Learning non-intrusive data-driven reduced models: application to large-scale systems

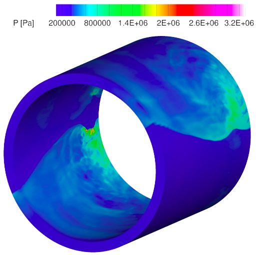
Ionuț-Gabriel Farcaș[1], Rayomand P. Gundevia[2], Ramakanth Munipalli[3], and Karen Willcox[1]

Workshop on Data-driven & Reduced Order Modeling for Multi-Scale Problems

[1]Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX
[2]Jacobs Engineering Group, Inc., Edwards Air Force Base, CA
[3]AFRL Combustion Devices (RQRC), Edwards Air Force Base, CA

August 31, 2023

Learning predictive data-driven reduced models of rotating-detonation rocket engine combustion chambers via Operator Inference



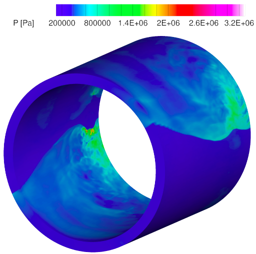
Goals

- construct physics-based data-driven reduced models of large-scale RDRE simulations with sufficient engineering accuracy
- use these ROMs for downstream tasks such as design optimization

Computational challenges

- the high-fidelity LES simulations are large-scale and computationally very expensive (a single simulation typically requires $\mathcal{O}(10^6)$ core-hours on supercomputers)
- the resulting training data sets are often sparse
- they comprise down-sampled time instants from the high-fidelity simulation
- only few parametric instances can realistically be simulated to generate training data

Learning predictive data-driven reduced models of rotating-detonation rocket engine combustion chambers via Operator Inference



Goals

- construct physics-based data-driven reduced models of large-scale RDRE simulations with sufficient engineering accuracy
- use these ROMs for downstream tasks such as design optimization

Computational challenges

- the high-fidelity LES simulations are large-scale and computationally very expensive (a single simulation typically requires $\mathcal{O}(10^6)$ core-hours on supercomputers)
- the resulting training data sets are often sparse
- they comprise down-sampled time instants from the high-fidelity simulation
- only few parametric instances can realistically be simulated to generate training data

Part I: constructing ROMs for large-scale simulations for predictions beyond the training time horizon

Discrete operator inference: general idea

- starting point: a physics-based model, typically described by PDEs or ODEs
- variable transformations that expose polynomial structure in the model
- lens of projection to define the form of a structure-preserving low-dimensional model

} define the structure of the reduced model

Operator inference learning problem

- non-intrusive learning by inferring reduced model operators from data

$$\underset{\hat{\mathbf{O}}}{\operatorname{argmin}} \|\hat{\mathbf{D}}\hat{\mathbf{O}} - \hat{\mathbf{R}}\|_F^2 + \text{regularization}$$

  - $\hat{\mathbf{O}}$: low-dimensional operators define the reduced model as a discrete system
  - $\hat{\mathbf{D}}, \hat{\mathbf{R}}$: data matrix/forcing from simulation and/or experimental data
  - minimum residual formulation leads to linear least-squares minimization
  - regularization is key to reduce overfitting, account for model misspecification etc.

Discrete operator inference: general idea

- starting point: a physics-based model, typically described by PDEs or ODEs
- variable transformations that expose polynomial structure in the model
- lens of projection to define the form of a structure-preserving low-dimensional model

} define the structure of the reduced model

Operator inference learning problem

- non-intrusive learning by inferring reduced model operators from data

$$\underset{\hat{\mathbf{O}}}{\operatorname{argmin}} \|\hat{\mathbf{D}}\hat{\mathbf{O}} - \hat{\mathbf{R}}\|_F^2 + \text{regularization}$$

  - $\hat{\mathbf{O}}$: low-dimensional operators define the reduced model as a discrete system
  - $\hat{\mathbf{D}}, \hat{\mathbf{R}}$: data matrix/forcing from simulation and/or experimental data
  - minimum residual formulation leads to linear least-squares minimization
  - regularization is key to reduce overfitting, account for model misspecification etc.

Steps to perform discrete operator inference

I. Start: full-order model. Suppose the process of interest is described through (possibly after applying a lifting transformation) the high-dimensional quadratic discrete model

$$\mathbf{w}[k+1] = \mathbf{A}\mathbf{w}[k] + \mathbf{H}(\mathbf{w}[k] \otimes \mathbf{w}[k])$$

II. Discrete operator inference:

1. training data: compute a set of n_t full-order model solutions (snapshots)
2. training data manipulation (lifting, centering, scaling) to get the snapshot matrix of the transformed variables $\mathbf{Q} \in \mathbb{R}^{n \times n_t}$
3. subspace identification: determine rank-r reduced basis $\mathbf{V}_r \in \mathbb{R}^{n \times r}$ via the (thin) singular value decomposition of $\mathbf{Q}$
4. projection: project each transformed snapshot: $\hat{\mathbf{Q}}[k] = \mathbf{V}_r^\top \mathbf{Q}[k] \in \mathbb{R}^{r \times n_t}$
5. learn reduced operators $\hat{\mathbf{A}} \in \mathbb{R}^{r \times r}$, $\hat{\mathbf{H}} \in \mathbb{R}^{r \times r^2}$, and $\hat{\mathbf{c}} \in \mathbb{R}^r$ via operator inference with regularization

III. Finish: reduced-order model. The learned reduced operators define the ROM

$$\hat{\mathbf{q}}[k+1] = \hat{\mathbf{A}}\hat{\mathbf{q}}[k] + \hat{\mathbf{H}}\left(\hat{\mathbf{q}}[k] \otimes \hat{\mathbf{q}}[k]\right) + \hat{\mathbf{c}}$$

We then use the reduced model to issue predictions beyond the training time horizon

Steps to perform discrete operator inference

I. Start: full-order model. Suppose the process of interest is described through (possibly after applying a lifting transformation) the high-dimensional quadratic discrete model

$$\mathbf{w}[k+1] = \mathbf{A}\mathbf{w}[k] + \mathbf{H}(\mathbf{w}[k] \otimes \mathbf{w}[k])$$

II. Discrete operator inference:

1. training data: compute a set of n_t full-order model solutions (snapshots)
2. training data manipulation (lifting, centering, scaling) to get the snapshot matrix of the transformed variables $\mathbf{Q} \in \mathbb{R}^{n \times n_t}$
3. subspace identification: determine rank-r reduced basis $\mathbf{V}_r \in \mathbb{R}^{n \times r}$ via the (thin) singular value decomposition of $\mathbf{Q}$
4. projection: project each transformed snapshot: $\hat{\mathbf{Q}}[k] = \mathbf{V}_r^\top \mathbf{Q}[k] \in \mathbb{R}^{r \times n_t}$
5. learn reduced operators $\hat{\mathbf{A}} \in \mathbb{R}^{r \times r}$, $\hat{\mathbf{H}} \in \mathbb{R}^{r \times r^2}$, and $\hat{\mathbf{c}} \in \mathbb{R}^r$ via operator inference with regularization

III. Finish: reduced-order model. The learned reduced operators define the ROM

$$\hat{\mathbf{q}}[k+1] = \hat{\mathbf{A}}\hat{\mathbf{q}}[k] + \hat{\mathbf{H}}\left(\hat{\mathbf{q}}[k] \otimes \hat{\mathbf{q}}[k]\right) + \hat{\mathbf{c}}$$

We then use the reduced model to issue predictions beyond the training time horizon

Steps to perform discrete operator inference

I. Start: full-order model. Suppose the process of interest is described through (possibly after applying a lifting transformation) the high-dimensional quadratic discrete model

$$\mathbf{w}[k+1] = \mathbf{A}\mathbf{w}[k] + \mathbf{H}(\mathbf{w}[k] \otimes \mathbf{w}[k])$$

II. Discrete operator inference:

1. training data: compute a set of n_t full-order model solutions (snapshots)
2. training data manipulation (lifting, centering, scaling) to get the snapshot matrix of the transformed variables $\mathbf{Q} \in \mathbb{R}^{n \times n_t}$
3. subspace identification: determine rank-r reduced basis $\mathbf{V}_r \in \mathbb{R}^{n \times r}$ via the (thin) singular value decomposition of $\mathbf{Q}$
4. projection: project each transformed snapshot: $\hat{\mathbf{Q}}[k] = \mathbf{V}_r^\top \mathbf{Q}[k] \in \mathbb{R}^{r \times n_t}$
5. learn reduced operators $\hat{\mathbf{A}} \in \mathbb{R}^{r \times r}$, $\hat{\mathbf{H}} \in \mathbb{R}^{r \times r^2}$, and $\hat{\mathbf{c}} \in \mathbb{R}^r$ via operator inference with regularization

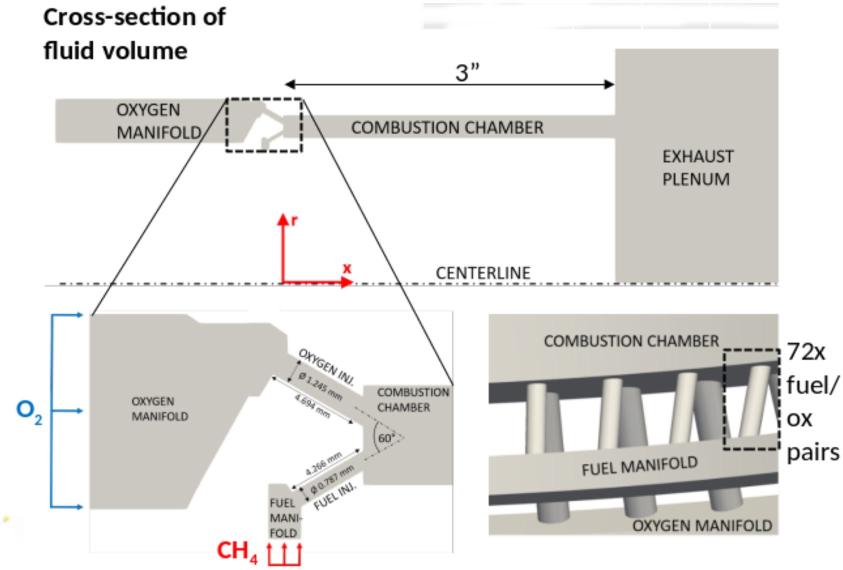
III. Finish: reduced-order model. The learned reduced operators define the ROM

$$\hat{\mathbf{q}}[k+1] = \hat{\mathbf{A}}\hat{\mathbf{q}}[k] + \hat{\mathbf{H}}\left(\hat{\mathbf{q}}[k] \otimes \hat{\mathbf{q}}[k]\right) + \hat{\mathbf{c}}$$

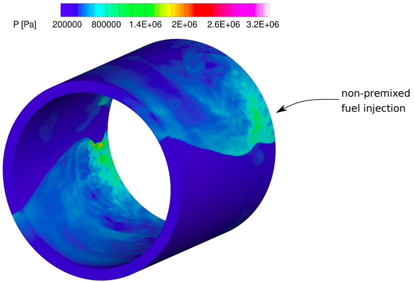
We then use the reduced model to issue predictions beyond the training time horizon

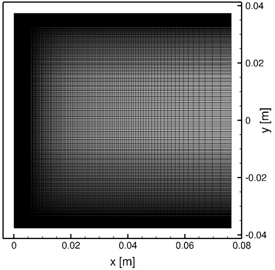
Numerical demonstration

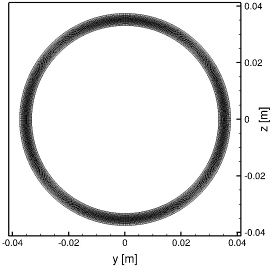
Modeling the combustion chamber of a rotating detonation rocket engine



Modeling the combustion chamber of a rotating detonation rocket engine

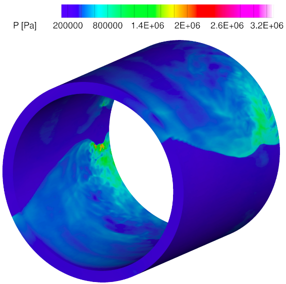


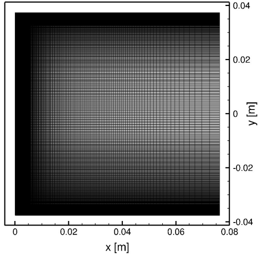


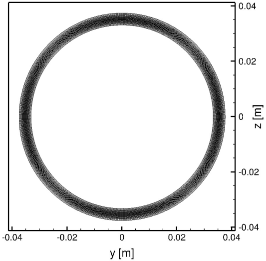


Modeling the combustion chamber of a rotating detonation rocket engine

- LES simulations of the reactive, viscous 3D Navier-Stokes equations
- skeletal chemistry mechanism based on the Foundational Fuel Chemistry Model (FFCMy-30)
- non-premixed fuel injection (gaseous methane and oxygen)
- injector design with 72 discrete injector pairs



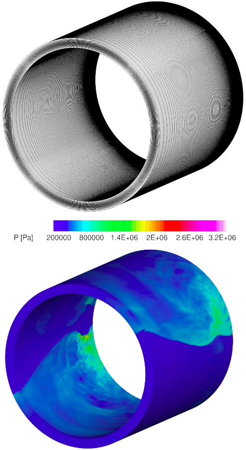




Modeling the combustion chamber of a rotating detonation rocket engine

1. Generate training data + data manipulation

- $\dot{m} = 0.267\ \mathrm{kg} \cdot \mathrm{s}^{-1}$ and $\Phi = 1.16$
- 2 ms of full-state solutions generated ~6M CPU hours on $>$ 16K cores
- the original simulation data has been interpolated on a structured mesh comprising $n_x = 4,204,200$ spatial DoF
- time step size $\Delta t \sim 10^{-9}$ s
- available data: 501 down-sampled snapshots over $[2.50, 3.75]$ ms (~4 periods of two-wave system)
- 18 transformed state variables: specific volume, pressure, 3D velocity, temperature, 12 species mass fractions (full chemistry data)
- training data: snapshot matrix $\mathbf{Q} \in \mathbb{R}^{76M \times 375}$

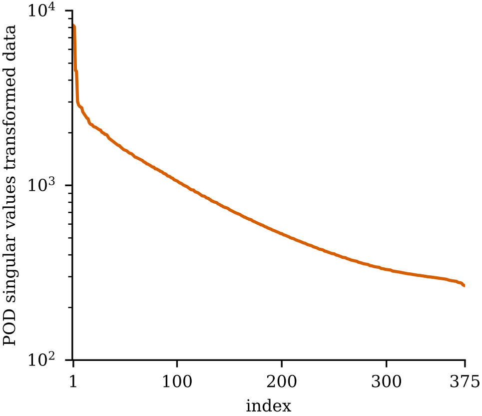


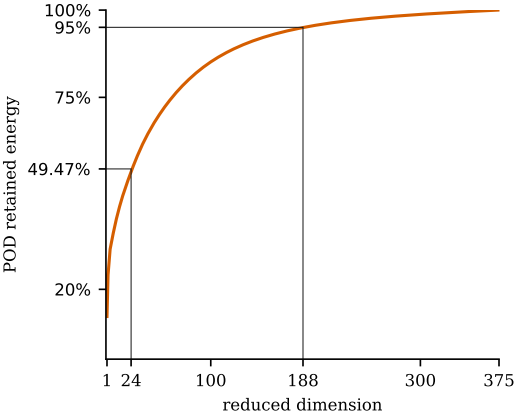
Two dominant co-rotating waves in the quasi-limit-cycle behavior of the flow

Modeling the combustion chamber of a rotating detonation rocket engine

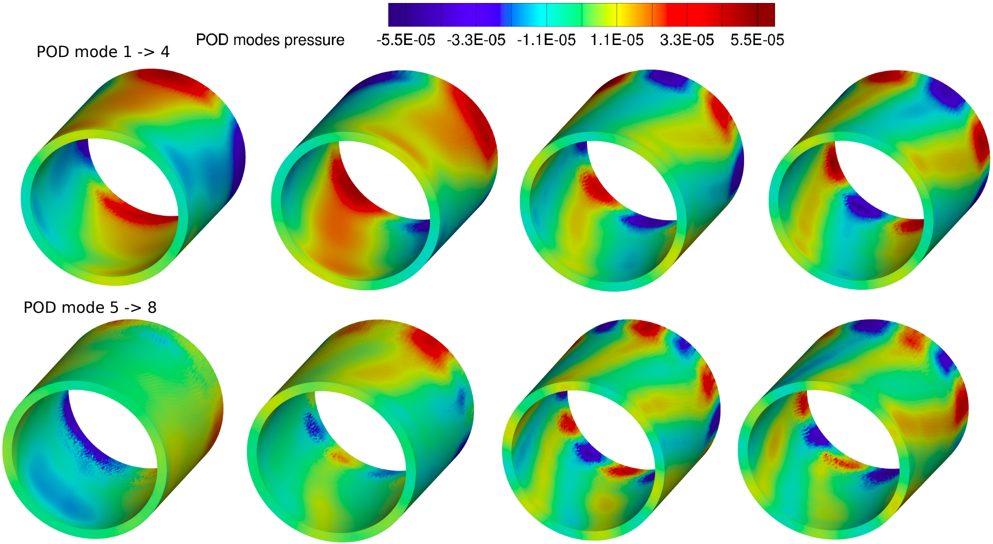
2. Compute POD basis

- snapshot matrix of transformed variables
- scale and center snapshot data
- compute POD basis $\mathbf{V}_r \in \mathbb{R}^{76M \times r}$
- low-data regime limits size of the non-intrusive ROM



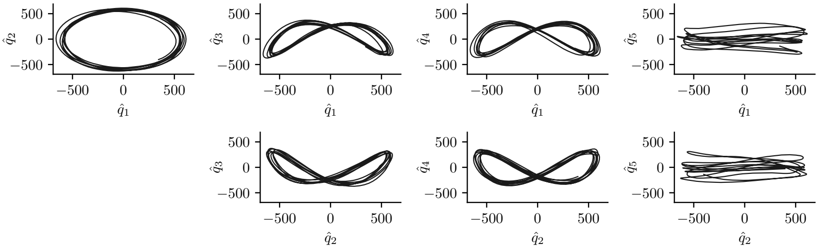


Modeling the combustion chamber of a rotating detonation rocket engine

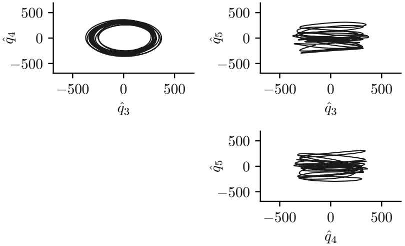


Modeling the combustion chamber of a rotating detonation rocket engine

Two-dimensional phase portraits of POD amplitudes



note that these amplitudes represent the data used in the OpInf inference problem

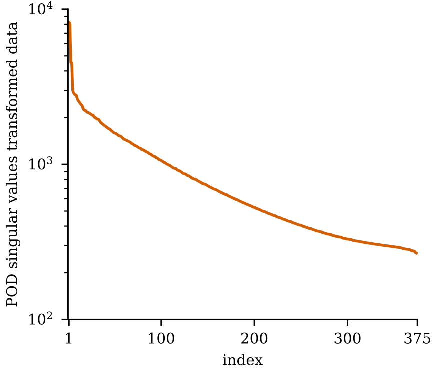


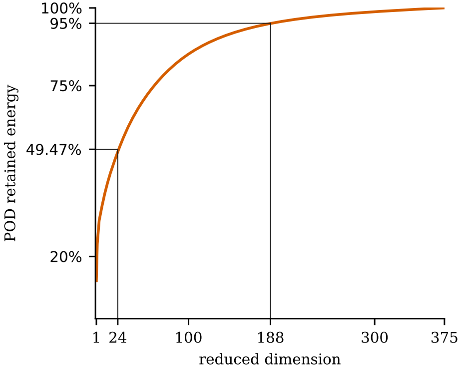
Modeling the combustion chamber of a rotating detonation rocket engine

3. Infer reduced operators

- compute reduced snapshot matrix $\hat{\mathbf{Q}} = \mathbf{V}_r^\top \mathbf{Q} \in \mathbb{R}^{r \times 375}$

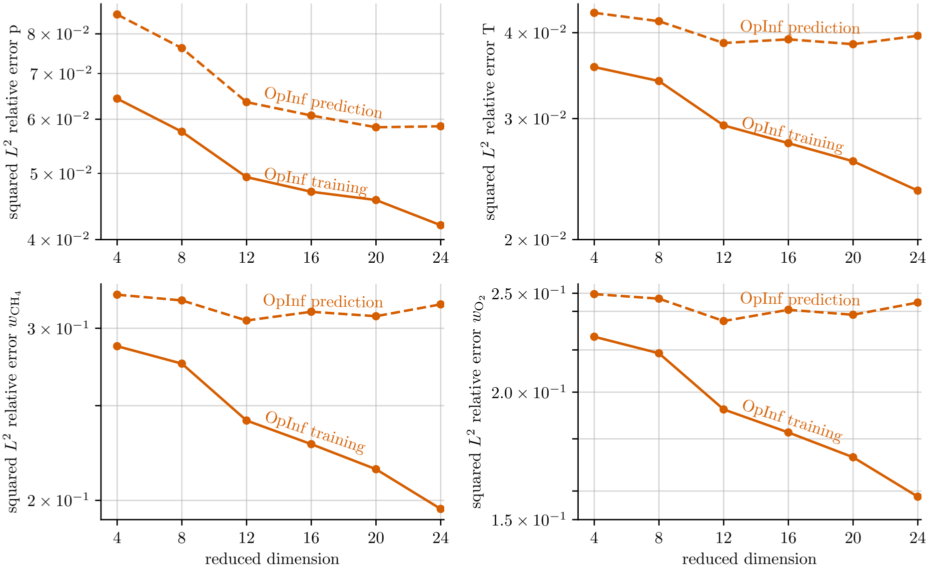
- learn a fully discrete quadratic ROM
- solve linear least squares to infer reduced operators



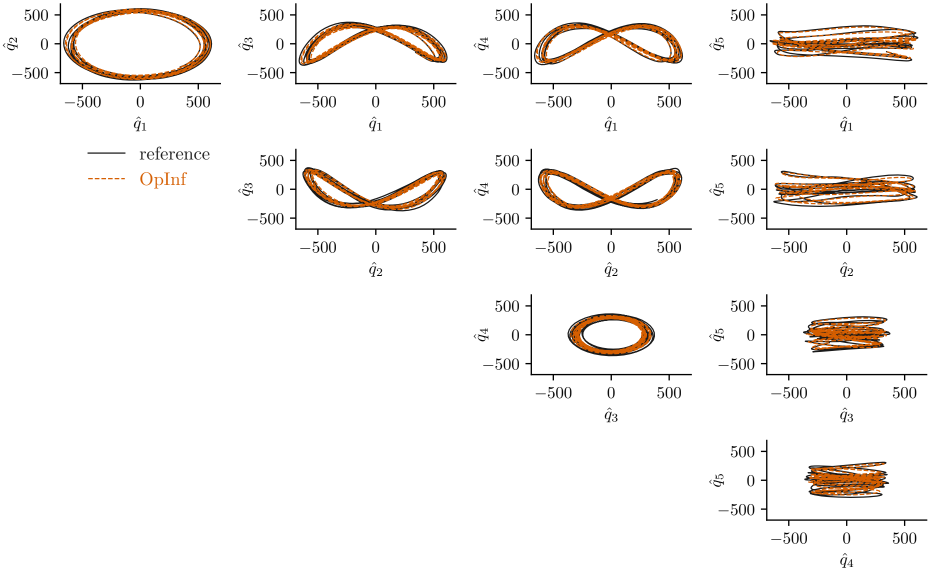


resulting reduced model is completely decoupled from the original CFD code

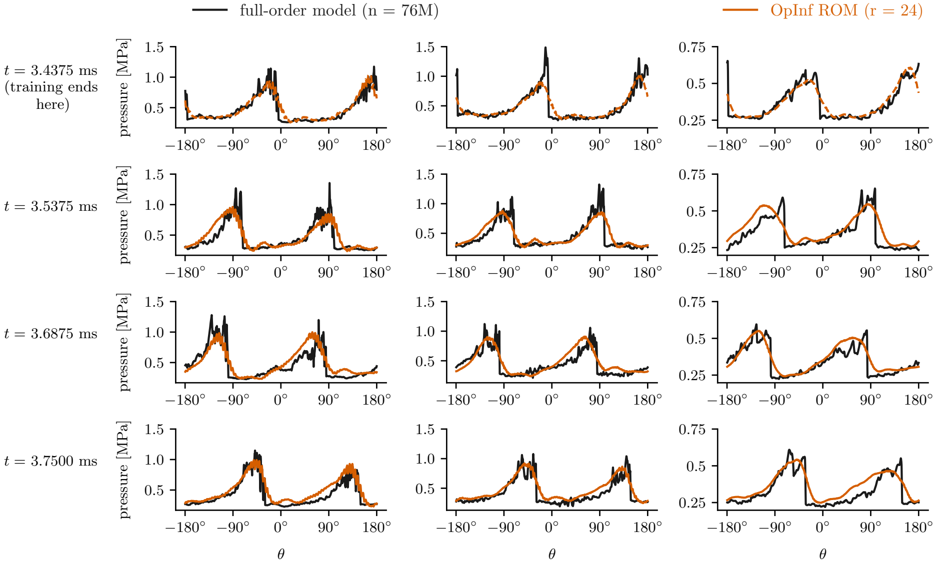
OpInf reduced model relative error



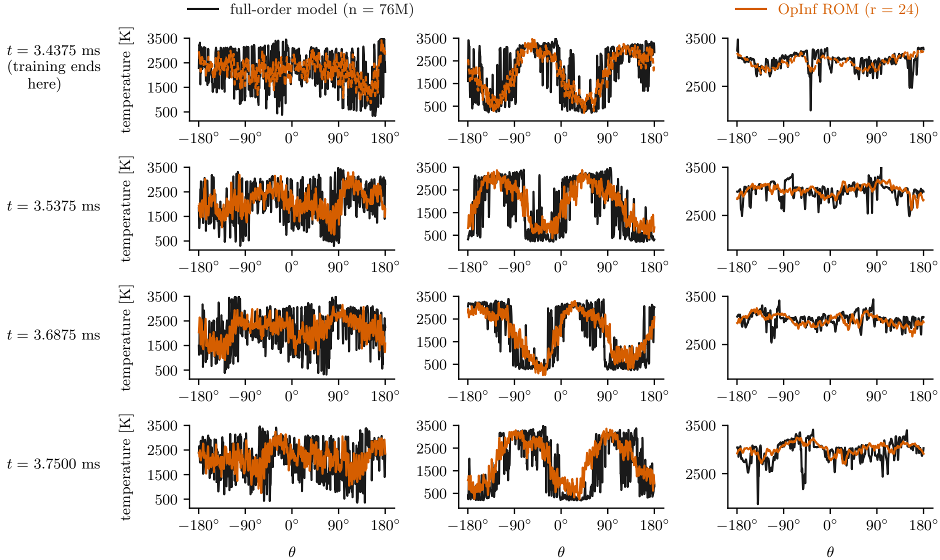
OpInf predictions for two-dimensional phase portraits



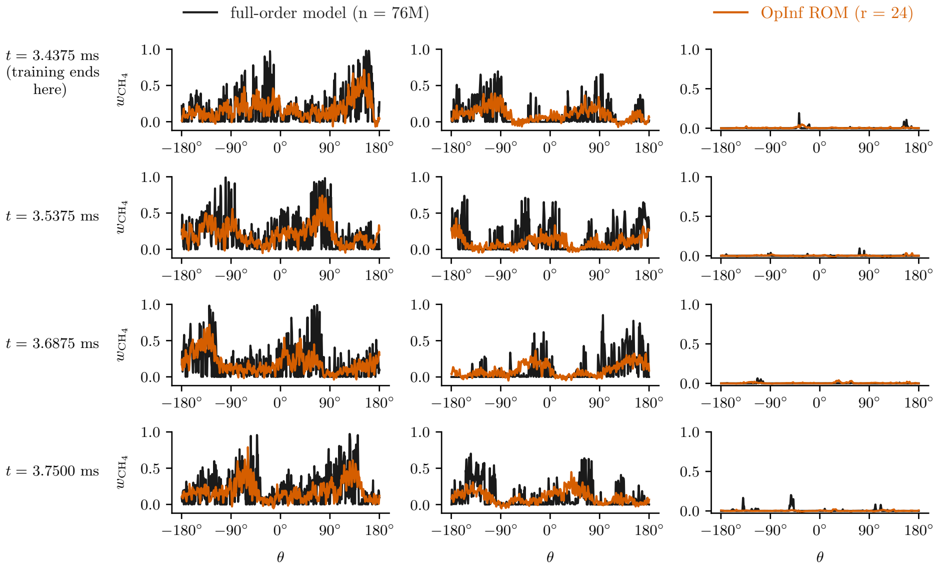
One-dimensional circumferential pressure profiles



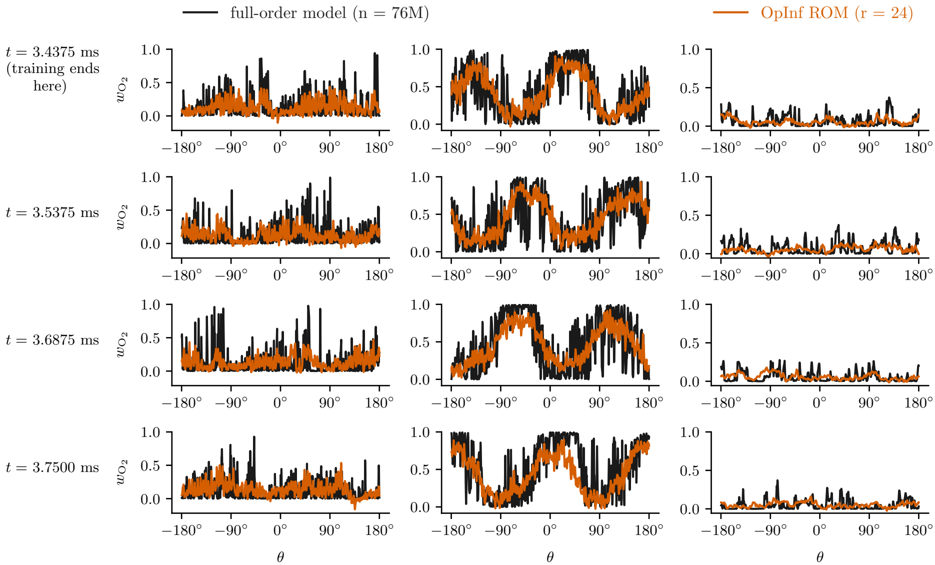
One-dimensional circumferential temperature profiles



One-dimensional circumferential fuel mass fraction profiles



One-dimensional circumferential oxidizer mass fraction profiles



Part II: towards predictive parametric ROMs of large-scale RDRE combustion chamber simulations

Parametric discrete operator inference: general idea

- a parametric physics-based model, typically described by PDEs or ODEs
- variable transformations that expose polynomial structure in the model
- lens of projection to define the form of a structure-preserving low-dimensional parametric model

} define the structure of the parametric reduced model

Discrete parametric operator inference learning problem

- non-intrusive learning by inferring reduced model parametric operators from data

$$\underset{\hat{\mathbf{O}}(\boldsymbol{\mu})}{\operatorname{argmin}} \|\hat{\mathbf{D}}\hat{\mathbf{O}}(\boldsymbol{\mu}) - \hat{\mathbf{R}}\|_F^2 + \text{regularization}$$

  - $\hat{\mathbf{O}}(\boldsymbol{\mu})$: low-dimensional parametric operators define the reduced model as a discrete system
  - $\hat{\mathbf{D}}, \hat{\mathbf{R}}$: data matrix/forcing from simulation and/or experimental data
  - minimum residual formulation leads to linear least-squares minimization
  - regularization is key to reduce overfitting, account for model misspecification etc.

Parametric discrete operator inference: general idea

- a parametric physics-based model, typically described by PDEs or ODEs
- variable transformations that expose polynomial structure in the model
- lens of projection to define the form of a structure-preserving low-dimensional parametric model

} define the structure of the parametric reduced model

Discrete parametric operator inference learning problem

- non-intrusive learning by inferring reduced model parametric operators from data

$$\underset{\hat{\mathbf{O}}(\boldsymbol{\mu})}{\operatorname{argmin}} \|\hat{\mathbf{D}}\hat{\mathbf{O}}(\boldsymbol{\mu}) - \hat{\mathbf{R}}\|_F^2 + \text{regularization}$$

  - $\hat{\mathbf{O}}(\boldsymbol{\mu})$: low-dimensional parametric operators define the reduced model as a discrete system
  - $\hat{\mathbf{D}}, \hat{\mathbf{R}}$: data matrix/forcing from simulation and/or experimental data
  - minimum residual formulation leads to linear least-squares minimization
  - regularization is key to reduce overfitting, account for model misspecification etc.

Steps to perform parametric discrete operator inference

I. Start: full-order model that depends on a d-dimensional parameter $\boldsymbol{\mu} \in \mathcal{D} \subset \mathbb{R}^d$. Suppose that the process of interest can be described (possibly after a lifting transformation) through the high-dimensional parametric quadratic discrete model

$$\mathbf{w}[k+1] = \mathbf{A}(\boldsymbol{\mu})\mathbf{w}[k] + \mathbf{H}(\boldsymbol{\mu})\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)$$

We seek a to construct a parametric reduced model via OpInf with reduced dimension $r \ll n$.

To this end, we consider a quadratic-bilinear model form for the parametric model:

$$\mathbf{w}[k+1] = \underbrace{\mathbf{A}_0\mathbf{w}[k]}_{\text{linear}} + \underbrace{\sum_{i=1}^{d} \mu_i \mathbf{A}_i \mathbf{w}[k]}_{\text{bilinear}} + \underbrace{\mathbf{H}_0\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)}_{\text{quadratic}} + \underbrace{\sum_{i=1}^{d} \mu_i \mathbf{H}_i\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)}_{\text{quadratic-linear}},$$

which is obtained by assuming an affine dependency on the parameters, Taylor series approximation of the parametric operators etc.

Steps to perform parametric discrete operator inference

I. Start: full-order model that depends on a d-dimensional parameter $\boldsymbol{\mu} \in \mathcal{D} \subset \mathbb{R}^d$. Suppose that the process of interest can be described (possibly after a lifting transformation) through the high-dimensional parametric quadratic discrete model

$$\mathbf{w}[k+1] = \mathbf{A}(\boldsymbol{\mu})\mathbf{w}[k] + \mathbf{H}(\boldsymbol{\mu})\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)$$

We seek a to construct a parametric reduced model via OpInf with reduced dimension $r \ll n$.

To this end, we consider a quadratic-bilinear model form for the parametric model:

$$\mathbf{w}[k+1] = \underbrace{\mathbf{A}_0\mathbf{w}[k]}_{\text{linear}} + \underbrace{\sum_{i=1}^{d} \mu_i \mathbf{A}_i \mathbf{w}[k]}_{\text{bilinear}} + \underbrace{\mathbf{H}_0\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)}_{\text{quadratic}} + \underbrace{\sum_{i=1}^{d} \mu_i \mathbf{H}_i\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)}_{\text{quadratic-linear}},$$

which is obtained by assuming an affine dependency on the parameters, Taylor series approximation of the parametric operators etc.

Steps to perform parametric discrete operator inference

I. Start: full-order model that depends on a d-dimensional parameter $\boldsymbol{\mu} \in \mathcal{D} \subset \mathbb{R}^d$. Suppose that the process of interest can be described (possibly after a lifting transformation) through the high-dimensional parametric quadratic discrete model

$$\mathbf{w}[k+1] = \mathbf{A}(\boldsymbol{\mu})\mathbf{w}[k] + \mathbf{H}(\boldsymbol{\mu})\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)$$

We seek a to construct a parametric reduced model via OpInf with reduced dimension $r \ll n$.

To this end, we consider a quadratic-bilinear model form for the parametric model:

$$\mathbf{w}[k+1] = \underbrace{\mathbf{A}_0\mathbf{w}[k]}_{\text{linear}} + \underbrace{\sum_{i=1}^{d} \mu_i \mathbf{A}_i \mathbf{w}[k]}_{\text{bilinear}} + \underbrace{\mathbf{H}_0\left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)}_{\text{quadratic}} + \underbrace{\sum_{i=1}^{d} \mu_i \mathbf{H}_i \left(\mathbf{w}[k] \otimes \mathbf{w}[k]\right)}_{\text{quadratic-linear}},$$

which is obtained by assuming an affine dependency on the parameters, Taylor series approximation of the parametric operators etc.

Steps to perform parametric discrete operator inference

II. Parametric OpInf for the quadratic-bilinear ROM:

1. training data: compute n_t full-order model snapshots for each training parameter instance $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_m \in \mathbb{R}^d$
2. training data manipulation (lifting, centering, scaling) to obtain the snapshot matrix of transformed data $\mathbf{Q} \in \mathbb{R}^{n \times m n_t}$
3. subspace identification: rank-r global POD basis $\mathbf{V}_r \in \mathbb{R}^{n \times r}$ using the thin SVD of $\mathbf{Q}$
4. projection: project each snapshot to obtain $\hat{\mathbf{Q}} = \mathbf{V}_r^\top \mathbf{Q} \in \mathbb{R}^{r \times m n_t}$
5. learn reduced operators $\hat{\mathbf{A}}_i$, $\hat{\mathbf{H}}_i$ and $\hat{\mathbf{c}}_i$ for $i = 0, 1, \dots, d$ via operator inference with regularization

Steps to perform parametric discrete operator inference

III. Finish: parametric reduced-order model.
The learned reduced operators define the parametric quadratic-bilinear reduced model

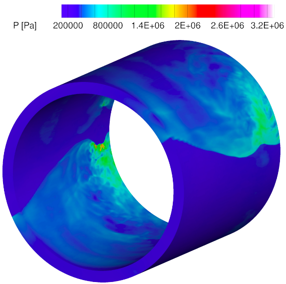
$$\hat{\mathbf{q}}[k+1] = \hat{\mathbf{A}}_0\hat{\mathbf{q}}[k] + \sum_{i=1}^{d} \mu_i \hat{\mathbf{A}}_i \hat{\mathbf{s}}[k] + \hat{\mathbf{H}}_0 \left(\hat{\mathbf{q}}[k] \otimes \hat{\mathbf{q}}[k]\right) + \sum_{i=1}^{d} \mu_i \hat{\mathbf{H}}_i \left(\hat{\mathbf{q}}[k] \otimes \hat{\mathbf{q}}[k]\right) + \hat{\mathbf{c}}_0 + \sum_{i=1}^{d} \mu_i \hat{\mathbf{c}}_i$$

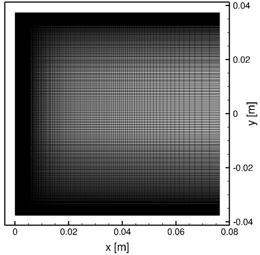
Given $\boldsymbol{\mu} \in \mathbb{R}^d$, we use the parametric ROM to issue parametric predictions beyond the training set

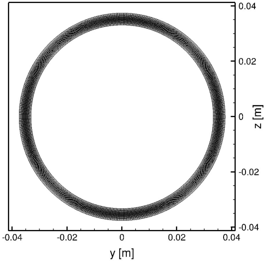
Numerical demonstration

Parametric modeling of an RDRE combustion chamber

- LES simulations of the reactive, viscous 3D Navier-Stokes equations
- skeletal chemistry mechanism based on the Foundational Fuel Chemistry Model (FFCMy-30)
- non-premixed fuel injection (gaseous methane and oxygen)
- injector design with 72 discrete injector pairs
- parametric variations in the flow conditions







Parametric modeling of an RDRE combustion chamber

Parametric description

- parametric variations in flow conditions (mass flow-rate, $\dot{m}$ $[\mathrm{kg} \cdot \mathrm{s}^{-1}]$, and equivalence ratio, Φ)
- these variations are characterized in our model using a scalar parameter $\mu \in \mathbb{R}$

$$\mu = \frac{\dot{m}}{\dot{m}_0} + \frac{\Phi}{\Phi_0},$$

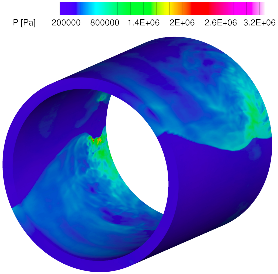
  where $\dot{m}_0$ and Φ_0 are the respective maximum mass flow-rate and equivalence ratio
- we have simulation data for three parameter instances

μ	Flow Condition	$\dot{m}$ $[\mathrm{kg} \cdot \mathrm{s}^{-1}]$	Φ	Time Interval	Periods	Dominant Waves	Secondary Waves
$\mu_1 = 1.41$	nominal	0.267	1.16	$[3.7525, 4.0000]$ ms	1.97	5	0
$\mu_2 = 1.73$	high Φ	0.266	1.71	$[1.8425, 2.0900]$ ms	1.64	6	8
$\mu_3 = 1.55$	high $\dot{m}$	0.333	1.09	$[3.7525, 4.0000]$ ms	1.97	5	0

Parametric modeling of an RDRE combustion chamber

1. Generate training data + data manipulation

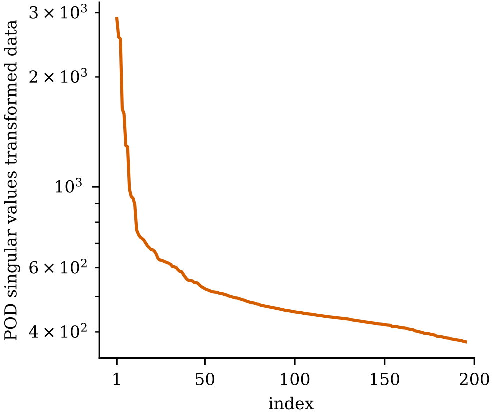
- 2 ms of full-state solutions for all three parameter instances generated ~6M CPU hours on > 16K cores
- the original simulation data has been interpolated on a structured mesh comprising $n_x = 4,204,200$ spatial DoF
- time step size $\Delta t \sim 10^{-9}$ s
- available data: 100 down-sampled snapshots for each parameter instance (about 0.25 ms of physical time)
- 8 transformed state variables: specific volume, pressure, 3D velocity, temperature, 2 species mass fractions (fuel and oxidizer)
- train ROM using data corresponding to μ_1 and μ_2, make ROM predictions for μ_3
- training data: snapshot matrix $\mathbf{Q} \in \mathbb{R}^{34M \times 200}$

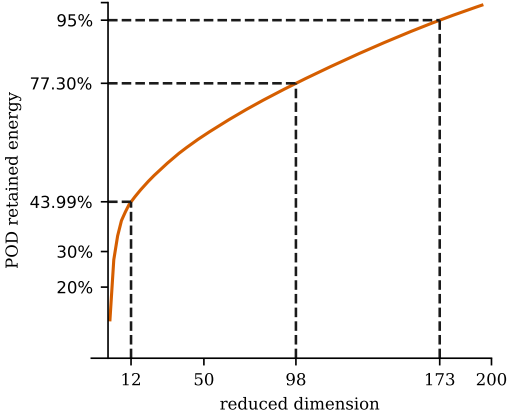


Parametric modeling of an RDRE combustion chamber

2. Compute (global) POD basis

- snapshot matrix of transformed variables
- scale and center snapshot data
- compute POD basis $\mathbf{V}_r \in \mathbb{R}^{34M \times r}$
- low-data regime limits size of the non-intrusive ROM

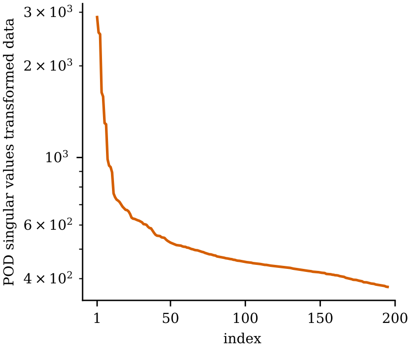


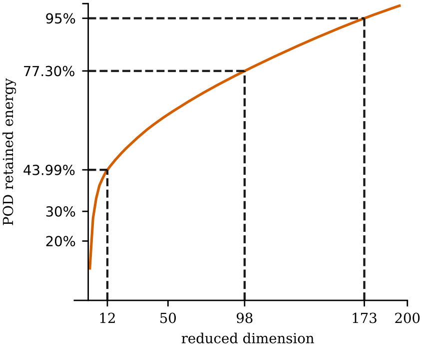


Parametric modeling of an RDRE combustion chamber

3. Inference reduced operators

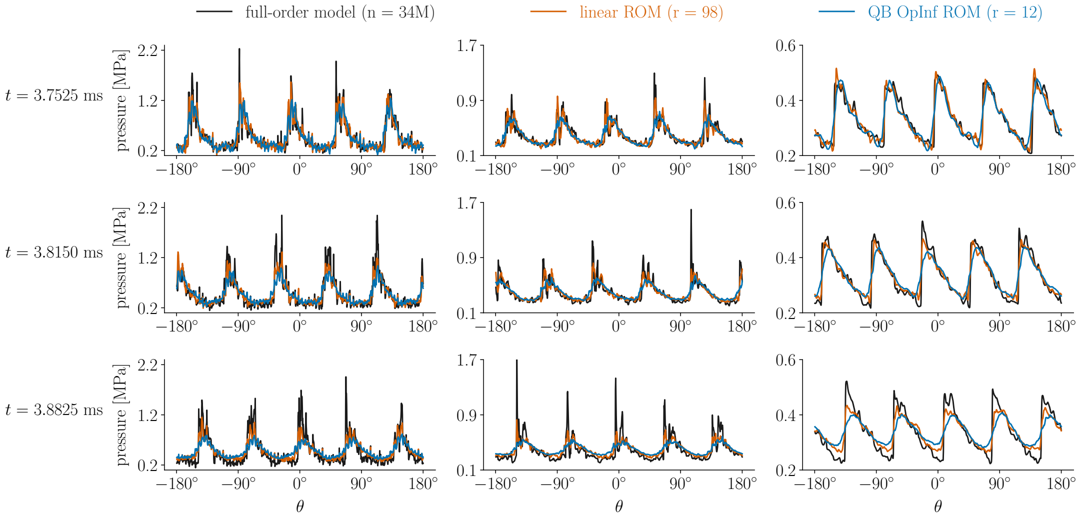
- compute reduced snapshot matrix $\hat{\mathbf{Q}} = \mathbf{V}_r^\top \mathbf{Q} \in \mathbb{R}^{r \times 200}$
- learn a fully discrete parametric ROM
- solve linear least squares to infer reduced operators



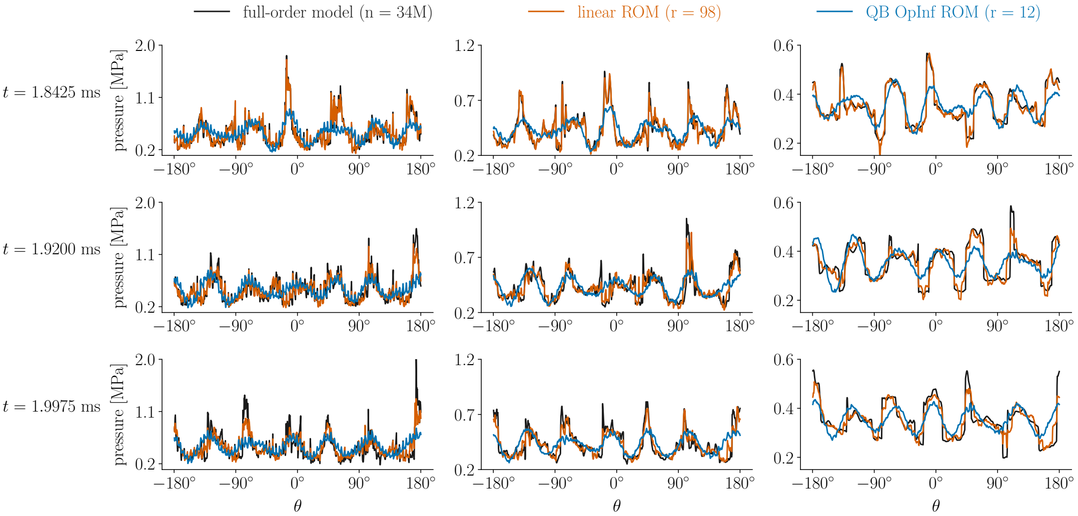


Due to the scarcity of the available data, we also consider linear parametric reduced models

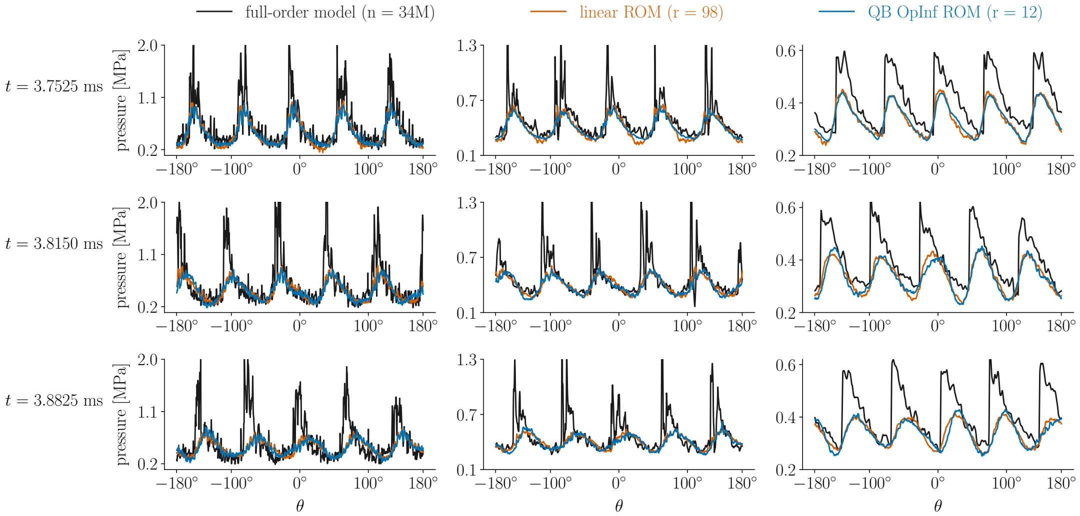
One-dimensional circumferential pressure profiles: 1st training parameter (nominal flow condition)



One-dimensional circumferential pressure profiles: 2nd training parameter (high equivalence ratio)



One-dimensional circumferential pressure profile prediction outside of the training set (3rd parameter, high mass flow-rate case)



Outlook

Domain-decomposed data-driven reduced models

- basis localization can lead to faster singular value decay $\rightarrow$ smaller r_i, although now have ROM of dimension r_i for each subdomain $i = 1, 2, \ldots, k$
- the decomposition also mitigates computational complexity of offline snapshot processing

Preprocess the training data set via filtering

- filter the snapshots in the training data set – variable by variable – prior to training the ROM
- filtering can reduce overfitting, improve the prediction accuracy of the ROM, and improve the condition of the Operator Inference learning problem

Summary - Non-intrusive ROMs

- non-intrusive formulation enables rapid deployment, ROM derivation without access to source code, and variable transformations to promote structure
- it moreover enables ROM development across multiple sites without the need to transfer large-scale data sets (e.g., training data generation, data manipulation, and POD basis computation at ARFL, and OpInf ROM construction and postprocessing at UT Austin)
- Operator Inference ROMs are completely decoupled from CFD code and very fast to simulate
- our ROMs evaluate within milliseconds on a laptop computer
- Operator Inference can be used to construct both ROMs for predictions beyond the training time horizon and parametric ROMs
- slow singular value decay is a challenge for linear-subspace approximations
- generation of training data is a major expense; for realistic combustion problems we generally lack sufficient training data to fully resolve the dynamics
- domain decomposition and filtering are promising approaches to address these challenges

References

- Farcas, I.G., Gundevia, R., Munipalli, R., Willcox, K.E., 2023. *Parametric non-intrusive reduced-order models via operator inference for large-scale rotating detonation engine simulations*. AIAA SciTech Forum 2023, National Harbor MD
- Farcas, I.G., Munipalli, R., Willcox, K.E., 2022. *On filtering in non-intrusive data-driven reduced-order modeling*. AIAA Aviation Forum 2022, Chicago IL
- Qian, E., Farcas, I.G., Willcox, K., 2022. *Reduced Operator Inference for Nonlinear Partial Differential Equations*. SIAM Journal on Scientific Computing 44, A1934–A1959.
- Peherstorfer, B., Willcox, K., 2016. *Data-driven operator inference for nonintrusive projection-based model reduction*. Computer Methods in Applied Mechanics and Engineering 306, 196–215.
- Qian, E., Kramer, B., Peherstorfer, B., Willcox, K., 2020. *Lift & Learn: Physics-informed machine learning for large-scale nonlinear dynamical systems*. Physica D: Nonlinear Phenomena 406, 132401.
- Swischuk, R., Kramer, B., Huang, C., Willcox, K., 2020. *Learning Physics-Based Reduced-Order Models for a Single-Injector Combustion Process*. AIAA Journal 58, 2658–2672.
- McQuarrie, S.A., Huang, C., Willcox, K.E., 2021. *Data-driven reduced-order models via regularised Operator Inference for a single-injector combustion process*. Journal of the Royal Society of New Zealand 51, 194–211.