Optimization of CFD simulations, with MRI applications

Matthew J. Zahr[†] and Per-Olof Persson

TESLA Seminar

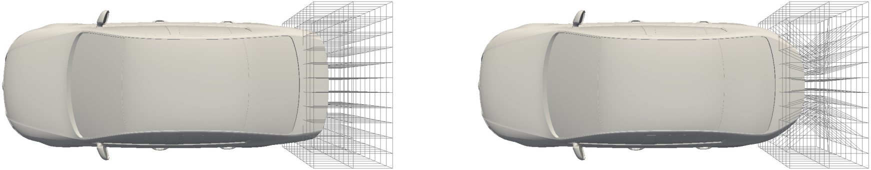
Lund University, Lund, Sweden

March 31, 2017

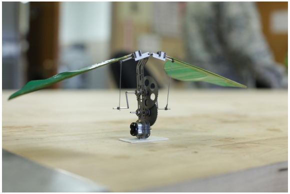
[†] Luis W. Alvarez Postdoctoral Fellow
Department of Mathematics
Lawrence Berkeley National Laboratory
University of California, Berkeley

PDE optimization is ubiquitous in science and engineering

Design: Find system that optimizes performance metric, satisfies constraints



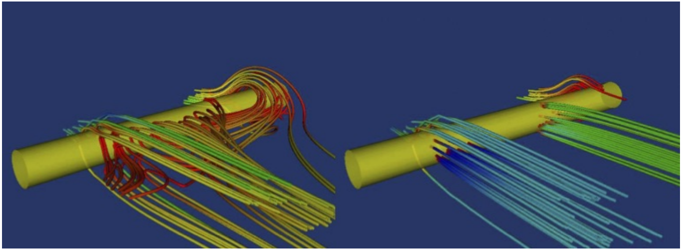
Aerodynamic shape design of automobile



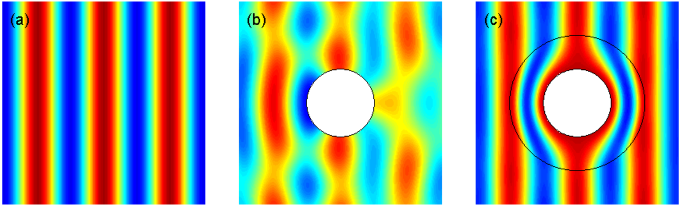
Optimal flapping motion of micro aerial vehicle

PDE optimization is ubiquitous in science and engineering

Control: Drive system to a desired state



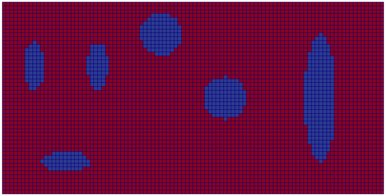
Boundary flow control

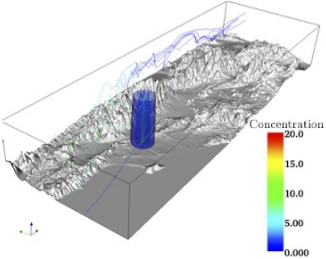


Metamaterial cloaking – electromagnetic invisibility

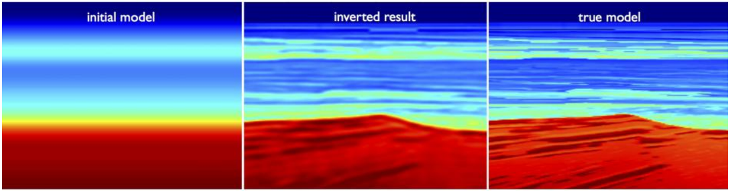
PDE optimization is ubiquitous in science and engineering

Inverse problems: Infer the problem setup given solution observations





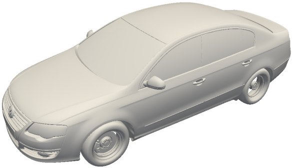
Left: Material inversion – find inclusions from acoustic, structural measurements
Right: Source inversion – find source of airborne contaminant from downstream measurements



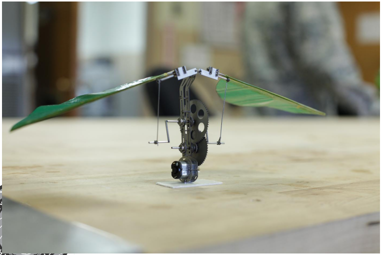
Full waveform inversion – estimate subsurface of Earth's crust from acoustic

Time-dependent PDE-constrained optimization

- Introduction of **fully discrete adjoint method** emanating from **high-order** discretization of governing equations
- **Time-periodicity** constraints
- Extension to high-order partitioned solver for **fluid-structure** interaction
- Solver acceleration via **model reduction**
- Applications: flapping flight, energy harvesting, MRI imaging

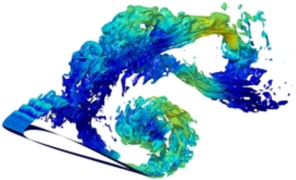


Volkswagen Passat



Micro aerial vehicle

Vertical windmill



LES flow past airfoil

Unsteady PDE-constrained optimization formulation

Goal: Find the solution of the *unsteady PDE-constrained optimization* problem

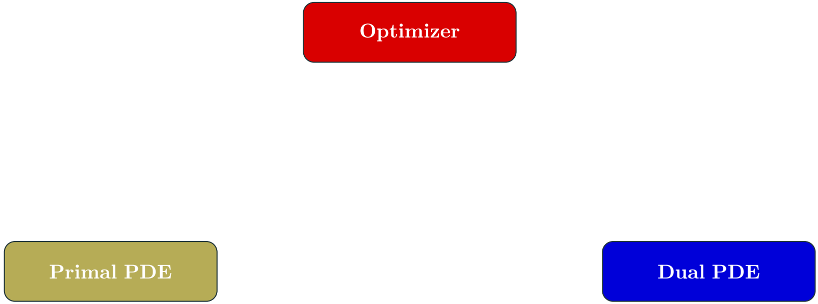
$$
\begin{aligned}
\underset{\boldsymbol{U},\ \boldsymbol{\mu}}{\text{minimize}} \quad & \mathcal{J}(\boldsymbol{U}, \boldsymbol{\mu}) \\
\text{subject to} \quad & \boldsymbol{C}(\boldsymbol{U}, \boldsymbol{\mu}) \leq 0 \\
& \frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}, \nabla \boldsymbol{U}) = 0 \quad \text{in} \quad v(\boldsymbol{\mu}, t)
\end{aligned}
$$

where

- $\boldsymbol{U}(\boldsymbol{x}, t)$ — PDE solution
- $\boldsymbol{\mu}$ — design/control parameters
- $\mathcal{J}(\boldsymbol{U}, \boldsymbol{\mu}) = \int_{T_0}^{T_f} \int_{\boldsymbol{\Gamma}} j(\boldsymbol{U}, \boldsymbol{\mu}, t)\, dS\, dt$ — objective function
- $\boldsymbol{C}(\boldsymbol{U}, \boldsymbol{\mu}) = \int_{T_0}^{T_f} \int_{\boldsymbol{\Gamma}} \mathbf{c}(\boldsymbol{U}, \boldsymbol{\mu}, t)\, dS\, dt$ — constraints

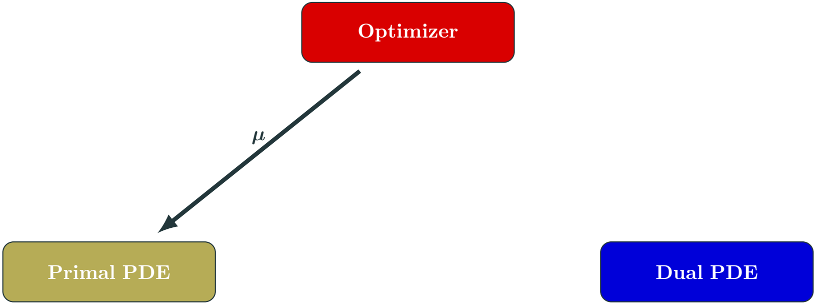
Nested approach to PDE-constrained optimization

PDE optimization requires repeated queries to primal and dual PDE



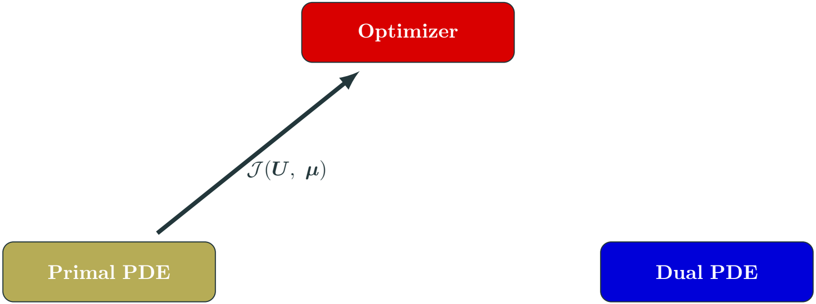
Nested approach to PDE-constrained optimization

PDE optimization requires repeated queries to primal and dual PDE



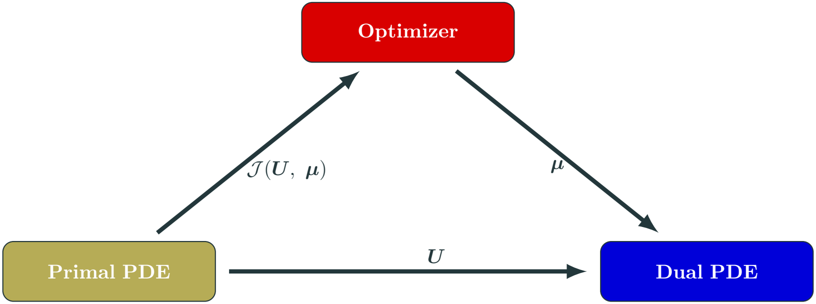
Nested approach to PDE-constrained optimization

PDE optimization requires repeated queries to primal and dual PDE



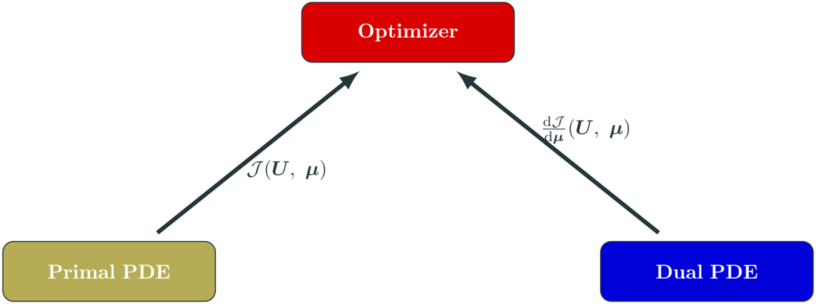
Nested approach to PDE-constrained optimization

PDE optimization requires repeated queries to primal and dual PDE



Nested approach to PDE-constrained optimization

PDE optimization requires repeated queries to primal and dual PDE



High-order discretization of PDE-constrained optimization

- *Continuous* PDE-constrained optimization problem

$$
\begin{aligned}
\underset{\boldsymbol{U},\ \boldsymbol{\mu}}{\text{minimize}} \quad & \mathcal{J}(\boldsymbol{U}, \boldsymbol{\mu}) \\
\text{subject to} \quad & \boldsymbol{C}(\boldsymbol{U}, \boldsymbol{\mu}) \leq 0 \\
& \frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}, \nabla \boldsymbol{U}) = 0 \quad \text{in} \quad v(\boldsymbol{\mu}, t)
\end{aligned}
$$

- *Fully discrete* PDE-constrained optimization problem

$$
\begin{aligned}
\underset{\substack{\boldsymbol{u}_0,\ \dots,\ \boldsymbol{u}_{N_t} \in \mathbb{R}^{N_{\boldsymbol{u}}},\\ \boldsymbol{k}_{1,1},\ \dots,\ \boldsymbol{k}_{N_t,s} \in \mathbb{R}^{N_{\boldsymbol{u}}},\\ \boldsymbol{\mu} \in \mathbb{R}^{n_{\boldsymbol{\mu}}}}}{\text{minimize}} \quad & J(\boldsymbol{u}_0,\ \dots,\ \boldsymbol{u}_{N_t},\ \boldsymbol{k}_{1,1},\ \dots,\ \boldsymbol{k}_{N_t,s},\ \boldsymbol{\mu}) \\
\text{subject to} \quad & \mathbf{C}(\boldsymbol{u}_0,\ \dots,\ \boldsymbol{u}_{N_t},\ \boldsymbol{k}_{1,1},\ \dots,\ \boldsymbol{k}_{N_t,s},\ \boldsymbol{\mu}) \leq 0 \\
& \boldsymbol{u}_0 - \boldsymbol{g}(\boldsymbol{\mu}) = 0 \\
& \boldsymbol{u}_n - \boldsymbol{u}_{n-1} - \sum_{i=1}^{s} b_i \boldsymbol{k}_{n,i} = 0 \\
& \boldsymbol{M} \boldsymbol{k}_{n,i} - \Delta t_n \boldsymbol{r}\left(\boldsymbol{u}_{n,i},\ \boldsymbol{\mu},\ t_{n,i}\right) = 0
\end{aligned}
$$

Adjoint method to efficiently compute gradients of QoI

- Consider the *fully discrete* output functional $F(\boldsymbol{u}_n, \boldsymbol{k}_{n,i}, \boldsymbol{\mu})$
  - Represents either the **objective** function or a **constraint**
- The *total derivative* with respect to the parameters $\boldsymbol{\mu}$, required in the context of gradient-based optimization, takes the form

$$\frac{\mathrm{d}F}{\mathrm{d}\boldsymbol{\mu}} = \frac{\partial F}{\partial \boldsymbol{\mu}} + \sum_{n=0}^{N_t} \frac{\partial F}{\partial \boldsymbol{u}_n} \frac{\partial \boldsymbol{u}_n}{\partial \boldsymbol{\mu}} + \sum_{n=1}^{N_t} \sum_{i=1}^{s} \frac{\partial F}{\partial \boldsymbol{k}_{n,i}} \frac{\partial \boldsymbol{k}_{n,i}}{\partial \boldsymbol{\mu}}$$

- The sensitivities, $\frac{\partial \boldsymbol{u}_n}{\partial \boldsymbol{\mu}}$ and $\frac{\partial \boldsymbol{k}_{n,i}}{\partial \boldsymbol{\mu}}$, are expensive to compute, requiring the solution of $n_{\boldsymbol{\mu}}$ linear evolution equations
- **Adjoint method**: alternative method for computing $\frac{\mathrm{d}F}{\mathrm{d}\boldsymbol{\mu}}$ that require one linear evolution evoluation equation for each quantity of interest, F

Dissection of fully discrete adjoint equations

- **Linear** evolution equations solved **backward** in time
- **Primal** state/stage, $\boldsymbol{u}_{n,i}$ required at each state/stage of dual problem
- Heavily dependent on **chosen ouput**

$$\boldsymbol{\lambda}_{N_t} = \frac{\partial F}{\partial \boldsymbol{u}_{N_t}}^T$$

$$\boldsymbol{\lambda}_{n-1} = \boldsymbol{\lambda}_n + \frac{\partial F}{\partial \boldsymbol{u}_{n-1}}^T + \sum_{i=1}^{s} \Delta t_n \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{u}}\left(\boldsymbol{u}_{n,i},\ \boldsymbol{\mu},\ t_{n-1} + c_i \Delta t_n\right)^T \boldsymbol{\kappa}_{n,i}$$

$$\boldsymbol{M}^T \boldsymbol{\kappa}_{n,i} = \frac{\partial F}{\partial \boldsymbol{k}_{n,i}}^T + b_i \boldsymbol{\lambda}_n + \sum_{j=i}^{s} a_{ji} \Delta t_n \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{u}}\left(\boldsymbol{u}_{n,j},\ \boldsymbol{\mu},\ t_{n-1} + c_j \Delta t_n\right)^T \boldsymbol{\kappa}_{n,j}$$

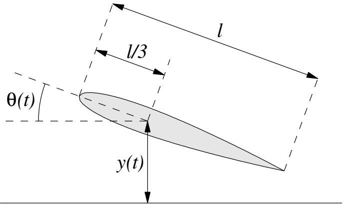
- Gradient reconstruction via dual variables

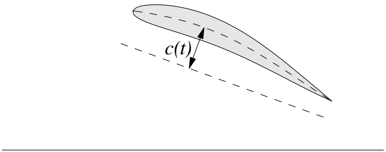
$$\frac{\mathrm{d}F}{\mathrm{d}\boldsymbol{\mu}} = \frac{\partial F}{\partial \boldsymbol{\mu}} + \boldsymbol{\lambda}_0{}^T \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{\mu}}(\boldsymbol{\mu}) + \sum_{n=1}^{N_t} \Delta t_n \sum_{i=1}^{s} \boldsymbol{\kappa}_{n,i}{}^T \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\mu}}\left(\boldsymbol{u}_{n,i},\ \boldsymbol{\mu},\ t_{n,i}\right)$$

Energetically optimal flapping under x-impulse constraint

$$\begin{aligned}
\underset{\boldsymbol{\mu}}{\text{minimize}} \quad & -\int_{2T}^{3T}\int_{\boldsymbol{\Gamma}} \boldsymbol{f}\cdot\boldsymbol{v}\, dS\, dt \\
\text{subject to} \quad & \int_{2T}^{3T}\int_{\boldsymbol{\Gamma}} \boldsymbol{f}\cdot\boldsymbol{e}_1\, dS\, dt = q \\
& \boldsymbol{U}(\boldsymbol{x},0) = \boldsymbol{g}(\boldsymbol{x}) \\
& \frac{\partial \boldsymbol{U}}{\partial t} + \nabla\cdot\boldsymbol{F}(\boldsymbol{U},\nabla\boldsymbol{U}) = 0
\end{aligned}$$

- Isentropic, compressible, Navier-Stokes
- Re $=$ 1000, M $=$ 0.2
- $y(t)$, $\theta(t)$, $c(t)$ parametrized via periodic cubic splines
- Black-box optimizer: SNOPT





Airfoil schematic, kinematic description

Optimal control - fixed shape

Fixed shape, optimal Rigid Body Motion (RBM), varied x-impulse

Initial Guess	Optimal RBM $J_x = 0.0$	Optimal RBM $J_x = -2.5$
Energy $= 9.4096$	Energy $= 0.45695$	Energy $= 4.9475$
x-impulse $=$ -0.1766	x-impulse $= 0.000$	x-impulse $=$ -2.500

Optimal control, time-morphed geometry

Optimal Rigid Body Motion (RBM) and Time-Morphed Geometry (TMG), varied x-impulse

Initial Guess	Optimal RBM/TMG $J_x = 0.0$	Optimal RBM/TMG $J_x = -2.5$
Energy $=$ 9.4096	Energy $=$ 0.45027	Energy $=$ 4.6182
x-impulse $=$ -0.1766	x-impulse $=$ 0.000	x-impulse $=$ -2.500

Optimal control, time-morphed geometry

Optimal Rigid Body Motion (RBM) and Time-Morphed Geometry (TMG),
x-impulse $= -2.5$

Energy $= 9.4096$	Energy $= 4.9476$	Energy $= 4.6182$
x-impulse = -0.1766	x-impulse = -2.500	x-impulse = -2.500
Initial Guess	Optimal RBM $J_x = -2.5$	Optimal RBM/TMG $J_x = -2.5$

Energetically optimal 3D flapping motions

Goal: Find energetically optimal flapping motion that achieves zero thrust

Energy = 1.4459e-01
Thrust = -1.1192e-01

Energy = 3.1378e-01
Thrust = 0.0000e+00

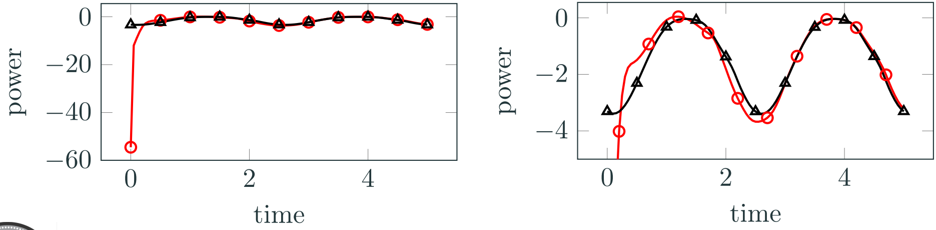
[Zahr and Persson, 2017]

Time-Periodic solutions desired when optimizing cyclic motion

- To properly optimize a cyclic, or periodic problem, need to simulate a **representative** period
- Necessary to avoid transients that will impact quantity of interest and may cause simulation to crash
- **Task**: Find initial condition, $\boldsymbol{u}_0$, such that flow is periodic, i.e. $\boldsymbol{u}_{N_t} = \boldsymbol{u}_0$

Time-periodic solutions desired when optimizing cyclic motion

Vorticity around airfoil with flow initialized from steady-state (left) and time-periodic flow (right)



Time history of power on airfoil of flow initialized from steady-state (—○—) and from a time-periodic solution (—△—)

Time-periodicity constraints in PDE-constrained optimization

Slight modification leads to fully discrete periodic PDE-constrained optimization

$$\begin{aligned}
\underset{\substack{\boldsymbol{u}_0,\ \dots,\ \boldsymbol{u}_{N_t} \in \mathbb{R}^{N_{\boldsymbol{u}}},\\ \boldsymbol{k}_{1,1},\ \dots,\ \boldsymbol{k}_{N_t,s} \in \mathbb{R}^{N_{\boldsymbol{u}}},\\ \boldsymbol{\mu} \in \mathbb{R}^{n_{\boldsymbol{\mu}}}}}{\text{minimize}} \quad & J(\boldsymbol{u}_0,\ \dots,\ \boldsymbol{u}_{N_t},\ \boldsymbol{k}_{1,1},\ \dots,\ \boldsymbol{k}_{N_t,s},\ \boldsymbol{\mu}) \\
\text{subject to} \quad & \mathbf{C}(\boldsymbol{u}_0,\ \dots,\ \boldsymbol{u}_{N_t},\ \boldsymbol{k}_{1,1},\ \dots,\ \boldsymbol{k}_{N_t,s},\ \boldsymbol{\mu}) \leq 0 \\
& \boldsymbol{u}_0 - \boldsymbol{u}_{N_t} = 0 \\
& \boldsymbol{u}_n - \boldsymbol{u}_{n-1} + \sum_{i=1}^{s} b_i \boldsymbol{k}_{n,i} = 0 \\
& \boldsymbol{M}\boldsymbol{k}_{n,i} - \Delta t_n \boldsymbol{r}\left(\boldsymbol{u}_{n,i},\ \boldsymbol{\mu},\ t_{n,i}\right) = 0
\end{aligned}$$

Adjoint method for periodic PDE-constrained optimization

- Following identical procedure as for non-periodic case, the adjoint equations corresponding to the periodic conservation law are

$$\boldsymbol{\lambda}_{N_t} = \boldsymbol{\lambda}_0 + \frac{\partial F}{\partial \boldsymbol{u}_{N_t}}^T$$

$$\boldsymbol{\lambda}_{n-1} = \boldsymbol{\lambda}_n + \frac{\partial F}{\partial \boldsymbol{u}_{n-1}}^T + \sum_{i=1}^{s} \Delta t_n \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{u}} \left(\boldsymbol{u}_{n,i},\ \boldsymbol{\mu},\ t_{n-1} + c_i \Delta t_n\right)^T \boldsymbol{\kappa}_{n,i}$$

$$\boldsymbol{M}^T \boldsymbol{\kappa}_{n,i} = \frac{\partial F}{\partial \boldsymbol{u}_{N_t}}^T + b_i \boldsymbol{\lambda}_n + \sum_{j=i}^{s} a_{ji} \Delta t_n \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{u}} \left(\boldsymbol{u}_{n,j},\ \boldsymbol{\mu},\ t_{n-1} + c_j \Delta t_n\right)^T \boldsymbol{\kappa}_{n,j}$$

- Dual problem is also periodic
- Solve *linear, periodic* problem using Krylov shooting method

Energetically optimal flapping: x-impulse, time-periodic

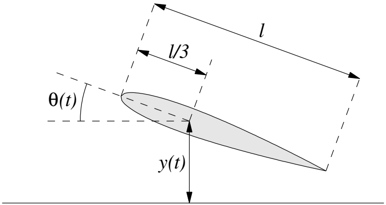
$$\underset{\boldsymbol{\mu}}{\text{minimize}} \quad -\int_0^T \int_{\boldsymbol{\Gamma}} \boldsymbol{f} \cdot \boldsymbol{v} \, dS \, dt$$

$$\text{subject to} \quad \int_0^T \int_{\boldsymbol{\Gamma}} \boldsymbol{f} \cdot \boldsymbol{e}_1 \, dS \, dt = q$$

$$\boldsymbol{U}(\boldsymbol{x}, 0) = \boldsymbol{U}(\boldsymbol{x}, T)$$

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}, \nabla \boldsymbol{U}) = 0$$

- Isentropic, compressible, Navier-Stokes
- Re = 1000, M = 0.2
- $y(t)$, $\theta(t)$, $c(t)$ parametrized via periodic cubic splines
- Black-box optimizer: SNOPT



Airfoil schematic, kinematic description

Solution of time-periodic, energetically optimal flapping

Energy $= 9.4096$	Energy $= 0.45695$
x-impulse $=$ -0.1766	x-impulse $= 0.000$

Data assimilation: inverse problem in medical imaging

Goal: Determine the boundary conditions that produces a high-resolution flow that matches low-resolution flow measurements ($\boldsymbol{d}^*$)

$$\begin{aligned} \underset{\boldsymbol{\mu}}{\text{minimize}} \quad & \frac{1}{2}|\boldsymbol{d}(\boldsymbol{U}) - \boldsymbol{d}^*|^2 & \\ \text{subject to} \quad & \frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}, \nabla \boldsymbol{U}) = 0 & \text{in } \Omega \\ & \boldsymbol{U} = \boldsymbol{\mu} & \text{on } \Gamma \end{aligned}$$

Data assimilation: inverse problem in medical imaging

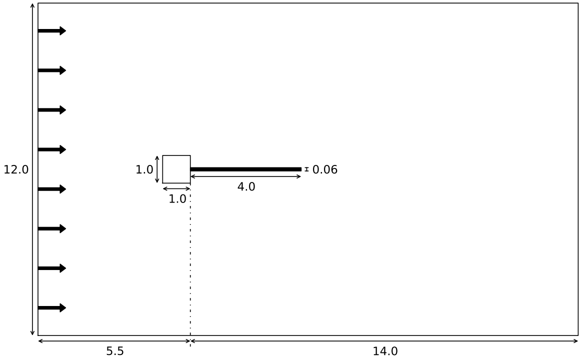
True flow

Data

Reconstructed flow

Fluid-structure interaction: cantilever system

- Standard FSI benchmark problem.
- Elastic cantilever behind a square bluff body in incompressible flow.



- Cantilever:
  $\rho_s = 100\,\text{kg/m}^3$, $\nu_s = 0.35$,
  $E = 2.5 \times 10^5$ Pa.
- Fluid & Flow:
  $\rho_f = 1.18\,\text{kg/m}^3$,
  $\nu_f = 1.54 \times 10^{-5}\,\text{m}^2/\text{s}$,
  $v_f = 0.513\,\text{m/s}$, $\text{Re} = 333$,
  $\text{Ma} = 0.2$.

- Vortex shedding frequency: ~ 6.3 Hz

  Cantilever first mode: 3.03 Hz

Fluid-structure interaction: cantilever system

Fluid-structure interaction: flow around membrane, 3D

- Angle of attack 22.6°, Reynolds number 2000.
- Flexible structure prevents leading edge separation.

Fluid-structure interaction: flow around membrane, 3D

- Angle of attack 22.6°, Reynolds number 2000.
- Flexible structure prevents leading edge separation.

Coupled fluid-structure formulation

- Write discretized fluid and structure equations as ODEs

$$
\begin{aligned}
\boldsymbol{M}^f \dot{\boldsymbol{u}}^f &= \boldsymbol{r}^f(\boldsymbol{u}^f; \boldsymbol{x}) \\
\boldsymbol{M}^s \dot{\boldsymbol{u}}^s &= \boldsymbol{r}^s(\boldsymbol{u}^s; \boldsymbol{t}) \\
&= \boldsymbol{r}^{ss}(\boldsymbol{u}^s) + \boldsymbol{r}^{sf} \cdot \boldsymbol{t}
\end{aligned}
$$

  in the fluid $\boldsymbol{u}^f$ and structure $\boldsymbol{u}^s$ variables
- Apply couplings
  - Structure-to-fluid: deform fluid domain $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{u}^s)$
  - Fluid-to-structure: apply boundary traction $\boldsymbol{t} = \boldsymbol{t}(\boldsymbol{u}^f)$
- Write coupled system as $\boldsymbol{M}\dot{\boldsymbol{u}} = \boldsymbol{r}(\boldsymbol{u})$

$$
\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}^f \\ \boldsymbol{u}^s \end{bmatrix} \qquad \boldsymbol{r}(\boldsymbol{u}) = \begin{bmatrix} \boldsymbol{r}^f(\boldsymbol{u}^f; \boldsymbol{x}(\boldsymbol{u}^s)) \\ \boldsymbol{r}^s(\boldsymbol{u}^s; \boldsymbol{t}(\boldsymbol{u}^f)) \end{bmatrix} \qquad \boldsymbol{M} = \begin{bmatrix} \boldsymbol{M}^f & \\ & \boldsymbol{M}^s \end{bmatrix}
$$

High-order partitioned FSI solver: IMEX Runge-Kutta

- Define **stage solutions**

$$\boldsymbol{u}^s_{n,i} = \boldsymbol{u}^s_{n-1} + \sum_{j=1}^{i} a_{ij}\boldsymbol{k}^s_{n,j} + \sum_{j=1}^{i-1} \hat{a}_{ij}\hat{\boldsymbol{k}}^s_{n,j}$$

$$\boldsymbol{u}^f_{n,i} = \boldsymbol{u}^f_{n-1} + \sum_{j=1}^{i} a_{ij}\boldsymbol{k}^f_{n,j}$$

- Define **traction predictor** as true traction at previous stage

$$\tilde{\boldsymbol{t}}_{n,i} = \boldsymbol{t}(\boldsymbol{u}_{n,i-1})$$

- Solve for **stage velocities** $(i = 1, \ldots, s)$

$$\boldsymbol{M}^s\boldsymbol{k}^s_{n,i} = \Delta t_n \boldsymbol{r}^s(\boldsymbol{u}^s_{n,i};\, \tilde{\boldsymbol{t}}_{n,i})$$
$$\boldsymbol{M}^f\boldsymbol{k}^f_{n,i} = \Delta t_n \boldsymbol{r}^f(\boldsymbol{u}^f_{n,i};\, \boldsymbol{x}(\boldsymbol{u}^s_{n,i}))$$
$$\boldsymbol{M}^s\hat{\boldsymbol{k}}^s_{n,i} = \Delta t_n \boldsymbol{r}^{sf}(\boldsymbol{t}(\boldsymbol{u}^f_{n,i}) - \tilde{\boldsymbol{t}}_{n,i})$$

- Update state solution at new time

$$\boldsymbol{u}^f_{n} = \boldsymbol{u}^f_{n-1} + \sum_{j=1}^{s} b_j\boldsymbol{k}^f_{n,j}, \qquad \boldsymbol{u}^s_{n} = \boldsymbol{u}^s_{n-1} + \sum_{j=1}^{s} b_j\boldsymbol{k}^s_{n,j} + \sum_{j=1}^{s} \hat{b}_j\hat{\boldsymbol{k}}^s_{n,j}$$

Adjoint equations for high-order partitioned IMEX FSI solver

- Define

$$\boldsymbol{r}^f_{n,i} = \boldsymbol{r}^f(\boldsymbol{u}^f_{n,i};\, \boldsymbol{x}(\boldsymbol{u}^s_{n,i})) \qquad \boldsymbol{r}^s_{n,i} = \boldsymbol{r}^s(\boldsymbol{u}^s_{n,i};\, \tilde{\boldsymbol{t}}_{n,i})$$

- **Final condition** for state Lagrange multipliers (F is quantity of interest)

$$\boldsymbol{\lambda}^f_{N_t} = \frac{\partial F}{\partial \boldsymbol{u}^f_{N_t}}^T, \quad \boldsymbol{\lambda}^s_{N_t} = \frac{\partial F}{\partial \boldsymbol{u}^s_{N_t}}^T$$

- Solve for **stage** Lagrange multipliers ($j = s, \ldots, 1$)
  - **Explicit structure** stage

$$\boldsymbol{M}^{s\,T}\hat{\boldsymbol{\kappa}}^s_{n,j} = \frac{\partial F}{\partial \hat{\boldsymbol{k}}^s_{n,j}}^T + \hat{b}_j \boldsymbol{\lambda}^s_n + \Delta t_n \sum_{i=j+1}^{s} \hat{a}_{ij} \frac{\partial \boldsymbol{r}^f_{n,i}}{\partial \boldsymbol{u}^s}^T \boldsymbol{\kappa}^f_{n,i} + \Delta t_n \sum_{i=j+1}^{s} \hat{a}_{ij} \frac{\partial \boldsymbol{r}^s_{n,i}}{\partial \boldsymbol{u}^s}^T \boldsymbol{\kappa}^s_{n,i}$$

  - **Implicit fluid** stage

$$\boldsymbol{M}^{f\,T}\boldsymbol{\kappa}^f_{n,j} = \frac{\partial F}{\partial \boldsymbol{k}^f_{n,j}}^T + b_j \boldsymbol{\lambda}^f_n + \Delta t_n \sum_{i=j}^{s} a_{ij} \frac{\partial \boldsymbol{r}^f_{n,i}}{\partial \boldsymbol{u}^f}^T \boldsymbol{\kappa}^f_{n,i} + \Delta t_n \sum_{i=j+1}^{s} a_{ij} \frac{\partial \tilde{\boldsymbol{t}}_{n,i}}{\partial \boldsymbol{u}^f}^T \boldsymbol{r}^{sf\,T} \boldsymbol{\kappa}^s_{n,i}$$

$$- \Delta t_n \sum_{i=j}^{s} a_{ij} \frac{\partial \boldsymbol{t}_{n,i}}{\partial \boldsymbol{u}^f}^T \boldsymbol{r}^{sf\,T} \hat{\boldsymbol{\kappa}}^s_{n,i} + \Delta t_n \sum_{i=j+1}^{s} a_{ij} \frac{\partial \tilde{\boldsymbol{t}}_{n,i}}{\partial \boldsymbol{u}^f}^T \boldsymbol{r}^{sf\,T} \hat{\boldsymbol{\kappa}}^s_{n,i}$$

  - **Implicit structure** stage

$$\boldsymbol{M}^{s\,T}\boldsymbol{\kappa}^s_{n,j} = \frac{\partial F}{\partial \boldsymbol{k}^s_{n,j}}^T + b_j \boldsymbol{\lambda}^s_n + \Delta t_n \sum_{i=j}^{s} a_{ij} \frac{\partial \boldsymbol{r}^f_{n,i}}{\partial \boldsymbol{u}^s}^T \boldsymbol{\kappa}^f_{n,i} + \Delta t_n \sum_{i=j}^{s} a_{ij} \frac{\partial \boldsymbol{r}^s_{n,i}}{\partial \boldsymbol{u}^s}^T \boldsymbol{\kappa}^s_{n,i}$$

- Update state Lagrange multipliers at new time

$$\boldsymbol{\lambda}_{n-1}^f = \boldsymbol{\lambda}_n^f + \frac{\partial F}{\partial \boldsymbol{u}_{n-1}^f}^T + \Delta t_n \sum_{i=1}^s \frac{\partial \boldsymbol{r}_{n,i}^f}{\partial \boldsymbol{u}^f}^T \boldsymbol{\kappa}_{n,i}^f + \Delta t_n \sum_{i=1}^s \frac{\partial \tilde{\boldsymbol{t}}_{n,i}}{\partial \boldsymbol{u}^f}^T \boldsymbol{r}_{n,i}^{sf\,T} \boldsymbol{\kappa}_{n,i}^s + \Delta t_n \sum_{i=1}^s \left[\frac{\partial \tilde{\boldsymbol{t}}_{n,i}}{\partial \boldsymbol{u}^f} - \frac{\partial \boldsymbol{t}_{n,i}}{\partial \boldsymbol{u}^f} \right]^T \boldsymbol{r}_{n,i}^{sf\,T} \hat{\boldsymbol{\kappa}}_{n,i}^s$$

$$\boldsymbol{\lambda}_{n-1}^s = \boldsymbol{\lambda}_n^s + \frac{\partial F}{\partial \boldsymbol{u}_{n-1}^s}^T + \Delta t_n \sum_{i=1}^s \frac{\partial \boldsymbol{r}_{n,i}^f}{\partial \boldsymbol{u}^s}^T \boldsymbol{\kappa}_{n,i}^f + \Delta t_n \sum_{i=1}^s \frac{\partial \boldsymbol{r}_{n,i}^s}{\partial \boldsymbol{u}^s}^T \boldsymbol{\kappa}_{n,i}^s$$

- Reconstruct **total derivative** of quantity of interest F as

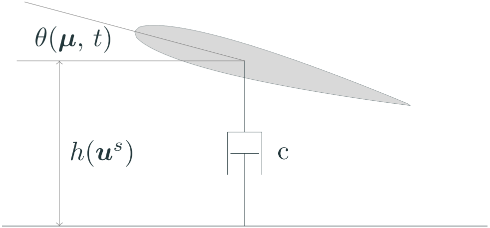
$$\frac{\mathrm{d}F}{\mathrm{d}\boldsymbol{\mu}} = \frac{\partial F}{\partial \boldsymbol{\mu}} + \boldsymbol{\lambda}_0^{f\,T} \frac{\partial \bar{\boldsymbol{u}}^f}{\partial \boldsymbol{\mu}} + \boldsymbol{\lambda}_0^{s\,T} \frac{\partial \bar{\boldsymbol{u}}^s}{\partial \boldsymbol{\mu}} - \sum_{n=0}^{N_t} \Delta t_n \sum_{i=1}^s \boldsymbol{\kappa}_{n,i}^{f\,T} \frac{\partial \boldsymbol{r}_{n,i}^f}{\partial \boldsymbol{\mu}} - \sum_{n=0}^{N_t} \Delta t_n \sum_{i=1}^s \boldsymbol{\kappa}_{n,i}^{s\,T} \frac{\partial \boldsymbol{r}_{n,i}^s}{\partial \boldsymbol{\mu}} - \sum_{n=0}^{N_t} \Delta t_n \sum_{i=1}^s \hat{\boldsymbol{\kappa}}_{n,i}^{s,T} \frac{\partial \boldsymbol{r}_{n,i}^{sf}}{\partial \boldsymbol{\mu}}$$

Optimal energy harvesting from foil-damper system

Goal: Maximize energy harvested from foil-damper system

$$\underset{\boldsymbol{\mu}}{\text{maximize}} \quad \frac{1}{T}\int_0^T (c\dot{h}^2(\boldsymbol{u}^s) - M_z(\boldsymbol{u}^f)\dot{\theta}(\boldsymbol{\mu}, t))\,dt$$

- Fluid: Isentropic Navier-Stokes on deforming domain (ALE)
- Structure: Force balance in y-direction between foil and damper
- Motion driven by *imposed* $\theta(\boldsymbol{\mu}, t) = \mu_1 \cos(2\pi f t)$; $\mu_1 \in (-45°, 45°)$

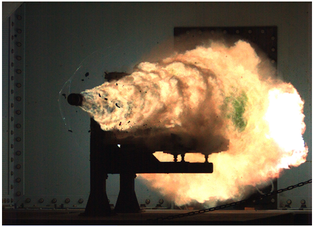


$\mu_1^* = 45°$

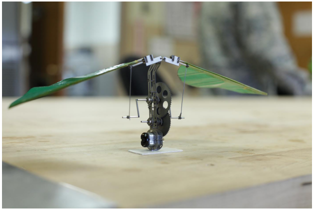
PDE optimization – a key player in next-gen problems

Current interest in ***computational physics*** *reaches far beyond analysis of a single configuration of a physical system into* ***design*** *(shape and topology) and* ***control*** *in an* ***uncertain*** *setting*

Engine System



EM Launcher



Micro-Aerial Vehicle

Repeated queries to **high-fidelity simulations** required by optimization and uncertainty quantification may be **prohibitively time-consuming**

Proposed approach: managed inexactness

Replace expensive PDE with inexpensive approximation model

- **Reduced-order models** used for *inexact PDE evaluations*

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \;\; F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \;\; m_k(\boldsymbol{\mu})$$

[1] Must be *computable* and apply to general, nonlinear PDEs

Proposed approach: managed inexactness

Replace expensive PDE with inexpensive approximation model

- **Reduced-order models** used for *inexact PDE evaluations*

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \;\; F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \;\; m_k(\boldsymbol{\mu})$$

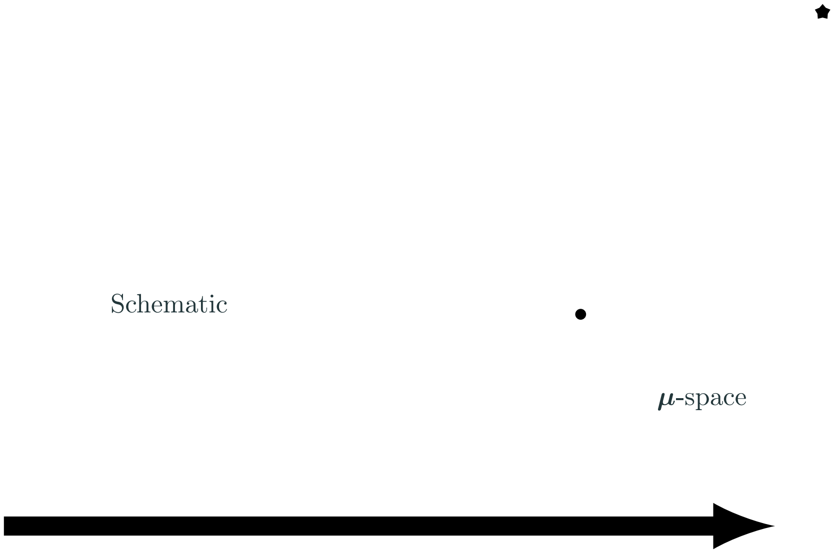
Manage inexactness with trust region method

- Embedded in globally convergent **trust region** method
- **Error indicators**[1] to account for *all* sources of inexactness
- **Refinement** of approximation model using *greedy algorithms*

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \;\; F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \begin{aligned} &\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} && m_k(\boldsymbol{\mu}) \\ &\text{subject to} && ||\boldsymbol{\mu} - \boldsymbol{\mu}_k|| \leq \Delta_k \end{aligned}$$

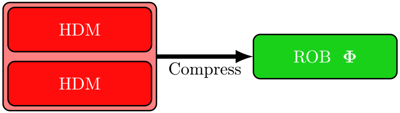
[1] Must be *computable* and apply to general, nonlinear PDEs

Trust region framework for optimization with ROMs

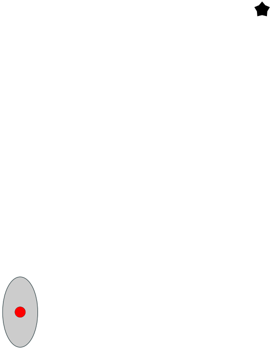


Breakdown of Computational Effort

Trust region framework for optimization with ROMs



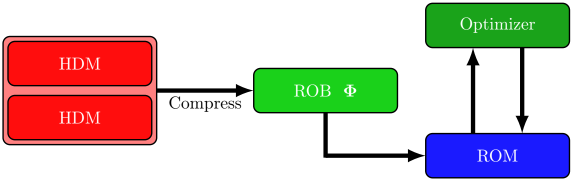
Schematic



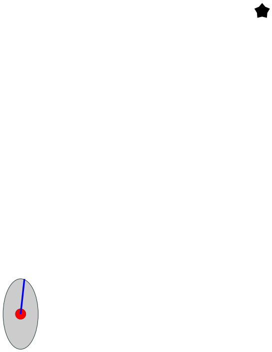
$\boldsymbol{\mu}$-space

Breakdown of Computational Effort

Trust region framework for optimization with ROMs



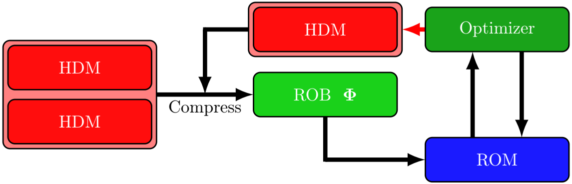
Schematic



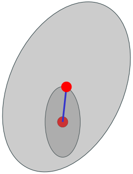
$\boldsymbol{\mu}$-space

Breakdown of Computational Effort

Trust region framework for optimization with ROMs



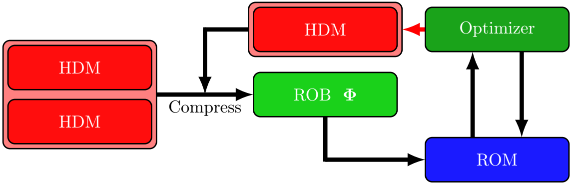
Schematic



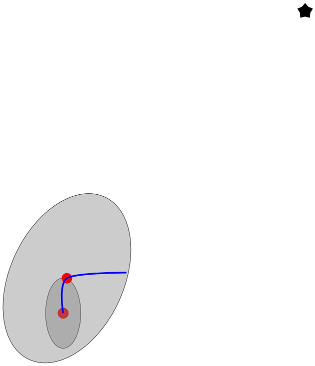
$\boldsymbol{\mu}$-space

Breakdown of Computational Effort

Trust region framework for optimization with ROMs



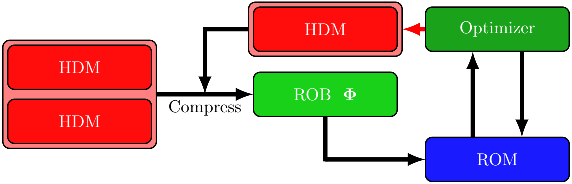
Schematic



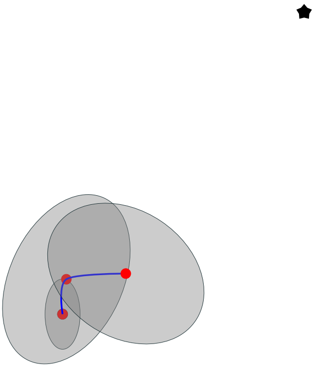
μ-space

Breakdown of Computational Effort

Trust region framework for optimization with ROMs



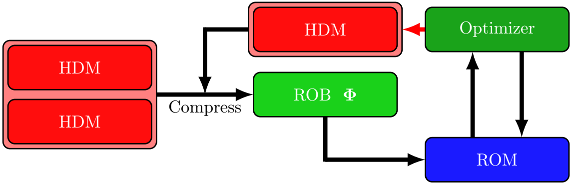
Schematic



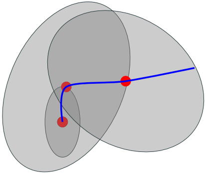
$\boldsymbol{\mu}$-space

Breakdown of Computational Effort

Trust region framework for optimization with ROMs



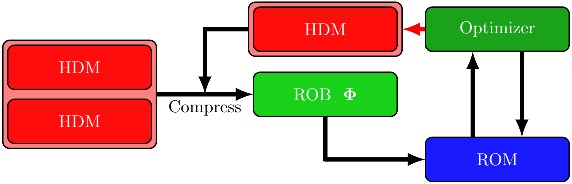
Schematic



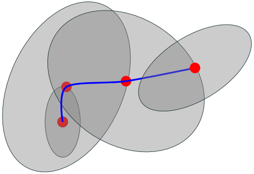
μ-space

Breakdown of Computational Effort

Trust region framework for optimization with ROMs



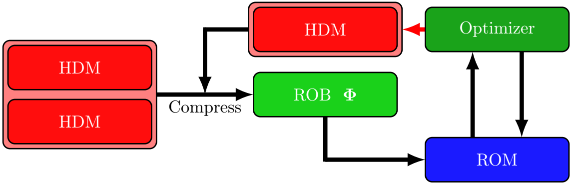
Schematic



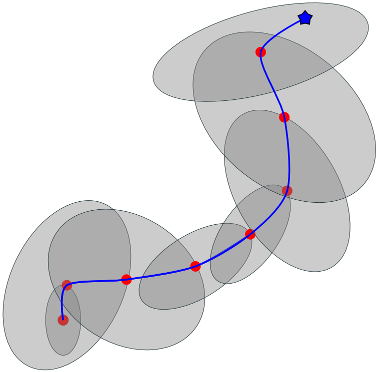
μ-space

Breakdown of Computational Effort

Trust region framework for optimization with ROMs



Schematic

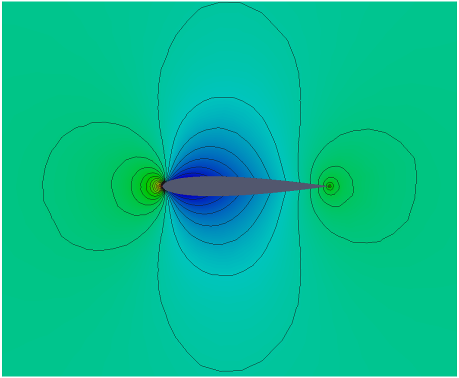


$\boldsymbol{\mu}$-space

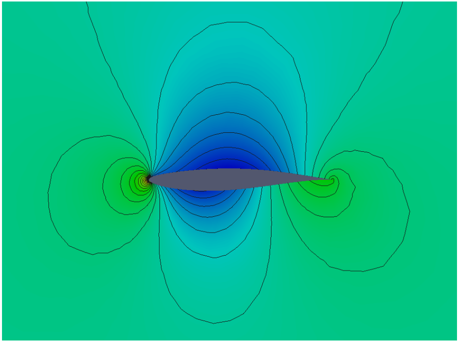
Breakdown of Computational Effort

Compressible, inviscid airfoil design

Pressure discrepancy minimization (Euler equations)



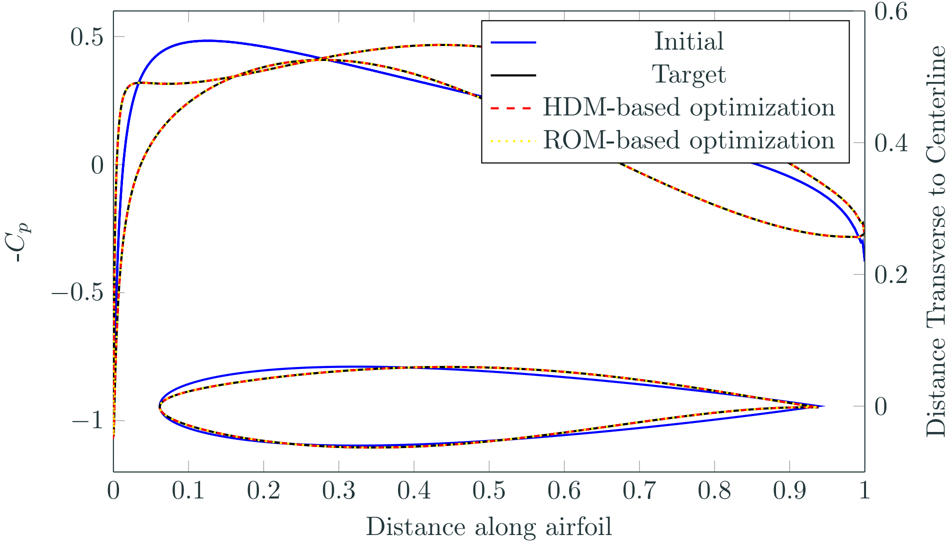
NACA0012: Initial



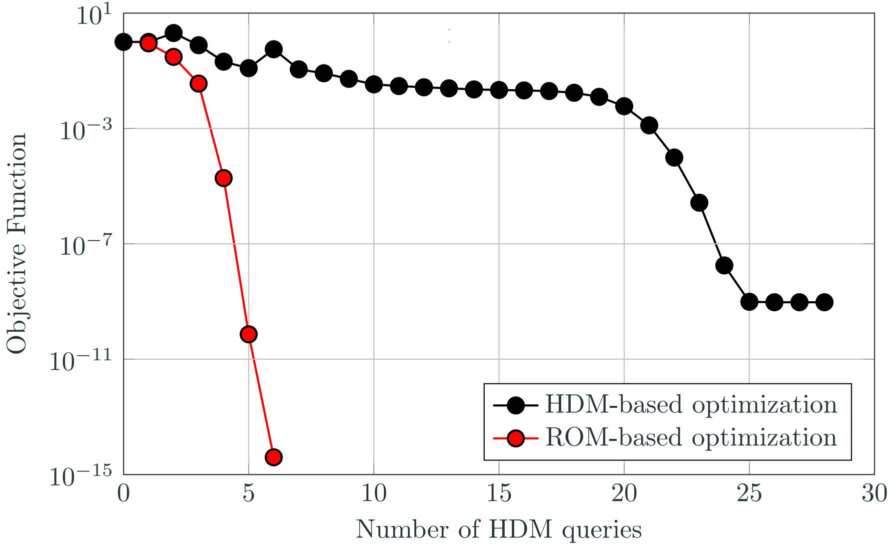
RAE2822: Target

Pressure field for airfoil configurations at $M_\infty = 0.5$, $\alpha = 0.0^\circ$

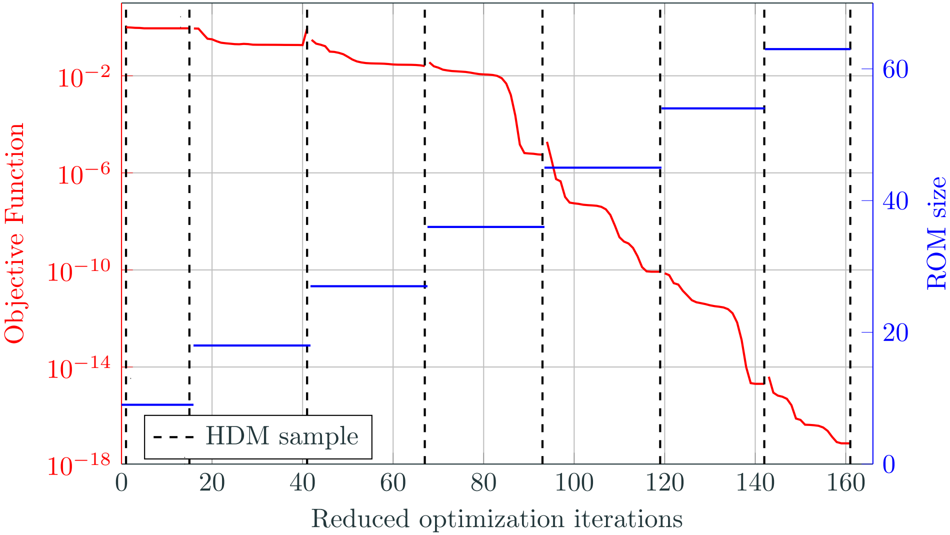
Proposed method: recovers target airfoil



Proposed method: $4\times$ fewer HDM queries



At the cost of ROM queries

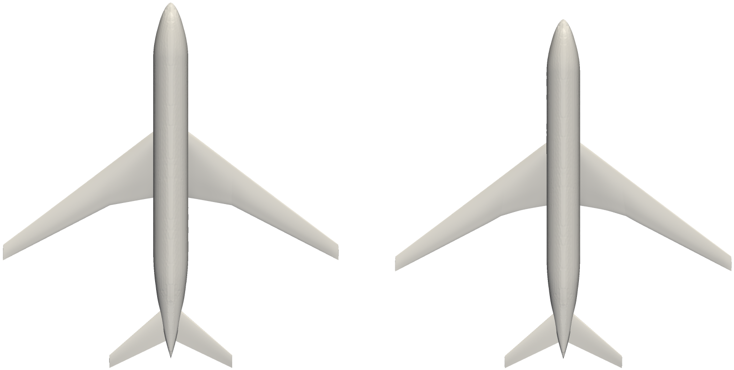


Shape optimization of aircraft in turbulent flow

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^4}{\text{minimize}} \quad -L_z(\boldsymbol{\mu})/L_x(\boldsymbol{\mu})$$
$$\text{subject to} \quad L_z(\boldsymbol{\mu}) = \bar{L}_z$$

- **Flow**: $M = 0.85 \quad \alpha = 2.32^\circ \quad Re = 5 \times 10^6$
- **Equations**: RANS with Spalart-Allmaras
- **Solver**: Vertex-centered finite volume method
- **Mesh**: **11.5M** nodes, **68M** tetra, **69M** DOF

$$\boldsymbol{\mu} = [\mathbf{L} \quad r_x \quad \phi \quad r_z]$$

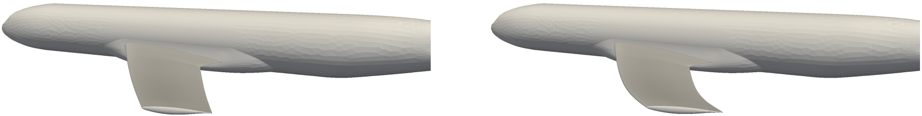


Shape optimization of aircraft in turbulent flow

$$\underset{\boldsymbol{\mu} \in \mathbb{R}^4}{\text{minimize}} \quad -L_z(\boldsymbol{\mu})/L_x(\boldsymbol{\mu})$$
$$\text{subject to} \quad L_z(\boldsymbol{\mu}) = \bar{L}_z$$

- **Flow**: $M = 0.85 \quad \alpha = 2.32^\circ \quad Re = 5 \times 10^6$
- **Equations**: RANS with Spalart-Allmaras
- **Solver**: Vertex-centered finite volume method
- **Mesh**: **11.5M** nodes, **68M** tetra, **69M** DOF

$$\boldsymbol{\mu} = [L \quad \mathbf{r_x} \quad \phi \quad r_z]$$



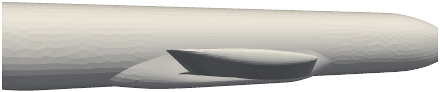
Localized sweep

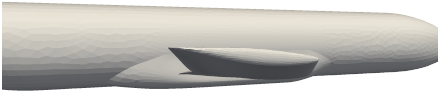
Shape optimization of aircraft in turbulent flow

$$\underset{\boldsymbol{\mu} \in \mathbb{R}^4}{\text{minimize}} \quad -L_z(\boldsymbol{\mu})/L_x(\boldsymbol{\mu})$$
$$\text{subject to} \quad L_z(\boldsymbol{\mu}) = \bar{L}_z$$

- **Flow**: $M = 0.85 \quad \alpha = 2.32^\circ \quad Re = 5 \times 10^6$
- **Equations**: RANS with Spalart-Allmaras
- **Solver**: Vertex-centered finite volume method
- **Mesh**: **11.5M** nodes, **68M** tetra, **69M** DOF

$$\boldsymbol{\mu} = [L \quad r_x \quad \phi \quad r_z]$$





Twist

Shape optimization of aircraft in turbulent flow

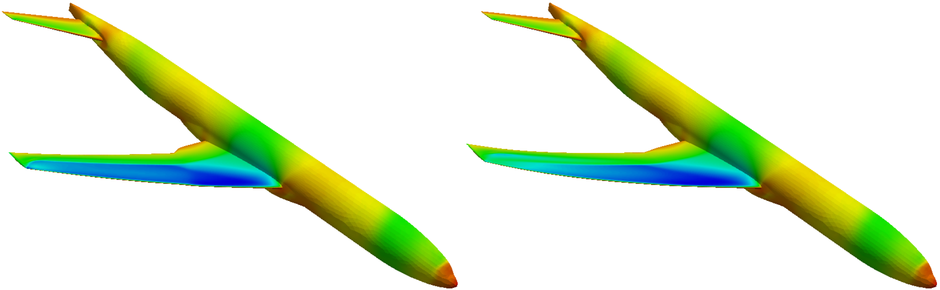
$$\underset{\boldsymbol{\mu}\in\mathbb{R}^4}{\text{minimize}} \quad -L_z(\boldsymbol{\mu})/L_x(\boldsymbol{\mu})$$
$$\text{subject to} \quad L_z(\boldsymbol{\mu}) = \bar{L}_z$$

- **Flow**: $M = 0.85 \quad \alpha = 2.32^\circ \quad Re = 5 \times 10^6$
- **Equations**: RANS with Spalart-Allmaras
- **Solver**: Vertex-centered finite volume method
- **Mesh**: **11.5M** nodes, **68M** tetra, **69M** DOF

$$\boldsymbol{\mu} = [L \quad r_x \quad \phi \quad \mathbf{r_z}]$$

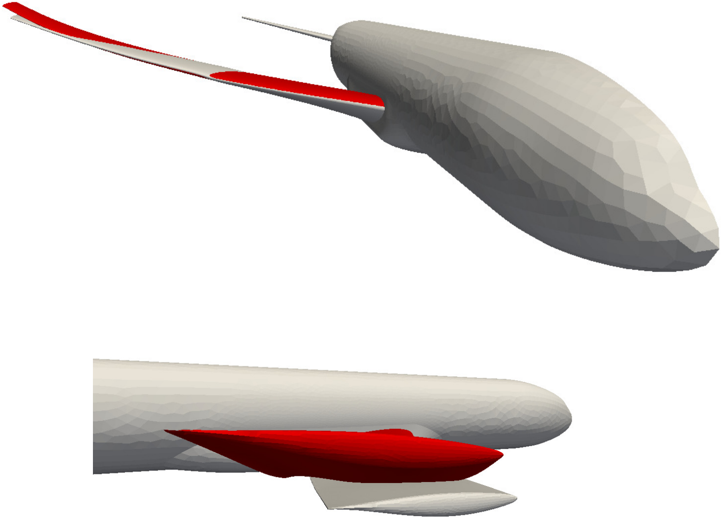
Localized dihedral

Optimized shape: reduction in 2.2 drag counts



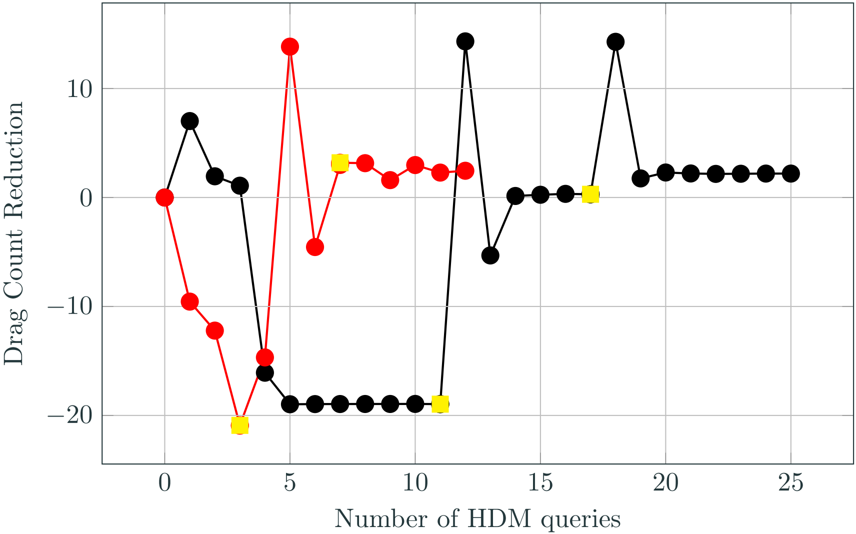
Baseline (left) and optimized (right) shape – colored by C_p

Optimized shape: reduction in 2.2 drag counts



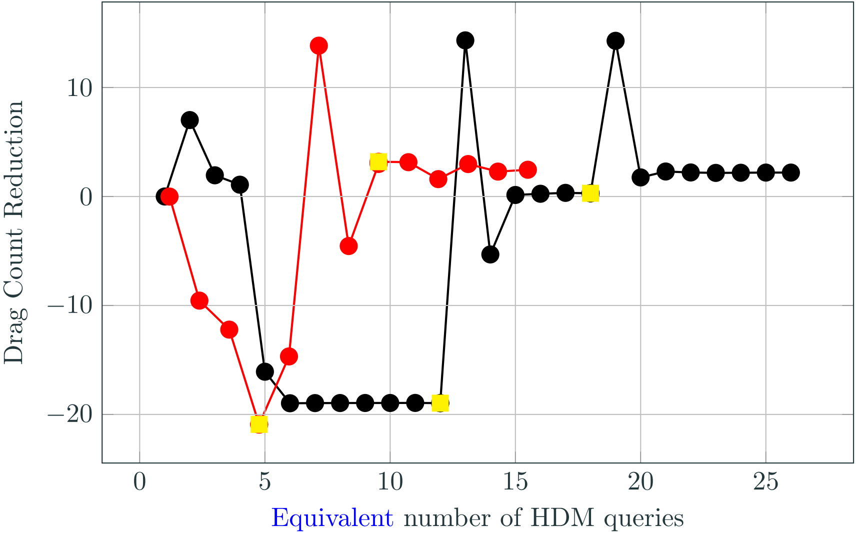
Baseline (gray) and optimized shape (red) – $2\times$ magnification

Proposed method: 2x reduction in number of HDM queries



(—●—) HDM optimization, (—●—) ROM optimization,
(■) Augmented Lagrangian update

Proposed method: 1.6x reduction in overall cost



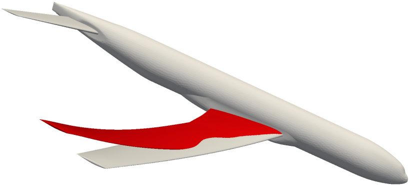
(—●—) HDM optimization, (—●—) ROM optimization,
(■) Augmented Lagrangian update

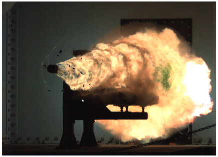
High-order methods for PDE-constrained optimization

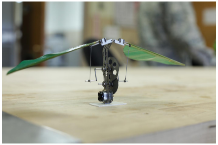
- Developed **fully discrete adjoint method** for **high-order** numerical discretizations of PDEs and QoIs
- Used to compute **gradients** of QoI for use in gradient-based numerical optimization method
- Explicit enforcement of **time-periodicity constraints**
- Extension to **multiphysics** (fluid-structure interaction)
- Applications: optimal flapping flight and energy harvesting, data assimilation

Leveraging inexactness to accelerate PDE-constrained optimization

- Framework introduced for accelerating **deterministic** and **stochastic** PDE-constrained optimization problems
  - Adaptive *model reduction*
  - *Partially converged* primal and adjoint solutions
  - Dimension-adaptive *sparse grids*
- Inexactness **managed** with flexible **trust region** method
- Applied to variety of problems in computational mechanics and outperforms state-of-the-art methods
  - **1.6**× speedup on (deterministic) shape design of aircraft
  - **100**× speedup on (stochastic) optimal control of 1D flow



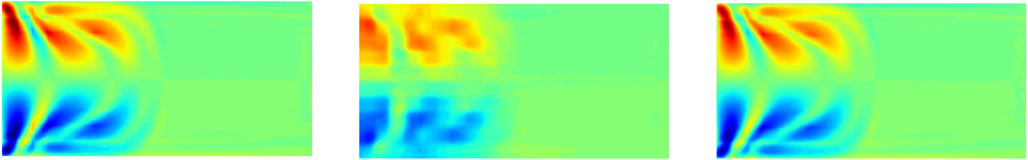




Outlook: Connection to flow reconstruction from MRI images

Goal: Use computational physics and optimization to reconstruction high-resolution blood flow from low-resolution, noisy MRI images

- Fully discrete adjoint method and PDE-constrained optimization
  - single physics: static vessels
  - multiphysics: objects where flow induces significant deformation or involves other types of physics, i.e., electromagnetics
- Huge computational cost of data assimilation in 4D, particularly multiphysics
  - high performance computing
  - globally convergent model reduction
- Extension: Bayesian inference and importance sampling
  - compute probability distribution over set of high-resolution flows instead of single flow [Morzfeld, Chorin]



True high-resolution flow | Low-resolution flow data | Reconstruction

Gerstner, T. and Griebel, M. (2003).
Dimension–adaptive tensor–product quadrature.
Computing, 71(1):65–87.

Kouri, D. P., Heinkenschloss, M., Ridzal, D., and van Bloemen Waanders, B. G. (2013).
A trust-region algorithm with adaptive stochastic collocation for pde optimization under uncertainty.
SIAM Journal on Scientific Computing, 35(4):A1847–A1879.

Kouri, D. P., Heinkenschloss, M., Ridzal, D., and van Bloemen Waanders, B. G. (2014).
Inexact objective function evaluations in a trust-region algorithm for PDE-constrained optimization under uncertainty.
SIAM Journal on Scientific Computing, 36(6):A3011–A3029.

References II

Moré, J. J. (1983).
Recent developments in algorithms and software for trust region methods.
Springer.

Washabaugh, K. (2016).
Faster Fidelity For Better Design: A Scalable Model Order Reduction Framework For Steady Aerodynamic Design Applications.
PhD thesis, Stanford University.

Zahr, M. J., Avery, P., and Farhat, C. (2016a).
A multilevel projection-based model order reduction framework for nonlinear dynamic multiscale problems in structural and solid mechanics.
International Journal for Numerical Methods in Engineering.

References III

Zahr, M. J., Carlberg, K., and Kouri, D. P. (2016b).
Adaptive stochastic collocation for PDE-constrained optimization under uncertainty using sparse grids and model reduction.
SIAM Journal on Uncertainty Quantification.

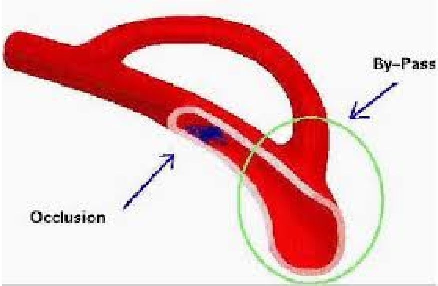
Zahr, M. J. and Persson, P.-O. (2016).
An adjoint method for a high-order discretization of deforming domain conservation laws for optimization of flow problems.
Journal of Computational Physics.

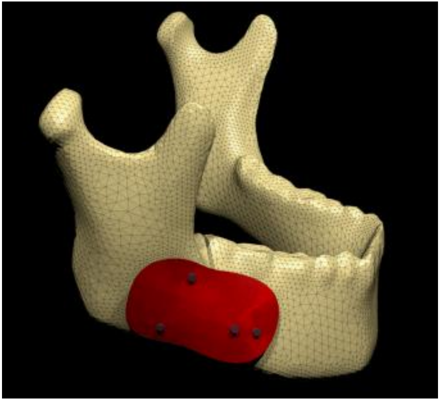
Zahr, M. J. and Persson, P.-O. (2017).
Energetically optimal flapping wing motions via adjoint-based optimization and high-order discretizations.
In *Frontiers in PDE-Constrained Optimization*. Springer.

Zahr, M. J., Persson, P.-O., and Wilkening, J. (2016c).
A fully discrete adjoint method for optimization of flow problems on deforming domains with time-periodicity constraints.
Computers & Fluids.

PDE optimization is ubiquitous in science and engineering

Design: Find system that optimizes performance metric, satisfies constraints





Shape design of arterial bypass (left) and shape/topology design of patient-specific implant (right)

- **Arbitrary Lagrangian-Eulerian** Formulation:
  Map, $\mathcal{G}(\cdot, \boldsymbol{\mu}, t)$, from physical $v(\boldsymbol{\mu}, t)$ to reference V

$$\left.\frac{\partial \boldsymbol{U}_{\boldsymbol{X}}}{\partial t}\right|_{\boldsymbol{X}} + \nabla_{\boldsymbol{X}} \cdot \boldsymbol{F}_{\boldsymbol{X}}(\boldsymbol{U}_{\boldsymbol{X}},\ \nabla_{\boldsymbol{X}} \boldsymbol{U}_{\boldsymbol{X}}) = 0$$

- **Space Discretization**: Discontinuous Galerkin

$$\boldsymbol{M}\frac{\partial \boldsymbol{u}}{\partial t} = \boldsymbol{r}(\boldsymbol{u}, \boldsymbol{\mu}, t)$$

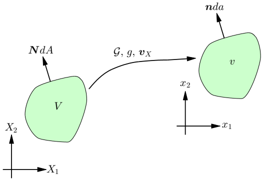
- **Time Discretization**: Diagonally Implicit RK

$$\boldsymbol{u}_n = \boldsymbol{u}_{n-1} + \sum_{i=1}^{s} b_i \boldsymbol{k}_{n,i}$$

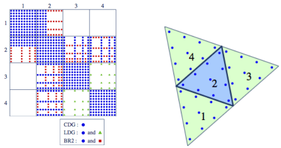
$$\boldsymbol{M}\boldsymbol{k}_{n,i} = \Delta t_n \boldsymbol{r}\left(\boldsymbol{u}_{n,i},\ \boldsymbol{\mu},\ t_{n,i}\right)$$

- **Quantity of Interest**: Solver-consistent

$$F(\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{N_t}, \boldsymbol{k}_{1,1}, \ldots, \boldsymbol{k}_{N_t,s})$$



Mapping-Based ALE



DG Discretization

$$\begin{array}{c|cccc}
c_1 & a_{11} & & & \\
c_2 & a_{21} & a_{22} & & \\
\vdots & \vdots & \vdots & \ddots & \\
c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\
\hline
 & b_1 & b_2 & \cdots & b_s
\end{array}$$

Butcher Tableau for DIRK

Adjoint equation derivation: outline

- Define **auxiliary** PDE-constrained optimization problem

$$\begin{aligned}
\underset{\substack{\boldsymbol{u}_0,\ \ldots,\ \boldsymbol{u}_{N_t} \in \mathbb{R}^{N_{\boldsymbol{u}}},\\ \boldsymbol{k}_{1,1},\ \ldots,\ \boldsymbol{k}_{N_t,s} \in \mathbb{R}^{N_{\boldsymbol{u}}}}}{\text{minimize}} \quad & F(\boldsymbol{u}_0,\ \ldots,\ \boldsymbol{u}_{N_t},\ \boldsymbol{k}_{1,1},\ \ldots,\ \boldsymbol{k}_{N_t,s},\ \boldsymbol{\mu}) \\
\text{subject to} \quad & \boldsymbol{R}_0 = \boldsymbol{u}_0 - \boldsymbol{g}(\boldsymbol{\mu}) = 0 \\
& \boldsymbol{R}_n = \boldsymbol{u}_n - \boldsymbol{u}_{n-1} - \sum_{i=1}^{s} b_i \boldsymbol{k}_{n,i} = 0 \\
& \boldsymbol{R}_{n,i} = \boldsymbol{M}\boldsymbol{k}_{n,i} - \Delta t_n \boldsymbol{r}\left(\boldsymbol{u}_{n,i},\ \boldsymbol{\mu},\ t_{n,i}\right) = 0
\end{aligned}$$

- Define **Lagrangian**

$$\mathcal{L}(\boldsymbol{u}_n,\ \boldsymbol{k}_{n,i},\ \boldsymbol{\lambda}_n,\ \boldsymbol{\kappa}_{n,i}) = F - {\boldsymbol{\lambda}_0}^T \boldsymbol{R}_0 - \sum_{n=1}^{N_t} {\boldsymbol{\lambda}_n}^T \boldsymbol{R}_n - \sum_{n=1}^{N_t} \sum_{i=1}^{s} {\boldsymbol{\kappa}_{n,i}}^T \boldsymbol{R}_{n,i}$$

- The solution of the optimization problem is given by the **Karush-Kuhn-Tucker (KKT) sytem**

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{u}_n} = 0, \quad \frac{\partial \mathcal{L}}{\partial \boldsymbol{k}_{n,i}} = 0, \quad \frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}_n} = 0, \quad \frac{\partial \mathcal{L}}{\partial \boldsymbol{\kappa}_{n,i}} = 0$$

Structure: semi-discretization, first-order form

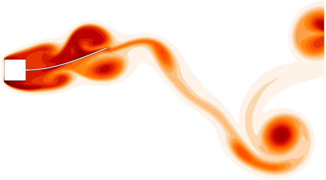
$$\boldsymbol{M}^s \frac{\partial \boldsymbol{u}^s}{\partial t} = \boldsymbol{r}^s(\boldsymbol{u}^s; \boldsymbol{t}) = \boldsymbol{r}^{ss}(\boldsymbol{u}^s) + \boldsymbol{r}^{sf} \cdot \boldsymbol{t}$$

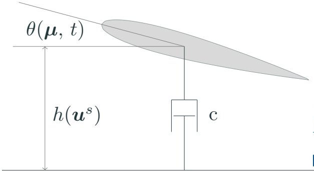
- Semidiscretization (CG-FEM) of **continuum** (hyperelasticity)

$$\begin{aligned} \frac{\partial \boldsymbol{p}}{\partial t} - \nabla \cdot \boldsymbol{P}(\boldsymbol{G}) &= \boldsymbol{b} \qquad &\text{in } \Omega_0 \\ \boldsymbol{P}(\boldsymbol{G}) \cdot \boldsymbol{N} &= \boldsymbol{t} \qquad &\text{on } \Gamma_N \\ \boldsymbol{x} &= \boldsymbol{x}_D \qquad &\text{on } \Gamma_D \end{aligned}$$

- Force balance on **rigid body**

$$\boldsymbol{M} \frac{\partial^2 \boldsymbol{q}}{\partial t^2} + \boldsymbol{C} \frac{\partial \boldsymbol{q}}{\partial t} + \boldsymbol{K} \boldsymbol{q} = \boldsymbol{t}$$





Trust region ingredients for global convergence

Approximation model

$$m_k(\boldsymbol{\mu})$$

Error indicator

$$||\nabla F(\boldsymbol{\mu}) - \nabla m_k(\boldsymbol{\mu})|| \leq \xi \varphi_k(\boldsymbol{\mu}) \qquad \xi > 0$$

Adaptivity

$$\varphi_k(\boldsymbol{\mu}_k) \leq \kappa_\varphi \min\{||\nabla m_k(\boldsymbol{\mu}_k)|| \,, \, \Delta_k\}$$

Global convergence

$$\liminf_{k \to \infty} \; ||\nabla F(\boldsymbol{\mu}_k)|| = 0$$

Proposed approach: managed inexactness

Replace expensive PDE with inexpensive approximation model

- **Reduced-order models** used for *inexact PDE evaluations*

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad m_k(\boldsymbol{\mu})$$

Manage inexactness with trust region method

- Embedded in globally convergent **trust region** method
- **Error indicators** to account for *all* sources of inexactness
- **Refinement** of approximation model using *greedy algorithms*

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \begin{aligned} &\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} && m_k(\boldsymbol{\mu}) \\ &\text{subject to} && ||\boldsymbol{\mu} - \boldsymbol{\mu}_k|| \leq \Delta_k \end{aligned}$$

Source of inexactness: projection-based model reduction

- Model reduction ansatz: *state vector lies in low-dimensional subspace*

$$\boldsymbol{u} \approx \boldsymbol{\Phi} \boldsymbol{u}_r$$

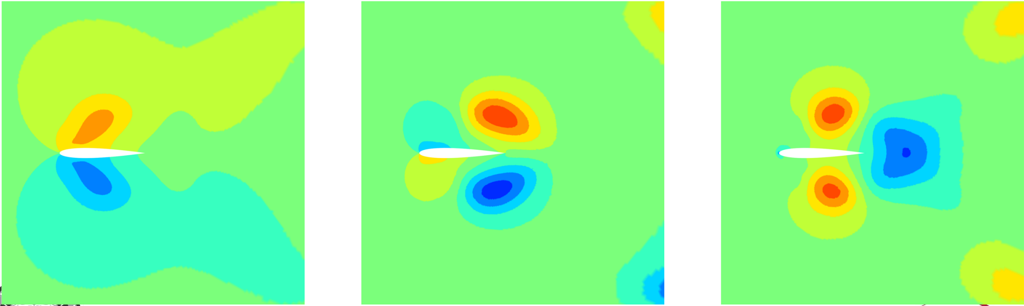
  - $\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\phi}^1 & \cdots & \boldsymbol{\phi}^{k_{\boldsymbol{u}}} \end{bmatrix} \in \mathbb{R}^{n_{\boldsymbol{u}} \times k_{\boldsymbol{u}}}$ is the reduced (trial) basis ($n_{\boldsymbol{u}} \gg k_{\boldsymbol{u}}$)
  - $\boldsymbol{u}_r \in \mathbb{R}^{k_{\boldsymbol{u}}}$ are the reduced coordinates of $\boldsymbol{u}$

- Substitute into $\boldsymbol{r}(\boldsymbol{u}, \boldsymbol{\mu}) = 0$ and project onto columnspace of a test basis $\boldsymbol{\Psi} \in \mathbb{R}^{n_{\boldsymbol{u}} \times k_{\boldsymbol{u}}}$ to obtain a square system

$$\boldsymbol{\Psi}^T \boldsymbol{r}(\boldsymbol{\Phi} \boldsymbol{u}_r, \boldsymbol{\mu}) = 0$$

Few global, data-driven basis functions v. many local ones

- Instead of using traditional *local* shape functions, use **global shape functions**
- Instead of a-priori, analytical shape functions, leverage data-rich computing environment by using **data-driven modes**



Trust region method: ROM approximation model

Approximation models based on reduced-order models

$$m_k(\boldsymbol{\mu}) = \mathcal{J}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})$$

Error indicators from residual-based error bounds

$$\varphi_k(\boldsymbol{\mu}) = \left|\left|\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})\right|\right|_{\boldsymbol{\Theta}} + \left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\Psi}_k \boldsymbol{\lambda}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})\right|\right|_{\boldsymbol{\Theta}^{\boldsymbol{\lambda}}}$$

Adaptivity to refine basis at trust region center

$$\boldsymbol{\Phi}_k = \begin{bmatrix} \boldsymbol{u}(\boldsymbol{\mu}_k) & \boldsymbol{\lambda}(\boldsymbol{\mu}_k) & \texttt{POD}(\boldsymbol{U}_k) & \texttt{POD}(\boldsymbol{V}_k) \end{bmatrix}$$

$$\boldsymbol{U}_k = \begin{bmatrix} \boldsymbol{u}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{u}(\boldsymbol{\mu}_{k-1}) \end{bmatrix} \qquad \boldsymbol{V}_k = \begin{bmatrix} \boldsymbol{\lambda}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{\lambda}(\boldsymbol{\mu}_{k-1}) \end{bmatrix}$$

$$\text{Interpolation property} \implies \varphi_k(\boldsymbol{\mu}_k) = 0$$

Trust region method: ROM approximation model

Approximation models based on reduced-order models

$$m_k(\boldsymbol{\mu}) = \mathcal{J}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})$$

Error indicators from residual-based error bounds

$$\varphi_k(\boldsymbol{\mu}) = ||\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})||_{\boldsymbol{\Theta}} + \left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\Psi}_k \boldsymbol{\lambda}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})\right|\right|_{\boldsymbol{\Theta}^{\boldsymbol{\lambda}}}$$

Adaptivity to refine basis at trust region center

$$\boldsymbol{\Phi}_k = \begin{bmatrix} \boldsymbol{u}(\boldsymbol{\mu}_k) & \boldsymbol{\lambda}(\boldsymbol{\mu}_k) & \texttt{POD}(\boldsymbol{U}_k) & \texttt{POD}(\boldsymbol{V}_k) \end{bmatrix}$$

$$\boldsymbol{U}_k = \begin{bmatrix} \boldsymbol{u}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{u}(\boldsymbol{\mu}_{k-1}) \end{bmatrix} \qquad \boldsymbol{V}_k = \begin{bmatrix} \boldsymbol{\lambda}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{\lambda}(\boldsymbol{\mu}_{k-1}) \end{bmatrix}$$

$$\text{Interpolation property} \implies \varphi_k(\boldsymbol{\mu}_k) = 0$$

$$\liminf_{k\to\infty} \; ||\nabla \mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu}_k),\, \boldsymbol{\mu}_k)|| = 0$$

Overview of global convergence theory[2]

Let $\{\boldsymbol{\mu}_k\}$ be a sequence of iterates produced by the algorithm and suppose there exists $\epsilon > 0$ such that $||\nabla m_k(\boldsymbol{\mu}_k)|| > 0$

Lemma 1: $\Delta_k \to 0$

- Fraction of Cauchy decrease
- $|F(\boldsymbol{\mu}_k) - F(\hat{\boldsymbol{\mu}}_k) + \psi_k(\hat{\boldsymbol{\mu}}_k) - \psi_k(\boldsymbol{\mu}_k)| \leq \sigma\left[\eta \min\{m_k(\boldsymbol{\mu}_k) - m_k(\hat{\boldsymbol{\mu}}_k),\, r_k\}\right]^{1/\omega}$

Lemma 2: $\rho_k \to 1$

- Fraction of Cauchy decrease
- $|F(\boldsymbol{\mu}_k) - F(\hat{\boldsymbol{\mu}}_k) + m_k(\hat{\boldsymbol{\mu}}_k) - m_k(\boldsymbol{\mu}_k)| \leq \zeta \Delta_k$

Theorem 1: $\liminf ||\nabla F(\boldsymbol{\mu}_k)|| = 0$

- Contradiction from Lemma 1 and 2 $\implies$ $\liminf ||\nabla m_k(\boldsymbol{\mu}_k)|| = 0$
- $||\nabla F(\boldsymbol{\mu}_k) - \nabla m_k(\boldsymbol{\mu}_k)|| \leq \xi \min\{||\nabla m_k(\boldsymbol{\mu})||\,,\, \Delta_k\}$

[2] Closely parallels convergence theory in [Moré, 1983, Kouri et al., 2014]

Hyperreduction to reduce complexity of nonlinear terms

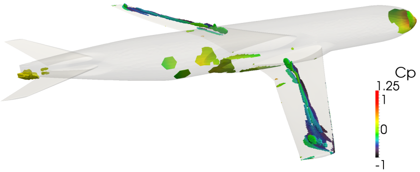
Despite **reduced dimensionality**, $\mathcal{O}(n_{\boldsymbol{u}})$ operations are required to evaluate

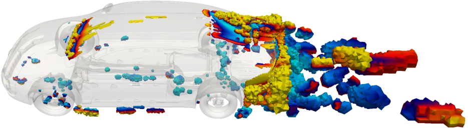
$$\boldsymbol{\Psi}^T \boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r,\, \boldsymbol{\mu}) \qquad \boldsymbol{\Psi}^T \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{u}}(\boldsymbol{\Phi}\boldsymbol{u}_r,\, \boldsymbol{\mu})\boldsymbol{\Phi}$$

Solution: only perform minimization over a *subset* of the spatial domain

$$\underset{\boldsymbol{u}_r \in \mathbb{R}^{k_{\boldsymbol{u}}}}{\text{minimize}} \quad ||\boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r,\, \boldsymbol{\mu})||_{\boldsymbol{\Theta}} \implies \underset{\boldsymbol{u}_r \in \mathbb{R}^{k_{\boldsymbol{u}}}}{\text{minimize}} \quad \left|\left|\boldsymbol{P}^T \boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r,\, \boldsymbol{\mu})\right|\right|_{\boldsymbol{\Theta}}$$

and **hyperreduced** model[3] is independent of $n_{\boldsymbol{u}}$





Sample mesh for CRM (left) and Passat (right) [Washabaugh, 2016]

[3] Masked minimum-residual property and weaker definitions of optimality, monotonicity, and interpolation hold

Proposed approach: managed inexactness

Replace expensive PDE with inexpensive approximation model

- **Reduced-order models** used for *inexact PDE evaluations*
- **Partially converged solutions** used for *inexact PDE evaluations*
- **Anisotropic sparse grids** used for *inexact integration* of risk measures

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad m_k(\boldsymbol{\mu})$$

Manage inexactness with trust region method

- Embedded in globally convergent **trust region** method
- **Error indicators** to account for *all* sources of inexactness
- **Refinement** of approximation model using *greedy algorithms*

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \begin{array}{ll} \underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} & m_k(\boldsymbol{\mu}) \\ \text{subject to} & ||\boldsymbol{\mu}-\boldsymbol{\mu}_k|| \leq \Delta_k \end{array}$$

Source of inexactness: Partially Converged Solution (PCS)

A τ-partially converged primal solution $\boldsymbol{u}^{\tau}(\boldsymbol{\mu})$ is any $\boldsymbol{u}$ satisfying

$$||\boldsymbol{r}(\boldsymbol{u},\, \boldsymbol{\mu})||_{\boldsymbol{\Theta}} \leq \tau$$

A τ_1-τ_2-partially converged adjoint solution $\boldsymbol{\lambda}^{\tau_1,\,\tau_2}(\boldsymbol{\mu})$ is any $\boldsymbol{\lambda}$ satisfying

$$\left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{u}^{\tau_1}(\boldsymbol{\mu}),\, \boldsymbol{\lambda},\, \boldsymbol{\mu})\right|\right|_{\boldsymbol{\Theta}^{\boldsymbol{\lambda}}} \leq \tau_2$$

Trust region method: ROM/PCS approximation model

Approximation models based on ROMs and partially converged solutions

$$m_k(\boldsymbol{\mu}) = \mathcal{J}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}), \boldsymbol{\mu}) \qquad \psi_k(\boldsymbol{\mu}) = \mathcal{J}(\boldsymbol{u}^{\tau_k}(\boldsymbol{\mu}), \boldsymbol{\mu})$$

Error indicators from residual-based error bounds

$$\begin{aligned}
\vartheta_k(\boldsymbol{\mu}) &= \left\|\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}_k), \boldsymbol{\mu}_k)\right\|_{\boldsymbol{\Theta}} + \left\|\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}), \boldsymbol{\mu})\right\|_{\boldsymbol{\Theta}} \\
\varphi_k(\boldsymbol{\mu}) &= \left\|\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}), \boldsymbol{\mu})\right\|_{\boldsymbol{\Theta}} + \left\|\boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}), \boldsymbol{\Psi}_k \boldsymbol{\lambda}_r(\boldsymbol{\mu}), \boldsymbol{\mu})\right\|_{\boldsymbol{\Theta}^{\boldsymbol{\lambda}}} \\
\theta_k(\boldsymbol{\mu}) &= \left\|\boldsymbol{r}(\boldsymbol{u}^{\tau_k}(\boldsymbol{\mu}_k), \boldsymbol{\mu}_k)\right\|_{\boldsymbol{\Theta}} + \left\|\boldsymbol{r}(\boldsymbol{u}^{\tau_k}(\boldsymbol{\mu}), \boldsymbol{\mu})\right\|_{\boldsymbol{\Theta}}
\end{aligned}$$

Adaptivity to refine basis at trust region center

$$\boldsymbol{\Phi}_k = \begin{bmatrix} \boldsymbol{u}^{\alpha_k}(\boldsymbol{\mu}_k) & \boldsymbol{\lambda}^{\alpha_k, \beta_k}(\boldsymbol{\mu}_k) & \texttt{POD}(\boldsymbol{U}_k) & \texttt{POD}(\boldsymbol{V}_k) \end{bmatrix}$$

$$\boldsymbol{U}_k = \begin{bmatrix} \boldsymbol{u}^{\alpha_0}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{u}^{\alpha_{k-1}}(\boldsymbol{\mu}_{k-1}) \end{bmatrix} \qquad \boldsymbol{V}_k = \begin{bmatrix} \boldsymbol{\lambda}^{\alpha_0, \beta_0}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{\lambda}^{\alpha_{k-1}, \beta_{k-1}}(\boldsymbol{\mu}_{k-1}) \end{bmatrix}$$

and α_k, β_k, τ_k selected such that

$$\vartheta_k(\boldsymbol{\mu}_k) \leq \kappa_\vartheta \Delta_k \qquad \varphi_k(\boldsymbol{\mu}_k) \leq \kappa_\varphi \min\{\left\|\nabla m_k(\boldsymbol{\mu}_k)\right\|, \Delta_k\}$$

$$\theta_k^\omega(\hat{\boldsymbol{\mu}}_k) \leq \eta \min\{m_k(\boldsymbol{\mu}_k) - m_k(\hat{\boldsymbol{\mu}}_k), r_k\}$$

Trust region method: ROM/PCS approximation model

Approximation models based on ROMs and partially converged solutions

$$m_k(\boldsymbol{\mu}) = \mathcal{J}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu}) \qquad \psi_k(\boldsymbol{\mu}) = \mathcal{J}(\boldsymbol{u}^{\tau_k}(\boldsymbol{\mu}),\, \boldsymbol{\mu})$$

Error indicators from residual-based error bounds

$$\vartheta_k(\boldsymbol{\mu}) = ||\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}_k),\, \boldsymbol{\mu}_k)||_{\boldsymbol{\Theta}} + ||\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})||_{\boldsymbol{\Theta}}$$

$$\varphi_k(\boldsymbol{\mu}) = ||\boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})||_{\boldsymbol{\Theta}} + \left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}),\, \boldsymbol{\Psi}_k \boldsymbol{\lambda}_r(\boldsymbol{\mu}),\, \boldsymbol{\mu})\right|\right|_{\boldsymbol{\Theta}^{\boldsymbol{\lambda}}}$$

$$\theta_k(\boldsymbol{\mu}) = ||\boldsymbol{r}(\boldsymbol{u}^{\tau_k}(\boldsymbol{\mu}_k),\, \boldsymbol{\mu}_k)||_{\boldsymbol{\Theta}} + ||\boldsymbol{r}(\boldsymbol{u}^{\tau_k}(\boldsymbol{\mu}),\, \boldsymbol{\mu})||_{\boldsymbol{\Theta}}$$

Adaptivity to refine basis at trust region center

$$\boldsymbol{\Phi}_k = \begin{bmatrix} \boldsymbol{u}^{\alpha_k}(\boldsymbol{\mu}_k) & \boldsymbol{\lambda}^{\alpha_k,\,\beta_k}(\boldsymbol{\mu}_k) & \texttt{POD}(\boldsymbol{U}_k) & \texttt{POD}(\boldsymbol{V}_k) \end{bmatrix}$$

$$\boldsymbol{U}_k = \begin{bmatrix} \boldsymbol{u}^{\alpha_0}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{u}^{\alpha_{k-1}}(\boldsymbol{\mu}_{k-1}) \end{bmatrix} \qquad \boldsymbol{V}_k = \begin{bmatrix} \boldsymbol{\lambda}^{\alpha_0,\,\beta_0}(\boldsymbol{\mu}_0) & \cdots & \boldsymbol{\lambda}^{\alpha_{k-1},\,\beta_{k-1}}(\boldsymbol{\mu}_{k-1}) \end{bmatrix}$$

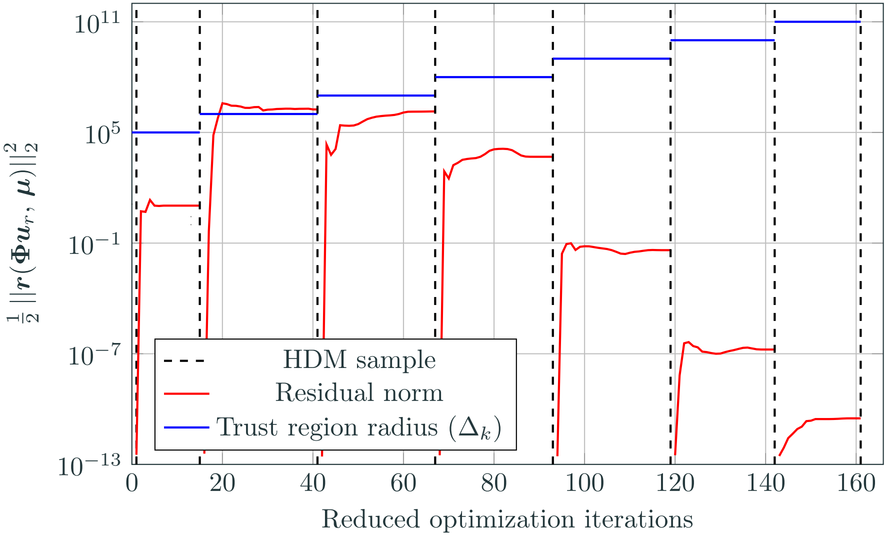
and $\alpha_k,\, \beta_k,\, \tau_k$ selected such that

$$\vartheta_k(\boldsymbol{\mu}_k) \leq \kappa_\vartheta \Delta_k \qquad \varphi_k(\boldsymbol{\mu}_k) \leq \kappa_\varphi \min\{||\nabla m_k(\boldsymbol{\mu}_k)||\,,\, \Delta_k\}$$

$$\theta_k^\omega(\hat{\boldsymbol{\mu}}_k) \leq \eta \min\{m_k(\boldsymbol{\mu}_k) - m_k(\hat{\boldsymbol{\mu}}_k),\, r_k\}$$

$$\liminf_{k\to\infty} \; ||\nabla \mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu}_k),\, \boldsymbol{\mu}_k)|| = 0$$

Error-aware trust region behavior



Source of inexactness: anisotropic sparse grids

1D Quadrature Rules: Define the difference operator

$$\Delta_k^j \equiv \mathbb{E}_k^j - \mathbb{E}_k^{j-1}$$

where $\mathbb{E}_k^0 \equiv 0$ and $\mathbb{E}_k^j$ as the level-j 1d quadrature rule for dimension k

Anisotropic Sparse Grid: Define the index set $\mathcal{I} \subset \mathbb{N}^{n_{\boldsymbol{\xi}}}$ and

$$\mathbb{E}_{\mathcal{I}} \equiv \sum_{\mathbf{i} \in \mathcal{I}} \Delta_1^{i_1} \otimes \cdots \otimes \Delta_{n_{\boldsymbol{\xi}}}^{i_{n_{\boldsymbol{\xi}}}}$$

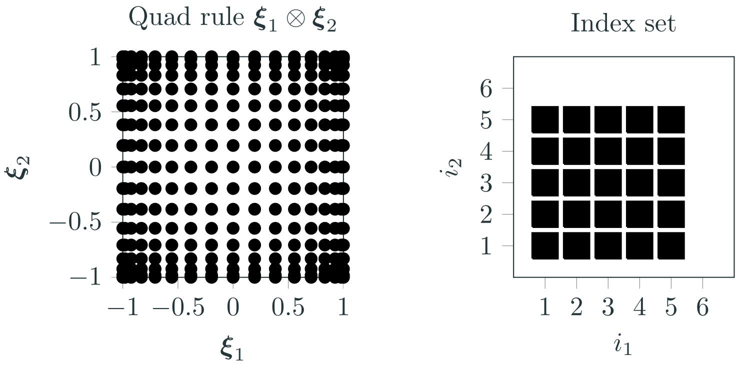
Neighbors: Let $\mathcal{I}^c = \mathbb{N}^{n_{\boldsymbol{\xi}}} \setminus \mathcal{I}$

$$\mathcal{N}(\mathcal{I}) = \{\boldsymbol{i} \in \mathcal{I}^c \mid \boldsymbol{i} - \boldsymbol{e}_j \in \mathcal{I},\ j = 1, \ldots, n_{\boldsymbol{\xi}}\}$$

Truncation Error: [Gerstner and Griebel, 2003, Kouri et al., 2013]

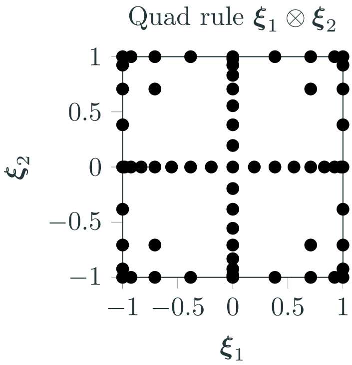
$$\mathbb{E} - \mathbb{E}_{\mathcal{I}} = \sum_{\mathbf{i} \in \mathcal{I}^c} \Delta_1^{i_1} \otimes \cdots \otimes \Delta_{n_{\boldsymbol{\xi}}}^{i_{n_{\boldsymbol{\xi}}}} \approx \sum_{\mathbf{i} \in \mathcal{N}(\mathcal{I})} \Delta_1^{i_1} \otimes \cdots \otimes \Delta_{n_{\boldsymbol{\xi}}}^{i_{n_{\boldsymbol{\xi}}}} = \mathbb{E}_{\mathcal{N}(\mathcal{I})}$$

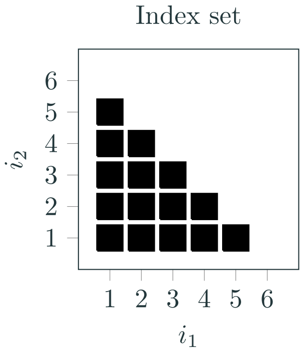
Tensor product quadrature



Index set ($\mathcal{I}$) – ●   Neighbors ($\mathcal{N}(\mathcal{I})$) – ●

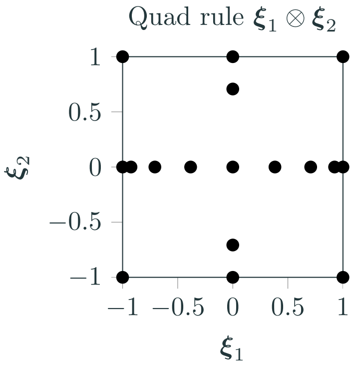
Isotropic sparse grid quadrature

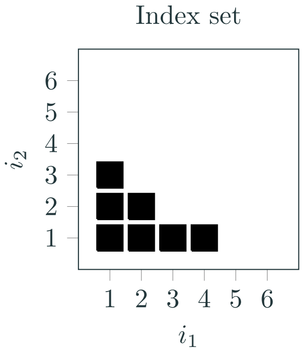




Index set ($\mathcal{I}$) – ● Neighbors ($\mathcal{N}(\mathcal{I})$) – ●

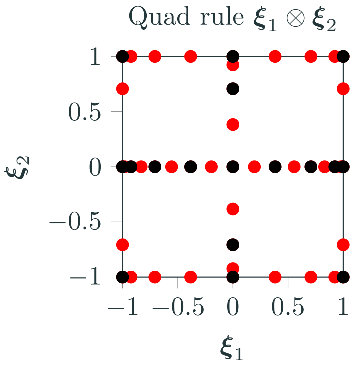
Anisotropic sparse grid quadrature

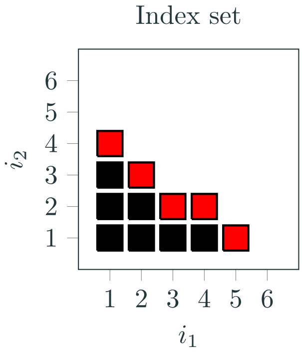




Index set ($\mathcal{I}$) – ●   Neighbors ($\mathcal{N}(\mathcal{I})$) – ●

Anisotropic sparse grid quadrature: neighbors





Index set ($\mathcal{I}$) – ●   Neighbors ($\mathcal{N}(\mathcal{I})$) – ●

Derivation of gradient error indicator

For brevity, let

$$\begin{aligned}
\mathcal{J}(\boldsymbol{\xi}) &\leftarrow \mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\nabla\mathcal{J}(\boldsymbol{\xi}) &\leftarrow \nabla\mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\mathcal{J}_r(\boldsymbol{\xi}) &= \mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\nabla\mathcal{J}_r(\boldsymbol{\xi}) &= \nabla\mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\boldsymbol{r}_r(\boldsymbol{\xi}) &= \boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\boldsymbol{r}_r^{\boldsymbol{\lambda}}(\boldsymbol{\xi}) &= \boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\Psi}\boldsymbol{\lambda}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi})
\end{aligned}$$

Separate total error into contributions from ROM inexactness and SG truncation

$$||\mathbb{E}[\nabla\mathcal{J}] - \mathbb{E}_{\mathcal{I}}[\nabla\mathcal{J}_r]|| \leq \mathbb{E}\left[||\nabla\mathcal{J} - \nabla\mathcal{J}_r||\right] + ||\mathbb{E}\left[\nabla\mathcal{J}_r\right] - \mathbb{E}_{\mathcal{I}}\left[\nabla\mathcal{J}_r\right]||$$

Derivation of gradient error indicator

For brevity, let

$$\begin{aligned}
\mathcal{J}(\boldsymbol{\xi}) &\leftarrow \mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu}, \boldsymbol{\xi}), \boldsymbol{\mu}, \boldsymbol{\xi}) \\
\nabla\mathcal{J}(\boldsymbol{\xi}) &\leftarrow \nabla\mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu}, \boldsymbol{\xi}), \boldsymbol{\mu}, \boldsymbol{\xi}) \\
\mathcal{J}_r(\boldsymbol{\xi}) &= \mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu}, \boldsymbol{\xi}), \boldsymbol{\mu}, \boldsymbol{\xi}) \\
\nabla\mathcal{J}_r(\boldsymbol{\xi}) &= \nabla\mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu}, \boldsymbol{\xi}), \boldsymbol{\mu}, \boldsymbol{\xi}) \\
\boldsymbol{r}_r(\boldsymbol{\xi}) &= \boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu}, \boldsymbol{\xi}), \boldsymbol{\mu}, \boldsymbol{\xi}) \\
\boldsymbol{r}_r^{\boldsymbol{\lambda}}(\boldsymbol{\xi}) &= \boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu}, \boldsymbol{\xi}), \boldsymbol{\Psi}\boldsymbol{\lambda}_r(\boldsymbol{\mu}, \boldsymbol{\xi}), \boldsymbol{\mu}, \boldsymbol{\xi})
\end{aligned}$$

Separate total error into contributions from ROM inexactness and SG truncation

$$\begin{aligned}
||\mathbb{E}[\nabla\mathcal{J}] - \mathbb{E}_{\mathcal{I}}[\nabla\mathcal{J}_r]|| &\leq \mathbb{E}\left[||\nabla\mathcal{J} - \nabla\mathcal{J}_r||\right] + ||\mathbb{E}\left[\nabla\mathcal{J}_r\right] - \mathbb{E}_{\mathcal{I}}\left[\nabla\mathcal{J}_r\right]|| \\
&\leq \zeta'\mathbb{E}\left[\alpha_1\,||\boldsymbol{r}|| + \alpha_2\left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}\right|\right|\right] + \mathbb{E}_{\mathcal{I}^c}\left[||\nabla\mathcal{J}_r||\right]
\end{aligned}$$

Derivation of gradient error indicator

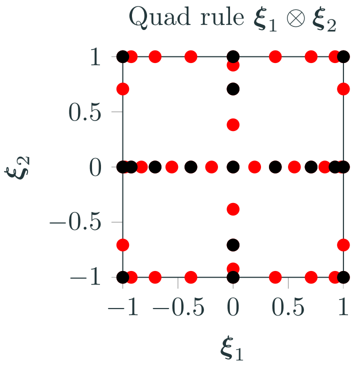
For brevity, let

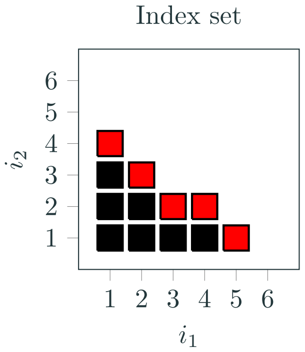
$$\begin{aligned}
\mathcal{J}(\boldsymbol{\xi}) &\leftarrow \mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\nabla\mathcal{J}(\boldsymbol{\xi}) &\leftarrow \nabla\mathcal{J}(\boldsymbol{u}(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\mathcal{J}_r(\boldsymbol{\xi}) &= \mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\nabla\mathcal{J}_r(\boldsymbol{\xi}) &= \nabla\mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\boldsymbol{r}_r(\boldsymbol{\xi}) &= \boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi}) \\
\boldsymbol{r}_r^{\boldsymbol{\lambda}}(\boldsymbol{\xi}) &= \boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\Psi}\boldsymbol{\lambda}_r(\boldsymbol{\mu},\,\boldsymbol{\xi}),\,\boldsymbol{\mu},\,\boldsymbol{\xi})
\end{aligned}$$

Separate total error into contributions from ROM inexactness and SG truncation

$$\begin{aligned}
||\mathbb{E}[\nabla\mathcal{J}] - \mathbb{E}_{\mathcal{I}}[\nabla\mathcal{J}_r]|| &\leq \mathbb{E}\left[||\nabla\mathcal{J} - \nabla\mathcal{J}_r||\right] + ||\mathbb{E}\left[\nabla\mathcal{J}_r\right] - \mathbb{E}_{\mathcal{I}}\left[\nabla\mathcal{J}_r\right]|| \\
&\leq \zeta'\mathbb{E}\left[\alpha_1\,||\boldsymbol{r}|| + \alpha_2\,\left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}\right|\right|\right] + \mathbb{E}_{\mathcal{I}^c}\left[||\nabla\mathcal{J}_r||\right] \\
&\lesssim \zeta\left(\mathbb{E}_{\mathcal{I}\cup\mathcal{N}(\mathcal{I})}\left[\alpha_1\,||\boldsymbol{r}|| + \alpha_2\,\left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}\right|\right|\right] + \alpha_3\mathbb{E}_{\mathcal{N}(\mathcal{I})}\left[||\nabla\mathcal{J}_r||\right]\right)
\end{aligned}$$

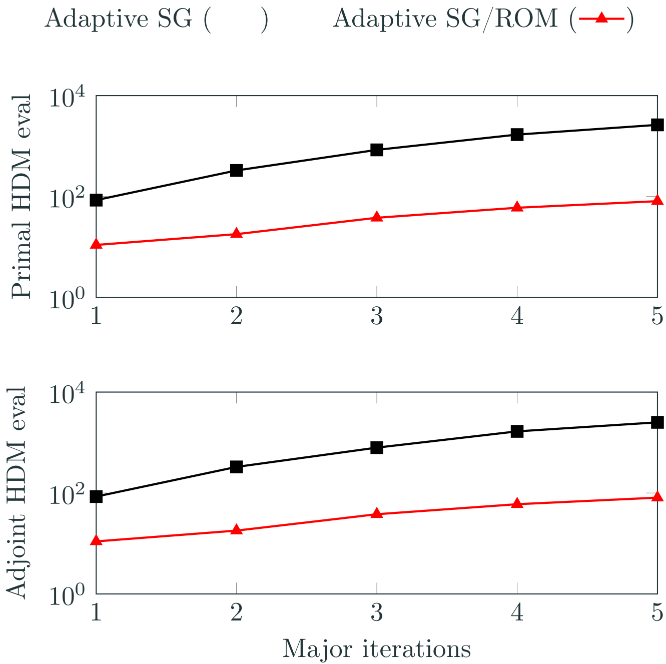
Adaptivity: Dimension-adaptive greedy method



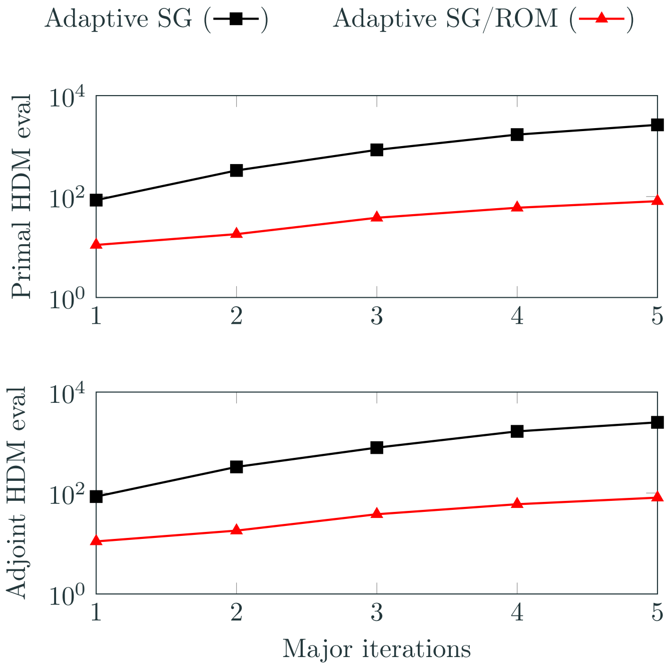


Index set ($\mathcal{I}$) – ● Neighbors ($\mathcal{N}(\mathcal{I})$) – ●

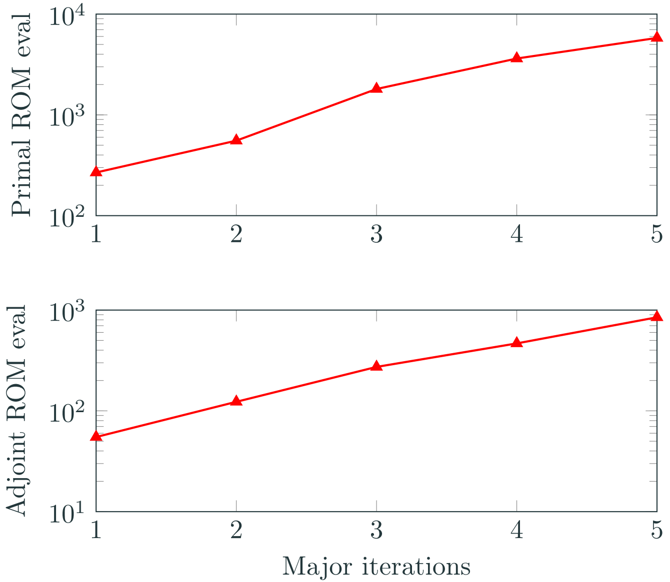
Significant reduction in number of queries to HDM in comparison to state-of-the-art [Kouri et al., 2014]



Significant reduction in number of queries to HDM in comparison to state-of-the-art [Kouri et al., 2014]



At a price ... a large number of ROM evaluations



Trust region method: ROM/SG approximation model

Approximation models built on two sources of inexactness

$$m_k(\boldsymbol{\mu}) = \mathbb{E}_{\mathcal{I}_k}\left[\mathcal{J}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot\,),\,\boldsymbol{\mu},\,\cdot\,)\right]$$
$$\psi_k(\boldsymbol{\mu}) = \mathbb{E}_{\mathcal{I}'_k}\left[\mathcal{J}(\boldsymbol{\Phi}'_k \boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot\,),\,\boldsymbol{\mu},\,\cdot\,)\right]$$

Error indicators that account for both sources of error

$$\vartheta_k(\boldsymbol{\mu}) = ||\boldsymbol{\mu} - \boldsymbol{\mu}_k||$$
$$\varphi_k(\boldsymbol{\mu}) = \alpha_1 \mathcal{E}_1(\boldsymbol{\mu};\,\mathcal{I}_k,\,\boldsymbol{\Phi}_k) + \alpha_2 \mathcal{E}_2(\boldsymbol{\mu};\,\mathcal{I}_k,\,\boldsymbol{\Phi}_k) + \alpha_3 \mathcal{E}_4(\boldsymbol{\mu};\,\mathcal{I}_k,\,\boldsymbol{\Phi}_k)$$
$$\theta_k(\boldsymbol{\mu}) = \beta_1(\mathcal{E}_1(\boldsymbol{\mu};\,\mathcal{I}'_k,\,\boldsymbol{\Phi}'_k) + \mathcal{E}_1(\boldsymbol{\mu}_k;\,\mathcal{I}'_k,\,\boldsymbol{\Phi}'_k)) + \beta_2(\mathcal{E}_3(\boldsymbol{\mu};\,\mathcal{I}'_k,\,\boldsymbol{\Phi}'_k) + \mathcal{E}_3(\boldsymbol{\mu}_k;\,\mathcal{I}'_k,\,\boldsymbol{\Phi}'_k))$$

Reduced-order model errors

$$\mathcal{E}_1(\boldsymbol{\mu};\,\mathcal{I},\,\boldsymbol{\Phi}) = \mathbb{E}_{\mathcal{I}\cup\mathcal{N}(\mathcal{I})}\left[||\boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot),\,\boldsymbol{\mu},\,\cdot\,)||\right]$$
$$\mathcal{E}_2(\boldsymbol{\mu};\,\mathcal{I},\,\boldsymbol{\Phi}) = \mathbb{E}_{\mathcal{I}\cup\mathcal{N}(\mathcal{I})}\left[\left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot),\,\boldsymbol{\Psi}\boldsymbol{\lambda}_r(\boldsymbol{\mu},\,\cdot\,),\,\boldsymbol{\mu},\,\cdot\,)\right|\right|\right]$$

Sparse grid truncation errors

$$\mathcal{E}_3(\boldsymbol{\mu};\,\mathcal{I},\,\boldsymbol{\Phi}) = \mathbb{E}_{\mathcal{N}(\mathcal{I})}\left[|\mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot\,),\,\boldsymbol{\mu},\,\cdot\,)|\right]$$
$$\mathcal{E}_4(\boldsymbol{\mu};\,\mathcal{I},\,\boldsymbol{\Phi}) = \mathbb{E}_{\mathcal{N}(\mathcal{I})}\left[||\nabla\mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot\,),\,\boldsymbol{\mu},\,\cdot\,)||\right]$$

Final requirement for convergence: Adaptivity

With the approximation model, $m_k(\boldsymbol{\mu})$, and gradient error indicator, $\varphi_k(\boldsymbol{\mu})$

$$m_k(\boldsymbol{\mu}) = \mathbb{E}_{\mathcal{I}_k}\left[\mathcal{J}(\boldsymbol{\Phi}_k \boldsymbol{u}_r(\boldsymbol{\mu}, \cdot), \boldsymbol{\mu}, \cdot)\right]$$
$$\varphi_k(\boldsymbol{\mu}) = \alpha_1 \mathcal{E}_1(\boldsymbol{\mu}; \mathcal{I}_k, \boldsymbol{\Phi}_k) + \alpha_2 \mathcal{E}_2(\boldsymbol{\mu}; \mathcal{I}_k, \boldsymbol{\Phi}_k) + \alpha_3 \mathcal{E}_4(\boldsymbol{\mu}; \mathcal{I}_k, \boldsymbol{\Phi}_k)$$

the sparse grid $\mathcal{I}_k$ and reduced-order basis $\boldsymbol{\Phi}_k$ must be constructed such that the gradient condition holds

$$\varphi_k(\boldsymbol{\mu}_k) \leq \kappa_\varphi \min\{||\nabla m_k(\boldsymbol{\mu}_k)||, \Delta_k\}$$

Define dimension-adaptive greedy method to target each source of error such that the stronger conditions hold

$$\mathcal{E}_1(\boldsymbol{\mu}_k; \mathcal{I}, \boldsymbol{\Phi}) \leq \frac{\kappa_\varphi}{3\alpha_1} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||, \Delta_k\}$$
$$\mathcal{E}_2(\boldsymbol{\mu}_k; \mathcal{I}, \boldsymbol{\Phi}) \leq \frac{\kappa_\varphi}{3\alpha_2} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||, \Delta_k\}$$
$$\mathcal{E}_4(\boldsymbol{\mu}_k; \mathcal{I}, \boldsymbol{\Phi}) \leq \frac{\kappa_\varphi}{3\alpha_3} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||, \Delta_k\}$$

Adaptivity: Dimension-adaptive greedy method

while $\mathcal{E}_4(\boldsymbol{\Phi},\, \mathcal{I},\, \boldsymbol{\mu}_k) > \frac{\kappa_\varphi}{3\alpha_3} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||\,,\, \Delta_k\}$ **do**

Refine index set: Dimension-adaptive sparse grids

$$\mathcal{I}_k \leftarrow \mathcal{I}_k \cup \{\mathbf{j}^*\} \qquad \text{where} \qquad \mathbf{j}^* = \underset{\mathbf{j}\in\mathcal{N}(\mathcal{I}_k)}{\arg\max}\; \mathbb{E}_{\mathbf{j}}\left[||\nabla \mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot\,),\, \boldsymbol{\mu},\,\cdot\,)||\right]$$

Adaptivity: Dimension-adaptive greedy method

while $\mathcal{E}_4(\boldsymbol{\Phi}, \mathcal{I}, \boldsymbol{\mu}_k) > \frac{\kappa_\varphi}{3\alpha_3} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||\,,\, \Delta_k\}$ **do**

<u>**Refine index set**</u>: Dimension-adaptive sparse grids

$$\mathcal{I}_k \leftarrow \mathcal{I}_k \cup \{\mathbf{j}^*\} \qquad \text{where} \qquad \mathbf{j}^* = \underset{\mathbf{j}\in\mathcal{N}(\mathcal{I}_k)}{\arg\max}\ \mathbb{E}_{\mathbf{j}}\left[||\nabla \mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot\,),\,\boldsymbol{\mu},\,\cdot\,)||\right]$$

<u>**Refine reduced-order basis**</u>: Greedy sampling

while $\mathcal{E}_1(\boldsymbol{\Phi}, \mathcal{I}, \boldsymbol{\mu}_k) > \frac{\kappa_\varphi}{3\alpha_1} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||\,,\, \Delta_k\}$ **do**

$$\boldsymbol{\Phi}_k \leftarrow \begin{bmatrix} \boldsymbol{\Phi}_k & \boldsymbol{u}(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}^*) & \boldsymbol{\lambda}(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}^*) \end{bmatrix}$$

$$\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}\in\boldsymbol{\Xi}_{\mathbf{j}^*}}{\arg\max}\ \rho(\boldsymbol{\xi})\,||\boldsymbol{r}(\boldsymbol{\Phi}_k\boldsymbol{u}_r(\boldsymbol{\mu}_k,\,\boldsymbol{\xi}),\,\boldsymbol{\mu}_k,\,\boldsymbol{\xi})||$$

end while

while $\mathcal{E}_4(\boldsymbol{\Phi}, \mathcal{I}, \boldsymbol{\mu}_k) > \frac{\kappa_\varphi}{3\alpha_3} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||\,,\, \Delta_k\}$ **do**

Refine index set: Dimension-adaptive sparse grids

$$\mathcal{I}_k \leftarrow \mathcal{I}_k \cup \{\mathbf{j}^*\} \qquad \text{where} \qquad \mathbf{j}^* = \underset{\mathbf{j}\in\mathcal{N}(\mathcal{I}_k)}{\arg\max}\; \mathbb{E}_{\mathbf{j}}\left[||\nabla \mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r(\boldsymbol{\mu},\,\cdot\,),\, \boldsymbol{\mu},\,\cdot\,)||\right]$$

Refine reduced-order basis: Greedy sampling

while $\mathcal{E}_1(\boldsymbol{\Phi}, \mathcal{I}, \boldsymbol{\mu}_k) > \frac{\kappa_\varphi}{3\alpha_1} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||\,,\, \Delta_k\}$ **do**

$$\boldsymbol{\Phi}_k \leftarrow \begin{bmatrix} \boldsymbol{\Phi}_k & \boldsymbol{u}(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}^*) & \boldsymbol{\lambda}(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}^*) \end{bmatrix}$$

$$\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}\in\boldsymbol{\Xi}_{\mathbf{j}^*}}{\arg\max}\; \rho(\boldsymbol{\xi})\, ||\boldsymbol{r}(\boldsymbol{\Phi}_k\boldsymbol{u}_r(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}),\, \boldsymbol{\mu}_k,\, \boldsymbol{\xi})||$$

end while

while $\mathcal{E}_2(\boldsymbol{\Phi}, \mathcal{I}, \boldsymbol{\mu}_k) > \frac{\kappa_\varphi}{3\alpha_2} \min\{||\nabla m_k(\boldsymbol{\mu}_k)||\,,\, \Delta_k\}$ **do**

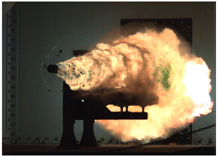
$$\boldsymbol{\Phi}_k \leftarrow \begin{bmatrix} \boldsymbol{\Phi}_k & \boldsymbol{u}(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}^*) & \boldsymbol{\lambda}(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}^*) \end{bmatrix}$$

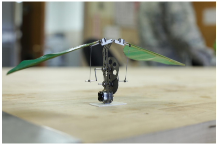
$$\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}\in\boldsymbol{\Xi}_{\mathbf{j}^*}}{\arg\max}\; \rho(\boldsymbol{\xi}) \left|\left|\boldsymbol{r}^{\boldsymbol{\lambda}}(\boldsymbol{\Phi}_k\boldsymbol{u}_r(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}),\, \boldsymbol{\Psi}_k\boldsymbol{\lambda}_r(\boldsymbol{\mu}_k,\, \boldsymbol{\xi}),\, \boldsymbol{\mu}_k,\, \boldsymbol{\xi})\right|\right|$$

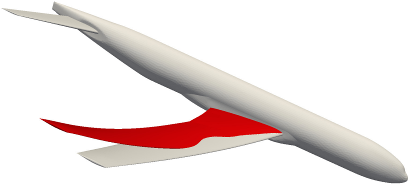
end while

Leveraging inexactness to accelerate PDE-constrained optimization

- Framework introduced for accelerating **deterministic** and **stochastic** PDE-constrained optimization problems
    - Adaptive *model reduction*
    - *Partially converged* primal and adjoint solutions
    - Dimension-adaptive *sparse grids*
- Inexactness **managed** with flexible **trust region** method
- Applied to variety of problems in computational mechanics and outperforms state-of-the-art methods
    - **1.6**$\times$ speedup on (deterministic) shape design of aircraft
    - **100**$\times$ speedup on (stochastic) optimal control of 1D flow







Extension to time-dependent problems

- **Applications**: inverse problems, optimal flapping flight and swimming[4] and design of helicopter blades, wind turbines, and turbomachinery
- Monolithic **space-time** formulation of reduced-order model
  - Increased speed due to natural **parallelism** in *space and time*
  - Treat as **steady state** problem in $n_{sd} + 1$ dimensions
- **Error indicators and adaptivity** algorithms in space-time setting to solve with multifidelity trust region method

Un-optimized flapping motion (left), optimal control (center), and optimal control and time-morphed geometry (right)

4. Insight into bio-locomotion, design of micro-aerial vehicles

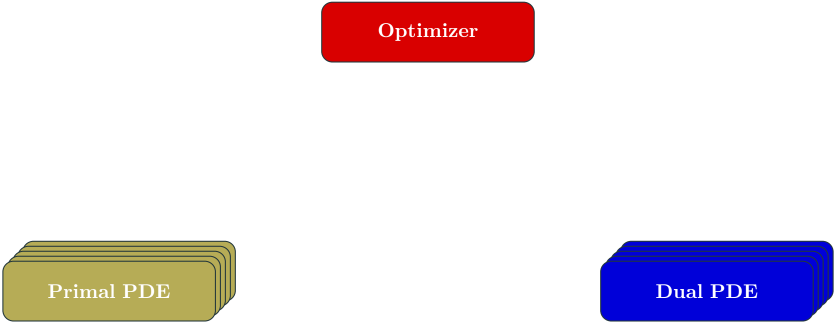
Stochastic PDE-constrained optimization formulation

$$\begin{aligned} \underset{\boldsymbol{\mu} \in \mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad & \mathbb{E}[\mathcal{J}(\boldsymbol{u}, \boldsymbol{\mu}, \cdot)] \\ \text{subject to} \quad & \boldsymbol{r}(\boldsymbol{u}; \boldsymbol{\mu}, \boldsymbol{\xi}) = 0 \quad \forall \boldsymbol{\xi} \in \boldsymbol{\Xi} \end{aligned}$$

- $\boldsymbol{r} : \mathbb{R}^{n_{\boldsymbol{u}}} \times \mathbb{R}^{n_{\boldsymbol{\mu}}} \times \mathbb{R}^{n_{\boldsymbol{\xi}}} \rightarrow \mathbb{R}^{n_{\boldsymbol{u}}}$ — discretized stochastic PDE
- $\mathcal{J} : \mathbb{R}^{n_{\boldsymbol{u}}} \times \mathbb{R}^{n_{\boldsymbol{\mu}}} \times \mathbb{R}^{n_{\boldsymbol{\xi}}} \rightarrow \mathbb{R}$ — quantity of interest
- $\boldsymbol{u} \in \mathbb{R}^{n_{\boldsymbol{u}}}$ — PDE state vector
- $\boldsymbol{\mu} \in \mathbb{R}^{n_{\boldsymbol{\mu}}}$ — (deterministic) optimization parameters
- $\boldsymbol{\xi} \in \mathbb{R}^{n_{\boldsymbol{\xi}}}$ — stochastic parameters
- $\mathbb{E}[\mathcal{F}] \equiv \int_{\boldsymbol{\Xi}} \mathcal{F}(\boldsymbol{\xi}) \rho(\boldsymbol{\xi}) \, d\boldsymbol{\xi}$

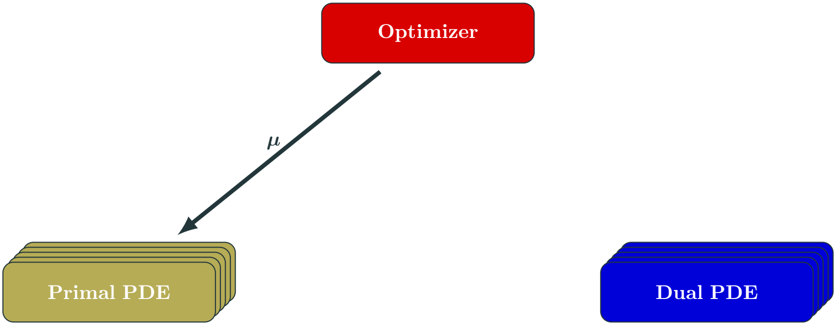
Each function evaluation requires integration over stochastic space – ***expensive***

Ensemble *of primal/dual PDE solves increases cost by* ***orders of magnitude***



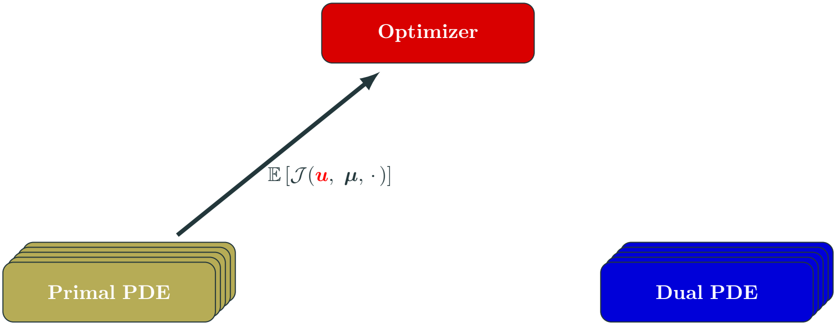
Nested approach to stochastic PDE-constrained optimization

Ensemble *of primal/dual PDE solves increases cost by* ***orders of magnitude***



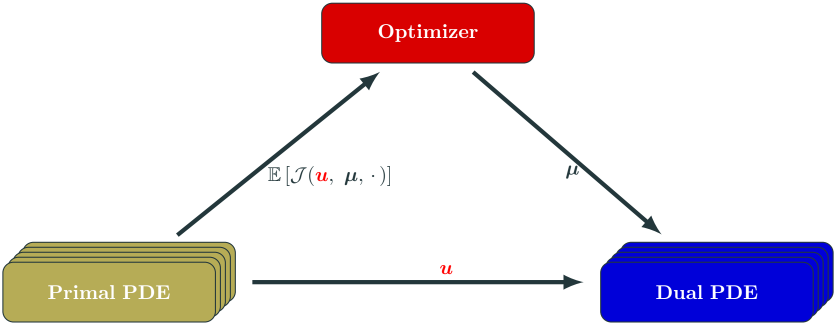
Nested approach to stochastic PDE-constrained optimization

Ensemble *of primal/dual PDE solves increases cost by* ***orders of magnitude***



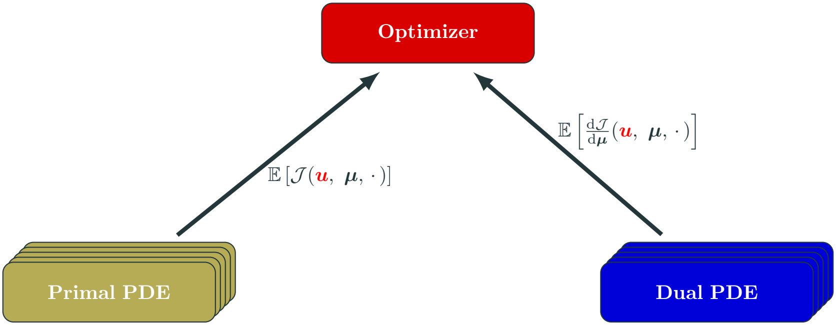
Nested approach to stochastic PDE-constrained optimization

Ensemble *of primal/dual PDE solves increases cost by* ***orders of magnitude***



Nested approach to stochastic PDE-constrained optimization

Ensemble *of primal/dual PDE solves increases cost by* ***orders of magnitude***



First source of inexactness: anisotropic sparse grids

Stochastic collocation using anisotropic sparse grid nodes to approximate integral with summation

$$\begin{array}{ll} \underset{\boldsymbol{u}\in\mathbb{R}^{n_{\boldsymbol{u}}},\ \boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} & \mathbb{E}[\mathcal{J}(\boldsymbol{u},\,\boldsymbol{\mu},\,\cdot\,)] \\ \text{subject to} & \boldsymbol{r}(\boldsymbol{u},\,\boldsymbol{\mu},\,\boldsymbol{\xi}) = 0 \quad \forall \boldsymbol{\xi} \in \boldsymbol{\Xi} \end{array}$$

$$\Downarrow$$

$$\begin{array}{ll} \underset{\boldsymbol{u}\in\mathbb{R}^{n_{\boldsymbol{u}}},\ \boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} & \mathbb{E}_{\mathcal{I}}[\mathcal{J}(\boldsymbol{u},\,\boldsymbol{\mu},\,\cdot\,)] \\ \text{subject to} & \boldsymbol{r}(\boldsymbol{u},\,\boldsymbol{\mu},\,\boldsymbol{\xi}) = 0 \quad \forall \boldsymbol{\xi} \in \boldsymbol{\Xi}_{\mathcal{I}} \end{array}$$

[Kouri et al., 2013, Kouri et al., 2014]

Second source of inexactness: reduced-order models

Stochastic collocation of the reduced-order model over anisotropic sparse grid nodes used to approximate integral with cheap summation

$$\begin{aligned} \underset{\boldsymbol{u}\in\mathbb{R}^{n_{\boldsymbol{u}}},\ \boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad & \mathbb{E}[\mathcal{J}(\boldsymbol{u},\ \boldsymbol{\mu},\ \cdot\ )] \\ \text{subject to} \quad & \boldsymbol{r}(\boldsymbol{u},\ \boldsymbol{\mu},\ \boldsymbol{\xi}) = 0 \quad \forall \boldsymbol{\xi} \in \boldsymbol{\Xi} \end{aligned}$$

$$\Downarrow$$

$$\begin{aligned} \underset{\boldsymbol{u}\in\mathbb{R}^{n_{\boldsymbol{u}}},\ \boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad & \mathbb{E}_{\mathcal{I}}[\mathcal{J}(\boldsymbol{u},\ \boldsymbol{\mu},\ \cdot\ )] \\ \text{subject to} \quad & \boldsymbol{r}(\boldsymbol{u},\ \boldsymbol{\mu},\ \boldsymbol{\xi}) = 0 \quad \forall \boldsymbol{\xi} \in \boldsymbol{\Xi}_{\mathcal{I}} \end{aligned}$$

$$\Downarrow$$

$$\begin{aligned} \underset{\boldsymbol{u}_r\in\mathbb{R}^{k_{\boldsymbol{u}}},\ \boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad & \mathbb{E}_{\mathcal{I}}[\mathcal{J}(\boldsymbol{\Phi}\boldsymbol{u}_r,\ \boldsymbol{\mu},\ \cdot\ )] \\ \text{subject to} \quad & \boldsymbol{\Psi}^T\boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{u}_r,\ \boldsymbol{\mu},\ \boldsymbol{\xi}) = 0 \quad \forall \boldsymbol{\xi} \in \boldsymbol{\Xi}_{\mathcal{I}} \end{aligned}$$

Proposed approach: managed inexactness

Replace expensive PDE with inexpensive approximation model

- **Reduced-order models** used for *inexact PDE evaluations*
- **Anisotropic sparse grids** used for *inexact integration* of risk measures

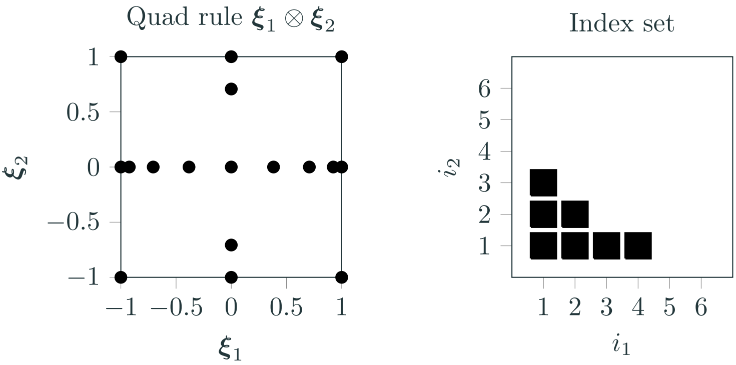
$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad m_k(\boldsymbol{\mu})$$

Manage inexactness with trust region method

- Embedded in globally convergent **trust region** method
- **Error indicators** to account for *all* sources of inexactness
- **Refinement** of approximation model using *greedy algorithms*

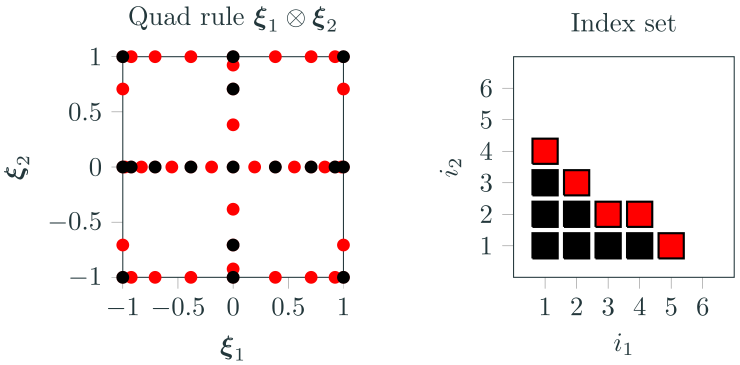
$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad F(\boldsymbol{\mu}) \qquad \longrightarrow \qquad \begin{array}{ll} \underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} & m_k(\boldsymbol{\mu}) \\ \text{subject to} & ||\boldsymbol{\mu}-\boldsymbol{\mu}_k|| \leq \Delta_k \end{array}$$

Source of inexactness: anisotropic sparse grids



Index set ($\mathcal{I}$) – ●   Neighbors ($\mathcal{N}(\mathcal{I})$) – ●

Source of inexactness: anisotropic sparse grids



Index set $(\mathcal{I})$ – ●   Neighbors $(\mathcal{N}(\mathcal{I}))$ – ●

Optimal control of steady Burgers' equation

- **Optimization problem:**

$$\underset{\boldsymbol{\mu}\in\mathbb{R}^{n_{\boldsymbol{\mu}}}}{\text{minimize}} \quad \int_{\boldsymbol{\Xi}} \rho(\boldsymbol{\xi}) \left[\int_0^1 \frac{1}{2}(u(\boldsymbol{\mu}, \boldsymbol{\xi}, x) - \bar{u}(x))^2 \, dx + \frac{\alpha}{2} \int_0^1 z(\boldsymbol{\mu}, x)^2 \, dx \right] d\boldsymbol{\xi}$$

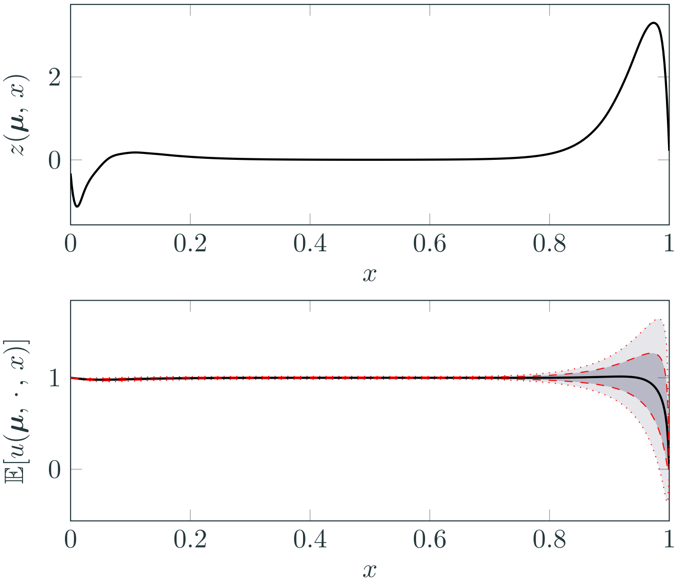
where $u(\boldsymbol{\mu}, \boldsymbol{\xi}, x)$ solves

$$-\nu(\boldsymbol{\xi})\partial_{xx}u(\boldsymbol{\mu}, \boldsymbol{\xi}, x) + u(\boldsymbol{\mu}, \boldsymbol{\xi}, x)\partial_x u(\boldsymbol{\mu}, \boldsymbol{\xi}, x) = z(\boldsymbol{\mu}, x) \quad x \in (0, 1), \quad \boldsymbol{\xi} \in \boldsymbol{\Xi}$$
$$u(\boldsymbol{\mu}, \boldsymbol{\xi}, 0) = d_0(\boldsymbol{\xi}) \qquad u(\boldsymbol{\mu}, \boldsymbol{\xi}, 1) = d_1(\boldsymbol{\xi})$$

- **Target state:** $\bar{u}(x) \equiv 1$
- **Stochastic Space:** $\boldsymbol{\Xi} = [-1, 1]^3$, $\rho(\boldsymbol{\xi})d\boldsymbol{\xi} = 2^{-3}d\boldsymbol{\xi}$

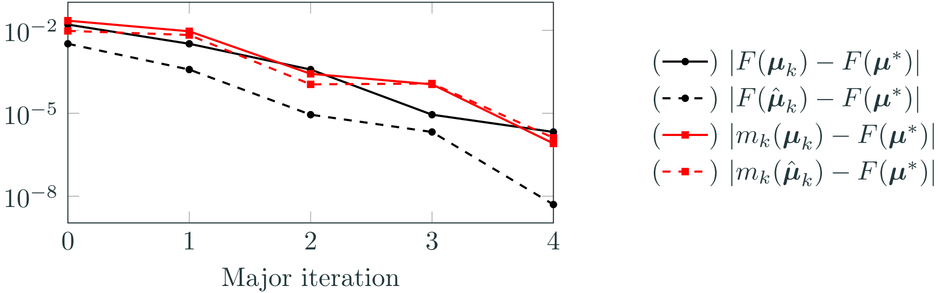
$$\nu(\boldsymbol{\xi}) = 10^{\xi_1 - 2} \qquad d_0(\boldsymbol{\xi}) = 1 + \frac{\xi_2}{1000} \qquad d_1(\boldsymbol{\xi}) = \frac{\xi_3}{1000}$$

- **Parametrization:** $z(\boldsymbol{\mu}, x)$ – cubic splines with 51 knots, $n_{\boldsymbol{\mu}} = 53$



Optimal control and corresponding mean state (——) ± one (- - -) and two (·······) standard deviations

Global convergence without pointwise agreement

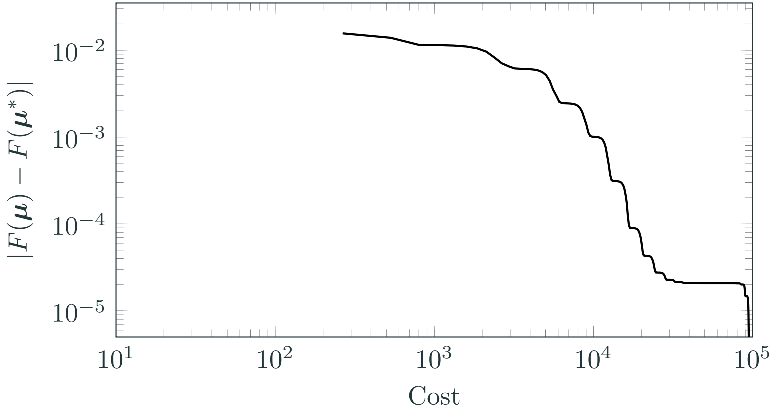


$F(\boldsymbol{\mu}_k)$	$m_k(\boldsymbol{\mu}_k)$	$F(\hat{\boldsymbol{\mu}}_k)$	$m_k(\hat{\boldsymbol{\mu}}_k)$	$\|\|\nabla F(\boldsymbol{\mu}_k)\|\|$	ρ_k	Success?
6.6506e-02	7.2694e-02	5.3655e-02	5.9922e-02	2.2959e-02	1.0257e+00	1.0000e+00
5.3655e-02	5.9593e-02	5.0783e-02	5.7152e-02	2.3424e-03	9.7512e-01	1.0000e+00
5.0783e-02	5.0670e-02	5.0412e-02	5.0292e-02	1.9724e-03	9.8351e-01	1.0000e+00
5.0412e-02	5.0292e-02	5.0405e-02	5.0284e-02	9.2654e-05	8.7479e-01	1.0000e+00
5.0405e-02	5.0404e-02	5.0403e-02	5.0401e-02	8.3139e-05	9.9946e-01	1.0000e+00
5.0403e-02	5.0401e-02	-	-	2.2846e-06	-	-

Convergence history of trust region method built on two-level approximation

Significant reduction in cost, even if (largest) ROM only $10\times$ faster than HDM

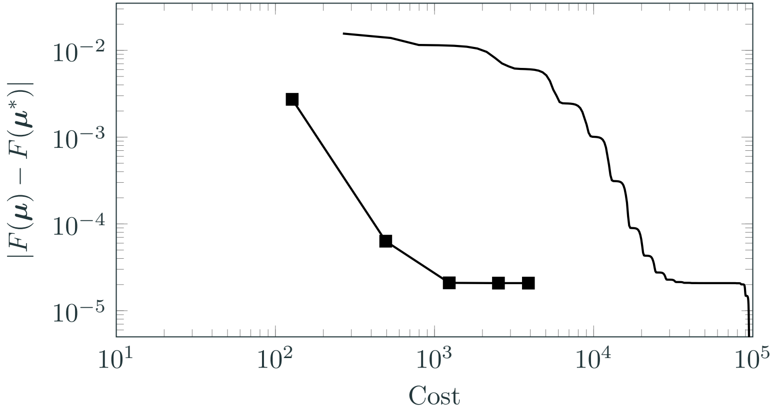
$$\text{Cost} = \text{nHdmPrim} + 0.5 \times \text{nHdmAdj} + \tau^{-1} \times (\text{nRomPrim} + 0.5 \times \text{nRomAdj})$$



5-level isotropic SG (——), dimension-adaptive SG [Kouri et al., 2014] ( ), and proposed ROM/SG for $\tau = 1$ ( ), $\tau = 10$ ( ), $\tau = 100$ ( ), $\tau = \infty$ ( )

Significant reduction in cost, even if (largest) ROM only $10\times$ faster than HDM

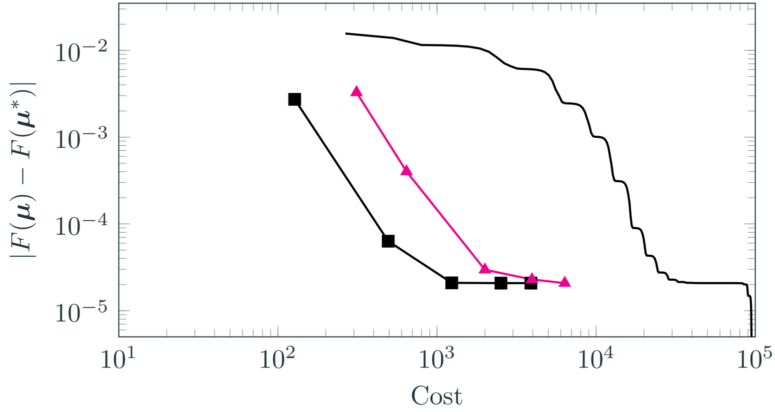
$$\text{Cost} = \text{nHdmPrim} + 0.5 \times \text{nHdmAdj} + \tau^{-1} \times (\text{nRomPrim} + 0.5 \times \text{nRomAdj})$$



5-level isotropic SG (——), dimension-adaptive SG [Kouri et al., 2014] (—■—), and proposed ROM/SG for $\tau = 1$ ( ), $\tau = 10$ ( ), $\tau = 100$ ( ), $\tau = \infty$ ( )

Significant reduction in cost, even if (largest) ROM only $10\times$ faster than HDM

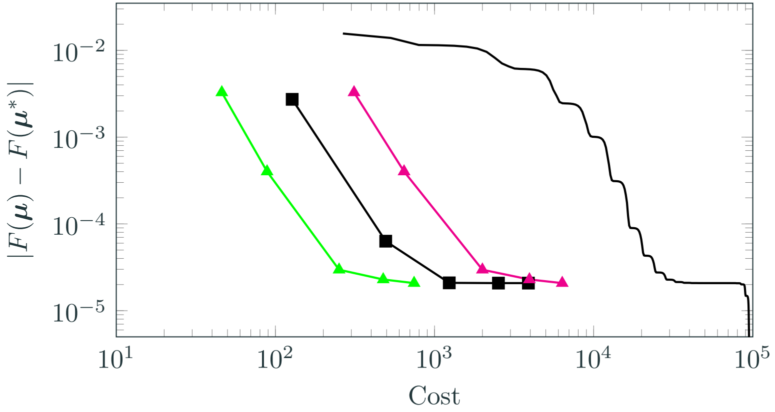
$$\text{Cost} = \text{nHdmPrim} + 0.5 \times \text{nHdmAdj} + \tau^{-1} \times (\text{nRomPrim} + 0.5 \times \text{nRomAdj})$$



5-level isotropic SG (—), dimension-adaptive SG [Kouri et al., 2014] (■), and proposed ROM/SG for $\tau = 1$ (▲), $\tau = 10$ ( ), $\tau = 100$ ( ), $\tau = \infty$ ( )

Significant reduction in cost, even if (largest) ROM only $10\times$ faster than HDM

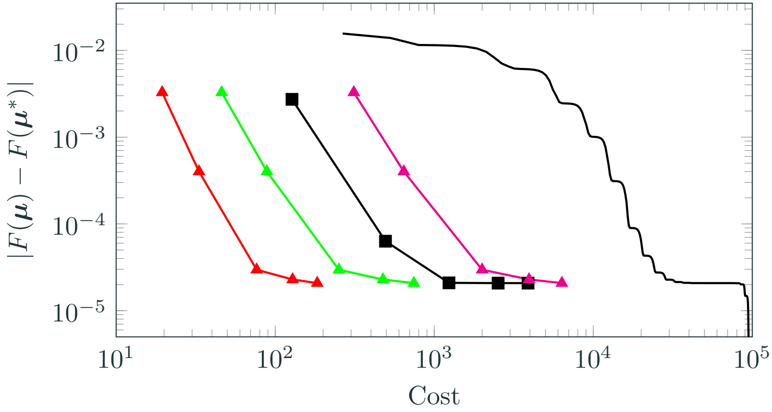
$$\text{Cost} = \text{nHdmPrim} + 0.5 \times \text{nHdmAdj} + \tau^{-1} \times (\text{nRomPrim} + 0.5 \times \text{nRomAdj})$$



5-level isotropic SG (——), dimension-adaptive SG [Kouri et al., 2014] (—■—), and proposed ROM/SG for $\tau = 1$ (—▲—), $\tau = 10$ (—▲—), $\tau = 100$ ( ), $\tau = \infty$ ( )

Significant reduction in cost, even if (largest) ROM only $10\times$ faster than HDM

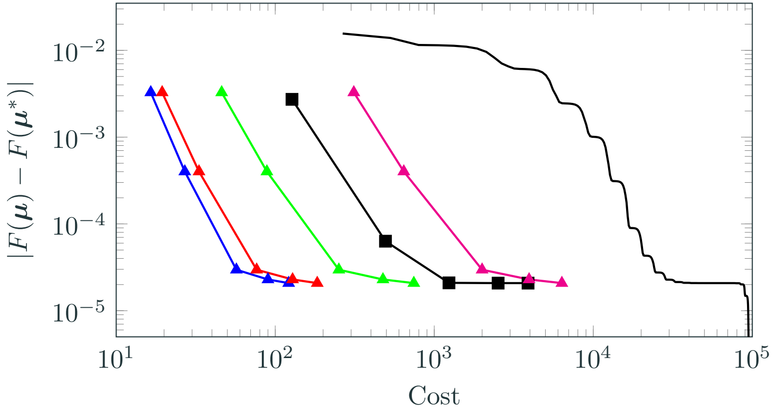
$$\text{Cost} = \text{nHdmPrim} + 0.5 \times \text{nHdmAdj} + \tau^{-1} \times (\text{nRomPrim} + 0.5 \times \text{nRomAdj})$$



5-level isotropic SG (—), dimension-adaptive SG [Kouri et al., 2014] (—■—), and proposed ROM/SG for $\tau = 1$ (—▲—), $\tau = 10$ (—▲—), $\tau = 100$ (—▲—), $\tau = \infty$ (—▲—)

Significant reduction in cost, even if (largest) ROM only $10\times$ faster than HDM

$$\text{Cost} = \text{nHdmPrim} + 0.5 \times \text{nHdmAdj} + \tau^{-1} \times (\text{nRomPrim} + 0.5 \times \text{nRomAdj})$$



5-level isotropic SG (—), dimension-adaptive SG [Kouri et al., 2014] (■), and proposed ROM/SG for $\tau = 1$ (▲), $\tau = 10$ (▲), $\tau = 100$ (▲), $\tau = \infty$ (▲)