Computational Modeling of Fluid-Fluid Flows Employing Adaptive Mesh Refinement, Front-Tracking, Immersed Boundary and Volume-of-Fluid

Prof. Dr. Aristeu da Silveira Neto (UFU/FEMEC)

Work Team

- Supervisors:
  - Prof. Dr. Aristeu da Silveira Neto - FEMEC-UFU/MG
  - Prof. Dr. Alexandre Meggiorim Roma - IME-USP/SP
- Researches: Doctors and Masters
  - Dra. Millena M. Villar Vale
  - Dr. Márcio Ricardo Pivello
  - Msc. Renato Pacheco
  - Msc. Rodrigo Lisita
  - Msc. Franco Barbi
  - Msc. Lucas Vela
- Scientific Initiation Students
  - Hélio Ribeiro Neto
  - Fernando Muniz
  - Lucas Alvarenga
- technical Support
  - Luizmar Lopez, Rodrigo Saramago e Bruno Martins
- External Collaborators
  - Prof. Dr. Berend Van-Wachen (Imperial College - London)
  - Prof. Dra. Catalina Rua (Universidad de Nariño - Colombia)
  - Prof. Dr. Rafael Sene (UFTPr)
  - Prof. Dra. Priscila Calegari

Infrastructure: MFLab

Infrastructure: Cluster - MFLab

SGI ICE X e SGI Altix XE

- 2 racks e 54 nodes;
- Total numbers of cores (real + virtual): 1632 cores;
- Total memory: 5.3 TeraBytes;
- Disk space for data storage: 85TeraBytes;
- Theoretical peak performance: 19.1 TeraFlops = 19178 GigaFlops
- Interconnection: InfiniBand

Beowulf

- Computational nodes: 36
- Total numbers of cores: 284 cores;
- Total memory: 406 GigaBytes
- Disk space for data storage: 12.4TeraBytes;
- Interconnection: Gigabit Ethernet

Objetivos

Desenvolvimento de uma ferramenta computacional capaz de simular escoamentos complexos presentes na indústria petrolífera. Exemplos:

- Escoamentos monofásicos;
- Escoamentos bifásicos/multifásicos;
- Escoamentos reativos;
- Escoamentos com a presença de geometrias complexas;
- Escoamentos à variados números de Reynolds;
- Interação fluido-estrutura.

Physical flows aspects that must be modeled

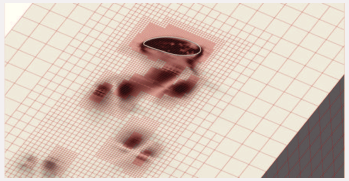
1. Shape interfaces;
2. Deformable mobile interfaces;
3. physical properties with high aspect ratio;
4. Detachment and replacement drops/droplets;
5. Bubbles/drops $\ll$ domain $\Rightarrow$ located refinement;
6. High Reynolds number $\Rightarrow$ turbulence modeling;
7. Surface tension and physical properties $\Rightarrow$ discontinuities;
8. Triple contact presence: solid, liquid and gas;
9. Physical mechanisms of objects transports and formation (drops, solid particles) and high number and different scales of time and length.

AMR3d code features

Adaptive Mesh Refinement - AMR3d

- Based on local structured adaptive mesh refinement in space and time;
- Paralle, with MPI domain partition;
- Second order in space and time;
- Temporal discretization: semi-implicit (two phases flow) and implicit (reactives);
- Spatial discretization: finite difference (two phases flow) and finite volumes (reactives);
- Linear Systems: Multigrid-multilvel method, Strong Implicit Procedure (SIP), PETSC;

Local structured adaptive mesh refinement

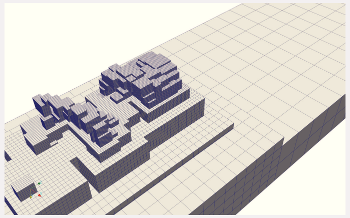


AMR3d code features

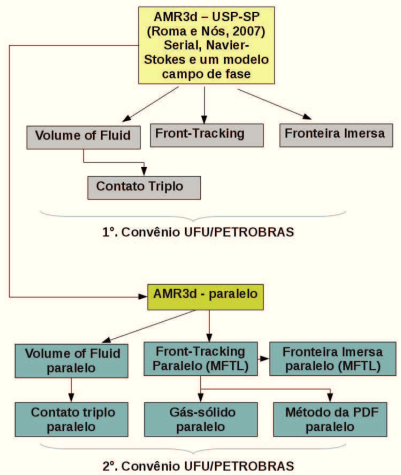
Adaptive Mesh Refinement - AMR3d

- Volume-of-Fluid Method and e Front-Tracking Method to represent the fluid-fluid interface;
- Immersed Boundary Method to represent the static and rigid budies;
- LES methodology to the turbulence modeling;
- Euler-Lagrange modeling to droplets transport;
- Triple contact line modeling.

Local structured adaptive mesh refinement



Development historical code



Mathematical Modeling

Governement Equation

$$\rho(\phi)[\mathbf{u_t} + (\mathbf{u}\cdot\nabla)\mathbf{u}] = \nabla\cdot[\mu(\phi)(\nabla\mathbf{u} + \nabla\mathbf{u}^{\dagger})] - \nabla\mathbf{p} + \rho(\phi)\mathbf{g} + \mathbf{f}_{\sigma} + \mathbf{f_s},$$

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

$$\rho(\phi) = \rho_c + (\rho_d - \rho_c)\,\phi(\mathbf{X}, t)$$
$$\mu(\phi) = \mu_c + (\mu_d - \mu_c)\,\phi(\mathbf{X}, t)$$

ϕ: indicator function
F-T method: level set with sign
VoF method: color function

Temporal discretization

IMEX Schemes

$$\frac{\rho(\phi)^{n+1}}{\Delta t}(\alpha_2\mathbf{u}^{n+1}+\alpha_1\mathbf{u}^{n}+\alpha_0\mathbf{u}^{n-1})=\beta_1 f(\mathbf{u}^{n})+\beta_0 f(\mathbf{u}^{n-1})+$$
$$\lambda\left[\theta_2\nabla^2\mathbf{u}^{n+1}+\theta_1\nabla^2\mathbf{u}^{n}+\theta_0\nabla^2\mathbf{u}^{n-1}\right]-\nabla p^{n}+\rho^{n+1}(\phi)\mathbf{g},$$
$$f(\mathbf{u})=-\lambda\nabla^2\mathbf{u}+\nabla\cdot\left[\mu(\nabla\mathbf{u}+\nabla\mathbf{u}^T\right]-\mathbf{u}\cdot\nabla\mathbf{u}+\mathbf{f}_\sigma$$

IMEX parameters for variable time step

$$\alpha_2=\frac{\Delta t_0+2\gamma\Delta t_1}{\Delta t_0+\Delta t_1},\quad \theta_2=\gamma+c\frac{\Delta t_1}{\Delta t_0+\Delta t_1}$$
$$\alpha_1=\frac{\Delta t_1-\Delta t_0-2\gamma\Delta t_1}{\Delta t_0},\quad \theta_1=1-\gamma-c\frac{\Delta t_1}{\Delta t_0}$$
$$\alpha_0=-\alpha_1-\alpha_2,\quad \theta_0=c\left(\frac{\Delta t_1}{\Delta t_0}-\frac{\Delta t_1}{\Delta t_0+\Delta t_1}\right),$$

$$\beta_1=\frac{\Delta t_0+\gamma\Delta t_1}{\Delta t_0},\quad \beta_0=-\gamma\frac{\Delta t_1}{\Delta t_0},$$

Temporal discretization

IMEX Schemes

- SBDF (Semi Backward Difference Formula): (γ, c) = (1,0);
- CNAB (Crack-Nicolson Adans-Bashforth): (γ, c)= $(\frac{1}{2}, 0)$;
- MCNAB (Modified Crack-Nicolson Adans-Bashforth): (γ, c)= $(\frac{1}{2}, \frac{1}{8})$;
- CNLF (Cranck-Nicolson Leap-Frog): (γ, c) = (0,1).

Spatial discretization

Example: finite difference to the diffusive term

$$\nabla\cdot\left[\mu_{ef}(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\right],$$

$$\mu_{ef}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+\mu_{ef}\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right)+2\frac{\partial\mu_{ef}}{\partial x}\frac{\partial u}{\partial x}+\frac{\partial\mu_{ef}}{\partial y}\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right),$$

$$\frac{\mu_{ef_{i,j}}+\mu_{ef_{i-1,j}}}{2}\left(\frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{\Delta x^2}+\frac{u_{i,j+1}-2u_{i,j}+u_{i,j-1}}{\Delta y^2}\right)+$$
$$\frac{\mu_{ef_{i,j}}+\mu_{ef_{i-1,j}}}{2}\left(\frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{\Delta x^2}+\frac{v_{i,j+1}-v_{i,j}}{\Delta x\Delta y}-\frac{v_{i-1,j+1}-v_{i-1,j}}{\Delta x\Delta y}\right)+$$
$$2\left(\frac{\mu_{ef_{i,j}}-\mu_{ef_{i-1,j}}}{\Delta x}\frac{u_{i+1,j}-u_{i-1,j}}{2\Delta x}\right)+\left(\frac{\mu_{ef_{i,j+1}}-\mu_{ef_{i,j-1}}}{4\Delta y}+\frac{\mu_{ef_{i-1,j+1}}-\mu_{ef_{i-1,j-1}}}{4\Delta y}\right)$$
$$\left[\frac{u_{i,j+1}-u_{i,j-1}}{2\Delta y}+\frac{v_{i,j+1}-v_{i-1,j+1}}{2\Delta x}+\frac{v_{i,j}-v_{i-1,j}}{2\Delta x}\right].$$

Advective term

Quick, SOU, cubist, Barton Scheme, Euler first order, Euler second order, Weno, ENO second order

Mathematical Modeling

Pressure-velocity Coupling Method

Fractional Step Method

- Pressure correction equation: Poisson equation;

$$\nabla \cdot \left[\frac{1}{\rho(\phi)^{n+1}} \nabla q^{n+1} \right] = \frac{\alpha_2}{\Delta t} \nabla \cdot \mathbf{u}^{**},$$

- Velocity correction

$$\mathbf{u}^{n+1} = \mathbf{u}^{**} - \frac{\Delta t \nabla q^{n+1}}{\alpha_2 \rho(\phi)^{n+1}},$$

- Pressure

$$p^{n+1} = p^n + q^{n+1}.$$

Numerical Methodology

Discretization of pressure correction equation

$$\nabla \cdot \left[\frac{1}{\rho^n} \nabla q^{n+1}\right] = \frac{\alpha_2}{\Delta t} \nabla \cdot \tilde{\mathbf{u}}^{n+1}$$

$$\frac{\partial}{\partial x}\left(\frac{1}{\rho}\frac{\partial q}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{1}{\rho}\frac{\partial q}{\partial y}\right) = \frac{\alpha_2}{\Delta t}\left(\frac{\partial \tilde{u}}{\partial x} + \frac{\partial \tilde{v}}{\partial y}\right)$$

$$\frac{1}{\rho^n_{i+\frac{1}{2},j}}\left(\frac{q_{i+1,j} - q_{i,j}}{\Delta x^2}\right) - \frac{1}{\rho^n_{i-\frac{1}{2},j}}\left(\frac{q_{i,j} - q_{i-1,j}}{\Delta x^2}\right) + \frac{1}{\rho^n_{i,j+\frac{1}{2}}}\left(\frac{q_{i,j+1} - q_{i,j}}{\Delta y^2}\right) - \frac{1}{\rho^n_{i,j-\frac{1}{2}}}\left(\frac{q_{i,j} - q_{i,j-1}}{\Delta y^2}\right) =$$

$$\frac{\alpha_2}{\Delta t}\left(\frac{\tilde{u}_{i+1,j} - \tilde{u}_{i,j}}{\Delta x} + \frac{\tilde{v}_{i,j+1} - \tilde{v}_{i,j}}{\Delta y}\right).$$

$$a_n q_{i,j+1} + a_s q_{i,j-1} + a_e q_{i+1,j} + a_w q_{i-1,j} + a_p q_{i,j} = \Theta_{i,j}$$

Numerical Methodology

Discretization of pressure correction equation

$$a_e = \frac{\rho_e}{\Delta x^2}, \quad \rho_e = \frac{1}{\rho_{i+\frac{1}{2},j}} = \frac{1}{2}\left(\frac{1}{\rho_{i+1,j}} + \frac{1}{\rho_{i,j}}\right),$$

$$a_w = \frac{\rho_w}{\Delta x^2}, \quad \rho_w = \frac{1}{\rho_{i-\frac{1}{2},j}} = \frac{1}{2}\left(\frac{1}{\rho_{i-1,j}} + \frac{1}{\rho_{i,j}}\right),$$

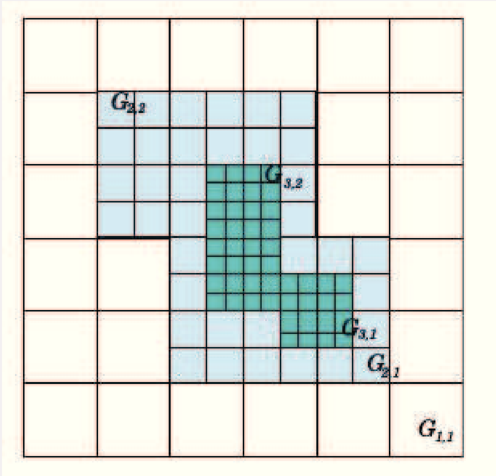
$$a_n = \frac{\rho_n}{\Delta y^2}, \quad \rho_n = \frac{1}{\rho_{i,j+\frac{1}{2}}} = \frac{1}{2}\left(\frac{1}{\rho_{i,j+1}} + \frac{1}{\rho_{i,j}}\right),$$

$$a_s = \frac{\rho_s}{\Delta y^2}, \quad \rho_s = \frac{1}{\rho_{i,j-\frac{1}{2}}} = \frac{1}{2}\left(\frac{1}{\rho_{i,j-1}} + \frac{1}{\rho_{i,j}}\right),$$

$$a_p = -(a_e + a_w + a_n + a_s).$$

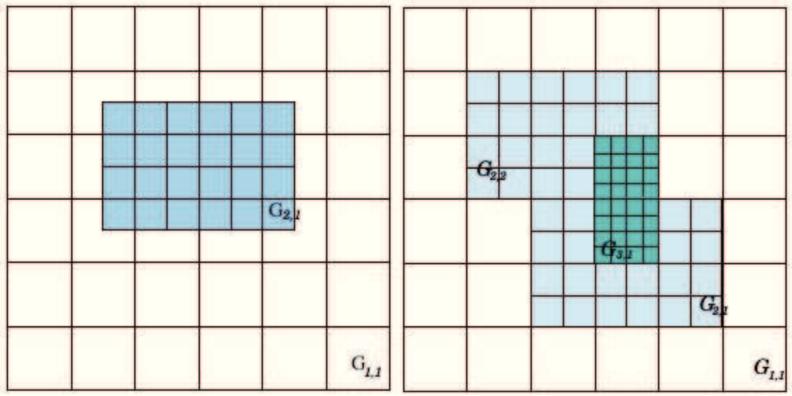
Adaptive mesh refinement

Properly nested mesh



Adaptive mesh refinement

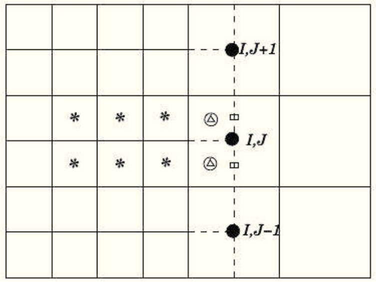
Not properly nested mesh



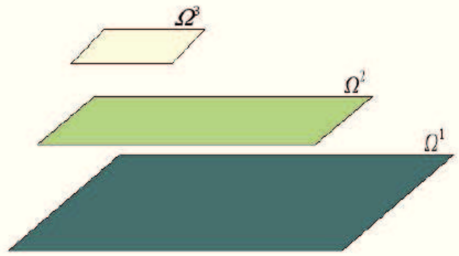
Ghosts cells on an adaptive mesh refinement

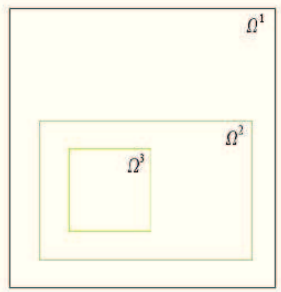
- Extrapolation on the same level ($* \rightarrow \triangle$);
- Interpolation on the coarse level, $l-1$ ($\bullet \rightarrow \square$);
- Interpolation between l and $l-1$, ($\triangle$ and $\square \rightarrow \circ$)
- Importing ghosts cells from simbling grid;
- Apply the real boundary condition.

Ghosts cells

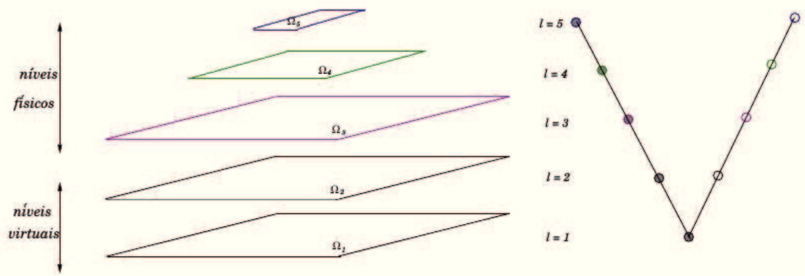


Linear system on adaptive mesh refinement





Linear system on adaptive mesh refinement



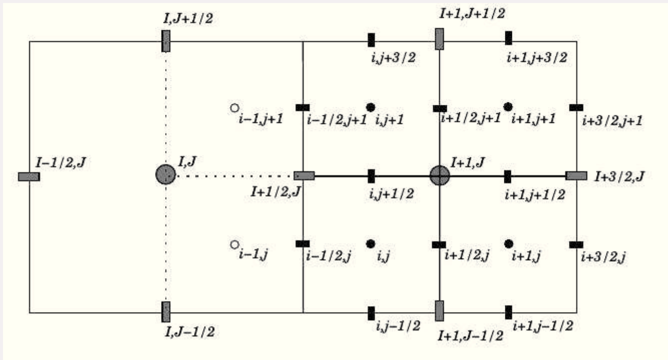
Based on restriction and interpolation process to a sequential of coarser meshes.

Linear system on adaptive mesh refinement

Flux correction

$$\int_{\Omega} \nabla \cdot \mathbf{f} = \int_{\delta\Omega} \mathbf{f} \cdot \mathbf{n},$$

$$h_l^2 \sum_{i,j} (D \cdot \mathbf{f})_{ij} = h_l \sum_{m} (\mathbf{f} \cdot \mathbf{n})_m,$$



Linear system on adaptive mesh refinement

Flux correction

$$\sum(D\cdot\mathbf{f})h_l^2 = \frac{f_{x_{I+\frac{1}{2},J}} - f_{x_{I-\frac{1}{2},J}}}{h_l}h_l^2 + \frac{f_{y_{I,J+\frac{1}{2}}} - f_{y_{I,J-\frac{1}{2}}}}{h_l}h_l^2 +$$

$$\frac{f_{x_{i+\frac{1}{2},j}} - f_{x_{i-\frac{1}{2},j}}}{h_{l+1}}h_{l+1}^2 + \frac{f_{y_{i,j+\frac{1}{2}}} - f_{y_{i,j-\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2 +$$

$$\frac{f_{x_{i+\frac{1}{2},j+1}} - f_{x_{i-\frac{1}{2},j+1}}}{h_{l+1}}h_2^2 + \frac{f_{y_{i,j+\frac{3}{2}}} - f_{y_{i,j+\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2 +$$

$$\frac{f_{x_{i+\frac{3}{2},j+1}} - f_{x_{i+\frac{1}{2},j+1}}}{h_{l+1}}h_{l+1}^2 + \frac{f_{y_{i+1,j+\frac{3}{2}}} - f_{y_{i+1,j+\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2 +$$

$$\frac{f_{x_{i+\frac{3}{2},j}} - f_{x_{i+\frac{1}{2},j}}}{h_{l+1}}h_{l+1}^2 + \frac{f_{y_{i+1,j+\frac{1}{2}}} - f_{y_{i+1,j-\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2,$$

$$\sum(D\cdot\mathbf{f})h_l^2 = \left[\left(\frac{f_{y_{i,j+\frac{3}{2}}} + f_{y_{i+1,j+\frac{3}{2}}}}{2}\right) - \left(\frac{f_{y_{i,j-\frac{1}{2}}} + f_{y_{i+1,j-\frac{1}{2}}}}{2}\right) +\right.$$

$$\left.\left(\frac{f_{x_{i+\frac{3}{2},j+1}} + f_{x_{i+\frac{3}{2},j}}}{2}\right) - f_{x_{I-\frac{1}{2},J}} + f_{y_{I,J+\frac{1}{2}}} - f_{y_{I,J-\frac{1}{2}}}\right]h_l,$$

Linear system on adaptive mesh refinement

Flux correction

$$\sum(D\cdot\mathbf{f})h_l^2 = \frac{f_{x_{I+\frac{1}{2},J}} - f_{x_{I-\frac{1}{2},J}}}{h_l}h_l^2 + \frac{f_{y_{I,J+\frac{1}{2}}} - f_{y_{I,J-\frac{1}{2}}}}{h_l}h_l^2 +$$

$$\frac{f_{x_{i+\frac{1}{2},j}} - f_{x_{i-\frac{1}{2},j}}}{h_{l+1}}h_{l+1}^2 + \frac{f_{y_{i,j+\frac{1}{2}}} - f_{y_{i,j-\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2 +$$

$$\frac{f_{x_{i+\frac{1}{2},j+1}} - f_{x_{i-\frac{1}{2},j+1}}}{h_{l+1}}h_2^2 + \frac{f_{y_{i,j+\frac{3}{2}}} - f_{y_{i,j+\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2 +$$

$$\frac{f_{x_{i+\frac{3}{2},j+1}} - f_{x_{i+\frac{1}{2},j+1}}}{h_{l+1}}h_{l+1}^2 + \frac{f_{y_{i+1,j+\frac{3}{2}}} - f_{y_{i+1,j+\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2 +$$

$$\frac{f_{x_{i+\frac{3}{2},j}} - f_{x_{i+\frac{1}{2},j}}}{h_{l+1}}h_{l+1}^2 + \frac{f_{y_{i+1,j+\frac{1}{2}}} - f_{y_{i+1,j-\frac{1}{2}}}}{h_{l+1}}h_{l+1}^2,$$

$$f_{x_{I+\frac{1}{2},J}} = \frac{f_{x_{i-\frac{1}{2},j}} + f_{x_{i-\frac{1}{2},j+1}}}{2}.$$

Multigrid-Multilevel Algorithm

for $l = l_{top}$ *a* 1 **do**
  if $l = l_{top}$ **then**
    $e^{l_{ltop}} = 0$
    Calcule $L(\overline{\phi})$, em $\Omega^{l_{top}}$
    $R^{l_{top}} \leftarrow B^{l_{top}} - L(\overline{\phi})^{l_{top}}$ em $\Omega^{l_{top}}$
    $e^{l_{ltop}} \leftarrow RBGS(A^{l_{ltop}}, e^{l_{ltop}}, R^{l_{ltop}})$ em $\Omega^{l_{top}}$
  else
    $e_l = 0$
    Calcule $L(\overline{\phi})^l$, em Ω^l
    Calcule $L(e)^l$, em $\delta\Omega^{l+1}$
    $R^l \leftarrow B^l - L(\overline{\phi})^l$, em $\Omega^l - \Omega^{l+1}$
    $R^l \leftarrow \mathscr{R}_l^{l+1}(R^l - L(e^l))$, em $P(\Omega_l^{l+1})$
    $e^l \leftarrow RBGS(A^l, e^l, R^l)$ em Ω^l
  end if
end for

Interface Modeling

Method	Pros	Cons
Level Set	Conceptually simple Easy implementation	Limited precision Non conservative
Shock Capture	Easy implementation Multiple advective Schemes available	Numerical diffusion Requires fine meshes
Marker Particle	Extremely accurate Robust Can handle great topological changes	High computational cost Marker particles must be redistributed
SLIC VOF	Conceptually simple Easy extension to 3D	Numeric diffusion Limited precision Artificial fragmentation and coalescence
PLIC VOF	Relatively simple Precise Supports great topological changes	Artificial fragmentation and coalescence
Lattice Boltzamnn	Precise Supports great topological changes	Difficult to implement Artificial fragmentation and coalescence
Front Tracking	Extremely Precise Robust Supports great topological changes No artificial coalescence or fragmentation	Requires mapping Requires dynamic remeshing

Volume of Fluid Method

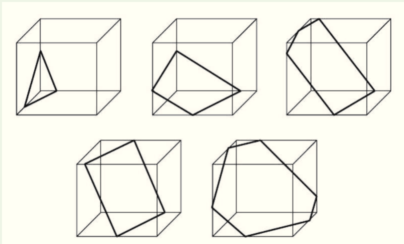
$$\frac{\partial C}{\partial t} + (\mathbf{u}\cdot\nabla)C = 0.$$
$$\mathbf{n}\cdot\mathbf{x} = n_x x + n_y y + n_z z = \alpha$$

- Interface reconstruction: the geometric parameters are evaluated, α e $\mathbf{n}$

$$Volume = \frac{1}{6n_1 n_2 n_3}\left[\alpha^3 - \sum_{i=1}^{3} F_3(\alpha - n_i \Delta x_i) + \sum_{i=1}^{3} F_3(\alpha - \alpha_{\max} + n_i \Delta x_i)\right]$$

- Interface advection:

$$n_x^{(*)} = \frac{n_x^{(n)}}{1 + A\Delta t + \frac{1}{2}A^2\Delta t^2}$$
$$\alpha^{(*)} = \alpha^{(n)} + \frac{n_x^{(n)}(\frac{1}{2}AB\Delta t^2 + B\Delta t)}{1 + A\Delta t + \frac{1}{2}A^2\Delta t^2}$$
$$A = (U_R - U_L)/\Delta x,\ B = U_L$$



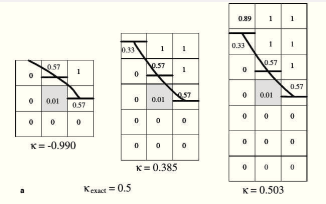
Volume of Fluid Method

Surface tension

$$\mathbf{F}_\sigma = \sigma\kappa\delta\mathbf{n} = \sigma\kappa\nabla C.$$

Curvature

- Height Function: second order method. It becomes inconsistent when the radius of curvature of the interface becomes comparable to the mesh size. $\kappa = \frac{h''}{(1+h'^2)^{3/2}}$



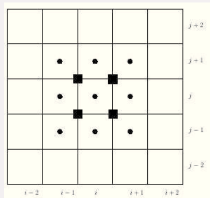
Volume of Fluid Method

Curvature

- Parabolic fitting: fit a curve, or mathematical function, that has the best fit to a series of data points.
  Fit a parabola (paraboloid in 3D) by minimising $F(a_i) \equiv \sum_{1 \leqslant j \leqslant n} [z_j^{'} - f(a_i, x_j^{'})]$
  with $f(a_i, x)] \equiv a_0 x^2 + a_1 y^2 + a_2 xy + a_3 X + a_4 y + a_5$
  $\kappa \equiv 2 \frac{a_0(1+a_4^2)+a_1(1+a_3^2)-a_2 a_3 a_4}{(1+a_3^2+a_4^2)^{3/2}}$
- Least Square: based on a least-squares fit of a Taylor series to determine the derivatives of the colour function field and the derivatives of the interface normal vector.
  Taylor series is developed for the colour function field around cell P with its neighbours Q:
  $\gamma_Q \equiv \gamma_P + \frac{\partial \gamma}{\partial x_i}|_P (x_{i,Q} - x_{i,P}) + \frac{\partial^2 \gamma}{\partial x_i \partial x_j}|_P (x_{i,Q} - x_{i,P})(x_{j,Q} - x_{j,P}) + O(\Delta x_i^3)$
  $A \cdot \phi = b$
  $\kappa = -\frac{\partial m_i}{\partial x_i}|_P$

Volume of Fluid Method

Curvature

- 27 cells/ Shirani's method...: Discretize the normal $\mathbf{n} = \frac{\nabla \mathbf{C}}{|\nabla \mathbf{C}|}$ with finite differences. 

$$\mathbf{n}_{ur} = \frac{1}{4}(\mathbf{n_{i,j}} + \mathbf{n_{i,j+1}} + \mathbf{n_{i+1,j}} + \mathbf{n_{i+1,j+1}})$$

$$\kappa = \frac{(\mathbf{n}_{ur}+\mathbf{n}_{lr})_x - (\mathbf{n}_{ul}+\mathbf{n}_{ll})_x}{2\Delta x} + \frac{(\mathbf{n}_{ul}+\mathbf{n}_{ur})_y - (\mathbf{n}_{ll}+\mathbf{n}_{lr})_y}{2\Delta y}$$

Front- Tracking

Equations

$$\delta F_\sigma = \oint_{\delta\Gamma} \sigma \mathbf{t} \times \mathbf{n} d\Gamma,$$

$$\mathbf{f}_\sigma(\mathbf{x},t) = \int \mathbf{F}_\sigma(\mathbf{X},t) D(\mathbf{X}-\mathbf{x}) d\mathbf{X},$$

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) = \nabla\cdot\left(\mu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)\right) - \nabla p + \rho\mathbf{g} + \mathbf{f}_\sigma,$$

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

$$\mathbf{U}(\mathbf{X},t) = \int \mathbf{u}(\mathbf{x},t) D(\mathbf{x}-\mathbf{X}) d\mathbf{x}$$

$$\frac{d\mathbf{X}(t)}{dt} = \mathbf{U}(\mathbf{X}(t),t).$$

Front- Tracking

Equations - Spreading and Interpolation

$$D(\mathbf{x}-\mathbf{X}) = \frac{1}{h_x h_y h_z} W\left(\frac{x-X}{h_x}\right) W\left(\frac{y-Y}{h_y}\right) W\left(\frac{z-Z}{h_z}\right),$$

where

$$W(r) = \begin{cases} \frac{1}{4}(1+cos(\frac{\pi}{2}r)), & r<2, \\ 0, & r \geq 2, \end{cases}$$

and

$$r = \frac{x-X}{h_x}, \frac{y-Y}{h_y}, \frac{z-Z}{h_z}.$$

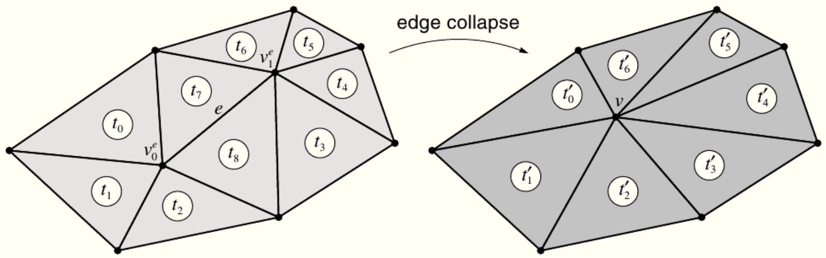
$$H(\varphi) = \begin{cases} 1, & \varphi > \gamma \\ 0.5(1+\frac{\varphi}{\gamma}+\frac{1}{\pi}sin(\frac{\pi\varphi}{\gamma})), & \|\varphi\| \leq \gamma \\ 0, & \varphi < -\gamma \end{cases}$$

$$\Psi(\varphi) = H(\varphi)\Psi_1 + (1-H(\varphi))\Psi_2$$

Lagrangian Interface

GTS - *GNU Triangulated Library*

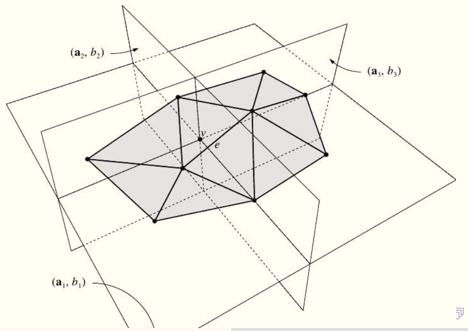
- Conservative Remeshing based on edge collapse.
  - Memoryless Polygon Simplification, Lindstrom and Turk (1999)
- Preserves geometry volume, area and shape; element quality



Lagrangian Interface

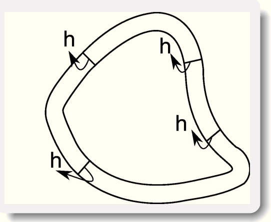
GTS - *GNU Triangulated Library*

- Conservative Remeshing based on edge collapse.
  - Memoryless Polygon Simplification, Lindstrom and Turk (1999)
- Preserves geometry volume, area and shape; element quality



VRA - Volume Recovery Algorithm

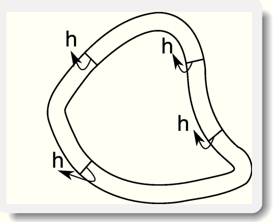
- Volume change is small and uniform over the whole surface



Front-Tracking

VRA - Volume Recovery Algorithm

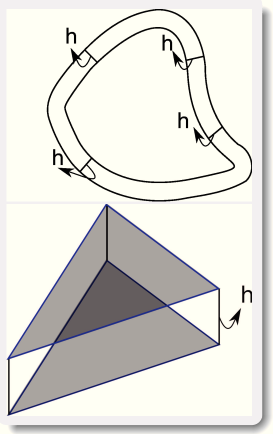
- Only at the normal direction



Front-Tracking

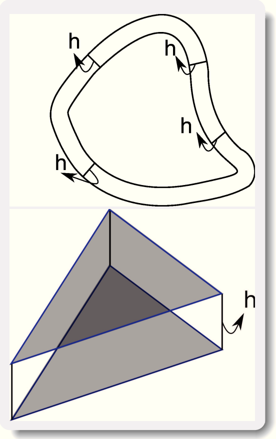
VRA - Volume Recovery Algorithm

- Volume of a given element: $V = A_T \cdot h$



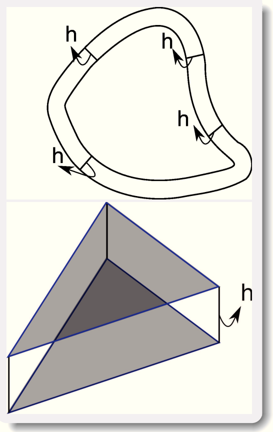
VRA - Volume Recovery Algorithm

- Integral over the surface:
$$\Delta V = \sum_{i=1}^{Nt} A_{T\,i} \cdot h = A_S \cdot h$$



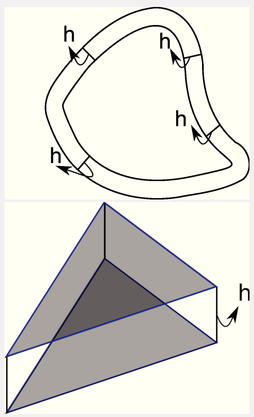
VRA - Volume Recovery Algorithm

- Volume change is known:
  $\Delta V = V_{atual} - V_{inicial}$



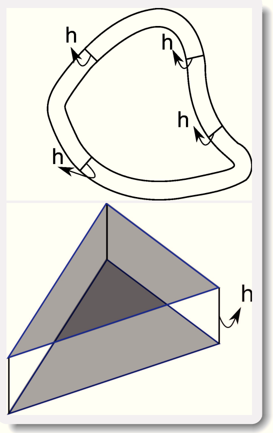
VRA - Volume Recovery Algorithm

- $h = \dfrac{\Delta V}{A_S}$



VRA - Volume Recovery Algorithm

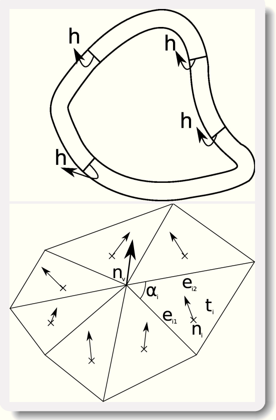
- $\mathbf{X}_{\mathbf{new}} = \mathbf{X}_{\mathbf{old}} - h \cdot \mathbf{n}_v$



Front-Tracking

VRA - Volume Recovery Algorithm

- $\mathbf{n}_v = \sum_{i=1}^{N_t} \frac{\mathbf{n}_i sin(\alpha_i)}{\|e_{i1}\|\|e_{i2}\|}$



Immersed Boundary

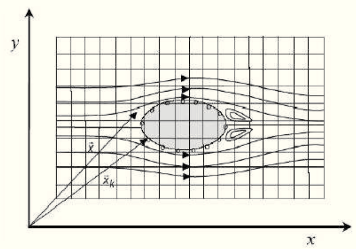
Multi-Direct Forcing

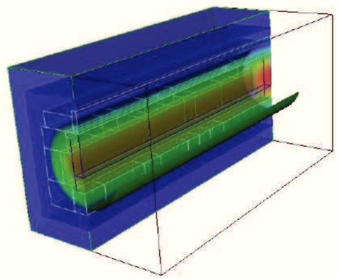
$$\mathbf{U}^* = \sum D(\mathbf{x}-\mathbf{X})\mathbf{u}^*(\mathbf{x})$$

$$\frac{\mathbf{F}(\mathbf{X})}{\rho(\mathbf{X})} = q(\mathbf{X}) = \frac{\alpha_2(\mathbf{U}_\Gamma - \mathbf{U}^*)}{\Delta t}$$

$$\mathbf{f}(\mathbf{x}) = \sum D(\mathbf{x}-\mathbf{X})q(\mathbf{X})$$

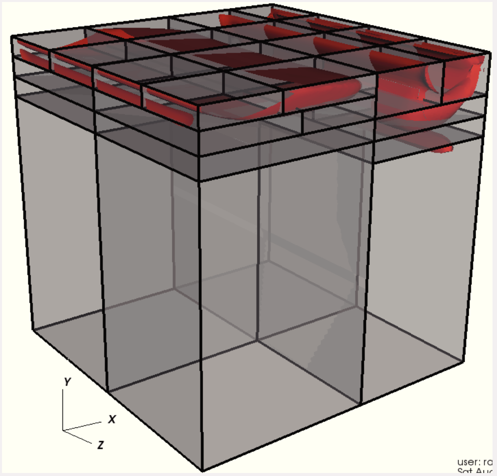
$$\mathbf{u}^* = \mathbf{u}^* + \frac{\Delta t \mathbf{f}(\mathbf{x})}{\alpha_2}$$





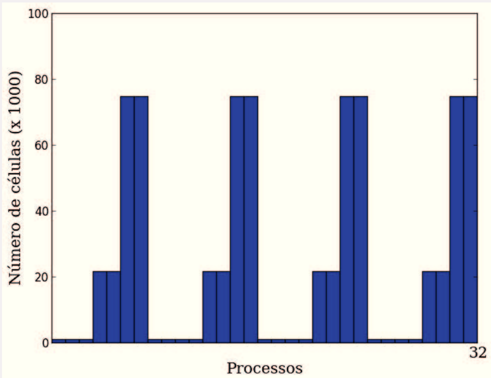
Simulated examples in AMR3d code

One phase flow: driven cavity at $Re = 1000$



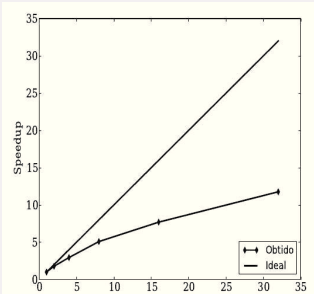
Simulated examples in AMR3d code

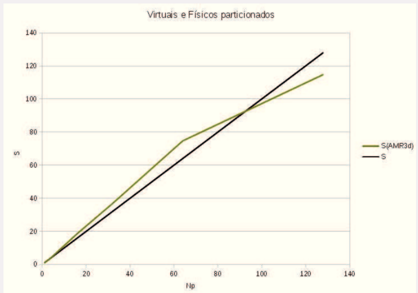
One phase flow: load balancing



Simulated examples in AMR3d code

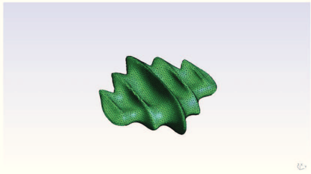
Speedup



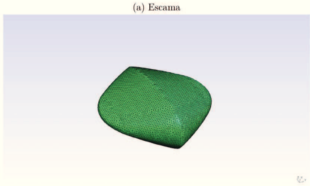


Simulated examples in AMR3d code

One phase flow: wake downstream of a shark scale



(a) Escama



(b) Referência

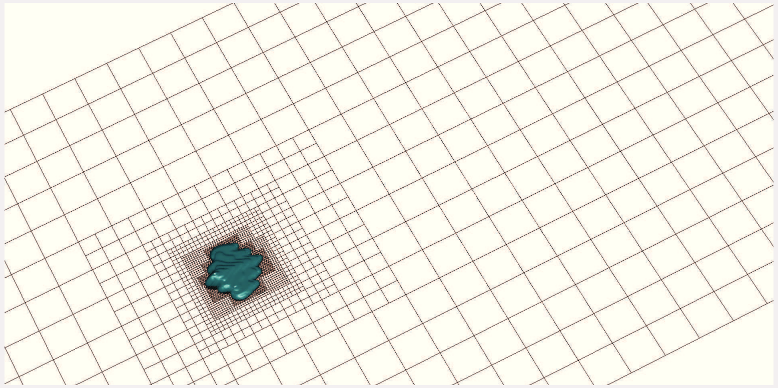
Coeficiente de Arrasto (C_d)			
Re	Referência	Escama	%
10^2	1.8035	1.9448	$+7,83$
10^3	0.7252	0.7490	$+3,28$
10^4	0.6467	0.6103	$-5,63$
10^5	0.6276	0.6052	$-3,57$
10^6	0.6286	0.6070	$-3,44$

Simulated examples in AMR3d code

One phase flow: wake downstream of a shark scale

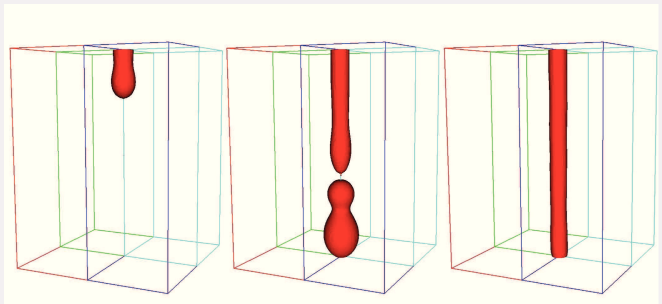
Simulated examples in AMR3d code

One phase flow: wake downstream of a shark scale



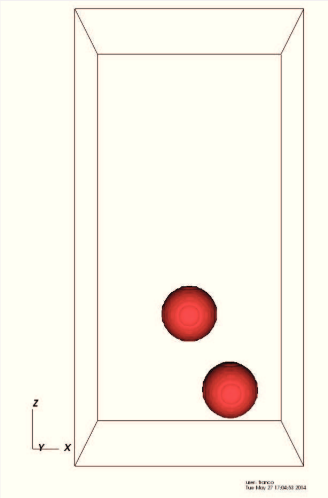
Simulated examples in AMR3d code

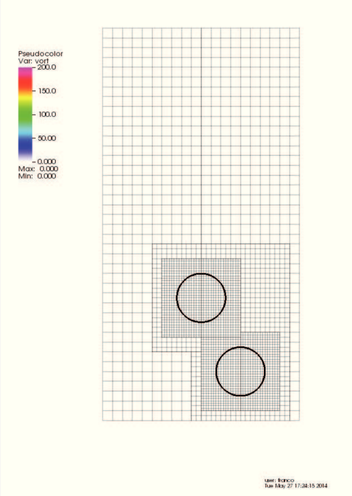
Two phases flow: fluid-film formation



Simulated examples in AMR3d code

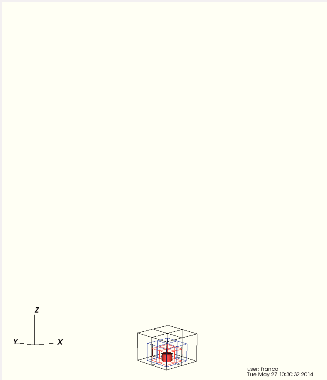
Two phases flow: bubble ascending with VoF

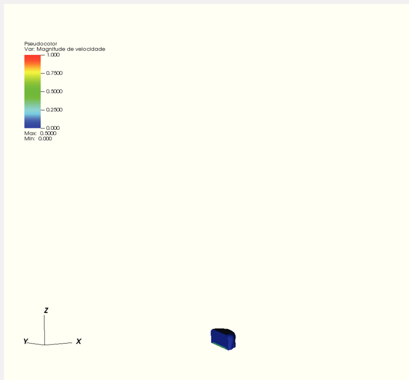




Simulated examples in AMR3d code

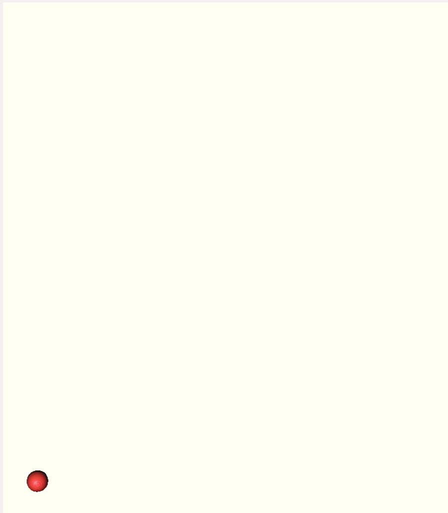
Two phases flow: bubble detachment with VoF





Simulated examples in AMR3d code

Two phases flow: wobbling with VoF

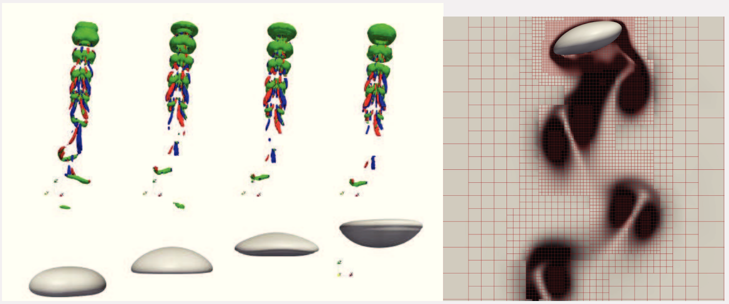


Simulated examples in AMR3d code

Two phases flow: wobbling with VoF

Simulated examples in AMR3d code

Two phases flow: bubble ascending with FT

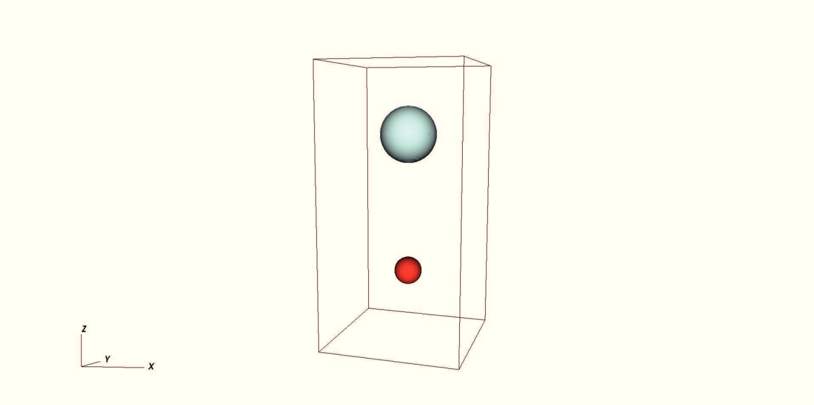


Simulated examples in AMR3d code

Two phases flow : bubble ascending with FT

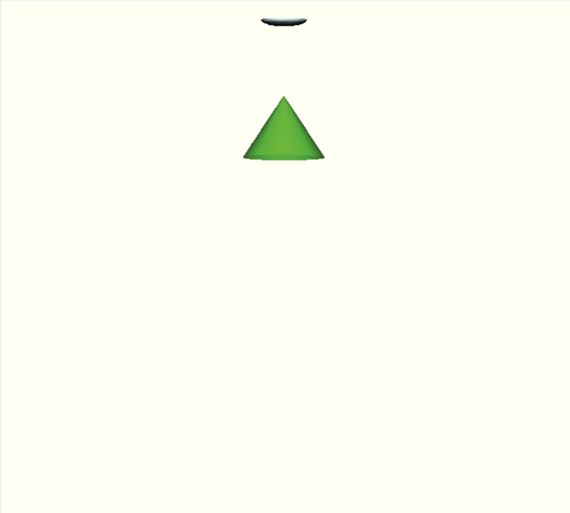
Simulated examples in AMR3d code

Two phases flow : Drop impact (VoF+IB)



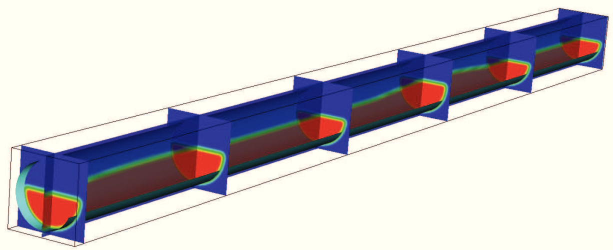
Simulated examples in AMR3d code

Two phases flow : film fluid fragmentation (VoF+FI)



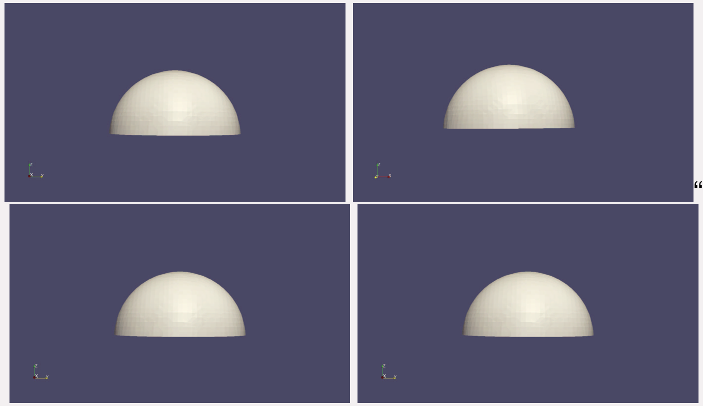
Simulated examples in AMR3d code

Two phases flow : Oil/Water Dispersion (VoF+FI)



Simulated examples in AMR3d code

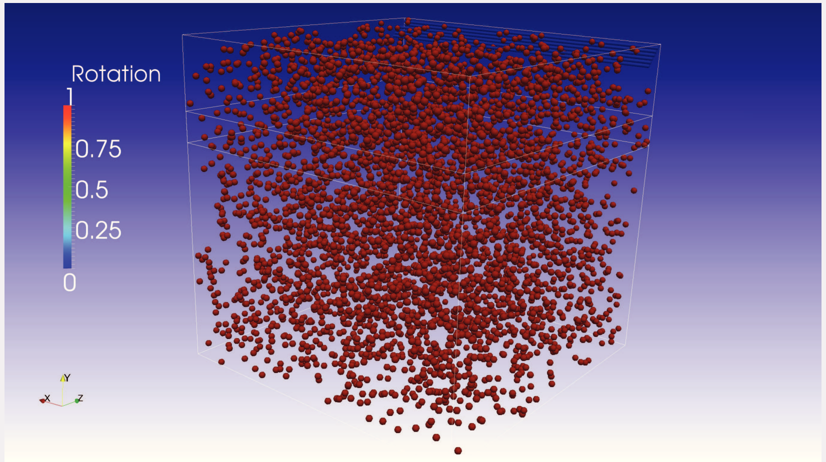
Two phases flow : Triple contact angle $\theta = 30, 60, 90$ e 150



“

Simulated examples in AMR3d code

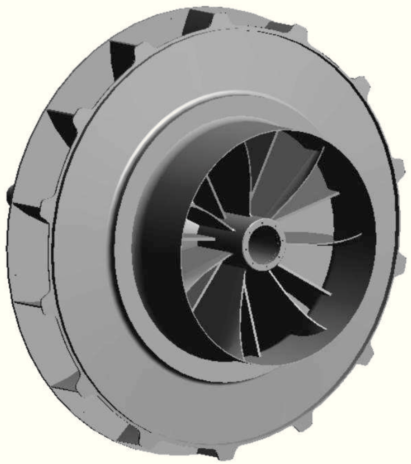
Solid-gas flow : fluid particles



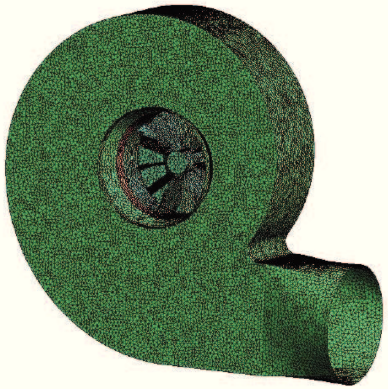
Flow in fans

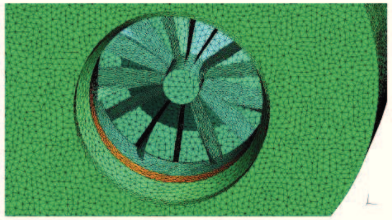
General data

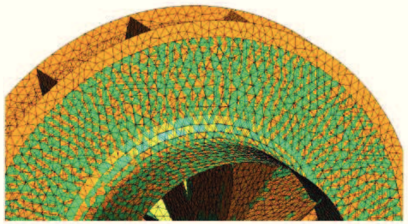
- Volumetric flow 60 T/h, ou 16 Nm^3/s with rotation of 1650 RPM;
- Work temperature 69.2 ^oC;
- Density: $\rho = 0.5842\,kg/m^3$;
- Viscosity: 8.659×10^{-6} Pa.s;
- Maximum reynolds number, based in the inlet boundary condition $\approx$ 2.2 milhões;
- Maximum reynolds number, based in rotor speed $\approx$ 36.85 milhões;



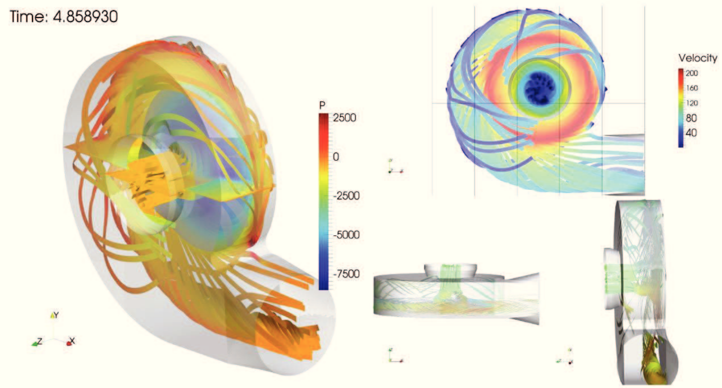
Flow in fans



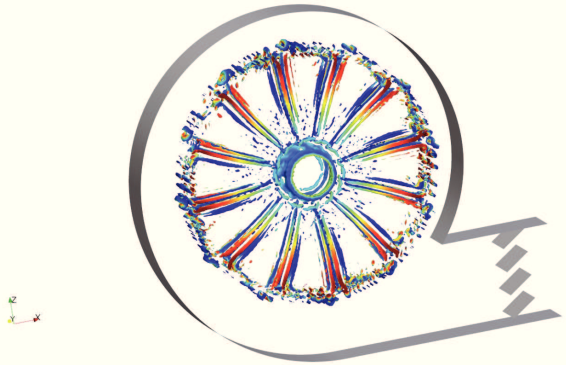




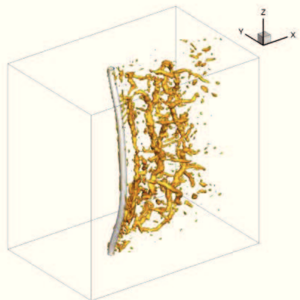
Flow in fans

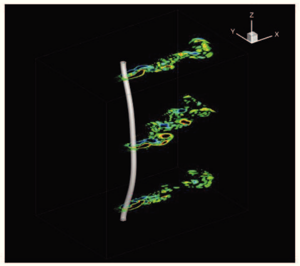

Time: 4.858930
P
2500
0
-2500
-5000
-7500
Y
Z
X
Velocity
200
160
120
80
40

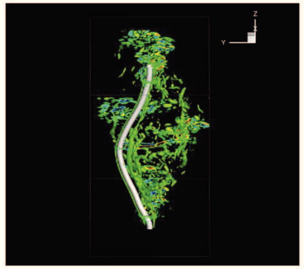
Flow in fans



Fluid structure interaction example







Numerical features that are in progress

- Flow with high physical ratio;
- Solvers with best speedup;
- Isothermic flows;
- Contact triple modeling in complex geometries;
- Euler-lagrange modeling to droplets transport;
- Flows in presence of mobile complex geometries;
- Turbulence Modeling;
- Graphic interface.

Acknowledgements

- PETROBRAS
- CNPq
- Fapemig
- FEMEC/UFU