IsoAdvector: Free, fast and accurate VOF on arbitrary meshes

Johan Roenby[1], Henrik Bredmose[2], and Hrvoje Jasak[3]

[1]*DHI, Department of Ports and Offshore Technology, Denmark, jro@dhigroup.com*
[2]*Technical University of Denmark, Department of Wind Energy, Denmark, hbre@dtu.dk*
[3]*University of Zagreb, Department of Mechanical Engineering and Naval Architecture, Croatia hrvoje.jasak@fsb.hr*

In many CFD applications, one needs to track a surface moving with the flow. Often, this surface represents the sharp interface between two fluids, such as air and water. An example from the marine sector is the water surface tracked in simulations of wave loads on offshore and coastal structures. In industrial applications, we encounter sharp fluid interfaces e.g. in modelling of inkjet printers, metal and plastic casting processes, automotive aquaplaning etc.

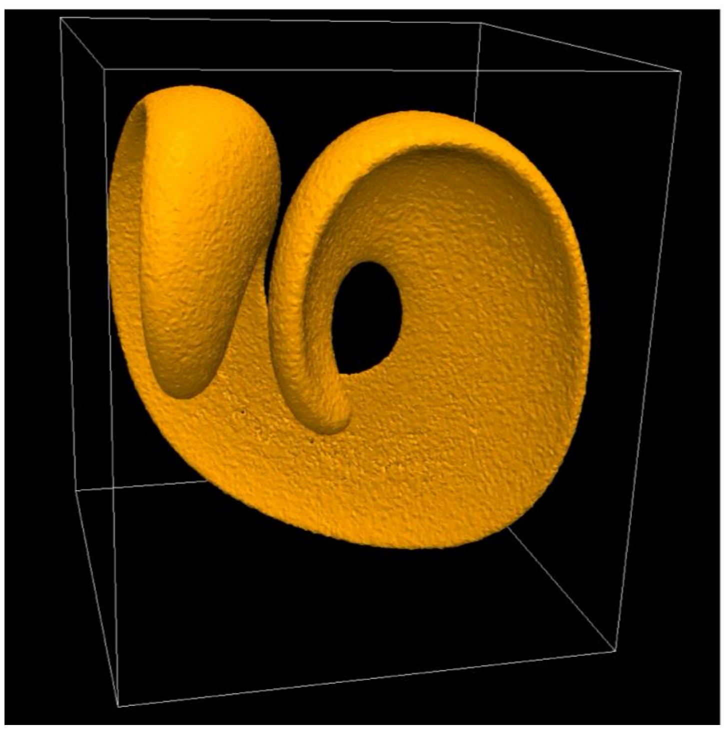

Figure 1. Initially spherical surface advected in a spatially varying velocity field. IsoAdvector on random tet mesh.

In all these applications, it is vital to have a fast and accurate algorithm for advection of a surface in a given velocity field. A snapshot from an "academic" surface advection test case is shown in Fig. 1. In OpenFOAM, the interFoam solver family is dedicated to flows, where the pressure-velocity solver is coupled with the advection of a sharp fluid interface. These solvers apply the OpenFOAM specific algebraic Volume-of-Fluid (VOF) scheme called MULES (Multidimensional Universal Limiter with Explicit Solution) for the task of advecting the sharp interface[1]. The MULES scheme has a number of desirable properties:

1) it preserves the volume of fluid, i.e. it does not artificially create or destroy fluid,
2) it keeps the so-called volume fraction field in the physically meaningful range between 0 and 1,
3) the interface stays sharp to within a few cell widths,
4) it works on unstructured meshes both in 2D and 3D,
5) it is efficient so only a minor fraction of the calculation time is spent on interface advection.

Unfortunately, it can also be shown that the MULES scheme may sometimes be rather inaccurate in its solution of the interface advection problem. In fact, as illustrated in Fig. 2 (top row), the scheme does not seem to converge to the correct solution even for very simple test cases.

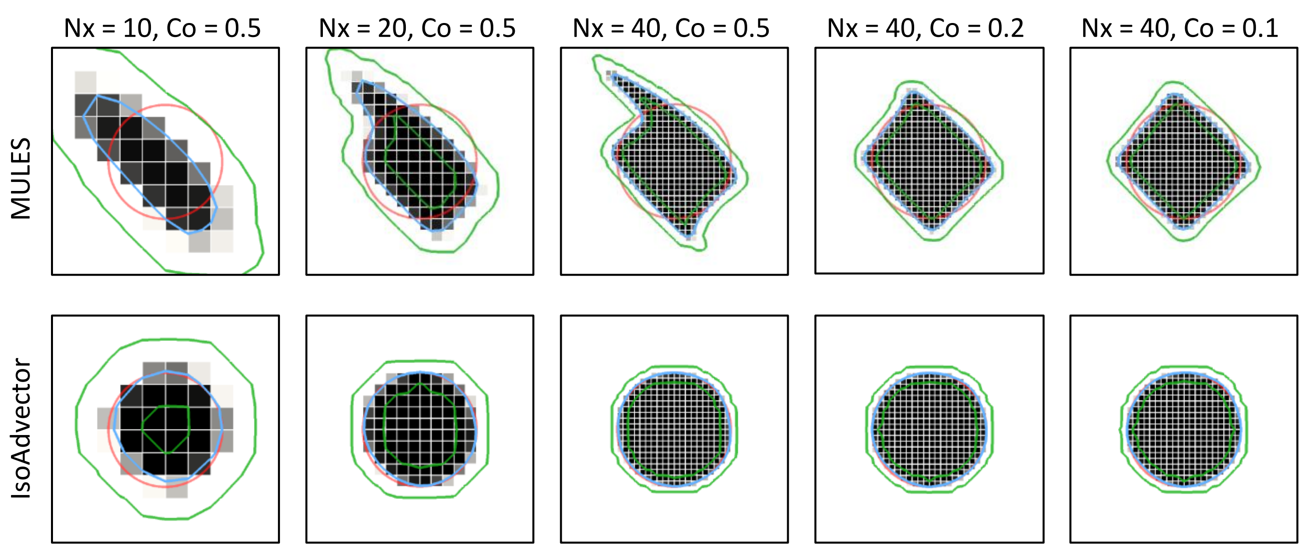

Figure 2. Shape of an initially circular volume of fluid after 4 seconds advection in a constant, uniform velocity field U = (1, 0.5). Red circles are exact solution. Blue curve is VOF = 0.5 contour and green curves are VOF = 0.01 and 0.99 contours. Figures from [2].

Based on this observation, we have sought an alternative to MULES, which has all the properties listed above *and* gives accurate solutions to the interface advection problem. Being unable to find such a scheme in the literature, we have developed a new VOF algorithm denoted IsoAdvector. The new scheme is based on geometric considerations both in the interface *reconstruction* step and in the interface *advection* step. As such, the scheme belongs to the category of geometric VOF algorithms. However, in contrast to most existing geometric VOF algorithms, IsoAdvector is not restricted to hexahedral meshes, but is by design invented for general polyhedral cells (see Fig. 1).

For the cases tested so far, IsoAdvector performs better than MULES on all the parameters mentioned above. Most notably, it is much more accurate, as illustrated in Fig. 2 (bottom row), and significantly faster. It also keeps the interface sharper, as illustrated in Fig. 3. The main restriction is that the geometrical considerations implemented in the IsoAdvector algorithm only make sense for interface Courant numbers less than unity. However, good accuracy can be expected even for Courant numbers close to this limit.

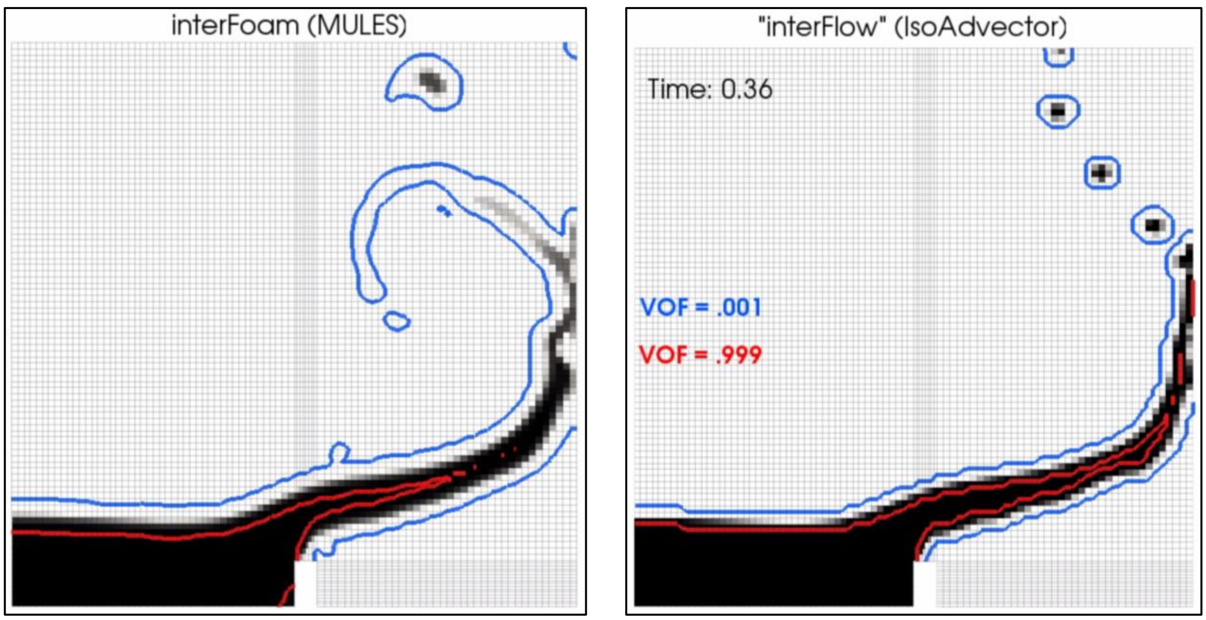

Figure 3. Snapshot from damBreak case. Left: interFoam (2.2.0) using MULES. Right: interFlow which is interFoam with MULES replaced by IsoAdvector. The distance between VOF = 0.001 and 0.999 contours illustrates the interface width.

The IsoAdvector algorithm is implemented as an extension module to OpenFOAM. More details about the algorithmic design and performance of the scheme are given in the presentation and can be found in a recently submitted preprint[2]. The code will be released as open source together with this publication. It is our hope that the IsoAdvector scheme will be used and further developed by the CFD community, and will eventually result in an overall improvement of solution quality for interfacial flow simulations.

Literature

[1] Deshpande, Suraj S., Lakshman Anumolu, and Mario F. Trujillo. "Evaluating the performance of the two-phase flow solver interFoam." Computational science & discovery 5.1 (2012): 014016.

[2] Roenby, Johan, Henrik Bredmose, and Hrvoje Jasak. "A Computational Method for Sharp Interface Advection." arXiv preprint arXiv:1601.05392 (2016).