



Multi-fidelity aerodynamic database construction for atmospheric entry spacecrafts

Arda Ozuzun¹, Michele Capriati¹, Domenico Massari¹, Pierre Schrooyen¹, Guillaume Grossir¹

Abstract

The prediction of hypersonic flight vehicle trajectory depends on having an accurate aerodynamic database across the flight envelope. Numerical simulation to evaluate the entire database is computationally intensive hence the need to have a multi-fidelity framework to efficiently and accurately compute the aerodynamic coefficients. A Multi-Task Gaussian Process is used to fuse low-fidelity data, stemming from Newtonian theory, and high-fidelity data, obtained by simulating the flow around the object by means of CFD. Accurate computational grids for CFD are obtained by employing a shock-fitting technique. This is achieved by employing an in-house library HEROES, which is also described. The full framework is used to construct the aerodynamic database for the DRACO capsule. Results show that the use of a multi-fidelity strategy largely increases the accuracy of the surrogate model when limited high-fidelity data are available.

Keywords: *Hypersonic, aerodynamic database, multi-fidelity, meshing*

Nomenclature

Latin

AoA – Angle of attack

CFD – Computational Fluid Dynamics

C_L – Lift coefficient

C_D – Drag coefficient

C_M – Pitching moment coefficient

f – Task

GPR – Gaussian Process Regression

k – Kernel

MT – Multi-Task

MNM – Modified Newtonian Method

NRMSE – Normalized root mean square error

p – Pressure

q – Dynamic pressure

T – Temperature

x – Input variable

y – Output variable

1. Introduction

Assessing the performance of a capsule (re-)entering the atmosphere requires analyzing several aspects: its trajectory, its stability, and its thermal response, to name a few. These aspects can be investigated by generating an aerothermodynamic database. It includes the aerodynamic forces and the thermal loads at selected free-stream Mach numbers, altitudes, and attitudes. The numerical realm offers different approaches for the computation of such quantities, spanning different levels of fidelity. Low-fidelity models, e.g., based on the Newtonian theory, allow for covering the range of desired conditions at a low computational cost. They are preferred when one is mostly interested in the efficiency of the computations. However, the accuracy of the predictions remains questionable, especially considering certain geometry or freestream conditions. To guarantee higher accuracy, the flow around the capsule can be simulated by means of CFD solvers. Yet, CFD computations dramatically increase the computational cost associated with the construction of the database. Furthermore, the accuracy of the results is largely influenced by the mesh quality, and a rigorous (and time-consuming) grid-independence study should be performed. An intermediate solution is to use surrogate models, e.g. Kriging or artificial neural network, to limit the computational cost, while preserving accuracy. Surrogate models are mathematical approximations of the CFD response and are trained on a relatively small set of simulations. Thus, those reduce the overall cost associated with the aerodynamic database construction to the one needed

¹*von Karman Institute for Fluid Dynamics, Waterloosesteenweg 72, 1640 Sint-Genesius-Rode, BE*

to train the surrogate model. Nevertheless, when dealing with complex 3D simulations, such as those around a capsule traveling at hypersonic speed, accurately training a surrogate model on high-fidelity points can still be very computationally expensive. In these cases, multi-fidelity strategies, which merge low- and high-fidelity information, become very attractive to enhance the efficiency.

In the hypersonic community, different studies explored how to combine high- and low-fidelity data [22, 24, 23, 10, 19]. For example, Quinlan et al. [22] proposed an adaptive/multi-fidelity approach for aerodynamic database generation. In this study, a recursive cokriging method is employed to model the lift and drag coefficients of a blunted cone geometry. A comparison of Leave-One-Out Prediction Errors between the multi-fidelity and single high-fidelity models shows that the multi-fidelity approach achieves improved predictive accuracy. This indicates that lower-fidelity data contributes significantly to the model, providing a computationally efficient alternative instead of adding new high-fidelity data.

The predictive capability of a surrogate model depends upon the accuracy of the high-fidelity data used for its training. Aside from physical modeling aspects, the accuracy of conventional 2nd-order upwind finite-volume formulations for hypersonic flows largely depends upon the quality of the mesh. The best practice is to discretize the shock layer with hexahedral elements aligned with the shock and stretched tangentially. Furthermore, normal-to-the-wall hexahedra are desired to well capture the boundary layer developing in front of the object. This translates into large man-hours spent in constructing adequate computational grids. This problem is critical when employing CFD to build an aerothermodynamic database. In fact, the conditions to be simulated should account for different altitudes and attitudes of the spacecraft, hence requiring updating the grid for each configuration. In this case, it would be desirable to automate the meshing process, for example, using techniques such shock-fitting [21, 25] and mesh-adaptation [5, 6, 14, 15].

The goal of this work is twofold. First, we present a multi-fidelity framework that allows for accurately and efficiently generating a surrogate-based aerodynamic database. Secondly, we present a methodology to automatically generate high-quality meshes for hypersonic computations. Such a methodology is included in the Hybrid mEsh adapter fOr hypErSonic (HEROES) library [9], developed at the von Karman Institute. It automatically I) generates high-quality meshes, II) runs CFD computations, and III) post-processes the solution. Such a library, coupled to the CFD++ solver [1], is used to generate the high-fidelity data used to train the model. The full methodology is applied to create the aerodynamic database for the DRACO capsule [20]. It predicts lift, drag, and moment coefficients along the capsule's re-entry trajectory at different Mach numbers and angles of attack. Low-fidelity training points are generated by means of the ANTARES code [13]. Low- and high-fidelity data are fused using a Multi-Task Gaussian Process.

This paper is structured as follows: the main ingredients to generate the multi-fidelity aerodynamic database are presented in Sec. 2, the HEROES library is detailed in Sec. 3, results are shown in Sec. 4, and conclusion and future perspective are given in Sec. 5.

2. Aerothermodynamic database generation

The methodology proposed for the generation of the aerothermodynamics database is sketched in Fig. 1. First, the multi-fidelity surrogate model is trained using both low- and high-fidelity data. A Multi-Task Gaussian Process Regression, described in Sec. 2.1, is used to this end. Low-fidelity data are obtained with the ANTARES code, described in Sec. 2.2, while high-fidelity data with the CFD++ solver, introduced in Sec. 2.3. Accurate computational grids are generated using HEROES, detailed in Sec. 3. Once the surrogate model is trained, its predictive capabilities are verified against independent high-fidelity points. Finally, the trained and verified surrogate model can be used for the generation of the aerothermodynamic database.

2.1. Multi-Task Gaussian Process Regression

Multi-Task Gaussian Process Regression (MT-GPR) extends the Gaussian Process Regression (GPR) framework to model multiple related tasks jointly. No hierarchy or structure among the tasks is as-

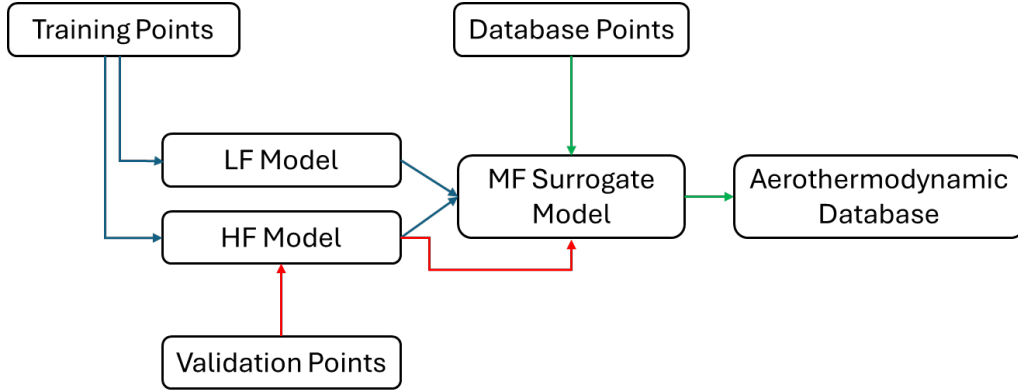


Fig 1. Methodology for the generation of the aerothermodynamics database.

sumed. The task correlations is captured through a task covariance matrix. This results in a free-form model capable of learning from partially observed tasks. For simplicity, we consider the case of two tasks, $f_1(x)$ and $f_2(x)$, defined over a common input space. The joint covariance between function values from the two tasks is modeled as:

$$\text{Cov}(f_i(x), f_j(x')) = K_f(i, j) \cdot k_x(x, x'), \quad (1)$$

where $k_x(x, x')$ is the input kernel shared across all tasks, and $K_f(i, j)$ is the task kernel term that quantifies how similar task i is to task j . The full joint distribution over the outputs y_1 and y_2 from the two tasks is given by:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \sim \mathcal{N} \left(0, \begin{bmatrix} K_f(1, 1)K_x & K_f(1, 2)K_x \\ K_f(2, 1)K_x & K_f(2, 2)K_x \end{bmatrix} \right), \quad (2)$$

where K_x is the covariance matrix computed over the input points using the kernel function $k_x(x, x')$, and each block corresponds to the covariance within and between tasks. The model includes two types of hyperparameters: those of the input kernel, which control smoothness over the input space, and the entries of the task covariance matrix K_f , which quantify task similarity. A common approach is to represent K_f using a low-rank decomposition $K_f = BB^\top$, where the number of columns in B determines the rank. For example, in the rank-1 case for two tasks, K_f is defined as:

$$B = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \Rightarrow K_f = \begin{bmatrix} b_1^2 & b_1 b_2 \\ b_1 b_2 & b_2^2 \end{bmatrix}. \quad (3)$$

The hyperparameters of $k_x(x, x')$ and B are learned jointly by maximizing the marginal log-likelihood. Predictions are made after training by conditioning the joint multivariate normal distribution over all tasks and inputs on the observed data. This enables each task to benefit from observations of the others, improving predictions, especially when data is sparse.

The training of the Multi-Task Gaussian Process was performed using GPyTorch [16].

2.2. ANTARES

The ANTARES code [13] relies on the modified Newtonian method to compute the aerodynamic pressure distribution over arbitrary 3D geometries exposed to a hypersonic flow. It assumes that the local surface pressure coefficient is determined by the maximum pressure coefficient observed on at the stagnation point of the body and the angle between the surface normal and the freestream. Aerodynamic forces and moments are obtained by integrating the surface pressure.

2.3. CFD++

The flow around the capsule is simulated by means of the CFD++ solver [1]. Specifically, we solved viscous 3D Navier-Stokes equations with a single-species perfect gas with a constant heat capacity

ratio of $\gamma = 1.4$. Chemistry and turbulence effects are neglected. Transport properties are computed according to the Gupta-Yos mixture rule [18], while inviscid fluxes are approximated by means of a second-order nodal discretization. The computational domain includes a freestream inlet, an adiabatic wall for the capsule, and a supersonic outlet boundary conditions. Aerodynamic coefficients are obtained as output.

3. HEROES

The structure of the python library HEROES is sketched in Fig. 2.

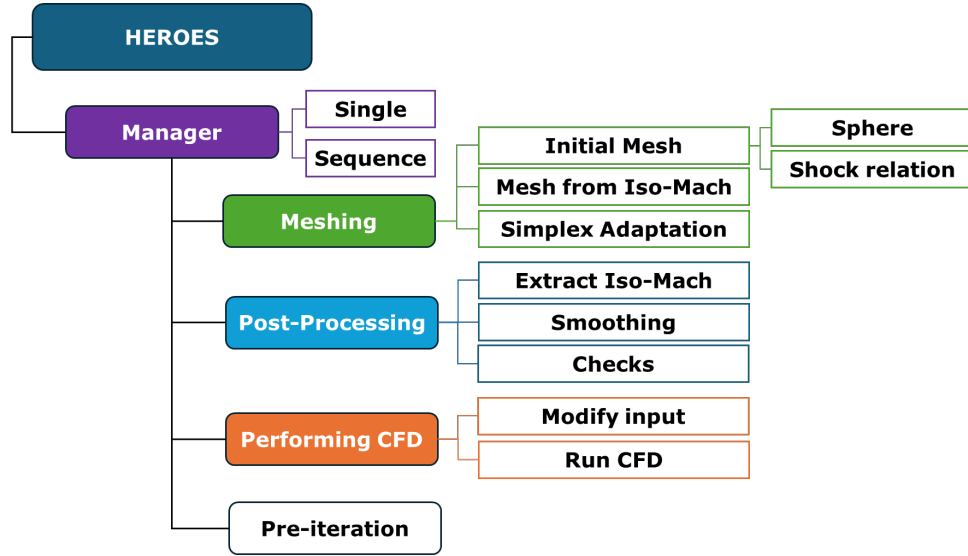


Fig 2. Sketch of HEROES architecture.

An overarching `MANAGER` module set-up the operation sequence and coupling between the different modules. The user can chose to perform a single operation, *e.g.* create a mesh or run a CFD software, or a sequence of operations, *e.g.* first create a mesh and then run the CFD. According to the user-specified operations, HEROES calls specific modules: `MESHING`, `CFD`, and `POST PROCESSING`. The following sections describe in details the current capabilities for each module.

3.1. MESHING

The `MESHING` module relies on tailored scripts for GMSH [17] or Pointwise [2] for mesh generation. The module also includes calls to MAdLib [11] for anisotropic mesh adaptation[12]. For the mesh generation, the meshing scripts feature several options depending on the type of mesher used. Hypersonic applications require the use of structured elements across the shock and in the boundary layer. While GMSH works very well for generating structured layer in 2D, complex geometry in 3D is more challenging. The T-REX algorithm in Pointwise is a structured mesh extrusion technique used for anisotropic mesh generation normal to a surface. This feature is very suitable to ensure quality mesh while keeping flexible meshing capabilities for complex geometries. Within this paper, we focus on the capabilities of the HEROES library using the Pointwise mesher. The automatic mesh generation should be capable of either creating an initial mesh without prior knowledge of the shock location or utilizing a predefined shock position to generate a high-quality mesh. First, when the shock location is unknown, a spherical domain can be built around the object. Inputs for its construction are:

- the mesh of the object (both CGNS and POINTWISE project formats are supported);
- the center and the radius of the spherical domain;
- the angle of attack. It is used to split the outer sphere into an inlet and an outlet domain;
- the mesh resolution.

Secondly, when the shock location is known, an STL file representing the shock surface can be passed to the MESHING module. Such an STL file can be created by post-processing a previously obtained solution (e.g. by means of the POST PROCESSING module), or using shock correlations, e.g. the Billing correlation [7]. A utility to construct a shock STL file from the Billing correlation is also available in the library. The shock surface is used to create the volume mesh. Inputs for its construction are:

- the mesh of the object (both CGNS and POINTWISE project format are accepted);
- the desired wall and shock resolution;
- the maximum T-REX layers;
- whether to use or not use a source to cluster mesh elements in specific parts of the domain.

Following Tang [25], both the spherical and the shock-enveloped meshes are constructed using the Pointwise T-REX algorithm [2], as it was proven to generate high-quality meshes for these applications. The T-REX algorithm is used to build structured-like layers of hexahedral cells along the object and the shock, when its shape is provided. The T-REX layers are built until the cells reach an aspect ratio of 1 or the maximum T-REX layers constraint is met. These layers are built orthogonally to the surfaces. Thus, they guarantee accuracy and robustness both in the boundary and in the shock layers. The rest of the domain is constructed using the Delaunay algorithm [2]. Furthermore, some layers of cells are extruded normally to the shock in the outer direction to guarantee that the shock is fully contained in the numerical domain. An example of a mesh obtained with the Pointwise T-REX algorithm is shown in Fig. 3. As one can see, hexahedral cells are built orthogonally both to the surface and the shock.

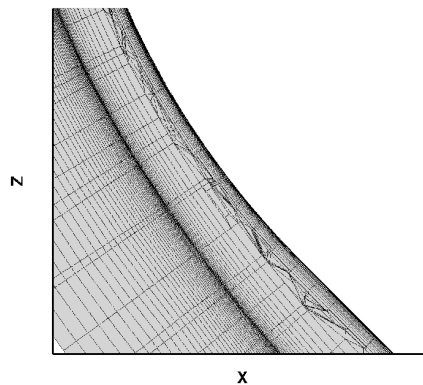


Fig 3. Zoom of a mesh obtained with Pointwise T-REX algorithm.

The MESHING module can also call the MAdLib library [12] for simplex adaptations. MAdLib uses an Hessian based method to refine/unrefine the grid based on a provided solution. The library is capable of adapting based on high-order solution but in this work, solution are usually provided by low order finite volume scheme. The library will adapt the mesh to capture secondary flow features, such as wakes and secondary shocks. For now, the adaptation is decoupled from the conforming mesh operation using Pointwise hence both can not be performed simultaneously and anisotropic adaptation is usually performed when a smooth shock fitting is achieved.

3.2. CFD

The CFD module is used to set up and run the CFD computations. Currently, the library supports both US3D [8] and CFD++ [1] solvers. The MESHING module outputs the mesh in the format required by the selected CFD solver. First, the CFD module modifies the CFD input file according to user-defined inputs. Options available include i) using or not a restarting solution, ii) running viscous or inviscid computations, and iii) changing boundary conditions. If the computation is restarted from a previous solution, the module interpolates such a solution on the new grid. The mesh is then partitioned according

to the number of cores specified in the input file, and the CFD is run until the stop criterion is met. By enabling the library to modify the boundary conditions, a matrix of conditions to run can be passed to HEROES, which can automatically run all the conditions.

3.3. POST PROCESSING

Once the CFD is converged, the POST PROCESSING module post-processes the solution. Both Tecplot [3] and Paraview [4] software are supported. They are used to extract specific quantities of interest. For example, the library can extract the iso-Mach surface corresponding to a fraction of the free-stream Mach number (typically 95%) to be used as shock location in the MESHING module. Depending on the complexity of the flow, different surfaces might be extracted from the solution, for example, because of the development of secondary shocks. A function was built to detect the surface corresponding to the main shock and remove the unnecessary surfaces. The shock surface is then smoothed. Both the Laplacian and the Taubin smoothers [26] are available, being the latter recommended as it limits the surface contraction while smoothing.

After smoothing the surface, different checks are performed on the surface. For example, it is checked whether the surface normal is well-directed. The normal is flipped if the check is not passed. The smoothed and checked shock surface is finally passed to the MESHING module for mesh generation.

3.4. PRE ITERATION

A PRE ITERATION function allows the user to perform different operations before a new iteration is executed. Operations include: i) copying files before they are overwritten by the new iteration, ii) changing meshing parameters, and iii) changing CFD settings.

3.5. Sequence definition

The scheme of a typical workflow is given in Fig. 4. It was observed that, starting from a spherical domain, the shock location converges after 3-4 inviscid CFD runs on meshes with progressively more refined shock surfaces. As these inviscid simulations are performed on meshes that are not refined in the boundary layer, the overall computational cost is limited. Thus, the shock location can be well captured in a relatively short computational time. Once the shock location is converged, the viscous terms can be activated. Consequently, the resolution of the wall should be increased to the desired level of accuracy. Eventually, the wall resolution can be increased until the boundary layer gradients are well captured. After that, the mesh can be adapted, e.g. based on the Mach number gradient, to improve the solution accuracy of secondary flow features, such as the wake.

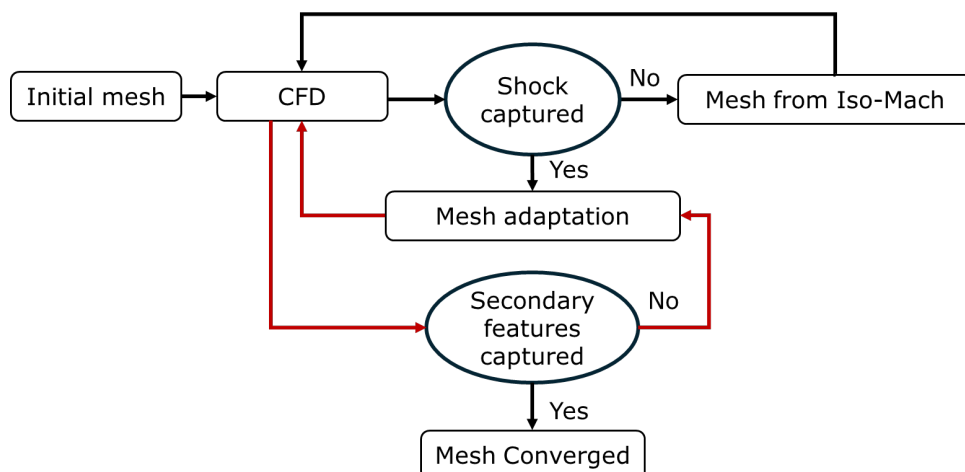


Fig 4. Sketch of HEROES sequence.

3.6. HEROES on the DRACO capsule

An example of HEROES applied to a CFD computation around the DRACO capsule is shown in Fig. 5. As one can see, the shock is well captured after 4 iterations, and the final iteration allows to anisotropically

adapt the mesh in the wake of the capsule. To ensure convergence, aerodynamic coefficients are monitored throughout the HEROES mesh iteration process. Fig. 6 shows the history of the lift coefficient across successive meshes. As the mesh and shock location converge, the lift coefficient also does.

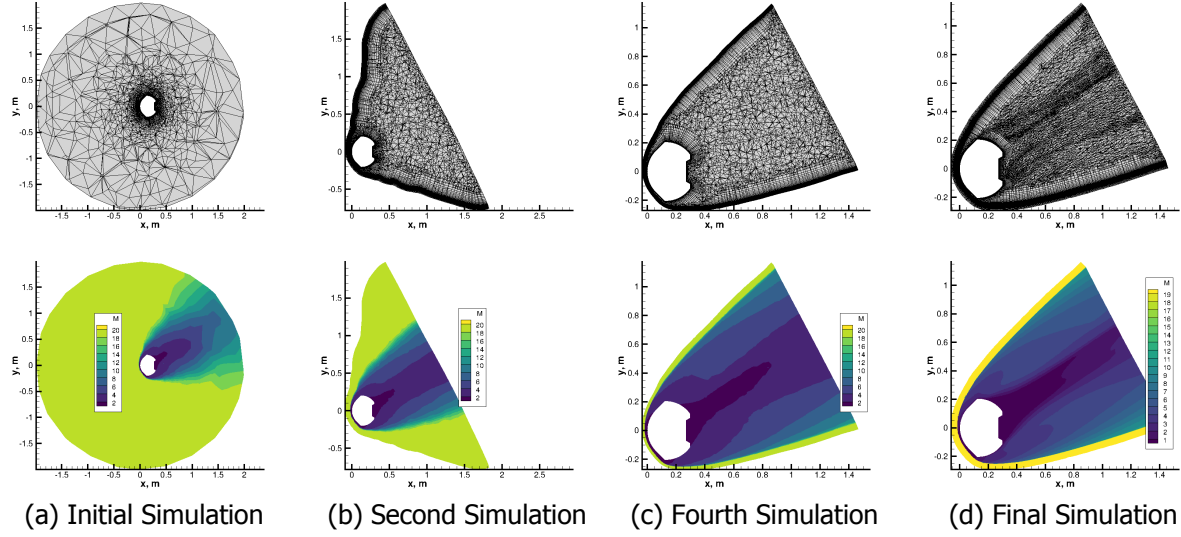


Fig 5. Progression of HEROES generated meshes aligned with the shock

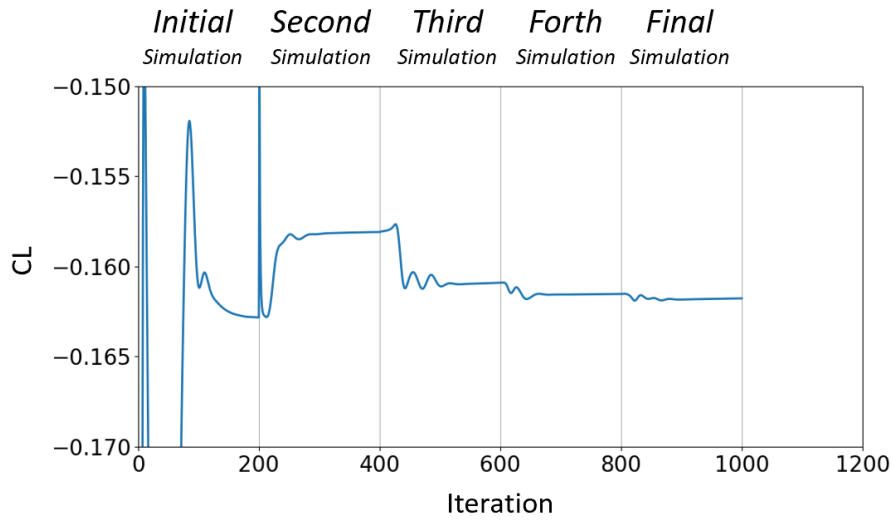


Fig 6. Lift coefficient history across HEROES generated simulations

4. Results

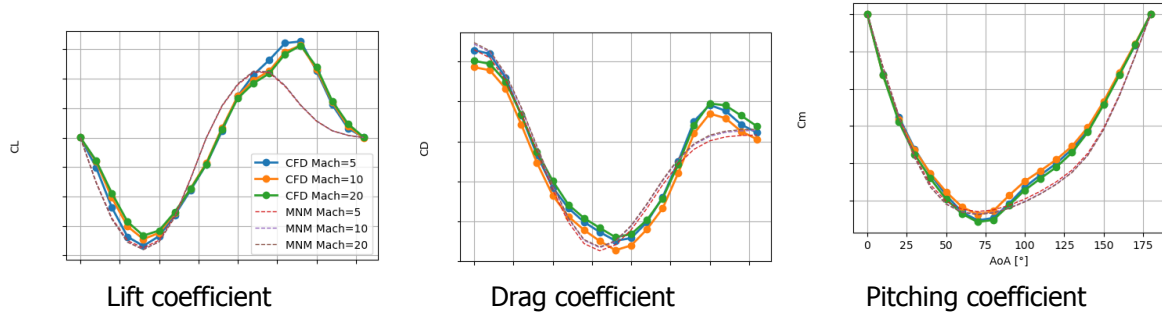
For the generation of the aerodynamic database, we are mostly interested in predicting lift, drag, and pitching coefficients (CL , CD , CM) at different altitudes (and thus Mach numbers), and angles of attack. Three Mach numbers (5, 10, and 20) were chosen to build the aerodynamic database. Leveraging the symmetry of the capsule, we considered angles of attack between 0 and 180 deg. Conditions of interest, along with free stream static and dynamic pressure, and temperature are given in Tab. 1.

Following the methodology explained in Sec. 3.6, 57 uniformly-distributed high-fidelity simulations were performed (3 Mach number for 19 angles of attack). An equal number of low-fidelity simulations were carried out to complement the dataset. A comparison of low- and high-fidelity (CFD++ *vs* ANTARES)

Table 1. Freestream conditions used for CFD simulations

M	p_∞ [Pa]	T_∞ [K]	q_∞ [Pa]	AoA
20	23	248	6578	0°–180°
10	136	266	9548	0°–180°
5	398	243	6975	0°–180°

predictions at different angles of attack is shown in Fig. 7. Both approaches show similar trends in the lift, drag, and moment coefficients across the range of conditions. However, the Mach number dependence is more evident in the CFD results. Furthermore, discrepancies between the two become more pronounced when the angle of attack increases beyond the point where the flow begins to interact with the rear side of the vehicle. This is mainly due to the geometry of the DRACO capsule, which has a cavity at the back. This feature is not accurately captured by the Modified Newtonian Method (MNM).


 Fig 7. High-fidelity (CFD) *vs* low-fidelity (MNM) predictions

Different sets of high-fidelity points were employed to train the surrogate models, see Tab. 2. These points are used to train I) a surrogate model built only with high-fidelity points, and II) a multi-fidelity model, which also accommodates the 57 low-fidelity points. In order to build several multi-fidelity model, the number of high-fidelity training points is increased from 5 to 40 to evaluate the performance of the surrogate models and estimate how data availability influences predictive performance. To evaluate the

Table 2. Training and Test Samples for 2D Case

Input Variables	Mach, AoA
HF Train Samples	5, 10, 15, 20, 30, 40
LF Train Samples	57
HF Test Samples	57

model accuracy, we used a test set consisting of 57 high-fidelity points. For each training sample size, the surrogate models make predictions at these test locations. The prediction error is quantified using the normalized root mean square error (NRMSE). It is defined as:

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}}{\max(y) - \min(y)} \quad (4)$$

where y_i and $\hat{y}_i$ are the test and predicted values at the i -th test point, respectively, and N is the total number of test points. This metric provides a relative measure of prediction accuracy, normalized by

the range of the reference data.

The evolution of the models (only high-fidelity *vs* multi-fidelity) accuracy with respect to the number of high-fidelity training points is shown in Fig. 8. For each sample size, we randomly select training points from the full high-fidelity dataset and repeat this sampling 20 times, building a separate model for each selection. This approach provides a more reliable measure of model performance at the given sample size, rather than relying on a single selection. The NRMSE is calculated for each selection, and the mean, along with the upper and lower bounds, is then computed and shown on the plots to reflect the variability associated with that sample size. It can be seen that multi-fidelity models achieve lower errors than high-fidelity models when the number of high-fidelity samples is limited. As more high-fidelity data is added, both models improve, gradually converging toward similar accuracy levels and approaching zero error.

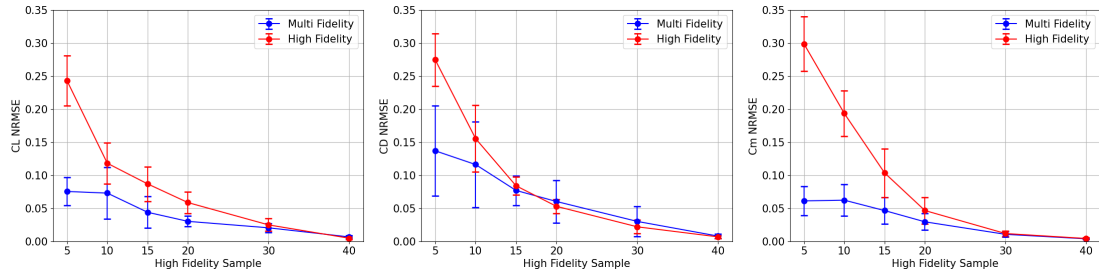


Fig 8. Comparison of model accuracies between high-fidelity and multi-fidelity approaches for the 2D case

Fig. 9 compares high-fidelity and multi-fidelity models for predicting the moment coefficient, both trained using 10 high-fidelity samples. Predictions are evaluated at all high-fidelity test points. The x -axis shows the high-fidelity test values, and the y -axis shows the model predictions. Each point represents one prediction. If a prediction is perfect, it will lie on the diagonal line, meaning the predicted value matches the high-fidelity test value. Points farther from the diagonal indicate larger errors. The multi-fidelity model has most of its points close to the diagonal, while the high-fidelity model is more scattered, showing that the multi-fidelity model gives more accurate predictions overall.

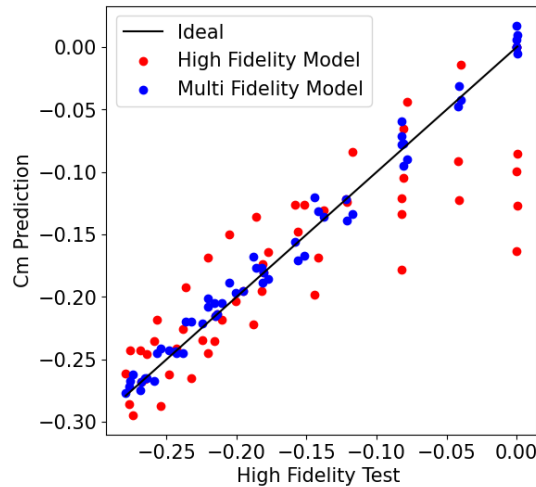


Fig 9. Prediction and high fidelity test values for moment coefficient

5. Conclusion

The objective of this work is twofold. First, we presented the HEROES library. It allows for automatically obtaining high-quality meshes to perform hypersonic simulations using a shock-fitting technique and running a large number of test case with a CFD solver. Specifically, the library was used, coupled with CFD++, to compute the high fidelity data for aerodynamic forces and moments at different freestream conditions (Mach and angle of attack).

Secondly, a multi-fidelity surrogate model was trained to construct the aerodynamic database for the DRACO mission. The high-fidelity CFD simulations were complemented with low-fidelity predictions based on the Newtonian flow theory. The results show that combining high- and low-fidelity data reduces the number of required high-fidelity simulations, offering a substantial reduction in computational cost. It is therefore especially valuable in early design phases or exploratory studies, where computational resources are limited and fast predictions are needed across wide input ranges. Future perspectives involves to include aerothermodynamic analysis (to characterize heat load) and to be able to involve other type of low-fidelity analysis. Other levels could include coarser mesh using CFD, inviscid flow analysis or engineering approaches to estimate aerodynamic database.

ACKNOWLEDGMENTS

The authors acknowledge the ESA contract n° 4000142160/23/D/BL (DRACO-TPS) in which the design of the DRACO capsule was performed, geometry used in this work to illustrate the methodology. Interesting discussion with M.A. Mendez, T. Verstraete and O. Coulaud are also acknowledge.

References

- [1] Cfd++. <https://www.metacomptech.com/index.php/features/icfd>.
- [2] Fidelity pointwise. <https://www.cfd-technologies.co.uk/fidelity-pointwise/>.
- [3] Tecplot. <https://tecplot.com/>.
- [4] J. Ahrens, B. Geveci, and C. Law. Paraview: An end-user tool for large data visualization. *Visualization Handbook*, 01 2005.
- [5] F. Alauzet, L. Frazza, and D. Papadogiannis. Periodic adjoints and anisotropic mesh adaptation in rotating frame for high-fidelity rans turbomachinery applications. *Journal of Computational Physics*, 450, 2022.
- [6] A. Balan, M. Michael, Park, S. Wood, and W. Anderson. Verification of anisotropic mesh adaptation for complex aerospace applications. In *AIAA Scitech 2020 Forum*, Orlando, Florida, 2020. American Institute of Aeronautics and Astronautics.
- [7] F. S. Billing. Shock-wave shapes around spherical-and cylindrical-nosed bodies. *Journal of Spacecraft and Rockets*, 4(6):822–823, 1967.
- [8] G. Candler, H. Johnson, I. Nompelis, V. Gidzak, P. Subbareddy, and M. Barnhardt. Development of the US3D Code for Advanced Compressible and Reacting Flow Simulations. In *53rd AIAA Aerospace Sciences Meeting*, Kissimmee, Florida, 2015. American Institute of Aeronautics and Astronautics.
- [9] M. Capriati, D. Massari, and P. Schrooyen. Development of heroes: Hybrid mesh adapter for hypersonic. In *3rd International Conference on flight vehicles, Aerothermodynamics and Re-entry*, Arcachon, France, 2025.
- [10] M. Capriati, A. Turchi, P. M. Congedo, and T. E. Magin. Holistic characterization of an under-expanded high-enthalpy jet under uncertainty. *Physics of Fluids*, 36(6):066110, 06 2024.
- [11] G. Compere, J. Remacle, J. Jansson, and J. Hoffman. A mesh adaptation framework for dealing with large deforming meshes. *International Journal for Numerical Methods in Engineering*, 82(7):843–867, 2010.

- [12] O. Coulaud, A. Loseille, and P. Schrooyen. Anisotropic mesh adaptation for high-order finite elements spaces with the log-simplex method. application to discontinuous galerkin methods. *Journal of Computational Physics*, 501:112774, 2024.
- [13] T. Durbin, G. Grossir, and O. Chazot. Hypersonic aerodynamic predictions for arbitrary geometries using ANTARES. In *HiSST: 2nd International Conference on High-Speed Vehicle Science and Technology*, Brugges, Belgium, Sept. 2022.
- [14] D. Ekelschot and J. Brock. Enabling metric-based mesh adaptation for advanced compressible flow simulations using us3d. In *AIAA SCITECH*, San Diego, California, January 2022. American Institute of Aeronautics and Astronautics.
- [15] P. J. Frey and F. Alauzet. Anisotropic mesh adaptation for CFD computations. *Comput. Methods Appl. Mech. Engrg.*, 194(48-49):5068–5082, 2005.
- [16] J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson. Gpytorch: blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS’18, page 7587–7597, Red Hook, NY, USA, 2018. Curran Associates Inc.
- [17] C. Geuzaine and J.-F. Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.
- [18] R. Gupta, J. Yos, R. Thompson, and K. Lee. A Review of Reaction Rates and Thermodynamic and Transport Properties for an 11-Species Air Model for Chemical and Thermal Nonequilibrium Calculations to 30000K. Technical report, NASA, 1990.
- [19] Y. Kuya, K. Takeda, X. Zhang, and A. I. J. Forrester. Multifidelity Surrogate Modeling of Experimental and Computational Aerodynamic Data Sets. *AIAA Journal*, 49(2):289–298, Feb. 2011.
- [20] D. Massari, M. Capriati, D. Martins, and B. Helber. Aerodynamic stability analysis and design of a re-entry capsule for a demise assessment platform. In *3rd International Conference on flight vehicles, Aerothermodynamics and Re-entry*, Arcachon, France, 2025.
- [21] P. McCloud. Best practices for unstructured grid shock fitting. In *55th AIAA Aerospace Sciences Meeting*, Grapevine, Texas, January 2017. American Institute of Aeronautics and Astronautics.
- [22] K. R. Quinlan, J. Movva, and E. V. Stein. Multi-Fidelity Aerodynamic Databases for Efficient Representation of Hypersonic Design Spaces. In *AIAA Ascend*, Las Vegas, Nevada, 2021. American Institute of Aeronautics and Astronautics.
- [23] J. A. Rataczak, I. D. Boyd, and J. W. McMahon. Surrogate Models for Hypersonic Aerothermodynamics and Aerodynamics using Gaussian Process Regression. In *AIAA SCITECH 2024 Forum*, Orlando, Florida, Jan. 2024. American Institute of Aeronautics and Astronautics.
- [24] J. Tancred and M. P. Rumpfkeil. Aerodynamic Response Quantification of Complex Hypersonic Configurations using Variable Fidelity Surrogate Modeling. In *53rd AIAA Aerospace Sciences Meeting*, Kissimmee, Florida, Jan. 2015. American Institute of Aeronautics and Astronautics.
- [25] C. Tang. Rapid hypersonic simulations using us3d and pointwise. In *11th International Conference on Computational Fluid Dynamics*, Maui, Hawaii, 2022. American Institute of Aeronautics and Astronautics.
- [26] G. Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’95, page 351–358, New York, NY, USA, 1995. Association for Computing Machinery.