Cranfield University

Turbulence Modelling Assignment

Student:	**Rafaël Sanvicente**	*s431116*
Referring teacher:	Dr. Tamás István Józsa	

November 19, 2024

Cranfield University, Centre for Computational Engineering Sciences
College Rd, Wharley End, Bedford MK43 0AL

Abstract

This study presents an in-depth analysis of turbulence modelling in a wind tunnel simulation, focusing on the visualization of vortex motions, flow field and pressure field. The methodology revolves around the calculation of reference values such as sonic speed, velocity, density and kinematic viscosity using equations derived from the ideal gas law and experimental data. The simulation carried out in Fluent required calibration of these values to match the conditions at the pitot tube site[15][9]. Computational resources and grid convergence were analysed[10], the different simulations were performed on the High Performance Computing (HPC) facility Crescent2 at our disposal at Cranfield University.

The flow different features were visualized using ParaView, highlighting key features such as pressure and velocity fields and identifying vortex motions. The study included a spatially evolving boundary layer analysis and a comparative study with existing research by M. Szőke et al[15]. and S. Pirozzoli et al[12][2]. This comparison included normalized velocity, static pressure distributions and vortex motion analysis.

In addition, the study looked at estimating the resources required for a direct numerical simulation (DNS) of the studied case at a much lower duration. This involved calculating the Kolmogorov length scale, time step size and estimating memory requirements and computation time. The predicted memory requirement of the DNS study was substantial, with a computational time extending over decades, highlighting the immense computational challenge of such simulations.

Acknowledgments

The redaction of this report was made using external linguistic tools, in this case, Grammarly. Codeium extension for VS Code was used to assist code writing and ChatGPT was used to help debugging some features, it was also used to help write and structure the introduction and abstract of this assignment. The author would like to thank P. Hoyos[5] for creating and providing the Matlab formatting scripts used in task 7.

Contents

List of Figures

List of Tables

Chapter 1

Nomenclature

Table 1.1: Personal computation and analysis parameters data described in tables 1 and 2 of the assignment [9]

Property	***Designation***	***Value***	***Unit***
Reference Mach number	Ma_{ref}	0,082	NA
Reference velocity/viscosity ratio	Uref/ mu	$1,7.10^6$	$[fraclm]$
Local Stagnation temperature	Tref tot	297,2	K
Local stagnation pressure	Pref tot	944450	P_a
Local static Pressure	Pref	94009	P_a
Turbulence model	RSM	NA	NA
Inlet mean flow	Uniform		
Inlet turbulence	TI	O%	NA
Calculation case	DNS		

Table 1.2: Ansys calibrated input parameters

Property	***Designation***	***Value***	***Unit***
Inlet velocity	v_{inlet}	3.09	m/s
Dynamic viscosity	ν	$1.64.10^{-5}$	Pa.s
Reference density	ρ^{ref}	1.102	$\frac{Kg}{m^3}$

Chapter 2

Introduction

Wind tunnels are an indispensable tool in aerodynamic research, enabling the study of air flow dynamics over or around various objects, a critical aspect in sectors such as aerospace, automotive and civil engineering. The motivation to digitally simulate these wind tunnels stems from several compelling advantages: cost and time efficiencies, the ability to explore a wider range of test scenarios, and to gain detailed insights into complex flow dynamics that are difficult to capture in physical experiments[15]. Recent years have seen significant advances in the field of Computational Fluid Dynamics (CFD), which have greatly improved our understanding of flow characteristics in low-speed, closed-loop wind tunnels. This progress finds itself in the literature, which focuses on several key areas: boundary layer development and behaviour[12], challenges and solutions in accurate turbulence modelling[2][11].

In understanding the flow characteristics in wind tunnels and other study fields, particular emphasis is placed on the development of the boundary layer - a thin layer of fluid close to the surface of the object[7], which is critical to understanding phenomena such as drag and flow separation[8]. Accurate modelling of turbulence is also a key issue, essential for predicting and analysing flow behaviour in aerodynamic testing[15]. In addition, the influence of wall obstructions in a closed-circuit wind tunnel is a complex factor that requires careful consideration, as it can alter the flow around the test object, requiring specific corrections in both simulations and physical experiments.

The aim of this report is to improve our knowledge of turbulence modelling using advanced techniques in a simulated wind tunnel environment. Our study is driven by several key objectives: firstly, to accurately simulate the flow within a low-speed, wind tunnel test section using CFD tools; secondly, to analyse and visualize key flow features such as vortex motions, pressure fields and boundary layer characteristics, drawing comparisons with existing academic[12][2] and industrial studies[15]; thirdly, to perform a thorough grid convergence study to ensure the reliability and accuracy of our simulation results[10]; and finally, to evaluate and estimate the resources required to perform a Direct Numerical Simulation (DNS) of the case, thus assessing its feasibility.

This report, therefore, aims to integrate theoretical knowledge with advanced computational techniques, providing a comprehensive analysis of wind tunnel flow dynamics. The insights gained from this study are expected to be of significant value for the author.

Chapter 3

Methodology

The methodology used to achieve the necessary tasks of this assignment is described in this chapter:

Calculating reference values: Based on the reference data we have in table 1.1, we can start calculating the reference speed of sound at the location of the pitot tube to then calculate the reference velocity.

We know that[1]:

$$a = \sqrt{\gamma RT}$$

Which means we can calculate the reference speed of sound according to our specific parameters, as it only depends on the temperature:

$$a^{ref} = \sqrt{\gamma RT^{ref}_{tot}} \tag{3.1}$$

We obtain a calculated value of $\boxed{a^{ref} = 345,56m/s}$

As it is given in table 2 of the assignment[9]:

$$Ma^{ref} = \tfrac{u^{ref}}{a^{ref}}$$

So, we can calculate u^{ref} like so:

$$u^{ref} = Ma^{ref} * a^{ref} \tag{3.2}$$

To calculate ρ^{ref} we use the ideal gas law:

$$PV = nRT \tag{3.3}$$

As the fluid (air) can be considered a medium gas, we can write: $P = \rho RT$

which then leads to the equation:

$$\rho^{ref} = \frac{P^{ref}}{R.T^{ref}} \tag{3.4}$$

To calculate the kinematic viscosity, we simply take the ratio given in table 2 of the assignment and solve for ν using u^{ref} found using equation 3.2 :

$$1,7.10^6 = \tfrac{u^{ref}}{\nu}$$

Which gives:

$$\nu = \frac{u^{ref}}{1,7.10^6} \tag{3.5}$$

Fluent Calibration: To calibrate the simulation in Fluent we needed several parameters such as ρ^{ref}, μ^{ref} and u^{ref}. These values were calculated in order to feed fluent some input values and adjust them so we have the same values at the pitot tube location as we calculated in the upper section. With a trial and error method, we adjusted the inlet velocity until we matched the reference velocity at the pitot tube location indicated in the assignment [9].

Computation and grid convergence study: All computations were performed on the HPC facility to reduce the computation time and handle the finer meshes of the model that a desktop Fluent application was not able to handle. We used a configuration with 96 cores to compute the finest studied mesh. The very coarse and coarse mesh could be calculated using the $k - kl - \omega$ viscous model to initialize the problem and then switching to the Reynolds stress model. For the medium mesh, however, we initialized a first time using the spalart–allmaras model, then switched to the $k - kl - \omega$ and finally to the Reynolds stress model. This approached allowed for the model to use simpler viscous models to initialize the solution and start calculations, which allowed the final RSM model to converge more easily.

We then carried out a grid convergence study, this was conducted following the guidance provided by NASA[10], Roache article[13] and a script created by the present author to perform the data acquisition and computation.

Visualizing the flow field: To visualize the resulting flow field and key features of our simulations, we used Paraview, an open source post-processing and visualization software. Once the simulation were completed, we exported the Fluent data in a CGNS format to process it, this consisted in applying various filters, functions and calculations to visualize the key flow features.

Spatially developing boundary layer: To demonstrate that a spatial boundary layer is present and developing in the test section, we plot the X-velocity over 3 different lines using ParaView, these lines are located near the entrance, in the middle and near the exit of the test section. We then recreate the law of the wall curve[6] to try and fit it to our curve. To calculate the length of the lines to ensure we capture the whole boundary layer thickness, we use the following equation[6]

$$\frac{\delta}{x} = \frac{5}{R_e^{1/2}} \tag{3.6}$$

Uploading data to the shared folder: To upload the required data in the required format to the shared folder, we first exported the variables of interest listed in the assignment[9] from fluent. we then formatted the data to the correct format using P.Hoyos matlab script available on Github[5] and uploaded it to the shared folder.

Determining parameters for a DNS study: With the dimensions of the test section found in [15] (summarized in table 4.5) and the parameters from our personal simulation, we are able to compute the Kolmogorov length scale as such [7][2]:

$$\eta = (\frac{\nu^3}{\epsilon})^{\frac{1}{4}} \tag{3.7}$$

To compute the time step size, we apply the following formula[4]:

$$\Delta t \approx \frac{\eta}{U} \tag{3.8}$$

where U is the characteristic velocity of our simulation.

To estimate the memory requirement, we need to know the number of operation the study has to perform, for that we first need to calculate the number of points that the grid will contain in agreement with the dimensions of the domain, these are given by the following equation:

$$n_i = L_i(\frac{\epsilon^{1/4}}{\nu^{3/4}}) \tag{3.9}$$

Which, for a 3D grid of a square test section, gives us a total number of points computed as such:

$$N_{tot} = N_x * N_y^2 \tag{3.10}$$

as $N_y = N_x$

With this, we can calculate the number of step needed to complete the simulation:

$$\Delta Nt = \frac{T_{end}}{\Delta t} \tag{3.11}$$

We then deduct the number of floating point operations as such:

$$N_f = N_{tot}.\Delta N_t \tag{3.12}$$

We can then estimate the amount of memory as such:

$$M_{req} = N_{tot} * N_v * S \tag{3.13}$$

Where M_{req} is the memory required, N_v is the number of stored variables at every grid points and S is the size in bytes of each data point stored.

To estimate the computation time needed to complete the DNS study, we use the following formula:

$$T_{comp} = \frac{\Delta Nt * N_{tot} * NumberofOperationsPerGridPointperTimestep}{ProcessorSpeed[FLOPS]} \tag{3.14}$$

Chapter 4

Results and discussion

4.1 Task 1:

The goal of this task is to calculate the values of the reference velocity, density and kinematic viscosity at the location of the pitot tube, based on our personal assignment data that can be found in table 1.1
Based on the methodology described just before, we can calculate the necessary values.

Starting with the reference velocity, referring to the equation 3.2, we calculate a numerical value of:

$$\boxed{u^{ref} = 28,33m/s}$$

Same approach with ρ^{ref}, using equation 3.4, we obtain a numerical value of

$$\boxed{\rho^{ref} = 1,10\frac{kg}{m^3}}$$

And for the kinematic viscosity, using equation 3.5 we obtain a numerical value of :

$$\boxed{\nu = 1,64.10^{-5}Pa.s}$$

The incompressible and isothermal flow assumptions we made can be justified by the fact that:

- The Mach number is inferior to 0,3, in our case $M_a^{ref} = 0,082 < 0,3$ (Incompressible flow)
- There is no significant temperature changes as the wind tunnel is a closed and controlled environment (Isothermal)

4.2 Task 2:

We can now calibrate our simulation using the provided mesh. To calibrate our simulation, we start by ensuring that we find the same speed we calculated at the pitot tube location (u_{ref}) in the simulation. As we can only provide the inlet velocity in Fluent, we need to calculate this velocity to obtain a rough idea of what it should be and then manually refine this value to adjust our case.

To do so, we use the well-known equation 4.1 as we are in an incompressible, isothermal flux.

$$v_1 S_1 = v_2 S_2 \tag{4.1}$$

We then measure the mesh dimensions in Pointwise to obtain the following inlet and outlet areas:

Table 4.1: Inlet and outlet surface dimensions measured in Pointwise

Property	***Designation***	***Value***	***Unit***
Inlet area	S_1	459,34	$inch^2$
Outlet area	S_2	47,97	$inch^2$

We can now calculate v_1 that corresponds to u_{inlet} in the following manner:

$$u_{inlet} = \frac{u_{ref}.S_{outlet}}{S_{inlet}} \tag{4.2}$$

We obtain a value of:

$$\boxed{u_{inlet} = 2,93m/s}.$$

By setting the velocity inlet speed at 2,93m/s in fluent we can start calibrating the simulation using the very coarse mesh, we start the simulation using a transition $K - KL\omega$ model to initialize the first steps, then switch to our assigned Reynolds Stress Model (RSM) to complete the calculation. we then sought for the X-Velocity at the pitot tube location and adjusted the inlet velocity to match our calculated u_{ref}.

Table 4.2: Adjustment of input speed

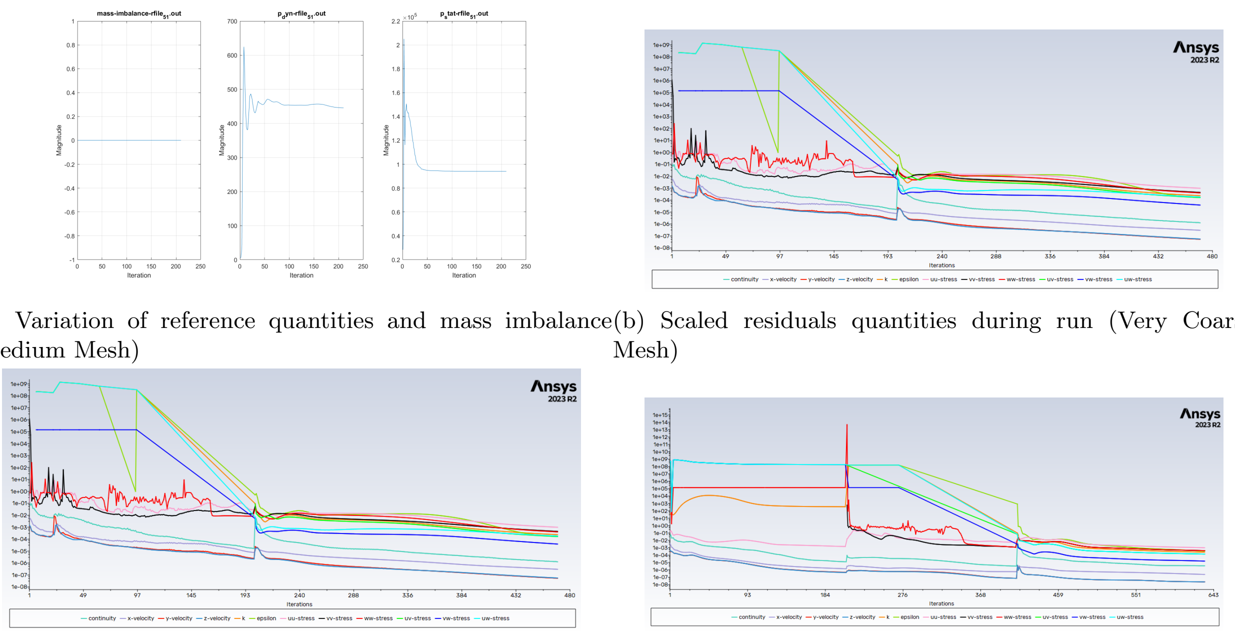
Property	***Designation***	***Value***	***Resulting*** u^{ref}
Inlet velocity	v_{inlet}	3.18 m/s	29.8 m/s
	v_{inlet}	3.15 m/s	28.9 m/s
	v_{inlet}	3.13 m/s	28.7 m/s
	v_{inlet}	3.09 m/s	28.3 m/s

We see with this table that we need to set the inlet velocity to $\boxed{3.09m/s}$ to match our u_{ref} speed. The rest of the simulations have been performed using this inlet velocity value.

4.3 Task 3:

We carried out the computation on the Crescent2 HPC facility and processed the results with ParaView afterwards.

While performing the computation, we kept an eye on the residuals, dynamic pressure, static pressure and mass imbalance to ensure of the quality of the simulation. We noted that for every simulation we performed, the mass imbalance converged to 0 with no flagrant fluctuation, the dynamic and static pressure curves both stabilize throughout the simulation (see fig 4.1) suggesting no abnormal behaviour during the simulation.



(a) Variation of reference quantities and mass imbalance (medium Mesh)

(b) Scaled residuals quantities during run (Very Coarse Mesh)

(c) Scaled residuals quantities during run (Coarse Mesh)

(d) Scaled residuals quantities during run (Medium Mesh)

Figure 4.1: Variation of reference quantities and mass imbalance

Once we gathered the data from the 3 simulations, we visualized the flow properties into ParaView to check that they were consistent with the experiments conducted by M. Szőke et al.[15], which was the case as we obtain the same flow and pressure field (see figure 4.3).

In their report, they present different findings that we will use to compare our results to theirs.

We then performed a grid convergence study following the guidance of the NASA/NPARC Alliance article[10].

In Fluent, and for every mesh, we extracted the streamwise velocity data along a centreline that runs through the entire tunnel length. These data are summarized in table 4.3. Using a python script[1] the necessary grid convergence values were computed (tables 4.3 & 4.4) and plotted in figure 4.2.

regarding this analysis, we can comment that regarding the Richardson extrapolation, the medium mesh captures the best the streamwise fluid velocity with a precision magnitude of $.10^{-2}$. Regarding the Asymptotic range check and GCI values, as the value is close to one $(0.999 \approx 1)$ we can estimate that a further grid would not be rewarding in terms of the ratio of accuracy gain and computation cost. The Grid convergence indexes (GCI) indicate a low relative error between each grid, reflecting good convergence.

Table 4.3: Grid convergence data

Property	***min. X-velocity***	***Grid Size***
Very Coarse mesh (3)	27.7826	4
Coarse Mesh (2)	27.7605	2
Medium Mesh (1)	27.7486	1

Table 4.4: Grid convergence analysis

Property	***Value***	***Unit***
p computed value	0.906	NA
Refinement ration, r	1.98	NA
Richardson extrapolation	27.734	m/s
$GCI_{1,2}$	0.062	%
$GCI_{2,3}$	0.116	%
Asymptotic range check	0.999	NA

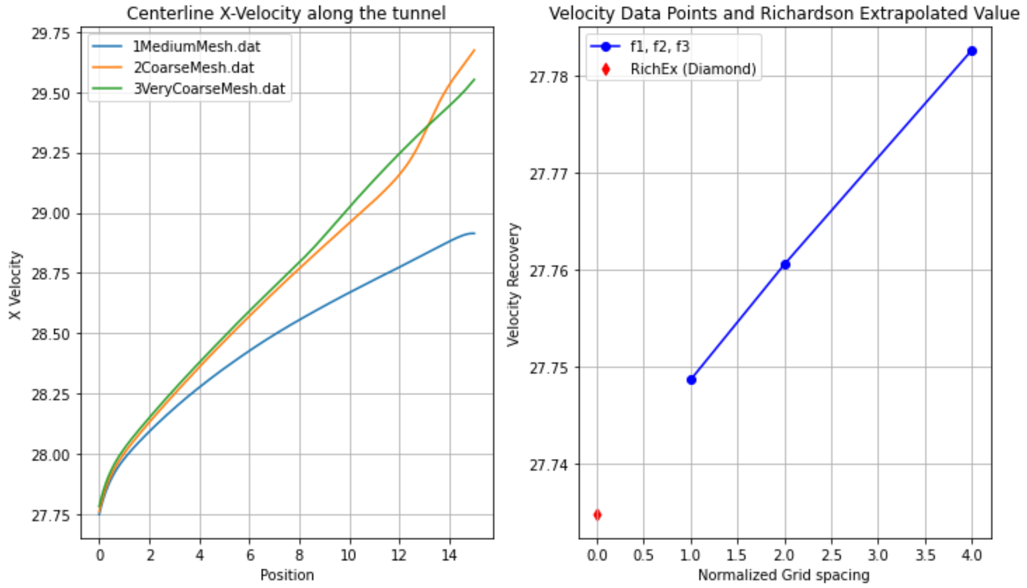
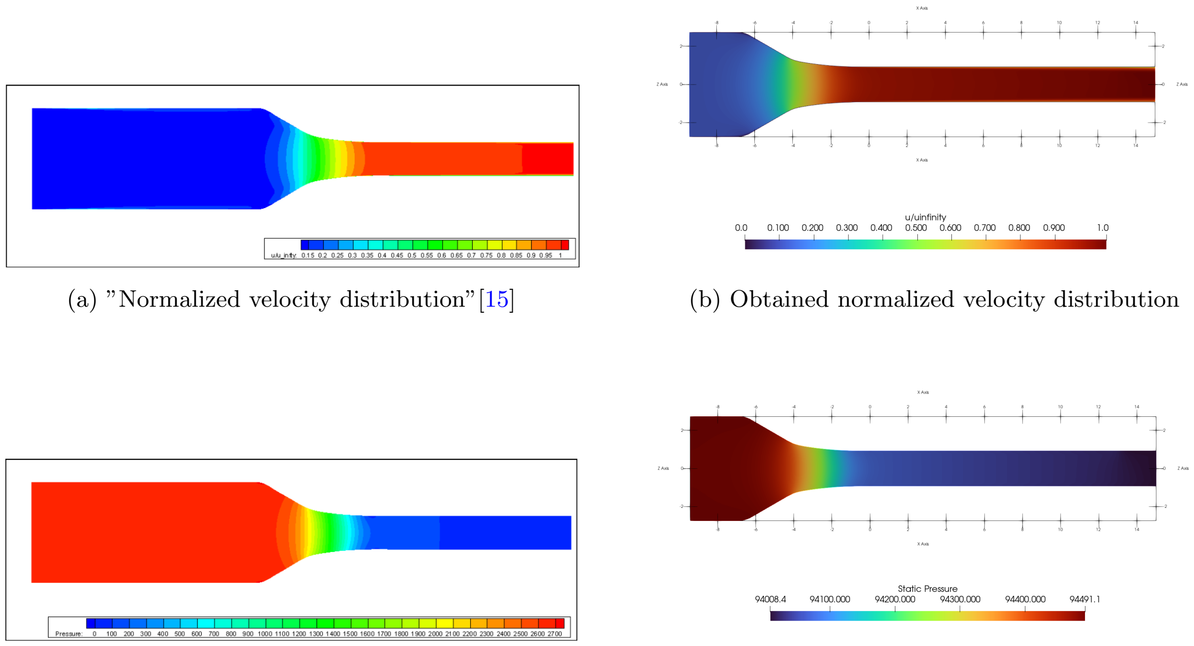


Figure 4.2: Grid convergence analysis curves: velocity profiles and extrapolation of the results

[1] The mentioned script has been submitted under the name ”s431116_GridConvergenceCheck, and is available on the author Github repository[14]”

4.4 Task 4:

Using Paraview, we visualized the flow field and its key-features. We will first focus on the pressure field of the fluid-flow, comparing the data obtained by the team at Virginia Tech and our simulations.
Looking at figure 4.3, we can observe that as expected and stated in the Virginia Tech study [15], as the flow velocity grows in the tunnel the static pressure drops. We also pay attention to the pressure drop contours and also observe a smooth decrease, which can lead us to assume that there is no separation in the domain as expressed in [15].



(a) "Normalized velocity distribution"[15]

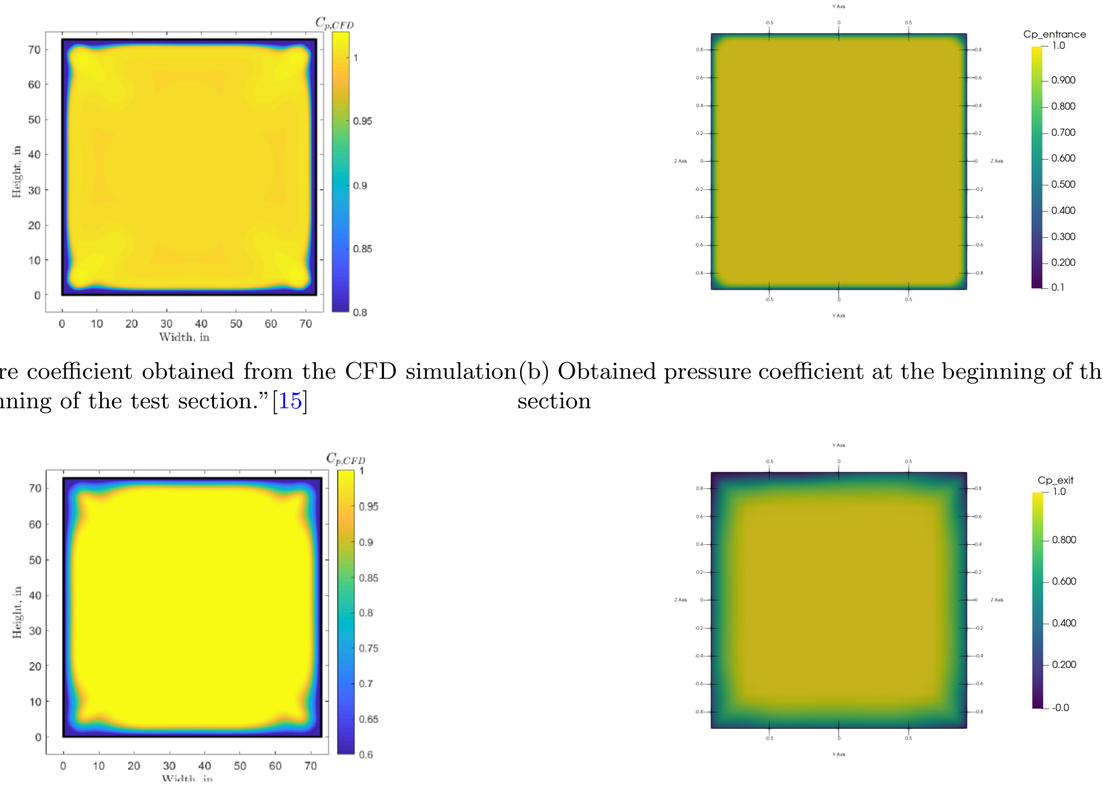
(b) Obtained normalized velocity distribution

(c) "Static pressure distribution"[15]

(d) Obtained static pressure distribution

Figure 4.3: Comparative analysis of normalized velocity and static pressure distributions: Virginia Tech for reference[15]

We also computed the Pressure coefficient at the entrance ($x = 0m$) and at the exit of the test section on fig 4.4. We observed relatively the same flow behaviour as in the Virginia Tech study, that is, a uniform pressure distribution[15] at the centre of the channel, although it should be noted that our simulation is not as accurate as the Virginia Tech one.



(a) "Pressure coefficient obtained from the CFD simulation at the beginning of the test section."[15]

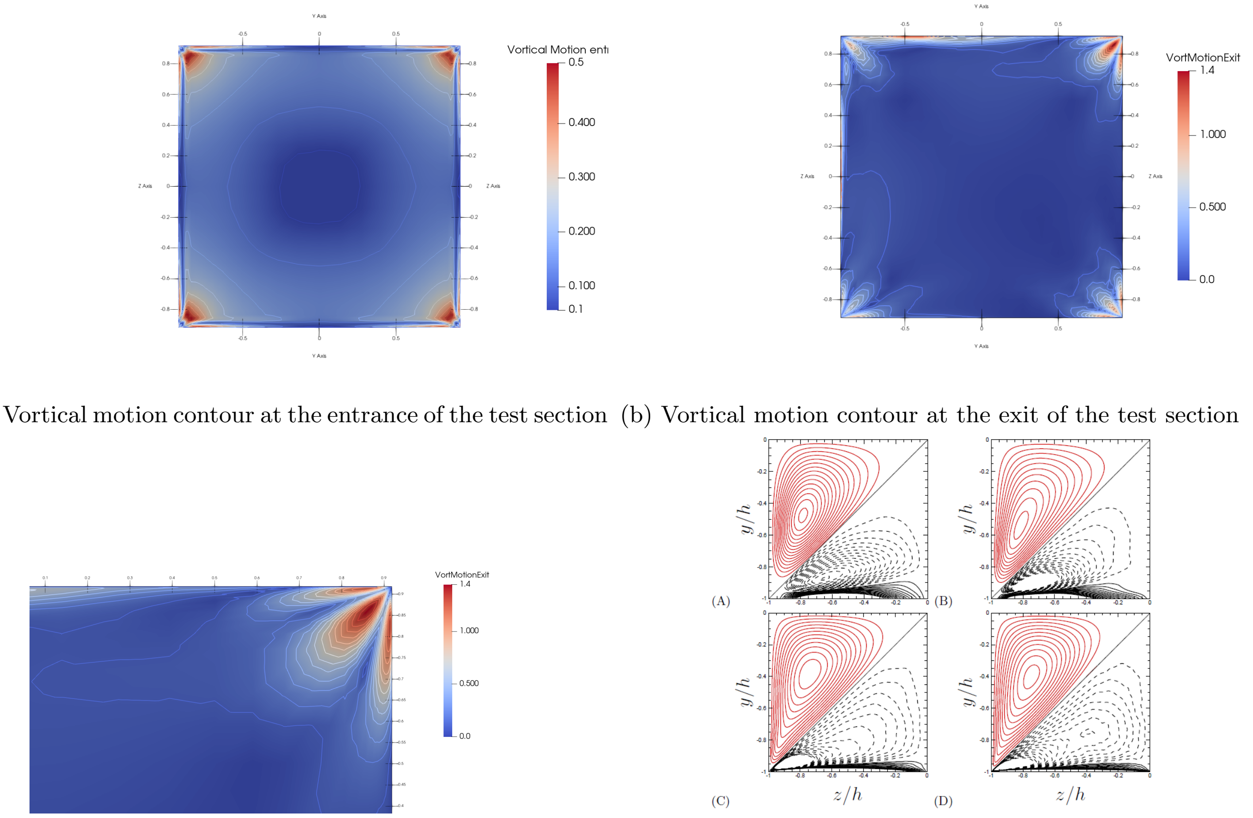
(b) Obtained pressure coefficient at the beginning of the test section

(c) "Pressure coefficient contour map at the exit of the test section obtained from the CFD simulation."[15]

(d) Obtained pressure coefficient at the exit of the test section

Figure 4.4: Comparative analysis of Pressure coefficient at the beginning and exit of the test section: Virginia Tech for reference[15]

We then turned our attention to the vortical motions of the flow field. In order to do this, we computed the secondary flow motions perpendicular to the streamwise velocity, at the entrance, and exit of the test section. We observed the behaviour of the flow using contours and streamlines on the cross-section. Looking at the results of the Virginia Tech study [15] and the S. Pirozzoli et al. study conducted in a square duct flow[12] (similar geometry as the test section). The vorticity contours observed in the DNS Study[12] are similar to the ones we exposed in figure 4.5, the same tendencies are found looking in the corners areas of the test section when plotting the cross flow contours.



(a) Vortical motion contour at the entrance of the test section

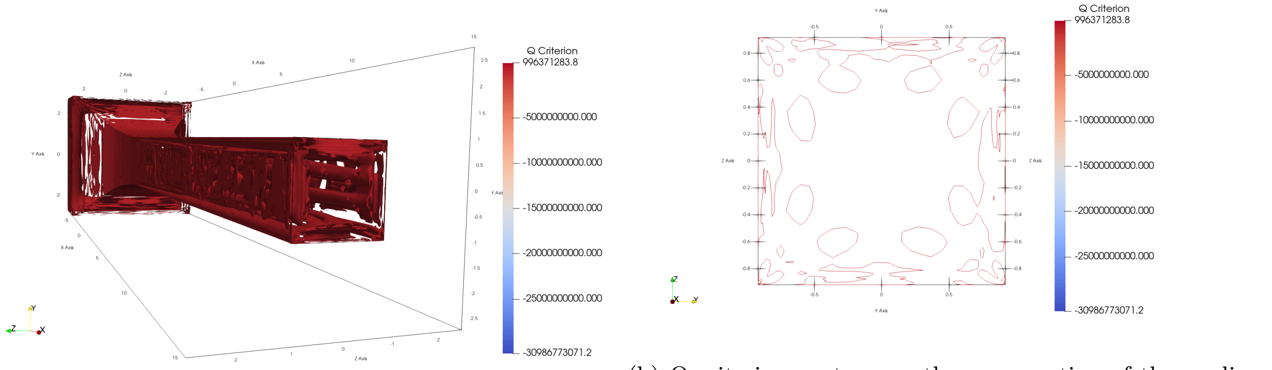
(b) Vortical motion contour at the exit of the test section

(c) Vortical motion: focus on the top right corner of the exit cross-section

(d) "Vorticity contours (lower octant) and mean cross-flow streamlines (upper octant)"[12]

Figure 4.5: Comparative analysis of vortical motions: S.Pirozzoli study for reference[12]

We then computed and plotted the Q-Criterion contour along the wind tunnel to have a visual representation of the vorticity zones and quantity captured by the simulation 4.6a. We clearly see the presence of vortical elements close to the walls of the wind tunnel and especially in the corners of the section as shown in fig 4.6b, supporting our previous observation and corroborating the analysis of the S.Pirozzoli study[12].



(a) Q-criterion contour visualization of the medium mesh

(b) Q-criterion contour on the cross-section of the medium mesh at the test section exit

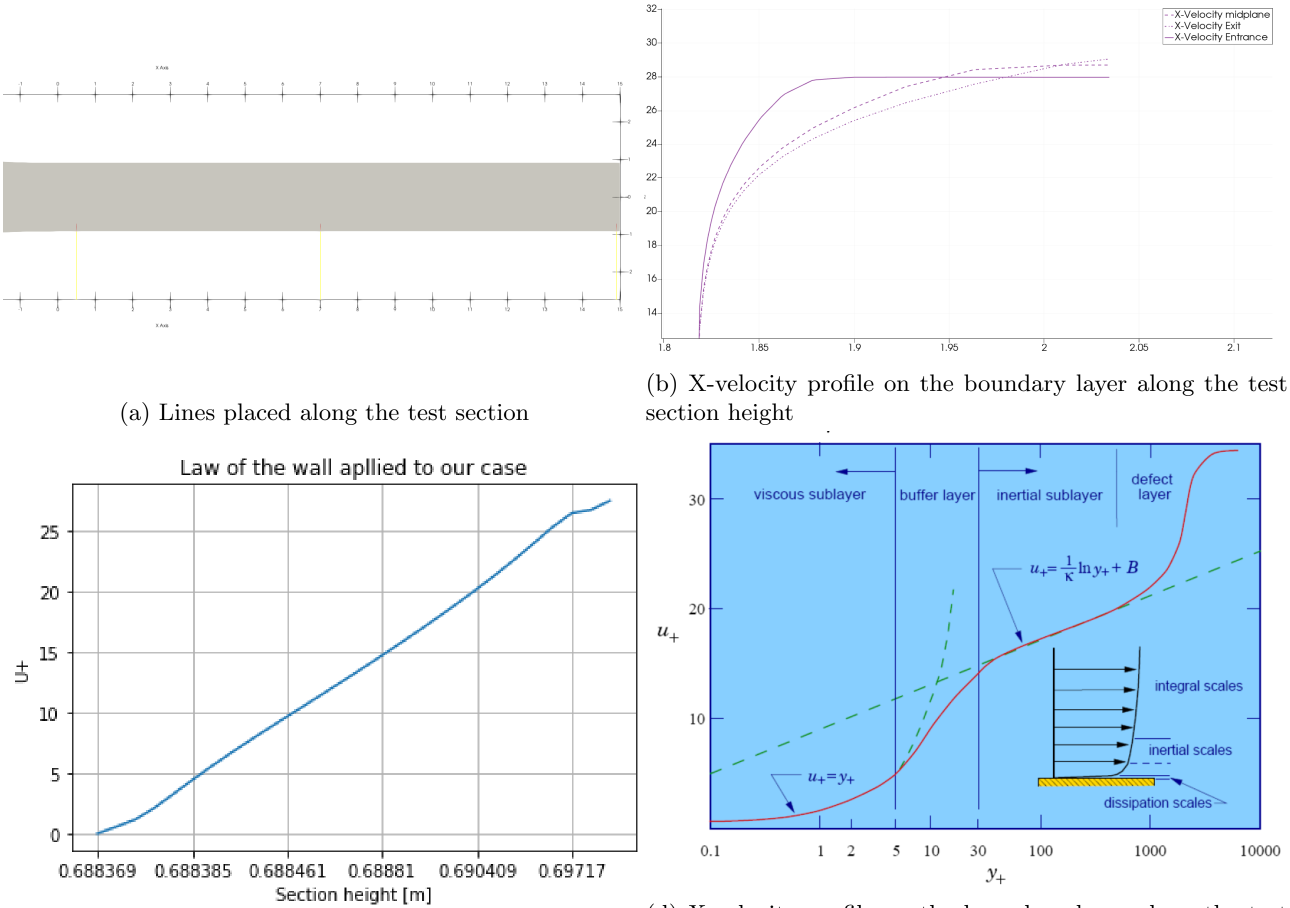
Figure 4.6: Visualization of the Q-criterion contours

4.5 Task 5:

With the simulation dataset imported in Paraview, we plotted the X-velocity along the channel height, which in our case, using equation 3.6, gives us a value of $\boxed{\delta = 4.10^{-2}m}$

While observing the behaviour of the velocity profile along the channel length, we can see that it evolves, suggesting a boundary layer that vary depending on the location of the observation is present. This is also observable on figures 4.8b and 4.8a as the velocity field contours are much more compact at the entrance of the test section than at the exit, which could be indicating a growing boundary layer influencing the streamwise flow velocity.

We then plotted the velocity profile along the line against the section cell height to try and fit the law of the wall curve described in figure4.7d [6] to our case, the results were not that convincing as the obtained profile does not really match the desired curve, and some doubts persists about the plotting algorithm used.



(a) Lines placed along the test section

(b) X-velocity profile on the boundary layer along the test section height

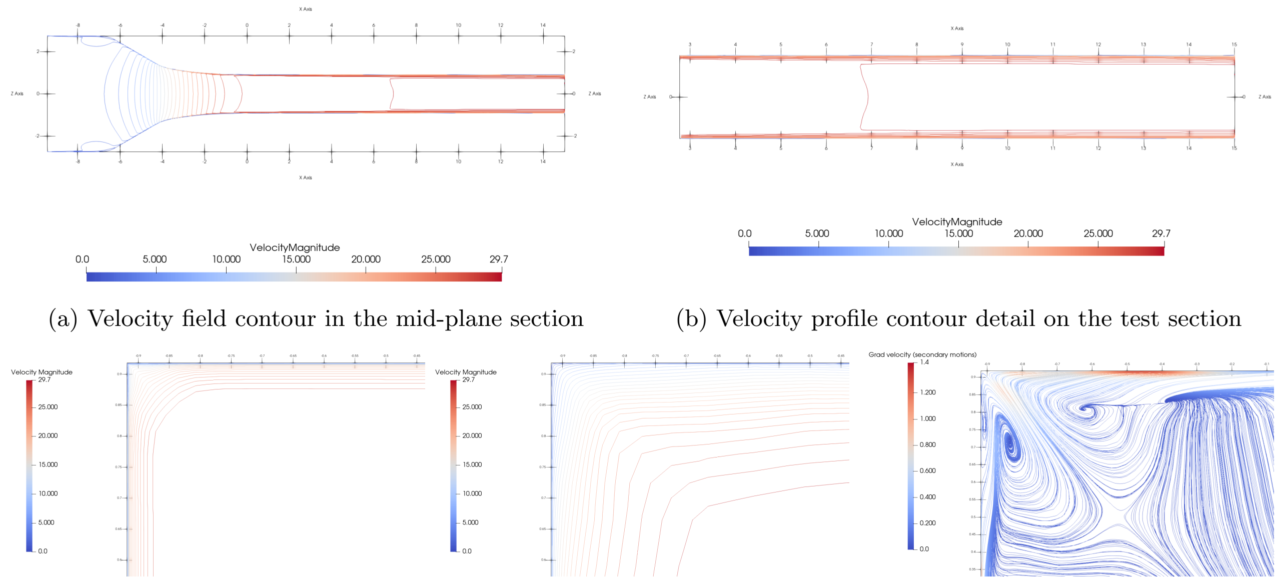
(c) Fitted law of the wall curve (Attempt)

(d) X-velocity profile on the boundary layer along the test section[6]

Figure 4.7: Visualization of the spatially developing boundary layer and the law of the wall curve (Dr. Tamás Józsa lecture for reference)

4.6 Task 6:

Visualizing the velocity field in ParaView help us understand the behaviour of the flow in the corner zones of the test section. Looking at the velocity profile contour of figure 4.8a and 4.8b, we can assume that the turbulence created by the interacting wall is reducing the fluid velocity in the near wall zone. Concerning the corner zones, figures 4.8c and 4.8d shows the velocity field in this particular region, we can see that their influence on the velocity is greater than the walls. We already visualized the contour of the vortical motions (figure 4.5c) explaining the influence of corners in the creation of vortical motions, on top of that we visualized the flow recirculation motion in a corner zone, explaining how these vortical motions originated from (4.8e).



(a) Velocity field contour in the mid-plane section

(b) Velocity profile contour detail on the test section

(c) Velocity Field contour at the entrance of the test section

(d) Velocity Field contour at the exit of the test section

(e) Flow recirculation at the exit of the test section

Figure 4.8: Visualization of the velocity field in the mid-plane and in the corner zones of the test section

4.7 Task 7:

To format the data to the exact template provided, while including the simulation personal parameters, the scripts written by student Pedro Munoz Hoyos and made available for student use on Github[5] were used. The data files containing the variables of interest have been uploaded to the shared folder.

4.8 Task 8:

The wind tunnel test section dimensions are expressed in the table 4.5 summarizing data given in the Virginia Tech study[15].

Table 4.5: Virginia Tech wind tunnel test section dimensions

Property	***Value***	***Unit***
Length	7.3	m
Square test section height	1.85	m

With these dimensions and the parameters from our personal simulation, we are able to compute the Kolmogorov length scale using eq:3.7 as we already calculated the kinematic viscosity. ϵ, the turbulent dissipation rate, can be extracted from the Fluent simulation.

Which, taking an ϵ value of $1,6.10^6$ gives us a final η value of:

$$\boxed{\eta = 7,24.10^{-6}m}$$

corresponding to the spatial resolution of the DNS study.

To compute the time step size, we use equation 3.8, where U is our characteristic velocity. In our case, this calculation gives us:

$$\boxed{\Delta t = 2,55.10^{-7}s}$$

To calculate the number of points that the grid will contain in agreement with the dimensions of the domain, we use equation 3.9, which in our case gives the following results:

$$\boxed{N_x = 1007442 points} \text{ and } \boxed{N_y = N_z = 255311 points}$$

Which in the eventuality of a tridimensional DNS study corresponds to a grid containing:

$$\boxed{N_{tot} = N_x * N_y^2 = 6,56.10^{16} points}$$

As the Memory requirement and number of operation will depend on the length of the simulation, we will assume a duration of $\boxed{T_{end} = 0,1s}$

With this, we can calculate the number of step needed to complete the simulation using equation 3.11, which gives us:

$$\boxed{\Delta N_t = 392157 iterations}$$

To calculate the number of floating point operations that need to be computed, we use equation 3.12:

To estimate the memory requirements, we need to know how many variables we are going to store at each grid points for the simulation. If we assume that we store the velocity, pressure and temperature for each grid points, with a double precision approach, the level of precision is assumed to be of 8 bits to store each variable. We can then estimate the amount of memory required using equation 3.13:

$$\boxed{M_{req} = 1,57.10^{18} bytes \Leftrightarrow M_{req} = 1570 Petabytes \Leftrightarrow M_{req} = 12560 Petabits}$$

To estimate the computation time needed to complete the DNS study, we use equation 3.14 assuming a number of operations per grid point per Time step to be 50 (arbitrary value), and assume that we submit this study to the ARCHER2's HPC facility which is capable of computing over 28 Petaflops/s [3], we obtain a total computation time of:

$$\boxed{T_{comp} = 31901 days \Leftrightarrow 87.4 years}$$

These results show how demanding DNS calculations can be for CFD calculations, even with state-of-the-art resources at our disposal. Of course those values are only a rough estimation of the required resources, and it could certainly be optimized to be more efficient and less demanding, we could for example reduce the length of our domain to reduce the grid size and the computation requirements.

Chapter 5

Conclusion

This study effectively used advanced turbulence modelling in a wind tunnel simulation to accurately capture the flow dynamics. Key achievements include the accurate calculation and calibration of reference values for Fluent simulation, and the use of high performance computing to handle complex models and large cases. Visualization with ParaView provided insightful comparisons with existing research and improved understanding of flow behaviours such as vortices and boundary layers. Investigation of the feasibility of direct numerical simulation (DNS) revealed the significant computational resources required, highlighting the challenges of detailed turbulence simulation. Overall, even if the finished work does not meet all expectations, the project successfully integrated theoretical and computational approaches, providing valuable and tangible learning outcomes in the field of turbulence modelling.

Bibliography

[1] NASA Glenn Research Center. *Speed of Sound*. URL: https://www.grc.nasa.gov/www/k-12/VirtualAero/BottleRocket/airplane/sound.html#:~:text=The%20speed%20of%20sound%20(a,the%20absolute%20temperature%20(T)..

[2] "DNS of passive scalars in turbulent pipe flow". In: *Journal of Fluid Mechanics* 940 (June 2022). ISSN: 14697645. DOI: 10.1017/jfm.2022.265.

[3] University of Edinburgh. *ARCHER2 Hardware and software details*. URL: https://www.archer2.ac.uk/about/hardware.html.

[4] Altair engineering. *Altair Acu-Solve manual: Direct Numerical Simulation*. URL: https://help.altair.com/hwcfdsolvers/acusolve/topics/acusolve/training_manual/direct_numerical_simulation_r.htm.

[5] Pedro Munoz Hoyos. *ICI-Turbulence-Modelling Git Repository*. URL: https://github.com/Pedro-Munoz-Hoyos/ICI-Turbulence-Modelling.

[6] Tamás Józsa. *Characterising turbulent flows Averaging, correlations, and spectra*. 2023. URL: www.cranfield.ac.uk.

[7] Tamás Józsa and Ben Thornber. *Lecture1: Intoduction-Turbulence, the unsolved problem of classical physics*.

[8] Tamás István Józsa. *Drag Reduction by Passive In-Plane Wall Motions in Turbulent Wall-Bounded Flows*. 2018.

[9] Tamás István Józsa. *Turbulence Modelling Computational Modelling of Turbulent Air Flow in the High-Speed Leg of the Virginia Tech Stability Wind Tunnel*. 2023.

[10] NASA. *Examining Spatial (Grid) Convergence*. URL: https://www.grc.nasa.gov/www/wind/valid/tutorial/spatconv.html.

[11] Denis J. Phares and Gaurav Sharma. "A DNS study of aerosol deposition in a turbulent square duct flow". In: *Aerosol Science and Technology* 40 (11 Nov. 2006), pp. 1016–1024. ISSN: 02786826. DOI: 10.1080/02786820600919416.

[12] S. Pirozzoli et al. "Turbulence and secondary motions in square duct flow". In: (July 2017). DOI: 10.1017/jfm.2018.66. URL: http://arxiv.org/abs/1707.03638%20http://dx.doi.org/10.1017/jfm.2018.66.

[13] P.J Roache. "Perspective: A method for uniform reporting of grid refinement studies". In: (). URL: https://www3.nd.edu/~coast/jjwteach/www/www/60130/CourseLectureNotes/Roache_1994.pdf.

[14] Rafaël Altaïr Sanvicente. *Turbulence modelling GitHub Repository*. 2024. URL: https://github.com/RafaelSanv/TurbModel.

[15] Máté Szőke et al. "Developing a numerical model of the virginia tech stability wind tunnel for uncertainty quantification based on real-world geometry". In: vol. 1 PartF. American Institute of Aeronautics and Astronautics Inc, AIAA, 2020. ISBN: 9781624105951. DOI: 10.2514/6.2020-0343. URL: https://arc.aiaa.org/doi/epdf/10.2514/6.2020-0343.