CFD Direct

The Architects of OpenFOAM

Home   OpenFOAM   Cloud   Training   Support   Contact

Interface Capturing in OpenFOAM

OpenFOAM is software for computational fluid dynamics with a strong reputation for multiphase flows. It includes several solvers that capture interfaces between liquid-gas phases (e.g. water-air) based on the volume of fluid (VoF) method. The method solves transport equations for one or more phase fractions "α" ($0 \leq \alpha \leq 1$), where the surface at $\alpha = 0.5$ describes the interface. Spanning a period of 20+ years, Henry Weller of CFD Direct made VoF in OpenFOAM robust, usable and efficient by key innovations, including:

- the MULES algorithm — semi-implicit and second order in time — to ensure α remains between strict bounds of 0 and 1;
- the interface compression scheme, based on counter-gradient transport, to maintain sharp interfaces during a simulation.

Users have adopted this industrial-strength methodology for small scale flows in mechanical and medical engineering. It works for intermediate scales in process engineering and for the largest scales of civil engineering hydraulics and marine hydrodynamics.

Interpolation for interface capturing

In July 2020, CFD Direct released a framework for **surface interpolation in interface capturing** in the public, development line of OpenFOAM (`OpenFOAM-dev`). All interpolation schemes, used for flux calculation in advection, operate with the existing MULES-based solvers, e.g. *interFoam.* They are inherently **robust** with bounded, conservative numerics that can operate 2nd-order in time and efficiently with large time steps. The **redesign** of flux interpolation is **extensible** to new schemes. There is improved **usability** with a simple selection

OpenFOAM Training

28 Sep Virtual, Europe

18 Oct Virtual, Americas

02 Nov Virtual, Americas

23 Nov Virtual, Europe

30 Nov Virtual, Europe

25 Jan Virtual, Americas

07 Feb Virtual, Europe

22 Feb Virtual, Europe

14 Mar Virtual, Americas

of interpolation scheme in the *fvSchemes* file, e.g. the standard interface compression with `vanLeer` interpolation:

```
div(phi,alpha)      Gauss interfaceCompression vanLeer 1;
```

Piecewise-linear interface calculation (PLIC)

To accompany the new framework, Jakub Kníř (formerly, CFD Direct) developed a new family of interpolation schemes based on **piecewise-linear interface calculation** (PLIC). PLIC represents an interface by surface-cuts which split each cell to match the volume fraction of the phase in that cell. The surface-cuts are oriented according to the point field of the local phase fraction. The phase fraction on each cell face — the interpolated value — is then calculated from the amount submerged below the surface-cut.
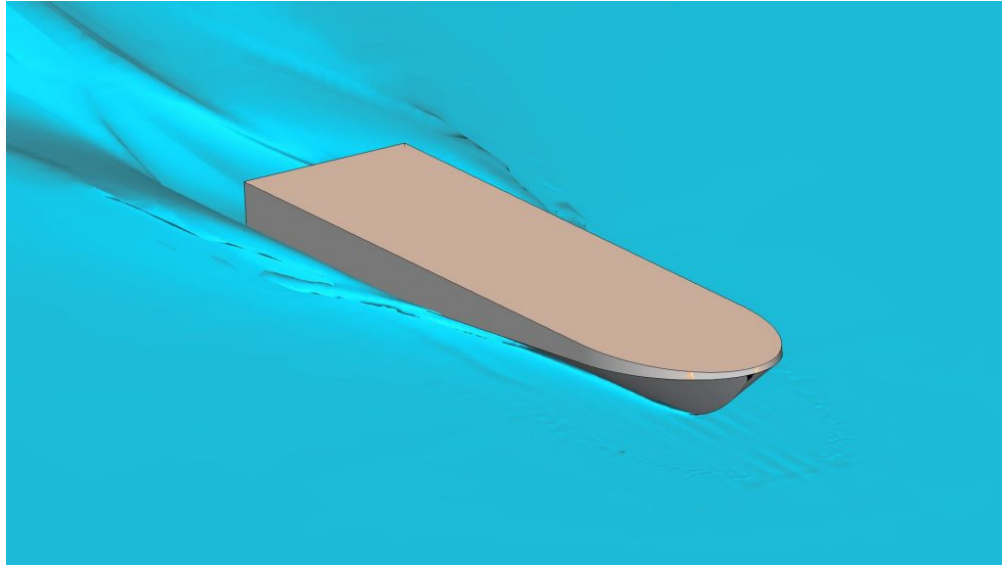
The basic PLIC method generates a single cut so cannot handle cells in which there are multiple interfaces or where the interface is not fully resolved. In those cells, the interpolation reverts to an alternative scheme, typically standard interface compression. PLIC, with a fallback to interface compression, produces robust solutions for real engineering cases. It can run with large time steps so can solve problems like hydrodynamics of a planing hull, with rigid body motion of the hull (above). The user selects PLIC by the following setting in *fvSchemes*.

```
div(phi,alpha)      Gauss PLIC interfaceCompression vanLeer 1;
```

OpenFOAM v8 Example Cases: Tank Sloshing in Zero Gravi...

▶

Monthly Newsletter

[First Name]

[Last Name]
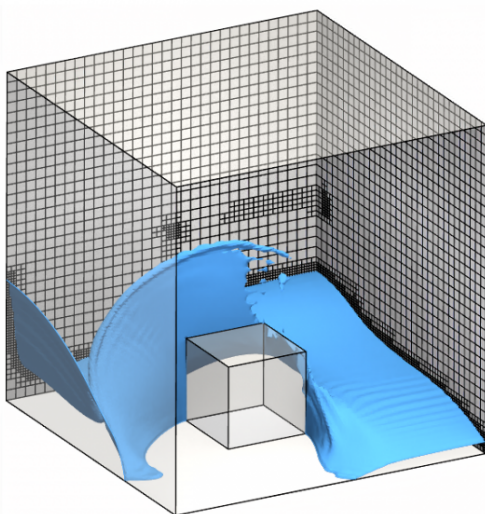
[Email Address]

[Type of ⌄]

[Countr ⌄]

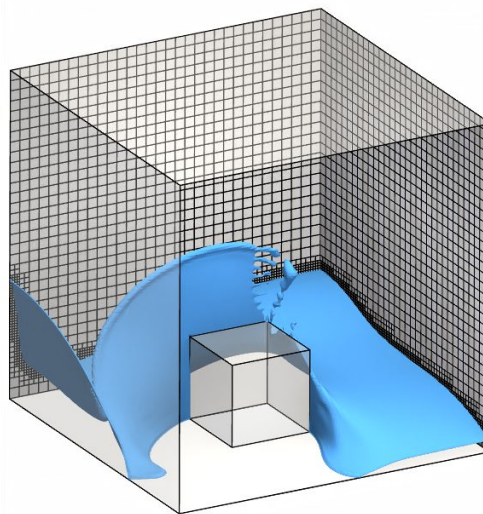☐ * I consent to my data being processed in accordance with the Privacy Policy

Subscribe!

Multicut PLIC (MPLIC)

A separate multicut PLIC (MPLIC) scheme extends PLIC to handle multiple surface-cuts. Where a single cut is insufficient, MPLIC performs a topological face-edge-face walk to produce multiple splits of a cell. If that is still insufficient, MPLIC decomposes the cell into tetrahedrons on which the cuts are applied. The extra cutting carries an additional computational cost but requires no fallback. The user selects MPLIC by the following setting in the *fvSchemes* file:

```
div(phi,alpha) Gauss MPLIC;
```

Latest News on Twitter

New pack (20210922) of #OpenFOAM-dev for @ubuntu installable on Windows 10, with Docker images for @Linux and macOS https://t.co/QzGL00 sKvg18 hours ago

The comfort function object is extended to

dynamic refinement with interface compression

dynamic refinement with MPLIC

Refinement patterns

The PLIC and MPLIC methods are **more precise** than interface compression for meshes with **refinement patterns**. With interface compression, the `damBreakWithObstacle` example above reveals numerical oscillations of the interface, due to dynamic mesh refinement. With (M)PLIC, the oscillations disappear. Interface oscillations also occur for the planing hull when the mesh includes refinement patterns. The oscillations causes excessive entrainment of air beneath the hull, leading to an inaccurate prediction of forces on the hull. (M)PLIC reduces these oscillations for a better prediction of the forces.

High shear

Variants of the PLIC and MPLIC schemes are also available which use velocities at the face points to calculate the face flux. These `PLICU` and `MPLICU` schemes are likely to be more accurate in regions of interface under high shear force.

● Chris Greenshields   📅 2nd July 2020   📁 Free Software

← CFD Direct Year 5: 2019-2020

OpenFOAM v8 User Guide: 1 Introduction →

calculate draught rate in **#OpenFOAM-dev**, contributed by **@HolzmannCFD https://t.co/ZdoWwCArGW** 4 days ago

We design our OpenFOAM Training so that users can be productive with **#OpenFOAM: https://t.co/KixhnYo94E** 4 days ago

Contact Us   Twitter   LinkedIn   YouTube   Website Terms of Use   Privacy