ODEN INSTITUTE
FOR COMPUTATIONAL ENGINEERING & SCIENCES

Predictive data science for physical systems

From model reduction to scientific machine learning

Professor Karen E. Willcox

ICIAM 2019
Valencia, Spain
July 19, 2019

TEXAS
The University of Texas at Austin

Contributors

Dr. Boris Kramer

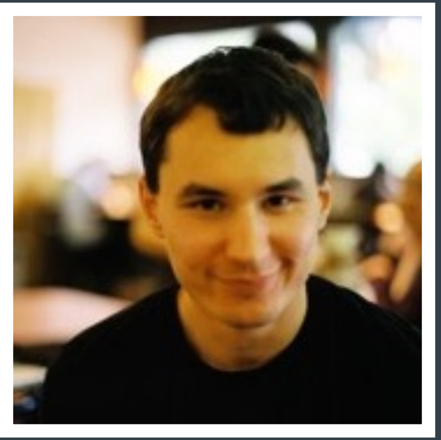
MIT → UCSD

Elizabeth Qian

MIT

Renee Swischuk

MIT



Prof. Benjamin Peherstorfer

Courant Institute

Funding sources: US Air Force **Computational Math Program** (F. Fahroo);
US Air Force **Center of Excellence on Rocket Combustion** (M. Birkan, F. Fahroo, D. Talley);
SUTD-MIT **International Design Centre**

Outline

1 **Predictive Data Science**

What & why

2 **Lift and Learn**

Projection-based model reduction as a lens through which to learn predictive models

3 **Application example**

Rocket engine combustion

4 **Conclusions & Outlook**

Predictive Data Science

What and Why

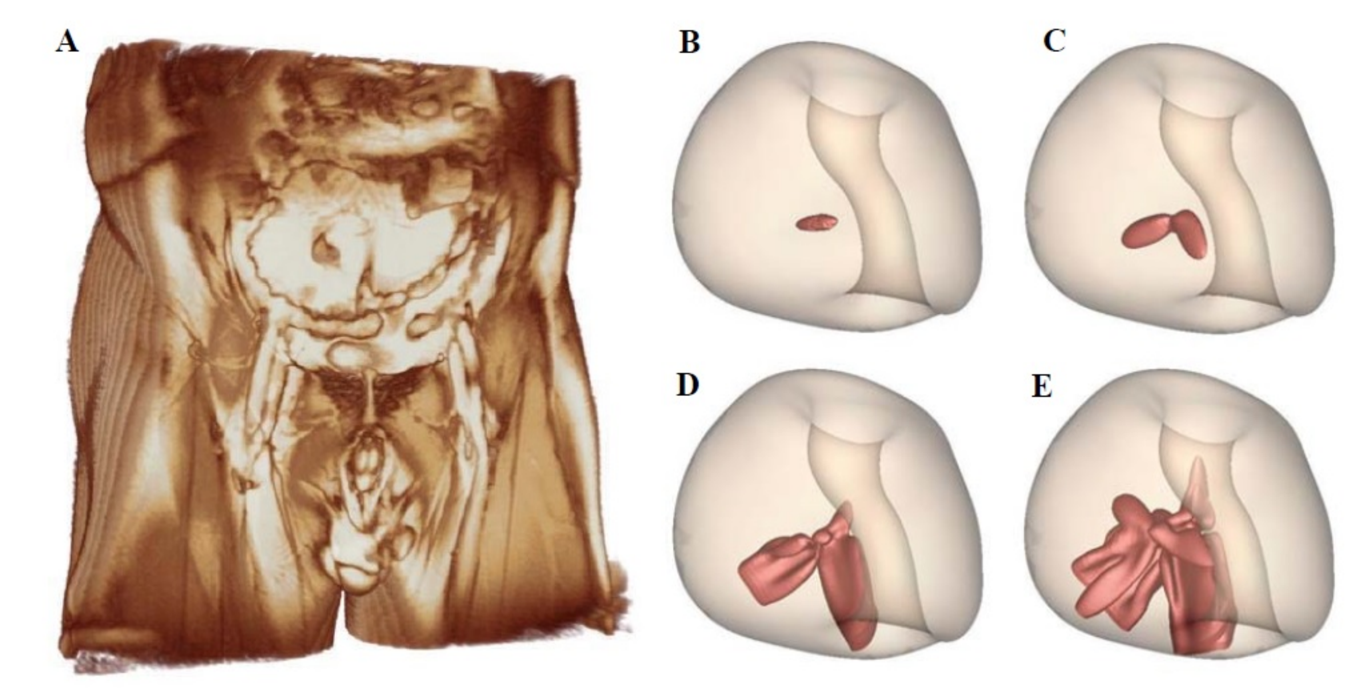
ODEN INSTITUTE
FOR COMPUTATIONAL ENGINEERING & SCIENCES

How do we harness the explosion of data to extract knowledge, insight and decisions?

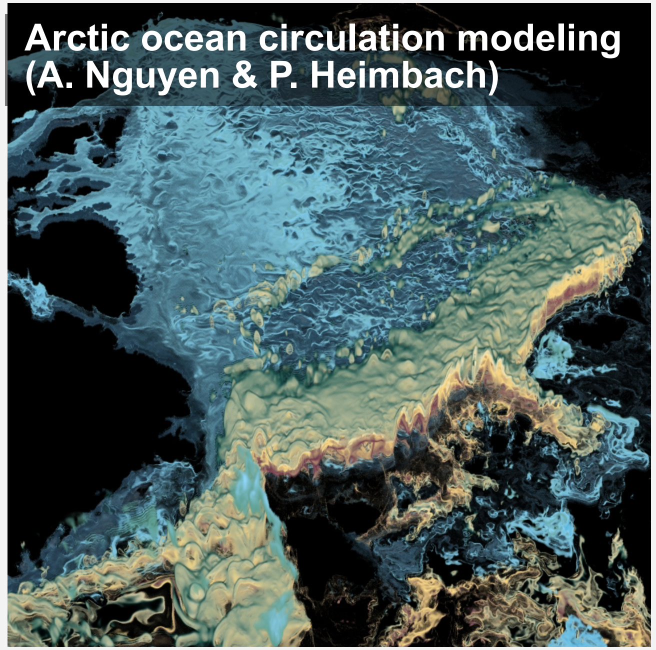
Big decisions need more than just big data…

they need **big models** too

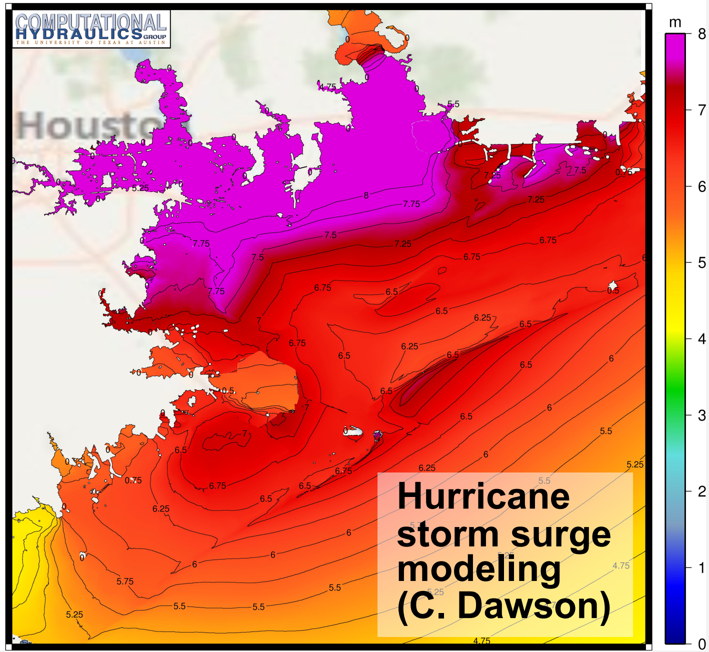
Inspired by
Coveney, Dougherty, Highfield “Big data need big theory too”



Patient-specific prostate tumor modeling (T. Hughes)



Arctic ocean circulation modeling (A. Nguyen & P. Heimbach)



Hurricane storm surge modeling (C. Dawson)

Big decisions need more than just big data…

Big decisions must incorporate the **predictive power**, **interpretability**, and **domain knowledge** of physics-based models

Predictive Data Science

a **convergence** of
Data Science and
Computational Science & Engineering

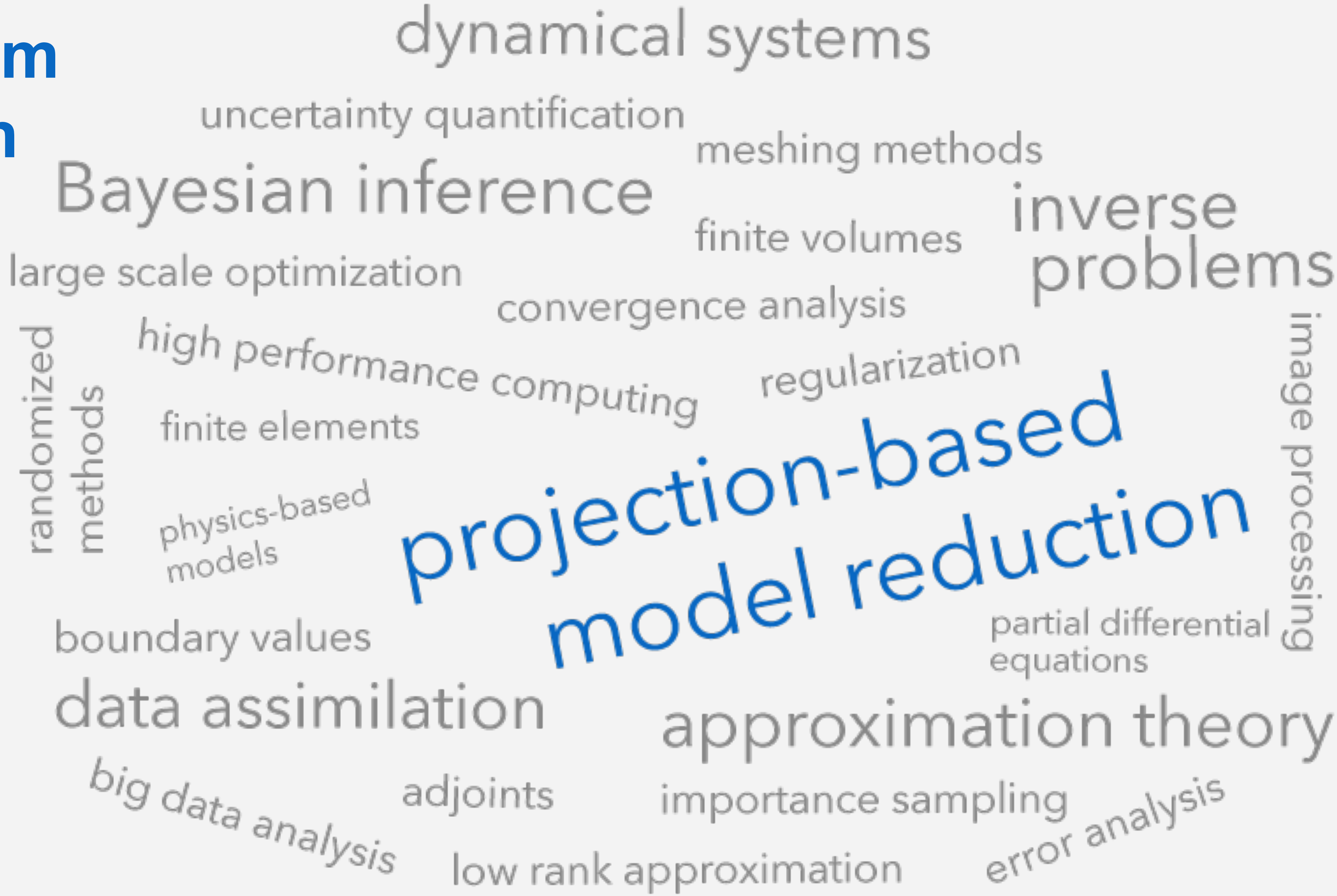
Challenges

1. **high-consequence** applications are characterized by **complex multiscale multiphysics** dynamics
2. **high** (and even infinite) **dimensional parameters**
3. **data** are relatively **sparse** and **expensive to acquire**
4. **uncertainty quantification** in model inference and **certified predictions** in regimes **beyond training data**

Learning from data through the lens of models…

dynamical systems
uncertainty quantification
meshing methods
Bayesian inference
inverse problems
finite volumes
large scale optimization
convergence analysis
high performance computing
regularization
randomized methods
finite elements
image processing
physics-based models
projection-based model reduction
partial differential equations
boundary values
data assimilation
approximation theory
big data analysis
adjoints
importance sampling
error analysis
low rank approximation

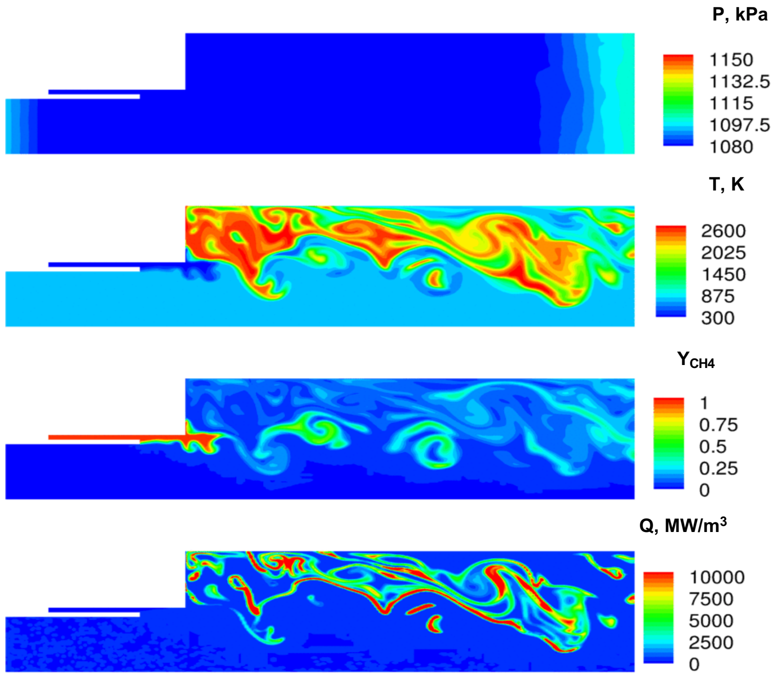
Learning from data through the lens of models…



Lift & Learn

Projection-based model reduction as a lens through which to learn predictive models

Lift & Learn: Ingredients



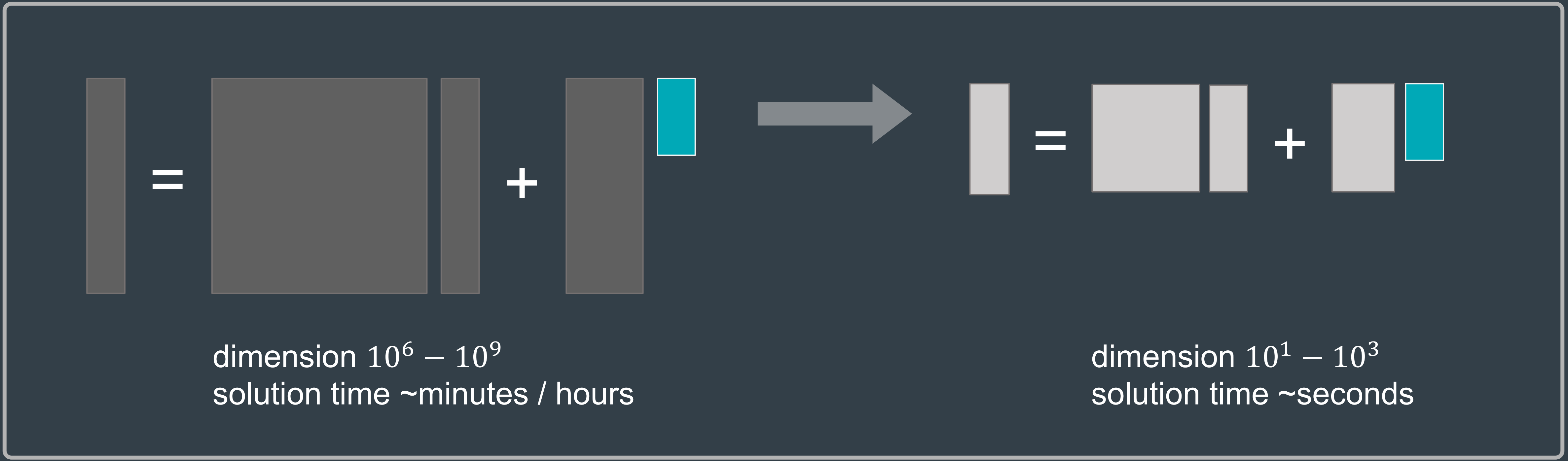
1. A **physics-based model**
   Example: modeling combustion in a rocket engine
   Conservation of mass (ρ), momentum ($\rho\vec{w}$), energy (E), species ($Y_{\mathrm{CH_4}}, Y_{\mathrm{O_2}}, Y_{\mathrm{CO_2}}, Y_{\mathrm{H_2O}}$)

$$\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{p})$$

2. Lens of **projection** to define a low-dimensional model

3. **Variable transformations** that expose polynomial structure in the model

4. **Non-intrusive learning** of the reduced model → work with transformed variables



Projection-based model reduction

1 **Label**: Solve PDEs to generate training data (snapshots)
2 **Identify structure**: Compute a low-dimensional basis
3 **Train**: Project PDE model onto the low-dimensional subspace

Reduced models

1 Label

2 Identify structure

3 Train

$\mathbf{E}_r = \mathbf{V}^\top \mathbf{E}\mathbf{V}$

$\mathbf{A}_r = \mathbf{V}^\top \mathbf{A}\mathbf{V}$

$\mathbf{H}_r = \mathbf{V}^\top \mathbf{H}(\mathbf{V} \otimes \mathbf{V})$

$\mathbf{B}_r = \mathbf{V}^\top \mathbf{B}$

Full-order model (FOM)
state $\mathbf{x} \in \mathbb{R}^N$

$$\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

Approximate
$\mathbf{x} \approx \mathbf{V}\mathbf{x}_r$
$V \in \mathbb{R}^{N \times r}$

Residual: N eqs $\gg$ r dof

$$\mathbf{r} = \mathbf{E}\mathbf{V}\dot{\mathbf{x}}_r - \mathbf{A}\mathbf{V}\mathbf{x}_r - \mathbf{B}\mathbf{u}$$

Project
$\mathbf{W}^\top \mathbf{r} = 0$
(Galerkin: $\mathbf{W} = \mathbf{V}$)

Reduced-order model (ROM)
state $\mathbf{x}_r \in \mathbb{R}^r$

$$\mathbf{E}_r\dot{\mathbf{x}}_r = \mathbf{A}_r\mathbf{x}_r + \mathbf{B}_r\mathbf{u}$$

Linear Model

FOM: $\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$

ROM: $\mathbf{E}_r\dot{\mathbf{x}}_r = \mathbf{A}_r\mathbf{x}_r + \mathbf{B}_r\mathbf{u}$

Precompute the ROM matrices:

$$\mathbf{A}_r = \mathbf{V}^\top\mathbf{A}\mathbf{V},\ \mathbf{B}_r = \mathbf{V}^\top\mathbf{B},\ \mathbf{E}_r = \mathbf{V}^\top\mathbf{E}\mathbf{V}$$

Quadratic Model

FOM: $\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{H}(\mathbf{x} \otimes \mathbf{x}) + \mathbf{B}\mathbf{u}$

ROM: $\mathbf{E}_r\dot{\mathbf{x}}_r = \mathbf{A}_r\mathbf{x}_r + \mathbf{H}_r(\mathbf{x}_r \otimes \mathbf{x}_r) + \mathbf{B}_r\mathbf{u}$

Precompute the ROM matrices and tensor:

$$\mathbf{H}_r = \mathbf{V}^\top\mathbf{H}(\mathbf{V} \otimes \mathbf{V})$$

projection preserves structure $\longleftrightarrow$ structure embeds physical constraints

Machine learning

“Machine learning is a field of computer science that uses statistical techniques to give computer systems the ability to "learn" with data, without being explicitly programmed.” [Wikipedia]

Reduced-order modeling

“Model order reduction (MOR) is a technique for reducing the computational complexity of mathematical models in numerical simulations.” [Wikipedia]

What is the connection between reduced-order modeling and machine learning?

Model reduction methods have grown from CSE, with a focus on *reducing* high-dimensional models that arise from physics-based modeling, whereas machine learning has grown from CS, with a focus on *creating* low-dimensional models from black-box data streams. Yet recent years have seen an increased blending of the two perspectives and a recognition of the associated opportunities.
[Swischuk et al., *Computers & Fluids*, 2018]

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\\ E\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2+p\\ (E+p)u\end{pmatrix}=0$$

$$E=\frac{p}{\gamma-1}+\frac{1}{2}\rho u^2$$

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ u\\ p\end{pmatrix}+\begin{pmatrix}\rho\frac{\partial u}{\partial x}+u\frac{\partial \rho}{\partial x}\\ u\frac{\partial u}{\partial x}+\frac{1}{\rho}\frac{\partial p}{\partial x}\\ \gamma p\frac{\partial u}{\partial x}+u\frac{\partial p}{\partial x}\end{pmatrix}=0$$

$$\frac{\partial}{\partial t}\begin{pmatrix}u\\ p\\ q\end{pmatrix}+\begin{pmatrix}u\frac{\partial u}{\partial x}+q\frac{\partial p}{\partial x}\\ \gamma p\frac{\partial u}{\partial x}+u\frac{\partial p}{\partial x}\\ q\frac{\partial u}{\partial x}+u\frac{\partial q}{\partial x}\end{pmatrix}=0$$

Variable Transformations & Lifting

The physical governing equations reveal variable transformations and manipulations that expose polynomial structure

There are multiple ways to write the Euler equations

Different choices of variables leads to different *structure* in the discretized system
→ ***lifting***

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho \\ \rho w \\ E\end{pmatrix} + \frac{\partial}{\partial z}\begin{pmatrix}\rho w \\ \rho w^2 + p \\ (E+p)w\end{pmatrix} = 0$$

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho w^2$$

conservative variables
mass, momentum, energy

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho \\ w \\ p\end{pmatrix} + \begin{pmatrix}\rho \frac{\partial w}{\partial x} + w \frac{\partial \rho}{\partial x} \\ w \frac{\partial w}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} \\ \gamma p \frac{\partial w}{\partial x} + w \frac{\partial p}{\partial x}\end{pmatrix} = 0$$

primitive variables
mass, velocity, pressure

- Define specific volume: $q = {}^{1}/_{\rho}$
- Take derivative: $\frac{\partial q}{\partial t} = \frac{-1}{\rho^2}\frac{\partial \rho}{\partial t} = \frac{-1}{\rho^2}\left(-\rho\frac{\partial u}{\partial x} - u\frac{\partial \rho}{\partial x}\right) = q\frac{\partial u}{\partial x} - u\frac{\partial q}{\partial x}$

$$\frac{\partial}{\partial t}\begin{pmatrix}w \\ p \\ q\end{pmatrix} + \begin{pmatrix}w \frac{\partial w}{\partial x} + q \frac{\partial p}{\partial x} \\ \gamma p \frac{\partial w}{\partial x} + w \frac{\partial p}{\partial x} \\ q \frac{\partial w}{\partial x} + w \frac{\partial q}{\partial x}\end{pmatrix} = 0$$

specific volume variables

$$\mathbf{E}\dot{\mathbf{x}} = \underbrace{\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}}_{\text{linear}} + \underbrace{\mathbf{H}(\mathbf{x} \otimes \mathbf{x})}_{\text{quadratic}}$$

transformed system
has linear-quadratic structure

cf. $\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{p})$

Simple example

Lifting a nonlinear (quartic) ODE to quadratic-bilinear form

Can either lift to a system of ODEs or to a system of DAEs

Consider the quartic system $\dot{x} = x^4 + u$

Introduce auxiliary variables: $w_1 = x^2 \quad w_2 = w_1^2$

Chain rule:

$$\dot{w}_1 = 2x[w_1^2 + u] = 2x[w_2 + u]$$

$$\dot{w}_2 = 2w_1\dot{w}_1 = 4xw_1[w_2 + u]$$

Need additional variable to make auxiliary dynamics quadratic:

$$w_3 = xw_1 \qquad \dot{w}_3 = \dot{x}w_1 + x\dot{w}_1$$

$$= w_1w_2 + w_1u + 2w_1w_2 + 2w_1u$$

QB-ODE

$$\begin{aligned} \dot{x} &= w_2 + u \\ \dot{w}_1 &= 2xw_2 + 2xu \\ \dot{w}_2 &= 4w_2w_3 + 4w_3u \\ \dot{w}_3 &= 3w_1w_2 + 3w_1u \end{aligned}$$

QB-DAE

$$\begin{aligned} \dot{x} &= w_1^2 + u \\ 0 &= w_1 - x^2 \end{aligned}$$

Many different forms of nonlinear equations can be lifted to polynomial form

$$\dot{\psi} = \frac{1}{Pe}\psi_{ss} - \psi_s - \mathcal{D}\psi e^{\gamma - \frac{\gamma}{\theta}}$$
$$\dot{\theta} = \frac{1}{Pe}\theta_{ss} - \theta_s - \beta(\theta - \theta_{\text{ref}}) + \mathcal{B}\mathcal{D}\psi e^{\gamma - \frac{\gamma}{\theta}}$$

original equations

$$\dot{\psi} = \underbrace{\frac{1}{Pe}\psi_{ss} - \psi_s - \mathcal{D}w_4}_{\text{linear}}$$
$$\dot{\theta} = \underbrace{\frac{1}{Pe}\theta_{ss} - \theta_s - \beta(\theta - \theta_{\text{ref}}) + \mathcal{B}\mathcal{D}w_4}_{\text{linear}}$$
$$\dot{w}_1 = \gamma\ w_6 \left[\frac{1}{Pe}\psi_{ss} - \psi_s\right] + \gamma\mathcal{B}\mathcal{D}\ w_4 w_6$$
$$\dot{w}_2 = -2\ w_5 \odot \left[\frac{1}{Pe}\psi_{ss} - \psi_s\right] - 2\mathcal{B}\mathcal{D}\ w_4 w_5$$
$$\dot{w}_3 = -w_2 \odot \left[\frac{1}{Pe}\psi_{ss} - \psi_s\right] - \mathcal{B}\mathcal{D}\ w_2 w_4$$
$$0 = w_4 - w_1\psi$$
$$0 = w_5 - w_2 w_3$$
$$0 = w_6 - w_1 w_2$$

quadratic-bilinear lifted equations

Operator inference

Non-intrusive learning of the reduced models from simulation snapshot data

Given state data, learn the system

In principle could learn a large, sparse system e.g., Schaeffer, Tran & Ward, 2017

$$\mathbf{E}\dot{\mathbf{x}} = \underbrace{\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}}_{\text{linear}} + \underbrace{\mathbf{H}(\mathbf{x} \otimes \mathbf{x})}_{\text{quadratic}}$$

Given state data ($\mathbf{X}$) and velocity data ($\dot{\mathbf{X}}$):

$$\mathbf{X} = \begin{bmatrix} | & & | \\ \mathbf{x}(t_1) & \dots & \mathbf{x}(t_K) \\ | & & | \end{bmatrix} \qquad \dot{\mathbf{X}} = \begin{bmatrix} | & & | \\ \dot{\mathbf{x}}(t_1) & \dots & \dot{\mathbf{x}}(t_K) \\ | & & | \end{bmatrix}$$

Find the operators $\mathbf{A}, \mathbf{B}, \mathbf{E}, \mathbf{H}$
by solving the least squares problem:

$$\min_{\mathbf{A},\mathbf{B},\mathbf{E},\mathbf{H}} \left\| \mathbf{X}^\top \mathbf{A}^\top + (\mathbf{X} \otimes \mathbf{X})^\top \mathbf{H}^\top + \mathbf{U}^\top \mathbf{B}^\top - \dot{\mathbf{X}}^\top \mathbf{E} \right\|$$

Given *reduced* state data, learn the *reduced* model

Operator Inference

Peherstorfer & W. Data-driven operator inference for nonintrusive projection-based model reduction, *Computer Methods in Applied Mechanics and Engineering*, 2016

$$\widehat{\mathbf{E}}\dot{\widehat{\mathbf{x}}} = \widehat{\mathbf{A}}\widehat{\mathbf{x}} + \widehat{\mathbf{B}}\mathbf{u} + \widehat{\mathbf{H}}(\widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}})$$

Given reduced state data ($\widehat{\mathbf{X}}$) and velocity data ($\dot{\widehat{\mathbf{X}}}$):

$$\widehat{\mathbf{X}} = \begin{bmatrix} | & & | \\ \hat{\mathbf{x}}(t_1) & \dots & \hat{\mathbf{x}}(t_K) \\ | & & | \end{bmatrix} \qquad \dot{\widehat{\mathbf{X}}} = \begin{bmatrix} | & & | \\ \dot{\hat{\mathbf{x}}}(t_1) & \dots & \dot{\hat{\mathbf{x}}}(t_K) \\ | & & | \end{bmatrix}$$

Find the operators $\widehat{\mathbf{A}}, \widehat{\mathbf{B}}, \widehat{\mathbf{E}}, \widehat{\mathbf{H}}$
by solving the least squares problem:

$$\min_{\widehat{\mathbf{A}},\widehat{\mathbf{B}},\widehat{\mathbf{E}},\widehat{\mathbf{H}}} \left\| \widehat{\mathbf{X}}^\top \widehat{\mathbf{A}}^\top + \left(\widehat{\mathbf{X}} \otimes \widehat{\mathbf{X}}\right)^\top \widehat{\mathbf{H}}^\top + \mathbf{U}^\top \widehat{\mathbf{B}}^\top - \dot{\widehat{\mathbf{X}}}^\top \widehat{\mathbf{E}} \right\|$$

Under certain conditions, recovers the intrusive POD reduced model

Lift & Learn

Variable transformations to expose structure

+ non-intrusive learning that frees us to choose our variables

Learning a low-dimensional model

Using only snapshot data from the high-fidelity model (non-intrusive) but learning the POD reduced model

Lift & Learn [Qian, Kramer, Peherstorfer & W., 2019]

1. Generate full state trajectories (snapshots) (from high-fidelity simulation)

$$\mathbf{X_{orig}} = \begin{bmatrix} | & & | \\ \mathbf{x}(t_1) & \dots & \mathbf{x}(t_K) \\ | & & | \end{bmatrix} \quad \dot{\mathbf{X}}_{\mathbf{orig}} = \begin{bmatrix} | & & | \\ \dot{\mathbf{x}}(t_1) & \dots & \dot{\mathbf{x}}(t_K) \\ | & & | \end{bmatrix}$$

Learning a low-dimensional model

Using only snapshot data from the high-fidelity model (non-intrusive) but learning the POD reduced model

Lift & Learn [Qian, Kramer, Peherstorfer & W., 2019]

1. Generate full state trajectories (snapshots) (from high-fidelity simulation)
2. Transform snapshot data to get lifted snapshots (analyze the PDEs to expose system polynomial structure)

$$\mathbf{X}_{\mathbf{orig}} \longrightarrow \mathbf{X} \qquad \dot{\mathbf{X}}_{\mathbf{orig}} \longrightarrow \dot{\mathbf{X}}$$

Learning a low-dimensional model

Using only snapshot data from the high-fidelity model (non-intrusive) but learning the POD reduced model

Lift & Learn [Qian, Kramer, Peherstorfer & W., 2019]

1. Generate full state trajectories (snapshots) (from high-fidelity simulation)
2. Transform snapshot data to get lifted snapshots
3. Compute POD basis from lifted trajectories

$$\mathbf{X} = \mathbf{V}\,\boldsymbol{\Sigma}\,\mathbf{W}^{\top}$$

Learning a low-dimensional model

Using only snapshot data from the high-fidelity model (non-intrusive) but learning the POD reduced model

Lift & Learn [Qian, Kramer, Peherstorfer & W., 2019]

1. Generate full state trajectories (snapshots) (from high-fidelity simulation)
2. Transform snapshot data to get lifted snapshots
3. Compute POD basis from lifted trajectories
4. Project lifted trajectories onto POD basis, to obtain trajectories in low-dimensional POD coordinate space

$$\hat{\mathbf{X}} = \mathbf{V}^\top \mathbf{X}$$

Learning a low-dimensional model

Using only snapshot data from the high-fidelity model (non-intrusive) but learning the POD reduced model

Lift & Learn [Qian, Kramer, Peherstorfer & W., 2019]

1. Generate full state trajectories (snapshots) (from high-fidelity simulation)
2. Transform snapshot data to get lifted snapshots
3. Compute POD basis from lifted trajectories
4. Project lifted trajectories onto POD basis, to obtain trajectories in low-dimensional POD coordinate space
5. Solve least squares minimization problem to infer the low-dimensional model

$$\min_{\hat{\mathbf{A}},\hat{\mathbf{B}},\hat{\mathbf{E}},\hat{\mathbf{H}}} \left\| \hat{\mathbf{X}}^\top \hat{\mathbf{A}}^\top + \left(\hat{\mathbf{X}} \otimes \hat{\mathbf{X}}\right)^\top \hat{\mathbf{H}}^\top + \mathbf{U}^\top \hat{\mathbf{B}}^\top - \dot{\hat{\mathbf{X}}}^\top \hat{\mathbf{E}} \right\|$$

Learning a low-dimensional model

Using only snapshot data from the high-fidelity model (non-intrusive) but learning the POD reduced model

Lift & Learn [Qian, Kramer, Peherstorfer & W., 2019]

1. Generate full state trajectories (snapshots) (from high-fidelity simulation)
2. Transform snapshot data to get lifted snapshots
3. Compute POD basis from lifted trajectories
4. Project lifted trajectories onto POD basis, to obtain trajectories in low-dimensional POD coordinate space
5. Solve least squares minimization problem to infer the low-dimensional model

Under certain conditions, recovers the intrusive POD reduced model

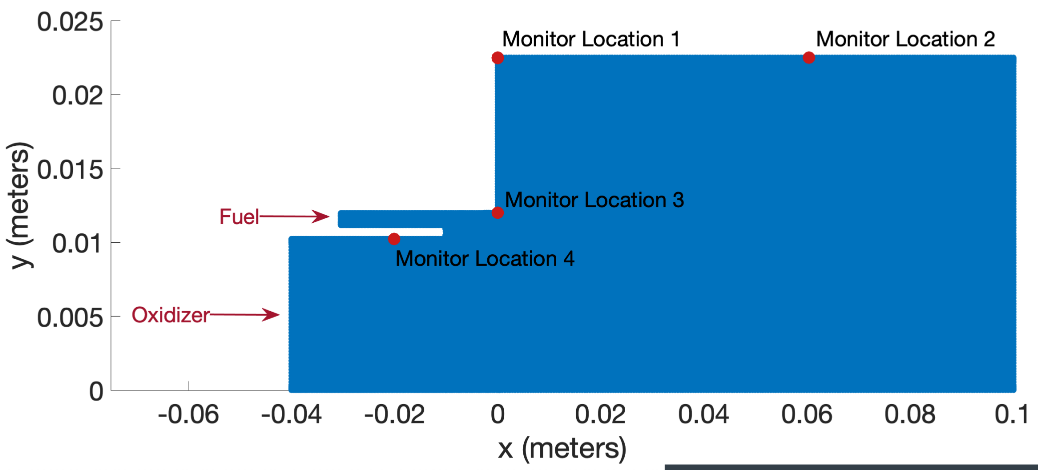
→ **convenience** of black-box learning +
rigor of projection-based reduction +
structure imposed by physics

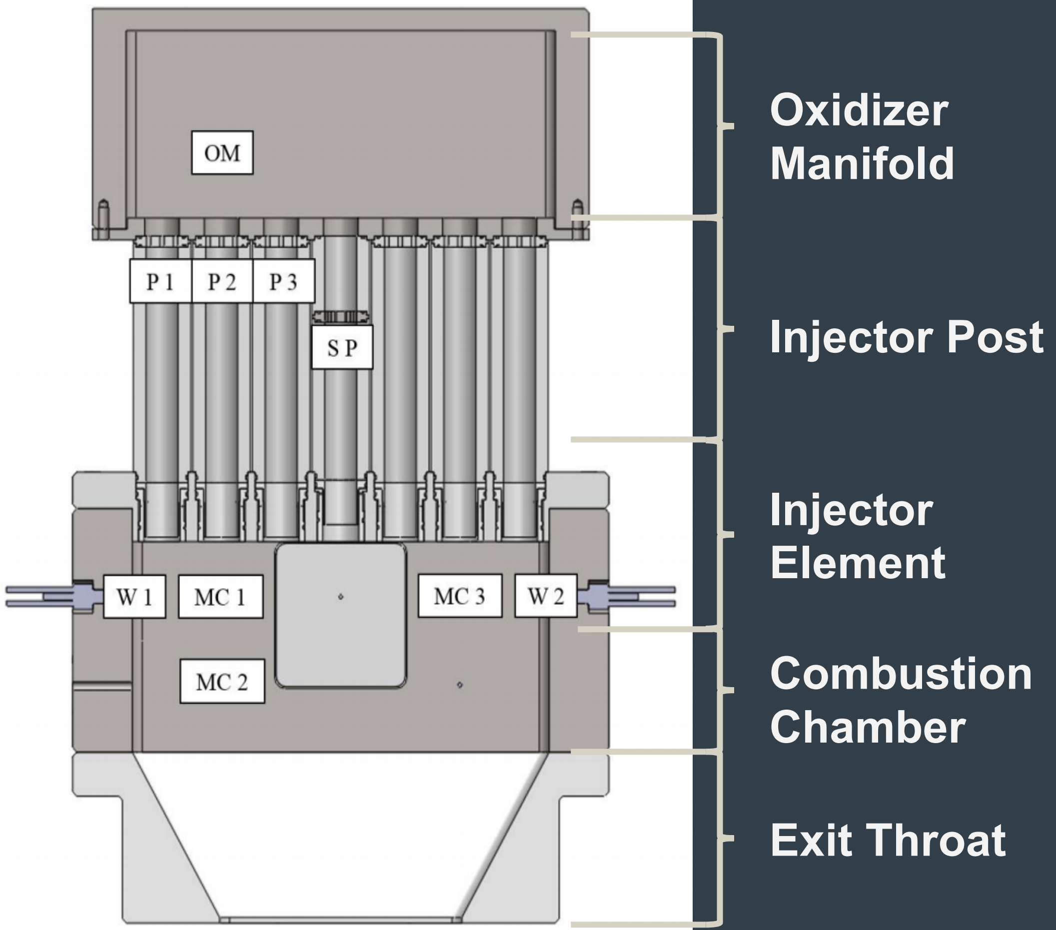
Rocket Engine Combustion

Lift & Learn reduced models for a complex Air Force combustion problem

Modeling a single injector of a rocket engine combustor

- Spatial domain discretized into 38,523 cells
- Pressure monitored at 4 locations
- Oxidizer input: 0.37 $\frac{kg}{s}$ of 42% O_2 / 58% H_2O
- Fuel input: 5.0 $\frac{kg}{s}$ of CH_4
- Governing equations: conservation of mass, momentum, energy, species
- Forced by a back pressure boundary condition at exit throat

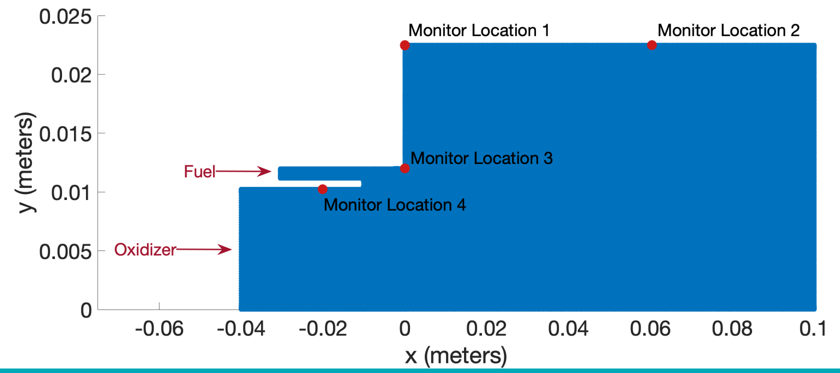




Modeling a single injector of a rocket engine combustor

Training data

- 1ms of full state solutions generated using Air Force GEMS code (~200 hours CPU time)
- Timestep $\Delta t = 10^{-7}$s; 10,000 total snapshots
- Variables used for learning ROMs
  $$\mathbf{x} = [\mathbf{p} \quad \mathbf{u} \quad \mathbf{v} \quad \mathbf{1/\rho} \quad \mathbf{Y_{CH_4}} \quad \mathbf{Y_{O_2}} \quad \mathbf{Y_{CO_2}} \quad \mathbf{Y_{H_2O}}]$$
  makes many (but not all) terms in governing equations quadratic
- Snapshot matrix $\mathbf{X} \in \mathbb{R}^{308,184 \times 10,000}$



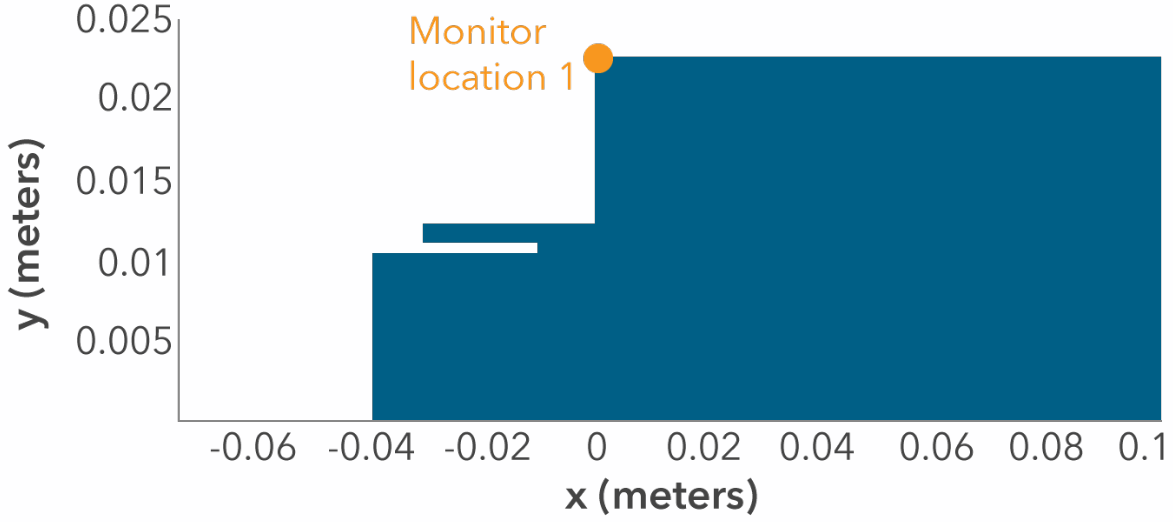
Test data

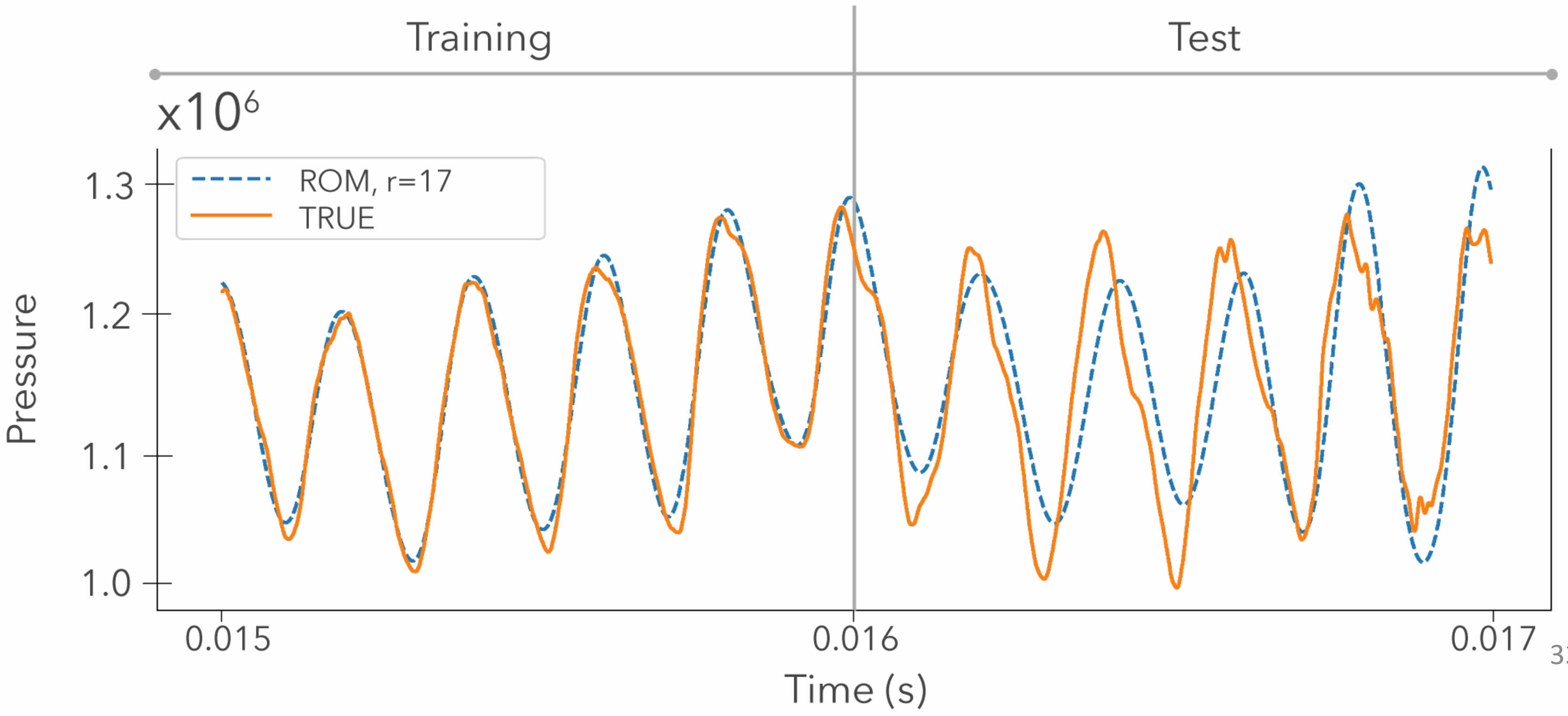
Additional 1 ms of data at monitor locations (10,000 timesteps)

Performance of learned quadratic ROM

Pressure time traces at monitor location 1

Basis size $r = 17$

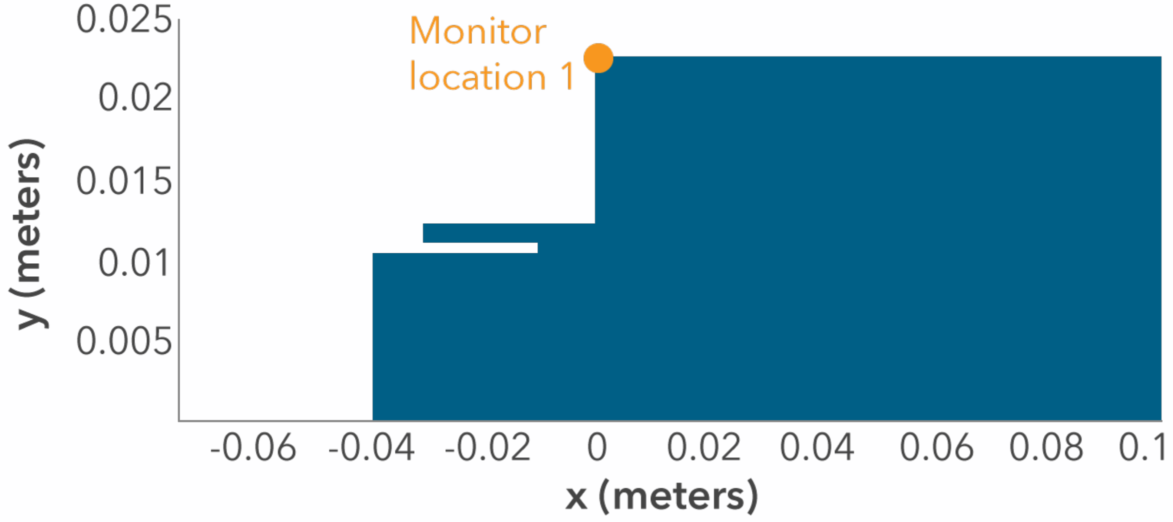


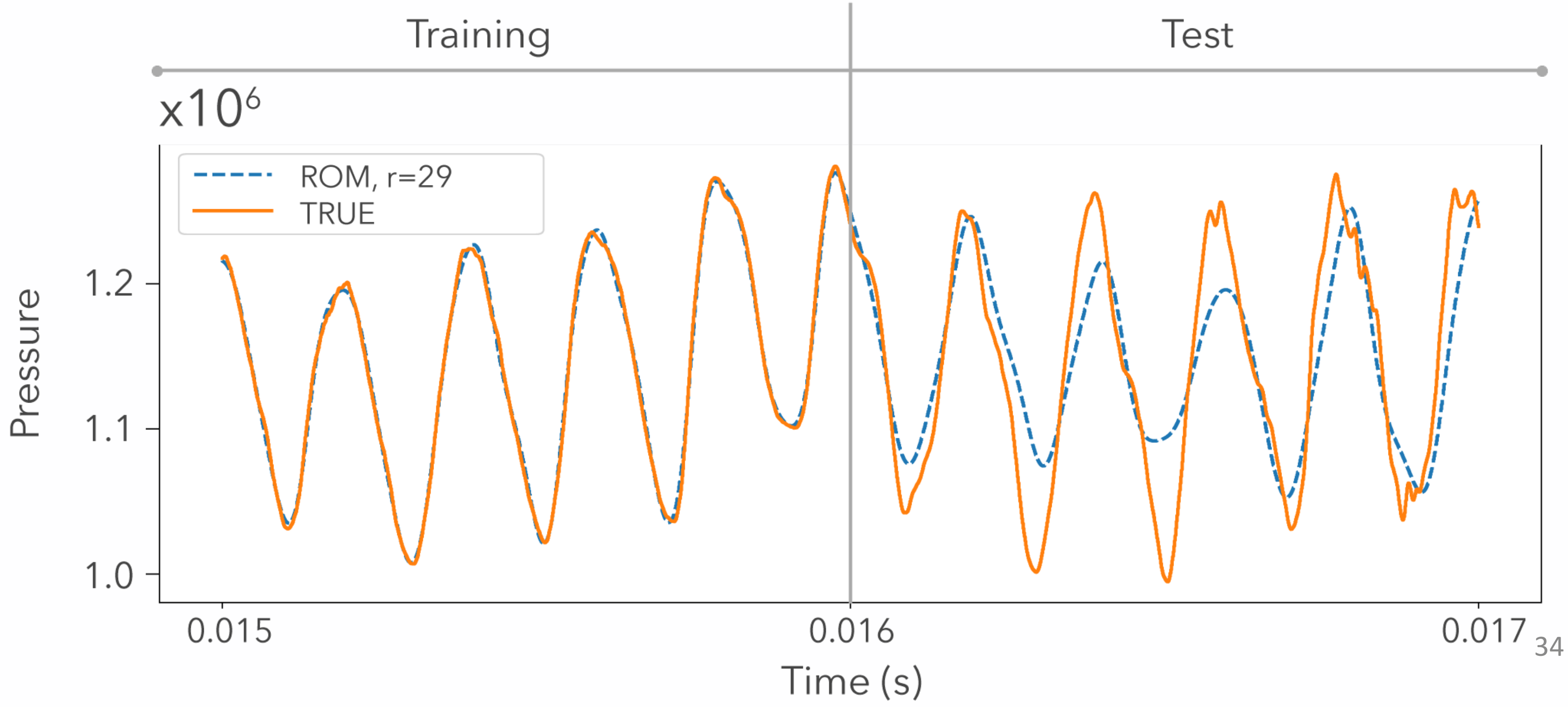


Performance of learned quadratic ROM

Pressure time traces at monitor location 1

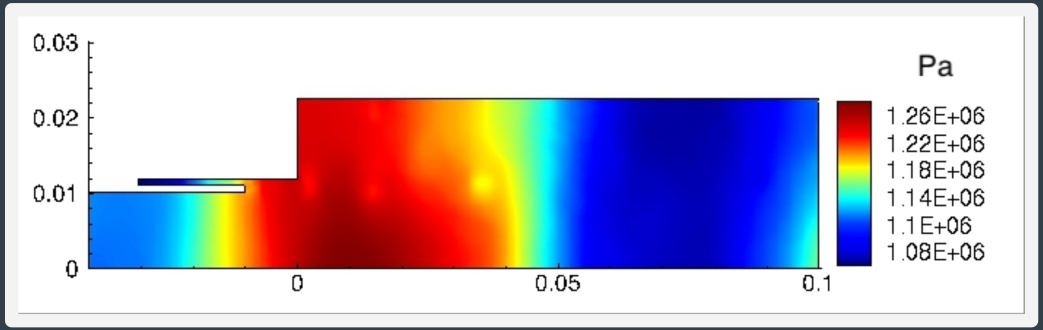
Basis size $r = 29$

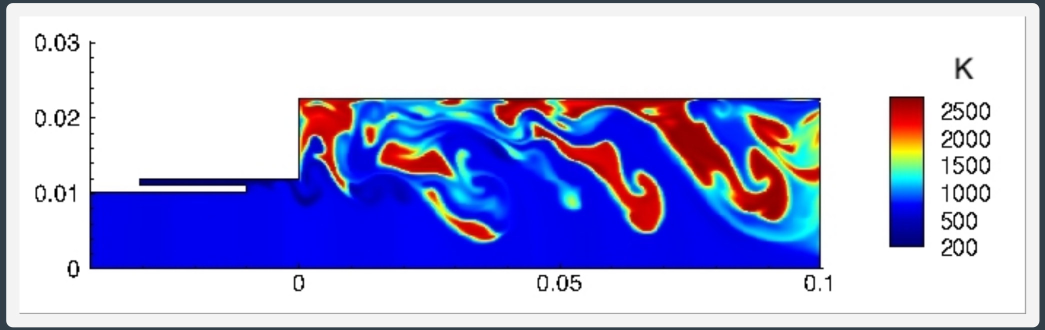




True

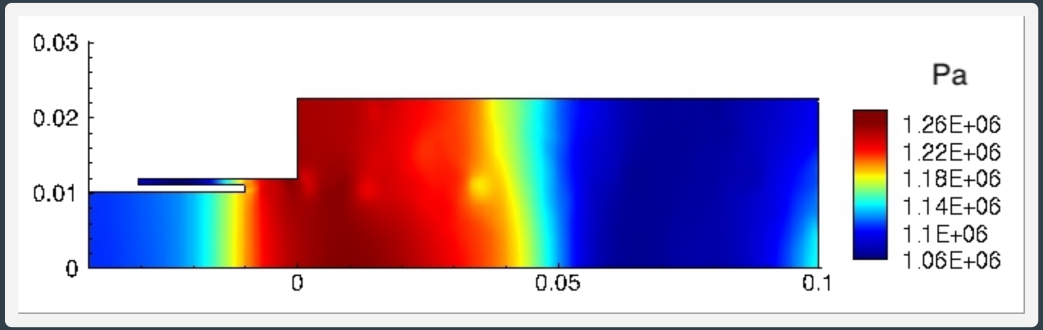
Pressure **Temperature**

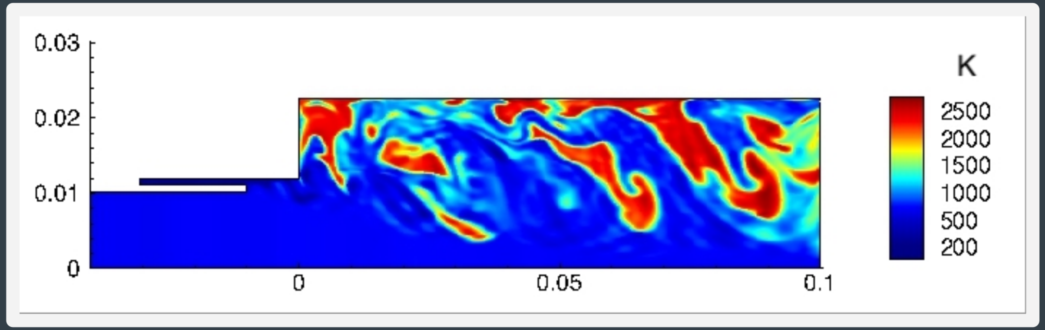




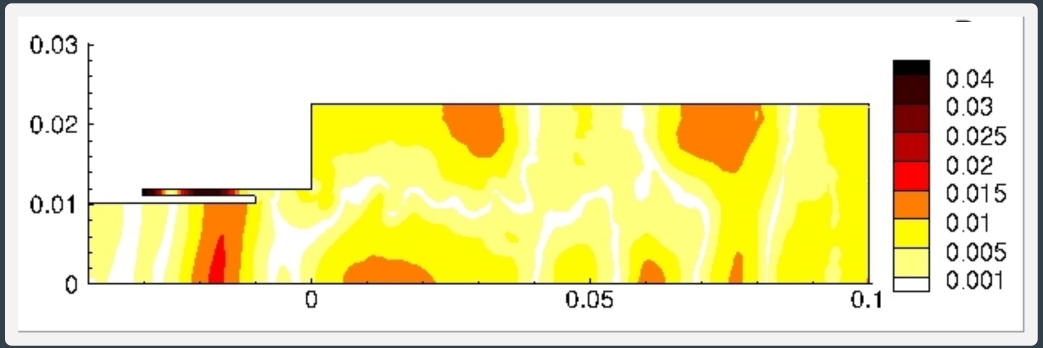
Predicted

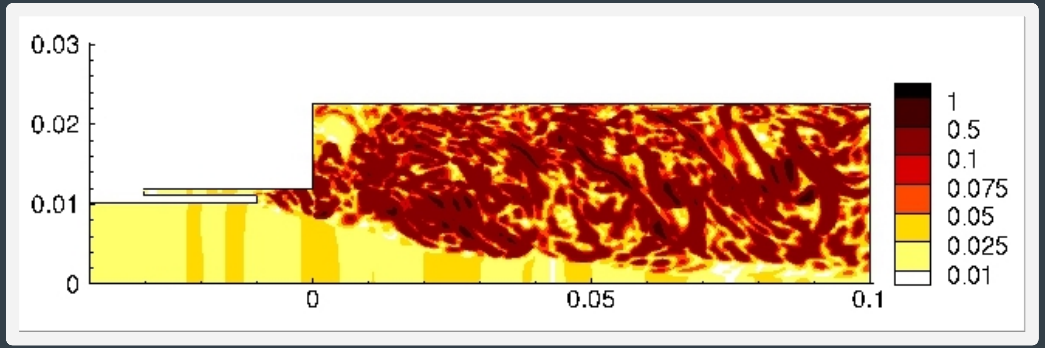
$r = 29$ POD modes





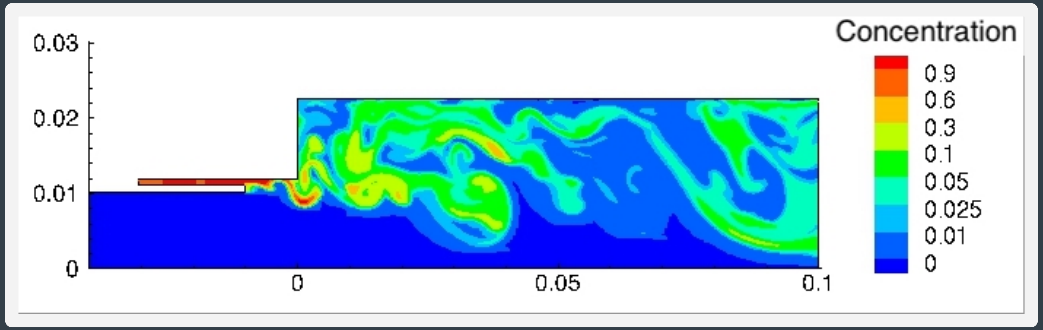
Relative error

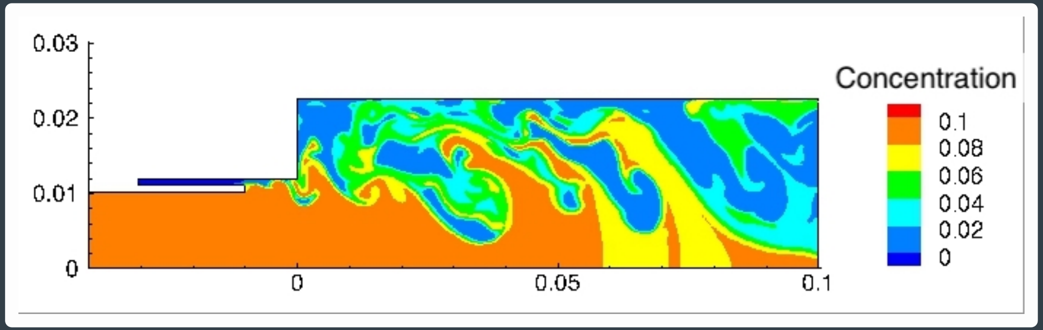




True

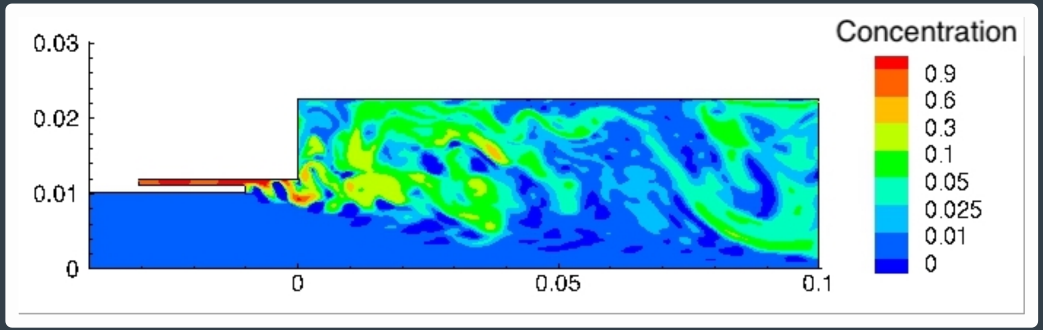
CH_4 O_2

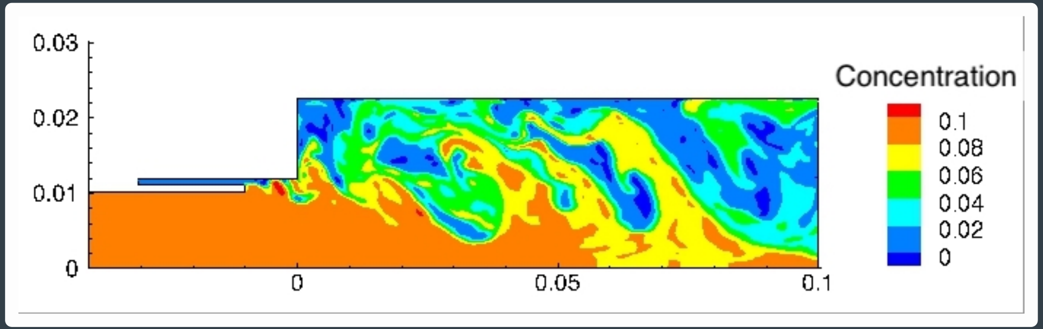




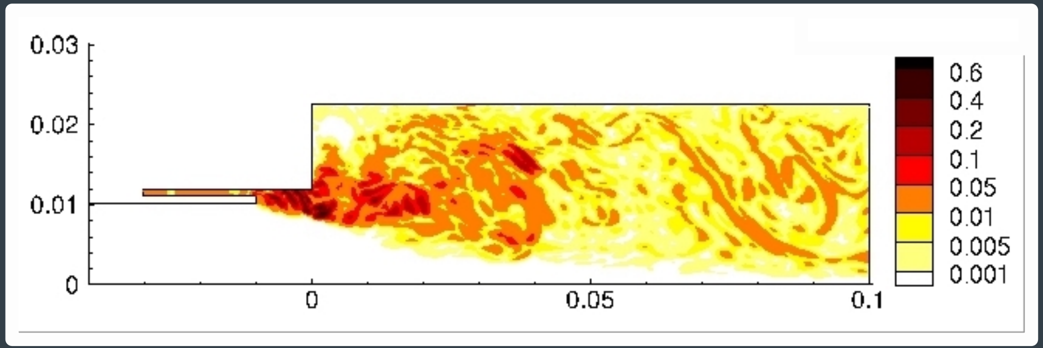
Predicted

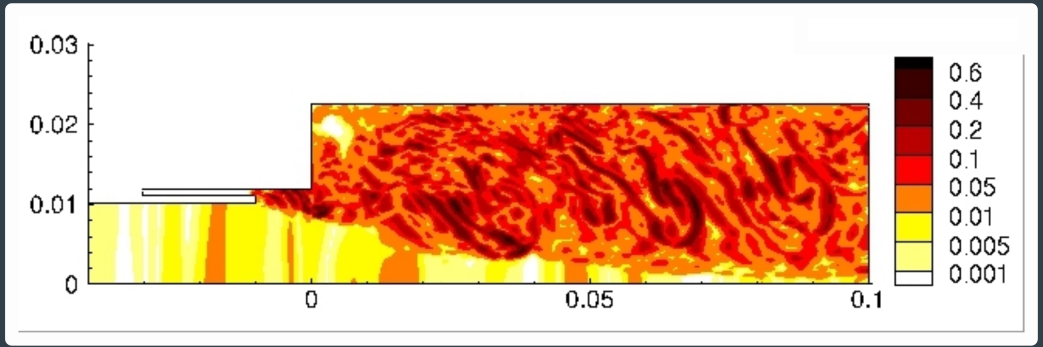
$r = 29$ POD modes





Normalized absolute error





Conclusions & Outlook

The future of Predictive Data Science

Data Science

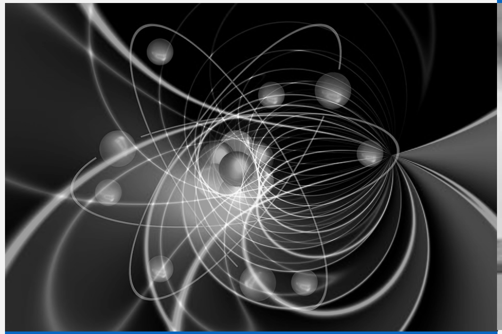
Computational Science & Engineering

Predictive Data Science

Revolutionizing decision-making for **high-consequence applications** in science, engineering & medicine

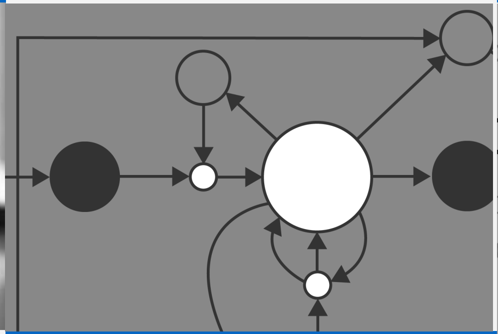
Predictive Data Science

Learning from data through the lens of models is a way to exploit structure in an otherwise intractable problem.



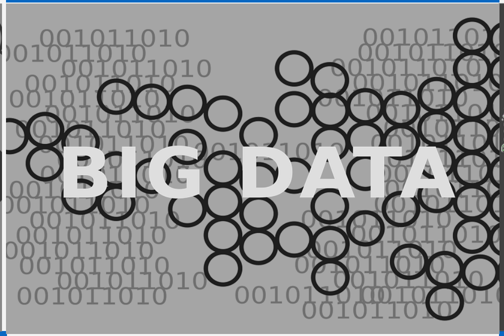
Respect physical constraints

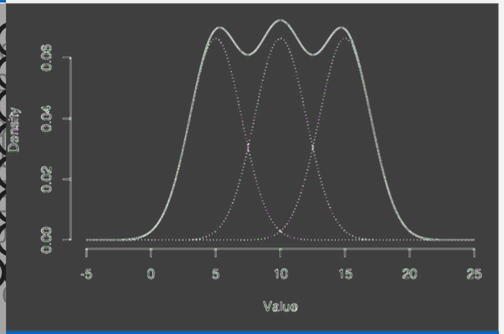
Embed domain knowledge



Bring interpretability to results

Integrate heterogeneous, noisy & incomplete data





Get predictions with quantified uncertainties

…

Predictive Data Science

Needs **interdisciplinary research** & **education** at the interfaces

1. Embedding domain knowledge
2. Learning from data through the lens of models
3. Principled approximations that exploit low-dimensional structure
4. Explicit modeling & treatment of uncertainty

Data-driven decisions

building the mathematical foundations and computational methods to enable design of the next generation of engineered systems

KIWI.ODEN.UTEXAS.EDU

ODEN INSTITUTE
FOR COMPUTATIONAL ENGINEERING & SCIENCES

Our papers on this topic

1. Peherstorfer, B. and Willcox, K., Data-driven operator inference for nonintrusive projection-based model reduction, *Computer Methods in Applied Mechanics and Engineering*, Vol. 306, pp. 196-215, 2016.

2. Kramer, B. and Willcox, K., Nonlinear model order reduction via lifting transformations and proper orthogonal decomposition, *AIAA Journal*, Vol. 57 No. 6, pp. 2297-2307, 2019.

3. Qian, E., Kramer, B., Marques, A. and Willcox, K., Transform & Learn: A data-driven approach to nonlinear model reduction. In Proceedings of AIAA Aviation Forum & Exhibition, Dallas, TX, June 2019.