On the implementation of Structured Surfaces to FS3D

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

Introduction: the DNS multi-phase code Free Surface 3D (FS3D)

Free Surface 3D (FS3D) is a multi-phase solver for the incompressible Navier-Stokes equations:

$$\frac{(\partial \rho \boldsymbol{u})}{\partial t} + \nabla \cdot [(\rho \boldsymbol{u}) \otimes \boldsymbol{u}] = -\nabla p + \nabla \cdot \mathbf{S} + \rho \boldsymbol{g} + \mathbf{f}_\sigma \tag{1}$$

Space discretization is based on finite volumes on a Cartesian grid. According to the VOF approach [1], the interface of the i-th phase is tracked by solving a transport equation for f_i:

$$\frac{(\partial f_i)}{\partial t} + \nabla \cdot (f_i \boldsymbol{u}) = 0 \tag{2}$$

where f_i is the volume fraction of the considered phase in the considered control volume.

Figure 1: Schematic overview of FS3D's numerical methods.

Representation of embedded boundaries

Similarly to [2], embedded boundaries are represented by their volume fraction f_b and their surface is approximated with the PLIC [3] scheme. The boundary is treated as a rigid body with infinite density. As a consequence, the fluid cannot enter those boundary cells ($0 < f_b < 1$) where the boundary occupies ($f_{b\,MCV} > 0$) 5 (3 in 2D) of the surrounding momentum control volumes (see figure 2). We bypass this problem with the following strategy:

- Identification of critical boundary cells (*slave* cells).
- Linking of *slave* cells to their neighbour (*master*) in the direction of the largest normal component $\boldsymbol{n}_b$.
- VOF-advection in *master-slave* couples.

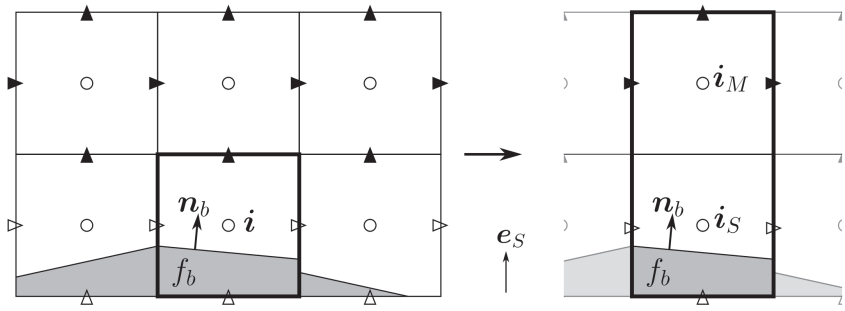

Figure 2: Left: "critical" boundary cell. Right: linking of this *slave* cell with its master. The vector $\boldsymbol{e}_S$ is the direction of the largest component of the normal vector $\boldsymbol{n}_b$.

FS3D's split advection scheme

VOF-advection occurs separately in each spatial direction [4], [3]:

$$\frac{f_i^* - f_i^n}{\Delta t} = -\frac{F_{i+\frac{1}{2}e_1} - F_{i-\frac{1}{2}e_1}}{\Delta x_i} + [(1-\beta)f_i^n + \beta f_i^*]\frac{u_{i+\frac{1}{2}e_1} - u_{i-\frac{1}{2}e_1}}{\Delta x_i} \tag{3a}$$

$$\frac{f_i^{**} - f_i^*}{\Delta t} = -\frac{F_{i+\frac{1}{2}e_2} - F_{i-\frac{1}{2}e_2}}{\Delta y_i} + [(1-\beta)f_i^* + \beta f_i^{**}]\frac{v_{i+\frac{1}{2}e_2} - v_{i-\frac{1}{2}e_2}}{\Delta y_i} \tag{3b}$$

$$\frac{f_i^{n+1} - f_i^{**}}{\Delta t} = -\frac{F_{i+\frac{1}{2}e_3} - F_{i-\frac{1}{2}e_3}}{\Delta z_i} + [(1-\beta)f_i^{**} + \beta f_i^{n+1}]\frac{w_{i+\frac{1}{2}e_3} - w_{i-\frac{1}{2}e_3}}{\Delta z_i} \tag{3c}$$

The first term on the right hand side is the contribution of the numerical fluxes through the cell boundaries; the second term is the divergence correction. Each one dimensional step in equations (3) requires three sub-steps:

- Interface reconstruction: calculation of interface position and interface normal vector $\boldsymbol{n}_b$.
- Geometrical calculation of the numerical fluxes F.
- Update of f by means of equations (3).

Each sub-step was adapted to the presence of the embedded boundaries.

Flux calculation in merged (*master-slave*) cells

An averaged velocity on master-slave upstream faces is used for geometrical calculation of the numerical fluxes (see figure 3).

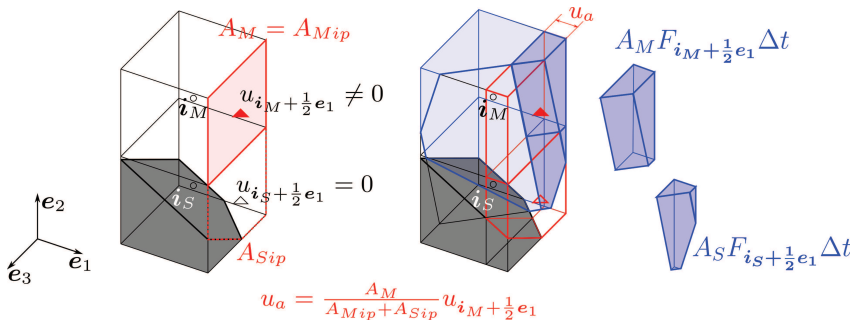

Figure 3: Left: velocity averaging on *master-slave* faces. Right: example of flux calculation in *master-slave* cells.

VOF-update in *master-slave* cells

Equations (3) are re-formulated as follows:

$$\frac{f_i^* - f_i^n}{\Delta t} = -\frac{F_{OUT} - F_{IN}}{V_{i_S} + V_{i_M}} + [(1-\beta)f_i^n + \beta f_i^*]\frac{\dot{V}_{OUT} - \dot{V}_{IN}}{V_{i_S} + V_{i_M}} \tag{4}$$

The divergence correction is modified to account for the presence of the boundary (see figure 4).

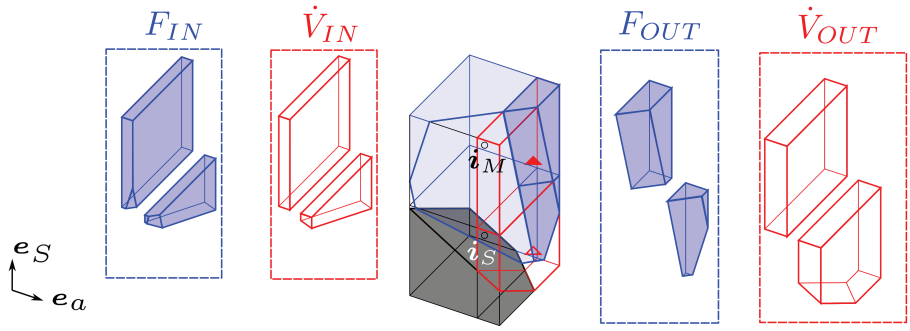

Figure 4: VOF-update in *master-slave* cells for the case $e_a \cdot e_S = 0$, where e_a is the advection direction, e_S. the direction by the largest n_b component in the slave cell.

Results

Results The method has been tested for drop impact on different geometries. The results for drop impact on hemispherical features are shown in figure 5.

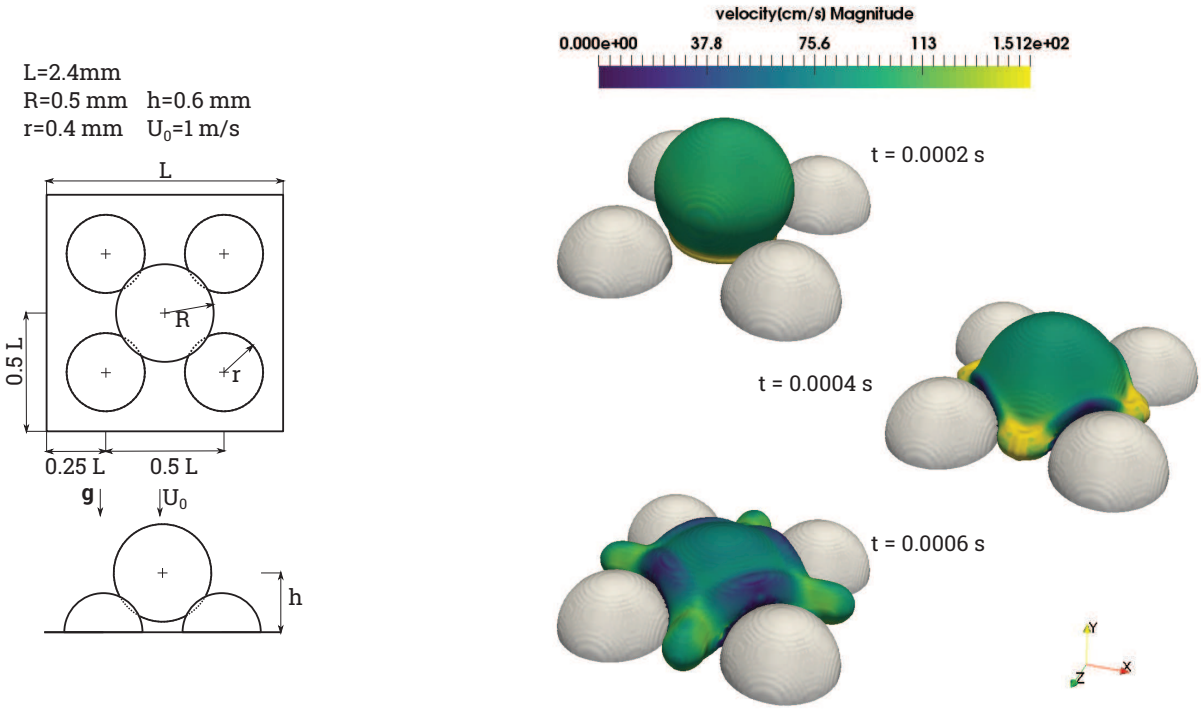

Figure 5: Results for drop impact on hemispherical features for $128 \times 64 \times 128$ computational cells.

Acknowledgements

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

References

[1] C.W. Hirt and B.D. Nichols. In: *J. Comput. Phys* 1.39 (1981), pp. 201–225.

[2] P. Rauschenberger and B. Weigand. In: *J. Comput. Phys* 291 (2015), pp. 238–253.

[3] W.J. Rider and D.B. Kothe. In: *J. Comput. Phys* 2.141 (1998), pp. 112–152.

[4] M. Rieber. Doctoral thesis. University of Stuttgart, 2004.