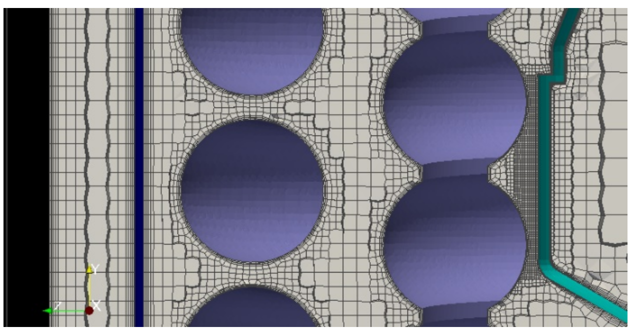
Version 7.04 (Apr 26, 2017)

Pre-processing

cfMesh Library

- GPL meshing library from Creative Fields d.o.o. (https://sourceforge.net/projects/cfmesh/)
- Generation of meshes of arbitrary cell types. The currently implemented applications generate:
  - Cartesian cells in both 2D and 3D
  - Tetrahedral cells
  - Arbitrary polyhedral cells
  - Boundary layer cells
- Includes parallelization using both shared memory parallelization (SMP) and distributed memory parallelization using MPI.
- Efficient memory usage

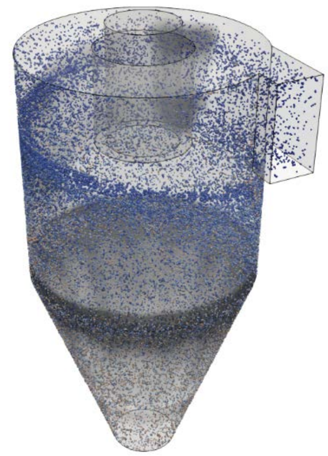


Cartesian mesh on complex geometry with boundary layer cells generated with cfMesh application

Solvers

Multiphase

- **`pimpleParcelSolver`**: incompressible solver with 2-way coupled Lagrangian particle tracking.
- **`vofDyMSolver`**: dynamic mesh variant of 2-phase incompressible multiphase solver. [OpenFOAM Foundation]
- Semi-implicit version of flux corrected transport equation for volume fraction in VOF-based solvers [OpenFOAM Foundation].
- Enhancements to interface compression algorithm in VOF-based solvers [Applied CCM].
- Added transient cell zone injection for Lagrangian particle tracking [OpenFOAM Foundation].



Cyclone tutorial solution using Lagrangian particles and erosion model with `pimpleParcelSolver` multiphase solver

Utilities

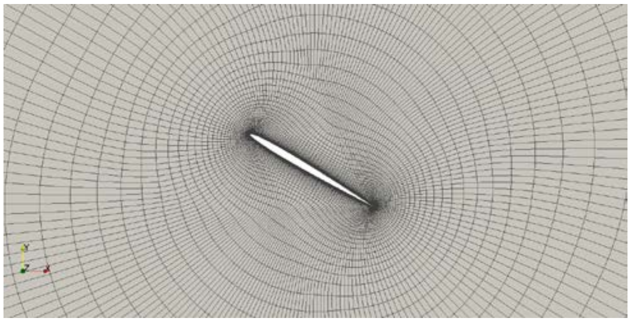
Parallel

- Decompose and reconstruct models in parallel with **`redistributePar`** [OpenFOAM Foundation]:
  - will behave as **`decomposePar`** when run on an un-decomposed case (or with *-decompose* option) and decompose the mesh and fields for a single time step.
  - if the number of domains in the *decomposeParDict* dictionary is set to 1 (or with *-reconstruct* option) will behave as **`reconstructParMesh`** or **`reconstructPar`**
  - if neither the -decompose nor -reconstruct options are provided will operate in redistribution mode and re-distribute a single time step.
- all decomposition constraints, *i.e.* zones, sets, etc, integrated into decomposition libraries and available to all applications.

Library enhancements

Interpolation

- Radial Basis Function (RBF) interpolator. Implemented in RBF mesh motion solver [LEMOS project; Frank Bos and and Dubravko Matijasevic].



Mesh motion of flapping wing (plunge + rotation) with Radial Basis Function (RBF) mesh motion solver

Discretization

- Second-order gradient scheme using nodal average Green-Gauss, e.g. `grad(p) NAGauss`. [Applied CCM]
- Corrected face tangent surface normal gradient scheme with non-orthogonal correction, e.g. `correctedFT`. [Applied CCM]
- Fixed delta calculation at patches for least squares gradient (LSQ) to remove error. LSQ now second order accurate in internal cells and boundaries. [Applied CCM]

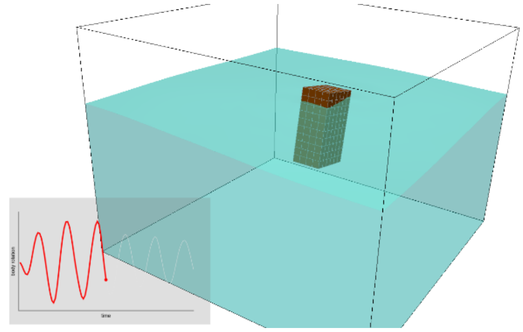
Linear Algebra

- Added stabilized variant of conjugate gradient, non-symmetric solver (`PBiCGStab`). [OpenFOAM Foundation]
- New `SPAI(0)` preconditioners and smoothers for symmetric and non-symmetric matrices suitable on very large parallel applications. [Applied CCM]

Models

Rigid Body Dynamics

- Rigid body dynamics of one or more bodies linked with one or more constraints (nDoF). [OpenFOAM Foundation]
- Mesh motion library updated to handle rigid body dynamics. [OpenFOAM Foundation]



Floating object tutorial solution using rigid body dynamics model with `vofDyMSolver` multiphase solver in Caelus v7.04

Virtual Blade Model

- Implemented Virtual Blade Model per *"Development of Virtual Blade Model for Modelling Helicopter Rotor Downwash in OpenFOAM"* Stefano Wahono, Defence Science and Technology Organisation, DSTO-TR-2931, 2013. [OpenFOAM Foundation]

Turbulence

- Add incompressible and compressible variants of the standard **`kEpsilon`** model. [OpenFOAM Foundation]

Development

Build System

- Build system overhauled to use site_scons directory to centralize custom builders and tools. External contribution from Shreyas Ananthan.