Multiphase flow in *Code_Saturne* using Level-Set method

Quentin ROLLAND [1*], Clément JAUSE-LABERT [1]

[1] ABMI, Research & Innovation Department, Lyon office

[*] Corresponding author: q.rolland@abmi-groupe.com

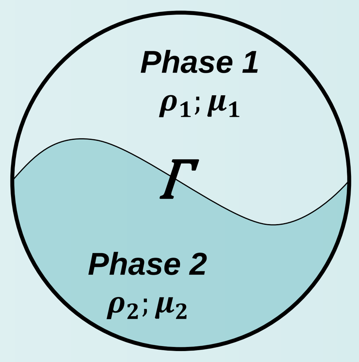
ABSTRACT

In the present work, a two-phase flow algorithm is proposed for the *Code_Saturne* software. Multiphasic simulation is more and more required for various applications in the field of fluid mechanics, such as micro-fluidics or liquid jets, among others. There are several methods currently used in CFD softwares, among which Volume Of Fluid (VOF) and Level-Set (LS) methods are the most common. The first presents the benefit of being mass-conservative but is not well suited for interface capturing. The latter method however introduces a signed distance function, from which a more precise interface between the two fluids can be computed.

A first approach consists in introducing a Level-Set method in *Code_Saturne* source code. Due to the moving interface, a re-initialization algorithm for the signed function had to be introduced. In order to reduce mass loss during the calculation process, a high order scheme (WENO5) has been implemented in a square lattice mesh. However, a orthogonal mesh does not match general purpose models requiring, depending on the geometry, different mesh types.

To overcome this problem, a second approach consists in implementing new algorithms so as to prevent mass loss by using the same scheme order used in *Code_Saturne* for any mesh type. Recent work has improved the accuracy of the distance function: for example, a Conservative Level-Set (CLS) is introduced to minimize the re-initialization effort and compared with previous methods regarding its mass conservation. We finally end-up with perspectives of improvement to have a fully functional algorithm in the coming future for *Code_Saturne*.

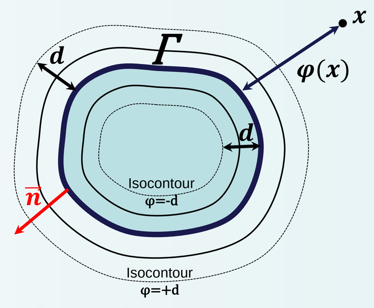
LEVEL-SET METHOD



- **Precise position of the interface**
  - Physical properties interface jump
  - Surface tension: Continuum Surface Force (CSF)

$$F_\sigma = \int_\Omega \sigma\kappa(\varphi)\,\nabla\varphi\,\delta(\varphi)\,d\Omega$$

- **Interface capturing method**
  - Scalar function level-set: φ
  - Signed distance function



$$\begin{cases} \varphi(x,t) = 0, x \in \Gamma \\ \varphi(x,t) < 0, x \in \Omega_1 \\ \varphi(x,t) > 0, x \in \Omega_2 \end{cases}$$

- **Re-initialization of the level-set method:**
  - The signed distance has the following property: $|\nabla d| = 1$
  - Re-initialization function of method function:

$$\frac{\partial\varphi}{\partial\tau} + \mathrm{sign}(\varphi_0)(|\nabla\varphi| - 1) = 0, \qquad \varphi_0 = \varphi(\tau = 0, x)$$

RESULTS

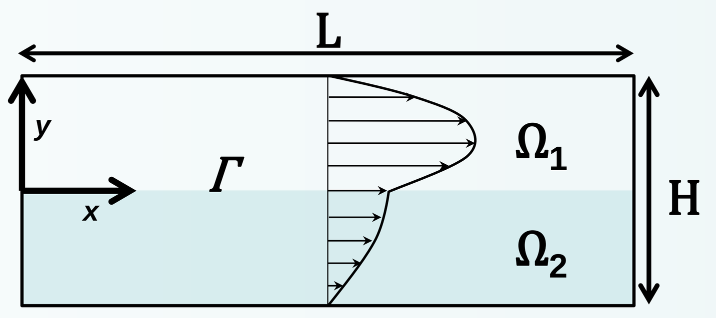
Results with high order discretization scheme (WENO5) Grid meshes

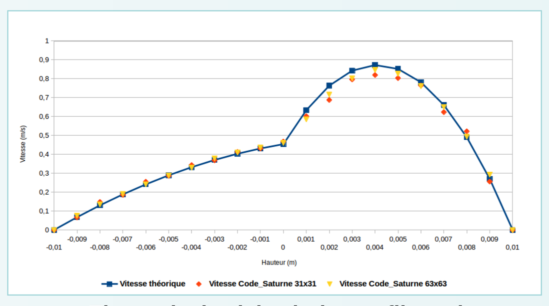
- **Results computed in *Code_Saturne***

SALOME — CODE SATURNE — ParaView

- **Numerical results and validation from the literature :**
  - **Poiseuille two phase flow test**
    - ➢ Analytical results compared with numerical results
    - ➢ Validation of physical property changes

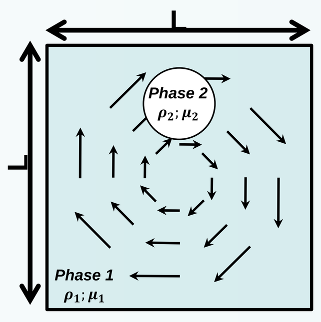
$$\rho(\varphi) = \rho_1 + (\rho_2 - \rho_1)H(\varphi)$$
$$\mu(\varphi) = \mu_1 + (\mu_2 - \mu_1)H(\varphi)$$
with the heavyside function $H(\varphi)$

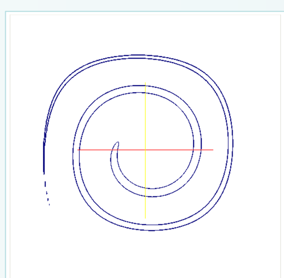




Theoretical axial celerity profile and Code_Saturne numerical results

  - **Vortex test**
    - ➢ Validation of Level-Set scalar transport

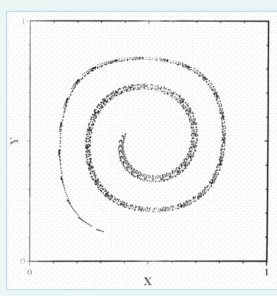




Code_Saturne computing results for t=3s

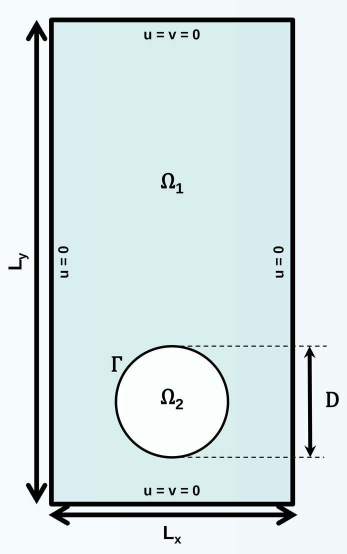
Discretization scheme	Mass loss (%)	Shape recognition
Upwind 1 – Euler 1	100%	No
Quick 3 – Euler 1	65,9%	Bad
Quick 3– RK 2	17,9%	Good
WENO 5 – Adams-Bashforth 2	6,6%	Very good

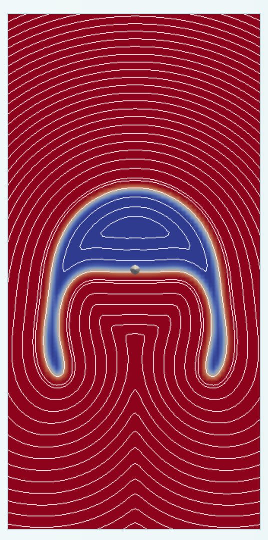
Mass loss with different discretization scheme



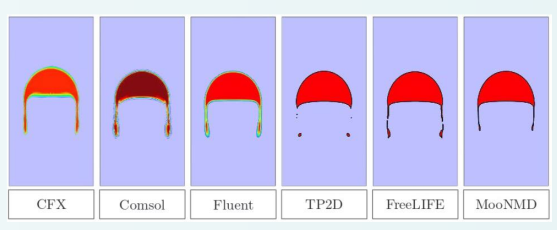
« Exact solution » for t = 3s (PhD theses Tanguy, 2004)

  - **Rising bubble test**
    - ➢ Validation of the re-initialization function





Rising bubble computed with Code_Saturne



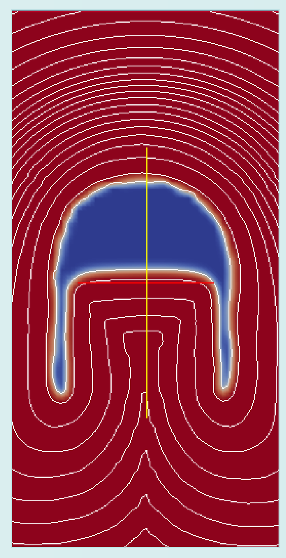
Rising bubble computation from various CFD software [2008 : Quantitative benchmark computations of two-dimensional bubble dynamics. MOX Report, Politecnico di Milano]

Results with Centered scheme (*Code_Saturne*) Any mesh type

- **Level-Set method**
  - Re-initialization function with small time steps
  - Modification of the re-initialization function during the iteration loop

$$\frac{\partial\varphi}{\partial\tau} = \pm\mathrm{sgn}(\varphi_0)(1 - |\nabla\varphi|)$$

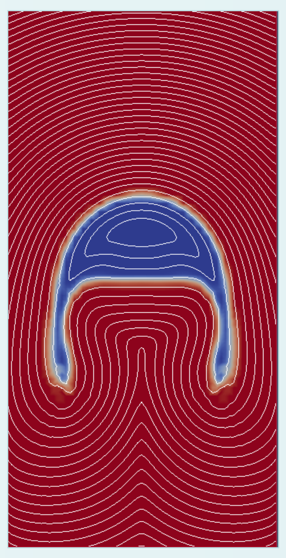
➢ 5% of mass loss
➢ 600~700 iterations every 10 time steps
➢ Bubble shape changed



- **Level-Set method with diffusion**
  [Implementation of a standard level set method for incompressible two-phase flow simulations, N. Johansson]
  - Adding a diffusion term

$$\frac{\partial\varphi}{\partial\tau} = \mathrm{sgn}(\varphi_0)(1 - |\nabla\varphi|) + \alpha\nabla^2\varphi$$
$$\alpha = \frac{\max(\Delta x, \Delta y)}{2}$$

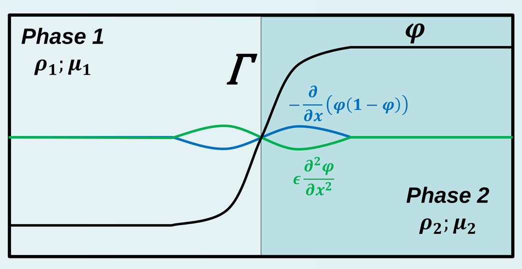
➢ 5% of mass loss
➢ 4~10 iterations every 10 time steps
➢ No precise interface



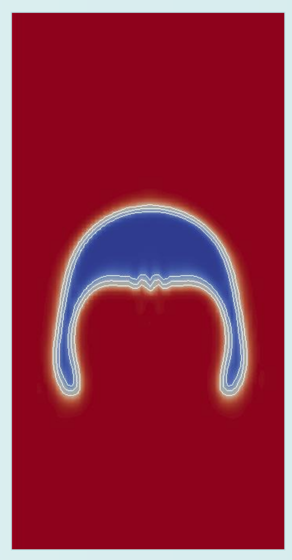
- **Conservative Level-Set method (CLS)**
  [J of Computational Physics 225 (2007) 785-807, E. Olsson]
  - Adding a compression term to balance the diffusion term
  - Initialization for φ=0,5

$$\varphi = \frac{1}{2}\left(1 + \tanh\left(\frac{x}{2\varepsilon}\right)\right)$$

➢ 10% of mass loss, artificial oscillations



$$\frac{\partial\varphi}{\partial\tau} = \varepsilon\nabla\cdot(\nabla\varphi) - \nabla(\varphi(1-\varphi)\hat{n})$$
$$\hat{n} = \frac{\nabla\varphi(x,\tau=0)}{|\nabla\varphi(x,\tau=0)|}$$



CONCLUSIONS AND PERSPECTIVES

- **Algorithms with low order discretization scheme show convincing results but are still work in progress**
- **Up-coming improvement of the re-initialization function:**
  - ➢ **Improvement of the CLS method algorithm (with helps of CS internal functions)**
  - ➢ **Development of the VOSET method (coupled Volume-Of-Fluid and Level-Set methods) to prevent mass loss**