Linear Systems Theory and Model Reduction

Elnaz Rezaian
Computational Aerosciences Laboratory
University of Michigan

AF/COE Workshop

Summer 2023

UNIVERSITY OF MICHIGAN

Overview

- Introduction to Balanced Truncation
- Review of Linear Systems Theory
  - External Description
  - Internal Description
  - Stability
  - Reachability and Observability
  - System Realization
  - System Gramians
- Linear Model Reduction Techniques
  - Balanced Truncation
  - Balanced POD
  - Eigensystem Realization Algorithm
- Applications
  - One-dimensional Reactive Flow
  - Aeroacoustic Prediction

Projection-based Model Reduction

Let's define a test basis $\mathbf{W}$ and project the equation onto the test subspace,

$$\mathbf{W}^T\mathbf{V}\frac{d\mathbf{q}_r(t)}{dt} = \mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{q}_r(t), t), \quad \mathbf{W}^T\mathbf{V}\mathbf{q}_r(0) = \mathbf{W}^T\mathbf{q}_0$$

Special case: Galerkin projection

If the test subspace is the same as the trial subspace (i.e., $\mathbf{W} = \mathbf{V}$), then the ROM ODE is,

$$\frac{d\mathbf{q}_r(t)}{dt} = \mathbf{V}^T\mathbf{f}(\mathbf{V}\mathbf{q}_r(t), t), \quad \mathbf{q}_r(0) = \mathbf{V}^T\mathbf{q}_0$$

Balanced Truncation

- Being biased towards the energetic structures, imposes limitations on the effectiveness of POD in control applications. As the modes captured by POD are not necessarily controllable.

- On the other hand, highly observable modes that contain low energy are also ignored by POD, which makes POD-based ROMs inaccurate and even unstable in certain applications.

Balanced Truncation (developed by Moore in 1981) accounts for the controllability and observability of the modes in building ROMs.

Balanced Truncation

Consider the LTI system (high-fidelity model),

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t),$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u},$$

We still want to find a hierarchical modal representation for the system, but the measure of "hierarchy" in balanced truncation is different (more useful) from POD.

Balanced Truncation

We still want to find a hierarchical modal representation for the system, but the measure of "hierarchy" in balanced truncation is different (more useful) from POD.

Switch from **Most energetic** as in POD, to **Most controllable and observable**

Most energetic ↓ Least energetic

Most controllable and observable ↓ Least controllable and observable

Balanced Truncation

Gramians

$$\mathbf{A}\mathcal{W}_p + \mathcal{W}_p\mathbf{A}^* + \mathbf{B}\mathbf{B}^* = 0 \qquad \mathcal{W}_p = \int_0^\infty e^{\mathbf{A}t}\mathbf{B}\mathbf{B}^* e^{\mathbf{A}^*t} dt$$

$$\mathbf{A}^*\mathcal{W}_o + \mathcal{W}_o\mathbf{A} + \mathbf{C}^*\mathbf{C} = 0 \qquad \mathcal{W}_o = \int_0^\infty e^{\mathbf{A}^*t}\mathbf{C}^*\mathbf{C} e^{\mathbf{A}t} dt$$

Balancing transformation

$$\mathbf{T}^{-1}\mathcal{W}_c(\mathbf{T}^{-1})^* = \mathbf{T}^*\mathcal{W}_o\mathbf{T} = \mathbf{\Sigma}$$

➡ Diagonal matrix

Balancing modes

$$\mathcal{W}_p = \mathbf{U}\mathbf{U}^* \quad \mathcal{W}_o = \mathbf{L}\mathbf{L}^* \qquad \mathbf{T}_r = \mathbf{U}\mathbf{W}_r\mathbf{\Sigma}_r^{-1/2}$$

$$\mathbf{U}^*\mathbf{L} = \mathbf{W}\mathbf{\Sigma}\mathbf{V}^* \qquad \mathbf{T}_r^{-1} = \mathbf{\Sigma}_r^{-1/2}\mathbf{V}_r^*\mathbf{L}^*$$

Balanced ROM:

$$\dot{\mathbf{x}}_r(t) = \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{B}_r u(t)$$

$$\mathbf{y}(t) = \mathbf{C}_r\mathbf{x}(t),$$

$$\mathbf{A}_r = \mathbf{T}_r^{-1}\mathbf{A}\mathbf{T}_r$$

$$\mathbf{B}_r = \mathbf{T}_r^{-1}\mathbf{B}$$

$$\mathbf{C}_r = \mathbf{C}\mathbf{T}_r$$

Error bounds:

$$\|\mathbf{G} - \mathbf{G}_r\|_\infty > \sigma_{r+1}$$

$$\|\mathbf{G} - \mathbf{G}_r\|_\infty < 2\sum_{i=r+1}^{n}\sigma_i$$

$\boldsymbol{G}$: transfer function of FOM

$\boldsymbol{G_r}$: transfer function of ROM

σ_i: the i^th^ diagonal element of $\boldsymbol{\Sigma}$

Antoulas, "Approximation of large-scale dynamical systems ", *SIAM*, 2005

External Description of a System

Consider a linear discrete-time system. External description of the system is a mapping from the inputs to the outputs,

$$\mathbf{u} \longmapsto \mathbf{y} = \mathcal{S}(\mathbf{u}), \quad \mathbf{y}(i) = \sum_{j \in \mathbb{Z}} \mathbf{h}(i, j)\mathbf{u}(j), \qquad i \in \mathbb{Z}$$

$$\mathbf{y} = \mathcal{S}(\mathbf{u}) = \mathbf{h} * \mathbf{u}$$

$\mathcal{S}$: a linear operator

$\mathbf{h}$: weighting pattern

External Description of a System

Consider a linear discrete-time system. External description of the system is a mapping from the inputs to the outputs,

$$\mathbf{u} \longmapsto \mathbf{y} = \mathcal{S}(\mathbf{u}), \quad \mathbf{y}(i) = \sum_{j \in \mathbb{Z}} \mathbf{h}(i, j)\mathbf{u}(j), \qquad i \in \mathbb{Z}$$

In matrix form,

$$\left(\begin{array}{c} \vdots \\ \mathbf{y}(-2) \\ \mathbf{y}(-1) \\ \hline \mathbf{y}(0) \\ \mathbf{y}(1) \\ \vdots \end{array}\right) = \left(\begin{array}{ccc|ccc} \ddots & \vdots & \vdots & \vdots & \vdots & \iddots \\ \cdots & \mathbf{h}(-2,-2) & \mathbf{h}(-2,-1) & \mathbf{h}(-2,0) & \mathbf{h}(-2,1) & \cdots \\ \cdots & \mathbf{h}(-1,-2) & \mathbf{h}(-1,-1) & \mathbf{h}(-1,0) & \mathbf{h}(-1,1) & \cdots \\ \hline \cdots & \mathbf{h}(0,-2) & \mathbf{h}(0,-1) & \mathbf{h}(0,0) & \mathbf{h}(0,1) & \cdots \\ \cdots & \mathbf{h}(1,-2) & \mathbf{h}(1,-1) & \mathbf{h}(1,0) & \mathbf{h}(1,1) & \cdots \\ \iddots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}\right) \left(\begin{array}{c} \vdots \\ \mathbf{u}(-2) \\ \mathbf{u}(-1) \\ \hline \mathbf{u}(0) \\ \mathbf{u}(1) \\ \vdots \end{array}\right)$$

External Description of a System

Consider a linear discrete-time system. External description of the system is a mapping from the inputs to the outputs,

$$\mathbf{u} \longmapsto \mathbf{y} = \mathcal{S}(\mathbf{u}), \quad \mathbf{y}(i) = \sum_{j \in \mathbb{Z}} \mathbf{h}(i, j)\mathbf{u}(j), \qquad i \in \mathbb{Z}$$

The system is time-invariant if,

$$\mathbf{h}(i, j) = \mathbf{h}_{i-j} \quad \in \mathbb{R}^{p \times m}$$

p: number of outputs
m: number of inputs

The system is causal if,

$$\mathbf{h}(i, j) = \mathbf{0}, \qquad i \le j$$

External Description of a System

The weighting matrix for a linear time-invariant system is defined by the sequence of constant matrices,

$$\mathbf{h} = (\ldots, \mathbf{h}_{-2}, \mathbf{h}_{-1}, \mathbf{h}_0, \mathbf{h}_1, \mathbf{h}_2, \ldots)$$

This sequence is the output of the system when the input is excited by the unit impulse,

$$\mathbf{u}(t) = \delta(t) = \begin{cases} 1, & t = 0, \\ 0, & t \neq 0. \end{cases}$$

External Description of a System

Definition:

- For a time-invariant, causal, and smooth continuous-time system, and
- For a time-invariant, causal discrete-time system,

the sequence of $p \times m$ matrices $\mathbf{h}_i$,

$$\boldsymbol{M} = \begin{bmatrix} \mathbf{h}_0 & \mathbf{h}_1 & \dots & \mathbf{h}_i & \dots \end{bmatrix} \qquad \mathbf{h}_i \in \mathbb{R}^{p \times m}$$

which is called the sequence of Markov parameters, can be computed by the impulse response of the system.

- This sequence identifies the external description of the system.

External Description of a System

Consider the external description of the system,

$$\mathbf{y} = \mathcal{S}(\mathbf{u}) = \mathbf{h} * \mathbf{u}$$

Taking Laplace transform of the external description,

$$\mathbf{Y}(s) = \mathbf{H}(s)\mathbf{U}(s)$$

And the transfer function of the system is,

$$\mathbf{G}(s) = \frac{\mathbf{Y}(s)}{\mathbf{U}(s)}$$

Internal Description of a System

Internal description of a system employs input $\mathbf{u}$, and state $\mathbf{x}$. For a linear continuous-time system we have,

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \qquad t \in \mathbb{R}$$

Which is called the state-space representation of the system.

- If $\mathbf{A}$ and $\mathbf{B}$ are constant, the above system is a linear time-invariant (LTI) system.

Internal Description of a System

The state-space representation of a linear discrete-time system is defined as,

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

k: time index

The output equation is an algebraic equation for both the continuous-time and the discrete-time systems,

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

C: output map

Stability of Linear Systems

- A linear continuous-time system is stable if all of the eigenvalues of $\mathbf{A}$ are located in the left half of the complex plane.

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \qquad t \in \mathbb{R}$$

- A linear discrete-time system is stable if all of the eigenvalues of $\mathbf{A}$ are located inside the unit circle.

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

Internal Description of a System

Consider the continuous-time system,

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \qquad t \in \mathbb{R}$$

The analytical solution of the system is,

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 + \int_{t_0}^{t} e^{\mathbf{A}(\mathbf{t}-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau, \qquad t \geq t_0$$

Internal Description of a System

Consider the output of the system,

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

The analytical solution of the continuous-time system is,

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 + \int_{t_0}^{t} e^{\mathbf{A}(\mathbf{t}-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau, \qquad t \geq t_0$$

With $x_0 = 0$, the impulse response is,

$$\mathbf{y}(t) = \begin{cases} \mathbf{C}e^{\mathbf{A}t}\mathbf{B} + \delta(t)\mathbf{D}, & t \geq 0, \\ \mathbf{0}, & t \leq 0. \end{cases}$$

Internal Description of a System

Consider the discrete-time system,

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

The solution of the discrete-time system is,

$$\mathbf{x}_k = \mathbf{A}^{k-k_0}\mathbf{x}_{k_0} + \sum_{j=k_0}^{k-1} \mathbf{A}^{k-1-j}\mathbf{B}\mathbf{u}(j), \qquad k \geq k_0$$

Therefore, the impulse response of the system is,

$$\mathbf{y}_k = \begin{cases} \mathbf{C}\mathbf{A}^{k-1}\mathbf{B}, & t > 0, \\ \mathbf{D}, & t = 0, \\ \mathbf{0}, & t < 0, \end{cases}$$

Internal Description of a System

Knowing the impulse response of the discrete-time system,

$$\mathbf{y}_k = \begin{cases} \mathbf{C}\mathbf{A}^{k-1}\mathbf{B}, & k > 0, \\ \mathbf{D}, & k = 0, \\ \mathbf{0}, & k < 0, \end{cases}$$

We can form the sequence of Markov parameters,

$$\boldsymbol{M} = \begin{bmatrix} \mathbf{D} & \mathbf{C}\mathbf{B} & \mathbf{C}\mathbf{A}^2\mathbf{B} & \dots & \mathbf{C}\mathbf{A}^{k-1}\mathbf{B} & \dots \end{bmatrix}$$

Proposition:

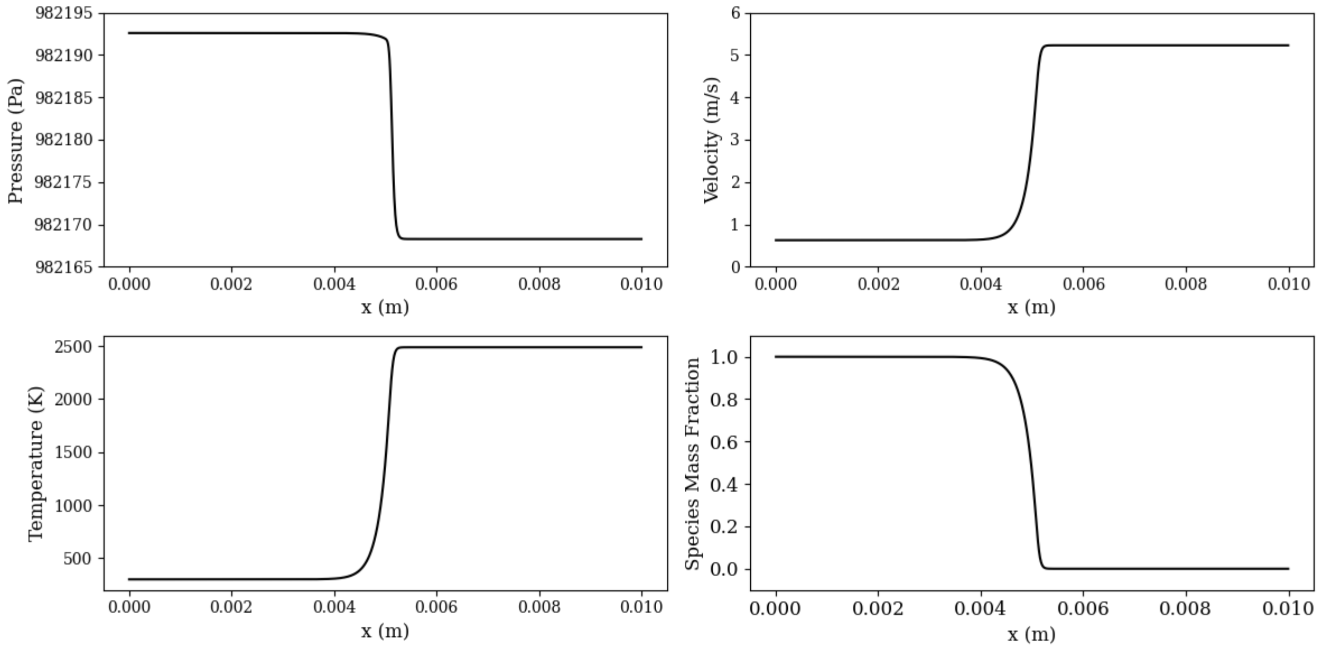
The system transfer function and Markov parameters are invariant under coordinate transformation.

Example 1: 1-D Combustion problem

The one-dimensional Navier-Stokes equations with species transport and reaction are linearized and solved with a finite-volume approach.

The steady-state solution is a stationary flame in a two-species reaction.

Physical time step	1×10^{-8}
Spatial DoF	1000
CFL	0.1
Upstream pressure	984.284 kPa
Upstream temperature	300.16 K
Upstream species mass fraction	[1.0, 0.0]
Back pressure	976.139 kPa



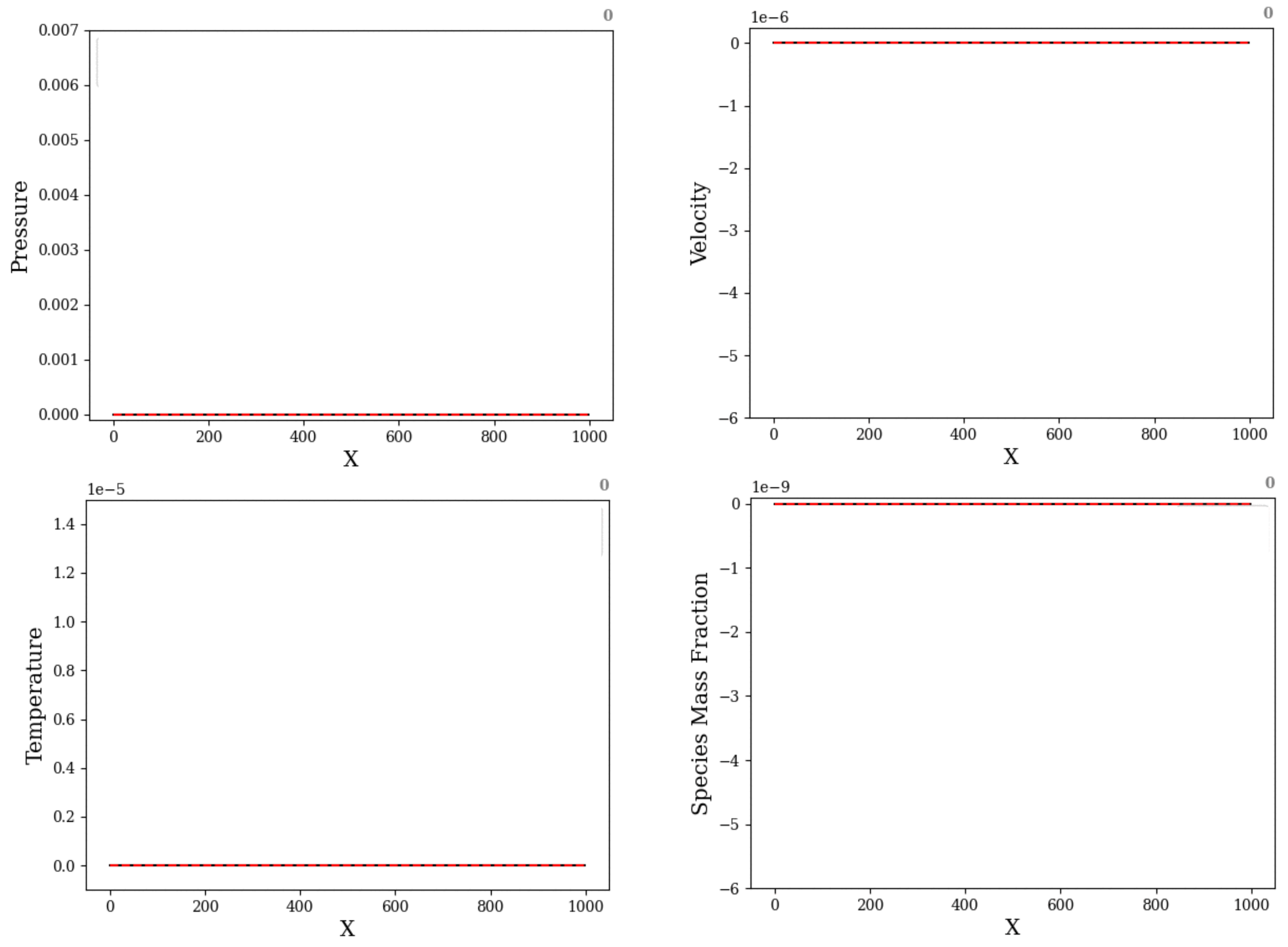
Example 1: 1-D Combustion problem

In order to compute the sequence of Markov parameters in this particular case, we perturb the back pressure with unit impulse and collect the snapshots, which represent the Markov parameters.

$$p_{back} = \delta(t) = \begin{cases} 1 \quad Pa \quad & t = 0 \\ 0 \quad Pa \quad & t > 0 \end{cases}$$

Each snapshot is one column of M:

$$M = [\mathbf{D} \quad \mathbf{CB} \quad \mathbf{CAB} \quad \ldots \quad \mathbf{CA}^{k-1}\mathbf{B} \quad \ldots]$$



Reachability

Consider the reachability matrix, $\mathscr{P} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \dots & \mathbf{A}^{k-1}\mathbf{B} \end{bmatrix}$

Definition:

A system is reachable if all of its states can be excited by the control action.

Theorem:

An LTI system (both continuous-time and discrete-time) is reachable, if and only if $\mathcal{R}(\mathcal{P}) = n$.

⬇

- In an LTI system, reachability reduces to an algebraic definition that depends only on the system matrices rather than time or input function.

Reachability

$$\mathscr{P} = [\mathbf{B} \quad \mathbf{AB} \quad \mathbf{A}^2\mathbf{B} \quad \dots \quad \mathbf{A}^{k-1}\mathbf{B}]$$

Notes:

- Reachability is basis independent. For a nonsingular transformation matrix $\mathbf{T}$,

$$\mathcal{R}(\mathbf{T}\mathscr{P}\mathbf{T}^*) = \mathbf{T}\mathcal{R}(\mathscr{P})$$

The Duality Principle in Linear Systems

The dual system of a system with matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ is defined as,

$$\mathbf{\Sigma}^* = \left(\begin{array}{c|c} -\mathbf{A}^* & -\mathbf{C}^* \\ \hline \mathbf{B}^* & \mathbf{D}^* \end{array} \right) \in \mathbb{R}^{(n+m)\times(n+p)}$$

(*) denotes complex conjugate transpose.

$-\mathbf{C}^*$: input map

$\mathbf{B}^*$: output map

$-\mathbf{A}^*$: dynamics matrix

Primal system:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

Observability

Observability determines whether we are able to identify the state from the output of the system.

Let's define the observability matrix as,

$$\mathscr{O} = \begin{bmatrix} \mathbf{C}^* & \mathbf{A}^*\mathbf{C}^* & (\mathbf{A}^*)^2\mathbf{C}^* & \dots & (\mathbf{A}^*)^{n-1}\mathbf{C}^* \end{bmatrix}^*$$

An LTI system (both continuous-time and discrete-time) is observable, if and only if $\mathcal{R}(\mathcal{O}) = n$.

Certain state variables may be inaccessible ($y(t) = 0$ for all $t \geq 0$) and therefore unobservable.

Observability

Similar to the reachability, observability is also invariant under coordinate transformation.

Theorem:

Observability and reachability are dual concepts.

- A system is reachable, if and only if its dual (adjoint) system is observable.

Overview

- Introduction to Balanced Truncation
- Review of Linear Systems Theory
  - External Description
  - Internal Description
  - Stability
  - Reachability and Observability
  - System Realization
  - System Gramians
- Linear Model Reduction Techniques
  - Balanced Truncation
  - Balanced POD
  - Eigensystem Realization Algorithm
- Applications
  - One-dimensional Reactive Flow
  - Aeroacoustic Prediction

Highlights:

- System's external description is identified by the input-output behavior.
- For a linear, time-invariant, causal system, this behavior is given by the response of the system to unit impulse.
- System's internal description is given by the state-space representation.
- The concepts of reachability and observability, determine whether all of the states of a system are reachable or observable.
- Reachability, observability, and Markov parameters are transformation-invariant.

System Realization

Given the external description of the system,

$$\mathbf{u} \longmapsto \mathbf{y} = \mathcal{S}(\mathbf{u}), \quad \mathbf{y}(i) = \sum_{j \in \mathbb{Z}} \mathbf{h}(i, j)\mathbf{u}(j), \qquad i \in \mathbb{Z}$$

the goal of system realization is to obtain the internal description (i.e., the triplet $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$),

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

$$\mathbf{y} = \mathbf{C}\mathbf{x}$$

System Realization

The solution of the system realization problem is not unique:

- For any system, there are infinitely many realizations that generate the same output for a particular input.

The realization with the smallest state-space dimension is called the minimum realization.

System Realization

System realization relies on the Hankel matrix that is constructed based on the sequence of Markov parameters (impulse response for an LTI system):

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \cdots & \mathbf{h}_{m_p} \\ \mathbf{h}_2 & \mathbf{h}_3 & \cdots & \mathbf{h}_{m_p+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}_{m_o} & \mathbf{h}_{m_o+1} & \cdots & \mathbf{h}_{m_o+m_p+1}. \end{bmatrix}$$

The Eigensystem Realization Algorithm (ERA) uses this matrix to build the minimum realization (we will learn this method by the end of the semester).

The Reachability Gramian

For a stable continuous-time system, we can define the reachability Gramian as,

$$\mathscr{W}_p = \int_0^\infty e^{\mathbf{A}\tau}\mathbf{B}\mathbf{B}^* e^{\mathbf{A}^*\tau} d\tau$$

This Gramian is the solution to the following Lyapunov equation,

$$\mathbf{A}\mathscr{W}_p + \mathscr{W}_p\mathbf{A}^* + \mathbf{B}\mathbf{B}^* = \mathbf{0}$$

The Reachability Gramian

In discrete-time systems the reachability Gramian is defined as,

$$\mathscr{W}_p(t) = \mathscr{C}\mathscr{C}^* = \sum_{k=0}^{t-1} \mathbf{A}^k \mathbf{B}\mathbf{B}^* (\mathbf{A}^*)^k, \qquad t \in \mathbb{Z}_+$$

$\mathcal{C}$: the reachability matrix

The largest eigenvalues of $\mathcal{W}_p$ correspond to the most reachable states.

The infinite reachability Gramian of a stable discrete-time system is the solution of the following Lyapunov equation:

$$\mathbf{A}\mathscr{W}_p\mathbf{A}^* + \mathbf{B}\mathbf{B}^* = \mathscr{W}_p$$

The Observability Gramian

For a stable continuous-time system, we can define the observability Gramian as,

$$\mathscr{W}_o = \int_0^\infty e^{\mathbf{A}^*\tau}\mathbf{C}^*\mathbf{C}e^{\mathbf{A}\tau}d\tau$$

This Gramian is the solution to the following Lyapunov equation,

$$\mathbf{A}^*\mathscr{W}_o + \mathscr{W}_o\mathbf{A} + \mathbf{C}^*\mathbf{C} = \mathbf{0}$$

The Observability Gramian

In discrete-time systems the observability Gramian is defined as,

$$\mathscr{W}_o(t) = \mathscr{O}^* \mathscr{O} \qquad t \in \mathbb{Z}_+$$

$\mathcal{O}$: the observability matrix

The largest eigenvalues of $\mathcal{W}_o$ correspond to the most observable states.

The infinite observability Gramian of a stable discrete-time system is the solution of the following Lyapunov equation:

$$\mathbf{A}^* \mathscr{W}_o \mathbf{A} + \mathbf{C}^* \mathbf{C} = \mathscr{W}_o$$

Implementation

Use the control systems library in Python or use MATLAB,

```
sys = ss(A, B, C, D);
C = ctrb(sys.A,sys.B);
O = obsv(sys.A,sys.C);
Wc = gram(sys, 'c');
Wo = gram(sys, 'o');
```

➡ Controllability matrix

Implementation

Use the control systems library in Python or use MATLAB,

```
sys = ss(A, B, C, D);
C = ctrb(sys.A,sys.B);
O = obsv(sys.A,sys.C);
Wc = gram(sys, 'c');
Wo = gram(sys, 'o');
```

➡ Observability matrix

Implementation

Use the control systems library in Python or use MATLAB,

```
sys = ss(A, B, C, D);
C = ctrb(sys.A,sys.B);
O = obsv(sys.A,sys.C);
Wc = gram(sys, 'c');
Wo = gram(sys, 'o');
```

➡ Controllability Gramian

Implementation

Use the control systems library in Python or use MATLAB,

```
sys = ss(A, B, C, D);
C = ctrb(sys.A,sys.B);
O = obsv(sys.A,sys.C);
Wc = gram(sys, 'c');
Wo = gram(sys, 'o');
```

→ Observability Gramian

Overview

- Introduction to Balanced Truncation
- Review of Linear Systems Theory
  - External Description
  - Internal Description
  - Stability
  - Reachability and Observability
  - System Realization
  - System Gramians
- Linear Model Reduction Techniques
  - Balanced Truncation
  - Balanced POD
  - Eigensystem Realization Algorithm
- Applications
  - One-dimensional Reactive Flow
  - Aeroacoustic Prediction

Highlights:

- System realization consists of determining the internal description from the external description.
- In non-intrusive model reduction we are looking for the lowest-dimensional system that performs this task.
- The system Gramians determine the degree of reachability and observability.

Balanced Truncation

Consider the LTI system (high-fidelity model),

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t),$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u},$$

The goal is to find a coordinate transformation $\mathbf{T}$ such that the transformed Gramians are equal and diagonal,

$$\mathbf{T}\mathscr{W}_p\mathbf{T}^* = \mathbf{T}^{-*}\mathscr{W}_o\mathbf{T}^{-1} = \mathbf{\Sigma}$$

Balanced Truncation

The goal is to find a coordinate transformation $\mathbf{T}$ such that the transformed Gramians are equal and diagonal,

$$\mathbf{T}\mathscr{W}_p\mathbf{T}^* = \mathbf{T}^{-*}\mathscr{W}_o\mathbf{T}^{-1} = \mathbf{\Sigma}$$

Why balancing?

Because we want to give equal weight to the controllability and observability.

As a result, states that are difficult to reach are also the states that are difficult to observe (we can truncate them!).

Balanced Truncation

Gramians

$$\mathbf{A}\mathcal{W}_p + \mathcal{W}_p\mathbf{A}^* + \mathbf{B}\mathbf{B}^* = 0 \qquad \mathcal{W}_p = \int_0^\infty e^{\mathbf{A}t}\mathbf{B}\mathbf{B}^* e^{\mathbf{A}^*t}dt$$

$$\mathbf{A}^*\mathcal{W}_o + \mathcal{W}_o\mathbf{A} + \mathbf{C}^*\mathbf{C} = 0 \qquad \mathcal{W}_o = \int_0^\infty e^{\mathbf{A}^*t}\mathbf{C}^*\mathbf{C} e^{\mathbf{A}t}dt$$

Balancing transformation

$$\mathbf{T}^{-1}\mathcal{W}_c(\mathbf{T}^{-1})^* = \mathbf{T}^*\mathcal{W}_o\mathbf{T} = \boldsymbol{\Sigma}$$

➡ Diagonal matrix

Balancing modes

$$\mathcal{W}_p = \mathbf{U}\mathbf{U}^* \quad \mathcal{W}_o = \mathbf{L}\mathbf{L}^* \qquad \mathbf{T}_r = \mathbf{U}\mathbf{W}_r\boldsymbol{\Sigma}_r^{-1/2}$$

$$\mathbf{U}^*\mathbf{L} = \mathbf{W}\boldsymbol{\Sigma}\mathbf{V}^* \qquad \mathbf{T}_r^{-1} = \boldsymbol{\Sigma}_r^{-1/2}\mathbf{V}_r^*\mathbf{L}^*$$

Balanced ROM:

$$\dot{\mathbf{x}}_r(t) = \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{B}_r u(t)$$

$$\mathbf{y}(t) = \mathbf{C}_r\mathbf{x}(t),$$

$$\mathbf{A}_r = \mathbf{T}_r^{-1}\mathbf{A}\mathbf{T}_r$$

$$\mathbf{B}_r = \mathbf{T}_r^{-1}\mathbf{B}$$

$$\mathbf{C}_r = \mathbf{C}\mathbf{T}_r$$

Error bounds:

$$\|\mathbf{G} - \mathbf{G}_r\|_\infty > \sigma_{r+1}$$

$$\|\mathbf{G} - \mathbf{G}_r\|_\infty < 2\sum_{i=r+1}^{n}\sigma_i$$

G: transfer function of FOM

G_r: transfer function of ROM

σ_i: the i[th] diagonal element of $\boldsymbol{\Sigma}$

Antoulas, "Approximation of large-scale dynamical systems ", *SIAM*, 2005

Empirical Gramians

Balanced truncation generates ROMs with stability guarantees, however, computing the analytical Gramians becomes restrictive when,

- We don't have access to the internal description.
- The system is unstable*.
- Dimension of the system is higher than a few thousands**.

* It is possible to decompose the system into stable and unstable sub-systems to bypass this issue.

** We can still use analytical BT for systems with up to a million DoFs using low-rank Lyapunov solvers.

Empirical Gramians

Balanced truncation generates ROMs with stability guarantees, however, computing the analytical Gramians becomes restrictive when,

- We don't have access to the internal description.
- The system is unstable*.
- Dimension of the system is higher than a few thousands**.

One way to address these limitations is to use empirical Gramians.

* It is possible to decompose the system into stable and unstable sub-systems to bypass this issue.

** We can still use analytical BT for systems with up to a million DoFs using low-rank Lyapunov solvers.

Empirical Gramians

Consider the discrete-time direct system,

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

The direct system impulse response snapshots matrix is,

$$\mathscr{P} = \begin{bmatrix} \mathbf{B} & \mathbf{A}\mathbf{B} & \dots & \mathbf{A}^{m_p-1}\mathbf{B} \end{bmatrix}$$

Empirical Gramians

Consider the discrete-time direct system,

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

The direct system impulse response snapshots matrix is,

$$\mathscr{P} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{m_p-1}\mathbf{B} \end{bmatrix}$$

Therefore, the empirical reachability Gramian can be computed as,

$$\mathscr{W}_p = \mathscr{P}\mathscr{P}^*$$

Empirical Gramians

Consider the discrete-time adjoint system,

$$\mathbf{x}_{k+1} = \mathbf{A}^* \mathbf{x}_k + \mathbf{C}^* \mathbf{y}_k$$

The adjoint system impulse response snapshots matrix is,

$$\mathscr{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{m_o - 1} \end{bmatrix}$$

Empirical Gramians

Consider the discrete-time adjoint system,

$$\mathbf{x}_{k+1} = \mathbf{A}^*\mathbf{x}_k + \mathbf{C}^*\mathbf{y}_k$$

The adjoint system impulse response snapshots matrix is,

$$\mathscr{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{m_o-1} \end{bmatrix}$$

And the empirical observability Gramian is given by, $\mathscr{W}_o = \mathscr{O}^*\mathscr{O}$

Empirical Gramians

Empirical Gramians approach the analytical Gramians (i.e., they are accurate) only if we collect impulse response snapshots with a high enough frequency and for a long enough time.

Empirical Gramians

Empirical Gramians approach the analytical Gramians (i.e., they are accurate) only if we collect impulse response snapshots with a high enough frequency and for a long enough time.

Sampling frequency and total sampling time are adhoc parameters.

- Collect until the snapshots matrices are full-rank?
- Collect until the Impulse response dies out?

The rule of thumb is to collect until the dynamics of all active eigenvectors are captured.

Empirical Gramians

Empirical Gramians approach the analytical Gramians (i.e., they are accurate) only if we collect impulse response snapshots with a high enough frequency and for a long enough time.

The rule of thumb is to collect until the dynamics of all active eigenvectors are captured.

You can use the "emgr" package to compute the empirical Gramians (time step and total time are still decided by you.).

Balanced POD

Instead of the empirical Gramians, use the product $\mathcal{OP}$ to compute the Hankel matrix,

$$\mathbf{H} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{m_o-1} \end{bmatrix} \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{m_p-1}\mathbf{B} \end{bmatrix} = \begin{bmatrix} \mathbf{CB} & \mathbf{CAB} & \dots & \mathbf{CA}^{m_p-1}\mathbf{B} \\ \mathbf{CAB} & \mathbf{CA}^2\mathbf{B} & \dots & \mathbf{CA}^{m_p}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{m_o-1}\mathbf{B} & \mathbf{CA}_{m_o}\mathbf{B} & \dots & \mathbf{CA}^{m_p+m_o-2}\mathbf{B} \end{bmatrix}$$

Where $\mathcal{O}$ and $\mathcal{P}$ matrices are built by the impulse response of the adjoint and direct systems, respectively.

Balanced POD

1. Instead of the empirical Gramians, use the product $\mathcal{OP}$ to compute the Hankel matrix.
2. Compute SVD of the Hankel matrix: $\mathbf{H} = \mathbf{U\Sigma V}^* = [\mathbf{U}_r \quad \mathbf{U}_t] \begin{bmatrix} \mathbf{\Sigma}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_t \end{bmatrix} [\mathbf{V}_r^* \quad \mathbf{V}_t^*]$
3. Truncate singular vectors corresponding to smaller singular values.
4. Compute the direct and adjoint modes:

$$\mathbf{\Psi} = \mathscr{P}\mathbf{V}_r\mathbf{\Sigma}_r^{-1/2} \qquad \mathbf{\Phi} = \mathscr{O}^*\mathbf{U}_r\mathbf{\Sigma}_r^{-1/2}$$

5. Build the balanced ROM matrices:

$$\mathbf{A}_r = \mathbf{\Phi}^*\mathbf{A\Psi} \qquad \mathbf{B}_r = \mathbf{\Phi}^*\mathbf{B} \qquad \mathbf{C}_r = \mathbf{C\Psi}$$

Willcox and Peraire, "Balanced Model Reduction via the Proper Orthogonal Decomposition ", AIAAJ, 2002

Rowley, "Model Reduction for Fluids, using Balanced Proper Orthogonal Decomposition ", IJBC, 2005

Balanced POD

- Unlike POD, BPOD accounts for observability of the modes.
- Unlike balanced truncation, BPOD is applicable to large-scale systems.
- Unlike balanced truncation, BPOD ROMs are NOT guaranteed to be stable.

BPOD Example: Supersonic Engine Inlet

- 2D Euler equations are solved with $N = 11{,}370$
- $p = 1$: number of inputs (density disturbance of the inlet flow)
- $q = 1$: number of outputs (average Mach number at the diffuser throat)

Amsallem and Farhat, "Stabilization of projection-based reduced-order models", IJNME, 2012

BPOD Example: Supersonic Engine Inlet

ROM size k	10	11	12	13	14	15	16	17	18	19	20
ROM stability status	s	s	s	s	u	s	s	s	u	s	s
ROM size k	21	22	23	24	25	26	27	28	29	30	
ROM stability status	s	s	s	s	s	s	u	u	s	u	

Amsallem and Farhat, "Stabilization of projection-based reduced-order models", IJNME, 2012

Eigensystem Realization Algorithm (ERA)

ERA builds balanced ROMs using only the direct system impulse response.

- More suitable for multi-output systems
- Applicable to experiments
- Does not require access to the FOM operators
- Can be applied to highly stiff systems (unlike the analytical BT)
- Stability relies on the sampling properties.

The Eigensystem Realization Algorithm

ERA is the data-driven non-intrusive extension of BT for discrete-time systems (Ma et al., 2011).

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}u_k$$
$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k$$

$$\mathscr{P} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \quad \mathscr{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix}$$

$$\mathcal{W}_p = \mathscr{P}\mathscr{P}^*$$
$$\mathcal{W}_o = \mathscr{O}^*\mathscr{O}$$
$$\mathbf{H} = \mathscr{O}^*\mathscr{P} = \mathbf{U}\Sigma\mathbf{V}^*$$

Hankel matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{y}_1 & \mathbf{y}_2 & \cdots & \mathbf{y}_{m_p} \\ \mathbf{y}_2 & \mathbf{y}_3 & \cdots & \mathbf{y}_{m_p+1} \\ \vdots & \dots & \ddots & \vdots \\ \mathbf{y}_{m_o} & \mathbf{y}_{m_o+1} & \cdots & \mathbf{y}_{m_o+m_p-1} \end{bmatrix}$$

$$\mathbf{y}_i = \mathbf{CA}^i\mathbf{B}, \qquad i = 0, \dots, n-1$$

Shifted Hankel matrix:

$$\mathbf{H}' = \begin{bmatrix} \mathbf{y}_2 & \mathbf{y}_3 & \cdots & \mathbf{y}_{m_p+1} \\ \mathbf{y}_3 & \mathbf{y}_4 & \cdots & \mathbf{y}_{m_p+2} \\ \vdots & \dots & \ddots & \vdots \\ \mathbf{y}_{m_o+1} & \mathbf{y}_{m_o+2} & \cdots & \mathbf{y}_{m_o+m_p} \end{bmatrix}$$

Balanced ROM matrices:

$$\mathbf{A}_r = \Sigma_r^{-1/2}\mathbf{U}_r^*\mathbf{H}'\mathbf{V}_r\Sigma_r^{-1/2} \qquad \mathbf{B}_r = \Sigma_r^{1/2}\mathbf{V}_r^*\begin{bmatrix} \mathbf{I}_p & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad \mathbf{C}_r = \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{U}_r\Sigma_r^{1/2}$$

Ma et al., "Reduced-order models for control of fluids using the eigensystem realization algorithm ", *Theoret. Comput. Fluid Dyn.*, 2011

Overview

- Introduction to Balanced Truncation
- Review of Linear Systems Theory
  - External Description
  - Internal Description
  - Stability
  - Reachability and Observability
  - System Realization
  - System Gramians
- Linear Model Reduction Techniques
  - Balanced Truncation
  - Balanced POD
  - Eigensystem Realization Algorithm
- Applications
  - One-dimensional Reactive Flow
  - Aeroacoustic Prediction

Highlights:

- The analytical balanced truncation is an intrusive model reduction method that provides theoretical error bounds.
- The computation cost of analytical Gramians encourages us to use empirical Gramians.
- BPOD bypasses computation of the Gramians, but BPOD ROMs do not necessarily satisfy theoretical error bounds.
- ERA is a non-intrusive version of balanced truncation that bypasses the computation of Gramians and adjoint system simulations.

Nonlinear FOM

One-dimensional Navier-Stokes equations with species transport and reaction,

$$\frac{\partial \mathbf{q}_c}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} - \frac{\partial \mathbf{f}_v}{\partial x} = \mathbf{s},$$

$$\mathbf{q}_c = \begin{bmatrix} \rho \\ \rho u \\ \rho h^0 - p \\ \rho Y_l \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho h^0 u \\ \rho Y_l \end{bmatrix}, \quad \mathbf{f}_v = \begin{bmatrix} 0 \\ \tau \\ u\tau - q \\ -\rho V_l Y_l \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dot{\omega}_l \end{bmatrix}$$

Y_l: mass fraction of the l^{th} species
h^0: stagnation enthalpy
τ: shear stress
q: heat flux
V_l: diffusion velocity of the l^{th} species
$\dot{\omega}_l$: production rate of the l^{th} species

Nonlinear FOM is solved with the finite volume approach using the second-order Roe scheme and dual time-stepping.

$$\mathbf{\Gamma}\frac{\partial \mathbf{q}}{\partial \tau} + \frac{\partial \mathbf{q}_c}{\partial t} + \nabla \cdot (\mathbf{f} - \mathbf{f}_v) = \mathbf{s} \qquad \boldsymbol{\Gamma} = \frac{\partial \mathbf{q}_c}{\partial \mathbf{q}}$$

$\boldsymbol{q}_c$: conservative variables
$\boldsymbol{q}$: primitive variables
$\boldsymbol{f}$: inviscid fluxes
$\boldsymbol{f}_v$: viscous fluxes
$\boldsymbol{s}$: source term
τ: pseudo time

PERFORM

The nonlinear FOM is solved with PERFORM (Prototyping Environment for Reacting Flow Order Reduction Methods).

Code is available at https://github.com/cwentland0/perform

PERFORM is a framework for rapid testing of model reduction methods with readily available one-dimensional benchmark problems including:

- One-dimensional standing and transient flames
- Contact surfaces
- Sod shock tube

Documentation and more information:

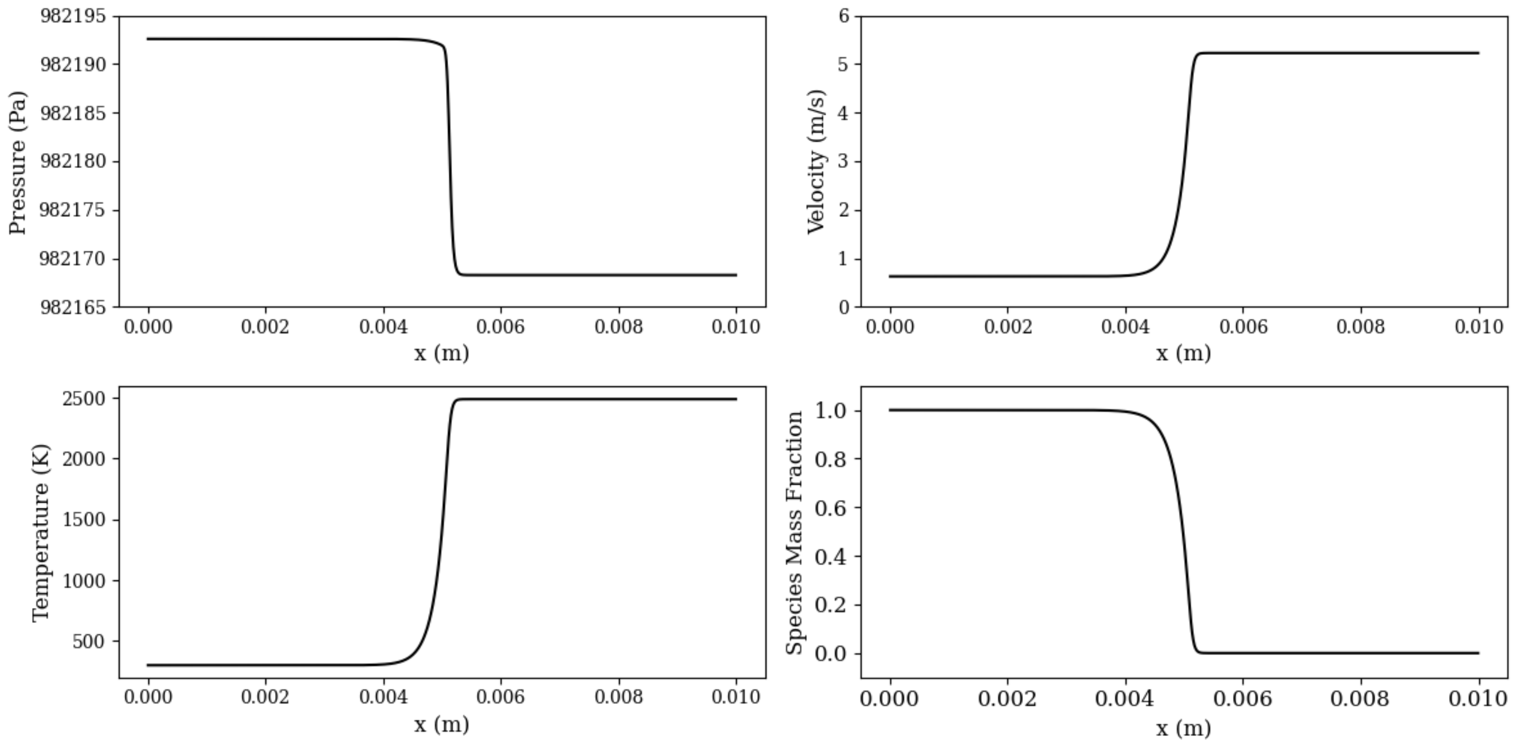
- https://perform.readthedocs.io/en/latest/
- https://romworkshop.engin.umich.edu/test-cases

One-dimensional example demonstration: today at 4:30 p.m.

Steady-state Solution

Nonlinear FOM is solved with non-reflective characteristic boundary conditions at the inlet and outlet. Steady-state solution represents a stationary flame in a two-species reaction.

Physical time step	1×10^{-8}
Spatial DoF	1000
CFL	0.1
Upstream pressure	984.284 kPa
Upstream temperature	300.16 K
Upstream species mass fraction	[1.0, 0.0]
Back pressure	976.139 kPa



Linearized FOM

The nonlinear FOM is linearized about the steady-state solution to create linear ROMs.

Linearized FOM

$$\frac{d\mathbf{q}(t)}{dt} = \mathbf{J}\mathbf{q}(t) + \mathbf{B}u(t), \qquad \mathbf{q}(0) = \mathbf{q}_0$$

$$\mathbf{J} = \left[\boldsymbol{\Gamma}^{-1}(\mathbf{D} - \mathbf{L}_R + \mathbf{L}_L)\right]_{\bar{\mathbf{q}}}$$

$\kappa(J) = 4.82\times10^{13}$

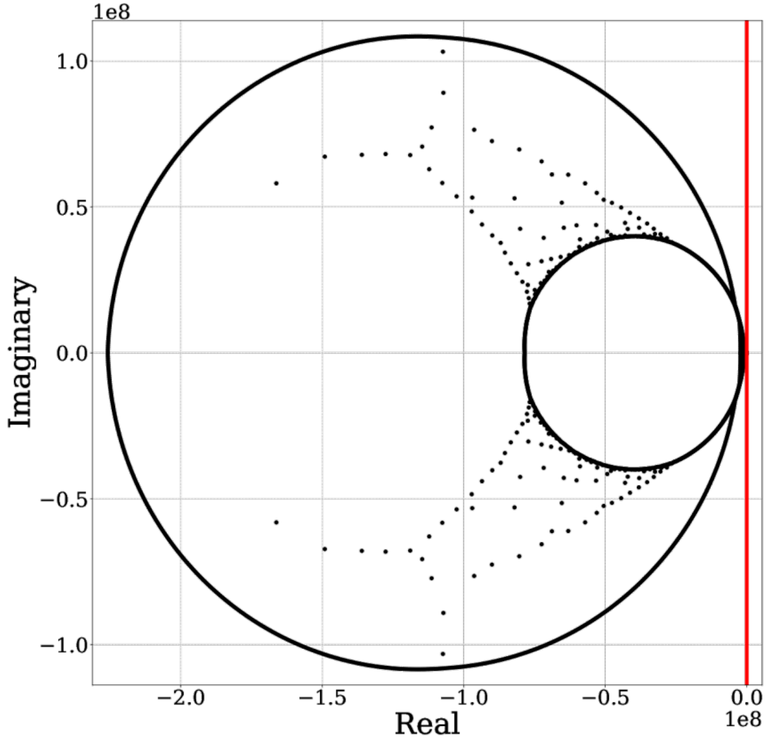
Integrated with the 3rd order Runge-Kutta scheme ($\Delta t = 1\times10^{-9}$).

$$\boldsymbol{\Gamma} = \frac{\partial \mathbf{q}_c}{\partial \mathbf{q}}$$

$$\mathbf{D} = \frac{\partial \mathbf{s}}{\partial \mathbf{q}}$$

$$\mathbf{L} = \frac{\partial \mathbf{f}}{\partial \mathbf{q}} + \frac{\partial \mathbf{f_v}}{\partial \mathbf{q}}$$

$$\mathbf{q} = \begin{bmatrix} p \\ u \\ T \\ Y_k \end{bmatrix}$$



The linearized FOM is stable with eigenvalues located in the left-half of the complex plane.

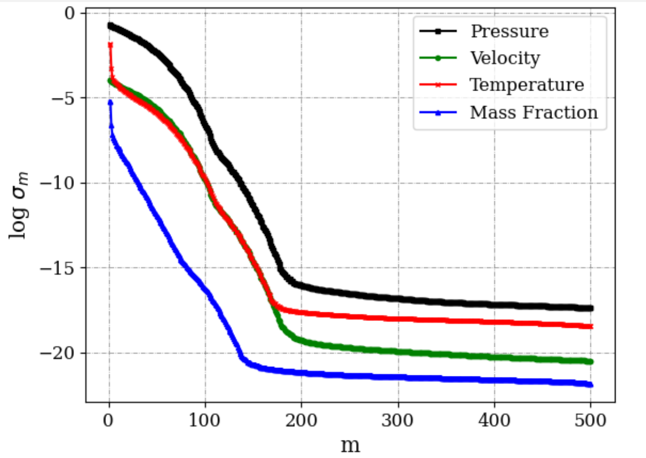
$$\max\big(Re(\lambda)\big) = -180.444$$

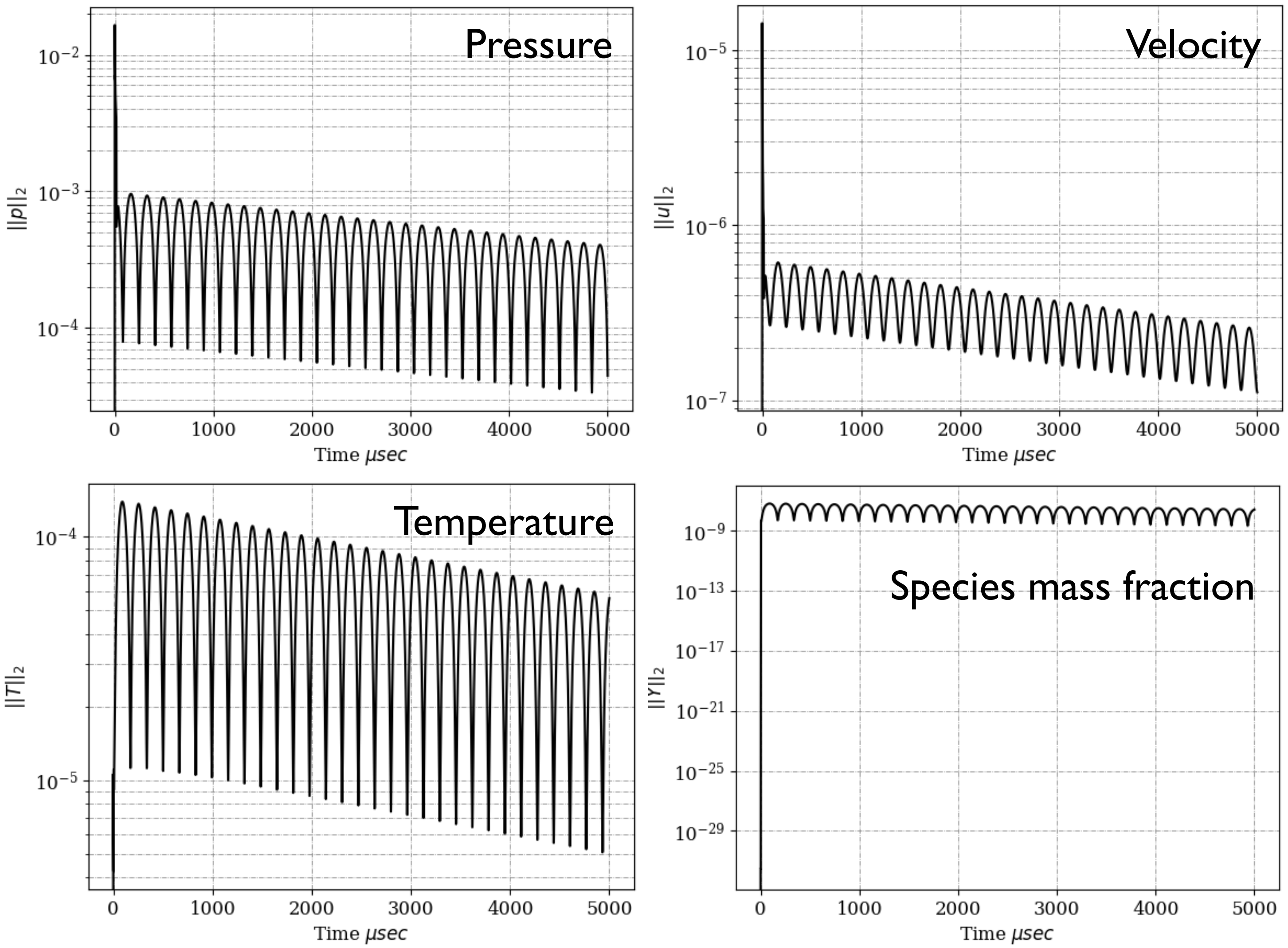
Long-term Behavior of the System

Norm of the state based on unit impulse response demonstrates lightly damped oscillations that need to be captured in the training samples for construction of Hankel matrix.

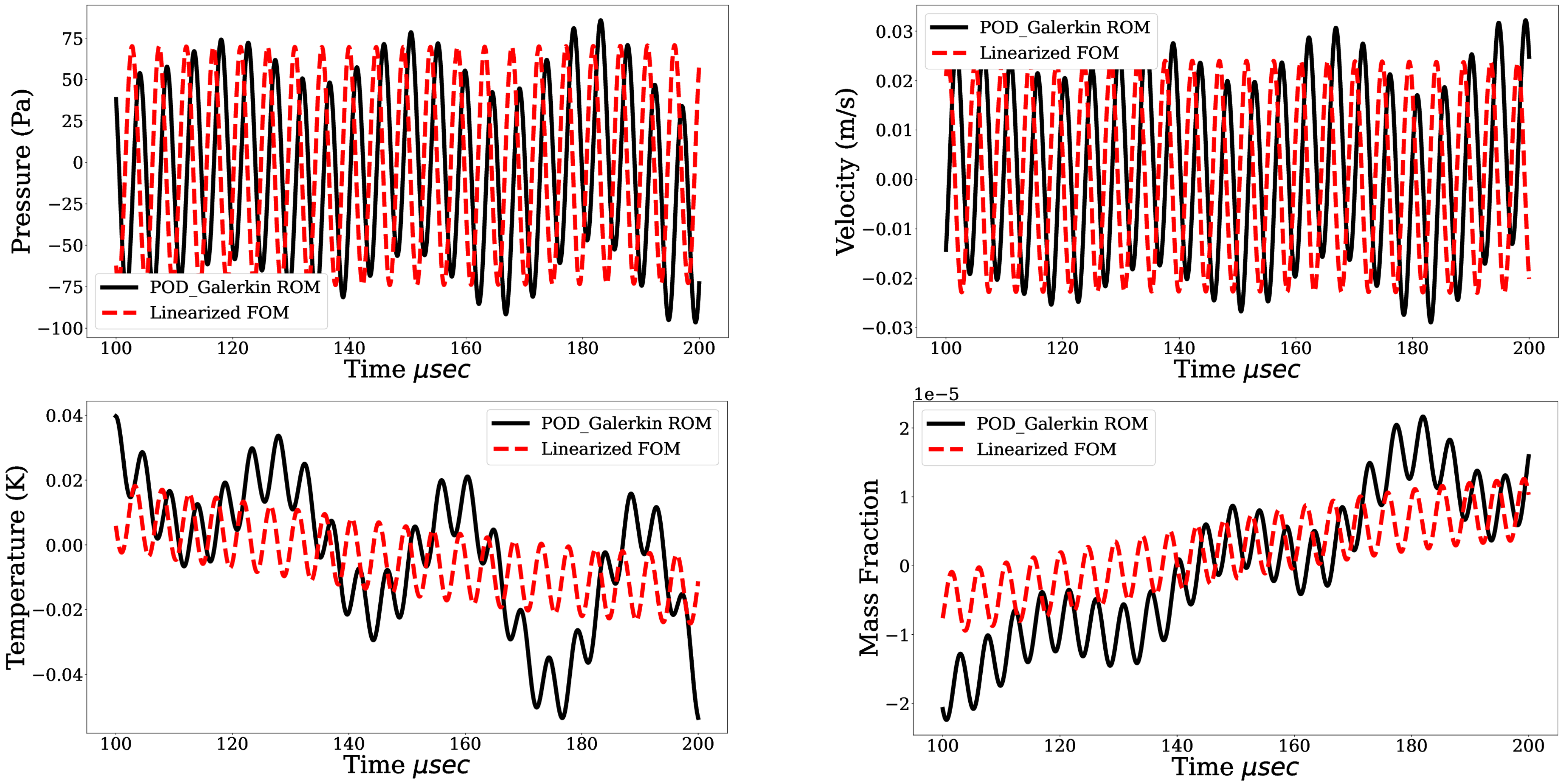
$$p_{back} = \delta(t) = \begin{cases} 1 \quad Pa & t = 0 \\ 0 \quad Pa & t > 0 \end{cases}$$

77 pressure and velocity modes, 50 temperature and 19 mass fraction modes capture 99.99% of the input-output energy.





POD-Galerkin ROM Prediction for $f = 215\ kHz$



ROM prediction compared against the linearized FOM at $x = 0.0045$. ROM is trained with impulse response and tested with sinusoidal input when back pressure is perturbed with $p_{back} \times 10^{-4}$ amplitude and $f = 215\ kHz$ frequency.

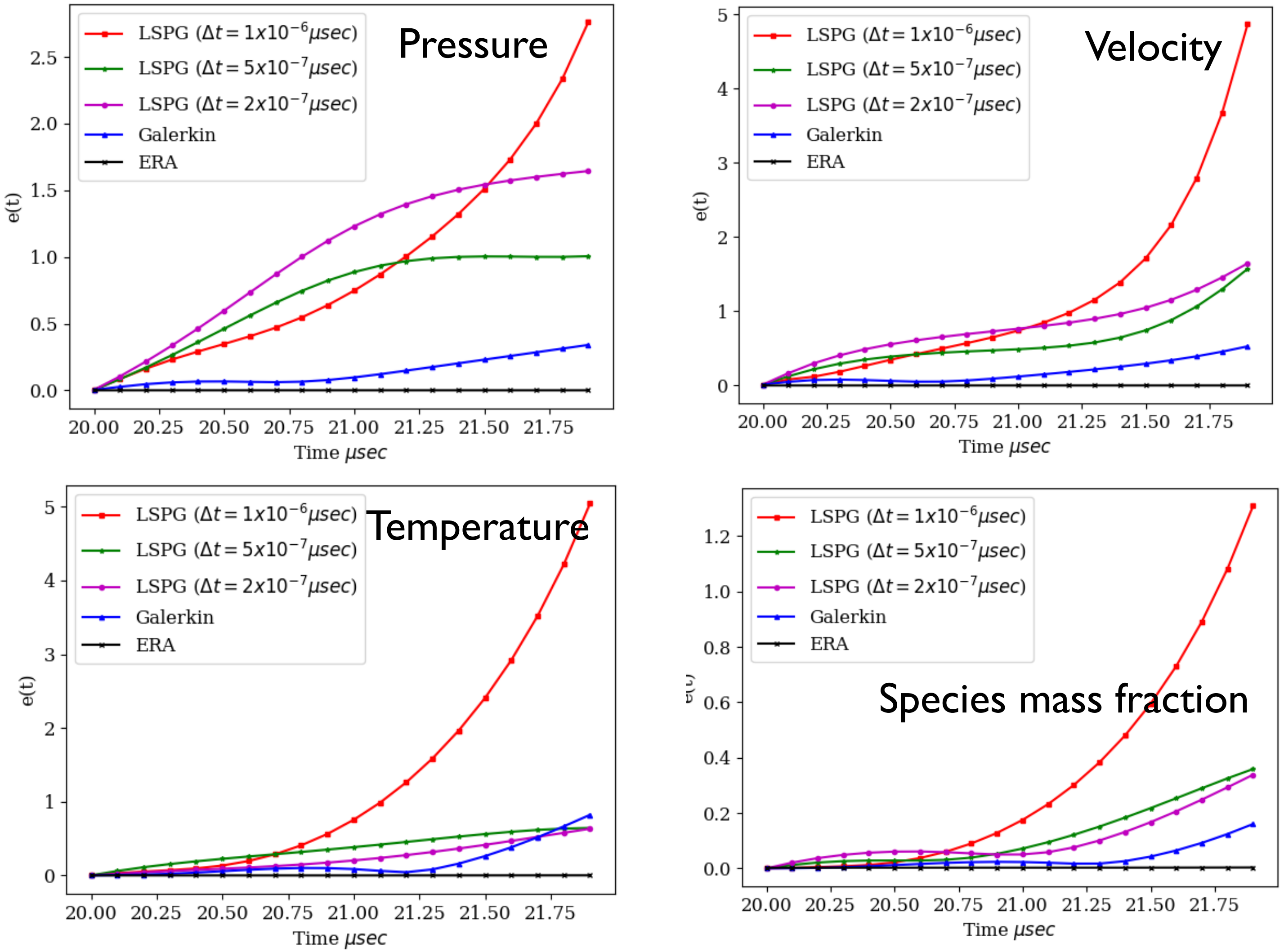
Predictive Performance of ROMs at 215 kHz

Relative error of ROM predictions for forcing frequency of 215 kHz. Galerkin and LSPG ROMs are trained with snapshots of 200, 210 and 220 kHz.

Impulse response snapshots are collected every 100 time steps and total impulse response sampling time is $100\ \mu sec$. The last 100 cells are not included in the analysis.

$$e^k = \frac{\left\| \mathbf{q}^k - \hat{\mathbf{q}}^k \right\|_2}{\left\| \mathbf{q}^k \right\|_2}, \qquad k = 1, \ldots, n_t$$

n_t: number of time steps.

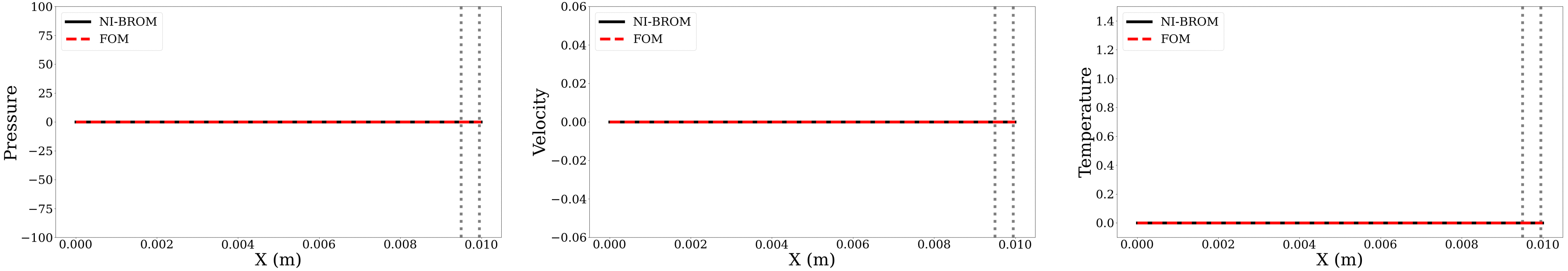


ERA with Domain Decomposition and Tangential Interpolation

Wall-clock Time (sec)	ROM 1	ROM 2	ROM 3	Total
With Tangential Interpolation	6.17	495.35	No improvements	1603.85
Without Tangential Interpolation	81.89	1609.32	1102.33	2793.54

One-time offline costs.

Online	0.116 sec
Online speedup	138.12



Balanced ROM predictions for the case with back pressure perturbation with an amplitude of $p_{back}\times10^{-4}$ and a frequency of $f = 215\ kHz$.

Rezaian et al., "Non-intrusive balancing transformation of highly stiff systems with lightly damped impulse response", *Philosophical Transactions of the Royal Society A*, 2022

Aeroacoustic Response Prediction

The high-fidelity model is based in the solution of the two-dimensional nonlinear Euler equations,

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0$$

$$p = (\gamma - 1)\left[e - \tfrac{\rho}{2}(u^2 + v^2)\right]$$

$$\mathbf{q} = [\rho \;\; \rho u \;\; \rho v \;\; e]^T$$

$$\mathbf{f} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ (e+p)u \end{bmatrix}, \qquad \mathbf{g} = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ (e+p)v \end{bmatrix}$$

u: horizontal component of velocity
v: vertical component of velocity
p: pressure
ρ: density
e: total energy
γ: ratio of specific heats

A cell-centered finite volume approach is used with the second-order Roe scheme to compute the fluxes subject to far-field boundary conditions. The second-order R-K scheme is used for time integration.

The equations are also linearized about a steady-state solution:

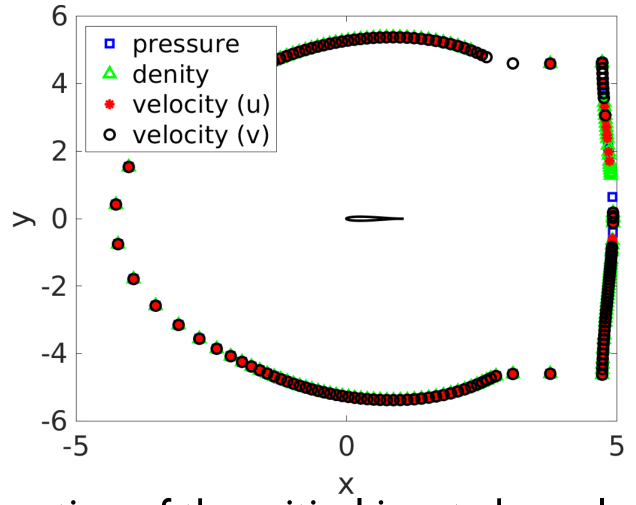
$$\dot{\mathbf{q}}' = \mathbf{J}\mathbf{q}' + \mathbf{b}$$

$$\mathbf{q}(\mathbf{x}, t) = \bar{\mathbf{q}}(\mathbf{x}) + \mathbf{q}'(\mathbf{x}, t) \qquad \mathbf{J} = \left[-\left(\frac{\partial(\mathbf{f}+\mathbf{g})}{\partial \mathbf{q}}\right)_C + \left(\frac{\partial(\mathbf{f}+\mathbf{g})}{\partial \mathbf{q}}\right)_L - \left(\frac{\partial(\mathbf{f}+\mathbf{g})}{\partial \mathbf{q}}\right)_R\right]_{\mathbf{q}=\bar{\mathbf{q}}}$$

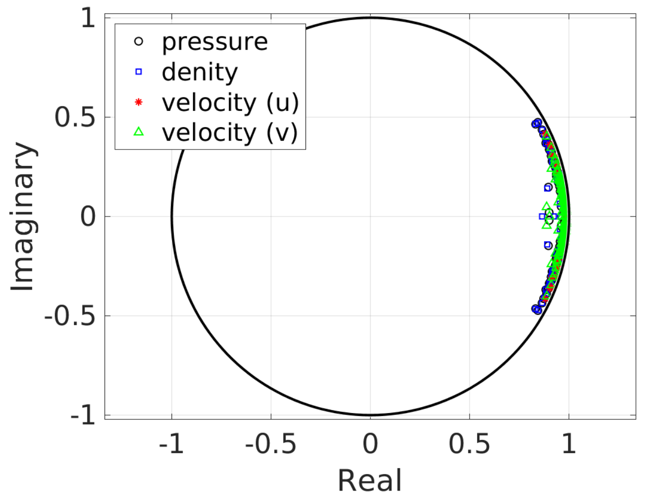
Actuator Selection with the Low-fidelity Gappy POD Method

Motivation: to avoid running the FOM 604 times to collect the Markov sequence corresponding to each input channel for training the ERA ROM.

1. Markov sequences corresponding to all of the input channels are computed using the coarser grid and stored in matrix $\mathbf{Q}$.
2. POD modes are computed based on SVD of $\mathbf{Q}$.
3. The observation matrix is defined as: $\mathbf{S} \approx \mathbf{P\Phi a}$
4. Considering an over-sampled case, the sampling matrix $\mathbf{P}$ is obtained by QR factorization with column pivoting:
$$(\mathbf{\Phi\Phi}^T)\tilde{\mathbf{P}} = \tilde{\mathbf{Q}}\tilde{\mathbf{R}} \qquad \mathbf{P}^T = \tilde{\mathbf{P}}(:, 1 : p_s)$$
5. The POD coefficients are computed as: $\mathbf{a} = (\mathbf{P\Phi})^+\mathbf{S}$
6. The POD modes and coefficients are then used to reconstruct the Markov sequence: $\hat{\mathbf{Q}} = \mathbf{\Phi a}$



Location of the critical input channels chosen by the low-fidelity gappy POD approach.



Eigenvalues of the ERA ROMs.

Rezaian and Duraisamy, "Data-driven Balanced Truncation for Predictive Model Order Reduction of Aeroacoustic Response", AIAAJ, 2023

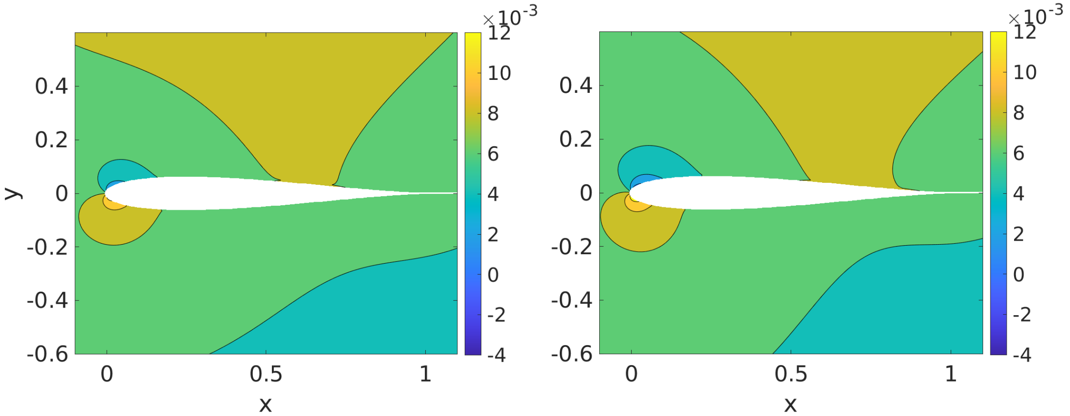
ERA ROMs in a Purely Predictive Setting

State fluctuations computed at $(x, y) = (0.4628, -0.3129)$, when the far-field boundary is perturbed by a sinusoidal input,

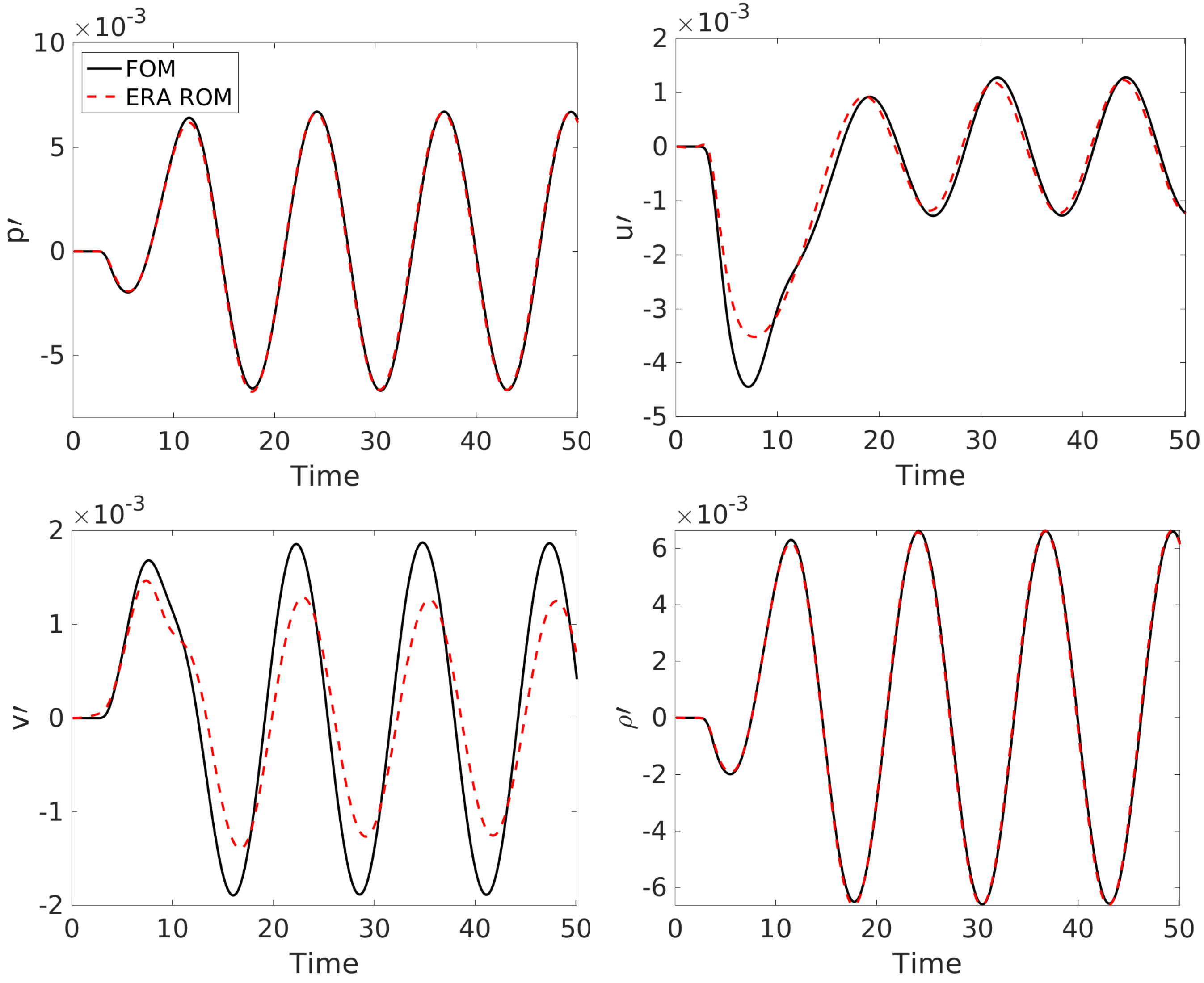
$$u_g = -\frac{\epsilon w}{\sqrt{2}} \sin(\omega t); \quad v_g = \frac{\epsilon w}{\sqrt{2}} \sin(\omega t)$$

$$\epsilon = 0.02,\ w = \sqrt{u_\infty^2 + v_\infty^2},\ \omega = \frac{2kw}{c}$$

$$\boxed{u'_\infty = u_g(t) \qquad v'_\infty = v_g(t) \qquad p' = \rho' = 0}$$



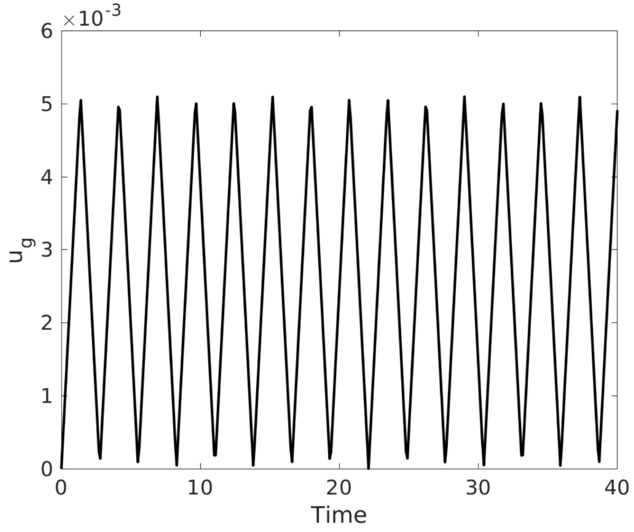
FOM (left) and ROM (right) snapshots at $t = 48.5$.

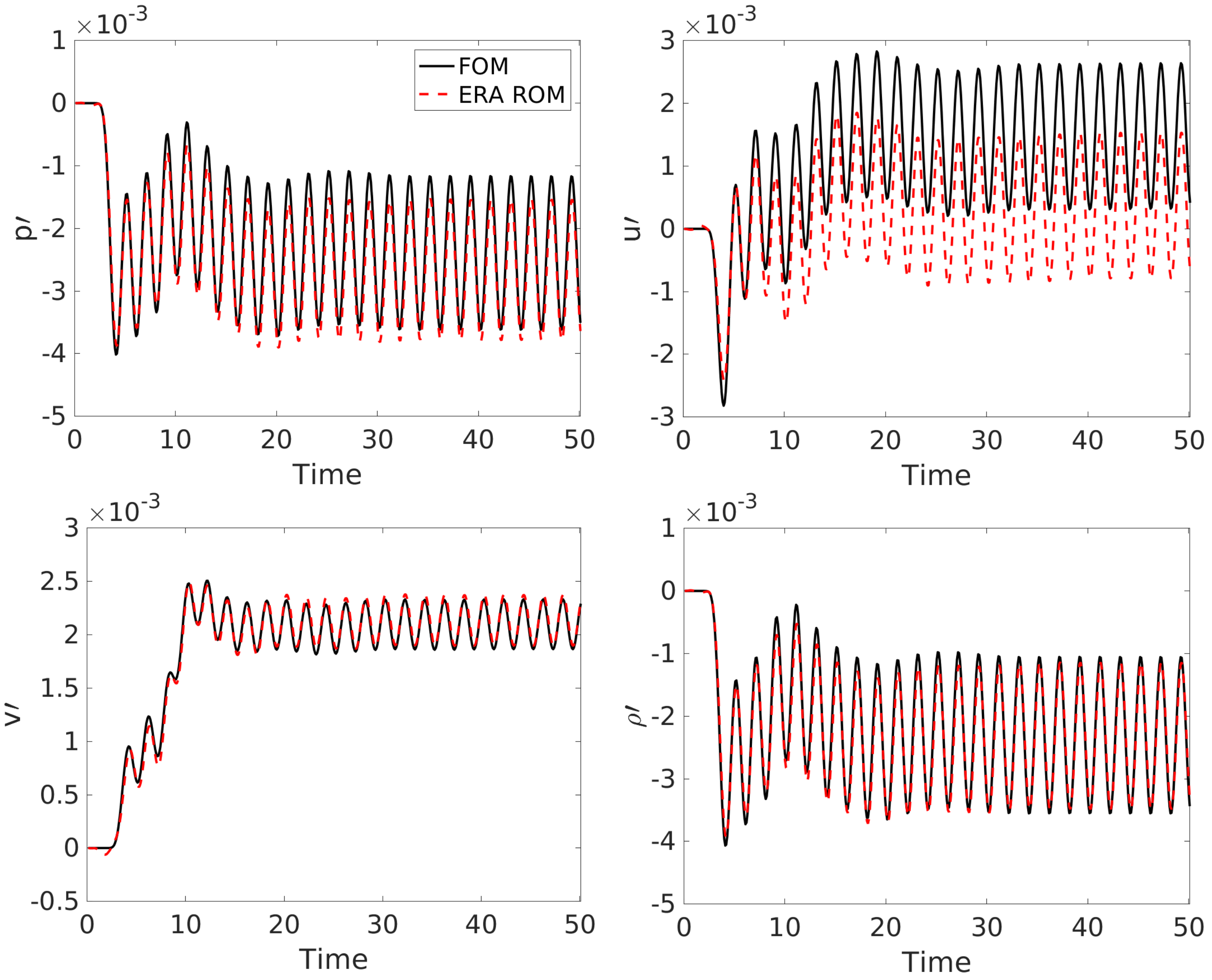


ERA ROMs in a Purely Predictive Setting

State fluctuations computed at $(x, y) = (0.2920, 0.3221)$, when the far-field boundary is perturbed by a triangular wave input,

$$u_g = -\frac{2\epsilon w}{\sqrt{2}}\left|\omega t - \left\lfloor \omega t + \frac{1}{2} \right\rfloor\right|; \quad v_g = \frac{2\epsilon w}{\sqrt{2}}\left|\omega t - \left\lfloor \omega t + \frac{1}{2} \right\rfloor\right|$$





ERA ROMs in a Purely Predictive Setting

State fluctuations computed at $(x, y) = (0.4628, -0.3129)$, when the far-field boundary is perturbed by a step input,

$$u_g = \begin{cases} -0.005 & 0.5 \le t < 1 \\ 0 & otherwise \end{cases} \quad ; \quad v_g = \begin{cases} 0.005 & 0.5 \le t < 1 \\ 0 & otherwise \end{cases}$$

Speedup	
Offline	1. Linearization reduced the cost of computing each Markov sequence by an order of magnitude. 2. The low-fidelity gappy POD approach reduced the input channels by a factor of 4 and the computation time of the training data by 353 hr. 3. Constructing ROM matrices without the gappy POD approach was infeasible.
Online	The ERA ROM achieved a speedup factor of 258.