# Preconditioners for the Incompressible Navier Stokes Equations

Paolo Crosetto, Simone Deparis, Gwenol Grandperrin, Alfio Quarteroni

*MATHICSE - Chair of Modelling and Scientific Computing (CMCS)*
*EPFL, Lausanne, Switzerland*

SIAM Conference on Computational Science & Engineering

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

February 25th, 2013

# Acknowledgements

CSCS
Swiss National Supercomputing Centre

HP2C High Performance and High Productivity Computing

Trilinos

Dr. E. Cyr
Dr. J. Gaidamour
Dr. M. Heroux
Dr. M. Hoemmen
Dr. J. Hu
Dr. A. Salinger
Dr. C. Siefert

EMORY UNIVERSITY

Prof. Michele Benzi

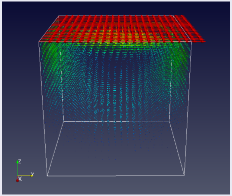
LifeV
http://www.lifev.org

Davide Forti
Paolo Tricerri

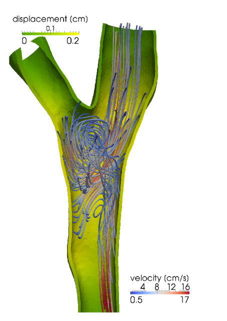
# Outline

- Metrics for parallel preconditioners
- Approximate preconditioners for the Navier–Stokes equations
- From Navier–Stokes to FSI preconditioners
- Experimental results
  - Strong scalability analysis for building the preconditioner and solving the Navier–Stokes equations
  - Analog results for FSI

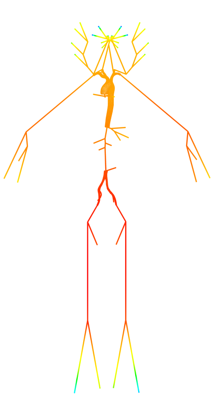
# Motivation



N–S




FSI



Multiscale model

Malossi, Blanco, Deparis, Quarteroni. Algorithms for the partitioned solution of weakly-coupled fluid models. 2010. Submitted.

Crosetto, Deparis, Fourestey, Quarteroni. Parallel algorithms for fluid-structure interaction problems in haemodynamics. *SIAM J. Sci. Comput., 2011.*

# Mathematical model

The Navier–Stokes equations for an incompressible viscous flow read:

$$
\begin{aligned}
\frac{\partial}{\partial t}\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} - \nu\Delta\mathbf{u} + \nabla p &= \mathbf{f} && \text{in } \Omega\times(0,T] \\
\nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega\times(0,T] \\
\mathbf{u} &= \varphi && \text{on } \Gamma_D\times(0,T] \\
\nu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n} &= 0 && \text{on } \Gamma_N\times(0,T] \\
\mathbf{u} &= \mathbf{u}_0 && \text{at } t=0
\end{aligned}
$$

where $\Gamma_D$ and $\Gamma_N$ are the Dirichlet and Neumann parts of the boundary respectively, $\mathbf{u}$ is the fluid velocity, $p$ the pressure, $\nu$ the kinematic viscosity of the fluid, and $\mathbf{f}$ the external forces.

# Mathematical model
## Discretization

Time discretization using e.g. semi-implicit Euler scheme:

$$\begin{aligned}
\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^{n+1} - \nu \Delta \mathbf{u}^{n+1} + \nabla p^{n+1} &= \mathbf{f}^{n+1} && \text{in } \Omega \\
\nabla \cdot \mathbf{u}^{n+1} &= 0 && \text{in } \Omega \\
\mathbf{u}^{n+1} &= \varphi && \text{on } \Gamma_D \\
\nu \frac{\partial \mathbf{u}^{n+1}}{\partial \mathbf{n}} - p^{n+1}\mathbf{n} &= 0 && \text{on } \Gamma_N
\end{aligned}$$

FE discretization using $\mathbb{P}_2 - \mathbb{P}_1$ finite elements on tetrahedral unstructured meshes:

$$\begin{pmatrix} F(\mathbf{U}^n) & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} \mathbf{U}^{n+1} \\ \mathbf{P}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{G}^{n+1}(\mathbf{U}^n) \\ \mathbf{0} \end{pmatrix}$$

# Metrics for parallel preconditioners

First perspective: To solve large scale problems as efficiently as possible by parallel algorithms.

**Definition (Strong scalability)**

Let $T_1$ and $T_P$ be the computational time needed to execute a task with fixed amount of computational work using one and $P$ processes respectively. An application is said to be strongly scalable if

$$T_P = \frac{T_1}{P}.$$

In particular, the preconditioned iterations of the numerical solver should be strongly scalable.

$\Rightarrow$ The preconditioner plays a key role in the scalability.

# Metrics for parallel preconditioners

Second perspective: solving bigger and bigger problems while keeping the computational time constant, provided that suitable resources are available.

**Definition (Weak scalability)**

Let $W_1$ and $W_2$ be the workload to solve a given problem using $P_1$ and $P_2$ processes respectively, such that

$$\frac{W_1}{P_1} = \frac{W_2}{P_2}.$$

An application is said to be weakly scalable if, for any couple $(W_1, P_1)$ and $(W_2, P_2)$, the computational time of the application is the same.

# Metrics for parallel preconditioners

## Definition (Preconditioner scalability)

A preconditioner $\mathcal{P}$ of $\mathcal{A}$ is said to be scalable if the rate of convergence of the iterative method used to solve the preconditioned system does not deteriorate when the number of processes grows.

## Definition (Preconditioner optimality)

A preconditioner is said to be optimal if for $\mathcal{A} \in \mathbb{R}^{N \times N}$

1. the number of preconditioned iterations to achieve a given error tolerance is bounded with respect to the dimension $N$ of $\mathcal{A}$;
2. the total computational costs to assemble and to use the preconditioner increase linearly with respect to the dimension $N$ of $\mathcal{A}$.

# Metrics for parallel preconditioners

**Definition (Preconditioner robustness)**

A preconditioner is said to be robust if the convergence rate of the iterative method does not depend on the physical parameters (e.g. viscosity) that characterize the PDE.

This property ensures that the preconditioner handles a wide range of Reynolds numbers; for medical simulations the Navier–Stokes equations have to be solved for a wide range of Reynolds from $\Re e = 0.003$ (capillary) to $\Re e = 4000$ (ascending aorta).

David N. Ku. Blood flow in arteries. *Annu. Rev. Fluid Mech., 1997.*

Formaggia, Quarteroni, Veneziani. Cardiovascular mathematics, volume 1 of *MS&A. Modeling, Simulation and Applications*. Springer-Verlag Italia, Milan, 2009.

# Designing a Navier–Stokes preconditioner for HPC

Dream list:

1. The algorithms involved to build and apply the preconditioner must be weakly and strongly scalable.
2. The preconditioner should be optimal.
3. The preconditioner should be scalable.
4. The preconditioner should be robust with respect to the viscosity $\nu$.

# Designing a Navier–Stokes preconditioner for HPC

The matrix of the linearized N–S system after discretization can be factorized as, e.g.

$$\mathcal{A} = \begin{pmatrix} F & B^T \\ B & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ BF^{-1} & I \end{pmatrix} \begin{pmatrix} F & B^T \\ 0 & -S \end{pmatrix}$$

where $S = BF^{-1}B^T$ is the Schur complement.

Idea: Exploit the block structure of the problem matrix:
We consider the following factor as right preconditioner

$$P = \begin{pmatrix} F & B^T \\ 0 & -S \end{pmatrix}$$

One can prove that GMRES converges in at most 2 iterations!

Murphy, Golub, Wathen. A note on preconditioning for indefinite linear systems. *SIAM J. Sci. Comput., 2000.*

Quarteroni, Saleri, Veneziani. Factorization methods for the numerical approximation of Navier–Stokes equations. *Comput. Methods Appl. Mech. Engrg., 2000.*

Elman, Howle, Shadid, Shuttleworth, Tuminaro. A taxonomy and comparison of parallel block multi-level preconditioners for the incompressible Navier–Stokes equations. *J. Comput. Phys., 227(3):1790–1808, 2008.*

# Classical preconditioners for N–S

- SIMPLE

$$P_{SIMPLE}^{-1} = \begin{pmatrix} D^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -B^T \\ 0 & I \end{pmatrix} \begin{pmatrix} D & 0 \\ 0 & \frac{1}{\alpha} I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -\tilde{S}^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -B & I \end{pmatrix} \begin{pmatrix} F^{-1} & 0 \\ 0 & I \end{pmatrix},$$

where $\alpha \in (0,1]$ is a damping parameter and $\tilde{S} = BD^{-1}B^T$.

Patankar, Spalding. A calculation procedure for heat, mass and momentum transfer in three dimensional parabolic flows. *International J. on Heat and Mass Transfer, 15:1787–1806, 1972.*

- Yosida

$$P_{Yosida}^{-1} = \begin{pmatrix} F^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -B^T \\ 0 & I \end{pmatrix} \begin{pmatrix} F & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -S^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -B & I \end{pmatrix} \begin{pmatrix} F^{-1} & 0 \\ 0 & I \end{pmatrix},$$

with $S = \Delta t B M_{\mathbf{u}}^{-1} B^T$.

Alfio Quarteroni, Fausto Saleri, and Alessandro Veneziani. Analysis of the Yosida method for the incompressible Navier–Stokes equations. *J. Math. Pures Appl., 1999.*

- PCD

$$P_{PCD}^{-1} = \begin{pmatrix} F^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -B^T \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -A_p^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & F_p \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & M_p^{-1} \end{pmatrix}.$$

Silvester, Elman, Kay, Wathen. Efficient preconditioning of the linearized Navier-Stokes equations for incompressible flow. *J. Comput. Appl. Math., 2001.*

Elman, Tuminaro. Boundary conditions in approximate commutator preconditioners for the Navier–Stokes equations.*Electron. Trans. Numer. Anal., 2009.*

# Approximate preconditioners for N–S

- Approximate SIMPLE (aSIMPLE)

$$P_{aSIMPLE}^{-1} = \begin{pmatrix} D^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -B^T \\ 0 & I \end{pmatrix} \begin{pmatrix} D & 0 \\ 0 & \frac{1}{\alpha} I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -\hat{\tilde{S}}^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -B & I \end{pmatrix} \begin{pmatrix} \hat{F}^{-1} & 0 \\ 0 & I \end{pmatrix},$$

where $\alpha \in (0,1]$ is a damping parameter and $\tilde{S} = BD^{-1}B^T$.

- Approximate Yosida (aYosida)

$$P_{aYosida}^{-1} = \begin{pmatrix} \hat{F}^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -B^T \\ 0 & I \end{pmatrix} \begin{pmatrix} F & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -\hat{S}^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -B & I \end{pmatrix} \begin{pmatrix} \hat{F}^{-1} & 0 \\ 0 & I \end{pmatrix},$$

with $S = \Delta t B M_{\mathbf{u},\ell}^{-1} B^T$.

- Approximate PCD (aPCD)

$$P_{aPCD}^{-1} = \begin{pmatrix} \hat{F}^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -B^T \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -\hat{A}_p^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & F_p \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & \hat{M}_p^{-1} \end{pmatrix}.$$

where ^ denotes the use of a preconditioner to approximate the inverse

Deparis, Grandperrin, Quarteroni, Approximate preconditioners for the Navier-Stokes equations in hemodynamic simulations, *Submitted, 2013.*

# Inverses approximation, our optimal choice

## Details on the preconditioners

- $\hat{F}^{-1}$ and $(BM_{\mathbf{u},\ell}^{-1}B^T)^{-1}$ are replaced with a 2-level Schwarz preconditioner; the first level is applied without overlap with a coarse grid correction. The subdomain problems are solved using exact factorization.
- $\hat{A}_p^{-1}$ and $(BD^{-1}B^T)^{-1}$ are replaced using a V-cycle AMG with 2 sweeps of symmetric Gauss-Seidel as smoother (presmoothing only), exact factorization for the coarsest level. The AMG is implemented in the `ML` package in `Trilinos`.

  Sala, An Object-Oriented Framework for the Development of Scalable Parallel Multilevel Preconditioners", *ACM Transactions on Mathematical Software, 2006.*

- $\hat{M}_p^{-1}$ is replaced by the inverse of the diagonal lumped mass matrix.

# Fluid-Structure Interaction (FSI)

## Coupled Problem

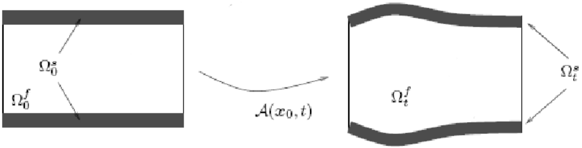



Figure: The ALE frame of reference

ALE map $\mathcal{A}_t : \Omega_o^{\mathrm{f}} \longrightarrow \Omega_t^{\mathrm{f}}$

Property of the ALE derivative:

$$\partial_t \mathbf{u_f}|_{\mathbf{x}_o}(\mathbf{x}, t) = \partial_t \mathbf{u_f}(\mathbf{x}, t) + (\mathbf{w}(\mathbf{x}, t) \cdot \nabla)\, \mathbf{u_f}(\mathbf{x}, t)$$

with $\mathbf{w}(\mathbf{x}) = \frac{d\mathcal{A}_t(\mathbf{x}_o)}{dt}$, $\mathbf{x} = \mathcal{A}_t(\mathbf{x}_o)$, the fluid domain velocity.

# The fluid and structure models

How to obtain $\mathbf{w}$ from a given vessel wall displacement $\mathbf{d}_s$:

**The Harmonic Extension equation for the fluid domain:**

$$
\begin{aligned}
-\Delta \mathbf{d}_f &= 0 && \text{in } \Omega_o^{\mathrm{f}} \\
\mathbf{d}_f &= \mathbf{d}_s && \text{on } \Gamma_o \\
\mathcal{A}_t(\mathbf{x}_o) &= \mathbf{x}_o + \mathbf{d}_f(\mathbf{x}_o, t) && \forall \mathbf{x}_o \in \Omega_o^{\mathrm{f}}
\end{aligned}
$$

**The Navier–Stokes equations in ALE form:**

$$
\begin{aligned}
\rho_f \partial_t \mathbf{u_f}|_{x_o} + \rho_f(\mathbf{u_f} - \mathbf{w}) \cdot \nabla \mathbf{u_f} - \nabla \cdot \boldsymbol{\sigma}_f &= \mathbf{f}_f && \text{in } \Omega_t^{\mathrm{f}} \\
\nabla \cdot \mathbf{u_f} &= 0 && \text{in } \Omega_t^{\mathrm{f}}
\end{aligned}
$$

where $\sigma_s = \frac{1}{J}\sigma_{os}\mathbf{F}^T$ is the Cauchy stress tensor.

**Lagrangian formulation for the structure**

$$
\rho_s \frac{\partial^2 \mathbf{d}_s}{\partial t^2} - \nabla_o \cdot \sigma_{os}(\mathbf{d}_s) = \rho_o \mathbf{f}_s, \quad \text{in } \Omega_o^{\mathrm{s}},
$$

where $\sigma_{os}$ is the first Piola-Kirchhoff stress tensor

# FSI problem: coupling conditions

- Continuity of stresses

$$\boldsymbol{\sigma}_{os} \cdot \mathbf{n}_o = J \boldsymbol{\sigma}_f \mathbf{F}^{-T} \circ \mathcal{A}_t \cdot \mathbf{n}_o = \text{ on } \Gamma_o$$

- Continuity of velocities

$$\mathbf{u}_f \circ \mathcal{A}_t = \frac{d\mathbf{d}_s}{dt} \text{ on } \Gamma_o$$

- Geometry adherence

$$\begin{aligned} \mathcal{A}_t(\mathbf{x}_o) = \mathbf{x}_o + \mathbf{d}_s(\mathbf{x}_o) \text{ on } \Gamma_o, \quad & \text{or} \\ \mathbf{d}_f = \mathbf{d}_s \quad & \text{on } \Gamma_o \end{aligned}$$

# Nonlinearities and discretizations

**Nonlinearity due to**

- convective term in Navier–Stokes equations;
- moving fluid integration domain.

**Time and space discretizations**

- fully implicit (FI)
- Galerkin Finite Element Method

| Fluid | Structure | ALE | (Harmonic Extension) |
|---|---|---|---|
| $\mathbb{P}_1$ bubble-$\mathbb{P}_1$ | $\mathbb{P}_1$ | $\mathbb{P}_1$ bubble | |

# FSI preconditioner

The fully coupled linearized FSI system $\mathcal{A}\mathbf{x} = \mathbf{b}$ can be written as a 4x4 block matrix with the blocks fluid, structure and geometry blocks on the diagonal:

$$A_{FSI} = \begin{pmatrix} \mathcal{F} & 0 & \mathcal{C}_1^T & \mathcal{D} \\ 0 & \mathcal{S} & \mathcal{C}_2^T & 0 \\ \mathcal{C}_1 & \mathcal{C}_2 & 0 & 0 \\ 0 & \mathcal{C}_3 & 0 & \mathcal{G} \end{pmatrix}.$$

We neglect partially the coupling in $A_{FSI}$ to form a preconditioner:

$$P_{FSI} = \begin{pmatrix} \mathcal{F} & 0 & \mathcal{C}_1^T & \mathcal{D} \\ 0 & \mathcal{S} & 0 & 0 \\ \mathcal{C}_1 & \mathcal{C}_2 & 0 & 0 \\ 0 & \mathcal{C}_3 & 0 & \mathcal{G} \end{pmatrix}.$$

Crosetto, Deparis, Fourestey, Quarteroni. Parallel algorithms for fluid-structure interaction problems in haemodynamics. *SIAM J. Sci. Comput., 2011.*

# FSI preconditioner

To apply $P_{FSI}^{-1}$, we use the factorization

$$P_{FSI} = \underbrace{\begin{pmatrix} \mathcal{I} & 0 & 0 & 0 \\ 0 & \mathcal{S} & 0 & 0 \\ 0 & 0 & \mathcal{I} & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix}}_{P_{\mathcal{S}}} \begin{pmatrix} \mathcal{I} & 0 & 0 & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ 0 & 0 & \mathcal{I} & 0 \\ 0 & \mathcal{C}_3 & 0 & \mathcal{I} \end{pmatrix} \underbrace{\begin{pmatrix} \mathcal{I} & 0 & 0 & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ 0 & 0 & \mathcal{I} & 0 \\ 0 & 0 & 0 & \mathcal{G} \end{pmatrix}}_{P_{\mathcal{G}}}$$
$$\begin{pmatrix} \mathcal{I} & 0 & 0 & \mathcal{D} \\ 0 & \mathcal{I} & 0 & 0 \\ 0 & 0 & \mathcal{I} & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathcal{I} & 0 & 0 & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ 0 & \mathcal{C}_2 & \mathcal{I} & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix} \underbrace{\begin{pmatrix} \mathcal{F} & 0 & \mathcal{C}_1^T & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ \mathcal{C}_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix}}_{P_{\mathcal{F}}}.$$

The factorization shows that it is possible to tackle the structure and the geometry subproblem using specific preconditioners while in the case of the fluid matrix block some coupling block are still to be considered.

We now discuss some strategies to approximate the inverse of the fluid block $P_{\mathcal{F}}$.

# FSI–SIMPLE preconditioner

The SIMPLE preconditioner for the Navier–Stokes equations reads

$$P_{SIMPLE} = \underbrace{\begin{pmatrix} F & 0 \\ B & -B diag(F)^{-1} B^T \end{pmatrix}}_{P_{SIMPLE,1}} \underbrace{\begin{pmatrix} I & diag(F)^{-1} B^T \\ 0 & \alpha I \end{pmatrix}}_{P_{SIMPLE,2}},$$

where $\alpha \in (0,1]$ is a parameter that damps the pressure update.

We use SIMPLE to approximate the inverse of the fluid block $P_{\mathcal{F}}$. To make the computation cheaper, the inverse of the matrix block $\mathcal{F}$ is replaced by its diagonal:

$$\begin{pmatrix} \mathcal{F} & 0 & \mathcal{C}_1^T & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ \mathcal{C}_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix} \cong \begin{pmatrix} P_{SIMPLE,1} & 0 & 0 & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ \mathcal{C}_1 & 0 & -\mathcal{C}_1 diag(\mathcal{F})^{-1} \mathcal{C}_1^T & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix} \begin{pmatrix} P_{SIMPLE,2} & 0 & diag(\mathcal{F})^{-1} \mathcal{C}_1^T & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ 0 & 0 & \mathcal{I} & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix}.$$

# FSI–PCD preconditioner

The PCD preconditioner for the Navier–Stokes equations reads

$$P_{PCD} = \begin{pmatrix} F & B^T \\ 0 & -M_p F_p^{-1} A_p \end{pmatrix}.$$

We use the PCD preconditioner to approximate the inverse of the fluid block $P_{\mathcal{F}}$:

$$\begin{pmatrix} \mathcal{F} & 0 & \mathcal{C}_1^T & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ \mathcal{C}_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix} \cong \begin{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -M_p F_p A_p \end{pmatrix} & 0 & 0 & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ 0 & 0 & \mathcal{I} & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix} \begin{pmatrix} \begin{pmatrix} I & B^T \\ 0 & I \end{pmatrix} & 0 & 0 & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ 0 & 0 & \mathcal{I} & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix} \begin{pmatrix} \begin{pmatrix} F & 0 \\ 0 & I \end{pmatrix} & 0 & \mathcal{C}_1^T & 0 \\ 0 & \mathcal{I} & 0 & 0 \\ \mathcal{C}_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{I} \end{pmatrix}$$

# Numerical results
## Simulation protocol

- Linear problem solved at each timestep with preconditioned GMRES;
- Stopping criteria based on the residual scaled by the right hand side:

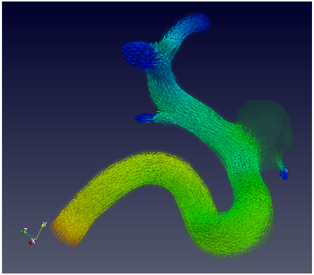
$$\|\mathbf{b} - \mathcal{A}\mathbf{x}_k\|_2 \leq 10^{-6}\|\mathbf{b}\|_2,$$

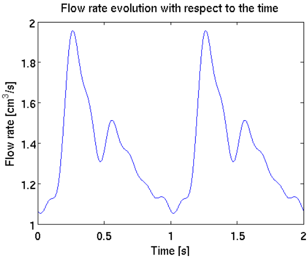
  where $\|\cdot\|_2$ denotes the $\ell_2$ norm of the vector of the nodal finite element solution.
- GMRES is never restarted.
- The simulations were carried out using `LifeV` (www.lifev.org) on the Monte Rosa Cray XE6 at the CSCS, Lugano, Switzerland.

| | |
|---|---|
| Number of nodes | 1496 |
| Number of processors per node | 2x16-core AMD Interlagos |
| Processors frequency | 2.1 GHz |
| Processors shared memory | 32 GB DDR3 |
| Peak performance | 402 Teraflop/s. |
| Network | Gemini 3D torus |

# Blood-flow in rigid geometry (N–S)

All our preconditioners are tested and tuned on a benchmark relevant for medical applications ($\Re e = 400$).

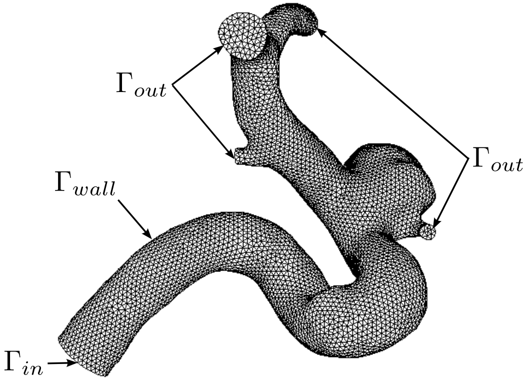





| Mesh | Velocity DoFs | Pressure DoFs | $h_{min}$ | $h_{av}$ | $h_{max}$ |
|---|---|---|---|---|---|
| Coarse | 597,093 | 27,242 | 0.015 | 0.035 | 0.059 |
| Medium | 4,557,963 | 199,031 | 0.005 | 0.018 | 0.051 |
| Fine | 35,604,675 | 1,519,321 | 0.0026 | 0.0097 | 0.0277 |

Baek, Jayaraman, Richardson, Karniadakis. Flow instability and wall shear stress variation in intracranial aneurysms. *J R Soc Interface*, 2010.

# Blood-flow in rigid geometry (N–S)

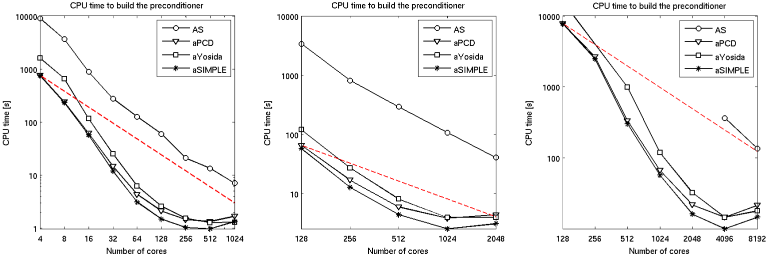



$$
\begin{aligned}
\mathbf{u} &= 0 && \text{on } \Gamma_{wall}, \\
\mathbf{u} &= \varphi_{flux}\mathbf{n} && \text{on } \Gamma_{in}, \\
\nu\frac{\partial \mathbf{u}}{\partial \mathbf{n}} - p\mathbf{n} &= 0 && \text{on } \Gamma_{out},
\end{aligned}
$$

# Blood-flow in rigid geometry (N–S)

## Preconditioner build

Coarse max. 1024 CPU | Medium max. 2048 CPU | Fine max. 8192 CPU

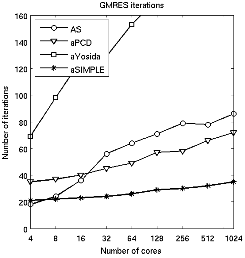



- The curves are superlinear due to the computation of the local LU factorizations.
- When the assembly time goes below a given threshold, the communication time overcomes the computation time for aPCD, aSIMPLE, and aYosida.
- The AS preconditioner is clearly longer to build.

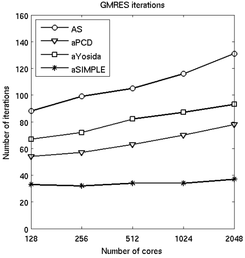
# Blood-flow in rigid geometry (N–S)

## GMRES iterations

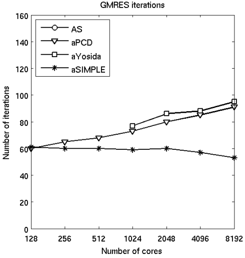
Coarse
max. 1024 CPU




Medium
max. 2048 CPU




Fine
max. 8192 CPU

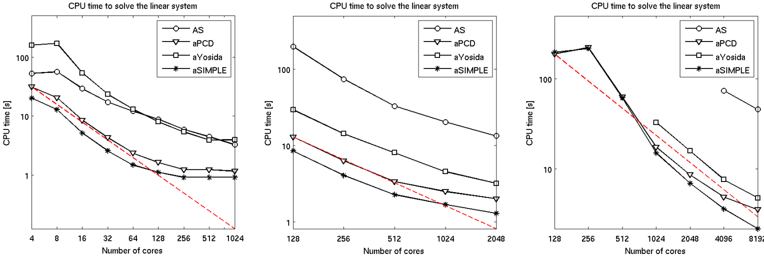



- aSIMPLE is scalable (flat curves).
- With aPCD, the iterations count is moderately increasing
- Problems of convergence are encounter with coarse mesh and aYosida.
- GMRES converges slower when the AS preconditioner is used.

# Blood-flow in rigid geometry (N–S)

## Time to solve the linear system

| Coarse | Medium | Fine |
|---|---|---|
| max. 1024 CPU | max. 2048 CPU | max. 8192 CPU |




- For the coarse mesh, the AS prec. is not strongly scalable.
- Under $\sim 1$ s. the communication time overcomes the computation time for aPCD, aSIMPLE, and aYosida (coarse mesh)
- For the medium and fine meshes, the preconditioners are strongly scalable.

# Greenshields-Weller benchmark (FSI)

## Established 3D FSI benchmark for hemodynamic

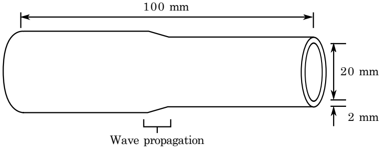



Figure: Domain

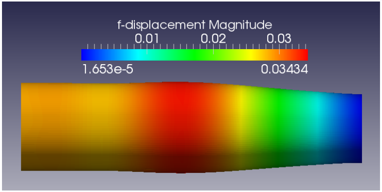



Figure: Displacement distribution after 10 ms.

3 discretizations of a 10x2 cm cylinder (∼300k-7M DoF).

Boundary conditions:

- pressure step function on the fluid inlet
- zero normal displacement on the inlet-outlet for both fluid and structure
- Neumann homogeneous everywhere else

Greenshields, Weller. A unified formulation for continuum mechanics applied to fluid-structure interaction in flexible tubes. *Internat. J. Numer. Methods Engrg., 2005.*

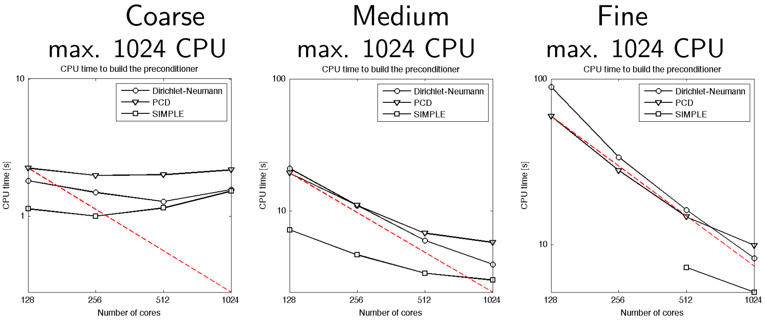
# Greenshields-Weller benchmark (FSI)

## Parameters of the simulation

- Comparison in terms of GMRES iterations, time to perform one GMRES iteration, and time spent to build the preconditioner.
- time discretization: second order discretization in time (BDF). $\Delta t = 10^{-3}$ s.
- space discretization: finite element $\mathbb{P}_1 Bubble - \mathbb{P}_1$ for the fluid, $\mathbb{P}_1$ for the structure, and $\mathbb{P}_1 Bubble$ for the harmonic extension equations.

# Greenshields-Weller benchmark (FSI)

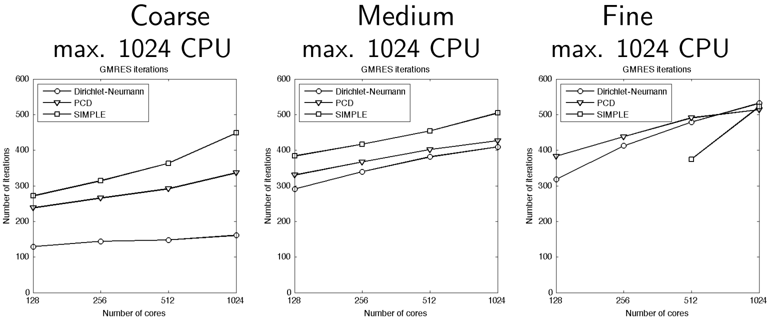
## Preconditioner build




- No strong scalability is obtained using the coarse mesh.
- When the accuracy of the mesh is increased, the preconditioners tends to be strongly scalable.

# Greenshields-Weller benchmark (FSI)

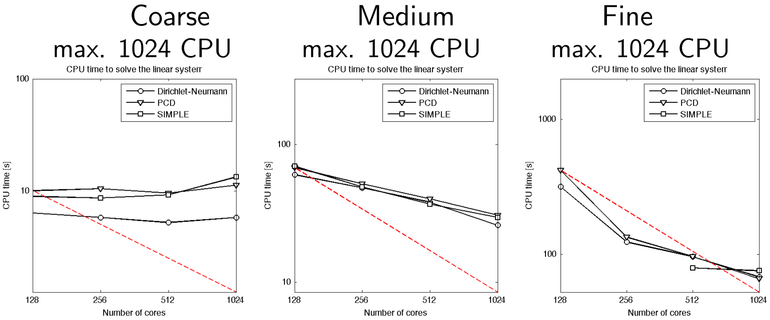
## GMRES iterations




- None of the preconditioners makes GMRES converge in a constant number of iterations.
- Between the two finer meshes, the number of iterations is of the same order.

# Greenshields-Weller benchmark (FSI)

## Time to solve the linear system




- No strong scalability is obtained using the coarse mesh.
- When the accuracy of the mesh is increased, the preconditioners tends to be strongly scalable.

# Conclusion and work in progress

Navier-Stokes preconditioners:

- We developed preconditioners for solving Hemodynamic simulations.
- We tested the weak and strong scalability of our algorithms.
- The proposed preconditioners are scalable (i.e. number of iterations remains constant wrt the number of processors).

FSI preconditioners:

- We tested of the Dirichlet–Neumann, FSI–SIMPLE, and FSI–PCD preconditioners on the Greenshields–Weller test case.

Future investigations:

- Investigate and optimize preconditioners for the structure and harmonic extension part of the FSI linear system.
- Consider test cases using a more realistic geometry, e.g. a femoropopliteal bypass geometry or an aorta geometry.