The University of Texas at Austin
Institute for Computational Engineering and Sciences

MODEL ORDER REDUCTION

Approximate yet accurate surrogates for large-scale simulation

KAREN E. WILLCOX

Science at Extreme Scales:
Where Big Data Meets Large-Scale Computing Tutorials

Institute for Pure and Applied Mathematics

September 17, 2018

ICES

Tutorial Outline

These slides include contributions from many MIT postdocs and students, including B. Kramer, B. Peherstorfer, E. Qian, V. Singh

1. Motivation
2. General projection framework
3. Computing the basis
4. Approximating nonlinear terms
5. Error analysis and guarantees
6. Adaptive data-driven ROMs
7. Challenges

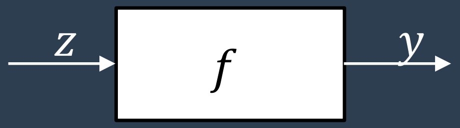
1. Motivation

Use cases and benefits of ROMs

Outer-loop applications

"Computational applications that form outer loops around a model – where in each iteration an input z is received and the corresponding model output $y = f(z)$ is computed, and an overall outer-loop result is obtained at the termination of the outer loop."

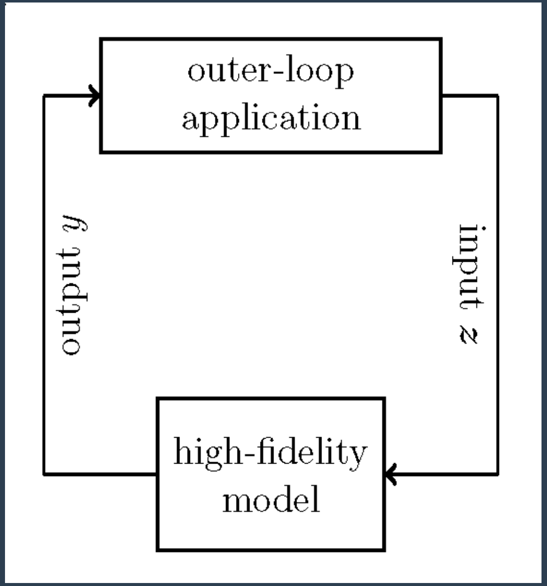
Peherstorfer, W., Gunzburger, *SIAM Review*, 2018



forward model

Examples

- Optimization
  outer-loop result = optimal design
- Uncertainty propagation
  outer-loop result = estimate of statistics of interest
- Inverse problems
- Data assimilation
- Control problems
- Sensitivity analysis

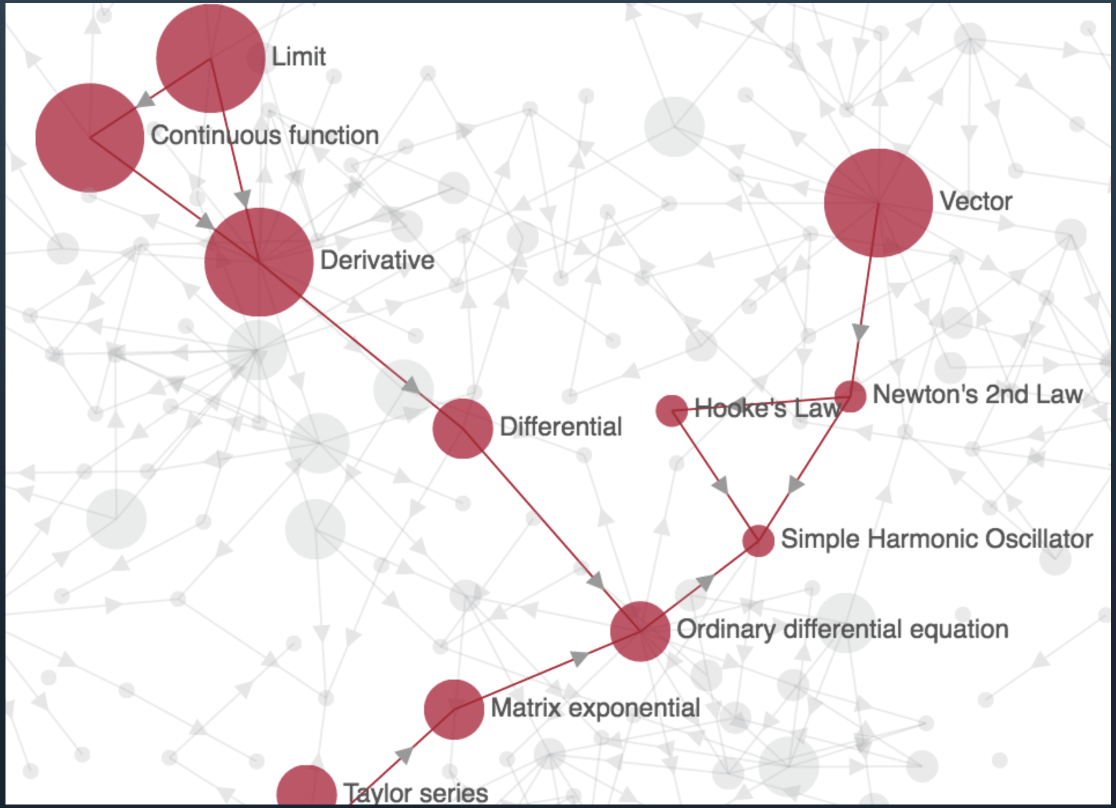


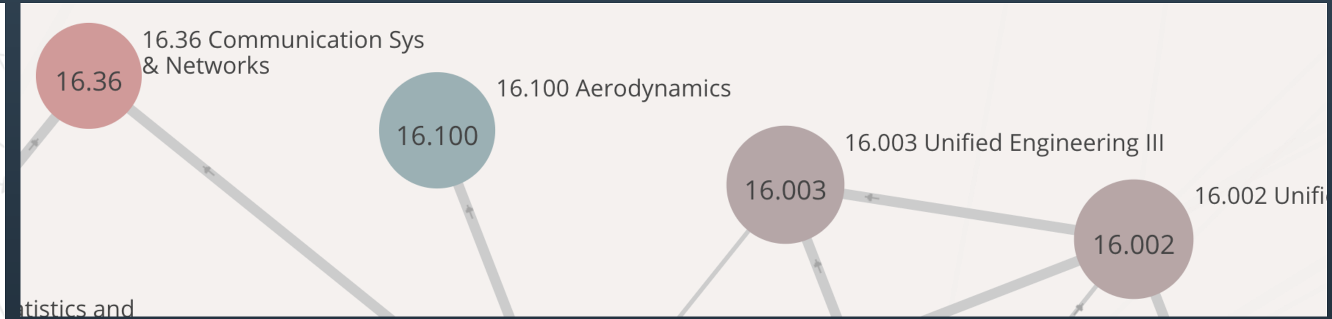
outer-loop application

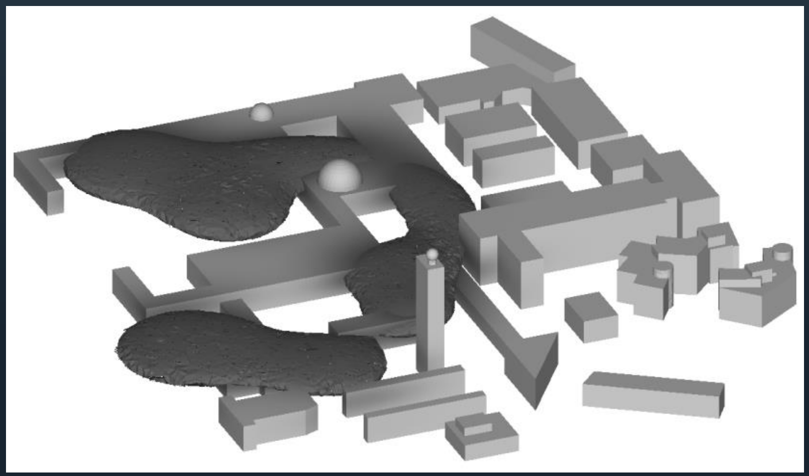
New Technologies + Data + Computational Power

a revolution in the world around us

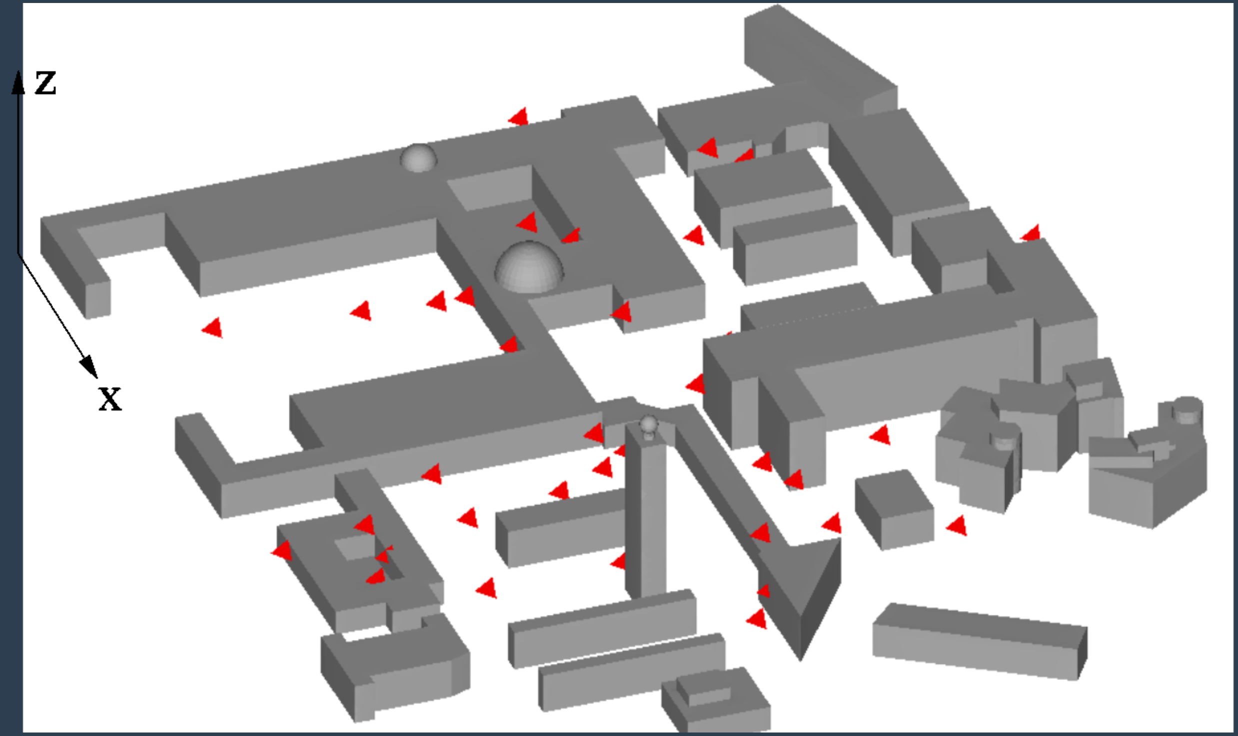
needing new data-enabled computational science and engineering

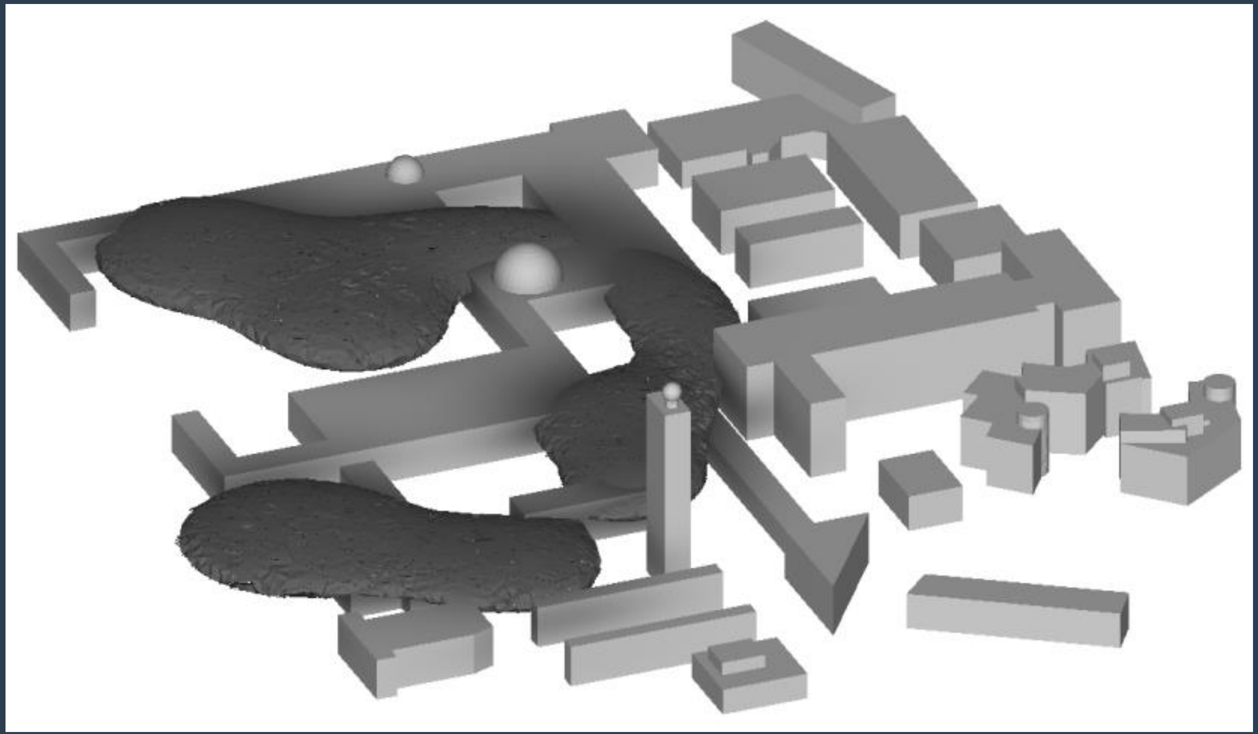






Data + Models:
real-time adaptive emergency response

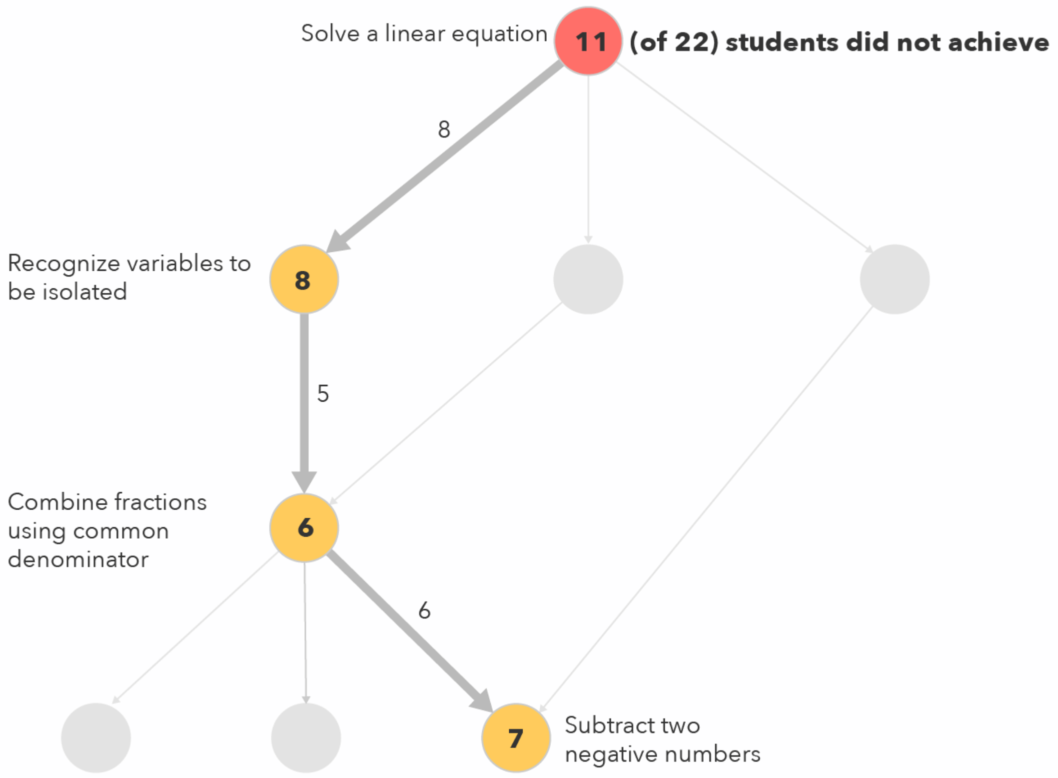


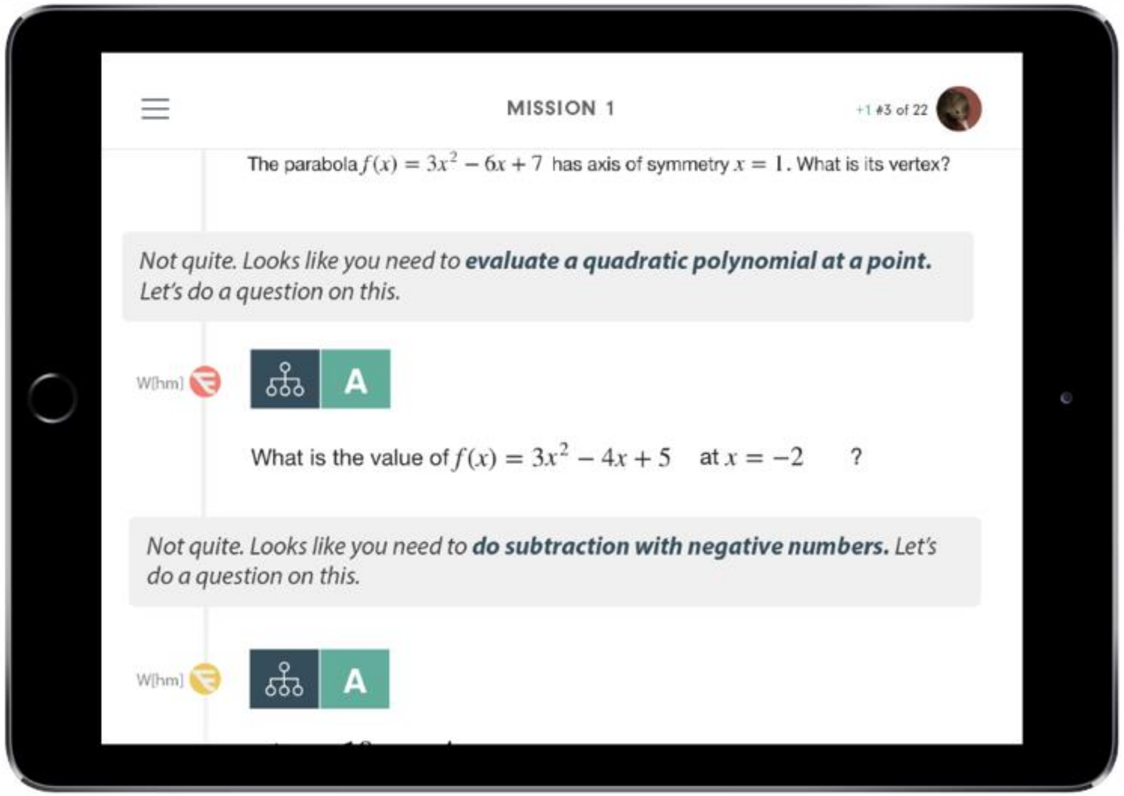


SENSE → **INFER** → **PREDICT** → **ACT**

Lieberman, Fidkowski, W., van Bloemen Waanders, *Int. J. Num. Meth. Fluids*, 2013

Data + Models:
real-time adaptive teaching & learning

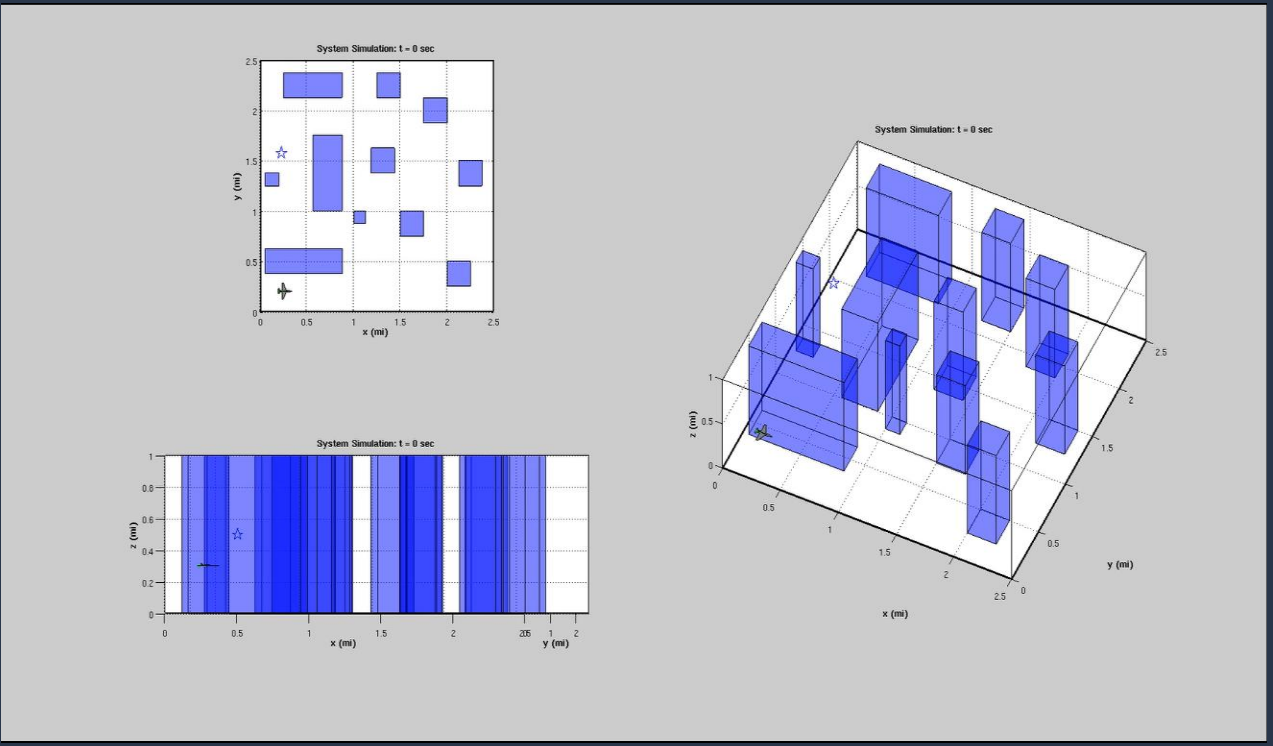


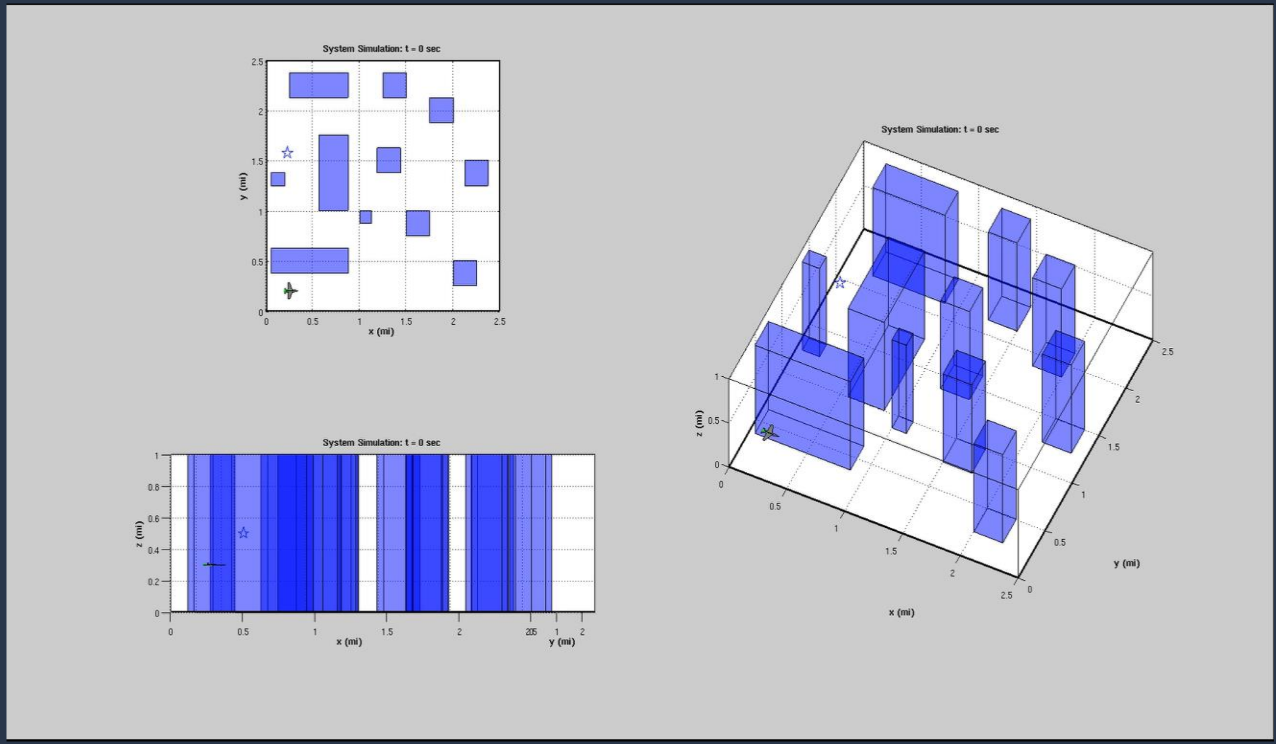


SENSE → **INFER** → **PREDICT** → **ACT**

U.S. Department of Education First in the World Fly-by-Wire project **fbw.mit.edu**

Data + Models:
self-aware aerospace vehicles





SENSE → **INFER** → **PREDICT** → **ACT**

Singh & W., *AIAA J.*, 2017

Model reduction leverages an **offline/online** decomposition of tasks

Offline

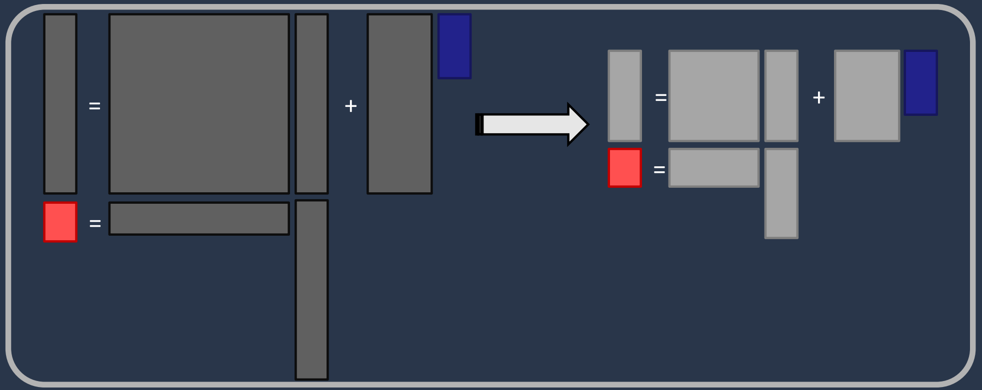
- Generate snapshots/libraries, using high-fidelity models
- Generate reduced models

Online

- Select appropriate library records and/or reduced models
- Rapid {prediction, control, optimization, UQ} using multi-fidelity models

Reduced models enable rapid **prediction**, **inversion**, **design**, and **uncertainty quantification** of large-scale scientific and engineering systems.

1 modeling the data-to-decisions flow **2** exploiting synergies between physics-based models & data **3** principled approximations to reduce computational cost **4** explicit modeling & treatment of uncertainty



2. Projection-based model reduction

extracting the essence of complex problems to make them easier and faster to solve

Start with a physics-based model

large-scale and expensive to solve

Arising, for example, from systems of ODEs or spatial discretization of PDEs describing the system of interest

- which in turn arise from governing physical principles (conservation laws, etc.)

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{p})\mathbf{x} + \mathbf{B}(\mathbf{p})\mathbf{u} \\ \mathbf{y} &= \mathbf{C}(\mathbf{p})\mathbf{x} \end{aligned}$$

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{p}, \mathbf{u}) \\ \mathbf{y} &= g(\mathbf{x}, \mathbf{p}, \mathbf{u}) \end{aligned}$$

$\mathbf{x} \in \mathbf{R}^N$: state vector

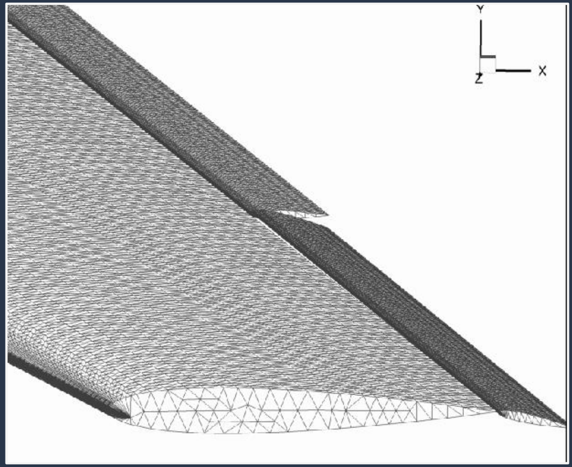
$\mathbf{u} \in \mathbf{R}^{N_i}$: input vector

$\mathbf{p} \in \mathbf{R}^{N_p}$: parameter vector

$\mathbf{y} \in \mathbf{R}^{N_o}$: output vector

Example: CFD systems

modeling the flow over an aircraft wing

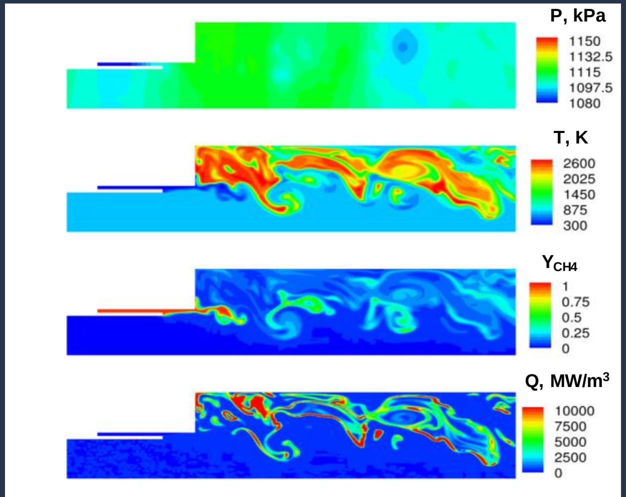


$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}(\mathbf{p})\mathbf{x} + \mathbf{B}(\mathbf{p})\mathbf{u} \\ \mathbf{y} &= \mathbf{C}(\mathbf{p})\mathbf{x}\end{aligned}$$

$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{p}, \mathbf{u}) \\ \mathbf{y} &= g(\mathbf{x}, \mathbf{p}, \mathbf{u})\end{aligned}$$

- $\mathbf{x}(t)$: vector of N flow unknowns
  e.g., 2D incompressible Navier Stokes
  P grid points, $N = 3P$
  $\mathbf{x} = [u_1\ v_1\ p_1\ u_2\ v_2\ p_2 \cdots\cdots u_P\ v_P\ p_P]^T$
- $\mathbf{p}$: input parameters
  e.g., shape parameters, PDE coefficients
- $\mathbf{u}(t)$: forcing inputs
  e.g., flow disturbances, wing motion
- $\mathbf{y}(t)$: outputs
  e.g., flow characteristic, lift force

Example: modeling combustion instability



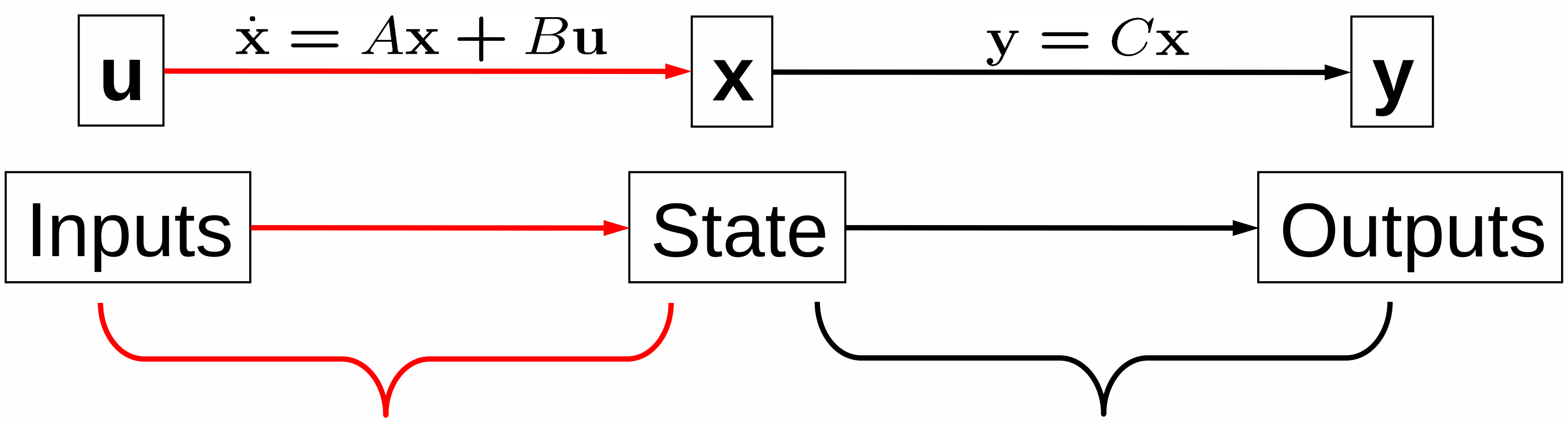
$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}(\mathbf{p})\mathbf{x} + \mathbf{B}(\mathbf{p})\mathbf{u} \\ \mathbf{y} &= \mathbf{C}(\mathbf{p})\mathbf{x}\end{aligned}$$

$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{p}, \mathbf{u}) \\ \mathbf{y} &= g(\mathbf{x}, \mathbf{p}, \mathbf{u})\end{aligned}$$

- $\mathbf{x}(t)$: vector of N reacting flow unknowns
  p', u', T', Y'_{ox} discretized over computational domain
- $\mathbf{p}$: input parameters
  e.g., fuel-to-oxidizer ratio, combustion zone length, fuel temperature, oxidizer temperature
- $\mathbf{u}(t)$: forcing inputs
  e.g., periodic oscillation of inlet mass flow rate, stagnation temperature, back pressure
- $\mathbf{y}(t)$: output quantities of interest
  e.g., pressure oscillation at sensor location

Which states are important?

Is there a low-dimensional structure underlying the input-output map?



"Controllable" modes
("Reachable" modes)

- easy to reach, require small control energy
- dominant eigenmodes of a controllability gramian matrix

"Observable" modes

- generate large output energy
- dominant eigenmodes of an observability gramian matrix

Which states are important?

Is there a low-dimensional structure underlying the input-output map?

- Rigorous theories and scalable algorithms in the linear time-invariant (LTI) case
  - Hankel singular values

- Strong foundations for linear parameter-varying (LPV) systems
  - handling high-dimensional parameters can be a challenge

- Many open questions for the nonlinear case
  - linear methods are founded on the notion of a low-dimensional subspace
  - works well for some nonlinear problems but certainly not all
  - additional challenges related to efficient solution of the ROM

Reduced models

low-cost but accurate approximations of high-fidelity models via projection onto a low-dimensional subspace

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{p})\mathbf{x} + \mathbf{B}(\mathbf{p})\mathbf{u} \\ \mathbf{y} &= \mathbf{C}(\mathbf{p})\mathbf{x} \end{aligned}$$

FOM

$\mathbf{x} \approx \mathbf{V}\mathbf{x}_r$

$$\begin{aligned} \mathbf{r} &= \mathbf{V}\dot{\mathbf{x}}_r - \mathbf{A}\mathbf{V}\mathbf{x}_r - \mathbf{B}\mathbf{u} \\ \mathbf{y}_r &= \mathbf{C}\mathbf{V}\mathbf{x}_r \end{aligned}$$

$\mathbf{W}^T\mathbf{r} = 0$

$$\begin{aligned} \mathbf{A}_r(\mathbf{p}) &= \mathbf{W}^T\mathbf{A}(\mathbf{p})\mathbf{V} \\ \mathbf{B}_r(\mathbf{p}) &= \mathbf{W}^T\mathbf{B}(\mathbf{p}) \\ \mathbf{C}_r(\mathbf{p}) &= \mathbf{C}(\mathbf{p})\mathbf{V} \end{aligned}$$

$$\begin{aligned} \dot{\mathbf{x}}_r &= \mathbf{A}_r(\mathbf{p})\mathbf{x}_r + \mathbf{B}_r(\mathbf{p})\mathbf{u} \\ \mathbf{y}_r &= \mathbf{C}_r(\mathbf{p})\mathbf{x}_r \end{aligned}$$

ROM

$\mathbf{x} \in \mathbf{R}^N$: state vector
$\mathbf{p} \in \mathbf{R}^{N_p}$: parameter vector
$\mathbf{u} \in \mathbf{R}^{N_i}$: input vector
$\mathbf{y} \in \mathbf{R}^{N_o}$: output vector

$\mathbf{x}_r \in \mathbf{R}^n$: reduced state vector
$\mathbf{V} \in \mathbf{R}^{N \times n}$: reduced basis

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

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]

The difference in fields is perhaps largely one of history and perspective: model reduction methods have grown from the scientific computing community, with a focus on *reducing* high-dimensional models that arise from physics-based modeling, whereas machine learning has grown from the computer science community, 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]

3. Computing the basis

Many different methods to identify the low-dimensional subspace

(Some) Large-Scale Reduction Methods

Different mathematical foundations lead to different ways to compute the basis and the reduced model

Overview in Benner, Gugercin & Willcox, *SIAM Review*, 2015

- **Proper orthogonal decomposition** (POD) *(Lumley, 1967; Sirovich, 1981; Berkooz, 1991; Deane et al. 1991; Holmes et al. 1996)*
  - use data to generate empirical eigenfunctions
  - time- and frequency-domain methods
- **Krylov-subspace** methods *(Gallivan, Grimme, & van Dooren, 1994; Feldmann & Freund, 1995; Grimme, 1997, Gugercin et al., 2008)*
  - rational interpolation
- **Balanced truncation** *(Moore, 1981; Sorensen & Antoulas, 2002; Li & White, 2002)*
  - guaranteed stability and error bound for LTI systems
  - close connection between POD and balanced truncation
- **Reduced basis** methods *(Noor & Peters, 1980; Patera & Rozza, 2007)*
  - strong focus on error estimation for specific PDEs
- **Eigensystem realization algorithm** (ERA) *(Juang & Pappa, 1985)*, **Dynamic mode decomposition** (DMD) *(Schmid, 2010)*, **Loewner** model reduction *(Mayo & Antoulas, 2007)*
  - data-driven, non-intrusive

Computing the Basis: Proper Orthogonal Decomposition (POD)

(aka Karhunen-Loève expansions, Principal Components Analysis, Empirical Orthogonal Eigenfunctions, …)

- Consider *K* **snapshots** $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K \in \mathcal{R}^N$ [Sirovich, 1991] (solutions at selected times or parameter values)
- Form the snapshot matrix $X = [\mathbf{x}_1 \ \mathbf{x}_2 \ \ldots \ \mathbf{x}_K]$
- Choose the *n* basis vectors $\mathbf{V} = [\mathbf{V}_1 \ \mathbf{V}_2 \cdots \mathbf{V}_n]$ to be left **singular vectors** of the snapshot matrix, with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq \sigma_{n+1} \geq \cdots \geq \sigma_K$
- This is the optimal projection in a least squares sense:

$$\min_{\mathbf{V}} \sum_{i=1}^{K} ||\mathbf{x}_i - \mathbf{V}\mathbf{V}^T\mathbf{x}_i||_2^2 = \sum_{i=n+1}^{K} \sigma_i^2$$

4. Nonlinear model reduction

General projection framework applies, but leads to complications

Projection-based nonlinear reduced models

approximation of high-fidelity models via projection onto a low-dimensional subspace

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{p}, \mathbf{u}) \\ \mathbf{y} &= g(\mathbf{x}, \mathbf{p}, \mathbf{u}) \end{aligned}$$

FOM

$\mathbf{x} \in \mathbf{R}^N$: state vector
$\mathbf{p} \in \mathbf{R}^{N_p}$: parameter vector
$\mathbf{u} \in \mathbf{R}^{N_i}$: input vector
$\mathbf{y} \in \mathbf{R}^{N_o}$: output vector

$\mathbf{x} \approx \mathbf{V}\mathbf{x}_r$

$$\begin{aligned} \mathbf{r} &= \mathbf{V}\dot{\mathbf{x}}_r - f(\mathbf{V}\mathbf{x}_r, \mathbf{p}, \mathbf{u}) \\ \mathbf{y_r} &= g(\mathbf{V}\mathbf{x}_r, \mathbf{p}, \mathbf{u}) \end{aligned}$$

$\mathbf{x}_r \in \mathbf{R}^n$: reduced state vector
$\mathbf{V} \in \mathbf{R}^{N \times n}$: reduced basis

$\mathbf{W}^T\mathbf{r} = 0$

$$\begin{aligned} \dot{\mathbf{x}}_r &= \mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r, \mathbf{u}) \\ \mathbf{y}_r &= \mathbf{g}(\mathbf{V}\mathbf{x}_r) \end{aligned}$$

ROM

dimension is reduced, but evaluating nonlinear term still scales with large dimension N

Nonlinear POD ROMs

For nonlinear systems, standard POD projection approach leads to a model that is low order but still expensive to solve

FOM

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} &= \mathbf{g}(\mathbf{x})\end{aligned}$$

$\mathbf{x} = \mathbf{V}\mathbf{x}_r$ →

ROM

$$\begin{aligned}\dot{\mathbf{x}}_r &= \mathbf{V}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r, \mathbf{u}) \\ \mathbf{y}_r &= \mathbf{g}(\mathbf{V}\mathbf{x}_r)\end{aligned}$$

- The cost of evaluating the nonlinear term

$$\mathbf{f}_r(\mathbf{x}_r, \mathbf{u}) = \mathbf{V}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r, \mathbf{u})$$

  still depends on *N*, the dimension of the large-scale system

- Can achieve efficient nonlinear reduced models via interpolation, e.g., (**Discrete**) **Empirical Interpolation Method** [Barrault et al., 2004; Chaturantabut & Sorensen, 2010], **Missing Point Estimation** [Astrid et al., 2008], **GNAT** [Carlberg et al., 2013]

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

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\mathbf{u}) \\ \mathbf{y} &= \mathbf{g}(\mathbf{x})\end{aligned} \quad \xrightarrow{\mathbf{x}=\mathbf{V}\mathbf{x}_r} \quad \begin{aligned}\dot{\mathbf{x}}_r &= \mathbf{V}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r,\mathbf{u}) \\ \mathbf{y}_r &= \mathbf{g}(\mathbf{V}\mathbf{x}_r)\end{aligned}$$

Discrete Empirical Interpolation Method (DEIM)

Additional layer of approximation to make the reduced-order nonlinear term fast to evaluate

Chaturantabut & Sorensen, *SISC*, 2010

- Collect snapshots of $\mathbf{f}(\mathbf{x},\mathbf{u})$; compute DEIM basis $\mathbf{U}$ for the nonlinear term (use POD to identify a linear subspace)
- Select m interpolation points in $\mathbf{P} \in \mathbb{R}^{m\times N}$ at which to sample $\mathbf{f}$
- Approximate $\mathbf{f}_r(\mathbf{x}_r,\mathbf{u})$:

$$\mathbf{V}^T\mathbf{f}(\mathbf{V}\mathbf{x}_\mathrm{r},\mathbf{u}) \approx \underbrace{\mathbf{V}^T\mathbf{U}(\mathbf{P}^\mathrm{T}\mathbf{U})^{-1}}_{\substack{n\times m \\ \text{(precompute)}}}\underbrace{\mathbf{P}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r,\mathbf{u})}_{\substack{\text{evaluate just} \\ m \text{ entries of } \mathbf{f}}}$$

- Considerable success on a range of problems
- But some open challenges
  - for strongly nonlinear systems, require **so many DEIM points** that ROM is inefficient (e.g., Huang et al., AIAA 2018)
  - introduces **additional approximation**; **difficult to analyze** error convergence, stability, etc.

Linear Model

FOM: $$\mathbf{E}\dot{\mathbf{x}} = \underbrace{\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}}_{\text{linear}}$$

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

Precompute the ROM matrices:

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

Quadratic Model

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

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

Precompute the ROM matrices and tensor:

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

Quadratic-bilinear (QB) systems

Advantages:

- efficient offline/online decomposition
- amenable to analysis (errors, stability, etc.)

FOM:

$$\mathbf{E}\dot{\mathbf{x}} = \underbrace{\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}}_{\text{linear}} + \underbrace{\mathbf{H}(\mathbf{x} \otimes \mathbf{x})}_{\text{quadratic}} + \underbrace{\sum_{k=1}^{m} \mathbf{N}_k \mathbf{x} u_k}_{\text{bilinear}}$$

- Quadratic tensor $\mathbf{H} \in \mathbb{R}^{n \times n^2}$
- Bilinear interaction: $\mathbf{N}_k \in \mathbb{R}^{n \times n}, \;\; k = 1, \ldots, m$

ROM:

$$\widehat{\mathbf{E}}\dot{\widehat{\mathbf{x}}} = \widehat{\mathbf{A}}\widehat{\mathbf{x}} + \widehat{\mathbf{B}}\mathbf{u} + \widehat{\mathbf{H}}(\widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}}) + \sum_{k=1}^{m} \widehat{\mathbf{N}}_k \widehat{\mathbf{x}} u_k$$

$$\widehat{\mathbf{A}} = \mathbf{V}^\top \mathbf{A}\mathbf{V} \qquad \widehat{\mathbf{N}}_k = \mathbf{V}^\top \mathbf{N}_k \mathbf{V}$$

$$\widehat{\mathbf{B}} = \mathbf{V}^\top \mathbf{B} \qquad \widehat{\mathbf{H}} = \mathbf{V}^\top \mathbf{H}(\mathbf{V} \otimes \mathbf{V})$$

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

Polynomial systems

Could keep going to higher order

Model becomes more complex but retains efficient offline/online decomposition

FOM:

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

$$+ \underbrace{\mathbf{G}^{(3)}(\mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x})}_{\text{cubic}} + \underbrace{\mathbf{G}^{(4)}(\mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x})}_{\text{quartic}}$$

$$+ \underbrace{\sum_{k=1}^{m} \mathbf{N}_k^{(1)} \mathbf{x} u_k}_{\text{bilinear}} + \underbrace{\sum_{k=1}^{m} \mathbf{N}_k^{(2)} (\mathbf{x} \otimes \mathbf{x}) u_k}_{\text{quadratic-linear}}$$

ROM:

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

$$+ \widehat{\mathbf{G}}^{(3)}(\widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}}) + \widehat{\mathbf{G}}^{(4)}(\widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}})$$

$$+ \sum_{k=1}^{m} \widehat{\mathbf{N}}_k^{(1)} \widehat{\mathbf{x}} u_k + \sum_{k=1}^{m} \widehat{\mathbf{N}}_k^{(2)} (\widehat{\mathbf{x}} \otimes \widehat{\mathbf{x}}) u_k$$

Possibility to pre-compute reduced tensors is major advantage

$$\widehat{\mathbf{G}}^{(4)} = \mathbf{V}^\top \mathbf{G}^{(4)} (\mathbf{V} \otimes \mathbf{V} \otimes \mathbf{V} \otimes \mathbf{V})$$

$$\widehat{\mathbf{G}}^{(3)} = \mathbf{V}^\top \mathbf{G}^{(3)} (\mathbf{V} \otimes \mathbf{V} \otimes \mathbf{V})$$

5. Error analysis and guarantees

(or lack thereof)

Error analysis and guarantees

What rigorous statements can we make about the quality of the reduced-order models?

- Strong theoretical foundations in the LTI case (error bounds, error estimators)
- Solid theoretical foundations for some classes of linear parametrized PDEs (error estimators)
- Error indicators may be available (e.g., residual)
- Few/no guarantees available otherwise
- Nonlinear systems are a particular challenge
- Many important open research questions

Error analysis and guarantees

What rigorous statements can we make about the quality of the reduced-order models?

- POD

  Hinze M. and Volkwein, S. Error estimates for abstract linear-quadratic optimal control problems using proper orthogonal decomposition, Comput. Optim. Appl., 39 (2008), pp. 319–345.

- Reduced basis method has a strong focus on error estimates that exploit underlying structure of the PDE

  Elliptic PDES:

  Patera, A. and Rozza, G. *Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations*, Version 1.0, MIT, Cambridge, MA, 2006.

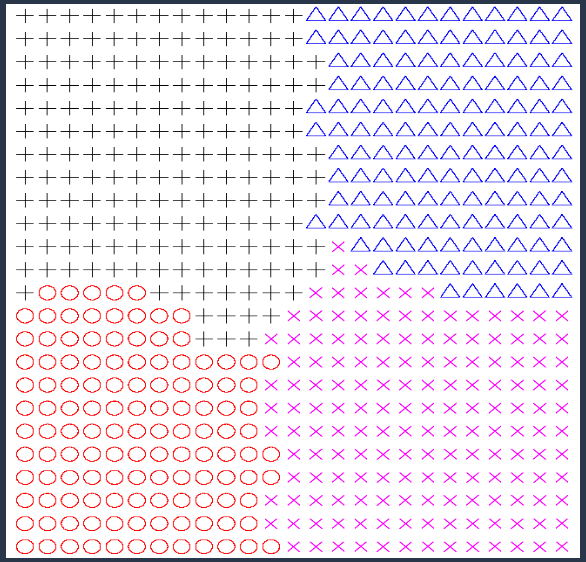
  Prud'homme, C., Rovas, D., Veroy, K., Maday, Y., Patera, A. and Turinici, G. Reliable real-time solution of parameterized partial differential equations: Reduced-basis output bound methods, *J. Fluids Engrg*., 124 (2002), pp. 70–80.

  Veroy, K., Prud'homme, C., Rovas, D., and Patera, A. (2003). A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. AIAA Paper 2003-3847, Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, Orlando, FL.

  Veroy, K. and Patera, A. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: Rigorous reduced-basis a posteriori error bounds, *Internat. J. Numer. Methods Fluids*, 47 (2005), pp. 773–788.

  Parabolic PDES:

  Grepl, M. and Patera, A. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations, *M2AN Math. Model. Numer. Anal*., 39 (2005), pp. 157–181.



6. Adaptive and Data-driven ROMs

Towards effective, efficient ROMs for a broader class of complex systems

Model reduction leverages an **offline/online** decomposition of tasks

Offline

- Generate snapshots/libraries, using high-fidelity models
- Generate reduced models

Online

- Select appropriate library records and/or reduced models
- Rapid {prediction, control, optimization, UQ} using multi-fidelity models

Classically

- Reduced models are built and used in a **static** way:
  - offline phase: sample a high-fidelity model, build a low-dimensional basis, project to build the reduced model
  - online phase: use the reduced model

Data-driven reduced models

- Recognize that conditions may change and/or initial reduced model may be inadequate
  - offline phase: build an initial reduced model
  - online phase: **learn** and **adapt** using dynamic data

A data-driven **offline/online** approach

Offline

- Generate snapshots/libraries, using high-fidelity models
- Generate reduced models

models

Online

- Dynamically collect data from sensors/simulations
- Classify system behavior
- Select appropriate library records and/or reduced models
- Rapid {prediction, control, optimization, UQ} using multi-fidelity models
- Adapt reduced models
- Adapt sensing strategies

models
+
data

Data-driven reduced models

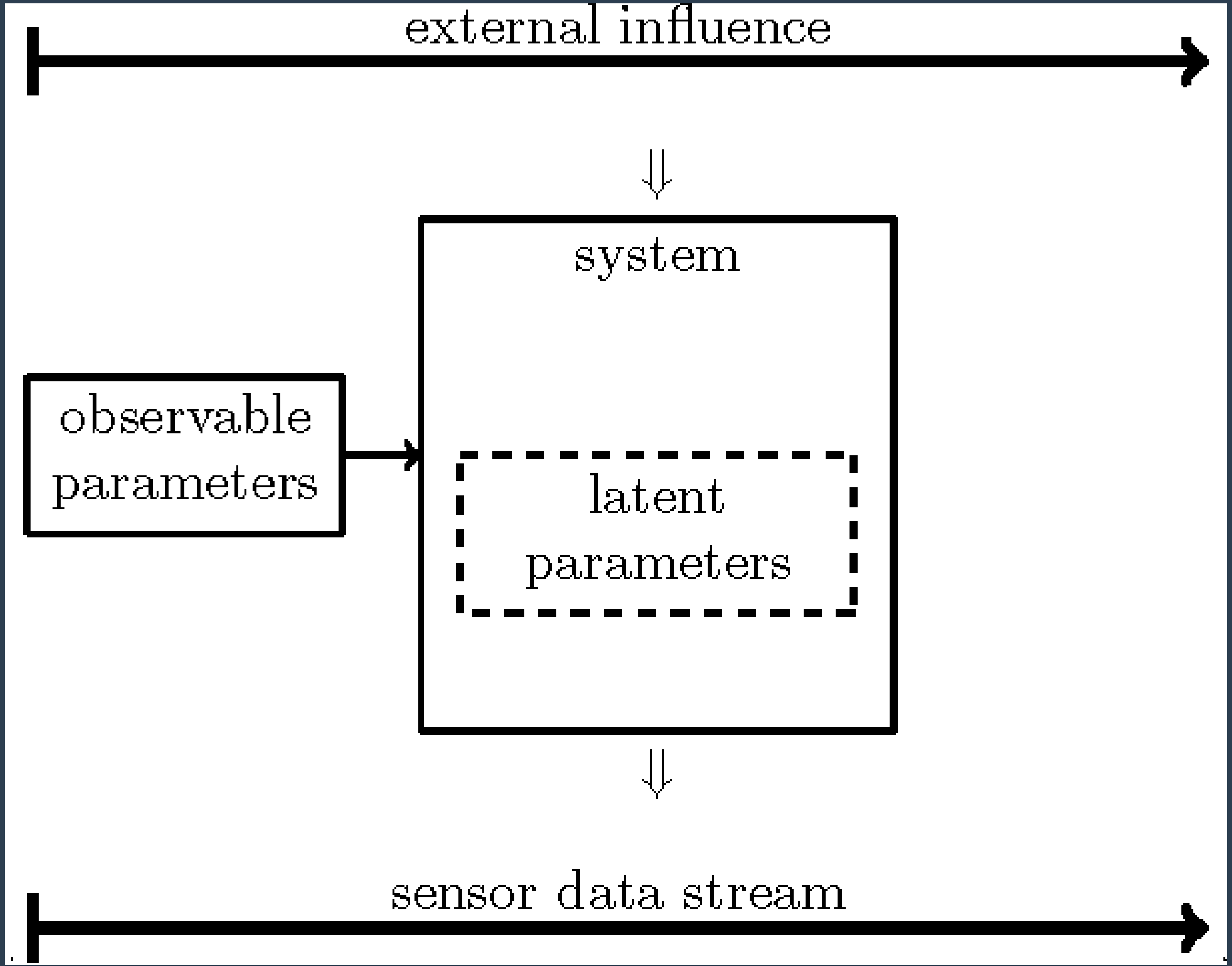
exploiting the synergies of physics-based models and dynamic data

- **Adaptation** and **learning** are **data-driven**
  - sensor data collected online
    (e.g., structural sensors on board an aircraft)
  - simulation data collected online
    (e.g., over the path to an optimal solution)

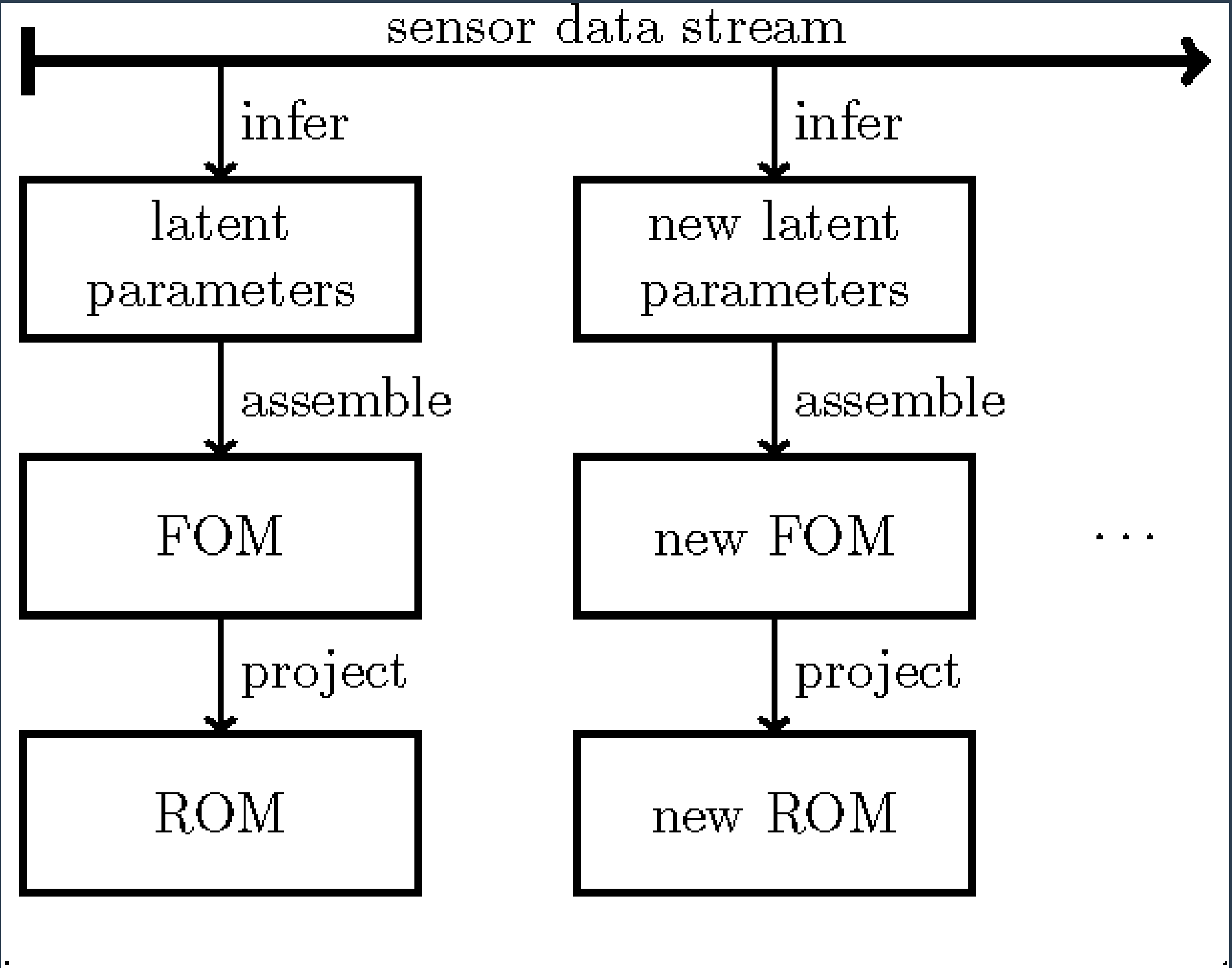
but the **physics-based model** remains as an underpinning.

- Achieve **adaptation** in a variety of ways:
  - adapt the basis *(Cui, Marzouk, W., 2014)*
  - adapt the way in which nonlinear terms are approximated *(***ADEIM***: Peherstorfer, W., 2015)*
  - adapt the reduced model itself *(Peherstorfer, W., 2015)*
  - construct localized reduced models; adapt model choice *(***LDEIM***: Peherstorfer, Butnaru, W., Bungartz, 2014)*

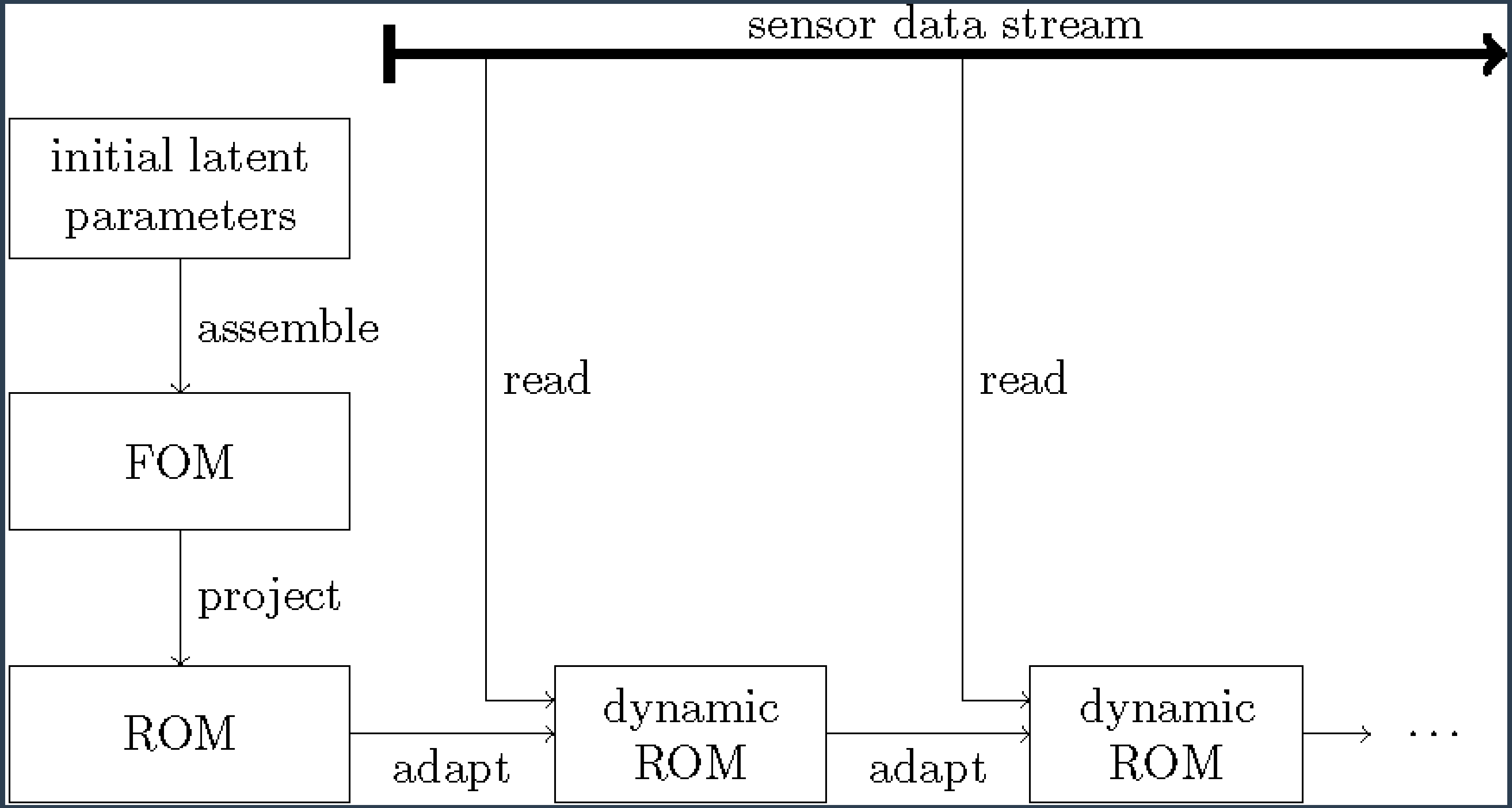
Consider a system with **observable** and **latent parameters**



Classical approaches build the new reduced model from scratch

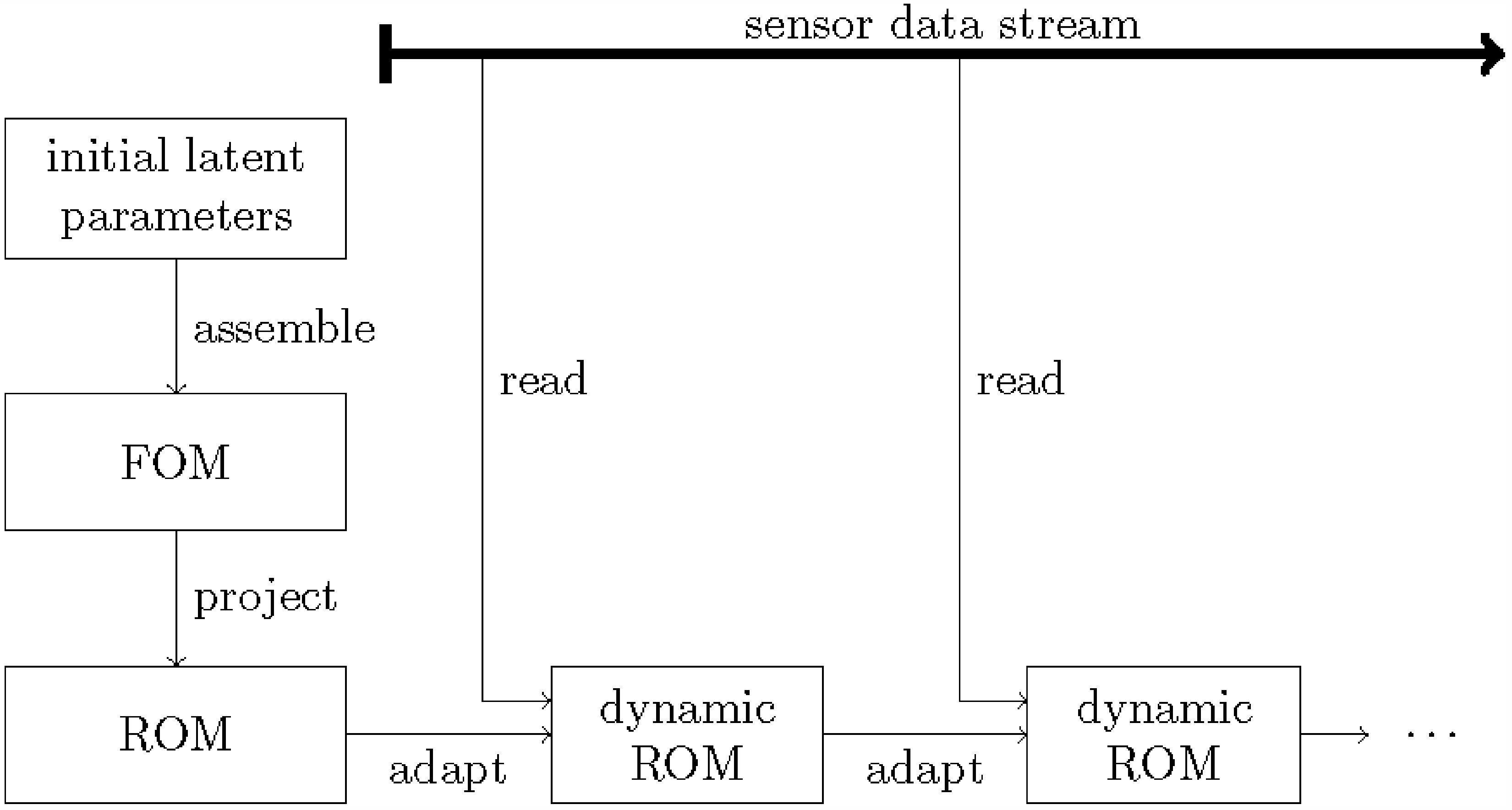


A **dynamic reduced model** adapts in response to the data, without recourse to the full model



Data-driven reduced models

- **adapt** directly from sensor data
- **avoid** (expensive) inference of latent parameter
- **avoid** recourse to full model

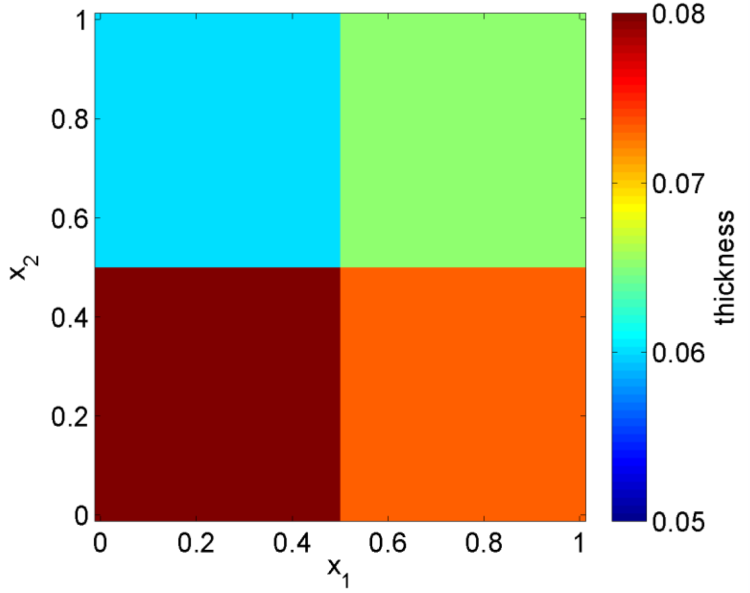


- incremental SVD methods (exploit structure of a rank-one snapshot update)
- operator inference methods (non-intrusive)
- convergence guarantees in idealized noise-free case

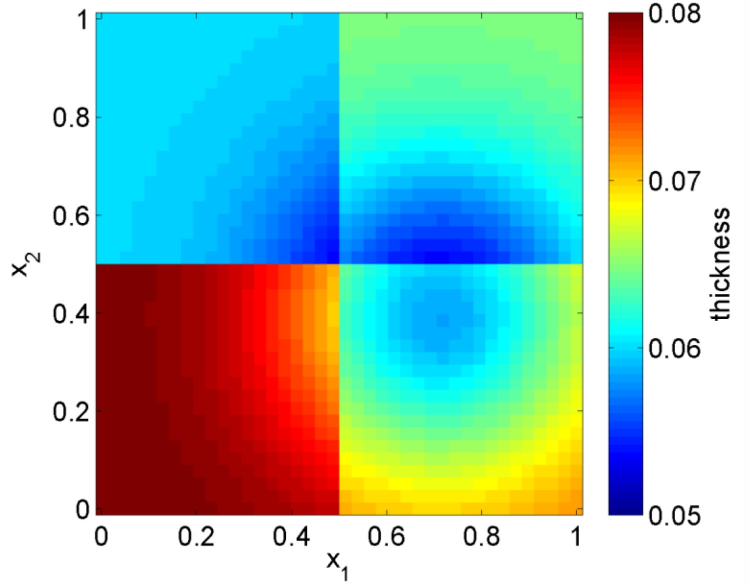
Example: locally damaged plate

High-fidelity:
finite element model

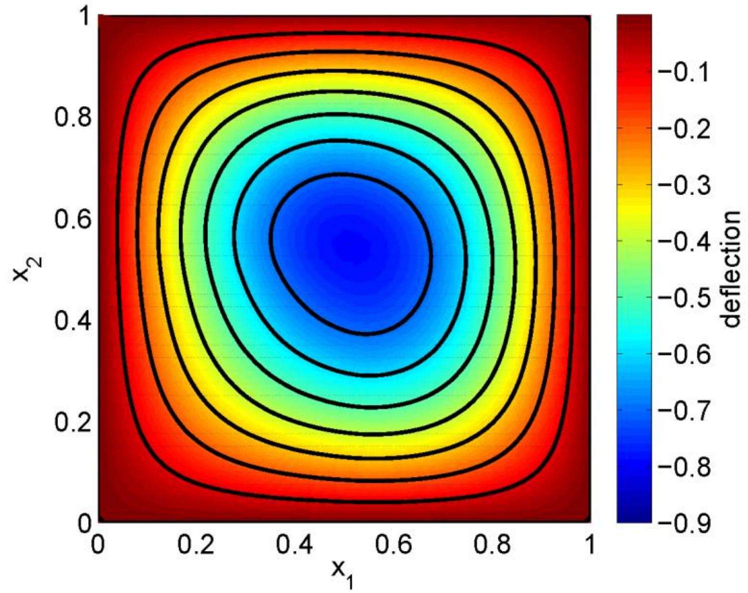
Reduced model:
proper orthogonal decomposition



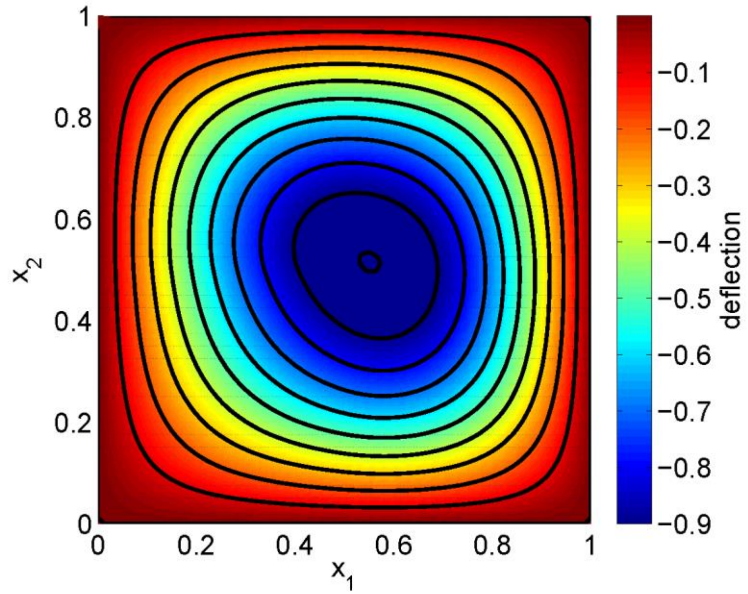
thickness, no damage



thickness, damage up to 20%



deflection, no damage

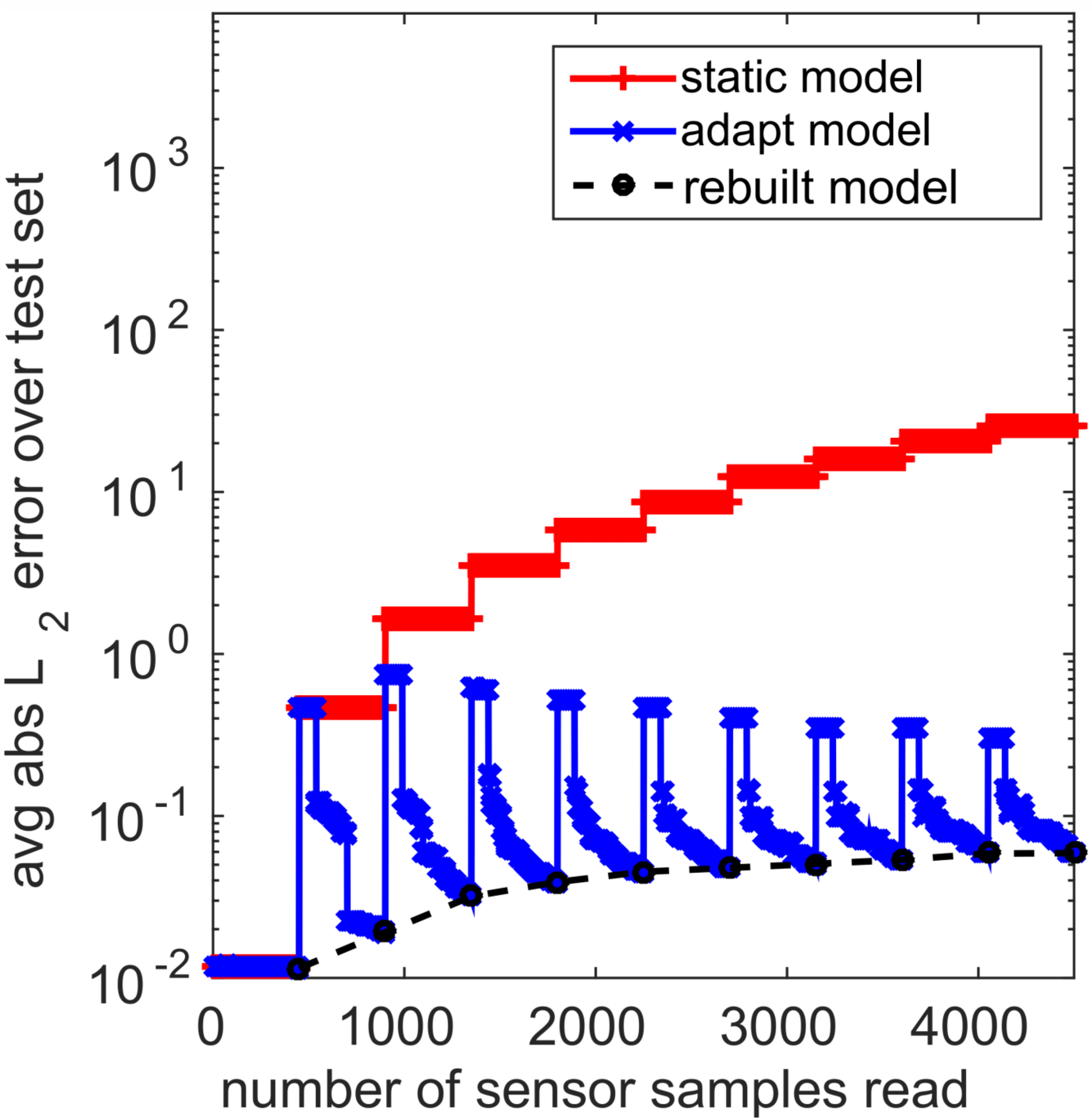


deflection, damage up to 20%

Data-driven adaptation: locally damaged plate

Adapting the ROM after damage

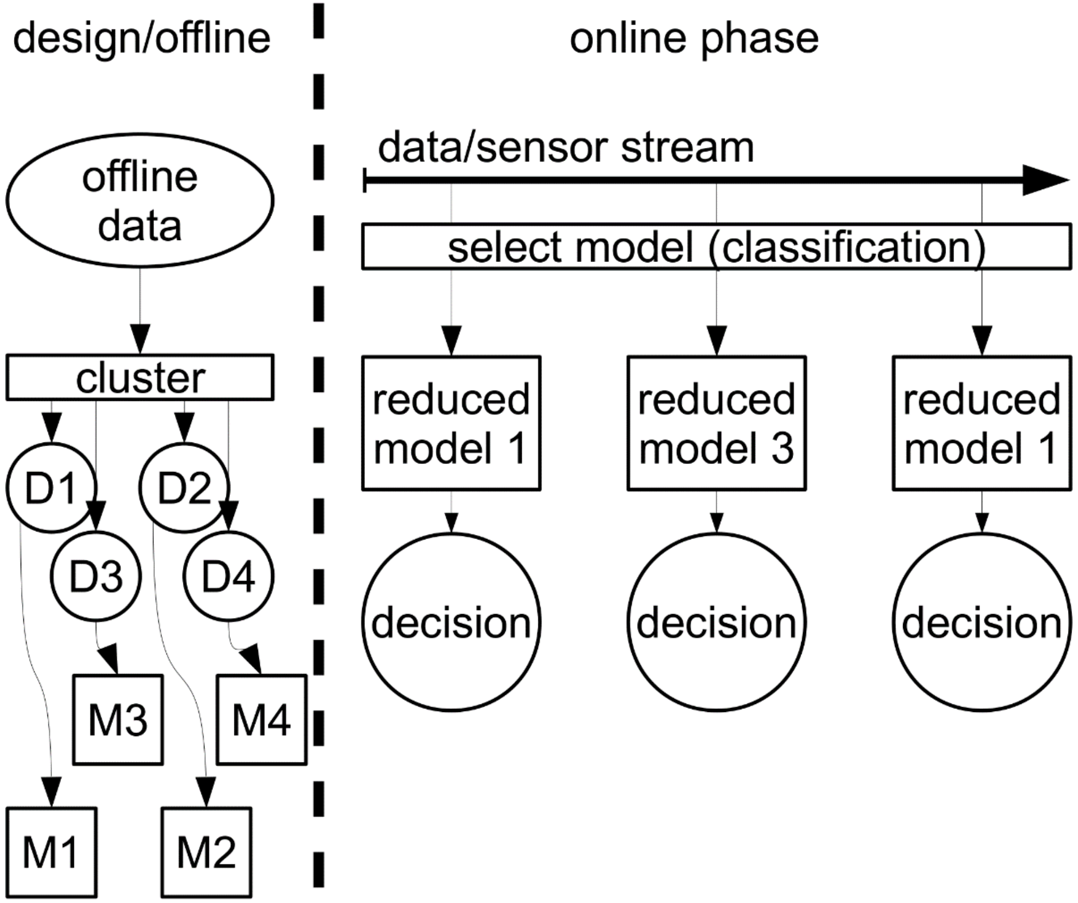
Speedup of 10^4 cf. rebuilding ROM



Localized and adaptive reduced models

[Peherstorfer, Butnaru, W., Bungartz; SISC 2014]

- Automatic model management based on machine learning
  - **Cluster** set of snapshots $\mathcal{S} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_M\} \subset \mathbb{R}^N$ into $\mathcal{S} = \mathcal{S}_1 \uplus \cdots \uplus \mathcal{S}_k$ (using e.g. k-means)
  - Create a separate **local reduced model** for each cluster
  - Derive a basis $Q \in \mathbb{R}^{N \times m}, m \ll N$ to obtain low-dimensional **indicator** $z_i = Q^T x_i$ that describes state x_i
  - Learn a **classifier** $g\colon \mathcal{Z} \to \{1, \dots, k\}$ to map from low-dimensional indicator z to model index (using e.g. nearest neighbors)
  - Classify current state/indicator online and select model



→ Localized DEIM (**LDEIM**): Reduced models are tailored to local system behavior

Localized and adaptive reduced models

[Peherstorfer, Butnaru, W., Bungartz; SISC 2014]

- Example: Reacting flow with one-step reaction

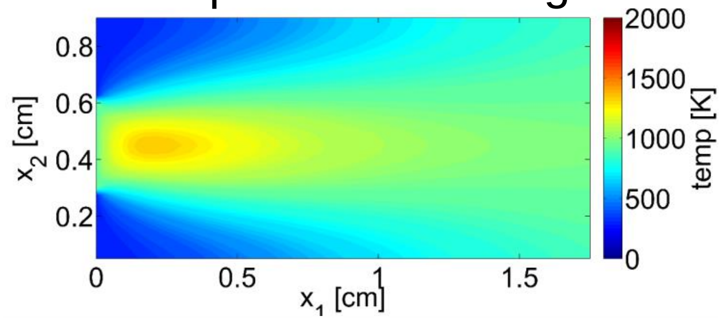
$$2\mathrm{H}_2 + \mathrm{O}_2 \rightarrow 2\mathrm{H}_2\mathrm{O}$$

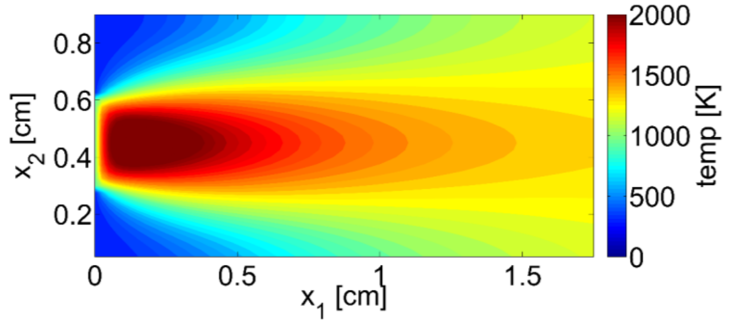
- Governed by convection-diffusion-reaction equation

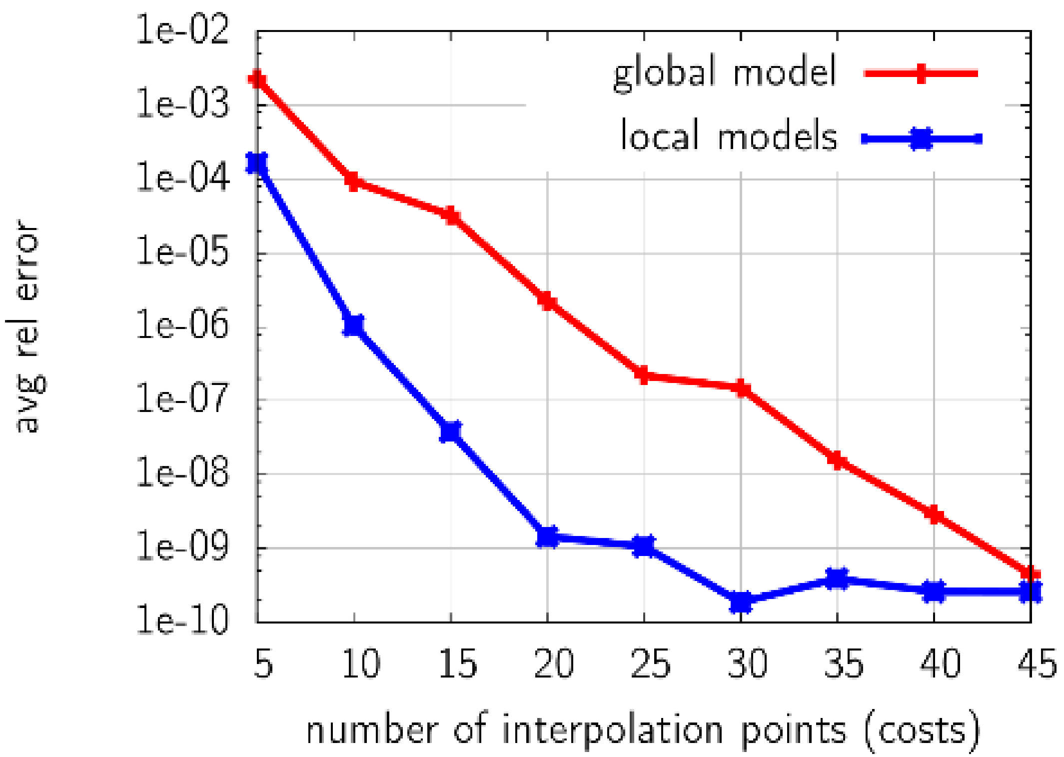
$$\kappa \Delta \boldsymbol{y} - \nu \nabla \boldsymbol{y} + \boldsymbol{F}(\boldsymbol{y}, \boldsymbol{\mu}) = 0 \quad \text{in } \Omega$$

- Exponential nonlinearity (Arrhenius-type source term)

Temperature field of flame for different parameter configurations







POD-LDEIM: Combining **4 local models** with machine-learning-based model management achieves **accuracy improvement by up to two orders of magnitude** compared to a single, global model

7. Conclusions and Challenges

Conclusions

- Many engineered systems of the future will have abundant sensor data
- Many systems of the future will leverage edge computing

→ an important role for reduced models, adaptive modeling, multifidelity modeling, uncertainty quantification

→ important to leverage the relative strengths of models and data

Challenges

Where do existing theories and methods fall short?

- Nonlinear parameter-varying systems
  - → moving beyond linear subspaces
  - → effective & efficient approximation of nonlinear terms
  - → adaptive, data-driven methods
- Multiscale problems
  - → effects of unresolved scales (closure)
  - → ROMs across multiple scales
- Lack of rigorous error guarantees
  - → especially for nonlinear problems
- Model inadequacy
- Intrusiveness of most existing model reduction methods has limited their impact

Useful References: Survey & Overview papers

Antoulas, A.C. *Approximation of Large-Scale Dynamical Systems*, SIAM, Philadelphia, PA, 2005.

Benner, P., Gugercin, S. and Willcox, K., A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems, *SIAM Review*, Vol. 57, No. 4, pp. 483–531, 2015.

Dowell, E. and Hall, K. Modeling of fluid-structure interaction. *Annual Review of Fluid Mechanics*, 33:445-90, 2001.

Gugercin, S. and Antoulas, A. (2004). A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77:748-766.

Patera A. and Rozza, G. *Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations*, Version 1.0, MIT, Cambridge, MA, 2006.

Useful References

Amsallem, D. and Farhat, C. Interpolation method for the adaptation of reduced-order models to parameter changes and its application to aeroelasticity, *AIAA Journal*, 46 (2008), pp. 1803–1813.

Amsallem, D. and Farhat, C. An online method for interpolating linear parametric reduced-order models, *SIAM J. Sci. Comput*., 33 (2011), pp. 2169–2198.

Amsallem, D., Zahr, M.s and Farhat, C. Nonlinear model order reduction based on local reduced-order bases, *International Journal for Numerical Methods in Engineering*, 92 (2012), pp. 891–916.

Antoulas, A.C. *Approximation of Large-Scale Dynamical Systems*, SIAM, Philadelphia, PA, 2005.

Astrid, P., Weiland, S. Willcox, K. and Backx, T. Missing point estimation in models described by proper orthogonal decomposition, IEEE Trans. Automat. Control, 53 (2008), pp. 2237–2251.

Barrault, M. Maday, Y. Nguyen, N. and Patera, A. An “empirical interpolation” method: Application to efficient reduced-basis discretization of partial differential equations, C. R. Math. Acad. Sci. Paris, 339 (2004), pp. 667–672.

Barthelemy, J-F. M. and Haftka, R.T., “Approximation concepts for optimum structural design – a review”, *Structural Optimization*, 5:129-144, 1993.

Bashir, O., Willcox, K., Ghattas, O., van Bloemen Waanders, B., and Hill, J., “Hessian-Based Model Reduction for Large-Scale Systems with Initial Condition Inputs,” *International Journal for Numerical Methods in Engineering*, Vol. 73, Issue 6, pp. 844-868, 2008.

Benner, P., Gugercin, S. and Willcox, K., A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems, *SIAM Review*, Vol. 57, No. 4, pp. 483–531, 2015.

Useful References

Bui-Thanh, T., Willcox, K., and Ghattas, O. Model reduction for large-scale systems with high-dimensional parametric input space. SIAM Journal on Scientific Computing, 30(6):3270-3288, 2008.

Bui-Thanh, T., Willcox, K., and Ghattas, O. Parametric reduced-order models for probabilistic analysis of unsteady aerodynamic applications. AIAA Journal, 46(10):2520-2529, 2008.

Carlberg, K. Farhat, C. Cortial, J. and Amsallem, D. The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows, *J. Comput. Phys.,* 242 (2013), pp. 623–647.

Chaturantabut, S. and Sorensen, D. C. Nonlinear model reduction via discrete empirical interpolation, *SIAM J. Sci. Comput*., 32 (2010), pp. 2737–2764.

Cui, T., Marzouk, Y. and Willcox, K. Data-driven model reduction for the Bayesian solution of inverse problems, *International Journal for Numerical Methods in Engineering*, Vol. 102, No. 5, pp. 966-990, 2014.

Deane, A., Kevrekidis, I., Karniadakis, G., and Orszag, S. Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders. Phys. Fluids, 3(10):2337-2354, 1991.

Dowell, E. and Hall, K. Modeling of fluid-structure interaction. Annual Review of Fluid Mechanics, 33:445-90, 2001.

Gallivan, K., Grimme, E., and Van Dooren, P. (1994). Pade approximation of large-scale dynamic systems with Lanczos methods. Proceedings of the 33rd IEEE Conference on Decision and Control.

Giunta, A.A. and Watson, L.T.,"A comparison of approximation modeling techniques: polynomial versus interpolating models", AIAA Paper 98-4758, 1998.

Grepl, M. and Patera, A. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations, M2AN Math. Model. Numer. Anal., 39 (2005), pp. 157–181.

Useful References

Grimme, E. (1997). Krylov Projection Methods for Model Reduction. PhD thesis, Coordinated-Science Laboratory, University of Illinois at Urbana-Champaign.

Gugercin, S. and Antoulas, A. (2004). A survey of model reduction by balanced truncation and some new results. International Journal of Control, 77:748-766.

Hinze M. and Volkwein, S. Error estimates for abstract linear-quadratic optimal control problems using proper orthogonal decomposition, Comput. Optim. Appl., 39 (2008), pp. 319–345.

Jones, D.R., “A taxonomy of global optimization methods based on response surfaces,” Journal of Global Optimization, 21, 345-383, 2001.

Kennedy, M. and O'Hagan, A. (2001). Bayesian calibration of computer models. Journal of the Royal Statistical Society, 63(2):425-464.

Lall, S. Marsden, J., and Glavaski, S. A subspace approach to balanced truncation for model reduction of nonlinear control systems, Internat. J. Robust Nonlinear Control, 12 (2002), pp. 519–535.

LeGresley, P.A. and Alonso, J.J., “Airfoil design optimization using reduced order models based on proper orthogonal decomposition”, AIAA Paper 2000-2545, 2000.

Lophaven, S., Nielsen, H., and Sondergaard, J. (2002). Aspects of the Matlab toolbox DACE. Technical Report IMM-REP-2002-13, Technical University of Denmark.

Moore, B. Principal component analysis in linear systems: Controllability, observability, and model reduction, IEEE Trans. Automat. Control, 26 (1981), pp. 17–32.

Noor, A. and Peters, J. (1980). Reduced basis technique for nonlinear analysis of structures. AIAA Journal, 18(4):455-462.

Patera, A. and Rozza, G. Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations, Version 1.0, MIT, Cambridge, MA, 2006.

Peherstorfer, B., Butnaru, D., Willcox, K. and Bungartz, H.-J., Localized discrete empirical interpolation method, *SIAM Journal on Scientific Computing*, Vol. 36, No. 1, pp. A168-A192, 2014.

Useful References

Peherstorfer, B. and Willcox, K., Online Adaptive Model Reduction for Nonlinear Systems via Low-Rank Updates, *SIAM Journal on Scientific Computing*, Vol. 37, No. 4, pp. A2123-A2150, 2015.

Peherstorfer, B. and Willcox, K., Dynamic data-driven reduced-order models, *Computer Methods in Applied Mechanics and Engineering*, Vol. 291, pp. 21-41, 2015.

Penzl, T. (2006). Algorithms for model reduction of large dynamical systems. Linear Algebra and its Applications, 415(2-3):322-343.

Prud’homme, C., Rovas, D., Veroy, K., Maday, Y., Patera, A. and Turinici, G. Reliable real-time solution of parameterized partial differential equations: Reduced-basis output bound methods, J. Fluids Engrg., 124 (2002), pp. 70–80.

Simpson, T., Peplinski, J., Koch, P., and Allen, J. (2001). Metamodels for computer based engineering design: Survey and recommendations. Engineering with Computers, 17:129-150.

Sirovich, L. (1987). Turbulence and the dynamics of coherent structures. Part 1: Coherent structures. Quarterly of Applied Mathematics, 45(3):561-571.

Sorensen, D. and Antoulas, A. (2002). The Sylvester equation and approximate balanced reduction. Linear Algebra and its Applications, 351-352:671-700.

Swischuk, R., Mainini, L., Peherstorfer, B. and Willcox, K., Projection-based model reduction: Formulations for physics-based machine learning, Computers and Fluids, to appear, 2018.

Vanderplaats, G.N., *Numerical Optimization Techniques for Engineering Design,* Vanderplaats R&D, 1999.

Veroy, K., Prud'homme, C., Rovas, D., and Patera, A. (2003). A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. AIAA Paper 2003-3847, Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, Orlando, FL.

Veroy, K. and Patera, A. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: Rigorous reduced-basis a posteriori error bounds, Internat. J. Numer. Methods Fluids, 47 (2005), pp. 773–788.

Willcox, K. and Peraire, J. Balanced model reduction via the proper orthogonal decomposition, *AIAA J.*, 40 (2002), pp. 2323–2330.