1 Construction of Boundary values

The interface between fluid F and solid S in the VTF application is modeled as a contact discontinuity [2, 4]. The compatibility conditions are the continuity of normal velocity and normal stress across the interface, while shear stresses have to vanish in case of an inviscid fluid, i.e.

$$\begin{aligned} v_n^S &= v_n^F, \\ \sigma_n^S &= -p^F, \\ \sigma_t^S &= 0. \end{aligned} \tag{1}$$

For a fluid-fluid coupling we have

$$\begin{aligned} v_n^{F_1} &= v_n^{F_2}, \\ p^{F_1} &= p^{F_2}. \end{aligned} \tag{2}$$

As the Eulerian fluid calculation uses *ghost cells* to set boundary conditions the practial question arises, how such values have to be chosen to fulfill the latter conditions in the *discrete sense*. Apparently, for the consistent matching of discrete values their location on the computational mesh (the *staggering*) can't be neglected and we start with the precription of Dirichlet boundary conditions along the boundary in a cell-centered discretization. This question is of course also relevant for moving embedded boundaries.

1.1 Moving Wall Boundary Conditions for Euler Equations

At a moving wall the velocity component in the normal direction is required to be the wall velocity w, i.e.

$$v_n = w. \tag{3}$$

The Galiean transformation $v_n' = v_n - w = 0$ leads to the usual wall boundary condition

$$v_n' = 0,$$

which we satisfy along the interface $i + \frac{1}{2}$ by choosing $-v_{n,i}' = -(v_{n,i} - w)$ as normal velocity in the ghost cell $i + 1$. Transforming this into the stationary frame of reference gives $2w - v_{n,i}$ in the ghost cell $i + 1$. Obviously, this choice is consistent with the boundary condition (3) at the boundary $i + \frac{1}{2}$, cf. Fig. 1.

In primitive variables the values of ρ, $\mathbf{V} = (v_n, v_t)$, p have to be *mirrored* across the interface. The ghost cell velocities are then calculated by evaluating the vector expression

$$\mathbf{V}_{Gh}^{F_1} = (2\mathbf{W} \cdot \mathbf{n} - \mathbf{V}_M^{F_1} \cdot \mathbf{n})\mathbf{n} + (\mathbf{V}_M^{F_1} \cdot \mathbf{t})\mathbf{t} = 2\left((\mathbf{W} - \mathbf{V}_M^{F_1}) \cdot \mathbf{n}\right)\mathbf{n} + \mathbf{V}_M^{F_1}.$$

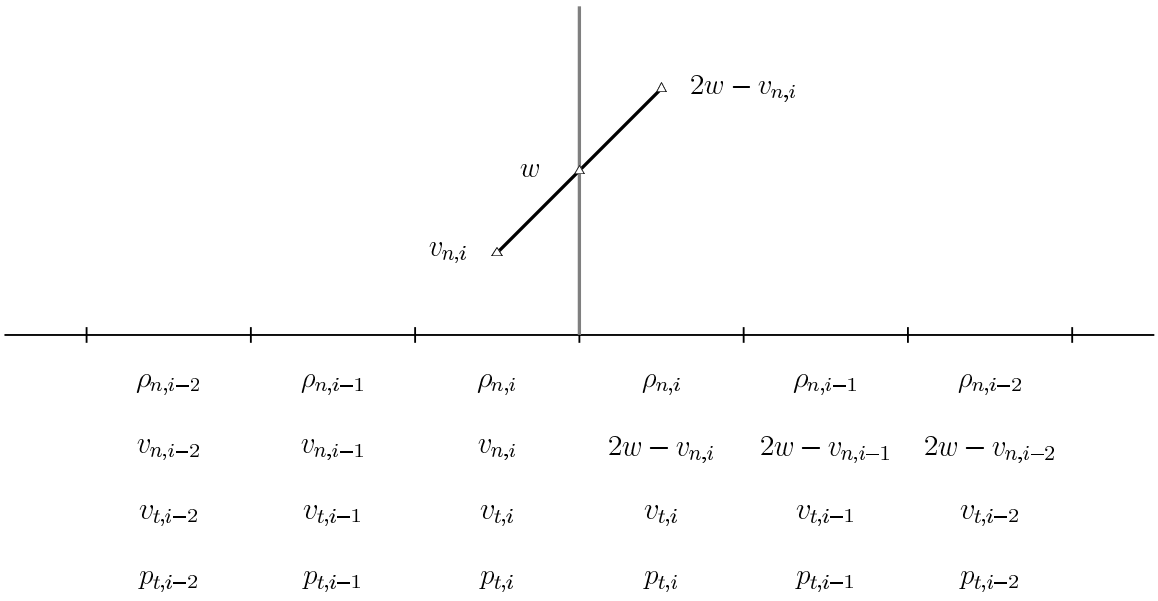

Figure 1: Ghost cell setting for a moving wall boundary for Euler Equations.

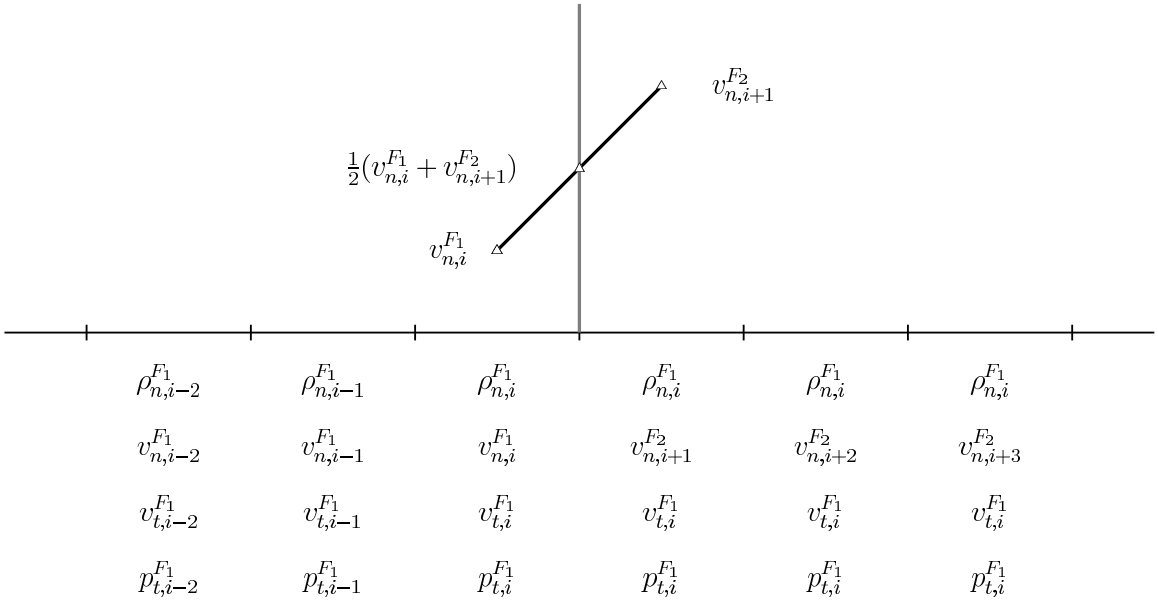

Figure 2: Ghost cell setting in a ghost fluid method for fluid F_1.

1.2 Fluid-Fluid Coupling

Fluid solvers typically employ a *cell-centered* finite volume discretizations. A ghost fluid method consistent with the compatability conditions (2) can easily be constructed just by extrapolating values from one side and replacing pressure and normal velocity in the ghost cells cell-center by cell-center [3]. See Fig. 2 for the setting of the ghost cells in fluid F_1 right of the interface $i+\frac{1}{2}$. If both fluid solvers use the same averaging at cell interfaces, e.g. the Roe-averaging

$$\hat{u} = \frac{\sqrt{\rho_l}u_l + \sqrt{\rho_r}u_r}{\sqrt{\rho_l} + \sqrt{\rho_r}},$$

they satisfy the same discrete compatibility conditions, because both solvers would use $\hat{v}_{n,i+\frac{1}{2}} = \frac{1}{2}(v^{F_1}_{n,i} + v^{F_2}_{n,i+1})$ and $\hat{p}_{i+\frac{1}{2}} = \frac{1}{2}(p^{F_1}_i + p^{F_2}_{i+1})$ to evaluate the flux at the interface.

In multiple space-dimensions an effective implementation would solve the Eikonal equation

$$I_t + \mathbf{n} \cdot \nabla = 0$$

with boundary data from the cells near the interface to extrapolate ρ_i, $v_{t,i}$, p_i. The velocity treatment in the ghost cells for F_1 can also be expressed as

$$\mathbf{V}^{F_1}_{Gh} = (\mathbf{V}^{F_2} \cdot \mathbf{n})\mathbf{n} + (\mathbf{V}^{F_1}_{Ext} \cdot \mathbf{t})\mathbf{t} = (\mathbf{V}^{F_2} \cdot \mathbf{n})\mathbf{n} + \mathbf{V}^{F_1}_{Ext} - (\mathbf{V}^{F_1}_{Ext} \cdot \mathbf{n})\mathbf{n}.$$

1.3 Solid-Fluid Coupling

A solid solver typically uses a *node-centered* finite element discretization. If the finite element mesh is conformal with the finite volume mesh, the values from the solid computation are assigned to cell interfaces. If the boundary velocity in the normal direction is presribed by the solid computation, but the pressure load on the finite element nodes is taken from the finite volume mesh [2], the values in the ghost cells next to the boundary are already fully prescribed, cf. Figs. 3 and 4. The discrete coupling conditions at interface $i+\frac{1}{2}$, e.g. with Roe-averaging in the fluid solver, are

$$v^S_{n,i+\frac{1}{2}} = \hat{v}_{n,i+\frac{1}{2}} = \tfrac{1}{2}(v^{F_1}_{n,i} + 2v^S_{n,i+\frac{1}{2}} - v^{F_1}_{n,i}),$$
$$p^S_{n,i+\frac{1}{2}} = \hat{p}_{i+\frac{1}{2}} = \tfrac{1}{2}(p^{F_1}_{n,i} + p^{F_1}_{n,i}).$$

If the values in the remaining cells are set by constant value extrapolation, the velocity treatment in the ghost cells could be expressed as

$$\mathbf{V}^{F_1}_{Gh} = (2\mathbf{V}^S \cdot \mathbf{n} - \mathbf{V}^{F_1}_{Ext} \cdot \mathbf{n})\mathbf{n} + (\mathbf{V}^{F_1}_{Ext} \cdot \mathbf{t})\mathbf{t} = 2\left((\mathbf{V}^S - \mathbf{V}^{F_1}_{Ext}) \cdot \mathbf{n}\right)\mathbf{n} + \mathbf{V}^{F_1}_{Ext}.$$

If the ghost cell values are derived by mirroring across the interface, cf. Fig. 4, the interface is seen by the fluid solver as a moving rigid wall and the expression for the ghost cell velocities is again

$$\mathbf{V}^{F_1}_{Gh} = (2\mathbf{V}^S \cdot \mathbf{n} - \mathbf{V}^{F_1}_{M} \cdot \mathbf{n})\mathbf{n} + (\mathbf{V}^{F_1}_{M} \cdot \mathbf{t})\mathbf{t} = 2\left((\mathbf{V}^S - \mathbf{V}^{F_1}_{M}) \cdot \mathbf{n}\right)\mathbf{n} + \mathbf{V}^{F_1}_{M}. \qquad (4)$$

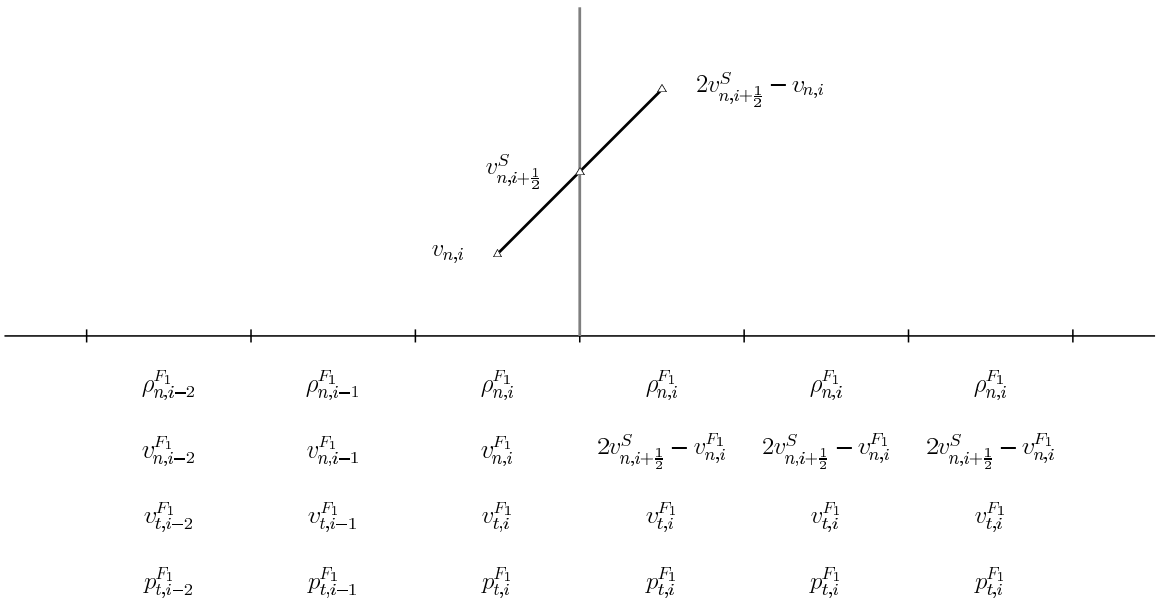

Figure 3: Ghost cell setting in an embedded boundary method for fluid F_1 with prescribed velocity derived from a node-centered solid dynamics calculation. Remaining fluid values extrapolated.

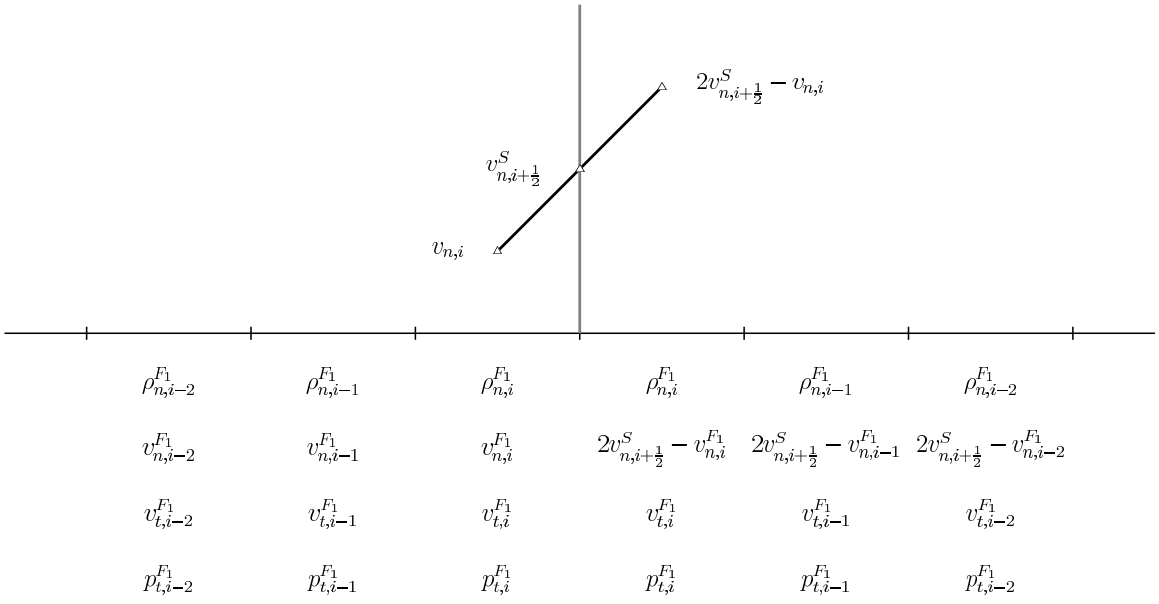

Figure 4: Ghost cell setting in an embedded boundary method for fluid F_1 with prescribed velocity derived from a node-centered solid dynamics calculation. Remaining fluid values mirrored.

1.4 Spatial Interpolation on Cartesian Meshes for Diffused Boundary Methods

For the VTF application the treatment of the solid boundary as a moving rigid body in the Eulerian fluid solver seems to be most appropriate. This is also suggested by previous investigations [1]. Solving the Eikonal equation $I_t + \mathbf{n} \cdot \nabla = 0$ with pseudo time steps would be possible to achieve a multi-dimensional extrapolation, but an interpolation/extrapolation routine to calculate mirrored values efficiently is more desirable as they are required to construct ghost cell values for moving rigid bodies in non-coupled simulations anyhow.

The construction of a mirrored ghost cell value in a ghost cell center $\mathbf{x}_{ij}$ requires the calculation on a space interpolated value in the point

$$\tilde{\mathbf{x}} = \mathbf{x}_{ij} + 2\varphi_{ij}\mathbf{n}_{ij}.$$

The standard two-dimensional bilinear interpolation on a Cartesian grid is

$$u(x_1, x_2) = (1-t)(1-s)u_{ij} + t(1-s)u_{i+1,j} + (1-t)su_{i,j+1} + tsu_{i+1,j+1}$$

with

$$\begin{aligned} t &= (x_1 - x_{1,i})/(x_{1,i+1} - x_{1,i}), \\ s &= (x_2 - x_{2,j})/(x_{2,j+1} - x_{2,j}). \end{aligned}$$

As only discrete values from the interior (e.g. with $\varphi < 0$) can be considered the latter formula must be adopted appropriately, if less than four points are available in the interior. See the two left case in Fig. 5. In case of three points we approximate the missing point value by linear extrapolation:

$$\begin{array}{lcrrrr} \tilde{u}_{ij} & = & & u_{i+1,j} & +u_{i,j+1} & -u_{i+1,j+1} \\ \tilde{u}_{i+1,j} & = & u_{ij} & & -u_{i,j+1} & +u_{i+1,j+1} \\ \tilde{u}_{i,j+1} & = & u_{ij} & -u_{i+1,j} & & +u_{i+1,j+1} \\ \tilde{u}_{i+1,j+1} & = & -u_{ij} & +u_{i+1,j} & +u_{i,j+1} & \end{array}$$

In case of only two points we move the interpolation stencil by a full row or column and apply an extrapolation into $u(x_1, x_2)$ with $|t| > 1$ and/or $|s| > 1$, cf. lower left case in Fig. 5.

- The same interpolation/extrapolation routine can be applied to approximate fluid values at the nodal locations of the solid mesh, cf. right red case in Fig. 5.
- In order to achieve optimal computational performance the interpolation routine is called only once for each AMR patch. Therefore it must take a *list* of points as an argument.
- A "slope-limiting" based on the ratio $(u_{i+1} - u_i)/(u_i - u_{i-1})$ is highly desirable to avoid the inappropriate interpolation across discontinuities.

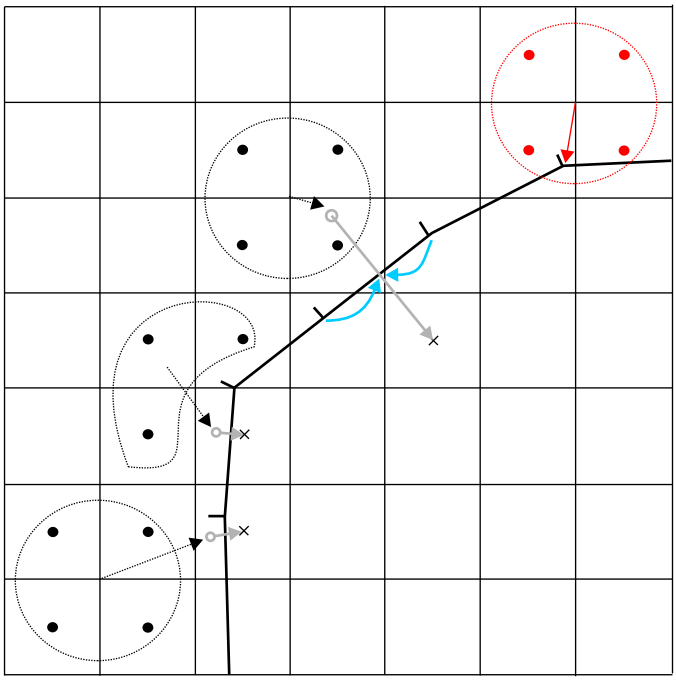

Figure 5: Ghost cell setting in an embedded boundary method for fluid F_1 with prescribed velocity derived from a node-centered solid dynamics calculation. Remaining fluid values mirrored.

References

[1] M. Arienti, P. Hung, E. Morano, and J. E. Shepherd. A level set approach to Eulerian-Lagrangian copuling. *J. Comput. Phys.*, 185:213–251, 2003.

[2] R. P. Fedkiw. Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method. *J. Comput. Phys.*, 175:200–224, 2002.

[3] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152:457–492, 1999.

[4] G. H. Miller and P. Colella. A conservative three-dimensional Eulerian method for coupled solid-fluid shock capturing. *J. Comput. Phys.*, 183:26–82, 2002.