Model Reduction Methods

Martin A. Grepl[1], Gianluigi Rozza[2]

[1]IGPM, RWTH Aachen University, Germany
[2] MATHICSE - CMCS, Ecole Polytechnique Fédérale de Lausanne, Switzerland

Summer School "Optimal Control of PDEs"
Cortona (Italy), July 12-17, 2010

The rbMIT software package

- **rbMIT**: - Reduced Basis Software Library in Matlab(C) Environment
- developed at MIT by DBP Huynh, CN Nguyen, G Rozza and AT Patera
- based on extensive use of Matlab ToolBoxes like Symbolic, PDEs, Optimization
- The user must describe the problem. The input can be separated into three parts:

The User Input

- **geometry**: $\Omega_o(\mu)$ is defined by providing points coordinates, straight/curvy edges describing all regions $\Omega_o^k(\mu)$
- **material properties**: coefficients are provided for differential operator in each region $\Omega_o^k(\mu)$ and for boundary conditions.
- **parameter control and settings**: parameter domain $\mathcal{D}$, reference parameters and other RB information (e.g. $N_{\max}$)

The rbMIT software package

rbMIT Users' Interface

```
% -----         rbMIT  Software Copyright MIT 2006-09        -----
% --------- DBP Huynh, NC Nguyen, AT Patera, G Rozza ---------

probname = 'Tfin';
```

Geometry

```
points = '[0,0; 1/2,0; 1/2,3/5; 3/20,3/5; 0,3/5; 3/20,3/5+mu2/2;
          3/20,3/5+mu2; 0,3/5+mu2; 0,3/5+mu2/2]';

edge = [1,2;2,3;3,4;4,5;4,6;6,7;7,8;8,9;9,5;5,1];

geometry{1} = [1,2,3,4,10];
geometry{2} = [4,5,6,7,8,9];
gflag = [1,1];
```

Parameters

```
muref  = [.1,4,1];
mu_min = [.01,2,1];
mu_max = [0.5,8,10];
mu_bar = [.1,4,1];
```

PDE and BC

```
kappa{1} = '[mu3, 0, 0; 0, mu3, 0; 0, 0, 0]' ;
kappa{2} = '[1, 0, 0; 0, 1, 0; 0, 0, 0]' ;

dirichlet = '[1,0; 2,1; 4,0]' ;
nload     = '[3,0,0,1; 5, mu1,0,0;6,mu1,0,0]' ;
```

Output

```
outputname = 'basetemp' ;
oload='[1,2]';
```

Problem Formulation, Offline and Online Steps

The Problem Formulation Step

- Domain Decomposition/geometric transformations: coefficients $\Theta_q(\mu)$ are generated for each sub-domain (coupled with material μ-properties)
- FE mesh is generated, discrete FE stiffness matrices/vectors are assembled for each sub-domain to form the μ-independent components

The RB Offline Step

- By a greedy algorithm the RB parameter sample set is obtained
- FE/RB matrices are saved in order to be used by the Online Step

The RB Online Step

- Given $\mu \in \mathcal{D}$, the RB Online Evaluator returns output and error bound
  `Online_RB (probname,` μ`, outputname, ...)`: $\mu \to s_N^{\mathcal{N}}(\mu),\ \Delta_N^s(\mu)$
- The RB Visualizer renders the field variable(s) and provides error bounds
  `Vis_RB (probname,` μ`)`: $\mu \to \Omega,\ u_N^{\mathcal{N}}(x;\mu)$ for all x in $\Omega_o(\mu)$

Problem Formulation, Offline and Online Steps

The Problem Formulation Step

- Domain Decomposition/geometric transformations: coefficients $\Theta_q(\mu)$ are generated for each sub-domain (coupled with material μ-properties)
- FE mesh is generated, discrete FE stiffness matrices/vectors are assembled for each sub-domain to form the μ-independent components

The RB Offline Step

- By a greedy algorithm the RB parameter sample set is obtained
- FE/RB matrices are saved in order to be used by the Online Step

The RB Online Step

- Given $\mu \in \mathcal{D}$, the RB Online Evaluator returns output and error bound
  `Online_RB (probname,` μ`, outputname, ...)`: $\mu \to s_N^{\mathcal{N}}(\mu),\ \Delta_N^s(\mu)$
- The RB Visualizer renders the field variable(s) and provides error bounds
  `Vis_RB (probname,` μ`)`: $\mu \to \Omega,\ u_N^{\mathcal{N}}(x;\mu)$ for all x in $\Omega_o(\mu)$

Problem Formulation, Offline and Online Steps

The Problem Formulation Step

- Domain Decomposition/geometric transformations: coefficients $\Theta_q(\mu)$ are generated for each sub-domain (coupled with material μ-properties)
- FE mesh is generated, discrete FE stiffness matrices/vectors are assembled for each sub-domain to form the μ-independent components

The RB Offline Step

- By a greedy algorithm the RB parameter sample set is obtained
- FE/RB matrices are saved in order to be used by the Online Step

The RB Online Step

- Given $\mu \in \mathcal{D}$, the RB Online Evaluator returns output and error bound

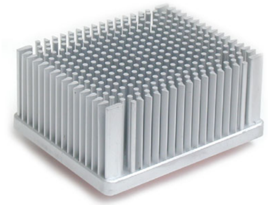
  `Online_RB (probname,` μ`, outputname, ...)`: $\mu \rightarrow s_N^{\mathcal{N}}(\mu),\ \Delta_N^s(\mu)$
- The RB Visualizer renders the field variable(s) and provides error bounds

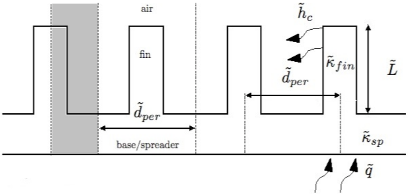
  `Vis_RB (probname,` μ`)`: $\mu \rightarrow \Omega,\ u_N^{\mathcal{N}}(x;\mu)$ for all x in $\Omega_o(\mu)$

rbMIT Example: the Thermal Fin problem

Engineering aspects

- Heat sink designed for cooling of high-density electronic components
- Shaded domain due to assumed periodicity and symmetry (multi-fin sink)
- Flowing air is modelled though a simple convection HT coefficient: to compute temperature at the base of the spreader





Physical and geometrical parametrization

$\mu_1 = \mathrm{Bi} = \tilde{h}_c \tilde{d}_{\mathrm{per}} / \tilde{\kappa}_{\mathrm{fin}}$	Biot number	$\mu_1 \in [0.01, 0.5]$
$\mu_2 = L = \tilde{L} / \tilde{d}_{\mathrm{per}}$	nondimensional fin height	$\mu_2 \in [2, 8]$
$\mu_3 = \kappa = \tilde{\kappa}_{\mathrm{sp}} / \tilde{\kappa}_{\mathrm{fin}}$	spreader/fin conductivity ratio	$\mu_3 \in [1, 10]$

rbMIT Example: the Thermal Fin problem

- Modeling: temperature $u_o(\mu)$ over $\Omega_o(\mu)$ satisfies the steady heat equation
- Output: average temperature over the base of the spreader (component to be cooled, being the hottest location in the system)

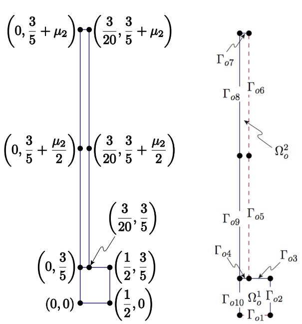
$$-\frac{\partial}{\partial x_{o\,i}}\Bigg(\underbrace{\begin{bmatrix}\mu_3 & 0\\ 0 & \mu_3\end{bmatrix}}_{\kappa^1_{o\,ij}}\frac{\partial}{\partial x_{o\,j}}u_o(\mu)\Bigg)=0 \quad \text{in } \Omega_o^1$$

$$-\frac{\partial}{\partial x_{o\,i}}\Bigg(\underbrace{\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}}_{\kappa^2_{o\,ij}}\frac{\partial}{\partial x_{o\,j}}u_o(\mu)\Bigg)=0 \quad \text{in } \Omega_o^2(\mu_2)$$

$$\mathbf{n}_{o\,i}\kappa^1_{o\,ij}\frac{\partial u_o}{\partial x_{o\,j}}(\mu)=1 \quad \text{on} \quad \Gamma_{o\,1}$$

$$\mathbf{n}_{o\,i}\kappa^2_{o\,ij}\frac{\partial u_o}{\partial x_{o\,j}}(\mu)+(\mu_1)u_o=0 \quad \text{on} \quad \Gamma_R=\Gamma_{o\,5}\cup\Gamma_{0\,6}$$

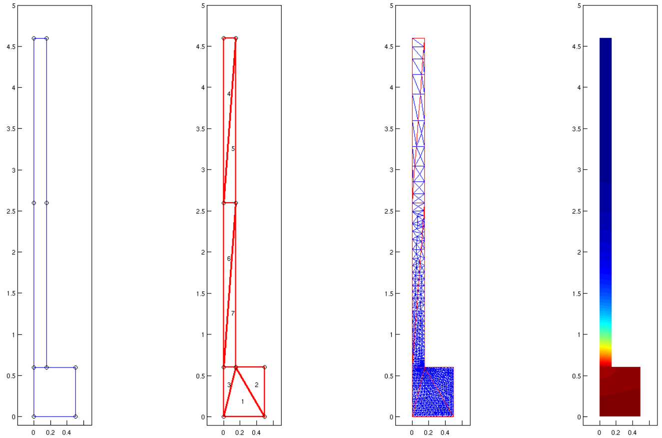
$$\mathbf{n}_{o\,i}\kappa_{o\,ij}\frac{\partial}{\partial x_{o\,j}}u_o(\mu)=0 \quad \text{on} \quad \Gamma\setminus(\Gamma_{o\,1}\cup\Gamma_R)$$



output: $\quad T_{o\,\mathrm{av}}(\mu)=2\int_{\Gamma_{o\,1}}u_o(\mu)$

rbMIT Example: the Thermal Fin problem

- Example of geometry and field visualizations provided by `rbMIT` package



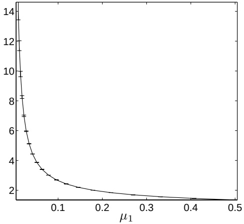
Example of initial geometry, domain decompostion, FE mesh and RB solution visualization for a thermal fin problem

rbMIT Example: the Thermal Fin problem

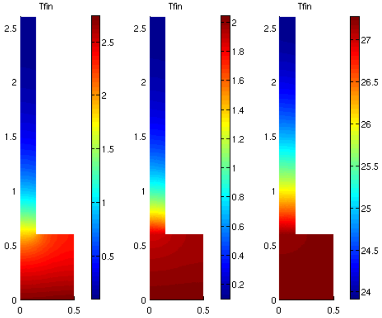
Approximation property	
# of mesh nodes $\mathcal{N}$	$\mathbf{4198}$
# of RB functions N	$\approx \mathbf{10}$

Reduced Basis vs Finite Elements	
RB online evaluation time	$\mathbf{0.13}s\ (N = \mathbf{7})$ $\mathbf{0.15}s\ (N = \mathbf{13})$
FEM sol. $\mu \rightarrow s^{\mathcal{N}}(\mu)$	$\mathbf{1.96}s$

- ⋆ Reduction of 400:1 in linear system dimension
- ⋆ Online evaluation $\approx \mathbf{5} - \mathbf{6}\%$ of the FEM computational cost



RB output/error bars -$[s_N(\mu) - \Delta_N^s(\mu), s_N(\mu) + \Delta_N^s(\mu)]$ as a function of μ_1 for $\mu_2 = 2$, $\mu_3 = 1$ and $N = 6$.



RB temperature field for different choices of parameters: $\mu = (0.5, 2, 1)$, $\mu = (0.5, 2, 5)$, $\mu = (0.01, 2, 10)$.

Geometrical Parametrization: the goal

- The parametrized (original) domain $\Omega_o(\mu)$ is the image of a fixed (reference) domain Ω through a map $T(\cdot;\mu) : \Omega \rightarrow \Omega_o(\mu)$
- In order to recover the affine parameter dependence, the parametric map $T(\cdot;\mu)$ has to be an **affine map**.
- The `rbMIT` software allows to deal with more complex configurations by means of automatically built affine mappings

Geometrical Parametrization: Domain Decomposition

Domain decomposition: definition

Original Domain $\boldsymbol{\Omega}_{\mathrm{o}}(\boldsymbol{\mu})$, $\boldsymbol{u}_{\mathrm{o}}^{\mathrm{e}} \in \boldsymbol{X}_{\mathrm{o}}^{\mathrm{e}}(\boldsymbol{\Omega}_{\mathrm{o}}(\boldsymbol{\mu}))$

$$\overline{\boldsymbol{\Omega}}_{\mathrm{o}}(\boldsymbol{\mu}) = \bigcup_{\boldsymbol{k}=1}^{\boldsymbol{K}_{\mathrm{dom}}} \overline{\boldsymbol{\Omega}}_{\mathrm{o}}^{\boldsymbol{k}}(\boldsymbol{\mu}) \; ;$$

Reference domain $\boldsymbol{\Omega}$, $\boldsymbol{u}^{\mathrm{e}} \in \boldsymbol{X}^{\mathrm{e}}(\boldsymbol{\Omega})$

$$\overline{\boldsymbol{\Omega}} = \bigcup_{\boldsymbol{k}=1}^{\boldsymbol{K}_{\mathrm{dom}}} \overline{\boldsymbol{\Omega}}^{\boldsymbol{k}} \; ,$$

common configuration

where $\boldsymbol{\Omega} = \boldsymbol{\Omega}_{\mathrm{o}}(\boldsymbol{\mu}_{\mathrm{ref}})$ for $\boldsymbol{\mu}_{\mathrm{ref}} \subset \boldsymbol{\mathcal{D}}^{\dagger}$.

For Ω^k, $\Omega_{\mathrm{o}}^k(\mu)$ we choose in $\mathbb{R}^2$ triangles, elliptical triangles and curvy triangles. In $\mathbb{R}^3$ we choose parallelepipeds (and in theory tetrahedra).

† Connectivity requirement: subdomain intersections must be an entire edge, a vertex, or null.

Geometrical Parametrization: Domain Decomposition

Domain decomposition: definition

Original Domain $\boldsymbol{\Omega}_{\mathrm{o}}(\boldsymbol{\mu})$, $\quad\quad u_{\mathrm{o}}^{\mathrm{e}} \in X_{\mathrm{o}}^{\mathrm{e}}(\boldsymbol{\Omega}_{\mathrm{o}}(\boldsymbol{\mu}))$

$$\overline{\boldsymbol{\Omega}}_{\mathrm{o}}(\boldsymbol{\mu}) = \bigcup_{k=1}^{K_{\mathrm{dom}}} \overline{\boldsymbol{\Omega}}_{\mathrm{o}}^{k}(\boldsymbol{\mu}) \ ;$$

Reference domain $\boldsymbol{\Omega}$, $\quad\quad u^{\mathrm{e}} \in X^{\mathrm{e}}(\boldsymbol{\Omega})$

$$\overline{\boldsymbol{\Omega}} = \bigcup_{k=1}^{K_{\mathrm{dom}}} \overline{\boldsymbol{\Omega}}^{k} \ , \qquad \text{common configuration}$$

where $\boldsymbol{\Omega} = \boldsymbol{\Omega}_{\mathrm{o}}(\boldsymbol{\mu}_{\mathrm{ref}})$ for $\boldsymbol{\mu}_{\mathrm{ref}} \subset \mathcal{D}^{\dagger}$.

For $\boldsymbol{\Omega}^{k}$, $\boldsymbol{\Omega}_{\mathrm{o}}^{k}(\boldsymbol{\mu})$ we choose in $\mathbf{R}^{2}$ triangles, elliptical triangles and curvy triangles. In $\mathbf{R}^{3}$ we choose parallelepipeds (and in theory tetrahedra).

---

†Connectivity requirement: subdomain intersections must be an entire edge, a vertex, or null.

Geometrical Parametrization: Affine Mappings

Require $\forall \mu \in \mathcal{D}$

$$\overline{\Omega}_{\rm o}^{k}(\mu) = \mathcal{T}^{{\rm aff},k}(\overline{\Omega}^{k};\mu)\ , 1 \le k \le K_{\rm dom}\ ,$$

where

$$\mathcal{T}^{{\rm aff},k}(x;\mu) = C^{{\rm aff},k}(\mu) + G^{{\rm aff},k}(\mu)x\ ,$$

is an invertible affine mapping from $\overline{\Omega}^{k}$ onto $\overline{\Omega}_{\rm o}^{k}(\mu)$.

Further require $\forall \mu \in \mathcal{D}$

$$\mathcal{T}^{{\rm aff},k}(x;\mu) = \mathcal{T}^{{\rm aff},k'}(x;\mu), \quad \forall\, x \in \overline{\Omega}^{k} \cap \overline{\Omega}^{k'}, \quad 1 \le k, k' \le K_{\rm dom}\ ,$$

to ensure a *continuous* piecewise-affine global mapping $\mathcal{T}^{\rm aff}(\,\cdot\,;\mu)$ from $\overline{\Omega}$ onto $\overline{\Omega}_{\rm o}(\mu)$[†].

† It follows that for $w_{\rm o} \in H^1(\Omega_{\rm o}(\mu))$, $w_{\rm o} \circ \mathcal{T}^{\rm aff} = H^1(\Omega)$.

Geometrical Parametrization: Affine Mappings

Require $\forall \mu \in \mathcal{D}$

$$\overline{\Omega}_{\rm o}^{k}(\mu) = \mathcal{T}^{{\rm aff},k}(\overline{\Omega}^{k};\mu)\ , 1 \le k \le K_{\rm dom}\ ,$$

where

$$\mathcal{T}^{{\rm aff},k}(x;\mu) = C^{{\rm aff},k}(\mu) + G^{{\rm aff},k}(\mu)x\ ,$$

is an invertible affine mapping from $\overline{\Omega}^{k}$ onto $\overline{\Omega}_{\rm o}^{k}(\mu)$.

Further require $\forall \mu \in \mathcal{D}$

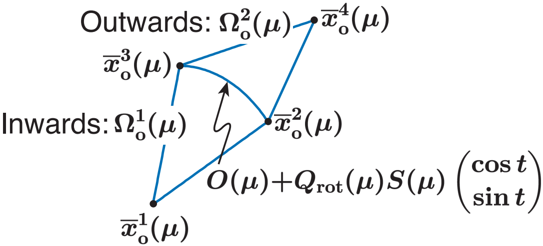
$$\mathcal{T}^{{\rm aff},k}(x;\mu) = \mathcal{T}^{{\rm aff},k'}(x;\mu), \quad \forall\ x \in \overline{\Omega}^{k} \cap \overline{\Omega}^{k'}, \quad 1 \le k, k' \le K_{\rm dom}\ ,$$

to ensure a *continuous* piecewise-affine global mapping $\mathcal{T}^{\rm aff}(\,\cdot\,;\mu)$ from $\overline{\Omega}$ onto $\overline{\Omega}_{\rm o}(\mu)$†.

† It follows that for $w_{\rm o} \in H^1(\Omega_{\rm o}(\mu))$, $w_{\rm o} \circ \mathcal{T}^{\rm aff} = H^1(\Omega)$.

Geometrical Parametrization: Affine Mappings

Elliptical Triangles: definition



$$O(\mu) = (x_{o1}^{\rm cen}, x_{o2}^{\rm cen})^{\rm T}$$

$$Q_{\rm rot}(\mu) = \begin{pmatrix} \cos\phi(\mu) & -\sin\phi(\mu) \\ \sin\phi(\mu) & \cos\phi(\mu) \end{pmatrix}$$

$$S(\mu) = \operatorname{diag}(\rho_1(\mu), \rho_2(\mu))$$

Geometrical Parametrization: Affine Mappings

Elliptical Triangles: constraints

Given $\overline{\boldsymbol{x}}_{\mathrm{o}}^{2}(\boldsymbol{\mu}), \overline{\boldsymbol{x}}_{\mathrm{o}}^{3}(\boldsymbol{\mu})$, find $\overline{\boldsymbol{x}}_{\mathrm{o}}^{1}(\boldsymbol{\mu}), \overline{\boldsymbol{x}}_{\mathrm{o}}^{4}(\boldsymbol{\mu})$ $(\Rightarrow \mathcal{T}^{\mathrm{aff},1\&2})$

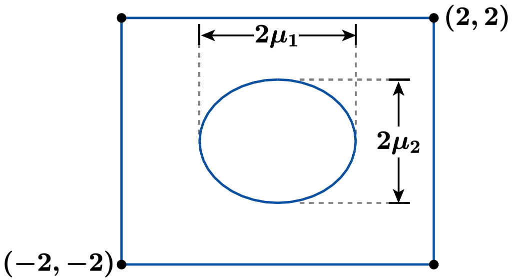
$$\left.\begin{array}{ll} (i) & \text{produce desired elliptical arc} \\ (ii) & \text{satisfy internal angle criterion} \end{array}\right\} \forall \boldsymbol{\mu} \in \mathcal{D};$$

these conditions ensure *continuous invertible* mappings.

† Explicit recipes for admissible $\boldsymbol{x}_{\mathrm{o}}^{1}(\boldsymbol{\mu})$ (Inwards case) and $\boldsymbol{x}_{\mathrm{o}}^{4}(\boldsymbol{\mu})$ (Outwards case) are readily obtained.

Geometrical Parametrization: Affine Mappings

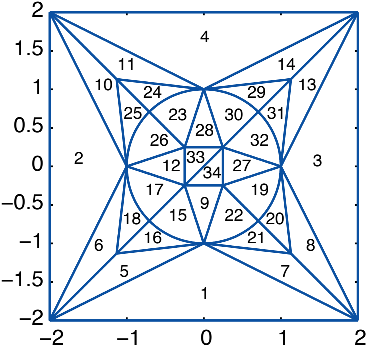
Elliptical Triangles: example (CinS triangulation)



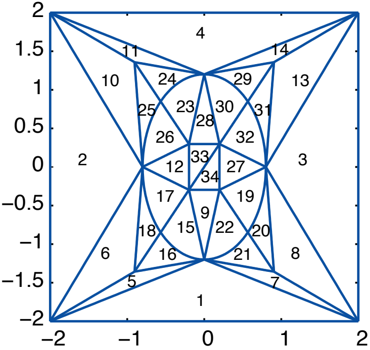
$$\Omega_o(\mu):\ \ \mu = (\mu_1, \mu_2, \ldots) \subset \mathcal{D} \equiv [0.8, 1.2]^2 \times \ldots$$

Affine Mappings

Elliptical Triangles: example (CinS triangulation)



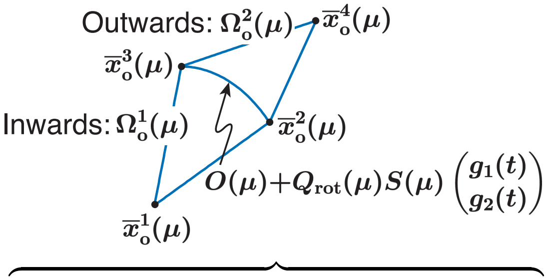
$\Omega = \Omega_o(\mu_{\text{ref}} = (1,1))$



$\Omega_o(\mu = (0.8, 1.2))$

Geometrical Parametrization: Affine Mappings

Curvy Triangles: definition



$$
\begin{aligned}
O(\mu) &= (x_{\rm o1}^{\rm cen}, x_{\rm o2}^{\rm cen})^{\rm T} \\
Q_{\rm rot}(\mu) &= \begin{pmatrix} \cos\phi(\mu) & -\sin\phi(\mu) \\ \sin\phi(\mu) & \cos\phi(\mu) \end{pmatrix} \\
S(\mu) &= \mathrm{diag}(\rho_1(\mu), \rho_2(\mu))
\end{aligned}
$$

Geometrical Parametrization: Affine Mappings

Curvy Triangles: constraints

Given $\overline{\boldsymbol{x}}_{\mathbf{o}}^{2}(\boldsymbol{\mu}), \overline{\boldsymbol{x}}_{\mathbf{o}}^{3}(\boldsymbol{\mu})$, find $\overline{\boldsymbol{x}}_{\mathbf{o}}^{1}(\boldsymbol{\mu}), \overline{\boldsymbol{x}}_{\mathbf{o}}^{4}(\boldsymbol{\mu})$ $\quad (\Rightarrow \boldsymbol{\mathcal{T}}^{\mathbf{aff},\mathbf{1\&2}})$

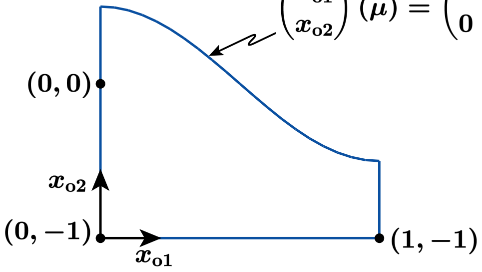
$$\left.\begin{array}{ll} (i) & \text{produce desired curvy arc} \\ (ii) & \text{satisfy internal angle criterion} \end{array}\right\} \forall \boldsymbol{\mu} \in \boldsymbol{\mathcal{D}};$$

these conditions ensure *continuous invertible* mappings.

† Quasi-explicit recipes for admissible $\overline{\boldsymbol{x}}_{\mathbf{o}}^{1}(\boldsymbol{\mu})$ and $\overline{\boldsymbol{x}}_{\mathbf{o}}^{4}(\boldsymbol{\mu})$ can (sometimes) be obtained in the convex/concave case.

Geometrical Parametrization: Affine Mappings

Elliptical Triangles: example (Cosine triangulation)



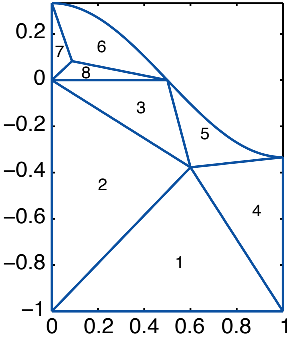
$$\begin{pmatrix} x_{o1} \\ x_{o2} \end{pmatrix}(\mu) = \begin{pmatrix} 1 & 0 \\ 0 & \mu_1 \end{pmatrix} \begin{pmatrix} t \\ \cos \pi t \end{pmatrix}$$

(say)

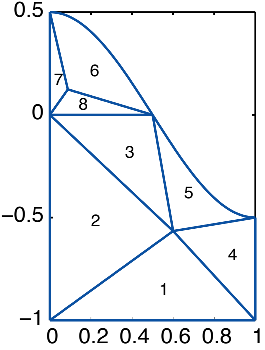
$$\Omega_o(\mu): \ \mu = (\mu_1, \ldots) \subset \mathcal{D} \equiv [\tfrac{1}{6}, \tfrac{1}{2}] \times \ldots$$

Geometrical Parametrization: Affine Mappings

Elliptical Triangles: example (Cosine triangulation)



$\Omega = \Omega_o(\mu_{\mathrm{ref}} = \frac{1}{3})$



$\Omega_o(\mu = \frac{1}{2})$

Geometrical Parametrization: Tensor Transformations

Transformation: Formulation on original domain ($\mathbb{R}^2$)

For $w, v \in H^1(\Omega_o(\mu))$† $\qquad u_o^e(\mu) \in H_0^1(\Omega_o(\mu))$

$$a_o(w,v;\mu) = \sum_{k=1}^{K_{dom}} \int_{\Omega_o^k(\mu)} \left[\begin{array}{ccc} \frac{\partial w}{\partial x_{o1}} & \frac{\partial w}{\partial x_{o2}} & w \end{array} \right] \mathcal{K}_{oij}^k(\mu) \left[\begin{array}{c} \frac{\partial v}{\partial x_{o1}} \\ \frac{\partial v}{\partial x_{o2}} \\ v \end{array} \right]$$

where $\mathcal{K}_o^k\colon\ \mathcal{D} \rightarrow \mathrm{R}^{3\times 3}$, SPD for $1 \leq k \leq K_{dom}$

(note $\mathcal{K}_o^k$ affine in x_o is also permissible).

† We consider the scalar case; the vector case (linear elasticity) admits an analogous treatment.

Geometrical Parametrization: Tensor Transformations

Transformation: Formulation on reference domain

For $w, v \in H^1(\Omega)$ $\qquad u^{\mathrm{e}}(\mu) \in H_0^1(\Omega)$

$$a(w,v;\mu) = \sum_{k=1}^{K_{\mathrm{dom}}} \int_{\Omega^k} \begin{bmatrix} \frac{\partial w}{\partial x_1} & \frac{\partial w}{\partial x_2} & w \end{bmatrix} \mathcal{K}_{ij}^k(\mu) \begin{bmatrix} \frac{\partial v}{\partial x_1} \\ \frac{\partial v}{\partial x_2} \\ v \end{bmatrix}$$

$$\mathcal{K}^k(\mu) = |\det G^{\mathrm{aff},k}(\mu)| D(\mu) \mathcal{K}_{\mathrm{o}}^k(\mu) D^T(\mu), \text{ and}$$

$$D(\mu) = \begin{pmatrix} (G^{\mathrm{aff},k})^{-1} & & \begin{matrix} 0 \\ 0 \end{matrix} \\ 0 & 0 & 1 \end{pmatrix}.$$

Geometrical Parametrization: Tensor Transformations

Transformation: Affine form

Expand

$$a(w, v; \mu) = \underbrace{\mathcal{K}^1_{11}(\mu)}_{\Theta^1(\mu)} \underbrace{\int_{\Omega^1} \frac{\partial w}{\partial x_1} \frac{\partial v}{\partial x_1}}_{a^1(w,v)} + \dots$$

with as many as $Q = 4K$ terms.

We can often greatly reduce the requisite Q.

Achtung! Many interesting problems are **not** affine (or require Q very large).

For example, $\mathcal{K}^k_{\rm o}(x; \mu)$ for general x dependence; and nonzero Neumann conditions on curvy $\partial\Omega$ yield non-affine $a(\,\cdot\,,\,\cdot\,;\mu)$.

Geometrical Parametrization: Tensor Transformations

Transformation: Affine form

Expand

$$a(w, v; \mu) = \underbrace{\mathcal{K}^1_{11}(\mu)}_{\Theta^1(\mu)} \underbrace{\int_{\Omega^1} \frac{\partial w}{\partial x_1} \frac{\partial v}{\partial x_1}}_{a^1(w,v)} + \dots$$

with as many as $Q = 4K$ terms.

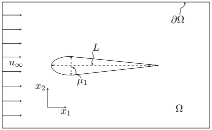
We can often greatly reduce the requisite Q.

Achtung! Many interesting problems are **not** affine (or require Q very large).

For example, $\mathcal{K}^k_{\mathrm{o}}(x; \mu)$ for general x dependence; and nonzero Neumann conditions on curvy $\partial\Omega$ yield non-affine $a(\,\cdot\,,\,\cdot\,;\mu)$.

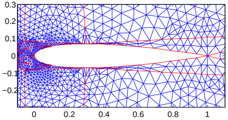
Automatic Geometrical Parametrization: Airfoil Example

- Airfoil of the NACA 4-digits family (symmetric case)



Thickness distribution 4-digits profile

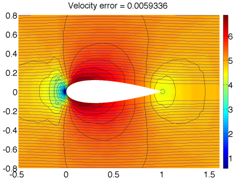
$$x_2 = \frac{\mu_1}{20}(0.2969\sqrt{x_1} - 0.1260 x_1 - 0.3520 x_1^2 + 0.2832 x_1^3 - 0.1021 x_1^4$$

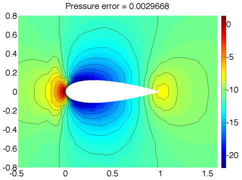


Airfoil geometry description for the rbMIT software

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 & 0 \\ 0 & \pm\mu_1/20 \end{pmatrix} \begin{pmatrix} 1 - t^2 \\ 0.2969t - 0.1260t^2 - 0.3520t^4 + 0.2832t^6 - 0.1021t^8 \end{pmatrix}, \; t \in [0, \sqrt{0.3}]$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & \pm\mu_1/20 \end{pmatrix} \begin{pmatrix} 1 - t^2 \\ 0.2969t - 0.1260t^2 - 0.3520t^4 + 0.2832t^6 - 0.1021t^8 \end{pmatrix}, \; t \in [\sqrt{0.3}, 1]$$



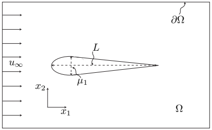


§ Complexity (potential flow):

- # of subdomains $K = 42/84$
- # of bilinear forms $Q = 16/20$

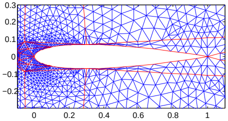
Automatic Geometrical Parametrization: Airfoil Example

- Airfoil of the NACA 4-digits family (symmetric case)



Thickness distribution 4-digits profile

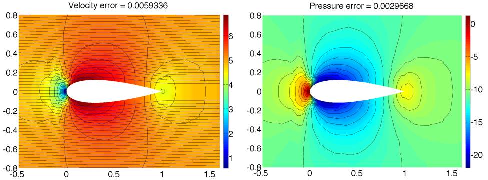
$$x_2 = \frac{\mu_1}{20}(0.2969\sqrt{x_1} - 0.1260x_1 - 0.3520x_1^2 + 0.2832x_1^3 - 0.1021x_1^4)$$



Airfoil geometry description for the rbMIT software

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 & 0 \\ 0 & \pm\mu_1/20 \end{pmatrix} \begin{pmatrix} 1 - t^2 \\ 0.2969t - 0.1260t^2 - 0.3520t^4 + 0.2832t^6 - 0.1021t^8 \end{pmatrix}, \ t \in [0, \sqrt{0.3}]$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & \pm\mu_1/20 \end{pmatrix} \begin{pmatrix} 1 - t^2 \\ 0.2969t - 0.1260t^2 - 0.3520t^4 + 0.2832t^6 - 0.1021t^8 \end{pmatrix}, \ t \in [\sqrt{0.3}, 1]$$

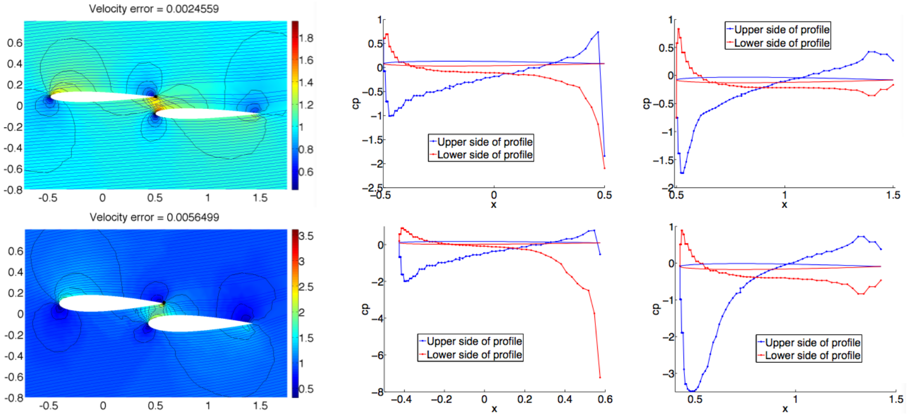


§ Complexity (potential flow):

# of subdomains $K = 42/84$

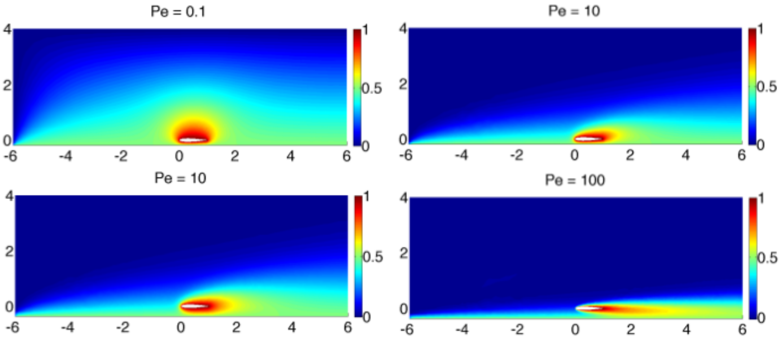
# of bilinear forms $Q = 16/20$

Potential Flows: Output Evaluation



- Parameters: thickness $\mu_1 \in [4, 24]$, angle of attack $\mu_2 \in [0.01, \pi/16]$ and profiles distance $(\mu_3, \mu_4) \in [0.85, 1] \times [1.2, 1.6]$
- Output: pressure coefficient $c_p = \frac{p - p_{in}}{\frac{1}{2}\rho |\mathbf{u}_{in}|^2}$ around the profiles

Thermal Flows: Output Evaluation



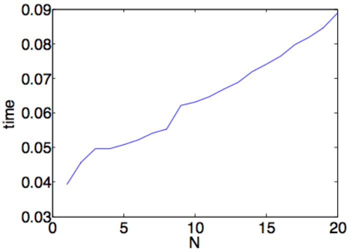
- Thermal Flows around an airfoil with ground effect included
- Physical and geometrical parametrization: thickness $\mu_1 \in [4, 24]$, ground distance $\mu_2 \in [1.5, 3]$, Peclet number $\mu_3 \in [1, 100]$
- The thermal boundary layer on the ground and on profile becomes thinner and more separated for higher Peclet numbers

Potential/Thermal Flows: Computational Costs

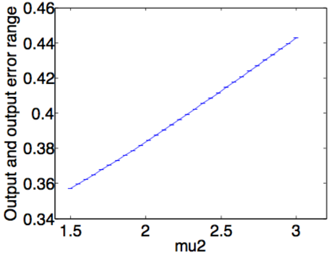
Potential Flow	
mesh nodes $\mathcal{N}$	**3693**
subdomains	**163**
affine operator components Q_a	102
affine rhs components Q_f	3
reduced basis functions N	20
t_{FE}^{online}	$\mathbf{17.15}s$
t_{RB}^{online}	$\mathbf{0.08}s$
speedup	**195**

Thermal Flow	
mesh nodes $\mathcal{N}$	**6727**
subdomains	**88**
affine operator components Q_a	42
reduced basis functions N_{pr}	70
reduced basis functions N_{du}	61
t_{FE}^{online}	$\mathbf{16.97}s$
t_{RB}^{online}	$\mathbf{0.26}s$
speedup	**65**

RB online time as funcion of N (potential flow)



Average outflow temperature w.r. to ground distance μ_2 (thermal flow)



Potential Flow: Shape Optimization Problem

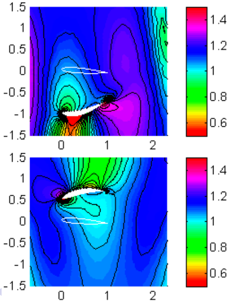
Airfoil inverse design problem

$$\min_{\mu \in \mathcal{D}} \left(\int_0^1 |p(s,\mu) - p_{\text{target}}(s)|^2 \, ds \right)^{1/2} + \lambda \left[\alpha(\mu) - 5^\circ \right]^2 ,$$

$$\text{s.t.} \quad \int_{\Omega_o(\mu)} \nabla u \cdot \nabla v \, d\Omega_o = \int_{\Omega_o(\mu)} f v \, d\Omega_o \; \forall v \in H^1(\Omega_o(\mu))$$

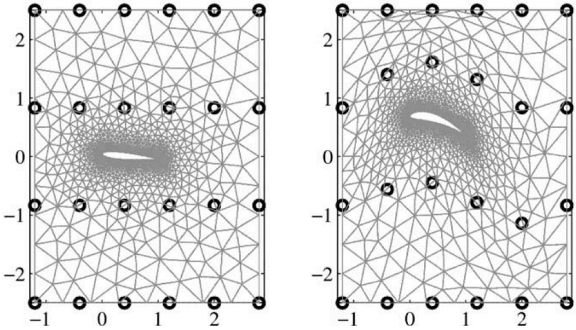
$$u = 0 \text{ on } \Gamma_{out}, \quad \frac{\partial u}{\partial n} = -1 \text{ on } \Gamma_{in}, \quad \frac{\partial u}{\partial n} = 0 \quad \text{elsewhere}$$

- Choose target airfoil (ex: NACA4412) and compute pressure distribution p_{target} on its surface using the Bernoulli equation ($p = p_0 - \frac{1}{2}|\nabla u|^2$)
- Objective: find small perturbation of reference airfoil NACA0012 s.t. pressure distribution on the airfoil surface is close to p_{target}
- Add penalty term to enforce the constraint on the angle of attack (AOA $= 5^\circ$)



Potential Flow: Shape Optimization Problem

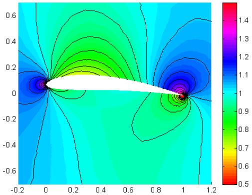
Geometrical parametrization: Free-Form Deformation Techniques



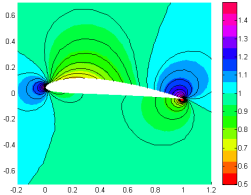
- Geometrical parameters $\mu_1, \ldots, \mu_P$ are chosen as the perturbations of a (small) lattice of FFD control points

Potential Flow: Shape Optimization Problem

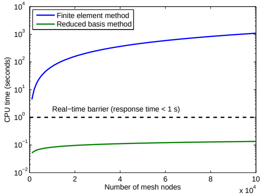
Pressure distributions and computational cost (online solution of the parametric PDE) *



(a) Inverse design



(b) Target airfoil



(c) Computational costs

Number of mesh nodes $\mathcal{N}$	$\mathbf{8043}$
Lattice of FFD control points	$\mathbf{6 \times 4}$
Number of shape parameters†	$\mathbf{8}$
Number of reduced basis functions N‡	$\mathbf{52}$
Error tolerance for RB greedy ε_{tol}^{RB}	$\mathbf{10^{-4}}$
Number of affine expansion terms Q_a	$\mathbf{80}$
Error tolerance for EIM greedy ε_{tol}^{EIM}	$\mathbf{2.5 \times 10^{-3}}$

* Results from T. Lassila, G. Rozza, Comput. Methods Appl. Mech. Engrg. 199 (2010) 1583–1592

† Reduction of 50:1 in parametric complexity compared to explicit nodal deformation

‡ Reduction of 200:1 in linear system dimension

Thermal Flows: Optimal Design of Airfoils

Optimal heat exchange problem

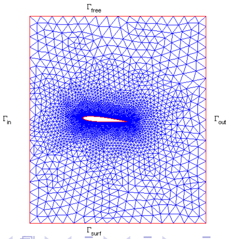
$$\min_{\mu \in \mathcal{D}} \quad \left[\overline{u}_{target} - \frac{1}{|\Gamma_{out}|} \int_{\Gamma_{out}} u \, d\Gamma \right]^2 + \lambda \left[\alpha(\mu) - \alpha_0 \right]^2 ,$$

$$\text{s.t.} \quad \int_{\Omega_o(\mu)} \left(\varepsilon \nabla u \cdot \nabla v + v \vec{b} \cdot \nabla u \right) \, d\Omega_o = \int_{\Omega_o(\mu)} f v \, d\Omega_o$$

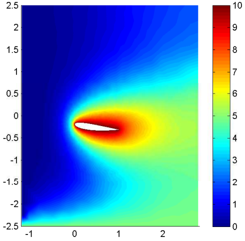
$$\frac{\partial u}{\partial n} = 0 \text{ on } \Gamma_{out}, \; u = T_0 \text{ on } \Gamma_{in} \cup \Gamma_{free},$$

$$u = T_1 \text{ on } \Gamma_{surf}, \quad u = T_2 \text{ on airfoil}$$

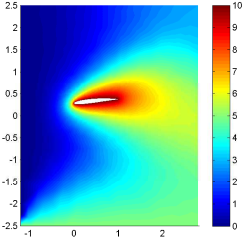
- Objective: find airfoil shape and vertical position s.t. average temperature over outflow equals $\overline{u}_{target}$ and angle of attack equals α_0
- Heat exchange of an airfoil in exterior flow with $\vec{b} = [1; 0]$ and $\varepsilon = 0.2$ is considered
- Penalty term enforces the constraint on the angle of attack (AOA $= \alpha_0$)



Thermal Flows: Optimal Design of Airfoils*



$\alpha_0 = 7^{\circ}$, $\overline{u}_{target} = 4.1$



$\alpha_0 = -5^{\circ}$, $\overline{u}_{target} = 4.5$

Number of mesh nodes $\mathcal{N}$	15718
Lattice of FFD control points	6×6
Number of shape parameters†	8
Number of reduced basis functions N‡	36
Error tolerance for RB greedy ε_{tol}^{RB}	10^{-5}
Number of affine expansion terms Q_a	108
Error tolerance for EIM greedy ε_{tol}^{EIM}	10^{-4}

† Reduction of 100:1 in parametric complexity compared to explicit nodal deformation

‡ Reduction of 436:1 in linear system dimension

* Results from G. Rozza, T. Lassila, A. Manzoni, Proc. of Icosahom Conference, 2009, in press

Environmental Flows: Control of Air Pollution

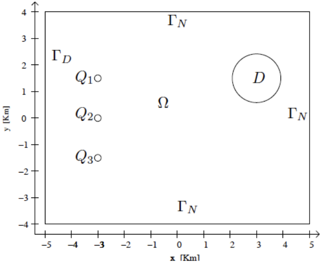
Control of Air pollution

$$\min_{\mu \in \mathcal{D}} \quad J(\mu) = \int_{D_o(\mu_g)} |y(u(\mu_u)) - z_d|^2 d\Omega,$$

$$\text{s.t.} \quad \int_{\Omega_o(\mu_g)} (\nu(\mu_p)\nabla y \cdot \nabla v + \mathbf{V}(\mu_p) \cdot \nabla y) \; d\Omega_o = \int_{\Omega_o(\mu_g)} u(\mu_p) v \; d\Omega_o$$

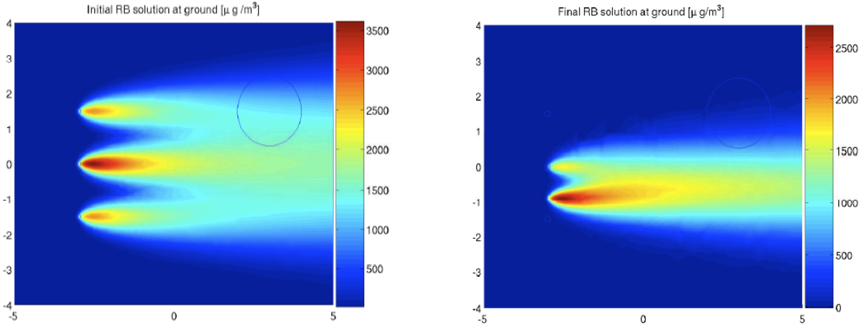
$$\frac{\partial y}{\partial n} = 0 \text{ on } \Gamma_N, \qquad y = 0 \text{ on } \Gamma_D$$

- Goal: to regulate the pollutant emission by industrial plants in order to keep the pollutant level below a fixed threshold over an observation area
- Parameters: **control input** μ_u, which define the control function $u = u(\mu_u)$; **physical input** μ_p (e.g. viscosity, advection velocity); **geometrical input** μ_g (domain configuration)



Environmental Flows: Control of Air Pollution*

Control input μ_u : variable emission rate

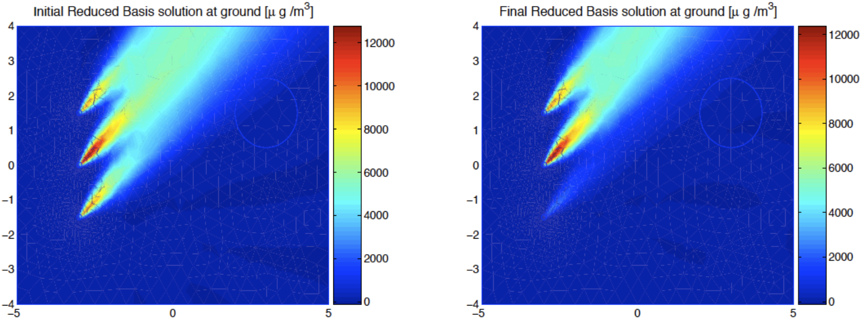


- Use of reduced basis method for the efficient solution of the state and the adjoint parametrized problems

* Results from A. Quarteroni, G. Rozza, A, Quaini, Advances in Numerical Mathematics, 2007, p. 193-216

Environmental Flows: Control of Air Pollution*

Physical Input μ_p : wind direction

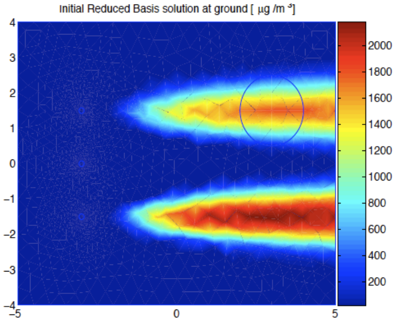


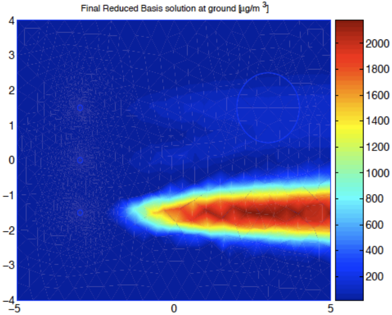
- Use of reduced basis method for the efficient solution of the state and the adjoint parametrized problems

*Results from A. Quarteroni, G. Rozza, A, Quaini, Advances in Numerical Mathematics, 2007, p. 193-216

Environmental Flows: Control of Air Pollution*

Geometrical Input μ_g : parametrized domain





- Use of reduced basis method for the efficient solution of the state and the adjoint parametrized problems

* Results from A. Quarteroni, G. Rozza, A, Quaini, Advances in Numerical Mathematics, 2007, p. 193-216