On the implementation of Structured Surfaces to FS3D

M. Baggio[1] and B. Weigand[1]
[1]Institute of Aerospace Thermodynamics, University of Stuttgart, Germany

The multi-phase program Free Surface 3D (FS3D)

Free Surface 3D (FS3D) is a code for the direct numerical simulation of incompressible multi-phase flows developed at the Institute of Aerospace Thermodynamics at the University of Stuttgart. Its fundamentals are (see figure 1):

- Spatial discretization with finite volumes on a MAC-staggered [1] Cartesian grid.
- Use of the Volume-of-Fluid [2] method for interface tracking.
- Use of the Piecewise Linear Interface Calculation (PLIC, [4]) algorithm for interface reconstruction in scalar control volumes.

The governing equations of the here considered case of an isothermal flow with no phase change characterized by a single liquid phase immersed in a continuous gas phase are:

- The zero divergence condition for the conservation of mass:

$$\nabla \cdot \boldsymbol{u} = 0 \tag{1}$$

- The incompressible Navier-Stokes equations for momentum transport

$$\frac{(\partial \rho \boldsymbol{u})}{\partial t} + \nabla \cdot [(\rho \boldsymbol{u}) \otimes \boldsymbol{u}] = -\nabla p + \nabla \cdot \mathbf{S} + \rho \boldsymbol{g} + \mathbf{f}_\sigma \tag{2}$$

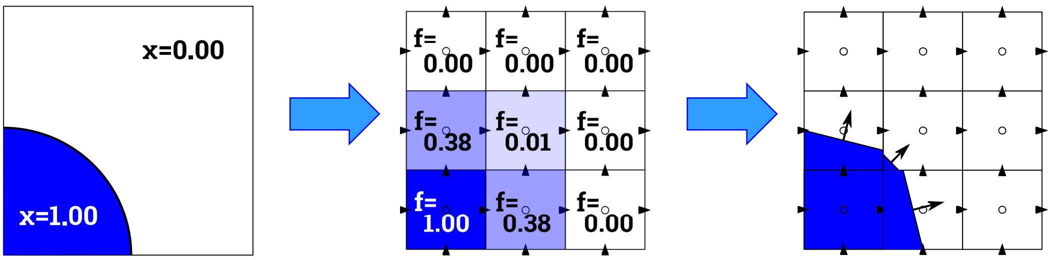
- Transport of the volume of fluid fraction variable f:

$$\frac{(\partial f)}{\partial t} + \nabla \cdot (f \boldsymbol{u}) = 0 \tag{3}$$

An equation for pressure is obtained from the zero-divergence constraint (1):

$$\nabla \cdot \left[\frac{1}{\rho} \nabla p\right] = \frac{\nabla \cdot \tilde{\boldsymbol{u}}}{\partial t} \tag{4}$$

Its discretization leads to a system of equations $[a]\boldsymbol{p} = \boldsymbol{b}$ whose solution is handled by a multigrid solver embedded in FS3D.



Space discretization (FV).

$$f = \frac{1}{\Omega} \int_\Omega \chi(\boldsymbol{x}) d\Omega$$

Interface reconstruction (PLIC)

Figure 1: FS3D fundamentals.

Implementation of structured surfaces

In a preliminary approach, embedded boundaries were represented as rigid bodies with infinite density. This resulted in the following steps:

1. Introduction of an additional volume fraction variable f_b and use of the PLIC scheme for boundary interface reconstruction (see figure 2 a).
2. Solution of the Poisson equation (4) for the "stair-stepped" approximation of the boundaries (see figure 2 b).
3. Cell-linking and averaging of the velocity field in near-boundary regions (see figure 2 c).

By modifying the velocity field in near boundary regions, an error is introduced and mass is not conserved. However, the error in mass conservation was very limited in our test simulations ($E_{m\,max} = \|(m_t - m_0)/(m_0)\|_\infty < 1.0 \times 10^{-3}$).

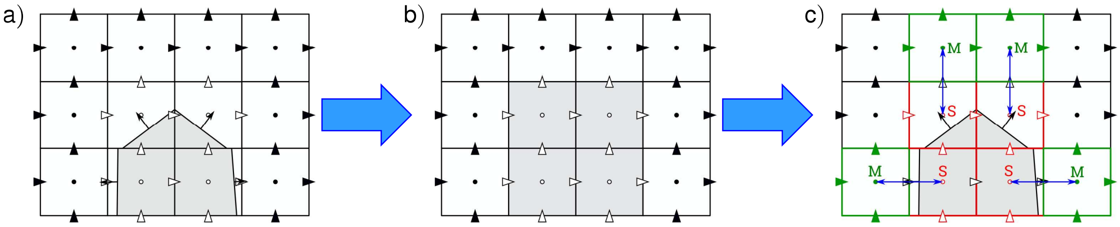


Figure 2: Treatment of embedded solid structures.

Towards a conservative method

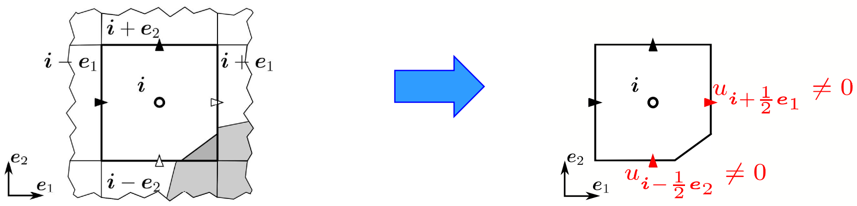


Figure 3: Implementation of a "cut-cell" method.

boundary cell

`real:`	l_b^*, f_b
`real(1:3):`	$\boldsymbol{n}_b$
`integer(1:3):`	$\boldsymbol{i} = i\boldsymbol{e}_1 + j\boldsymbol{e}_2 + k\boldsymbol{e}_3$
`real(1:3,1:2):`	$A_{b\,\boldsymbol{i}}$

boundary cell array

`type(bou_cell):`	$bcell_c$
`type(bou_cell):`	$bcell_x$
`type(bou_cell):`	$bcell_y$
`type(bou_cell):`	$bcell_z$

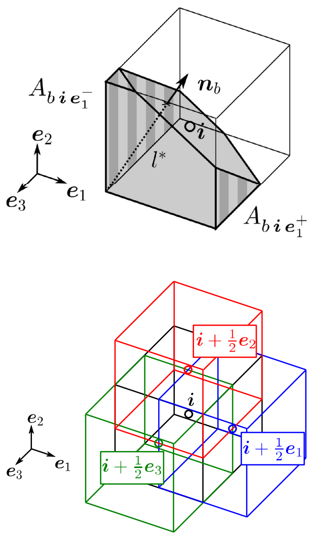


Figure 4: The structured type *boundary cell* and its implementation on a staggered grid.

We are currently developing a cut-cell approach inspired by the implementation of Popinet [3]. Indeed, for conservativeness, non-zero velocities are needed at boundary cell faces (see figure 3). Transport equations have then to be written in terms of cut-cell volume $(1 - f_b)h^3$ and cut-cell faces $A_{\boldsymbol{i}\,\boldsymbol{e}_d^{+,-}} h^2$ (here: split-scheme [5] and equidistant grid with spacing h):

$$h(1 - f_{b\,i})\frac{\partial \phi}{\partial t} = -\left(A_{i\,e_d^+} F(u_{i+\frac{1}{2}e_d}) - A_{i\,e_d^-} F(u_{i-\frac{1}{2}e_d})\right) + \text{divergence correction} \tag{5}$$

where ϕ is a generic scalar variable, F are the numerical fluxes, and d is the direction of the split advection step. Complex data structures are needed to store the necessary quantities on a staggered grid (see figure 4).

Testing

The new method has been tested against the old for a simple case of a water drop impacting on a solid sphere with a Weber number $We = \frac{\rho D {U_0}^2}{\sigma} = 46.15$ (see figure 5, top). These preliminary tests have shown that the new method tends to be unstable because very high velocity values are reached near the boundary surface. This occurs because probably our discretization scheme for the Poisson equation (4) is not very accurate in small cut-cells. However, an improved accuracy in mass conservation could be obtained (see figure 5, bottom).

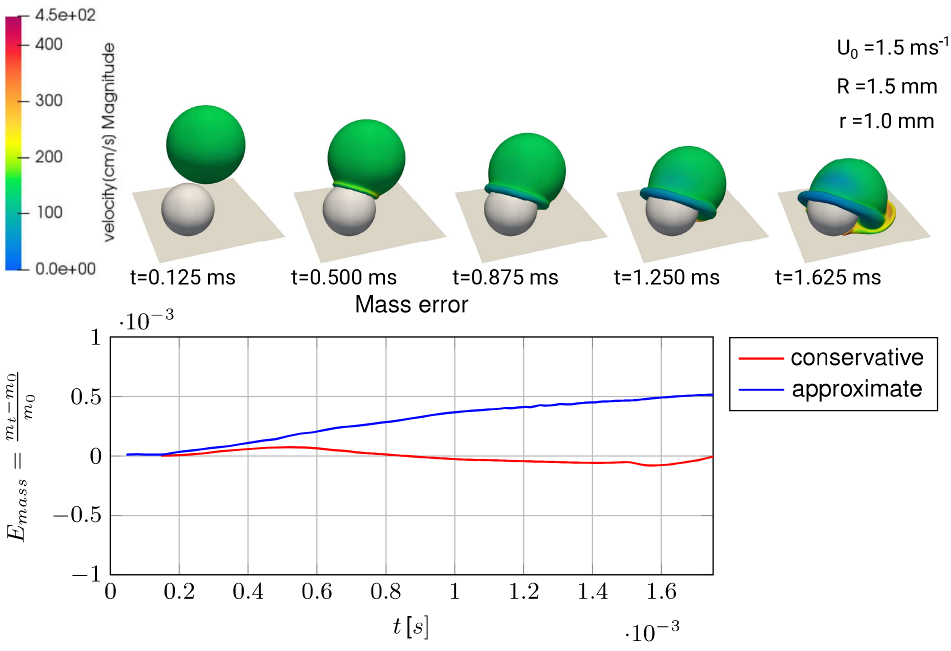


Figure 5: Comparison of the new approach with the approximate one for the case of a water drop impacting on a solid sphere with Weber Number $We = \frac{\rho D {U_0}^2}{\sigma} = 46.15$.

Acknowledgements

We kindly thank the German Science Foundation (DFG) for the financial support within the international training group DROPIT (Droplet Interaction Technologies), GRK2160/1.

Nomenclature

χ	colour function
ρ	density [kg m^{-3}]
σ	surface tension [kg s^{-1}]
ϕ	general scalar variable
A	cut-cell to whole face area ratio
D	drop diameter [m]
$\mathbf{e}_{1,2,3}$	orthonormal basis of $\mathbb{R}^3$ and $\mathbb{Z}^3$
E_m	mass error
F	numericsl flux
f	volume fraction
$\mathbf{f}_\sigma$	surface tension force per unit volume [N m^{-3}]

h	equidistant mesh spacing [m]
$\boldsymbol{i} = i\boldsymbol{e}_1 + j\boldsymbol{e}_2 + k\boldsymbol{e}_3$	cell index
M	master attribute of the data structure $\mathrm{boundary\ cell}$
l	interface distance [m]
m	mass [kg]
$\boldsymbol{n}$	normal vector [m^{-1}]
p	pressure [N m^{-2}]
R	drop radius [m]
$\mathbb{S}$	viscous stress tensor [N m^{-2}]
S	slave attribute of the data structure $\mathrm{boundary\ cell}$
r	radius of the spherical feature [m]
t	time [s]
U_0	impact velocity [m s^{-1}]
$\boldsymbol{u} = u\boldsymbol{e}_1 + v\boldsymbol{e}_2 + w\boldsymbol{e}_3$	velocity vector [m s^{-1}]
$\boldsymbol{x} = x\boldsymbol{e}_1 + y\boldsymbol{e}_2 + z\boldsymbol{e}_3$	space position [m]

References

[1] F.H. Harlow and J.E. Welch. 'Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface'. In: *The Physics of Fluids* 8.12 (1965), pp. 2182–2189.

[2] C.W. Hirt and B.D. Nichols. 'Volume of fluid (VOF) method for the dynamics of free boundaries'. In: *Journal of Computational Physics* 1.39 (1981), pp. 201–225.

[3] S. Popinet. 'Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries'. In: *Journal of Computational Physics* 190 (2003), pp. 572–600.

[4] W.J. Rider and D.B. Kothe. 'Reconstructing volume tracking'. In: *Journal of Computational Physics* 2.141 (1998), pp. 112–152.

[5] M. Rieber. 'Numerische Modellierung der Dynamik freier Grenzflächen in Zweiphasenströmungen'. Doctoral thesis. University of Stuttgart, 2004.