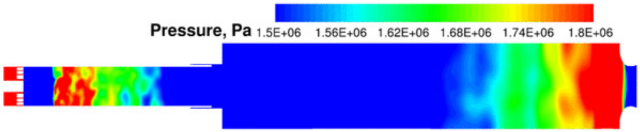
Online adaptive model reduction with applications to rotating detonation waves

Wayne Isaac Tan Uy, Rodrigo Singh, and Benjamin Peherstorfer
Courant Institute of Mathematical Sciences, New York University

August 2023

Intro: Reducing transport-dominated problems



Linear approximations fail for transport-dominated problems

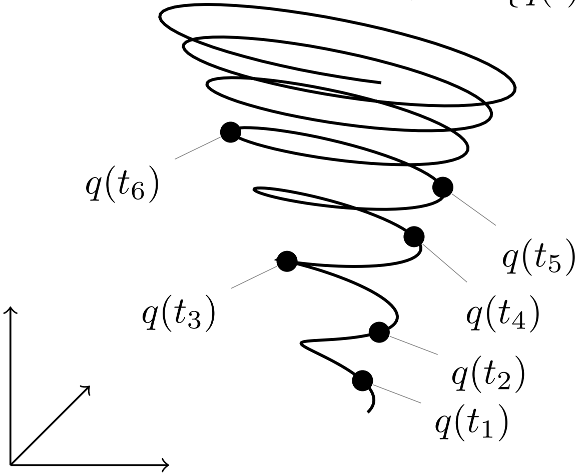
- Sharp gradients ("flame front") in solutions lead to slow decay of singular values
- There is *no* low-dimensional subspace that approximates solutions well

Stability, Gibbs-like phenomena

- Gibbs-like phenomena
- Unstable behavior of reduced models (increasing modes, increases error)

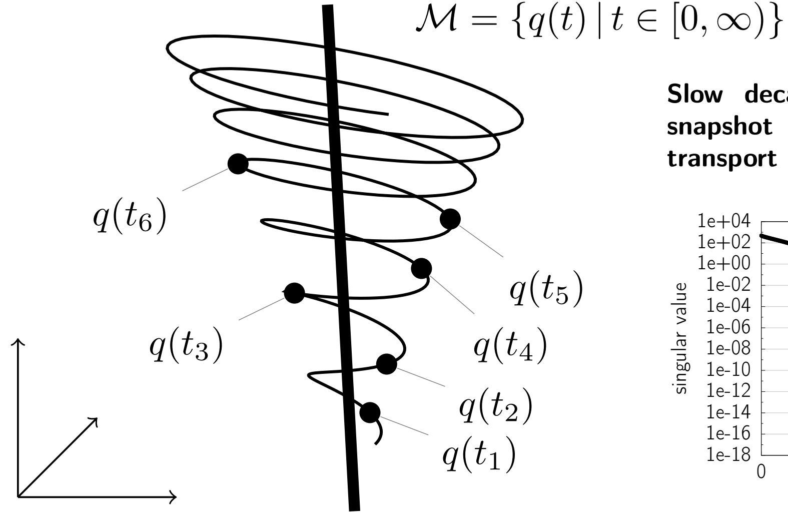
Intro: Linear approximations of manifolds

$$\mathcal{M} = \{q(t) \,|\, t \in [0, \infty)\}$$

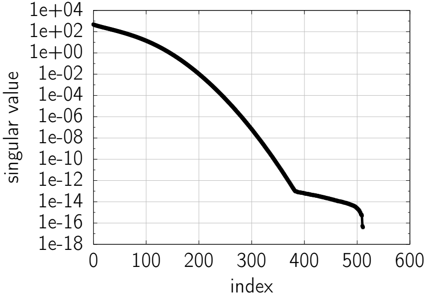


Reduction of transport phenomena can be only achieved via nonlinear approximations

Intro: Linear approximations of manifolds



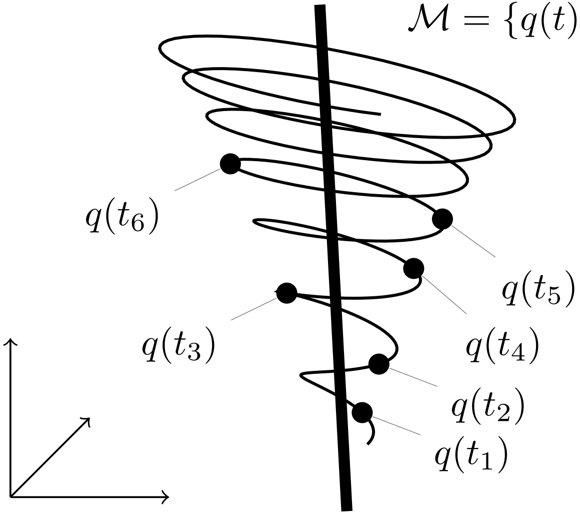
Slow decay of singular values of snapshot matrix in presence of transport phenomena



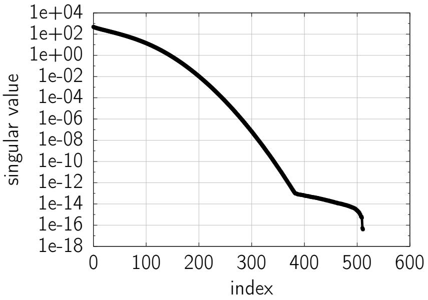
Reduction of transport phenomena can be only achieved via nonlinear approximations

Intro: Linear approximations of manifolds

$$\mathcal{M} = \{q(t) \,|\, t \in [0, \infty)\}$$



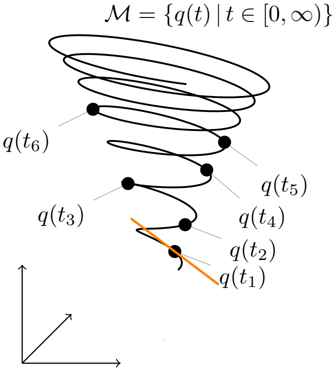
Slow decay of singular values of snapshot matrix in presence of transport phenomena



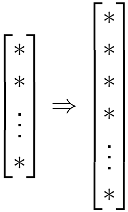
Reduction of transport phenomena can be only achieved via nonlinear approximations

Outline: Online adaptive model reduction

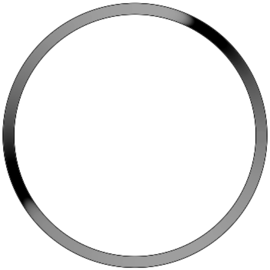
1. Nonlinear approximations via adaptive spaces



2. The importance of sampling ("training")

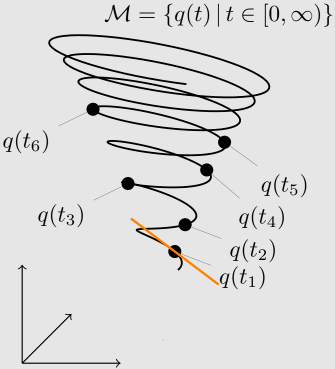


3. Applications

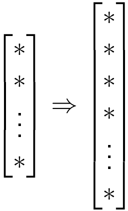


Outline: Online adaptive model reduction

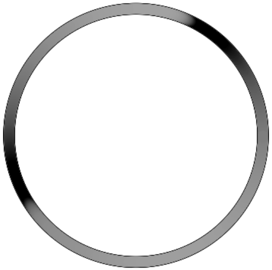
1. Nonlinear approximations via adaptive spaces



2. The importance of sampling ("training")



3. Applications



ADEIM: Full model

Full model

$$\boldsymbol{q}_k(\mu) = \boldsymbol{f}(\boldsymbol{q}_{k+1}(\mu); \mu)\,, \qquad k = 1, \dots, K$$

- Time steps $k = 1, \dots, K$
- Parameter $\mu \in \mathcal{D} \subset \mathbb{R}^d$
- State $\boldsymbol{q}_k(\mu) \in \mathbb{R}^N$ at time step k and parameter μ
- Function $\boldsymbol{f} : \mathbb{R}^N \times \mathcal{D} \to \mathbb{R}^N$
- Trajectory $\boldsymbol{Q}(\mu) = [\boldsymbol{q}_1(\mu), \dots, \boldsymbol{q}_K(\mu)] \in \mathbb{R}^{N \times K}$

Construct POD basis $\boldsymbol{U} = [\boldsymbol{u}_1, \dots, \boldsymbol{u}_n] \in \mathbb{R}^{N \times n}$ of reduced space $\mathcal{U}$ from snapshots

$$\boldsymbol{Q} = [\underbrace{\boldsymbol{q}_1(\mu_1), \dots, \boldsymbol{q}_K(\mu_1)}_{\boldsymbol{Q}(\mu_1)}, \dots, \underbrace{\boldsymbol{q}_1(\mu_M), \dots, \boldsymbol{q}_K(\mu_M)}_{\boldsymbol{Q}(\mu_M)}] \in \mathbb{R}^{N \times MK}$$

ADEIM: Reduced model with linear approximation

Empirical interpolation for approximating nonlinear $\boldsymbol{f}$ [Barrault et al., 2004]

- Select interpolation points $p_1, \dots, p_n \in \{1, \dots, N\}$ corresponding to $\boldsymbol{U}$
- Construct interpolation points matrix

$$\boldsymbol{P} = [\boldsymbol{e}_{p_1}, \dots, \boldsymbol{e}_{p_n}] \in \mathbb{R}^{N \times n}$$

$$\begin{bmatrix} f_{p_1}(\boldsymbol{q}) \\ \vdots \\ f_{p_n}(\boldsymbol{q}) \end{bmatrix} = \begin{bmatrix} & \boldsymbol{P}^T & \end{bmatrix} \begin{bmatrix} f_1(\boldsymbol{q}) \\ f_2(\boldsymbol{q}) \\ f_3(\boldsymbol{q}) \\ f_4(\boldsymbol{q}) \\ \vdots \\ f_N(\boldsymbol{q}) \end{bmatrix}$$

- Define approximation of $\boldsymbol{f}$ via sparse sampling as

$$\tilde{\boldsymbol{f}}(\tilde{\boldsymbol{q}}; \mu) = (\boldsymbol{P}^T \boldsymbol{U})^{-1} \boldsymbol{P}^T \boldsymbol{f}(\boldsymbol{U}\tilde{\boldsymbol{q}}; \mu)$$

  so that $\boldsymbol{U}\tilde{\boldsymbol{f}}(\tilde{\boldsymbol{q}}(\mu); \mu) \in \mathcal{U}$ approximates $\boldsymbol{f}(\boldsymbol{U}\tilde{\boldsymbol{q}}(\mu); \mu) \in \mathbb{R}^N$

Reduced model based on empirical interpolation with fixed space $\mathcal{U}$

$$\tilde{\boldsymbol{q}}_k(\mu) = \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{q}}_{k+1}(\mu); \mu), \qquad \tilde{\boldsymbol{q}}_k(\mu) \in \mathbb{R}^n, \qquad k = 1, \dots, K$$

[Barrault et al., 2004], [Grepl et al., 2007], [Astrid et al., 2008], [Chaturantabut et al., 2010], [Carlberg et al., 2011], [Farhat, Cortial, Chapman, 2012], [Drohmann et al., 2012], [Farhat, Avery, Chapman, Cortial, 2014], [Drmac, Gugercin, 2016]

ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

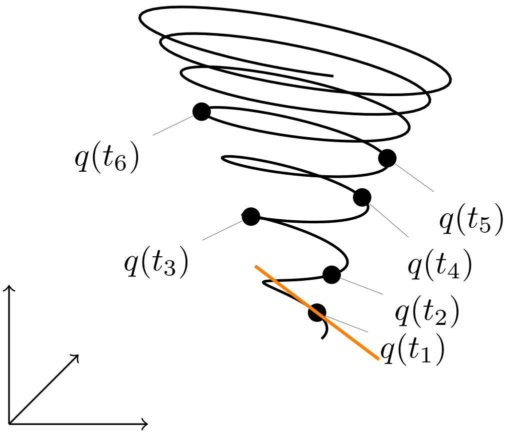
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

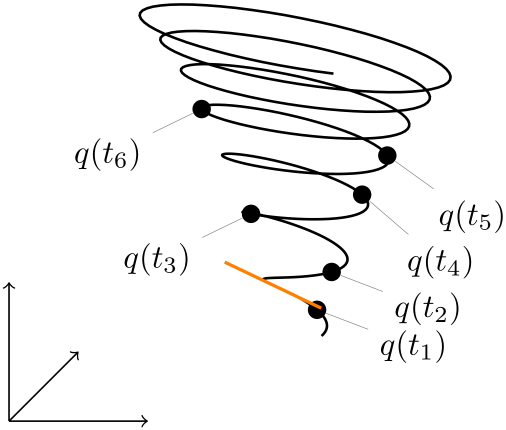
$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T\,, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation



Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T, \qquad k = 0, 1, 2, \dots$$

ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

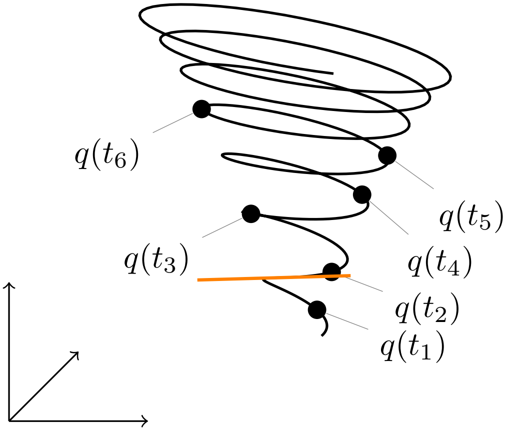
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T \,, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

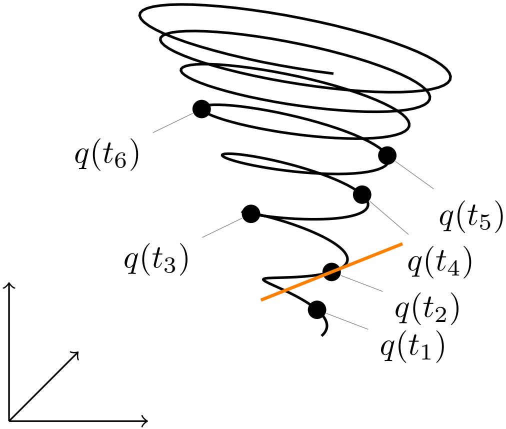
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

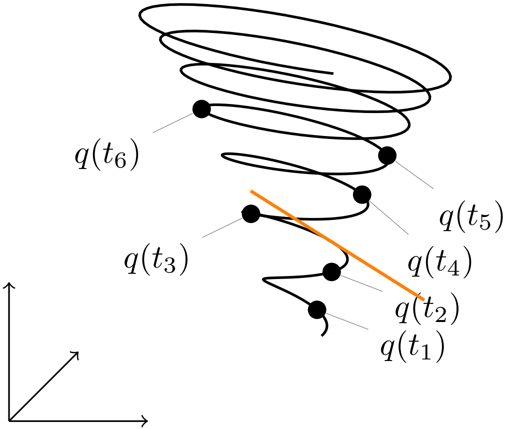
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T\,, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

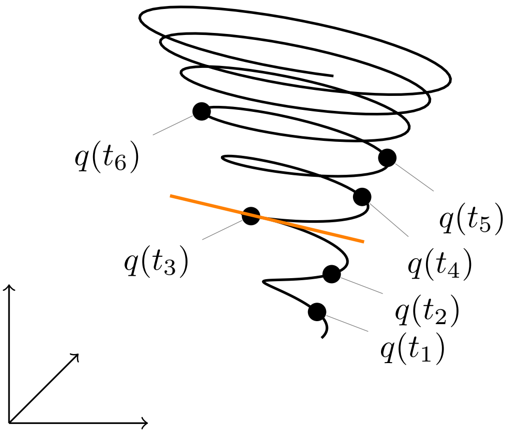
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

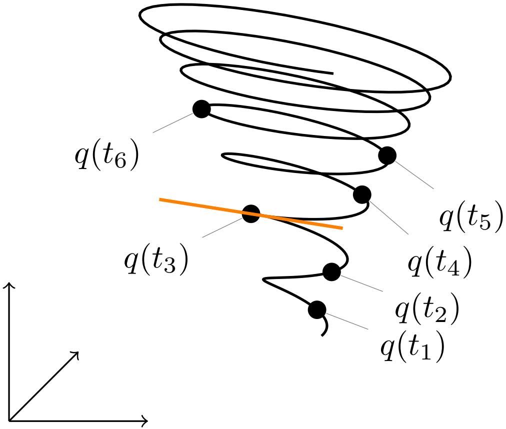
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T\,, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

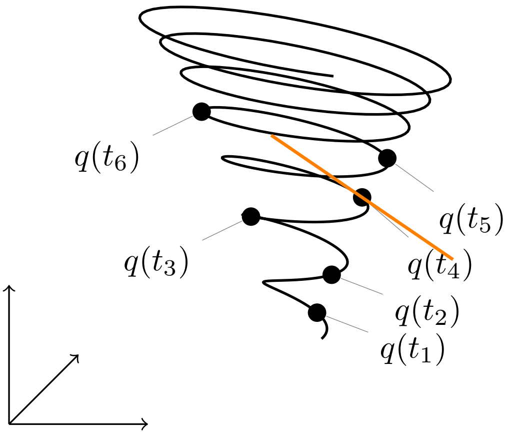
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T\,, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

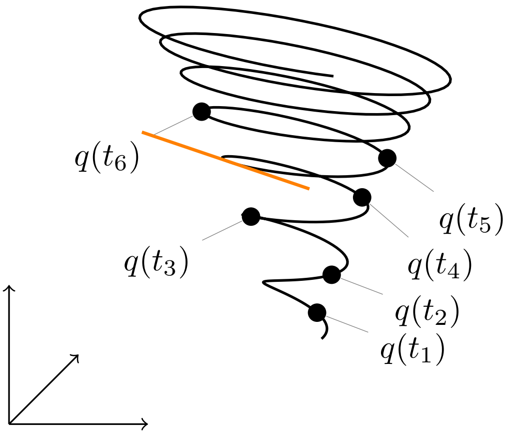
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T, \qquad k = 0, 1, 2, \dots$$



ADEIM: Adaptive empirical interpolation (ADEIM)

Follow manifold by adapting spaces

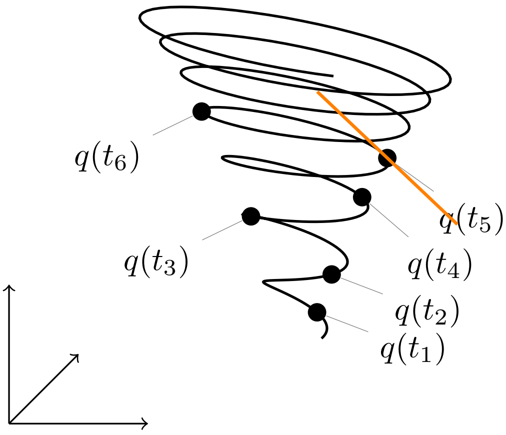
- Leverages local-in-time low-rank structure
- Builds on sparse sampling to drive adaptation

Space at time step k is spanned by columns of

$$\boldsymbol{U}_k = \begin{bmatrix} | & & | \\ \boldsymbol{u}_k^{(1)} & \dots & \boldsymbol{u}_k^{(n)} \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

Adapt basis of reduced model via low-rank updates

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T , \qquad k = 0, 1, 2, \dots$$



ADEIM: Online steps of ADEIM

Step 1: Solve reduced model with empirical interpolation at time step k to compute state $\tilde{\boldsymbol{q}}_{k+1}$

$$\tilde{\boldsymbol{q}}_k(\boldsymbol{\mu}) = \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{q}}_{k+1}(\boldsymbol{\mu}); \boldsymbol{\mu})$$

Step 2: Query sparse full-model state information to update data matrix

$$\boldsymbol{F}_k = [\hat{\boldsymbol{q}}_{k-w-1}, \dots, \hat{\boldsymbol{q}}_k]$$

- Would like to adapt space to full-model solution $\boldsymbol{q}_k$; however, solution $\boldsymbol{q}_k$ unavailable

$$\boldsymbol{q}_k(\mu) = \boldsymbol{f}(\boldsymbol{q}_{k+1}(\mu); \mu)$$

- Use $\boldsymbol{f}(\boldsymbol{U}_k \tilde{\boldsymbol{q}}_{k+1}(\mu))$ as surrogates for $\boldsymbol{q}_k$ to fill columns of $\boldsymbol{F}_k$ via

$$\boldsymbol{S}_k^T \hat{\boldsymbol{q}}_k = \underbrace{\boldsymbol{S}_k^T \boldsymbol{f}(\boldsymbol{U}_k \tilde{\boldsymbol{q}}_{k+1}(\boldsymbol{\mu}); \boldsymbol{\mu})}_{\substack{\text{sample full-model } \boldsymbol{f} \\ \text{at } m \text{ sampling points}}}, \qquad \breve{\boldsymbol{S}}_k^T \hat{\boldsymbol{q}}_k = \underbrace{\breve{\boldsymbol{S}}_k^T \boldsymbol{U}_k (\boldsymbol{S}_k^T \boldsymbol{U}_k)^+ \boldsymbol{S}_k^T \boldsymbol{f}(\boldsymbol{U}_k \tilde{\boldsymbol{q}}_{k+1}(\boldsymbol{\mu}); \boldsymbol{\mu})}_{\text{approximate other components via EIM}}$$

ADEIM: Low-rank basis updates

Step 3: Adapt space $\boldsymbol{U}_k \in \mathbb{R}^{N \times n}$ with low-rank update $\boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T \in \mathbb{R}^{N \times n}$

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T$$

- The ADEIM update $\boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T$ minimizes

$$\left\| \boldsymbol{S}_k^T \left(\left(\boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T \right) \boldsymbol{C}_k - \boldsymbol{F}_k \right) \right\|_F^2$$

- Sampling points matrix $\boldsymbol{S}_k \in \mathbb{R}^{N \times m}$ of m points $s_1, \dots, s_m \in \{1, \dots, N\}$
- Coefficient matrix $\boldsymbol{C}_k = (\boldsymbol{P}_k^T \boldsymbol{U}_k)^{-1} \boldsymbol{P}_k^T \boldsymbol{F}_k$
- Costs of obtaining update are in $\mathcal{O}(mw^2)$ with SVD of $m \times w$ matrix

Step 4: Update sampling points $\boldsymbol{S}_k$ to $\boldsymbol{S}_{k+1}$, update empirical-interpolation points $\boldsymbol{P}_k$ to $\boldsymbol{P}_{k+1}$

[P., Willcox, SISC, 2015], [P., SISC, 2020]

ADEIM: Analysis of ADEIM in ideal setting

Distance measure between subspaces

$$d(\bar{\mathcal{U}}_k, \mathcal{U}_k) = \|\bar{\boldsymbol{U}}_k - \boldsymbol{U}_k \boldsymbol{U}_k^T \bar{\boldsymbol{U}}_k\|_F^2$$

Proposition 1
Let $\boldsymbol{F}_k = \bar{\boldsymbol{U}}_{k+1}\tilde{\boldsymbol{F}}_k$ with $\tilde{\boldsymbol{F}}_k$ full rank and set $\boldsymbol{R}_k = \boldsymbol{U}_k\boldsymbol{C}_k - \boldsymbol{F}_k$. Let $\bar{r}$ be the rank of $\boldsymbol{S}_k^T\boldsymbol{R}_k$ and $\sigma_1 \geq \cdots \geq \sigma_{\bar{r}}$ be its singular values. Set $\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k\boldsymbol{\beta}_k^T$ with the rank-r ADEIM update $\boldsymbol{\alpha}_k\boldsymbol{\beta}_k^T$, then,

$$d(\bar{\mathcal{U}}_{k+1}, \mathcal{U}_{k+1}) \leq \frac{b_k(\boldsymbol{S}_k)}{\sigma^2_{\min}(\boldsymbol{F}_k)}$$

with

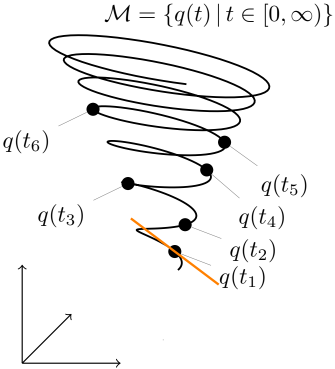
$$b_k(\boldsymbol{S}_k) = \|\boldsymbol{R}_k\|_F^2 - \sum_{i=1}^{r}\sigma_i^2 = \|\breve{\boldsymbol{S}}_k^T\boldsymbol{R}_k\|_F^2 + \sum_{i=r+1}^{\bar{r}}\sigma_i^2$$

- Complementary sampling points matrix $\breve{\boldsymbol{S}}_k$
- Establishes importance of sampling points

P., ***Model reduction for transport-dominated problems via online adaptive bases and adaptive sampling***, SISC 2020.

Outline: Online adaptive model reduction

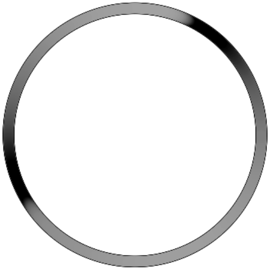
1. Nonlinear approximations via adaptive spaces



2. The importance of sampling (“training”)

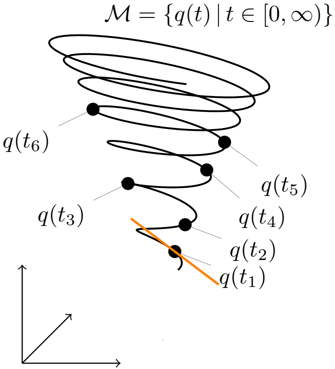
$$\begin{bmatrix} * \\ * \\ \vdots \\ * \end{bmatrix} \Rightarrow \begin{bmatrix} * \\ * \\ * \\ * \\ \vdots \\ * \end{bmatrix}$$

3. Applications

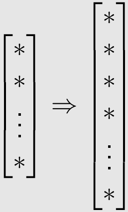


Outline: Online adaptive model reduction

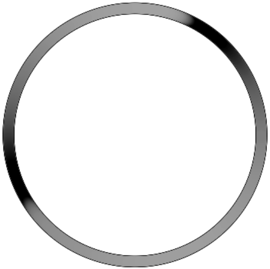
1. Nonlinear approximations via adaptive spaces



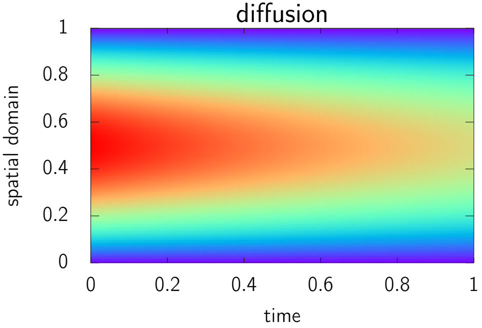
2. The importance of sampling ("training")

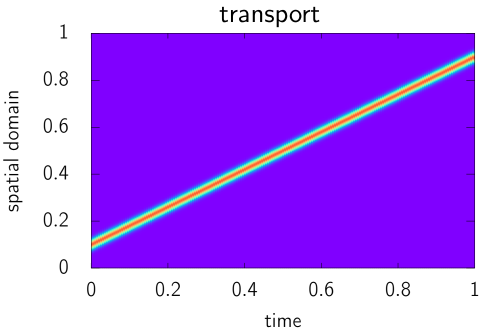


3. Applications



Sampling: Physics determines properties of reduced spaces





$$\boldsymbol{Q} = \begin{bmatrix} * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

$$\boldsymbol{Q} = \begin{bmatrix} & & & & & * & * & \\ & & & & * & * & & \\ & & & * & * & & & \\ & & * & * & & & & \\ & * & * & & & & & \\ * & * & & & & & & \\ * & & & & & & & \end{bmatrix}$$

Sampling: Low coherence

Measures how much unit vector $\boldsymbol{e}_j$ gets "scrambled"

- Projection onto a subspace $\mathcal{U}$

$$\boldsymbol{U}_\parallel = \boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^T$$

- Define coherence of subspace $\mathcal{U}$ as

$$\gamma(\mathcal{U}) = \frac{N}{d} \max_{j=1,\dots,N} \|\boldsymbol{U}_\parallel \boldsymbol{e}_j\|_2^2$$

with canonical unit vectors $\boldsymbol{e}_j \in \mathbb{R}^N$

$$\begin{bmatrix} & & \\ & \boldsymbol{U}_\parallel & \\ & & \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} * \\ \vdots \\ * \\ * \\ * \\ \vdots \\ * \end{bmatrix}$$

Low coherence (diffusion)

- Unit vector poorly represented in subspace
- All components are informative

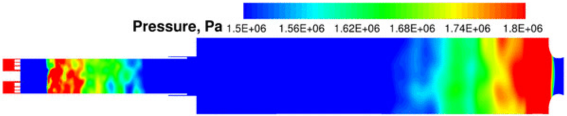
Subspace with low coherence (diffusion), then uniform sampling works well to gather information

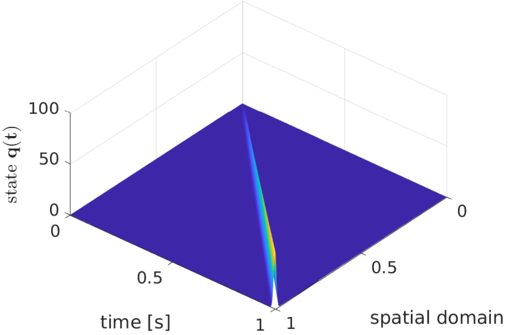
Sampling: Coherent subspaces

Unit vector can be represented well in space

$$\begin{bmatrix} & & \\ & \boldsymbol{U}_{\|} & \\ & & \end{bmatrix}\begin{bmatrix}0\\ \vdots \\ 0\\ 1\\ 0\\ \vdots \\ 0\end{bmatrix} = \begin{bmatrix}0\\ \vdots \\ 0\\ 1\\ 0\\ \vdots \\ 0\end{bmatrix}$$

- Need to sample jth component
- Adaptive sampling necessary





Convection of local features leads to high coherence

- Nonzero values correspond to wave front
- Want subspaces that approximate well $\boldsymbol{e}_j$

Sampling: Local coherence

Ordering $j_1, \dots, j_N$ of $1, \dots, N$ with fast decay of local coherence

$$\gamma_{j_i}(\mathcal{U}) \lesssim \exp\left(-c i^a\right), \qquad i = 1, \dots, N$$

- Rate $a > 1$ and constant $c > 0$
- Dimension N is finite and therefore constant and rate are important

Residual of ADEIM approximation in space $\mathcal{U}$ inherits local coherence

$$\|(\boldsymbol{U}\boldsymbol{C} - \boldsymbol{F})^T \boldsymbol{e}_i\|_2^2 \lesssim \exp\left(-c' i^{a'}\right), \quad i = 1, \dots, N.$$

- Residual is local in the spatial domain
- Only few components of residual actually contribute and carry information about residual
- Observing the residual at few components is sufficient, if we observe the right components

Sampling: Towards adaptive sampling

From Proposition 1 know that DEIM residual plays critical role

$$d(\bar{\mathcal{U}}_{k+1}, \mathcal{U}_{k+1}) \leq \frac{\boldsymbol{b}_k(\boldsymbol{S}_k)}{\sigma^2_{\min}(\boldsymbol{F}_k)}$$

with

$$\boldsymbol{b}_k(\boldsymbol{S}_k) = \|\boldsymbol{R}_k\|_F^2 - \sum_{i=1}^{r} \sigma_i^2 = \|\breve{\boldsymbol{S}}_k^T \boldsymbol{R}_k\|_F^2 + \sum_{i=r+1}^{\bar{r}} \sigma_i^2$$

Insights from ideal case with full-rank updates

- Decay factor is $\boldsymbol{b}_k(\boldsymbol{S}_k) = \|\breve{\boldsymbol{S}}_k^T \boldsymbol{R}_k\|_F^2$
- Error $d(\bar{\mathcal{U}}_{k+1}, \mathcal{U}_{k+1})$ decays up to constants as fast as local coherence
- Local coherence means few residual components matter, thus few samples required for update
- However, need to select the components that carry high residual

Sampling: Adaptive sampling

Optimal sampling points

$$\boldsymbol{S}_k^{\text{opt}} = \arg\min_{\boldsymbol{S}} b_k(\boldsymbol{S})$$

- Optimal points that minimize error bound
- No algorithm faster than combinatorial known to compute $\boldsymbol{S}_k^{\text{opt}}$

The AADEIM sampling points

$$\boldsymbol{S}_k^{\text{AADEIM}} = \arg\max_{\boldsymbol{S}} \|\boldsymbol{S}^T(\boldsymbol{U}_k\boldsymbol{C}_k - \boldsymbol{F}_k)\|_F^2$$

- Define $r_i = \|(\boldsymbol{U}_k\boldsymbol{C}_k - \boldsymbol{F}_k)^T\boldsymbol{e}_i\|_2^2$ for $i = 1, \ldots, N$
- Let $j_1, \ldots, j_N$ be such that

$$r_{j_1} \geq r_{j_2} \geq \cdots \geq r_{j_N}$$

- Select $\boldsymbol{S} = [\boldsymbol{e}_{j_1}, \ldots, \boldsymbol{e}_{j_m}]$
- Optimal if full-rank update $r = \bar{r}$

P., ***Model reduction for transport-dominated problems via online adaptive bases and adaptive sampling***, SISC 2020.

Sampling: Quasi optimally of AADEIM samples

Proposition
For an ADEIM update with any rank r, the AADEIM sampling points achieve

$$b_k(\boldsymbol{S}_k^{\text{AADEIM}}) \leq 2 b_k(\boldsymbol{S}_k^{\text{opt}})$$

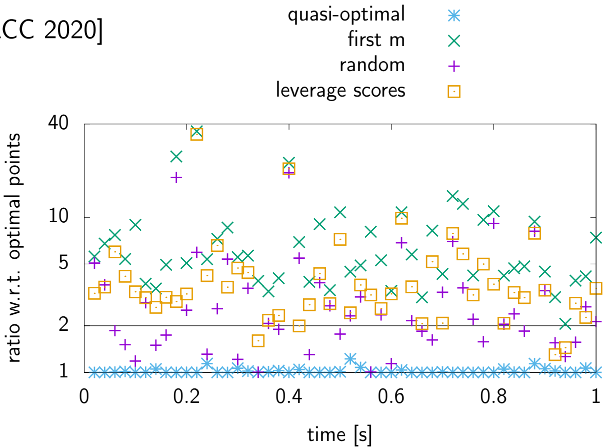
This bound is tight.
Proof [Cortinovis, Kressner, Massei, P., ACC 2020]

Toy example

- RC ladder with $N = 12$ states
- Plot ratio

$$\frac{b_k(\boldsymbol{S}_k^{\text{AADEIM}})}{b_k(\boldsymbol{S}_k^{\text{opt}})}$$

- Compare to other sampling schemes



Remark: Nonlinear approximations with deep networks

Analogous sampling issues arise in other settings of nonlinear parametrizations, e.g., deep neural networks.

Representation $\varphi_1, \dots, \varphi_n$ adapted to target $q(t, \cdot)$ via features $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_{n'}]^T$

$$\tilde{q}(t, \boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \beta_i \varphi_i(t, \boldsymbol{x}; \boldsymbol{\alpha})$$

Remark: Nonlinear approximations with deep networks

Analogous sampling issues arise in other settings of nonlinear parametrizations, e.g., deep neural networks.

Representation $\varphi_1, \dots, \varphi_n$ adapted to target $q(t, \cdot)$ via features $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_{n'}]^T$

$$\tilde{q}(t, \boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \beta_i \varphi_i(t, \boldsymbol{x}; \boldsymbol{\alpha})$$

Fit parameter $\boldsymbol{\theta} = [\boldsymbol{\alpha}, \boldsymbol{\beta}]$ by minimizing PDE residual $R(t, \boldsymbol{x}; \boldsymbol{\theta})$ over time-space domain $\mathcal{T} \times \Omega$

$$\min_{\boldsymbol{\theta} \in \Theta} \mathbb{E}_{(t, \boldsymbol{x}) \sim \nu} \left[R(t, \boldsymbol{x}; \boldsymbol{\theta})^2 \right]$$

Remark: Nonlinear approximations with deep networks

Analogous sampling issues arise in other settings of nonlinear parametrizations, e.g., deep neural networks.

Representation $\varphi_1, \dots, \varphi_n$ adapted to target $q(t, \cdot)$ via features $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_{n'}]^T$

$$\tilde{q}(t, \boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \beta_i \varphi_i(t, \boldsymbol{x}; \boldsymbol{\alpha})$$

Fit parameter $\boldsymbol{\theta} = [\alpha, \beta]$ by minimizing PDE residual $R(t, \boldsymbol{x}; \boldsymbol{\theta})$ over time-space domain $\mathcal{T} \times \Omega$

$$\min_{\boldsymbol{\theta} \in \Theta} \mathbb{E}_{(t, \boldsymbol{x}) \sim \nu} \left[R(t, \boldsymbol{x}; \boldsymbol{\theta})^2 \right]$$

- Draw samples $\{(t_i, \boldsymbol{x}_i)\}_{i=1}^{m}$ from some distribution ν from time-space domain $\mathcal{T} \times \Omega$

Remark: Nonlinear approximations with deep networks

Analogous sampling issues arise in other settings of nonlinear parametrizations, e.g., deep neural networks.

Representation $\varphi_1, \dots, \varphi_n$ adapted to target $q(t, \cdot)$ via features $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_{n'}]^T$

$$\tilde{q}(t, \boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \beta_i \varphi_i(t, \boldsymbol{x}; \boldsymbol{\alpha})$$

Fit parameter $\boldsymbol{\theta} = [\alpha, \beta]$ by minimizing PDE residual $R(t, \boldsymbol{x}; \boldsymbol{\theta})$ over time-space domain $\mathcal{T} \times \Omega$

$$\min_{\boldsymbol{\theta} \in \Theta} \mathbb{E}_{(t, \boldsymbol{x}) \sim \nu} \left[R(t, \boldsymbol{x}; \boldsymbol{\theta})^2 \right]$$

- Draw samples $\{(t_i, \boldsymbol{x}_i)\}_{i=1}^m$ from some distribution ν from time-space domain $\mathcal{T} \times \Omega$
- Fit parameter $\boldsymbol{\theta}$ by minimizing squared residual R at sampled points $\{(t_i, \boldsymbol{x}_i)\}_{i=1}^m$

$$\min_{\boldsymbol{\theta} \in \Theta} \frac{1}{m} \sum_{i=1}^{m} R(t_i, \boldsymbol{x}_i; \boldsymbol{\theta})^2 , \qquad (t_i, \boldsymbol{x}_i) \sim \nu , \qquad i = 1, \dots, m$$

Remark: Nonlinear approximations with deep networks

Analogous sampling issues arise in other settings of nonlinear parametrizations, e.g., deep neural networks.

Representation $\varphi_1, \dots, \varphi_n$ adapted to target $q(t, \cdot)$ via features $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_{n'}]^T$

$$\tilde{q}(t, \boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \beta_i \varphi_i(t, \boldsymbol{x}; \boldsymbol{\alpha})$$

Fit parameter $\boldsymbol{\theta} = [\alpha, \beta]$ by minimizing PDE residual $R(t, \boldsymbol{x}; \boldsymbol{\theta})$ over time-space domain $\mathcal{T} \times \Omega$

$$\min_{\boldsymbol{\theta} \in \Theta} \mathbb{E}_{(t, \boldsymbol{x}) \sim \nu} \left[R(t, \boldsymbol{x}; \boldsymbol{\theta})^2 \right]$$

- Draw samples $\{(t_i, \boldsymbol{x}_i)\}_{i=1}^m$ from some distribution ν from time-space domain $\mathcal{T} \times \Omega$
- Fit parameter $\boldsymbol{\theta}$ by minimizing squared residual R at sampled points $\{(t_i, \boldsymbol{x}_i)\}_{i=1}^m$

$$\min_{\boldsymbol{\theta} \in \Theta} \frac{1}{m} \sum_{i=1}^{m} R(t_i, \boldsymbol{x}_i; \boldsymbol{\theta})^2 , \qquad (t_i, \boldsymbol{x}_i) \sim \nu , \qquad i = 1, \dots, m$$

Very active research area: [Dissanayake et al., 1994], [Berg et al., 2018], [Khoo et al., 2018], [E and Yu, 2018], [Han, Jentzen, E, 2018], [Haber, Ruthotto, 2018], [Sirignano et al., 2018], [Han et al., 2018], [Nonino, Ballarin, Rozza, Maday, 2019], [Raissi et al., 2019], [Rudy, Kutz, Brunton, 2019], [Lee, Carlberg, 2020], [Du, Zaki, 2021], ...

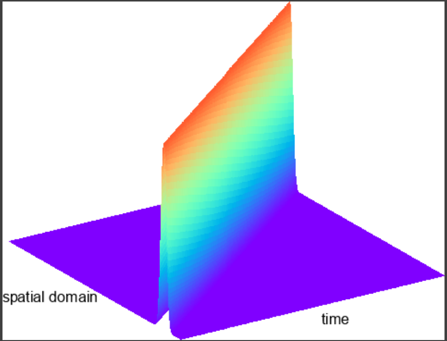
Remark: Challenges of training nonlinear parametrizations

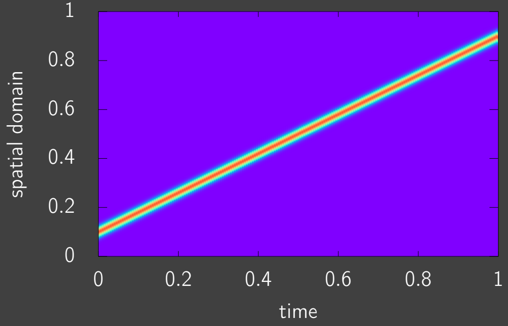
Fit parameter $\boldsymbol{\theta} = [\alpha, \beta]$ by minimizing estimated PDE residual

$$\min_{\boldsymbol{\theta} \in \Theta} \; 1/m \sum_{i=1}^{m} R(t_i, \boldsymbol{x}_i; \boldsymbol{\theta})^2 \,, \qquad (t_i, \boldsymbol{x}_i) \sim \nu \,, \qquad i = 1, \dots, m$$

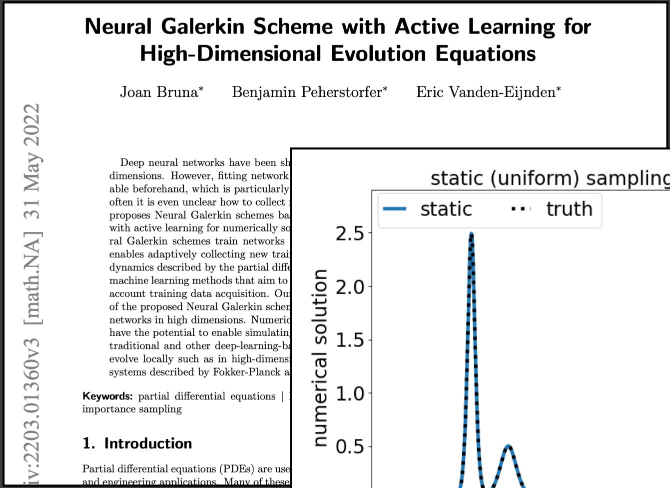
Sampling issue of transport-dominated problems

- Local features (e.g., waves) travel over time
- Collocation needs to discover local features
- Can require lots of samples from $\mathcal{T} \times \Omega$
- Gets exponentially harder with dimension





Remark: Neural Galerkin schemes with active learning



Neural Galerkin Scheme with Active Learning for High-Dimensional Evolution Equations

Joan Bruna* Benjamin Peherstorfer* Eric Vanden-Eijnden*

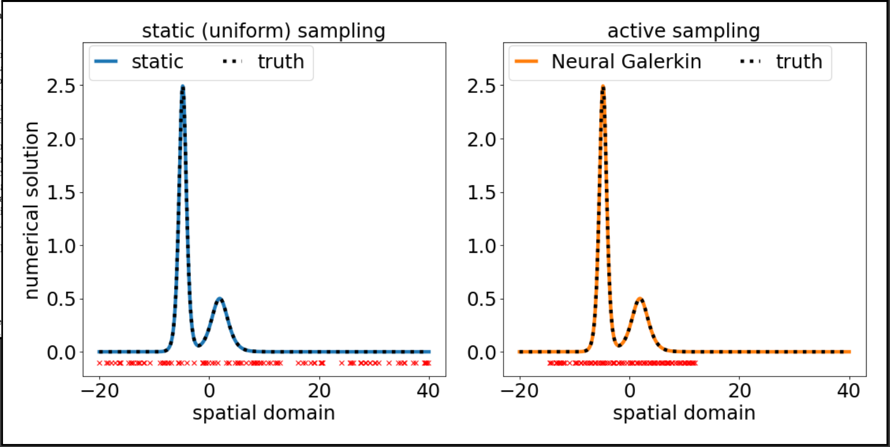
iv:2203.01360v3 [math.NA] 31 May 2022

Deep neural networks have been sh
dimensions. However, fitting network
able beforehand, which is particularly
often it is even unclear how to collect
proposes Neural Galerkin schemes ba
with active learning for numerically so
ral Galerkin schemes train networks
enables adaptively collecting new trai
dynamics described by the partial diffe
machine learning methods that aim to
account training data acquisition. Ou
of the proposed Neural Galerkin schem
networks in high dimensions. Numeric
have the potential to enable simulatin
traditional and other deep-learning-ba
evolve locally such as in high-dimensi
systems described by Fokker-Planck a

Keywords: partial differential equations |
importance sampling

1. Introduction

Partial differential equations (PDEs) are use
and engineering applications. Many of these



Sampling: Points for empirical interpolation

1. Solve reduced model with empirical interpolation at time step k to compute state $\tilde{\boldsymbol{q}}_{k+1}$

$$\tilde{\boldsymbol{q}}_k(\boldsymbol{\mu}) = \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{q}}_{k+1}(\boldsymbol{\mu}); \boldsymbol{\mu})$$

2. Query sparse full-model state information to update data matrix $\boldsymbol{F}_k = [\hat{\boldsymbol{q}}_{k-w-1}, \dots, \hat{\boldsymbol{q}}_k]$

$$\boldsymbol{S}_k^T \hat{\boldsymbol{q}}_k = \boldsymbol{S}_k^T \boldsymbol{f}(\tilde{\boldsymbol{q}}_{k+1}(\boldsymbol{\mu}); \boldsymbol{\mu}), \qquad \breve{\boldsymbol{S}}_k^T \hat{\boldsymbol{q}}_k = \breve{\boldsymbol{S}}_k^T \boldsymbol{U}_k (\boldsymbol{S}_k^T \boldsymbol{U}_k)^+ \boldsymbol{S}_k^T \boldsymbol{f}(\tilde{\boldsymbol{q}}_{k+1}(\boldsymbol{\mu}); \boldsymbol{\mu})$$

3. Update basis with rank-one update $\boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T \in \mathbb{R}^{N \times n}$

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T$$

4. Update sampling points $\boldsymbol{S}_k$ to $\boldsymbol{S}_{k+1}$, update empirical-interpolation points $\boldsymbol{P}_k$ to $\boldsymbol{P}_{k+1}$

Sampling: Perturbing DEIM approximations

Approximation with empirical interpolation

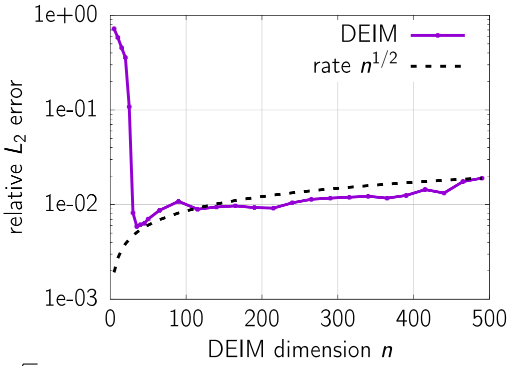
$$\boldsymbol{u} \approx \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^{-1}\boldsymbol{P}^T\boldsymbol{u}$$

Samples with Gaussian noise and std. deviation σ

$$\boldsymbol{u}_\epsilon = \boldsymbol{u} + \boldsymbol{\epsilon}$$

DEIM approximation with perturbed samples

$$\boldsymbol{u} \approx \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^{-1}\boldsymbol{P}^T(\boldsymbol{u} + \boldsymbol{\epsilon})$$



leads to error bound

$$\mathbb{E}\left[\|\boldsymbol{u} - \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^{-1}\boldsymbol{P}^T(\boldsymbol{u} + \boldsymbol{\epsilon})\|_2\right] \lesssim \|\boldsymbol{u} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{u}\|_2 + \sigma\sqrt{n}$$

Instabilities of empirical interpolation and related methods observed in many other works **[Farhat, Cortial, Chapman, 2012], [Farhat, Avery, Chapman, Cortial, 2014], [Drmac, Gugercin, 2016], [Argaud, Bouriquet, Gong, Maday, and Mula, 2017], [Wentland, Huang, Duraisamy, 2021]**

Sampling: Perturbing DEIM approximations

Approximation with empirical interpolation

$$\boldsymbol{u} \approx \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^{-1}\boldsymbol{P}^T\boldsymbol{u}$$

Samples with Gaussian noise and std. deviation σ

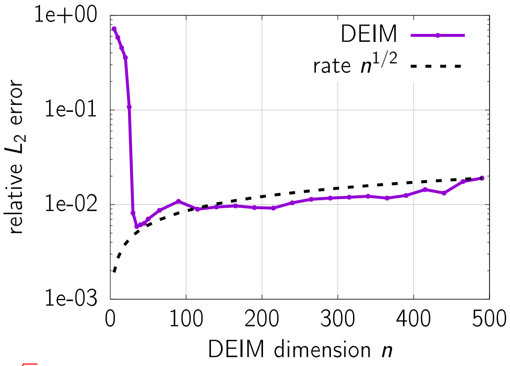
$$\boldsymbol{u}_\epsilon = \boldsymbol{u} + \boldsymbol{\epsilon}$$

DEIM approximation with perturbed samples

$$\boldsymbol{u} \approx \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^{-1}\boldsymbol{P}^T(\boldsymbol{u} + \boldsymbol{\epsilon})$$

leads to error bound

$$\mathbb{E}\left[\|\boldsymbol{u} - \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^{-1}\boldsymbol{P}^T(\boldsymbol{u}+\boldsymbol{\epsilon})\|_2\right] \lesssim \|\boldsymbol{u} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{u}\|_2 + \sigma\sqrt{n}$$



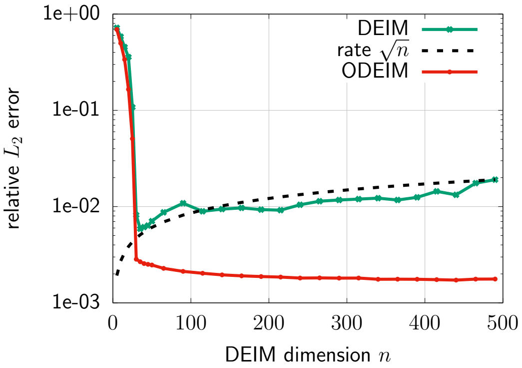
Instabilities of empirical interpolation and related methods observed in many other works [Farhat, Cortial, Chapman, 2012], [Farhat, Avery, Chapman, Cortial, 2014], [Drmac, Gugercin, 2016], [Argaud, Bouriquet, Gong, Maday, and Mula, 2017], [Wentland, Huang, Duraisamy, 2021]

Sampling: Oversampling (gappy POD) stabilizes DEIM

Take more points $m > n$ than basis vectors n

$$\boldsymbol{u} \approx \underbrace{\boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^+}_{\text{regression}} \underbrace{\boldsymbol{P}^T\boldsymbol{u}}_{\text{select}}$$

- Oversampling DEIM leads to gappy POD
  [Everson, Sirovich, 1995], [Astrid, Weiland, Willcox, Backx, 2004, 2008]
- Numerically observed that oversampling helps



Our contribution: Mathematical statement **[P., Drmac, Gugercin, SISC, 2020]**

$$\mathbb{E}\left[\|\boldsymbol{u} - \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^+\boldsymbol{P}^T(\boldsymbol{u} + \epsilon)\right] \lesssim \|\boldsymbol{u} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{u}\|_2 + \sigma\sqrt{\frac{n}{m}}$$

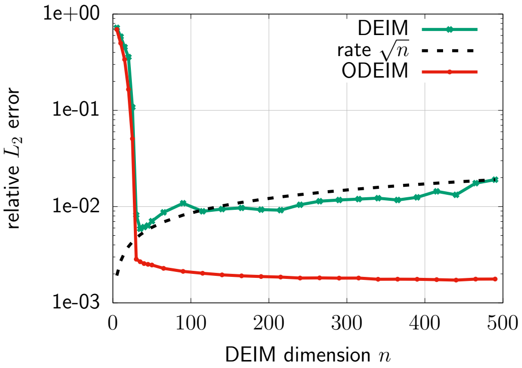
Analysis based on high-dimensional statistics/concentration inequalities gives insights for selecting oversampling points

Sampling: Oversampling (gappy POD) stabilizes DEIM

Take more points $m > n$ than basis vectors n

$$\boldsymbol{u} \approx \underbrace{\boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^+}_{\text{regression}}\underbrace{\boldsymbol{P}^T\boldsymbol{u}}_{\text{select}}$$

- Oversampling DEIM leads to gappy POD
  [Everson, Sirovich, 1995], [Astrid, Weiland, Willcox, Backx, 2004, 2008]
- Numerically observed that oversampling helps



Our contribution: Mathematical statement [P., Drmac, Gugercin, SISC, 2020]

$$\mathbb{E}\left[\|\boldsymbol{u} - \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^+\boldsymbol{P}^T(\boldsymbol{u}+\epsilon)\right] \lesssim \|\boldsymbol{u} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{u}\|_2 + \sigma\sqrt{\frac{n}{m}}$$

Analysis based on high-dimensional statistics/concentration inequalities gives insights for selecting oversampling points

Sampling: The ODEIM

Error bound of DEIM

$$\|\boldsymbol{u} - \boldsymbol{Q}(\boldsymbol{P}^T\boldsymbol{Q})^+\boldsymbol{P}^T\boldsymbol{u}\|_2 \lesssim \|(\boldsymbol{P}^T\boldsymbol{Q})^+\|_2\|\boldsymbol{u} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{u}\|_2$$

- Control the error term $\|\boldsymbol{u} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{u}\|_2$ with the subspace spanned by $\boldsymbol{Q}$
- Control the error term $\|(\boldsymbol{P}^T\boldsymbol{Q})^+\|_2$ with the choice of the sampling points

Goal: Find sampling points matrix $\boldsymbol{P}$ that minimizes

$$\underset{\boldsymbol{P}\in\{0,1\}^{N\times n}}{\arg\min} \|(\boldsymbol{P}^T\boldsymbol{Q})^+\|_2$$

$\rightsquigarrow$ combinatorial complexity in dimension N of full model

[P, Drmac, Gugercin, Stability of discrete empirical interpolation and gappy proper orthogonal decomposition with randomized and deterministic sampling points. SISC, 2020.]

Sampling: Sampling points and singular values

Reformulate the criterion as

$$\|(\boldsymbol{P}^T\boldsymbol{Q})^+\|_2 = s_{\max}((\boldsymbol{P}^T\boldsymbol{Q})^+) = \frac{1}{s_{\min}(\boldsymbol{P}^T\boldsymbol{Q}))}$$

$\rightsquigarrow$ select points that maximize the smallest singular value of $\boldsymbol{P}^T\boldsymbol{Q}$

Consider m points with matrix $\boldsymbol{P}_m$ and the SVD of $\boldsymbol{P}_m^T\boldsymbol{Q}$ as

$$\boldsymbol{P}_m^T\boldsymbol{Q} = \boldsymbol{V}_m\boldsymbol{\Sigma}_m\boldsymbol{W}_m^T$$

Recall that adding a sampling point means including one more row of $\boldsymbol{Q}$

$$\boldsymbol{P}_{m+1}^T\boldsymbol{Q} = \begin{bmatrix}\boldsymbol{P}_m^T\boldsymbol{Q}\\ \boldsymbol{u}_+\end{bmatrix} \in \mathbb{R}^{m+1\times n}$$

Represent $\boldsymbol{P}_{m+1}^T\boldsymbol{Q}$ in terms of SVD of $\boldsymbol{P}_m^T\boldsymbol{Q}$

$$\boldsymbol{P}_{m+1}^T\boldsymbol{Q} = \begin{bmatrix}\boldsymbol{V}_m & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_m\\ \boldsymbol{u}_+\end{bmatrix}\boldsymbol{W}_m^T$$

[P, Drmac, Gugercin, Stability of discrete empirical interpolation and gappy proper orthogonal decomposition with randomized and deterministic sampling points. SISC, 2020.]

Sampling: Updating singular value decompositions

The singular values of $\boldsymbol{P}_{m+1}^T \boldsymbol{Q}$ are the square roots of the eigenvalues of

$$(\boldsymbol{P}_{m+1}^T \boldsymbol{Q})^T (\boldsymbol{P}_{m+1}^T \boldsymbol{Q}) = \boldsymbol{W}_m (\boldsymbol{\Sigma}_m^2 + \boldsymbol{W}_m^T \boldsymbol{u}_+^T \boldsymbol{u}_+ \boldsymbol{W}_m) \boldsymbol{W}_m^T$$

Set $\bar{\boldsymbol{u}}_+ = \boldsymbol{u}_+ \boldsymbol{W}_m$, then have

$$\boldsymbol{\Lambda}_{m+1} = \boldsymbol{\Sigma}_m^2 + \bar{\boldsymbol{u}}_+^T \bar{\boldsymbol{u}}_+$$

- Square roots of EVs of $\boldsymbol{\Lambda}_{m+1}$ are the SVs of $\boldsymbol{P}_{m+1} \boldsymbol{Q}$ ($\boldsymbol{W}_m$ orthonormal; do not change EVs)
- Matrix $\boldsymbol{\Lambda}_{m+1}$ is a symmetric rank-one update of $\boldsymbol{\Sigma}_m^2$, which contains singular values of $\boldsymbol{P}_m^T \boldsymbol{Q}$

Eigenvalues of $\boldsymbol{\Sigma}_m^2$

$$\lambda_1^{(m)} \geq \cdots \geq \lambda_n^{(m)}$$

Eigenvalues of $\boldsymbol{\Lambda}_{m+1}$

$$\lambda_1^{(m+1)} \geq \cdots \geq \lambda_n^{(m+1)}$$

[P, Drmac, Gugercin, Stability of discrete empirical interpolation and gappy proper orthogonal decomposition with randomized and deterministic sampling points. SISC, 2020.]

Sampling: Lower bounds of eigenvalues after rank-one updates

Bound [Ipsen, Nadler, 2009]

$$\lambda_m^{(m+1)} \geq \lambda_m^{(m)} + \frac{1}{2}\left(g + \|\bar{\boldsymbol{u}}_+\|_2^2 - \sqrt{(g + \|\bar{\boldsymbol{u}}_+\|_2^2)^2 - 4g(\boldsymbol{z}_m^{(m)\,T}\bar{\boldsymbol{u}}_+)}\right)$$

- Eigenvalue $\lambda_n^{(m+1)}$ of $\boldsymbol{\Lambda}_{m+1}$
- Eigenvalue $\lambda_n^{(m)}$ of $\boldsymbol{\Sigma}_m$
- Eigengap $g = \lambda_{n-1}^{(m)} - \lambda_n^{(m)}$
- Eigenvector $\boldsymbol{z}_n^{(m)}$ of $\boldsymbol{\Sigma}_m^2$ for $\lambda_n^{(m)}$ ($\boldsymbol{\Sigma}_m$ is diagonal, thus $\boldsymbol{z}_n^{(m)}$ is canonical unit vector)
- Update vector $\bar{\boldsymbol{u}}_+$ that contains selected row $\boldsymbol{u}_+$ of $\boldsymbol{Q}$

$\rightsquigarrow$ all of these quantities are computable

Greedy criterion: Add row $\bar{\boldsymbol{u}}_+$ of $\boldsymbol{Q}$ at step m that maximizes

$$g + \|\bar{\boldsymbol{u}}_+\|_2^2 - \sqrt{(g + \|\bar{\boldsymbol{u}}_+\|_2^2)^2 - 4g(\boldsymbol{z}_m^{(m)\,T}\bar{\boldsymbol{u}}_+)}$$

Sampling: Selecting sampling points with ODEIM

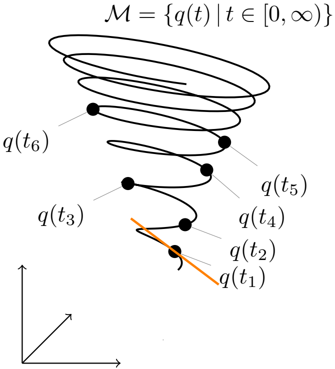
Greedy criterion

$$g + \|\bar{\boldsymbol{u}}_+\|_2^2 - \sqrt{(g + \|\bar{\boldsymbol{u}}_+\|_2^2)^2 - 4g(z_m^{(m)\,T}\bar{\boldsymbol{u}}_+)}$$

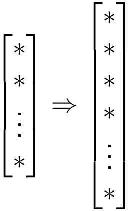
```
function [ p ] = gpode( U, m )
[~, ~, p] = qr(U', 'vector'); p = p(1:size(U, 2))';
for i=length(p)+1:m
    [~, S, W] = svd(U(p, :), 0);
    g = S(end-1, end-1).^2 - S(end, end)^2;
    Ub = W'*U';
    r = g + sum(Ub.^2, 1);
    r = r-sqrt((g+sum(Ub.^2,1)).^2-4*g*Ub(end, :).^2);
    [~, I] = sort(r, 'descend');
    e = 1;
    while any(I(e) == p)
        e = e + 1;
    end
    p(end + 1) = I(e);
end
```

Outline: Online adaptive model reduction

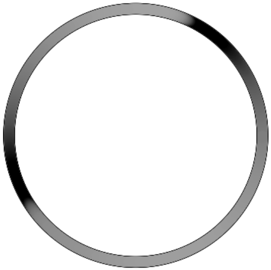
1. Nonlinear approximations via adaptive spaces



2. The importance of sampling ("training")

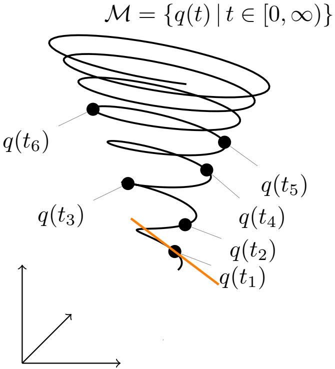


3. Applications

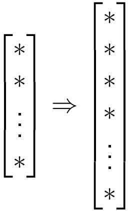


Outline: Online adaptive model reduction

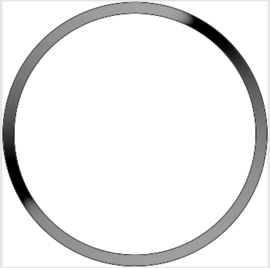
1. Nonlinear approximations via adaptive spaces



2. The importance of sampling ("training")

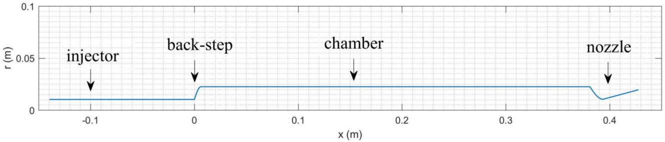


3. Applications

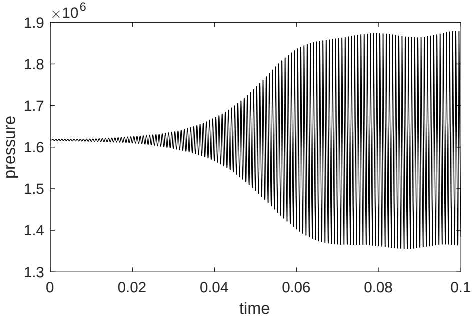


App: Michigan's model combustor

Quasi-1D Eulerian solver with geometry

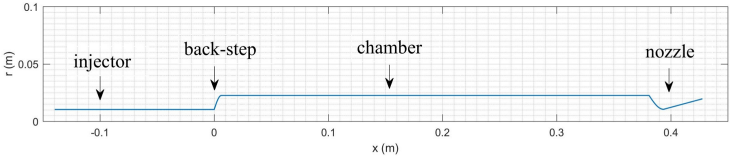


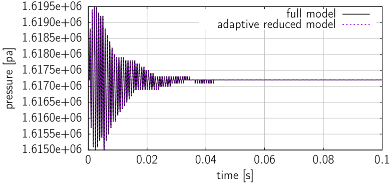
Disturbance added to mass flow rate to excite pressure oscillations



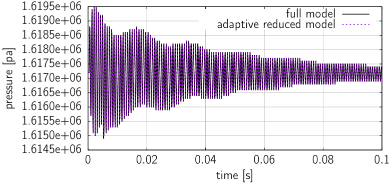
[Frezzotti, Nasuti, Huang, Merkle, Anderson, *Parametric Analysis of Response Function in Modeling Combustion Instability by a Quasi-1D Solver*, 2015], [Frezzotti, Nasuti, Huang, Merkle, Anderson, *Response Function Modeling in the Study of Longitudinal Combustion Instability by a Quasi-1D Eulerian Solver*,2015]

App: Speedup plots for Michigan's model combustor

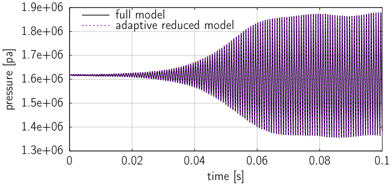


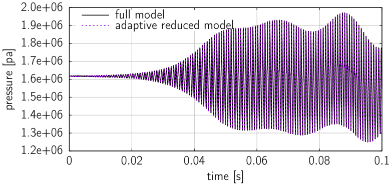


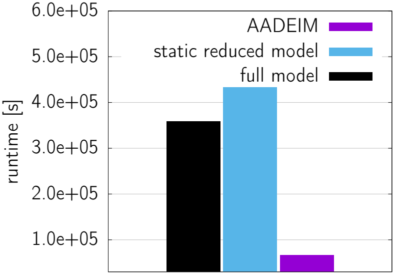
(a) $\mu = 2.4$



(b) $\mu = 3$







- AADEIM 6 times faster than full model
- Adapts to very different dynamics (steady-state, LCO, instability)

App: Huang (Kansas) and Duraisamy (Michigan) groups

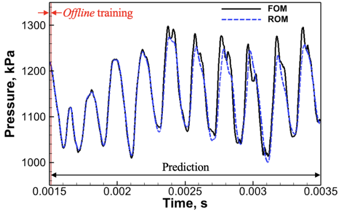
AADEIM ROM: 2D Single-injector Rocket Combustor

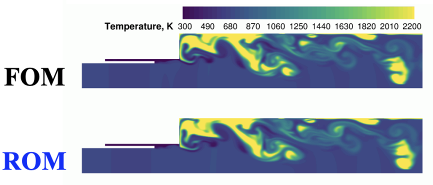
- Dimension: 5
- Sampling points update frequency: 20
- Components sampled: 0.5%

→ ~ **O(20) acceleration *excluding parallel operations***

➤ 0.01*ms offline* training → 2*ms* prediction

***Local* Pressure Time Trace**





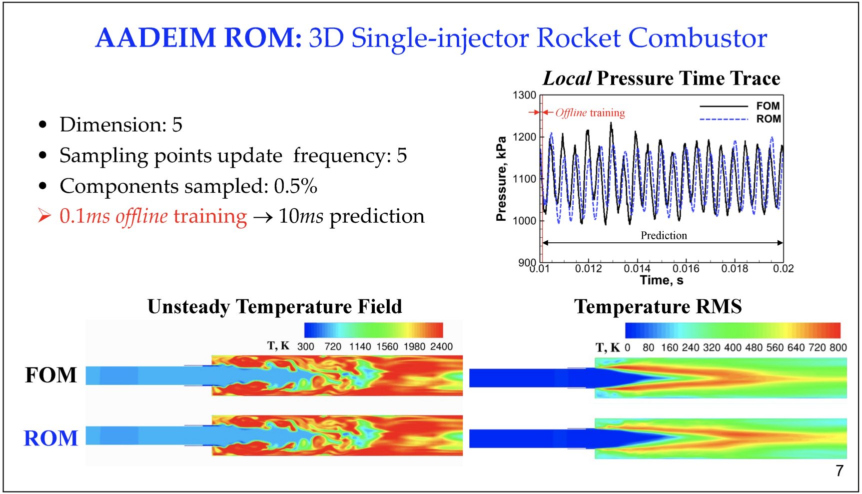
Sampling Points Adaptation

1

- Cheng Huang (Kansas)
- Chris Wentland (Michigan)
- Karthik Duraisamy and group (Michigan)

[Huang, Wentland, Duraisamy, Merkle, Model Reduction for Multi-Scale Transport Problems using Model-form Preserving Least-Squares Projections with Variable Transformation, JCP, 2022], [Wentland, Huang, Duraisamy, Investigation of sampling strategies for reduced-order models of rocket combustors. AIAA, 2021], [Huang, Duraisamy, Merkle, Investigations and improvement of robustness of reduced-order models of reacting flow, AIAA, 2019]

App: Huang (Kansas) and Duraisamy (Michigan) groups



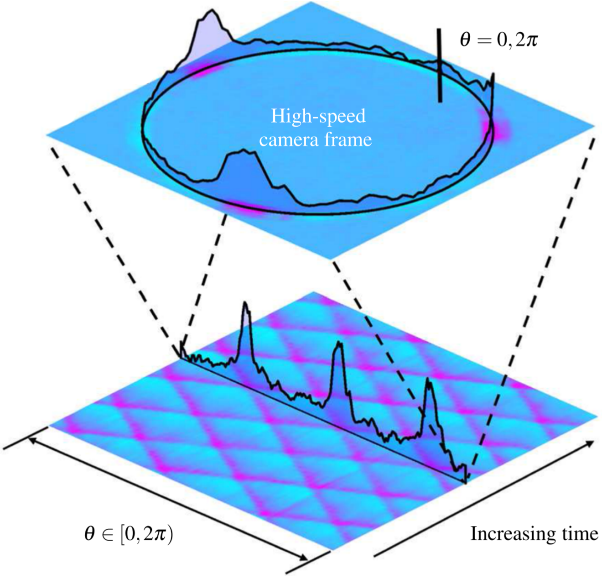
- Cheng Huang (Kansas)
- Chris Wentland (Michigan)
- Karthik Duraisamy and group (Michigan)

[Huang, Wentland, Duraisamy, Merkle, Model Reduction for Multi-Scale Transport Problems using Model-form Preserving Least-Squares Projections with Variable Transformation, JCP, 2022], [Wentland, Huang, Duraisamy, Investigation of sampling strategies for reduced-order models of rocket combustors. AIAA, 2021], [Huang, Duraisamy, Merkle, Investigations and improvement of robustness of reduced-order models of reacting flow, AIAA, 2019]

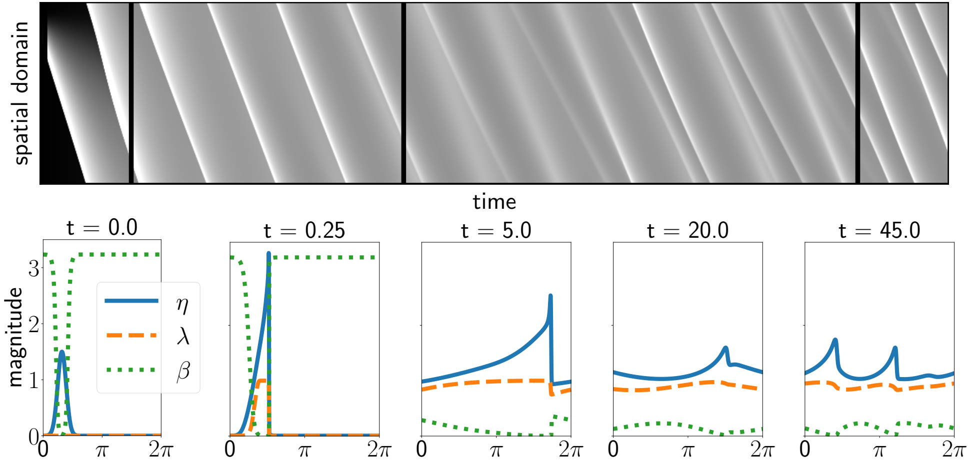
App: Rotating detonation waves

- Models motivated by rotating detonation engines [Koch et al., 2020a, 2020b]
- Single pulse initial condition
- Detonation ("shock") spawns waves
- Waves travel (rotate) over time
- Bifurcations lead to new waves

[Figure: Koch et al., 2020b]



App: Rotating detonation waves



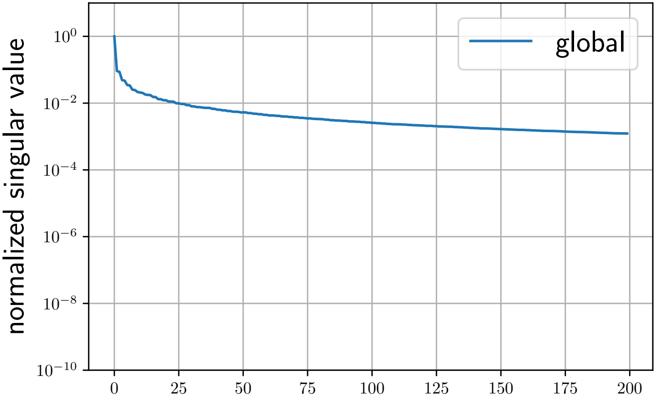
- Detonation (“shock”) from single pulse initial condition
- Wave circulates and spawns second wave eventually

App: Challenges for static model reduction

Having static basis is insufficient

$$\boldsymbol{U} = \begin{bmatrix} | & & | \\ \boldsymbol{u}_1 & \dots & \boldsymbol{u}_n \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times n}$$

- Sharp gradients in solution that travel over time
- Predictions over long time ranges
- Waves travel and new ones are spawned
- Discontinuities in solution



App: Reduced models with AADEIM

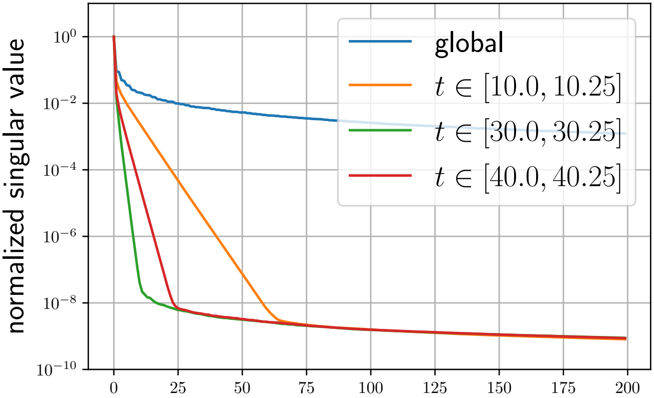
Adapt basis with AADEIM

$$\boldsymbol{U}_{k+1} = \boldsymbol{U}_k + \boldsymbol{\alpha}_k \boldsymbol{\beta}_k^T$$

over time steps $k = 1, \ldots, K$

Prerequisites for AADEIM

- Locally low-rank structure in time
- Local residual in spatial domain (e.g., low number of waves)



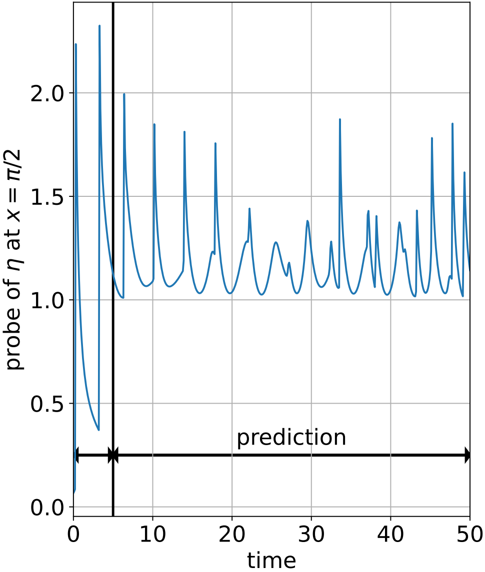
App: Reduced model of rotating waves with AADEIM

Handling shock at beginning

- Run full model to predict shock
- Then switch to AADEIM (vertical line)
- Leaves handling shock at beginning to full model

App: Setup of AADEIM

- Full model dimension $N = 2048$
- Reduced dimension $n = 20$
- Initial window size 5000
- Sample 50% of all components
- Adapt samples every 4th time step



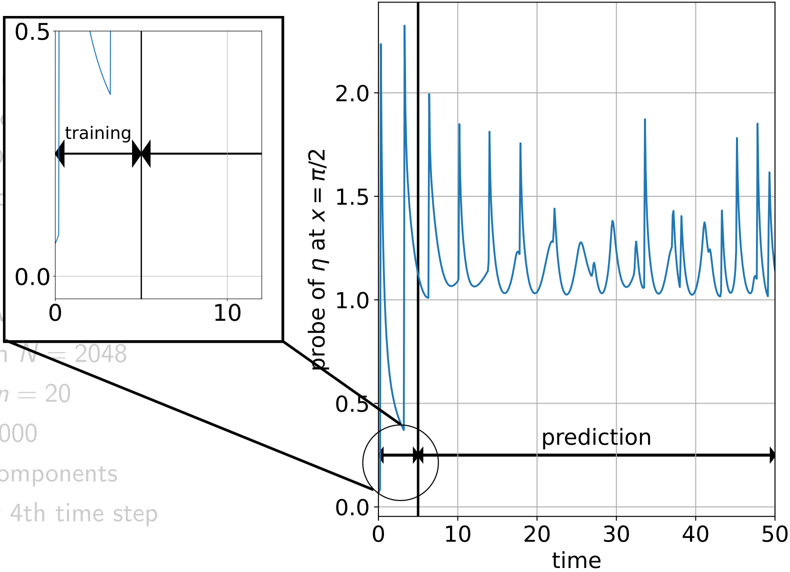
App: Reduced model of rotating waves with AADEIM

Handling shock at begi

- Run full model to pr
- Then switch to AAD
- Leaves handling shoc model

App: Setup of AADEIM

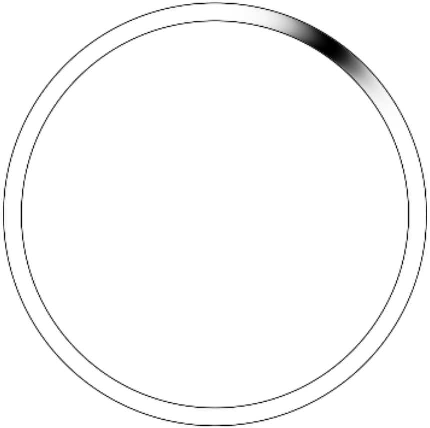
- Full model dimension $N = 2048$
- Reduced dimension $n = 20$
- Initial window size 5000
- Sample 50% of all components
- Adapt samples every 4th time step



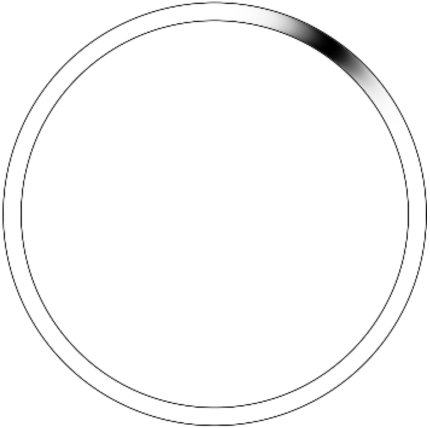
App: AADEIM model of rotating waves

Run full model to predict shock, then switch to AADEIM
AADEIM: dimension 20, sampling points update frequency 4, 50% of components sampled

full model

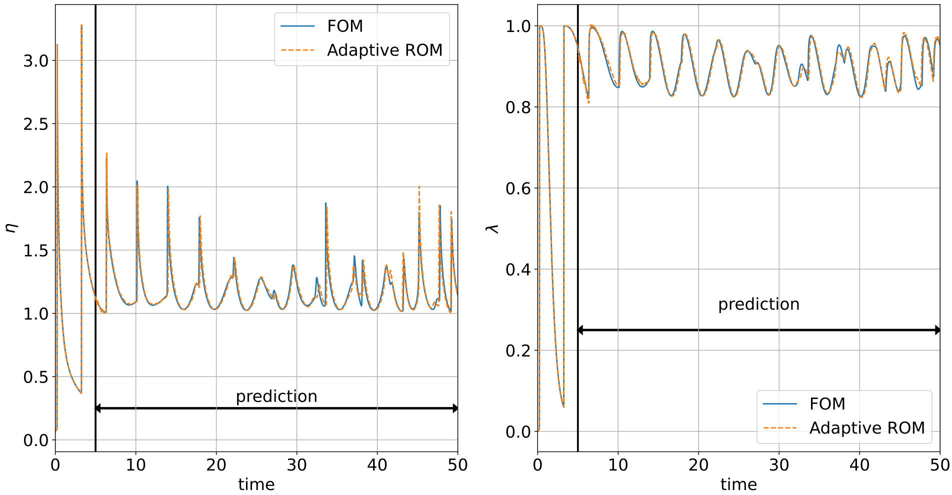


AADEIM

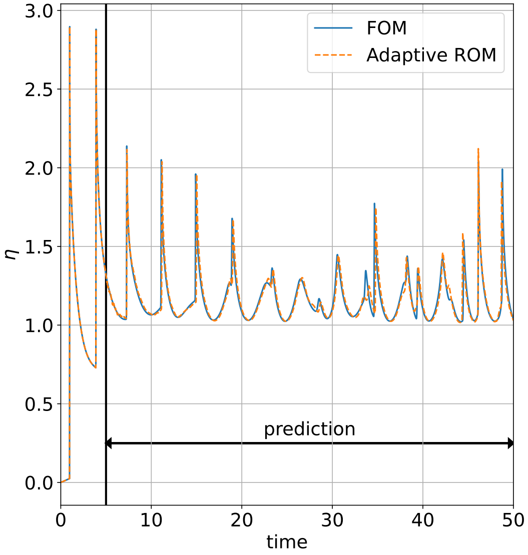


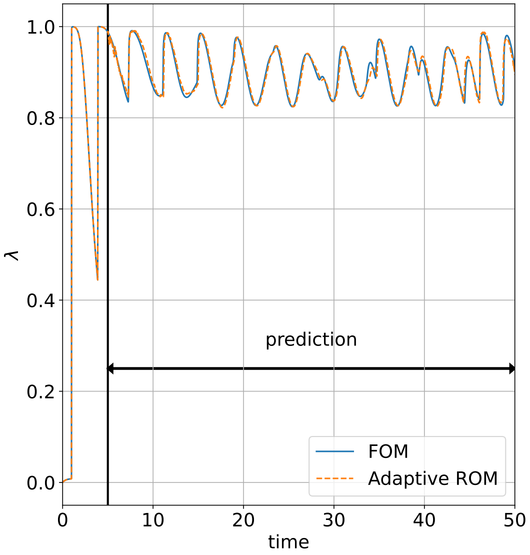
App: Probe at $x = \pi/2$

Run full model to predict shock, then switch to AADEIM (black vertical line)
AADEIM: dimension 20, sampling points update frequency 4, 50% of components sampled

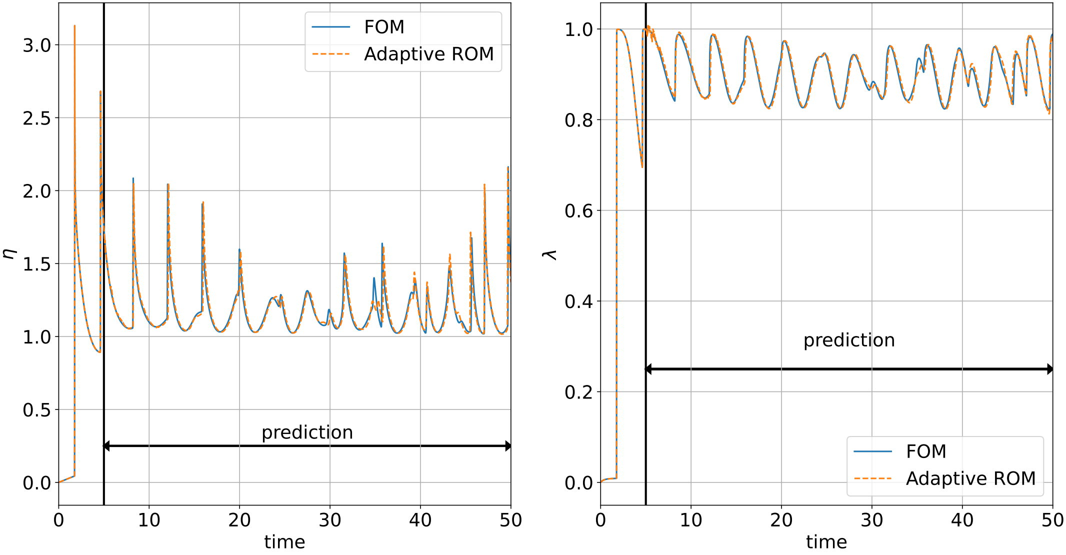


App: Probe at $x = \pi$

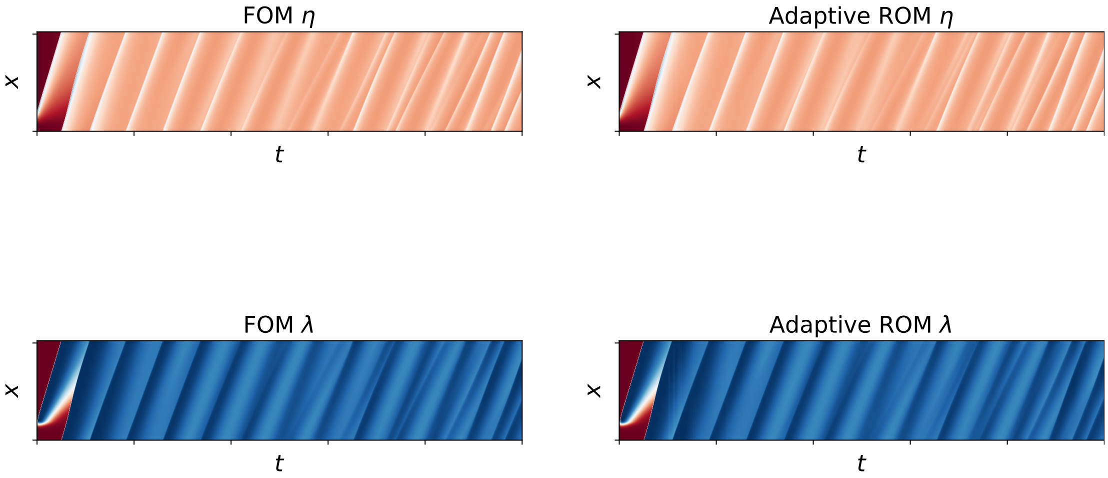




App: Probe at $x = 3\pi/2$



App: Predictions over time and space

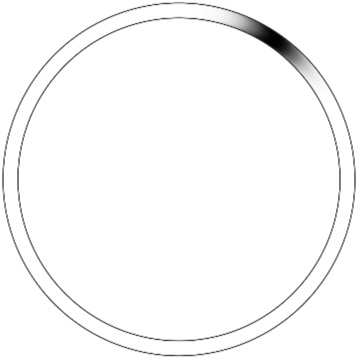


App: Rotating waves with diffusion

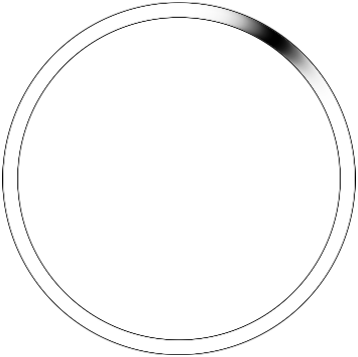
Prevent discontinuities in solution by adding diffusion as suggested by [Koch et al., 2020b]

Allows switching to AADEIM 10$\times$ earlier than without diffusion
AADEIM: dimension 10, sampling points update frequency 3, 40% of components sampled

full model

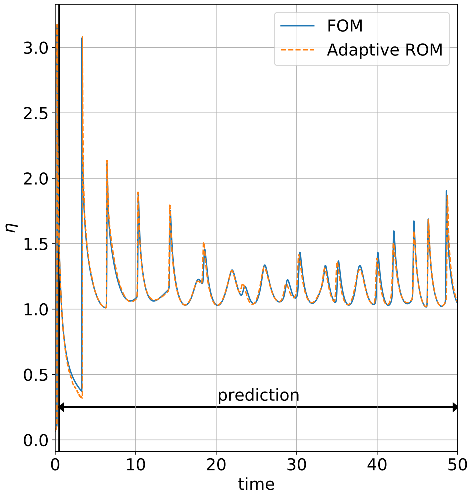


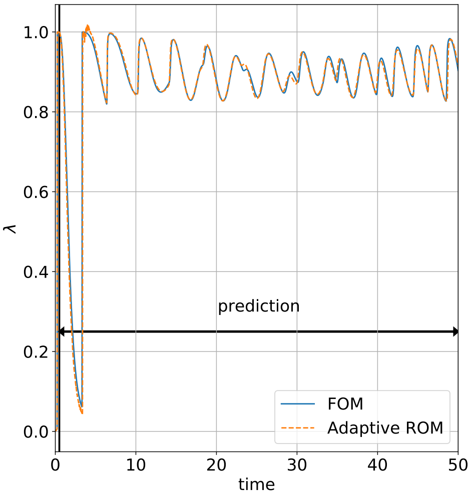
AADEIM



App: Probe at $x = \pi/2$

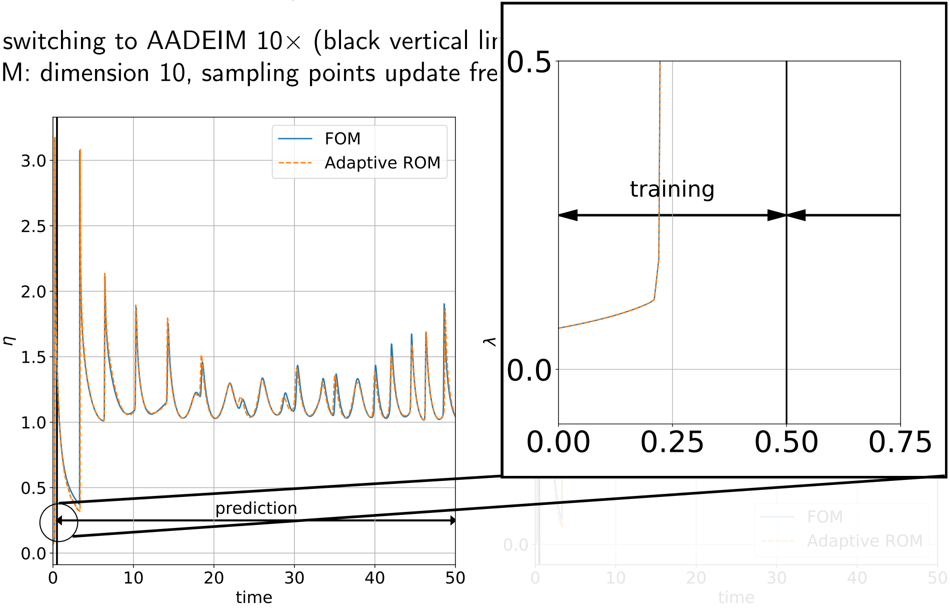
Allows switching to AADEIM 10× (black vertical line) earlier than without diffusion
AADEIM: dimension 10, sampling points update frequency 3, 40% of components sampled



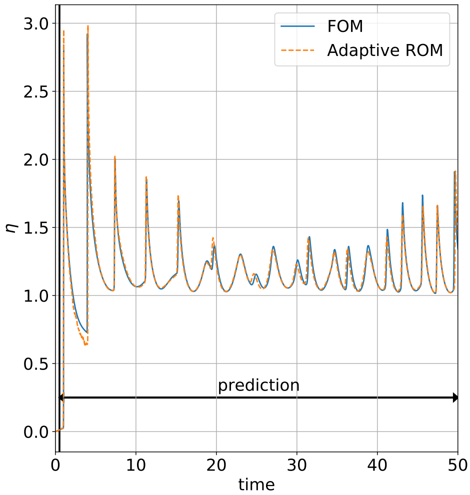


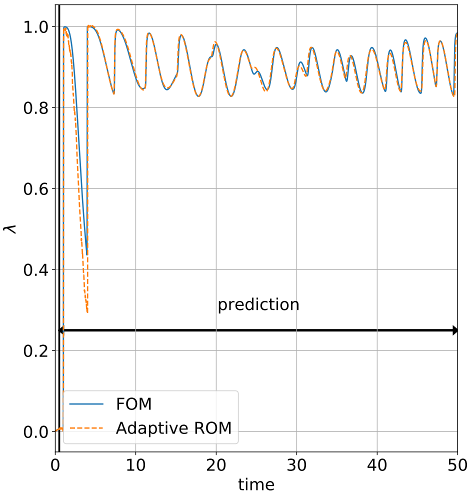
App: Probe at $x = \pi/2$

Allows switching to AADEIM 10$\times$ (black vertical lir
AADEIM: dimension 10, sampling points update fre

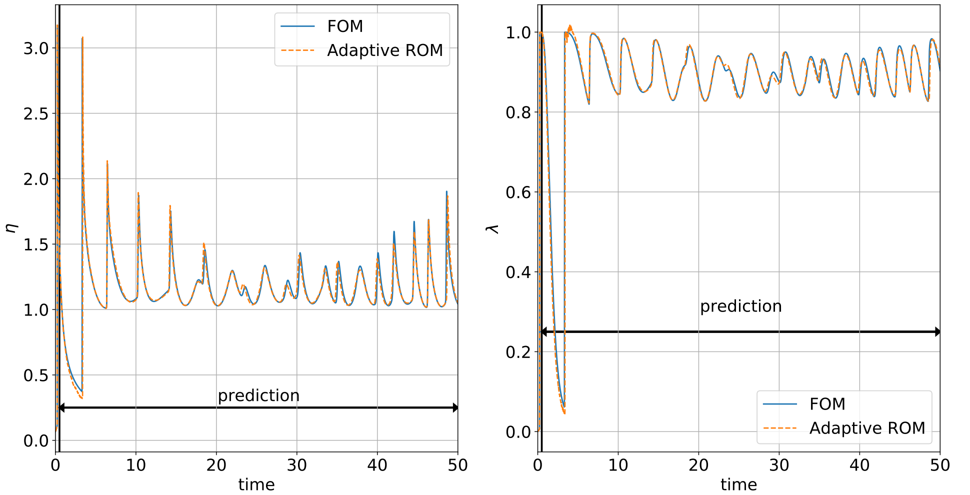


App: Probe at $x = \pi$



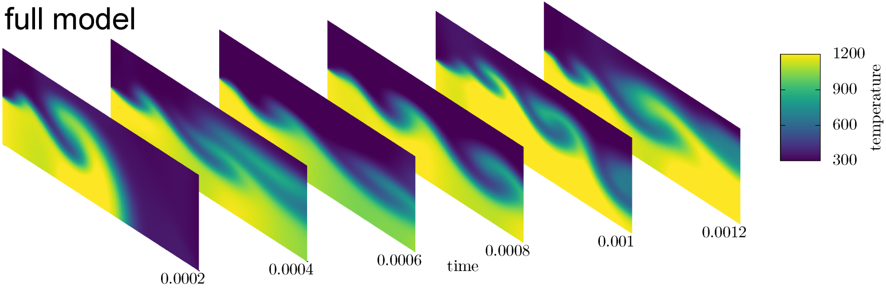


App: Probe at $x = 3\pi/2$

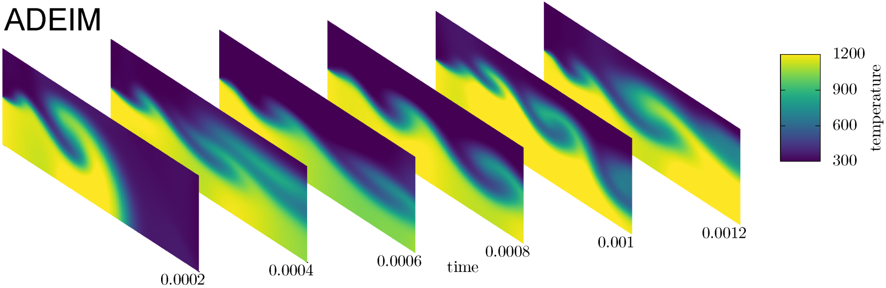


App: Mixing layer

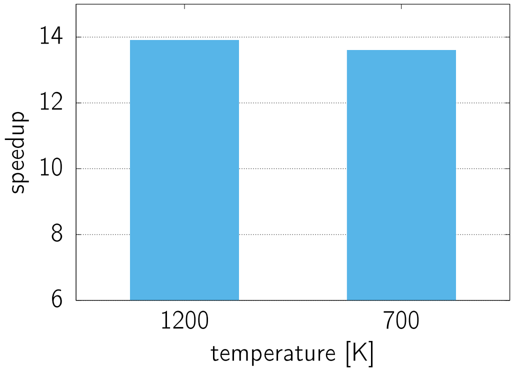
full model



ADEIM



App: Mixing layer: Speedups



Reduced models with ADEIM achieve more than one order of magnitude speedup

- Consider a low- (700K) and high-temperature (1200K) case
- Reduced model is predictive for different parameters of the model

Conclusions

- Reduced models based on AADEIM can handle transport-dominated dynamics
  - True predictions with short offline phase, little training data
  - Can capture transient regimes, rather than just dynamics within a cycle
  - Predictions over multiple parameters because basis adapts to changes in physical parameters
  - Scalable because of many practical improvements

- With AADEIM, full-model solvers stay in the loop to inform adaptation

- Sampling drives adaptation (independent of the used parametrization) and is key

- Shocks remain a major challenge

References:

- **Uy, Wentland, Huang, P., Reduced models with nonlinear approximations of latent dynamics for model premixed flame problems. ArXiv, 2022**
- **P., Model reduction for transport-dominated problems via online adaptive bases and adaptive sampling. SISC, 2020**
- **Cortinovis, Kressner, Massei, P., Quasi-Optimal Sampling to Learn Basis Updates for Online Adaptive Model Reduction with Adaptive Empirical Interpolation. ACC, 2020.**
- **P., Drmac, Gugercin. Stability of discrete empirical interpolation and gappy proper orthogonal decomposition with randomized and deterministic sampling points. SISC, 2020.**