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, Vol. 100, N. 3, 2018, p. 797-827
doi:10.1007/s10494-017-9881-3

Flow, Turbulence and Combustion manuscript No.
(will be inserted by the editor)

A Study of the Organized in-Cylinder Motion by a Dynamic Adaptive Scale-Resolving Turbulence Model

Y. Wu[1] · A. Montorfano[1] · F. Piscaglia[1] · A. Onorati[1]

Received: date / Accepted: date

Abstract The feasibility of using a scale-adaptive turbulence model for the simulation of the unsteady flow in Internal Combustion (IC) Engines is investigated. The original theory of the Dynamic Length-Scale Resolution Model (DLRM), which includes an adaptive rescaling procedure for the modeled turbulent length and time scales, has been trimmed and applied to the simulation of the flow field of a geometrically simplified square-piston engine working under motored conditions. The flow field exhibits a strong tumbling motion, which is a major characteristic of modern turbo-charged, direct-injection gasoline engines. Multiple consecutive cycles have been computed and turbulent statistics, including the ensemble mean and the variance of the velocity field, have been compared with both experimental measurements and simulation results predicted by URANS and conventional LES with identical numerical setups.

Keywords scale-adaptive turbulence model · DLRM · square-piston engine · LES of IC Engines · OpenFOAM · layer addition/removal

1 Introduction

The intrinsically unsteady character of turbulent flows is one of the many critical points to face when simulating Internal Combustion (IC) engines. Turbulence strongly influences air/fuel mixing and thus the overall speed of combustion, as well as heat transfer in the fluid flow. Changes in charge composition and turbulence level at the end of the intake stroke might result in very different local conditions at ignition time and this in turn affect the mixture characteristic burning speed, thus the optimal timings of spark or injection also varies among different engine cycles, which have direct impacts on engine efficiency and emission. To study the Cycle to Cycle Variability (CCV) in IC engines, accurate information on the evolution of turbulent structure in the engine flow field is required, which could be obtained with the aid of Computational Fluid Dynamics (CFD). There are many interesting strategies for unsteady turbulence modeling: Spalart [66] gave an overview and discussion about the advantages and limitations of many of these when applied to general problems. For industrial simulations of IC engines, Unsteady Reynolds-Averaged Navier-Stokes (URANS) equations are established as a standard tool: the complete turbulence behavior is enclosed within an appropriate single point closure model for turbulence, which relies on the assumption of the similarity of the turbulent spectrum, leading to only one characteristic turbulence length scale defining the entire spectrum [62,13]. Turbulent length and time scales are estimated by (ideally) grid-independent parameters and model transport equations are solved on reasonably coarse grids, that make this approach relatively cheap in terms of computational cost. Despite URANS allows addressing phase-averaged time evolution and gives reasonable approximation of the wall shear

---

✉ Yan Wu
yan.wu@polimi.it

Dr. Andrea Montorfano
andrea.montorfano@polimi.it

Prof. Federico Piscaglia
federico.piscaglia@polimi.it

[1] Politecnico di Milano, Via Lambruschini 4, I-20156, Milano, Italy

stress [64], it does not yield accurate distinctions between effects of large and small instantaneous scales. Consequently, the physics of the flows dominated by the organized, large-scale, coherent structures is sometimes not satisfactorily captured by such an approach. It is generally agreed that, if compared to the classical URANS, the LES can offer significant advantages, such as resolving more flow structures and their time-accurate evolution, thus it should provide better predictions on turbulent fluctuations and mixing conditions in complex configurations [7,54,31,61]. Examples of successful LES simulations of piston engine flows have been proposed through the years by several research groups; a comprehensive overview of the different approaches available in the literature may be found in [19] and in [64]. On the other hand, there are still many difficulties related to the application of conventional LES to IC engines and to complex engineering problems in general. Inappropriate grid and time step in conventional LES may cause inaccurate results [63,27,40,56,66,67] and it is quite difficult to properly reproduce near-wall effects, unless very fine meshes are used with consequently additional computational costs [62]. Indeed, for complex flows such as those in IC engines, the validity of wall models for conventional LES is questionable and most of the times they do not improve the quality of the results. A recent review by Rutland [64] evidences all these aspects and critically discusses the major modeling approach used for LES of engines, including combustion and spray modeling. Also, in the context of conventional LES, the methodology to be followed to define the optimal mesh resolution is an open question, as evidenced by the very different grid resolutions used by different research groups performing LES simulation of similar engine geometries [64,19,50]. Despite several attempts have been done to a-priori determine the proper mesh resolution in conventional LES [11,34,5,67,12], this still remains an unresolved aspect.

Hybrid models [23] want to be a link between URANS and LES, since they try to use the best of both worlds: they resolve part of the flow scales where possible and model all of them elsewhere. As a consequence, they try to use coarse grids to keep computational efficiency of URANS (including the capability of modeling near-wall effects with reasonably low grid resolution) and take advantage of the potential of LES to resolve large flow scales. Examples are Detached-Eddy Simulation (DES) [14,73], Partially-Averaged Navier-Stokes (PANS) [24,3,4] and scale-adaptive simulation (SAS) [17]. In [67,14,73], the ability of Detached-Eddy simulation (DES) to predict cyclic variations of in-cylinder flow has been achieved by Hasse *et al.* in [28,29] on the same geometry studied in this paper [8]. As apparent by the various literature on the subject, experience gathered from existing hybrid formulations is beginning to confirm that they can be an interesting solution to simulate in-cylinder flows.

Constructing a general hybrid method that interfaces seamlessly between URANS and LES (and vice-versa) requires additional considerations related to spatial/temporal resolution and the transfer of modeled turbulence kinetic energy into resolvable structures. The model equations themselves should be able to account for the energy exchange via a reconstruction or synthesis of an unsteady turbulent field, which is being represented, in a statistical sense, by the URANS data [5]. The hybrid model would then be in charge of transferring the appropriate fraction of turbulent energy into resolved kinetic energy. Among the different available approaches, the one followed in this work is a derivation of the one originally developed by Willems [77] and then by Speziale [69,68], Fasel [22] and Gyllenram [26]: a so-called rescaling function [77] is used to damp the stress tensor calculated by the underlying URANS model, consequently increasing the amount of the resolved portion of the turbulent kinetic energy. In [60], a variant of the rescaling function depending on the length and velocity scales and on the time-step of the integration has been proposed and applied to the $k-\omega$ SST turbulence model. In this paper, a few modifications to the original formulation of the Dynamic Length-scale Resolution Model (DLRM) [60] are proposed and reported in the theory section; model validation is performed on a simplified engine geometry [8,9], to assess the model's capability to predict the turbulence dynamics in a moving boundary problem. Additional simulations with URANS and LES have been used to evaluate model performance. The influence of mesh resolution and time-step advancement of the proposed methodology is also studied. For post-processing of the results, statistical moments of first and second order (mean velocity field or Reynolds stresses) are computed and compared with the experimental measurements.

2 Scale-adaptive Methods for Turbulence Modeling

The approach followed in this work for turbulence modeling belongs to the family of VLES methods. In contrast to LES, where the mean length scales of all unresolved turbulence are assumed proportional to the local grid spacing, VLES is usually based on statistical turbulence models where the turbulent length scale is calculated and will depend on the flow field: consequently, the rescaling procedure [77] can be formulated in a dynamic and general way. The rescaling function can be activated locally in the space-time domain depending on the ratio between an estimation of the resolved turbulent length scales and the magnitude of the modeled turbulent length scales. The essence of the approach is therefore the identification of the resolvable and non-resolvable fractions of the turbulence kinetic energy (and their respective dissipation rates) and hence the identification of the unresolved portion of the Reynolds stress-tensor, which in turn influences the flow through the effects of the sub-grid motion [77]. There are many ways of formulating a dynamical rescaling function in a scale-resolving model: in this work, the functional form is arbitrarily defined as exponential, following the approach originally developed by Willems [77] and then by Speziale [69,68] and Fasel [22]. The function multiplies the modeled Reynolds stress tensor before solving the averaged momentum equations in order to limit the dissipation effect of the turbulence model in regions where part of the flow scales can be resolved:

$$\mu_t = g^2 \cdot \mu_{t,\mathrm{URANS}} \tag{1}$$

Hence, the task reduces to that of multiplying the eddy viscosity calculated by an underlying URANS model by a rescaling (or damping) function g^2, which must be bounded between 0 and 1 in the inner domain, whereas it is forced to 1 on wall boundaries. It is important to note that the proposed method does not constitute a zonal approach, since the same set of equations is solved throughout the entire domain. The present approach is thought to simulate wall-bounded turbulent flows at high Reynolds numbers in complex geometries and to work with grid resolutions that are not sufficient for conventional LES to resolve smaller structures at the walls or in some specific regions of the inner domain [60]. This is a typical case occurring in multi-cycle simulation of piston engine flows: being both the solver and the turbulence modeling limited by CFL number, the reduction of the grid resolution is often used to allow significant time step acceleration, but then the applicability of conventional LES sgs models might be questionable [64]. Finally, the approach for turbulence modeling is strictly linked to the physics to study for the specific problem; in this sense, extensions to the code have been done to generalize the formulation of g^2 in Eq. (1) to any eddy-viscosity (compressible and incompressible) URANS model:

$$g^2 \equiv (\ell_t / L_t)^{4/3} \tag{2}$$

where ℓ_t is the minimum integral length scale that can be computed (either resolved or modeled) and L_t is the integral length scale as estimated by URANS.

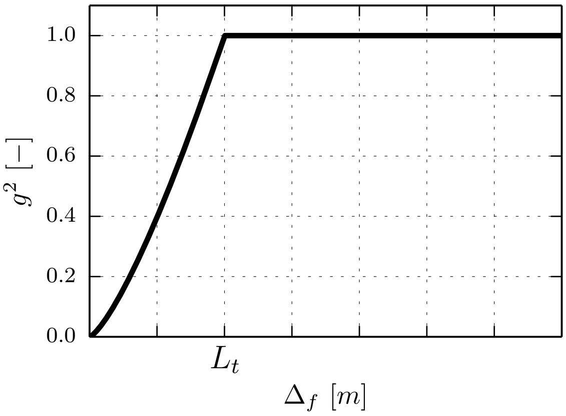


Fig. 1: The rescaling function $g^2(\Delta_f)$ is clipped to 1 as Δ_f is equal to the integral length of the modeled (URANS) scales L_t and it tends to zero in the fine grid limit.

The theory behind the definition reported in Eq. (2) is discussed in [77,26,60]: first, g^2 goes naturally to zero in the fine grid limit:

$$\lim_{\Delta_f \to 0} g^2 = 0 \tag{3}$$

meaning that the scale-resolving model tends to reduce the turbulent viscosity to zero when $g^2 \to 0$. It is important to note that, as it is apparent from Eqs. (2), (4) and (5), g^2 may be very low but never zero.

Hence the turbulent viscosity μ_t in Eq. (1) is practically never nullified and direct numerical simulation regime is recovered only in the limit of extremely high resolutions.

2.1 Formulation of the rescaling function

Scale-adaptive models differ for the formulation of the rescaling function g^2 and of the filter width $\varDelta_f$, that is the upper limit of the modeled turbulent length scale and corresponds to the lower limit of the resolved turbulent length scale. In the original proposal by Speziale [69], the stress damping was determined as a function of the Kolmogorov scale, this idea was pursued also in [2,55,26] and partially in [60], where ℓ_t in the rescaling function g^2 of Eq. (2) is defined as:

$$\ell_t \equiv \min(\varDelta_f, L_t) \tag{4}$$

In Eq. 4, $\varDelta_f$ is the estimator of the minimum resolvable length scale, determined by the time step size δt, the mesh resolution $\bar{\varDelta}$ and the local flow condition:

$$\varDelta_f = \alpha \max(\beta|\mathbf{U}|\delta t,\ \bar{\varDelta}_\ell) \tag{5}$$

Where $\beta|\mathbf{U}|\delta t$ is regarded as temporal resolution $\bar{\varDelta}_\tau$ while $\bar{\varDelta}_\ell$ is the spatial resolution. The coefficient β in the temporal resolution controls whether the limiting resolution has to be considered either the time-step or the mesh size. It can be demonstrated that the reciprocal of β, in fully Cartesian grids, corresponds to the maximum (local) CFL number above which the time discretization may be considered as the limiting factor of resolvable scales. In this work, a constant value $\beta = 5$ is used. As for the spatial resolution, $\bar{\varDelta}_\ell$ is regarded as the spatial filter used in conventional LES, in this work it is calculated as the cube root of the local cell volume. In Eq. (5), the coefficient α represents the minimum number of grid points needed to resolve a turbulent structure. As in the very initial implementation of DLRM [60], the choice of the maximum value of α=3 follows the choice of Gyllenram [26]. The calibration of α affects the operation of the rescaling function and, in turn, the performance of the hybrid model: in particular, for low values of α, the minimum length scale estimated by the the rescaling function is smaller and the turbulence model tends to resolve more scales; conversely, in authors' experience, small variations in the results of engine calculations are noticed for $\alpha > 3$. Therefore, with the aim of limiting as much as possible the model calibration, author's choice was to assume a constant value of α=3.

Compared with the original definition of $\varDelta_f$ in [60], a few modifications have been applied on Eq. 5. Firstly, for the temporal resolution, the modified DLRM in present study considers only the local condition to relax the constraint applied in [60], which is rather conservative as the maximum $|\mathbf{U}|\delta t$ in the entire mesh region, instead of the local $|\mathbf{U}|\delta t$, is regarded as the local temporal resolution. Secondly, due to the first modification, the coefficients before temporal and spatial resolution have to be considered separately, so new coefficient β is introduced for the temporal resolution. Finally, the implicit-LES enforcement based on Length Scale Resolution (LSR) [10,60] has been removed: the estimation of the Kolmogorov scale in [60] is based on the operation of the turbulence model which is in turn also dependent on local grid resolution. For this reason, applying at run-time a filter based on LSR would not be consistent and it could lead to an inconsistent clipping on the rescaling function g^2. This does not hold if LSR is included in the filter function when it is used a-posteriori.

The formulation of the rescaling function g^2 ensures that its derivative with respect to the estimated filter width $\varDelta_f$ (as long as $\varDelta_f < L_t$):

$$\frac{\partial\left(g^2\right)}{\partial \varDelta_f} = \frac{4}{3}\left(\frac{\varDelta_f}{L_t}\right)^{1/3} \tag{6}$$

tends to zero as $\varDelta_f$ tends to zero:

$$\left.\frac{\partial\left(g^2\right)}{\partial \varDelta_f}\right|_{\varDelta_f \to 0} \to 0 \tag{7}$$

This shows that the eddy viscosity asymptotically approaches a constant in the fine grid limit, as long as the model equations for turbulent kinetic energy and specific dissipation rate do not explicitly depend on the local grid spacing themselves [26]. The rescaling function g^2 limits the contribution of modeled turbulent kinetic energy and specific dissipation rate calculated by the parent URANS model and increases the amount of the resolved portions of the flow field. In Fig. 1 an example of the rescaling function g^2 is shown: the function is clipped to 1 if Δ_f is greater than the integral length of the modeled URANS scales L_t, while the second derivative of the curve near the fine grid limit must be positive, to ensure that in that region small variations of the grid size corresponds to small variations of the resolved scales. Since the hybrid model degenerates to the URANS model wherever the local resolution becomes too coarse to support LES, the best possible URANS model for the specific engineering problem studied should be chosen. There could be an obvious risk in applying a filter width that is too small in boundary layers because, if $\Delta_f < L_t$, the formulation of the eddy viscosity would not assume the proper expected behavior $\mu_t \sim n^3$ (being n the wall-normal coordinate) [76]. If the wall boundary condition for the turbulent kinetic energy $k_{wall} = 0$ were used, this would also limit the modeled turbulent length scale and should thereby inactivate the rescaling function before the wall limit is reached. Being one of the purposes of DLRM to avoid the need of extremely high near-wall resolution, the application of a zero-flux condition is specified for the turbulence kinetic energy equation while μ_t is calculated from the wall-function of the underlying URANS model. As a consequence, grid resolution at the boundary layer and the underlying URANS model adopted must be chosen accordingly [26].

3 Experimental and numerical setup

3.1 Experimental setup

The validation of the proposed approach has been carried out on the experimental setup of Boreé *et al.* [8, 9], who made comprehensive experimental studies on the generation, compression and breakdown of the tumbling motion in a model compression machine, reproducing an engine-type setup, which allows for good optical access and easier reproducibility of the boundary conditions in simulations. The compression engine has a square cylinder (100 × 100 mm) equipped with a flat head. The distance between the piston and cylinder head at Bottom Dead Center (BDC) is h=100 mm. The volumetric compression ratio r of the engine is about 4; the stroke is about 80 mm and the head clearance at Top Dead Center (TDC) around 25 mm. The actual value of r in the experiments differs from the geometric one, for a little air leakage was detected between the piston and the cylinder. Channel length is 30 times its width, and this guarantees that turbulence is well developed when the inlet flow reaches the cylinder. The piston is driven at 206 rpm by an alternative machine tool and the maximum piston velocity reaches one meter per second; a magnetic ruler fixed on the machine tool was set to accurately measure the location of the piston within the chamber. During the intake stroke, the bi-dimensional intake jet flow, tangential to the cylinder floor, is deflected by the moving piston and generates the tumbling vortex.

A guillotine device may close the intake port to allow for compression. Two operating modes were tested in the experiments and, consequently, in the simulations : a) the so-called "uncompressed-vortex" case, with the guillotine valve fixed during the piston motion and the inlet channel fully open; b) the "compressed-vortex" case, with the guillotine that opens during the exhaust and intake strokes, reproducing the operation of a four-stroke engine cycle, with turbulence that generates by disruption of the organized motion during the compression stroke and that decays during the expansion stroke. Two-dimensional PIV measurements of more than 100 consecutive cycles at eleven crank-angle positions were available [8] for the symmetry plane of the compression chamber. Due to the motion of the piston, resolutions of the 2D PIV image vary from 78 × 16 to 78 × 78 pixels, with a constant size of the PIV pixel equal to 1.2×1.2 mm^2.

3.2 Numerical treatment

The software used for the simulations is OpenFOAM®, in the development version released by the OpenFOAM Foundation [72], with necessary extensions for LES turbulence modeling [16, 57] and run-time dynamic grids [46]. Pressure-velocity coupling of the subsonic flow is solved by the `PIMPLE` (merged `PISO-SIMPLE`) algorithms [46], where convergence of pressure-velocity is enforced by iterating the p-U coupling procedure within each time-step. Description of the extensions made to the solver and examples of its application to non-conformal dynamic grids can be found in [46, 58].

3.2.1 Dynamic mesh

Fig. 3 sketches the basic dimensions of the computational domain, which consists of the cylinder, the intake channel and a large plenum. The plenum has a pipe connecting it with the atmosphere. Depending

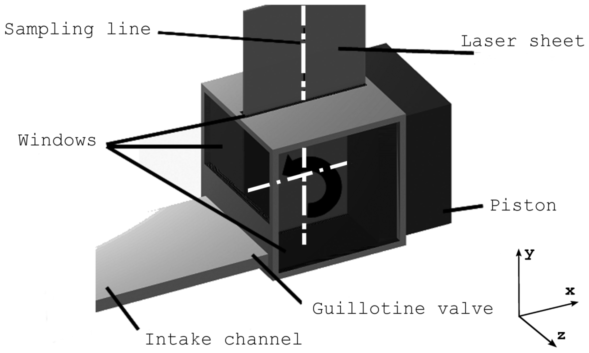


Fig. 2: Sketch of experimental setup. The plenum connected to intake channel is not shown in the figure. The laser sheet used for PIV is passing through the piston axis.

on the fluid-dynamic conditions in the inner domain, both inflow and outflow can happen at the inlet (outlet) boundary. For subsonic inflow to the boundary, the outside total pressure and static temperature are prescribed, and the direction of the velocity vector is specified as normal to the inlet. For subsonic outflow, the static pressure is set at the face-centers of the boundary and a zero-gradient boundary condition is applied on velocity. On solid walls the no-slip condition has been used, and turbulence variables in URANS and DLRM are treated by $k-\omega$ SST wall function. A constant temperature of 22.5° C was set on the piston surface, in accordance with the measurements.

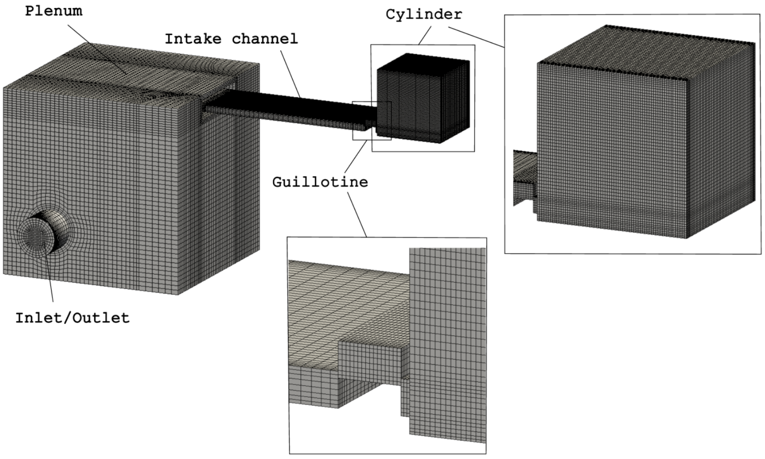


Fig. 3: Representation of the FV computational mesh used in the simulations ("moving guillotine" or "compressed-vortex" case).

A dynamic mesh handling, based on addition/removal of cell layers [46,58], is applied to the moving piston in the cylinder region. Uniform grid velocity of the moving piston boundary is also assigned to the nearby layers of cells, that act as a rigid body, while the remaining mesh points are fixed in space. Therefore only the thickness of the single cell layer between the moving and the static mesh regions has to be changed: when the layer thickness becomes too high, the deforming layer splits and new internal faces are added (layer addition). Reversely, if layer thickness is lower than a given threshold, it is merged into the moving region (layer removal). Threshold thickness values for addition/removal of layers can be either specified by the user or automatically calculated on the basis of the average cell size of the static region. As a consequence of dynamic layer addition/removal, the total cell count for the coarsest hexahedral mesh ranges between 0.2 M cells at the Top Dead Center (TDC) and 0.4 M cells at the Bottom Dead Center (BDC). The outlet plenum and in the intake channel consume about 0.12 M of cell count, as refinement is applied in the cylinder volume. Grid resolution progressively decreases from the cylinder outlet towards the open-end, as shown in Fig. 3. Dynamic layering has two major advantages over cell stretching or re-meshing: firstly, this approach allows to preserve the initial mesh quality (skewness, non-orthogonality and aspect ratio) during the whole engine cycle, since grid at different time steps differs only by the added or removed layers of fully orthogonal hexahedral cells. It is therefore possible to keep the cell size almost unchanged, and turbulence treatment could be decoupled from mesh changes [36]. Secondly, minor changes in the mesh morphology favor faster convergence of the solution in multi-block grids. The technique is computationally more efficient, because the mesh changes are triggered only locally and the global morphology of the mesh is not recalculated during the simulation. Enforcement of the conservation laws during mesh motion is detailed in [46], while the conservativeness of the dynamic layering method employed is briefly discussed in Appendix A.

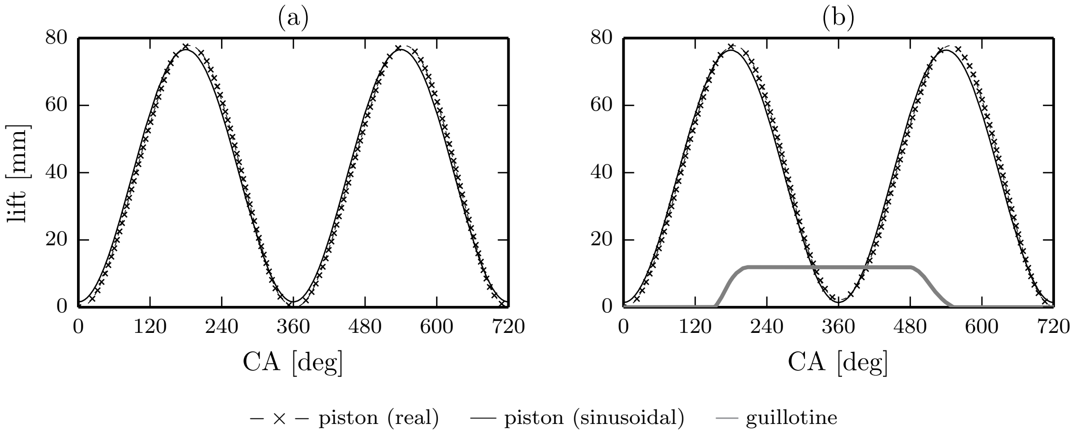


Fig. 4: Piston and guillotine lift: a) “uncompressed-vortex” mode; b) “compressed-vortex” mode. Piston lift slightly changes between the two operating modes at the end of second stroke (TDC, CA = 360 deg).

Since measurements of the piston motion [8] show small deviations from the ideal harmonic motion, prescribed motion of the piston is set by providing measurements as input data to the code. The motion laws of the guillotine valve and piston are reported in Fig. 4. The end of second stroke is at 360 crank-angle (CA) deg, where the volume of the chamber is at its minimum [8]. The motion of the guillotine (Fig. 4-b), when present, has been handled by solving a multiple domain problem based on non-conformal interfaces, where the tangential relative displacement of the guillotine and the large deformations of the piston are accounted simultaneously. The communication between multiple domains (cylinder, guillotine) has been addressed by the Arbitrary Coupled Mesh Interface (ACMI) method [72], a computationally efficient interpolation technique which is based on the local super-meshing approach [21, 20]. If compared to sliding-interface based methods [58, 46], the advantage of ACMI is its reduced computational cost, because mesh topology is not updated for each time step. Besides, ACMI does not require a constrained domain decomposition, favoring better flexibility with parallel computation, especially when applied in combination with dynamic piston layering: one full engine-cycle on the mesh including the guillotine motion (“compressed-vortex” case) could be calculated in about 4 hours by a Linux machine powered by an Intel-Xeon CPU E5-2650 (8 physical processors at 2.30 GHz). Wall-time per engine-cycle was reduced almost exactly by half in the simulations of the configuration with the fixed guillotine (“uncompressed-vortex” case). This is on one hand natural, since the frequency of the engine-cycle is half when the guillotine is present. On the other hand, this also proves the efficiency of the algorithm to handle mesh motion with non-conformal interfaces, whose presence has little influence on the overall computational cost.

3.2.2 Discretization schemes

All simulations have been carried out with a dynamic adjustment of the time-step, to comply with both the maximum CFL number allowed (as it will be discussed in the further section) and the requirement of the dynamic layering [46]. To allow a stable solution of LES on coarse grids, an implicit backward scheme has been used in all simulations for temporal discretization; this configuration is not optimal because it is still too dissipative for a well resolved LES and should be avoided if possible [48,70]. On the other hand, it could be argued that for finite-volume methods solving moving boundary problems involving topological mesh changes, full second-order accuracy in time is difficult to achieve. There are examples in the literature of other different (conceptually or numerically) approaches to manage the use of high-order temporal schemes with dynamic grids: Moureau *et.al* [49,50,74] managed to practice an explicit third order Lax-Wendroff scheme for engine simulation, using the finite-element method. An interesting solution for finite-volume method was proposed by Nguyen *et al.* [52] by treating the moving boundaries as solid particles, so that mesh motion is avoided. Although such “immersed boundary” technique is computationally more expensive, implementation of high order scheme is straightforward, and the results using explicit third order Runge-Kutta scheme are encouraging on properly refined grids.

For spatial discretization, linear interpolation from cell centers to face centers has been used for gradients integration and the Laplacian terms. Linear-Upwind Stabilized Transport (LUST) scheme is used for convection terms to achieve a stable solution. LUST blends between centered and up to 25% linear-upwind schemes with the blending factor depending on the flow direction with respect to the cell edge [75]. Non-central discretization is normally too dissipative for LES, while in engine LES some upwind-blending or gradient limiting is often accepted [64,25,6]. Moureau *et al.* [49,74] managed to achieve better discretization on engine LES through a 3rd order Lax-Wendroff scheme. A Mach-number dependent blending is also developed by Janas *et al.* [30] to minimize the dissipation effect from upwind, where blended scheme is only used in regions with Mach-numbers greater than 0.3. Nguyen *et al.* investigated the influence of numerical dissipation on the LES of engine combustion, they found that the numerical dissipation take effects mainly on the small scale features which have direct impact on flame-wrinkling and burning rates [51]. In the simulations, discretization schemes to minimize the numerical dissipation have been chosen; the aim was to preserve a consistent numerical setup as the hybrid model tends to work at low values of the rescaling function. A complete discussion about the performance of the discretization schemes used in this work applied to LES could be found in [39]. With hybrid models, the turbulence viscosity is blended and strongly depends on the function g^2, whose behavior is dependent on the mesh size and the time-step advancement, that is influenced by g^2 in turn. For this reason, it is very difficult to draw general conclusions about the level of dissipation introduced by the numerical schemes together with the hybrid turbulence model, even on canonical cases. This aspect is currently under study.

3.2.3 Convergence of flow statistics

For each numerical setup, 30 engine cycles were calculated, and flow statistics are then sampled from the third cycle on, discarding the first two cycles that are strongly influenced by the initial conditions, as seen in previous works on engine-like geometries [47]. As shown in Fig. 5, 25 engine cycles could already provide a converged description of the second moment flow statistics, in agreement with other DES studies on the same geometry [28]. In the following sections, most of the discussion is focused

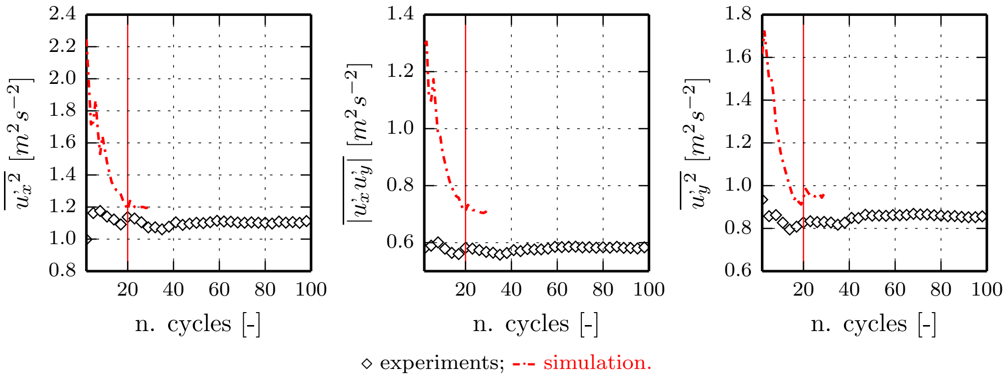


Fig. 5: Convergence of the Variance of the resolved velocity field, spatially averaged on the symmetry plane.

on the study of the intake stroke, where the local Reynolds numbers of the jet-flow are high and the operation of the turbulence model is apparent. For completeness, some crank angles from the exhaust stroke are also analyzed, despite their flow conditions quite differ from the operation of a real engine. Additional information is also given to present certain properties of the turbulence model, in order to study its sensitivity to grid resolution and to time-step size, which influence its capacity to generate turbulence. This will be attested by comparing the ensemble mean and variance (Reynolds stresses) of the resolved velocity field. In the following, u_i stands for the velocity components along direction i . The global reference frame used is reported in Fig. 2. The overline attribute will be used to represent ensemble (phase-averaged) quantities. For post-processing of the results, statistical moments of first and second order (ensemble mean and the variance of the velocity field) from LES and hybrid simulations were directly computed from the resolved fields without considering the additional contribution of unresolved scales. For LES, even though several methods are found in literature to evaluate subgrid-scale contributions [1, 45, 44], it is generally true that statistical moments are essentially related to scale ranges contained in the resolved field. Besides, uncertainties present in the techniques used for subgrid-scale contribution evaluation makes the direct comparison of resolved quantities and reference data a generally accepted procedure [65]. For scale-resolving simulations, neglecting the additional contribution of unresolved scales may instead lead to a significant error, in particular in those regions where most of the turbulence scales are modeled. On the other hand, the method followed for post-processing is consistent with the scope of the work, which is focused on the capability of hybrid methods to resolve (rather than model) turbulence unsteadiness on reasonably coarse grids. It should also be noted that results from URANS are not phase-averaged, following one of the common practices of the engine community [71]: three URANS cycles are calculated and only results from the last one are used for comparisons. Comparisons between simulations and experiments are shown along the vertical and horizontal middle lines on the symmetry x-y plane.

4 Results

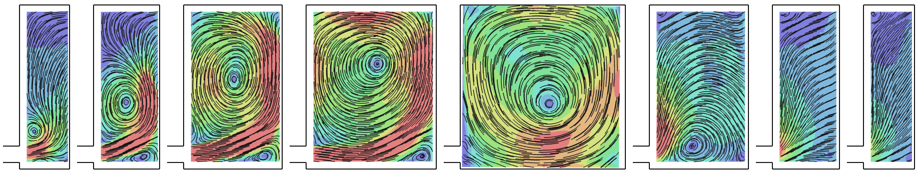
A block-structured grid labeled as “coarse” is considered as baseline case (total cell count 0.2 M $\sim$ 0.4 M, typical cell size $\approx$ 2 mm), where the maximum CFL number is limited to be lower than 1 (0.9 to be exact). Simulation results of DLRM on the baseline grid are compared with experimental measurement at different crank angles [8, 9], along with results calculated by conventional LES and URANS. In particular, the σ sgs model [53, 54] has been used for conventional LES. The σ model is based on the singular values of the velocity gradient tensor, which naturally vanishes at the wall, and it is expected to be one of the best performing models when applied to LES of (IC) Engines [7, 35, 43]. Validation and implementation of the σ model on OpenFOAM® can be found in [39]. As for URANS model, standard k-ω SST model is used, based on the formulation given by Menter [42] with updated coefficients from [41].

For the “uncompressed-vortex” case, a second block-structured grid labeled as “fine” has been used to test the influence of grid resolution on the model behavior. Cells count of the “fine” grid varies from 1 M at the TDC to 1.8 M at the BTC, with typical cell size $\approx$ 1 mm. Boundary layers in the compression chamber are also refined in the wall normal direction: y^+ is kept lower than 3, while axial and tangential resolutions are $\Delta z^+ < 20$ and $\Delta x^+ < 20$. Influence of the time-step size on the behavior of DLRM is investigated by changing the maximum CFL number allowed during simulation. Despite most of the analysis is focused on the study of the “uncompressed-vortex” mode,results from simulations with the moving guillotine are reported in the final section, as additional test with more complex mesh motion strategy.

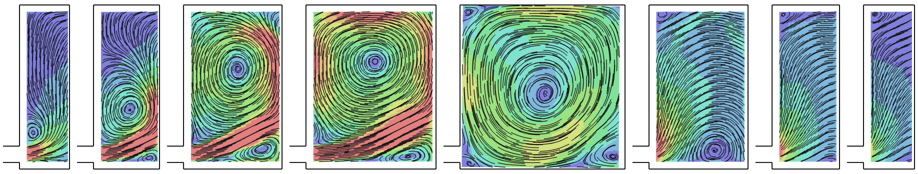
Before the comparison and analysis of the results, it must be noted that the major goals of the proposed approach are to analyze the evolution of the tumble vortex in a engine-like geometry and to evaluate the potential of the hybrid model to resolve large-scale flow structures with low computation cost. Hence the computational grids used are not sufficiently refined at all for LES, which in general requires much higher spatial resolution for engine flow simulation [30, 47, 18, 52]. Although the grid resolutions used in this work are similar to those in the literature simulating the same case [50, 28], we assume that the “coarse grid” is just sufficient for URANS modeling, while the “fine grid” is merely adequate for LES. Such a coarse grid for the baseline setup is only chosen to make the study possible.

4.1 Uncompressed-vortex case

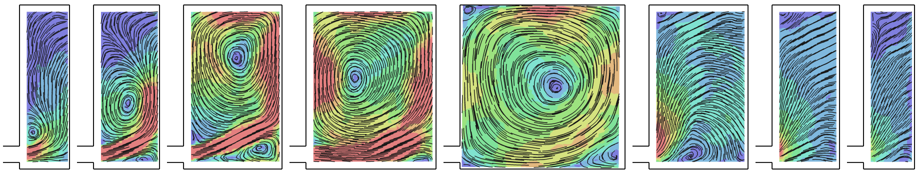
Fig. 6 shows a sequence of snapshots of the trajectories and the contour plot of the phase-averaged velocity magnitude on the x-y symmetry plane, with the development of two counter-rotating vortexes due to the roll-up of the upper and lower mixing layers of the jet. These voices are strongly influenced by the piston, which deflects both of them. The upper shear layer undergoes periodic eddy-detachment, and develops along the surface of the piston which generates the large-scale tumble. The performance of the different turbulence models is apparent in Fig. 6. During the first part of the intake stroke (angles 34, 56) all approaches lead to similar results. As jet velocity increases (CA = 89 deg) some differences appear: the size of corner vortex between jet and piston, which is very clearly visible in the experiments, is slightly overestimated by both LES and DLRM, while it is barely noticeable in the URANS results. On the other hand, the location of the main vortex core is well captured by URANS and DLRM, but it is considerably offset towards the cylinder head in the LES at 121 CA-deg. Close to Bottom Dead Center (CA = 185 deg) results are similar for all approaches, being the greater difference the size of the already mentioned corner vortex, which is anyway not visible in the experiments. Along the exhaust stroke (CA 276-327 deg) the flow configuration changes significantly: the large tumble vortex is replaced by a small recirculation region between stream flowing out the cylinder and the lower wall. In this phase DLRM behaves similarly to URANS, even though the latter produces a more disorganized flow in the upper part of the cylinder. The size of the vortex is in a greater accordance with experiments than LES that, on the other hand, better predicts the vortex location.



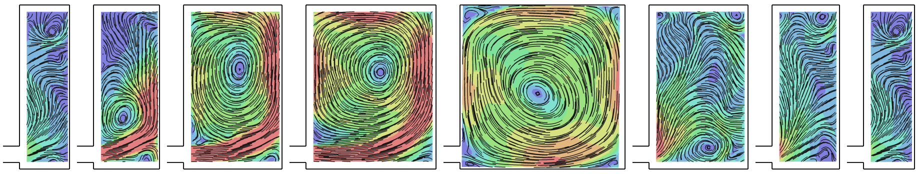
(a) Experiments



(b) DLRM



(c) LES



(d) URANS

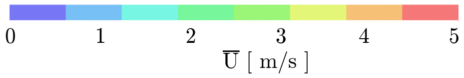
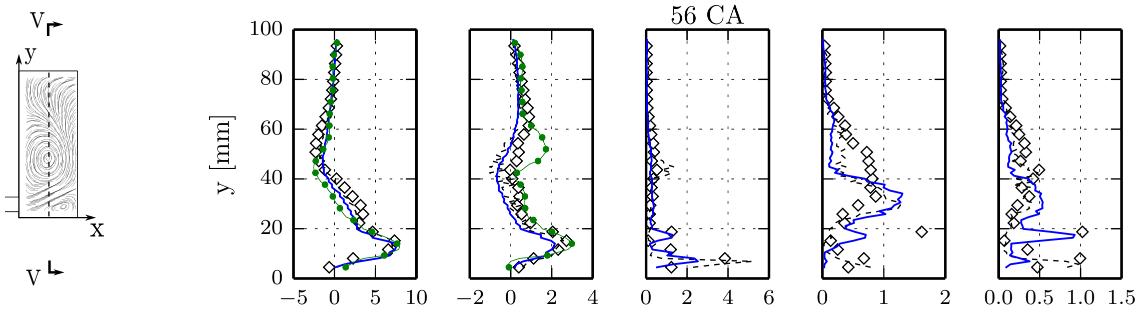
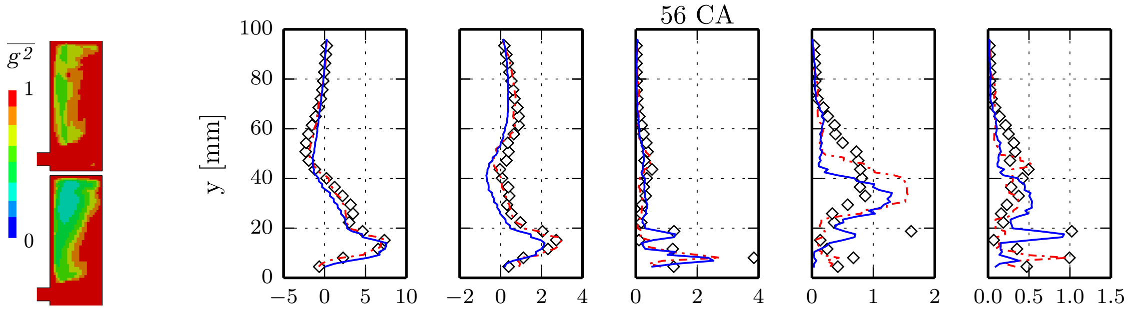


Fig. 6: Evolution of the ensemble mean flow on x-y plane over the engine cycle at (left to right): 34, 56, 89, 121, 185, 276, 306, 327 CA-deg. a) Experiments (phase-averaging over 100 PIV realizations); b) “baseline” simulation, DLRM, CFL<1; c) conventional LES with σ-sgs; d) URANS.



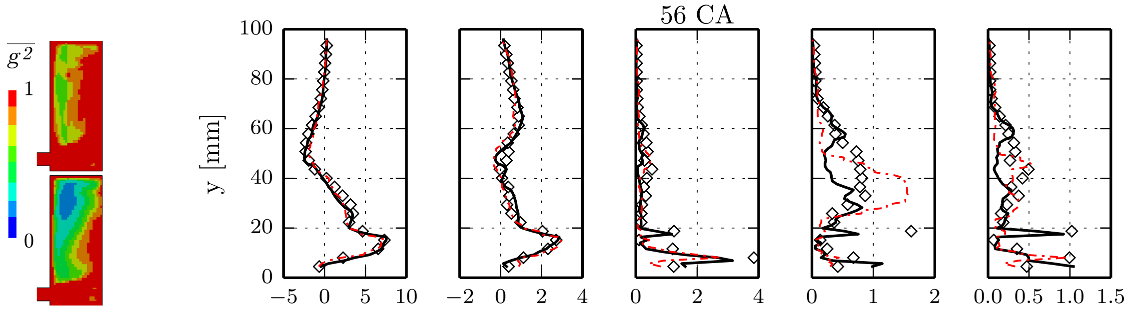
(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g) top:CFL<5 bot:CFL<1 (h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

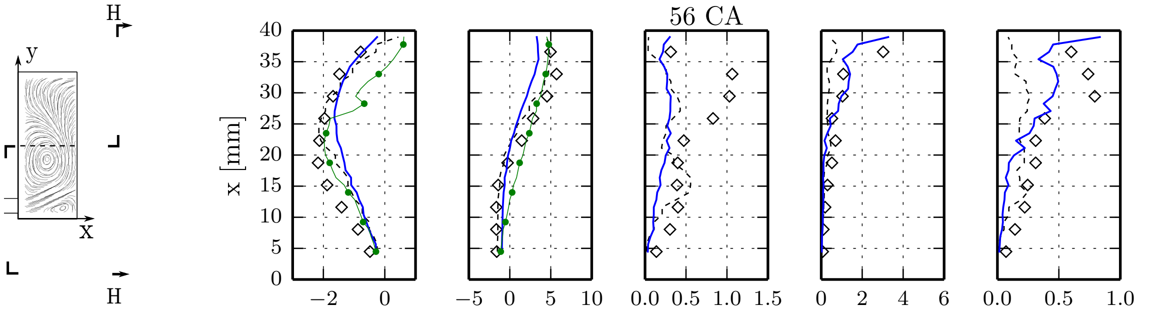
Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5



(m) top:coarse bot:fine (n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

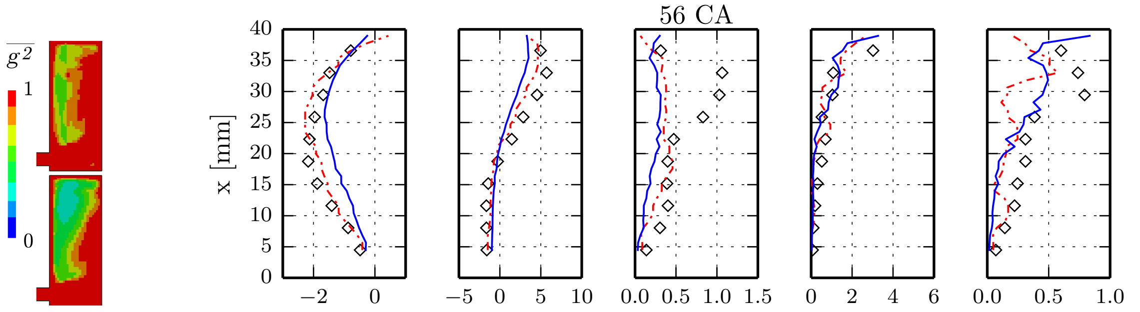
Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 7: Comparison of the phase-average and variance of the velocity field at 56 CA-deg, calculated by different turbulence models on a centered *vertical* line over the mid-cross x-y measurement plane.



(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

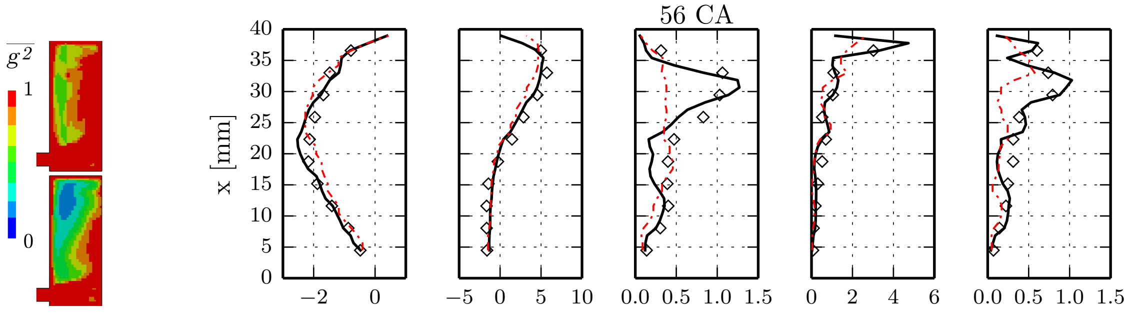
Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g)
top:CFL<5
bot:CFL<1

(h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5

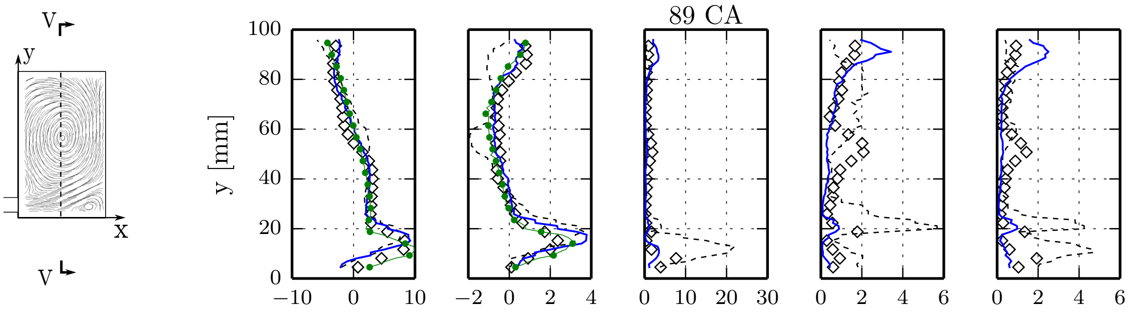


(m)
top:coarse
bot:fine

(n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

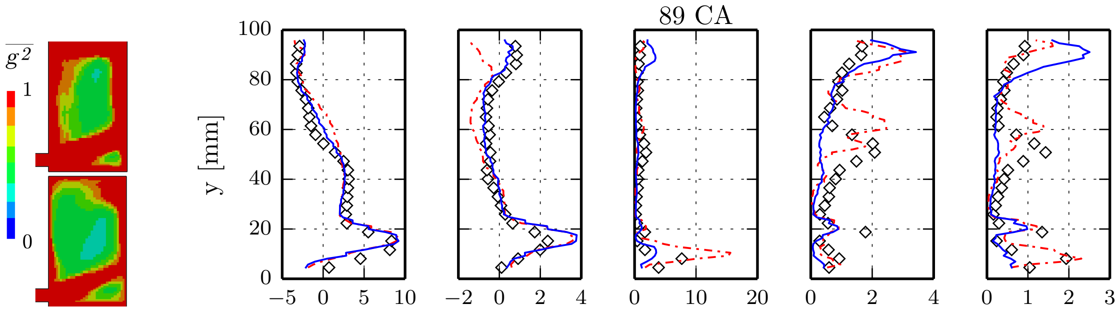
Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 8: Comparison of the phase-average and variance of the velocity field at 56 CA-deg, calculated by different turbulence models on a centered *horizontal* line over the mid-cross x-y measurement plane.



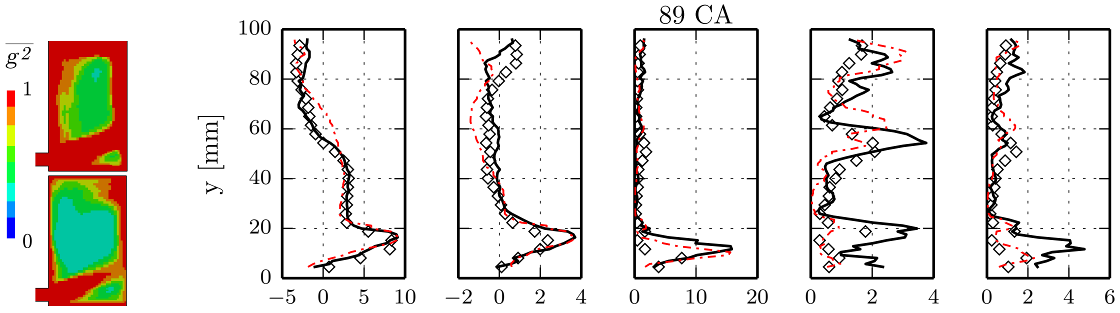
(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g) top:CFL<5 bot:CFL<1 (h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

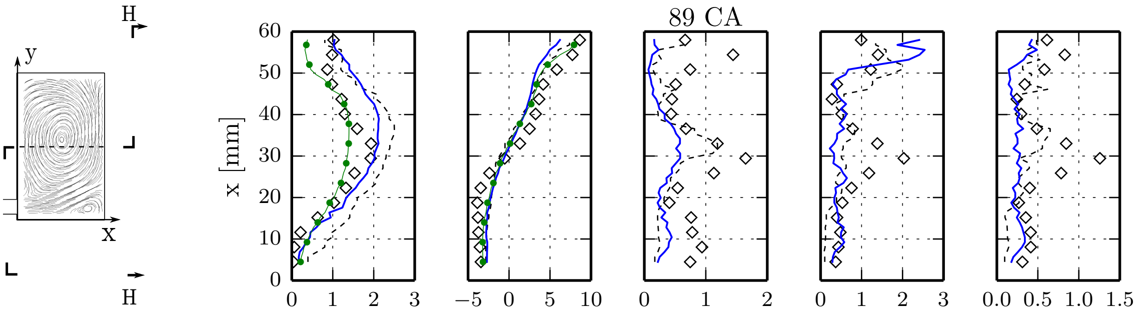
Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5



(m) top:coarse bot:fine (n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

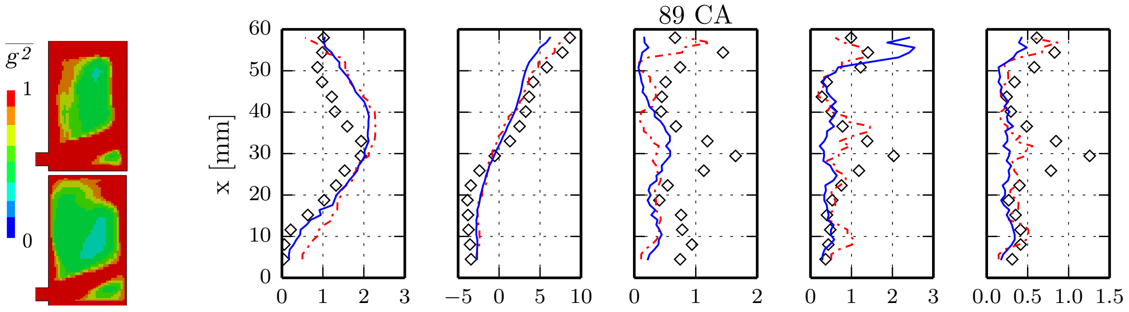
Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 9: Comparison of the phase-average and variance of the velocity field at 89 CA-deg, calculated by different turbulence models on a centered *vertical* line over the mid-cross x-y measurement plane.



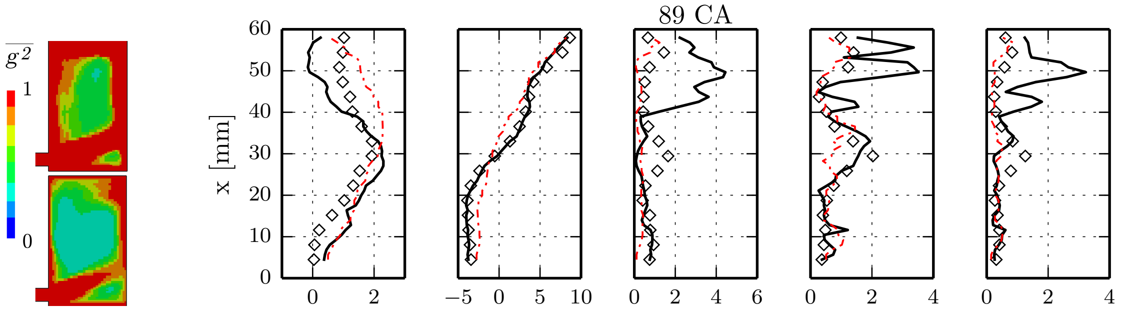
(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g) top:CFL<5 bot:CFL<1 (h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

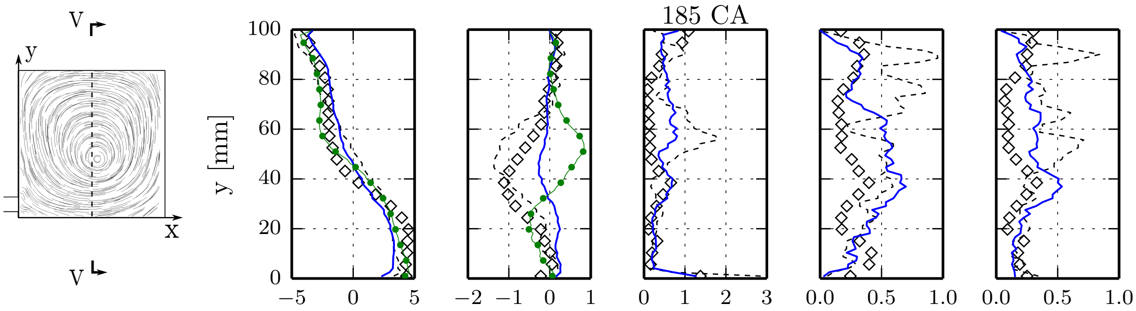
Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5



(m) top:coarse bot:fine (n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

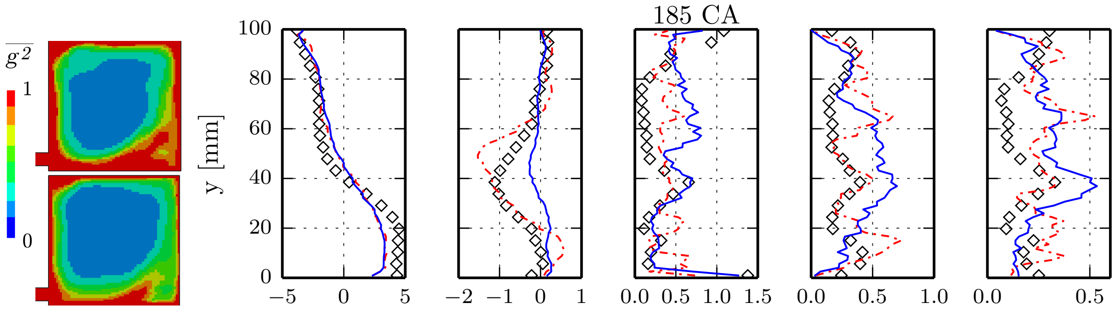
Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 10: Comparison of the phase-average and variance of the velocity field at 89 CA-deg, calculated by different turbulence models on a centered *horizontal* line over the mid-cross x-y measurement plane.



(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

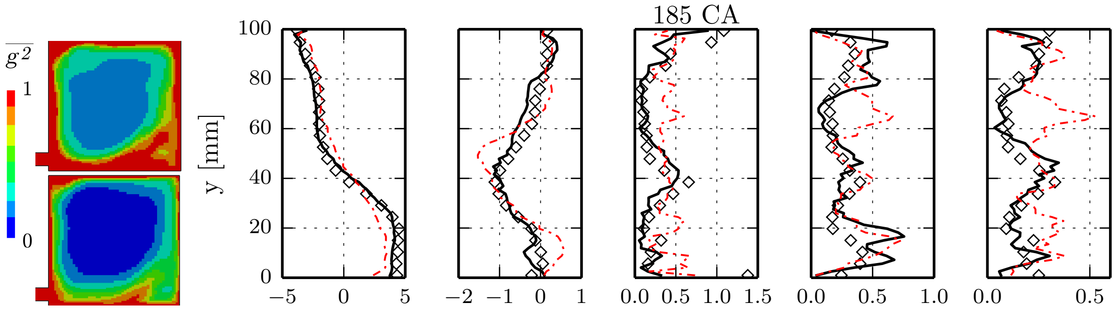
Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g)
top:CFL<5
bot:CFL<1

(h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5

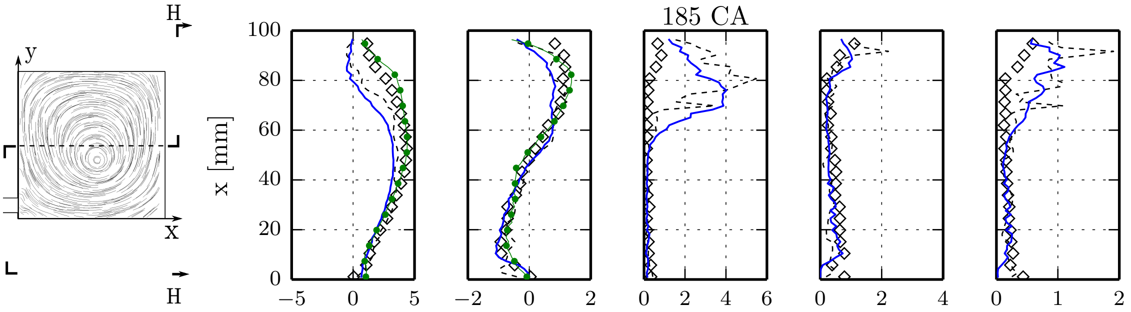


(m)
top:coarse
bot:fine

(n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

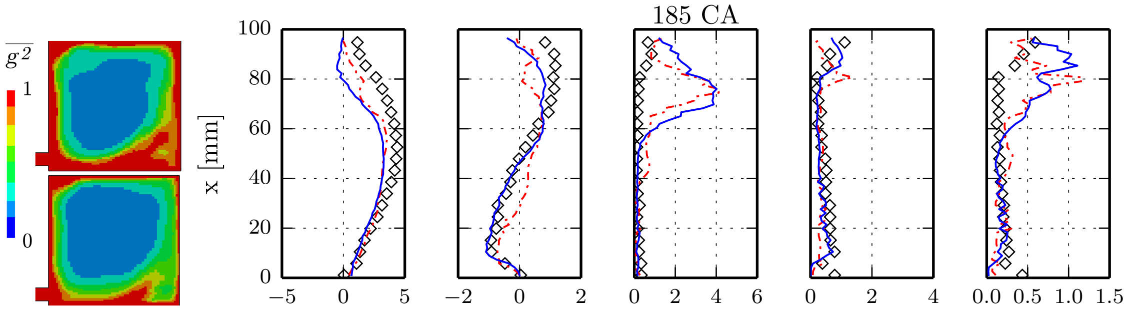
Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 11: Comparison of the phase-average and variance of the velocity field at 185 CA-deg, calculated by different turbulence models on a centered *vertical* line over the mid-cross x-y measurement plane.



(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

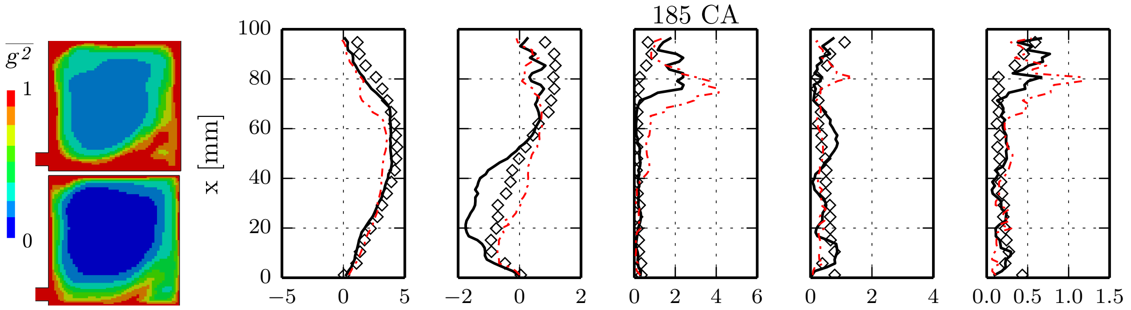
Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g)
top:CFL<5
bot:CFL<1

(h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5

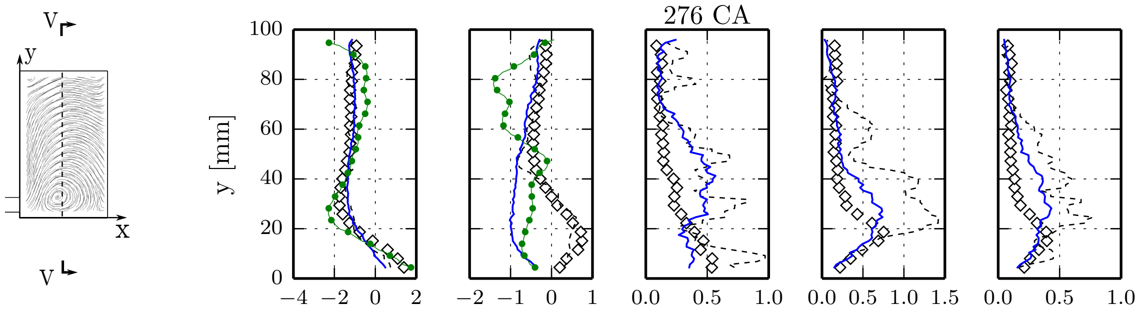


(m)
top:coarse
bot:fine

(n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

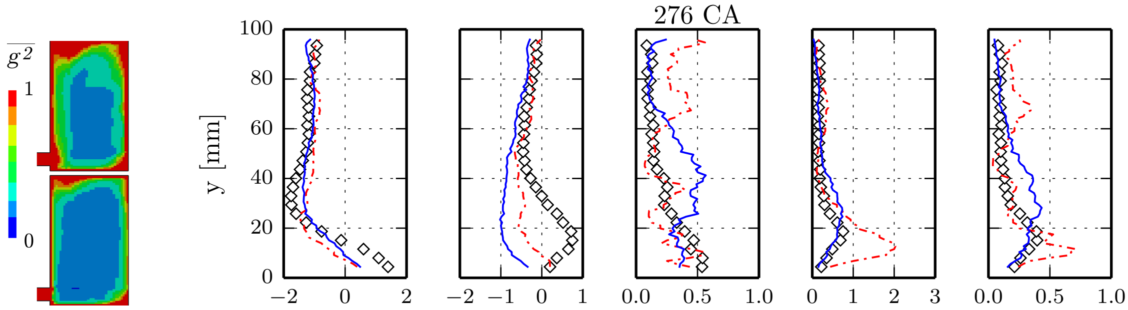
Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 12: Comparison of the phase-average and variance of the velocity field at 185 CA-deg, calculated by different turbulence models on a centered *horizontal* line over the mid-cross x-y measurement plane.



(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

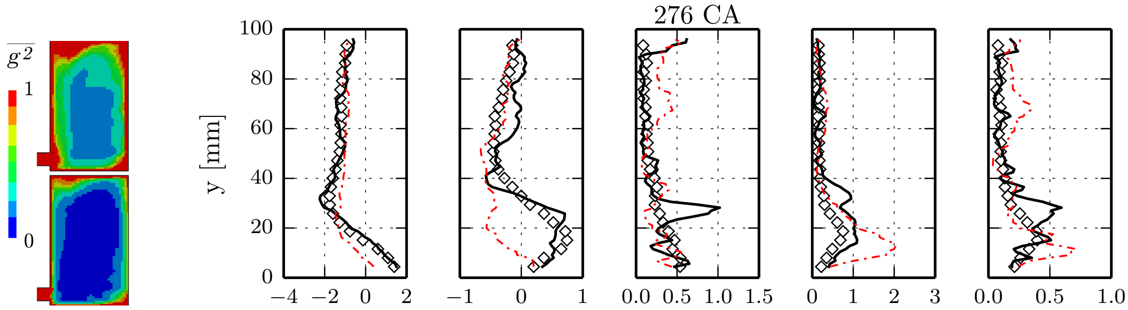
Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g)
top:CFL<5
bot:CFL<1

(h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5

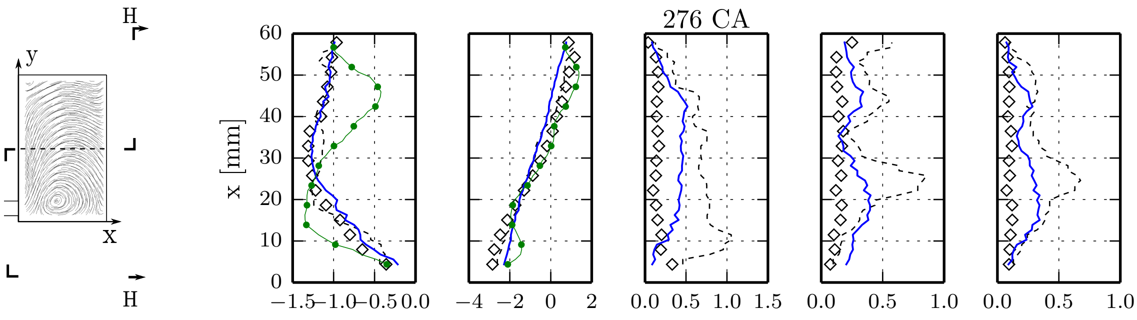


(m)
top:coarse
bot:fine

(n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

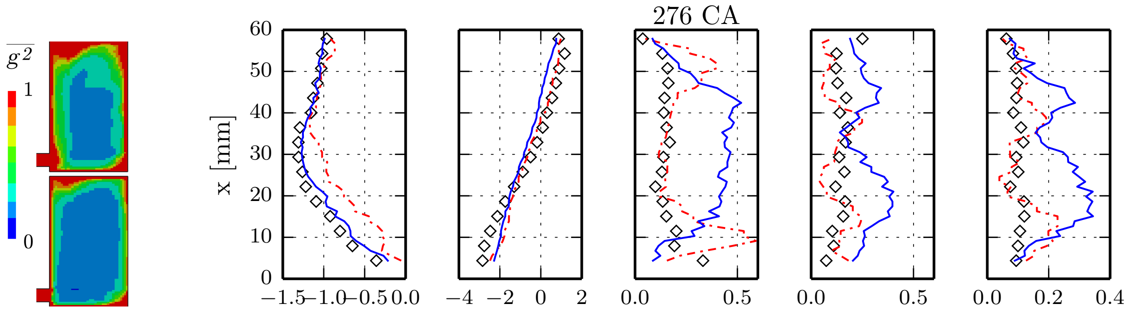
Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 13: Comparison of the phase-average and variance of the velocity field at 276 CA-deg, calculated by different turbulence models on a centered *vertical* line over the mid-cross x-y measurement plane.



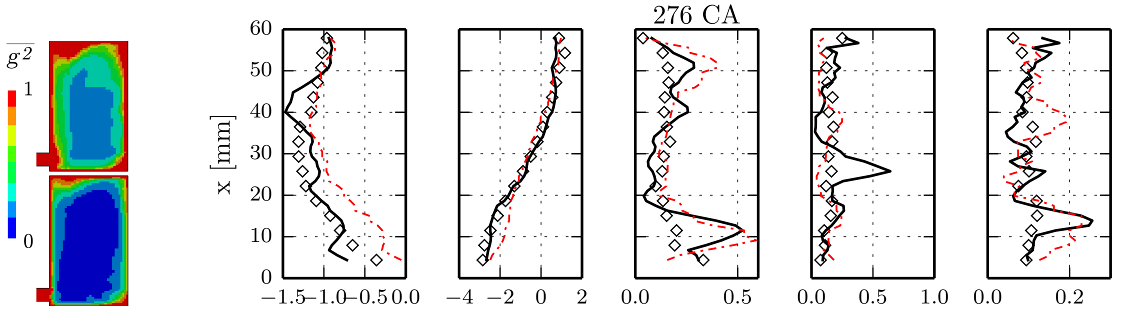
(a) sample line (b) $\overline{U}_x$ (c) $\overline{U}_y$ (d) $\overline{u'_{xx}}$ (e) $\overline{u'_{yy}}$ (f) $\overline{u'_{xy}}$

Comparison of different turbulence models (coarse grid, CFL<1): ◇ Exp; — DLRM; - - - LES; ● ● k-ω SST



(g) top:CFL<5 bot:CFL<1 (h) $\overline{U}_x$ (i) $\overline{U}_y$ (j) $\overline{u'_{xx}}$ (k) $\overline{u'_{yy}}$ (l) $\overline{u'_{xy}}$

Influence of time step advancement (DLRM, coarse grid): — CFL<1; -·- CFL<5



(m) top:coarse bot:fine (n) $\overline{U}_x$ (o) $\overline{U}_y$ (p) $\overline{u'_{xx}}$ (q) $\overline{u'_{yy}}$ (r) $\overline{u'_{xy}}$

Influence of mesh resolution (DLRM, CFL<5): -·- coarse grid; — fine grid

Fig. 14: Comparison of the phase-average and variance of the velocity field at 276 CA-deg, calculated by different turbulence models on a centered *horizontal* line over the mid-cross x-y measurement plane.

More quantitative comparison of the flow behavior can be found in Figs. 7 to 14, where the unsteady flow fields from both experiment and simulation are sampled over the horizontal and vertical sample lines intersecting the center of the x-y plane of the volume chamber, at CA = 56, 89, 185 and 276 CA deg.

The first rows (Sub-figures (a) to (f)) of Figs. 7 to 14 show the different performance of different turbulence models tested with baseline numerical setups. On such coarse mesh DRLM (blue line) naturally works in URANS mode in the near wall regions and where the flow velocity is progressively higher: the phase-averaged g^2 (sub-figures (g)) reaches to its maximum value and the turbulence viscosity of URANS is fully recovered. In the recirculating regions g^2 tends to lower values so that DLRM works in hybrid mode. Minimum values of g^2 are typically found in the middle of compression chamber, where large scale tumble motion dominates.

For the prediction of the phase-averaged flow (Figs. 7 to 14-b and -c) URANS gives good results except for a few crank angles(e.g. CA = 56, 185 and 276 CA-deg in Figs. 7-b, 8-b, 11-c,13-c, 14-b), where DLRM provides similar accuracy as conventional LES. Exceptions could be found at CA = 185 deg (Fig. 11-c) and CA = 276 deg (Fig. 13-c), where DLRM is strongly influenced by the parent URANS model which possibly provide biased turbulence statistics. Increasing grid resolution is the most straightforward way to overcome such influence, as showed in the last row (Sub-figures (m) to (r)) of Figs. 7 to 14, since smaller cell size not only improves the spatial resolution, but also demands for higher temporal resolution by limiting the maximum CFL number allowed.

Along with the phase-average flow, components of the resolved Reynolds Stress tensor (variance of the velocity field) are also displayed in Figs. 7 to 14, to evaluate the capability of the turbulence models to resolve flow fluctuations. The proportion of resolved (rather than modeled) Reynolds stress in DLRM depends directly on the value of the rescaling function g^2, while variance of the velocity field provided by LES is computed using all scales down to the cutoff length corresponding to the filter size, which depends on the mesh resolution only. As expected, prediction of $\overline{u'_{xx}}$, $\overline{u'_{yy}}$, $\overline{u'_{xy}}$ by DLRM are comparable to those given by conventional LES in regions with an average-low value of g^2, like in Figs. 7 and 8, sub-figures (d) to (f). This region corresponds to the large tumble vortex, whose most energetic length scale is sufficiently large. On the other hand, in correspondence of the jet stream from/to the intake duct, DLRM operates in URANS mode ($g^2 \approx 1$), so the fluctuating part of the velocity is almost entirely modeled and it significantly differs from conventional LES, as shown in Figs. 9, 11 and 13, subplots (d)-(f). It is very important to underline, however, that the performance of conventional LES in this comparison is a direct consequence of the unfairly coarse grid resolution and dissipative numerical scheme adopted, which makes sense only for URANS and potentially also hybrid models. The coarse grid and low order schemes are only chosen to make the study possible, as one of the major topics of this work is to evaluate the resolving capability of DLRM for engine flow with sufficiently low CPU budget. For the situation where computational cost is less of a concern, it is expected that conventional LES would improve greatly its performance with finer grids.

The influence of the time-step advancement on the operation of the g^2 through Eq. (5) is shown in plots (g) to (l) of Figs. 7-14. On the same "coarse" grid of the baseline setup, simulations having different maximum CFL numbers allowed ($\mathrm{CFL}_{\mathrm{max}} = 1$ and $\mathrm{CFL}_{\mathrm{max}} = 5$, respectively) are performed. As already mentioned, g^2 serves as an indicator of the amount of fluctuation that is considered being resolved rather than modeled by DLRM: higher values of g^2 indicate that more turbulence scales are modeled. Since value of g^2 is based on the integral length scale estimated by URANS model, the choice of the parent URANS model has a severe impact on the quality of the results as well, especially in current case where DLRM always works under URANS mode at the walls and in high-speed flow regions. As expected, the g^2 function differs in the two simulations, since g^2 function works as a low-pass filter for the flow scales that cannot be locally resolved. However, it is also to be noted that discrepancy in the results for different time-steps is in general not very high. This is due to the fact that there is a large difference between the velocity of the jet flow and in other regions. So the limit on the maximum CFL number has an impact almost only in the cells near the jet flow, rather than the entire flow domain. Since in both cases the jet is modeled by URANS (see contour plots (m), Fig. 7-14), the influence of CFL number on the results is limited.

Unlike using smaller time steps, sub-figures (m)-(r) in Fig. 7-14, suggest that a direct increase of the grid resolution brings more concrete improvement on the simulation results, especially in terms of prediction of the Reynolds stresses. DLRM resolves more turbulent scales (the value of g^2 decreases over the computational domain) with the "fine grid", which provides not only higher spatial resolution, but also better temporal resolution by limiting the CFL number. In particular, at CA = 56 deg, there is a noticeable improvement in the prediction of $\overline{u}_{yy}$ with the finer grid (Figs. 7-q and 8-q); the same applies to CA = 89 deg (Figs. 9-q and CA = 185 deg (Figs. 11-q and 12-q). It should also be noted that flow unsteadiness is sometimes overestimated on refined grid, especially in the near wall regions of the moving piston (see Fig. 10-(p-r), for example). As the flow simulation is already in convergence (situation improves little if more than 15 engine cycles are included in the calculation), the reason remains unclear.

One possible explanation could be found in the near-wall treatment: DLRM directly adopts the wall-functions of the parent URANS model and the instantaneous turbulence variables transported to the fine grid could be way too different compared with an ensemble-averaged prediction.

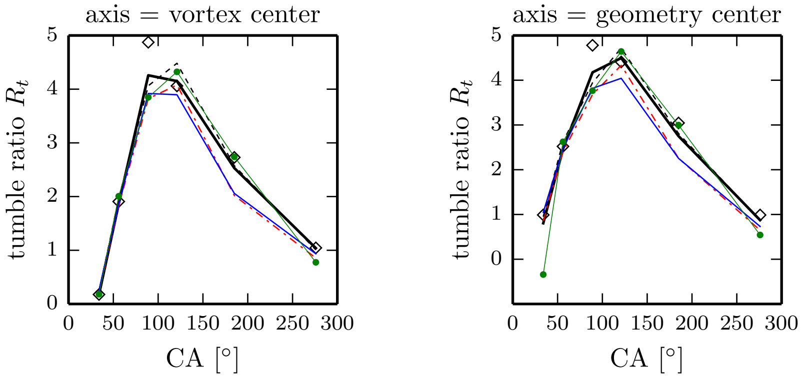
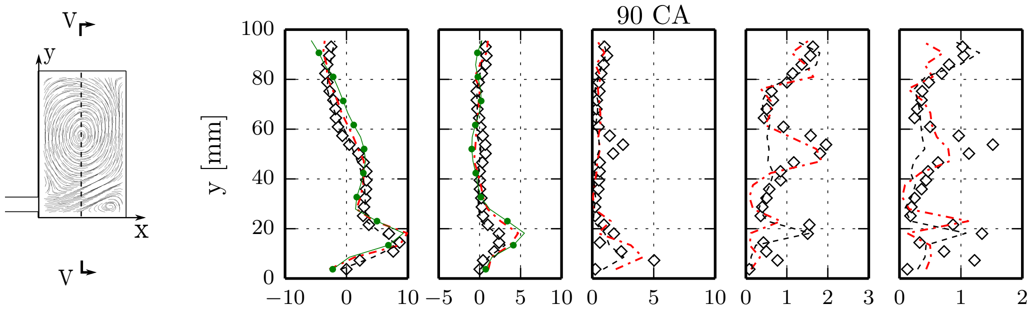


Fig. 15: Mean tumble ratio of the uncompressed-vortex case: ◇ Exp; — DLRM (coarse mesh, CFL≤1); -·- DLRM (coarse mesh, CFL≤5); — DLRM (fine mesh, CFL≤5); - - - LES; ● ● k-ω SST.

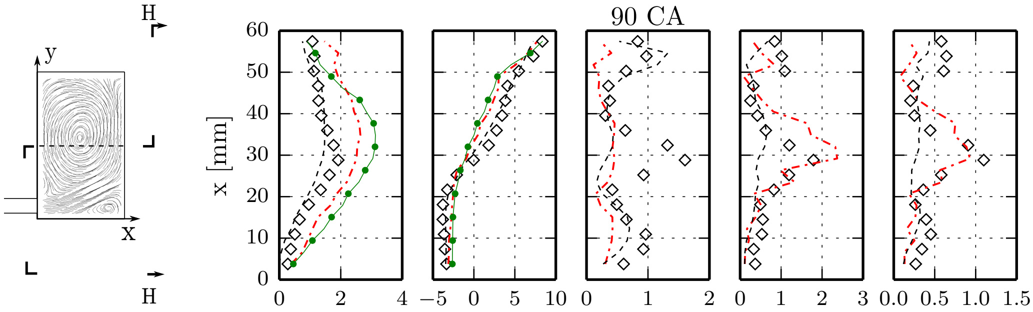
Finally, comparisons on the mean tumble ratio (TR) are presented in Fig. 15. The tumble ratio is defined as the ratio between the mean angular velocity of the two-dimensional flow measured and the angular velocity of the crankshaft [38]. The mean angular velocity is calculated individually for each CA, which includes 28 realizations from different cycles. For each phase CA, firstly all simulation results were mapped onto the 2D grid of the PIV measurement; then, the angular velocity of each grid cell was calculated with respect to the reference rotation axes and all values were averaged over the cylinder domain. Finally, the mean angular velocity was computed as the arithmetic mean of the domain-averaged angular velocity over all the 28 realizations. Two sets of TR were calculated with different choices of the reference rotation axes [33]: the TR calculated using the center of the large scale vortex as reference axis (Fig. 15, left) slightly differs from the calculation performed by assuming the geometric center of the x-y plane as the reference axis (Fig. 15, right). In both cases, predictions of DLRM show good agreement with the experiments, even with a low grid resolution.

4.2 Compressed-vortex case

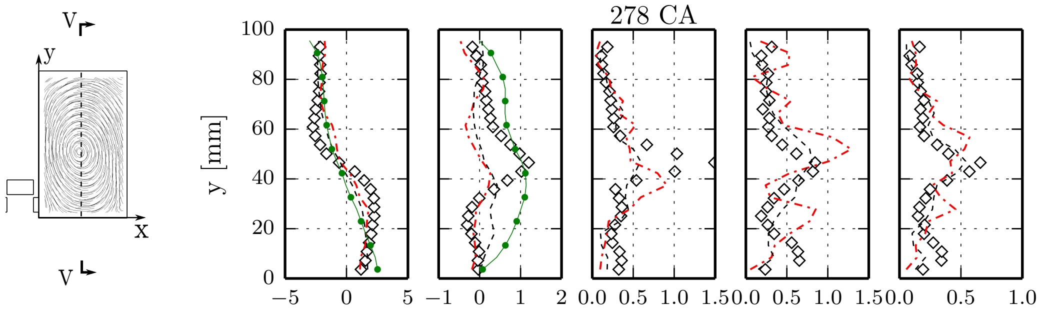
The second set of simulations includes the guillotine motion (Fig. 4) in the calculation and it is more representative of the operation of a four-stroke engine [8]. Also the mesh motion algorithm required to move the piston and the guillotine valve together is closer to the need of real world engine simulations [46]. The guillotine is described by a region mesh, which is dynamically connected and disconnected with the rest of the domain by non-conformal interfaces (as described in Sec. 3), and moves according to a prescribed motion [58,46,59]. The dynamic non-conformal interfaces allow to properly model the flow-blockage of the guillotine when it is closed. For the whole engine cycle, the adopted strategy requires only one mesh being defined, which would then be automatically recalculated during flow simulation. Similarly to the uncompressed-vortex case, 30 consecutive engine cycles are simulated and the sampling of the flow quantities begins from the third cycle. The grid previously referred as “coarse” is used for the simulations described in this section. In Fig. 16 the comparison on two plotting lines over the x-y plane at two different angles are reported: 90 CA-deg, corresponding to the middle of the intake stroke when the jet flow from the inlet channel is almost at its highest velocity and 278 CA-deg, when the recirculating vortex is compressed. Despite the coarse grid being used, some details of the flow features seem to be properly reproduced, both in terms of mean values and variance.



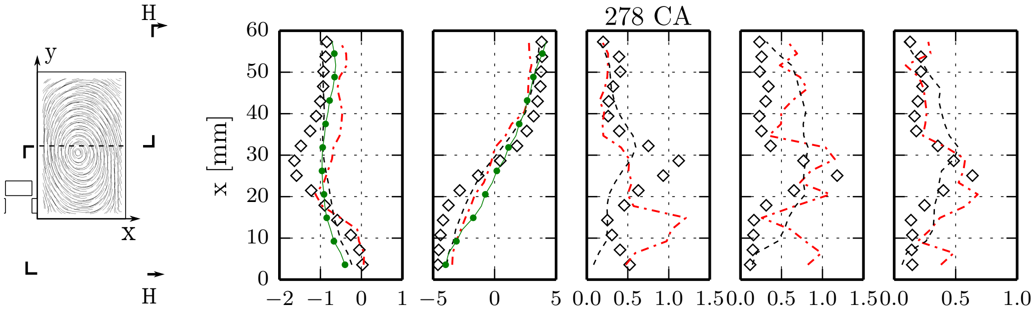
(a) $\overline{U}_x$ (b) $\overline{U}_y$ (c) $\overline{u'_{xx}}$ (d) $\overline{u'_{xy}}$ (e) $\overline{u'_{xy}}$



(f) $\overline{U}_x$ (g) $\overline{U}_y$ (h) $\overline{u'_{xx}}$ (i) $\overline{u'_{xy}}$ (j) $\overline{u'_{xy}}$



(k) $\overline{U}_x$ (l) $\overline{U}_y$ (m) $\overline{u'_{xx}}$ (n) $\overline{u'_{xy}}$ (o) $\overline{u'_{xy}}$



(p) $\overline{U}_x$ (q) $\overline{U}_y$ (r) $\overline{u'_{xx}}$ (s) $\overline{u'_{xy}}$ (t) $\overline{u'_{xy}}$

Fig. 16: Phase-averaged velocity field including variance on the *vertical* and *horizontal* lines centered over the mid-cross x-y measurement plane. ◇ Exp; -·- DLRM - - - LES; ● ● k-ω SST.

5 Conclusions

A modified implementation of the rescaling function in the DLRM turbulence model has been validated on the simulation of the compressible flow in a engine-like moving geometry at two different operating conditions. In the proposed model, the rescaling function smoothly controls the transition from cells where turbulence is modeled (URANS) to cells where turbulent scales are partially resolved. The formulation of the function used by DLRM allows to avoid discontinuities in the turbulent viscosity between neighboring cells, in particular in the regions where transition between different formulations of the turbulent scales is present. Since DLRM degenerates to its parent URANS model wherever the local resolution becomes too low to resolve the turbulent scales of interest, the described approach can also be seen as a criterion for resolving (or modeling) turbulence in a computational cell at certain flow conditions. Being one of the purposes of the scale-adaptive approach to avoid the need of extremely high near-wall resolution, the application of the wall-functions of the underlying URANS model are always forced at the walls.

The computational cost of the model and the case setup, in terms of boundary and initial conditions required, are the same of the underlying URANS model. If compared with conventional LES performed on the same grid, DLRM is computationally more expensive, since it is always applied to an underlying two-equation URANS model. On the other hand, DLRM is designed to work on coarser grids with respect to the ones commonly used in conventional LES, so it should allow for the reduction of the grid resolution and time-step acceleration.

Validation of the proposed numerical methods are performed on the flow field in a simplified engine geometry, with a baseline grid which is unfairly coarse in the context of conventional LES. It is shown that with such “URANS like” grid resolution, DLRM could provide reasonably good predictions on both first and second moment of the flow statistics. Clearly, the performance of the conventional LES would improve greatly if properly refined grids and high-order numerical schemes are adopted, which could possibly also boost the quality of the URANS and hybrid results. In this sense, it is expected that DLRM serves potentially as a link between URANS and LES, with compromised computational efficiency and resolving capacity, rather than any replacement to either of them.

Extension of the validation on more realistic engine geometry in the future work is still necessary though. The flow in a real engine is characterized by high level of velocities, non-homogeneous boundary layers, turbulence generated by the jet separation from the valve and recirculating flows that are sensitive to small variations. The prediction of a vortex impingement and propagation along the wall cannot be properly handled with a URANS model while in LES it is really sensitive to the mesh size and sub-grid scale turbulent viscosity. More detailed study of the flow behavior and its complex interactions with the walls would also be helpful to examine the potential and limitations of DLRM. Development of more appropriate near wall treatments for DLRM would be particularly important, as the current hybrid model completely inherits both the advantages and the deficiencies of the wall function from its parent URANS model.

Acknowledgements Authors would like to kindly thank Prof. Jacques Borée from ISAE-ENSMA (Institut Supérieure de l'Aérotechniqueet de l'Espace, France) for having provided the detailed set of experimental data for model validation and the Laboratory Computing Resource Center, Argonne National Laboratory (Illinois, USA), for making available the computing resources through the HPC cluster Blues under the `PETSc-FOAM` project.

Conflict of Interest: The authors declare that they have no conflict of interest.

References

1. Agee, E., Gluhovsky, A.: LES model sensitivities to domains, grids and large-eddy timescales. Journal of the Atmospheric Sciences **56**(4), 599–604 (1999). URL `10.1175/1520-0469(1999)056<0599:LMSTDG>2.0.CO;2`
2. Arunajatesan, S., N. Sinha, N.: Unsteady RANS-LES simulations of cavity flowfields. 39th Aerospace Sciences Meeting and Exhibit, Aerospace Sciences Meetings (2001). URL `http://dx.doi.org/10.2514/6.2001-516`
3. Basara, B., Girimaji, S.: Modelling of the cut-off scale supplying variable in bridging methods for turbulence flow simulation. In: Proceedings of International Conference on Jets, Wakes, and Separated Flows, Nagoya, Japan, pp. 17–21 (2013)
4. Basara, B., Poredos, A., Gorensek, P.: Scale-Resolving Simulations of the Flow in Intake Port Geometries. In: SAE Technical Paper. SAE International (2016). URL `http://dx.doi.org/10.4271/2016-01-0589`
5. Batten, P., Goldberg, U., Chakravarthy, S.: LNS - an approach towards embedded LES. American Institute of Aeronautics and Astronautics (2002). URL `http://dx.doi.org/10.2514/6.2002-427`
6. Baumann, M., di Mare, F., Janicka, J.: On the Validation of Large Eddy Simulation Applied to Internal Combustion Engine Flows Part II: Numerical Analysis. Flow, Turbulence and Combustion **92**(1-2), 299–317 (2014). DOI 10.1007/s10494-013-9472-x. URL `http://link.springer.com/10.1007/s10494-013-9472-x`
7. Baya Toda, H., Cabrit, O., Truffin, K., Bruneaux, G., Nicoud, F.: Assessment of subgrid-scale models with a large-eddy simulation-dedicated experimental database: The pulsatile impinging jet in turbulent cross-flow. Physics of Fluids **26**(7), 075108 (2014). URL `http://dx.doi.org/10.1063/1.4890855`
8. Borée, J., Maurel, S., Bazile, R.: Disruption of a compressed vortex. Physics of Fluids **14**(7), 2543–2556 (2002). URL `http://dx.doi.org/10.1063/1.1472505`
9. Borée, J., Miles, P.C.: In-Cylinder Flow. John Wiley & Sons, Ltd (2014). URL `10.1002/9781118354179.auto119`
10. Brusiani, F., Forte, C., Bianchi, G.: Assessment of a Numerical Methodology for Large-Eddy Simulation of ICE Wall Bounded Non-Reactive Flows. In: SAE Technical Paper 2007-01-4145 (2007). URL `doi:10.4271/2007-01-4145`
11. Celik, I.B., Cehreli, Z.N., Yavuz, I.: Index of resolution quality for large-eddy simulations. Journal of Fluids Engineering **127**(5), 949–958 (2005)
12. Davidson, L.: Large-Eddy Simulations: How to Evaluate Resolution. Int. J. Heat Fluid Flow **30**(5), 1016 – 1025 (2009). URL `http://dx.doi.org/10.1016/j.ijheatfluidflow.2009.06.006`. The 3rd International Conference on Heat Transfer and Fluid Flow in MicroscaleThe 3rd International Conference on Heat Transfer and Fluid Flow in Microscale
13. Davidson, L.: Fluid mechanics, turbulent flow and turbulence modeling. Lecture notes for courses TME075, TME256 and TME270, International Master's Programme "Solid and Fluids Mechanics", Chalmers University of Technology, Göteborg, Sweden (2010). URL `http://www.tfd.chalmers.se/~lada/turbulent_flow/lecture_notes.html`
14. Deck, S.: Recent improvements in the zonal detached eddy simulation (zdes) formulation. Theoretical and Computational Fluid Dynamics **26**(6), 523–550 (2012). DOI 10.1007/s00162-011-0240-z. URL `http://dx.doi.org/10.1007/s00162-011-0240-z`
15. Demirdžić, I., Perić, M.: Space conservation law in finite volume calculations of fluid flow. Int J Numer Methods Fluids **8**, 1037–1050 (1988)
16. Dietzel, D., Messig, D., Piscaglia, F., Montorfano, A., Olenik, G., Stein, O., Kronenburg, A., Onorati, A., Hasse, C.: Evaluation of Scale Resolving Turbulence Generation Methods for Large Eddy Simulation of Turbulent Flows . Computers & Fluids **93**, 116 – 128 (2014). DOI http://dx.doi.org/10.1016/j.compfluid.2014.01.013. `http://www.sciencedirect.com/science/article/pii/S0045793014000206`
17. Egorov, Y., Menter, F.R., Lechner, R., Cokljat, D.: The Scale-Adaptive Simulation Method for Unsteady Turbulent Flow Predictions. Part 2: Application to Complex Flows. Flow, Turbulence and Combustion **85**(1), 139–165 (2010). URL `http://dx.doi.org/10.1007/s10494-010-9265-4`
18. Enaux, B., Granet, V., Vermorel, O., Lacour, C., Pera, C., Angelberger, C., Poinsot, T.: Les study of cycle-to-cycle variations in a spark ignition engine. Proceedings of the Combustion Institute **33**(2), 3115–3122 (2011)
19. Ercoftac - Best Practice Guidelines: Computational Fluid Dynmaics of Turbulent Combustion. Edited by L. Vervisch and D. Roekaerts (2016)
20. Farrell, P., Maddison, J.: Conservative interpolation between volume meshes by local Galerkin projection. Computer Methods in Applied Mechanics and Engineering **200**(14), 89 – 100 (2011). URL `http://dx.doi.org/10.1016/j.cma.2010.07.015`
21. Farrell, P., Piggott, M., Pain, C., Gorman, G., Wilson, C.: Conservative interpolation between unstructured meshes via supermesh construction. Computer Methods in Applied Mechanics and Engineering **198**(33-36), 2632–2642 (2009)
22. Fasel, H., Seidel, J., Wernz, S.: A Methodology for Simulations of Complex Turbulent Flows. J. Fluids Eng. **124**, 933–942 (2002)
23. Fröhlich, J., von Terzi, D.: Hybrid LES/RANS methods for the simulation of turbulent flows. Progress in Aerospace Sciences **44**(5), 349–377 (2008). URL `http://dx.doi.org/10.1016/j.paerosci.2008.05.001`
24. Girimaji, S., Abdol-Hamid, K.: Partially-Averaged Navier Stokes Model for Turbulence: Implementation and Validation. 43rd AIAA Aerospace Sciences Meeting and Exhibit pp. 139–165 (2005). DOI doi:10.2514/6.2005-502. URL `http://dx.doi.org/10.2514/6.2005-502`
25. Goryntsev, D., Nishad, K., Sadiki, A., Janicka, J.: Application of LES for Analysis of Unsteady Effects on Combustion Processes and Misfires in DISI Engine. Oil & Gas Science and Technology Revue d'IFP Energies nouvelles **69**(1), 129–140 (2014). URL `http://ogst.ifpenergiesnouvelles.fr/10.2516/ogst/2013125`
26. Gyllenram, W., Nilsson, H.: Design and Validation of a Scale-Adaptive Filtering Technique for LRN Turbulence Modeling of Unsteady Flow. Journal of Fluids Engineering **130**, 051,401 (2008). URL `http://dx.doi.org/10.1115/1.2911685`
27. Hanjalić, K., Hadžiabdić, M., Temmerman, L., Leschziner, M.: Merging LES and RANS Strategies: Zonal or Seamless Coupling?, pp. 451–464. Springer Netherlands, Dordrecht (2004). DOI 10.1007/978-1-4020-2313-2_48. URL `http://dx.doi.org/10.1007/978-1-4020-2313-2_48`

28. Hasse, C., Sohm, V., Durst, B.: Detached eddy simulation of cyclic large scale fluctuations in a simplified engine setup. Int. J. Heat Fluid Flow **30**(1), 32–43 (2009). URL `http://dx.doi.org/10.1016/j.ijheatfluidflow.2008.10.001`
29. Hasse, C., Sohm, V., Durst, B.: Numerical investigation of cyclic variations in gasoline engines using a hybrid URANS/LES modeling approach. Computers & Fluids **39**(1), 25–48 (2010). URL `10.1016/j.compfluid.2009.07.001`
30. Janas, P., Wlokas, I., Böhm, B., Kempf, A.: On the Evolution of the Flow Field in a Spark Ignition Engine. Flow, Turbulence and Combustion pp. 1–28 (2016). DOI 10.1007/s10494-016-9744-3
31. Janicka, J., Sadiki, A.: Large-eddy simulation of turbulent combustion systems. Proceedings of the Combustion Institute **30**(1), 537 – 547 (2005)
32. Jasak, H.: Error analysis and estimation in the finite volume method with applications to fluid flows. Ph.D. thesis, Imperial College, University of London (1996)
33. Khalighi, B.: Study of the intake tumble motion by flow visualization and particle tracking velocimetry. Experiments in Fluids **10**(4), 230–236 (1991). DOI 10.1007/BF00190393
34. Klein, M.: An attempt to assess the quality of large-eddy simulations in the context of implicit filtering. Flow, Turbulence and Combustion **75**, 131–147 (2005)
35. Laget, O., Reveille, B., Martinez, L., Truffin, K., Habchi, C.: LES calculations of a four cylinder engine. SAE Technical Paper 2011-01-0832 (2011). (2011). URL `10.4271/2011-01-0832`
36. Leonard, S., Terracol, M., Sagaut, P.: Commutation error in LES with time-dependent filter width. Computers and Fluids **36**(3), 513–519 (2007). URL `http://dx.doi.org/10.1016/j.compfluid.2005.06.010`
37. Lev Landau, E.L.: Physique thorique. ditions Mir (1989)
38. Lumley, J.: Engines: An Introduction. Cambridge University Press (1999)
39. Martínez, J., Piscaglia, F., Montorfano, A., Onorati, A., Aithal, S.: Influence of spatial discretization schemes on accuracy of explicit LES: Canonical problems to engine-like geometries. Computers & Fluids **117**(0), 62 – 78 (2015). `"http://dx.doi.org/10.1016/j.compfluid.2015.05.007"`
40. Menter, F., Egorov, Y.: The scale adaptive simulation method for unsteady turbulent flow prediction, Part 1: Theory and Model Description. Flow Turbul. Combust. **85**, 113–138 (2010). URL `10.1007/s10494-010-9264-5`
41. Menter, F., Kuntz, M., Langtry, R.: Ten Years of Industrial Experience with the SST Turbulence Model. In: K. Hanjalic, Y. Nagano, & M. Tummers (ed.) Turbulence, Heat and Mass Transfer 4, pp. 1598–1605. Begell House, Inc. (2003)
42. Menter, F.R.: Two-equation eddy-viscosity turbulence models for engineering applications. AIAA Journal **32**(8), 1598–1605 (1994)
43. Misdariis, A., Robert, A., Vermorel, O., Richard, S., Poinsot, T.: Numerical Methods and Turbulence Modeling for LES of Piston Engines: Impact on Flow Motion and Combustion. Oil and Gas Science and Technology - Rev. IFP Energies nouvelles **69**(1), 83–105 (2013). DOI 10.2516/ogst/2013121
44. Misra, A., Pullin, D.I.: A vortex-based subgrid stress model for large-eddy simulation. Physics of Fluids **9**(8), 2443–2454 (1997). DOI 10.1063/1.869361
45. Moeng, C.H., Wyngaard, J.C.: Spectral analysis of large-eddy simulations of the convective boundary layer. Journal of the Atmospheric Sciences **45**(23), 3573–3587 (1988). DOI 10.1175/1520-0469(1988)045⟨3573:SAOLES⟩2.0.CO;2
46. Montorfano, A., Piscaglia, F., Onorati, A.: An Extension of the Dynamic Mesh Handling with Topological Changes for LES of ICE in OpenFOAM. In: SAE Technical Paper 2015-01-0384. SAE International (2015). URL `http://dx.doi.org/10.4271/2015-01-0384`
47. Montorfano, A., Piscaglia, F., Schmitt, M., Wright, Y.M., Frouzakis, C.E., Tomboulides, A.G., Boulouchos, K., Onorati, A.: Comparison of direct and large-eddy simulations of the turbulent flow in a valve/piston assembly. Flow, Turbulence and Combustion **95**(2), 461–480 (2015). DOI 10.1007/s10494-015-9620-6. URL `http://dx.doi.org/10.1007/s10494-015-9620-6`
48. Moureau, V., Domingo, P., Vervisch, L.: From Large-Eddy Simulation to Direct Numerical Simulation of a lean premixed swirl flame: Filtered laminar flame-PDF modeling. Combustion and Flame **158**(7), 1340–1357 (2011). DOI 10.1016/j.combustflame.2010.12.004. URL `http://dx.doi.org/10.1016/j.combustflame.2010.12.004http://linkinghub.elsevier.com/retrieve/pii/S0010218010003585`
49. Moureau, V., Lartigue, G., Sommerer, Y., Angelberger, C., Colin, O., Poinsot, T.: Numerical methods for unsteady compressible multi-component reacting flows on fixed and moving grids. Journal of Computational Physics **202**(2), 710–736 (2005). DOI 10.1016/j.jcp.2004.08.003. URL `http://linkinghub.elsevier.com/retrieve/pii/S0021999104003134`
50. Moureau, V. and Barton, I. and Angelberger, C. and Poinsot, T.: Towards Large-Eddy Simulation in Internal-Combustion Engines: Simulation of a Compressed Tumble Flow. In: SAE Technical Paper 2004-01-1995. SAE International (2004). URL `http://dx.doi.org/10.4271/2004-01-1995`
51. Nguyen, T., Kempf, A.M.: Investigation of Numerical Effects on the Flow and Combustion in LES of ICE. Oil & Gas Science and Technology Revue d'IFP Energies nouvelles **72**(4), 25 (2017). DOI 10.2516/ogst/2017023. URL `http://ogst.ifpenergiesnouvelles.fr/10.2516/ogst/2017023`
52. Nguyen, T.M., Proch, F., Wlokas, I., Kempf, A.M.: Large Eddy Simulation of an Internal Combustion Engine Using an Efficient Immersed Boundary Technique. Flow, Turbulence and Combustion **97**(1), 191–230 (2016). DOI 10.1007/s10494-015-9683-4. URL `http://dx.doi.org/10.1007/s10494-015-9683-4http://link.springer.com/10.1007/s10494-015-9683-4`
53. Nicoud, F., Ducros, F.: Subgrid-scale stress modelling based on the square of the velocity gradient tensor. Flow, Turbulence and Combustion **62**, 183–200 (1999). DOI 10.1023/A:1009995426001
54. Nicoud, F., Toda, H.B., Cabrit, O., Bose, S., Lee, J.: Using singular values to build a subgrid-scale model for large-eddy simulations. Physics of Fluids (1994-present) **23**(8), 085106 (2011). DOI http://dx.doi.org/10.1063/1.3623274. URL `http://scitation.aip.org/content/aip/journal/pof2/23/8/10.1063/1.3623274`
55. Peltier, L.J., Zajaczkowski, F.J., Wyngaard, J.C.: A Hybrid RANS/LES Approach to Large-Eddy Simulation of High-Reynolds-Number Wall-Bounded Turbulence. In: Proceedings of the 2000 ASME Fluids Engineering Division Summer Meeting, Boston, Massachusetts (2000)

56. Piomelli, U., Balaras, E., Pasinato, H., Squires, K., Spalart, P.: The Inner-Outer Layer Interface in Large-Eddy Simulations with Wall-layer Models. Int. J. Heat Fluid Flow **24**, 538–550 (2003)
57. Piscaglia, F., Montorfano, A., Onorati, A.: Development of a Non-Reflecting Boundary Condition for Multidimensional Nonlinear Duct Acoustic Computation. Journal of Sound and Vibration **332**(4), 922–935 (2013). `http://dx.doi.org/10.1016/j.jsv.2012.09.030.`
58. Piscaglia, F., Montorfano, A., Onorati, A.: Development of Fully-Automatic Parallel Algorithms for Mesh Handling in the OpenFOAM-2.2.x Technology. SAE Technical Paper 2013-24-0027 (2013). `http://dx.doi.org/10.4271/2013-24-0027`
59. Piscaglia, F., Montorfano, A., Onorati, A.: A Compressible Dynamic Solver for the Simulation of Turbulent Flows in IC Engine Geometries. In: International Multidimensional Engine Modeling User's Group Meeting At the SAE Congress (2015). `https://imem.cray.com/2015/Meeting-2015/12-IMEM2015-Milano.pdf`
60. Piscaglia, F., Montorfano, A., Onorati, A.: A Scale Adaptive Filtering Technique for Turbulence Modeling of Unsteady Flows in IC Engines. SAE Int. J. Engines, Paper n. 2015-01-0395 (2015). `http://dx.doi.org/10.4271/2015-01-0395`
61. Pitsch, H.: Large-Eddy Simulation of Turbulent Combustion. Annu. Rev. Fluid Mech. **38**, 453–482 (2006)
62. Pope, S.: Turbulent Flows. Cambridge University Press (2000)
63. Pope, S.B.: Ten questions concerning the large-eddy simulation of turbulent flows. New Journal of Physics **6**(1), 35 (2004). URL `http://stacks.iop.org/1367-2630/6/i=1/a=035`
64. Rutland, C.J.: Large-eddy Simulations for Internal Combustion Engines - a Review. Int. J. Eng. Res. **12**(5), 421–451 (2011). URL `http://dx.doi.org/10.1177/1468087411407248`
65. Sagaut, P.: Large-Eddy Simulation for Incompressible Flows: an Introduction. Scientific computation. Springer-Verlag (2006)
66. Spalart, P.R.: Strategies for Turbulence Modelling and Simulations. Int. J. Heat Fluid Flow **21**(3), 252–263 (2000). URL `http://dx.doi.org/10.1016/S0142-727X(00)00007-2`
67. Spalart, P.R. and Jou, W.H. and Stretlets, M. and Allmaras, S.R.: Comments on the Feasibility of LES for Wings and on the Hybrid RANS/LES Approach. In: Advances in DNS/LES, Proceedings of the First AFOSR International Conference on DNS/LES (1997)
68. Speziale, C.: A Combined Large-Eddy Simulation and Time-Dependent RANS Capability for High-Speed Compressible Flows. J. Sci. Comput. **13(3)**, 253–274 (1998)
69. Speziale, C.: Turbulence Modeling for Time-Dependent RANS and VLES: A Review. AIAA Journal **36**, 173–184 (1998)
70. Tezduyar, T.E., Sathe, S., Schwaab, M., Conklin, B.S.: Arterial fluid mechanics modeling with the stabilized space-time fluidstructure interaction technique. International Journal for Numerical Methods in Fluids **57**(5), 601–629 (2008). DOI 10.1002/fld.1633
71. The Engine Combustion Network (ECN): URL `https://ecn.sandia.gov`
72. The OpenFOAM® Foundation: URL `http://www.openfoam.org/dev.php`
73. Travin, A.K. and Shur, M.L. and Spalart, P.R. and Strelets, M.K.: Improvement of Delayed Detached-Eddy Simulation for LES with Wall Modelling. In: ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics, Egmond aan Zee, The Netherlands, September 5-8, 2006. Delft University of Technology; European Community on Computational Methods in Applied Sciences (ECCOMAS) (2006)
74. Truffin, K., Angelberger, C., Richard, S., Pera, C.: Using large-eddy simulation and multivariate analysis to understand the sources of combustion cyclic variability in a spark-ignition engine. Combustion and Flame **162**(12), 4371–4390 (2015). DOI 10.1016/j.combustflame.2015.07.003. URL `http://linkinghub.elsevier.com/retrieve/pii/S0010218015002047`
75. Weller, H.: Controlling the computational modes of the arbitrarily structured C grid. Monthly Weather Review **140**, 3220–3234 (2012). DOI http://dx.doi.org/10.1175/MWR-D-11-00221.1. URL `http://journals.ametsoc.org/doi/pdf/10.1175/MWR-D-11-00221.1`
76. Wilcox, D.C.: Turbulence Modeling for CFD. DCW Industries Inc. (2006)
77. Willems, W.: Numerische Simulation turbulenter Scherströmungen mit einem Zwei-Skalen-Turbulenzmodell. Ph.D. thesis, Institut fur Technische Mechanik, RWTH Aachen (1996)

A Treatment of flow variables across topological changes

When solving conservation problems on moving grids, the discrete form of governing equations must be modified in order to account for the change of cell volumes during the solution. Let us consider the generic conservation equation for variable ϕ in its semi-discrete form for a cell of volume V:

$$\frac{\partial}{\partial t} V \rho \, \phi + \sum_f \rho_f \, \phi_f \left(\varphi_f - \varphi_{M,f}\right) - \sum_f \gamma_\phi \nabla \phi_f = V s_\phi \tag{8}$$

In the above equation, ρ is the density, γ_ϕ is the diffusion coefficient, s_ϕ are the volume sources/sinks of ϕ, φ_f is the cell face flux $\varphi_f = \mathbf{U}_f \cdot \mathbf{n}$ and $\varphi_{M,f}$ is the corresponding flux due to point motion. To avoid spurious mass sources, the formulation of $\varphi_{M,f}$ must fulfill the so-called Geometric Conservation Law (GCL) [15]:

$$\frac{d}{dt} \int_V dV - \int_S \mathbf{u}_\mathrm{b} \cdot d\mathbf{S} = 0 \tag{9}$$

or, in semi-discrete form:

$$\left.\frac{dV}{dt}\right|_{t=n-1}^{t=n} - \sum_f \varphi_{m,f} = 0 \tag{10}$$

Fulfillment of the GCL in its discrete form (DGCL), Eq. (10) imposes the way mesh fluxes $\varphi_{M,f}$ are computed. For first-order implicit schemes this simply means:

$$\varphi_{M,f} = \mathrm{d}V_f \tag{11}$$

where $\mathrm{d}V_f$ is the volume swept by each cell face across Δt. A correct implementation of the Discrete GCL (DGCL) guarantees mass conservation in a constant-topology moving grid.

Discretization of time derivative depends upon the chosen scheme, and for first-order Euler one has:

$$\frac{\partial \rho V \phi}{\partial t} \approx \frac{V^{(n+1)} \rho^{(n+1)} \phi^{(n+1)} - V^{(n)} \rho^{(n)} \phi^{(n)}}{\Delta t} \tag{12}$$

Where the superscript $(n+1)$ and (n) indicate respectively 'new' and 'old' solution variables. The latter are mapped from the old to the new mesh: if the topology does not change there is a one-to-one correspondence between cells of the two:

$$\mathcal{M}_c\big(\{c_0 \dots c_m\}^n\big) = \{c_0 \dots c_m\}^n \mapsto \{c_0 \dots c_m\}^{n+1} \tag{13}$$

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

$$\phi\big(t^n, \mathbf{x}^{(n+1)}\big) = \mathcal{M}_c\big[\phi(t^n, \mathbf{x}^n)\big] \tag{14}$$

If cells are split or merged across a time step (as it occurs in case of layer addition/removal), the one-to-one cell map between the old and the new mesh is lost for cells and faces that undergo the topology change, with three important consequences. Firstly, old-time values that enter time derivatives must be approximated in added/merged cells for the computation of $\partial\phi/\partial t$. Secondly, since layer addition/removal is always performed in conjunction with the piston motion, DGCL must be preserved throughout the mesh change. Finally, since the face-to-face map cannot be computed across a topology change, cell face fluxes φ_f^n that fulfill continuity must be recovered for the whole mesh.

Hence, at the occurring of layer addition/removal, the mesh change is computed in two steps. First, the topology is modified and variables are remapped according to Eq. (14) in unchanged cells, whereas initialization of solution variables in new cells is performed as follows:

- If cells are split (addition of a new layer), in the newly created cells all intensive variables will have the same value they had in the original (unsplit) cell.
- If cells are merged (removal of a layer), in the new cells intensive variable are volume-averaged from the original cells.

After the topology change points are displaced according to piston motion. During this step it is possible to enforce the DGCL by computing mesh fluxes as in Eq. (11). Finally, continuity is enforced by solving a Poisson equation (Eq. (15)) for a pressure corrector p' that is used for correcting remapped fluxes (Eq. (16)) [32]:

$$\frac{\partial p'}{\partial t} - \nabla^2 p' + \nabla \cdot \varphi\big(t^n, \mathbf{x}^{(n+1)}\big) - \Big[\nabla \cdot \varphi\big(t^{(n)}, \mathbf{x}^{(n)}\big)\Big]^{(n+1)} = 0 \tag{15}$$

$$\varphi^{(n+1)} = \varphi\big(t^{(n)}, \mathbf{x}^{(n+1)}\big) + \nabla p' \tag{16}$$

The above-outlined treatment of cell- and face-centered variables ensures conservativeness of the main flow quantities, at least as long as the DGCL is fulfilled and continuity is enforced by Eqs. (15) and (16). It must be noted, on the other hand, that if momentum equation is solved for velocity $\mathbf{U}$ and not $\rho\mathbf{U}$, strict conservation of momentum is not attainable across remapping. This would cause a small error in moving cells, which is proportional to Δt but is not directly related to topology change.

To test the numerical properties of the solver a test case consisting in an uniformly accelerated piston (represented in Fig. 17) has been simulated and results were compared against the theoretical solution [37] for the cases of outward- and inward-moving piston, which require respectively addition and removal of cells.

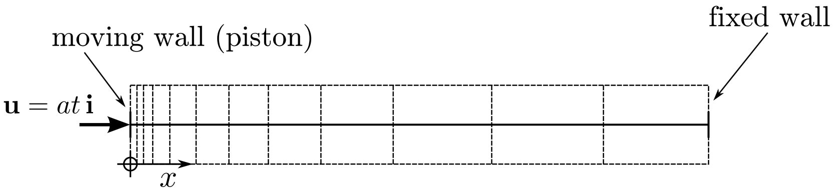
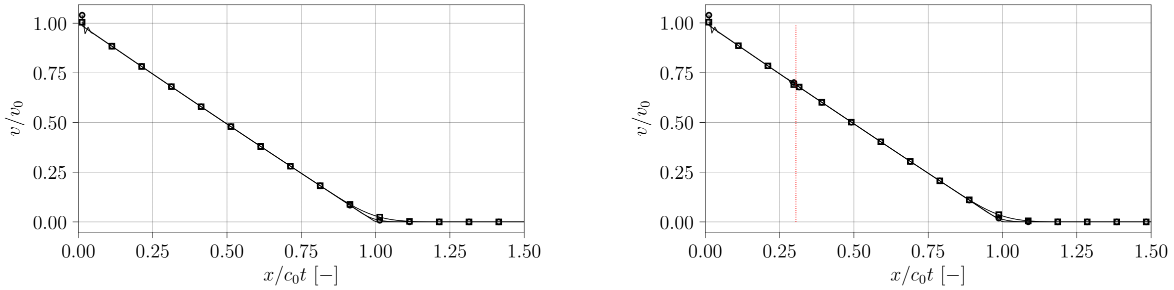


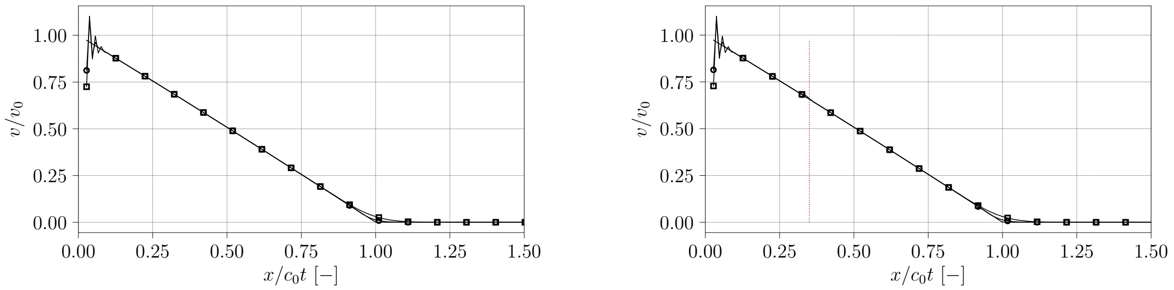
Fig. 17: Schematic representation of the Uniformly Accelerated Piston test case.



(a) cell stretching only

(b) layer addition

Fig. 18: Velocity profile along the axis of an uniformly accelerated piston moving outwards (cell addition). — analytical solution, –○– $\Delta t = 10^{-3}$ s; –□– $\Delta t = 8 \cdot 10^{-3}$ s. Dotted vertical line represents the location of layer addition.



(a) cell stretching only

(b) layer addition

Fig. 19: Velocity profile along the axis of an uniformly accelerated piston moving inwards (cell removal). — analytical solution, –○– $\Delta t = 10^{-3}$ s; –□– $\Delta t = 8 \cdot 10^{-3}$ s. Dotted vertical line represents the location of layer removal.

Figs. 18 and 19 report results for inward- and outward- moving piston for two cases: with cell stretching only (sub-figures (a)) and with topological changes (sub-figures (b)). As one can see, the agreement with analytical solution is very good, with no visible effect in the layer A/R region (dotted line in Figs. 18(b) and 19(b)). The discrepancy between numerical and analytical solution around $x/c_0t = 1$ (with c_0 as the undisturbed speed-of-sound) is due only to the discretization error. Velocity under- and over-shoot close to $x/c_0t = 0$ are caused by the cell motion near the boundary and they are not related to the treatment of flow variables across topological changes. They become negligible if strong advection dominates the flow close to the piston boundary, as in the cases studied in the present paper.