JAVIER HERRERA MONTOJO

SHOCK ESTIMATION IN SUPERSONIC VEHICLES

Submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

2017

UNIVERSITY OF STRATHCLYDE
DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Javier Herrera Montojo

August 15, 2017

CONTENTS

ABSTRACT

Improving the design of future high-Mach vehicles is crucial to enhance performance, safety and sustainability of future air travel. Advanced designs can be realized by improving low fidelity modelling in the context of Multidisciplinary Design Optimization (MDO) on one side and making the use of high fidelity tools more efficient in the context of Multi-fidelity Design Approaches (MFDA) on the other.

This thesis presents the formulation of an advanced low order model for the estimation of the shock structure generated by vehicles flying in the supersonic and hypersonic regimes. Taking as inputs the geometry and the flow conditions, the proposed approach addresses, in one cohesive methodology, attached and detached shocks in two and three dimensions as well as shock structures composed of multiple shocks. The procedure is based on classical supersonic flow theories and has been verified against computational fluid dynamics simulations.

The proposed methodology allows for a cost-effective estimation of shock wave patterns and their impingement on the vehicle surface on one side, while on the other it can be used to realize a-priori shock-fitted meshes reducing some of the uncertainty while generating them for high-fidelity CFD simulations.

1 INTRODUCTION

As is the case with any complex system, the design of aerospace vehicles is challenging and plagued with conflicting objectives. The need of low mass structures and components while maintaining structural integrity, the push for higher aerodynamic efficiency inside a given thermal envelope, robust design versus peak nominal performance and similar other objectives. All of these factors drive the design in opposite directions and compromises have to be made.

During the conceptual and preliminary design phases, multiple trade off analysis have to be performed to select a suitable configuration. This process has traditionally being done by experienced engineers, drawing from practical knowledge on the development of similar systems.

This classical approach has been formalized and polished by the advent of multi-disciplinary design optimization (MDO). MDO refers to the use of optimization during design, taking into account multiple aspects of the problem concurrently, as opposed to the sequential optimization of each discipline. MDO has the potential to generate better designs, giving some insight on the interrelationships between the disciplines involved in the problem[41].

MODELLING IN THE CONTEXT OF MDO

The use of MDO requires the evaluation of the different discipline models a high number of times during the optimization process. This places constraints on the evaluation speed of the models used.

One approach to increase computational efficiency is the use of low order models. These are computationally cheap methods that approximate the behaviour of a complex system by introducing simplifications of the underlying physics. These models are either derived from full models through order reduction, developed from first principles or inferred from empirical or engineering correlations.

While there are situations where the comparatively low fidelity of low order models is enough to characterize the behaviour of the system and to guide the optimization in a meaningful way, sometimes their limitations might drive the process to designs that are considered infeasible in later more detailed analysis. This is the main motivation behind multi-fidelity design approaches (MFDA). Here, low and high fidelity models are used concurrently, with the high fidelity only applied when the results of low fidelity are considered to be not sufficiently accurate or when a higher degree of confidence in the design is needed. This combination can lead to the generation of the same designs obtained using purely high fidelity models but a much lower computational cost. Examples of these approaches are that of Minisci[32] or Choi[11].

The use of low order modelling is currently extensive, in particular during conceptual design. However the increase of available computational resources is making MFDA more attractive.

In the context of aero-thermal modelling in the the supersonic and hypersonic ranges, low order models are generally based on local inclination methods, as they allow the computation of the surface values on arbitrary geometries in a simple manner, while the high fidelity tool is computational fluid dynamics (CFD).

To improve design process in this context, two strategies are possible. One would be to extend the low order models so the use of high fidelity is not required, while the other would be to improve the efficiency of the use of high fidelity tools.

LIMITATIONS OF LOW ORDER AERO-THERMODYNAMIC MODELS

Regarding low order aero-thermal models, codes like HyFlow[45], the aerodynamic preliminary analysis system (APAS)[7], the configuration-based aerodynamics (CBAero)[25] tool or the supersonic/hypersonic arbitrary body program (S/HABP)[16] are especially relevant for the design of supersonic and hypersonic vehicles.

Their main limitation is that they give no information about the shock structure. While this information is usually not required during preliminary design, there are situations where the inclusion of shock related phenomena might be

required in order to guarantee the survivability of the vehicle in presence of the strong pressure and heat gradients due to shock waves hitting some part of the vehicle surface.

In fact, when a shock impacts the vehicle, heat transfer rates can grow up to 17 times the values measured at stagnation regions and pressure can reach up to 8 times that of free stream[20]. These extremely high temperatures and pressures can have an impact on the survivability of the vehicle if they are not dealt with carefully, either by correctly designing the thermal protection system (TPS) or by modifying the geometry such as the impingement is no longer present.

A classic example of the effects of impingement is that of the X-15 vehicle. This was a manned experimental rocket plane operated by NASA in the 1960s and, as a piece of trivia, it still holds the speed record for a manned powered aircraft. During one of its test flights, a dummy ramjet was attached on the underside of the vehicle to explore the use of the X-15 as a test bed for supersonic air-breathing propulsion. The engine nacelle was attached through an Inconel pylon coated with an ablative TPS. During a flight at Mach 6.7, the dummy ramjet detached from the vehicle. Postflight investigation[44] showed that the shock generated on the nacelle had impacted the pylon, melting trough it and causing structural damage to the plane.

Along with thermal effects, the peak pressure resulting from impingement modifies the pressure field over the vehicle, influencing the aerodynamic forces possibly leading to controllability and manoeuvrability issues[26]. This situation is especially interesting in the presence of shock interference from proximal bodies, as those encountered in stage separation of Two Stage To Orbit (TSTO) vehicles. This complex, time dependant, problem has to be studied to assure that there is no risk of the stages colliding during separation.

These examples highlight the importance of accounting for shock related phenomena as early as possible in the design process.

Along this perspective, the development of a shock structure estimation method compatible with low order models would greatly expand their usability in the context of MDO of high Mach vehicles.

APPLICATION OF CFD IN THE SUPERSONIC/HYPERSONIC REGIME

In the supersonic and hypersonic regimes, the presence of shock waves requires adopting carefully crafted meshes in order to cluster points in the regions of strong gradients and avoid numerical diffusion arising from meshes that are too coarse.

Apart from using heuristic methods, the only consistent approach is the adoption of an adaptive procedure where an original, non optimal, mesh is iteratively improved by the solution computed on previous meshes[4]. This approach is quite time intensive, making meshing of high Mach flows one important factor in the relative slowness of the application of high fidelity tools. The shock structure estimation method mentioned earlier would contribute also to the high-fidelity CFD, by providing the shock structure directly from the geometry and flow conditions, making the use of CFD simulations faster in the context of MFDA.

PREVIOUS ATTEMPTS ON SHOCK ESTIMATION

During the early days of supersonic flight theory there was an abundance of approximate models that provided the shock location, either for its own sake or as a by-product of the computation of pressures over the surface of the vehicle[21]. The interest for these approximations can be explained by the need of predictive capability for the design of supersonic vehicles coupled with very limited computational resources.

These models can be classified into two main categories according to the development path. One group of methods was produced as solutions to the supersonic problem under simplifying assumptions, either simplifying the physics, the geometries that the model could address or both. The other category was that of correlations with experimental data. The spirit of the present method lays between these two groups, somewhat closer to the second category.

For models produced as solutions to the supersonic problem, a good starting point is the Method of Characteristics (MOC). This numerical method is of great importance both historically and as background for approximate methods. The solutions produced are exact within numerical error under the assumptions.

While easy to apply in the two dimensional case, it requires a great number of calculations. It is for this reason that analytical solutions were sought. This was achieved through a perturbative solution of the equations. Decomposing the velocity in the x and y directions as

$$u = u_\infty + u'; \quad v = v'$$

considering

$$u' << u_\infty; \quad v' << u_\infty$$

a set of equations is obtained which is known as small perturbation theory. If, in addition, $u' << v'$, the equations are further simplified and slender body theory is obtained. This model, linear in supersonic flow and non-linear in the hypersonic regime [1], readily produces surface pressures and shock shape for attached shocks on slender bodies and saw important application on the development of supersonic airfoils and fuselages. From these perturbed equations, another simplified method came from the realization that the equations for slender bodies in steady flow also describe unsteady flow in one less spatial dimension. This model is known as *blast wave theory* and, unlike slender body theory, can tackle blunted bodies by modelling the initial conditions of the unsteady flow as an instantaneous energy release.

Another approximate method worth mentioning is that of thin shock layer theory, developed by Maslen. Here the simplification comes from assuming that the shock, the body and the streamlines in between share the same shape. This assumption is closest to reality in the high hypersonic regime, as Mach number tends to infinity.

A detailed description of all these methods can be found in Anderson[2].

Regarding correlation based methods, there was much interest in addressing blunted geometries using experimental data. Examples of this are works like that of Kaattari [23] for blunt bodies and high angle of attack airfoils or Love [30] for detached shock waves. This interest came from the fact that detached shocks were not very amenable to approximate solutions. Even numerical approaches were difficult, due to the mixed character of the flow[2]. From these methods,

1 For small perturbation theory, the slenderness ratio $\delta << 1$. In the supersonic regime $M_\infty \simeq 1$, $M_\infty \delta << 1$ while in the hypersonic regime this is no longer true and $M_\infty \delta \simeq 1$

2 The PDEs are elliptical in space in the subsonic part of the field, behind the near normal part of the bow shock, while hyperbolic in the supersonic part. This was eventually solved by the realisation

Billig's[5] engineering formula for stand-off distance and shock curvature for spheres and cylinders in supersonic flow is arguably the most important, and it is still in use today.

These theories were expanded over time to address more general cases, but for complex shapes or interactions the only tool available still was experimental testing.

With the increase of available computational power and the advent of Computational Fluid Dynamics (CFD), engineers could now have a detailed description of the flow with a relatively easy, albeit time consuming, method. While classic approximate methods of shock estimation are still used, new developments are sparse. An example of one of such developments is that of a shock shape model for axis aligned cylinders[31].

The current available computational power, however, is an opportunity to develop simplified methods for shock estimation that were simply not possible at the beginnings of supersonic flight theory. As stated before, these methods could be a helpful addition to aero-thermodynamic low order models used in the contexts of MDO and MFDA.

THESIS OBJECTIVE

The objective of this thesis is to produce and verify an advanced low order model for shock structure estimation. It should take as inputs the geometry and the flow conditions and address, in one cohesive methodology, attached and detached shocks in two and three dimensions as well as shock structures composed of multiple shocks. While partial models have been developed since the beginning of supersonic flight theory, no unified (in the sense of addressing all these situations with a single model) approach exists yet. This method is to be coupled with a low order aero-thermodynamic model in order to improve its predictions when shock related phenomena are present. The intended field of applicability is preliminary design, especially in the context of MDO. It is also to serve to improve the efficiency of CFD simulation in the context of MFDA.

The design favours fast execution over high precision while maintaining useful predictive capabilities.

---

that the equations are hyperbolic in time for both subsonic and supersonic cases and therefore a pseudo-time marching scheme can be used for mixed steady flows.

THESIS STRUCTURE

In chapter 2, a methodology for shock estimation on arbitrary geometries, compatible with low order aero-thermodynamic models, is described. Chapter 3 contains the verification of the various aspects of the methodology through comparisons with CFD simulations. The method is applied to relevant engineering problems in chapters 4 (improving low order models) and 5 (informing high fidelity tools). Finally, chapter 6 contains some concluding remarks and ideas for future work.

2 METHODOLOGY

In this chapter, the methodology for shock estimation will be presented. The procedure is first developed for conceptual geometries (two dimensional convex geometries with attached shocks in inviscid flow) and then generalized, introducing modifications that allow for the removal of assumptions on the geometry.

Along the chapter, the results of the shock estimation will be compared with CFD simulations. These have been performed with OpenFOAM[34], a continuous mechanics library with precoded solvers, using the rhoCentralFoam [19] solver. This solver has been used in the past for hypersonic flows[40] and it is considered verified. Therefore, in this work, the results from rhoCentralFoam are taken as reference to which compare the estimated shocks against. Unless otherwise stated, the CFD simulations are inviscid, using perfect gas model ($\rho = \frac{1}{RT}P$) and run in normalized gas conditions. This means the use of an idealized perfect gas with a molecular weight that results in a speed of sound of $a = \sqrt{\gamma RT} = 1m/s$ for $T = 1K$ at $P = 1Pa$.

SHOCK ESTIMATION IN TWO DIMENSIONS

The starting point for shock estimation is the equations that describe the flow. The most general set of these equations is that of Navier-Stokes, where the only assumption is that the fluid can be modelled as a continuum[1]. Disregarding viscosity effects, this set of equations can be simplified into the Euler equations that describe inviscid flows. These, in turn, can be further simplified assuming the flow to be irrotational ($\nabla \times V = 0$). While not strictly true in the case of slender bodies with attached shocks, the assumption is reasonable due to the low curvature of the shock. In the irrotational case, the velocity can be written in the form of a potential. Assuming, additionally, the flow to also be isentropic

1 defined as having a Knudsen number $Kn = \frac{\lambda}{L} << 1$, where λ is the mean free path and L is a characteristic length of the flow

and steady, the Euler equations combine into one expression for the potential that in the two dimensional case has the form

$$\left(1-\frac{\phi_x^2}{a^2}\right)\phi_{xx}+\left(1-\frac{\phi_y^2}{a^2}\right)\phi_{yy}-\frac{2\phi_x\phi_y}{a^2}\phi_{xy}=0 \tag{1}$$

where ϕ is the velocity potential ($\nabla\phi = V$) and a the speed of sound. For the following discussion, the flow is considered to be steady, inviscid, irrotational, isentropic and two dimensional.

The method of characteristics (MOC) can now be used to solve for the potential in the fluid domain. The background of the method is the realization that a partial differential equation can be transformed into a family of ordinary differential equations expressed along certain curves referred to as characteristics. These ODEs are known as compatibility equations. Their solutions, from a set initial condition, are a set of curves that form a solution surface of the original PDE. The equations of supersonic flow are hyperbolic, there are therefore two intersecting families of characteristics, which allow for the numerical solution of the equations by marching along the characteristics from a known initial line.

For this flow, the equation for the characteristic curves is

$$\frac{dy}{dx}=\tan(\theta\mp\mu) \tag{2}$$

where θ is the inclination of the streamline through the point and $\mu = \arcsin\left(\frac{1}{M}\right)$ is the Mach angle. As commented before, two solutions exist. The C^+ characteristic, also known as left running, on the left hand side of the streamline. And the C^- characteristic, known as right running, on its right hand side.

The compatibility equations that hold along the streamlines are

$$\theta+\nu(M)=K^- \tag{3}$$

for the right running characteristic and

$$\theta-\nu(M)=K^+ \tag{4}$$

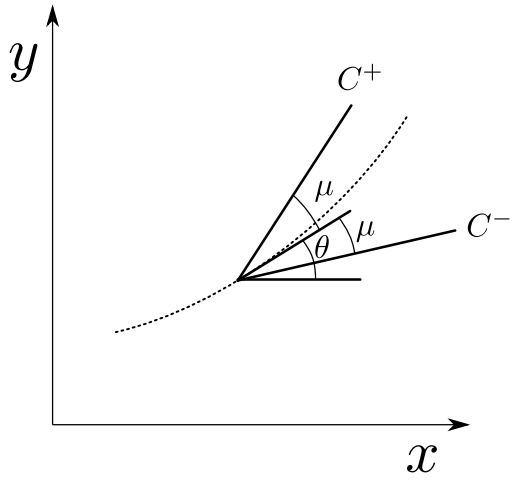


Figure 1: Point over streamline showing the two intersecting characteristics

for the left running. Here K are constants and $\nu(M)$ is the Prandtl-Meyer function

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}} \arctan \sqrt{\frac{\gamma-1}{\gamma+1}(M^2-1)} - \arctan \sqrt{M^2-1} \tag{5}$$

where M is the Mach number and γ is the ratio of specific heats of the gas.

In this simple case the compatibility equations are algebraic instead of differential.

The procedure starts from an initial line where the solution is known. This initial line is selected near the shock attachment point, where the geometry can be considered a wedge. The flow properties on the line are the after-shock conditions of the attached oblique shock on that wedge. From here, the network of characteristics is grown.

The first point of the second ray is computed from the intersection of the first C^- characteristic from the initial line and the body. The body is a streamline, therefore θ_b is known. Knowing that K^- is conserved along the C^- characteristic

$$\nu_b = K^- - \theta_b \tag{6}$$

The rest of the points of this ray are computed along the C^+ characteristic from this first point, intersecting with the C^- characteristics that emanate from

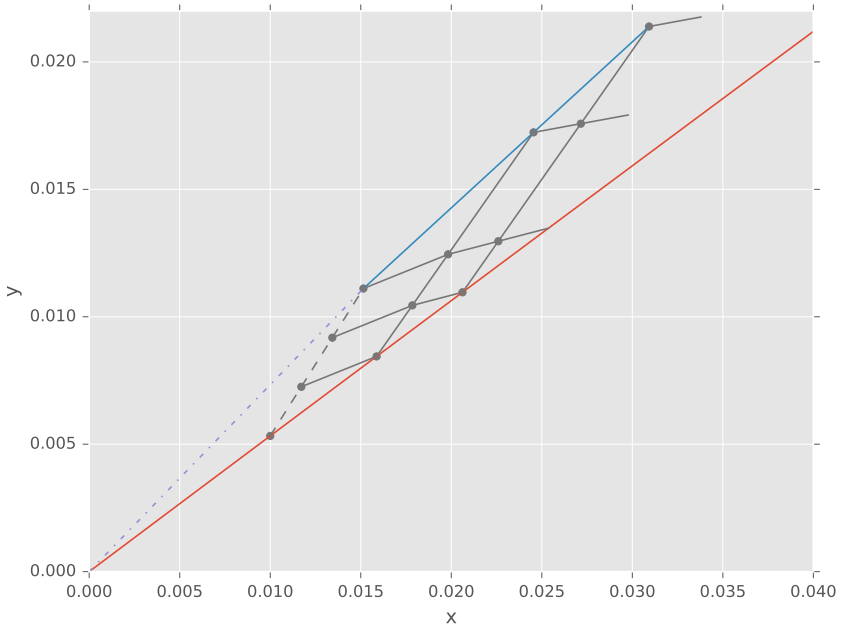


Figure 2: Start of the method of characteristics. The initial line, corresponding oblique shock and the first few characteristics of the network are shown

the initial line. On this intersection, two constants are conserved along the two streamline families. The values at the point can then be computed as

$$\theta_p = \frac{1}{2}(K^- + K^+) \tag{7}$$

$$\nu_p = \frac{1}{2}(K^- - K^+) \tag{8}$$

Finally, the shock point of this ray is computed. Here, K^+ is conserved along the ray (C^+ characteristic). Therefore

$$K^+ = \theta_s - \nu_s \tag{9}$$

The other free variable is the local inclination of the shock, β, which, given the free stream Mach number, gives the after shock Mach number and velocity angle that should correspond to the $\theta_s - \nu_s$ set earlier. This problem is solved iteratively. The procedure continues with the next ray, repeating the previous calculations. A detailed derivation of the method can be found in Anderson[3].

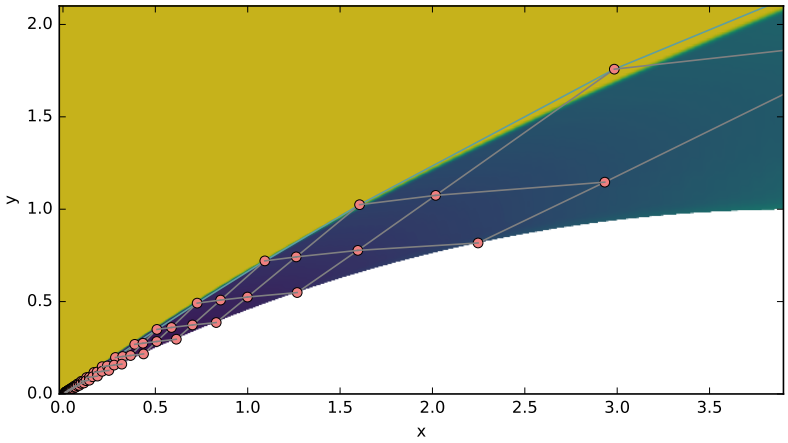


Figure 3: Computation of shock position using MOC over CFD Mach field

Figure 3 shows the computed shock position for a two dimensional tangent ogive at Mach 9. The results match those of the CFD simulation where the density of the network of characteristics is high enough. This relatively low resolution has been selected to be able to clearly show the characteristics network, and increasing the number of points per ray would reduce the error.

The method of characteristics is very precise, but it suffers from certain drawbacks. While it can be applied in three dimensions, its implementation is very convoluted and other numerical approaches (i.e. finite volumes) are preferred. It also requires a starting line where the solution is known.

If computational efficiency is more important than accuracy, the MOC can be simplified by ignoring the effect of the C^- family of characteristics. This simplification is most reasonable in the hypersonic regime, where the reflected C^- characteristics from the shock impact the geometry far downstream, with its validity reducing with decaying Mach number. An approximate method called shock expansion is so obtained. In this theory, the geometry is discretized into segments. The conditions at the first segment are computed as the after-shock conditions of the attached oblique shock. The conditions on subsequent segments are computed through a Prandtl-Meyer expansion fan between them. This method is usually applied to the computation of surface pressures, where the pressure value is constant inside each segment of the geometry. Shock expansion theory can be adapted to compute the shock position over a geometry. If the shock resulting from the interaction between oblique shock and Prandtl-Meyer expan-

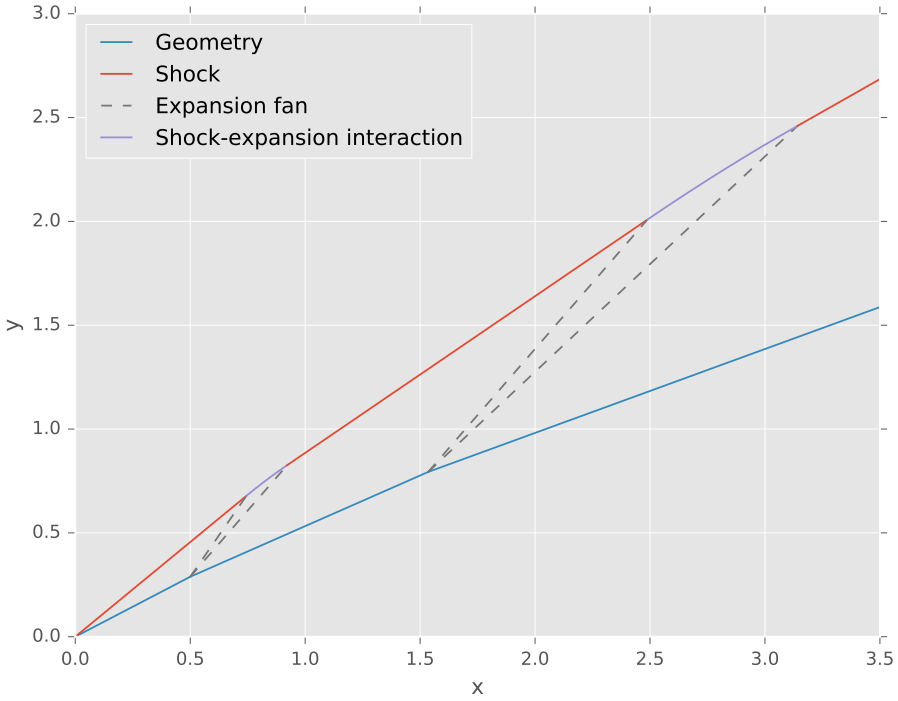


Figure 4: Shock estimation through shock expansion

sion fan is approximated as a second order polynomial[28], the general shock can be discretized as straight and parabolic sections, starting from the attached oblique shock of the the first segment and continuing along the geometry.

This is a simplistic view of an oblique shock overtaking a Prandtl-Meyer expansion fan. The flow is rotational and non simple after the interaction zone. This results in different pressure across the shock than through the PM expansion. While it would be wrong to use this model to describe the flow, the shock position is reasonably well captured.

If the sought result is the shock position and not necessarily the surface pressures, modelling the oblique shock - expansion fan interaction as a parabolic section might be unnecessary, and an instantaneous turn of the shock on the first Mach wave of the expansion fan might be preferred. Under this assumption, shock expansion theory can be further simplified applying ideas from another surface pressure method, namely tangent wedge, to compute local shock inclinations.

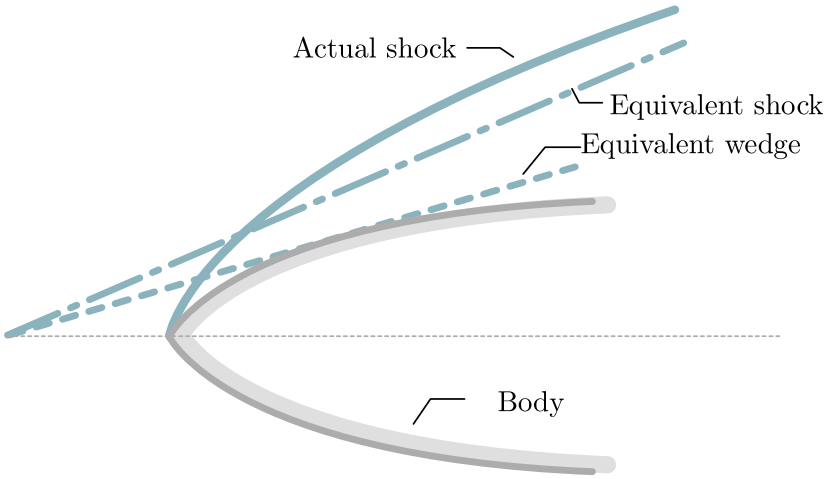


Figure 5: Tangent wedge, as illustrated by Anderson

The tangent wedge method refers to a way of estimating surface pressures on a body in hypersonic flight. The method equates the pressure at any point of the surface of the vehicle with the one on the surface of a wedge with the same inclination with respect to the flow as the local inclination at the point and at the same free stream conditions. The pressure for the equivalent wedge is then computed through the use of the oblique shock relations. In contrast with shock expansion theory, tangent wedge method has very weak theoretical foundations. It is used because it produces reasonable results with very little effort in a wide range of flow conditions, but it lacks a strong physical justification.

When estimating pressures, the tangent wedge method assumes a local shock inclination. The actual position of the shock is however unknown. Considering the local shock as a straight line with known inclination, determining a point in it is all that is needed to uniquely identify it. In Anderson [2], the explanation of tangent method is accompanied by a figure that shows a symmetric ogive. In this case, the selected point is the apex of the equivalent wedge when intersecting the axis. This produces an approximation of the local shock.

The global shock would be the envelope[2] of these local shock lines. This approach only requires local information, all points are independent and is therefore embarrassingly parallel. While this explanation has great didactical value, it is of little practical use. This point selection oversimplifies the flow and ignores the expansion over the body, producing results quite far from the correct shock

2 Curve tangent to all curves of a family.

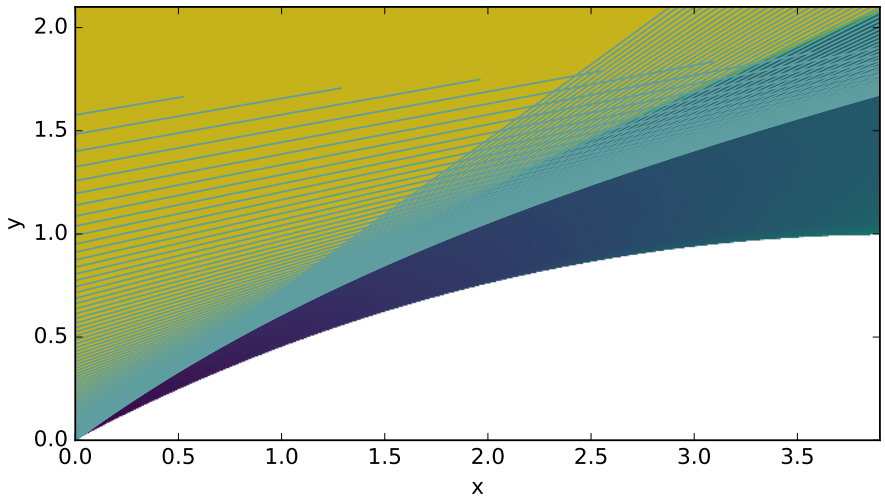


Figure 6: Shock estimation for an ogive through tangent wedge over CFD Mach field. The family of local shocks is shown

structure. Figure 6 shows the estimated local shocks for a bi-dimensional ogive over a Mach number field obtained with CFD.

The envelope of those shocks falls closer to the surface than the shock computed through CFD, highlighting the limitations of this approach.

The accuracy of the method can be improved by taking into account the expansion over the body combining, in a way, tangent wedge with the shock expansion theory showed earlier. For attached two-dimensional shocks, and considering the geometry to be a polyline, the procedure is as follows.

- The shock for the first segment of the geometry is computed as an attached wedge shock. Its inclination given by solving for shock inclination (β) in equation 10 through a a numerical root finding algorithm.

$$\tan\theta = 2\cot\beta\frac{M_1^2\sin^2\beta - 1}{M_1^2(\gamma + \cos 2\beta) + 2} \tag{10}$$

  θ is the inclination of the wedge, M_1 the Mach number before the shock and γ the ratio of specific heats.

- The next shock section has the inclination given by the equivalent wedge of the second segment of the polyline. Its origin is determined by the intersection of the first shock segment with the Mach wave produced at

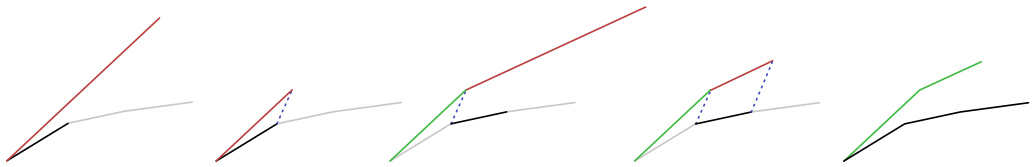


Figure 7: Procedure for flow expansion correction

the end of the first segment of the geometry. The inclination of this Mach wave is given by $\mu = \arcsin(1/M)$, where the M is the after shock Mach number. [3]

- This continues for all sections of the geometry polyline.

The procedure is illustrated in figure 7. This turns the method from local information to advancing front. While this makes the procedure much more precise, it also makes it slower. Another disadvantage is that the errors are now cumulative, as the position of each local shock is influenced by the position of the previous one.

Compared with figure 6, figure 8 matches the CFD results much closer. The cumulative error in the streamwise direction can also be seen in the way the estimated shock slowly drifts away from the CFD solution.

The resulting method considers the expansion between segments as infinitesimal Mach waves, which is reasonable if the polyline is a discretization of a C^1 underlying geometry, and takes the local shock inclination and Mach number from the tangent wedge method. The advantages of this hybrid method against traditional shock expansion will be made apparent in the next section.

Modification for detached shocks

For detached shocks, the starting procedure employed on the attached shock case is not applicable. There is no attached equivalent wedge in the first segment of the geometry. A solution for the detached zone must be imposed. Assuming the geometry to be a circle in the vicinity of the stagnation point, Billig [5] provides

3 Computed through the oblique shock relations:
$M_2 = \frac{1}{\sin(\beta-\theta)}\sqrt{\frac{1+\frac{\gamma-1}{2}M_1^2\sin^2\beta}{\gamma M_1^2\sin^2\beta-\frac{\gamma-1}{2}}}$

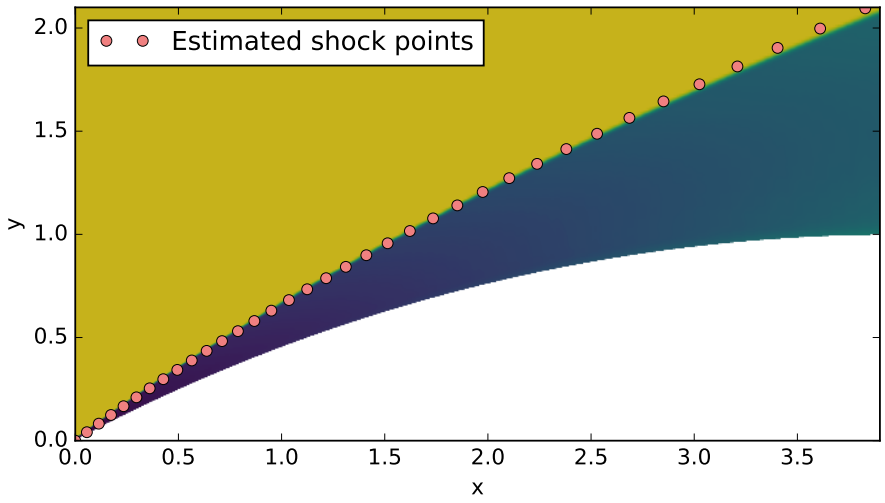


Figure 8: Shock estimation for an ogive with flow expansion correction over CFD Mach filed

such a solution. The detached shock around the stagnation point is described as a hyperbola

$$x = R + \Delta - R_c \cot^2\theta \left[\left(1 + \frac{y^2 \tan^2\theta}{R_c^2}\right)^{1/2}\right] \tag{11}$$

being R the radius of curvature of the geometry at the stagnation point, R_c the radius of curvature of the shock at the vertex, Δ the stand off distance and θ the asymptotic angle of the hyperbola. This is the free stream Mach angle in the case of a cylinder and the attached shock angle in the case of a cylinder with a wedge afterbody. The axes have their origin at the centre of curvature of the geometry, with the x axis aligned with the flow. The vertex radius of curvature and stand off distance are given by

$$R_c/R = 1.386 \ \exp[1.8/(M-1)^{0.75}] \tag{12}$$

$$\Delta/R = 0.386 \ \exp(4.67/M^2) \tag{13}$$

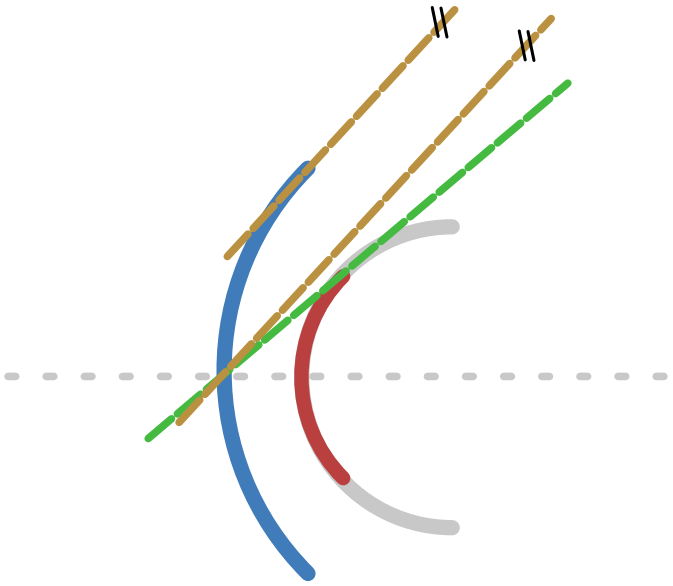


Figure 9: Starting procedure for detached shocks, showing: Geometry (grey), geometry section with subsonic after shock conditions for its local tangent wedge (red), Billig's solution for that section (blue), first local tangent wedge with supersonic after shock conditions (green), and corresponding local shock tangent to Billig's solution (yellow)

where M is the Mach number of the free stream. The shock estimation is patched to this hyperbola at the inclination where the first equivalent wedge from the geometry produces an attached weak shock (post shock Mach number is supersonic). The estimation process then continues as in the attached case.

While this procedure provides a suitable start for the shock estimation method, the results are far worse than the ones obtained in the case of attached shocks. This can be seen in figure 10. It is not surprising, though, as the assumptions made earlier do not hold any more. The flow after the bow shock is rotational and its evolution after is non isentropic.

A careful look at the estimation of the shock for a cylinder suggest a way to improve the results. The proposed formulation considers the way subsequent points of the shock are estimated by looking at the intersection of local shocks with Mach waves. For the first point, this Mach wave should lie on the sonic line for the procedure to be correct. In the case of the cylinder, Billig's hyperbola provides an approximation for the whole shock, and therefore the position of the sonic line. In general, the first Mach wave and the sonic line position estimated

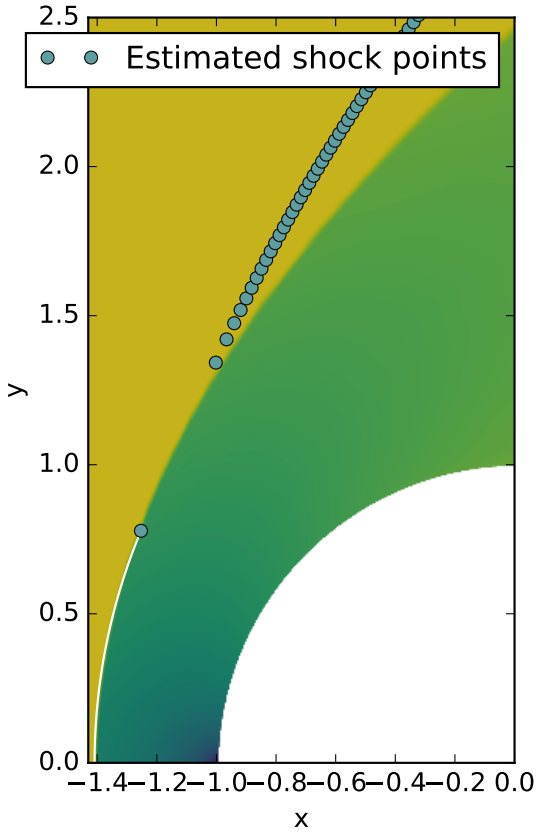


Figure 10: Shock estimation without detachment correction

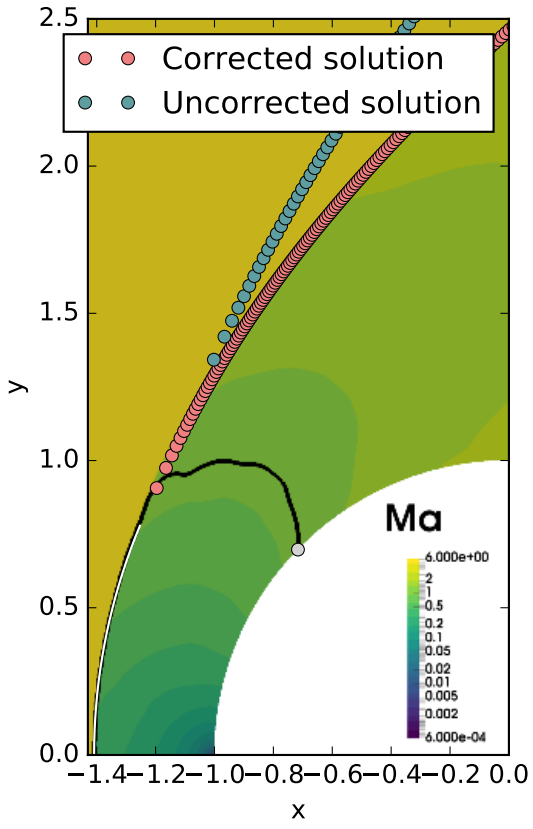


Figure 11: Detachment correction procedure

by Billig do not match. The procedure can be improved by multiplying the Mach angle of all Mach waves by a correction factor that matches these two first lines.

Figure 11 illustrates the correction. The CFD sonic line appears in black. The first point of the cylinder geometry with a local tangent wedge that produces a shock with supersonic after shock conditions is shown in grey. From this point, the Mach wave intersecting the first local shock tangent to Billig's solution produces the first point of the uncorrected estimated shock. This Mach wave is far from the estimated sonic line (marked by the end of Billig's solution and the grey cylinder point). Matching this Mach wave with the estimated sonic line gives the correction factor, which is then used for all the other Mach wave computations.

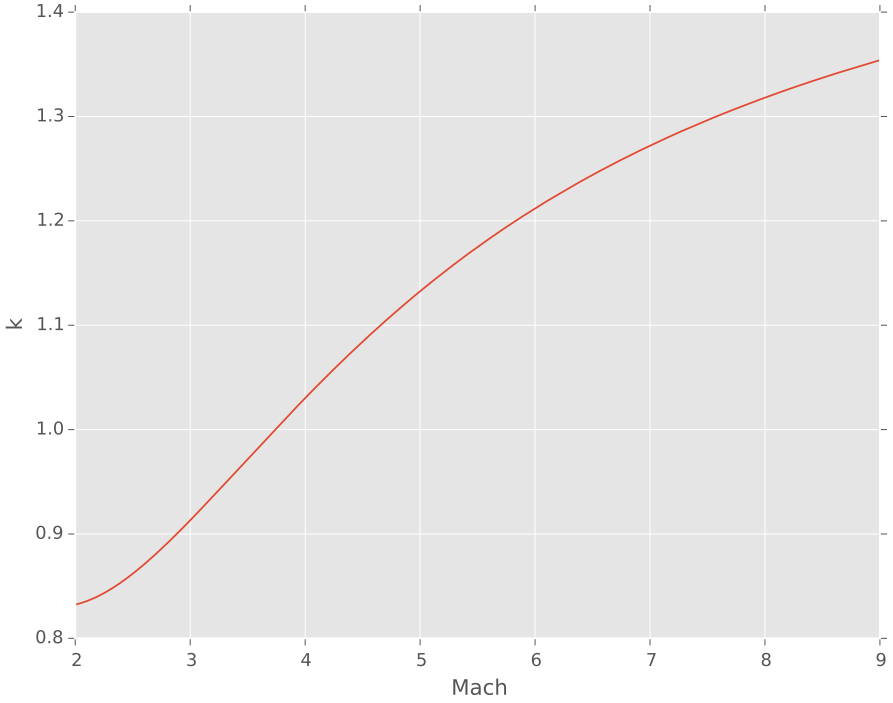


Figure 12: Correction factor for detached shocks

The correction factor is a function of Mach number and is shown in figure 12. Applying it to all Mach wave computations greatly improves the results for shock estimation on detached shocks.

In the detached case, the usefulness of the proposed formulation is observed. While shock expansion is not applicable in this situation, tangent wedge still produces useful results. Through the hybridization process and the novel treatment of detached shocks, the method produces a relatively accurate estimate of shock position, again at the expense of little theoretical foothold.

Modification for stream wise body concavities

In previous sections, only convex geometries have been considered. This makes the shock estimation problem simpler, as only one shock is produced by the geometry. When concavities are present, additional shocks might be produced.

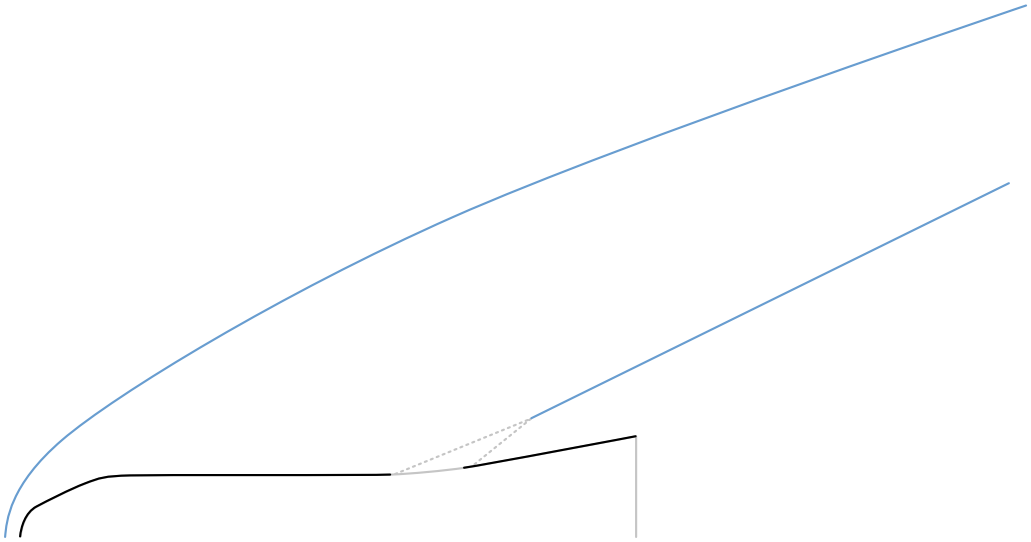


Figure 13: Convex sections and estimated shock

The shocks can now interact with each other or impinge on the geometry. Some modifications have to be introduced to deal with these situations.

First, the geometry is separated into convex sections and concavity zones. This is done by computing the angle between any two connected segments of the geometry. If that angle increases stream wise, a concavity is detected. This concave zone persists until the angle between segments stops increasing. Then, a new convex section starts.

From the point of view of the methodology, each convex section will produce a shock, with each concave zone in between producing a compression fan.

The estimation of the shock structure is as follows. The shock produced by the first convex section is estimated as previously described. The first shock point for the next convex section is placed at the apex of a compression fan defined in the intervening concave zone. The free stream conditions for these computations are taken as the area weighted mean of the conditions after the first shock, discarding subsonic values. Specifically, the after shock conditions for each shock section are computed using the oblique shock relations. A single value of for the whole shock is then obtained by averaging the results for all shock sections (that produce supersonic values), taking into account their length. While this assumption lacks a strong theoretical justification, it does produce reasonable results. The shock estimation then continues in the usual manner on the next convex section. Therefore, the compression fan is only used to locate

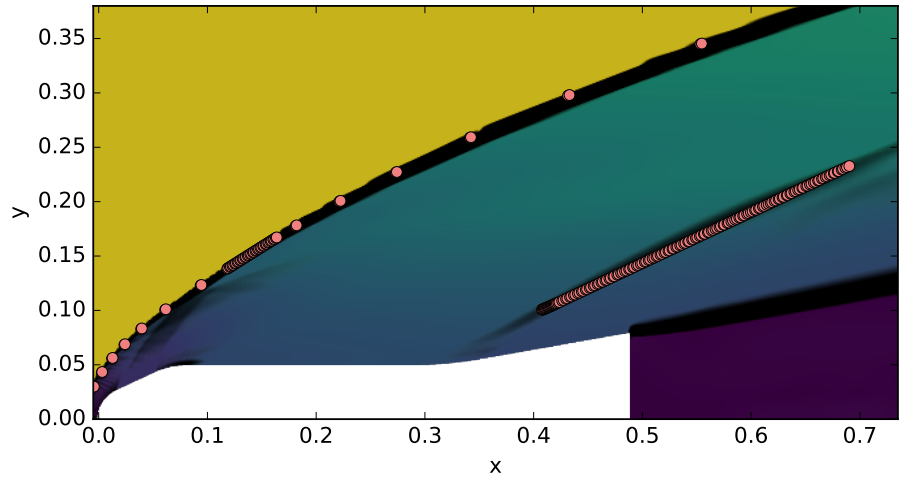


Figure 14: 2D section of HB-2 body. Shock estimation over CFD Mach field with Mach gradient overlay

the start of the second shock, while its structure is given by the preceding convex section.

Figure 14 shows the estimation of the shock structure for the HB-2 geometry over CFD. The HB-2[18] is a calibration model defined by the Advisory Group for Aeronautical Research and Development (AGARD) that has been used extensively to study hypersonic flows. The computation captures both the first and the weaker second shock.

Viscosity effects

Up until now, the method has not taken into account the effect of viscosity on the shock position. This effect is most apparent when an inviscid flow would not produce a shock while one would be obtained in the viscous case, as for example in a supersonic flat plate at zero angle of incidence. In the inviscid case, the flow would be unperturbed, while in the viscous case a boundary layer would grow over the plate that would interact with the free stream creating an oblique shock.

The previously presented method can be applied to this case, maintaining the inviscid assumption for the rest of the flow, by modifying the geometry to account for the boundary layer displacement thickness. This thickness can be

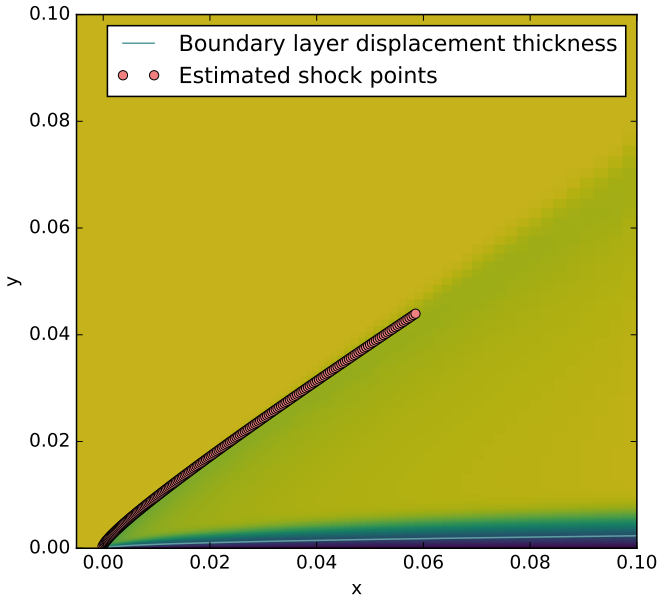


Figure 15: Shock estimation for a laminar viscous flat plate at Mach 2. The computed displacement thickness of the boundary layer is also shown

estimated from Blasius[6] solution for laminar boundary layers of incompressible flows and results in

$$\delta^* = \frac{1.72x}{\sqrt{Re_x}} \tag{14}$$

where

$$Re_x = \frac{\rho u_0 x}{\mu} \tag{15}$$

being ρ the density of the fluid, u_0 the free stream velocity, x the coordinate along the plate in the stream wise direction, and μ the viscosity of the fluid.

While the use of a result for incompressible flows in supersonic conditions is suspect, this has been done as Blasius solution is much simpler than its compressible counterpart and produces results that are accurate enough in the context of this low order shock estimation methodology.

The result of the shock estimation is shown in figure 15 for a laminar viscous flat plate at Mach 2. The estimation accurately captures the shock position.

SHOCK ESTIMATION IN THREE DIMENSIONS

The shock estimation procedure, as explained before, can be applied to three dimensional geometries by carrying out the computation over surface streamlines on the body and then combining these 2D shock estimations into a 3D envelope. While this explanation is straight forward, the application of the method to three dimensional geometries requires some adaptations and introduces certain complications. In an implementation note, while in 2D the language chosen for the code was Python3, C++11[4] has been selected for the 3D cases.

The geometrical manipulations that were trivial in two dimensions are complex and cumbersome in 3D. For this reason, the implementation of the shock estimation in three dimensions is supported by CGAL[42], a computational geometry library that provides algorithms for common geometrical manipulation tasks. The use of CGAL is the main driver behind the selection of C++11 as implementation language, being speed another important factor.

In the next subsections, the adaptations required for three dimensional cases will be explored.

Local inclination method

While in 2D the local shocks were computed using tangent wedge method, in 3D tangent cone is more suitable. This local inclination method is conceptually similar to tangent wedge, substituting the flow field used on the estimation from supersonic wedge to supersonic cone. The solution of this flow field is given by the Taylor-Maccoll equation.

$$\frac{\gamma-1}{2}\left[1-V_r'^2-\left(\frac{dV_r'}{d\theta}\right)^2\right]\left[2V_r'+\frac{dV_r'}{d\theta}\cot\theta+\frac{d^2V_r'}{d\theta^2}\right] - \frac{dV_r'}{d\theta}\left[V_r'\frac{dV_r'}{d\theta}+\frac{dV_r'}{d\theta}\frac{d^2V_r'}{d\theta^2}\right]=0 \quad (16)$$

4 The 2011 standard of the C++ programming language

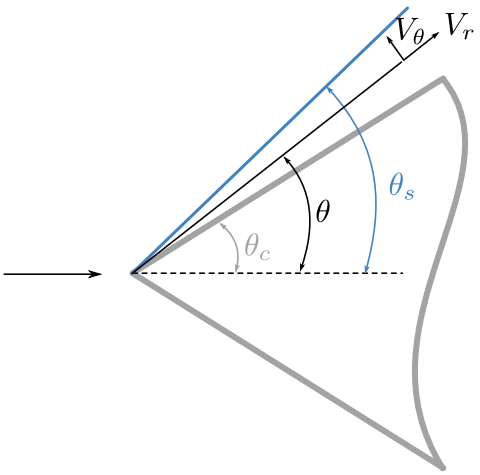


Figure 16: Conical flow

with

$$V' = \left[\frac{2}{(\gamma-1)M^2} + 1\right]^{-1/2} \tag{17}$$

The equation is expressed in a spherical reference frame, with r and θ as radial and angular coordinates (with constant values on ψ), as shown in figure 16. Additionally, γ is the ratio of specific heats, and V'_r is the radial component of V'.

To solve this equation, an inverse procedure similar to the one by Anderson[3] was followed. Knowing the semi-angle of the shock, the integration of equation 16 would provide the semi-angle of the cone that supports that shock. Calling this integration $f(x)$, the shock semi-angle is obtained from the cone one by solving $f(x) - \theta_c = 0$.

The integration and root finding are implemented using the Boost libraries [5].

5 Boost is a collection of peer-reviewed C++ source libraries that provides functionality not present in the C++ Standard Library (https://www.boost.org/).

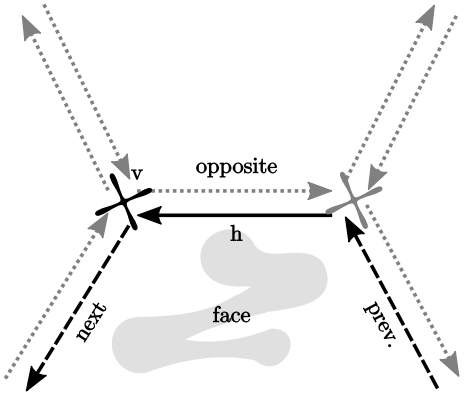


Figure 17: Half-edge data structure

Geometry representation

The geometry in the 2D cases was described by a polyline. In 3D, the geometry is discretized as a triangular surface mesh. The actual implementation of this mesh in code is carried out using a half-edge data structure.

In a half-edge data structure, each edge of the mesh is represented through two oriented half-edges. Figure 17 shows a minimal diagram of this structure. The structure stores each half-edge (h, in the figure) with references to its opposite, previous and next half-edges around the incident face. Each vertex, as well as each face, stores a reference to one of its incident half-edges.

The structure is general, in that it can represent a polyhedral mesh of any kind (being triangular meshes only one of them). While this form of storage is more complex than the common face-vertex approach, it makes for more efficient manipulation and traversal. For example, finding the face that is connected to another face through an edge is trivial in the case of the half-edge data structure (h $\rightarrow$ opposite $\rightarrow$ face), while it is convoluted in the face-vertex approach (knowing the indices of the vertices of the edge, it requires traversal of the face list to find the face that also contains those two indices).

2D sections

In three dimensions, the procedure presented for 2D is applied over surface streamlines. Streamlines are curves that are tangent to the velocity vector of

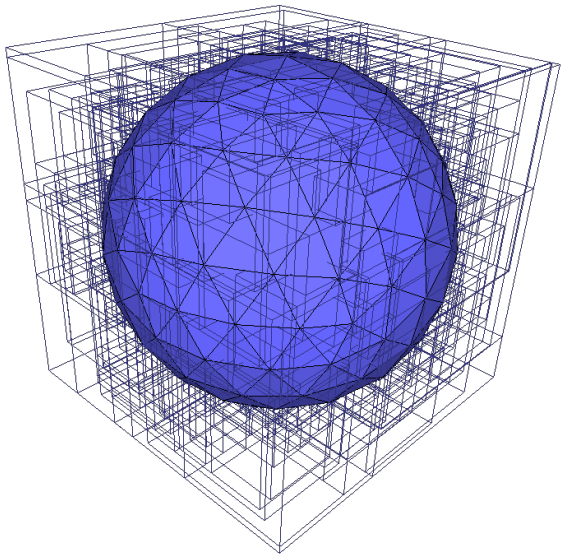


Figure 18: AABB tree of an icosphere

the flow. When the flow is steady (the derivative of the velocity field with respect with time is zero), streamlines and pathlines (trajectories followed by fluid particles) are the same and can be computed integrating equation 18.

$$\frac{d\mathbf{x}}{dt} = \mathbf{u}(\mathbf{x}) \tag{18}$$

being $\mathbf{u}$ the velocity vector of the flow and $\mathbf{x}$ the position of the fluid particle.

Two computational geometry algorithms are used to simplify the streamline computation, namely Axis Aligned Bounding Box (AABB) trees and K-D trees.

An AABB tree is a hierarchy of axis-aligned bounding boxes where the last elements (leafs) of the hierarchy are bounding boxes around the triangles of the geometry. The procedure for creating the tree starts by creating AABBs around all the triangles on the geometry and sorting them along the longest axis, these boxes are then grouped into two equal size sets and new AABBs are created around those sets. Those sets are then again grouped into (now four) equal size set, creating new AABBs around them. This process is applied recursively until the leaf level is reached (top-down method).

For distance queries, a sphere is created around the query point. The sphere is then tested for intersections against the tree. Only if the parent box is intersected are then the next level boxes tested. When a leaf is intersected by the sphere, the distance between the closest point on the triangle and the query point is

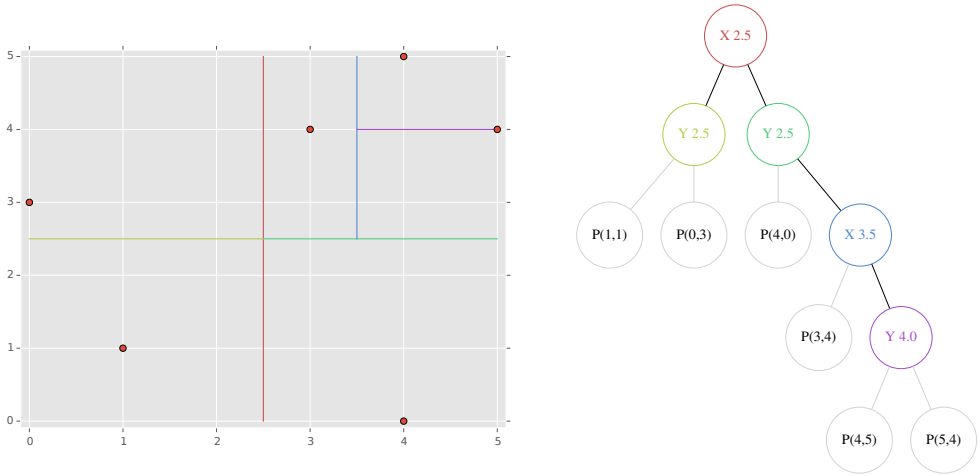


Figure 19: Points in 2D and associated K-D tree

computed and the sphere radius is shrunk to this distance. The initial radius of the sphere is selected so as to be small while still intersecting the geometry. This procedure greatly reduces the number of distance queries between the query point and the triangles of the geometry. The procedure is implemented using the 3D Fast Intersection and Distance Computation[1] package of CGAL.

A K-D tree is a data structure to store k dimensional points. Each node of the tree generates an hyperplane that divides space in two, through one of the k-dimensions. The left branch generated at that node will correspond to one space side according to one of the k-dimensions, with the other side covered by the right branch.

The structure is useful to perform efficient computation of nearest neighbours in point clouds, testing by bisection against the tree. The structure has been implemented in code using the nanoflann library [6]

The procedure followed here to compute surface streamlines is similar to that of the original implementation of HyFlow[45]. An outline of the surface streamline computation is as follows. The free stream velocity vector is projected over the geometry, this creates an interpolant of surface velocities. From the centroid of some facets, equation 18 is integrated backwards, until an stagnation point is reached. The algorithm for selecting streamline origins, the method for surface

6 nanoflann is a C++ header-only library for building KD-Trees (https://github.com/jlblancoc/nanoflann).

interpolation of the velocity and the algorithm used to maintain the streamline integration on the surface, differ from HyFlow's original implementation.

For creating the surface interpolant, the free stream velocity vector is projected over each face as

$$\mathbf{u}_s = \mathbf{u}_\infty - (\mathbf{u}_\infty \cdot \mathbf{n}) \cdot \mathbf{n} \tag{19}$$

where $\mathbf{u}_s$ is the surface velocity vector, $\mathbf{u}_\infty$ is the free stream velocity vector, and $\mathbf{n}$ is the normal to the face. From here, the velocity vector associated to each vertex of the geometry is computed as an area averaged mean of the values of the connected faces. The value of the velocity at any point of the surface can now be calculated as the barycentric mean[7] of the values of the vertices of the face to which the point belongs.

The goal of the computation is to uniformly cover the surface with streamlines with the least amount of calculations. The initial point of the inverted streamlines are the centroids of the facets and its selection plays a big role in this reduction of computational cost. The facet centroids are added to a list. The list is ordered in such a way that elements closer to the beginning of the list are farther with respect to direction of the free stream. That way, the inverted streamlines produced by the first elements will hopefully be the longest ones present over the geometry. The first element of the list is taken as a source. The streamline is integrated, with a time step equal to the quotient between the mean minimum triangle edge length and the maximum surface velocity. At each time step, the computed streamline point is projected back to the surface. This projection requires the computation of the closest triangle and the closest surface point to the computed streamline point. In a naive approach, this would require testing all triangles of the geometry against the point. This approach is infeasible due to the huge computational cost it would entail. The projection is simplified by the use of an AABB tree. When the distance between two consecutive computed streamline points fall bellow a certain threshold, a stagnation point has been reached and the integration stops.

Once the streamline is calculated, the centroid list is filtered against the computed streamline. As shown in figure 20, the midpoints between streamline points are calculated and spheres are defined with centres on those midpoints and in contact with each other. If a centroid on the list lies inside one of those

7 Linear interpolation on the face using barycentric coordinates

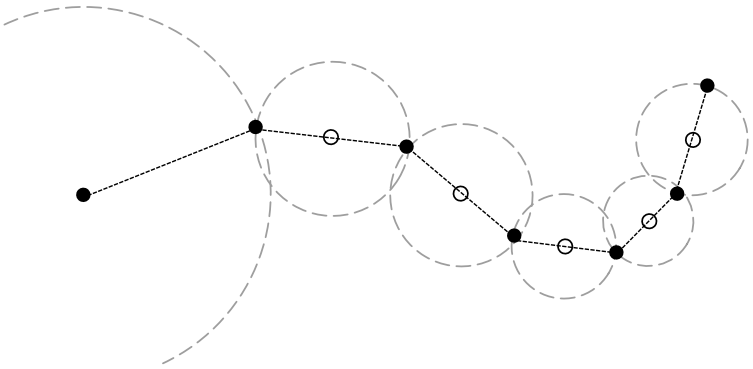


Figure 20: Source filtering

spheres, it is deleted as a streamline source as it is too close to the current computed streamline. For the first point of the streamline, the sphere is centred on the point instead of on the first segment midpoint. This is done as to filter out the stagnation region together. Again, this nearest neighbours calculation would be highly computationally expensive if preformed in a naive way. The calculation is supported by a K-D tree.

After this filtering, the next centroid from the list is chosen as a source and the integration starts again. When the centroid list is empty, the streamline computation ends.

Shock envelope

Once the shocks are computed over the streamlines, the result is a point cloud without connectivity information. To turn this cloud into a triangular mesh, and advancing front meshing procedure is used. This is implemented in code using the Advancing Front Surface Reconstruction[12] module of CGAL.

This meshing procedure is a surface based Delaunay sequential algorithm. A 3D Delaunay mesh of the input surface cloud is created. From that mesh, some triangles are selected to form the output surface. The selection starts from a seed triangle and continues with the best connected triangle. Best here means the smallest connected triangle with internal angles smaller than a threshold, with a difference between its normal and the normal of the connected triangle smaller than another threshold, and that keeps the surface manifold.

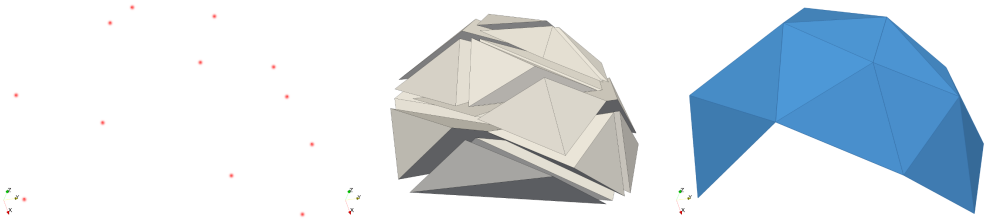


Figure 21: Advancing Front meshing. From left to right: initial point cloud, 3D Delaunay mesh (shrank) and final surface mesh from selected faces

Detached shocks

In the two dimensional case, the geometry around the stagnation point was approximated by a circle. The detached shock was then described using Billig's engineering formula for cylinders. In three dimensions, the situation is more complicated. A general stagnation point can be locally described by two principal curvatures, k_1 and k_2, along the principal directions. Billig's engineering formula provides stand off distance and shock radius of curvature for spheres and cylinders in supersonic flow. Between a sphere and a cylinder with the same radius (R), there is a continuum of tori with constant maximum curvature $(k_1 = 1/R)$ and descending minimum curvature $(k_2 = [0, 1/R])$. Billig's formula can be generalized to describe those tori and therefore describe the local conditions around general stagnation points in supersonic flow.

For stand off distance and minimum radius of curvature, this generalization is computed by blending Billig's solutions for sphere and cylinder. The blending function should take the value for spheres when the ratio $\lambda = k_1/k_2 = \rho_2/\rho_1 = 1$ and the value for cylinders when $\lambda = k_1/k_2 = \rho_2/\rho_1 \to \infty$, where ρ are the radii of curvature $(k = 1/\rho)$. Inspired by Billig's formulas, the following functional form has been formulated

$$f_{tori} = f_{cylinder} \cdot \exp\left(\frac{\ln\left(\frac{f_{sphere}}{f_{cylinder}}\right)}{\lambda^n}\right) \tag{20}$$

where f is either the stand off distance or the radius of curvature of the shock at the vertex in the direction of the maximum curvature. The constant n is fitted to the results of CFD simulations of tori of different ratios of curvatures at Mach number nine. Its value has been computed as $n = 0.57987417201950564$

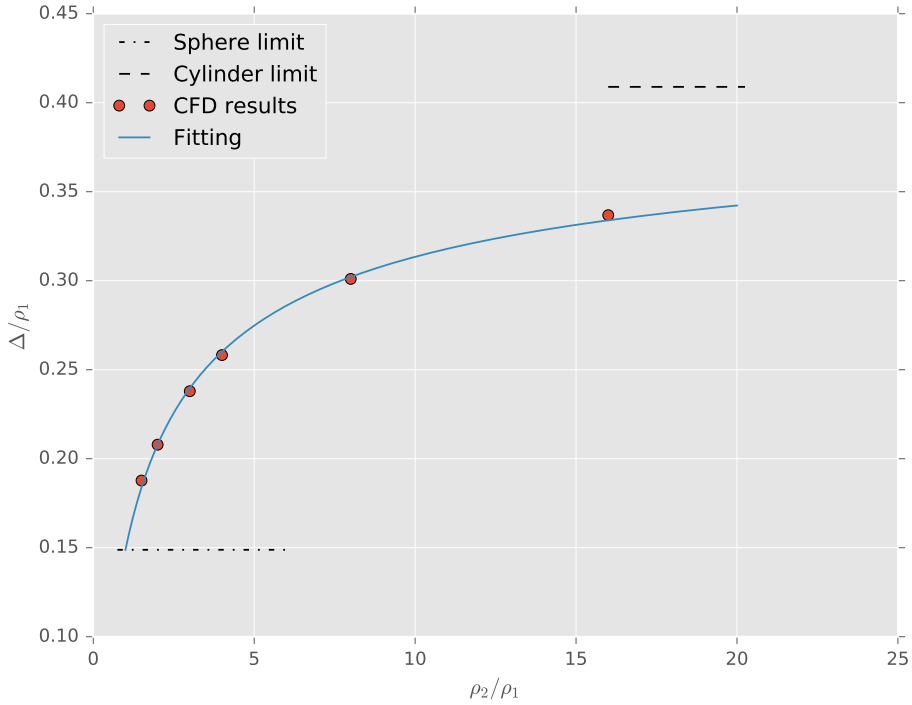


Figure 22: Shock stand off distance as a function of ratio of principal radii of curvature of the geometry $(M = 9)$

Figure 22 shows the shock stand off distance divided by the minimum radius of curvature of the geometry at the stagnation point as a function of the ratio of principal radii of curvature of the geometry at the stagnation point. The dots show the results of the CFD simulations, the dashed lines show the limits for sphere and cylinder while the solid line shows the fitted blending function.

Similarly, figure 23 shows the shock minimum radius of curvature at the apex divided by the minimum radius of curvature of the geometry at the stagnation point as a function of the ratio of principal radii of curvature of the geometry at the stagnation point.

The fitting seems to correctly capture the behaviour with variable ratio of radii.

Billig's solution assumes a hyperbola as the shock shape. For extracting stand off distance and shock radius of curvature from the CFD simulations, the equation of the hyperbola has been fitted to the shock position on the CFD simulations for both principal directions. This generates Δ and R_c values which are consistent with the assumptions of Billig's formula.

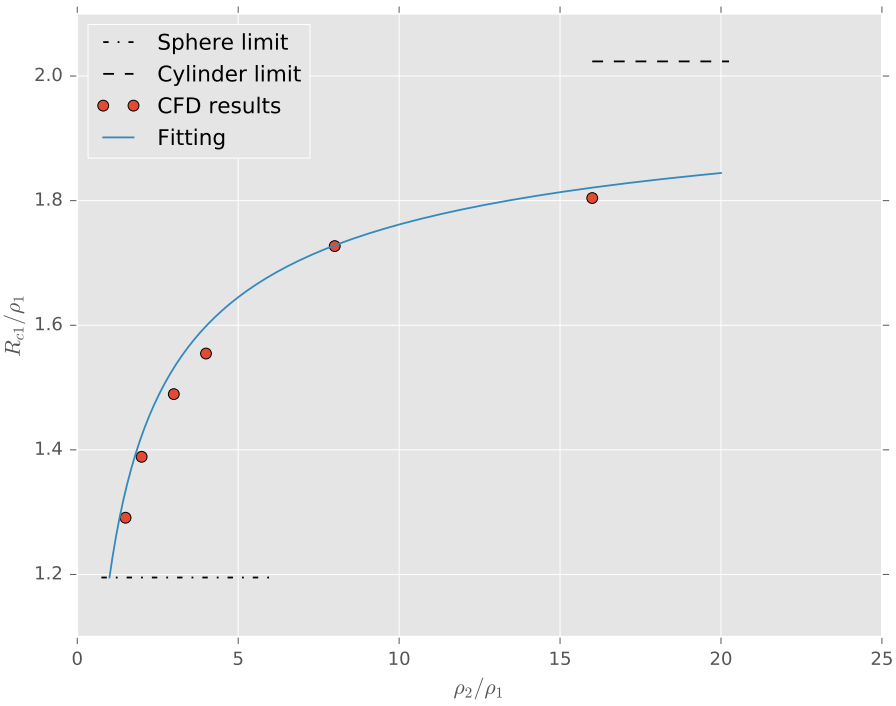


Figure 23: Minimum radius of curvature of the shock as a function of ratio of principal radii of curvature of the geometry $(M = 9)$

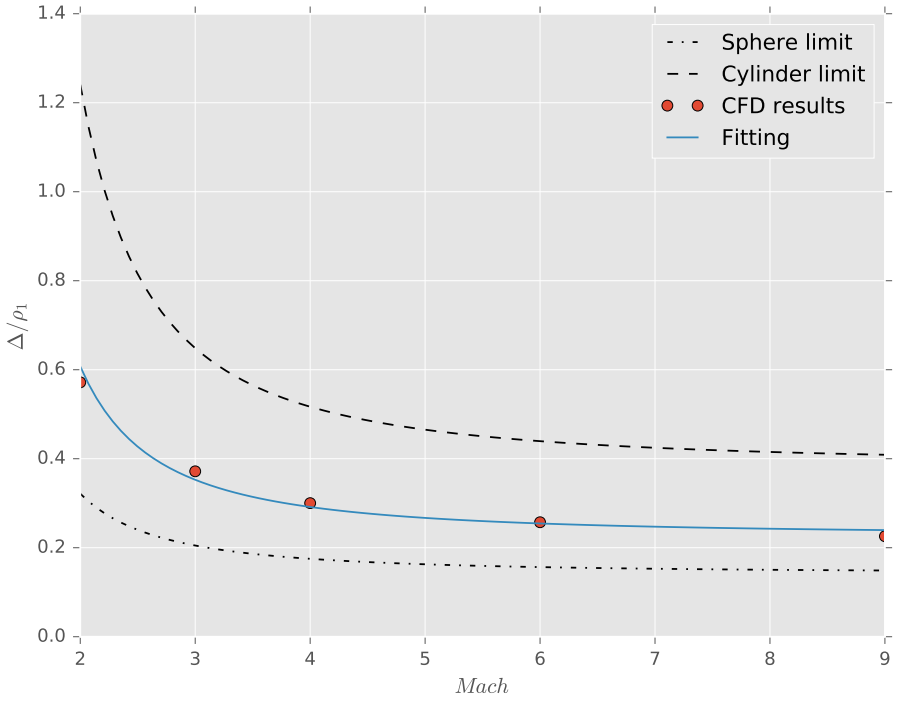


Figure 24: Shock stand off distance as a function of Mach number ($\lambda = 3$)

The functional dependency of the stand off distance with the Mach number is already captured by Billig's formulas for cylinders and spheres. Therefore, the blending function presented in equation 20 does not have to contain it explicitly. Figure 24 shows this functional dependency for $\lambda = 3$. The fact that the formula shows good agreement with CFD results suggests that the fitting parameter was correctly selected.

For minimum radius of curvature, figure 25 shows the behaviour with Mach number. The fitting is slightly worse than in the case of the stand off distance. This might be due to the difficulty of computing the radius of curvature of the shock from the CFD.

Finally, figure 26 shows the maximum radius of curvature of the shock as a function of ratio of principal radii of curvature of the geometry. For this variable, a linear fitting has been selected in the form

$$R_{c2} = n_2\lambda + (R_s - n_2) \tag{21}$$

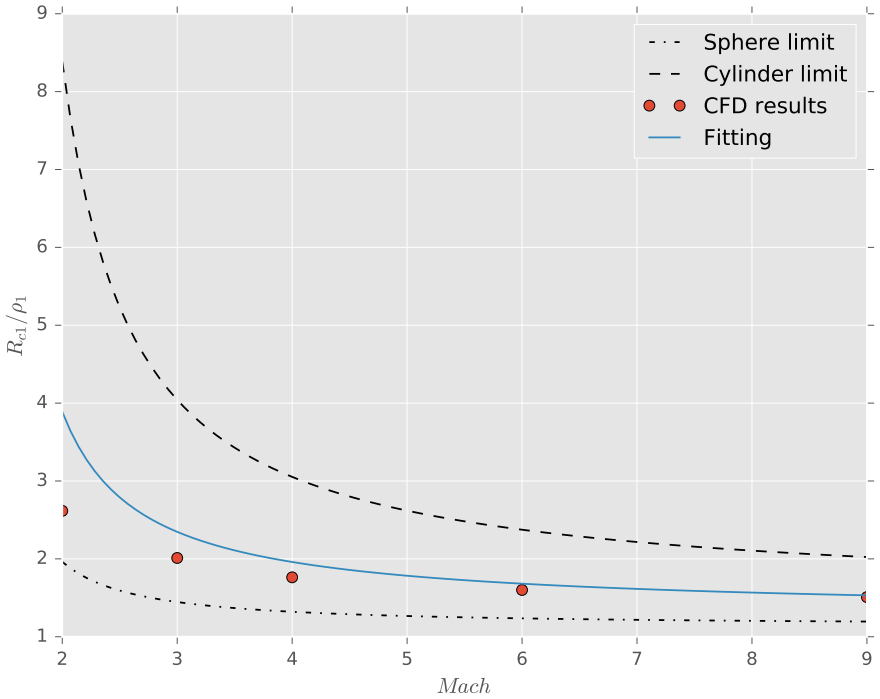


Figure 25: Minimum radius of curvature of the shock as a function of Mach number ($\lambda = 3$)

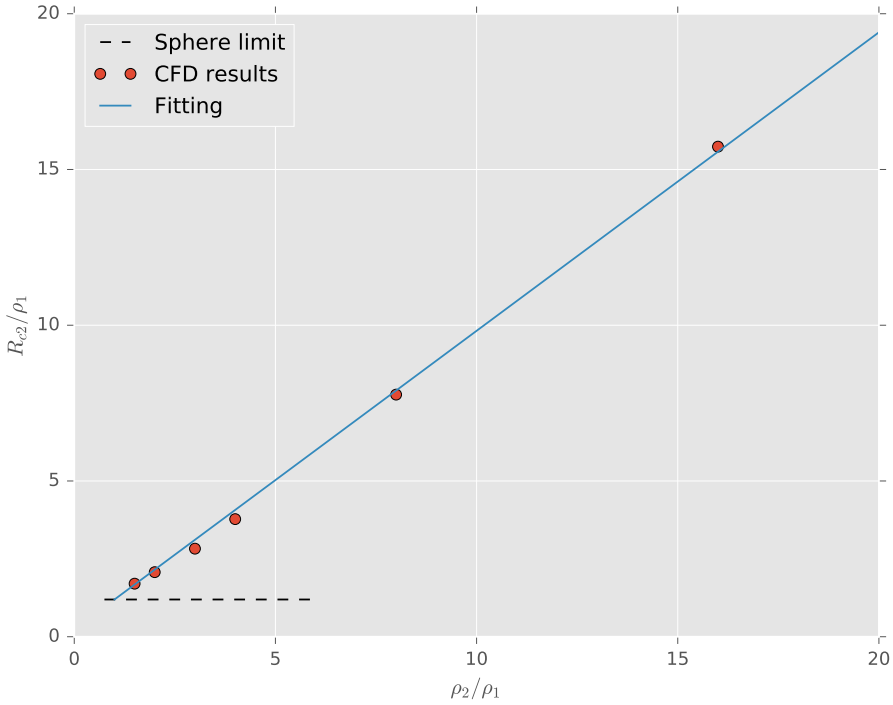


Figure 26: Maximum radius of curvature of the shock as a function of ratio of principal radii of curvature of the geometry ($M = 9$)

with $n_2 = 0.9583619656145238$ computed through least squares. The fitting is again computed at Mach number nine.

From figure 27, the fitting for the maximum radius of curvature seems worse than for stand off distance and minimum radius of curvature. The relative error, however is still below 10%

Tangent cone eccentricity

In 3D, the local inclination method that best describes the flow around a point of the geometry depends on the cross-flow curvature of the surface at that point. For mainly flat surfaces, tangent wedge would be more adequate while, for (cross-flow) curved surfaces, some version of tangent cone would better represent the flow. In general, the flow field produced by an elliptic cone at zero angle of attack could be used to address any of these situations.

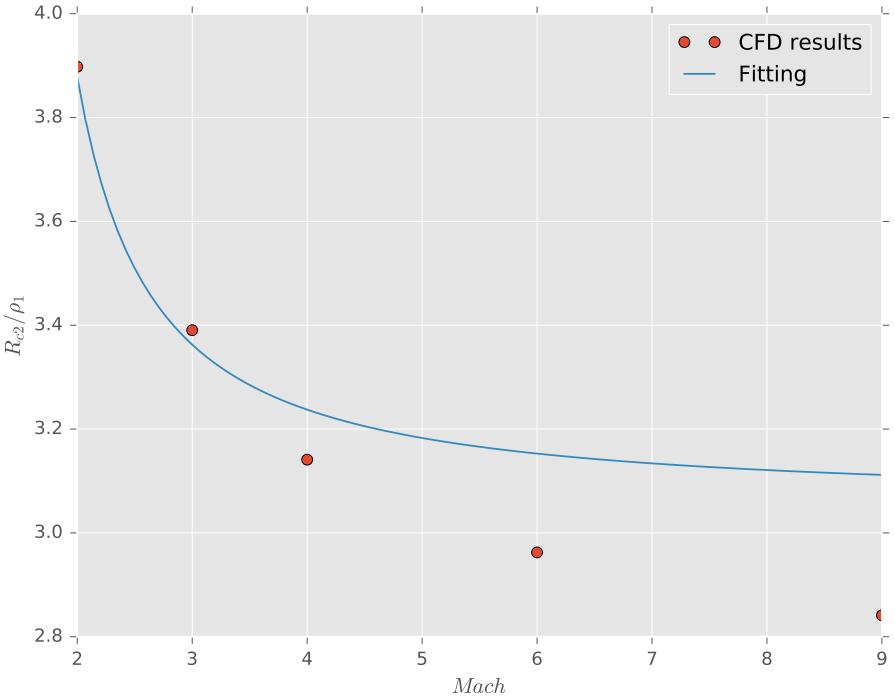


Figure 27: Maximum radius of curvature of the shock as a function of Mach number ($\lambda = 3$)

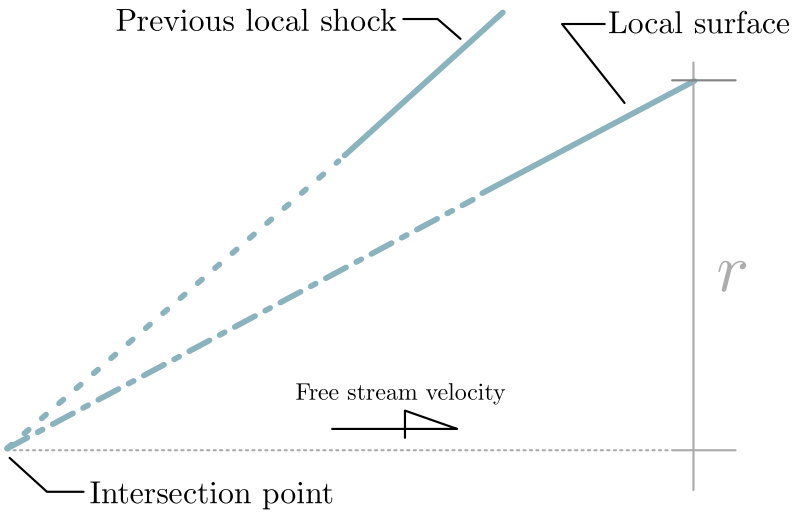


Figure 28: Local radius in the stream-wise direction

The eccentricity of of the local tangent elliptic cone is computed as

$$e = \sqrt{1 - \frac{b^2}{a^2}} \tag{22}$$

where a is the semi-major axis and b the semi-minor axis of the elliptical cross-section of the cone. These values are the maximum and minimum equivalent radii of the local tangent cone in the stream-wise and cross stream-wise directions.

For the stream-wise direction, the radius of the equivalent cone is calculated intersecting in the streamline plane the local surface tangent with the previously computed local shock. This construction is shown in figure 28.

For the cross stream-wise direction, radius of the equivalent cone is calculated from the curvature of the surface section perpendicular to the axis of the local tangent cone. The curvature of a curve embedded in a surface can be expressed as

$$\overline{k} = k_n \overline{n} + k_g (\overline{n} \times \overline{\tau}) \tag{23}$$

where k_n is the normal curvature, $\overline{n}$ is the normal to the surface at the point, k_g the is the geodesic curvature, and $\overline{\tau}$ is the tangent vector of the curve a the

point. The normal curvature can be computed, knowing the principal curvatures of the surface at the point (k_1, k_2), as

$$k_n = k_1 \cos^2 \theta + k_2 \sin^2 \theta \tag{24}$$

where θ is the angle between the tangent vector of the curve and principal direction corresponding to the maximum principal curvature. From here, the curvature of the curve can be computed multiplying equation 23 by $\overline{n}$

$$\overline{k} \cdot \overline{n} = k_n \overline{n} \cdot \overline{n} + k_g (\overline{n} \times \overline{\tau}) \cdot \overline{n}$$

$$k \cos \psi = k_n$$

where ψ is the angle between the normal to the curve and the normal to the surface. Finally obtaining

$$k = \frac{k_n}{\cos \psi} \tag{25}$$

and $r = 1/k$. As stated before, a and b will be the maximum and minimum of these computed radii. These parameters are shown in figure 29.

The principal curvatures and principal directions are computed using the "Estimation of Local Differential Properties of Point-Sampled Surfaces" [38] package of CGAL.

To compute the shock angle that corresponds to the tangent elliptic cone, the perturbation solution shown in Rasmussen[39] is used.

Given the eccentricity (e) of the local ellipse, the semi-major axis of the perturbed cone can be computed as $b = k \tan \theta$, where $k = \sqrt{\frac{e+1}{1-e}}$ and θ the local surface inclination. From here, the mean apex semi-angle of the perturbed cone is computed as $\theta_m = \arctan(b\sqrt{1-e})$. The semi-angle of the basic circular cone that supports the perturbation solution is then

$$\delta = \theta_m + (e^2/32)[3 - 2\sin^2 \theta_m] \sin 2\theta_m \tag{26}$$

Finally, the shock angle of the elliptic cone results

$$\theta_s = \beta - \epsilon g \cos 2\phi \tag{27}$$

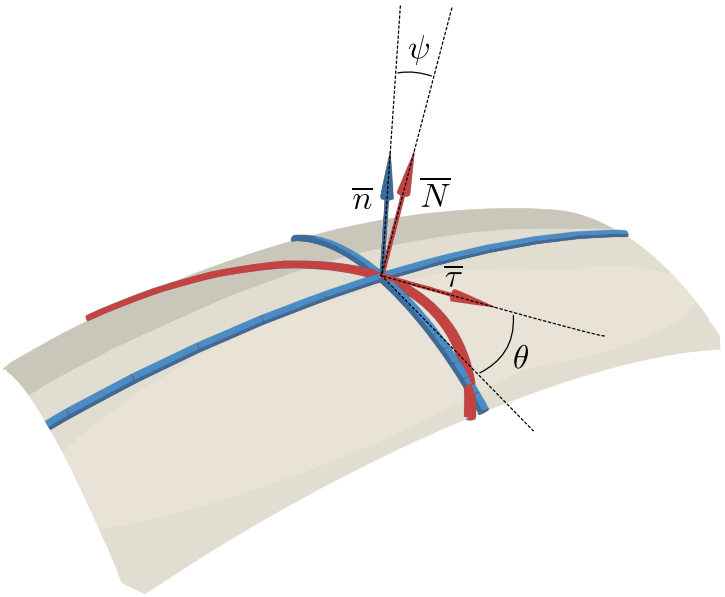


Figure 29: Surface embedded curve (red). Principal directions (blue) and relevant angles are shown

where β is the shock angle that corresponds to the basic circular cone of semi-angle δ (obtained by solving Taylor Maccoll equation),

$$g = 6\sigma^3 \left[\frac{3\cos^{-1}(1/\sigma)}{\sqrt{\sigma^2 - 1}} + \frac{6}{\gamma + 1}(\sigma^6 + \sigma^2) + 3\sigma^4 - \sigma^2 - 5 \right]^{-1} \tag{28}$$

and

$$\epsilon = (e/4)[1 + (e^2/32)[15 - 20\sin^2\theta_m + 8\sin^4\theta_m]]\sin 2\theta_m \tag{29}$$

being $\sigma = \beta/\delta$. The angle ϕ takes the value $\phi = 0$ for surfaces flatter than the local circular cone and $\phi = \frac{\pi}{2}$ for situations where the surface is sharper. These variables are shown in figure 30.

Rasmussen perturbation solution for eccentricity $e = 0.2$ is plotted in figure 31, along with the limits for wedge and circular cone.

Figure 32 shows, for a given local surface inclination of $\theta = 25°$, the variation of the shock angle with eccentricity. The value of the shock angle goes from the circular cone solution (for $e = 0$) to the wedge solution (for $e = 1$).

For high eccentricities, the computed shock angle should approach the one computed for a supersonic wedge. Figure 33 shows shock angle against cone

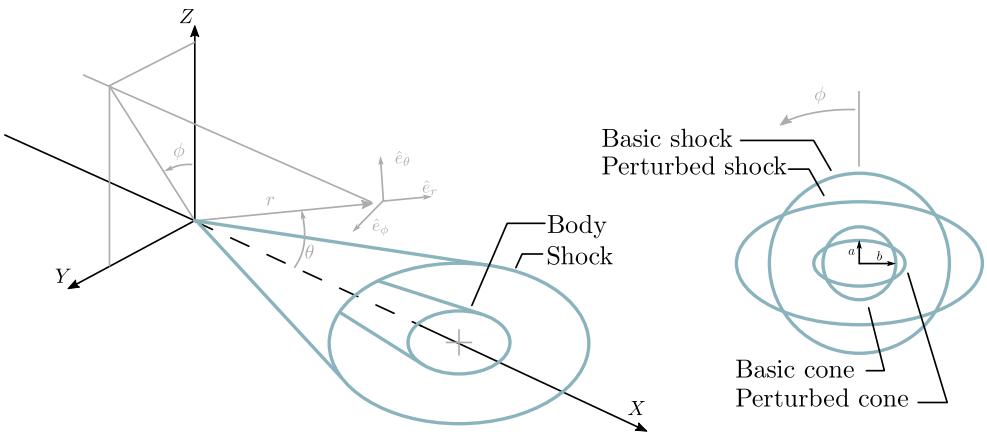


Figure 30: Geometrical quantities involved in the perturbation solution

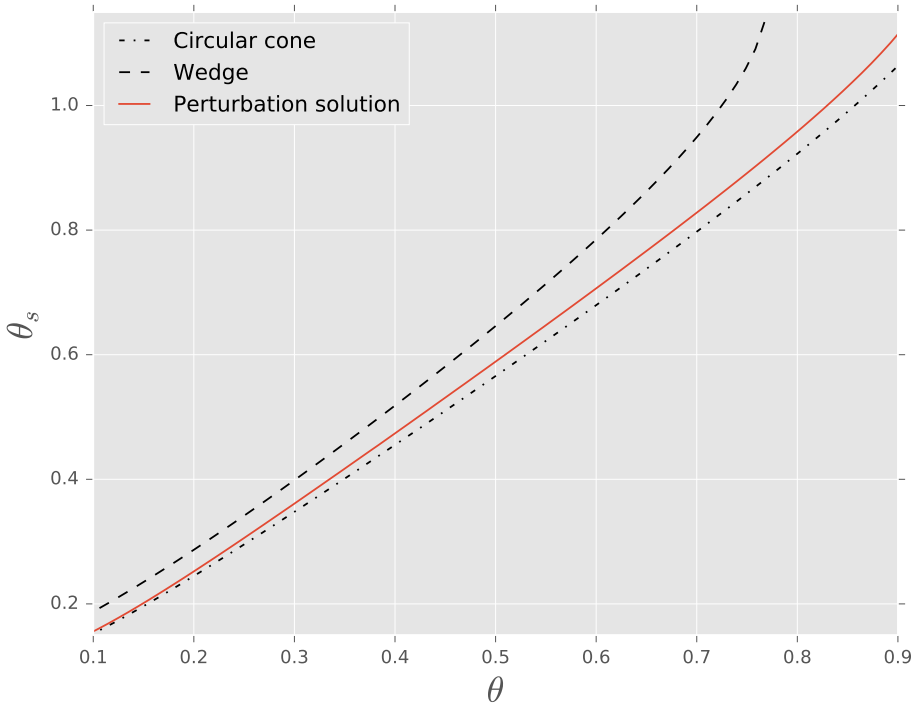


Figure 31: Shock angle as a function of local surface inclination for an elliptic cone with $e = 0.2$

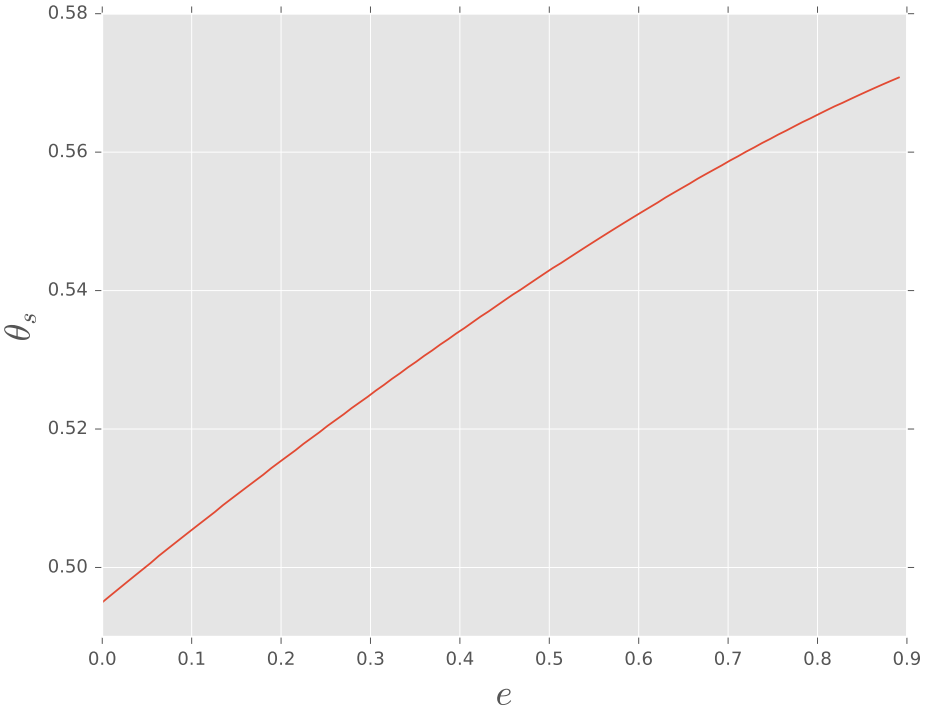


Figure 32: Shock angle for an elliptic cone as a function of eccentricity

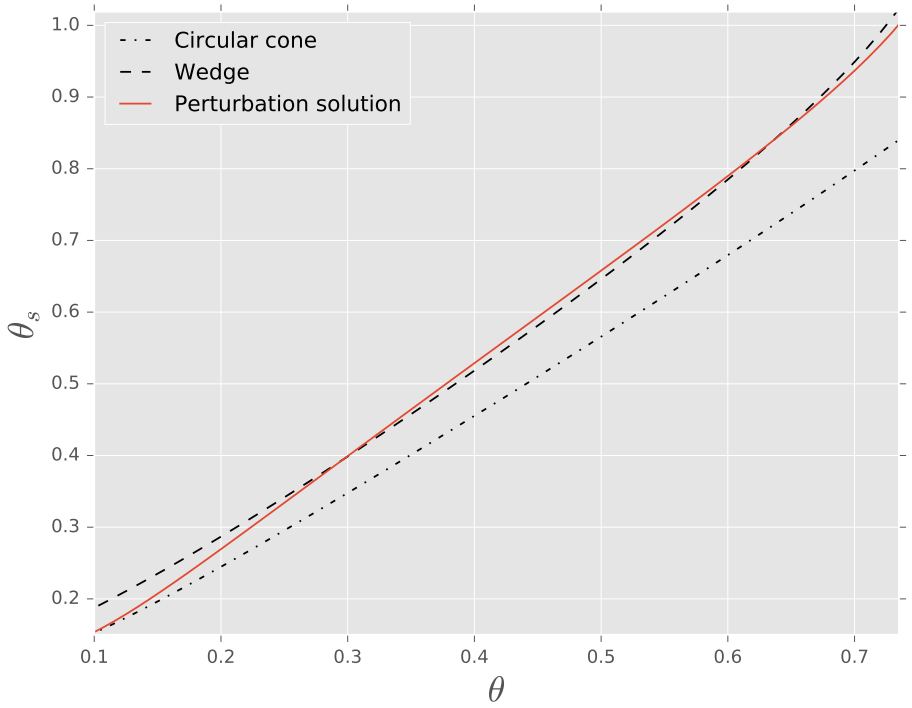


Figure 33: Perturbed solution that corresponds to a highly elliptical cone ($e = 0.99$)

angle for an elliptic cone with $e = 0.99$. The solution is closer to circular cone than to wedge for low angles but is reasonably accurate in the rest of the domain.

While it is possible to use elliptic tangent cone for all geometries, using circular tangent cone in axially-symmetric cases is preferred as the execution of this model is faster and its accuracy, for these cases, is higher.

Concavity in 3D

In three dimensions, two types of concavity are possible, stream wise concavity and cross stream wise concavity. The former refers to the case where the computed streamlines are not convex while the later refers to geometries and flow conditions that present more than one stagnation region. The cross stream direction will be addressed first.

When using the present method on geometries and flow conditions with cross stream wise concavity, even though each streamline is involved in the generation

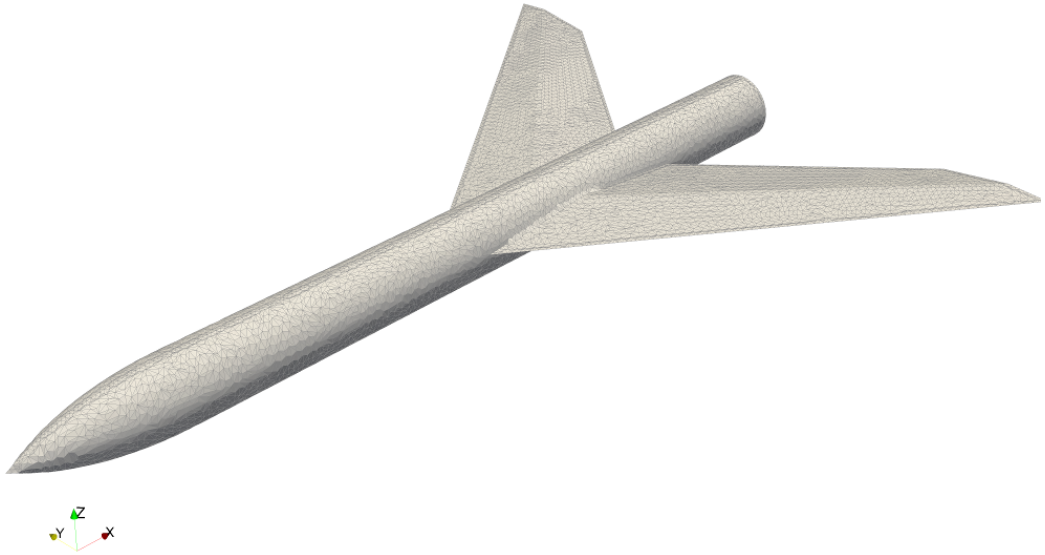


Figure 34: Geometry of an aircraft-like vehicle

of only one local shock not all local shocks produced over streamlines should coalesce into the same global shock. Indeed, on these geometries, it is expected to obtain multiple distinct shocks that interact with each other.

The aircraft-like vehicle shown in figure 34 has been selected as an example to guide this section. Here, the expected shock structure is composed of three shocks, one attached on the nose and two attached on the wing. At certain Mach numbers the shock from the nose of the vehicle could impact the wing.

The shock estimation method described in previous sections produces the point cloud shown in figure 35. In the current method, all those points would be meshed together to obtain the surface of the global shock. That result is not satisfactory, as the different origin of the shocks is not accounted for. Furthermore, this approach would prevent any meaningful study of shock impingement from the nose shock on the wings or any nose shock - wing shock interaction.

Classifying the local shock point cloud into distinct shocks is infeasible. The solution adopted here is to group together streamlines, applying a clustering algorithm on detected stagnation points. These groups will then be meshed separately to produce distinct shock surfaces.

The algorithm selected to group the stagnation points is the density based spatial clustering of applications with noise (DBSCAN)[15]. As its name suggests, it is a density based clustering algorithm. It has been considered the best for this

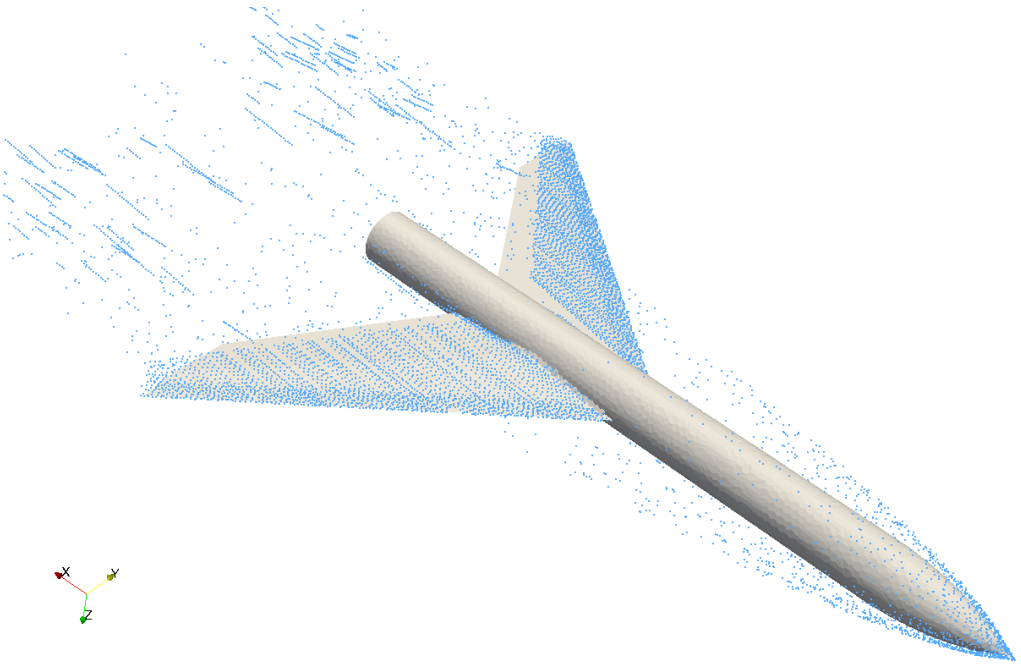


Figure 35: Estimated shock points on a aircraft-like vehicle

application as it supports the discovery of clusters of arbitrary shape, it does not require to know the number of clusters in advance, and it has only two control parameters (ϵ and $\mathtt{MinPoints}$).

The algorithm has been implemented in C++ and it works as follows. Given a point cloud, core points are defined as those which neighbourhood of radius ϵ contains at least $\mathtt{MinPoints}$. Those neighbour points are said to be directly density-reachable from the core point. An arbitrary point q is density-reachable from p if there is a chain of points directly density-reachable from each other where p and q are the extrema of the chain. Finally, two arbitrary points (p,q) are density-connected if there is a point from which both p and q are density-reachable. A point belongs to a cluster if it is density-connected to it. Points not belonging to any cluster are classified as noise. The procedure starts by selecting a random point. If its ϵ neighbourhood contains less than $\mathtt{MinPoints}$ it is classified as noise and another point is selected. Otherwise, a cluster is grown through all density-connected points to it. These points are labelled as belonging to the cluster. Once all density-connected points are reached, a new non-labelled point is selected and a new cluster is grown. The process continues

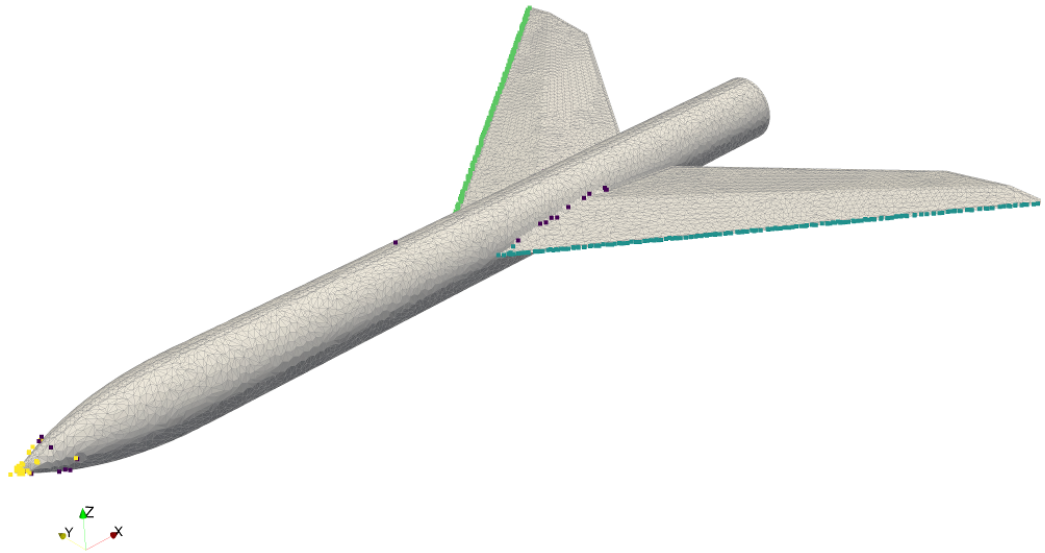


Figure 36: Clusters computed from stagnation points

in this manner until all points have been classified as either belonging to a cluster or noise.

Applying DBSCAN on the stagnation points detected on the plane-like vehicle, three clusters are obtained as seen in figure 36. The first one is the nose stagnation point, while the other two are stagnation lines on the wing leading edge. The non clustered points are considered noise by the algorithm. These three clusters are now ordered in the stream wise direction, with the goal of estimating which clusters are influenced by others. The assumption here is that all clusters upwind of a given one might influence its shock structure. Now that the streamlines are classified, the local shocks can be meshed into these three global shocks. Once the first global shock (nose) is estimated, the flow conditions after it are computed. The streamlines from other clusters which stagnation points lie inside the convex hull of this first shock are assigned its after shock flow conditions instead of using those of the free stream. The local shock and Mach wave inclinations will then be different than those computed if this interference was not taken into account. The procedure continues cluster by cluster in a stream wise order until all clusters are meshed.

The result of this procedure is shown in figure 37. The only influence of the nose shock on the wing shock currently taken into account is that the flow conditions inside the former are not free stream. This will influence the shock

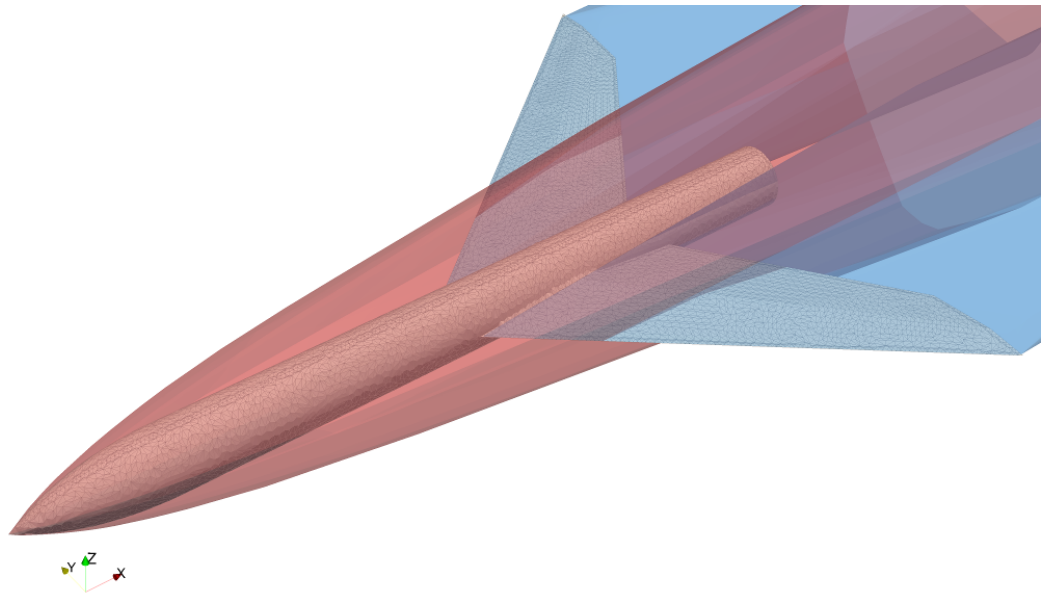


Figure 37: Shock structure around a aircraft-like vehicle

around the wing roots, which will be less inclined with respect of the free stream as the Mach number used for the computation is lower. The shocks are otherwise considered transparent, with no shock-shock interaction.

Concavity in the stream wise direction is addressed in a way similar to the one employed in 2D. Each streamline would generate more than one shock, as opposed to the cross stream wise concavity case. As such, each streamline is divided into concave and convex sections, with each convex section producing a local shock. Between two of such sections of the same streamline, an expansion fan is computed. The start of each convex section is considered a stagnation point for the purpose of clustering.

Once the clusters have been computed, the estimation proceeds as in the cross stream wise concavity case. The clusters are ordered in the stream wise direction and after shock conditions are applied to any stagnation point (meaning here any streamline or streamline section start) contained inside the convex hull of any previously computed shock.

The procedure is shown, in figures 38 and 39, applied to the HB2 vehicle at zero angle of attack. The streamline breaking and clustering produces two distinct clusters, one at the stagnation point on the nose of the vehicle and another annular group at the beginning of the flange.

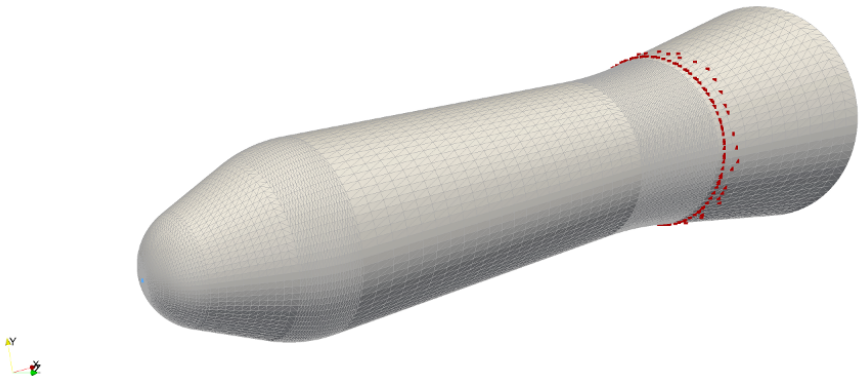


Figure 38: Clusters computed from streamline section starts

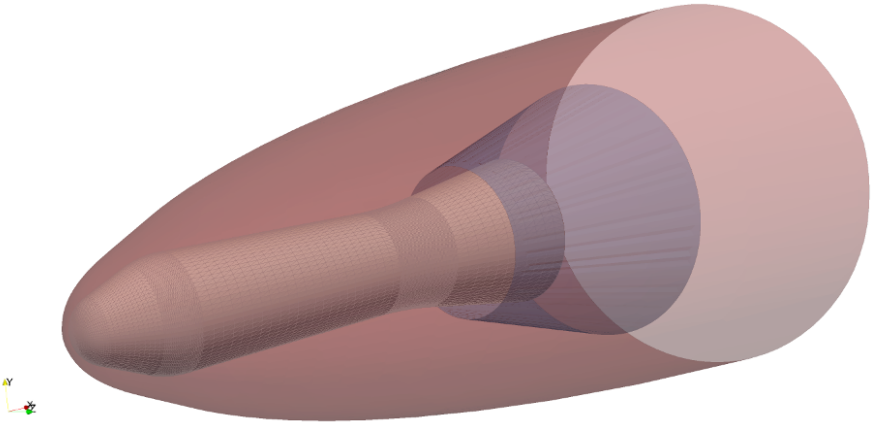


Figure 39: Shock structure around the HB2 vehicle

Two shock estimations are then performed. The bow shock is generated first, using exclusively the first convex part of the streamlines. Then, the conical flange shock is computed over the convex part of the streamlines that lay on the flange. For this zone, the conditions after the bow shock are applied to the computation. It is important to note that this second sock does not start from the surface of the vehicle, but from the convergence point of the compression fan formed on the curved body-flange interface.

3 VERIFICATION

The verification of the method has been performed by comparing the shock estimation against CFD simulations. Each test case explores one aspect of the estimation procedure.

To compare the estimated shock with the CFD simulations, the shock position has to be extracted from CFD. There are multiple strategies to do so. Post et al[37] give an overview of the most succesful ones. The approach taken in the present work is the one presented by Pagendarm and Seitz [35]. The shock position is described as the locus of the maxima of density gradient in the direction of the velocity. This locus is constructed as a zero level isosurface of the second derivative of the density in the direction of the velocity. To detect only maxima (and discard minima and zones of low gradients) this surface is filtered by selecting only the regions where the first derivative is higher than a certain threshold ϵ. This threshold is case dependent, and is selected as to filter erroneous shocks while still capturing the whole shock structure. This shock extraction strategy has been implemented inside Paraview, the postprocessing software used in this work.

The error in the estimation is computed as the mean minimum distance between the vertices of mesh of the estimated shock and the shock position detected from CFD, non-dimensionalized with the length of the geometry. That is, the minimum distance from each vertex of the mesh of the estimated shock to the extracted shock surface from CFD is calculated, and the mean of these values is computed.

CASE 1: CONICAL FLOW

The first case addressed is that of the circular cone at zero angle of attack. This very simple case is used to test that there are no major issues on any of the components (i.e. streamline tracing, conical flow solver, shock surface mesher

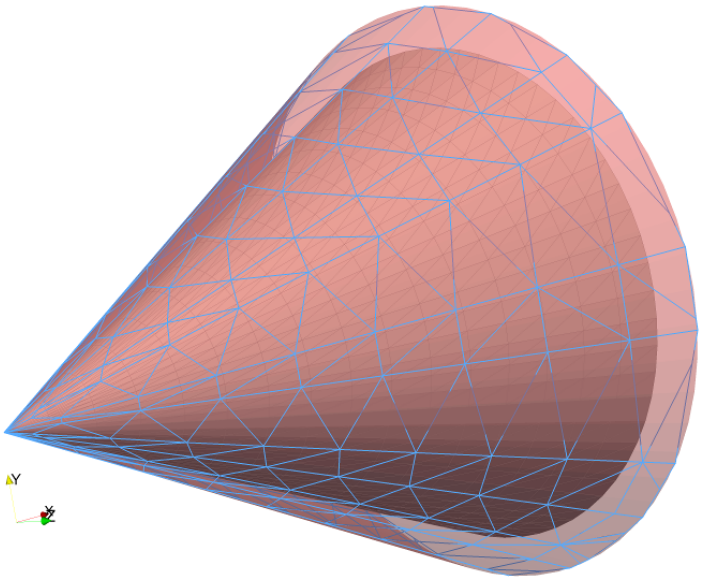


Figure 40: Estimated shock and comparison with TM result for a cone at Mach 9

etc.) of the shock estimation procedure. The solution of the Taylor Maccoll equation readily provides a shock position to which compare the estimation.

Figure 40 shows the mesh of the estimated shock over the conical shock obtained from the Taylor Maccoll equation.

The estimation is as good as the tolerance selected while numerically solving the Taylor Maccoll equation.

CASE 2: AXIALLY-SYMMETRIC ATTACHED SHOCK. TANGENT OGIVE BODY

The basic shock estimation procedure, including the effect of the expansion of the flow along a streamline, is illustrated with a tangent ogive body. This geometry is obtained by revolution of a circle arc, which corresponds to the expression

$$r = \sqrt{\rho^2 - (L - x)^2} + R - \rho$$

where

$$\rho = \frac{R^2 + L^2}{2R}$$

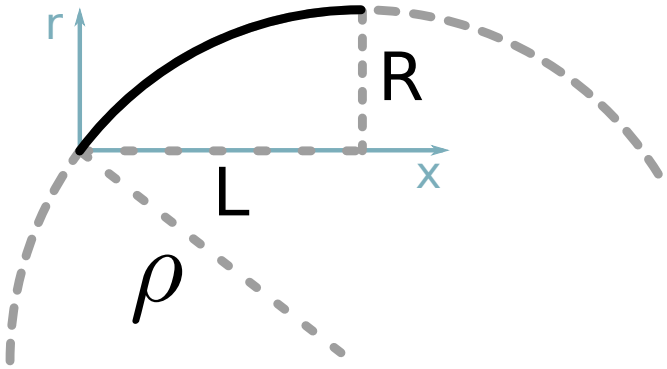


Figure 41: Tangent ogive as a revolution of a circle arc

being R the maximum ogive radius and L the ogive length.

The shock generated is attached and axially-symmetric. This simple test case is a good benchmark to judge how well the basic method works, where no detachment or three dimensional effects are present.

Figure 42 shows the estimated shock, compared with CFD, for Mach 9 conditions. As expected, the distance between the estimation and the actual shock increases with the distance to the stagnation point.

Figure 43 shows the distance between estimated shock and CFD shock, non-dimensionalized with the length of the ogive, along the same section showed in the previous figure. The estimation is quite accurate, with a maximum error below 2.5%.

Regarding the behaviour with Mach number, the error is expected to decrease as Mach number increases. This comes as a result of the reduced effect of the perturbations reflected from the shock. This is indeed the case until Mach 6, with the error slightly increasing after that.

CASE 3: AXIALLY-SYMMETRIC DETACHED SHOCK. BLUNTED TANGENT OGIVE BODY

A blunted tangent ogive has been selected to verify the modification for detached shocks. This geometry is obtained blunting the tangent ogive previously de-

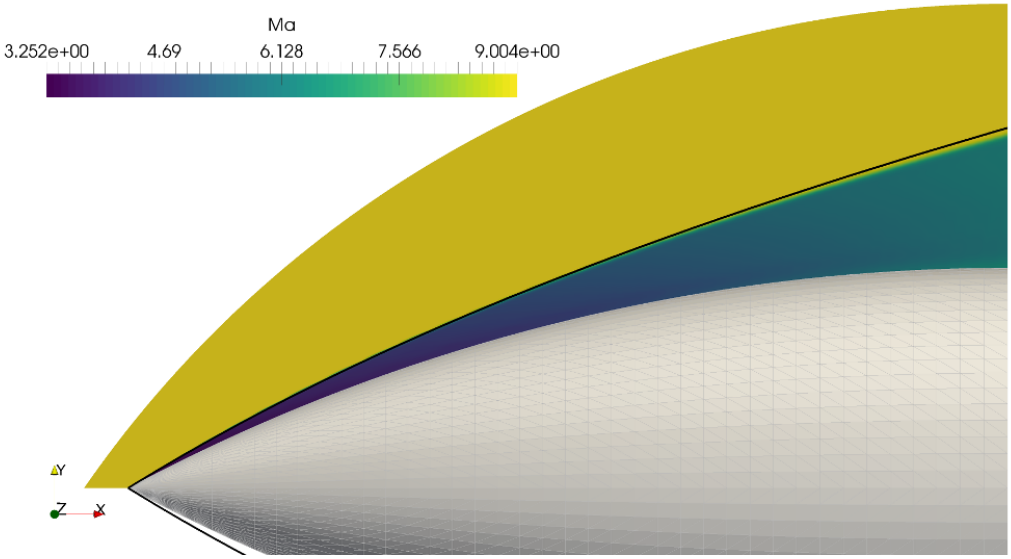


Figure 42: Estimated shock and comparison with CFD for an ogive at Mach 9 and zero angle of attack

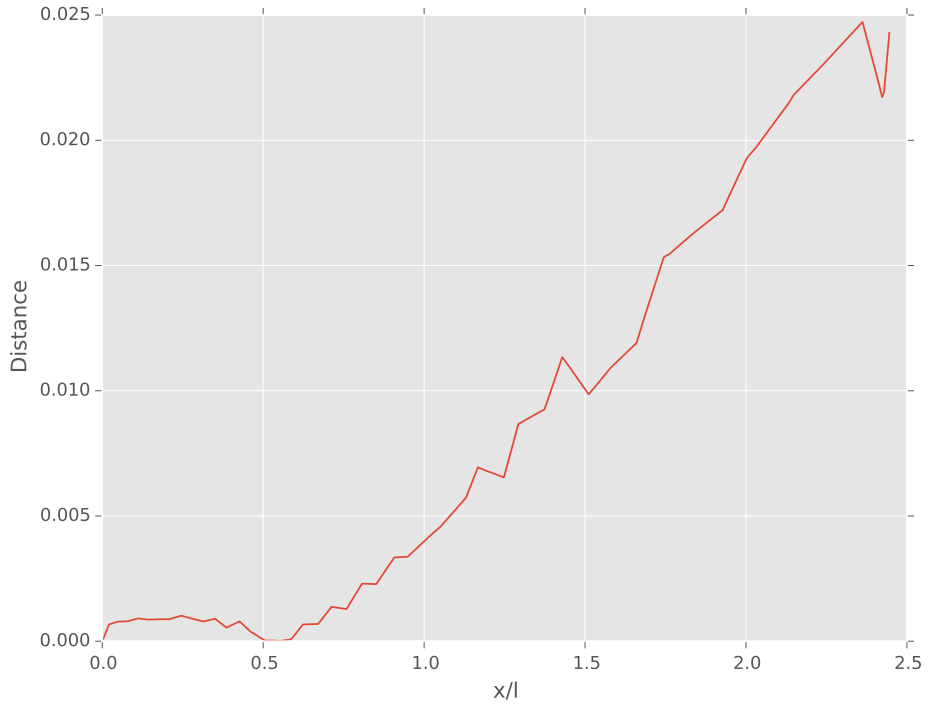


Figure 43: Distance between estimated shock and CFD shock along the surface of the ogive

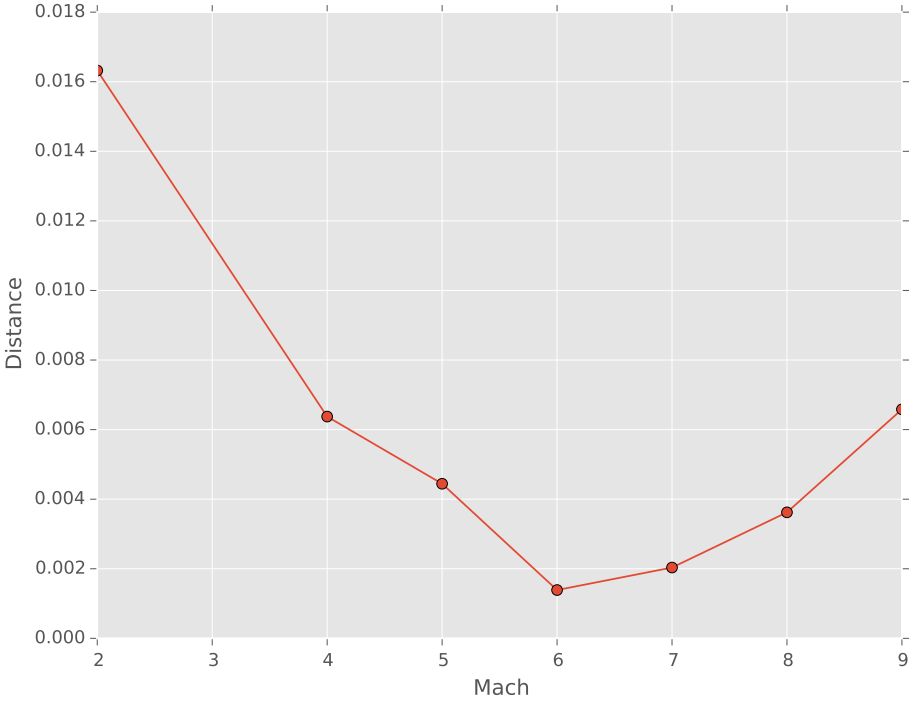


Figure 44: Mean distance between estimated shock and CFD shock as a function of Mach

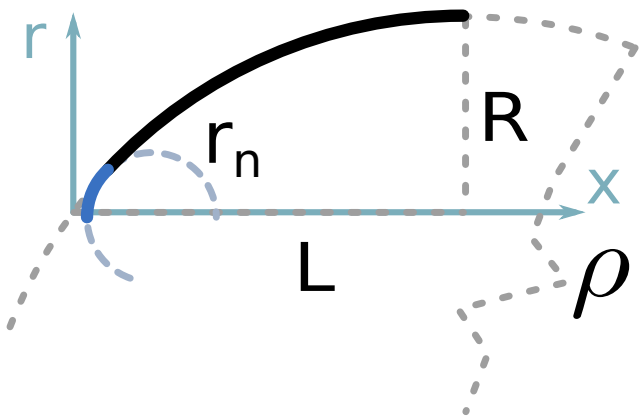


Figure 45: Spherically blunted tangent ogive

scribed with a tangent spherical nose. The center of the spherical nose is given by

$$x_0 = L - \sqrt{(\rho - r_n)^2 - (\rho - R)^2}$$

where r_n is the nose radius and R and L are again the maximum ogive radius and the ogive length, being ρ the radius of the supporting circle. From these values, the tangent point is given by

$$y_t = \frac{r_n(\rho - R)}{\rho - r_n}; \quad x_t = x_0 - \sqrt{r_n^2 - y_t^2}$$

being the apex point

$$x_a = x_0 - r_n$$

Figure 47 shows the error in the estimated shock with Mach number. The error is higher than in the attached case due to the fact that the estimation is not started from a known condition (attached shock). The solution is started from Billig's generalized formula, which is on its own an approximation. On top of that, the estimation of the principal curvatures of the geometry (triangular mesh) at the stagnation point required by the formula is complex and prone to errors. This last point will be explored later in the chapter.

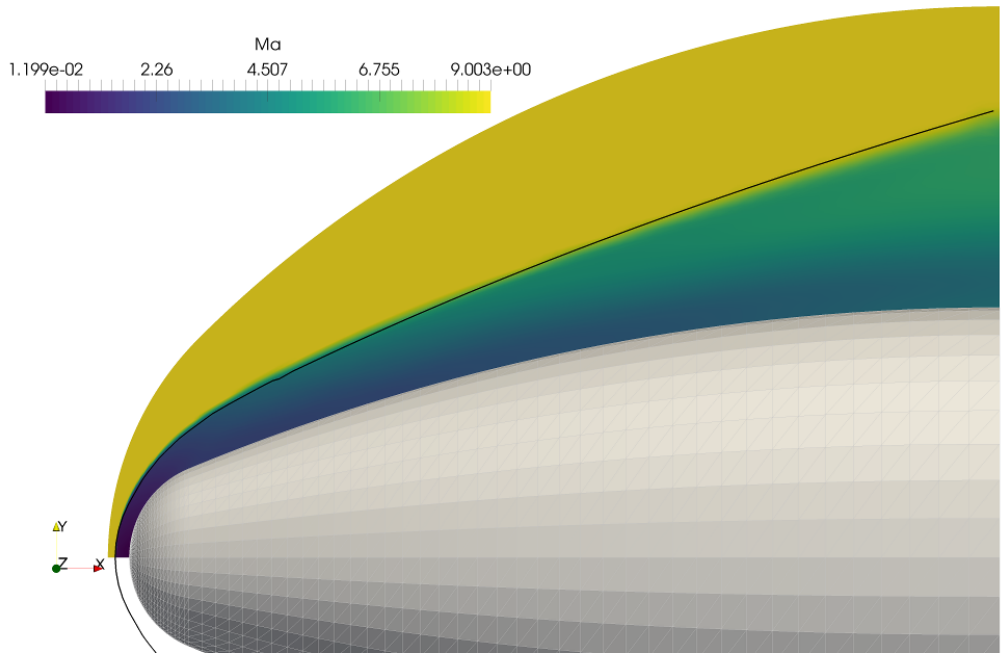


Figure 46: Estimated shock and comparison with CFD for a blunted ogive at Mach 9 and zero angle of attack

CASE 4: ASYMMETRICAL ATTACHED SHOCK. ELLIPTIC TANGENT OGIVE BODY

Next, the effect of non axial symmetry is explored. An elliptic tangent ogive is selected as a test case. This ogive is the same one used as the first test case but scaled in the z direction by a factor of two.

The lack of axial symmetry suggests the use of elliptic tangent cone theory instead of the traditional (circular) tangent cone theory. Figure 49 shows the error in the estimated shock as a function of Mach number. The error is noticeably higher than in the first test case. This is due to the additional approximations being introduced in the elliptic cone perturbation solution, and the need to compute surface curvatures in order to calculate the eccentricity of the local elliptic cone.

Even with the required additional approximations, the result obtained using elliptic tangent cone theory is much better than the one obtained using circular tangent cone theory. Figure 50 shows a comparison between the two estimations,

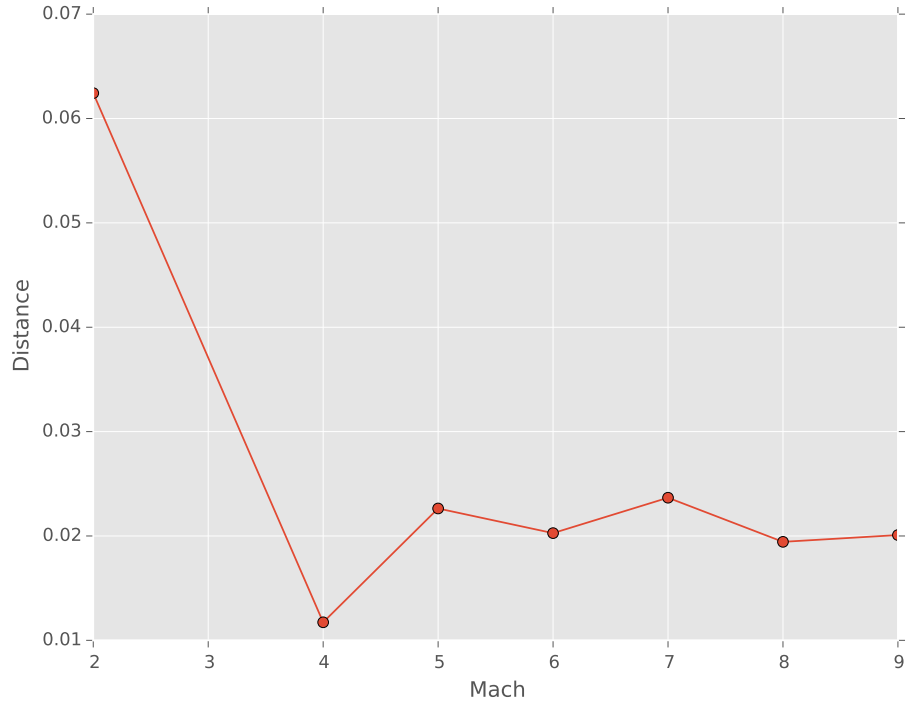


Figure 47: Mean distance between estimated shock and CFD shock as a function of Mach

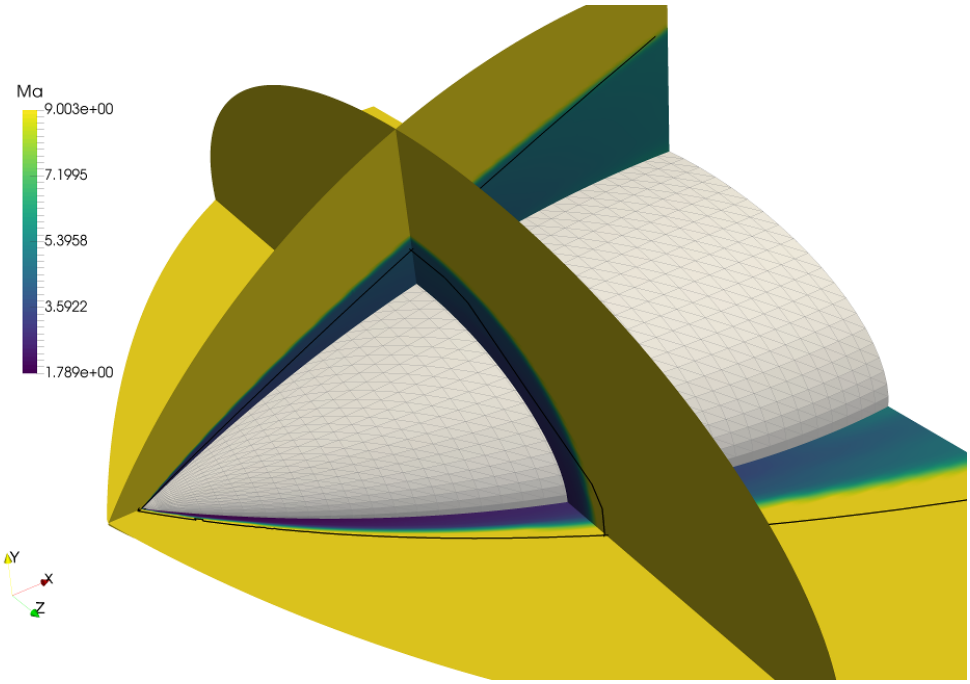


Figure 48: Estimated shock and comparison with CFD for an elliptic ogive at Mach 9 and zero angle of attack

with the circular cone theory presenting errors twice as high as the ones present with elliptic cone theory.

CASE 5: ASYMMETRICAL DETACHED SHOCK. BLUNTED ELLIPTIC TANGENT OGIVE BODY

The combined effect of bluntness and lack of axial symmetry provides a challenging test case. The geometry is a blunted elliptic tangent ogive. The main feature of the method tested here is is the generalized Billig formula, required to start the shock estimation. As shown in figure 52, the formula accurately captures the shock shape around the stagnation point, with the expected higher error along the principal direction associated with the smallest principal curvature.

Figure 53 shows the error in the estimated shock with Mach number. The error increases with Mach number due to the increased error in the generalized Billig's estimation, as seen in figure 27.

The effects of the curvature calculation are also apparent in this test case.

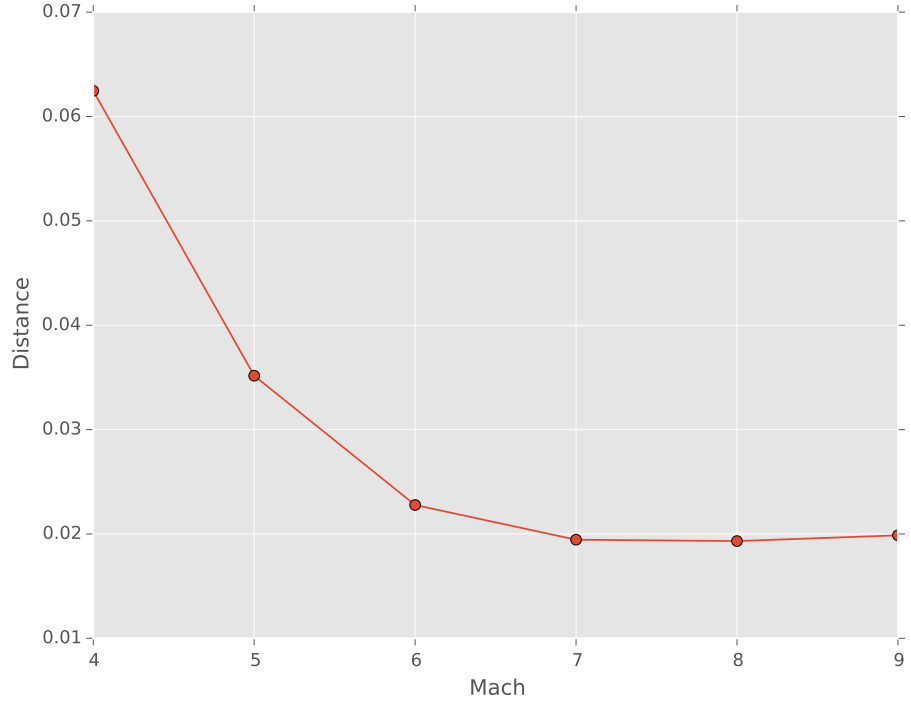


Figure 49: Mean distance between estimated shock and CFD shock as a function of Mach

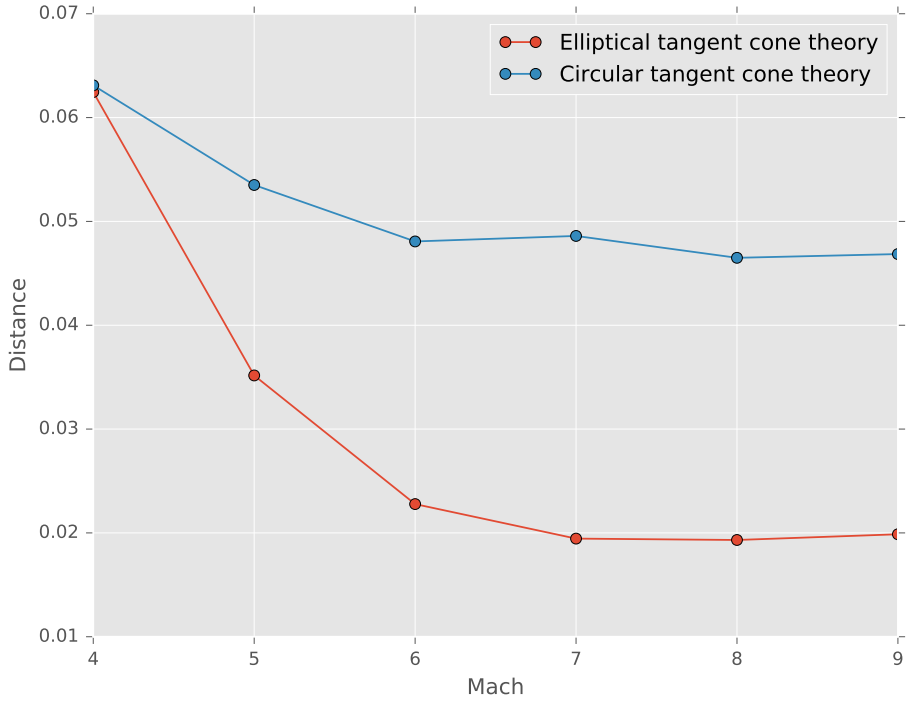


Figure 50: Comparison between estimations using circular tangent cone theory and elliptical tangent cone theory

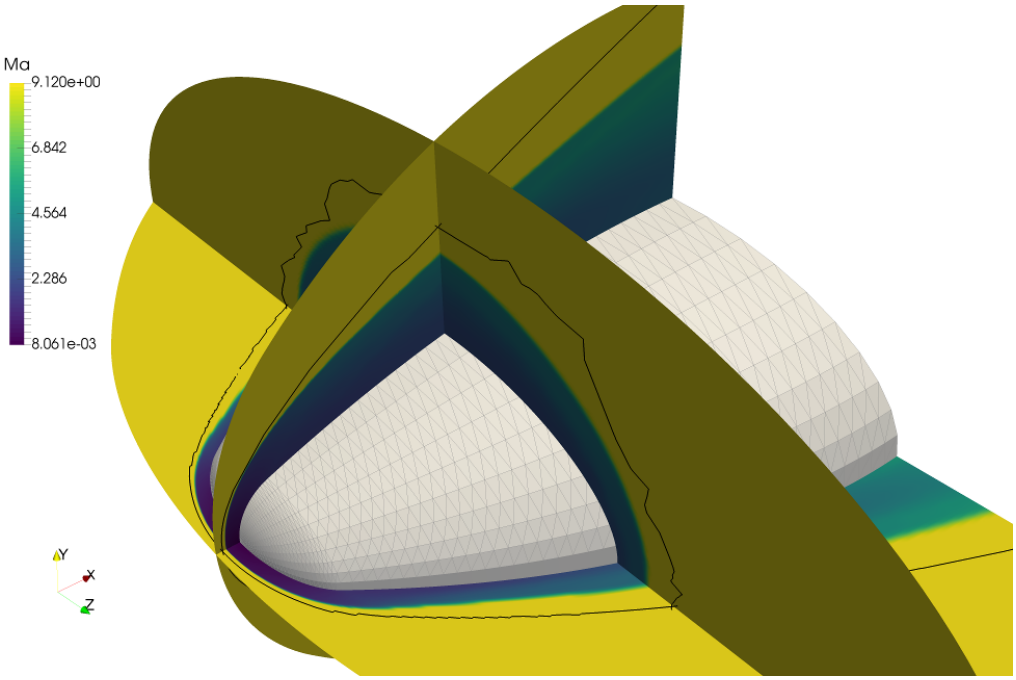


Figure 51: Estimated shock and comparison with CFD for a blunted elliptic ogive at Mach 9 and zero angle of attack

CASE 6: MULTI-COMPONENT SHOCK STRUCTURE. HB-2

The test case selected for analysing the multi-component shock structure estimation is that of HB-2. Here, two shocks are generated, a bow shock around the blunted nose and a conical shock on the aft flange of the geometry. These two shock are shown in figure 54 along with the Mach field resulting from a CFD simulation.

The error in the estimated bow shock with Mach number is shown in figure 55. No clear behaviour can be discerned, but the estimation is quite accurate at all tested Mach numbers.

While the procedures required for the estimation of the bow shock position have been verified before (blunted tangent ogive), the secondary conical shock showcases new aspects of the computation. The estimation of the position of the secondary shock requires the flow conditions after the bow shock, as well as the computation of the compression fan present in the body-flange shoulder.

Figure 56 shows the error in the estimated secondary shock with Mach number. The geometry of the flange itself is very simple, a cone as the one verified in

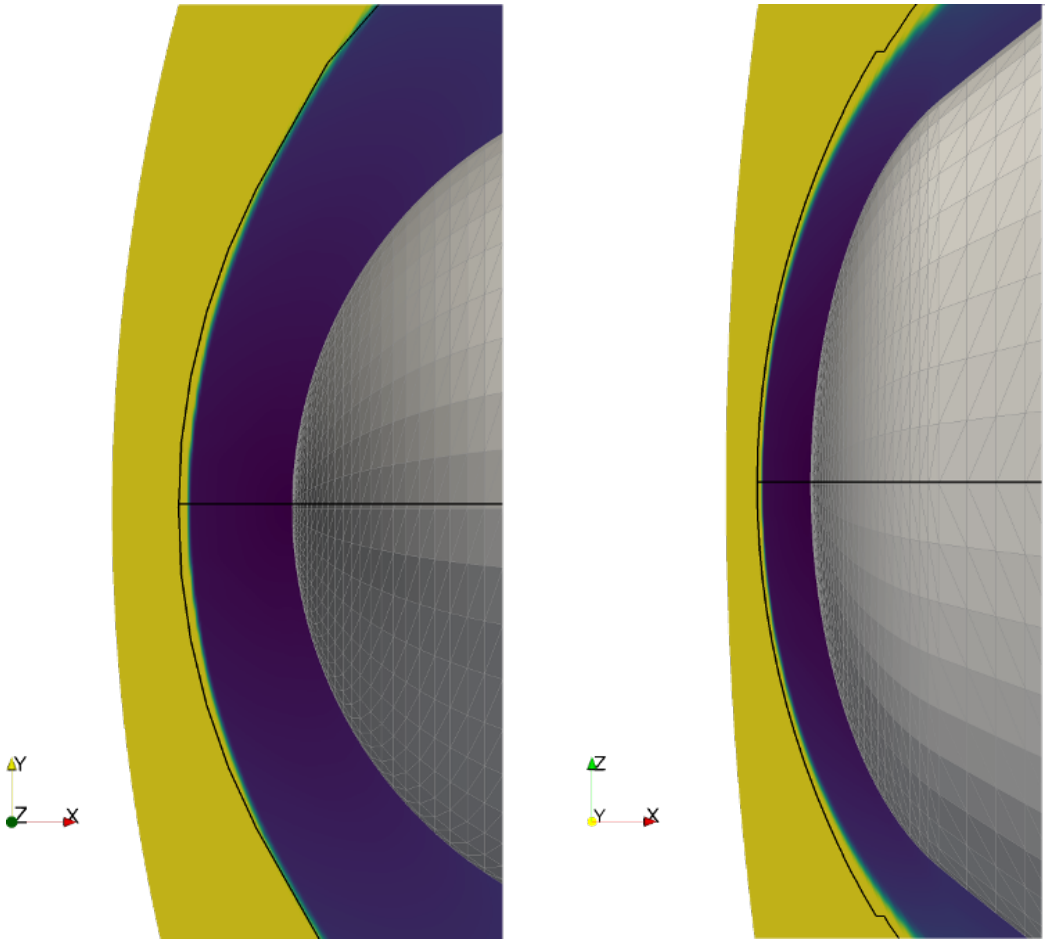


Figure 52: Detail of shock estimation around the stagnation point over CFD Mach field. Left, principal direction of maximum curvature. Right, principal direction of minimum curvature

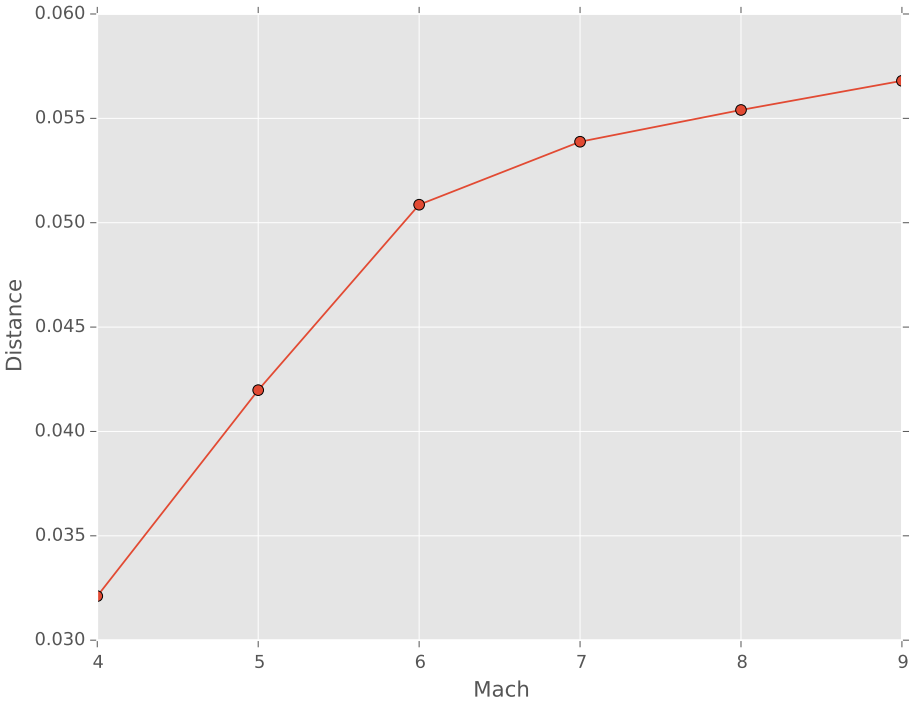


Figure 53: Mean distance between estimated shock and CFD shock as a function of Mach

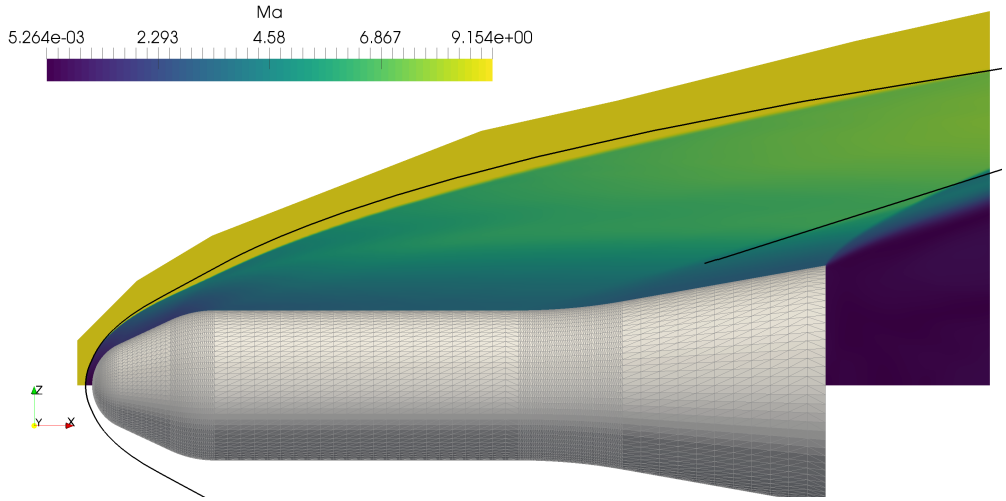


Figure 54: Estimated shock structure and comparison with CFD for the HB-2 at Mach 9 and zero angle of attack

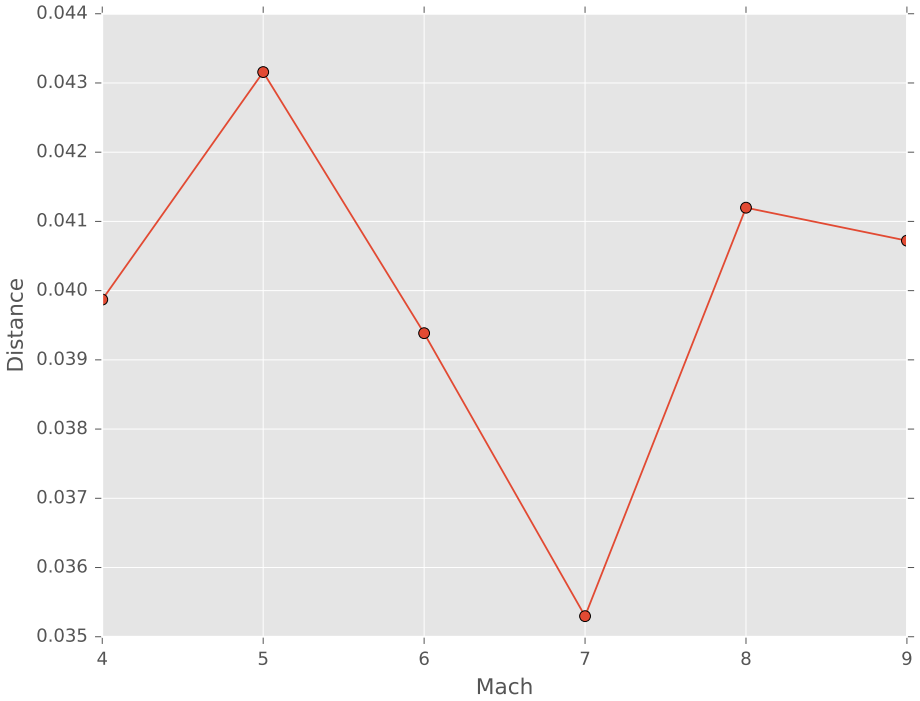


Figure 55: Mean distance between estimated (bow) shock and CFD shock as a function of Mach

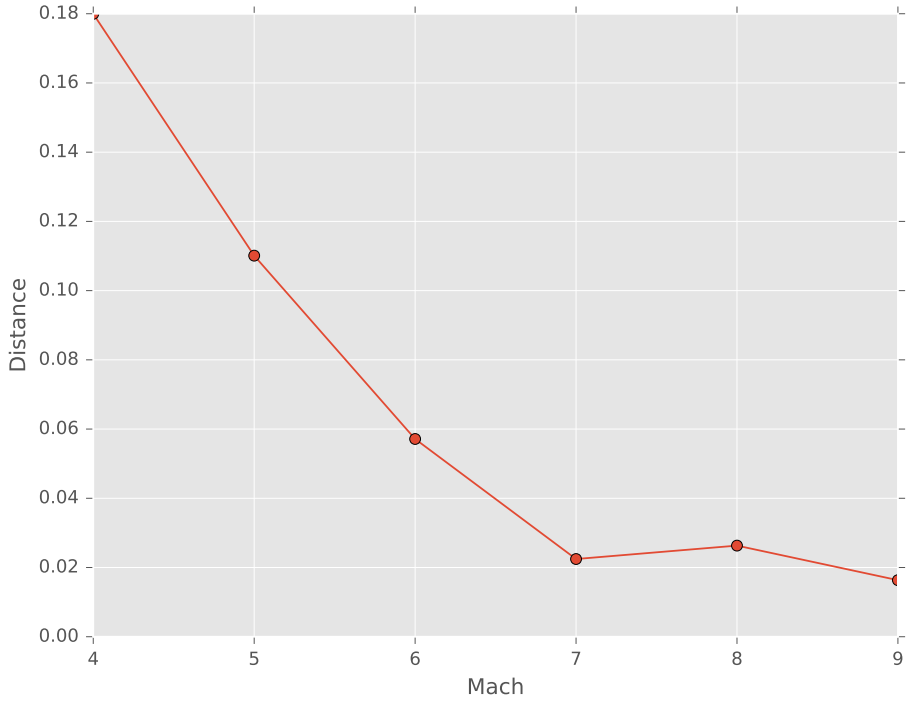


Figure 56: Mean distance between estimated (flange) shock and CFD shock as a function of Mach

case 1 to great accuracy. A lot of aspects contribute to the final error shown in the figure. The error on the estimation of the bow shock propagates to the computation of its after shock flow conditions, that are then used to compute the compression fan that sets the first point of the estimated flange shock and that feed the estimation process. The solution degrades with decreasing Mach number, as it does the estimation of the bow shock and its after shock conditions.

CASE 7: ANGLE OF ATTACK. TANGENT OGIVE BODY

In this section, the effect of the angle of attack is studied. The geometry used is the tangent ogive, seen before while verifying attached shocks.

The angle of attack introduces an asymmetry in what would otherwise be an axially symmetric flow. As such, elliptical tangent cone might be more suitable

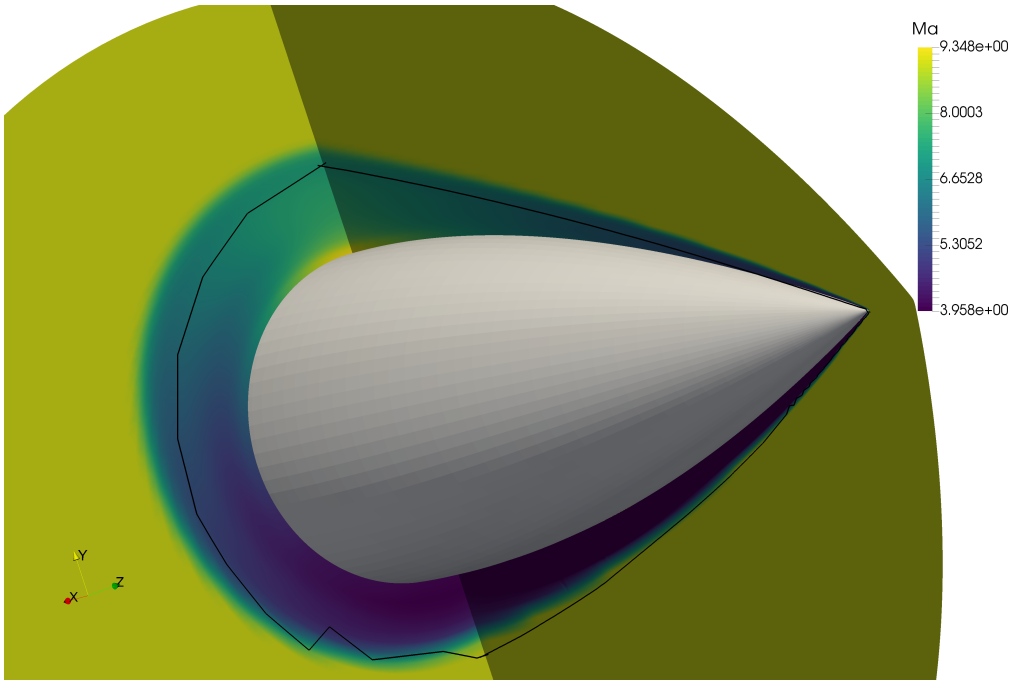


Figure 57: Estimated shock and comparison with CFD for an elliptic ogive at Mach 9 and 10° angle of attack

than traditional circular cone. Figure 58 shows the error in the estimated shock while using elliptical tangent cone for the computation of local shocks.

Meanwhile, figure 59 shows a comparison between the error in the estimated shock while using elliptical tangent cone versus circular tangent cone for the computation of local shocks. While in the circular case the errors are overall smaller, they increase with angle of attack as the asymmetry becomes more pronounced. In the elliptic case, however, the errors seem independent from the angle of attack. Elliptic cone appears therefore theoretically better suited for this case due to this error independence, even if practically it is still convenient to use the circular approximation until the problems with elliptic cone are addressed. The higher error comes from the need to compute curvature values on the geometry.

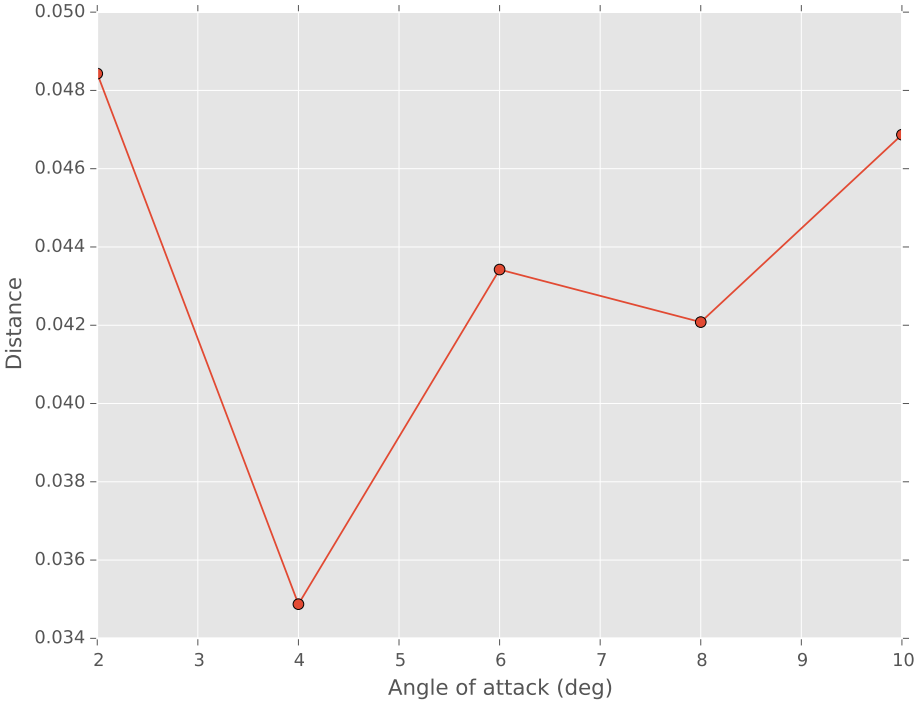


Figure 58: Mean distance between estimated shock and CFD shock as a function of angle of attack

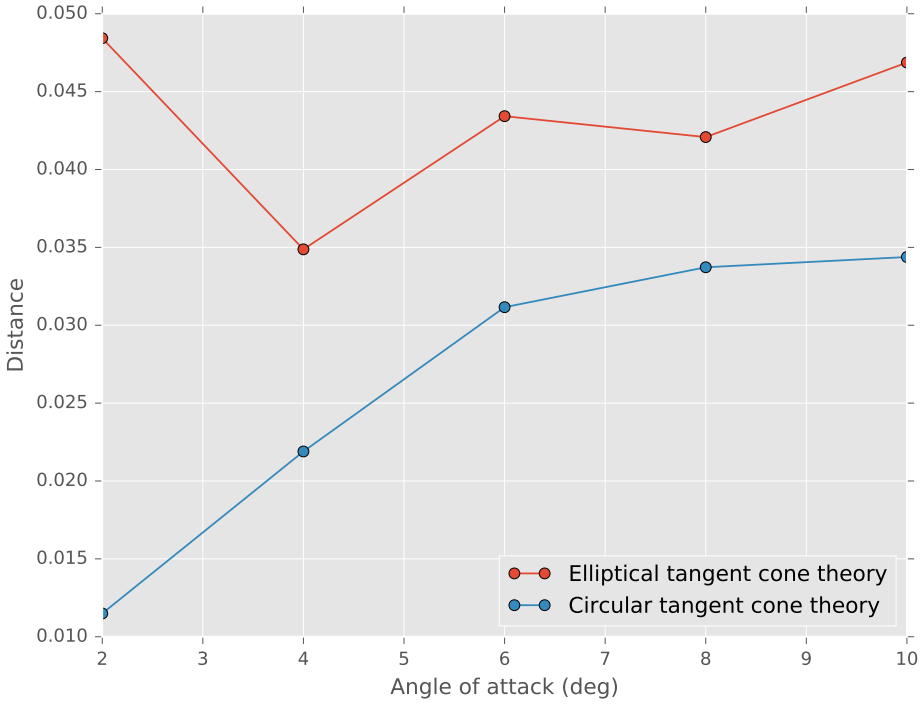


Figure 59: Comparison between estimations using circular tangent cone theory and elliptical tangent cone theory

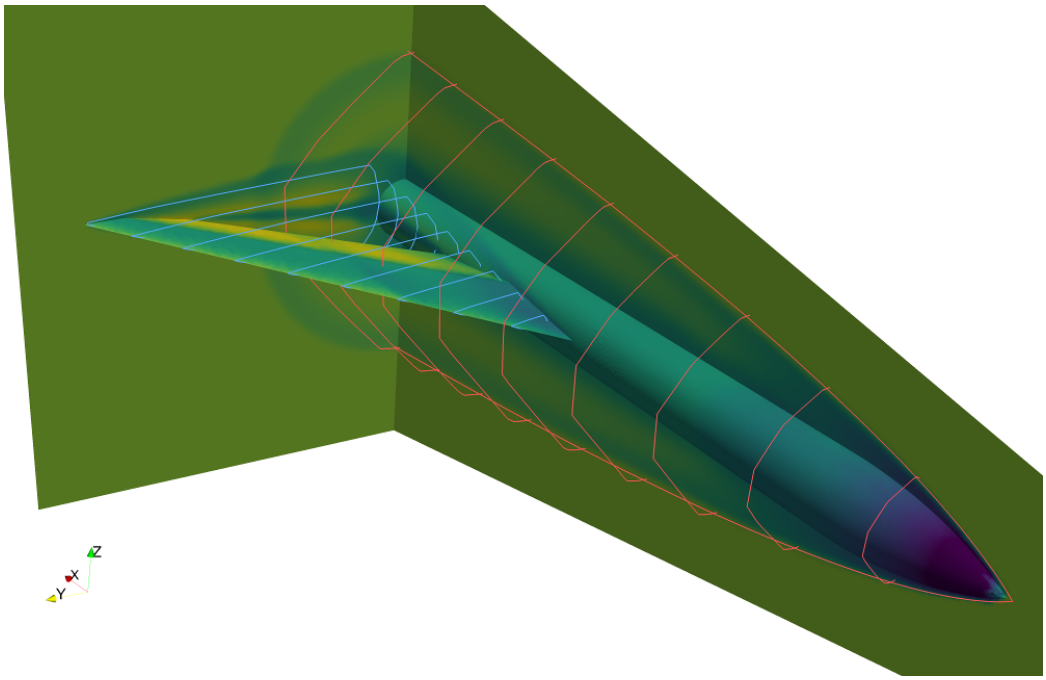


Figure 60: Shock structure over CFD results

CASE 8: COMPLEX SHOCK STRUCTURE. AIRCRAFT-LIKE GEOMETRY

A more general geometry of engineering interest is showcased here. The CFD simulation for this case is very computationally expensive, therefore only one flight condition (Mach 9 at zero angle of attack) has been simulated.

The estimated shocks for both fuselage and wing qualitatively match the shock position from CFD. There is a complex wing shock - nose shock interference which is not captured by the estimation, with the only effect taken into account being the flow conditions after the nose shock for the part of the wing shock that lies in its wake. The complex shock interference, however is limited to a small zone of the flow, with the overall result being satisfactory.

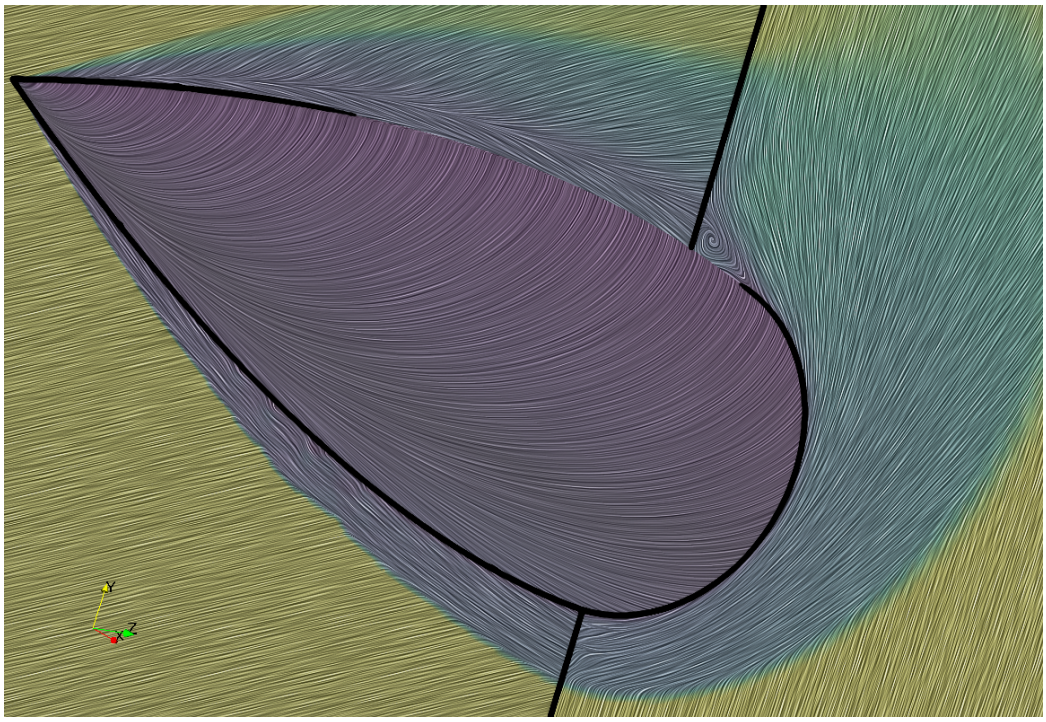


Figure 61: Tangent ogive at high angle of attack

LIMITATIONS OF THE METHOD

To finalize the verification, the limitations of the method are explored. These are situations where the models underlying the shock estimation procedure are ill suited for the conditions where they are applied or where their shortcomings become apparent.

The tangent ogive body at an angle of attack will serve to illustrate some of these issues.

The first stage of the shock estimation procedure is the computation of surface streamlines over the geometry. These are computed under the assumptions of Newtonian flow (flow particles transfer momentum normal to the surface while maintaining their momentum tangent to it). This assumption is reasonably accurate on the windward part of the geometry, but fails on the leeward side. An extreme manifestation of this can be seen in a tangent ogive at high angle of attack. In this case, along with the usual shadowing not contemplated by Newtonian theory, there is flow separation which further complicates the surface streamlines.

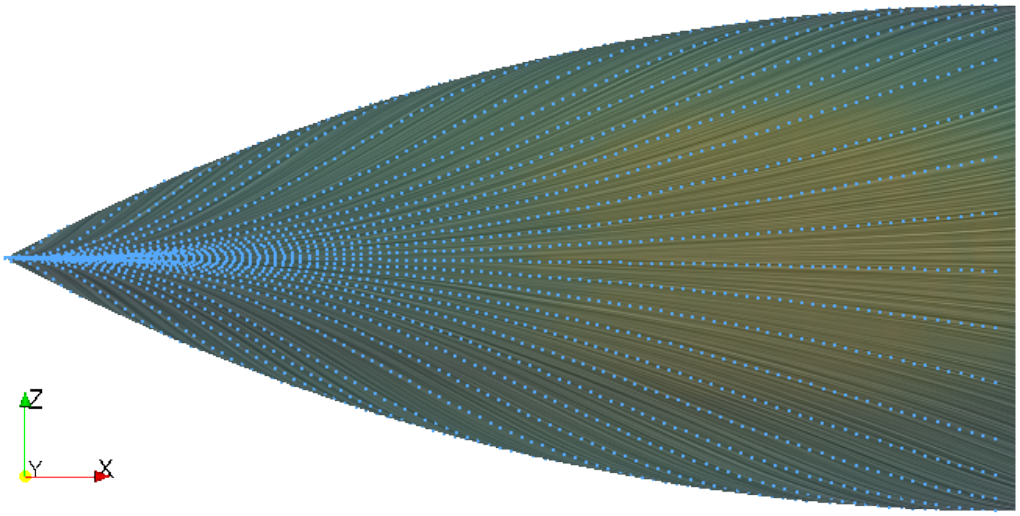


Figure 62: Windward side of the ogive

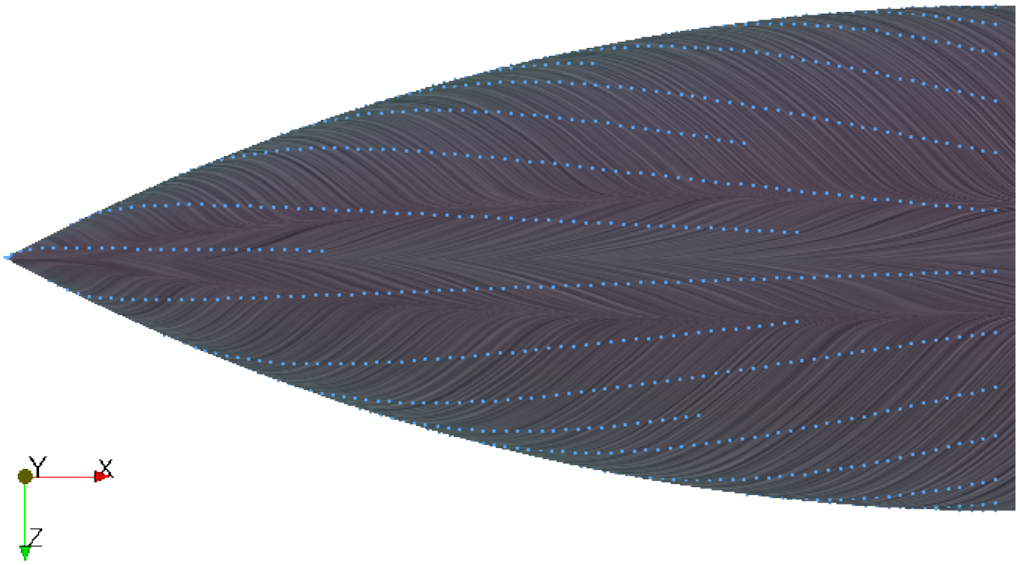


Figure 63: Leeward side of the ogive

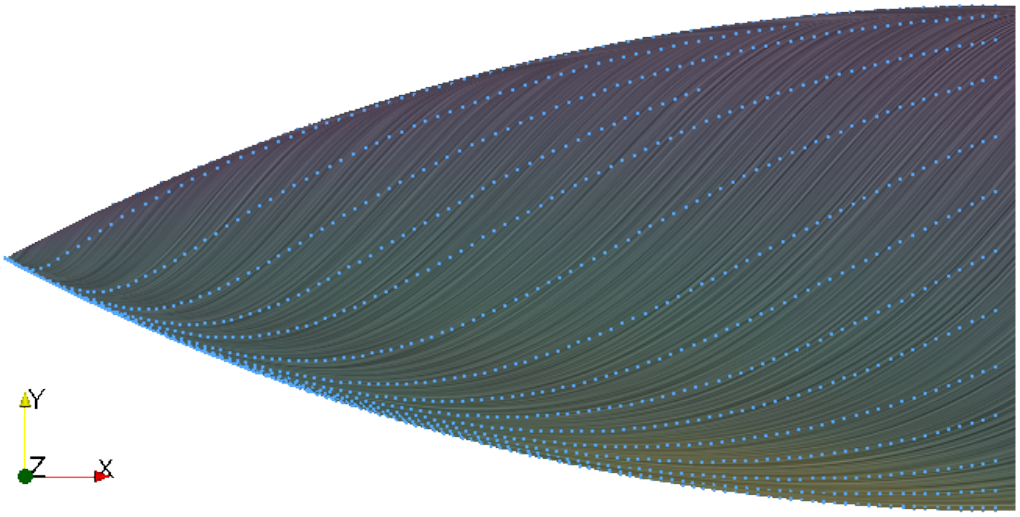


Figure 64: Side view of computed streamlines over CFD surface streamlines

The computed streamlines successfully approximate the CFD surface streamlines for the windward side of the body. On the leeward side, however, there is little similarity between computed and actual streamlines.

Figure 64 shows a side view of the ogive, with the computed streamlines superimposed over the streamlines obtained from CFD. It can be seen how the computed streamlines drift way from the actual ones as the flow wraps around the geometry.

The difference between actual streamlines and approximated ones will cause the shock to be computed from the wrong conditions. This effect, however, is not as serious as it might seem. Due to the expansion over the body, a relative small zone around the stagnation point is responsible for the shock structure at a distance many times its size. The streamline segments on the leeward side of the geometry only influence the shock far away downstream. This can be observed in figure 65.

Another aspect that can be studied from this test case is the influence of the curvature estimation. The presence of angle of attack suggest the use of elliptical tangent cone theory to compute the local shocks. The application of this theory requires the computation of principal curvatures and principal directions over

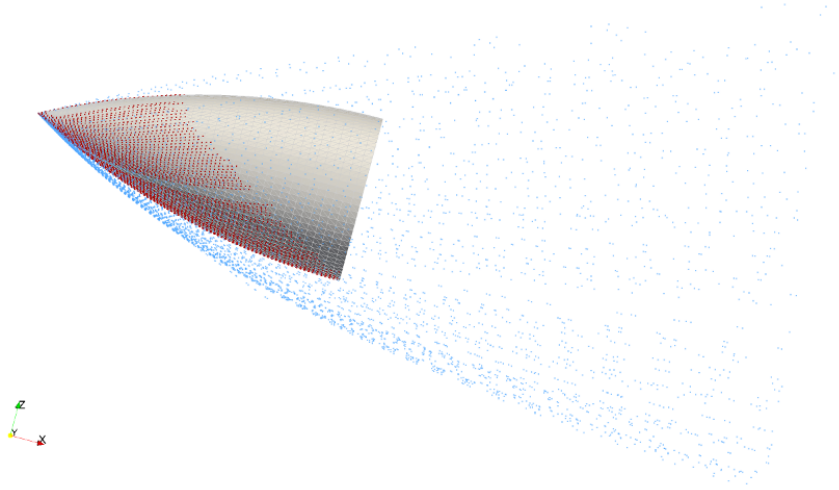


Figure 65: Surface streamlines (red) responsible for shock structure (blue)

the geometry. As the geometry is described by a triangular mesh, this procedure is complex and prone to numerical errors.

For the tangent ogive geometry, the curvatures are easily computed analytically as follows.

The radius of the circle that supports the ogive can be computed from the maximum ogive radius R and the ogive length L

$$\rho = \frac{R^2 + L^2}{2R}$$

The radius of the ogive at any x coordinate along it is then

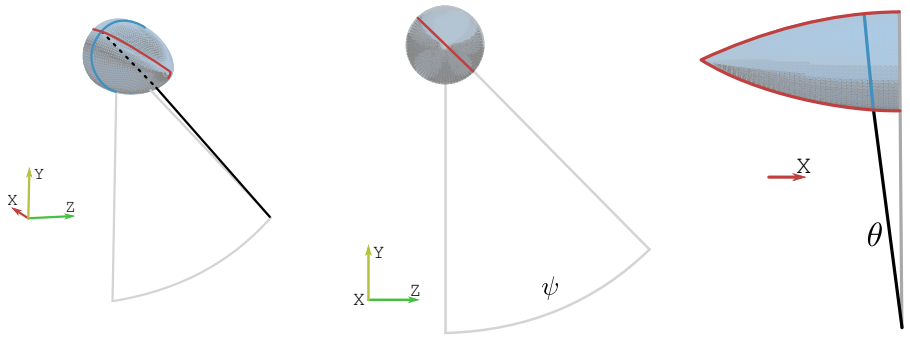
$$r = \sqrt{\rho^2 - (L - x)^2} + R - \rho$$

with $x \in [0, L]$. The point where the curvatures are computed can be defined by the angles θ and ψ

$$\theta = \arctan\left(\frac{L - x}{\rho - R}\right)$$

$$\psi = \arctan\left(\frac{z}{y}\right)$$

The curvatures are then calculated as



(a) Isometric view **(b)** Front view **(c)** Side view

Figure 66: Principal directions for a point on a tangent ogive

$$k_1 = -\frac{1}{\rho}$$

$$k_2 = -\frac{1}{r/\cos\theta}$$

with the principal directions (v_1, v_2) and the normal vector (n) given by

$$v_1 = (-\cos(\theta),\ -\sin(\theta)\cos(\psi),\ -\sin(\theta)\sin(\psi))$$

$$v_2 = (0,\ -\sin(\psi),\ \cos(\psi))$$

$$n = (-\sin(\theta),\ \cos(\theta)\cos(\psi),\ \cos(\theta)\sin(\psi))$$

Figure 67 shows an histogram of the values per vertex of the first principal curvature as computed by CGAL. The analytical result indicates that this value should be constant and, for the parameters L and R considered in this test, $k_1 = -0.117647$. The distribution of CGAL values is centred on the analytical value, but the presence of outliers is readily seen on the plot.

Regarding the second principal curvature, figure 68 shows a comparison between CGAL obtained and analytical values. While the distributions are similar, differences can be seen again between the two. Differences increase as the radius of curvature becomes smaller, towards the nose of the ogive. The estimation in zones of high curvature is harder due to the limitations imposed by the mesh

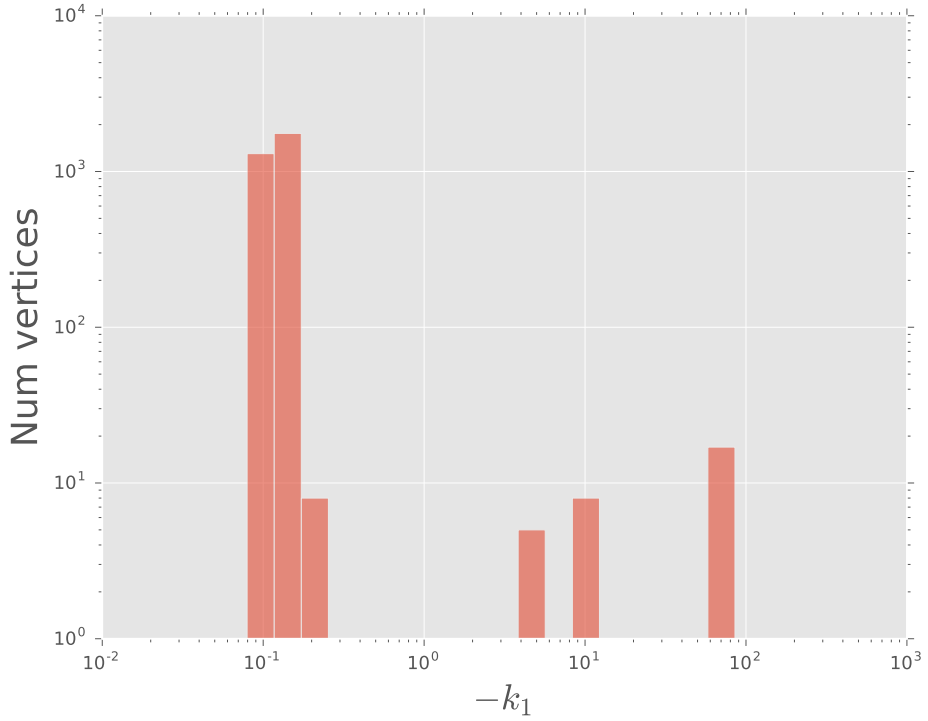


Figure 67: Histogram of first principal curvature at each vertex of the geometry as computed by CGAL

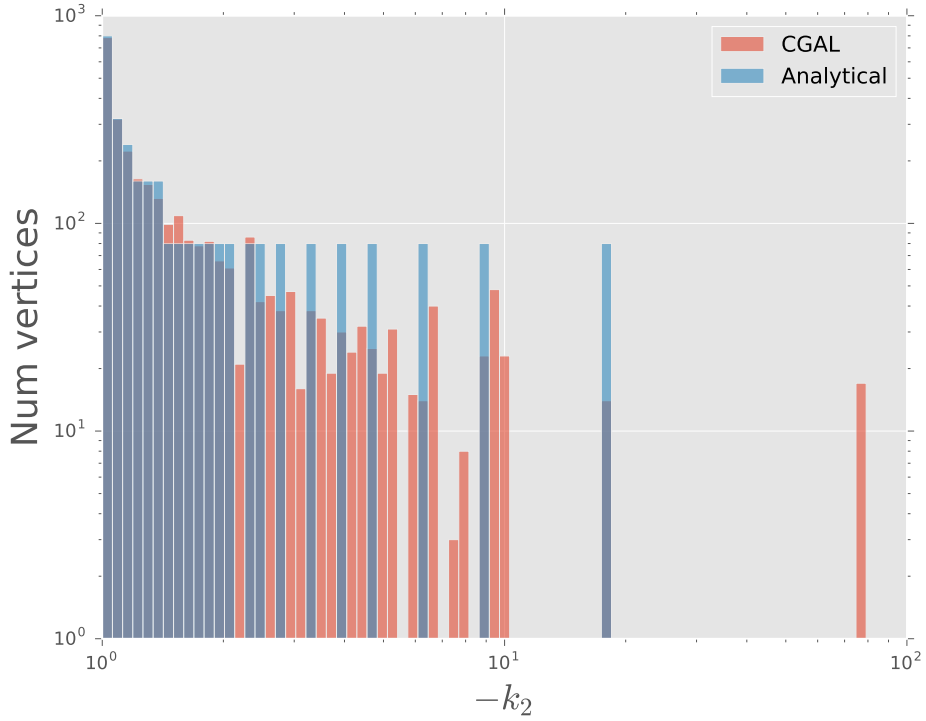


Figure 68: Histogram of second principal curvature at each vertex of the geometry comparing CGAL with the analytical result

geometry[1]. This situation is unfortunate, as the shock estimation starts from the nose of the vehicle, and therefore in a zone of high curvature, and the errors here will propagate along the estimation.

These differences will introduce an error in the shock estimation procedure. The error introduced by the curvature calculation can be quantified by comparing the shock estimations when using analytical curvatures against the fully automated method.

For a tangent ogive at a moderate angle of attack (10°), figure 69 shows a comparison for estimated shock using CGAL and analytical curvatures. While both shocks look qualitatively similar, the errors in curvature computation show in the shock estimation as noise in the shock position.

The next limitation is illustrated by a cone with high apex angle at zero angle of attack. A detailed study of this case can be found at Leyva[27]. Here, the

1 A finer surface mesh for the geometry would reduce these errors, concentrating them in a smaller and smaller high curvature zone around the ogive apex.

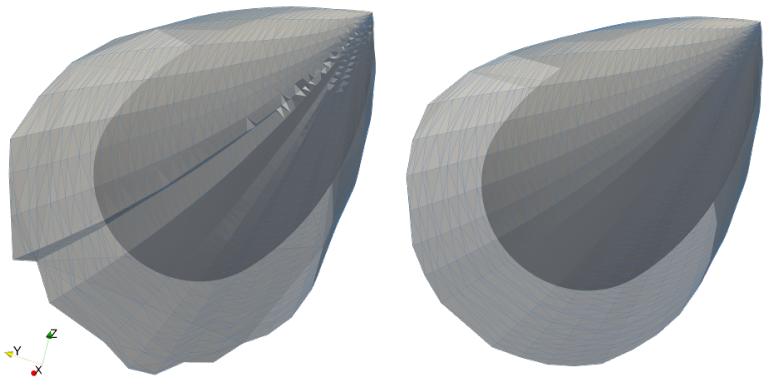


Figure 69: Estimated shock for a tangent ogive at 10° angle of attack. Curvatures computed by CGAL on the left and analytical curvatures on the right

shock is detached but the stagnation point is located at a singular point of the surface, therefore the principal curvatures there cannot be computed. As it is not possible to apply the starting procedure for detached shocks (generalized Billig's formula) on this geometry, the shock cannot be estimated. This can also be seen as a consequence of the global origin of the shock. The stand off distance and the shock curvature depends of the actual size of the cone geometry, and not on local information around the stagnation point. A similar situation is encountered with a cylinder aligned with the flow.

DISCUSSION

The present method has been shown to be able to address a wide range of geometries and Mach numbers in a unified manner. Especially relevant, due to its novelty, is the ability to predict shock position for three dimensional cases and the treatment of multi-shock situations. Despite its limitations, the present method produces predictions that are accurate enough to be useful in the design and analysis of hypersonic vehicles. This usefulness will be substantiated in the next chapters, dedicated to applications.

4 APPLICATIONS: IMPROVING LOW ORDER MODELS

The shock estimation method can be used as an addition to a low order aerothermodynamic model for supersonic and hypersonic vehicles. As such, its main application is the improvement of the predictions and increase of capabilities of that model when shock related phenomena is present.

The low order model that this shock estimation method was designed to complement is HyFlow[45]. This is an aerodynamic model optimized for its use in MDO of reusable space vehicles. The code is implemented in Python3, with some computationally expensive routines written in C++, and follows an object oriented programming paradigm. HyFlow was created in the Laboratory for Future Air-Space Transportation Technology (FASTT-Lab) at the University of Strathclyde.

There are a number of shock related situations that cannot be described by the surface inclination methods at the heart of HyFlow. These are opportunities to apply the shock estimation algorithm shown in the present work. The first one treated here is the estimation of off body flow conditions. The other is the effect of shock impingement, both from one body interacting with its own shock and from shock interference from proximal bodies.

FLOW FIELD INTERPOLATION

Local inclination methods lack any inherit capability for off-body flow predictions. This limitation is often not important during the design process, as the goal is the computation of surface values. Off-body conditions are however relevant in certain situations, like radiative flows or in proximal body interference. A fast prediction of these values could allow for the study of these situations in the context of low order modelling.

The estimation of the shock position provides a way to compute an approximated flow field around the vehicle. The procedure starts by computing the

shock position. Then, after-shock conditions are calculated for each facet of the estimated shock, using the oblique shock relations, and then interpolated into its vertices. HyFlow readily provides surface values using its internal methods. This results in a shock and geometry point cloud with associated pressure, temperature and velocity values.

To obtain an approximation of the flow field, this cloud is meshed using a Delaunay algorithm. This provides connectivity information to the cloud. Finally, point values are interpolated into each tetrahedron of the resulting Delaunay mesh, creating a volumetric field.

The resulting connectivity information is quite simplistic and mostly results in interpolations in the direction normal to the surface of the vehicle. Connectivity along Mach waves would be more theoretically sound but it is unclear how, if possible, this implementation could be carried out. Even with this limitation, however, the resulting flow fields are quite accurate (in the context of low order modelling).

The procedure is exemplified by a blunted tangent ogive at Mach 9. The estimated flow field is compared with a CFD simulation on the same domain. A qualitative comparison for pressure values is shown in figure 70.

The overall solution is quite satisfactory, with both fields showing similar structure and pressure values. T and U fields show similar behaviour.

A quantitative comparison (figure 73) shows mean relative error[1] for pressure between CFD and estimated flow field around 20%, with a 40% maximum value. Two zones are important to highlight. The first one is the subsonic zone around the nose, where the error is quite small (below 10%). The other is the high error line around the shock, due to the different position between estimated and actual shock.

SHOCK IMPINGEMENT

An important shock-related phenomenon is that of shock impingement. When a shock impinges on the surface of the vehicle, a zone of extremely high temperature and pressure is generated. This has two important effects. First, the thermal protection system has to be carefully designed for this situation, otherwise the

1 $\mathrm{abs}\left(\frac{\mathrm{CFD}-\mathrm{Estimation}}{\mathrm{CFD}}\right)$

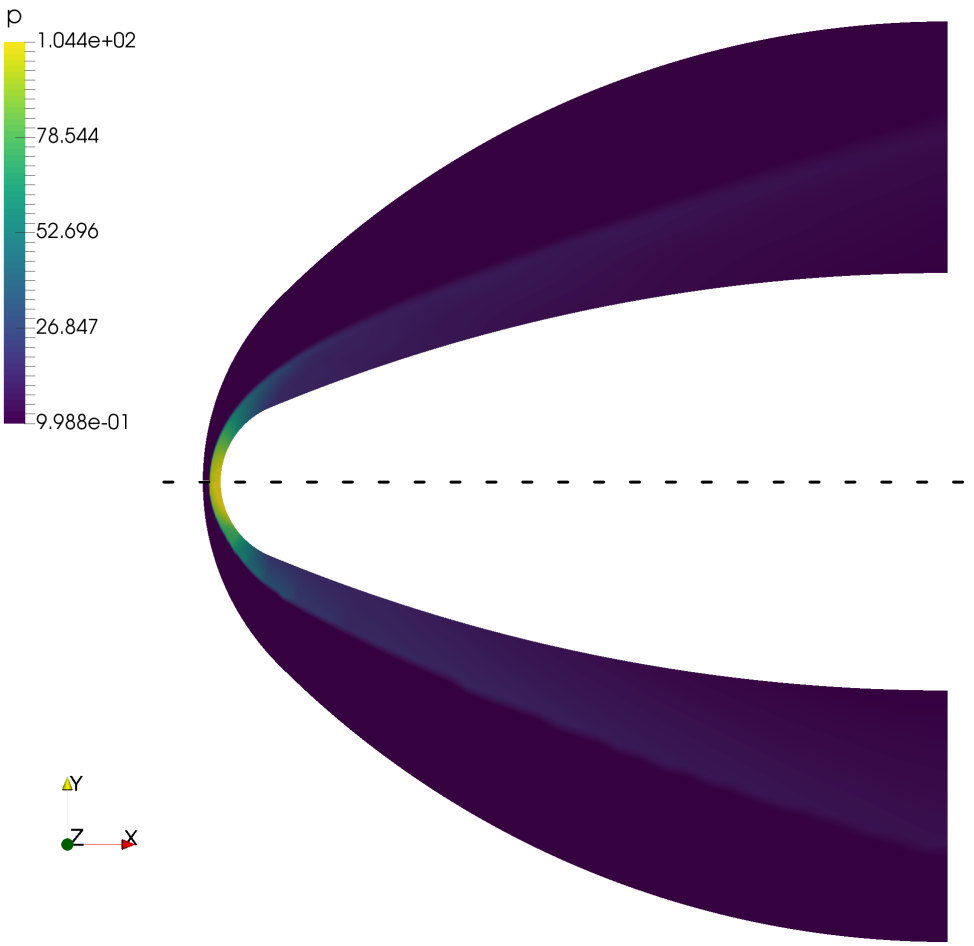


Figure 70: Comparison for pressure between CFD solution (top) and estimated flow field (bottom)

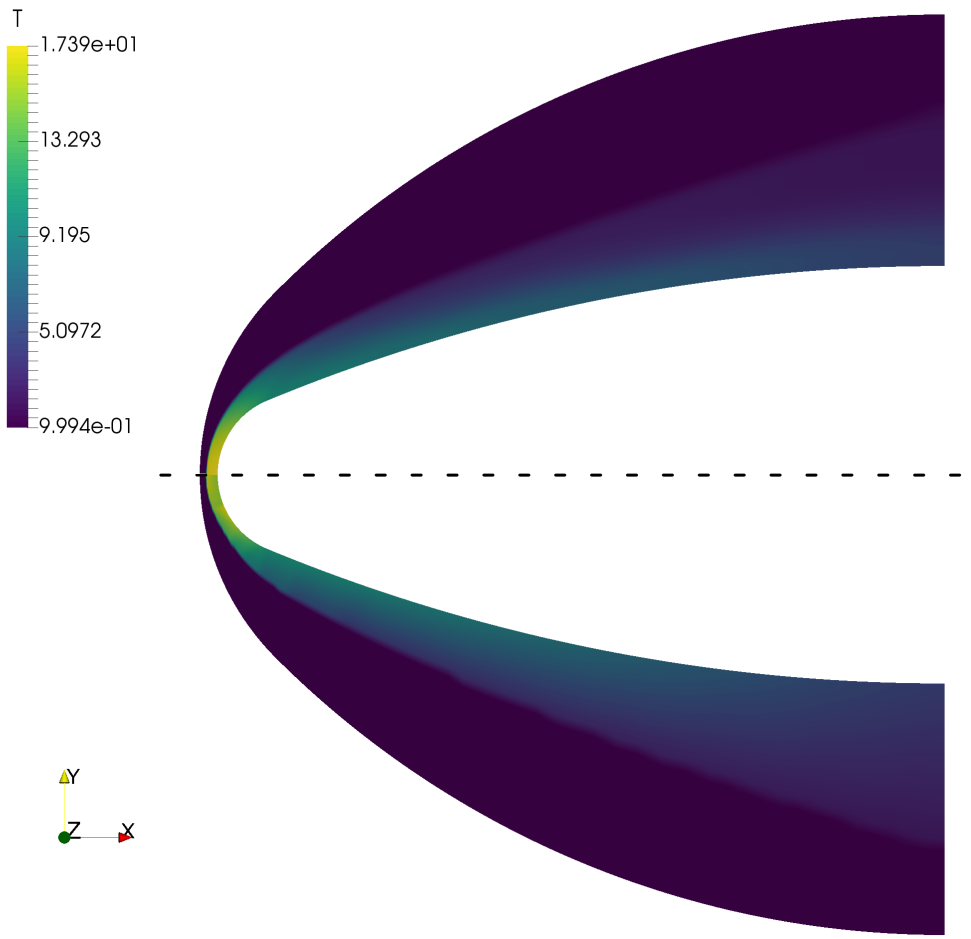


Figure 71: Comparison for temperature between CFD solution (top) and estimated flow field (bottom)

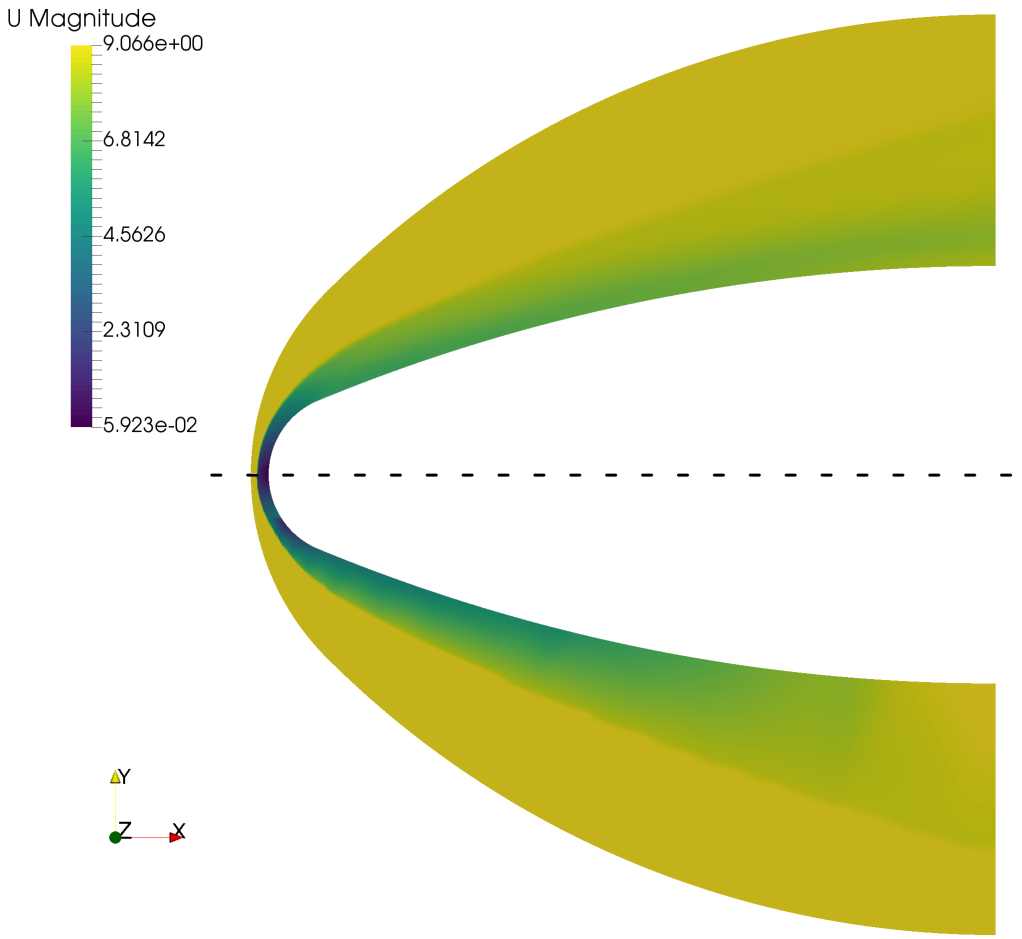


Figure 72: Comparison for velocity magnitude between CFD solution (top) and estimated flow field (bottom)

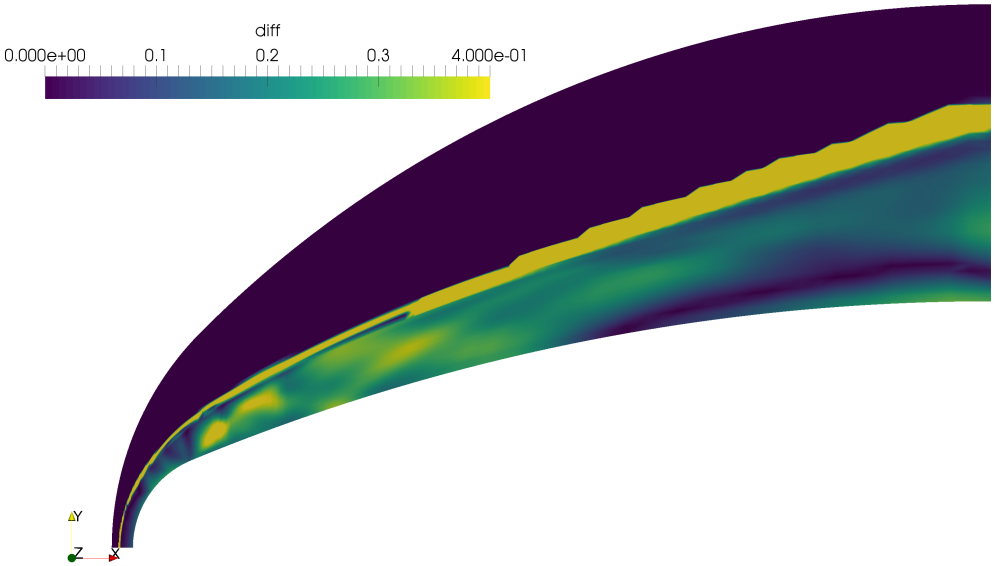


Figure 73: Relative error for pressure between CFD and estimated flow field

survivability of the vehicle could be compromised. Second, the peak pressure has to be accounted for when estimating the forces over the vehicle, as ignoring this effect could affect its controllability.

As both shock and geometry are described by triangular meshes, the problem of detecting impingement is equivalent to the problem of detecting intersections between these meshes.

Computing pairwise intersection for all triangles is computationally expensive. In order to speed up this process, axis aligned bounding boxes are created around each triangle of the geometry and the shock. These boxes are then tested against each other (a much easier operation than with triangles) and only if an intersection is detected here the underlying triangles are tested. This algorithm is implemented using the Intersecting Sequences of dD Iso-oriented Boxes[24] package of CGAL.

It is important to note that the shocks are considered transparent. That is, the shocks pass through each other as there is no interaction computed between them.

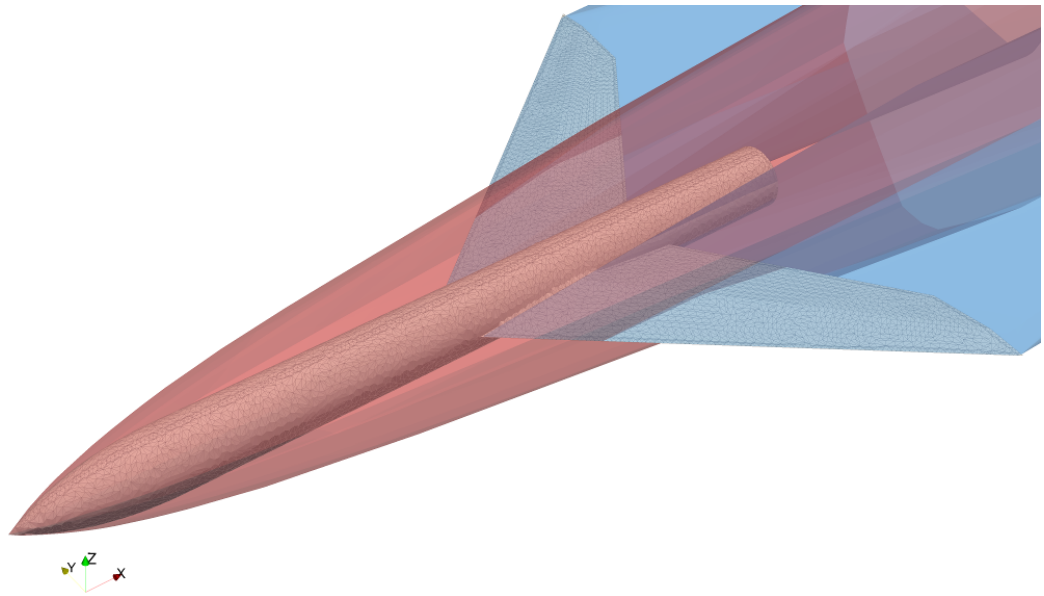


Figure 74: Shock structure around a aircraft-like vehicle

The procedure is illustrated with the synthetic example of an aircraft-like geometry. The shock structure obtained by the estimation method, for the vehicle at Mach 9 and zero angle of attack, is shown in figure 74.

From the figure, it can readily be seen that the shock generated at the nose of the vehicle impacts somewhere on the wing. The exact location of the impingement can be detected by the procedure described before.

Figure 75 shows the triangles of the aircraft-like geometry impinged by one of the shocks of the estimated shock structure. Two zones stand out, the impingement of the nose shock over the wing and the impingement of the wing shock over the fuselage. Along with this two zones, the attachment line on the wing leading edge and the attachment point at the nose of the vehicle can be seen. This is due to the fact that the stagnation points are part of the shock and lay on the geometry. These faces could be filtered out by not using the shock triangles that contain stagnation points in the impingement detection.

While this synthetic example might be ill designed for the flight conditions considered, shock impingement is a problem in real vehicles. One well studied case is that of Skylon. This vehicle is an SSTO design used Reaction Engines Ltd. to showcase their combined cycle air-breathing engine (SABRE).

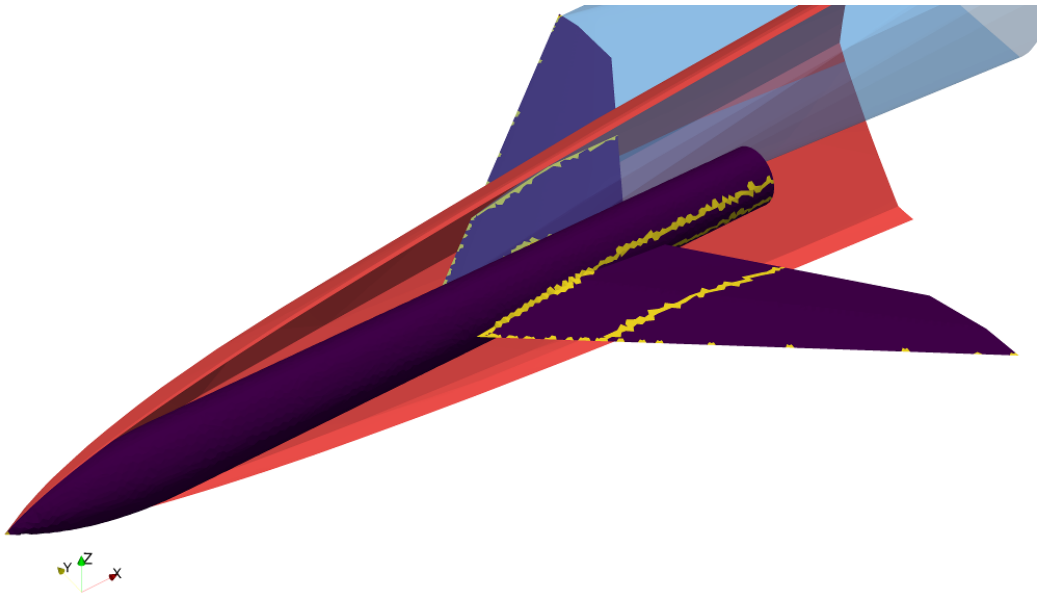


Figure 75: Impinged triangles of the aircraft-like vehicle

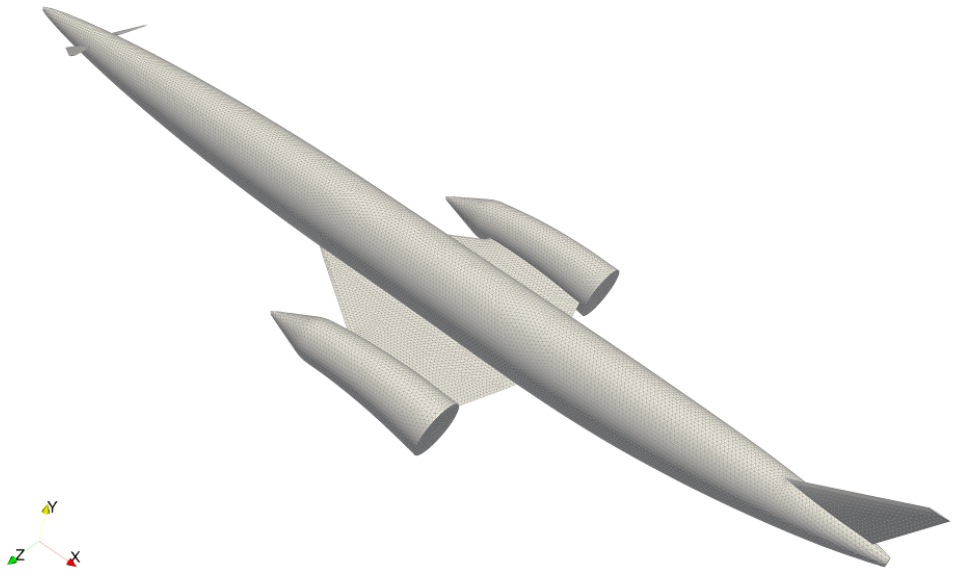


Figure 76: Skylon geometry

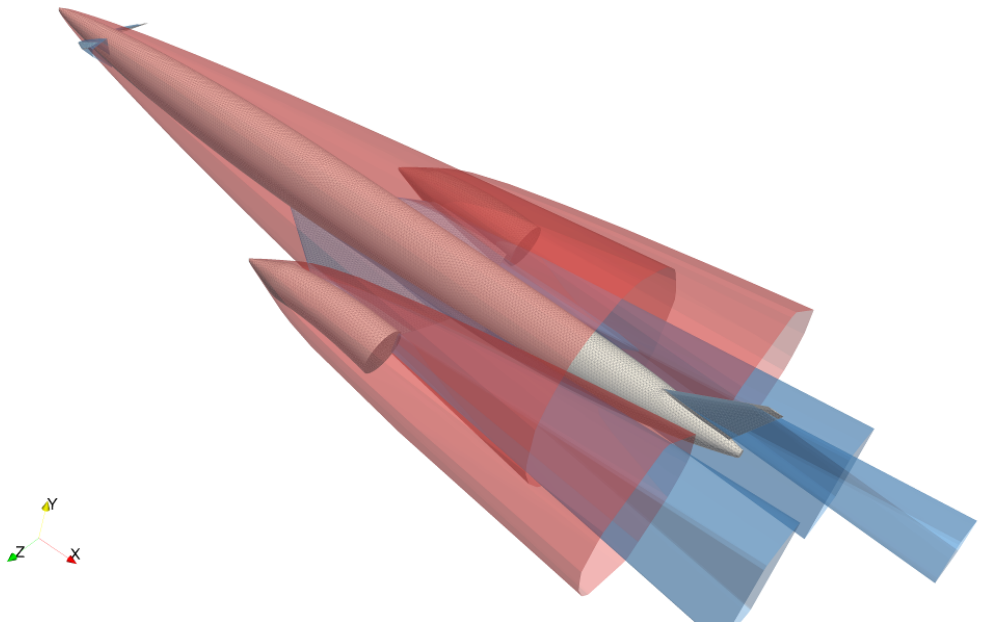


Figure 77: Shock structure around the Skylon vehicle

The shock structure in this case is much more complicated due to its multiple control surfaces and the presence of the engine nacelles. Figure 77 shows the estimated shock structure for the vehicle at Mach 9 and zero angle of attack.

Both the shock from the nose and the shock from the nacelles impact the wing. The detection procedure captures this situation, as shown in figure 78.

The nacelle-wing shock-shock interaction has been explored by Eggers[14]. This impingement might require the use of active cooling to protect the structure of the vehicle.

At this point, the present code only computes the impingement locations. An important improvement would be the inclusion of a model to estimate overpressures and increased heat transfer rates from the impingement detection. This addition would allow for studies such as the one of Eggers fully in the context of low order modelling.

Along with the high pressure peak at the impingement zone, shock impingement has an additional effect on the pressure distribution over the vehicle. The part of the vehicle enclosed downwind of a shock is immersed in a perturbed flow, with flow conditions different from those of the free stream. To accurately

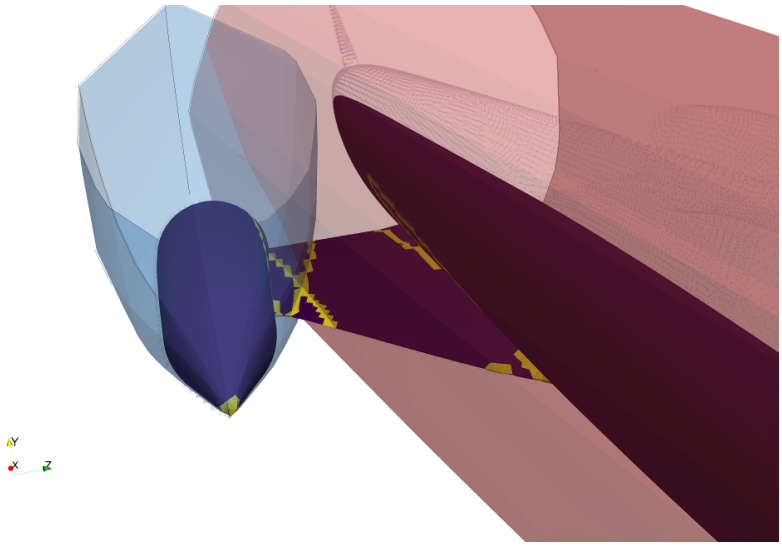


Figure 78: Impinged triangles of the Skylon vehicle

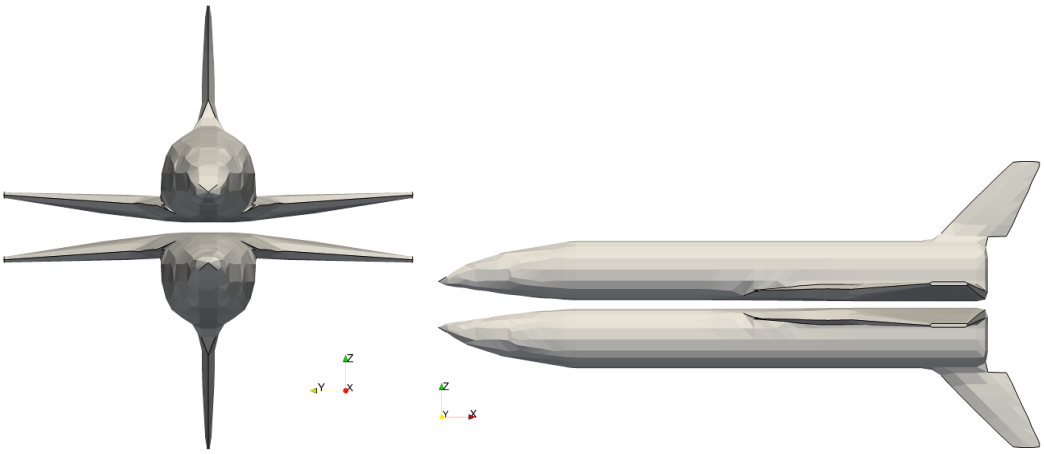
predict the forces over the vehicle using local inclination methods, this situation has to be taken into account.

The shock estimation procedure described in the present work can be coupled with HyFlow to estimate the forces over the vehicle in the presence of shock interference. Once a shock structure is estimated, the triangles of the geometry of the vehicle which fall inside each shock are computed. For those triangles, the flow conditions used to estimate pressures are the ones after the enclosing shock.

The shock is described as a triangular mesh. The flow conditions after it are computed using the oblique shock relations on each triangle. A single value for the whole shock is extracted by computing the area averaged supersonic conditions, disregarding the subsonic part in the case of a detached shock. This is a simplistic approach, but it produces reasonable results.

The detection of the enclosed triangles is done using the Polygon Mesh Processing package[29] of CGAL. As this algorithm requires a closed mesh against which to perform queries, the convex hull of the shock is computed and the vehicle is tested against it. The convex hull calculations are also helped by CGAL, using the 3D convex hull package[22].

An interesting case where this situation appears is in the presence of interference from proximal bodies, for example in the stage separation of a TSTO vehicle. This is a time dependant problem with a broad parameter space describ-



(a) Front view **(b)** Side view

Figure 79: TSTO configuration

ing all the possible separation strategies. A fast approximate method would be an interesting tool to study the problem.

The case shown in Bordelon et al[8] is analysed here. It consists of a TSTO configuration with an orbiter and a booster of identical shape mated in a belly-to-belly bimese arrangement. The shape of the stages is that of the Langley-Glide-Back Boster (LGBB), a generic wing-body vehicle developed for system analysis studies. Front and side views of this bimese configuration are shown in figures 79a and 79b.

Murphy et al[33] explored the possibility of using APAS to address this same separation problem. The APAS code, however, does not implement any shock estimation procedures, and therefore the shock structure has to be provided externally. With HyFlow and the present shock estimation method, the problem can be rapidly analysed using exclusively low order methods.

The surface pressure field of the orbiter is computed taking into account that a zone of it is immersed in the after shock conditions from the booster. From this pressure distribution, the forces and moments on the orbiter are computed.

For a separation purely in the z direction, figure 80 shows the aerodynamic force on the orbit in the z direction as a function of the separation. Figure 81 shows the same result but for the aerodynamic moment in the y direction.

While this is the same separation trajectory analysed by Murphy with APAS, it is impossible to perform a meaningful comparison with wind tunnel data due to the uncertainty in the geometry. The mesh for this test case has been recreated for

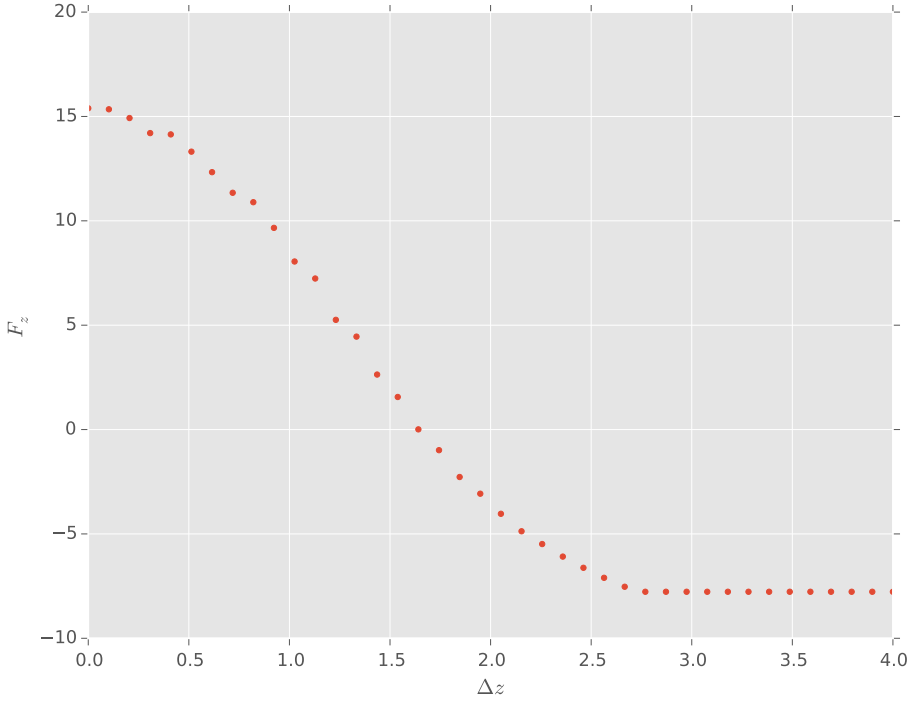


Figure 80: Aerodynamic force along the z axis as a function of the separation distance

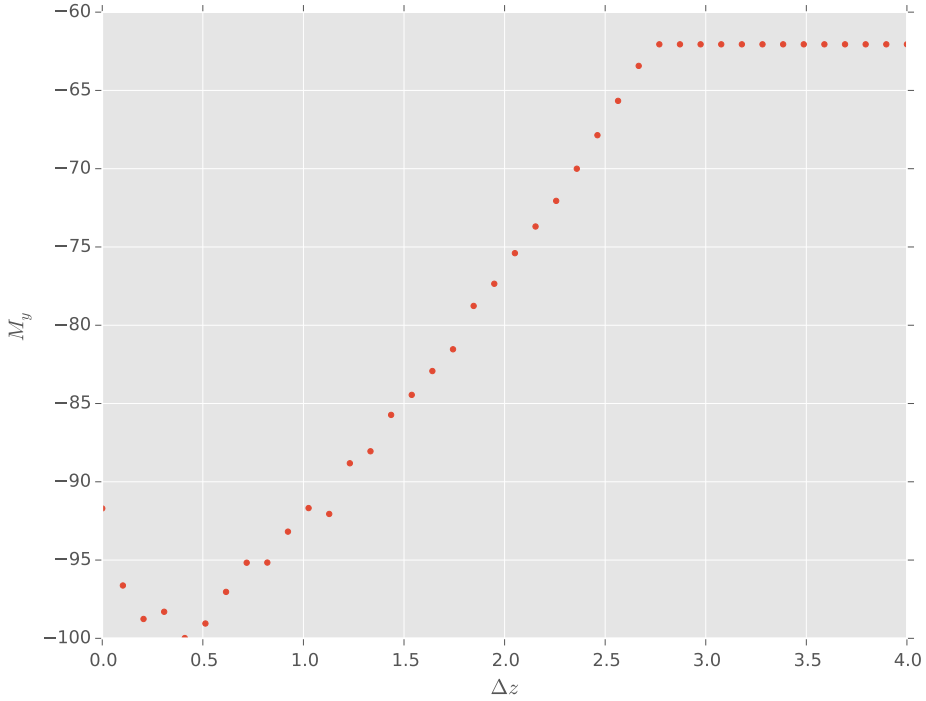


Figure 81: Aerodynamic moment along the y axis as a function of the separation distance

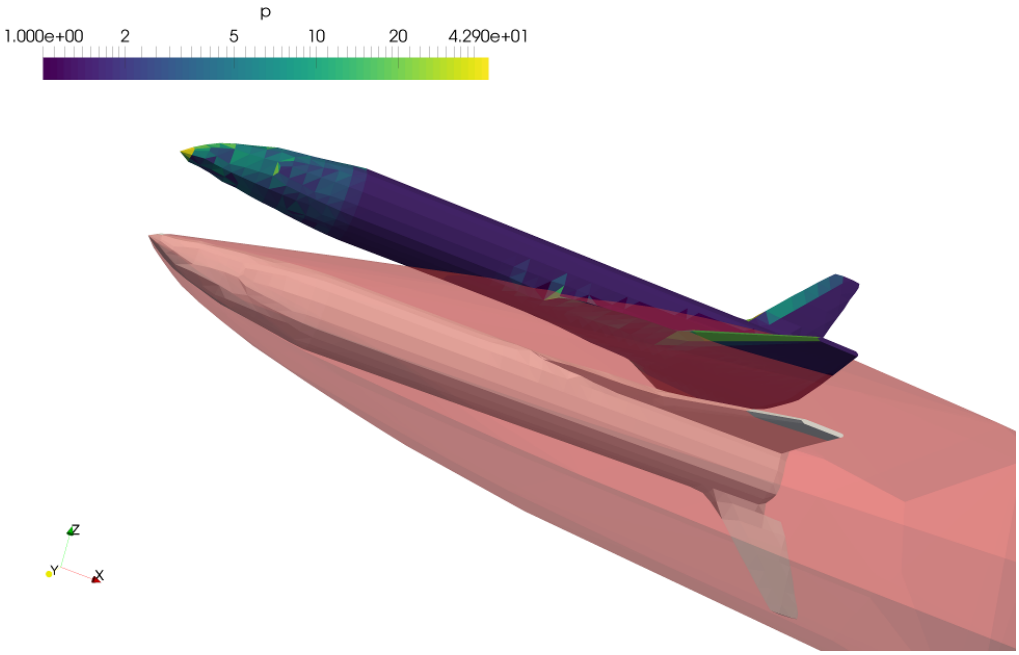


Figure 82: Shock interference from booster to orbiter and resulting pressure field

the present work using the figures of the paper as reference. While the overall shape is captured, there is not enough information to model all details of the geometry.

The speed of the presented method also allows for efficient parametric studies. Instead of analysing a single separation trajectory, it is possible to compute forces an moments for a whole matrix of orbiter positions. The analysis is similar to the one performed by Buning [9], although using low order methods instead of inviscid CFD.

Figure 83 shows the aerodynamic force on the orbiter along the z axis as a function of Δx and Δz. It can be seen that the orbiter leaves influence of the shock of the booster at $\Delta Z \simeq 3.0$.

Similarly, figure 84 shows the aerodynamic moment on the orbiter along the y axis as a function of Δx and Δz.

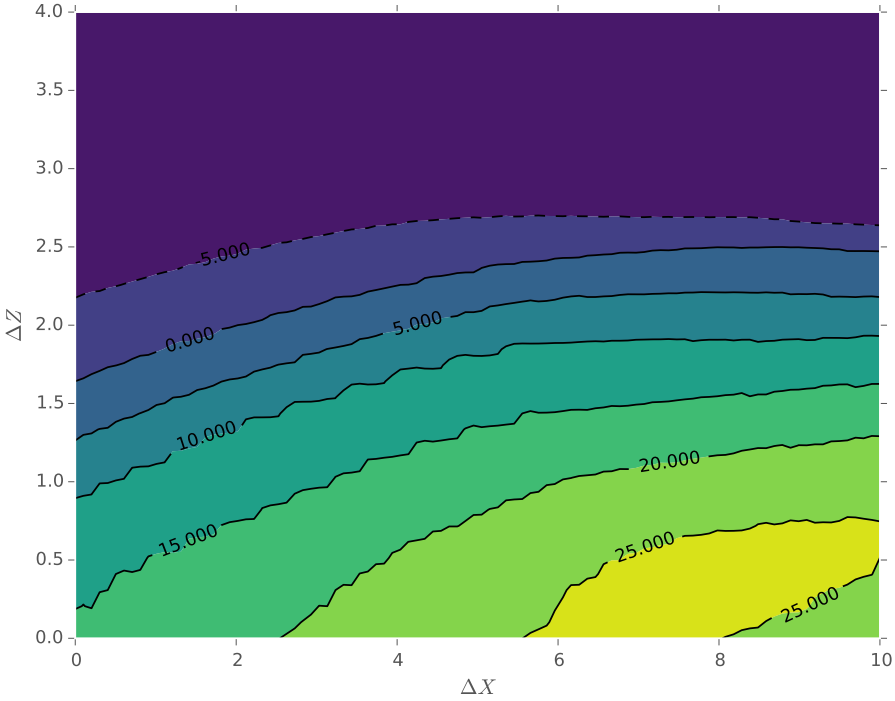


Figure 83: Aerodynamic force along the z axis as a function of Δx and Δz

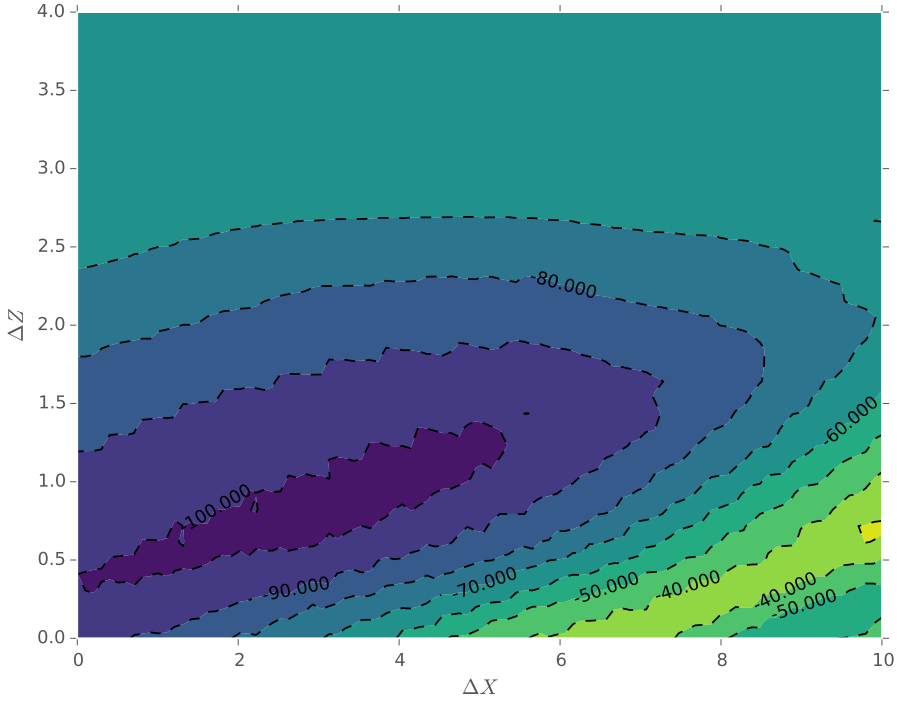


Figure 84: Aerodynamic moment along the y axis as a function of Δx and Δz

5 APPLICATIONS: INFORMING HIGH FIDELITY TOOLS

In the context of high fidelity tools, the method shows great promise in informing meshing for CFD in supersonic and hypersonic conditions. It can also be used to provide suitable initial conditions to CFD simulations.

CFD MESHING

CFD in supersonic flows has the particularity of having to deal with shocks. These present themselves as real discontinuities in flow properties in the case of Euler equations or strong gradients in zones of finite but very small thickness in the viscous case.

Two main families of methods have been used to deal with these discontinuities, namely shock-fitting and shock-capturing techniques. Shock-fitting explicitly treats the shock as an actual discontinuity, tracking it separately from the underlying mesh and imposing the jump of flow properties according to Rankine-Hugoniot equations as boundary conditions for the rest of the solution. These methods are algorithmically complex to implement and its use is nowadays limited. The other family is that of shock-capturing techniques. Here, the shock is not explicitly treated, arising as part of the solution. Numerical dissipation is introduced to limit high frequency oscillations in the shock front. This is the most commonly used method and is the one implemented in OpenFOAM, the solver selected for the present work. A more detailed overview of these methods can be found in Pirozzoli[36].

As the shock is not explicitly treated in shock-capturing codes, the suitability of the mesh with respect to the solution is paramount. Mesh refinement and alignment around shock surfaces is required in order to capture sharp shocks and avoid oscillations. The position of the shocks, however, is only known after the simulation is completed. It is therefore necessary to perform a preliminary simulation to guide meshing for a second more detailed one or to use adap-

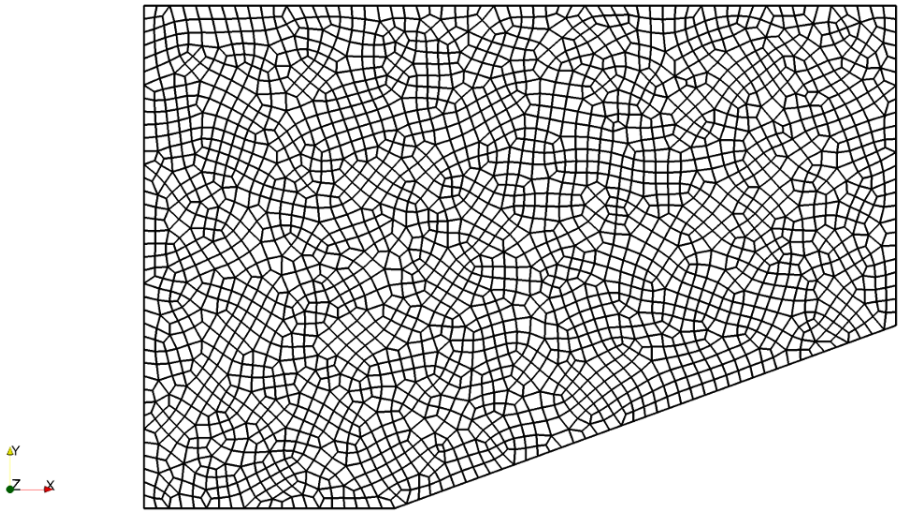


Figure 85: Uniform mesh

tive mesh refinement (AMR) to compute the solution and optimized mesh in an coupled manner.

The shock estimation procedure presented here can, however, potentially provide the shock structure directly from the geometry and the flow conditions. That would allow for adapted meshing without the need of preliminary simulations to precompute the shock structure or the use of AMR.

A two dimensional supersonic wedge will be used as a proof of concept for the procedure. The wedge has a 20 degree semi-angle and it is immersed in a Mach 5 free stream. The solution field for this problem is well known. It corresponds to an oblique shock with 29.8 degree inclination [1]. The meshing has been performed using gmsh[17]. Two unstructured meshes, with similar number of cells have been produced. The first one has uniform sizing, while the second one has the cell size linked to the distance to the shock (that, in this case, is known before the simulation). The meshes are quad dominant, produced by recombining a Delaunay triangular mesh. OpenFOAM is cell centred, it is therefore the cell number what sets the degrees of freedom. The cell counts of the uniform and non-uniform meshes are 2522 and 2476 respectively.

Figure 87 shows converged the solution for Mach number on the uniform mesh, while figure 88 shows the same field for the non-uniform one.

1 As computed by:
$\tan\theta = 2\cot\beta \frac{M_1^2 \sin^2\beta - 1}{M_1^2(\gamma + \cos 2\beta) + 2}$

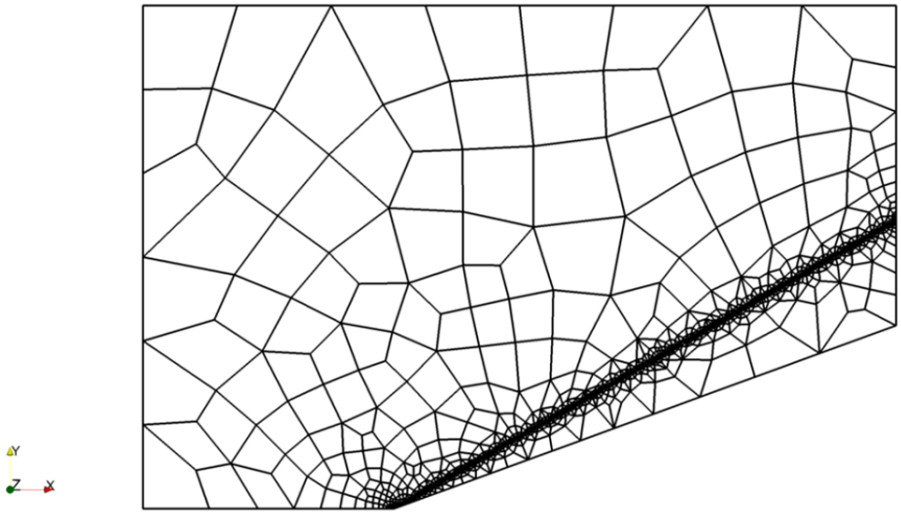


Figure 86: Non-uniform mesh

From a qualitatively point of view, the solution for the mesh that takes into account the shock position looks markedly better. This is apparent in this case due to the a priori knowledge of the correct solution. In general, the solution will not be known before the simulation, so a different strategy is needed to compare the quality of the meshes.

This problem is similar to that present in adaptive mesh refinement (AMR). Here, an estimation of the local error is used in order to evaluate the necessity of local refinement or coarsening. The mesh is then optimized towards an uniform distribution of the error, coarsening the zones of low error while refining the high error zones.

The error evaluated with these estimators is that of interpolation of a particular physical property. It arises due to the representation of the continuous physical solution onto the discrete mesh and is a function of the position and the direction. Crucially, it can be seen as a metric tensor of a Riemann geometry. An isotropic mesh in this geometry would have uniform error, while being anisotropic in Euclidean space. It has been shown that this metric tensor is the Hessian matrix of some flow variable [13]. In these supersonic simulations, the Mach number has been selected as the relevant fluid property from which to compute the Hessian.

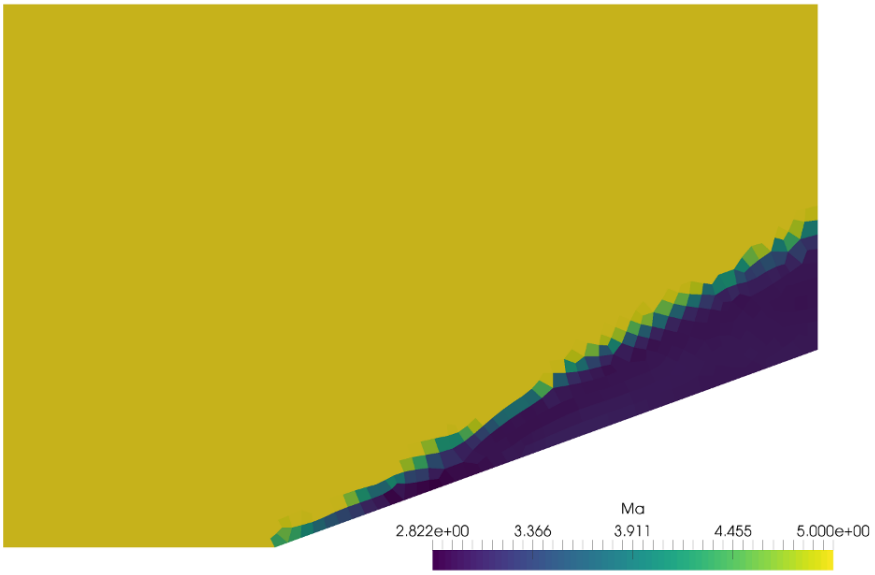


Figure 87: Mach number field for the uniform mesh

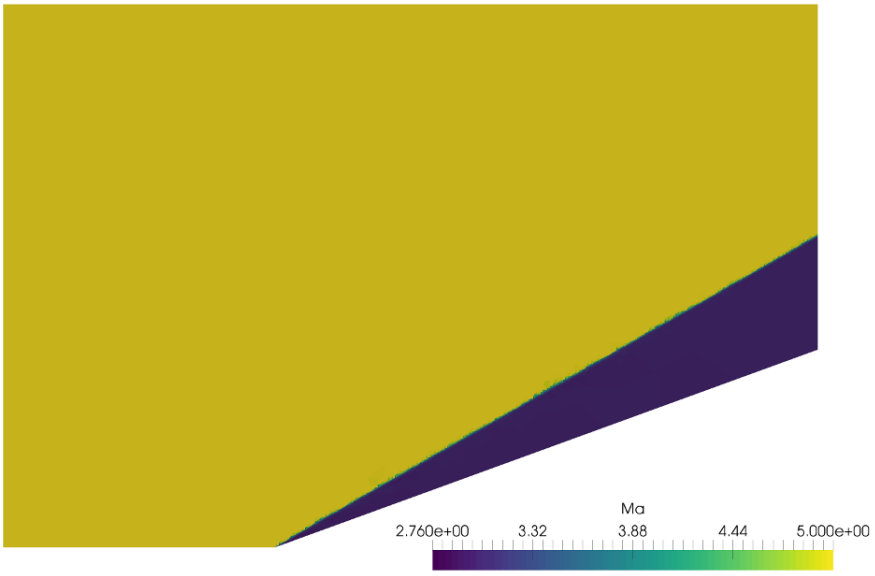


Figure 88: Mach number field for the non-uniform mesh

When applied to AMR[10], the metric tensor is first computed as

$$M = R \begin{pmatrix} |\lambda_1| & 0 & 0 \\ 0 & |\lambda_2| & 0 \\ 0 & 0 & |\lambda_3| \end{pmatrix} R^{-1} \tag{30}$$

where λ_i and R are the eigenvalues and a matrix composed from the eigenvectors of the Hessian matrix respectively. The lengths of segments in this metric are then calculated as

$$l = \frac{2}{3} \frac{l_0^2 + l_0 l_1 + l_1^2}{l_0 + l_1} \tag{31}$$

with $l_i = \sqrt{(\gamma')^t M \gamma'}$, where γ' is the derivative of the parametrization of the segment. If the segment is parametrized with s as $x = x_0 + s(x_1 - x_0)$, $\gamma' = x_1 - x_0$ is readily obtained.

Solution and mesh adaptation are calculated together in an iterative manner. Once the solution is computed on the initial mesh, the metric tensor is calculated from that solution using a relevant physical property. In this defined metric, the lengths of the segments of the mesh would be uniformed by refinement and coarsening, equidistributing the error. In a vertex centred code, these segments would be the edges of the triangulation, while in a cell centred one, this segments would be edges of the dual mesh. The solution is then interpolated into this new mesh and the computation continues.

The use of the a posteriori error estimator is somewhat similar in the context of mesh comparison. Here, two meshes and two solution fields are available. The goal is to compare the error distribution of the two solutions on the two meshes. Four mesh-solution pairs are created, namely (mesh a - solution a), (mesh b - interpolated solution a), (mesh a - interpolated solution b) and finally (mesh b - solution b). In the case presented here, one mesh would be the one with uniform sizing while the other would be the shock refined one.

A metric tensor is derived from each solution. Then, the lengths of the inter-cell segments (edges of the dual mesh) on each mesh are computed in this metric and stored on cell faces. Finally, cell values are computed by averaging cell face values. The error distribution is quantified by the standard deviation of these cell errors.

For a given solution, a smaller standard deviation is expected for the shock refined mesh with respect to the one with uniform sizing. And this is indeed the

Table 1: Standard deviations of error distribution

	mesh A	mesh B
solution A	0.4435	0.3418
solution B	1.3832	0.6281

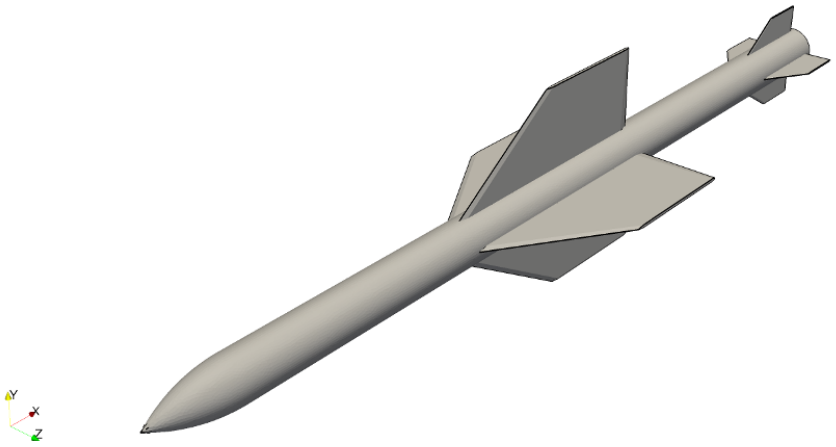


Figure 89: Missile geometry

case, as shown in the table (mesh A is the uniformly sized mesh and solution A is its resulting field).

The error is more uniformly distributed (smaller standard deviation) on the shock aware mesh (B).

In order to compare the meshes on the same metric, interpolating the solution fields between meshes is required. Therefore an additional interpolation error is introduced. Comparing both meshes on both metrics, which requires A-¿B interpolation and the reciprocal B-¿A, mitigates this effect.

With the concept demonstrated, the procedure will be applied to relevant engineering cases. The first one is a missile flying at Mach 6 and zero angle of attack. The missile geometry shown in figure 89.

The shock structure is estimated and then used to guide the meshing for the CFD simulation. Gmsh has again been the tool selected to create the meshes. They have been generated as non structured tetrahedral and then dualized into polyhedral meshes to reduce skewness in OpenFOAM cell centered code.

Figure 90 shows the CFD domain and the estimated shock structure.

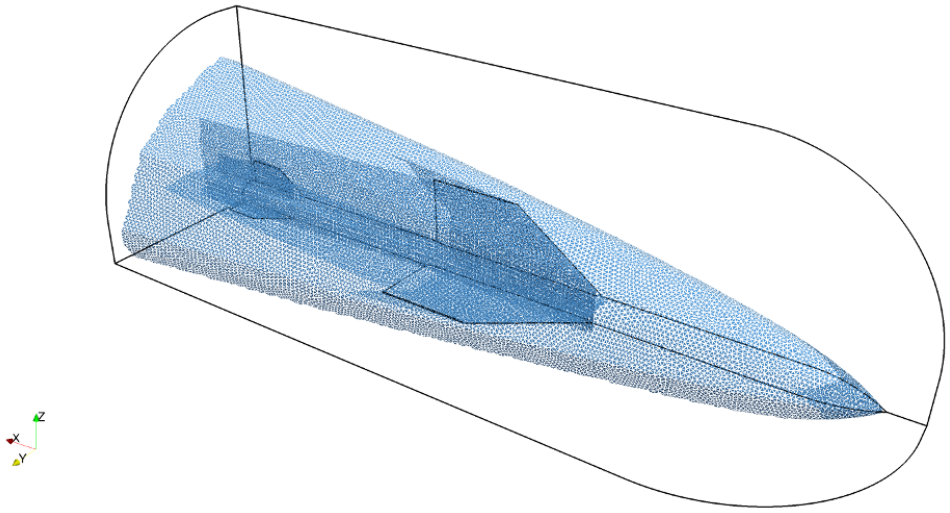


Figure 90: CFD domain and estimated shock

Figure 91 shows the non uniform mesh of 1505938 cells. The cell sizing takes into account the distance to the shock structure as well as the distance from the nose of the missile. This is done to address the increasing error of the shock estimation in the stream wise direction.

Figure 92 shows the uniformly sized mesh of 1503465 cells.

Overall, the converged the solutions for Mach number on both meshes look markedly similar, with the non-uniform one showing slightly better shock definition. It is in the details, however, where the adapted solution shows its potential. Zones like the nose, the wing-body shock interaction or the domain exit plane are much better resolved in this adapted mesh. Figures 93 and 94 show the solution for Mach number on the whole domain. Figures 95 to 97 show a side by side comparison of certain zones of the simulations.

The shock around the nose, shown in figure 95, is much better defined in the non-uniform case. This is important, as differences here will propagate downstream affecting the shock structure.

Figure 96 shows the shock interaction at the wing root. On this figure, the color map has been squeezed to highlight this interaction. Care must be taken in this zone during design, as the impinging shock may generate pressure and heat transfer peaks over the body. While both meshes show this interaction, the result looks smeared on the uniform mesh as compared with the shock aware one.

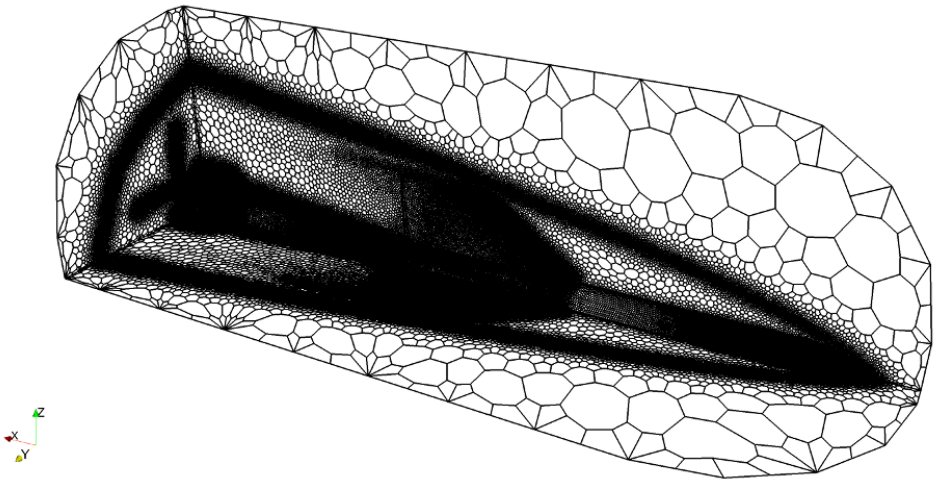


Figure 91: Non-uniform mesh

Figure 92: Uniform mesh

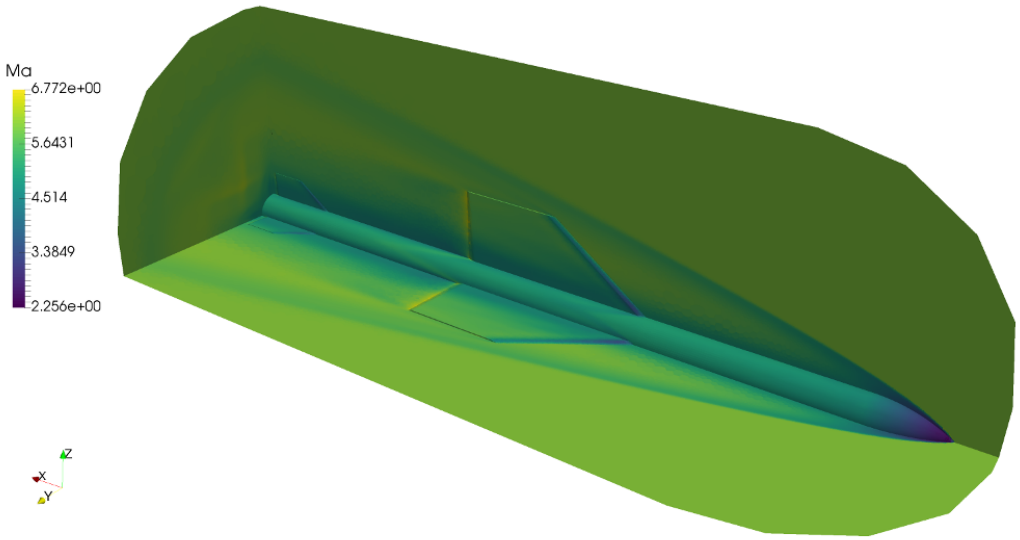


Figure 93: Mach number field for the non-uniform mesh

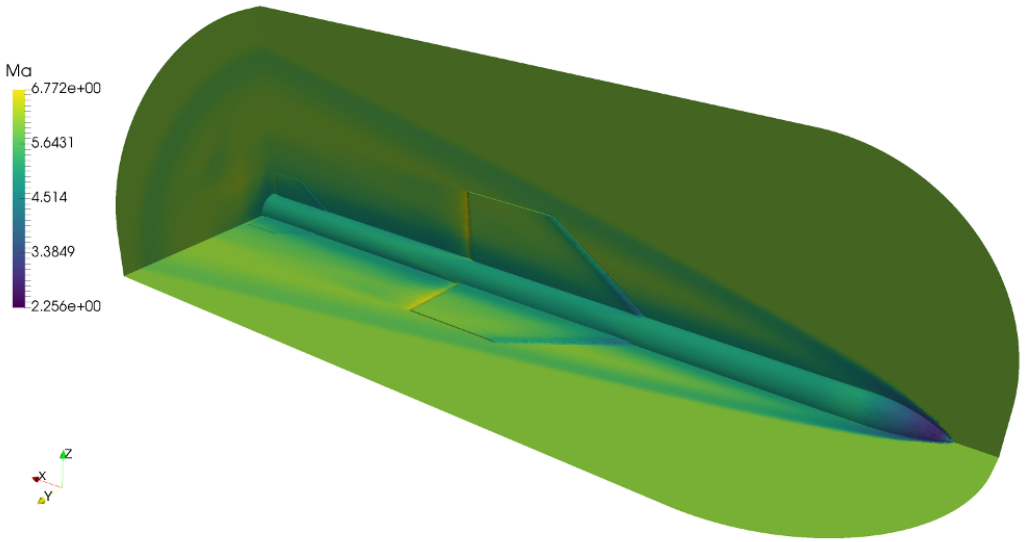


Figure 94: Mach number field for the uniform mesh

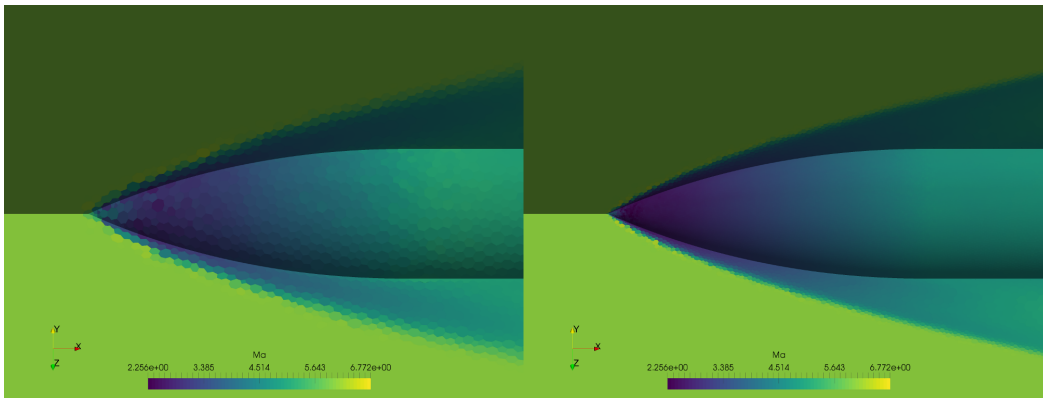


Figure 95: Comparison of the nose zone on uniform (left) and non-uniform (right) meshes

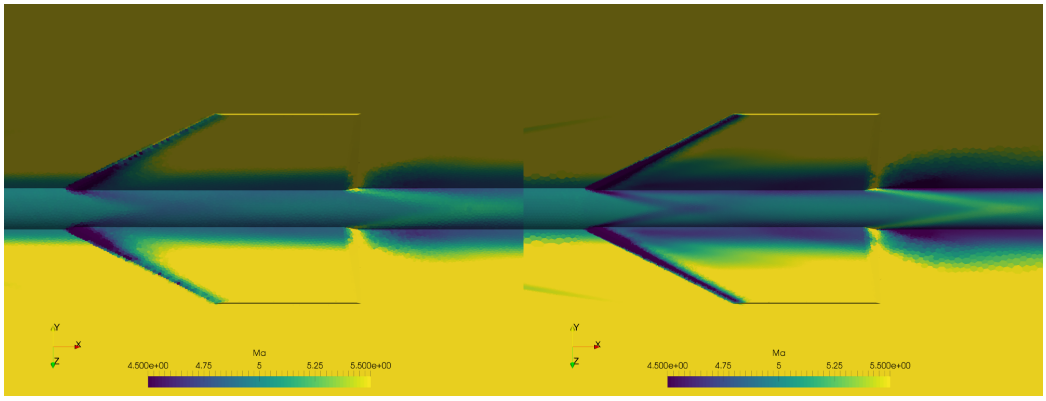


Figure 96: Comparison of the wing-body shock interaction on uniform (left) and non-uniform (right) meshes

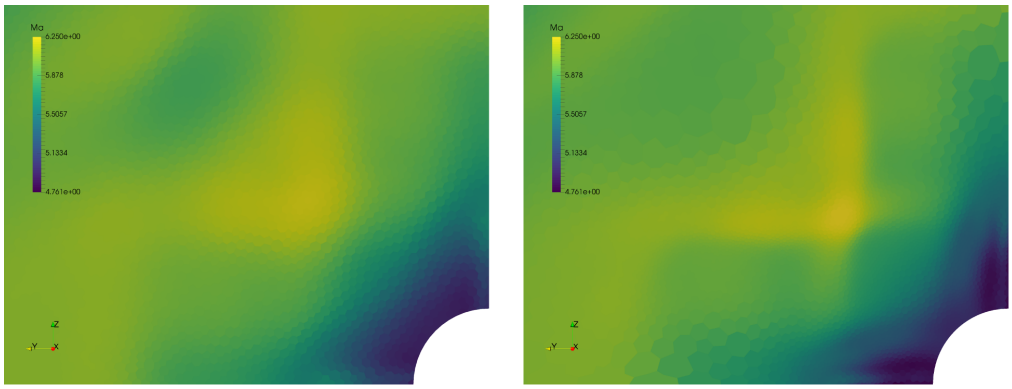


Figure 97: Comparison of domain exit plane on uniform (left) and non-uniform (right) meshes

Table 2: Standard deviations of error distribution

	mesh A	mesh B
solution A	0.7067	0.6383
solution B	0.8988	0.8553

Finally, figure 97 shows the exit plane of the CFD domain. The interacting wing shock structure as well as the shock and wake structure from the tail fins can be seen in the plot. Again, while overall similar, the smeared result on the uniform mesh is quite noticeable.

As with the supersonic wedge case, the two meshes are quantitatively compared by measuring the standard deviation of the error distribution. Table 2 shows the standard deviations, where A refers to the uniformly sized mesh and B to the non-uniform one.

The shock aware mesh (B) shows a more uniform error distribution, that translates into a mesh closer to optimal. This can also be seen by plotting the histogram of the error for each solution on both meshes. Figure 98 shows this comparison for solution A, while solution B is shown in 99

The X43 vehicle has been selected as the second engineering test case. This is a hypersonic experimental craft developed by NASA for in-flight testing of scramjet propulsion. An overview of the program can be found in Voland et al.[43]

The vehicle has been simulated at Mach 6 in open cowl configuration. The geometry is shown in figure 100, while the CFD domain and the estimated shocks are shown in figure 101.

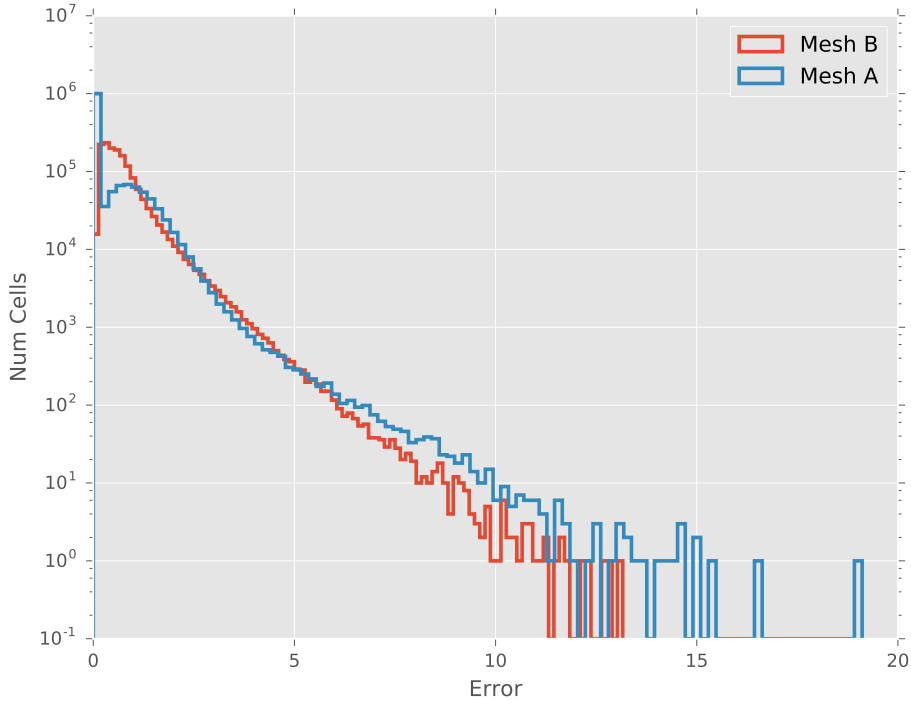


Figure 98: Histograms of cell error for solution A. Mesh B is shown in red with mesh A appearing in blue

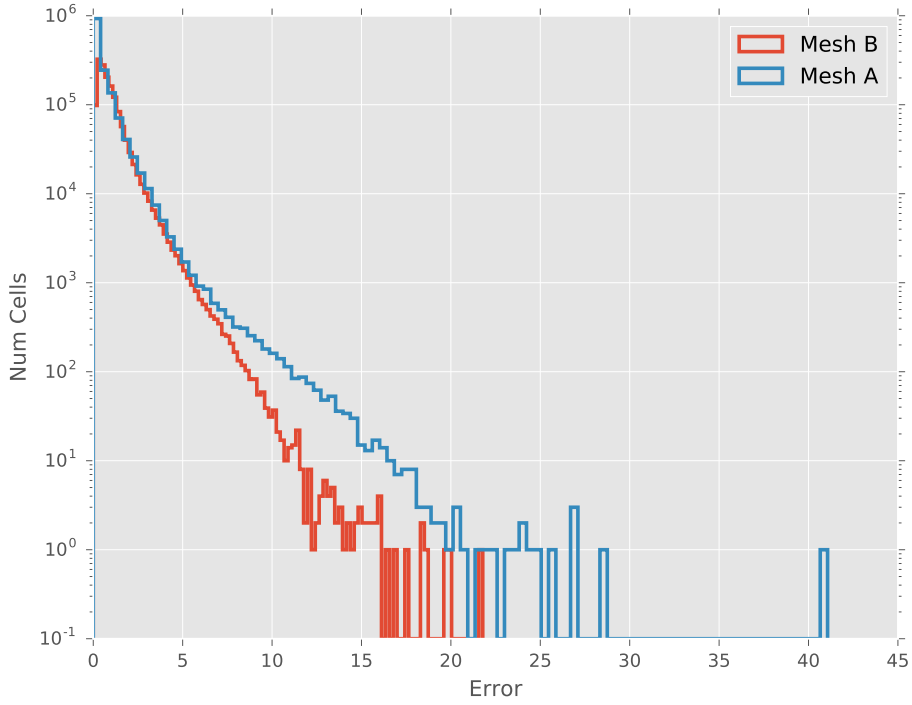


Figure 99: Histograms of cell error for solution B. Mesh B is shown in red with mesh A appearing in blue

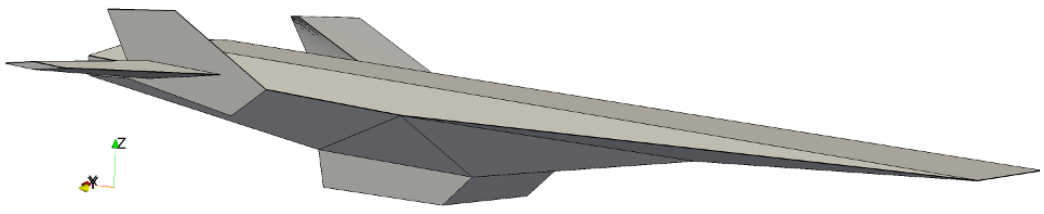


Figure 100: X43 geometry

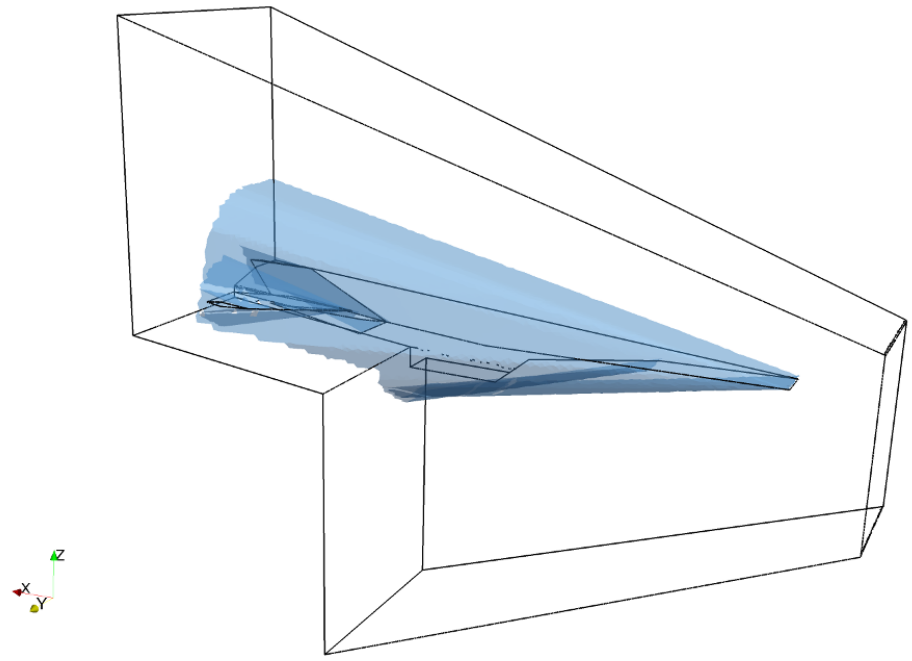


Figure 101: CFD domain and estimated shock

Again, two meshes have been generated, a uniformly sized and a non-uniform, shock aware one. The uniform mesh has 1522862 cells and is shown in figure 102. Figure 103 shows the non-uniform mesh, with a 1499886 cell count.

From a qualitative point of view, the Mach number field obtained from the converged solution on the non-uniform mesh (figure 104) shows sharper shock fronts as compared with the uniform mesh solution (figure 105). Otherwise the results are very similar, as expected from two converged simulations of the same test case.

Once again, it is in the details where the adapted mesh proves most useful. This is explored by comparing side by side, the shock structure in front of the ramjet, the fin-body shock interaction and the domain exit plane on both meshes.

The ramjet inlet conditions have an important effect in the performances of the propulsive system. The X43 vehicle has two compression ramps that condition this inlet flow. The first ramp produces a very weak shock, followed by a stronger one produced over the second ramp. Figure 106 shows a side by side comparison of this shock structure on both meshes. The uniform mesh shows a very diffuse,

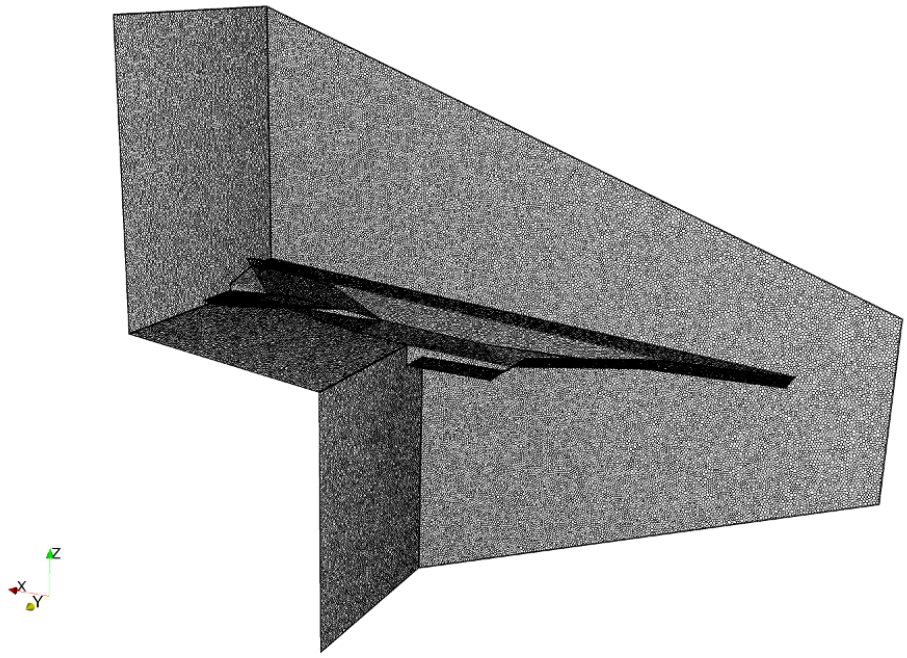


Figure 102: Uniform mesh

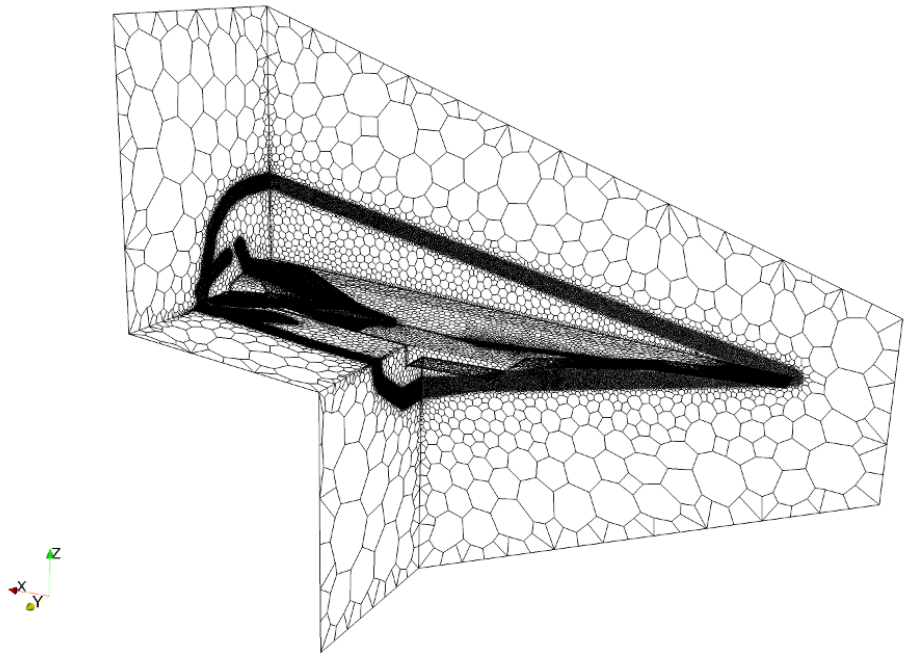


Figure 103: Non-uniform mesh

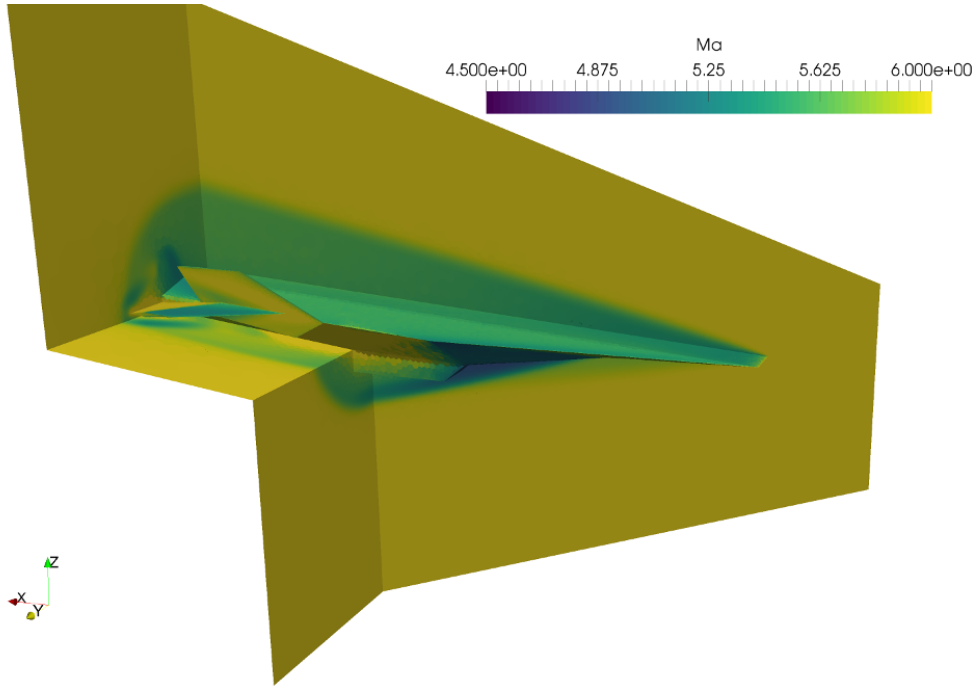


Figure 104: Mach number field for the non-uniform mesh

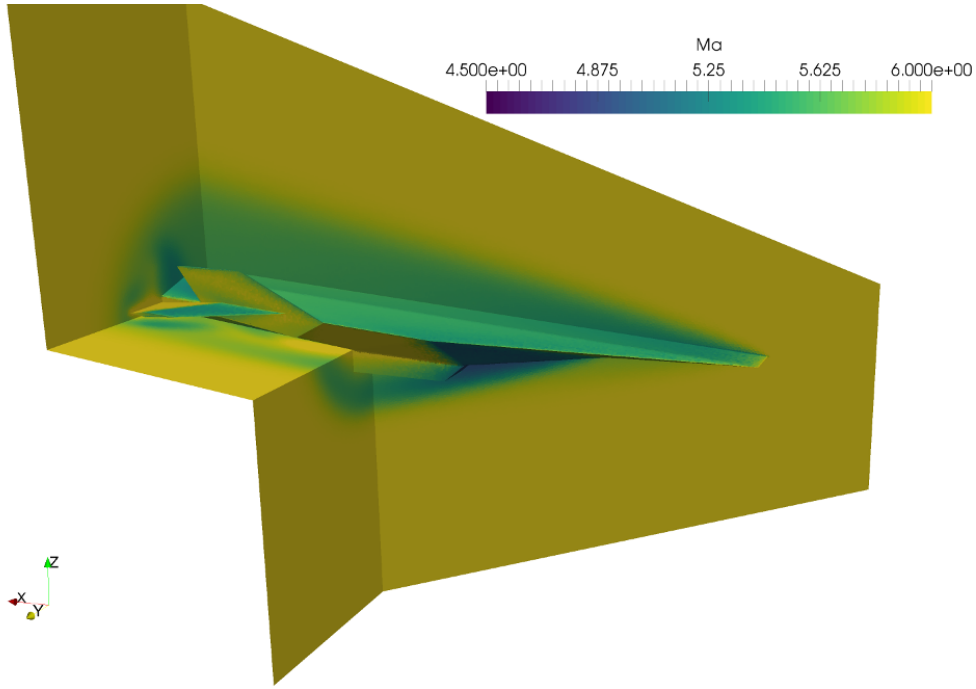


Figure 105: Mach number field for the uniform mesh

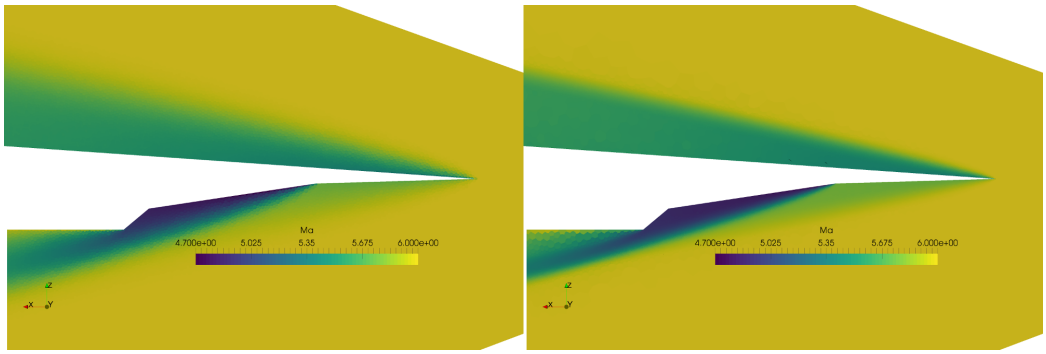


Figure 106: Comparison of plane of symmetry on uniform (left) and non-uniform (right) meshes

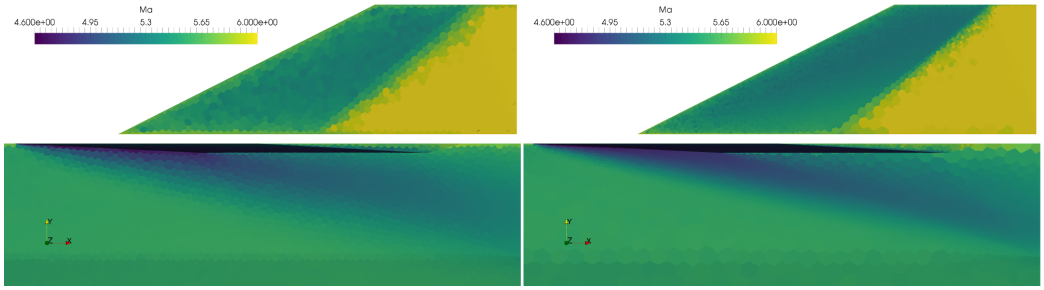


Figure 107: Comparison of the fin-body shock interaction (top side) on uniform (left) and non-uniform (right) meshes

almost non existent, first shock in contrast with the more defined front of the refined mesh. This, in turn, affects the second shock and the inlet conditions.

Another zone of interest lies in the interaction between the vertical fin shock and the body. This interaction is present in both the top and bottom sides of the geometry. The footprint of the shock impingement is much better resolved in the non-uniform mesh, which makes its results more useful than the ones provided by the uniform mesh when studying possible thermal issues.

Finally, a comparison of domain exit plane on both simulations is shown (figure 109). This presents a good overview of the effect of the non-uniformity of the mesh on the solution. The fin wakes and the shock envelope are enhanced by this meshing.

A quantitative comparison of the results is also performed, following the same procedure as in the previous test case. The standard deviations of error distribution for both solutions on both meshes is shown in table 3. Again, A refers to the uniformly sized mesh and B to the non-uniform one.

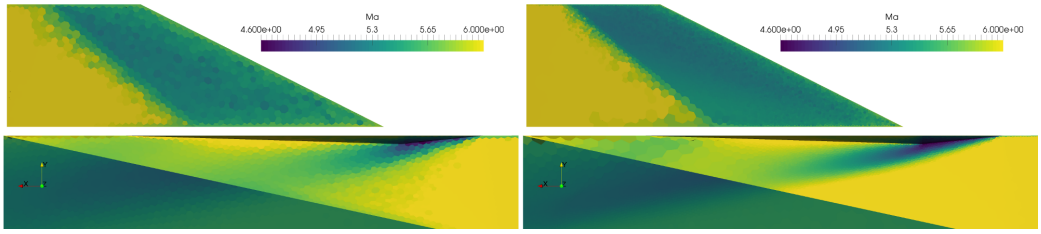


Figure 108: Comparison of the fin-body shock interaction (bottom side) on uniform (left) and non-uniform (right) meshes

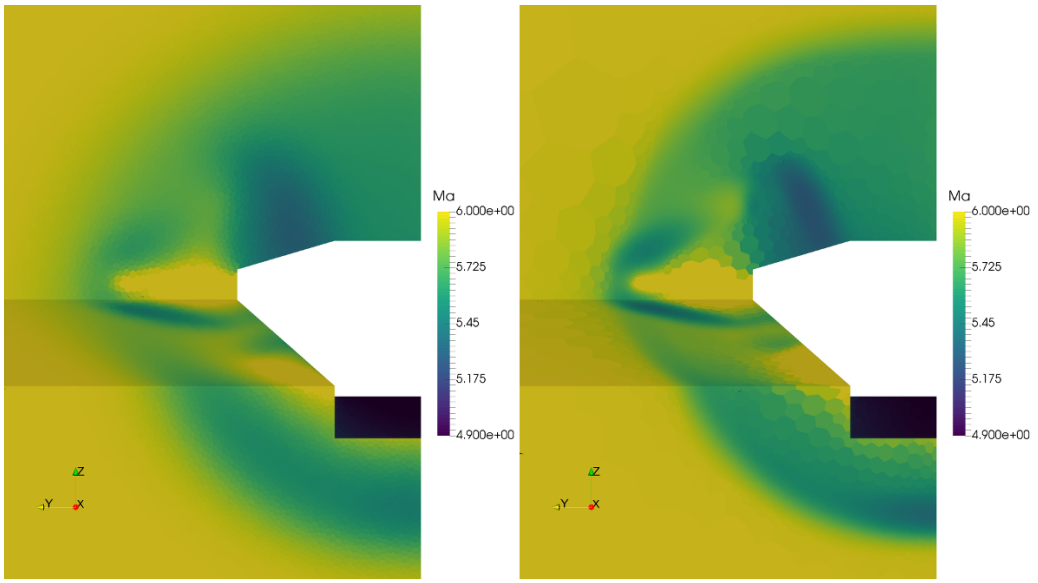


Figure 109: Comparison of domain exit plane on uniform (left) and non-uniform (right) meshes

Table 3: Standard deviations of error distribution

	mesh A	mesh B
solution A	0.4148	0.3919
solution B	0.5946	0.5023

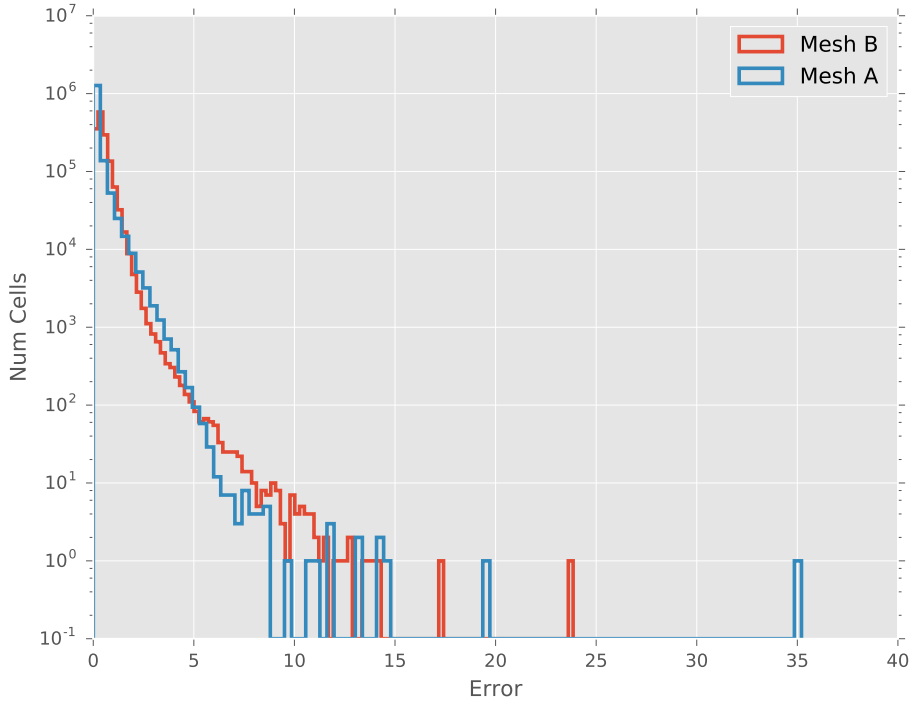


Figure 110: Histograms of cell error for solution A. Mesh B is shown in red with mesh A appearing in blue

The results here are closer than before, but the shock aware mesh (B) still shows better optimality (through a more uniform error distribution). The error histograms (figure 110 for solution A and figure 111 for solution B) indicate a similar error distribution on solution A, with a more differentiated behaviour on solution B. This can be attributed to the very low shock definition on solution A, which partially defeats the advantage of a finer mesh around the shock.

FLOW FIELD INTERPOLATION. GENERATING INITIAL CONDITIONS

The procedure for flow field estimation showed in the previous chapter in the context of low order modelling can also be applied to the generation of initial conditions for CFD simulations. The application is illustrated with the same

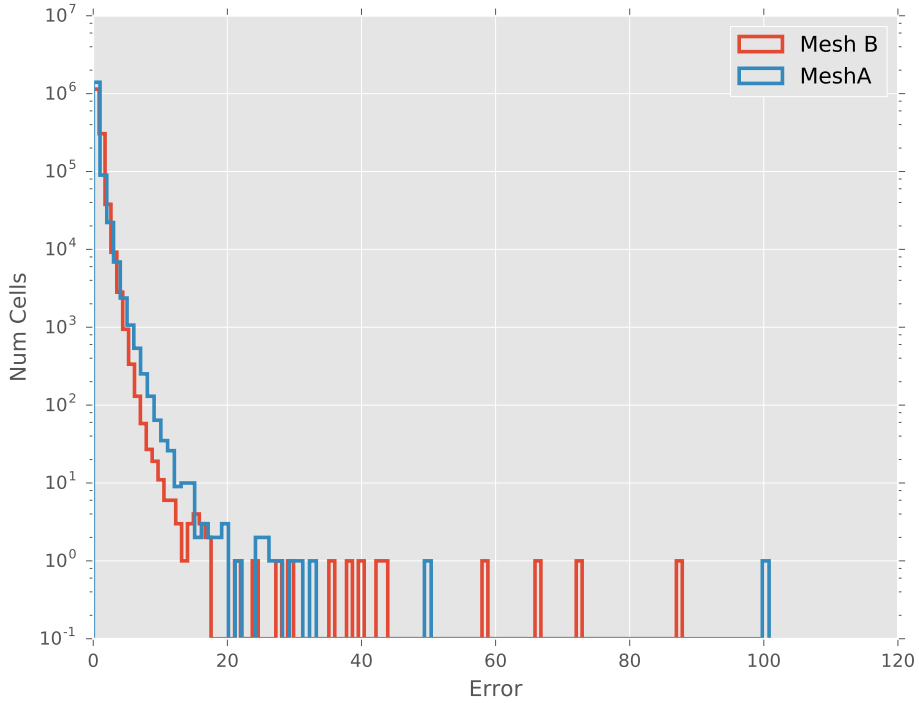


Figure 111: Histograms of cell error for solution B. Mesh B is shown in red with mesh A appearing in blue

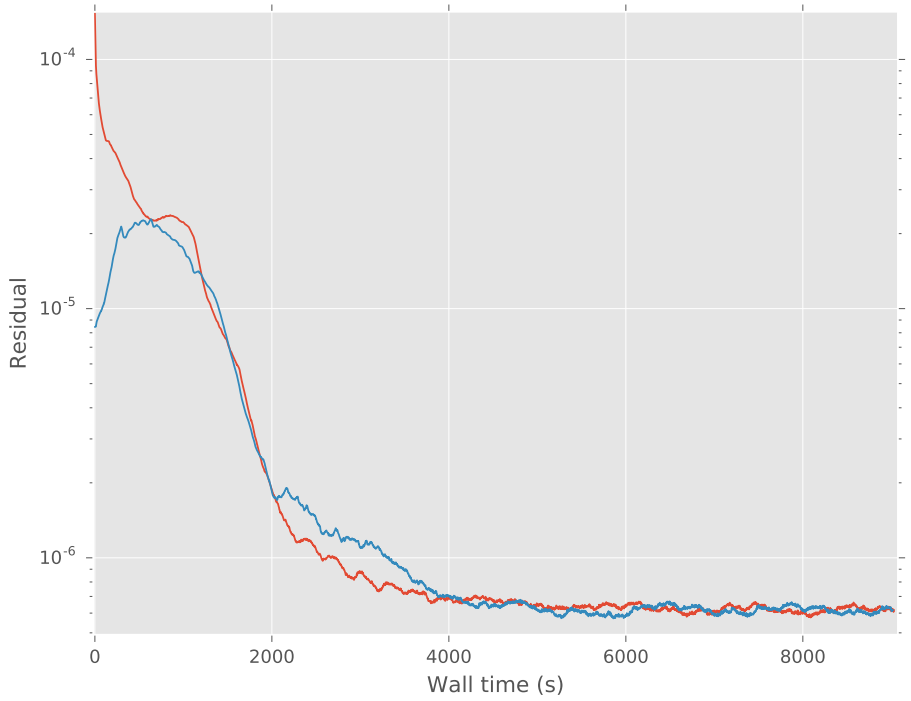


Figure 112: Comparison of convergence rates between uniformly initialized (blue) and initialized from interpolated flow field (red)

blunted ogive case showed before. Convergence rates are compared between a simulation initialized from uniform conditions (T=1,p=1,U=9) and one initialized from interpolated flow field. The convergence is estimated by computing a residual as the difference of the density field between two consequent time steps and then obtaining the volumetric average of the absolute value of such difference field ($volAverage(abs(\rho_{t+1} - \rho_t))$).

The results are shown in figure 112, where the simulation initialized from interpolated flow field shows slightly better convergence behaviour.

6 CONCLUSIONS

SUMMARY OF RESULTS

An approximate method for shock structure estimation in supersonic and hypersonic vehicles has been presented. The procedure applies a combination of supersonic flow theories over surface streamlines, generating local shocks. These are then meshed together to obtain one or multiple global shocks, taking into account multiple origins through stagnation point clustering. Detached shocks are addressed starting the procedure from an engineering correlation around the stagnation point and correcting the expansion through a correction factor. The method has been verified against CFD simulations, with each of the test cases highlighting a different aspect of the estimation procedure. Overall, the approximated shock structures show good agreement with the reference CFD results, especially taking into account the speed with which those approximations were generated. Higher error corresponds to three dimensional geometries, particularly when shock detachment is present. The source of those errors has been identified and could be addressed in future work to improve the accuracy of the estimations. These sources are the elliptic cone perturbation solution, that produces inaccurate results at low angles of incidence, and the curvature estimation from the triangular surface mesh, that introduces noise in all steps of the algorithm.

The procedure is novel in addressing two and three dimensional geometries with attached and detached shocks as well as shock structures composed of multiple shocks in one cohesive methodology. Specifically, the treatment for detached shocks with subsonic zone approximation patched to the attached shock estimation method under a correction factor, the engineering formula for detached shocks around general 3D stagnation points and the use of a clustering algorithm for multi-shock reconstruction are all original contributions.

IMPACT

The impact of the present thesis is greater in the fields of MDO, improving low order models, and MFDA, making the use of high fidelity tools more efficient and streamlined.

In the context of MDO, the method is compatible with low order aero - thermodynamic models in both assumptions and execution speed and can be used to improve the predictions of such models when shock related phenomena are present. Three examples have been presented (flow field interpolation, shock impingement detection and shock effects between proximal bodies due to perturbed flow conditions). Flow field interpolation produces a rough approximation of the flow around a supersonic vehicle. This approximation could be used to analyse, in the context of low order models, some situations where off-body conditions are important. An example of this could be the estimation of heat fluxes on re-entry capsules, taking into account radiative fluxes from the shock. The shock estimation procedure has also been used to detect shock impingement. While this is already of interest, as it highlights problematic areas to analyse in greater detail using higher fidelity models, a very important addition for the future would be to estimate shock impingement effects in the context of low order modelling. Finally, an example of proximal body interference has been shown. A fast analysis of the separation of a two stage to orbit vehicle has been performed using solely low order models.

In the context of MFDA, the method can be applied to inform the use of high fidelity tools. This would allow for a more efficient and effective use of them. Specifically, the estimation procedure shows great promise in informing meshing and initial conditions for supersonic and hypersonic simulations. The knowledge of the shock position is necessary to obtain accurate results from CFD. The shock estimation procedure can potentially provide the shock structure and flow field approximation directly from the geometry and the flow conditions, allowing for adapted meshing without the usual need of performing a preliminary simulation to estimate the shock position and inform meshing or using adaptive mesh refinement to compute mesh and solution in a coupled manner.

Overall, the present low order method for shock structure estimation shows applicability in varied areas of aerospace vehicle design, and great potential to be applied in other areas with focused future work.

AVENUES FOR FUTURE WORK

The future work should be directed to improve predictions of the method, and expand its field of application. As noted earlier, the two main sources of error are the elliptic cone perturbation solution and the surface curvature estimation. Any improvement in those two topics would greatly increase the accuracy of the method. The other big avenue for improvement would be to address shock interference and reflection. This would allow the analysis of internal flows, with very important application in propulsive analysis. And continuing in the expansion of the field of application, a low order model for shock impingement effect would be of great interest. Now that the shock position is known, this addition would make the estimation much more useful in the context of vehicle design.

BIBLIOGRAPHY

[1] Pierre Alliez, Stéphane Tayeb, and Camille Wormser. "3D Fast Intersection and Distance Computation". In: *CGAL User and Reference Manual*. 4.8. CGAL Editorial Board, 2016. URL: http://doc.cgal.org/4.8/Manual/packages.html#PkgAABB_treeSummary.

[2] John David Anderson. *Hypersonic and high temperature gas dynamics*. Aiaa, 2000. ISBN: 1-56347-459-X.

[3] John David Anderson. *Modern compressible flow: with historical perspective*. Vol. 12. McGraw-Hill New York, 1990.

[4] Timothy J. Baker. "Mesh adaptation strategies for problems in fluid dynamics". In: *Finite Elements in Analysis and Design* 25.3 (1997), pp. 243–273. URL: http://www.sciencedirect.com/science/article/pii/S0168874X96000327 (visited on 06/26/2017).

[5] Frederick S Billig. "Shock-wave shapes around spherical-and cylindrical-nosed bodies." In: *Journal of Spacecraft and Rockets* 4.6 (1967), pp. 822–823. ISSN: 0022-4650.

[6] H. Blasius. "The boundary layers in fluids with little friction". In: (1950). URL: http://ntrs.nasa.gov/search.jsp?R=20050028493 (visited on 07/05/2016).

[7] E Bonner, W Clever, and K Dunn. *Aerodynamic preliminary analysis system 2. Part 1: Theory*. NASA Contractor Report 182076. Rockwell International Corporation, 1991.

[8] Wayne J. Bordelon, Alonzo L. Frost, and Darren K. Reed. "Stage separation wind tunnel tests of a generic two-stage-to-orbit launch vehicle". In: *AIAA Paper* 4227 (2003), p. 2003. URL: http://arc.aiaa.org/doi/pdf/10.2514/6.2003-4227 (visited on 07/14/2016).

[9] Pieter G. Buning, Reynaldo J. Gomez, and William I. Scallion. "CFD approaches for simulation of wing-body stage separation". In: *AIAA Paper* 4838 (2004), p. 2004. URL: http://arc.aiaa.org/doi/pdf/10.2514/6.2004-4838 (visited on 08/03/2016).

[10] MJ Castro-Diaz, F Hecht, B Mohammadi, and O Pironneau. "Anisotropic unstructured mesh adaption for flow simulations". In: *International Journal for Numerical Methods in Fluids* 25.4 (1997), pp. 475–491. ISSN: 0271-2091.

[11] S. Choi, J. J. Alonso, I. M. Kroo, and M. Wintzer. "Multifidelity Design Optimization of Low-Boom Supersonic Jets". In: *Journal of Aircraft* 45.1 (Jan. 2008), pp. 106–118. ISSN: 0021-8669, 1533-3868. DOI: 10.2514/1.28948. URL: http://arc.aiaa.org/doi/10.2514/1.28948 (visited on 03/25/2017).

[12] Tran Kai Frank Da and David Cohen-Steiner. "Advancing Front Surface Reconstruction". In: *CGAL User and Reference Manual*. 4.8. CGAL Editorial Board, 2016. URL: http://doc.cgal.org/4.8/Manual/packages.html#PkgAdvancingFrontSurfaceReconstructionSummary.

[13] Eduardo F. D'Azevedo and R. Bruce Simpson. "On optimal triangular meshes for minimizing the gradient error". In: *Numerische Mathematik* 59.1 (1991), pp. 321–348. URL: http://link.springer.com/article/10.1007/BF01385784 (visited on 01/11/2017).

[14] Th Eggers. "Numerical Investigation on the Potential of Steam Cooling for the Skylon Spaceplane in Hypersonic Flow". In: 28th International Congress of the Aeronautical Sciences (ICAS). 2012, pp. 2012–2.

[15] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: Kdd. Vol. 96. 1996, pp. 226–231.

[16] Arvel E Gentry, Douglas N Smyth, and Wayne R Oliver. *The Mark IV Supersonic-Hypersonic Arbitrary-Body Program. Volume II. Program Formulation*. AFFDL-TR-73-159, Vol. II. Douglas Aircraft Company, 1973.

[17] Christophe Geuzaine and Jean-François Remacle. "Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities". In: *International Journal for Numerical Methods in Engineering* 79.11 (Sept. 10, 2009), pp. 1309–1331. ISSN: 00295981, 10970207. DOI: 10.1002/nme.2579. URL: http://doi.wiley.com/10.1002/nme.2579 (visited on 01/09/2017).

[18] J. Don Gray. *Summary report on aerodynamic characteristics of standard models HB-1 and HB-2*. DTIC Document, 1964. URL: http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=AD0602769 (visited on 08/02/2016).

[19] Christopher J. Greenshields, Henry G. Weller, Luca Gasparini, and Jason M. Reese. "Implementation of semi-discrete, non-staggered central schemes in a colocated, polyhedral, finite volume framework, for high-speed viscous flows". In: *International Journal for Numerical Methods in Fluids* (2009), n/a–n/a. ISSN: 02712091, 10970363. DOI: 10.1002/fld.2069. URL: http://doi.wiley.com/10.1002/fld.2069 (visited on 05/22/2015).

[20] Frank D Hains and J Wayne. "Shock Interference Bell Aerospace, Heating in Hypersonic Flows". In: *AIAA Journal* 10.11 (1972), pp. 1441–1447. ISSN: 0001-1452.

[21] Wallace Hayes. *Hypersonic flow theory*. Elsevier, 2012. ISBN: 0-323-14876-X.

[22] Susan Hert and Stefan Schirra. "3D Convex Hulls". In: *CGAL User and Reference Manual*. 4.8.1. CGAL Editorial Board, 2016. URL: http://doc.cgal.org/4.8.1/Manual/packages.html#PkgConvexHull3Summary.

[23] George E. Kaattari. *A method for predicting shock shapes and pressure distributions for a wide variety of blunt bodies at zero angle of attack*. NASA Technical Note TN D-4539. NASA, 1968. URL: https://ntrs.nasa.gov/search.jsp?R=19680012472 (visited on 02/06/2017).

[24] Lutz Kettner, Andreas Meyer, and Afra Zomorodian. "Intersecting Sequences of dD Iso-oriented Boxes". In: *CGAL User and Reference Manual*. 4.8.1. CGAL Editorial Board, 2016. URL: http://doc.cgal.org/4.8.1/Manual/packages.html#PkgBoxIntersectionDSummary.

[25] David J Kinney. "Aero-thermodynamics for conceptual design". In: *AIAA Paper* 13382 (2004).

[26] Stuart J. Laurence, R. Deiterding, and G. Hornung. "Proximal bodies in hypersonic flow". In: *Journal of Fluid Mechanics* 590 (Nov. 2007). ISSN: 0022-1120, 1469-7645. DOI: 10.1017/S0022112007007987. URL: http://www.journals.cambridge.org/abstract_S0022112007007987 (visited on 05/12/2017).

[27] Ivett A Leyva. "Shock detachment process on cones in hypervelocity flows". PhD thesis. California Institute of Technogy, 1999.

[28] H. Li and G. Ben-Dor. "Oblique-shock/expansion-fan interaction - Analytical solution". In: *AIAA Journal* 34.2 (Feb. 1996), pp. 418–421. ISSN: 0001-1452, 1533-385X. DOI: 10.2514/3.13081. URL: http://arc.aiaa.org/doi/abs/10.2514/3.13081 (visited on 01/06/2016).

[29] Sébastien Loriot, Jane Tournois, and Ilker O. Yaz. "Polygon Mesh Processing". In: *CGAL User and Reference Manual*. 4.8.1. CGAL Editorial Board, 2016. URL: http://doc.cgal.org/4.8.1/Manual/packages.html#PkgPolygonMeshProcessingSumm

[30] Eugene S Love. *A reexamination of the use of simple concepts for predicting the shape and location of detached shock waves*. Technical Note 4170. Langley Aeronautical Laboratory: NACA, 1957.

[31] John D Martel and Bruce Jolly. "Analytical Shock Standoff and Shape Prediction with Validation for Blunt Face Cylinder". In: AIAA Atmospheric Flight Mechanics Conference. 2015, p. 0523.

[32] E. Minisci, M. Vasile, and H. Liqiang. "Robust multi-fidelity design of a micro re-entry unmanned space vehicle". In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 225.11 (Nov. 1, 2011), pp. 1195–1209. ISSN: 0954-4100, 2041-3025. DOI: 10.1177/0954410011410124. URL: http://pig.sagepub.com/lookup/doi/10.1177/0954410011410124 (visited on 03/25/2017).

[33] Kelly J. Murphy, Pieter G. Buning, Bandu N. Pamadi, William I. Scallion, and Kenneth M. Jones. "Overview of transonic to hypersonic stage separation tool development for multi-stage-to-orbit concepts". In: *AIAA paper* 2595 (2004), p. 2004. URL: http://arc.aiaa.org/doi/pdf/10.2514/6.2004-2595 (visited on 08/03/2016).

[34] *OpenFOAM*. http://www.openfoam.com/. Accessed: 13-04-2016.

[35] Hans-Georg Pagendarm and B Seitz. "An algorithm for detection and visualization of discontinuities in scientific data fields applied to flow data with shock waves". In: *Scientific Visualization: Advanced Software Techniques*. Ellis Horwood Workshop Series. 1993, pp. 161–177.

[36] Sergio Pirozzoli. "Numerical methods for high-speed flows". In: *Annual review of fluid mechanics* 43 (2011), pp. 163–194. ISSN: 0066-4189.

[37] Frits H. Post, Benjamin Vrolijk, Helwig Hauser, Robert S. Laramee, and Helmut Doleisch. "The state of the art in flow visualisation: Feature extraction and tracking". In: *Computer Graphics Forum*. Vol. 22. Wiley Online Library, 2003, pp. 775–792. URL: http://onlinelibrary.wiley.com/doi/10.1111/j.1467-8659.2003.00723.x/full (visited on 08/21/2015).

[38] Marc Pouget and Frédéric Cazals. "Estimation of Local Differential Properties of Point-Sampled Surfaces". In: *CGAL User and Reference Manual*. 4.8. CGAL Editorial Board, 2016. URL: http://doc.cgal.org/4.8/Manual/packages.html#PkgJet_fitting_3Summary.

[39] M. L. Rasmussen. "Waverider Configurations Derived from Inclined Circular and Elliptic Cones". In: *Journal of Spacecraft and Rockets* 17.6 (Nov. 1980), pp. 537–545. ISSN: 0022-4650, 1533-6794. DOI: 10.2514/3.57771. URL: http://arc.aiaa.org/doi/abs/10.2514/3.57771 (visited on 04/04/2016).

[40] Thomas Scanlon, Rodrigo Cassineli Palharini, Craig White, Daniel Espinoza, and Vincent Casseau. "Simulations of rarefied and continuum hypersonic flow over re-entry objects". In: *8th European Symposium on Aerothermodynamics for Space Vehicles*. 2015. URL: http://strathprints.strath.ac.uk/51961/ (visited on 08/04/2016).

[41] Jaroslaw Sobieszczanski-Sobieski and Raphael T Haftka. "Multidisciplinary aerospace design optimization: survey of recent developments". In: *Structural optimization* 14.1 (1997), pp. 1–23. ISSN: 0934-4373.

[42] The CGAL Project. *CGAL User and Reference Manual*. 4.8. CGAL Editorial Board, 2016. URL: http://doc.cgal.org/4.8/Manual/packages.html.

[43] Randall T. Voland, Lawrence D. Huebner, and Charles R. McClinton. "X-43A hypersonic vehicle technology development". In: *Acta Astronautica* 59.1 (2006), pp. 181–191. URL: http://www.sciencedirect.com/science/article/pii/S0094576506001184 (visited on 06/14/2015).

[44] Joe D Watts. "Flight experience with shock impingement and interference heating on the X-15-2 research airplane". In: *NASA TM X-1669* (1968).

[45] Romain Wuilbercq. "Multi-disciplinary modelling of future space-access vehicles". Thesis (Ph.D.) University of Strathclyde, 2015.