Multi space RB preconditioners for large-scale parametrized PDEs

Niccolò Dal Santo*

* École Polytechnique Fédérale de Lausanne, CMCS, Institute of Mathematics, Switzerland

niccolo.dalsanto@epfl.ch

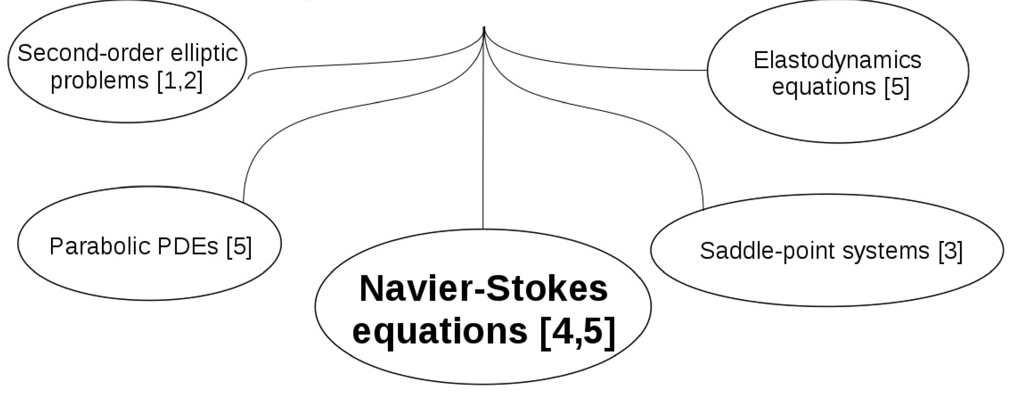
1. The main ideas

Goal: speed up the iterative solution of parametrized finite element linear systems with a preconditioner which exploits the reduced basis method.

Preconditioning techniques
HPC OPTIMALITY
SCALABILITY

Reduced basis method
PARAMETRIZED PROBLEMS
ACCURACY EFFICIENCY

Multi space reduced basis preconditioners



- Efficient solution of the linear systems arising from the finite element (FE) discretization of parametrized PDEs.
- Focus on the parametrized Navier-Stokes (NS) equations.
- Combination of a fine grid preconditioner with a reduced basis (RB) coarse component to exploit the parameter dependence.
- Introduction of a new Multi space RB method for NS equations:
  - Efficient treatment of the linearized term of the NS equations.
  - Approximated well-posedness of the underlying RB coarse component through a velocity-enrichment strategy.

2. The parametrized unsteady Navier-Stokes problem

Consider the NS equations in a $\boldsymbol{\mu}$-dependent domain $\Omega(\boldsymbol{\mu})$

$$\begin{cases} \dfrac{\partial \vec{u}(\boldsymbol{\mu})}{\partial t} + \vec{u}(\boldsymbol{\mu}) \cdot \nabla \vec{u}(\boldsymbol{\mu}) - \nabla \cdot \boldsymbol{\sigma}(\boldsymbol{\mu}) = \vec{0} & \text{in } \Omega(\boldsymbol{\mu}) \times (0,T) \\ \nabla \cdot \vec{u}(\boldsymbol{\mu}) = 0 & \text{in } \Omega(\boldsymbol{\mu}) \times (0,T) \\ + \text{ b.c.} \\ + \text{ i.c.} \end{cases}$$

$$\boldsymbol{\sigma}(\boldsymbol{\mu}) = \boldsymbol{\sigma}(\vec{u}(\boldsymbol{\mu}), p(\boldsymbol{\mu})) = -p(\boldsymbol{\mu})\mathbf{I} + \nu\left(\nabla \vec{u}(\boldsymbol{\mu}) + \nabla \vec{u}(\boldsymbol{\mu})^T\right).$$

The FE method and a semi-implicit time advance scheme yield for any $\boldsymbol{\mu} \in \mathcal{D}$ and for any $n = 0, \dots, N_t - 1$, the linear system

$$\underbrace{\begin{bmatrix} \frac{\alpha_1}{\Delta t}\mathbf{M}^u(\boldsymbol{\mu}) + \mathbf{D}(\boldsymbol{\mu}) + \mathbf{C}(\mathbf{u}^{n,*}(\boldsymbol{\mu}); \boldsymbol{\mu}) & \mathbf{B}^T(\boldsymbol{\mu}) \\ \mathbf{B}(\boldsymbol{\mu}) & \mathbf{0} \end{bmatrix}}_{\mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu}) \ \in \mathbb{R}^{N_h \times N_h}} \underbrace{\begin{bmatrix} \mathbf{u}^{n+1}(\boldsymbol{\mu}) \\ \mathbf{p}^{n+1}(\boldsymbol{\mu}) \end{bmatrix}}_{\mathbf{z}^{n+1}(\boldsymbol{\mu})} = \mathbf{g}^{n+1}(\boldsymbol{\mu})$$

Currently employed techniques to deal with the NS system are:

- **Preconditioned Krylov methods**: domain decomposition, multilevel, block preconditioners (PCD, LSC, SIMPLE).
- **RB methods**: **exploits the parameter dependence** but its efficiency may deteriorate when targeting accurate approximations (due to affine approximations and RB space accuracy).

3. How to exploit the RB method for preconditioning?

The RB method in a nutshell: approximate $\mathbf{u}^n(\boldsymbol{\mu})$, $\mathbf{p}^n(\boldsymbol{\mu})$ in two *RB spaces* $\mathbf{V}_{N_u}$ and $\mathbf{V}_{N_p}$

$$\mathbf{u}^n(\boldsymbol{\mu}) \approx \mathbf{V}_{N_u}\mathbf{u}_N^n(\boldsymbol{\mu}), \qquad \mathbf{p}^n(\boldsymbol{\mu}) \approx \mathbf{V}_{N_p}\mathbf{p}_N^n(\boldsymbol{\mu})$$

- **offline**: build $\mathbf{V}_{N_u}$, $\mathbf{V}_{N_p}$ through POD by FE snapshots and up to a prescribed tolerance ε_{POD} (e.g. with a double POD-approach), then $\mathbf{V} = diag(\mathbf{V}_{N_u}, \mathbf{V}_{N_p})$.

  ! Warning: $\mathbf{V}_{N_u}$ must be enriched with a supremizer basis to obtain an (approximately) inf-sup stable RB formulation.
- **online**: given a new $\boldsymbol{\mu}$, for any n solve a *small* RB problem:

$$\underbrace{\mathbf{V}^T\mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{V}}_{\mathbf{N}_N(\mathbf{V}_{N_u}\mathbf{u}_N^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})} \underbrace{\begin{bmatrix} \mathbf{u}_N^{n+1}(\boldsymbol{\mu}) \\ \mathbf{p}_N^{n+1}(\boldsymbol{\mu}) \end{bmatrix}}_{\mathbf{z}_N^{n+1}(\boldsymbol{\mu})} = \underbrace{\mathbf{V}^T\mathbf{g}^{n+1}(\boldsymbol{\mu})}_{\mathbf{g}_N^{n+1}(\boldsymbol{\mu})}, \qquad (1)$$

- See [4, 5] for the stability and the efficient construction of (1).

A RB low-rank solver $\mathbf{Q}_N(\boldsymbol{\mu})$ s.t.

$$\mathbf{z}^{n+1}(\boldsymbol{\mu}) \approx \mathbf{V}\mathbf{z}_N^{n+1}(\boldsymbol{\mu}) = \mathbf{Q}_N(\boldsymbol{\mu})\mathbf{g}^{n+1}(\boldsymbol{\mu})$$
$$= \mathbf{V}\mathbf{N}_N^{-1}(\mathbf{V}_{N_u}\mathbf{u}_N^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{V}^T\mathbf{g}^{n+1}(\boldsymbol{\mu}),$$

we use $\mathbf{Q}_N(\boldsymbol{\mu})$ as **coarse component in a two level preconditioner** $\mathbf{Q}(\boldsymbol{\mu})$ with $\mathbf{P}(\boldsymbol{\mu}) \in \mathbb{R}^{N_h \times N_h}$ as fine component, s.t.

$$\mathbf{Q}^{-1}(\boldsymbol{\mu}) = \mathbf{P}^{-1}(\boldsymbol{\mu}) + \mathbf{Q}_N(\boldsymbol{\mu})\left(\mathbf{I}_{N_h} - \mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{P}^{-1}(\boldsymbol{\mu})\right)$$

4. Multi space reduced basis preconditioners (MSRB) [1]

! Warning: At iteration k of GMRES, $\mathbf{Q}^{-1}(\boldsymbol{\mu})$ is applied to the k-th Krylov basis $\mathbf{v}_k^n = \mathbf{v}_k^n(\boldsymbol{\mu})$, that is

$$\mathbf{Q}^{-1}(\boldsymbol{\mu})\mathbf{v}_k^n = \mathbf{P}^{-1}(\boldsymbol{\mu}) + \mathbf{Q}_N(\boldsymbol{\mu}) \underbrace{\left(\mathbf{I}_{N_h} - \mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{P}^{-1}(\boldsymbol{\mu})\right)\mathbf{v}_k^n}_{\mathbf{v}^n_{k+\frac{1}{2}}(\boldsymbol{\mu})}$$

⇒ the RB low-rank solver must be trained to accurately solve

$$\mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{y}_k^n(\boldsymbol{\mu}) = \mathbf{v}^n_{k+\frac{1}{2}}(\boldsymbol{\mu}), \qquad (2)$$

a NS system with solution $\mathbf{y}_k^n(\boldsymbol{\mu}) = [\mathbf{y}_{u,k}^n(\boldsymbol{\mu}), \mathbf{y}_{p,k}^n(\boldsymbol{\mu})]^T$.

The key idea: compute from (2) a set of snapshots $\{\mathbf{y}_k^n(\boldsymbol{\mu}_i)\}_{i=1,n=1}^{n_s,N_t}$ and build with POD a sequence of k-dependent RB spaces $k = 1, 2, \dots$ to approximate the velocity $\mathbf{y}_{u,k}^n(\boldsymbol{\mu})$ and pressure $\mathbf{y}_{p,k}^n(\boldsymbol{\mu})$

$$\mathbf{V}_{uk} = \text{POD}\left([\mathbf{y}_{u,k}^n(\boldsymbol{\mu}_i)]_{i=1,n=1}^{n_s,N_t}, \varepsilon^k_{\text{POD}}\right),$$
$$\mathbf{V}_{pk} = \text{POD}\left([\mathbf{y}_{p,k}^n(\boldsymbol{\mu}_i)]_{i=1,n=1}^{n_s,N_t}, \varepsilon^k_{\text{POD}}\right), \qquad (3)$$
$$\mathbf{V}_k = diag(\mathbf{V}_{uk}, \mathbf{V}_{pk}).$$

In $\mathbf{V}_k$ an accurate RB approximation to $\mathbf{y}_k(\boldsymbol{\mu})$ is found

⇒ **Multi space reduced basis (MSRB) preconditioner**.

Key fact: each RB space is constructed up to a tolerance $\varepsilon^k_{\text{POD}}$ and the error decreases of a factor $\varepsilon^k_{\text{POD}}$ at iteration k.

To each $\mathbf{V}_k$ we associate

$$\mathbf{N}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu}) = \mathbf{V}_k^T\mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{V}_k$$

$$= \begin{bmatrix} \frac{\alpha_1}{\Delta t}\mathbf{M}^{\vec{u}}_{N_k}(\boldsymbol{\mu}) + \mathbf{D}_{N_k}(\boldsymbol{\mu}) + \mathbf{C}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu}) & \mathbf{B}^T_{N_k}(\boldsymbol{\mu}) \\ \mathbf{B}_{N_k}(\boldsymbol{\mu}) & 0 \end{bmatrix},$$

$$\mathbf{Q}_{N_k}(\boldsymbol{\mu}) = \mathbf{V}_k\mathbf{N}^{-1}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{V}_k^T,$$

The multiplicative combination of $\mathbf{P}^{-1}(\boldsymbol{\mu})$ and $\mathbf{Q}_{N_k}(\boldsymbol{\mu})$ leads to the k-dependent operator

$$\mathbf{Q}_{\text{MSRB},k}(\boldsymbol{\mu}) = \mathbf{P}^{-1}(\boldsymbol{\mu}) + \mathbf{Q}_{N_k}(\boldsymbol{\mu})\left(\mathbf{I}_{N_h} - \mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{P}^{-1}(\boldsymbol{\mu})\right),$$

Key fact: $\mathbf{Q}_{\text{MSRB},k}(\boldsymbol{\mu})$ is iteration dependent ⇒ flexible GMRES.

Stability of the RB coarse operators: (3) does not guarantee $\mathbf{N}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})$ to be nonsingular ⇒ **velocity enrichment strategy**: to each $\mathbf{y}_{p,k}^n(\boldsymbol{\mu}_i)$ we associate the supremizer $\mathbf{y}_{t,k}^n(\boldsymbol{\mu}_i)$ s.t.

$$\mathbf{X}_{hu}(\boldsymbol{\mu}_i)\mathbf{y}_{t,k}^n(\boldsymbol{\mu}_i) = \mathbf{B}^T(\boldsymbol{\mu}_i)\mathbf{y}_{p,k}^n(\boldsymbol{\mu}_i),$$

with $\mathbf{X}_{hu}$ the H^1_0 scalar product matrix; then, we set

$$\mathbf{V}_{sk} = \text{POD}\left([\mathbf{y}_{t,k}^n(\boldsymbol{\mu}_i)]_{i=1,n=1}^{n_s,N_t}, \varepsilon^k_{\text{POD}}\right).$$

and use it to augment the velocity space

$$\mathbf{V}_{uk} \leftarrow \text{Gram-Schmidt}\left([\mathbf{V}_{uk}, \mathbf{V}_{sk}]\right),$$

Efficient assembly of the RB coarse operators: Matrix Discrete Empirical Interpolation Method (MDEIM) is used to efficiently assemble $\mathbf{N}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})$:

$$\mathbf{N}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu}) \approx \widetilde{\mathbf{N}}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})$$

$$= \begin{bmatrix} \frac{\alpha_1}{\Delta t}\widetilde{\mathbf{M}}^{\vec{u}}_{N_k}(\boldsymbol{\mu}) + \widetilde{\mathbf{D}}_{N_k}(\boldsymbol{\mu}) + \widetilde{\mathbf{C}}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu}) & \widetilde{\mathbf{B}}^T_{N_k}(\boldsymbol{\mu}) \\ \widetilde{\mathbf{B}}_{N_k}(\boldsymbol{\mu}) & 0 \end{bmatrix},$$

by replacing the blocks with the MDEIM affinely approximated ones:

- MDEIM used in a standard way on the linear terms $\mathbf{M}^{\vec{u}}_{N_k}(\boldsymbol{\mu})$, $\mathbf{D}_{N_k}(\boldsymbol{\mu})$, and $\mathbf{B}_{N_k}(\boldsymbol{\mu})$;
- double-MDEIM strategy to affinely approximate $\mathbf{C}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})$:
  1. for any $\boldsymbol{\mu}_i$, build a *MDEIM in time* basis $\mathbf{C}(\mathbf{u}^{n,*}(\boldsymbol{\mu}_i), \boldsymbol{\mu}_i)$;
  2. gather the *MDEIM in time* bases to construct a final *MDEIM in parameter* and affinely approximate $\mathbf{C}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})$.
- approximated RB coarse component $\mathbf{Q}_{N_k}(\boldsymbol{\mu}) \approx \widetilde{\mathbf{Q}}_{N_k}(\boldsymbol{\mu})$, leading to an affinely approximated MSRB preconditioner

$$\widetilde{\mathbf{Q}}_{\text{MSRB},k}(\boldsymbol{\mu}) = \mathbf{P}^{-1}(\boldsymbol{\mu}) + \widetilde{\mathbf{Q}}_{N_k}(\boldsymbol{\mu})\left(\mathbf{I}_{N_h} - \mathbf{N}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu})\mathbf{P}^{-1}(\boldsymbol{\mu})\right).$$

Key fact: a coarser affine approximation than the one used for the standard RB method can be used without affecting the local accuracy and hence the overall convergence.

Nonsingularity of the preconditioner: for any $\boldsymbol{\mu} \in \mathcal{D}$, assume that $\mathbf{P}(\boldsymbol{\mu}) \in \mathbb{R}^{N_h \times N_h}$ is a nonsingular matrix and the matrices $\mathbf{V}_k^T\mathbf{P}(\boldsymbol{\mu})\mathbf{V}_k$, $\widetilde{\mathbf{N}}_{N_k}(\mathbf{u}^{n,*}(\boldsymbol{\mu});\boldsymbol{\mu}) \in \mathbb{R}^{N_k \times N_k}$ are invertible. Then the matrix $\mathbf{Q}_{\text{MSRB},k}(\boldsymbol{\mu})$ is nonsingular.

Dealing with the time dependence: algorithms are developed in a time slab framework, where $[0, T]$ is divided in time slabs and a MSRB preconditioner for each time slab is constructed.

5. Numerical results

- Results produced with *rb-LifeV*, the module of *LifeV* (www.lifev.org) forRB methods for parametrized problems.

- Computational resources have been provided by the Swiss National Supercomputing Center (www.cscs.ch), project ID s796.

Blood flow in parametrized carotid bifurcations:

Consider the NS system with final time $T = 0.64$s, viscosity $\nu = 0.035\text{cm}^2\text{s}^{-1}$ and $\boldsymbol{\mu} = (\mu_1, \mu_2) \in [0.2, 0.4] \times [0.85, 1]$:

- μ_1 narrows the bifurcation section by a displacement which extends harmonically a parametrized load $h(\boldsymbol{\mu}) = h(\mu_1)$ applied on the region A.
- μ_2 scales the inlet flowrate $Q_{\text{CCA}}(t;\boldsymbol{\mu}) = \mu_2 Q^0_{\text{CCA}}(t)$.
- A 5×4 a tensorial grid with is used train the RB coarse operators.
- SIMPLE is used as fine operator $\mathbf{P}(\boldsymbol{\mu})$ in MSRB preconditioner.
- 5 RB coarse operators with $N_k = 220$ basis function for velocity, pressure and supremizer (dimension $3N_k = 660$ in total).
- $\mathcal{P}_2 - \mathcal{P}_1$ FE for $\vec{u}(\boldsymbol{\mu})$ and $p(\boldsymbol{\mu})$ ($N_h = 259'930$).
- Simulation on 32 cores on Cray XC40 computing nodes.

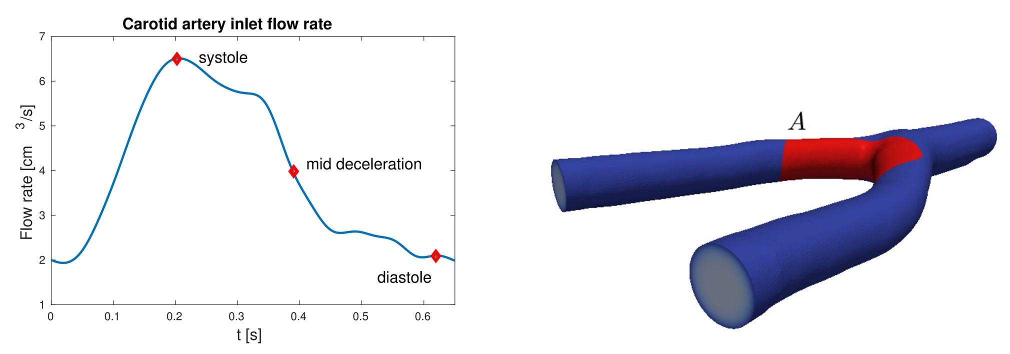


Figure 3: *Reference flow rate $Q^0_{\text{CCA}}(t)$ and application area A of $h(\boldsymbol{\mu})$.*

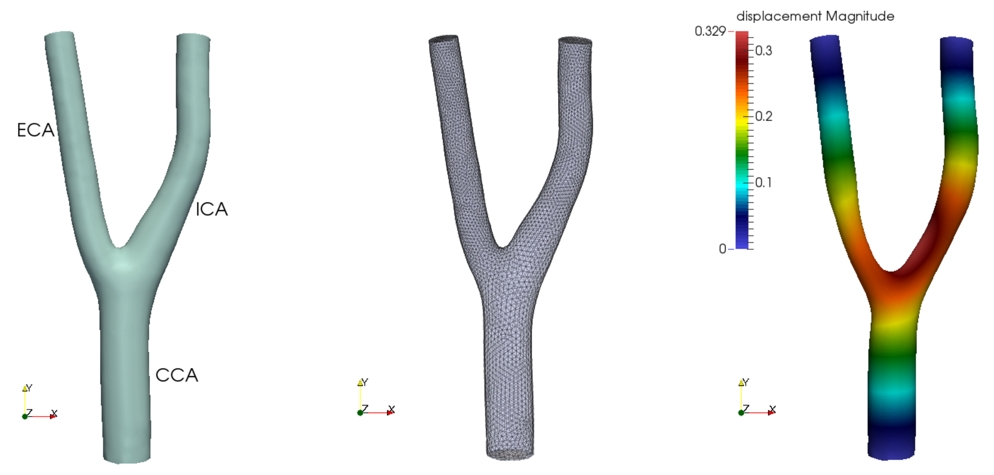


Figure 4: *Carotid bifurcation, reference mesh and displacement example.*

- Competitive computational time per timestep with MSRB preconditioner ($t^{\text{onl}}_{\text{MSRB}}$) compared to SIMPLE preconditioner (t_{SIMPLE}).
- The average iteration count per timestep significantly lower with MSRB preconditioner ($It^{\text{onl}}_{\text{MSRB}}$) than using only SIMPLE preconditioner (It_{SIMPLE}).

$t^{\text{onl}}_{\text{MSRB}}$	$It^{\text{onl}}_{\text{MSRB}}$	t_{SIMPLE}	It_{SIMPLE}	t_{off}	Speedup
6.33	**3**	55.06	91	98074.1	8.68

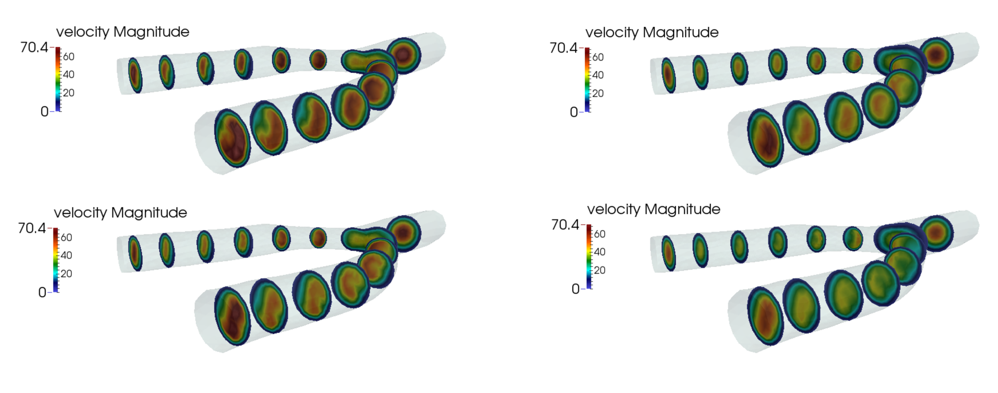


Figure 5: *Velocity magnitude for $\boldsymbol{\mu} = (0.375, 0.975)$ (left) $\boldsymbol{\mu} = (0.225, 0.875)$ (right) at $t = 0.2$ (top) and $t = 0.4$ (bottom).*

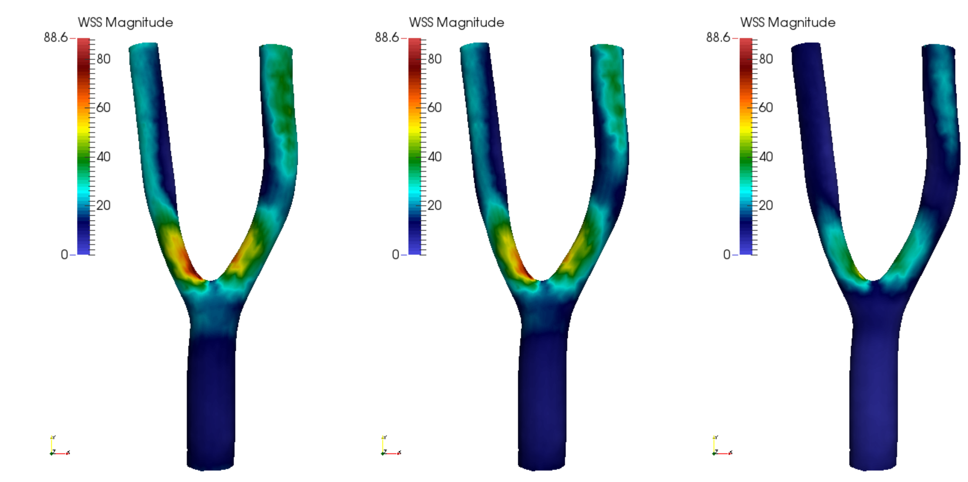


Figure 6: *Wall shear stress at times $t = 0.2, 0.4, 0.6$ for $\boldsymbol{\mu} = (0.375, 0.975)$.*

References

[1] N. Dal Santo, S. Deparis, A. Manzoni, and A. Quarteroni. Multi space reduced basis preconditioners for large-scale parametrized PDEs. SIAM Journal on Scientific Computing, 40(2):A954–A983, 2018.

[2] N. Dal Santo, S. Deparis and A. Manzoni. A numerical investigation of multi space reduced basis preconditioners for parametrized elliptic advection-diffusion equations. Communication in Applied and Industrial Mathematics, 8(1):282-297.

[3] N. Dal Santo, S. Deparis, A. Manzoni, and A. Quarteroni. Multi space reduced basis preconditioners for parametrized Stokes equations. Mathicse report 03.2018.

[4] N. Dal Santo and A. Manzoni. Hyper-reduced order models for parametrized unsteady Navier-Stokes equations on domains with variable shape. In preparation.

[5] N. Dal Santo. Multi space reduced basis preconditioners parametrized PDEs. PhD thesis, EPFL, 2018.