International Journal for
NUMERICAL METHODS IN
FLUIDS

Unsteady shock-fitting for unstructured grids

Journal:	*International Journal for Numerical Methods in Fluids*
Manuscript ID:	FLD-15-0175
Wiley - Manuscript type:	Research Article
Date Submitted by the Author:	25-May-2015
Complete List of Authors:	Bonfiglioli, Aldo; Università degli Studi della Basilicata, Scuola di Ingegneria Paciorri, Renato; Universita' degli Studi di Roma ``La Sapienza'', Dipartimento di Ingegneria Meccanica e Aerospaziale Campoli, Lorenzo; Universita' degli Studi di Roma ``La Sapienza'', Dipartimento di Ingegneria Meccanica e Aerospaziale
Keywords:	ALE: Arbitrary Lagrangian Eulerian, Compressible flow, Euler flow, Verification, Transonic, Embedded boundary

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2010; **00**:1–16
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld

Unsteady shock-fitting for unstructured grids.

A. Bonfiglioli[1*] R. Paciorri[2], L. Campoli[2]

[1]*Scuola di Ingegneria, Università degli Studi della Basilicata, Via dell'Ateneo Lucano 10, 85100 Potenza, Italy*
[2]*Dipartimento di Ingegneria Meccanica e Aerospaziale, Università degli Studi di Roma "La Sapienza", via Eudossiana 18, 00184 Rome, Italy*

SUMMARY

An unstructured, shock-fitting algorithm, originally developed to simulate steady flows, has being further developed to make it capable of dealing with un-steady flows. The present paper discusses and analyses the additional features required to extend to *unsteady* flows the steady algorithm. The properties of the unsteady version of this novel unstructured shock-fitting technique are tested by reference to the inviscid interaction between a vortex and a planar shock: a comparative assessment of shock-capturing and shock-fitting is made for the same test problem, using nearly identical grids. Copyright © 2010 John Wiley & Sons, Ltd.

Received . . .

KEY WORDS: Shock-fitting, shock-capturing, un-steady flows, unstructured meshes

1. INTRODUCTION

When shock-capturing schemes are used to model the interaction of shock waves with compressible turbulence or sound waves, the discretization errors generated along the captured shock wave can severely degrade the fidelity of the flow simulation within the entire shock-downstream region [1, 2].

These limitations appear to be rooted in the fundamental ingredients of shock-capturing discretizations, namely the existence of intermediate shock points, located in between the pre- and post-shock states, that are a mere numerical artefact and have nothing to do with the true internal structure of the shock-wave [3, 4]. This observation points to a fundamental weakness of the shock-capturing paradigm, so that it is not surprising that Pirozzoli in [2] affirms that: "These limitations .... can only be overcome by some form of shock-fitting".

Shock-fitting algorithms on structured grids, which still nowadays find their way in compressible DNS, see e.g. [5, 6], are limited to simple flow configurations. Indeed, two different shock-fitting approaches have been developed since the late 60s in the structured-grid framework: boundary shock-fitting [7] and floating shock-fitting [8, 9]. In the former, the shock is made to coincide with a boundary of the grid and it moves, thus deforming the overall mesh. In the latter approach, the mesh is fixed and the shock is free to move independently of the grid. These two approaches have intrinsic limitations that reduce their applicability. In the boundary shock-fitting approach, the shock plays the role of a boundary condition and this significantly simplifies the implementation; nevertheless, the constraint of using grid-block boundaries as fitted shocks, limits the applicability of the technique to simple flows in which the shocks do not change in number and kind of interactions,

*Correspondence to: Scuola di Ingegneria, Università degli Studi della Basilicata, Via dell'Ateneo Lucano, 10, 85100 Potenza, Italy. E-mail: aldo.bonfiglioli@unibas.it
†Please ensure that you use the most up to date class file, available from the FLD Home Page at
http://www3.interscience.wiley.com/journal/2861/home

i.e. the flow topology, during the transient. The floating shock-fitting approach, although more versatile than the former from a topological point of view, becomes algorithmically very complex when high order scheme are used, see e.g. [10], and the implementation of this technique in an existing gas-dynamic solver requires significant modifications of its computational kernel.

In the attempt to relieve most of the algorithmic difficulties encountered by shock-fitting methods when used on structured grids, the authors have recently developed an *unstructured*, shock-fitting algorithm capable of simulating *steady* flows in two [11, 12, 13] and three [14] spatial dimensions. The unstructured, shock-fitting algorithm has features of both the boundary and floating variants of the shock-fitting technique that had been proposed and used in the structured-grid framework over the last fifty years: the fitted shocks are treated as interior *boundaries* of zero thickness that are free to *float* throughout a triangular/tetrahedral mesh that covers the entire computational domain and locally adapts to follow the shock motion. The Rankine–Hugoniot jump relations are used to compute the Lagrangian motion of the discontinuities and an unstructured, vertex-centred, shock-capturing solver is used to discretise the governing PDEs in the smooth regions of the flow-field.

The aforementioned methodology is here being further developed to make it capable of dealing with un-steady flows. This can be accomplished by introducing three new ingredients: *i*) the shock-capturing code must be made capable of working in an Arbitrary Lagrangian Eulerian (ALE) setting; *ii*) the temporal accuracy of the Lagrangian shock motion must be raised to second order; and *iii*) the algorithm must be capable of automatically detecting changing flow topologies, such as those that may occur when a shock meets another shock or a solid wall.

The former two issues are addressed in this paper, whereas the latter is left for future work.

The paper is organised as follows. Sect. 2 describes the numerical method: the un-steady shock-fitting algorithm is described in details in Sect. 2.1, whereas the discretization adopted in the shock-capturing code is only briefly summarised in Sect. 2.2. The numerical results are presented in Sect. 3: a smooth flow case is used in Sect. 3.1 to confirm that the ALE shock-capturing discretization is second-order-accurate in both space and time, whereas a classical problem that involves the interaction between a vortex and a steady planar shock is used in Sect. 3.2 to demonstrate the current capabilities of the un-steady, unstructured, shock-fitting technique, also by comparison with shock-capturing calculations.

2. THE NUMERICAL METHOD

The numerical method we propose consists in the loose coupling between the unstructured shock-fitting algorithm that is used to model the discontinuities (both shocks and slip-lines) and an unstructured shock-capturing code that is used to discretise the governing PDEs in smooth regions of the flow-field.

The two codes are loosely coupled in the sense that the shock-capturing code is invoked as a black box by the shock-fitting one. This has obvious consequences in terms of algorithmic simplicity, since it allows to re-use any existing shock-capturing code, as long as its discretization is vertex centred. Recent work [15] has indeed shown that modularity can be a key feature of the proposed unstructured shock-fitting technique.

The unstructured shock-fitting algorithm consists of two key ingredients: 1) a local re-meshing technique that constructs a time-dependent mesh in which the fitted discontinuities are internal boundaries of zero thickness and 2) an algorithm for solving the Rankine–Hugoniot jump relations that provides the Lagrangian velocity of the discontinuity and an updated set of dependent variables within the downstream side of the fitted shock. The shock-fitting algorithm provides a moving, internal boundary of zero thickness to the unstructured shock-capturing solver that is used to discretised the governing PDEs away from the fitted discontinuities.

In the next two sections we will describe the shock-fitting algorithm and shock-capturing code.

2.1. Shock fitting algorithm

The unstructured shock-fitting algorithm can handle both shocks and contact discontinuities, or slip-lines. Hereafter we will refer to shock waves, because this is the only kind of discontinuity that has

DOI: 10.1002/fld

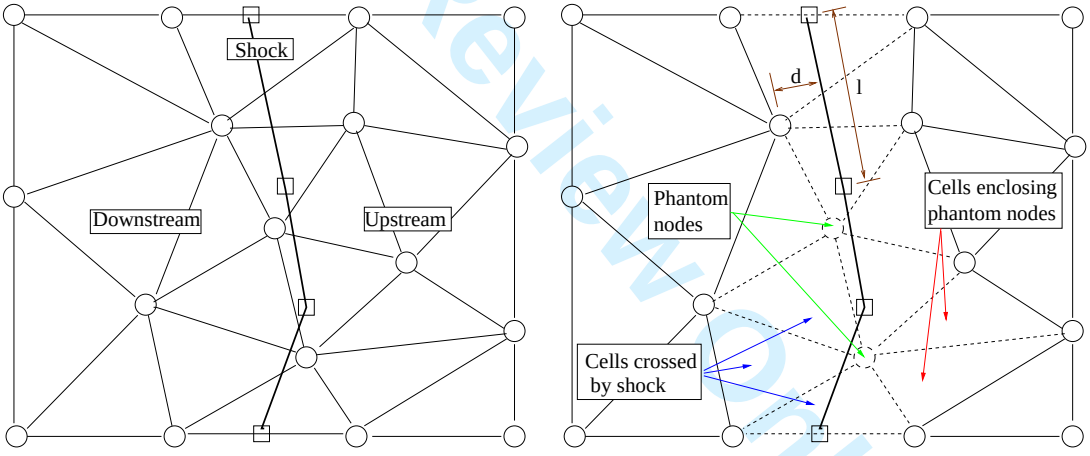
been modelled in the numerical examples of Sect. 3. Details concerning the numerical modelling of contact discontinuities as well as interacting discontinuities can be found in [12, 16].

In order to illustrate the algorithmic features of proposed algorithm, let us consider a two-dimensional domain and a shock front crossing this domain at time level t (see Fig. 1(a)). In two space dimensions, the fitted shock fronts are made of a connected series of line segments (which we call the shock edges) that join the shock points (marked by squares in Fig. 1(a)). These shocks are free to move throughout a background triangular mesh (whose nodes are denoted by circles in Fig. 1(a)), that covers the entire computational domain. It is worth underlining that the position of the shock points is completely independent of the location of the nodes of the background grid. Moreover, two sets of values, corresponding to the upstream and downstream states, are assigned to each shock point, whereas each node of the background mesh is characterised by a single set of dependent variables.

We assume that at time t the solution Z and nodal grid velocity w is known at all grid and shock points. Within the shock points the grid velocity coincides with the shock speed, within the grid-points of the background triangulation the grid velocity is zero, unless the grid moves and/or deforms.

The process that leads from time t to an updated mesh and solution at time $t + \Delta t$ can be split into several steps that will be described in detail in the following paragraphs.

2.1.1. Cell removal around the shock front The first step consists in removing cells around the shock front. As shown in Fig. 1(b), all cells of the background mesh that are crossed by the shock line are removed along with the mesh points that are located too close to the shock front. We call



(a) Shock front moving over the background triangular mesh at time t.

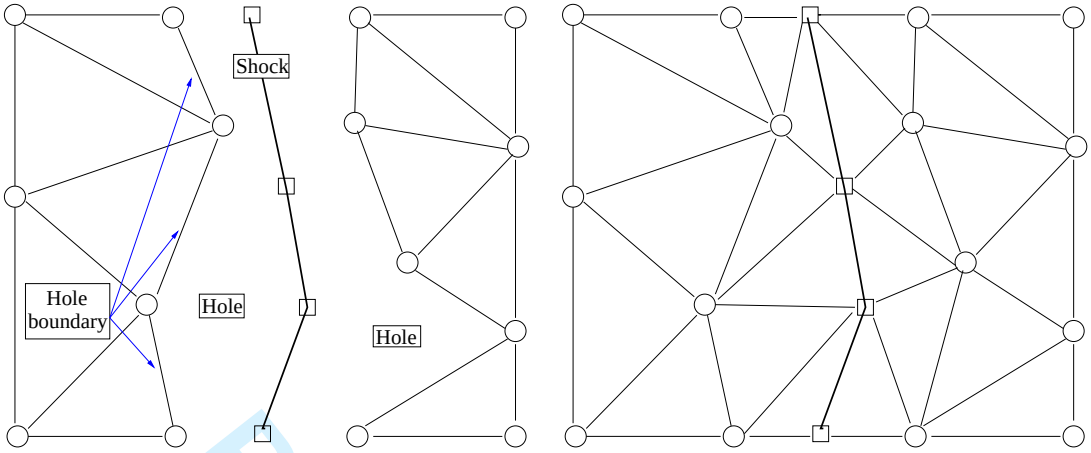
(b) Dashed lines mark the cells to be removed; dashed circles denote the phantom nodes.

Figure 1. Background triangulation and shock front.

"phantom" those grid-points of the background mesh (shown using dashed circles in Fig. 1(b)) that are removed due to their proximity to the shock front; moreover, all cells having at least one phantom node among their vertices are also removed from the background triangulation; these are the cells shown using dashed edges in Fig. 1(b). Further details concerning the criterion used to remove the phantom nodes can be found in [11].

2.1.2. Local re-meshing around the shock front Following the cell removal step, the background triangulation has been split into two or more disjoint sub-domains, as shown in Fig. 2(a). The hole dug by the fitted front is then re-meshed using a constrained Delaunay triangulation (CDT): both the shock segments and the edges belonging to the boundary of the hole are constrained to be part of the final triangulation. No further mesh point is added by the CDT, which is currently performed using the `triangle` [17] code.

DOI: 10.1002/fld



(a) The background mesh is split into disjoint sub-domains by a hole which encloses the shock.

(b) The triangulation around the shock has been rebuilt.

Figure 2. The shock front digs a hole in the mesh which is later re-meshed.

Upon completion of this stage, the computational domain is discretised by a mesh in which the shock points and the shock edges are part of the triangulation. This is what we call the "shock-fitting mesh", it differs from the background triangulation only in the neighbourhood of the shock front: compare figs. 1(a) and 2(b).

2.1.3. Computation of the tangent and normal unit vectors In order to be able to apply the jump relations, tangent and normal unit vectors are needed within each shock point. This is accomplished using finite-difference (FD) formulae which involve the coordinates of the shock point itself and those of its neighbouring shock points. When the shock-downstream flow is subsonic, the FD formula is centred about the shock point where the normal has to be computed. When the shock-downstream flow is supersonic, however, the FD formulae have to be one-sided in order to respect the domain of dependence and thus avoid that geometrical instabilities arise along the shock front. Details concerning the criterion used to select the shock points to be used in the one-sided FD formulae are given in [11] and one-sided formulae that are second-order accurate for shock-fronts made up of un-evenly spaced shock nodes are given in [18].

2.1.4. Solution update using the shock-capturing code Using the shock-fitting mesh as input, a single time step calculation is performed using the unstructured shock-capturing solver which provides updated nodal values within all grid and shock points at time level $t + \Delta t$. Since the discontinuities are seen by the shock-capturing code as internal boundaries that move with the shock speed, the solution for the mesh points located on the upstream and downstream sides of a discontinuity will be updated using information coming only from the computational sub-domain that is attached to that side of the discontinuity.

If we consider the upstream state of a shock wave, the update provided by the shock-capturing solver at time level $t + \Delta t$ is entirely correct. Indeed, within the supersonic, upstream (low-pressure) region, all waves (acoustic, entropy and vorticity) propagate towards the shock so that no boundary condition is required on this side of the internal boundary. The situation is different within the subsonic region attached to the downstream (high-pressure) side of the shock. Here the entropy, vorticity and forward moving acoustic wave propagate away from the discontinuity. Therefore, the provisional values computed by the shock-capturing code for the grid-points located on the downstream side of the shock are wrong. The downstream flow is however subsonic in the shock-normal direction and the backward moving acoustic wave conveys the following signal:

$$R_d^{t+\Delta t} = \tilde{a}_d^{t+\Delta t} + \frac{\gamma - 1}{2} \left(\tilde{u}_n\right)_d^{t+\Delta t}, \tag{1}$$

DOI: 10.1002/fld

from the downstream region towards the downstream side of the shock. One can assume that the Riemann variable defined by Eq. (1) is correctly computed by the shock-capturing code.

In Eq. (1) $\tilde{a}_d^{t+\Delta t}$ and $\tilde{\vec{u}}_d^{t+\Delta t}$ are the values of the acoustic and flow velocity of the downstream state of the shock nodes computed by the shock-capturing solver. These flow variables have been marked with a "tilde" to emphasise the fact that these are the provisional (incorrect) values computed at time $t+\Delta t$ by the shock-capturing code before enforcing the jump relation across the discontinuity, as described in Sect. 2.1.5. By contrast, the Riemann variable $R_d^{t+\Delta t}$ in Eq. (1), even if computed using the provisional values, is correct and, therefore, it has not been marked by the tilde.

2.1.5. Enforcement of the jump relations The missing pieces of information that are needed to correctly update the solution within all grid-points located on the discontinuities are provided in the current step, which consists in enforcing the Rankine-Hugoniot relations between the upstream and downstream states.

For notational convenience, we introduce the flow velocity relative to the discontinuity:

$$\vec{v} = \vec{u} - \vec{w} \tag{2}$$

where $\vec{w}$ is the velocity of the discontinuity relative to an inertial reference frame. As explained in Sect. 2.1.4, the shock upstream state ($\vec{u}_u$, a_u, ρ_u) has been correctly updated at time level $t+\Delta t$ by the shock-capturing solver. Within the downstream state, only the Riemann variable $R_d^{t+\Delta t}$ is correctly computed by the shock-capturing solver.

The "correct" downstream state and the shock speed component w normal to the discontinuity are then obtained by solving a system of five non-linear algebraic equations. Four of these are the Rankine-Hugoniot jump relations:

$$\rho_u^{t+\Delta t}\,(v_n)_u^{t+\Delta t} = \rho_d^{t+\Delta t}\,(v_n)_d^{t+\Delta t} \tag{3a}$$

$$p_u^{t+\Delta t} + \rho_u^{t+\Delta t}\left(v_n^2\right)_u^{t+\Delta t} = p_d^{t+\Delta t} + \rho_d^{t+\Delta t}\left(v_n^2\right)_d^{t+\Delta t} \tag{3b}$$

$$(u_\tau)_u^{t+\Delta t} = (u_\tau)_d^{t+\Delta t} \tag{3c}$$

$$H_u^{t+\Delta t} = H_d^{t+\Delta t} \tag{3d}$$

$$R_d^{t+\Delta t} = a_d^{t+\Delta t} + \frac{\gamma-1}{2}\,(u_n)_d^{t+\Delta t} \tag{3e}$$

while the fifth equation uses the only "correct" information, given by Eq. (1), computed by the shock-capturing solver on the downstream side of the shock. In Eqs. (3) H is the specific total enthalpy of the relative motion:

$$H = \frac{\gamma}{\gamma-1}\frac{p}{\rho} + \frac{v_n^2+v_t^2}{2}.$$

Observe that all variables in the RHS of Eqs. (3) are unknown, whereas all values on the LHS (except the shock speed w) are those "correctly" updated by the shock-capturing solver at time level $t+\Delta t$. The five unknown quantities are $a_d^{t+\Delta t}$, $\rho_d^{t+\Delta t}$, the two Cartesian components of the velocity vector $\vec{u}_u^{t+\Delta t}$ and the shock speed $w^{t+\Delta t}$, since pressure can be expressed as a function of sound speed and density. A Newton-Raphson algorithm is used to solve the system of Eqs. (3) within each shock point.

2.1.6. Shock displacement The enforcement of the jump relations described in Sect. 2.1.5 provides the speed at which the points located on both the upstream and downstream side of the discontinuity move along the local normal vector. The position of the discontinuity at time level $t+\Delta t$ can therefore be computed by displacing all the points located along the discontinuity; this is shown in Fig. 3(b) where the dashed and solid lines represents the discontinuity at time level t, resp. $t+\Delta t$. When simulating steady flows, this can be accomplished using the following first-order-accurate integration formula:

$$P_i^{n+1} = P_i^n + w_{i,sh}^n \Delta t. \tag{4}$$

DOI: 10.1002/fld

The low order-of-accuracy of Eq. (4) does not affect the spatial accuracy of the steady state solution which depends on the spatial accuracy of the gas-dynamics solver and that of the tangent and normal unit vectors. Therefore, Eq. (4) can be safely used for steady computations in conjunction with second-order-accurate (in space) flow solvers and the shock-normal calculation described in Sect. 2.1.3.

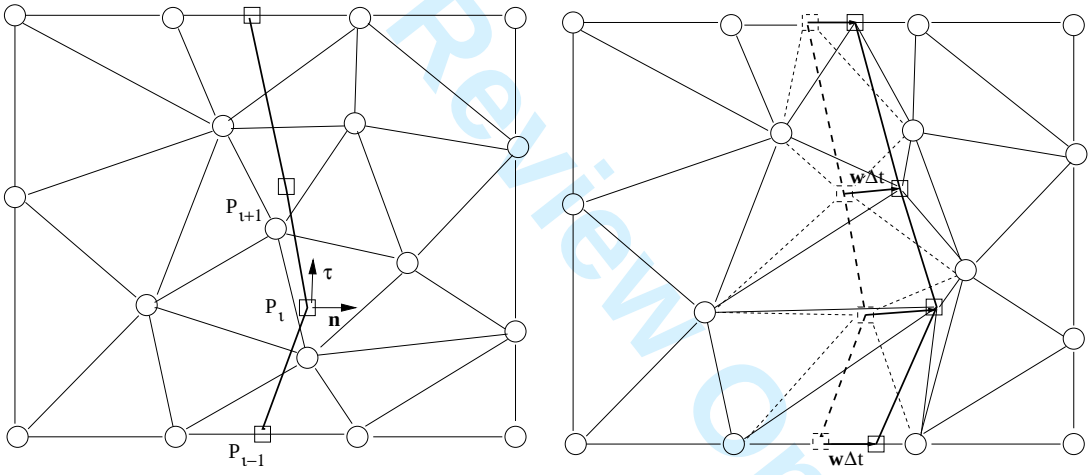
On the contrary, when dealing with unsteady flows, the temporal accuracy of the shock motion has to be the same as that of the shock-capturing solver, i.e. second-order-accurate in our case. The order of accuracy of the shock trajectory has been raised to second order by implementing a predictor-corrector type temporal integration scheme. More specifically, the predictor step estimates the position of the shock at time level $n + 1/2$ using the explicit Euler scheme:

$$P_i^{n+1/2} = P_i^n + w_{i,sh}^n \mathbf{n}_i^n \Delta t/2. \tag{5a}$$

The shock speed $w_{i,sh}^{n+1/2}$ and the normal unit vector $\mathbf{n}_i^{n+1/2}$ at time level $n + 1/2$ are then computed using the intermediate shock position $P_i^{n+1/2}$ and, finally, the position of each shock point is updated at time level $n + 1$ in the corrector step:

$$P_i^{n+1} = P_i^n + w_{i,sh}^{n+1/2} \mathbf{n}_i^{n+1/2} \Delta t. \tag{5b}$$

In practice, the seven steps (described in Sect. 2.1.1 to 2.1.7) that make up the shock-fitting



(a) Calculation of the shock-tangent and shock-normal unit vectors.

(b) The shock displacement induces mesh deformation.

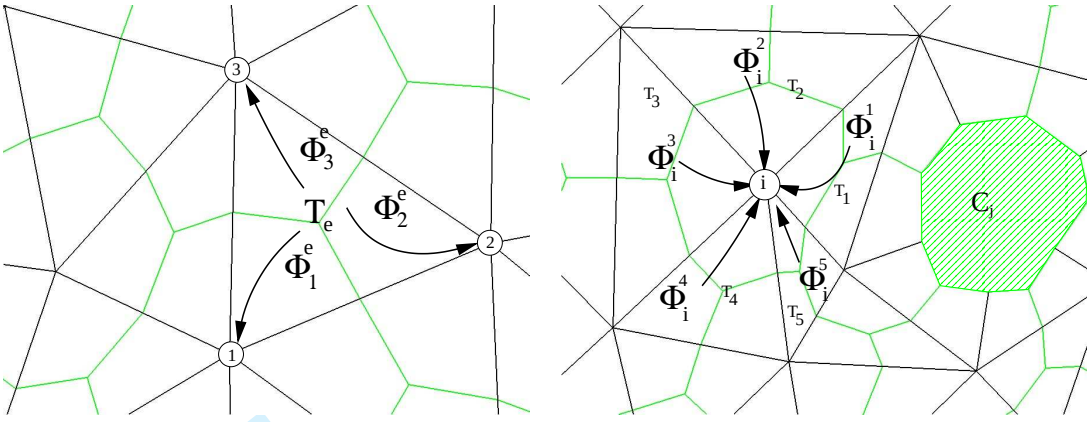
Figure 3. Mesh topology in the neighbourhood of the discontinuities.

algorithm are repeated twice per physical time step: once for the predictor and once for the corrector stage.

Figure 3(b) also shows that even when the background mesh is fixed in space, the triangular cells that abut on the shock front have one of their edges that moves with the shock, thus deforming the cell. This implies that the shock capturing solver that is used in Step 2.1.4 must be capable of dealing with moving meshes.

Finally, the choice of the physical time step Δt to be used in the shock-capturing code and in Eqs. (5) to move the shock, is not only constrained by the stability criterion of the shock-capturing solver, but it is also chosen in such a way that during the time interval $[t, t + \Delta t]$ the shock will remain within the hole that it has dug in the background mesh, see Fig. 2(a). By doing so, none of the grid-points of the shock-fitting mesh will be overcome by the moving discontinuity, as shown in Fig. 3(b).

DOI: 10.1002/fld



(a) The flux balance of triangle e is scattered among its vertices.

(b) Grid-point i gathers the fractions of cell residuals from the surrounding cells.

Figure 4. Residual distribution concept.

2.1.7. Interpolation of the phantom nodes In the previous steps all nodes of the shock-fitting mesh have been updated at time level $t + \Delta t$. However, the phantom nodes, which belong to the background mesh but do not belong to the shock-fitting mesh, have not been updated. During the current time step, the shock front might have moved sufficiently far away from its previous position, that some of the phantom nodes may re-appear in the shock-fitting mesh at the next time step. It follows that also the phantom nodes need to be updated to time level $t + \Delta t$. The update of the phantom nodes is easily accomplished by transferring, using an interpolation that preserves the spatial order-of-accuracy of the discretization, the solution at time level $t + \Delta t$ from the current shock-fitting mesh to the grid-points of the background one.

Once the phantom nodes have been updated, the shock-fitting mesh used in the current time interval has completed its task and can be removed. At this stage the numerical solution has correctly been updated at time level $t + \Delta t$, taking into account the shock front displacement. The next time level can be computed re-starting from the first step 2.1.1 of the algorithm.

2.2. *Shock-capturing*

The `eulfs` code is an in-house, unstructured shock-capturing CFD solver that has been developed over the last fifteen years; see [19] for a detailed description of its basic features and [20] for more recent developments. It relies on Fluctuation Splitting (FS), or Residual Distribution [21, 22, 23] schemes for the spatial discretisation. In the FS approach the dependent variables are stored at the vertices of the computational mesh which is made up of triangles in the 2D space, and tetrahedra in 3D and are assumed to vary linearly and continuously in space. The inviscid flux balance Φ^e (also referred to as the cell residual or cell fluctuation) is evaluated over each triangular/tetrahedral element e by means of a conservative linearisation [24] based on the parameter vector: $Z = \left(\sqrt{\rho}, \sqrt{\rho}H, \sqrt{\rho}u, \sqrt{\rho}v\right)^T$, and scattered to the element vertices using signals Φ_i^e, see Fig. 4(a). Within a cell e, the signals have to sum up to the net flux for conservation: $\sum_{i\in e} \Phi_i^e = \Phi^e$. The nodal residual is then assembled by collecting fractions Φ_i^e of the net fluxes Φ^e associated with all the elements by which the node i is surrounded, as schematically shown in Fig. 4(b). The various FS schemes proposed in the literature differ by the way cell residuals are split into signals. It is possible to construct schemes that depend linearly upon the solution (when solving a linear PDE) and are either monotonicity preserving, but limited to first order of accuracy or, if second order accurate, lead to oscillatory behaviour in the neighbourhood of captured discontinuities. The N scheme [25] belongs to the former class, whereas the FS version [26, 27] of the popular Lax-Wendroff (LW) scheme to the latter. The LW scheme is the shock-capturing scheme used in conjunction with shock-fitting throughout this paper, since it allows to achieve second-order-accuracy in both space and

DOI: 10.1002/fld

time while retaining an explicit time stepping. Other explicit schemes that might be used with the proposed shock-fitting algorithms are the Runge-Kutta schemes described in [28]. Using the LW scheme, the signals sent to vertex i of cell e are:

$$\Phi_i^e = \left[\frac{1}{d+1}\, I + \frac{1}{2}\frac{\Delta t}{|T^e|} K_i^e\right] \Phi^e \tag{6}$$

where I is the identity matrix of order $d+2$ and K_i^e, the so-called inflow parameter, is a matrix that depends upon the cell-averaged Jacobian matrix of the inviscid fluxes and the normal to the edge opposite vertex i, see [19] for details. When the grid moves and/or deforms, matrix K_i^e also depends upon the cell-averaged grid velocity.

The following explicit update formula is obtained for the ALE-LW scheme:

$$|C_i|^{n+1} U_i^{n+1} \quad = \quad |C_i|^n U_i^n + \Delta t \sum_{e \ni i} \Phi_i^e \tag{7}$$

where $|C_i|$ denotes the area of the median dual cell centred about grid-point i, see Fig. 4(b), and U is the vector of the conserved variables which can be computed from parameter vector using the following identity:

$$U = \frac{1}{2}\left(\frac{\partial U}{\partial Z}\right) Z.$$

The Geometric Conservation Law is satisfied as long as the signals in Eq. (6) are evaluated on the mesh at time level $n + \frac{1}{2}$ and the telescoping property of the fluxes is guaranteed using the approach described in [29].

3. NUMERICAL RESULTS

3.1. *Verification of the ALE formulation*

The spatial and temporal accuracy of the ALE-LW scheme has been verified using an exact solution of the un-steady Euler equations which consists in a vortex (characterised by the perturbation velocity field, $\tilde{\mathbf{u}}$), convected by a uniform stream of magnitude $|\mathbf{u}_\infty|$. More precisely, using a polar coordinate system with the origin attached to the centre of the vortex and moving at constant speed $|\mathbf{u}_\infty|$, the perturbation velocity field, which is a particular solution of the steady Euler equations, consists in a clockwise vortex characterised by a purely tangential velocity component:

$$\tilde{u}_\theta \quad = \quad -\epsilon\, |\mathbf{u}_\infty|\, \tau\, e^{\alpha\left(1-\tau^2\right)} \tag{8a}$$

$$\tilde{u}_r \quad = \quad 0 \tag{8b}$$

In Eq. (8) $\tau = r/r_c$ is the non-dimensional radial distance from the pole of the moving reference frame and ϵ, α and r_c are parameters that control the shape and magnitude of the perturbation. Other kinematic features of this particular solution of the Euler equations are: a divergence-free velocity field, which implies that density is constant along the streamlines and a non zero vorticity field. Since the free-stream flow is isentropic, the thermodynamic variables are easily obtained from the linear momentum equation. Moreover, Crocco's form of the steady momentum equation implies that there must be a gradient of the perturbation total enthalpy in the radial direction, so that total enthalpy also changes across the streamlines. Making the following choice of constants in (8): $\epsilon = 0.3$, $\alpha = 0.204$ and $r_c/L = 0.05$, the maximum velocity perturbation is about 0.35 % of the free-stream velocity magnitude and the radius of the vortex about $0.35\, L$, L being the reference length scale. The uniform, background flow is supersonic: $M_\infty = 1.8$ and the reference time scale is $L/|\mathbf{u}_\infty|$.

The time-dependent computational domain $\Omega(t)$ is a rectangle; its initial size is $\Omega(0) = [-0.5\, L, 1.5\, L] \times [-0.5\, L, 0.5\, L]$. Starting from a coarse Delaunay grid, five levels of nested triangulations have been created by recursive subdivision of the coarsest one; their characteristics are summarised in Tab. I.

DOI: 10.1002/fld

Table I. Inviscid vortex convected by a supersonic stream: characteristics of the various nested grid levels.

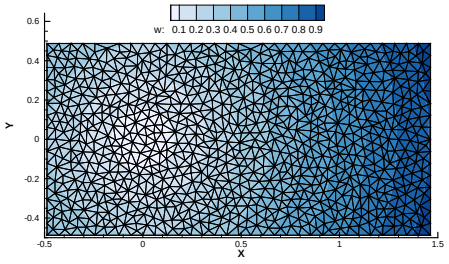
level	grid-points	triangles
1	892	1688
2	3471	6752
3	13693	27008
4	54393	108032
5	216817	432128

Each mesh deforms according to the following analytical mapping:

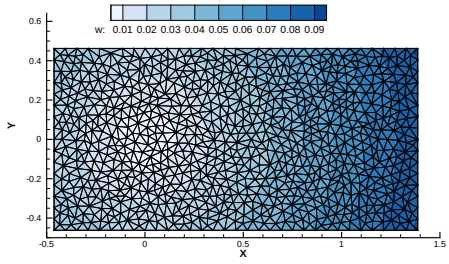
$$(\mathbf{x}_i^n - \mathbf{x}_v) = \left(\mathbf{x}_i^0 - \mathbf{x}_v\right) \{1 - B_1 \left[\cos\left(\omega t\right) - 1\right]\} \qquad n = 0, .., N \tag{9}$$

In Eq. (9) $\omega = 2\pi/T$ is the angular frequency, $T \approx .355$ is the final non-dimensional time, $\mathbf{x}_i^n$ the location of grid-point i at time level n and $\mathbf{x}_v$ the initial location of the vortex core, which is set equal to the origin of the Cartesian reference frame. Choosing $B_1 = (1.5\omega)^{-1}$ in Eq. (9) makes the maximum grid velocity of the same order of magnitude of the un-disturbed free-stream flow. Figures 5(a) and 5(b) show the grid velocity magnitude and the coarsest (level 1) triangulation at $t = T/4$ and $t = T$, respectively. At these two time instants the grid velocity attains its maximum, resp. minimum value (different contour levels are used in the two frames).

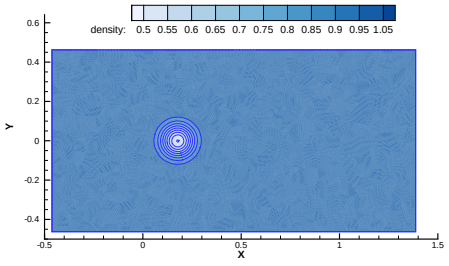
One calculation has been conducted for each of the five grid levels of Tab. I. The simulation settings are reported in Table II; observe that the time step length, Δt, has been recursively halved when passing from the coarsest grid level to the finer one and the number of time steps, N, doubled. Figures 5(c) and 5(d) show the density iso-contours on grid level 4 when $t = T/2, T$, respectively. These two time instants correspond to the minimum, resp. maximum size of the computational domain.



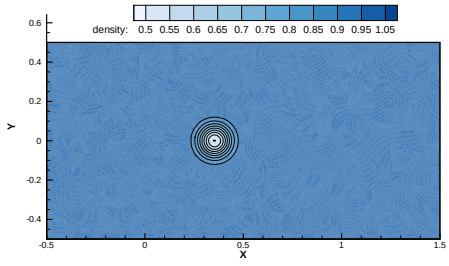
(a) Grid velocity magnitude and level 1 triangulation at $t = T/4$.



(b) Grid velocity magnitude and level 1 triangulation at $t = T/2$.



(c) Density iso-contours on the level 4 triangulation at $t = T/2$.



(d) Density iso-contours on the level 4 triangulation at $t = T$.

Figure 5. Inviscid vortex convected by a supersonic stream.

DOI: 10.1002/fld

Table II. Inviscid vortex convected by a supersonic stream: L_2-norm of the discretization error and measured order of convergence.

grid level	Δt	N	$\sqrt{\rho}$ L_2	$\tilde{p}$	$\sqrt{\rho}H$ L_2	$\tilde{p}$	$\sqrt{\rho}u$ L_2	$\tilde{p}$	$\sqrt{\rho}v$ L_2	$\tilde{p}$
1	.0098601	36	0.9065E-02	-	0.2258E-01	-	0.1747E-01	-	0.2241E-01	-
2	.0049300	72	0.3159E-02	1.52	0.7871E-02	1.52	0.6082E-02	1.52	0.7831E-02	1.52
3	.0024650	144	0.8820E-03	1.84	0.2180E-02	1.85	0.1702E-02	1.84	0.2273E-02	1.79
4	.0012325	288	0.2329E-03	1.92	0.5678E-03	1.94	0.4335E-03	1.97	0.5867E-03	1.95
5	.0006163	576	0.6144E-04	1.92	0.1451E-03	1.97	0.1092E-03	1.99	0.1481E-03	1.99

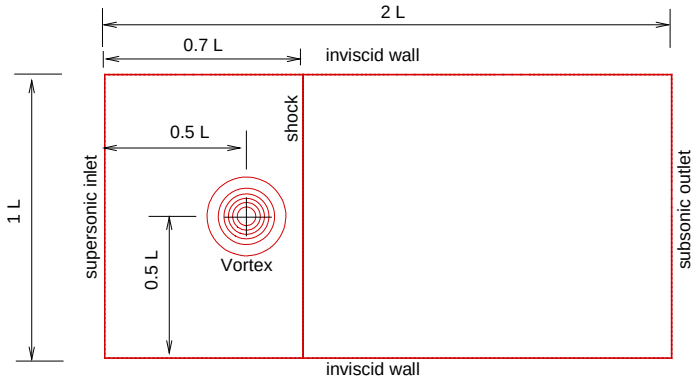


Figure 6. Shock-vortex interaction ($M_s = 1.21$, $M_v = 0.3$): computational domain and boundary conditions.

Denoting by u_0 and u_k, the exact, resp. numerical solution (on grid level k) of the governing PDEs, a global measure of the rate at which the discretization error $e_k = u_0 - u_k$ decreases as the mesh is refined can be estimated as follows:

$$\tilde{p} = \log\left(I_{k+1}/I_k\right) / \log\left(h_{k+1}/h_k\right) \tag{10}$$

where I_k is an integral measure of the L_2-norm of the discretization error (at the final time T) over the computational domain $\Omega\left(T\right)$:

$$I_k = \left(\frac{1}{|\Omega\left(T\right)|} \int_{\Omega} e_k^2 \, \mathrm{d}\Omega \right)^{1/2} \tag{11}$$

and the ratio h_{k+1}/h_k takes the constant value 2 when triangular meshes are refined by recursive subdivision as described above. Table II shows the L_2-norm of the discretization error at the final time for each component of Roe's parameter vector along with the global measure $\tilde{p}$ of the order-of-convergence computed according to Eq. (10) for each pair of consecutive grid levels; it can be seen that design order is recovered.

3.2. *Shock-vortex interaction*

This second test case consists in the interaction between a stationary shock and a vortex and provides a useful testbed for comparing shock-capturing versus shock-fitting. It has been frequently reported in the literature, not only as a code verification case [30, 27], but primarily as a tool for understanding the fundamental mechanisms [31, 32, 33, 34] that account for noise generation due to the interaction between a shock-wave and a turbulent flow. The computational domain, which is sketched in Fig. 6 along with the boundary conditions applied on its boundaries, is the rectangle $[0, 2L] \times [0, L]$. The computational domain has been discretised using a sequence of nested Delaunay triangulation. The coarsest mesh has been generated using the Triangle code [17] and all other grid levels have been obtained by recursive subdivision of the coarsest one. Table III summarises the characteristics of the

Table III. Shock-vortex interaction ($M_s = 1.21, M_v = 0.3$): characteristics of the various meshes used.

Grid level, k	Grid points	Triangles	h/L	Δt
1	3481	6776	0.025	$2.0\ 10^{-4}$
2	13737	27104	0.0125	$1.0\ 10^{-4}$
3	54577	108416	0.00625	$0.5\ 10^{-4}$

various meshes that have been used both in the shock-capturing calculation and as "background" triangulations in the shock-fitting calculation; the symbol h refers to the linear mesh spacing along each of the boundaries.

A uniform, supersonic stream, characterised by a shock-upstream Mach number $M_s = 1.21$, carries a vortex, from the left to the right of Fig. 6, towards a stationary normal shock. At the initial time, $t = 0$, the vortex is centred in $(x_v/L, y_v/L) = (0.5, 0.5)$ and the shock is located $0.2L$ downstream of the vortex centre. We have used the same vortical structure already described in Sect. 3.1 and defined by Eq. (8). Using the following choice of parameters: $r_0/L = 0.15$, $\alpha = 0.204$, $\epsilon \approx 0.21$ (which are different from those used in Sect. 3.1) gives a vortex Mach number:

$$M_v = \frac{\max\left(\tilde{u}_\theta\right)}{a_\infty} = M_s \epsilon \left(2\alpha\right)^{-\frac{1}{2}} e^{\left(\alpha - \frac{1}{2}\right)} \approx 0.296, \tag{12}$$

where a_∞ is the sound speed of the free-stream, shock-upstream flow.

The topological pattern that arises once the vortex impinges on the shock depends upon the shock and vortex strengths. We shall hereafter refer to the taxonomy adopted by Grasso and Pirozzoli [34], who define *weak* shock-vortex interactions as those that do not exhibit any shock reflection and *strong* shock-vortex interactions as those that feature reflected and diffracted shocks; strong interactions can be further classified depending on whether a reflection or Mach reflection occurs. The same authors identify in the (M_s, M_v) couple the dimensionless parameters that govern the interaction: when the vortex Mach number is sufficiently low, weak interactions always occur; however, at a given shock strength, an increase in the vortex Mach number above the $M_v \approx 0.1 \div 0.2$ threshold causes the shock to fold thus producing reflected and diffracted shocks that yield either a regular or a Mach reflection, depending upon the value of M_v.

According to the aforementioned classification, a Mach reflection is expected (as sketched in Fig. 7) for the (M_s = 1.21, M_v = 0.3) couple used in the present numerical calculation.

Simulations have been advanced up to a final, non-dimensional time $T = 0.5$, when the vortex has travelled about two vortex radii downstream of the standing shock.

A pointwise grid convergence analysis was carried out, as described in Ref. [35], using the solutions computed on the three mesh levels. Denoting by p the order of the leading error term in the series representation of the discretization error e_k:

$$e_k = u_0 - u_k = g_p h_k^p + \mathcal{O}\left(h_k^{p+1}\right), \tag{13}$$

the measured order p can be estimated pointwise as follows [36]:

$$p = \frac{\log\left(1/R\right)}{\log 2} \tag{14}$$

within all grid-points of the coarsest mesh level 1, using the numerical solutions available on all three grid levels of Tab. III. In Eq. (14), R is the so-called grid convergence monitor:

$$R = \frac{u_2 - u_1}{u_3 - u_2} = \frac{e_1 - e_2}{e_2 - e_3} \tag{15}$$

DOI: 10.1002/fld

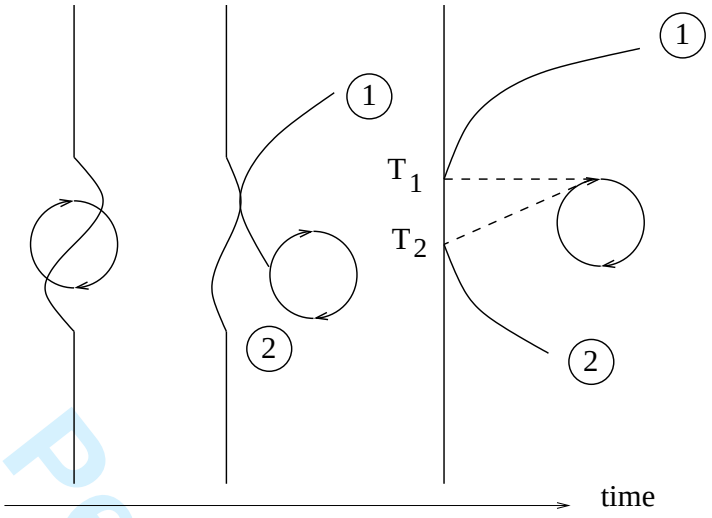


Figure 7. Evolution of the shock wave following the interaction with the vortex: 1 and 2 are the reflected shocks, T_1 and T_2 the triple points and the dashed lines represent the contact discontinuities; time increases from left to right.

which allows to classify the convergence behaviour within the grid-points as follows:

$$\begin{array}{ll} \text{Monotone convergence} & 0 \leq R \leq 1 \\ \text{Oscillatory convergence} & -1 < R < 0 \\ \text{Oscillatory Divergence} & R < -1 \\ \text{Monotonic Divergence} & R > 1 \end{array} \tag{16}$$

The convergence analysis was limited to those regions shown in Fig. 8, in order to exclude areas of uniform flow, both upstream and downstream of the shock, where the discretization error is of the order of machine accuracy.

Table IV summarises the results of the pointwise grid convergence analysis; its columns report: the percentage of monotonically converging grid-points, followed by the average measured order $< p >$ and its standard deviation $\sigma\,(p)$, both calculated only within the subset of monotonically converging grid-points; the percentage of grid-points that experience oscillatory convergence, followed by the averaged measured order $< p^* >$, computed by taking the absolute value of the convergence ratio R in Eq. (15); finally, the last two columns report the percentage of grid-points in which the discretization error increases as the mesh is refined.

The following observations can be made:

- At the earliest time $t = 0.3$, nearly half of the grid-points experience monotonic convergence at design order.
- As time progresses, the percentage of grid-points where convergence is monotonic slightly decreases and, correspondingly, the percentage of grid-points where convergence is oscillatory increases. The standard deviation $\sigma\,(p)$ also increases with time. These behaviour is probably due to the appearance of the secondary reflected shocks T_1 and T_2 of Fig. 7 that are captured, rather then fitted.
- At all times, the discretization error increases ($|R| > 1$) with mesh refinement within about 30% of the grid-points.

The spatial distribution of the observed order-of-convergence for the dependent variable $\sqrt{\rho}$ is displayed in Fig. 8 for all three time levels. Figure 8 clearly shows the presence of wide regions where the observed order-of-convergence matches design order. The regions where the observed order is larger or smaller than design order form circular patterns centred in the vortex core. These patterns can be viewed as the footprint of the acoustic waves generated by the shock-vortex

DOI: 10.1002/fld

Table IV. Shock-vortex interaction ($M_s = 1.21, M_v = 0.3$): pointwise convergence analysis for the shock-fitting calculation using the LW scheme at three time instants.

	$0 < R < 1$	$< p >$	$\sigma(p)$	$-1 < R \leq 0$	$< p^* >$	$R > 1$	$R < -1$
			$t = 0.3$				
$\sqrt{\rho}$	46.8 %	1.851	1.376	24.7 %	2.190	15.5 %	13.0 %
$\sqrt{\rho}H$	47.6 %	1.871	1.349	26.1 %	2.152	13.6 %	12.7 %
$\sqrt{\rho}u$	53.9 %	2.015	1.522	20.7 %	2.037	14.7 %	10.7 %
$\sqrt{\rho}v$	53.0 %	1.815	1.286	23.4 %	2.353	15.4 %	8.2 %
			$t = 0.4$				
	$0 < R < 1$	$< p >$	$\sigma(p)$	$-1 < R \leq 0$	$< p^* >$	$R > 1$	$R < -1$
$\sqrt{\rho}$	38.9 %	1.937	1.545	35.2 %	2.171	13.0 %	13.0 %
$\sqrt{\rho}H$	40.0 %	1.923	1.560	36.4 %	2.282	12.1 %	11.5 %
$\sqrt{\rho}u$	49.3 %	2.112	1.592	31.0 %	2.267	11.8 %	7.9 %
$\sqrt{\rho}v$	45.9 %	1.938	1.573	34.3 %	2.501	11.5 %	8.2 %
			$t = 0.5$				
	$0 < R < 1$	$< p >$	$\sigma(p)$	$-1 < R \leq 0$	$< p^* >$	$R > 1$	$R < -1$
$\sqrt{\rho}$	35.3 %	1.877	1.643	34.2 %	2.114	16.5 %	13.9 %
$\sqrt{\rho}H$	38.3 %	1.969	1.630	33.4 %	2.130	16.0 %	12.3 %
$\sqrt{\rho}u$	38.3 %	2.123	1.639	33.4 %	2.306	15.6 %	12.7 %
$\sqrt{\rho}v$	42.0 %	1.968	1.552	30.6 %	2.346	17.8 %	9.6 %

interaction at previous times. This observation is in line with the same kind of finding made in Ref. [35] in connection with steady supersonic flows. Indeed, in the supersonic case the iso-contour lines of the observed order-of-convergence approximately follow the characteristic lines whereas in the present unsteady case they follow the intersections between the characteristic cone (in the x, y, t space) and planes at constant time.

A qualitative comparison between the capturing and fitting modelling practices will now be given. Figure 9 shows the pressure iso-contours at three different time levels computed using: *i*) the explicit, shock-capturing LW scheme (LW-SC), *ii*) the same scheme, but with the main shock fitted (LW-SF) and *iii*) the time-implicit shock-capturing MM-PG-LDA scheme (LDA-SC) described in Ref [20]. The two shock-capturing solutions were computed on the same stationary grid, identical to the background mesh used in the shock-fitting calculation. The comparison between the shock-capturing and shock-fitting solutions using the same LW scheme shows the significant improvement in solution quality that shock-fitting allows to achieve for a given discretization scheme. In particular, shock-fitting makes second-order-accurate, linear, non-monotone schemes, such as the LW scheme, usable in the simulation of flows with shocks.

A considerably better shock-capturing solution, see Figs. 9(g)-9(i), has been obtained using the time-implicit MM-PG-LDA scheme; even so, the quality of the MM-PG-LDA solution is not as good as that computed by shock-fitting and the LW scheme. Indeed, the MM-PG-LDA solution shows oscillations in the iso-lines whose origin is clearly numerical and un-physical. It should be mentioned that the coupling between this scheme and the shock-fitting algorithm is not possible at present because the current shock-fitting implementation is explicit in time.

The entropy iso-contours shown in Fig. 10 turn out to be particularly revealing in highlighting the differences between the shock-fitting and shock-capturing calculations. The shock-fitting solution, displayed in Figs. 10(d)-10(f), shows entropy variations only in the region where the vortex impinges on the shock. These entropy gradients are caused by the spatially variable oblique shock angle and hence shock intensity. By contrast, spurious entropy disturbances are present in the shock-capturing solutions also in regions where the shock is straight. These disturbances pollute the downstream region in the MM-PG-LDA calculation, Figs. 10(g)-10(i), and both the upstream and downstream regions, Figs. 10(a)-10(c), in the LW solution, since the latter scheme is not entirely upwind.

4. CONCLUSIONS

The shock-fitting technique for unstructured grids that the authors have been developing over the last few years has so far been successfully applied to two- and three-dimensional steady flows featuring interacting shocks; in the present paper it has been successfully extended to unsteady flows characterised by a fixed shock topology.

In all applications tested so far, this new technique has shown huge potential benefits with respect to the shock-capturing approach and a reduced algorithmic complexity with respect to the shock-fitting techniques developed in the 70s and 80s within the structured grid framework. Indeed, the present shock-fitting technique has completely bypassed the dichotomy that characterised the two different shock-fitting approaches used in the structured mesh context: the boundary shock-fitting approach, easy to implement, but limited to simple flow topologies and the floating shock-fitting approach, more versatile, but extremely complex to code. In an unstructured grid framework all fitted shocks and all interaction points are treated as interior boundaries, regardless of the flow complexity.

In spite of this success, we are aware that much more work is needed in order to obtain an unsteady shock-fitting solver can be as general-purpose as any common shock-capturing solver. The major challenge that needs to be faced to further pursue the development of the proposed unstructured shock-fitting technique consists in managing the topological changes that typically occur in an unsteady flow, such as the formation of new shocks due to the coalescence of compression waves or the interaction of shocks with solid surfaces or other shocks. The shock-fitting technique must be capable of explicitely modeling all these topological changes, a capability which is not needed at all when using a shock-capturing approach. This striking difference is the key feature that allows to understand in which respect these two modeling approaches are different, why shock-fitting always gives better performance and, finally, why shock-fitting is very expensive in terms of modelling and programming effort. Put it simply, “fitting” means: understanding, modeling and, finally, simulating, whereas “capturing” means simulating, then, eventually, understanding.

This work is dedicated to the memory of Gino Moretti, the great researcher who has largely contributed to the development and popularity of the shock-fitting technique. Gino Moretti passed away on March 15th 2015.

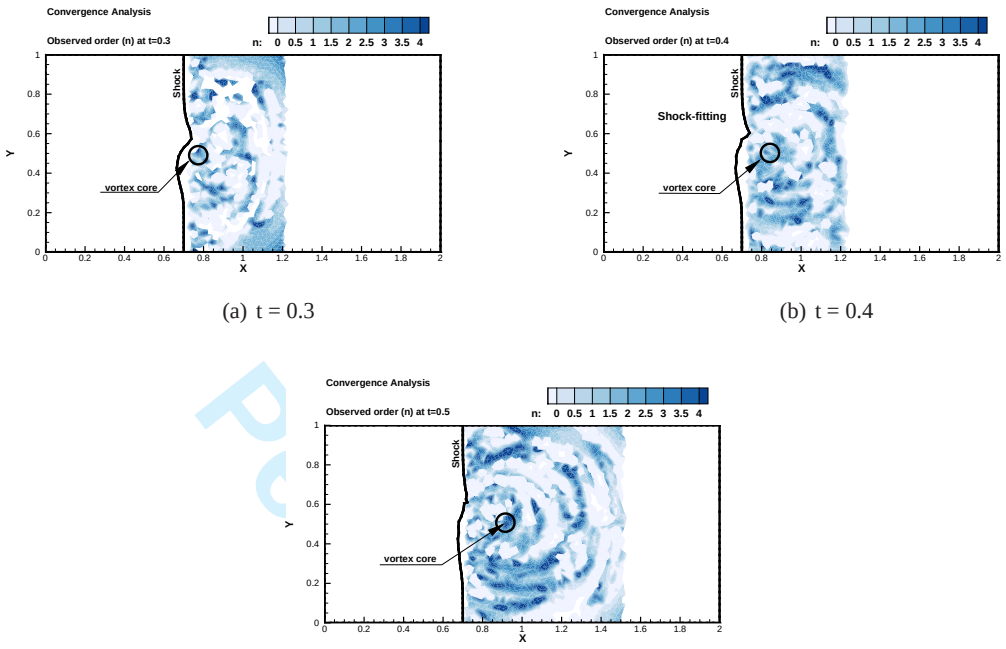
REFERENCES

1. Johnsen E, Larsson J, Bhagatwala AV, Cabot WH, Moin P, Olson BJ, Rawat PS, Shankar SK, Sjögreen B, Yee H, *et al.*. Assessment of high-resolution methods for numerical simulations of compressible turbulence with shock waves. *Journal of Computational Physics* 2010; **229**(4):1213 – 1237, doi:DOI:10.1016/j.jcp.2009.10.028. URL http://www.sciencedirect.com/science/article/B6WHY-4XH5MS8-5/2/78a62583706d9d63ab6a070b68e362ac
2. Pirozzoli S. Numerical methods for high-speed flows. *Annual Review of Fluid Mechanics* 2011; **43**(1):163–194, doi:10.1146/annurev-fluid-122109-160718. URL http://www.annualreviews.org/doi/abs/10.1146/annurev-fluid-122109-160718.
3. Arora M, Roe PL. On postshock oscillations due to shock capturing schemes in unsteady flows. *Journal of Computational Physics* 1997; **130**(1):25 – 40, doi:http://dx.doi.org/10.1006/jcph.1996.5534. URL http://www.sciencedirect.com/science/article/pii/S0021999196955345.
4. Zaide D, Roe P. Shock capturing anomalies and the jump conditions in one dimension. *Fluid Dynamics and Co-located Conferences*. American Institute of Aeronautics and Astronautics, 2011; –, doi:10.2514/6.2011-3686. URL http://dx.doi.org/10.2514/6.2011-3686.
5. Zhong X. Leading-edge receptivity to free-stream disturbance waves for hypersonic flow over a parabola. *Journal of Fluid Mechanics* 2001; **441**(-1):315–367, doi:10.1017/S0022112001004918. URL http://journals.cambridge.org/action/displayAbstract?fromPage=online&aid=82987&fulltextType=RA&
6. Sesterhenn J, Dohogne JF, Friedrich R. Direct numerical simulation of the interaction of isotropic turbulence with a shock wave using shock-fitting. *Comptes Rendus Mcanique* 2005; **333**(1):87 – 94, doi:http://dx.doi.org/10.1016/j.crme.2004.09.017. URL http://www.sciencedirect.com/science/article/pii/S1631072104002347, high-order methods for the numerical simulation of vortical and turbulent flows.
7. Moretti G, Abbett M. A time-dependent computational method for blunt body flows. *AIAA Journal* 1966; **4**(12):2136–2141.
8. Salas MD. Shock fitting method for complicated two-dimensional supersonic flows. *AIAA Journal* May 1976; **14**(5):583–588, doi:10.2514/3.61399. URL http://arc.aiaa.org/doi/abs/10.2514/3.61399.
9. Moretti G. Efficient euler solver with many applications. *AIAA Journal* Jun 1988; **26**(6):655–660, doi:10.2514/3.9950. URL http://arc.aiaa.org/doi/abs/10.2514/3.9950.

10. Rawat PS, Zhong X. On high-order shock-fitting and front-tracking schemes for numerical simulation of shock-disturbance interactions. *Journal of Computational Physics* 2010; **229**(19):6744 – 6780, doi:DOI:10.1016/j.jcp.2010.05.021. URL http://www.sciencedirect.com/science/article/B6WHY-506J3V5-2/2/3b16534cd05fe731a64c947815a0ee16
11. Paciorri R, Bonfiglioli A. A shock-fitting technique for 2d unstructured grids. *Computers & Fluids* 2009; **38**(3):715 – 726, doi:10.1016/j.compfluid.2008.07.007. URL http://www.sciencedirect.com/science/article/pii/S0045793008001503.
12. Ivanov MS, Bonfiglioli A, Paciorri R, Sabetta F. Computation of weak steady shock reflections by means of an unstructured shock-fitting solver. *Shock Waves* 2010; **20**(4):271–284, doi:10.1007/s00193-010-0266-y. URL http://dx.doi.org/10.1007/s00193-010-0266-y.
13. Paciorri R, Bonfiglioli A. Shock interaction computations on unstructured, two-dimensional grids using a shock-fitting technique. *Journal of Computational Physics* 2011; **230**(8):3155 – 3177, doi:10.1016/j.jcp.2011.01.018. URL http://www.sciencedirect.com/science/article/pii/S0021999111000362.
14. Bonfiglioli A, Grottadaurea M, Paciorri R, Sabetta F. An unstructured, three-dimensional, shock-fitting solver for hypersonic flows. *Computers & Fluids* 2013; **73**(0):162–174, doi:10.1016/j.compfluid.2012.12.022. URL http://www.sciencedirect.com/science/article/pii/S0045793013000121.
15. Pepe R, Bonfiglioli A, Paciorri R, Lani A, Garicano-Mena J, Ollivier-Gooch CF. Towards a modular approach for unstructured shock-fitting. *6th European Congress on Computational Fluid Dynamics (ECFD VI)*, Onate E, Oliver J, Huerta A (eds.), International Center for Numerical Methods in Engineering: Barcelona, Spain, 2014; 7175–7186. URL http://wccm-eccm-ecfd2014.org/admin/files/filePaper/p3220.pdf.
16. Paciorri R, Bonfiglioli A. Shock interaction computations on unstructured, two-dimensional grids using a shock-fitting technique. *Journal of Computational Physics* 2011; **230**(8):3155 – 3177, doi:http://dx.doi.org/10.1016/j.jcp.2011.01.018. URL http://www.sciencedirect.com/science/article/pii/S0021999111000362.
17. Shewchuk JR. Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. *Applied Computational Geometry: Towards Geometric Engineering*, *Lecture Notes in Computer Science*, vol. 1148, Lin MC, Manocha D (eds.). Springer-Verlag, 1996; 203–222. From the First ACM Workshop on Applied Computational Geometry.
18. Salas M. *A Shock-Fitting Primer*. 1^{st} edn., CRC Applied Mathematics & Nonlinear Science, Chapman & Hall, 2009.
19. Bonfiglioli A. Fluctuation splitting schemes for the compressible and incompressible Euler and Navier-Stokes equations. *International Journal of Computational Fluid Dynamics* 2000; **14**:21–39.
20. Bonfiglioli A, Paciorri R. A mass-matrix formulation of unsteady fluctuation splitting schemes consistent with Roe's parameter vector. *International Journal of Computational Fluid Dynamics* 2013; **27**(4-5):210–227, doi:10.1080/10618562.2013.813491. URL http://www.tandfonline.com/doi/abs/10.1080/10618562.2013.813491.
21. Deconinck H, Paillère H, Struijs R, Roe P. Multidimensional upwind schemes based on fluctuation-splitting for systems of conservation laws. *Computational Mechanics* 1993; **11**(5/6):323–340.
22. van der Weide E, Deconinck H, Issman E, Degrez G. A parallel, implicit, multi-dimensional upwind, residual distribution method for the Navier-Stokes equations on unstructured grids. *Computational Mechanics* 1999; **23**:199–208.
23. Abgrall R. Residual distribution schemes: Current status and future trends. *Computers and Fluids* 2006; **35**(7):641 – 669, doi:10.1016/j.compfluid.2005.01.007. URL http://www.sciencedirect.com/science/article/pii/S0045793005001738, special Issue Dedicated to Professor Stanley G. Rubin on the Occasion of his 65th Birthday.
24. Deconinck H, Roe P, Struijs R. A Multi-dimensional Generalization of Roe's Flux Difference Splitter for the Euler Equations. *Computers and Fluids* 1993; **22**(2/3):215–222.
25. Deconinck H, Paillère H, Struijs R, Roe PL. Multidimensional upwind schemes based on fluctuation-splitting for systems of conservation laws. *Computational Mechanics* 1993; **11**:323–340. URL http://dx.doi.org/10.1007/BF00350091, 10.1007/BF00350091.
26. Struijs R. A multi-dimensional upwind discretization method for the Euler equations on unstructured grids. PhD Thesis, Technische Universiteit Delft, The Nederlands 1994.
27. Rossiello G, De Palma P, Pascazio G, Napolitano M. Second-order-accurate explicit fluctuation splitting schemes for unsteady problems. *Computers & Fluids* 2009; **38**(7):1384 – 1393, doi:DOI:10.1016/j.compfluid.2008.01.021. URL http://www.sciencedirect.com/science/article/pii/S0045793008000674, special Issue Dedicated to Professor Alain Lerat on the Occasion of his 60th Birthday.
28. Ricchiuto M, Abgrall R. Explicit Runge-Kutta residual distribution schemes for time dependent problems: Second order case. *Journal of Computational Physics* 2010; **229**(16):5653 – 5691, doi:10.1016/j.jcp.2010.04.002. URL http://www.sciencedirect.com/science/article/pii/S0021999110001786.
29. Michler C, De Sterck H, Deconinck H. An arbitrary lagrangian eulerian formulation for residual distribution schemes on moving grids. *Computers & Fluids* 2003; **32**(1):59 – 71, doi:10.1016/S0045-7930(01)00095-0. URL http://www.sciencedirect.com/science/article/pii/S0045793001000950.
30. Abgrall R, Mezine M. Construction of second order accurate monotone and stable residual distribution schemes for unsteady flow problems. *Journal of Computational Physics* 2003; **188**(1):16 – 55, doi:http://dx.doi.org/10.1016/S0021-9991(03)00084-6. URL http://www.sciencedirect.com/science/article/pii/S0021999103000846.
31. Pao S, Salas M. A numerical study of two-dimensional shock vortex interaction. *Fluid Dynamics and Colocated Conferences*. American Institute of Aeronautics and Astronautics, 1981; –, doi:10.2514/6.1981-1205. URL http://dx.doi.org/10.2514/6.1981-1205.
32. Meadows KR, Kumar A, Hussaini M. Computational study on the interaction between a vortex and a shock wave. *AIAA Journal* Feb 1991; **29**(2):174–179, doi:10.2514/3.59916. URL

http://dx.doi.org/10.2514/3.59916.

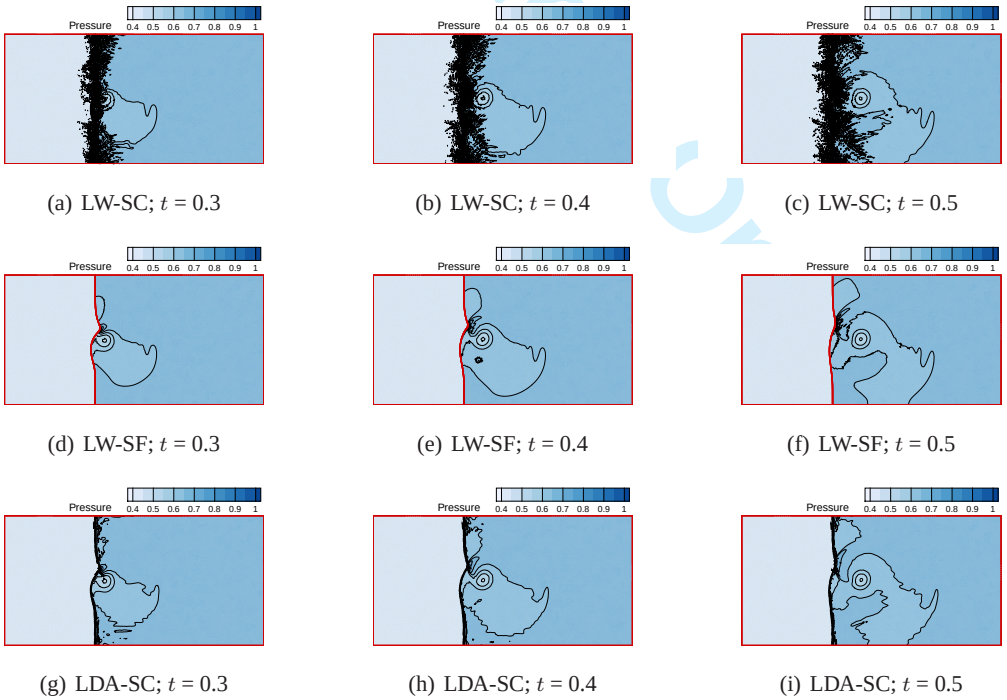
33. Inoue O, Hattori Y. Sound generation by shockvortex interactions. *Journal of Fluid Mechanics* 1999; **380**:81 – 116, doi:doi:10.1017/S0022112098003565.
34. Grasso F, Pirozzoli S. Shock-wavevortex interactions: Shock and vortex deformations, and sound production. *Theoretical and Computational Fluid Dynamics* 2000; **13**(6):421–456, doi:10.1007/s001620050121. URL http://dx.doi.org/10.1007/s001620050121.
35. Bonfiglioli A, Paciorri R. Convergence analysis of shock-capturing and shock-fitting solutions on unstructured grids. *AIAA Journal* 2014; **52**(7):1404–1416. URL http://arc.aiaa.org/doi/abs/10.2514/1.J052567.
36. Roy CJ. Grid Convergence Error Analysis for Mixed-Order Numerical Schemes. *AIAA Journal* April 2003; **41**(4):595–604.



(a) t = 0.3 (b) t = 0.4

(c) t = 0.5

Figure 8. Shock-vortex interaction ($M_s = 1.21, M_v = 0.3$): pointwise measured order of $\sqrt{\rho}$ at different time levels, Lax-Wendroff shock-fitting calculation.



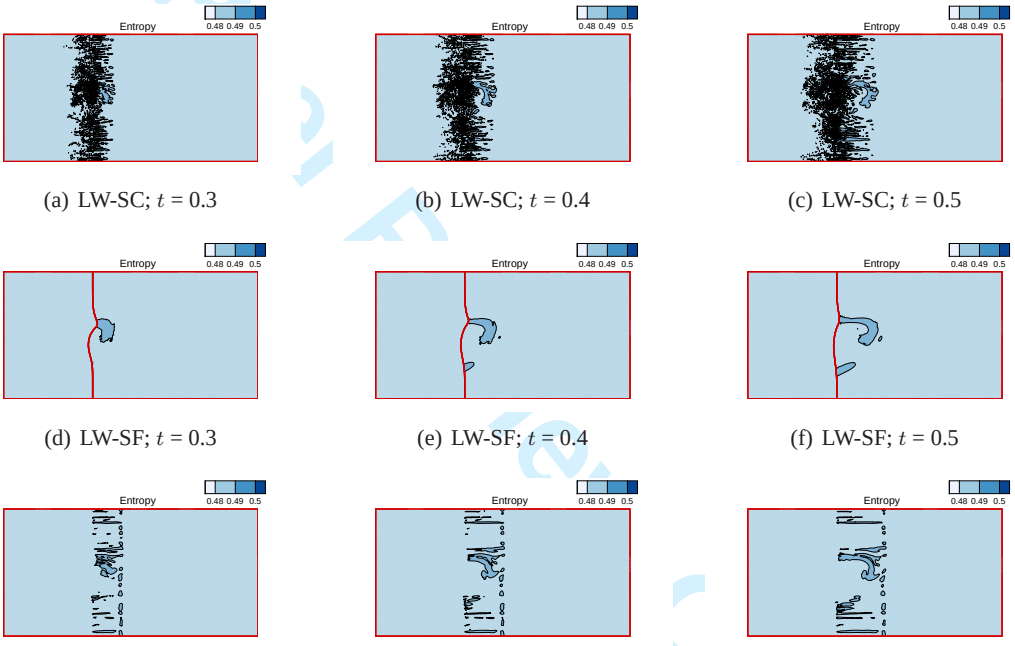
(a) LW-SC; t = 0.3 (b) LW-SC; t = 0.4 (c) LW-SC; t = 0.5

(d) LW-SF; t = 0.3 (e) LW-SF; t = 0.4 (f) LW-SF; t = 0.5

(g) LDA-SC; t = 0.3 (h) LDA-SC; t = 0.4 (i) LDA-SC; t = 0.5

Figure 9. Shock-vortex interaction ($M_s = 1.21, M_v = 0.3$): pressure iso-contours at different time levels $t = 0.3, 0.4, 0.5$

DOI: 10.1002/fld



(a) LW-SC; $t = 0.3$ (b) LW-SC; $t = 0.4$ (c) LW-SC; $t = 0.5$

(d) LW-SF; $t = 0.3$ (e) LW-SF; $t = 0.4$ (f) LW-SF; $t = 0.5$

(g) LDA-SC; $t = 0.3$ (h) LDA-SC; $t = 0.4$ (i) LDA-SC; $t = 0.5$

Figure 10. Shock-vortex interaction ($M_s = 1.21, M_v = 0.3$): entropy iso-contours at different time levels $t = 0.3, 0.4, 0.5$

DOI: 10.1002/fld