Structured Autoencoders for Operator-theoretic decomposition and Model reduction

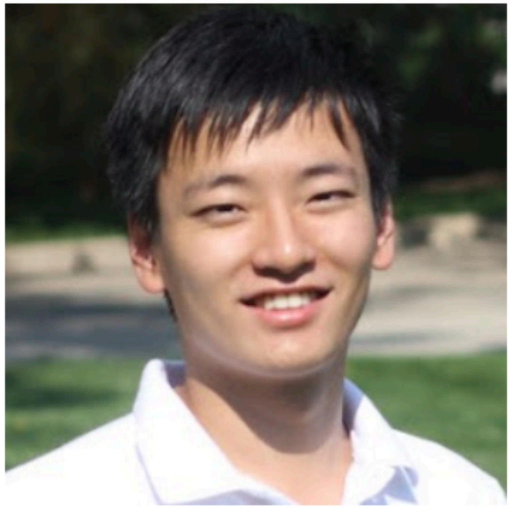
Karthik Duraisamy

UNIVERSITY of MICHIGAN

Thanks to...

Shaowu Pan

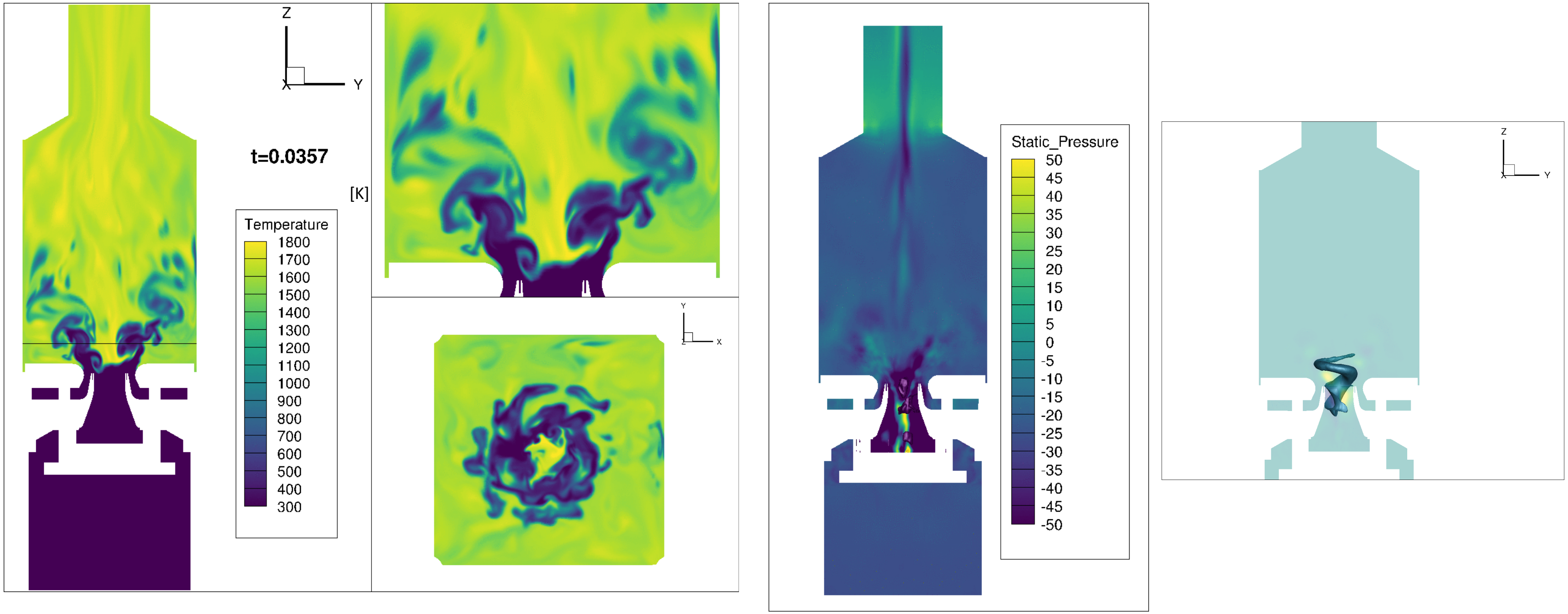
Chris Wentland



David Xu

Motivation

Decomposition & Reduced Order Modeling of Complex Multiscale Problems



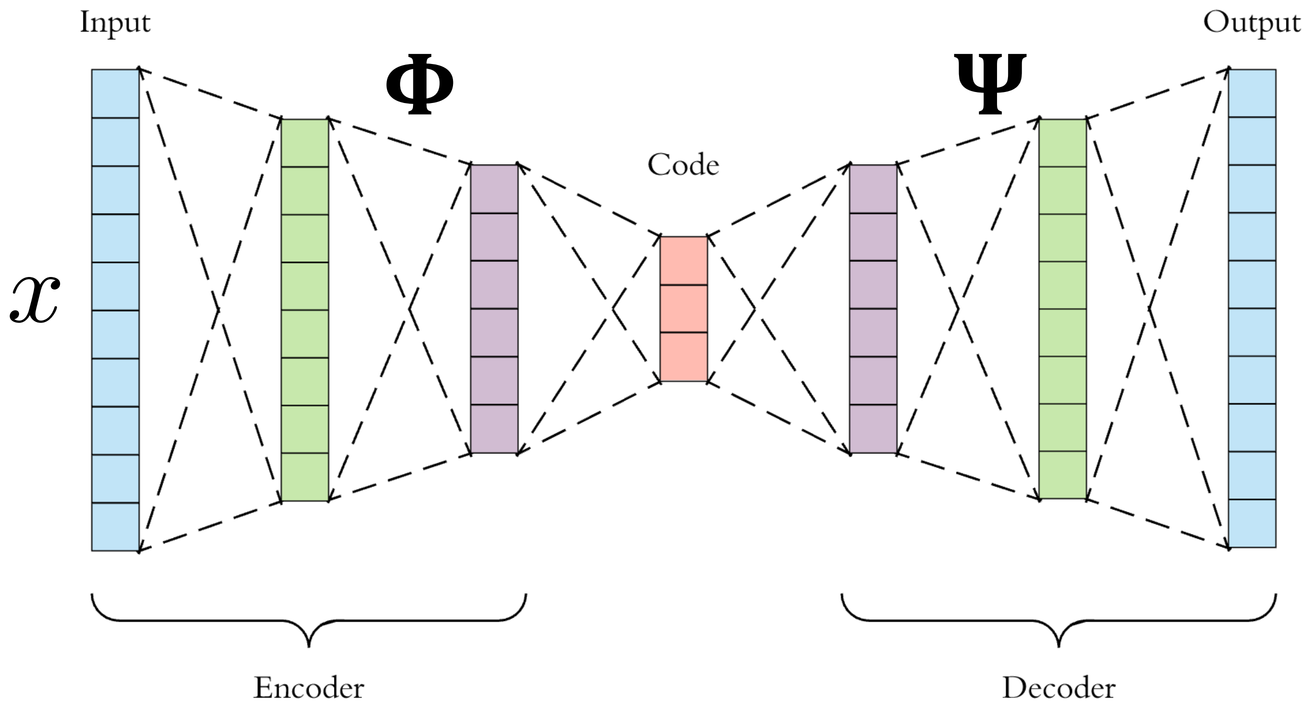
Large scale simulations O(10^6)- O(10^8) CPU hours / run

Complex physics : Flow, turbulence, combustion, heat transfer, etc

The Autoencoder

Structure: encoder (compression) + decoder (decompression)

- Encoder $\mathbf{\Phi}(\mathbf{x}; \boldsymbol{\theta}_{\mathbf{\Phi}})$
- Decoder $\mathbf{\Psi}(.\,; \boldsymbol{\theta}_{\mathbf{\Psi}})$
- POD: $\mathbf{\Phi} \rightarrow \mathbf{U}^{T}(\cdot)$, $\mathbf{\Psi} \rightarrow \mathbf{U}(\cdot)$
- Trained as one single network, $\boldsymbol{\theta}_{\mathbf{\Phi}}$ and $\boldsymbol{\theta}_{\mathbf{\Psi}}$ are optimized jointly
- Automatically separated into encoder and decoder by cutting at the "bottleneck"

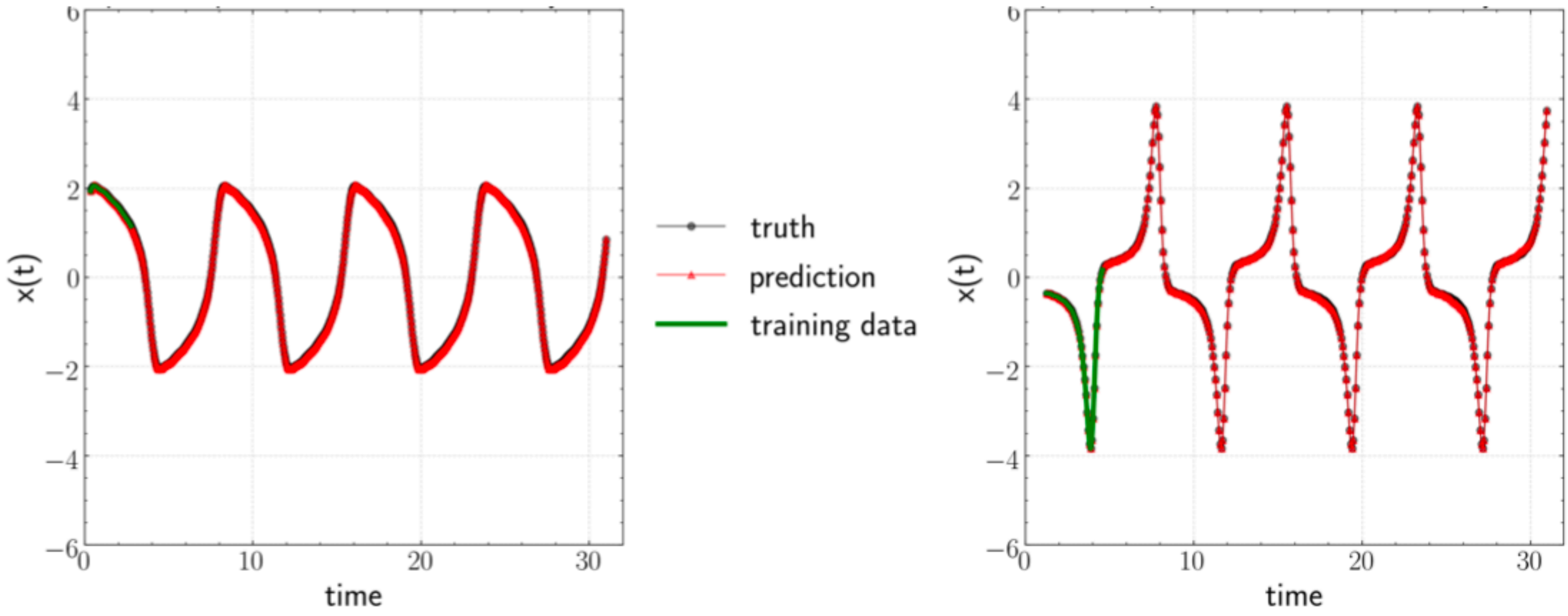


$$\tilde{x} = \Psi(\cdot, \theta_{\Psi}) \circ \Phi(x, \theta_{\Phi})$$

"Applied Deep Learning"
https://towardsdatascience.com/

Embedding (in the right coordinates)

$$\frac{d}{dt}\mathbf{x} = \mathbf{F}(\mathbf{x}(t)) \quad \longrightarrow \quad \hat{\mathbf{x}}_{j+1} = \mathbf{W}_0\mathbf{x}_j + \mathbf{W}_1\mathbf{x}_{j-1} + \ldots + \mathbf{W}_L\mathbf{x}_{j-L}$$



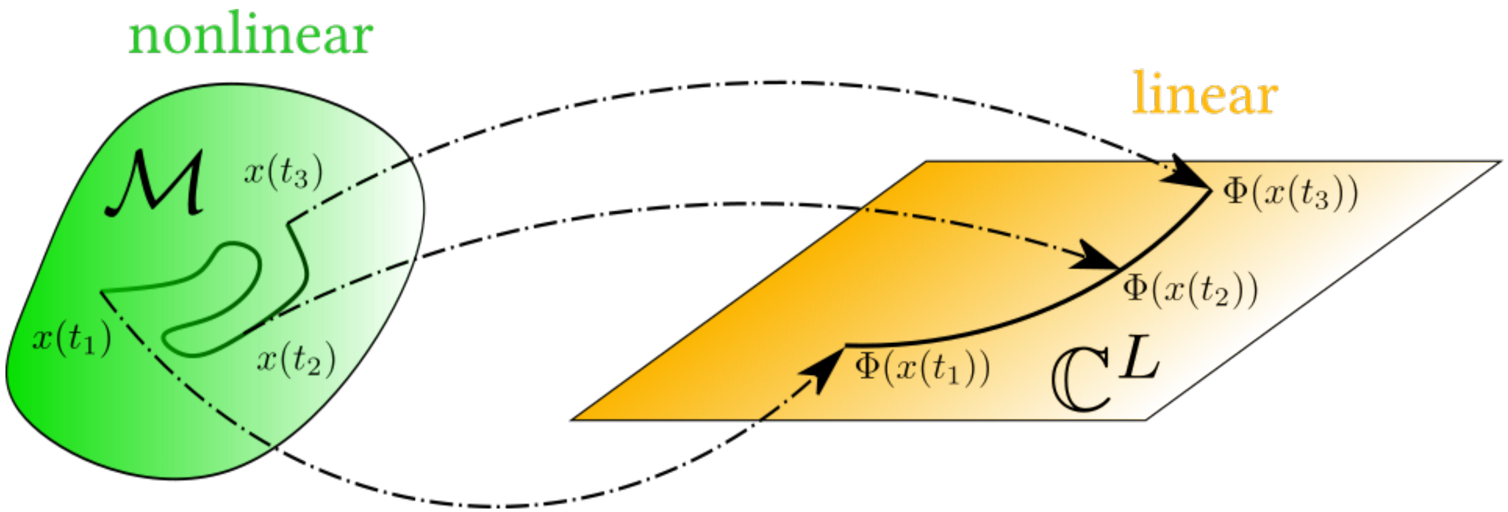
Pan & Duraisamy, *On the structure of time delay embedding in linear models of non-linear dynamical systems*, arXiv:1902.05198, 2019.

Part 1

Operator-theoretic Learning & Decomposition

Koopman operator and linear embedding

Dynamical system $\dot{x} = f(x) \quad ; \quad x \in \mathcal{M} \subset \mathbb{R}^N$
Dynamics of observables[1] $\mathcal{K}_t h = h \circ \phi_t$,
Koopman operator $\mathcal{K}_t : \mathcal{F} \mapsto \mathcal{F}$, $\mathcal{F} = L^2(\mathcal{M}, \mu)$
h is any observable in $\mathcal{F}$, ϕ_t is the flow.



Significance

$\mathcal{K}_t$ is linear $\rightarrow$ global linearization, but inherently infinite-dimensional.

[1] Koopman (1931), Mezić (2005)

Connections of Koopman to other operators

Dynamical system $\dot{x} = f(x) \quad ; \quad x \in \mathcal{M} \subset \mathbb{R}^N$

Liouville operator $$\mathcal{L} := f \cdot \nabla_x$$

Liouville PDE $$\frac{\partial u}{\partial t} = \mathcal{L} u \quad ; \quad u(\cdot, 0) = h$$

Generator $$\mathcal{K}_t h = h \circ \phi_t = e^{t\mathcal{L}} h$$

Liouville generates Koopman

Perron-Frobenius operator $$\rho(\cdot, t) = \mathcal{P}_t \circ \rho$$

Duality $$\langle h, \mathcal{P}_t \rho \rangle = \langle \mathcal{K}_t h, \rho \rangle$$

Perron-Frobenius is adjoint of Koopman

Spectral expansion of Koopman operators

$$\mathcal{K}_t h(x) \triangleq h \circ \phi_t = \sum_{j=0}^{\infty} e^{\lambda_j t} \phi_j(x) s_j + K_t^r h(x)$$

Point spectral resolution of Koopman operator

$$\mathcal{K}_t h(x) \triangleq h \circ \phi_t = \sum_{j=0}^{D} e^{\lambda_j t} \phi_j(x) s_j,$$

where

- $\mathcal{F}_D = \mathrm{span}\{\phi_1, \ldots, \phi_D\} \subset \mathcal{F}$: a finite Koopman invariant subspace that contains h
- s_j: Koopman modes, determined by projecting h onto $\mathcal{F}_D$

Applications

Dynamical systems analysis, Optimal control, modal decomposition, model reduction, etc.

Koopman operators & “Deep” Learning

Several works since 2018

Previous works	continuous /discrete	nonlinear reconstruction	continuous spectrum	uncertainty	stability
Yeung et al. [81]	discrete	✗	✗	✗	✗
Li et al. [43]	discrete	✗	✗	✗	✗
Takeishi et al. [73]	discrete	✓	✗	✗	✗
Otto and Rowley [60]	discrete	✓	✗	✗	✗
Lusch et al. [48]	discrete	✓	✓	✗	✗
Morton et al. [58]	discrete	✓	✗	✓	✗
Erichson et al. [17]	discrete	✓	✗	✗	✓
Our framework	continuous	✓	✗	✓	✓

Extracting a Koopman-invariant subspace

Goal: Extracting the Koopman operator defined on $\mathcal{F} = L^2(\mathcal{M}, \mu)$,

Observation functionals $\phi : \mathcal{M} \mapsto \mathbb{R}$, $\|\phi\|_{\mathcal{F}} \triangleq \sqrt{\int_{\mathcal{M}} |\phi|^2 d\mu} < \infty$.

$$\boxed{\mathcal{J}[\mathbf{\Phi}] = \max_{\psi \in \{\phi_1, \ldots, \phi_D\}} \min_{f \in \mathcal{F}_D} \|f - \mathcal{K}\psi\|_{\mathcal{F}}^2}$$

We are also interested in retrieving the state x

Arbabi & Mezic 2019

Enforcing structure for Learning : “Physics information”

Searching the set of Koopman eigenfunctions $\mathbf{\Phi}(x)$:

$$\mathbf{\Phi}(x) = \begin{bmatrix} \phi_1(x) & \phi_2(x) & \dots & \phi_D(x) \end{bmatrix} \in \mathcal{F}_D \subset \mathcal{F},$$

such that

$$\frac{d\mathbf{\Phi}}{dt} \triangleq \frac{dx}{dt} \cdot \nabla_x \mathbf{\Phi} = \boxed{f(x) \cdot \nabla_x \mathbf{\Phi} = \mathbf{\Phi}(x)\mathbf{K}}$$

Key observations

- if we knew the physics: $f(x)$, we can sample uniformly over the domain of interest without the need to generate trajectory
- if $\mathbf{\Phi}$ is fixed, classical KDMD is equivalent to the Galerkin method with Lebesgue measure over the domain of interest
- Need $\mathbf{\Psi} : \mathbb{R}^D \mapsto \mathcal{M}$, such that $\mathbf{\Psi} \circ \mathbf{\Phi} = \mathcal{I}$, Therefore, we can recover the state x from Φ.

Enforcing structure for Learning : Tractable optimization

$$\mathcal{J}[\mathbf{\Phi}] = \max_{\psi \in \{\phi_1, \ldots, \phi_D\}} \min_{f \in \mathcal{F}_D} \|f - \mathcal{K}\psi\|_{\mathcal{F}}^2$$

$$\downarrow$$

$$\mathbf{\Phi}^* = \operatorname*{argmin}_{\substack{\mathbf{\Phi} \in \mathcal{F}^D, \exists \mathbf{\Psi}: \mathbb{R}^D \mapsto \mathcal{M} \\ \mathbf{\Psi} \circ \mathbf{\Phi} = \mathcal{I}}} \mathcal{J}[\mathbf{\Phi}]$$

$$\downarrow$$

$$\mathbf{W}_{\mathbf{\Phi}}^* = \operatorname*{argmin}_{\substack{\mathbf{W}_{\mathbf{\Phi}}, \exists \mathbf{\Psi} \in C(\mathbb{R}^D, \mathbb{R}^N) \\ \mathbf{\Psi} \circ \mathbf{\Phi}(\cdot; \mathbf{W}_{\mathbf{\Phi}}) = \mathcal{I}}} \mathcal{J}[\mathbf{\Phi}(\cdot; \mathbf{W}_{\mathbf{\Phi}})]$$

$$\downarrow$$

$$\boxed{\widehat{\mathbf{W}}_{\mathbf{\Phi}}, \widehat{\mathbf{W}}_{\mathbf{\Psi}}, \widehat{\mathbf{K}} = \operatorname*{argmin}_{\mathbf{W}_{\mathbf{\Phi}}, \mathbf{W}_{\mathbf{\Psi}}, \mathbf{K}} \widetilde{\mathcal{J}}[\mathbf{\Phi}(\cdot; \mathbf{W}_{\mathbf{\Phi}}), \mathbf{K}] + \mathcal{R}[\mathbf{\Phi}(\cdot; \mathbf{W}_{\mathbf{\Phi}}), \mathbf{\Psi}(\cdot; \mathbf{W}_{\mathbf{\Psi}})]}$$

$$\widetilde{\mathcal{J}}[\mathbf{\Phi}, \mathbf{K}] = \|\mathbf{\Phi}\mathbf{K} - \mathcal{K}\mathbf{\Phi}\|_{\mathcal{F}^D}^2 \qquad \mathcal{R}[\mathbf{\Phi}, \mathbf{\Psi}] = \|\mathbf{\Psi} \circ \mathbf{\Phi} - \mathcal{I}\|_{\mathcal{F}^N}^2$$

"Data-free", "Physics-informed"

$$\widehat{\mathbf{W}}_{\boldsymbol{\Phi}}, \widehat{\mathbf{W}}_{\boldsymbol{\Psi}}, \widehat{\mathbf{K}} = \underset{\mathbf{W}_{\boldsymbol{\Phi}}, \mathbf{W}_{\boldsymbol{\Psi}}, \mathbf{K}}{\operatorname{argmin}} \widetilde{\mathcal{J}}[\boldsymbol{\Phi}(\cdot; \mathbf{W}_{\boldsymbol{\Phi}}), \mathbf{K}] + \mathcal{R}[\boldsymbol{\Phi}(\cdot; \mathbf{W}_{\boldsymbol{\Phi}}), \boldsymbol{\Psi}(\cdot; \mathbf{W}_{\boldsymbol{\Psi}})]$$

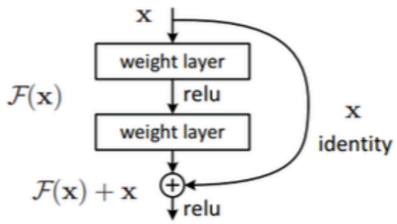
$$\widetilde{\mathcal{J}}[\boldsymbol{\Phi}, \mathbf{K}] = \|\boldsymbol{\Phi}\mathbf{K} - \mathcal{K}\boldsymbol{\Phi}\|^2_{\mathcal{F}^D} \qquad \mathcal{R}[\boldsymbol{\Phi}, \boldsymbol{\Psi}] = \|\boldsymbol{\Psi} \circ \boldsymbol{\Phi} - \mathcal{I}\|^2_{\mathcal{F}^N}$$

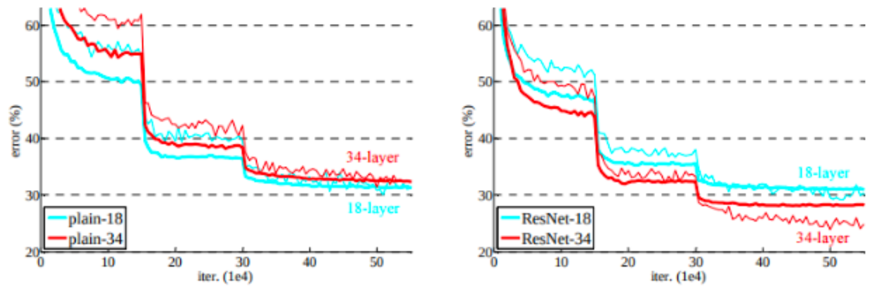
Trajectory data, "Unknown physics"

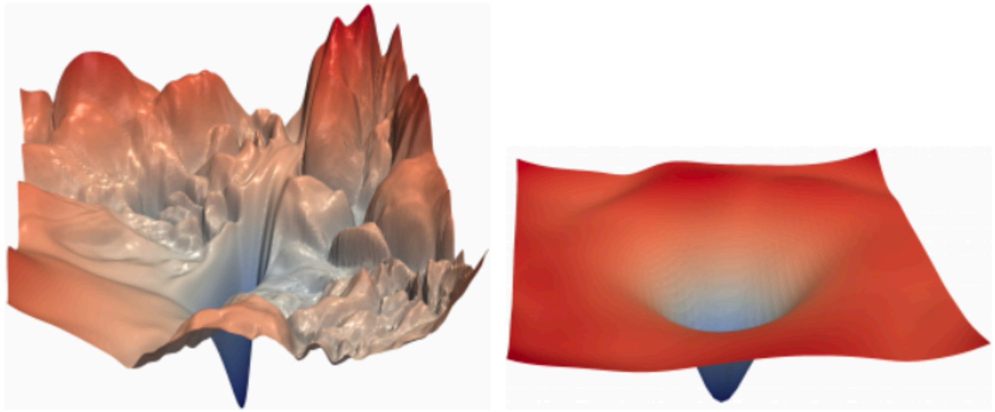
$$\widehat{\mathbf{W}}_{\boldsymbol{\Phi}}, \widehat{\mathbf{W}}_{\boldsymbol{\Psi}}, \widehat{\mathbf{K}} = \underset{\mathbf{W}_{\boldsymbol{\Phi}}, \mathbf{W}_{\boldsymbol{\Psi}}, \mathbf{K}}{\operatorname{argmin}} \widetilde{\mathcal{J}}_r[\boldsymbol{\Phi}(\cdot; \mathbf{W}_{\boldsymbol{\Phi}}), \mathbf{K}] + \widehat{\mathcal{P}}[\boldsymbol{\Phi}(\cdot; \mathbf{W}_{\boldsymbol{\Phi}}), \boldsymbol{\Psi}(\cdot; \mathbf{W}_{\boldsymbol{\Psi}}), \mathbf{K}]$$

$$\widetilde{\mathcal{J}}_r[\boldsymbol{\Phi}, \mathbf{K}] = \left\|\boldsymbol{\Phi} e^{t\mathbf{K}} - \mathcal{K}_t \boldsymbol{\Phi}\right\|^2_{\mathcal{G}^D} \qquad \widetilde{\mathcal{P}}[\boldsymbol{\Phi}, \boldsymbol{\Psi}, \mathbf{K}] = \left\|\boldsymbol{\Psi} \circ \boldsymbol{\Phi} e^{t\mathbf{K}} - \mathcal{K}_t \mathcal{I}\right\|^2_{\mathcal{G}^N}$$

Enforcing structure for Learning : "DMD ResNet"







(a) without skip connections

(b) with skip connections

Figure 1: The loss surfaces of ResNet-56 with/without skip connections. The proposed filter normalization scheme is used to enable comparisons of sharpness/flatness between the two figures.

$$\mathbf{\Phi}_{svd}(\mathbf{z}) = \mathbf{z}\mathbf{\Lambda}\mathbf{V}_D, \quad \mathbf{\Psi}_{svd}(\mathbf{\Phi}) \triangleq \mathbf{\Phi}\mathbf{V}_D^\top\mathbf{\Lambda}^{-1},$$

$$\mathbf{\Phi}(\mathbf{z}) = \underbrace{\mathbf{\Phi}_{nn}(\mathbf{z})W_{enc,L} + b_{enc,L}}_{\text{nonlinear observables}} + \underbrace{\mathbf{\Phi}_{svd}(\mathbf{z})W_{enc,L}}_{\text{linear observables}},$$

$$\mathbf{\Psi}(\mathbf{\Phi}) = \underbrace{\mathbf{\Psi}_{nn}(\mathbf{\Phi})}_{\text{nonlinear reconstruction}} + \underbrace{\mathbf{\Psi}_{svd}(\mathbf{\Phi}W_{dec,1})}_{\text{linear reconstruction}},$$

Enforcing structure for Learning : Stability

We propose : (where $\zeta_1, \ldots, \zeta_{D-1}, \sigma_1, \ldots, \sigma_D \in \mathbb{R}$.)

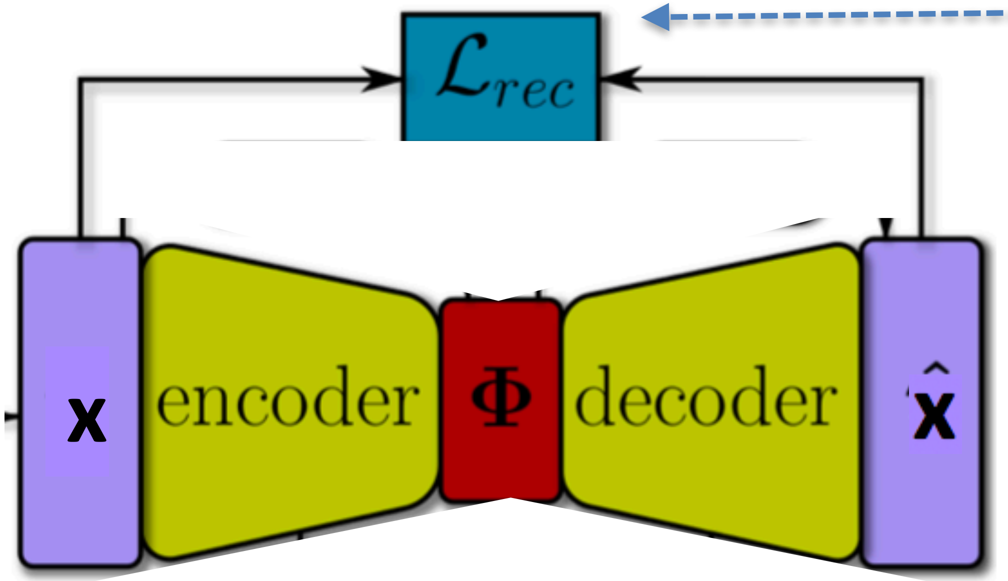
$$\mathbf{K}_{stable} = \begin{bmatrix} -\sigma_1^2 & \zeta_1 & & \\ -\zeta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \zeta_{D-1} \\ & & -\zeta_{D-1} & -\sigma_D^2 \end{bmatrix}, \tag{5}$$

Theorem

$\forall D \in \mathbb{N}$, for any real square diagonalizable matrix $\mathbf{K} \in \mathbb{R}^{D \times D}$ that only has non-positive real part of the eigenvalues $D \geq 2$, there exists a set of $\zeta_1, \ldots, \zeta_{D-1}, \sigma_1, \ldots, \sigma_D \in \mathbb{R}$ such that $\mathbf{K}_{stable}$ is similar to $\mathbf{K}$ over $\mathbb{R}$. Moreover, for any $\zeta_1, \ldots, \zeta_{D-1}, \sigma_1, \ldots, \sigma_D \in \mathbb{R}$, the real part of the eigenvalue of $\mathbf{K}_{stable}$ is non-positive.

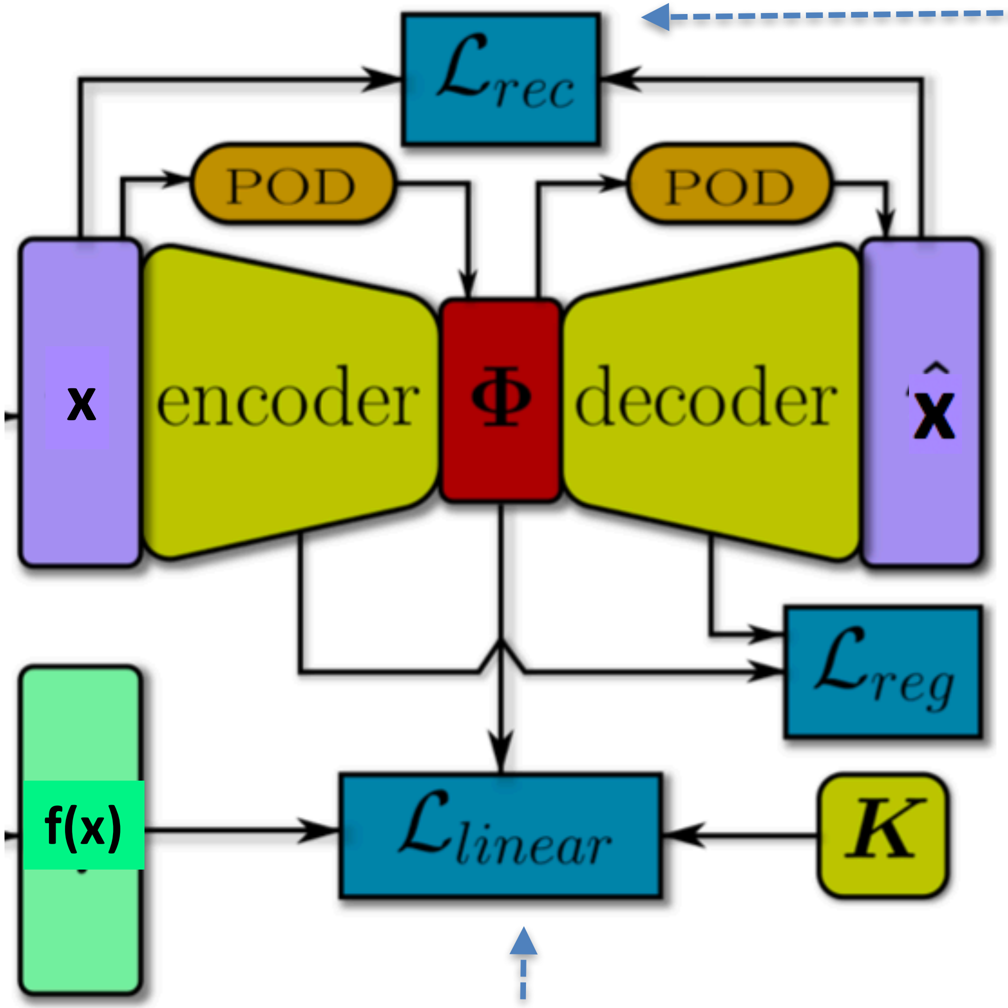
Unconditionally stable, and “expressive” $\rightarrow$ any diagonalizable matrix corresponding to a stable Koopman operator can be represented without loss of information

Naïve "Autoencoders"



$$\frac{1}{M}\sum_{i=1}^{M}\|\boldsymbol{\Psi}(\boldsymbol{\Phi}(\mathbf{x}_i)) - \mathbf{x}_i\|^2$$

Putting it all together (deterministic form)



$$\frac{1}{M}\sum_{i=1}^{M}\|\boldsymbol{\Psi}(\boldsymbol{\Phi}(\mathbf{x}_i)) - \mathbf{x}_i\|^2$$

$$\boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{\Phi}_{dmd}(\boldsymbol{x}) + \boldsymbol{\Phi}_{nn}(\boldsymbol{x}),$$
$$\boldsymbol{\Psi}(\boldsymbol{\Phi}) = \boldsymbol{\Psi}_{dmd}(\boldsymbol{\Phi}(\boldsymbol{x})) + \boldsymbol{\Psi}_{nn}(\boldsymbol{\Phi}(\boldsymbol{x})).$$

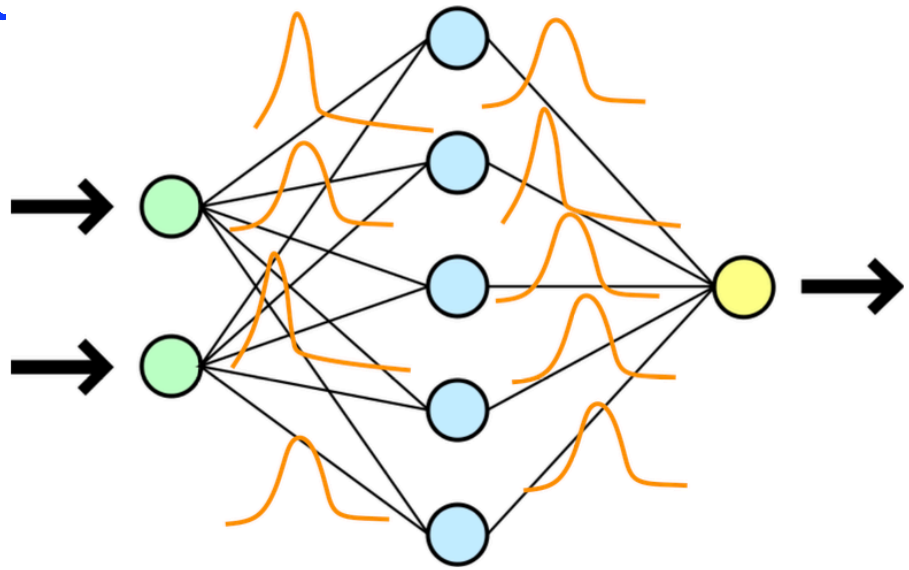
$$\frac{1}{M}\sum_{i=1}^{M}\|\mathbf{f}(\mathbf{x}_i)\cdot\nabla_{\mathbf{x}}\boldsymbol{\Phi}(\mathbf{x}_i) - \boldsymbol{\Phi}(\mathbf{x}_i)\mathbf{K}\|^2$$

Pan. S. & Duraisamy, K., *Physics-Informed Probabilistic Learning of Linear Embeddings of Non-linear Dynamics With Guaranteed Stability*, SIAM J. of Applied Dynamical Systems, 2020.

Bayesian Neural Networks & Variational Inference

TensorFlow

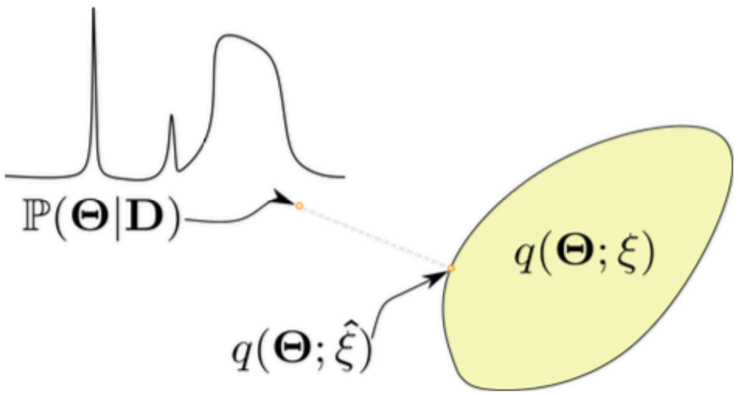
Edward
Python library for probabilistic modeling, inference, and criticism

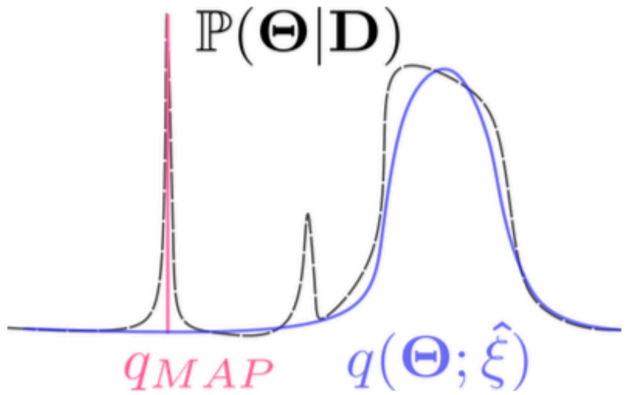


- a faster alternative to MCMC for Bayesian inference
- traditionally requires tedious model-specific derivations and implementation
  - Automatic Differentiation Variational Inference (ADVI)[3] leverages automatic differentiation (AD) to makes implementation of VI easier
- we build our framework based on Tensorflow + ADVI functionality in Edward[4]

Variational Inference

- minimize the KL divergence: $\min_\xi \mathbb{KL}(q(\boldsymbol{\Theta};\xi)\|\mathbb{P}(\boldsymbol{\Theta}|\mathbf{D}))$





- equivalently maximize the evidence lower bound (ELBO)

$$\hat{\xi} = \arg\max_{\xi} \mathcal{L}(\xi) = \arg\max_{\xi} \left(\mathbb{E}_q[\log \mathbb{P}(\boldsymbol{\Theta},\mathbf{D})] - \mathbb{E}_q[\log q(\boldsymbol{\Theta};\xi)]\right)$$

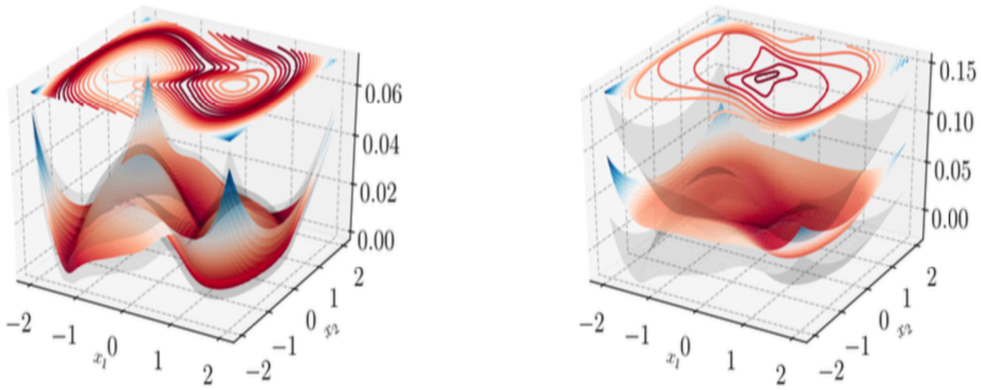
by re-parameterization-trick + stochastic gradient descent

$$\boldsymbol{\Theta} \sim \mathcal{N}(\mathbf{0},\boldsymbol{\Lambda})$$

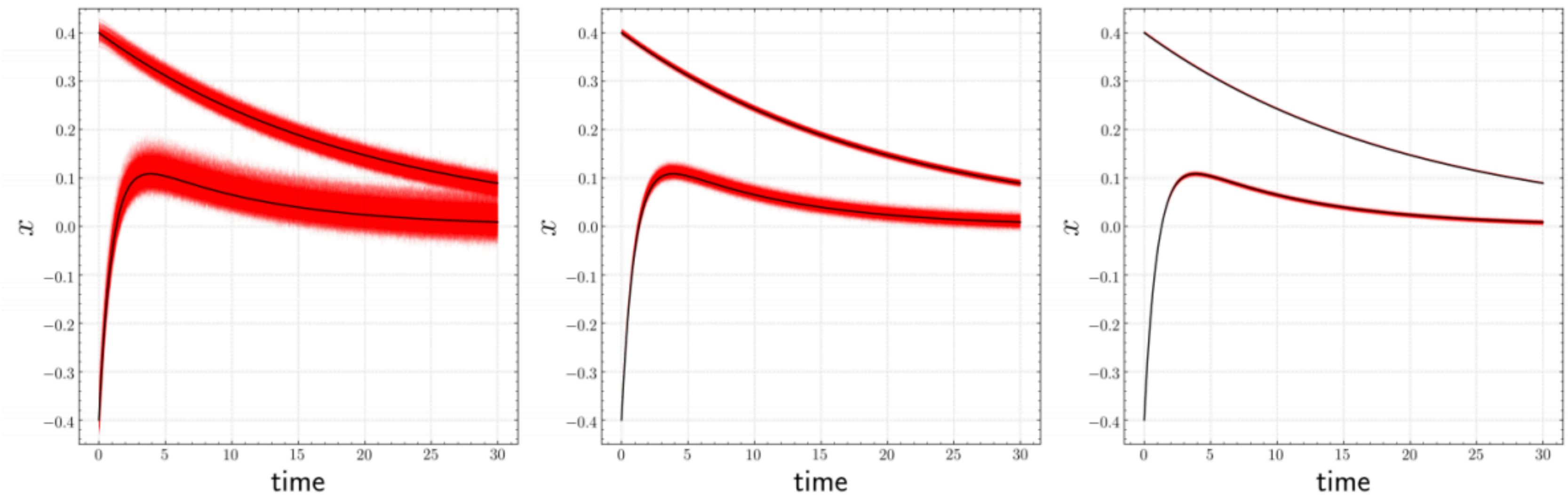
$$\mathbf{D}|\boldsymbol{\Theta} \sim \prod_{i=1}^{M} \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\Psi}(\boldsymbol{\Phi}(\mathbf{x}_i)) - \mathbf{x}_i \\ \mathbf{f}(\mathbf{x_i}) \cdot \nabla_{\mathbf{x}}\boldsymbol{\Phi}(\mathbf{x}_i) - \boldsymbol{\Phi}(\mathbf{x}_i)\mathbf{K} \end{bmatrix}; \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \sigma_{rec}^2\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \sigma_{lin}^2\mathbf{I} \end{bmatrix}\right)$$

Verification on Model dynamical systems

Duffing oscillator: Eigenfunctions (with uncertainty)

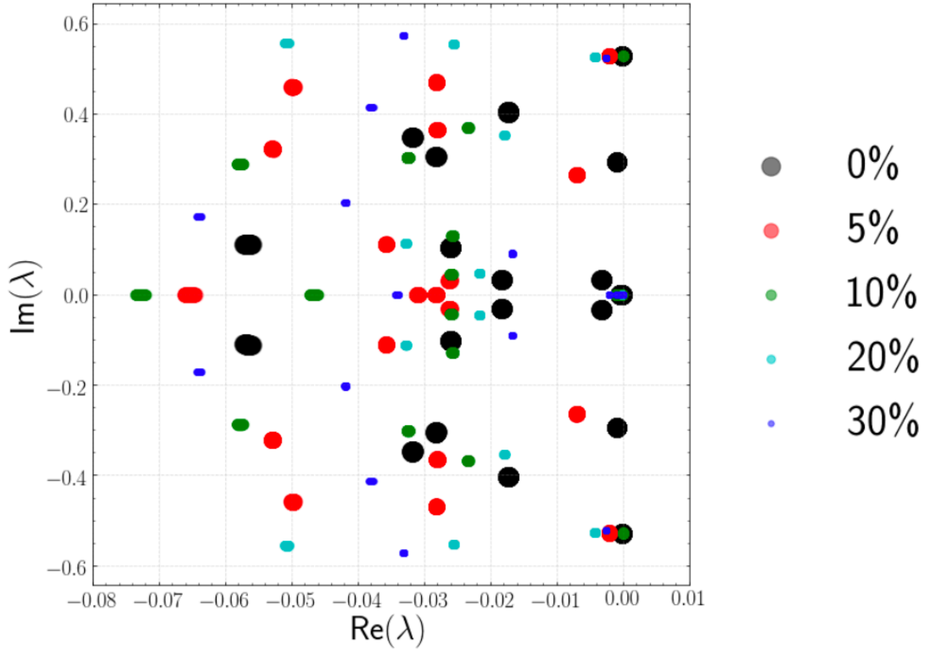


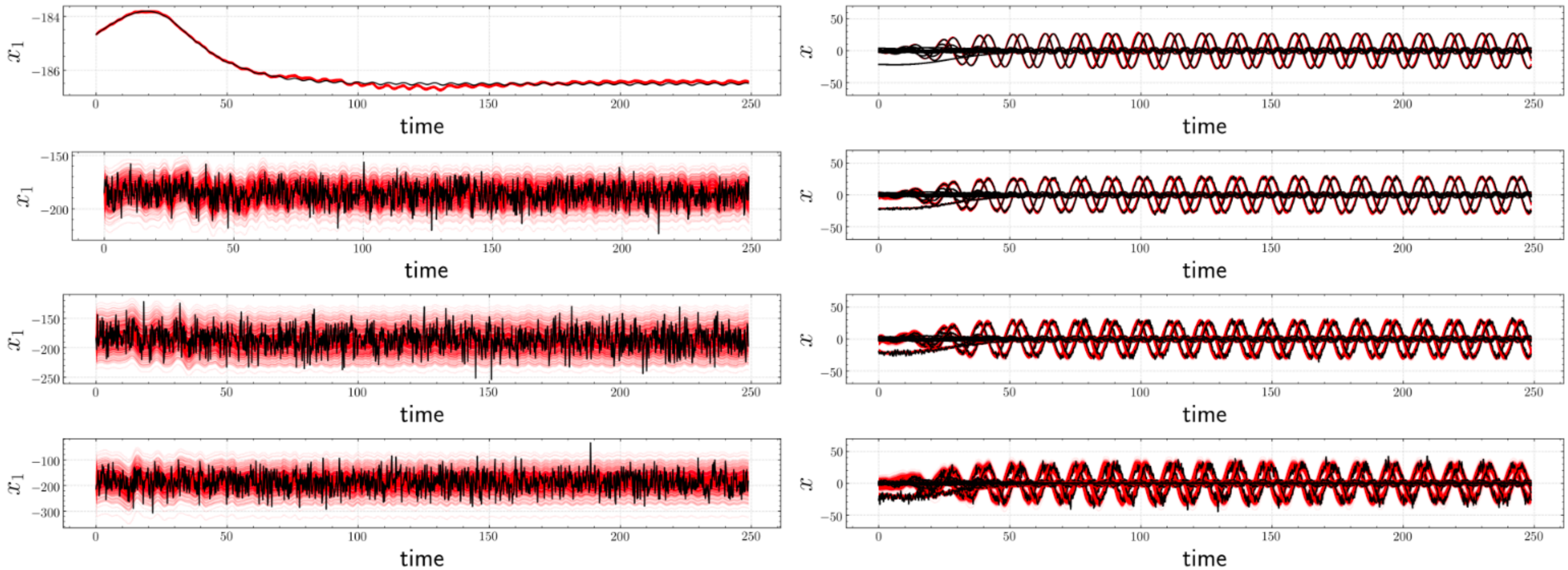
Prediction and sensitivity to data



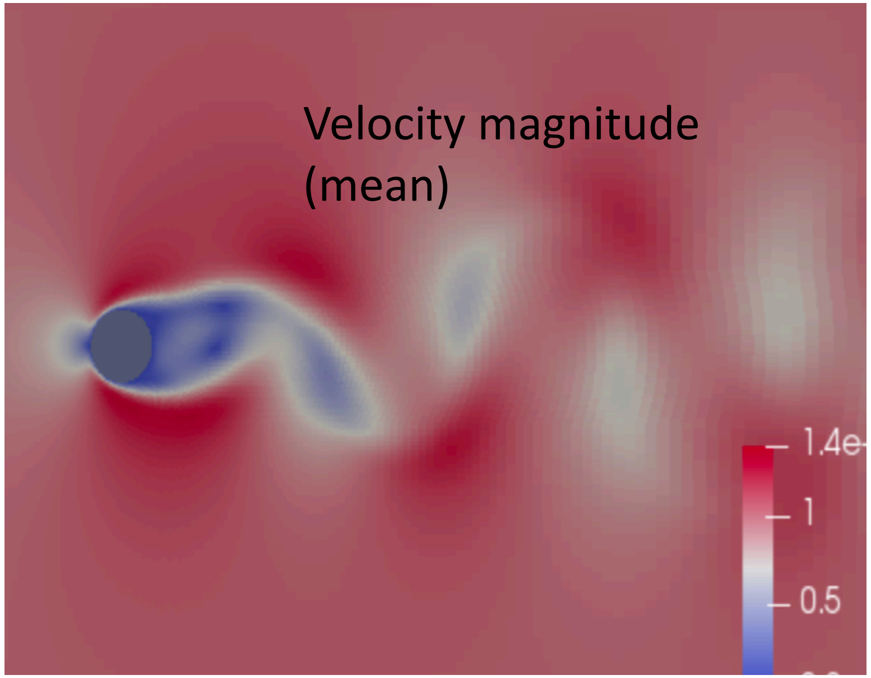
Flow over cylinder: Prediction with uncertainties

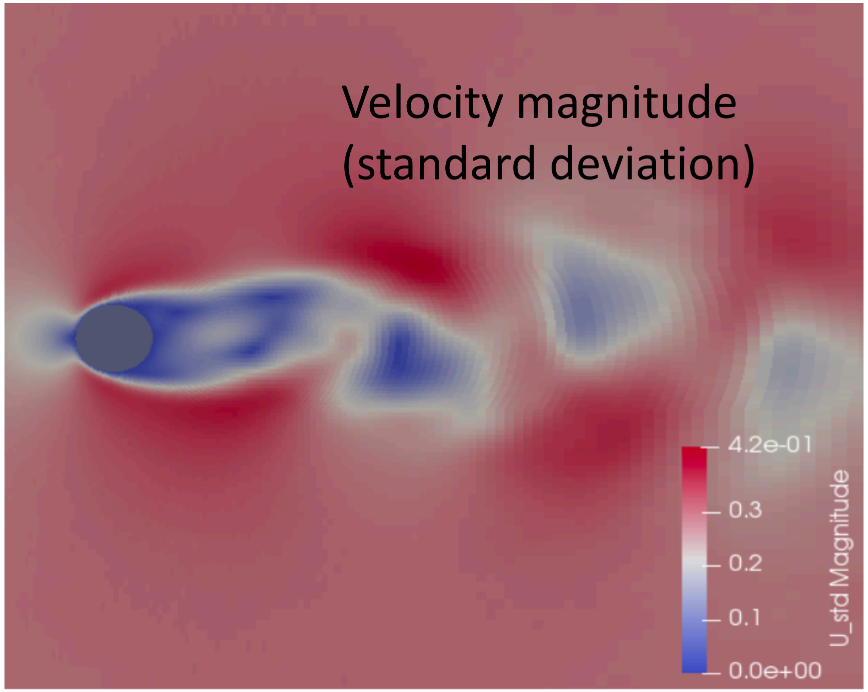
- Gaussian white noise added





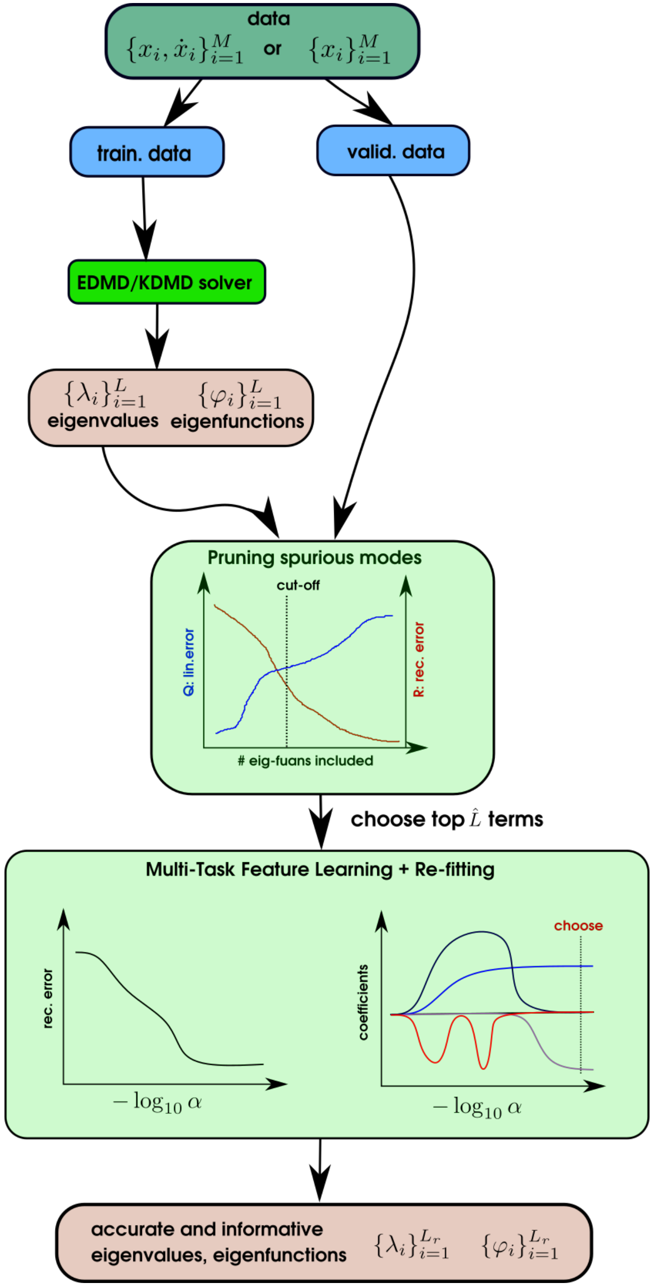
Flow over cylinder: Prediction with uncertainties





Physics-Informed Probabilistic Learning of Linear Embeddings of Non-linear Dynamics With Guaranteed Stability, Pan, S., and Duraisamy, K., SIADS, 2020

Multi-task learning framework to extract sparse Koopman-invariant subspaces

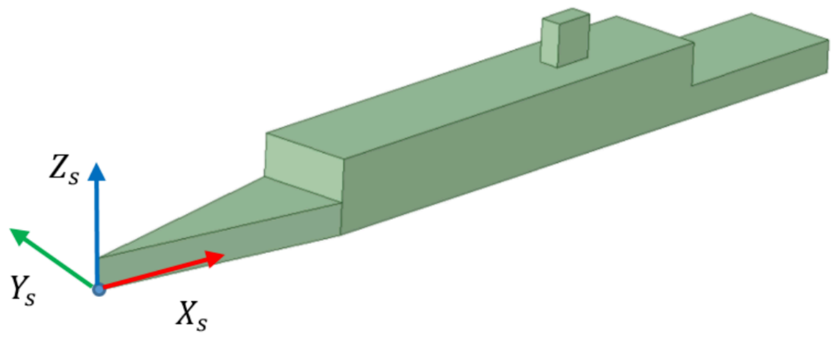


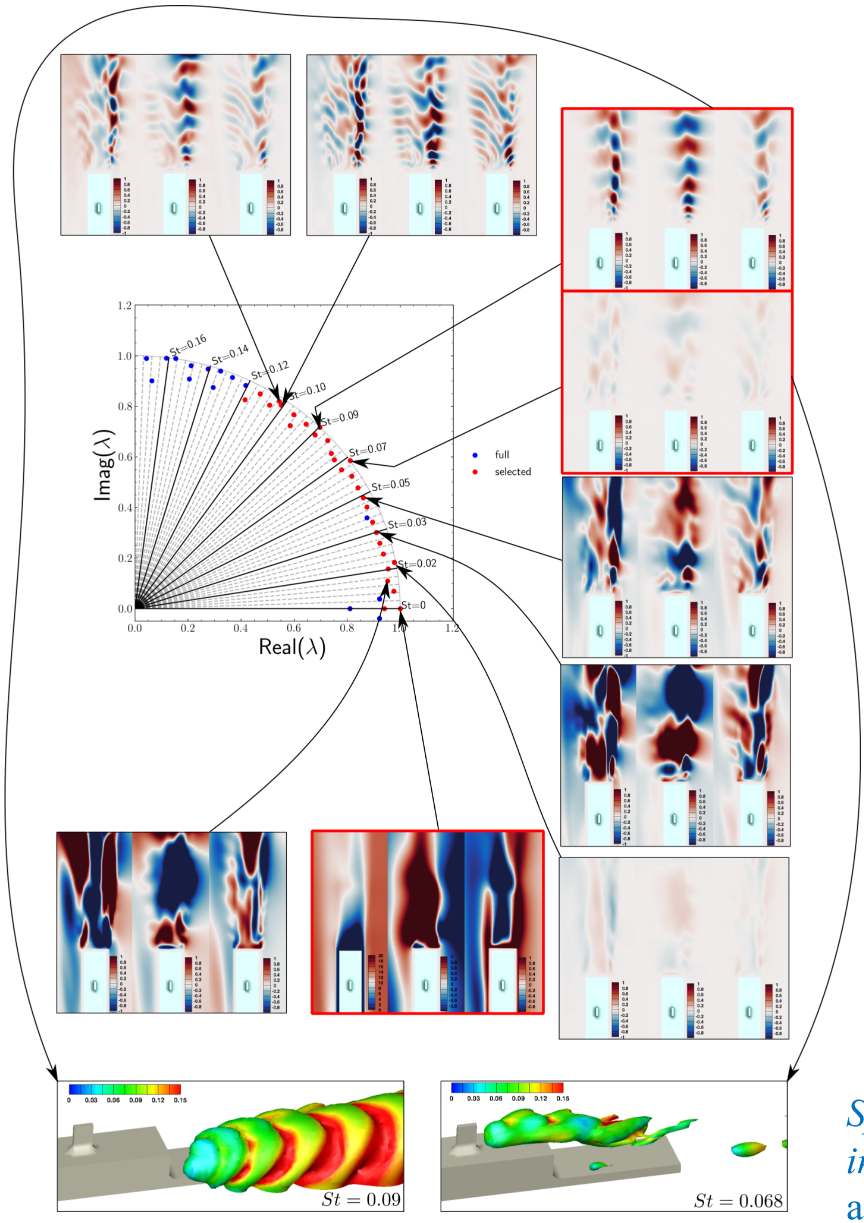
Steps:

- a-priori cross validation to choose an appropriate hyperparameter
- mode-by-mode error analysis
- choose a trade-off between reconstruction error and linear evolving error
- sparse reconstruction of system with multi-task learning

Sparsity-promoting algorithms for the discovery of informative Koopman invariant subspaces, Pan, S., N. A-M and Duraisamy, K., arXiv:2002.10637

Turbulent Ship Airwake





- transient behavior is accurately reconstructed
- stable modes are successfully extracted from strongly nonlinear transient data
- left mode: due to side edge of superstructure. right mode: due to funnel

Sparsity-promoting algorithms for the discovery of informative Koopman invariant subspaces, Pan, S., N. A-M and Duraisamy, K., arXiv:2002.10637

Summary

- Expressibility of deep neural nets $\rightarrow$ rich Ω_K
- Nonlinear reconstruction $\rightarrow$ linear embedding
- Differential form $\rightarrow$ known governing eqns / no data and recurrent form $\rightarrow$ trajectory data
- Guaranteed stability
- SVD-DMD as a short-cut similar to ResNet
- Mean-field variational inference (MFVI) with hierarchical Bayesian model for uncertainty

Many opportunities to enforce structure in Autoencoders ➔ flexible and powerful tools

Part 2

Learning Reduced Order Models of
Parametric Spatio-temporal dynamics

Non-intrusive data-driven ROMs

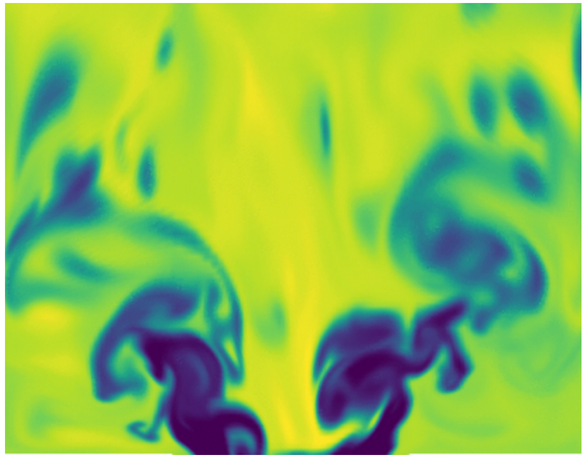
$$\mathbf{q}_l^{n+1} = f(\mathbf{q}_l^{n+1}, \mathbf{q}_l^{n}, \ldots .\mathbf{q}_l^{n-l}, B(\mathbf{u}^{n+1}), \mu)$$

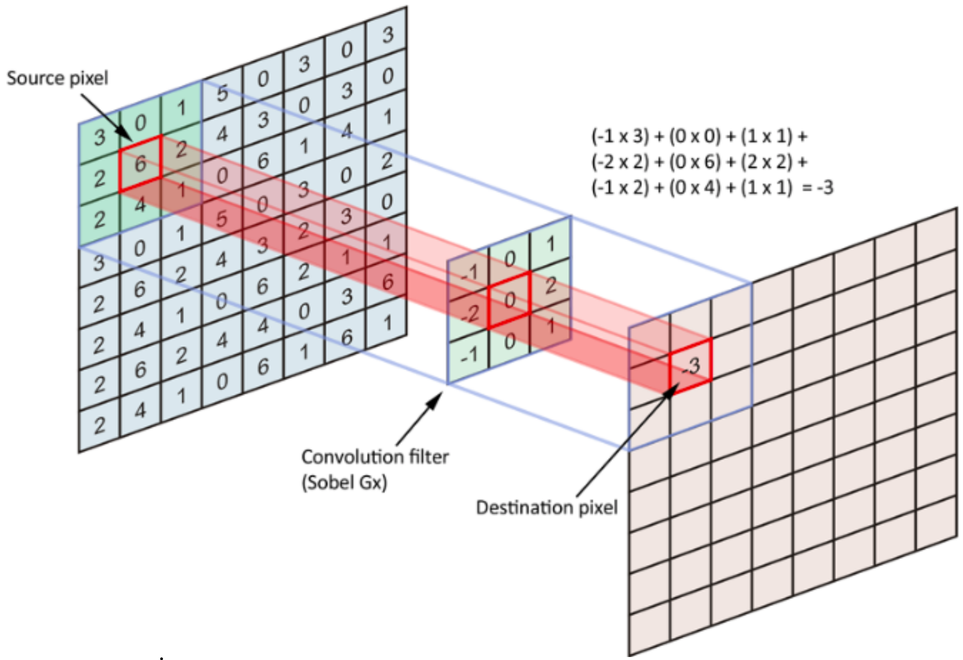
Some recent works:

- B. Kramer, K. E. Willcox, AIAA Journal ,2019
- M. Guo, J. S. Hesthaven, CMAME, 2018.
- A. Mohan, D. Daniel, M. Chertkov, D. Livescu, arXiv, 2019
- S. Lee, D. You, arXiv, 2019.
- Q. Wang, J. S. Hesthaven, D. Ray, JCP, 2019.

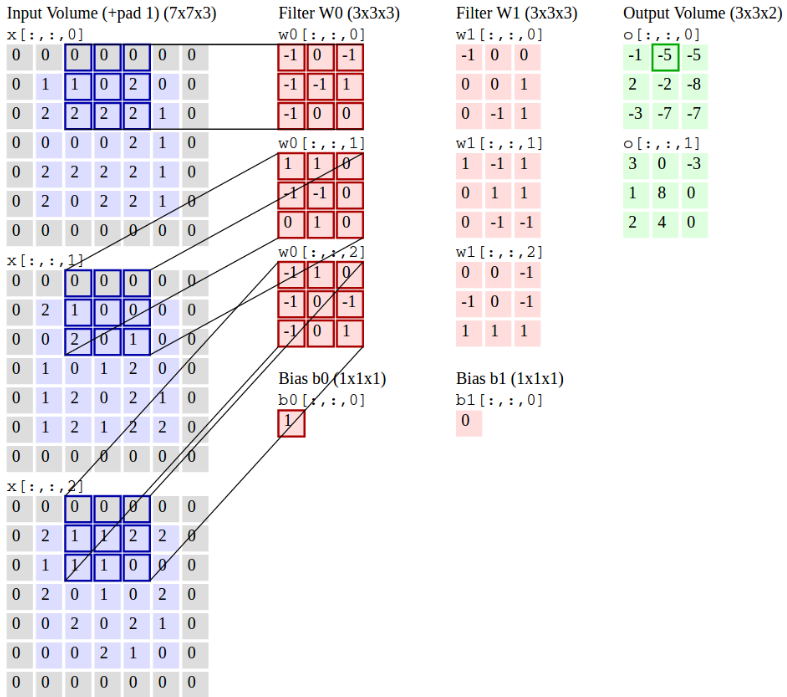
Basic Component: Convolutional Layer

- Convolutional layers preserve complex spatio-temporal “information”
- Convolutional operation on a local window *w*
  - $(x * w)_{ij} = \sum_{p=a}^{-a} \sum_{q=b}^{-b} x_{i-p,j-q} w_{p,q}$
- Ideal for “localized” feature identification
- Rotation and translation invariant, if properly constructed





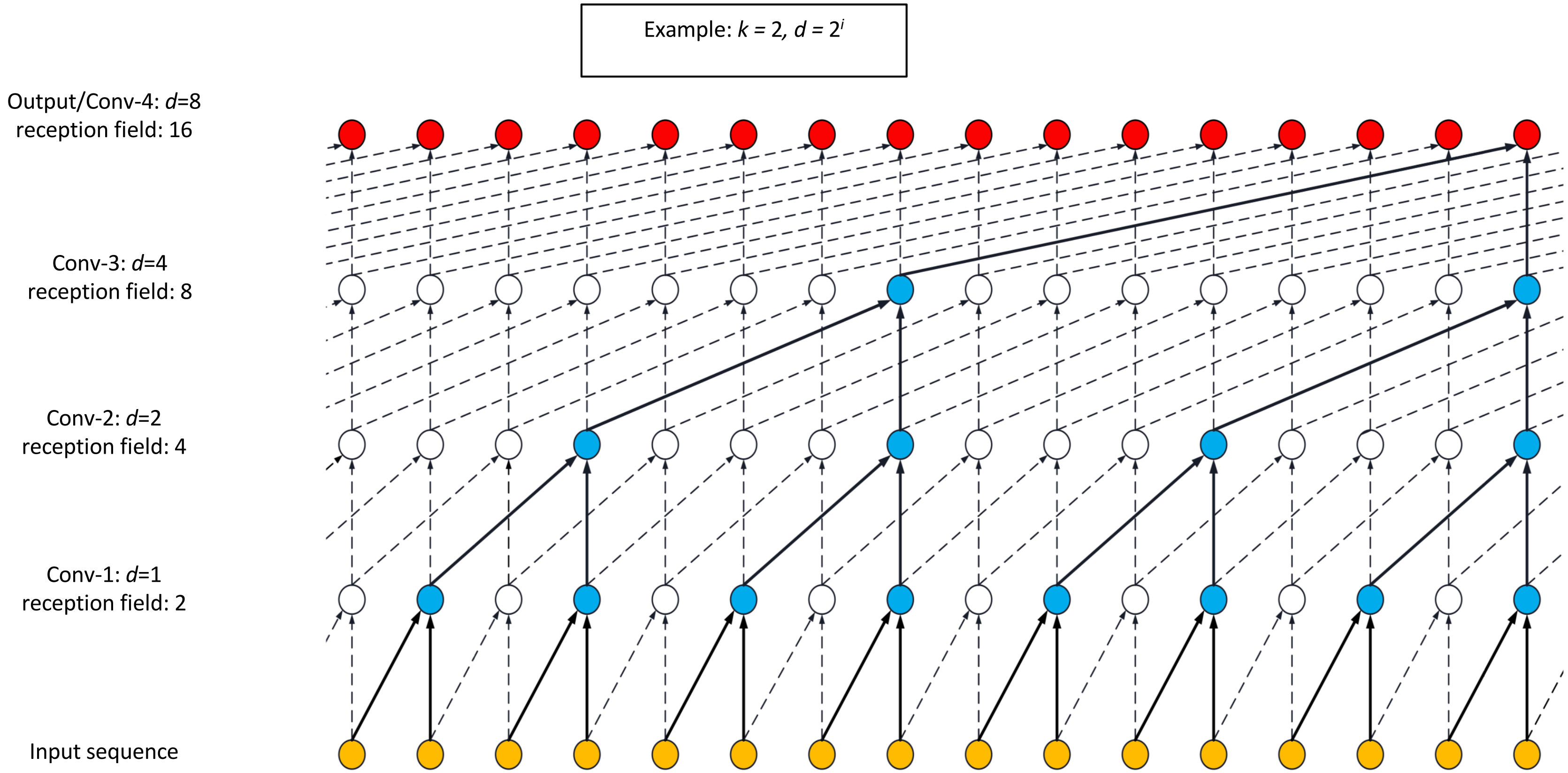
:
“Applied Deep Learning”
https://towardsdatascience.com/



http://cs231n.github.io/convolutional-networks/

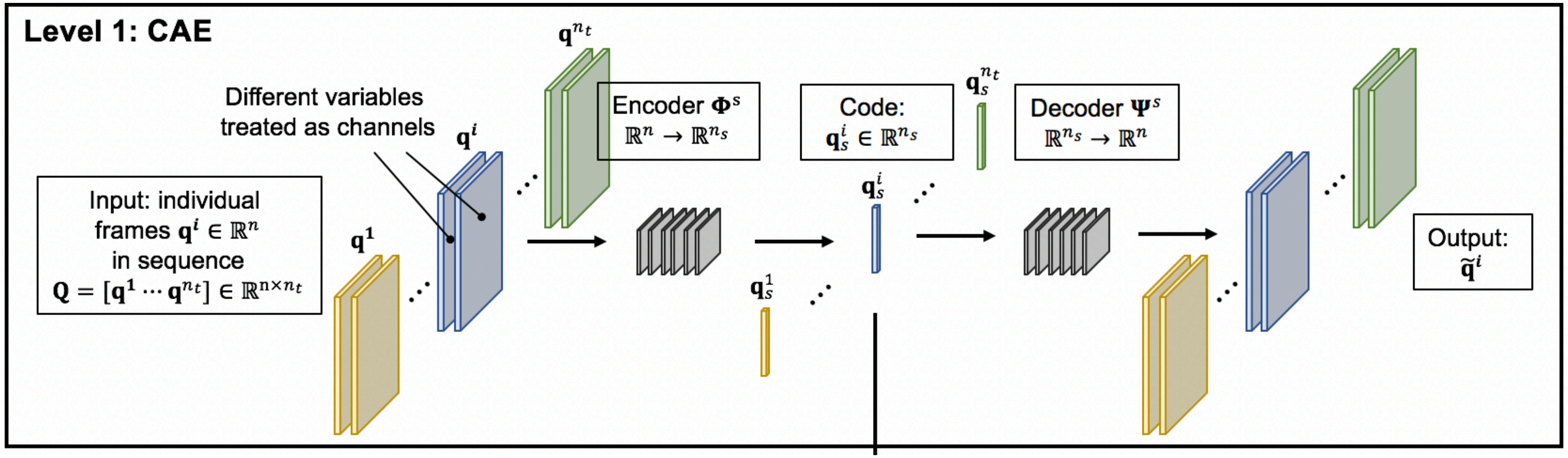
Temporal Convolutional

- Performs dilated 1D convolutional operation in temporal/sequential direction
  - $(x *_d w)_i = \sum_{p=0}^{k-1} x_{i-dp} w_p$
- Exponential increase in reception field ➔ an increasingly popular alternative to RNN/LSTM

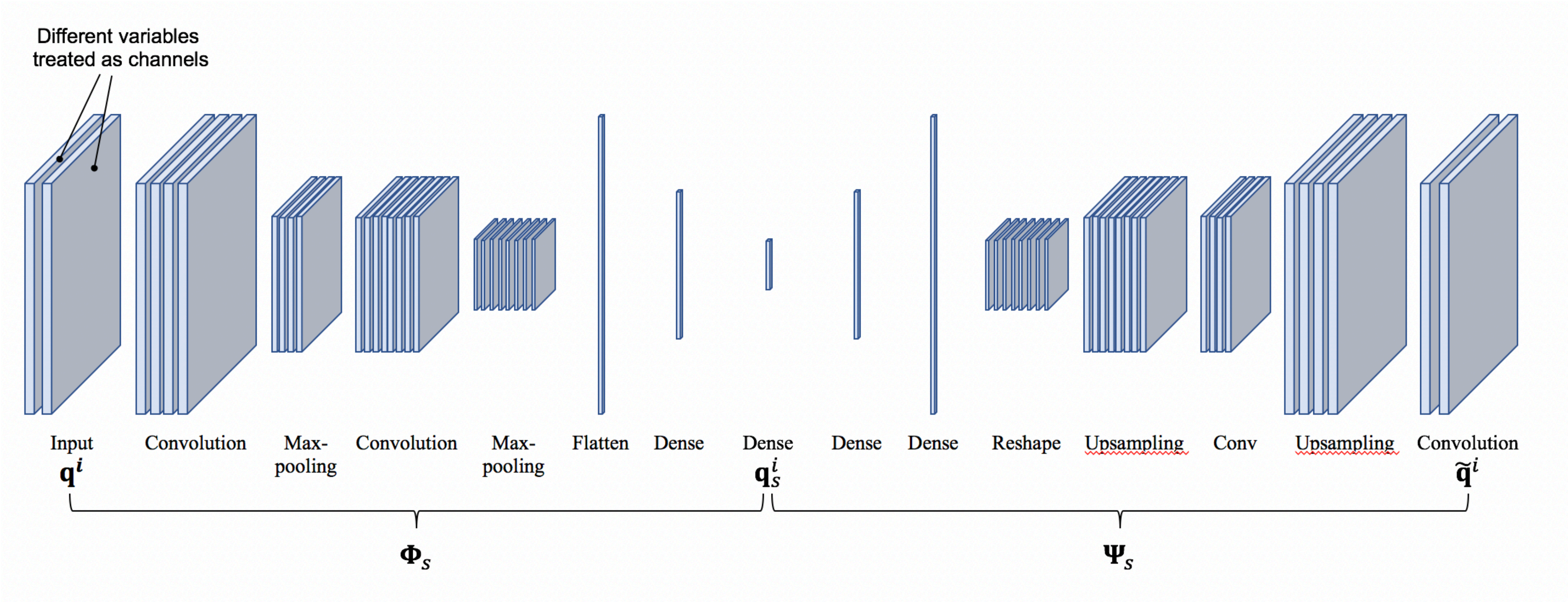


Source of image: github.com/philipperemy/keras-tcn

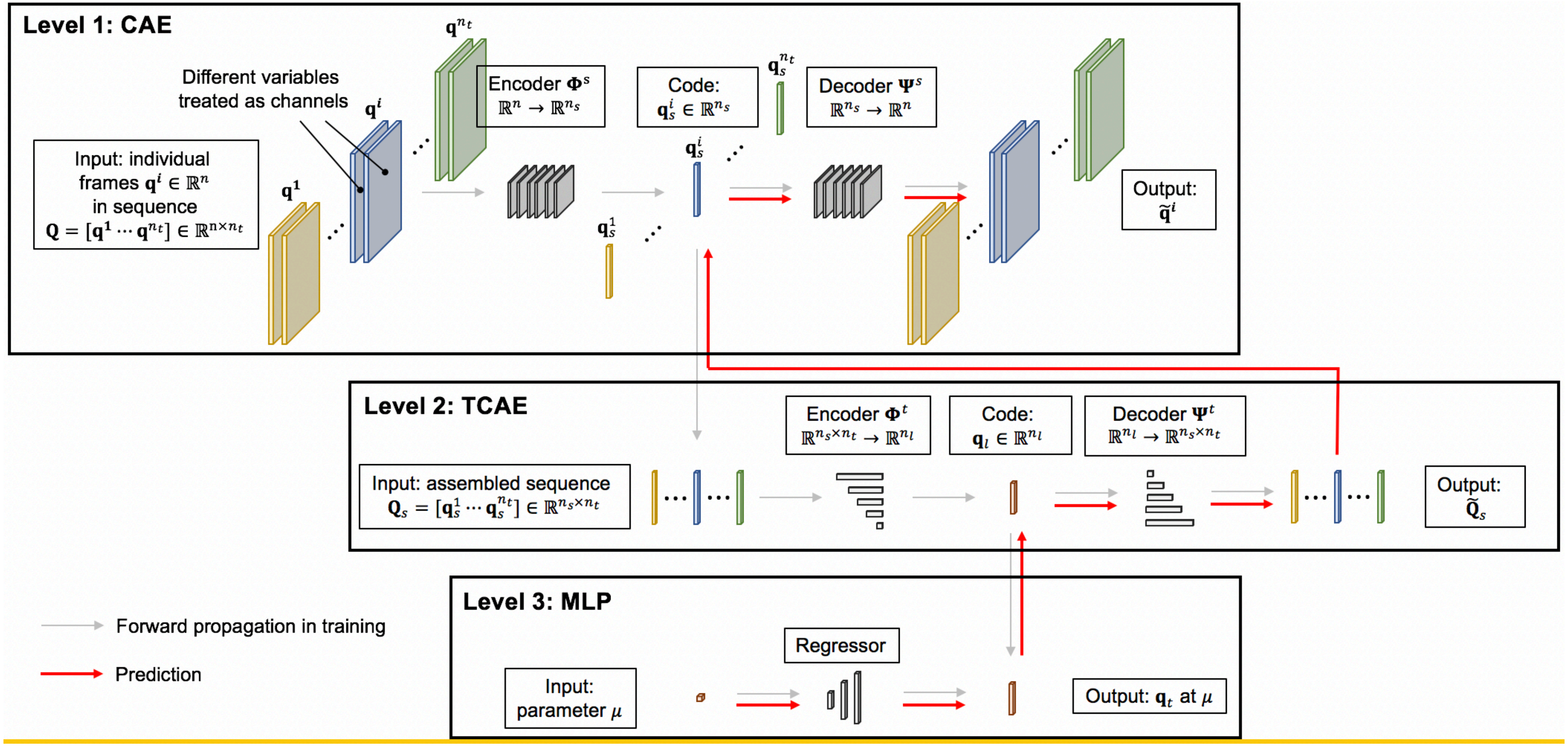
Training Multi-level convolutional AE networks



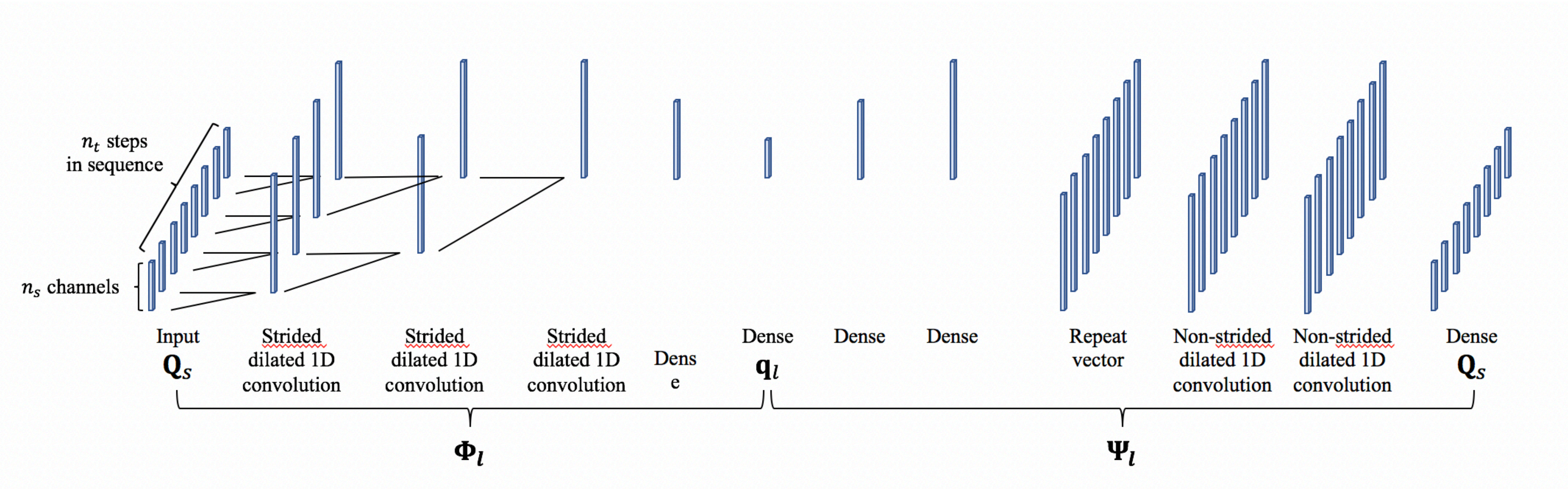
Example CAE architecture



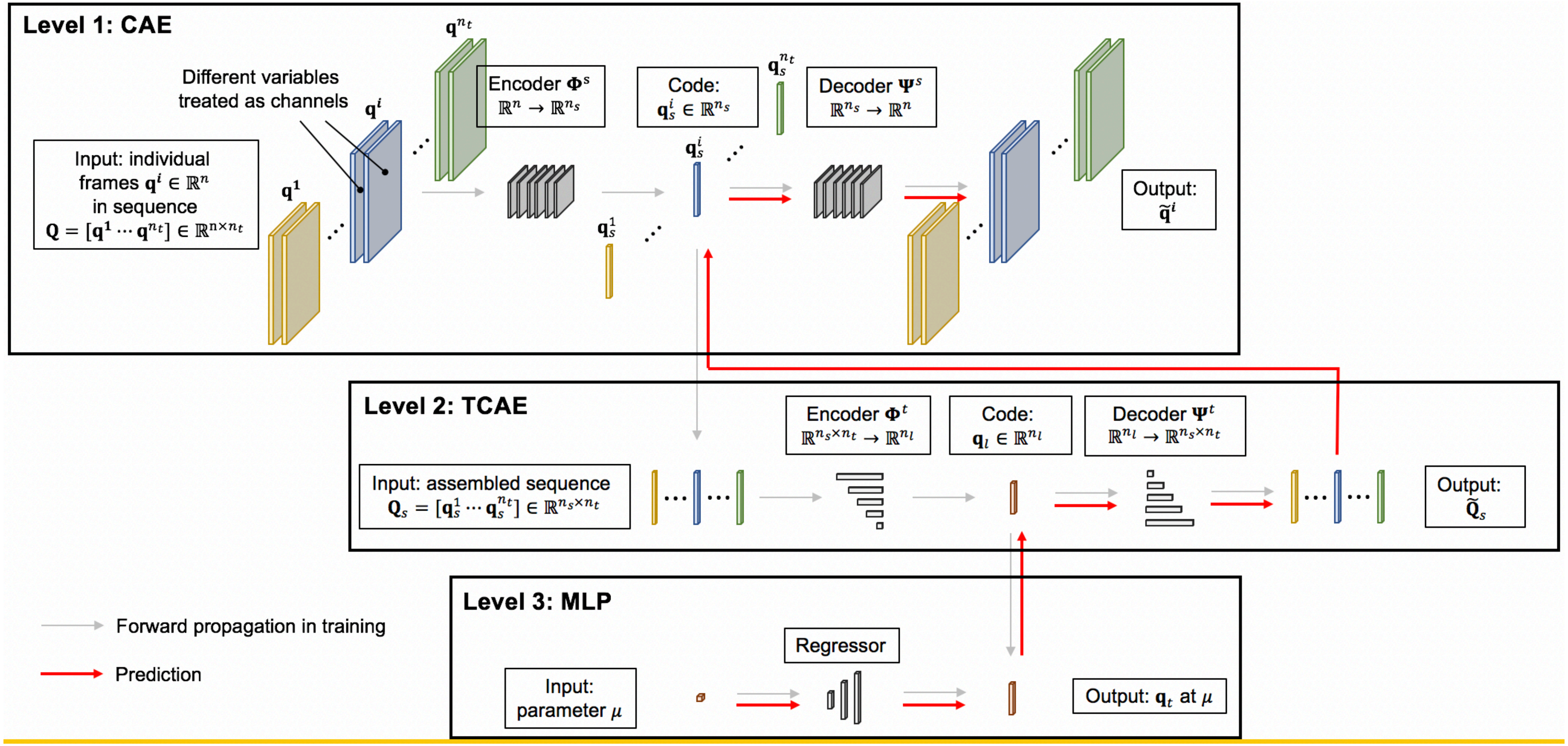
Prediction using Multilevel AE networks



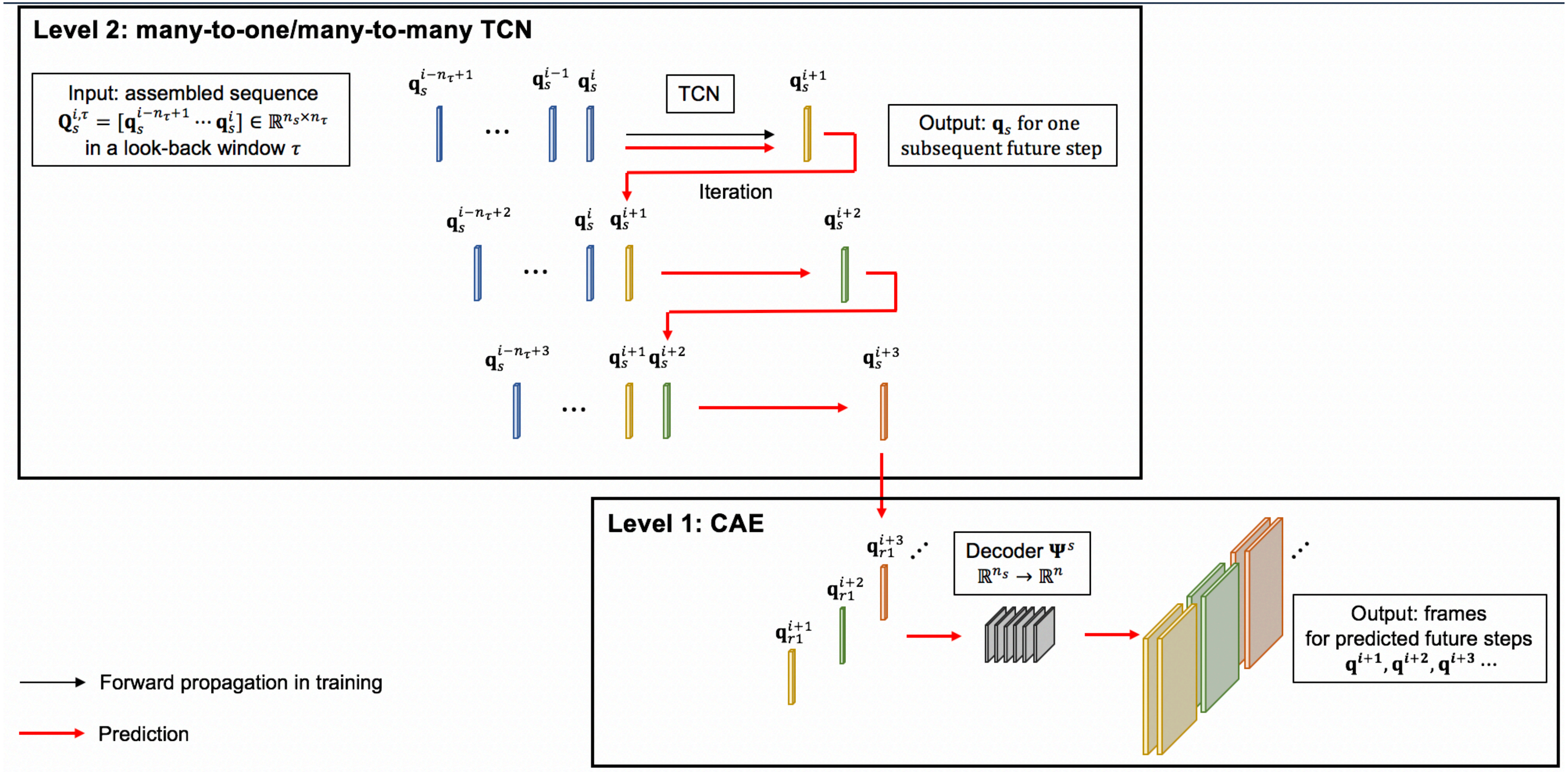
Example TCAE architecture



Prediction using Multilevel AE networks

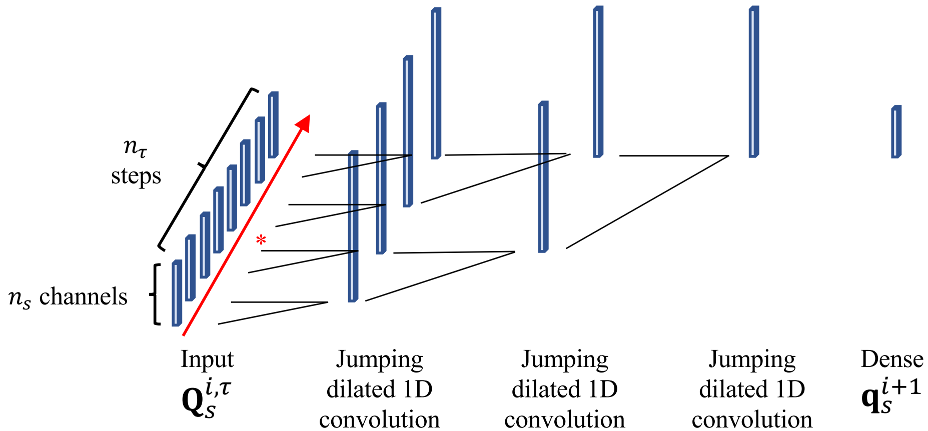


Time stepping



Example TCN architecture

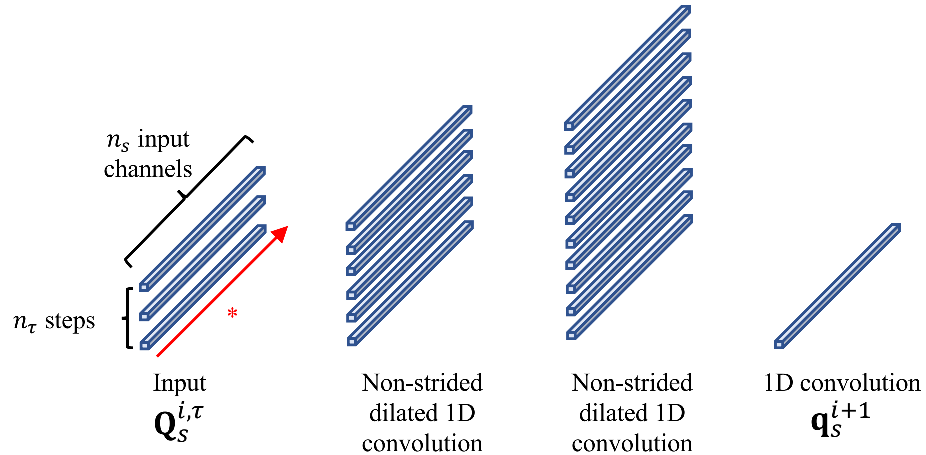
0	1	0	1	0	1	0
1	0	0	1	0	0	1



*: Convolution direction

Many-to-one

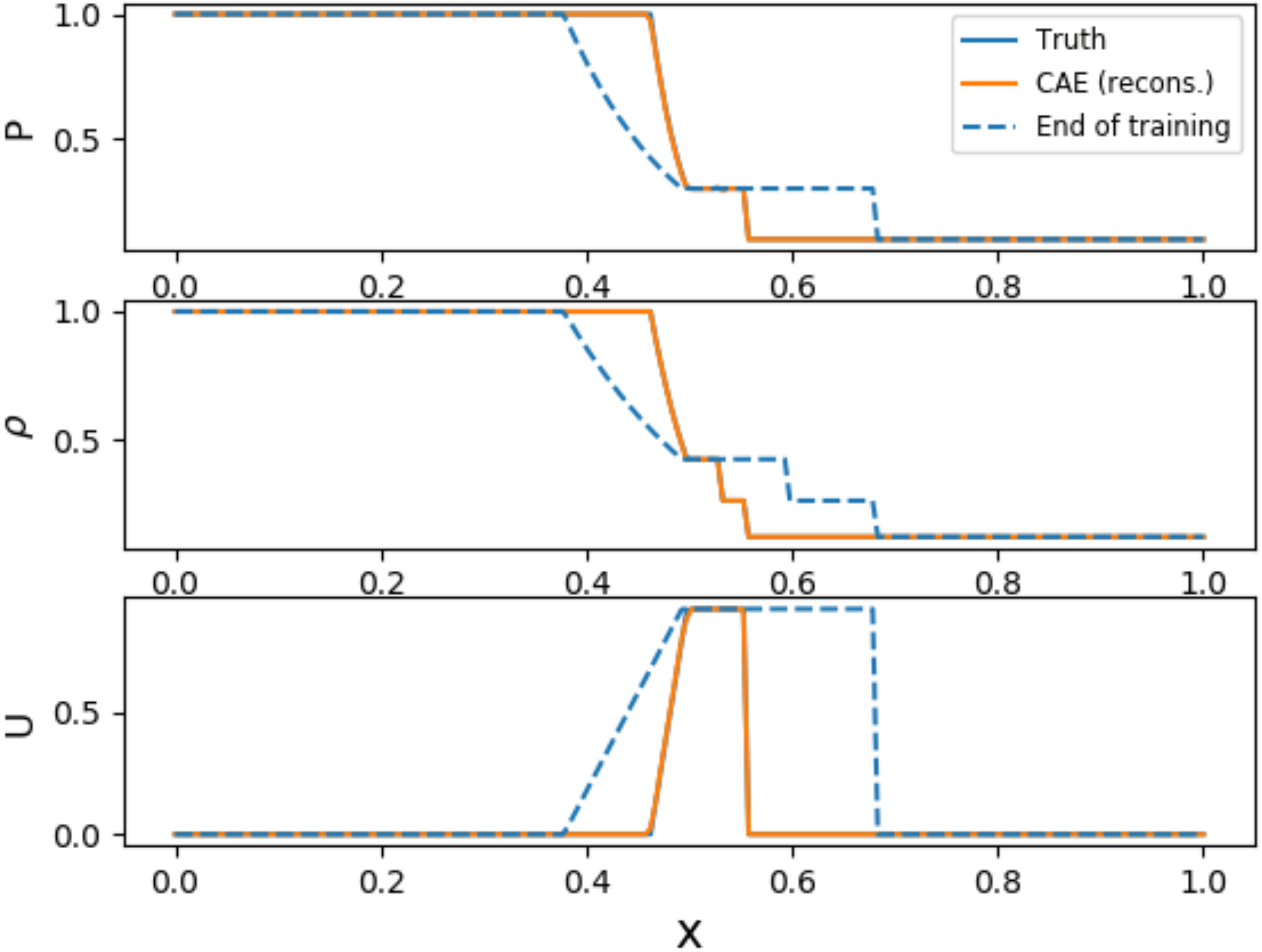
0	0	0	1	1	1	1
0	0	0	0	1	1	1
0	0	0	0	0	1	1
0	0	0	0	0	0	1



*: Convolution direction

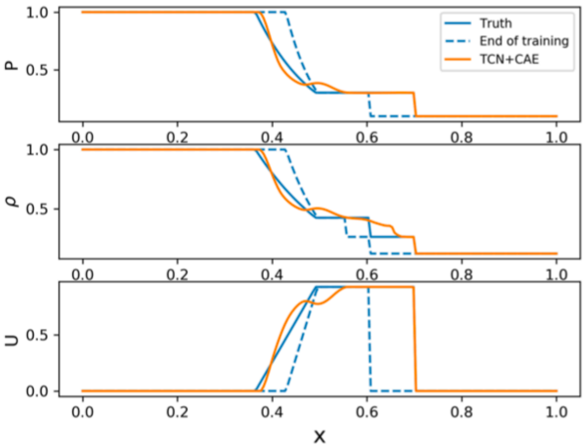
Many-to-many

Numerical Tests: Discontinuous compressible flow

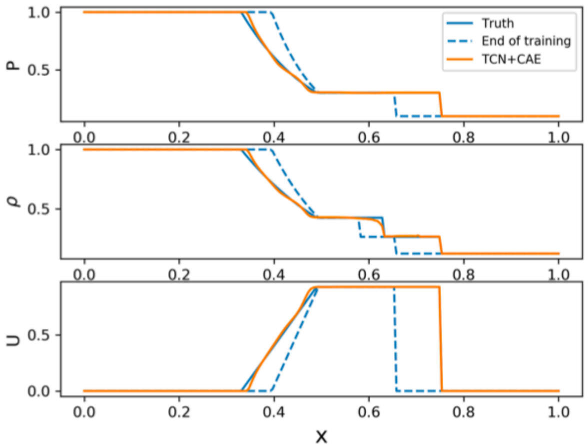


Component	**CAE reconstruction**		**CAE + TCN** (final step)
	Training	Testing	
Pressure	0.04%	0.1%	0.12%
Density	0.01%	0.04%	0.14%
Velocity	0.04%	0.08%	0.13%

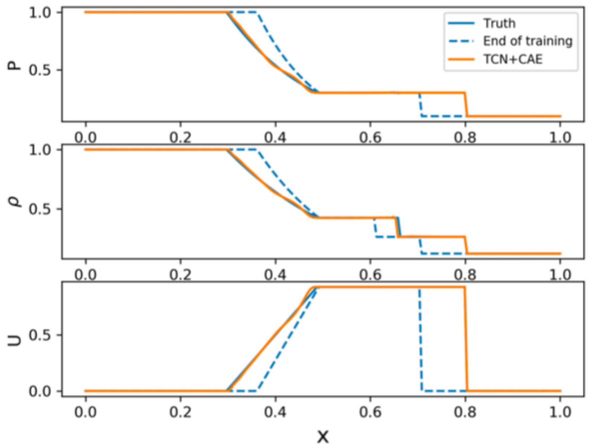
Discontinuous compressible flow : Impact of data



(a) $n_t = 20$

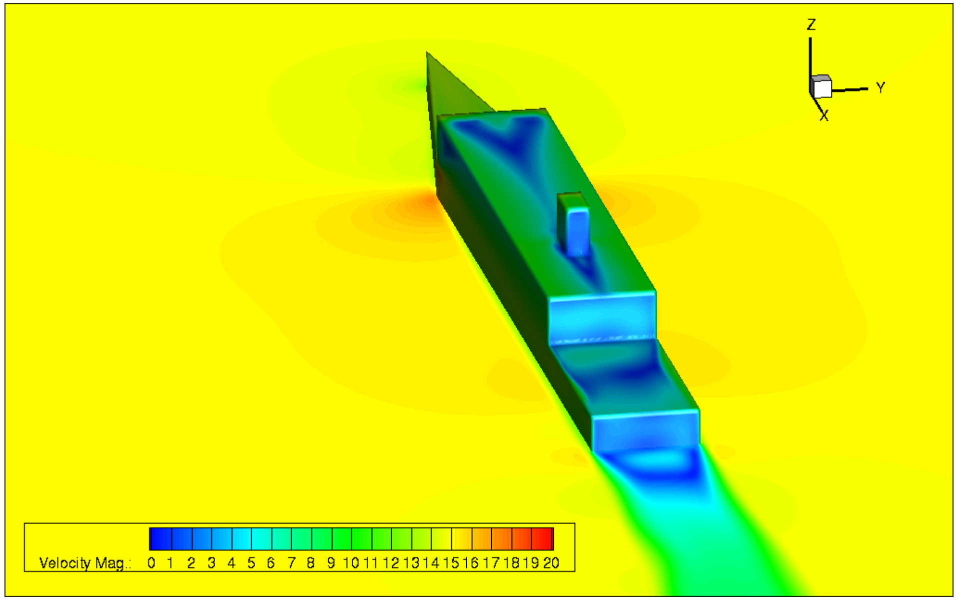


(b) $n_t = 30$

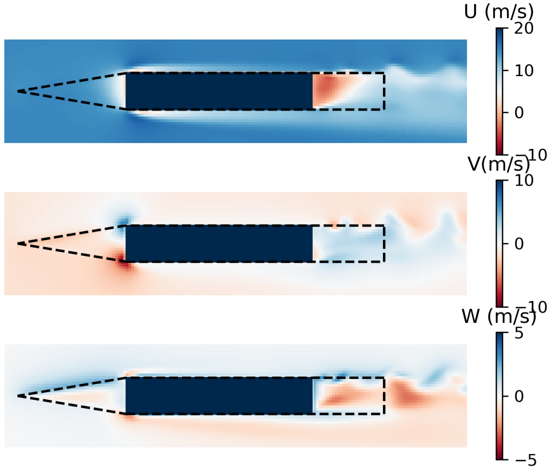


(c) $n_t = 40$

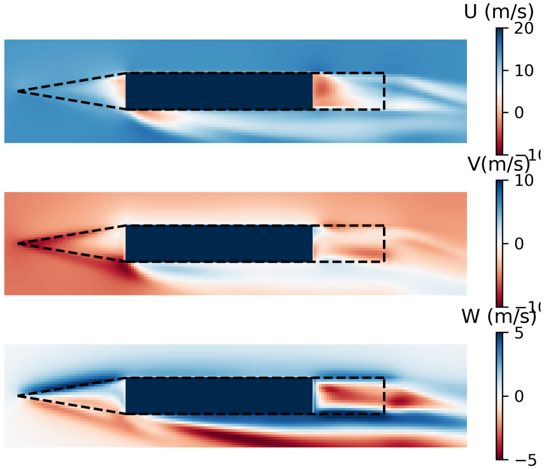
Numerical Tests : 3D Ship Airwake



- Incompressible Navier-Stokes
- 576k DOF, 400 time snapshots
- Global parameter: sliding angle α
- Training: α = 5° :5° :20°
- Prediction: α = 12.5

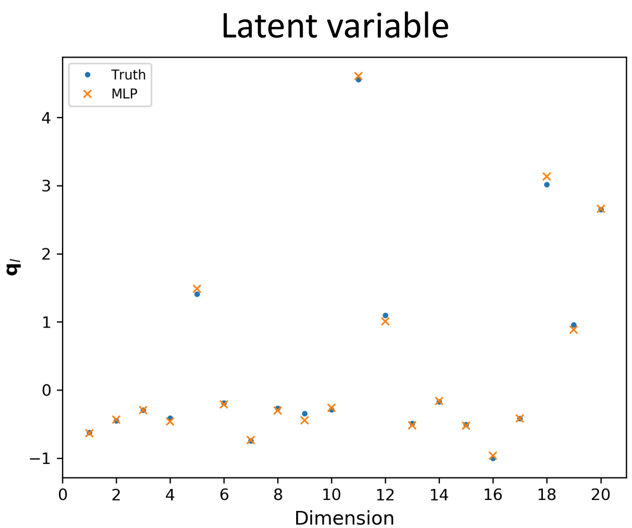


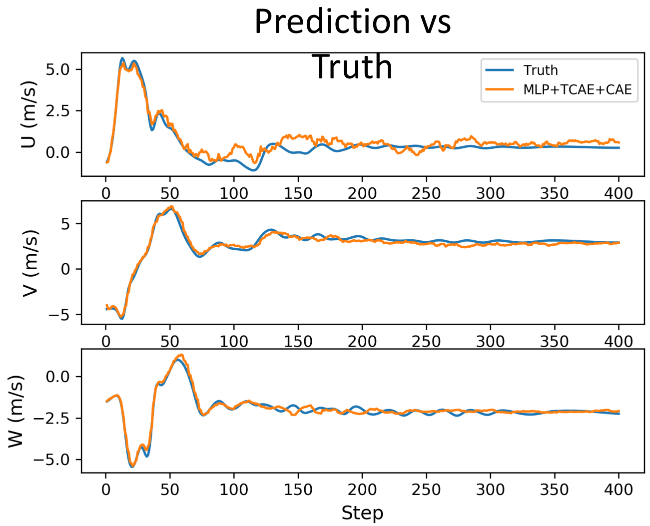
α = 5°

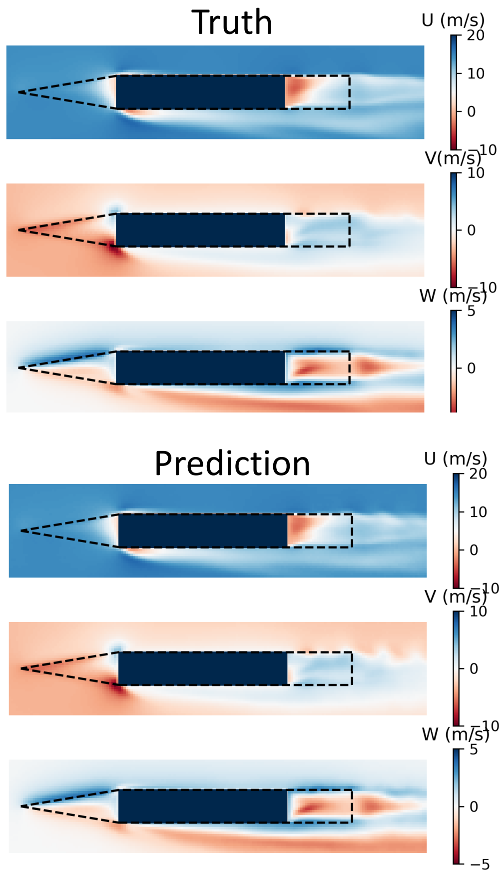


α = 20°

Numerical Test: 3D Ship Airwake







Component	CAE reconstruction		MLP + TCAE + CAE
	Training	Testing	
U	0.12%	0.30%	0.51%
V	0.09%	0.38%	0.89%
W	0.08%	0.29%	0.62%

Relative absolute error

Manuscript "Multi-level Convolutional Autoencoder Networks for Parametric Prediction of Spatio-temporal Dynamics," Submitted CMAME

Summary

- Fully Data-driven framework
  - Multi-level neural network architecture
  - Convolutions in space & time
- Non-linear manifolds
- Fast training, faster prediction
  - Up to 6 orders of reduction in DoF
  - Total training time: 3.6 hours on one NVIDIA Tesla P100 GPU for 3D ship air wake
  - Prediction time: Seconds for a new parameter or hundreds of future steps

Caveats

- Require large amounts of data
- No indicator for choice of latent dimensions ➔ use singular values to find an upper bound

Manuscript “Multi-level Convolutional Autoencoder Networks for Parametric Prediction of Spatio-temporal Dynamics” to be submitted to ArXiv in a week

Acknowledgments

- DARPA Physics of AI program (Technical Monitor: Dr. Ted Senator)
- Air Force Center of Excellence grant (Program Managers: Dr. Mitat Birkan and Dr. Fariba Fahroo)
- Office of Naval Research (Program manager: Dr. Brian Holm-Hansen)
- Computational infrastructure : NSF-MRI (Program manager: Dr. Stefan Robila)

Numerical Tests