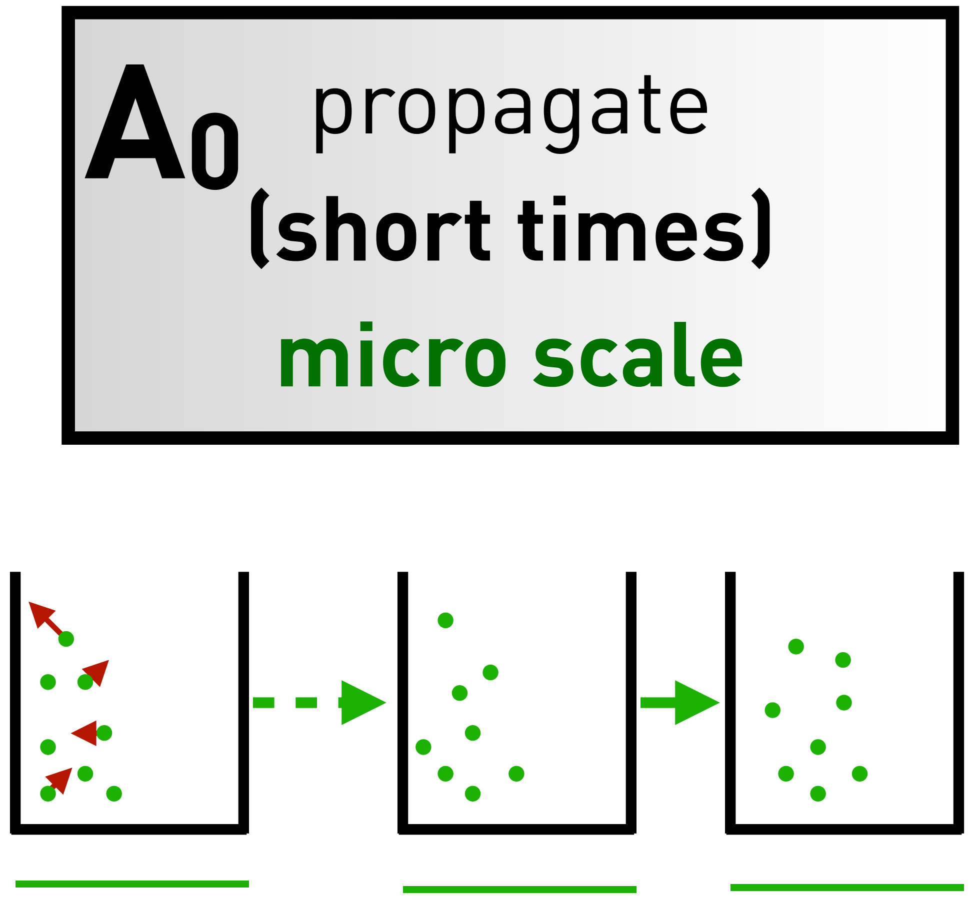
Equation-Free Framework (EFF) - Kevrekidis et. al.

- Complex Multiscale systems: ***Micro*** scale ("particles") and ***Macro*** scale ("continuum") dynamics
- ***Microscale simulations:*** accurate **but** expensive to evaluate/not available
- ***Macroscale simulations:*** approximate **but** inexpensive to evaluate

- **C Theodoropoulos, YH Qian, IG Kevrekidis,** *Coarse stability and bifurcation analysis using time-steppers: a reaction-diffusion example,* **Proc. Natl. Acad. Sci., 2000**
- **CW Gear, IG Kevrekidis, C Theodoropoulos,** *Coarse integration/bifurcation analysis via microscopic simulators: micro-Galerkin methods,* **Computers and Chemical Engineering, 2002**

AND MANY MANY MORE …

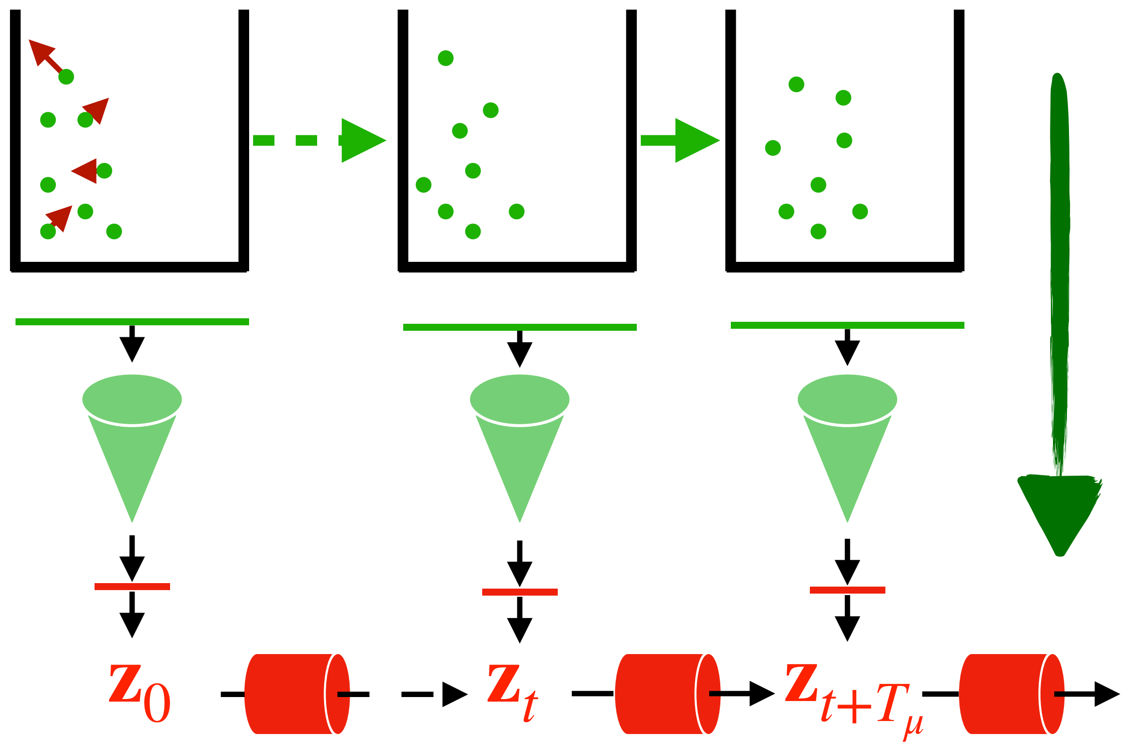
Equation Free Framework



A

Equation Free Framework

A0 propagate
(short times)
micro scale



B0 initialise
macro scale

Equation Free Framework

RESTRICTING /AVERAGING
(micro → macro)
e.g. PCA / DiffMaps / analytic

A0 propagate
(short times)
micro scale

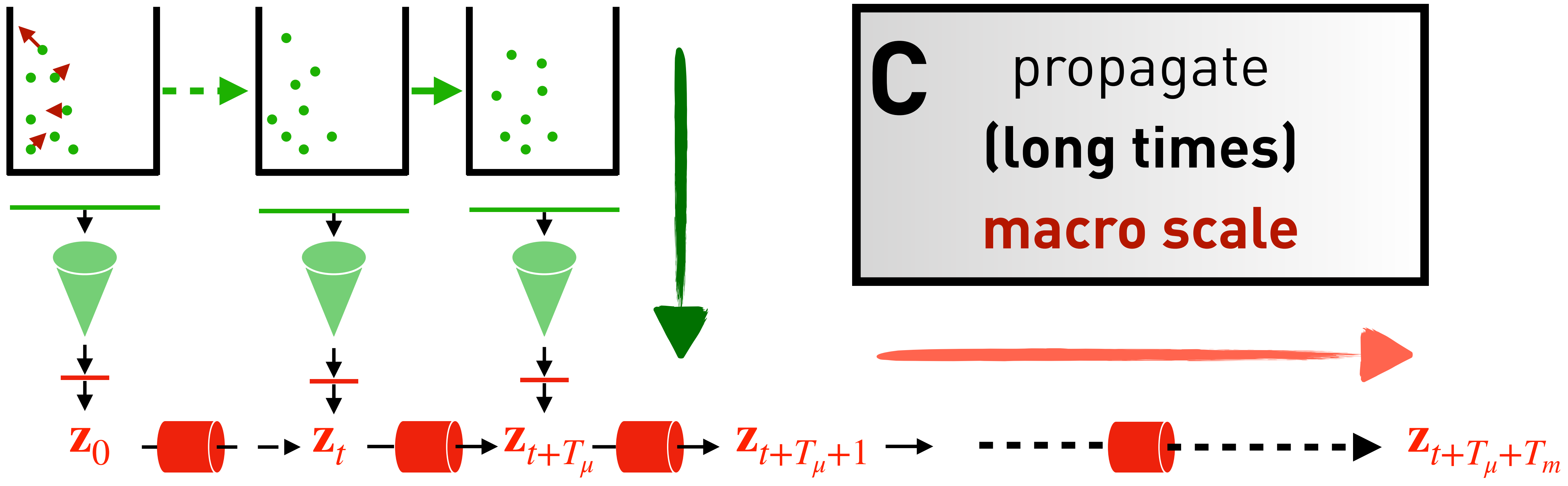
RESTRICTING /AVERAGING
(micro → macro)
e.g. PCA / DiffMaps / analytic

B

macro dynamics propagator
e.g. ODE / analytic

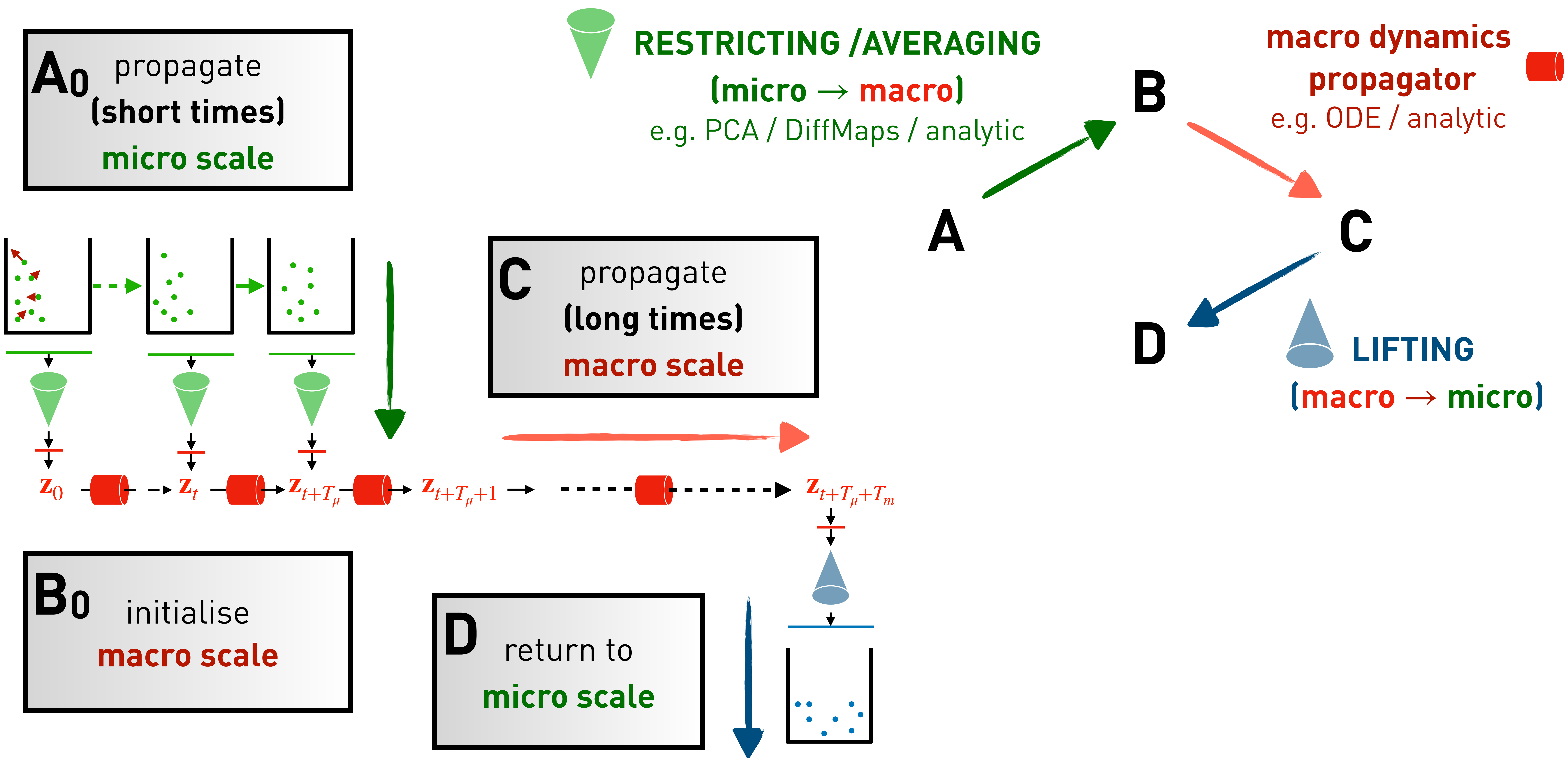
A → **B** → **C**

C propagate
(long times)
macro scale

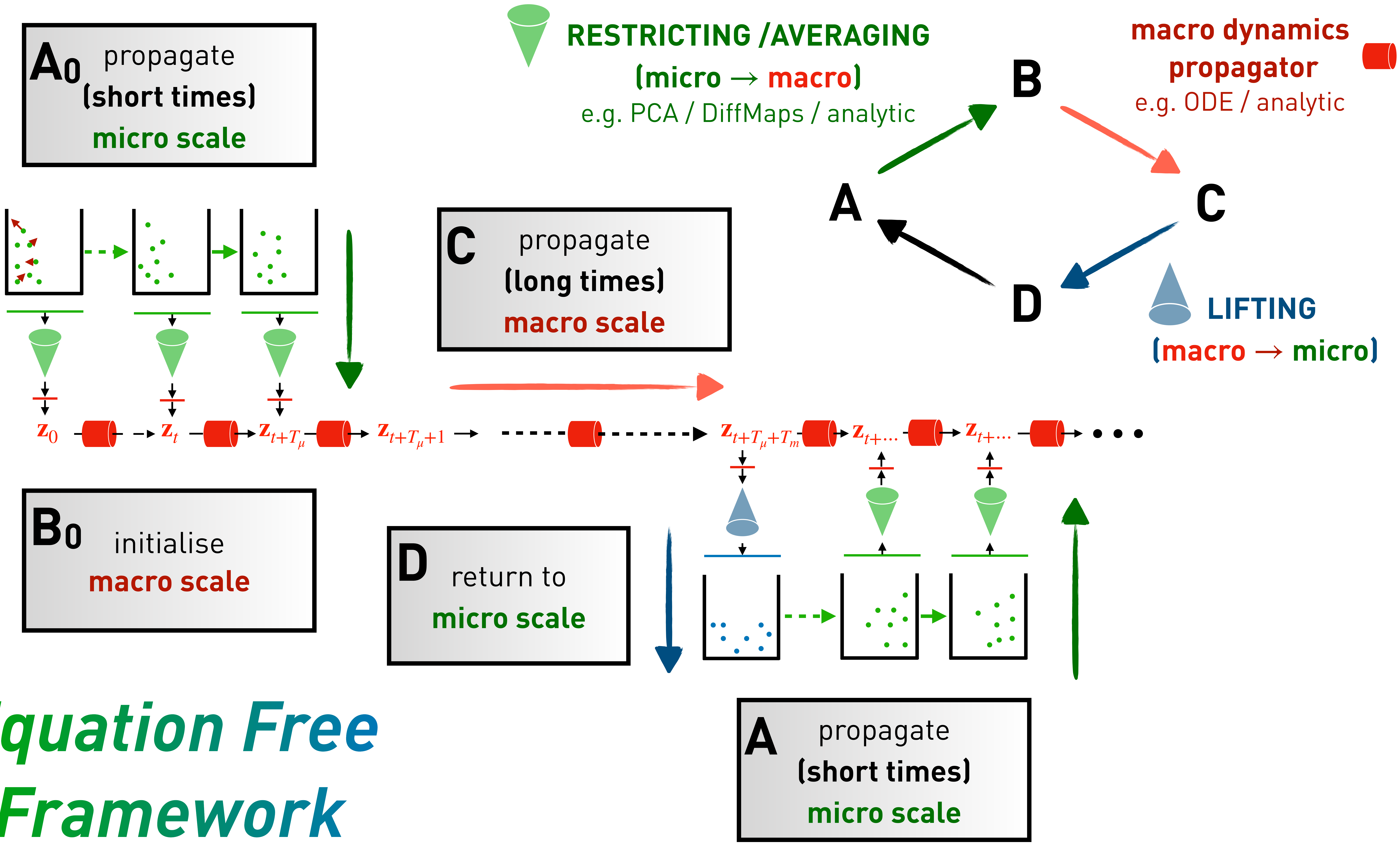


B0 initialise
macro scale

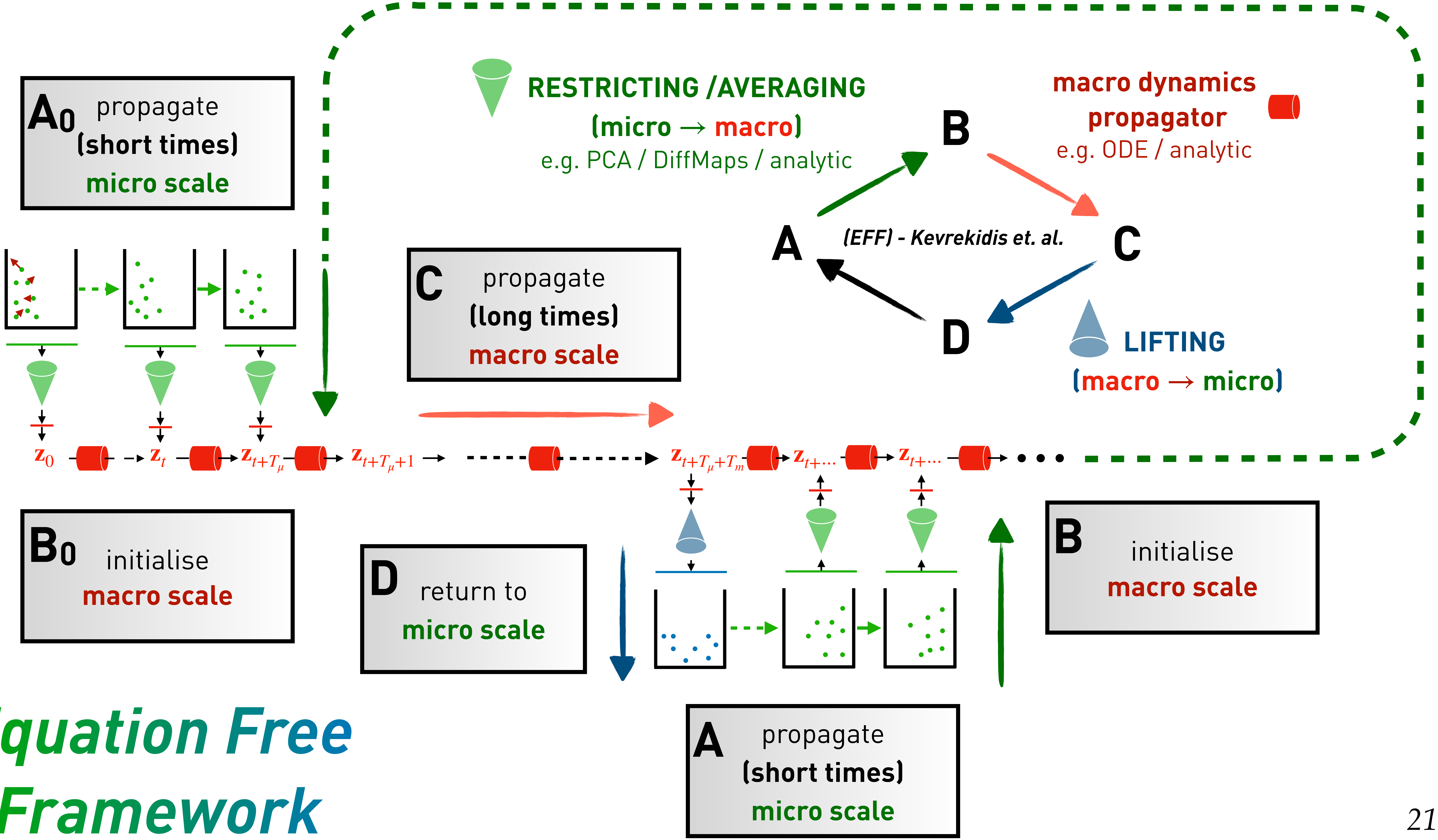
Equation Free Framework



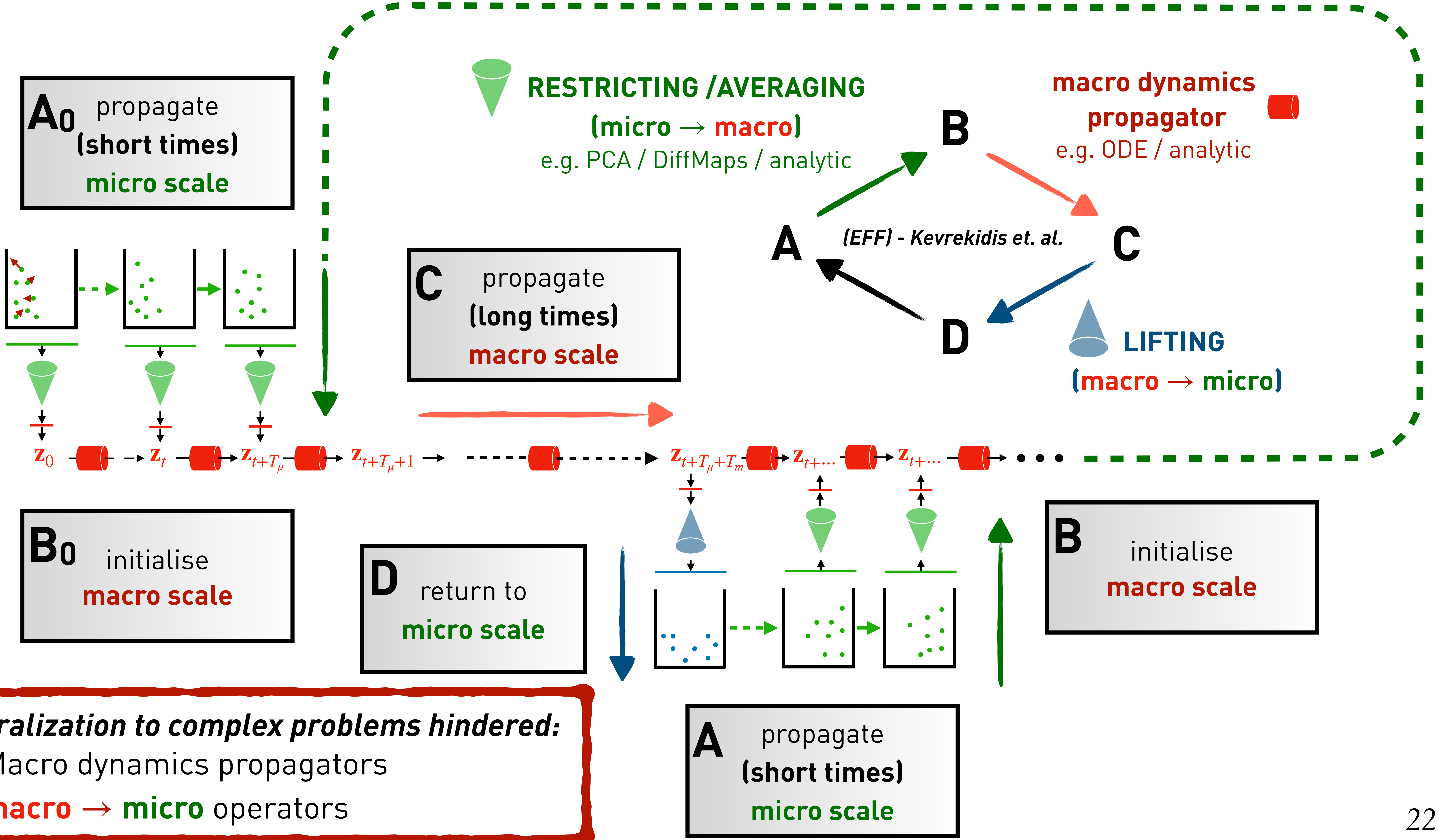
Equation Free Framework



Equation Free Framework



Equation Free Framework



Generalization to complex problems hindered:

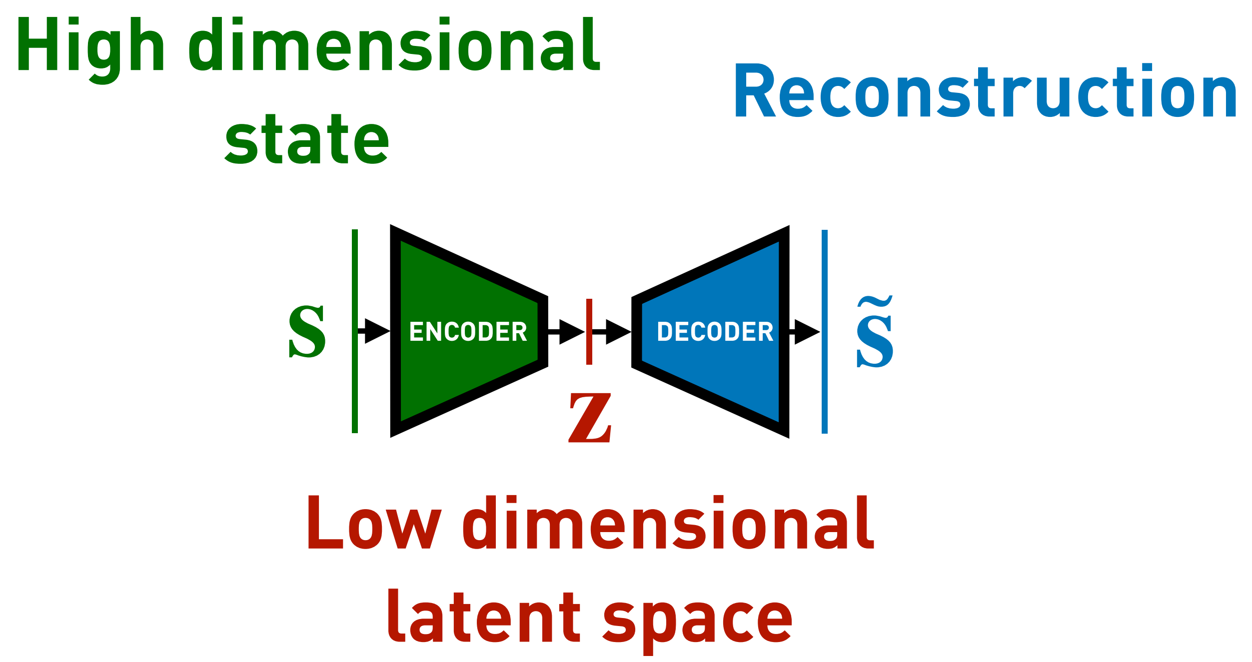
A. Macro dynamics propagators

B. **macro → micro** operators

Operators from Machine Learning

Operators from Machine Learning

(CONVOLUTIONAL) AUTOENCODERS

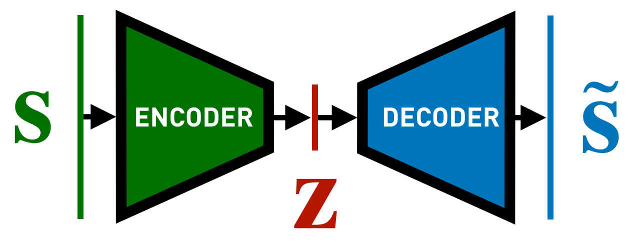


- Full high dimensional description of dynamical system system $\mathbf{s}$
- e.g. positions of atoms / micro scale / angles, bonds
- Loss Function $\mathscr{L} = |\mathbf{s} - \tilde{\mathbf{s}}|_2^2$
- Ideally after training $\mathbf{s} \approx \tilde{\mathbf{s}}$

Operators from Machine Learning

(CONVOLUTIONAL) AUTOENCODERS

High dimensional state **Reconstruction**

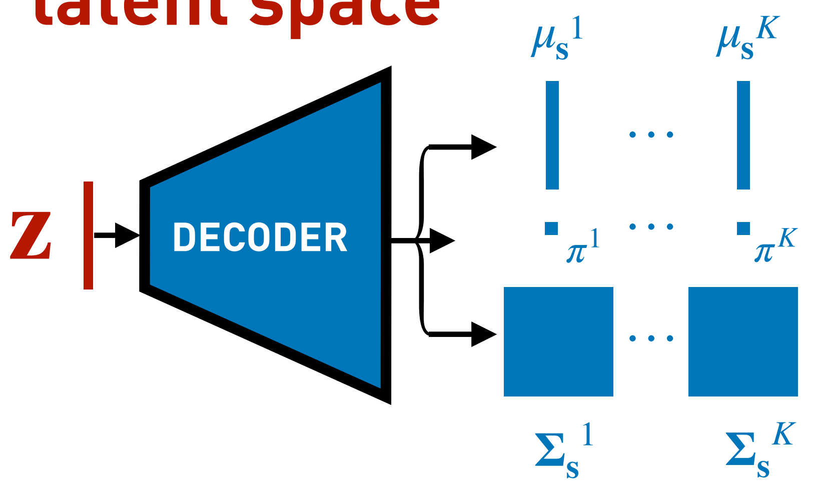


Low dimensional latent space

- Full high dimensional description of dynamical system system $\mathbf{s}$
- e.g. positions of atoms / micro scale / angles, bonds
- Loss Function $\mathscr{L} = |\mathbf{s} - \tilde{\mathbf{s}}|_2^2$
- Ideally after training $\mathbf{s} \approx \tilde{\mathbf{s}}$

MIXTURE DENSITY NETWORKS

Low dimensional latent space



Parametrisation of $p(\mathbf{s}\,|\,\mathbf{z})$

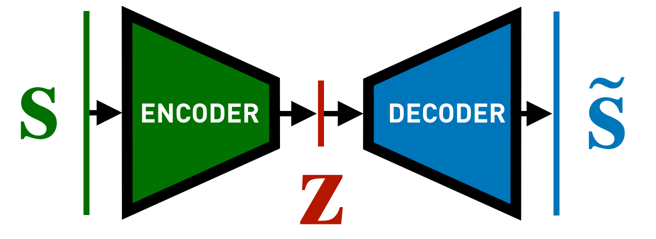
- Coarse (latent) representation has limited information
- Mapping $\mathbf{z} \to \mathbf{s}$ can be probabilistic !
- Generative network
- $p(\mathbf{s}\,|\,\mathbf{z})$ as **mixture model**
- $$p(\mathbf{s}\,|\,\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\mu_{\mathbf{s}}^k, \Sigma_{\mathbf{s}}^k\right)$$

Operators from Machine Learning

(CONVOLUTIONAL) AUTOENCODERS

High dimensional state

Reconstruction

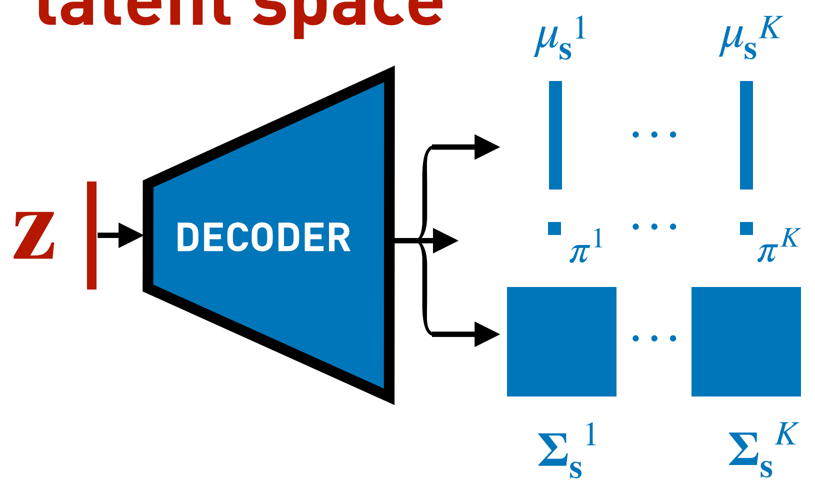


Low dimensional latent space

- Full high dimensional description of dynamical system system $\mathbf{s}$
- e.g. positions of atoms / micro scale / angles, bonds
- Loss Function $\mathscr{L} = |\mathbf{s} - \tilde{\mathbf{s}}|_2^2$
- Ideally after training $\mathbf{s} \approx \tilde{\mathbf{s}}$

MIXTURE DENSITY NETWORKS

Low dimensional latent space

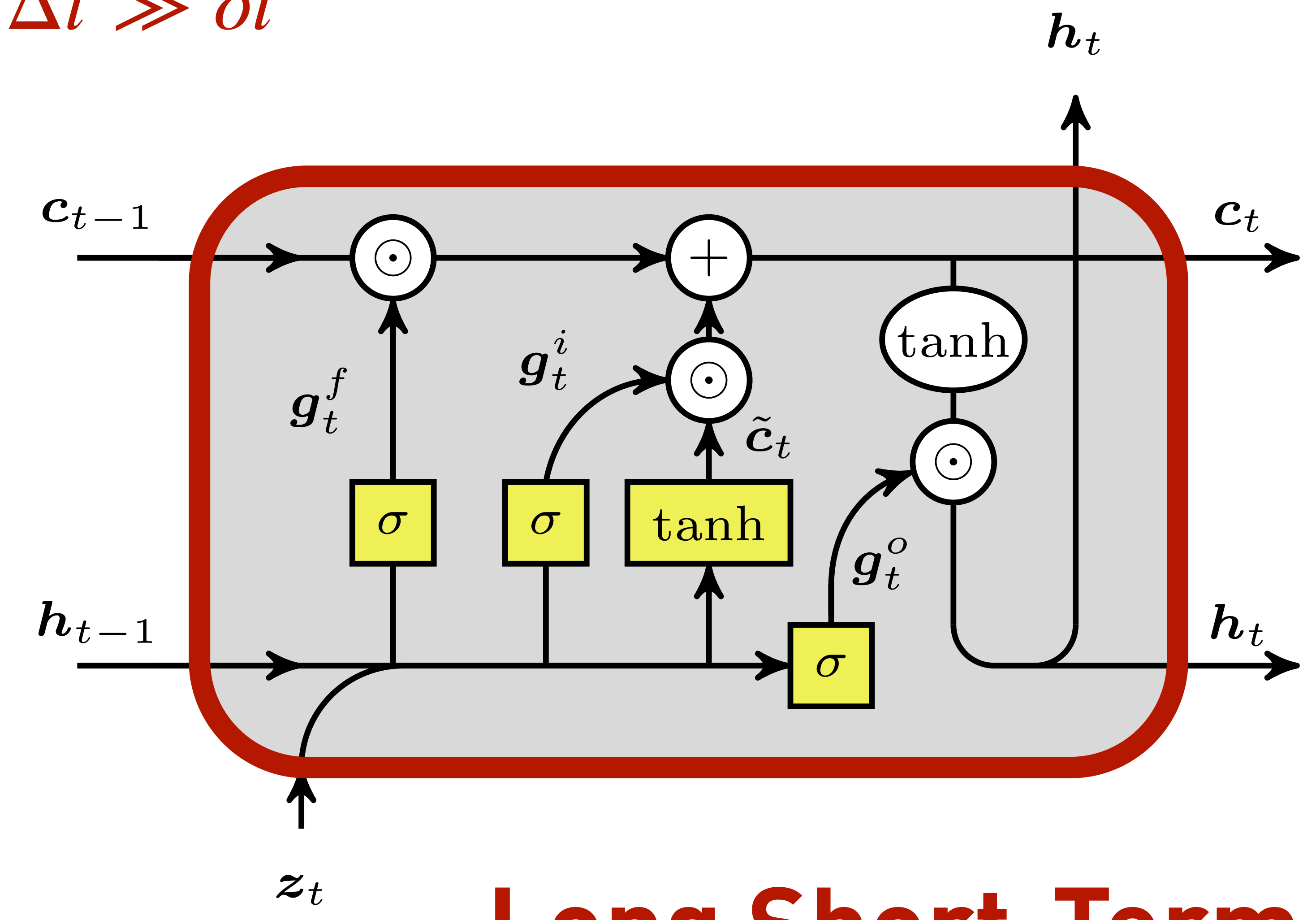


Parametrisation of $p(\mathbf{s}\,|\,\mathbf{z})$

- Coarse (latent) representation has limited information
- Mapping $\mathbf{z} \rightarrow \mathbf{s}$ can be probabilistic !
- Generative network
- $p(\mathbf{s}\,|\,\mathbf{z})$ as **mixture model**
- $$p(\mathbf{s}\,|\,\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathscr{N}\left(\mu_{\mathbf{s}}^k, \Sigma_{\mathbf{s}}^k\right)$$

RECURRENT NEURAL NETWORKS

- Non-linear non-Markovian dynamics $\mathbf{z}$ **(macro dynamics)**
- **Forecasting** using **RNNs**
- Tracking the **history** of the low order state $\mathbf{z}$ to model **non-Markovian** dynamics
- Forecasting $\mathbf{z}_{t+\Delta t}$ from short-term history
- Δt timestep of RNN, δt time step of micro dynamics
  $\Delta t \gg \delta t$



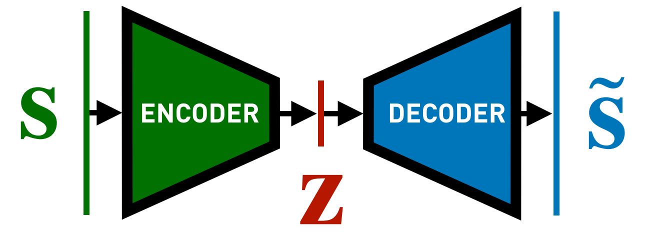
Long Short-Term Memory

Operators from Machine Learning

(CONVOLUTIONAL) AUTOENCODERS

High dimensional state

Reconstruction

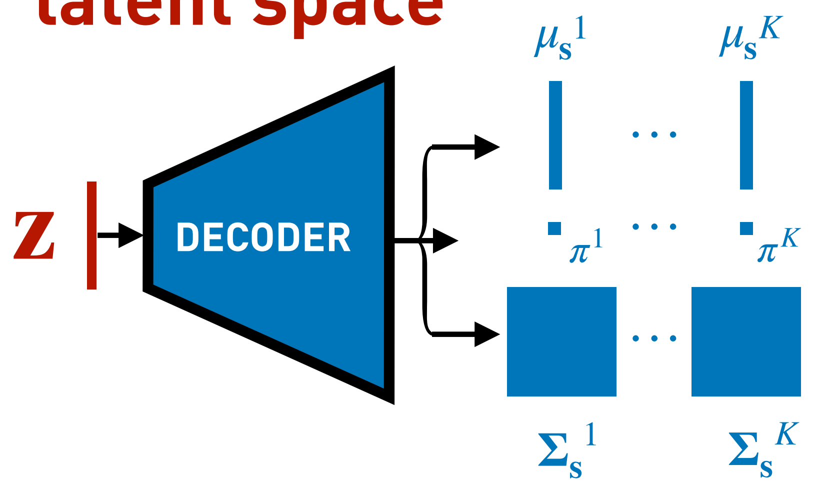


Low dimensional latent space

- Full high dimensional description of dynamical system system $\mathbf{s}$
- e.g. positions of atoms / micro scale / angles, bonds
- Loss Function $\mathscr{L} = |\mathbf{s} - \tilde{\mathbf{s}}|_2^2$
- Ideally after training $\mathbf{s} \approx \tilde{\mathbf{s}}$

MIXTURE DENSITY NETWORKS

Low dimensional latent space

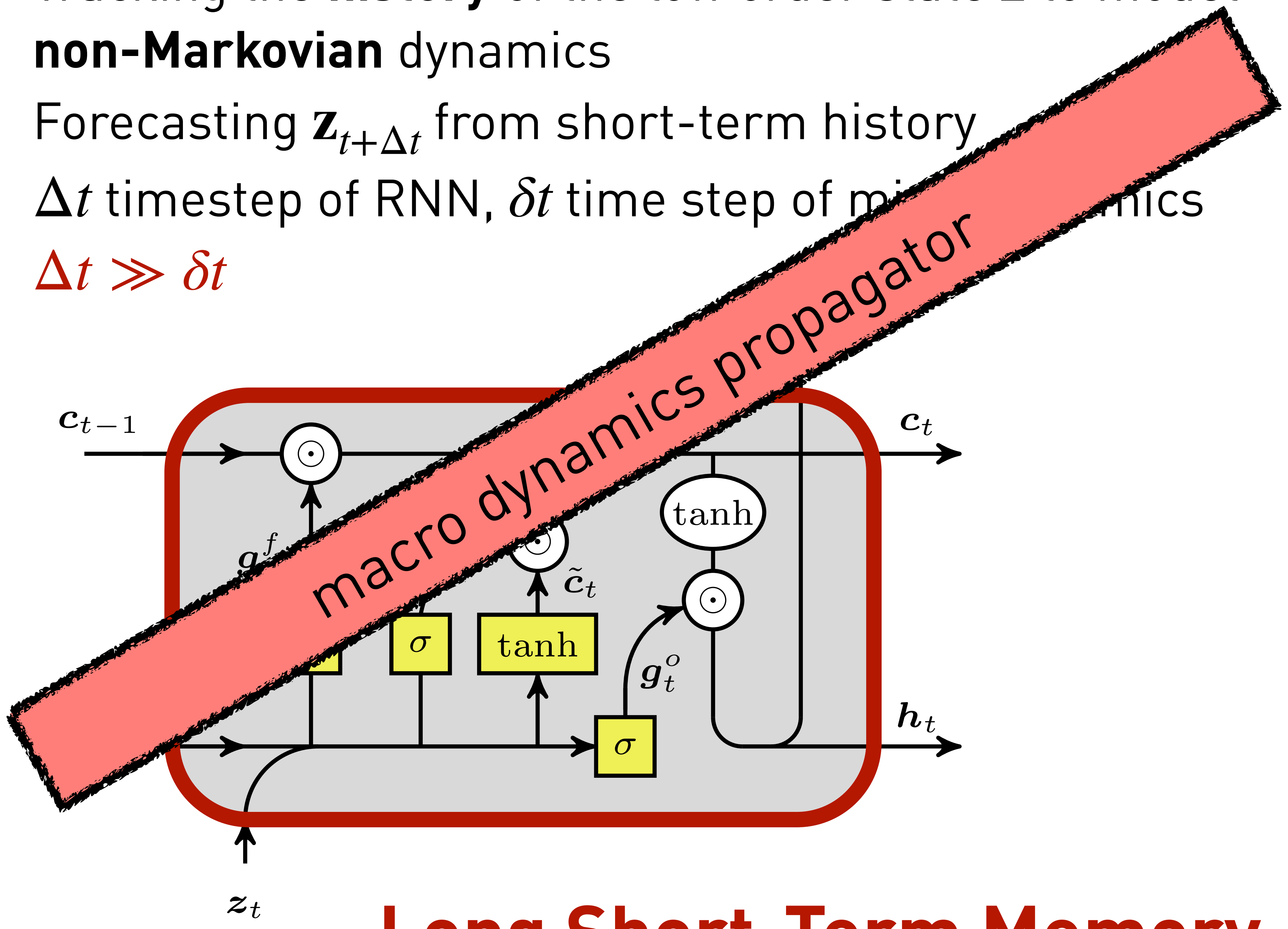


Parametrisation of $p(\mathbf{s}\,|\,\mathbf{z})$

- Coarse (latent) representation has limited information
- Mapping $\mathbf{z} \to \mathbf{s}$ can be probabilistic !
- Generative network
- $p(\mathbf{s}\,|\,\mathbf{z})$ as **mixture model**
- $p(\mathbf{s}\,|\,\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathscr{N}\left(\mu_{\mathbf{s}}^k, \Sigma_{\mathbf{s}}^k\right)$

RECURRENT NEURAL NETWORKS

- Non-linear non-Markovian dynamics $\mathbf{z}$ **(macro dynamics)**
- **Forecasting** using **RNNs**
- Tracking the **history** of the low order state $\mathbf{z}$ to model **non-Markovian** dynamics
- Forecasting $\mathbf{z}_{t+\Delta t}$ from short-term history
- Δt timestep of RNN, δt time step of micro dynamics
  $\Delta t \gg \delta t$



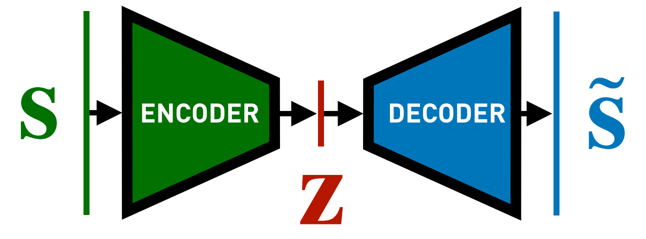
Long Short-Term Memory

Operators from Machine Learning

(CONVOLUTIONAL) AUTOENCODERS

High dimensional state

Reconstruction



Low dimensional latent spac

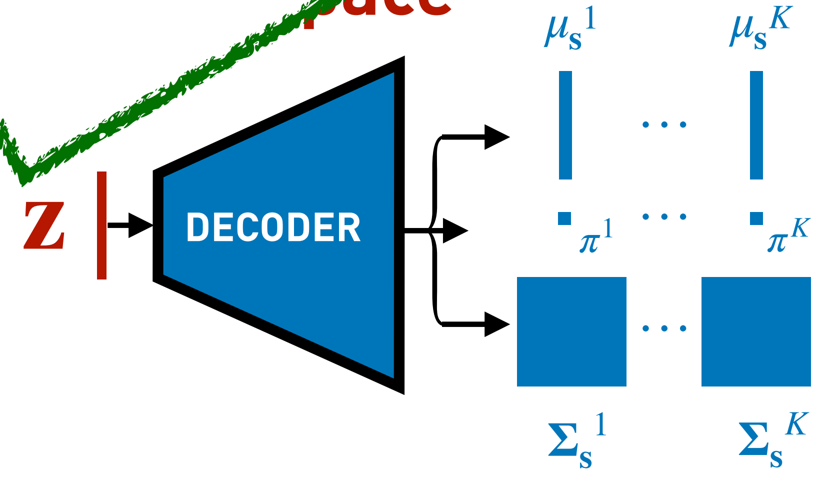
- Full high di... descri...
- ...of atoms / ...ate / angles, bonds
- ...s Function $\mathscr{L} = |\mathbf{s} - \tilde{\mathbf{s}}|_2^2$
- Ideally after training $\mathbf{s} \approx \tilde{\mathbf{s}}$

RESTRICTING/AVERAGING

(micro → macro)

...E DENSITY NETWORKS

...at ...ace

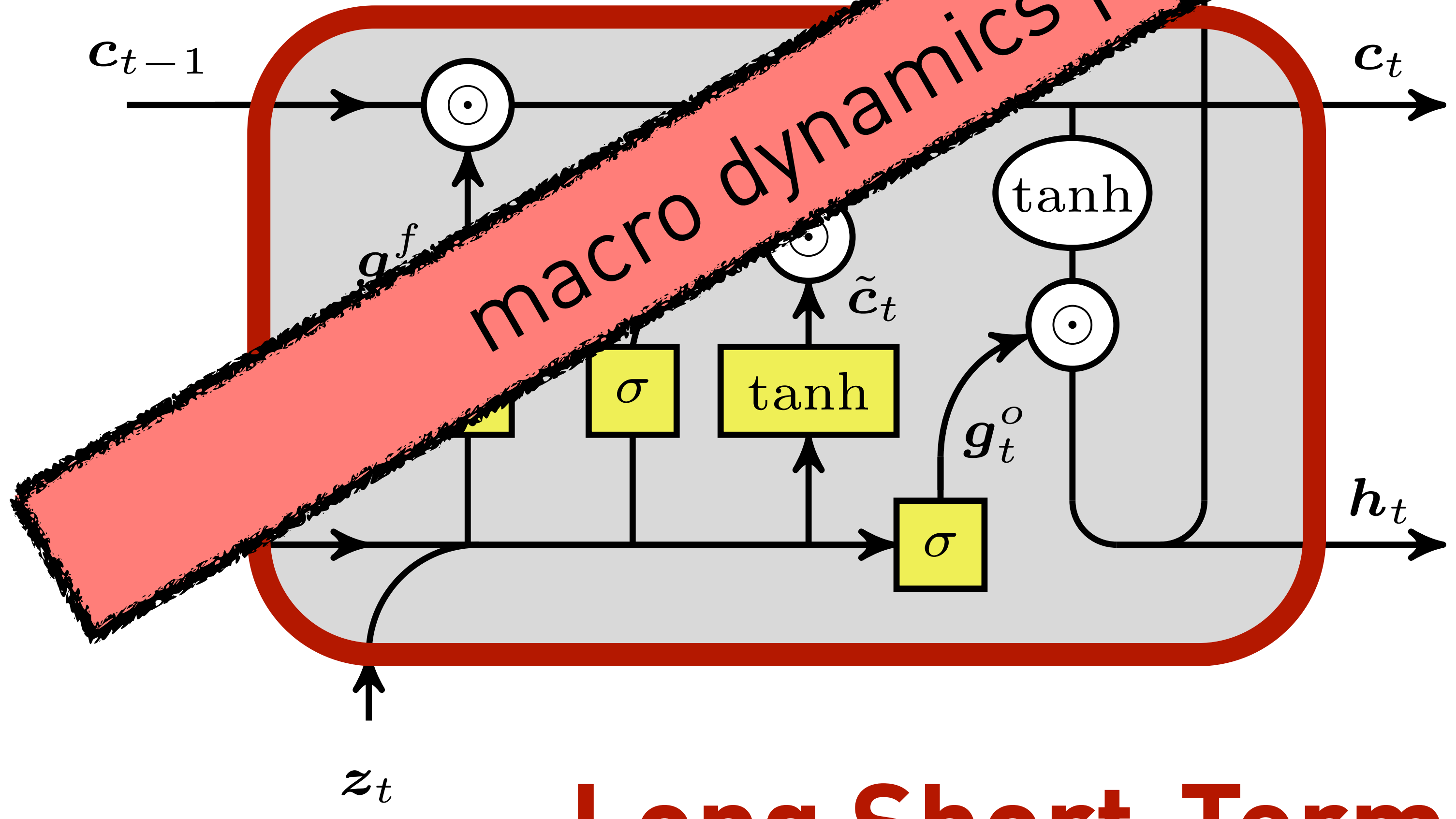


Parametrisation of $p(\mathbf{s}\,|\,\mathbf{z})$

- Coarse (latent) representation has limited information
- Mapping $\mathbf{z} \rightarrow \mathbf{s}$ can be probabilistic !
- Generative network
- $p(\mathbf{s}\,|\,\mathbf{z})$ as **mixture model**
- $p(\mathbf{s}\,|\,\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathscr{N}\left(\mu_{\mathbf{s}}^k, \Sigma_{\mathbf{s}}^k\right)$

RECURRENT NEURAL NETWORKS

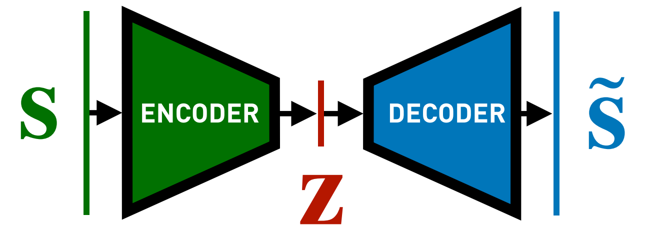
- Non-linear non-Markovian dynamics $\mathbf{z}$ **(macro dynamics)**
- **Forecasting** using **RNNs**
- Tracking the **history** of the low order state $\mathbf{z}$ to model **non-Markovian** dynamics
- Forecasting $\mathbf{z}_{t+\Delta t}$ from short-term history
- Δt timestep of RNN, δt time step of mi...mics
  $\Delta t \gg \delta t$



Long Short-Term Memory

Operators from Machine Learning

(CONVOLUTIONAL) AUTOENCODERS

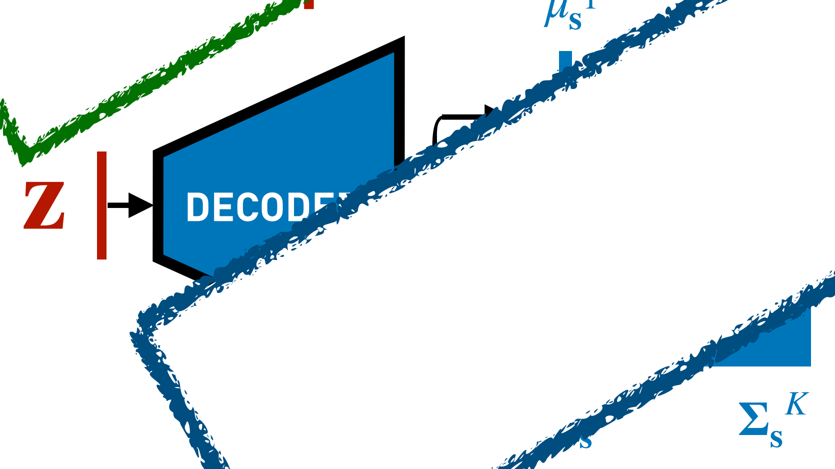


- Full high di... descri...
- ...of atoms / ...ate / angles ...
- ...s Function ...
- Ideally aft...

RESTRICTING/AVERAGING (micro → macro)

LIFTING (macro → micro)

...E DENSIT... ...RKS

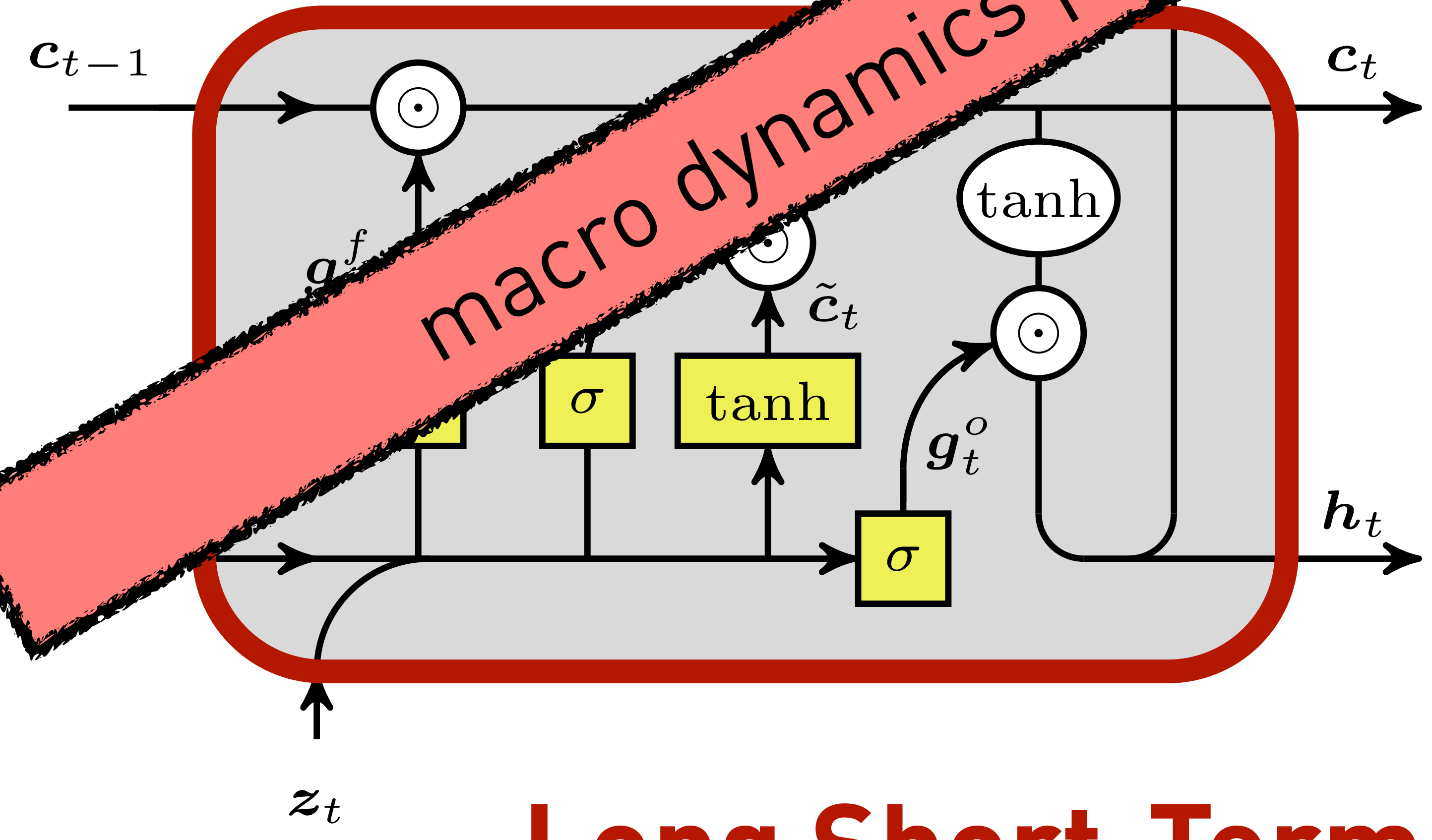


- ...nt) representation has ...nformation
- ...pping $\mathbf{z} \to \mathbf{s}$ can be probabilistic ! Generative network
- $p(\mathbf{s}\,|\,\mathbf{z})$ as **mixture model**
- $p(\mathbf{s}\,|\,\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\mu_{\mathbf{s}}^k, \Sigma_{\mathbf{s}}^k\right)$

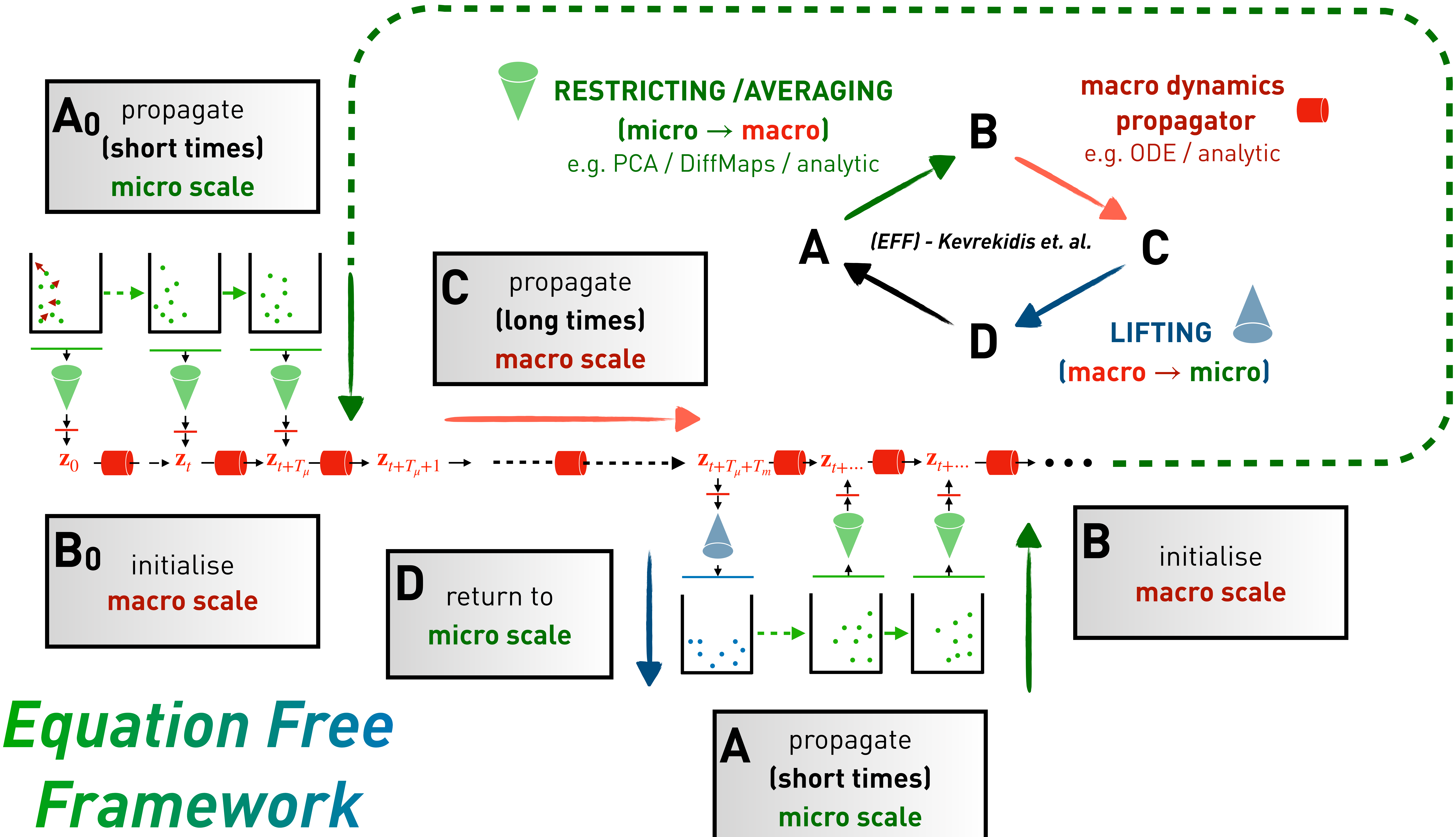
RECURRENT NEURAL NETWORKS

- Non-linear non-Markovian dynamics $\mathbf{z}$ **(macro dynamics)**
- **Forecasting** using **RNNs**
- Tracking the **history** of the low order state $\mathbf{z}$ to model **non-Markovian** dynamics
- Forecasting $\mathbf{z}_{t+\Delta t}$ from short-term history
- Δt timestep of RNN, δt time step of mi... ...mics
  $\Delta t \gg \delta t$

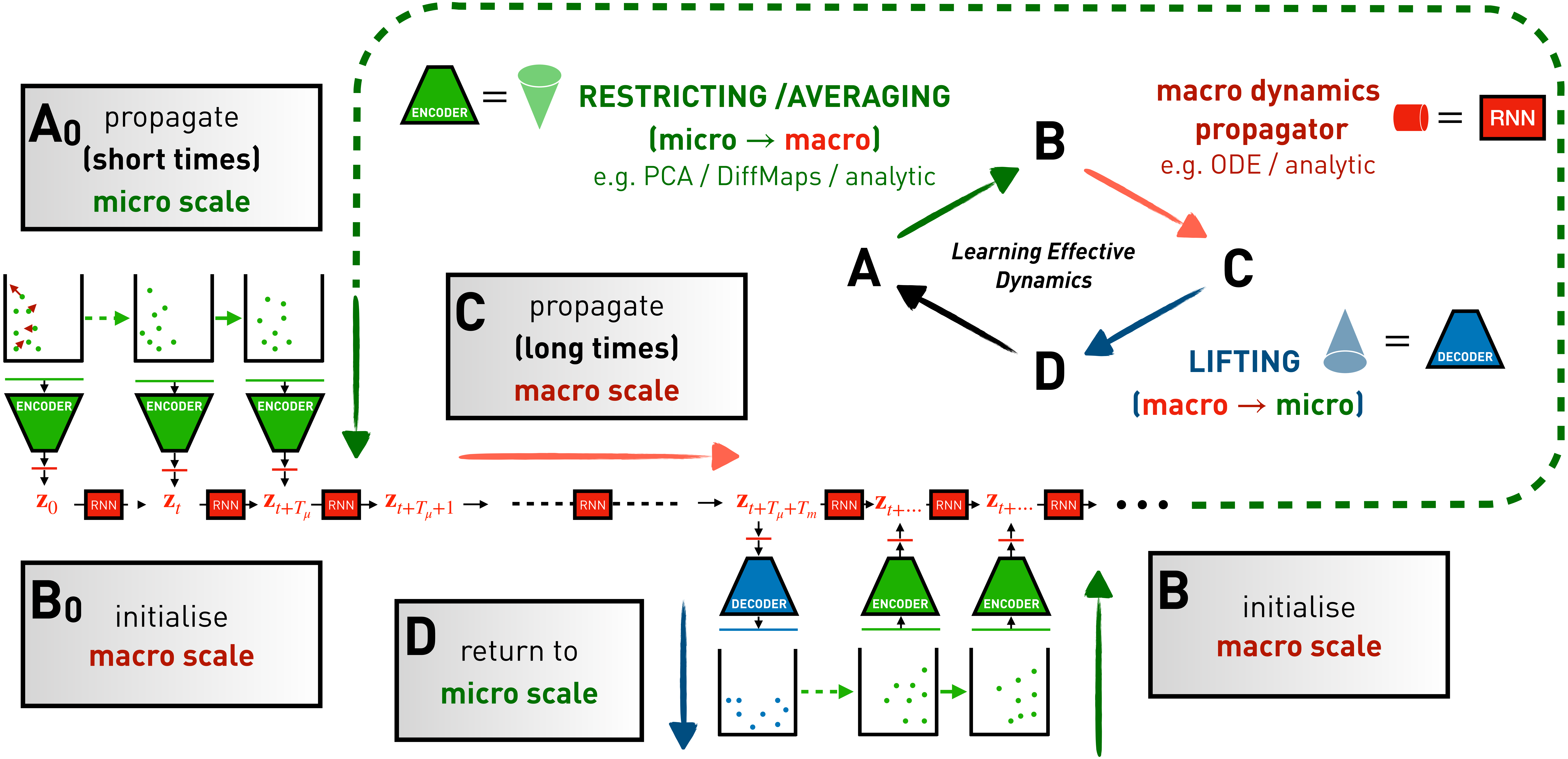
macro dynamics propagator



Long Short-Term Memory



RESTRICTING /AVERAGING
(micro → macro)
e.g. PCA / DiffMaps / analytic
macro dynamics
propagator
e.g. ODE / analytic
A
B
C
D
(EFF) - Kevrekidis et. al.
LIFTING
(macro → micro)
A0 propagate
(short times)
micro scale
C propagate
(long times)
macro scale
z_0
z_t
z_{t+T_μ}
$z_{t+T_\mu+1}$
$z_{t+T_\mu+T_m}$
$z_{t+\dots}$
$z_{t+\dots}$
B0 initialise
macro scale
D return to
micro scale
B initialise
macro scale
A propagate
(short times)
micro scale
Equation Free
Framework

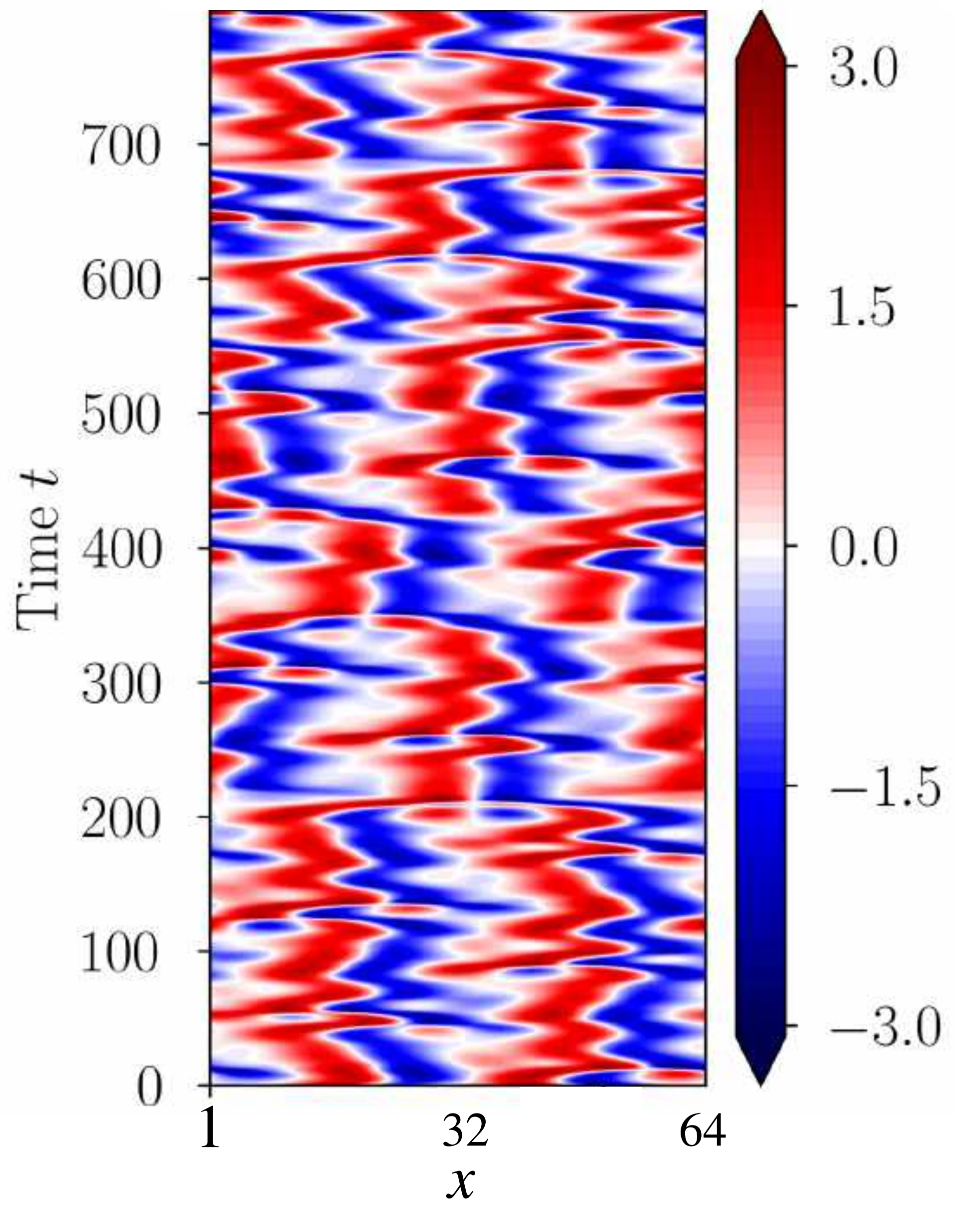


Learning Effective Dynamics (LED)

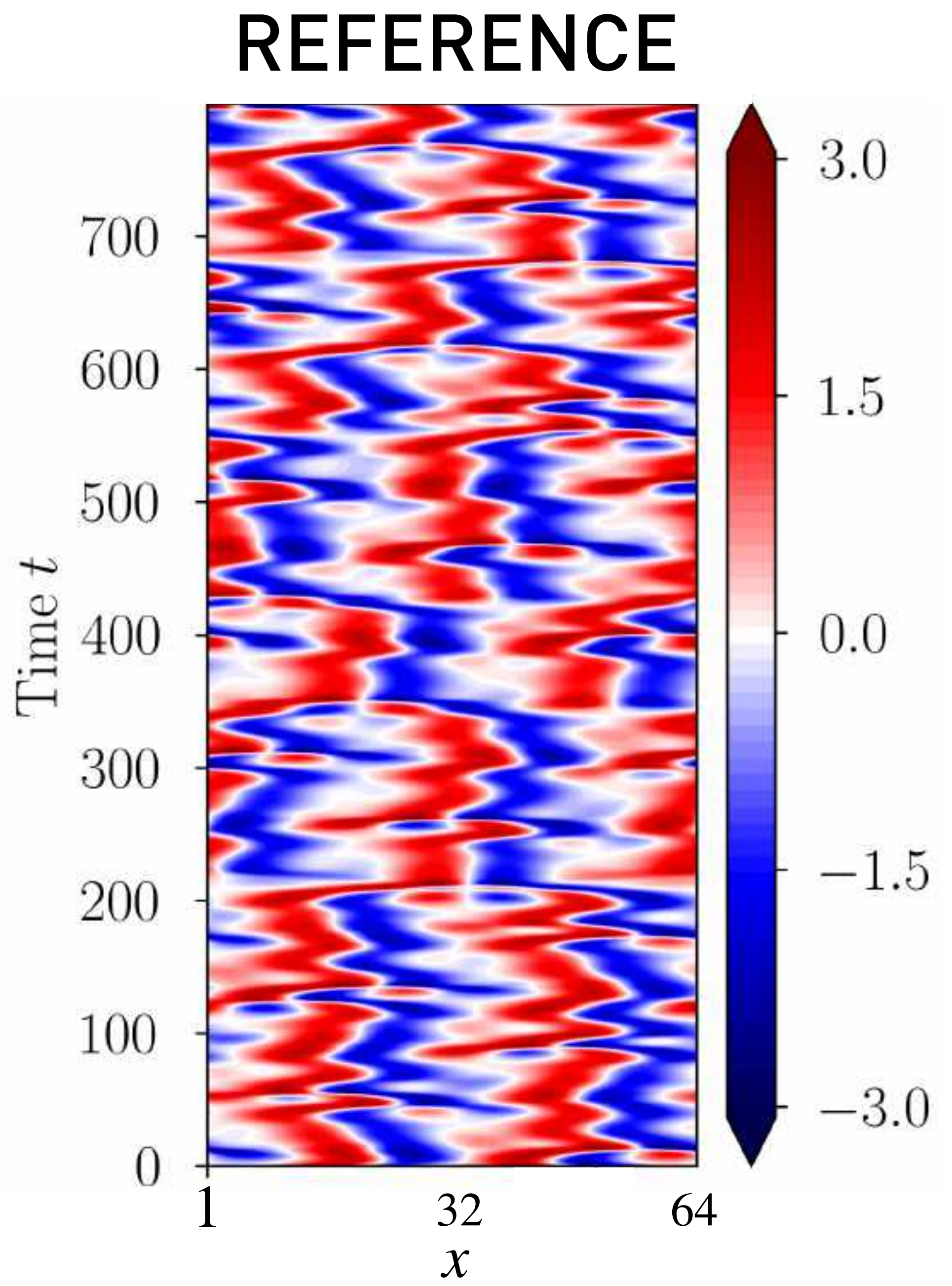
Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

REFERENCE



Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

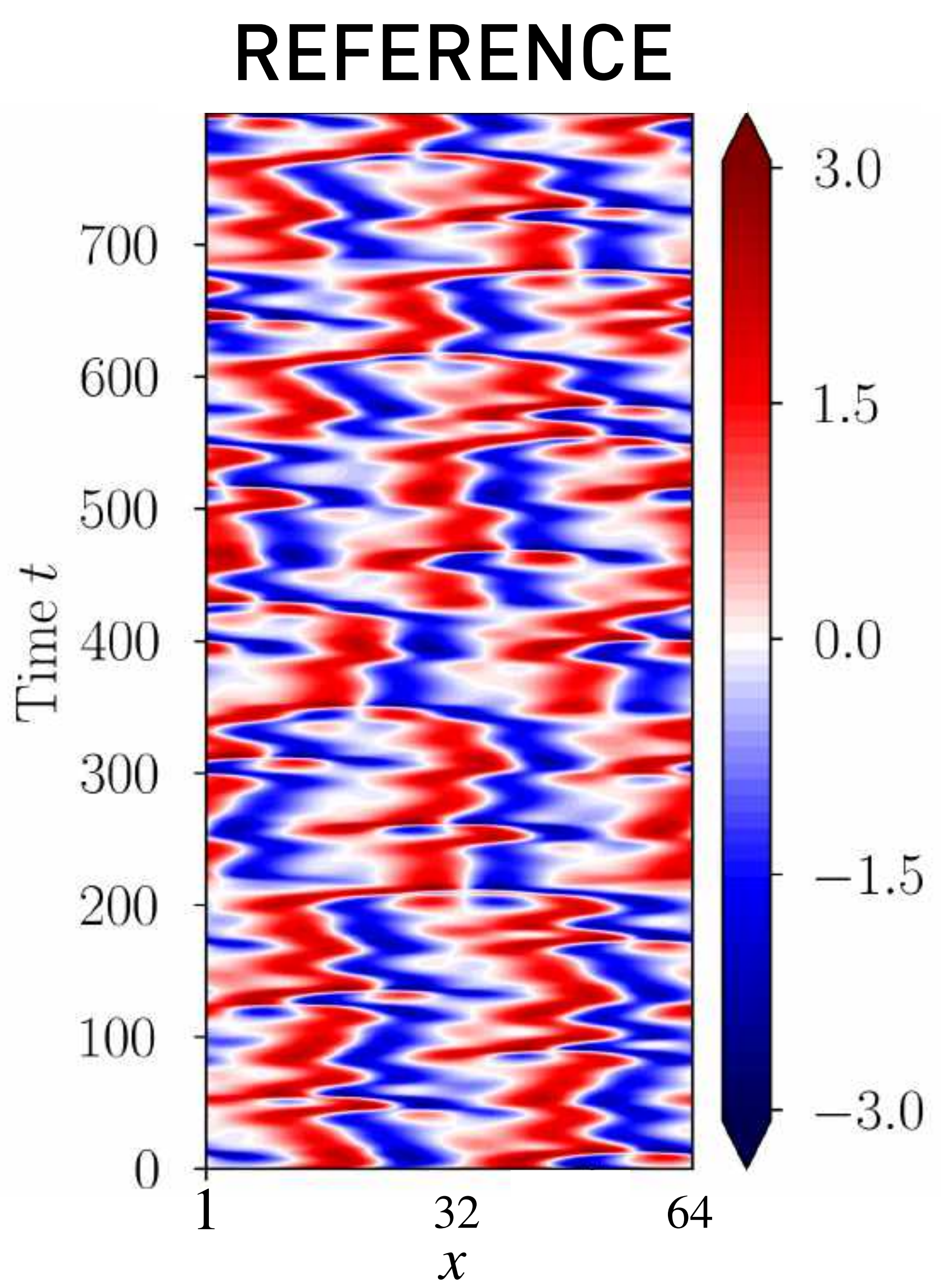


- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult

[1] **Robinson, J. C.** "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, 190–193 (1994).
[2] **Linot, A. J. & Graham, M. D.** "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101*, 062209 (2020).

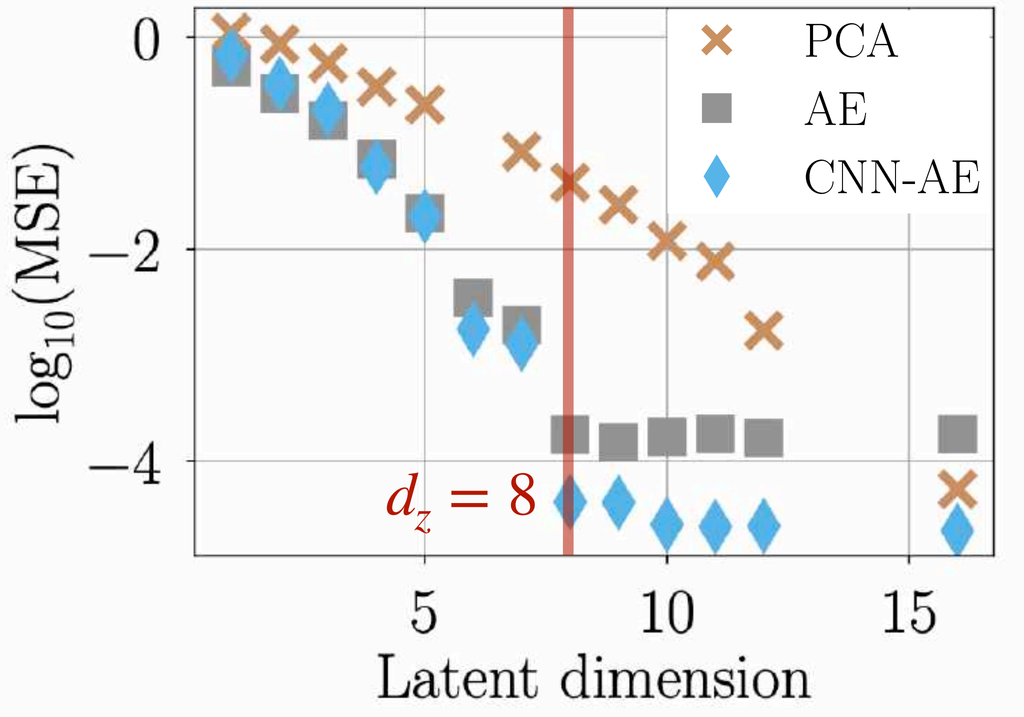
Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)



- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult

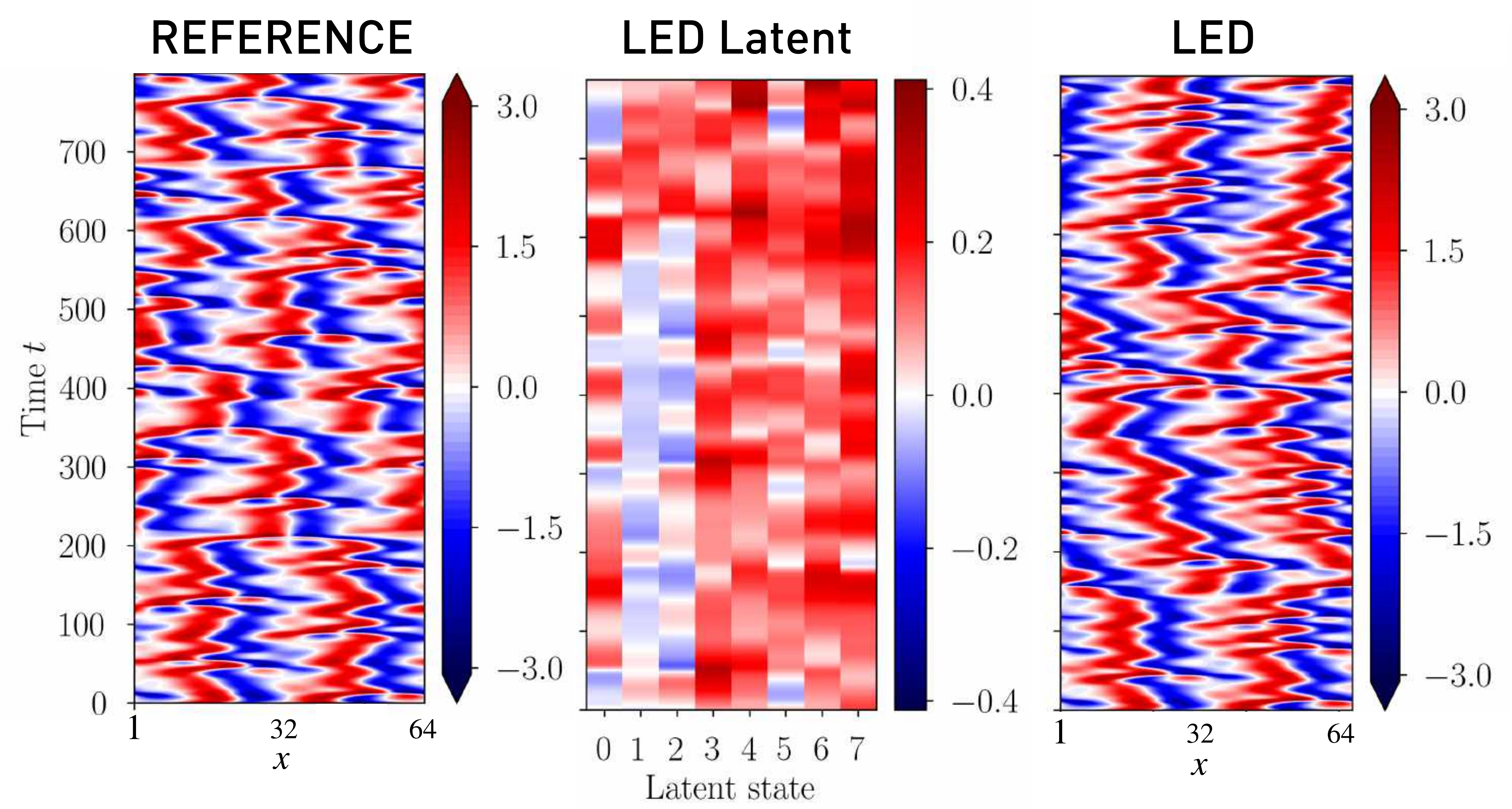
The AE error saturates after $d_z = 8$



[1] **Robinson, J. C.** "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, *190–193 (1994).*
[2] **Linot, A. J. & Graham, M. D.** "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101, 062209 (2020).*

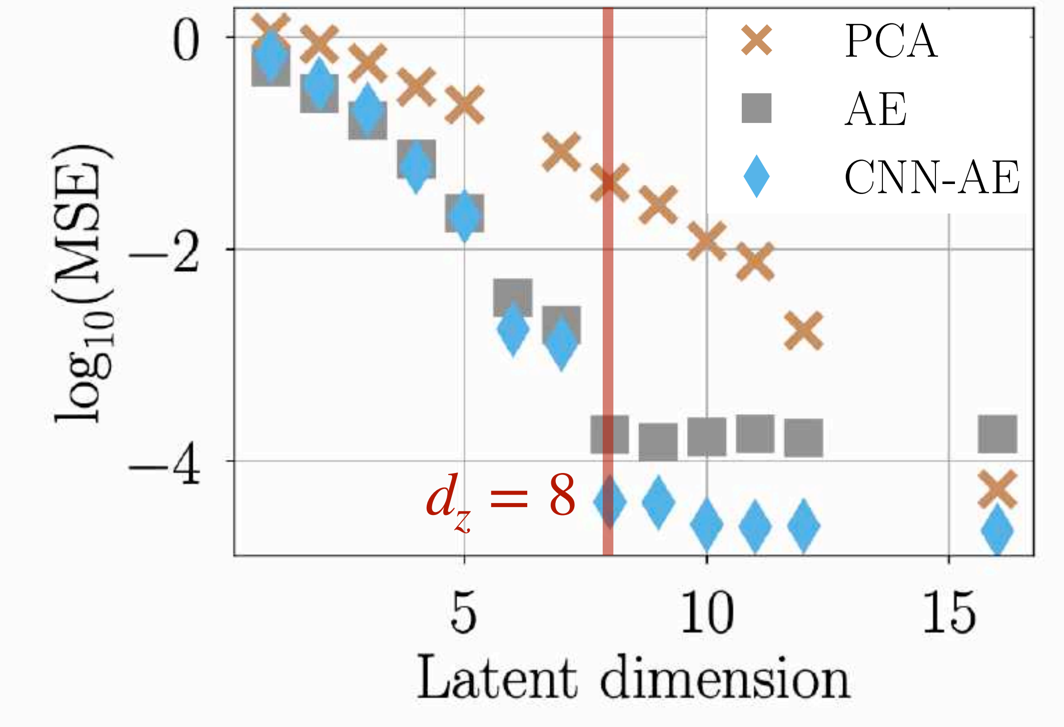
Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)



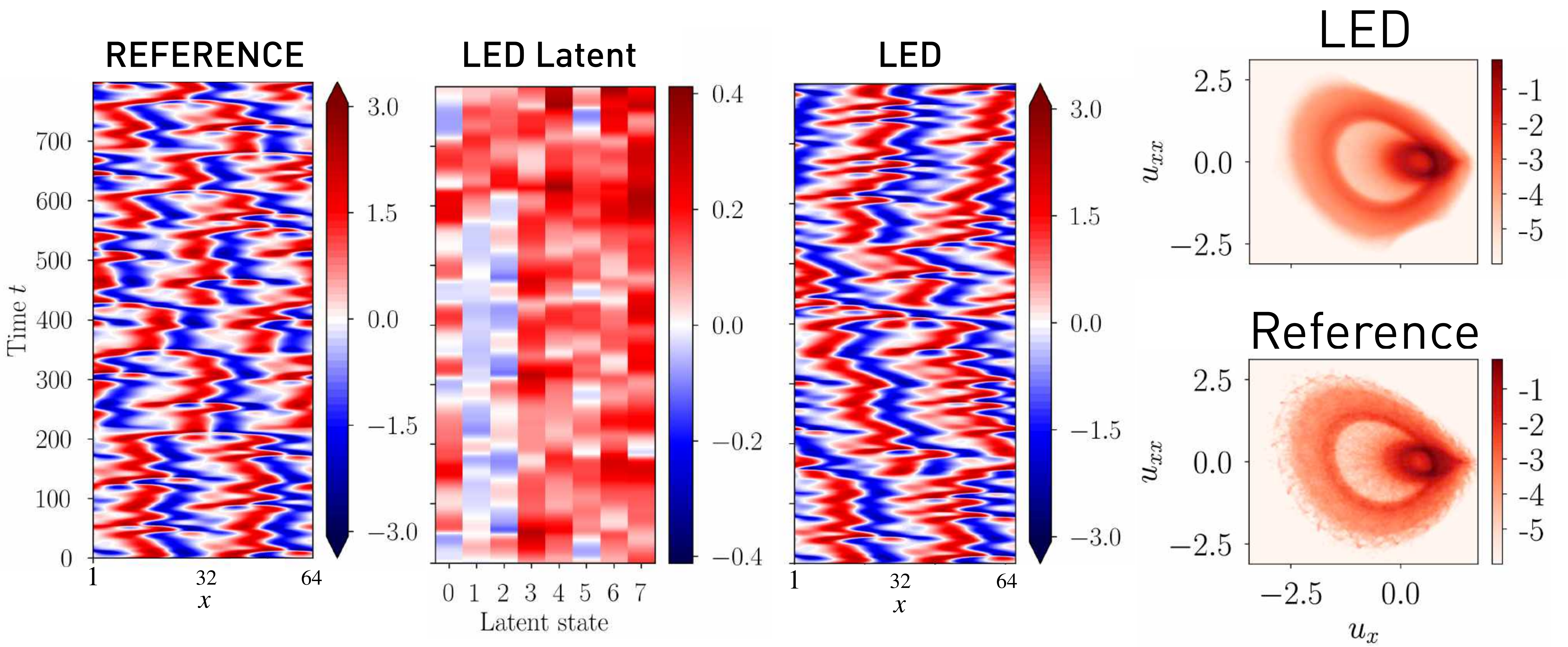
- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult
- LED can identify and **reconstruct the dynamics** on an **8 dimensional** manifold

The AE error saturates after $d_z = 8$



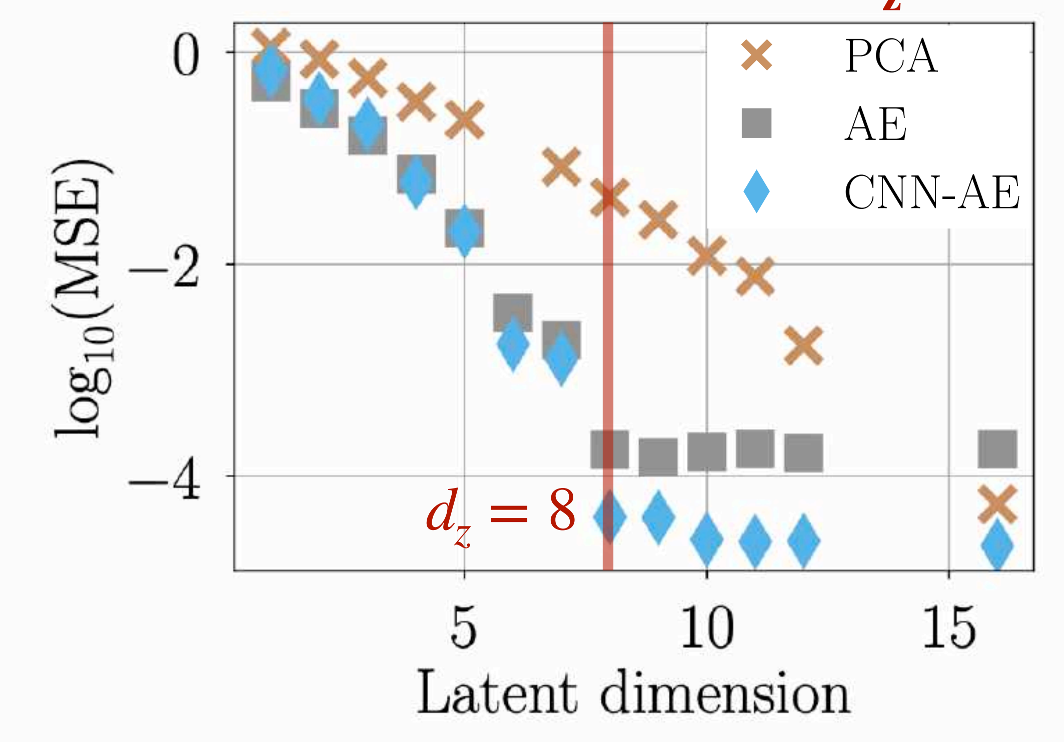
[1] Robinson, J. C. "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, *190–193 (1994).*
[2] Linot, A. J. & Graham, M. D. "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101, 062209 (2020).*

Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)



- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult
- LED can identify and **reconstruct the dynamics** on an **8 dimensional** manifold
- Reproducing the statistics/long-term climate accurately

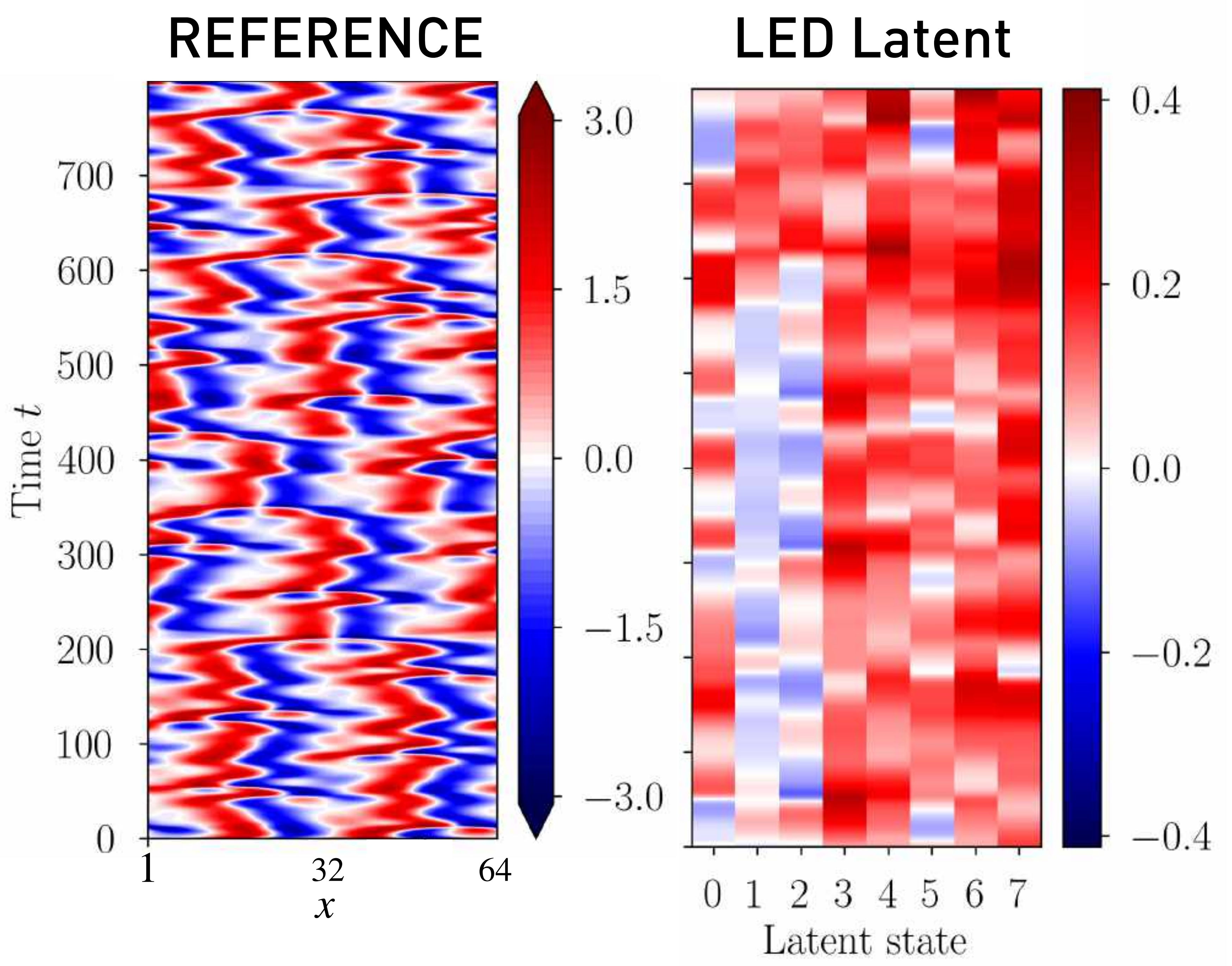
The AE error saturates after $d_z = 8$

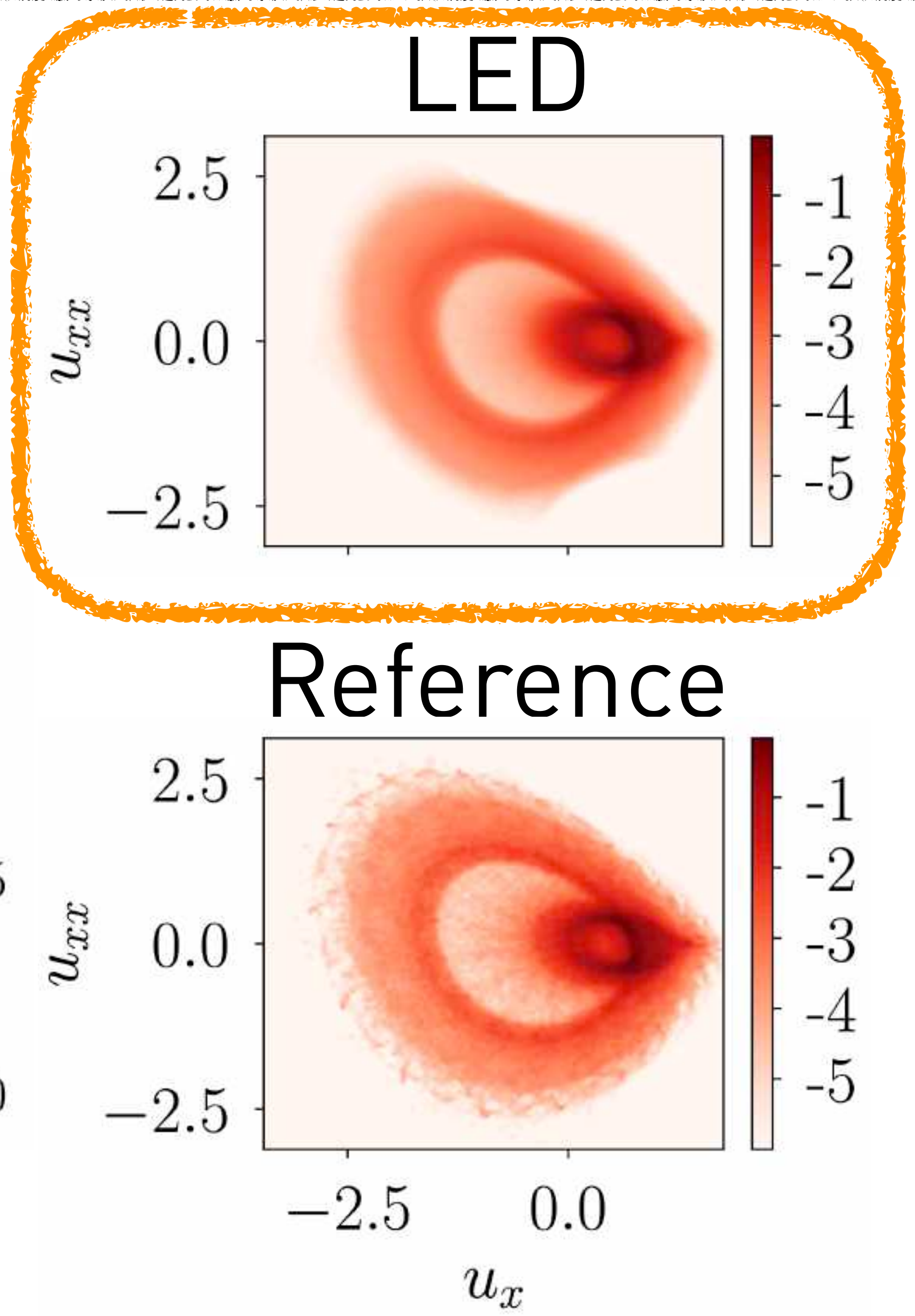


[1] **Robinson, J. C.** "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, *190–193 (1994).*
[2] **Linot, A. J. & Graham, M. D.** "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101, 062209 (2020).*

Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

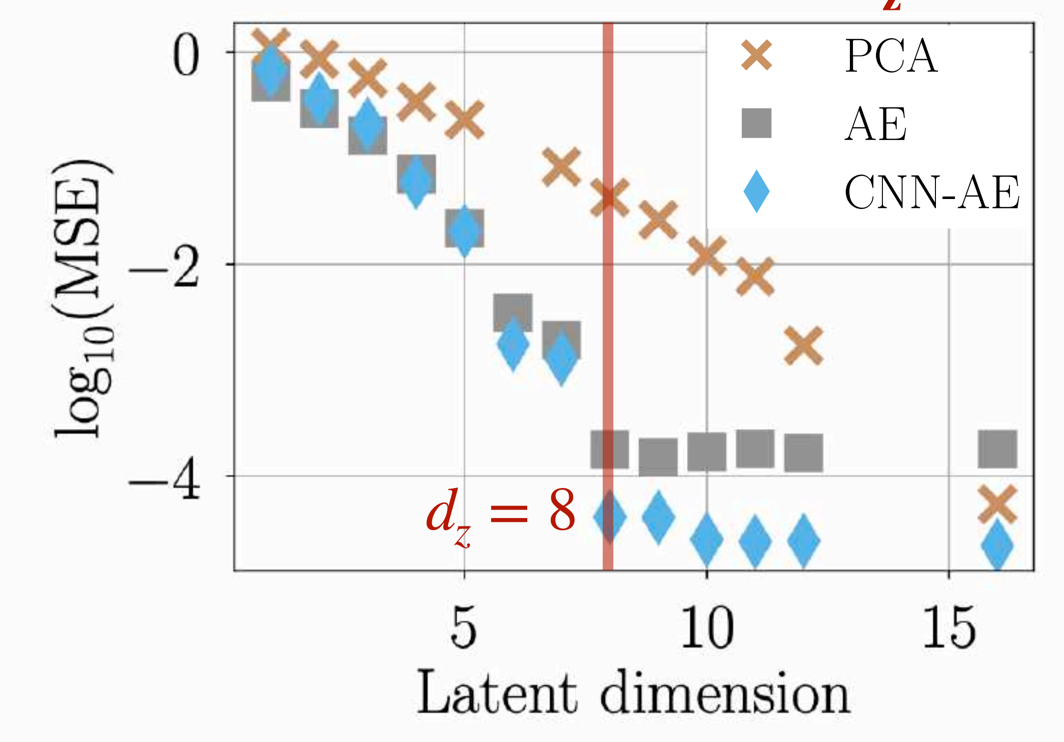
PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)





- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult
- LED can identify and **reconstruct the dynamics** on an **8 dimensional** manifold
- Reproducing the statistics/long-term climate accurately

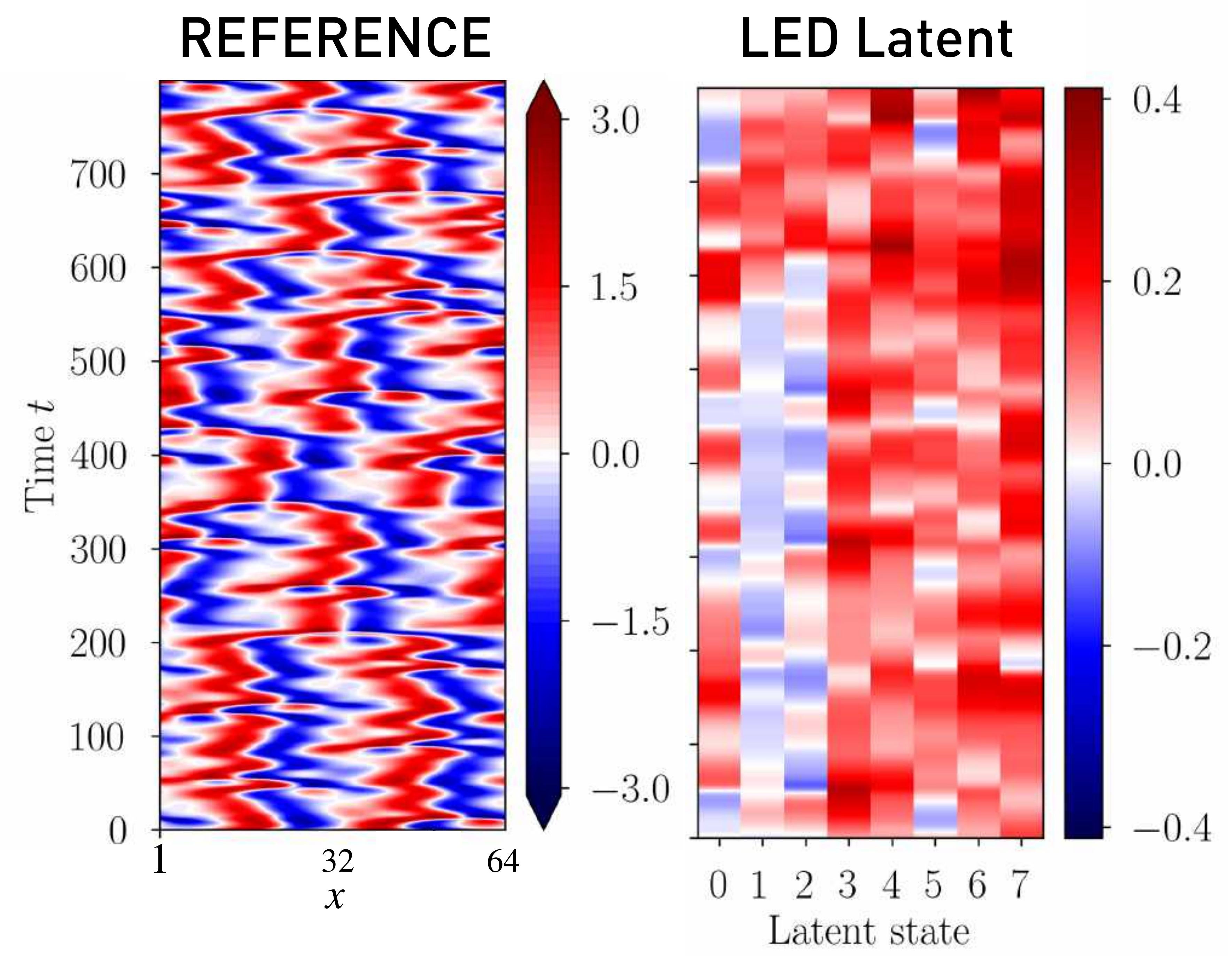
The AE error saturates after $d_z = 8$

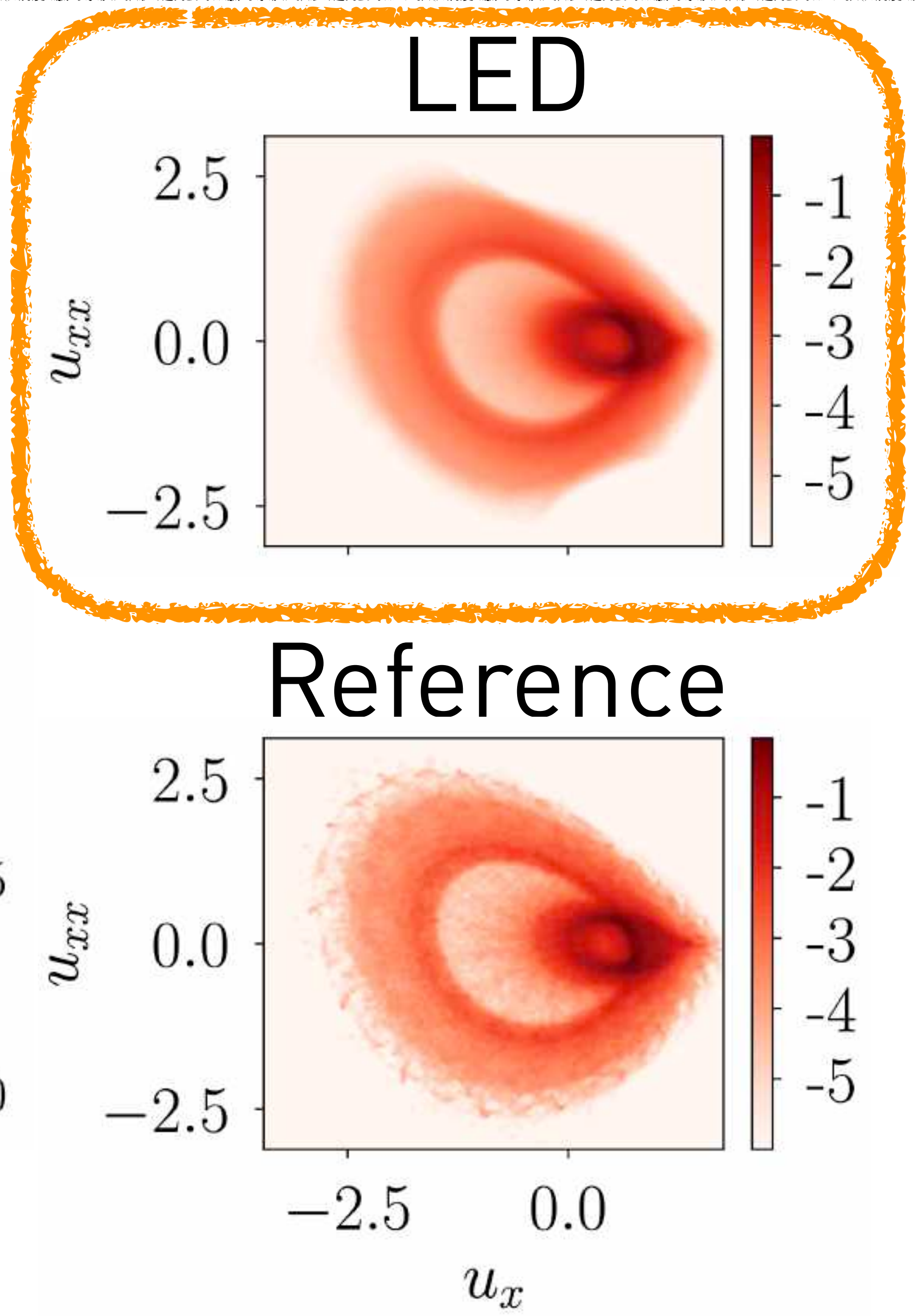


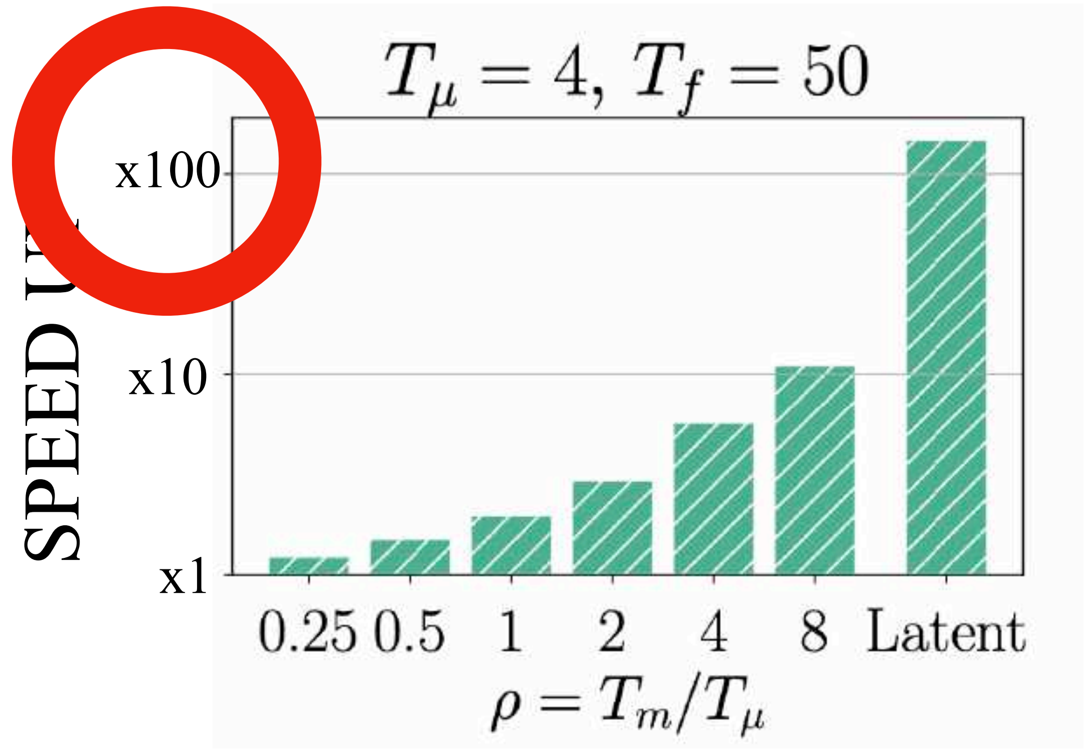
[1] **Robinson, J. C.** "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, *190–193 (1994).*
[2] **Linot, A. J. & Graham, M. D.** "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101, 062209 (2020).*

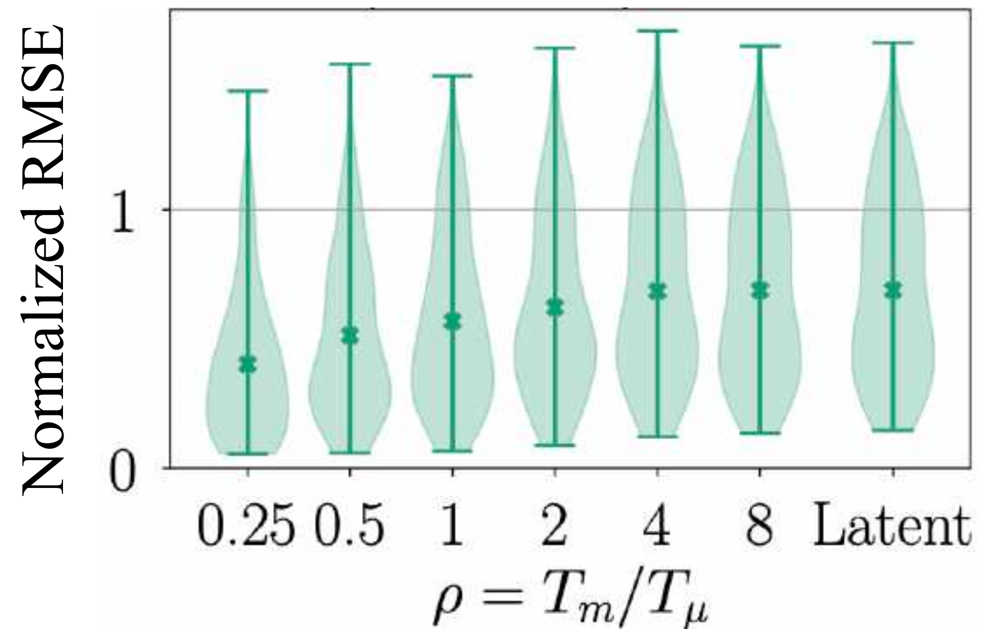
Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

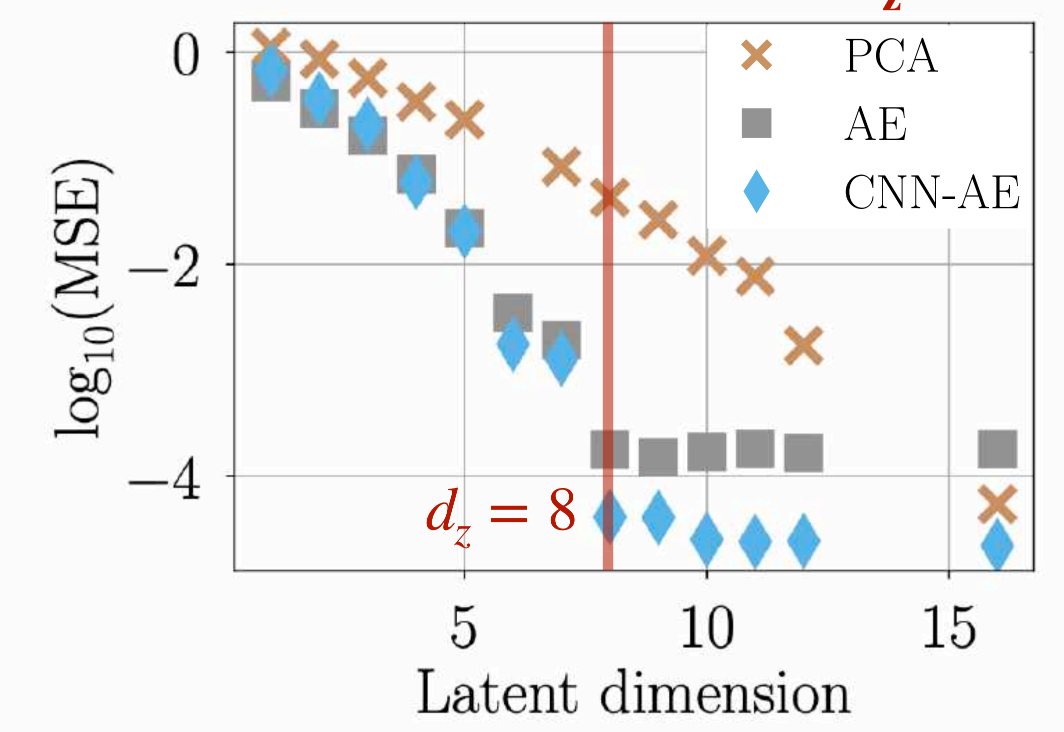








The AE error saturates after $d_z = 8$

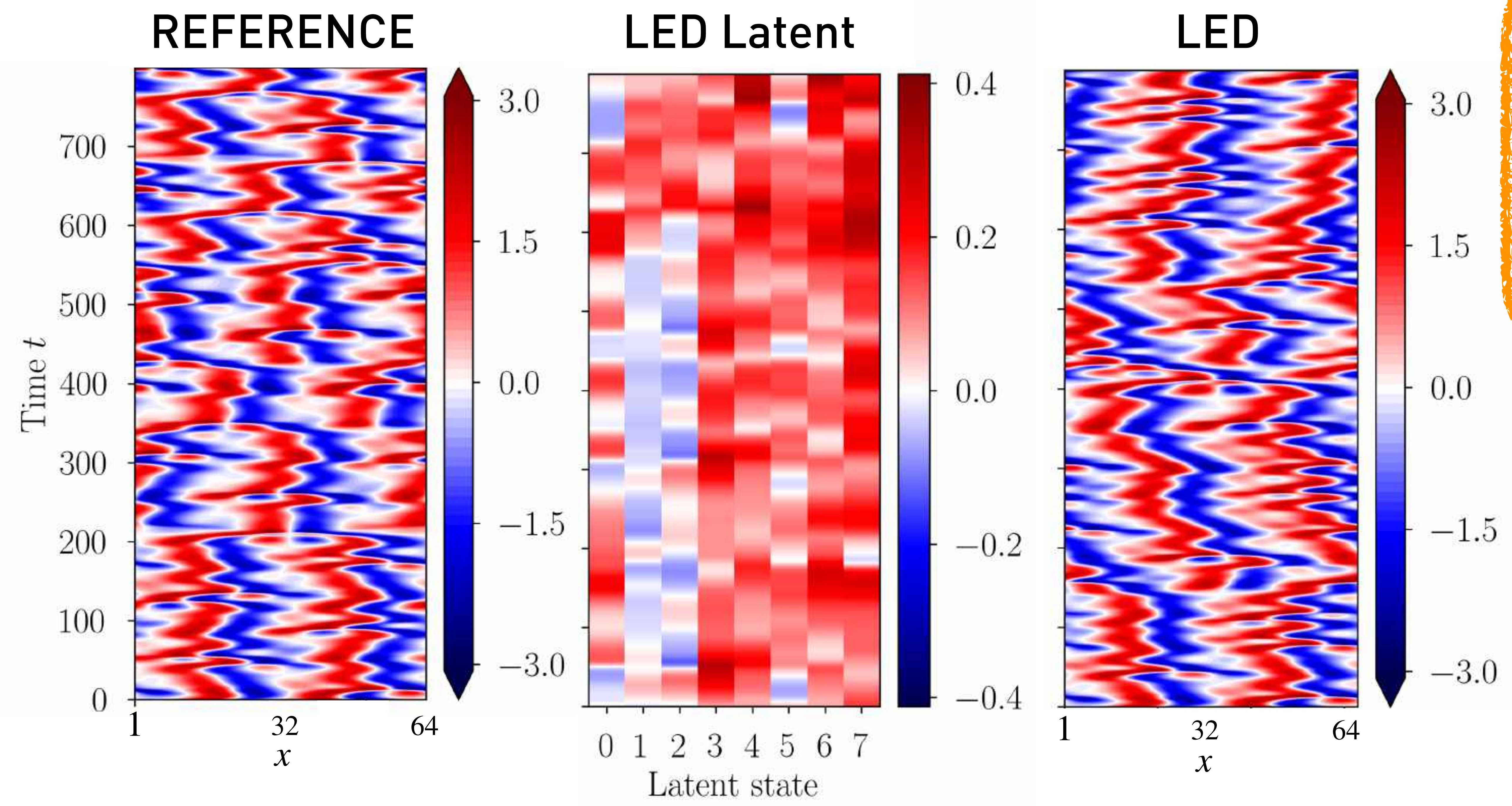


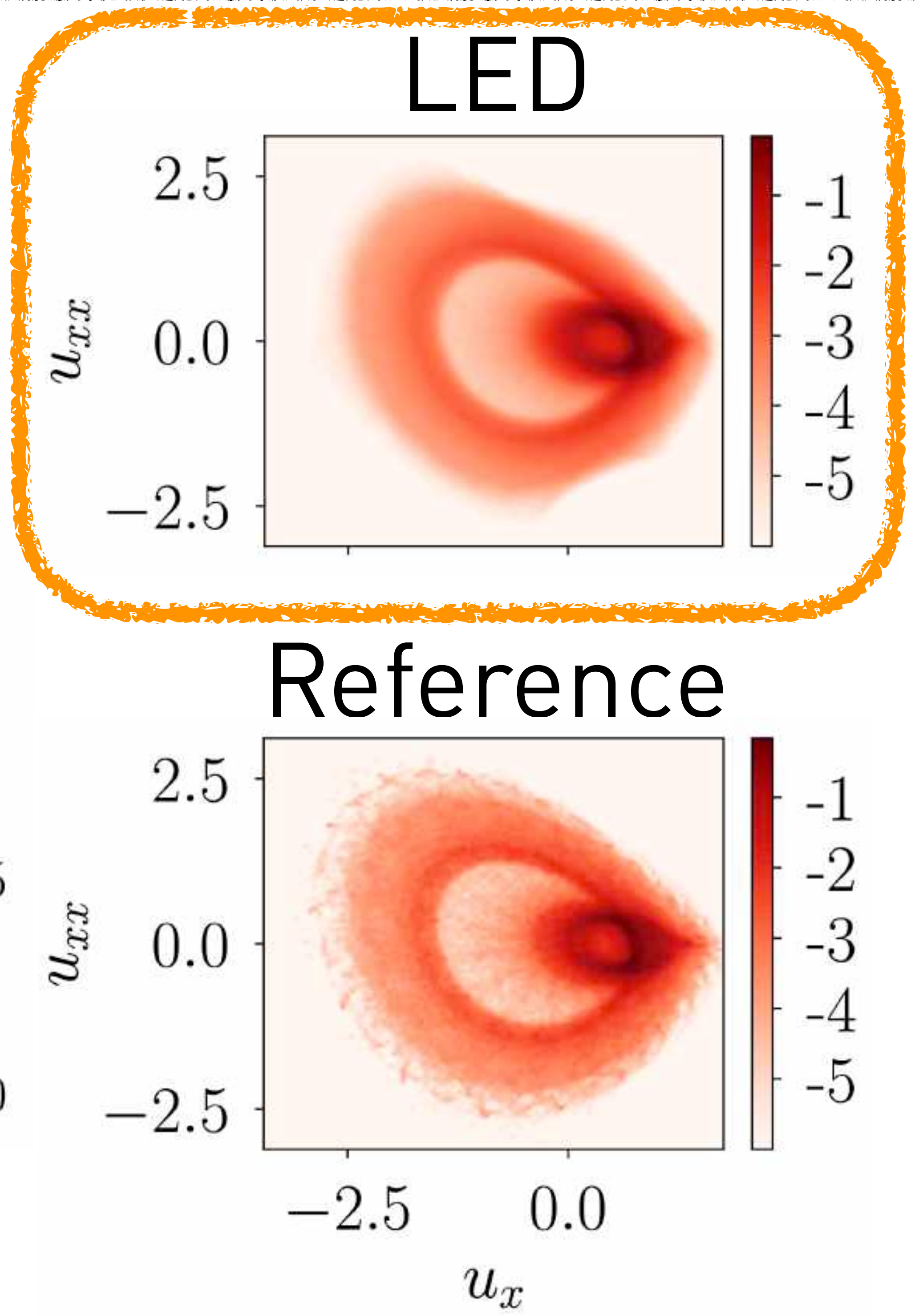
- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult
- LED can identify and **reconstruct the dynamics** on an **8 dimensional** manifold
- Reproducing the statistics/long-term climate accurately
- **Two orders** of magnitude faster compared to the stiff ODE solver used

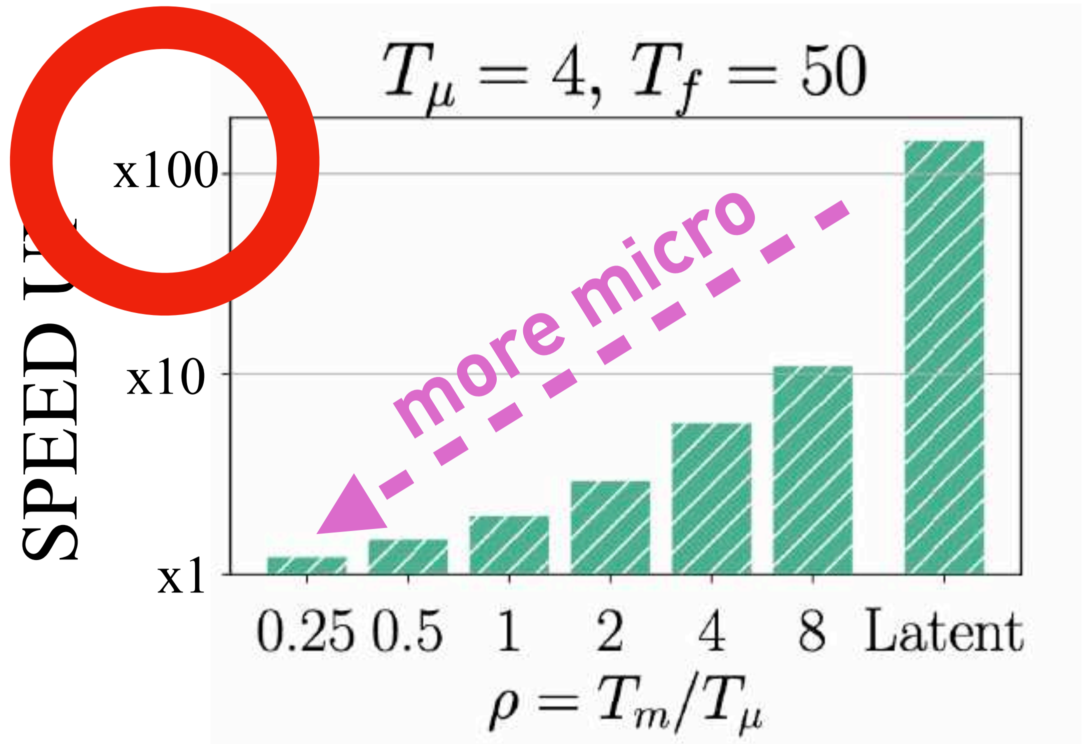
[1] **Robinson, J. C.** "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, *190–193 (1994).*
[2] **Linot, A. J. & Graham, M. D.** "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101, 062209 (2020).*

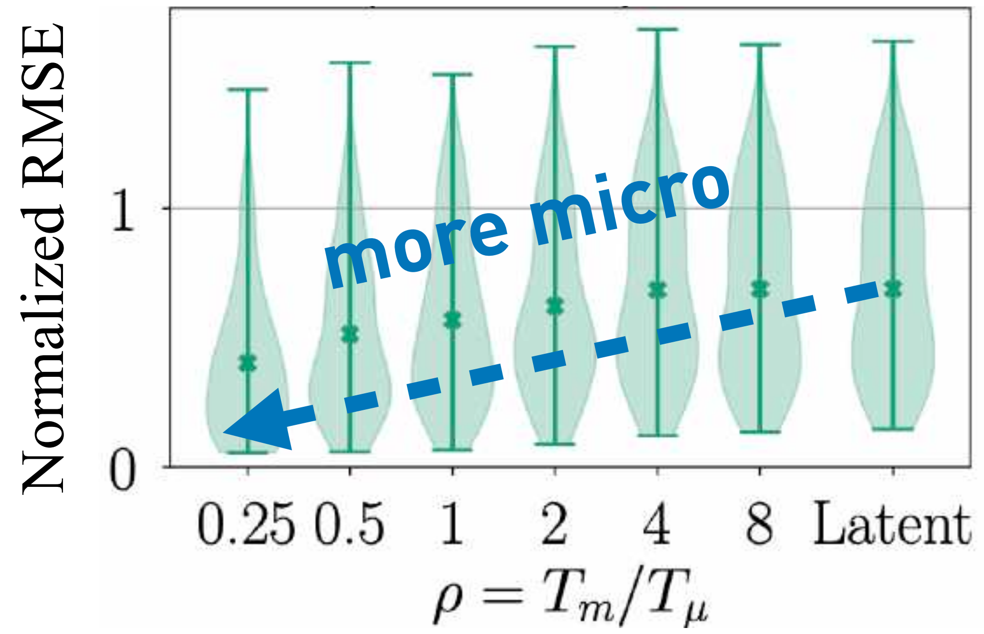
PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

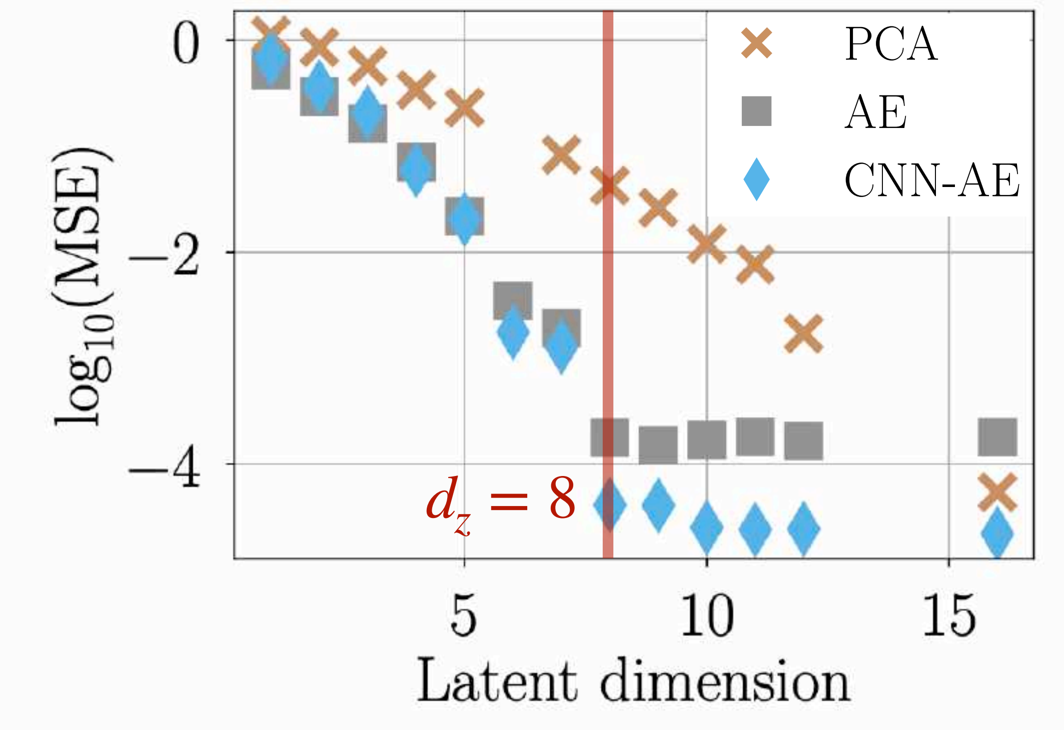








The AE error saturates after $d_z = 8$

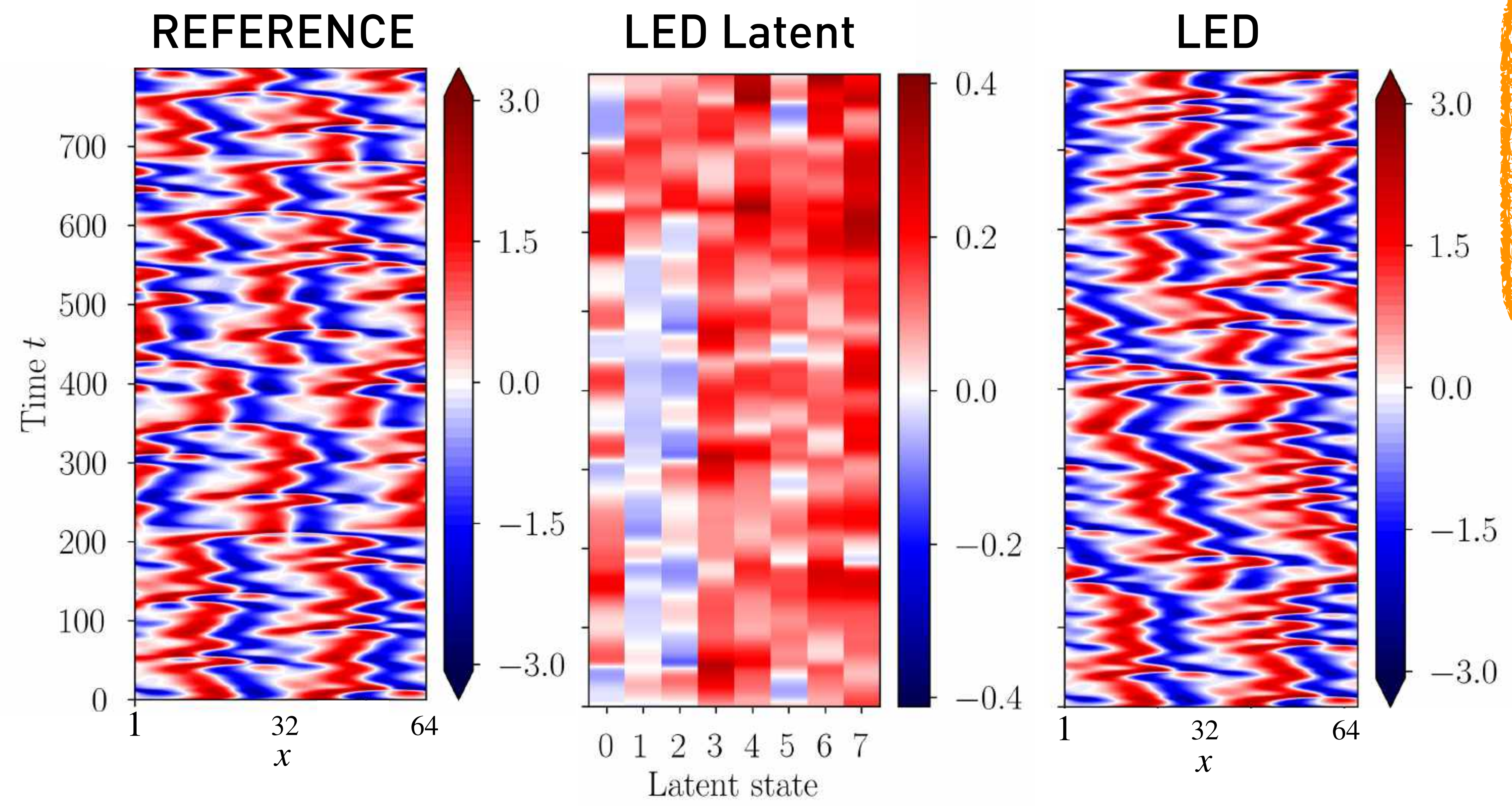


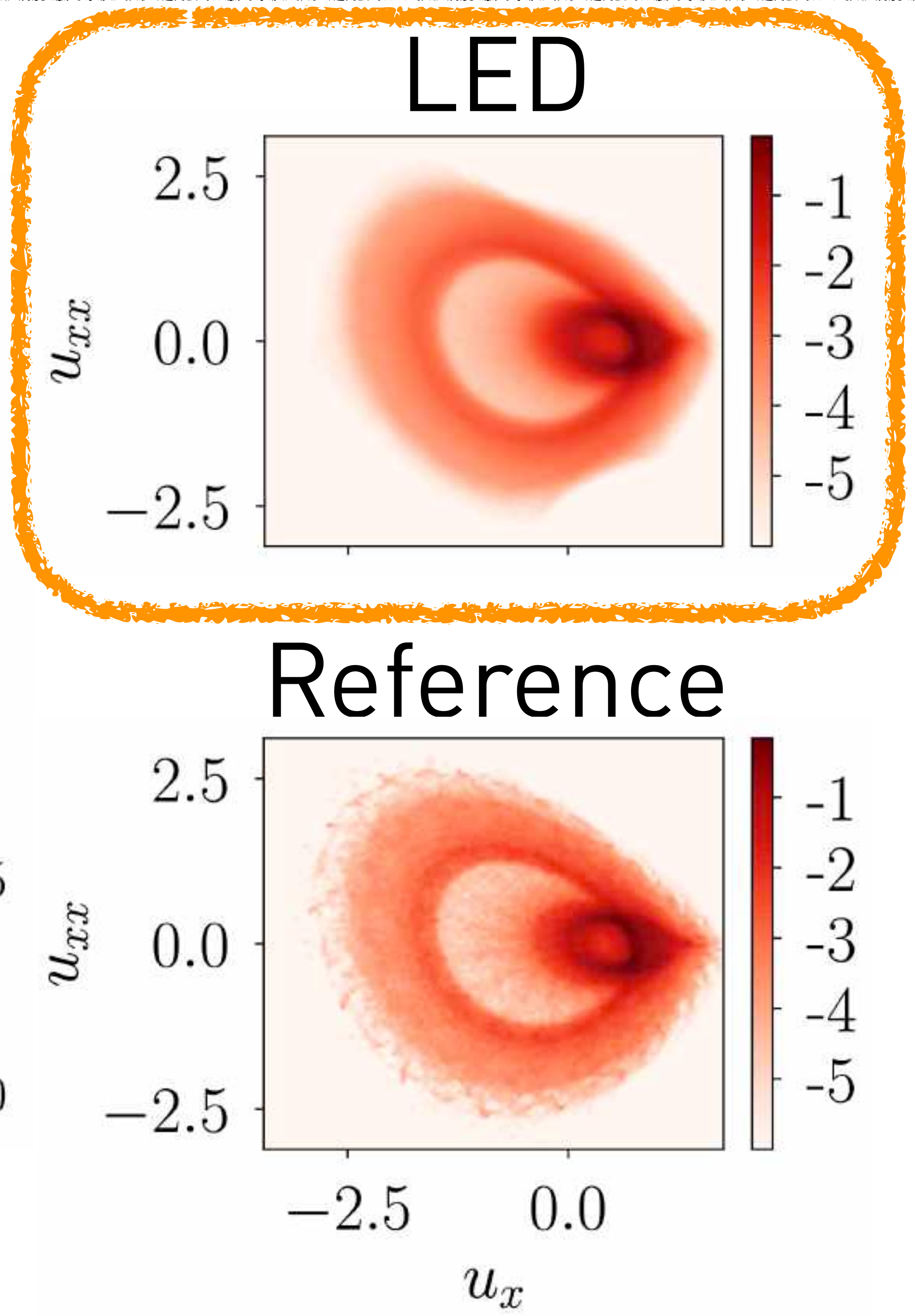
- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult
- LED can identify and **reconstruct the dynamics** on an **8 dimensional** manifold
- Reproducing the statistics/long-term climate accurately
- **Two orders** of magnitude faster compared to the stiff ODE solver used
- LED: Control the tradeoff between **approximation error** and speed-up

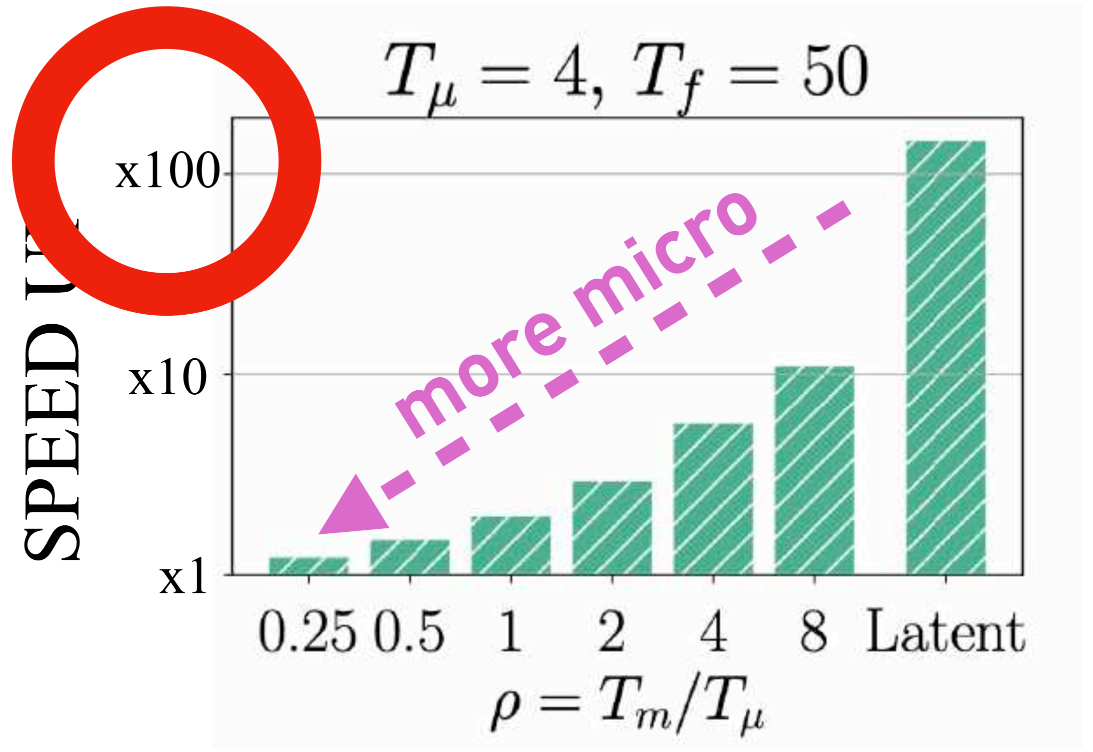
[1] **Robinson, J. C.** "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, 190–193 (1994).
[2] **Linot, A. J. & Graham, M. D.** "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101, 062209 (2020).*

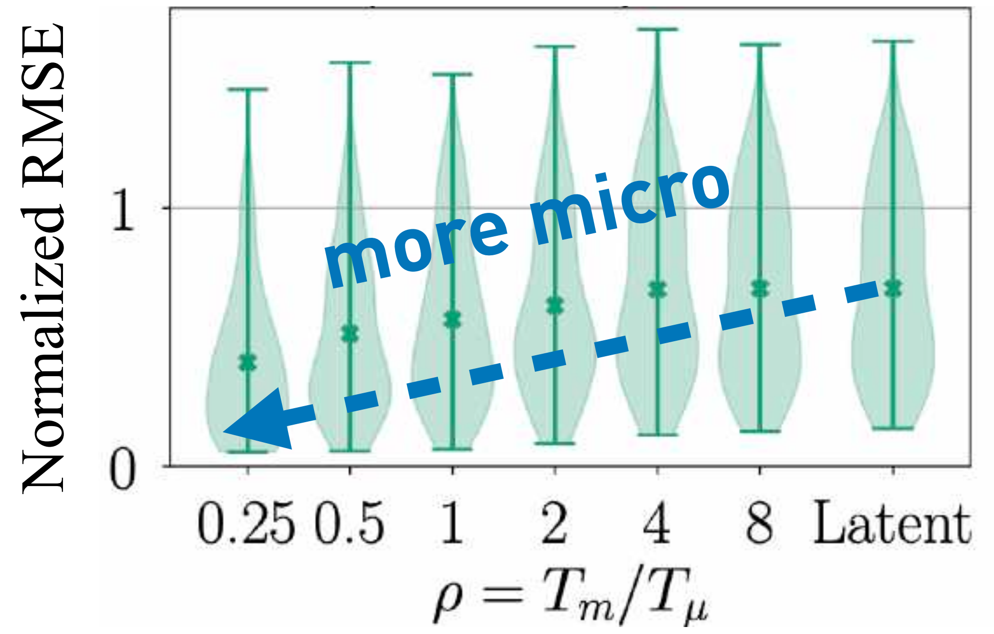
Kuramoto-Sivashinsky ($\tilde{L} \approx 3.5$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

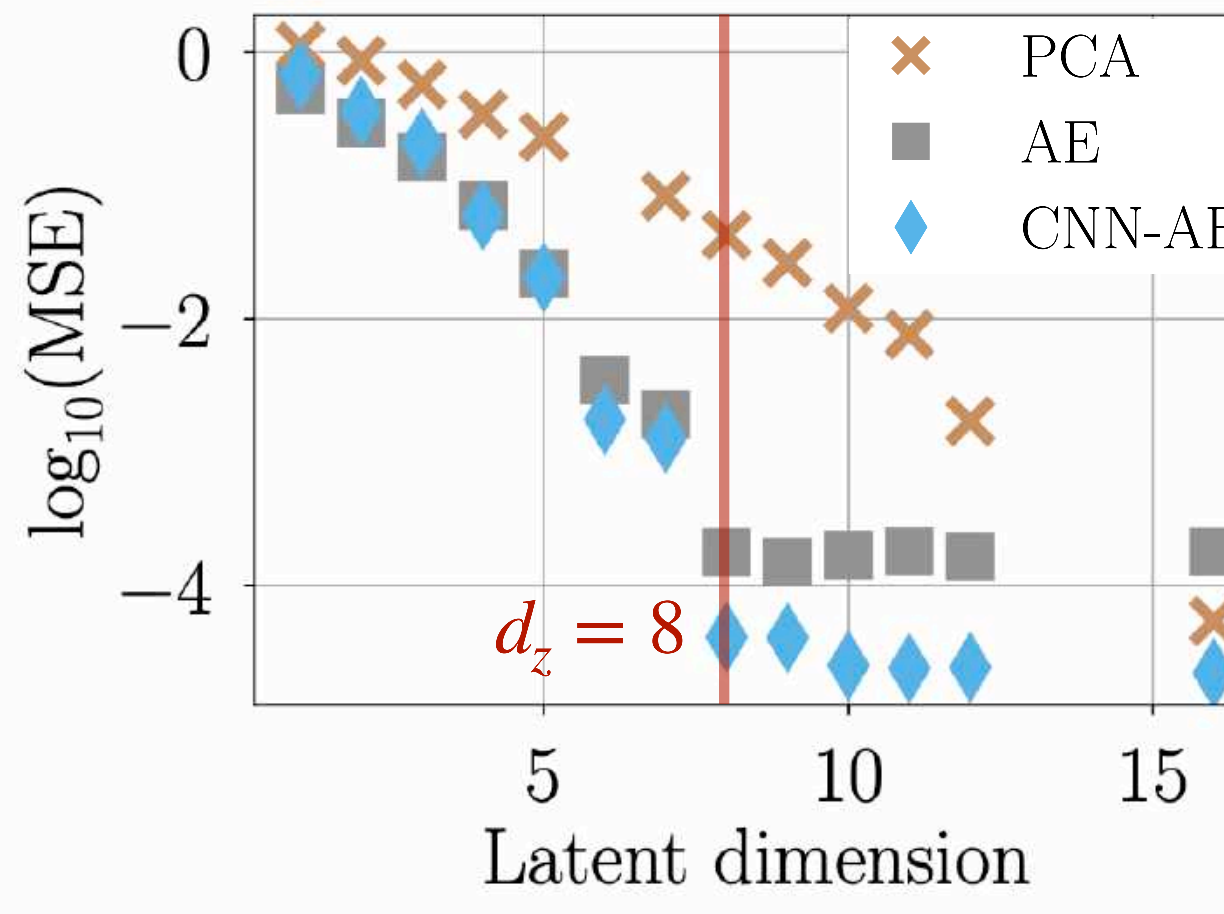








The AE error saturates after $d_z = 8$

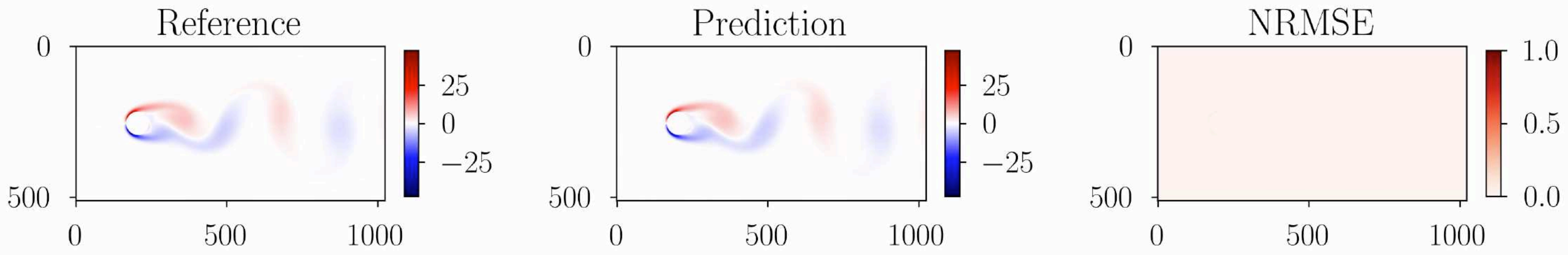


- For $L = 22$, $\nu = 1$, and periodic boundaries ***effective dynamics lie on an 8 dimensional manifold [1, 2]*** but learning a **propagator** of these dynamics is difficult
- LED can identify and **reconstruct the dynamics** on an **8 dimensional** manifold
- Reproducing the statistics/long-term climate accurately
- **Two orders** of magnitude faster compared to the stiff ODE solver used
- LED: Control the tradeoff between **approximation error** and speed-up

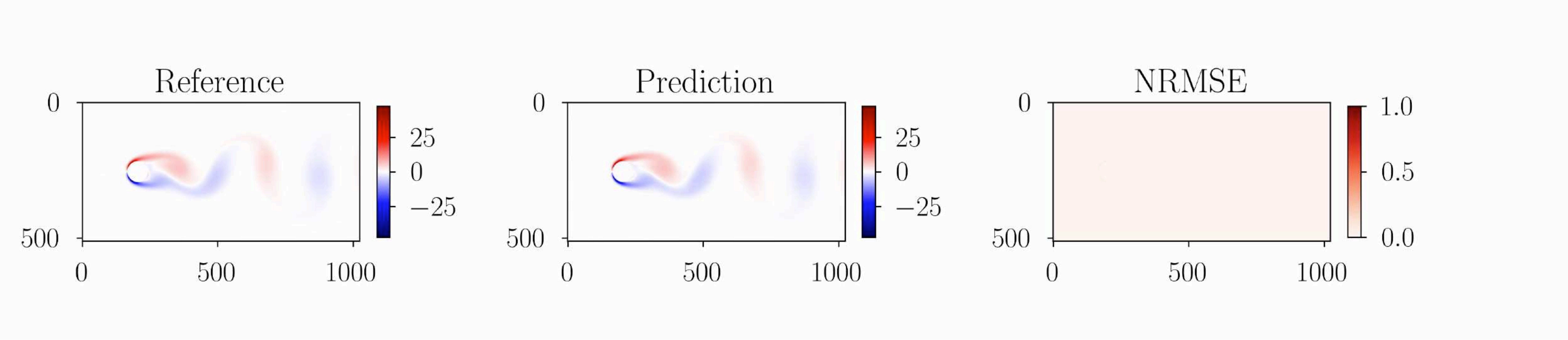
[1] **Robinson, J. C.** "Inertial manifolds for the kuramoto-sivashinsky equation." *Physics Letters A* **184**, *190–193 (1994).*
[2] **Linot, A. J. & Graham, M. D.** "Deep learning to discover and predict dynamics on an inertial manifold." *Physical Review E 101, 062209 (2020).*

Cylinder at $Re = 100$ - (LED $d_z = 4$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)



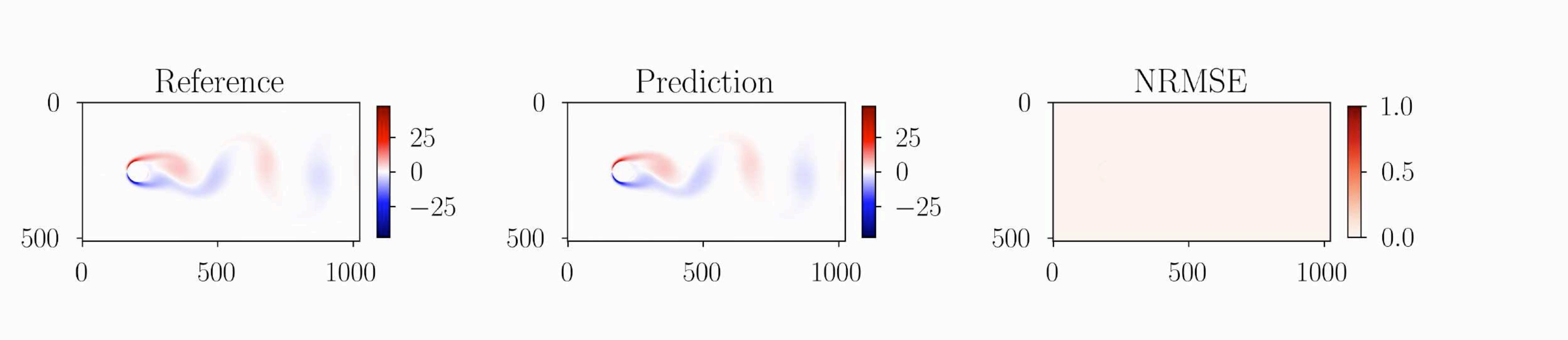
Cylinder at $Re = 100$ - (LED $d_z = 4$)



- Micro solver: Finite Differences (CubimUP2D) employing 12 cores

Cylinder at $Re = 100$ - (LED $d_z = 4$)

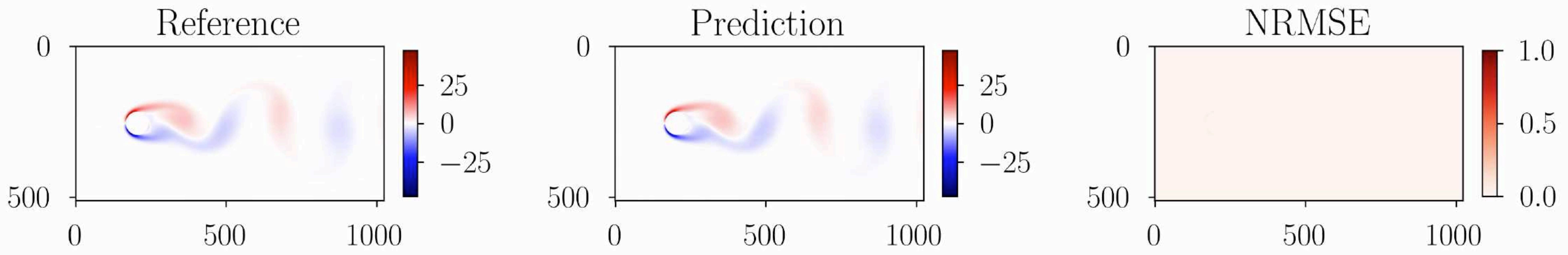
PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)



- Micro solver: Finite Differences (CubimUP2D) employing 12 cores
- State: velocity in x- and y- direction, pressure, and ***vorticity*** $\mathbf{s}_t \in \mathbb{R}^{4\times512\times1024}$

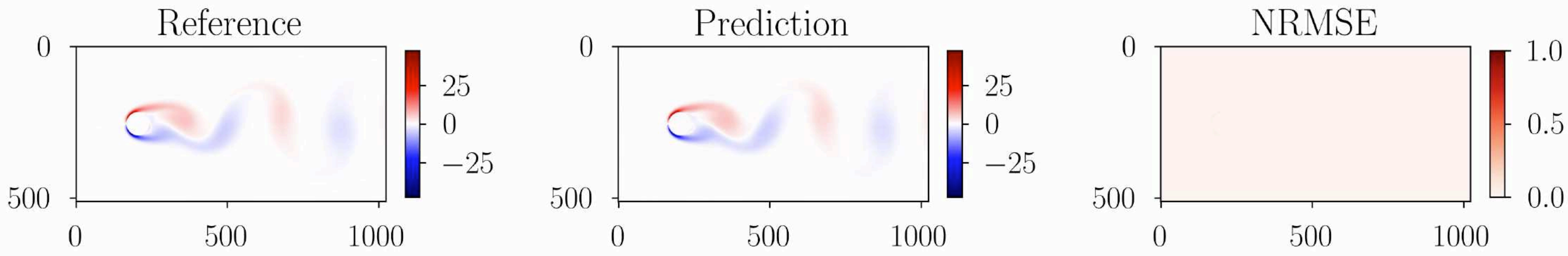
Cylinder at $Re = 100$ - (LED $d_z = 4$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)



- Micro solver: Finite Differences (CubimUP2D) employing 12 cores
- State: velocity in x- and y- direction, pressure, and ***vorticity*** $\mathbf{s}_t \in \mathbb{R}^{4\times512\times1024}$
- LED with latent dimension of $d_{\mathbf{z}} = 4$, $\Delta t = 0.2$

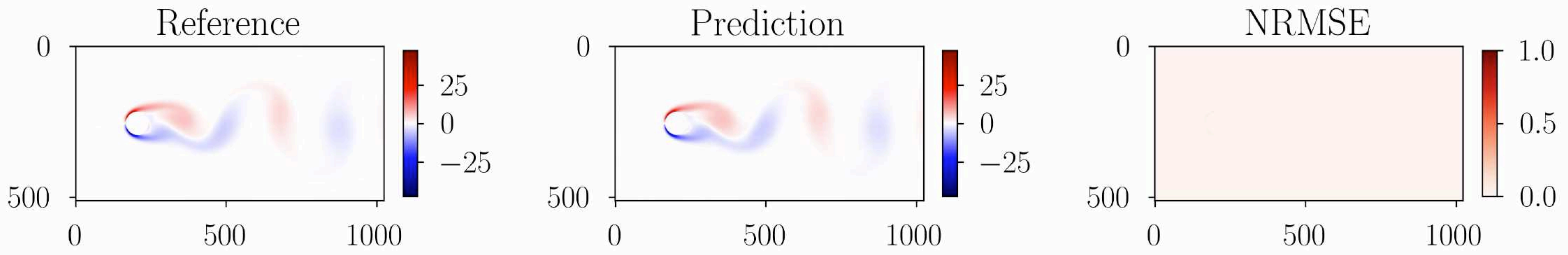
Cylinder at $Re = 100$ - (LED $d_z = 4$)

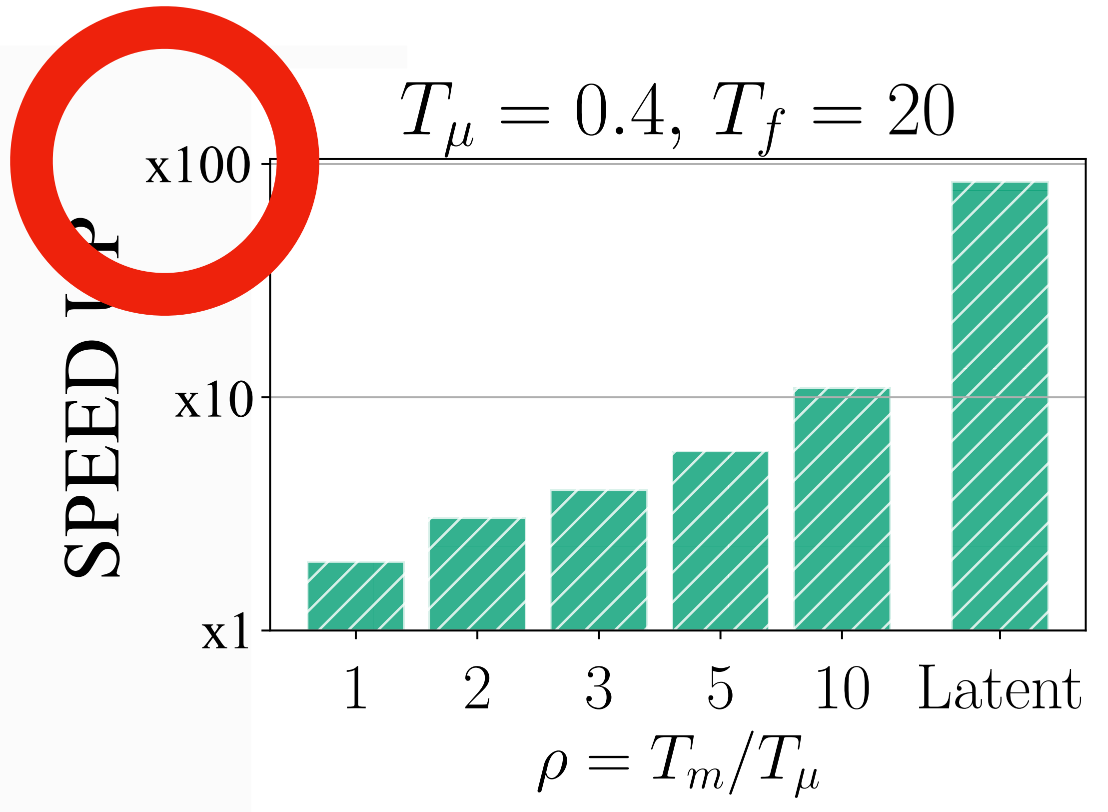


- Micro solver: Finite Differences (CubimUP2D) employing 12 cores
- State: velocity in x- and y- direction, pressure, and *vorticity* $\mathbf{s}_t \in \mathbb{R}^{4\times512\times1024}$
- LED with latent dimension of $d_\mathbf{z} = 4$, $\Delta t = 0.2$
- LED captures **long-term evolution** of velocity and pressure fields (low NRMSE)

Cylinder at $Re = 100$ - (LED $d_z = 4$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

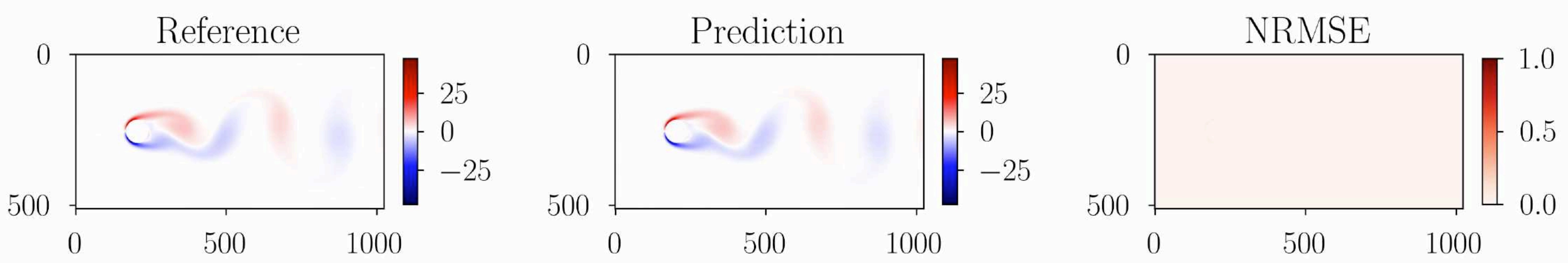


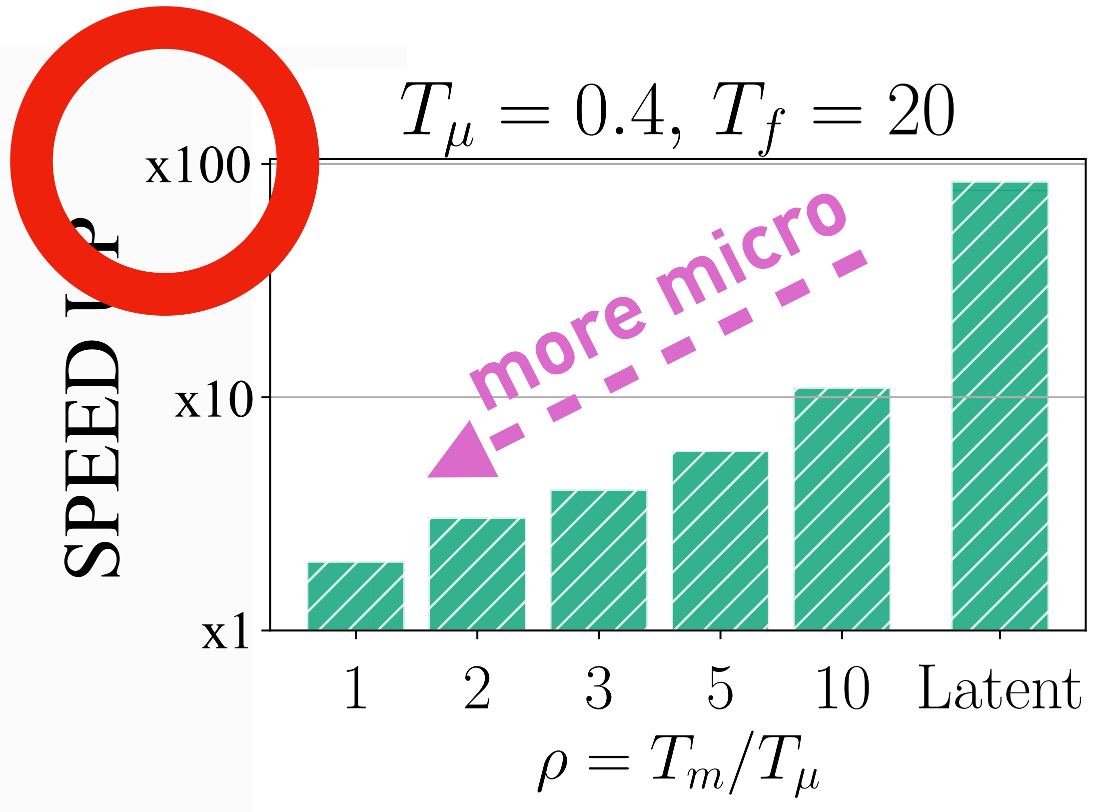


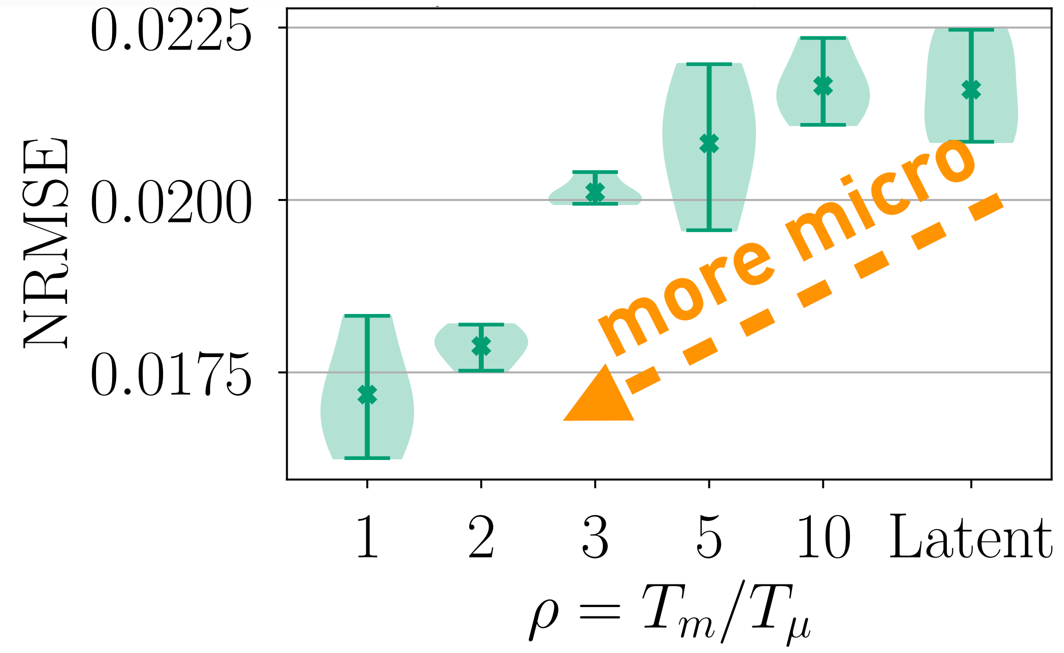
- Micro solver: Finite Differences (CubimUP2D) employing 12 cores
- State: velocity in x- and y- direction, pressure, and ***vorticity*** $\mathbf{s}_t \in \mathbb{R}^{4\times 512\times 1024}$
- LED with latent dimension of $d_{\mathbf{z}} = 4$, $\Delta t = 0.2$
- LED captures **long-term evolution** of velocity and pressure fields (low NRMSE)
- LED is up to **two orders of magnitude** faster than CubismUP2D

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

Cylinder at $Re = 100$ - (LED $d_z = 4$)



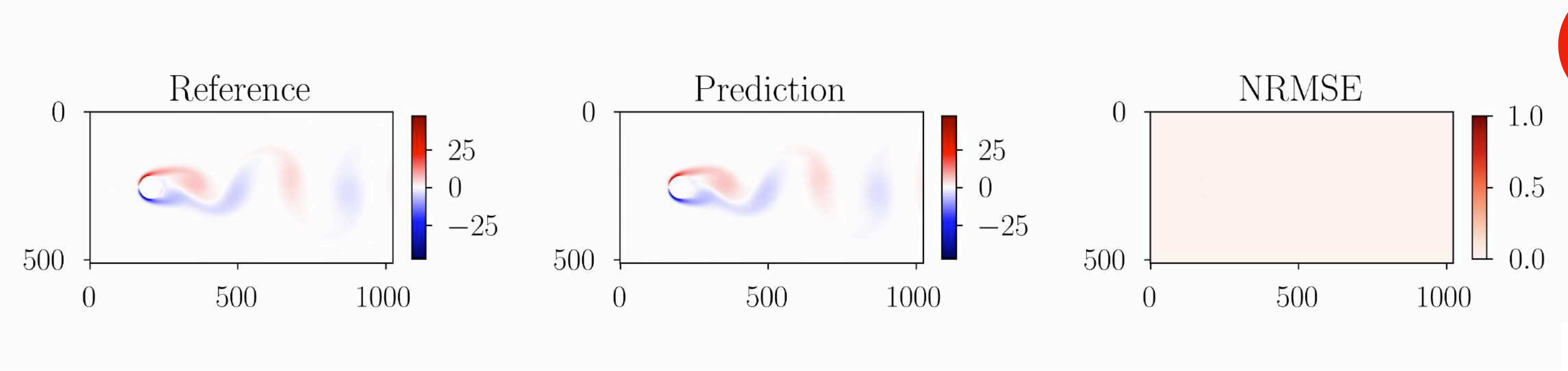


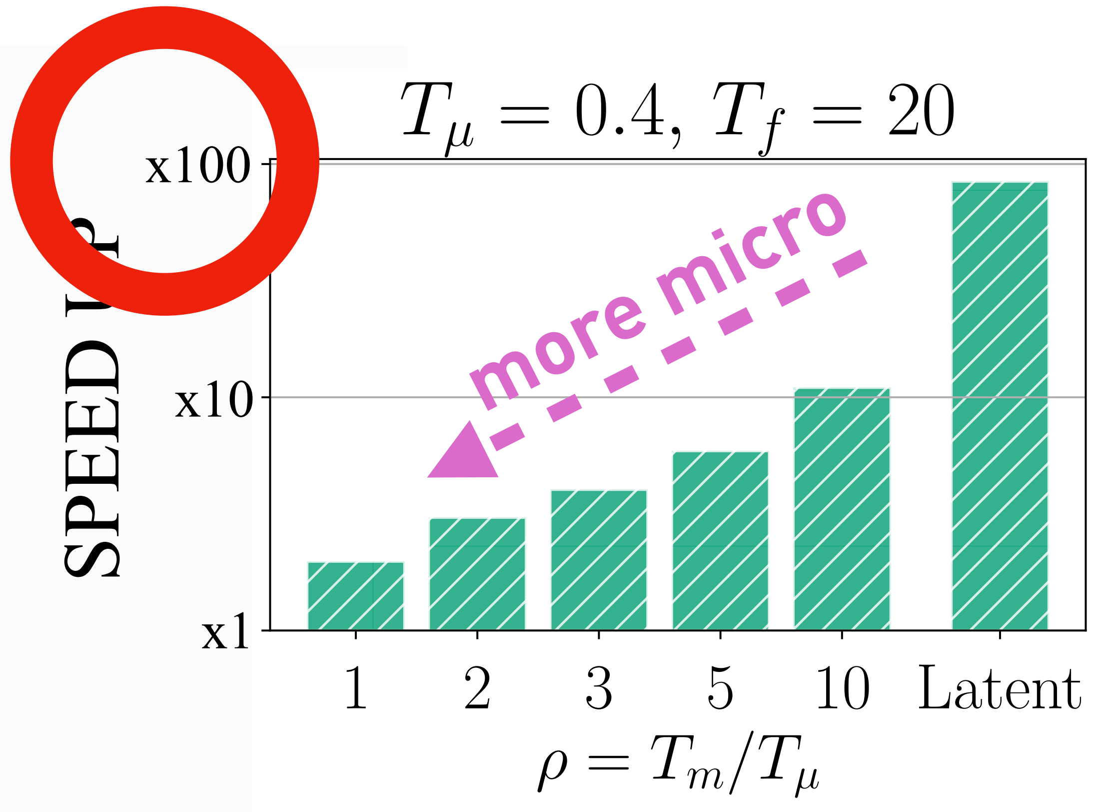


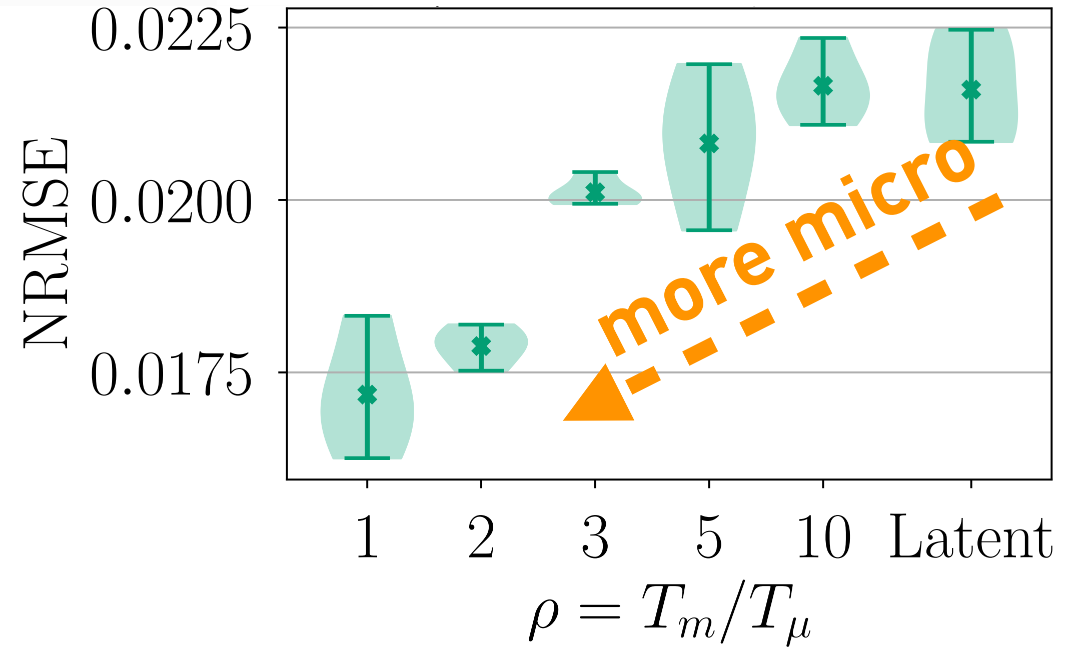
- Micro solver: Finite Differences (CubimUP2D) employing 12 cores
- State: velocity in x- and y- direction, pressure, and ***vorticity*** $\mathbf{s}_t \in \mathbb{R}^{4 \times 512 \times 1024}$
- LED with latent dimension of $d_\mathbf{z} = 4$, $\Delta t = 0.2$
- LED captures **long-term evolution** of velocity and pressure fields (low NRMSE)
- LED is up to **two orders of magnitude** faster than CubismUP2D
- Multiscale LED reduces the approximation error at the cost of reduced speed-up

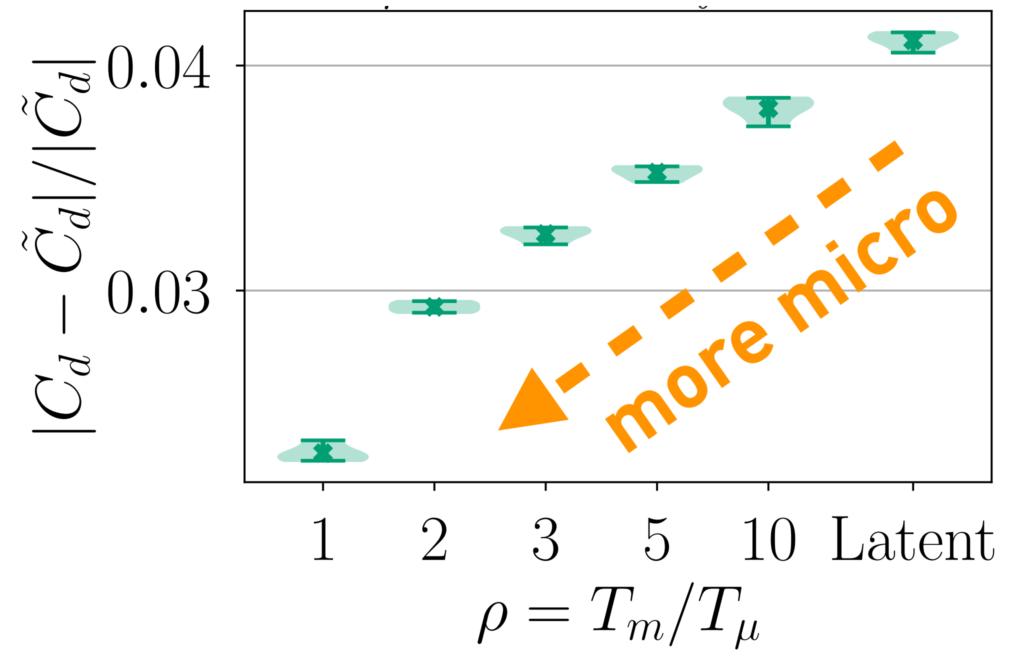
Cylinder at $Re = 100$ - (LED $d_z = 4$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)





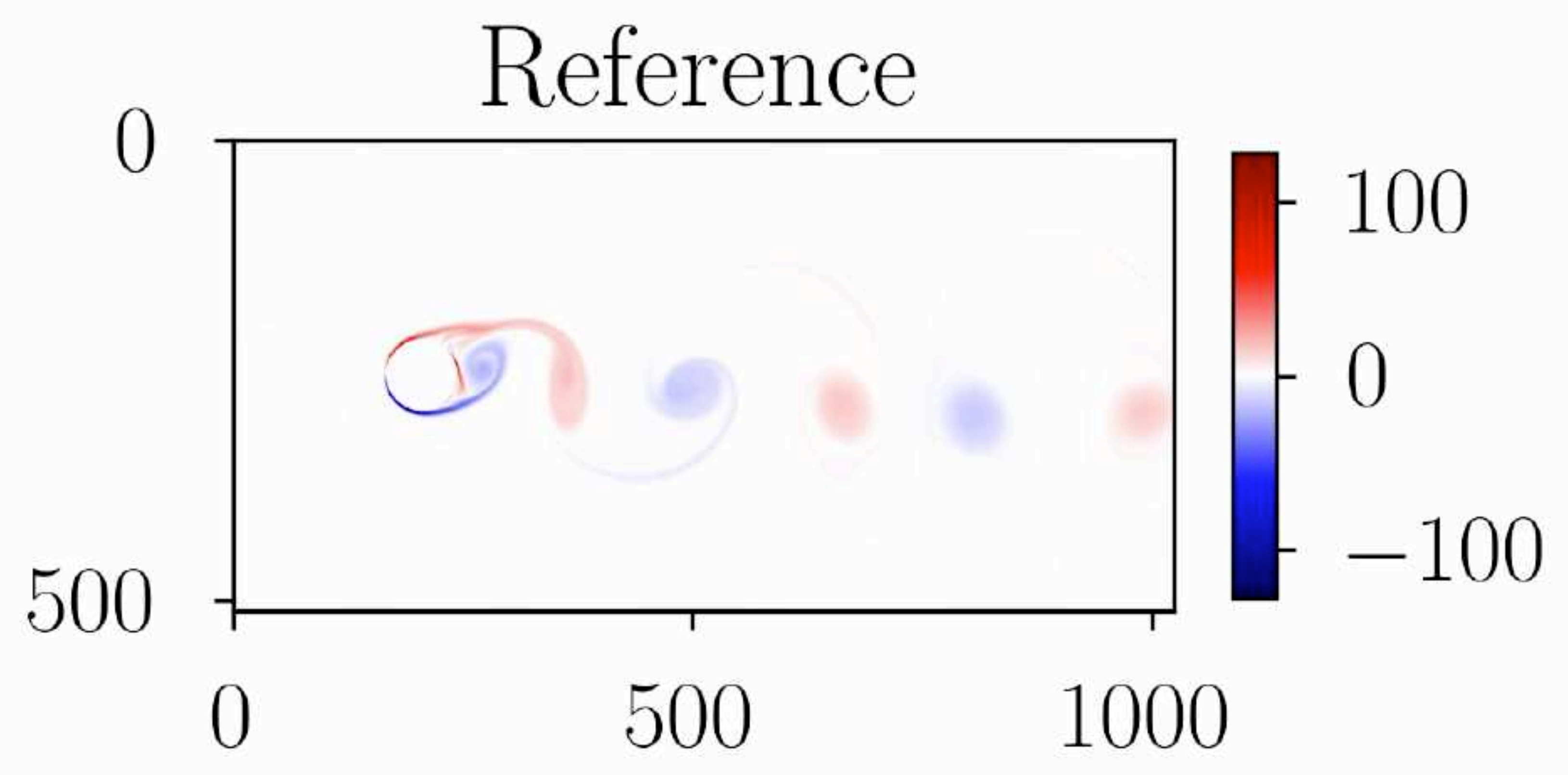


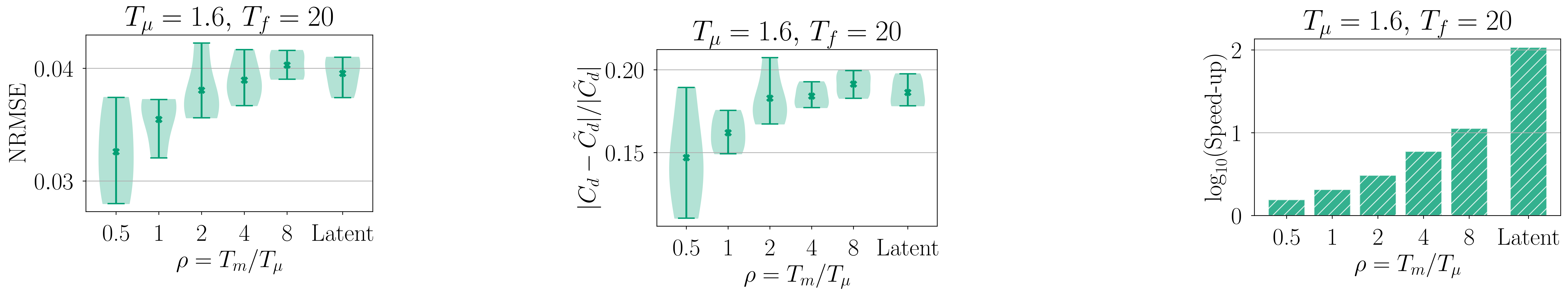


- Micro solver: Finite Differences (CubimUP2D) employing 12 cores
- State: velocity in x- and y- direction, pressure, and ***vorticity*** $\mathbf{s}_t \in \mathbb{R}^{4\times512\times1024}$
- LED with latent dimension of $d_{\mathbf{z}} = 4$, $\Delta t = 0.2$
- LED captures **long-term evolution** of velocity and pressure fields (low NRMSE)
- LED is up to **two orders of magnitude** faster than CubismUP2D
- Multiscale LED reduces the approximation error at the cost of reduced speed-up
- Recovers drag coefficient with $\approx 2 - 4\,\%$ error

Cylinder at $Re = 1000$ (LED $d_z = 10$)

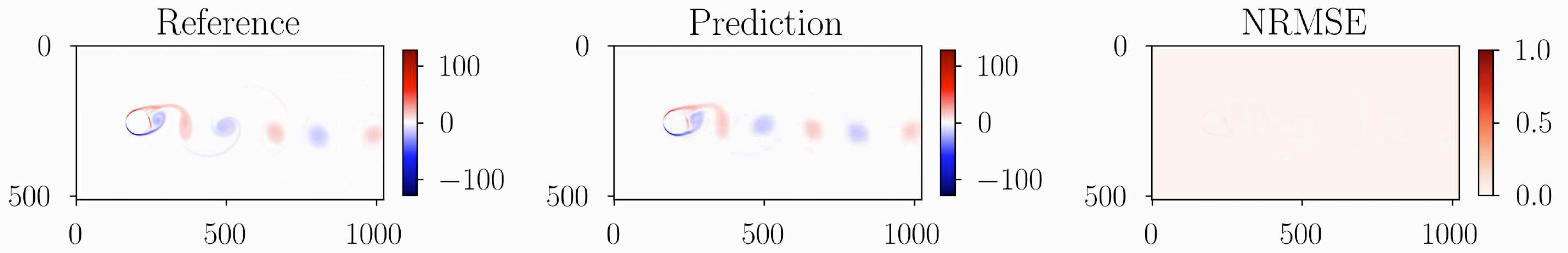
PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

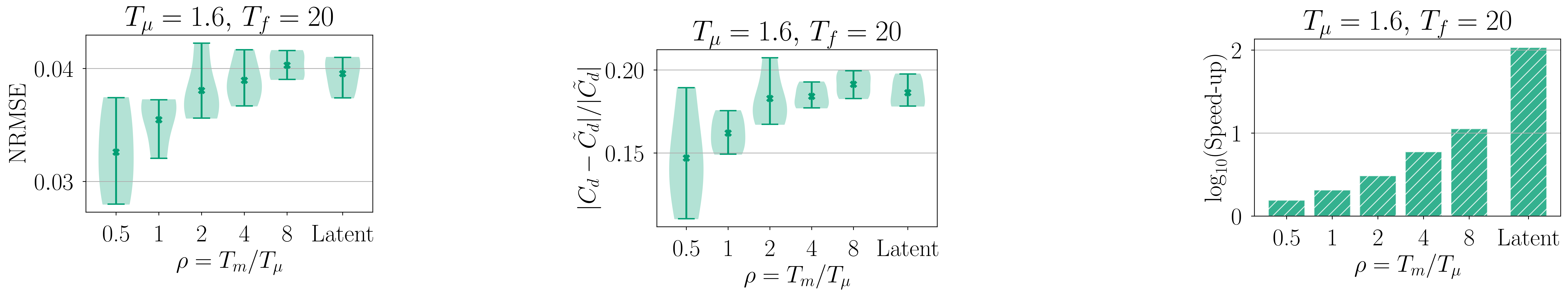




Cylinder at $Re = 1000$ (LED $d_z = 10$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

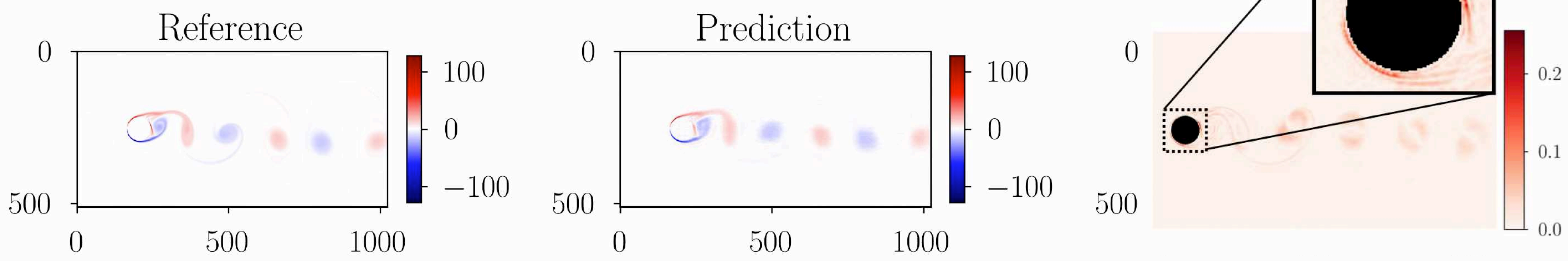


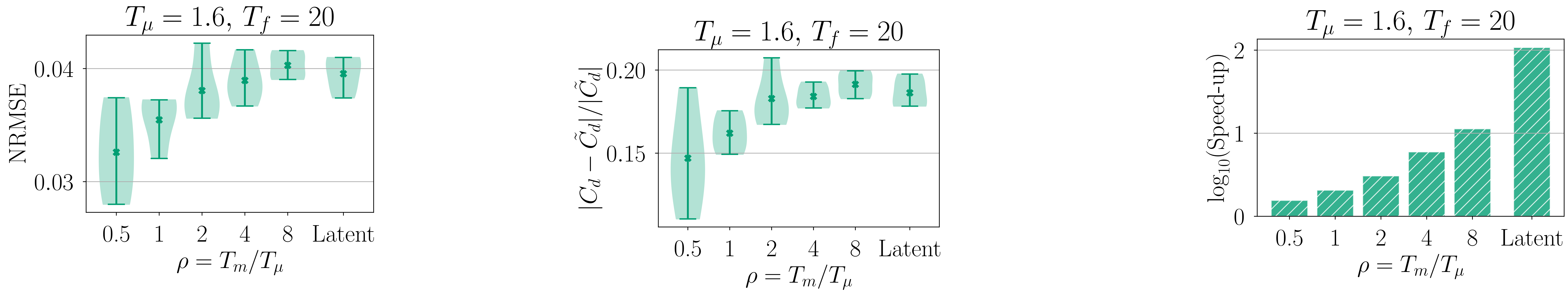


Cylinder at $Re = 1000$ (LED $d_z = 10$)

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

Most error is concentrated around the cylinder !





Comparisons of Latent Propagators

PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos,
Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,
Nature Machine Intelligence, (to appear 2022)

SINDy — **SL Brunton, JL Proctor, JN Kutz,** *Discovering governing equations from data by sparse identification of nonlinear dynamical systems,* **PNAS (2016)**

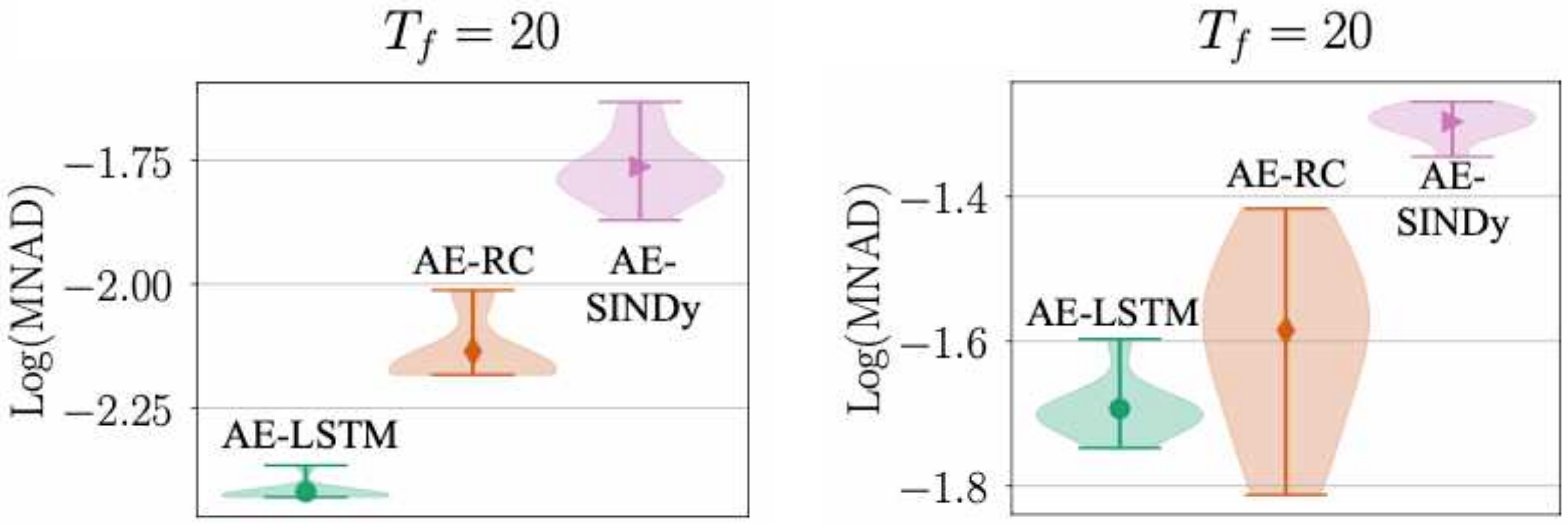
RC — **J Pathak, B Hunt, M Girvan, Z Lu, E Ott,** *Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach,* **Physical review letters, 2018**

LSTM — **S Hochreiter, J Schmidhuber,** *Long short-term memory,* **Neural Computation, 1997**

Mean normalised absolute difference:

$$\mathrm{NAD}(t_j) = \frac{1}{N_x}\sum_{i=1}^{N_x} \frac{|\,y(x_i,t_j) - \hat{y}(x_i,t_j)\,|}{\max_{i,j}(y(x_i,t_j)) - \min_{i,j}(y(x_i,t_j))}$$

$$\mathrm{MNAD} = \frac{1}{N_T}\sum_{j=1}^{N_T} \mathrm{NAD}(t_j)$$



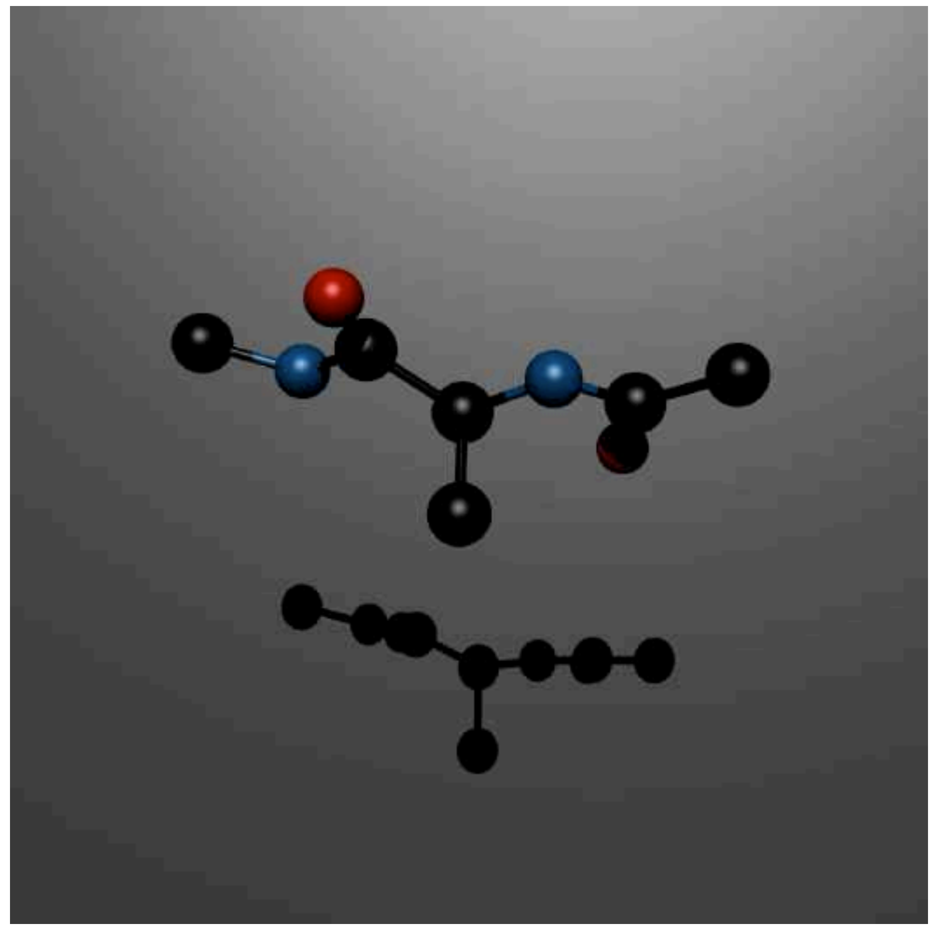
$Re = 100$ $Re = 1000$

III

Learning Effective Dynamics for Molecular Systems

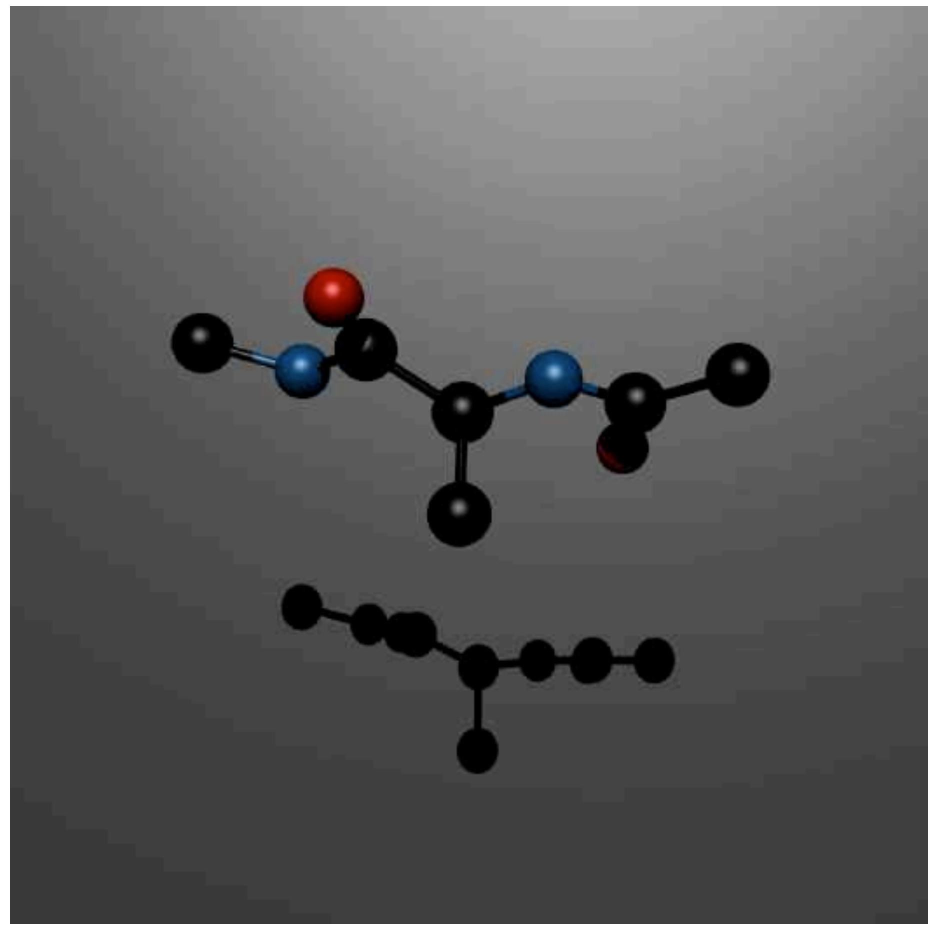
Alanine Dipeptide

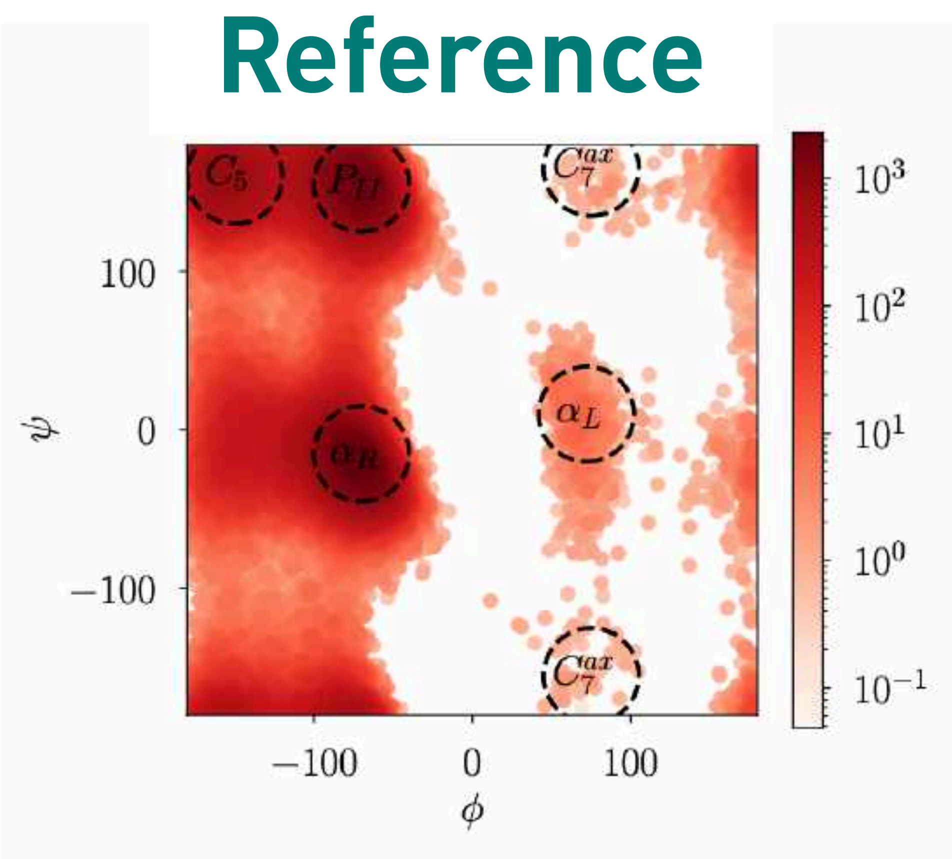
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



Alanine Dipeptide

PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



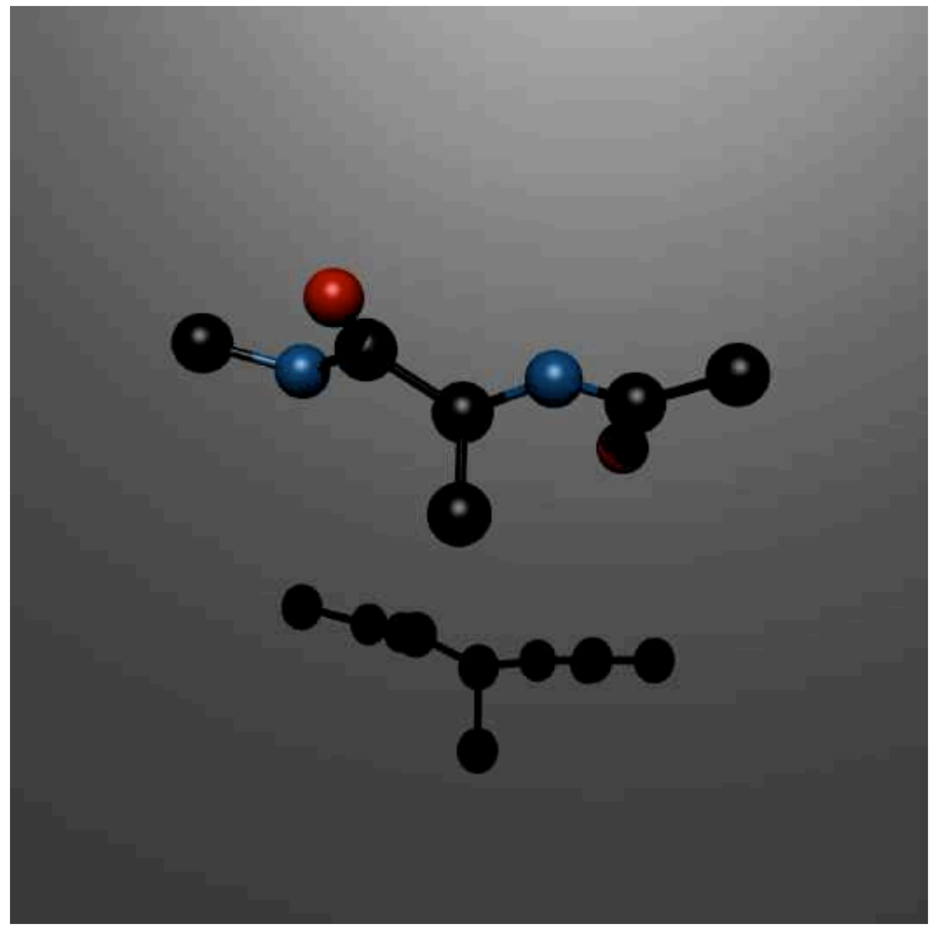


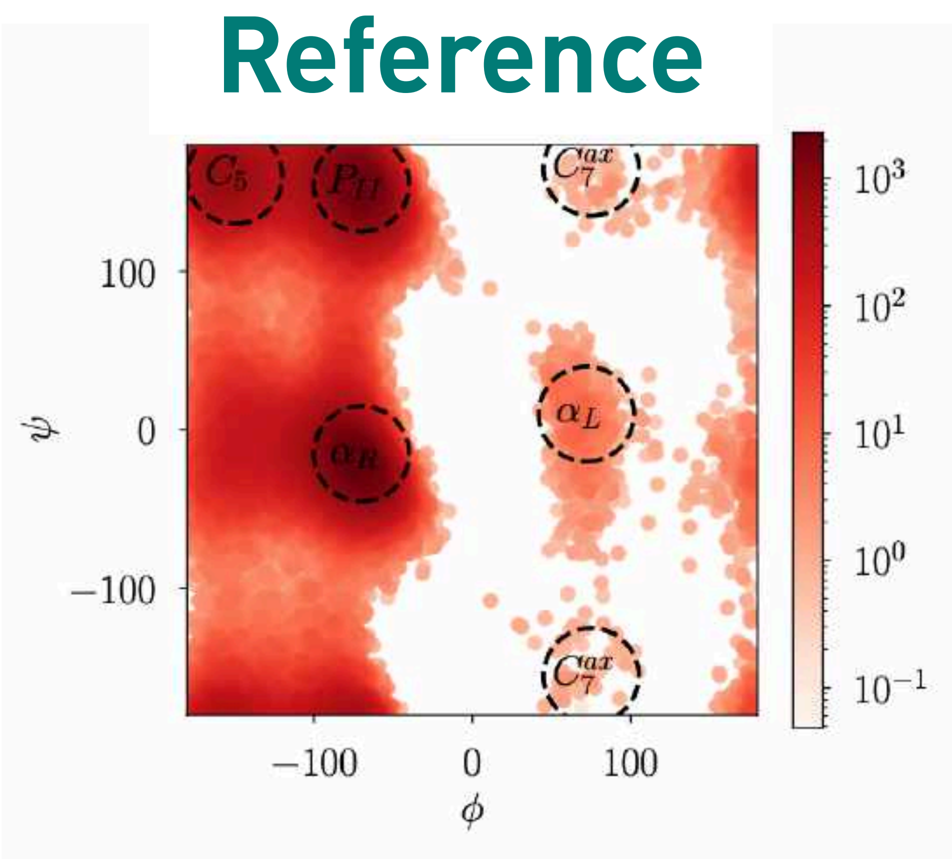
Ramachandran space spanned by dihedral angles (ϕ, ψ)

- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)

Alanine Dipeptide

PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



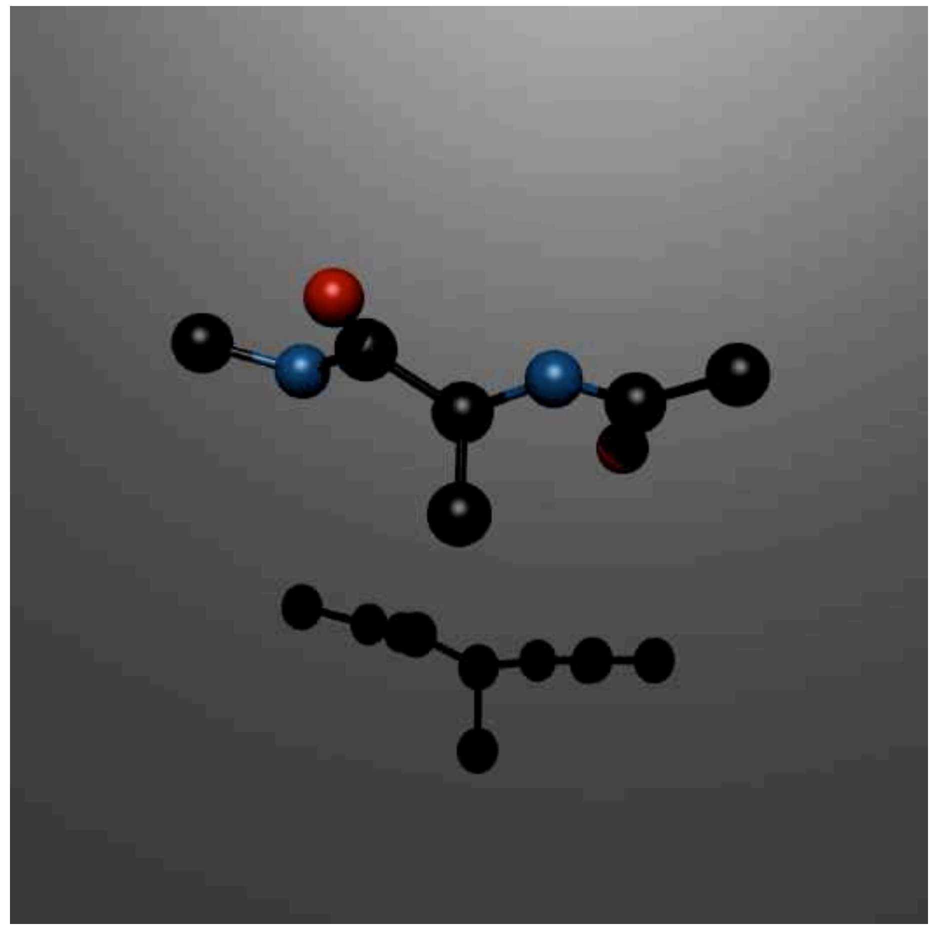


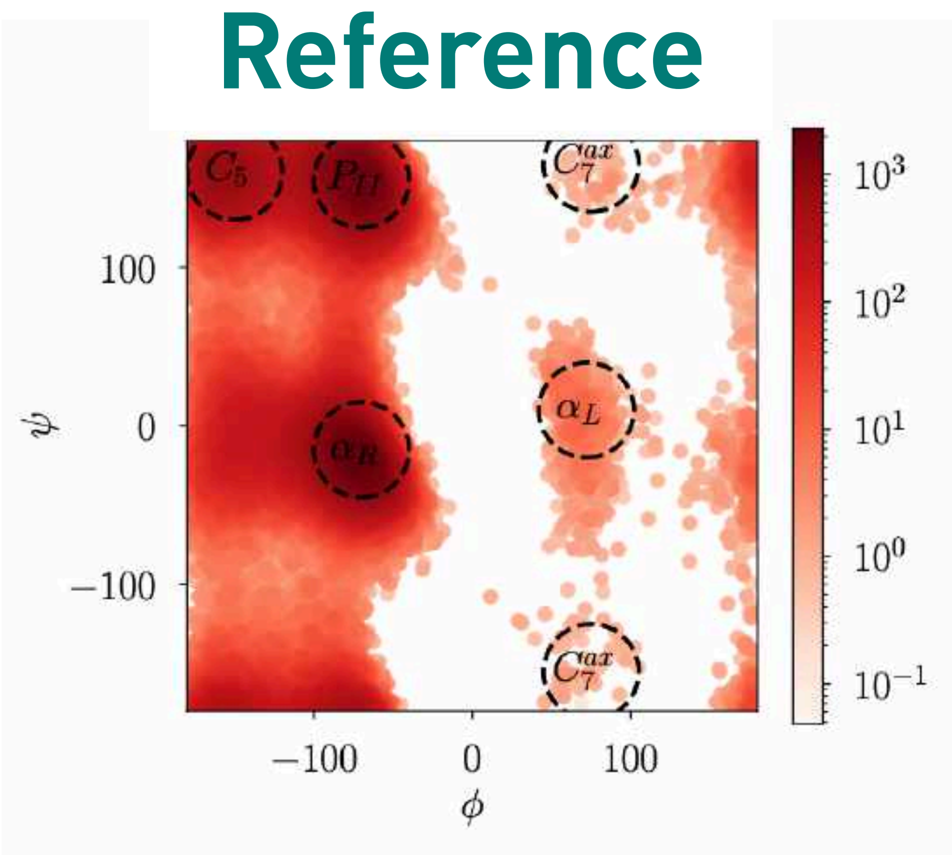
Ramachandran space spanned by dihedral angles (ϕ, ψ)

- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$

PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021

Alanine Dipeptide



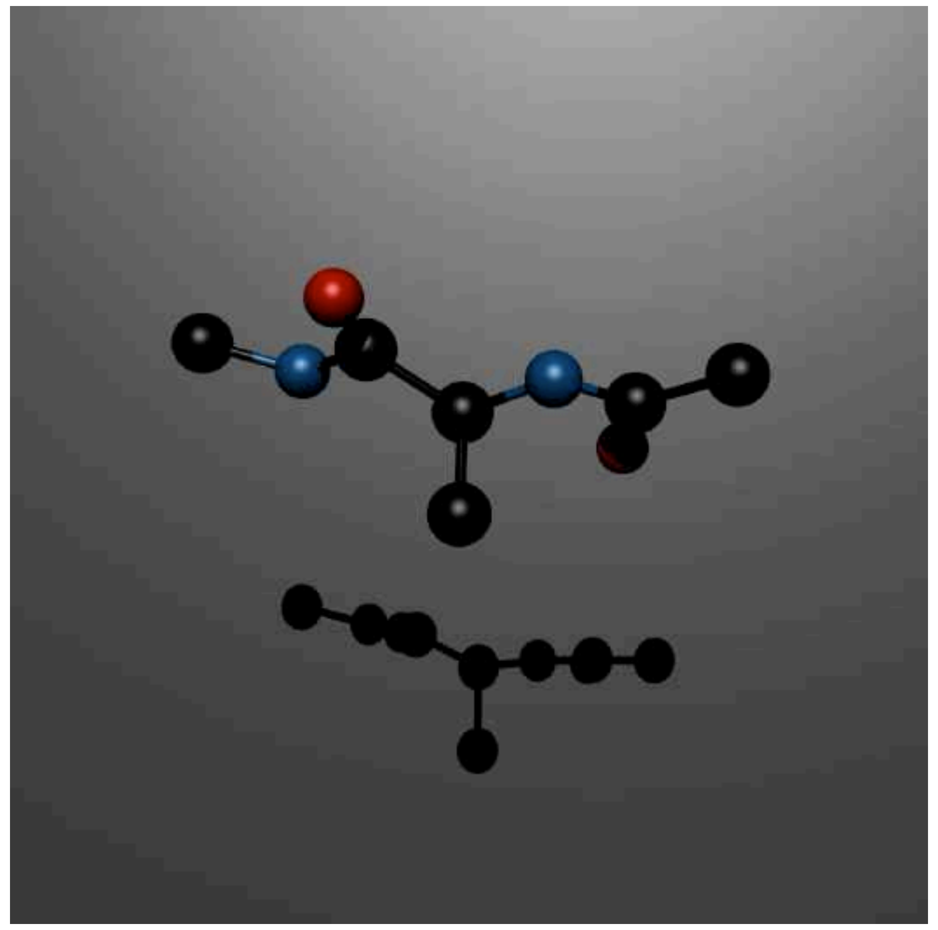


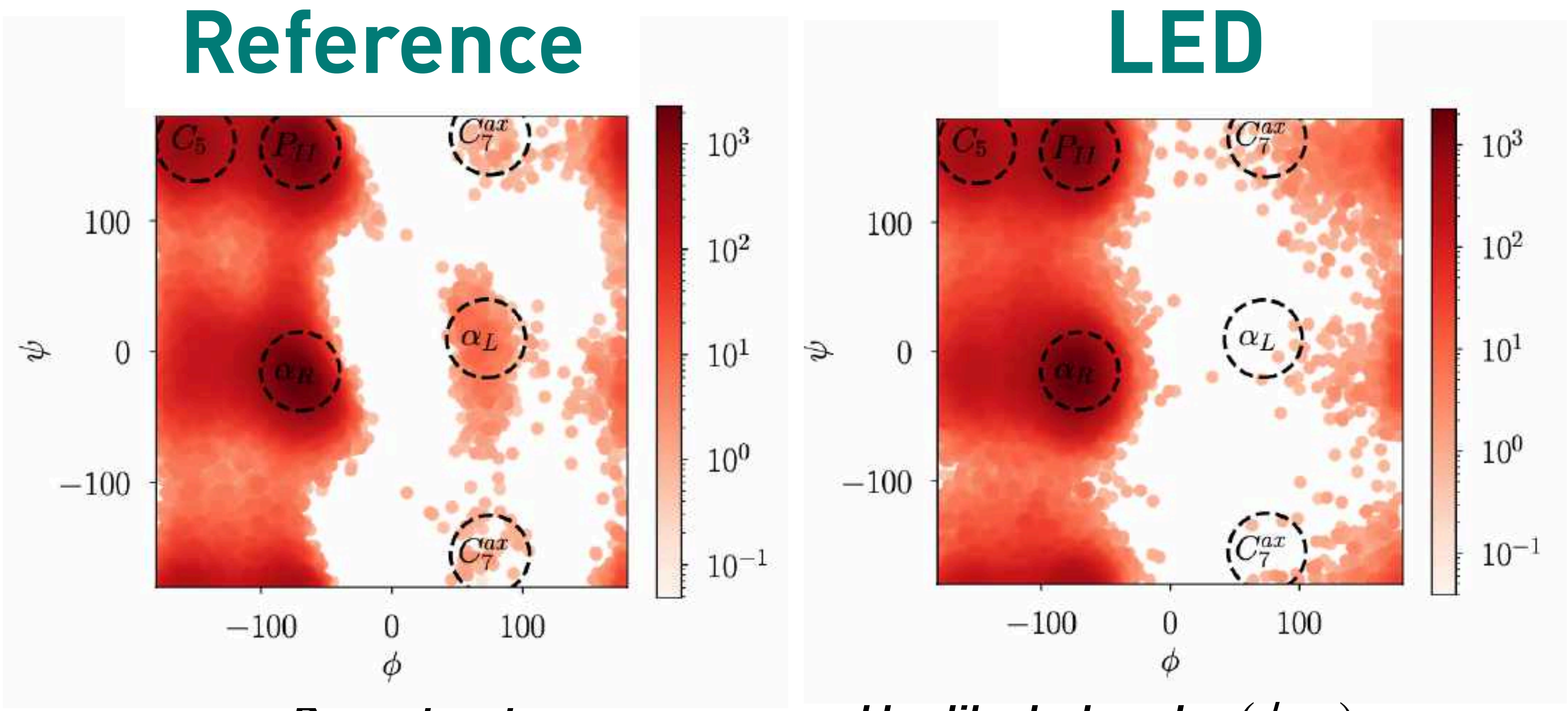
Ramachandran space spanned by dihedral angles (ϕ, ψ)

- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O, N, C**), ignore hydrogens, solvent.

Alanine Dipeptide

PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



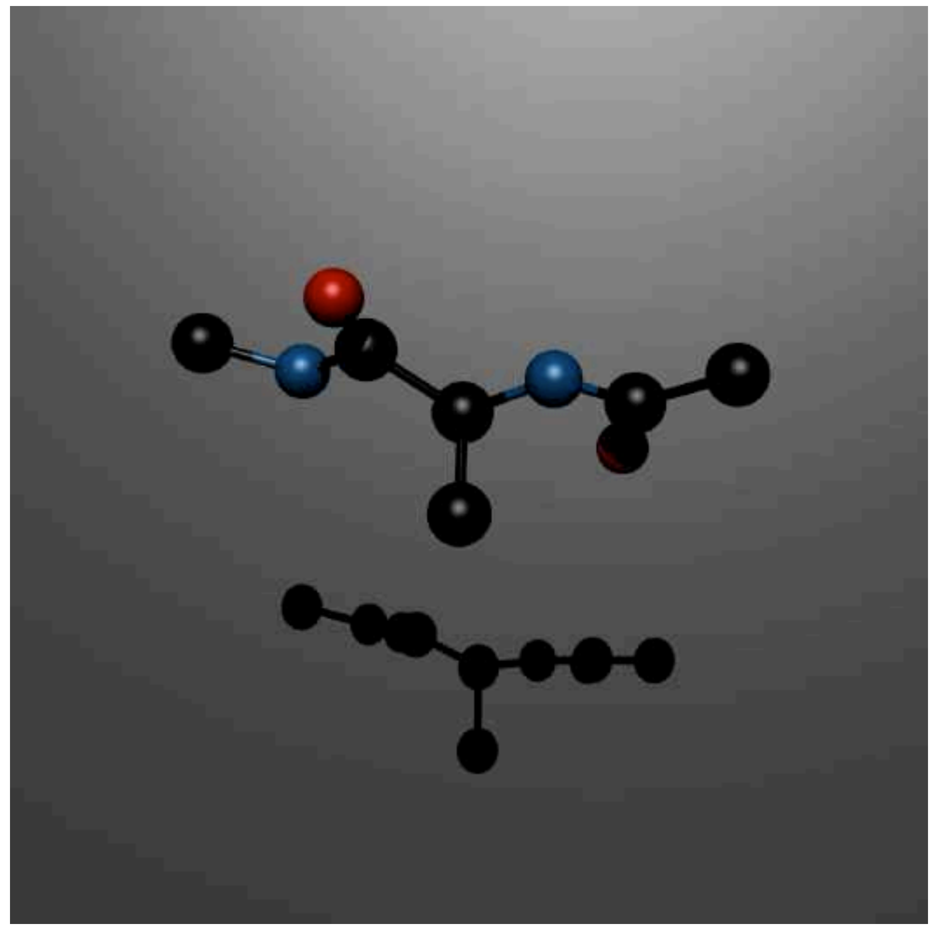


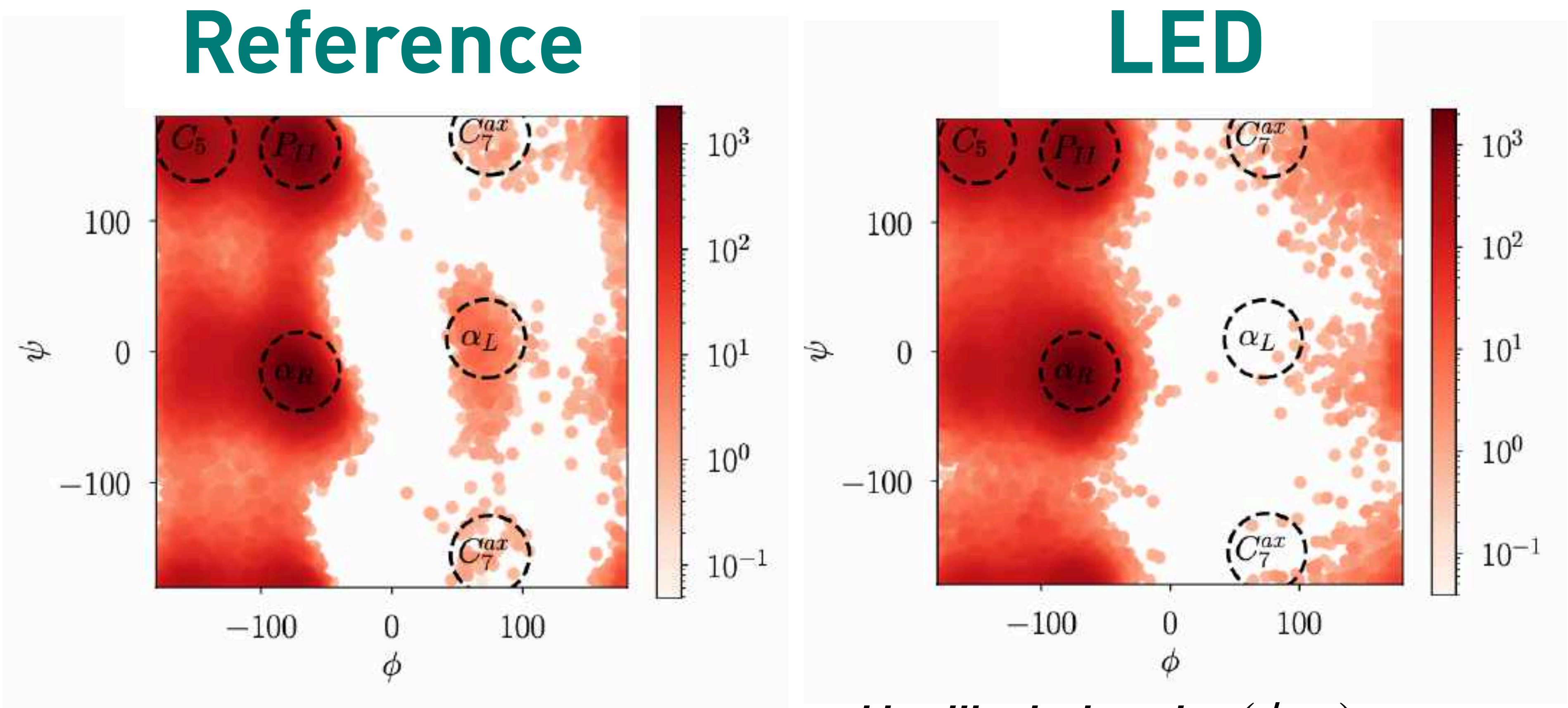
Ramachandran space spanned by dihedral angles (ϕ, ψ)

- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O, N, C**), ignore hydrogens, solvent.
- **LED with $d_z = 1$, *train on 80* ns *of data, randomly sample 500* ns *!***

Alanine Dipeptide

PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



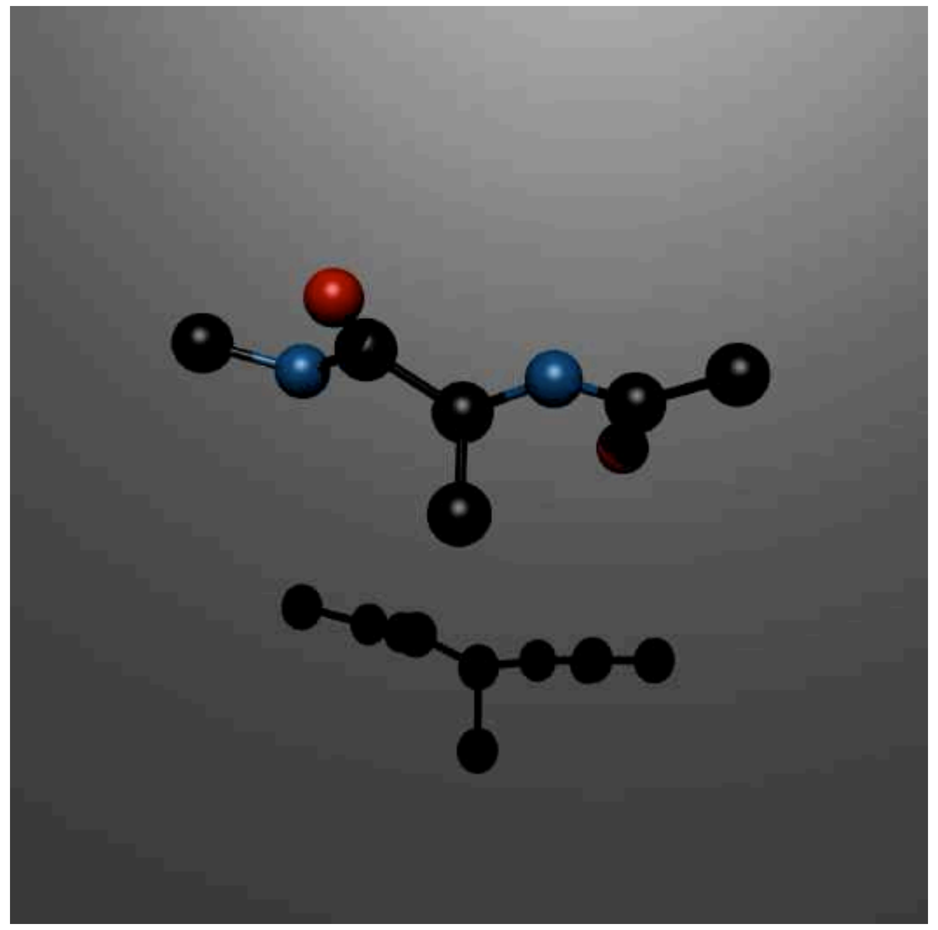


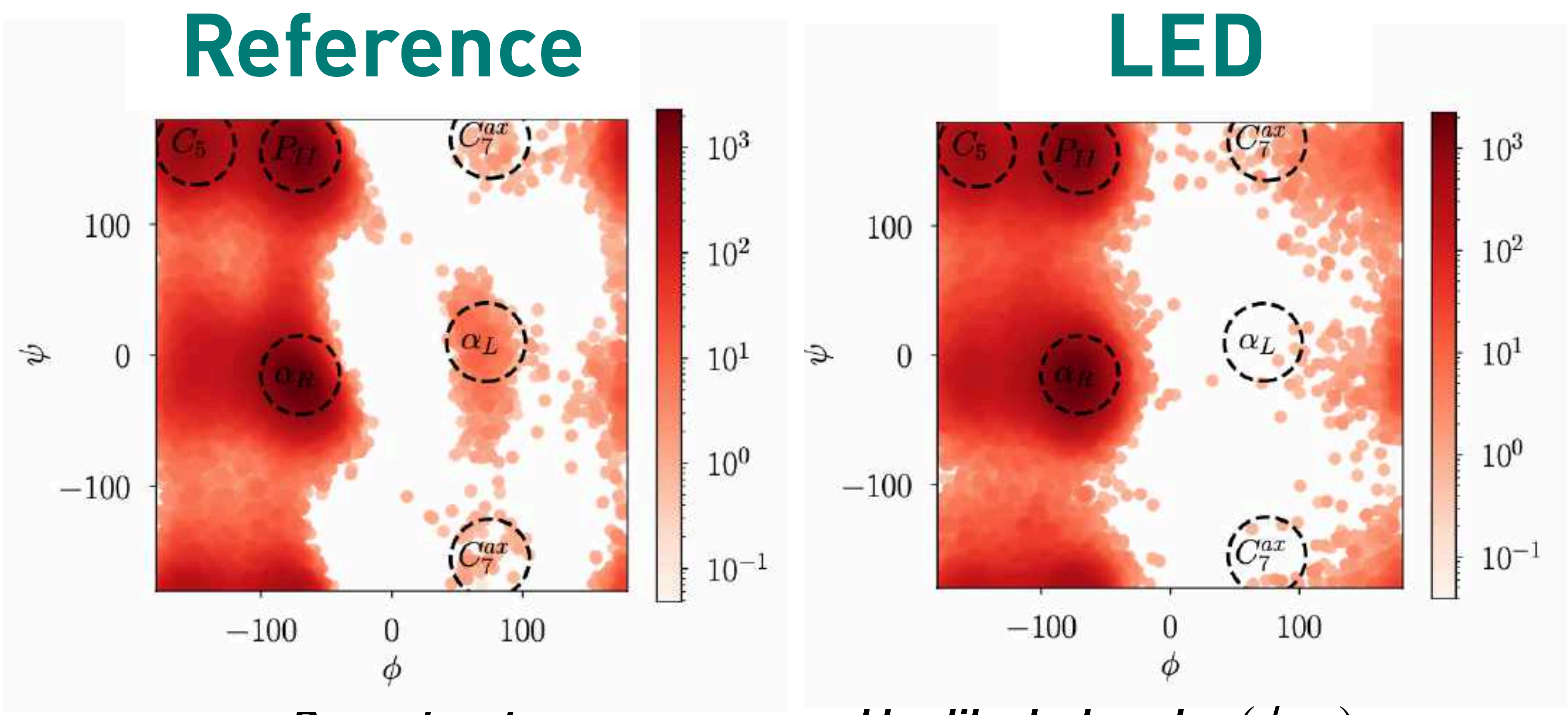
Ramachandran space spanned by dihedral angles (ϕ, ψ)

- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O, N, C**), ignore hydrogens, solvent.
- **LED with** $d_{\mathbf{z}} = 1$, ***train on 80*** **ns** ***of data, randomly sample 500*** **ns** ***!***
  - reproduces coarsely the statistics

Alanine Dipeptide

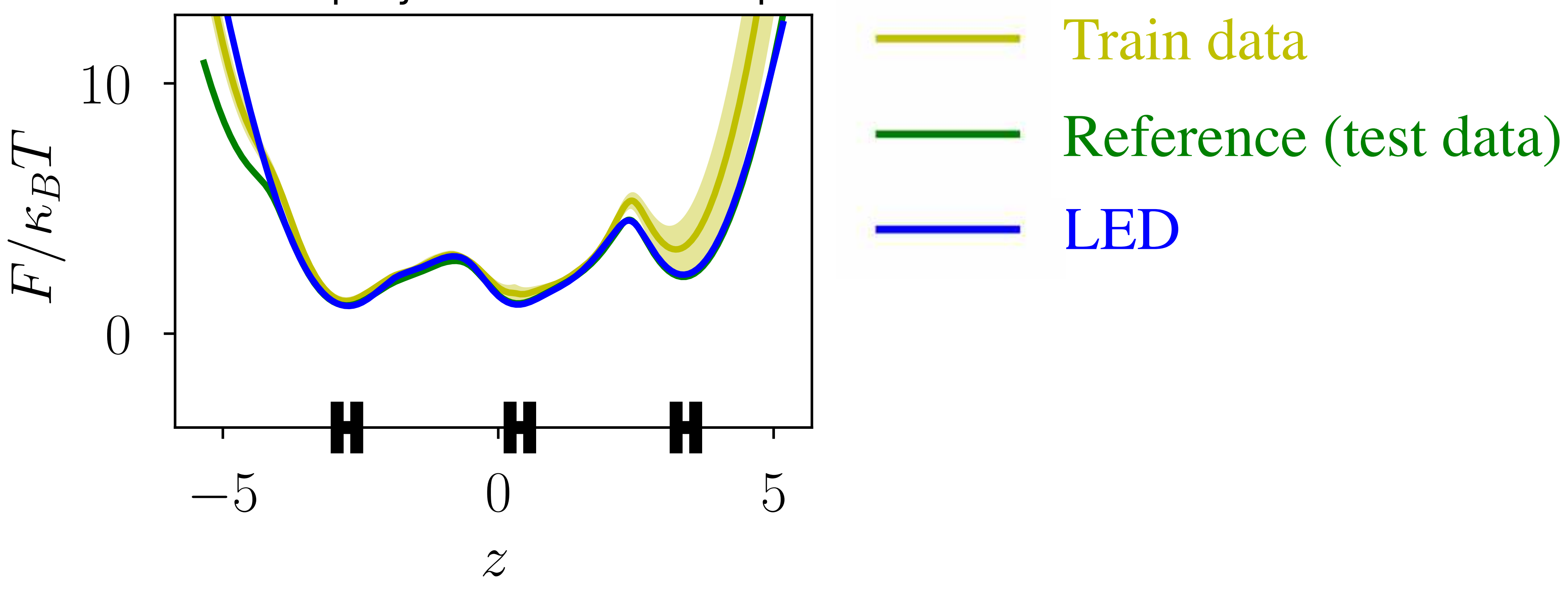
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos, *Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics*, **Journal of Chemical Theory & Computation, 2021**





Ramachandran space spanned by dihedral angles (ϕ, ψ)

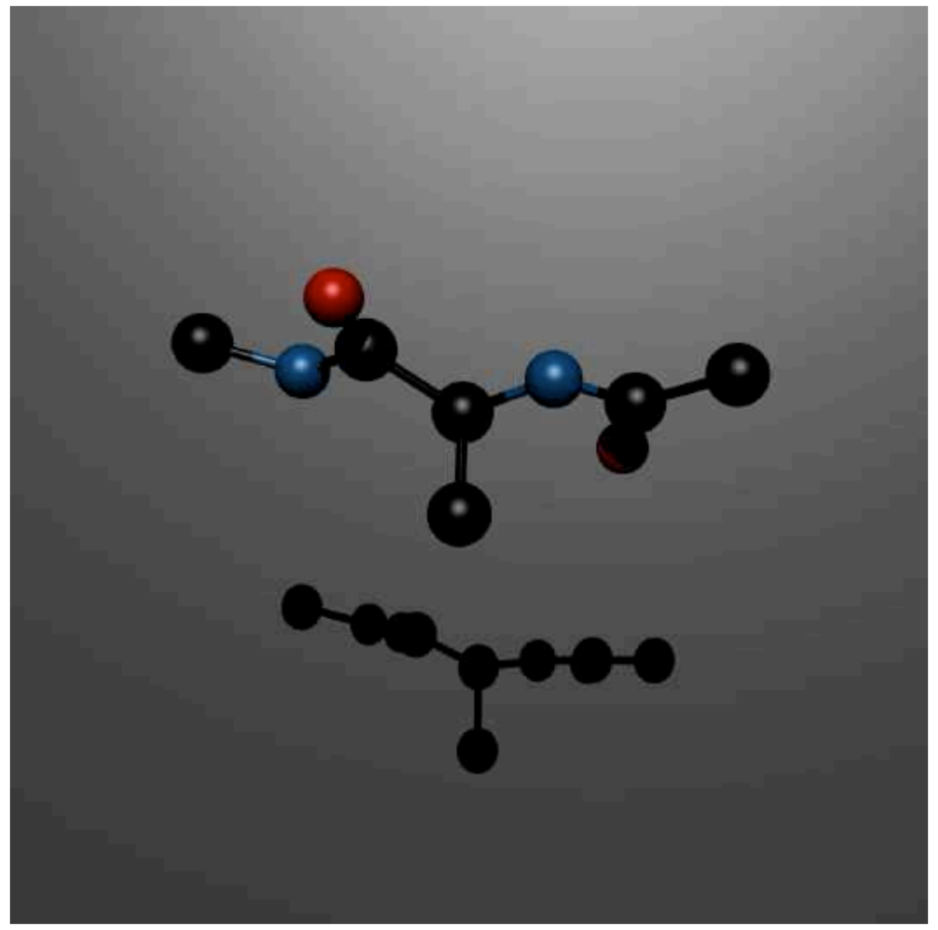
Free energy $F = -\kappa_B T \ln p(z)$: molecular data projected to latent space !

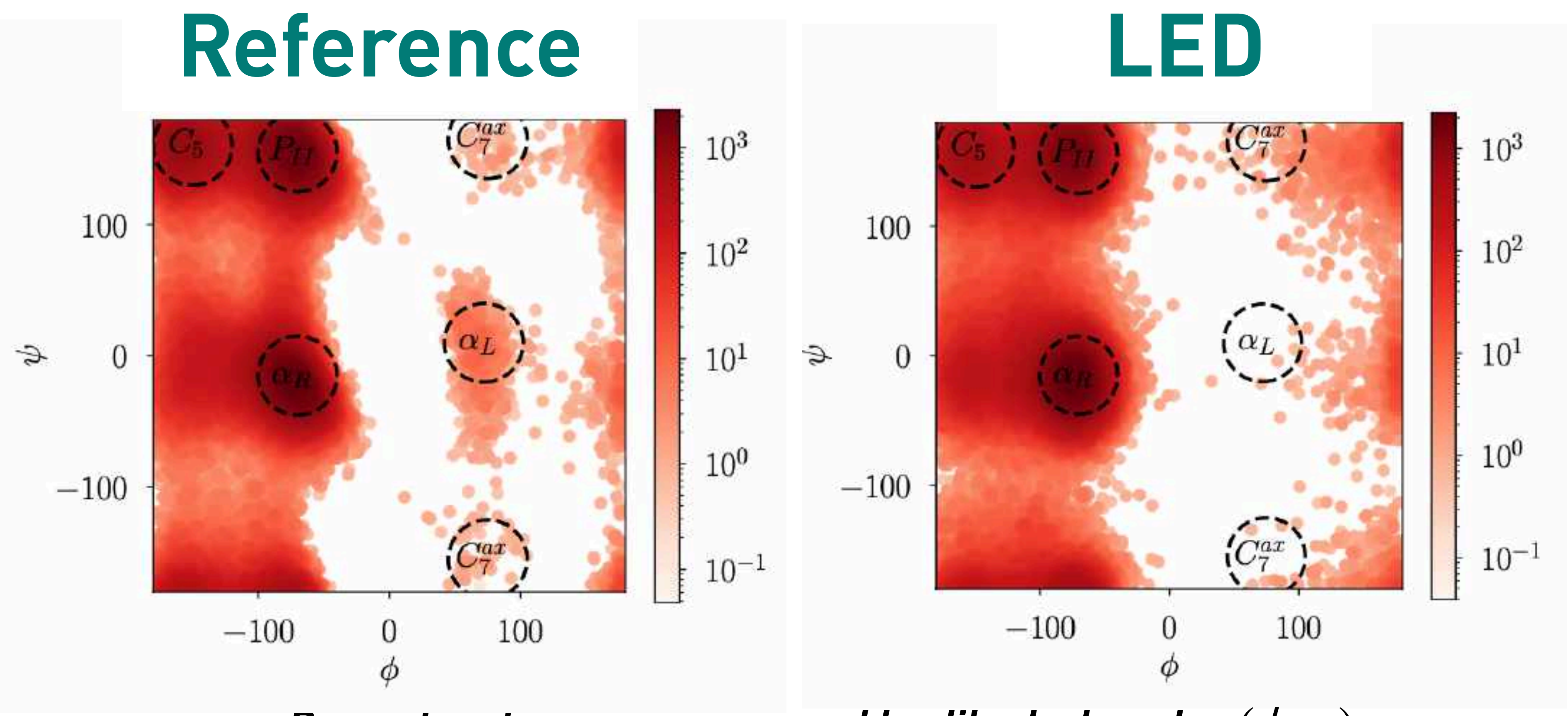


- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O**, **N**, **C**), ignore hydrogens, solvent.
- **LED with $d_z = 1$, *train on 80* ns *of data, randomly sample 500* ns *!***
  - reproduces coarsely the statistics

Alanine Dipeptide

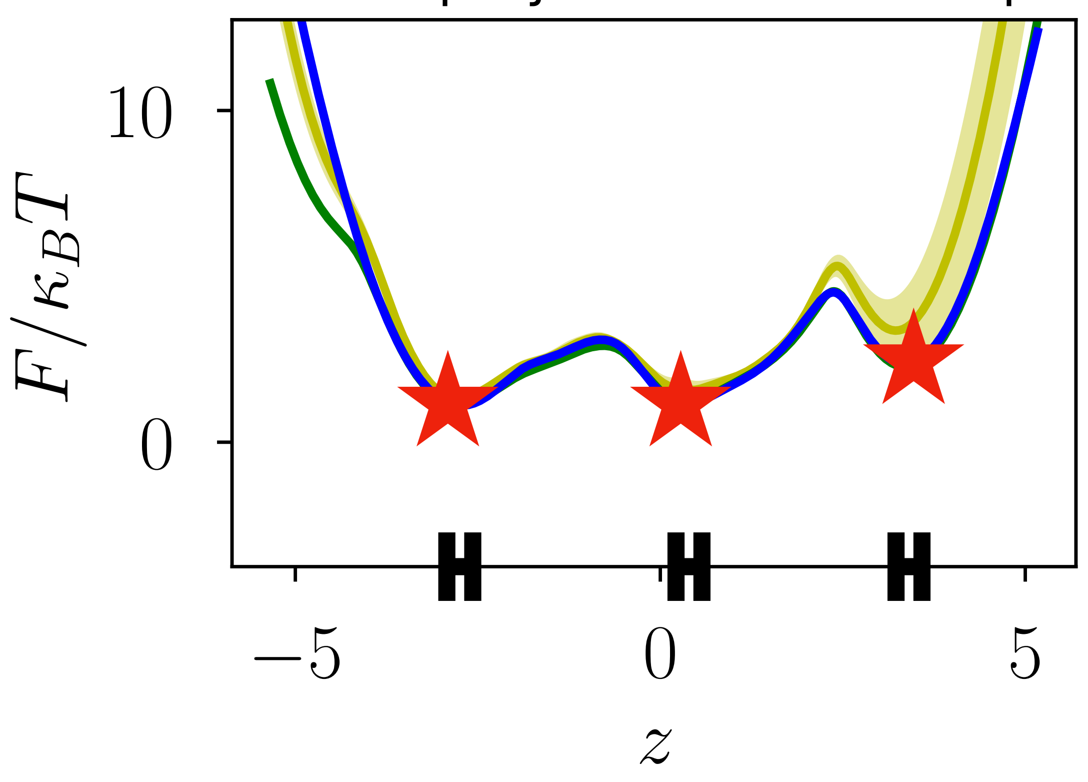
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021





Ramachandran space spanned by dihedral angles (ϕ, ψ)

Free energy $F = -\kappa_B T \ln p(z)$: molecular data projected to latent space !

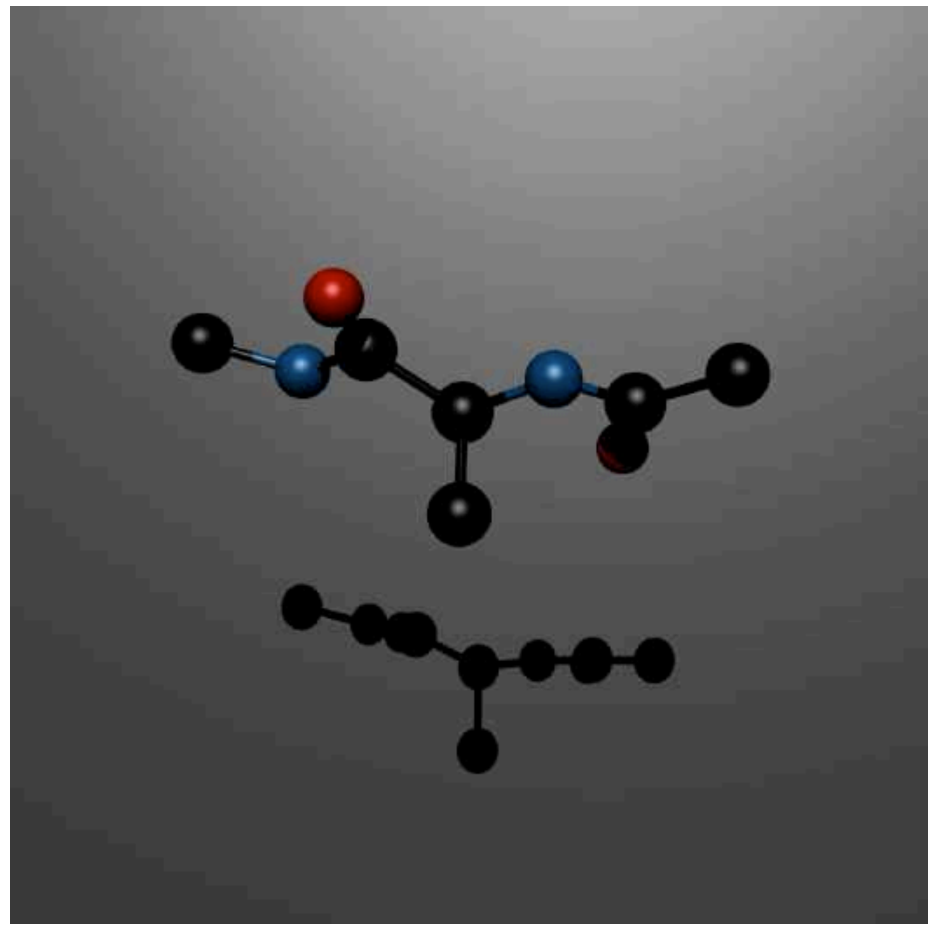


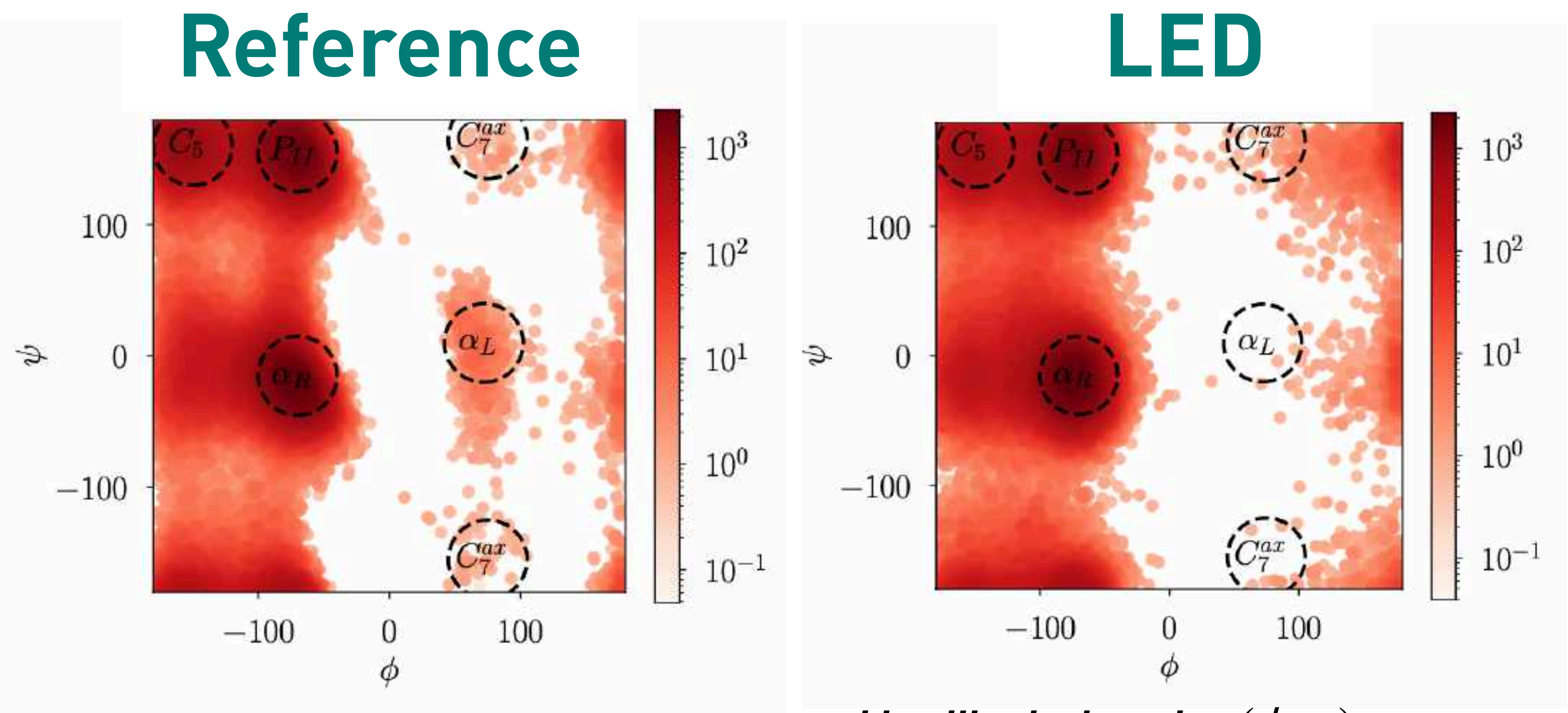
Low free-energy ⟹ exponentially more frequently visited state

- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O, N, C**), ignore hydrogens, solvent.
- **LED with $d_z = 1$, *train on 80* ns *of data, randomly sample 500* ns *!***
  - reproduces coarsely the statistics

Alanine Dipeptide

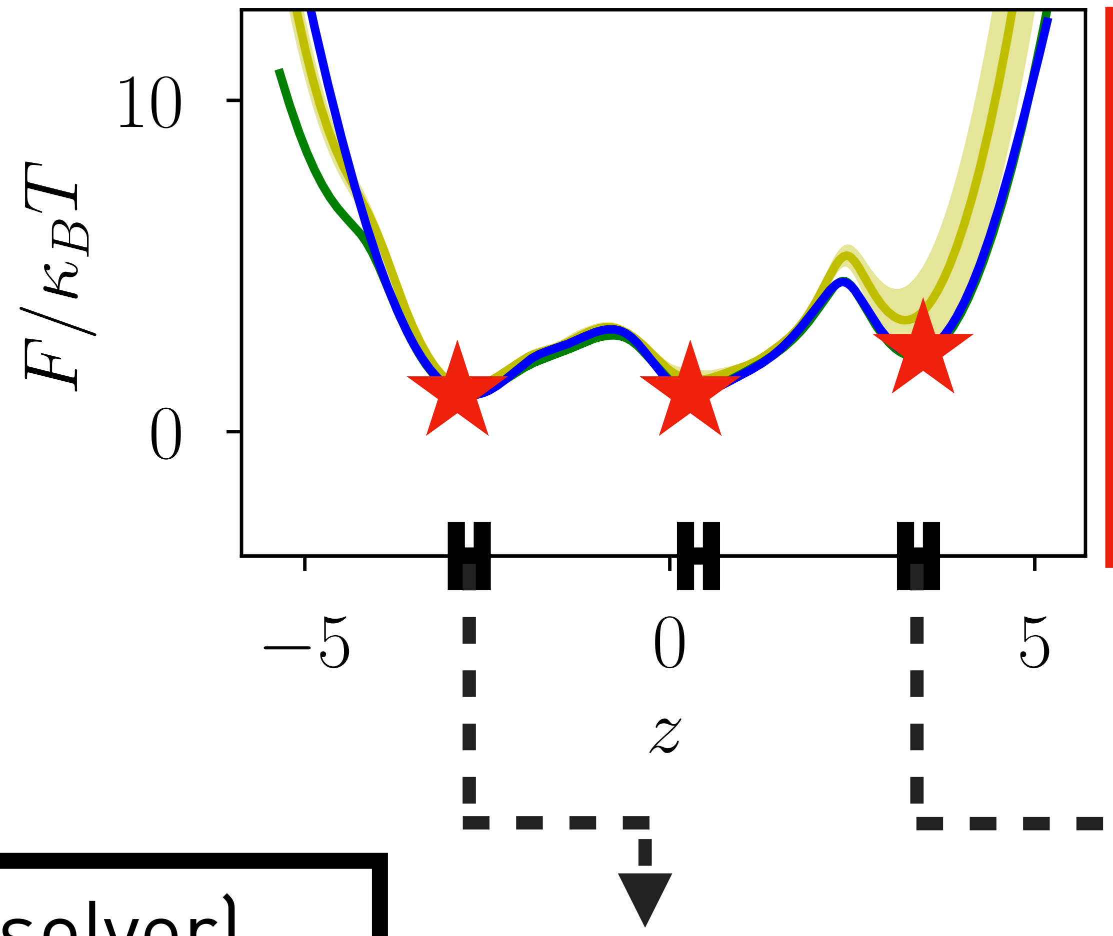
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



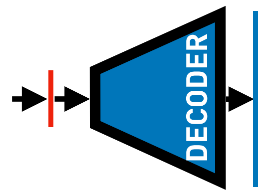


Ramachandran space spanned by dihedral angles (ϕ, ψ)

Free energy $F = -\kappa_B T \ln p(z)$: molecular data projected to latent space !



Low free-energy ⟹ exponentially more frequently visited state

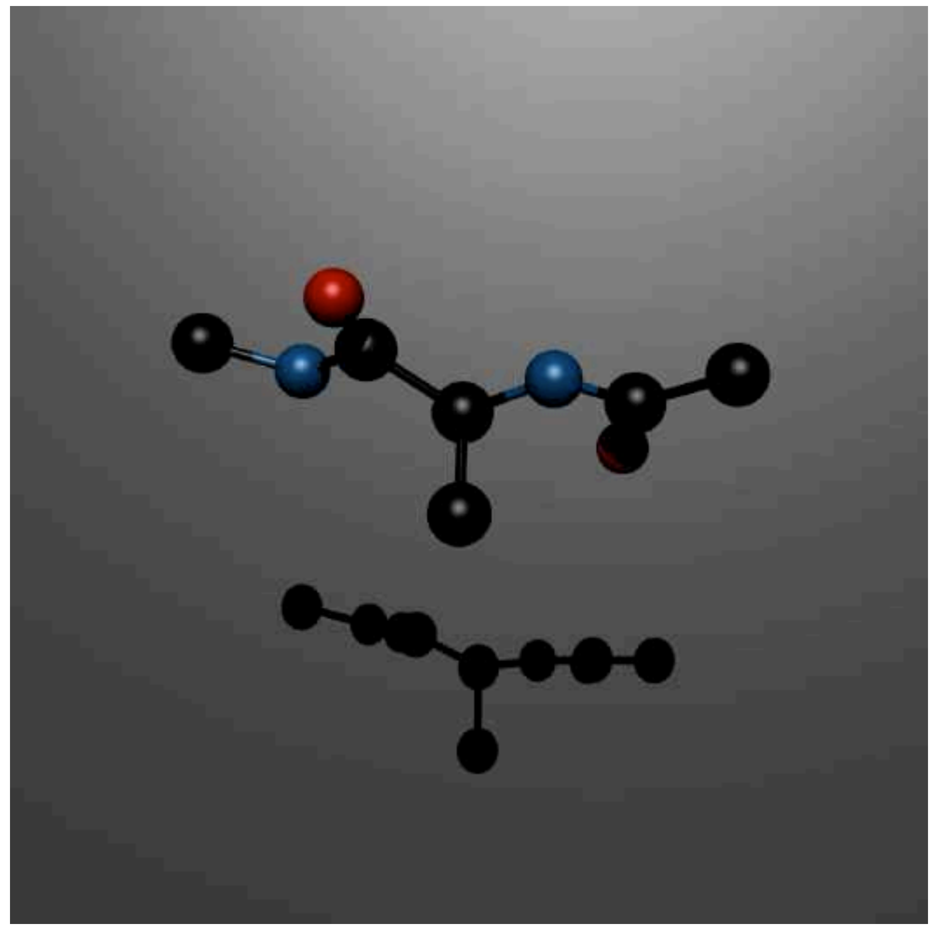


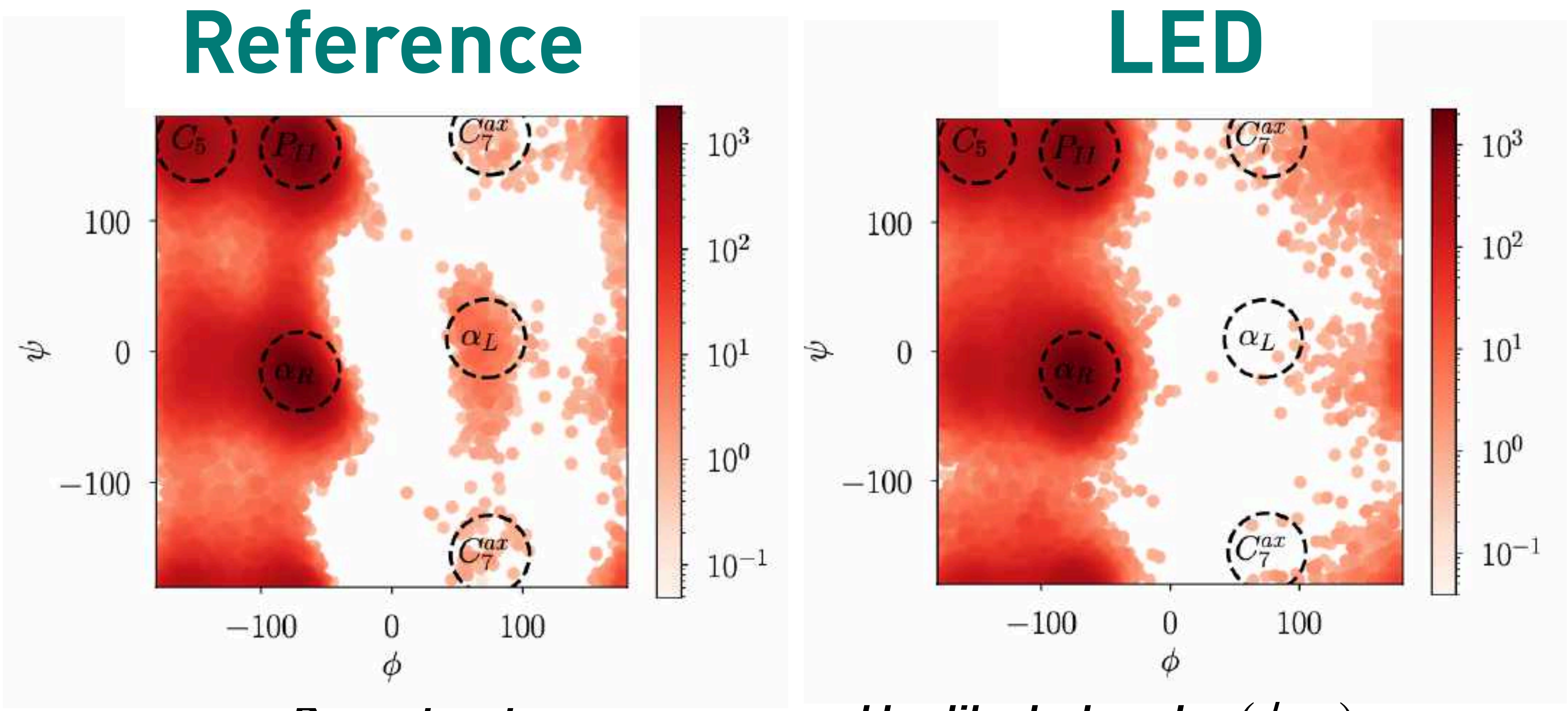
$\mathbf{z}_t \rightarrow$ DECODER $\sim \mathbf{s}_t$

- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O**, **N**, **C**), ignore hydrogens, solvent.
- **LED with $d_z = 1$, *train on 80* ns *of data, randomly sample 500* ns *!***
  - reproduces coarsely the statistics

Alanine Dipeptide

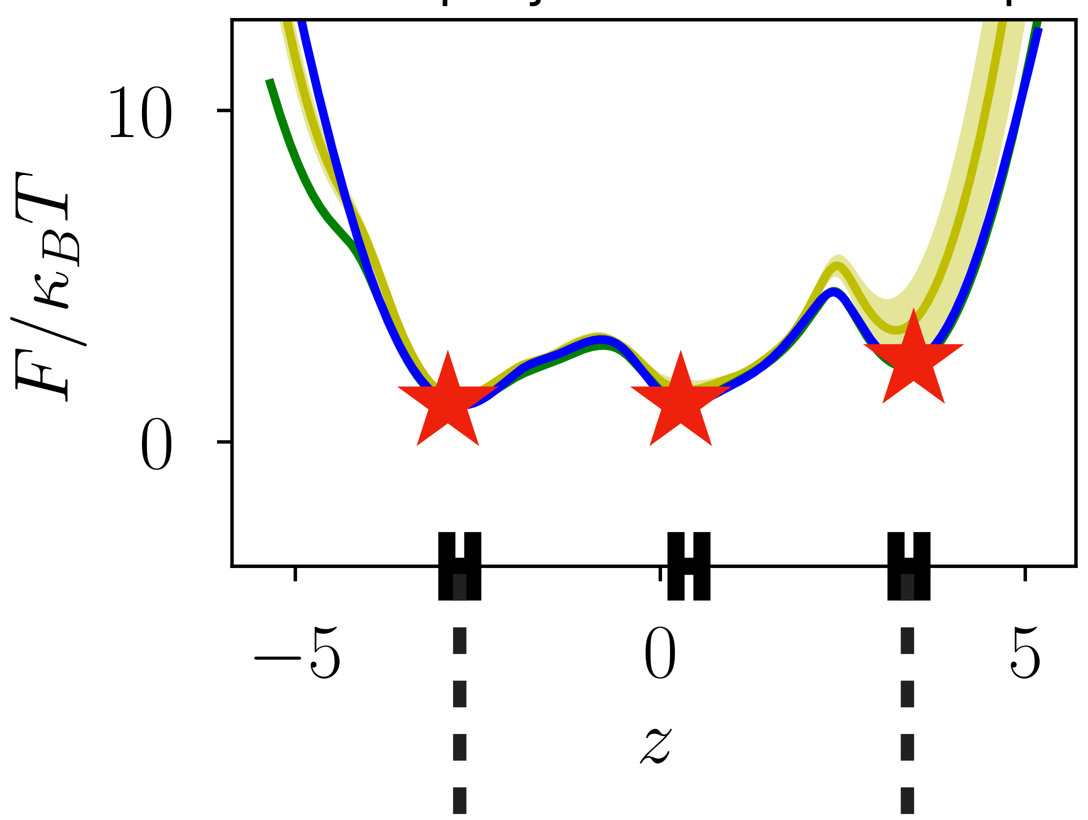
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



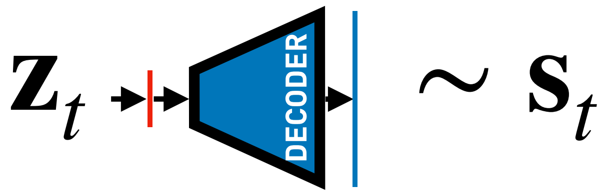


Ramachandran space spanned by dihedral angles (ϕ, ψ)

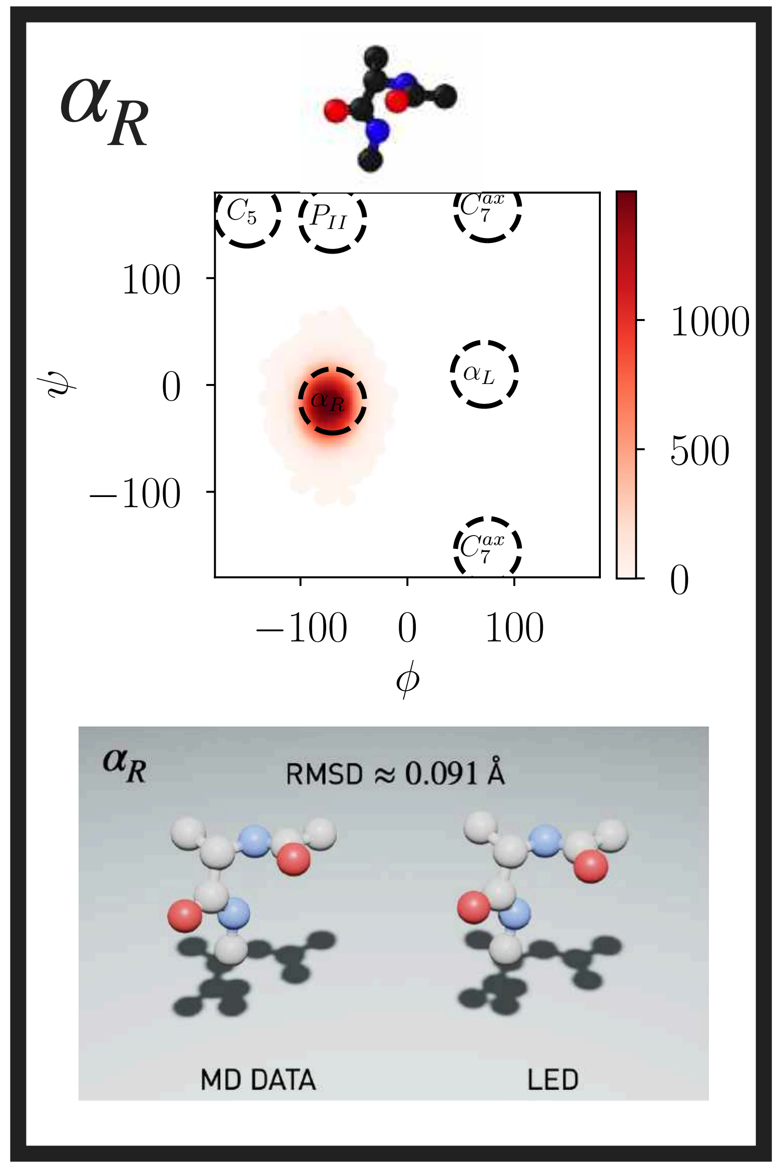
Free energy $F = -\kappa_B T \ln p(z)$: molecular data projected to latent space !

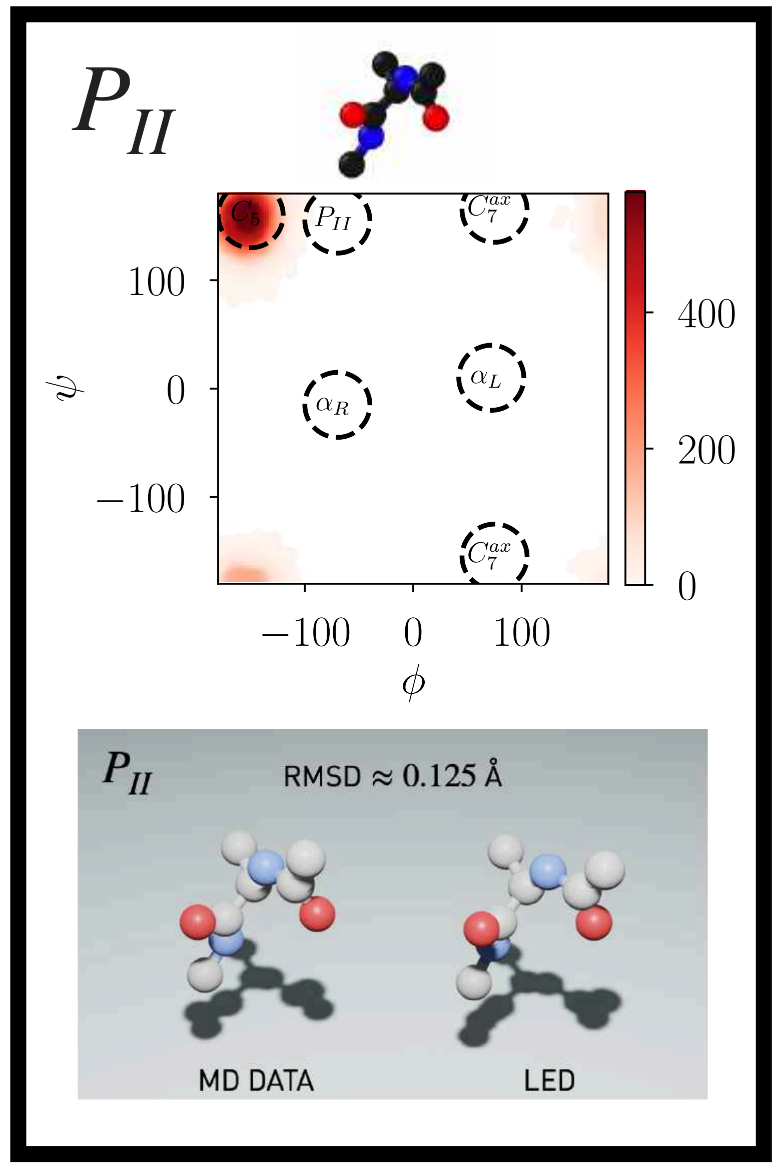


Low free-energy ⟹ exponentially more frequently visited state



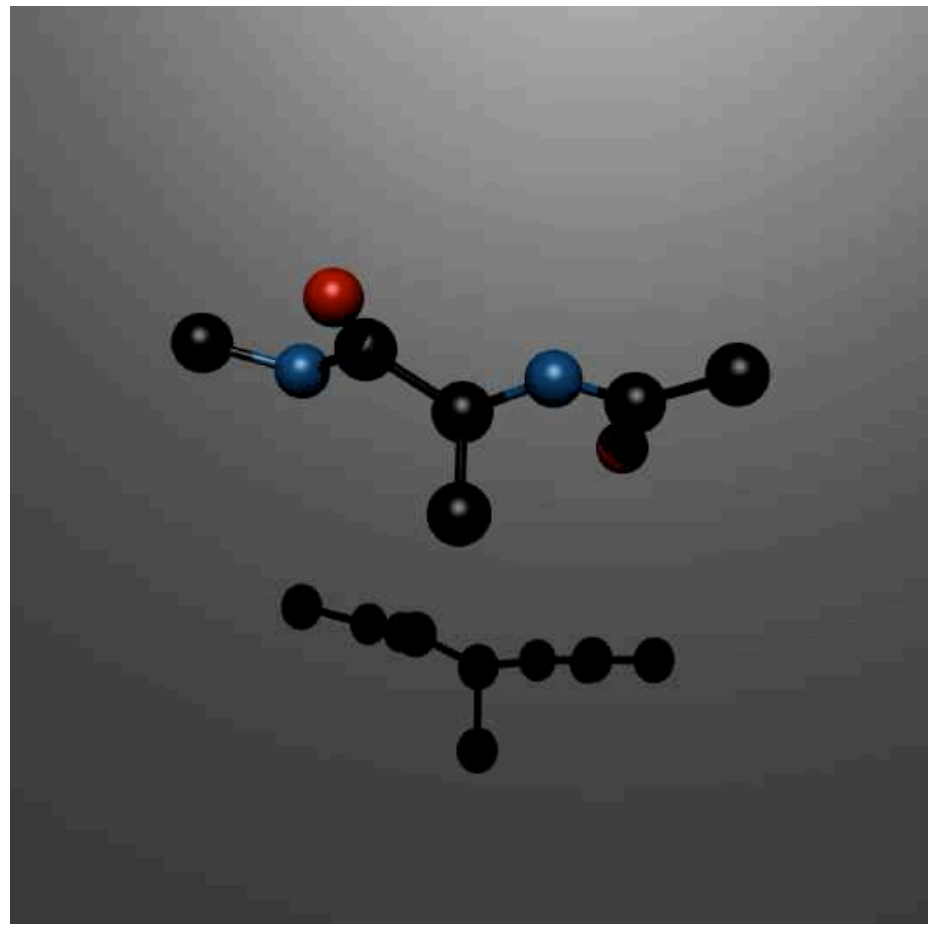
- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O, N, C**), ignore hydrogens, solvent.
- **LED with $d_z = 1$, *train on 80* ns *of data, randomly sample 500* ns *!***
  - reproduces coarsely the statistics
  - reproduces kinetics, and mean transition times
  - free energy profile ***minima*** are mapped to configurations of meta stable states

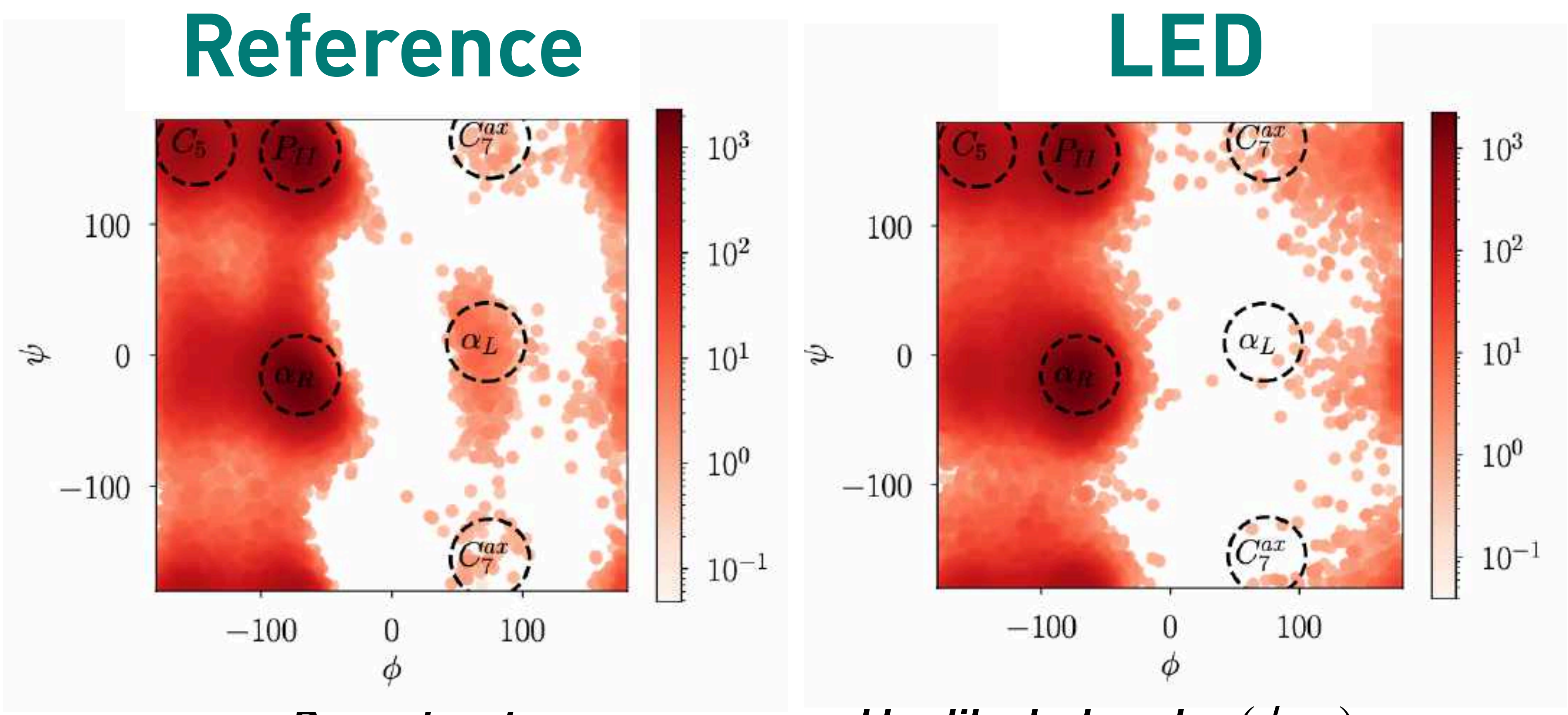




Alanine Dipeptide

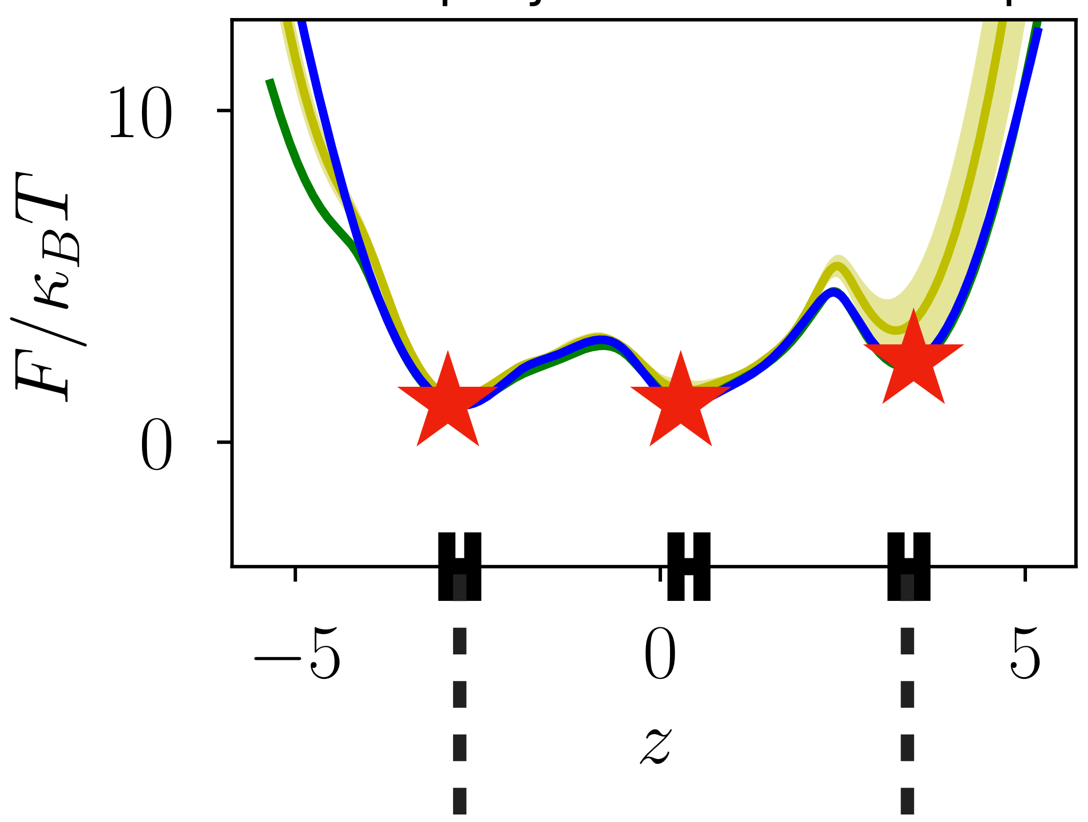
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos,
Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,
Journal of Chemical Theory & Computation, 2021



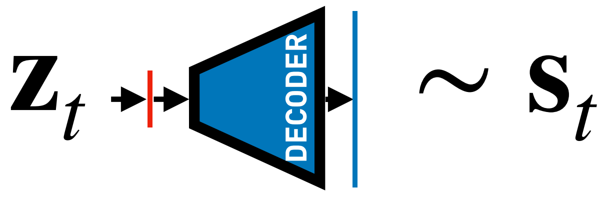


Ramachandran space spanned by dihedral angles (ϕ, ψ)

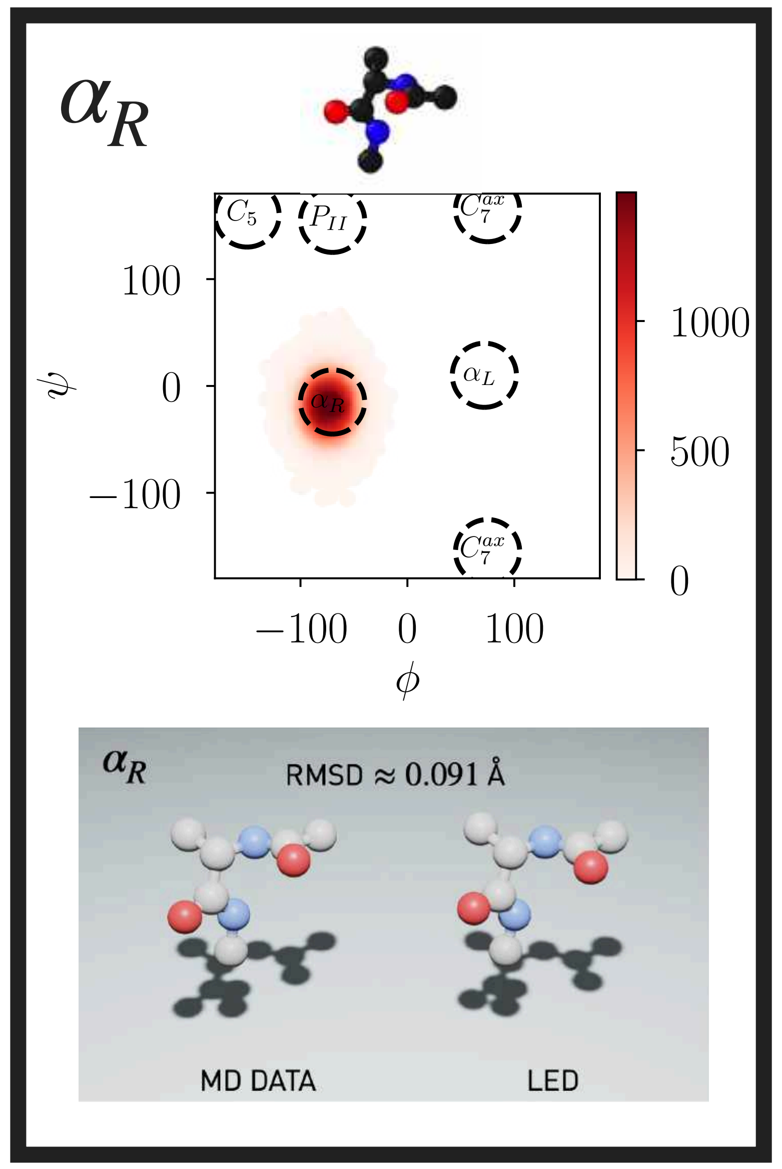
Free energy $F = -\kappa_B T \ln p(z)$: molecular data projected to latent space !

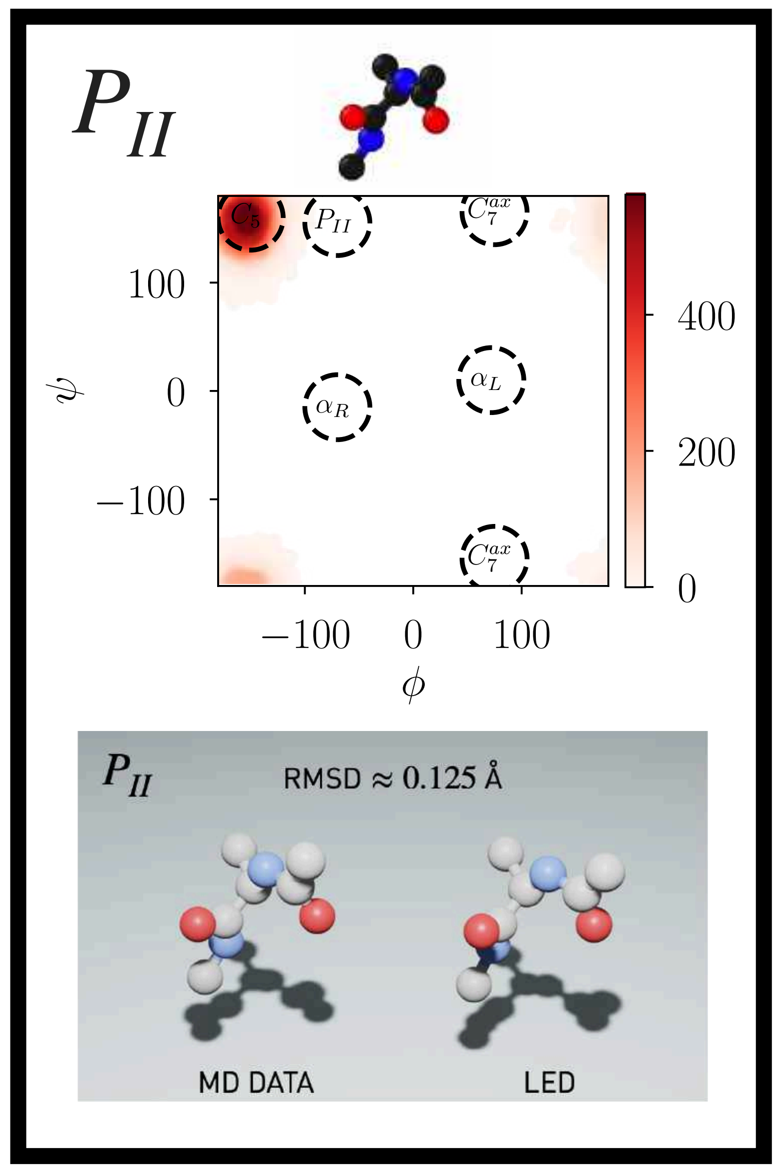


Low free-energy $\Longrightarrow$ exponentially more frequently visited state



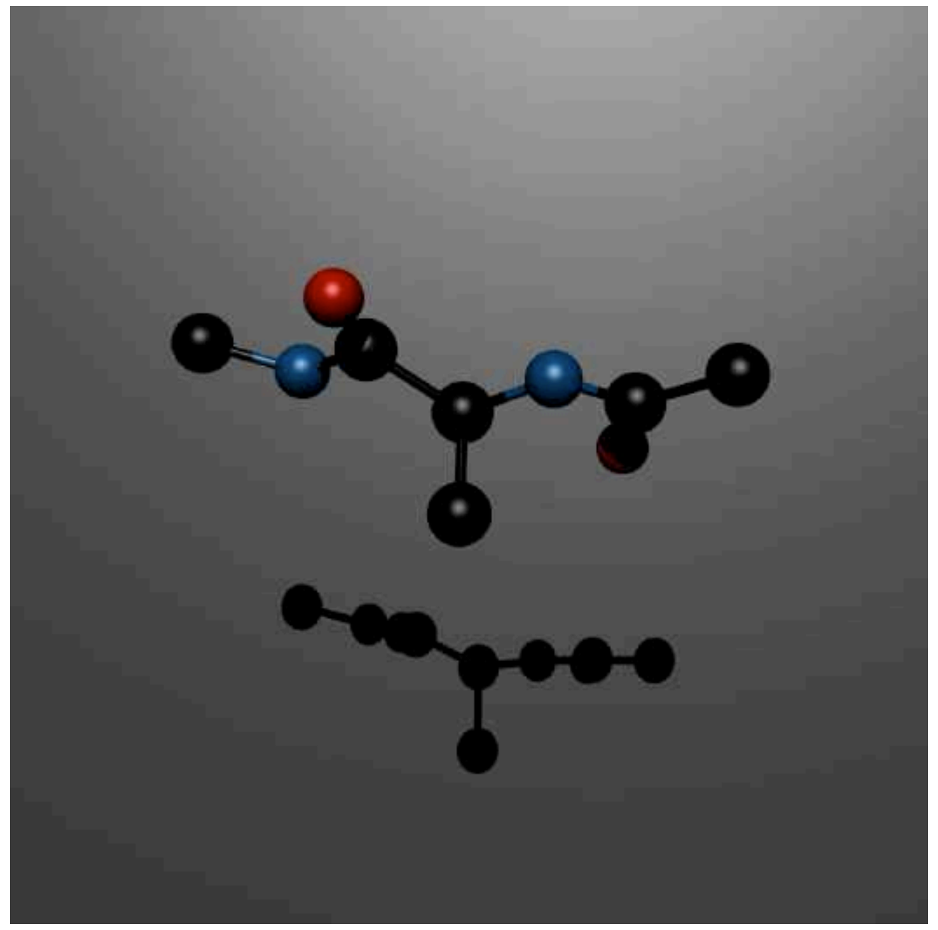
- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O, N, C**), ignore hydrogens, solvent.
- **LED with $d_\mathbf{z} = 1$, *train on 80* ns *of data, randomly sample 500* ns *!***
  - reproduces coarsely the statistics
  - reproduces kinetics, and mean transition times
  - free energy profile ***minima*** are mapped to configurations of meta stable states
  - samples realistic configurations

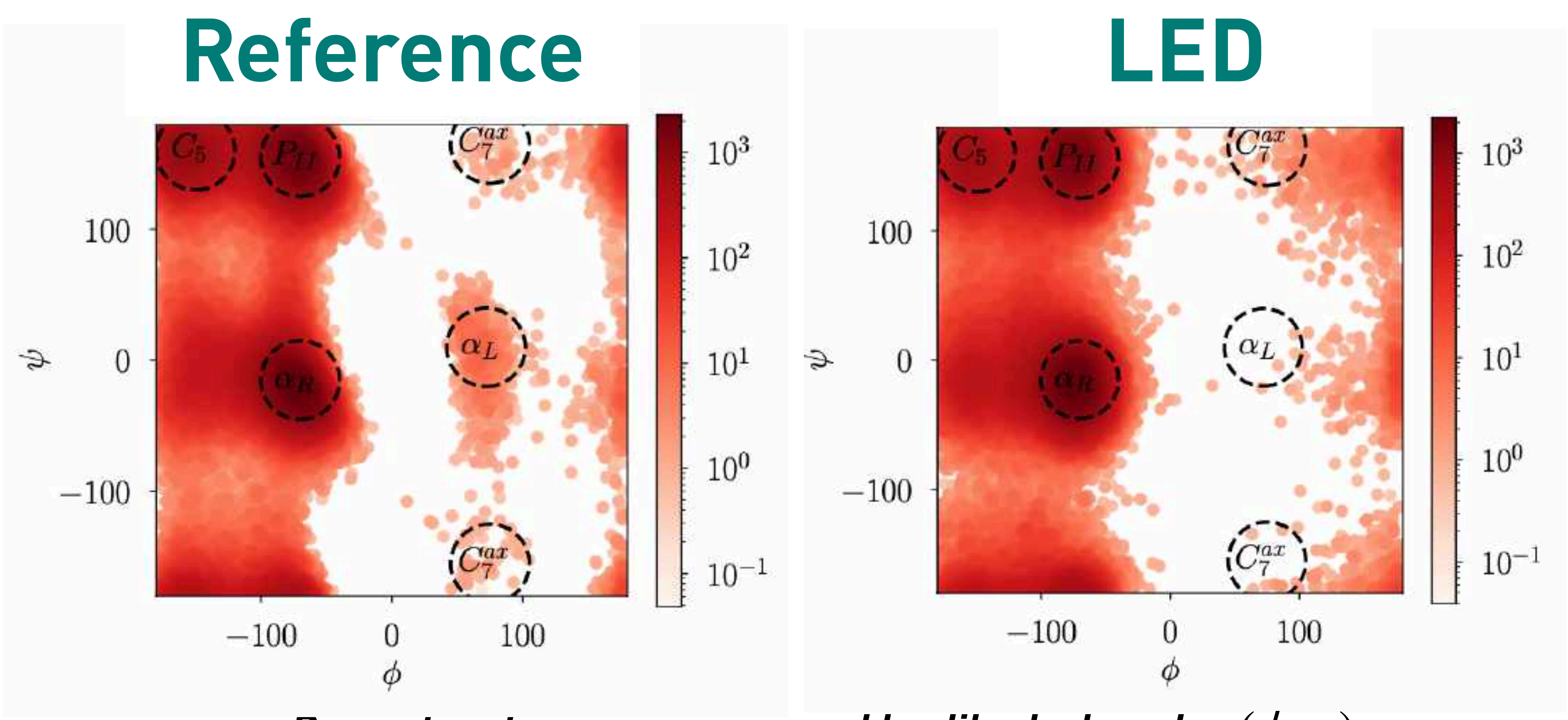




Alanine Dipeptide

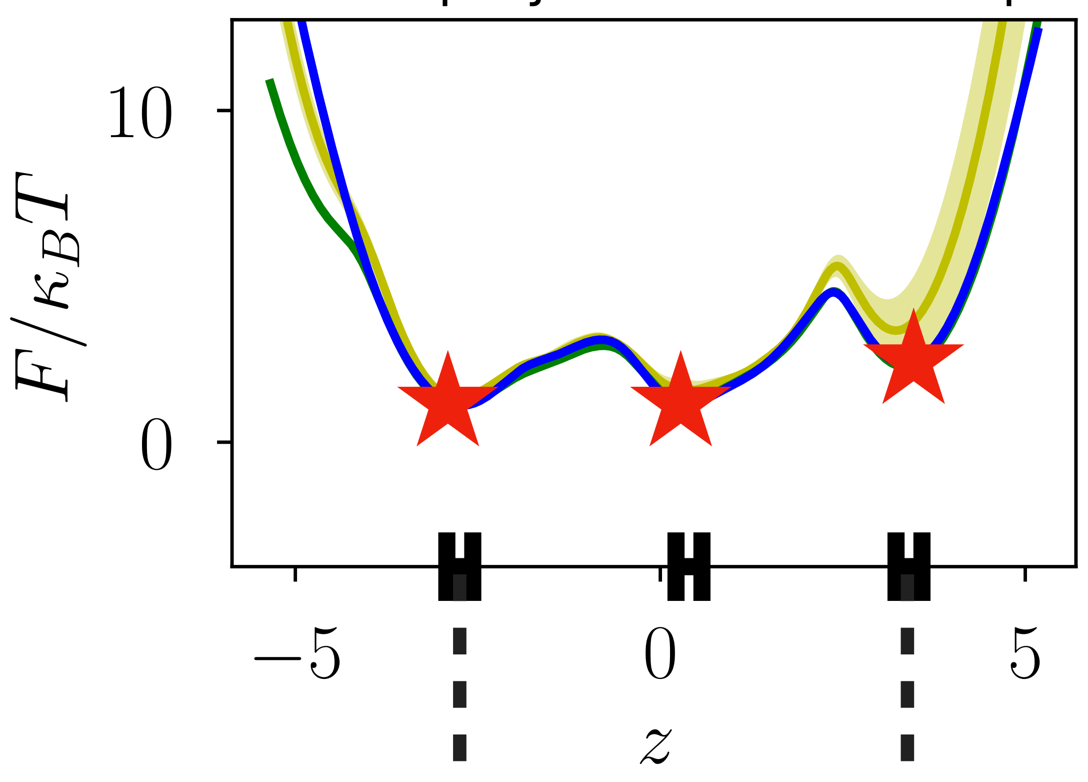
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos, *Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics*, **Journal of Chemical Theory & Computation, 2021**



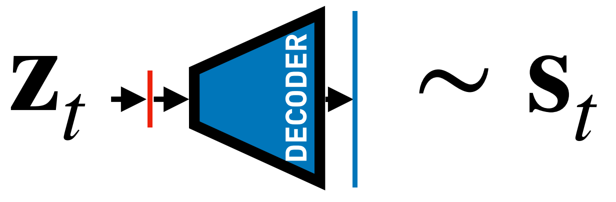


Ramachandran space spanned by dihedral angles (ϕ, ψ)

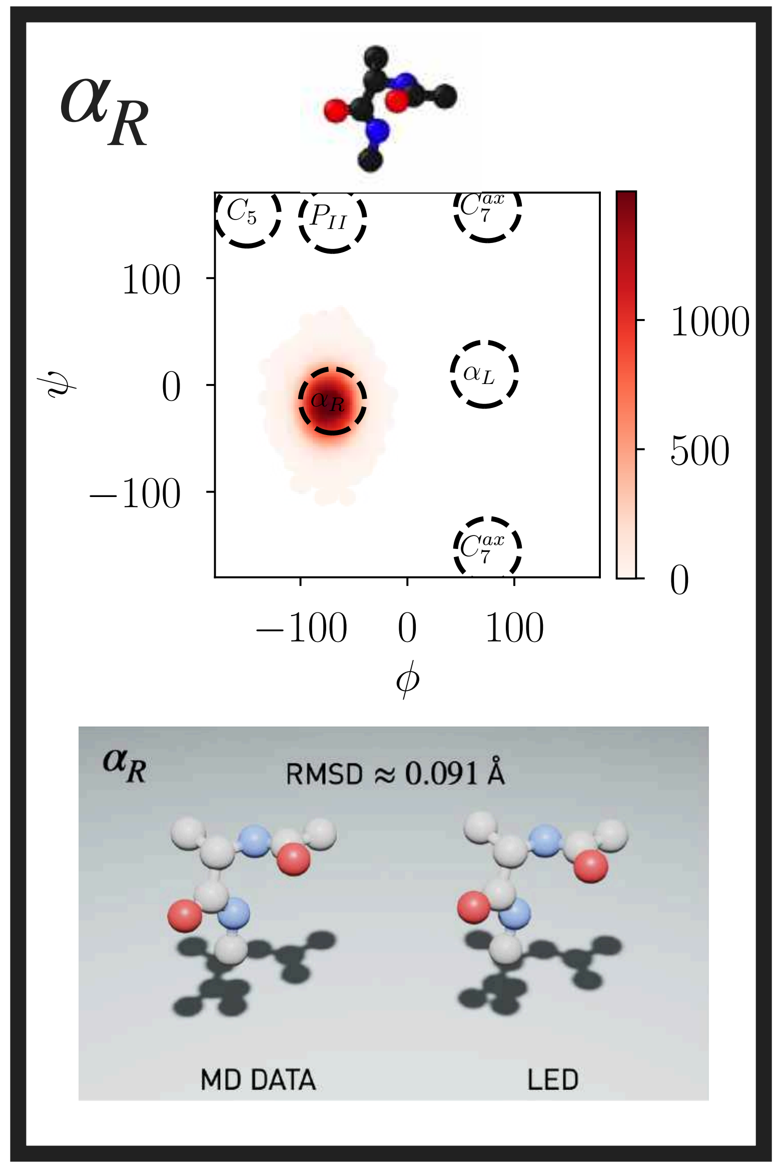
Free energy $F = -\kappa_B T \ln p(z)$: molecular data projected to latent space !

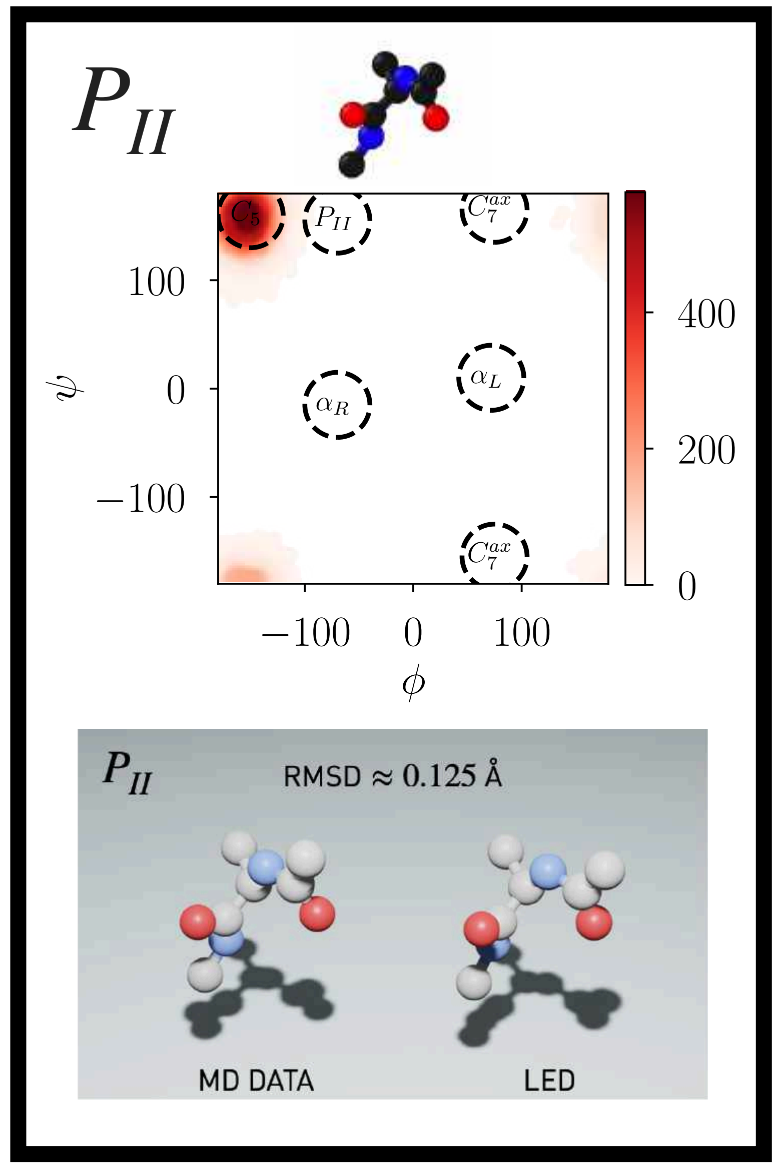


Low free-energy ⟹ exponentially more frequently visited state



- Alanine dipeptide dynamics in water solved with Molecular Dynamics (MD solver)
- Stochastic transitions between ***5 meta-stable states*** $\{P_{II}, C_5, \alpha_R, \alpha_L, C_7^{ax}\}$
- State $\mathbf{s}_t \in \mathbb{R}^{24}$ (bond lengths, angles) of only the heavy atoms (**O**, **N**, **C**), ignore hydrogens, solvent.
- **LED with** $d_\mathbf{z} = 1$, ***train on 80 ns of data, randomly sample 500 ns !***
  - reproduces coarsely the statistics
  - reproduces kinetics, and mean transition times
  - free energy profile ***minima*** are mapped to configurations of meta stable states
  - samples realistic configurations
- LED is at least **three orders faster (x1000)** than MD solver





Avenues for Future Research

Avenues for Future Research

- Investigate alternative architectures

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)
- Applications

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)
- Applications
  - Scale to 3D fluid flows

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)
- Applications
  - Scale to 3D fluid flows
  - Employ on RL for data-efficiency

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)
- Applications
  - Scale to 3D fluid flows
  - Employ on RL for data-efficiency
  - AdaLED on real world experiments (challenge: how to reinitialise experiment from micro-scale)

Avenues for Future Research

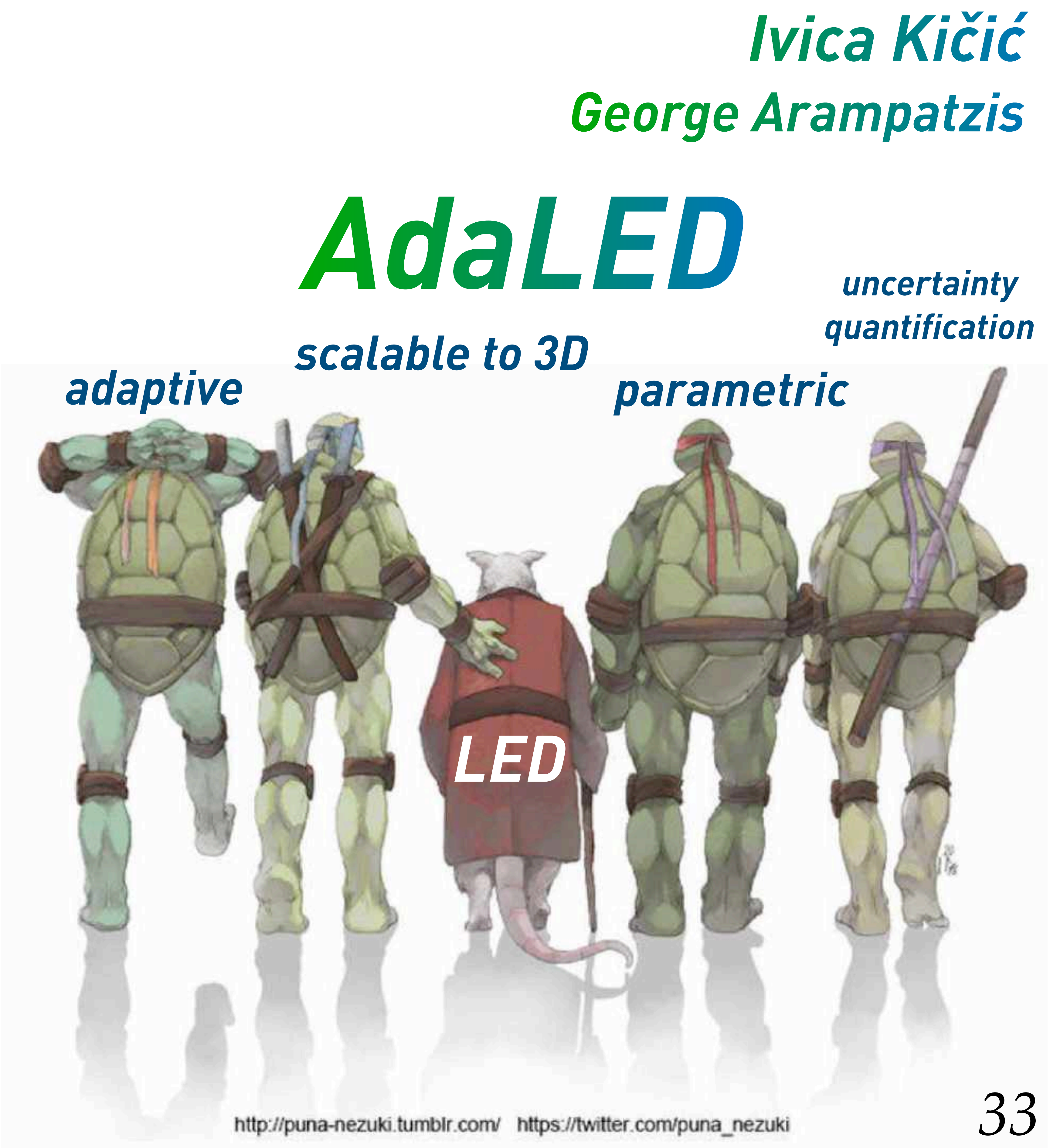
- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)
- Applications
  - Scale to 3D fluid flows
  - Employ on RL for data-efficiency
  - AdaLED on real world experiments (challenge: how to reinitialise experiment from micro-scale)
  - Climate data

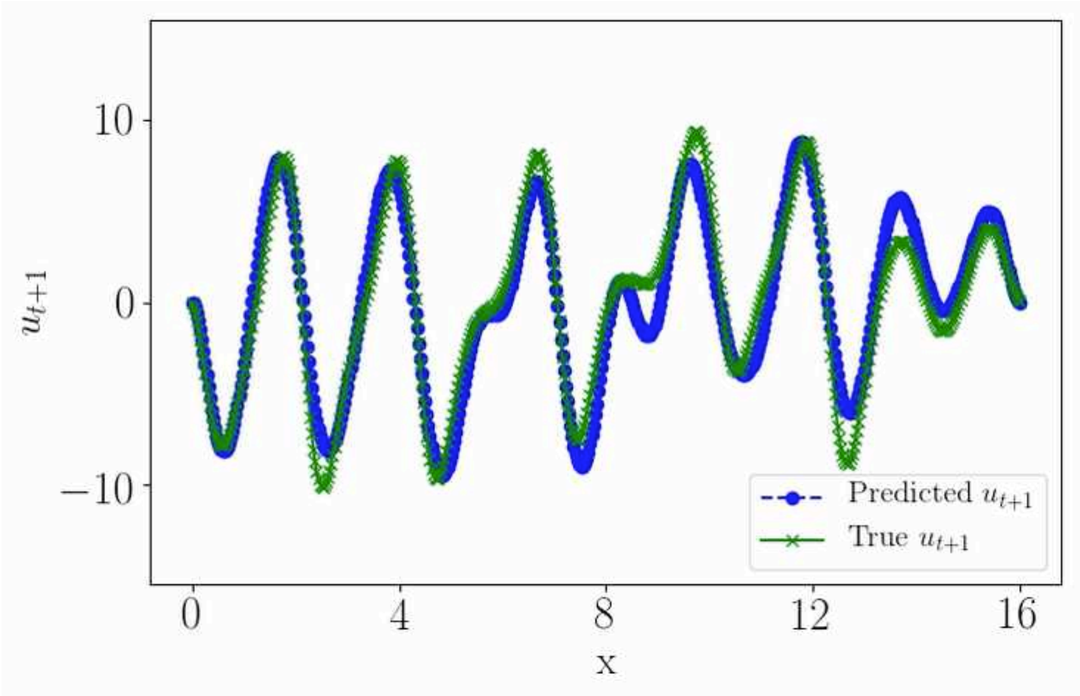
Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)
- Applications
  - Scale to 3D fluid flows
  - Employ on RL for data-efficiency
  - AdaLED on real world experiments (challenge: how to reinitialise experiment from micro-scale)
  - Climate data
  - Digital twin paradigm

Avenues for Future Research

- Investigate alternative architectures
  - learnable wavelet filter networks, auto-encoding normalising flows, or graph mixture density networks
  - multi timescale recurrent neural networks
- Learning Effective Dynamics (LED)
  - Adaptive online learning (AdaLED), with uncertainty quantification
  - parametric (dependence on physical parameters)
  - system properties into architecture (e.g. energy conservation, divergence free velocity field, etc.)
  - Adaptive resolution (AMR)
- Applications
  - Scale to 3D fluid flows
  - Employ on RL for data-efficiency
  - AdaLED on real world experiments (challenge: how to reinitialise experiment from micro-scale)
  - Climate data
  - Digital twin paradigm

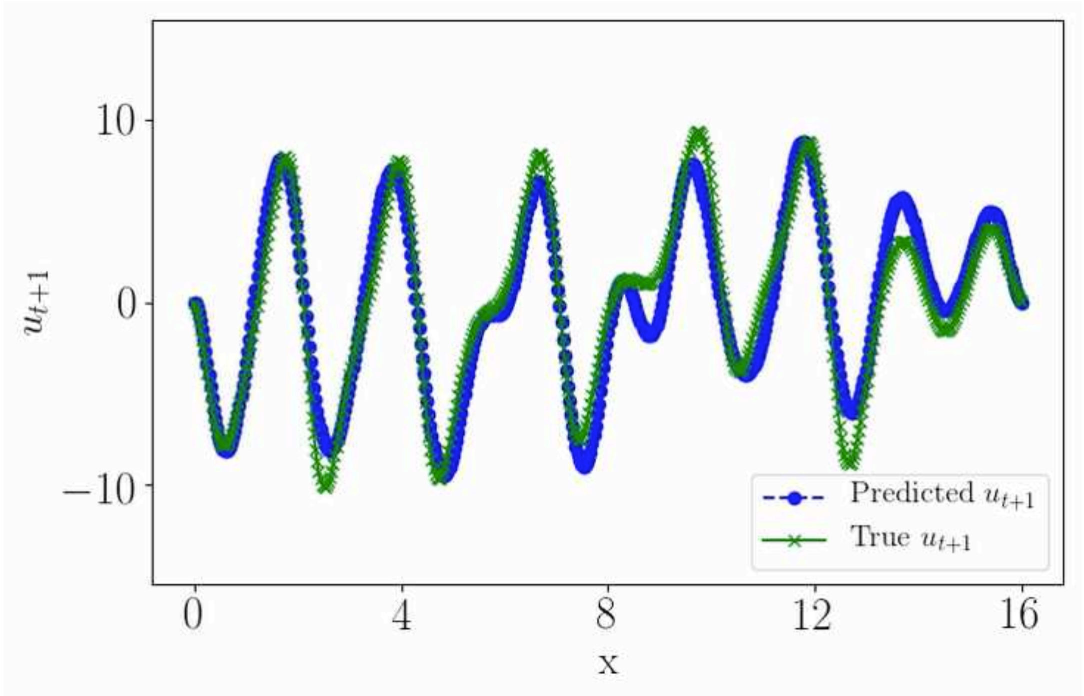




HYBRID LSTM - MSM

$$\dot{z}_t = \begin{cases} \mathrm{LSTM}^{\mathbf{W}}(z_t, z_{t-1}, z_{t-2}, \ldots) & \textbf{if } p_{train}(z_t) \geq \theta \\ \mathrm{MSM}^{\zeta,c}(z_t) & \textbf{if } p_{train}(z_t) < \theta \end{cases}$$

PR Vlachas, W Byeon, Z Wan, T Sapsis, P Koumoutsakos, *Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks,* **Proc. Roy. Soc. A , 2018**

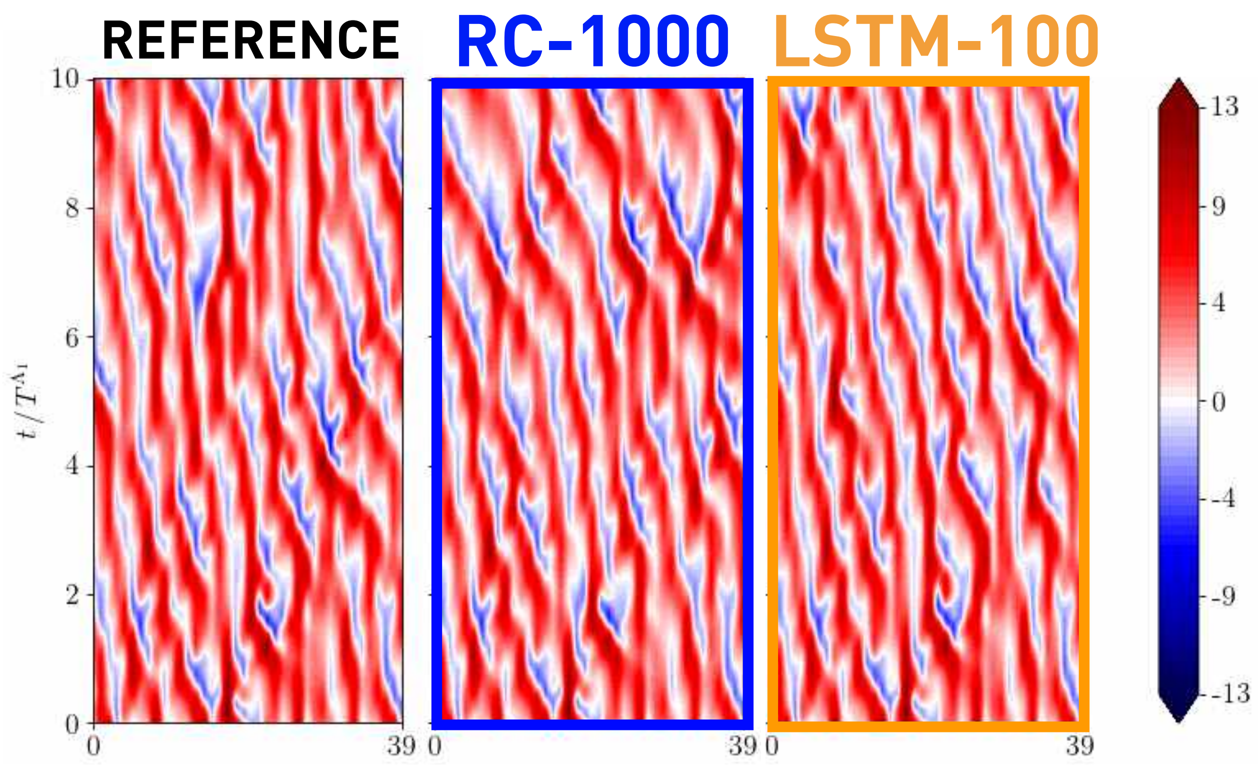


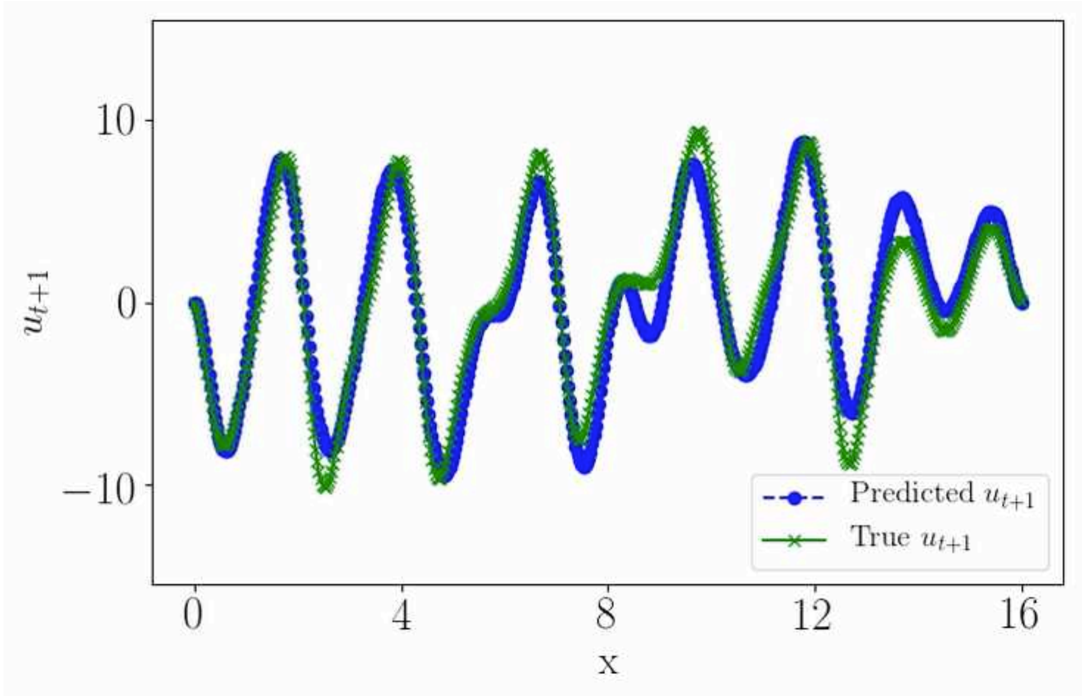
HYBRID LSTM - MSM

$$\dot{z}_t = \begin{cases} \mathrm{LSTM}^{\mathbf{W}}(z_t, z_{t-1}, z_{t-2}, \ldots) & \text{if } p_{train}(z_t) \geq \theta \\ \mathrm{MSM}^{\zeta,c}(z_t) & \text{if } p_{train}(z_t) < \theta \end{cases}$$

PR Vlachas, W Byeon, Z Wan, T Sapsis, P Koumoutsakos, *Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks,* **Proc. Roy. Soc. A , 2018**

PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **Journal of Neural Networks, 2020**



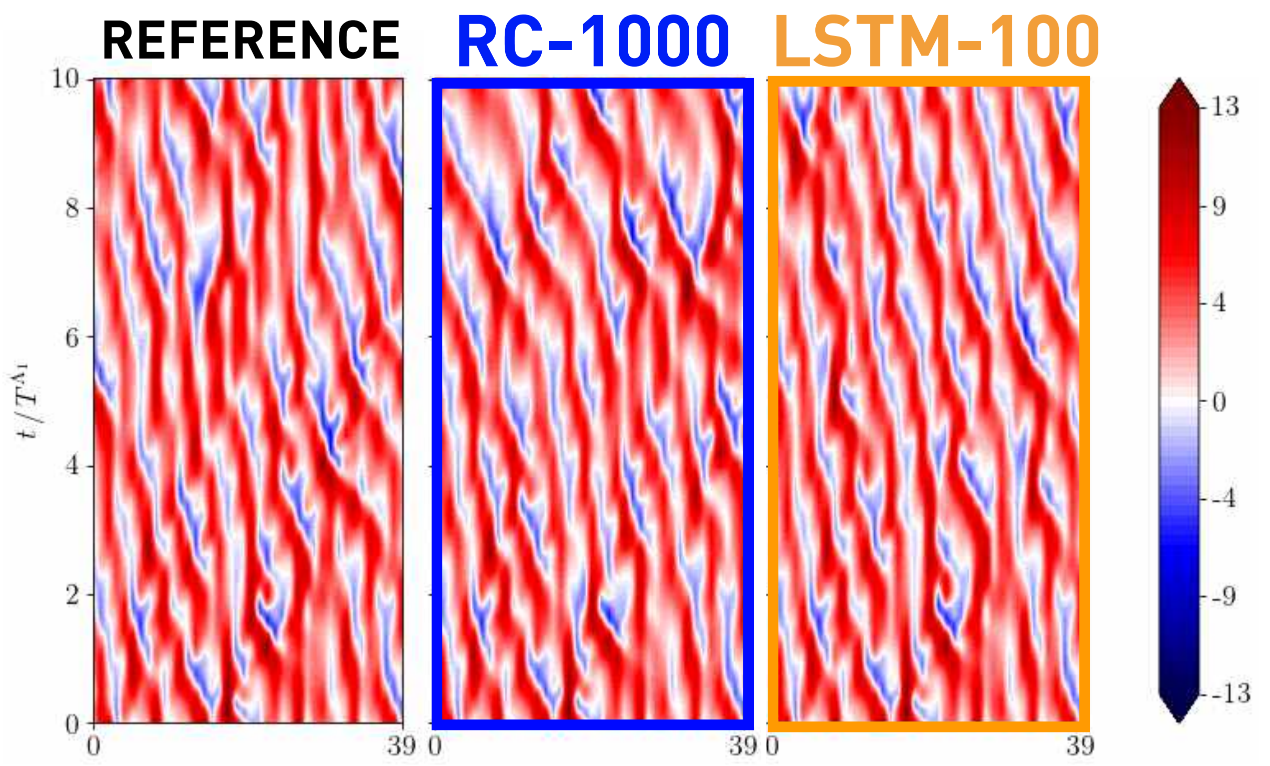


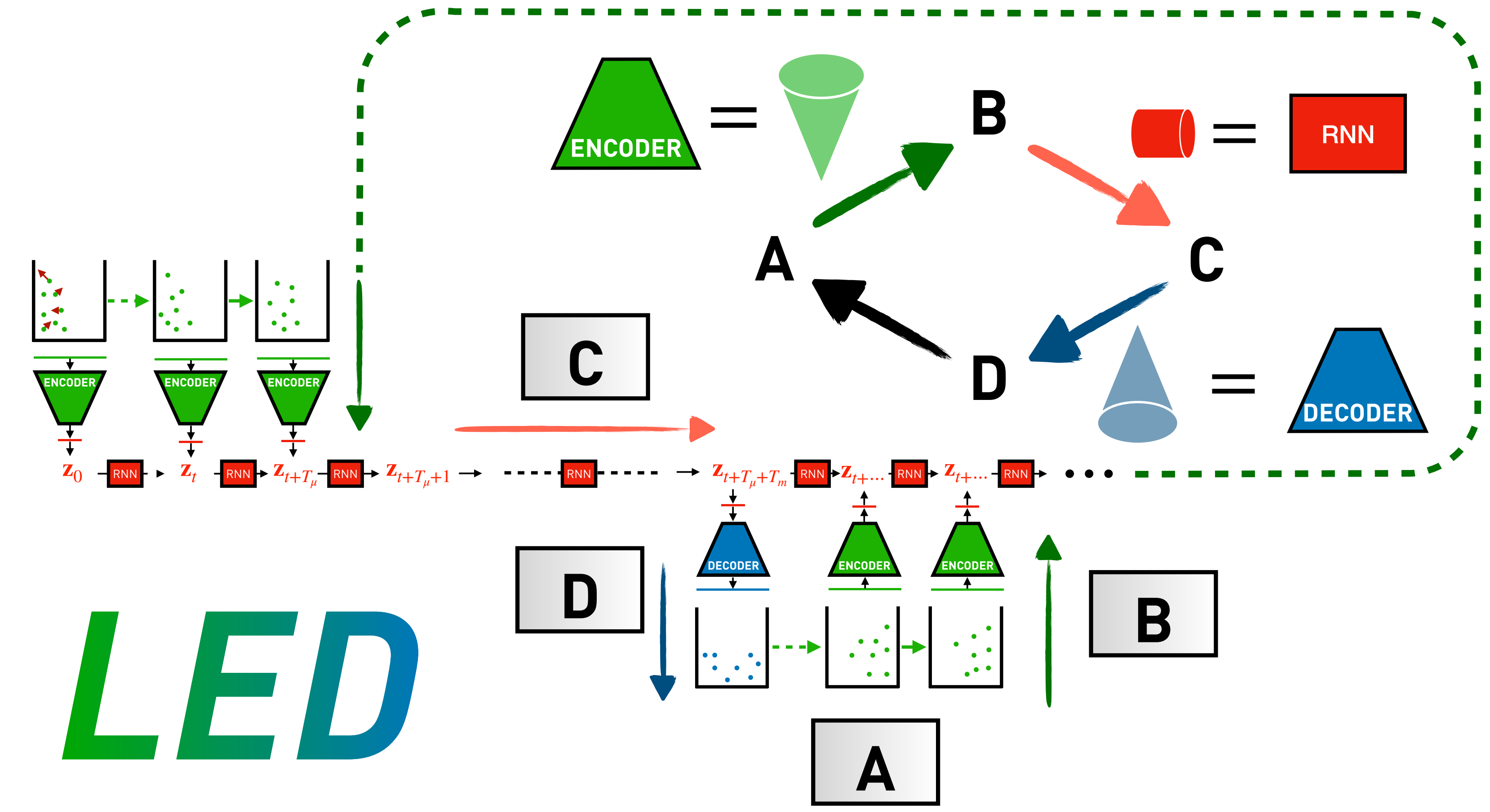
HYBRID LSTM - MSM

$$\dot{z}_t = \begin{cases} \mathrm{LSTM}^{\mathbf{W}}(z_t, z_{t-1}, z_{t-2}, \dots) & \text{if } p_{train}(z_t) \geq \theta \\ \mathrm{MSM}^{\zeta,c}(z_t) & \text{if } p_{train}(z_t) < \theta \end{cases}$$

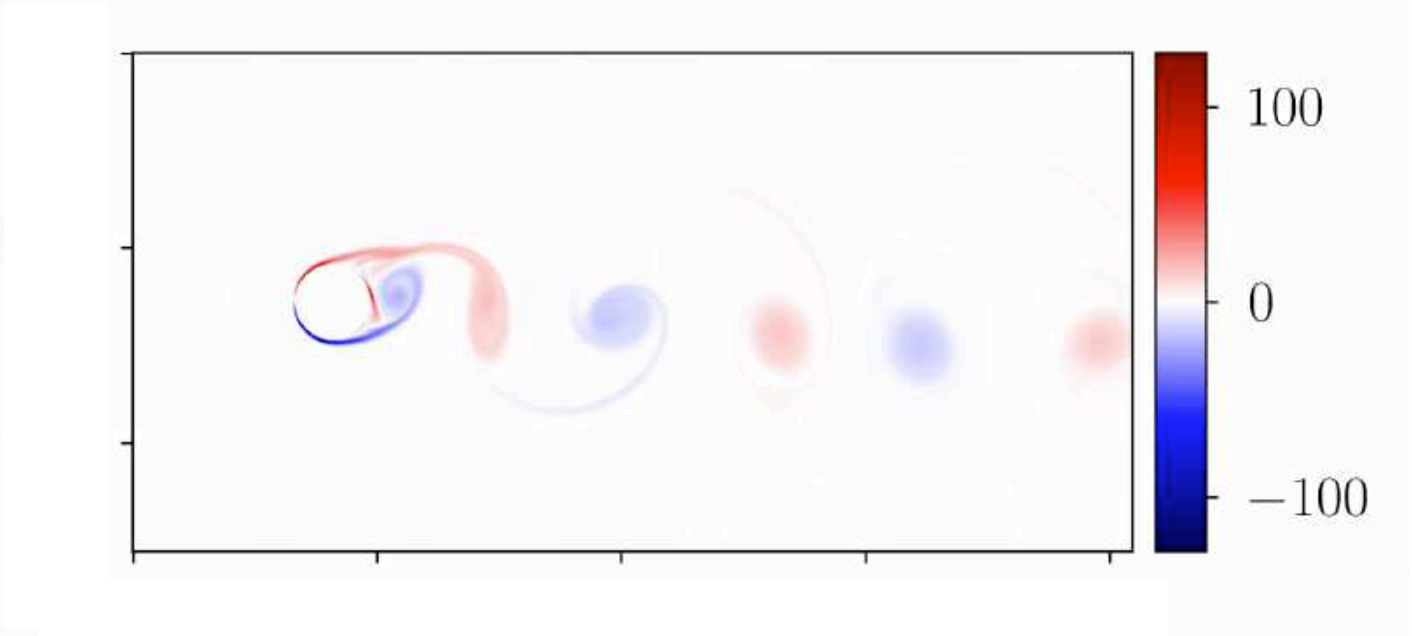
PR Vlachas, W Byeon, Z Wan, T Sapsis, P Koumoutsakos, *Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks,* **Proc. Roy. Soc. A , 2018**

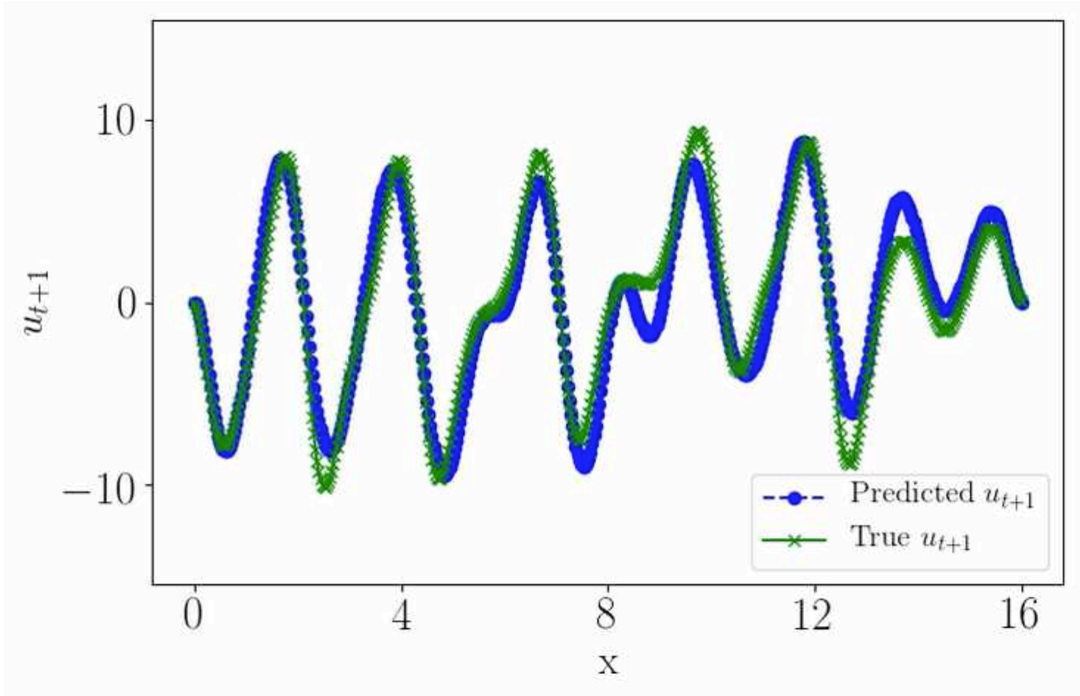
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **Journal of Neural Networks, 2020**





PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos, *Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,* **Nature Machine Intelligence, 2022**



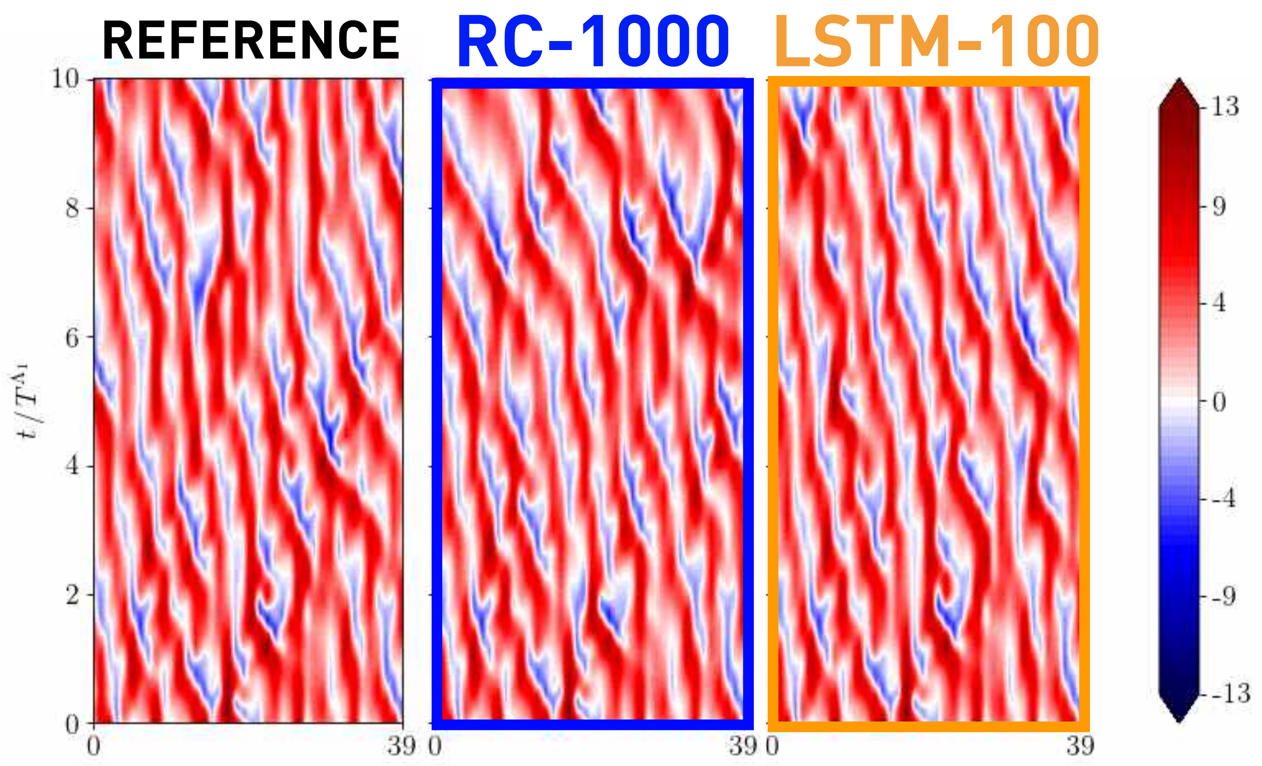


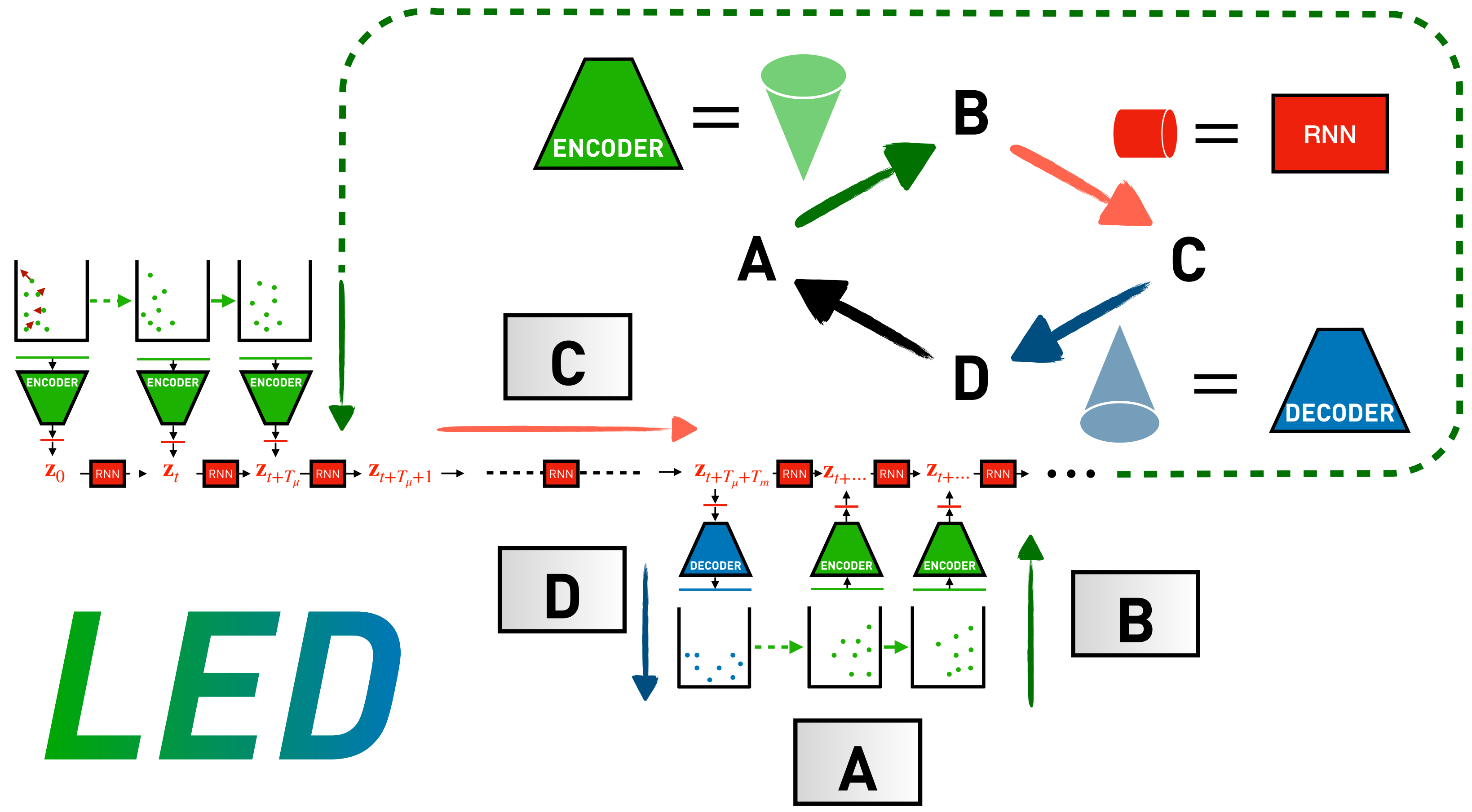
HYBRID LSTM - MSM

$$\dot{z}_t = \begin{cases} \mathrm{LSTM}^{\mathbf{W}}(z_t, z_{t-1}, z_{t-2}, \ldots) & \text{if } p_{train}(z_t) \geq \theta \\ \mathrm{MSM}^{\zeta,c}(z_t) & \text{if } p_{train}(z_t) < \theta \end{cases}$$

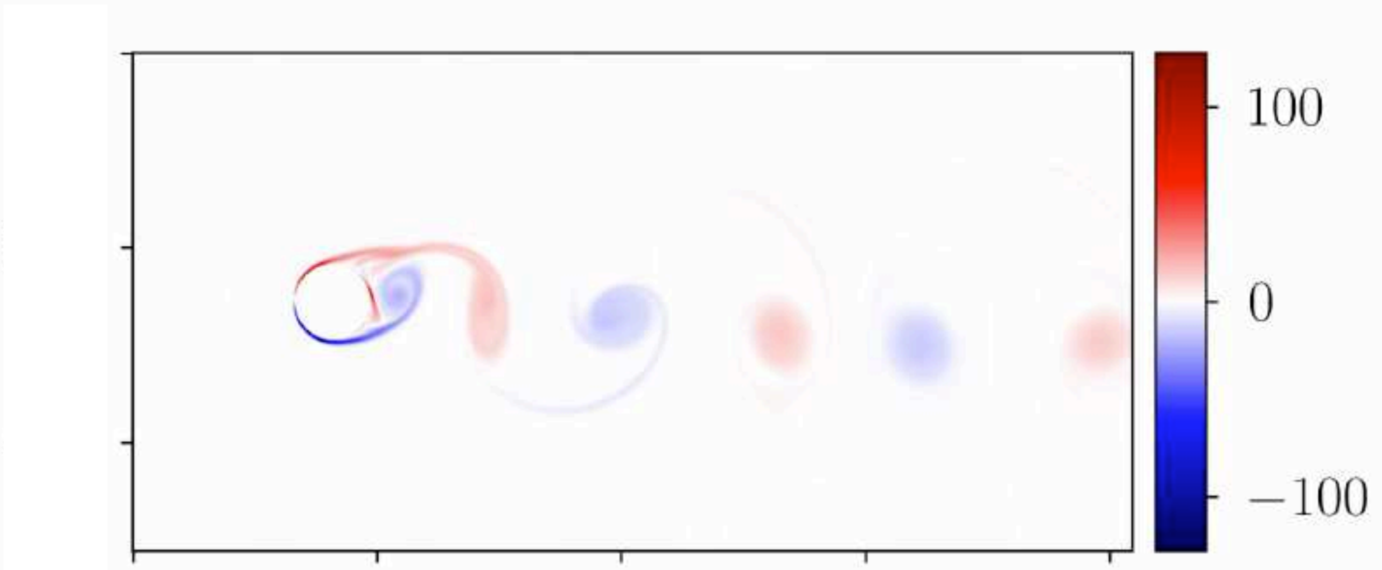
PR Vlachas, W Byeon, Z Wan, T Sapsis, P Koumoutsakos, *Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks,* **Proc. Roy. Soc. A , 2018**

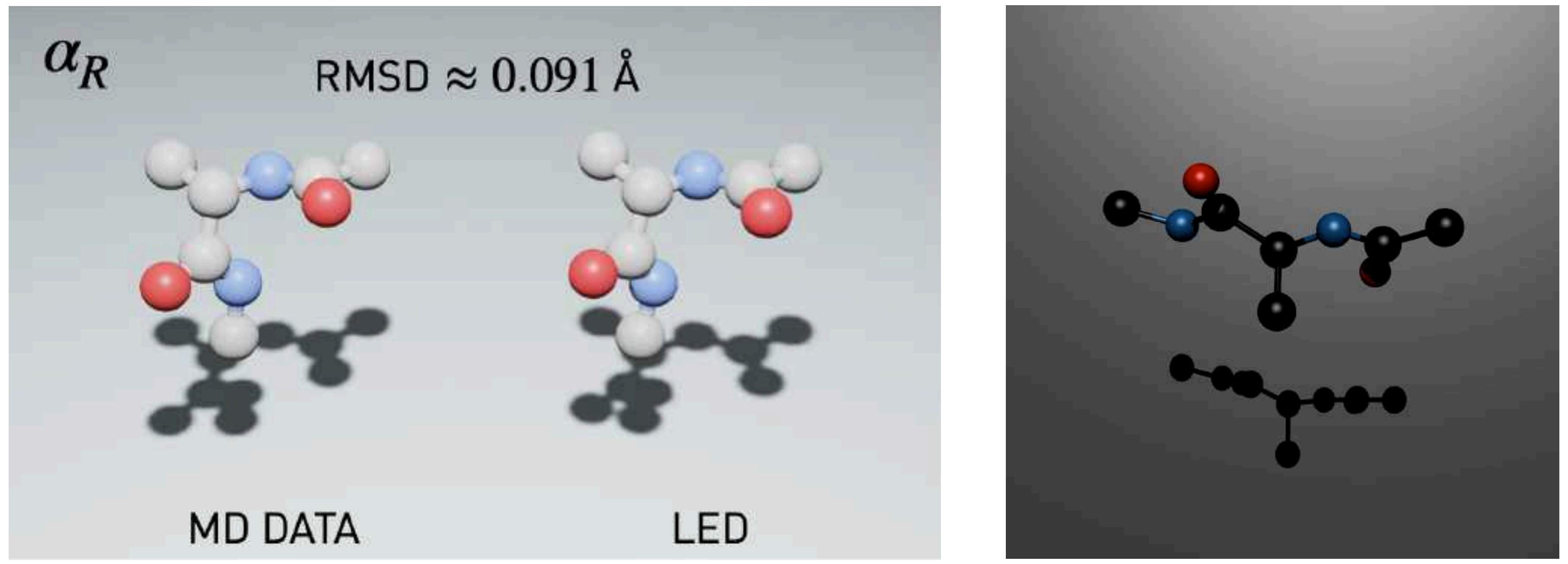
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **Journal of Neural Networks, 2020**



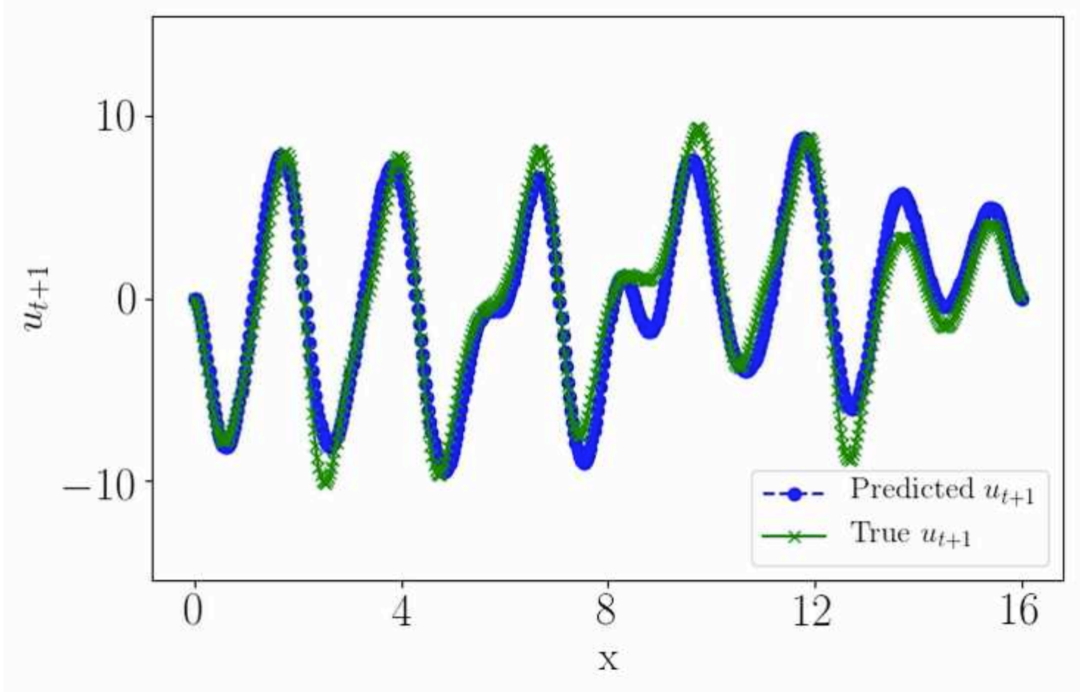


PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos, *Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,* **Nature Machine Intelligence, 2022**





PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos, *Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,* **Journal of Chemical Theory & Computation, 2021**

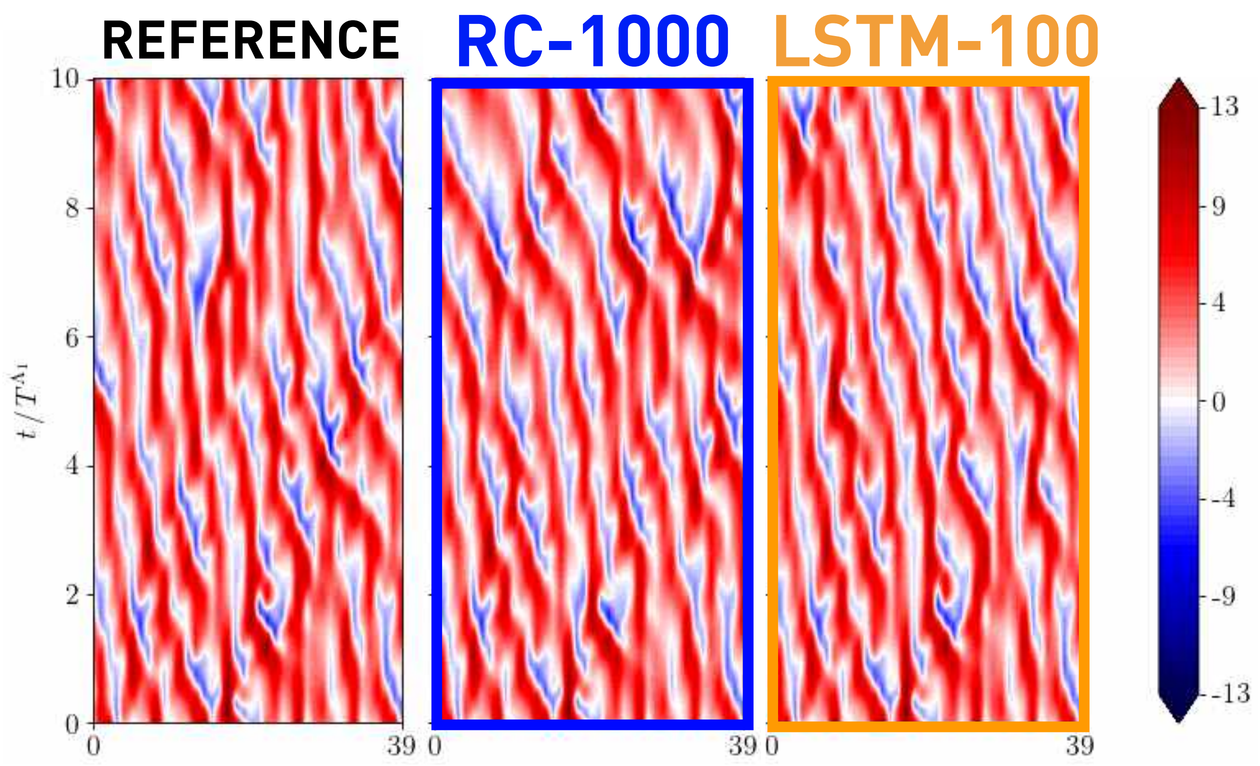


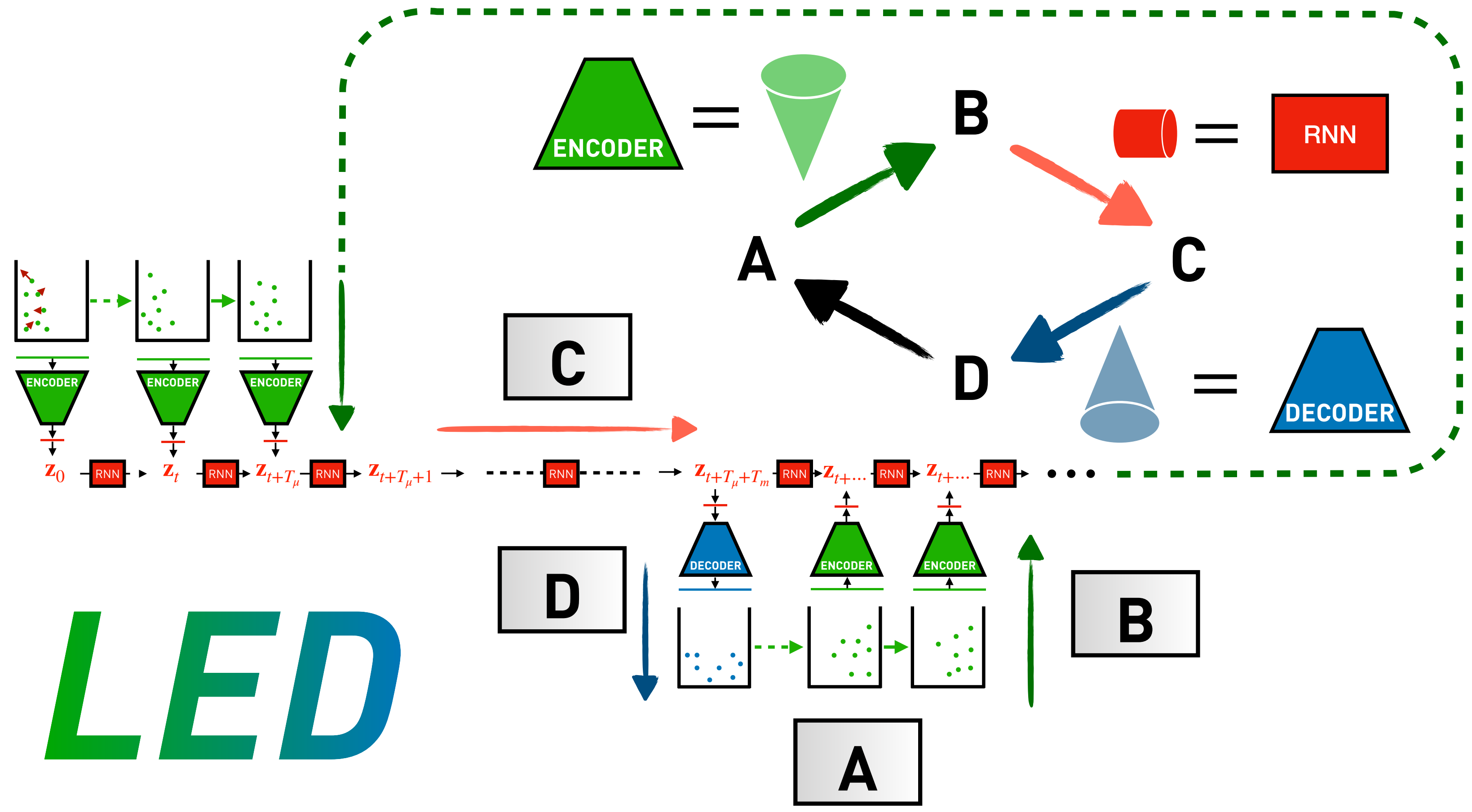
HYBRID LSTM - MSM

$$\dot{z}_t = \begin{cases} \mathrm{LSTM}^{\mathrm{W}}(z_t, z_{t-1}, z_{t-2}, \ldots) & \text{if } p_{train}(z_t) \geq \theta \\ \mathrm{MSM}^{\zeta,c}(z_t) & \text{if } p_{train}(z_t) < \theta \end{cases}$$

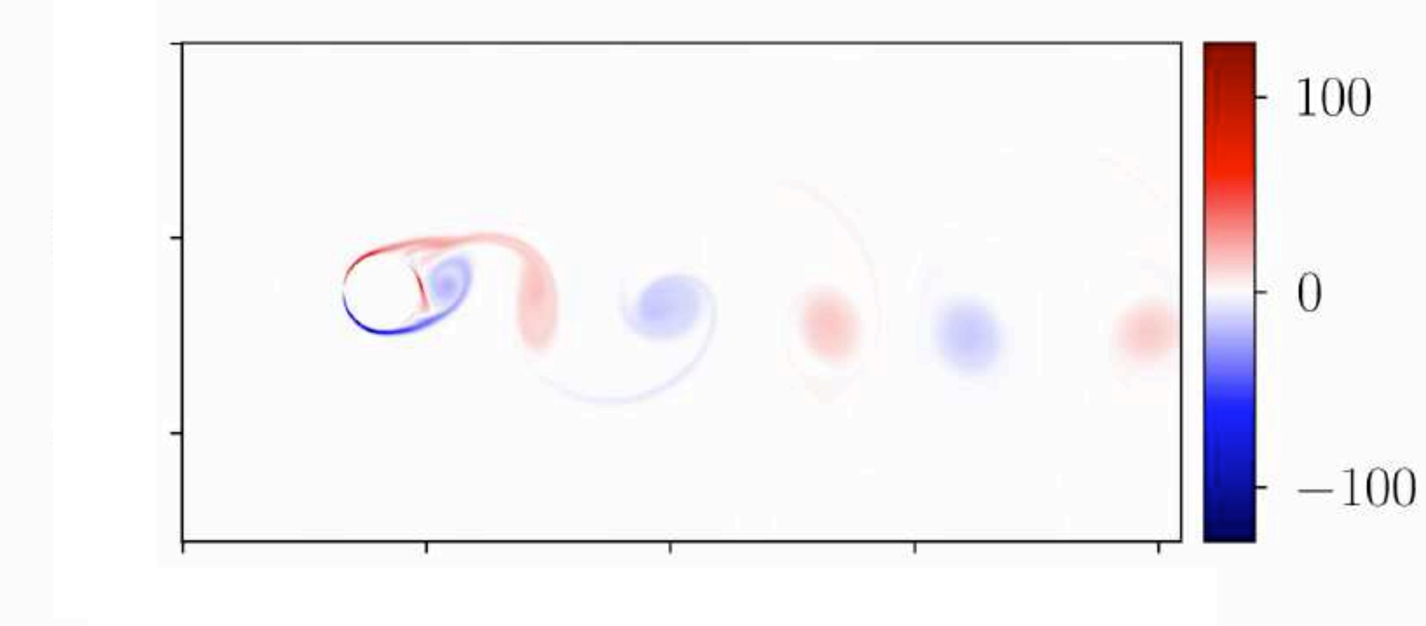
PR Vlachas, W Byeon, Z Wan, T Sapsis, P Koumoutsakos, *Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks,* **Proc. Roy. Soc. A , 2018**

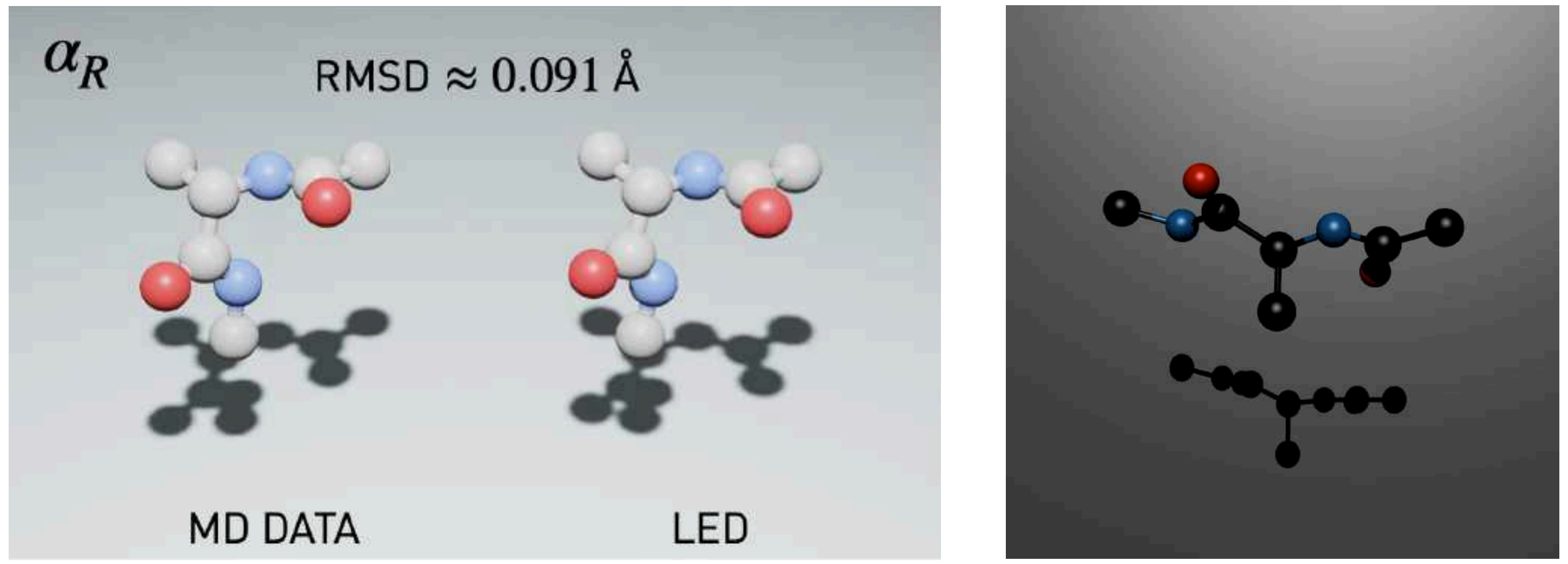
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **Journal of Neural Networks, 2020**



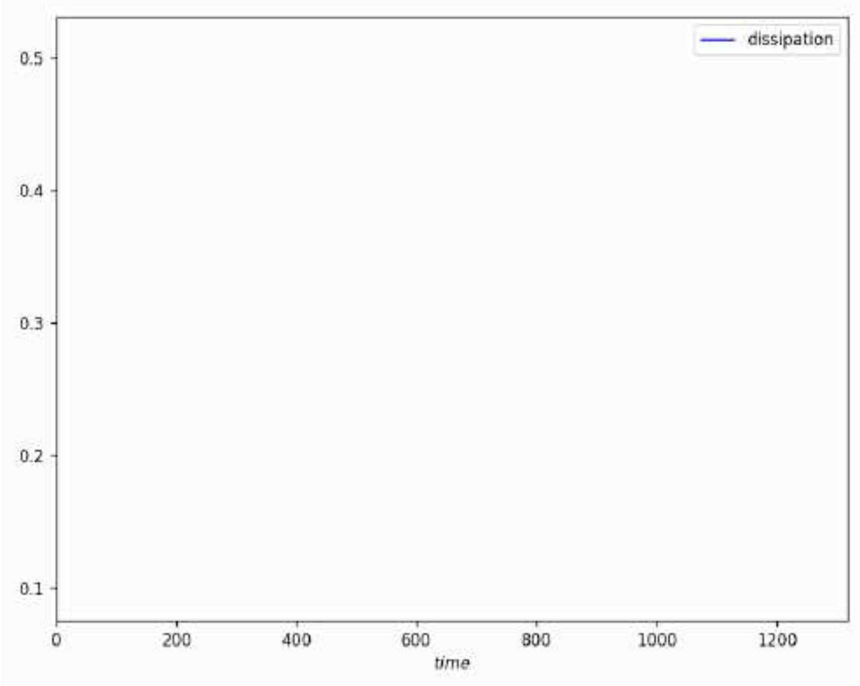


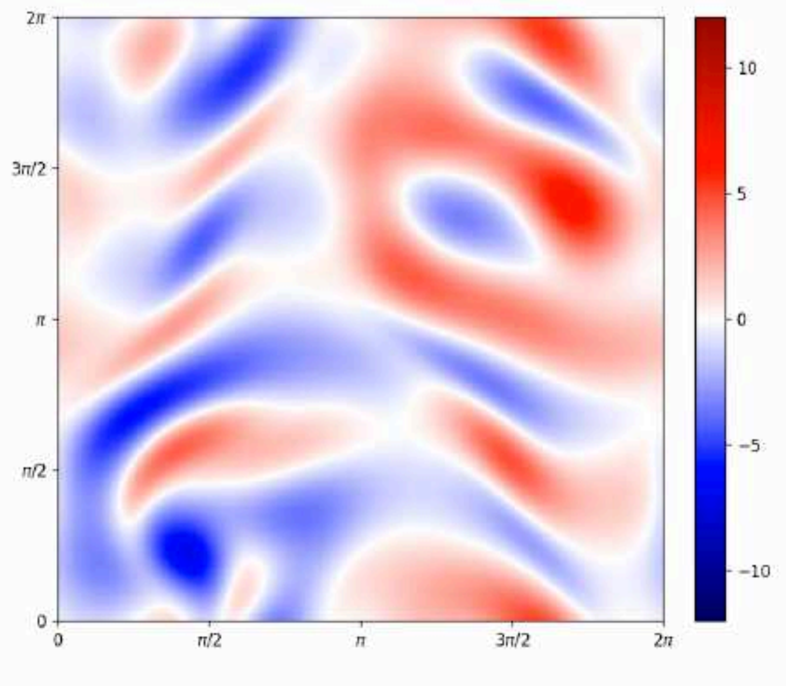
PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos, *Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,* **Nature Machine Intelligence, 2022**





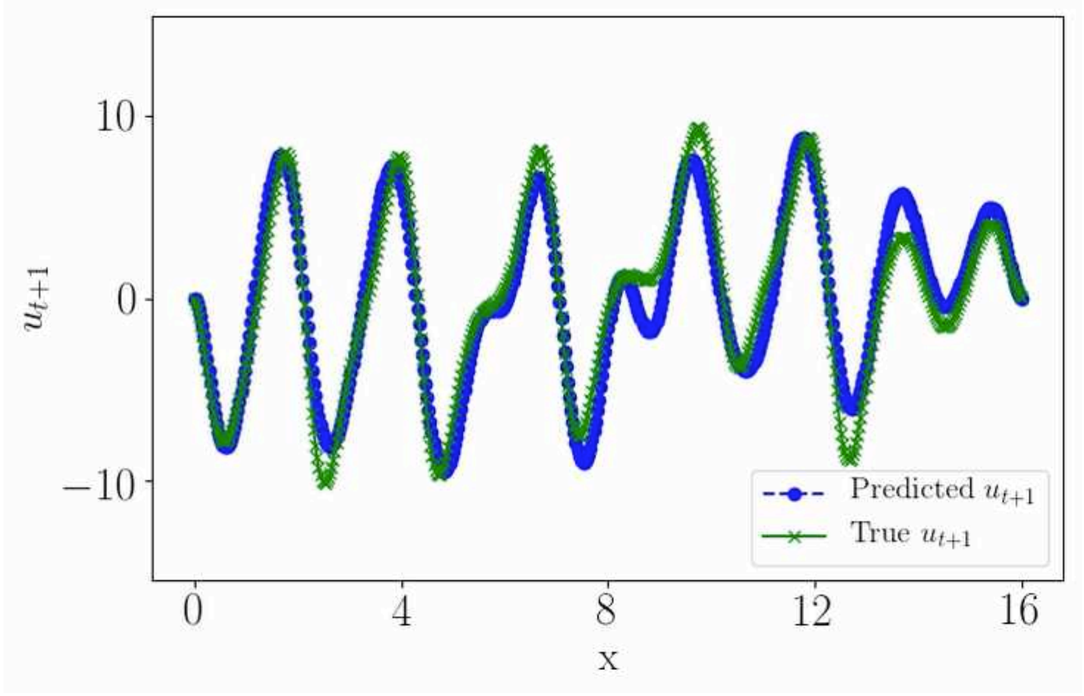
PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos, *Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,* **Journal of Chemical Theory & Computation, 2021**





ZY Wan, P Vlachas, P Koumoutsakos, T Sapsis, *Data-assisted reduced-order modeling of extreme events in complex dynamical systems,* **PloS one, 2018**

$$\dot{\xi}_t = F(\xi_t) + \tilde{G}(\xi_t, \xi_{t-1}, \xi_{t-2}, \ldots)$$

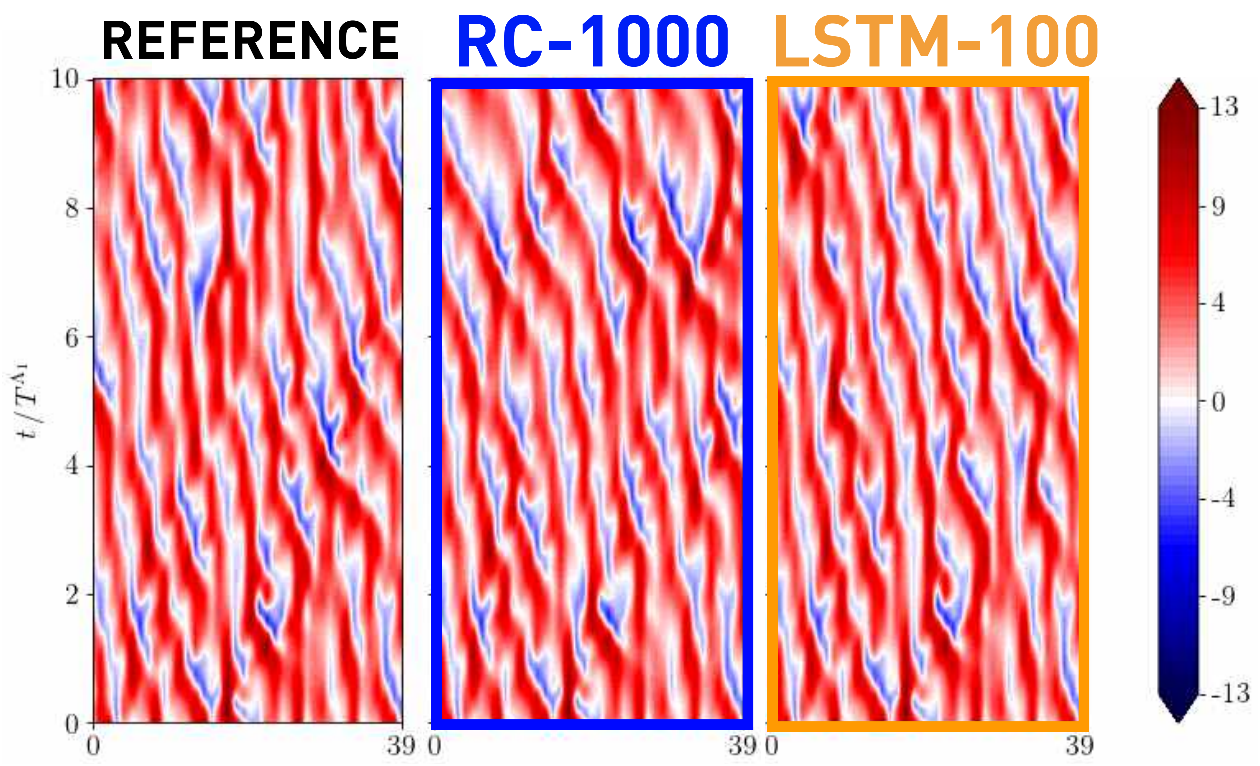


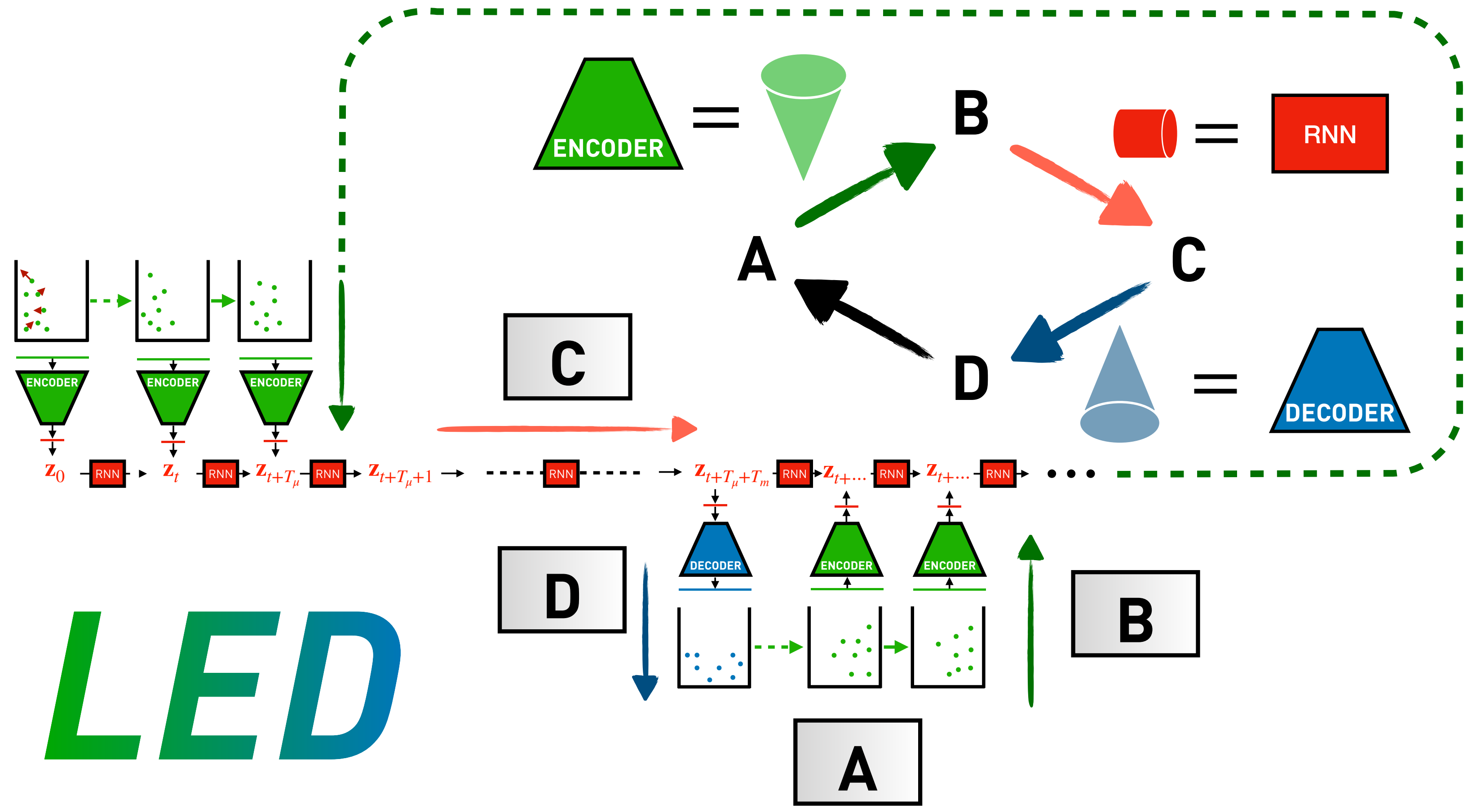
HYBRID LSTM - MSM

$$\dot{z}_t = \begin{cases} \mathrm{LSTM}^{\mathrm{W}}(z_t, z_{t-1}, z_{t-2}, \dots) & \text{if } p_{train}(z_t) \geq \theta \\ \mathrm{MSM}^{\zeta,c}(z_t) & \text{if } p_{train}(z_t) < \theta \end{cases}$$

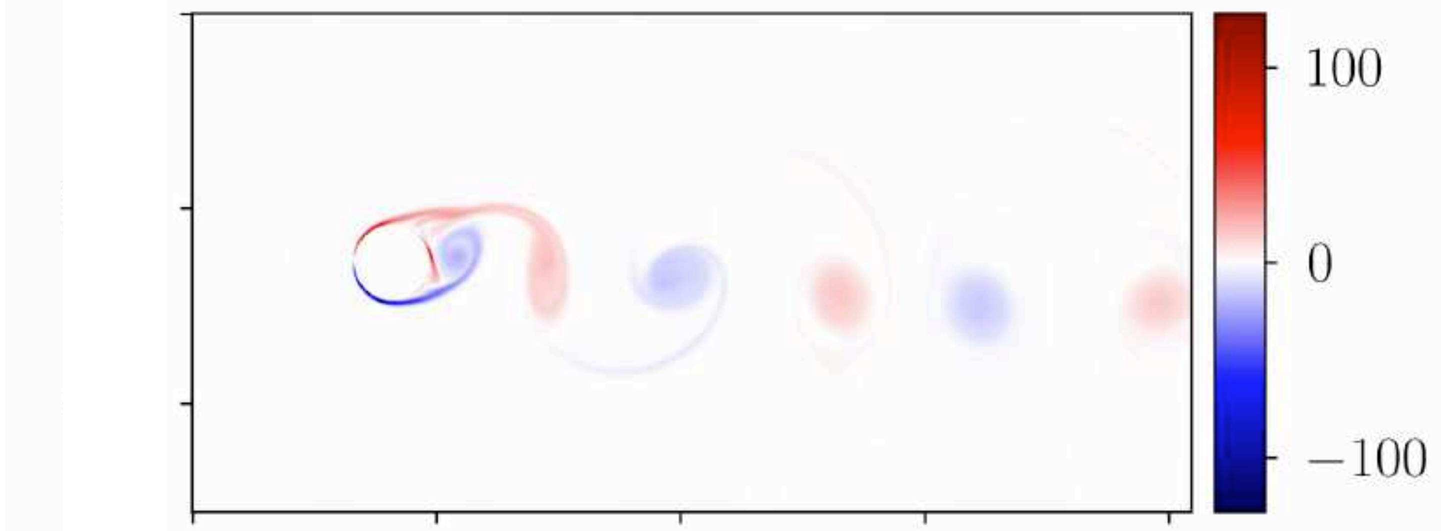
PR Vlachas, W Byeon, Z Wan, T Sapsis, P Koumoutsakos, *Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks,* **Proc. Roy. Soc. A , 2018**

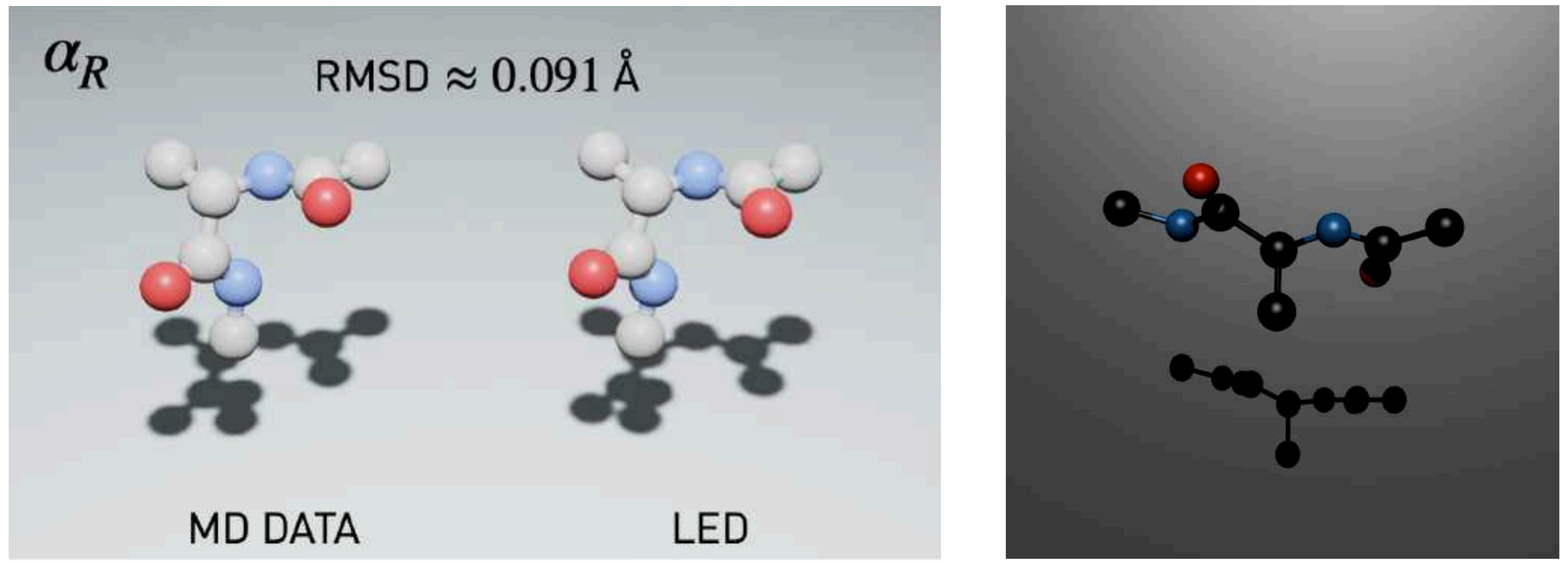
PR Vlachas , J Pathak , BR Hunt , TP Sapsis , M Girvan, E Ott and P Koumoutsakos, *Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics,* **Journal of Neural Networks, 2020**



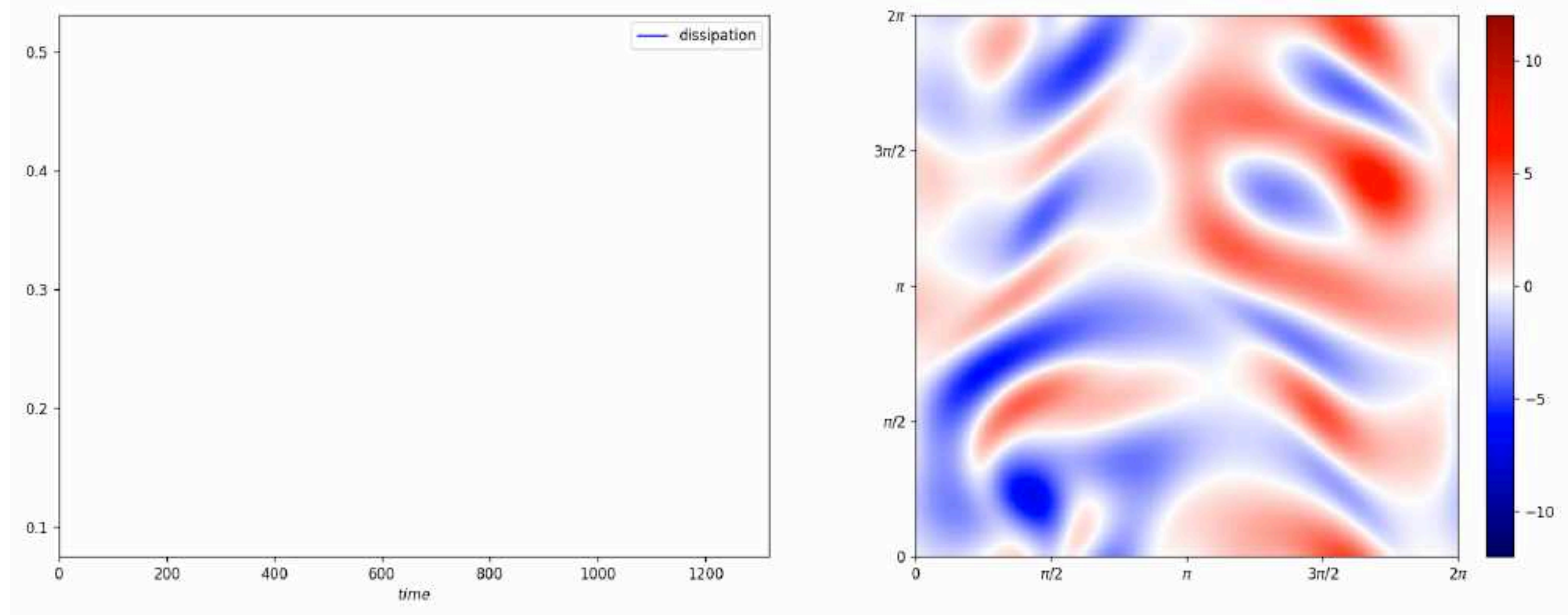


PR Vlachas, G Arampatzis, C Uhler, P Koumoutsakos, *Multiscale Simulations of Complex Systems by Learning their Effective Dynamics,* **Nature Machine Intelligence, 2022**





PR Vlachas, J Zavadlav, M Praprotnik, P Koumoutsakos, *Accelerated Simulations of Molecular Systems through Learning of their Effective Dynamics,* **Journal of Chemical Theory & Computation, 2021**



ZY Wan, P Vlachas, P Koumoutsakos, T Sapsis, *Data-assisted reduced-order modeling of extreme events in complex dynamical systems,* **PloS one, 2018**

$$\dot{\xi}_t = F(\xi_t) + \tilde{G}(\xi_t, \xi_{t-1}, \xi_{t-2}, \dots)$$

PR Vlachas, P Koumoutsakos, *Scheduled Autoregressive Backpropagation Through Time for Long-Term Forecasting,* **(in preparation)**