Parallel Inexact Newton-Type Methods for the Solution of Steady Three Dimensional Incompressible Viscoplastic Flows with the SUPG/PSPG Finite Element Formulation

R. N. Elias,
A. L. G. A. Coutinho and
M. A. D. Martins

COPPE/UFRJ
Rio de Janeiro, RJ
July de 2004

Introduction

- Objectives;
- Applications;
- Stabilized Finite Element Method;
- Fluid Rheology;
- Parallel issues;
- Results.

NACAD

Objectives

- Incompressible fluid flow;
- Newtonian and Non-Newtonian fluids
- Large scale problems;

Applications

- Oil and gas industry:
  - Flow of clay;
  - Salt domes formation;
  - Flow of drilling mud in borehole annulus.

Main focus

- Other industries:
  - Injection and extrusion of polymers;
  - Food flow;
  - Pastes;
  - Blood flow into arteries;
  - Etc...

Governing Equations
(Steady State)

Momentum conservation

$$\rho(\mathbf{u}\cdot\nabla\mathbf{u}-\mathbf{f})-\nabla\cdot\boldsymbol{\sigma}(p,\mathbf{u})=0$$

$$\boldsymbol{\sigma}(p,\mathbf{u})=-p\mathbf{I}+\mathbf{T}$$

$$\mathbf{T}=2\mu(\dot{\gamma})\boldsymbol{\varepsilon}(\mathbf{u})$$

Newtonian: $\mu(\dot{\gamma})=\mu=const$

Power Law: $\mu(\dot{\gamma})=\begin{cases}\mu_0 k\dot{\gamma}^{n-1} & se\,\dot{\gamma}>\dot{\gamma}_0\\ \mu_0 k\dot{\gamma}_0^{n-1} & se\,\dot{\gamma}\le\dot{\gamma}_0\end{cases}$

Bingham: $\mu(\dot{\gamma})=\mu_0+\dfrac{\boldsymbol{\sigma}_Y}{\dot{\gamma}}$

Continuity:

$$\nabla\cdot\mathbf{u}=0$$

NACAD

Fluid Rheology

(Viscous models)

NEWTONIAN FLUIDS: → $\mu(\gamma) = \mu = constant$

NON-NEWTONIAN FLUIDS:

Power Law:

$$\mu(\gamma) = \mu_0 k \gamma^{n-1}$$

***Bingham*:**

$$\mu(\gamma) = \mu_0 + \frac{\sigma_Y}{\gamma}$$

Others:

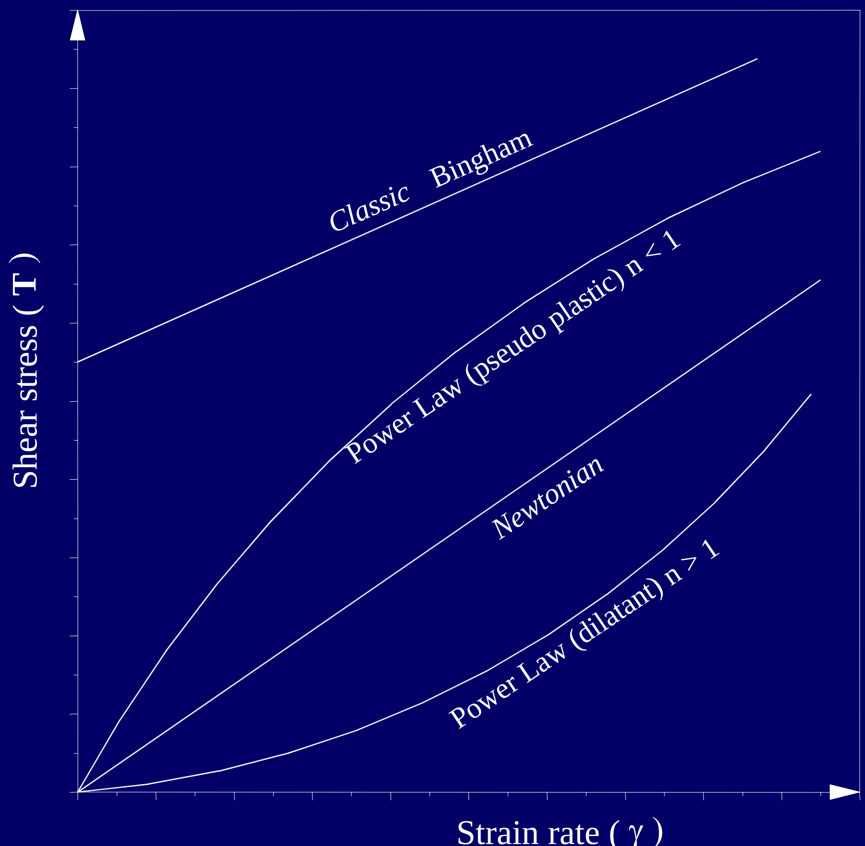
Oldroyd,

Carreau, etc...

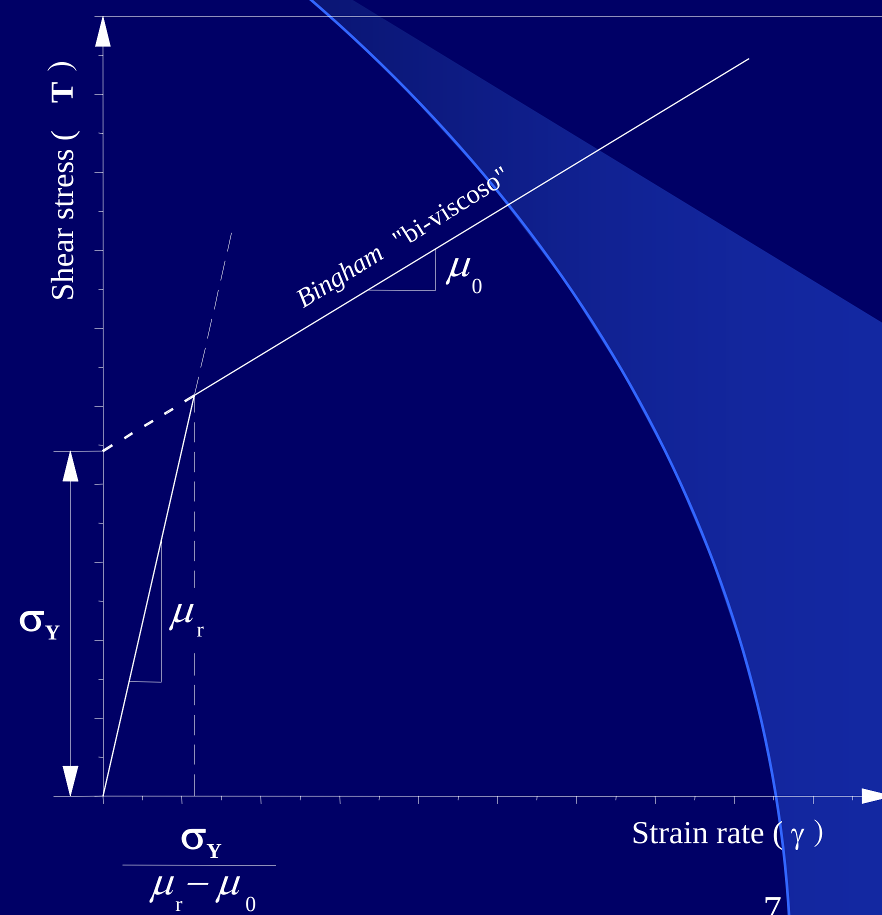
NACAD

Fluid Rheology

(Viscous models)



Bi-viscosity Bingham model



SUPG/PSPG Finite Element Formulation

(variational form of *Navier-Stokes*)

$$\int_\Omega \mathbf{w}^h \cdot \rho\left(\mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}\right) d\Omega + \int_\Omega \boldsymbol{\varepsilon}\left(\mathbf{w}^h\right) : \boldsymbol{\sigma}\left(p^h, \mathbf{u}^h\right) d\Omega - \int_\Gamma \mathbf{w}^h \cdot h \, d\Gamma + \int_\Omega q^h \nabla \cdot \mathbf{u}^h d\Omega$$

$$+ \sum_{e=1}^{nel} \int_\Omega \left(\tau_{\mathrm{SUPG}} \mathbf{u}^h \cdot \nabla \mathbf{w}^h\right) \cdot \left[\rho\left(\mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}\right) - \nabla \cdot \boldsymbol{\sigma}\left(p^h, \mathbf{u}^h\right)\right] d\Omega$$

$$+ \sum_{e=1}^{nel} \int_\Omega \left(\frac{1}{\rho} \tau_{\mathrm{PSPG}} \nabla q^h\right) \cdot \left[\rho\left(\mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}\right) - \nabla \cdot \boldsymbol{\sigma}\left(p^h, \mathbf{u}^h\right)\right] d\Omega = 0$$

SUPG = *Streamline Upwind Petrov/Galerkin*

PSPG = *Pressure Stabilizing Petrov/Galerkin*

(T. E. TEZDUYAR, 1992)

SUPG/PSPG Formulation

(Matrix form)

$$\begin{bmatrix} \mathbf{N}(\mathbf{u})+\mathbf{N}_{SUPG}(\mathbf{u})+(\mathbf{K}+\mathbf{K}_{SUPG}) & -(\mathbf{G}+\mathbf{G}_{SUPG}) \\ \mathbf{G}^T+\mathbf{N}_{PSPG}(\mathbf{u})+\mathbf{K}_{PSPG} & \mathbf{G}_{PSPG} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{f}+\mathbf{f}_{SUPG} \\ \mathbf{e}+\mathbf{e}_{PSPG} \end{bmatrix}$$

Rank problem solved with PSPG stabilization

Compact matrix form:

$$\mathbf{A}(\mathbf{x})\mathbf{x} = \mathbf{b}$$

Nonlinearity = Troubles!!!

Solution $\rightarrow$ non-linear methods.

Newton, Picard, *Broyden's*, etc...

Nonlinear Solution Procedures

Newton Method:

Considering the following non-linear problem:

$$\mathbf{N}(\mathbf{d}) \cong 0$$

Let ($\mathbf{d}^*$) be an approximation to the exact solution ($\mathbf{d}$), sucha that:

$$\mathbf{d} \cong \mathbf{d}^* + \Delta\mathbf{d}$$

Performing *Taylor* series expansion:

$$\mathbf{N}(\mathbf{d}) \cong \mathbf{N}(\mathbf{d}^*) + \frac{\partial \mathbf{N}}{\partial \mathbf{d}}(\mathbf{d}^*)\Delta\mathbf{d} + \ldots$$

Substituting some terms:

$$\underbrace{-\frac{\partial \mathbf{N}}{\partial \mathbf{d}}(\mathbf{d}^*)}_{\mathbf{K}_T}\Delta\mathbf{d} = \underbrace{\mathbf{N}(\mathbf{d}^*)}_{\mathbf{r}}$$

Picard Method:

Consider the folowing linearized system:

$$\mathbf{N}(\mathbf{d}_i)\mathbf{d}_{i+1} \cong 0$$

We have to solve:

$$\mathbf{N}(\mathbf{d}_i)\mathbf{d}^* \cong 0$$

And update:

$$\mathbf{d}_{i+1} \cong \alpha\mathbf{d}^i + (1-\alpha)\mathbf{d}^*$$

Where α is a relaxation or speedup parameter: $0 < \alpha < 1$

$\mathbf{K}_T$ = Jacobian matrix

$\mathbf{r}$ = nonlinear residual

"Classic" Newton Method

(Iterative procedure)

Initial "kick": $\mathbf{d}^0 = 0$

FOR i = 0 UNTIL CONVERGENCE DO

Solve linear system:

$$\mathbf{K}_T \Delta \mathbf{d}^i = \mathbf{r}$$

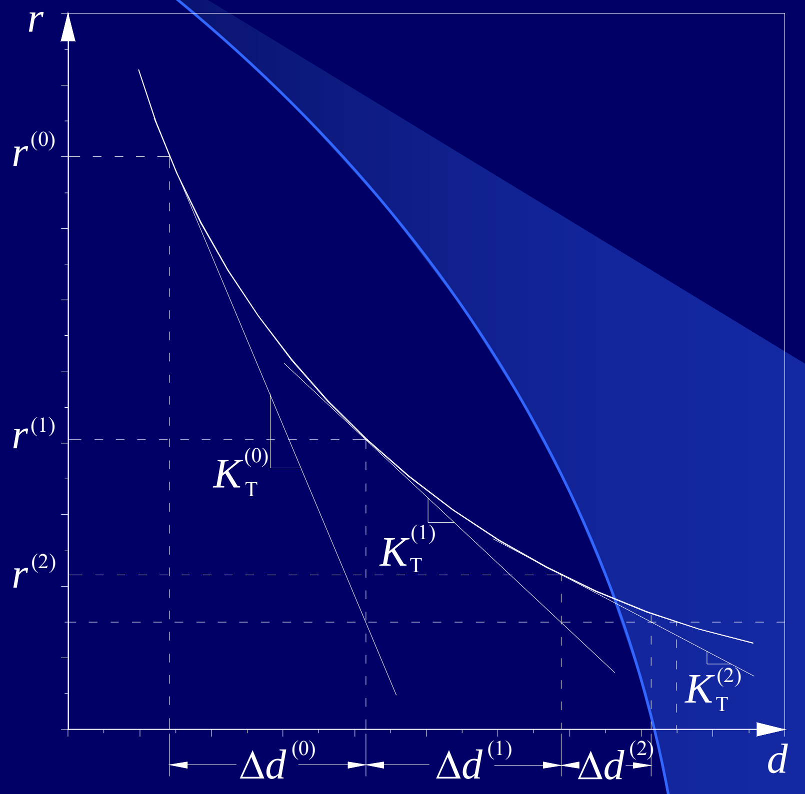
Update:

$$\mathbf{d}^{i+1} = \mathbf{d}^i + \Delta \mathbf{d}^i$$

END FOR

Convergence criteria:

$$\left(\frac{\|\mathbf{r}^i\|}{\|\mathbf{r}^0\|} \geq \tau_{res} \right) \textbf{ AND } \left(\frac{\|\Delta \mathbf{d}^i\|}{\|\mathbf{d}^i\|} \geq \tau_{\Delta \mathbf{d}} \right) \textbf{ OR } \left(i > \text{numax} \right)$$



Backtracking

Linear

Properties of Nonlinear methods

(Some considerations)

- **COMPUTATIONAL COST:**
  - Tangent matrix evaluation (*Jacobian*) at each nonlinear iteration;
  - Locally linear system solution.

- **STABILITY AND CONVERGENCE**
  - **NEWTON METHODS:**
    Quadratic for good initial solutions (kick);

  - **PICARD:**
    Linear asymptotic.

NACAD

Optimization Possibilities

Linear systems

- Iterative solvers;
- Tolerance relaxation;
- Etc...

Jacobian

- Do not update;
- Do not evaluate;
- Alternative approaches;
- Etc...

Convergence

- Incremental methods
- Globalization with (*Backtracking*);
- Etc...

Large scale problems

- Parallelization
- Vetorization
- Both of them
- Etc...

Inexact Newton Method

Initial "kick": $\mathbf{d}^0 = 0$

FOR i = 0 UNTIL CONVERGENCE DO:

Find $\eta^i \in [0,1)$ and $\Delta\mathbf{d}^i$ such that:

$$\left\| \mathbf{N}\left(\mathbf{d}^i\right) + \mathbf{K}_T\left(\mathbf{d}^i\right)\Delta\mathbf{d}^i \right\| \le \eta^i \left\| \mathbf{N}\left(\mathbf{d}^i\right) \right\|$$

Update:

$$\mathbf{d}^{i+1} = \mathbf{d}^i + \Delta\mathbf{d}^i$$

END FOR

$\eta_i \rightarrow$ **Forcing function**

Forcing Function (Linear tolerance)

(How to compute...)

Some possibilities:

1a.) Select any $\eta_0 \in [0,1)$ and choose:

$$\eta^{i+1} = \frac{\left| \left\| \mathbf{N}\left(\mathbf{d}^{i+1}\right) \right\| - \left\| \mathbf{N}\left(\mathbf{d}^{i}\right) + \mathbf{K}_T\left(\mathbf{d}^{i}\right)\Delta\mathbf{d}^{i} \right\| \right|}{\left\| \mathbf{N}\left(\mathbf{d}^{i}\right) \right\|}$$

2a.) For $\gamma \in [0,1)$ e $\alpha \in [1,2)$, select any $\eta_0 \in [0,1)$ and choose:

$$\eta^{i+1} = \gamma \left(\frac{\left\| \mathbf{N}\left(\mathbf{d}^{i+1}\right) \right\|}{\left\| \mathbf{N}\left(\mathbf{d}^{i}\right) \right\|} \right)^{\alpha}$$

Our choice:

$\alpha = 2$ e $\gamma = 0.99$

(S. C. EISENSTAT & H. F. WALKER, 1994)

Inexact Newton with backtracking

Initial "kick": $\mathbf{d}^0 = 0$

FOR i = 0 UNTIL CONVERGENCE DO:

a). Find $\eta^i \in [0,1)$ and $\Delta\mathbf{d}^i$ such that:

$$\left\| \mathbf{N}(\mathbf{d}^i) + \mathbf{K}_T(\mathbf{d}^i)\Delta\mathbf{d}^i \right\| \le \eta^i \left\| \mathbf{N}(\mathbf{d}^i) \right\|$$

b). DO $\lambda = 1$

i. Update: $\mathbf{d}_t^{i+1} = \mathbf{d}^i + \lambda\,\Delta\mathbf{d}^i$

ii. IF $\left\| \mathbf{N}(\mathbf{d}_t^{i+1}) \right\| < (1-\alpha\lambda)\left\| \mathbf{N}(\mathbf{d}^{i+1}) \right\|$ **THEN:** $\mathbf{d}^{i+1} = \mathbf{d}_t^{i+1}$

ELSE:

Choose $\sigma \in [\sigma_0, \sigma_1]$

$\lambda = \sigma\lambda$

GO TO (b).i

END FOR

NACAD

Newton x Picard

(Mixed method – A proposal...)

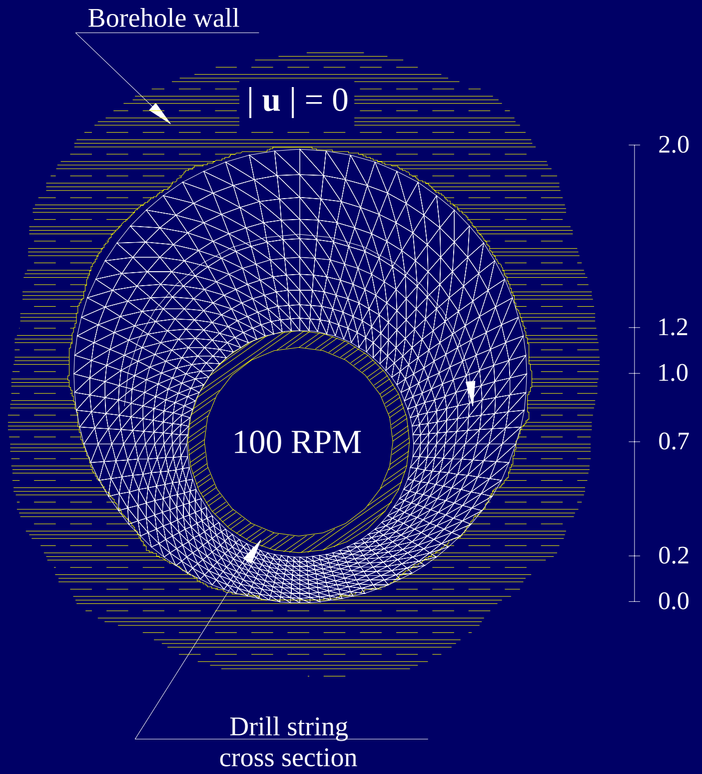
- **INEXACT NEWTON (IN)=** The Jacobian matrix is evaluated;
- **INEXACT PICARD (IP)=** The Jacobian is **NOT** evaluated;
- **Mixed method (*n*-IP+IN)=** It is performed *n* Picard iterations, then the Newton method is turned on;

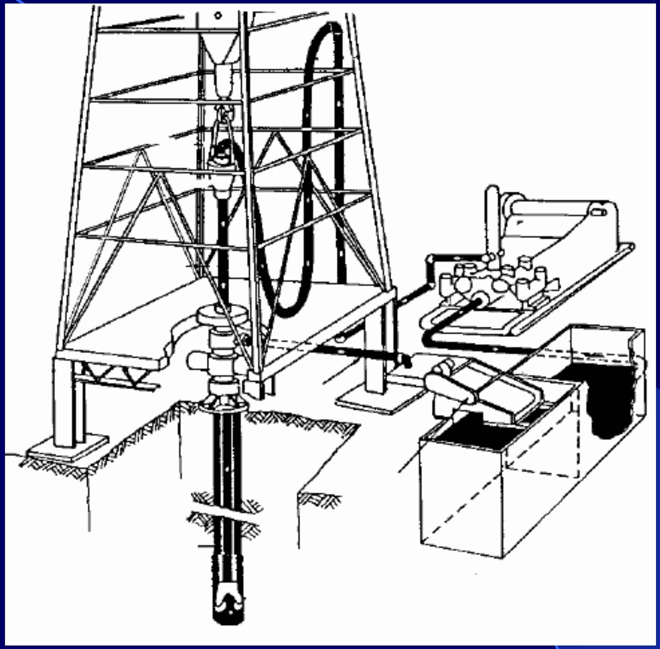
Computational Resources

- *Cluster ITAUTEC Infoserver* do NACAD-COPPE/UFRJ.

  16 nodes *Intel Dual Pentium III* with 1Ghz, 512 Mb of memory and 256 Kb of cache, over *Red Hat Linux* platform;

Rotational eccentric annulus flow

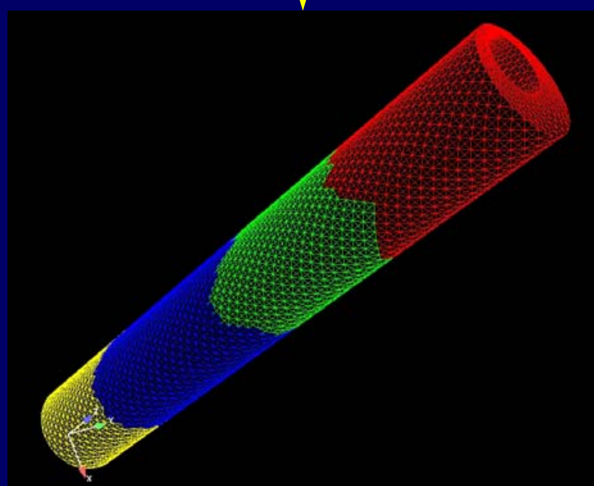




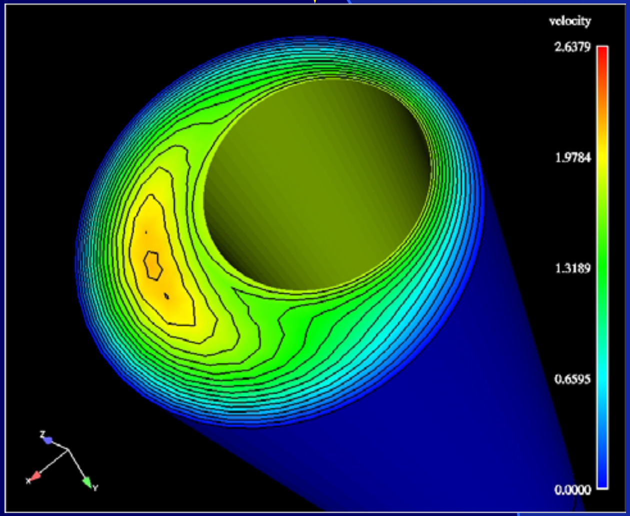
Solution Procedures

REAL PROBLEM

DISCRETE PROBLEM
PARALLELIZED



CLUSTER INFOSERVER ITAUTEC



RESULT

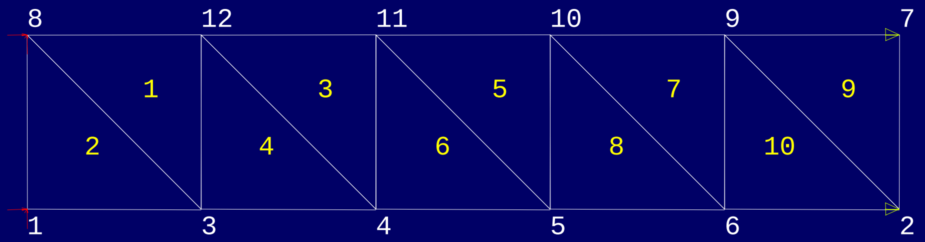
NACAD

Parallel Issues:
FEM with Message Passing

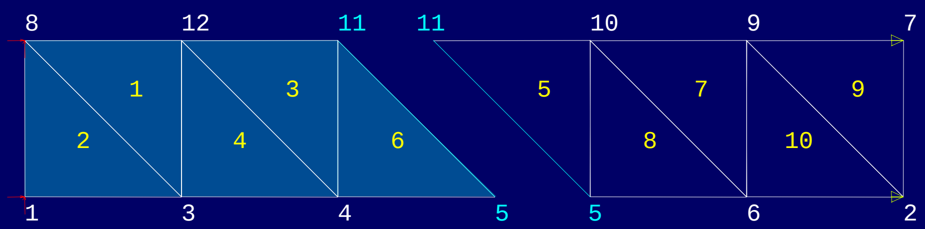
- Preprocessing:
  - Domain decomposition with METIS library;
  - Reodering partitions.
- Solution:
  - Data distribution with **MPI** (MPI_BCAST, MPI_SEND ande MPI_RECV);
  - Updates with MPI_ALLREDUCE.

Mesh preprocessing

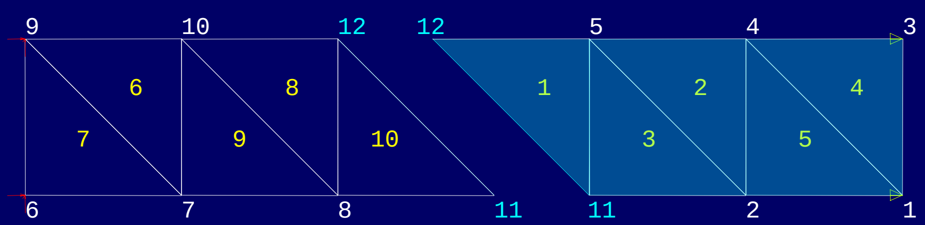
Original mesh



Partitioned mesh by METIS library



Partitioned and reordered mesh



Partition 2 Partition 1

Nodal coordinates before reordering

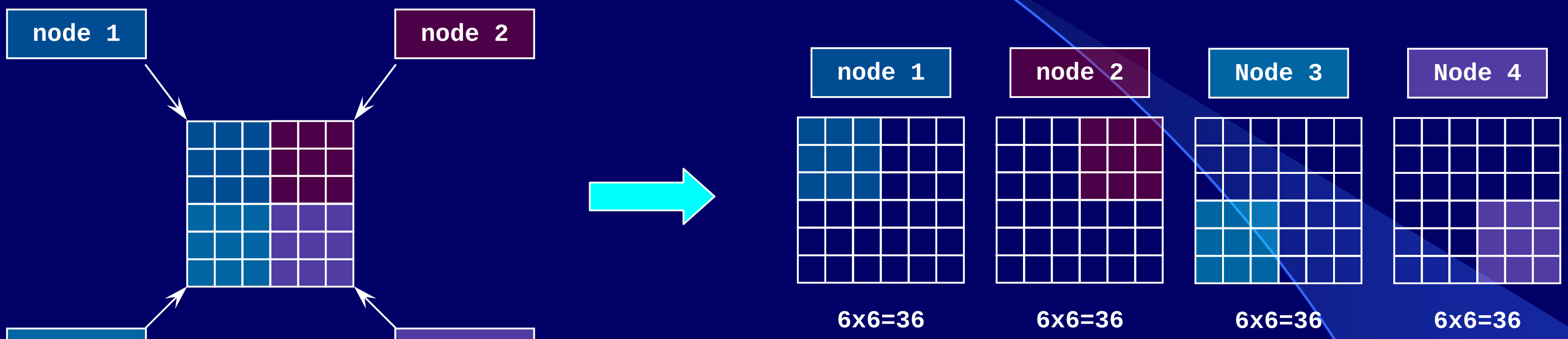
1	0.000E+00	0.000E+00	0.000E+00
2	1.000E+01	0.000E+00	0.000E+00
3	2.000E+00	0.000E+00	0.000E+00
4	4.000E+00	0.000E+00	0.000E+00
5	6.000E+00	0.000E+00	0.000E+00
6	8.000E+00	0.000E+00	0.000E+00
7	1.000E+01	2.000E+00	0.000E+00
8	0.000E+00	2.000E+00	0.000E+00
9	8.000E+00	2.000E+00	0.000E+00
10	6.000E+00	2.000E+00	0.000E+00
11	4.000E+00	2.000E+00	0.000E+00
12	2.000E+00	2.000E+00	0.000E+00

Nodal coordinates after reordering

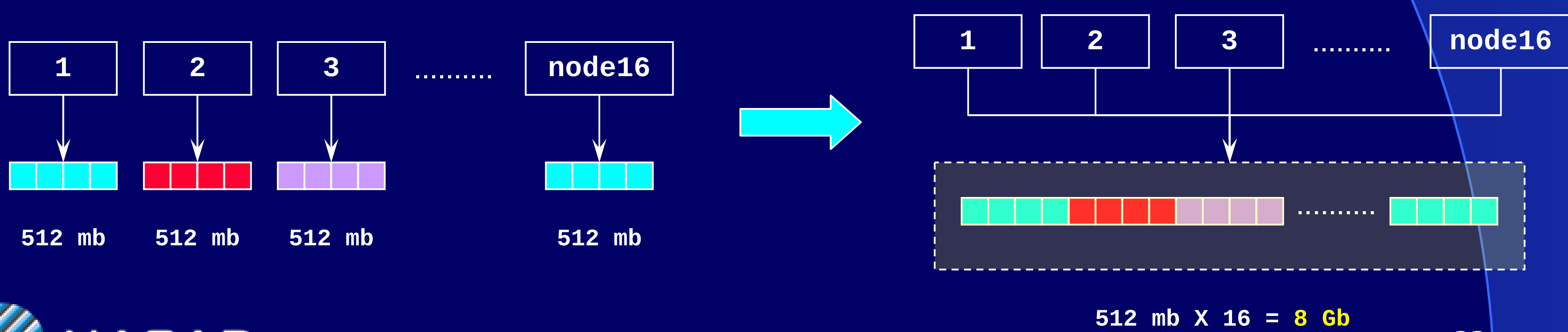
1	1.000E+01	0.000E+00	0.000E+00
2	8.000E+00	0.000E+00	0.000E+00
3	1.000E+01	2.000E+00	0.000E+00
4	8.000E+00	2.000E+00	0.000E+00
5	6.000E+00	2.000E+00	0.000E+00
6	0.000E+00	0.000E+00	0.000E+00
7	2.000E+00	0.000E+00	0.000E+00
8	4.000E+00	0.000E+00	0.000E+00
9	0.000E+00	2.000E+00	0.000E+00
10	2.000E+00	2.000E+00	0.000E+00
11	6.000E+00	0.000E+00	0.000E+00
12	4.000E+00	2.000E+00	0.000E+00

Partitioning data:

Avoiding memory loss:

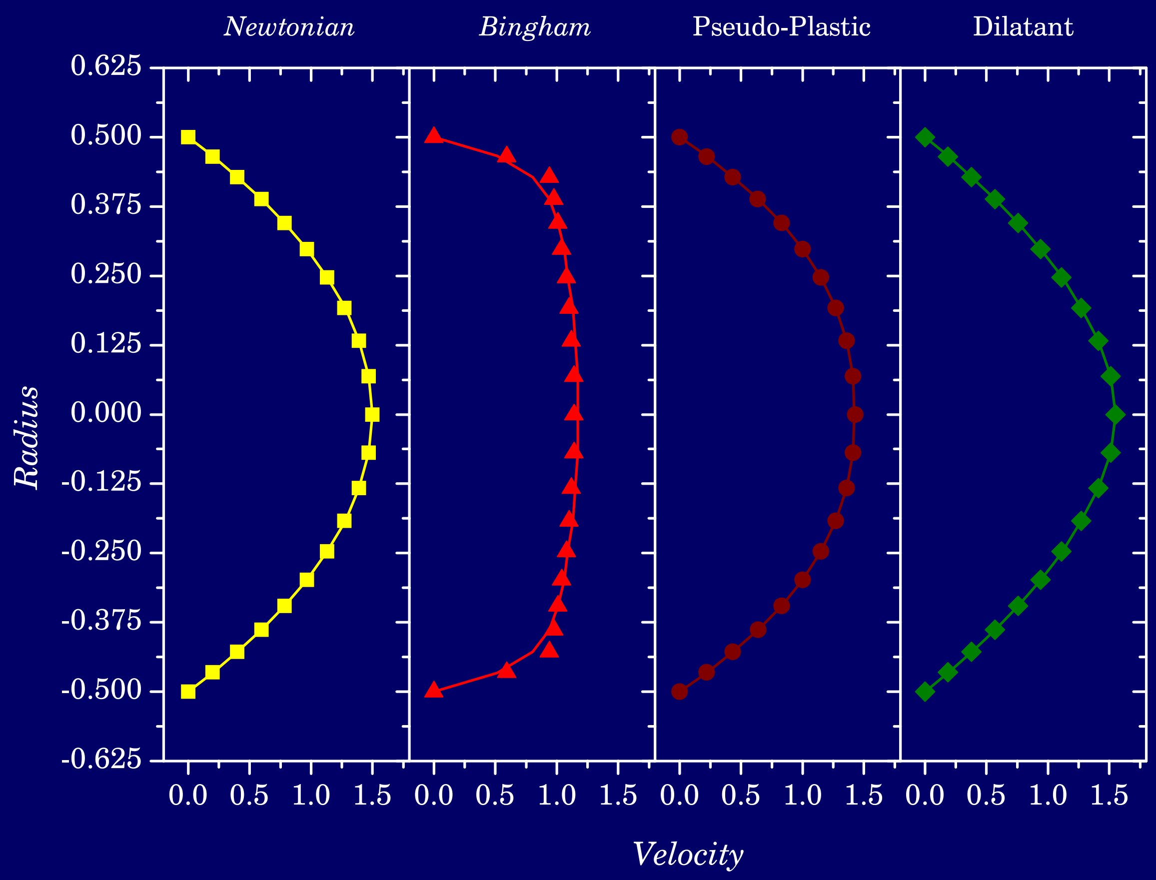


Cluster Itautec: 16 nodes with 512 mb per node



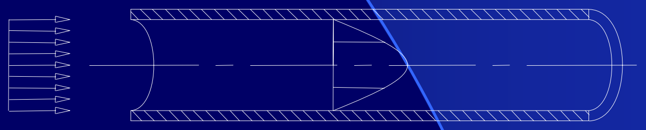
Validation

Escoamento no interior de um duto circular



Line = Comertial software (Ansys/Flotran),

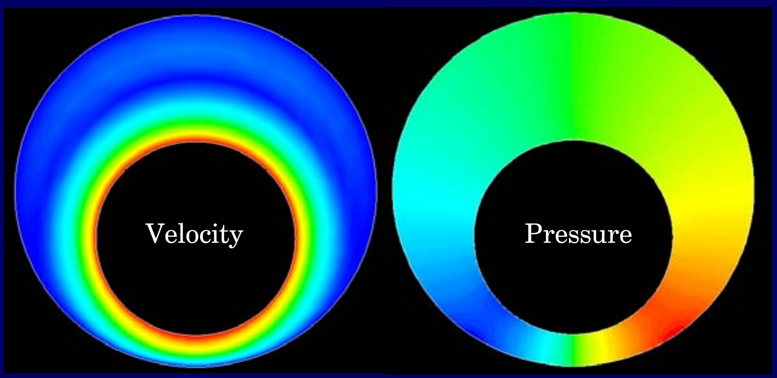
Dots = Software developed



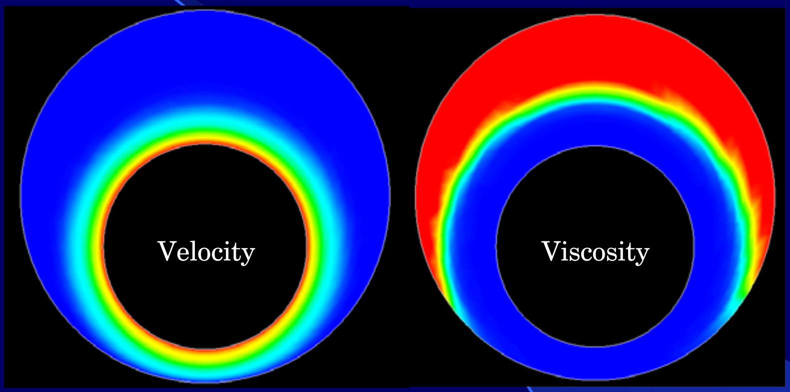
Cross Section

Rotational eccentric annulus flow

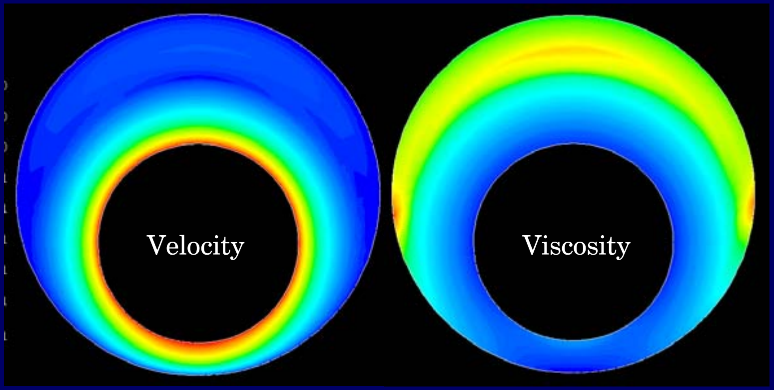
(Results)



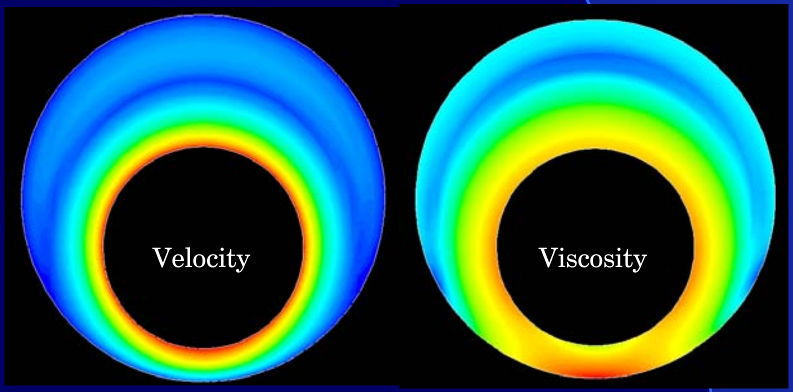
Newtonian



Bingham

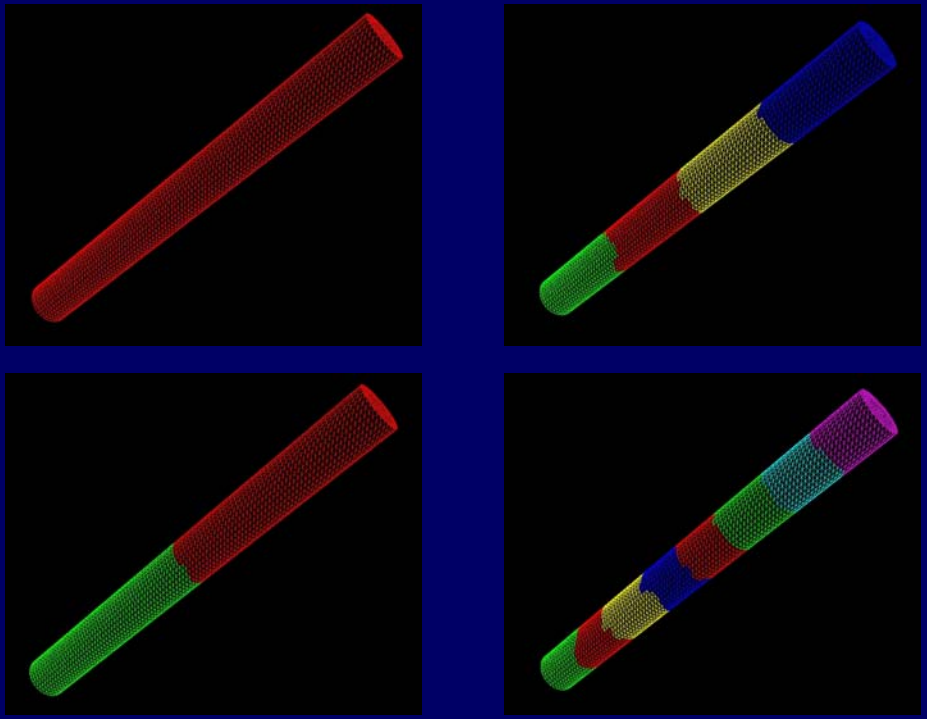


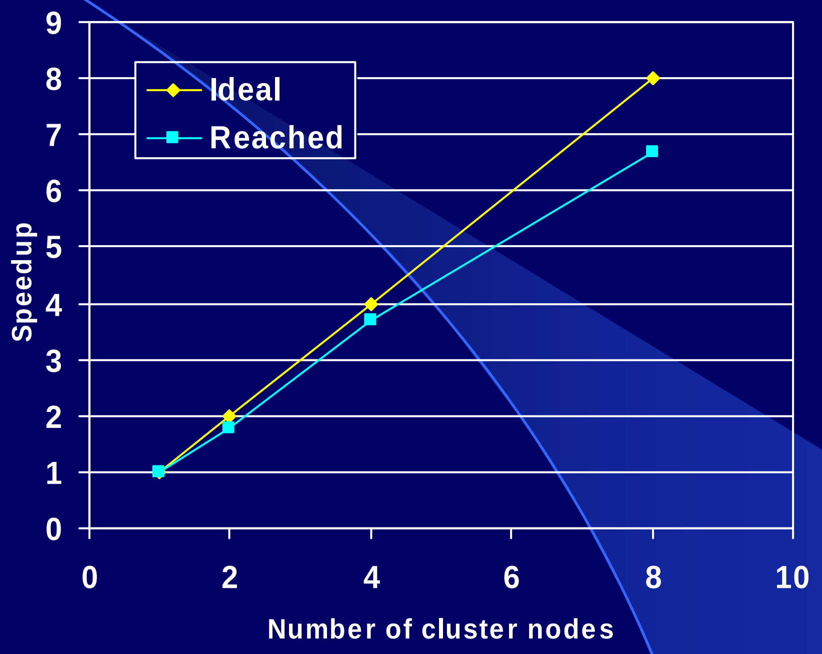
Pseudo-plastic



Dilatant

Performance (Speedup)





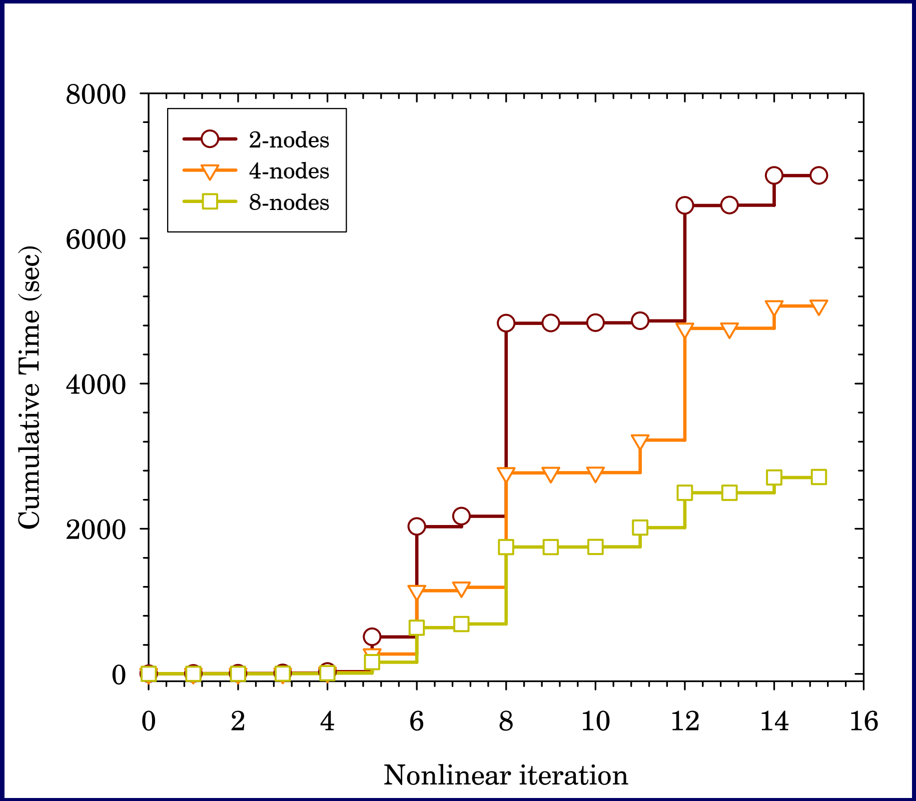

```
Technical Info:
========= ====
Nodes.................:
Tetrahedron...........:
Velocity equations....:
Pressure equations....:
```

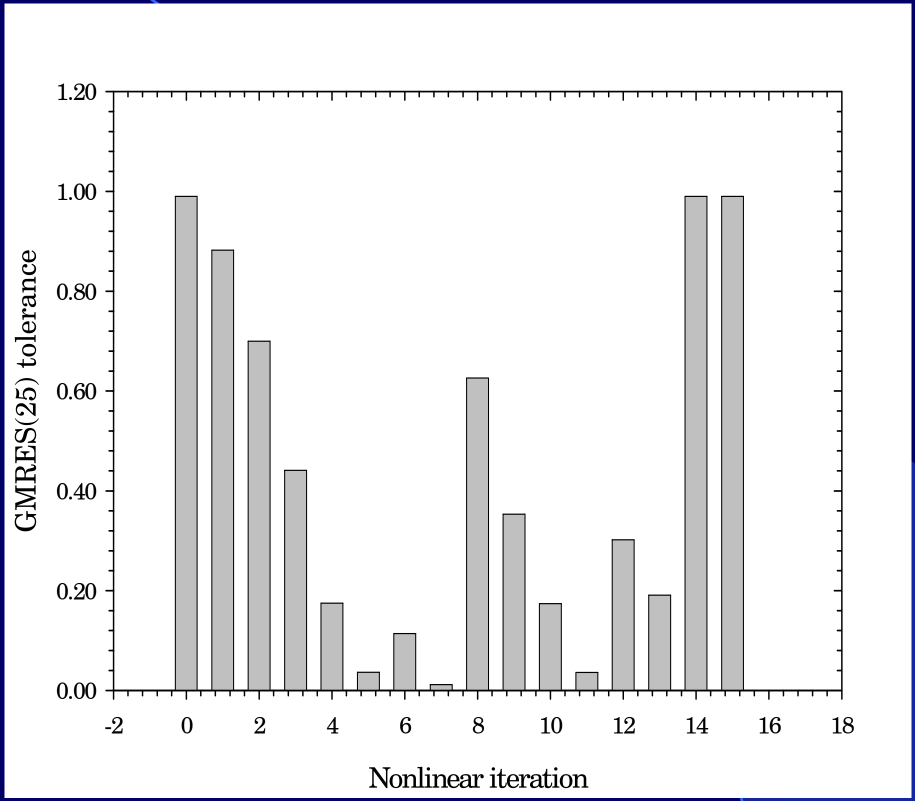
Minor → Major

Communication

Inexact Picard

How it works...

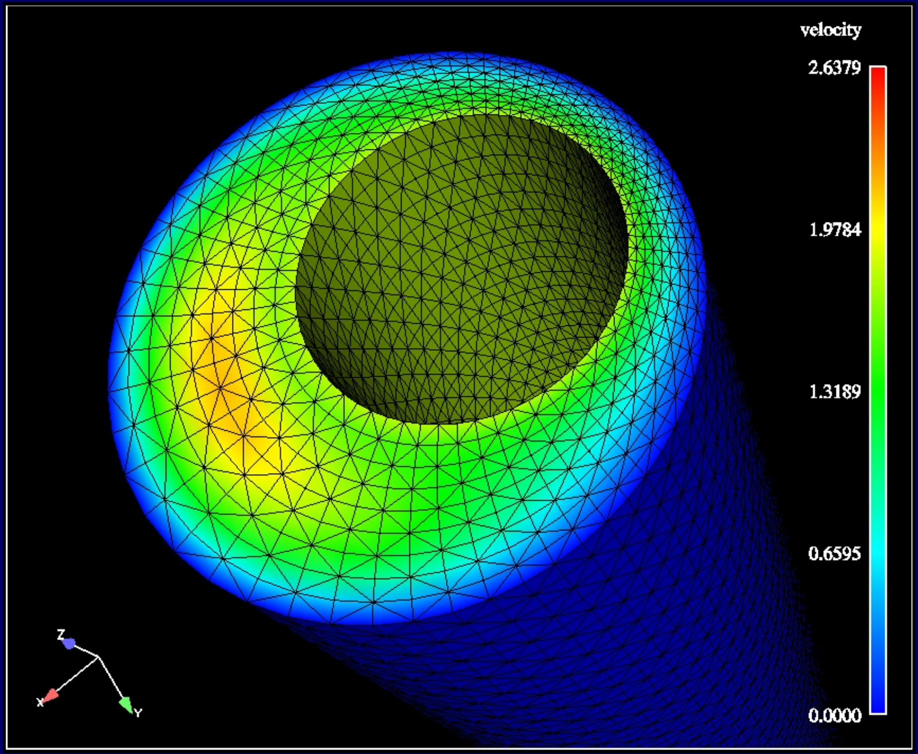




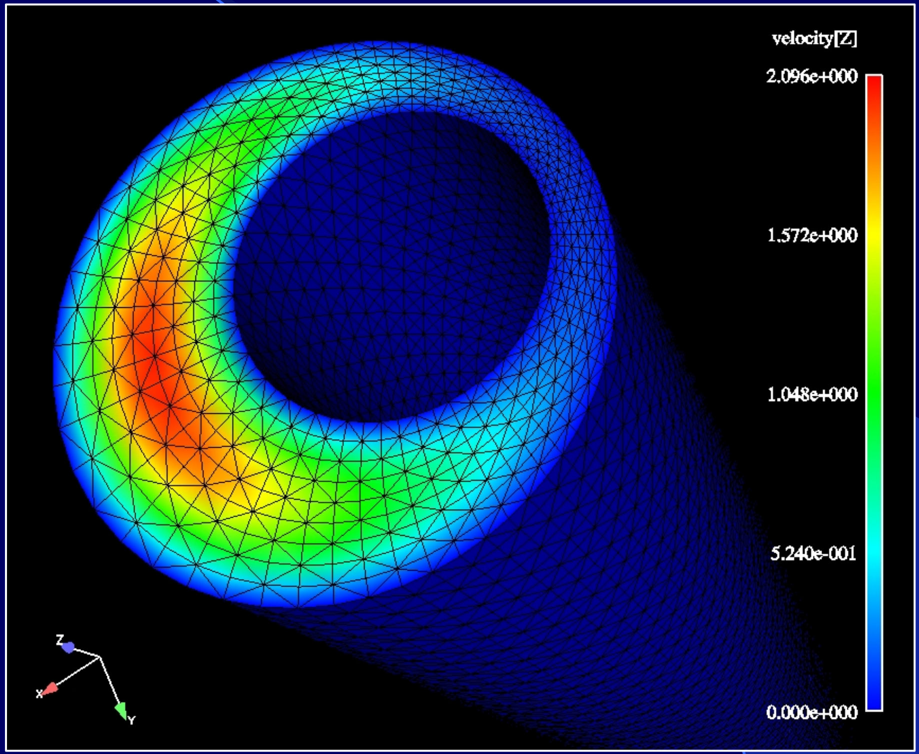
Rotational eccentric anulus flow with Power Law (0.75) fluid

Results
(3D parallel inexact-solver)

Power Law (0.75)

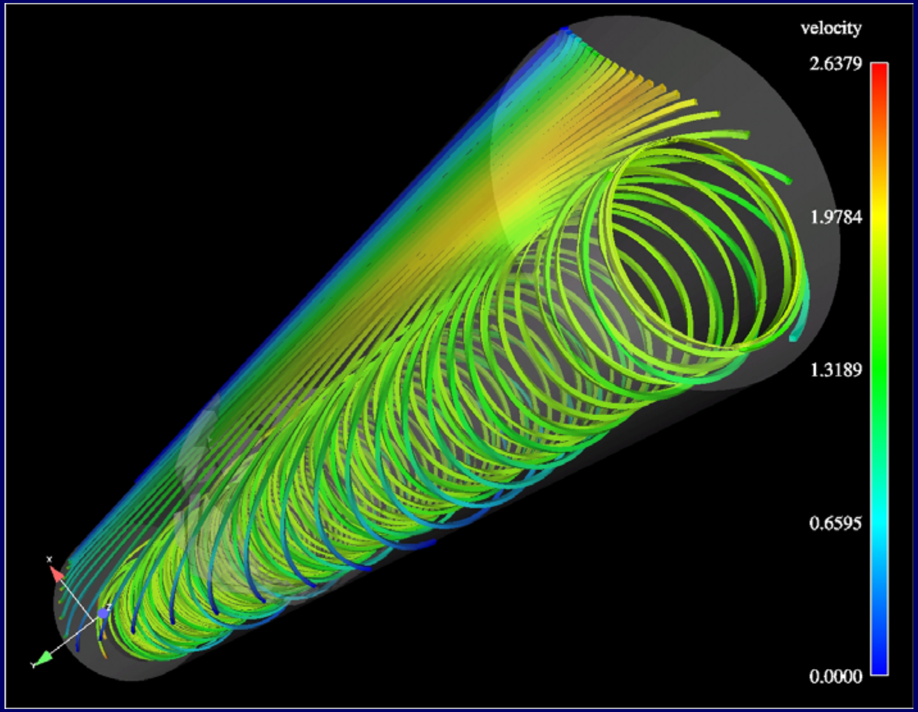


Velocity field

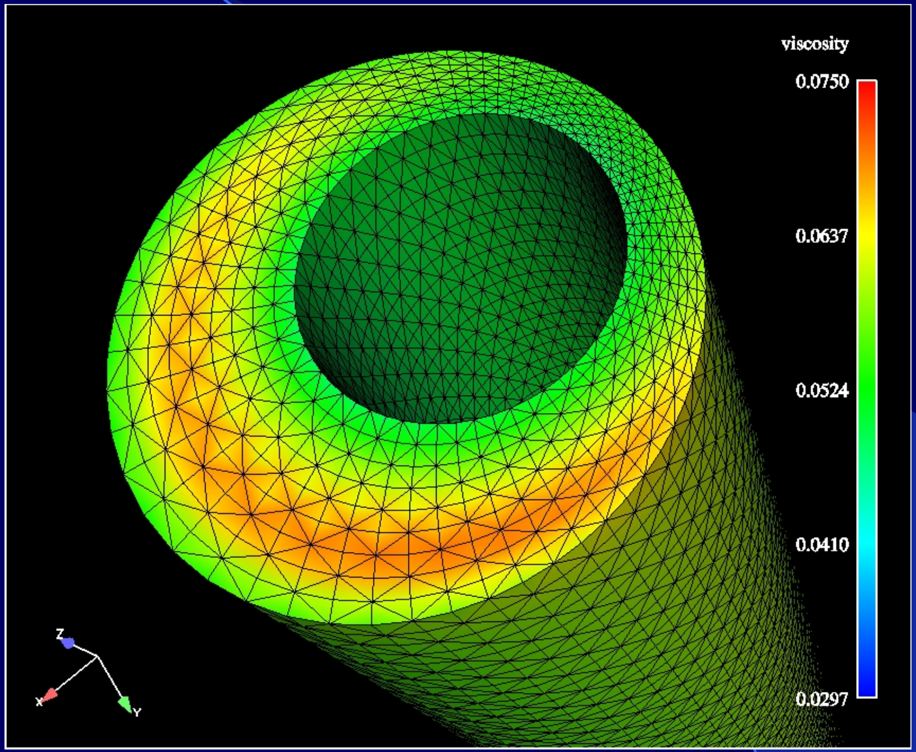


Axial Velocity

Results
(3D parallel inexact-solver)



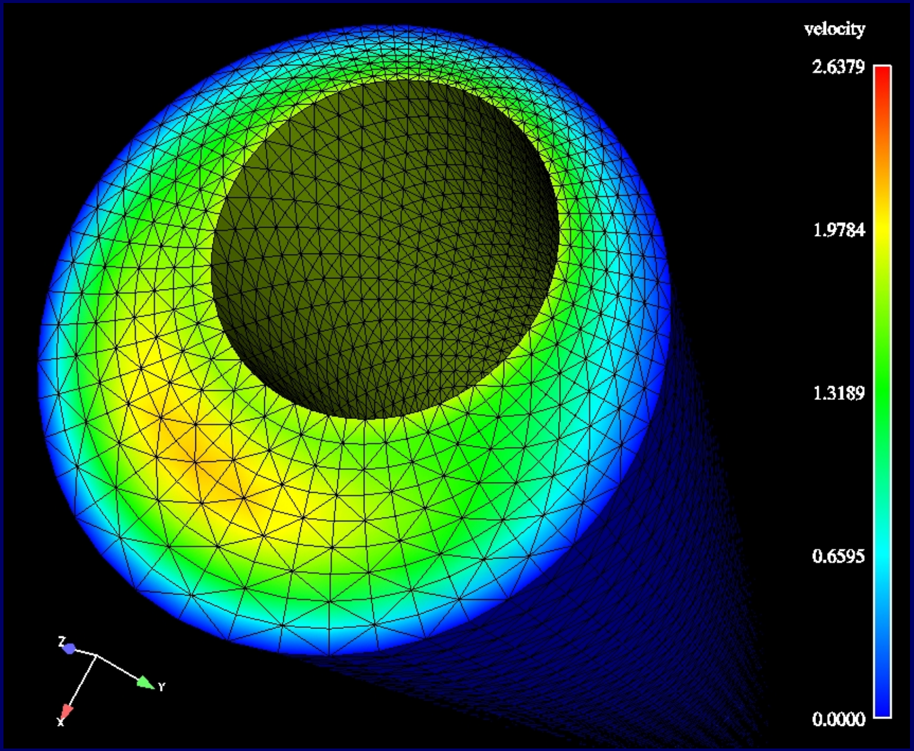
Streamlines



Viscosity

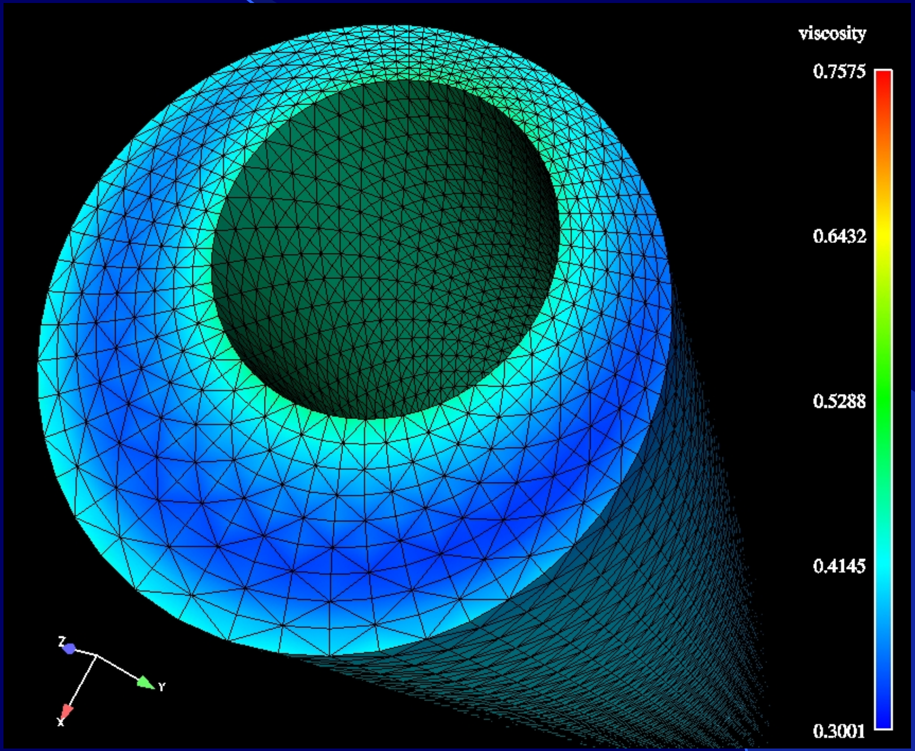
Results
(3D parallel inexact-solver)

Power Law (1.25)



Dilatant

Velocity field

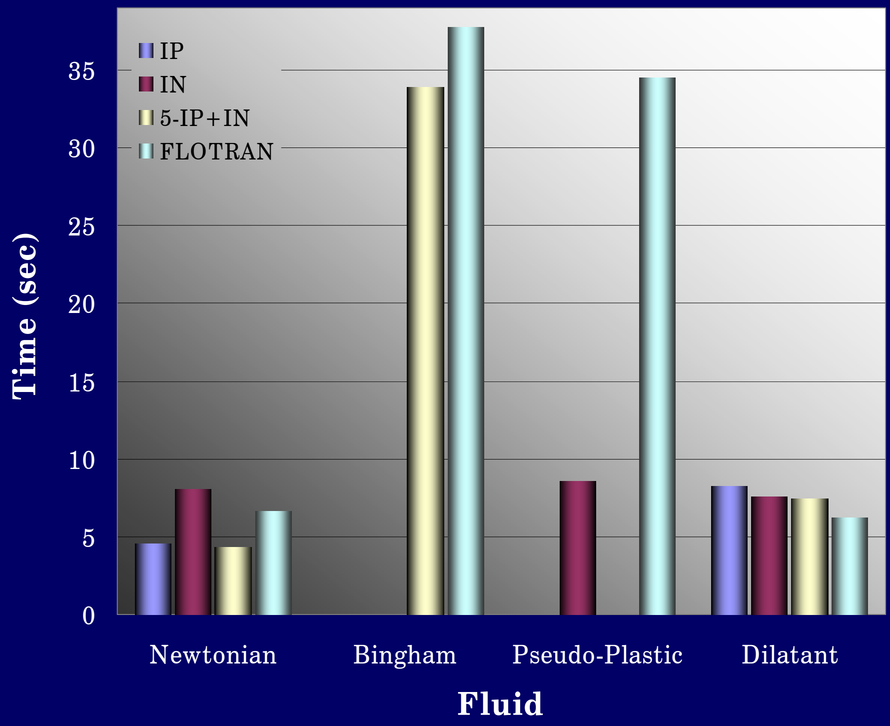


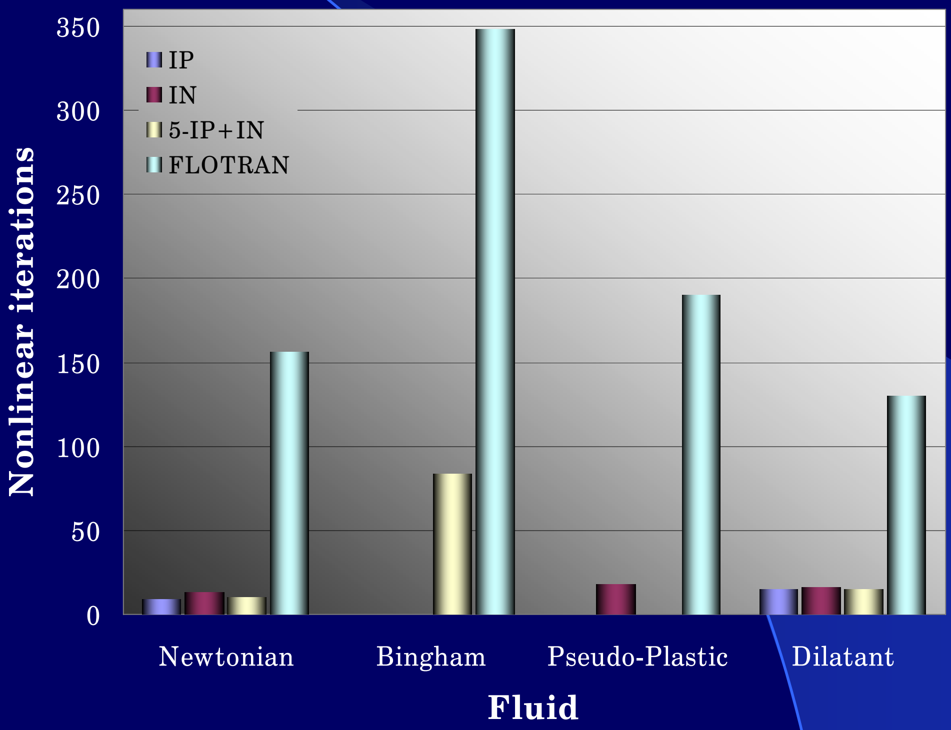
Dilatant

Viscosity field

Non-Newtonian Fluid Flow

(Comparatives)



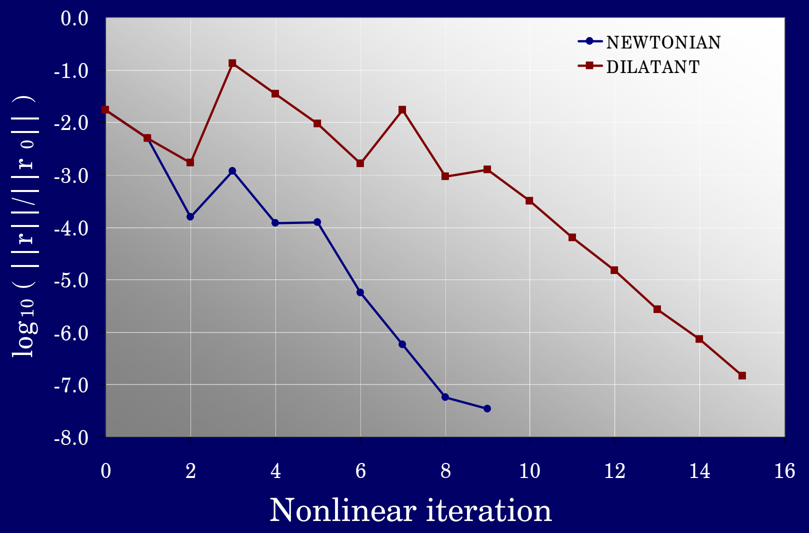


Rotational Eccentric Anulus flow (2D section)

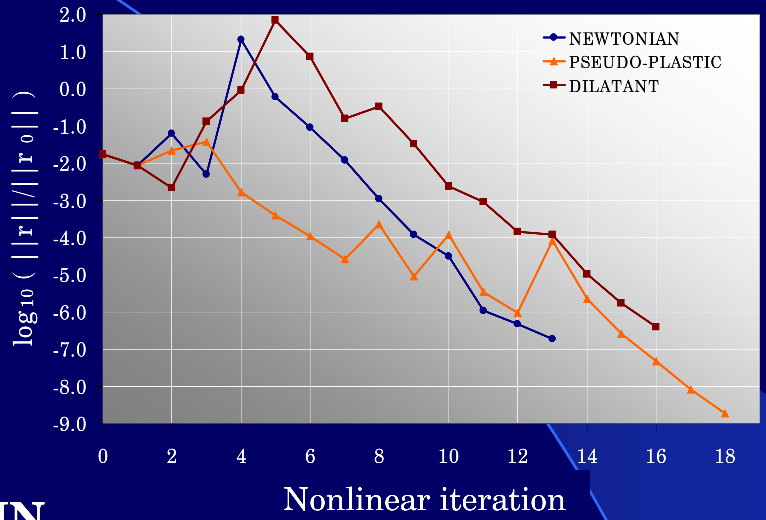
Method comparatives IP, IN ou 5-IP+IN

(Convergence)

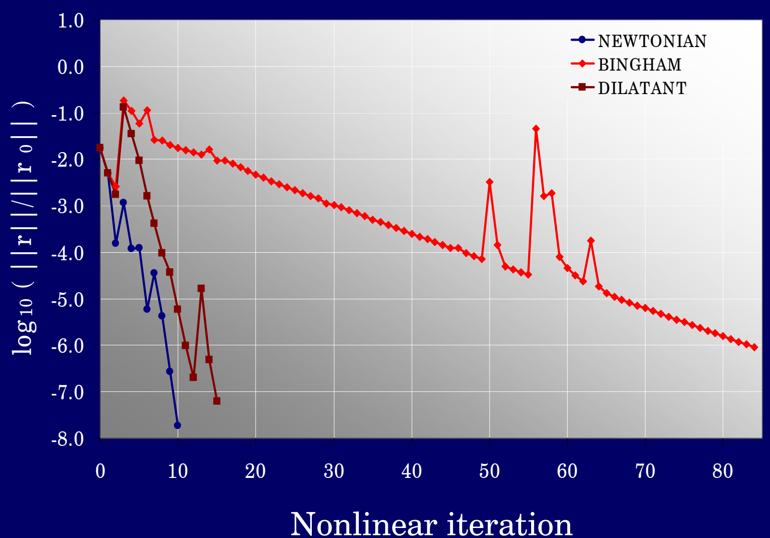
IP



IN



5-IP+IN



NACAD

Conclusions

- Parallel algorithms presented good performance;
- Parallel non-Newtonian flow simulations are suitable to predict large scale problems, such as, well drilling;
- Inexact Newton-type methods were faster than their classic versions;
- Implementation presented good agreement with Ansys/Flotran commercial software;
- Jacobian based in numeric derivatives presented inconstant perfomance;
- Among non-Newtonian fluids considered the Bingham plastic was the hardest of being computed;