F. Piscaglia, F. Giussani, A. Montorfano, J. Hélie, S.M. Aithal
A MultiPhase Dynamic-VoF Solver to Model Primary Jet Atomization and Cavitation Inside High-Pressure Fuel Injectors in OpenFOAM
Acta Astronautica, Vol. 158, 2019, p. 375-387
doi:10.1016/j.actaastro.2018.07.026

A MultiPhase Dynamic-VoF Solver to Model Primary Jet Atomization and Cavitation inside High-Pressure Fuel Injectors using OpenFOAM

F. Piscaglia[*,a], F. Giussani[a], A. Montorfano[a], J. Hélie[b], S.M. Aithal[c]

[a] *Politecnico di Milano, Milan, Italy*

[b] *Continental Automotive SAS, Tolouse, Cedex, France*

[c] *Argonne National Laboratory, Lemont, IL 60439, United States*

Abstract

This paper describes the development of a dynamic two-phase Volume-of-Fluid (VOF) solver to study the physics of the primary jet breakup and flow transients induced by the nozzle geometry during the injector opening event in high-pressure injection using the OpenFOAM technology. The dynamic solver has been extended to support second-order discretization in time for moving mesh problems with automatic topology changes. The solver extension includes curvature effect in the interface tracking. Phase change at the interface was modeled using the Schnerr and Sauer model. While the solver is compatible with any kind of turbulence model, turbulence effect have been treated using Large-Eddy Simulations (LES). Detailed numerical studies are presented to demonstrate the conservation preservation property and accuracy of the solver. Code validation was performed by comparing numerical results with experiments on a Continental XL 3.0 6-hole prototype injector. Finally, limits of applicability of the cavitation model with a two-phase solver for the simulation of internal nozzle flows of injectors are discussed.

Key words: spray VOF, internal nozzle flow, primary atomization, OpenFOAM, GDI injection simulation, direct injection, dynamic mesh, needle motion, deforming domain, topological changes.

1. Introduction

Study of the fluid-dynamic behavior of single-orifice nozzles and fuel injectors with complex geometries is gaining significance among researchers [1, 2, 3, 4, 2, 5, 6], largely because of their widespread use in the automotive and aerospace industry. Atomization of the liquid spray ejected from the nozzle is mainly determined by the high-pressure, high-Reynolds upstream internal flow which, in turn, is strongly affected by cavitation. Cavitation is classified in two distinct forms, namely, "geometry-induced" and "vortex" (or string) cavitation. The geometry-induced cavitation is initiated at sharp corners where the pressure falls below the saturation value [7, 8] because of a sudden flow detachment and the accompanying recirculation region. String (or vortex) cavitation, conversely, develops by the evolution of the vorticity which allows the formation of geometry-scale vortices, and is significantly influenced by the walls and interaction with other vortices. Cavitation plays a pivotal role in achieving finer atomization of the spray, required for improving the fuel economy and reducing emissions [5]; however, it may also lead to a significant reduction in the nozzle volumetric efficiency and in the stability of the spray. Cavitation can also lead to potential damage and wear of the hardware components, leading to reduced reliability and increased cost. Furthermore, the role of cavitation is bound to gain importance as the fuel pressure and the geometric complexity of injectors increase, due to the tight coupling between cavitation and wall-generated turbulence. After a surface is initially affected by cavitation, it tends to erode at an accelerated pace: the cavitation pits increase the turbulence of the flow and creates crevices that act as nucleation sites for new cavitation bubbles, thus leading to an avalanche effect. The study of cavitation has spurred a renewed interest towards the understanding of the complicated flow physics inside injectors. Cavitation in Diesel injectors has been studied in detail by many researchers [1, 2, 3, 4]. In [2], enlarged replicas of valve-covered orifice injectors, incorporating tapered converging holes, were studied. A similar work on a large-scale nozzle has been presented in [3], where the compressibility of the liquid/vapor in the multiple phase mixture was accounted by a homogeneous equilibrium model. A detailed characterization of such large-scale vortical structures in a real geometry is a daunting task, because these vortical structures are highly

*Corresponding author, e-mail: federico.piscaglia@polimi.it

transient and strongly affected by the nozzle geometry, operating conditions of the injector and the needle motion [9, 10, 11, 12]. The simulation of the transient behavior of high-pressure injectors is therefore of great importance in the prediction of real spray characteristics, since the transient nature of the flow is greatly affected by the needle movement.

The aim of the present work is to implement a consistent framework for the simulation of internal nozzle flows in high pressure injectors, to study the influence of the sac-filling process and flow acceleration on the internal turbulent structures and the subsequent primary atomization. The multi-phase methodology used is the Volume-Of-Fluid (VOF) [13, 14] approach, in which the Navier-Stokes equations are solved for a fictitious fluid and weighted averaging of the fluid properties (mixture density and viscosity) is applied to form a closed system of equations. The volume fraction of the i-th component α_i is solved in an additional transport equation to account for the presence of a fluid interface. A summary of the theory behind the authors' development of a two-phase VOF solver for two isothermal immiscible fluids supporting cavitation is introduced in Sec. 3. The solver is an extension of the `interPhaseChangeFoam` solver-based family, originally available in all the releases of OpenFOAM based on the Foundation version [15, 16]. Modifications to the calculation of the interface curvature in solvers supporting cavitation are described in Sec. 5 and validation against numerical test cases is provided. In Sec. 6, the formulation of the advective term in the transport equation of the void fraction is derived, with the goal of ensuring proper conservation of the fluid-dynamic quantities in moving mesh problems; enforcement of the Geometry Conservation Law (GCL) to support second-order discretization in time and mesh topology changes are presented. The resulting dynamic solver and setup has been included in a set of dynamic libraries compatible with the latest versions of OpenFOAM released by the Foundation [15]. The novel methodology has been applied to the simulation of the needle opening event in a high-pressure injector, that was originally designed and produced by Continental Automotive SAS within the FUI-MAGIE Framework [17, 18] for the most recent Gasoline Direct Injection (GDI) engines. In Sec. 9, the implementation of advanced domain decomposition algorithms to ensure good scaling performance with mesh topology changes on large numbers of computing nodes is presented; finally, results from the simulations carried out on the HPC facilities made available by the Argonne National Laboratory are reported and discussed in Sec. 10. Comments about modeling condensation by the Schnerr-Sauer model in combination with a two-phase solver for the simulation of internal nozzle flows is discussed in Sec. 11.

2. The Continental XL 3.0 6-hole prototype injector

The geometry used for validation is the Continental XL 3.0 6-hole prototype injector (Fig. 1a), an injector especially developed to enable a detailed comparison of experiments with simulations. Fig. 8a shows a 1/6 section of the injector domain (the needle, the sac volume and the metering hole) and its immediate near-field, represented by a large inlet volume. The needle is located at the top: with the needle closed, a dead space is present in the sac volume, preventing the needle from touching the sensitive area of the hole inlet. Holes are oriented at 45 degrees with respect to the vertical direction and are straight (there is no nominal convergence factor); the hole length-to-diameter ratio is 1.1, the length (calculated along the axis) is equal to 0.221 mm and the hole diameter is equal to 200 mm. For a fully open needle position, the static mass flow rate has been measured on a real injector at different fuel pressures. An external cylindrical domain is placed below the hole to capture the primary atomization. Measured experimental needle lift data (Fig. 1-b) were obtained from Argonne National Laboratory (private communication).

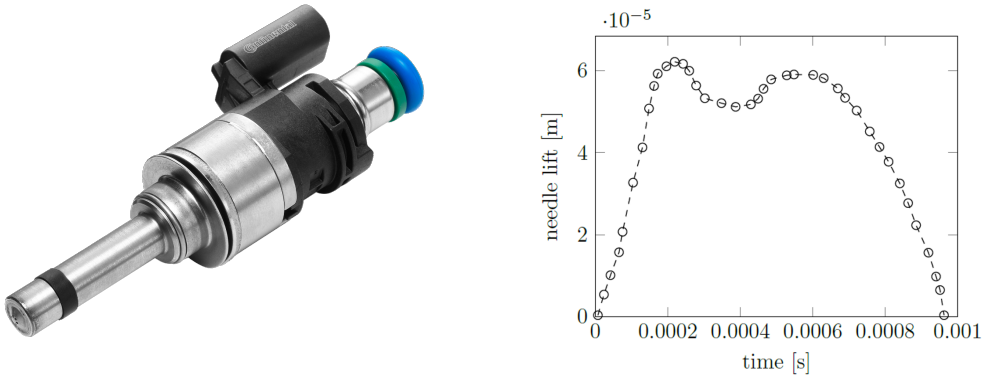
3. Two-phase VOF solver supporting cavitation

The basic idea of the VOF approach is that the two-phase system can be represented as a mixture of the phases in which the phase-fraction distribution includes a sharp yet resolved transition between the phases. The phase-fraction equations for VOF with incompressible two-phase flow are:

$$\frac{\partial \alpha_i}{\partial t} + \nabla \cdot \alpha_i(\mathbf{U} - \mathbf{U}_b) = S_\alpha \tag{1}$$

where $\mathbf{U}_b$ is the velocity of the moving grid, S_α is a source term proportional to the mass involved in the phase change (either cavitation or condensation). Each phase i in Eq. (1) is defined by its local volume fraction α_i and has a partial volume V_i, that is a fraction of the volume V of the cell element ($V_i \subseteq V$); $\alpha_i \in [0;1]$ and

$$\sum_i \alpha_i = 1 \tag{2}$$



(a) (b)

Fig. 1: (a) The Continental XL 3.0 6-hole prototype injector (b) measured experimental needle lift data (Dr. J. Wang, Argonne National Laboratory).

Eq. (1) is written in a general low-Mach formulation and considers a mixture of (two) fluids with different densities:

$$\rho = \sum_i \rho_i \tag{3}$$

and a mixture viscosity of:

$$\mu = \sum_i \alpha_i \mu_i \tag{4}$$

The source term S_α in the RHS of Eq. (1) includes the phase-change at the gas-liquid interface through the cavitation model and couples the effects of the cavitation with the evolution of the interface directly; the formulation used for the term S_α follows the theory of [19], as discussed in Sec. 4. In general, standard finite-volume discretization schemes are applied to all terms in the VOF system of equations except for the convection term (and possibly the transient term) in the phase-fraction equation. Numerical diffusion, which is very high in the transport term in second-order spatial discretization, "smears" the sharp liquid-gas inter-phase. In OpenFOAM, the strategy commonly followed in multiphase VOF solvers to model the transport of the void fraction consists of an add convection-based term which compresses the interface and preserves boundedness; this is similar to what is done for the treatment of the scalar-flux second-moment closure used for the "counter-gradient" transport in some complex combustion models describing the dynamic of turbulent flames [20]. In the VOF treatment, a common closure used for counter-gradient transport has the form:

$$\nabla \cdot [\mathbf{U}_c \, \alpha \, (1 - \alpha)] \tag{5}$$

where $\mathbf{U}_c$ is the compression velocity at the interface between the phases, which is a consequence of the different densities and the term $\alpha(1-\alpha)$ ensures boundedness. In the VOF solver used, the compression velocity is modeled as:

$$\mathbf{U}_c = c_\alpha |\mathbf{U}| \hat{\mathbf{n}} \tag{6}$$

C_α is the compression coefficient, that is typically of the order of 1 [21]. The compression rate should be set such that it ensures interface sharpness. Higher values of the compression rate might introduce numerical instability or slow convergence; in order to ensure that the compression term does not bias the solution, it should only introduce flow of α normal to the interface in the direction of the volume average interface normal $\hat{\mathbf{n}}$:

$$\hat{\mathbf{n}} = \frac{\nabla \alpha}{\|\nabla \alpha\|} \tag{7}$$

In the convention adopted, $\hat{\mathbf{n}}$ always points towards fluid 2, as shown in Fig. 2. In practical cases, $1 \leqslant C_\alpha \leqslant 4$ represents a good compromise to maintain a sharp interface in complex flows including, for instance, the breakup of the liquid-jet.

Finally, the transport equation for the void fraction

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot [\alpha(\mathbf{U} - \mathbf{U}_b)] + \nabla \cdot [\boldsymbol{U}_c \, \alpha \, (1 - \alpha)] = S_\alpha \tag{8}$$

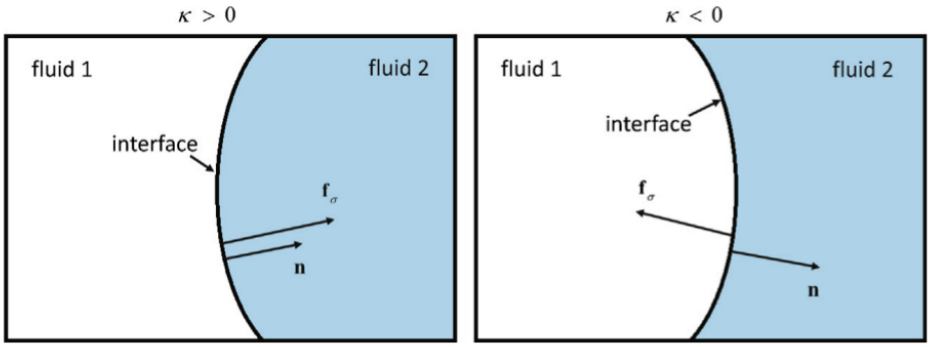


Fig. 2: Schematic of the interface between two fluids. $\mathbf{f}_\sigma$ is the surface tension force; $\hat{\mathbf{n}}$ and κ are namely the interface normal and the curvature; according to the sign convention adopted, $\mathbf{f}_\sigma$ is always oriented towards the concave interface, $\hat{\mathbf{n}}$ always points towards fluid 2.

is solved together with the governing equations for an isothermal fluid. The momentum equation reads:

$$\frac{\partial(\rho\mathbf{U})}{\partial t} + \nabla\cdot[\rho\mathbf{U}(\mathbf{U}-\mathbf{U}_b)] = -\nabla p + \nabla\cdot\boldsymbol{\tau} + \rho\mathbf{g} + \mathbf{f}_\sigma + S_U \tag{9}$$

where S_U are the momentum source terms, $\rho\mathbf{g}$ is the gravitational force term, $\mathbf{f}_\sigma$ is the surface tension calculated at the fluid interface and it is always oriented towards the concave interface (Fig. 2):

$$\mathbf{f}_\sigma = \sigma\,\kappa\,\hat{\mathbf{n}} \tag{10}$$

In (10), σ is the surface tension, $\hat{\mathbf{n}}$ is the unit normal vector of the liquid interface and κ is the interface curvature, which is defined as:

$$\kappa \equiv \nabla\cdot\hat{\mathbf{n}} \tag{11}$$

Eq. (9) is combined with the divergence-free constraint for incompressible flows to construct a Poisson equation for pressure, where the source-terms representing the mass transfer between the phases are included.

4. Cavitation model

The formulation of the term S_α of Eq. (1) is based on the model by Schnerr-Sauer [19]; in this work, the condensation term has been deactivated, as it will be discussed in Sec. 11; hence, S_α is defined as:

$$S_\alpha = -C_v\frac{\rho_v\rho_l}{\rho}\frac{3\alpha_l\alpha_v}{R_B}\sqrt{\frac{2}{3}\left(\frac{|p-p_{sat}|}{\rho_l}\right)} \tag{12}$$

where ρ is the mixture density, ρ_l and ρ_v are the liquid and vapor density respectively, α_v is the vapor volume fraction and p_v is the saturation vapor pressure, C_v is a tuning parameter for the model (set to 1 in this work); R_b denotes the radius of the bubbles, that is related to the density of nuclei n_b and the diameter of nuclei d_{nuc} by:

$$\alpha_v = (1-\alpha_l+\alpha_{nuc}) = \frac{\frac{4}{3}\pi R_b^3 n_b}{1+\frac{4}{3}\pi R_b^3 n_b} \tag{13}$$

with:

$$\alpha_{nuc} = \frac{\frac{\pi d_{nuc}^3 n_b}{6}}{1+\frac{\pi d_{nuc}^3 n_b}{6}} \tag{14}$$

The cavitation model by Schnerr-Sauer is based on the Rayleigh-Plesset equation and therefore belongs to the class of bubble based continuum models: in this family of models, usually applied to incompressible phases, the cavitation is assumed to be blubby and the density of nuclei is required as an input parameter.

5. Modification at the interface treatment in the solver

At the time of writing of this paper, the standard implementation of the two-phase VOF solver with cavitation (`interPhaseChangeFoam`) available in any official distribution of OpenFOAM does not calculate the fluid interface curvature κ at each time-step, after the transport of the void fraction with cavitation is solved. Following the approach of other existing multiphase solvers available in the code [21], the Continuum Surface Force (CSF) model of Brackbill [22] is used to model the unit normal vector to the liquid interface:

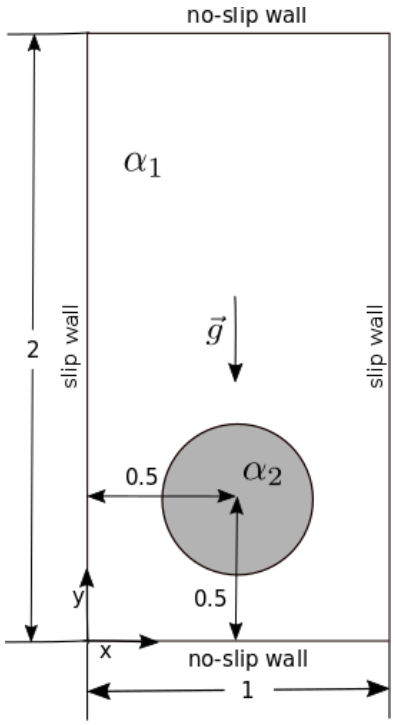
$$\hat{\mathbf{n}} = \frac{\nabla\alpha}{\|\nabla\alpha\|} \simeq \hat{\mathbf{n}}_{12} = \frac{\nabla\alpha_1|_f}{\|\nabla\alpha_1\|_f} \tag{15}$$

and the curvature of the interface between the two phases

$$\kappa = -\sum_f \mathbf{n}_{12} \cdot \mathbf{S}_f \tag{16}$$

is calculated and updated at each time-step. The subscript f (for face) in Eqs. (15) and (16) is used for quantities that are interpolated on cell faces, starting from their value at the cell center.

Quantitative validation of the described code extensions was done by comparing the results of our simulations with other CFD simulations on a benchmark problem proposed by [23]. We compare our results with three incompressible interfacial flow codes used in [23]: TP2D (Transport Phenomena in 2D) [24, 25], FreeLIFE (Free-Surface Library of Finite Element) [26] and MooNMD (Mathematics and Object-oriented Numerics in MagDeburg) [27]. TP2D and FreeLIFE are based on the level-set approach applied on a static grid, while in MooNMD the movement of the interface is done in a Lagrangian manner, after which the inner mesh points are fitted to the new interface by an elastic mesh update by solving a linear elasticity problem. A similar comparison with the VOF OpenFOAM solver `interFoam` has been published in [28]; since the integration of the surface tracking on the modified cavitating solver `interPhaseChangeFoam` has never been tested, a new validation was deemed to be necessary. In the tests reported in this section, the phase-change-related term S_α in Eq. (1) has been deactivated. The test case studied consists of a two-dimensional rising bubble problem, where a gas bubble immersed in a chamber filled with liquid moves until it breaks up. The case setup and boundary conditions are described in Fig. 3. Liquid and gas phase fractions in the fluid are identified by α_1 and α_2 respectively. Forces acting on the bubble are surface tension and gravity. The domain has an aspect ratio height/width=0.5; no-slip boundary conditions on the velocity are set at the upper and lower boundaries, while free-slip is applied at the right and left bounds; gravity $\mathbf{g}$ is oriented towards the negative y direction. At time $t = 0$ s, the bubble center is located at $(x, y) = (0.5, 0.5)$ and the bubble radius is $r_{b,0} = 0.25$ m. For a fair comparison with [23], the grid used is Cartesian with a resolution 320x640 cells.



ρ_1	$[\mathrm{kg\,m^{-3}}]$	1000
ρ_2	$[\mathrm{kg\,m^{-3}}]$	1
μ_1	$[\mathrm{kg\,(ms)^{-1}}]$	10
μ_2	$[\mathrm{kg\,(ms)^{-1}}]$	0.1
$\mathbf{g}$	$[\mathrm{m\,s^{-2}}]$	-0.98
σ	$[\mathrm{N\,m^{-1}}]$	1.96
Re	[–]	35
Eo	[–]	125

Fig. 3: Domain, boundary conditions and fluids physical properties of the "bubble rising" test case

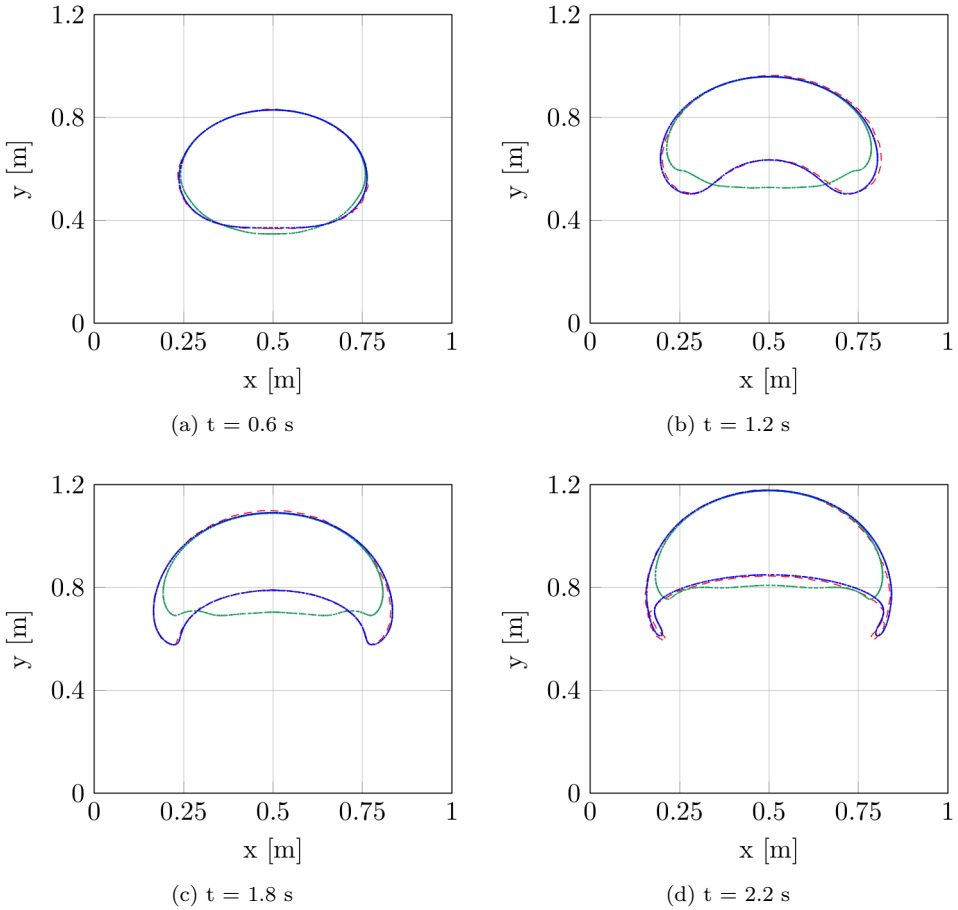
The physical properties of the fluid have been reported in the table in Fig. 3: Eo is the Eötvös number, defined as the ratio between the buoyancy force and surface tension:

$$\mathrm{Eo} = \frac{\rho_1 U_g^2 L}{\sigma} \tag{17}$$

while Re is the Reynolds number of the liquid;

$$\mathrm{Re} = \frac{\rho_1 U_g L}{\mu_1} \tag{18}$$

where $L = 2r_{b,0}$ the characteristic length scale and $U_g = \sqrt{2gr_{b,0}}$ is the characteristic rising velocity. At high values of Eo, the observed bubble shape will be in between the shape observed for the skirted and the dimpled ellipsoidal-cap regimes, implying that breakup is likely to occur [29]. Problems with high values of Eo challenging for interface tracking algorithms and can yields different numerical solutions in the evolution and formation of newly created droplets.

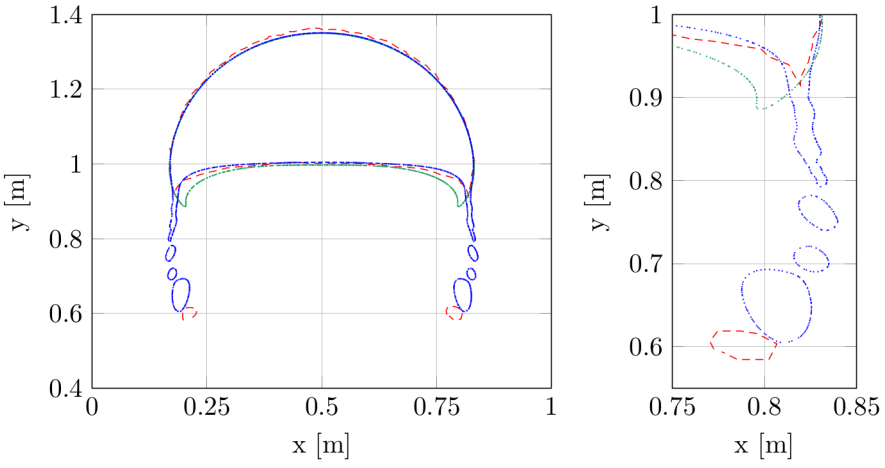


(a) t = 0.6 s (b) t = 1.2 s

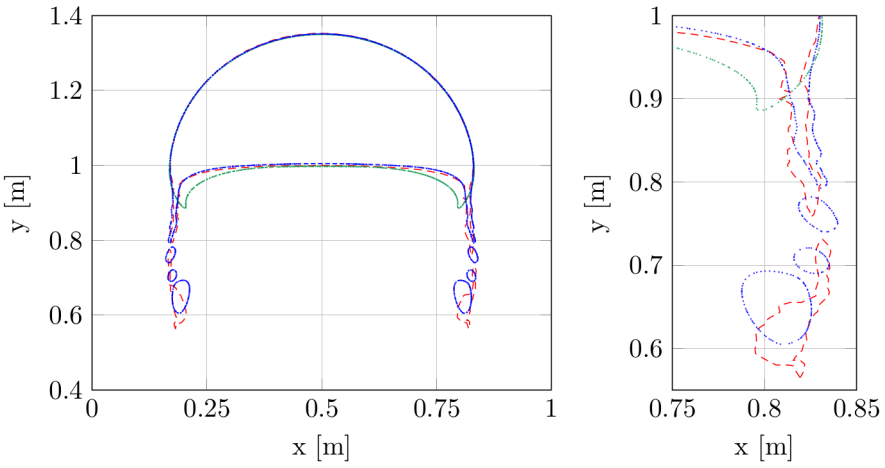
(c) t = 1.8 s (d) t = 2.2 s

Fig. 4: Bubble evolution from 0.6 to 2.2 s: — `interPhaseChangeFoam` (standard); — `interPhaseChangeFoam` (modified); - - TP2D code [23].

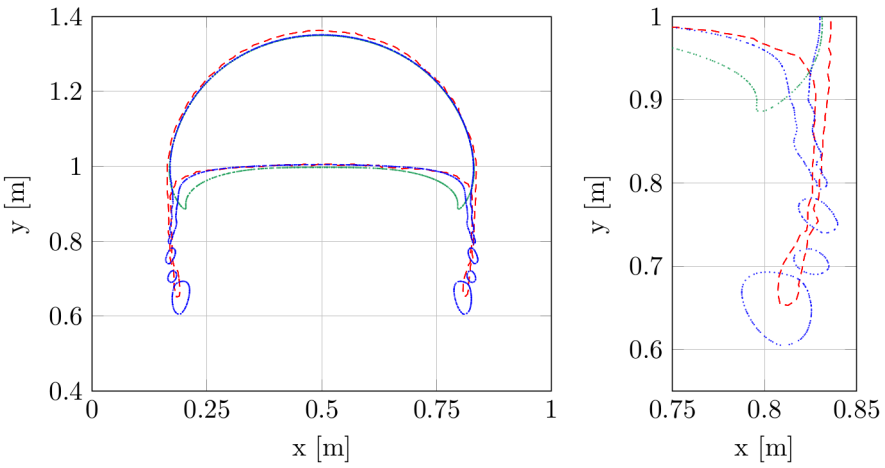
Similar to [23], bubble evolution has been tracked in the simulations for a total time of 3 time units, the latter being $T = L/U_g$ and a fixed time step with $\Delta t = 1/640$ s has been used for time marching. Second-order backward differencing schemes have been used for time derivatives, while second-order central schemes were used for the discretization of spatial operators. Initial conditions are defined by setting $\alpha_2 = 0$ to indentify the gas region (bubble) and $\alpha_1 = 1$ for the surrounding liquid. Since this leads to a stair-cased shaped interface, a preliminary simulation with $\mathbf{g} = 0$ has been performed to relax the interface and to obtain a smooth bubble shape. Results of a precursor simulation at $t = 3$ s were used as initial values for the rising bubble simulation.



(a) TP2D



(b) FreeLife



(c) MooNMD

Fig. 5: Bubble breakup at time t=3 s; comparison between — `interPhaseChangeFoam` (standard), — `interPhaseChangeFoam` (modified) and: (a) TP2D; (b) FreeLife, (c) MooNMD.

Comparisons of the calculation of the evolution of the bubble interface, identified by $\alpha_1 = \alpha_2 = 0.5$, have been compared at different time intervals reported in [23] and are shown in Fig. 4 and 5. Starting from a quiescent situation, the bubble first deforms towards the top, progressively reaching the skirted shape; later on, thin filaments are formed and, in the end, droplets detach from the edges. The green line in Fig. 4 is the interface calculated by the standard `interPhaseChangeFoam` solver; a clear discrepancy with the benchmark case is apparent in Fig. 4 (a) at t = 0.6 s and worsens as the simulation proceeds; neither the skirted shape, nor

the detached droplets are predicted. Results obtained from our modified solver are plotted in blue in Figs. 4 and 5 and compared with predictions of the bubble breakup from other codes. The modified solver (blue line) shows significant the improvement in the interface tracking as compared with the standard `interPhaseChangeFoam` solver (green line). In Fig. 5, comparisons of the bubble breakup at $t = 3$ s is reported. Each approach provides different predictions of bubble breakup and, in absence of experimental data, it is hard to state which method is the most accurate, as also stated in [23]. On the other hand, it is apparent that from Fig. 5 that without the proposed code extensions the bubble breakup is not captured by the code.

For a more quantitative analysis of the results achieved, a comparison of the geometrical metrics [23] is provided:

1) *bubble center of mass*:

$$\mathbf{x}_c = \frac{\iint_A \alpha_2 \mathbf{x}_c \, \mathrm{d}x\mathrm{d}y}{\iint_A \alpha_2 \, \mathrm{d}x\mathrm{d}y} \tag{19}$$

2) *degree of circularity* for a two-dimensional domain [30]:

$$C = \frac{\text{perimeter of equivalent circle}}{\text{perimeter of bubble}} = \frac{2\pi r_{eq}}{\iint_A \nabla\alpha_2 \, \mathrm{d}x\mathrm{d}y} \tag{20}$$

where r_{eq} is the equivalent radius, defined as:

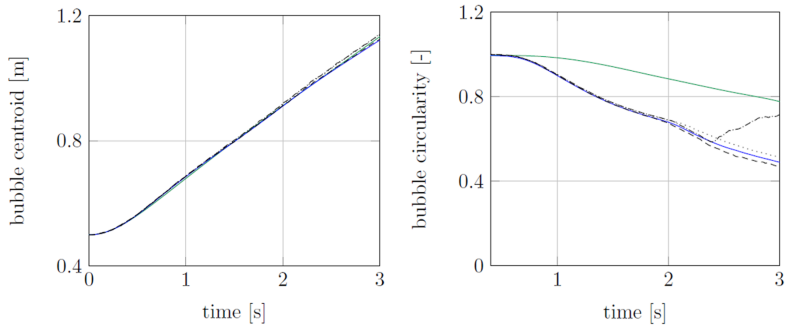
$$r_{eq} = \frac{1}{\pi}\left(\iint_A \alpha_2 \, \mathrm{d}x\mathrm{d}y\right) \tag{21}$$

the parameter C is equal to unity for a perfectly circular bubble and lower than unity for other cases;

3) *mean rising velocity*:

$$\mathbf{u}_c = \frac{\iint_A \alpha_2 \mathbf{u} \, \mathrm{d}x\mathrm{d}y}{\iint_A \alpha_2 \, \mathrm{d}x\mathrm{d}y} \tag{22}$$

The temporal evolution of $\mathbf{x}_c$, C and $\mathbf{u}_c$ is shown in Figs. 6 and 7, respectively. The bubble centroid location is satisfactorily predicted by all models, with no appreciable differences at least up to $t = 2$ s. The circularity index and the rising velocity predicted by the VOF method are very close to the benchmark simulations; without the correction on the curvature, a strong deviation of the solver from the reference starting at t = 0.5 s is apparent; in particular, the greatest deviation is observed in the circularity index, because it is linked to the bubble shape. It is likely that the procedure of modeling the surface tension effects in the different codes is the the main reason for the difference in the evolution of the bubble shape at about t=2.4 s, when breakup starts. While the authors' VOF solver begins to produce elongated filaments while transitioning to the skirted bubble shape, the TP2D presents a retraction of the filaments causing the bubble circularity to grow again after the breakup has occurred; FreeLife and MoonMD behave similarly. Since there are no experimental results, it is hard to reliably ascertain which of the methods is more accurate. Fig 7 shows the rising velocity $\mathbf{u}_c$, which is captured fairly well by the standard VOF solver, with the exception of the local maximum around t = 2 s. The modified solver shows good qualitative comparisons with other published results but discrepancies still exist between various models possibly due to the differences in capturing the breakup (Fig. 5(b)), as discussed earlier.



(a) bubble centroid (b) bubble circularity

Fig. 6: Two-dimensional rising bubble problem, validation test case. Evolution in time of: (a) bubble centroid location C; (b) bubble circularity $C \in [0; 1]$; circularity is equal to unity if the bubble shape is a perfect circle. Legend: — `interPhaseChangeFoam` (standard), — `interPhaseChangeFoam` (modified), — · — TP2D code, — — — FreeLIFE code, ··· MooNMD.

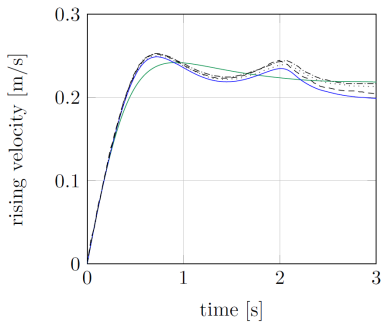


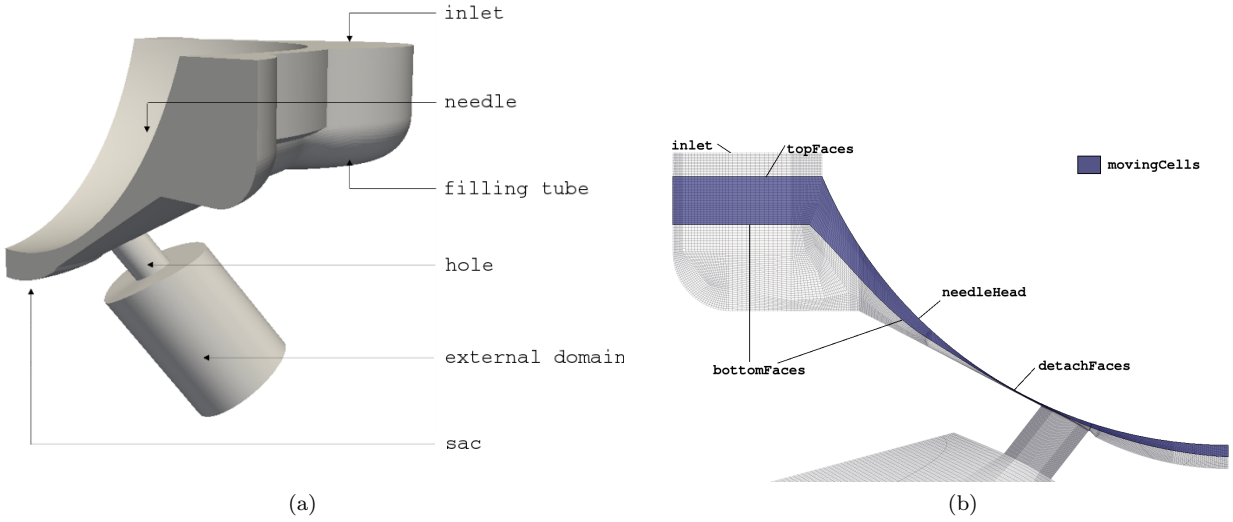
Fig. 7: Two-dimensional rising bubble problem, validation test case. Evolution in time of the bubble rising velocity $\mathbf{u}_c$. Legend: — `interPhaseChangeFoam` (standard), — `interPhaseChangeFoam` (modified), — · — TP2D code, — — — FreeLIFE code, · · · MooNMD.

6. Extension of the VOF formulation to moving boundary problems

Two kinds of moving mesh strategies can be employed to handle the displacement of the cell vertices in the computational mesh, in order to simulate the prescribed motion of the needle:

- the displacement of the mesh interior domain vertices can be calculated by a semi-constrained motion solver to preserve the mesh quality during its dynamic update. This functionality is available in the official releases of OpenFOAM [15] and it therefore can be applied without any extension to the code. However, this functionality has some inherent limitations, which are related to the continuous change of the cell volumes and shapes: in particular, as cells are compressed, a reduction of their size Δx leads to a reduction of the time-step advancement due to the CFL criterion. Furthermore, the quality of the grid deteriorates with large deformation. While the use of a Laplacian-based mesh smoother can help preserve good mesh quality for longer time intervals, this aspect of change in cell volumes is still a limitation. Hence, mesh-to-mesh interpolation is usually adopted in combination [31, 32] with the Laplacian-based smoothing approach.

- An alternative strategy (Fig. 8b) to move the mesh may be based on the use of automatic topological changes. In this case, a single initial mesh is provided and an ad-hoc algorithm handles the mesh motion in a fully automatic fashion, by adding and removing layers of hexahedral cells [33]. At the time of writing of this paper, only basic features that employ topological changes are available in OpenFOAM and hence significant developments were required to conduct the simulations presented in this work [34]. There are multiple advantages of using a technique based on automatic topological changes for problems with large domain deformations. Firstly, the approach preserves the initial mesh quality (skewness, non-orthogonality and aspect ratio) during the entire simulation, since grids at different time steps differs only in layers of fully orthogonal hexahedral cells. Furthermore, the changes in the resulting multi-block hex-dominant grid are triggered only locally and the global morphology is not recalculated at runtime making the computations extremely fast and efficient. Additional details about the methodology may be found in [33, 34, 35].



(a) (b)

Fig. 8: (a) 1/6 section of the injector domain (the needle, the sac volume and the metering hole) and its immediate near-field, represented by a large inlet volume. (b) Details of the face sets used to perform dynamic layering: `topFaces` and `bottomfaces` are used to perform layer addition/removal, while `detachFaces` are used to perform dynamic attach/detach; `needle-head` is the needle patch. Generation of the hexablock mesh is employed by an in-house automatic tool developed by the authors within the OpenFOAM technology framework.

The computational grid of the injector, shown in Fig. 8b, is initially divided into two regions (the injector sac and the nozzle) by an inner baffle, which is used to disconnect the injector sac from the nozzle at needle closure. The needle closure is automatically triggered by the code when the distance between the needle surface and the nozzle hole is lower than a set threshold value; in the simulations, the threshold value was assumed to be 0.5 μm (less than 1% of the maximum lift). For the moving needle, the user must define:

- the list of cells (`cellSet`) that will be rigidly moved along a prescribed direction by a prescribed needle lift motion. In Fig. 8b, the area colored in purple represents the set of cells of the needle that are moving; cells belonging to this `cellSet` can be of any kind (hex, tet, prism, hybrid) and they are adjacent to a layer of hexahedral cells in the upper region;
- one or multiple lists of faces (`faceSets`), where dynamic layer addition/removal will be applied. For any given face set, the mesh handling algorithm automatically calculates the average cell height of the cell layer (hexahedral or prismatic) at run-time, to determine whether layer addition and/or removal must be triggered. Cells in a deleted layer are merged into neighboring cells, with the value in the resulting cell being the volume-average of deleted and neighboring cells; cell thickness of added layers can be defined either by a dictionary or calculated by the motion solver to optimize the mesh quality. Checks for boundary proximity and prevention of topological inconsistencies is performed by the code at run-time and the dynamic removal of cell layers is automatically deactivated, if needed [33];
- a single list of faces (`detachFaces`, see Fig. 8b) where dynamic attach/detach of the boundaries is applied. In particular, the attach/detach mesh modifier is automatically triggered by the dynamic mesh solver to temporarily attach/detach two regions of the mesh through an arbitrary (conformal or not conformal) set of faces, namely the `detachFaces`, to simulate opening/closure events.

The implemented dynamic VOF solver supports all of the above-mentioned strategies to move the grid; the order of accuracy obtained by the two strategies is similar. The authors have extended the dynamic solver used in this work to preserve second order accuracy in time with topology changes [36]. This extension will be briefly described in sec. 7. The use of topological changes in the mesh motion requires a particular structure of the mesh during generation and ad-hoc decomposition methods (discussed in sec. 9), while no particular requirements are needed during pre-processing if cell deformation is applied. The mesh motion strategy based on topological changes thus leads to a significant increase in the simulation speed when compared to the other strategies. In Fig. 9, the performance of the two meshing strategies is compared: during the transient simulation the needle moves according to the law reported in Fig. 1b). During needle opening (time $t \leq 2 \cdot 10^{-4}$ s), cells are expanded with cell stretching and the mesh motion does not significantly impact the time step advancement; if a method based on topological changes is applied, hexahedral layers of cells are added and the mesh size progressively increases. As a result, the simulation times of the two methods are comparable: the time required by the continuous topology changes is compensated by the lower time needed by the solver to converge. When

the needle reaches the maximum lift at time $t \simeq 2 \cdot 10^{-4}$ s (see Fig. 1b), the convergence rate of the solver on the mesh where layers were added is much faster, since the initial quality of the mesh has been preserved. Additionally, during needle closure (not shown in Fig. 9), cell size decreases during compression and the time-step becomes progressively lower; on the other hand, with layer removal, cells-size do not change and the time-step size during time marching is not influenced by the mesh, but only by the local flow velocity. The resulting overall solution time of the transient simulation,including opening and closure, is cut by over 40% if a mesh motion based on topological changes is used.

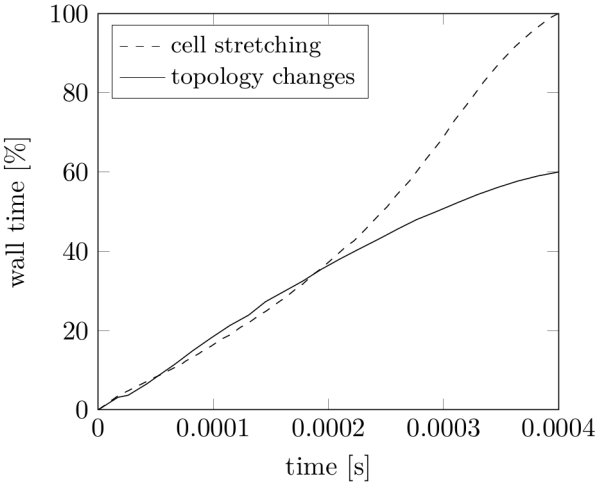


Fig. 9: Comparison in performance between different mesh motion strategies to perform a transient dynamic simulation of the needle opening. In the y axis, non-dimensional walltime needed to complete the transient simulation (opening and closure event) of the injector is plotted against the physical time (x-axis). The grid used with cell stretching was a 15 M cell mesh generated by `snappyHexMesh` [15], of comparable quality (calculated at the initial time-step of the simulation) with respect to the grid used in this work (Fig. 8b).

There are several reasons behind this reduction of computational time: first, the calculation of the Laplace equation (or any point smoother) is not applied at each time step (unless specifically required by the user), since only few points are moving due to the prescribed rigid motion. As a consequence: a) grids at different time-steps differs only for layers of hexahedral cells and the initial mesh quality (skewness, non-orthogonality and aspect ratio) is preserved during the entire simulation, enabling a high convergence rate of the solution over the multi-block grid; b) the time-step of the simulation is affected only by the local flow conditions because the local mesh size, with the exception of a single layer of cells, does not change in time. With the cell stretching methodolgy, a reduction of the cell size occurs during the needle closure leading to a significant reduction of the time step due to the CFL condition [33]. The proposed method is computational very efficient also when applied to large grids, since mesh changes are triggered only locally on some processors and the morphology of the mesh is not recalculated globally [34].

7. Dynamic VOF solver: numerical considerations

The discretized form of governing equations with moving grids must be modified to account for the change of the cell volumes during the solution. Since the location of the grid is known as a function of time, the solution of the Navier-Stokes equations must be written using relative velocity components at the cell faces; the conservation equations for scalar quantities are then easily derived from the corresponding equations for a fixed Control Volume (CV), by replacing the flow velocity vector $\boldsymbol{U}$ in the convective term with the relative velocity $\boldsymbol{U} - \boldsymbol{U}_b$, $\mathbf{U}_b$ being the velocity at which the grid (integration boundary) moves. Hence, the phase transport equation (1) takes the form:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \alpha(\boldsymbol{U} - \boldsymbol{U}_b) + \nabla \cdot [\boldsymbol{U}_c \, \alpha \, (1 - \alpha)] = \boldsymbol{S}_\alpha \tag{23}$$

When topological changes are triggered, the reconstruction of the history of the volume changes during mesh motion is non-trivial; for a generic variable ϕ, the conservation equation in the semi-discretized form for a cell of volume V is defined as:

$$\frac{\partial}{\partial t}(\rho \, \phi) + \sum_f \rho_f \, \phi_f \, (\varphi_f - \varphi_{m,f}) - \sum_f \gamma_\phi \nabla \phi_f = S_\phi \tag{24}$$

In Eq. (24), ρ is the flow density, γ_ϕ is the diffusion coefficient, S_ϕ are the volume sources/sinks of ϕ, φ_f is the absolute cell face flux $\varphi_f = \mathbf{U}_f \cdot \mathbf{n}$; $\varphi_{M,f}$ is the mesh flux due to point motion. To avoid any spurious mass sources, the formulation of $\varphi_{M,f}$ must fulfill the Geometric Conservation Law (GCL) [37]:

$$\frac{d}{dt}\int_V dV - \int_S \mathbf{u}_\mathrm{b} \cdot d\mathbf{S} = 0 \tag{25}$$

which in its semi-discretized form is:

$$\left.\frac{dV}{dt}\right|_{t=n-1}^{t=n} - \sum_f \varphi_{m,f} = 0 \tag{26}$$

The fulfillment of the Discrete Geometric Conservation Law form (DGCL, Eq. 26) determines the method of solution for the mesh flux $\varphi_{M,f}$; a correct solution of the Discrete GCL (DGCL) ensures a proper mass conservation in a constant-topology moving grid. In a moving boundary problem, variables at time (n) are mapped onto the new mesh by a mapping procedure (interpolation) $\mathcal{M}_c^{n+1}$, at $t = n+1$, where the new solution must be solved. If the mesh topology is not varying, there is a one-to-one correspondence in the addressing of the cells of the grids at different timesteps and:

$$\mathcal{M}_c(\{c_0 \dots c_m\}^n) = \{c_0 \dots c_m\}^n \mapsto \{c_0 \dots c_m\}^{n+1} \tag{27}$$

All flow intensive variables (pressure, velocity, etc.) can therefore be simply transformed from the old to the new mesh as:

$$\phi(t^n, \mathbf{x}^{(n+1)}) = \mathcal{M}_c[\phi(t^n, \mathbf{x}^n)] \tag{28}$$

With layer addition/removal, if cells are split or merged across a time step the one-to-one correspondence between the cells of the old and the new mesh is lost and the fulfillment of Eq. (26) throughout the mesh change becomes more difficult to accomplish. Besides, the transported quantity ϕ and the cell volumes at the old times (V^{n-1} and, if second order discretization in time is used, V^{n-2}) must be estimated in added/merged cells to compute the time derivatives $\partial\phi/\partial t$ of Eq. (24). Hence, when layer addition/removal occurs, the mesh change is computed in two steps. First, the mesh topology is modified and the volume history together with the conserved variables are mapped onto the new mesh, before cell points are moved, in order to store the mapped quantities for the calculation of the time derivatives of the conserved variables at the new time step; since the face-to-face map cannot be computed across a topology change, cell face fluxes φ_f^n are then recovered for the whole mesh. In the second step, points are displaced according to the needle motion and DGCL is enforced by a modified form of Eq. (28) and (27).

Even if the conservation of the DGCL is ensured, the recovered fluxes mapped over the faces of the CVs are no longer compatible with the pressure field derived from the continuity equation calculated by the segregated solver at the old timestep with the old mesh. When the mesh is moving, this causes an error in any conservation equation where an advective term is present. For incompressible flows, where velocity $\mathbf{U}$ is the primary variable in the momentum equation, the error of non-conservation in the momentum across remapping is apparent in the moving cells and it is proportional to Δt; on the other hand, with compressible flows, the error is less apparent (in spite of being present) because it is split among the transported conserved variables $\rho\mathbf{U}$, for which the momentum equation is solved. It is important to note that this source of error is not directly related to the use of topological changes, but to the moving mesh in general. For each time step of the dynamic solver it is therefore necessary to solve an additional Poisson equation (Eq. 29) for a pressure correction p' [38]

$$\frac{\partial p'}{\partial t} - \nabla^2 p' + \nabla \cdot \varphi^n(\mathbf{x}^{n+1}) - [\nabla \cdot \varphi^n(\mathbf{x}^n)]^{n+1} = 0 \tag{29}$$

to ensure consistency between the face fluxes and the pressure field, which is related to the mass conservation:

$$\varphi^{n+1} = \varphi^n(\mathbf{x}^{n+1}) + \nabla p' \tag{30}$$

The above-mentioned treatment of cell- and face-centered variables ensures conservativeness of the main flow quantities, as long as the DGCL is fulfilled and continuity is enforced by Eqs. (29) and (30). Only in this way the boundedness and conservativeness of Eq. (23) can be satisfied with mesh motion, in particular with topology changes.

In multiphase flow calculations, the flux inconsistency between time-steps leads to a significant non-conservation and unboundedness of the phase-fraction equation. The flux inconsistency is more apparent if the transport equation for the phase fraction is solved explicitly: in OpenFOAM, the error induced by the face-to-face mapping becomes large with the flux corrected transport technique (called MULES, Multidimensional Universal Limiter for Explicit Solution [39]) used for the calculation of the continuity of the phase fraction. For each case studied in this work, it was verified that the void fraction was lower than 1 during the entire simulation.

8. Numerical solution method and computations

A three-dimensional 15 million-cell grid was used to simulate a 30 degree circumferential segment of the injector, which shows a symmetry in the circumferential direction. The modeled domain includes a volume representing the ambient where the flow jet penetrates. A pressure ratio of 30 bar was set between the inlet and the outlet section, to reproduce the experimental conditions; hence, total pressure of 30 bar was set at the inlet section, while a (total) pressure of 1 bar was set at the outlet. Since transient conditions were studied, the injector was assumed to be closed at the beginning of the simulation and the liquid was assumed to be stagnant in the injector sac; no liquid was assumed to be present in the nozzle region and in the part of the injector sac which remains linked to the nozzle during the needle closure: this initial setup represents the experimental testing condition where the injector is perfectly "clean" before the test (no residual liquid drops from previous tests are present in the sac before the opening of the needle). The WALE model [40] was chosen to model turbulence. A no-slip boundary condition for velocity was applied at the walls, while cyclic boundary conditions, implicitly coupled for all flow independent variables, were used for the circumferential boundaries of the solution domain. Second-order discretization in time and space was used for the algebraic operators in the governing equations. The transport of the phase-fraction is solved explicitly at the beginning of each time step by means of the Multi Universal Limiter for Explicit Solution (MULES) [21], a Flux Corrected Transport method to ensure the boundedness of the α flux. In addition, special Total Variation Diminishing (TVD) methods were applied to velocity compression terms. Experiments were performed at ambient temperature and the heat flux measured at the walls was negligible; it was therefore reasonable to assume density variations linked only to pressure (in particular if cavitation is present) and to neglect the influence of the temperature in the simulations. Simulations were run on the Bebop Cluster at Argonne National Laboratory, which is equipped with 352 Intel Xeon Phi 7230 KNL (Knights Landing) nodes operating at 1.30 GHz with a 128 GB/node RAM (2 GB/core). The KNL architecture has many new features for speeding up computations. Each core has two 512-bit vector units (compared to 256-bits with Haswells/Sandy Bridge) and supports AVX-512 SIMD instructions leading to significant improvement in scalar and vector performance. The vector peak performance is close to 3 TF for double precision computations and close to 6 TF for single precision computations. Additional features such as four-way multi-threading (or 256 threads per node), multiple memory tiers, large high bandwidth memory (MCDRAM) and large bulk memory can enable better performance on compute bound, memory latency bound and memory bandwidth bound problems. The dynamic solver could complete a full simulation of the injector opening in less than 48 hours on 512 KNL cores.

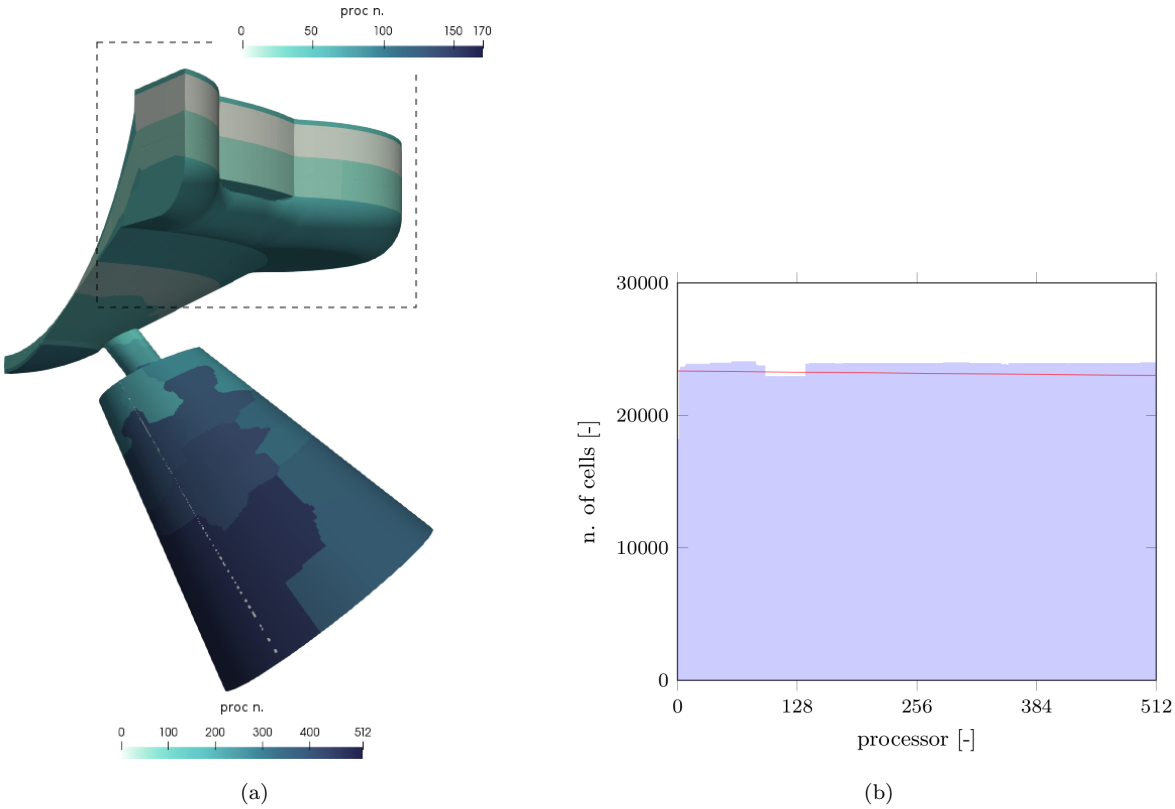
9. Domain decomposition with topological changes

To run parallel simulations, the Finite-Volume (FV) mesh has to be decomposed into a set of sub-domains, each to be assigned to a single core for processing. In simulations involving mesh motion with topological changes, new constraints in domain decomposition arise. In OpenFOAM, as well as in most of the CFD codes, topological changes cannot occur across inter-processor patches between neighboring sub-domains and the decomposition algorithm must be constrained in this sense; in other words, decomposition must account for the constraints given by the dynamic layering and, as a consequence, it might lead to unbalanced decomposition if automatic procedures are used (see [33]).

A new application `calcDecomposition` has been developed to employ a semi-constrained (balanced) domain decomposition and to simplify an inherently complex process. The following steps are performed:

1. different mesh regions, in accordance with the decomposition constraints (layer addition/removal zones, attach/detach zones, unconstrained), are defined;
2. each region can be independently decomposed by selecting a specific decomposition algorithm to be applied to an arbitrary (user-defined) number of sub-domains;
3. mesh regions where dynamic layering will be applied are decomposed to ensure processor boundaries to be perpendicular to the layering faces (see Fig. 8b);
4. the remaining cells of the mesh are distributed among processors using non-constrained decomposition.

The domain decomposition strategy described is very flexible and allows for balanced decomposition with complex geometries and topological changes, as shown in Fig. 10b. Almost linear speed-up (strong scaling) could be achieved with 512 cores; tests on the same setup have proven that this would not have been possible with standard/automatic decomposition algorithms [41] typically applied for small/medium grids.



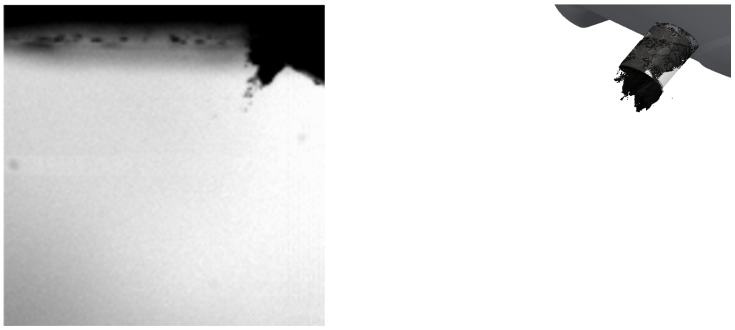
(a) (b)

Fig. 10: a) Balanced domain decomposition is achieved by selecting multiple regions in the mesh and by decomposing them among different numbers of processors; each region can be decomposed by a different logic; b) decomposition of the injector mesh on 512 cores resulting from the developed algorithm. The number of cells per core is very well balanced (imbalance is $< 5\%$). The red line represents the number of cells per processor for a perfect balance. For the case considered, linear scalability was achieved up to 896 cores.

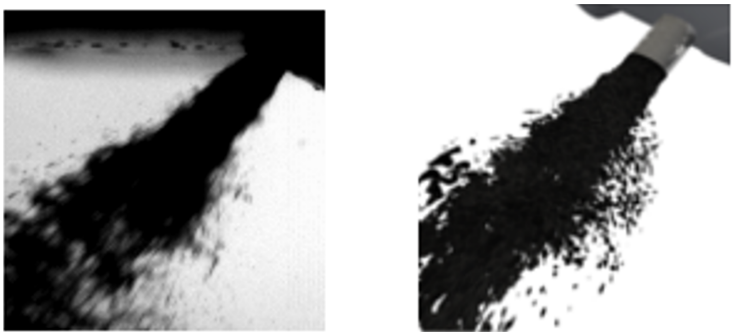
10. Injector opening: simulation results and validation

In the VOF method, the two phases are described in the Eulerian framework and an additional equation is solved for the liquid volume fraction to track the position of the gas-liquid interface. The main advantage of the method is that it can handle liquid structures of any shape and that the wakes caused by droplets are directly resolved on the computational grid: this limits the use of models in the simulation to a minimum and therefore leads to a high accuracy. Additionally, including the modeling of phase-change at the interface, in the interface transport equation, couples the effects of cavitation to the evolution of the interface directly. The main drawback with this approach is the high computational cost, as a fine resolution is required to achieve high accuracy. The focus of this work is to study the evolution of the cavitating fluid across the nozzle, with particular attention to the fluid composition near the nozzle exit. With the HPC facilities usually available for calculation, the mesh resolution required by a VOF method is prohibitive to accurately predict fluid phenomena such as the secondary breakup of the liquid droplets, the aerodynamic drag or the evaporation of droplets. Hence, the undisturbed region (the external cylindrical domain far from the nozzle) must be mostly intended to limit the influence of the outlet boundary condition on the numerical solution of the internal flow field.

In Fig. 11, isosurfaces of the liquid phase fraction describing the evolution of the jet-flow near the nozzle exit are compared with high-resolution high-speed camera visualizations acquired at the experimental facility at Continental Automotive SAS. The temporal evolution of the spray plume, the primary atomization and the near-field spray breakup structures seem to be properly captured; the spray angle on the cavitating side of the jet (Fig. 11b), where droplets are formed, is also well described. The simulation is able to capture the flow separation near the sealing band, that causes the deviation of the spray plume trajectory from the nozzle hole axis and a reduction of the total spray angle.

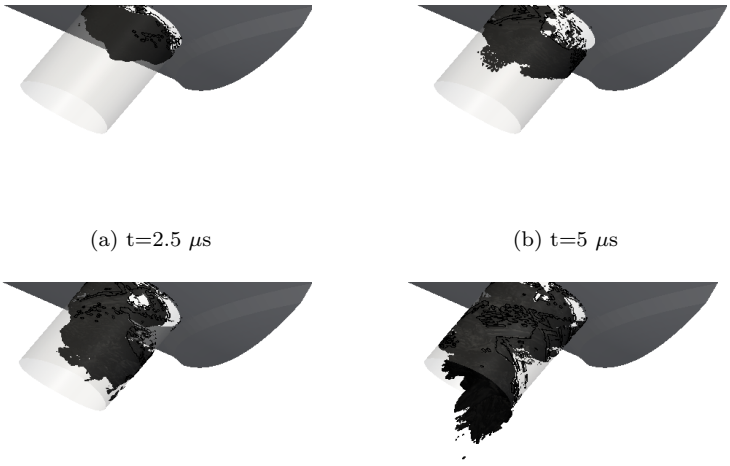


(a) t=0.9 μs aSOI



(b) $t = 200\,\mu$s aSOI

Fig. 11: Comparison between experimental high-resolution/high-speed camera visualization carried out at Continental SAS on the XL 3.0 injector (left column) and the isosurfaces of the liquid phase fraction from the simulations (right column). Snapshots refer to: (a) t=0.9 μs; (b) $t = 200\,\mu$s after the start of injection (aSOI). Injection pressure is 30 bar.



(a) t=2.5 μs

(b) t=5 μs

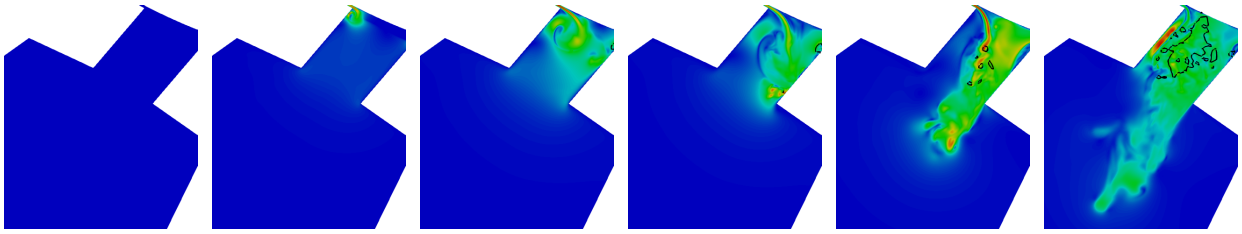
(c) t=7.5 μs

(d) t=10 μs

Fig. 12: Evolution of the flow at the early stage of injection, from the needle opening. The formation of a main vortex, whose axis is parallel to the hole axis, is evidenced. The injector is conventionally assumed to start opening at time t= 0 [s].

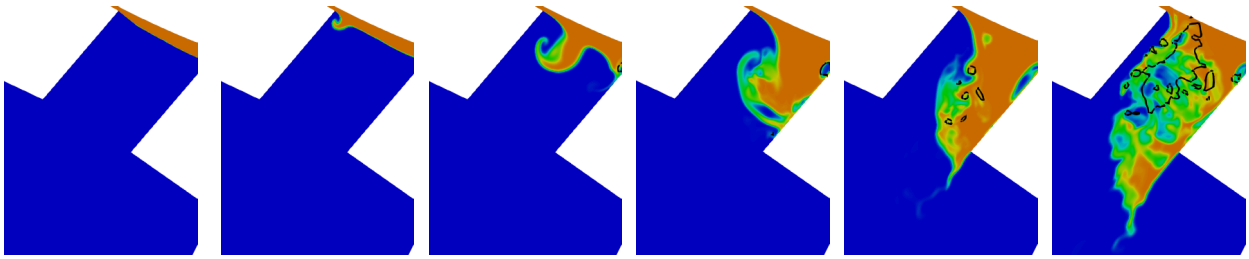
Since the predicted evolution and the main features of the jet flow are in good agreement with the experimental visualizations, simulations have been considered suitable for a deeper analysis also in those regions where experimental visualizations are impossible to perform. Fig. 12 shows that, as the needle starts opening, the pressurized jet accelerates, impinges on the nozzle wall and generates a recirculating region at the nozzle inlet. While separating, the flow progressively deviates from the nozzle hole axis; a non-symmetric evolution of the flow in the form of main vortex, whose axis is parallel to the hole axis, is registered along the nozzle and in the near-field region (ambient). In Fig. 13a-13e, the contour plot of the velocity flow field is reported over a cross-plane; on the same plane, the evolution of the phase fraction α is reported in Figs. 14a-14f. Figs. 13 and 14 show how cavitation evolves inside the nozzle: a jet flow generates near the region where flow is separating

(Fig 13a-13d) and a "geometry induced" cavitation appears (Fig. 14a-14a). In Fig. 15 the evolution of the volume fraction, at the early stage of injection, on three different planes cutting the injector nozzle is shown. Constant pressure (black) isolines at $p = p_{sat}$ are assumed to limit the cavitating regions; there are some small areas in the nozzle where $p = p_{sat}$, which most probably delimit the cavitation. A recirculating region and a pair of counter-rotating vortices, whose axis is normal to the nozzle plane, is formed in the jet core together with finer turbulent structures responsible for the instability of the jet (Fig. 13e).



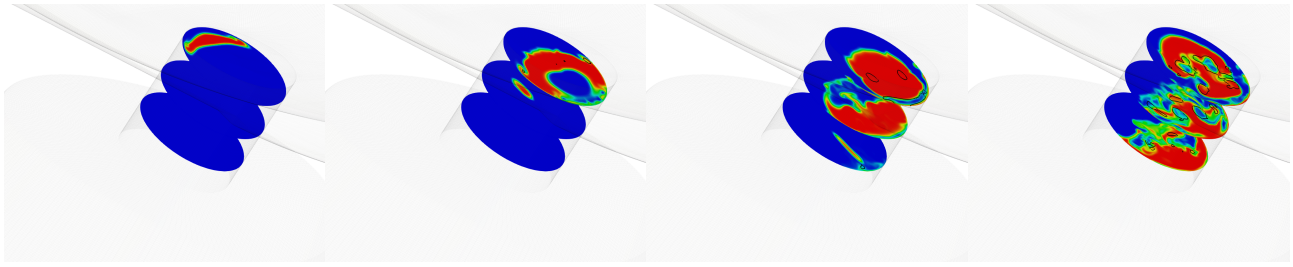
(a) t=2.5 μs (b) t=5 μs (c) t=7.5 μs (d) t=10 μs (e) t=12.5 μs (f) t=15 μs

Fig. 13: Contour plot of the velocity flow field over a mid-plane across the injector. Black lines limit the cavitating regions at $p \leq p_{sat}$.



(a) t=2.5 μs (b) t=5 μs (c) t=7.5 μs (d) t=10 μs (e) t=12.5 μs (f) t=15 μs

Fig. 14: Contour plot of the liquid phase fraction (blue=liquid, red=gas) over a mid-plane across the injector. Colors indicates the liquid fraction, which ranges from blue (full liquid, α=1) to red (gaseous phase, α=0). Black lines limit the cavitating regions at $p \leq p_{sat}$.

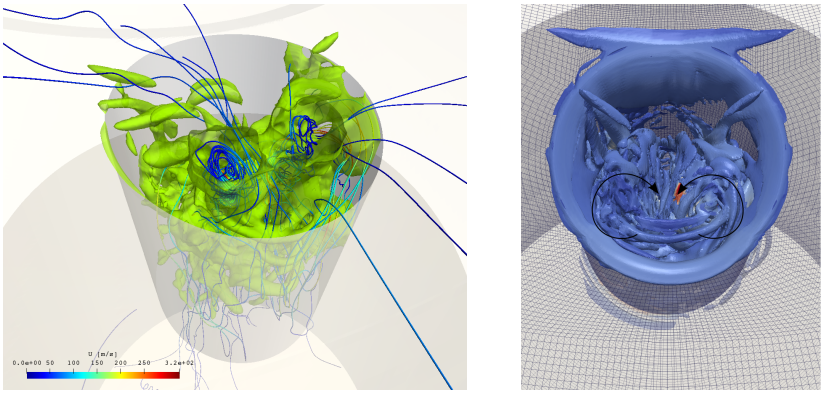


(a) t=2.5 μs (b) t=5 μs (c) t=7.5 μs (d) t=10 μs

Fig. 15: Evolution of the volume fraction, at the early stage of injection, on 3 different planes along the nozzle axis. Black lines are the contour of the saturation pressure, where cavitation occurs. Colors indicates the liquid fraction, which ranges from blue (full liquid, α=1) to red (gaseous phase, α=0).

Between t=12.5 μs and t=15 μs, the cavitating gas near the jet tip (black contour, Fig. 14f) expands and compresses the liquid, that in turn increases its velocity over the tangential direction, to form another additional counter-rotating vortex pair, whose axis of rotation is now parallel to the nozzle axis: this is evidenced by the red region in Fig. 13e, where the highest velocity magnitude is located not far from the wall and in Fig. 16, where a greater detail of the turbulent structures is provided. The low pressure in the vortex core causes the fuel to cavitate and elongated vapor regions are formed along the nozzle axis. In Fig. 16, isosurfaces represent the areas at $p = p_{sat}$, where cavitation mainly occurs. Cavitating zones are mostly located in the proximity of

two counter-rotating string vortices, that are formed by the flow pattern which develops at the injector opening and that are evidenced by the lines calculated by the Q criterion.

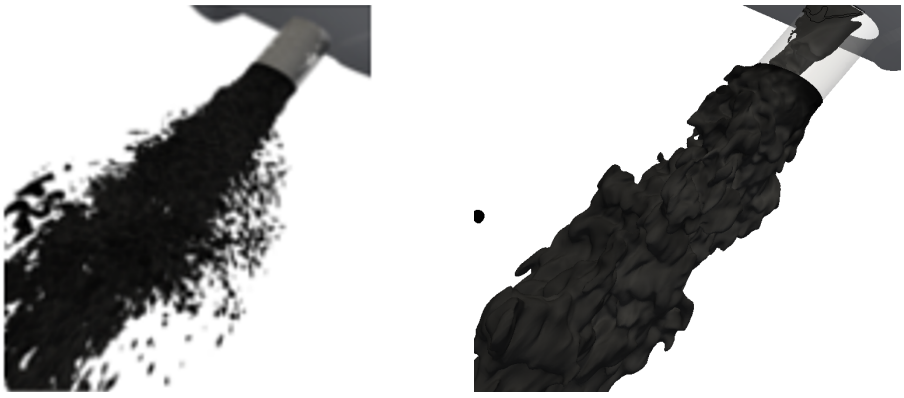


(a) velocity field (b) Q criterion

Fig. 16: (a): iso-surfaces identifying the regions at $p = p_{sat}$; velocity streamlines show the formation of vortical structures; (b) a counter-rotating vortex pair is identified by the Q criterion, that has been used to identify vortical structures inside the nozzle at the early stages of the injector opening. Time instant: t=20 μs.

11. Important remarks about condensation

The results presented in the paper were obtained after deactivating the condensation term in the Schnerr-Sauer model [19]. The solver used in this work accounts only for two-phases (liquid and gaseous), where each phase is represented by a single-component fluid. With `interPhaseChangeFoam`, it is therefore not possible to track the air and the fuel vapor within the gaseous phase. Fig. 17 shows the effects of including the condensation term in the current two-phase solver. Fig. 17 shows that inclusion of the condensation term in the Schnerr and Sauer model tends to favor the phase-change of air into fuel in the condensing regions, introducing a significant error in the calculation of the liquid phase fraction particular in the regions near the nozzle exit as shown in Fig. 17(b). The air trapped within the jet as shown in Fig. 17(a), that is usually depressurized because of its high speed, is absent and manifests itself as additional condensed liquid fuel. The resulting jet looks more dense and the cone angle is underestimated.



(a) no condensation term (b) with condensation

Fig. 17: Modeling condensation with a two-phase single-component solver 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. (a): simulation of the jet flow with the condensation term deactivated in the Schnerr and Sauer model. (b) simulation of the jet flow including condensation.

Neglecting condensation is justified for the problem considered in this work, largely because the nozzle geometry and the operating conditions studied have been specifically designed to study cavitation and greatly limit condensation of the cavitating bubbles in the nozzle. For this reason, the simulations show good agreement with experimental results during the early stage of injection. Condensation would have been important to

capture the flow physics far from the nozzle exit and to simulate the transient operation at the needle closure. In regions far from the nozzle exit, the jet plume is pressurized by the drag force against the stagnant air in the chamber, which induces the dispersed fuel vapor to be condensed into liquid: this usually happens in the regions where the applicability of the VOF is questionable and therefore it is not of primary interest for the present work. During the needle closure, the pressure in the nozzle hole decreases and the gas phase (air+vapor) from the tank flows back and recirculates into the nozzle hole. For this case, tracking tracking the evolution of the air and the fuel-vapor separately by the transport of their void fractions along with condensation effects, is essential. Detailed development and validation of a multiphase solver supporting the tracking of three interfaces (liquid/air, liquid/fuel vapour and air/fuel vapour) to simulate needle closure is currently underway. Another alternative would be to model the flow as a multi-component mixture: in this case, the air and the fuel vapor would be represented as two components identified by their mass fractions in a two-phase VOF single-fluid solver. The underlying assumption at the base of this approach is the perfect mixing between the two gaseous components (fuel vapor and air), which is strong assumption if swirling cavitation in the injector nozzle is studied.

12. Conclusions

The development and validation of a two-phase dynamic VOF solver for the simulation of internal nozzle flows has been presented. The derivation of the dynamic formulation of the transport term has been combined with the low-mach formulation of the transport equation of the liquid fraction. While the results presented in this paper use mesh handling based on topological changes developed by the authors, the proposed mathematical formulation is valid for any kind of dynamic mesh handling. Effects of the interface curvature have also been included in the developed solver, to improve the ability of the solver to predict sharp interfaces. Validation of the experiments carried out on an injector specifically designed for gasoline direct injection (GDI) engines, provided by Continental Automotive SAS, shows that there is very good agreement in the description of the temporal evolution of the spray plume and of the primary atomization, demonstrating that the temporal evolution of the liquid-phase nozzle flow and the near-field spray breakup structures are properly captured. The dynamic solver presented is second-order accurate in space and time, which is not trivial when dynamic topologically changing grids are present. The implementation of a robust domain decomposition methodology enabled very good balanced decomposition, leading to excellent strong scalability of the code on large supercomputers: a full LES simulation of the injector opening could run in less than two days on the new Intel Xeon Phi 7230 KNL (Knights Landing) nodes. The approach can be considered to be reliable for investigations of the internal-nozzle flows during transient operation, to drive the early design stage of high-pressure GDI injectors. On the other hand, as discussed in the paper, tracking only two void fractions may be an oversimplifying assumption when condensation of the fuel vapor plays an important role in the physics of the problem, as it happens with some injector nozzle geometries and at needle closure. Extensions to the solver to track three void fractions and two interfaces, together with an ad-hoc cavitation model represent current authors' work.

Acknowledgments

Authors gratefully acknowledge the Laboratory Computing Resource Center (LCRC) at Argonne National Laboratory for the computing resources provided to the authors through the `KNL-OPENFOAM-VOF` project and Dr. Jin Wang (Advanced Photon Source, Argonne National Laboratory) and his Research Group, where X-Ray measurements of the needle lift were performed. The code development used for the simulations was linked as a set of dynamic libraries to OpenFOAM-dev, the version of the code released by the OpenFOAM Foundation (`commit 3e1ab675f6e59bbc99e0574c19d7d1ac3d94cc36`).

References

[1] B. A. Reid, G. K. Hargrave, C. P. Garner, G. Wigley, An investigation of string cavitation in a true-scale fuel injector flow geometry at high pressure, Physics of Fluids 22 (3) (2010) 031703–031703.

[2] M. Gavaises, A. Andriotis, D. Papoulias, N. Mitroglou, A. Theodorakakos, Characterization of string cavitation in large-scale diesel nozzles with tapered holes, Physics of Fluids 21 (5).

[3] A. Papoutsakis, A. Theodorakakos, E. Giannadakis, D. Papoulias, Les predictions of the vortical flow structures in diesel injector nozzles, SAE Technical Paper 2009-01-0833.

[4] M. Ghiji, L. Goldsworthy, P. Brandner, V. Garaniya, P. Hield, Numerical and experimental investigation of early stage diesel sprays, Fuel 175 (Supplement C) (2016) 274–286.

[5] S. Makhlouf, J. Hélie, G. Grimoux, J. Cousin, L. Gestri, A. Wood, G. Wigley, Large eddy simulation of cavitation and atomization in injector flows using openfoam, in: ICLASS 2012, 12th Triennial Int. Conference on Liquid Atomization and Spray Systems, Heidelberg, Germany, 2012.

[6] B. Befrui, G. Corbinelli, D. Robart, W. Reckers, H. Weller, LES Simulation of the Internal Flow and Near-Field Spray Structure of an Outward-Opening GDi Injector and Comparison with Imaging Data, SAE Technical Paper 2008-01-0137.
[7] H. Chaves, M. Knapp, A. Kubitzek, F. Obermeier, et al., Experimental study of cavitation in the nozzle hole of diesel injectors using transparent nozzles, SAE Technical Paper 950290.
[8] U. Iben, A. Morozov, E. Winklhofer, F. Wolf, Laser-pulse interferometry applied to high-pressure fluid flow in micro channels, Experiments in Fluids 50 (3) (2011) 597–611.
[9] L. NaiXian, F. X. Demoulin, J. Reveillon, J. Chesnel, Large Eddy Simulation of Cavitation and Atomization in Injector Flows using OpenFOAM, in: ILASS-Europe 2014, 26th Annual Conference on Liquid Atomization and Spray Systems, Bremen, Germany, 2014.
[10] F. Giussani, A. Montorfano, F. Piscaglia, A. Onorati, J. Hélie, S. Aithal, Dynamic vof modelling of the internal flow in gdi fuel injectors, Energy Procedia 101 (Supplement C) (2016) 574 – 581, aTI 2016 - 71st Conference of the Italian Thermal Machines Engineering Association. doi:https://doi.org/10.1016/j.egypro.2016.11.073.
URL http://www.sciencedirect.com/science/article/pii/S1876610216312826
[11] H. Watanabe, M. Nishikori, T. Hayashi, M. Suzuki, N. Kakehashi, M. Ikemoto, Visualization analysis of relationship between vortex flow and cavitation behavior in diesel nozzle, Int. Journal of Engine Research (1) (2014) 5–12.
[12] E. de Villiers, A. Gosman, H. Weller, Large eddy simulation of primary diesel spray atomization, in: SAE Technical Paper, SAE Int., 2004.
[13] O. Ubbink, R. Issa, A method for capturing sharp fluid interfaces on arbitrary meshes, J. Comput. Phys. (153). doi:10.1006/jcph.1999.6276.
[14] O. Ubbink, Numerical Prediction of Two Fluid Systems with Sharp Interfaces, Ph.D. thesis, University of London, UK (1997).
[15] T. O. Foundation, Openfoam-dev user guide.
URL https://cfd.direct/openfoam/user-guide
[16] ESI-OpenCFD. [link].
URL www.opencfd.co.uk
[17] M. M. Khan, J. Hèlie, M. Gorokhovski, N. Sheikh, Experimental and numerical study of flash boiling in gasoline direct injection sprays, Applied Thermal Engineering 123 (9) (2017) 227–389.
[18] J. Hèlie, M. M. Khan, M. Gorokhovski, Large eddy simulation of a turbulent spray jet generated by high pressure injection: impact of the in-nozzle flow, Journal of Turbulence 17 (9).
[19] J. Sauer, G. H. Schnerr, Unsteady cavitating flow: A new cavitation model based on modified front capturing method and bubble dynamics (2000 2000).
[20] W. T.-F. Section, H. G. Weller, The development of a new flame area combustion model using conditional averaging (1993).
[21] H. Weller, A new approach to VOF-based interface capturing methods for incompressible and compressible flow, Technical report TR/HGW/04, OpenCFD Ltd. (2008).
[22] J. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, Journal of Computational Physics 100 (2) (1992) 335 – 354. doi:https://doi.org/10.1016/0021-9991(92)90240-Y.
URL http://www.sciencedirect.com/science/article/pii/002199919290240Y
[23] S. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, L. Tobiska, Quantitative benchmark computations of two-dimensional bubble dynamics, Int. Journal for Numerical Methods in Fluids 60 (11) (2009) 1259–1288.
[24] S. Osher, J. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations, J. Comput. Phys. (79) (1988) 12–49. doi:10.1016/0021-9991(88)90002-2.
[25] S. Turek, Efficient solvers for incompressible flow problems, an algorithmic and computational approach, in: Lecture notes in computational science and engineering, Vol. 6, Springer, Berlin, 1999.
[26] N. Parolini, E. Burman, A finite element level set method cor viscout free-surface flows, in: Applied and Industrial Mathematics in Italy, Proceedings of SIMAI 2004, World Scientific, Singapore, 2005, pp. 417–427.
[27] V. John, G. Matthies, MooNMD – a program package based on mapped finite element methods, Computeing and visualization in science (9) (2004) 1179–1195. doi:10.1016/j.apnum.2006.03.003.
[28] J. Klostermann, K. Schaake, R. Schwarze, Numerical simulation of a single rising bubble by vof with surface compression, Int. Journal for Numerical Methods in Fluids 71 (8) (2013) 960–982. doi:10.1002/fld.3692.
URL http://dx.doi.org/10.1002/fld.3692
[29] R. Clift, J. Grace, M. Weber, Bubbles, Drops, and Particles, Dover Civil and Mechanical Engineering Series, Dover Publications, 2005.
[30] H. Wadell, Sphericity and roundness of rock particles, The Journal of Geology 41 (3) (1933) 310–331.
URL http://www.jstor.org/stable/30058841
[31] Jasak, H. and Z. Tukovic, Dynamic Mesh Handling in OpenFOAM Applied to Fluid-Structure Interaction Simulations, in: Proceedings of the V European Conference on Computational Fluid Dynamics (ECCOMAS CFD), Lisbon, 14-17 June, 2010.
[32] F. Piscaglia, A. Montorfano, A. Onorati, Development of Fully-Automatic Parallel Algorithms for Mesh Handling in the OpenFOAM-2.2.x Technology, SAE Technical Paper 2013-24-0027.
[33] A. Montorfano, F. Piscaglia, A. Onorati, An Extension of the Dynamic Mesh Handling with Topological Changes for LES of ICE in OpenFOAM, SAE Technical Paper 2015-01-0384.
[34] F. Piscaglia, *Developments in Transient Modeling, Moving Mesh, Turbulence and Multiphase Methodologies in OpenFOAM*, 2016, Keynote Lecture at The 4th Annual OpenFOAM User Conference 2016 [link].
[35] F. Piscaglia, A. Montorfano, A. Onorati, A Compressible Dynamic Solver for the Simulation of Turbulent Flows in IC Engine Geometries, in: Int. Multidimensional Engine Modeling User's Group Meeting At the SAE Congress, 2015.
[36] Y. Wu, A. Montorfano, F. Piscaglia, A. Onorati, A Study of the Organized in-Cylinder Motion by a Dynamic Adaptive Scale-Resolving Turbulence Model, Flow, Turbulence and Combustion.
[37] I. Demirdžić, M. Perić, Space conservation law in finite volume calculations of fluid flow, Int J Numer Methods Fluids 8 (1988) 1037–1050.
[38] H. Jasak, Error analysis and estimation in the finite volume method with applications to fluid flows, Ph.D. thesis, Imperial College, University of London (1996).
[39] S. T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, Journal of Computational Physics 31 (3) (1979) 335 – 362. doi:https://doi.org/10.1016/0021-9991(79)90051-2.
URL http://www.sciencedirect.com/science/article/pii/0021999179900512
[40] F. Nicoud, F. Ducros, Subgrid-scale stress modelling based on the square of the velocity gradient tensor, Flow, Turbulence and Combustion 62 (1999) 183–200. doi:10.1023/A:1009995426001.
[41] Montorfano, A. and Giussani, F. and Piscaglia, F. and J. Helie, Parallel Performance of OpenFOAM in Industrial Applications, in: Project PRACE n. pa3053 - Report, 2016.

[42] A. Montorfano, F. Piscaglia, M. Schmitt, Y. M. Wright, C. E. Frouzakis, A. G. Tomboulides, K. Boulouchos, A. Onorati, Comparison of direct and large eddy simulations of the turbulent flow in a valve/piston assembly, Flow, Turbulence and Combustion 95 (2) (2015) 461–480, http://dx.doi.org/10.1007/s10494-015-9620-6. doi:10.1007/s10494-015-9620-6.

[43] F. Piscaglia, A. Montorfano, A. Onorati, A Scale Adaptive Filtering Technique for Turbulence Modeling of Unsteady Flows in IC Engines, SAE Int. J. Engines, Paper n. 2015-01-0395.

[44] F. Piscaglia, A. Montorfano, A. Onorati, Development of a Non-Reflecting Boundary Condition for Multidimensional Nonlinear Duct Acoustic Computation, Journal of Sound and Vibration 332 (4) (2013) 922–935, http://dx.doi.org/10.1016/j.jsv.2012.09.030.