A sharp cartesian method for the simulation of air-water interface

Lisl Weynans, Michel Bergmann, Francky Luddens

Institut de Mathématiques de Bordeaux & INRIA Bordeaux-Sud Ouest

Numerical Analysis Seminar
Texas A&M University
Sept. 11th, 2014

Motivation: sharp simulation of air-water interface

- Starting point:
  - NaSCar: a 3D parallel incompressible code with fluid-solid interaction
  - a second order cartesian method to solve elliptic problems with discontinuities across interfaces

  ⇒adaptation of this method to solve accurately the pressure at the interface between fluids with strong density ratios

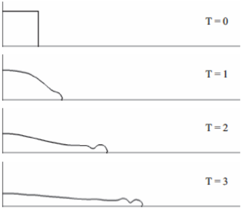
- Our aim: simulation of solids interacting with fluids with strong density ratios (air-water interface)
  - wave breaking
  - marine engineering
  - wave energy converters



Bibliography

Only few sharp methods on cartesian grids in this context:

- Kang, Fedkiw and Liu 2000:
  application of the famous "Ghost Fluid Method",
  pioneering work, but non-physical effects due to poor momentum preservation for each fluid
- Raessi and Pitsch 2012: "cut-cell"-like method
- Zhou et al 2012: only for fixed interface



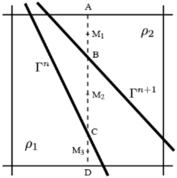
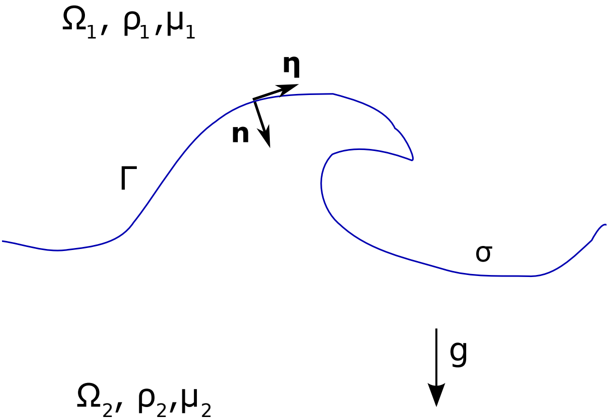


Figure 3: The phase interface Γ at time steps n and $n+1$ intersecting with a flux surface (dashed line).

Figure: Left: non physical behavior for dam-break problem (Kang et. al.), right: "cut-cell" interface reconstruction (Raessi and Pitsch)


Ω1, ρ1,μ1
η
n
Γ
σ
g
Ω2, ρ2,μ2

Fluid model

- Incompressible Navier-Stokes equations in Ω_1 and Ω_2 :

$$\rho(\boldsymbol{u}_t + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u}) = -\nabla p + (\nabla.\tau)^T + \rho \boldsymbol{g},$$
$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0$$

- Continuity of velocity and velocity divergence:

$$[u] = [v] = 0,$$
$$[(u_n, v_n).\boldsymbol{n}] = 0.$$

- Jump conditions on Γ:

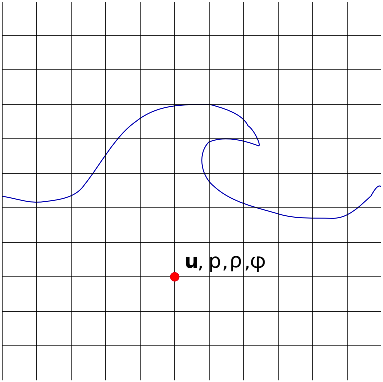
$$[\mu(u_n, v_n).\boldsymbol{\eta} + \mu(u_\eta, v_\eta).\boldsymbol{n}] = 0,$$
$$[p] = \sigma\kappa + 2[\mu](u_n, v_n).\boldsymbol{n}.$$

- Material derivative of the velocity continuity

$$[\frac{\nabla p}{\rho}] = [\frac{(\nabla.\tau)^T}{\rho}].$$

Discretization

- In the fluid, all variables on the same grid points
- Possible corrective term to avoid parasitic modes of the pressure



Numerical scheme in the fluid

Predictor-corrector scheme:

- Prediction (we take $p = 0$)

$$\frac{\boldsymbol{u}^* - \boldsymbol{u}^n}{\Delta t} = -[(\boldsymbol{u} \cdot \nabla)\boldsymbol{u}]^{n+\frac{1}{2}} + \frac{(\nabla.\tau^n)^T}{\rho} - \boldsymbol{g}$$

- Resolution of ellliptic equation

$$\nabla \cdot (\frac{1}{\rho} \nabla p) = \frac{\nabla \cdot \boldsymbol{u}^*}{\Delta t}$$

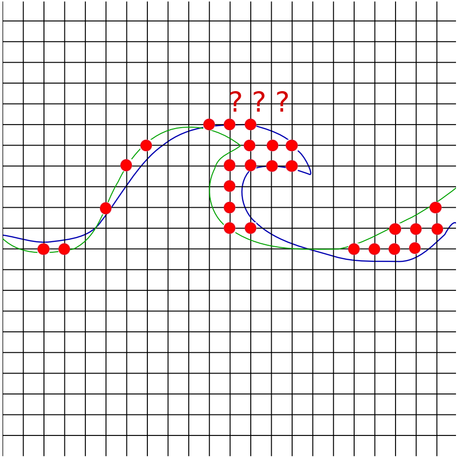
- Correction

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^* - \frac{\Delta t}{\rho} \nabla p$$

Why do we take $p = 0$ in prediction step?

$$[p] = \sigma\kappa + 2[\mu](u_n, v_n).\boldsymbol{n}.$$

$\Rightarrow$ p possibly discontinuous if the interface crosses a grid point during Δt



Numerical scheme in the fluid

- Prediction

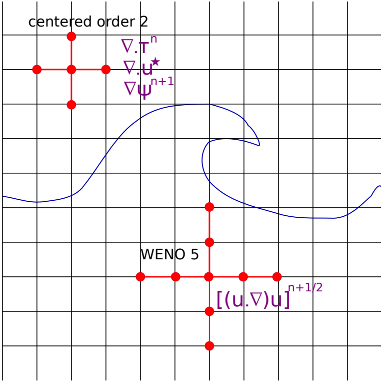
$$\frac{\boldsymbol{u}^* - \boldsymbol{u}^n}{\Delta t} = -[(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}]^{n+\frac{1}{2}} + \frac{(\nabla.\tau^n)^T}{\rho} - \boldsymbol{g}$$

- Elliptic equation:

$$\nabla\cdot(\frac{1}{\rho}\nabla p) = \frac{\nabla\cdot\boldsymbol{u}^*}{\Delta t}$$

- Correction :

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^* - \frac{\Delta t}{\rho}\nabla p$$



Discretization near the interface

- Prediction

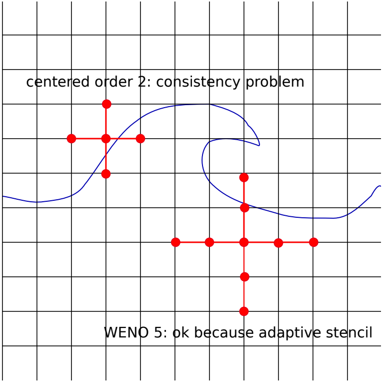
$$\frac{\boldsymbol{u}^* - \boldsymbol{u}^n}{\Delta t} = -[(\boldsymbol{u} \cdot \nabla)\boldsymbol{u}]^{n+\frac{1}{2}} + \frac{(\nabla.\tau^n)^T}{\rho} - \boldsymbol{g}$$

• Diffusion:
derivative of velocity discontinuous

$\Rightarrow$ lack of consistency

• Convection:
WENO 5 naturally adaptative,
continuous velocity

$\Rightarrow$ less worrying



Discretization near the interface

- Elliptic equation:

$$\nabla \cdot (\frac{1}{\rho}\nabla p) = \frac{\nabla \cdot \boldsymbol{u}^*}{\Delta t}$$

Discontinuity of ρ, jump conditions

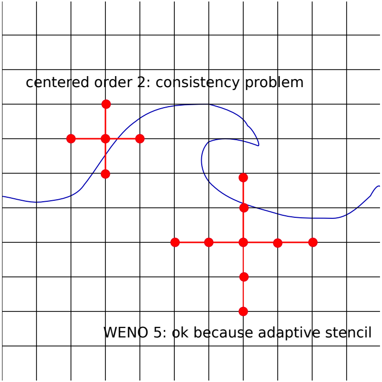
$\Rightarrow$ lack of consistency

- Correction :

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^* - \frac{\Delta t}{\rho}\nabla p$$

Discontinuity of ρ and ψ

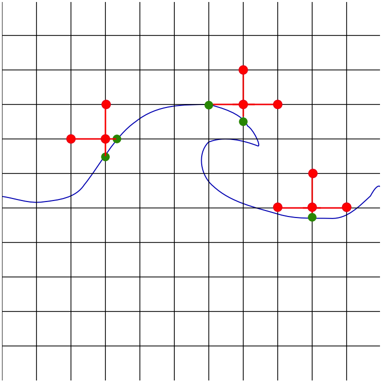
$\Rightarrow$ lack of consistency



Discretization near the interface

Our solution:

- Creation of additional unknowns for u^* and p on the interface, used for a sharp resolution of the pressure
- Regularization of μ and ρ only to account for viscous terms
  $\rightarrow$ no discontinuity any more in viscous terms



Discretization near the interface

- Elliptic problem
  - In the fluid:

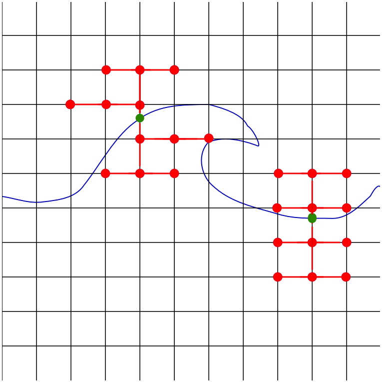
$$\nabla \cdot (\frac{1}{\rho}\nabla p) = \frac{\nabla \cdot \boldsymbol{u}^*}{\Delta t}.$$

(extrapolation of grid values for u^*)

  - On the interface:

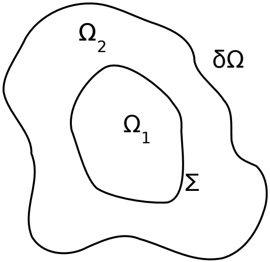
$$[p] = \sigma\,\kappa,$$

$$[\frac{\nabla p}{\rho}] = 0.$$

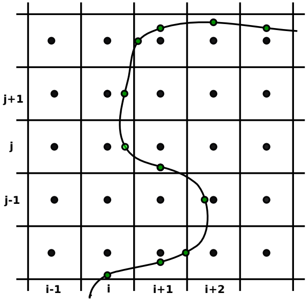


Interlude: elliptic problems with immersed interfaces

$$
\begin{aligned}
\nabla.(k\nabla u) &= f \text{ on } \Omega = \Omega_1 \cup \Omega_2 \\
[\![u]\!] &= \alpha \text{ on } \Sigma \\
[\![k\frac{\partial u}{\partial n}]\!] &= \beta \text{ on } \Sigma \\
u &= g \text{ on } \delta\Omega
\end{aligned}
$$



Discretization strategy



Creation of additional unknowns on the interface:

- used to discretize the elliptic operator on each side of the interface
- obtained by a discretization of jump conditions across the interface

Which accuracy is needed near the interface?

To obtain a second order convergence (in max norm), we need:

- a first-order truncation error for the elliptic operator near the interface
  $\Rightarrow$ avoid linear extrapolations (for instance: ghost-cell like)
- a second-order truncation error of the fluxes discretization
  $\Rightarrow$ use of an expanded stencil

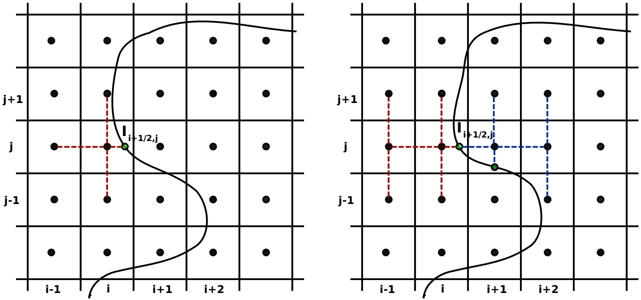
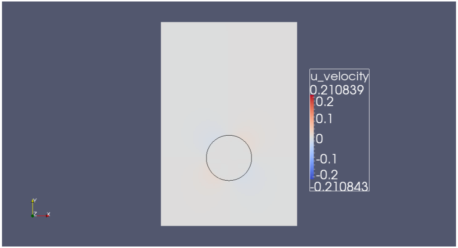
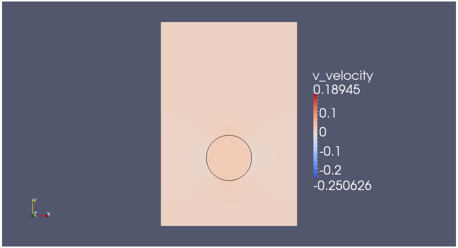


Figure: Examples of stencils for the discretization of elliptic operator and fluxes on each side of the interface.

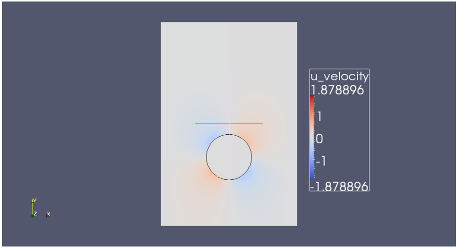
Rising of a small air bubble in water

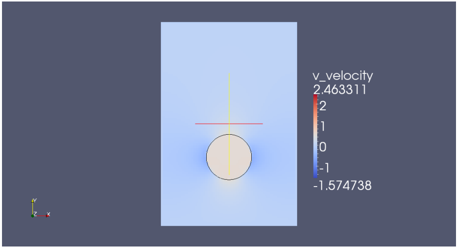




Water: $\rho = 1000$ kg$/m^3$, $\mu = 1,137.10^{-3}$ kg/ms,
Air: $\rho = 1$kg$/m^3$, $\mu = 1,78.10^{-5}$ kg/ms,
$\sigma = 0.0728$ kg$/s^2$, bubble radius $1/300$ m, $Tf = 0.05s$.

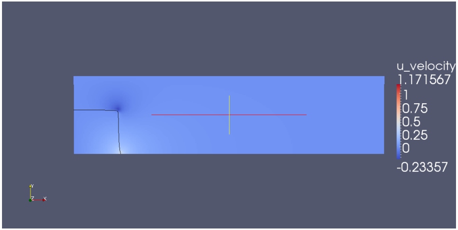
Rising of a large air bubble in water

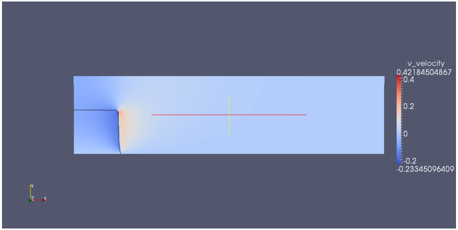




$$\text{Water: } \rho = 1000 \text{ kg}/m^3,\ \mu = 1,137.10^{-3} \text{ kg/ms},$$
$$\text{Air: } \rho = 1\text{kg}/m^3,\ \mu = 1,78.10^{-5} \text{ kg/ms},$$
$$\sigma = 0.0728 \text{ kg}/s^2, \text{ bubble radius } 1/3 \text{ m},\ Tf = 0.5s.$$

Dam break problem



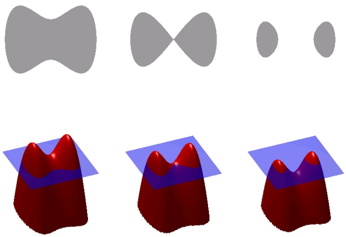


Water: $\rho = 1000 \text{ kg}/m^3$, $\mu = 1,137.10^{-3}$ kg/ms,
Air: $\rho = 1,226\text{kg}/m^3$, $\mu = 1,78.10^{-5}$ kg/ms,
$\sigma = 0.0728 \text{ kg}/s^2$, height of water column $h = 5.715$ cm, domain 40×10 cm

Sharp capture of the interface: level-set approach

- Γ captured as $\{\phi = 0\}$ with ϕ defined over Ω,
- easily handle complex geometries,
- straightforward computation of geometric properties:

$$\boldsymbol{n} = \nabla\phi, \qquad \kappa = \nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right),$$



- transport of Γ with the flow $\boldsymbol{u}$,
- in practice, ϕ is the signed distance to Γ ($\Rightarrow |\nabla\phi| = 1, \;\; \kappa = \Delta\phi$).

Reminder: elliptic problem with jumps on the interface

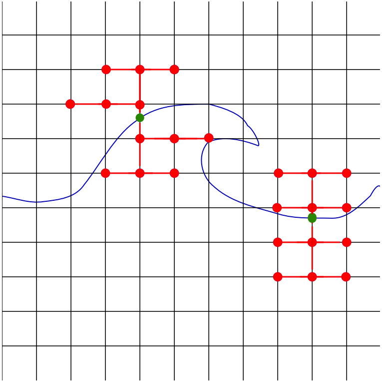
- Elliptic problem
  - In the fluid:

$$\nabla \cdot (\frac{1}{\rho} \nabla p) = \frac{\nabla \cdot \boldsymbol{u}^*}{\Delta t}.$$

(extrapolation of grid values for u^*)

  - On the interface:

$$[p] = \sigma \kappa,$$
$$[\frac{\nabla p}{\rho}] = 0.$$

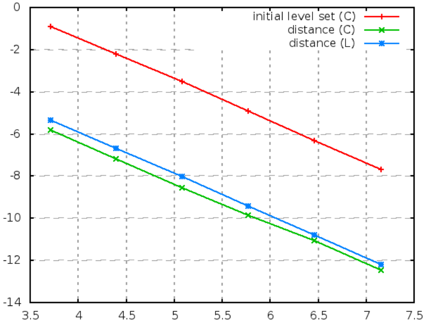


$\rightsquigarrow$ **consistent (at least first-order) scheme for κ required.**

Why using the distance function?

Objective: ensure consistency for the computation of κ AND low amplitude of the error.

- Example of computation of the curvature of a circle (second order FD formulas):
- with a level-set function with large and small gradients, $\nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right)$,
- with the distance function, $\nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right)$,
- with the distance function, $\Delta \phi$,
- **reduction of the error by a factor** 100**.**



- transport of ϕ with $\boldsymbol{u}$, e.g.

$$\phi^* = \phi^n - \Delta t\, \boldsymbol{u}^n \nabla \phi^n,$$

- every 5 or 10 steps, reinitialize ϕ^*, e.g. with

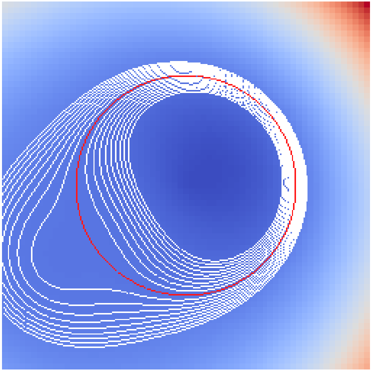
$$\partial_\tau \phi + sign(\phi^*)\,(|\nabla\phi| - 1) = 0,$$
$$\phi_{|\tau=0} = \phi^*.$$

- RK3-TVD scheme for τ, t, WENO-5 scheme for $\nabla\phi$,
- $\Delta\tau = \dfrac{\Delta x}{2}$.

Main issues:

- WENO-5 scheme for reinitialization is not accurate enough near the interface,
- too many reinitialization steps,
- reinitialization steps might be too expansive.

Test case with large and small gradients



$$d = \sqrt{x^2 + y^2} - r_0$$

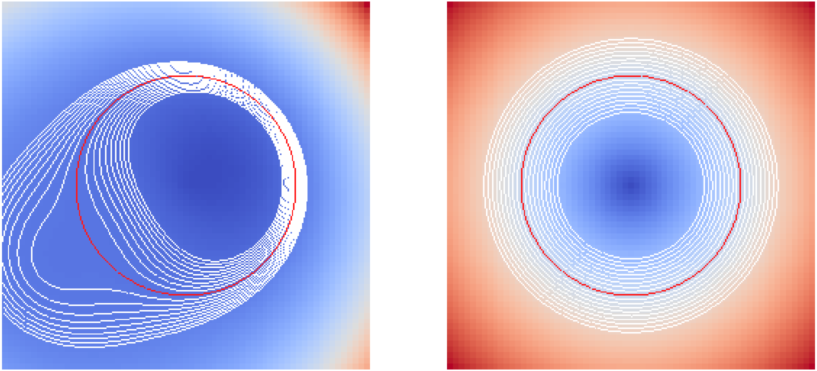
$$\phi_0 = \frac{d}{r_0}\left(\epsilon + (x - x_0)^2 + (y - y_0)^2\right)$$

$$\Omega = (-1, 1)^2$$

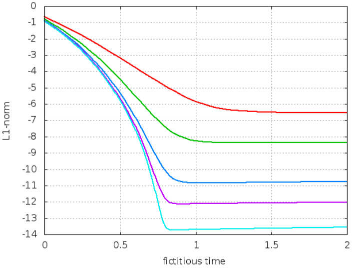
$$r_0 = 0.6$$

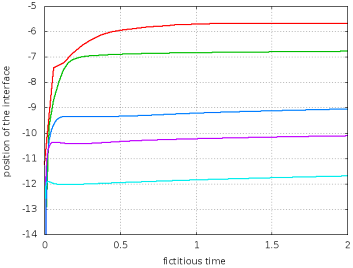
$$\epsilon = 0.1,\ x_0 = -0.7,\ y_0 = -0.4$$

Test case with large and small gradients



Accuracy of the standard approach





Main issue : WENO scheme uses informations on the wrong side of the interface

Subcell fix (Russo & Smereka)

Modify scheme near the interface :

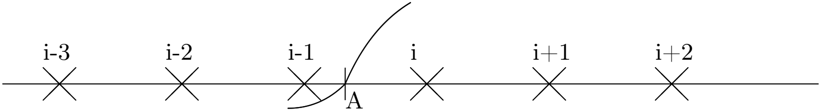
$$\phi_{i,j}^{n+1} = \phi_{i,j}^{n} - \frac{\Delta \tau}{\Delta x} \left(sign(\phi_{i,j}^{0}) |\phi_{i,j}^{n}| - d_{i,j}^{n} \right),$$

$d_{i,j}^{n}$ approximate value of the signed distance.

$\rightsquigarrow$ 2nd order for the position of the interface.

Main idea : use informations on the interface (almost upwind scheme)

Reducing interface displacement (high order)

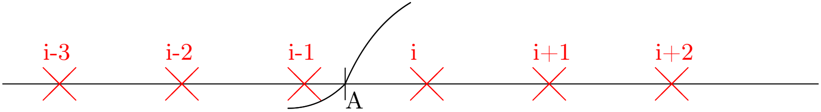


Standard WENO-5 stencil for $D_x^- \phi_i$: $(x_{i-3}, x_{i-2}, \cdots, x_{i+2})$

Use point A in the stencil : $(x_{i-2}, x_{i-1}, x_A, x_i, x_{i+1}, x_{i+2})$.
ENO scheme of high-order (Russo & Smereka, Gibou *et al*, Sussman & Fatemi).

- Away from the interface, use WENO scheme,
- close to the interface, use subcell fix with ENO scheme.

Reducing interface displacement (high order)

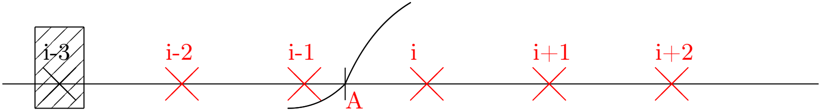


Standard WENO-5 stencil for $D_x^- \phi_i$: $(x_{i-3}, x_{i-2}, \cdots, x_{i+2})$

Use point A in the stencil : $(x_{i-2}, x_{i-1}, x_A, x_i, x_{i+1}, x_{i+2})$.
ENO scheme of high-order (Russo & Smereka, Gibou *et al*, Sussman & Fatemi).

- Away from the interface, use WENO scheme,
- close to the interface, use subcell fix with ENO scheme.

Reducing interface displacement (high order)

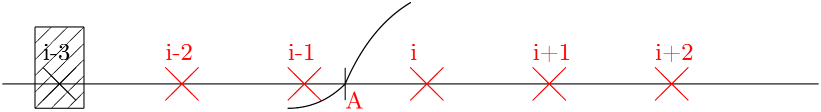


Standard WENO-5 stencil for $D_x^- \phi_i$: $(x_{i-3}, x_{i-2}, \cdots, x_{i+2})$

Use point A in the stencil : $(x_{i-2}, x_{i-1}, x_A, x_i, x_{i+1}, x_{i+2})$.
ENO scheme of high-order (Russo & Smereka, Gibou *et al*, Sussman & Fatemi).

- Away from the interface, use WENO scheme,
- close to the interface, use subcell fix with ENO scheme.

Reducing interface displacement (high order)

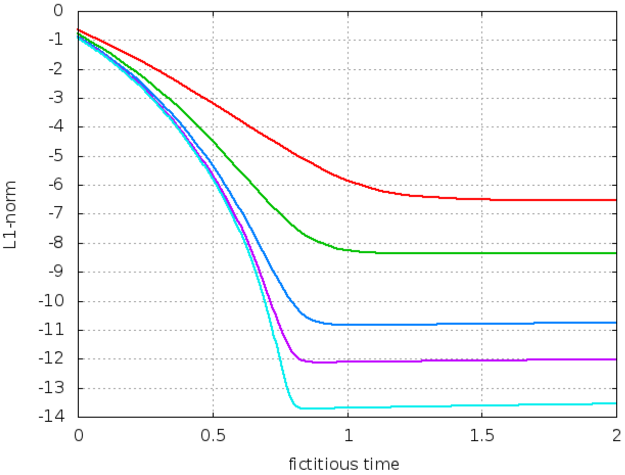


Standard WENO-5 stencil for $D_x^- \phi_i$: $(x_{i-3}, x_{i-2}, \cdots, x_{i+2})$

Use point A in the stencil : $(x_{i-2}, x_{i-1}, x_A, x_i, x_{i+1}, x_{i+2})$.
ENO scheme of high-order (Russo & Smereka, Gibou *et al*, Sussman & Fatemi).

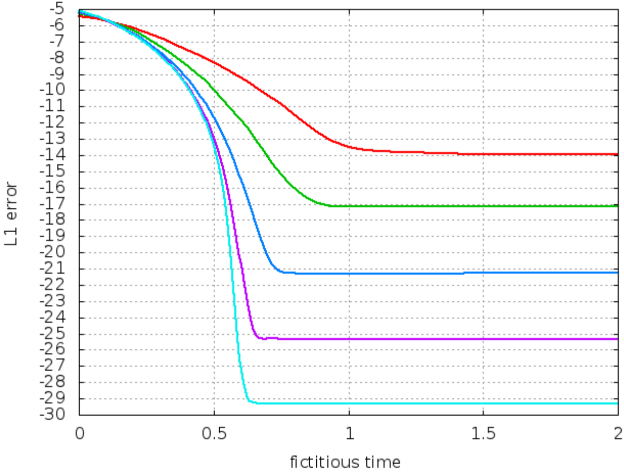
- Away from the interface, use WENO scheme,
- close to the interface, use subcell fix with ENO scheme.

Global accuracy



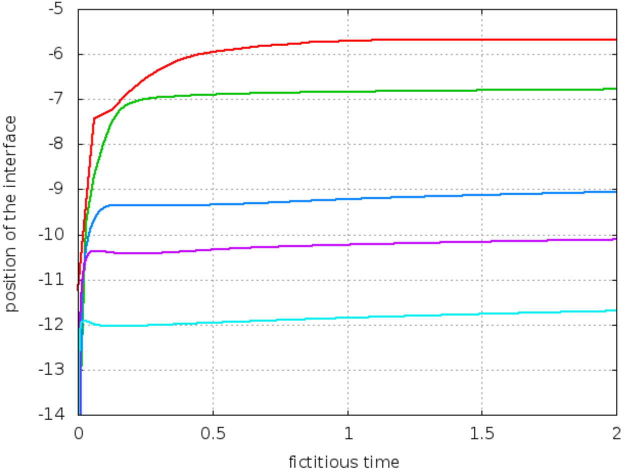
WENO scheme

Global accuracy



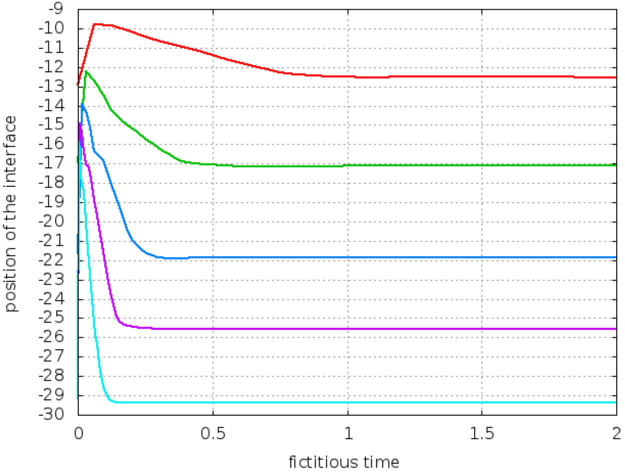
Subcell fix of order 3

Accuracy near the interface



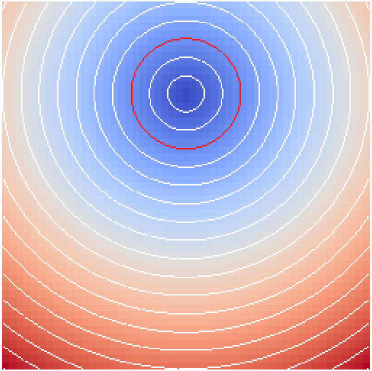
WENO scheme

Accuracy near the interface



Subcell fix of order 3

Coupling with transport



$$\Omega = (0,1)^2$$

$$\phi_{|t=0} = \sqrt{((x-0.5)^2 + (y-0.75)^2)} - 0.15$$

$$\boldsymbol{u} = \cos\left(\frac{\pi t}{T}\right) \nabla^{\perp}\omega$$

$$\omega = \sin(\pi x)^2 \sin(\pi y)^2$$

Coupling with transport (naive approach)

- evaluate spatial error in the scheme,
- previous example with $T = 0.046875$, results for $t = T$,
- $\Delta t = \Delta x^2$.

	every time step		only 5 reinitializations	
Δx	err.	coc	err.	coc
1/16	1.59e-02	-	2.76e-02	-
1/32	5.07e-03	1.65	5.05e-03	2.45
1/64	1.46e-04	5.12	8.48e-05	5.90
1/128	3.22e-05	2.18	2.50e-06	5.09
1/256	1.64e-04	-2.35	1.83e-07	3.77

Table: L^∞ error near the computed interface

$\rightsquigarrow$ **avoid reinitialization at every time step**

How to control the number of reinitializations?

- Introduce $r_g := \| |\nabla \phi| - 1 \|$,
- $r_g = 0$ for ϕ a distance function,
- while $r_g < \varepsilon$, use only transport for ϕ,
- once $r_g \geq \varepsilon$, replace ϕ by the distance function.

mesh	L^∞ error	coc	L^∞ error	coc
40	0.919E+00	-	0.108E+01	-
80	0.291E+00	1.66	0.363E+00	1.57
160	0.934E-01	1.64	0.921E-01	1.98
320	0.301E-01	1.64	0.343E-01	1.43
640	0.369E-02	3.03	0.422E-02	3.02

Table: Previous example with T=2, $\varepsilon = 0.1$. L^∞-errors on the curvature of the interface: at $t = 2$ (left) and $t = 4$ (right). For each computation : 16 reinitializations.

Numerical illustration

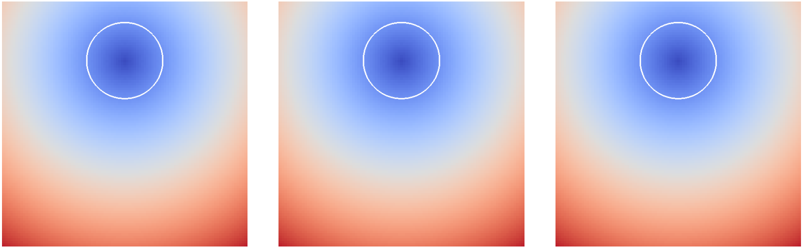


Figure: Previous example with $T = 6$ (600 time steps), $\varepsilon = 0.15$: without reinitialization (left), naive approach (middle, 600 reinitializations) and new strategy (right, 14 reinitializations).

Final strategy

Strategy based on the approximation of a continuous problem:

- *initialization:* start with ϕ the signed distance function,
- *transport:* as long as $r_g < \varepsilon$, evolve ϕ according to

$$\partial_t \phi + \boldsymbol{u} \cdot \nabla \phi = 0,$$

- *reinitialization:* if $r_g = \varepsilon$, replace ϕ by the distance function; go to transport.

Pros

• third order convergence for ϕ, first order convergence for κ,
• ϕ remains close to a distance function $\Rightarrow$ lower amplitudes of the error on κ.

Cons

• choice of ε,
• requires a full computation of the distance function. Relaxation might be expansive $\Rightarrow$ coupling relaxation method and fast-sweeping scheme away from the interface.

Analogy with linear systems

Relaxation method $\leftrightarrow$ relaxed Jacobi method,
(very accurate near the interface, but slow),

Fast-Sweeping method $\leftrightarrow$ Gauss-Seidel method,
(very fast, but requires a good guess near the interface).

In practice

- Use relaxation method in a band of length $5\Delta x$,
- iterate until convergence in the narrow band,
- then Fast-Sweeping away from the interface,
- second order FS, with fixed number of iterations.

Add-on: numerical illustration

Previous reinitialization example.

mesh	$\|d - \phi_h\|$	coc	$\|\kappa - \kappa_h\|$	coc	number of it.
40	6.14E-04	-	5.97E-02	-	23
80	5.28E-05	3.54	2.65E-02	1.17	28
160	5.60E-06	3.24	1.25E-02	1.08	32
320	6.47E-07	3.11	6.29E-03	1.00	36
640	8.44E-08	2.94	3.17E-03	0.99	41

Table: Errors and computed order of convergence (coc) for the mixed method. L^∞-error in the narrow band on ϕ (left), L^∞-error on κ on the interface (middle) and total number of iterations (right).

- (not shown here) second order accuracy for the global error on ϕ,
- with only relaxation, number of iterations $\sim 2/\Delta x$.

Conclusion

- Development of sharp cartesian method for air-water interfaces with a second order treatment of the pressure,
- High order level-set technique, allowing consistent computation of the curvature of the interface, even for long times.

What's next:

- Coupling of the two techniques,
- Implementation, validation in 3D,
- Application to fluid-solid interaction.