Eleventh International Conference on Computational Fluid Dynamics (ICCFD11), Maui, HI, USA 2022

ICCFD11-2022-0304

Effect of anisotropic mesh adaptation on surface pressure predictions for atmospheric entry simulations.

D. Ekelschot and J.M. Brock
Corresponding author: dirk.ekelschot@nasa.gov

Analytical Mechanics Associates, Inc.
at NASA Ames Research Center, Moffett Field, CA 94035.

Abstract:
A metric-based mesh adaptation capability has been developed [1] for the advanced compressible flow solver US3D [2]. Metric-based mesh adaptation is a solution informed mesh adaptation strategy that improves the resolution of the mesh along anisotropic flow features, like strong shock waves or shear layers. An anisotropic error indicator is reconstructed based on a provided Computational Fluid Dynamics (CFD) solution which prescribes the locally desired element size and orientation. The benefits of this mesh adaptation strategy is twofold: first, it simplifies the manual meshing labor that typically needs to be carried out by the CFD engineer because the mesh adaptation is informed by the provided solution. Second, the computational cost is kept to a minimum while improving the resolution of the simulation.
The aim here is to demonstrate how anisotropic mesh adaptation can be employed to improve the predictions of surface pressure for atmospheric entry vehicles. In this work, we compare CFD simulations that use this newly developed anisotropic mesh adaptation capability for US3D [1] with pressure data that was collected by Mars Entry, Descent, and Landing Instrumentation 2 (MEDLI2) during the atmospheric entry phase of Mars 2020.

Keywords: Anisotropic mesh adaptation, Computational Fluid Dynamics, Atmospheric entry simulations, Mars Entry, Descent, and Landing Instrumentation 2 (MEDLI2).

1 Introduction

The Perseverance rover landed safely on the surface of Mars on February 18th, 2021. The Mars 2020 entry capsule that delivered the Perseverance rover was equipped with various measurement instrumentation on the heatshield and backshell, including pressure transducers, thermocouples, heatflux sensors and a radiometer. This suite of measurement instrumentation is referred to as The Mars Entry, Descent, and Landing Instrumentation 2 or MEDLI2. The data collected by MEDLI2 is used to study aerodynamic and aerothermal performance during the atmospheric entry phase of Mars 2020. This work is part of the MEDLI2 Deep Dive effort which leverages the measured data in order to validate new tools that are currently being developed under the Entry Systems and Modeling (ESM) project within NASA.

In this work, we use the advanced compressible flow solver US3D [2] in conjunction with the newly developed mesh adaptation capability [1] in order to study pressure predictions on the heatshield and backshell during the supersonic atmospheric entry phase of Mars 2020. The mesh adaptation capability allows the CFD engineer to start off with a coarse computational mesh and iteratively improve the CFD simulation results by employing solution informed mesh refinement. Examples of this iterative mesh refinement process applied to atmospheric entry vehicles are shown in [1].

The work presented here shows a comparison between pressure obtained at different mesh refinement levels with the measured pressures obtained by the MEDLI2 during the entry descent and landing phase

of Mars 2020. The goals here are to demonstrate that the initial mesh generation can be highly simplified and that the resolution of the pressure predictions can be improved by iteratively refining the computational mesh.

This paper is outlined as follows: first, a brief description is given of the measured pressures collected by MEDLI2. Second, an introduction is provided of the newly developed metric-based mesh adaptation capability that is linked to the US3D flow solver. This is followed by a detailed discussion of the numerical results and how they compare against the Mars 2020 flight data. As a test case, we consider a supersonic flow ($M = 4.18$) past a 70° sphere-cone entry vehicle that replicates the Mars 2020 capsule geometry. Finally, conclusions are drawn based on the presented results.

2 Mars Entry, Descent and Landing Instrumentation 2 (MEDLI2)

The Mars Entry, Descent and Landing Instrumentation 2 (MEDLI2) collected pressure data during the atmospheric entry phase of the Mars 2020 Perseverance rover through Mars' atmosphere. One of the main hardware components of MEDLI2 was the pressure measurement system, MEDLI2 Entry Atmospheric Data System or MEADS. MEADS gathered accurate pressure data on seven different locations on the heatshield and one location on the backshell. The collected pressure data and the corresponding locations of the pressure transducers on the heatshield and the backshell are shown in Figure 2a and 2b respectively. In the figures 1 and 2a-2b MPH and MPB stands for "MEDLI2 Pressure heatshield" and "MEDLI2 Pressure backshell" respectively.

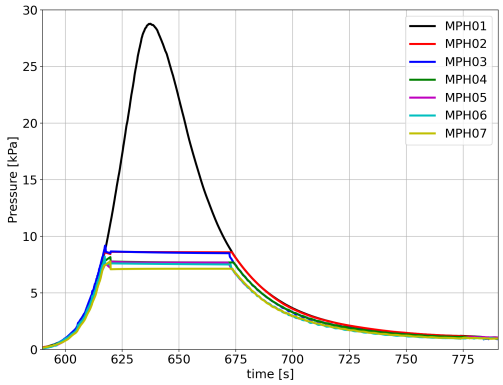
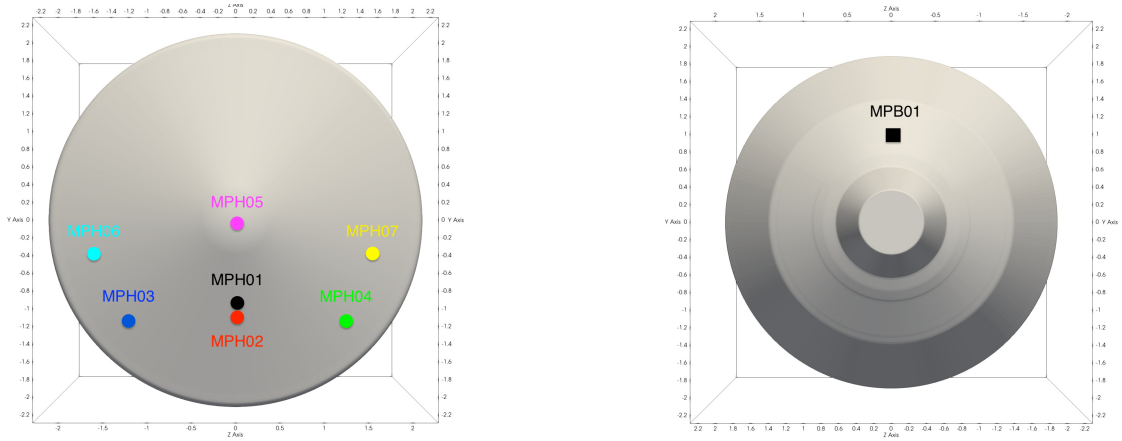


Figure 1: Measured pressure data on the heatshield.

One strain-gauge type pressure transducer (MPH01) is located at the hypersonic stagnation point which has a full scale pressure range of 35kPa. This pressure transducer was able to measure the pressure with an uncertainty of 20Pa. Next to that, there are six peizo-resistive type pressure transducers spread over the lower half of the heatshield. These transducers have a range of 7kPa and measured pressure with an uncertainty of 13.3Pa. They were used to improve the pressure measurement resolution during the supersonic flight regime of Mars 2020. Saturation happens during the hypersonic flight regime due to their lower range which can be observed in Figure 1. One of these pressure transducers is located at the supersonic trim angle stagnation point (MPH02) and one of them is located on the vehicles apex (MPH05). Furthermore two pairs of supersonic pressure transducers, (MPH03,MPH04) and (MPH06,MPH07), were spread over the lower side of the heatshield. Finally, Mars 2020 was instrumented with a single variable-reluctance type pressure transducer on the backshell which has a range of 700Pa and which has an uncertainty of 3Pa. The locations of all pressure transducers are also listed in the Table 1.

Karlgaard et al. [3] and Dutta et al. [4] utilized the MEADS pressure data in conjunction with other on-board measurements in order to reconstruct the trajectory of the vehicle and the free-stream quantities experienced during the entry phase. The reconstructed data [3] is used here to determine appropriate free-stream conditions and vehicle attitude in order to perform an informed comparison between the in-flight



(a) Locations of the MEADs pressure transducers on the heatshield.

(b) Locations of the MEADs pressure transducers on the backshell.

Figure 2: Collected pressure data by MEADS during the atmospheric entry phase of Mars2020.

	MPH01	MPH02	MPH03	MPH04	MPH05	MPH06	MPH07	MPB01
x $[m]$	0.293	0.323	0.518	0.518	0.0	0.522	0.523	2.127
y $[m]$	1.005	1.089	1.092	1.092	0.0	0.365	0.365	1.078
z $[m]$	0.0	0.0	-1.200	1.200	0.0	-1.593	1.593	0.0

Table 1: Locations of the pressure transducers on the heatshield.

measurements and the CFD simulations.

Recently, laminar and turbulent CFD simulations were compared against the MEADS pressure data by Edquist et al. [5]. Good agreement was found between the CFD simulations and the pressure measurements. The results shown by Edquist et al. [5] were calculated using structured hexahedral meshes. However, the goal here in this work is to determine the effect of anisotropic mesh adaptation of highly unstructured hybrid meshes that consist of prisms and tetrahedra. Hybrid meshes that consist of prisms and tetrahedra provide greater flexibility when it comes to mesh adaptation. Furthermore, these meshes have shown to provide accurate predictions of the pressure at the wall [6]. However, they do cause oscillations when predicting quantities of interest that are proportional to the first derivative of the flow quantities, like the heat flux at the wall for example [6, 7]. This will be a topic of research in the near future.

3 Metric Informed Mesh Improvement Capability (MIMIC) for US3D

This section describes the Metric-Informed Mesh Improvement Capability (MIMIC) for the US3D flow solver [1]. MIMIC consists of three main components. First, it computes the metric tensor field based on a provided mesh and initial solution field. Second, it passes the calculated metric tensor field to an open-source anisotropic mesh adaptation Application Programming Interface (API) called ParMMG [8, 9]. ParMMG then computes a new mesh that is aligned with the underlying metric. The last component involves the output into HDF5 format where the distributed adapted mesh is stored in a single mesh file that is compatible with the US3D flow solver. In this way a new flow solution can be calculated on the adapted mesh.

3.1 Reconstruction of the metric tensor field

The goal of metric-based mesh adaptation is to derive a mesh that optimally fits the underlying set of equations and the numerical scheme used to achieve the optimal solution resolution while keeping the com-

putational cost low. Metric-based mesh adaptation achieves this by aligning and sizing the local edges in the mesh with the directionality and size of the local interpolation error. This idea has been first introduced by Hecht and Mohammadi [10] and various formulations for error indicators that prescribe the desired orientation and mesh size followed [11, 12, 13, 14, 15, 16, 17, 18, 19, 20]. In this work, we use the formulation introduced by [18, 19, 20]:

$$\|\mathbf{u}_h - \mathbf{\Pi}_h \mathbf{u}_h\|_{L_p} \leq N^{-\frac{2}{3}} \left(\int_\Omega (\det |\mathbf{H}_a|)^{\frac{p}{2p+3}} \, d\Omega \right)^{\frac{2p+3}{3p}} \tag{1}$$

The formulation given in Equation (1) relies on the fact that the local interpolation error for a given flow quantity is approximated by the Hessian, $\mathbf{H}_a$. In the examples shown in this paper, we use the local Mach number. The local metric tensor at each vertex of the mesh is then approximated by:

$$\mathbf{M}_a = \boldsymbol{D}_{L_p} (\det |\mathbf{H}_a|)^{\frac{1}{2p+3}} \mathbf{R}(\mathbf{x}) |\Lambda|(\mathbf{x}) \mathbf{R}(\mathbf{x})^{-1} \tag{2}$$

where $|\Lambda|$ is a diagonal matrix with modified eigenvalues $h_i^{-2} = \min(\max(\xi|\lambda_i(\mathbf{x})|, h_{max}^{-2}), h_{min}^{-2})$. The values for h_{min} and h_{max} are set by the user and define the minimum and maximum allowed edge length in the mesh. The scalar ξ can be used to amplify the metric field. The global scaling constant, $\boldsymbol{D}_{L_p}$ is defined as:

$$\boldsymbol{D}_{L_p} = N^{\frac{2}{3}} \int_\Omega \left(det(|\boldsymbol{H}_a|)^{\frac{p}{2p+3}} \right)^{-\frac{2}{3}} \tag{3}$$

and is evaluated by integrating the determinant on the Hessian over the computational domain, Ω.

The Hessian is reconstructed using a weighted least-squares gradient reconstruction approach. For a scalar u_0 at a vertex $\boldsymbol{x}_0 = [x_0, y_0, z_0]^t$, we approximate the gradients, $\partial \mathbf{u}_{\boldsymbol{x},0} = [\partial u/\partial x, \partial u/\partial y, \partial u/\partial z]_0^t$, by considering the solution values, u_i, in the centers of the neighboring elements, $\boldsymbol{x}_i = [x_i, y_i, z_i]^t$, where $1 \leq i \leq n$ and where n represents the number of surrounding neighboring elements for $\boldsymbol{x}_0$. By considering the Taylor series expansion of the solution at $\boldsymbol{x}_i$, and considering the local stencil of direct neighboring elements, we can approximate the gradient at $\boldsymbol{x}_0$ by solving the following overdetermined linear system:

$$\mathbf{W} \partial \mathbf{u}_{\boldsymbol{x},0} = \mathbf{S} \tag{4}$$

where the matrix $\mathbf{W}$ and the right-hand side $\mathbf{S}$ are defined as:

$$\mathbf{W} = \begin{bmatrix} w_1 a_1, & w_1 b_1 & w_1 c_1 \\ w_2 a_2, & w_2 b_2 & w_2 c_2 \\ \vdots & \ddots & \\ w_n a_n, & w_n b_n & w_n c_n \end{bmatrix} \quad , \mathbf{S} = \begin{bmatrix} w_1 (u_1 - u_0) \\ w_2 (u_2 - u_0) \\ \vdots \\ w_n (u_n - u_0) \end{bmatrix} \tag{5}$$

where w_i denotes the inverse distance between the neighboring vertex $\boldsymbol{x}_i$ and $\boldsymbol{x}_0$ and where the Taylor series coefficients a_i, b_i and c_i are defined by $\boldsymbol{x}_i$ and $\boldsymbol{x}_0$ as follows:

$$a_n = (x_n - x_0) \tag{6}$$

$$b_n = (y_n - y_0) \tag{7}$$

$$c_n = (z_n - z_0) \tag{8}$$

Equation (4) is solved using QR factorization. Once the gradients are reconstructed, we can repeat this process to reconstruct the second derivatives i.e. Hessian at each location in the computational mesh. The Hessian is then diagonalized in order to obtain the eigenvalues and eigenvectors. The absolute Hessian is created by recombining the absolute values of the eigenvalues with the eigenvectors in order to ensure that the Hessian is symmetric positive definite as shown in Equation (2). The eigenvalues essentially prescribe the local sizing of the element in each direction while the eigenvectors prescribe the local orientation of the element.

More recently, an extended stencil has been implemented in MIMIC that considers the solution values of

the neighboring elements of neighbors as well. This allows us to expand the gradient vector to also include the second derivatives:

$$\partial \mathbf{u}_{\mathbf{x},0} = \left[\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial u}{\partial z}, \frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial xy}, \frac{\partial^2 u}{\partial xz}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 u}{\partial yx}, \frac{\partial^2 u}{\partial z^2}\right]^t \tag{9}$$

The matrix $\mathbf{W}$ and right-hand side $\mathbf{S}$ in Equation (4) can now be expanded so that we can perform a quadratic gradient reconstruction. Considering more than nine surrounding data points, so $n > 9$, $\mathbf{W}$ becomes:

$$\mathbf{W} = \begin{bmatrix} w_1 a_1, & w_1 b_1 & w_1 c_1 & w_1 a_2^2 & w_1 a_1 b_1 & w_1 a_1 c_1 & w_1 b_1^2 & w_1 c_1 b_1 & w_1 c_1^2 \\ w_2 a_2, & w_2 b_2 & w_2 c_2 & w_n a_2^2 & w_2 a_2 b_2 & w_2 a_2 c_2 & w_2 b_2^2 & w_2 c_2 b_2 & w_2 c_2^2 \\ \vdots & \ddots & & & & & & & \\ w_n a_n, & w_n b_n & w_n c_n & w_n a_n^2 & w_n a_n b_n & w_n a_n c_n & w_n b_n^2 & w_n c_n b_n & w_n c_n^2 \end{bmatrix} \tag{10}$$

The right-hand-side vector $\mathbf{S}$ remains the same form as what is shown in Equation (5) but it now also includes additional coefficients that correspond to the solution values in the neighboring elements of neighbors. With the quadratic gradient reconstruction approach we are able to reconstruct both the first and second order gradients in one loop over the elements and no recursive gradient reconstruction is required. The results shown in the next section all utilize the new quadratic gradient reconstruction methodology in order to construct the metric tensor field.

3.2 Metric-aligned mesh generation using ParMMG

Once the metric tensor field is computed, we provide it to the anisotropic mesh adaptation library ParMMG. The ParMMG library is the parallel version of the open-source mesh adaptation MMG3D and it performs anisotropic Delaunay mesh adaptation in parallel based on the provided metric tensor field [21]. The adaptation is performed solely on tetrahedral shaped elements. At the start of the adaptation procedure, the mesh is partitioned and each partition carries out a local adaptation keeping the shared faces between partitions fixed. ParMMG makes sure that the vertices on those shared interfaces are accounted for as well by iteratively re-partitioning and re-adapting the mesh. Hence, after each partition is done adapting, a re-partitioning of the global mesh is carried out such that the vertices and faces that were shared are now internal and therefore taken into account during the next adaptation phase. Typically four to six re-partitioning iterations are used for each adaptation iteration in order to ensure that all vertices in the mesh are accounted for.

3.3 MIMIC Workflow

First, a coarse initial hybrid mesh, $\mathcal{T}_0$, is generated that consists of prisms and tetraheda. We define a prismatic boundary layer mesh in order to maintain a specified resolution of the mesh that is consistent with the y^+ requirements for a given flow configuration and vehicle geometry. The rest of the computational domain is tessellated using tetrahedra. A first stationary state, $\mathcal{S}_0$, is calculated using this coarse mesh. $\mathcal{S}_0$ provides a first indication of where the important flow features like the bow shock and shear layers will be located in the computational domain. The metric tensor field is computed in parallel using this first coarse solution approximation. Next, the tetrahedra are separated from the prisms and are redistributed over the available processors before providing them to ParMMG. This redistribution is done to ensure optimal load-balancing when carrying out the adaptation procedure. The distributed prisms are stored in memory and once ParMMG has computed a new anisotropic tetrahedra mesh, we patch the adapted tetrahedra back onto the distributed prisms. The new adapted mesh, $\mathcal{T}_1$, is outputted. We then interpolate the stationary state $\mathcal{S}_0$ that was obtained using $\mathcal{T}_0$ onto $\mathcal{T}_1$ and use that interpolated solution, $\mathcal{S}_{1,init}$, as a new initial condition. $\mathcal{S}_{1,init}$ is then used to obtain a new stationary state, $\mathcal{S}_1$, on $\mathcal{T}_1$. This process is carried out iteratively to improve the resolution of the solution. The process discussed in this subsection is described in further detail in [1]. The results shown in [1] demonstrates the first application of iterative anisotropic mesh adaptation in the context of atmospheric entry simulations using the US3D flow solver.

4 Supersonic flow ($M = 4.18$) past a 70° sphere cone entry vehicle

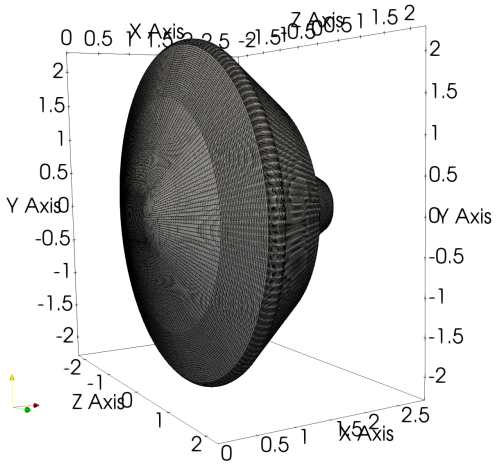
Reconstructed trajectory data [3, 4] is used in order to determine free-stream conditions and the vehicles attitude that define the flow configuration of the simulations presented here. The examples shown in previous work mostly demonstrate metric-based mesh adaptation being applied to low supersonic or transonic flows [1]. Consequently, a slightly higher Mach number is considered here and free-stream conditions were picked that occur approximately 149 s after Entry Interface (EI). The free-stream quantities were reconstructed and tabulated by [3] as part of the MEDLI2 post-flight data analysis effort. Based on that data, the following free-stream conditions were picked:

U_∞ $[\frac{m}{s}]$	T $[K]$	ρ $[\frac{kg}{m^3}]$	M $[-]$	α $[°]$	β $[°]$	$t - t_{EI}$ $[s]$
939.54	198.13	4.36E − 03	4.18	19.35	−0.159	149

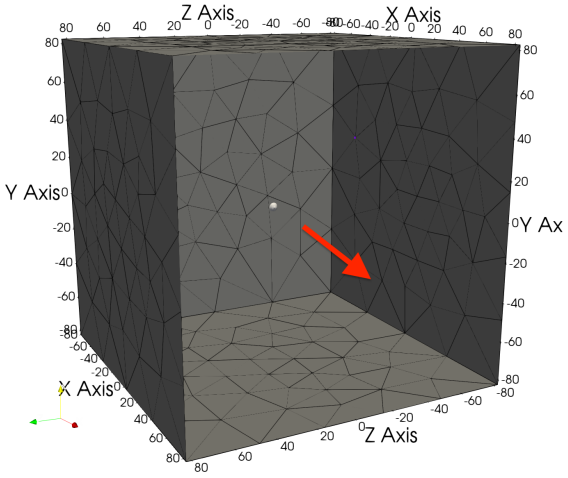
Table 2: Free-stream conditions.

We consider CO_2 and we assume that the gas behaves as a perfect gas. Furthermore, a smooth outer mold line (OML) of a 70° sphere cone entry vehicle is considered that replicates the Mars 2020 capsule. The flight scale geometry is considered here which has a diameter of $D = 4.57m$. Furthermore, we are using the Detached Eddy Simulation (DES) approach that is available in US3D [22] to model turbulence.

First, a coarse initial mesh is generated in order to calculate a first approximation of the flow. One of the benefits of employing MIMIC in the context of US3D flow simulations is that the initial mesh generation process is highly simplified. The geometry of the vehicle is depicted in Figure 3a. The computational domain that surrounds the 70° sphere cone geometry is a simple box with dimensions $\Omega \in [-80, 80]^3$. The inflow faces with the geometry in the center of the computational domain are shown in Figure 3b. The coarse solution



(a) Computational mesh of the smooth outer mold line of a 70° sphere cone entry vehicle at flight scale.

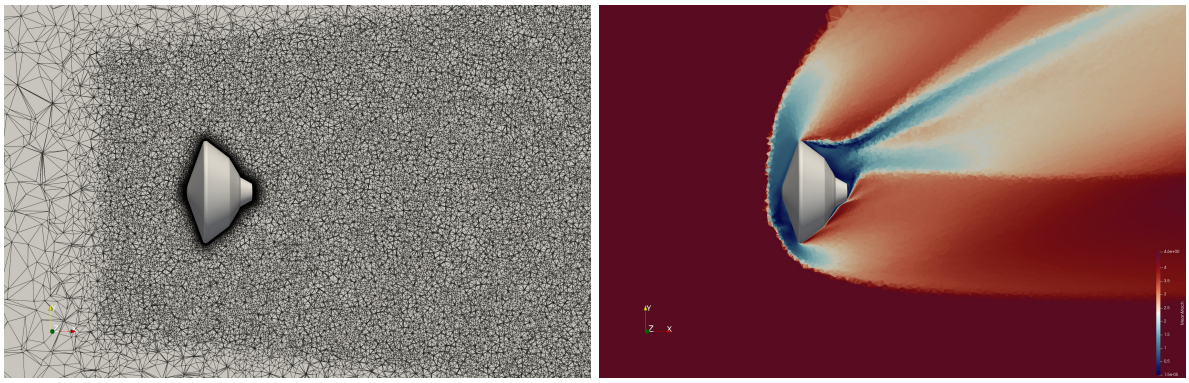


(b) The computational domain looking in upstream direction with the outflow boundary surfaces removed and the flow direction indicated by the red arrow.

Figure 3: Computational domain.

that is calculated on this mesh serves as an initial driver for the iterative mesh refinement procedure. The surface of the wall is tessellated into triangles and these triangles are extruded in normal direction. This procedure creates a layer of prismatic elements near the wall. For this test case, we used an initial spacing of $\Delta x_w = 1.0\text{E} - 05\text{m}$ in order to satisfy the $y^+ < 1$ criteria. The prisms are growing with a factor of 1.1 in normal direction with respect to the geometry. The prismatic boundary layer mesh consists of 69 prisms in normal direction which results in an initial mesh that consists of $n_p = 7.97\text{E}06$ prisms clustered near the geometry and $n_t = 5.71\text{E}06$ tetrahedra in the remainder of the domain.

A first statistically converged solution of the flow is obtained using the free-stream conditions and geometrical orientation of the body listed in Table 2. Convergence is monitored by monitoring the L_2 norm between each time step and looking at the body forces that act on the vehicle. For this initial mesh, the solution appears to be in steady state which can be ascribed to the high numerical dissipation due to the coarseness of the mesh. A stationary state is achieved relatively quickly after approximately ten Convective Time Units (CTU) where CTU $= D/U_\infty$. Figure 4a shows the mesh in the (x, y) plane at $z = 0$ and Figure 4b shows the corresponding mean Mach contours. The Mach contours shown in Figure 4b show the



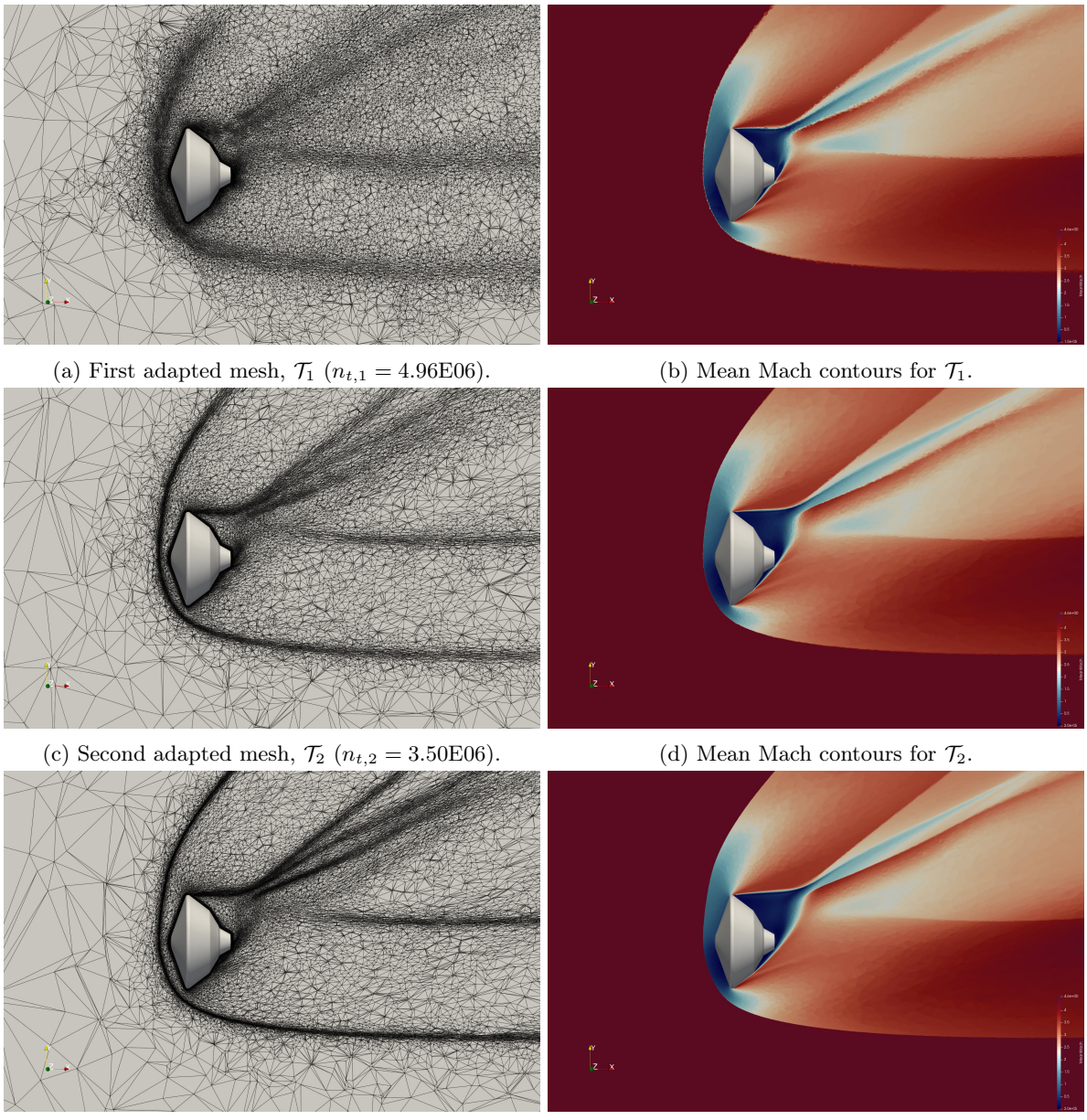
(a) Initial mesh, $\mathcal{T}_0$ ($n_{t,0} = 5.71$E06). (b) Mean Mach contours for $\mathcal{T}_0$.

Figure 4: Comparing mesh adaptation based on Mach number with mesh adaptation for the three adaptation cycles.

strong bow shock that is created upstream of the vehicle. On the upper side of the vehicle, the flow is being expanded and accelerated past the shoulder and then compressed again due to the wake that is formed. This causes a secondary weaker shock wave downstream of the body. Meanwhile at the lower side of the vehicle, the flow is expanded around the shoulder as well and the flow appears to follow the geometry closely. The flow is expanded due to the convexity of the geometry resulting in further acceleration after which it is deflected upwards into the wake due to the sharp corner towards the end of the back shell. The flow then impinges on the lower side of the wake structure creating a weak shock that bounds off the wake region. Overall, the solution is very coarse however it does provide a useful first indication of where important flow features are located in the computational domain.

MIMIC computes the anisotropic error indicator based on the Mach contours shown in Figure 4b. The error indicator is then used to compute a new adapted mesh, $\mathcal{T}_1$ which is shown in Figure 5a. We used minimum and maximum edge length values of $h_{min} = 1.0\text{E} - 02\text{m}$ and $h_{max} = 20.0\text{m}$ in order to compute the metric tensor field which is defined in Equation (2). The number of prisms in this mesh is kept constant ($n_p = 7.97$E06) however, the number of tetrahedra is decreased since $n_{t,1} = 4.96$E06. Figure 5a shows that the mesh is significantly refined in a relatively wide band that surrounds the strong bow shock upstream of the vehicle. However, significant coarsening of the mesh is performed in the farfield. Additionally, significant refinement is achieved near the weaker compression shocks that bound the wake flow. Subtle mesh refinement can be noted near the closure of the wake. Three refinement regions downstream of the wake can now be identified. The top one refines the solution near the weak shock that is formed due to the fact that the flow is directed upwards by the thinner wake region. The lower two refinement areas are targeting the shear layers down stream.

The solution shown in Figure 4b is interpolated onto $\mathcal{T}_1$ and used as a new initial condition from which the simulation is restarted. The errors induced by the interpolation routine are flushed out first. A new stationary state is obtained and the mean Mach number is calculated by considering the last $80 - 100$CTU. The mean Mach number contours ($\mathcal{S}_1$) for $\mathcal{T}_1$ are plotted in Figure 5b. A drastic improvement of the solution is observed in the vicinity of the bow shock, when comparing the mean Mach contours plotted in Figure 5b with the contours plotted in Figure 4b Furthermore, the flow on the lower side of the back shell changes significantly after the first adaptation iteration. Figure 5b shows a small separation bubble that generates a weak shock after the first deflection of the back shell. $\mathcal{S}_1$ is now used to derive a new metric tensor field



(a) First adapted mesh, $\mathcal{T}_1$ ($n_{t,1} = 4.96\text{E}06$).

(b) Mean Mach contours for $\mathcal{T}_1$.

(c) Second adapted mesh, $\mathcal{T}_2$ ($n_{t,2} = 3.50\text{E}06$).

(d) Mean Mach contours for $\mathcal{T}_2$.

(e) Third adapted mesh, $\mathcal{T}_3$ ($n_{t,3} = 7.67\text{E}06$).

(f) Mean Mach contours for $\mathcal{T}_3$.

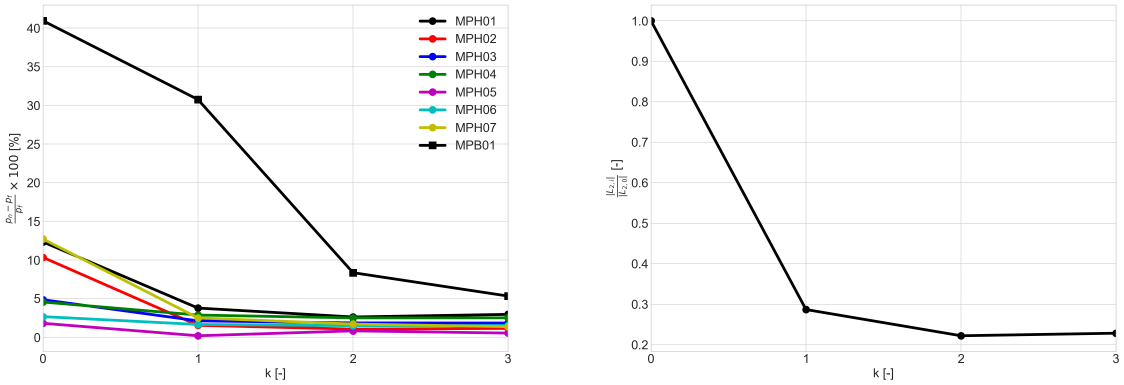
Figure 5: Comparing mesh adaptation based on Mach number with mesh adaptation for the three adaptation cycles.

that will inform where to refine the mesh further.

The next adaptation iteration is shown in Figure 5c and Figure 5d. $\mathcal{T}_2$ has a reduced number of tetrahedra compared to $\mathcal{T}_1$ since $n_{t,2} = 3.50\text{E}06$. The refinement process mostly targets the anisotropic features of the flow like the bow shock and shear layers. Particularly the weak shock that is generated in the lower side of the wake region is being refined significantly. Furthermore, the weak shock that emanates from the first deflection on the lower side of the back shell appears to be captured further downstream. Next to that, the second weak shock that emanates from the second deflection on the lower side of the back shell, which is visible in Figure 5b, appears to have vanished in Figure 5d. This can be ascribed to the fact that the separation bubble on the lower side further increased in size which removes the sharp deflection of the flow towards to wake region.

Finally, the third adaptation iteration is shown in Figures 5e and 5f. In this case the minimum allowable spacing is reduced to $h_{min} = 1.0\text{E} - 03\text{m}$ and $h_{max} = 20.0\text{m}$ and as a results the number of elements is increased. The number of tetrahedra in $\mathcal{T}_3$ is more than doubled with respect to $\mathcal{T}_2$ and counts $n_{t,3} = 7.67\text{E}06$. The bow shock definition becomes even more sharp compared to the second adaptation iteration. Figure 5e indicates that the additional mesh element are mostly dedicated to refining the areas in the computational domain where the anisotropic features are present.

Ultimately, the effect of mesh adaptation on pressure predictions is studied. As mentioned in section 2, the pressure on the heatshield and backshell has been recorded during the atmospheric entry phase of the Mars 2020. The pressure data is collected during the CFD simulations discussed above at similar locations on the heatshield and backshell as the ones listed in Table 1. The numerical pressure predictions are referred to as p_n and the pressure values that were measured in flight are referred to as p_f. We compare the CFD simulations with the pressure measurements taken at $t - t_{EI} \approx 149\text{s}$ as indicated in Table 2. The locations of the pressure probes at the wall in the CFD simulations are not matching exactly with the locations listed in Table 1 since we are limited by the local spatial resolution of the surface mesh. However, the probes are within a radius of $1.5cm$ of the requested locations listed in Table 1. As a first comparison, the percentage difference is computed between the measured pressure data at $t \approx 149s$ after EI, (p_f) and the computed pressure that was obtained using the CFD (p_n). The computed pressure is recorded over the same time span that is used to compute the mean Mach number for the next adaptation cycle $(80 - 100\text{CTU})$. The percentage difference is then taken for each pressure transducer at each k^{th} adaptation iteration and the results are shown in Figure 6a.



(a) Percentage differences between the measured and calculated pressure values for each pressure transducer (colors) at each adaptation iteration (x-axis).

(b) L_2-norm based on the difference between the measured and calculated pressure values over all pressure transducers for each adaptation iteration (x-axis).

Figure 6: Comparison between the MEADS data and the calculated pressure values that follow from CFD simulations that employ metric-based mesh adaptation.

The original mesh that was shown in Figure 4a corresponds to k being equal to zero. The colors and symbols shown in Figure 6a correspond to the colors and symbols used in Figures 2a and 2b. Figure 6a

illustrates that the error in pressure predictions for MPH03, MPH04, MPH05 and MPH06 are already below 5% compared to the measure pressure in flight using the initial computational mesh. The difference is largest for the backshell pressure (MPB01). The percentage differences for the pressure data collected by MPH01, MPH02 and MPH07 are above 10%. This difference is diminished after a single adaptation iteration is reduced to approximately 3%. The difference in backshell pressure requires an additional two adaptation iterations to reduce the percentage difference to approximately 5%.

Furthermore, the L_2-norm is calculated by taken the differences between the computed and the measured pressure values for all locations, eight in total, at each adaptation iteration. For the k^{th} adaptation iteration, the L_2-norm is computed as:

$$L_{2,k} = \sqrt{\sum_{i=1}^{8} \left(p_{n,i} - p_{f,i}\right)^2} \tag{11}$$

This L_2 norm is computed for each adaptation iteration and normalized by the L_2 that is computed using the initial mesh $\mathcal{T}_0$. The calculated values for each adaptation iteration are plotted in Figure 6b. It can be seen that the error is significantly decreased after the first adaptation cycle. However, not much is gained after during the refinement iterations that follow.

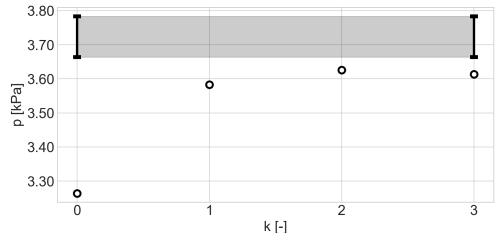
Finally, the direct values of the predicted pressure for each adaptation iteration are compared against the measured pressures. As mentioned before, the MPH01 transducer was used at the trim angle stagnation point during hypersonic flight. This pressure transducer measures the pressure with an uncertainty of 20Pa (σ_h) . The other six pressure transducers on the heatshield measure pressure with an uncertainty of 13.3Pa (σ_s). The pressure transducer located on the backshell measures pressure with an uncertainty of approximately 3Pa (σ_l). The comparisons for all eight pressure measurement locations are shown in Figure 7. The measured values with the corresponding 3σ is indicated by the errorbars and gray shade. The predicted pressure at each adaptation iteration is indicated by the symbols. Figures 7a-7h show a similar trend as shown in Figures 6a and 6b. The symbols trend towards the shaded domain for all pressure measurement location which indicates that the pressure predictions improve with each adaptation iteration. The biggest gain is achieved after the first adaptation iteration.

5 Concluding remarks

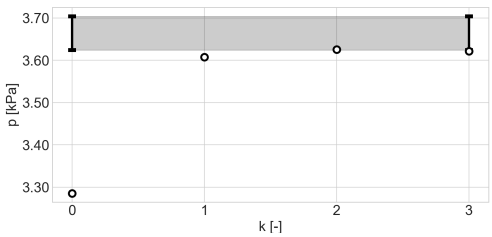
Metric-based mesh adaptation has been applied to a set of US3D flow simulations using the Metric-Informed Mesh Improvement Capability (MIMIC). In this work, a supersonic flow ($M = 4.18$) past a 70° sphere cone entry vehicle is considered. The simulations were performed using CO_2 and assuming that the gas behaves as a perfect gas. Free-stream conditions and vehicle attitude have been determined based on the reconstructed trajectory that was reconstructed during the post-flight analysis of the Mars 2020 EDL phase [3, 4]. First, An initial simulation using a coarse mesh was performed and used to initiate three mesh adaptation cycles. Each new mesh was determined by reconstructing a metric tensor field based on the mean Mach number. The adaptation iterations show significant refinement near the bow shock and shear layers that emanate from the shoulder of the vehicle.

The pressure was recorded at the same locations on the heatshield and the backshell as where the pressure transducers were located on the Mars 2020 entry capsule (see Table 1 and Figures 2a-2b). The maximum error/difference in probe location between the CFD simulations and the in-flight pressure transducers is approximately $1.5cm$. This discrepancy is due to the local sizing of the mesh on the wall. Improvement is observed in pressure predictions after a single adaptation iteration particularly for the pressure at nose of the vehicle. The percentage difference between the simulations and measurements is below 5% for MPH03, MPH04, MPH05 and MPH06 when using the initial mesh. For MPH01, MPH02 and MPH07 the percentage difference lies above 10% as shown in Figure 6a but this error is reduced to below 5% after a single adaptation iteration. The biggest reduction in percentage difference is observed on the backshell (MPB01). For the initial mesh the percentage difference is around 40% while after finishing the three adaptation cycles, this difference is reduced to about 5%.

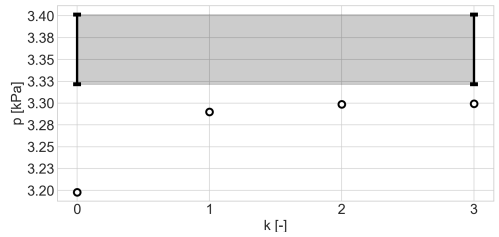
Additionally, the L_2 norm is computed based on the difference of the simulated results and the measured pressure data over all pressure measurement locations on the heatshield and backshell. This is done for each adaptation iteration and this error measure also indicates that the L_2-norm drops by approximately 70%



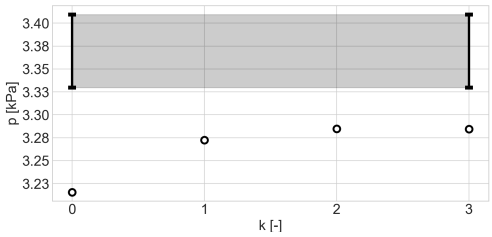
(a) CFD versus measurement with $3\sigma_h$ for MPH01.



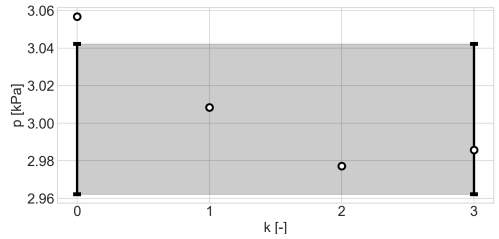
(b) CFD versus measurement with $3\sigma_s$ for MPH02.



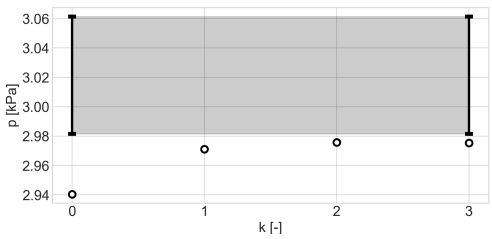
(c) CFD versus measurement with $3\sigma_s$ for MPH03.



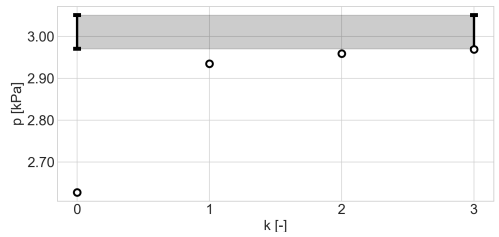
(d) CFD versus measurement with $3\sigma_s$ for MPH04.



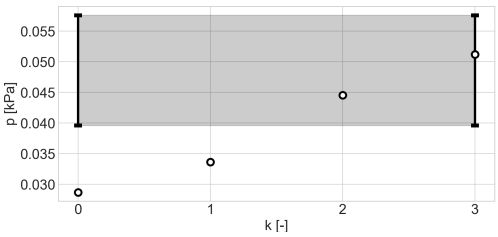
(e) CFD versus measurement with $3\sigma_s$ for MPH05.



(f) CFD versus measurement with $3\sigma_s$ for MPH06.



(g) CFD versus measurement with $3\sigma_s$ for MPH07.



(h) CFD versus measurement with $3\sigma_l$ for MPB01.

Figure 7: Comparing the predicted pressure at each adaptation iteration against the measured pressure with their 3σ error bounds where $\sigma_h = 20$Pa, $\sigma_s = 13.3$Pa and $\sigma_l = 3$Pa.

after a single adaptation iteration as shown in Figure 6b.

Finally, a direct comparison between the predicted pressure values and the measured pressure data is plotted in Figure 7. We see that for all of the pressure transducer locations, the predictions improve with the number of adaptation iterations that are applied. However, the results also indicate that the biggest gain is achieved after the first adaptation iteration and that not all predictions lie within the 3σ error bounds of the measurement. Multiple sources of error could be at play here. One factor can be that the vehicle is oscillating slightly around a trim angle of attack and similar oscillatory behaviour has been observed regarding the slip angle. The CFD simulations consider the vehicle to travel at at fixed velocity while maintaining a fixed attitude.

Acknowledgements

Support for this work was provided by the Entry Systems Modeling (ESM) project under the NASA Game Changing Development (GCD) program. The authors would also like to thank Christopher Karlgaard for the informative discussions regarding the reconstruction of the MEDLI2 pressure measurement data.

References

[1] D. Ekelschot and J.M. Brock. Enabling metric-based mesh adaptation for advanced compressible flow simulations using us3d. In *AIAA SciTech Forum, AIAA 2022-1866*, San Diego, California, US, 2022.

[2] G. V. Candler, H. B. Johnson†, I. Nompelis, P. K. Subbareddy, T. W. Drayna, V. Gidzak, and M. D. Barnhardt. Development of the us3d code for advanced compressible and reacting flow simulations. In *AIAA SciTech Forum, AIAA 2015-1893*, Kissimmee, Florida, US, 2015.

[3] C. Karlgaard, M. Schoenenberger, S. Dutta, and D.W. Way. Mars entry, descent, and landing instrumentation 2 trajectory, aerodynamics, and atmosphere reconstruction. In *AIAA SciTech Forum, AIAA 2022-0423*, San Diego, California, US, 2022.

[4] S. Dutta, C. Karlgaard, D. Kass, M. Mischna, and G.G. Villar. Post-flight analysis of atmospheric properties from mars 2020 entry, descent, and landing. In *AIAA SciTech Forum, AIAA 2022-0422*, San Diego, California, US, 2022.

[5] K.T. Edquist, M. Mahzari, and H.S. Alpert. Mars 2020 reconstructed aerothermal environments and design margins. In *AIAA SciTech Forum, AIAA 2022-0553*, San Diego, California, US, 2022.

[6] I. Nompelis, T.W. Drayna, and G.V. Candler. A parallel unstructured implicit solver for hypersonic reacting flow simulation. In *17th AIAA Computational Fluid Dynamics Conference*, Toronto, Ontario, Canada, 2005.

[7] G. V. Candler, M.D. Barnhardt, T. W. Drayna, I. Nompelis, D.M. Peterson, and P. Subbareddy. Unstructured grid approaches for accurate aeroheating simulations. In *18th AIAA Computational Fluid Dynamics Conference, AIAA Paper 2007-3959*, Miami, Florida, US, 2007.

[8] A. Froehly L. Cirrottola. Parallel unstructured mesh adaptation using iterative remeshing and repartitioning. Technical Report hal- 02386837, INRIA Bordeaux, équipe CARDAMOM, 2019.

[9] A. Froehly L. Cirrottola. Parallel unstructured mesh adaptation based on iterative remeshing and repartitioning. In *WCCM-Eccomas 2020 - 14th World Congress on Computational Mechanics*, Paris / Virtual, France, 2021.

[10] F. Hecht and B. Mohammadi. Mesh adaption by metric control for multi-scale phenomena and turbulence. *AIAA 97-0859*, 1997.

[11] A. Loseille, A. Dervieux, and F. Alauzet. A 3D goal-oriented anisotropic mesh adaptation applied to inviscid flows in aeronautics. In *48th AIAA aerospace sciences meeting including the new horizons forum and aerospace Exposition, AIAA 2010-1067*, Orlando, Florida, US, 2010.

[12] A. Loseille. Metric-orthogonal anisotropic mesh generation. *Procedia Engineering*, 82(Supplement C):403 – 415, 2014. 23rd International Meshing Roundtable (IMR23).

[13] D. Marcum and F. Alauzet. Aligned metric-based anisotropic solution adaptive mesh generation. *Procedia Engineering*, 82:428–444, 2014.

[14] N. Barral, F. Alauzet, and A. Loseille. Metric-based anisotropic mesh adaptation for three-dimensional time-dependent problems involving moving geometries. In *53rd AIAA Aerospace Sciences Meeting, AIAA 2015-2039*, Kissimi, Florida, US, 2015.

[15] F. Alauzet and A. Loseille. A decade of progress on anisotropic mesh adaptation for computational fluid dynamics. *Computer-Aided Design*, 72:13–39, 2016.

[16] Nicolas Barral, Géraldine Olivier, and Frédéric Alauzet. Time-accurate anisotropic mesh adaptation for three-dimensional time-dependent problems with body-fitted moving geometries. *Journal of Computational Physics*, 331:157–187, February 2017.

[17] Michael A. Park, Bil Kleb, W. Kyle Anderson, Stephen L. Wood, Aravind Balan, Beckett Y. Zhou, and Nicolas R. Gauger. Exploring unstructured mesh adaptation for hybrid reynolds-averaged navier-stokes/large eddy simulation. AIAA Paper 2020–1139, 2020.

[18] A. Loseille, A. Dervieux, P. Frey, and F. Alauzet. Achievement of global second order mesh convergence

for discontinuous flows with adapted unstructured meshes. In *18th AIAA Computational Fluid Dynamics Conference*, Miami, Florida, 2007.

[19] A. Loseille and F. Alauzet. Continuous mesh framework part 1: Well-posed continuous interpolation error. *SIAM Journal on Numerical Analysis*, 49(1):38–60, 2011.

[20] A. Loseille and F. Alauzet. Continuous mesh framework part 2: Validations and applications. *SIAM Journal on Numerical Analysis*, 49(1):61–86, 2011.

[21] C. Dobrzynski and P. Frey. Anisotropic delaunay mesh adaptation for unsteady simulations. *Procedia Engineering*, 2008. 23rd International Meshing Roundtable (IMR17).

[22] M.L. Schur, P.R. Spalart, M.K. Strelets, and A.K.Travin. A hybrid rans-les approach with delayed-des and wall-modelled les capabilities. *International Journal of Heat and Fluid Flow*, 29:1638–1649, 2008.