UNIVERSITY OF THESSALY
ENGINEERING SCHOOL
DEPARTMENT OF MECHANICAL ENGINEERING

FAST TRANSIENT WORKFLOW FOR AUTOMOTIVE EXTERNAL AERODYNAMIC SIMULATIONS

by
Christina Peristeri

Supervisors
Prof. D. Valougeorgis, University of Thessaly
Dr. P. Hutcheson, Ansys Hellas

Diploma Thesis

Submitted in partial fulfillment of the requirements for the
Diploma of Mechanical Engineering

Volos 2023

UNIVERSITY OF THESSALY
ENGINEERING SCHOOL
DEPARTMENT OF MECHANICAL ENGINEERING

FAST TRANSIENT WORKFLOW FOR AUTOMOTIVE EXTERNAL AERODYNAMIC SIMULATIONS

by
Christina Peristeri

Supervisors
Prof. D. Valougeorgis, University of Thessaly
Dr. P. Hutcheson, Ansys Hellas

Diploma Thesis

Submitted in partial fulfillment of the requirements for the
Diploma of Mechanical Engineering

Volos, 2023

Approved by the Committee on Final Examination:

Supervisor:	Prof. D. Valougeorgis Department of Mechanical Engineering, University of Thessaly
Member:	Prof. N. Pelekasis Department of Mechanical Engineering, University of Thessaly
Member:	Assistant Prof. K. Ritos Department of Mechanical Engineering, University of Thessaly

Acknowledgements

I would like to express my gratitude to Prof. D. Valougeorgis for giving me the opportunity to be his student. I thank him for his advice and guidance throughout my diploma thesis work. I would also like to thank Prof. N. Pelekasis and Assistant Prof. K. Ritos for serving in my thesis committee.

I would like to express my sincere gratitude to my manager Dr. P. Hutcheson for his advice and assistance during my internship at Ansys Hellas. I would also like to thank my mentor Robert Winstanley, as well as Tobias Berg, Domenico Caridi, Florian Menter and Samir Patel for their practical support during the research for my diploma thesis. Thank you to all my colleagues at the Ansys Hellas office and every other Ansys office in the world. I am excited and motivated to be working with every single one of you.

Θα ήθελα να ευχαριστήσω θερμά τους γονείς μου Κωνσταντίνο και Παρασκευή, και τα αδέρφια μου Δημήτρη και Γιώργο, οι οποίοι πίστεψαν σε εμένα και χωρίς την αγάπη και την συνεχή υποστήριξή τους δεν θα είχα πετύχει τους μέχρι τώρα στόχους μου.

Τα χρόνια φοίτησής μου στον Βόλο δεν θα ήταν τα ίδια χωρίς μερικούς σημαντικούς ανθρώπους που έχω τη χαρά να θεωρώ φίλους μου. Θα ήθελα συγκεκριμένα να ευχαριστήσω τους Μιχάλη, Μάριο, Μενεξέ, Βίκυ, Ναταλία, Θάνο, Στέργιο, Γιάννη, Γιάννη, Μαριάννα, Κωνσταντίνο, Σπύρο.

Τέλος θα ήθελα να ευχαριστήσω τα μέλη της ομάδας Κένταυρος του Πανεπιστημίου Θεσσαλίας, με τα οποία έζησα κάποιες από τις πιο αξιοσημείωτες στιγμές των ακαδημαϊκών μου χρόνων.

FAST TRANSIENT WORKFLOW FOR AUTOMOTIVE EXTERNAL AERODYNAMIC SIMULATIONS

CHRISTINA PERISTERI

Department of Mechanical Engineering, University of Thessaly, 2023

Supervisors:

Prof. Dimitris Valougeorgis, Professor of Mesoscale Methods in Flows and Transport Phenomena

Dr. Paul Hutcheson, Application Engineering Manager

Abstract

Computational Fluid Dynamics (CFD) is an essential tool for judging the performance of aerodynamic devices in the automotive industry. In recent years, the demand for rapid innovations, along with more strict regulations and the general move towards electrification have significantly increased the need for accurate and accelerated workflows for the CFD simulations of external automotive aerodynamic flows. The present thesis aims to provide such a workflow, that is suitable for most modern commercial automotive cases. The workflow includes both the meshing and the solution steps. The geometry that was selected for correlation of the simulation results with wind tunnel experiments was the SAE Notchback Closed Cooling DrivAer model, an industry standard for CFD correlation studies. A mesh independence study was conducted, using the "Poly-Hexcore" mesh type available in Ansys Fluent Meshing, while the Ansys Fluent solver was used for an initial steady-state RANS simulation as well as a Scale Resolving Simulation (SRS) using the SBES numerical model. The medium level of mesh refinement, with a 178 million total cell count, was found to be sufficient for capturing most of the off-body flow phenomena and the resultant average force values were very close to those of the finest mesh, with 480 million cell count. The results from the medium mesh simulation also showed good agreement with the experimental data. The final proposed workflow, with the minimum number of iterations for the steady-state simulation, and minimum number of timesteps for the transient simulation, that were required to obtain stable averaged results, showed significant speedup. During a design circle engineers could run up to 11 simulations within a week, using on average 2048 CPU cores.

Key words: Automotive, Aerodynamics, CFD, RANS, SRS, DrivAer

ΑΕΡΟΔΥΝΑΜΙΚΗ ΠΡΟΣΟΜΟΙΩΣΗ ΜΕΤΑΒΑΤΙΚΗΣ ΡΟΗΣ ΓΥΡΩ ΑΠΟ ΟΧΗΜΑ

ΧΡΙΣΤΙΝΑ ΠΕΡΙΣΤΕΡΗ

Τμήμα Μηχανολόγων Μηχανικών, Πανεπιστήμιο Θεσσαλίας, 2023

Επιβλέπων Καθηγητής: Δρ. Δημήτρης Βαλουγεώργης, Πανεπιστήμιο Θεσσαλίας

Επιβλέπων: Δρ. Paul Hutcheson, Ansys Hellas

Περίληψη

Η Υπολογιστική Ρευστοδυναμική (CFD) αποτελεί ένα σημαντικό εργαλείο για την αξιολόγηση της απόδοσης των αεροδυναμικών συσκευών, που χρησιμοποιείται στην αυτοκινητοβιομηχανία. Προσφάτως, η μεγάλη ζήτηση για νέες και πρωτότυπες εξελίξεις, σε συνδυασμό με πιο αυστηρούς κανονισμούς που αφορούν την ασφάλεια των αεροδυναμικών συσκευών και την γενική μετάβαση της αυτοκινητοβιομηχανίας στην ηλεκτροκίνηση, έχουν οδηγήσει στην αύξηση των απαιτήσεων για περισσότερο ακριβείς και γρήγορες προσομοιώσεις εξωτερικής αεροδυναμικής. Η παρούσα διπλωματική εργασία έχει ως στόχο την σύνθεση μιας μεθοδολογίας προσομοιώσεων που να καλύπτει τις προαναφερθείσες προδιαγραφές, και να έχει εφαρμογή σε σύγχρονα επιβατικά οχήματα. Η μεθοδολογία αυτή περιλαμβάνει τόσο τα στάδια δημιουργίας του υπολογιστικού πλέγματος όσο και τα στάδια της επίλυσης. Η γεωμετρία που επιλέχθηκε και για την οποία διατίθενται τα ανάλογα πειραματικά δεδομένα για σύγκριση με τα αποτελέσματα των προσομοιώσεων, είναι το SAE Notchback Closed Cooling DrivAer model. Στα πλαίσια της διπλωματικής διεξήχθη έρευνα απεξάρτησης των αποτελεσμάτων από το υπολογιστικό πλέγμα, χρησιμοποιώντας τον τύπο πλέγματος “Poly-Hexcore” στο Ansys Fluent Meshing. Για την υπολογιστική επίλυση χρησιμοποιήθηκε το πρόγραμμα Ansys Fluent, τόσο για την αρχική μόνιμη προσομοίωση RANS, όσο και για την προσομοίωση Scale Resolving Simulation (SRS) με χρήση του υπολογιστικού μοντέλου SBES. Το πλέγμα μεσαίας πύκνωσης, με συνολικό αριθμό κελιών 178 εκατομμυρίων, κρίθηκε επαρκές για την αποτύπωση των περισσότερων φαινομένων που λαμβάνουν χώρα στην ροή γύρω από το όχημα, και τα αποτελέσματα για την μέση τιμή των αεροδυναμικών δυνάμεων συμφωνούσαν με τα αντίστοιχα αποτελέσματα που προέκυψαν από το πιο πυκνό πλέγμα συνολικού αριθμού κελιών 450 εκατομμυρίων. Επίσης, τα αποτελέσματα της προσομοίωσης του μεσαίου πλέγματος ήταν πολύ κοντά στις πειραματικές τιμές. Η τελική μεθοδολογία προσομοιώσεων, που χρησιμοποιεί τον ελάχιστο αριθμό επαναλήψεων για την προσομοίωση RANS και τον ελάχιστο αριθμό χρονικών βημάτων για την προσομοίωση SBES που απαιτούνται για την απόκτηση σταθερής μέσης τιμής των αποτελεσμάτων, προσφέρει σημαντική επιτάχυνση της διαδικασίας προσομοίωσης των αεροδυναμικών ροών. Κατά την διάρκεια του σχεδιαστικού κύκλου, οι μηχανικοι χρησιμοποιώντας την προτιθέμενη μεθοδολογία μπορούν να πραγματοποιήσουν έως και 11 προσομοιώσεις εντός μίας εβδομάδας με χρηση 2048 CPU πυρήνων.

Λέξεις-κλειδιά: Αεροδυναμική, Αυτοκινητοβιομηχανία, CFD, RANS, SRS, DrivAer

FUNDING

This thesis has been funded completely by Ansys Inc. and was conducted at the Ansys Hellas office located in Chalandri, Greece.

Table of Contents:

List of Tables:

List of Figures:

Nomenclature

Γ_φ	Diffusion coefficient for φ	
δ	Boundary layer thickness	m
ε	Turbulent dissipation rate	m^2/s^3
μ	Dynamic viscosity	kg/ms
μ_t	Eddy viscosity	kg/ms
ν	Kinematic viscosity	m^2/s
ρ	Density	kg/s
φ	Scalar	
ω	Specific turbulent dissipation rate	1/s
A	Area	m^2
Cd	Drag coefficient	
Cl	Lift coefficient	
Clf	Front lift coefficient	
Clr	Rear lift coefficient	
Cp	Pressure coefficient	
Cpt	Total pressure coefficient	
f_s	SBES shielding function	
k	Turbulent kinetic energy	m^2/s^2
l	Length	m
Lt	Characteristic length	m
p	Pressure	Pa
Re	Reynolds number	
S_φ	Source of φ per unit volume	
$\overline{S_{ij}}$	Mean rate of strain tensor	
τ_{ij}	Reynolds stress tensor	
τ_w	Wall shear stress	Pa
U	Velocity	m/s
U^+	Non-dimensional near wall velocity	
u_τ	Friction velocity	m/s
$\bar{u}$	Mean velocity component	m/s
V	Volume	m^3
y^+	Non-dimensional distance from the wall	

Acronyms

BCD	Bounded Central Differencing
CBC	Convection Boundedness Criterion
CFD	Computational Fluid Dynamics
DES	Detached Eddy Simulation
DNS	Direct Numerical Simulation
DDES	Delayed Detached Eddy Simulation
GEKO	Generalized k-ω
LES	Large Eddie Simulation
SBES	Stress-Blended Eddy Simulation
SRS	Scale Resolving Simulations
SST	Shear Stress Transport

1. Introduction

1.1 The need for simulation

As the automotive industry moves rapidly toward electrification, aerodynamics has become a major component in the design process as it affects the achievable range for a given set of batteries. Furthermore, the new set of regulations regarding aerodynamic devices implemented by the European Commission in 2019, Directive 96/53/EC [17], state the importance of improved aerodynamic efficiency of road vehicles, following specific authorized dimensions, as a way to reduce fuel consumption and CO2 emissions. The safety of retractable or foldable aerodynamic rear devices especially, must be tested and ensured by car manufacturers under various conditions and speed limits.

Considering the complexity, high energy consumption and cost of the more traditional aerodynamic tests in wind tunnels, the automotive industry has turned towards simulation as a means to validate aerodynamic design for multiple variants and flow conditions. Computational Fluid Dynamics (CFD) codes have been developed since the 1950s and have been used in various industries both for internal and external fluid flows. In order to accurately resolve the turbulent flow structures and separation in the flow surrounding a vehicle, which are the main cause of drag force generation and thus affect the overall aerodynamic efficiency, Scale Resolving Simulations (SRS) have become the current trend among engineers. These transient simulations require high levels of mesh refinement, significant computational power, long computing hours and have a greater computational cost than steady state RANS simulations, which were the industry standard until recently. The biggest difference between RANS and SRS simulations lies in the modeling of flow structures and how they account for unsteadiness in the flow. A new method has been introduced which blends the two, and thus reduces the computational cost and time of the simulation.

The purpose of the current thesis is to construct a fast simulation workflow for automotive aerodynamic cases, using the mixed method of blending between RANS and SRS models. A mesh independence study was carried out to define the optimal computational mesh. Then the minimum number of iterations and timesteps was defined that would provide stable averaged results for the aerodynamic forces. The workflow includes both the meshing steps and the computation.

The geometry used for this thesis is the DrivAer model, an industry standard for CFD validation studies, for which an extensive amount of reliable experimental data exists.

The current thesis was conducted during an internship program at Ansys Hellas. The main tools used for the simulations were the Ansys Fluent Meshing and the Ansys Fluent solver.

1.2 DrivAer model description

The coordinate system used throughout this thesis is shown in Figure 1 below. Positive x is the direction of the flow, z is the direction normal to the ground and y is the direction parallel to the front and rear wheel axis of the vehicle.

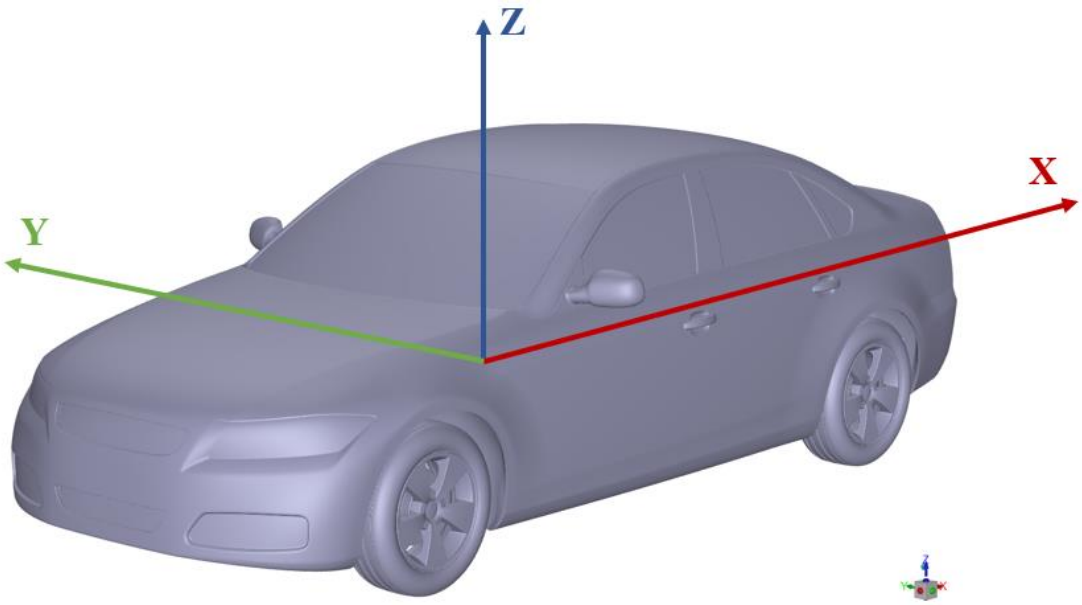


Figure 1: Coordinate system for the DrivAer case

The geometry used in the current thesis is the DrivAer model. The DrivAer model was introduced in 2012 by Heft et al. [3] and has since become the standard generic aerodynamic benchmark used for CFD correlation in the automotive industry. The thesis uses experimental data of detailed surface pressure and flow field measurements for the DrivAer – Notchback test case, that was presented at the 2nd Auto CFD prediction workshop by B. Hupertz [1]. Further information for the wind tunnel experiment for the DrivAer model are mentioned in [2].

The geometry consists of the notchback version of the DrivAer in the so called “closed cooling” configuration and is shown in Figure 2.

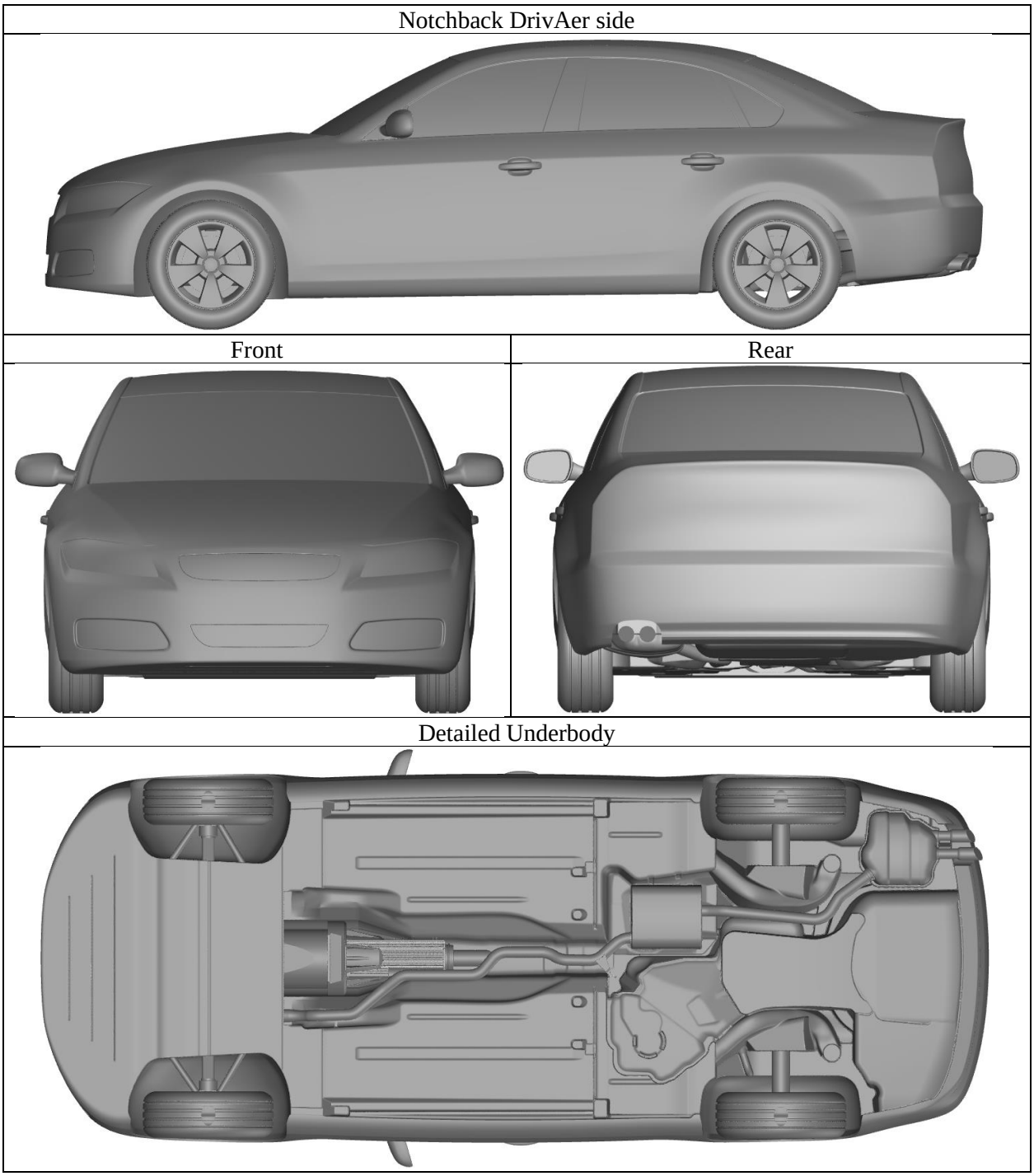


Figure 2: Closed Cooling Notchback DrivAer model geometry.

The wind tunnel testing on the DrivAer was conducted in the Pininfarina full-scale automotive wind tunnel, shown in Figure 3, with static floor and static rigid model tires. The open return wind tunnel consists of an open jet test section with an 11 (m^2) nozzle. The boundary layer control system was active. The configuration of the wind tunnel floor is shown in Figure .

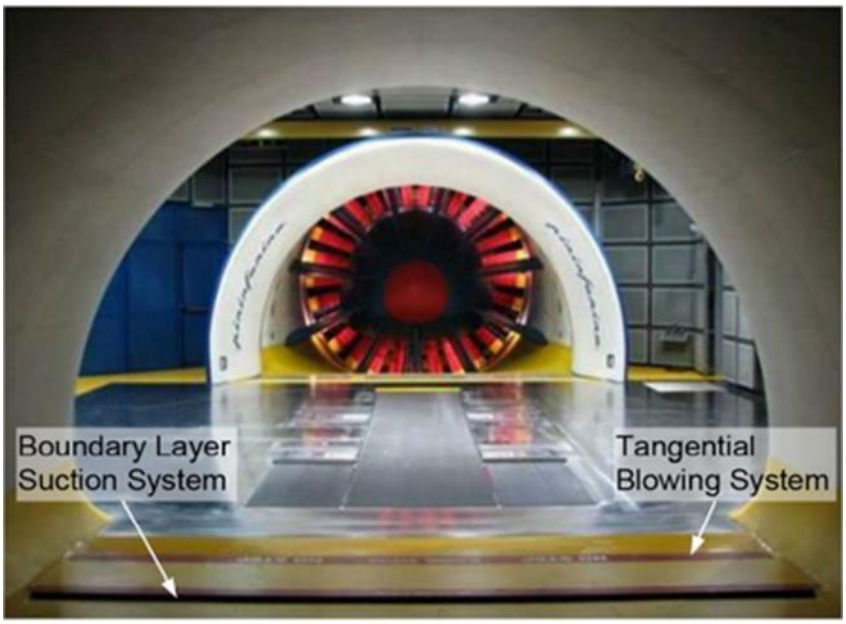


Figure 3: Pininfarina wind tunnel [1].

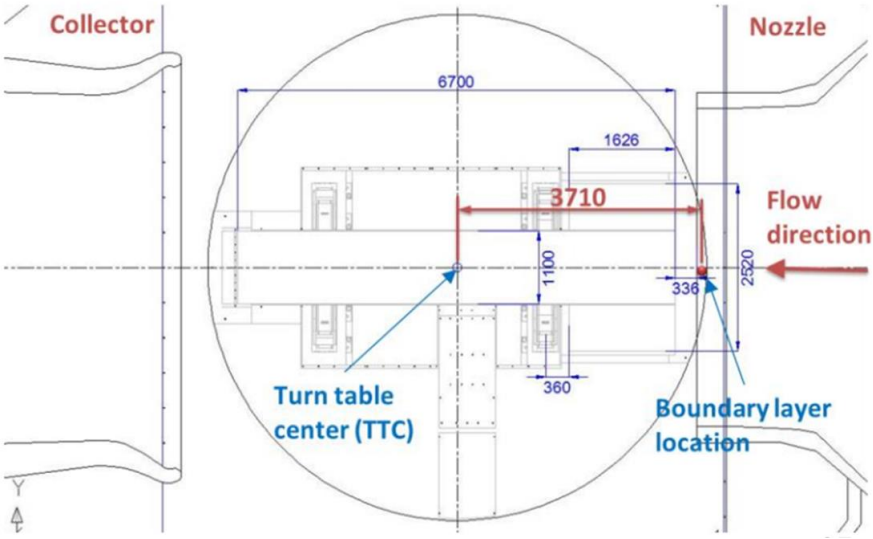


Figure 4: Pininfarina wind tunnel floor [5].

Measurements in the empty test section of the Pininfarina wind tunnel indicate a boundary layer thickness at the center of the turntable δ_{TTC} equal to 55mm. The theoretical starting point of the fully turbulent boundary layer is calculated by the following equation, [5]:

$$\delta_{TTC}(x) = \frac{0.37 x_{BL}^{4/5} v^{1/5}}{U_{\infty}^{1/5}} \tag{1.1}$$

For ambient conditions in the Pininfarina wind tunnel and a wind speed of U_{∞} =140kph the boundary starting point has been calculated as x_{BL}=− 2.339 (m), or 3.710 (m) upstream from the turntable center [5].

Figure 3: The Ford 1:1 scale notchback Open Cooling DrivAer model [1].

Figure 4: Grill insert based closure of the front grills [1].

The ride height positioning of the model in the wind tunnel, shown in Figure 5, was for the front ride height 686 (mm) and for the rear ride height 682 (mm). The wheel center was located 317.6 (mm) above the ground.

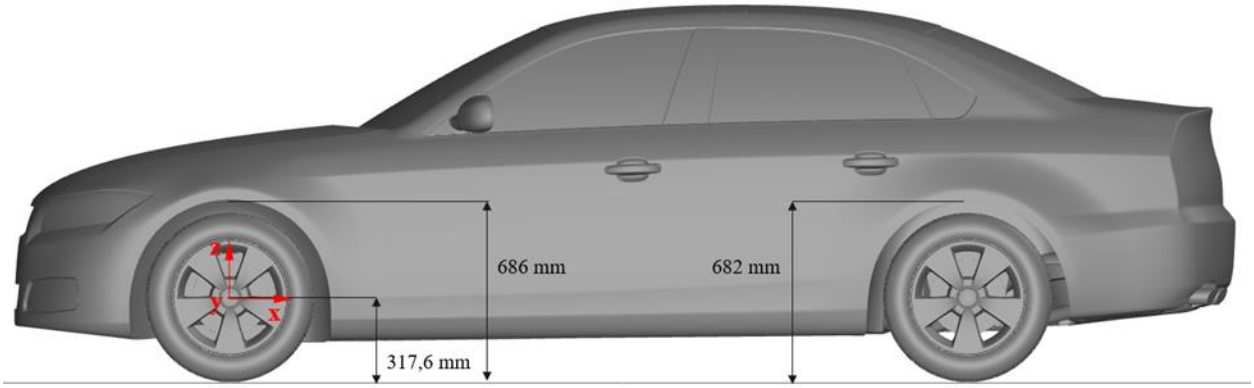


Figure 5: DrivAer wind tunnel test setup.

The test conditions are described in the following table:

Table 1:Wind tunnel test conditions [67]

Vehicle speed	140 (kph)
Yaw angle	0°
Turbulence intensity	0.26%
Turbulent length scale	5 (m)
Ambient air pressure	101325 (Pa)
Ambient air density	1,204 (kg/m^3)
Dynamic viscosity	1.8138E-5 (Pa·s)

1.3 Following chapters description

The following chapters of this thesis include a brief overview of the aerodynamics theory for automotive cases, the numerical and CFD theory that applies to external aerodynamic flow simulations, the numerical setup including both mesh and solver characteristics, results from the mesh dependency study, comparison of simulation results with experimental data for the DrivAer Notchback model, and finally, remarks and ideas for future research.

2. Aerodynamics and CFD Theory

The theory section of this thesis is separated into two categories. The first includes a brief description of the aerodynamics of conventional vehicles and the definition of the aerodynamic forces and coefficients. The second refers to Computational Fluid Dynamics theory. It begins with the RANS (Reynolds Averaged Navier Stokes) concept, continues with analyzing the numerical methods in the Ansys Fluent solver used for the simulations of the thesis and lastly, it includes a description of Scale Resolving Simulations and the SBES model.

2.1 Car Aerodynamics

Modern automotive vehicles are considered complex aerodynamic devices operating in close proximity to the ground. As the vehicle moves through the surrounding air, forces are generated on the vehicle. The greatest effect on the dynamics of the vehicle on the road come from drag and lift forces. The drag force opposes forward motion, acceleration and deceleration, as well as the maximum speed. The lift force directly affects the ability of the vehicle to stay on the ground and of the vehicle's tires to generate grip and provide tractive forces during acceleration, braking and cornering. Drag and lift forces will be analyzed in the following sections of the thesis.

When studying automotive aerodynamics, the "principle of relative motion" is being used, according to which the movement of a body through the air has the same effect as the movement of air over a stationary body.

2.1.1 Aerodynamic forces

When a vehicle is in relative motion to the surrounding air, it generates an aerodynamic force in a rearward direction, at an angle determined by the direction of relative motion. This aerodynamic force can be broken down into two components, both acting through the center of pressure, the drag force and the lift force (or its negative downforce).

Drag is the net force component parallel to the direction of relative motion. The different contributions to drag are:

- Pressure drag (or form drag) due to the pressure field generated because of the shape and the size of the body
- Viscous drag (or skin friction drag) due to the friction generated by the vehicle surface on the fluid

Drag force is proportional to the square of the vehicle speed:

$$Drag = C_d \frac{1}{2} \rho A U^2 \tag{2.1}$$

Where:

- C_d is the drag force coefficient
- ρ is the density of the air around the vehicle
- A is the frontal area of the vehicle

- U is the vehicle speed

Lift is the net force component perpendicular to the direction of relative motion.

$$Lift = C_l \frac{1}{2} \rho A U^2 \tag{2.2}$$

Where:

- C_l is the lift force coefficient

In the present thesis, two components of the lift force are used for correlation with experimental results, front lift force (C_{lf}) which acts on the front axle and rear lift force (C_{lr}) which acts on the rear axle of the DrivAer model. An example of the aerodynamic forces acting on a vehicle is shown in Figure 6.

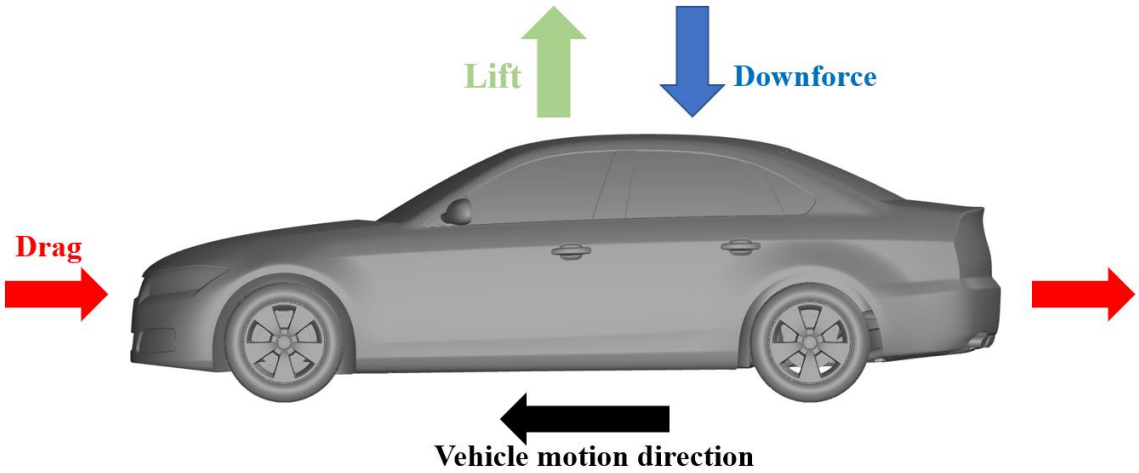


Figure 6: Aerodynamic forces acting on a vehicle.

The DrivAer model has the generic shape of modern passenger vehicles, which can also be characterized as bluff bodies. When these bodies are moving through air, in close proximity to the ground and in relatively low speeds, with the average speed for passenger vehicles being around 30 (km/h) [23], the major aerodynamic force acing on the vehicle is the drag force, which can significantly affect the fuel or energy consumption. Passenger vehicles are generally considered lift-neutral bodies, since they generate very small forces in the direction normal to the ground, while side forces are also usually neglected when considering straight line motion.

The side force and the moments acting on a vehicle are shown in Figure 7.

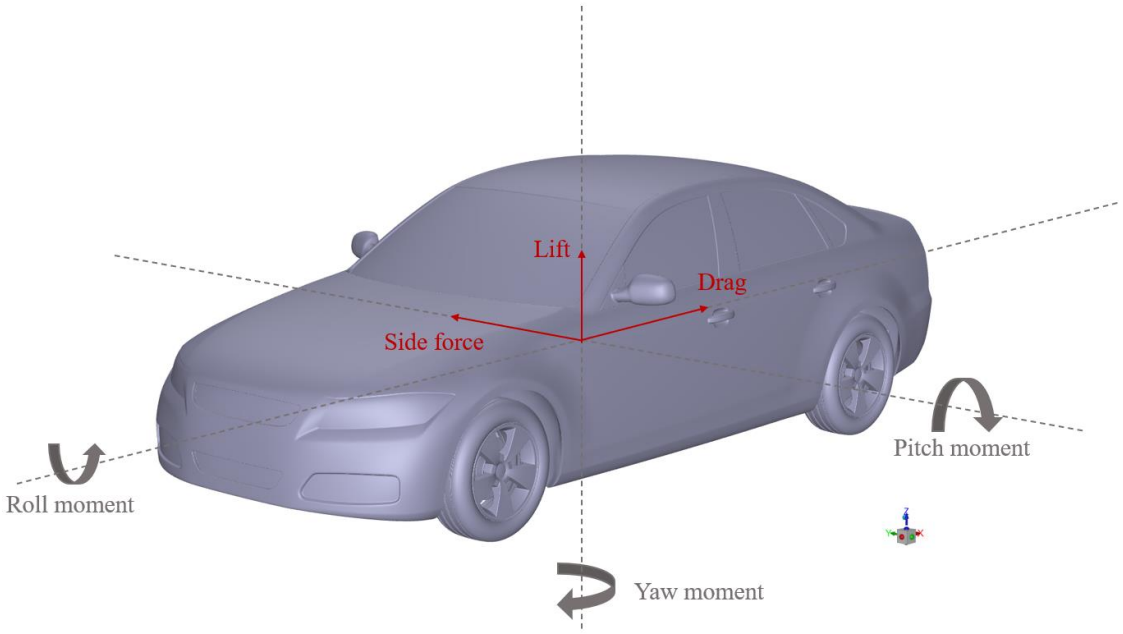


Figure 7: Aerodynamic moments and forces acting on a vehicle.

2.1.2 Flow separation

A major contributor to the overall generation of drag force is flow separation over the surfaces of the vehicle. In order to describe how flow separation occurs, the concept of a boundary layer needs to be introduced.

The boundary layer is formed as the fluid moves past an object and the molecules right next to the surface stick to it because of molecular viscosity, causing the fluid flow to decelerate. The fluid tends to flow over a surface as if it were in thin layers. The deceleration of the fluid layer right next to the surface is then transferred successively to the above flow layers by viscous shear stresses acting between them. The further a layer is from the surface, the weakest those stresses get and finally the flow velocity becomes equal to the free stream velocity. The resulting velocity profile for a laminar boundary layer is shown in Figure 8. The thickness of the boundary layer δ is defined as the distance normal to the surface, from the surface up to the point where the velocity becomes equal to 99% of the free stream value.

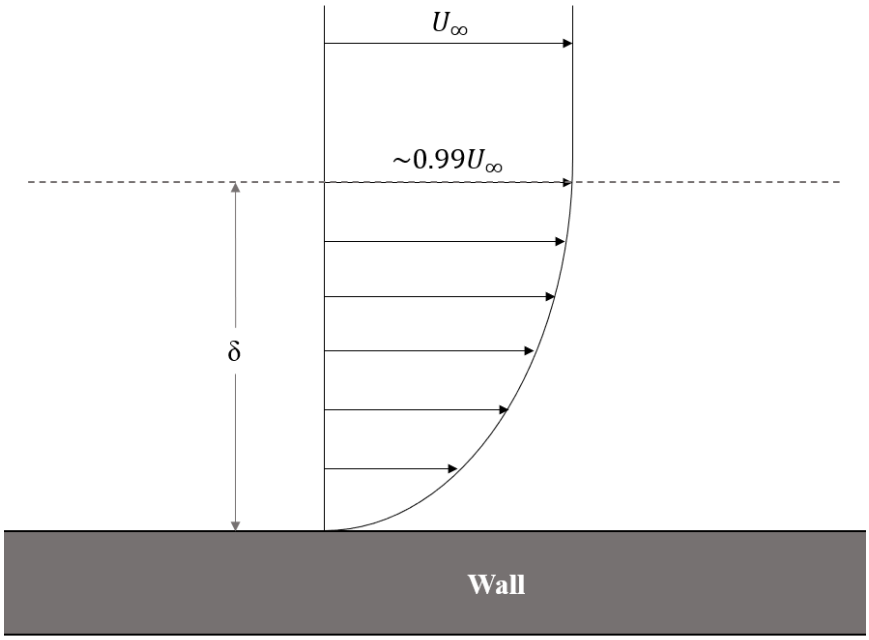


Figure 8: Velocity profile of a laminar boundary layer.

The boundary layer is laminar for lower Reynolds numbers, and the streamwise velocity changes as shown in the left-hand side of Figure 10: Velocity profiles for laminar and turbulent regions of the boundary layer. The particles within the fluid layers of the boundary layer are moving in the same direction as the freestream flow. As the Reynolds number increases (with x), the flow becomes unstable and the viscous effects remove energy from the flow. Finally, the boundary layer becomes turbulent for higher Reynolds numbers, and the streamwise velocity is characterized by unsteady (changing with time) swirling three-dimensional flow structures inside the boundary layer. This transition from laminar to turbulent boundary layer flow is shown in Figure 9. The transition occurs when Reynolds number at x exceeds Rex ~ 500,000.

The Reynolds number, which is the ratio of inertial forces to viscous forces plays a critical role in the formation of the boundary layer.

$$Re = \frac{\rho U l}{\mu} \tag{2.3}$$

The boundary layer begins from a stagnation point as the free stream flow reaches the surface of a flat plate and gradually increases in thickness due to molecular diffusion. Further downstream the flow gradually transitions from laminar to turbulent. The transition point can be influenced by free stream disturbances, pressure gradients etc.

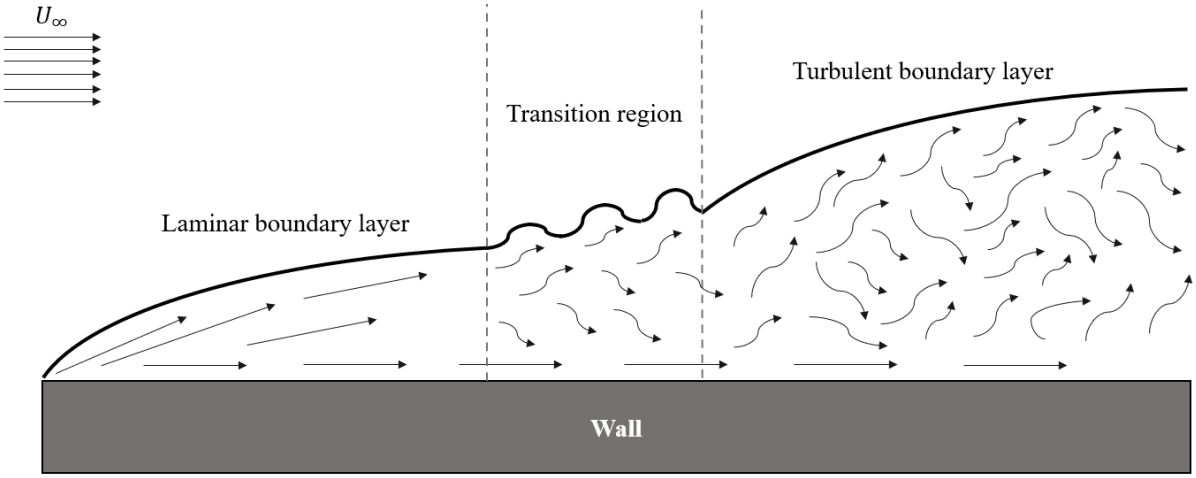


Figure 9: Boundary layer transition from laminar to turbulent.

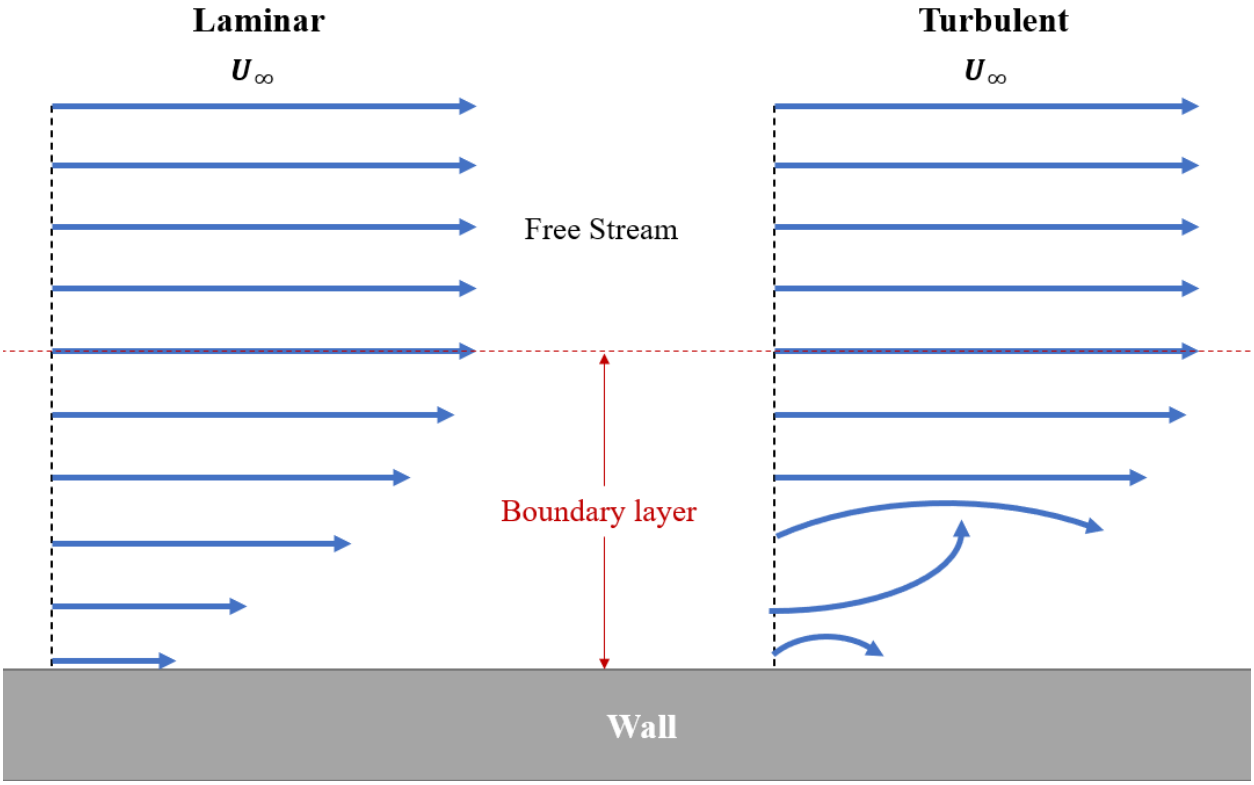


Figure 10: Velocity profiles for laminar and turbulent regions of the boundary layer.

The flow in the boundary layer is often subject to pressure variations due to change in the surface curvature or external forces. When the pressure decreases in the flow direction, it is known as a favorable or negative pressure gradient, which leads to acceleration of the flow and keeps it attached to the surface. But when the

pressure decreases in the direction of the flow, it is called an adverse or positive pressure gradient. The fluid "struggles" to flow against an adverse pressure gradient, the velocity decreases and when it becomes zero, the flow reverses in the opposite direction, as shown in Figure 11. The point where the wall shear stress becomes zero is called separation point. The thickness of the boundary layer increases and gets detached from the surface. This is called boundary layer separation and it is one of the contributors to the generation of pressure drag.

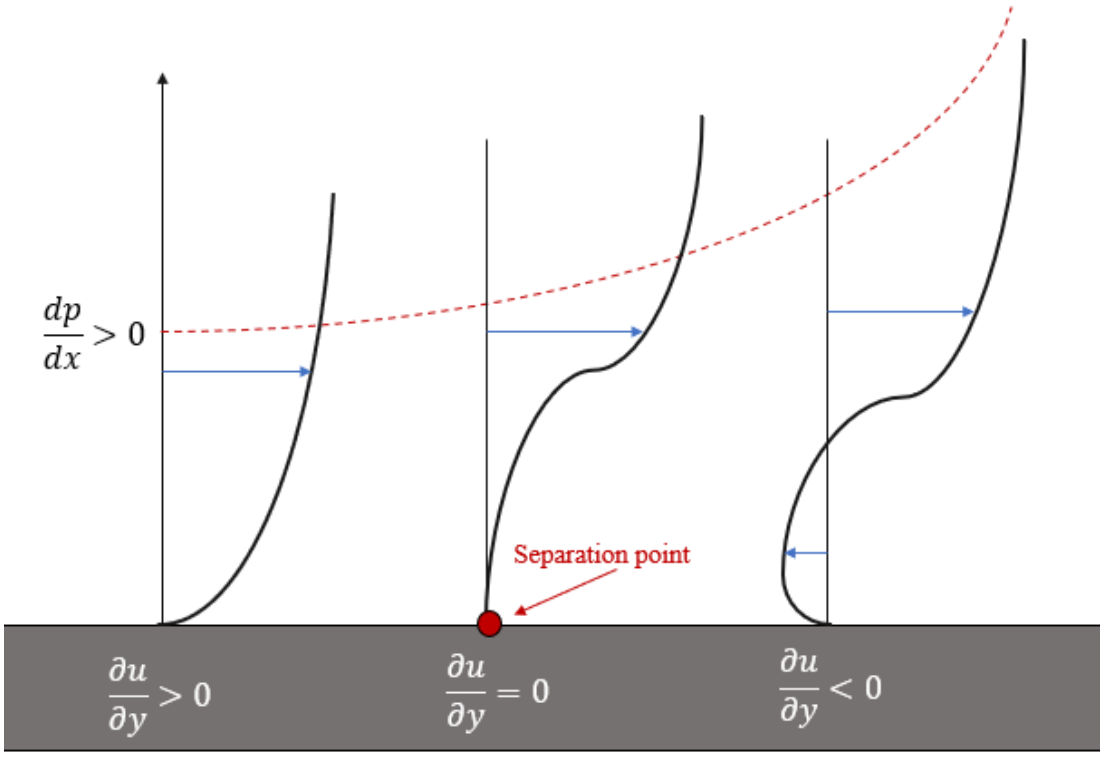


Figure 11: Boundary layer separation.

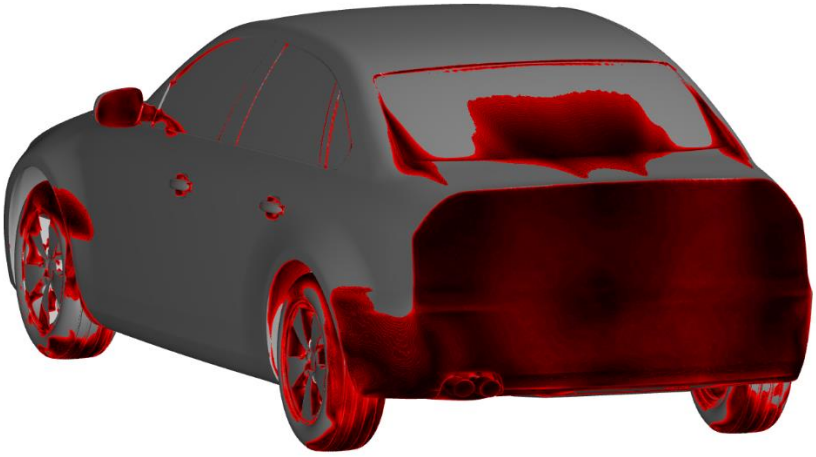
In general, turbulent flows are less susceptible to separation under adverse gradients than laminar flows due to higher momentum transfer normal to the wall. Thus, a turbulent boundary layer could sometimes delay separation. It is also possible for a separated flow to reattach itself to the surface. This can happen either when a favorable pressure gradient is recovered past the separation point, or, in the case of a separated laminar flow, when it transitions to turbulent, then it has a better chance of reattachment.

Further information on the above topic can be found in the references: [8], [14], [20] and [24].

In commercial vehicles, flow separation usually occurs where there is a sharp lip and rapid change in the angle of an aerodynamic surface relative to the flow direction. The main regions of separation are located on the rear window or backlight, and the rear end, behind the mirrors and in various locations on the underbody and the wheel housing. Separation also occurs at the stagnation region of the flow at the front of the vehicle. An example of flow separation regions on the DrivAer model is shown in Figure 12. This image comes from a simulation conducted during the research of the current thesis. The regions of separation are in red color and indicate areas on the surface of the DrivAer model characterized by a value of wall shear stress magnitude equal to zero.

Front View

Rear View



Detailed Underbody

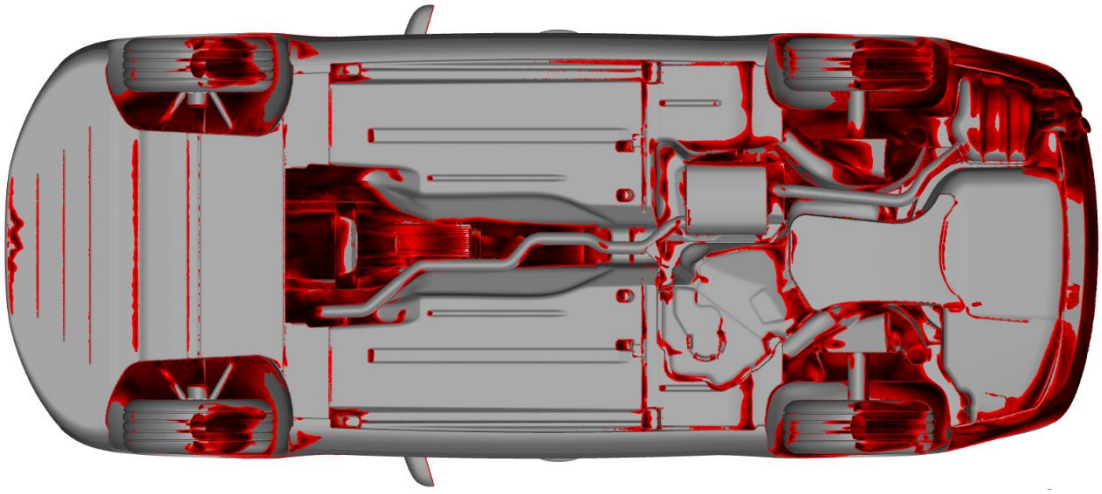


Figure 12: Regions of separation (red) on the Closed Cooling DrivAer Notchback model.

2.1.3 Wake

When the fluid over a surface of a body is fully separated and cannot be reattached to it, it results in a region of a decelerated flow. This region of the flow around a body is called a wake and it is characterized by large eddies and vortices that shed downstream. Large energy dissipation and total pressure losses in that region are major contributors to drag force generation. An example of separated flow behind a vehicle is shown in Figure 13.

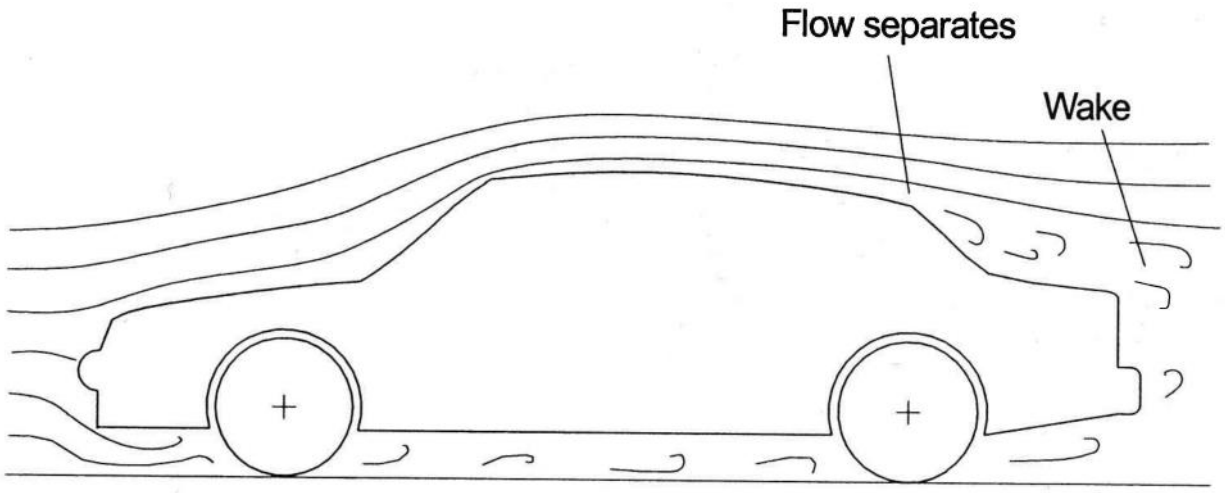
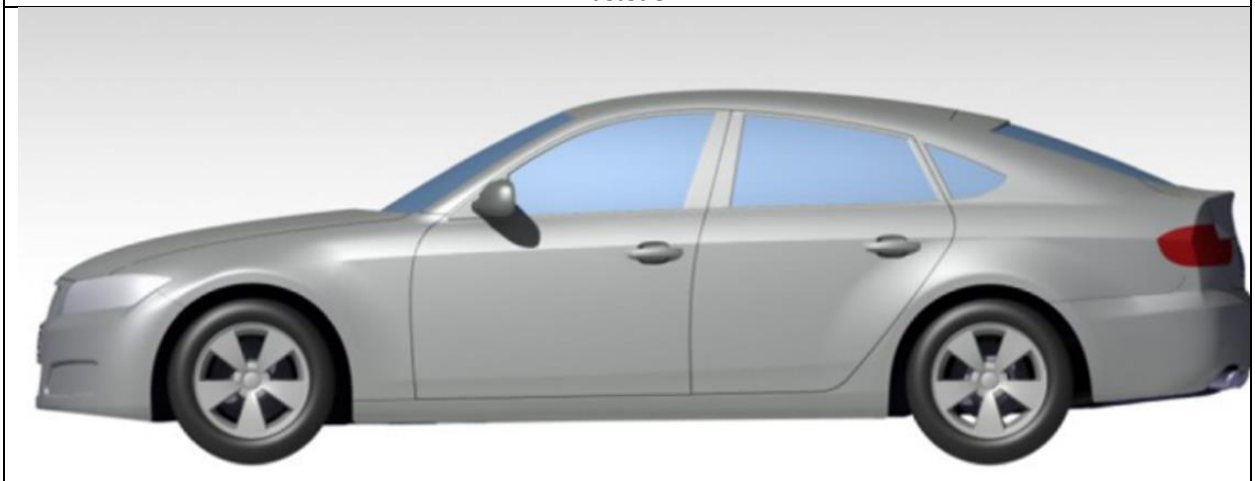


Figure 13: Separated flow over a generic shaped vehicle [25].

In the automotive industry, one of the biggest challenges that engineers face is the reduction of the wake region when designing the aerodynamic surfaces of a car. Throughout the years many different aerodynamic shapes have been introduced in an attempt to find the most efficient one.

For the DrivAer model three different shapes of the rear-end have been introduced: the fastback, the notchback, and the estate back. The different rear-end configurations of the DrivAer model are shown in Figure 14, while the different wake formations are shown in Figure 15.

Fastback



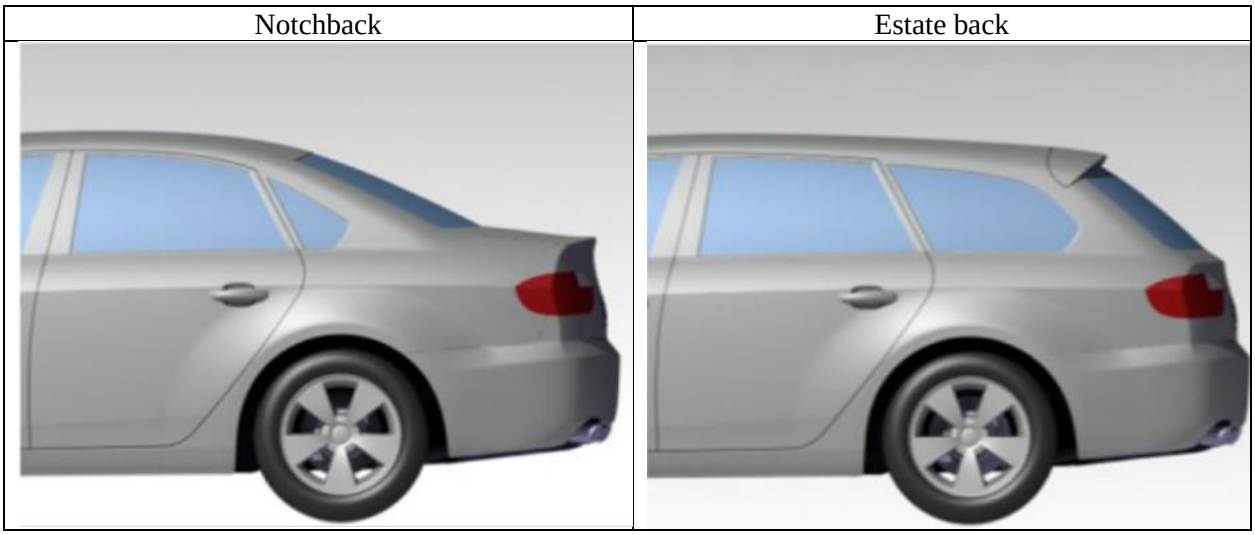


Figure 14: DrivAer model with different rear configurations [3].

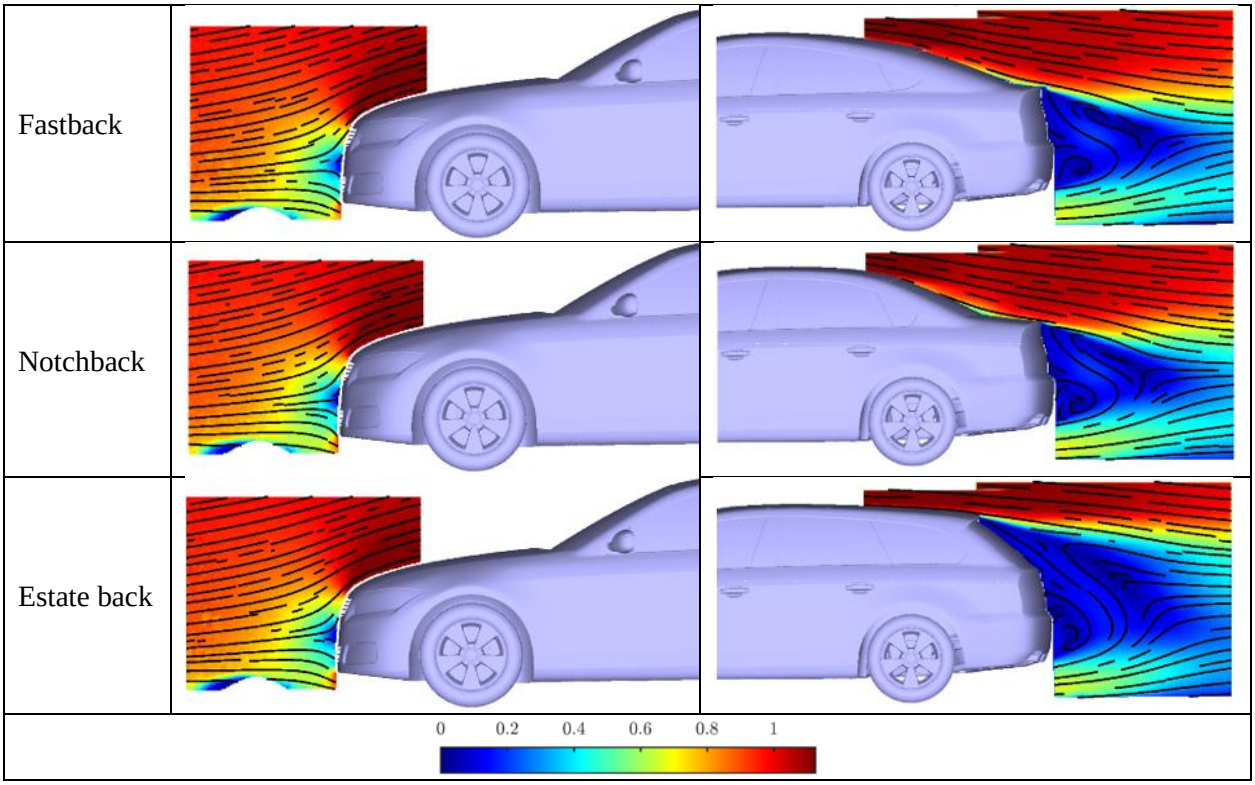


Figure 15: Normalized velocity magnitude contours at y = 0 (mm) on the three different DrivAer rear configurations [18].

In the present thesis the Notchback variant of the DrivAer model was used.

2.1.4 Pressure coefficient

Before moving to the next theory section, the pressure coefficient should be mentioned, which will be used extensively in the post processing of the simulation results. The pressure coefficient, C_p, refers to the static pressure and is expressed as:

$$Cp = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty U_\infty} \tag{2.4}$$

Where:

- p: local static pressure
- p_∞: static pressure of the freestream flow
- ρ_∞: density of the freestream flow
- U_∞: velocity free stream flow

2.2 CFD Methodology

2.2.1 Navier Stokes equations

The continuity equation for a finite control volume fixed in space in partial differential form is given as:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}) = 0 \tag{2.4}$$

The x, y, z components of the momentum equation in differential form are:

$$\frac{\partial (\rho u)}{\partial t} + \nabla \cdot (\rho u \boldsymbol{V}) = -\frac{\partial p}{\partial x} + \rho f_x + (F_x)_{viscous} \tag{2.5}$$

$$\frac{\partial (\rho v)}{\partial t} + \nabla \cdot (\rho v \boldsymbol{V}) = -\frac{\partial p}{\partial y} + \rho f_y + \left(F_y\right)_{viscous} \tag{2.6}$$

$$\frac{\partial (\rho w)}{\partial t} + \nabla \cdot (\rho w \boldsymbol{V}) = -\frac{\partial p}{\partial z} + \rho f_z + (F_z)_{viscous} \tag{2.7}$$

Where the first term in the left-hand side of all three equations above represents the time rate of change of momentum due to unsteady fluctuations of flow properties inside the control volume, the second term is the net flow of momentum out of control volume across the control surface. The first term in the right-hand side of the above momentum equations represents the pressure forces, the second term the body forces and the third term the total viscous forces.

Considering incompressible flow, the continuity equation can be written as:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.8}$$

the Navier-Stokes equations take the following form:

$$\rho\left(\frac{\partial u}{\partial t}+\mathrm{u}\frac{\partial u}{\partial x}+\mathrm{v}\frac{\partial u}{\partial y}+\mathrm{w}\frac{\partial u}{\partial z}\right)=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right) \quad (2.9)$$

$$\rho\left(\frac{\partial v}{\partial t}+\mathrm{u}\frac{\partial v}{\partial x}+\mathrm{v}\frac{\partial v}{\partial y}+\mathrm{w}\frac{\partial v}{\partial z}\right)=-\frac{\partial p}{\partial y}+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right) \quad (2.10)$$

$$\rho\left(\frac{\partial w}{\partial t}+\mathrm{u}\frac{\partial w}{\partial x}+\mathrm{v}\frac{\partial w}{\partial y}+\mathrm{w}\frac{\partial w}{\partial z}\right)=-\frac{\partial p}{\partial z}+\mu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right) \quad (2.11)$$

The left-hand side of the equations represent the rate of momentum change. The first component on the right-hand side represents the isotropic forces, due to the pressure field, and the second term represents the Newtonian Viscous Stress, due to fluid viscosity and shear stresses. More on the derivation of the above equations can be found in [A2. Derivation of the Navier Stokes equation].

Details on the Navier Stokes equations can also be found in reference: [8] and [9].

In the following sections, all the information is based on the Ansys Fluent Theory Guide [9].

2.2.2 Reynolds-Averaged Navier–Stokes Equations

The Reynolds-Averaged Navier–Stokes equations (RANS) are time-averaged equations of motion for fluid flow and are used to describe turbulent flows.

In order to simulate large-scale dynamics of the flow field, averaging is performed over the small-scale fluctuations along with modeling of the nonlinear influence from the small-scale fluctuations in the governing equations, that can alter the large-scale fluid motion. To average a velocity field u_i that has fluctuating components, time averaging can be used for a statistically steady flow.

After applying the Reynolds decomposition technique (described in [A3. Reynolds decomposition]), the RANS equations are derived as:

$$\underbrace{\rho\left(\frac{\partial \bar{u}}{\partial t}+\bar{u}\frac{\partial(\bar{u})}{\partial x}+\bar{v}\frac{\partial(\bar{u})}{\partial y}+\bar{w}\frac{\partial(\bar{u})}{\partial z}\right)}_{\boldsymbol{A}}+\underbrace{\rho\left(\frac{\partial\overline{(u'u')}}{\partial x}+\frac{\partial\overline{(v'u')}}{\partial y}+\frac{\partial\overline{(w'u')}}{\partial z}\right)}_{\boldsymbol{B}} \quad (2.12)$$
$$=\underbrace{-\frac{d\bar{p}}{dx}}_{\boldsymbol{C}}+\underbrace{\mu\left(\frac{\overline{\partial^2(\bar{u})}}{\partial x^2}+\frac{\overline{\partial^2(\bar{u})}}{\partial y^2}+\frac{\overline{\partial^2(\bar{u})}}{\partial z^2}\right)}_{\boldsymbol{D}}$$

- A: Rate of Momentum change.
- B: Reynolds Stress, due to fluid flow fluctuations
- C: Isotropic Stress, due to pressure field
- D: Newtonian Viscous Stress, due to viscosity and shear stress.

RANS equation in tensor form:

$$\rho \frac{\partial \overline{u_i}}{\partial t} + \rho \overline{u_j} \frac{\partial \overline{u_i}}{\partial x_j} = \left(\rho \overline{f_i}\right) + \frac{\partial}{\partial x_j}\left[-\bar{p}\delta_{ij} + 2\mu \overline{S_{ij}} - \rho(\overline{u_i' u_j'}\right) \tag{2.13}$$

where:

- $\overline{S_{ij}} = \frac{1}{2}\left(\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i}\right)$ is the mean rate of strain tensor.
- $\rho(\overline{u_i' u_j'})$ is the Reynolds stress.

The value of the Reynolds stress needs to be calculated to find the value of the mean velocity component from the RANS equations. This is considered a closure problem.

2.2.3 Turbulence Models

Turbulence models are mathematical models that predict the effects of turbulence. Since the averaged flow and turbulent fluctuations are coupled due to the non-linear nature of turbulence, the effect of turbulence appears in the averaged governing equations as turbulent stress. The model for such stress is referred to as a turbulence model.

In a flow, large-scale vortices are strongly influenced by the boundary geometry, while the small-scale vortices are more universal, isotropic, and dissipative. Capturing the large-scale vortical fluctuations is important since they affect the average flow quantities, while the full turbulence spectrum is not of interest in most applications.

The choice of turbulence model will depend on considerations such as the flow features involved, the established practice for a specific class of problem, the level of accuracy required, the available computational resources, and the amount of time available for the simulation.

2.2.4 Boussinesq Approach

A common method of modeling the Reynolds stresses in the RANS equations employs the Boussinesq hypothesis to relate the Reynolds stresses to the mean velocity gradients. The momentum transfer caused by turbulent eddies can be modeled with an eddy viscosity. The Boussinesq assumption states that the Reynolds stress tensor, τ_{ij}, is proportional to the trace-less mean strain rate tensor, S_{ij}*, and can be written in the following way:

$$\tau_{ij} = 2\mu_t S_{ij}^* - \frac{2}{3}\rho k \delta_{ij} \tag{2.16}$$

where μ_t is a scalar property called the eddy viscosity. The same equation can be written more explicitly as:

$$-\rho\left(\overline{u_i' u_j'}\right) = \mu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) - \frac{2}{3}\rho k \delta_{ij} \tag{2.17}$$

2.2.5 GEKO model

The GEKO model [13] is based on a two-equation k-ω formulation and features four free coefficients that can be tuned within given limits, without negative effects on the underlying calibration for wall boundary layers at zero pressure gradient, as well as channel and pipe flows. Two equation RANS models for k-ε and k-ω formulations are mentioned in [A4. Turbulence models].

It is not possible to cover all flows with sufficient accuracy using a single RANS model, but at the same time, finding the right model for a particular application may be challenging, as not all models are of the same quality in terms of robustness, interoperability with other models and near wall treatment. The GEKO model is an alternative approach that offers a single model with enough flexibility to cover a wide range of applications, based on two-equation models. This is different from classical models, where the coefficients of the model are provided but cannot be justifiably changed by the user because they are inherently intertwined, and any change would typically lead to a loss of calibration.

The GEKO model coefficients are:

- C_{SEP} - Parameter to optimize flow separation from smooth surfaces.

  $0.7< C_{SEP} <2.5$ (default $C_{SEP}= 1.75$)

  Increasing C_{SEP} leads to earlier and stronger separation.

  $C_{SEP}=1$ mimics the standard model, while $C_{SEP}=1.75$ is close in performance to the SST model.

- C_{NW} - Parameter to optimize flow in non-equilibrium near wall regions (such as heat transfer or C_f).

- C_{MIX} - Parameter to optimize strength of mixing in free shear flows.

- C_{JET} - Parameter to optimize free shear layer mixing (optimize free jets independent of mixing layer).

The default values of $C_{SEP} = 1.75$, $C_{MIX} = 0.3$, $C_{NW} = 0.5$, and $C_{JET} = 0.9$ give a close approximation of the k-ω SST model (especially in terms of separation behavior). For free shear flows, this combination is superior compared to SST. For automotive aerodynamics though, the default values tend to lead to overprediction of flow separation on the surfaces of the vehicle, so lower values for the separation coefficient C_{SEP} are recommended. For this thesis the GEKO model with $C_{SEP} = 1$ was used, while the rest of the coefficients were used as default. Tests with different values for C_{SEP} were not carried out during this research, but the current recommended by Ansys value for similar automotive cases was used.

2.2.6 Solver Theory

2.2.6.1 *Overview*

There are two numerical methods available in Ansys Fluent:

- Pressure-based solver
- Density-based solver

In both methods the velocity field is obtained from the momentum equations. In the density-based approach, the continuity equation is used to obtain the density field while the pressure field is determined from the equation of state. This approach was initially developed for high-speed compressible flows.

In the pressure-based approach, the pressure field is extracted by solving a pressure or pressure correction equation which is obtained by manipulating continuity and momentum equations. This approach was developed initially for low-speed incompressible flows. For automotive aerodynamic flows especially, the changes in the pressure field are much larger compared to changes in the density of the flow, so the pressure-based approach is preferred.

Ansys Fluent solves the governing integral equations for the conservation of mass and momentum. In both cases a control-volume-based technique is used that consists of:

- Division of the domain into discrete control volumes using a computational mesh.
- Integration of the governing equations on the individual control volumes to construct algebraic equations for the discrete dependent variables ("unknowns") such as velocities and pressure.
- Linearization of the discretized equations and solution of the resultant linear equation system to yield updated values of the dependent variables

For this thesis the pressure-based solver was used.

2.2.6.2 *Pressure-based solver*

Using the pressure-based solver, the constraint of mass conservation (continuity) of the velocity field is achieved by solving a pressure equation. The pressure equation is derived from the continuity and the momentum equations in such a way that the velocity field, corrected by the pressure, satisfies the continuity. Since the governing equations are nonlinear and coupled to one another, the solution process involves iterations wherein the entire set of governing equations is solved repeatedly until the solution converges.

Two pressure-based solver algorithms are available in Ansys Fluent. A segregated algorithm, and a coupled algorithm.

2.2.6.3 *The Pressure-Based Segregated Algorithm*

In the segregated algorithm, the individual governing equations for the solution variables (for example, u, v, w, p, k, ω) are solved one after another. Each governing equation, while being solved, is decoupled from other equations. The segregated algorithm is memory-efficient, since the discretized equations need only to be stored in the memory one at a time. However, the solution convergence is relatively slow.

With the segregated algorithm, each iteration consists of the steps illustrated in Figure 16.

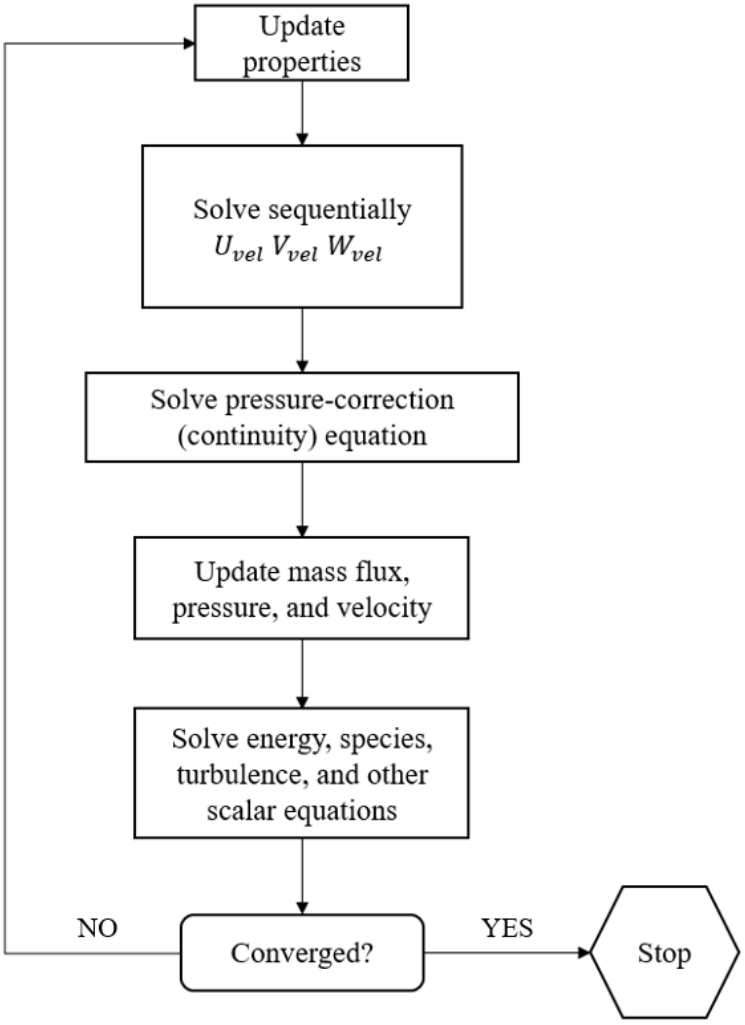


Figure 16: Pressure-based segregated algorithm steps [9].

These steps are continued until the convergence criteria are met.

2.2.6.4 *Coupled Algorithm*

The coupled scheme obtains a robust and efficient single-phase implementation for steady-state flows, with superior performance compared to the segregated solution schemes. Unlike the segregated algorithm, the pressure-based coupled algorithm solves a coupled system of equations comprising the momentum equations and the pressure-based continuity equation. Thus, in the coupled algorithm, Steps 2 and 3 in the segregated solution algorithm are replaced by a single step in which the coupled system of equations are solved. The remaining equations are solved in a decoupled fashion as in the segregated algorithm.

Since the momentum and continuity equations are solved in a closely coupled manner, the rate of solution convergence significantly improves when compared to the segregated algorithm. However, the memory requirement increases by 1.5 – 2 times that of the segregated algorithm since the discrete system of all

momentum and pressure-based continuity equations must be stored in the memory when solving for the velocity and pressure fields (rather than just a single equation, as is the case with the segregated algorithm).

The steps of the pressure-based coupled solver are illustrated in Figure 17.

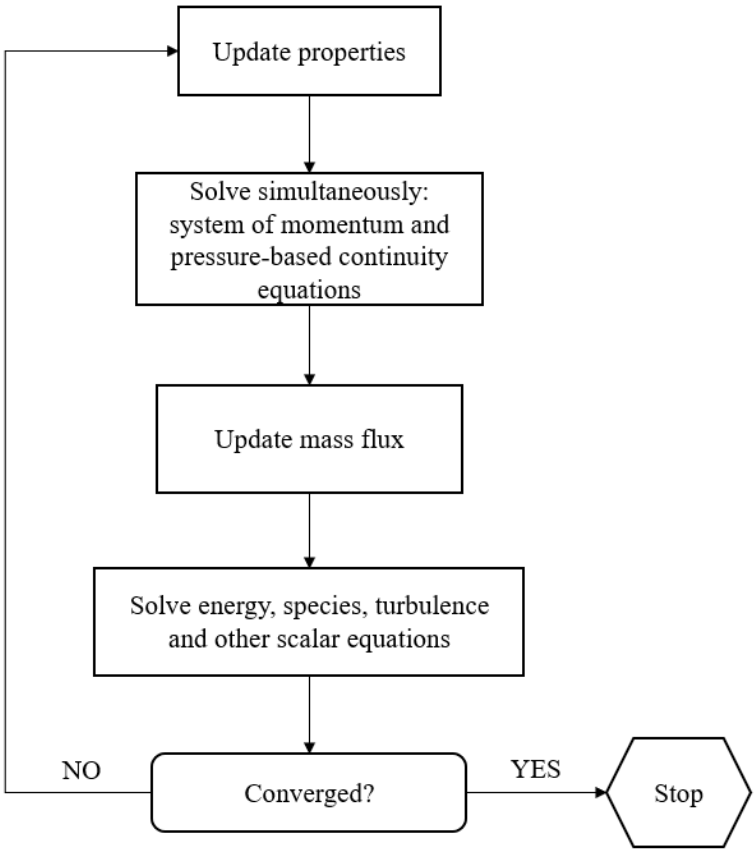


Figure 17: Pressure-based coupled algorithm steps [9].

2.2.6.5 *Discretization*

A control-volume-based technique is used to convert a general scalar transport equation to an algebraic equation that can be solved numerically. This control-volume technique consists of integrating the transport equation about each control volume, yielding a discrete equation that expresses the conservation law on a control-volume basis.

Discretization of the governing equations can be illustrated most easily by considering the unsteady conservation equation for transport of a scalar quantity φ. This is demonstrated by the following equation written in integral form for an arbitrary control volume V as follows:

$$\int_V \frac{\partial \rho \varphi}{\partial t} dV + \oint \rho \varphi \vec{v} \cdot d\vec{A} = \oint \Gamma_\varphi \nabla \varphi \cdot d\vec{A} + \int_V S_\varphi dV \tag{2.18}$$

where:

- ρ = density
- $\vec{v}$ = velocity vector
- $\vec{A}$ = surface vector
- Γ_φ = diffusion coefficient for φ
- $\nabla\varphi$ = gradient coefficient for φ
- S_φ = source of φ per unit volume

The first term in the left-hand side of equation (2.18) is the transient term and the second term is the convection term. The first term in the right-hand side of equation (2.18) is the diffusion term and the second is the source term.

Equation (2.18) is applied to each control volume, or cell, in the computational domain. The two-dimensional, triangular cell shown in Figure 18 is an example of such a control volume. Discretization of equation (2.18) on a given cell yields:

$$\frac{\partial \rho\varphi}{\partial t}V + \sum_{f}^{N_{faces}} \rho_f \overrightarrow{v_f}\varphi_f \cdot \overrightarrow{A_f} = \sum_{f}^{N_{faces}} \Gamma_f \nabla\varphi_f \cdot \overrightarrow{A_f} + S_\varphi V \tag{2.19}$$

Where:

- N_{faces} = number of faces enclosing cell
- φ_f = value of φ convected through face f
- $\rho_f \overrightarrow{v_f} \cdot \overrightarrow{A_f}$ = mass flux through the face
- $\overrightarrow{A_f}$ = area vector of face f
- $\nabla\varphi_f$ = gradient of φ at face f
- V = cell volume

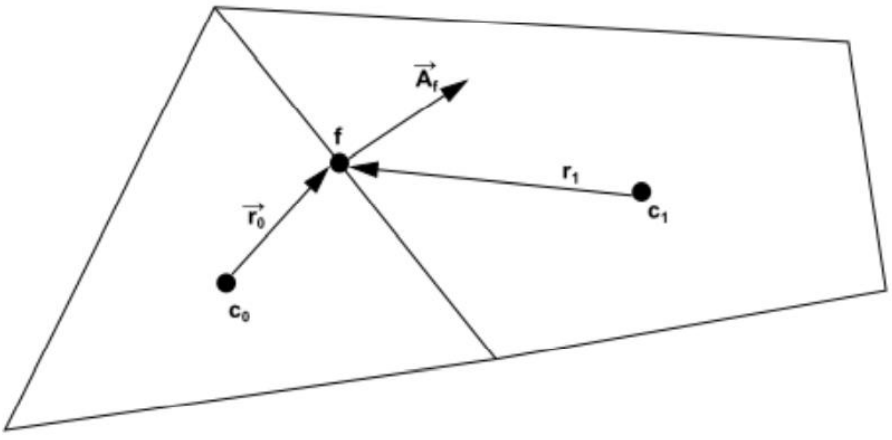


Figure 18: Two-dimensional triangular cell.

The discretized scalar transport equation (2.19) contains the unknown scalar variable φ at the cell center as well as the unknown values in surrounding neighbor cells. This equation will, in general, be nonlinear with respect to these variables. A linearized form of equation (2.19) can be written as:

$$\alpha_p \varphi = \sum_{nb} \alpha_{nb} \varphi_{nb} + b \tag{2.20}$$

where the subscript nb refers to neighbor cells, and α_p and α_{nb} are the linearized coefficients for φ and φ_{nb}.

The number of neighbors for each cell depends on the mesh topology but will typically equal the number of faces enclosing the cell.

Ansys Fluent stores discrete values of the scalar at the cell centers. However, face values φ_f are required for the convection terms in equation (2.19) and must be interpolated from the cell center values. This is accomplished using an upwind scheme. Upwind means that the face value φ_f is derived from quantities in the cell upstream, or "upwind," relative to the direction of the normal velocity v_n in equation (2.19).

The diffusion terms in equation (2.19) are central-differenced and are always second-order accurate.

a. First-Order Upwind Scheme

When first-order accuracy is desired, quantities at cell faces are determined by assuming that the cell-center values of any field variable represent a cell-average value and hold throughout the entire cell, meaning that the face quantities are identical to the cell quantities. Thus, when first-order upwind is selected, the face value φ_f is set equal to the cell-center value φ of in the upstream cell.

b. Second-Order Upwind Scheme

When second-order accuracy is desired, quantities at cell faces are computed using a multidimensional linear reconstruction approach. In this approach, higher-order accuracy is achieved at cell faces through a Taylor series expansion of the cell-centered solution about the cell centroid. Thus, when using second-order upwind discretization method, the face value is computed using the following expression:

$$\varphi_{f,SOU} = \varphi + \nabla\varphi \cdot \vec{r} \tag{2.20}$$

Where φ and ∇φ are the cell-centered value and its gradient in the upstream cell, and $\vec{r}$ is the displacement vector from the upstream cell centroid to the face centroid. This formulation requires the determination of the gradient ∇φ in each cell. Finally, the gradient is limited so that no new maxima or minima are introduced.

c. Central-Differencing Scheme

A second order-accurate central-differencing discretization scheme is available for the momentum equations when using a Scale-Resolving Simulation (SRS) turbulence model (SRS models are analyzed in the next sections), such as LES. The central-differencing scheme calculates the face value for a variable φ_f as follows:

$$\varphi_{f,CD} = \frac{1}{2}(\varphi_0 + \varphi_1) + \frac{1}{2}(\nabla\varphi_0 \cdot \overrightarrow{r_0} + \nabla\varphi_1 \cdot \overrightarrow{r_1}) \tag{2.21}$$

Where the indices 0 and 1 refer to the cells that share face f, $\nabla\varphi_{r,0}$ and $\nabla\varphi_{r,1}$ are the reconstructed gradients at cells 0 and 1, respectively, and $\vec{r}$ is the vector directed from the cell centroid toward the face centroid.

It is well known that central-differencing schemes can produce unbounded solutions and non-physical wiggles, which can lead to stability problems for the numerical procedure.

d. Bounded Central Differencing Scheme

The central differencing scheme is an ideal choice for Scale-Resolving Simulation (SRS) turbulence models because of its low numerical diffusion. However, it often leads to unphysical oscillations in the solution fields. The bounded central differencing (BCD) scheme consists of a pure central differencing, a blended scheme of a central differencing and an upwind scheme, and the first-order upwind scheme. A tunable version of the BCD scheme is implemented in the pressure-based solver of Ansys Fluent. The boundedness strength of BCD can be controlled using a parameter, which allows to relax the strict CBC and to keep using the central differencing with the locally non-monotonous solution field, when the non-monotonicity is relatively low.

This parameter can be specified as a constant or an expression and can be changed within the range from 0 to 1. The maximum value of 1 corresponds to the strict CBC, which makes the BCD scheme more stable although more dissipative. The minimum value of 0 deactivates the BCD boundedness completely and turns the scheme to the pure central differencing.

For the DrivAer case the value was set to 1.

e. High Order Term Relaxation

Higher order schemes can be written as a first-order scheme with additional terms for the higher order scheme. The higher-order relaxation can be applied to these additional terms. The under-relaxation of high order terms follows the standard formulation for any generic property φ.

$$\varphi_{NEW} = \varphi_{OLD} + f(\varphi_{Intermediate} - \varphi_{OLD}) \tag{2.22}$$

Where f is the under-relaxation factor. Note that the default value of f for steady-state cases is 0.25 and for transient cases is 0.75. The same factor is applied to all equations solved.

f. Pseudo Time Method Under-Relaxation

The pseudo time method was used, which is an advanced form of implicit under-relaxation that adjusts the relaxation factor dynamically during the simulation according to the flow field behavior. After introducing the pseudo time method into the generic transport equation, the integral form of governing equation for steady-state calculations is of the following form:

$$\int_V \frac{\partial \rho\varphi}{\partial \tau} dV + \oint_S \rho\varphi\vec{v} \cdot \overrightarrow{dA} = \oint_S \Gamma_\varphi \nabla\varphi \cdot d\vec{A} + \int_V S_\varphi dV \tag{2.23}$$

Where τ denotes the pseudo time. After discretizing the governing equation using the finite volume method, the following algebraic form of the steady-state equation is obtained:

$$\rho_p \Delta V \frac{\varphi_p - \varphi_p^{old}}{\Delta\tau} + \alpha_p \varphi_p = \sum_{nb} a_{nb} \varphi_{nb} + b \tag{2.24}$$

Where φ_p^{old} denotes the value of at the previous iteration, and $\Delta\tau$ is the pseudo time step size that can be estimated by characteristic length and velocity scales.

2.2.6.6 *Hybrid Initialization*

Initialization methods are used to produce an initial velocity and pressure field in the entire computational domain, as required for the solution of partial differentials equations, to provide a better starting point for the main calculation and reduce the number of iterations needed for convergence. Hybrid Initialization is a combination of various methods. It solves the Laplace equation to produce a velocity field that conforms to complex domain geometries, and a pressure field which smoothly connects high and low pressure values in the computational domain. It is preferred to other methods, like standard initialization which used constant values in the entire field domain. The systems of equations solved for hybrid initialization are mentioned in the Appendix.

2.2.7 Scale Resolving Simulations

2.2.7.1 *Large Eddy Simulation (LES) Model*

Turbulent flows are characterized by eddies with a wide range of length and time scales. The largest eddies are typically comparable in size to the characteristic length of the mean flow, and the smallest scales are responsible for the dissipation of turbulence kinetic energy. It is possible, in theory, to directly resolve the whole spectrum of turbulent scales using an approach known as direct numerical simulation (DNS). No modeling is required in DNS. However, DNS is not feasible for practical engineering problems involving high Reynolds number flows. The cost required for DNS to resolve the entire range of scales is proportional to Re_t^3, where Re_t is the turbulent Reynolds number. Clearly, for high Reynolds numbers, the cost becomes prohibitive.

Large-Eddy-Simulation (LES) is based on the concept of resolving only the large scales of turbulence and to model the small scales. The classical motivation for LES is that the large scales are problem-dependent and difficult to model, whereas the smaller scales become more and more universal and isotropic and can be modeled more easily.

Resolving only the large eddies allows for a much coarser mesh and larger time-step sizes in LES than in DNS. However, LES still requires substantially finer meshes than those typically used for RANS calculations. In addition, LES has to be run for a sufficiently long flowtime to obtain stable statistics of the flow being modeled. As a result, the computational cost involved with LES is normally orders of magnitudes higher than that for steady RANS calculations in terms of memory (RAM) and CPU time.

2.2.7.2 *Detached Eddy Simulation (DES)*

Detached Eddy Simulation (DES) eliminates the main limitation of LES models by proposing a hybrid formulation that switches between RANS and LES based on the mesh resolution provided. By this formulation, the wall boundary layers are entirely covered by the RANS model and the free shear flows away from walls are typically computed in LES mode.

Within DES models, the switch between RANS and LES is based on a criterion like:

$$C_{DES}\Delta_{max} > L_t \rightarrow RANS \tag{2.25}$$

$$C_{DES}\Delta_{max} < L_t \rightarrow LES \tag{2.26}$$

Where $\Delta_{max} = \max(\Delta_x, \Delta_y, \Delta_z)$ is the maximum edge length of the local computational cell, L_t is the turbulence length scale and C_{DES} is a coefficient.

As the mesh is refined below the limit $\Delta_{max} < L_t$ the DES-limiter is activated and switches the model from RANS to LES mode. For wall boundary layers this translates into the requirement that the RANS formulation is preserved as long as the following conditions holds: $\Delta_{max} > \delta$, where δ is the boundary layer thickness. The intention of the model is to run in RANS mode for attached flow regions, and to switch to LES mode in detached regions away from walls.

Further information on the SRS models available in Ansys Fluent are available in reference [12].

2.2.7.3 *Stress-Blended Eddy Simulation (SBES)*

The SBES model is a member of the DDES model family [12]. It is using the shielding function f_s to explicitly switch between different turbulence model formulations in RANS and LES mode. In general terms that means for the turbulence stress tensor:

$$\tau_{ij}^{SBES} = f_s \cdot \tau_{ij}^{RANS} + (1 - f_s) \cdot \tau_{ij}^{LES} \tag{2.27}$$

Where τ_{ij}^{RANS} is the RANS part and τ_{ij}^{LES} the LES part of the modelled stress tensor. In case both model portions are based on eddy-viscosity concepts, the formulation simplifies to:

$$v_t^{SBES} = f_s \cdot v_t^{RANS} + (1 - f_s) \cdot v_t^{LES} \tag{2.28}$$

Such a formulation would not be feasible without strong shielding. When using the conventional shielding functions from the DDES model, the corresponding model would not be able to maintain a zero pressure gradient RANS boundary layer on any mesh. The SBES model formulation is currently recommended over the other global hybrid RANS-LES methods, since it offers a quicker transition between RANS and LES zones and is less mesh dependent.

For this thesis the SBES model was selected for the SRS simulations.

3. Numerical setup

In this chapter the overall simulation setup will be described, starting from the construction of the computational mesh, and continuing to the solver numerics used for both the steady-state and the transient simulations.

3.1 Mesh

In this section, a description of the "Poly-Hexcore" meshes that were used for the simulations of the DrivAer model, is given, along with the analysis of the mesh dependency study that was conducted for this thesis.

3.1.1 Input geometry

The computational domain used for the DrivAer test case is shown in Figure 19:

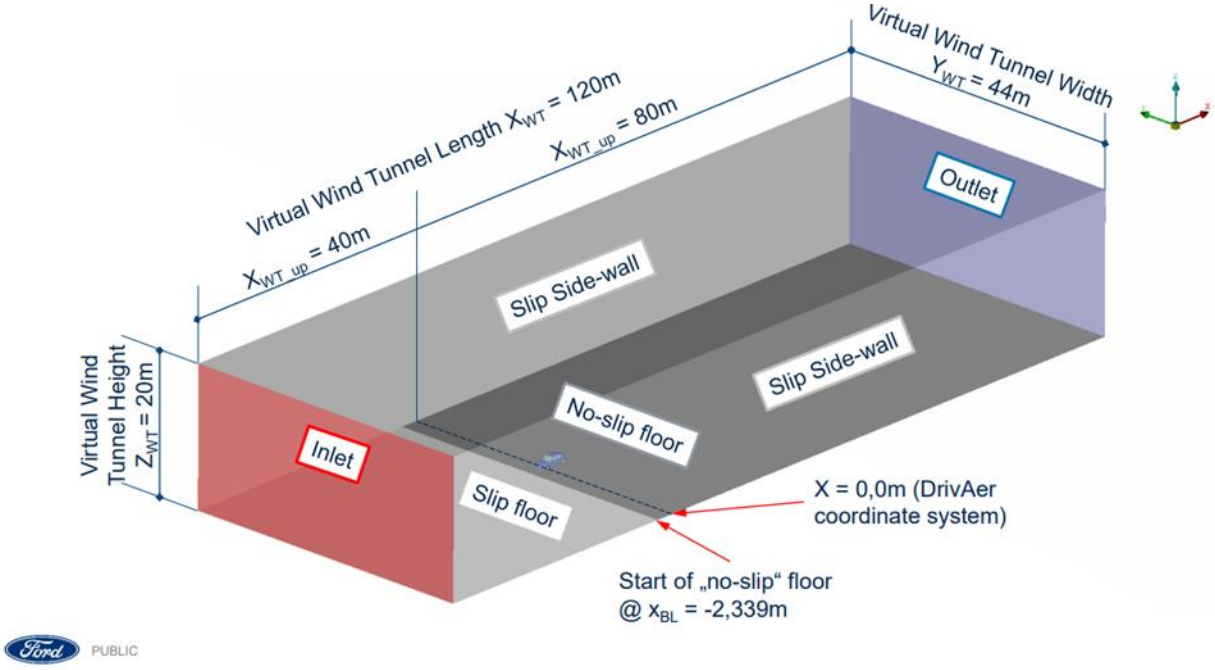


Figure 19: Computational domain for the DrivAer case [4].

In order to achieve the same boundary layer characteristics on the ground floor, slip condition was applied upstream of the car geometry and up to the point where the theoretical starting point of the boundary layer is. The rest of the ground was set as a separate part and no-slip condition was applied.

An initial volume mesh consisted of tetrahedral and hexahedral cells generated in ANSA that was provided by the 2nd AutoCFD Prediction Workshop was used [6]. The mesh file was imported in Fluent; the volume mesh was deleted, and the resulting surface mesh was triangulated. After remeshing the triangulated surface mesh using the desired local refinements, which are mentioned in the following sections, a new "Poly-Hexcore" volume mesh was generated in Fluent Meshing.

3.1.2 Mosaic meshing Technology

Mosaic meshing technology [16] is a patent pending Ansys Fluent meshing technology for computational fluid dynamic simulations, that accelerates the meshing process with a reduced face count, higher quality cells and efficient parallel scalability. Mosaic meshing technology enables polyhedral connections between disparate mesh types. An example of this type of mesh is shown in Figure 20.

For this study, a "Poly-Hexcore" mesh was used, which is a mix of high-quality hexahedral elements in the bulk region, polyhedral prismatic layers close to the wall to resolve the boundary layer region and fast polyhedral transitions between the prisms and the hexahedral elements. This type of mesh provides high quality elements and low cell count compared to traditional hexahedral-tetrahedral meshes.

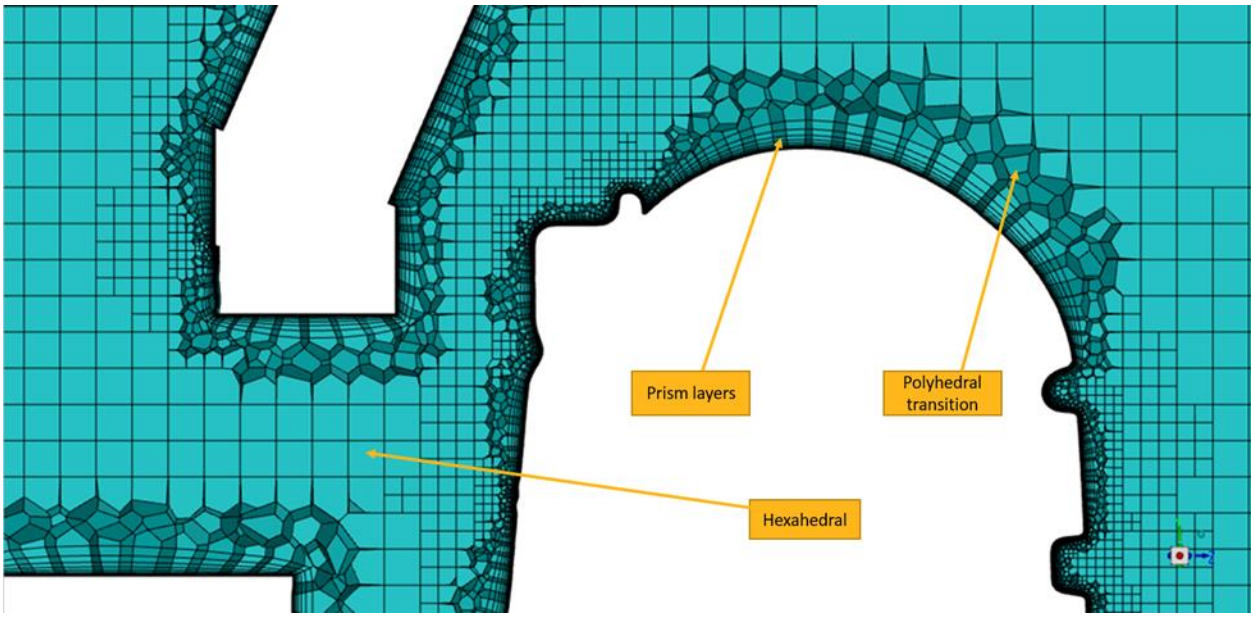


Figure 20: Mosaic mashing example.

3.1.3 Quality Metrics

The following metrics were used to judge the quality of the "Poly-Hexcore" meshes.

3.1.3.1 *Surface Skewness*

Skewness determines how close to ideal (that is, equilateral or equiangular) a face or cell is.

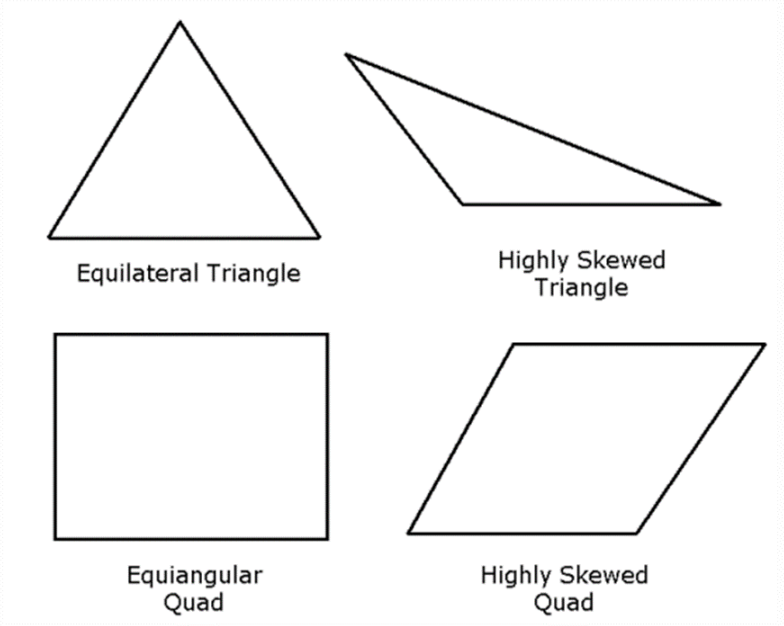


Figure 21: Skewed geometrical shapes [9].

Skewness is defined as:

$$Skewness = \frac{Optimal\ Cell\ Size - Cell\ Size}{Optimal\ Cell\ Size} \tag{3.1}$$

Where, the optimal cell size is the size of an equilateral cell with the same circumradius.

Table 2: Skewness value range

Skewness	***Cell quality***
1	Degenerate
0	Equilateral

According to the definition of skewness, a value of 0 indicates an equilateral cell (best quality) and a value of 1 indicates a completely degenerate cell. Degenerate cells (slivers) are characterized by nodes that are nearly coplanar. Cells with a skewness value above 1 are invalid.

The mesh of highly complicated geometries for external aerodynamic simulations is usually characterized by a maximum skewness greater than 0.6. There are skewness correction methods available in Fluent solver to counteract this issue.

3.1.3.2 *Orthogonal quality*

The orthogonal quality for a cell is computed using the face normal vector $\overrightarrow{A_i}$, for each face, the vector from the cell centroid to the centroid of each of the adjacent cells $\overrightarrow{c_i}$, and the vector from the cell centroid to the centroid of each face $\overrightarrow{f_i}$, as shown in Figure 22. For each face, the cosines of the angle between $\overrightarrow{A_i}$

and $\overrightarrow{c_i}$, and between $\overrightarrow{A_i}$ and $\overrightarrow{f_i}$ are calculated. The smallest calculated cosine value is the orthogonality of the cell. Then, inverse orthogonality is found by subtracting this cosine value from 1.

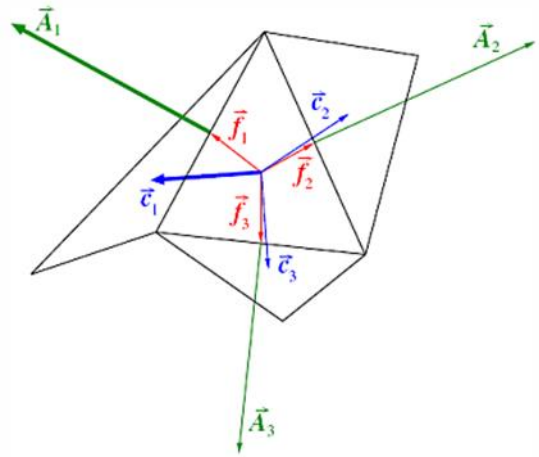


Figure 22: The Vectors Used to Compute Orthogonality [9].

Table 3: Orthogonal quality value range

Orthogonal Quality	***Cell quality***
0	Poor
1	Excellent

Cells with quality less than 0.01 may cause convergence problems in the solver.

3.1.3.3 *Fluent Aspect Ratio*

The Fluent aspect ratio is a measure of the stretching of a cell. It is computed as the ratio of the maximum value to the minimum value of the normal distances between the cell centroid and face centroids, and the distances between the cell centroid and nodes, as shown in Figure 23. For a unit, the maximum distance is 0.866, and the minimum distance is 0.5, so the aspect ratio is 1.732. This type of definition can be applied on any type of mesh, including polyhedral.

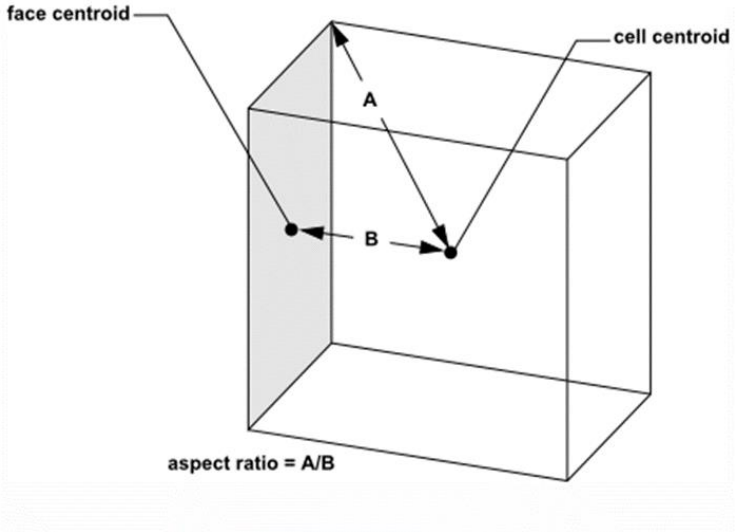


Figure 23: Calculating the Aspect Ratio for a Unit Cube [9].

3.1.4 Mesh characteristics

3.1.4.1 *Meshing Workflow*

The "Watertight Geometry Workflow" in Fluent Meshing was chosen for the mesh generation. After specifying the cell sizes for different parts of the DrivAer model, in order to sufficiently resolve the geometry, a triangulated surface mesh was generated. A larger cell size was chosen for the ground floor parts and an even larger for the domain side walls, since a fine mesh is not needed in the parts of the flow domain far from the car geometry.

For surfaces with higher curvature, the "Curvature" size function was used, which computes edge and face sizes using their size and normal angle parameters. It uses the normal angle parameter as the maximum allowable angle that one element edge may span. The value for the curvature normal angle parameter for the DrivAer model was chosen to be 12˚.

Some details from the surface mesh are depicted in Figure 24.

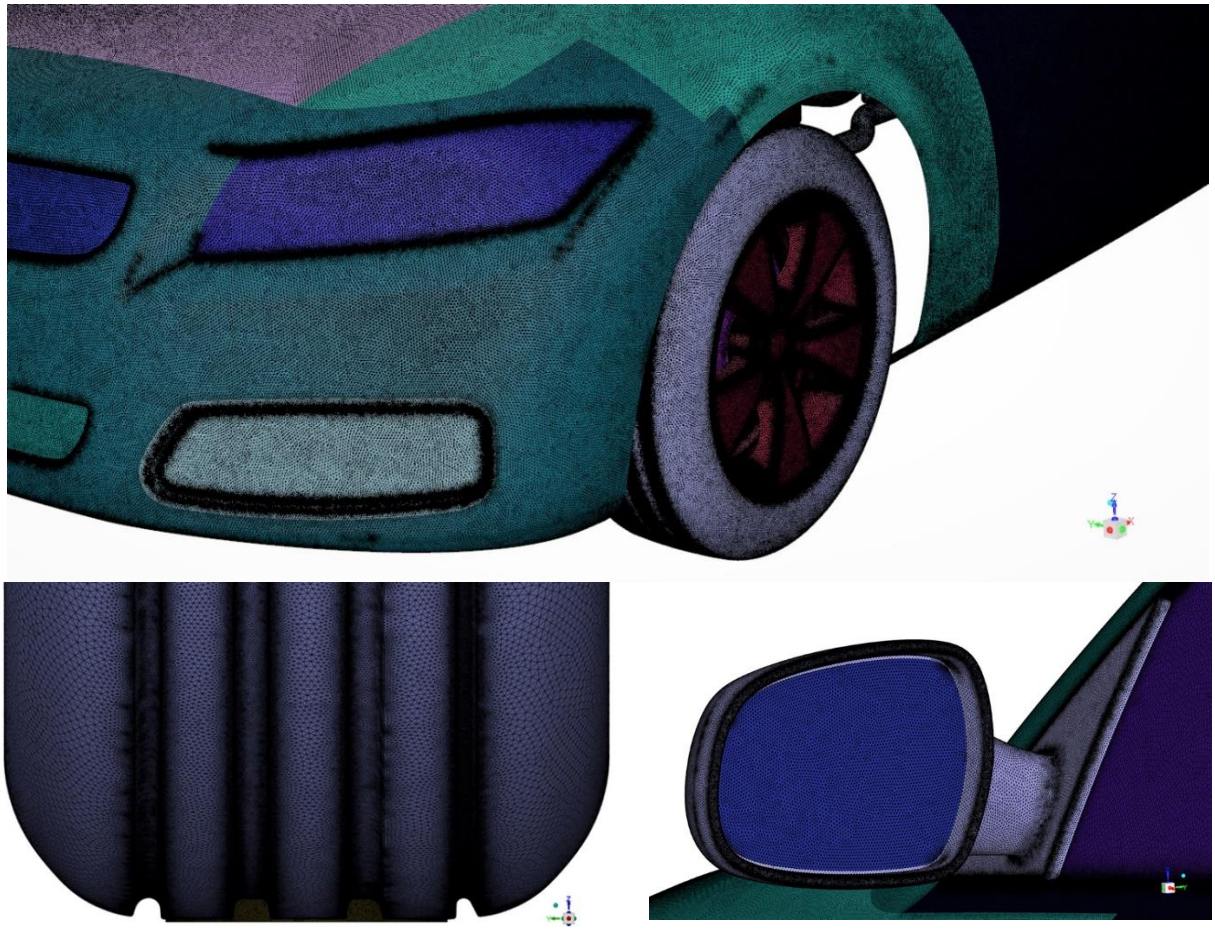


Figure 24: Surface mesh details on the DrivAer geometry.

3.1.4.2 *Bodies of Influence (BOIs)*

Extra refinement was needed in the regions of the flow domain around the car surface and in the wake region. The Offset BOI (Bodies of Influence) method was used to create a coarse wrap of the car surface, project it outwards and extended toward the flow direction. 4 BOIs were created, as shown in Figure 25, one "Boundary Layers Level" closer to the surface, and 3 "Wake Levels".

Table 4: Bodies of Influence (BOIs) cell lengths used for the mesh of the DrivAer model

Offset BOI	***Max cell length (mm)***
Boundary Layer Level	8
Wake Level 1	16
Wake Level 2	32
Wake Level 3	64

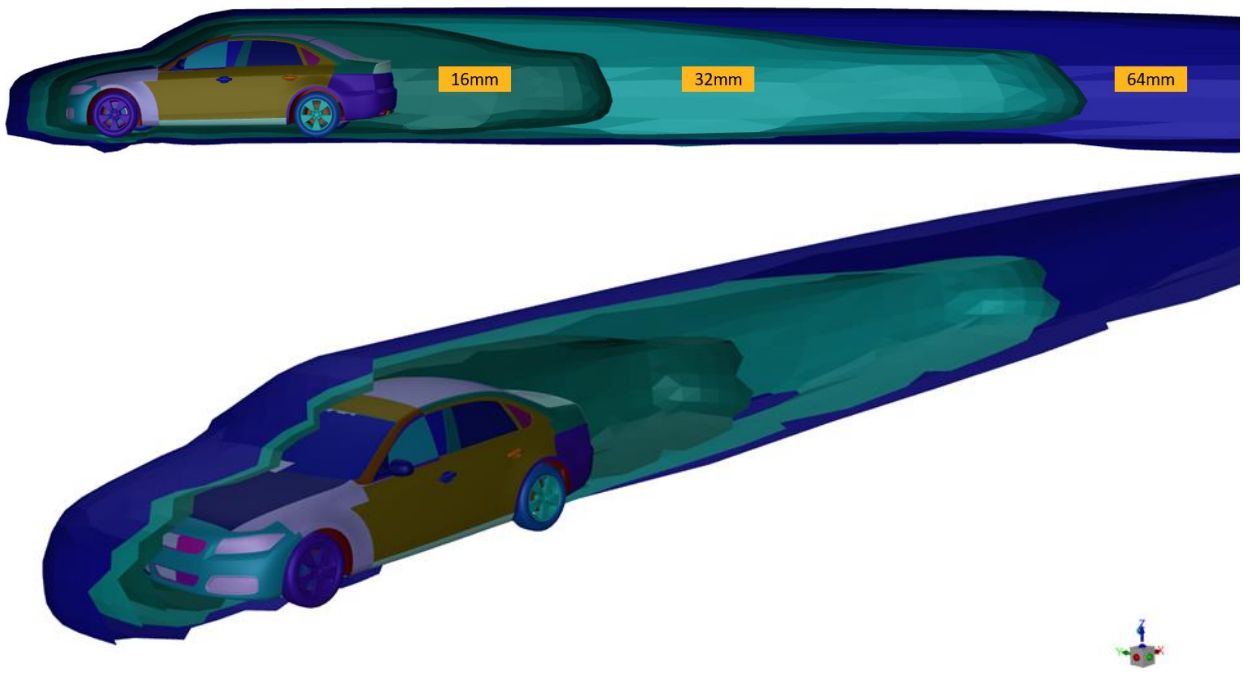


Figure 25: Bodies of Influence (BOIs) around the DrivAer geometry.

3.1.4.3 *Layers*

The region of the flow domain close to the surface of the car, in the boundary layer region, was resolved using polyhedral prismatic layers.

In Fluent Meshing the "Last Ratio" method was used, which allows control over the aspect ratio of the prism cells that are extruded from the base boundary zone. After specifying the height of the first prism layer, local base mesh size is used to find out the offset height for the last layer. This is shown in Figure 26. By specifying 40 as the "Last Percent" value, the last layer will be 0.4 times the local base mesh size. A local growth rate was used to calculate the other intermediate offset heights exponentially.

An example of the layer formation on the DrivAer geometry using the "Last Ratio" method is shown in Figure 27.

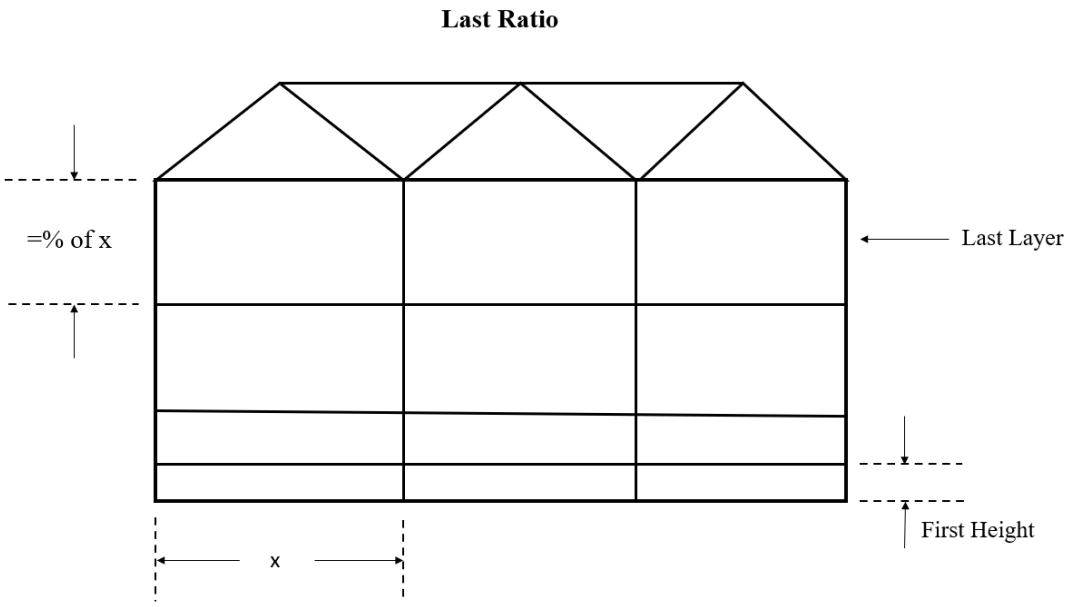


Figure 26: Prismatic layers last ratio.

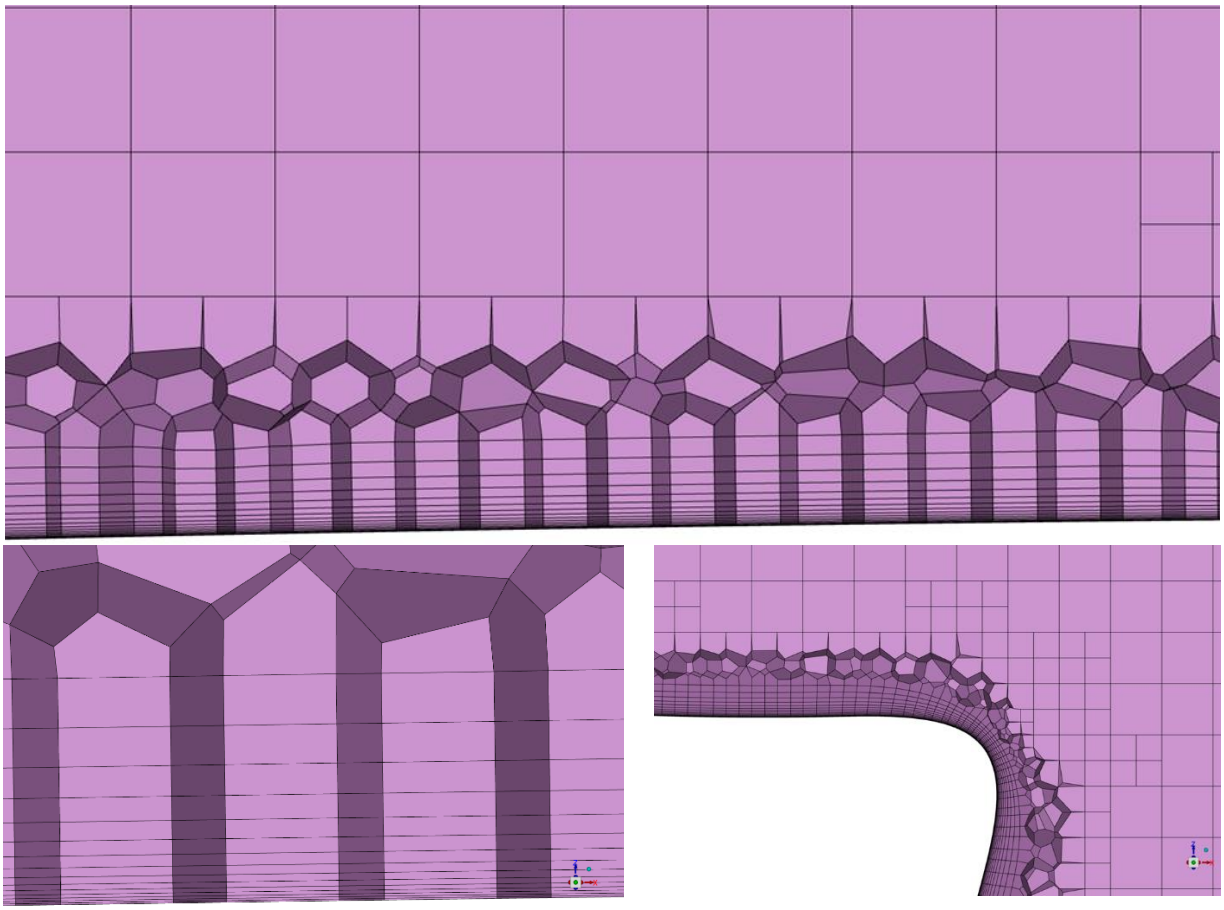


Figure 27: Prismatic layers on the DrivAer geometry.

3.1.5 Mesh Independence study

In order to ensure a mesh independent solution, meaning that the final result would not be dependent on the level of mesh refinement, a mesh independence study was carried out. Three meshes were generated, with the level of refinement varying from coarse to fine. A factor of 2 was used to increase the refinement (reduce the cell length) between each mesh.

3.1.5.1 *Wall Resolved mesh*

The aim was two create wall resolved meshes, meaning that the first prismatic layer would be located in the viscous sublayer region of the flow boundary layer.

The velocity profile in the boundary layer exhibits the following layer structure, which is also shown in Figure 28, identified from dimensional analysis:

- Outer layer:
  - Dependent upon mean flow
- Logarithmic layer:
  - log-law applies
  - The level of turbulent kinetic energy production and the level of dissipation are nearly equal
- Viscous sublayer:
  - Viscous forces rule, $U = f(\rho, \tau_w, \mu, y)$
  - The level of dissipation is greater than the level of turbulent kinetic energy production

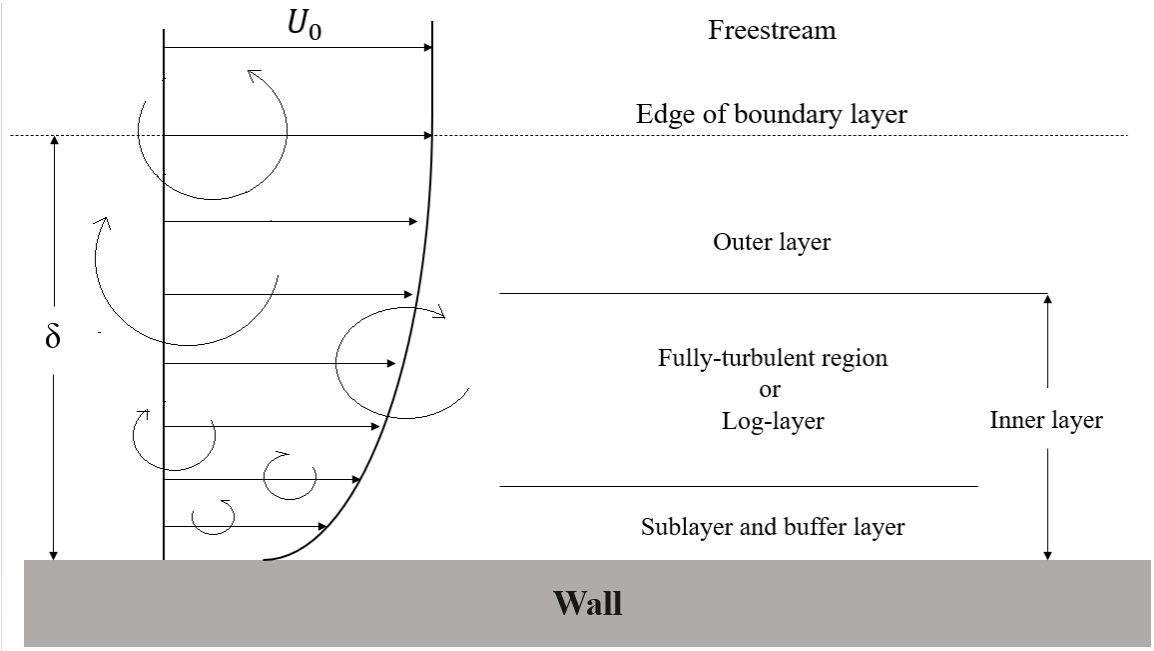


Figure 28: Boundary layer zones.

The non-dimensional quantities used to describe the boundary layer zones are:

- $U^+ = \frac{U}{u_\tau}$ (3.2)
- $y^+ = \frac{\Delta y u_\tau}{v}$ (3.3)

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \quad \rightarrow \quad U^+ = F(y^+) \tag{3.4}$$

Where U is the known velocity tangent to the wall at a distance Δy from the wall, u_τ is called friction velocity, U^+ is the non-dimensional near wall velocity, y^+ is the non-dimensional distance from the wall, τ_w is the wall shear stress. v is the kinematic viscosity, and y is the distance from the wall.

In the logarithmic layer, the logarithmic relation for the near wall velocity is:

$$U^+ = \frac{1}{\kappa}\ln(y^+) + C \tag{3.5}$$

Where κ, C are constants. It is assumed that in this region the non-dimensional velocity distribution is independent of the flow type.

In the viscous sublayer, where the molecular viscosity plays a dominant role in momentum and heat transfer, turbulent fluctuations are damped out and the wall shear stress is almost entirely viscous. The u-momentum equation reduces to:

$$\mu\frac{\partial U}{\partial y} = \tau_w, \quad constant \tag{3.6}$$

Which yields a linear velocity profile:

$$U = \frac{\tau_w}{\mu}y, \quad U^+ = y^+ \tag{3.7}$$

The near wall universal profiles are shown in *Figure 29: Near wall universal profiles [14]*.

For aerodynamic flows, it is common practice to have at least 10 cell layers inside the boundary layer, and for highly accurate simulations even up to 40 layers.

When the first layer is located in the log-layer, for y+>30, additional wall functions are needed to model the near wall behavior of the flow. When the first layer is located in the viscous sublayer, there is no need for wall functions. K-ω based models are considered y+-insensitive.

Further information on the current topic can be found in [15].

For the present thesis, all three meshes were generated having the first prism layer located inside the viscous sublayer. The resolution of the boundary layer changed from coarse to fine respectively as shown in **Error! Not a valid bookmark self-reference.** following table.

Table 5: Inflation layer characteristics for the mesh independence study on the DrivAer model

	Coarse	***Medium***	***Fine***
Target y+	7	3.5	1.5
First layer height (mm)	0.075	0.0375	0.01875
Number of prism layers	10	15	20

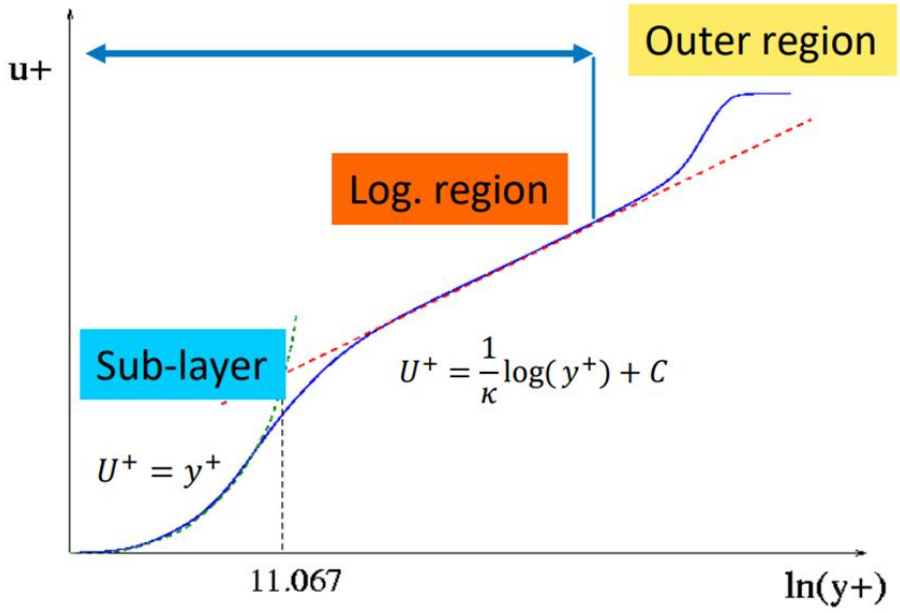


Figure 29: Near wall universal profiles [14].

3.1.5.2 *Cell size selection*

The details for min and max cell sizes, cell count and quality metrics for the three meshes are written in the table below.

Table 6: Mesh characteristics for the mesh independence study on the DrivAer model

	Coarse	***Medium***	***Fine***
Car surface cell size (min/ max) (mm)	4/ 8	2/ 4	1/ 2
Road/ Domain cell size (mm)	512/ 1024	256/ 512	128/ 256
Cell count (million)	55	178	450
Max Surface Skewness	0.827	0.827	0.81
Min Orthogonal quality	0.07	0.08	0.08
Max Aspect ratio	810	560	450

3.1.5.3 *Mesh details*

Details of the flow domain from the three meshes generated are shown in the follow pictures.

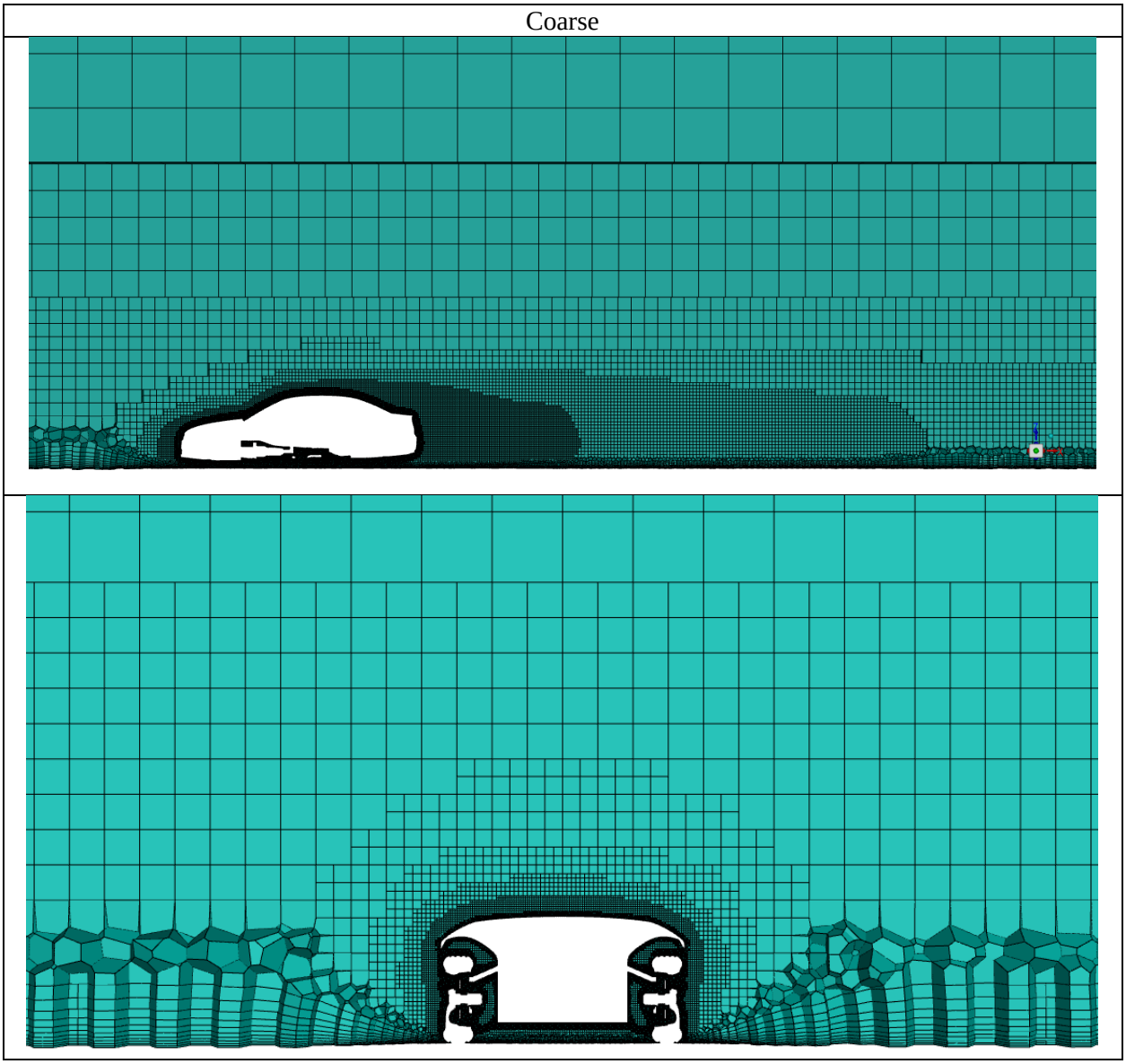


Figure 30: Volume mesh details from the coarse mesh of the DrivAer model.

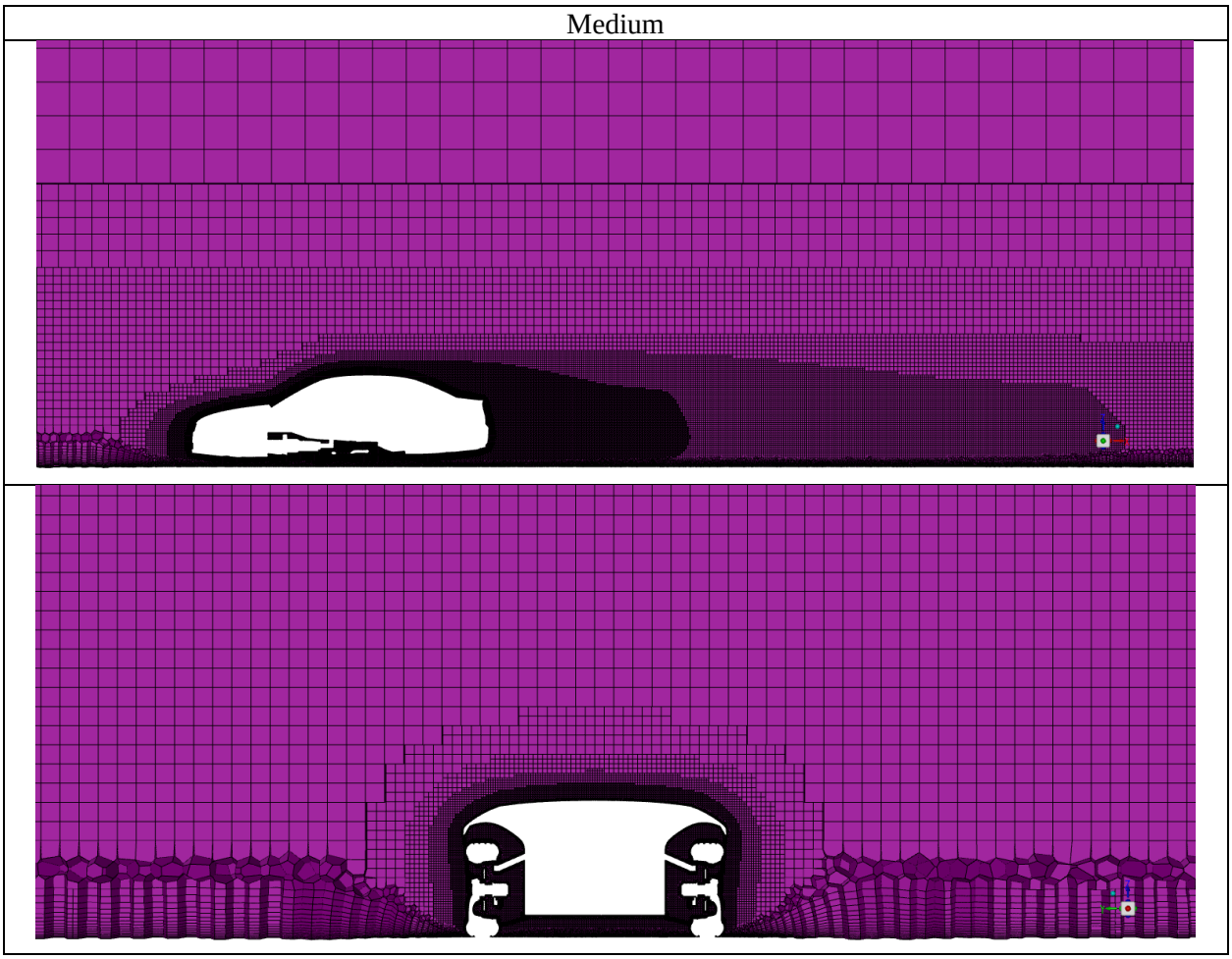
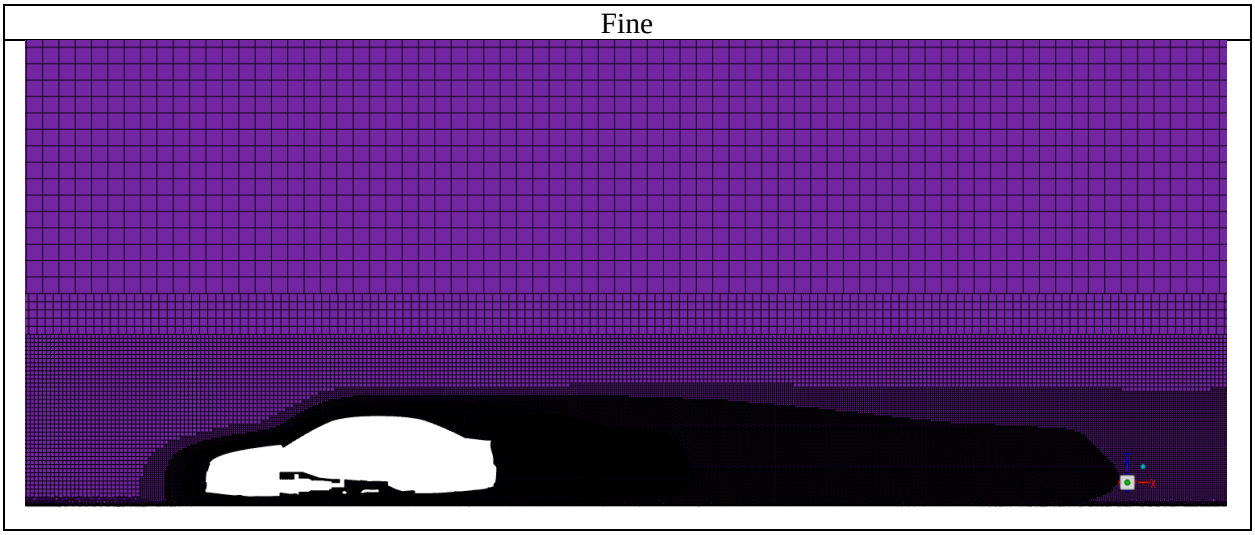


Figure 31: Volume mesh details from the medium mesh of the DrivAer model.



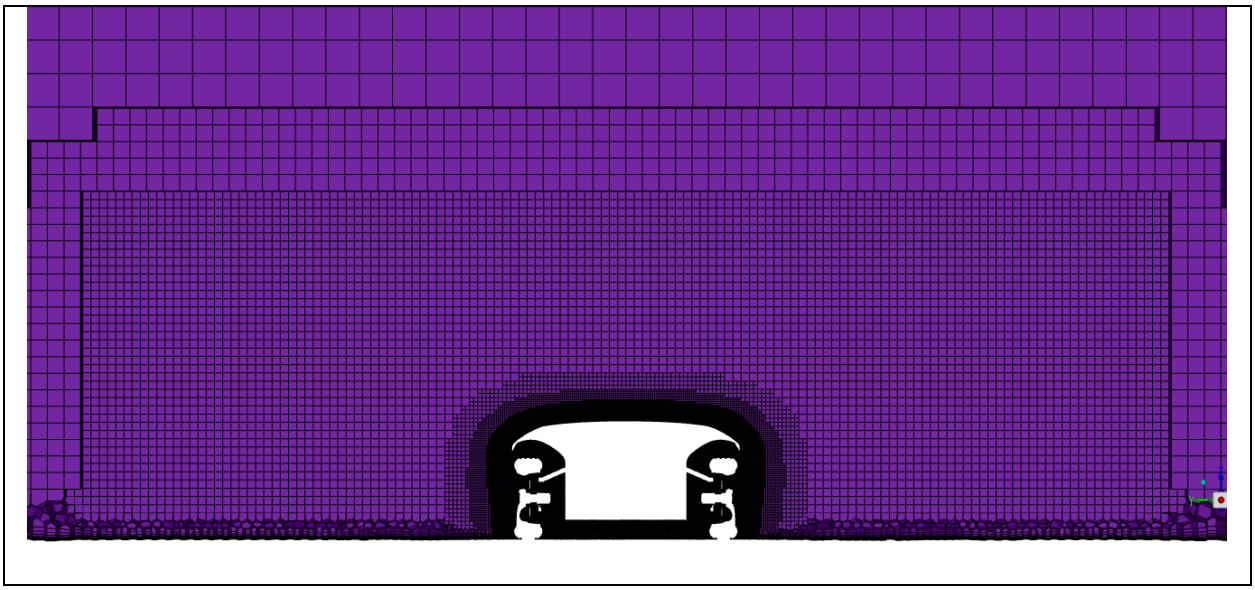


Figure 32: Volume mesh details from the fine mesh of the DrivAer model.

3.2 Solver set up

In this section the solver setup for the steady state RANS and the transient SBES simulations will be described.

3.2.1 Steady state simulation

A RANS steady state simulation was used to initialize the transient SRS simulation. Starting the transient simulation from a converged steady state result provides more accurate initial conditions.

The steady state simulation setup characteristics are written in the table below:

Table 7: Steady-state RANS simulation solver setup

Solver	Ansys Fluent
Version	22R2
Type	Pressure-Based
Viscous Model	k-omega GEKO
GEKO coefficients	C_{SEP}=1, C_{NW}, C_{MIX}, C_{JET} : default
Pressure – Velocity Coupling	Coupled
Spatial Discretization	
Pressure	Second Order
Momentum	Second Order Upwind
Turbulent Kinetic Energy	Second Order Upwind
Specific Dissipation Rate	Second Order Upwind
Pseudo Time Method	Global Time Step

In order to accurately predict near wall separation, a k-ω based model was used. The GEKO model was selected with Csep=1 which would help avoid overprediction of separation on the surface of the car. The rest of the GEKO coefficients were used as default.

The Hybrid Initialization Method was used with 10 iterations and both the "Use External-Aero Favorable Settings", and "Maintain Constant Velocity Magnitude" options enabled.

- ***Use External-Aero Favorable Settings*** is used in order to have the velocity potential patched with a linear value to help accelerate convergence of Scalar Equation–0 and to obtain a better guess of the velocity field for external-aero problems, such as flow over wings, airfoils, or automobiles.
- ***Maintain Constant Velocity Magnitude*** is selected in order to use the flow direction obtained from solving the velocity potential (Scalar Equation–0), while maintaining a constant velocity magnitude throughout the computational domain.

The coupled pressure-velocity coupling method was used to accelerate convergence in a small number of iterations, along with the pseudo-time method. For the conservative Phase 1, a smaller pseudo-timestep was used, along with lower under relaxation factors and the High Order Term Relaxation enabled, to ensure stability at the beginning of the simulation and avoid divergence of the solution. For the aggressive Phase 2, a higher pseudo-timestep was used, along with higher Under Relaxation Factors, to accelerate the convergence rate, and the HOTR disabled.

The different steps of the steady-state workflow are shown in the following table.

Table 8: Steady-state RANS simulation, pseudo-transient method steps

	Phase 1	***Phase 1.1 (transition)***	***Phase 2***
Timestep (s)	0.005	0.01	0.01
Iterations	10	10	380
HOTR	ON	ON	OFF
	UDF		
Pressure	0.25	0.25	0.4
Momentum	0.25	0.25	0.4
Density	1	1	1
Body Forces	1	1	1
Turbulent Kinetic Energy	0.5	0.6	0.8
Specific Dissipation Rate	0.5	0.6	0.8
Turbulent Viscosity	1	1	1

The above methodology was designed after trial and error, to achieve fast convergence and a robust overall solution.

3.2.2 Transient Simulation

The pressure-velocity coupling method was switched from Coupled to SIMPLEC, which requires less RAM memory and less time for a single iteration, which makes it more efficient for transient SRS simulations. It increases though the risk of divergence and floating points. Therefore, smaller URFs were used.

The SBES model was selected with the WALE Subgrid-Scale model [9]. All the model constants were used as default.

The transient simulation setup characteristics are written in the table below:

Table 9: Transient SBES simulation solver setup

Hybrid Model	SBES
Subgrid-Scale Model	WALE
Pressure-Velocity coupling	SIMPLEC
Spatial Discretization	Least Squares Cell Based
Gradient	Least Squares Cell Based
Pressure	Second Order
Momentum	Bounded Central Differencing
Turbulent Kinetic Energy	Second Order Upwind
Specific Dissipation Rate	Second Order Upwind
BCD Scheme Boundedness	1
Transient Formulation	Bounded Second Order Implicit

3.2.2.1 *Time step*

Even though implicit solvers are usually less sensitive to numerical instability and so larger values of CFL may be tolerated, still nonlinearities in the governing equations will often limit stability. That is why a CFL value closer to 1 is recommended for the biggest portion of the flow domain.

In order to have CFL close to 1 for the transient simulation, the timestep was derived based on the steady state results. The CFL Courant number is defined as:

$$Courant\ Number = \frac{U\Delta t}{\Delta x} \tag{3.8}$$

Using the Custom Field Function Calculator in Fluent, a pseudo-time scale was calculated based on the steady state solution, for each cell in the domain, which was defined as:

$$time\ scale = \frac{(cell\ volume)^{1/3}}{(cell\ velocity)} \tag{3.9}$$

The minimum time scale can then be calculated and the rounded value for the required time step can be obtained.

For the medium mesh, for example, the custom time step is 0.0001s for the biggest part of the domain. Smaller time steps than 0.0001s can be located close the wall in regions of flow separation. This is expected behavior and it is not taken under consideration when selecting the time step for the transient simulation.

Table 10: Timestep used for the transient simulations of the coarse, medium and fine meshes

	Coarse	Medium	Fine
Δt (s)	0.0002	0.0001	0.00005

In order for the flow over the car to be considered fully developed, the number of timesteps is selected to equal multiple flow times. Flow time is defined as the time it takes for the flow to get from the front part of the car to the rear part, in the given speed.

$$flow\ time = \frac{lenght\ of\ the\ car}{V_\infty} = \frac{4.6(m)}{38.89(\frac{m}{s})} = 0.118(s) \approx 0.12(s) \quad (3.10)$$

At first, a bigger time step is selected, along with lower URF and more sub-iterations. This ensures robust convergence in the initial steps. Then gradually the timestep gets reduced to the smallest timestep that will give CFL=1, while the URFs are increased and the number of sub iterations decreased, for faster convergence rate.

The different steps of the transient workflow are shown in the following table.

Table 11: Transient SBES simulation steps for the medium mesh

	Δt (s)	***Number of timesteps***	***Sub-iterations***	***URF***	***Flows over the car***	***HOTR***
Step 1	0.0025	48	8	0.4	1	ON
Step 2	0.0005	240	6	0.6	1	OFF
Step 3	0.0002	626	4	0.7	1	OFF
Step 4	0.0001	600	2	0.8	0.5	OFF
Step 5	0.0001	200	2	085	0.2	OFF
Step 6	0.0001	200	2	0.9	0.2	OFF
Step 7	0.0001	200	2	0.95	0.2	OFF
Sample 1	0.0001	6250	2	0.95	5	OFF
Sample 2	0.0001	18900	2	0.95	16	OFF

The above methodology was designed after a series of trial and error, to achieve robust behavior, and stable oscillations of force monitors.

4. Results and Discussions

The main values of interest that were used to compare the results of the three meshes for the mesh independence study were primarily the C_d value, which is in general the main value used for judging the aerodynamic efficiency of road vehicles, and secondarily the C_l, C_{lf}, C_{lr} values. Those values were monitored during the solution stage and were used as a way of determining convergence and stability. For transient simulations it is more common to use quantities like force coefficient values of mass flow values, for example, to monitor convergence instead of solution residuals.

4.1 Steady state results

The steady state simulation was used for initialization of the transient one, so the goal was to find the minimum number of iterations that were required for the averaged result to be stable. In the following Cd/Iterations graphs in Figure 33, it is shown that at least 200 iterations are required for the C_d value to drop to a certain level. After that, different numbers of iterations were tested, as shown in Table 12, to find the average C_d value. Assuming that the largest number of iterations, that is 1200 iterations, would result in the most accurate averaged value, the minimum number of iterations that would give the same averaged value was 400 iterations in total.

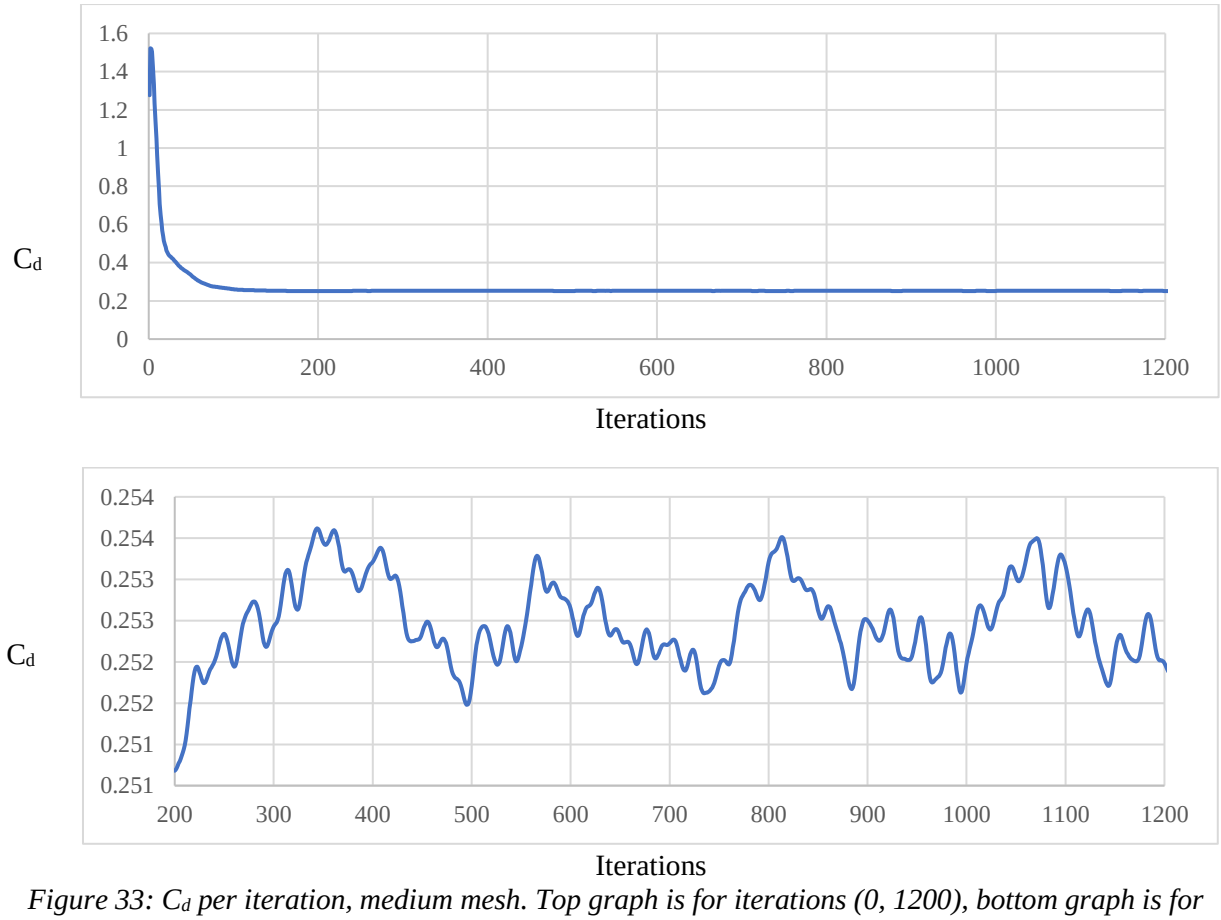


Figure 33: C_d per iteration, medium mesh. Top graph is for iterations (0, 1200), bottom graph is for iterations (200, 1200)

Table 12: Steady-state RANS simulation C_d value for different averaging periods

	C_d
Average: 200 – 1200	0.253
Average: 200 – 750	0.253
Average: 200 - 500	0.253
Average: 200 - 400	0.253
Average: 200 - 300	0.252

Using C_d as the main value not only to monitor convergence but also to judge mesh independence, while assuming that the finer mesh would give the most accurate result, it can be considered that a mesh independent solution was reached with the medium level of mesh refinement, as it can be seen in the table below.

Table 13: Steady-state RANS simulations' force coefficient average results for the mesh independence study, comparison with experimental values

Mesh	C_d	C_l	C_{lf}	C_{lr}
Coarse	0.257	0.082	0.006	0.157
Medium	0.253	0.084	0.006	0.161
Fine	0.253	0.09	0.014	0.165
Experimental	0.255	0.087	-0.0231	0.11

Yet, since these are results from RANS simulations, it is important to validate mesh independence using the transient SRS results.

4.2 Transient results

Following a similar process with the steady state simulation for selecting the minimum averaging period, or number of time steps for the transient simulation, the required flow time was 3s, with averaging starting from 0.5s, shown in Table 14. The instantaneous C_d value per flow time (s), for medium mesh is shown in Figure 34.

Table 14: Transient SBES simulation C_d values for different averaging periods, medium mesh

	Averaging period (s)	C_d
Average: 0.5 – 4.5 (s)	4	0.268
Average: 0.5 – 3 (s)	2.5	0.268
Average: 0.5 – 2 (s)	1.5	0.269
Average: 0.5 – 1.5 (s)	1	0.267

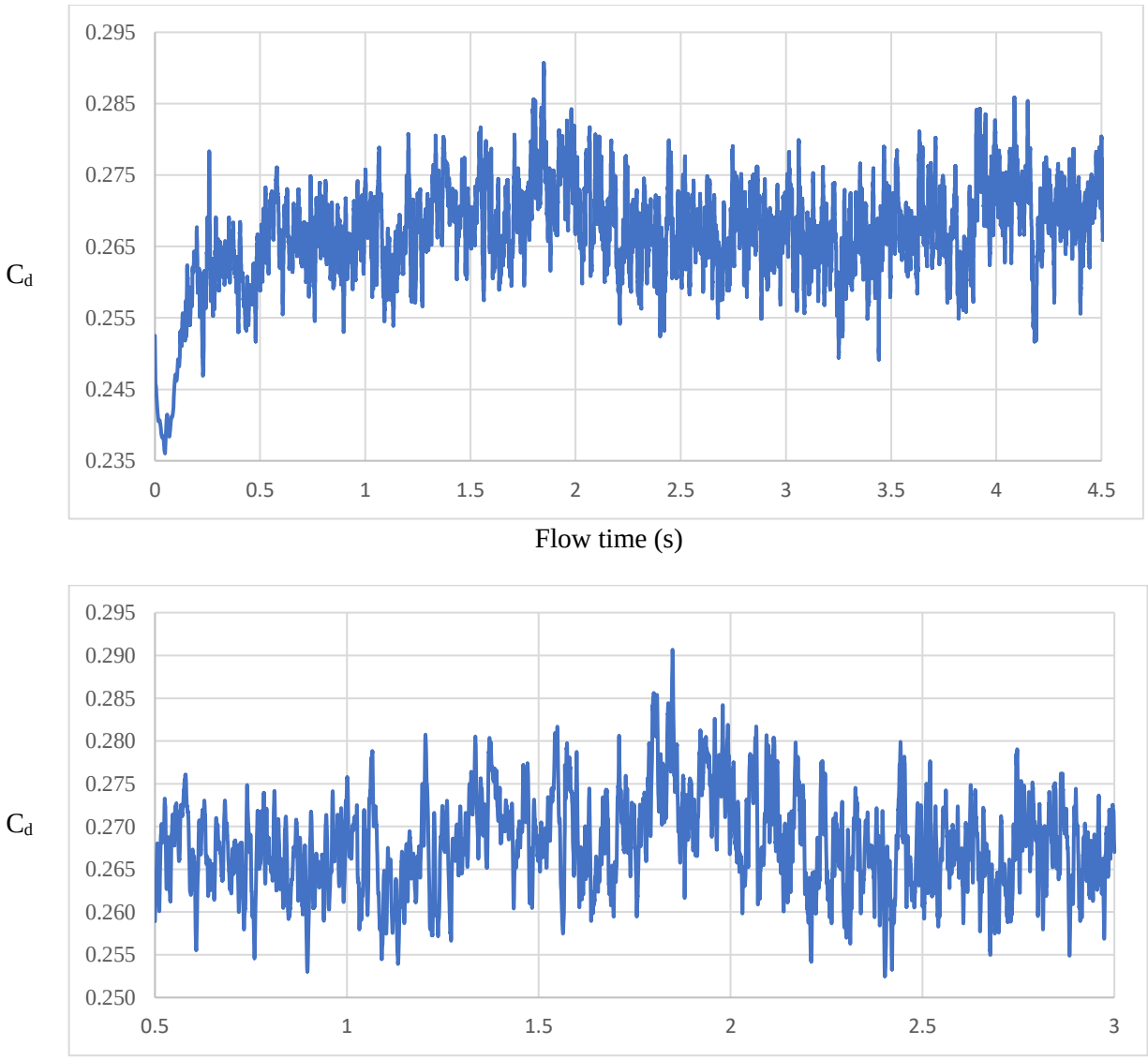


Figure 34: C_d per flow time (s), medium mesh. Top graph is for flow time (0, 4.5) (s), bottom graph is for flow time (0.5, 3) (s).

Looking at the C_d value in Table 15**: Transient SBES simulations' force coefficient results for the mesh independence study, comparison with experimental values** and Figure 35, once again the solution from the medium mesh can be considered mesh independent since it gave the same result as the finer mesh. For the rest of the coefficients, the averaged values from the medium mesh tend to be closer to the value of the fine mesh, in comparison with the coarser one, especially C_{lf} which has the same average value for both the medium and fine meshes.

Table 15: Transient SBES simulations' force coefficient results for the mesh independence study, comparison with experimental values

Mesh	***C_d***	***C_l***	***C_{lf}***	***C_{lr}***
Coarse	0.264	0.042	-0.039	0.124
Medium	0.268	0.03	-0.046	0.106

Fine	0.268	0.035	-0.046	0.115
Experimental	0.255	0.087	-0.0231	0.11

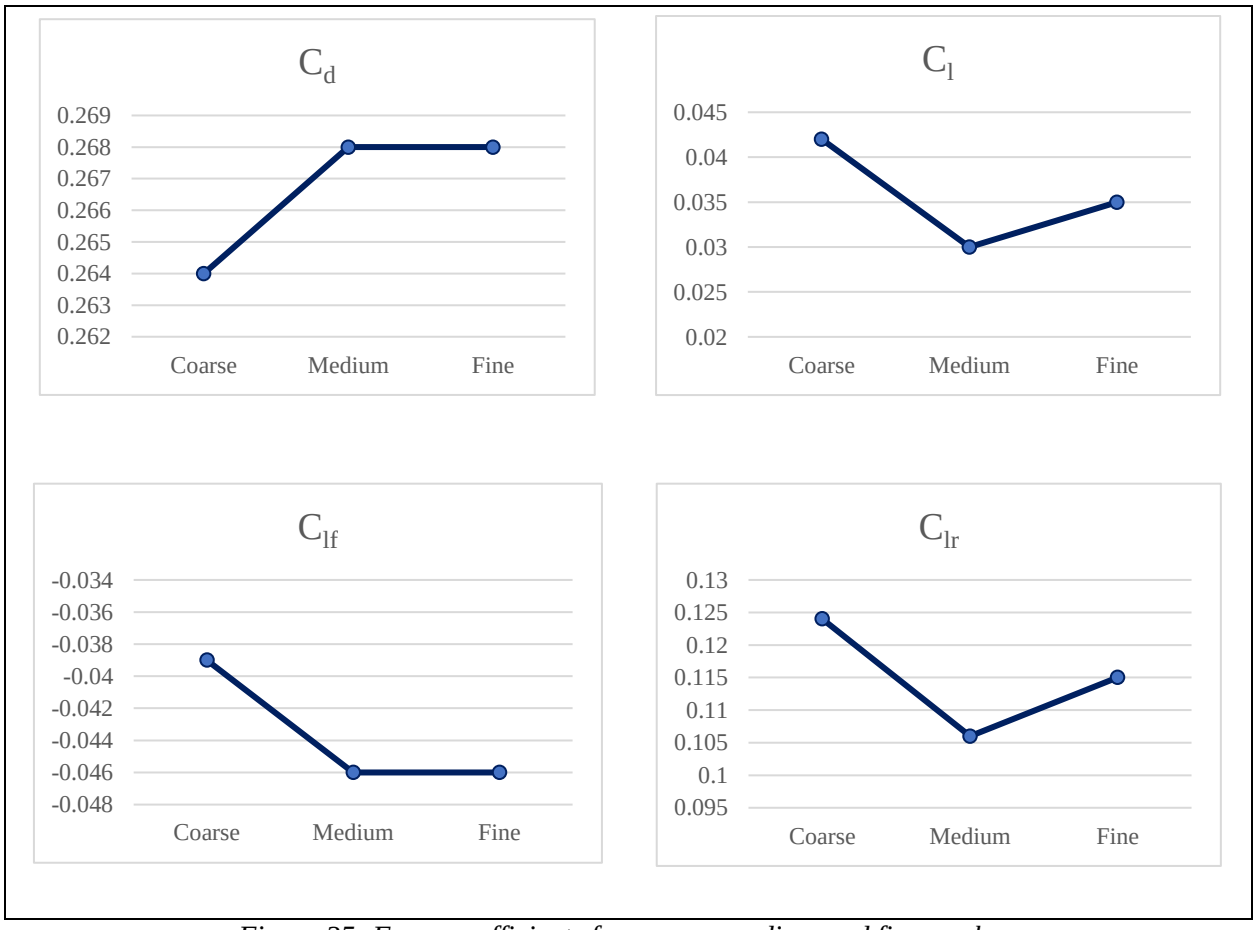


Figure 35: Force coefficients for coarse, medium and fine meshes.

4.3 Comparison of medium mesh with experimental results

4.3.1 Force Coefficients

Comparing the average force coefficient values from the transient medium mesh simulation with the experimental values, shown in Figure 36, the SBES simulation seems to have over-predicted C_d by 0.013, C_{lf} by 0.023, and under-predicted C_l by 0.052 and C_{lr} by 0.004, as shown in Figure 36.

When comparing simulation results with experimental data one must always take into consideration the reliability of the wind tunnel test data. As mentioned in the SAE paper "imperfections of the facility, the measurement system, the test model and test model setup might still lead to a situation in which wind tunnel tests conducted in the same wind tunnel, using the same measurement system and the same test vehicle lead to slightly different test results" [1]. For the static pressure measurements variation, it is mentioned that "the majority (~90%) of the observed pressure differences fall into a tolerance band of ΔC_p = +/−0.02 whereby ~70% of the measurements vary by no more than ΔC_p = +/−0.01" [1]. The highest pressure

variation occurs at the leading edge of the hood, an area for which test data from four different wind tunnels show ΔC_p=0.1, and the leading edge of the underbody, for which a pressure difference of up to ΔC_p=0.4 was observed between the same four tests. These two being highly sensitive areas due to strong pressure gradients could explain the $\Delta C_{lf} = -0.023$ between the medium mesh SBES simulation and the reported experimental value. More details on the variations between different test data can be found in the SAE paper. The same reasoning can be applied for the $\Delta C_l = 0.057$.

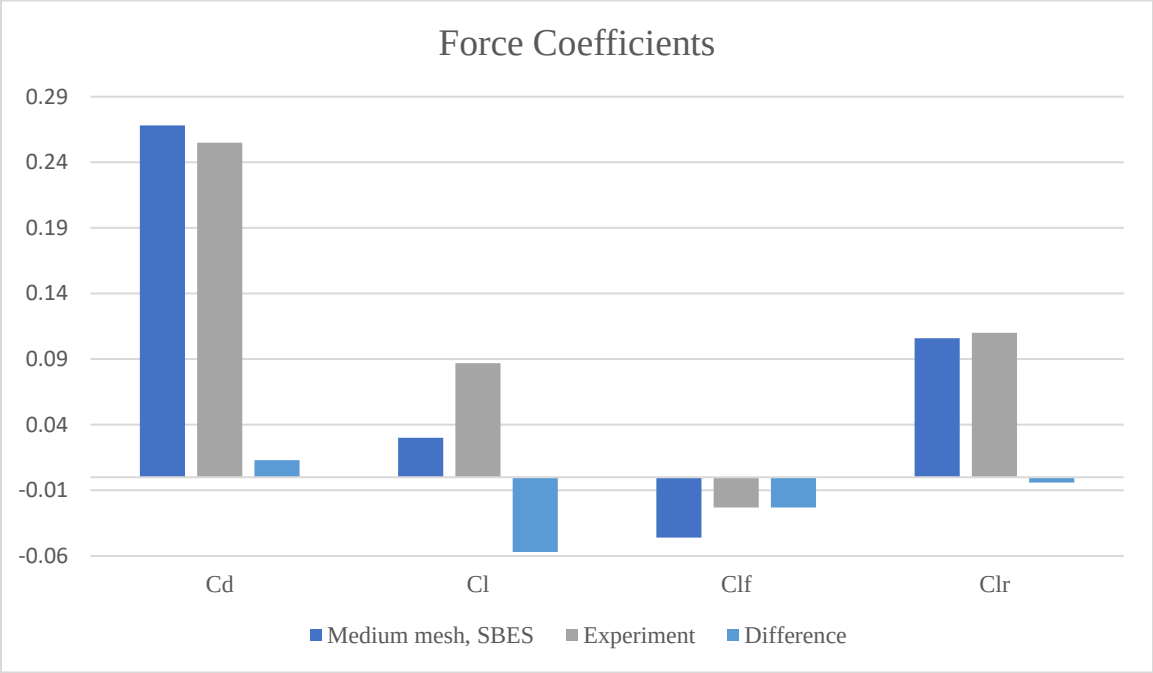
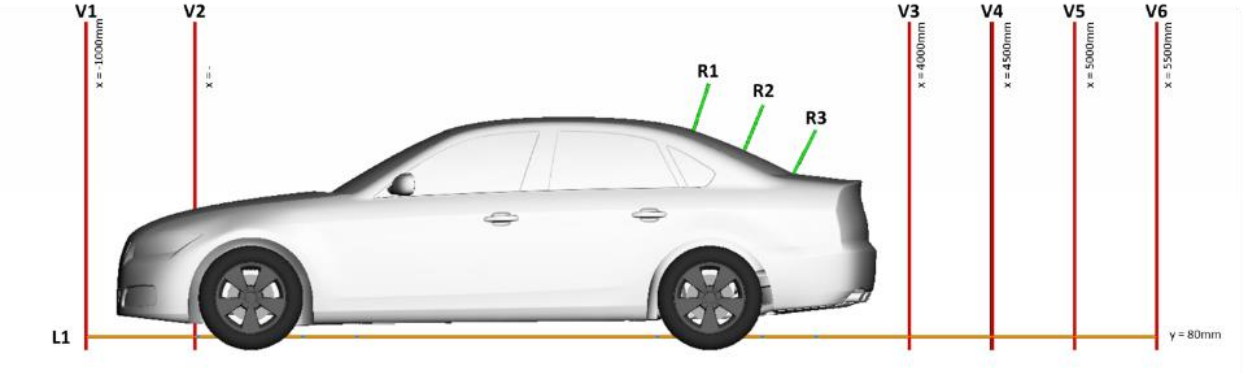


Figure 36: Force coefficients comparison between medium mesh SBES simulation and experiments.

4.3.2 Velocity plots

Velocity profile experimental data for normalized velocity are provided at the locations V1, V2, V4, V5, L1 and U1-6 shown in Figure 37 below.



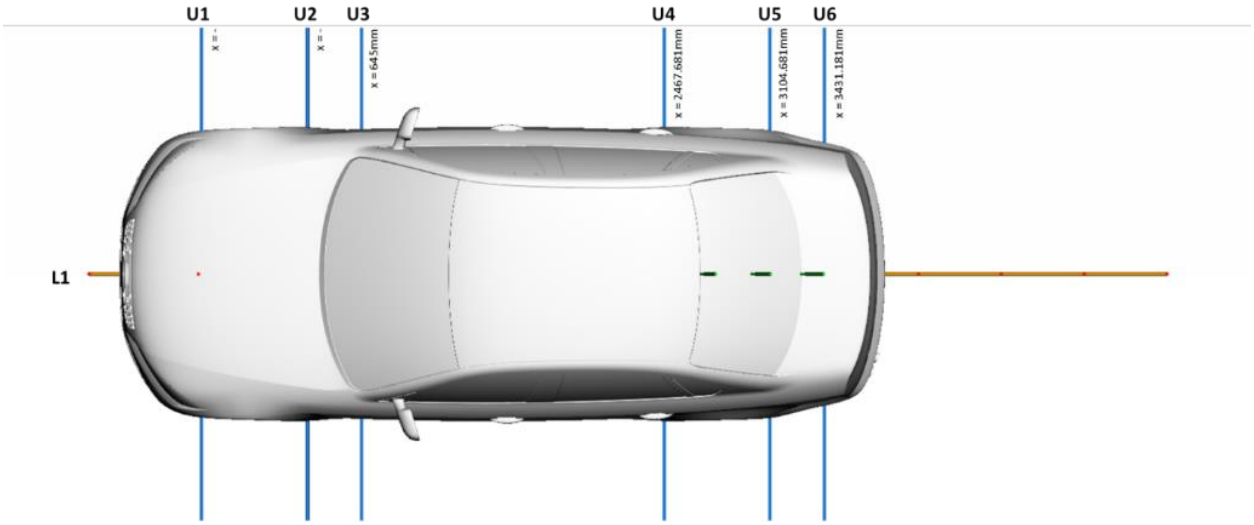


Figure 37: Velocity profile experiment measurement locations [7].

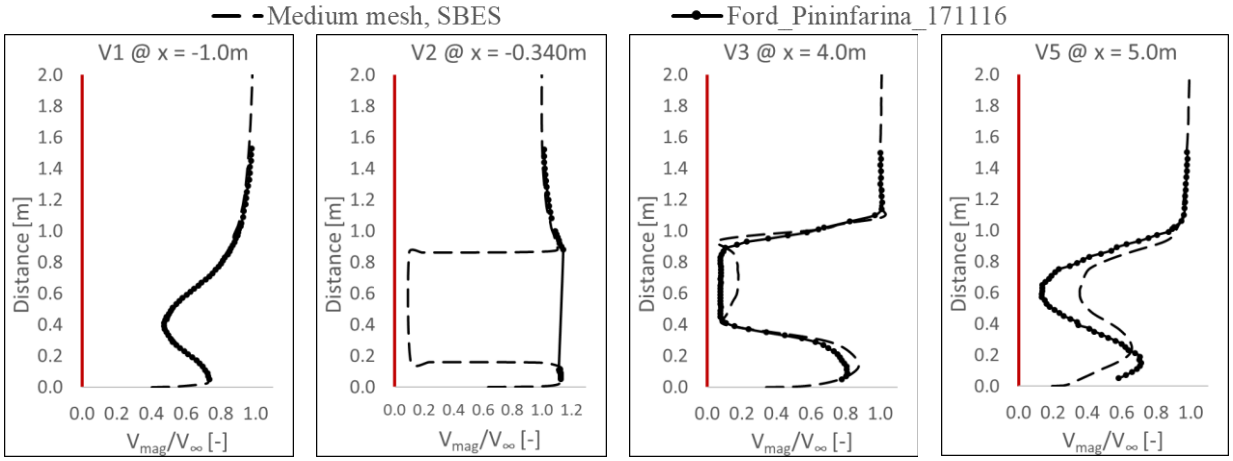


Figure 38: Comparison between medium mesh SBES simulation and experiment, for the V1, V2, V3 and V5 locations, Normalized velocity magnitude per distance (m).

V1 plot in Figure 38 is located in front of the car geometry and is used to validate the upstream flow and the boundary layer formation over the ground floor. The simulation results seem to match the experimental data. Same behavior is also observed in V2 plot.

V3 plot in Figure 38 is located in the wake area, close to the rear of the car. For $z \in (0.4, 0.9)m$ an overprediction of velocity value is observed. This could mean that lower pressure values dominate this flow region, which leads to the overprediction of the wake and finally the overprediction of drag force. Looking at plot V5, the same trend of velocity overprediction remains for $z \in (0.2, 1.0)m$. The difference in V5 between simulation and experiments is even greater which could explain the difference in the value of C_d.

The above speculations for the overprediction of the wake can be confirmed by the total pressure coefficient C_{pt}, the static pressure coefficient C_p and the normalized velocity contours in Figure 40 and Figure 42.

Indeed, the lower C_p and C_{pt} values are present in the wake region which means greater energy losses and thus higher drag force values.

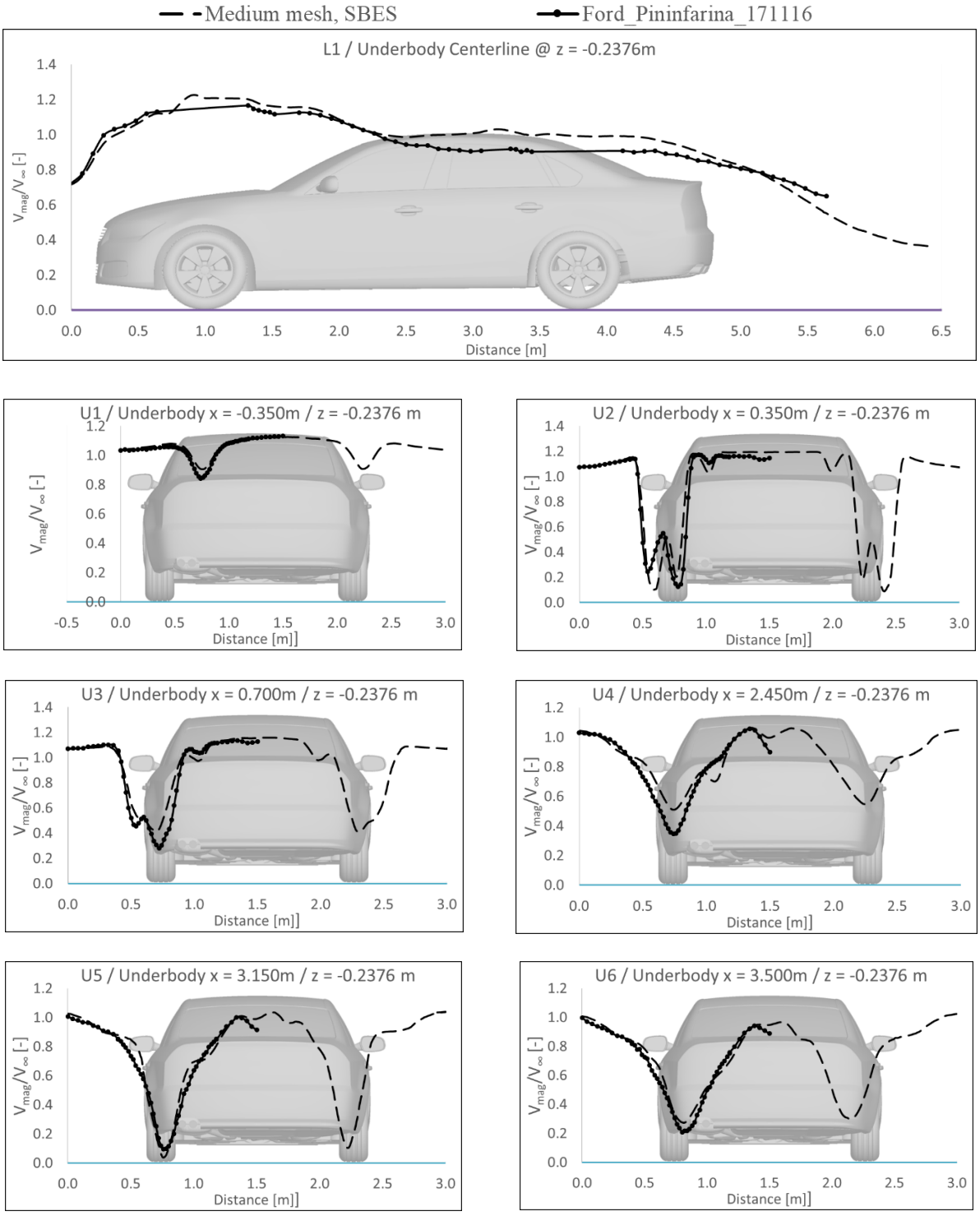


Figure 39: Comparison between medium mesh SBES simulation and experiment, for the L1, U1, U2, U3, U4, U5 and U6 locations, Normalized velocity magnitude per distance (m).

Plot U2 in Figure 39 is just behind the front wheels, where a vortex is generated as the high-speed flow meets the wake of the wheels. This effect can be seen in Figure 43. Simulation results show lower velocity values in the vortex generation region, indicating a stronger vortex. This can be confirmed by the lower pressure values in the simulation results compared to the experimental data in the C_p contour in Figure 43. These results could be the effect of the SBES model overpredicting separation.

Continuing to plot U3 in Figure 39, this time the simulation results are closer to the experiment, yet the slight overprediction of velocity could indicate the underprediction of the effect of the wheel in the downstream flow. This observed diffusion of the vortex could indicate numerical diffusion either due to SBES model or due to the mesh resolution in that particular area of the domain. The same explanation could also be given for plot U4 where again the overprediction of velocity indicate the numerical diffusion.

The depression of velocity due to diffusion observed in the regions upstream the rear wheel contributes to a less strong vortex generation in the rear outer corner of the rear wheels which could explain the agreement between simulation and experiment in U5 and U6 plots in Figure 39.

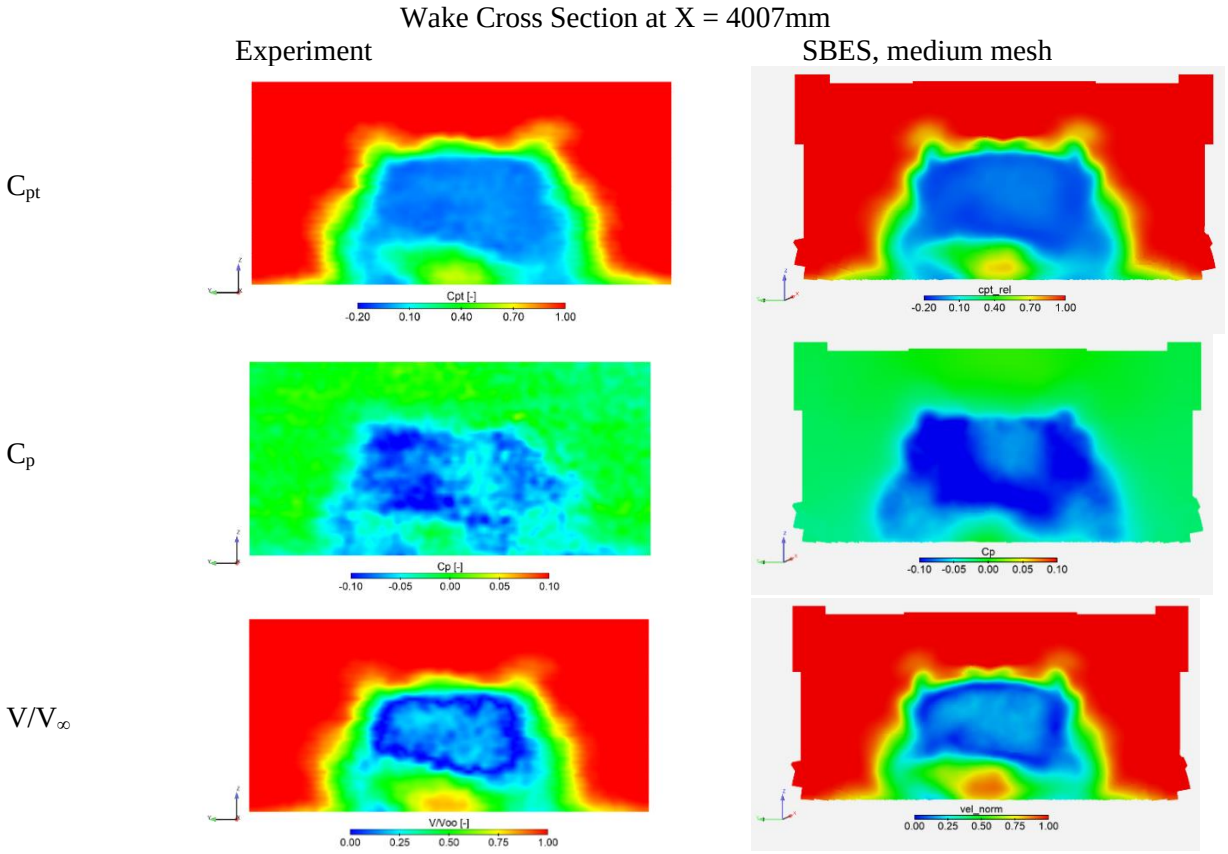


Figure 40: Comparison between medium mesh SBES simulation and experiment at X = 4007mm, C_{pt}, C_p and normalized velocity magnitude contours

Front Wheel Wake Section at X = 407mm

Experiment SBES, medium mesh

C_{pt}

C_p

V/V_∞

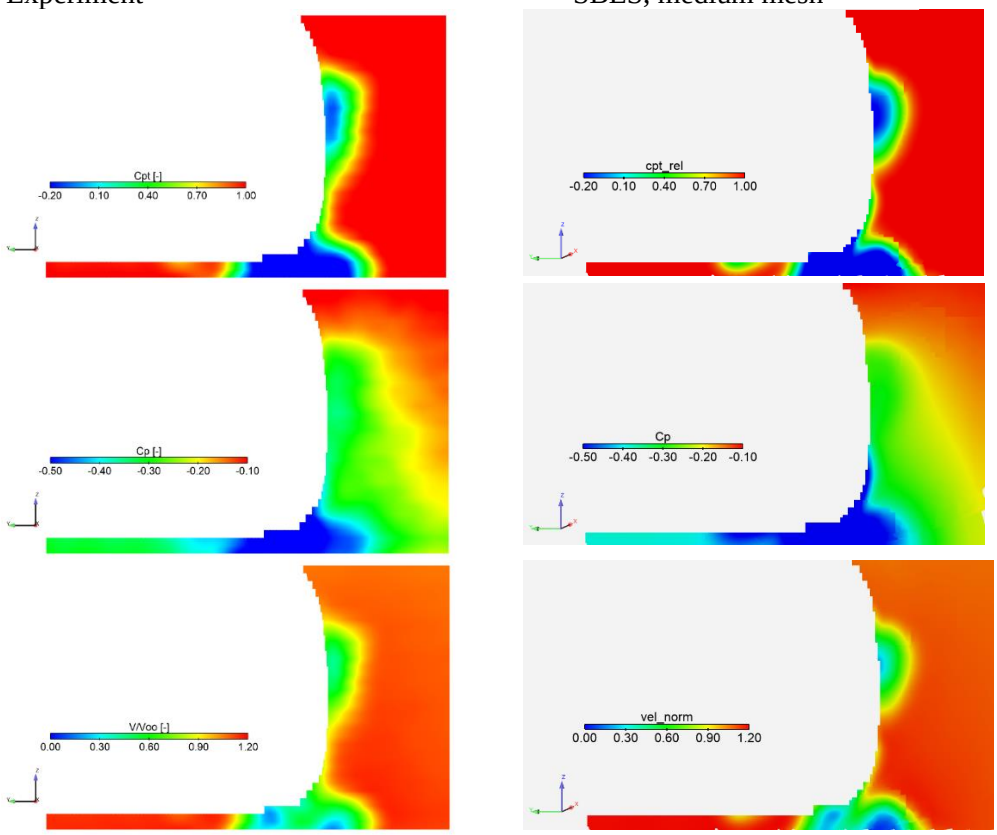
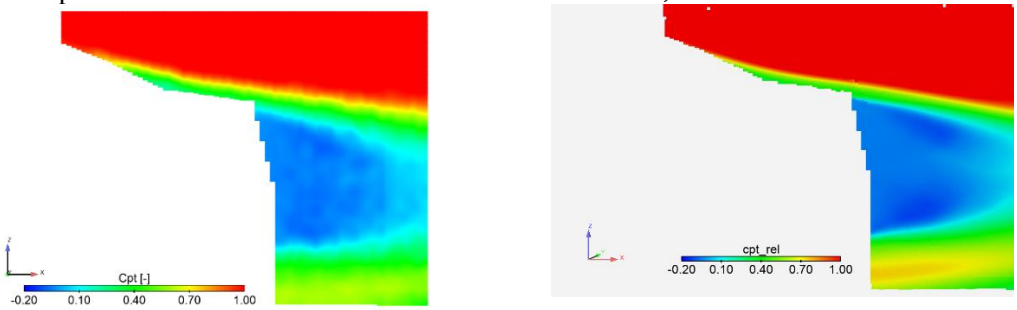


Figure 41: Comparison between medium mesh SBES simulation and experiment at X = 407mm, C_{pt}, C_p and normalized velocity magnitude contours

Vehicle Wake Cross Section at y = 0mm

Experiment SBES, medium mesh

C_{pt}



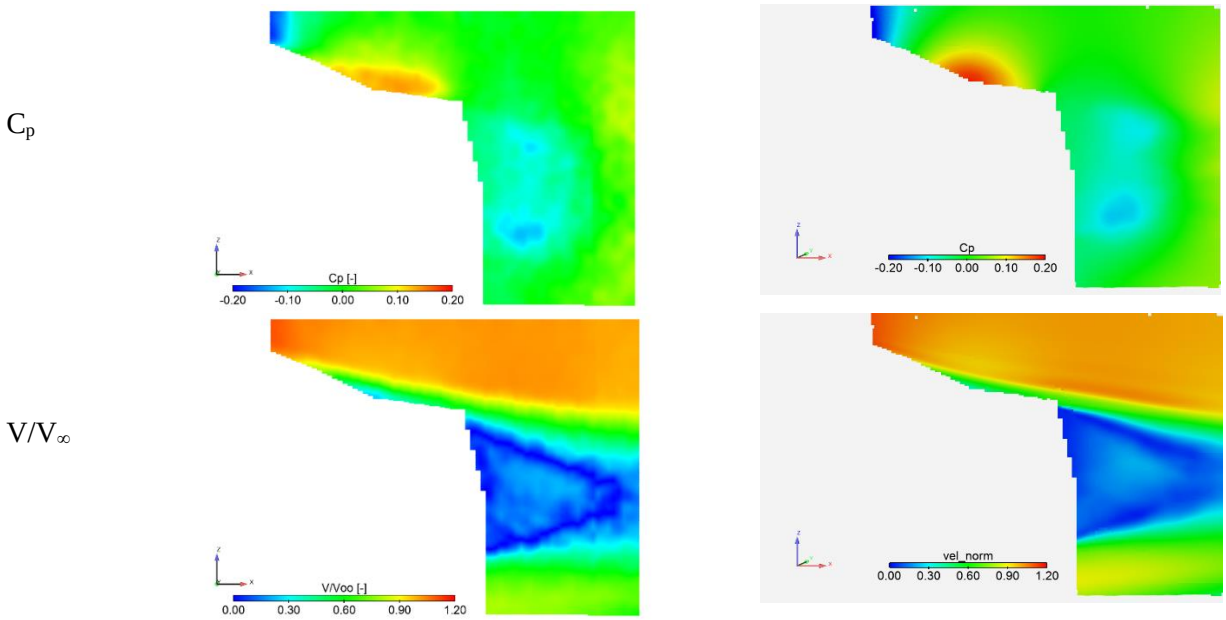


Figure 42: Comparison between medium mesh SBES simulation and experiment at Y=0mm, C_{pt}, C_p and normalized velocity magnitude contours

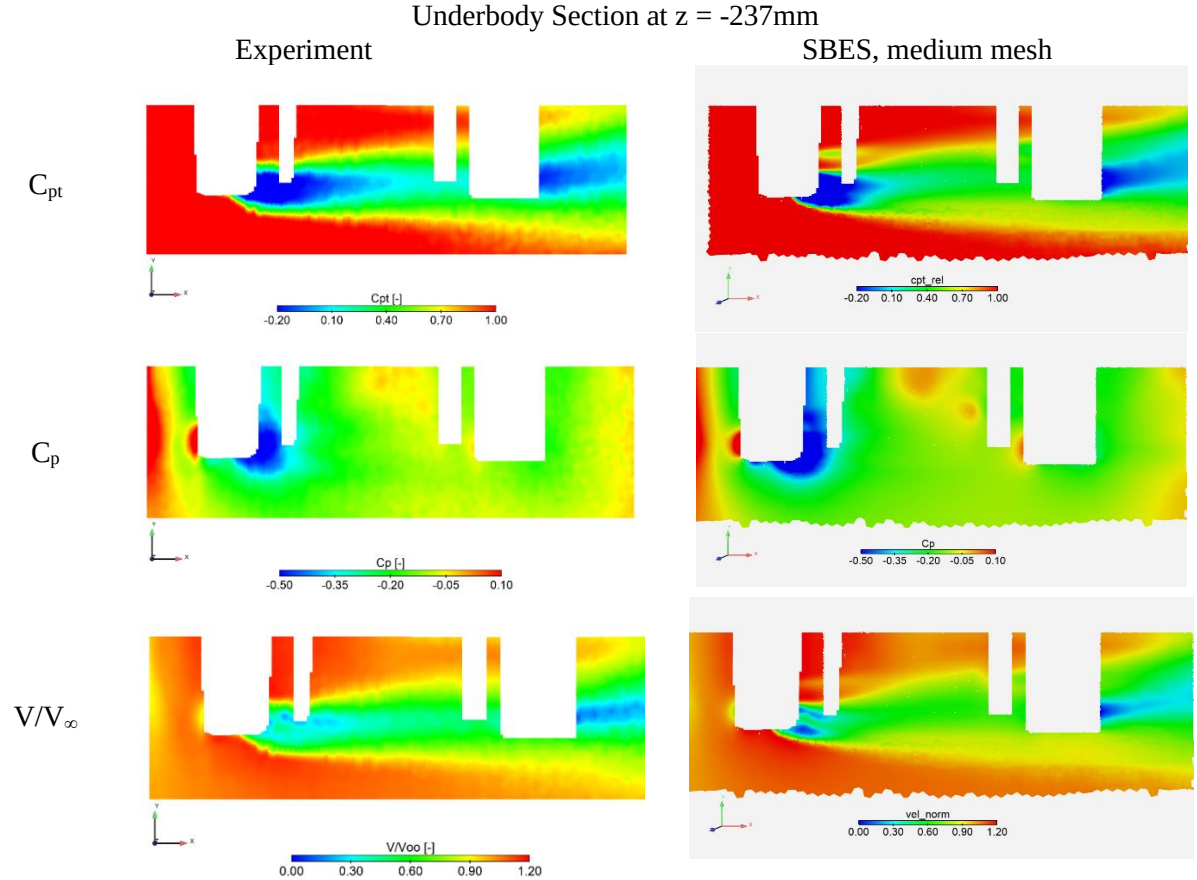


Figure 43: Comparison between medium mesh SBES simulation and experiment at Z=-237mm, C_{pt}, C_p and normalized velocity magnitude contours.

Finally taking a look at the cumulative plot of averaged C_d distribution along the x axis of the car in Figure 44, the areas that contribute more to the drag force are towards the rear of the car, which ties with the above comparisons between simulation and experiments.

The C_l cumulative plot in Figure 44 shows that the highest lift force, or downforce in this case, is generated along the middle of the car in the x axis, between the wheels, in the suction area due to the floor's proximity to the ground. This is also known as ground effect.

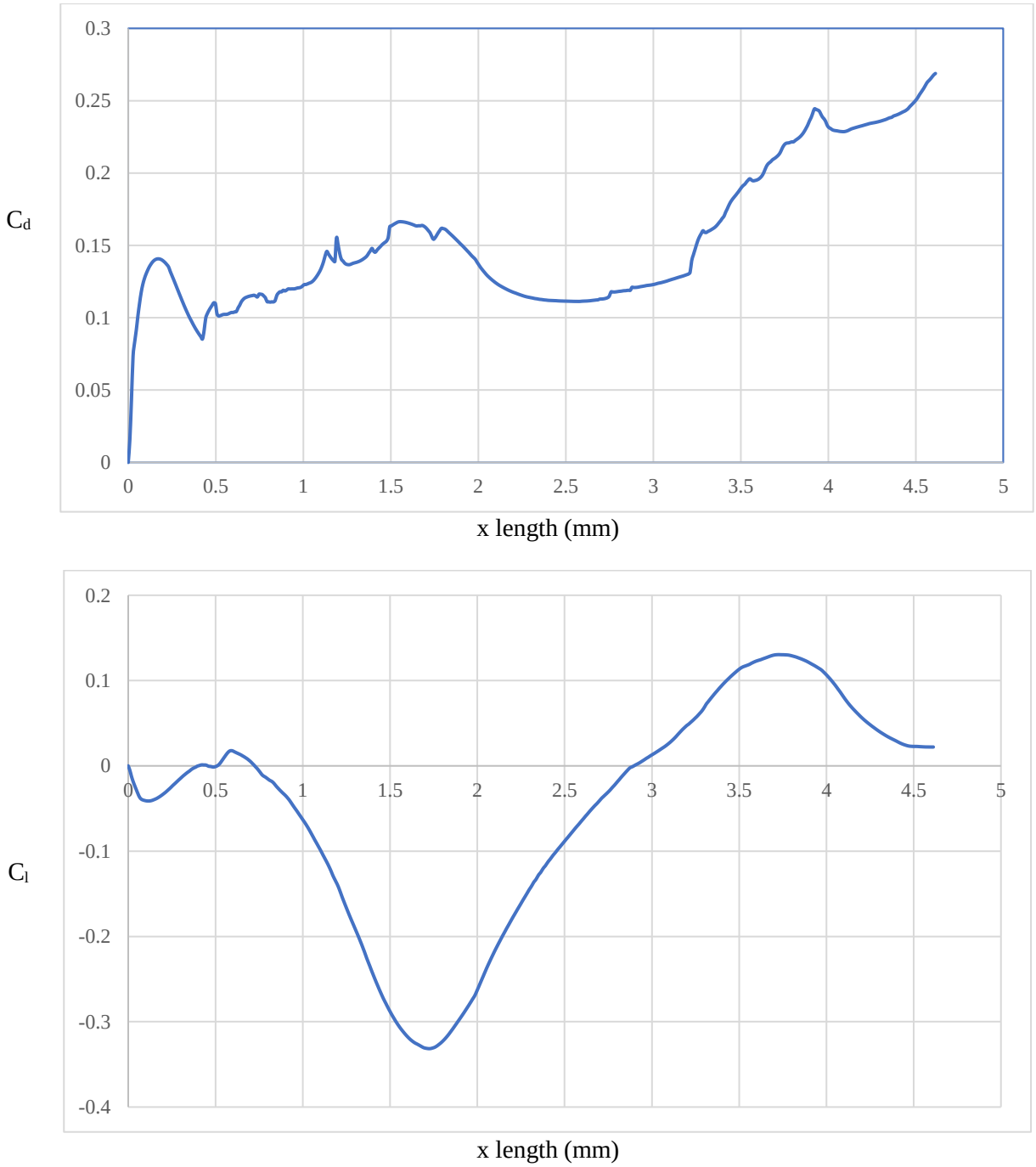


Figure 44: C_d and C_l cumulative plots over x length of the DrivAer geometry, medium mesh SBES simulation.

Next, taking a looκ at the pressure distribution on the upper centerline of the car, C_p plot in Figure 45 shows agreement of the SBES simulation with the experimental data in most parts. The biggest difference is located towards the rear window, where simulation seems to overpredict separation on the top part of the rear window and then underpredict it on the lower part where it the trunk. Looking again in the C_p contours in Figure 42, the observation can be confirmed as higher pressure values appear in the SBES contour in comparison to the experimental one in the lower part of the rear window, and lower pressure values towards the top of the window.

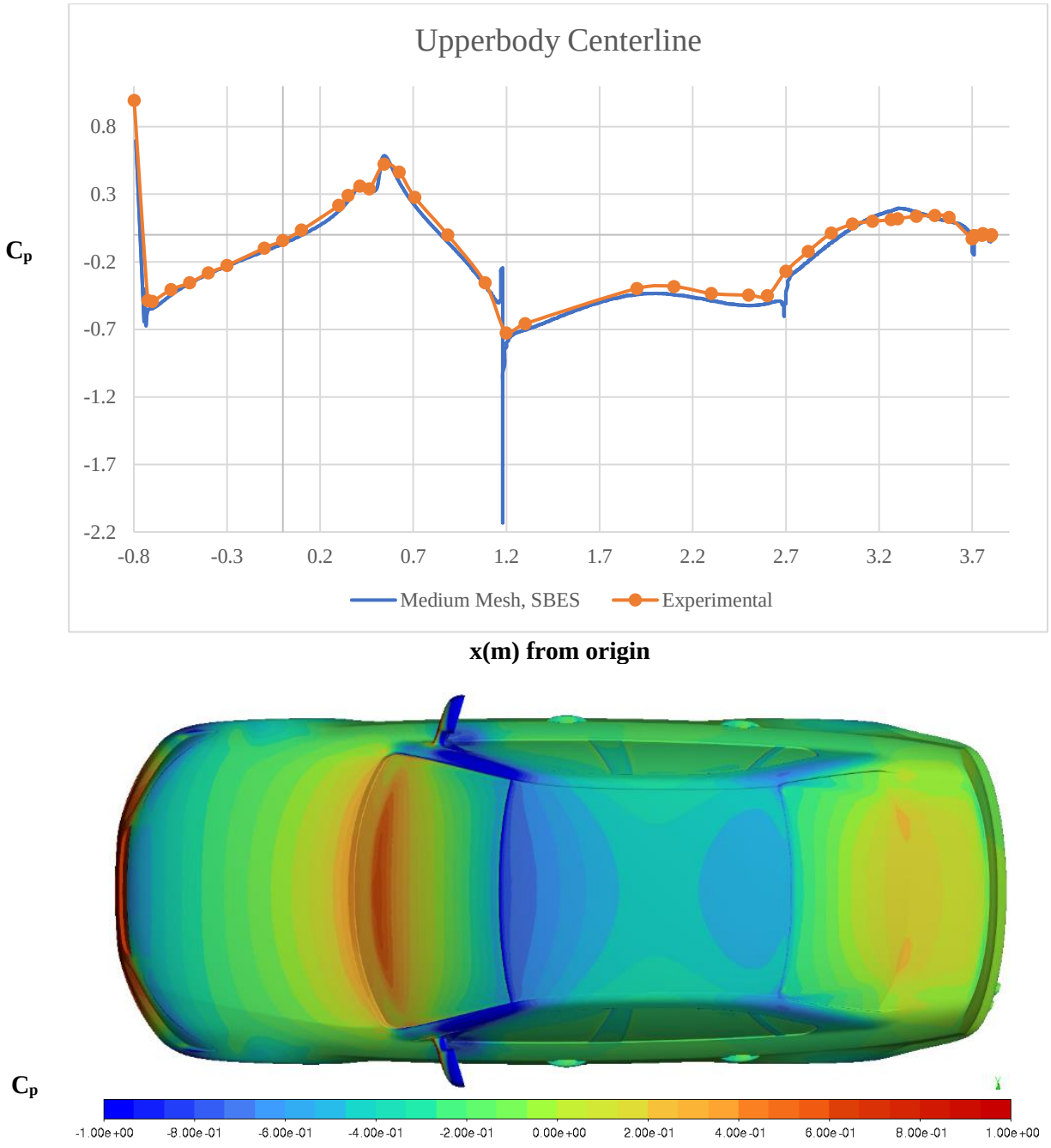


Figure 45: Upperbody centerline Cp distribution on the DrivAer model, medium mesh SBES simulation

4.4 Turbulent flow structures

The three following figures show Q criterion raw of 20000 iso-surfaces colored by x-vorticity. Turbulent flow structures appear in the wall boundary layer and in the wake region in the rear of the car, behind the mirrors and the wheels.

Q criterion is defined as, [27]:

$$Q = \frac{1}{2}(\Omega_{ij}\Omega_{ij} - S_{ij}S_{ij}) \tag{4.1}$$

Where Ω_{ij} and S_{ij} are the antisymmetric and symmetric parts of the velocity gradient tensor $\frac{\partial u_i}{\partial x_j}$, respectively.

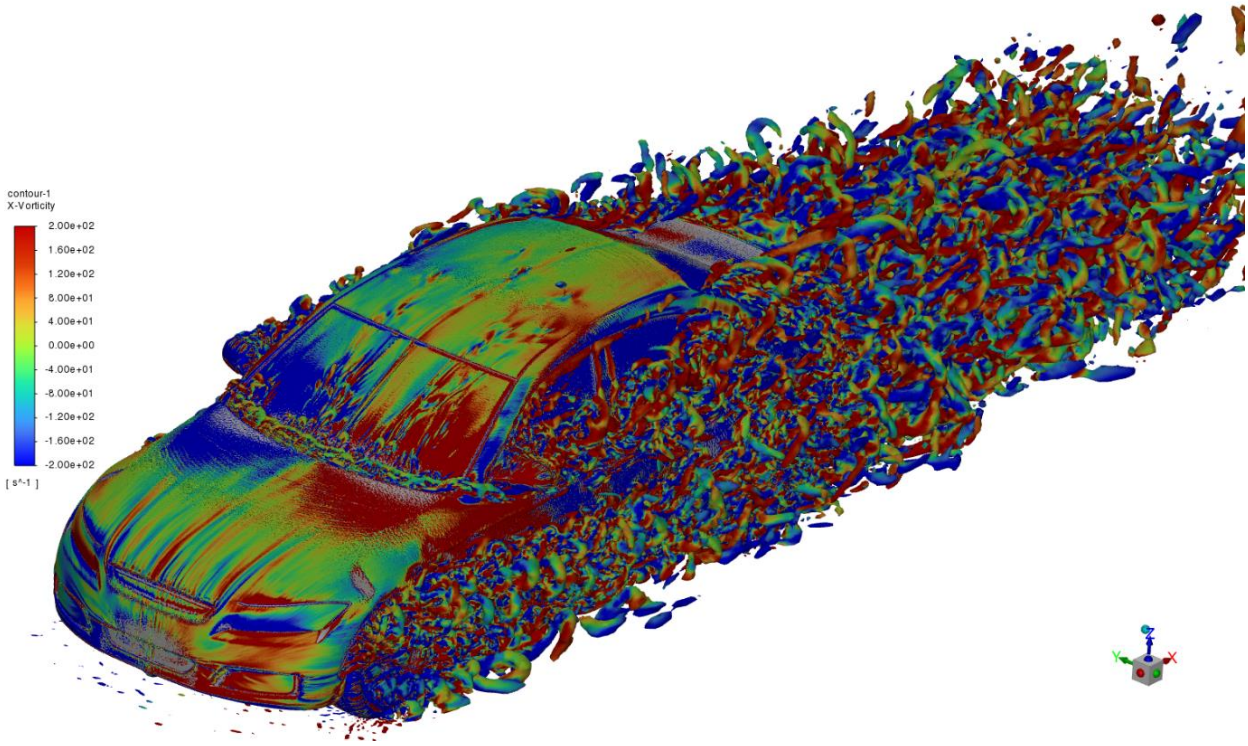


Figure 46: Q criterion raw with a value of 20000

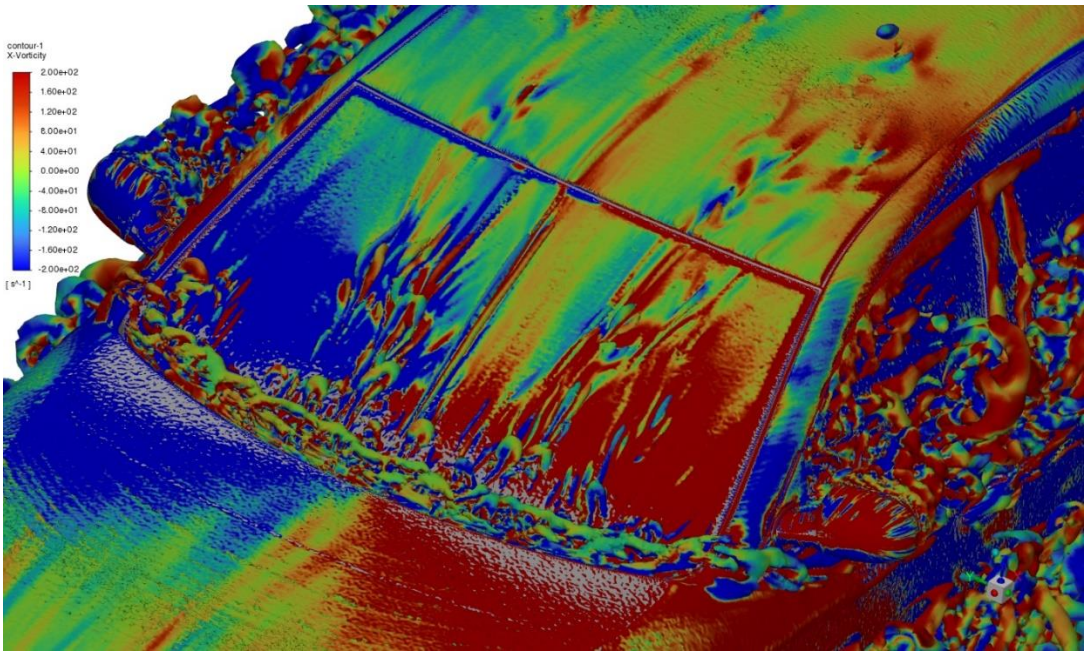


Figure 47: Q criterion raw with a value of 20000, front window

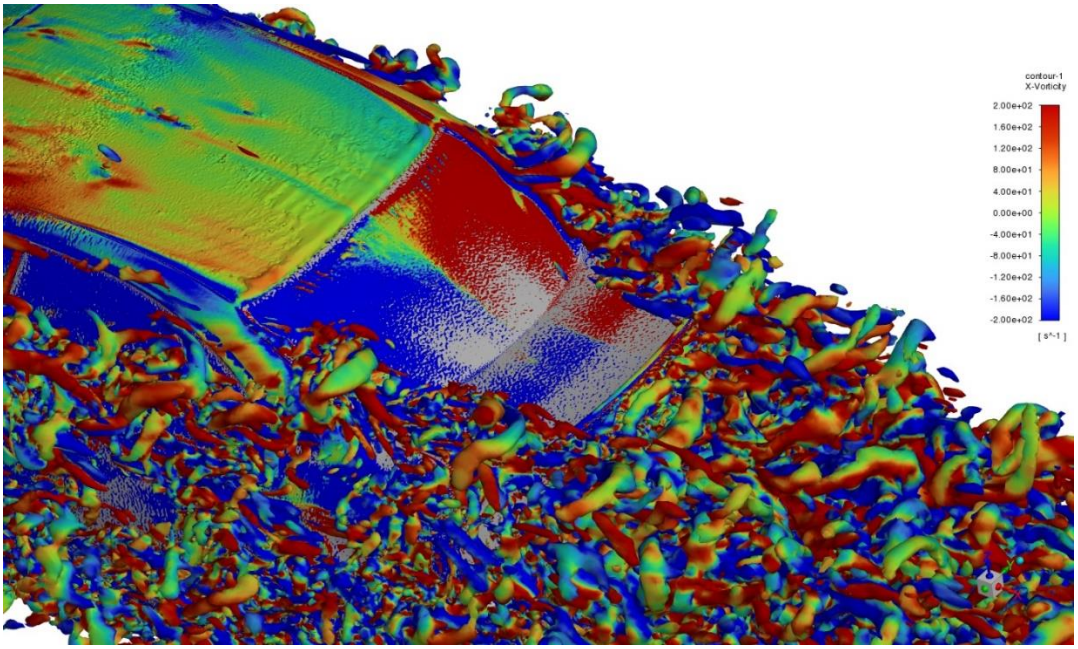


Figure 48: Q criterion raw with a value of 20000, rear window

4.5 Timings

4.5.1 Meshing time

As the purpose of this thesis is to propose a fast transient workflow, the meshing time should also be added in the total timing calculation. Mosaic meshing enables for massive parallel volume meshing. Yet, surface mesh generation was done in serial. A volume mesh scalability test was conducted using 5 different numbers of CPU cores, as shown in Figure 49. For example, with 128 cores approximately 8 million cells are being generated per minute time.

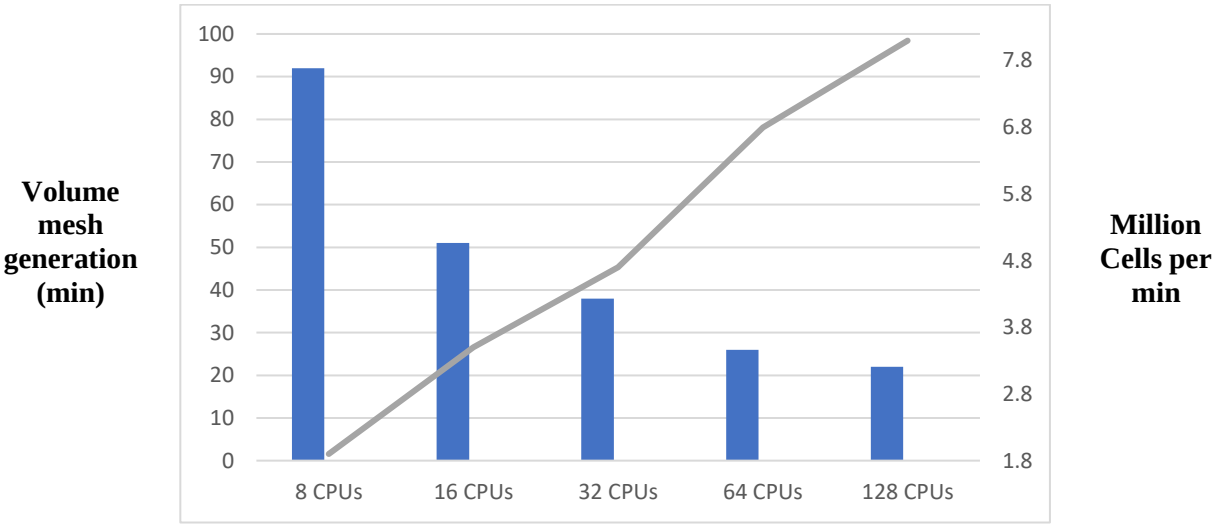


Figure 49: Poly-Hexcore mosaic mesh scalability for volume mesh generation.

In order to attain an approximate total timing for mesh generation, 32 CPU cores were selected, which give 67 min in total.

4.5.2 Solution time

The timings of the steady-state and transient simulations are shown in the following table.

Table 16: Solution timings for the RANS and SBES simulations

Simulation	*Simulation time (min): 320 CPU cores*	*CPUh*
RANS 400 iterations	**93.4**	**498.13**
SBES	**5770**	**30773.33**

In 320 CPU cores, the total simulation time is 5863.4 min, or 97.7 hours. In CPUh, which is the product of the amount of CPU cores used and the hours it took for the simulation to be completed, the total simulation time is 31271.47 CPUh. Due to the scalability of the Fluent solver, by using double the amount of CPU cores, the simulation time would be cut in half, so with 640 CPU cores it would take around 2 days for a simulation using this particular workflow and similar total cell count to be completed. Considering the mean amount of CPU cores used in the industry for similar applications to be 2048, the required solution time is 15 hours.

5. Conclusion and Future work

The proposed workflow provides significant speedup for external aerodynamic simulations. With the calculated overall time, car designers could run up to 11 simulations within a week using on average 2048 CPU cores, during the design circle of the aerodynamic package.

The mesh independence study has indicated that a modest cell count is sufficient to obtain accurate time-averaged values for the aerodynamic forces. The SBES simulation with the medium level of mesh refinement, 178 million cells, captured most of the off-body turbulent flow structures. This simulation shows good agreement with the experimental measurements for velocity and pressure values. The biggest differences between simulation and experiment lie in the vortices generated from the outer and lower edges of the tires and the wake region. This could indicate an overprediction of separation using the SBES and the GEKO models.

The comparison between the simulation and the experiment with regards the force coefficient needs a more careful interpretation. The overall trends are the same, yet the difference in values for C_d, for example, should not considered an absolute measure when selecting a simulation setup. In the automotive industry the difference between two design iterations is ultimately more important when comparing simulation with experiment.

In order to ensure a better correlation between simulation and experiment, the same workflow should be tested on different variants of the DrivAer model, following the example set at the 3rd AutoCFD Prediction workshop in 2022. Such a simulation setup would be sufficient to capture the trends between two different designs, even when the absolute values of drag do not match the experiment exactly.

To improve the accuracy of the turbulence modelling, a first step could be to tune the GEKO model coefficients to try to better capture the separation behavior.

The current trend in the automotive industry is to move towards a Wall-Function based Large Eddy Simulation (WF-LES) workflow. This approach has shown significantly better correlation with experimental data for the Closed Cooling DrivAer Notchback case, as described in [22]. Yet, such a simulation would significantly increase the cell count requirement and thus the computational time. The "Rapid Octree" method could then be used instead of the "Poly-Hexcore" method for the mesh generation, as it provides a fast, scalable, and robust octree generation. This type of workflow, with the WF-LES numerical model and a Rapid Octree mesh type, when used on the DrivAer model, has shown significant improvements in the prediction of the off-body flow phenomena.

The DrivAer case was used on both the Ansys Fluent Multi-GPU solver and the Fluent CPU solver for the 3rd AutoCFD prediction workshop [26]. The comparison of the speedup between the two solvers showed the Multi-GPU solver to be almost 4 times faster than the CPU solver. Following the theme of the present thesis, the next step towards a faster transient workflow is the use of the Multi-GPU solver.

Appendix

A1. *Continuity equation*

The continuity equation for a finite control volume fixed in space in integral form is given as:

$$\frac{\partial}{\partial t}\iiint_V \rho dV + \oiint_S \rho \boldsymbol{V} \cdot \boldsymbol{dS} = 0 \tag{A.1.1}$$

Applying the divergence theorem, the second term is expressed as:

$$\oiint_S \rho \boldsymbol{V} \cdot \boldsymbol{dS} = \iiint_V \nabla \cdot (\rho \boldsymbol{V}) dV \tag{A.1.2}$$

and the continuity equation in the form of partial differential equation is obtained:

$$\iiint_V \left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V})\right] dV = 0 \Rightarrow \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}) \tag{A.1.3}$$

A2. Derivation of the Navier Stokes equation

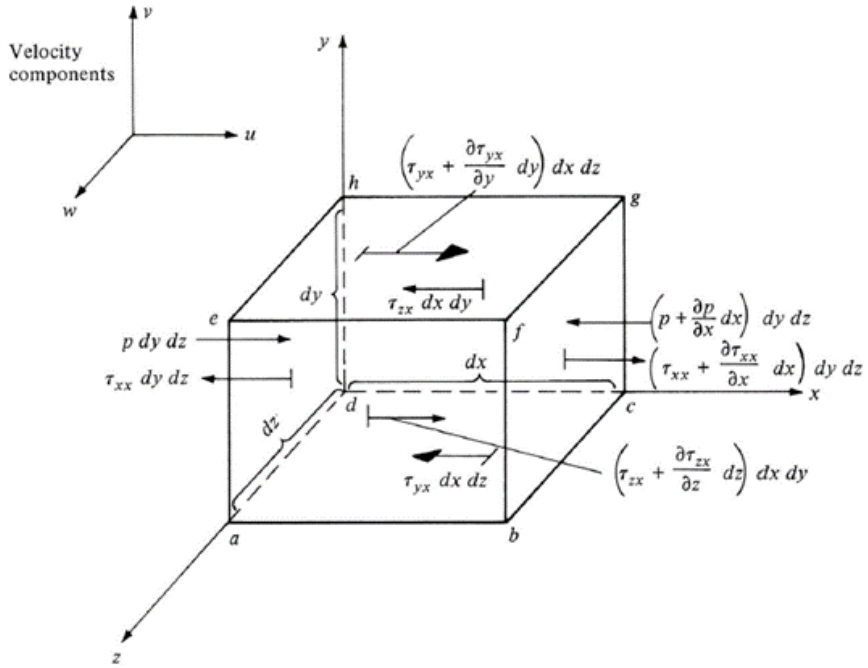


Figure 50: Forces acting on a fluid element [8].

For an infinitesimally small moving fluid element of fixed mass, as it is depicted in Figure 50, Newton's second law **F = ma** is being applied. In the x direction, $F_x = ma_x$, where F_x is the sum of all the body and surface forces acting on the fluid element in the x direction. Neglecting the body forces, the net force acting on the element is due to the pressure and viscous stress distributions over the surface of the element. For

example, on face abcd, the only force in the x di-rection is that due to shear stress, (τ_{yx}dxdz). Face efgh is a distance dy above face abcd; hence, the shear force in the x direction on face efgh is [τ_{yx} +($\partial\tau_{yx}/\partial$y) dy]dxdz. The directions of the shear stress on faces abcd and efgh; on the bottom face, τ_{yx} is to the left (the negative x direction), whereas on the top face, τ_{yx} +($\partial\tau_{yx}/\partial$y) dy is to the right (the positive x direction). These directions are due to the convention that positive increases in all three components of velocity, **u, v**, and **w**, occur in the positive directions of the axes.
The net force in the x direction acting on the fluid element:

$$F_x = \left[p - \left(p + \frac{\partial p}{\partial x}dx\right)\right]dydz + \left[\left(\tau_{xx} + \frac{\partial \tau_{xx}}{\partial x}dx\right) - \tau_{xx}\right]dydz + \left[\left(\tau_{xy} + \frac{\partial \tau_{xy}}{\partial y}dy\right) - \tau_{xy}\right]dxdz + \left[\left(\tau_{xz} + \frac{\partial \tau_{xz}}{\partial z}dz\right) - \tau_{xz}\right]dxdy \Rightarrow F_x = \left(-\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z}\right)dxdydz \quad \text{(A.2.1)}$$

The mass of the fluid element is fixed and equal to:

$$m = \rho dxdydz \quad \text{(A.2.2)}$$

while the component of acceleration in the x direction, denoted by a_x, is simply the time rate of change of the velocity u; this time rate of change is given by the substantial derivative.

$$a_x = \frac{Du}{Dt} \quad \text{(A.2.3)}$$

Substituting equations A2.1, A2.2 and A2.3 in Newton's second law the following scaler equations are derived:

$$\rho\frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} \quad \text{(A.2.4)}$$

$$\rho\frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z} \quad \text{(A.2.5)}$$

$$\rho\frac{Dw}{Dt} = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} \quad \text{(A.2.6)}$$

These, along with the continuity equation and an equation for energy, constitute the Navier Stokes equations.

For the shear stresses:

$$\tau_{xy} = \tau_{yx} = \mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right) \quad \text{(A.2.7)}$$

$$\tau_{xz} = \tau_{zx} = \mu\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right) \quad \text{(A.2.8)}$$

$$\tau_{zy} = \tau_{yz} = \mu\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \quad \text{(A.2.9)}$$

$$\tau_{xx} = \lambda(\nabla \cdot V) + 2\mu\left(\frac{\partial u}{\partial x}\right) \quad \text{(A.2.10)}$$

$$\tau_{yy} = \lambda(\nabla \cdot V) + 2\mu\left(\frac{\partial v}{\partial y}\right) \quad \text{(A.2.11)}$$

$$\tau_{zz} = \lambda(\nabla \cdot V) + 2\mu(\frac{\partial w}{\partial z}) \tag{A.2.12}$$

Where λ=-(2/3)μ. So the Navier-Stokes equations can be written as:

$$\rho\frac{\partial u}{\partial t} + \rho u\frac{\partial u}{\partial x} + \rho v\frac{\partial u}{\partial y} + \rho w\frac{\partial u}{\partial z} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\left(\lambda(\nabla \cdot V) + 2\mu\left(\frac{\partial u}{\partial x}\right)\right) + \frac{\partial}{\partial y}\left(\mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)\right) + \frac{\partial}{\partial z}\left(\mu\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)\right) \tag{A.2.13}$$

$$\rho\frac{\partial v}{\partial t} + \rho u\frac{\partial v}{\partial x} + \rho v\frac{\partial v}{\partial y} + \rho w\frac{\partial v}{\partial z} = -\frac{\partial p}{\partial y} + \frac{\partial}{\partial x}\left(\mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)\right) + \frac{\partial}{\partial y}\left(\lambda(\nabla \cdot V) + 2\mu\left(\frac{\partial v}{\partial y}\right)\right) + \frac{\partial}{\partial z}\left(\mu\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)\right) \tag{A.2.14}$$

$$\rho\frac{\partial w}{\partial t} + \rho u\frac{\partial w}{\partial x} + \rho v\frac{\partial w}{\partial y} + \rho w\frac{\partial w}{\partial z} = -\frac{\partial p}{\partial z} + \frac{\partial}{\partial x}\left(\mu\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)\right) + \frac{\partial}{\partial y}\left(\mu\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)\right) + \frac{\partial}{\partial z}\left(\lambda(\nabla \cdot V) + 2\mu\left(\frac{\partial w}{\partial z}\right)\right) \tag{A.2.15}$$

Information on the above topic can be found in [8].

A3. Reynolds decomposition

Reynolds decomposition is a mathematical technique used to separate the expectation value of a quantity from its fluctuations. For a quantity **u** the decomposition would be:

$$u(x,y,z,t) = \overline{u(x,y,z,t)} + u'(x,y,z,t) \tag{A.3.1}$$

Where $\bar{u}$ is the mean (time-averaged) component of the velocity field and u' is the fluctuating component. A Reynolds operator possesses some important properties.

- the mean of the fluctuating quantity is equal to zero. $\overline{u'} = 0$
- $\overline{u_i'\overline{u_j}} = 0$
- $\overline{\bar{u}_i} = \bar{u}_i$
- $\overline{u_i u_j} = \bar{u}_i\bar{u}_j + \overline{u_i'u_j'}$

The Reynolds averaging process of the Navier-Stokes equations, for example, for the x component is as follows:

$$\rho\left(\frac{\partial u}{\partial t}+\mathrm{u}\frac{\partial u}{\partial x}+\mathrm{v}\frac{\partial u}{\partial y}+\mathrm{w}\frac{\partial u}{\partial z}\right)=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right) \quad \text{(A.3.2)}$$

$$\Rightarrow \rho\left(\frac{\partial u}{\partial t}+\frac{\partial uu}{\partial x}+\frac{\partial vu}{\partial y}+\frac{\partial wu}{\partial z}\right)=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)$$

applying next the Reynolds averaging for the velocity component:

$$\rho\left(\frac{\partial(\bar{u}+u')}{\partial t}+\frac{\partial(\bar{u}+u')(\bar{u}+u')}{\partial x}+\frac{\partial(\bar{u}+u')(\bar{v}+v')}{\partial y}+\frac{\partial(\bar{u}+u')(\bar{w}+w')}{\partial z}\right) \quad \text{(A.3.3)}$$

$$=-\frac{\partial(\bar{p}+p')}{\partial x}+\mu\left(\frac{\partial^2(\bar{u}+u')}{\partial x^2}+\frac{\partial^2(\bar{u}+u')}{\partial y^2}+\frac{\partial^2(\bar{u}+u')}{\partial z^2}\right)\Rightarrow$$

$$\rho\left(\frac{\partial(\bar{u}+u')}{\partial t}+\frac{\partial[(\bar{u}\bar{u})+(\bar{u}u')+(\bar{u}u')+(u'u')]}{\partial x}+\frac{\partial[(\bar{u}\bar{v})+(\bar{v}u')+(\bar{u}v')+(v'u')]}{\partial y}\right. \quad \text{(A.3.4)}$$

$$\left.+\frac{\partial[(\bar{w}\bar{u})+(\bar{w}u')+(\bar{u}w')+(w'u')]}{\partial z}\right)$$

$$=-\frac{\partial(\bar{p}+p')}{\partial x}+\mu\left(\frac{\partial^2(\bar{u})}{\partial x^2}+\frac{\partial^2(u')}{\partial x^2}+\frac{\partial^2(\bar{u})}{\partial y^2}+\frac{\partial^2(u')}{\partial y^2}+\frac{\partial^2(\bar{u})}{\partial z^2}+\frac{\partial^2(u')}{\partial z^2}\right)$$

By taking the average of the entire equation we get:

$$\rho\left(\overline{\frac{\partial(\bar{u}+u')}{\partial t}}+\overline{\frac{\partial[(\bar{u}\bar{u})+(\bar{u}u')+(\bar{u}u')+(u'u')]}{\partial x}}+\overline{\frac{\partial[(\bar{u}\bar{v})+(\bar{v}u')+(\bar{u}v')+(v'u')]}{\partial y}}\right. \quad \text{(A.3.5)}$$

$$\left.+\overline{\frac{\partial[(\bar{w}\bar{u})+(\bar{w}u')+(\bar{u}w')+(w'u')]}{\partial z}}\right)$$

$$=-\overline{\frac{\partial(\bar{p}+p')}{\partial x}}+\mu(\overline{\frac{\partial^2(\bar{u})}{\partial x^2}}+\overline{\frac{\partial^2(u')}{\partial x^2}}+\overline{\frac{\partial^2(\bar{u})}{\partial y^2}}+\overline{\frac{\partial^2(u')}{\partial y^2}}+\overline{\frac{\partial^2(\bar{u})}{\partial z^2}}+\overline{\frac{\partial^2(u')}{\partial z^2}})$$

Using the properties of the Reynolds operator the above equation takes the following form:

$$\rho\left(\frac{\partial\bar{u}}{\partial t}+\frac{\partial(\bar{u}\bar{u})}{\partial x}+\frac{\partial\overline{(u'u')}}{\partial x}+\frac{\partial(\bar{u}\bar{v})}{\partial y}+\frac{\partial\overline{(v'u')}}{\partial y}+\frac{\partial(\bar{w}\bar{u})}{\partial z}+\frac{\partial\overline{(w'u')}}{\partial z}\right) \quad \text{(A.3.6)}$$

$$=-\frac{d\bar{p}}{dx}+\mu\left(\overline{\frac{\partial^2(\bar{u})}{\partial x^2}}+\overline{\frac{\partial^2(\bar{u})}{\partial y^2}}+\overline{\frac{\partial^2(\bar{u})}{\partial z^2}}\right)\Rightarrow$$

$$\underbrace{\rho\left(\frac{\partial\bar{u}}{\partial t}+\bar{u}\frac{\partial(\bar{u})}{\partial x}+\bar{v}\frac{\partial(\bar{u})}{\partial y}+\bar{w}\frac{\partial(\bar{u})}{\partial z}\right)}_{A}+\underbrace{\rho\left(\frac{\partial\overline{(u'u')}}{\partial x}+\frac{\partial\overline{(v'u')}}{\partial y}+\frac{\partial\overline{(w'u')}}{\partial z}\right)}_{B} \quad \text{(A.3.7)}$$

$$=\underbrace{-\frac{d\bar{p}}{dx}}_{C}+\underbrace{\mu\left(\overline{\frac{\partial^2(\bar{u})}{\partial x^2}}+\overline{\frac{\partial^2(\bar{u})}{\partial y^2}}+\overline{\frac{\partial^2(\bar{u})}{\partial z^2}}\right)}_{D}$$

A4. Turbulence models

A4.1 k-ε

k-ε models are two-equation turbulence models, which use two extra transport equations for the turbulent kinetic energy k and the turbulence dissipation rate ε.

Equation for k:

$$\underbrace{\frac{\partial(\rho k)}{\partial t}}_{A} + \underbrace{\nabla\cdot(\rho \boldsymbol{u} k)}_{B} = \underbrace{\nabla\cdot\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\nabla k\right]}_{C} + \underbrace{G_k + G_b - \rho\varepsilon - Y_M + S_k}_{D} \Rightarrow \frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_i)}{\partial x_i} \quad \text{(A.4.1)}$$

$$= \frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + G_k + G_b - \rho\varepsilon - Y_M + S_k$$

Where:

- A: Time evolution
- B: Convection term
- C: Diffusion term
- D: Sources and sink terms

Equation for ε:

$$\underbrace{\frac{\partial(\rho\varepsilon)}{\partial t}}_{A} + \underbrace{\nabla\cdot(\rho \boldsymbol{u}\varepsilon)}_{B} = \underbrace{\nabla\cdot\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\nabla\varepsilon\right]}_{C} + \underbrace{C_1\frac{\varepsilon}{k}(G_k + C_3 G_b) - C_2\rho\frac{\varepsilon^2}{k} + S_\varepsilon}_{D} \Rightarrow \quad \text{(A.4.2)}$$

$$\frac{\partial(\rho\varepsilon)}{\partial t} + \frac{\partial(\rho\varepsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial\varepsilon}{\partial x_j}\right] + C_1\frac{\varepsilon}{k}(G_k + C_3 G_b) - C_2\rho\frac{\varepsilon^2}{k} + S_\varepsilon$$

Where:

- A: Time evolution
- B: Convection term
- C: Diffusion term
- D: Sources and sink terms

G_k represents the generation of turbulence kinetic energy due to the mean velocity gradients, G_b is the generation of turbulence kinetic energy due to buoyancy, Y_M represents the contribution of the fluctuating dilatation in compressible turbulence to the overall dissipation rate, σ_k is the turbulent Prandtl numbers for k and S_k is user-defined source term.

C1, C2, C3 are constants, σ_ε is the turbulent Prandtl numbers for ε, S_ε is user-defined source term.

The turbulent (or eddy) viscosity, μ_t is computed by combining k and ε as follows:

$$\mu_t = C_\mu\frac{k^2}{\varepsilon} \quad \text{(A.4.3)}$$

Where C_μ is a constant.

A4.2 Standard k- ω model

The standard k-ω model is a two-equations model that includes transport equations for the turbulence kinetic energy k and the specific dissipation rate ω, which can also be thought of as the ratio of k to ε.

$\omega = \frac{\varepsilon}{C_\mu k}, \left[\frac{1}{s}\right]$, C_μ= 0.09

Equation for k:

$$\frac{\partial(\rho k)}{\partial t}+\frac{\partial(\rho k u_i)}{\partial x_i}=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right]+G_k-Y_M+S_k \quad \text{(A.4.4)}$$

Equation for ω:

$$\frac{\partial(\rho \omega)}{\partial t}+\frac{\partial(\rho \omega u_i)}{\partial x_i}=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right]+G_\omega-Y_\omega+S_\omega \quad \text{(A.4.5)}$$

G_k represents the generation of turbulence kinetic energy due to mean velocity gradients. Gω represents the generation of ω. Y_k and Yω, represent the dissipation of k and ω due to turbulence. All the above terms are calculated as described below. S_k and S_ω are user-defined source terms. The turbulent viscosity μ_t is computed as follows:

$$\mu_t=a^*\frac{\rho k}{\omega} \quad \text{(A.4.6)}$$

A4.3 Shear-Stress Transport (SST) k- ω model

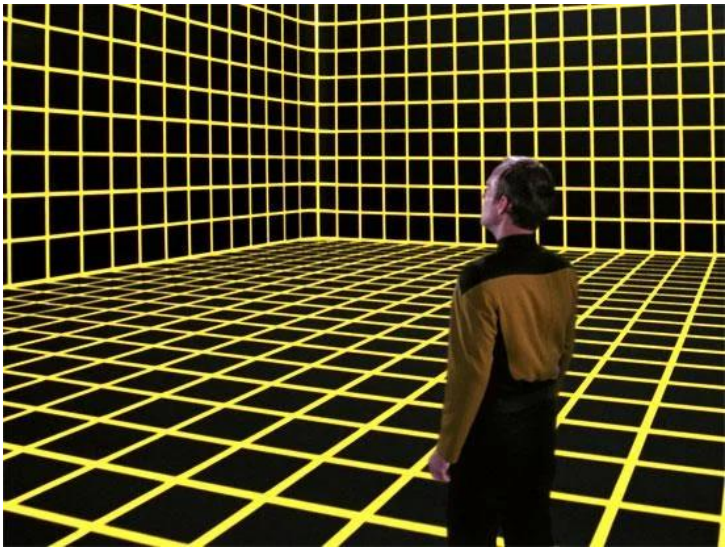
In order to combine the robust and accurate formulation of the k- ω model in the near-wall region with the free-stream independence of the k- ε model in the far field, the shear-stress transport (SST) k-ω model was developed. To achieve this, the k- ε model is converted into a k- ω formulation. The SST k-ω model is similar to the standard k- ω model, but includes the following refinements:

- The standard k- ω model and the transformed k- ε model are both multiplied by a blending function and both models are added together. The blending function is designed to be one in the near-wall region, which activates the standard k- ω model, and zero away from the surface, which activates the transformed k- ε model
- The SST model incorporates a damped cross-diffusion derivative term in the ω equation.
- The definition of the turbulent viscosity is modified to account for the transport of the turbulent shear stress.
- The modeling constants are different.

References:

1. AutoCFD2 Open Cooling DrivAer - Notchback Test Case Information, B Hupertz, July 6th, 2021 Hupertz, B., Chalupa, K., Krueger, L., Howard, K. et al., "On the Aerodynamics of the Notchback Open Cooling DrivAer: A Detailed Investigation of Wind Tunnel Data for Improved Correlation and Reference" SAE Technical Paper 2021-01-0958, 2021, doi:10.4271/2021-01-0958.
2. Hupertz, B., Lewington, N., Mockett, C., Ashton, N. et al., "Towards a Standardized Assessment of Automotive Aerodynamic CFD Prediction Capability - AutoCFD 2: Ford DrivAer Test Case Summary," SAE Technical Paper 2022-01-0886, 2022
3. Heft, A., Indinger, T., and Adams, N., "Introduction of a New Realistic Generic Car Model for Aerodynamic Investigations," SAE Technical Paper 2012-01-0168, 2012
4. Hupertz, B., Kruger, L., Chalupa, K. et al., "Introduction of a New Full-Scale Open Cooling Version of the DrivAer Generic Car Model" 11th FKFS Conference "Progress in Vehicle Aerodynamics and Thermal Management", Stuttgart, 2017.
5. AutoCFD2 Open Cooling DrivAer -Notchback, Test Case Information, B Hupertz, July 6th, 2021 PowerPoint Presentation, https://autocfd2.s3.eu-west-1.amazonaws.com/test-cases/case2/AutoCFD2_OCDA_Info_210706.pdf
6. https://autocfd2.s3.eu-west-1.amazonaws.com/test-cases/presentations/Vangelis+Skaperdas+-+Skaperdas_BETA_CAE_2ndACW.pdf
7. https://autocfd.eng.ox.ac.uk/
8. Bruce R. Munson, Donald F. Young, Theodore H. Okiishi, "Foundamentals of Fluid Mechanics 8th edition".
9. Ansys, Inc. "Ansys Fluent Theory Guide", Release 2022 R2.
10. Ansys, Inc. "Fluent User's Guide", Release 2022 R2
11. F.R. Menter, R. Lechner and A. Matyushenko "Best Practice: RANS Turbulence Modeling in Ansys CFD", Version 1.0.
12. F.R. Menter "Best Practice: Scale-Resolving Simulations in Ansys CFD", Version 2.00.
13. F.R. Menter, R. Lechner and A. Matyushenko "Best Practice: Generalized k-ω Two-Equation Turbulence Model in Ansys CFD (GEKO)".
14. Ansys Learning Hub "Real External Flows – Lesson 4: Flow Separation and Reattachment"
15. Ansys Learning Hub "Ansys CFD (Fluent & CFX) Turbulence Modelin, Lecture 03: Near Wall Modeling"
16. https://www.ansys.com/products/fluids/ansys-fluent/mosaic-meshing
17. COMMISSION IMPLEMENTING REGULATION (EU) 2019/1916 of 15 November 2019 laying down detailed provisions as regards the use of rear aerodynamic devices pursuant to Council Directive 96/53/EC.
18. M. Varney, M. Passmore, F. Wittmeier, T. Kuthada, "Experimental Data for the Validation of Numerical Methods: DrivAer Model", 8 December 2020.
19. https://www1.grc.nasa.gov/beginners-guide-to-aeronautics/what-is-drag/
20. https://www.grc.nasa.gov/www/k-12/rocket/presar.html
21. https://www.grc.nasa.gov/www/k-12/BGP/boundlay.html
22. R. Winstanley, D. Flad, A. Main, R. Jia, R. Borker, A. Singh, J. Maruszewski and F. Menter "Next Generation Ansys GPU CFD Solver Technology for External Aerodynamic Applications"
23. https://movotiv.com/statistics
24. F. H. Abernathy "Film Notes for Fundamentals of Boundary Layers", National Committee for Fluid Mechanics Films, No 21623.

25. S. McBeath “Competition Car Aerodynamics”, 3rd Edition, 2015, Veloce Publishing Limited, Veloce House
26. Th1720_3rdAutoCFDPW_ANSYS_RW.pdf (amazonaws.com)
27. https://books.google.gr/books?id=FWmsrNv3BYoC&pg=PA23&lpg=PA23&dq=Q+criterion&source=bl&ots=CY63QRmgk3&sig=ACfU3U2IRLr3VB6NMxLlN5auOrgtvkDVXg&hl=en&sa=X&ved=2ahUKEwj498PQguX8AhVegf0HHZAvC24Q6AF6BAgYEAM#v=onepage&q=Q%20criterion&f=false



Will the future of simulation look like the Holodeck from Star Trek?

Live long and prosper my friends!