The 2D Hysing benchmark

Lionel GAMET

November 17, 2021

1 Introduction

This case is a reference test case for VoF simulations. It is an elementary quantitative benchmark configuration that was originally established in 2D by Hysing et al. [1]. The case consists in a single rising bubble in a quiescent liquid. Both test cases number 1 and 2 as described by Hysing et al. [1] are modeled here.
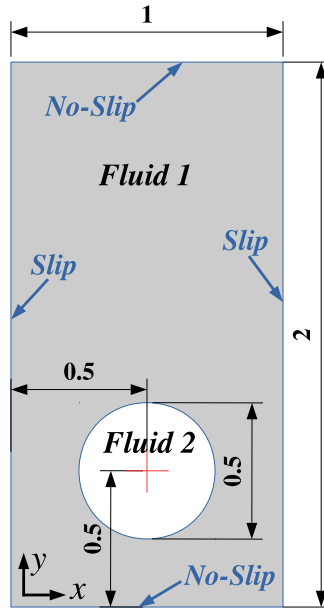

Figure 1: Configuration and boundary conditions for 2D bubble benchmark.

The 2D case setup is schematized in Figure 1. The first phase is the liquid, while second phase is the gas. The properties of both phases are summarized in the table 1 below. The last column reports the surface tension between the fluids.

Table 1: Fluid properties for test cases C1 and C2.

Case	ρ_1 (kg m^{-3})	ρ_2 (kg m^{-3})	μ_1 (kg m^{-1} s^{-1})	μ_2 (kg m^{-1} s^{-1})	σ (N/m)
C1	1000	100	10	1	24.5
C2	1000	1	10	0.1	1.96

Gravity is taken as $g = 0.98$ m/s^2. The chosen values are not representative of real fluids. This test case has originally been designed for comparison purposes between CFD solvers. The domain is of size 1×2 in the x and y directions.

2   Setting up the test case in OpenFOAM

This test case uses incompressible VoF solvers in OpenFOAM, namely `interFoam` or `interIsoFoam` solvers.

A uniform Cartesian grid built with `blockMesh` is used for 2D simulations. The default grid size is 20×40. When running the grid sensitivity script (see section 3), finer grid levels will be used. The bubble is initialized as a cylinder in 2D using the `setAlphaField` utility.

NB: We use the `invert` *option in* `setAlphaFieldDict` *to reverse the initial field from a drop to a bubble.*

In the `fvSchemes` file, a Crank-Nicolson second order time scheme with blending coefficient 0.9 is chosen. `Gauss limitedLinearV 1` is used to treat the convective term, and `Gauss vanLeer` is used for the α convective term for MULES simulations. The `Gauss linear` scheme is used by default for all gradient terms.

In the `fvSolution` file, the GAMG implicit solver is used for pressure terms, while the smooth solver is used for the velocity. The `PIMPLE` algorithm uses 3 `nOuterCorrectors` and 3 PISO correctors (`nCorrectors`=3). It was found that `momentumPredictor` needed to be set to true to get a correct solution in terms of rising velocity and bubble circularity, in particular with isoAdvector. Both isoAdvector and MULES numerical parameters are present and appear separately in `fvSolution`, so that bot `interFoam` and `interIsoFoam` can be run from the same input files.

Computations are done by default at a constant CFL of 0.075. However, a constant time step is set-up by the grid sensitivity script, starting at $\Delta t = 0.002$ s for the 20×40 coarsest level and reduced for finer grids to keep the maximum CFL number in the order of 0.05. Computations are run up to time $t = 3$ s in 2D.

Post-processing quantities of interest are described in details in [1]. These are the vertical position of the bubble centroid, the bubble rise velocity, the bubble circularity, bubble volume and area. All these quantities are computed through a coded functionObject inlined in the `controlDict` file. The results appear in the log file of the solver. We use a `sampledIsoSurface` object, defined as the isosurface $\alpha = 0.5$, to compute the bubble area. We also use volume integrals of the gas fraction overall the computational domain to compute the

bubble volume, centroid and velocity. A bubble equivalent diameter named D_A is computed from the bubble volume. A bubble equivalent diameter named D_B is computed from the bubble area. The circularity is defined in 2D as the ratio D_A/D_B. This number takes the value 1 at $t = 0$ as the bubble is initialized as a perfect cylinder, and then decreases with time as the bubble rises and deforms.

Finally, the bubble shape is output at `writeTime` frequency through a `surfaces` sampling functionObject, based upon an isosurface $\alpha = 0.5$. Both interpolated and constant iso bubble shapes are output.

3   Running the case

The `Allrun` script will run the default grid size at 20×40 on Hysing case number 2 by default. The grid is first constructed by running `blockMesh`. Then the `setAlphaField` utility is run to initialize the bubble as a cylinder. Finally, the application (default to `interFoam`) is run. Post-processing quantities of interest (bubble volume, area, centroid, velocity and circularity) are extracted from the solver log file through a `grep` command.

The `Allrun_sensitivity` is a script at a level above `Allrun`. It is used to run a grid sensitivity for solvers `interFoam`, `interIsoFoam`, and also the enhanced version of isoAdvector by H. Scheufler [2], now partially integrated in the v2006 with the plicRDF reconstruction scheme, but also publicly available from the TwoPhaseFlow git repository. The `Allrun_sensitivity` will create subdirectories ready to run, with a number of cells 20, 40, 80, 160, 320 and 640. It will also create subdirectories for the 2 different cases C1 and C2. You need to enter each generated subdirectory and type `Allrun` to start the computation.

For comparisons, reference 2D data can be found at the bibliography link. OpenFOAM results are also provided for reference.

4   The prismatic grid variant

A variant of the case setting is proposed on prismatic grids. In that case, you can choose the grid generator tool (default to Pointwise) in `Allrun_sensitivity`. For this kind of grids, the default `gradSchemes` has been changed from `Gauss linear` to `pointCellsLeastSquares`, and `nNonOrthogonalCorrectors` was set to 1.

References

[1] S. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, and L. Tobiska, "Quantitative benchmark computations of two-dimensional bubble dynamics," *I.J.N.M.F.*, vol. 60, no. 11, pp. 1259–1288, 2009. [Online]. Available: http://www.featflow.de/en/benchmarks/cfdbenchmarking/bubble.html

[2] H. Scheufler and J. Roenby, "Accurate and efficient surface reconstruction from volume fraction data on general meshes," *J. Comp. Phys.*, vol. 383, pp. 1 – 23, 2019.