Extrapolated Shock Fitting for Two-Dimensional Flows on Structured Grids

Alessia Assonitis, Renato Paciorri, Mirco Ciallella, Mario Ricchiuto, Aldo Bonfiglioli

Extrapolated Shock Fitting for Two-Dimensional Flows on Structured Grids

Alessia Assonitis* and Renato Paciorri†
University of Rome La Sapienza, 00184 Rome, Italy
Mirco Ciallella‡ and Mario Ricchiuto§
Institut National de Recherche en Informatique et Automatique, U. Bordeaux, Centre National de la Recherche Scientifique, Bordeaux Institut national polytechnique, Insitut de Mathématiques de Bordeaux, Unité mixte de recherche 5251, 33405 Talence Cedex, France
and
Aldo Bonfiglioli¶
University of Basilicata, 85100 Potenza, Italy

https://doi.org/10.2514/1.J061893

Over the years the development of structured-grid shock-fitting techniques faced two main problems: the handling of a moving discontinuity on a fixed background grid and the capability of simulating complex flow configurations. In the proposed work, the authors present a new shock-fitting technique for structured-grid solvers that is capable of overcoming the limitations that affected the different approaches originally developed. The technique presented here removes the tight link between grid topology and shock topology, which characterizes previous shock fitting as well as front tracking methods. This significantly simplifies their implementation and more importantly reduces the computational overhead related to these geometrical manipulations. Interacting discontinuities and shocks interacting with a solid boundary are discussed and analyzed. Finally, a quantitative investigation of the error reduction obtained with the approach proposed via a global grid convergence analysis is presented.

Nomenclature

a	=	speed of sound
E	=	total energy, J/kg
H	=	total enthalpy, J/kg
h	=	mesh spacing
ℓ_i	=	length of the ith edge of a quadrilateral mesh element
M	=	Mach number
$\tilde{n}$	=	observed (or measured) order of convergence of a numerical scheme
$\boldsymbol{n}_i$	=	normal unit vector to the ith shock point or the ith edge of a mesh cell
P	=	static pressure, Pa
$\boldsymbol{r}_i$	=	position vector of the ith shock point
T_0	=	total temperature, K
$\boldsymbol{u}$	=	velocity vector, m/s
x, y	=	Cartesian coordinates
Δt	=	time step
ϵ_h	=	discretization error on a mesh of mesh spacing h
ρ	=	density, kg/m^3
ϕ	=	generic dependent variable
ϕ_h	=	discrete numerical solution computed on a mesh of mesh spacing h
Ω_A	=	area of a quadrilateral cell A

Received 29 March 2022; revision received 24 June 2022; accepted for publication 12 July 2022; published online 12 August 2022.

*Ph.D. Student, Department of Mechanical and Aerospace Engineering, Via Eudossiana 18–Rome.
†Associate Professor, Department of Mechanical and Aerospace Engineering, Via Eudossiana 18–Rome.
‡Ph.D. Student, Team CARDAMOM.
§Senior Research Scientist, Team CARDAMOM.
¶Associate Professor, Scuola di ingegneria, Via dell'Ateneo Lucano 10–Potenza.

I. Introduction

THE numerical techniques commonly used within the continuum framework to simulate flows with shock waves are essentially three: shock-capturing (S-C), shock-fitting [1] (S-F), and front-tracking [2] (F-T) methods. When using S-C discretizations the same integral conservation law form of the governing equations is discretized at all control volumes of the computational mesh, regardless of the presence of discontinuities. Because of this modeling feature, S-C discretizations are algorithmically simple, but are also plagued by a number of troubles, related to the thickness and structure of the captured shock [3,4]. The carbuncle phenomenon [5] and accuracy degradation [6] within the entire shock-downstream region are among the most striking examples of the troubles incurred by S-C methods. Despite the great efforts made by numerous researchers over the last decades to develop better S-C methods, the aforementioned numerical problems are still nowadays largely unsolved.

An alternative to S-C is offered by the S-F/F-T approaches, which not only allow to overcome most of the numerical troubles that affect S-C methods, but also provide very accurate solutions using coarser meshes than those needed to perform the same simulation using S-C. Modeling shock waves via S-F/F-T consists in explicitly identifying the shock as a line (or a surface in three-dimensional [3D]) within the flowfield and computing its motion, as well as the shock-upstream and shock-downstream flow states, according to the Rankine–Hugoniot (R-H) equations, whereas the governing Partial Differential Equations (PDEs) are discretized only within the smooth regions of the flowfield. Of course, this tracking procedure introduces several modeling and algorithmic challenges, which is why S-F/F-T methods are not as popular as the S-C ones.

S-F techniques, as conceived by Gino Moretti and his collaborators starting in the 1960s, have been originally developed within the structured-grid framework with two different variants blossoming over the years: "boundary" [1] and "floating" [7,8] S-F. In the first approach, the shock is made to coincide with one of the boundaries of the computational domain, which greatly simplifies the coding, because the enforcement of the R-H jump relations amounts to prescribe boundary conditions. Boundary S-F has, however, clear limitations in dealing with shocks that are not present ab initio in a time-dependent simulation or requires ad-hoc coding to deal with shock interactions (see, e.g., [9]). The floating S-F version [7,8] was

developed to deal with more complex flow configurations. In this approach, discontinuities can freely move over a background structured mesh and the shock fronts are described by their intersections with the grid lines. Even though the floating S-F approach allows to overcome most of the limitations incurred by the boundary S-F, it is very complex to code as it requires to modify the computational stencil in the neighborhood of the floating discontinuities to avoid taking differences across the discontinuity (see, e.g., [10]).

F-T methods [11–13], as conceived by James Glimm and collaborators, have some similarities with floating S-F. The F-T approach of [14] involved solving the flow equations on both sides of the discontinuity never using the data on the front, and used ghost states at intersections to obtain finite difference expressions to evolve the closest points. This approach has many common points with classical well-known cut cell approaches and introduces small cells, which may degrade the accuracy and stability (CFL condition) of the method (cf., e.g., [15] and references therein). Similar approaches were also followed by other researchers [16–18]. All these techniques require quite extensive local geometrical operations to side step the small cut cell issue. In 2009, Paciorri and Bonfiglioli [19] introduced a new, unstructured-grid S-F technique that takes advantage of the geometrical flexibility offered by unstructured meshes and combines features of both the boundary and floating S-F algorithms developed in the structured-grid setting. This new unstructured-grid S-F technique, which allows to overcome at least some of the difficulties incurred by the "traditional" S-F/F-T methods, has been further improved in recent years by making it capable of dealing with multiple interacting discontinuities [20], shock/boundary-layer interactions [21], and unsteady two-dimensional (2D) flows [22,23]. Most of the aforementioned developments have also been included in an open-source platform [24], whereas the extension of this technique to 3D flows is described in [25,26]. In recent years the authors' work on unstructured S-F methods sparked a renewed interest toward S-F/shock tracking techniques, which have been applied by several research teams worldwide in conjunction with different discretization techniques, including finite volume (FV) and finite element (FE) methods. The group headed by Jun Liu at Dalian University of Technology, China, developed the Mixed Capturing and Fitting Solver (MCFS) by combining an S-F algorithm with an existing S-C, cell-centered FV solver using triangular meshes [27–30]. S-F/F-T ideas made their way also through the FE community. We refer to the Streamline Upwind Petrov-Galerkin (SUPG) technique of [31] and the discontinuous Galerkin (DG) FE methods independently developed by two different research teams: [32–38]. All three aforementioned techniques simultaneously solve for the location of the grid points, in addition to the flow variables, so as to constrain certain edges of the tessellation to be aligned with the discontinuities. The use of shape functions that are continuous across the element interfaces, which is the case with SUPG, or discontinuous, such as in DG, has implications on how discontinuities are fitted. In the SUPG-FE algorithm of [31] the discontinuities are internal boundaries of zero thickness: by doing so, a finite jump in the dependent variables takes place while crossing the discontinuity. On the other hand, numerical methods that employ a data representation that is discontinuous across the cell interfaces, which is the case with DG-FEM, but also with cell-centered FV methods, allow to fit discontinuities as a collection of edges of the mesh, without introducing internal boundaries.

Regardless of the chosen approach, in all the aforementioned *new* unstructured-grid S-F/F-T techniques the mesh has to be modified, either locally or globally and in a (pseudo) time-dependent fashion, to follow the motion of the discontinuities. This may not always be an easy task from the meshing view point, especially when dealing with complicated shock interactions and/or 3D flows, in which case it may also be computationally costly.

For these reasons, the authors have recently developed the *extrapolated DIscontinuity Tracking* (eDIT) technique, which combines the unstructured-grid S-F technique developed by Paciorri and Bonfiglioli [19] with the shifted boundary method by Song et al. [39]. In eDIT [40,41] the fitted (or tracked) discontinuity carves a mesh-less hole in the computational domain over which it is floating, and data transfer between the discontinuity and the boundaries of the hole relies on extrapolation via truncated Taylor series expansions. This approach allows to retain high-order convergence properties without imposing any constraint on the topology of the mesh, and on the data structure of the underlying flow solver and without requiring complex mesh operations other than flagging the cells crossed by the discontinuity. The eDIT's features lend themselves well also to the structured-grid framework. Using this feature, in this paper we propose a new approach that we baptize *structured extrapolated shock-fitting* (SESF). One of the key features of SESF is its capability to keep track of the discontinuities and compute the surrounding smooth-flow areas without re-meshing around the discontinuity; by doing so, SESF breaks the tight link between grid topology and shock topology that troubled for many years the development of S-F methods for structured grids.

Moreover, a limited amount of extra coding is required to couple SESF with virtually any existing structured-grid computational fluid dynamics (CFD) code as long as the latter is capable of handling cell blanking, which amounts to disable the computation within a given set of cells. CFD codes dealing with overset meshes typically possess this capability. Taking advantage of cell-blanking, the CFD code can be seen as a black-box by the SESF algorithm, which only takes care of supplying an input solution and the list of blanked cells. This algorithmic flexibility greatly simplifies the coding effort compared to the "traditional" floating S-F technique, while retaining the capability to handle flows featuring complex shock patterns. Therefore, the SESF technique can be considered as an important achievement because structured-grid solvers are the ones that can take the greatest advantage of S-F modeling. Indeed, this class of solvers is still very much in use today for simulating turbulent and aero-acoustic flows (via DNS or LES) because they are computationally more efficient and accurate than the unstructured-grid ones. Therefore, these simulations could really benefit from a more accurate and efficient shock-modeling, free of all those problems associated with the S-C process.

II. Generalities

We consider the numerical approximation of solutions of the steady limit of the Euler equations in two dimensions reading

$$\partial_t \boldsymbol{U} + \nabla \cdot \boldsymbol{F} = 0 \quad \text{in} \quad \Omega \subset \mathbb{R}^2 \tag{1}$$

with conserved variables and fluxes given by

$$\boldsymbol{U} = \begin{bmatrix} \rho \\ \rho \boldsymbol{u} \\ \rho E \end{bmatrix}, \qquad \boldsymbol{F} = \begin{bmatrix} \rho \boldsymbol{u} \\ \rho \boldsymbol{u} \otimes \boldsymbol{u} + P\mathbb{I} \\ \rho H \boldsymbol{u} \end{bmatrix} \tag{2}$$

In this paper, we work with the classical perfect gas equation of state:

$$p = (\gamma - 1)\rho e \tag{3}$$

with γ the constant (for a perfect gas) ratio of specific heats.

In general, in d space dimensions, discontinuities are represented by $d-1$ manifolds governed by the well-known R-H jump conditions reading

$$[\![\boldsymbol{F}]\!] \cdot \boldsymbol{n} = w[\![\boldsymbol{U}]\!] \tag{4}$$

having denoted by $\boldsymbol{n}$ the local unit vector normal to the shock, by $[\![\cdot]\!]$ the corresponding jump of a quantity across the discontinuity, and with w the normal component of the shock speed.

III. Structured Extrapolated Shock-Fitting Algorithm

To illustrate the algorithmic features of the SESF (see Fig. 1), we consider a 2D computational domain and a compressible flowfield featuring shocks and contact discontinuities. For the purpose of illustrating the algorithm, reference will be made to a shock wave, but contact discontinuities can also be handled by using the appropriate jump relations, and a simulation involving a contact discontinuity will be presented in Sec. IV.C. The example of Fig. 2a shows the

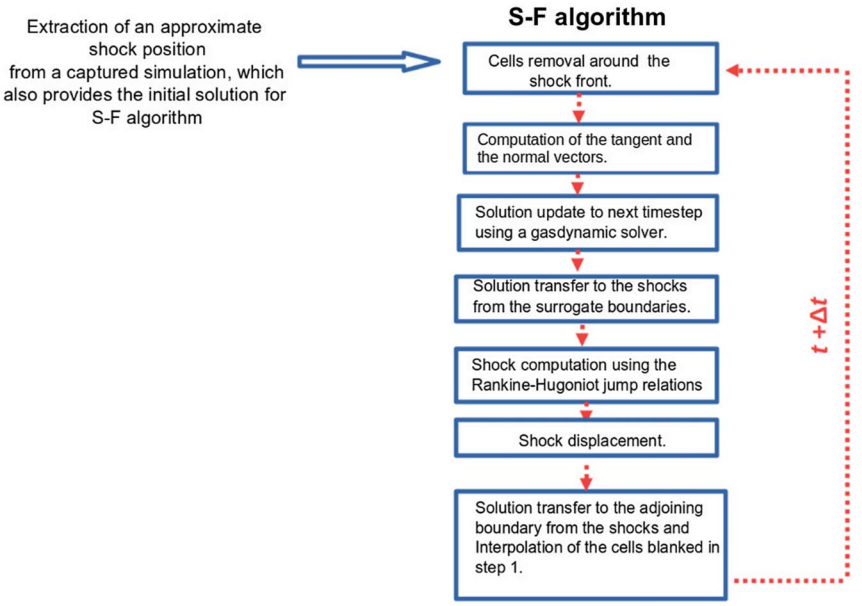
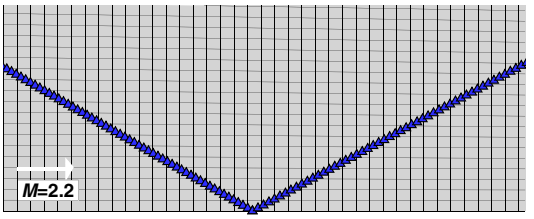
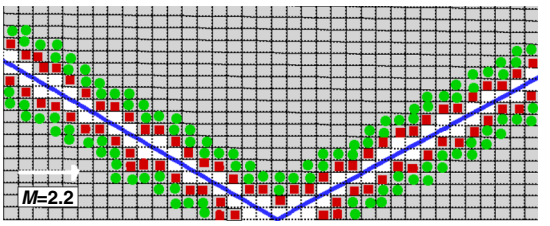


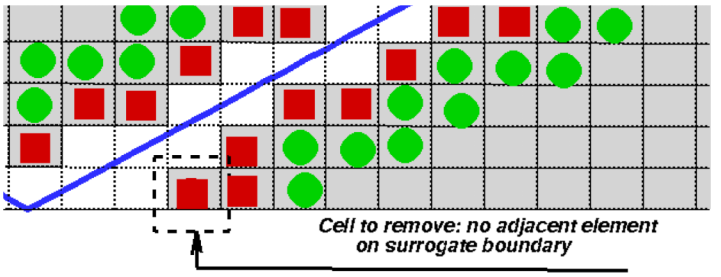
Fig. 1 SESF algorithm flowchart.



a) Regular reflection on a plate: background Cartesian grid and shocks (marked by triangular points)



b) Regular reflection on a plate: an example of computational mesh definition



c) Regular reflection on a plate: check on cavity boundary

Fig. 2 Regular reflection on a flat plate ($M = 2.2$): creation of the computational grid.

Cartesian grid used to simulate a regular shock reflection from a flat plate. We run first an S-C calculation that supplies the initial solution and approximate shock location to the SESF simulation. SESF makes use of both a *background* mesh, which coincides with the grid used in the S-C calculation, and a *shock mesh*. In 2D the shock mesh is made up of a collection of points, the *shock points*, marked with triangles in Fig. 2a, which are mutually joined to form an ordered sequence of connected straight segments, the *shock edges*. In 3D (see [42] for recent developments), the shock mesh is a triangulated surface. Regardless of the dimensionality of the physical space, a single set of dependent variables is associated with each cell of the background mesh, whereas two sets of values, corresponding to the upstream and downstream states, are assigned to each shock point. The initial shape of the shock front is obtained by postprocessing the S-C solution by means of an automatic shock-detection technique, such as the ones described in [43,44] or [29], which are applicable also to 3D flows. Contact discontinuities can also be detected by choosing the appropriate sensor. When dealing with steady flows, it is not important that the initial shock shape and the initial values of the dependent variables within the shock points are accurately computed, because by enforcing the R-H jump relation across each pair of shock points the algorithm is capable of driving the shock toward its steady-state location.

The flowchart in Fig. 1 shows the seven algorithmic steps that allow to advance the solution and shock position from time t to $t + \Delta t$. We assume that at time t the dependent variables are known within all cells of the background mesh and all shock points of the shock mesh. A detailed description of each step will be given in the following paragraphs.

A. Step 1: Cell Removal Around the Shock Front

The first step consists in removing all those cells of the background mesh that are crossed by the discontinuity. More precisely, the algorithm checks for intersections between the edges of the background mesh and the shock edges. Wherever this occurs, the two cells sharing the edge are blanked. By doing so, a hole enclosing the discontinuity is carved within the background mesh, as shown in Fig. 2b. We shall hereafter call *computational* mesh the background mesh with the

blanked cells removed. Figure 2b also points out two important features of the proposed SESF technique: i) the cells marked by red squares represent the *adjoining boundary*, which is the collection of cells that have at least an adjacent blanked cell, and ii) the cells marked with green circles form the *surrogate boundary*, which is made up of those cells sharing at least one vertex with the cells on the adjoining boundary. Some further geometrical processing might be required to build the computational mesh from the background one. This is because, to be able to perform the extrapolation described in steps 5 and 7, the cells on the adjoining boundary need to be adjacent to at least one cell on the surrogate boundary. This may not be the case, however, in regions where different shocks mutually interact or a shock reaches a boundary: for example, the red cell highlighted by the dashed line in Fig. 2c has no neighbors belonging to the surrogate boundary and is therefore flagged for removal.

B. Step 2: Computation of Tangent and Normal Vectors

To apply the R-H jump relations, the tangent, $\boldsymbol{\tau}$, and normal, $\boldsymbol{n}$, unit vectors have to be calculated within each shock point. The computation of $\boldsymbol{\tau}_i$, where the subscript i refers to the numbering of shock points, relies on finite difference formulas that involve the coordinates of shock point i and those of its neighboring shock points. By reference to Fig. 3, $\boldsymbol{r}_i$ denotes the position of shock point i at time level t. Shock points $i-1$ and $i+1$ are located on the two sides of shock point i and their position $\boldsymbol{r}_{i-1}$ and $\boldsymbol{r}_{i+1}$ at time level t will be used to compute the tangent and normal unit vectors in shock point i. Depending on the local flow regime, it may be necessary to use upwind-biased formulas to avoid the appearance of geometrical instabilities along the fitted discontinuities. The criterion used to assess which of the shock points neighboring i falls in its range of influence is described in [19]. In this respect, three different situations may arise:

1) Both shock points $i-1$ and $i+1$ are in the range of influence of shock point i.

2) Shock point $i+1$ is outside the range of influence of shock point i.

3) Shock point $i-1$ is outside the range of influence of shock point i.

When case 1 applies, the computation of $\boldsymbol{\tau}_i$ must involve the shock points on both sides; therefore,

$$\boldsymbol{\tau}_i = \frac{l_{i+(1/2)}}{l_{i-(1/2)}(l_{i-(1/2)}+l_{i+(1/2)})}\boldsymbol{\Delta r}_{i-(1/2)} + \frac{l_{i-(1/2)}}{l_{i+(1/2)}(l_{i-(1/2)}+l_{i+(1/2)})}\boldsymbol{\Delta r}_{i+(1/2)} \tag{5a}$$

where $l_{i-(1/2)} = |\boldsymbol{\Delta r}_{i-(1/2)}|$.

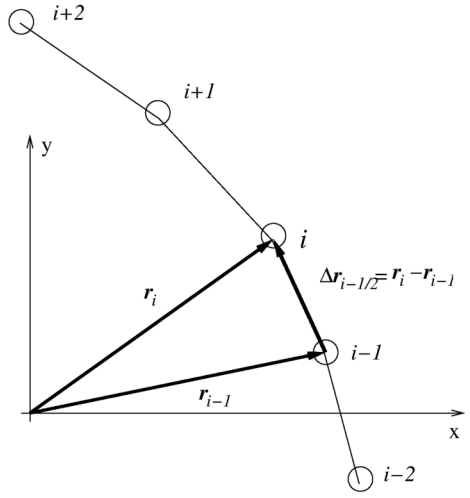


Fig. 3 Details of the shock-front geometry used in the tangent vector calculation at shock point i.

When case 2 applies, shock point $i+1$ must not be used in the computation of the tangent vector $\boldsymbol{\tau}$, and the upwind-biased formula (5b), which involves shock point $i-2$, instead of $i+1$, is used:

$$\boldsymbol{\tau}_i = \frac{l_{i-(1/2)}+l_{i-(3/2)}}{l_{i-(1/2)}l_{i-(3/2)}}\boldsymbol{\Delta r}_{i-(1/2)} - \frac{l_{i-(1/2)}}{l_{i-(3/2)}(l_{i-(1/2)}+l_{i-(3/2)})}\left(\boldsymbol{\Delta r}_{i-(1/2)}+\boldsymbol{\Delta r}_{i-(3/2)}\right) \tag{5b}$$

The third case is specular to the second one, but the corresponding formula involves shock points i, $i+1$, and $i+2$.

The grid convergence properties of the Finite Difference (FD) formulas (5) have been analyzed in [21].

Finally, the normal unit vector $\boldsymbol{n}_i$, which is perpendicular to $\boldsymbol{\tau}_i$, is chosen such that it points toward the shock-upstream region, i.e., $\boldsymbol{u}\cdot\boldsymbol{n} < 0$.

C. Step 3: Solution Update to Time $t+\Delta t$ Using a Gas-Dynamic Solver

The solution within the computational mesh is advanced in time from t to $t+\Delta t$ using a CFD solver, without imposing any boundary condition on the adjoining boundary. In the present work, a gas-dynamic CFD code developed by one of the authors [45–47]) was used: it is a second-order-accurate FV solver based on Godunov's scheme, capable of simulating inviscid and viscous 2D/3D flows using structured, multiblock meshes. The choice of the structured-grid CFD solver is independent of the SESF algorithm described in this paper, as long as it is capable of dealing with blanked cells. This capability, which is available in several commercial CFD code, has been implemented in the in-house code used in the present study.

D. Step 4: Solution Transfer from the Surrogate Boundaries to the Discontinuities

The first transfer performed by the algorithm is required to update both the shock-upstream and shock-downstream values of the dependent variables at all shock points. It is important to underline that data transfer toward the discontinuities is different depending on whether the upstream or the downstream side of the shock is considered.

The shock-upstream surrogate boundary behaves like a supersonic outflow: even though no boundary conditions are applied, the update to time $t+\Delta t$ performed by the gas-dynamic solver is correct and, therefore, extrapolation of all dependent variables from the shock-upstream surrogate boundary toward the shock upstream side of the shock mesh is used.

On the shock-downstream surrogate boundary, however, three (in the 2D space) out of the four characteristic variables are convected downstream and away from the shock, and only the Riemann variable (6) associated with the slow acoustic wave

$$R_d^{t+\Delta t} = \tilde{a}_d^{t+\Delta t} + \frac{\gamma-1}{2}\tilde{\boldsymbol{u}}_d^{t+\Delta t}\cdot\boldsymbol{n} \tag{6}$$

moves upstream toward the shock. The shock-downstream surrogate boundary behaves like a subsonic inflow boundary, and three boundary conditions would be required. Because we do not apply boundary conditions along the surrogate boundaries, the updated values computed by the CFD solver at time $t+\Delta t$ within the cells belonging to the shock-downstream surrogate boundary are incorrect [or provisional, hence the tilde in Eq. (6)], because of the three missing boundary conditions. However, due to the upwind nature of the discretization used in the CFD code, the Riemann variable (6), whose domain of dependence lies within the shock-downstream region, is correctly computed by the CFD code, even though the individual quantities $\tilde{a}_d^{t+\Delta t}$ and $\tilde{\boldsymbol{u}}_d^{t+\Delta t}$ may be wrong. It follows that on the shock-downstream side we only extrapolate the Riemann variable (6) from the shock-downstream surrogate boundary toward the downstream side of the shock. The values of the dependent variables on the downstream side of the shock points will be subsequently corrected by enforcing the R-H jump relations, as described in step 5.

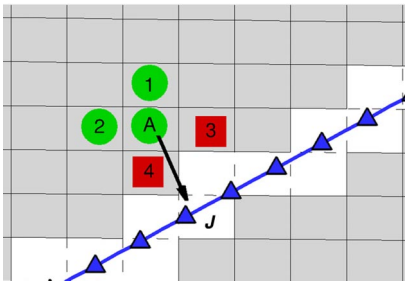


Fig. 4 Grid detail of Fig. 2b: extrapolation from the surrogate boundary to shock point j.

The extrapolation process for both transfers is based on a Taylor series expansion truncated to the second term. By reference to Fig. 4, we use

$$\phi_j = \phi_A + (\nabla\phi)_A \cdot (\boldsymbol{r}_j - \boldsymbol{r}_A) \tag{7}$$

to compute the dependent variable ϕ_j in shock point j using the cell-averaged value ϕ_A of a reference cell A that belongs to the surrogate boundary. In Eq. (7) $\boldsymbol{r}_A$ is the position vector of the centroid of cell A and $\nabla\phi$ is the gradient computed in cell A; to achieve second order of accuracy in the calculation of ϕ_j, the approximation of the gradient in Eq. (7) only needs to be consistent, i.e., first-order accurate. To select the reference cell, we first identify the cell on the adjoining boundary that is closer to shock point j; this is cell 4 in the example of Fig. 4. Then, cell A is chosen among the cells sharing a vertex with cell 4, which also belong to the surrogate boundary; when multiple choices for A are available, we take the cell that is closer to the line passing through shock point j and parallel to the shock-normal vector $\boldsymbol{n}_j$. Finally, gradient reconstruction in cell A is performed by means of the cell-based Green–Gauss formula, which reads,

$$\nabla\phi = \frac{1}{\Omega_A}\sum_{i=1}^{4}\phi_{e_i}\ell_i\boldsymbol{n}_i \tag{8}$$

In Eq. (8) the summation ranges over the four quadrilateral cells neighboring cell A, $\boldsymbol{n}_i$ is the unit vector normal to the edge shared by cells A and i, and ϕ_{e_i} the solution at the edge, which is computed using the arithmetic mean of the cell-averaged values of cells A and i:

$$\phi_{e_i} = \frac{\phi_A + \phi_i}{2} \tag{9}$$

Whenever the ith cell in Eq. (9) lies on the adjoining boundary, its ϕ_i value must be replaced using an alternative extrapolation, because the cells on the adjoining boundary will only be updated in step 7. This is the case, for instance, of cells 3 and 4 in Fig. 4 in which case ϕ_3 is extrapolated along the x axis using values ϕ_A and ϕ_2, and ϕ_4 along the y axis using ϕ_A and ϕ_1. In both cases the interpolation involves the cell-averaged value in A.

To analyze the main properties of the gradient reconstruction, a grid convergence analysis has been carried out and reported in the Appendix.

E. Step 5: Shock Computation Enforcing the Rankine–Hugoniot Jump Relations

As stated before, the shock-downstream values of the dependent variables within the shock points need to be corrected by enforcing the R-H jump relations across each pair of shock points. This consists in solving, using Newton's root-finding algorithm, the following system of five (in the 2D space) nonlinear algebraic equations:

$$\rho_d^{t+\Delta t}\left[(u_n)_d^{t+\Delta t} - w_s^{t+\Delta t}\right] = \rho_u^{t+\Delta t}\left[(u_n)_u^{t+\Delta t} - w_s^{t+\Delta t}\right] \tag{10a}$$

$$\rho_d^{t+\Delta t}\left((u_n)_d^{t+\Delta t} - w_s^{t+\Delta t}\right)^2 + p_d^{t+\Delta t} = \rho_u^{t+\Delta t}\left((u_n)_u^{t+\Delta t} - w_s^{t+\Delta t}\right)^2 + p_u^{t+\Delta t} \tag{10b}$$

$$\frac{\gamma}{\gamma-1}\frac{p_d^{t+\Delta t}}{\rho_d^{t+\Delta t}} + \frac{1}{2}\left((u_n)_d^{t+\Delta t} - w_s^{t+\Delta t}\right)^2 = \frac{\gamma}{\gamma-1}\frac{p_u^{t+\Delta t}}{\rho_u^{t+\Delta t}} + \frac{1}{2}\left((u_n)_u^{t+\Delta t} - w_s^{t+\Delta t}\right)^2 \tag{10c}$$

$$(u_\tau)_d^{t+\Delta t} = (u_\tau)_u^{t+\Delta t} \tag{10d}$$

$$a_d^{t+\Delta t} + \frac{\gamma-1}{2}(u_n)_d^{t+\Delta t} = R_d^{t+\Delta t} \tag{10e}$$

Equations (10a–10d) are the R-H jump relations, and Eq. (10e) accounts for the characteristic variable that is conveyed toward the shock from the shock-downstream region [see the discussion already made by reference to Eq. (6)]. Observe that in writing Eq. (10) the fluid velocity has been written using its components in the $(\boldsymbol{n}, \boldsymbol{\tau})$ reference frame defined in step 2. The five unknowns in Eq. (10) are the four shock-downstream values of the primitive variables (ρ, u_n, u_τ, p) and the component of the shock speed, w_s, in the shock-normal direction. This will be used in step 6 to move the shock front.

The same approach is used when dealing with contact discontinuities, except that the jump relations must be modified accordingly (see [20] for details).

F. Step 6: Shock Displacement

The new shock position at time $t + \Delta t$ is obtained by displacing all shock points using the local shock speed w_s computed in step 5 and the shock-normal unit vector $\boldsymbol{n}$ computed in step 2. When dealing with steady flows, the following first-order-accurate (in time) formula

$$\boldsymbol{r}_i^{t+\Delta t} = \boldsymbol{r}_i^t + w_{s_i}\boldsymbol{n}_i\Delta t \tag{11}$$

is used. When dealing with time-dependent flows, the temporal order of accuracy of Eq. (11) can be increased by resorting, for instance, to a predictor–corrector scheme (see [22,23]).

G. Step 7: Solution Transfer from the Shocks to the Adjoining Boundary

Once the dependent variables on the shock-downstream side of the shock mesh have been correctly updated by enforcing the R-H jump relations, they can be used to update the cell-averaged values of those cells that belong to the adjoining boundaries, as shown in Fig. 5. Once again, a generic cell A on the adjoining boundary is updated by means of a Taylor series expansion truncated to the second term using the closest shock point j, which reads

$$\phi_A = \phi_j + (\nabla\phi)_j \cdot (\boldsymbol{r}_A - \boldsymbol{r}_j) \tag{12}$$

Although Eq. (12) looks specular to Eq. (7), it is important to highlight some differences with respect to the data transfer described in

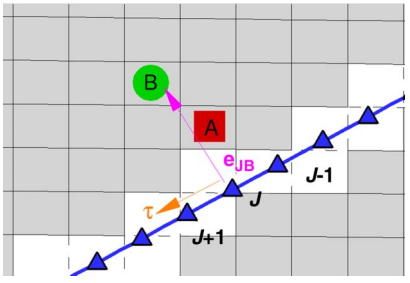


Fig. 5 Grid detail of Fig. 2b: adjoining boundary is updated using data computed in step 4 along the shock-downstream side of the shock mesh.

step 4. First of all, in the current step, data transfer takes place only on the downstream side of the shock, because the upstream side has already been updated in step 3. Secondly, the gradient reconstruction in shock point j does not rely on the Green–Gauss formula, but it is instead computed along two (not mutually orthogonal) directions, $\boldsymbol{\tau}$ and $\boldsymbol{e}_{JB}$, shown in Fig. 5. The unit vector $\boldsymbol{\tau}$ is tangent to the discontinuity and has already been computed in step 2. The unit vector $\boldsymbol{e}_{JB}$ is parallel to the direction linking shock point j to one of the cells on the surrogate boundary sharing a vertex with cell A. Also in this case, when multiple choices are available, the algorithm considers the cell that is closer to the line passing through shock point j and parallel to the shock-normal vector $\boldsymbol{n}_j$; in the example shown in Fig. 5 this is cell B. The component of the gradient along the discontinuity uses a stencil of shock points that belong to the range of influence of shock point j:

$$\nabla\phi\cdot\boldsymbol{\tau}=\begin{cases}\dfrac{\phi_{j+1}-\phi_{j-1}}{|\Delta\boldsymbol{r}_{j+1/2}+\Delta\boldsymbol{r}_{j-1/2}|} & \text{if both } j+1 \text{ and } j-1 \text{ are in the range of influence of } j\\ \dfrac{\phi_j-\phi_{j-1}}{|\Delta\boldsymbol{r}_{j-1/2}|} & \text{if only } j-1 \text{ is in the range of influence of } j\\ \dfrac{\phi_{j+1}-\phi_j}{|\Delta\boldsymbol{r}_{j+1/2}|} & \text{if only } j+1 \text{ is in the range of influence of } j\end{cases} \tag{13}$$

The gradient along $\boldsymbol{e}_{JB}$ is instead calculated using the following finite difference formula:

$$\nabla\phi\cdot\boldsymbol{e}_{JB}=\frac{\phi_B-\phi_J}{|\boldsymbol{r}_B-\boldsymbol{r}_j|} \tag{14}$$

Finally, the yet-unknown Cartesian components of the gradient are obtained by solving a 2×2 linear system using the two known components computed using Eqs. (13) and (14).

A similar procedure has been used for updating the blanked cells. This is necessary because, as long as the shock keeps moving, those cells that have been blanked during the current time step may be reinserted into the computational mesh at a later time when they are not any longer crossed by the shock mesh.

Upon completion of step 7, the dependent variables of all cells of the background mesh and of all shock points of the shock mesh have been updated to time $t+\Delta t$, the computational mesh can be discarded, and the algorithm is restarted from step 1.

IV. Numerical Results

In the following sections we will analyze and discuss the results obtained for three inviscid simulations of high-speed flows using both the S-C and SESF approaches. All numerical simulations have been run using the in-house S-C code described in Sec. IV, which has also been used in the smooth regions of the flowfield in the SESF simulations. The comparison of the results allows to highlight the differences between these two shock-modeling options by studying solution quality and order-of-convergence properties.

A. Circular Cylinder

The first test case consists in the supersonic flow ($M_\infty=4$) past a circular cylinder. Despite its simplicity, it represents an important benchmark for the proposed technique, because the presence of a nonuniform subsonic region around the stagnation point and the transition to supersonic flow across the sonic line may be particularly challenging for the extrapolation procedures employed in the SESF algorithm. This test case has been simulated using three nested structured grids obtained by recursively coarsening the finest one to evaluate the qualitative and quantitative convergence properties of both the S-C and SESF techniques. The advantages of the SESF technique are clearly highlighted by Fig. 6, which shows the density flowfield computed using both the S-C and SESF approaches on all grid levels: the fitted bow shock is shown using a pink solid line in all three SESF solutions. By looking at the computed flowfields, it is clear that the S-C solution on the coarsest grid is plagued by the carbuncle phenomenon [5], a numerical instability that affects captured shock waves and is probably due to the unphysical states that arise inside the captured shock [3,4]. By modeling the bow shock as a true discontinuity, the SESF technique is not affected by this kind of numerical anomaly and provides high-quality solutions even on coarse grids. Moreover, it can be seen that on all grid levels SESF provides a much cleaner density distribution within the shock layer, compared to S-C.

A grid convergence analysis of the two shock-modeling techniques is also presented by analyzing the global total temperature error. Indeed, because the flow is steady and the freestream flow uniform, total temperature T_0 should be preserved throughout the whole computational domain. Table 1 presents the L_1 norm of the global error computed on all pairs of grid levels and the corresponding order of convergence, calculated according to [48]:

$$\tilde{n}_{i,i+1}=\frac{\log(\epsilon_i/\epsilon_{i+1})}{\log(r)} \tag{15}$$

In Eq. (15) the subscript $i=0,1,2$ refers to the coarse, intermediate, and fine grids, respectively, and the grid refinement ratio is constant and equal to $r=2$. The computation of the global discretization error involves a summation over all cells of the computational mesh, i.e., excluding the blanked cells. This has been done not only for the SESF solution, but also for the S-C one because the hypotheses leading from a truncated Taylor series expansion to Eq. (15) do not hold close to a discontinuity. By looking at the data collected in Table 1 and shown in Fig. 7, we see that the numerical error of the S-C calculation is larger than that obtained with SESF on the same grid level and, maybe more importantly, the SESF results are characterized by a second-order convergence trend, whereas S-C drops slightly below first order.

A local grid convergence analysis has also been performed at stagnation point, where pressure and density, besides total temperature, can be analytically computed. This is because the bow shock is a normal shock at the point where it is crossed by the stagnation streamline. Using the results provided by both the SESF technique and the S-C approach on all three grid levels, and listed in Table 2, it is possible to estimate the observed order of accuracy [48–50] $\tilde{n}$, which is computed as follows:

$$\tilde{n}=\frac{\log(\phi_0-\phi_1/\phi_1-\phi_2)}{\log r} \tag{16}$$

where ϕ_i denotes the variable of interest computed on the ith grid level. The analysis can be further detailed by relying on the Richardson extrapolated (RE) value [48,49]:

$$\tilde{\phi}_{\mathrm{RE}}=\phi_2-\frac{\phi_1-\phi_2}{r^{\tilde{n}}-1} \tag{17}$$

and the Grid Convergence Index (GCI). For the finest grid level, GCI is computed as follows [48]:

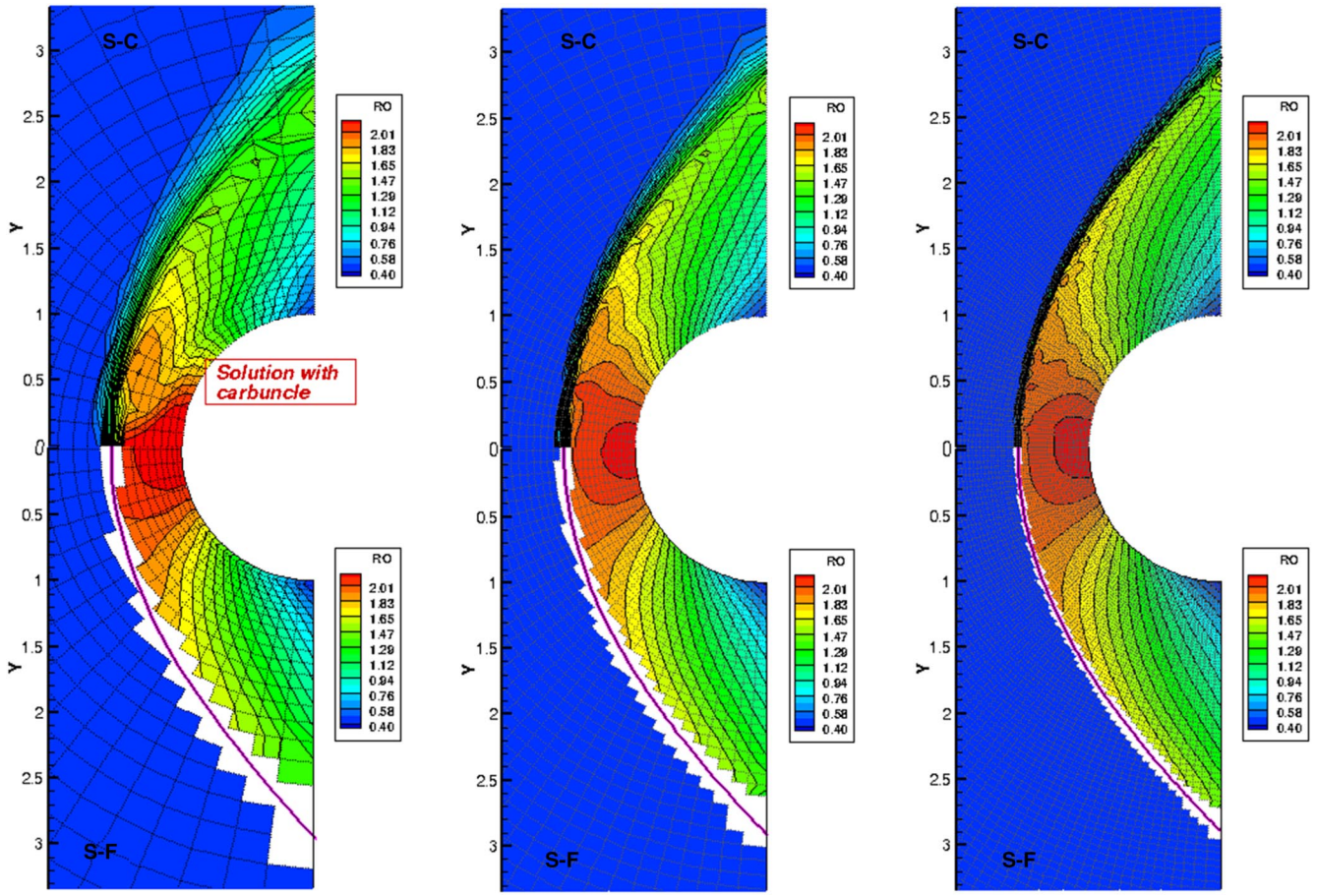


Fig. 6 Supersonic flow past a circular cylinder ($M = 4$): comparison between the density flowfield computed on all the grid levels using the two different shock-modeling approaches.

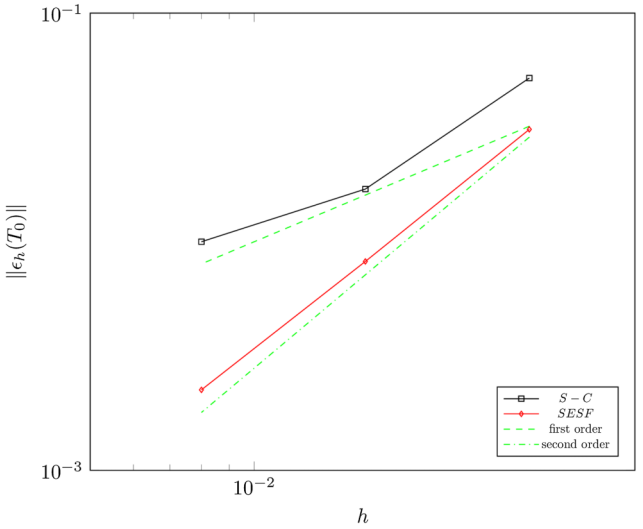


Fig. 7 Convergence trend of L_1 norm.

Table 1 Supersonic flow past a circular cylinder: convergence analysis

Mesh		SESF		S-C	
Grid level	h	L_1	$\tilde{n}$	L_1	$\tilde{n}$
0	0.032	$3.10 \cdot 10^{-2}$	— —	$8.61 \cdot 10^{-2}$	— —
1	0.016	$8.2 \cdot 10^{-3}$	1.91	$1.7 \cdot 10^{-2}$	— —
2	0.008	$2.25 \cdot 10^{-3}$	1.86	$1.0 \cdot 10^{-2}$	0.96

$$\mathrm{GCI}_2 = 3\frac{|\phi_2 - \phi_1|}{1 - r^p} \tag{18}$$

where p is the formal order of accuracy equal to 2. Specifically, GCI values referred to the coarser grid levels can be recursively computed from the GCI on the finest level according to Eq. (19), as defined in Ref. [48]:

$$\mathrm{GCI}_i = r^p \mathrm{GCI}_{i+1} \quad \text{for } i = 0, 1 \tag{19}$$

In particular, GCI values, which play the role of a numerical uncertainty, are reported in Table 2 with respect to each simulated value (simulated value $\pm$GCI). Furthermore, this table collects also the measured order of accuracy, the RE solution and the analytical values referred to the normalized pressure, density and total temperature. It is important to note that the calculation of both the observed order of accuracy and the RE values cannot be carried out for the normalized pressure and density computed by the S-C method, because of their non-monotonic convergence. This trend is due to the carbuncle that plagues the coarsest grid solution, especially regarding the pressure and density flowfield.

By comparing the data in Table 2, it is evident that the agreement between the SESF results on the finest grid and the analytical values is excellent and, for all three quantities, the exact solution falls within the uncertainty band defined by the GCI. As can be seen, the GCI data in Table 2 show that all GCI values are significantly smaller for the SESF solutions than the corresponding ones for S-C. This feature is particularly relevant when considering the solutions on the finest grid level, where, for instance, the ratio between the S-F and S-C GCI values regarding the normalized total temperature is one-seventh. Table 2 also shows that the measured order of the SESF technique is close to design (second) order, which allows to reliably use the RE value as a close approximation of the corresponding analytical

Table 2 Supersonic flow past a circular cylinder: local grid convergence study

	SESF			S-C		
Grid	P/P_∞	ρ/ρ_∞	T_0/T	P/P_∞	ρ/ρ_∞	T_0/T
Coarse	20.669 ± (0.301)	4.971 ± (0.088)	4.156 ± (0.013)	22.719 ± (1.25)	5.451 ± (0.419)	4.168 ± (0.024)
Intermediate	20.74 ± (0.075)	4.992 ± (0.022)	4.153 ± (0.003)	20.662 ± (0.313)	4.961 ± (0.105)	4.159 ± (0.013)
Fine	20.759 ± (0.019)	4.997 ± (0.005)	4.152 ± (0.001)	20.741 ± (0.078)	4.987 ± (0.026)	4.155 ± (0.007)
$\tilde{n}$	1.93	1.91	1.86	-	-	0.94
$\tilde{\phi}_{RE}$	20.767	5.002	4.151	-	-	4.149
Analytical values	20.765	5.000	4.152	20.765	5.000	4.152

quantity, which is indeed confirmed when comparing values in the two bottom rows of Table 2. On the contrary, S-C observed order of accuracy computed with respect to the total temperature values is close to first order, as highlighted also by the global convergence analysis previously discussed.

B. Regular Shock Reflection

The proposed SESF technique has also been used to compute flows featuring shock–wall interactions, which is the case, for example, of the regular reflection of an oblique shock that impinges on a flat plate. In the chosen flow configuration the Mach number ahead of the incident shock is $M_\infty = 2.2$ and the flow undergoes a deflection $\alpha = 3.75°$ through the incident shock. Despite the simplicity of this test case, which is characterized by three regions of uniform flow properties bounded by the incident and reflected shocks, it is impossible to recover the exact solution by modeling the shock waves using S-C.

The computational grid is an H-grid made of 200 × 80 cells evenly spaced along each coordinate line in both directions, as shown in Fig. 8.

As already shown in previous publications [20,24,29,41], S-F simulations can be performed in two different ways: fully fitted or hybrid. In *fully fitted* mode, all discontinuities are fitted and the point(s) where different discontinuities mutually interact or interact with a solid boundary must also be modeled. In *hybrid* mode, only some of the discontinuities are fitted and it is left to the S-C solver to capture all the other discontinuities as well as the interaction between fitted and captured discontinuities. Figures 9–11 allow to compare the differences among S-C, hybrid, and fully fitted SESF. First of all, Fig. 9b clearly reveals that the corner of the upper wall, where the incident shock originates, is a large source of error in the S-C calculation. This anomaly is absent in the SESF simulations. Not surprisingly, solution quality improves as the number of fitted shocks increases. In the hybrid SESF simulation shown in Fig. 10 the incident shock has been fitted and the reflected one captured, which is sufficient to significantly reduce the (unphysical) width of the captured reflected shock with respect to the S-C calculation: compare Fig. 10a with Fig. 9a. As shown in Fig. 11a, solution quality further improves when both shocks are fitted and their interaction point at the wall is modeled as described in [20]. In particular, the fully fitted solution of Fig. 11a allows to recover the uniform, analytical solution, within the two regions downstream of the incident and reflected shocks. It may be argued that this is of little practical use given the simplicity of the

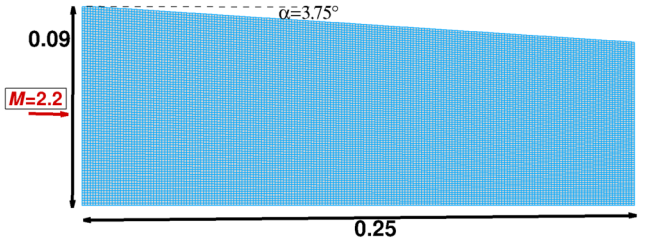
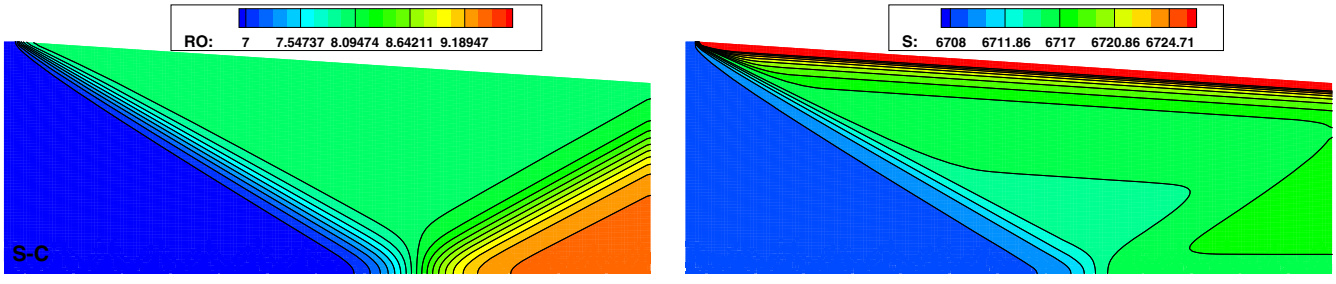
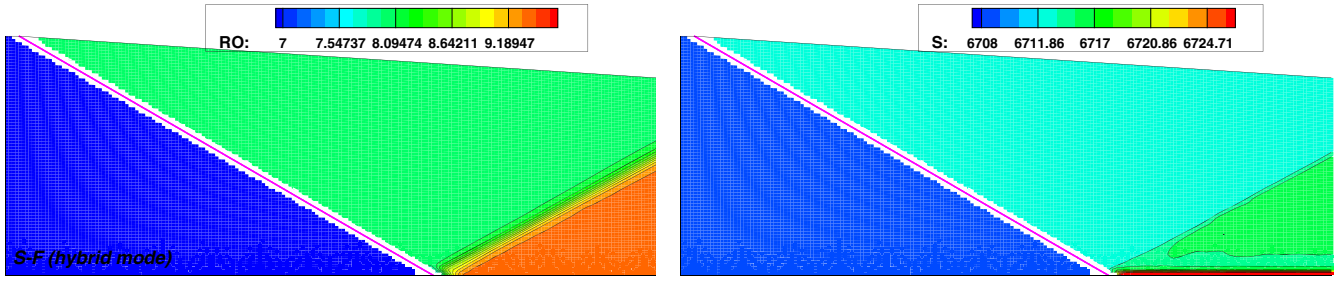


Fig. 8 Regular reflection: freestream conditions, domain, and background grid (200 × 80 cells).



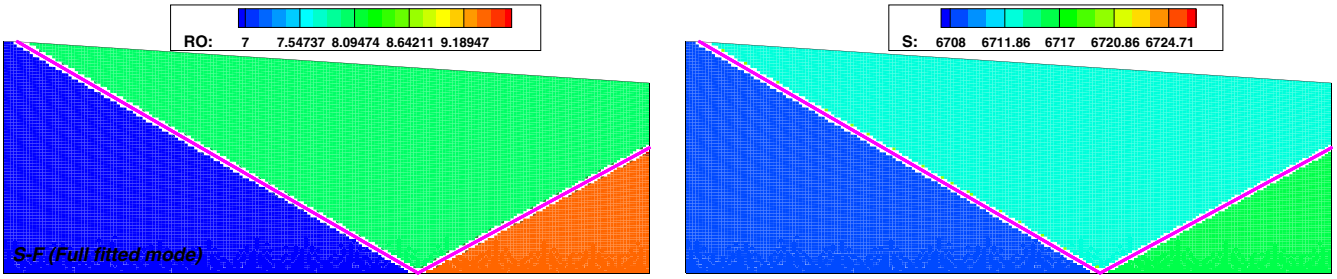
a) Density flow-field **b) Entropy flow-field**

Fig. 9 Regular shock reflection: S-C computation.



a) Density flow-field **b) Entropy flow-field**

Fig. 10 Regular shock reflection: hybrid S-F computations (fitted shock is depicted using pink solid line).



a) Density flow-field b) Entropy flow-field

Fig. 11 Regular shock reflection: full S-F computation (fitted shocks are depicted using pink solid lines).

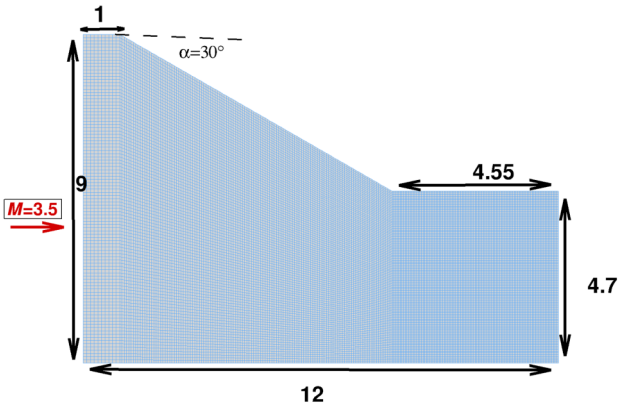


Fig. 12 Mach reflection: freestream conditions, domain, and background grid (232 × 102 cells).

shock-interaction pattern that is being analyzed. The test case to be presented in Sec. IV.C demonstrates that SESF outperforms S-C also when dealing with more complex shock-interaction topologies.

C. Mach Reflection

This last test case has been included in order to show that SESF is capable of dealing with complex shock patterns, such as the triple point that arises in a Mach reflection. The freestream conditions and the computational domain are reported in Fig. 12: the computational mesh is an H-grid made of 232 × 102 cells evenly spaced, along each coordinate line, in both directions.

The SESF simulation has been run in fully fitted mode: all four discontinuities that meet at the triple point, i.e., three shocks (the incident and reflected shocks and the Mach stem) and the contact discontinuity, have been fitted and the triple point has been modeled as described in [51].

The Mach number flowfield computed with both S-C and SESF is displayed in Figs. 13a and 13b, respectively. It is clearly shown that the SESF solution exhibits a better quality with respect to the S-C simulation: the Mach number distribution is smoother, especially downstream of the Mach stem and the reflected shock. This is evident when looking at the Mach distribution along the $x = 6.85$ line: Fig. 13c shows that the captured Mach stem gives rise to unphysical oscillations in the entire region downstream of the Mach stem, which are absent in the SESF calculation.

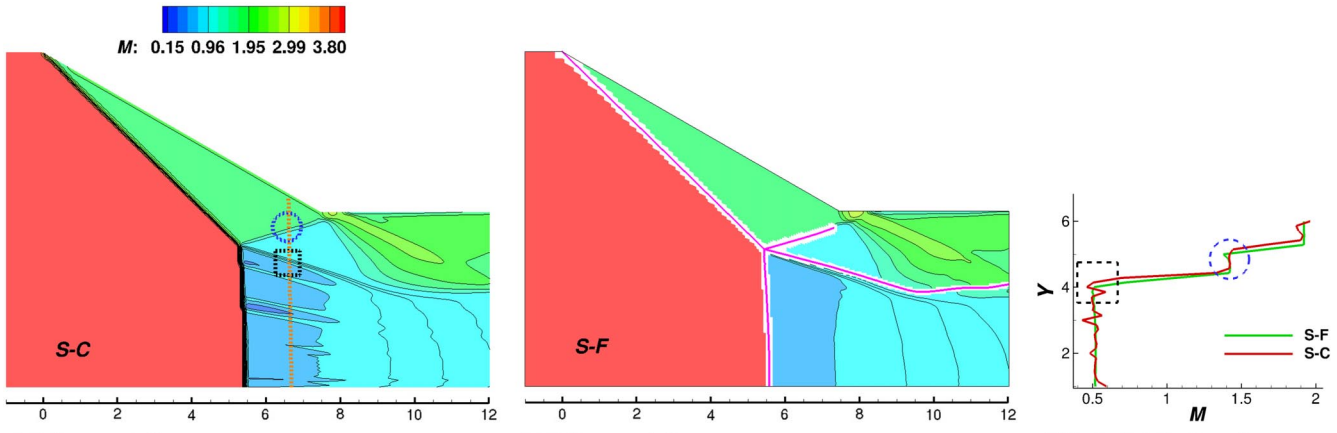
V. Conclusions

We have described a new computational technique for simulating high-speed compressible flows featuring discontinuous solutions (both shock waves and contact discontinuities) using structured-grid solvers. The technique, which we call structured extrapolated shock-fitting (SESF), has been described in detail, highlighting its algorithmic features, novel features, and its key advantages over the two S-F approaches, namely, boundary and floating S-F, which had been developed for structured-grid solvers starting in the 1960s.

SESF has been used to simulate three different inviscid 2D flows: the supersonic flow past a circular cylinder and both a regular and Mach reflection. The blunt-body flow has also been used to evaluate the local and global grid convergence properties of the new technique, which shows an observed order of convergence very close to the theoretical one, both globally and at stagnation point.

Moreover, the SESF solution is immune to the carbuncle phenomenon, which instead plagued the S-C solution on the coarsest mesh.

Last but not least, the SESF algorithm has been designed with the aim of developing an S-F technique that [52] "could be used as a black box in a variety of complicated problems." This is accomplished by treating as a black-box the structured-grid CFD solver that is used to discretize the governing PDEs in smooth regions of the flowfield, the only requirement being that the CFD code is capable of dealing with blanked cells.



a) S-C computation b) S-F computation c) Mach distribution along $x = 6.85$

Fig. 13 Mach reflection: Mach flowfield (fitted shocks are depicted using pink bold lines) and Mach distribution along the line $x = 6.85$ (orange dashes).

To summarize, the technique has been shown to be able to overcome typical problems encountered by S-C solvers (order-of-accuracy degradation and anomalous solutions) and to compute flows with shock–wall and shock–shock interaction, providing high-quality solutions also on coarse meshes.

Appendix: Order of Accuracy Study of the Extrapolation

To analyze the order of accuracy of the truncated Taylor series expansion (7) described in step 4 of Sec. III, a simple test case has been set up. It consists in extrapolating the following analytical function:

$$u_0(x, y) = -\cos\left(4\pi(\sqrt{3}x - y)\right)e^{0.7e^{(-x^2-4y^2)}}\cos(7\pi xy) \qquad \text{(A1)}$$

from the surrogate boundaries to three different curves, which play the role of the shock fronts in the SESF simulations. To perform a grid convergence analysis, three nested grid levels have been used; the coarsest one and the three curves, marked using solid lines, are shown in Fig. A1a, whereas the carpet plot of the analytical function is shown in Fig. A1b. We studied the grid convergence trend of the discretization error ϵ_h, which is the difference between the extrapolated solution u_h and the exact solution (A1):

$$\epsilon_h = u_h - u_0 \qquad \text{(A2)}$$

The grid convergence properties of Eq. (7) are shown in Table A1, which collects the L_1-norm (computed at all points along each curve) of the discretization error on all grid levels and the measured order of convergence $\tilde{n}_{i,i+1}$ for each pair of grid levels, computed using Eq. (15). All three tests confirm that the discretization error decays at design (second) order, as also clearly highlighted by the plot in Fig. A2.

Table A1 Convergence analysis for each curve in Fig. 14a

		Curve 1		Curve 2		Curve 3	
Grid level	H	L_1	$\tilde{n}$	L_1	$\tilde{n}$	L_1	$\tilde{n}$
0	0.032	1.54	— —	3.56	— —	4.92	— —
1	0.016	$3.92 \cdot 10^{-1}$	1.98	0.81	2.1	1.18	2.05
2	0.008	$1.08 \cdot 10^{-1}$	1.95	0.21	1.95	0.30	1.94

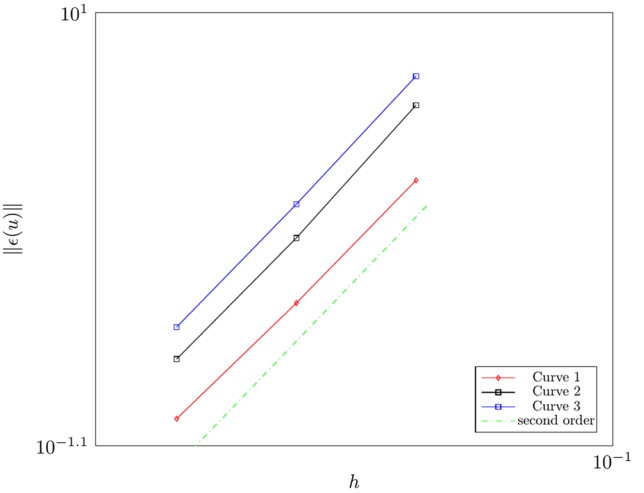


Fig. A2 Convergence behavior of the L_1-norm for each of the three curves shown in Fig. A1a.

Acknowledgments

This study was supported in part by a Starting Research Grant (named "Avvio alla Ricerca," ID: AR12117A628AD6BB) awarded by La Sapienza University of Rome, which is gratefully acknowledged.

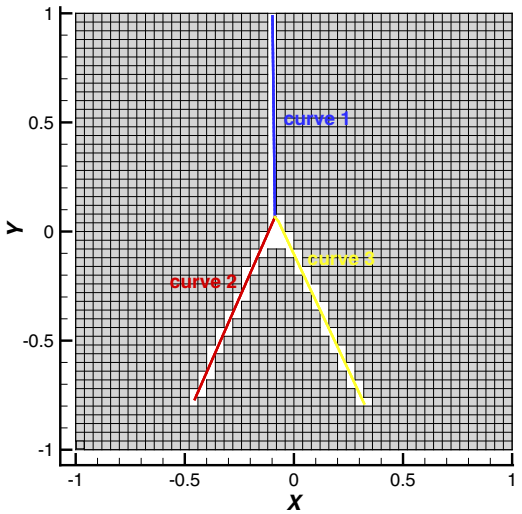
References

[1] Abbett, M., and Moretti, G., "A Time-Dependent Computational Method for Blunt Body Flows," *AIAA Journal*, Vol. 4, No. 12, 1966, pp. 2136–2141. https://doi.org/10.2514/3.3867

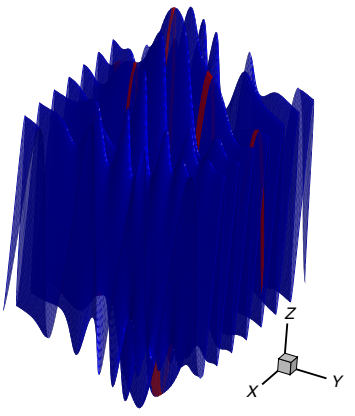
[2] Chern, I.-L., Glimm, J., Mcbryan, O., Plohr, B., and Yaniv, S., "Front Tracking for Gas Dynamics," *Journal of Computational Physics*, Vol. 62, No. 1, 1986, pp. 83–110. https://doi.org/10.1016/0021-9991(86)90101-4

[3] Zaide, D., and Roe, P., "Shock Capturing Anomalies and the Jump Conditions in One Dimension," *Fluid Dynamics and Co-Located Conferences*, AIAA Paper 2011-3686, AIAA, Reston, VA, 2011. https://doi.org/10.2514/6.2011-3686

[4] Zaide, D. W., and Roe, P. L., "A Second-Order Finite Volume Method That Reduces Numerical Shockwave Anomalies in One Dimension," *Fluid Dynamics and Co-Located Conferences*, AIAA Paper 2013-2699, AIAA, Reston, VA, 2013. https://doi.org/10.2514/6.2013-2699



a) Computational domain and curves definition



b) Carpet plot of Eq. (20) computed on the finest grid: red strips refer to the blanked regions in Fig. 14a

Fig. A1 Convergence test: computational domain and exact solution.

[5] Pandolfi, M., and D'Ambrosio, D., "Numerical Instabilities in Upwind Methods: Analysis and Cures for the 'Carbuncle' Phenomenon," *Journal of Computational Physics*, Vol. 166, No. 2, 2001, pp. 271–301. https://doi.org/10.1006/jcph.2000.6652

[6] Carpenter, M. H., and Casper, J. H., "Accuracy of Shock Capturing in Two Spatial Dimensions," *AIAA Journal*, Vol. 37, No. 9, 1999, pp. 1072–1079. https://doi.org/10.2514/2.835

[7] Yamamoto, Y., and Karashima, K., "Floating Shock Fitting for Three-Dimensional Inviscid Supersonic Flows," *AIAA Journal*, Vol. 20, No. 1, 1982, pp. 9–17. https://doi.org/10.2514/3.7897

[8] Marsilio, R., and Moretti, G., "Shock-Fitting Method for Two-Dimensional Inviscid, Steady Supersonic Flows in Ducts," *Meccanica*, Vol. 24, No. 4, 1989, pp. 216–222. https://doi.org/10.1007/BF01556453

[9] Marconi, F., and Salas, M., "Computation of Three Dimentional Flows About Aircraft Configurations," *Computers & Fluids*, Vol. 1, No. 2, 1973, pp. 185–195. https://doi.org/10.1016/0045-7930(73)90017-0

[10] Rawat, P. S., and Zhong, X., "On High-Order Shock-Fitting and Front-Tracking Schemes for Numerical Simulation of Shock-Disturbance Interactions," *Journal of Computational Physics*, Vol. 229, No. 19, 2010, pp. 6744–6780. https://doi.org/10.1016/j.jcp.2010.05.021

[11] Glimm, J., Grove, J. W., Li, X. L., Shyue, K.-M., Zeng, Y., and Zhang, Q., "Three-Dimensional Front Tracking," *SIAM Journal on Scientific Computing*, Vol. 19, No. 3, 1998, pp. 703–727. https://doi.org/10.1137/S1064827595293600

[12] Glimm, J., Isaacson, E., Marchesin, D., and McBryan, O., "Front Tracking for Hyperbolic Systems," *Advances in Applied Mathematics*, Vol. 2, No. 1, 1981, pp. 91–119. https://doi.org/10.1016/0196-8858(81)90040-3

[13] Du, J., Fix, B., Glimm, J., Jia, X., Li, X., Li, Y., and Wu, L., "A Simple Package for Front Tracking," *Journal of Computational Physics*, Vol. 213, No. 2, 2006, pp. 613–628. https://doi.org/10.1016/j.jcp.2005.08.034

[14] Klingenberg, C., and Plohr, B., "An Introduction to Front Tracking," *Multidimensional Hyperbolic Problems and Computations*, Springer, Berlin, 1991, pp. 203–216. https://doi.org/10.1137/S106482750343028X

[15] Helzel, C., Berger, M. J., and LeVeque, R. J., "A High-Resolution Rotated Grid Method for Conservation Laws with Embedded Geometries," *SIAM Journal on Scientific Computing*, Vol. 26, No. 3, 2005, pp. 785–809. https://doi.org/10.1137/S106482750343028X

[16] Chern, I.-L., and Colella, P., "*A Conservative Front Tracking Method for Hyperbolic Conservation Laws*," Lawrence Livermore National Lab. Rept. No. UCRL-97200, Vol. 2, 1987, pp. 83–110.

[17] Harten, A., and Hyman, J. M., "Self Adjusting Grid Methods for One-Dimensional Hyperbolic Conservation Laws," *Journal of Computational Physics*, Vol. 50, No. 2, 1983, pp. 235–269. https://doi.org/10.1016/0021-9991(83)90066-9

[18] Mao, D. K., "A Treatment of Discontinuities for Finite Difference Methods in the Two-Dimensional Case," *Journal of Computational Physics*, Vol. 104, No. 2, 1993, pp. 377–397. https://doi.org/10.1006/jcph.1993.1038

[19] Paciorri, R., and Bonfiglioli, A., "A Shock-Fitting Technique for 2D Unstructured Grids," *Computers & Fluids*, Vol. 38, No. 3, 2009, pp. 715–726. https://doi.org/10.1016/j.compfluid.2008.07.007

[20] Paciorri, R., and Bonfiglioli, A., "Shock Interaction Computations on Unstructured, Two-Dimensional Grids Using a Shock-Fitting Technique," *Journal of Computational Physics*, Vol. 230, No. 8, 2011, pp. 3155–3177. https://doi.org/10.1016/j.jcp.2011.01.018

[21] Assonitis, A., Paciorri, R., and Bonfiglioli, A., "Numerical Simulation of Shock/Boundary-Layer Interaction Using an Unstructured Shock-Fitting Technique," *Computers & Fluids*, Vol. 228, Oct. 2021, Paper 105058. https://doi.org/10.1016/j.compfluid.2021.105058

[22] Bonfiglioli, A., Paciorri, R., and Campoli, L., "Unsteady Shock-Fitting for Unstructured Grids," *International Journal for Numerical Methods in Fluids*, Vol. 81, No. 4, 2016, pp. 245–261. https://doi.org/10.1002/fld.4183

[23] Campoli, L., Quemar, P., Bonfiglioli, A., and Ricchiuto, M., "Shock-Fitting and Predictor-Corrector Explicit ALE Residual Distribution," *Shock Fitting: Classical Techniques, Recent Developments, and Memoirs of Gino Moretti*, edited by M. Onofri, and R. Paciorri, Springer International Publishing, Cham, Switzerland, 2017, pp. 113–129. https://doi.org/10.1007/978-3-319-68427-7_5

[24] Campoli, L., Assonitis, A., Ciallella, M., Paciorri, R., Bonfiglioli, A., and Ricchiuto, M., "UnDiFi-2D: An Unstructured Discontinuity Fitting Code for 2D grids," *Computer Physics Communications*, Vol. 271, Feb. 2022, Paper 108202. https://doi.org/10.1016/j.cpc.2021.108202

[25] Bonfiglioli, A., Grottadaurea, M., Paciorri, R., and Sabetta, F., "An Unstructured, Three-Dimensional, Shock-Fitting Solver for Hypersonic Flows," *Computers & Fluids*, Vol. 73, 2013, pp. 162–174. https://doi.org/10.1016/j.compfluid.2012.12.022

[26] Ollivier-Gooch, C., Paciorri, R., Assonitis, A., and Bonfiglioli, A., "An Unstructured Shock-Fitting Technique for Three-Dimensional Flows with Shock Interactions," *WCCM-ECCOMAS Congress*, Vol. 200, ESP, 2021, pp. 1–12. https://doi.org/10.23967/wccm-eccomas.2020.065

[27] Liu, J., and Zou, D., *A Shock-Fitting Technique for ALE Finite Volume Methods on Unstructured Dynamic Meshes*, Springer International Publishing, Cham, Switzerland, 2017, pp. 131–149. https://doi.org/10.1007/978-3-319-68427-7_6

[28] Zou, D., Xu, C., Dong, H., and Liu, J., "A Shock-Fitting Technique for Cell-Centered Finite Volume Methods on Unstructured Dynamic Meshes," *Journal of Computational Physics*, Vol. 345, Sept. 2017, pp. 866–882. https://doi.org/10.1016/j.jcp.2017.05.047

[29] Chang, S., Bai, X., Zou, D., Chen, Z., and Liu, J., "An Adaptive Discontinuity Fitting Technique on Unstructured Dynamic Grids," *Shock Waves*, Vol. 29, No. 8, 2019, pp. 1103–1115. https://doi.org/10.1007/s00193-019-00913-3

[30] Zou, D., Bonfiglioli, A., Paciorri, R., and Liu, J., "An Embedded Shock-Fitting Technique on Unstructured Dynamic Grids," *Computers & Fluids*, Vol. 218, 2021, Paper 104847. https://doi.org/10.1016/j.compfluid.2021.104847

[31] D'Aquila, L. M., Helenbrook, B. T., and Mazaheri, A., "A Novel Stabilization Method for High-Order Shock Fitting with Finite Element Methods," *Journal of Computational Physics*, Vol. 430, 2021 Apr., Paper 110096. https://doi.org/10.1016/j.jcp.2020.110096

[32] Zahr, M., and Persson, P.-O., "An Optimization-Based Approach for High-Order Accurate Discretization of Conservation Laws with Discontinuous Solutions," *Journal of Computational Physics*, Vol. 365, July 2018, pp. 105–134. https://doi.org/10.1016/j.jcp.2018.03.029

[33] Zahr, M. J., Shi, A., and Persson, P.-O., "Implicit Shock Tracking Using an Optimization-Based High-Order Discontinuous Galerkin Method," *Journal of Computational Physics*, Vol. 410, June 2020, Paper 109385. https://doi.org/10.1016/j.jcp.2020.109385

[34] Zahr, M. J., and Powers, J. M., "High-Order Resolution of Multidimensional Compressible Reactive Flow Using Implicit Shock Tracking," *AIAA Journal*, Vol. 59, No. 1, 2021, pp. 150–164. https://doi.org/10.2514/1.J059655

[35] Huang, T., and Zahr, M. J., "A Robust, High-Order Implicit Shock Tracking Method for Simulation of Complex, High-Speed Flows," *Journal of Computational Physics*, Vol. 454, 2022 Apr, Paper 110981. https://doi.org/10.1016/j.jcp.2022.110981

[36] Corrigan, A. T., Kercher, A. D., Kessler, D. A., and Wood-Thomas, D., "Application of the Moving Discontinuous Galerkin Method with Interface Condition Enforcement to Shocked Compressible Flows," *AIAA Aviation Forum*, AIAA Paper 2018-4272, AIAA, Reston, VA, 2018. https://doi.org/10.2514/6.2018-4272

[37] Corrigan, A. T., Kercher, A., and Kessler, D. A., "The Moving Discontinuous Galerkin Method with Interface Condition Enforcement for Unsteady Three-Dimensional Flows," *AIAA SciTech Forum*, AIAA Paper 2019-0642, AIAA, Reston, VA, 2019. https://doi.org/10.2514/6.2019-0642

[38] Corrigan, A., Kercher, A. D., and Kessler, D. A., "A Moving Discontinuous Galerkin Finite Element Method for Flows with Interfaces," *International Journal for Numerical Methods in Fluids*, Vol. 89, No. 9, 2019, pp. 362–406. https://doi.org/10.1002/fld.4697

[39] Song, T., Main, A., Scovazzi, G., and Ricchiuto, M., "The Shifted Boundary Method for Hyperbolic Systems: Embedded Domain Computations of Linear Waves and Shallow Water Flows," *Journal of Computational Physics*, Vol. 369, Sept. 2018, pp. 45–79. https://doi.org/10.1016/j.jcp.2018.04.052

[40] Ciallella, M., Ricchiuto, M., Paciorri, R., and Bonfiglioli, A., "Extrapolated Shock Tracking: Bridging shock-Fitting and Embedded Boundary Methods," *Journal of Computational Physics*, Vol. 412, July 2020,

Paper 109440.
https://doi.org/10.1016/j.jcp.2020.109440

[41] Ciallella, M., Ricchiuto, M., Paciorri, R., and Bonfiglioli, A., "Extrapolated DIscontinuity Tracking for Complex 2D Shock Interactions," *Computer Methods in Applied Mechanics and Engineering*, Vol. 391, March 2022, Paper 114543.
https://doi.org/10.1016/j.cma.2021.114543

[42] Assonitis, A., Ciallella, M., Paciorri, R., Ricchiuto, M., and Bonfiglioli, A., "A Shock-Fitting Technique for 2D/3D Flows with Interactions Using Structured Grids," *AIAA Aviation Forum*, AIAA Paper 2022-4123, 2022.
https://doi.org/10.2514/6.2022-4123

[43] Lani, A., and De Amicis, V., "SF: An Open Source Object-Oriented Platform for Unstructured Shock-Fitting Methods," *Shock Fitting: Classical Techniques, Recent Developments, and Memoirs of Gino Moretti*, edited by M. Onofri, and R. Paciorri, Springer International Publishing, Cham, Switzerland, 2017, pp. 85–112.
https://doi.org/10.1007/978-3-319-68427-7_4

[44] Paciorri, R., and Bonfiglioli, A., "Accurate Detection of Shock Waves and Shock Interactions in Two-Dimensional Shock-Capturing Solutions," *Journal of Computational Physics*, Vol. 406, April 2020, Paper 109196.
https://doi.org/10.1016/j.jcp.2019.109196

[45] Geron, M., Paciorri, R., Nasuti, F., and Sabetta, F., "Flowfield Analysis of a Linear Clustered Plug Nozzle with Round-to-Square Modules," *Aerospace Science and Technology*, Vol. 11, Nos. 2–3, 2007, pp. 110–118.
https://doi.org/10.1016/j.ast.2006.08.004

[46] Pizzarelli, M., Nasuti, F., Paciorri, R., and Onofri, M., "Numerical Analysis of Three-Dimensional Flow of Supercritical Fluid in Cooling Channels," *AIAA Journal*, Vol. 47, No. 11, 2009, pp. 2534–2543.
https://doi.org/10.2514/1.38542

[47] Paciorri, R., Sabetta, F., and Bonfiglioli, A., "Turbulence Modeling of Base Drag on Launcher in Subsonic Flight," *Journal of Spacecraft and Rockets*, Vol. 51, No. 5, 2014, pp. 1673–1680.
https://doi.org/10.2514/1.A32757

[48] Roache, P. J., "Quantification of Uncertainty in Computational Fluid Dynamics," *Annual Review of Fluid Mechanics*, Vol. 29, No. 1, 1997, pp. 123–160.
https://doi.org/10.1146/annurev.fluid.29.1.123

[49] Di Mascio, A., Paciorri, R., and Favini, B., "Truncation Error Analysis in Turbulent Boundary Layers," *Journal of Fluid Engineering*, Vol. 124, No. 3, 2002, pp. 657–663.
https://doi.org/10.1115/1.1478564

[50] Bonfiglioli, A., and Paciorri, R., "Convergence Analysis of Shock-Capturing and Shock-Fitting Solutions on Unstructured Grids," *AIAA Journal*, Vol. 52, No. 7, 2014, pp. 1404–1416.
https://doi.org/10.2514/1.J052567

[51] Ivanov, M. S., Bonfiglioli, A., Paciorri, R., and Sabetta, F., "Computation of Weak Steady Shock Reflections by Means of an Unstructured Shock-Fitting Solver," *Shock Waves*, Vol. 20, No. 4, 2010, pp. 271–284.
https://doi.org/10.1007/s00193-010-0266-y

[52] Moretti, G., "Thirty-Six Years of Shock Fitting," *Computers & Fluids*, Vol. 31, Nos. 4–7, 2002, pp. 719–723.
https://doi.org/10.1016/S0045-7930(01)00072-X

J. Larsson
Associate Editor