NUMAP-FOAM 2010

An Overview

Oliver Borm

2010-11-19

Outline

Aim of the OpenFOAM SummerSchool

Getting together PhD students and young researchers who spend two weeks working under supervision on their own OpenFoam project

Activities of the OpenFOAM Summer School

- Project work
- Exchange of experiences with colleagues
- Group lectures on the subjects:
  - Numerical modelling
  - Computational Fluid Dynamics (CFD)
  - Object-Oriented Programming and C++
  - Aspects of physical modelling

Activities of the OpenFOAM Summer School

- CFD: **C**roatian **F**ood & **D**rinks
- Having a nice dinner/evening together
- Weekend trips (e.g. clubbing & culture)

General Information

- When does the Summer School take place?
  $\rightarrow$ Once a year at the beginning of september

General Information

- When does the Summer School take place?
  → Once a year at the beginning of september
- Where does the Summer School take place?
  → At the University of Zagreb, Croatia

General Information

- When does the Summer School take place?
  → Once a year at the beginning of september
- Where does the Summer School take place?
  → At the University of Zagreb, Croatia
- How to apply?
  → Application with a description of the project for the Summer School and current problems and goals

General Information

- When does the Summer School take place?
  $\rightarrow$ Once a year at the beginning of september
- Where does the Summer School take place?
  $\rightarrow$ At the University of Zagreb, Croatia
- How to apply?
  $\rightarrow$ Application with a description of the project for the Summer School and current problems and goals
- Who can apply?
  $\rightarrow$ All students on MSc and PhD university courses
  $\rightarrow$ Young researchers in commercial companies with OpenFOAM experiences

General Information

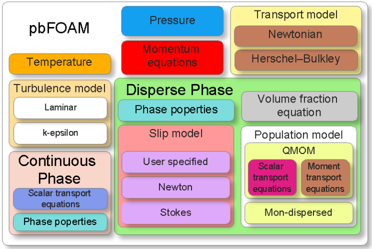
- When does the Summer School take place?
  → Once a year at the beginning of september
- Where does the Summer School take place?
  → At the University of Zagreb, Croatia
- How to apply?
  → Application with a description of the project for the Summer School and current problems and goals
- Who can apply?
  → All students on MSc and PhD university courses
  → Young researchers in commercial companies with OpenFOAM experiences

Note: The Summer School is not an introductory OpenFOAM course

Develop a poly-disperse multi-phase solver

Darrin Stevens

- Based upon a mixture model formulation
- Handles the poly-disperse nature of the phases using the Quadrature Method of Moments (QMOM).
- Handles reactions between scalars transported with each phase
- Both segregated and block coupled solvers were implemented

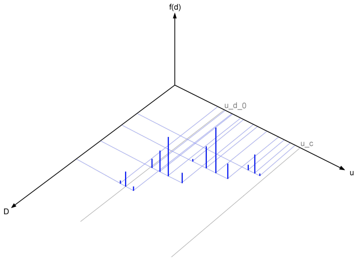


pbFoam flow chart

Implementation of the Direct Quadrature Method of Moments (DQMOM)

Patrick Dems

- Eulerian method for polydispersed two-phase flows
- Transporting optimally chosen size classes
- Here: Consideration of the drag force only

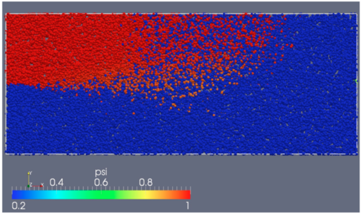


DQMOM

Implementation of hybrid Finite Volume / Monte Carlo particle in OpenFOAM

Heng Xiao

- Based on joint PDF
- To solve turbulence reactive flows with multiple species
- Hydrodynamic with FVM (simpleFoam) and the species and concentrations with particle methods (using the particle capabilities in OpenFOAM)

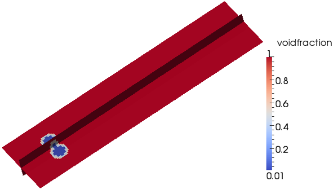


Mixing process of the particle concentration.

Coupling of CFD and discrete element method

Alice Hager

- insertion of big particles via voidfraction
- correction of a porous media solver (pressure-velocity-decoupling)

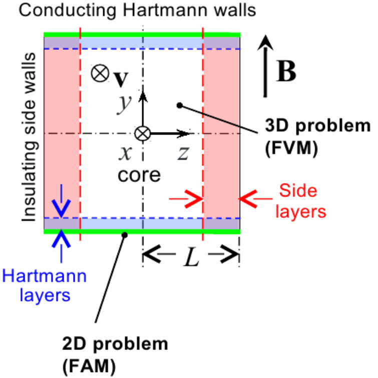


Void fraction

MHD = study of interaction of moving electrically conducting fluids with applied magnetic fields

Chiara Mistrangelo
Interaction of a moving liquid metal with an imposed magnetic field **B**.
Implementation of thin wall condition for simulating MHD flows in electrically conducting channels:

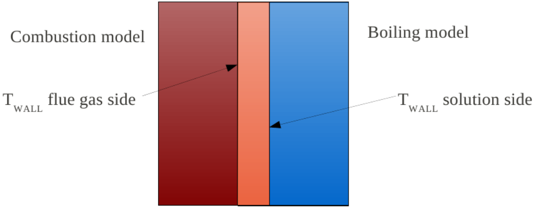
- Current entering the wall distributes only in tangential direction along the wall
- FAM problem (2D current sheet in wall) coupled with FVM problem (3D fluid flow)



Coupled numerical procedure for solving fluid dynamic fields of a direct fired generator

Peter Benovsky

- Find an approach for the simulation of the conjugate heat transfer in a direct fired heat generator of an absorption heat pump.
- The wall temperature resolution of the generator was of interest.
- conjugateHeatFoam was taken as a basis for the development of the new solver.



Premixed and partially premixed combustion in engines with LES

Roman Keppeler

- (partial-) premixed combustion for internal combustion engines (ICE) using LES for turbulence modelling
- Implementing an implicit, local LES-model (ALDM), that takes discretization effects into account, via utilizing a WENO scheme

Modeling Large-Scale Fire

Yi Wang

- Buoyancy-driven turbulent diffusion flame
- Enhancement of stability and efficiency
- Topological change to account for burning though of solid surfaces

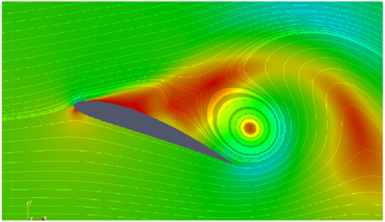
Bill Rosemurgy

- mesh handling using fvMeshSubset and other tools
- boundary patch triangularization
- efficient programming in OpenFOAM (from 12+ hours/run to $<$ 30 min/run)

Fluid-structure interaction and advanced aero-elastic models

Rasoul Shirzadeh

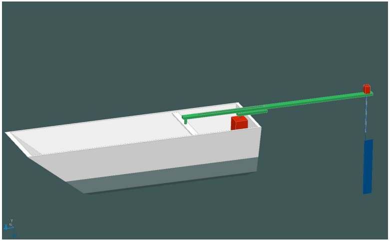
- Improve aeroelastic models by numerical and experimental approaches
- Worked on two basic models (cylinder and airfoil)
- Get correct Drag & Lift coefficients for a wide range of Reynolds numbers.
- Fluid forces have a great effect on the response of aeroelastic models.



Robotic ships and flapping foils: moving mesh free surface flow

Joris Mattheijssens

- Unmanned ships in minehunting operations with low propeller noise.
- A biomimetic fin will be developped to propel small ships.
- In order to do simulations, a dynamic mesh application was written, using both sliding interfaces and deforming cells.

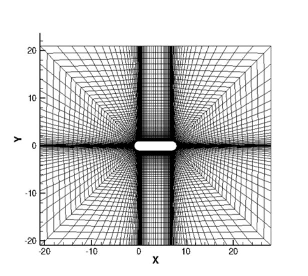


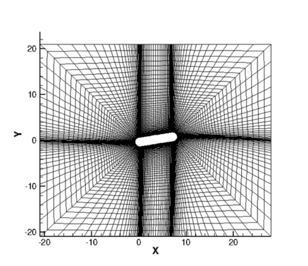
Ship and Fin

Body motion in super-cavitating flow

Sunho Park

- Mesh Motion (GGI and forced motion)
- SRF
- 6 DOF

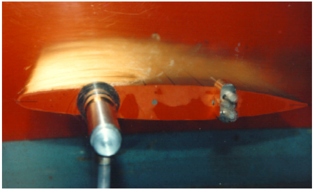




Model the transition from sheet to cloud cavitation

Aurelia Vallier

- include a switching from volume of fluid method to lagrangian particle tracking method
- account for all the forces applied to the bubbles
- solve Rayleigh-Plesset equation for the dynamics of the bubbles



Inception / Steady attached cavity

Cavity break-off /Cloud cavitation

Turbomachinery and compressible transonic flows

Oliver Borm

- Implementation of a Godunov-like solver
- Approximate Riemann Solver:
  - HLLC formulation from Batten et. al [BLG97]
  - Roe & Pike scheme with Harten's entropy fix
  - HLLC ALE formulation from Luo et. al [LBL04]
- 2nd Order space accuracy
- Local and Dual Time Stepping
- Multi-Stage Runge-Kutta
- Adapted total boundary conditions for internal flows
- Extension of MRF and SRF models

TODO:

- Extension to implicit time integration
- Testing of different Limiters and Schemes

Upwind splitting scheme - Riemann solvers

- Input - primitive variables
- Output - conservative fluxes (computed internally from primitive variables)
- Boundary Condition formulated for primitive state vector $\rightarrow$ Riemann solver is fed with this state vector to compute conservative fluxes at boundary faces
- GGI is working, as Riemann solver uses primitive variables as input
- Turbulence Modelling: Added as diffusive fluxes

2nd order space accuracy - Slope Limiter

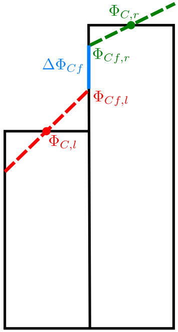
- Linear reconstruction of any input variable at faces as first term in Taylor series expansion from the cell centered value of this variable:

$$\Phi(x) = \Phi(a) + \frac{d\Phi(a)}{dx} \bullet [x - a] \qquad (1)$$

- Procedure is repeated for the left and right state vector of each face
- For stability and monotinicity reasons, the gradient has to be limited with a Limiter Ψ (Minmod ATM) in the following way:

$$\Phi(Cf) = \Phi(C) + \Psi\{\nabla\Phi(C) \bullet [Cf - C]\} \quad (2)$$

- Limiter is identical for each primitive input variable at both sides of a face



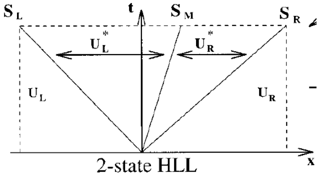
Slope Limiting

Upwind flux splitting - HLLC

HLLC

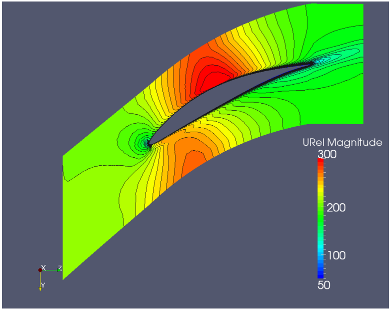
- Determine Signal speeds (left, right, contact wave) for Euler Equation (smallest and largest eigenvalues): $S_L = \min\left[\lambda_1\left(\vec{W}_l\right), \lambda_1\left(\vec{W}^{Roe}\right)\right]$ $S_R = \max\left[\lambda_m\left(\vec{W}_R\right), \lambda_m\left(\vec{W}^{Roe}\right)\right]$
- Numerical Fluxes at faces (for implicit TS more terms arising):

$$F_{HLLC} = \begin{cases} F\left(\vec{W}_l\right) & \text{if } S_L > 0 \\ F\left(\vec{W}_l^*\right) & \text{if } S_L \leq 0 < S^* \\ F\left(\vec{W}_r^*\right) & \text{if } S^* \leq 0 \leq S_R \\ F\left(\vec{W}_r\right) & \text{if } S_R < 0 \end{cases}$$

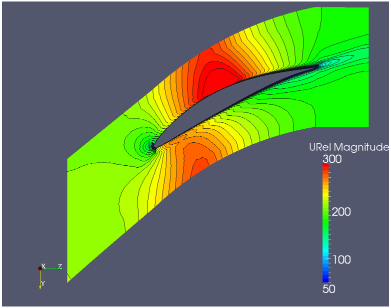


Case 2

Testcase - Subsonic compressor rotor

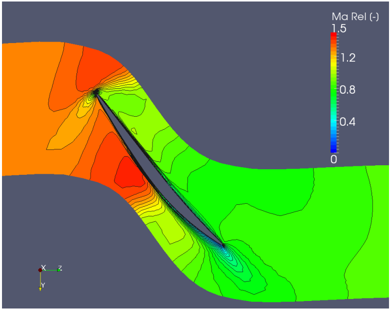


transonicMRFDyMFoam

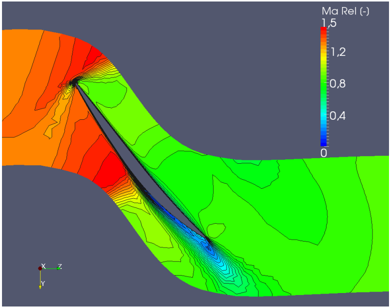


transonicSteadySRFFoam

Testcase - NASA Rotor 37



transonicSteadySRFFoam



numeca

Literature

P. Batten, M. A. Leschziner, and U. C. Goldberg.
Average-State Jacobians and Implicit Methods for Compressible Viscous and Turbulent Flows.
Journal of Computational Physics, 137:38–78, 1997.

Hong Luo, Joseph D. Baum, and Rainald Löhner.
On the computation of multi-material flows using ALE formulation.
Journal of Computational Physics, 194:304–328, 2004.