Weighted reduced order methods for parametrized PDEs with random inputs

Francesco Ballarin[1], Davide Torlo[2], Luca Venturi[3], Gianluigi Rozza[1]

[1] mathLab, Mathematics Area, SISSA, Trieste, Italy
[2] Universität Zürich, Switzerland
[3] Courant Institute of Mathematical Sciences, New York, United States

QUIET 2017
Trieste – July 18, 2017

Introduction and outline of the talk

- **Parametrized** partial differential equations (elliptic, mainly);
- parameters $\rightarrow$ **random inputs**;

- evaluation of some **statistics** on the solution u, e.g. $\mathbb{E}[u]$;
- using **reduced order models** to accelerate **Monte Carlo** methods:
  - how to **weight** solutions during the construction of the ROM;
  - how to **sample** the parameter space during the construction of the ROM;
  - **weighted reduced basis method**

    [P. Chen et al., SIAM J. Numer. Anal., 2013], [**D. Torlo** et al., 2017]
  - **weighted proper orthogonal decomposition method**

    [**L. Venturi** et al., 2017]

- combination with **reduced order stabilization** techniques for advection dominated problems.

Parametrized stochastic partial differential equations

- $\Omega \subset \mathbb{R}^d$, $d = 1, 2, 3$, a **domain**;

- $(A, \mathcal{F}, P)$ a complete **probability** space;
- $\boldsymbol{\mu} : (A, \mathcal{F}) \to (\mathcal{D}, \mathcal{B})$, a **random vector**:
  - $\mathcal{D} \subset \mathbb{R}^p$, a compact set in the **parameter space**;
  - $\boldsymbol{\mu}(\omega) = (\mu_1(\omega), \ldots, \mu_p(\omega))$ independent identically distributed and absolutely continuous **random variables**;

- $H_0^1(\Omega) \subset \mathbb{V} \subset H^1(\Omega)$;
- $S(\Omega) := L^2(A) \bigotimes \mathbb{V}$;
- $u : \Omega \times A \to \mathbb{R}$, i.e. $u \in S(\Omega)$, a **random field**;
- elliptic PDE, e.g., advection–diffusion stochastic equation

$$-\varepsilon(\boldsymbol{\mu}(\omega))\Delta u(\boldsymbol{\mu}(\omega)) + \boldsymbol{\beta}(\boldsymbol{\mu}(\omega)) \cdot \nabla u(\boldsymbol{\mu}(\omega)) = f(\boldsymbol{\mu}(\omega)) \qquad \text{in } \Omega,$$

  s.t. suitable boundary conditions on $\partial\Omega$.

Reduced basis methods: the greedy algorithm

```
Sample Ξ_train ⊂ 𝒟
Pick arbitrary μ^1 ∈ Ξ_train
Define S_0 = ∅, X_0^RB = ∅
for N = 1, ..., N_max
    Perform a PDE solve to compute u(μ^N)
    S_N = S_{N-1} ∪ {μ^N}
    X_N^RB = X_{N-1}^RB ⊕ {u(μ^N)}
    [ε_N, μ^N] = max_{μ∈Ξ_train} Δ_N(μ)
    if ε_N ≤ tol
        break
    end
end
```

where $\Delta_N(\boldsymbol{\mu})$ is a sharp, *inexpensive a posteriori error bound* for $\|u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_{\mathbb{V}}$, being $u_N(\boldsymbol{\mu})$ the RB solution of dimension N.

J. S. Hesthaven, G. Rozza, B. Stamm. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations.* SpringerBriefs in Mathematics. Springer International Publishing, 2015

Weighted reduced basis methods: motivation

The introduction of a **weight** in the greedy algorithm reflects our desire of minimizing the squared norm error of the reduced order approximation, i.e.

$$\mathbb{E}\left[\|u-u_N\|_{\mathbb{V}}^2\right] = \int_A \|u(\boldsymbol{\mu}(\omega)) - u_N(\boldsymbol{\mu}(\omega))\|_{\mathbb{V}}^2 \; dP(\omega) =$$

$$= \int_{\mathcal{D}} \|u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_{\mathbb{V}}^2 \, \rho(\boldsymbol{\mu}) \; d\boldsymbol{\mu},$$

being $\rho : A \to \mathbb{R}$ the **probability density distribution** of $\boldsymbol{\mu}$.
Thus,

$$\mathbb{E}\left[\|u-u_N\|_{\mathbb{V}}^2\right] \leq \int_{\mathcal{D}} \Delta_N(\boldsymbol{\mu})^2 \rho(\boldsymbol{\mu}) d\boldsymbol{\mu},$$

This motivates the choice of the weight

$$w(\boldsymbol{\mu}) = \sqrt{\rho(\boldsymbol{\mu})}$$

and the introduction of the error bound

$$\Delta_N^w(\boldsymbol{\mu}) := \Delta_N(\boldsymbol{\mu}) \sqrt{\rho(\boldsymbol{\mu})}.$$

P. Chen, A. Quarteroni, and G. Rozza. A weighted reduced basis method for elliptic partial differential equations with random input data. SIAM Journal on Numerical Analysis, 51(6):3163–3185, 2013.

Weighted reduced basis methods: the greedy algorithm

```
Properly sample Ξ_train ⊂ 𝒟
Pick arbitrary μ^1 ∈ Ξ_train
Define S_0 = ∅, X_0^RB = ∅
for N = 1, ..., N_max
    Perform a PDE solve to compute u(μ^N)
    S_N = S_{N-1} ∪ {μ^N}
    X_N^RB = X_{N-1}^RB ⊕ {u(μ^N)}
    [ε_N, μ^N] = max_{μ∈Ξ_train} Δ_N^w(μ)
    if ε_N ≤ tol
        break
    end
end
```

Properly sample $\Xi_{\text{train}} \subset \mathcal{D}$

Pick arbitrary $\boldsymbol{\mu}^1 \in \Xi_{\text{train}}$

Define $S_0 = \varnothing, X_0^{RB} = \varnothing$

for $N = 1, \ldots, N_{\max}$

  Perform a PDE solve to compute $u(\boldsymbol{\mu}^N)$

  $S_N = S_{N-1} \cup \{\boldsymbol{\mu}^N\}$

  $X_N^{RB} = X_{N-1}^{RB} \oplus \{u(\boldsymbol{\mu}^N)\}$

  $[\varepsilon_N, \boldsymbol{\mu}^N] = \max_{\boldsymbol{\mu}\in\Xi_{\text{train}}} \Delta_N^w(\boldsymbol{\mu})$

  if $\varepsilon_N \leq \text{tol}$

    break

  end

end

P. Chen, A. Quarteroni, and G. Rozza. A weighted reduced basis method for elliptic partial differential equations with random input data. SIAM Journal on Numerical Analysis, 51(6):3163–3185, 2013.

Reduction by proper orthogonal decomposition

Sample $\Xi_{\text{train}} \subset \mathcal{D}$
for $N = 1, \ldots, M \equiv \text{card}(\Xi_{\text{train}})$
    Pick the N-th element $\boldsymbol{\mu}^N$ in Ξ_{train}
    Perform a PDE solve to compute $u(\boldsymbol{\mu}^N)$
end

Assemble the correlation matrix $\mathbb{C} \in \mathbb{R}^{M \times M}$,
with entries

$$C_{ij} = \frac{1}{M}(u(\boldsymbol{\mu}^i), u(\boldsymbol{\mu}^j))_{\mathbb{V}}, \qquad i, j = 1, \ldots, M$$

Compute the eigenvalues of $\mathbb{C}$, sort them in decreasing order, and retain the ones that are larger than tol, as well as their corresponding eigenfunctions.

J. S. Hesthaven, G. Rozza, B. Stamm. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. SpringerBriefs in Mathematics. Springer International Publishing, 2015

Weighted proper orthogonal decomposition: motivation

Standard POD algorithm results in the optimal N-dimensional subspace of $\mathbb{V}$ which minimizes

$$\frac{1}{M}\sum_{i=1}^{M}\left\|u(\boldsymbol{\mu}^i)-u_N(\boldsymbol{\mu}^i)\right\|_{\mathbb{V}}^2$$

Now, consider a **weight** $w:\mathcal{D}\to\mathbb{R}^+$ and minimize instead

$$\frac{1}{M}\sum_{i=1}^{M} w(\boldsymbol{\mu}^i)\ \left\|u(\boldsymbol{\mu}^i)-u_N(\boldsymbol{\mu}^i)\right\|_{\mathbb{V}}^2$$

In practice, this amounts to computing the eigenvalues of the following **weighted correlation matrix**

$$\mathbb{C}_w=\mathbb{W}^{\frac{1}{2}}\mathbb{C}$$

where $\mathbb{W}=\mathrm{diag}\{w(\boldsymbol{\mu}^1),\dots,w(\boldsymbol{\mu}^M)\}$.

L. Venturi, F. Ballarin, and G. Rozza. Weighted POD–Galerkin methods for parametrized partial differential equations in uncertainty quantification problems. In preparation, 2017

Weighted proper orthogonal decomposition: offline stage

```
Sample Ξ_train ⊂ 𝒟
for N = 1, …, M ≡ card(Ξ_train)
    Pick the N-th element μ^N in Ξ_train
    Perform a PDE solve to compute u(μ^N)
end
```

Sample $\Xi_{\text{train}} \subset \mathcal{D}$
for $N = 1, \dots, M \equiv \text{card}(\Xi_{\text{train}})$
Pick the N-th element $\boldsymbol{\mu}^N$ in Ξ_{train}
Perform a PDE solve to compute $u(\boldsymbol{\mu}^N)$
end

Assemble the weighted correlation matrix $\mathbb{C}_w \in \mathbb{R}^{M \times M}$.

Compute the eigenvalues of $\mathbb{C}_w$, sort them in decreasing order, and retain the ones that are larger than tol, as well as their corresponding eigenfunctions.

L. Venturi, F. Ballarin, and G. Rozza. Weighted POD–Galerkin methods for parametrized partial differential equations in uncertainty quantification problems. In preparation, 2017

Weighted proper orthogonal decomposition: choice of the weight

- the choice $w(\boldsymbol{\mu}) \equiv \rho(\boldsymbol{\mu})$ with $\boldsymbol{\mu}^i \sim \text{Unif}(\mathcal{D})$ results in the minimization of

$$\frac{1}{M}\sum_{i=1}^{M} \rho(\boldsymbol{\mu}^i) \; \left\| u(\boldsymbol{\mu}^i) - u_N(\boldsymbol{\mu}^i) \right\|_{\mathbb{V}}^2 \approx$$

$$\approx \int_{\mathcal{D}} \| u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu}) \|_{\mathbb{V}}^2 \, \rho(\boldsymbol{\mu}) \; d\boldsymbol{\mu} = \mathbb{E}\left[\| u - u_N \|_{\mathbb{V}}^2 \right],$$

  i.e. a **Uniform Monte Carlo quadrature** for the squared norm error of the reduced order approximation.

- the choice $w(\boldsymbol{\mu}) \equiv 1$ with $\boldsymbol{\mu}^i \sim \text{Distribution}(\mathcal{D})$ results in the minimization of

$$\frac{1}{M}\sum_{i=1}^{M} \left\| u(\boldsymbol{\mu}^i) - u_N(\boldsymbol{\mu}^i) \right\|_{\mathbb{V}}^2 \approx$$

$$\approx \int_{\mathcal{D}} \| u(\boldsymbol{\mu}(\omega)) - u_N(\boldsymbol{\mu}(\omega)) \|_{\mathbb{V}}^2 \, dP(\omega) = \mathbb{E}\left[\| u - u_N \|_{\mathbb{V}}^2 \right],$$

  i.e. a **Monte Carlo quadrature** for the squared norm error of the reduced order approximation.

L. Venturi, F. Ballarin, and G. Rozza. Weighted POD–Galerkin methods for parametrized partial differential equations in uncertainty quantification problems. In preparation, 2017

Weighted proper orthogonal decomposition: choice of **weight and sampling**

- the previous choices resulted in the minimization of a (Uniform) Monte Carlo **quadrature** for the squared norm error of the reduced order approximation.
- moreover, we can consider more general quadrature rules of the form

$$\mathcal{U}(f) = \sum_{i=1}^{M} \omega^i f(\boldsymbol{x}^i)$$

for every integrable function $f : \mathcal{D} \to \mathbb{R}$, where $\boldsymbol{x}^1, \ldots, \boldsymbol{x}^M \in \mathcal{D}$ are the nodes of the quadrature and $\omega^1, \ldots, \omega^M$ are the respective weights. This results in the following approximation:

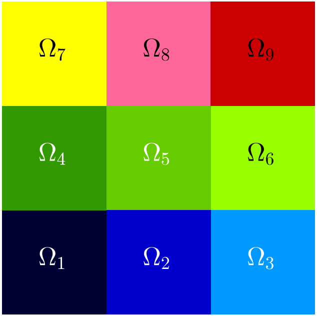
$$\mathbb{E}\left[\|u - u_N\|_{\mathbb{V}}^2\right] \approx \frac{1}{M} \sum_{i=1}^{M} \omega_i \rho(\boldsymbol{x}^i) \ \left\|u(\boldsymbol{x}^i) - u_N(\boldsymbol{x}^i)\right\|_{\mathbb{V}}^2$$

which motivates the following choice:

- **sample** the parameter spaces with $\{\boldsymbol{\mu}^i \equiv \boldsymbol{x}^i\}_{i=1}^M$, and
- **weigh** as $w(\boldsymbol{\mu}^i) \equiv \omega^i \rho(\boldsymbol{\mu}^i)$, $i = 1, \ldots, M$.

L. Venturi, F. Ballarin, and G. Rozza. Weighted POD–Galerkin methods for parametrized partial differential equations in uncertainty quantification problems. In preparation, 2017

Test case 1: 3×3 thermal block

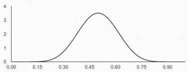


RB	w^i	$\lvert\Xi_{\text{train}}\rvert$
Standard	1	2000
Weighted - Unif.	$\sqrt{\rho^i}$	2000
Weighted - Distr.	$\sqrt{\rho^i}$	2000

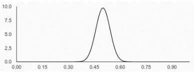
POD	w^i	$\lvert\Xi_{\text{train}}\rvert$
Standard	1	500
Monte-Carlo	1	500
Uniform M.-C.	ρ^i	2000
Gauss-Jacobi	ω^i	512
Gauss-Legendre	$\omega^i \rho^i$	512

$$\sum_{i=1}^{9} \mu_i \int_{\Omega_i} \nabla u(\mu) \cdot \nabla v \; dx = \int_{\Omega} v \; dx, \quad \forall v \in \mathbb{V}$$

$$\boldsymbol{\mu} \in [1,3]^9 : \frac{\mu_i - 1}{2} \sim \text{Beta}(\alpha, \beta)$$

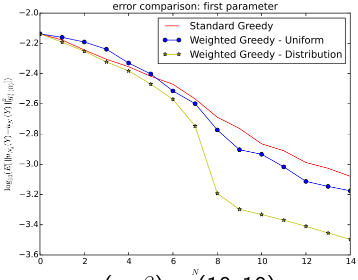


$\alpha = \beta = 10$

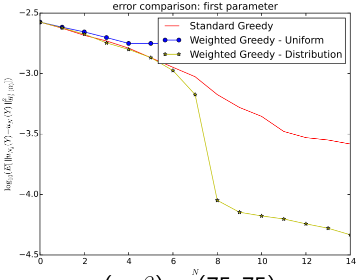


$\alpha = \beta = 75$

Test case 1: **RB vs weighted RB**



$(\alpha, \beta) = (10, 10)$

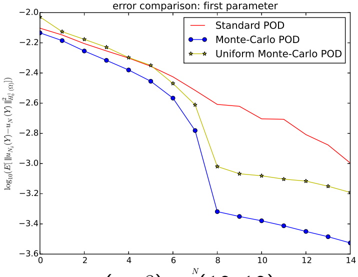


$(\alpha, \beta) = (75, 75)$

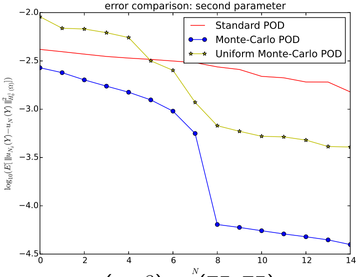
- both **weighting** and **correct sampling** are necessary to obtain good results;
- **weighted** Greedy with sampling from **distribution** guarantees best results;
- **weighted** Greedy with **uniform** sampling is comparable to standard greedy (left) or even worse (right)

Test case 1: POD vs weighted POD

- Monte Carlo POD minimizes $\frac{1}{M}\sum_{i=1}^{M} \left\| u(\boldsymbol{\mu}^i) - u_N(\boldsymbol{\mu}^i) \right\|_{\mathbb{V}}^2$ for $\boldsymbol{\mu}^i \sim$ **Distribution**$(\mathcal{D})$;
- Uniform Monte Carlo POD minimizes $\frac{1}{M}\sum_{i=1}^{M} \rho(\boldsymbol{\mu}^i) \left\| u(\boldsymbol{\mu}^i) - u_N(\boldsymbol{\mu}^i) \right\|_{\mathbb{V}}^2$ for $\boldsymbol{\mu}^i \sim$ **Unif**$(\mathcal{D})$.



$(\alpha, \beta) = (10, 10)$

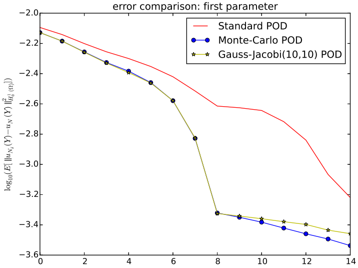


$(\alpha, \beta) = (75, 75)$

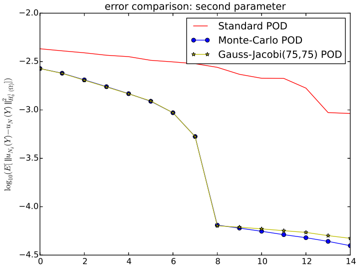
- both **weighting** and **correct sampling** are necessary to obtain good results;
- **weighted** POD with sampling from **distribution** guarantees best results;
- **weighted** POD with **uniform** sampling is better than standard POD but worse than the best weighted approach.

Test case 1: POD vs weighted POD

- Monte Carlo POD minimizes $\frac{1}{M}\sum_{i=1}^{M} \left|\left|u(\boldsymbol{\mu}^i) - u_N(\boldsymbol{\mu}^i)\right|\right|_{\mathbb{V}}^2$ for $\boldsymbol{\mu}^i \sim$ **Distribution**$(\mathcal{D})$;
- Gauss-Jacobi(α, β) employs $(\boldsymbol{\mu}^i, \omega^i)$ from the **quadrature rule**.



$(\alpha, \beta) = (10, 10)$

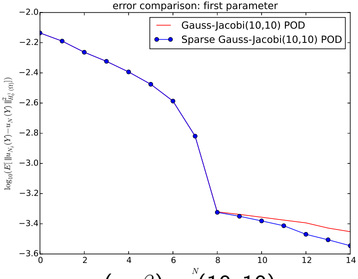


$(\alpha, \beta) = (75, 75)$

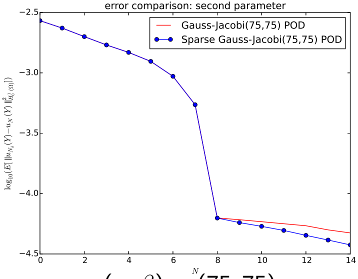
- Gauss-Jacobi(α, β) performs **as well as** Monte Carlo;
- in all cases, the difference between weighted and standard approaches are more marked when the distribution is **highly concentrated** (right).
- we have been using **tensor product** Gauss-Jacobi(α, β) in $\mathbb{R}^9$. Can we retain the good approximation properties of this weighted POD even a less expensive training phase with sampling based on **sparse grids**?

Test case 1: using sparse grids

- Gauss-Jacobi(α, β) employs $(\boldsymbol{\mu}^i, \omega^i)$ from the **tensor product quadrature rule**, resulting in $|\Xi_{\text{train}}| = 512$;
- Sparse Gauss-Jacobi(α, β) employs $(\boldsymbol{\mu}^i, \omega^i)$ from the **sparse Smolyak quadrature rule of order** $q = 11$, resulting in $|\Xi_{\text{train}}| = 181$.



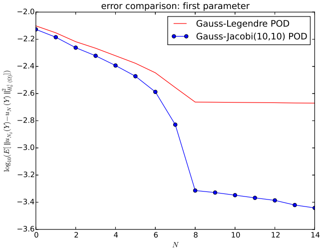
$(\alpha, \beta) = (10, 10)$

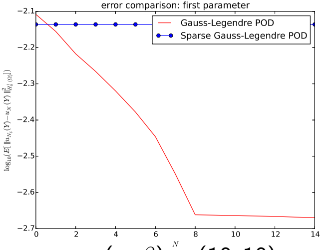


$(\alpha, \beta) = (75, 75)$

- **Sparse** Gauss-Jacobi(α, β) performs as well as the corresponding tensor product rule;
- **Sparse** Gauss-Jacobi(α, β) allows to save more than 60% of offline computatoins.

Test case 1: **watch out when sparsify-ing!**





$(\alpha, \beta) = (10, 10)$

- (tensor) Gauss-Legendre formula is **not as representative** as (tensor) Gauss-Jacobi(α, β);
- trying to **sparsify it** results in an extremely **bad** reduced order model.

Stabilized reduced basis methods for advection dominated problems

Advection dominated stochastic PDE:

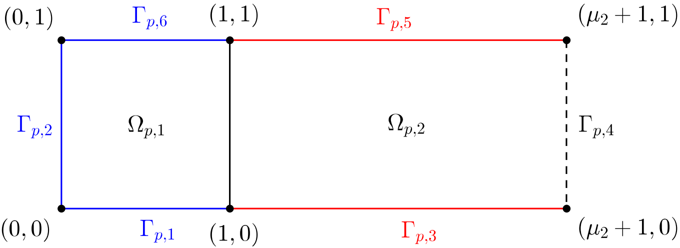
$$-\varepsilon(\boldsymbol{\mu}(\omega))\Delta u(\boldsymbol{\mu}(\omega)) + \boldsymbol{\beta}(\boldsymbol{\mu}(\omega)) \cdot \nabla u(\boldsymbol{\mu}(\omega)) = f(\boldsymbol{\mu}(\omega)) \qquad \text{in } \Omega,$$

s.t. suitable boundary conditions on $\partial\Omega$.

- requires *Offline* **stabilization**, e.g. SUPG, to numerically handle cells which high local Péclet number;
- what about *Online*?
  - do stabilize also *Online*, to guarantee consistency $\rightarrow$ ***Offline-Online* stabilized RB method**
  - do not stabilize *Online*, to avoid assembly of all stabilization terms and (possibly) gain in performance $\rightarrow$ ***Offline-only* stabilized RB method**

P. Pacciarini and G. Rozza, Stabilized reduced basis method for parametrized advection–diffusion PDEs, Comput. Methods Appl. Mech. Engrg., 274:1-18, 2014.

Test case 2: Graetz problem

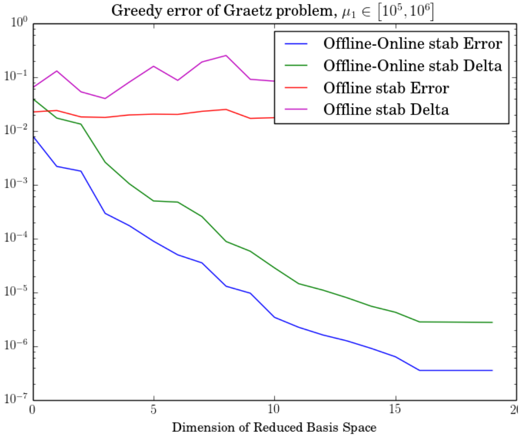


$$\begin{cases} -\frac{1}{\mu_1}\Delta u(\boldsymbol{\mu}) + 4y(1-y)\partial_x u(\boldsymbol{\mu}) = 0 & \text{in } \Omega_p(\boldsymbol{\mu}) \\ u(\boldsymbol{\mu}) = 0 & \text{on } \Gamma_{p,1}(\boldsymbol{\mu}) \cup \Gamma_{p,2}(\boldsymbol{\mu}) \cup \Gamma_{p,6}(\boldsymbol{\mu}) \\ u(\boldsymbol{\mu}) = 1 & \text{on } \Gamma_{p,3}(\boldsymbol{\mu}) \cup \Gamma_{p,5}(\boldsymbol{\mu}) \\ \frac{1}{\mu_1}\frac{\partial u}{\partial \boldsymbol{n}} = 0 & \text{on } \Gamma_{p,4}(\boldsymbol{\mu}). \end{cases}$$

$$\begin{array}{lll} \mu_1 \sim 10^{1+5\cdot X_1} & \text{where } X_1 \sim \text{Beta}(4,2), & \mu_1 \in [10^1, 10^6] \\ \mu_2 \sim 0.5 + 3.5X_2 & \text{where } X_2 \sim \text{Beta}(3,4), & \mu_2 \in [0.5, 4] \end{array}$$

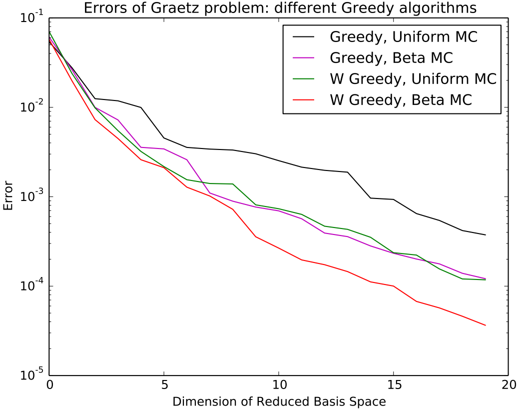
D. Torlo, F. Ballarin, and G. Rozza. Stabilized weighted reduced basis methods for parametrized advection dominated problems with random inputs. Submitted, 2017

Test case 2: **(deterministic) Offline-Online vs Offline-only**



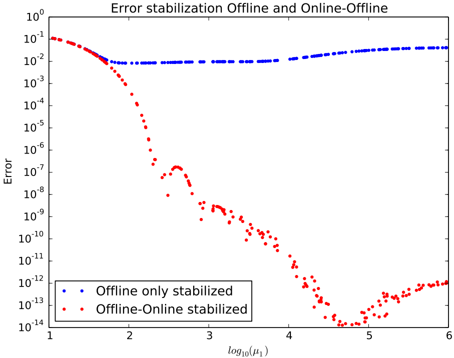
- *Offline-Online* stabilized RB outperforms *Offline-only* method by several order of magnitudes for **large** Péclet numbers;
- a similar analysis shows that, instead, *Offline-Online* and *Offline-only* methods are comparable for **small** Péclet numbers, where stabilization is not needed.

Test case 2: **stabilized RB vs stabilized weighted RB**



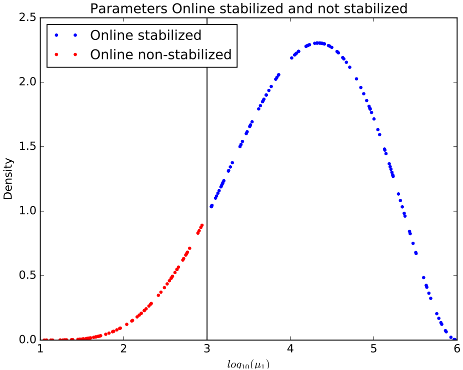
- both **weighting** and **correct sampling** are necessary to obtain good results;
- **weighted** Greedy with sampling from **distribution** guarantees best results;
- **weighted** Greedy with **uniform** sampling is comparable to standard greedy with **sampling from distribution**; both are better than Greedy with **uniform** sampling.

Test case 2: selective online stabilization



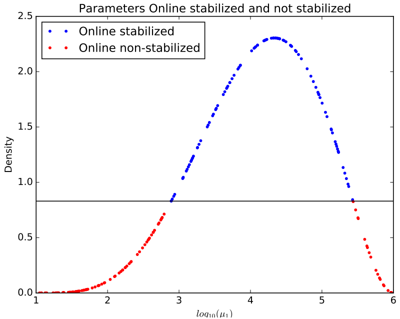
- for low Péclet number ($\mu_1 \leq 10^2$), *Offline-Online* stabilization and *Offline only* stabilization produce very similar results. Thus, we would prefer the **less expensive** *Offline only* stabilization procedure;
- in the regions where the density of $\boldsymbol{\mu}$ is very small, even a large error would be **less relevant** in terms of the probabilistic mean error;
- $\Rightarrow$ enable the more expensive online stabilization only for parameters with high **density** (which would affect more the mean error) or parameters with large **Péclet numbers** (were the more expensive assembly is fully justified by the convection dominated regime)

Test case 2: **selective online stabilization**



Threshold $\widetilde{\mu}_1$	Error	Percentage non-stabilized
10^1	$7.9673 \cdot 10^{-4}$	0%
$10^{1.5}$	$8.0704 \cdot 10^{-4}$	10%
10^2	$10.0060 \cdot 10^{-4}$	20%
$10^{2.5}$	$18.2806 \cdot 10^{-4}$	33%
10^3	$33.4593 \cdot 10^{-4}$	45%
10^6	0.021128	100%

Test case 2: **selective online stabilization**



Threshold $\widetilde{\nu}$	Threshold $\widetilde{\rho}$	Error	Percentage non-stabilized
0	0	$7.9673 \cdot 10^{-4}$	0%
0.001	0.02233	$9.3222 \cdot 10^{-4}$	15%
0.002	0.04423	$9.6456 \cdot 10^{-4}$	17%
0.005	0.09094	$14.7861 \cdot 10^{-4}$	21%
0.01	0.13877	$15.9482 \cdot 10^{-4}$	25%
0.02	0.21433	$25.6017 \cdot 10^{-4}$	30%
0.05	0.38244	$49.1931 \cdot 10^{-4}$	38%
0.1	0.89068	$66.7488 \cdot 10^{-4}$	45%
1	∞	0.021128	100%

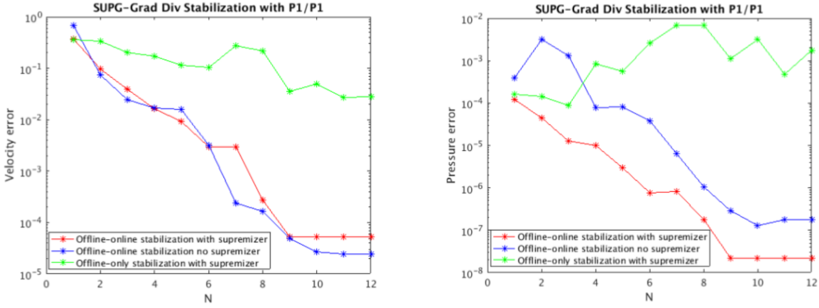
RBniCS

```
from dolfin import *
from RBniCS import *

class Graetz(EllipticCoerciveRB):
 ...

 def assemble_truth_a(self):
  a0 = inner(grad(u),grad(v))*dx(1)
  ...
  a4 = h*y*(1-y)*u.dx(0)*v.dx(0)*dx(2)
  ...

graetz = Graetz()
...
graetz.offline()
...
graetz.error_analysis()
...
```

http://mathlab.sissa.it/rbnics

Extensions to CFD problems (with Shafqat Ali, SISSA)

- working on extending these ideas to **CFD** problems with (possibly) large Reynolds numbers;
- the effect of **stabilization** is twofold:
    - convection dominated regime;
    - inf-sup stability at the reduced order level (supremizers: competition? collaboration?);



(a) Velocity for $\mathbb{P}_1/\mathbb{P}_1$ **(b)** Pressure for $\mathbb{P}_1/\mathbb{P}_1$

Conclusion

- **stochastic** parametrized partial differential equations;

- **reduced order methods** based on:
  - **weighted reduced basis method**;
  - **weighted proper orthogonal decomposition method**;

- need to **weigh** and **sample** from relevant distribution during the construction stage;
- weights and samples based on (possibly sparse) quadrature rules;
- stabilization for **advection dominated** problems;
- opportunity to **selectively enable** online stabilization based either on probability density function or on the Péclet number.

References

1. P. Chen, A. Quarteroni, and G. Rozza. A weighted reduced basis method for elliptic partial differential equations with random input data. SIAM Journal on Numerical Analysis, 51(6):3163–3185, 2013.
2. P. Pacciarini and G. Rozza, Stabilized reduced basis method for parametrized advection–diffusion PDEs, Comput. Methods Appl. Mech. Engrg., 274:1-18, 2014.
3. **L. Venturi, F. Ballarin, and G. Rozza**. Weighted POD–Galerkin methods for parametrized partial differential equations in uncertainty quantification problems. In preparation, 2017.
4. **D. Torlo, F. Ballarin, and G. Rozza**. Stabilized weighted reduced basis methods for parametrized advection dominated problems with random inputs. Submitted, 2017.
5. **S. Ali, F. Ballarin, and G. Rozza**. Stabilized reduced basis methods for parametrized Stokes and Navier-Stokes equations. In preparation, 2017.

Thanks for your attention!

Acknowledgements: European Research Council (ERC) AROMA-CFD project.