

An OpenFOAM Library for the Implicit Density-based Solution of High-Enthalpy Hypersonic Flows

F. Piscaglia^{1,†}, F. Ghioldi¹, F. Chiappari¹, F. Lentini², I. Spisso³, J. E. Guerrero³

¹*Dept. of Aerospace Science and Technology (DAER), Politecnico di Milano*

²*Dip. di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano*

³*Leonardo Spa, Genova*

[†]Corresponding author: federico.piscaglia@polimi.it

Abstract

The implementation of a time-accurate, implicit density-based solution procedure for the Navier-Stokes equations at hypersonic speeds in the form of external library modules developed in the OpenFOAM Technology is described. The strong conservation form of the governing equations makes use of primitive variables as unknowns. Physicochemical properties associated with partially ionized gases in various degrees of thermal nonequilibrium are considered. A two-temperature model is used to include the effects of thermal non-equilibrium between the translational-rotational and vibrational-electronic temperatures, respectively. The implicit density-based solution of the high-speed flow can run on CPUs, or the linear algebra problem can be optionally offloaded on GPUs. Validation tests encompass simulation of high-Mach and high-enthalpy flows typical of atmospheric re-entry and supersonic combustion of scramjet engines.

1. Introduction

The AIAA CFD2030 Integrating Committee has proposed maneuvering aircraft, turbo-machinery engines, and space launch vehicles as grand challenges for simulation of hypersonic vehicles.¹⁴ Main requirements for modern simulations include the capability to simulate a 15-20 minute trajectory by a fully coupled, unsteady solver able to account for fluid-thermal-structural interactions. Hypersonic computational fluid dynamics (CFD) are challenging because of the extreme conditions and complex physical phenomena involved.^{3,16,24,29} Hypersonic speeds, typically defined as Mach 5 and above, introduce a range of difficulties, namely: a) hypersonic flows are characterized by strong shock waves that can cause significant aerodynamic heating and pressure loads on the vehicle's surface; b) the accurate modeling of high-temperature gas dynamics; c) capturing the detailed features of hypersonic flows necessitates fine spatial resolution to accurately resolve shock waves. Small time steps are required to maintain numerical stability, prolonging the simulation time and the required computational workload. At hypersonic velocities, the kinetic energy of the flow translates into substantial thermal energy, leading to high temperatures that can cause the dissociation of air molecules and ionization. These effects result in complex chemical reactions and the formation of a plasma state, necessitating multi-species and thermochemical models for the gas and of a two-temperature model¹⁸ to include the effects of thermal non-equilibrium between the translational-rotational and vibrational-electronic temperatures.

2. Highlights

We present the implementation and validation of external solver modules that encapsulate the implicit solution of flow transport across all speed regimes. These modules can be linked to C++ libraries that extend the concept of OpenFOAM's generalized thermophysical modeling framework for multicomponent mixtures. Specifically, we couple a density-based implicit solver for high-speed flows with a thermophysical package that supports both finite-rate and equilibrium chemistry of ionized gases and non-equilibrium thermodynamics,¹³ for hypersonic flow simulations. The implemented technology includes:

- A general implementation that, inspired by the concept of block matrices in coupled solvers, constructs a single linear system of scalar equations encompassing all governing equations. By avoiding explicit block structures,

the approach allows the use of Krylov solvers originally designed for scalar non-symmetric matrices and facilitates the integration with external linear algebra libraries. This design ensures full compatibility with the non-symmetric Krylov solvers designed for scalar matrices. Also, the implementation supports any solution algorithm for single or multi-region cases (e.g., Conjugate Heat Transfer, CHT) available in OpenFOAM.^{26,27} In authors' knowledge, existing implementations of hypersonic flow solvers in the OpenFOAM technology are based on the explicit solution of the flow transport^{3,4} or on block-coupled solvers;^{6,17,21}

- the extension of the formulation of the Jacobian transformation matrix $\mathbf{J}$, to convert conservative into primary variables, to support any gas type, either perfect or real. The Jacobian matrix favors a consistent formulation of boundary conditions across either pressure- or density-based solvers;
- extensions to include non-equilibrium thermodynamics and non-equilibrium chemistry;
- the solution of the linear system for flow transport can optionally be offloaded onto NVIDIA GPUs and accelerated using the AmgX library.¹⁵ With respect to,^{12,22,31} novel algorithmic developments include: 1) the implementation of a wrapper in the form of a C++ dynamic library that directly links OpenFOAM and AmgX¹⁵ without the need for external third-party library modules as done in.²²

3. Governing Equations for High-Speed Viscous Flow Computations.

In a density-based CFD solver, the sequential solution of the governing equations for the transport of a high-speed compressible flow in a Eulerian frame of reference reads:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \quad (1)$$

$$\frac{\partial (\rho \mathbf{U})}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \mathbf{S}_U \quad (2)$$

$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\rho \mathbf{U} E) + \nabla \cdot (\mathbf{U} p) = -\nabla \cdot \mathbf{q} + \nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{U}) + \mathbf{S}_e \quad (3)$$

In Eqs. (1)-(3), the generalized form of the Newton's law of viscosity applies:

$$\boldsymbol{\tau} = \mu \left[\nabla \mathbf{U} + (\nabla \mathbf{U})^T \right] + \left(\frac{2}{3} \mu \nabla \cdot \mathbf{U} \right) \mathbf{I} \quad (4)$$

where μ is the dynamic viscosity and $\mathbf{I}$ is the identity matrix. Eqs. (1)-(3) are approximated by a general finite-volume form:

$$\frac{\partial \mathbf{W}}{\partial t} V + \sum_S (\mathbf{F}_c - \mathbf{F}_v) \cdot \mathbf{n} S = \mathbf{G}_i V \quad (5)$$

where:

$$\mathbf{W} = \begin{bmatrix} \rho \\ \rho \mathbf{U} \\ \rho E \end{bmatrix} \quad (6)$$

$$\mathbf{F}_c = \begin{bmatrix} \rho \mathbf{U} \\ (\rho \mathbf{U} \cdot \mathbf{U}) + p \\ \rho \mathbf{U} E + \mathbf{U} p \end{bmatrix} \quad (7)$$

$$\mathbf{F}_v = \begin{bmatrix} 0 \\ \boldsymbol{\tau} \\ \boldsymbol{\tau} \cdot \mathbf{U} \end{bmatrix} \quad (8)$$

$$\mathbf{G}_i = \begin{bmatrix} 0 \\ \mathbf{S}_U \\ \dot{q} + \mathbf{S}_e \end{bmatrix} \quad (9)$$

and $\mathbf{n}$ is the face normal vector, ρ is the density, $\mathbf{u}$ is the velocity, E is the total internal energy, H is the total enthalpy, p is the static pressure, $\boldsymbol{\tau}$ is the viscous stress tensor and $\mathbf{q}$ is the heat flux vector. Eqs. (5) can be written as:

$$\frac{\partial \mathbf{W}}{\partial t} V = \mathbf{R} \quad (10)$$

being $\mathbf{R}$ the residual vector:

$$\mathbf{R} = - \sum_S (\mathbf{F}_c - \mathbf{F}_v) \cdot \mathbf{n} S \quad (11)$$

In the implicit method, the residual is linearized from iteration n to $n + 1$ as:

$$\mathbf{R}^{n+1} \approx \mathbf{R}^n + \left. \frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right|_n (\mathbf{W}^{n+1} - \mathbf{W}^n) \quad (12)$$

A pseudo time-derivative of the solution⁸ with respect to a dual time τ is added to march in dual time to steady-state:

$$\frac{\partial \mathbf{W}}{\partial \tau} V + \frac{\partial \mathbf{W}}{\partial t} V = \mathbf{R} \quad (13)$$

The pseudo-time-derivative is driven to zero at each physical time level by a series of inner iterations using the implicit time-marching algorithm. While the physical time step Δt is limited only by the level of the desired temporal accuracy, the pseudo-time-step $\Delta \tau$ is determined by the CFL condition of the time-marching scheme. When steady state solutions are of interest ($\tau \rightarrow \infty$), physical time terms are dropped and only pseudo-time terms are retained; Eq. (10) then becomes:

$$\frac{\partial \mathbf{W}}{\partial \tau} V = \mathbf{R} + \mathbf{G} \quad (14)$$

At each iteration (pseudo-time step), the following linear system is solved to compute $\mathbf{W}^{n+1}$:

$$\left(\frac{V}{\Delta \tau} \mathbf{I} - \left. \frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right|_n \right) (\mathbf{W}^{n+1} - \mathbf{W}^n) = \mathbf{R} \quad (15)$$

In (15), approximation of the flux Jacobian $\left. \frac{\partial \mathbf{R}}{\partial \mathbf{W}} \right|_n$ is employed by applying the Steger-Warming flux-vector splitting scheme to divide the convective fluxes,¹⁰ that are reconstructed by the Monotonic Upstream-centered Scheme for Conservation Laws (MUSCL) scheme. TVD slope-limiters are used in the vicinity of shocks to avoid the appearance of spurious oscillations that may lead to stability issues. After the reconstruction, left and right states are used to compute the numerical flux through approximate Riemann solvers at each cell interface. Upon integrating the fluxes over all faces of the cell, as indicated by the second term in Eq. (11), the result is the numerical flux residual denoted as $\mathbf{R}$.

Boundary conditions in density-based solvers. To achieve a formulation of the boundary conditions based on the primitive variables $\mathbf{P} = [p \ U \ T]^T$ in density-based solvers (using conservative variables), it is necessary to apply a transformation to the flux Jacobian:

$$\mathbf{J} = \frac{\partial \mathbf{R}}{\partial \mathbf{W}} = \frac{\partial \mathbf{R}}{\partial \mathbf{P}} \frac{\partial \mathbf{P}}{\partial \mathbf{W}} \quad (16)$$

where the conservative variable vector is $\mathbf{W} = [\rho \ \rho U \ \rho E]^T$. The transformation necessitates the inclusion of information about gas properties in the boundary conditions. An implicit relationship can now be established between the boundary conditions and the variable values in the cells adjacent to the boundary faces. This is accomplished by applying a matrix, $\mathbf{C}$, to the flux Jacobian within the boundary cells:

$$\mathbf{J} = \mathbf{C} \frac{\partial \mathbf{R}}{\partial \mathbf{P}} \frac{\partial \mathbf{P}}{\partial \mathbf{W}} \quad (17)$$

The matrix $\mathbf{C}$ in Eq. (17) is characterized by coefficients that vary based on the specific boundary conditions applied to p, T, U . In the case of a Dirichlet boundary condition, the coefficient is set to zero, indicating its absence from the system matrix but presence solely in the source term. Conversely, for a Neumann condition, the coefficient is assigned a value of 1, as its exclusive contribution lies within the system matrix. The conservative to primitive jacobian in general form can be written as:

$$\frac{\partial \mathbf{P}}{\partial \mathbf{W}} = \begin{bmatrix} \left. \frac{\partial p}{\partial \rho} \right|_T & \left. \frac{\partial p}{\partial (\rho U)} \right|_T & \left. \frac{\partial p}{\partial (\rho E)} \right|_T \\ \frac{\partial U}{\partial \rho} & \frac{\partial U}{\partial (\rho U)} & \frac{\partial U}{\partial (\rho E)} \\ \left. \frac{\partial T}{\partial \rho} \right|_p & \left. \frac{\partial T}{\partial (\rho U)} \right|_p & \left. \frac{\partial T}{\partial (\rho E)} \right|_p \end{bmatrix} \quad (18)$$

For a real fluid, Eq. (18) is written as:

$$\left. \frac{\partial \mathbf{P}}{\partial \mathbf{W}} \right|_{\text{real}} = \begin{bmatrix} \frac{\rho h_T + \rho_T(H - |U|^2)}{a_1} & \frac{\rho_T u_x}{a_1} & \frac{\rho_T u_y}{a_1} & \frac{\rho_T u_z}{a_1} & -\frac{\rho_T}{a_1} \\ -\frac{u_x}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ -\frac{u_y}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ -\frac{u_z}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 \\ \frac{1 - \rho_p(H - |U|^2) - \rho h_p}{a_1} & -\frac{\rho_p u_x}{a_1} & -\frac{\rho_p u_y}{a_1} & -\frac{\rho_p u_z}{a_1} & \frac{\rho_p}{a_1} \end{bmatrix} \quad (19)$$

where

$$|U|^2 = u_x^2 + u_y^2 + u_z^2 \quad (20)$$

$$a_1 = \rho \rho_p h_T + \rho_T(1 - \rho h_p) \quad (21)$$

$$\rho_p = \left. \frac{\partial \rho}{\partial p} \right|_T \quad (22)$$

$$\rho_T = \left. \frac{\partial \rho}{\partial T} \right|_p \quad (23)$$

$$h_p = \frac{1 - \alpha_T T}{\rho} \quad (24)$$

$$h_T = C_p \quad (25)$$

and α_T is the compressibility coefficient at constant temperature.

4. Nonequilibrium Thermodynamics

A set of constitutive relations for hypersonic flows is provided to achieve the closure of the system. For temperatures $T < 20000$ K, thermodynamic properties of pure species and mixture quantities, such as enthalpy, entropy, and specific heats, are obtained either from JANAF polynomial fits or computed using the Rigid Rotor-Harmonic Oscillator (RRHO) model. The Gupta-Yos curve-fitted temperature function⁵ is used to model transport properties for high-temperature air mixtures:

$$\mu_i = e^{[(A_\mu \ln T + B_\mu) \ln T + C_\mu]} \quad i = 1 \dots N \quad (26)$$

where A_μ , B_μ and C_μ are constants determined for each species. Thermochemical data for the gas phase are imported from the Cantera transport database or from publicly available mechanism.² Thermodynamic properties for the i -th gaseous species are then computed similarly to NASA chemical equilibrium codes. Finally, mixture thermodynamic quantities (viscosity μ , thermal conductivity k and diffusion) are calculated by the Wilke's mixing rule,³⁰ starting from the properties of the single transported specie.

5. Transport of Chemical Species in a Hypersonic Flows

In addition to the governing equations for the fluid flow, a set of $N_s - 1$ convection-diffusion partial differential equations (PDEs) is solved to determine the local mass fraction Y_i of each of the N_s chemical species transported by the reactive fluid mixture in each Control Volume (CV):

$$\frac{\partial (\rho Y_i)}{\partial t} + \nabla \cdot (\rho \mathbf{U}) Y_i = \nabla \cdot (\rho D_i \nabla Y_i) + \bar{\omega}_i \quad \text{for } i \in [1, N_s - 1] \quad (27)$$

The mass fraction of the inert species N_s is determined as:

$$Y_{N_s} = 1 - \sum_{i=1}^{N_s-1} Y_i \quad (28)$$

In Eq. (27), D_i is the mass diffusion coefficient; in reactive simulations, $\bar{\omega}_i$ is defined as:

$$\bar{\omega}_i = K_i \dot{\omega}_i \quad (29)$$

where the reaction rate of the i -th specie $\dot{\omega}_i$

$$\dot{\omega}_i = W_i \sum_{j=1}^{N_R} \nu_{i,j} Q_j \quad (30)$$

is scaled by a specific set of coefficients K_i depending on the selected turbulence-chemistry interaction model, to account eventually for the interaction between turbulent mixing and chemistry in the CV. If the laminar finite-rate model is used, $K_i=1$ in Eq. (29). In Eq. (30), W_i is the molecular weight of the i -th species; $\nu_{i,j}$ is the i -th species stoichiometric coefficient, Q_j is the non-equilibrium reaction rate of the j -th reaction:

$$Q_j = \kappa_{f,j}(T, p) \prod_{i \in P} \left(\frac{\rho Y_i}{W_i} \right)^{\nu'_{i,j}} - \kappa_{r,j}(T, p) \prod_{i \in R} \left(\frac{\rho Y_i}{W_i} \right)^{\nu''_{i,j}} \quad (31)$$

In Eq. (31), $\kappa_{f,j}(T, p)$ and $\kappa_{r,j}(T, p)$ are the forward and reverse rate constants at the local fluid dynamic conditions, while the ratio $\frac{\rho Y_i}{W_i}$ is the molar concentration of the i -th species. The heat released by the chemistry $\dot{Q}$ is finally obtained as:

$$\dot{Q} = \sum_{i=1}^{N_s} (\bar{\omega}_i H_{f,i}) \quad (32)$$

where H_f is the enthalpy of formation. The mechanism for ionized air proposed by Park (11 species, 22 reactions)^{18,19} is used in the simulations and detailed in the table provided in Appendix A. Chemical mechanisms for ionized air can be specified as input through tables in the input files, enabling users to select the most appropriate mechanism based on the operating conditions.

To account for the vibrational excitation to lag behind translational/rotational excitation (due to slower relaxation) and that chemical reactions (dissociation, ionization) occur in a non-equilibrium manner, a two-temperature model²⁰ is used.

The model computes the conservation of vibrational/electronic energy

$$\frac{\partial(\rho e_v)}{\partial t} + \nabla \cdot (\rho e_v \mathbf{u}) = -\nabla \cdot \mathbf{q}_v + \tau_v + \dot{Q}_{v-tr} \quad (33)$$

where the vibration-translational energy exchange in the Landau-Teller reads:

$$\dot{Q}_{v-tr} = \sum_s \rho_s \frac{e_{v,s}^{eq}(T) - e_{v,s}(T_v)}{\tau_{v,s}} \quad (34)$$

and

$$\tau_{v,s} = \frac{1}{p} \exp \left(A_s \left(\frac{T}{1000} \right)^{-1/3} - B_s \right) \quad (35)$$

being ρ_s the density of the specie s , e_v the vibrational/electronic energy per unit mass, $\vec{q}_v$ the vibrational/electronic heat flux, $\dot{Q}_{v-tr}$ the vibrational-translational energy exchange rate (source term); τ_v is viscous dissipation related to vibrational modes, whose contribution is often negligible.

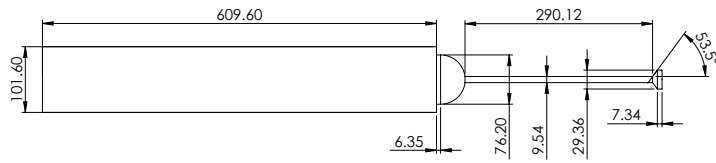
6. Validation Test Cases: results

Following,^{9,14,25} test cases were selected to facilitate the implementation of "nose-to-tail" computations⁹ for the forebody, inlet, combustor, and nozzle of a hypersonic vehicle. To this end, the implemented framework for hypersonic flows was assessed using geometric models from the literature, for which experimental data are available. Two test cases are discussed in this study, that are representative of high-Mach and high-enthalpy flows, characteristic of destructive atmospheric re-entry. These cases are described in detail in the following sections. CFD simulations were performed using polyhedral meshes, which offer enhanced numerical stability and accuracy for complex geometries compared to traditional mesh types. These meshes provide improved resolution of flow features while maintaining computational efficiency, making them particularly suitable for the high-fidelity modeling required in this work.

6.1 Aerospiked Missile at Mach 6

The geometry and the operating conditions of the spiked blunt body configuration (aerospike-nosed missile) operating at Mach 6⁷ is reported in Fig. 1; a sting is mounted on the nose of the missile to offset the shockwave away from the main body. That promotes a reduction of the drag, whereas relieving the thermal and the structural loads sensed by the vehicle. The diameter of the core body of the missile is $D = 101.60$ mm; its length is $L = 609.60$ mm (6 times the diameter); the nose, upon an offset of 6.35 mm, is a hemisphere of diameter $d = 76.20$ mm. The hemisphere is truncated at the top to accommodate the sting. The cylindrical sting has a diameter $d_s = 9.54$ mm, a length $l = 290.12$ mm and an inverted truncated cone head. The semi-cone angle of the head is 53.5° as proposed in Fig. 1.

The aerospike operates in a freestream where $p = 1951$ Pa and $T = 58.25$ K; the imposed air velocity is $u = 927$ m/s aligned with the sting direction. Hence, the Mach number is $Ma = 6.06$.



Parameter	Value
Ma	6.06
p_s	1951 Pa
T_s	58.25 K
AoA	0, 10 deg
fluid	air (non-reacting)

Figure 1: Geometry of the aerospike-nosed missile (dimensions are in mm) and operating conditions. Only the configuration at Angle of Attack (AoA) = 0 deg is presented in this work.

The problem is radially symmetric. The employed computational grid is a three-dimensional polyhedral mesh shaped as half-truncated cone. The two diameters are 0.48 m and 1.36 m, respectively; its height is 1.125 m. Five inflation layers are extruded along the walls of the geometry to accurately capture the boundary layer. The first layer height is 0.03 mm; an expansion ratio allows to progressively reach a total thickness of 0.18 mm. Refinement zones are applied on top of the background mesh to better capture the complex shock pattern. The overall number of cells in the studied domain is 5 M. Fig. 2 includes a 3D view of the computational domain.

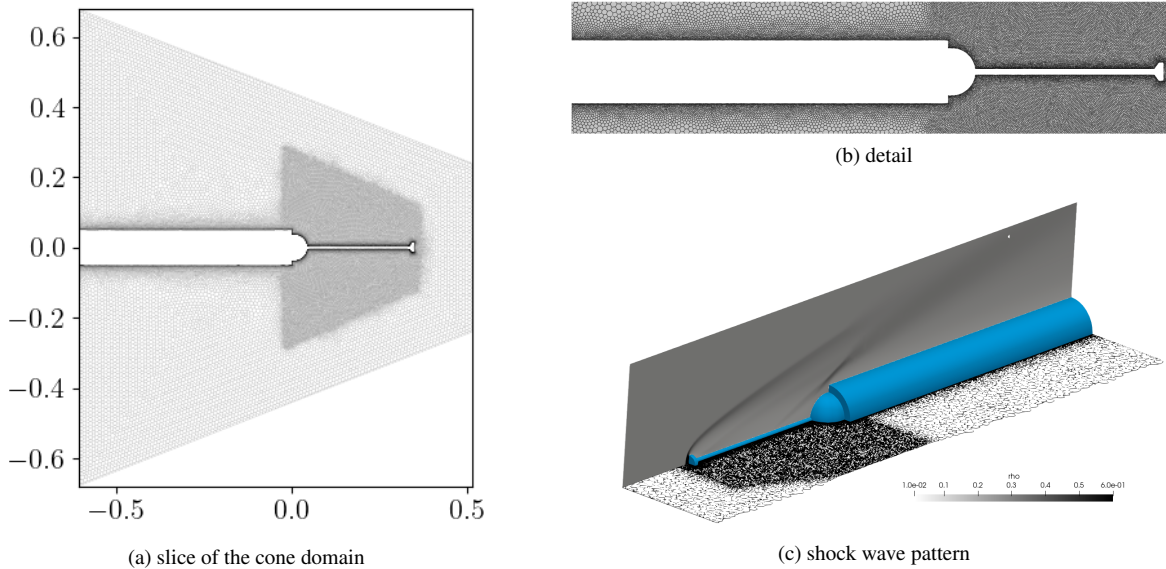


Figure 2: Detail of the polyhedral mesh used to solve the aerospike-nosed missile.

In the simulation, the external sides have a freestream-type boundary condition; that is not the case for the symmetry plane, where a symmetry BC is enforced accordingly. Non-reflecting boundary conditions are set at the outlet. The body walls are treated using wall-corrected boundary conditions. The RANS $k-\omega$ SST turbulence model is selected, as proposed in,²⁸ and the working fluid is air intended as perfect gas.

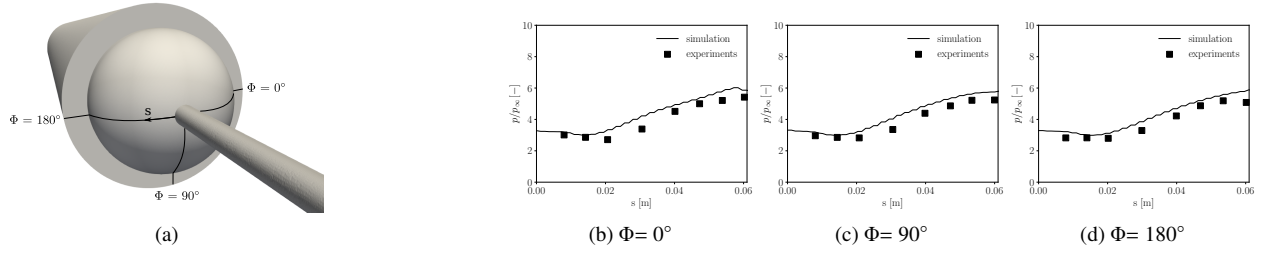


Figure 3: Aerospike-nosed missile: a) Location of the probes used for experimental measurements; b) Validation against experimental data.

The pressure distribution is measured using pressure orifices sensors mounted on top of the hemispherical dome.⁷ The position of the pressure probes is shown in Fig. 3a. A comparison between the experimental and numerical results is proposed in Fig. 3. The solution at the three locations looks symmetrical as expected, in face of a 0° angle of attack (AoA). The flow field surrounding the aerospike-nosed missile is visualized and compared against experimentally acquired Schlieren images.⁷ In Fig. 4b, the numerical solution of the complex shock wave pattern is compared against an experimental visualization.

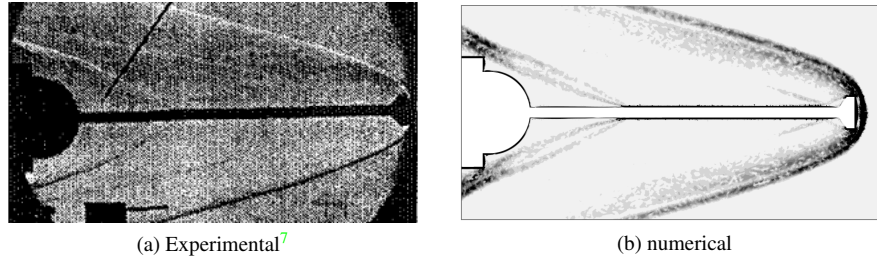


Figure 4: Shock wave pattern around the aerospike geometry at $Ma = 6.06$.

The aerospike generates a bow shock close to the end tip of the spike: there, the flow experiences strong pressure and density gradients. Inside the region delimited by the bow shock, the numerical solution well captures the “post disk compression” region,⁷ that is the zone confined into two weaker shock waves near the disk of the aerospike. Towards the hemispherical dome, pressure and density remain almost constant, until the flow separation shock is reached. Behind the separation shock, the flow experiences strong recirculation and a sudden temperature increment. That can be appreciated via Fig. 5a and 5b.

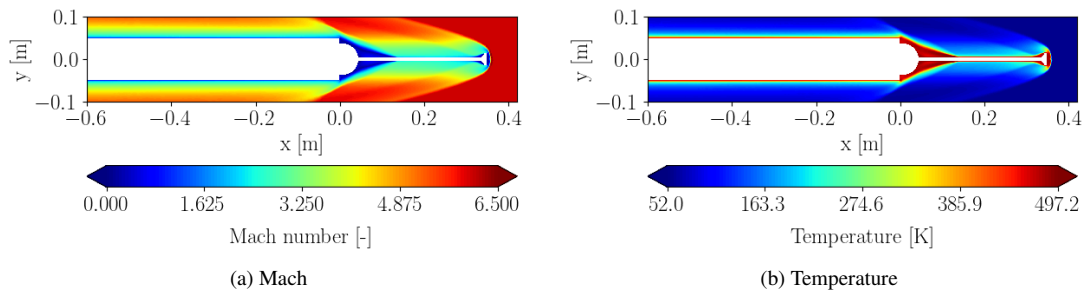
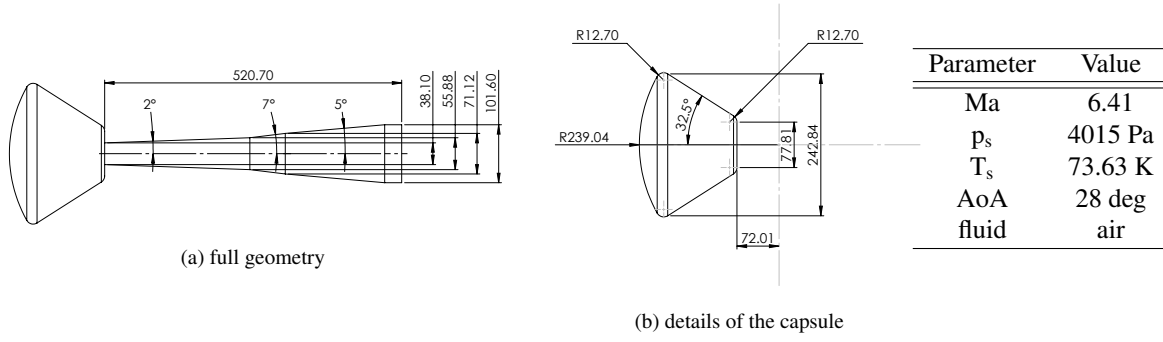


Figure 5: Aerospike-nosed missile. Contour plots on the symmetry plane of: a) flow velocity; b) temperature field.

6.2 Orion Multi-Purpose Crew Vehicle (MPCV) atmospheric re-entry

The Orion Multi-Purpose Crew Vehicle (MPCV), referred to here as the Orion capsule, is a partially reusable spacecraft developed by NASA to support missions under the Artemis program. A view of the geometrical sketch is proposed in Fig. 6. The entire structure consists of the capsule and a mounting rod, which were also used during the experimental campaigns.¹¹ The rod is composed of truncated cones, with an overall length $L = 520.70$ mm. The smallest and largest

Figure 6: Geometry of the Orion capsule.¹¹

circular cross-sections have diameters of 31.80 mm and 101.60 mm, respectively. The capsule is mounted at the top of the rod, with its geometric details shown in Fig. 6b. The simulation domain consists of a cylindrical body with a top hemisphere surrounding the capsule and its mounting rod. The external cylinder has a diameter of $D = 35$ cm and a height $H = 65$ cm. In the domain, the geometry is vertically offset by 8 cm because the flow direction is not aligned with the cylinder axis in this investigation. This vertical offset allows for a detailed analysis of the flow wake, where complex flow structures may develop.

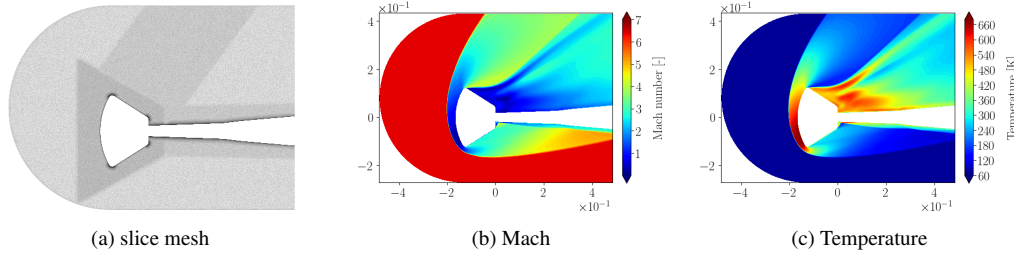


Figure 7: Two-dimensional contour plots of temperature and velocity on the mid-plane of the Orion capsule.

The three-dimensional computational grid is made of 46 million polyhedrons (Fig. 7-a). Five inflation layers are extruded along the walls of the geometry to accurately capture the boundary layer. The height of the first layer measures $5 \mu\text{m}$; a total thickness equal to 0.03 mm is achieved thanks to a prescribed expansion ratio. The mesh also features three different refinement zones: they force the maximum cell dimension $\Delta_x = 1.5$ mm. The refinements are intended to capture more accurately: a) the bow shock position and orientation; b) the fluid dynamic behavior in proximity of the capsule and the mounting rod. The first refinement is shaped as truncated cone (base diameters are $d_{r,1} = 8$ cm and $D_{r,1} = 24$ cm) whose height is 29 cm. A cylindrical refinement box ($d_{r,2} = 8$ cm) is placed around the boom. The third cylindrical-shape refinement is used to better capture the wake zone ($d_{r,3} = 15$ cm). A smooth transition is enforced among the refined regions and the unrefined mesh regions.

The case operates in freestream air with conditions $p = 4015$ Pa and $T = 73.63$ K. Subjected to a velocity of $u = 1100$ m/s, the flow corresponds to a Mach number of $\text{Ma} = 6.41$. The velocity is inclined with an angle of attack $\text{AoA} = 28^\circ$. Freestream-type boundary conditions are applied to the external surface of the cylinder. The capsule and boom surfaces are maintained at a constant temperature of $T_w = 295.56$ K, simulating the capsule cooling system, as suggested in.¹ Turbulence is modeled using the RANS $k-\omega$ -SST approach. The working fluid, air, is treated as a perfect gas. Fig. 7 clearly shows the inclined bow shock induced by the presence of the Orion capsule body. As expected, the shock is detached. Behind the shock, close to the capsule, the flow is subsonic. The temperature increment is sensible and the thermal analysis of that region - particularly of the capsule bottom surface - becomes of predominant interest. The Mach progressively increases behind the bow, far away from the capsule, making the bow shock behave like a weak shock system. In the upper part of the capsule, because of the capsule's curved end, the flow accelerates; a shear layer detaches from the capsule's upper surface; a new complex shock system is formed. The region extending from just downstream of the capsule's afterbody to the shear layer is of particular interest: the flow is subsonic and highly turbulent. That complicates accurate heat flux estimations. A comprehensive investigation of the heat flux on the afterbody is beyond the scope of this work.

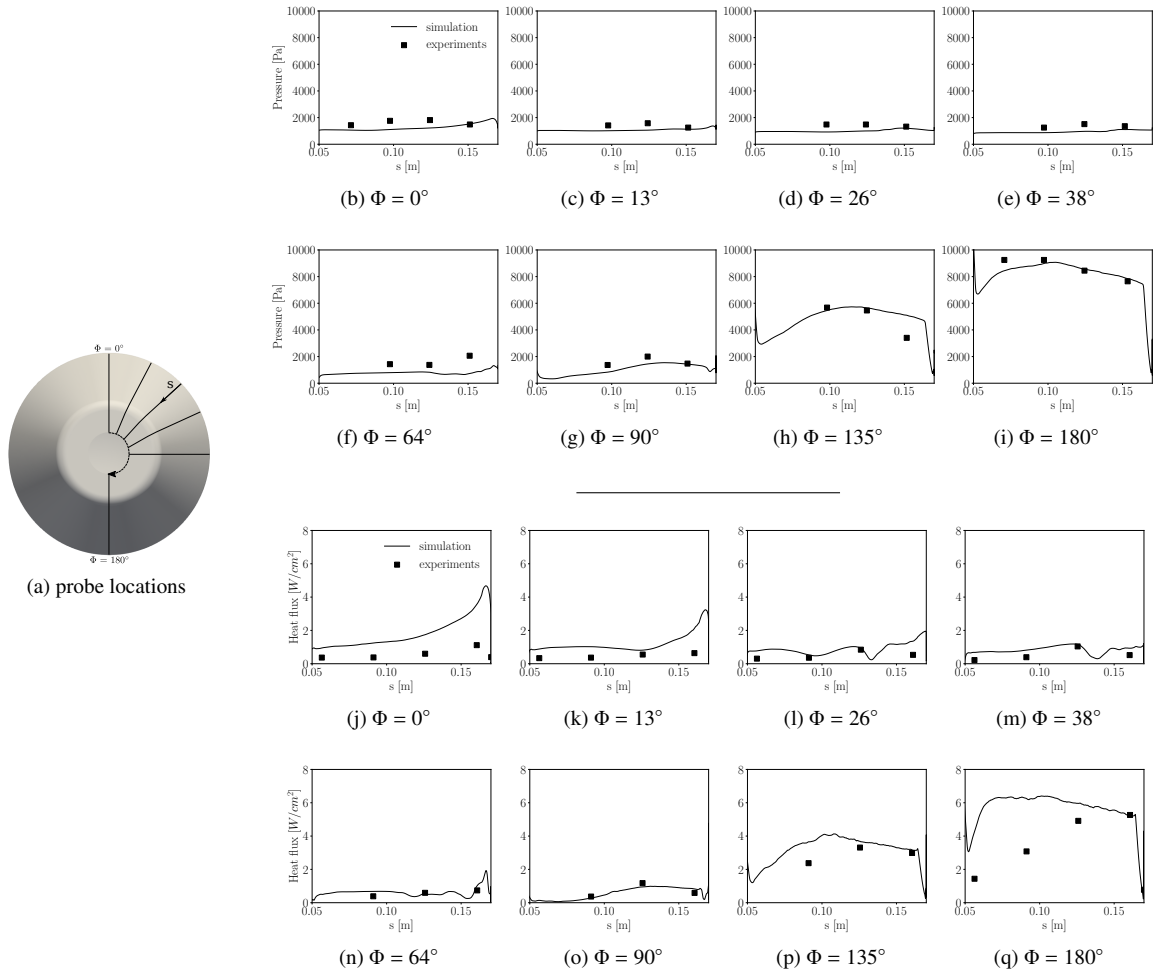


Figure 8: Comparison between simulations and experiments of pressure gauges (b-i)-and wall heat flux (j-q) at various angles on the capsule's afterbody (a).

Experimental measurements of pressure gauges and wall heat flux are available from,²³ along the capsule's afterbody, at various angles, as reported in Fig. 8. In the same figure, a comparison between simulations and experiments is reported. The wall pressure is slightly underestimated by probes at $\phi = 0^\circ$ to 90° , located in the wake region; probes at $\phi = 135^\circ$ and 160° show less accuracy in capturing the pressure trend. In contrast, probes at $\phi = 180^\circ$, positioned in undisturbed flow, align well with the experimental data.

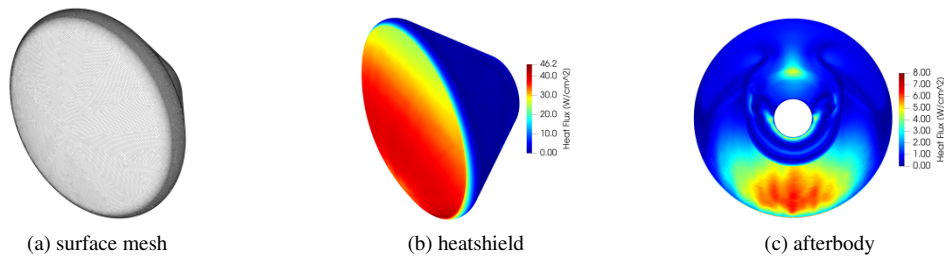


Figure 9: Orion capsule: heat flux distribution on the capsule surfaces.

Fig. 8-j to q present results for $\phi = 26^\circ$ to 90° . The agreement deteriorates in the lower region, where the computational grid is coarser. Ongoing analysis, focused on heat transfer modeling, aims to better align the simulation with experimental findings. Finally, Fig. 9 illustrates the heat flux contours for both the heat shield and the afterbody.

7. Conclusions

Numerical results demonstrate that the developed implicit density-based solver is both robust and efficient, effectively capturing the critical features of the flowfield. The solver, implemented as a module within the OpenFOAM framework, is integrated with libraries that compute thermochemical dissociation effects using a laminar finite-rate formulation for reaction modeling. The accuracy of the results for the test cases is in good agreement with findings reported in the literature. Algorithmic developments are integrated into external library modules within the OpenFOAM Technology, developed at the Department of Aerospace Science and Technology (DAER), Politecnico di Milano.

8. Acknowledgments

The work presented in this paper has been partially funded by the NextGenerationEU European initiative through the Italian Ministry of University and Research.

References

- [1] AB Blair Jr, Jerry M Allen, and Gloria Hernandez. Effect of tail-fin span on stability and control characteristics of a canard-controlled missile at supersonic mach numbers. Technical report, NASA, 1983. 19830019688.
- [2] Caltech. Chemical reaction and thermodynamic data. [\[link\]](#).
- [3] Vincent Casseau, Daniel E. R. Espinoza, Thomas J. Scanlon, and Richard E. Brown. A two-temperature open-source cfd model for hypersonic reacting flows, part two: Multi-dimensional analysis. *Aerospace*, 3(4), 2016.
- [4] F. Ghioldi and F. Piscaglia. Acceleration of supersonic/hypersonic reactive cfd simulations via heterogeneous cpu-gpu supercomputing. *Computers & Fluids*, 266:106041, 2023.
- [5] R. N. Gupta, J. M. Yos, R. A. Thompson, and K. P. Lee. A review of reaction rates and thermodynamic and transport properties for an 11-species air model. Technical Report NASA RP-1232, NASA, Washington, D.C., 1990.
- [6] Johan Heyns and Oliver Oxtoby. Modelling High-speed Viscous Flows in OpenFOAM. In *9th South African Conference on Computational and Applied Mechanics, SACAM 2014*, page 8, 1 2014.
- [7] Lawrence Huebner, Anthony Mitchell, and Ellis Boudreaux. Experimental results on the feasibility of an aerospoke for hypersonic missiles. In *33rd Aerospace sciences meeting and exhibit*, page 737, 1995.
- [8] A. Jameson, W. Schmidt, and E. Turkel. Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes. *AIAA paper*, 1259, 01 1981.
- [9] Joseph G. Marvin. A CFD Validation Roadmap For Hypersonic Flows. Technical Memorandum NASA/TM-1992-103935, NASA, April 2022.
- [10] Hong Luo, Joseph D. Baum, and Rainald Löhner. A fast, matrix-free implicit method for compressible flows on unstructured grids. *Journal of Computational Physics*, 146(2):664–690, 1998.
- [11] Matthew MacLean, Erik Mundy, Timothy Wadhams, Michael Holden, Michael Barnhardt, and Graham Candler. Experimental and numerical study of laminar and turbulent base flow on a spherical capsule. In *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, page 783, 2009.
- [12] Matt Martineau, Stan Posey, and Filippo Spiga. Amgx gpu solver developments for openfoam. In *Proceedings of the 8th OpenFOAM Conference, Virtual*, pages 13–15, 2020.
- [13] B. J. McBride, M. J. Zehe, and S. Gordon. NASA Glenn Coefficients for Calculating Thermodynamic Properties of Individual Species. *NASA/TP-2002-211556*, 2022.
- [14] Scott A. Morton and Russell M. Cummings. Cfd 2030: Hypersonic modeling & simulation grand challenge. In *AIAA SCITECH 2024 Forum*, page 8, 2024.
- [15] M. Naumov, M. Arsaev, P. Castonguay, J. Cohen, J. Demouth, J. Eaton, S. Layton, N. Markovskiy, I. Reguly, N. Sakharnykh, V. Sellappan, and R. Strzodka. Amgx: A library for gpu accelerated algebraic multigrid and preconditioned iterative methods. *SIAM Journal on Scientific Computing*, 37(5):S602–S626, 2015.

- [16] Adam Norman, Valerio Viti, Keigan Maclean, and Varun Chitta. Improved CFD methodology for compressible and hypersonic flows using a Hessian-based adaption criteria. In *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, page 10, 2022.
- [17] Stefano Oliani, Ettore Fadiga, Ivan Spisso, Luigi Capone, and Federico Piscaglia. GPU-accelerated Linear Algebra for Coupled Solvers in Industrial CFD Applications with OpenFOAM, 2024.
- [18] Chul Park. Review of chemical-kinetic problems of future NASA missions. I - Earth entries. *Journal of Thermophysics and Heat Transfer*, 7(3):385–398, 1993.
- [19] Chul Park, Richard L Jaffe, and Harry Partridge. Chemical-kinetic parameters of hyperbolic earth entry. *Journal of Thermophysics and Heat transfer*, 15(1):76–90, 2001.
- [20] Park, C. Assessment of a two-temperature kinetic model for dissociating and weakly ionizing Nitrogen. *Journal of Thermophysics and Heat Transfer*, pages 8–16, 1988.
- [21] Federico Piscaglia, Federico Ghioldi, Ivan Spisso, and Joel Enrique Guerrero Rivas. *An Implicit Density-Based GPU Coupled Solver for Hypersonic Flow Computations in OpenFOAM*.
- [22] Piscaglia, F. and F. Ghioldi. GPU Acceleration of CFD Simulations in OpenFOAM. *Aerospace*, 10(9), 2023.
- [23] Roberto Roveda. Benchmark CFD study of spiked blunt body configurations. In *47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 367, 2009.
- [24] Jian-Shun Shuen, Kuo-Huey Chen, and Yunho Choi. A coupled implicit method for chemical non-equilibrium flows at all speeds. *Journal of Computational Physics*, 106(2):306–318, 1993.
- [25] Slotnick, J. and Khodadoust, A. and Alonso, J. and Darmofal, D. and Gropp, W. and Lurie, E. and Mavriplis, D. CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences. Technical Memorandum NASA/CR-2014-218178, NASA, Langley Research Center Prepared, Hampton, Virginia 23681-219, USA, March 2014.
- [26] The OpenFOAM®Foundation. OpenFOAM 12. <https://openfoam.org/version/12/>, 2024.
- [27] The OpenFOAM®Foundation. OpenFOAM-dev. <http://www.openfoam.org/dev.php>, 2024.
- [28] Tessa Uroić, Hrvoje Jasak, and Henrik Rusche. Implicitly coupled pressure–velocity solver. In *OpenFOAM®: Selected Papers of the 11th Workshop*, pages 249–267. Springer, 2019.
- [29] V. Viti, V. Rao, and J. Abanto. CFD simulations of super/hypersonic missiles: validation, sensitivity analysis and improved design. In *AIAA Scitech 2020 Forum*, page 10, 2020. doi.org/10.2514/6.2020-2123.
- [30] C. R. Wilke. A viscosity equation for gas mixtures. *The Journal of Chemical Physics*, 18(4):517–519, 04 1950.
- [31] Zampini, S. and Bnà, S. and Valentini, M. and I. Spisso. GPU accelerated OpenFOAM simulations with PETSc4FOAM. In *Proceedings of the 8th OpenFOAM Conference, Virtual*, pages 13–15, 2020.