Accelerating PDE-Constrained Optimization using Progressively-Constructed Reduced-Order Models

Matthew J. Zahr and Charbel Farhat

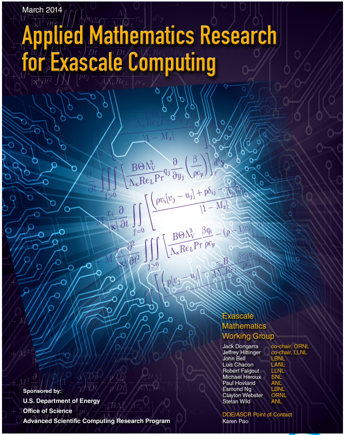
Institute for Computational and Mathematical Engineering
Farhat Research Group
Stanford University

ROM Workshop
Sandia National Laboratories
August 7, 2014

Outline

Scientific Grand Challenges

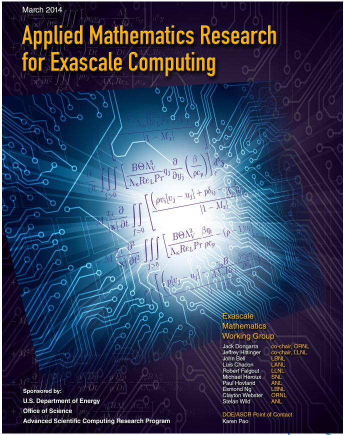
- Combustion
  - Design of next-generation engines
- Climate
  - "... estimate global temperature response to increases in greenhouse gases"
  - "quantify how the climate system would respond to an increase in temperature"
    - predict major climatic events
- Material
  - Artificial light harvesting
  - Bridge between atomistic and macroscale



Exascale as Enabling Technology

Scientific Grand Challenges: Combustion

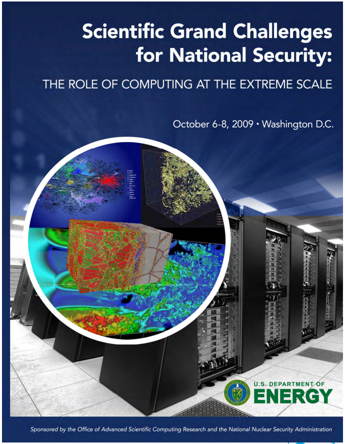
- Goal: Design of next-generation engines
  - High-efficiency, low-emission, biodiesel
- Computational model
  - High-pressure turbulent reacting flow
  - Complex geometry
  - High-pressure/velocity fuel injection
  - Intermediary particulate soot
- **Uncertainty Quantification** (UQ)
- **Design optimization**
  - Multiobjective: fuel efficiency and emissions
  - Multi-point: design for multiple operating points
  - Optimization under uncertainty



Many-Query Analyses and Grand Challenges

Optimization and UQ

- Multiphysics simulations
  - Example: aerodynamic optimization
    - Frame design
    - Noise mitigation
    - Jet turbine design
- Material science
- Computational chemistry
- Nonproliferation
- UQ and error analysis
  - Climate modeling



Difficulty of Many-Query Analyses: Optimization

Difficulty of Many-Query Analyses: Optimization

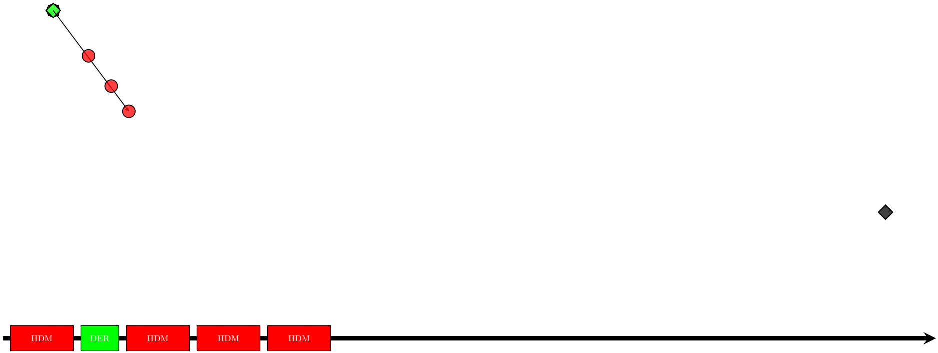
Difficulty of Many-Query Analyses: Optimization

Difficulty of Many-Query Analyses: Optimization

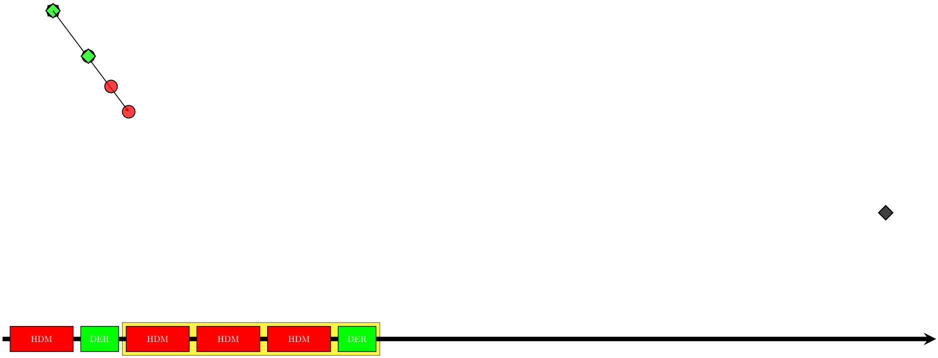
Difficulty of Many-Query Analyses: Optimization

Difficulty of Many-Query Analyses: Optimization

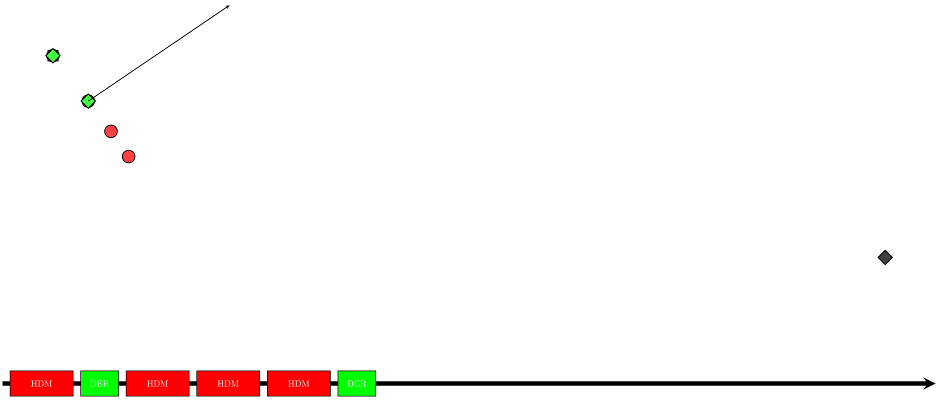
Difficulty of Many-Query Analyses: Optimization



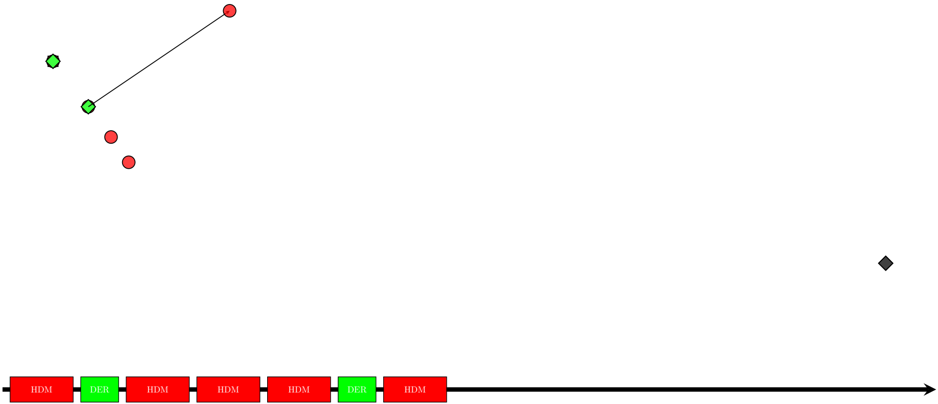
Difficulty of Many-Query Analyses: Optimization



Difficulty of Many-Query Analyses: Optimization

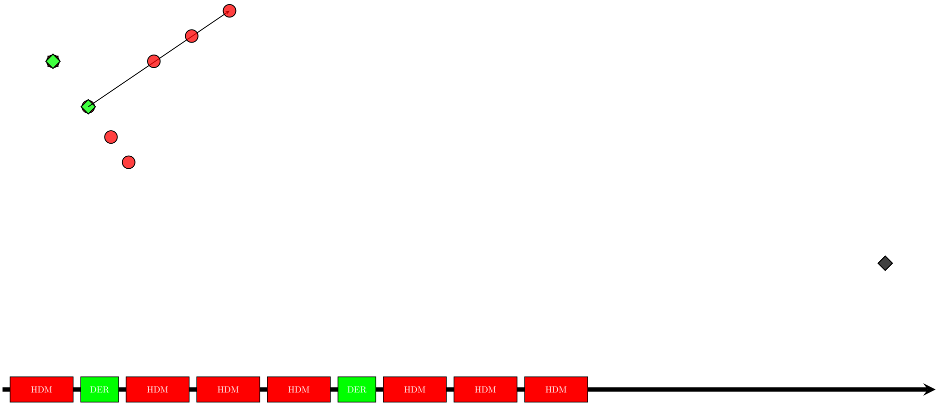


Difficulty of Many-Query Analyses: Optimization

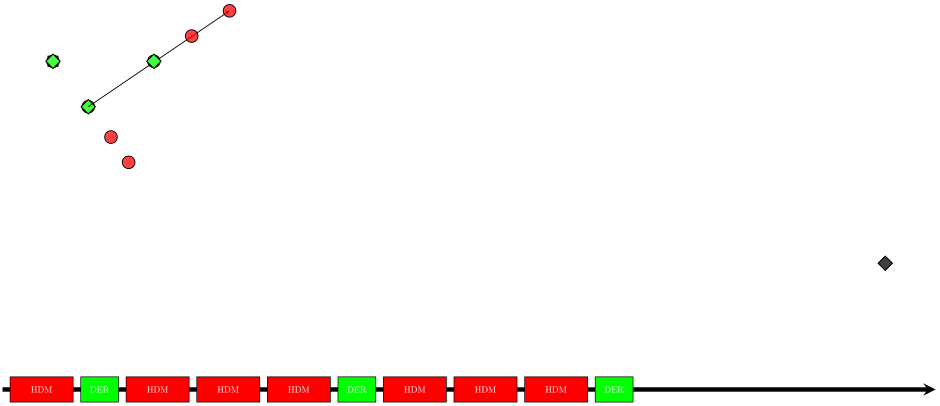


Difficulty of Many-Query Analyses: Optimization

Difficulty of Many-Query Analyses: Optimization

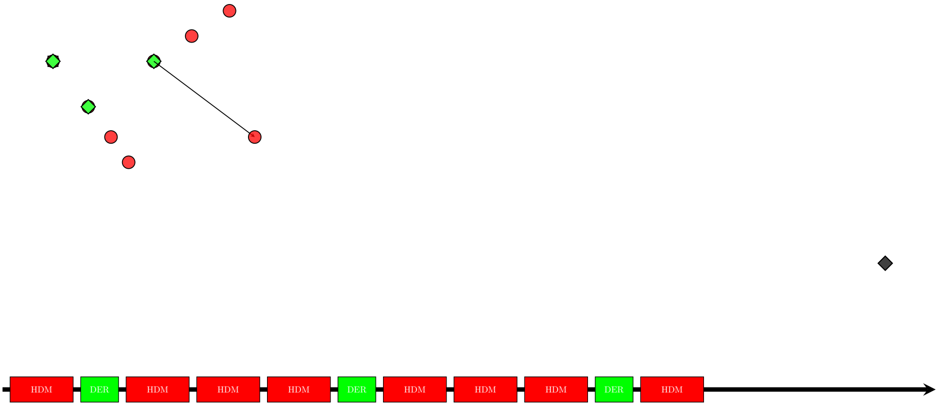


Difficulty of Many-Query Analyses: Optimization



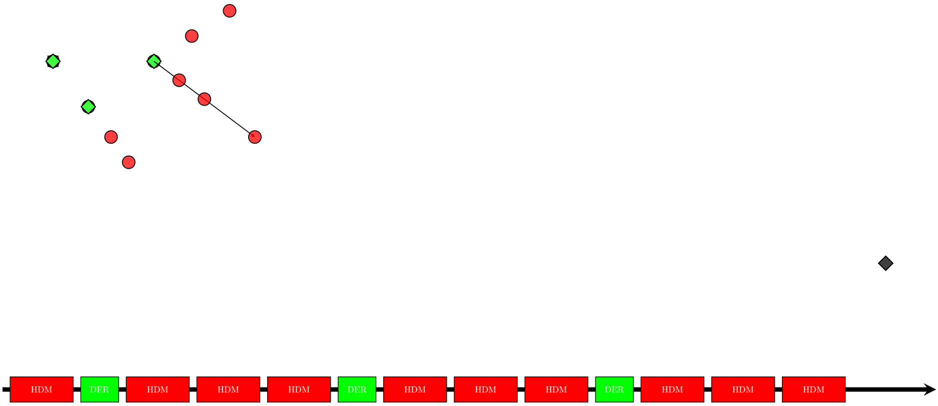
Difficulty of Many-Query Analyses: Optimization

Difficulty of Many-Query Analyses: Optimization

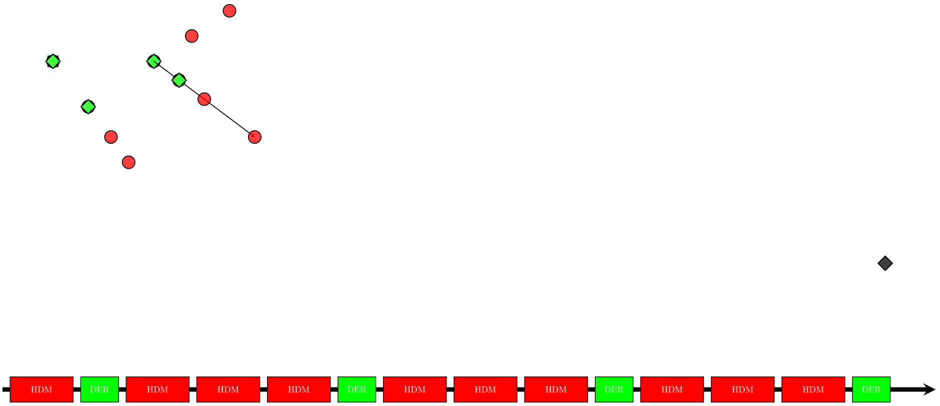


Difficulty of Many-Query Analyses: Optimization

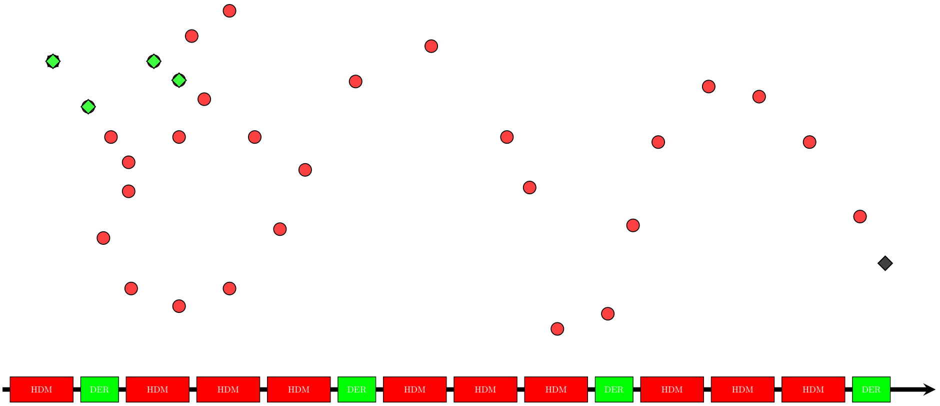
Difficulty of Many-Query Analyses: Optimization



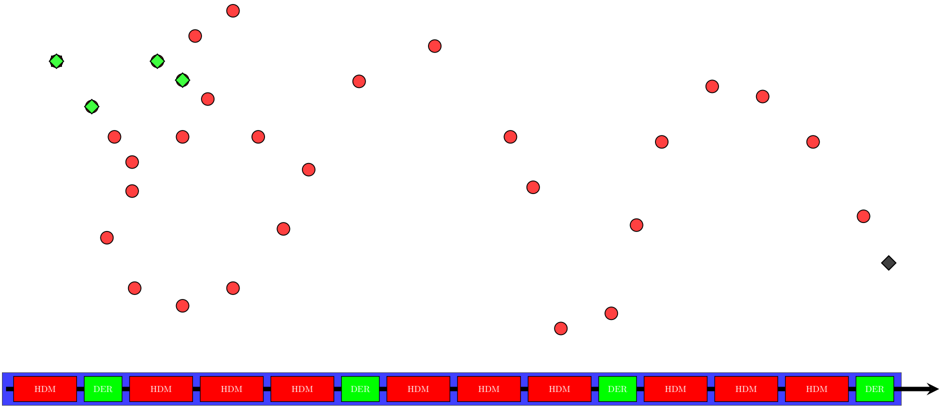
Difficulty of Many-Query Analyses: Optimization



Difficulty of Many-Query Analyses: Optimization



Difficulty of Many-Query Analyses: Optimization



Reduced-Order Models (ROMs)

ROMs and Exascale

- Very similar goals
  - enable computational analysis, design, UQ, control of highly-complex systems not feasible with existing tools/technology
  - use computational tool to solve relevant scientific and engineering problems
- Pursue goals with opposite approaches
  - ROMs: systematic dimensionality reduction while preserving fidelity to drastically reduce cost of simulation
  - Exascale: Leverage $\mathcal{O}(10^{18})$ FLOPS to enable direct simulation of high-fidelity systems
- *Not mutually exclusive*!

Reduced-Order Models (ROMs)

ROMs as Enabling Technology

- Many-query analyses
    - Optimization: design, control
        - Single objective, single-point
        - Multiobjective, multi-point
    - Uncertainty Quantification
        - Optimization under uncertainty
- Real-time analysis
    - Model Predictive Control (MPC)

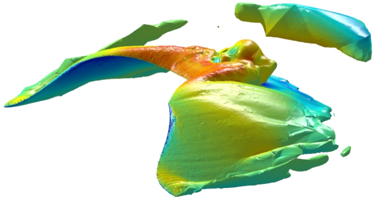
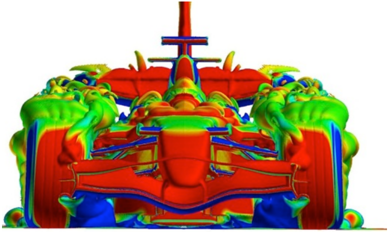
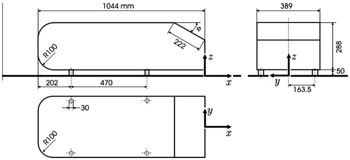


Figure: Flapping Wing (Persson et al., 2012)

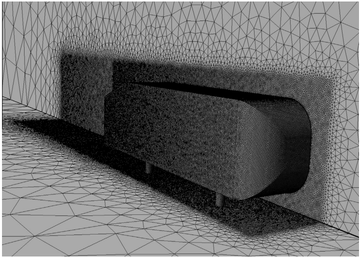


Application I: Compressible, Turbulent Flow over Vehicle

- Benchmark in automotive industry
- Mesh
  - 2,890,434 vertices
  - 17,017,090 tetra
  - 17,342,604 DOF
- CFD
  - Compressible Navier-Stokes
  - DES + Wall func
- Single forward simulation
  - $\approx$ 0.5 day on 512 cores
- Desired: shape optimization
  - unsteady effects
  - minimize average drag



(a) Ahmed Body: Geometry (Ahmed et al, 1984)



(b) Ahmed Body: Mesh (Carlberg et al, 2011)

Application II: Turbulent Flow over Flapping Wing

- Biologically-inspired flight
  - Micro aerial vehicles
- Mesh
  - 43,000 vertices
  - 231,000 tetra ($p = 3$)
  - 2,310,000 DOF
- CFD
  - Compressible Navier-Stokes
  - Discontinuous Galerkin
- Desired: shape optimization + control
  - unsteady effects
  - maximize thrust

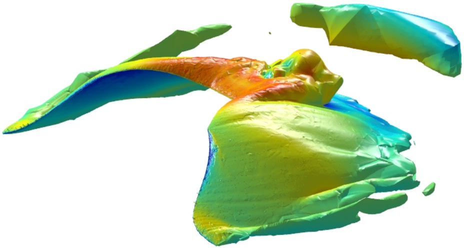
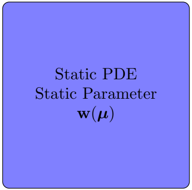


Figure: Flapping Wing (Persson et al., 2012)

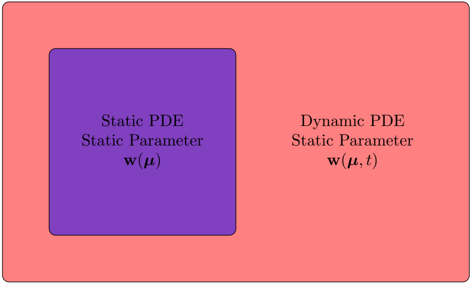
Outline

1. Motivation
2. PDE-Constrained Optimization
3. Reduced-Order Models
   - Construction of Bases
   - Speedup Potential
4. ROM-Constrained Optimization
   - Reduced Sensitivities
   - Training
5. Numerical Experiments
   - Rocket Nozzle Design
   - Airfoil Design
6. Conclusion
   - Overview
   - Outlook
   - Future Work

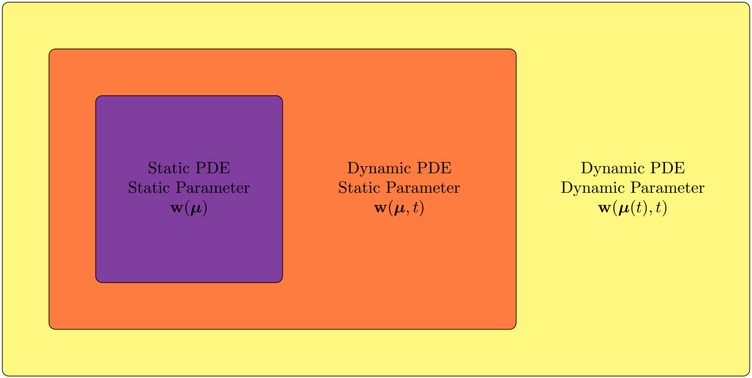
Hierarchy of PDE-Constrained Optimization



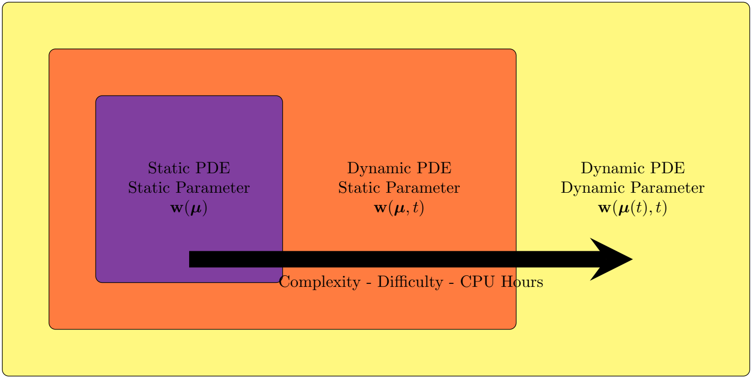
Hierarchy of PDE-Constrained Optimization



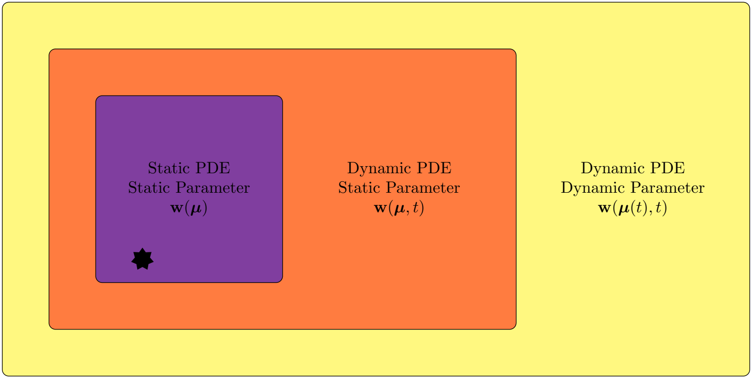
Hierarchy of PDE-Constrained Optimization



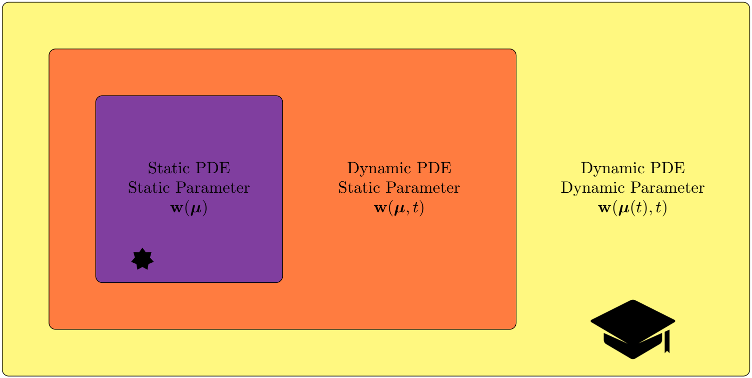
Hierarchy of PDE-Constrained Optimization



Hierarchy of PDE-Constrained Optimization



Hierarchy of PDE-Constrained Optimization



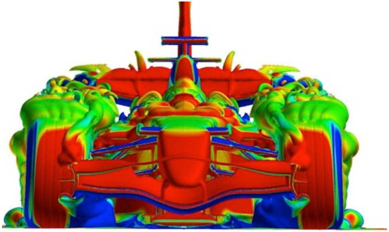
Problem Formulation

Goal: Rapidly solve PDE-constrained optimization problems of the form

$$\begin{aligned} \underset{\mathbf{w}\in\mathbb{R}^N,\ \boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\mathbf{w},\boldsymbol{\mu}) \\ \text{subject to} \quad & \mathbf{R}(\mathbf{w},\boldsymbol{\mu}) = 0 \end{aligned}$$

Discretize-then-optimize

where $\mathbf{R} : \mathbb{R}^N \times \mathbb{R}^p \rightarrow \mathbb{R}^N$ is the discretized (steady, nonlinear) PDE, $\mathbf{w}$ is the PDE state vector, $\boldsymbol{\mu}$ is the vector of parameters, and N is **assumed to be very large**.



Two Approaches

Simultaneous Analysis and Design (SAND)

$$\begin{aligned} \underset{\mathbf{w}\in\mathbb{R}^N,\ \boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\mathbf{w}, \boldsymbol{\mu}) \\ \text{subject to} \quad & \mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) = 0 \end{aligned}$$

- Treat *state* and *parameters* as optimization variables

Nested Analysis and Design (NAND)

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad f(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

- $\mathbf{w} = \mathbf{w}(\boldsymbol{\mu})$ through $\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) = 0$
- Treat *parameters* as only optimization variables
- Enforce nonlinear equality constraint at every iteration

(Gunzburger, 2003), (Hinze et al., 2009)

Sensitivity Derivation

- Consider some functional $\mathcal{F}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu})$ to be differentiated (i.e. objective function or constraint)
  - $\dfrac{\mathrm{d}\mathcal{F}}{\mathrm{d}\boldsymbol{\mu}} = \dfrac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}} + \dfrac{\partial \mathcal{F}}{\partial \mathbf{w}} \dfrac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$

Sensitivity Derivation

- Consider some functional $\mathcal{F}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu})$ to be differentiated (i.e. objective function or constraint)
  - $$\frac{\mathrm{d}\mathcal{F}}{\mathrm{d}\boldsymbol{\mu}} = \frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathcal{F}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$$
- $\mathbf{R}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0$ for all $\boldsymbol{\mu}$ $\implies \dfrac{\mathrm{d}\mathbf{R}}{\mathrm{d}\boldsymbol{\mu}} = 0 = \dfrac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} + \dfrac{\partial \mathbf{R}}{\partial \mathbf{w}} \dfrac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$
  - $$\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

Sensitivity Derivation

- Consider some functional $\mathcal{F}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu})$ to be differentiated (i.e. objective function or constraint)
  - $\frac{\mathrm{d}\mathcal{F}}{\mathrm{d}\boldsymbol{\mu}} = \frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathcal{F}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$
- $\mathbf{R}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0$ for all $\boldsymbol{\mu} \implies \frac{\mathrm{d}\mathbf{R}}{\mathrm{d}\boldsymbol{\mu}} = 0 = \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$
  - $\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$

Gradient of Functional

$$\frac{\mathrm{d}\mathcal{F}}{\mathrm{d}\boldsymbol{\mu}} = \frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}} - \frac{\partial \mathcal{F}}{\partial \mathbf{w}} \left(\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} \right) = \frac{\partial \mathcal{F}}{\partial \boldsymbol{\mu}} - \left(\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-T} \frac{\partial \mathcal{F}}{\partial \mathbf{w}}^{T} \right)^{T} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

Summary: NAND formulation, Sensitivity Approach

Nested Analysis and Design (NAND)

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad f(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

- $\mathbf{w} = \mathbf{w}(\boldsymbol{\mu})$ through $\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) = 0$

Gradient of Objective Function (Sensitivity Approach)

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\mu}}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = \frac{\partial f}{\partial \boldsymbol{\mu}} + \frac{\partial f}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$$

- $\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = \left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$ from $\frac{\mathrm{d}\mathbf{R}}{\mathrm{d}\boldsymbol{\mu}} = \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = 0$

Outline

Reduced-Order Model

- Model Order Reduction (MOR) assumption: *state vector lies in low-dimensional affine subspace*

$$\mathbf{w} \approx \mathbf{w}_r = \bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} \approx \frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}} = \mathbf{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

where $\mathbf{y} \in \mathbb{R}^n$ are the reduced coordinates of $\mathbf{w}_r$ in the basis $\mathbf{\Phi} \in \mathbb{R}^{N \times n}$, and $n \ll N$

Reduced-Order Model

- Model Order Reduction (MOR) assumption: *state vector lies in low-dimensional affine subspace*

$$\mathbf{w} \approx \mathbf{w}_r = \bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} \approx \frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

where $\mathbf{y} \in \mathbb{R}^n$ are the reduced coordinates of $\mathbf{w}_r$ in the basis $\boldsymbol{\Phi} \in \mathbb{R}^{N \times n}$, and $n \ll N$

- Substitute assumption into High-Dimensional Model (HDM), $\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) = 0$

$$\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) \approx 0$$

Reduced-Order Model

- Model Order Reduction (MOR) assumption: *state vector lies in low-dimensional affine subspace*

$$\mathbf{w} \approx \mathbf{w}_r = \bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} \approx \frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}} = \mathbf{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

  where $\mathbf{y} \in \mathbb{R}^n$ are the reduced coordinates of $\mathbf{w}_r$ in the basis $\mathbf{\Phi} \in \mathbb{R}^{N \times n}$, and $n \ll N$

- Substitute assumption into High-Dimensional Model (HDM), $\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) = 0$

$$\mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y}, \boldsymbol{\mu}) \approx 0$$

- Require projection of residual in low-dimensional *left subspace*, with basis $\mathbf{\Psi} \in \mathbb{R}^{N \times n}$ to be zero

$$\boxed{\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = \mathbf{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0}$$

Reduced Optimization Problem

- Reduce-then-optimize[1]

ROM-Constrained Optimization - NAND Formulation

$$\underset{\boldsymbol{\mu} \in \mathbb{R}^p}{\text{minimize}} \quad f(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

- $\mathbf{y} = \mathbf{y}(\boldsymbol{\mu})$ through $\boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$

- Issues that must be considered
  - Construction of bases
  - Speedup potential
  - Reduced sensitivity derivation
  - Training

[1] (Manzoni, 2012)

Definition of $\mathbf{\Phi}$: Proper Orthogonal Decomposition[3]

- Recall MOR assumption

$$\mathbf{w} - \bar{\mathbf{w}} \approx \mathbf{\Phi} \mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} \approx \mathbf{\Phi} \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

- Implication: we desire

$$\{\mathbf{w}(\boldsymbol{\mu}) - \bar{\mathbf{w}}\} \bigcup \left\{ \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}) \right\} \subseteq \text{range } \mathbf{\Phi}$$

- Include translated state vectors *and* sensitivities as snapshots
- Previous work considering sensitivity snapshots [2]

[2] (Carlberg and Farhat, 2008), (Hay et al., 2009), (Carlberg and Farhat, 2011)
[3] (Sirovich, 1987)

Definition of $\boldsymbol{\Phi}$: Proper Orthogonal Decomposition

- Recall MOR assumption

$$\mathbf{w} - \bar{\mathbf{w}} \approx \boldsymbol{\Phi}\mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} \approx \boldsymbol{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

State-Sensitivity[4] POD

- Collect state and sensitivity snapshots by sampling HDM

$$\mathbf{X} = \begin{bmatrix} \mathbf{w}(\boldsymbol{\mu}_1) - \bar{\mathbf{w}} & \mathbf{w}(\boldsymbol{\mu}_2) - \bar{\mathbf{w}} & \cdots & \mathbf{w}(\boldsymbol{\mu}_n) - \bar{\mathbf{w}} \end{bmatrix}$$

$$\mathbf{Y} = \begin{bmatrix} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_1) & \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_2) & \cdots & \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_n) \end{bmatrix}$$

[4] (Washabaugh and Farhat, 2013),(Zahr and Farhat, 2014)

Definition of $\mathbf{\Phi}$: Proper Orthogonal Decomposition

- Recall MOR assumption

$$\mathbf{w} - \bar{\mathbf{w}} \approx \mathbf{\Phi}\mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} \approx \mathbf{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

State-Sensitivity[4] POD

- Collect state and sensitivity snapshots by sampling HDM

$$\mathbf{X} = \begin{bmatrix} \mathbf{w}(\boldsymbol{\mu}_1) - \bar{\mathbf{w}} & \mathbf{w}(\boldsymbol{\mu}_2) - \bar{\mathbf{w}} & \cdots & \mathbf{w}(\boldsymbol{\mu}_n) - \bar{\mathbf{w}} \end{bmatrix}$$
$$\mathbf{Y} = \begin{bmatrix} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_1) & \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_2) & \cdots & \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_n) \end{bmatrix}$$

- Use Proper Orthogonal Decomposition to generate reduced bases from each *individually*

$$\mathbf{\Phi}_{\mathbf{X}} = \mathrm{POD}(\mathbf{X})$$
$$\mathbf{\Phi}_{\mathbf{Y}} = \mathrm{POD}(\mathbf{Y})$$

[4](Washabaugh and Farhat, 2013),(Zahr and Farhat, 2014)

Definition of $\mathbf{\Phi}$: Proper Orthogonal Decomposition

- Recall MOR assumption

$$\mathbf{w} - \bar{\mathbf{w}} \approx \mathbf{\Phi}\mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} \approx \mathbf{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

State-Sensitivity[4] POD

- Collect state and sensitivity snapshots by sampling HDM

$$\mathbf{X} = \begin{bmatrix} \mathbf{w}(\boldsymbol{\mu}_1) - \bar{\mathbf{w}} & \mathbf{w}(\boldsymbol{\mu}_2) - \bar{\mathbf{w}} & \cdots & \mathbf{w}(\boldsymbol{\mu}_n) - \bar{\mathbf{w}} \end{bmatrix}$$
$$\mathbf{Y} = \begin{bmatrix} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_1) & \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_2) & \cdots & \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}_n) \end{bmatrix}$$

- Use Proper Orthogonal Decomposition to generate reduced bases from each *individually*

$$\mathbf{\Phi}_{\mathbf{X}} = \mathrm{POD}(\mathbf{X})$$
$$\mathbf{\Phi}_{\mathbf{Y}} = \mathrm{POD}(\mathbf{Y})$$

- Concatenate to get ROB

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_{\mathbf{X}} & \mathbf{\Phi}_{\mathbf{Y}} \end{bmatrix}$$

[4] (Washabaugh and Farhat, 2013),(Zahr and Farhat, 2014)

Definition of $\boldsymbol{\Psi}$: Minimum-Residual ROM

- ROM governing equation: $\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) \equiv \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$
- Standard options for choice of left basis $\boldsymbol{\Psi}$
  - $\boldsymbol{\Psi} = \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Galerkin
  - $\boldsymbol{\Psi} = \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Least-Squares Petrov-Galerkin (LSPG)[5,6]

[5] (Bui-Thanh et al., 2008)
[6] (Carlberg et al., 2011)
[7] (Fahl, 2001)

Definition of $\boldsymbol{\Psi}$: Minimum-Residual ROM

- ROM governing equation: $\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) \equiv \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$
- Standard options for choice of left basis $\boldsymbol{\Psi}$
  - $\boldsymbol{\Psi} = \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Galerkin
  - $\boldsymbol{\Psi} = \dfrac{\partial \mathbf{R}}{\partial \mathbf{w}} \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Least-Squares Petrov-Galerkin (LSPG)[5,6]

Minimum-Residual Property

A ROM possesses the minimum-residual property if $\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = 0$ is equivalent to the optimality condition of ($\Theta \succ 0$)

$$\underset{\mathbf{y} \in \mathbb{R}^n}{\text{minimize}} \quad ||\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})||_{\Theta}$$

[5] (Bui-Thanh et al., 2008)
[6] (Carlberg et al., 2011)
[7] (Fahl, 2001)

Definition of $\boldsymbol{\Psi}$: Minimum-Residual ROM

- ROM governing equation: $\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) \equiv \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$
- Standard options for choice of left basis $\boldsymbol{\Psi}$
  - $\boldsymbol{\Psi} = \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Galerkin
  - $\boldsymbol{\Psi} = \dfrac{\partial \mathbf{R}}{\partial \mathbf{w}} \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Least-Squares Petrov-Galerkin (LSPG)[5,6]

Minimum-Residual Property

A ROM possesses the minimum-residual property if $\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = 0$ is equivalent to the optimality condition of $(\Theta \succ 0)$

$$\underset{\mathbf{y} \in \mathbb{R}^n}{\text{minimize}} \quad ||\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})||_{\Theta}$$

- LSPG possesses minimum-residual property[6]

[5] (Bui-Thanh et al., 2008)
[6] (Carlberg et al., 2011)
[7] (Fahl, 2001)

Definition of $\boldsymbol{\Psi}$: Minimum-Residual ROM

- ROM governing equation: $\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) \equiv \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$
- Standard options for choice of left basis $\boldsymbol{\Psi}$
  - $\boldsymbol{\Psi} = \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Galerkin
  - $\boldsymbol{\Psi} = \dfrac{\partial \mathbf{R}}{\partial \mathbf{w}} \boldsymbol{\Phi} \quad \Longrightarrow \quad$ Least-Squares Petrov-Galerkin (LSPG)[5,6]

Minimum-Residual Property

A ROM possesses the minimum-residual property if $\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = 0$ is equivalent to the optimality condition of $(\Theta \succ 0)$

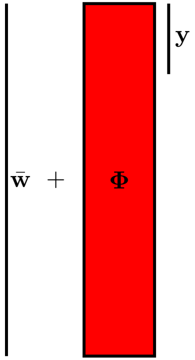
$$\underset{\mathbf{y} \in \mathbb{R}^n}{\text{minimize}} \quad ||\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})||_{\Theta}$$

- LSPG possesses minimum-residual property[6]
- Implications
  - Recover exact solution when basis not truncated (consistent[6])
  - Monotonic improvement of solution as basis size increases
  - Ensures sensitivity information in $\boldsymbol{\Phi}$ cannot degrade state approximation[7]

[5] (Bui-Thanh et al., 2008)
[6] (Carlberg et al., 2011)
[7] (Fahl, 2001)

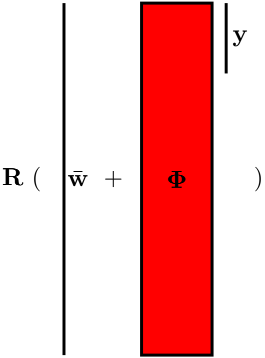
Nonlinear ROM Bottleneck

$$\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$$



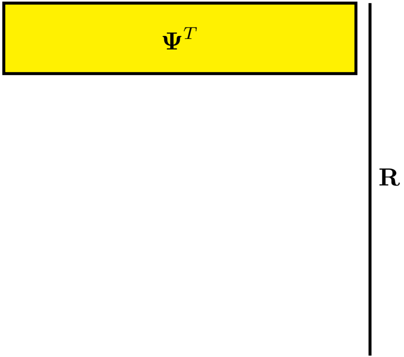
Nonlinear ROM Bottleneck

$$\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$$



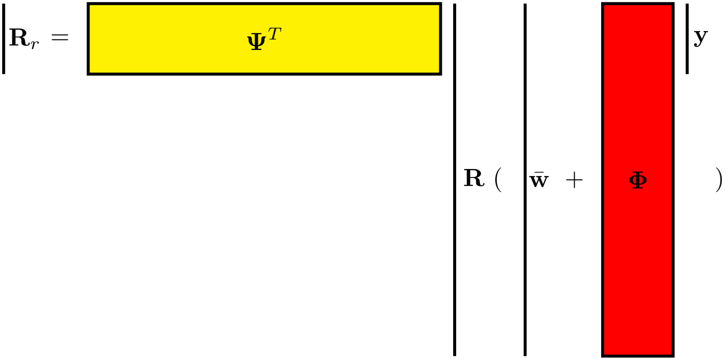
Nonlinear ROM Bottleneck

$$\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$$



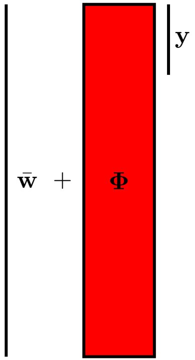
Nonlinear ROM Bottleneck

$$\mathbf{R}_r(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0$$



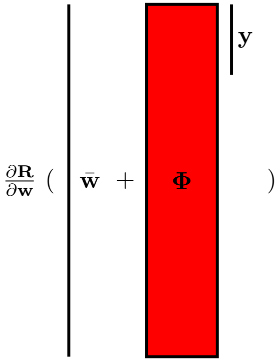
Nonlinear ROM Bottleneck

$$\frac{\partial \mathbf{R}_r}{\partial \mathbf{y}}(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \frac{\partial \mathbf{R}}{\partial \mathbf{w}}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})\boldsymbol{\Phi} = 0$$



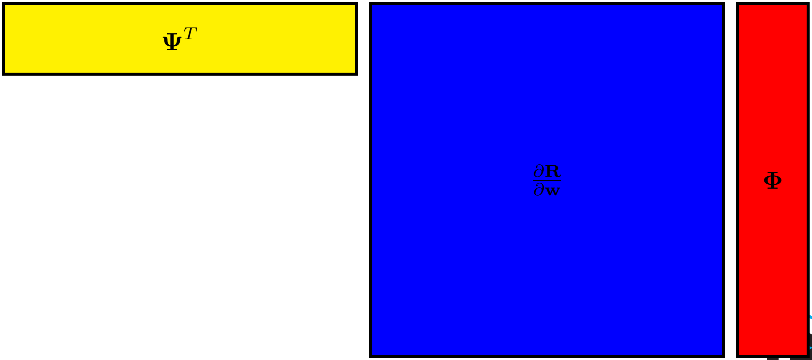
Nonlinear ROM Bottleneck

$$\frac{\partial \mathbf{R}_r}{\partial \mathbf{y}}(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \frac{\partial \mathbf{R}}{\partial \mathbf{w}}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})\boldsymbol{\Phi} = 0$$



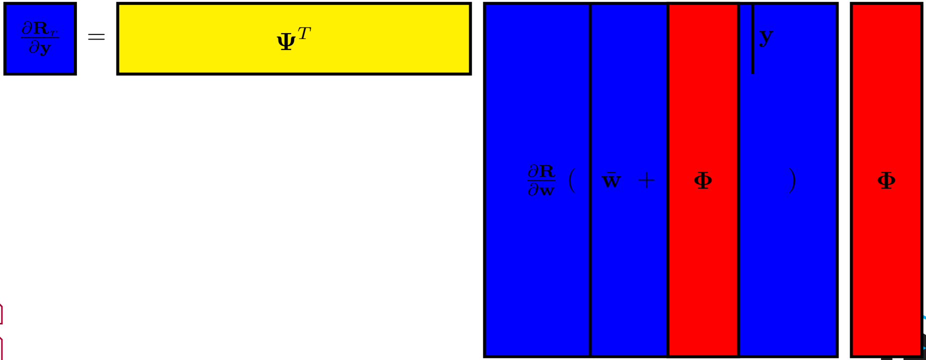
Nonlinear ROM Bottleneck

$$\frac{\partial \mathbf{R}_r}{\partial \mathbf{y}}(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \frac{\partial \mathbf{R}}{\partial \mathbf{w}}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})\boldsymbol{\Phi} = 0$$



Nonlinear ROM Bottleneck

$$\frac{\partial \mathbf{R}_r}{\partial \mathbf{y}}(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \frac{\partial \mathbf{R}}{\partial \mathbf{w}}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})\boldsymbol{\Phi} = 0$$



Hyperreduction

Several different forms of *hyperreduction* exist to alleviate bottleneck caused by nonlinear terms

- If nonlinearity polynomial, precompute tensorial coefficients
- Linearize (or “polynomialize”) about specific points in state space [8]
- Gappy POD to reconstruct nonlinear residual from a few entries [9]
    - Empirical Interpolation Method (EIM) [10]
    - Discrete Empirical Interpolation Method (DEIM) [11]
    - Gauss-Newton with Approximated Tensors (GNAT) [12]

[8] (Rewienski, 2003)
[9] (Everson and Sirovich, 1995)
[10] (Barrault et al., 2004)
[11] (Chaturantabut and Sorensen, 2010)
[12] (Carlberg et al., 2011),(Carlberg et al., 2013)

Hyperreduction: Gappy POD [13]

- Assume nonlinear terms (residual/Jacobian) lie in low-dimensional subspace

$$\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) \approx \boldsymbol{\Phi}_R \mathbf{r}(\mathbf{w}, \boldsymbol{\mu})$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{N \times n_R}$ and $\mathbf{r} : \mathbb{R}^N \times \mathbb{R}^p \to \mathbb{R}^{n_R}$ are the reduced coordinates; $n_R \ll N$

[13] (Everson and Sirovich, 1995),(Chaturantabut and Sorensen, 2010),(Carlberg et al., 2011)

Hyperreduction: Gappy POD [13]

- Assume nonlinear terms (residual/Jacobian) lie in low-dimensional subspace

$$\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) \approx \boldsymbol{\Phi}_R \mathbf{r}(\mathbf{w}, \boldsymbol{\mu})$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{N \times n_R}$ and $\mathbf{r} : \mathbb{R}^N \times \mathbb{R}^p \rightarrow \mathbb{R}^{n_R}$ are the reduced coordinates; $n_R \ll N$

- Determine $\mathbf{R}$ by solving *gappy* least-squares problem

$$\mathbf{r}(\mathbf{w}, \boldsymbol{\mu}) = \underset{\mathbf{a} \in \mathbb{R}^{n_R}}{\arg\min} ||\mathbf{Z}^T \boldsymbol{\Phi}_R \mathbf{a} - \mathbf{Z}^T \mathbf{R}(\mathbf{w}, \boldsymbol{\mu})||$$

where $\mathbf{Z}$ is a restriction operator

[13] (Everson and Sirovich, 1995),(Chaturantabut and Sorensen, 2010),(Carlberg et al., 2011)

Hyperreduction: Gappy POD [13]

- Assume nonlinear terms (residual/Jacobian) lie in low-dimensional subspace

$$\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) \approx \boldsymbol{\Phi}_R \mathbf{r}(\mathbf{w}, \boldsymbol{\mu})$$

  where $\boldsymbol{\Phi} \in \mathbb{R}^{N \times n_R}$ and $\mathbf{r} : \mathbb{R}^N \times \mathbb{R}^p \to \mathbb{R}^{n_R}$ are the reduced coordinates; $n_R \ll N$
- Determine $\mathbf{R}$ by solving *gappy* least-squares problem

$$\mathbf{r}(\mathbf{w}, \boldsymbol{\mu}) = \underset{\mathbf{a} \in \mathbb{R}^{n_R}}{\arg\min} ||\mathbf{Z}^T \boldsymbol{\Phi}_R \mathbf{a} - \mathbf{Z}^T \mathbf{R}(\mathbf{w}, \boldsymbol{\mu})||$$

  where $\mathbf{Z}$ is a restriction operator
- Analytical solution

$$\mathbf{r}(\mathbf{w}, \boldsymbol{\mu}) = \left(\mathbf{Z}^T \boldsymbol{\Phi}_R\right)^{\dagger} \left(\mathbf{Z}^T \mathbf{R}(\mathbf{w}, \boldsymbol{\mu})\right)$$

[13] (Everson and Sirovich, 1995),(Chaturantabut and Sorensen, 2010),(Carlberg et al., 2011)

Hyperreduction: Gappy POD [13]

- Assume nonlinear terms (residual/Jacobian) lie in low-dimensional subspace

$$\mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) \approx \boldsymbol{\Phi}_R \mathbf{r}(\mathbf{w}, \boldsymbol{\mu})$$

  where $\boldsymbol{\Phi} \in \mathbb{R}^{N \times n_R}$ and $\mathbf{r} : \mathbb{R}^N \times \mathbb{R}^p \to \mathbb{R}^{n_R}$ are the reduced coordinates; $n_R \ll N$

- Determine $\mathbf{R}$ by solving *gappy* least-squares problem

$$\mathbf{r}(\mathbf{w}, \boldsymbol{\mu}) = \underset{\mathbf{a} \in \mathbb{R}^{n_R}}{\arg\min} ||\mathbf{Z}^T \boldsymbol{\Phi}_R \mathbf{a} - \mathbf{Z}^T \mathbf{R}(\mathbf{w}, \boldsymbol{\mu})||$$

  where $\mathbf{Z}$ is a restriction operator

- Analytical solution

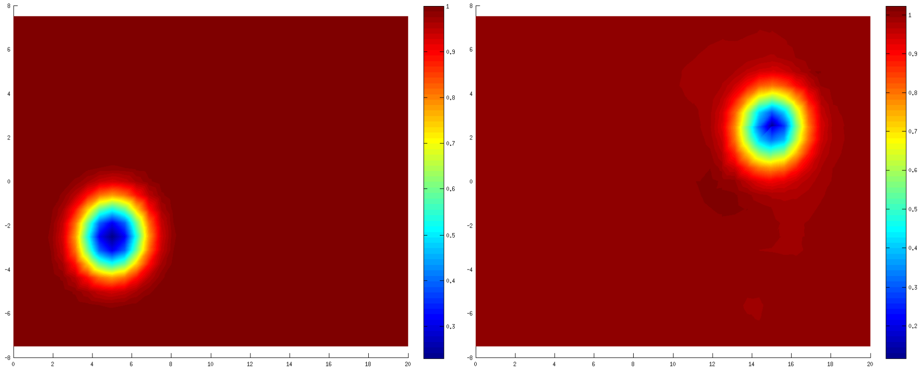
$$\mathbf{r}(\mathbf{w}, \boldsymbol{\mu}) = \left(\mathbf{Z}^T \boldsymbol{\Phi}_R\right)^{\dagger} \left(\mathbf{Z}^T \mathbf{R}(\mathbf{w}, \boldsymbol{\mu})\right)$$

- Hyperreduced model

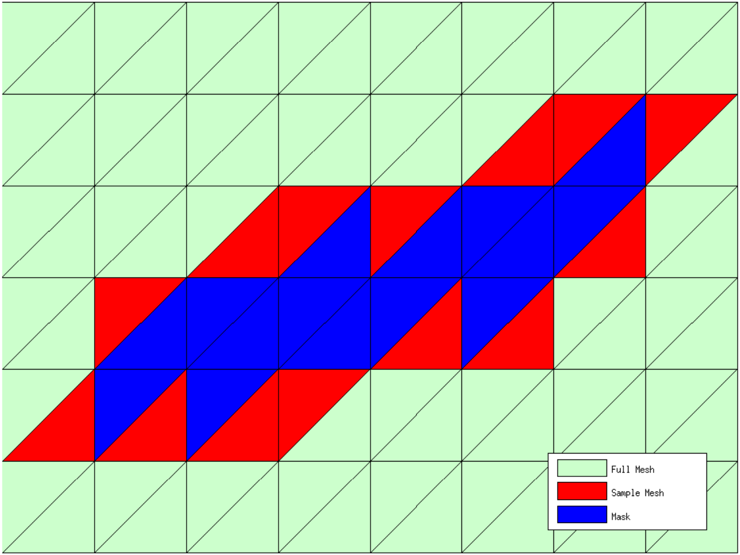
$$\boxed{\mathbf{R}_g(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \boldsymbol{\Phi}_R \left(\mathbf{Z}^T \boldsymbol{\Phi}_R\right)^{\dagger} \left(\mathbf{Z}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})\right) = 0}$$

[13] (Everson and Sirovich, 1995),(Chaturantabut and Sorensen, 2010),(Carlberg et al., 2011)

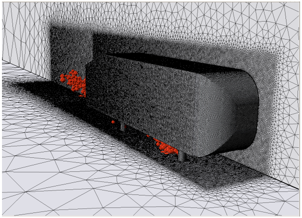
Gappy POD in Practice: Euler Vortex



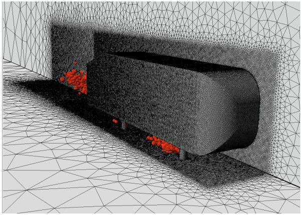
Gappy POD in Practice: Euler Vortex



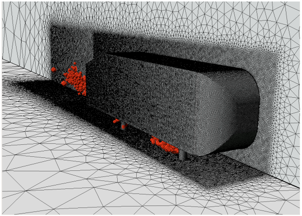
Gappy POD in Practice: Ahmed Body



(a) 253 sample nodes



(b) 378 sample nodes



(c) 505 sample nodes

Bottleneck Alleviation

Using the Gappy POD approximation, the hyper-reduced governing equations are

$$\mathbf{R}_h(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \boldsymbol{\Phi}_R \left(\mathbf{Z}^T \boldsymbol{\Phi}_R\right)^\dagger \left(\mathbf{Z}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})\right) = 0$$

where

$$\mathbf{E} = \boldsymbol{\Psi}^T \boldsymbol{\Phi}_R \left(\mathbf{Z}^T \boldsymbol{\Phi}_R\right)^\dagger$$

is known *offline* and can be precomputed

$$\mathbf{R}_g = \boxed{\mathbf{E}}\; \mathbf{Z}^T\mathbf{R}$$

- Size scales independent of large dimension N
- Amenable to online or deployed computations

Outline

Reduced Optimization Problem

ROM-Constrained Optimization - NAND Formulation

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad f(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

- $\mathbf{y} = \mathbf{y}(\boldsymbol{\mu})$ through $\mathbf{r}(\mathbf{y}, \boldsymbol{\mu}) = 0$
  - For ROM only: $\mathbf{r}(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})$
  - For ROM + hyperreduction: $\mathbf{r}(\mathbf{y}, \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \boldsymbol{\Phi}_R \left(\mathbf{Z}^T \boldsymbol{\Phi}_R\right)^{\dagger} \left(\mathbf{Z}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})\right)$

- Issues that must be considered
  - Construction of bases
  - Speedup potential
  - Reduced sensitivity derivation
  - Training

Gradient of Reduced Objective Function

- Recall MOR assumption:

$$\mathbf{w}_r = \bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y} \qquad \Longrightarrow \qquad \frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}} = \mathbf{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

- For gradient-based optimization, the gradient of the *reduced objective function* is required

$$\begin{aligned}\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\mu}}(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu}) &= \frac{\partial f}{\partial \boldsymbol{\mu}} + \frac{\partial f}{\partial(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y})}\frac{\partial(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y})}{\partial \mathbf{y}}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} \\ &= \frac{\partial f}{\partial \boldsymbol{\mu}} + \frac{\partial f}{\partial \mathbf{w}_r}\mathbf{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} \\ &= \frac{\partial f}{\partial \boldsymbol{\mu}} + \frac{\partial f}{\partial \mathbf{w}_r}\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}}\end{aligned}$$

- Recall HDM gradient:

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\mu}}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = \frac{\partial f}{\partial \boldsymbol{\mu}} + \frac{\partial f}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$$

Sensitivities

HDM sensitivities

$$\mathbf{R}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \implies \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = 0 \implies \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

Sensitivities

HDM sensitivities

$$\mathbf{R}(\mathbf{w}(\boldsymbol{\mu}),\boldsymbol{\mu}) = 0 \implies \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = 0 \implies \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1}\frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

ROM sensitivities

Recall:

$$\mathbf{w}_r = \bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y} \qquad \mathbf{R}_r(\mathbf{y}(\boldsymbol{\mu}),\boldsymbol{\mu}) = \boldsymbol{\Psi}^T\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}(\boldsymbol{\mu}),\boldsymbol{\mu})$$

Sensitivities

HDM sensitivities

$$\mathbf{R}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \implies \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = 0 \implies \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

ROM sensitivities

Recall:

$$\mathbf{w}_r = \bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y} \qquad \mathbf{R}_r(\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

$$\mathbf{R}_r(\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \implies \frac{\partial \mathbf{R}_r}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}_r}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} = 0 \implies \boxed{\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi} \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi}\mathbf{A}^{-1}\mathbf{B}}$$

Sensitivities

HDM sensitivities

$$\mathbf{R}(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \implies \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = 0 \implies \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} = -\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

ROM sensitivities

Recall:

$$\mathbf{w}_r = \bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y} \qquad \mathbf{R}_r(\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu}) = \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

$$\mathbf{R}_r(\mathbf{y}(\boldsymbol{\mu}), \boldsymbol{\mu}) = 0 \implies \frac{\partial \mathbf{R}_r}{\partial \boldsymbol{\mu}} + \frac{\partial \mathbf{R}_r}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} = 0 \implies \boxed{\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi} \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi}\mathbf{A}^{-1}\mathbf{B}}$$

$$\mathbf{A} = \sum_{j=1}^{N} \mathbf{R}_j \frac{\partial \left(\boldsymbol{\Psi}^T \mathbf{e}_j\right)}{\partial \mathbf{w}} \boldsymbol{\Phi} + \boldsymbol{\Psi}^T \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \boldsymbol{\Phi}, \qquad \mathbf{B} = -\left(\sum_{j=1}^{N} \mathbf{R}_j \frac{\partial \left(\boldsymbol{\Psi}^T \mathbf{e}_j\right)}{\partial \boldsymbol{\mu}} + \boldsymbol{\Psi}^T \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}\right)$$

Minimum-Error Reduced Sensitivities

- ROM sensitivities
  - May not represent HDM sensitivities well
  - May be difficult to compute if $\boldsymbol{\Psi} = \boldsymbol{\Psi}(\boldsymbol{\mu})$

Minimum-Error Reduced Sensitivities

- ROM sensitivities
    - May not represent HDM sensitivities well
    - May be difficult to compute if $\boldsymbol{\Psi} = \boldsymbol{\Psi}(\boldsymbol{\mu})$
        - LSPG: $\boldsymbol{\Psi} = \frac{\partial \mathbf{R}}{\partial \mathbf{w}}\boldsymbol{\Phi} \implies \frac{\partial \left(\boldsymbol{\Psi}^T \mathbf{e}_j\right)}{\partial \mathbf{w}}, \frac{\partial \left(\boldsymbol{\Psi}^T \mathbf{e}_j\right)}{\partial \boldsymbol{\mu}}$ involve $\frac{\partial^2 \mathbf{R}}{\partial \mathbf{w} \partial \mathbf{w}}, \frac{\partial^2 \mathbf{R}}{\partial \mathbf{w} \partial \boldsymbol{\mu}}$

Minimum-Error Reduced Sensitivities

- ROM sensitivities
  - May not represent HDM sensitivities well
  - May be difficult to compute if $\boldsymbol{\Psi} = \boldsymbol{\Psi}(\boldsymbol{\mu})$
    - LSPG: $\boldsymbol{\Psi} = \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \boldsymbol{\Phi} \implies \frac{\partial \left(\boldsymbol{\Psi}^T \mathbf{e}_j\right)}{\partial \mathbf{w}}, \frac{\partial \left(\boldsymbol{\Psi}^T \mathbf{e}_j\right)}{\partial \boldsymbol{\mu}}$ involve $\frac{\partial^2 \mathbf{R}}{\partial \mathbf{w} \partial \mathbf{w}}, \frac{\partial^2 \mathbf{R}}{\partial \mathbf{w} \partial \boldsymbol{\mu}}$
- Define quantity that *minimizes* the sensitivity error in some norm $\boldsymbol{\Theta} \succ 0$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = \arg\min_{\mathbf{a}} ||\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} - \boldsymbol{\Phi}\mathbf{a}||_{\boldsymbol{\Theta}}$$

Minimum-Error Reduced Sensitivities

- ROM sensitivities
  - May not represent HDM sensitivities well
  - May be difficult to compute if $\mathbf{\Psi} = \mathbf{\Psi}(\boldsymbol{\mu})$
    - LSPG: $\mathbf{\Psi} = \frac{\partial \mathbf{R}}{\partial \mathbf{w}}\mathbf{\Phi} \implies \frac{\partial\left(\mathbf{\Psi}^T\mathbf{e}_j\right)}{\partial \mathbf{w}}, \frac{\partial\left(\mathbf{\Psi}^T\mathbf{e}_j\right)}{\partial \boldsymbol{\mu}}$ involve $\frac{\partial^2 \mathbf{R}}{\partial \mathbf{w}\partial \mathbf{w}}, \frac{\partial^2 \mathbf{R}}{\partial \mathbf{w}\partial \boldsymbol{\mu}}$
- Define quantity that *minimizes* the sensitivity error in some norm $\mathbf{\Theta} \succ 0$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = \arg\min_{\mathbf{a}} ||\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}} - \mathbf{\Phi}\mathbf{a}||_{\mathbf{\Theta}}$$

$$\implies \widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = -\left(\mathbf{\Theta}^{1/2}\mathbf{\Phi}\right)^{\dagger}\mathbf{\Theta}^{1/2}\frac{\partial \mathbf{R}}{\partial \mathbf{w}}^{-1}\frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

Minimum-Error Reduced Sensitivities

- Similar in spirit to the derivation of LSPG, select $\boldsymbol{\Theta}^{1/2} = \frac{\partial \mathbf{R}}{\partial \mathbf{w}}$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = -\left(\frac{\partial \mathbf{R}}{\partial \mathbf{w}} \boldsymbol{\Phi}\right)^{\dagger} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

Minimum-Error Reduced Sensitivities

- Similar in spirit to the derivation of LSPG, select $\mathbf{\Theta}^{1/2} = \frac{\partial \mathbf{R}}{\partial \mathbf{w}}$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = -\left(\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\mathbf{\Phi}\right)^{\dagger} \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

- Instead of true objective gradient

$$\frac{\mathrm{d} f_r}{\mathrm{d}\boldsymbol{\mu}}(\mathbf{w}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) = \frac{\partial f_r}{\partial \boldsymbol{\mu}} + \frac{\partial f_r}{\partial \mathbf{w}}\mathbf{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

use $\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}}$ as a surrogate for $\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$

$$\boxed{\widehat{\frac{\mathrm{d} f_r}{\mathrm{d}\boldsymbol{\mu}}}(\mathbf{w}_r, \boldsymbol{\mu}) = \frac{\partial f_r}{\partial \boldsymbol{\mu}} + \frac{\partial f_r}{\partial \mathbf{w}}\mathbf{\Phi}\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}}}$$

Minimum-Error Reduced Sensitivities

Minimum-Error Reduced Sensitivities

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = -\left(\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\boldsymbol{\Phi}\right)^{\dagger}\frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} \qquad \widehat{\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}}} = \boldsymbol{\Phi}\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}}$$

- Advantages
  - Error between HDM/ROM sensitivities decreases monotonically as vectors added to $\boldsymbol{\Phi}$
  - If $\left\{\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}\right\} \subset \text{range}\ \boldsymbol{\Phi}$, exact sensitivities recovered $\widehat{\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}}} = \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$
    - If *sensitivity* basis not truncated, *exact derivatives* recovered at training points

Minimum-Error Reduced Sensitivities

Minimum-Error Reduced Sensitivities

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = -\left(\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\boldsymbol{\Phi}\right)^{\dagger}\frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}} \qquad \widehat{\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}}} = \boldsymbol{\Phi}\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}}$$

- Advantages
  - Error between HDM/ROM sensitivities decreases monotonically as vectors added to $\boldsymbol{\Phi}$
  - If $\left\{\frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}\right\} \subset \operatorname{range} \boldsymbol{\Phi}$, exact sensitivities recovered $\widehat{\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}}} = \frac{\partial \mathbf{w}}{\partial \boldsymbol{\mu}}$
    - If *sensitivity* basis not truncated, *exact derivatives* recovered at training points
- Disadvantages
  - In general, $\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} \neq \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} \implies \widehat{\frac{\mathrm{d} f_r}{\mathrm{d} \boldsymbol{\mu}}} \neq \frac{\mathrm{d} f_r}{\mathrm{d} \boldsymbol{\mu}}$
    - Convergence issues for reduced optimization problem

Minimum-Error Reduced Sensitivities and LSPG

ROM sensitivities

$$\frac{\partial \mathbf{w}_r}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi}\mathbf{A}^{-1}\mathbf{B}$$

$$\mathbf{A} = \sum_{j=1}^{N} \mathbf{R}_j \frac{\partial\left(\boldsymbol{\Psi}^T\mathbf{e}_j\right)}{\partial \mathbf{w}}\boldsymbol{\Phi} + \boldsymbol{\Psi}^T\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\boldsymbol{\Phi}, \qquad \mathbf{B} = -\left(\sum_{j=1}^{N} \mathbf{R}_j \frac{\partial\left(\boldsymbol{\Psi}^T\mathbf{e}_j\right)}{\partial \boldsymbol{\mu}} + \boldsymbol{\Psi}^T\frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}\right)$$

For LSPG ROM

$$\boxed{\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}} \text{ with second derivatives dropped}}$$

$$||\mathbf{R}|| \rightarrow 0 \implies \widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}} \rightarrow \frac{\partial \mathbf{y}}{\partial \boldsymbol{\mu}}$$

Offline-Online (Database) Approach

Offline-Online Approach to ROM-Constrained Optimization

- Identify samples in *offline* phase to be used for training
  - Space-fill sampling (i.e. latin hypercube)
  - Greedy sampling
- Collect snapshots from HDM
- Build ROB $\boldsymbol{\Phi}$
- Solve optimization problem

$$\begin{aligned} \underset{\mathbf{y}\in\mathbb{R}^n,\ \boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) \\ \text{subject to} \quad & \boldsymbol{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0 \end{aligned}$$

(LeGresley and Alonso, 2000), (Lassila and Rozza, 2010), (Rozza and Manzoni, 2010), (Manzoni et al., 2012)

Offline-Online Approach

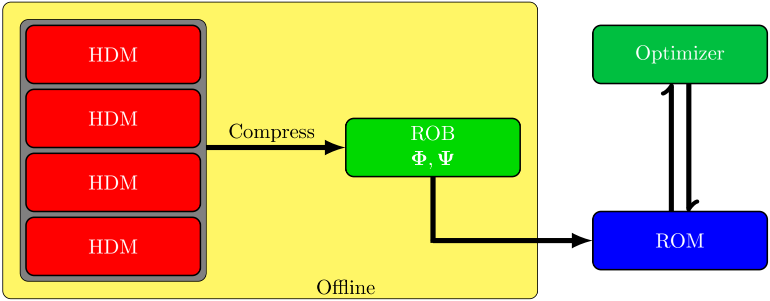
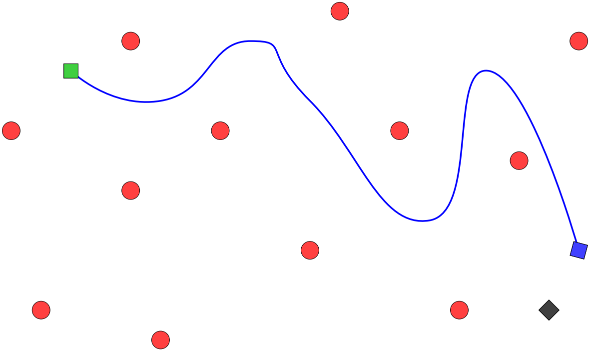
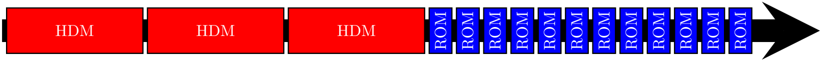


Figure: Schematic of Algorithm

Offline-Online Approach



(a) Idealized Optimization Trajectory: Parameter Space



(b) Breakdown of Computational Effort

Progressive/Adaptive Approach

Progressive Approach to ROM-Constrained Optimization

- Collect snapshots from HDM at *sparse sampling* of the parameter space
  - Initial condition for optimization problem
- Build ROB $\mathbf{\Phi}$ from sparse training
- Solve optimization problem

$$\begin{aligned} \underset{\mathbf{y}\in\mathbb{R}^n,\ \boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y}, \boldsymbol{\mu}) \\ \text{subject to} \quad & \mathbf{\Psi}^T \mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0 \\ & \frac{1}{2}||\mathbf{R}(\bar{\mathbf{w}} + \mathbf{\Phi}\mathbf{y}, \boldsymbol{\mu})||_2^2 \leq \epsilon \end{aligned}$$

- Use solution of above problem to enrich training and repeat until convergence

(Arian et al., 2000), (Fahl, 2001), (Afanasiev and Hinze, 2001), (Kunisch and Volkwein, 2008), (Hinze and Matthes, 2013), (Yue and Meerbergen, 2013), (Zahr and Farhat, 2014)

Progressive Approach

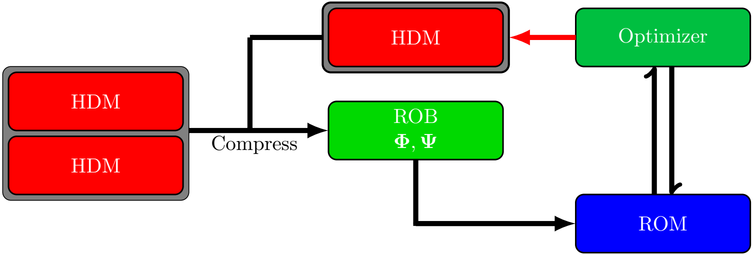
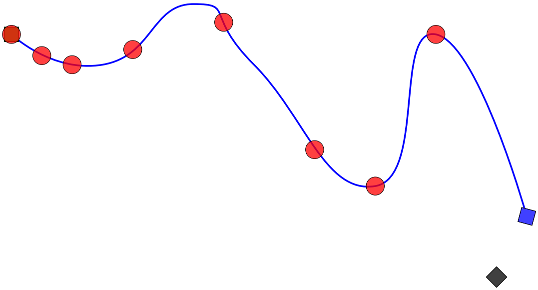
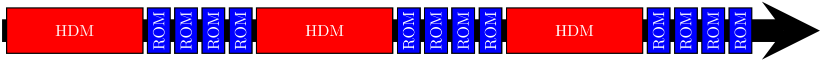


Figure: Schematic of Algorithm

Progressive Approach



(a) Idealized Optimization Trajectory: Parameter Space



(b) Breakdown of Computational Effort

Progressive Approach

Ingredients of Proposed Approach (Zahr and Farhat, 2014)

- Minimum-residual ROM (LSPG) and minimum-error sensitivities
  - $\frac{\mathrm{d}f_r}{\mathrm{d}\boldsymbol{\mu}}(\boldsymbol{\mu}) = \frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\mu}}(\boldsymbol{\mu})$ for training parameters $\boldsymbol{\mu}$
- Reduced optimization (sub)problem

$$\begin{aligned} \underset{\mathbf{y}\in\mathbb{R}^n,\ \boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & f(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) \\ \text{subject to} \quad & \boldsymbol{\Psi}^T\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu}) = 0 \\ & \frac{1}{2}||\mathbf{R}(\bar{\mathbf{w}} + \boldsymbol{\Phi}\mathbf{y}, \boldsymbol{\mu})||_2^2 \leq \epsilon \end{aligned}$$

- Reference vector $\bar{\mathbf{w}}$ and initial guess for each reduced optimization problem
  - $f_r(\boldsymbol{\mu}) = f(\boldsymbol{\mu})$ for training parameters $\boldsymbol{\mu}$
- Efficiently update ROB with additional snapshots or new translation vector
  - Without re-computing SVD of entire snapshot matrix
- Adaptive selection of $\epsilon \rightarrow$ trust-region approach

Initial guess for reduced optimization

Let

$$\begin{aligned}\boldsymbol{\mu}_{-1}^* = \boldsymbol{\mu}_0^{(0)} &= \text{initial condition for PDE-constrained optimization}\\ \boldsymbol{\mu}_j^{(k)} &= k\text{th iteration of } j\text{th reduced optimization problem}\\ \boldsymbol{\mu}_j^* &= \text{solution of } j\text{th reduced optimization problem}\end{aligned}$$

Define

$$\begin{aligned}\mathcal{S}_j^{\boldsymbol{\mu}} &= \{\boldsymbol{\mu}_{-1}^*, \boldsymbol{\mu}_0^*, \ldots, \boldsymbol{\mu}_j^*\}\\ \mathcal{S}_j^{\mathbf{w}} &= \{\mathbf{w}(\boldsymbol{\mu}_{-1}^*), \mathbf{w}(\boldsymbol{\mu}_0^*), \ldots, \mathbf{w}(\boldsymbol{\mu}_j^*)\}\end{aligned}$$

$$\rho_j = \frac{f(\mathbf{w}(\boldsymbol{\mu}_j^*), \boldsymbol{\mu}_j^*) - f(\mathbf{w}(\boldsymbol{\mu}_{j-1}^*), \boldsymbol{\mu}_{j-1}^*)}{f(\mathbf{w}_r(\boldsymbol{\mu}_j^*), \boldsymbol{\mu}_j^*) - f(\mathbf{w}_r(\boldsymbol{\mu}_{j-1}^*), \boldsymbol{\mu}_{j-1}^*)}$$

Initial Guess for Reduced Optimization: Parameter Space

$$\boldsymbol{\mu}_{j+1}^{(0)} = \arg \min_{\boldsymbol{\mu} \in \mathcal{S}_j^{\boldsymbol{\mu}}} f(\mathbf{w}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

- Robustness to poor selection of ϵ

Affine offset and initial guess for ROM solve

Let

$$\begin{aligned}\boldsymbol{\mu}_{-1}^* = \boldsymbol{\mu}_0^{(0)} &= \text{initial condition for PDE-constrained optimization}\\ \boldsymbol{\mu}_j^{(k)} &= k\text{th iteration of } j\text{th reduced optimization problem}\\ \boldsymbol{\mu}_j^* &= \text{solution of } j\text{th reduced optimization problem}\end{aligned}$$

Define

$$\begin{aligned}\mathcal{S}_j^{\boldsymbol{\mu}} &= \{\boldsymbol{\mu}_{-1}^*, \boldsymbol{\mu}_0^*, \dots, \boldsymbol{\mu}_j^*\}\\ \mathcal{S}_j^{\mathbf{w}} &= \{\mathbf{w}(\boldsymbol{\mu}_{-1}^*), \mathbf{w}(\boldsymbol{\mu}_0^*), \dots, \mathbf{w}(\boldsymbol{\mu}_j^*)\}\end{aligned}$$

$$\rho_j = \frac{f(\mathbf{w}(\boldsymbol{\mu}_j^*), \boldsymbol{\mu}_j^*) - f(\mathbf{w}(\boldsymbol{\mu}_{j-1}^*), \boldsymbol{\mu}_{j-1}^*)}{f(\mathbf{w}_r(\boldsymbol{\mu}_j^*), \boldsymbol{\mu}_j^*) - f(\mathbf{w}_r(\boldsymbol{\mu}_{j-1}^*), \boldsymbol{\mu}_{j-1}^*)}$$

Initial Guess for ROM Solve: State Space

$$\bar{\mathbf{w}} = \mathbf{w}^{(0)}$$

$$\mathbf{w}^{(0)} = \underset{\boldsymbol{\mu} \in \mathcal{S}_j^{\mathbf{w}}}{\arg\min} \, ||\mathbf{R}(\mathbf{w}, \boldsymbol{\mu})||$$

- ROM *exact* at training points $\Longrightarrow$ ROM/HDM objective identical

Adaptive Selection of Trust-Region Radius

Let

$$\begin{aligned}\boldsymbol{\mu}_{-1}^* = \boldsymbol{\mu}_0^{(0)} &= \text{initial condition for PDE-constrained optimization}\\ \boldsymbol{\mu}_j^{(k)} &= k\text{th iteration of } j\text{th reduced optimization problem}\\ \boldsymbol{\mu}_j^* &= \text{solution of } j\text{th reduced optimization problem}\end{aligned}$$

Define

$$\begin{aligned}\mathcal{S}_j^{\boldsymbol{\mu}} &= \{\boldsymbol{\mu}_{-1}^*, \boldsymbol{\mu}_0^*, \dots, \boldsymbol{\mu}_j^*\}\\ \mathcal{S}_j^{\mathbf{w}} &= \{\mathbf{w}(\boldsymbol{\mu}_{-1}^*), \mathbf{w}(\boldsymbol{\mu}_0^*), \dots, \mathbf{w}(\boldsymbol{\mu}_j^*)\}\end{aligned}$$

$$\rho_j = \frac{f(\mathbf{w}(\boldsymbol{\mu}_j^*), \boldsymbol{\mu}_j^*) - f(\mathbf{w}(\boldsymbol{\mu}_{j-1}^*), \boldsymbol{\mu}_{j-1}^*)}{f(\mathbf{w}_r(\boldsymbol{\mu}_j^*), \boldsymbol{\mu}_j^*) - f(\mathbf{w}_r(\boldsymbol{\mu}_{j-1}^*), \boldsymbol{\mu}_{j-1}^*)}$$

Trust-Region Radius

$$\epsilon' = \begin{cases} \frac{1}{\tau}\epsilon & \rho_k \in [0.5, 2] \\ \epsilon & \rho_k \in [0.25, 0.5) \cup (2, 4] \\ \tau\epsilon & \text{otherwise} \end{cases}$$

Fast Updates to Reduced-Order Basis

Two situations where snapshot matrix modified (Zahr and Farhat, 2014)

- Additional snapshots to be incorporated

$$\mathbf{\Phi}' = \mathrm{POD}(\begin{bmatrix}\mathbf{X} & \mathbf{Y}\end{bmatrix}) \qquad \text{given} \qquad \mathbf{\Phi} = \mathrm{POD}(\mathbf{X})$$

- Offset vector modified

$$\mathbf{\Phi}' = \mathrm{POD}(\mathbf{X} - \tilde{\mathbf{w}}\mathbf{1}^T) \qquad \text{given} \qquad \mathbf{\Phi} = \mathrm{POD}(\mathbf{X} - \bar{\mathbf{w}}\mathbf{1}^T)$$

Fast Updates to Reduced-Order Basis

Two situations where snapshot matrix modified (Zahr and Farhat, 2014)

- Additional snapshots to be incorporated

$$\mathbf{\Phi}' = \mathrm{POD}\left(\begin{bmatrix}\mathbf{X} & \mathbf{Y}\end{bmatrix}\right) \qquad \text{given} \qquad \mathbf{\Phi} = \mathrm{POD}(\mathbf{X})$$

- Offset vector modified

$$\mathbf{\Phi}' = \mathrm{POD}(\mathbf{X} - \tilde{\mathbf{w}}\mathbf{1}^T) \qquad \text{given} \qquad \mathbf{\Phi} = \mathrm{POD}(\mathbf{X} - \bar{\mathbf{w}}\mathbf{1}^T)$$

Compute new basis using singular factors of existing basis complete without complete recomputation

Fast Updates to Reduced-Order Basis

Two situations where snapshot matrix modified (Zahr and Farhat, 2014)

- Additional snapshots to be incorporated

$$\mathbf{\Phi}' = \text{POD}(\begin{bmatrix}\mathbf{X} & \mathbf{Y}\end{bmatrix}) \qquad \text{given} \qquad \mathbf{\Phi} = \text{POD}(\mathbf{X})$$

- Offset vector modified

$$\mathbf{\Phi}' = \text{POD}(\mathbf{X} - \tilde{\mathbf{w}}\mathbf{1}^T) \qquad \text{given} \qquad \mathbf{\Phi} = \text{POD}(\mathbf{X} - \bar{\mathbf{w}}\mathbf{1}^T)$$

Compute new basis using singular factors of existing basis complete without complete recomputation

Fast, Low-Rank Updates to ROB

Compute (Brand, 2006)

$$\mathbf{\Phi}' = \text{POD}(\mathbf{X} + \mathbf{A}\mathbf{B}^T) \qquad \text{given} \qquad \mathbf{\Phi} = \text{POD}(\mathbf{X})$$

- Large-scale SVD ($N \times n_{\text{snap}}$) replaced by small SVD (independent of N)
- Error incurred by using *truncated* basis $\propto \ \sigma_{n+1}$ (Zahr et al., 2014)
  - Usually small in MOR applications

Interpretation of Proposed Progressive Approach

The proposed approach to PDE-constrained optimization using progressively-constructed ROMs can be interpreted as:

- A nonlinear trust region algorithm for nonlinear programming
  - Nonlinear trust region defined by HDM residual norm
  - Trust region "radius" adaptively selected using traditional trust region techniques
- Trust region model problems defined by the ROM-constrained optimization problem[14]
  - Objective and gradient of ROM-constrained model problem match the HDM quantities at the initial guess of subproblem

[14] (Fahl, 2001)

Outline

1. Motivation
2. PDE-Constrained Optimization
3. Reduced-Order Models
   - Construction of Bases
   - Speedup Potential
4. ROM-Constrained Optimization
   - Reduced Sensitivities
   - Training
5. Numerical Experiments
   - Rocket Nozzle Design
   - Airfoil Design
6. Conclusion
   - Overview
   - Outlook
   - Future Work

Quasi-1D Euler Flow

Quasi-1D Euler equations:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{1}{A}\frac{\partial (A\mathbf{F})}{\partial x} = \mathbf{Q}$$

where

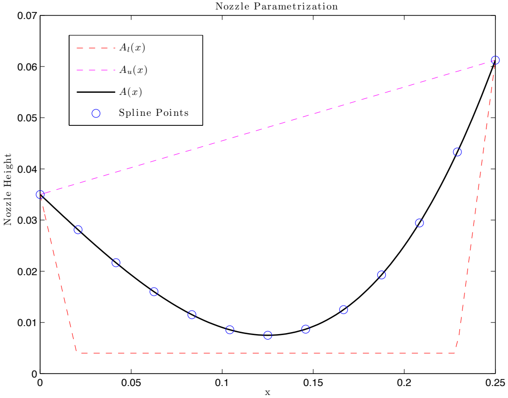
$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \qquad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (e+p)u \end{bmatrix}, \qquad \mathbf{Q} = \begin{bmatrix} 0 \\ \frac{p}{A}\frac{\partial A}{\partial x} \\ 0 \end{bmatrix}$$

- Semi-discretization
  - Finite Volume Method: constant reconstruction, 500 cells
  - Roe flux and entropy correction
- Full discretization
  - Backward Euler
  - Pseudo-transient integration to steady state

Nozzle Parametrization

Nozzle parametrized with *cubic splines* using 13 control points and constraints requiring

- convexity $A''(x) \geq 0$
- bounds on $A(x)$ $A_l(x) \leq A(x) \leq A_u(x)$
- bounds on $A'(x)$ at inlet/outlet $A'(x_l) \leq 0,\ A'(x_r) \geq 0$



Parameter Estimation/Inverse Design

For this problem, the goal is to determine the parameter $\boldsymbol{\mu}^*$ such that the flow achieves some optimal or desired state $\mathbf{w}^*$

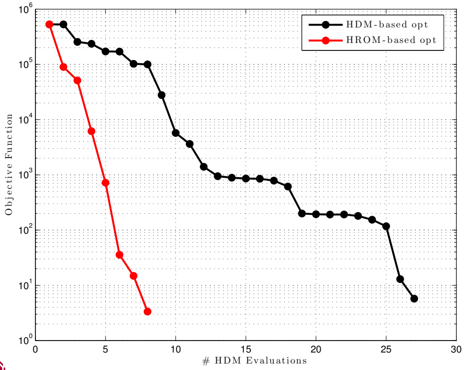
$$\begin{aligned} \underset{\mathbf{w}\in\mathbb{R}^N,\ \boldsymbol{\mu}\in\mathbb{R}^p}{\text{minimize}} \quad & ||\mathbf{w}(\boldsymbol{\mu}) - \mathbf{w}^*|| \\ \text{subject to} \quad & \mathbf{R}(\mathbf{w}, \boldsymbol{\mu}) = 0 \\ & \mathbf{c}(\mathbf{w}, \boldsymbol{\mu}) \leq 0 \end{aligned} \tag{1}$$

where $\mathbf{c}$ are the nozzle constraints.

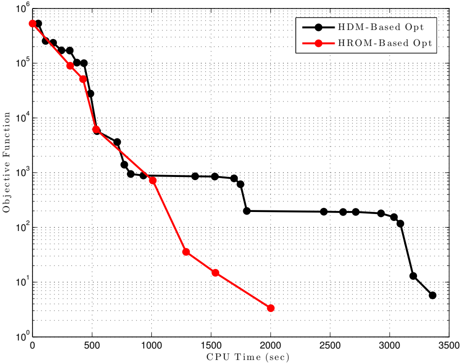
- This problem is solved using
  - the HDM as the governing equation
    - HDM-based optimization
  - the HROM as the governing equation
    - HROM-based optimization

Objective Function Convergence

(a) Convergence (# HDM Evals)

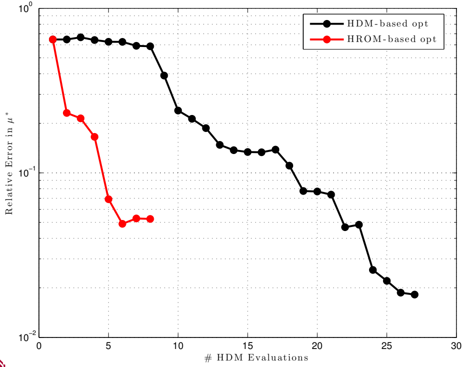


(b) Convergence (CPU Time)

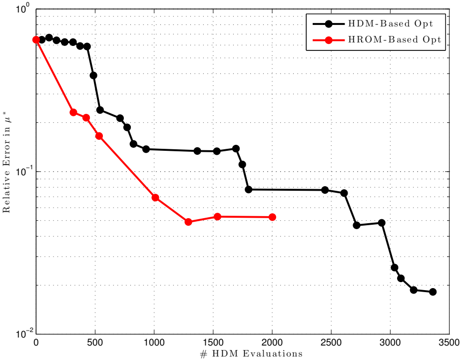


Parameter Estimation Convergence

(a) Convergence (# HDM Evals)

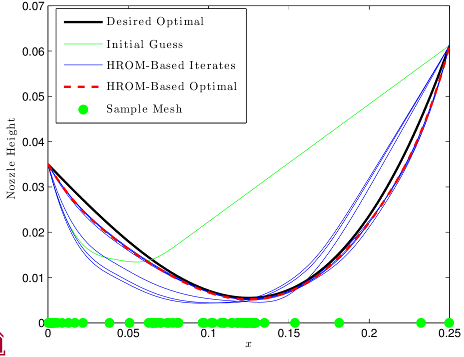


(b) Convergence (CPU Time)

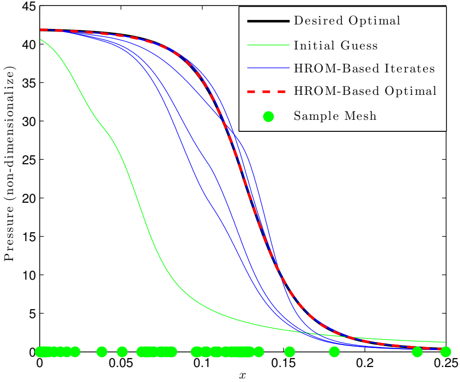


Hyper-Reduced Optimization Progression

(a) Parameter ($\boldsymbol{\mu}$) Progression



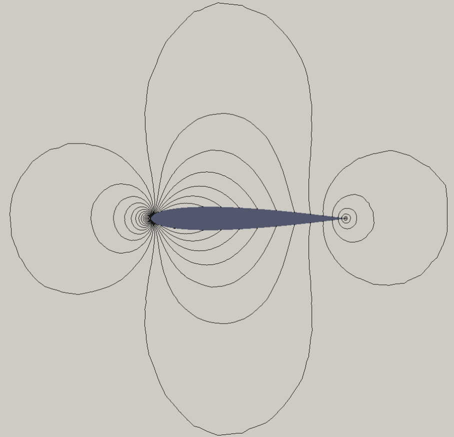
(b) Pressure Progression



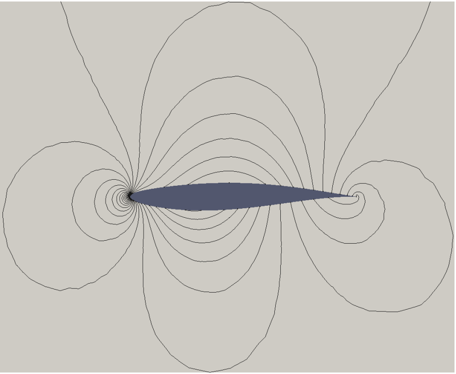
Optimization Summary

	HDM-Based Opt	HROM-Based Opt
Rel. Error in μ^* (%)	1.82	5.26
Rel. Error in w* (%)	0.11	0.12
# HDM Evals	27	8
# HROM Evals	0	161
CPU Time (s)	3361.51	2001.74

Compressible, Inviscid Airfoil Inverse Design



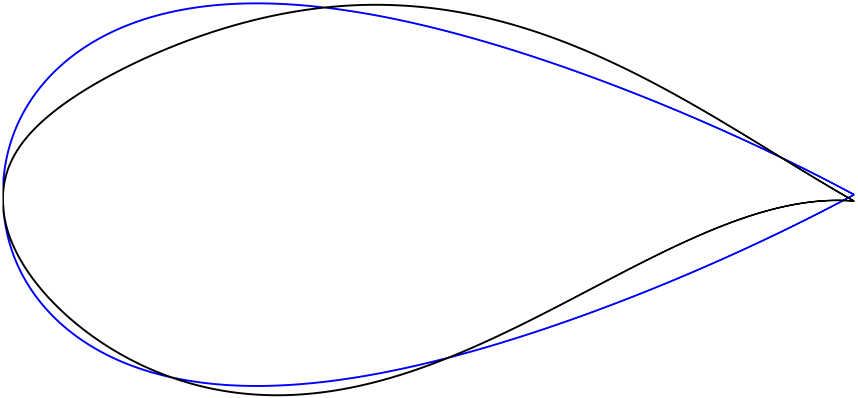
(a) NACA0012: Pressure field ($M_\infty = 0.5$, $\alpha = 0.0°$)



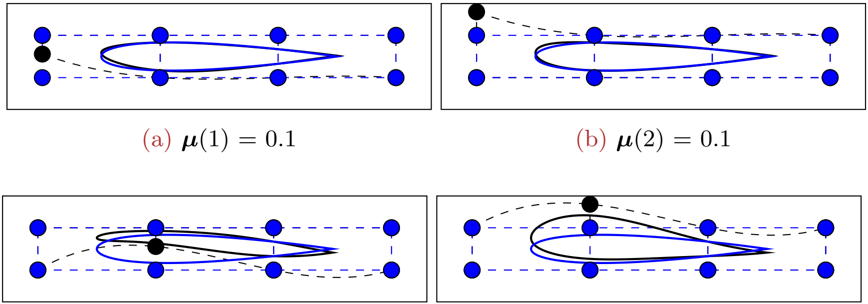
(b) RAE2822: Pressure field ($M_\infty = 0.5$, $\alpha = 0.0°$)

- Pressure discrepancy minimization (Euler equations)
    - Initial Configuration: NACA0012
    - Target Configuration: RAE2822

Initial/Target Airfoils: Scaled



Shape Parametrization

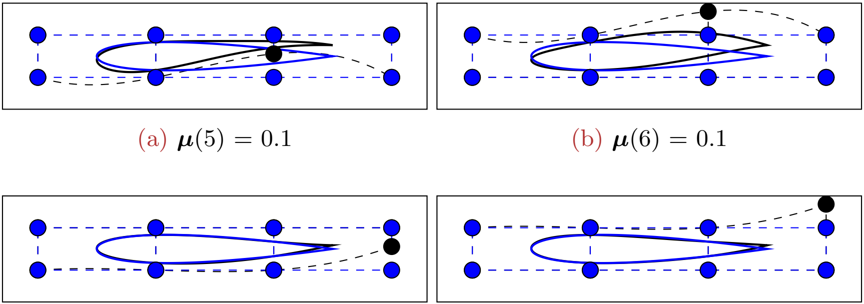


(a) $\boldsymbol{\mu}(1) = 0.1$ (b) $\boldsymbol{\mu}(2) = 0.1$

(c) $\boldsymbol{\mu}(3) = 0.1$ (d) $\boldsymbol{\mu}(4) = 0.1$

Figure: Shape parametrization of a NACA0012 airfoil using a *cubic* design element

Shape Parametrization



(a) $\boldsymbol{\mu}(5) = 0.1$

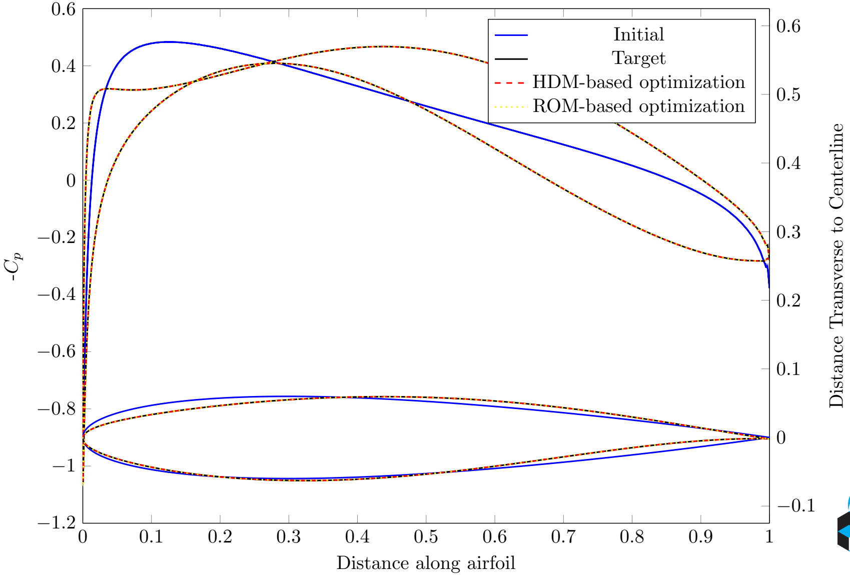
(b) $\boldsymbol{\mu}(6) = 0.1$

(c) $\boldsymbol{\mu}(7) = 0.1$

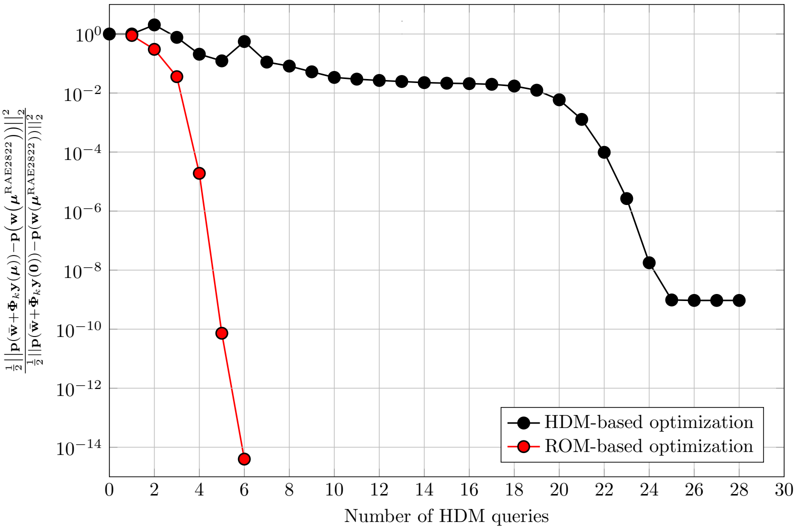
(d) $\boldsymbol{\mu}(8) = 0.1$

Figure: Shape parametrization of a NACA0012 airfoil using a *cubic* design element

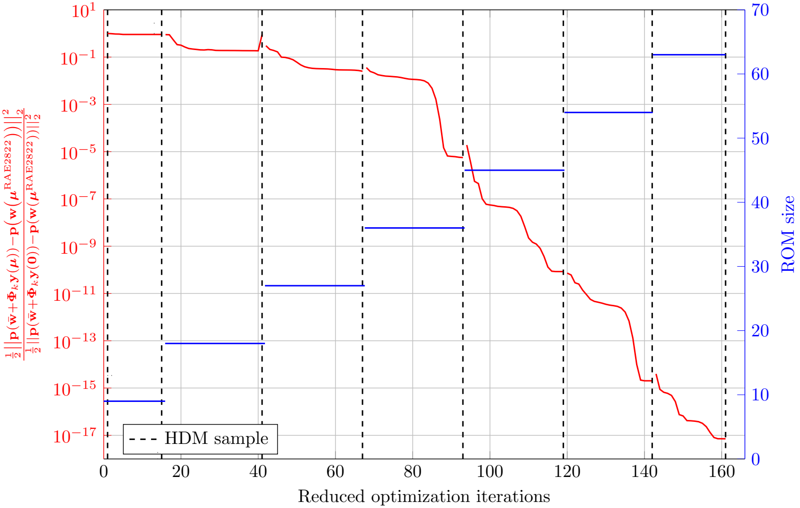
Optimization Results



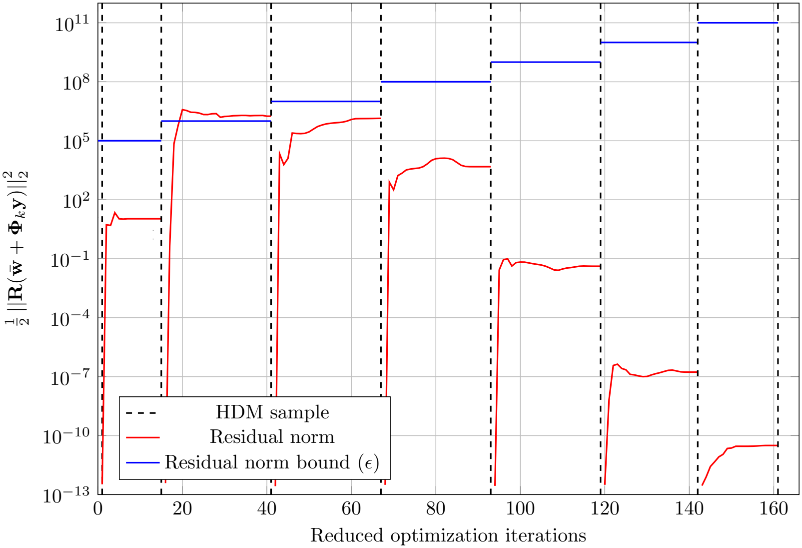
Optimization Results



Optimization Results



Optimization Results



Optimization Results

	HDM-based optimization	ROM-based optimization
# of HDM Evaluations	29	7
# of ROM Evaluations	-	346
$\frac{\|\|\boldsymbol{\mu}^* - \boldsymbol{\mu}^{RAE2822}\|\|}{\|\|\boldsymbol{\mu}^{RAE2822}\|\|}$	$2.28 \times 10^{-3}\%$	$4.17 \times 10^{-6}\%$

Table: Performance of the HDM- and ROM-based optimization methods

Outline

1. Motivation
2. PDE-Constrained Optimization
3. Reduced-Order Models
   - Construction of Bases
   - Speedup Potential
4. ROM-Constrained Optimization
   - Reduced Sensitivities
   - Training
5. Numerical Experiments
   - Rocket Nozzle Design
   - Airfoil Design
6. Conclusion
   - Overview
   - Outlook
   - Future Work

Summary

Summary

- Introduced progressive, nonlinear trust region framework for reduced optimization
- Proposed minimum-error reduced sensitivity analysis
  - Reconstructed reduced sensitivities minimize error to true sensitivities
- Demonstrated approach on canonical problem from aerodynamic shape optimization
  - Factor of 4 fewer queries to HDM than standard PDE-constrained optimization approaches
- Preliminary results on toy problem regarding extension of framework to hyperreduction

Difficulty of Breaking Offline-Online Barrier

Offline-Online Approach

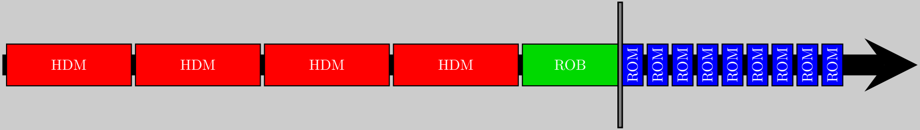


Figure: Offline-Online Approach

- Offline/Online Barrier
  - \+ Enables *large online* speedups
  - \- Difficult to construct accurate, robust ROM
- Minimize ROM!

Difficulty of Breaking Offline-Online Barrier

Progressive Approach

Figure: Progressive Approach

- Requires minimizing HDM, ROB, and ROM!
  - Cost and Quantity

Minimizing Cost of ROM Construction (POD-Based)

- ROM construction ROB cost comes from **SVD** underlying POD
  - R-SVD scales as $\mathcal{O}(6mn^2 + 20n^3)$ for $\mathbf{A} \in \mathbb{R}^{m \times n}$ (Golub and Van Loan, 2012)
  - Our case: $m = \#$DOF in HDM, $n = \#$ snapshots
  - Scales very poorly as snapshots are added
- Competing goals
  - few snapshots to minimize SVD cost
  - many snapshots to maximize accuracy/robustness of ROM
- Applications where smaller, faster SVDs beneficial
  - Computation of state ROB, $\mathbf{\Phi}$, from snapshots
  - Computation of residual ROB, $\mathbf{\Phi}_R$, from snapshots
    - Potential for *HUGE* number of snapshots
  - Compute SVD of snapshot matrix leveraging SVD of subset of columns

Minimizing Cost of ROM Construction (POD-Based)

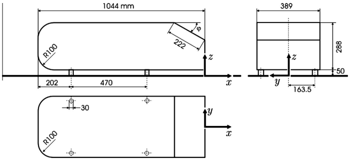
- ROM construction ROB cost comes from **SVD** underlying POD
  - R-SVD scales as $\mathcal{O}(6mn^2 + 20n^3)$ for $\mathbf{A} \in \mathbb{R}^{m \times n}$ (Golub and Van Loan, 2012)
  - Our case: $m = $ #DOF in HDM, $n = $ # snapshots
  - Scales very poorly as snapshots are added
- Solutions
  - Approximate SVD (Halko et al., 2011)
  - Low-rank SVD updates (Brand, 2006), (Zahr et al., 2014)
  - Local ROMs (Dihlmann et al., 2011), (Amsallem et al., 2012)
    - Column partition snapshot; compute SVD of each *local* snapshot set
    - Several SVD computations on matrices with *fewer columns*
  - Adaptive h-refinement (Carlberg, 2014)
    - Fewer snapshots required offline since basis refined online
- Investigation currently underway (Washabaugh, Zahr) to demonstrate "offline" speedup potential of these ideas on large-scale, parametric problem

Minimizing Cost of ROM Evaluation

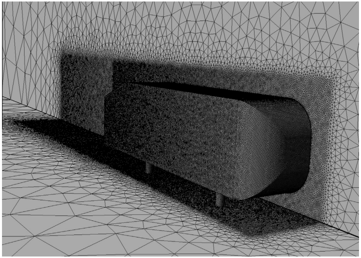
- Many-query setting: number of ROM ROM evaluations will be *LARGE*
  - ROM query as **fast as possible**
    - Reduce computational cost/complexity of evaluating nonlinear terms
    - ROBs as small as possible
  - ROM **accurate** in regions of parameter space of interest
- Solutions
  - Hyperreduction
    - Treatment of nonlinearities
  - Local ROMs
    - Reduce size of ROB at a given time step
  - Adaptive h-refinement
    - Refine ROB only when/where necessary to prevent unnecessarily large bases
  - Temporal forecasting (Carlberg et al., 2012)
    - Reduce temporal complexity

Numerical Example: Ahmed Body

- Benchmark in automotive industry
- Mesh
  - 2,890,434 vertices
  - 17,017,090 tetra
  - 17,342,604 DOF
- CFD
  - Compressible Navier-Stokes
  - DES + Wall func
- Local ROM
  - 4 ROBs: 76, 68, 30, 20
  - Sized by energy (99.75%)



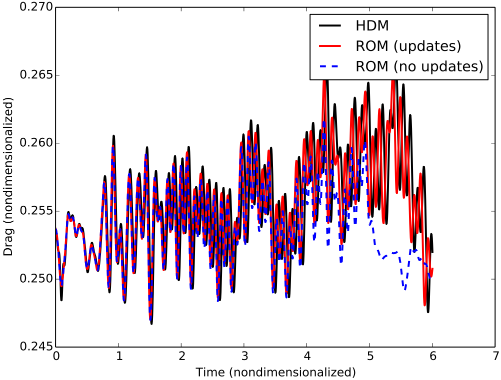
(a) Ahmed Body: Geometry [Ahmed et al 1984]



(b) Ahmed Body: Mesh [Carlberg et al 2011]

Low-Rank SVD Updates

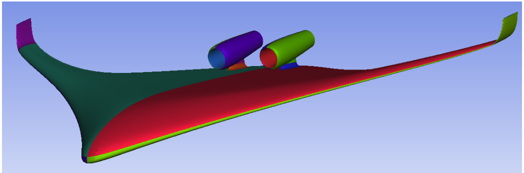
- Potential impact of low-rank SVD updates for ROM applications demonstrated (Zahr et al., 2014) [15]
  - Local ROMs with *online* basis updates
  - Better accuracy for given size of online bases than without updates



[15] Work presented at SIAM Annual Meeting 2014 - Chicago, IL

Future Work

- Incorporate state-of-the-art **ROM technology** into proposed framework
  - Local ROMs, ROB updates, approx SVD, temporal forecasting, ROMES[16]
- **Convergence proof** for proposed progressive optimization framework
- Further development of **hyperreduced sensitivity framework**
- Extensive study to compare with **existing methods**
- Detailed parametric study to assess contribution of **each component**
- Extend ideas to **adjoint approach** (vs. sensitivity approach)
- Application to **large-scale**, 3D problems



- Extension to unsteady PDEs with static parameters
- Extension to unsteady PDEs with dynamic parameters

[16](Drohmann and Carlberg, 2014)

References I

Afanasiev, K. and Hinze, M. (2001).
Adaptive control of a wake flow using proper orthogonal decomposition.
Lecture Notes in Pure and Applied Mathematics, pages 317–332.

Amsallem, D., Zahr, M. J., and Farhat, C. (2012).
Nonlinear model order reduction based on local reduced-order bases.
International Journal for Numerical Methods in Engineering.

Arian, E., Fahl, M., and Sachs, E. W. (2000).
Trust-region proper orthogonal decomposition for flow control.
Technical report, DTIC Document.

Barrault, M., Maday, Y., Nguyen, N. C., and Patera, A. T. (2004).
An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations.
Comptes Rendus Mathematique, 339(9):667–672.

Brand, M. (2006).
Fast low-rank modifications of the thin singular value decomposition.
Linear algebra and its applications, 415(1):20–30.

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

Carlberg, K. (2014).
Adaptive h-refinement for reduced-order models.
arXiv preprint arXiv:1404.0442.

Carlberg, K., Bou-Mosleh, C., and Farhat, C. (2011).
Efficient non-linear model reduction via a least-squares petrov–galerkin projection and compressive tensor approximations.
International Journal for Numerical Methods in Engineering, 86(2):155–181.

References II

Carlberg, K. and Farhat, C. (2008).
A compact proper orthogonal decomposition basis for optimization-oriented reduced-order models.
AIAA Paper, 5964:10–12.

Carlberg, K. and Farhat, C. (2011).
A low-cost, goal-oriented compact proper orthogonal decompositionbasis for model reduction of static systems.
International Journal for Numerical Methods in Engineering, 86(3):381–402.

Carlberg, K., Farhat, C., Cortial, J., and Amsallem, D. (2013).
The gnat method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows.
Journal of Computational Physics.

Carlberg, K., Ray, J., and Waanders, B. v. B. (2012).
Decreasing the temporal complexity for nonlinear, implicit reduced-order models by forecasting.
arXiv preprint arXiv:1209.5455.

Chaturantabut, S. and Sorensen, D. C. (2010).
Nonlinear model reduction via discrete empirical interpolation.
SIAM Journal on Scientific Computing, 32(5):2737–2764.

Dihlmann, M., Drohmann, M., and Haasdonk, B. (2011).
Model reduction of parametrized evolution problems using the reduced basis method with adaptive time partitioning.
Proc. of ADMOS, 2011.

Drohmann, M. and Carlberg, K. (2014).
The romes method for statistical modeling of reduced-order-model error.
SIAM Journal on Uncertainty Quantification.

Everson, R. and Sirovich, L. (1995).
Karhunen–loeve procedure for gappy data.
JOSA A, 12(8):1657–1664.

References III

Fahl, M. (2001).
Trust-region methods for flow control based on reduced order modelling.
PhD thesis, Universitätsbibliothek.

Golub, G. H. and Van Loan, C. F. (2012).
Matrix computations, volume 3.
JHU Press.

Gunzburger, M. D. (2003).
Perspectives in flow control and optimization, volume 5.
Siam.

Halko, N., Martinsson, P.-G., and Tropp, J. A. (2011).
Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions.
SIAM review, 53(2):217–288.

Hay, A., Borggaard, J. T., and Pelletier, D. (2009).
Local improvements to reduced-order models using sensitivity analysis of the proper orthogonal decomposition.
Journal of Fluid Mechanics, 629:41–72.

Hinze, M. and Matthes, U. (2013).
Model order reduction for networks of ode and pde systems.
In *System Modeling and Optimization*, pages 92–101. Springer.

Hinze, M., Pinnau, R., Ulbrich, M., and Ulbrich, S. (2009).
Optimization with PDE constraints.
Springer, New York.

Kunisch, K. and Volkwein, S. (2008).
Proper orthogonal decomposition for optimality systems.
ESAIM: Mathematical Modelling and Numerical Analysis, 42(1):1.

References IV

Lassila, T. and Rozza, G. (2010).
Parametric free-form shape design with pde models and reduced basis method.
Computer Methods in Applied Mechanics and Engineering, 199(23):1583–1592.

LeGresley, P. A. and Alonso, J. J. (2000).
Airfoil design optimization using reduced order models based on proper orthogonal decomposition.
In *Fluids 2000 conference and exhibit, Denver, CO*.

Manzoni, A. (2012).
Reduced models for optimal control, shape optimization and inverse problems in haemodynamics.
PhD thesis, EPFL.

Manzoni, A., Quarteroni, A., and Rozza, G. (2012).
Shape optimization for viscous flows by reduced basis methods and free-form deformation.
International Journal for Numerical Methods in Fluids, 70(5):646–670.

Persson, P.-O., Willis, D., and Peraire, J. (2012).
Numerical simulation of flapping wings using a panel method and a high-order navier–stokes solver.
International Journal for Numerical Methods in Engineering, 89(10):1296–1316.

Rewienski, M. J. (2003).
A trajectory piecewise-linear approach to model order reduction of nonlinear dynamical systems.
PhD thesis, Citeseer.

Rozza, G. and Manzoni, A. (2010).
Model order reduction by geometrical parametrization for shape optimization in computational fluid dynamics.
In *Proceedings of ECCOMAS CFD*.

Sirovich, L. (1987).
Turbulence and the dynamics of coherent structures. i-coherent structures. ii-symmetries and transformations. iii-dynamics and scaling.
Quarterly of applied mathematics, 45:561–571.

References V

Washabaugh, K. and Farhat, C. (2013).
A family of approaches for the reduction of discrete steady nonlinear aerodynamic models.
Technical report, Stanford University.

Yue, Y. and Meerbergen, K. (2013).
Accelerating optimization of parametric linear systems by model order reduction.
SIAM Journal on Optimization, 23(2):1344–1370.

Zahr, M. J. and Farhat, C. (2014).
Progressive construction of a parametric reduced-order model for pde-constrained optimization.
International Journal for Numerical Methods in Engineering, Special Issue on Model Reduction(http://arxiv.org/abs/1407.7618).

Zahr, M. J., Washabaugh, K., and Farhat, C. (2014).
Basis updating in model reduction.
International Journal for Numerical Methods in Engineering.