Assessment of Implicit Implementation of the AUSM $^+$ Method and the SST Model for Viscous High Speed Flow

S. Colonia **, R. Steijl* ** *and G.N. Barakos* ***
**PhD candidate*
colonia@liv.ac.uk
***Lecturer*
rsteijl@liv.ac.uk
****Professor*
gbarakos@liv.ac.uk
Computational Fluid Dynamics Laboratory,
School of Engineering, University of Liverpool, Liverpool, L69 3GH, United Kingdom

Abstract
CFD analysis of high speed flow fields requires numerical methods able to cope with a wide range of flow phenomena. At the University of Liverpool, the Helicopter Multi-Block CFD code is used to model different subsonic and transonic flows. An analytical Jacobian definition for the AUSM $^+$ and the scheme itself have been implemented into the fully implicit code so that high Mach number flows can be also modelled. A description of the derivation procedures needed to obtain the analytical Jacobian is given in this paper along with a brief evaluation of the performances of the implicit scheme for different test cases, including turbulent flows. As examples of aerospace interest, a blunt body, a single cone with blunt nose, a shock-wave/turbulent boundary-layer interaction generated by a ramp, and the Orion spacecraft were considered. The SST turbulence model has been employed for the turbulent cases.

List of symbols

a	Sound speed	u_n	Normal velocity to the cell face
$\mathbf{F}$	Flux vector	$\mathbf{J}$	Jacobian matrix
$\mathbf{W}$	Vector of conservative variables	$\mathbf{P}$	Vector of primitive variable
$\mathbf{L}$	Left eigenvectors	$\mathbf{R}$	Right eigenvectors
H	Total enthalpy	γ	Specific heat ratio
n_x	Unit vector in x direction	n_y	Unit vector in y direction
n_z	Unit vector in z direction	t	Physical time-step
Re	Reynolds number	M	Mach number
ρ	Density	u	Velocity component in x direction
v	Velocity component in y direction	w	Velocity component in z direction
p	pressure	α	Incidence angle
δ	Boundary layer thickness / Standoff distance	θ	Ramp angle

Superscripts and subscripts

ϕ_i, ϕ_L	Left state	ϕ_{i+1}, ϕ_R	Right state
ϕ_n	Normal to the cell face	ϕ_∞	Free stream property

1. Introduction

Efficient and accurate computation of the aerodynamic and thermal environment of hypersonic vehicles is essential in their design and development. Computational fluid dynamics methods have gained significant prominence in recent years and have been used in hypersonic vehicle design; however, a number of challenges remain, including devising accurate and robust numerical schemes for the convective flux computation of Navier-Stokes solvers as well as the turbulent flow field development.
For this work the CFD code developed at the CFD Laboratory of the University of Liverpool, HMBv2, has been extended and used. This solver uses finite volume spatial discretisation and fully un-factored time discretisation with a GCG/ILU(0) linear system solver. It has been used successfully for a wide range of aerospace applications including subsonic and transonic flows ([1] and [2]). The code generally employs the Roe or Osher schemes for the inviscid fluxes evaluation. In the context of the present work, the AUSM $^+$ scheme has been implemented for high Mach number flows.

Nowadays, upwind flux functions have overcome the challenge to compute compressible flow phenomena reliably with reasonable accuracy. Several approximation procedures for solving the Riemann problem have been proposed in the literature: Roe, Van Leer, HLLE, AUSM and others. The main difference between these schemes is the way they approximate the exact wave structure. In order to solve the interface Riemann problem, the left and the right state at the surfaces of each finite volume need to be extrapolated from the centroid. The most common choice is the Monotone Upstream-Centred Scheme for Conservation Laws, MUSCL, introduced by Van Leer. Another interpolation technique that has been successfully used in the literature for high speed flows ([3], [4]) is the Spekreijse's interpolation introduced in [5]. As a limiter for both the schemes the Van Albada limiter seems to be the most popular. Among the different Riemann solvers the AUSM-family has been shown to be capable of solving flow fields at a wide range of Mach number and to perform reasonably well solving high speed flows. The original AUSM scheme has been introduced for the first time by M.-S. Liou in [6] and then improved in [7] obtaining the AUSM $^+$. The aim of the AUSM-family is to combine the desirable attribute belonging to both flux difference (Roe) and vector (Van Leer) splitting. The basic idea is the recognition of the convection and acoustic waves as two physically distinct processes.
The AUSM $^+$ compared to the widely used Roe scheme has been shown to be less prone to develop shock anomalies as the carbuncle phenomena ([7], [8] and [9]). Moreover in [10] a study of grid refinement and spatial order of accuracy for unsteady flow fields around different transonic airfoil has been conducted using AUSM $^+$ and Roe numerical fluxes. The results have been compared to measured data and the AUSM $^+$ solutions were in better agreement. Furthermore the AUSM $^+$ scheme yields little numerical dissipation, so it could be preferred to schemes like the HLLE which is widely considered free from the carbuncle anomaly but highly dissipative. A key issue in hypersonic flow computations is the accurate prediction of heating. In [8] and [9] three properties that a flux function should have for accurate computation of surface heat transfer rates have been proposed: 1) shock stability/robustness, 2) conservation of total enthalpy and 3) resolving boundary layer (which means solving contact discontinuities, as demonstrated by Van leer et al. in [11]). The Roe scheme can resolve contact discontinuities but it does not guarantee the enthalpy preservation and the shock stability, while the Van Leer scheme gives good prediction of shock structures but it is not enthalpy-preserving and not formulated to incorporate effects of contact discontinuities. The AUSM $^+$, instead, is formulated to guarantee the enthalpy preservation and resolve contact discontinuities, but its robustness is not always guaranteed. For example in [8] and [9] some convergence issues due to shock instabilities were reported. However, it has to be highlighted that in these works the formulation of interface speed of sound such that a normal shock can be exactly resolved between two discontinuous states is adopted. In the present work, as shown later on, the entropy satisfying formulation has been chosen and seems to perform well without showing instabilities. Unfortunately, in [9], it has been also shown that no flux function between Roe, Van Leer, HLLE and AUSM $^+$ yields satisfactory predictions of the heat fluxes for a wide range of test cases. In particular the AUSM $^+$-type fluxes gave reasonable predictions for the heat transfer for a 2-D problem like a cylinder but not for a 3-D sphere.
Various explicit and implicit schemes have been proposed to advance in time the system of ordinary differential equations, obtained by the spatial discretisation. In an explicit method, the solution at the time step $n + 1$ depends only on the known solution at the previous time step. In the implicit schemes, the new solution does not only depend on the known solution at the previous time step, but also on a coupling between the grid point variables at the new time step. Indeed, an implicit approach, after the linearisation of the residual at the new time step, results in a large system of linear equations which as the time step tends to infinity results in the standard Newton's method The system of linear equations needs to be solved and this task can be accomplished using direct or iterative methods. The former are based on the exact inversion of the system sparse matrix by means of the Gaussian elimination, as can be found in [12], or a direct sparse matrix method, like the Boeing Real Sparse Library [13]. Although the recovery of the standard Newton's method, and of the quadratic convergence, was demonstrated on both structured and unstructured grids [12], [14], their application in complex three dimensional problems requires an excessively high computational effort. Thus, in these cases the linear system has to be solved using an iterative matrix inversion methodology. Different iterative methods have been proposed in the literature. Among them there are the Alternating Direction Implicit (ADI) scheme [15], the line Jacobi scheme [16], the Lower-Upper Symmetric Gauss-Siedel (LU-SGS) scheme [17] and the Newton-Krylov's methods. The first four methods are based on a splitting of the implicit operator into a sum or products of, decoupled, parts which can be inverted more easily. For this reason they introduce a factorisation error and are less strictly implicit (the coupling between the grid point variables at the new time step is not considered overall but only along particular directions). The Krylov subspace methods, instead, treat the system of linear equation in a more global way, allowing a fully un-factored approach in which the new time level is introduced simultaneously for all the cells. Obviously, this leads to an increment of the computational effort. Among the Krylov methods two schemes need to be noticed: the Generalised Conjugate Gradient (GCG) methods introduced in [18] and [19]-[20] and the Generalised Minimal Residual (GMRES) method introduced in [21]. Finally, it has to be remembered that the efficency of Krylov-subspace methods depends strongly on the preconditioning operation. The purpose of the preconditioning is to cluster the eigenvalues of the system matrix around unity. One of the most successful preconditioners is the Incomplete Lower Upper

factorisation method, [22], with different levels of fill-in (commonly zero, ILU(0)).
Until the early nineties memory intensive methods like the Newton's method were severely restricted by computer technology. One of the first attempts to employ a Newton's method solver to high speed flows is reported in [3] and [4]. The work follows some of the ideas developed in [12] for subsonic and transonic flows, one in particular was to add a dumping term to the Jacobian matrix diagonal to alleviate the start-up problems of the Newton's method. In both these line of works [3]-[4] and [12] a direct method has been used, but in following works, [23] and [24] respectively, iterative methods have been taken into account with an ILU(0) factorisation. In the recent literature, to the knowledge of the authors, only one work has successfully solved, efficiently, high speed, turbulent flow fields (taking into account also chemical reaction) with a fully un-factored implicit solver. In this work, [25]-[26], the authors use a finite element method with a Petrov-Galerkin scheme for the spatial discretisation and a fully implicit time discretisation with a GMRES/ILU(0) solver.
For the implicit formulation, the derivatives of the interface fluxes are needed. In [10], [27] and [28] a numerical Jacobian for the AUSM $^+$, AUSM $^+$up and AUSMPW $^+$ schemes, respectively, has been chosen and successfully employed for low Mach, subsonic, transonic and hypersonic cases. However, when possible, an analytical Jacobian is preferred because it is more efficient. An analytical Jacobian for the AUSM and AUSM $^+$ scheme has been studied in [29] and [30] respectively. In [29] a comparative study of the analytical Jacobian for different schemes is proposed and the AUSM Jacobian failed to converge, but not enough information is given about the derivation of the analytical Jacobian. In [30], instead, a complete study of the derivation of a simplified analytical Jacobian for the AUSM $^+$ was presented. Moreover the latter has been successfully applied in a point implicit Runge-Kutta scheme to solve subsonic and transonic flows. Nevertheless, since the authors were not interested in high speed cases no discussion about the impact of the simplifications on the solution of these flow fields was reported.
In this work a fully analytical Jacobian of the AUSM $^+$ has been implemented in a finite volume and fully un-factored implicit solver. Subsequently, the CFD code has been used to predict different high speed test cases in order to evaluate the capabilities of the scheme. Not many works, dealing with high Mach flow, used a fully un-factored implicit approach, among them the work of Kirk et al., [25]-[26], has to be noticed. However, this differs form the present work since a finite element method is employed. Regarding the definition of a AUSM-type scheme Jacobian, a simplified derivation for the AUSM $^+$ has been defined and successfully applied to subsonic and transonic flows by Langer and Li, [30], while for high Mach flows numerical approximations are commonly used.
In the high speed flow simulation, turbulence modelling remains a great challenge as a major source of errors in the prediction of aerodynamic forces and heat transfer. Based on the review paper of Roy [31] it is clear that a lot of work has been done in order to validate different turbulence models for simple high speed flows. However, validation works are still going on for more complex cases. In this work we want to give a small contribute to this topic by employing the two-equation SST turbulence model, with the AUSM $^+$, solving shock-wave/turbulent boundary-layer interactions. The solutions have been compared with the experimental results of [32].
Finally, the turbulence model and the approximate Riemann solver has been used to evaluate the aerodynamic coefficients of the ORION CEV at different angles of attack. To validate the results, the latter have been compared to the experimental data collected in [33].

2. HMB solver

2.1 Fully implicit formulation for a steady case

The Helicopter Multi-Block (HMB) code, developed at Liverpool University, is used in the present work. The Navier-Stokes (NS) equations are discretised using a cell-centred finite volume approach. The computational domain is divided into a finite number of non-overlapping control-volumes, and the governing equations are applied in integral-conservation form at each cell. The equations are written in a curvilinear co-ordinate system. The spatial discretisation of the NS equations leads to a set of ordinary differential equations in time,

$$\frac{d}{dt}(\mathbf{W}_{i,j,k}V_{i,j,k}) = -\mathbf{R}_{i,j,k}(\mathbf{W}) \tag{1}$$

where $\mathbf{W}$ and $\mathbf{R}$ are the vectors of the cell conserved variables and residuals, respectively. Using an implicit time discretisation on the pseudo-time t^*,

$$\frac{\mathbf{W}^{m+1}_{i,j,k} - \mathbf{W}^{m}_{i,j,k}}{\Delta t^*} = -\frac{1}{V_{i,j,k}}\mathbf{R}_{i,j,k}(\mathbf{W}^{m+1}_{i,j,k}) \tag{2}$$

where the superscript $m + 1$ denotes the time level $(m + 1)\Delta t^*$ in pseudo-time. In equation (2) the flux residual on the right hand side is evaluated at the new time level $m + 1$ and is therefore expressed in terms of the unknown solution at this new time level. The flux residual $\mathbf{R}_{i,j,k}(\mathbf{W}^{m+1}_{i,j,k})$ is linearised in the pseudo-time variable t^* as follows,

$$\begin{aligned}
\mathbf{R}_{i,j,k}\left(\mathbf{W}^{m+1}\right) &= \mathbf{R}_{i,j,k}\left(\mathbf{W}^{m}\right) + \frac{\partial \mathbf{R}_{i,j,k}}{\partial t^*}\Delta t^* + O(\Delta t^{*2}) \\
&\approx \mathbf{R}_{i,j,k}\left(\mathbf{W}^{m}\right) + \frac{\partial \mathbf{R}_{i,j,k}}{\partial \mathbf{W}_{i,j,k}}\frac{\partial \mathbf{W}_{i,j,k}}{\partial t^*}\Delta t^* \\
&\approx \mathbf{R}_{i,j,k}\left(\mathbf{W}^{m}\right) + \frac{\partial \mathbf{R}_{i,j,k}}{\partial \mathbf{W}_{i,j,k}}\Delta \mathbf{W}_{i,j,k}
\end{aligned} \tag{3}$$

where $\Delta\mathbf{W}_{i,j,k} = \mathbf{W}^{m+1}_{i,j,k} - \mathbf{W}^{m}_{i,j,k}$. Substituting equations (3) into (2), and rewriting in terms of the primitive variables $\mathbf{P}$, the fully implicit system to be solved is as follows,

$$\left[\left(\frac{V_{i,j,k}}{\Delta t^*}\frac{\partial \mathbf{W}_{i,j,k}}{\partial \mathbf{P}_{i,j,k}} + \frac{\partial \mathbf{R}_{i,j,k}}{\partial \mathbf{P}_{i,j,k}}\right]\Delta \mathbf{P}_{i,j,k} = -\mathbf{R}_{i,j,k}(\mathbf{W}^m) \tag{4}$$

Note that the system is solved in the primitive variables formulation for simplicity and stability reasons.

2.2 Jacobian Formulation

Considering the inviscid part of the residual of the left face for cell "i", denoted by $f_{i-\frac{1}{2}}$, and following the general approach for Riemann solvers,

$$\mathbf{f}_{i-\frac{1}{2}} = \mathbf{f}_{i-\frac{1}{2}}(\mathbf{P}_l, \mathbf{P}_r) \tag{5}$$

where $\mathbf{P}_l$ and $\mathbf{P}_r$ are the left and right states of the Riemann problem. Applying MUSCL interpolation, both states are $\mathbf{P}_l = \mathbf{P}_l(\mathbf{P}_{i-2}, \mathbf{P}_{i-1}, \mathbf{P}_i)$ and $\mathbf{P}_r = \mathbf{P}_r(\mathbf{P}_{i-1}, \mathbf{P}_i, \mathbf{P}_{i+1})$, respectively. $\mathbf{f}_{i-\frac{1}{2}}$ is then computed using AUSM $^+$ [7] scheme. For the cell face $i - \frac{1}{2}$ there are four contributions to the Jacobian matrix

$$\frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i-2}}, \quad \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i-1}}, \quad \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i}}, \quad \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i+1}}. \tag{6}$$

To avoid ill-conditioning a first order Jacobian is employed, indeed the exact Jacobian matrix is approximated by removing the dependence in the MUSCL interpolation;

$$\frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i-2}} \approx \mathbf{0}; \quad \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i-1}} \approx \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{l}}; \quad \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i}} \approx \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{r}}; \quad \frac{\partial \mathbf{f}_i}{\partial \mathbf{P}_{i+1}} \approx \mathbf{0} \tag{7}$$

3. The AUSM $^+$ scheme

As a first step in common in all AUSM schemes, the inviscid flux is explicitly split into convective and pressure fluxes; then it is possible to write the numerical flux, Eq. (5), as follow:

$$\mathbf{f}_{i-1/2} = \dot{m}_{1/2}\mathbf{P} + \mathbf{p}_{1/2} \tag{8}$$

where using an upwind approach:

$$\mathbf{P} = \left\{ \mathbf{P}_L \quad \text{if } \dot{m}_{1/2} > 0; \quad \mathbf{P}_R \quad \text{otherwise} \right\} \tag{9}$$

3.1 Mass flux

The mass flux at the interface in accordance with the idea of up-winding has the form of:

$$\dot{m}_{1/2} = a_{1/2} M_{1/2} \left\{ \rho_L \quad \text{if } M_{1/2} > 0; \quad \rho_R \quad \text{otherwise} \right\} \tag{10}$$

where $a_{1/2}$ and $M_{1/2}$ are the interface speed of sound and Mach number respectively.
The interface Mach number is expressed in terms of the left and right Mach numbers, M_L and M_R, as follows

$$M_{1/2} = M^+_{(4)}(M_L) + M^-_{(4)}(M_R) \tag{11}$$

where

$$M_L = \frac{u_{n,L}}{a_{1/2}} \quad M_R = \frac{u_{n,R}}{a_{1/2}} \tag{12}$$

and the polynomials $M^{\pm}_{(4)}$ are defined next

$$M^{\pm}_{(4)}(M) = \begin{cases} M^{\pm}_{(1)}(M) & \text{if } |M| \geq 1 \\ M^{\pm}_{(2)}(M)(1 \mp 16\beta M^{\mp}_{(2)}(M)) & \text{otherwise} \end{cases} \tag{13}$$

with $\beta = 1/8$ and

$$M^{\pm}_{(1)}(M) = \tfrac{1}{2}(M \pm |M|); \quad M^{\pm}_{(2)}(M) = \pm\tfrac{1}{4}(M \pm 1)^2 \tag{14}$$

The entropy-satisfying definition of the interface speed of sound is used in this work, then $a_{\frac{1}{2}}$ is given by the following expression:

$$a_{1/2} = min(\widehat{a}_L, \widehat{a}_R); \quad \widehat{a}_L = a^{*2}_L / max(a^*_L, u_{n,L}), \quad \widehat{a}_R = a^{*2}_R / max(a^*_R, -u_{n,R}) \tag{15}$$

where $u_{n,L/R}$ and $a^*_{L/R}$ are the left and right normal velocities and critical speed of sounds at the cell face. The latter, under the hypothesis of perfect gas, can be expressed in terms of the total enthalpy.

$$a^{*2}_{L/R} = \frac{2(\gamma - 1)}{\gamma + 1} H_{L/R} \tag{16}$$

3.2 Pressure flux

In the AUSM-family a general interface pressure formula is used as a starting point

$$p_{1/2} = P^+_{(5)}(M_L)p_L + P^-_{(5)}(M_R)p_R \tag{17}$$

where $p_{L/R}$ are the interface left and right pressures and the polynomials $P^{\pm}_{(5)}$ are

$$P^{\pm}_{(5)}(M) = \begin{cases} \frac{1}{M} M^{\pm}_{(1)}(M) & \text{if } |M| \geq 1 \\ M^{\pm}_{(2)}(M)(\pm 2 - M \mp 16\alpha M M^{\mp}_{(2)}(M)) & \text{otherwise} \end{cases} \tag{18}$$

with $\alpha = 3/16$.

4. A Jacobian matrix for AUSM $^+$

The Jacobian matrix is calculated analytically by repeated application of the chain rule. The residual for one cell is built up as a summation of the fluxes through the cell faces. Then, considering the inviscid numerical flux, Eq. (8)

$$\frac{\partial \mathbf{f}_{i-\frac{1}{2}}}{\partial \mathbf{P}_{L/R}} = \frac{\partial \dot{m}_{1/2}}{\partial \mathbf{P}_{L/R}} \mathbf{P} + \dot{m}_{1/2} \frac{\partial \mathbf{P}}{\partial \mathbf{P}_{L/R}} + \frac{\partial \mathbf{p}_{1/2}}{\partial \mathbf{P}_{L/R}} \tag{19}$$

where $\mathbf{P}$ is defined in equation (9).

4.1 Derivatives of the interface speed of sound

As hinted in the introduction the interface speed of sound is critical in the AUSM $^+$. A good analytical Jacobian for this scheme has to represent its dependencies as best as possible. Differentiating expression (15), the presence of the min/max operators leads to a dual formulation at the borderline cases like $\widehat{a}_L = \widehat{a}_R$ and $a^*_{L/R} = u_{n,L/R}$; it has to be noticed here that the interface speed of sound is not represented by a continuous function. While it is not an issue in computing the numerical fluxes, the Jacobian should take into account the following possibilities (see equation (15)):

- when $\widehat{a}_L = \widehat{a}_R$ both the left and right state could be chosen by the min/max operators to evaluate $a_{1/2}$
- when $a^*_{L/R} = \pm u_{n,L/R}$ the interface speed of sound could be either a function only of the critical speeds of sound or a function of $a^*_{L/R}$ and the normal velocities $u_{n,L/R}$.

Here we suggest a simple approach to deal with these situations. When $\widehat{a}_L = \widehat{a}_R$ and $a^*_{L/R} = \pm u_{n,L/R}$ we consider the interface speed of sound derivative as in equation (20) where, at the points of discontinuity, the average of the left and right limits is used. With this choice we hope to consolidate the stability of the scheme. This discussion led to the following expression to evaluate the derivative of $a_{1/2}$

$$\frac{\partial a_{1/2}}{\partial \mathbf{P}_{L/R}} = \begin{cases} \text{for } \widehat{a}_L < \widehat{a}_R & \begin{cases} \dfrac{\partial a^*_L}{\partial \mathbf{P}_{L/R}} & \text{if } a^*_L > u_{n,L} \\ \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \dfrac{a^{*2}_L}{u_{n,L}} & \text{if } a^*_L < u_{n,L} \\ \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \left[\dfrac{1}{2} \left(a^*_L + \dfrac{a^{*2}_L}{u_{n,L}} \right) \right] & \text{if } a^*_L = u_{n,L} \end{cases} \\ \text{for } \widehat{a}_L > \widehat{a}_R & \begin{cases} \dfrac{\partial a^*_R}{\partial \mathbf{P}_{L/R}} & \text{if } a^*_R > -u_{n,R} \\ \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \dfrac{a^{*2}_R}{-u_{n,R}} & \text{if } a^*_R < -u_{n,R} \\ \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \left[\dfrac{1}{2} \left(a^*_R + \dfrac{a^{*2}_R}{-u_{n,R}} \right) \right] & \text{if } a^*_R = -u_{n,R} \end{cases} \\ \text{for } \widehat{a}_L = \widehat{a}_R & \begin{cases} \dfrac{1}{2} \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \left(a^*_L + a^*_R \right) & \text{if } a^*_L > u_{n,L} \ \& \ a^*_R > -u_{n,R} \\ \dfrac{1}{2} \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \left(\dfrac{a^{*2}_L}{u_{n,L}} + \dfrac{a^{*2}_R}{-u_{nR}} \right) & \text{if } \dfrac{a^{*2}_L}{u_{n,L}} = \dfrac{a^{*2}_R}{-u_{n,R}} \\ \dfrac{1}{2} \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \left(a^*_L + \dfrac{a^{*2}_R}{-u_{n,R}} \right) & \text{if } a^*_L = \dfrac{a^{*2}_R}{-u_{n,R}} \\ \dfrac{1}{2} \dfrac{\partial}{\partial \mathbf{P}_{L/R}} \left(\dfrac{a^{*2}_L}{u_{n,L}} + a^*_R \right) & \text{if } \dfrac{a^{*2}_L}{u_{L,n}} = a^*_R \end{cases} \end{cases} \tag{20}$$

where

$$\begin{gathered} \frac{\partial a^*_L}{\partial \mathbf{P}_L} = \frac{\gamma-1}{a^*_L(\gamma+1)} \left\{ -\frac{\gamma}{\gamma-1} \frac{p_L}{\rho^2_L},\ u_L,\ v_L,\ w_L,\ \frac{\gamma}{\gamma-1} \frac{1}{\rho_L} \right\} \\ \frac{\partial a^*_R}{\partial \mathbf{P}_R} = \frac{\gamma-1}{a^*_R(\gamma+1)} \left\{ -\frac{\gamma}{\gamma-1} \frac{p_R}{\rho^2_R},\ u_R,\ v_R,\ w_R,\ \frac{\gamma}{\gamma-1} \frac{1}{\rho_R} \right\} \\ \frac{\partial a^*_L}{\partial \mathbf{P}_R} = 0 \\ \frac{\partial a^*_R}{\partial \mathbf{P}_L} = 0 \end{gathered} \tag{21}$$

and

$$
\begin{aligned}
\frac{\partial u_{n,L}}{\partial \mathbf{P}_L} &= \left\{0,\ n_x,\ n_y,\ n_z,\ 0\right\} \\
\frac{\partial u_{n,R}}{\partial \mathbf{P}_R} &= \left\{0,\ n_x,\ n_y,\ n_z,\ 0\right\} \\
\frac{\partial u_{n,L}}{\partial \mathbf{P}_R} &= 0 \\
\frac{\partial u_{n,R}}{\partial \mathbf{P}_L} &= 0
\end{aligned}
\tag{22}
$$

$n = \left\{n_x,\ n_y,\ n_z\right\}^T$ is the cell face normal.

4.2 Derivatives of the interface Mach number

From equation (11) follows

$$
\frac{\partial M_{1/2}}{\partial \mathbf{P}_{L/R}} = \frac{\partial M^{+}_{(4)}(M_L)}{\partial \mathbf{P}_{L/R}} + \frac{\partial M^{-}_{(4)}(M_R)}{\partial \mathbf{P}_{L/R}} \tag{23}
$$

where considering equation (13)

$$
\frac{\partial M^{+}_{(4)}(M_L)}{\partial \mathbf{P}_{L/R}} = \begin{cases} \dfrac{\partial M^{+}_{(1)}(M_L)}{\partial \mathbf{P}_{L/R}} & \text{if } |M_L| \geq 1 \\ \dfrac{\partial M^{+}_{(2)}(M_L)}{\partial \mathbf{P}_{L/R}}(1 + 16\beta M^{-}_{(2)}(M_L)) - 16\beta M^{+}_{(2)}(M_L)\dfrac{\partial M^{-}_{(2)}(M_L)}{\partial \mathbf{P}_{L/R}} & \text{otherwise} \end{cases} \tag{24}
$$

$$
\frac{\partial M^{-}_{(4)}(M_R)}{\partial \mathbf{P}_{L/R}} = \begin{cases} \dfrac{\partial M^{-}_{(1)}(M_R)}{\partial \mathbf{P}_{L/R}} & \text{if } |M_R| \geq 1 \\ \dfrac{\partial M^{-}_{(2)}(M_R)}{\partial \mathbf{P}_{L/R}}(1 + 16\beta M^{+}_{(2)}(M_R)) + 16\beta M^{-}_{(2)}(M_R)\dfrac{\partial M^{+}_{(2)}(M_R)}{\partial \mathbf{P}_{L/R}} & \text{otherwise} \end{cases} \tag{25}
$$

and

$$
\frac{\partial M^{\pm}_{(1)}(M_{L/R})}{\partial \mathbf{P}_{L/R}} = \pm\frac{1}{2}\left(\frac{\partial M_{L/R}}{\partial \mathbf{P}_{L/R}} \pm \frac{\partial |M_{L/R}|}{\partial \mathbf{P}_{L/R}}\right) \tag{26}
$$

$$
\frac{\partial M^{\pm}_{(2)}(M_{L/R})}{\partial \mathbf{P}_{L/R}} = \pm\frac{1}{2}(M_{L/R} \pm 1)\frac{\partial M_{L/R}}{\partial \mathbf{P}_{L/R}} \tag{27}
$$

with $\frac{\partial M_L}{\partial \mathbf{P}_{L/R}}$, $\frac{\partial |M_L|}{\partial \mathbf{P}_{L/R}}$, $\frac{\partial M_R}{\partial \mathbf{P}_{L/R}}$ and $\frac{\partial |M_R|}{\partial \mathbf{P}_{L/R}}$ obtained from equations (12) taking in account equations (20), (21), (22). It has to be noticed that the split Mach number polynomials, see Eq. (13), are continuous functions and so result their derivatives, Eqs. (24) and (25).

4.3 Derivatives of the pressure flux

Finally regarding the pressure flux derivative, from equation (17)

$$\frac{\partial p_{1/2}}{\partial \mathbf{P}_{L/R}} = \frac{\partial P^+_{(5)}(M_L)}{\partial \mathbf{P}_{L/R}} p_L + \frac{\partial P^-_{(5)}(M_R)}{\partial \mathbf{P}_{L/R}} p_R + P^+_{(5)}(M_L) \frac{\partial p_L}{\partial \mathbf{P}_{L/R}} + P^-_{(5)}(M_R) \frac{\partial p_R}{\partial \mathbf{P}_{L/R}} \tag{28}$$

where considering equation (18)

$$\frac{\partial P^+_{(5)}(M_L)}{\partial \mathbf{P}_{L/R}} = \begin{cases} \frac{1}{M_L} \frac{\partial M^+_{(1)}(M_L)}{\partial \mathbf{P}_{L/R}} - \frac{M^+_{(1)}(M_L)}{M^2_L} \frac{\partial M_L}{\partial \mathbf{P}_{L/R}} & \text{if } |M_L| \geq 1 \\ \frac{\partial M^+_{(2)}(M_L)}{\partial \mathbf{P}_{L/R}} (2 - M_L - 16\alpha M_L M^-_{(2)}(M_L)) - & \\ & \text{otherwise} \\ M^+_{(2)}(M_L) \left(\frac{\partial M_L}{\partial \mathbf{P}_{L/R}} + 16\alpha (M_L \frac{\partial M^-_{(2)}(M_L)}{\partial \mathbf{P}_{L/R}} + M^-_{(2)}(M_L) \frac{\partial M_L}{\partial \mathbf{P}_{L/R}}) \right) & \end{cases} \tag{29}$$

$$\frac{\partial P^-_{(5)}(M_R)}{\partial \mathbf{P}_{L/R}} = \begin{cases} \frac{1}{M_R} \frac{\partial M^-_{(1)}(M_R)}{\partial \mathbf{P}_{L/R}} - \frac{M^-_{(1)}(M_R)}{M^2_R} \frac{\partial M_R}{\partial \mathbf{P}_{L/R}} & if \ |M_R| \geq 1 \\ \frac{\partial M^-_{(2)}(M_R)}{\partial \mathbf{P}_{L/R}} (-2 - M_R + 16\alpha M_R M^+_{(2)}(M_R)) - & \\ & \text{otherwise} \\ M^-_{(2)}(M_L) \left(\frac{\partial M_R}{\partial \mathbf{P}_{L/R}} - 16\alpha (M_R \frac{\partial M^+_{(2)}(M_R)}{\partial \mathbf{P}_{L/R}} + M^+_{(2)}(M_R) \frac{\partial M_R}{\partial \mathbf{P}_{L/R}}) \right) & \end{cases} \tag{30}$$

Also the derivatives of the fifth degree polynomials are continuous as the polynomials definitions (18).

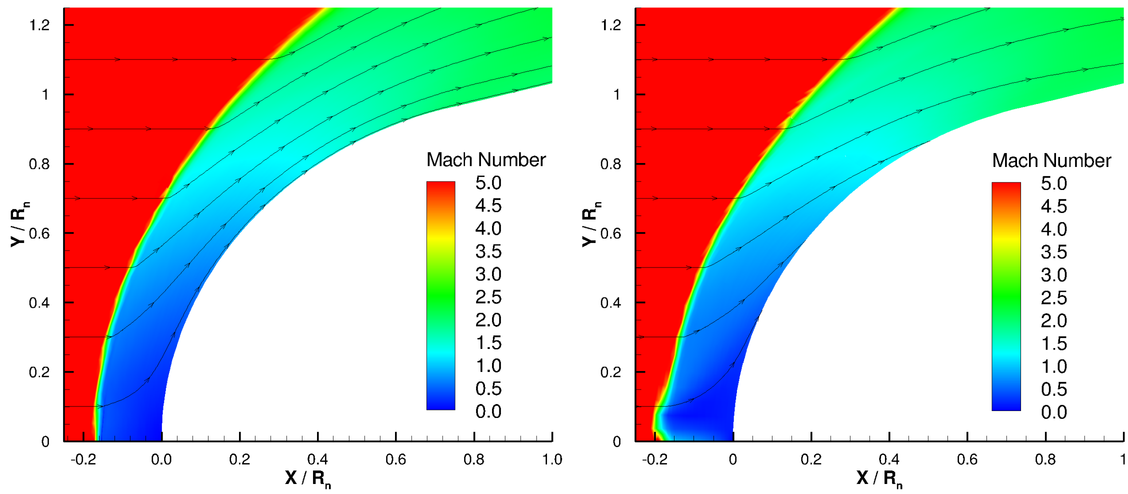
5. Comparison of the AUSM $^+$ and Roe schemes

The geometry considered for the comparison is a 15^o cone with a blunt nose of radius $R = 0.01L$, where L is the length of the cone. Some results are compared also with a theoretical approach and the correlation of [34], see table 1.
The AUSM $^+$ predictions showed the best agreement with the theory and the correlation results. The agreement given for the stagnation quantities is quite remarkable, the differences are less than 0.2%. The standoff distance is slightly under-predicted, about 7%. The Roe scheme, instead, gives over-predicted stagnation point quantities, 5% and 18%, and an underestimated, about 15%, standoff distance. Moreover, figures 1a and 1b show that the shock predicted by the AUSM $^+$ present less spurious oscillations unlike the Roe scheme, especially around the stagnation point.

Table 1: Comparison of AUSM $^+$ and Roe schemes with theory results and correlation [34].

Non-dimensional quantity	Method	Values
Stagnation	AUSM $^+$	5, 439
point	Roe	5.716
ρ/ρ_∞	THEORY	5, 442
Stagnation	AUSM $^+$	32.593
point	Roe	38.5
p/p_∞	THEORY	32.653
Standoff	AUSM $^+$	0.152
distance	Roe	0.128
δ/R_n[a]	[34]	0.163

[a] R_n is the nose curvature radius



(a) AUSM $^+$ (b) Roe

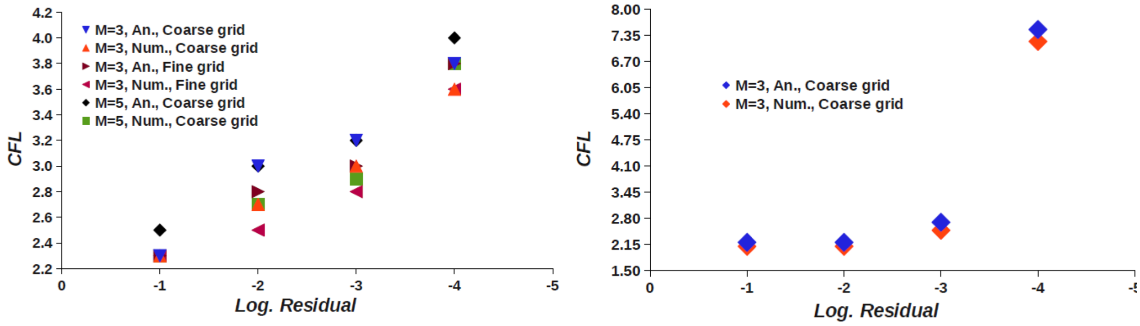
Figure 1: Inviscid flow around a 15^o cone with blunt nose ($R = 0.01L$) at $M = 5$; fully-implicit.

6. Performance of the implicit scheme

In this section, a review of the performance of the implemented implicit AUSM $^+$ schemes is given. The inviscid flow field around an infinite cylinder, as classical blunt body case of aerospace interest, has been considered. In order to evaluate the maximum CFL numbers that can be run at different norm-of-the-error levels with the implicit scheme, two different Mach numbers and grid refinements have been considered. The fine grid is obtained from the coarse by halving the cell size in the transverse direction to the shock. In the present work the following norm-of-the-error has been used:

$$log\left(\frac{L_2(Res.\ t > 0)}{L_2(Res.\ t = 0)}\right) \tag{31}$$

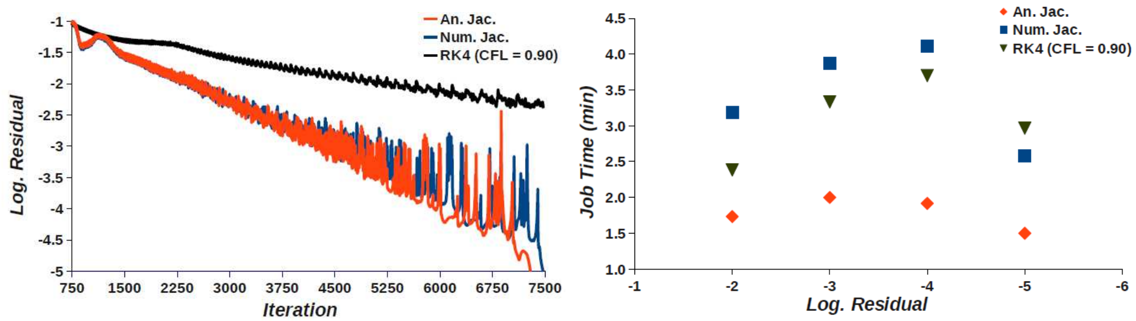
Looking at figure 2a it can be claimed that the analytical Jacobian is well defined. For both Mach 3 and 5, and grid refinements, the solver can run at a CFL numbers equal and often even higher than the respective numerical Jacobian. The numerical Jacobian is evaluated by second order central finite differences. The slightly lower performance, in terms of CFL, of the latter can be due to the fact that the interface speed of sound definition, Eq. (15), does not result in a continuous function and then it could poorly affect the numerical approximations of its derivative.



(a) Infinite cylinder, inviscid flow.

(b) Orion CEV, laminar flow ($Re = 1\ x\ 10^{5}$).

Figure 2: CFL comparison: numerical and analytical Jacobian, AUSM $^+$ with entropy satisfying $a_{1/2}$.



(a) Residual vs Iteration.

(b) Time vs Residual.

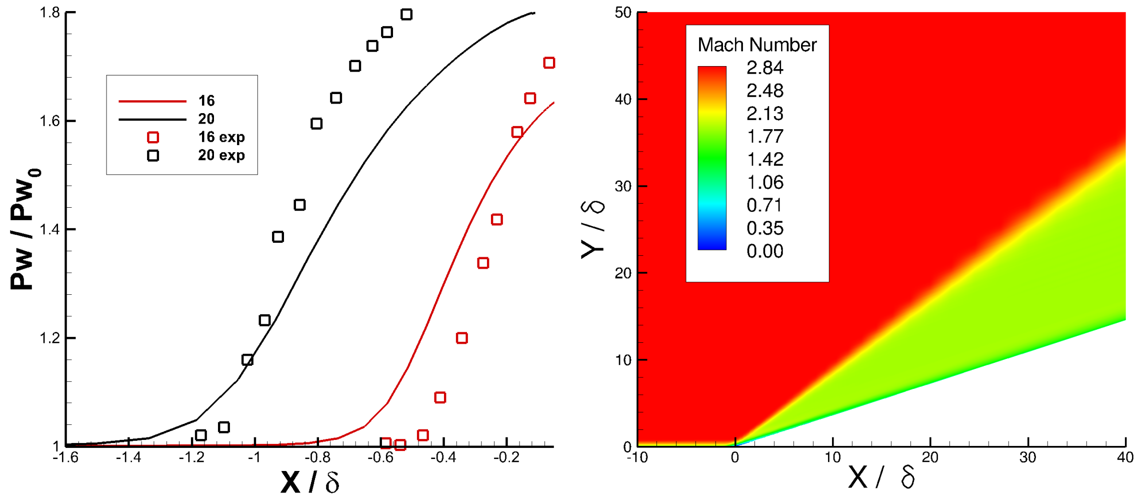
Figure 3: Computational cost comparison: infinite cylinder, coarse grid, inviscid flow, AUSM $^+$ scheme with entropy satisfying $a_{1/2}$, $M = 3$.

To evaluate the scheme behaviour for more complex cases a laminar flow field around the Orion CEV has been considered. The grid used, shown in figure 6b of section 8, has a spatial resolution normal to the shock similar to the infinite cylinder coarse grid. As it can be seen from figure 2b the implicit scheme allows to run at least CFL numbers around 2.5 also in presence of the strong shocks, expansions and interactions characterising the flow field around the Orion. Regarding the computational time a series of test has been conducted on a quad-core Xeon® CPU machine. Figures 3 shows some results for Mach 3. It can be noticed that the analytical Jacobian leads to a solver that is even two times faster then the respective numerical one. This is due mainly to the higher computational efficency of evaluating an analytical Jacobian compared to the numerical approach. In comparison to the explicit, 4-stage Runge Kutta, AUSM $^+$ the implicit scheme becomes 30% and 40% faster after the logarithm of the normalised residual has dropped to -1 and -2, respectively, due to the increased CFL numbers. Finally, a comparison between the times needed to obtain a solution for the infinite cylinder case, with a norm of the error equal to -7 has been conducted. The explicit, 4-stage Runge Kutta, time marching with a CFL number of 0.9 needed 19 *min* to obtain the solution. For the implicit method, with the analytical Jacobian, two approaches have been considered. The first involved the 4-stage Runge Kutta till a logarithm of the residual of -1 and then the implicit scheme till -7 with a CFL equal to 2.5. In the second one, instead, the explicit scheme till -2 and then the implicit method, with CFL of 3, till -7 have been used. In both latter cases, the time to obtain the solution has been decreased to 13 *min* and 11 *min*, respectively. So, this comparison confirms that the implicit approach is 30 – 40% faster than the fully explicit time marching.

7. Shock-wave/ turbulent boundary-layer interaction test case

In [32] shock-wave/boundary-layer interactions, generated using two-dimensional compression ramps, were studied experimentally. The characteristics of the incoming boundary layer were: $\delta = 24\ mm$, $M_\infty = 2.84$, $Re = 6.5\ x\ 10^{7}\ m^{-1}$ and different ramp angle have been considered. Among the results, the curves of the wall pressure along the recirculation zones are presented.

In this work, we solved numerically the same flow fields using the SST model and the AUSM $^+$ scheme. Figure 4a shows the comparison for the pressure curves, while in figures 4b, 5a and 5b the Mach contours are presented for two ramp angles.

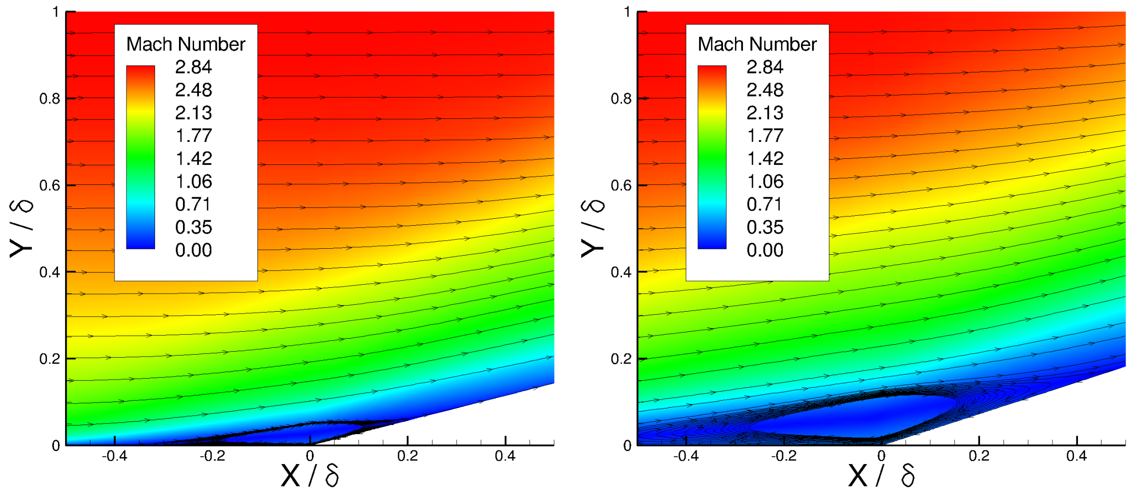


(a) Pressure curves for $\theta = 16\,^o$ and $\theta = 20\,^o$ ramp angles.

(b) Mach contours at $\theta = 20\,^o$.

Figure 4: SST, AUSM $^+$ with entropy satisfying $a_{1/2}$, $M = 2.84$, $Re = 6.5 \; x \; 10^{\,7} \; m^{-1}$.

The numerical solutions fit reasonably the experimental data. Indeed, the positions of the recirculation zones predicted by the CFD code are comparable to the ones given by the experiment. Figures 5a and 5b confirm that the SST model and the AUSM $^+$ scheme are able to capture the recirculation zones with a reasonable level of reliability.



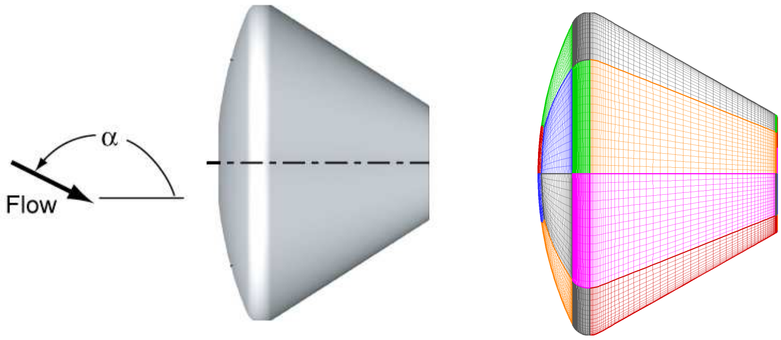
(a) $\theta = 16\,^o$.

(b) $\theta = 20\,^o$.

Figure 5: Mach contours: SST, AUSM $^+$ with entropy satisfying $a_{1/2}$, $M = 2.84$, $Re = 6.5 \; x \; 10^{\,7} \; m^{-1}$.

8. Orion CEV aerodynamic testing

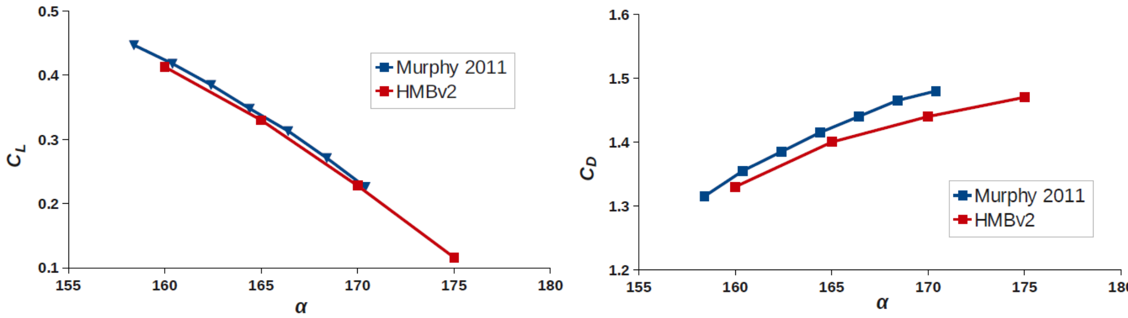
As a final test for the solver, in this section the prediction of the aerodynamic coefficients of the Orion, figures 6a-6b, are compared to the experimental results collected in [33]. Ref. [33] presented a summary of the experimental static aerodynamic data of the Orion CEV. These data were collected during the wind-tunnel test program executed at different facilities to support the development of the spacecraft.



(a) From [33]. (b)

Figure 6: Orion CEV sketch, (a), and surface grid, (b).

In this work the results for the test case at Mach 3 and Reynolds 1.5 x 10 ^{6} have been used. As it can be seen from figures 7a and 7b, the predictions given by the CFD code are in good agreement with the experimental data. Indeed, the relative differences between the numerical and the experimental results are not more than 3%.



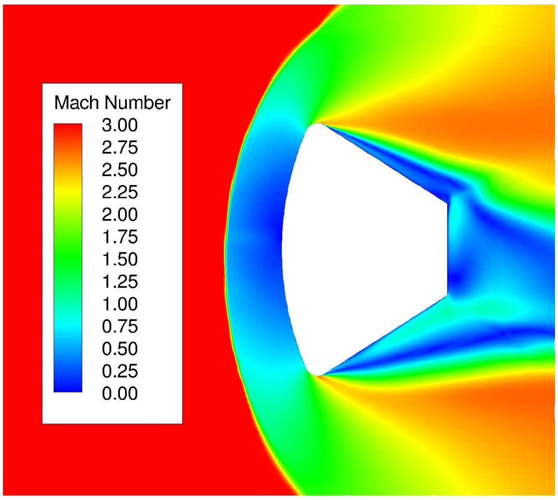
(a) (b)

Figure 7: Orion CEV C_L, (a), and C_D, (b): SST, AUSM $^+$ with entropy satisfying $a_{1/2}$, M = 3, Re = 1.5 x 10 ^{6}.

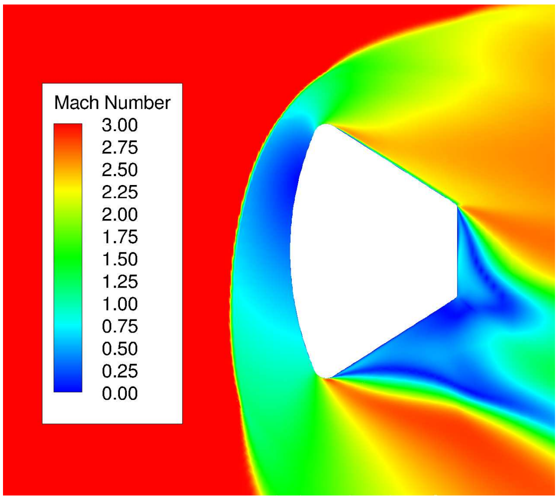
The Mach contours at two angles of attack are presented in figures 8a and 8b. It has to be highlighted that no shock instabilities have been observed during the simulations.

9. Conclusions

In the first part of this paper we presented the derivation of a fully analytical Jacobian for the AUSM $^+$ scheme. Then, the implicit scheme with the analytical Jacobian has been tested resulting to be faster than the same implicit scheme with a numerically approximate Jacobian and a 4-stage Runge-Kutta method. This is due to the higher computational efficency of evaluating an analytical Jacobian than a numerical approximation and the higher CFL number allowed by the implicit approach. Additional improvements are still possible and further investigations will be conducted to evaluate possible simplifications that can be made to the analytical Jacobian. The aim will is to improve the computational efficency of the latter without affecting the stability of the scheme. In the second part of this work, the SST turbulence model was employed together with the AUSM $^+$ scheme, to solve shock-wave/turbulent boundary-layer interactions for three different shock angles. The results, compared to experimental data, showed that the SST model has been able to capture the correct positions of the recirculation zones. Finally, the SST model and the AUSM $^+$ scheme have been employed in the prediction of some aerodynamic coefficients of the Orion spacecraft. Again, the comparison with experimental data has shown the reliability of the numerical approach.
In future works the authors will focus on hybrid continuum/kinetic Boltzmann methods for partially rarefied flow, in which the AUSM $^+$-family fluxes will represent the basis for the continuum part.



(a) α = 175 ^{o}.



(b) α = 160 ^{o}.

Figure 8: Mach contours: SST, AUSM $^+$ with entropy satisfying $a_{1/2}$, M = 3, Re = 1.5 x 10 ^{6}.

10. Acknowledgment

The financial support by the University of Liverpool is gratefully acknowledged. This work has made use of the facilities of N8 HPC provided and funded by the N8 consortium and EPSRC. The Centre is coordinated by the Universities of Leeds and Manchester.

References

[1] Lawson, S.-J., R. Steijl, M. Woodgate, and G.N. Barakos. 2012. High Performance Computing for Challenging Problems in Computational Fluid Dynamics. *Progress in Aerospace Science*, 52:19–29.

[2] Barakos, G.N., R. Steijl, A. Brocklehurst, and K. Badcock. 2005. Development of CFD Capability for Full Helicopter Engineering Analysis. In: *31st European Rotorcraft Forum*.

[3] Orkwis, P.D., and D. Scot McRae. 1992. Newton's Method Solver for High-Speed Viscous Separated Flowfields. *AIAA Journal*, 30(1):78–85.

[4] Orkwis, P.D., and D. Scot McRae. 1992. Newton's Method Solver for The axisymmetric Navier-Stokes Equations. *AIAA Journal*, 30(6):1507–1514.

[5] Spekreijse, S.P. 1987. Multigrid Solution of the Steady Euler Equations. *Ph.D. Thesis - Centrum voor Wiskunde en Informatica (Amsterdam)*.

[6] Liou, M.-S., and C.J. Steffen Jr. 1993. A New Flux Splitting Scheme. *Journal of Computational Physics*, 107:23–39.

[7] Liou, M.-S. 1996. A Sequel to AUSM: AUSM $^+$. *Journal of Computational Physics*, 129:364–382.

[8] Kitamura, K., E. Shima, and P.L. Roe. 2010. Evaluation of Euler Fluxes for Hypersonic Heating Computations. *AIAA Journal*, 48(4):763–776.

[9] Kitamura, K., E. Shima, and P.L. Roe. 2012. Carbuncle Phenomena and Other Shock Anomalies in Three Dimensions. *AIAA Journal*, 50(12):2655–2669.

[10] Darracq, D., S. Campagneux, and A. Corjon. 1999. Computation of Unsteady Turbulent Airfoils Flows with an Aeroelastic AUSM+ Implicit Solver. In: *16th AIAA Applied Aerodynamics Conference*.

[11] Van Leer, B., J.L. Thomas, P.L. Roe, and R.W. Newsome. 1987. A Comparison of Numerical Formulas for the Euler and Navier-Stokes Equations. In: *8th AIAA Computational Fluid Dynamics Conference*.

[12] Venkatakrishnan, V. 1989. Newton Solution of Inviscid and Viscous Problems. *AIAA Journal*, 27:885–891.

[13] Wigton, L.B. 1987. Application of MACSYMA and Sparse Matrix Technology to Multielement airfol Calculations. In: *8th AIAA Computational Fluid Dynamics Conference*.

[14] Venkatakrishnan, V., and T.J. Barth. 1989. Application of Direct Solvers to Unstructured Meshes for the Euler and Navier-Stokes Equations Using Upwind Schemes. In: *27th AIAA Aerospace Sciences Meeting*.

[15] Briley, W.R., and H. McDonald. 1977. Solution of the Multidimensional Compressible Navier-Stokes Equations by a Generalized Implicit Method. *Journal of Computational Physics*, 24(4):372–397.

[16] Chakravarthy, S.R. 1984. Relaxation methods for unfactored implicit upwind schemes. In: *22nd AIAA Aerospace Sciences Meeting*.

[17] Yoon, S., and A. Jameson. 1988. Lower-upper Symmetric-Gauss-Seidel Method for the Euler and Navier-Stokes Equations. *AIAA Journal*, 26(9):1025–1026.

[18] Axelsson, O. 1980. Conjugate Gradient Type Methods for Unsymmetric and inconsistent System of Linear Equation. *Linear Algebra and its Applications*, 29:1–16.

[19] Liu, Y., and C. Storey. 1991. Efficent Generalized Conjugate Gradient Algorithms, Part 1: Theory. *Journal of Optimization Theory and Applications*, 69.

[20] Hu, Y.F., and C. Storey. 1991. Efficent Generalized Conjugate Gradient Algorithms, Part 2: Implementation. *Journal of Optimization Theory and Applications*, 69.

[21] Saad, Y., and M.H. Schultz. 1986. GMRES: A Generalized Minimum Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, 7:856–869.

[22] Meijerink, J.A., and H.A. Van der Vorst. 1981. Guidelines for the Usage of Incomplete Decompositions in Solving Sets of Linear Equations as They Occur in Practical Problems. *Journal of Computational Physics*, 44(1):134–155.

[23] Orkwis, P.D. 1993. Comparison of Newton's and Quasi-Newton's Method Solvers for the Navier-Stokes Equations. *AIAA Journal*, 31(5):832–836.

[24] Venkatakrishnan, V. 1991. Preconditioned Cojugate Gradient Methods for the Compressible Navier-Stokes Equations. *AIAA Journal*, 29(7):1092–1100.

[25] Kirk, B., and G.F. Carey. 2010. Validation of Fully Implicit Parallel Finite Element Simulations of Laminar Hypersonic Flows. *AIAA Journal*, 48:1025–1036.

[26] Kirk, B., and T.A. Bova, G.F. Oliver. 2013. Validation of SUPG Finite Element Simulations of Shock-wave/Turbulent Boundary Layer Interactions in Hypersonic Flows. In: *51th AIAA Aerospace Sciences Meeting*.

[27] Vigneron, D., and J.-A. Vaassen, J.-M. Essers. 2008. An Implicit Finite Volume Method for the Solution of 3D Low Mach Number Viscous Flows Using Local Preconditioning Tehnique. *Journal of Computational and Applied Mathematics*, 215:610–617.

[28] Guo-Hao, D., and W. Wen-Long. 2011. Parallel Computation in Hypersonic Aerodynamic Heating Problem. *World Academy of Science, Engineering and Technology*, 55:190–196.

[29] Amaladas, S., and H. Kamath. 1999. Implicit and Multigrid Procedures for Steady-State Computations with Upwind Algorithms. *Computers & Fluids*, 28:187–212.

[30] Langer, S., and D. Li. 2011. Application of Point Implicit Runge-Kutta Method to Inviscid and Laminar Flow Problems using AUSM and AUSM $^+$ Upwinding. *International Journal of Computational Fluid Dynamics*, 25:255–269.

[31] Roy, C.J. and F.G. Blottner. 2006. Review and assessment of urbulence models for hypersonic flows. *Progress in Aerospace Sciences*, 42:469–530.

[32] Muck, K.-C., J. Andreopoulos, and J.-P. Dussauge. 1988. Unsteady Nature of Shock-Wave/Turbulent Boundary-Layer Interaction. *AIAA Journal*, 26(2):179–187.

[33] Murphy, K.J., K.L. Bibb, G.J. Brauckmann, M.N. Rhode, B. Owens, D.T. Chan, E.L. Walker, J.H. Bell, and T.M. Wilson. 2011. Orion Crew Module Aerodynamic Testing. In: *29th AIAA Applied Aerodynamics Conference*.

[34] Billing, F.S. 1967. Shock-Wave Shapes Around Spherical- and Cylindrical-Nosed Bodies. *Journal of Spacecraft and Rockets*, 4:822–834.