Developments in Transient Modeling, Moving Mesh, Turbulence and Multiphase Methodologies in OpenFOAM

Prof. **Federico Piscaglia**
POLITECNICO DI MILANO (Italy)
federico.piscaglia@polimi.it

Acknowledgments: Dr. *Andrea Montorfano* (Phoenix SpA)
Dr. *Jerome Helíe* (Continental Automotive SAS, France)
Dr. *Shashikant M. Aithal* (Argonne National Laboratory, US)

Research Activity: CFD

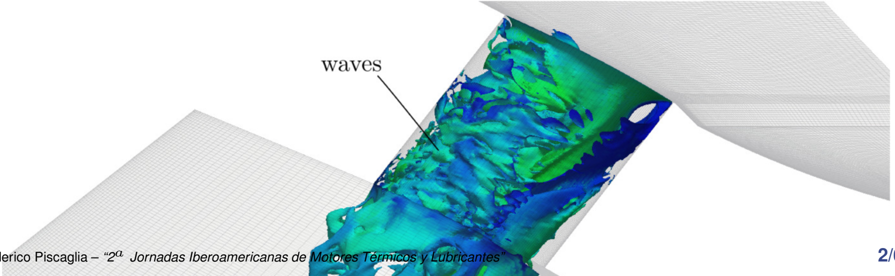
TURBULENT (REACTING) FLOWS IN MOVING BOUNDARY PROBLEMS

REQUIREMENTS:

1) **mesh motion** based on **automatic topological changes**
2) algorithms for **decomposition** with topoChanges
3) Fast and accurate **transient dynamic solvers**

+ WIDE RANGE OF PHYSICAL SUBMODELS

(turbulence, combustion, chemistry, heat transfer, wall film, multiphase, ...)



Objectives

BURNING QUESTIONS:

- how to limit the time spent for meshing/pre-processing, preserving the mesh quality
- how to speed up transient dynamic simulations?
- what is the **minimum mesh resolution** for time-resolved turbulence in wall-bounded flows?

↓

CODE DEVELOPMENT DONE AT DIFFERENT LEVELS:

- **meshing**
- **numerical solvers**
- **code parallelization**
- **turbulence modeling**
- **multiphase flows**: VOF single fluid, Eulerian two-fluids, Lagrangian methods
- **sub-models for specific physical problems** (reactive flows, cavitation, ...)

Why OpenFOAM®?

- **Ready-to-use CFD code**
  - applications and state-of-the-art models for CFD already available in the code
  - polyhedral mesh support
  - include mesh generator, conversion from different mesh formats
- **Open-source, object-oriented C++ at no license costs**:
  - code customization
  - maximum code re-use
  - minimum code maintainance
  - research in a collaborative environment

Teaching OpenFOAM@PoliMi

Prof. Federico Piscaglia

Computational Techniques for Thermochemical Propulsion

Master of Science in Aerospace Engineering, Politecnico di Milano

(*Cod. 051176*, *8 credits*. Duration: 3 months)

COURSE CONTENT:

1. **theory** (6 hours/week) about **CFD modeling of turbulent compressible reacting flows**.
2. **exercises/laboratory** (4 hours/week), about the **advanced use of OpenFOAM**. A significant part of this class is devoted to learn how to use the code, to understand its complex object-oriented structure and, finally, to extend its capabilities.

Fields of application

Development (and maintainance!) of CFD algorithms and methodologies for fast, scalable and reliable solutions both in a **RESEARCH** and in an **INDUSTRIAL CONTEXT**.

Continuous development of CFD algorithms and methodologies in the OpenFOAM Technology to provide a parallel, stable and validated code that can be applied by industry to the solution of **general CFD problems**, as an alternative to the most established commercial CFD codes.

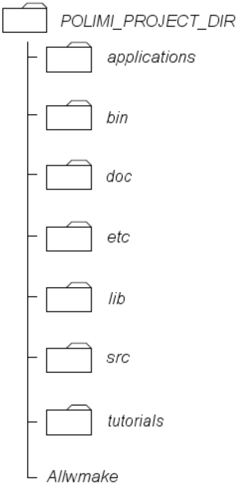
FIELDS OF APPLICATION are:

- multiphase flows
- meshing
- non-linear acoustics

→ VOF injection, lagrangian sprays, wall film (TODAY's TOPIC)

- reactive flows, IC engines
- aerospace
- heat transfer
- external aerodynamics
- pollutant dispersion

Code development



Code development is organized as a set of dynamic libraries in a C++ object-oriented library that replicates the original code structure of OpenFOAM:

- *bug fixes* for the base classes are cross-compiled with the original software libraries → maintenance;
  - ALL the original applications/solvers/utilities of the software are implicitly enabled and see the extensions
  - source files of the OpenFOAM® distribution are not changed
- physical sub-models are linked dynamically at run-time, accordingly to the standard procedure
- *DAILY* maintenance to preserve compatibility with the latest official release provided by the OpenFOAM Foundation

Architects and core developers of the code are F. Piscaglia and A. Montorfano. Part of the theory and validation of the code is available in their published literature.

Code development

Programming:

- GIT version control
- Library structure documented by DOXYGEN
- Module files for compilation on HPC
- Regression testing to ensure code integrity

Computational (HPC) facilities:

- Argonne National Lab
- sponsoring Companies, PRACE, LISA

Parallel Performance: BEBOP Cluster@ANL

352 nodes of **Knights Landing (KNL)**, Intel(R) **Xeon Phi**$^{(TM)}$ CPU 7230 @ 1.30Ghz, 128 GB/node

⇩

ABOUT 25 K CORES IN TOTAL

MAIN FEATURES

- Significant improvement in scalar and vector performance
- 512-bit vector units per core (= 256-bits earlier chips)
- Vector Peak Perf: 3+ TF DP (DP=double precision; TF = TeraFlops)
- Four-way multithreading (64x4)
- Lower power consumption

Parallel Performance: BEBOP Cluster@ANL

352 nodes of **Knights Landing (KNL)**, Intel(R) **Xeon Phi**$^{(TM)}$ CPU 7230 @ 1.30Ghz, 128 GB/node

⇩

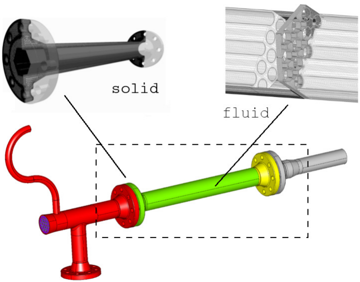
ABOUT 25 K CORES IN TOTAL

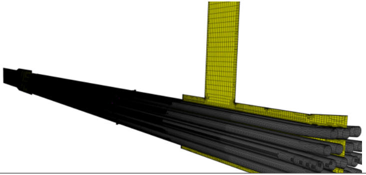
REMOTE VISUALIZATION/POST-PROCESSING:

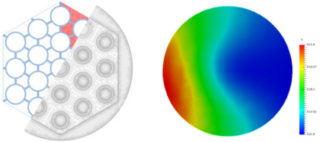
- remote data processing on HPC
- remote rendering on GPU (NVIDIA Quadro K5000)
- remote visualization by VGL Image Transport

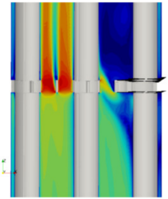
Conjugate Heat Transfer

CHT Simulation of IV-Gen Nuclear Reactors in collaboration with ANSALDO Nucleare, ENEA FSN-ING, CRS4 (Italy), NRG (Netherlands).

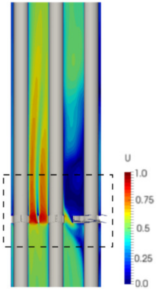






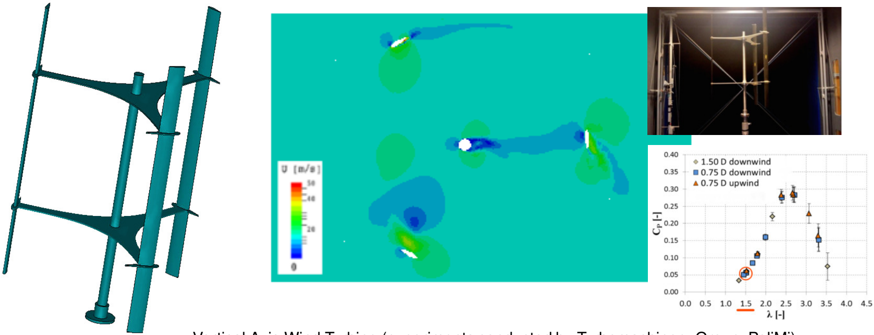


- fluid: LBE
- submodels for Pr_t



"CFD Analyses of the Internal Blockage in the NACIE-up fuel pin bundle simulator". **The 17th Int. Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-17), Xi'an, China, 2017.**

Wind Energy



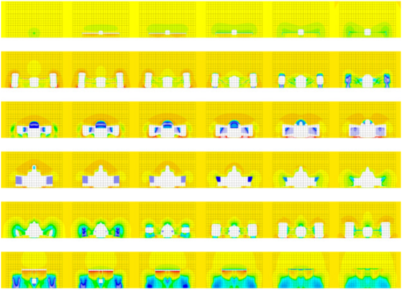
Vertical Axis Wind Turbine (experiments conducted by Turbomachinery Group, PoliMi)

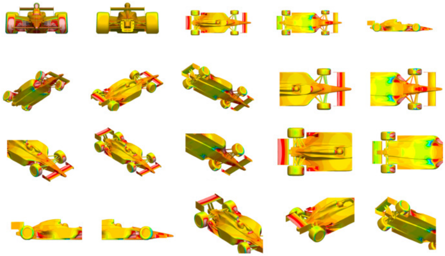
Example of application: H-type Darrieus turbine

- Full-scale simulation with dynamic mesh (AMI)
- Comparison with experiments (global coeffs)
- Transient simulation of turbine start up

A. Montorfano, F. Piscaglia et al. *"Application of a Dynamic Model with length scale-dependent RANS/LES hybrid functioning to a Wind Turbine Simulation"*. **Third Symposium on OpenFOAM® in Wind Energy, 2016**

External Aerodynamics





FULLY-AUTOMATIC WORKFLOW:

case setup and mesh generation

↓

computation

↓

post-processing

↓

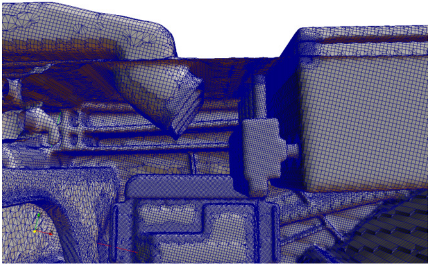
report

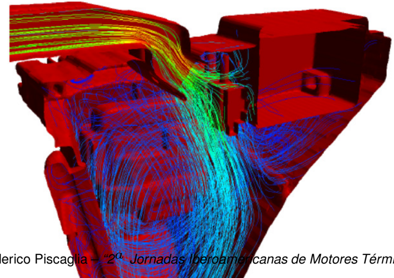
```
//- -----------------------------------------------------
   Using GLX wrapper... /bin/vglrun
//- -----------------------------------------------------
   plotResiduals:       true;
   averageCoeffs:       false;
   saveCuttingPlanes:   yes;
   saveWallFields:      no;
   writeReport:         yes;
   genCuttingPlanes:    yes;
   calcFunctionObjects: on;
   calcWallFields:      yes;
   writeReport:         yes;
     [I]  coeffTimeFolder =
     [I]  startXCoord =
     [I]  endXCoord =
     [I]  deltaXCoord = False
     [I]  carBodyFields = cp wallShearStress yPlus
     [I]  cuttingPlanesFields = helicity cp cpTot U
     [I]  nAverage =
//- -----------------------------------------------------
```

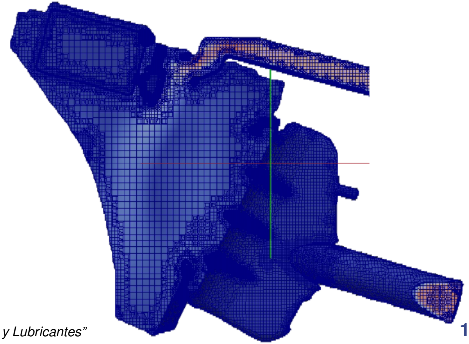
Intake System Optimization

`snappyHexMesh`

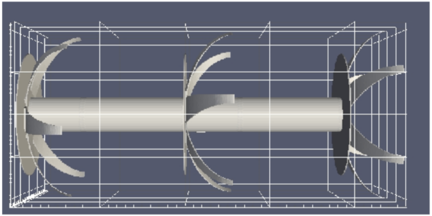
- The FVM method allows polyhedral support: fewer cells per volume, minimal distortion,near-wall layers
- Avoiding user-interaction: **reliable automatic meshing**

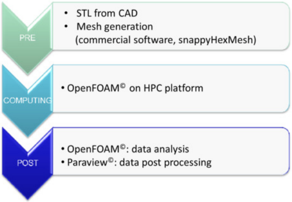




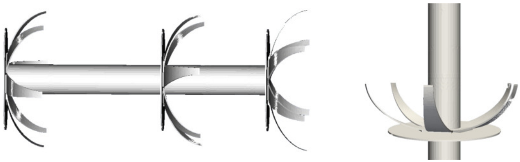


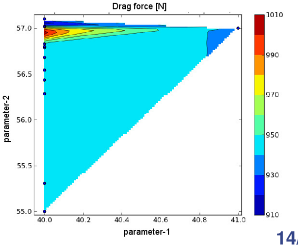
Shape optimization (OpenFOAM+Dakota)



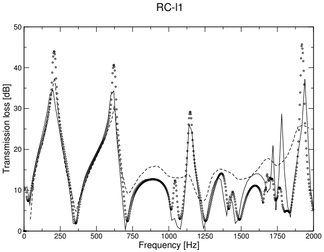


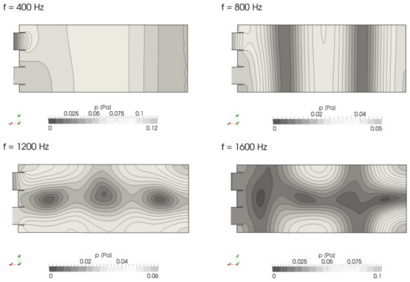
- Optimization loop to find the geometric configuration maximizing the drag of the device, used to monitor pipe systems with high pressure Methane
- **Parametric optimization is performed by coupling OpenFOAM® with DAKOTA**, an open-source Multilevel Parallel Object-Oriented Framework for design optimization





Non-Linear Acoustics





- **Implementation of a true non-reflecting outlet based on the NSCBC theory**: variables are computed on the boundaries by solving the conservation equations as in the inner domain
- absence of reflection is enforced by correcting the amplitude of the ingoing characteristic (wave reflected by the boundary)

F. Piscaglia, A. Montorfano, A. Onorati. "Development of a non-reflecting boundary condition for multi-dimensional non-linear duct acoustic computation", **Journal of Sound and Vibration**, Volume 332, Issue 4, Pages 922-935, ISSN 0022-460X, 10.1016/j.jsv.2012.09.030.

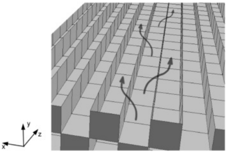
Exhaust after-treatment modeling

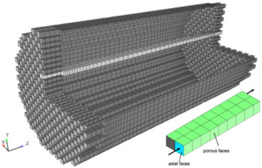
CENTRO RICERCHE FIAT

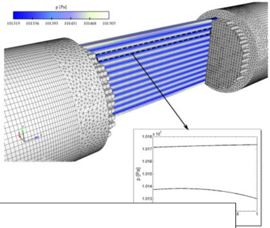
POLITECNICO MILANO

`dpfFoam` solver for compressible flows through porous media:

- Explicit staggered and parallel porous solver with friction model
- New internal face condition (`pressureJump`) to model:
  - steady-state propagation of a sudden finite change in flow properties
  - thin membranes with known velocity/pressure-drop characteristics, by the implementation of the Darcy law
- Implementation of the transport equations for soot, filtration and deposition model
- automatic mesh generation and case setup
- **Validation** against experiments[1,2]



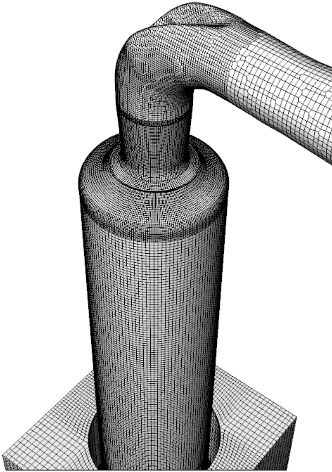




[1] F. Piscaglia, A. Montorfano et al. **SAE Technical Paper 2009-01-1965**
[2] F. Piscaglia, A. Montorfano et al. **SAE Technical Paper 2009-24-0137**

In-cylinder Flows

1) In-house Hexa-block mesh generator
2) Support for ANY mesh motion strategy in a single run:
   - automatic with topological changes
   - cell deformation/stretching + mesh-to-mesh interp.
3) Point motion solver
4) In-cylinder flow simulation:
   - piston crevice and blow-by modeling
   - modifications to piston motion, conrod deformation
   - real gas effects
   - dynamic specie transport (DST) with reactive flows for f
5) support for Lagrangian particle tracking
6) Finite-Volume Wall-Film treatment
7) Turbulence modeling: RANS, LES, PANS, scale-adaptive
8) Post-processing

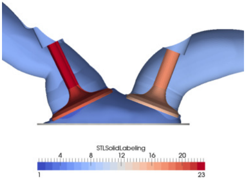


→ Full Engine-cycle simulation in 1 day

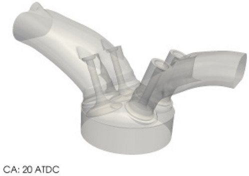
Mesh-to-mesh interpolation

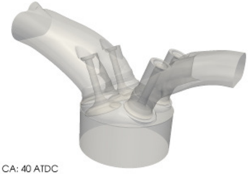
Starting from the desired CAD geometry in Stereolithography (STL) format, a discrete number of triangulated surface geometries are generated in order to cover the entire full cycle simulation:

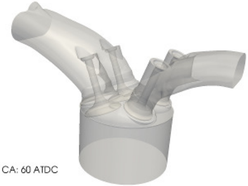
- **Step 1**: the template STL file is generated by separating different patches, to allow for the subsequent geometry modification, mesh local refinements and boundary condition assignments;



- **Step 2**: desired CA geometry is generated moving valves and piston patches of the STL file according to data (utility `surfaceEngineCreate`)







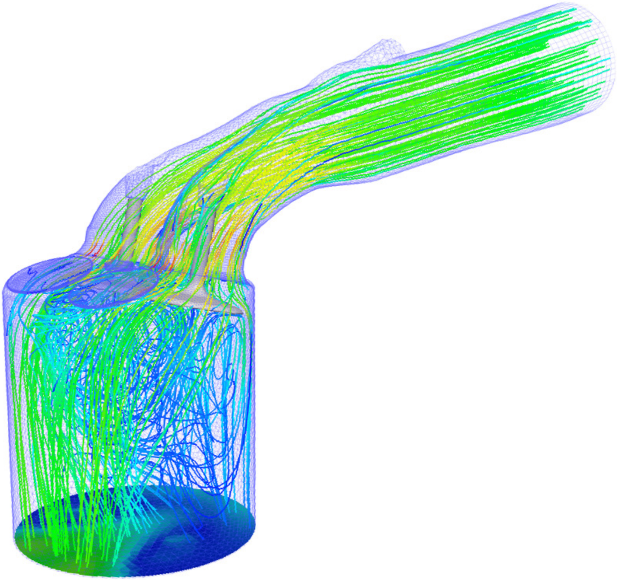
- **Step 3**: the mesh is generated by `snappyHexMesh`.

SAE 2013-24-0027

Mesh-to-mesh interpolation

Starting frc ... te number of
triangulate ... e simulation:

- **Step** ...
  rating
  geom
  bounc

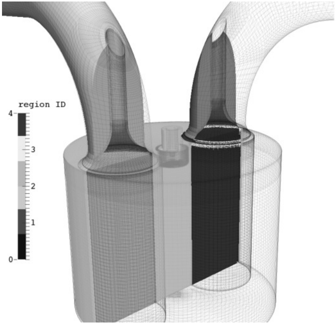


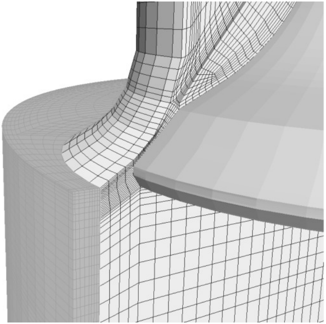
- **Step 2**: ... f the STL file
  accordi

CA: 20 ATDC

- **Step 3**: ... -24-0027

Four-stroke Engines - SAE 2015-01-0384

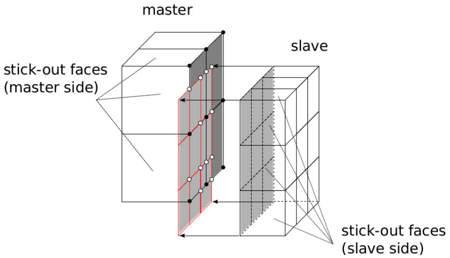


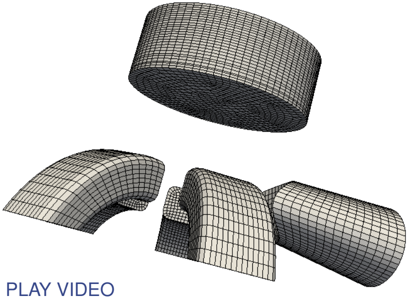


- Combination of different methods to handle non-conformal interfaces (IMEM 2015) (significant increase of speed of the mesh motion):
  - valve opening/closure → target-mesh approach (`attachDetach`)
  - sliding valve → supermesh approach (`AMI`)
- Integration of the several methodologies for mesh motion together with `layerAdditionRemoval` in one single fluid-dynamic solver

[1] A. Montorfano, F. Piscaglia et al. **SAE Technical Paper 2015-01-0384.**
[2] F. Piscaglia, A. Montorfano et al. **Int. Multidim. Engine Modeling Meeting 2015.** Downloadable at `https://imem.cray.com/`

Two-stroke Engines



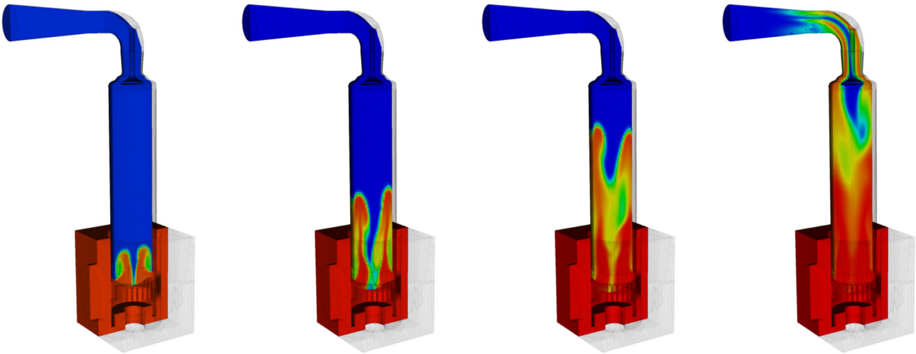


▷ PLAY VIDEO

1) **Different mesh motion strategies** in a single run:
   - automatic with topological changes
   - cell deformation/stretching + mesh-to-mesh interp.
2) **support any** `fvMotionSolvers`;
3) **2nd-order accuracy in time with topoChanges**;
4) **full integration** of the dynamic mesh handling with ANY existing solver in OpenFOAM®;
5) **fully-automatic workflow**

Two-stroke Engines

Gas Exchange/Fuel-Air Mixing

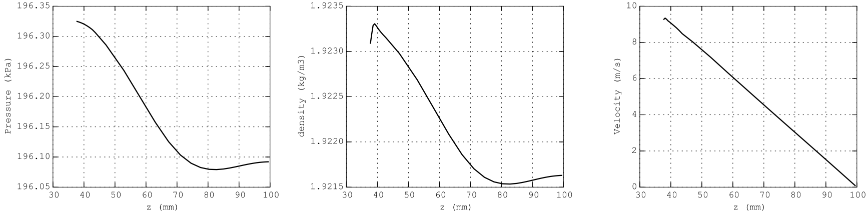


t = 153 CA-deg   t = 163 CA-deg   t = 173 CA-deg   t = 183 CA-deg

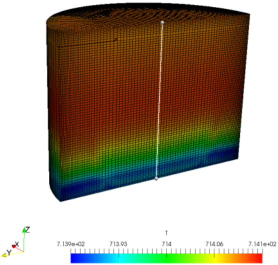
E. Baudoin, J. D. Kunoy, F. Piscaglia, A. Montorfano. *"In-cylinder flow simulations in large marine two-stroke engines".* 5th OpenFOAM User Conference, Frankfurt (Germany), 2017.

Dynamic solvers: conservativeness

<u>`Compressible flow case`</u>

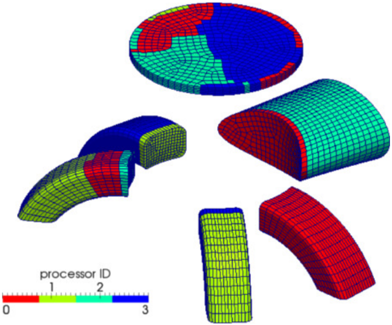


- solver type: PISO (convergence not forced by `outerCorrectors`
- `layerAdditionRemoval`
- GCL obeyed
- no discontinuity in the conserved variables (`U`, `p` and `T`) over the added/removed layers is observed
- improved energy conservation, scalar transport and continuity across topological changes with *2nd-order discretization in time*



Decomposition Method with TopoChanges

With parallelised topological changes, reference between initial global mesh and processor mesh is lost during simulation and cannot be implied.



SOLUTION:

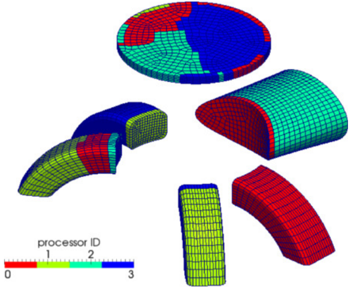
- with topological changes, constrained decomposition must be performed;
- global mesh is built from scratch, adding cells in order of processor index and assemble mapping data;
- fields on reconstructed mesh can be assembled or decomposed as before

Related publications: **SAE 2013-01-0024**, **IMEM 2013**, **SAE 2015-01-0384**

Automatic Constrained Decomposition

MAIN FEATURES:

- scotch/metis algorithm used as base algorithms for decomposition;
- implemented automatic algorithm interacts with the base decomposition method, automatically;
- cell decomposition over multiple processors is based on cell/face addressing, to minimize processor communications and to ensure optimal processor load balancing.



ADVANTAGES:

- Fully automated: minimum effort required to the user with topological changes

LIMITATIONS:

- Optimal solution when the calculation load is uniformly distributed

When complex sub-models (chemistry, cavitation,...) are involved together with topological changes, possible options are:

- automatic algorithm with specification of the load distribution
- **semi-automatic methods** $\rightarrow$ ICMF2016

Recognitions/Awards

- PRACE DIGEST, Feb 2013: development on **LES of IC Engines in OpenFOAM** on **large problems with moving boundaries** highlighted as reference for very fast and scalable implementations;

- The Connector (POINTWISE), July 2014: **dynamic mesh handling** highlighted as reference methodology for the simulation of IC engines:

  http://www.pointwise.com/theconnector/July-2014/Unsteady-Engine-Analysis.shtml

- **keynote talk** at the **4th annual OpenFOAM User Conference 2016** by *ESI-OpenCFD*

  https://www.esi-group.com/company/events/2016/4th-annual-openfoam-user-conference-2016

- **keynote talk** at the *VERIFI Workshop*, Argonne National Lab, 2016

Fields of application

Development (and maintainance!) of CFD algorithms and methodologies for fast, scalable and reliable solutions both in a **RESEARCH** and in an **INDUSTRIAL CONTEXT**.

Continuous development of CFD algorithms and methodologies in the OpenFOAM Technology to provide a parallel, stable and validated code that can be applied by industry to the solution of **general CFD problems**, as an alternative to the most established commercial CFD codes.

FIELDS OF APPLICATION are:

- multiphase flows
- meshing
- non-linear acoustics

→ VOF injection, lagrangian sprays, wall film (TODAY's TOPIC)

- reactive flows, IC engines
- aerospace
- heat transfer
- external aerodynamics
- pollutant dispersion

Internal Nozzle Flow

Acknowledgments

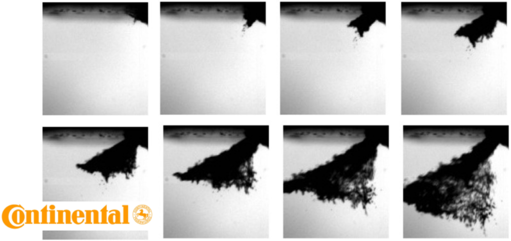
This presentation includes, in part, the collaborative technology developments for **Continental Automotive SAS**. Authors thank Continental Automotive SAS for permission to show part of the results in the presentation.

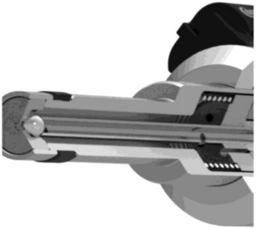
Computer facilities were kindly provided by the Laboratory Computing Resource Center (LCRC) at the Argonne National Lab within the `PETSC-Foam` and `KNL-VOF_OpenFOAM` projects.

VOF - Motivation

WHY

More efficient combustion and reduction of the emission limits $\rightarrow$ GDI Engines





HOW

Improved design of GDI injectors. Main focus on:

- **INJECTOR GEOMETRY**: for spray characteristics (and fuel/air mixing!);
- **INTERNAL NOZZLE FLOW PHYSICS**: for primary breakup of the liquid jet.

Aim of the work

Parametric tests to characterize the injector operation based on:

- injection pressure: 30 to 200 bar
- nozzle geometry
- needle opening strategy
- fuel composition

↓

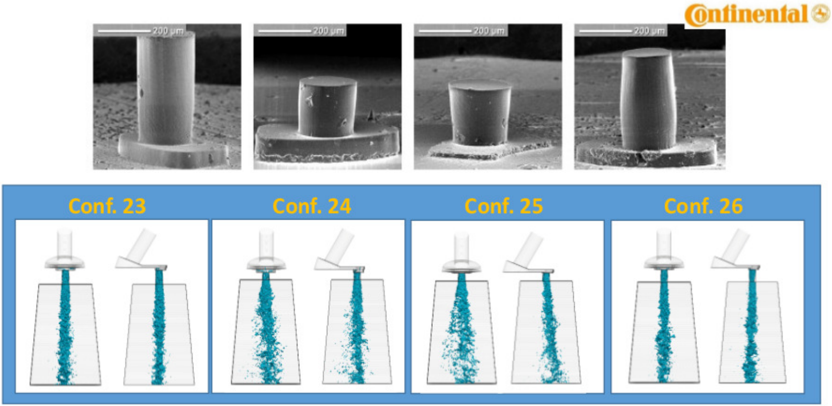
Test operated at two different conditions:

- STATIC: full needle opening
- TRANSIENT: needle opening and closure

Static needle operation

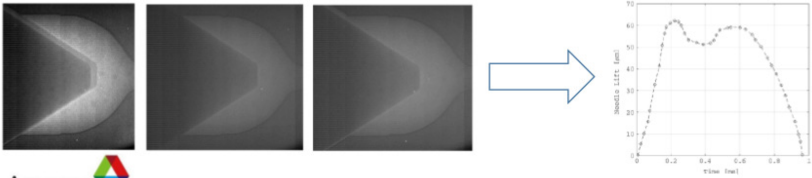
Simulations of different nozzle configurations under **STATIC** conditions:

- average grid size: ≈ 20 Million cells/simulation
- simulation time: up to 2 days, 512 cores (KNL)



Transient conditions (opening/closure)

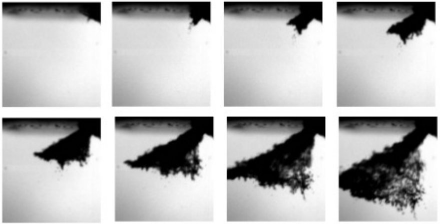
HIGH-SPEED X-RAY IMAGING OF INJECTOR NEEDLE MOTION (TRANSIENT)

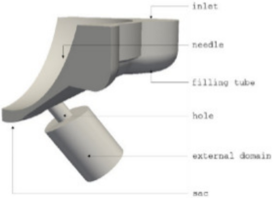


Argonne NATIONAL LABORATORY

Over 100 geometries to test!

HIGH-SPEED CAMERA VISUALIZATION OF PRIMARY ATOMIZATION IN SPRAYS



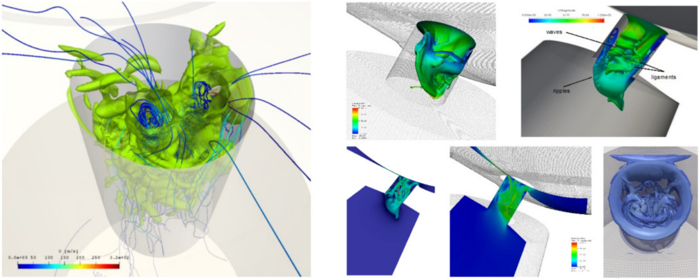


Continental

What is needed?

A **CFD tool** to substitute lack of experimental measurements on primary breakup during transients.

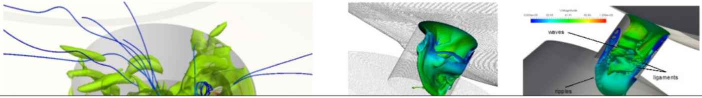
1) **automatic mesh generation** of multi-block, body-fitted, oriented grids (`snappyHexMesh` is NOT an option for such problems!);
2) **multiphase (dynamic) VOF solvers supporting phase change/cavitation**;
3) advanced **turbulence modeling** (LES/hybrid);
4) **moving mesh capabilities** to handle parametric geometries.



What is needed?

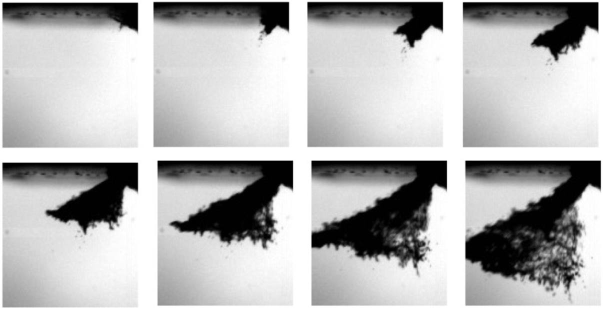
A **CFD tool** to substitute lack of experimental measurements on primary breakup during transients.

1) **automatic mesh generation** of multi-block, body-fitted, oriented grids (`snappyHexMesh` is NOT an option for such problems!);
2) **multiphase (dynamic) VOF solvers supporting phase change/cavitation**;
3) advanced **turbulence modeling** (LES/hybrid);
4) **moving mesh capabilities** to handle parametric geometries.



... and for **HIGH-FIDELITY (LES) simulations**: linear code scalability on large clusters, automatic offline efficient pre/post processing and automatic report generation, specific decomposition methods to improve load balancing.

Why a 3-phase (with phase-change) solver?



In GDI injectors, phase change involves:

liquid fuel $\longleftrightarrow$ fuel vapor

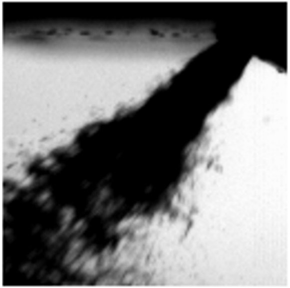
AIR is inert

The gas phase (air + vapor) must be modeled as two separated phases, to track the evolution of the air and the fuel-vapor and to properly account for condensation:

- at the nozzle exit and in regions near swirling cavitation;
- at the needle closure (transient simulations, dynamic mesh)

Do not use a 2-phase (with cavitation) solver!

This is what happens to use a 2-phase solver (`interPhaseChangeFoam`) to model fuel injection with cavitation...



Experiments

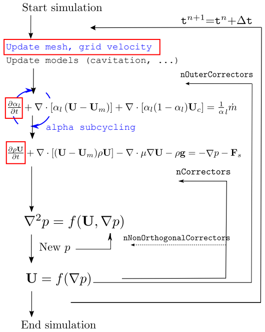
3-phase VOF

`interPhaseChangeFoam`
(without ad-hoc tuning...)

Modeling condensation with a two-phase single-component solver (like `interPhaseChangeFoam`) might potentially favor the conversion of air (gaseous) into liquid fuel in the condensing regions, introducting a significant error in the calculation of the liquid phase fraction.

F. Piscaglia, F. Giussani, A. Montorfano, J. Helie, S.M. Aithal. "A MultiPhase Dynamic-VoF Solver to Model Primary Jet Atomization and Cavitation inside High-Pressure Fuel Injectors using OpenFOAM". **Accepted for publication on Acta Aeronautica (Elsevier), 2018**

Why a 3-phase (with phase-change) solver?



Segregated, single fluid, 3-phase VOF solver

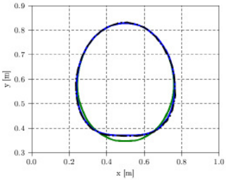
- accounts for 1 immiscible fluid and 2 miscible fluids
- support phase change/cavitation;
- revised formulation of the interface curvature;
- surface tension: enhanced formulation for 3-phases;
- dynamic mesh support w/wo topological changes;
- supports ANY turbulence model (LES, hybrid, RANS)

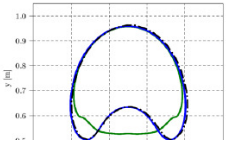
VOF is an interface tracking method. In the transport of the void fraction, used to track each phase in OpenFOAM, a convection-based term compresses the interface and preserves boundedness.

- allow for interface tracking, accurate predictions of the spray morphology
- computational cost still reasonable (single fluid approach)

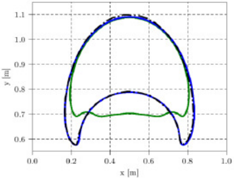
Several VOF solvers are available in OpenFOAM, but **none of them** is suitable to model phase change (i.e. cavitation) **with more than 2 phases** in high pressure fuel injection.

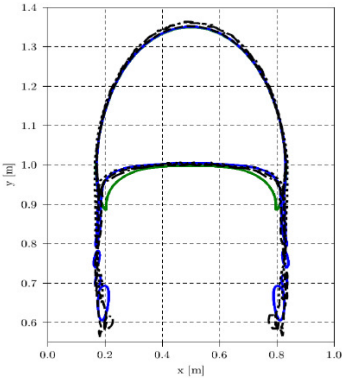
Surface curvature: effect on the void fraction

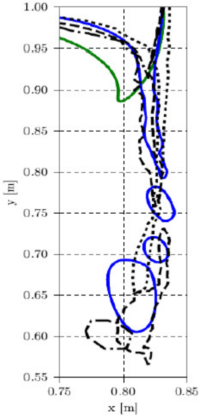




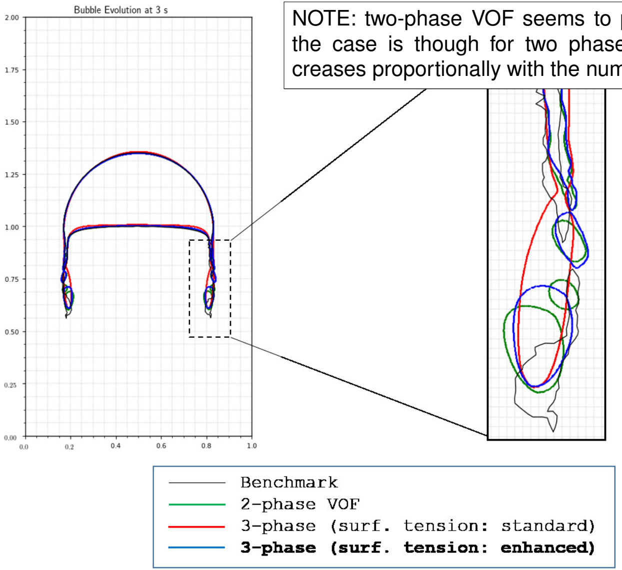
STANDARD vs **NEW**
curvature formulation







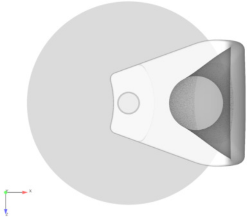
Validation: numerical test case



NOTE: two-phase VOF seems to perform slightly better, but the case is though for two phases; numerical diffusion increases proportionally with the number of phases.

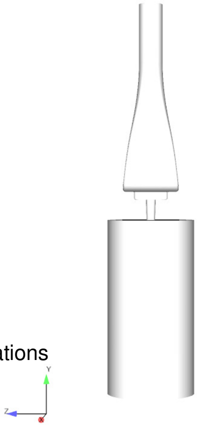
Experiments: Transparent GLASS Nozzles

Ad-hoc built glass TRANSPARENT nozzles for **code validation**.



Experiments (Dr. N. Lamarque):

- 10 ad-hoc built configurations
- laser and high-speed camera visualizations



Continental

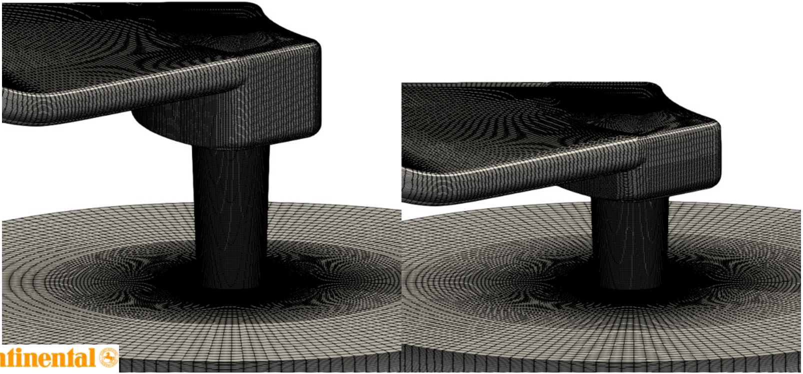
CAD generation:

Sketch with manufacturing tolerances → real injector → X-Ray measurements → CAD for simulations

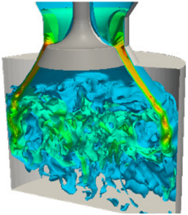
ID-3 and ID-10 Injector configurations

Ambient domain in radial direction (r=0.75 mm)

- 0 - 0.25 mm fine mesh [5e-07 - 2e-06] ≈ 3.8M cells
- 0.25 – 0.5 mm medium refinement [2e-06 - 1e-05] ≈ 2.7M cells
- 0.5 - 75 mm coarse mesh [1e-05 - 2e-05] ≈ 740K cells

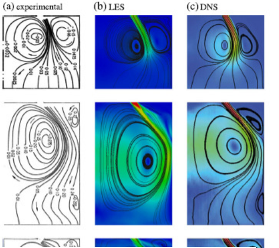


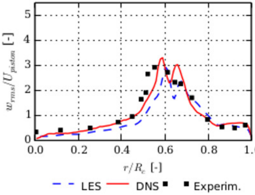
LARGE EDDY SIMULATION OF IN-CYLINDER FLOWS

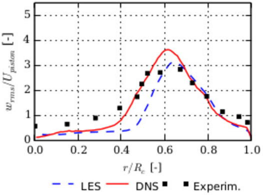


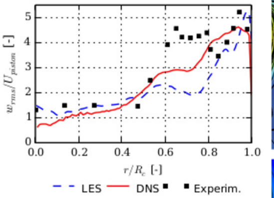
Comparison of streamlines at :

- 36° CA
- 90° CA
- 144° CA
- 270° CA



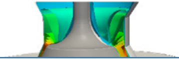






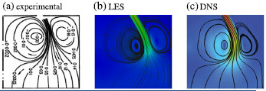
A. Montorfano, F. Piscaglia et al. **Flow Turbulence and Combust. (2015) 95:461-480**

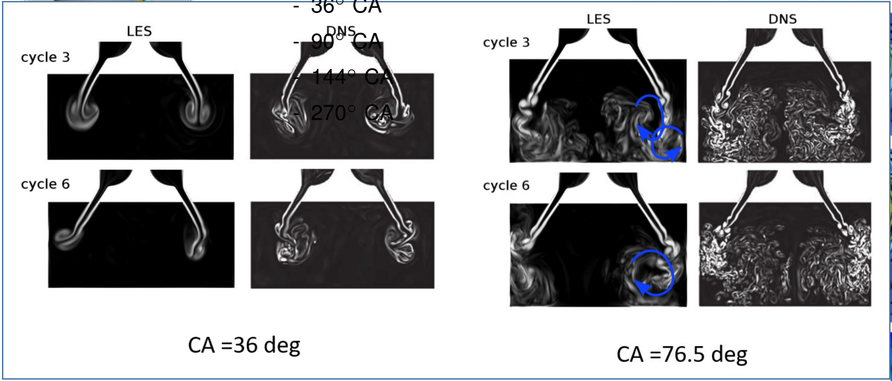
LARGE EDDY SIMULATION OF IN-CYLINDER FLOWS



Comparison of streamlines at :

- 36° CA
- 90° CA
- 144° CA
- 270° CA

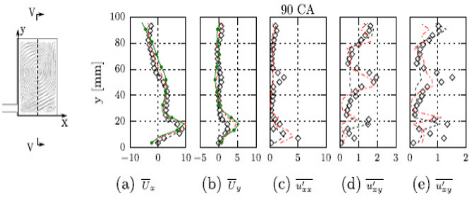


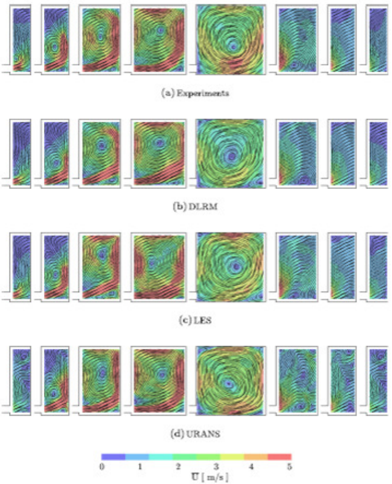


A. Montorfano, F. Piscaglia et al. **Flow Turbulence and Combust. (2015), 95:461-480**

Wall/Jet Interaction - Hybrid RANS/LES

HYBRID RANS/LES SIMULATION of IN-CYLINDER FLOWS



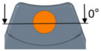


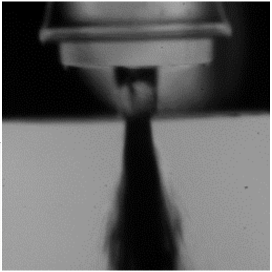
EXPERIMENTS provided by **Prof. J. Boreè**
LABORATOIRES DE RECHERCHE, Institut PPRIME

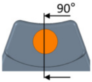
Y. Wu, A. Montorfano, F. Piscaglia et al. **Flow Turbulence and Combust. (2018), 100(3):797–827**

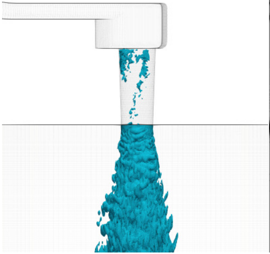
3-Phase Solver - Conf. ID 3

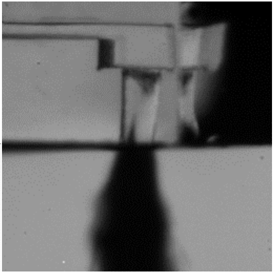
Experimental image deformed by glass





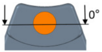


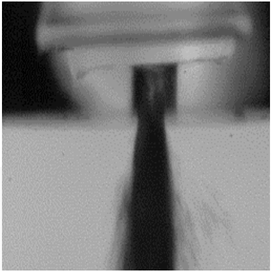


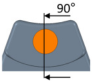


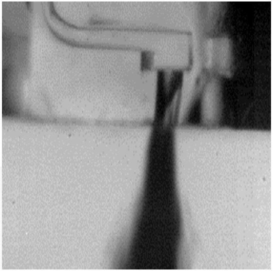
3-Phase Solver - Conf. ID 10

Experimental image deformed by glass

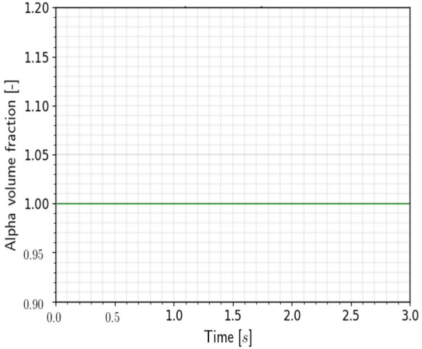








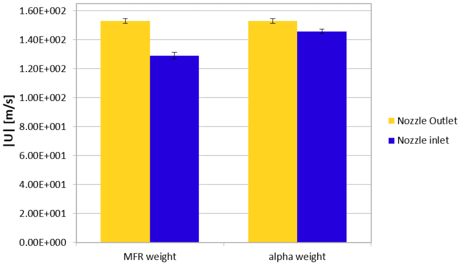
Boundedness and conservativeness



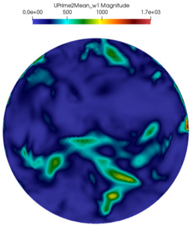
Volume void fraction is naturally bounded (and conserved) over the all domain

Spray analytics: few examples

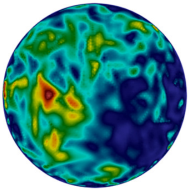
Velocity Magnitude at the Nozzle Ends



Nozzle Inlet U RMS



Nozzle Outlet U RMS



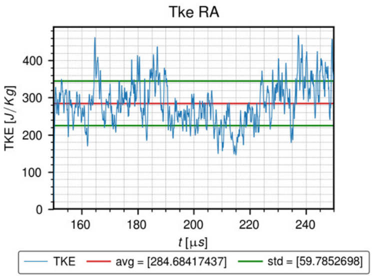
Weigthed Mass Flow Rate:

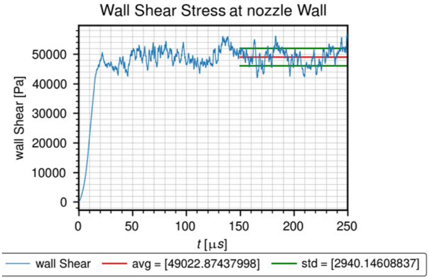
$$\langle \dot{m} \rangle = \left\langle \frac{\sum_{i=1}^{n} |\vec{U}_i| \rho_i |\vec{U}_i \cdot \vec{A}_i|}{\sum_{i=1}^{n} \rho_i |\vec{U}_i \cdot \vec{A}_i|} \right\rangle$$

Weigthed α (void fraction):

$$\langle \alpha \rangle = \left\langle \frac{\sum_{i=1}^{n} |\vec{U}_i| \alpha_{l,i} |\vec{A}_i|}{\sum_{i=1}^{n} \alpha_{l,i} |\vec{A}_i|} \right\rangle$$

Spray analytics: few examples

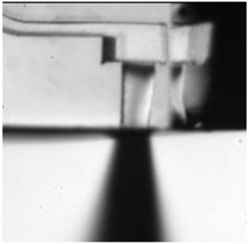


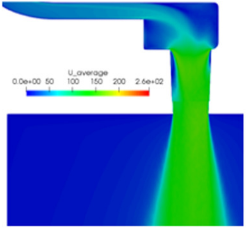


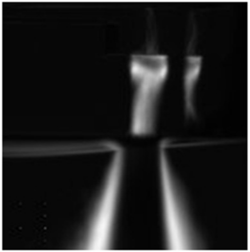
$$\frac{\int TKE_{FA}\langle\rho|\vec{U}\cdot d\vec{A}|\rangle}{\int\langle\rho|\vec{U}\cdot d\vec{A}|\rangle} = \frac{\sum_{i=1}^{n}|TKE_{FA}\langle\rho_i|\vec{U_i}\cdot\vec{A_i}|\rangle}{\sum_{i=1}^{n}\langle\rho_i|\vec{U_i}\cdot\vec{A_i}|\rangle} \qquad \left\langle\left|\frac{1}{A}\sum_{i=1}^{n}\left(\frac{\vec{\tau}}{\rho}\right)_i \rho_i|A_i|\right|\right\rangle$$

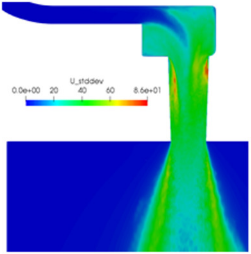
+ over 50 quantities monitored. Automatic checks on the monitored quantities is performed **(OPTIMIZATION)**.

ID-3: cone angle

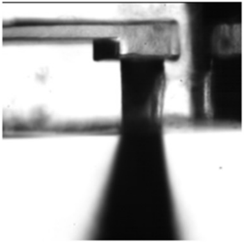


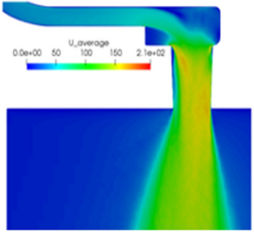


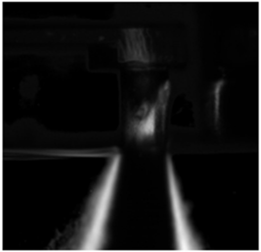


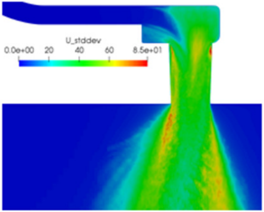


ID-10: cone angle



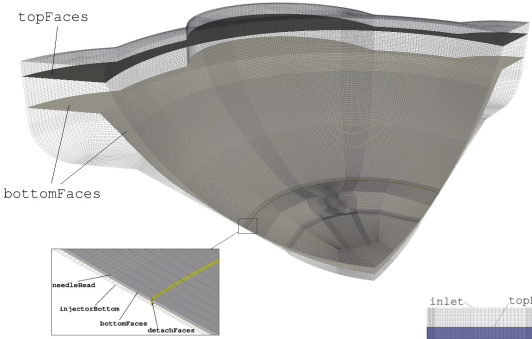


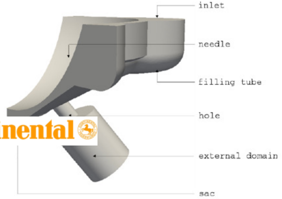


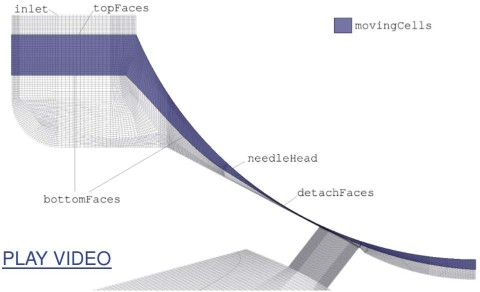


Continental

Needle transients

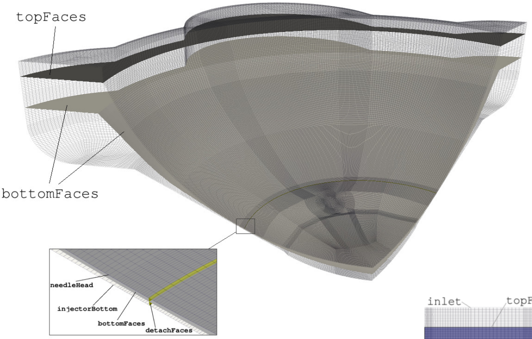






▷ PLAY VIDEO

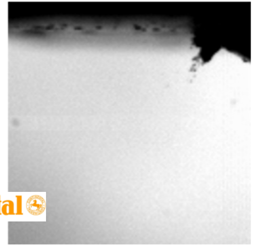
Needle transients

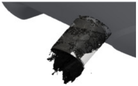


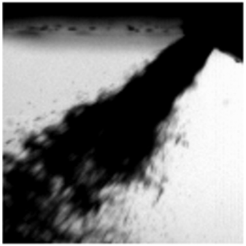
- F. Piscaglia, A. Montorfano, J. Helie, F.X. Demoulin. *"Development of a VOF Dynamic Solver in OpenFOAM®: an Application to the Simulation of the Opening and Closure Events in High Pressure GDI Injectors"*. **ICMF 2016 – International Conference on Multiphase Flows**, Italy, May 2016.
- F. Piscaglia, A. Montorfano et al. *"Hybrid RANS/LES of Moving Boundary Problems: Application to Cavitating Sprays and In-Cylinder Flows"*, **International Multidimensional Engine Modeling User's Group Meeting At the SAE Congress. April 2016**. https://imem.cray.com/2016/Meeting-2016/10-CI-spray-IMEM2016-PoliMi.pdf
- F. Giussani, A. Montorfano, F. Piscaglia, A. Onorati, J. Hélie, S. M. Aithal. *"Dynamic VOF modelling of the internal flow in GDI fuel injectors"*, **Energy Procedia**, 2016.

XL3.0 GDi Injector - Transient Simulation

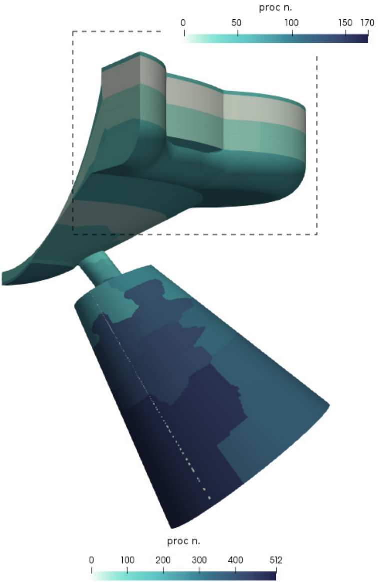
EXPERIMENTS | **SIMULATIONS**

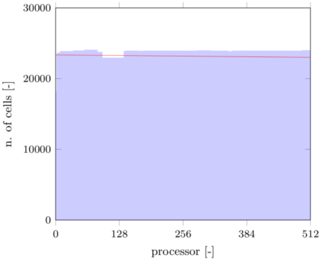






calcDecomposition



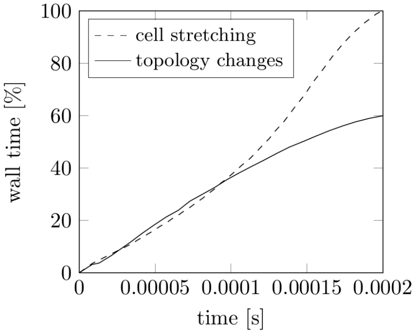


Massive parallelization with constrains from dynamicMesh handling 512 cores, 25M cells)

↓

CONSISTENT SPEED-UP (min 5x on large grids)

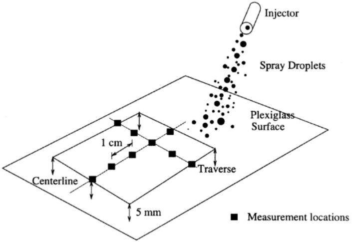
Parallel Performance

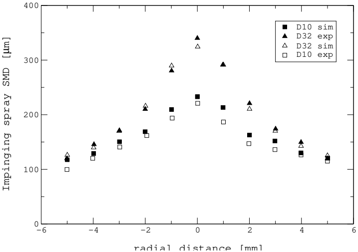


VOF Injection simulation: **solver performance with two different mesh motion strategies**.
Test carried out on 512 cores, 20 M cells.

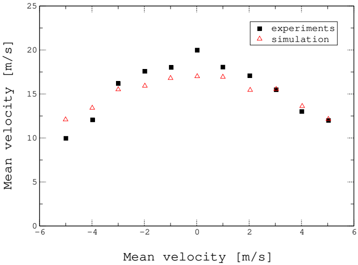
Lagrangian/Wall-Film Modeling

Spray modeling

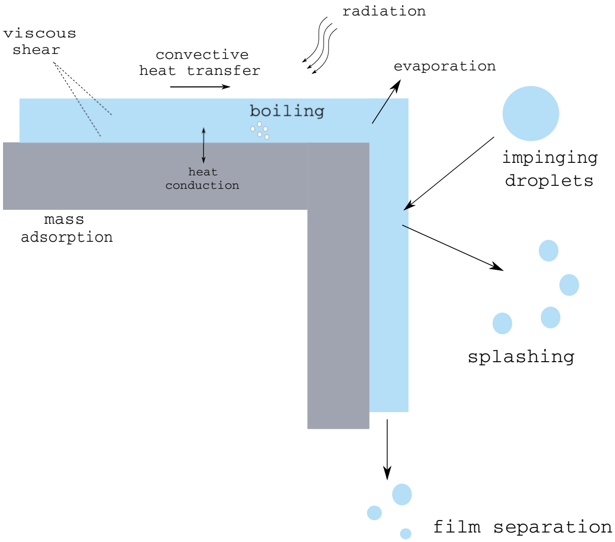




Authors	Models
L. Siebers	$\theta = 2\,\text{atan}\left(C_\theta((\frac{\rho_g}{\rho_f})^{0.19} - 0.0043(\frac{\rho_g}{\rho_f})^{0.5})\right)$
P. Cheng et al.	$\theta = \text{acos}\left(\frac{1}{\sqrt{1+TMR^2}}\right)$
S. C. Kong et al.	$\theta = 2\,\text{atan}\left(\frac{4\pi}{A}\sqrt{\frac{P_g}{\rho_g}}f(T)\right)$
J. M. Arrègles et al.	$\theta = 2\,\text{atan}\left((d_0)^{0.508}(P_{inj})^{0.00943}(\rho_g)^{0.335}\right)$
M. Arai et al.	$\theta = 0.05\left(\frac{\rho_g(P_{inj}-P_g)d_0^2}{\mu_g^2}\right)^{0.25}$

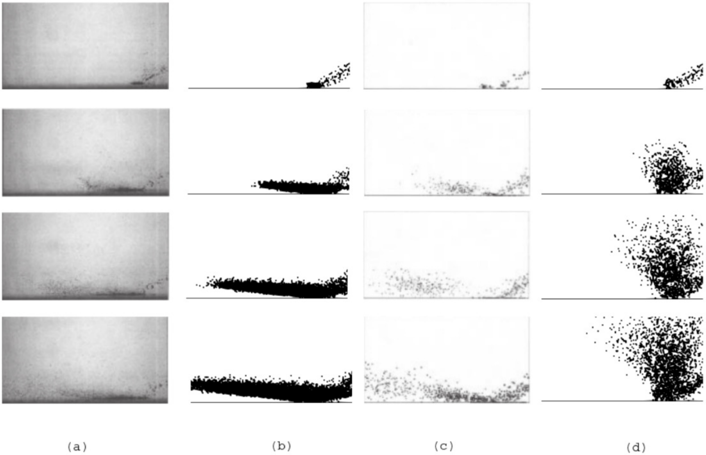


Finite-Volume (FV) surfaceFilm



Lagrangian/Wall-Film Modeling

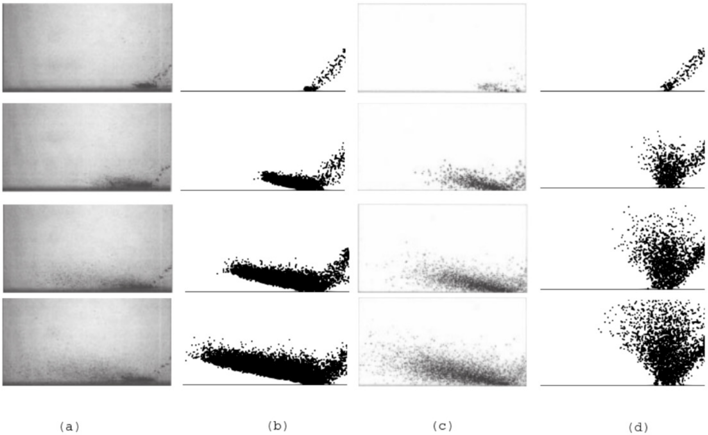
Spray angle = 30^o



a) experiments; b) LibPoliMi; c) KIVA-3V; d) OpenFOAM (standard)

Lagrangian/Wall-Film Modeling

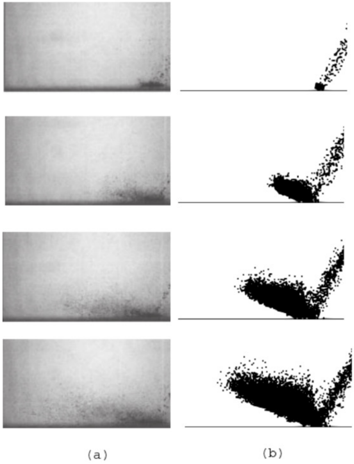
Spray angle = 45^o



a) experiments; b) LibPoliMi; c) KIVA-3V; d) OpenFOAM (standard)

Lagrangian/Wall-Film Modeling

Spray angle = 60^o



a) experiments; b) LibPoliMi; c) KIVA-3V; d) OpenFOAM (standard)

Summary of work

OpenFOAM is an ideal platform for research in CFD in **ACADEMIA**:

- most of the **state-of-the-art models** are already included and established in the code;
- it is **open-source**;
- the **object-oriented c++** structure allows a layered programming, which favors collaboration among different individuals: physical models are mostly compiled as dynamic c++ libraries and are easy to share among different developments.

Continuous development of OpenFOAM **by the principal developers** (and by the community) makes the code mature, stable and able to compete with the most established commercial CFD codes; for this reason, OpenFOAM has become a common tool also in **INDUSTRY**.

The aim of the work was to provide examples of the potential of OpenFoam and of its customization/extension by describing the implementation of a set of dynamic C++ libraries grouped in a **general purpose library** for the simulation of complex problems involving moving boundaries.

Thank you for your attention!

contact: **federico.piscaglia@polimi.it**