



# An unstructured, three-dimensional, shock-fitting solver for hypersonic flows

A. Bonfiglioli<sup>a,\*</sup>, M. Grottadaurea<sup>b</sup>, R. Paciorri<sup>b</sup>, F. Sabetta<sup>b</sup>

<sup>a</sup> Università degli Studi della Basilicata, Scuola di Ingegneria (SI), V.le dell'Ateneo Lucano 10, 85100 Potenza, Italy

<sup>b</sup> Università degli Studi di Roma La Sapienza, Dip.to di Ingegneria Meccanica e Aerospaziale (DIMA), Via Eudossiana 18, 00184 Roma, Italy

## ARTICLE INFO

### Article history:

Received 27 September 2011

Received in revised form 18 October 2012

Accepted 30 December 2012

Available online 11 January 2013

### Keywords:

Shock-capturing

Shock-fitting

Unstructured grids

Hypersonic flows

## ABSTRACT

A novel unstructured shock-fitting algorithm for three-dimensional flows is proposed in this paper. This new technique is able to work coupled with any unstructured vertex centred solver. The fitted shock front is described using a triangulated surface that is allowed to move, while obeying to the Rankine–Hugoniot jump relations, throughout a background tetrahedral mesh which covers the entire computational domain. At each time step, a local, constrained Delaunay tetrahedralization is applied in the neighbourhood of the shock front to ensure that the triangles, which make up the shock surface, belong to the overall tetrahedral grid. The fitted shock front acts as an interior boundary for the unstructured shock-capturing solver, used to solve the discretised governing equations in the smooth regions of the flow-field. The present algorithm has been tested against high speed flows past three-dimensional bodies, providing high quality results even using coarse grain tetrahedralization.

© 2013 Elsevier Ltd. All rights reserved.

## 1. Introduction

The numerical simulation of hypersonic flows past blunt bodies by means of shock-capturing solvers is not exempt from critical aspects, sometimes leading to nasty surprises. Stagnation point anomalies [1–5], spurious numerical oscillations [6,7], the carbuncle phenomenon [8–10] and the reduction of the order of accuracy of the solution in the entire region downstream of a captured shock [11,6,12,13] are examples of such drawbacks.

There is no general consensus about the causes behind these undesirable side effects of current shock-capturing schemes.

The lack of a truly multidimensional shock-capturing scheme, capable of making the computed solution insensitive to the orientation of the control volume faces relative to the shock front, is often [1,3–5] thought to be the cause of the anomalous stagnation heating predictions exhibited by state-of-the-art shock-capturing solvers, particularly when used in association with unstructured tetrahedral meshes. It might also explain why shock-capturing solvers generate spurious waves along the entire extent of the shock front when the faces of the mesh are not locally aligned with it [7]. These artificial waves propagate downstream and pollute the solution in the entire shock-layer [6].

\* Corresponding author. Tel.: +39 0971205203; fax: +39 0971205215.

E-mail addresses: [aldo.bonfiglioli@unibas.it](mailto:aldo.bonfiglioli@unibas.it) (A. Bonfiglioli), [marco.grottadaurea@googlemail.com](mailto:marco.grottadaurea@googlemail.com) (M. Grottadaurea), [renato.paciorri@uniroma1.it](mailto:renato.paciorri@uniroma1.it) (R. Paciorri), [filippo.sabetta@uniroma1.it](mailto:filippo.sabetta@uniroma1.it) (F. Sabetta).

The development of truly multi-dimensional upwind schemes has therefore gathered considerable interest in past years [14,15,4,5] and has even led to the development of a novel class of discretization schemes: Residual distribution or fluctuation splitting schemes [16–18].

However, even the so-called multi-dimensional upwind schemes are not entirely exempt from the aforementioned difficulties; for example, they develop carbuncles [19,10], which certainly represent the most bizarre and “pervasive” [10] of all shock-capturing anomalies.

Finally, loss of accuracy behind a captured shock has little to do with multi-dimensionality, since it is also observed in one-dimensional unsteady flows [20,13] and quasi-one-dimensional steady flows [11]. It is more likely connected with the fact that a numerical shock is captured with at least one grid point falling in between the upstream and downstream states. Indeed, if a cell interface is made to coincide with the shock (which is possible, at least in one space dimension, using some sort of shock-fitting or front-tracking), the design order of the scheme is recovered downstream of the discontinuity [20,13].

If one gives up the hope of finding a perfectly “mesh-independent” scheme, one can try to improve the solution quality by adapting the faces of the control volumes to the shock front. With structured grids, this is possible and produces good quality results as long as the flow is characterised by the presence of a unique strong shock (bow shock), see e.g. [6]. However, the presence of more shocks and the interactions between the bow shock and other internal or external shocks severely challenge the capability

of a structured-grid, mesh-adaptation and/or refinement procedure to provide satisfactory results in these complex cases, due to the topological restrictions that are inherent with the use of structured grids.

The use of unstructured meshes completely removes all topological restrictions, but hypersonic flow solutions computed with unstructured shock-capturing solvers are generally characterised by a low quality, especially in the neighbourhood of the stagnation point and in particular when fully tetrahedral grids are used [2–5]. Indeed, as observed by Gnoffo [4], “tetrahedral cell topologies by their nature have at least a surface that is skewed to the captured shock”. This gives a plausible explanation of the remarkable production of spurious waves observed in unstructured shock-capturing solutions and of the reduced effectiveness of the mesh-adaptation and mesh-refinement procedures in the unstructured grid framework with respect to the structured one, at least in hypersonic applications. Based on these observations, several authors [21,22,2] advocate the use of structured or semistructured meshes for hypersonic flow calculations and advise against the use of fully tetrahedral meshes [2].

Following a completely different route, “the application of Shock-Fitting in the context of unstructured grids may offer improved solution quality overcoming the difficulties associated with capturing the strong bow shock”, as noted by Gnoffo [4]. Indeed, the shock-fitting technique consists in finding the precise shock location and computing the shock motion by means of the Rankine–Hugoniot equations, thus completely by-passing the shock-capturing numerical model. Shock-fitting was proposed by Emons [23] and further developed and made popular in the structured grid context by Moretti and co-workers [24,25]. Beside Moretti, other researchers applied this technique to two-dimensional and three-dimensional flows in the 1970s and 1980s (see, for instance, Refs. [26–28]) and in the early 1990s [29,30]. In those years, the shock-fitting technique, used in association with structured-grid solvers based on the quasi-linear form of the equations, allowed to accurately compute flows characterised by strong discontinuities using the modest computational resources available at that time. In the 1990s, however, the development of the modern shock-capturing schemes based on the integral conservation equations, along with the strong growth in computing resources, diminished the interest in shock-fitting schemes, which were considered algorithmically too complex and not general enough as compared to the shock-capturing schemes.

Since then, the development and the application of the shock-fitting technique has been pursued only by a limited number of research groups world-wide. These include: Onofri and Nasuti [31,32] who used floating shock-fitting to simulate unsteady, internal flows; Kopriva [33,34], Zhong [35,36] and Sesterhenn [37] who coupled boundary shock-fitting with either spectral or high order finite-difference discretization schemes to simulate either hypersonic blunt body flows or to perform DNS of compressible turbulence. All the aforementioned applications of the shock-fitting technique were pursued in the structured-grid discretization framework.

Very recently, Paciorri and Bonfiglioli [38] have proposed a shock-fitting technique for two-dimensional unstructured meshes, obtaining promising results in the computation of flows with strong shocks and complex shock-shock interactions [39,40]. This two-dimensional technique produces high quality results [38–40] and seems to solve many, if not most, of the problems that affect shock-capturing, unstructured CFD solvers when used to simulate flows characterised by strong shocks. Moreover, the geometrical versatility offered by unstructured meshes has allowed to relieve some of the algorithmic difficulties that had contributed to the gradual dismissal of the structured-grid, shock-fitting technique over the last decades. In particular, we have shown in Refs.

[39,40] that fitting the shocks in an unstructured framework allows to deal with shock interactions much more properly than it used to be done with structured grids [41,42,31].

It is true, however, that even the unstructured, shock-fitting technique we propose is still far from being a general-purpose tool. Presently, it relies on an a priori knowledge of the flow topology and an approximate location of the discontinuities. This latter limitation could be lifted by implementing shock detection mechanisms that have already been proposed in the literature (see Ref. [43] for a comprehensive review). Even so, the capability of automatically identifying changes in the flow topology, such as those occurring in un-steady shock interactions, is a quite formidable task from an algorithmic viewpoint. However, as observed by Moretti [44], this has more to do with topology than fluid mechanics and some expertise from other areas of computer science might help in making shock-fitting a more versatile technique.

Despite these caveats the generalisation of the proposed shock-fitting algorithm to compute three-dimensional flows appears to be even more interesting and promising. However, the extension to the three-dimensional space poses additional complications that are not present in the two-dimensional scenario.

The key thread of this paper consists in describing the early developments in extending the unstructured shock-fitting methodology to the three-dimensional space, highlighting the differences that exist with respect to its two dimensional counterpart. A preliminary version of the three-dimensional shock-fitting algorithm for unstructured meshes will be described in Section 2 and applied in Section 3 to the steady state computation of the high speed flow past a sphere, a blunt-nosed body and a geometrically complex three-dimensional re-entry capsule.

## 2. Shock fitting algorithm

In the shock-fitting algorithm the fitted shocks are treated as true surface discontinuities. The fitted shock front is modelled using a zero-thickness, double-sided, triangulated surface that is allowed to move, while obeying to the Rankine–Hugoniot jump relations, throughout a background tetrahedral mesh which covers the entire computational domain. At each time step, a local, constrained Delaunay tetrahedralization is applied in the neighbourhood of the shock front to ensure that the triangles, which make up the shock surface, also belong to the overall tetrahedral grid. The fitted shock front acts as an interior boundary for a shock-capturing solver, which is used to solve the governing equations within the computational domain. In the present implementation, the shock-capturing solver is EulFS, an in-house code based on the Fluctuation Splitting approach which implements both first- and second-order-accurate schemes. A detailed description of the EulFS code can be found in [45,46], whereas the grid-convergence properties of the available schemes have been shown in [47–49].

The shock-fitting algorithm to be described in the present section should be regarded as a preliminary version, since a number of algorithmic issues which shall be addressed are currently either being ignored or treated in a nongeneral way. In particular, we refer to the management of the Steiner points, which shall be discussed in Section 2.2.

As already mentioned, we restrict our discussion to the simulation of steady flows of inviscid and calorically perfect gases. Steady solutions are approached asymptotically following a transient and the present section describes in details all the steps that need to be taken in order to advance in time the solution over a single time step of size  $\Delta t$ .

Let us suppose that at a given time  $t$  the solution is known at all grid-points of a “background” tetrahedral mesh that covers the entire computational domain. The shock front is separately available

as a surface triangulation of a set of shock nodes, as shown in Fig. 1a, and the flow variables at each shock point are also known. Each grid-point of the background tetrahedralization is characterised by a single state, whereas each of the grid-points belonging to the shock surface is characterised by two states: one corresponding to the low pressure (upstream) side of the shock, the other to the high-pressure (downstream) side.

We shall describe the case of shock waves, but with a few easy modifications, see [40], the present algorithm can be adapted to deal with contact discontinuities as well. It is important to underline that no shock-detection mechanism is available in the current version of the algorithm, so that the fitted shocks must be present within the flow domain at the beginning of the simulation. Although the lack of a shock-detection procedure reduces the generality of the present shock-fitting technique, this limitation does not represent a particularly serious obstacle to the simulation of steady flows, because the approximate location of the shocks can be estimated using a preliminary shock-capturing calculation on a coarse mesh.

Although the present version of the algorithm allows to deal with more than a single fitted shock surface, fitted shocks are currently not allowed to interact with each other. However, since the present shock-fitting technique has been coupled with an unstructured shock-capturing solver, shock-shock interactions can still be dealt with either by fitting one of the shocks and capturing the remaining ones or by fitting all shocks and capturing their interaction. Examples of application of these features will be given in the present article.

### 2.1. Step 2.1: cell removal around the shock front

The first step consists in removing: (i) all tetrahedra of the background mesh that are crossed by the shock surface; (ii) the mesh points (phantom or ghost nodes) of the background mesh which are located too close to the shock front; and (iii) the tetrahedra of the background mesh that have at least one phantom node among their vertices. The ghost cells correspond to the tetrahedra that have been removed because of conditions (i) and (iii). In Fig. 1a, some of the ghost cells of the background mesh are indicated by dashed arrows, whereas bullet points identify the phantom nodes. The grid-points of the background tetrahedralization are identified as phantom nodes whenever the following inequality is met:

$$d/l \leq d_{ref}/L \quad (1)$$

In Eq. (1)  $d$  is the grid-point distance from the nearest shock surface,  $l$  is the mean length of the edges of the triangles that make up the

shock front and  $d_{ref}/L$  is a dimensionless reference distance that is pre-set by the user. Fig. 1b shows the background mesh after the removal of all the ghost cells and the phantom nodes.

### 2.2. Step 2.2: local re-meshing around the shock front

Following the cell removal step, the remaining part of the background mesh is characterised by a hole containing the shock surface, as shown in Fig. 1b. This hole is then re-meshed using a constrained Delaunay tetrahedralization (CDT), as shown in Fig. 1c: both the shock triangular faces and those belonging to the boundary of the hole are constrained to be part of the final tetrahedral grid. The local re-meshing is performed using the TetGen mesh generator [50]. It is important to underline that, whereas in a two-dimensional space a CDT can always be built, this may not be the case in three dimensions, since not every polyhedron has a constrained tetrahedralization [51]. Whenever this occurs, additional (so-called Steiner) points need to be introduced into the tetrahedral grid. It has been observed, at least for the test cases examined herein, that the number of Steiner points is extremely limited and that a careful choice of the value  $d_{ref}/L$  in Eq. (1) can prevent altogether the addition of Steiner points in the tetrahedral mesh. The parameter  $d_{ref}/L$  is indeed representative of the local regularity of the cell; therefore, its careful choice ensures a rather regular tetrahedral cell shape close to the shock-fitted surface, thus preventing Steiner points to be added during the simulation. For this reason, the preliminary version of the shock-fitting algorithm presented in this paper does not include any special treatment for the Steiner points. Nevertheless, a more general version of the algorithm will require an ad hoc treatment of these points which, in any case, does not seem to pose any particular problem from the algorithmic point of view.

Upon completion of step 2.2, the computational domain is discretised by a “modified” computational mesh, different from the “background” one, in which the shock points and the (triangular) shock faces are part of a CDT that covers the entire computational domain.

Note that the re-meshing needed at each time step is local, being limited to the immediate neighbourhood of the shock surface.

### 2.3. Step 2.3: computation of the tangent and normal unit vectors

The normal unit vectors ( $\mathbf{n}$ ) on the shock surface have to be calculated within each shock point since this is required to compute the jump relations, as described in Section 2.5. The computation

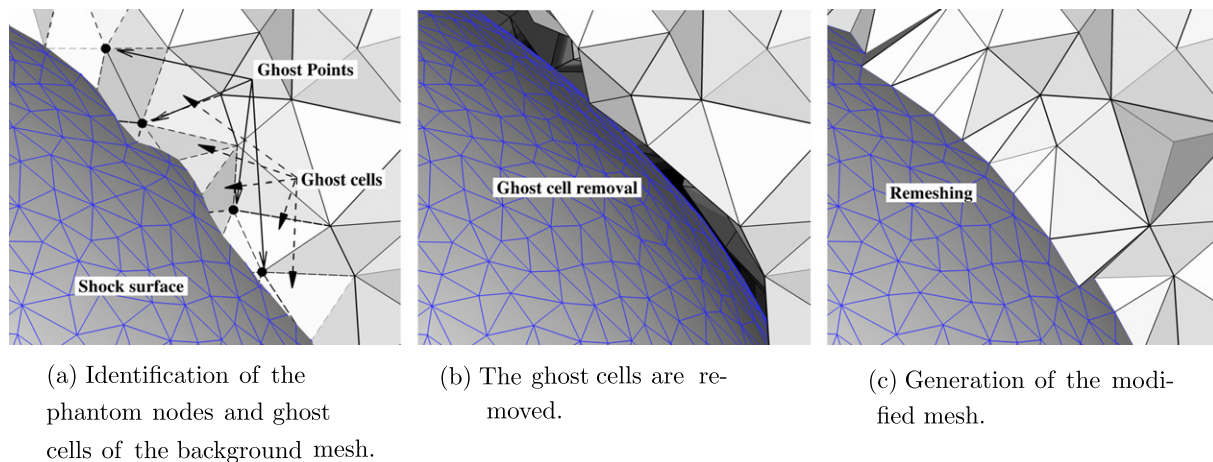


Fig. 1. Detail of the re-meshed zone.

of these vectors in a generic shock point can be carried out by averaging the face normals of all shock faces that share the given shock point. Similarly to the two-dimensional case [38], however, the averaging process has to take into account only the shock faces that belong to the range of influence of a shock point, see Ref. [52] for further details. By convention, the normal unit vector is oriented from the downstream region to the upstream one.

#### 2.4. Step 2.4: solution update on the shock nodes using the capturing code

The solution is updated at time level  $t + \Delta t$  using the unstructured, shock-capturing EulFS code. The gas-dynamic solver uses the modified mesh generated in step 2.2 as input and, therefore, the computational domain is cut into non-communicating parts by the shock surface. As shown in Fig. 2, this is achieved by replacing each shock point by two superimposed mesh points: one belonging to the downstream region and the other to the upstream region. The downstream and upstream states are correspondingly assigned to each pair of shock mesh nodes. A single time step calculation is performed by the unstructured solver which provides updated nodal values within all grid and shock points at time level  $t + \Delta t$ . The relative flow on the upstream (low pressure) side of the shock is supersonic; therefore, this side of the shock surface behaves like a supersonic outflow boundary so that no boundary conditions need to be specified and the upstream nodal values of the shock surface can be updated using information coming only from the upstream side of the computational domain. On the downstream (high pressure) side of the shock front, however, this is no longer true. Indeed, the downstream shock side behaves like a subsonic inflow boundary and a well posed computation requires four boundary conditions. There is one piece of information, however, which is convected towards the shock from the shock-downstream side of the domain. It is given by the following signal (or Riemann variable):

$$R_d^{t+\Delta t} = \tilde{a}_d^{t+\Delta t} + \frac{\gamma - 1}{2} \tilde{u}_d^{t+\Delta t} \cdot \mathbf{n} \quad (2)$$

which is associated with the acoustic wave which moves upstream towards the shock. In Eq. (2),  $\mathbf{n}$  is the unit vector normal to the shock surface already computed in step 2.3,  $\tilde{a}_d^{t+\Delta t}$  and  $\tilde{u}_d^{t+\Delta t}$  are the values of sound speed and flow velocity computed by the unstructured solver in the downstream state of the shock nodes. Due to the upwind nature of the discretization used in the unstructured shock-

capturing code, the acoustic signal related to  $R_d$ , see Eq. (2), is correctly evolved, even though the individual quantities  $\tilde{a}_d^{t+\Delta t}$  and  $\tilde{u}_d^{t+\Delta t}$  may be wrong. The left hand side of Eq. (2) is therefore assumed to be correct, whereas the quantities on its right hand side are only provisional values (hence the tilde) that will be correctly updated in the subsequent step 2.5 by enforcing the jump relations.

#### 2.5. Step 2.5: shock nodes calculation

Each shock point is characterised not only by its upstream and downstream states, but also by its shock speed, which is directed along the local shock normal  $\mathbf{n}$  and has magnitude  $w$ .

The upstream state has been “correctly” updated at time level  $t + \Delta t$  by the shock-capturing solver, as explained in the previous step 2.4. Concerning the downstream state, only the upstream moving signal  $R_d^{t+\Delta t}$  was correctly computed. The complete downstream state ( $\rho_d^{t+\Delta t}$ ,  $p_d^{t+\Delta t}$  and  $u_d^{t+\Delta t}$ ) and the shock speed magnitude  $w$  (six scalar unknowns) are obtained by solving the following system of six scalar algebraic non-linear equations at each shock point:

$$\rho_d^{t+\Delta t} (u_{dn} - w) = \rho_u^{t+\Delta t} (u_{un} - w) \quad (3a)$$

$$\rho_d^{t+\Delta t} (u_{dn} - w)^2 + p_d^{t+\Delta t} = \rho_u^{t+\Delta t} (u_{un} - w)^2 + p_u^{t+\Delta t} \quad (3b)$$

$$\frac{\gamma}{\gamma - 1} \frac{p_d^{t+\Delta t}}{\rho_d^{t+\Delta t}} + \frac{(u_{dn} - w)^2}{2} = \frac{\gamma}{\gamma - 1} \frac{p_u^{t+\Delta t}}{\rho_u^{t+\Delta t}} + \frac{(u_{un} - w)^2}{2} \quad (3c)$$

$$\mathbf{u}_d^{t+\Delta t} - u_{dn} \mathbf{n} = \mathbf{u}_u^{t+\Delta t} - u_{un} \mathbf{n} \quad (3d)$$

$$\sqrt{\gamma \frac{p_d^{t+\Delta t}}{\rho_d^{t+\Delta t}}} + \frac{\gamma - 1}{2} u_{dn} = R_d^{t+\Delta t} \quad (3e)$$

The Eqs. (3a)–(3d) are the Rankine–Hugoniot jump relations and the last Eq. (3e) is nothing but Eq. (2) where  $R_d^{t+\Delta t}$  is computed using the provisional values  $\tilde{a}_d^{t+\Delta t}$  and  $\tilde{u}_d^{t+\Delta t}$ . In Eqs. (3)  $u_{dn} = \mathbf{u}_d^{t+\Delta t} \cdot \mathbf{n}$  and  $u_{un} = \mathbf{u}_u^{t+\Delta t} \cdot \mathbf{n}$  are the downstream and upstream flow velocity components along the shock-normal direction. The solution of the system of Eqs. (3) is obtained using a Newton–Raphson algorithm; only a few iterations are typically required to achieve convergence to machine zero.

#### 2.6. Step 2.6: shock displacement

The new position of each shock node at time  $t + \Delta t$  is computed using the shock speed calculated in step 2.5 by the following first-order-accurate integration formula:

$$\mathbf{x}_p^{t+\Delta t} = \mathbf{x}_p^t + w^{t+\Delta t} \mathbf{n} \Delta t. \quad (4)$$

The low order of accuracy of this formula does not affect the accuracy of the steady state solution which depends only on the space-accuracy of the gas-dynamic solver and the normal vector computation. The new position of the shock surface is obtained by the displacement of all shock points. The time step  $\Delta t$ , used in Eq. (4) to displace the shock surface, is appropriately reduced to keep the shock surface at time  $t + \Delta t$  within the hole dug into the mesh during step 2.1. By doing so, we ensure that during its motion the shock surface can only overtake phantom nodes, but it does not sweep any of the active points of the background mesh. This simplifies the shock-fitting algorithm by avoiding the re-computation of the mesh nodes eventually overtaken by the shock; see Ref. [40] for more details.

#### 2.7. Step 2.7: interpolation of the phantom nodes

In the previous steps, all nodes of the modified mesh have been updated to the new time level  $t + \Delta t$ . Nevertheless the phantom nodes of the background mesh, which were excluded

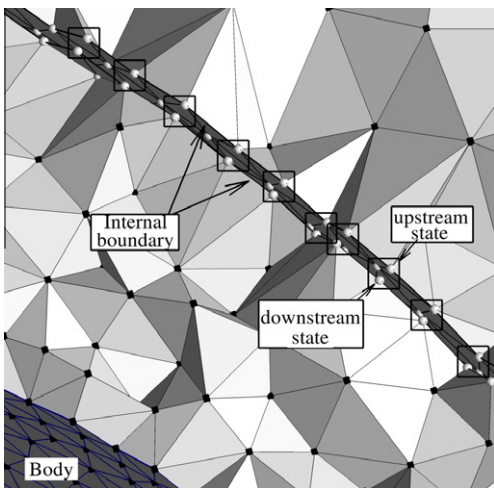


Fig. 2. The shock is treated as an internal boundary.



from the computation in step 2.1, have not been updated. It is important to update also these nodes at time  $t + \Delta t$  since at later times the shock front might move sufficiently far away from its current position that some of the phantom nodes may need to be re-inserted into the modified mesh. The update of the phantom nodes is performed by first locating each phantom node within a cell of the modified mesh and then by interpolating its state using the vertices of the tetrahedral cell that contains the phantom node.

### 3. Applications

The current version of the unstructured, three-dimensional, shock-fitting algorithm has been tested against three different test-cases in order to analyse its performances. The shock-fitting solutions have been compared with shock-capturing calculations obtained using the EulFS code on the same background meshes employed in the shock-fitting calculations.

As already described in Section 2, the shock-fitting algorithm uses as input the shock-capturing solution computed on the background mesh and a triangulated surface that initialises the position of the fitted shock. The generation of the initial shock surface, which has to be sufficiently close to the shock position at steady-state, is an important task that has been accomplished using various open-source software tools, as we shall now describe.

The shock detection algorithm described in Ma et al. [53] is used to extract a preliminary cloud of shock points from the shock-capturing solution. The MESHLAB [54] software is then used to perform a number of geometrical operations. First, the initial cloud of shock points is preprocessed to merge those shock points that are located too close to each other. Starting from this reduced set of shock points, the ball pivot surface reconstruction algorithm [55] is used to construct a triangulated shock surface; Laplacian filters are finally used to obtain a smooth shock surface. Since the MESHLAB software is designed for computer graphics applications, the surface triangulations are not always regular enough to be suited for CFD applications. Since the regularity of the shock-surface mesh is an important requirement to build a regular tetrahedralization around the shock-front, it was decided to further post-process the MESHLAB-generated shock surface using the YAMS [56] surface triangulator. YAMS is capable of generating isotropic and anisotropic triangulated surfaces starting from the shock surface triangulation already generated with MESHLAB. Local mesh refinement is used in regions of high surface curvature and the triangle area distribution can be preset, at least to a certain extent, by the user. Once the initial shock surface is available, the shock-fitting calculation can be advanced in time by repeated application of steps 2.1 to 2.7, as described in Section 2, until steady state is reached.

Note that in all test cases considered herein the shock surface, initialised according to the aforementioned procedure, does not move remarkably with respect to its initial position.

#### 3.1. Test case 1: hemisphere-cylinder

The first model problem consists in the  $M_\infty = 10$  flow over an hemisphere mounted on top of a circular cylinder. The free-stream flow is aligned with the body symmetry axis so that the flow field is axisymmetric. Despite its simple geometrical features, this test case represents a significant challenge for many of the algorithmic ingredients of this newly developed shock-fitting technique. Moreover, a reference solution is available [57,58]. It is worth underlining that the simulation has been performed on the full geometry without accounting for flow symmetry.

#### 3.1.1. Computational domain

The computational domain is an orthogonal parallelepiped having dimensions as shown in Fig. 3a. The radius of the hemisphere and the cylinder height are assumed as the reference length scale  $L$ . The axis of the cylinder is centred in the outflow plane and the nose of the hemisphere is located at a distance  $L$  behind the inflow plane. A two-levels mesh refinement has been used to study the hemisphere-cylinder test case. The background tetrahedral meshes have been generated using the GMSH mesh generator [59], starting from the triangulated surfaces of the body and of the external boundaries. Each bounding surface features an almost isotropic distribution of triangles, all having roughly the same area; this can be seen in Fig. 3a where the body surface (shown in red<sup>1</sup>) and a portion of the outer boundary are visible. The area of the triangular faces used to describe the body surface is about ten times smaller than that of the triangles that make up the outer boundaries. Therefore, the local mesh size of the background tetrahedral meshes is not uniform, but it increases as the outer boundaries are approached. The number of grid-points and tetrahedral elements of the coarse and fine background meshes are listed in Table 1 (in the column labelled “shock-capturing”) and the coarse background tetrahedral mesh is shown in Fig. 3a.

As described in Section 2.2, the shock-fitting algorithm uses a time-dependent mesh which differs from the background one due to the inclusion of the triangulated shock-surface. Fig. 3b shows the modified tetrahedral mesh obtained after the re-meshing which follows the insertion of the bow-shock surface, which is highlighted in blue. During time-integration, the shock surface moves, causing modifications of the tetrahedral grid, until it reaches its steady state position, corresponding to vanishing shock speed. Fig. 4 shows the evolution of the mesh from the initial tetrahedralization (shown in black) to the one at steady state (shown in red). Note that the differences between the initial and final tetrahedralizations are limited to the neighbourhood of the shock front. Table 1 also shows the number of grid-points and elements of the volumetric and shock-surface meshes used in the shock-fitting simulations once the shock surface has settled to its final position.

The shock-surface has been built using isotropic triangles having roughly the same area as those that make up the body surface. This constraint generates a shock-surface triangulation featuring a mesh size smaller than that of the background tetrahedral mesh. For this reason, the modified meshes generated by the shock-fitting algorithm are characterised by a larger number of cells and nodes with respect to the corresponding background meshes, as can be observed in Table 1. Even so, the additional nodes do not exceed 21% for the coarse discretization and 7% for the fine one. These figures are lowered to 17% and 3% as far as the number of extra cells is concerned.

Ideally, if we were able to re-mesh the shock surface using a variable mesh size, everywhere equal to the local mesh spacing of the background tetrahedral mesh, we could generate shock-fitting meshes having practically the same number of cells and nodes as the background mesh. In fact, the increase in the number of nodes due to the insertion of the shock surface would be balanced by the removal of the phantom nodes.

Finally, note that even though the bow shock reaches the outflow boundary of the computational domain, we have deliberately decided to fit only part of it; it is left to the shock-capturing code to capture the remainder of bow shock. By doing so, we do not need to deal with the intersection between the fitted shock surface and the far-field boundary triangulation.

<sup>1</sup> For interpretation of colour in Figs. 3, 7 and 8, the reader is referred to the web version of this article.

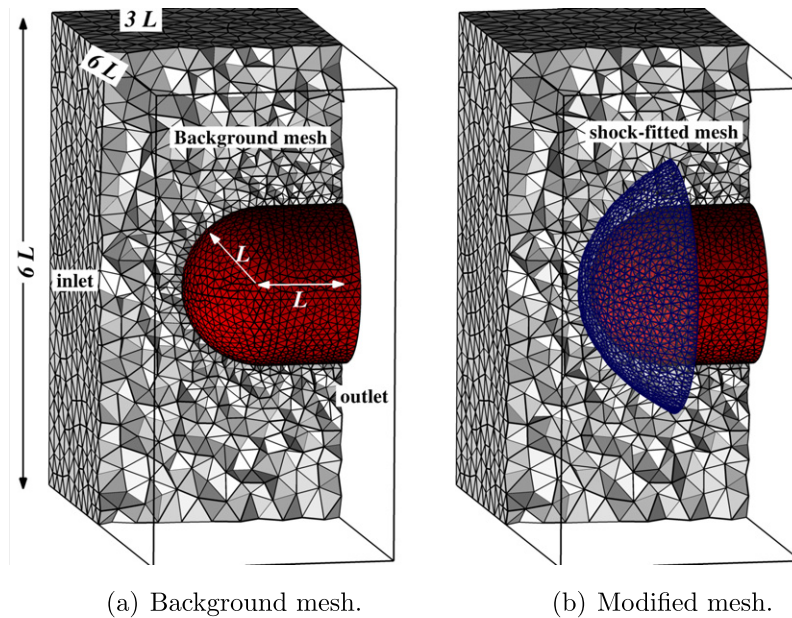


Fig. 3. Comparison between the background and the modified meshes used in the shock-fitting simulation.

**Table 1**  
Mesh statistics for the emisphère cylinder case.

Meshes	Shock-capturing		Shock-fitting		Shock surface	
	Coarse	Fine	Coarse	Fine	Coarse	Fine
#/ $10^3$ Points	12.26	158.3	14.84	168.8	1.5	2.8
#/ $10^3$ Elements	63.43	935.1	74.2	962.3	6.5	12.7

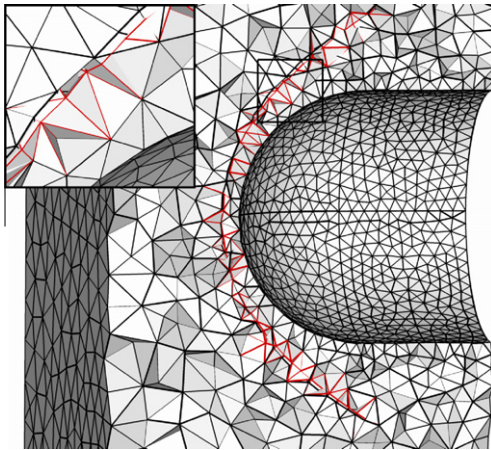


Fig. 4. Shock-fitting re-meshing on the coarse mesh. The black line is used for the initial tetrahedralization and the red line for the final one. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 3.1.2. Results

Both the shock-capturing and shock-fitting calculations reported in this section have been computed using nominally second-order-accurate schemes. Fig. 5 shows a detail of the normalised pressure iso-contours on the body in the subsonic region surrounding the stagnation point. The location of this region on the full body is shown, enclosed by a square box, in the small frame at the centre of Fig. 5.

The convergence of the coarse and the fine mesh shock-fitting solutions towards the reference one as well as the limited amount of numerical error associated with the shock-fitting solutions is proved by Fig. 5. On the other hand, even though the shock-capturing solutions converge towards the reference one, they are characterised by much higher numerical errors than the shock-fitting solutions on the corresponding discretization levels. Specifically, large differences in terms of both iso-line shape and maximum pressure value are observed in the stagnation region. It can be inferred that, while the shock-fitting solutions computed on the coarse and fine meshes are both nearly grid-converged, the shock-capturing solutions computed on the same discretization levels are not able to quantitatively reproduce the flow quantities, at least in the neighbourhood of the stagnation point. Pressure iso-contours levels within a section of the shock layer are shown in Fig. 6. The comparison between the two numerical solutions computed on the same grid level and the reference one is rather merciless. Indeed, the pressure distribution and the shock position predicted by the shock-fitting solution are almost superimposed to the reference one. On the contrary, large discrepancies can be seen between the shock-capturing solution and the reference one in the entire shock-layer.

### 3.2. Test case 2: IXV re-entry vehicle

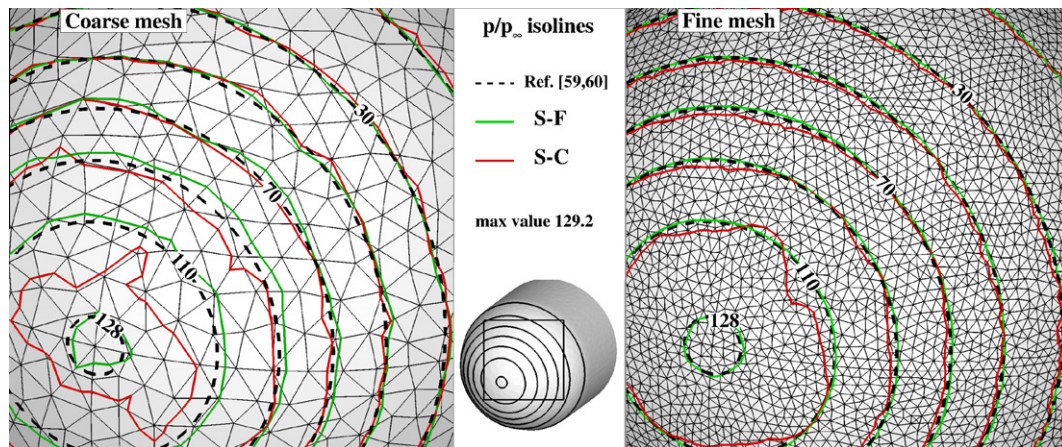
The second test case deals with the hypersonic flow past a new generation re-entry vehicle: the ESA IXV capsule [60,61]. IXV is a lifting reentry body, see Fig. 7a, equipped with two variable deflection flaps that allow an aerodynamically controlled re-entry. The selected flight conditions are:  $M_\infty = 25$  and  $45^\circ$  angle of attack, see Fig. 7a.

Two sets of calculations, corresponding to different flap deflection angles, have been performed. In the first flight condition the flaps are not deployed, whereas in the second one, the two flaps are set at different deflection angles:  $\delta_1 = 15^\circ$  and  $\delta_2 = 5^\circ$ , as shown in the small frame of Fig. 7a.

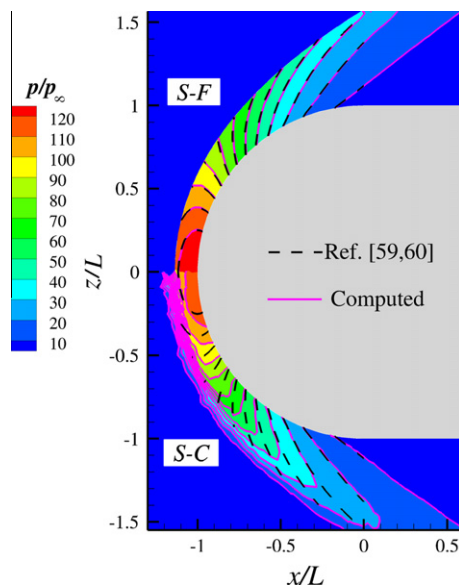
#### 3.2.1. Computational domain and grids

The computational domain used in the numerical simulation is shown in Fig. 7b. The body of the vehicle is coloured in blue,





**Fig. 5.** Normalised pressure  $p/p_\infty$  iso-contours on the body. Black lines are used for the reference solution, green lines for the shock-fitting solution and red lines for the shock-capturing solution. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 6.** Normalised pressure  $p/p_\infty$  iso-contour levels: black dashed lines are used for the reference solution and violet solid lines are used for the fine grid shock-fitting (upper half) and shock-capturing (lower half) solutions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

whereas the outer boundary of the computational domain is split in two sub-zones where different boundary conditions have been imposed: supersonic inflow conditions have been used on the red surface and extrapolation boundary conditions on the light red one.

Three meshes (a coarse, a medium and a fine) of decreasing mesh size have been generated. Fig. 8a shows the fine grid used to compute the shock-capturing solution and also as background mesh in the shock-fitting simulation. The body surface meshes and the outer boundaries of the computational domain have been generated using the YAMS [56] software. Over the outer boundary surface, all triangles have roughly the same area, whereas along the body surface the triangle area distribution is varied in order to reduce the mesh spacing in selected regions. These include areas characterised by high pressure values, such as the nose, and the windward side of the vehicle, where the flow gradients are larger than on its leeward side. In these regions the surface triangulation

features triangles of area roughly equal to one sixth of those used to mesh the far-field boundary. The ratio is increased to one fourth along the leeward side of the vehicle and over its flaps. The medium and fine grids have been generated by recursively reducing by a factor four the area of the body triangulation, whereas the far-field boundary triangulation is kept the same for all three grid levels. Starting from these three different surface triangulations, the background tetrahedral meshes have been generated using TetGen [50]. The characteristics of these three grid levels are reported in Table 2 for the zero flap deflection angle configuration; similar figures are obtained for the flight configuration with flaps deployed. The shock surface needed to initialise the shock-fitting computation has been obtained using the same procedure described for the previous test-case. An isotropic triangulation of the shock surface has been built, which is characterised by triangle areas nearly equal to those of the body surface triangles in the stagnation point region. By doing so, the mesh spacing along the shock surface is considerably smaller than that of the background tetrahedral mesh away from the body nose. For this reason, the modified meshes generated by the shock-fitting algorithm are characterised by an increased number of grid-points and cells with respect to the corresponding background meshes, as shown in Table 2. As in the previous test-case, the increase is fairly modest and does not exceed 5 % of the number of cells and 10% of the number of nodes of the background meshes.

Finally, as in the previous case, the fitted shock surface terminates before reaching the external boundary. The fitted shock surface at steady-state can be observed in Fig. 8b, coloured in grey.

### 3.2.2. Computational results

The IXV test-case has been computed by means of the shock-capturing and shock-fitting approaches on all three mesh levels described in Section 3.2.1. In addition to this, a reference solution has also been computed using the commercial software CFD++ [62] on a structured mesh made of about 4 million cells. We are confident that this reference solution is representative of the current capabilities of second-order-accurate shock-capturing discretizations. On the one hand, the multi-block structured mesh used to obtain this reference solution has been adapted in the shock region using a well established and validated mesh-adaptation procedure [63,64]. On the other hand, the numerical solution full fills all the requirements and checks that had been prescribed by the ESA designers for inclusion in the vehicle's aero-thermodynamic database.

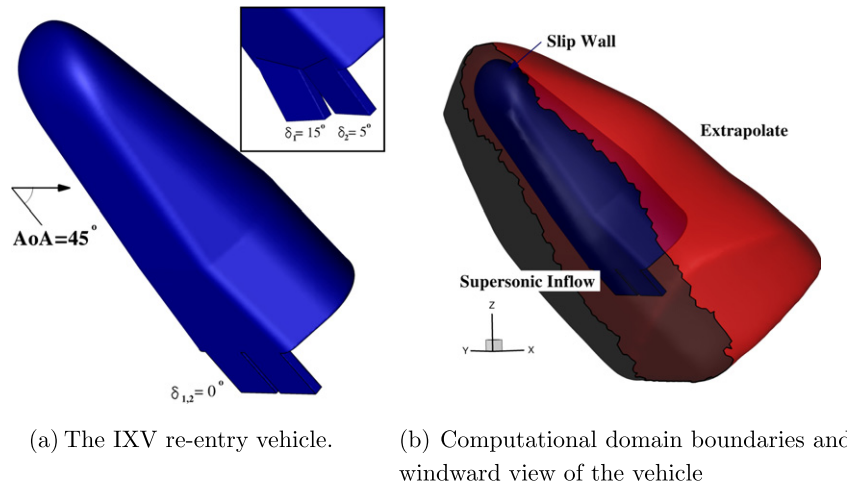


Fig. 7. Body and computational domain for the IXV vehicle.

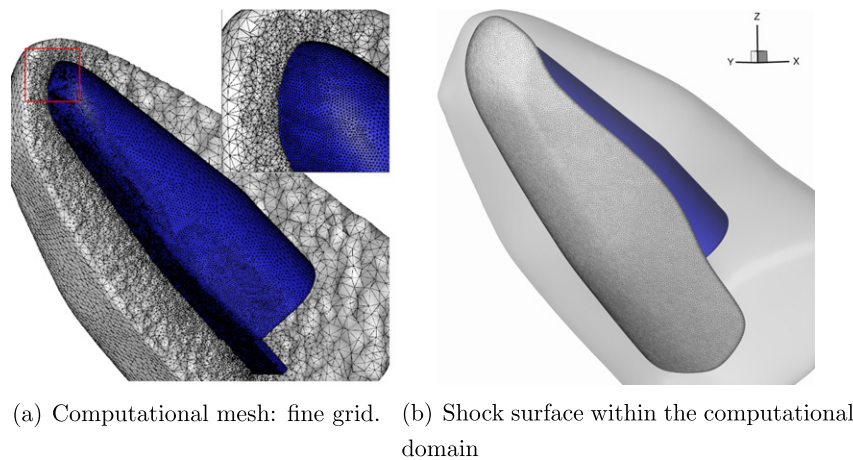


Fig. 8. Computational mesh for the IXV vehicle.

Table 2

Computational mesh statistics for the IXV vehicle; zero flap deflection angle configuration.

Meshes	Shock-capturing			Shock-fitting			Shock surface		
	Coarse	Medium	Fine	Coarse	Medium	Fine	Coarse	Medium	Fine
#/ $10^3$ Points	151	217	365	164	219	404	5.6	10.2	22.1
#/ $10^3$ Elements	888.3	1200	2169	910.3	1247	2271	10.9	20	43.9

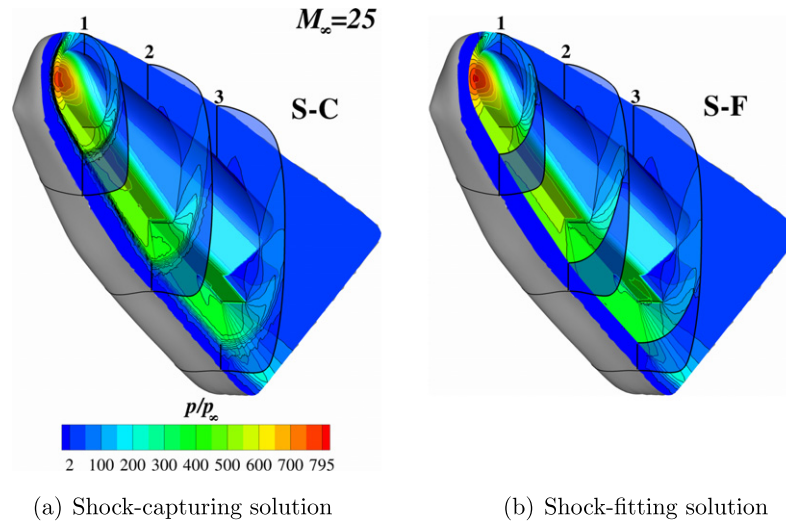
The shock-fitting solutions have been obtained using both first- and second-order-accurate schemes. On the contrary, only the first-order-accurate scheme could be used in the shock-capturing calculations: the shock-capturing calculations using the second-order-accurate scheme blew up due to numerical problems, such as the occurrence of negative densities within the captured shock. Note that numerical problems with the shock-capturing approach have been encountered even using the well-validated CFD++ code on the same tetrahedral meshes described in Section 3.2.1: the first-order-accurate calculation converged to steady-state without problems, but when the computation was switched to the second-order-accurate discretization scheme, a carbuncle appeared, completely spoiling the solution inside the entire shock layer.

We shall now first analyse the flight condition characterised by zero flap deflection. Fig. 9 compares the normalised pressure distribution

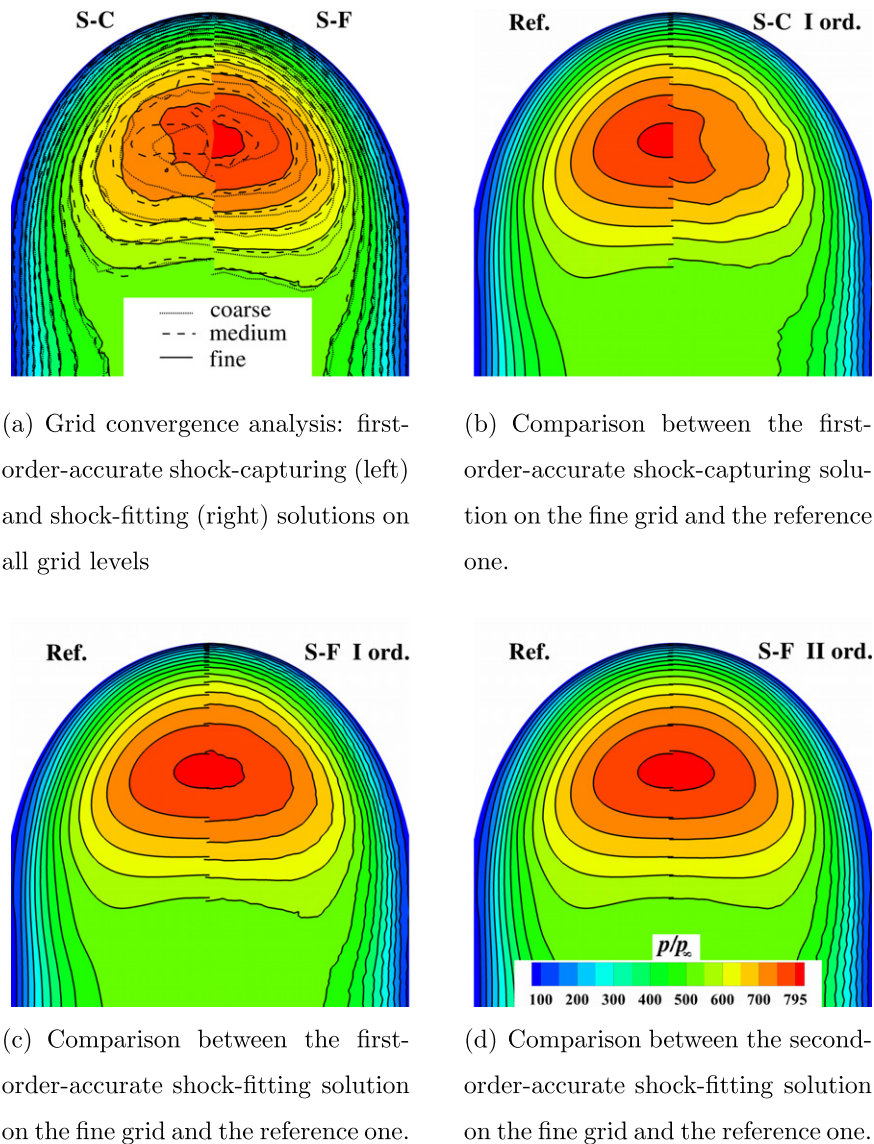
on the body surface and within three cross-flow planes computed by means of shock-capturing and shock-fitting. Both solutions are first order accurate and have been computed on the fine grid level. The comparison allows to appreciate the improvements in shock resolution brought in by the shock-fitting calculation. The unstructured shock-capturing solution, which is characterised by a large shock thickness (compare the two solutions within the cross-flow planes 1, 2 and 3), shows visible differences with respect to the shock-fitting solution inside the entire shock layer. These differences become even more pronounced in the stagnation region.

A closer view of the stagnation region is shown in Fig. 10, which allows to draw a comparison among the various unstructured grid solutions and the reference solution. Fig. 10a shows the effects of mesh refinement upon the first-order-accurate shock-capturing





**Fig. 9.** Normalised pressure iso-contours and floods for the IXV vehicle.



**Fig. 10.** Black solid lines and flood contours are used for the reference and fine grid solutions. Dashed and dotted lines are respectively used for the medium and coarse grid solutions.

and shock-fitting solutions; the reference solution is compared with the first-order-accurate, fine-grid, shock-capturing solution in Fig. 10b and with the first-order-accurate, fine-grid, shock-fitting solution in Fig. 10c. Finally, Fig. 10d compares the second-order-accurate shock-fitting solution computed on the fine mesh with the reference one. In Fig. 10a the shock-fitting solutions clearly show convergence towards a grid-independent solution over the entire nose region, so that the fine-grid, shock-fitting solution can be considered nearly grid-independent. On the contrary, the shock-capturing solutions do not display the same behaviour, particularly in the neighbourhood of the stagnation point where the shock-capturing solutions computed on the three grid levels are characterised by large mutual differences and do not seem to converge towards a grid-independent solution. The comparison, see Fig. 10b–d, among the various solutions computed on the fine unstructured grid and the reference one confirms the findings of the previous convergence analysis. Fair agreement between the first-order-accurate shock-fitting solution and the reference one can be seen in Fig. 10c; the agreement further improves when the order-of-accuracy is raised to second, as shown in Fig. 10d. On the other hand, significant differences exist, see Fig. 10b, between the reference solution and the first-order-accurate shock-capturing one, which predicts a stagnation pressure value 13% larger than the reference one. It is worth emphasising that the improvement exhibited by the first-order-accurate shock-fitting solution with respect to the shock-capturing one is entirely due to the numerical treatment of the bow shock, since within the shock layer, the same numerical scheme and almost identical grids are used for both computations.

The flight configuration with flaps deployed differs from the one just analysed due to the presence of an embedded shock which arises in the proximity of the flap hinge and interacts with the bow shock. This flow feature gives us the chance to demonstrate the hybrid functionality of the proposed shock-fitting approach whereby the bow shock has been fitted whereas the embedded shock has been captured. The computed flow-field differs from the one previously analysed only in the flap region and, therefore, the analysis of the results will be restricted to this area. Fig. 11 compares the normalised pressure iso-contours computed within two stream-wise planes (1 and 2 in the small frame of Fig. 11) that cut the two flaps in the middle. The comparison between the shock-capturing and the shock-fitting calculations clearly reveals the differences; in particular, the shock-fitting solution predicts

higher pressure values than the shock-capturing one. Again, this is due to the fact that fitting the bow shock gives a cleaner description of the shock layer and even though the embedded shock is captured in both calculations, the better flow-field resolution upstream of the embedded shock produces a different pressure jump and, therefore, higher pressure peaks downstream of the embedded shock. As expected, the differences observed in the pressure distribution within the flow-field also affect the pressure distribution on the flap surface, as shown in Fig. 12. Specifically, Fig. 12 compares the reference solution with the shock-capturing and shock-fitting solutions computed on the finest grid level. The comparison clearly reveals the remarkable improvements that even a partial use of the shock-fitting technique can bring. Note, in particular, that the second-order-accurate shock-fitting solution almost reproduces the reference one.

The present test case clearly shows that the proposed shock-fitting algorithm can be applied without problems to complex geometries. Moreover, the results we have obtained show remarkable improvements upon an un-adapted shock capturing calculation computed on a grid featuring a comparable number of grid-points and cells.

### 3.3. Test case 3: cylinder with an hemispherical nose and a conical flare

In the simulations described in the previous two sections only the bow shock has been fitted; whenever embedded shocks are also present, as in the case of the IXV vehicle, their “capture” is left to the shock-capturing code.

It is worth emphasising that the current version of the 3D shock-fitting algorithm is capable of fitting more than an individual shock surface, even though, at present, the interaction between the fitted shocks has to be captured, rather than fitted. This shortcoming is not related to difficulties in modelling shock-shock interactions in three dimensions, since two-dimensional interactions between fitted discontinuities have already been successfully addressed in two space dimensions [39,40], but has to do with limitations of the surface mesh generation tools we have been working with.

The capability of dealing with more than a single fitted shock surface is here demonstrated by reference to the high speed flow past a blunt object which consists in a cylinder with an hemispherical nose and a conical flare forming an angle of 30° with respect to

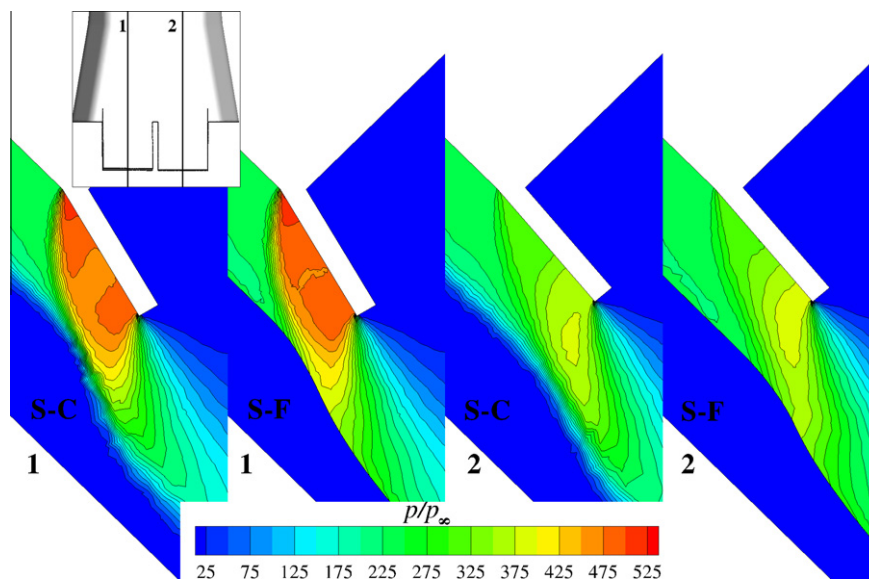


Fig. 11. Normalised pressure iso-contours on two sections (denoted in the top corner of the picture): shock-capturing solution vs. hybrid solution.

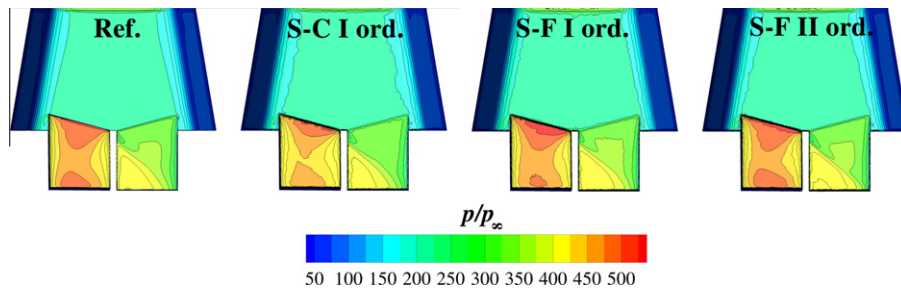


Fig. 12. Normalised pressure iso-contours on the vehicle surface: shock-capturing solution vs. hybrid solution.

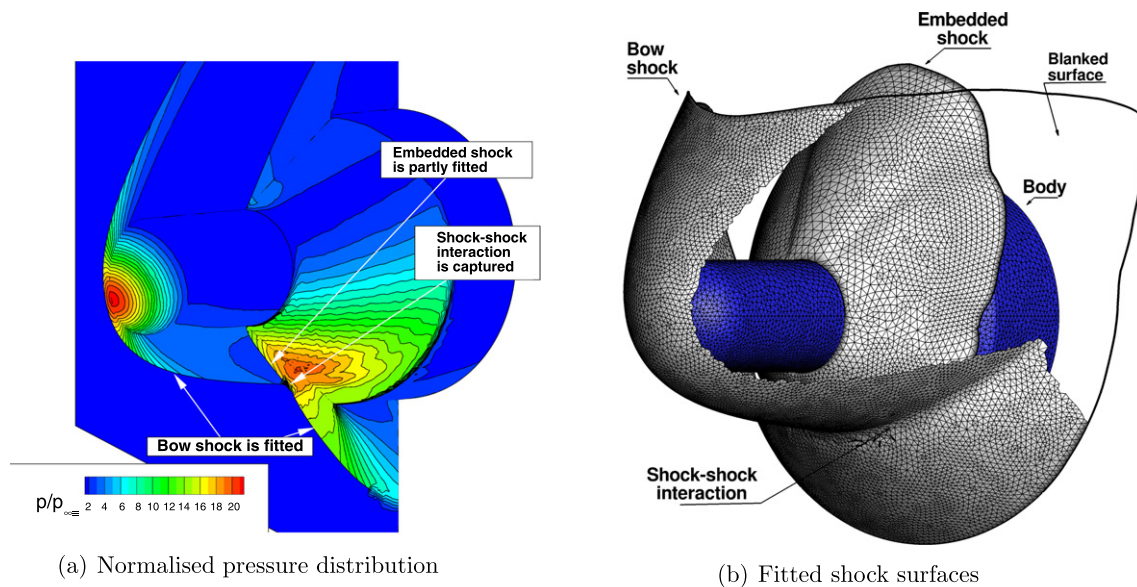


Fig. 13. Supersonic ( $M_\infty = 4.04$ ) flow past a cylinder with conical flare.

the cylinder's axis. The free-stream conditions are: Mach number  $M_\infty = 4.04$  and angle of attack equal to  $20^\circ$ . This flow configuration has been studied both experimentally and numerically by Houtman et al. [65]: it is characterised by a bow shock and an embedded shock which originates at the cylinder-cone junction; these two shocks interact along the windward side of the body, see Fig. 13a, giving rise to a type VI shock-shock interaction, as classified by Edney [66].

Fig. 13a shows the normalised pressure distribution on the body surface and on the symmetry plane and Fig. 13b shows the fitted surfaces (at steady state) of both the embedded and bow shocks. In the simulation shown in Fig. 13 the interaction between the two shocks has been handled by terminating the fitted surface of the embedded shock shortly before it reaches the bow shock and letting the shock-capturing code capture the shock-shock interaction. This can be seen from the pressure iso-contours displayed in Fig. 13a.

Further details concerning this specific test-case and, in particular, a comparison between the shock-capturing and shock-fitting solutions, can be found in [67,63].

#### 4. Conclusions

This paper describes the extension to three spatial dimensions of the two-dimensional, unstructured, shock-fitting technique proposed in [38–40]. The present shock-fitting algorithm can be viewed as a local re-meshing technique which provides a time-

dependent tetrahedral mesh in which the fitted shock fronts are double-sided triangulated surfaces which behave as interior boundaries for a shock-capturing code used to discretise the governing PDEs. The Rankine–Hugoniot jump relations govern the motion of the shock surface and allow to enforce appropriate boundary conditions for the shock-capturing code along the downstream side of the shock.

Despite the fact that conventional CFD wisdom has it that [10] “fitting the shock as a surface of discontinuity seems to be prohibitively complex in three dimensions”, the proposed technique turns out to be relatively easy to implement.

On one hand, due to the loose coupling between the shock-fitting algorithm and the shock-capturing solver, the latter could be used as a black box. Therefore, virtually any vertex-centred, shock-capturing code working with tetrahedral cells could be coupled with the proposed shock-fitting technique without requiring any modification.

On the other hand, surface mesh generation over the shock surfaces and volumetric mesh generation around the shock, the two key ingredients that make up the core of the shock-fitting algorithm, have both been accomplished by means of efficient public domain software.

Moreover, the coupling of the shock-fitting technique with a shock-capturing gas-dynamic solver allows a hybrid mode of operation whereby some of the shocks (or part of them) are fitted and the remaining ones (or the remaining parts) are captured. The use of this feature simplifies the treatment of the fitted shock on exter-



nal boundaries and also allows the computation of shock-shock interaction between a fitted shock and a captured one.

A geometrically simple test-case, which consists in the hypersonic flow past an hemisphere, has first been considered due to the availability of a reference solution. A more complex flow case, namely the hypersonic flow past the IXV re-entry vehicle, allowed us to demonstrate the hybrid operational mode of the proposed technique. Finally, a supersonic flow case was used to show that the current version of the algorithm allows to simultaneously deal with more than one fitted shock surface, a feature which fully justifies the use of a floating shock-fitting approach in contrast to the boundary shock-fitting approach which might suffice when only the bow shock is fitted. For most of these test-cases the results have been compared with those obtained using the same flow solver operating in shock-capturing mode and using grids featuring nearly the same number of grid-points and tetrahedra as the shock-fitting grids. In the cases examined herein it has been shown that the shock-fitting technique is capable of significantly improving the solution quality, particularly in the stagnation regions.

It is true, however, that even if the unstructured grid framework allows to relieve some of the algorithmic difficulties that decreed the dismissal of the shock-fitting technique in the structured-grid framework, the present technique is not yet fully developed and therefore suffers from some limitations. For instance, shock detection is an important algorithmic task which we have deliberately not addressed up to now since we have been considering only steady problems, where a preliminary shock-capturing calculation provides a reasonable initial guess for the shock location. Nonetheless, shock detection has been extensively addressed in the past shock-fitting literature and we will certainly take advantage of this large knowledge base in our future work.

Viscous flows pose additional challenges. The present shock-fitting technique has already been used for viscous flow calculations [68] that do not involve the interaction between the fitted shocks and the boundary layers. Whenever shock-boundary layer interactions occur, it becomes necessary to account for the fact that the fitted shocks do not originate at the wall, but rather form at some distance from it due to the coalescence of compression waves. Even though the present shock-fitting algorithm is not yet capable of modelling this phenomenon, the correct treatment of shock formation from the coalescence of compression waves in viscous flow has already been successfully addressed in the past by different researchers working in the field of shock-fitting [69–71]. We are therefore confident that its implementation within the unstructured framework should not pose any major challenge.

Even more challenging than the two aforementioned issues is the capability of automatically identifying changes in the flow topology, such as those that inevitably occur in un-steady shock reflections. However, as mentioned in the Introduction, this has more to do with topology than fluid mechanics and some expertise from other areas of computer science might prove helpful.

Beside the aforementioned issues, there are others that are peculiar to the three-dimensional case. One of the most important ones, which we have not dealt with in the present paper is shock-surface re-meshing. Surface re-meshing becomes an essential ingredient of the algorithm whenever the shock surfaces move and deform considerably during their time evolution. This was not the case with the steady flow-cases examined herein since a good initial guess of the final shock shape and location was made available by the preliminary shock-capturing calculation. Surface re-meshing will have to be taken into account when dealing with un-steady calculations; work on this issue is currently under way and preliminary results have been reported in [67,63].

As a closing remark, we note that the results shown in this paper can have a remarkable impact in the framework of the unstructured CFD solvers, since they prove that a number of drawbacks

incurred by tetrahedral mesh discretizations in hypersonic applications can be overcome by means of shock-fitting.

For all these reasons, the present shock-fitting technique, if further developed, may represents a valuable alternative to the universally adopted shock-capturing paradigm.

## Acknowledgments

This research has been carried out in the framework of a Project supported by the Italian Ministry of Education, University and Research (M.I.U.R.) under Grant PRIN-2007.

The IXV capsule calculations have been published with permission from the ESA IXV Project.

## References

- [1] Gnoffo P, White JA. Computational aerothermodynamic simulation issues on unstructured grids. AIAA-2004-2371; 2004.
- [2] Chandler GV, Barnhardt MD, Drayna TW, Nompelis I, Peterson DM, Subbareddy P. Unstructured grid approaches for accurate aerothermodynamic simulations. In: 18th AIAA computational fluid dynamics conference. Miami, FL; 2007. Paper 2007-3959.
- [3] Gnoffo PA. Simulation of stagnation region heating in hypersonic flow on tetrahedral grids. In: 18th AIAA computational fluid dynamics conference, 25–28 June, Miami, FL; 2007. Paper 2007-3960.
- [4] Gnoffo PA. Multi-dimensional, inviscid flux reconstruction for simulation of hypersonic heating on tetrahedral grids. In: 47th AIAA aerospace sciences meeting. Orlando, Florida; 2009. Paper 2009-599.
- [5] Gnoffo PA. Update to multi-dimensional flux reconstruction for hypersonic simulation on tetrahedral grids. In: 48th AIAA aerospace sciences meeting. Orlando, Florida; 2010. Paper 2010-1271.
- [6] Carpenter MH, Casper JH. Accuracy of shock capturing in two spatial dimensions. AIAA J 1999;37(9):1072–9.
- [7] Lee TK, Zhong X. Spurious numerical oscillations in simulation of supersonic flows using shock-capturing schemes. AIAA J 1999;37(3):313–9.
- [8] Robinet JC, Gressier J, Casalis G, Moschetta JM. Shock wave instability and the carbuncle phenomenon: same intrinsic origin? J Fluid Mech 2000;417:237–63.
- [9] Pandolfi M, D'Ambrosio D. Numerical instabilities in upwind methods: analysis and cures for the carbuncle phenomenon. J Comput Phys 2001;166(2):271–301.
- [10] Roe P, Nishikawa H, Ismail F, Scalabrin L. On carbuncles and other excrescences. In: 17th AIAA computational fluid dynamics conference. Toronto, Ontario, Canada; 2005. p. 1–10. Paper 2005-4872.
- [11] Bonhaus D. A higher order accurate finite element method for viscous compressible flows. Ph.D. thesis. Virginia Polytechnic Institute and State University; 1998.
- [12] Roy CJ, McWhorter-Payne MA, Oberkampf WL. Verification and validation for laminar hypersonic flowfields Part 1: verification. AIAA J 2003;41(10):1934–43.
- [13] Suresh A. Interaction of a shock with a density disturbance via shock fitting. J Comput Phys 2005;206(1):6–15.
- [14] Rumsey C, van Leer B. A grid-independent approximate riemann solver with applications to the euler and Navier–Stokes equations. In: 29th AIAA, aerospace sciences meeting, Reno, NV; 1991. p. 1991.
- [15] Dadone A, Grossman B. A multi-dimensional upwind scheme for the Euler equations. In: Napolitano M, Sabetta F, editors. Thirteenth international conference on numerical methods in fluid dynamics. Lecture notes in physics, vol. 414. Berlin, Heidelberg: Springer; 1993. p. 95–9.
- [16] Deconinck H, Paillère H, Struijs R, Roe P. Multidimensional upwind schemes based on fluctuation-splitting for systems of conservation laws. J Comput Mech 1993;11(5/6):323–40.
- [17] 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. Comput Mech 1999;23:199–208.
- [18] Abgrall R. Residual distribution schemes: current status and future trends. Comput Fluids 2006;35(7):641–69 [Special Issue Dedicated to Professor Stanley G. Rubin on the Occasion of his 65th Birthday].
- [19] Bonfiglioli A. Validation of fluctuation splitting schemes for hypersonic calculations; 2004. Unpublished, The University of Michigan. <http://www.unibas.it/utenti/bonfiglioli/Papersaldo/hefss.pdf>
- [20] Casper J, Carpenter MH. Computational considerations for the simulation of shock-induced sound. SIAM J Sci Comput 1998;19(3):813–28 [Also NASA Langley RC Technical Memorandum 110222].
- [21] Luke E. On robust and accurate arbitrary polytope cfd solvers. In: 18th AIAA Computational Fluid Dynamics Conference, 25–28 June, 2007, Miami, FL Paper 2007-3956.
- [22] Smith TM, Barone MF, Bond RB. Comparison of reconstruction techniques for unstructured mesh vertex centered finite volume schemes. In: 18th AIAA Computational Fluid Dynamics Conference, 25–28 June, 2007, Miami, FL Paper 2007-3958.

- [23] Emmons HW. The numerical solution of compressible fluid flow problems. NACA-TN 932. NASA; 1944.
- [24] Moretti G, Abbott M. A time-dependent computational method for blunt body flows. *AIAA J* 1966;4:2136–41.
- [25] Moretti G. Computation of flows with shocks. *Ann Rev Fluid Mech* 1987;19:313–7.
- [26] Salas MD. Shock-fitting method for complicated two-dimensional supersonic flows. *AIAA J* 1976;14:583–8.
- [27] Yamamoto O, Anderson DA, Salas MD. Numerical calculations of complex mach reflection. In: *AIAA fluid dynamics, plasma dynamics, and lasers conference*. Snowmass, CO; 1984. AIAA-1984-1679.
- [28] Marconi F, Salas M. Computation of three dimensional flows about aircraft configurations. *Comput Fluids* 1973;1:185–95.
- [29] Hartwich PM. Fresh look at floating shock fitting. *AIAA J* 1990;29:1084–91.
- [30] Dadone A, Fortunato B. Three-dimensional flow computations with shock fitting. *Comput Fluids* 1994;23:539–50.
- [31] Nasuti F, Onofri M. Analysis of unsteady supersonic viscous flows by a shock-fitting technique. *AIAA J* 1996;34:1428–34.
- [32] Nasuti F, Onofri M. Viscous and inviscid vortex generation during startup of rocket nozzles. *AIAA J* 1998;36:809–15.
- [33] Kopriva DA. Domain decomposition with both spectral and finite difference methods for the accurate computation of flows with shocks. *J Appl Numer Math* 1989;6:141–51 [Spectral multi-domain methods (Paris, 1988)].
- [34] Kopriva DA. Shock-fitted multidomain solution of supersonic flows. *Comput Methods Appl Mech Eng* 1999;175:383–94.
- [35] Zhong X. High-order finite-difference schemes for numerical simulation of hypersonic boundary-layer transition. *J Comput Phys* 1998;144(2):662–709.
- [36] Rawat PS, Zhong X. On high-order shock-fitting and front-tracking schemes for numerical simulation of shock-disturbance interactions. *J Comput Phys* 2010;229(19):6744–80.
- [37] Sesterhenn J, Dohogne JF, Friedrich R. Direct numerical simulation of the interaction of isotropic turbulence with a shock wave using shock-fitting. *CR Méc* 2005;333(1):87–94 [High-order methods for the numerical simulation of vortical and turbulent flows].
- [38] Paciorri R, Bonfiglioli A. A shock-fitting technique for 2D unstructured grids. *Comput Fluids* 2009;38(3):715–26.
- [39] Ivanov M, 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):271–84.
- [40] Paciorri R, Bonfiglioli A. Shock interaction computations on unstructured, two-dimensional grids using a shock-fitting technique. *J Comput Phys* 2011;230(8):3155–77.
- [41] Moretti G. Numerical studies of 2-dimensional flows. NASA-CR-3930; NAS 1.26:3930; GMAF, Inc.; Freeport, NY; 1985. Prepared for NASA Langley Research Center under contract NAS1-17263. <<http://hdl.handle.net/2060/19850026847>>.
- [42] Moretti G. A general-purpose technique for two-dimensional transonic flows. NASA-CR-194186; GMAF, Inc.; Freeport, NY; 1987b. Prepared for NASA Lewis Research Center under contract NAS3-24540. <<http://hdl.handle.net/2060/19940003743>>.
- [43] Salas M. A shock-fitting primer. CRC applied mathematics & nonlinear science, 1st ed., vol. 1. Chapman & Hall; 2009.
- [44] Moretti G. Thirty-six years of shock fitting. *Comput Fluids* 2002;31:719–23.
- [45] Bonfiglioli A. Fluctuation splitting schemes for the compressible and incompressible Euler and Navier–Stokes equations. *Int J Comput Fluid Dyn* 2000;14(1):21–39.
- [46] Bonfiglioli A, Campobasso MS, Carpentieri B. Parallel unstructured three-dimensional turbulent flow analyses using efficiently preconditioned Newton–Krylov solver. In: *Proceedings of the 19th AIAA computational fluid dynamics*; 2009. AIAA 2009-4137.
- [47] Paillère H, Deconinck H. Compact cell vertex convection schemes on unstructured meshes. *Notes on numerical fluid mechanics*, vol. 57. Vieweg; 1996. p. 1–46.
- [48] Paillère H, Deconinck H. Multidimensional upwind residual distribution schemes for the 2D Euler equations. *Notes on numerical fluid mechanics*, vol. 57. Vieweg; 1996. p. 51–112.
- [49] Guzik S, Groth C. Comparison of solution accuracy of multidimensional residual distribution and Godunov-type finite-volume methods. *Int J Comput Fluid Dyn* 2008;22(1–2):61–83.
- [50] Si H. TetGen a quality tetrahedral mesh generator and a 3D Delaunay triangulator; 2011. <<http://tetgen.berlios.de/>>.
- [51] Shewchuk J. Constrained Delaunay tetrahedralizations and provably good boundary recovery. In: *Eleventh international meshing roundtable*. Ithaca, New York: Sandia National Laboratories; 2002. p. 193–204.
- [52] Bonfiglioli A, Grottadaurea M, Paciorri R, Sabetta F. An unstructured, three-dimensional, shock-fitting solver for hypersonic flows. In: *40th Fluid dynamics conference and exhibit*; 2010. AIAA 2010-4450.
- [53] Ma KL, Van Rosendale J, Vermeer W. 3d shock wave visualization on unstructured grids. In: *VVS '96: proceedings of the 1996 symposium on volume visualization*. Piscataway, NJ, USA: IEEE Press; 1996. p. 87–104. ISBN:0-89791-865-7.
- [54] Callieri M, Corsini M, Dellepiane M, Ganovelli F, Pietroni N, Tarini M. Meshlab; 2011. <<http://meshlab.sourceforge.net/>>.
- [55] Bernardini F, Mittleman J, Rushmeier H, Silva C, Taubin G. The ball-pivoting algorithm for surface reconstruction. *IEEE Trans Visual Comput Graph* 1999;5(4):349–59.
- [56] Frey PJ. Yams: a fully automatic adaptive isotropic surface remeshing procedure. Rapport technique n° 0252; Institut National de Recherche en Informatique et en Automatique (I.N.R.I.A.); INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France); 2001. ISSN 0249-0803. <<http://www.ann.jussieu.fr/frey/software.html>>.
- [57] Lyubimov AN, Rusanov VV. Gas flow past blunt bodies. Part I: calculation method and flow analysis. Tech. Translation TT F-714; NASA; Washington DC; 1973a. Translation of Nauka Press, Moscow; 1970.
- [58] Lyubimov AN, Rusanov VV. Gas flow past blunt bodies. Part II: Tables of gasdynamic functions. Tech. Translation TT F-715; NASA; Washington DC; 1973b. Translation of Nauka Press, Moscow; 1970.
- [59] Geuzaine C, Remacle J. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *Int J Numer Methods Eng* 2009;79(11):1309–31.
- [60] Tumino G, Angelino E, Leleu F, Angelini R, Plotard P, Sommer J. The IXV project: the ESA re-entry system and technologies demonstrator paving the way to european autonomous space transportation and exploration endeavours. In: *Future launchers preparatory programme (FLPP) 3rd FLPP industrial workshop*. Noordwijk, The Netherlands; 15 October 2008.
- [61] Roncioni P, Ranuzzi G, Marini M, Cosson E. Hypersonic CFD characterization of the IXV vehicle. In: *West-east high speed flow field conference*. Moscow; 19–22 November 2007.
- [62] Metacomp Technologies. Cfd++ homepage. 2010. <<http://www.metacomptech.com/index.html>> [accessed 10.11.10].
- [63] Bonfiglioli A, Paciorri R, Di Mascio A. The role of mesh generation, adaptation, and refinement on the computation of flows featuring strong shocks. *Modell Simul Eng* 2012;1–15. <<http://dx.doi.org/10.1155/2012/631276>>. <<http://downloads.hindawi.com/journals/mse/2012/631276.pdf>>.
- [64] Paciorri R, Onofri M, Cardillo D, Cosson E, Binetti P, Walloschek T. Numerical assessment of wall catalytic effects on the IXV surface. In: *Sixth European symposium on aerothermodynamics for space vehicles*, vol. SP-659. European Space Agency, (Special Publication) ESA SP; 2009.
- [65] Houtman EM, Bannink WJ, Timmerman BH. Experimental and numerical investigation of the high-supersonic flow around an axi-symmetric blunt-cylinder-flare model. In: Hunt JJ, editor. *Aerothermodynamics for space vehicles*, vol. 367. ESA Special Publication; 1995. p. 517–22.
- [66] Edney B. Anomalous heat transfer and pressure distributions on blunt bodies at hypersonic speeds in the presence of an impinging shock. TR 115. Flygtekniska Forsökanstalten (The Aeronautical Research Institute of Sweden). Stockholm; 1968.
- [67] Bonfiglioli A, Grottadaurea M, Paciorri R, Sabetta F, Bianchi D, Onofri M. Numerical simulation of hypersonic flows past three-dimensional blunt bodies through an unstructured shock-fitting solver. In: *17th AIAA international space planes and hypersonic systems and technology conference*, USA: American Institute of Aeronautics and Astronautics (AIAA); 2011–2288.
- [68] Bonfiglioli A, Paciorri R. Comparative study of stagnation point anomalies by means of shock capturing and shock fitting unstructured codes. In: *Proceedings of the 6th European symposium on aerothermodynamics for space vehicles*. European Space Agency; NOORDWIJK; NLD: ESA Communication Production Office; 2009.
- [69] Moretti G, Marconi F, Onofri M. Shock-boundary layer interaction by shock fitting. In: *13th international conference on numerical methods in fluid dynamics*. Lecture notes in physics, vol. 57. Springer-Verlag; 1993. p. 345–9.
- [70] Nasuti F, Paciorri R, Onofri M. Computation of turbulent supersonic base flows by a shock-fitting quasi-linear solver. In: *Proceedings of the 1999 3rd ASME/JSME joint fluids engineering conference, FEDSM'99*, San Francisco, California, USA. 18–23 July 1999 (CD-ROM). FEDSM99-7316; 1999.
- [71] Nasuti F, Onofri M. Analysis of in-flight behavior of truncated plug nozzles. *J Propul Power* 2001;17(4):809–17.