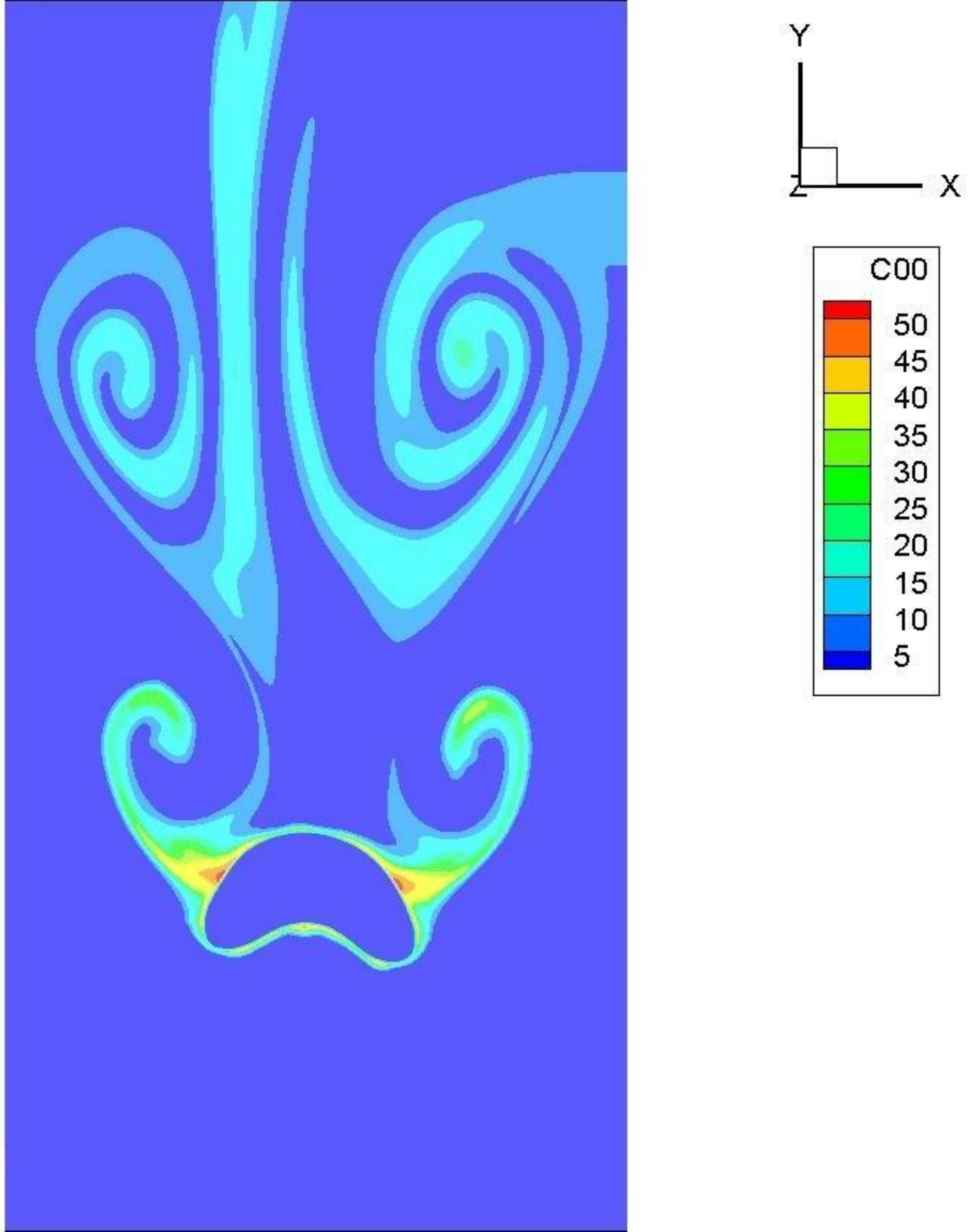
A 3D sharp interface method for modelling the mixing of two partially miscible fluid phases

Aniket Patankar

Department of Mechanical Engineering

Massachusetts Institute of Technology

2.29 Project, Spring 2018



A droplet of heavy oil (900kg/m3) falling under gravity in water (250kg/m3).
Conditions: 700K, 25MPa.
Oil partially miscible in water.

C00 in kg/m3

Dimensions: 4mm*8mm
Droplet initial diameter: 1mm
Total real time simulated: 0.15s

Grid 400*800 uniform squares.
Simulation performed on 16 cores for 20 hours.

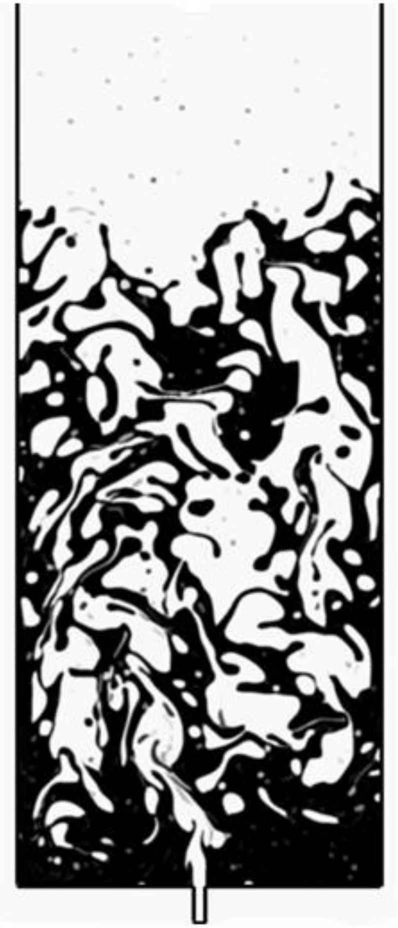
The code was developed within the OpenFOAM[1] (v. 2.3.0) framework at the Reacting Gas Dynamics Laboratory at MIT[2].

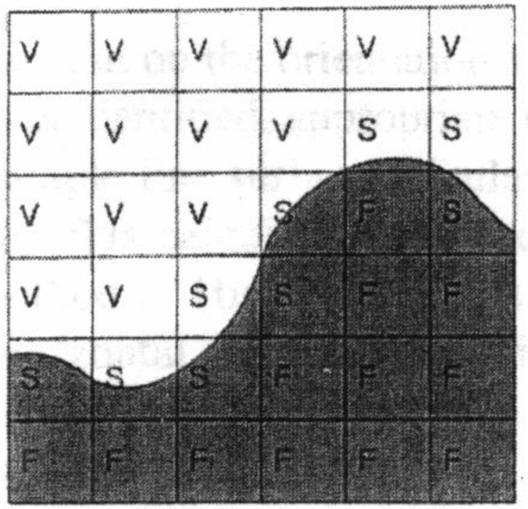
The Volume of Fluid Method

Lagrangian methods: different meshes for different phases. Meshes evolve in time.

Eulerian methods: Fixed grid.

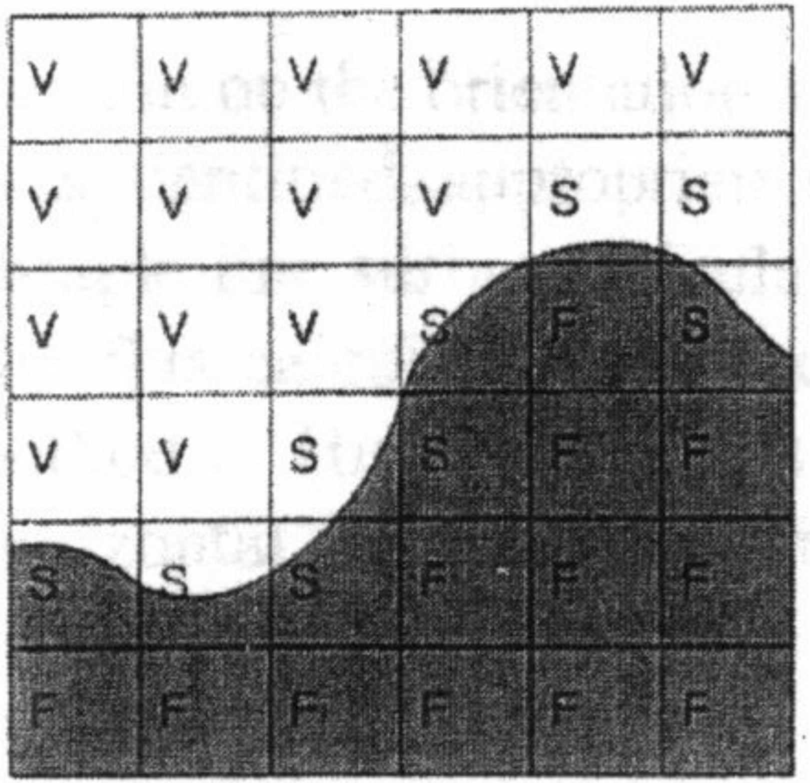
Examples: VoF, Level-Set methods





The Volume of Fluid Method

- The volume fraction as a field variable – that is the only handle we have on which phase is where.
- Single set of volume-averaged (using volume fraction) field equations for u, p, T, species concentration, etc.
- Things get complicated when interfacial mass transfer is present – partially miscible fluids.

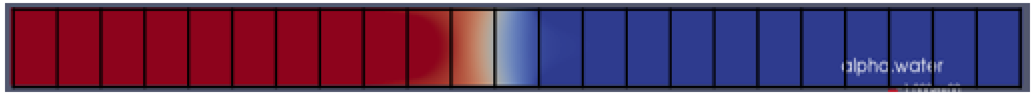


Advecting the volume fraction

Algebric Advection

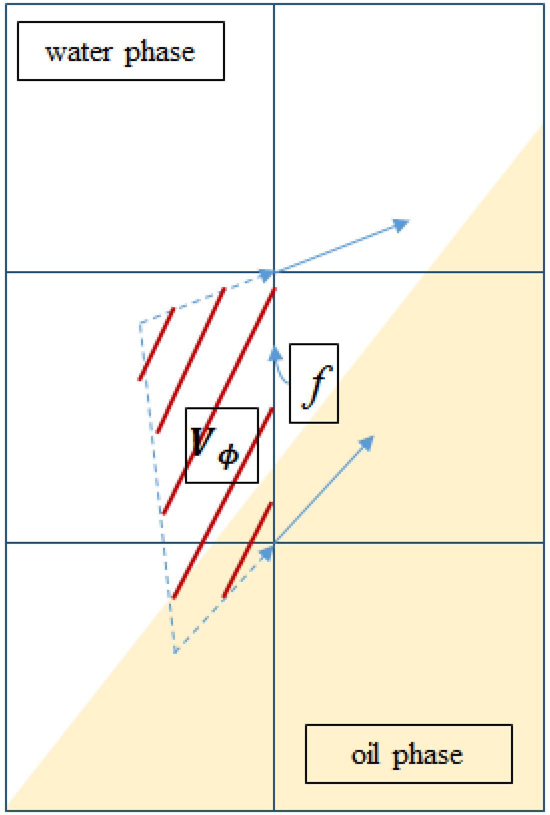
$$\frac{\partial f}{\partial t} + \nabla \cdot (f\vec{\mathrm{v}}_m) = 0$$

Solve a finite-volume conservation equation for volume fraction (f). In above equation, fluids are incompressible and there is no mass transfer

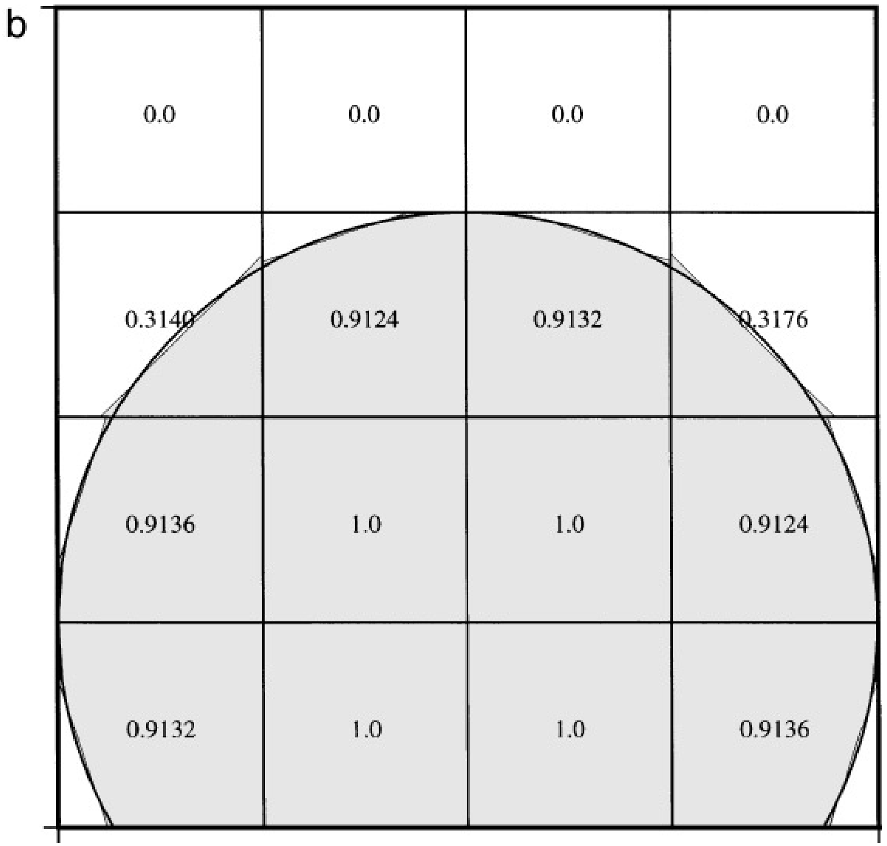


Leads to smearing of the interface due to numerical diffusion. In above figure, f changes from 0 to 1 over 4 cells.

Geometric Advection

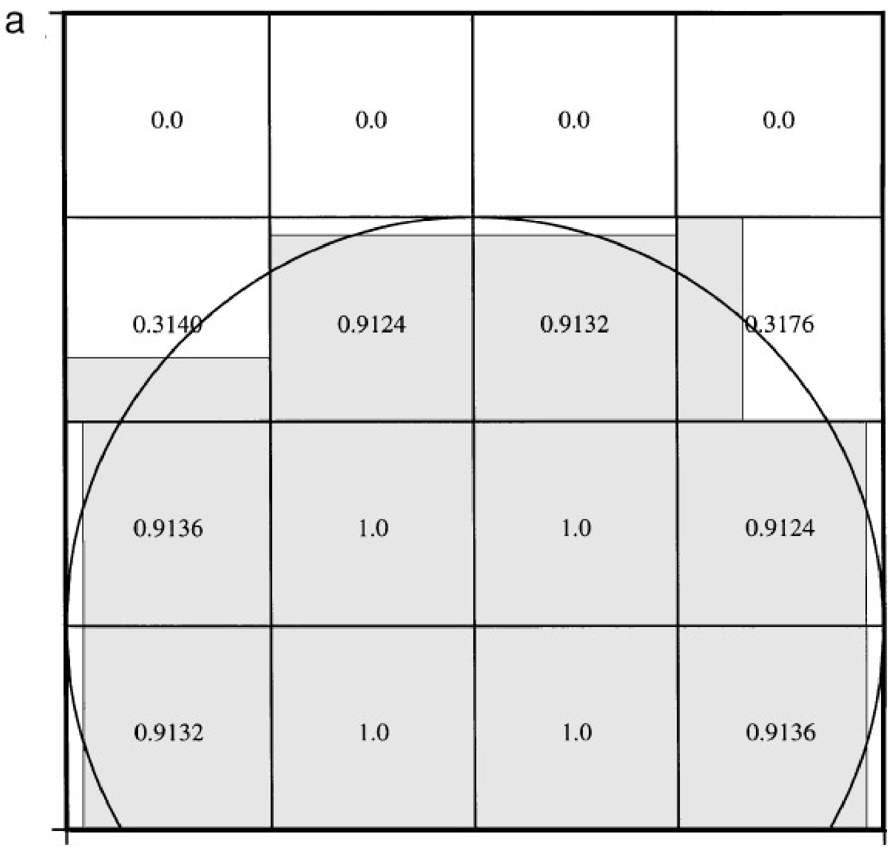


Reconstructing the interface



0.0	0.0	0.0	0.0
0.3140	0.9124	0.9132	0.3176
0.9136	1.0	1.0	0.9124
0.9132	1.0	1.0	0.9136

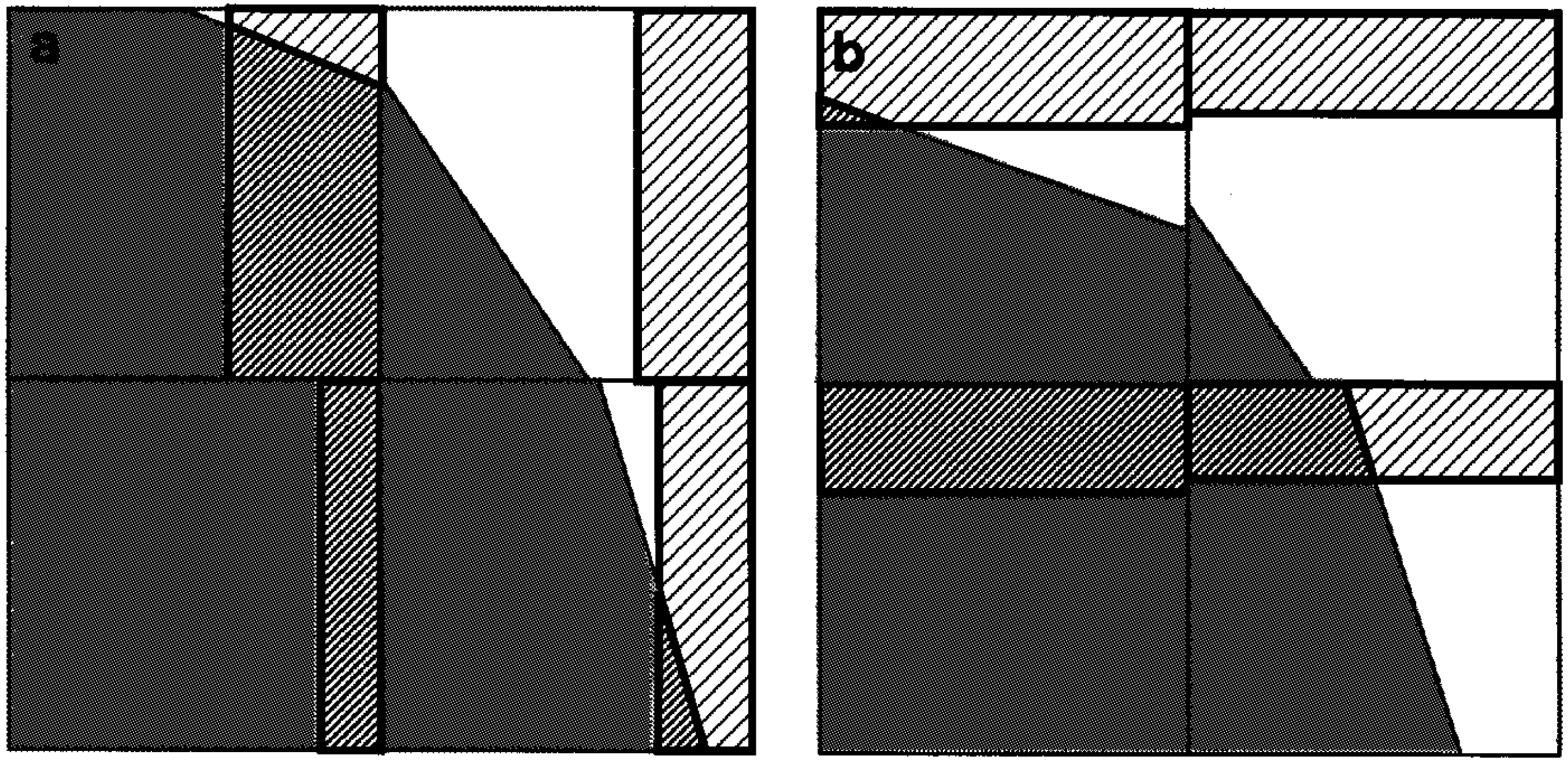
Piece-wise linear interface calculation (PLIC)[4]



0.0	0.0	0.0	0.0
0.3140	0.9124	0.9132	0.3176
0.9136	1.0	1.0	0.9124
0.9132	1.0	1.0	0.9136

Simple linear interface calculation (SLIC) [4]

Flux polygons - v1[4]



Operator-splitting schemes:

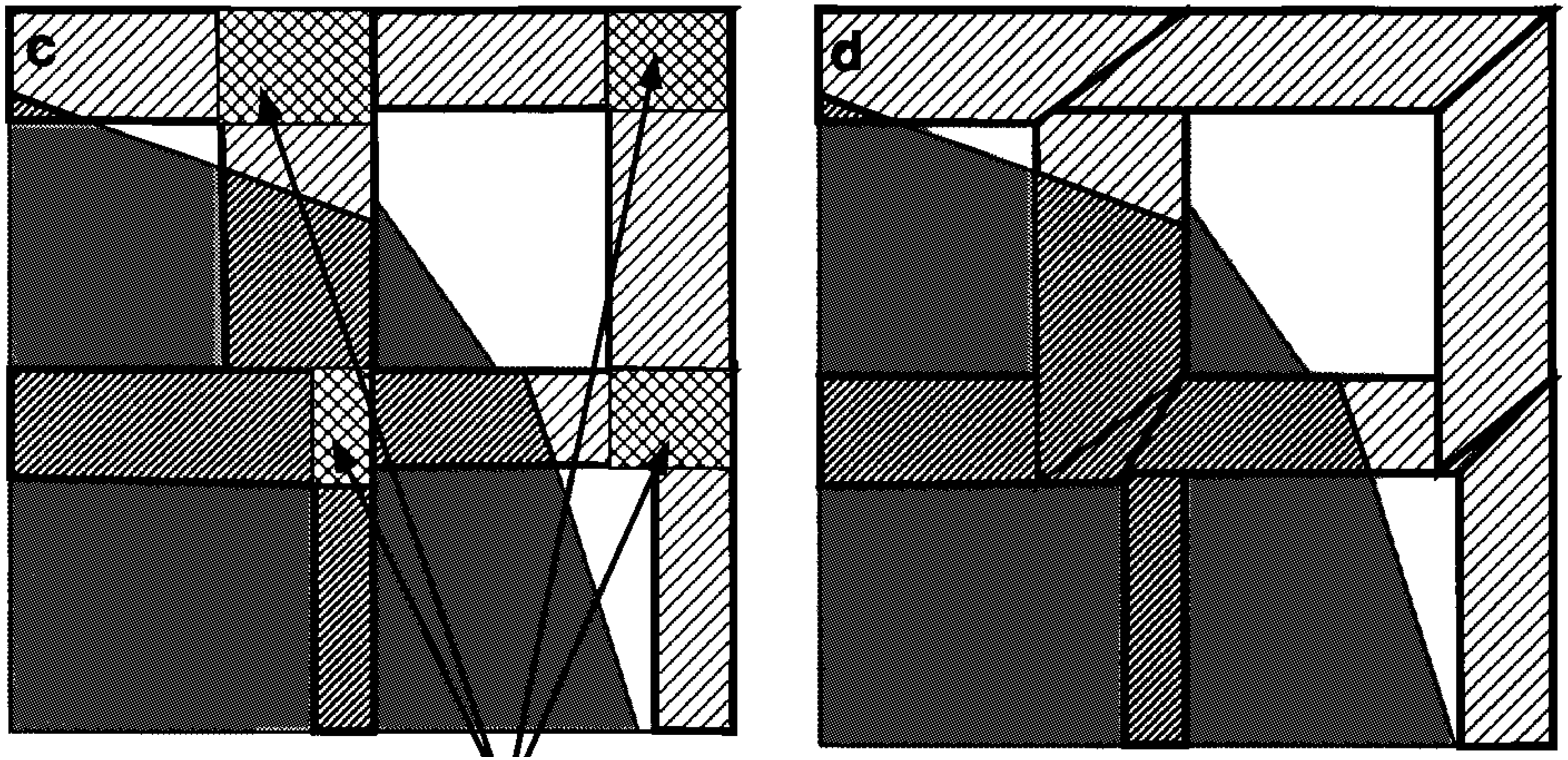
a. Advect in x direction

b. Advect in y direction

Strang splitting can improve order of accuracy

Lower order schemes can lead to formation of artificially deformed interfaces and **whisps**.

Flux polygons – v2[4]



‘Unsplit’ schemes:

c. Naïve approach. causes double counting.

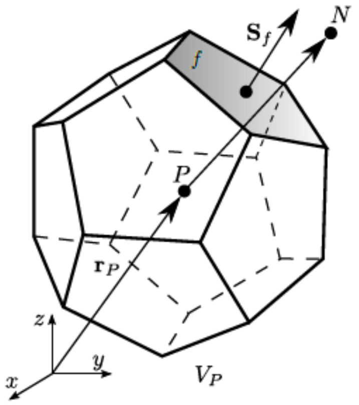
d. Use corner velocities to construct the flux polygon Strang splitting can improve order of accuracy

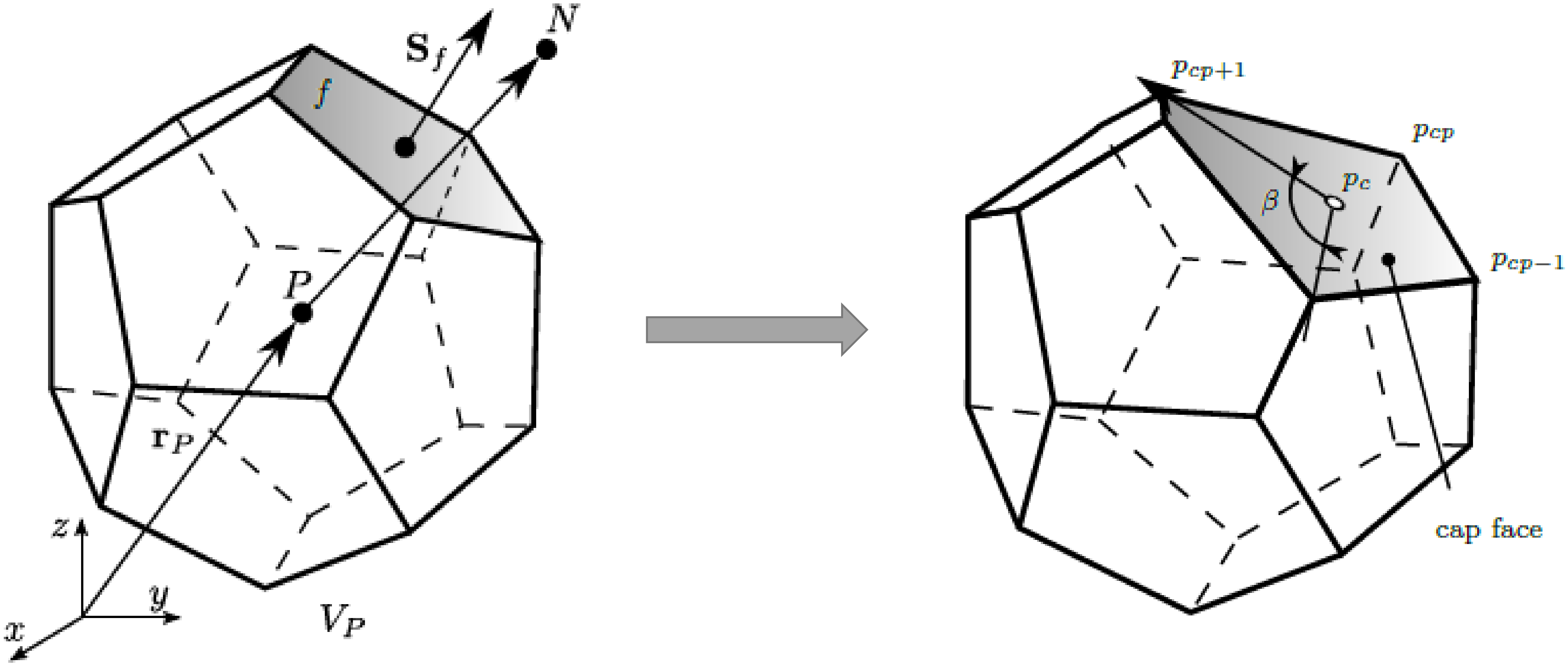
Challenges:
Keeping volume fraction between 0 and 1.
The fluxed volume may not correspond to a divergence-free velocity field.

Investing in a geometric toolbox[5]

We can draw on developments in computational geometry to help in constructing flux polygons in 3D.

1. Intersection of a plane and a line.
2. Intersection of 2 polygons in 3D
3. Intersection of a plane and a polyhedron – clipping and capping
4. Intersection of 2 polyhedral.





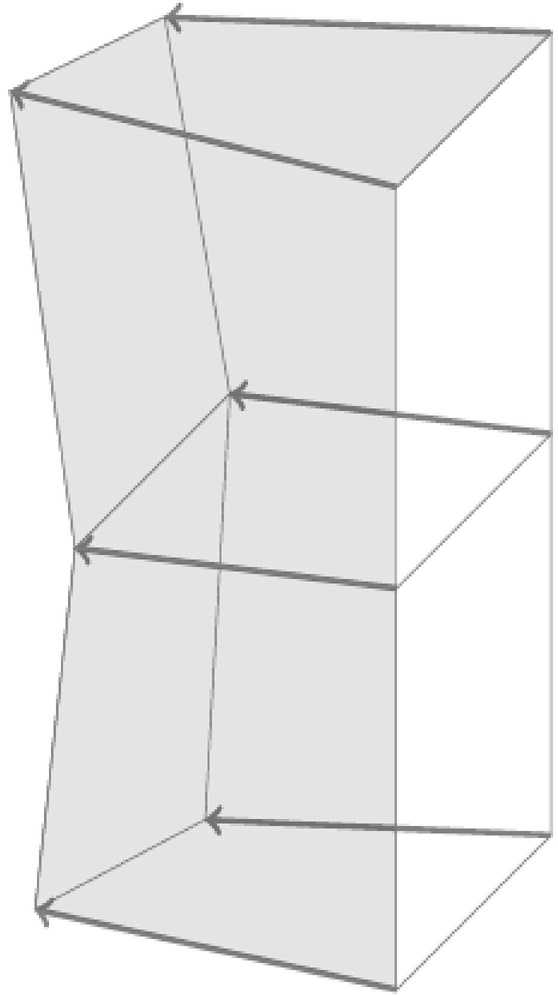
The clipping and capping algorithm[5]

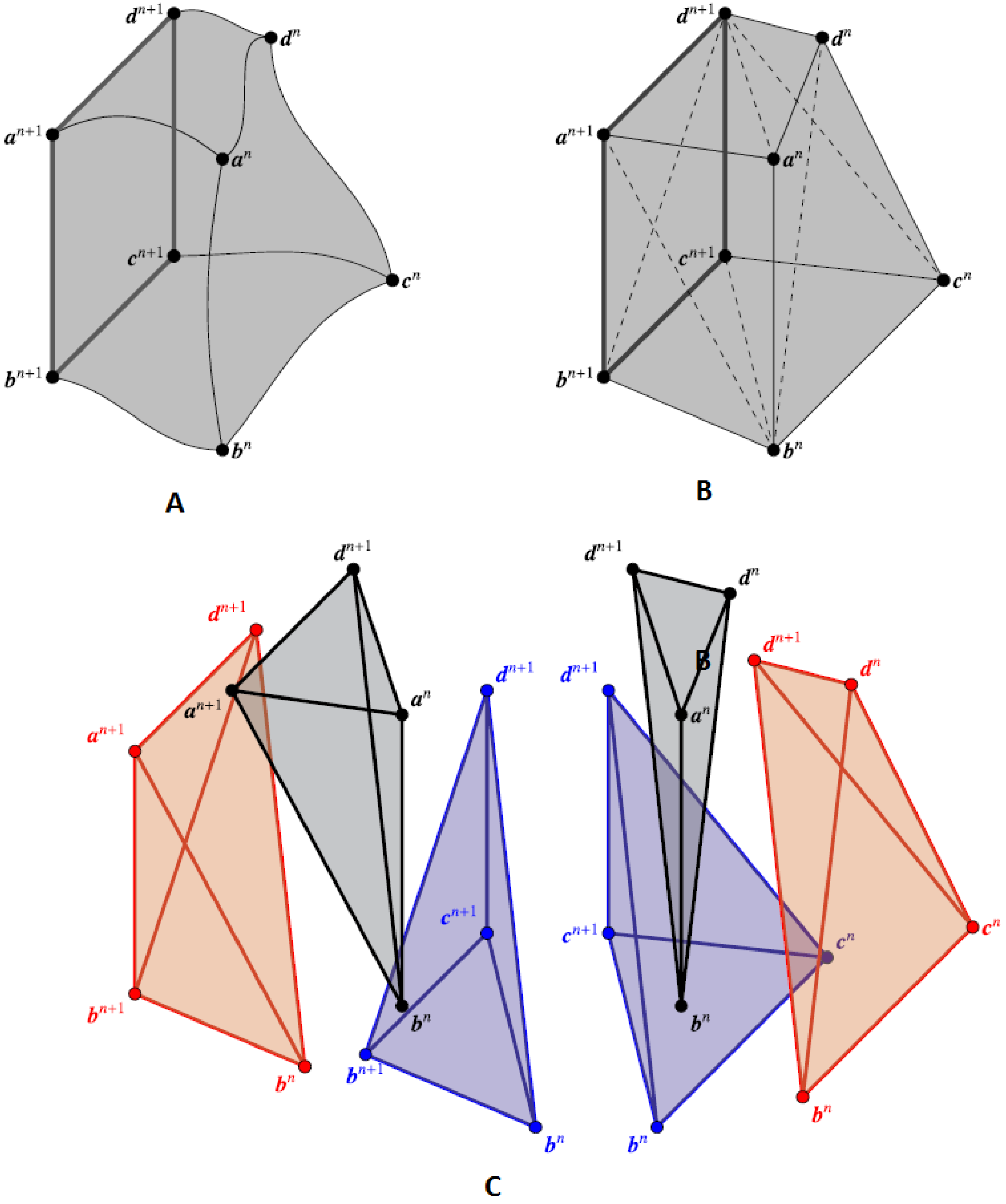
Flux polyhedrons in 3D[6]

Using vertex velocities creates non-overlapping flux polygons.

The faces of the flux polyhedron need not be planar.

Divide the flux polyhedron into tetrahedra (simplexes) to calculate the volume.
‘Tessellation’ of flux polygons/polyhedral.





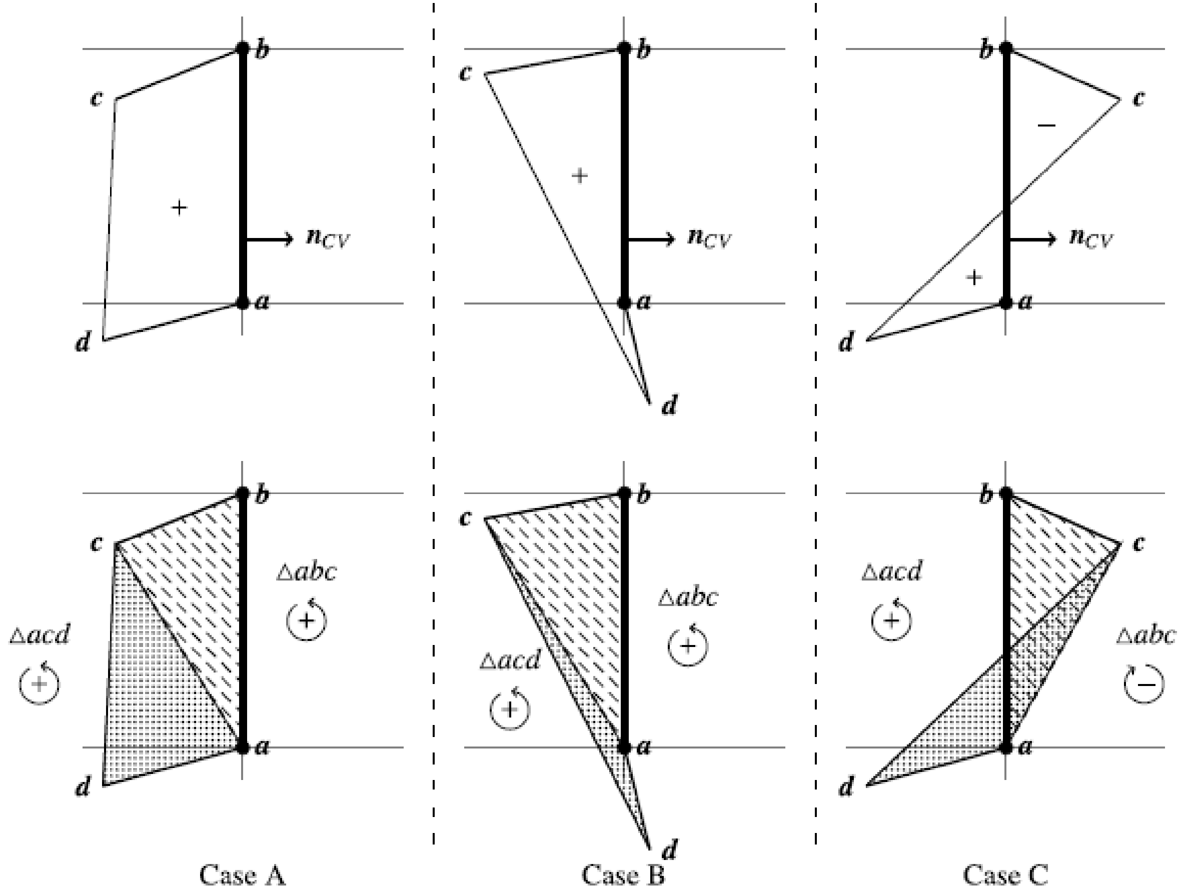
Two levels of discretization are used to calculate the volume. [6]

The levels A and B ensure non-overlapping fluxes. Going from B -> C needs some consistency. Difficult for arbitrary meshes (non-hexahedral cells).

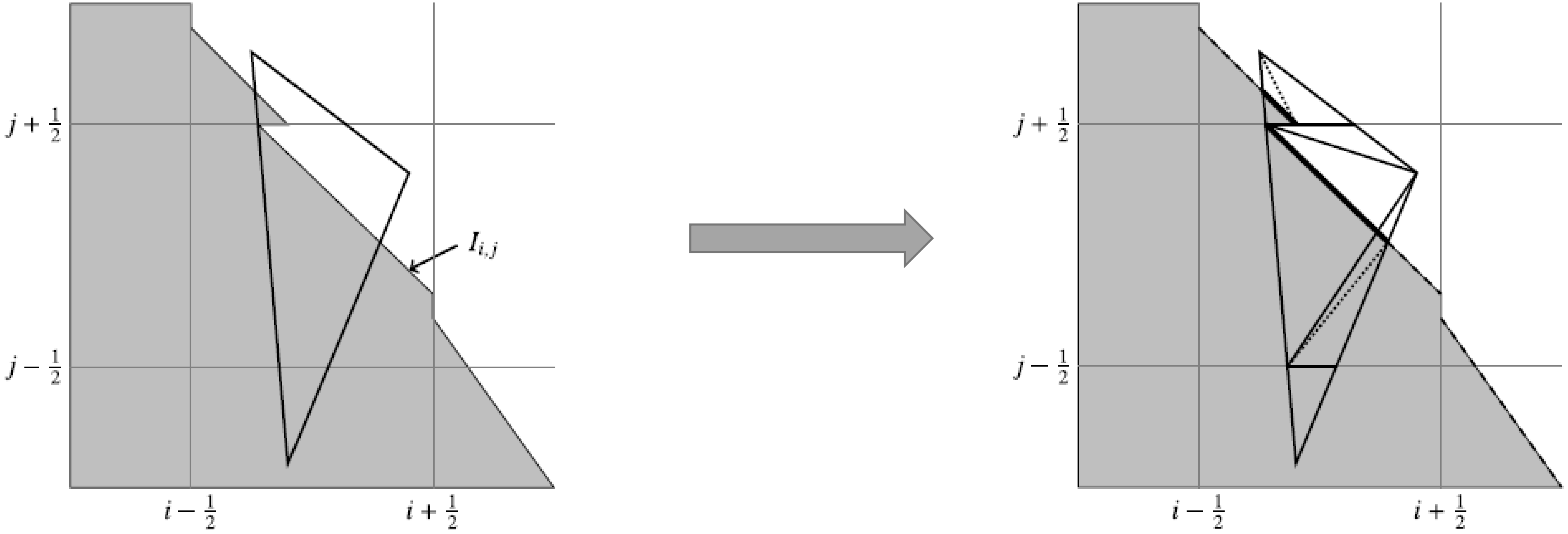
Concave Flux polygons[6]

Divide the flux polygon into triangles, and classify them as positive fluxes and negative fluxes.

This needs a clever ordering of the vertices of the polygon so that the clockwise ordering of vertices implies a positive flux.



Phase fluxes from the flux polygon[6]

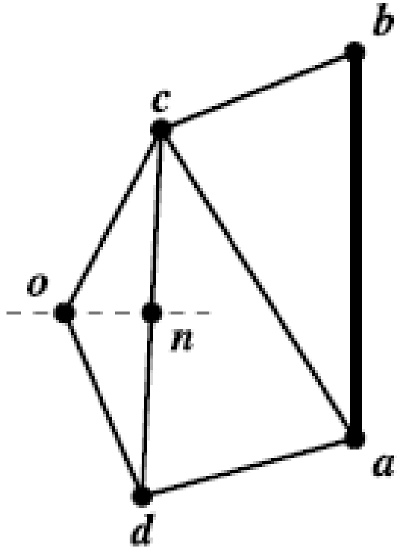


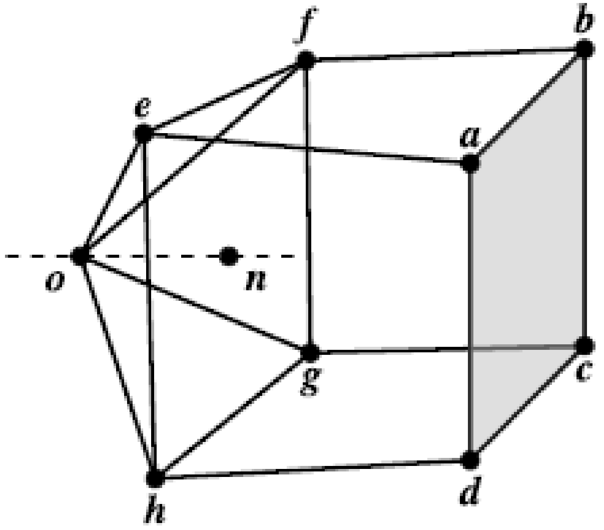
The algorithm cuts the flux polygon with cell faces , and by the interface in each cell. Tessellation is done after every step.

Adjusting for discreet mass conservation[6]

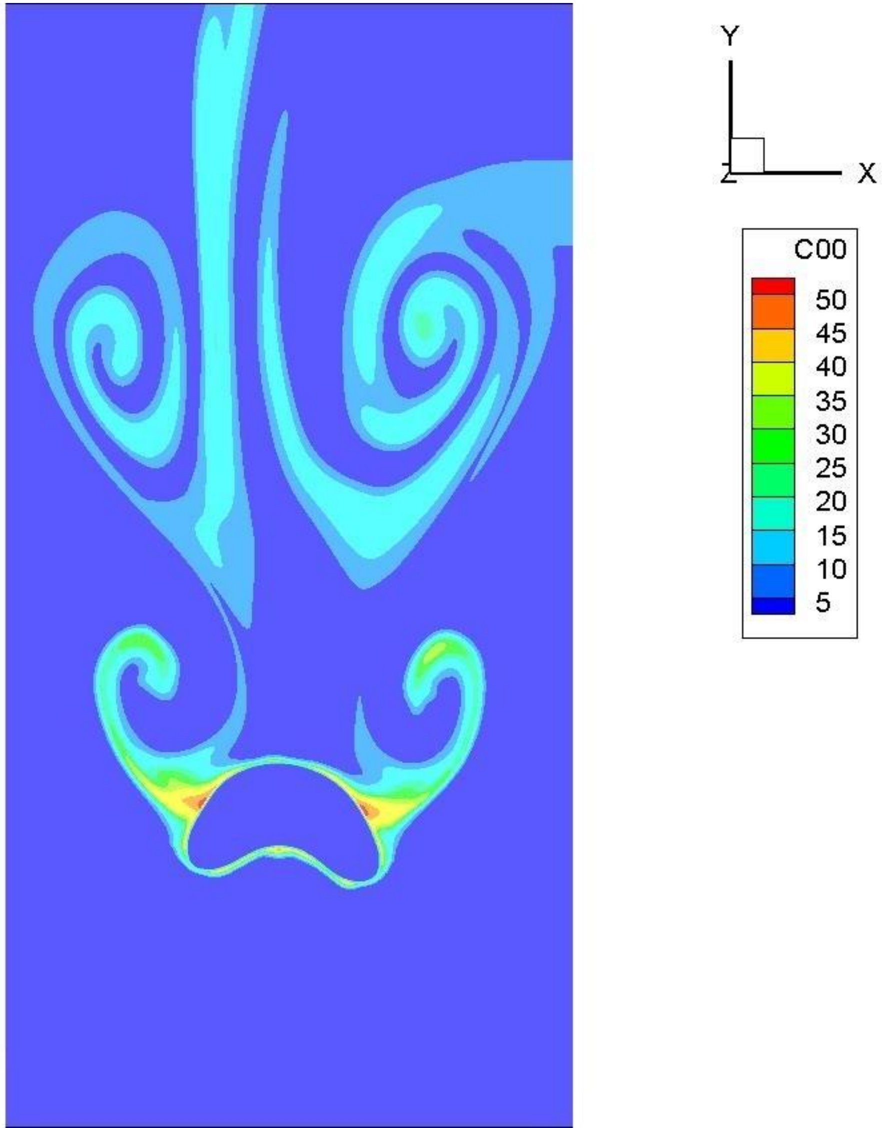
The volume of the flux polygon generally does not exactly match the calculated velocity field at the cell face-center.

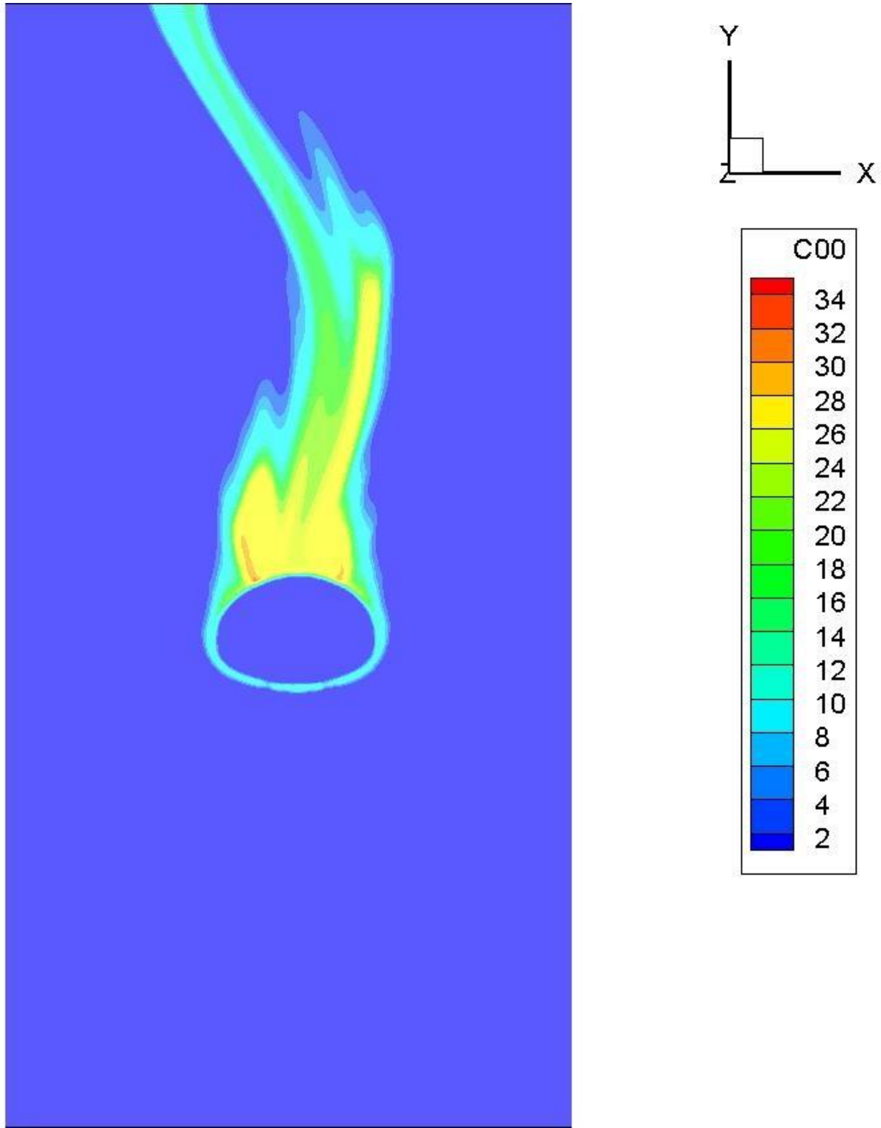
An additional simplex is added to the trailing face of the polygon to get the two volumes to be equal. This ensures discreet mass conservation, provided the calculated velocity field does so.





The effect of surface tension – increased by factor of 3 here

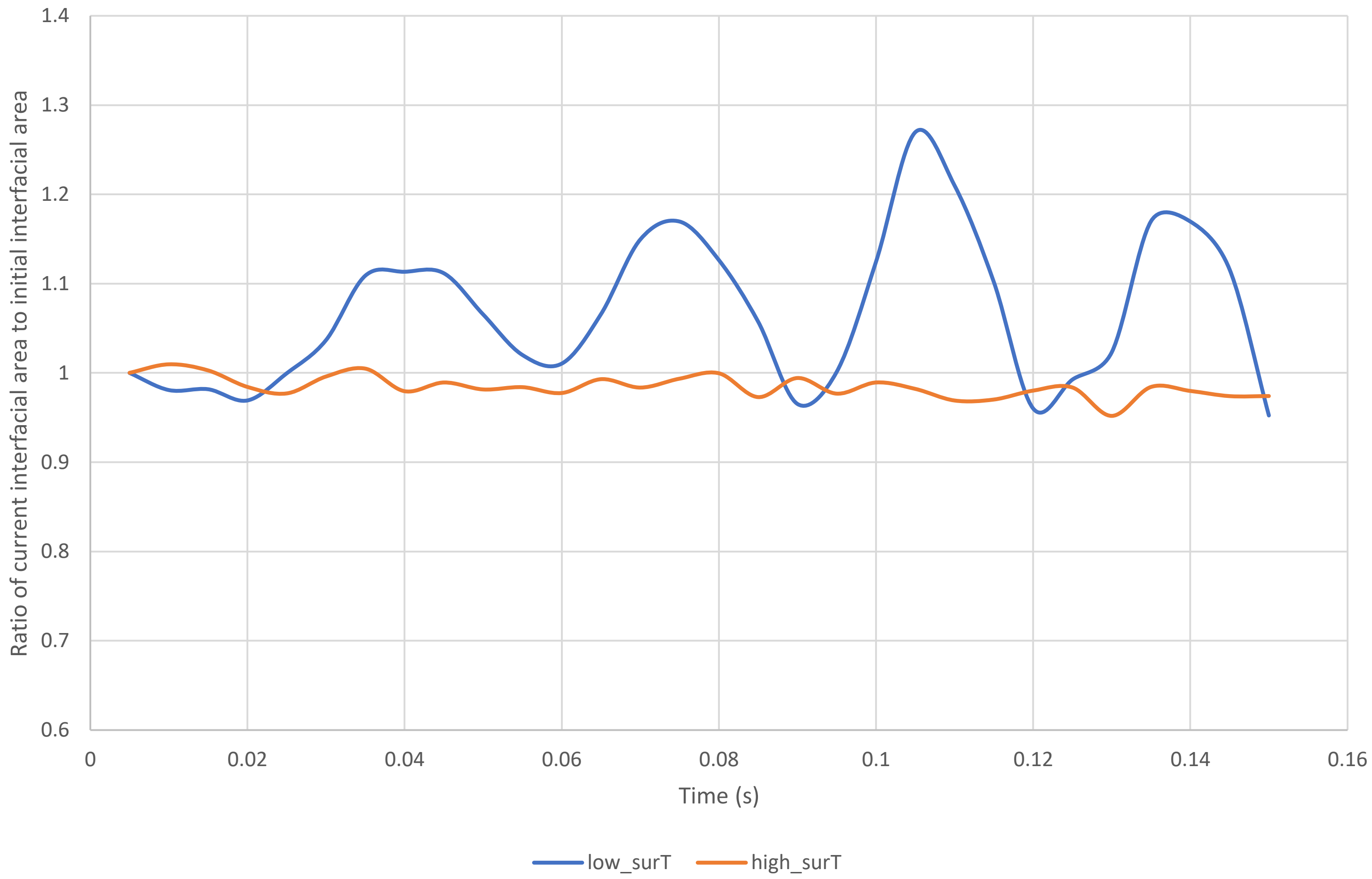




Total interfacial area

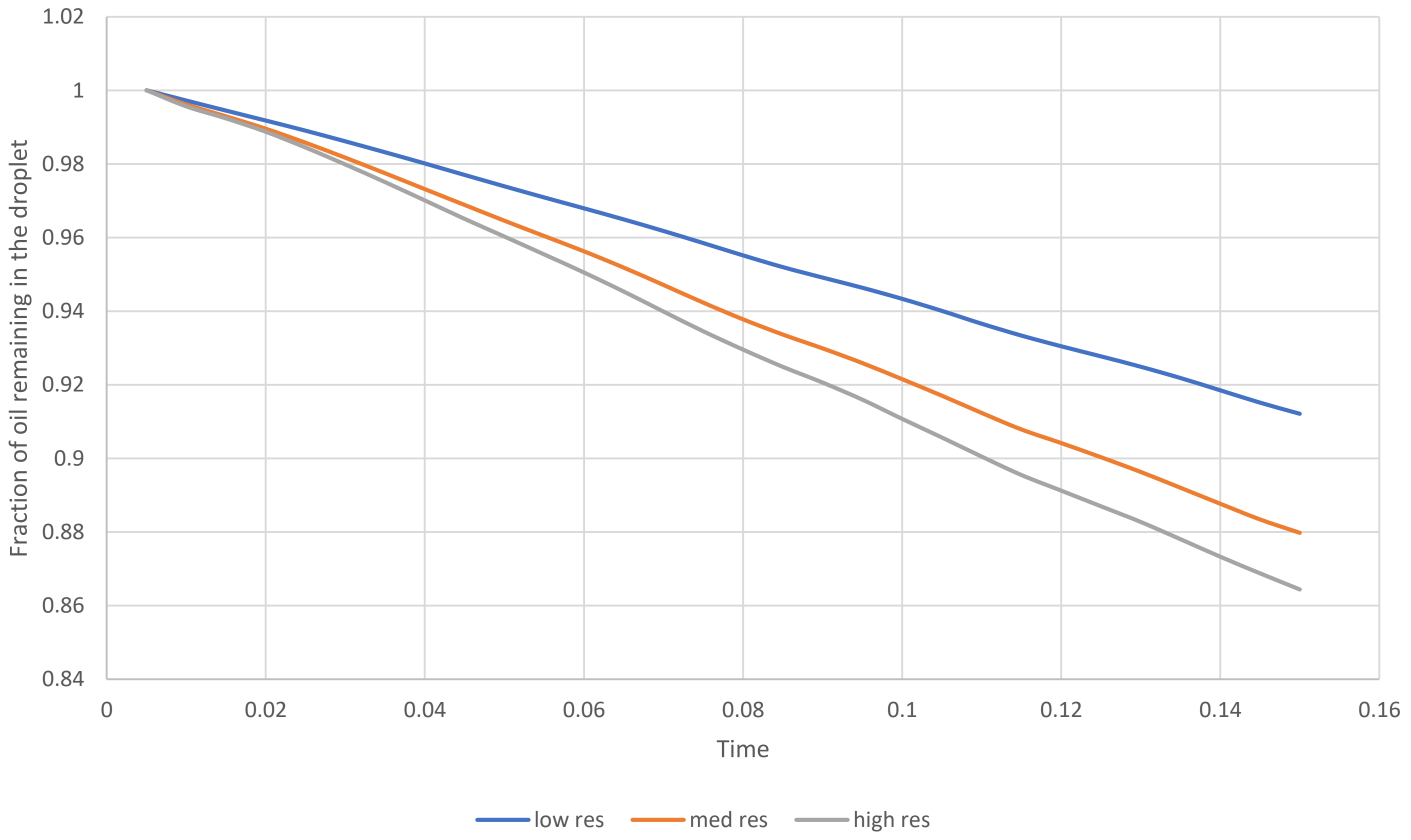
Fluctuations in interfacial area for the low surface tension case have a frequency of about 30Hz.

A Strouhal number calculation based on flow past a cylinder predicts a frequency of 20Hz



Grid convergence study

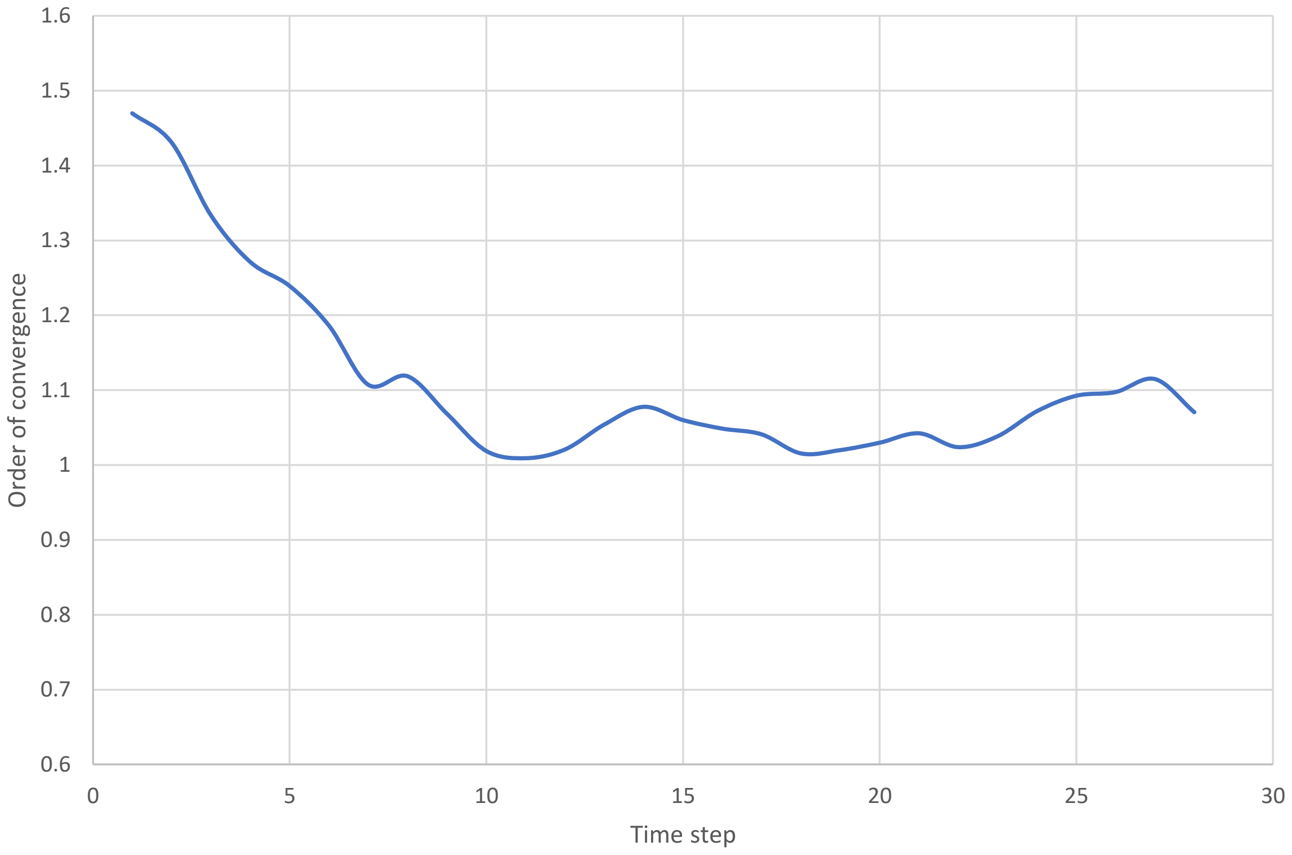
Low resolution: 100*200
Medium resolution: 200*400
High resolution: 400*800



Order of convergence

The scheme is overall 1st order accurate in space

$$p \approx \log\left(\frac{u_{2\Delta x} - u_{4\Delta x}}{u_{\Delta x} - u_{2\Delta x}}\right) \Big/ \log 2$$



Order of convergence

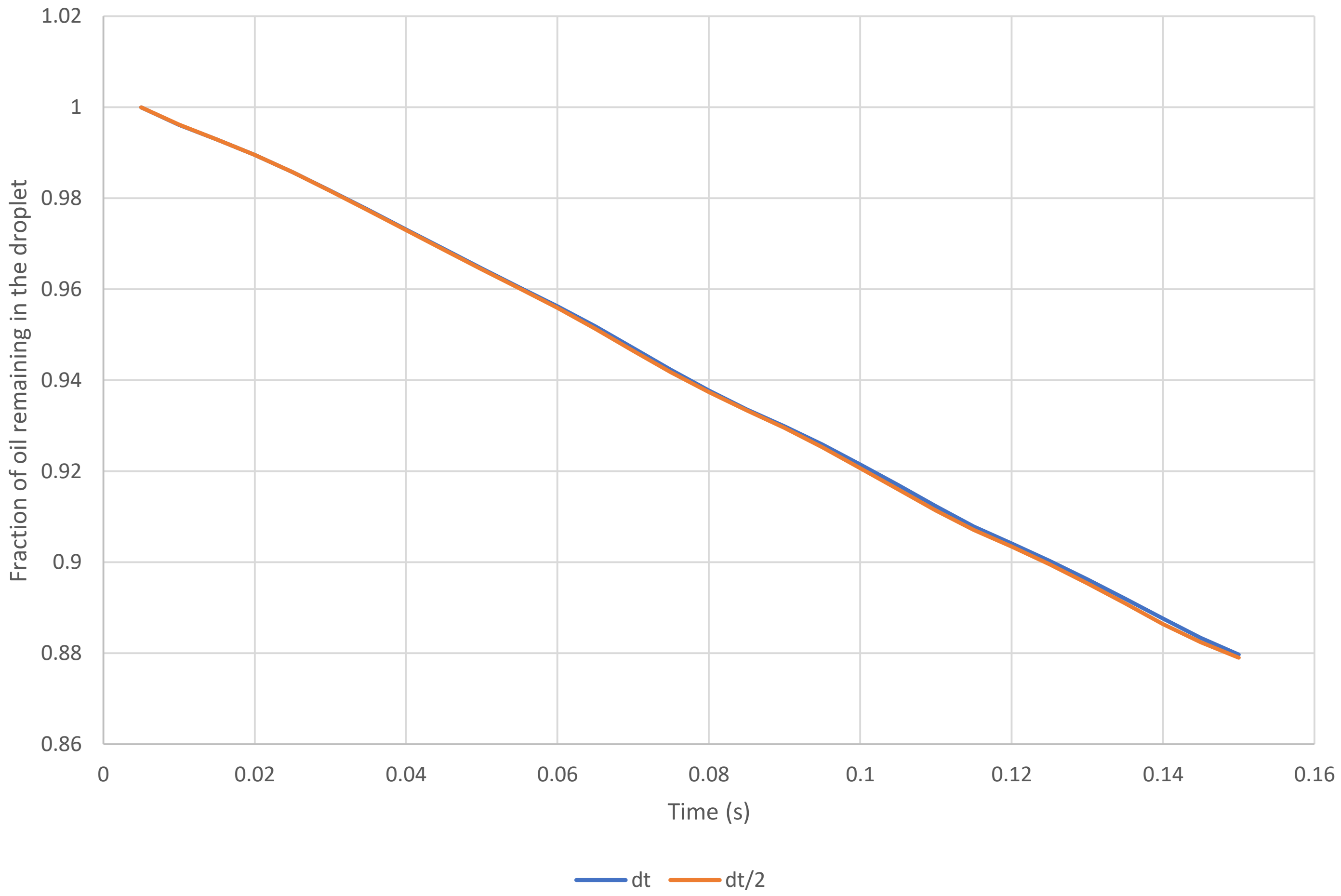
A fully explicit scheme is used.

The advection CFL condition is stronger than diffusion CFL condition as mass diffusivities is small in this problem.

Thus dt ~ dx,dy

To ensure that time discretization errors are no the dominant error terms, a simulation was run at half the admissible courant number. The solution did not change.

Thus, the spatial order of accuracy is indeed 1.



References

1. Weller, H. G., Tabor, G., Jasak, H., & Fureby, C. (1998). A tensorial approach to computational continuum mechanics using object-oriented techniques. Computers in physics, 12(6), 620-631.
2. http://web.mit.edu/rgd/www/
3. He, P., & Ghoniem, A. F. (2017). A sharp interface method for coupling multiphase flow, heat transfer and multicomponent mass transfer with interphase diffusion. *Journal of Computational Physics*, *332*, 316-332.
4. Rider, W. J., & Kothe, D. B. (1998). Reconstructing volume tracking. *Journal of computational physics*, *141*(2), 112-152.
5. Maric, T., Marschall, H., & Bothe, D. (2013). voFoam-a geometrical volume of fluid algorithm on arbitrary unstructured meshes with local dynamic adaptive mesh refinement using OpenFOAM. *arXiv preprint arXiv:1305.3417*.
6. Owkes, M., & Desjardins, O. (2014). A computational framework for conservative, three-dimensional, unsplit, geometric transport with application to the volume-of-fluid (VOF) method. *Journal of Computational Physics*, *270*, 587-612.