# A hierarchical iterative solver and Fractional timestepping schemes for the Navier-Stokes equations

Driss Yakoubi

joint works with:

J. Deteix and A. Fortin

GIREF, Universite Laval

2012 CMS, Winter Meeting

December 7-10

Montreal

# Outline

- Introduction
- Saddle Point approach
- Projection method
- Numerical results
- Conclusion

# Introduction

We will consider only flows in laminar regimes (no turbulence models)

$$\left\{\begin{array}{rcll} \rho \dfrac{\partial \boldsymbol{u}}{\partial t} + \rho\, \boldsymbol{u} \cdot \nabla\, \boldsymbol{u} - \nabla \cdot \sigma\,(\boldsymbol{u}, p) & = & \boldsymbol{f}, & \text{in } \Omega,\ t > 0, \\ \nabla \cdot \boldsymbol{u} & = & 0, & \text{in } \Omega,\ t > 0, \\ \boldsymbol{u} & = & \boldsymbol{g}_0, & \text{on } \Gamma_D,\ t > 0, \\ \sigma \cdot \boldsymbol{n} & = & \pi_0 \cdot \boldsymbol{n}, & \text{on } \Gamma_N,\ t > 0, \\ \boldsymbol{u} & = & \boldsymbol{u}_0, & \text{in } \Omega,\ t = 0, \end{array}\right.$$

$$\begin{array}{lrcl} \text{with} & \sigma\,(\boldsymbol{u}, p) & = & 2\mu\, \varepsilon\,(\boldsymbol{u}) - p\, I, \\ \text{and} & \varepsilon\,(\boldsymbol{u}) & = & \dfrac{1}{2}\,(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^t) \end{array}$$

Let us define the following functional spaces:

$$\mathbf{V} = H^1(\Omega)^d, \qquad \mathbf{V_0} = \{\boldsymbol{v} \in \mathbf{V},\ \boldsymbol{v} = 0 \quad on \quad \Gamma_D\} \quad \text{and} \quad M = L^2(\Omega)$$

Remark: if $\Gamma_D = \partial\Omega$, the pressure is defined only up to a constant.

Weak solutions exists for all time ([Leray '34]), (Hopf, '51]).

Uniqueness is an open issue in 3D

# Finite Element approximation: continuous in time

Find $(\boldsymbol{u}_h(t), p_h(t)) \in \mathbf{V}_h \times M_h, \quad \boldsymbol{u}_h(0) = \boldsymbol{u}_{h,0}, \ \boldsymbol{u}_h(t) = \boldsymbol{f}_h(t) \ on \ \Gamma_D,$

$$\int_\Omega \rho \left( \frac{\partial \boldsymbol{u}_h}{\partial t} + \boldsymbol{u}_h \cdot \nabla \boldsymbol{u}_h \right) \cdot \boldsymbol{v}_h + \int_\Omega 2\,\mu\,\varepsilon(\boldsymbol{u}_h) : \varepsilon(\boldsymbol{v}_h) - \int_\Omega p_h \nabla \cdot \boldsymbol{v}_h = \int_\Omega \boldsymbol{f}_h \cdot \boldsymbol{v}_h + \int_{\Gamma_N} \boldsymbol{g} \cdot \boldsymbol{v}_h$$

$$\int_\Omega q_h \nabla \cdot \boldsymbol{u}_h = 0, \qquad \forall (\boldsymbol{v}_h, q_h) \in \mathbf{V_h} \times M_h$$

The discrete spaces $\mathbf{V_h}$ and $M_h$ are chosen as follows:

$$\mathbf{V_h} = \left\{ \boldsymbol{v}_h \in \mathcal{C}^0(\Omega)^d, \ \boldsymbol{v}_h|_T \in \mathbb{P}_2^d \ \forall T \in \mathcal{T}_h \right\} \quad \text{and} \quad M_h = \left\{ q_h \in \mathcal{C}^0(\Omega), \ q_h|_T \in \mathbb{P}_1 \ \forall T \in \mathcal{T}_h \right\}$$

Then, the velocity and pressure can be decomposed as follows:

$$\boldsymbol{u}_h = \sum_{i=1}^{n} u_i\,\phi_i \quad \text{and} \quad p_h = \sum_{i=1}^{n} p_i\,\psi_i$$

Furthemore, the hierarchical basis allows the follows decompozition:

$$\mathbf{V_h} = \mathbf{V_l} \oplus \mathbf{V_q}, \qquad (\boldsymbol{u}_\mathbf{h} = \boldsymbol{u}_\mathbf{l} + \boldsymbol{u}_\mathbf{q})$$

$\mathbf{V_l}$ is the subspace of continuous piecewise linear polynomials, and

$\mathbf{V_q}$ is the comlementary subspace of continuous piecewise quadratic .

# Algebraic System

Replacing in the Galerkin formulation the expansion of $\boldsymbol{u}_h$ and $p_h$ on the F.E basis testing with shape functions.

We are led to the following system of ODEs:

$$\left\{ \begin{array}{rcl} M\dfrac{d\boldsymbol{U}}{dt} + D\,\boldsymbol{U} + N(\boldsymbol{U})\boldsymbol{U} + B^T\boldsymbol{P} & = & \boldsymbol{F}_{\boldsymbol{u}}(\boldsymbol{U}), \\ B\boldsymbol{U} & = & \boldsymbol{0}, \quad t > 0 \end{array} \right.$$

where:

$$\left\{ \begin{array}{rcl} M_{ij} & = & \displaystyle\int_\Omega \rho\,\varphi_j\,\varphi_i \ \text{ mass matrix}, \\ D_{ij} & = & \displaystyle\int_\Omega 2\mu\,\varepsilon(\varphi_j)\,\varepsilon(\varphi_i) \ \text{ stiffness matrix}, \\ N(\boldsymbol{U})_{ij} & = & \displaystyle\int_\Omega \rho\,\boldsymbol{u}_h\cdot\nabla\varphi_j\cdot\varphi_i, \ \text{ non linear term}, \\ B_{li} & = & \displaystyle -\int_\Omega \psi_l\,\nabla\cdot\varphi_i, \ \text{ divergence matrix}. \end{array} \right.$$

# Algebraic System: Temporal discretization

## First order scheme

A simple integration scheme: Implicit Euler with semi-implicit (for instance) treatment of convective term:

$$\rho \frac{\boldsymbol{u}^n - \boldsymbol{u}^{n-1}}{\delta t} - \boldsymbol{u}^{n-1} \cdot \nabla \boldsymbol{u}^n - \nabla \cdot (2\,\mu\,\varepsilon(\boldsymbol{u}_n)) + \nabla p^n = \boldsymbol{f}^n, \quad \text{in } \Omega,\ n = 1, 2, \ldots$$

In algebraic form this leads to a linear system to solve at every time step, of the form

$$\begin{bmatrix} \frac{M}{\delta t} + A + \tilde{N}(\boldsymbol{U}^{n-1}) & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^n \\ \boldsymbol{P}^n \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_{\boldsymbol{u}}^n + \frac{M}{\delta t} \boldsymbol{U}^{n-1} \\ \mathbf{0} \end{bmatrix}$$

# Algebraic System: Temporal discretization

## Second order scheme

As an example of second order scheme we consider a second order Backward differentiation (BDF2), with semi-implicit treatment of the convective term:

Approximation for the time derivative:

$$\frac{\partial \boldsymbol{u}}{\partial t}\Big|_{t^n} \approx \frac{3\boldsymbol{u}^n - 4\boldsymbol{u}^{n-1} + \boldsymbol{u}^{n-2}}{2\delta t}$$

Extrapolation of convective field:

$$\boldsymbol{u} \cdot \nabla \boldsymbol{u}|_{t^n} \approx \left(2\boldsymbol{u}^{n-1} - \boldsymbol{u}^{n-2}\right) \cdot \nabla \boldsymbol{u}^n$$

Resulting system:

$$\begin{bmatrix} \frac{3\,M}{2\delta t} + A + N(2\boldsymbol{U}^{n-1} - \boldsymbol{U}^{n-2}) & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^n \\ \boldsymbol{P}^n \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}^n_{\boldsymbol{u}} + \frac{M}{2\delta t}\left(4\boldsymbol{U}^{n-1} - \boldsymbol{U}^{n-2}\right) \\ \boldsymbol{0} \end{bmatrix}$$

# Solution of the linear system

After time discretization (Implicit Euler, BDF2,...) we are led to the linear system:

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^n \\ \boldsymbol{P}^n \end{bmatrix} = \begin{bmatrix} \boldsymbol{F_u} \\ \boldsymbol{0} \end{bmatrix}$$

with $A = \alpha M + D + N(\boldsymbol{U}^*)$

$\alpha, \boldsymbol{U}^*$ depending on the time marching scheme chosen

Solution by a direct solver is quite unfeasible. In 3D problems we have :
3 velocity components + the pressure -------> huge sparse linear system

Direct solver will fail because of memory requirements for fine meshes.

Iterative methods are suited, therefore.

# Solution of the linear system: Iterative Method

Our approach consists:

1. solve simultaneously the velocity and the pressure
2. GCR ou FGMRES (Krylov method)
3. Preconditioner:

$$\begin{bmatrix} A & B^t \\ B & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ B\,A^{-1} & I \end{bmatrix} \begin{bmatrix} A & 0 \\ B & -S \end{bmatrix} \begin{bmatrix} I & A^{-1}\,B^t \\ 0 & I \end{bmatrix},$$

where $S = B\,A^{-1}\,B^t$ is the Schur complement.

This factorization cannot be used as preconditioner because A is large-scale matrix and S is dense matrix.

# Solution of the linear system: Iterative Method

Then:

- we introduce an approximation $\tilde{A}$, $\tilde{S}$ of $A$ and $S$.
- And, we choose the follow preconditioner:

$$\mathcal{P}_R = \begin{bmatrix} \tilde{A} & B^T \\ 0 & -\tilde{S} \end{bmatrix}$$

- The action of this preconditioner on a residual vector can be rewritten as:

$$\left\{ \begin{array}{lcl} \delta p & = & -\tilde{S}^{-1}\, r_p \\ \delta \boldsymbol{u} & = & \tilde{A}^{-1}\,(r_{\boldsymbol{u}} - B^t\,\delta p)\,. \end{array} \right.$$

We have to solve 2 systems

# Iterative Method

# Preconditionning the matrix A

$$\delta \boldsymbol{u} = \tilde{A}^{-1} \left( r_{\boldsymbol{u}} - B^t \, \delta p \right) \quad \Leftrightarrow \quad A \, \delta \boldsymbol{u} = \tilde{r}_p .$$

We use the hierarchical basis for the quadratic FE discretization

Then, we can decompose the velocity into the linear part and a quadratic correction:

$$\boldsymbol{u}_h = \boldsymbol{u}_l + \boldsymbol{u}_q$$

Consequently, the matrix A can be rewritten as: $A = \begin{bmatrix} A_{ll} & A_{lq} \\ A_{ql} & A_{qq} \end{bmatrix}$

Finally, we use the following Algorithm proposed by El Maliki et al

1. Solve by few iterations of SOR: $A \, \delta = r$ where $\delta = (\delta_l \, , \, \delta_q)$ and $r = (r_l \, , \, r_q)^t$
2. Compute the residual: $d_l = r_l - A_{ll} \, \delta_l - A_{lq} \, \delta_q .$
3. Solve by a direct or few iterations of an iterative method: $A_{ll} \, \delta_l^* = d_l .$
4. Update the correction: $\delta = (\delta_l^* + \delta_l \, , \, \delta_q)^t$

# Iterative Method
# Schur complement approximation

Thanks to the discrete inf-sup condition proved by Brezzi-Fortin:

$$\beta^2 \le \frac{p^t \left(B\,D^{-1}\,B^t\right)}{p^t\,M_p\,p} \le \xi^2$$

we can remark that the matrix $B\,D^{-1}\,B^t$ is spectrally equivalent to the mass matrix $M_p$

Stokes problem:

We can chose this approximation

$$S \approx \tilde{S} = \frac{1}{\mu} M_p \approx \frac{1}{\mu} \mathrm{diag}(M_p)$$

Navier-Stokes problem:

Turek propose the additive preconditioner:

$$S^{-1} \approx M_p^{-1} \left(\alpha\, M_p + \mu\, D_p + \rho\, C_p\right)\, D_p^{-1}$$

Iterative Methods remain difficult and often need the addition of a stabilization term to the Navier Stokes equations [Olshanskii, Reusken'03]

$$\rho\,\frac{\partial \boldsymbol{u}}{\partial t} + \rho\,\boldsymbol{u}\cdot\nabla\,\boldsymbol{u} - \nabla\left(\xi\,\nabla\cdot\boldsymbol{u}\right) - \nabla\cdot\sigma\left(\boldsymbol{u},p\right) = \boldsymbol{f},$$

- How to choose parameter $\xi > 0$ ?

# Projection Methods: Chorin/Temam schemes

The idea of projection methods is to avoid the costly solution of the saddle point problem and to split the computation of velocity and pressure at each time step.

$$\rho \frac{\partial \boldsymbol{u}}{\partial t} + \underbrace{\mathcal{L}_1 \boldsymbol{u}}_{\text{viscous+transport terms}} + \underbrace{\mathcal{L}_2 \boldsymbol{u}}_{\text{incompress. constraint}} = \boldsymbol{f}$$

## Non-Incremental Chorin-Temam scheme

Simplest pressure-correction scheme: Chorin/Temam (1968-1969)

**Step I: Viscous prediction**

$$\left\{ \begin{array}{rcll} \frac{\rho}{\delta t}\left(\tilde{\boldsymbol{u}}^{k+1} - \boldsymbol{u}^{k}\right) - \nabla\cdot(\mu\,\varepsilon(\tilde{\boldsymbol{u}}^{k+1})) + \tilde{\boldsymbol{u}}^{k+1}\,\nabla\,\tilde{\boldsymbol{u}}^{k+1} & = & \boldsymbol{f}(t^{k+1}), & \text{in}\,\Omega \\ \tilde{\boldsymbol{u}}^{k+1} & = & \boldsymbol{g}_0, & \text{on}\,\Gamma_D \\ \mu\,\varepsilon(\tilde{\boldsymbol{u}}^{k+1})\cdot\boldsymbol{n} & = & \pi_0\cdot\boldsymbol{n}, & \text{on}\,\Gamma_N \end{array} \right.$$

**Step II: Projection**

$$\left\{ \begin{array}{rcll} \frac{\rho}{\delta t}\left(\boldsymbol{u}^{k+1} - \tilde{\boldsymbol{u}}^{k+1}\right) + \nabla\phi^{k+1} & = & 0, & \text{in}\,\Omega \\ \nabla\cdot\boldsymbol{u}^{k+1} & = & 0, & \text{in}\,\Omega \\ \boldsymbol{u}^{k+1}\cdot\boldsymbol{n} & = & g_0\cdot\boldsymbol{n}, & \text{on}\,\Gamma_D. \end{array} \right.$$

**Step III: Pressure correction**

$$p^{k+1} = \phi^{k+1}.$$

# Projection Methods
# Non-incremental pressure-correction schemes

## Implementation:

- Step 2 amounts to
$$\frac{\rho}{\delta t}\left(\boldsymbol{u}^{k+1} - \tilde{\boldsymbol{u}}^{k+1}\right) + \nabla\phi^{k+1} = 0,$$
with $\boldsymbol{u}^{k+1} \in H = \{\boldsymbol{v} \in L^2(\Omega)^d,\ \nabla\cdot\boldsymbol{v} = 0,\ \boldsymbol{v}|_{\Gamma_D} = g_0\}$

- Recalling $L^2(\Omega)^d = H \oplus \nabla\boldsymbol{H}^1(\Omega)$, this means
$$\boldsymbol{u}^{k+1} = P_H(\tilde{\boldsymbol{u}}^{k+1})$$

- Step 2 is a **projection** onto H.

I: $$-\Delta\,\phi^{k+1} = -\frac{\rho}{\delta t}\nabla\cdot\tilde{\boldsymbol{u}}^{k+1}, \qquad \nabla\phi^{k+1}\cdot\boldsymbol{n}|_{\Gamma_D} = 0, \quad \text{and} \quad \phi^{k+1}|_{\Gamma_N} = 0$$

II: $$\boldsymbol{u}^{k+1} = \tilde{\boldsymbol{u}}^{k+1} - \frac{\delta t}{\rho}\nabla\,\phi^{k+1}$$

*Non-incremental scheme: very simple and thus: very popular Alg.*

# Projection Methods
# Non-Incremental pressure-correction schemes

## Remarks:

- The boundary condition $\nabla p^{k+1} \cdot \boldsymbol{n}|_{\Gamma_D} = 0$ is ***enforced*** on the pressure
- Artificial Neumann BC $\Longrightarrow$ scheme *not fully first-order!*

Theorem: [Rannacher'91, Shen'92]

$$\|\boldsymbol{u} - \boldsymbol{u}_e\|_{H^1(\Omega)^d} + \|p - p_e\|_{L^2} \leq C\sqrt{\delta t}$$

$$\|\boldsymbol{u} - \boldsymbol{u}_e\|_{L^2(\Omega)^d} \leq C\,\delta t$$

The classical Chorin-Temam method has the inconvenience that it does not converge to the correct steady state solution. This comes from the fact that in step 1 the pressure does not appear at all!

# Projection Methods

## ~~Non-~~ Incremental pressure-correction schemes

*Easy remedy*: add old pressure in the first step

In the viscous step we add $\nabla p^k$ and we correct the pressure appropriately afterwards [Goda'79, Van Kan'86]:

Step I: Viscous prediction (BDF2)

$$\frac{\rho}{2\,\delta t}\left(3\,\tilde{\boldsymbol{u}}^{k+1} - 4\,\boldsymbol{u}^k + \boldsymbol{u}^{k-1}\right) - \nabla\cdot(\mu\,\varepsilon(\tilde{\boldsymbol{u}}^{k+1})) + \left(2\boldsymbol{u}^k - \boldsymbol{u}^{k-1}\right)\nabla\tilde{\boldsymbol{u}}^{k+1} + \nabla p^k = \boldsymbol{f}(t^{k+1}),$$

Step II: Projection

$$-\Delta\,\phi^{k+1} = -\frac{3\,\rho}{2\,\delta t}\nabla\cdot\tilde{\boldsymbol{u}}^{k+1},$$

Step III: Pressure correction

$$p^{k+1} = p^k + \phi^{k+1},$$

Finaly, update velocity:

$$\boldsymbol{u}^{k+1} = \tilde{\boldsymbol{u}}^{k+1} - \frac{2\,\delta t}{3\,\rho}\nabla\phi^{k+1}.$$

# Projection Methods
# Incremental pressure-correction schemes

## Results:

- Semi-discrete periodic chanel: [E-Liu'95]
- Semi-discrete : [Shen'96]
- Fully discrete: [Guermond'97,99], [Guermond-Quartapelle'98]

Theorem:

$$\begin{aligned} \|\boldsymbol{u} - \boldsymbol{u}_e\|_{H^1(\Omega)^d} + \|p - p_e\|_{L^2} &\leq C\,\delta t^1 \\ \|\boldsymbol{u} - \boldsymbol{u}_e\|_{L^2(\Omega)^d} &\leq C\,\delta t^2 \end{aligned}$$

- Again artificial BC: $\nabla p^{k+1} \cdot \boldsymbol{n} = \nabla p^k \cdot \boldsymbol{n} = \ldots . \nabla p^0 \cdot \boldsymbol{n} = 0.$
- Time stepping can be replaced by any 2nd order A-stable scheme

# Projection Methods

## Rotational incremental pressure-correction schemes

Use identity $\Delta \boldsymbol{u} = \nabla \nabla \cdot \boldsymbol{u} - \nabla \times \nabla \times \boldsymbol{u}$

This new idea was introduced first by Timmeramans, Minev and Van De Vosse'96.

Sum viscous prediction + projection, we obtain

$$\frac{\rho}{2\,\delta t}\left(3\,\boldsymbol{u}^{k+1} - 4\boldsymbol{u}^{k} + \boldsymbol{u}^{k-1}\right) + \mu\nabla\times\nabla\times\boldsymbol{u}^{k+1} + \nabla\left(p^{k} + \phi^{k+1} - 2\mu\nabla\cdot\tilde{\boldsymbol{u}}^{k+1}\right) = \boldsymbol{f}^{k+1}.$$

Step III: Pressure correction becomes $p^{k+1} = p^{k} + \phi^{k+1} - 2\mu\,\nabla\cdot\tilde{\boldsymbol{u}}^{k+1}$

- Why is it better? This implies consistent equations for the pressure:

$$\begin{cases} \Delta p^{k+1} = \nabla\cdot f^{k+1}, \\ \nabla p^{k+1}\cdot\boldsymbol{n} = \left(f^{k+1} - \mu\nabla\times\nabla\times\boldsymbol{u}^{k+1}\right)\cdot\boldsymbol{n} \end{cases}$$

- Where is the catch? The tangential component of $\boldsymbol{u}^{k+1}$ is still not correct! $\Longrightarrow$ sub-optimality

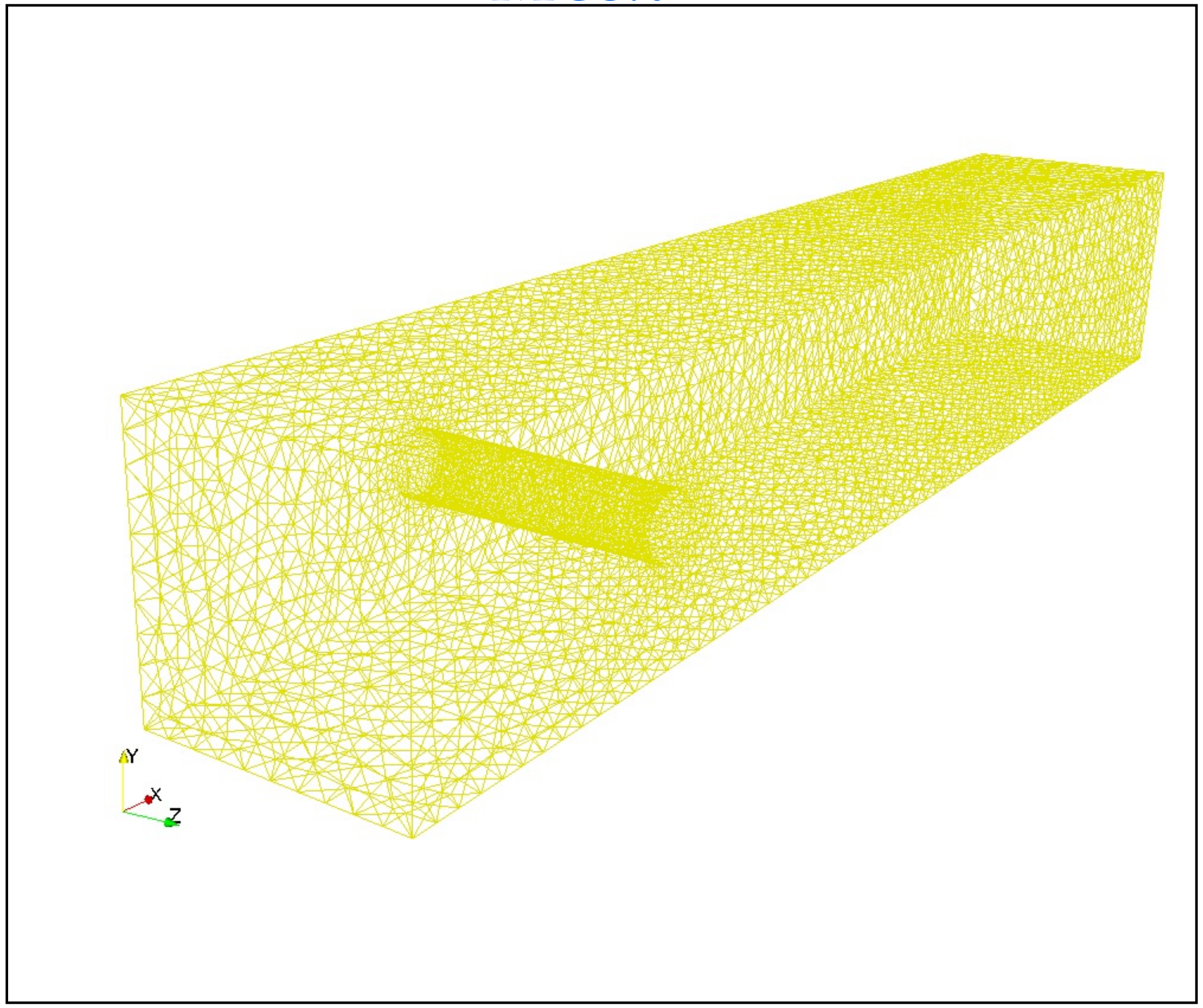
Theorem: [Guermond-Shen' 2006]

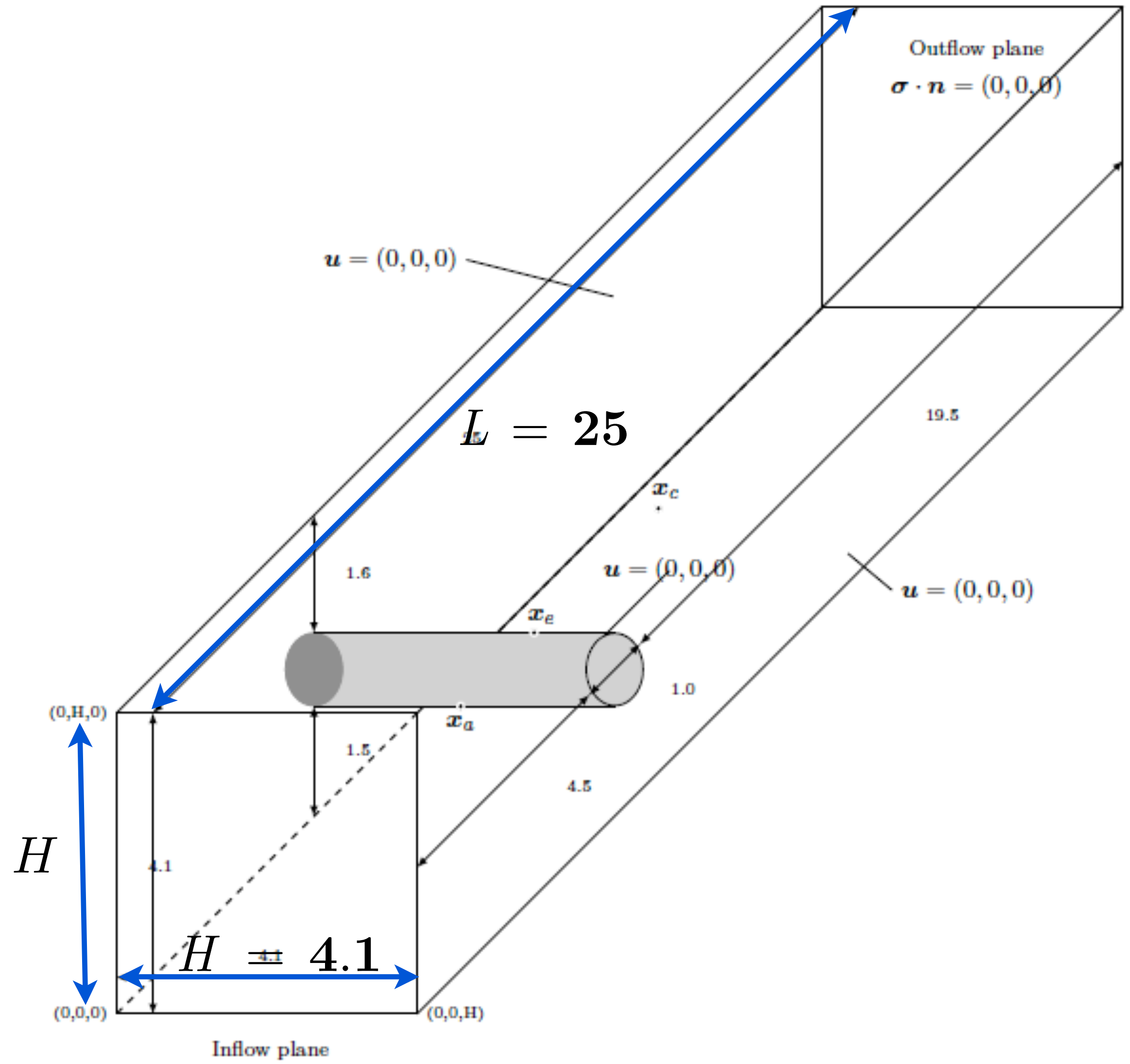
$$\|\boldsymbol{u} - \boldsymbol{u}_e\|_{H^1(\Omega)^d} + \|p - p_e\|_{L^2} \leq C\,\delta t^{\frac{3}{2}}$$

# Fluid flow around rigid objects

- ***Configuration and boundary conditions for flow around cylinder:***
  - The inflow condition is: $\boldsymbol{u}(0.y,z) \;=\; (u_x,0,0)) \quad \text{s.t} \quad u_x \;=\; \frac{72}{H^4}\,yz\,(H-y)(H-z))$
  - Reynolds number: $Re \;=\; 20$

***Mesh***

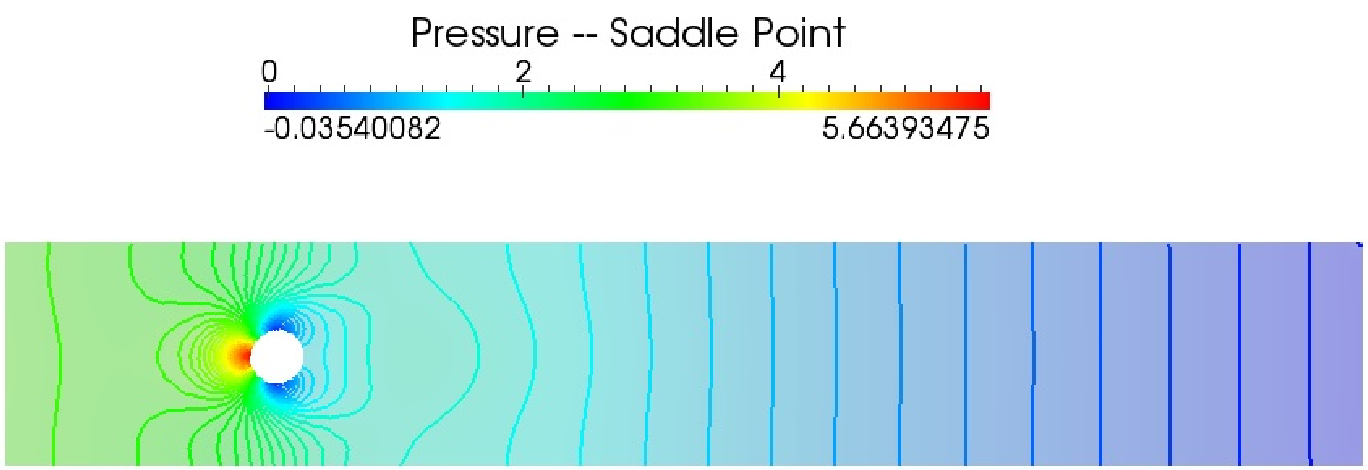


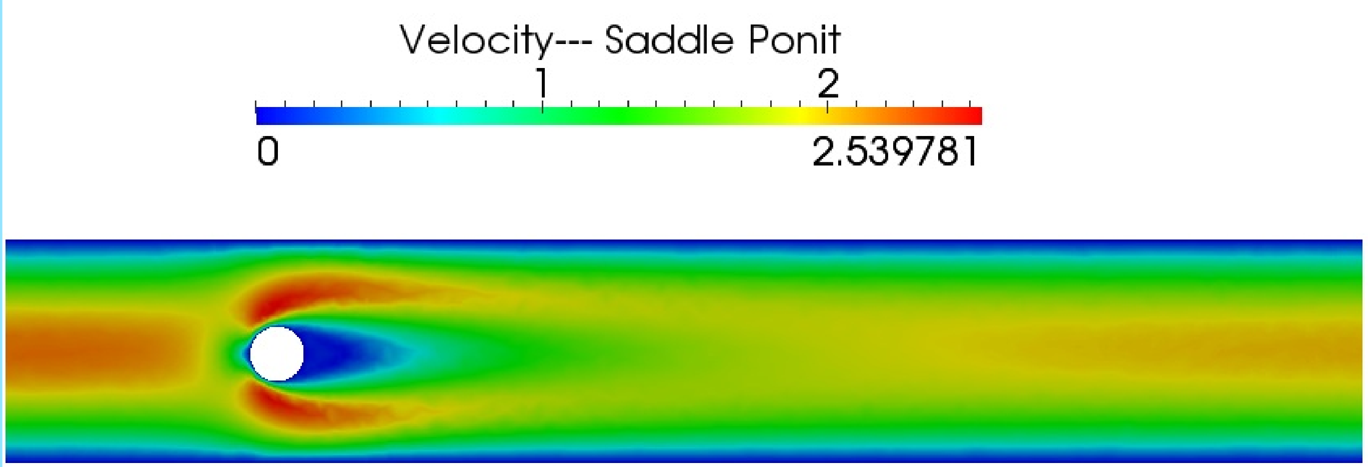



# Numerical simulations with MEF++

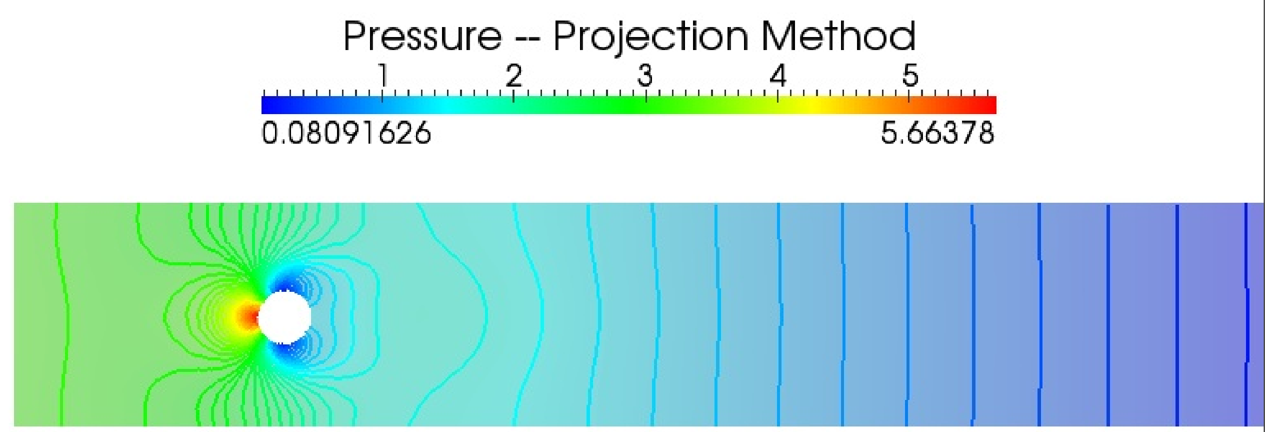
## Comparison: Projection / Saddle Point Methods

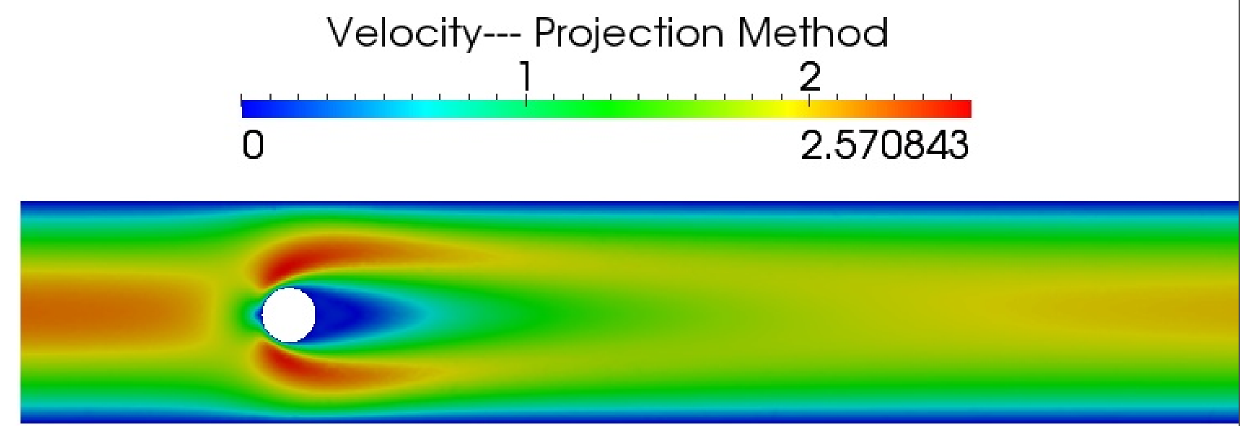
- ***Saddle Point Method:***




- ***Projection Method:***

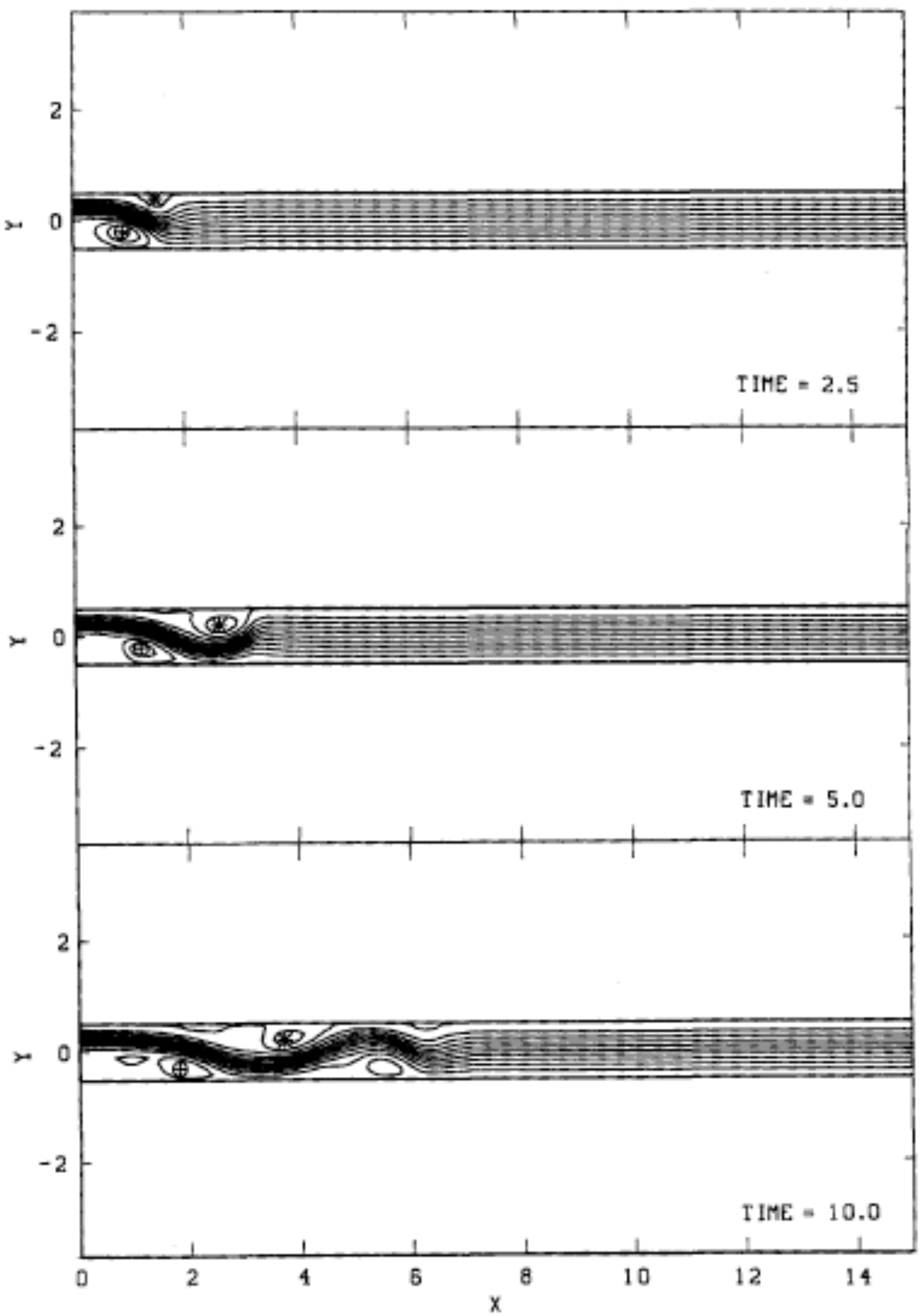


# Numerical simulations

The backward-facing step

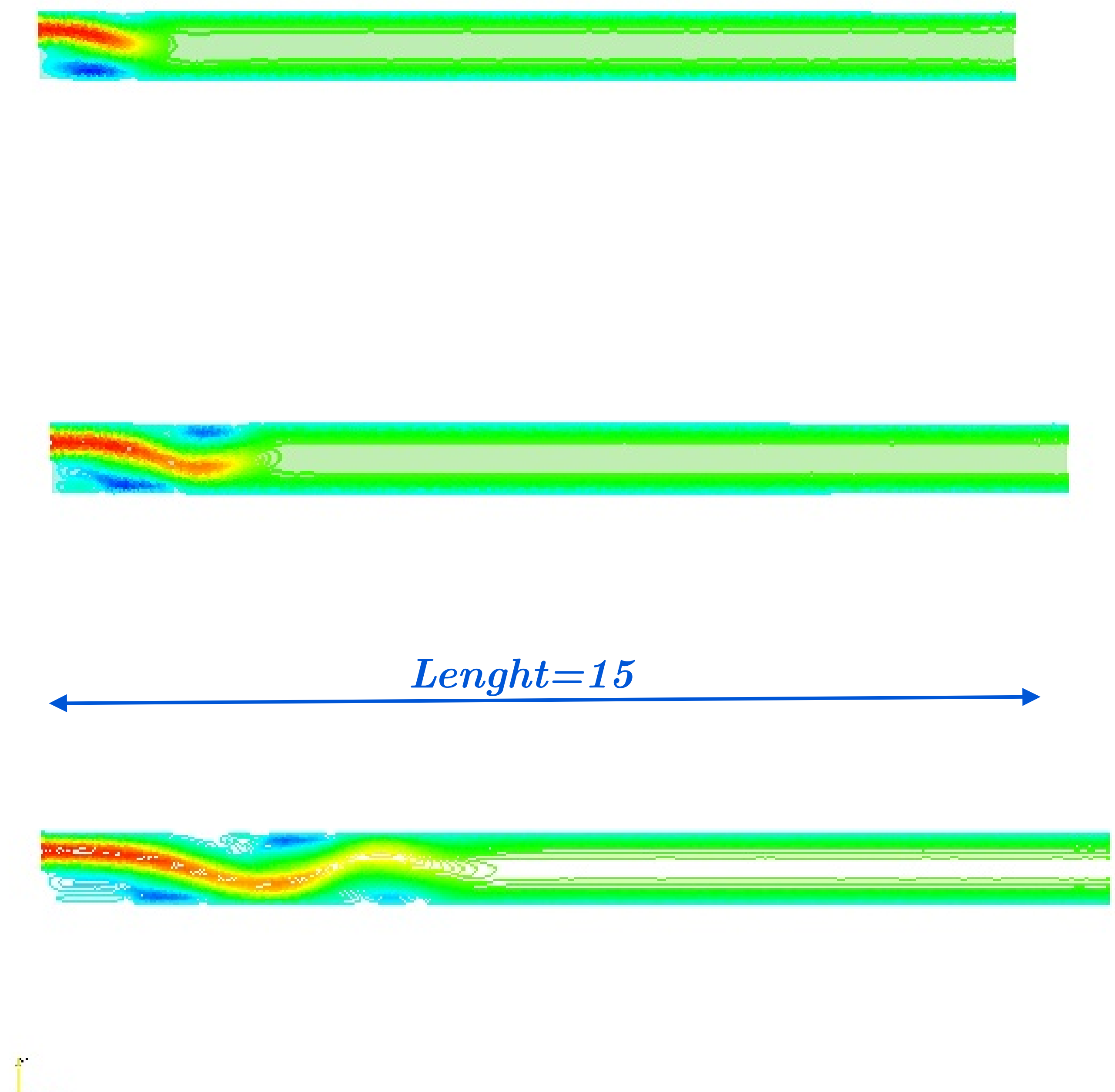
Comparison with Gresho results (Reynolds =800)




Time=2.5

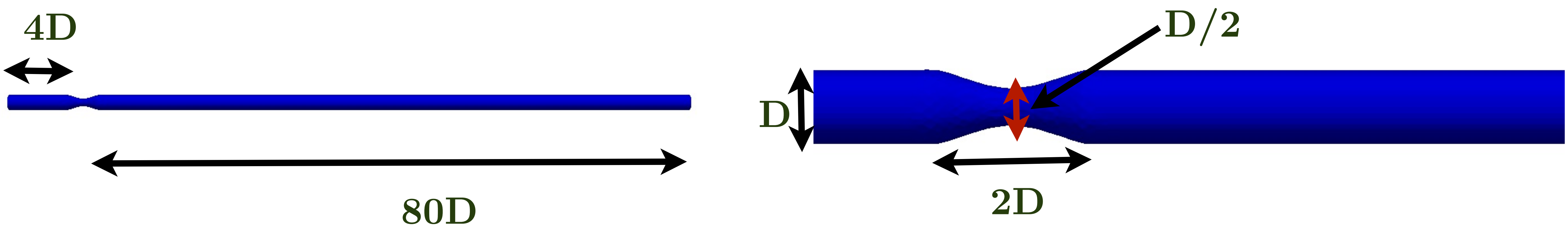
Time=5

Time=10



# Fluid flow in a sudden axisymmetric constriction

## Geometry




Physical experience:

Vetel, Garon:

*Symmetry breaking at*

*Reynolds number = **255***

Stability method:

S.J. Sherwin et al

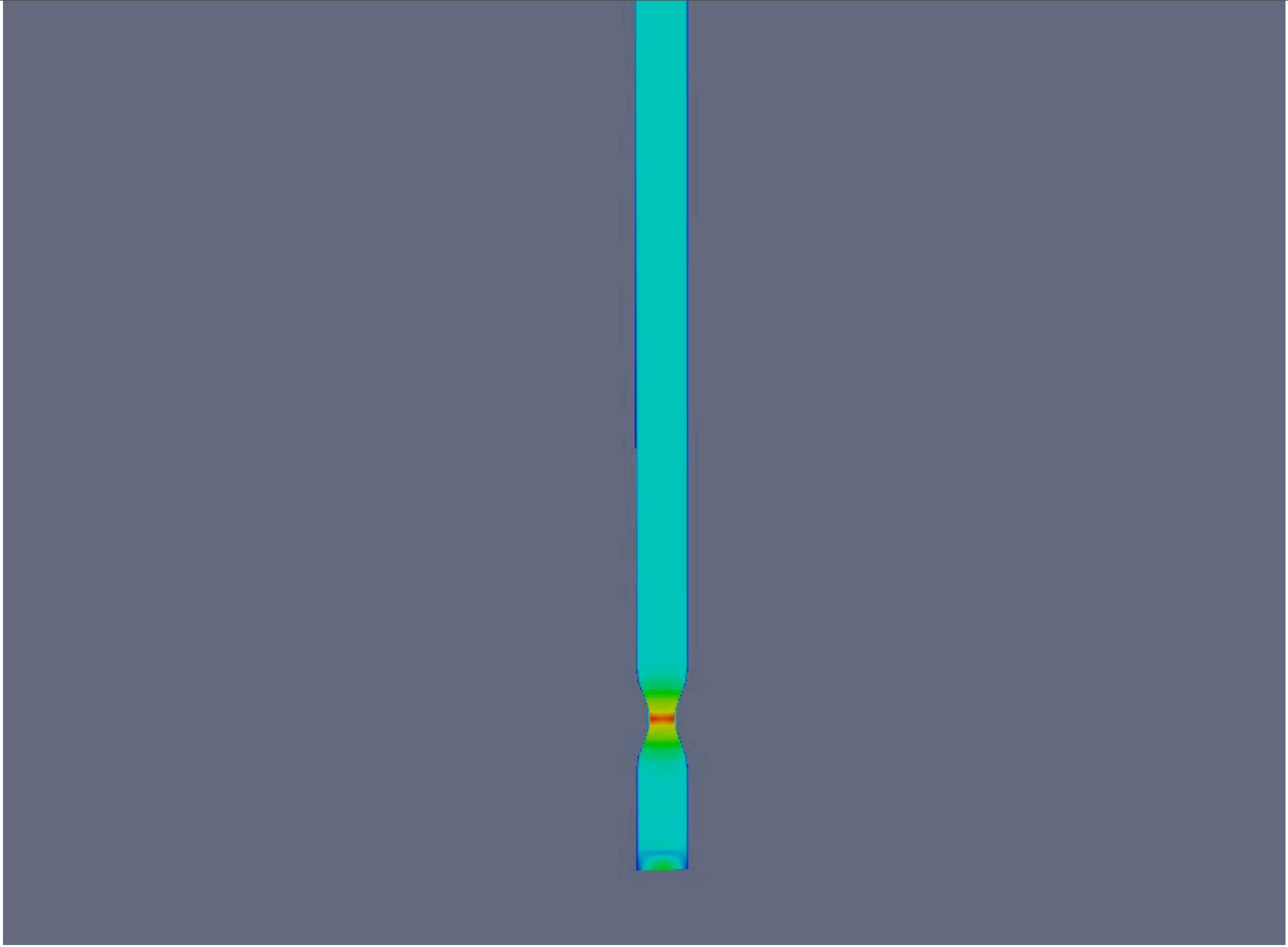
*Symmetry breaking at*

*Reynolds number = **722***

# Fluid flow in a sudden axisymmetric constriction

## Our strategy! (Reynolds number = 300)

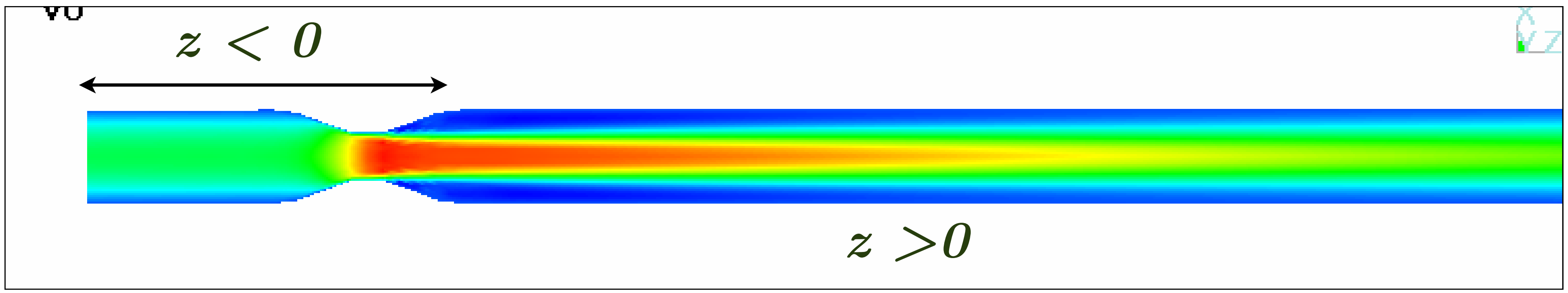
- We consider a symmetric Mesh
- By a Reynolds Continuation Procedure, we obtain a symetric solution
- Build a non-symmetric solution by imposing some boundary conditions
- Redo the simulation by setting this non-symmetric as initial data $u_0$



# Numerical simulations

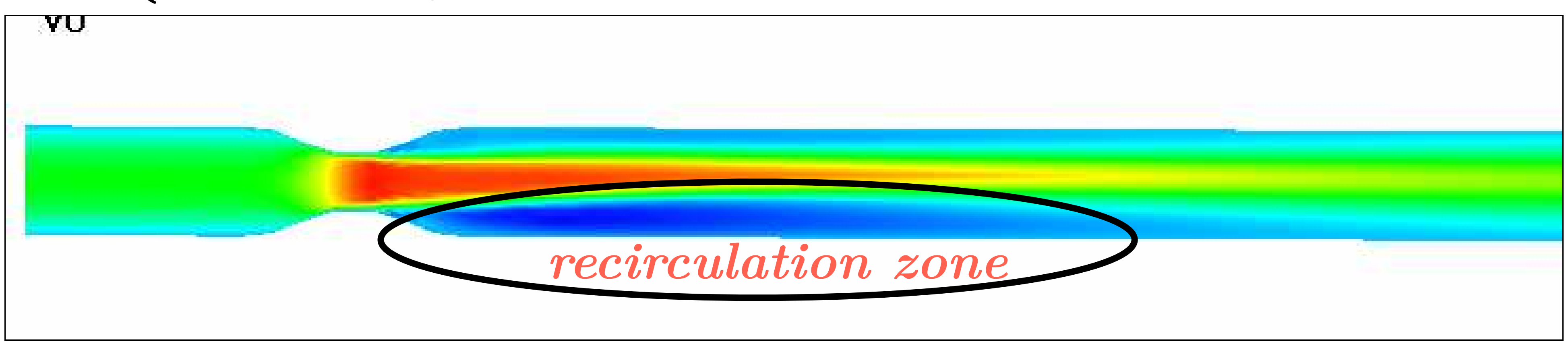
## Fluid flow in a sudden axisymmetric constriction

Symmetric solution at Reynolds = 300




Perturbation of the initial condition:

$$\begin{cases} u_x = 0,\ u_y = 0, & \text{on} \quad \{(x,y,z)\in\mathbb{R}^3 \text{ s.t } z>0,\ y>0\},\\ \boldsymbol{u}=(u_x,u_y,u_z)=\mathbf{0} & \text{on} \quad \{(x,y,z)\in\mathbb{R}^3 \text{ s.t } z\le 0,\ y\le 0\},\end{cases}$$



# Conclusion and ongoing works

## Conclusion

- Robust solver for Navier Stokes Equations

- Projection schemes is very simple and CPU:

  Chorin/Temam schemes or Saddle point ?

  ---> Chorin/Temam!

- We can simulate big problems...

## Ongoing works

- Verify experimental results of Vetel/Garon and Sherwin

- Fluid-Structure Interaction:

  Chorin-Temam scheme/Saddle Point...