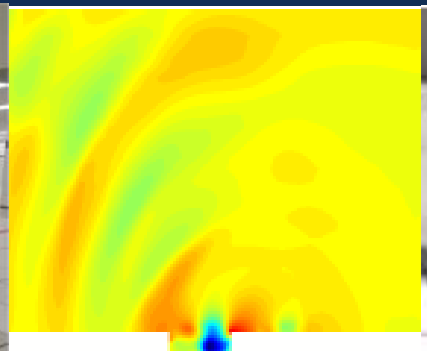
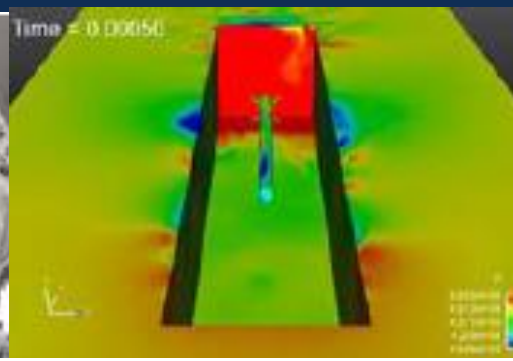
SAND2017-4018PE

Exceptional service in the national interest

Sandia National Laboratories





Approaches for building stable projection-based reduced order models for compressible flow

Irina K. Tezaur[1], Matt Barone[1], Jeff Fike[1], Maciej Balajewicz[2], Srini Arunajatesan[1], Bart van Bloemen Waanders[1], Earl Dowell[3]

[1] Sandia National Laboratories, Livermore CA and Albuquerque, NM
[2] University of Illinois Urbana-Champaign [3] Duke University

Virginia Tech Math Department Seminar – May 1, 2017

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

Outline

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

Outline

- I have worked on model reduction for compressible flows since 2007.
  → focus on ROM stability
- This talk surveys some of the approaches for building stable ROMs I have helped to develop.

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

Outline

1. **Motivation.**
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

Motivation

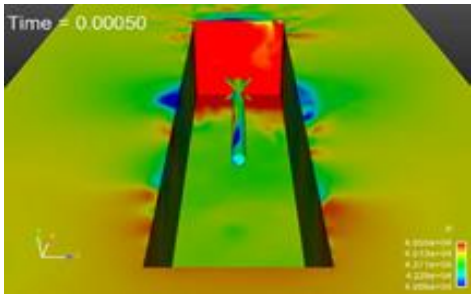
Despite improved algorithms and powerful supercomputers, **“high-fidelity” models** are often too expensive for use in a design or analysis setting.

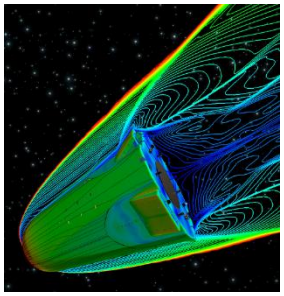
Motivation

Despite improved algorithms and powerful supercomputers, **"high-fidelity" models** are often too expensive for use in a design or analysis setting.

<u>Application areas in which this situation arises:</u>

- **Captive-carry** and **re-entry environments:** Large Eddy Simulations (LES) runs require very fine meshes and can take on the order of ***weeks***.





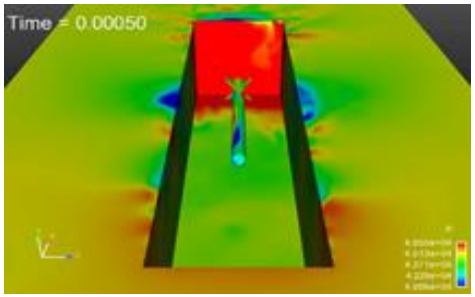
Motivation

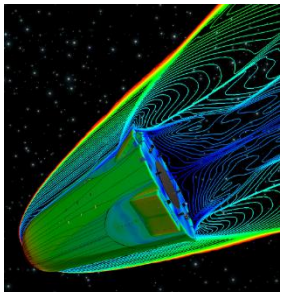
Despite improved algorithms and powerful supercomputers, **"high-fidelity" models** are often too expensive for use in a design or analysis setting.

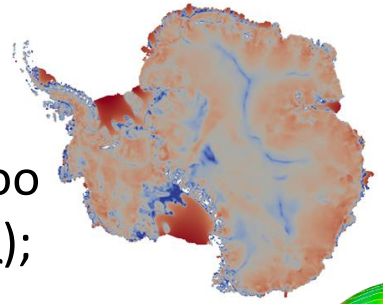
<u>Application areas in which this situation arises:</u>

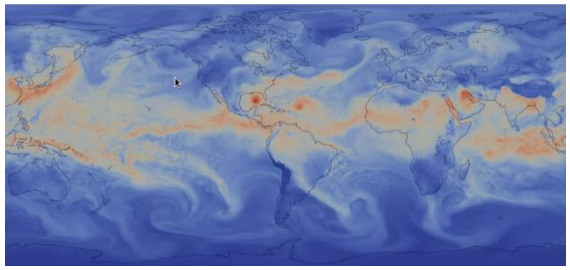
- **Captive-carry** and **re-entry environments:** Large Eddy Simulations (LES) runs require very fine meshes and can take on the order of ***weeks***.

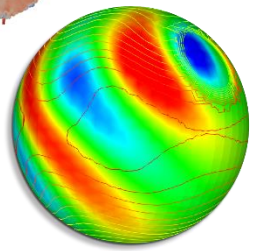
- **Climate modeling** (e.g., land-ice, atmosphere): high-fidelity simulations too costly for uncertainty quantification (UQ); Bayesian inference of high-dimensional parameter fields is intractable.







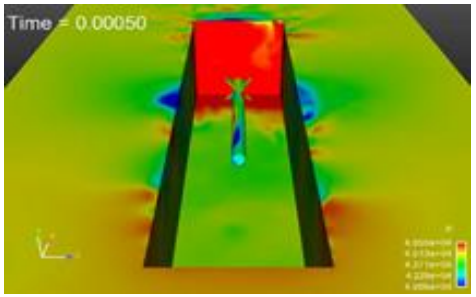


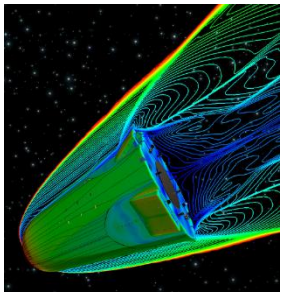


Motivation

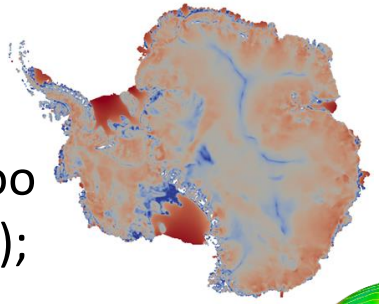
Despite improved algorithms and powerful supercomputers, **"high-fidelity" models** are often too expensive for use in a design or analysis setting.

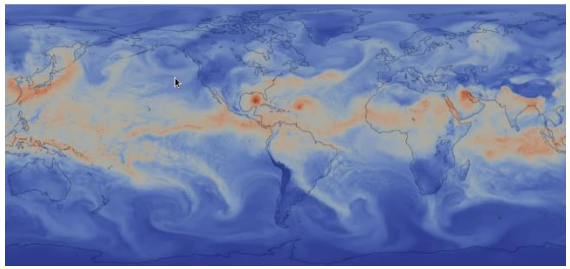
Application areas in which this situation arises:



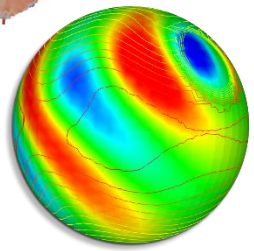


- **Captive-carry** and **re-entry environments:** Large Eddy Simulations (LES) runs require very fine meshes and can take on the order of ***weeks***.





- **Climate modeling** (e.g., land-ice, atmosphere): high-fidelity simulations too costly for uncertainty quantification (UQ); Bayesian inference of high-dimensional parameter fields is intractable.



- **Wind plant simulations:** wind turbines need to be controlled in real-time to attain optimal energy capture for wind plant system.

Surrogate models for time-critical prediction

Surrogate low-dimensional models can enable ***time-critical*** prediction and ***many-query*** analyses (e.g., design, optimization, control, UQ).

Surrogate models for time-critical prediction

Surrogate low-dimensional models can enable ***time-critical*** prediction and ***many-query*** analyses (e.g., design, optimization, control, UQ).

- Three flavors of "**surrogate models**" for time-critical prediction:

1. **Data-fit models** (e.g., response surfaces, Kriging): directly model I/O map
   - *Upsides:* easy to implement, low complexity
   - *Downsides:* blind to problem physics, impractical for large-dimensional input spaces

2. **Lower-fidelity/reduced-physics models** (e.g., coarse mesh, Euler vs. Navier-Stokes)
   - *Upsides:* physics-based, no training required, easy to implement in existing codes.
   - *Downsides:* evaluation times generally not fast enough for time-critical problems, may have inconsistency with higher fidelity models.

3. **Projection-based reduced order models (ROMs)**: approximately solve state equations, then compute outputs
   - *Upsides:* physics-based, can be evaluated in real or near-real time.
   - *Downsides:* intrusive to implement, robustness across wide range of parameters not guaranteed

Surrogate models for time-critical prediction

Surrogate low-dimensional models can enable ***time-critical*** prediction and ***many-query*** analyses (e.g., design, optimization, control, UQ).

- Three flavors of "**surrogate models**" for time-critical prediction:

1. **Data-fit models** (e.g., response surfaces, Kriging): directly model I/O map
   - *Upsides:* easy to implement, low complexity
   - *Downsides:* blind to problem physics, impractical for large-dimensional input spaces

2. **Lower-fidelity/reduced-physics models** (e.g., coarse mesh, Euler vs. Navier-Stokes)
   - *Upsides:* physics-based, no training required, easy to implement in existing codes.
   - *Downsides:* evaluation times generally not fast enough for time-critical problems, may have inconsistency with higher fidelity models.

3. **Projection-based reduced order models (ROMs)**: approximately solve state equations, then compute outputs

   This talk
   - *Upsides:* physics-based, can be evaluated in real or near-real time.
   - *Downsides:* intrusive to implement, robustness across wide range of parameters not guaranteed

Outline

1. Motivation.
2. **Projection-based model order reduction.**
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

Offline/online strategy

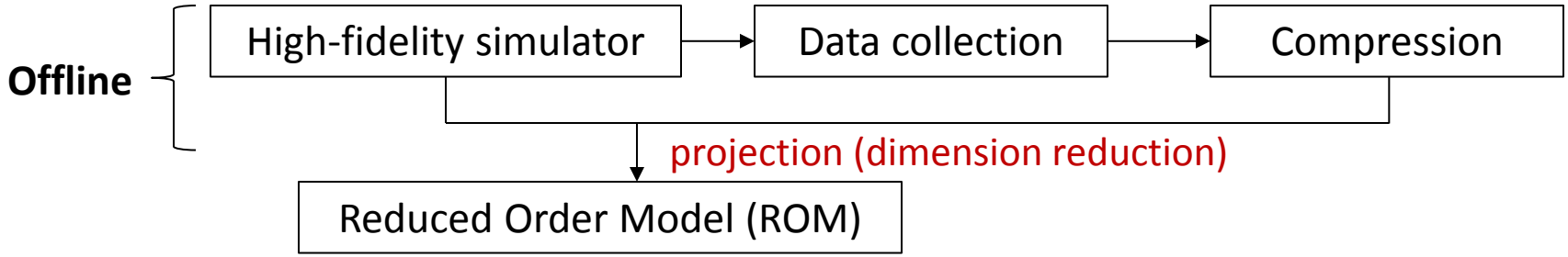
- Three underlying ***mathematical premises*** of projection-based model order reduction (MOR):

Offline/online strategy

Compression

- Three underlying ***mathematical premises*** of projection-based model order reduction (MOR):
  1. **Compression:** solution of governing parametric PDE or system of PDEs lies in a subspace of significantly lower dimension.

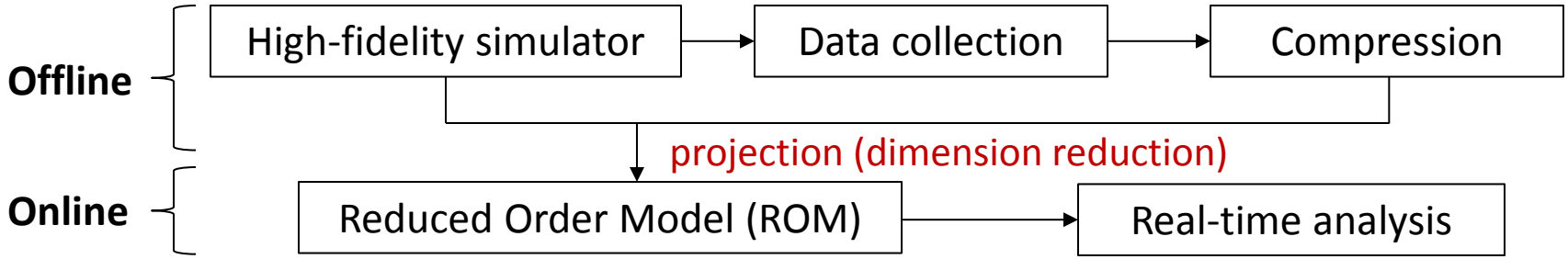
Offline/online strategy



- Three underlying ***mathematical premises*** of projection-based model order reduction (MOR):

  1. **Compression:** solution of governing parametric PDE or system of PDEs lies in a subspace of significantly lower dimension.
  2. **Offline training:** subspace can be identified/learned offline via training simulation and high-fidelity model can be reformulated with respect to this subspace.

Offline/online strategy



- Three underlying ***mathematical premises*** of projection-based model order reduction (MOR):

  1. **Compression:** solution of governing parametric PDE or system of PDEs lies in a subspace of significantly lower dimension.
  2. **Offline training:** subspace can be identified/learned offline via training simulation and high-fidelity model can be reformulated with respect to this subspace.
  3. **Online prediction:** identified parametric ROMs are capable of providing new solutions at a fraction of the computational cost.

Proper Orthogonal Decomposition (POD)/ Galerkin method to model reduction

High fidelity CFD simulations:

Snapshot 1
Snapshot 2
⋮
Snapshot K

Step 1 →

Fluid modal decomposition (POD):

$$\boldsymbol{u}(\boldsymbol{x},t) \approx \sum_{k=1}^{M} a_{M,k}(t)\,\boldsymbol{\phi}_k(\boldsymbol{x})$$

Step 2 →

Galerkin projection of fluid PDEs:

$$(\boldsymbol{\phi}_k, \dot{\boldsymbol{u}} + \nabla \cdot \boldsymbol{F}(\boldsymbol{u})) = 0$$

- ***Snapshot matrix***: $\boldsymbol{X} = (\boldsymbol{x}^1, \ldots, \boldsymbol{x}^K) \in \mathbb{R}^{NxK}$
- ***SVD:*** $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$
- ***Truncation:*** $\boldsymbol{\Phi}_M = (\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_M) = \boldsymbol{U}(:, 1{:}M)$

Basis energy = $\sum_{i=1}^{M} \sigma^2$

"Small" ROM ODE system:

$$\dot{a}_{M,k} = f(a_{M,1}, \ldots, a_{M,M})$$

FOM = full order model
N = # of dofs in FOM
K = # of snapshots
M = # of dofs in ROM
$(M << N, M << K)$

Continuous vs. discrete Galerkin projection

Continuous Projection

Governing PDEs
$\dot{\boldsymbol{q}} = \mathcal{L}\boldsymbol{q}$

↓

High-fidelity model
$\dot{\boldsymbol{q}}_N = \boldsymbol{A}_N \boldsymbol{q}_N$

↓

Continuous modal basis* $\boldsymbol{\phi}_j(\boldsymbol{x})$

↓

Projection of governing PDEs (numerical integration)

↓

ROM
$\dot{a}_j = (\boldsymbol{\phi}_j, \mathcal{L}\boldsymbol{\phi}_k) a_k$

If PDEs are linear or have polynomial non-linearities, projection can be calculated in **offline stage** of MOR.

Discrete Projection

Governing PDEs
$\dot{\boldsymbol{q}} = \mathcal{L}\boldsymbol{q}$

↓

High-fidelity model
$\dot{\boldsymbol{q}}_N = \boldsymbol{A}_N \boldsymbol{q}_N$

↓

Discrete modal basis $\boldsymbol{\Phi}$

↓

Projection of FOM (matrix operation)

↓

ROM
$\dot{\boldsymbol{a}}_M = \boldsymbol{\Phi}^T \boldsymbol{A}_N \boldsymbol{\Phi} \boldsymbol{a}_M$

* Continuous function space defined using e.g., finite elements.

Continuous vs. discrete Galerkin projection

Continuous Projection

Governing PDEs
$\dot{\boldsymbol{q}} = \mathcal{L}\boldsymbol{q}$

↓

High-fidelity model
$\dot{\boldsymbol{q}}_N = \boldsymbol{A}_N \boldsymbol{q}_N$

↓

Continuous modal basis* $\boldsymbol{\phi}_j(\boldsymbol{x})$

↓

Projection of governing PDEs (numerical integration)

↓

ROM
$\dot{a}_j = (\boldsymbol{\phi}_j, \mathcal{L}\boldsymbol{\phi}_k) a_k$

If PDEs are linear or have polynomial non-linearities, projection can be calculated in **offline stage** of MOR.

Most of my ROM work

Discrete Projection

Governing PDEs
$\dot{\boldsymbol{q}} = \mathcal{L}\boldsymbol{q}$

↓

High-fidelity model
$\dot{\boldsymbol{q}}_N = \boldsymbol{A}_N \boldsymbol{q}_N$

↓

Discrete modal basis $\boldsymbol{\Phi}$

↓

Projection of FOM (matrix operation)

↓

ROM
$\dot{\boldsymbol{a}}_M = \boldsymbol{\Phi}^T \boldsymbol{A}_N \boldsymbol{\Phi} \boldsymbol{a}_M$

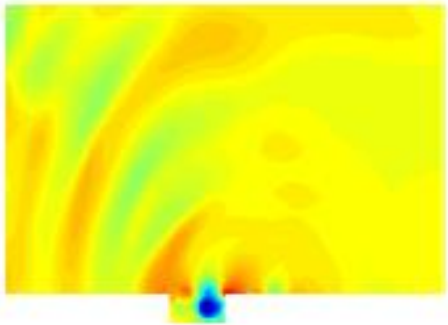
* Continuous function space defined using e.g., finite elements.

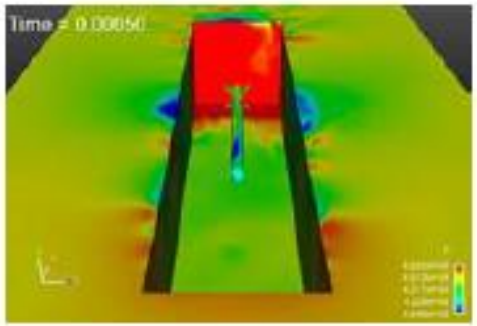
Outline

1. Motivation.
2. Projection-based model order reduction.
3. **Targeted application: compressible cavity flow.**
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

Targeted application: compressible flow

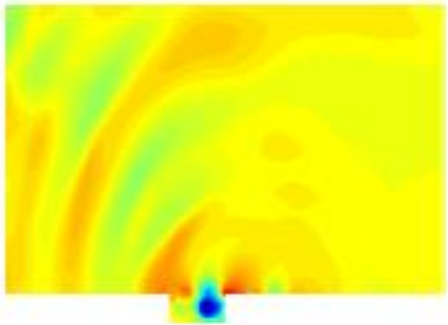
- We are interested in the ***compressible captive-carry problem***.

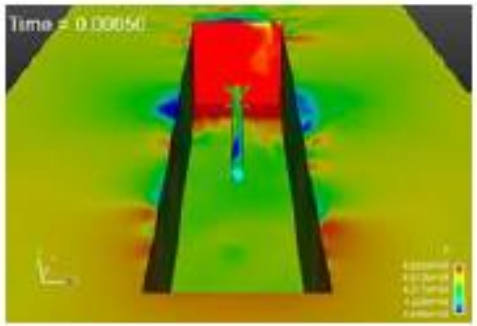




Targeted application: compressible flow

- We are interested in the ***compressible captive-carry problem***.

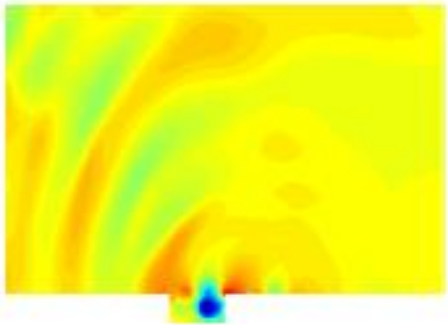


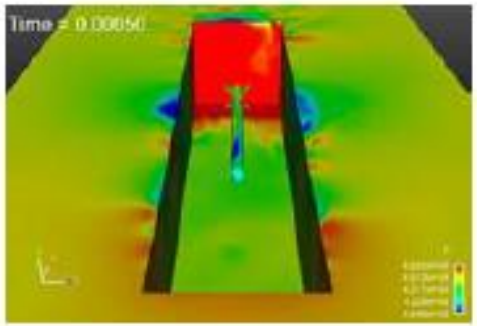


- Of <u>primary interest</u> are ***long-time predictive simulations***: ROM run at same parameters as FOM but much longer in time.

Targeted application: compressible flow

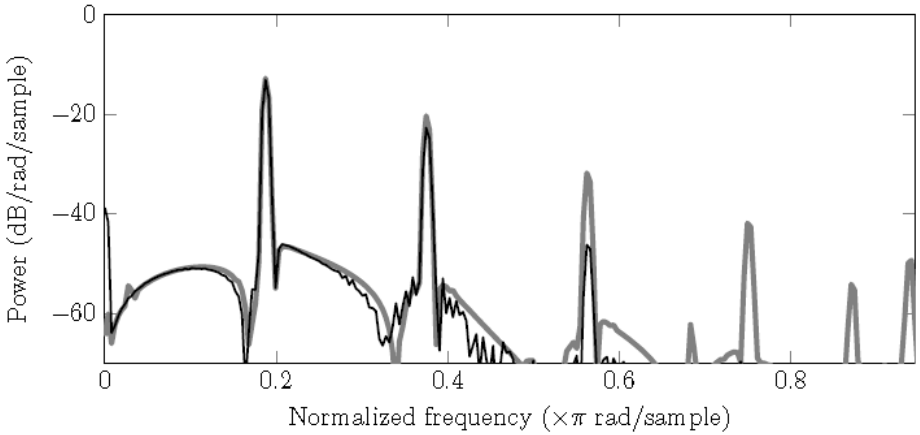
- We are interested in the ***compressible captive-carry problem***.





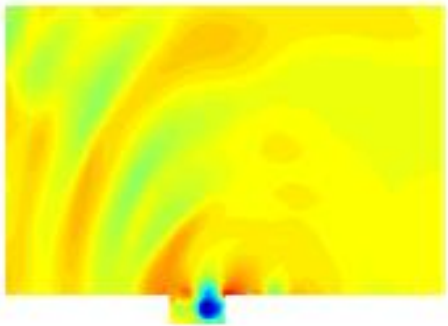
- Of <u>primary interest</u> are ***long-time predictive simulations***: ROM run at same parameters as FOM but much longer in time.

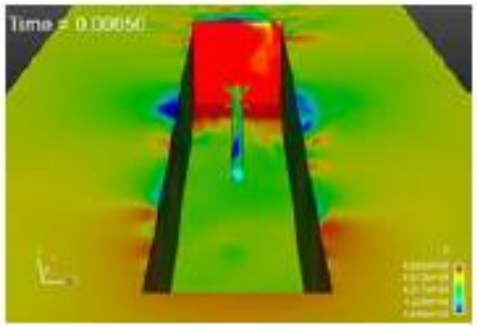
  → <u>QoIs:</u> statistics of flow, e.g., pressure Power Spectral Densities (***PSDs***) [right].



Targeted application: compressible flow

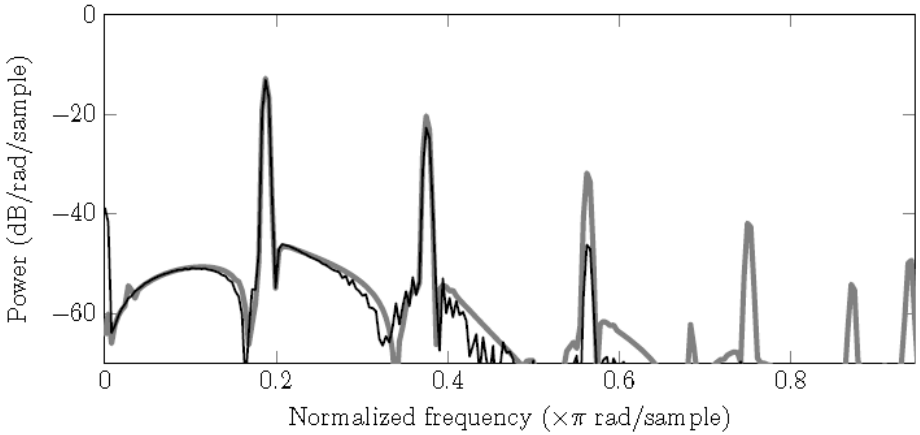
- We are interested in the ***compressible captive-carry problem***.





- Of <u>primary interest</u> are ***long-time predictive simulations***: ROM run at same parameters as FOM but much longer in time.

  → <u>QoIs:</u> statistics of flow, e.g., pressure Power Spectral Densities (***PSDs***) [right].



- <u>Secondary interest</u>: ROMs robust w.r.t. ***parameter changes*** (e.g., Reynolds, Mach number) for enabling ***uncertainty quantification***.

Targeted application: compressible flow

- ***Compressible flow equations***:

$$\rho_{,t} + (\rho u_j)_{,j} = 0$$
$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} + p_{,i} - \tau_{ij,j} = 0$$
$$(\rho e)_{,t} + (p u_i)_{,i} - (u_i \tau_{ij})_{,j} = 0$$

Targeted application: compressible flow

- ***Compressible flow equations***:

$$\rho_{,t} + (\rho u_j)_{,j} = 0$$
$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} + p_{,i} - \tau_{ij,j} = 0$$
$$(\rho e)_{,t} + (p u_i)_{,i} - (u_i \tau_{ij})_{,j} = 0$$

Majority of fluid ROMs in literature are for ***incompressible flow***.

Targeted application: compressible flow

- ***Compressible flow equations***:

$$\rho_{,t} + (\rho u_j)_{,j} = 0$$
$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} + p_{,i} - \tau_{ij,j} = 0$$
$$(\rho e)_{,t} + (p u_i)_{,i} - (u_i \tau_{ij})_{,j} = 0$$

- Compressible flow solutions can exhibit the following ***complex features***:
  - Sharp boundary layers.
  - Turbulence/chaotic dynamics.
  - Shocks.

Majority of fluid ROMs in literature are for ***incompressible flow***.

Targeted application: compressible flow

- ***Compressible flow equations***:

$$\rho_{,t} + (\rho u_j)_{,j} = 0$$
$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} + p_{,i} - \tau_{ij,j} = 0$$
$$(\rho e)_{,t} + (p u_i)_{,i} - (u_i \tau_{ij})_{,j} = 0$$

- Compressible flow solutions can exhibit the following ***complex features***:
  - Sharp boundary layers.
  - Turbulence/chaotic dynamics.
  - Shocks.

Majority of fluid ROMs in literature are for ***incompressible flow***.

- Desired numerical properties of ROMs:
  - ***Consistency*** (w.r.t. the continuous PDEs).
  - ***Stability:*** if full order model (FOM) is stable, ROM should be stable.
  - ***Convergence:*** requires consistency and stability.
  - ***Accuracy*** (w.r.t. FOM).
  - ***Efficiency***.
  - ***Robustness*** (w.r.t. time or parameter changes).

Targeted application: compressible flow

- ***Compressible flow equations***:

$$\rho_{,t} + (\rho u_j)_{,j} = 0$$
$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} + p_{,i} - \tau_{ij,j} = 0$$
$$(\rho e)_{,t} + (p u_i)_{,i} - (u_i \tau_{ij})_{,j} = 0$$

- Compressible flow solutions can exhibit the following ***complex features***:
  - Sharp boundary layers.
  - Turbulence/chaotic dynamics.
  - Shocks.

Majority of fluid ROMs in literature are for ***incompressible flow***.

- Desired numerical properties of ROMs:
  - ***Consistency*** (w.r.t. the continuous PDEs).
  - ***Stability:*** if full order model (FOM) is stable, ROM should be stable.
  - ***Convergence:*** requires consistency and stability.
  - ***Accuracy*** (w.r.t. FOM).
  - ***Efficiency***.
  - ***Robustness*** (w.r.t. time or parameter changes).

Focus of this talk is ***ROM stability***.

ROM instability problem

Stability can be a real problem for compressible flow ROMs!

ROM instability problem

Stability can be a real problem for compressible flow ROMs!

- A compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but ***unstable*** for other choices of basis size (Bui-Tanh *et al.* 2007; Barone *et al.* 2009).

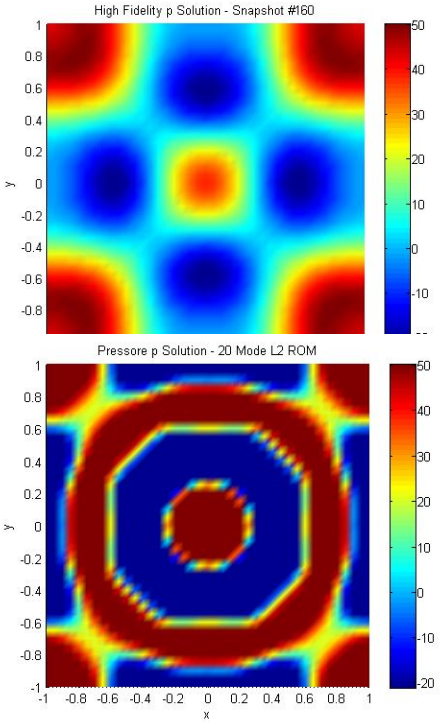
ROM instability problem

Stability can be a real problem for compressible flow ROMs!

- A compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but ***unstable*** for other choices of basis size (Bui-Tanh *et al.* 2007, Barone *et al.* 2009).
  - Instability can manifest itself in different ways, e.g., blow-up during time-integration, spurious oscillations, ROM computes non-physical quantities.

Top right: FOM

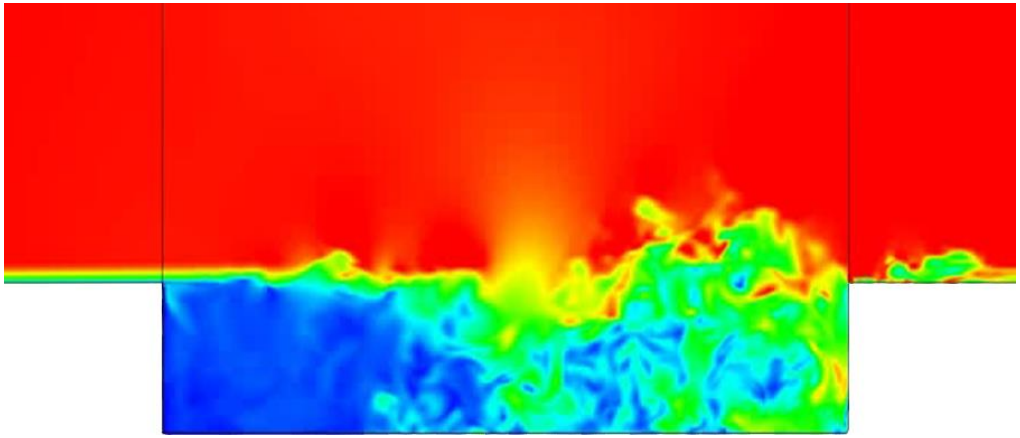
Bottom right: ROM



ROM instability problem

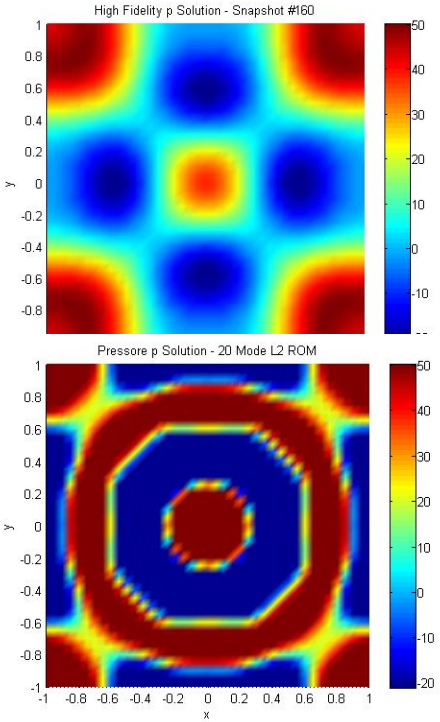
Stability can be a real problem for compressible flow ROMs!

- A compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but ***unstable*** for other choices of basis size (Bui-Tanh *et al.* 2007, Barone *et al.* 2009).
  - Instability can manifest itself in different ways, e.g., blow-up during time-integration, spurious oscillations, ROM computes non-physical quantities.
- There is **no *a priori* stability guarantee** for POD/Galerkin ROMs for compressible flow!



Top right: FOM

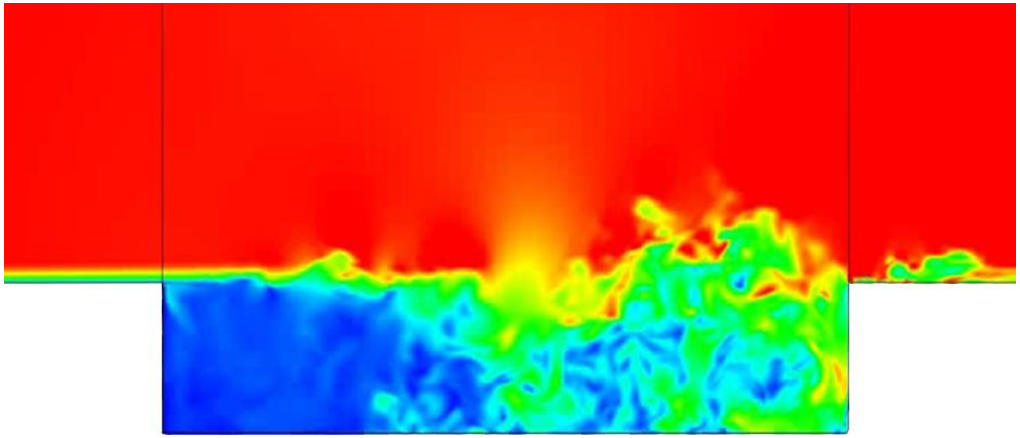
Bottom right: ROM



ROM instability problem

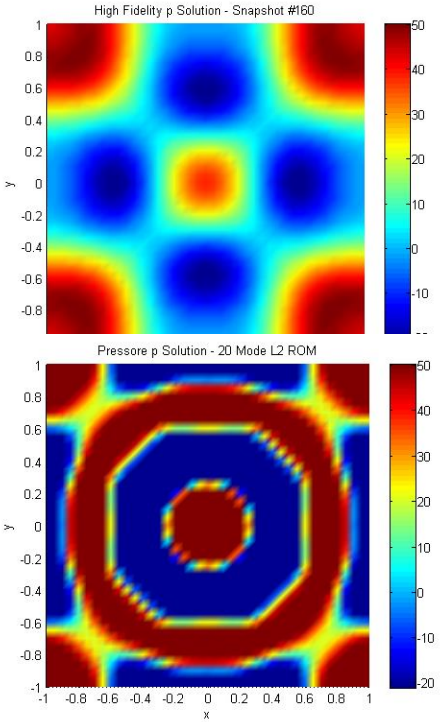
Stability can be a real problem for compressible flow ROMs!

- A compressible fluid POD/Galerkin ROM might be stable for a given number of modes, but ***unstable*** for other choices of basis size (Bui-Tanh *et al.* 2007, Barone *et al.* 2009).
  - Instability can manifest itself in different ways, e.g., blow-up during time-integration, spurious oscillations, ROM computes non-physical quantities.
- There is **no *a priori* stability guarantee** for POD/Galerkin ROMs for compressible flow!
- Stability of a ROM is commonly evaluated *a posteriori* – **RISKY!**



Top right: FOM

Bottom right: ROM



ROM instability problem

- Instability can be due to:

ROM instability problem

- Instability can be due to:
  1. ***Choice of inner product***: Galerkin projection + L^2 inner product is unstable.

ROM instability problem

- Instability can be due to:
  1. ***Choice of inner product***: Galerkin projection + L^2 inner product is unstable.
  2. ***Basis truncation*:** destroys balance between energy production & dissipation.

ROM instability problem

- Instability can be due to:
  1. ***Choice of inner product***: Galerkin projection + L^2 inner product is unstable.
  2. ***Basis truncation*:** destroys balance between energy production & dissipation.

- Some* remedies:

	Continuous Projection	**Discrete Projection**
Change projection (*a priori*)	Energy inner products (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	Eigenvalue reassignment (IKT *et al.*)
Change ROM basis (*a posteriori*)	Basis rotation (Balajewicz, IKT, *et al.*)	Optimization-based right basis modification (Amsallem *et al.*)

* This is not a complete list!

ROM instability problem

- Instability can be due to:
  1. ***Choice of inner product***: Galerkin projection + L^2 inner product is unstable.
  2. ***Basis truncation*:** destroys balance between energy production & dissipation.

- Some* remedies:

	Continuous Projection	**Discrete Projection**
Change projection (*a priori*)	**Energy inner products** (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	**Eigenvalue reassignment** (IKT *et al.*)
Change ROM basis (*a posteriori*)	**Basis rotation** (Balajewicz, IKT, *et al.*)	Optimization-based right basis modification (Amsallem *et al.*)

This talk is on *a priori* (intrusive) and *a posteriori* (non-intrusive) approaches for remedying ROM instability for POD/Galerkin ROMs.

* This is not a complete list!

Outline

	Continuous Projection	Discrete Projection
Change projection (*a priori*)	**Energy inner products** (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	Eigenvalue reassignment (IKT *et al.*)
Change ROM basis (*a posteriori*)	Basis rotation (Balajewicz, IKT, *et al.*)	Optimization-based right basis modificatiOon (Amsallem)

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. **Approaches for building *a priori* stable ROMs.**
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

Energy-stability

- **Practical Definition:** Numerical solution does not “blow up” in finite time.

Energy-stability

- **Practical Definition:** Numerical solution does not “blow up” in finite time.
- **More Precise Definition:** Numerical discretization does not introduce any spurious instabilities inconsistent with natural instability modes supported by the governing continuous PDEs.

Energy-stability

- **Practical Definition:** Numerical solution does not “blow up” in finite time.
- **More Precise Definition:** Numerical discretization does not introduce any spurious instabilities inconsistent with natural instability modes supported by the governing continuous PDEs.

Numerical solutions must maintain proper **energy balance**.

Energy-stability

- **Practical Definition:** Numerical solution does not "blow up" in finite time.
- **More Precise Definition:** Numerical discretization does not introduce any spurious instabilities inconsistent with natural instability modes supported by the governing continuous PDEs.

Numerical solutions must maintain proper **energy balance**.

- Stability of ROM is intimately tied to choice of **inner product** for the Galerkin projection (Rowley *et al.*, 2004; Barone & IKT, 2009; Serre *et al.*, 2012).

Energy-stability

- **Practical Definition:** Numerical solution does not "blow up" in finite time.
- **More Precise Definition:** Numerical discretization does not introduce any spurious instabilities inconsistent with natural instability modes supported by the governing continuous PDEs.

Numerical solutions must maintain proper **energy balance**.

- Stability of ROM is intimately tied to choice of **inner product** for the Galerkin projection (Rowley *et al.*, 2004; Barone & IKT, 2009; Serre *et al.*, 2012).
  - Norm induced by inner product in which ROM is constructed should correspond to **energy measure** for system.

Energy-stability

- **Practical Definition:** Numerical solution does not "blow up" in finite time.
- **More Precise Definition:** Numerical discretization does not introduce any spurious instabilities inconsistent with natural instability modes supported by the governing continuous PDEs.

Numerical solutions must maintain proper **energy balance**.

- Stability of ROM is intimately tied to choice of **inner product** for the Galerkin projection (Rowley *et al.*, 2004; Barone & IKT, 2009; Serre *et al.*, 2012).
  - Norm induced by inner product in which ROM is constructed should correspond to **energy measure** for system.
  - ***Incompressible flow:*** solution vector is $\boldsymbol{u} \Rightarrow L^2$ inner product induces norm representing global kinetic energy ($E = \frac{1}{2}||\boldsymbol{u}||_2^2$).

Energy-stability

- **Practical Definition:** Numerical solution does not "blow up" in finite time.
- **More Precise Definition:** Numerical discretization does not introduce any spurious instabilities inconsistent with natural instability modes supported by the governing continuous PDEs.

Numerical solutions must maintain proper **energy balance**.

- Stability of ROM is intimately tied to choice of **inner product** for the Galerkin projection (Rowley *et al.*, 2004; Barone & IKT, 2009; Serre *et al.*, 2012).
  - Norm induced by inner product in which ROM is constructed should correspond to **energy measure** for system.
  - ***Incompressible flow:*** solution vector is $\boldsymbol{u} \Rightarrow L^2$ inner product induces norm representing global kinetic energy ($E = \frac{1}{2}||\boldsymbol{u}||_2^2$).

    $\rightarrow L^2$ inner product is physically sensible, since POD modes represent optimally kinetic energy present in the ensemble from which they are constructed:

$$\boldsymbol{u} = \sum_{i=1}^{M} a_i(t)\boldsymbol{\phi}_i \qquad \Rightarrow \qquad 2E = (u,u) = \sum_{i=1}^{M} a_i(t)\boldsymbol{\phi}_i$$

Energy-stability

- **Practical Definition:** Numerical solution does not "blow up" in finite time.
- **More Precise Definition:** Numerical discretization does not introduce any spurious instabilities inconsistent with natural instability modes supported by the governing continuous PDEs.

Numerical solutions must maintain proper **energy balance**.

- Stability of ROM is intimately tied to choice of **inner product** for the Galerkin projection (Rowley *et al.*, 2004; Barone & IKT, 2009; Serre *et al.*, 2012).
  - Norm induced by inner product in which ROM is constructed should correspond to **energy measure** for system.
  - ***Incompressible flow:*** solution vector is $\boldsymbol{u} \Rightarrow L^2$ inner product induces norm representing global kinetic energy ($E = \frac{1}{2}||\boldsymbol{u}||_2^2$).

    $\rightarrow L^2$ inner product is physically sensible, since POD modes represent optimally kinetic energy present in the ensemble from which they are constructed:

$$\boldsymbol{u} = \sum_{i=1}^{M} a_i(t)\boldsymbol{\phi}_i \qquad \Rightarrow \qquad 2E = (u,u) = \sum_{i=1}^{M} a_i(t)\boldsymbol{\phi}_i$$

Not the case for compressible flow!

Energy-stability

- ***Energy inner product*** $(\cdot,\cdot)_E$ is one that
  - **<u>For linear system</u>**: leads to semi-boundedness of linear operator L w.r.t. inner product $(\cdot,\cdot)_E$, i.e.,

$$\left(\frac{d\boldsymbol{u}_M}{dt}, \phi\right)_E = (L\boldsymbol{u}_M, \phi)_E, \ \forall \phi \in \mathcal{H}$$

$\rightarrow$ leads to stability estimate $\frac{1}{2}\frac{d}{dt}||\boldsymbol{u}_M||_E^2 \leq \alpha\left|\left|\boldsymbol{u}_M\right|\right|_E^2$

  - **<u>For nonlinear system</u>**: induces norm that represents global energy quantity (e.g., kinetic energy, stagnation energy, total energy, etc.).

Energy-stability

- ***Energy inner product*** $(\cdot,\cdot)_E$ is one that
  - **<u>For linear system</u>**: leads to semi-boundedness of linear operator L w.r.t. inner product $(\cdot,\cdot)_E$, i.e.,

$$\left(\frac{d\boldsymbol{u}_M}{dt}, \phi\right)_E = (L\boldsymbol{u}_M, \phi)_E, \ \forall \phi \in \mathcal{H}$$

$\rightarrow$ leads to stability estimate $\frac{1}{2}\frac{d}{dt}||\boldsymbol{u}_M||_E^2 \leq \alpha\left|\left|\boldsymbol{u}_M\right|\right|_E^2$

  - **<u>For nonlinear system</u>**: induces norm that represents global energy quantity (e.g., kinetic energy, stagnation energy, total energy, etc.).

Practical implication of energy-stability analysis:
energy inner product ensures that any "bad" modes will not introduce spurious non-physical numerical instabilities into the Galerkin approximation.

Energy-stability

- ***Energy inner product*** $(\cdot,\cdot)_E$ is one that
  - **<u>For linear system</u>**: leads to semi-boundedness of linear operator L w.r.t. inner product $(\cdot,\cdot)_E$, i.e.,

$$\left(\frac{d\boldsymbol{u}_M}{dt}, \phi\right)_E = (L\boldsymbol{u}_M, \phi)_E, \ \forall \phi \in \mathcal{H}$$

$\rightarrow$ leads to stability estimate $\frac{1}{2}\frac{d}{dt}||\boldsymbol{u}_M||_E^2 \leq \alpha\left|\left|\boldsymbol{u}_M\right|\right|_E^2$

  - **<u>For nonlinear system</u>**: induces norm that represents global energy quantity (e.g., kinetic energy, stagnation energy, total energy, etc.).

Practical implication of energy-stability analysis:
energy inner product ensures that any "bad" modes will not introduce spurious non-physical numerical instabilities into the Galerkin approximation.

- Stability-preserving inner product derived using the **energy method**:
  - Defines numerical solution energy $E_M \equiv \frac{1}{2}||\boldsymbol{u}_M||_E{}^2$ and bounds it in a physical way.
  - Borrowed from spectral methods community.
  - Analysis is straightforward for ROMs constructed via **continuous projection.**

Outline

	Continuous Projection	Discrete Projection
Change projection (*a priori*)	**Energy inner products** (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	Eigenvalue reassignment (IKT *et al.*)
Change ROM basis (*a posteriori*)	Basis rotation (Balajewicz, IKT, *et al.*)	Optimization-based right basis modificatiOon (Amsallem)

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. **Approaches for building *a priori* stable ROMs.**
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

with: Matt Barone (SNL),
Srini Arunajatesan (SNL)

Linearized compressible flow equations

$\dot{\boldsymbol{x}}_N = \boldsymbol{A}_N \boldsymbol{x}_N$ **Energy-Stability for Linearized PDEs:** $\dot{\boldsymbol{x}}_M = \boldsymbol{A}_M \boldsymbol{x}_M$

FOM linearly stable ⇒ ROM built in energy inner product linearly stable ($Re(\lambda) < 0$)

(Barone *et al.* 2009, IKT *et al.* 2012)

Linearized compressible flow equations

$\dot{\boldsymbol{x}}_N = \boldsymbol{A}_N \boldsymbol{x}_N$ **Energy-Stability for Linearized PDEs:** $\dot{\boldsymbol{x}}_M = \boldsymbol{A}_M \boldsymbol{x}_M$

FOM linearly stable ⇒ ROM built in energy inner product linearly stable ($Re(\lambda) < 0$)

(Barone *et al.* 2009, IKT *et al.* 2012)

Linearized compressible Euler/Navier-Stokes equations are appropriate when a compressible fluid system can be described by small-amplitude perturbations about a steady-state mean flow.

Linearized compressible flow equations

$\dot{\boldsymbol{x}}_N = \boldsymbol{A}_N \boldsymbol{x}_N$ **Energy-Stability for Linearized PDEs:** $\dot{\boldsymbol{x}}_M = \boldsymbol{A}_M \boldsymbol{x}_M$

FOM linearly stable ⇒ ROM built in energy inner product linearly stable ($Re(\lambda) < 0$)

(Barone *et al.* 2009, IKT *et al.* 2012)

Full non-linear compressible flow equations

$$\rho_{,t} + (\rho u_j)_{,j} = 0$$
$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} + p_{,i} - \tau_{ij,j} = 0$$
$$(\rho e)_{,t} + (p u_i)_{,i} - (u_i \tau_{ij})_{,j} = 0$$

Linearized compressible flow equations

$\dot{\boldsymbol{x}}_N = \boldsymbol{A}_N \boldsymbol{x}_N$ **Energy-Stability for Linearized PDEs:** $\dot{\boldsymbol{x}}_M = \boldsymbol{A}_M \boldsymbol{x}_M$

FOM linearly stable ⇒ ROM built in energy inner product linearly stable ($Re(\lambda) < 0$)

(Barone *et al.* 2009, IKT *et al.* 2012)

Full non-linear compressible flow equations

$$\rho_{,t} + (\rho u_j)_{,j} = 0$$
$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} + p_{,i} - \tau_{ij,j} = 0$$
$$(\rho e)_{,t} + (p u_i)_{,i} - (u_i \tau_{ij})_{,j} = 0$$

- Linearization of full compressible Euler/Navier-Stokes equations obtained as follows:
  1. Decompose fluid field as **steady mean** plus **unsteady fluctuation**

$$\boldsymbol{q}(\boldsymbol{x}, t) = \overline{\boldsymbol{q}}(\boldsymbol{x}) + \boldsymbol{q}'(\boldsymbol{x}, t)$$

$\boldsymbol{q}$ = fluid solution vector
(e.g., $\boldsymbol{q}^T = (u_1, u_2, u_3, T, \rho)$)

  2. Linearize full nonlinear compressible Navier-Stokes equations around steady mean to yield **linear hyperbolic/incompletely parabolic** system

$$\dot{\boldsymbol{q}}' + \boldsymbol{A}_i(\overline{\boldsymbol{q}}) \frac{\partial \boldsymbol{q}'}{\partial x_i} - \frac{\partial}{\partial x_j}\left[\boldsymbol{K}_{ij}(\overline{\boldsymbol{q}}) \frac{\partial \boldsymbol{q}'}{\partial x_i}\right] = \boldsymbol{0}$$

Energy-stable ROMs for linearized compressible flow

Linearized compressible Euler/Navier-Stokes equations are **symmetrizable** (Barone & IKT, 2009; IKT & Arunajatesan, 2012).

Energy-stable ROMs for linearized compressible flow

Linearized compressible Euler/Navier-Stokes equations are **symmetrizable** (Barone & IKT, 2009; IKT & Arunajatesan, 2012).

- There exists a symmetric positive definite matrix $\boldsymbol{H} \equiv \boldsymbol{H}(\overline{\boldsymbol{q}})$ (system "**symmetrizer**") s.t.:
  - The convective flux matrices $\boldsymbol{HA}_i$ are symmetric
  - The following augmented viscosity matrix is symmetric positive semi-definite

$$\boldsymbol{K}^s = \begin{pmatrix} \boldsymbol{HK}_{11} & \boldsymbol{HK}_{12} & \boldsymbol{HK}_{13} \\ \boldsymbol{HK}_{21} & \boldsymbol{HK}_{22} & \boldsymbol{HK}_{23} \\ \boldsymbol{HK}_{31} & \boldsymbol{HK}_{32} & \boldsymbol{HK}_{33} \end{pmatrix}$$

Energy-stable ROMs for linearized compressible flow

Linearized compressible Euler/Navier-Stokes equations are **symmetrizable** (Barone & IKT, 2009; IKT & Arunajatesan, 2012).

- There exists a symmetric positive definite matrix $\boldsymbol{H} \equiv \boldsymbol{H}(\overline{\boldsymbol{q}})$ (system "**symmetrizer**") s.t.:
  - The convective flux matrices $\boldsymbol{HA}_i$ are symmetric
  - The following augmented viscosity matrix is symmetric positive semi-definite

$$\boldsymbol{K}^s = \begin{pmatrix} \boldsymbol{HK}_{11} & \boldsymbol{HK}_{12} & \boldsymbol{HK}_{13} \\ \boldsymbol{HK}_{21} & \boldsymbol{HK}_{22} & \boldsymbol{HK}_{23} \\ \boldsymbol{HK}_{31} & \boldsymbol{HK}_{32} & \boldsymbol{HK}_{33} \end{pmatrix}$$

Symmetry Inner Product (weighted L^2 inner product):

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{\boldsymbol{H}} = \int_{\Omega} \boldsymbol{q}_1 \boldsymbol{H} \boldsymbol{q}_2 d\Omega$$

Energy-stable ROMs for linearized compressible flow

Linearized compressible Euler/Navier-Stokes equations are **symmetrizable** (Barone & IKT, 2009; IKT & Arunajatesan, 2012).

- There exists a symmetric positive definite matrix $\boldsymbol{H} \equiv \boldsymbol{H}(\overline{\boldsymbol{q}})$ (system "**symmetrizer**") s.t.:
  - The convective flux matrices $\boldsymbol{HA}_i$ are symmetric
  - The following augmented viscosity matrix is symmetric positive semi-definite

$$\boldsymbol{K}^s = \begin{pmatrix} \boldsymbol{HK}_{11} & \boldsymbol{HK}_{12} & \boldsymbol{HK}_{13} \\ \boldsymbol{HK}_{21} & \boldsymbol{HK}_{22} & \boldsymbol{HK}_{23} \\ \boldsymbol{HK}_{31} & \boldsymbol{HK}_{32} & \boldsymbol{HK}_{33} \end{pmatrix}$$

Symmetry Inner Product (weighted L^2 inner product):

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{\boldsymbol{H}} = \int_{\Omega} \boldsymbol{q}_1 \boldsymbol{H} \boldsymbol{q}_2 d\Omega$$

- If ROM is built in **symmetry inner product**, Galerkin approximation will satisfy the same energy expression as continuous PDEs, e.g., it will be **energy-stable**:

$$\frac{1}{2}\frac{d}{dt}||\boldsymbol{q}'_M||_{\boldsymbol{H}}^2 \leq \alpha \left|\left|\boldsymbol{q}'_M\right|\right|_{\boldsymbol{H}}^2$$

Symmetrizers for several hyperbolic/ incompletely parabolic systems

- ***Wave equation:*** $\ddot{u} = a^2 \frac{\partial^2 u}{\partial x^2}$ or $\dot{\boldsymbol{q}} = \boldsymbol{A}\,\frac{\partial \boldsymbol{q}}{\partial x}$ where $\boldsymbol{q} = \left(\dot{u}, \frac{\partial u}{\partial x}\right)$

$$\Rightarrow \boldsymbol{H} = \begin{pmatrix} 1 & 0 \\ 0 & a^2 \end{pmatrix}$$

- ***Linearized shallow water equations:*** $\dot{\boldsymbol{q}}' + \boldsymbol{A}_i(\overline{\boldsymbol{q}})\frac{\partial \boldsymbol{q}'}{\partial xi} = \boldsymbol{0}$

$$\Rightarrow \boldsymbol{H} = \begin{pmatrix} \bar{\phi} & 0 & 0 \\ 0 & \bar{\phi} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- ***Linearized compressible Euler:*** $\dot{\boldsymbol{q}}' + \boldsymbol{A}_i(\overline{\boldsymbol{q}})\frac{\partial \boldsymbol{q}'}{\partial xi} = \boldsymbol{0}$

$$\Rightarrow \boldsymbol{H} = \begin{pmatrix} \bar{\rho} & 0 & 0 \\ 0 & \alpha^2\gamma\bar{\rho}^2\bar{p} & \bar{\rho}\alpha^2 \\ 0 & 0 & \frac{(1+\alpha^2)}{\gamma\bar{p}} \end{pmatrix}$$

- ***Linearized compressible Navier-Stokes:*** $\dot{\boldsymbol{q}}' + \boldsymbol{A}_i(\overline{\boldsymbol{q}})\frac{\partial \boldsymbol{q}'}{\partial xi} + \frac{\partial}{\partial xj}\left[\boldsymbol{K}_{ij}(\overline{\boldsymbol{q}})\frac{\partial \boldsymbol{q}'}{\partial xi}\right] = \boldsymbol{0}$

$$\Rightarrow \boldsymbol{H} = \begin{pmatrix} \bar{\rho} & 0 & 0 \\ 0 & \frac{\bar{\rho}R}{\bar{T}(\gamma-1)} & 0 \\ 0 & 0 & \frac{R\bar{T}}{\bar{\rho}} \end{pmatrix}$$

- Barone & IKT, *JCP*, 2009.
- IKT & Arunajatesan, *WCCM X*, 2012.
- IKT *et al.*, *SAND report*, 2014.

Discrete counterpart of symmetry-inner product: Lyapunov inner product

- Consider linear discrete (i.e., discretized in space) stable full order model

$$\boxed{\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}}$$

- ***Lyapunov function*** for this FOM: $V(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{P}\boldsymbol{x}$ where $\boldsymbol{P}$ is the solution of the Lyapunov equation

$$\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} = -\boldsymbol{Q}$$

  - S.p.d. solution to Lyapunov equation exists if $\boldsymbol{Q}$ is s.p.d. and $\boldsymbol{A}$ is stable.
  - There are solvers for solving the Lyapunov equation, e.g., lyap function in MATLAB control toolbox.

- Discrete analog of symmetry inner product: ***Lyapunov inner product*** (Rowley *et al.*, 2004):

$$(\boldsymbol{x}_1, \boldsymbol{x}_2)_{\boldsymbol{P}} \equiv \boldsymbol{x}_1^T \boldsymbol{P}\boldsymbol{x}_2$$

- **Can show**: if ROM is constructed in Lyapunov inner product, it is energy-stable, i.e.,

$$\frac{dE_N}{dt} \equiv \frac{1}{2}\frac{d}{dt}||\boldsymbol{x}_N||_2^2 \leq 0$$

Discrete counterpart of symmetry-inner product: Lyapunov inner product

- Consider linear discrete (i.e., discretized in space) stable full order model

$$\boxed{\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}}$$

- ***Lyapunov function*** for this FOM: $V(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{P}\boldsymbol{x}$ where $\boldsymbol{P}$ is the solution of the Lyapunov equation

$$\boxed{\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} = -\boldsymbol{Q}}$$

Intractable for large problems ($O(N^3)$ ops!

  - S.p.d. solution to Lyapunov equation exists if $\boldsymbol{Q}$ is s.p.d. and $\boldsymbol{A}$ is stable.
  - There are solvers for solving the Lyapunov equation, e.g., lyap function in MATLAB control toolbox.

- Discrete analog of symmetry inner product: ***Lyapunov inner product*** (Rowley *et al.*, 2004):

$$(\boldsymbol{x}_1, \boldsymbol{x}_2)_{\boldsymbol{P}} \equiv \boldsymbol{x}_1^T \boldsymbol{P}\boldsymbol{x}_2$$

- **Can show**: if ROM is constructed in Lyapunov inner product, it is energy-stable, i.e.,

$$\frac{dE_N}{dt} \equiv \frac{1}{2}\frac{d}{dt}||\boldsymbol{x}_N||_2^2 \leq 0$$

Steps to obtain stable ROM in symmetry i.p.

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2010; **83**:1345–1375
Published online 30 March 2010 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2867

On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment‡

I. Kalashnikova[1,2,*,†] and M. F. Barone[3]

- ***Galerkin-project*** equations using symmetry inner product:

$$\left(\boldsymbol{\phi}_k, \frac{\partial \boldsymbol{q}_M{}'}{\partial \boldsymbol{t}}\right)_{\boldsymbol{H}} = -\int_{\Omega} \boldsymbol{\phi}_k \boldsymbol{H}\left[\boldsymbol{A}_i \frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} - \frac{\partial}{\partial x_j}\left[\boldsymbol{K}_{ij} \frac{\partial \boldsymbol{q}_M{}'}{\partial x_i}\right]\right] d\Omega$$

Steps to obtain stable ROM in symmetry i.p.

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2010; **83**:1345–1375
Published online 30 March 2010 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2867

On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment‡

I. Kalashnikova[1,2,*,†] and M. F. Barone[3]

Remark: Galerkin projection in symmetry inner product is equivalent to ***Petrov-Galerkin*** projection with $\boldsymbol{\Psi}_M = \boldsymbol{H}\boldsymbol{\Phi}_M$.

- ***Galerkin-project*** equations using symmetry inner product:

$$\left(\boldsymbol{\phi}_k, \frac{\partial \boldsymbol{q}_M{}'}{\partial \boldsymbol{t}}\right)_{\boldsymbol{H}} = -\int_\Omega \boldsymbol{\phi}_k \boldsymbol{H}\left[\boldsymbol{A}_i \frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} - \frac{\partial}{\partial x_j}\left[\boldsymbol{K}_{ij}\frac{\partial \boldsymbol{q}_M{}'}{\partial x_i}\right]\right] d\Omega$$

Steps to obtain stable ROM in symmetry i.p.

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2010; **83**:1345–1375
Published online 30 March 2010 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2867

On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment‡

I. Kalashnikova[1,2,*,†] and M. F. Barone[3]

Remark: Galerkin projection in symmetry inner product is equivalent to ***Petrov-Galerkin*** projection with $\boldsymbol{\Psi}_M = \boldsymbol{H}\boldsymbol{\Phi}_M$.

- ***Galerkin-project*** equations using symmetry inner product:

$$\left(\boldsymbol{\phi}_k, \frac{\partial \boldsymbol{q}_M{}'}{\partial \boldsymbol{t}}\right)_{\boldsymbol{H}} = -\int_\Omega \boldsymbol{\phi}_k \boldsymbol{H}\left[\boldsymbol{A}_i \frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} - \frac{\partial}{\partial x_j}\left[\boldsymbol{K}_{ij}\frac{\partial \boldsymbol{q}_M{}'}{\partial x_i}\right]\right] d\Omega$$

Substitute BCs

- Integrate second term by parts and apply ***boundary conditions*** weakly:

$$\left(\boldsymbol{\phi}_k, \frac{\partial \boldsymbol{q}_M{}'}{\partial \boldsymbol{t}}\right)_{\boldsymbol{H}} = -\int_\Omega \left[\boldsymbol{\phi}_k \boldsymbol{H}\boldsymbol{A}_i \frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} + \frac{\partial \boldsymbol{\phi}_k}{\partial x_j}\boldsymbol{H}\boldsymbol{K}_{ij}\frac{\partial \boldsymbol{q}_M{}'}{\partial x_i}\right] d\Omega + \int_{\partial\Omega} \boldsymbol{\phi}_k \boldsymbol{H}\,\boldsymbol{K}_{ij}\frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} n_j dS$$

Steps to obtain stable ROM in symmetry i.p.

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2010; **83**:1345–1375
Published online 30 March 2010 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2867

On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment‡

I. Kalashnikova[1,2,*,†] and M. F. Barone[3]

Remark: Galerkin projection in symmetry inner product is equivalent to ***Petrov-Galerkin*** projection with $\boldsymbol{\Psi}_M = \boldsymbol{H}\boldsymbol{\Phi}_M$.

- ***Galerkin-project*** equations using symmetry inner product:

$$\left(\boldsymbol{\phi}_k, \frac{\partial \boldsymbol{q}_M{}'}{\partial \boldsymbol{t}}\right)_{\boldsymbol{H}} = -\int_\Omega \boldsymbol{\phi}_k \boldsymbol{H}\left[\boldsymbol{A}_i \frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} - \frac{\partial}{\partial x_j}\left[\boldsymbol{K}_{ij}\frac{\partial \boldsymbol{q}_M{}'}{\partial x_i}\right]\right] d\Omega$$

Substitute BCs

- Integrate second term by parts and apply ***boundary conditions*** weakly:

$$\left(\boldsymbol{\phi}_k, \frac{\partial \boldsymbol{q}_M{}'}{\partial \boldsymbol{t}}\right)_{\boldsymbol{H}} = -\int_\Omega \left[\boldsymbol{\phi}_k \boldsymbol{H}\boldsymbol{A}_i \frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} + \frac{\partial \boldsymbol{\phi}_k}{\partial x_j}\boldsymbol{H}\boldsymbol{K}_{ij}\frac{\partial \boldsymbol{q}_M{}'}{\partial x_i}\right] d\Omega + \int_{\partial\Omega} \boldsymbol{\phi}_k \boldsymbol{H}\,\boldsymbol{K}_{ij}\frac{\partial \boldsymbol{q}_M{}'}{\partial x_i} n_j dS$$

- Substitute ***modal decomposition*** $\boldsymbol{q}'_M(\boldsymbol{x},t) = \sum_{i=1}^{M} a_{M,i}(t)\boldsymbol{\phi}_{\boldsymbol{i}}(\boldsymbol{x})$to obtain an $M \times M$ linear dynamical system of the form $\dot{\boldsymbol{x}}_M = \boldsymbol{A}\boldsymbol{x}_M$.

Continuous projection implementation: "Spirit" code

"Spirit" ROM Code = 3D parallel C++ POD/Galerkin test-bed ROM code that uses data-structures and eigensolvers from **Trilinos** to build energy-stable ROMs for compressible flow problems
→ stand-alone code that can be synchronized with any high-fidelity code!

- POD modes defined using piecewise smooth finite elements.
- Gauss quadrature rules of sufficient accuracy are used to compute exactly inner products with the help of the `libmesh` library.
- Physics in Spirit:
  - ***Linearized compressible Euler*** (L^2, energy inner product).
  - ***Linearized compressible Navier-Stokes*** (L^2, energy inner product).
  - ***Nonlinear isentropic compressible Navier-Stokes*** (L^2, stagnation energy, stagnation enthalpy inner product).
  - ***Nonlinear compressible Navier-Stokes*** (L^2, energy inner product).

"SIGMA CFD" High-Fidelity Code = Sandia in-house finite volume flow solver derived from LESLIE3D (Genin & Menon, 2010), an LES flow solver originally developed in the Computational Combustion Laboratory at Georgia Tech.

Continuous projection implementation: "Spirit" code

"Spirit" ROM Code = 3D parallel C++ POD/Galerkin test-bed ROM code that uses data-structures and eigensolvers from **Trilinos** to build energy-stable ROMs for compressible flow problems
→ stand-alone code that can be synchronized with any high-fidelity code!

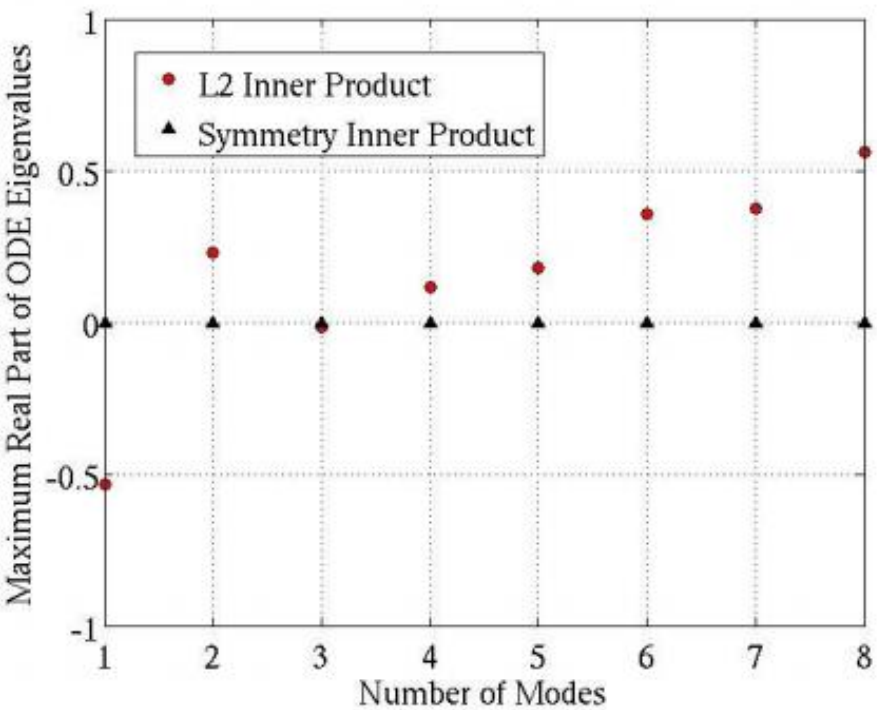
- POD modes defined using piecewise smooth finite elements.
- Gauss quadrature rules of sufficient accuracy are used to compute exactly inner products with the help of the `libmesh` library.
- Physics in Spirit:
  - ***Linearized compressible Euler*** (L^2, energy inner product).
  - ***Linearized compressible Navier-Stokes*** (L^2, energy inner product).
  - ***Nonlinear isentropic compressible Navier-Stokes*** (L^2, stagnation energy, stagnation enthalpy inner product).
  - ***Nonlinear compressible Navier-Stokes*** (L^2, energy inner product).

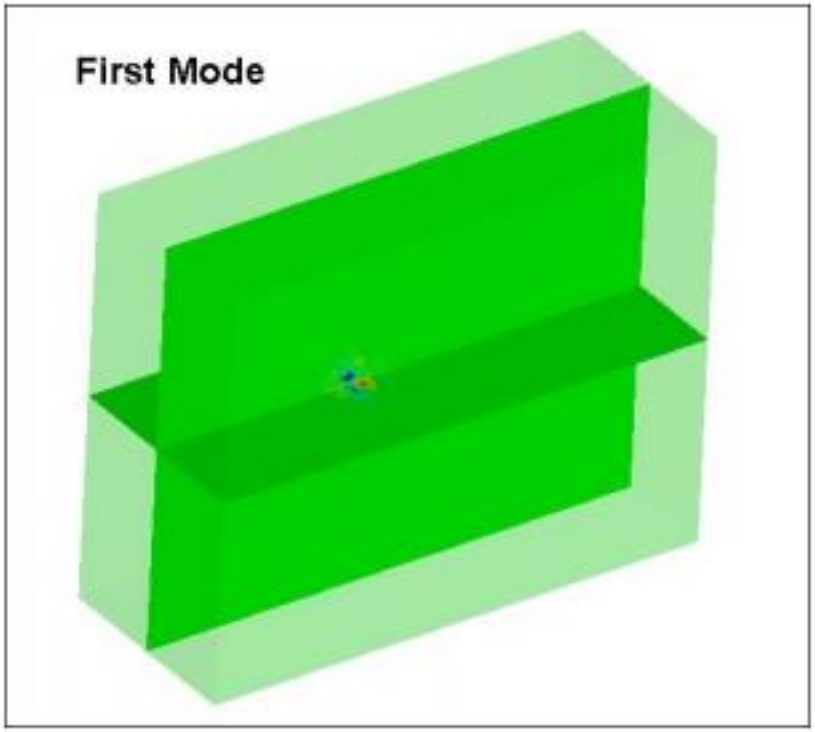
First, testing of ROMs for these physics

"SIGMA CFD" High-Fidelity Code = Sandia in-house finite volume flow solver derived from LESLIE3D (Genin & Menon, 2010), an LES flow solver originally developed in the Computational Combustion Laboratory at Georgia Tech.

Numerical results: random basis example

- ***Uniform base flow***: physically stable to any linear disturbance.
- Each mode is ***random disturbance*** field that decays to 0 at domain boundaries.
- Model problem for modes determined by numerical error: extreme case of ***"bad" modes***.

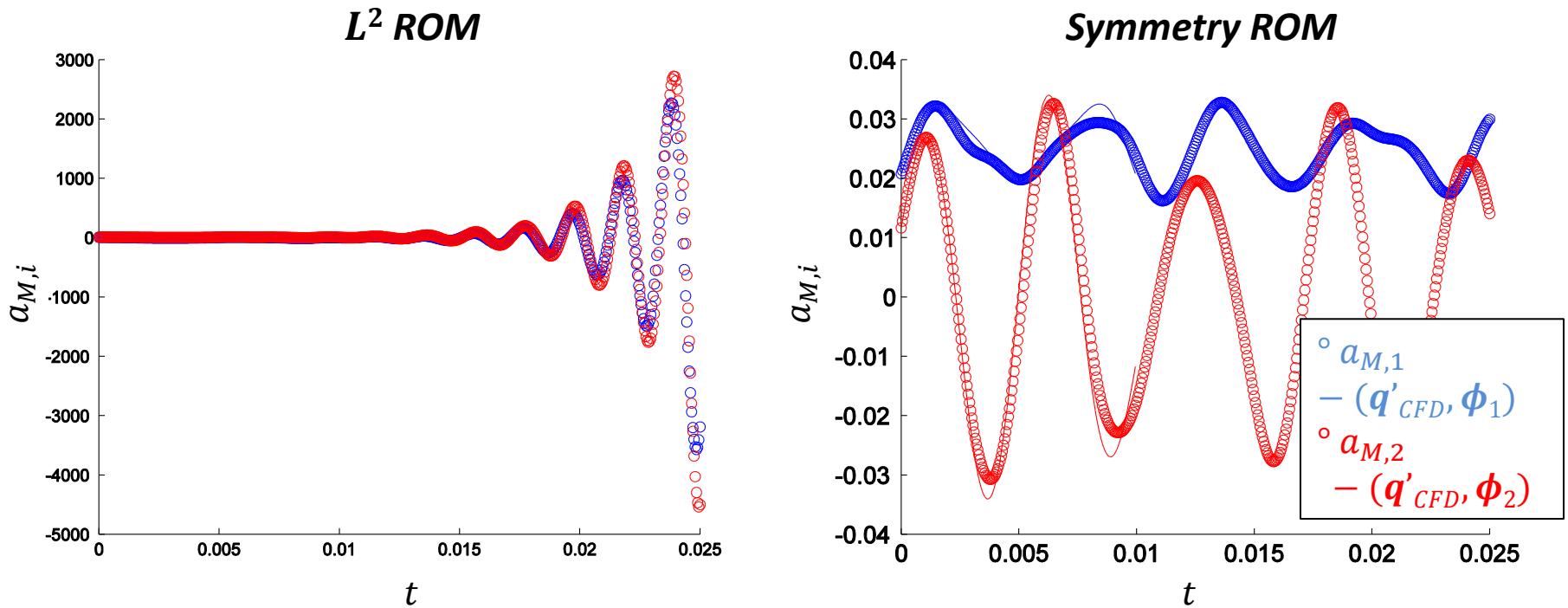




Numerical results: 2D inviscid pressure pulse

- Inviscid pulse in a uniform base flow.
- High-fidelity simulation run on mesh with 3362 nodes, up to time $t = 0.01$ seconds.
- 200 snapshots of solution used to construct $M = 20$ mode ROM in L^2 and symmetry inner products.

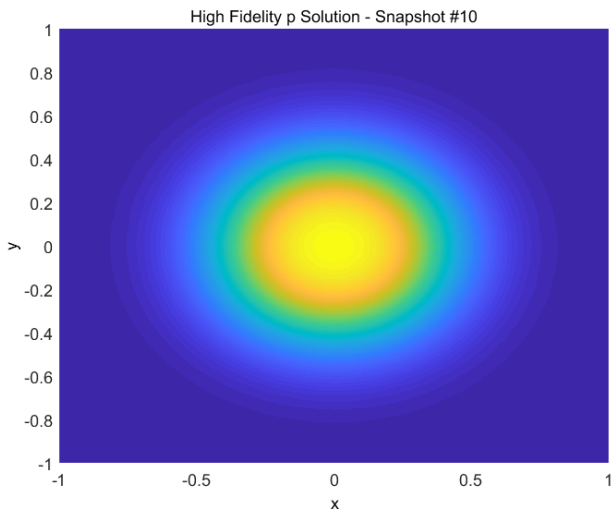
$a_{M,i}(t)$ vs.$(\boldsymbol{q}'_{CFD}, \boldsymbol{\phi}_i)$ for $i = 1,2$



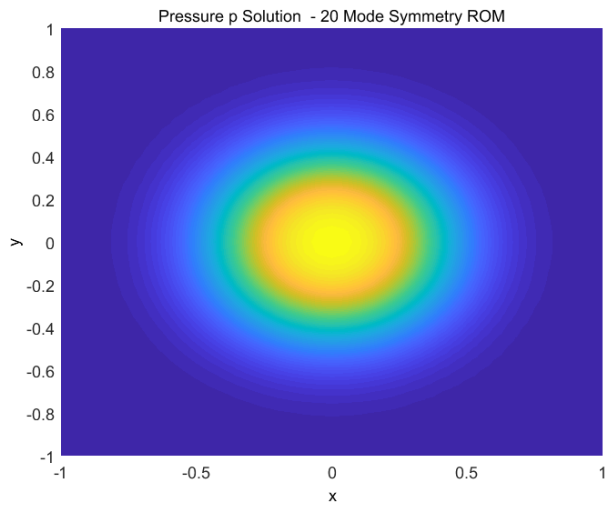
Numerical results: 2D inviscid pressure pulse

- Inviscid pulse in a uniform base flow (linear dynamics).
- High-fidelity simulation run on mesh with 3362 nodes, up to time $t = 0.01$ seconds.
- 200 snapshots of solution used to construct $M = 20$ mode ROM in L^2 and symmetry inner products.

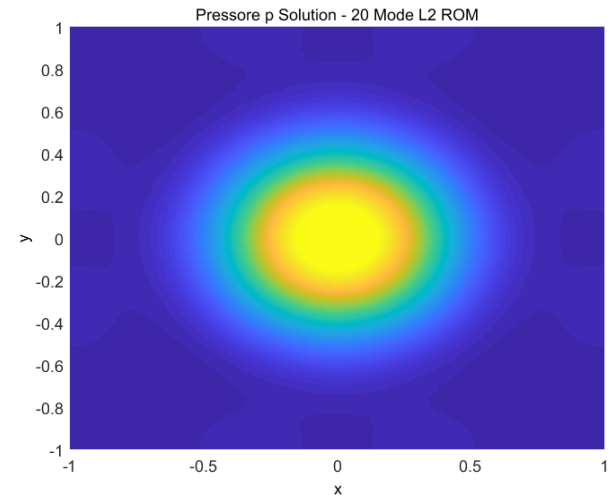
p': High-fidelity



p': Symmetry ROM



p': L^2 ROM

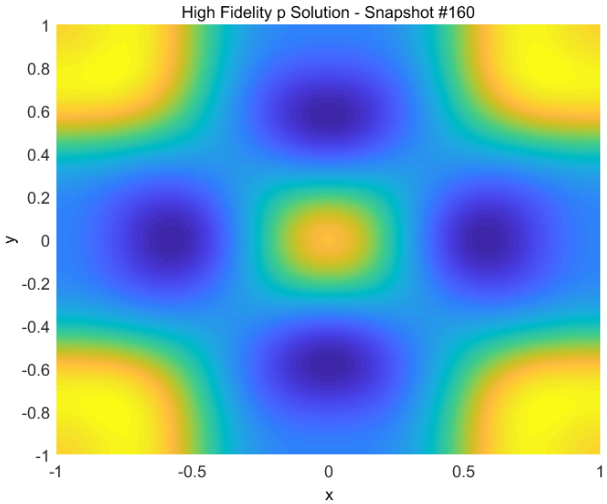


time of snapshot 10

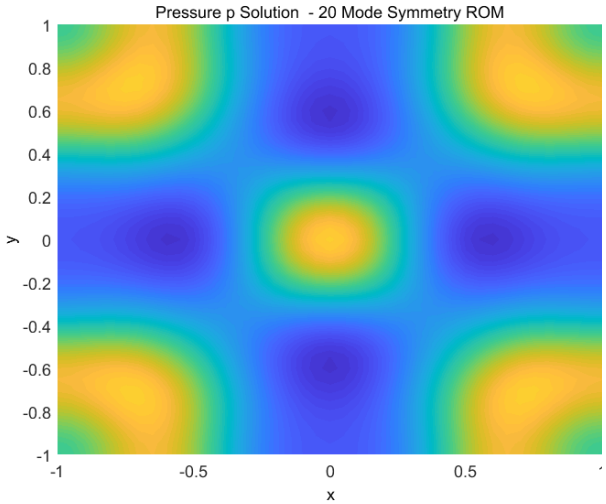
Numerical results: 2D inviscid pressure pulse

- Inviscid pulse in a uniform base flow (linear dynamics).
- High-fidelity simulation run on mesh with 3362 nodes, up to time $t = 0.01$ seconds.
- 200 snapshots of solution used to construct $M = 20$ mode ROM in L^2 and symmetry inner products.

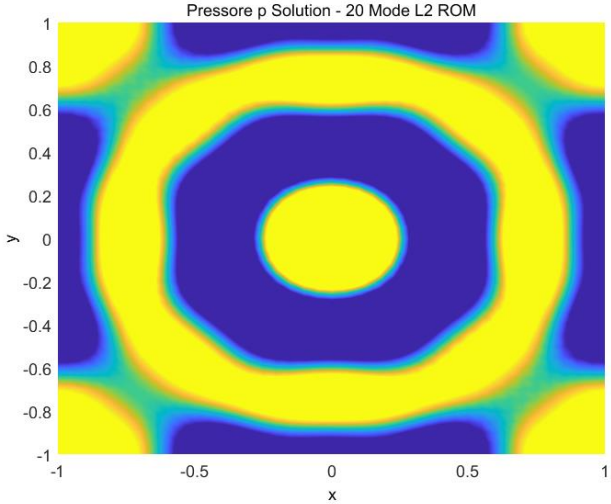
p': High-fidelity



p': Symmetry ROM



p': L^2 ROM



time of snapshot 160

Outline

	Continuous Projection	Discrete Projection
Change projection (*a priori*)	**Energy inner products** (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	Eigenvalue reassignment (IKT *et al.*)
Change ROM basis (*a posteriori*)	Basis rotation (Balajewicz, IKT, *et al.*)	Optimization-based right basis modificatiOon (Amsallem)

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. **Approaches for building *a priori* stable ROMs.**
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

with: Matt Barone (SNL), Jeff Fike (SNL), Srini Arunajatesan (SNL)

Nonlinear compressible flow equations

Energy-Stability for Nonlinear PDEs:
ROM built in energy inner product will preserve stability of an equilibrium point at 0 for the governing nonlinear system of PDEs (Rowley *et al.*, 2004; IKT *et al.,* 2014).

- Compressible isentropic Navier-Stokes equations (cold flows, moderate Mach #):

$$\frac{Dh}{Dt} + (\gamma - 1)h\nabla \cdot \boldsymbol{u} = 0$$
$$\frac{D\boldsymbol{u}}{Dt} + \nabla h - \frac{1}{Re}\Delta \boldsymbol{u} = \boldsymbol{0}$$

h = enthalpy
$\boldsymbol{u}$ = velocity vector
ρ = density
T = temperature
$\boldsymbol{\tau}$ = viscous stress tensor

- Full compressible Navier-Stokes equations:

$$\rho\frac{D\boldsymbol{u}}{Dt} + \frac{1}{\gamma M^2}\nabla(\rho T) - \frac{1}{Re}\nabla \cdot \boldsymbol{\tau} = \boldsymbol{0}$$
$$\frac{D\rho}{Dt} + \rho\nabla \cdot \boldsymbol{u} = 0$$
$$\rho\frac{DT}{Dt} + (\gamma - 1)\rho T\nabla \cdot \boldsymbol{u} - \frac{\gamma}{PrRe}\nabla \cdot (\kappa\nabla T) - \left(\frac{\gamma(\gamma - 1)M^2}{Re}\right)\nabla\boldsymbol{u} \cdot \boldsymbol{\tau} = 0$$

Energy-stable ROMs for nonlinear compressible flow: isentropic NS (Rowley *et al.*, 2004)

In (Rowley *et al.*, 2004), Rowley *et al.* showed that energy inner product for the compressible isentropic Navier-Stokes equations can be defined following a transformation of these equations.

Energy-stable ROMs for nonlinear compressible flow: isentropic NS (Rowley *et al.*, 2004)

In (Rowley *et al.*, 2004), Rowley *et al.* showed that energy inner product for the compressible isentropic Navier-Stokes equations can be defined following a transformation of these equations.

- Transformed compressible isentropic Navier-Stokes equations:

$$\frac{Dc}{Dt} + \frac{\gamma - 1}{2} c \nabla \cdot \boldsymbol{u} = 0$$

$$\frac{D\boldsymbol{u}}{Dt} + \frac{2}{\gamma - 1} c \nabla c - \frac{1}{Re} \Delta \boldsymbol{u} = \boldsymbol{0}$$

c = speed of sound ($c^2 = (\gamma - 1)h$)
$\boldsymbol{u}$ = velocity

Energy-stable ROMs for nonlinear compressible flow: isentropic NS (Rowley *et al.*, 2004)

In (Rowley *et al.*, 2004), Rowley *et al.* showed that energy inner product for the compressible isentropic Navier-Stokes equations can be defined following a transformation of these equations.

- Transformed compressible isentropic Navier-Stokes equations:

$$\frac{Dc}{Dt} + \frac{\gamma - 1}{2} c \nabla \cdot \boldsymbol{u} = 0$$

$$\frac{D\boldsymbol{u}}{Dt} + \frac{2}{\gamma - 1} c \nabla c - \frac{1}{Re} \Delta \boldsymbol{u} = \boldsymbol{0}$$

c = speed of sound ($c^2 = (\gamma - 1)h$)
$\boldsymbol{u}$ = velocity

- Family of inner products:

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_\alpha = \int_\Omega \left(\boldsymbol{u}_1 \cdot \boldsymbol{u}_2 + \frac{2\alpha}{\gamma - 1} c_1 c_2 \right) d\Omega$$

$$\alpha = \begin{cases} 1 \Rightarrow \|\boldsymbol{q}\|_\alpha = \text{stagnation enthalpy} \\ \frac{1}{\gamma} \Rightarrow \|\boldsymbol{q}\|_\alpha = \text{stagnation energy} \end{cases}$$

Energy-stable ROMs for nonlinear compressible flow: isentropic NS (Rowley *et al.*, 2004)

In (Rowley *et al.*, 2004), Rowley *et al.* showed that energy inner product for the compressible isentropic Navier-Stokes equations can be defined following a transformation of these equations.

- Transformed compressible isentropic Navier-Stokes equations:

$$\frac{Dc}{Dt} + \frac{\gamma - 1}{2} c \nabla \cdot \boldsymbol{u} = 0$$

$$\frac{D\boldsymbol{u}}{Dt} + \frac{2}{\gamma - 1} c \nabla c - \frac{1}{Re} \Delta \boldsymbol{u} = \boldsymbol{0}$$

c = speed of sound ($c^2 = (\gamma - 1)h$)
$\boldsymbol{u}$ = velocity

- Family of inner products:

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_\alpha = \int_\Omega \left(\boldsymbol{u}_1 \cdot \boldsymbol{u}_2 + \frac{2\alpha}{\gamma - 1} c_1 c_2 \right) d\Omega$$

$$\alpha = \begin{cases} 1 \Rightarrow \|\boldsymbol{q}\|_\alpha = \text{stagnation enthalpy} \\ \frac{1}{\gamma} \Rightarrow \|\boldsymbol{q}\|_\alpha = \text{stagnation energy} \end{cases}$$

If Galerkin projection step of model reduction is performed in α inner product, then the Galerkin projection will **preserve the stability of an equilibrium point at the origin** (Rowley *et al.*, 2004).

Energy-stable ROMs for nonlinear compressible flow: full NS

Our work extends ideas in (Rowley *et al.*, 2004) to **full compressible N-S equations**.
Requirement: *transformation/inner product yields PDEs with only polynomial non-linearities.*

Energy-stable ROMs for nonlinear compressible flow: full NS

Our work extends ideas in (Rowley *et al.*, 2004) to **full compressible N-S equations**.
Requirement: *transformation/inner product yields PDEs with only polynomial non-linearities.*

- First, full compressible Navier-Stokes equations are **transformed** into the following variables:

$$a = \sqrt{\rho}, \quad \boldsymbol{b} = a\boldsymbol{u}, \quad d = ae$$

e =internal energy

Energy-stable ROMs for nonlinear compressible flow: full NS

Our work extends ideas in (Rowley *et al.*, 2004) to **full compressible N-S equations**.
Requirement: *transformation/inner product yields PDEs with only polynomial non-linearities.*

- First, full compressible Navier-Stokes equations are **transformed** into the following variables:

$$a = \sqrt{\rho}, \quad \boldsymbol{b} = a\boldsymbol{u}, \quad d = ae$$

e =internal energy

- Next, the following **"total energy"** inner product is defined:

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{TE} = \int_\Omega (\boldsymbol{b}_1 \cdot \boldsymbol{b}_2 + a_1 d_2 + a_2 d_1) d\Omega$$

→ Norm induced by total energy inner product is the total energy of the fluid system:

$$||\boldsymbol{q}||_{TE} = \int_\Omega \left(\rho e + \frac{1}{2}\rho u_i u_i\right) d\Omega$$

Energy-stable ROMs for nonlinear compressible flow: full NS

Our work extends ideas in (Rowley *et al.*, 2004) to **full compressible N-S equations**.
Requirement: *transformation/inner product yields PDEs with only polynomial non-linearities.*

- First, full compressible Navier-Stokes equations are **transformed** into the following variables:

$$a = \sqrt{\rho}, \quad \boldsymbol{b} = a\boldsymbol{u}, \quad d = ae$$

e =internal energy

- Next, the following **"total energy"** inner product is defined:

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{TE} = \int_\Omega (\boldsymbol{b}_1 \cdot \boldsymbol{b}_2 + a_1 d_2 + a_2 d_1) d\Omega$$

→ Norm induced by total energy inner product is the total energy of the fluid system:

$$||\boldsymbol{q}||_{TE} = \int_\Omega \left(\rho e + \frac{1}{2}\rho u_i u_i\right) d\Omega$$

If Galerkin projection step of model reduction is performed in total energy inner product, then the Galerkin projection will **preserve the stability of an equilibrium point at the origin** (IKT *et al.*, 2014)

Energy-stable ROMs for nonlinear compressible flow: full NS

Our work extends ideas in (Rowley *et al.*, 2004) to **full compressible N-S equations**.
Requirement: *transformation/inner product yields PDEs with only polynomial non-linearities.*

- First, full compressible Navier-Stokes equations are **transformed** into the following variables:

$$a = \sqrt{\rho}, \quad \boldsymbol{b} = a\boldsymbol{u}, \quad d = ae$$

e =internal energy

- Next, the following **"total energy"** inner product is defined:

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{TE} = \int_\Omega (\boldsymbol{b}_1 \cdot \boldsymbol{b}_2 + a_1 d_2 + a_2 d_1) d\Omega$$

→ Norm induced by total energy inner product is the total energy of the fluid system:

$$||\boldsymbol{q}||_{TE} = \int_\Omega \left(\rho e + \frac{1}{2}\rho u_i u_i\right) d\Omega$$

If Galerkin projection step of model reduction is performed in total energy inner product, then the Galerkin projection will **preserve the stability of an equilibrium point at the origin** (IKT *et al.*, 2014)

☺ Transformed equations have only **polynomial non-linearities** (projection of which can be computed in offline stage of MOR and stored).

Energy-stable ROMs for nonlinear compressible flow: full NS

Our work extends ideas in (Rowley *et al.*, 2004) to **full compressible N-S equations**.
Requirement: *transformation/inner product yields PDEs with only polynomial non-linearities.*

- First, full compressible Navier-Stokes equations are **transformed** into the following variables:

$$a = \sqrt{\rho}, \quad \boldsymbol{b} = a\boldsymbol{u}, \quad d = ae$$

e =internal energy

- Next, the following **"total energy"** inner product is defined:

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{TE} = \int_\Omega (\boldsymbol{b}_1 \cdot \boldsymbol{b}_2 + a_1 d_2 + a_2 d_1) d\Omega$$

→ Norm induced by total energy inner product is the total energy of the fluid system:

$$||\boldsymbol{q}||_{TE} = \int_\Omega \left(\rho e + \frac{1}{2}\rho u_i u_i\right) d\Omega$$

If Galerkin projection step of model reduction is performed in total energy inner product, then the Galerkin projection will **preserve the stability of an equilibrium point at the origin** (IKT *et al.*, 2014)

☺ Transformed equations have only **polynomial non-linearities** (projection of which can be computed in offline stage of MOR and stored).

☹ Transformation introduces **higher order polynomial** non-linearities for viscous case.

Energy-stable ROMs for nonlinear compressible flow: full NS

Our work extends ideas in (Rowley *et al.*, 2004) to **full compressible N-S equations**.
Requirement: *transformation/inner product yields PDEs with only polynomial non-linearities.*

- First, full compressible Navier-Stokes equations are **transformed** into the following variables:

$$a = \sqrt{\rho}, \quad \boldsymbol{b} = a\boldsymbol{u}, \quad d = ae$$

e =internal energy

- Next, the following **"total energy"** inner product is defined:

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{TE} = \int_\Omega (\boldsymbol{b}_1 \cdot \boldsymbol{b}_2 + a_1 d_2 + a_2 d_1) d\Omega$$

→ Norm induced by total energy inner product is the total energy of the fluid system:

$$||\boldsymbol{q}||_{TE} = \int_\Omega \left(\rho e + \frac{1}{2}\rho u_i u_i\right) d\Omega$$

If Galerkin projection step of model reduction is performed in total energy inner product, then the Galerkin projection will **preserve the stability of an equilibrium point at the origin** (IKT *et al.*, 2014)

☺ Transformed equations have only **polynomial non-linearities** (projection of which can be computed in offline stage of MOR and stored).

☹ Transformation introduces **higher order polynomial** non-linearities for viscous case.

☺ Efficiency of online stage of MOR can be recovered using **hyper-reduction** (e.g., DEIM, gappy POD).

Continuous projection Implementation: "Spirit" Code

"Spirit" ROM Code = 3D parallel C++ POD/Galerkin test-bed ROM code that uses data-structures and eigensolvers from **Trilinos** to build energy-stable ROMs for compressible flow problems → stand-alone code that can be synchronized with any high-fidelity code!

- POD modes defined using piecewise smooth finite elements.
- Gauss quadrature rules of sufficient accuracy are used to compute exactly inner products with the help of the libmesh library.
- Physics in spirit:
  - ***Linearized compressible Euler*** (L^2, energy inner product).
  - ***Linearized compressible Navier-Stokes*** (L^2, energy inner product).
  - ***Nonlinear isentropic compressible Navier-Stokes*** (L^2, stagnation energy, stagnation enthalpy inner product).
  - ***Nonlinear compressible Navier-Stokes*** (L^2, energy inner product).

Now, testing of ROMs for these physics

"SIGMA CFD" High-Fidelity Code = Sandia in-house finite volume flow solver derived from LESLIE3D (Genin & Menon, 2010), a LES flow solver originally developed in the Computational Combustion Laboratory at Georgia Tech.

Numerical results: viscous laminar cavity

- Viscous cavity problem at $M = 0.6$, $Re = 1500$ (laminar regime).
- **High-fidelity simulation**: DNS based on full nonlinear compressible Navier-Stokes equations with $99{,}408$ nodes (right).
- 500 snapshots collected, every $\Delta t_{snap} = 1 \times 10^{-4}$ seconds.
- Snapshots used to construct $M = 5$ mode ROM for nonlinear compressible Navier-Stokes equations in $\boldsymbol{L^2}$ **and total energy inner products**.
- $M = 5$ mode POD bases capture $\approx 95\%$ of snapshot energy.

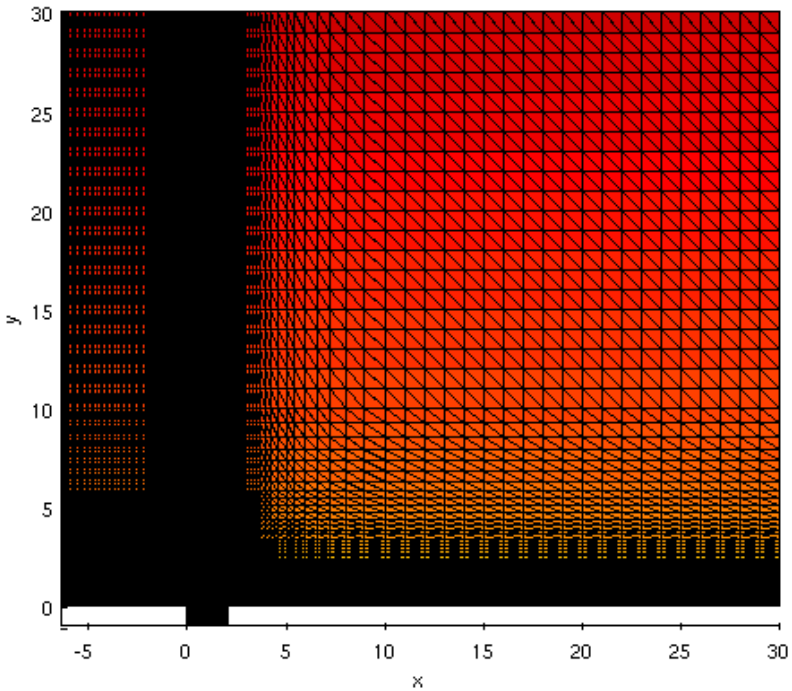
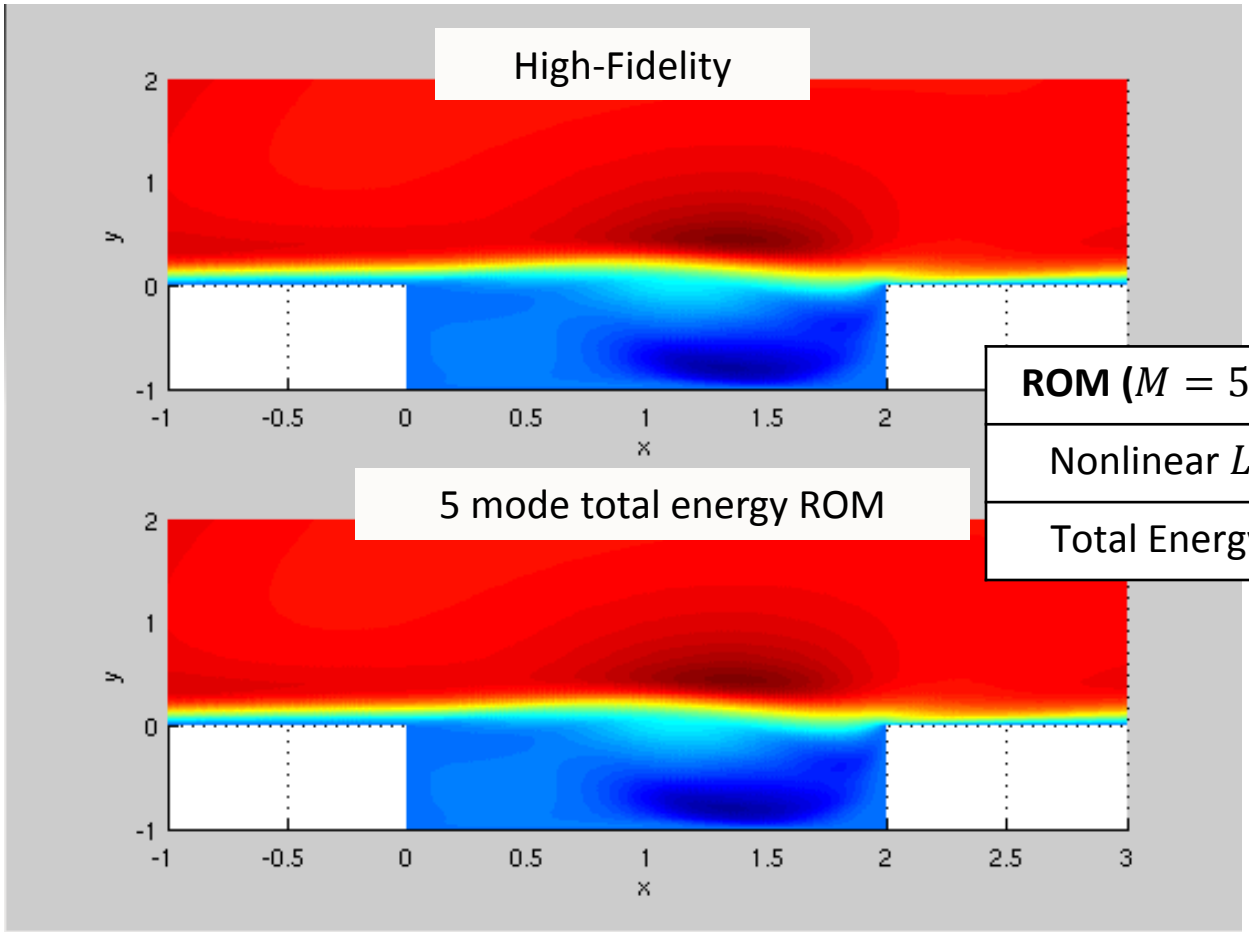


Figure above: viscous laminar cavity problem domain/mesh.

Numerical results: viscous laminar cavity



ROM ($M = 5$ modes)	Error (L^2 norm)
Nonlinear L^2 ROM	NaN
Total Energy ROM	5.52×10^{-2}

- L2 ROM blows up; energy ROM remains stable.

- **Future work:** improving efficiency of total energy ROMs through incorporation of hyper-reduction (e.g., DEIM, gappy POD).

Figure above: u-component of velocity as a function of time t

Outline

	Continuous Projection	Discrete Projection
Change projection (*a priori*)	Energy inner products (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	**Eigenvalue reassignment** (IKT *et al.*)
Change ROM basis (*a posteriori*)	Basis rotation (Balajewicz, IKT, *et al.*)	Optimization-based right basis modificatiOon (Amsallem)

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. **Approaches for stabilizing *a posteriori* unstable ROMs.**
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

with: Matt Barone (SNL), Srini Arunajatesan (SNL), Bart van Bloemen Waanders (SNL)

Stable ROMs for Linear Time-Invariant (LTI) systems

ies

Attention restricted to **Linear Time Invariant (LTI) systems**

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

as a first step towards the more general nonlinear case.

Stable ROMs for Linear Time-Invariant (LTI) systems

ies

Attention restricted to **Linear Time Invariant (LTI) systems**

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

as a first step towards the more general nonlinear case.

LTI Full Order Model (FOM)

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

LTI Reduced Order Model (ROM)

$$\dot{\boldsymbol{x}}_M(t) = \boldsymbol{A}_M\boldsymbol{x}_M(t) + \boldsymbol{B}_M\boldsymbol{u}(t)$$
$$\boldsymbol{y}_M(t) = \boldsymbol{C}_M\boldsymbol{x}_M(t)$$

$$\boldsymbol{A}_M = \boldsymbol{\Phi}^T\boldsymbol{A}\boldsymbol{\Phi}$$
$$\boldsymbol{B}_M = \boldsymbol{\Phi}^T\boldsymbol{B}$$
$$\boldsymbol{C}_M = \boldsymbol{C}\boldsymbol{\Phi}$$

Stable ROMs for Linear Time-Invariant (LTI) systems

ies

Attention restricted to **Linear Time Invariant (LTI) systems**

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

as a first step towards the more general nonlinear case.

LTI Full Order Model (FOM)

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

LTI Reduced Order Model (ROM)

$$\dot{\boldsymbol{x}}_M(t) = \boldsymbol{A}_M\boldsymbol{x}_M(t) + \boldsymbol{B}_M\boldsymbol{u}(t)$$
$$\boldsymbol{y}_M(t) = \boldsymbol{C}_M\boldsymbol{x}_M(t)$$

Problem: $\boldsymbol{A}$ stable $\not\Rightarrow$ $\boldsymbol{A}_M$ stable!

$$\boldsymbol{A}_M = \boldsymbol{\Phi}^T\boldsymbol{A}\boldsymbol{\Phi}$$
$$\boldsymbol{B}_M = \boldsymbol{\Phi}^T\boldsymbol{B}$$
$$\boldsymbol{C}_M = \boldsymbol{C}\boldsymbol{\Phi}$$

Stable ROMs for Linear Time-Invariant (LTI) systems

ies

Attention restricted to **Linear Time Invariant (LTI) systems**

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

as a first step towards the more general nonlinear case.

LTI Full Order Model (FOM)

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

LTI Reduced Order Model (ROM)

$$\dot{\boldsymbol{x}}_M(t) = \boldsymbol{A}_M\boldsymbol{x}_M(t) + \boldsymbol{B}_M\boldsymbol{u}(t)$$
$$\boldsymbol{y}_M(t) = \boldsymbol{C}_M\boldsymbol{x}_M(t)$$

Problem: $\boldsymbol{A}$ stable $\not\Rightarrow$ $\boldsymbol{A}_M$ stable!

$$\boldsymbol{A}_M = \boldsymbol{\Phi}^T\boldsymbol{A}\boldsymbol{\Phi}$$
$$\boldsymbol{B}_M = \boldsymbol{\Phi}^T\boldsymbol{B}$$
$$\boldsymbol{C}_M = \boldsymbol{C}\boldsymbol{\Phi}$$

Solution:

Unstable ROM ($\boldsymbol{A}_M$ unstable) → Black box: Stabilization Algorithm ($\boldsymbol{A}_M \leftarrow \widetilde{\boldsymbol{A}}_M$) → Stable and Accurate ROM ($\widetilde{\boldsymbol{A}}_M$ stable)

ROM stabilization via optimization-based eigenvalue reassignment (IKT *et al.*, 2014)

ROM Stabilization Optimization Problem
(Constrained Nonlinear Least Squares):

$$\min_{\lambda_i^u} \sum_{k=1}^{K} ||\boldsymbol{y}^k - \boldsymbol{y}_M{}^k||_2{}^2 \qquad (1)$$
$$s.t.\ Re(\lambda_i^u) < 0$$

Idea: modify ROM system s.t. $\boldsymbol{A}_M$ is stable and discrepancy b/w ROM output $\boldsymbol{y}_M(t)$ and FOM output $\boldsymbol{y}(t)$ is minimal.

Replace unstable $\boldsymbol{A}_M$ with stable $\widetilde{\boldsymbol{A}}_M$.

- λ_i^u = unstable eigenvalues of original ROM matrix $\boldsymbol{A}_M$.
- $\boldsymbol{y}^k = \boldsymbol{y}(t_k)$ = snapshot output at t_k.
- $\boldsymbol{y}_M{}^k = \boldsymbol{C}_M \left[\exp(t_k \boldsymbol{A}_M)\, \boldsymbol{x}_M(0) + \int_0^{t_k} \exp\{(t_k - \tau)\, \boldsymbol{A}_M\} \boldsymbol{B}_M u(\tau) d\tau\right]$ = ROM output at t_k.

ROM stabilization via optimization-based eigenvalue reassignment (IKT *et al.*, 2014)

ROM Stabilization Optimization Problem
(Constrained Nonlinear Least Squares):

$$\min_{\lambda_i^u} \sum_{k=1}^{K} ||\boldsymbol{y}^k - \boldsymbol{y}_M{}^k||_2{}^2 \qquad (1)$$
$$s.t.\ Re(\lambda_i^u) < 0$$

Idea: modify ROM system s.t. $\boldsymbol{A}_M$ is stable and discrepancy b/w ROM output $\boldsymbol{y}_M(t)$ and FOM output $\boldsymbol{y}(t)$ is minimal.

Replace unstable $\boldsymbol{A}_M$ with stable $\widetilde{\boldsymbol{A}}_M$.

- λ_i^u = unstable eigenvalues of original ROM matrix $\boldsymbol{A}_M$.
- $\boldsymbol{y}^k = \boldsymbol{y}(t_k)$ = snapshot output at t_k.
- $\boldsymbol{y}_M{}^k = \boldsymbol{C}_M \left[\exp(t_k \boldsymbol{A}_M)\, \boldsymbol{x}_M(0) + \int_0^{t_k} \exp\{(t_k - \tau)\, \boldsymbol{A}_M\} \boldsymbol{B}_M u(\tau) d\tau\right]$ = ROM output at t_k.
- ROM stabilization optimization problem is small: $< O(M)$.

ROM stabilization via optimization-based eigenvalue reassignment (IKT *et al.*, 2014)

ROM Stabilization Optimization Problem
(Constrained Nonlinear Least Squares):

$$\min_{\lambda_i^u} \sum_{k=1}^{K} ||\boldsymbol{y}^k - \boldsymbol{y}_M{}^k||_2{}^2 \qquad (1)$$
$$s.t.\ Re(\lambda_i^u) < 0$$

Idea: modify ROM system s.t. $\boldsymbol{A}_M$ is stable and discrepancy b/w ROM output $\boldsymbol{y}_M(t)$ and FOM output $\boldsymbol{y}(t)$ is minimal.

Replace unstable $\boldsymbol{A}_M$ with stable $\widetilde{\boldsymbol{A}}_M$.

- λ_i^u = unstable eigenvalues of original ROM matrix $\boldsymbol{A}_M$.
- $\boldsymbol{y}^k = \boldsymbol{y}(t_k)$ = snapshot output at t_k.
- $\boldsymbol{y}_M{}^k = \boldsymbol{C}_M \left[\exp(t_k \boldsymbol{A}_M)\, \boldsymbol{x}_M(0) + \int_0^{t_k} \exp\{(t_k - \tau)\, \boldsymbol{A}_M\} \boldsymbol{B}_M u(\tau) d\tau\right]$ = ROM output at t_k.
- ROM stabilization optimization problem is small: $< O(M)$.
- ROM stabilization optimization problem can be solved by standard optimization algorithms, e.g., interior point method.
  - We use `fmincon` function in MATLAB's optimization toolbox.
  - We implement ROM stabilization optimization problem in ***characteristic variables*** $\boldsymbol{z}_M(t) = \boldsymbol{S}_M{}^{-1} \boldsymbol{x}_M(t)$ where $\boldsymbol{A}_M = \boldsymbol{S}_M \boldsymbol{D}_M \boldsymbol{S}_M{}^{-1}$.

ROM stabilization via optimization-based eigenvalue reassignment

Algorithm

- Diagonalize the ROM matrix $\boldsymbol{A}_M$: $\boldsymbol{A}_M = \boldsymbol{S}_M \boldsymbol{D}_M \boldsymbol{S}_M{}^{-1}$.
- Initialize a diagonal $M \times M$ matrix $\widetilde{\boldsymbol{D}}_M$. Set $j = 1$.
- **for** $i = 1$ to M
  - **if** $Re(D_M\,(i,i) < 0)$, set $\widetilde{D}_M(i,i) = D_M(i,i)$.
  - **else**, set $\widetilde{D}_M(i,i) = \lambda_j{}^u$.
- Increment $j \leftarrow j + 1$.
- Solve the optimization problem (1) for the eigenvalues $\{\lambda_j{}^u\}$ using an optimization algorithm (e.g., interior point method).
- Evaluate $\widetilde{\boldsymbol{D}}_M$ at the solution of the optimization problem (1).
- Return the stabilized ROM system, given by $\boldsymbol{A}_M \leftarrow \widetilde{\boldsymbol{A}}_M = \boldsymbol{S}_M \widetilde{\boldsymbol{D}}_M \boldsymbol{S}_M{}^{-1}$.

ROM stabilization via optimization-based eigenvalue reassignment

Algorithm

- Diagonalize the ROM matrix $\boldsymbol{A}_M$: $\boldsymbol{A}_M = \boldsymbol{S}_M \boldsymbol{D}_M \boldsymbol{S}_M{}^{-1}$.
- Initialize a diagonal $M \times M$ matrix $\widetilde{\boldsymbol{D}}_M$. Set $j = 1$.
- **for** $i = 1$ to M
  - **if** $Re(D_M\,(i,i) < 0)$, set $\widetilde{D}_M(i,i) = D_M(i,i)$.
  - **else**, set $\widetilde{D}_M(i,i) = \lambda_j{}^u$.
- Increment $j \leftarrow j + 1$.
- Solve the optimization problem (1) for the eigenvalues $\{\lambda_j{}^u\}$ using an optimization algorithm (e.g., interior point method).
- Evaluate $\widetilde{\boldsymbol{D}}_M$ at the solution of the optimization problem (1).
- Return the stabilized ROM system, given by $\boldsymbol{A}_M \leftarrow \widetilde{\boldsymbol{A}}_M = \boldsymbol{S}_M \widetilde{\boldsymbol{D}}_M \boldsymbol{S}_M{}^{-1}$.

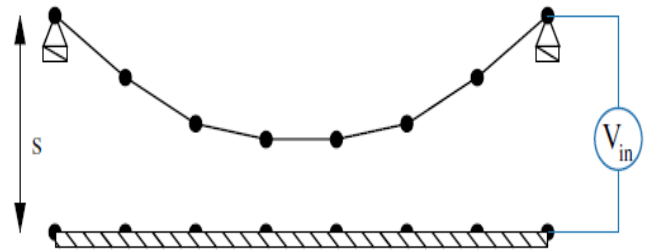
- Solution to optimization problem (1) may not be unique.
- Can solve (1) for real or complex-conjugate pair eigenvalues:
  - $\lambda_j{}^u \in \mathbb{R}$ s.t. constraint $\lambda_j{}^u < 0$.
  - $\lambda_j{}^u = \lambda_j{}^{ur} + i\,\lambda_j{}^{uc}$, $\lambda_{j+1}{}^u = \lambda_j{}^{ur} - i\,\lambda_j{}^{uc} \in \mathbb{C}$ where $\lambda_j{}^{ur}, \lambda_j{}^{uc} \in \mathbb{R}$ s.t. constraint $\lambda_j{}^{ur} < 0$.

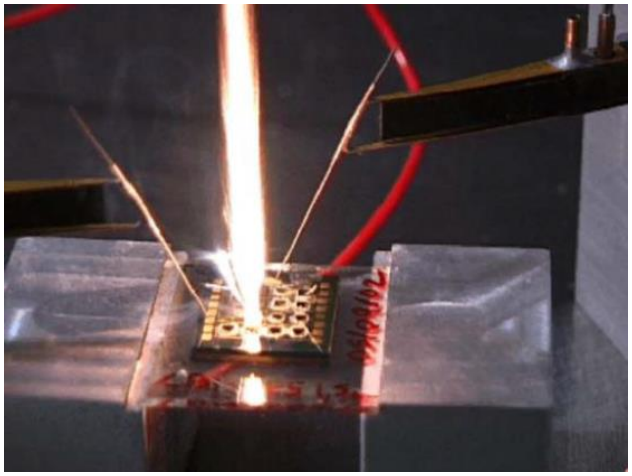
Numerical results: electrostatically actuated beam benchmark

- FOM = 1D model of electrostatically actuated beam.
- Application of model: microelectromechanical systems (MEMS) devices such as electromechanical radio frequency (RF) filters.
- 1 input corresponding to periodic on/off switching, 1 output, initial condition $\boldsymbol{x}(0) = \boldsymbol{0}_N$.
- Second order linear semi-discrete system of the form:

$$\boldsymbol{M}\ddot{\boldsymbol{x}}(t) + \boldsymbol{E}\dot{\boldsymbol{x}}(t) + \boldsymbol{K}\boldsymbol{x}(t) = \boldsymbol{B}\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)$$

- Matrices $\boldsymbol{M}, \boldsymbol{E}, \boldsymbol{K}, \boldsymbol{B}, \boldsymbol{C}$ specifying the problem downloaded from the Oberwolfach ROM repository*.
- 2nd order linear system re-written as 1st order LTI system for purpose of analysis/model reduction.





- FOM is stable.

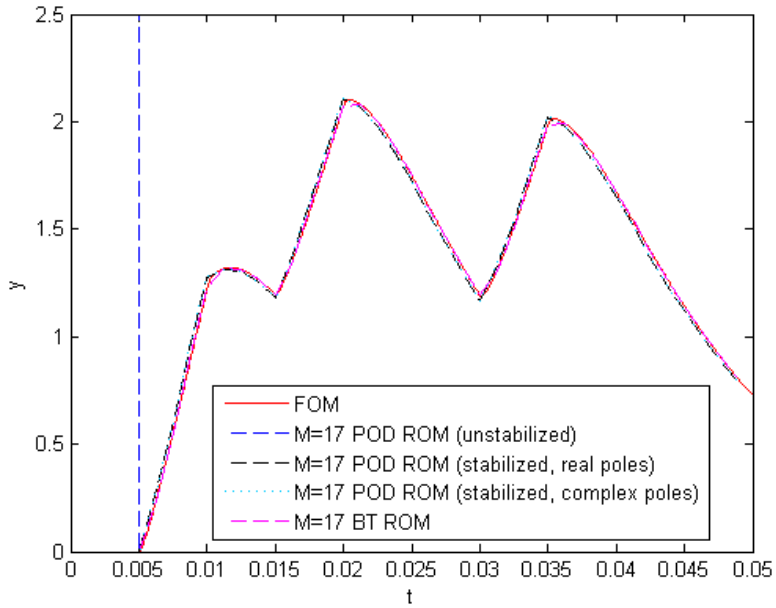
* Oberwolfach ROM benchmark repository: http://simulation.uni-freiburg.de/downloads/benchmark.

Numerical results: electrostatically actuated beam benchmark

- $M = 17$ POD/Galerkin ROM constructed from $K = 1000$ snapshots up to time $t = 0.05$.
- $M = 17$ POD/Galerkin ROM has 4 unstable eigenvalues (all real).
  - Two options for ROM stabilization optimization problem:

    Option 1: Solve for $\lambda_1, \lambda_2, \lambda_3, \lambda_4 \in \mathbb{R}$ s.t. the constraint $\lambda_1, \lambda_2, \lambda_3, \lambda_4 < 0$.

    Option 2: Solve for $\lambda_1 + \lambda_2 i, \lambda_1 - \lambda_2 i, \lambda_3 + \lambda_4 i, \lambda_3 - \lambda_4 i \in \mathbb{C}$ s.t. the constraint $\lambda_1, \lambda_3 < 0$.
- Initial guess for `fmincon` interior point method: $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = -1$.



ROM	$\frac{\sqrt{\sum_{k=1}^{K} \lVert \boldsymbol{y}^k - \boldsymbol{y}_M{}^k \rVert_2{}^2}}{\sqrt{\sum_{k=1}^{K} \lVert \boldsymbol{y}_k \rVert_2{}^2}}$
Unstabilized POD	NaN
Optimization Stabilized POD (Real Poles)	0.0194
Optimization Stabilized POD (Complex-Conjugate Poles)	0.0205
Balanced Truncation	$1.370e-6$

Consistency?

- One can show that $\widetilde{\boldsymbol{A}}_M$ from the algorithm on the previous slide is given by:

$$\widetilde{\boldsymbol{A}}_M = \boldsymbol{A}_M - \boldsymbol{B}_C \boldsymbol{K}_C$$

for a specific $\boldsymbol{B}_C$ and $\boldsymbol{K}_C$ (IKT *et al.* 2014).

Consistency?

- One can show that $\widetilde{\boldsymbol{A}}_M$ from the algorithm on the previous slide is given by:

$$\widetilde{\boldsymbol{A}}_M = \boldsymbol{A}_M - \boldsymbol{B}_C \boldsymbol{K}_C$$

for a specific $\boldsymbol{B}_C$ and $\boldsymbol{K}_C$ (IKT *et al.* 2014).

- Modifying system as $\boldsymbol{A}_M \leftarrow \widetilde{\boldsymbol{A}}_M$ can be viewed as adding a linear "controller" to the system:

$$\dot{\boldsymbol{x}}_M(t) = \boldsymbol{A}_M \boldsymbol{x}_M(t) + \boldsymbol{B}_M \boldsymbol{u}(t) + \boldsymbol{B}_C \boldsymbol{u}_C(t)$$
$$\boldsymbol{y}_M(t) = \boldsymbol{C}_M \boldsymbol{x}_M(t)$$

where $\boldsymbol{u}_C(t) = -\boldsymbol{K}_C \boldsymbol{x}_M(t)$.

Consistency?

- One can show that $\widetilde{\boldsymbol{A}}_M$ from the algorithm on the previous slide is given by:

$$\widetilde{\boldsymbol{A}}_M = \boldsymbol{A}_M - \boldsymbol{B}_C \boldsymbol{K}_C$$

  for a specific $\boldsymbol{B}_C$ and $\boldsymbol{K}_C$ (IKT *et al.* 2014).

- Modifying system as $\boldsymbol{A}_M \leftarrow \widetilde{\boldsymbol{A}}_M$ can be viewed as adding a linear "controller" to the system:

$$\dot{\boldsymbol{x}}_M(t) = \boldsymbol{A}_M \boldsymbol{x}_M(t) + \boldsymbol{B}_M \boldsymbol{u}(t) + \boldsymbol{B}_C \boldsymbol{u}_C(t)$$
$$\boldsymbol{y}_M(t) = \boldsymbol{C}_M \boldsymbol{x}_M(t)$$

  where $\boldsymbol{u}_C(t) = -\boldsymbol{K}_C \boldsymbol{x}_M(t)$.

- Deficiencies of approach:
  - ROM is ***inconsistent*** with FOM.
  - Efficient extensions to ***nonlinear*** case are unclear.

Consistency?

- One can show that $\widetilde{\boldsymbol{A}}_M$ from the algorithm on the previous slide is given by:

$$\widetilde{\boldsymbol{A}}_M = \boldsymbol{A}_M - \boldsymbol{B}_C \boldsymbol{K}_C$$

  for a specific $\boldsymbol{B}_C$ and $\boldsymbol{K}_C$ (IKT *et al.* 2014).

- Modifying system as $\boldsymbol{A}_M \leftarrow \widetilde{\boldsymbol{A}}_M$ can be viewed as adding a linear "controller" to the system:

$$\dot{\boldsymbol{x}}_M(t) = \boldsymbol{A}_M \boldsymbol{x}_M(t) + \boldsymbol{B}_M \boldsymbol{u}(t) + \boldsymbol{B}_C \boldsymbol{u}_C(t)$$
$$\boldsymbol{y}_M(t) = \boldsymbol{C}_M \boldsymbol{x}_M(t)$$

  where $\boldsymbol{u}_C(t) = -\boldsymbol{K}_C \boldsymbol{x}_M(t)$.

- Deficiencies of approach:
  - ROM is ***inconsistent*** with FOM.
  - Efficient extensions to ***nonlinear*** case are unclear.

Next section of talk (basis rotation): method for modifying *a posteriori* an unstable ROM that maintains ***consistency*** (and is applicable to ***nonlinear*** problems).

Outline

	Continuous Projection	Discrete Projection
Change projection (*a priori*)	Energy inner products (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	Eigenvalue reassignment (IKT *et al.*)
Change ROM basis (*a posteriori*)	**Basis rotation** (Balajewicz, IKT, *et al.*)	Optimization-based right basis modificatiOon (Amsallem)

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. **Approaches for stabilizing *a posteriori* unstable ROMs.**
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. Ongoing/future work.

with: Maciej Balajewicz (UIUC), Earl Dowell (Duke)

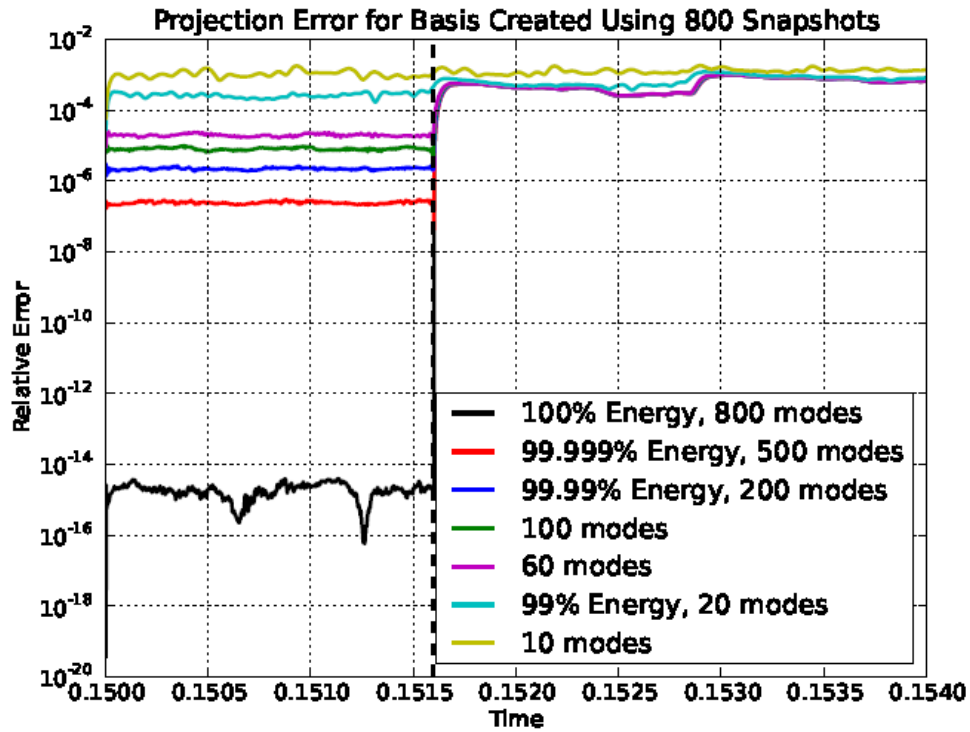
Extreme model reduction

- Most realistic applications (e.g., high *Re* compressible cavity): basis that captures >***99% snapshot energy*** is required to accurately reproduce snapshots.

  → leads to $M > O(1000)$ except for ***toy problems*** and/or ***low-fidelity*** models.

- Higher order modes are in general ***unreliable for prediction***, so including them in the basis is unlikely to improve the ***predictive*** capabilities of a ROM.

Figure (right) shows projection error for POD basis constructed using 800 snapshots for cavity problem. Dashed line = end of snapshot collection period.



We are looking for an approach that enables ***extreme model reduction***: ROM basis size is $O(10)$ or $O(100)$.

Mode truncation instability

Projection-based MOR necessitates ***truncation.***

Mode truncation instability

Projection-based MOR necessitates ***truncation.***

- POD is, by definition and design, biased towards the ***large, energy producing*** scales of the flow (i.e., modes with large POD eigenvalues).

Mode truncation instability

Projection-based MOR necessitates ***truncation.***

- POD is, by definition and design, biased towards the ***large, energy producing*** scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a ***data compression*** point of view (i.e., small POD eigenvalues) but are crucial for the ***dynamical equations***.

Mode truncation instability

Projection-based MOR necessitates ***truncation.***

- POD is, by definition and design, biased towards the ***large, energy producing*** scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a ***data compression*** point of view (i.e., small POD eigenvalues) but are crucial for the ***dynamical equations***.
- For fluid flow applications, higher-order modes are associated with energy ***dissipation***

Mode truncation instability

Projection-based MOR necessitates ***truncation.***

- POD is, by definition and design, biased towards the ***large, energy producing*** scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a ***data compression*** point of view (i.e., small POD eigenvalues) but are crucial for the ***dynamical equations***.
- For fluid flow applications, higher-order modes are associated with energy ***dissipation***

$\Longrightarrow$ low-dimensional ROMs can be ***inaccurate*** and ***unstable***.

Mode truncation instability

Projection-based MOR necessitates ***truncation.***

- POD is, by definition and design, biased towards the ***large, energy producing*** scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a ***data compression*** point of view (i.e., small POD eigenvalues) but are crucial for the ***dynamical equations***.
- For fluid flow applications, higher-order modes are associated with energy ***dissipation***

$\Longrightarrow$ low-dimensional ROMs can be ***inaccurate*** and ***unstable***.

For a low-dimensional ROM to be stable and accurate, the ***truncated/unresolved subspace*** must be accounted for.

Mode truncation instability

Projection-based MOR necessitates ***truncation.***

- POD is, by definition and design, biased towards the ***large, energy producing*** scales of the flow (i.e., modes with large POD eigenvalues).
- Truncated/unresolved modes are negligible from a ***data compression*** point of view (i.e., small POD eigenvalues) but are crucial for the ***dynamical equations***.
- For fluid flow applications, higher-order modes are associated with energy ***dissipation***

$\Longrightarrow$ low-dimensional ROMs can be ***inaccurate*** and ***unstable***.

For a low-dimensional ROM to be stable and accurate, the ***truncated/unresolved subspace*** must be accounted for.

Turbulence Modeling
(traditional approach)

Subspace Rotation
(our approach)

Governing equations and ROM

- 3D compressible Navier-Stokes equations in *primitive specific volume form*:

[PDEs]

$$\zeta_{,t} + \zeta_{,j} u_j - \zeta u_{j,j} = 0$$

$$u_{i,t} + u_{i,j} u_j + \zeta p_{,i} - \frac{1}{Re} \zeta \tau_{ij,j} = 0$$

$$p_{,t} + u_j p_{,j} + \gamma u_{j,j} p - \left(\frac{\gamma}{PrRe}\right) \left(\kappa (p\zeta)_{,j}\right)_{,j} - \left(\frac{\gamma - 1}{Re}\right) u_{i,j} \tau_{ij} = 0 \qquad (2)$$

Governing equations and ROM

- 3D compressible Navier-Stokes equations in *primitive specific volume form:*

[PDEs]

$$\zeta_{,t} + \zeta_{,j} u_j - \zeta u_{j,j} = 0$$
$$u_{i,t} + u_{i,j} u_j + \zeta p_{,i} - \frac{1}{Re} \zeta \tau_{ij,j} = 0$$
$$p_{,t} + u_j p_{,j} + \gamma u_{j,j} p - \left(\frac{\gamma}{PrRe}\right) \left(\kappa (p\zeta)_{,j}\right)_{,j} - \left(\frac{\gamma - 1}{Re}\right) u_{i,j} \tau_{ij} = 0 \qquad (2)$$

- POD discretization $\left(\boldsymbol{q}(\boldsymbol{x}, t) \approx \sum_{i=1}^{M} a_i(t) \boldsymbol{\phi}_i(\boldsymbol{x})\right)$ + Galerkin projection applied to (2) yields a system of *M coupled quadratic ODEs*:

[ROM]

$$\frac{d\boldsymbol{a}}{dt} = \boldsymbol{C} + \boldsymbol{L}\boldsymbol{a} + \left[\boldsymbol{a}^T \boldsymbol{Q}^{(1)} \boldsymbol{a} + \boldsymbol{a}^T \boldsymbol{Q}^{(2)} \boldsymbol{a} + \cdots + \boldsymbol{a}^T \boldsymbol{Q}^{(M)} \boldsymbol{a}\right]^T \qquad (3)$$

where $\boldsymbol{C} \in \mathbb{R}^M, \boldsymbol{L} \in \mathbb{R}^{M \times M}$ and $\boldsymbol{Q}^{(i)} \in \mathbb{R}^{M \times M}$ for all $i = 1, \ldots, M$.

Traditional linear eddy-viscosity approach to account for modal truncation

- Dissipative dynamics of truncated higher-order modes are modeled using an additional linear term:

$$\frac{d\boldsymbol{a}}{dt} = \boldsymbol{C} + \boldsymbol{L}\boldsymbol{a} + \left[\boldsymbol{a}^T \boldsymbol{Q}^{(1)} \boldsymbol{a} + \boldsymbol{a}^T \boldsymbol{Q}^{(2)} \boldsymbol{a} + \cdots + \boldsymbol{a}^T \boldsymbol{Q}^{(M)} \boldsymbol{a}\right]^T$$

Traditional linear eddy-viscosity approach to account for modal truncation

- Dissipative dynamics of truncated higher-order modes are modeled using an additional linear term:

$$\frac{d\boldsymbol{a}}{dt} = \boldsymbol{C} + (\boldsymbol{L} + \boldsymbol{L}_\nu)\boldsymbol{a} + \left[\boldsymbol{a}^T\boldsymbol{Q}^{(1)}\boldsymbol{a} + \boldsymbol{a}^T\boldsymbol{Q}^{(2)}\boldsymbol{a} + \cdots + \boldsymbol{a}^T\boldsymbol{Q}^{(M)}\boldsymbol{a}\right]^T$$

Traditional linear eddy-viscosity approach to account for modal truncation

- Dissipative dynamics of truncated higher-order modes are modeled using an additional linear term:

$$\frac{d\boldsymbol{a}}{dt} = \boldsymbol{C} + (\boldsymbol{L} + \boldsymbol{L}_\nu)\boldsymbol{a} + \left[\boldsymbol{a}^T\boldsymbol{Q}^{(1)}\boldsymbol{a} + \boldsymbol{a}^T\boldsymbol{Q}^{(2)}\boldsymbol{a} + \cdots + \boldsymbol{a}^T\boldsymbol{Q}^{(M)}\boldsymbol{a}\right]^T$$

- $\boldsymbol{L}_\nu$ is designed to decrease magnitude of positive eigenvalues and increase magnitude of negative eigenvalues of $\boldsymbol{L} + \boldsymbol{L}_\nu$ (for stability).

Traditional linear eddy-viscosity approach to account for modal truncation

- Dissipative dynamics of truncated higher-order modes are modeled using an additional linear term:

$$\frac{d\boldsymbol{a}}{dt} = \boldsymbol{C} + (\boldsymbol{L} + \boldsymbol{L}_\nu)\boldsymbol{a} + \left[\boldsymbol{a}^T\boldsymbol{Q}^{(1)}\boldsymbol{a} + \boldsymbol{a}^T\boldsymbol{Q}^{(2)}\boldsymbol{a} + \cdots + \boldsymbol{a}^T\boldsymbol{Q}^{(M)}\boldsymbol{a}\right]^T$$

- $\boldsymbol{L}_\nu$ is designed to decrease magnitude of positive eigenvalues and increase magnitude of negative eigenvalues of $\boldsymbol{L} + \boldsymbol{L}_\nu$ (for stability).

- <u>Disadvantages of this approach</u>:
  1. Additional term destroys <u>*consistency*</u> between ROM and Navier-Stokes equations.
  2. Calibration is necessary to derive optimal $\boldsymbol{L}_\nu$ and optimal value is <u>*flow dependent*</u>.
  3. Inherently a <u>*linear model*</u> → cannot be expected to perform well for all classes of problems (e.g., nonlinear).

Proposed new approach for modal truncation: basis rotation

Instead of modeling truncation via additional linear term, model the truncation *a priori* by "rotating" the projection subspace into a more dissipative regime

Proposed new approach for modal truncation: basis rotation

Instead of modeling truncation via additional linear term, model the truncation *a priori* by "rotating" the projection subspace into a more dissipative regime

Illustrative example

- *Standard approach:* retain only the most energetic POD modes, i.e., $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3, \boldsymbol{\phi}_4, \ldots$
- *Proposed approach:* choose some higher order basis modes to increase dissipation, i.e., $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_6, \boldsymbol{\phi}_8, \ldots$

Proposed new approach for modal truncation: basis rotation

Instead of modeling truncation via additional linear term, model the truncation *a priori* by "rotating" the projection subspace into a more dissipative regime

Illustrative example

- *Standard approach:* retain only the most energetic POD modes, i.e., $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3, \boldsymbol{\phi}_4, \ldots$
- *Proposed approach:* choose some higher order basis modes to increase dissipation, i.e., $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_6, \boldsymbol{\phi}_8, \ldots$

- *More generally:* approximate the solution using a linear superposition of $M + P$ (with $P > 0$) most energetic modes:

$$\widetilde{\boldsymbol{\phi}}_i = \sum_{j=1}^{M+P} X_{ij}\, \boldsymbol{\phi}_j, \quad i = 1, \ldots, M, \tag{4}$$

where $\boldsymbol{\phi} \in \mathbb{R}^{(M+P)\times M}$ is an orthonormal ($\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}_{M\times M}$) "rotation" matrix.

Goals of proposed new approach to account for modal truncation

Find $\boldsymbol{X}$ such that:

Goals of proposed new approach to account for modal truncation

Find $\boldsymbol{X}$ such that:

1. New modes $\widetilde{\boldsymbol{\phi}}$ remain <u>*good approximations*</u> of the flow

   $\rightarrow$ minimize the "rotation" angle, i.e., minimize $\left\|\boldsymbol{X} - \boldsymbol{I}_{(M+P),M}\right\|_F$

Goals of proposed new approach to account for modal truncation

Find $\boldsymbol{X}$ such that:

1. New modes $\widetilde{\boldsymbol{\phi}}$ remain <u>*good approximations*</u> of the flow

   → minimize the "rotation" angle, i.e., minimize $\left\|\boldsymbol{X} - \boldsymbol{I}_{(M+P),M}\right\|_F$

2. New modes produce <u>*stable*</u> and <u>*accurate*</u> ROMs.

   → ensure appropriate balance between energy production and energy dissipation.

Goals of proposed new approach to account for modal truncation

Find $\boldsymbol{X}$ such that:

1. New modes $\widetilde{\boldsymbol{\phi}}$ remain <u>*good approximations*</u> of the flow
   → minimize the "rotation" angle, i.e., minimize $\left\|\boldsymbol{X} - \boldsymbol{I}_{(M+P),M}\right\|_F$
2. New modes produce <u>*stable*</u> and <u>*accurate*</u> ROMs.
   → ensure appropriate balance between energy production and energy dissipation.

- Once $\boldsymbol{X}$ is found, the result is a system of the form (3) with:

$$Q^{(i)}{}_{jk} \leftarrow \sum_{s,q,r=1}^{M+P} X_{si} Q^{(s)}{}_{qr} X_{qr} X_{rk}, \quad \boldsymbol{L} \leftarrow \boldsymbol{X}^T \boldsymbol{L} \boldsymbol{X}, \quad \boldsymbol{C} \leftarrow \boldsymbol{X}^T \boldsymbol{C}^*$$

Minimal subspace rotation

- Trace minimization problem on the Stiefel manifold:

$$\begin{aligned} &\text{minimize}_{\boldsymbol{X}\in\mathcal{V}_{(M+P),M}} \quad -\operatorname{tr}\left(\boldsymbol{X}^T\boldsymbol{I}_{(M+P)\times M}\right) \\ &\text{subject to} \qquad \operatorname{tr}(\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}) = \eta \end{aligned} \tag{5}$$

- $\mathcal{V}_{(M+P),M} \in \left\{\boldsymbol{X} \in \mathbb{R}^{(M+P)\times M} : \boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}_M, P > 0\right\}$ is the *Stiefel manifold*.
- Constraint is traditional *linear eddy-viscosity closure model ansatz* → involves overall balance between *linear energy production* and *dissipation* / vanishing of *averaged total power* ($= \operatorname{tr}(\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}) +$ energy transfer).
  - $\eta \in \mathbb{R}$: proxy for the balance between linear energy production and energy dissipation (calculated iteratively using modal energy).
- Equation (5) is solved efficiently offline using the method of Lagrange multipliers (`Manopt MATLAB toolbox`).
- See (Balajewicz, IKT, Dowell, 2016) and Appendix slide for Algorithm.

Accounting for modal truncation: remarks

Proposed approach may be interpreted as an *a priori consistent* formulation of the eddy-viscosity turbulence modeling approach.

Accounting for modal truncation: remarks

Proposed approach may be interpreted as an *a priori consistent* formulation of the eddy-viscosity turbulence modeling approach.

- Advantages of proposed approach:
  1. Retains *consistency* between ROM and Navier-Stokes equations → no additional turbulence terms required.

Accounting for modal truncation: remarks

Proposed approach may be interpreted as an *a priori consistent* formulation of the eddy-viscosity turbulence modeling approach.

- Advantages of proposed approach:
  1. Retains *consistency* between ROM and Navier-Stokes equations → no additional turbulence terms required.
  2. Inherently a *nonlinear* model → should be expected to outperform linear models.

Accounting for modal truncation: remarks

Proposed approach may be interpreted as an *a priori consistent* formulation of the eddy-viscosity turbulence modeling approach.

- Advantages of proposed approach:
  1. Retains *consistency* between ROM and Navier-Stokes equations → no additional turbulence terms required.
  2. Inherently a *nonlinear* model → should be expected to outperform linear models.
  3. Works with *any* basis and Petrov-Galerkin projection.

Accounting for modal truncation: remarks

Proposed approach may be interpreted as an *a priori consistent* formulation of the eddy-viscosity turbulence modeling approach.

- Advantages of proposed approach:
  1. Retains *consistency* between ROM and Navier-Stokes equations → no additional turbulence terms required.
  2. Inherently a *nonlinear* model → should be expected to outperform linear models.
  3. Works with *any* basis and Petrov-Galerkin projection.

- Disadvantages of proposed approach:
  1. Off-line calibration of free parameter η is required.

Accounting for modal truncation: remarks

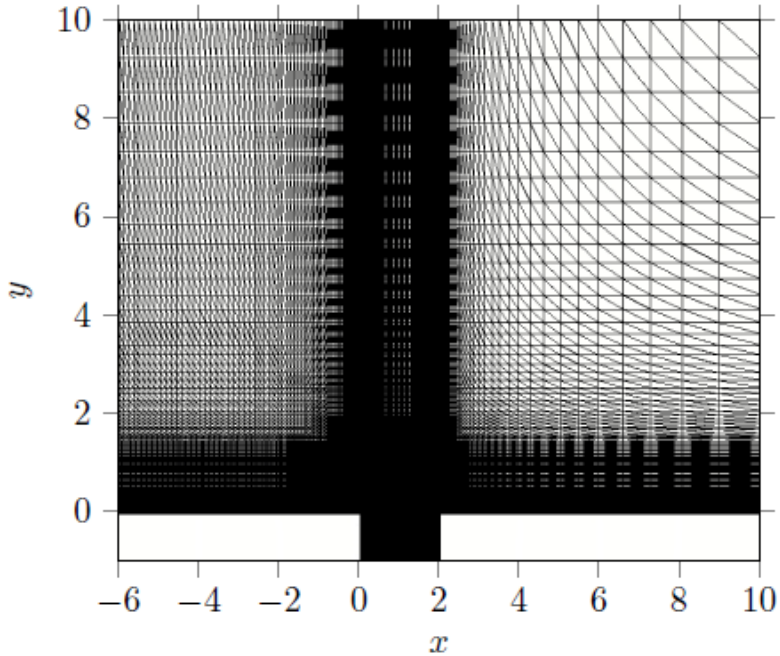
Proposed approach may be interpreted as an *a priori consistent* formulation of the eddy-viscosity turbulence modeling approach.

- Advantages of proposed approach:
  1. Retains *consistency* between ROM and Navier-Stokes equations → no additional turbulence terms required.
  2. Inherently a *nonlinear* model → should be expected to outperform linear models.
  3. Works with *any* basis and Petrov-Galerkin projection.

- Disadvantages of proposed approach:
  1. Off-line calibration of free parameter η is required.
  2. Stability cannot be proven like for incompressible case.

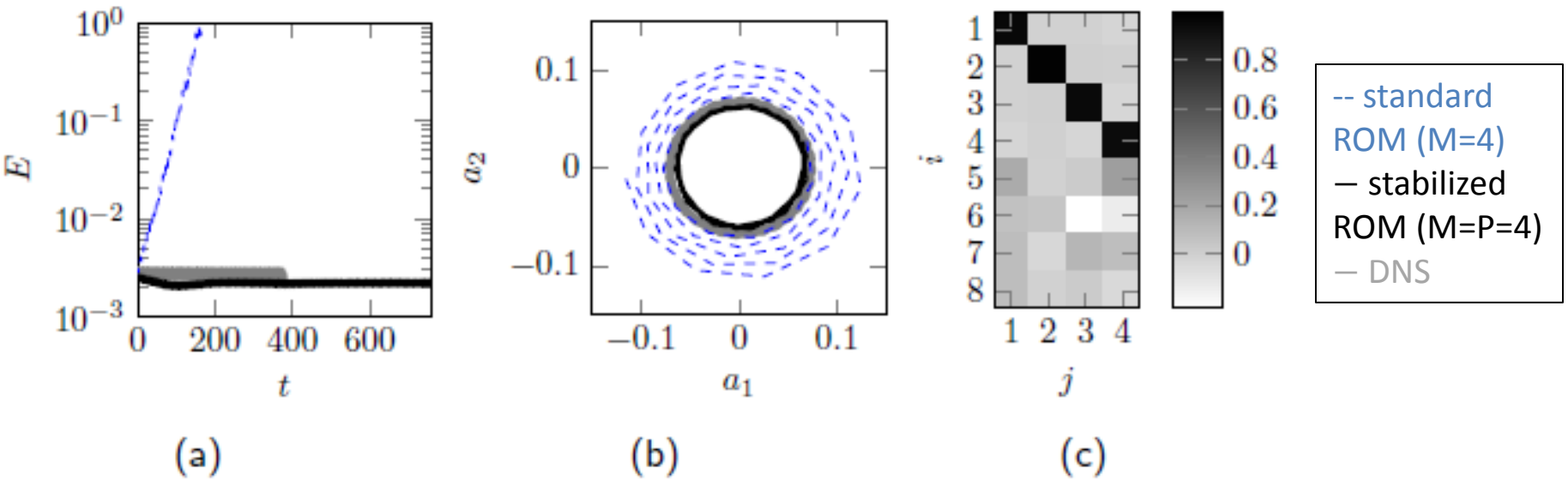
Numerical results: low Re number cavity

Flow over square cavity at Mach 0.6, Re = 1453.9, Pr = 0.72
$\Rightarrow M = 4$ ROM (91% snapshot energy).



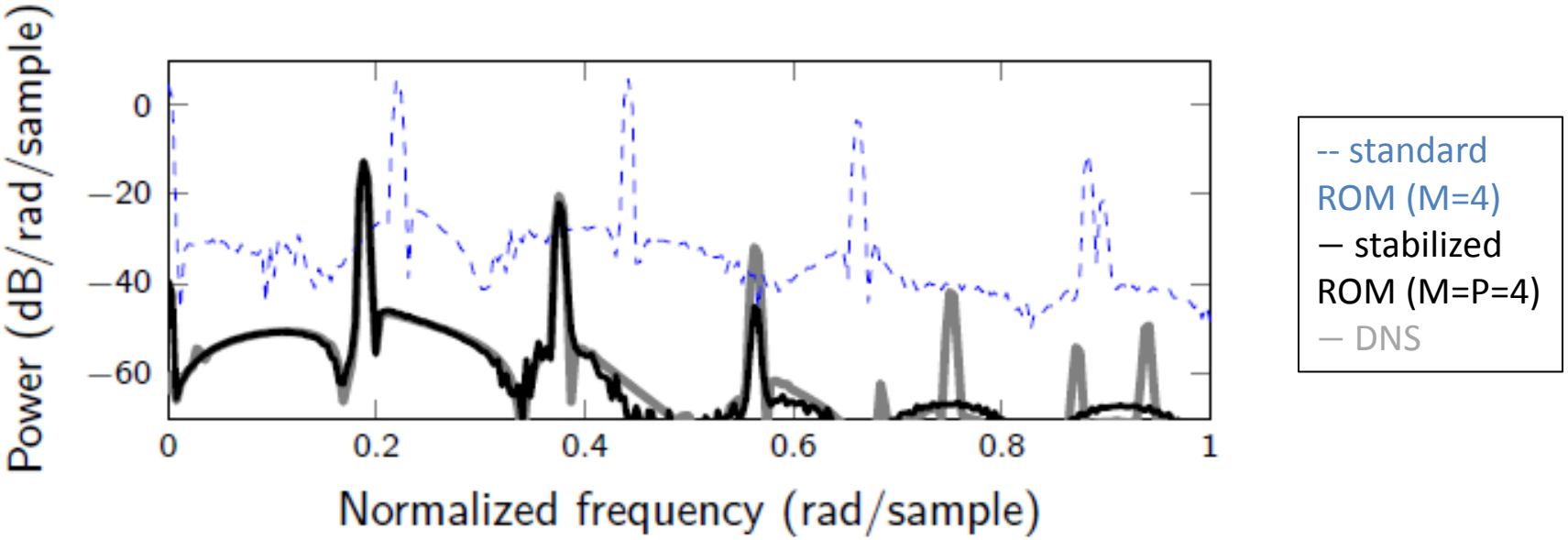
- *Above:* domain and mesh for viscous channel driven cavity problem.

Numerical results: low Re number cavity



- Figure (a) shows evolution of modal energy. Standard ROM is unstable.
- Figure (b) shows phase plot of first and second temporal basis $a_1(t)$ and $a_2(t)$. Stabilized ROM computes stable limit cycle; standard ROM computes unstable spiral.
- Figure (c) is an illustration of the stabilizing rotation matrix. Rotation is small: $\frac{\|\boldsymbol{X}-\boldsymbol{I}_{(M+P),M}\|_F}{M} = 0.188, \boldsymbol{X} \approx \boldsymbol{I}_{(M+P),M}$

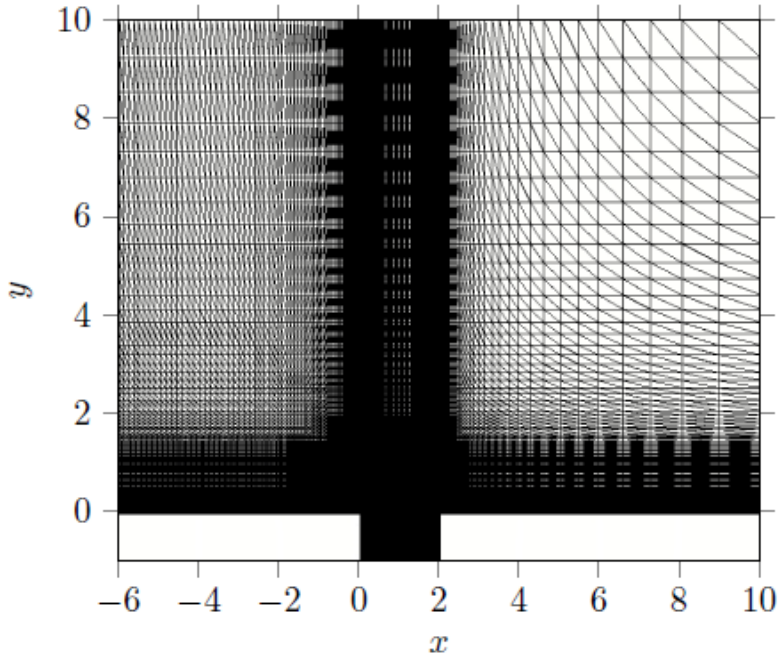
Numerical results: low Re number cavity



- Pressure power spectral density (PSD) at location $\boldsymbol{x} = (2, -1)$.

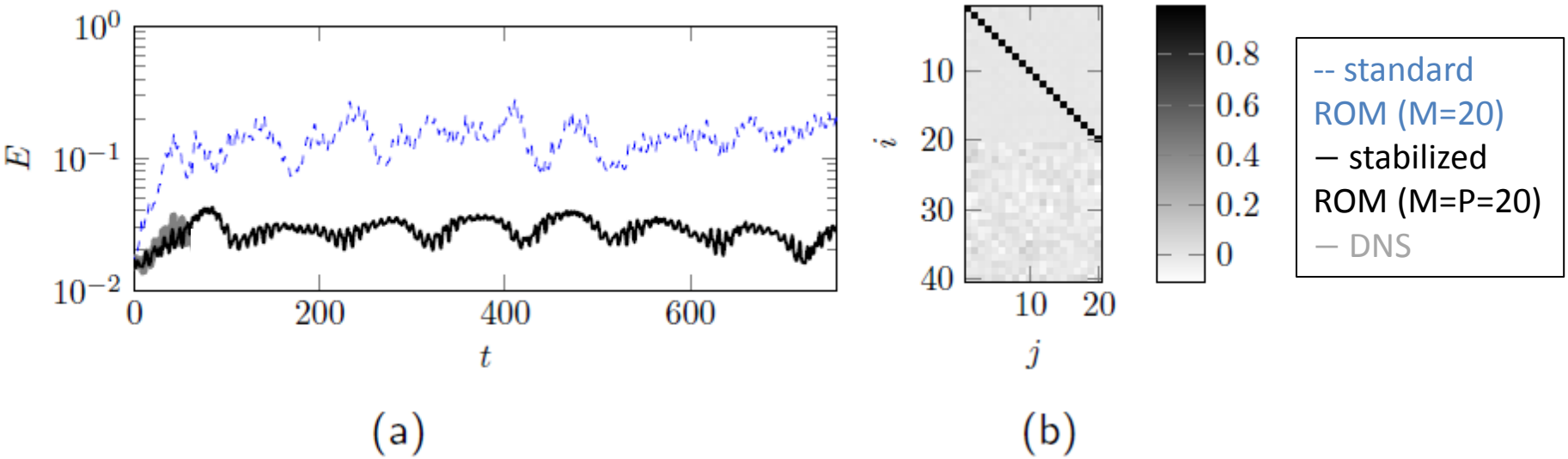
Numerical results: moderate Re number cavity

Flow over square cavity at Mach 0.6, Re = 5452.1, Pr = 0.72
$\Rightarrow M = 20$ ROM (71.8% snapshot energy).



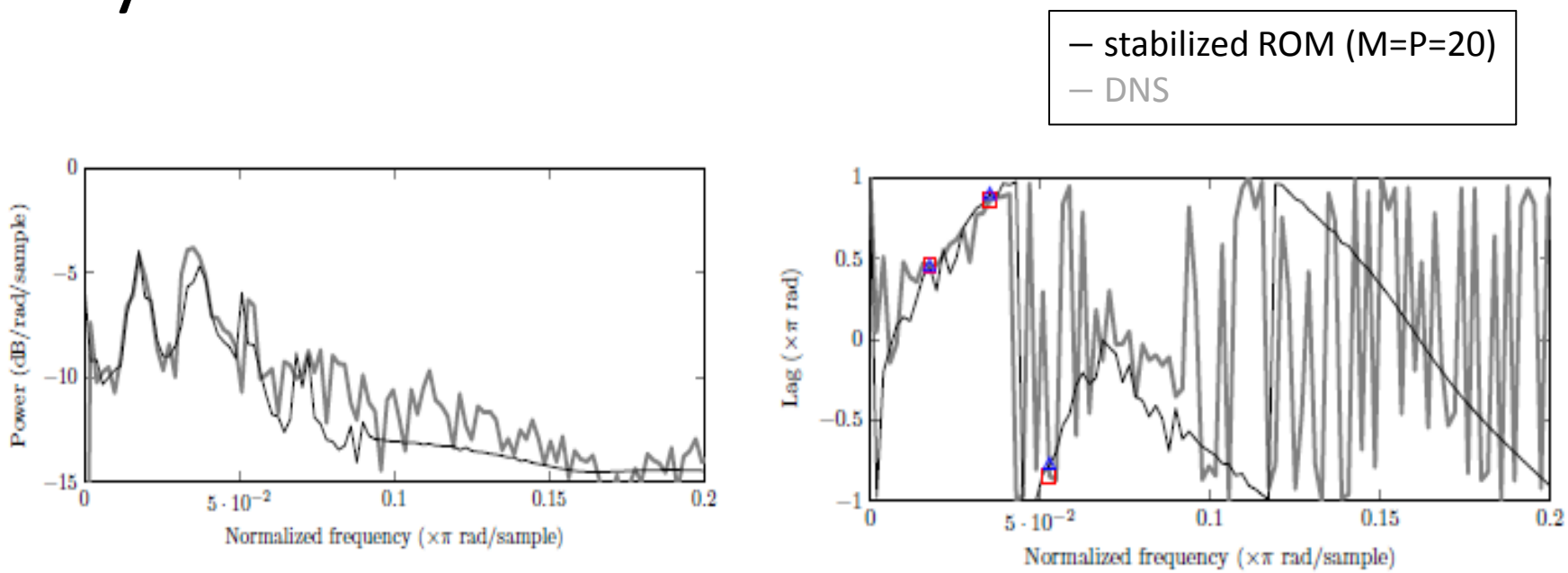
- *Above:* domain and mesh for viscous channel driven cavity problem.

Numerical results: moderate Re number cavity



- Figure (a) shows evolution of modal energy. Stabilized ROM energy closer to FOM.
- Figure (b) illustrates stabilizing rotation matrix. Rotation is small: $\frac{\|\boldsymbol{X}-\boldsymbol{I}_{(M+P),M}\|_F}{M} = 0.038, \boldsymbol{X} \approx \boldsymbol{I}_{(M+P),M}$

Numerical results: moderate Re number cavity



- Figures show pressure cross PSD of of $p(\boldsymbol{x}_1, t)$ and $p(\boldsymbol{x}_2, t)$ where $\boldsymbol{x}_1 = (2, -0.5)$, $\boldsymbol{x}_2 = (0, -0.5)$. Left: power; right: phase lag.

Power and phase lag at fundamental frequency, and first two super harmonics are predicted accurately using the fine-tuned ROM (Δ = stabilized ROM, $\square$ = DNS)

Future work (basis rotation)

- Application to *higher Reynolds number* problems.
- Extension of the proposed approach to problems with *generic nonlinearities*, where the ROM involves some form of hyper-reduction (e.g., DEIM, gappy POD).
- Extension of the method to *minimal-residual-based* nonlinear ROMs.
- Extension of the method to *predictive applications*, e.g., problems with varying Reynolds number and/or Mach number.
- Selecting different *goal-oriented* objectives and constraints in our optimization problem:

$$\begin{aligned} &\text{minimize}_{\boldsymbol{X}\in\mathcal{V}_{(M+P),M}} \quad f(\boldsymbol{X}) \\ &\text{subject to} \quad g(\boldsymbol{X},\boldsymbol{L}) = 0 \end{aligned}$$

e.g.,

- Maximize parametric robustness:

$$f = \sum_{i=1}^{k} \beta_i \|\boldsymbol{U}^*(\mu_i)\boldsymbol{X} - \boldsymbol{U}^*(\mu_i)\|_F.$$

- ODE constraints: $g = \|\boldsymbol{a}(t) - \boldsymbol{a}^*(t)\|$.

Outline

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. **Summary/perspectives.**
7. Ongoing/future work.

Summary/perspectives

	Continuous Projection	Discrete Projection
Change projection (*a priori*)	**Energy inner products** (Rowley *et al.*, Serre *et al.*, IKT *et al.*)	Energy inner products (Rowley, *et al.*), Petrov-Galerkin Projection (Carlberg *et al.*)
Change ROM equations (*a posteriori*)	Linear/nonlinear turbulence modeling (Iliescu, Borggaard, Xie, Wang, …)	**Eigenvalue reassignment** (IKT *et al.*)
Change ROM basis (*a posteriori*)	**Basis rotation** (Balajewicz, IKT, *et al.*)	Optimization-based right basis modification (Amsallem *et al.*)

- **Energy inner product + continuous projection**
  - *Pros*: consistent, inner product known analytically in closed form.
  - *Cons*: inner product is problem specific, expensive for nonlinear problems, BC implementation important for stability.
- **Eigenvalue reassignment**
  - *Pros*: black-box, easy-to-solve offline optimization problem
  - *Cons*: inconsistent, work required to extend to nonlinear and predictive problems.
- **Basis rotation**
  - *Pros*: black-box, consistent, works with *any* basis, applicable to nonlinear flow problems.
  - *Cons*: work required to extend to generic nonlinear PDEs.

Outline

1. Motivation.
2. Projection-based model order reduction.
3. Targeted application: compressible cavity flow.
4. Approaches for building *a priori* stable ROMs.
   (a) Energy-stable linearized compressible flow ROMs.
   (b) Energy-stable nonlinear compressible flow ROMs.
5. Approaches for stabilizing *a posteriori* unstable ROMs.
   (a) Eigenvalue reassignment (Linear Time-Invariant systems).
   (b) Basis rotation (nonlinear compressible flow).
6. Summary/perspectives.
7. **Ongoing/future work.**

Ongoing/future work

- ***Extension*** of ***basis rotation approach*** to problems with generic nonlinearities and LSPG ROMs [with Maciej Balajewicz (UIUC)].

Ongoing/future work

- ***Extension*** of ***basis rotation approach*** to problems with generic nonlinearities and LSPG ROMs [with Maciej Balajewicz (UIUC)].
- ***Implementation*** of Least-Squares Petrov-Galerkin (LSPG) ROMs within SPARC finite volume flow solver and Albany* open-source multi-physics code [with Jeff Fike (SNL), Kevin Carlberg (SNL)].

* https://github.com/gahansen/Albany.

Ongoing/future work

- ***Extension*** of ***basis rotation approach*** to problems with generic nonlinearities and LSPG ROMs [with Maciej Balajewicz (UIUC)].
- ***Implementation*** of Least-Squares Petrov-Galerkin (LSPG) ROMs within SPARC finite volume flow solver and Albany* open-source multi-physics code [with Jeff Fike (SNL), Kevin Carlberg (SNL)].
- Introduction of ***energy-based constraints*** (e.g., Clausius-Duhem inequality) into ROMs [with Kevin Carlberg (SNL), Danielle Maddix (Stanford), Margot Gerritsen (Stanford)].

* https://github.com/gahansen/Albany.

Ongoing/future work

- ***Extension*** of ***basis rotation approach*** to problems with generic nonlinearities and LSPG ROMs [with Maciej Balajewicz (UIUC)].
- ***Implementation*** of Least-Squares Petrov-Galerkin (LSPG) ROMs within SPARC finite volume flow solver and Albany* open-source multi-physics code [with Jeff Fike (SNL), Kevin Carlberg (SNL)].
- Introduction of ***energy-based constraints*** (e.g., Clausius-Duhem inequality) into ROMs [with Kevin Carlberg (SNL), Danielle Maddix (Stanford), Margot Gerritsen (Stanford)].
- Extension of ***filtering*** and ***LES-ROMs*** to compressible flow problems [with Traian Iliescu (VT), Xuping Xie (VT), Kevin Carlberg (SNL)].

* https://github.com/gahansen/Albany.

Ongoing/future work

- ***Extension*** of ***basis rotation approach*** to problems with generic nonlinearities and LSPG ROMs [with Maciej Balajewicz (UIUC)].
- ***Implementation*** of Least-Squares Petrov-Galerkin (LSPG) ROMs within SPARC finite volume flow solver and Albany* open-source multi-physics code [with Jeff Fike (SNL), Kevin Carlberg (SNL)].
- Introduction of ***energy-based constraints*** (e.g., Clausius-Duhem inequality) into ROMs [with Kevin Carlberg (SNL), Danielle Maddix (Stanford), Margot Gerritsen (Stanford)].
- Extension of ***filtering*** and ***LES-ROMs*** to compressible flow problems [with Traian Iliescu (VT), Xuping Xie (VT), Kevin Carlberg (SNL)].
- ***Improvement*** of previous techniques for ***time-predictive*** problems.

* https://github.com/gahansen/Albany.

Ongoing/future work

- ***Extension*** of ***basis rotation approach*** to problems with generic nonlinearities and LSPG ROMs [with Maciej Balajewicz (UIUC)].
- ***Implementation*** of Least-Squares Petrov-Galerkin (LSPG) ROMs within SPARC finite volume flow solver and Albany* open-source multi-physics code [with Jeff Fike (SNL), Kevin Carlberg (SNL)].
- Introduction of ***energy-based constraints*** (e.g., Clausius-Duhem inequality) into ROMs [with Kevin Carlberg (SNL), Danielle Maddix (Stanford), Margot Gerritsen (Stanford)].
- Extension of ***filtering*** and ***LES-ROMs*** to compressible flow problems [with Traian Iliescu (VT), Xuping Xie (VT), Kevin Carlberg (SNL)].
- ***Improvement*** of previous techniques for ***time-predictive*** problems.
- ***Hybrid ROM-FOMs*** via domain-decomposition-based coupling methods.
  - A. Mota, **I. Tezaur**, C. Alleman. "The Schwarz alternating method in solid mechanics", *Comput. Meth. Appl. Mech. Engng.* 319 (2017), 19-51.

* https://github.com/gahansen/Albany.

Ongoing/future work

- ***Extension*** of ***basis rotation approach*** to problems with generic nonlinearities and LSPG ROMs [with Maciej Balajewicz (UIUC)].
- ***Implementation*** of Least-Squares Petrov-Galerkin (LSPG) ROMs within SPARC finite volume flow solver and Albany* open-source multi-physics code [with Jeff Fike (SNL), Kevin Carlberg (SNL)].
- Introduction of ***energy-based constraints*** (e.g., Clausius-Duhem inequality) into ROMs [with Kevin Carlberg (SNL), Danielle Maddix (Stanford), Margot Gerritsen (Stanford)].
- Extension of ***filtering*** and ***LES-ROMs*** to compressible flow problems [with Traian Iliescu (VT), Xuping Xie (VT), Kevin Carlberg (SNL)].
- ***Improvement*** of previous techniques for ***time-predictive*** problems.
- ***Hybrid ROM-FOMs*** via domain-decomposition-based coupling methods.
  - A. Mota, **I. Tezaur**, C. Alleman. "The Schwarz alternating method in solid mechanics", *Comput. Meth. Appl. Mech. Engng.* 319 (2017), 19-51.

More info: www.sandia.gov/~ikalash

Thank you!

* https://github.com/gahansen/Albany.

References

- D. Amsallem, C. Farhat. Stabilization of projection-based reduced order models. IJNME (2011).
- M. Balajewicz, E. Dowell. Stabilization of projection-based reduced order models of the Navier-Stokes equation. *Nonlinear Dynamics* **70** (2), 1619-1632 (2012).
- M. Balajewicz, **I. Tezaur**, E. Dowell. Minimal subspace rotation on the Stiefel manifold for stabilization and enhancement of projection-based reduced order models for the compressible Navier-Stokes equations. *JCP* **321** 224-241 (2016).
- M. Barone, **I. Kalashnikova**, D. Segalman, H. Thornquist. Stable Galerkin reduced order models for linearized compressible flow. *JCP* **228** (6) 1932-1946 (2009).
- K. Carlberg, C. Farhat, J. Cortial, D. Amsallem. The GNAT method for nonlinear modele reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *JCP* **242** 623-647 (2013).
- **I. Kalashnikova**, M.F. Barone. On the Stability and Convergence of a Galerkin Reduced Order Model (ROM) of Compressible Flow with Solid Wall and Far-Field Boundary Treatment. *IJNME* **83** 1345-1375 (2010).
- **I. Kalashnikova**, B.G. van Bloemen Waanders, S. Arunajatesan, M.F. Barone. Stabilization of Projection-Based Reduced Order Models for Linear Time-Invariant Systems via Optimization-Based Eigenvalue Reassignment. *Comput. Meth. Appl. Mech. Engng.* **272** (2014) 251-270.
- **I. Kalashnikova**, S. Arunajatesan, M. Barone, B. van Bloemen Waanders, J. Fike. Reduced order modeling for prediction and control of large-scale systems. *Sandia Tech Report* (2014).
- C. Rowley, T. Colonius, R. Murray. Model reduction for compressible flows using POD and Galerkin projection. *Physica D* **189** (1) 115-129 (2004).
- G. Serre, P. Lafon, X. Gloerfelt, C. Bailly. Reliable reduced order models for time-dependent linearized Euler equations. *JCP* **231** (15) 5176-5194 (2012).

Appendix: Connection to Lyapunov stability

$$\dot{\boldsymbol{x}}_N = \boldsymbol{f}_N(\boldsymbol{x}_N),\ \ \boldsymbol{x}_N \in \mathbb{R}^N$$

- **Lyapunov stability**: if there exists a Lyapunov function V such that
  - $V > 0$ (positive definite), and
  - $\frac{dV}{dt} = \frac{dV}{d\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x}) \leq 0$

  in $B_r(\boldsymbol{x}_s)$, then $\boldsymbol{x}_s$ is ***locally stable in the sense of Lyapunov***.

- **Energy stability**: Let $E_N \equiv \frac{1}{2}||\boldsymbol{x}_N||^2$ denote the system energy. If

$$\frac{dE_N}{dt} \leq 0$$

is ***energy-stable***.

Remark: the system energy E_N satisfies the definition of a Lyapunov function!

Appendix: Symmetry (continuous) vs. Lyapunov (discrete) inner product

Symmetry Inner Product (Continuous)

$$(\boldsymbol{q}_1, \boldsymbol{q}_2)_{\boldsymbol{H}} \equiv \int_{\Omega} \boldsymbol{q}_1 \boldsymbol{H} \boldsymbol{q}_2 d\Omega$$

- For linear systems:

$$\dot{\boldsymbol{q}}' + \boldsymbol{A}_i(\overline{\boldsymbol{q}}) \frac{\partial \boldsymbol{q}'}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\boldsymbol{K}_{ij}(\overline{\boldsymbol{q}}) \frac{\partial \boldsymbol{q}'}{\partial x_i}\right] = \boldsymbol{0}$$

- Defined for unstable systems, but stability of ROM not guaranteed.
- Induced by Lyapunov function for system.
- Equation-specific ($\Longrightarrow$ embedded algorithm).
- Known analytically in closed form.

Lyapunov Inner Product (Discrete)

$$(\boldsymbol{x}_1, \boldsymbol{x}_2)_{\boldsymbol{P}} \equiv \boldsymbol{x}_1^T \boldsymbol{P} \boldsymbol{x}_2$$

- For linear systems:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}$$

- Undefined for unstable systems.
- Induced by Lyapunov function for system.
- Black-box.

Intractable for large problems!

- Computed numerically by solving Lyapunov equation ($O(N^3)$ ops).

Appendix: Accounting for modal truncation

Stabilization algorithm: returns stabilizing rotation matrix $\boldsymbol{X}$.

Inputs: Initial guess $\eta^{(0)} = \mathrm{tr}(L(1:n, 1:n))$ ($\boldsymbol{X} = \boldsymbol{I}_{(n+p)\times n}$), ROM size n and $p \geq 1$, ROM matrices associated with the first $n+p$ most energetic POD modes, convergence tolerance TOL, maximum number of iterations k_{max}.

for $k = 0, \cdots, k_{max}$

Solve constrained optimization problem on Stiefel manifold:

$$\begin{aligned} \underset{\boldsymbol{X}^{(k)} \in \mathcal{V}_{(n+p),n}}{\text{minimize}} \quad & -\mathrm{tr}\left(\boldsymbol{X}^{(k)\mathrm{T}} \boldsymbol{I}_{(n+p)\times n}\right) \\ \text{subject to} \quad & \mathrm{tr}(\boldsymbol{X}^{(k)\mathrm{T}} \boldsymbol{L} \boldsymbol{X}^{(k)}) = \eta^{(k)}. \end{aligned}$$

Construct new Galerkin matrices using (4).

Integrate numerically new Galerkin system.

Calculate "modal energy" $E(t)^{(k)} = \sum_i^n (a(t)_i^{(k)})^2$.

Perform linear fit of temporal data $E(t)^{(k)} \approx c_1^{(k)} t + c_0^{(k)}$, where $c_1^{(k)}$ =energy growth.

Calculate ϵ such that $c_1^{(k)}(\epsilon) = 0$ (no energy growth) using root-finding algorithm.

Perform update $\eta^{(k+1)} = \eta^{(k)} + \epsilon$.

if $||c_1^{(k)}|| < TOL$

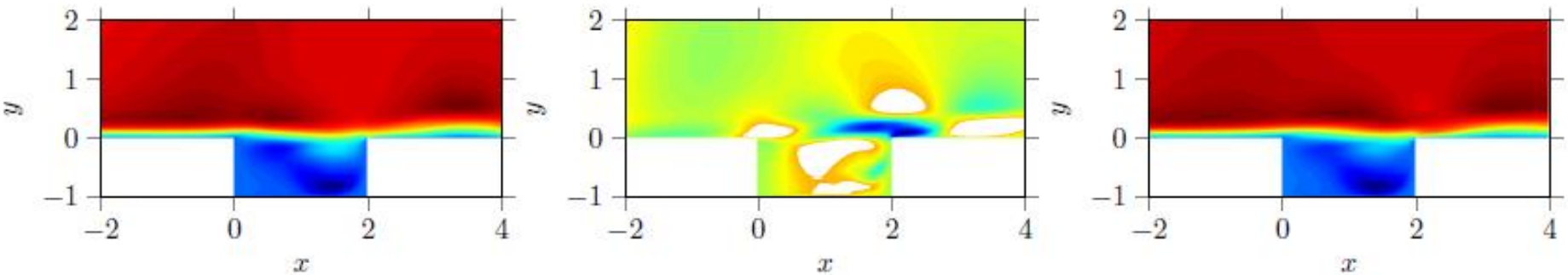
$\boldsymbol{X} := \boldsymbol{X}^{(k)}$.

terminate the algorithm.

end

end

Appendix: Numerical results: low Re number cavity



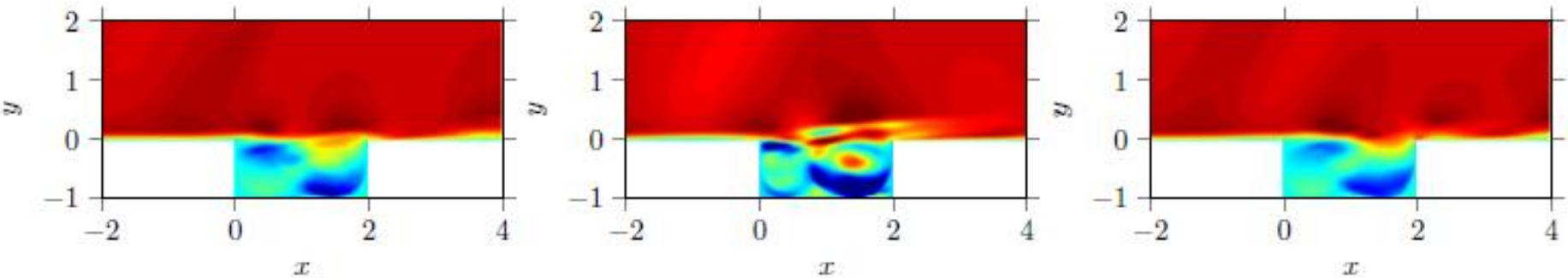
DNS

Standard ROM
$(M = 4)$

Stabilized ROM
$(M = P = 4)$

- u-velocity at time of final snapshot for low Reynolds number cavity. Left: FOM, middle: standard ROM, right: stabilized ROM.

Appendix: Numerical results: moderate Re number cavity



DNS

Standard ROM
($M = 20$)

Stabilized ROM
($M = p = 20$)

- u-velocity at time of final snapshot for low Reynolds number cavity. Left: FOM, middle: standard ROM, right: stabilized ROM.

Appendix: CPU times

- Tables gives CPU times (CPU-hours) for offline and online computations

	Procedure	Low Re Cavity	Moderate Re Cavity
	FOM # of DOF	288,250	243,750
offline	Time-integration of FOM	72 hrs	179 hrs
offline	Basis construction (size $M + P$ ROM)	0.88 hrs	3.44 hrs
offline	Galerkin projection (size $M + P$ ROM)	5.44 hrs	14.8 hrs
offline	Stabilization	14 sec	170 sec
	ROM # of DOF	4	20
online	Time-integration of ROM	0.16 sec	0.83 sec
online	Online computational speed-up	1.6e6	7.8e5

- Stabilization is *fast* (O(sec) or O(min)).
- Significant *online computational speed-up*!