CHALMERS

Method Development for CFD Analysis of Heavy Truck Aerodynamics

Investigations of Effects of Partitioning for Parallel Simulations, Bi-stable Flow States for Steady Simulations, and Effects of Split-line Variation on Drag

Master's thesis in Applied Mechanics

HENRIK LARSSON

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2020

MASTER'S THESIS IN APPLIED MECHANICS

Method Development for CFD Analysis of Heavy Truck Aerodynamics

Investigations of Effects of Partitioning for Parallel Simulations, Bi-stable Flow States for Steady Simulations, and Effects of Split-line Variation on Drag

HENRIK LARSSON

Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2020

Method Development for CFD Analysis of Heavy Truck Aerodynamics
Investigations of Effects of Partitioning for Parallel Simulations, Bi-stable Flow States for Steady Simulations, and Effects of Split-line Variation on Drag
HENRIK LARSSON

Master's thesis 2020:46
Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: +46 (0)31-772 1000

Cover:
Difference in velocity magnitude between the two bi-stable modes investigated in Part 2 of this thesis.

Chalmers Reproservice
Göteborg, Sweden 2020

Method Development for CFD Analysis of Heavy Truck Aerodynamics
Investigations of Effects of Partitioning for Parallel Simulations, Bi-stable Flow States for Steady Simulations, and Effects of Split-line Variation on Drag
Master's thesis in Applied Mechanics
HENRIK LARSSON
Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
Chalmers University of Technology

ABSTRACT

According to the European Commission, heavy trucks are responsible for about 4% of total European CO_2 emissions. One way to lower emissions is to reduce aerodynamic drag. In order to effectively design low drag trucks, computational fluid dynamics (CFD) simulations are used for the analysis of aerodynamic performance. This thesis consists of three parts that investigate CFD simulations for truck aerodynamics in three areas. Firstly, differences appearing between parallel simulations are investigated. As truck aerodynamics is very computationally expensive, simulations are run on parallel computer clusters. It has been seen that running simulations on different clusters may cause differences in simulation results, which should not should not occur according to theory. In the second part two bi-stable flow states that appear in steady simulations are investigated. Small geometrical variations cause the solution to enter one of two modes, with no clear pattern. Lastly, real trucks will deviate from the nominal geometry to some degree, due to natural variation. Simulations are however mostly performed on nominal geometry, which leaves some uncertainty regarding the performance of real vehicles. Therefore the effects of split-line variation on drag are investigated.

A design of experiment (DOE) approach was used to identify which parallel parameters were affecting results, both in parallel meshing and parallel simulations. It was found that parallel effects on the meshing was negligible. The differences were found to appear when running simulations on different clusters and different number of processors. The cause for the difference was an underlying instability in the form of the bi-stable flow states. Differences in the algebraic multigrid (AMG) linear solver or in floating point errors caused sensitive simulations to change mode. Differences between clusters could be eliminated using the command `-mppflags "-e MPI_COLL_FORCE_CONSTANTORDER=1"` in Star-CCM+ for clusters with Platform MPI (Message Passing Interface).

The bi-stable modes were investigated for some different cases. The flow fields, solution development, local convergence and streamwise pressure gradient were investigated. A mesh study was performed, and a turbulence model study with both Reynolds-averaged Navier-Stokes (RANS) models and a detached eddy simulation (DES) model. The modes were partially explained by interaction with the front door split-line. One of the modes, Mode 1, was found to be most probably physical. The physicality of the other mode could not be determined with certainty. It was however found not to be the dominant mode. A possible solution to the problem is to use the realizable $k-\varepsilon$ model, which was most consistent with the modes predicted by the DES. Alternatively a more accurate transient model such as DES can be used, if it can be afforded.

Split-lines in four different areas of the truck cab were investigated using a design of experiment approach. Gap and flush at minimum and maximum tolerance values were considered for the split-lines of the selected areas. It was found that the impact on drag is within ±5 DC (drag counts), and that most trucks will be within ±3 DC of nominal C_D. The most important of the investigated areas are the front corners corners of the cab. It was furthermore found that gap has a larger effect on drag than flush, for the split-lines and tolerances considered.

Keywords: CFD, Commercial Vehicles, Aerodynamics

PREFACE

This is the report for a 60 credit Master's thesis project in the field of vehicle aerodynamics and CFD analysis. The project was carried out between September 2019 and June 2020 at Volvo Group Trucks Technology on their behalf to improve the analysis methods and help improve the aerodynamic performance of the Volvo brand trucks. The project consists of three parts: an investigation of the effects of parallel computing on different computer clusters on steady CFD simulation results; an investigation of bi-stable flow flow states in steady simulations; and a study of the effects of geometrical variation on the aerodynamic drag of the vehicle.

Volvo CFD Analysis Engineer Johan Forsgren was industrial supervisor for the project and Professor Lars Davidson at Division of Fluid Dynamics, Department of Mechanics and Maritime Sciences, Chalmers University of Technology was academical supervisor and examiner.

ACKNOWLEDGEMENTS

This work has been made possible by the assistance and advice of several persons whom I would like to thank. First of all I would like to thank my supervisor Johan Forsgren for supporting me throughout the project and for practical and helpful advice, and my examiner and supervisor Lars Davidson for advice and encouragement.

I would also like to thank Mattias Hejdesten and the Cab Analysis team at Volvo for giving me the opportunity to work with this very interesting project, and the aerodynamics CFD team at Volvo for help, support and good companionship during my time at Volvo.

During this project people from various parts of the Volvo organization have contributed with knowledge, experience and valuable insight, for which I am grateful. Among these are Dennis Engberg at Volvo CAE IT, who has helped out with cluster issues and questions, and Sven Ahlinder and Krister Fredriksson who patiently cleared my initial confusion about DOE. I would also like to thank geometrical architecture and geometrical assurance engineers Patrik Stålcrona and Henrik Olsson and David Renborg, lead engineer Timo Arminen and Johan Pettersson-Strömer for help with tolerances and measured deviations, and Pierre Bevenhall, Ramo Mujanovic and Mats Wilkenson from quality engineering, production and audit for the production perspective of tolerances.

NOMENCLATURE

Abbreviations and Acronyms

CFD		Computational Fluid Dynamics
DOE		Design of Experiments
FFA		Full Factorial
RFA		Reduced Factorial
OVAT		One variable at a time
HPC		High Performance Computing/Computer/Computations
MPI		Message Passing Interface
RANS		Reynods Averaged Navier-Stokes
SKE		Standard $k-\varepsilon$ model
RKE		Realizable $k-\varepsilon$ model
SKO		Standard $k-\omega$ model
SST		Shear Stess Transport $k-\omega$ model
LES		Large Eddy Simulation
DES		Detached Eddy Simulation
TKE	k	Turbulent kinetic energy
TDR	ε	Turbulent dissipation rate
SDR	ω	Specific dissipation rate
YWA		Yaw weighted average
APG		Adverse pressure gradient
FPG		Adverse pressure gradient
FVM		Finite Volume Method
SOU		Second Order Upwind
ILU		Incomplete LU
AMG		Algebraic Multigrid
PIV		Particle Image Velocimetry

Definitions

drag count	[DC]	Drag force unit: 1 DC := $0.001C_D$

Symbols and Constants

D	$[N]$	Aerodynamic Drag Force; Drag
C_D	$[-]$	Drag coefficient
c_p	$[-]$	Pressure coefficient
$c_{p,tot}$	$[-]$	Total pressure coefficient
c_T	$[-]$	Durbin time scale limiter

Contents

1 Introduction

The motivation for this project is the ongoing effort to reduce CO_2 emissions for commercial road transports. According to the European Commission, heavy trucks are responsible for about 4% of the total CO_2 emissions in the EU. In order to reduce emissions from heavy vehicles, an EU regulation was passed in 2019, which states that the levels must be reduced by 15% from 2025 and by 30% from 2030, compared to a reference period (2019-07-01 to 2020-06-30). [7]

Aerodynamic drag, from now on referred to as drag, is one of the main resisting forces of trucks driving at highway cruise speed. Drag consequently has a direct impact on fuel consumption, and in extension CO_2 emissions. Lower drag will also improve the viability of electric and hybrid trucks by increasing their range, and benefit the customers using combustion engine trucks by reducing fuel costs. To reduce drag, new trucks must be designed with aerodynamic performance in mind. Currently one of the most efficient ways to evaluate the aerodynamic performance is by computer simulation using computational fluid dynamics (CFD).

The Volvo brand trucks are developed at Volvo Group Trucks Technology, from now on referred to as Volvo. This includes CFD analysis for the development of aerodynamic features. In order for the results to be useful in the design process the results need to be comparable and accurate. The workflow also needs to be efficient, so that the best concept from a large number of possible candidates can be selected. In this project the comparability and accuracy of CFD simulation results will be investigated in three areas.

CFD simulations for vehicle aerodynamics are very computationally heavy. Because of this, simulations are run on parallel high performance computing (HPC) clusters. In some cases, the same steady simulation run on different computer clusters have been discovered to yield different results, although the simulation settings were identical. This is a problem that affects the comparability of results, and will therefore to be investigated. The use of HPC clusters also significantly affect meshing and simulation time. An investigation of parallel HPC effects can therefore give additional information to help decrease simulation time, and thereby increase workflow efficiency.

It has also been seen that small geometry changes on a model will cause the result to enter one of two bi-stable flow states in steady simulations, without a clear pattern. The difference in results is significant, so that the comparison between results is affected. This problem also affects accuracy, as it is not clear whether either of the solution modes exists in reality. Because of this, the bi-stability problem will also be investigated.

Lastly, nominal CAD geometry is generally used for simulations. In reality there is however a natural variation in geometry, for example deviations in the gap and flush of split-lines caused in the production of the product. This causes an uncertainty in the accuracy of the simulation results compared to the performance of real vehicles. The variation in simulation drag results caused by geometrical variation in production will therefore also be investigated. Information from this investigation may also prove helpful in determining areas where geometrical deviations have a large impact on drag, and where it therefore is important to limit such deviations.

1.1 Aim and Objective

The project is divided into three parts that will investigate the three problems presented above. The main aim of all parts of the project is to improve the comparability and accuracy of the simulation results. For the first part of the project, a secondary aim is to improve efficiency. For the third part the aim is also to find which split-line deviations have the most impact on drag, to provide information to help reduce drag of real trucks.

In order to reach these aims, the objective is to answer the following questions for each part of the project.

Part 1: HPC effects on Simulation Results

- Can the problem be recreated?

- Why and when does HPC effects impact simulation results?
- How can the problem be solved or avoided?

The questions to answer for the secondary aim of improving efficiency are:

- Which HPC parameters affect meshing and simulation time?
- How do HPC parameters affect meshing and simulation time?

Part 2: Bi-stable Flow States

- What is the difference between the modes and how are the results affected?
- What is the cause for the difference between the modes?
- Does any of the modes reflect the true physical behavior?
- What can be done to predict the mode or modes best reflecting reality in the simulations?

Part 3: Impact of Geometrical Variation on Drag

- How large is the variation in drag caused by variations in split-line gap and flush?
- How do the investigated areas affect drag, and which areas have the largest impact?
- How do gap and flush deviations affect drag, and which has the largest impact?

1.2 Limitations

Because of the project time frame and the limit on computational resources the project has been delimited. In general a limited number of trucks and a few selected side wind yaw angles are investigated. The CFD software used is Star-CCM+. Both Star-CCM+ version 9.06 and version 19.1 (14.02 using the old release numbering) are used in the project. The versions will from now on be referred to as Star-1 and Star-2 respectively.

Part 1: HPC effects on Simulation Results

- All simulations are steady.
- An optimal combination of HPC parameters for short simulation times is not pursued.
- Both Star-1 and Star-2 are investigated.

Part 2: Bi-stable Flow States

- No real world test data exist. Simulations are correlated with more accurate simulation methods.
- The investigations are primarily performed in Star-1.

Part 3: Impact of Geometrical Variation on Drag

- Only one truck is investigated.
- Complete areas are investigated; not individual split-lines.
- The geometrical deviations are implemented by rigid body translations of involved components.
- An optimal configuration for low drag is not pursued.
- All simulations are steady.
- All simulations are performed in Star-2.

1.3 Thesis Outline

This thesis is divided into an introductory chapter, a theory chapter, one chapter describing each part of the project, and finally a chapter with concluding remarks. The method, results and discussion of the parts of the project are treated separately in the chapter of each, to avoid confusion. The subject of the chapters are listed in short below.

Chapter 1 presents the problem and its background as well as the project objectives and limitations.

Chapter 2 provides a theoretical background to the problems to be solved and methods used. The areas covered are the aerodynamics of commercial road vehicles, CFD analysis and some related topics, and design of experiment methodology.

Chapter 3 describes the investigation of the effects of partitioning on simulation results.

Chapter 4 describes the investigation of the bi-stable flow states appearing in steady simulations.

Chapter 5 describes the investigation of the effects of split-line variation on the drag of the vehicle.

Chapter 6 summarizes and concludes the project, and gives some suggestions for future work.

2 Theory Overview

In this chapter the basics of road vehicle aerodynamics are covered. Some of the concepts in Computational Fluid Dynamics (CFD), commonly used for the analysis of vehicle aerodynamics, are also described. The last topic of the chapter is Design of Experiment (DOE), the statistical methodology for planning, performing and analyzing experimental studies.

2.1 Aerodynamics of road vehicles

Aerodynamics of road vehicles is the study of the external flow of air around the body of a vehicle traveling on the ground. This field has many applications, such as the aerodynamic forces on the vehicle, aeroacoustics (wind noise), soiling, and underhood cooling. The present work is mainly concerned with the air resistance, referred to as aerodynamic drag, of heavy trucks.

2.1.1 Aerodynamic drag

Aerodynamic drag, drag in short, is the resisting force that the flow of air exerts on the vehicle in its traveling direction. The drag F_D is given by the expression

$$F_D = \frac{1}{2}\rho v^2 A C_D, \tag{2.1}$$

where ρ is the air density, V is the speed of the vehicle relative to the flow, A is the projected frontal area of the vehicle, and C_D is the dimensionless drag coefficient. Density and speed is out of control of the vehicle designer, and the area is mostly determined by functional requirements such as maximizing the transported volume. The designer is therefore generally restricted to lower drag by reducing C_D. [1, 11, 35]

The drag coefficient is sometimes expressed in drag counts (DC), especially differences in C_D between different concepts. One drag count is defined as

$$1 \text{ DC} := 0.001 C_D. \tag{2.2}$$

The total drag is the sum of two contributions, friction drag and pressure drag. Both friction drag and pressure drag depends on the shape of the vehicle. This is because the geometry affects the friction and pressure distribution by its influence on the flow field. The friction distribution is mainly decided by the location of the transition point from laminar to turbulent boundary layer. Surface roughness and geometrical features cause earlier transition, resulting in increased friction drag as a larger part of the boundary layer is turbulent. The pressure distribution is mostly decided by the separation of the flow. A larger separation gives increased pressure drag, as the pressure in the separated wake is lower than the stagnation pressure at the front of the vehicle. Vortices generated by the body may also affect the pressure distribution and increase drag. [1, 11, 35]

Separation

Separation is a very important phenomenon in vehicle aerodynamics. The flow is said to separate when it can no longer stay attached to the vehicle surface. This occurs when the air flows against increasing pressure in a boundary layer. This is commonly known as an adverse pressure gradient (APG), because the flow is retarded by the APG. If the APG is strong enough, the air does not have enough momentum to continue its path along the surface. Instead the air is forced to leave the surface, and the flow separates. Between the separated streamlines and the surface a low velocity recirculation region is formed. Between the fast flowing free stream and the low velocity region there is a separation shear layer with large velocity gradients. If the pressure gradient is negative, called favorable pressure gradient, separation is suppressed instead. A turbulent boundary layer is more resistant to separation than a laminar, since the increased mixing due to turbulent motion will transfer more momentum from the free stream. This decreases the retardation caused by the APG, and therefore delays the separation or prevents it entirely. The flow is said to reattach if the flow returns to the surface further downstream. This can

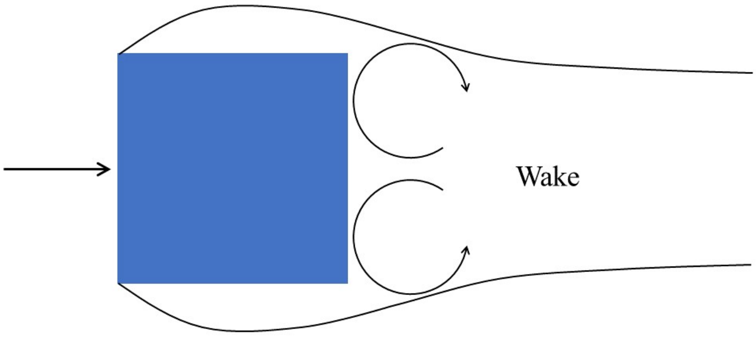


Figure 2.1: *Schematic illustration of the flow separation for a square body.*

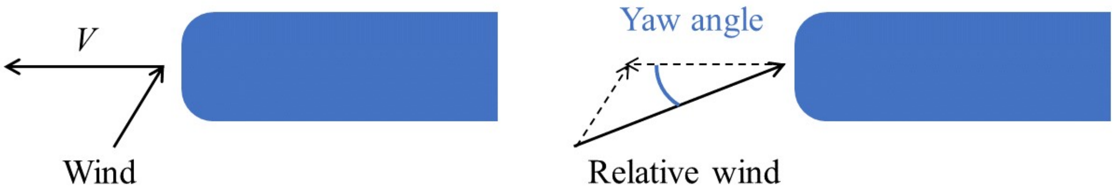


Figure 2.2: *Schematic illustration of side wind yaw angle.*

occur if the pressure condition is more favorable, or if new momentum is added from the free stream. [1, 11, 35, 4]

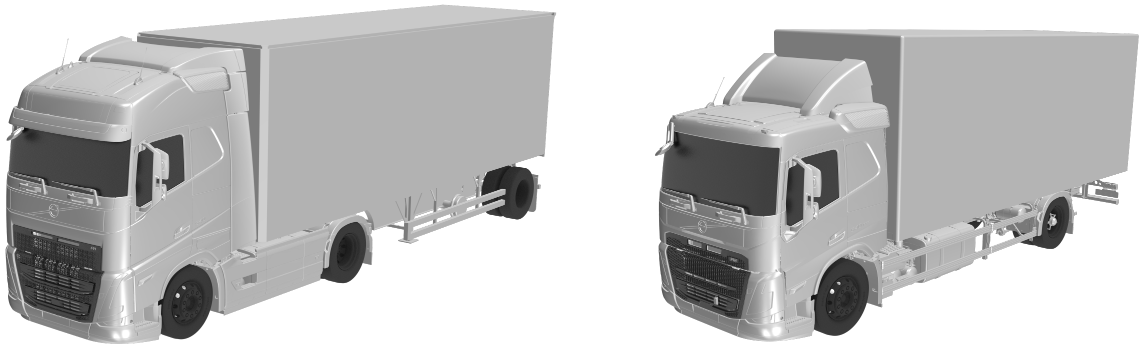
For external aerodynamics the pressure reaches its maximum in the stagnation region at the front of the body, and decreases downstream as the air is deflected and accelerated around the body. When the flow follows around the body the boundary layer slows down and the pressure rises again. If the body shape is not very streamlined, the resulting APG will be too strong and the flow will separate. The separated low velocity region extending behind the body is called wake. There may also be local separations that reattach to the body further downstream. A local separation may also be caused by the increased pressure in the stagnation region in front of a geometric protrusion. Because of the separation, the pressure in the wake cannot return to the value at the front of the body, which as mentioned above is the main contribution to pressure drag. The pressure difference between the free stream and a separated region may also create vortices, which lead to further losses. Blunt bodies, such as square blocks, are far from streamline shaped, and will therefore cause massive separation with a large low pressure wake, as illustrated in Figure 2.1. [1, 11, 35, 4]

Vortices

A vortex is a flow structure where the flow rotates around an axis line, and is an important feature of vehicle aerodynamics. Especially strong vortices are created in the wake of a blunt body. These are roughly perpendicular to the flow, and are called standing vortices. Geometric protrusions may cause vortices roughly aligned with the flow, called trailing vortices. Vortices created by separation become unsteady for higher flow velocities, due to vortex shedding. This makes blunt body flow at higher velocity highly unsteady in nature. [1, 11, 35, 4]

2.1.2 Ambient wind - yaw

In real driving conditions, there is often some ambient wind present. In that case the resulting relative velocity of the air flow will come at an angle to the vehicle driving direction, as is illustrated in Figure 2.2. This angle is called the yaw angle. The drag of a vehicle, especially heavy trucks, often becomes larger as the yaw angle is increased. This is because the flow separates earlier at the front, and because the less aerodynamically shaped underbody and trailer are more exposed to the flow. It is therefore important to take drag performance under yawed conditions into consideration when designing a vehicle. This is often done by calculating a yaw weighted average (YWA) for C_D evaluated at some representative yaw angles. [1, 11, 14]



(a) *FH high cab tractor-trailer truck.* **(b)** *FM medium cab rigid truck.*

Figure 2.3: *Truck types. The the FH truck is the baseline or reference for Part 2 and Part 3 of this project. Both trucks are investigated in Part 1. Note that a longer, three-wheel trailer is used for the FH in the investigations.*

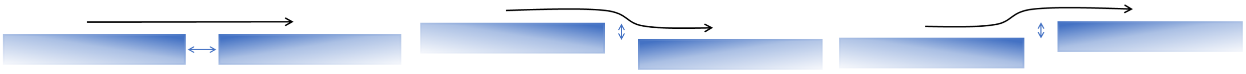
2.1.3 Truck aerodynamics

The body of a truck consists of a chassis, cab and trailer. European trucks are usually of cab-over-engine (COE) type, which means that the front is basically flat. Trucks can be of tractor-trailer or rigid type, see Figure 2.3. A tractor-trailer type has a pulling unit, the tractor, and a semi-trailer mounted on a turntable on the tractor chassis. The gap between the cab and the trailer is quite large to allow for rotation of the trailer on the turntable. On a rigid type truck the trailer is mounted directly on the truck chassis, and does not move independently. Because of this the gap between cab and trailer is smaller. On both types the trailer is typically taller than the cab, if it is a box trailer. [1, 11]

Modern trucks are fitted with some aerodynamic devices in order to reduce drag. The most important is the roof and side deflectors, mounted on the cab roof and rear. The purpose of the roof deflector is to lead the flow smoothly over the cab roof past the leading edge of the trailer. Otherwise the flow can impact the exposed trailer front, which will cause a separation there. This is often the single most effective measure to reduce drag of trucks. The side deflectors come into play when the air stream is yawed, as it hinders the flow from entering the gap between the cab and trailer. [1, 11]

Although the aerodynamic development of trucks has come a long way, a truck is still an inherently blunt body. The nature of truck aerodynamics is therefore highly unsteady in nature. Several studies on simplified vehicle shapes have been performed, both experimental and numerical, to learn more about the characteristics of road vehicle aerodynamics [9, 21, 15, 20, 22]. Here different bi-stable states have been observed, where body shape and yaw angle are important factors. In one study on a square body shape with rounded front edges, the wake was found to switch between two bi-stable states, at a long timescale $T = 10^3 H/U_0$ for a body height H and free-stream velocity U_0 [9]. These studies have however only investigated the transient wakes of simplified vehicle shapes. Bi-stability in local separations caused by geometrical detail have not been studied. The experimental studies have used particle image velocimetry (PIV) to aid the identification of the flow states. In PIV, small particles are released and photographed to visualize the flow field. The velocity field can then be calculated on a computer, from the images. The numerical studies have only been transient simulations, so no precedence has been found for bi-stability in steady simulations.

Due to functional requirement, requirements for production and assembly, and other non-aerodynamic considerations there is a number of geometrical features on real vehicles. These both affect the flow field shape and contribute to flow unsteadiness. The vorticity and turbulence generated by the blunt shape and geometrical features makes truck aerodynamics highly unsteady. One typical feature is split-lines between body panels and components. Split-lines introduce roughness that trigger separation or turbulent transition. [1, 11] The width of the split-line is called gap, and the levelness of the two components is called flush. This is illustrated in Figure 2.4. Furthermore, if the downstream component is offset under the upstream component (like a



(a) *Gap: split-line width.* **(b)** *Under-flush: downstep offset.* **(c)** *Over-flush: upstep offset.*

Figure 2.4: *Schematic illustration of gap and flush, as seen from the side.*

backwards facing step) it is here termed as under-flush, whereas if it is offset over the upstream component (like a forwards facing step) it is termed as over-flush. Wider gaps and greater flush offset generally increase drag by disturbing the flow more.

2.2 Computational Fluid Dynamics

One way to analyze vehicle aerodynamics is to simulate the flow field using CFD analysis. In CFD the computational domain is divided into a mesh, and the governing equations are discretized into a system of algebraic equations which are solved iteratively on a computer. Because the large scale of vehicle aerodynamics simulations it is extremely computationally intensive to solve the turbulent flow field directly. Instead turbulence modeling is used to decrease the computational cost. Even with turbulence modeling vehicle aerodynamics is expensive to simulate, so HPC clusters using multiple processors in parallel for computations, are used to run simulations.

The general workflow of CFD analysis consists of the three consecutive steps, preprocessing, running the simulation and postprocessing. In the preprocessing step, the computational domain is set up and meshed, boundary conditions are defined and the physics to be included in the simulation is selected. In the simulation step, the computer calculates and solves the discretized system of equations. In the postprocessing step, the results are prepared for analysis, for example calculating the quantities of interest, such as drag, and visualizing the flow field. The post-processing can also be carried one step further, for example by using the adjoint method to analyze the sensitivity of the vehicle geometry with respect to drag. In the industry commercial softwares are often used, such as Star-CCM+ by Siemens, which contains tools for all three steps above.

2.2.1 Governing equations of fluid dynamics

In road vehicle aerodynamics compressibility effects are neglected due to the low velocity, and the fluid is air, which is regarded as a Newtonian fluid. The governing equations for incompressible, Newtonian flow, expressed on tensor notation, are

$$\rho \frac{\partial v_i}{\partial x_i} = 0 \tag{2.3}$$

$$\rho \frac{\partial v_i}{\partial t} + \rho \frac{\partial v_i v_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \mu \frac{\partial^2 v_i}{\partial x_j \partial x_j}, \tag{2.4}$$

where (2.3) is the continuity equation which describes the transport of fluid mass and (2.4) are the momentum equations, which describe the transport of fluid momentum. Together these are called the Navier-Stokes (NS) equations. Here x_i , with $i = 1, 2, 3$, are the cardinal directions, v_i is the velocity, p is the pressure, and μ is the dynamic viscosity of the fluid. Sometimes the energy equation is included, but usually for external road vehicle aerodynamics it is excluded.

2.2.2 Turbulence modeling

The governing equations can be considered to provide the exact solution to a given flow case. However, in turbulent flow, the enormous computational effort required to resolve the physics correctly makes an exact solution for engineering applications unattainable at this point. In order to decrease the computational cost some of the physics is modeled instead. One approach is to model all of the turbulent fluctuations, solving for a time-averaged, steady solution. Another approach is to model only the smaller turbulent fluctuations, and resolving the larger turbulent motions. [5]

Time-averaged models - RANS

In the first approach, where all turbulence is modeled, the flow variables are split using Reynolds decomposition, $\phi = \overline{\phi} + \phi'$, where $\overline{\phi}$ is the time-averaged part of some variable ϕ, and ϕ' is the fluctuating part. Applying the decomposition to the governing equations and then time-averaging yields the so called Reynolds Averaged Navier-Stokes (RANS) equations [5]

$$\frac{\partial \overline{v}_i}{\partial x_i} = 0 \tag{2.5}$$

$$\rho \frac{\partial \overline{v}_i \overline{v}_j}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\mu \frac{\partial \overline{v}_i}{\partial x_j} + \tau_{ij}\right). \tag{2.6}$$

As RANS models are commonly used for the steady, time-averaged flow solution, the transient term is left out here. The new term $\tau_{ij} = -\rho \overline{v_i' v_j'}$ is the turbulent stress tensor that appears from the derivation. These stresses are new unknowns without equations. Although new equations can be introduced for the stresses, they are most commonly modeled using the Boussinesq assumption

$$-\overline{v_i' v_j'} = 2\nu_t \overline{s}_{ij} - \frac{2}{3}\delta_{ij} k \tag{2.7}$$

where ν_t is the so called eddy viscosity, $\overline{s}_{ij} = \frac{1}{2}\left(\frac{\partial \overline{v}_i}{\partial x_j} + \frac{\partial \overline{v}_j}{\partial x_i}\right)$ is the mean strain rate tensor and $k = \frac{1}{2}\overline{v_i' v_i'}$ is the turbulent kinetic energy (TKE) usually calculated from a turbulent velocity scale $\mathcal{U}$ and turbulent length scale ℓ or turbulent velocity scale and turbulent time scale $\mathcal{T}$

$$\nu_t \propto \mathcal{U}\ell = \mathcal{U}^2 \mathcal{T}. \tag{2.8}$$

Some commonly used RANS models are different variants of the $k - \varepsilon$ model and the $k - \omega$ model. Both are so called two-equation models, which solve for two quantities used to calculate the turbulent scales. [5]

The standard k-ε model (SKE) uses the TKE k and the turbulent dissipation rate (TDR) ε to calculate the turbulent scales for the turbulent viscosity

$$\mathcal{U} = k^{1/2}, \quad \mathcal{T} = \frac{k}{\varepsilon} \quad \Rightarrow \quad \nu_t = c_\mu \frac{k^2}{\varepsilon}. \tag{2.9}$$

The equations for the TKE and the TDR are

$$\overline{v}_j \frac{\partial k}{\partial x_j} = P_k - \varepsilon + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] \tag{2.10}$$

$$\overline{v}_j \frac{\partial \varepsilon}{\partial x_j} = \frac{\varepsilon}{k}\left[C_{\varepsilon 1} P_k - C_{\varepsilon 2}\varepsilon\right] + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right], \tag{2.11}$$

where $P_k = -\overline{v_i' v_j'}\frac{\partial \overline{v}_i}{\partial x_j}$ is the production of TKE and ε, the TDR, is the destruction term in the k-equation. The model constants are determined experimentally, or may be tuned for the application. The standard values are [5]

$$c_\mu = 0.09, \quad C_{\varepsilon 1} = 1.44, \quad C_{\varepsilon 2} = 1.92, \quad \sigma_k = 1, \quad \sigma_\varepsilon = 1.3. \tag{2.12}$$

The standard $k-\varepsilon$ model is widely used in the industry, as it is very stable and reasonably accurate for many flows. However, due to the nature of the equations, special treatment is needed in the near wall region to improve the prediction of the inner boundary layer. It has furthermore some disadvantages that may impact accuracy for external aerodynamics applications, namely the overprediction of turbulent activity in stagnation regions and difficulties in predicting separation on curved walls. [5, 34]

These problems stem in part from the fact that the Boussinesq assumption in combination with the definition of the turbulent viscosity may violate the two realizability constraints

$$\overline{v_i'^2} \geq 0, \tag{2.13}$$

$$\frac{\overline{v_i'v_j'}}{\left(\overline{v_i'^2 v_j'^2}\right)^2} \leq 1 (\text{no summation over } i, j), \tag{2.14}$$

which can lead to unphysical results. [5]

The issue with prediction in stagnation regions is caused by an overproduction of TKE, which results in negative normal stresses, see (2.13). That is for the normal stress $\overline{v_1'^2}$

$$\overline{v_1'^2} = \frac{2}{3}k - 2\nu_t \overline{s}_{11} = \frac{2}{3}k - 2c_\mu \frac{k^2}{\varepsilon}\overline{s}_{11}. \tag{2.15}$$

A way to remediate this is to implement a realizability constraint on the turbulent time scale

$$T = \min\left(\frac{k}{\varepsilon}, \frac{C_T}{c_\mu |\overline{s}|}\right), \tag{2.16}$$

where C_T is the so called Durbin scale limiter (DSL), with a standard value of $C_T = 0.6$, and $|\overline{s}| = (2\overline{s}_{ij}\overline{s}_{ij})^{1/2}$. For road vehicle aerodynamics $C_T = 1.2$ is recommended [27]. The Durbin realizability condition will increase the dissipation in (2.11), which limits the production of TKE in (2.10). [6]

The realizable k-ε model (RKE) is based on SKE, but uses a variable coefficient for c_μ to satisfy both realizability constraints of (2.13) and (2.14). It also uses a new equation for the TDR, derived from the transport equation for vorticity fluctuations, instead of repurposing the k-equation. The new definition of the coefficient c_μ is given by

$$c_\mu = \frac{1}{A_0 + A_s U^* k/\varepsilon}, \tag{2.17}$$

$$A_0 = 4, \quad A_s = 6^{1/2}\cos\left[\frac{1}{3}\cos^{-1}\left(6^{1/2}\frac{\overline{s}_{ij}\overline{s}_{jk}\overline{s}_{ki}}{(2|\overline{s}|)^3}\right)\right], \quad U^* = (\overline{s}_{ij}\overline{s}_{ij} + \overline{\omega}_{ij}\overline{\omega}_{ij})^{1/2} \tag{2.18}$$

where $\overline{\omega}_{ij} = \frac{1}{2}\left(\frac{\partial \overline{v}_i}{\partial x_j} - \frac{\partial \overline{v}_j}{\partial x_i}\right)$ is the mean vorticity rate tensor. [26]

The model equations are

$$\frac{\partial \overline{v}_j k}{\partial x_j} = P_k - \varepsilon + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] \tag{2.19}$$

$$\frac{\partial \overline{v}_j \varepsilon}{\partial x_j} = C_1 |\overline{s}| \varepsilon - C_2 \frac{\varepsilon^2}{k + (\nu\varepsilon)^{1/2}} + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right], \tag{2.20}$$

with the standard coefficients [26]

$$C_{\varepsilon 1} = \max\left(0.43, \frac{\eta}{4+\eta}\right), \quad C_{\varepsilon 2} = 1.9, \quad \sigma_k = 1, \quad \sigma_\varepsilon = 1.2, \tag{2.21}$$

$$\eta = |\overline{s}|k/\varepsilon. \tag{2.22}$$

The standard k-ω model (SKO) uses the TKE and the specific dissipation rate (SDR) $\omega \propto \varepsilon/k$ to calculate the turbulent scales and the turbulent viscosity

$$\mathcal{U} = k^{1/2}, \quad \ell = \frac{k^{1/2}}{\omega} \quad \Rightarrow \quad \nu_t = \frac{k}{\omega}. \tag{2.23}$$

The model has the advantage of not needing any special wall treatment, as opposed to the $k-\varepsilon$ models, and it also performs better in APG flows than SKE. It still overpredicts the shear stress however, and has the disadvantage of being dependent of the conditions in the free-stream, resulting in sensitivity to boundary conditions. [5]

The Shear Stress Transport k-ω model (SST) improves the shear stress prediction in APG flow compared to the SKO model by limiting the turbulent viscosity. The model furthermore combines the SKO in the inner boundary layer to remove the need for wall treatment with the SKE model in the outer boundary layer and free-stream to avoid the free-stream sensitivity. The new turbulent viscosity is given by [5]

$$\nu_t = \min\left(\frac{k}{\omega}, \frac{a_1 k}{|\overline{s}|F_2}\right), \tag{2.24}$$

$$\nu_{t,\mathrm{DSL}} = \min\left(\nu_t, \frac{C_T k}{3^{1/2}|\overline{s}|}\right), \tag{2.25}$$

$$F_2 = \tanh(\xi^2), \xi = \max\left(\frac{2k^{1/2}}{\beta^* \omega d}, \frac{500\nu}{\omega d^2}\right), \tag{2.26}$$

where the Durbin realizability condition is applied in (2.25) to the original SST turbulent viscosity in (2.24), and d is the wall distance. The model equations are

$$\frac{\partial \overline{v}_j k}{\partial x_j} = P_k - \beta^* \omega k + \frac{\partial}{\partial x_j}\left[\left(\nu + \sigma_k \nu_t\right)\frac{\partial k}{\partial x_j}\right] \tag{2.27}$$

$$\frac{\partial \overline{v}_j \omega}{\partial x_j} = \frac{\alpha}{\nu_t} P_k - \beta\omega^2 + (1-F_1)\sigma_{\omega,\omega}\frac{2}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} + \frac{\partial}{\partial x_j}\left[\left(\nu + \sigma_\omega \nu_t\right)\frac{\partial \omega}{\partial x_j}\right], \tag{2.28}$$

where the blending function $F_1 = 0$ by the wall and $F_1 = 1$ in the outer boundary layer, is used to smoothly switch between the $k-\varepsilon$ and $k-\omega$ behavior by

$$\varphi = F_1 \varphi_\omega + (1-F_1)\varphi_\varepsilon \tag{2.29}$$

for the model coefficients $\varphi = \alpha, \beta, \sigma_k, \sigma_\omega$. The standard coefficient values are [32, 23]

$$a_1 = 0.31, \quad \beta^* = 0.09, \tag{2.30}$$

$$\alpha_\omega = 5/9, \quad \beta_\omega = 3/40, \quad \sigma_{k,\omega} = 0.85, \quad \sigma_{\omega,\omega} = 0.5, \tag{2.31}$$

$$\alpha_\varepsilon = 0.44, \quad \beta_\varepsilon = 0.0828, \quad \sigma_{k,\varepsilon} = 1, \quad \sigma_{\omega,\varepsilon} = 0.856. \tag{2.32}$$

However, for vehicle aerodynamics it is recommend to use $a_1 = 1$ [27].

Scale resolving models

The second approach resolves the larger turbulent scales, whereas the turbulent motions smaller than the mesh size are modeled. The flow variables are therefore divided into $\phi = \widetilde{\phi} + \phi''$, where $\widetilde{\phi}$ is the space filtered part corresponding to the larger scales, and ϕ'' is the sub-grid part corresponding to the small scales. Applying this decomposition and space filtering to the governing equations yields

$$\rho \frac{\partial \widetilde{v}_i}{\partial x_i} = 0 \tag{2.33}$$

$$\rho \frac{\partial \widetilde{v}_i}{\partial t} + \rho \frac{\partial \widetilde{v}_i \widetilde{v}_j}{\partial x_j} = -\frac{\partial \widetilde{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\mu \frac{\partial \widetilde{v}_i}{\partial x_j} + \tau_{ij}\right), \tag{2.34}$$

where $\tau_{ij} = \rho(\widetilde{v_i v_j} - \widetilde{v}_i \widetilde{v}_j)$ is the subgrid stress tensor. Note that scale resolving models are transient, hence the transient term in (2.34). [5]

Large Eddy Simulation (LES) resolves the large scales everywhere in the domain, which requires a very fine mesh resolution near the wall to predict the boundary layer accurately. Because of this LES is still too expensive for most engineering applications. [32, 5]

Detached Eddy Simulation (DES) combines unsteady RANS (URANS) in the boundary layer to keep an affordable mesh resolution in near wall regions, and LES in the rest of the domain to resolve the large eddies there. DES uses the LES equations, (2.33) and (2.34), which can be seen to be mathematically identical to the URANS equations ((2.5) and (2.6) with the transient term included). The turbulent stresses can be modeled for example using a modified version of the SST model. The modification consists of replacing ω with ωF_{DES} in the SST k-equation, (2.27), with

$$F_{DES} = \max\left(\frac{L_t}{C_{DES}\Delta}, 1\right), L_t = \frac{k^{1/2}}{\beta^* \omega}, \tag{2.35}$$

where Δ is the distance to the farthest neighbor, and the coefficient $C_{DES} = 0.61$. The purpose of the F_{DES} parameter is to switch between RANS and LES behaviour; when $F_{DES} > 1$ the TKE and in extension turbulent viscosity and modeled stresses decreases, which leads to less damping of (2.34), so that the flow may go unsteady and the turbulent eddies are resolved. [5]

Since less modeling is used in scale resolving models, the accuracy is better than for RANS models, especially in bluff body flow, such as vehicle aerodynamics, where large turbulent structures play an important role. However, since the simulation is transient it is more expensive than steady RANS. [5, 34]

2.2.3 Numerical Solution

In order to solve a fluid flow problem numerically, the governing equations need to be discretized into a system of algebraic equations that can be solved on a computer. A widely used method for discretization is the finite volume method (FVM), which is based on a control volume approach to the cells of the mesh. The governing equations are therefore first integrated, after which Gauss divergence theorem is applied. The integral equations are then discretized. [34]

For the solution of the discretized equation system a linear solver is needed. Due to the size of the equation system, and the non-linear nature of the governing equations, an iterative direct solver such as Gauss elimination cannot be used. Instead an iterative solver is employed, often combined with a multigrid technique to improve the convergence rate of the solution.[34]

Finite Volume Discretization

The discretization of the integral equations comes in three parts. The first step is the approximation of the integrals as $\int \Phi dV \approx \Phi \Delta V$, $\int \Phi n_i dA \approx \sum_f (\Phi n_i \Delta A)_f$ for some function Φ, where ΔV is the cell volume, $\sum_f$ is the sum over all cell faces, n_i is the outwards facing normal and ΔA is the face area. The steady transport equation for a flow variable ϕ then becomes

$$\sum_f (\rho v_i \phi n_i \Delta A)_f = \sum_f \left(\Gamma \frac{\partial \phi}{\partial x_i} n_i \Delta A \right)_f + S_\phi \Delta V, \tag{2.36}$$

where the term on the left hand side is the convective flux, and on the right hand side is the diffusive flux, with diffusion coefficient Γ, and the source term for some source S_ϕ. The next parts of the discretization is the approximation of the convective and diffusive fluxes. [34]

The next steps are to discretize the convective and diffusive flux. The convective flux is discretized using one of several discretization schemes, for example the second order upwind (SOU) scheme. In SOU the face value of ϕ_f is calculated using linear interpolation from the values of the adjacent cell center points, with respect to flow direction at the face. That gives the convective flux at the face

$$(\dot{m}_i \phi)_f = \begin{cases} \dot{m}_{i,f} \phi_{f,P} & \text{if } v_{i,f} n_{i,f} \geq 0 \\ \dot{m}_{i,f} \phi_{f,N} & \text{if } v_{i,f} n_{i,f} < 0, \end{cases} \tag{2.37}$$

$$\phi_{f,P} = \phi_P + (x_{i,f} - x_{i,P}) \left(\frac{\Delta \phi}{\Delta x_i} \right)_P, \quad \phi_{f,N} = \phi_N + (x_{i,f} - x_{i,N}) \left(\frac{\Delta \phi}{\Delta x_i} \right)_N, \tag{2.38}$$

where P denotes the current cell center and N the cell center of the neighboring cell opposite the face f, and $\dot{m}_i = \rho v_i$ is the mass flux. [34]

The diffusive flux is discretized using central differencing for the face gradient, giving [34]

$$\left(\Gamma \frac{\partial \phi}{\partial x_i} n_i \Delta A \right)_f = \Gamma_f \frac{\phi_N - \phi_P}{n_{j,f}(x_{j,N} - x_{j,P})} (n_i n_i \Delta A)_f. \tag{2.39}$$

After inserting the discretized fluxes into (2.36) and grouping the terms by the cell values at the current cell center, ϕ_P, and by the values at all neighboring cells, ϕ_N, the final algebraic equation becomes

$$a_P \phi_P + \sum_N a_N \phi_N = S_{\phi,P}, \tag{2.40}$$

where a_P and a_N are the algebraic coefficients resulting from the discretization. By extending (2.40) to all cells of the computational domain, the complete system of equations can be written as

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \tag{2.41}$$

where $\boldsymbol{A}$ is the matrix containing the coefficients a_P and a_N, $\boldsymbol{x}$ is the vector of unknowns, and $\boldsymbol{b}$ is the vector of sources and boundary conditions. Note that $\boldsymbol{A}$ is sparse since only the nearby cells affect the current cell [34].

Solving the Algebraic equations: Iterative Solvers

To get the solution to the flow problem, the equation system in (2.41) has to be solved. A direct method, such as Gauss elimination, would require a number of operations of the order of n^3 to solve an $n \times n$ system of equations, and the storage of all n^2 coefficients of $\boldsymbol{A}$. Because the number of unknowns in a typical CFD simulation is very large, this makes a direct method too computationally expensive. Iterative methods on the other hand are more cost efficient, requiring only operations in the order of n per iteration to solve, and storage capacity in the order of n, as only the non-zero elements of the sparse $\boldsymbol{A}$ matrix are needed. The non-linear nature of the equations is a second reason for using an iterative solver, as the coefficient matrix $\boldsymbol{A}$ needs to be updated repeatedly. [34, 8]

The concept of an iterative method is to calculate a better solution $\boldsymbol{x}^{(k+1)} = \sum_i^n x_i^{(k+1)}$ from the values $\boldsymbol{x}^{(k)}$ of the previous iteration k, until the final solution is found. If the discretized problem is well behaved, the process will converge to the true solution, $\boldsymbol{x}^{(k+1)} \to \boldsymbol{x}$ of (2.41). [34]

The Jacobi method is a simple iterative method in which the solution for the current iteration is calculated from

$$a_{ii} x_i^{(k+1)} = b_i - \sum_{j \neq i}^{n} a_{ij} x_j^{(k)}, \tag{2.42}$$

with only the values $x_j^{(k)}$ from the previous iteration. [34]

The Gauss-Seidel method is similar to the Jacobi method, but instead of only using the values $x_j^{(k)}$ from the previous iteration, the most recent values

$$x_j^{(\kappa)} = \begin{cases} x_j^{(k+1)} & \text{if} j < i \\ x_j^{(k)} & \text{if} j > i \end{cases} \tag{2.43}$$

are used. This increases the convergence rate, and therefore Gauss-Seidel is commonly used in CFD. [34]

Incomplete Lower Upper method (ILU) uses incomplete LU factorization to calculate the current iteration solution from

$$\boldsymbol{LU}\big(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\big) = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(k)}, \tag{2.44}$$

$$\boldsymbol{LU} = \boldsymbol{A} + \boldsymbol{R}, \tag{2.45}$$

where $\boldsymbol{L}$ and $\boldsymbol{U}$ are the incomplete upper and lower triangular matrices of $\boldsymbol{A}$, calculated so that the difference $\boldsymbol{R}$ is small. In Star-CCM+ the MILU(0) method is used, meaning that all non-zero elements of $\boldsymbol{L}$ and $\boldsymbol{U}$ that are zero in $\boldsymbol{A}$ are dropped, and that the dropped terms are compensated for in the diagonals of $\boldsymbol{L}$ and $\boldsymbol{U}$. [32, 24]

Increasing Convergence: Multigrid Techniques

As the iterative solution progresses there will be a remaining, iterative, error at each iteration. This error is said to have different wavelengths; it is made up of local errors and errors spanning larger distances. One drawback of the iterative methods above is that they are not so effective at reducing the longer wavelength errors. This causes a strong decline in convergence rate as the mesh resolution is increased. [34]

This can be amended by using Multigrid Techniques, which are based on the fact that the relative wavelength of the error is shorter on a coarse mesh, and therefore can be reduced more effectively. A common version is

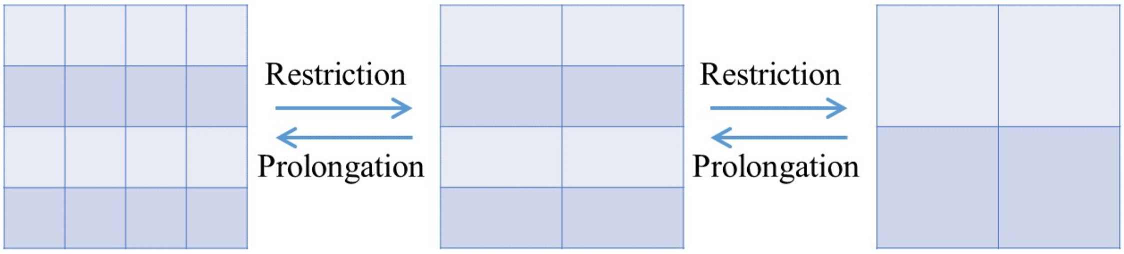


Figure 2.5: *Schematic illustration of the multigrid restriction and prolongation concept. In the restriction cells are combined to form a coarser mesh; in the prolongation the coarse correction are returned to the finer mesh. In AMG, these steps are disconnected from the mesh to decrease problems arising in complex geometries, and are instead based on the coefficients of the* $\boldsymbol{A}$ *matrix.*

the Algebraic Multigrid technique (AMG), in which the coarse level equation system $\boldsymbol{A}^c$ is based on arithmetic combinations of the fine level coefficients of $\boldsymbol{A}$, rather than on the geometrical mesh. [32] On the coarse levels a correction equation system

$$\boldsymbol{A}^c\boldsymbol{e}^c = \boldsymbol{r}^c, \tag{2.46}$$

is solved instead of (2.41), where $\boldsymbol{e} = \boldsymbol{x} - \boldsymbol{x}^{(k)}$ is the iterative error, $\boldsymbol{r} = \boldsymbol{A}\boldsymbol{x}^{(k)}\boldsymbol{b}$ is the residual for the current iteration and c denotes the coarse level. [34]

The multigrid method is built on a basic cycle that is nested and repeated into a complete multigrid cycle. The basic cycle consist of the steps [34, 8]

1. Pre-relaxation: application of iterative solver to finer level to smooth local errors
2. Restriction: transferring the equation system $\boldsymbol{A}$ and the residuals $\boldsymbol{r}$ to the coarser level
3. Coarse relaxation: application of iterative solver to the correction equation system (2.46)
4. Nested cycles
5. Prolongation: transferring the correction $\boldsymbol{e}^{c,(k)}$ back to the finer level
6. Post-relaxation: smoothing out any local errors introduced in the cycling

The concept of restriction and prolongation is illustrated in Figure 2.5. In the prolongation step the fine level correction is calculated by interpolation and used to correct the pre-cycle solution by [34]

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \boldsymbol{e}^{(k)} \tag{2.47}$$

for the finest level, or for coarse levels

$$\boldsymbol{e}^c_{\text{corrected}} = \boldsymbol{e}^c + \boldsymbol{e}^{cc}_{\text{prolonged}}. \tag{2.48}$$

Either of the solvers in subsection 2.2.3 can be used for the relaxation and solving in the restriction steps. The ILU method, although more expensive, has the advantage of being more efficient. This is because it is more robust, so more equations can be agglomerated when generating $\boldsymbol{A}^c$, resulting in more efficient cycles. The ILU method in Star-CCM+ for 3D simulations can handle an agglomeration group size of eight, whereas the Gauss-Seidel method is limited to four. The ILU method also scales well for parallel computations. [32]

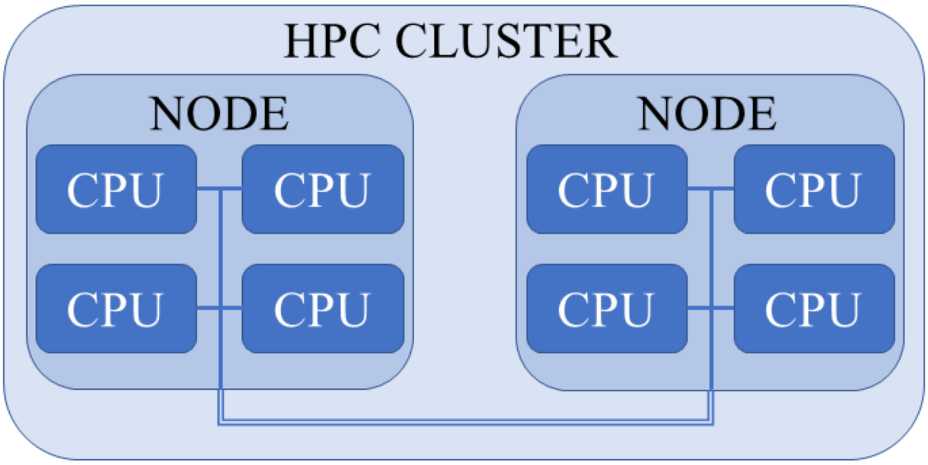


Figure 2.6: *Schematic illustration of an HPC cluster layout with two nodes and a total of eight processors (CPUs).*

2.2.4 Parallelization for High Performance Computing

Today the most cost effective method for High Performance Computing is using a network of parallel processors. Such a network is called a compute cluster, and the processors are grouped in nodes, as illustrated in Figure 2.6. The cluster can either be of shared memory type, in which processors communicate via the shared memory, or of distributed memory type, in which processors have individual local memories and communicate by sending and receiving messages. These messages follow a standard called Message Passing Interface (MPI). Only the distributed memory cluster is considered here. The processors in a distributed network are grouped into nodes, and each node is connected by a network forming the cluster. Communication within a node usually has lower latency, leading to faster communication, than between nodes. On the other hand, in some cases performance can increase by not assigning processes to all processors on a node, leaving one or more free for communication and other tasks. [10]

The motivation for HPC lies in the high demands for computational power that CFD simulations for engineering applications pose. Accurate simulations of realistic cases require larger meshes and more complex models to resolve the physics to a satisfactorily level. This both increases the simulation overturn time and simulation size. In order to keep simulation time within an acceptable limit and satisfy memory requirements a HPC cluster can be used to solve the problem divided into smaller parts.[2, 8]

In CFD parallel computing is implemented by dividing the computational domain into subdomains using domain decomposition, as illustrated in Figure 2.7, and assigning the corresponding partitioned problem to separate processors of the computer cluster. Because cells on the interfaces between sub-domains need information about the values of cells across the interface, communication between processors is required. Parallel efficiency is increased by a partitioning that minimizes communication requirements between processors and balances the computational load on each processor as far as possible. [3, 8] In Star-CCM+ the partitioning is performed by a cell connectivity graph. Load balancing is done by partitioning the graph so that the number of cells and cell faces per partition are balanced, and communication is minimized by minimizing the number of cut edges in the graph. [32]

As the algebraic solver is applied per partition in Star-CCM+, using the AMG solver will result in differences in the restriction and prolongation steps depending on the partitioning of the simulation, as illustrated in Figure 2.8. The results after one AMG cycle will therefore not be independent of the number of processors used. If the complete multigrid process for one linear solve is not run until convergence this can lead to small errors. In most cases, this should not be enough to affect the results of a well converged steady simulation; the results should be independent of the partitioning, although the path to the solution may have been different.

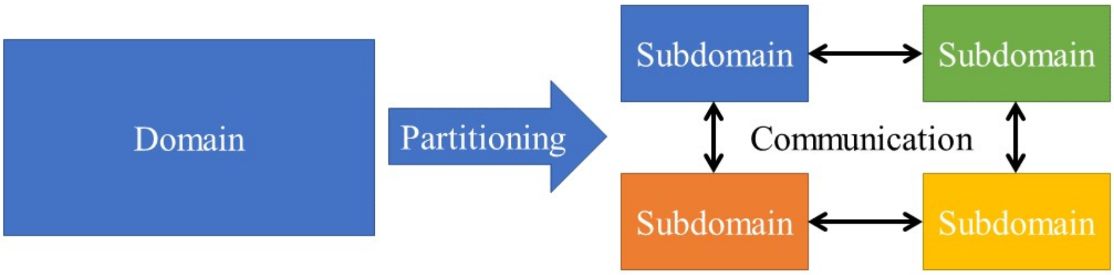


Figure 2.7: *Schematic illustration of domain decomposition. Each subdomain is solved on a separate processor, so information about interface cells must be communicated between processors with adjacent subdomains to propagate the solution through the whole domain.*

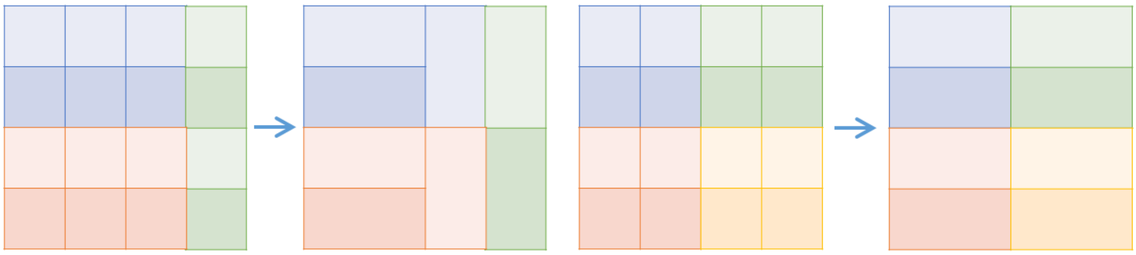


Figure 2.8: *Schematic illustration of differences in the restriction and prolongation process caused by different partitioning. Different colored fields mark different partitions: to the left 3 partitions, to the right 4 partitions.*

However, because of the non-linearity of the governing equations these small errors may cause the solution to take a different path. [28, 31]

The solution may also choose a different path due to floating point errors, mentioned in Section 2.2.5 below. This can be because an operation such as calculating the global residual $R \propto \sum_i^n r_i$ is dependent on the order of execution due to the non-associative nature of floating point arithmetic, as $(x+y)+z = x+(y+z)$ is not necessarily true due to the truncation. Therefore if the order of a reduction operation, such as the summing of residuals, is not constant for different partitions, the result of the operation will differ slightly, which may in a kind of butterfly effect cause the simulation to enter a different solution if it is unstable. A reduction operation is an operation that reduces an array to a scalar value, such as a vector sum. [13]

Meshing can also be done in parallel in Star-CCM+. The volume to be meshed is split up into subvolumes assigned to each processor. The subvolume meshes are then stitched together to form the complete mesh. For the trimmed mesher the mesh generated in each subvolume should be almost identical to a corresponding serial mesh. Differences can appear in the stitching however, because the stitching is heuristic. The difference in cell count should however be small, less than 3% compared to a serially created mesh. [30]

2.2.5 Errors in CFD

As CFD is a simulation of real flows it is unavoidable that the final simulation result will differ to some extent from the real world behaviour. This difference enters in the different steps of the simulation method as errors and uncertainties. Uncertainties are sources of difference that stem from a lack of knowledge. This can be uncertainties in the simulation inputs, or deficiencies in the physics models. Errors can either be numerical or caused by human mistakes, such as bugs in the simulation software or user errors. Only numerical errors are of interest here, and can be classified as computer round-off error, iterative convergence error and discretization error. [34]

Input uncertainties are for example geometrical uncertainties. These can be caused by the inability to capture geometrical features due to simplification of the CAD geometry to aid simulation convergence or lack of resolution of the mesh representation. They can also stem from differences in the CAD model compared to the real world geometry, for example due to natural geometrical variation appearing in production and manufacturing. [34]

Model uncertainties come from the approximation of real world phenomena as mathematical models. A typical example is the simplifications and assumptions involved in turbulence modeling. This uncertainty can be quantified by comparing the results of different models. [34, 33]

Round-off errors are caused by the truncation of floating point numbers on computers. A floating point number is limited by the number of bits in its binary representation, and all digits exceeding that number will be dropped in a form of Taylor series truncation. Round-off errors are usually negligible in comparison to other errors. [33]

Iterative convergence errors come from stopping the iterating the solution before it is fully converged. Usually it is enough to finish iterating when the residuals have dropped by some orders of magnitude, and stabilized. [34, 32]

Discretization errors come from the discretization in space, and time, of the governing equations. The spatial discretization is governed by the computational mesh used, and decreasing the mesh cell size towards zero will lead to a discretized solution tending to the continuum solution of the model equations. With higher mesh resolution the dependence on cell size decreases however, so that the results on finer meshes converges to the continuous solution. [33]

2.2.6 The Adjoint Method for Surface Sensitivity Analysis

The adjoint method is an efficient way of evaluating the sensitivity of some objective with respect to a large amount of input parameters, since only one solution is required per objective. The objective can for example be drag, and the input parameters can be the CAD geometry surface. The adjoint is calculated from the solution of the flow problem under consideration, called primal solution. Because the primal solution is already on discretized form in CFD, the discrete adjoint method is used. [32, 29]

In Star-CCM+ the sensitivity of the objective with respect to the model geometry can be evaluated in two ways, using the adjoint method. The first method, mesh deformation, is mainly used for automatic shape optimization. The second method, surface sensitivity, is computationally less expensive, and is primarily used to identify areas of the model with large impact on the objective. Only the surface sensitivity is explored in this project. [32]

To calculate the surface sensitivity, the sensitivity of the objective L with respect to the mesh is calculated

$$\frac{dL}{d\boldsymbol{X}} = \frac{\partial L}{\partial \boldsymbol{X}} + \frac{\partial L}{\partial \boldsymbol{W}}\frac{\partial \boldsymbol{W}}{\partial \boldsymbol{X}}, \tag{2.49}$$

where $\boldsymbol{X}$ is the mesh and $\boldsymbol{W}$ is the primal solution. Because the derivative matrix $\partial\boldsymbol{W}/\partial\boldsymbol{X}$ in (2.49) is computationally expensive to calculate, the derivative of the residuals $\boldsymbol{R}$

$$\frac{d\boldsymbol{R}}{d\boldsymbol{X}} = \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}} + \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}\frac{\partial \boldsymbol{W}}{\partial \boldsymbol{X}} = 0, \tag{2.50}$$

is used to substitute $\partial\boldsymbol{W}/\partial\boldsymbol{X}$ with

$$\frac{\partial \boldsymbol{W}}{\partial \boldsymbol{X}} = -\left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}\right)^{-1}\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}}. \tag{2.51}$$

A prerequisite for this is that the primal solution is well converged, so that (2.50) holds true. Inserting (2.51) into (2.49) gives

$$\frac{dL}{d\boldsymbol{X}} = \frac{\partial L}{\partial \boldsymbol{X}} - \underbrace{\frac{\partial L}{\partial \boldsymbol{W}}\left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}\right)^{-1}}_{\Lambda^T} \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}} = \frac{\partial L}{\partial \boldsymbol{X}} - \Lambda^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}}, \tag{2.52}$$

where Λ is the so called adjoint vector. The adjoint solution is achieved by solving the equation system

$$\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}^T \Lambda = \frac{\partial L}{\partial \boldsymbol{W}}^T . \tag{2.53}$$

Once the adjoint in (2.53) is solved, the mesh sensitivity in (2.52) can be calculated. Lastly the surface sensitivity is calculated as the sensitivity of the objective with respect to boundary displacements $\delta \boldsymbol{x}_b$: $dL/d(\delta \boldsymbol{x}_b)$. [32, 12]

As mentioned, a well converged steady primal solution is a prerequisite for the adjoint method. In Star-CCM+, adjoint analysis furthermore requires a coupled flow primal solution. In the version of Star-CCM+ used in this project, no turbulence models are implemented in the adjoint method, so the effect of turbulence is only indirectly included by its impact on the primal solution. [32]

2.3 Design of Experiment

Design of Experiment (DOE) is a methodology for planning, conducting and analyzing experiments to make objective and valid conclusions. An experiment is a series of tests in which some input variables are varied to observe changes in the output variables of the studied process. The input variables that are controlled are called factors, and the output variables of interest are called responses.[16]

There are many types of DOE designs. The simplest design is the one-variable-at-a-time (OVAT) approach in which one factor is varied while the others are held constant, as illustrated in Figure 2.9. A more refined type of design is the factorial design. The choice of design depends on the objective of the experiment. A common objective of a DOE is to determine which factors are most important for the process outcome, and which effects they have. For this objective, a factorial design type of DOE is suitable. [16]

The field of DOE was originally developed for physical experiments, which are subject to random effects. This is taken into consideration in a few different ways, such as random run order and replication of tests. Computer experiments such as steady CFD simulations are deterministic however, and not subject to randomness. This makes procedures such as randomization and replication unnecessary. [25, 16]

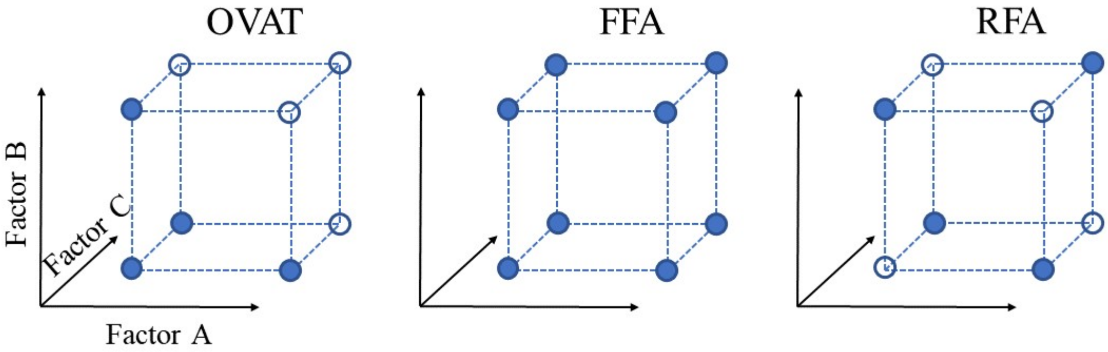


Figure 2.9: *Schematic illustration of three factor experimental designs. The axes illustrate the design space for the three factors A, B and C. The filled corner dots mark the factor level combinations tested in the design, empty corner dots mark not tested combinations.*

Table 2.1: Design table for a 2^3 FFA design.

Test	A	B	C	AB	AC	BC	ABC
1	-	-	-	+	+	+	-
2	+	-	-	-	-	+	+
3	-	+	-	-	+	-	+
4	+	+	-	+	-	-	-
5	-	-	+	+	-	-	+
6	+	-	+	-	+	-	-
7	-	+	+	-	-	+	-
8	+	+	+	+	+	+	+

2.3.1 Full Factorial Design

In a full factorial design (FFA) the factor values are set to some levels and tests are carried out for each combination of factor levels. FFA are useful for determining the main effects of the factors on the response, and also interaction effect between the factors, which the OVAT approach cannot predict. Most common is the 2-level FFA, with a high (+) and a low (-) level for the factor values, denoted a 2^k design where k is the number of factors. The factor levels should span a large range of the factor values, so that any effects on the response are easier to observe. [16]

A 2^3 design, forming a cube in the design space, is illustrated in Figure 2.9, and its design table is shown in Table 2.1. Here A, B and C are the three factors. There are also the interactions between the factors, three 2-factor interactions AB, AC, BC and one 3-factor interaction ABC.[16]

A main effect is the direct effect of a factor and an interaction effect is an effect caused by a combination of factors. A main effect E_A and interaction effect E_{AB} are defined as

$$E_A = \bar{R}_{A+} - \bar{R}_{A-}, \tag{2.54}$$

$$E_{AB} = \bar{R}_{(AB)+} - \bar{R}_{(AB)-}, \tag{2.55}$$

where $\bar{R}_{A+}$ is the average response at the high level of a factor A and $\bar{R}_{A-}$ the average response at the low level of A, and $\bar{R}_{(AB)+}$ and $\bar{R}_{(AB)-}$ is the average response at the high and low level respectively of a factor combination AB. [16]

2.3.2 Reduced Factorial Design

For moderate to large numbers of factors the number of tests needed for an FFA can be inhibitive. In that case a reduced factorial design (RFA) can be used, in which only a fraction of the tests of an FFA are performed. A two level RFA is denoted 2^{k-p}, where k as before is the number of factors, p is the degree of the fraction. For example a 2^{3-1} RFA design is the $2^{-1} = 1/2$ fraction of a 2^3 FFA design, and consists of $2^{3-1} = 4$ tests. This is illustrated in Figure 2.9 [16]

An RFA design is derived from the corresponding FFA design, but only uses the first 2^{k-p} tests. Just reducing the number of tests will mean that some factors are only tested at one level, for example factor C the design in Table 2.2a, which is only set to the low level. In order to account for this the levels of these factors are set to those of a combination of the k first factors. In this case factor C is defined as the combination AB, as shown in Table 2.2b, so that there are test combinations with C at both levels. [16]

Reducing the number of tests means that all main effects and interactions cannot be independently determined. In the 2^{3-1} RFA example this means that when the effect of C is calculated it is in fact the combined effect of $C + AB$ that is calculated. The effects of C and AB are said to be confounded. In the 2^{3-1} case A and BC, and B and AC are also confounded. This means that main effects are confounded with 2-factor interactions in

Table 2.2: Construction of a 2^{3-1} RFA design.

(a) Half of the 2^3 FFA design. Factor C is only at the low level.

Test	A	B	C	AB	AC	BC	ABC
1	-	-	-	+	+	+	-
2	+	-	-	-	-	+	+
3	-	+	-	-	+	-	+
4	+	+	-	+	-	-	-

(b) The 2^{3-1} design table. The factor C levels are set to those of the combination AB.

Test	A	B	C = AB	AC	BC	ABC
1	-	-	+	-	-	+
2	+	-	-	-	+	+
3	-	+	-	+	-	+
4	+	+	+	+	+	+

this design. This is often expressed using a measure called resolution, denoted by a Roman numeral subscript, in this case 2^{3-1}_{III}. The meaning of the most important resolutions are [16]:

Resolution III design: main effects are confounded with 2-factor interactions.

Resolution IV design: main effects are confounded with 3-factor interactions; 2-factor interactions are confounded with other 2-factor interactions.

Resolution V design: main effects are confounded with 4-factor interactions; 2-factor interactions are confounded with 3-factor interactions.

A higher resolution design requires more tests, but can estimate a larger number of effects. Usually, according to the Principle of Scarcity of Effects, the main effects and some low order interactions dominate a process, so the effects of higher order interactions can be neglected. Therefore a lower resolution design with fewer tests are often adequate to determine the most important effects. [16]

3 Part 1: HPC Effects on Simulation Results

Vehicle aerodynamics simulations require a very large mesh in order to accurately predict the flow. To be able to run such large simulations parallel HPC clusters are used. There are several such clusters at Volvo, where some have different hardware, and they are set up so that a simulation can only be run exclusively on one cluster. It was discovered that running two almost identical simulations on two different clusters could result in significant differences in the solutions. It was found that the differences were most apparent at higher yaw angles. The two geometries where this was discovered will be called Case X and Case Y. These differences prompted the current investigations, since what cluster the simulation is run on should not affect the results of a well converged steady simulation.

3.1 Method

A preliminary investigation of Case X and Y was conducted. After this, a factorial design investigation was set up to identify which parallel parameters had an effect on the differences in results. Both the parallel meshing and the parallel simulations were investigated for several trucks models, in both Star-1 and Star-2. Lastly, two measures to address the parallel differences discovered were tested: forcing the cluster MPI implementation to perform reduction operations in a fixed order, and changing the AMG solver settings.

3.1.1 Preliminary Investigation

The objective of the preliminary study was to gain some more information about the problem and determine if the difference in results was in fact caused by which cluster the simulation was run on. Case X and Y were therefore investigated and compared thoroughly to find any potential differences is geometry or settings.

An OVAT type study was also performed, for Case X and Y. The factors considered were

- simulation cluster
- volume mesh cluster
- Case (X and Y)

In Star-CCM+ both volume meshing and simulation can be done in parallel, so both steps were included to identify if the difference appeared in either the meshing step or simulation step.

First the simulations were run on different clusters for the same volume mesh (and hence build) of Case X. Then Case X was volume meshed on different clusters, while the simulation was run on the same cluster. Lastly Case Y was run once. The simulations were only run at a higher yaw angle, since the differences were more apparent there. This investigation was performed in Star-1, since it was in this version the problem first appeared.

3.1.2 DOE Investigation

The preliminary investigation was followed by a larger investigation using factorial design. Both the volume meshing step and the simulation step were investigated in order to verify in which step any differences would appear. Both Star-1 and Star-2 were investigated. Star-2 was investigated to determine if the problem would appear in a newer version of the software. Several different truck geometries were investigated to see if the problem was a one-time occurrence, or if it is more wide-spread. The trucks investigated were

1. FM
2. FH-1
3. FH-2
4. FH-3 - Case X
5. FH-4
6. FH-5

Table 3.1: The factors tested in the volume mesh investigation, with their corresponding code and level values, as well as the level values of the baseline case.

Factor	Code	Low level (-)	High level (+)	Baseline level
Processor count	P	224	448	1
Cluster	C	Cluster A	Cluster E	Cluster A (larger memory)
Node fill	N	50%	100%	N/A

where the FM truck has a medium sized cab with rigid trailer, and the FH trucks have a large sized cab and semi-trailer. The FH trucks are largely similar, but with some differences in geometrical detail. All trucks were investigated in Star-2, but in order to save time and computational resource only FH-1, FH-3 and FH-5 were tested in Star-1.

Volume Mesh DOE Study

The volume mesh DOE study focused on three factors that the user can control when submitting a parallel job, and which were assumed to potentially affect the simulation results or computation time. The processor count factor (factor P) was chosen because of the effects reported in the theory, Section 2.2.4, and because of scalability effects on cluster time. The levels were chosen as 224 and 448 processors, to give a large span for estimating the effect. A larger count would have been chosen for the high level (896 processors) if not for stability issues. The cluster factor (factor C) was selected because differences in hardware were assumed to affect results by different round-off error, and cluster time by different processor performance. The levels were chosen as cluster A and E, because the difference between the hardwares is significant. The node fill (factor N), that is the percentage of the processors used by the job, was chosen mainly because the assumed impact on cluster time due to more efficient communication. The levels were chosen as half (50%) and full nodes (100%) to give a large span for the effect. The response variable for mesh differences was chosen to be difference in total cell count compared to a baseline case. The baseline case was run in serial on a cluster A node with larger memory. The response for the cluster time was simply chosen to be the computational time of the cluster job. The factors and their levels are shown in Table 3.1.

A 2^3 FFA design was chosen, which resulted in 9 test runs per truck, including the baseline test. The tests were run and then the results analyzed.

Simulation DOE Study

The simulation DOE study was performed with the same parallel parameter factors as the mesh study, processor count (P), cluster (C) and node fill (N), with the addition of the factors yaw angle (factor Y) and mesh (factor M). The yaw angle was included to see if the problem was present for other yaw angles, and the mesh factor was included to see if the parallel meshing effects had any influence on the problem. The factor P levels were changed to 224 and 896 processors, and the factor N levels were changed to 75% and 100%. The change to 896 processors was made simply because the stability issues with 896 cores for the mesh DOE study were not present here. The change to 75% was made because it was deemed wasteful and no benefit was seen in only running on half nodes. The levels of factor Y were chosen as 0.0° and $-7.5°$, where the difference was first discovered. The levels of factor M were chosen as one mesh created in serial and one in parallel from the mesh study. The parallel mesh was created on cluster D with 224 cores and full (100%) nodes. The factors and their levels are shown in Table 3.2.

The response variables were chosen to be difference in C_D, and simulation time. The difference in C_D was measured compared to one baseline test for each yaw angle. The baselines could not be run in serial, both because of time and memory requirements, and were therefore run on 112 cores on cluster A, with full nodes and serial mesh. They were run on 112 processors because that was the least number of processors that would not cause a memory crash during the simulation. The baseline factor levels are shown in Table 3.2.

The $2_{\mathrm{V}}^{(5-1)}$ RFA design of [17], shown in Table 3.3, was selected to reduce the number of simulations to a tolerable number. The number of tests is 18, including the 2 baseline tests. A $2_{\mathrm{III}}^{(5-2)}$ design with 10 tests, including

Table 3.2: The factors tested in the simulation investigation, with their corresponding code and level values, as well as the level values of the baseline case.

Factor	Code	Low level (-)	High level (+)	Baseline level(s)
Yaw angle	Y	0.0°	−7.5°	0.0° & −7.5°
Volume mesh	M	Serial (S)	Parallel (P)	Serial
Processor count	P	224	896	112
Cluster	C	Cluster A	Cluster E	Cluster A
Node fill	N	75%	100%	100%

Table 3.3: Design table for the 2_{V}^{5-1} design.

Test no.	Y	M	P	C	N
	1	2	3	4	5 = 1234
1	-	-	-	-	+
2	+	-	-	-	-
3	-	+	-	-	-
4	+	+	-	-	+
5	-	-	+	-	-
6	+	-	+	-	+
7	-	+	+	-	+
8	+	+	+	-	-
9	-	-	-	+	+
10	+	-	-	+	-
11	-	+	-	+	-
12	+	+	-	+	+
13	-	-	+	+	-
14	+	-	+	+	+
15	-	+	+	+	+
16	+	+	+	+	-

baselines, was also considered, but dismissed since main effects are confounded with 2-factor interactions. The tests were run and then analyzed, including visual analysis of relevant flow field images.

3.1.3 Investigations Regarding Solution of Problem

Once the DOE investigation was done, one case from Star-1 and one from Star-2 with significant differences in results were identified. The next step was to find a solution to the dependency on parallel parameters. Two possible measures were identified in the theory, Section 2.2.4: forcing a constant order on reduction operations of the MPI, and changing the AMG solver settings.

Constant Order of MPI Reduction Operations

After contact with Star-CCM+ support, [13], the difference in results between clusters was judged to be caused by round-off errors due to differing order of reduction operations in the MPI implementation. In order to resolve this it was proposed to force a constant order of the operations. If the default MPI implementation in Star-CCM+, Platform MPI, is used a flag can be added when launching a simulation from the command line. The flag is:

```
-mppflags "-e MPI_COLL_FORCE_CONSTANTORDER=1"
```

The simulations for the two cases were run again with the constant order MPI reduction, and the results evaluated to see if the measure was effective. The run time was also considered, to determine if the solution would increase computational time.

Changing AMG Solver Settings

In order to find out if the parallel difference, mainly between different processor counts, was caused by insufficient convergence of the AMG solver, the effects of changing the AMG settings was investigated. Two different settings were tested

1. Reducing convergence tolerance by a factor 10^3 and increasing max cycles to 100; Iterative solver: Gauss-Seidel with group size 4 (default)
2. Reducing convergence tolerance by a factor 10^3 and increasing max cycles to 100; Iterative solver: ILU with group size 8 (7)

In Star-1, ILU with group size 7 was used since using group size 8 caused the simulation to crash.

The simulations for the two cases were run again with the new AMG settings, without the MPI flag, and the results evaluated to see if the measure was effective. The run time was also considered, to determine if the solution would increase computational time.

3.2 Results and Discussion

Here the results from the preliminary investigation and the DOE investigations are presented, as well as the effect of the proposed solutions.

3.2.1 Case X and Y Findings

The results of the preliminary OVAT style investigation is shown in Table 3.4. As can be seen, no significant difference can be observed in Case X for change of either mesh or simulation cluster, but the difference between Case X and Y can be replicated. The difference between the cases was later identified to be caused by the bi-stable modes investigated in Part 2, Chapter 4.

In the investigation and comparison of both cases, no difference was found in either the mesh or simulation settings, but there was a small difference in cell count in the mesh. The CAD files were identical, apart from one single component, a chassis rear side panel, which had a slightly coarser surface mesh in one case. It could not be verified that this was the sole cause for the difference, but it may contribute to the problem. This indicates that the problem is highly mesh dependent. Since the difference in mesh was so small, and in a region not nearby the largest difference in the flow field, the effect of some parallel parameter could not be ruled out.

3.2.2 Impact on Mesh

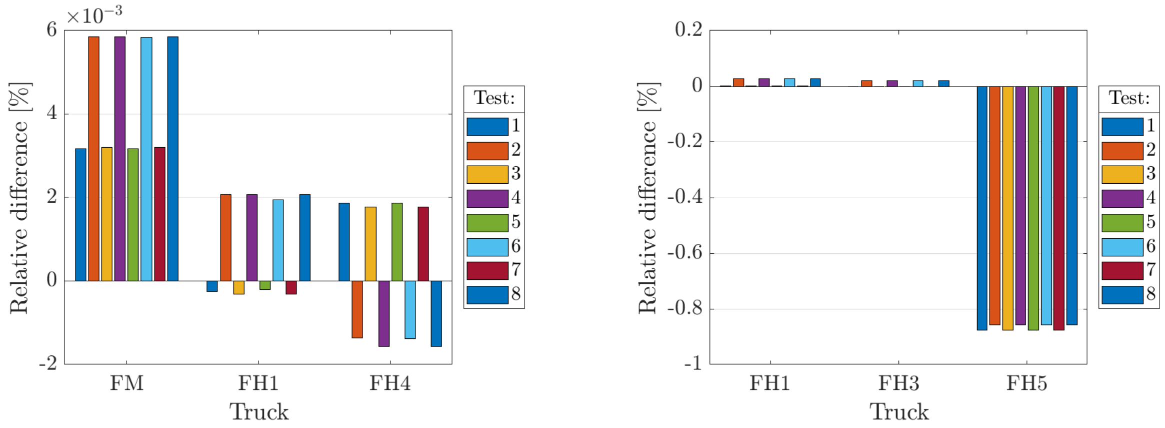
The impact on the mesh of the HPC factors was judged by visually inspecting section planes of the mesh in critical areas, and by comparing the total cell count and some mesh quality metrics. Among the trucks tested in Star-1, the parallel meshing of FH-2, FH-3 and FH-5 failed for all tests. The visual inspection did not reveal any significant differences in the mesh for the trucks that were successfully meshed. This is supported by the comparison of the mesh size and quality metrics. The difference in cell count compared to the serially created mesh is shown for all test in Figure 3.1. As can be seen the difference is in the order of 10^{-3}% in Star-2, and about the order of 10^{-2}% in Star-1, with the exception of FH-5, which deviates with almost 1%. Also the difference in the number of bad aspect ratio cells, shown in Figure 3.2, is small, about the order of 10^{-2} and 10^{-1} for the new and old version respectively, again with the exception of FH-5 at about 1.5%. Lastly, the difference in the number of highly skewed cells is shown in Figure 3.3. Here the difference is significant, but the effect of the difference may be small as the number of cells with high skewness angle still only amount to about 10^{-3}% of the total cell count. So, the overall impact on the mesh from parallel execution is seen to be very low, even lower than the 3% estimated by Star-CCM+ [30]. In most cases the resulting effects on the simulation results would be expected to be negligible, unless there is a very strong sensitivity to the mesh. FH-5 is the truck with the largest difference, so if the mesh difference affects the simulation results, it should show for this truck.

3.2.3 Impact on Simulation Results

Of all the trucks tested in the simulation DOE investigation, only the FM in the Star-2 and FH-3 in Star-1 did show any differences between the tests. The difference only appeared for −7.5° yaw. In both these cases the difference was found to be that the solution entered one of the two bi-stable modes in Part 2, Chapter 4. The

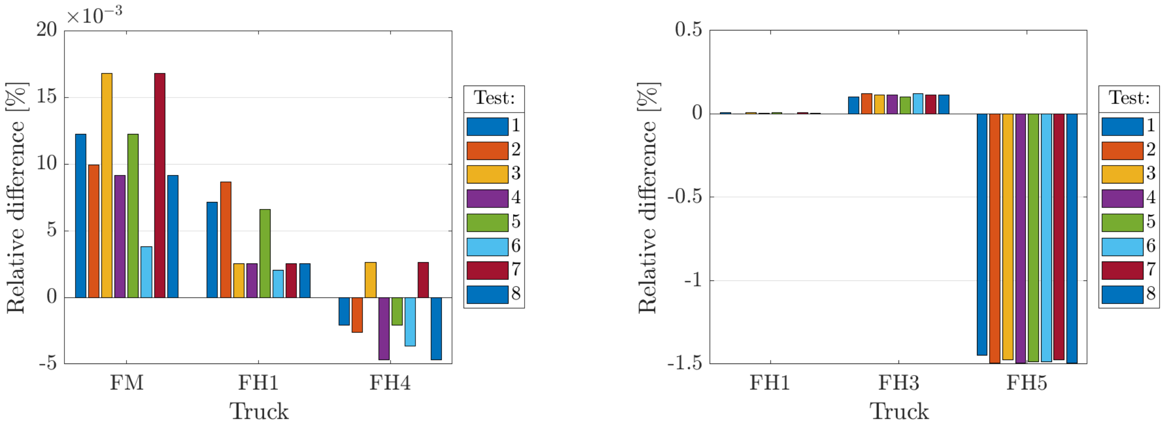
Table 3.4: Test matrix for OVAT-experiment of the preliminary investigation. “(X1)” denotes that the same mesh file was used as a previous test.

Test no.	Case	Mesh cluster	Simulation cluster	ΔC_D [%]
1	X	A	A	-
2	X	(X1)	C	0.03
3	X	(X1)	D	0.03
4	X	(X1)	A	0.00
7	X	B	A	0.05
8	Y	A	A	1.79



(a) *Star-2.* **(b)** *Star-1.*

Figure 3.1: *Relative difference in total cell count compared to the serial mesh.*

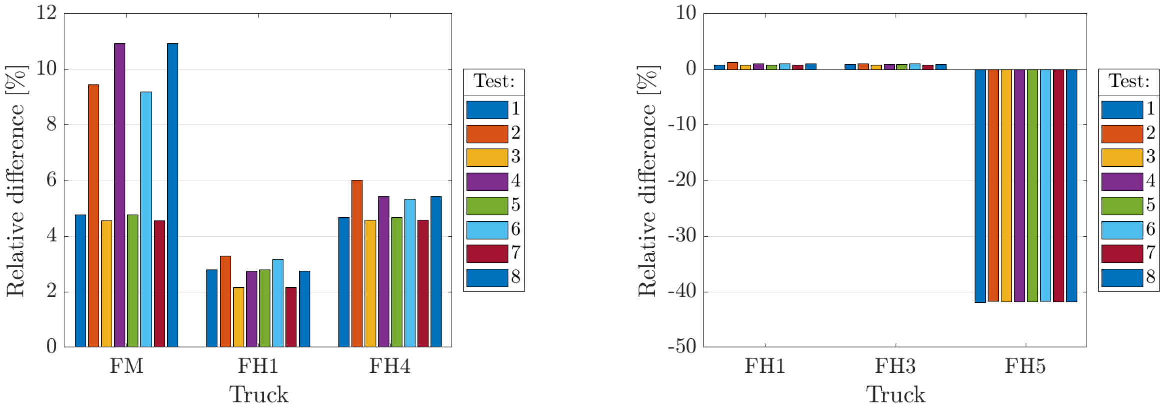


(a) *Star-2.* **(b)** *Star-1.*

Figure 3.2: *Relative difference in the number of cells with bad aspect ratio (< 0.01) compared to the serial mesh.*

differences from HPC effects is thus caused by an underlying bi-stability in the simulated case, in which the solution is highly sensitive to perturbations that otherwise would be negligible. The differences are therefore not continuous, but discrete: Mode 1 or Mode 2. This supports the theory in that differences due to HPC effects should not normally occur for steady simulations.

The mode observed for each test combination for the affected trucks are shown in Tables 3.5a and 3.5b. The FM can be seen to enter Mode 2 in the baseline test, and for 896 processors on cluster A, and for the parallel mesh test with 224 processors and full nodes on cluster E. The FH-3 only shows Mode 2 for 896 processors on cluster A. These results show that the most important HPC factors that may cause a change of mode are the simulation cluster and the processor count. Test number 12 for the FM, in Table 3.5a, indicate an interaction effect with either the mesh or the node fill. It is most probably a mesh effect, since the node fill should not affect the partitioning. In that case it seems that differences from parallel meshing may affect the results. This is probably more due to a strong mesh sensitivity of that case than significant differences in the parallel meshing, since the mesh differences for the FM was found to be very small. This further supports the theory in that the differences arise in the AMG solver, Section 2.2.4, either because of partitioning differences or floating point errors.



(a) *Star-2.* **(b)** *Star-1.*

Figure 3.3: *Relative difference in the number of cells with bad skewness angle (> 85°) compared to the serial mesh.*

Table 3.5: The factor level combination for each test with −7.5° yaw, and the resulting bi-stable mode for the two trucks with observed differences. The bi-stable flow states are investigated in Chapter 4

(a) FM truck, Star-2.

Test no.	M	P	C	N	Mode
Baseline	S	112	A	100%	2
2	S	224	A	75%	1
4	P	224	A	100%	1
6	S	896	A	100%	2
8	P	896	A	75%	2
10	S	224	E	75%	1
12	P	224	E	100%	2
14	S	896	E	100%	1
16	P	896	E	75%	1

(b) FH-3 truck, Star-1.

Test no.	M	P	C	N	Mode
Baseline	S	112	A	100%	1
2	S	224	A	75%	1
4	P	224	A	100%	1
6	S	896	A	100%	2
8	P	896	A	75%	2
10	S	224	E	75%	1
12	P	224	E	100%	1
14	S	896	E	100%	1
16	P	896	E	75%	1

3.2.4 Effects of Solution Measures

The differences between clusters can be avoided by always using the same cluster. However, that would limit analysis capacity and be wasteful of resources if several clusters are available. Another solution is to force a constant order of reduction operations, thus eliminating round-off error differences. The modes resulting from this solution measure is shown in Table 3.6. As can be seen, this caused the FM to only enter Mode 2, for all test combinations. This implies that the reduction order affects not only differences between clusters, but to some extent also other differences. For FH-3 all simulations with the same number of processors entered the same mode, which was Mode 1 for the baseline and 224 core tests, and Mode 2 for the 896 core tests. This shows that using constant order MPI reduction operations is effective at eliminating the differences between simulation clusters. It also indicates that the partitioning also can affect reduction operation order, causing round-off error differences.

The solution to eliminate differences caused by HPC effects in general is to run the AMG solver until convergence in each iteration. The modes resulting from decreasing the AMG convergence tolerance to 10^{-4} are shown in Table 3.7, using both the Gauss-Seidel and the ILU iterative solvers. As can be seen, decreasing the tolerance resulted in the FM only choosing Mode 2, except for in test 2 for the Gauss-Seidel solver, and the FH-3 only choosing Mode 1. This shows that reducing the convergence tolerance can be used to eliminate the differences, and that the ILU is possibly more efficient in this. Both differences caused by simulation cluster and number of processors are eliminated. This confirms that the problem is caused by insufficient convergence in the AMG solver. As the FM chooses Mode 2 and FH-3 chooses Mode 1, it appears that increasing the AMG convergence is not a solution to the underlying bi-stability problem however. Using a lower AMG convergence tolerance is therefore not a solution for the bi-stability problem.

Table 3.6: The bi-stable modes resulting from using constant order MPI reduction operations for the two trucks with observed differences.

(a) FM.

Test no.	M	P	C	N	Mode
Baseline	S	112	A	100%	2
2	S	224	A	75%	2
4	P	224	A	100%	2
6	S	896	A	100%	2
8	P	896	A	75%	2
10	S	224	E	75%	2
12	P	224	E	100%	2
14	S	896	E	100%	2
16	P	896	E	75%	2

(b) FH-3.

Test no.	M	P	C	N	Mode
Baseline	S	112	A	100%	1
2	S	224	A	75%	1
4	P	224	A	100%	1
6	S	896	A	100%	2
8	P	896	A	75%	2
10	S	224	E	75%	1
12	P	224	E	100%	1
14	S	896	E	100%	2
16	P	896	E	75%	2

Table 3.7: The bi-stable modes resulting from lowering the AMG convergence tolerance to 10^{-4}, using the Gauss-Seidel (G-S) scheme and using the ILU scheme.

(a) FM.

Test no.	M	P	C	N	G-S	ILU
Baseline	S	112	A	100%	2	2
2	S	224	A	75%	1	2
4	P	224	A	100%	2	2
6	S	896	A	100%	2	2
8	P	896	A	75%	2	2
10	S	224	E	75%	2	2
12	P	224	E	100%	2	2
14	S	896	E	100%	2	2
16	P	896	E	75%	2	2

(b) FH-3.

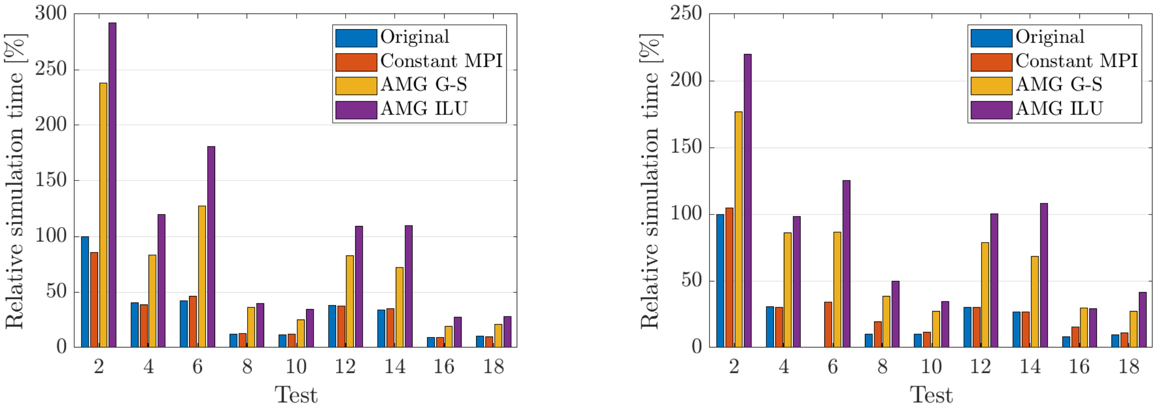
Test no.	M	P	C	N	G-S	ILU
Baseline	S	112	A	100%	1	1
2	S	224	A	75%	1	1
4	P	224	A	100%	1	1
6	S	896	A	100%	1	1
8	P	896	A	75%	1	1
10	S	224	E	75%	1	1
12	P	224	E	100%	1	1
14	S	896	E	100%	1	1
16	P	896	E	75%	1	1

If the proposed solution measure is to be feasible to implement, it cannot increase the simulation time too much. The simulation times for the proposed measures are shown in Figure 3.4. As can be seen, the simulation time is approximately the same using the constant MPI reduction order solution, making it a feasible solution. Lowering the AMG tolerance on the other hand can be seen to increase simulation time by a factor 2 to 4, making it too expensive to justify implementation. The proposed order of action to avoid impact of parallel parameters on results is therefore to use a standard number of processors for simulations, and thereby avoiding differences between processor counts, and implementing the constant MPI reduction order to avoid differences between clusters. Ideally the underlying problem with the instability of the simulations would be addressed first. This is the objective of the next part of the project, Chapter 4.

3.2.5 Impact on Computational Time

Only the computational times for Star-2 are considered here. The main effects of HPC effects on meshing time are shown in Figure 3.5. The meshing time can be seen to be only about 3.3% of the time of the serial meshing. The effect of the most important factor, the meshing cluster, is about 0.5 pp (percentage points), with cluster E being faster than cluster A. The next most important factor is the processor count, with a higher number of processors resulting in faster meshing by about 0.4 pp. The effect of node fill is insignificant. The interaction effects on meshing time are negligible and are not reported here.

From an efficiency point of view it is not only the absolute time that is interesting, but also the total time all nodes are occupied, defined as Total number of nodes × Total job time. This is shown in Figure 3.6. Here it can be seen that the occupied node time lies at about 55% of the serial occupied node time, and that the effects are larger. The most important factor here is the node fill, which shows a decrease of about 35 pp going



(a) *FM.* **(b)** *FH-3.*

Figure 3.4: *Comparison of simulation times relative to the baseline original time for the proposed solution measures.*

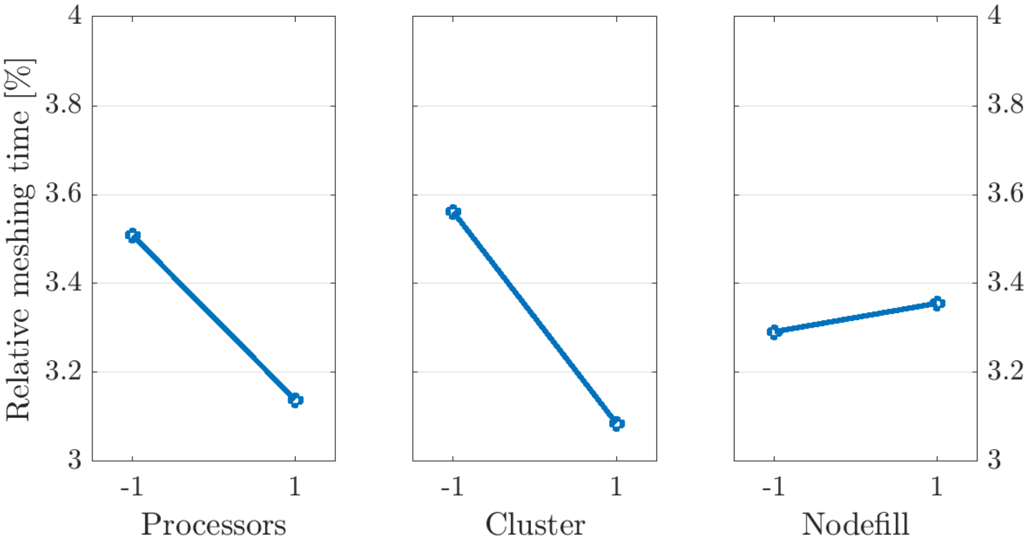


Figure 3.5: *Main effects plot of meshing time relative to the serial baseline.*

from half nodes to full nodes. The effect of processor count on the other hand shows an increase of about 30 pp using 448 processors compared to 224, as opposed to the decrease in absolute time. Cluster E can be seen to be moderately more efficient than cluster A, about 8 pp faster.

These results show that the benefit in absolute time of going from 224 to 448 processors is small, and the occupied node time in fact increases significantly. Therefore it is considered better use of computational resources to volume mesh using 224 processors in this case. This may also decrease cluster waiting time, as it is easier to allocate a smaller number of cores. Using cluster E is slightly faster than cluster A, so it could be advantageous to prefer using cluster E, but only marginally. The node fill has no significant effect on the absolute time, but significantly lowers efficiency by increasing occupied node time. This is probably so because there is less communication involved in the meshing compared to the simulation. It is therefore best to always use full nodes when meshing.

The main effects on simulation time are presented in Figure 3.7. The simulation time lies at about 20% of the baseline simulation time, using 112 processors. The most important effect is the processor count, which decreases from about 33% to 9%, going from 224 to 896 processors. This means that doubling the number

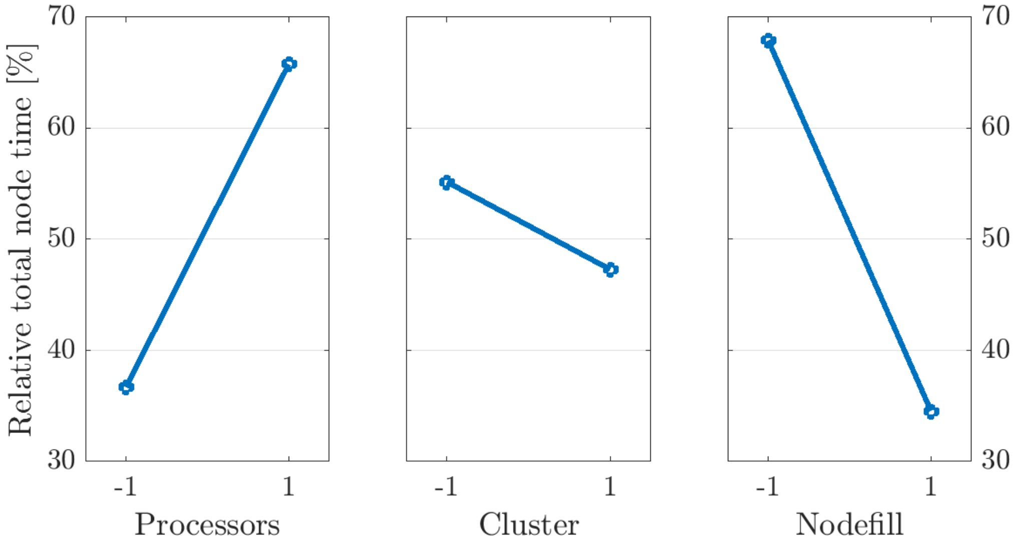


Figure 3.6: *Main effects plot of total node meshing time relative to the serial baseline. The total node time is the total time the compute nodes are occupied (total number of nodes × total job time).*

of processors from 112 to 224 gives a 3 times faster simulation: a 2/3 relation. Increasing from 224 to 896 processors gives a $4/3.67 = 12/11$ relation. So there is a trend diminishing returns for increasing the number of processors, with the breaking point somewhere between 224 and 896. Cluster E can furthermore be seen to be about 5 pp faster than cluster A. The effect of node fill shows an increase of about 3 pp in simulation time, going from 3/4 full nodes to full nodes. The interaction effects on simulation time are negligible and are not reported here.

Looking at the efficiency, the effects of total occupied node time for the simulations are shown in Figure 3.8. Occupied node time is seen to increase by 8 pp for 896 processors. The effect of both cluster and node fill is a decrease of about 17 pp.

These results show that increasing the processor count will significantly decrease simulation time, although at diminishing returns from number between 224 and 896 processors. This makes it less efficient to run simulations with 896 cores, but not by much. If the computational resource is available, the higher number of processors can definitely be used to get results faster. Cluster E is slightly faster and more efficient than cluster A, so it should be used as the first choice. This is the case also for the meshing, but as the meshing is much faster than the simulation, it can be beneficial to primarily use cluster E for simulations and cluster A for meshing. Full nodes should always be used, as it is significantly less efficient leave processors idle.

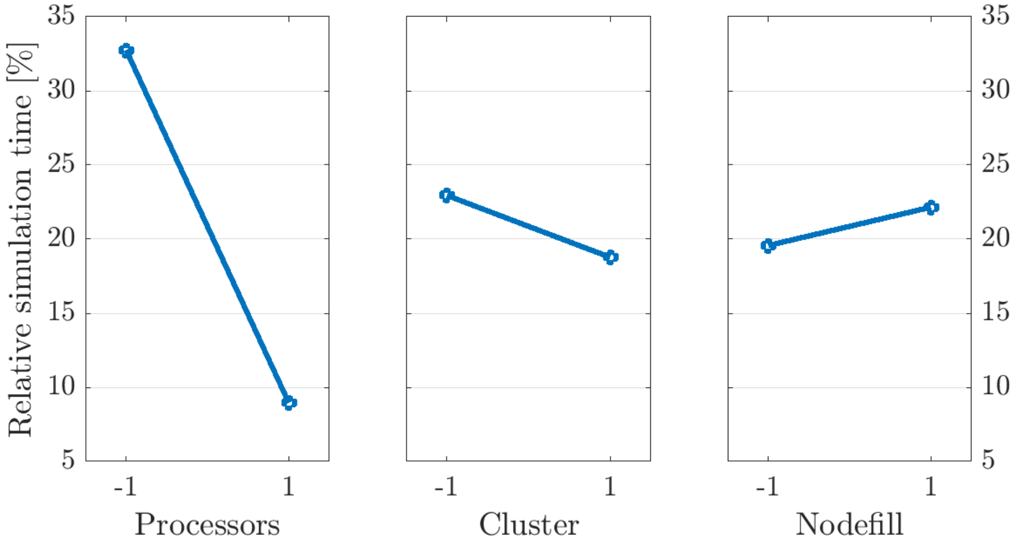


Figure 3.7: *Main effects plot of simulation time relative to the* 0.0° *yaw baseline.*

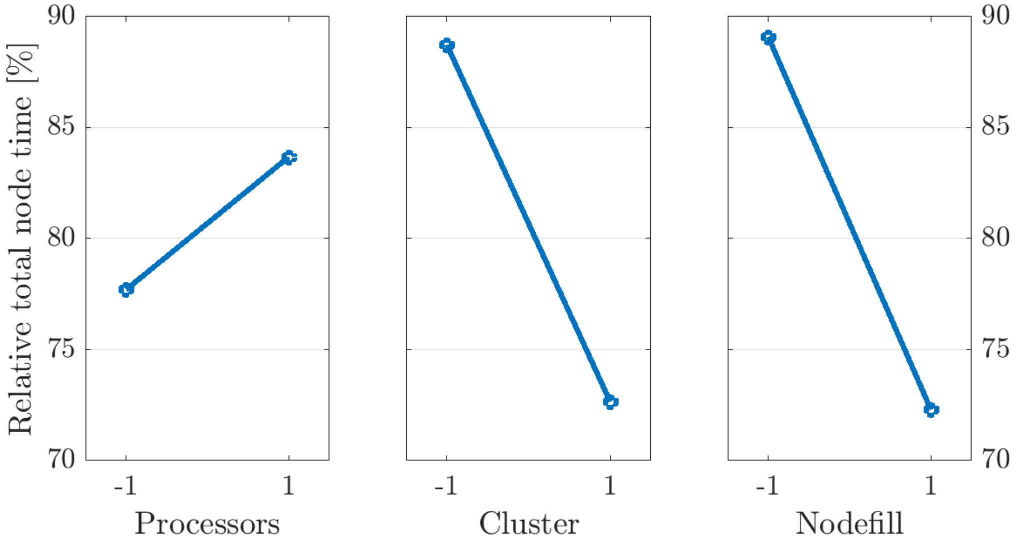


Figure 3.8: *Main effects plot of total node simulation time relative to the* 0.0° *yaw baseline. The total node time is the total time the compute nodes are occupied (total number of nodes* × *total job time).*

3.3 Conclusion

The conclusion of this part of the project is that the difference in results due to HPC effects can be replicated, and that it only appears for certain cases with an underlying instability. The issue is an exception rather than the norm, and steady simulations should normally not be affected by HPC effects. The problem arises in the simulation step, although differences from parallel meshing may have a small effect. The parallel meshing differences are otherwise negligible. The factors that affect the difference is the HPC cluster and the number of processors used for the simulations. The problem stems from the instability of the case, but is affected by HPC parameters because of insufficient convergence in the AMG solver at each solver iteration.

The differences can be eliminated by decreasing the AMG convergence tolerance to a sufficiently low value, but this severely impairs performance. Differences caused by different cluster hardware can instead be avoided by forcing a constant MPI reduction operation order. If Platform MPI is used, which is standard for Star-CCM+, this can be done using the command `-mppflags "-e MPI_COLL_FORCE_CONSTANTORDER=1"` when launching a simulation from the command line. Differences caused by running on different processor counts are most easily avoided by simply always using the same number of processors for simulations. The best course of action however would be to investigate the underlying stability issue and solve it if possible. This is the objective of the next part of the project, Chapter 4.

The effects of HPC parameters on the volume meshing and simulation time were also investigated, for Star-2. It was found that it is most efficient to volume mesh using 224 processors. The meshing time can be decreased by using more processors, but only marginally. Regarding simulations it is only slightly less efficient to use 896 processors than 224, so the choice depends on how fast the results are needed. In both cases cluster E is slightly more efficient than cluster A. As the simulation time is significantly longer than the meshing time it is recommended to run simulations primarily on cluster E and meshing on cluster A, but the difference is not large. Full nodes should always be used, as there is no benefit in efficiency in doing otherwise.

4 Part 2: Bi-stable Flow States

A part of aerodynamic development is the evaluation and comparison of different concepts. Geometrical differences can often be quite small, so the resulting flow fields are expected to be mostly similar. It was however discovered that small differences in geometry caused the solution to enter one of two bi-stable states, with significant differences in results. There was furthermore no predictable connection between geometry change and the resulting flow state. In some cases both flow states could occur for the same geometry, and the resulting state was affected by which cluster the simulation was run on, which was the topic of the previous part, Chapter 3. The bi-stable flow states were discovered in Star-1.

4.1 Method

The majority of the investigations was performed in Star-1. In order to save computational resources, these investigations are primarily performed on a few yaw angles from one side, where the differences between the modes were the largest.

A preliminary investigation of multiple cases of the affected project was investigated to gain more information about the phenomenon. The characteristics of the modes were studied, the geometry for which they appeared, dependence on side wind yaw angle, and differences in mesh and solver settings. Five cases were selected for further investigations to represent different combinations of the modes, based on the original simulations:

1. "Baseline case": only Mode 1
2. only Mode 1
3. both modes
4. both modes
5. "Extreme case": only Mode 2 except for 0.0° yaw

Case 1 is the baseline model, shown in Figure 2.3a. Case 2-5 all have some geometry modifications compared to the baseline. Case 3 and 4 have both modes, but for different yaw angles. The selected cases were run for yaw angles covering both sides, from 10.0° to −10.0°, to see if both modes were present on both sides. In Star-2, the yaw angles on one side, from 0.0° to −10.0° were run, to see if the problem was present in the newer versions of the software as well.

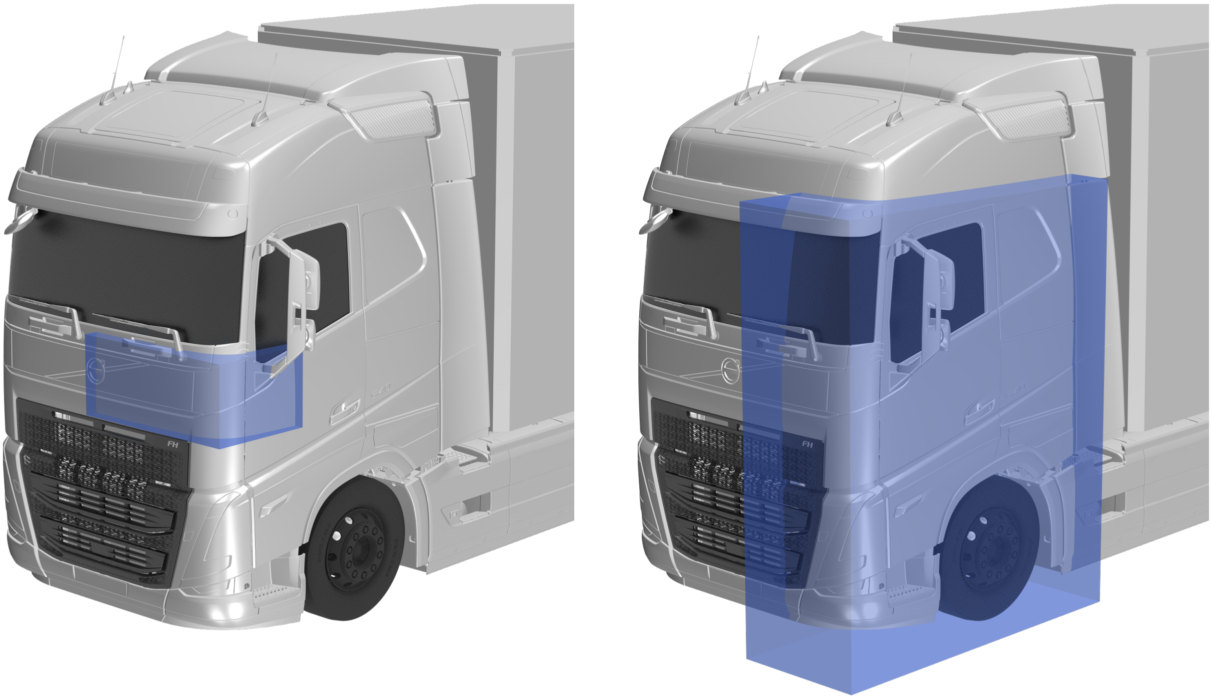
The preliminary investigations also included investigations of local convergence and monitoring the solution development. This led to an investigation of the effects of changing the initial and inlet conditions. A geometric modification was also tested, to see if the solution could be forced into one mode and to learn more about the modes. The preliminary investigation was followed by a mesh study and a study of turbulence models, in which both RANS models and a DES model was tested.

4.1.1 Study of Local Convergence

In order to see if there were any areas of interest that had a high remaining error, which could potentially affect the global solution, the local convergence was investigated.

4.1.2 Solution Development and Effect of Initial and Inlet Conditions

Images of the flow field were exported during the solution process to observe any instabilities in the development of the solution. This lead to a small investigation of the effects of changing the initial conditions for the simulations. This was done in two ways, both only for Case 1 and Case 5. The first way was to initiate the simulation with the solution from a simulation of another yaw angle. A 0.0° and a −15° initial were tested. The other way to change the initial conditions was to ramp the inlet velocity over the first 100 and the first 1000 iterations. Changing the inlet velocity to 80 and 100 kph was also tested.



(a) *Small refinement region.* **(b)** *Large refinement region.*

Figure 4.1: *The refinement regions used in the mesh study.*

4.1.3 Effect of Geometric Modification

Based on the observations from the previous studies a geometrical addition in the form of seal was investigated to prevent Mode 2, under the assumption that Mode 2 may be physical. The seal was placed between the front panel and the door to close off the split-line between them.

4.1.4 Mesh Study

In order to determine if either of the modes is non physical, by increasing the resolution of the flow in the region where difference originated, a local mesh study was performed. A small and a large region of refinement were studied, shown in Figure 4.1, with three levels of refinement, in percentage of cell base size:

1. 25%
2. 12.5%
3. 6.25%

4.1.5 Study of Turbulence Models

A study of turbulence models was performed to investigate if both modes appeared also for other models. Four RANS models were investigated, and one DES. The RANS models were

1. **SKE DSL=1.2**: Standard $k-\varepsilon$ model with Durbin realizability limit $C_T = 1.2$
2. **SKE DSL=2.4**: Standard $k-\varepsilon$ model with Durbin realizability limit $C_T = 2.4$
3. **SST**: Standard SST $k-\omega$ model with coefficients recommended for vehicle aerodynamics
4. **RKE**: Realizable $k-\varepsilon$ model with standard coefficients

The **SKE DSL=1.2** is the recommended model for road vehicle aerodynamics according to Siemens [27]. The **SKE DSL=2.4** model was tested because of the effects of the DSL on the separation. The **RKE** and

SST models were tested for their increased capability in predicting separation, which should give some more information on the physicality of the modes.

The DES was run with a **SKE DSL=1.2** precursor, to initiate the flow field. The SST model was used for the RANS layer. The objective was only to observe which if the modes were also present in a transient solution, and not to get accurate results statistics. Because of this, and to save computational expense, a shorter simulation time of 2 s was chosen for the DES. Two longer simulations were run: Case 1 $-5.0°$ yaw was run for 30 s, and Case 5 $-7.5°$ yaw was run for 60 s. This was to observe any mode switches during the simulation time. Ideally even longer simulations would be run, corresponding to the $T = 10^3 H/U_0$ timescale reported in [9]. This would require a simulation time of T ~120 s, which could not be afforded.

4.2 Results and Discussion

Here the results of the investigations are presented and discussed. An xy-section plane through the upper cab will be used to present the flow field differences between the modes.

4.2.1 Description of the Flow States

The two bi-stable modes are referred to as Mode 1 and Mode 2. They are mostly similar, but with one large difference that appear for larger yaw angles. The difference in the velocity field is shown in Figure 4.2. As can be seen, the difference appear by the lee-side upper front corner, where there is a separation bubble. In Mode 1 that separation bubble is quite small, and almost reattaches to the door side, before it joins the larger total lee-side separation. This can be seen as the blue shape in the velocity difference field. In Mode 2 the separation bubble instead is much larger, and does not reattach, but instead joins the total lee-side separation. This can be seen as the union of the blue and red shape in the velocity difference field. The red area clearly shows the difference, as the velocity in Mode 1 is much faster here because the flow has not yet separated.

The pressure field difference in the front corner area is shown in Figure 4.3a. It can be seen that the pressure around the corner is lower for Mode 1. It can also be seen that in Mode 1, there is first one local pressure minimum by the front corner. After this the pressure increases slightly, followed by a second local minimum by the front door split-line. This results in a smaller APG, followed by a second FPG, which suppresses the separation bubble in Mode 1. In Mode 2 there is only a single strong APG by the front corner, causing the larger separation. Looking at the difference in TKE in Figure 4.3b it can be seen that the TKE is higher in Mode 1 in the door split-line. This further shows that this is a sensitive area. It also indicates that turbulence modeling is of great importance for correctly predicting the separation.

The investigation of the local convergence also showed that there were areas with high remaining residuals in this region, particularly the turbulence residuals. The momentum and pressure residuals did not show such a clear pattern. This indicates that there may be some discretization error causing the bi-stability.

Yaw Dependence and Impact on Drag

Investigating several cases with similar geometry it was seen that Mode 2 is present for almost all cases, with only a few exceptions. The modes for the five selected cases are shown in Figure 4.4. The dotted columns signify that the simulation for that yaw angle was very sensitive and was discovered to enter different modes when simulated on different clusters. It can be seen that Mode 2 appears at about ±5.0° yaw and transitions back to Mode 1 at about ±7.5° to ±10° yaw. The difference increases with the yaw angle, as the separation bubble in Mode 2 grows with increasing yaw. Figure 4.4 also shows that both modes appear for side wind from both sides of the truck. It however seems that Mode 2 is more rare on the right hand side of the truck.

The impact on C_D increases as the bubble grows. This can be seen in Figure 4.5, which shows the average difference in C_D between the negative yaw values and the positive yaw values of Case 2 and 5 in Figure 4.4. Because both these cases have Mode 2 for all negative yaw angles and Mode 1 for all positive yaw angles, they can be used can be used to estimate the impact on C_D. This shows that the effect on C_D in fact is lower for Mode 2 for $-5.0°$ yaw, by about 9 DC. For $-7.5°$ Mode 2 instead gives about 15 DC higher C_D, and for -10.0 it is almost 35 DC higher. The impact on drag is therefore significant, especially at higher yaw angles, although it is interesting to note that at lower yaw the effect is negative. The lower drag for $-5.0°$ was found to be caused by the Mode 2 larger separation bubble suppressing another separation bubble, decreasing the size of the total separation.

In Figure 4.6 the modes resulting from simulations in Star-2 are shown. It can be seen that both modes are still present, but that Mode 2 only occurs for $-5.0°$ yaw. Mode 2 has however been seen for higher yaw in Star-2 in some simulations in Part 1, Chapter 3, so Mode 2 can still appear for larger yaw. It can also be seen in Figure 4.6 that Mode 2 does not show for Case 2, as opposed to Star-1. These results show that Mode 2 appears also in Star-2, but it also indicates that Mode 2 is less common and does not appear as often for higher yaw-angles.

4.2.2 Effects of Changing Initial Conditions

It was seen in the monitoring of the solution development during the simulations that all cases experienced a form of “splash” effect in the first 100 iterations, with a large separation bubble such as the one seen in Mode 2, which by 200 iterations changed more or less into the final Mode 2. After 200 iterations the separation bubble starts to recede for the cases which end up in Mode 1. From this it is clear that the final mode is determined

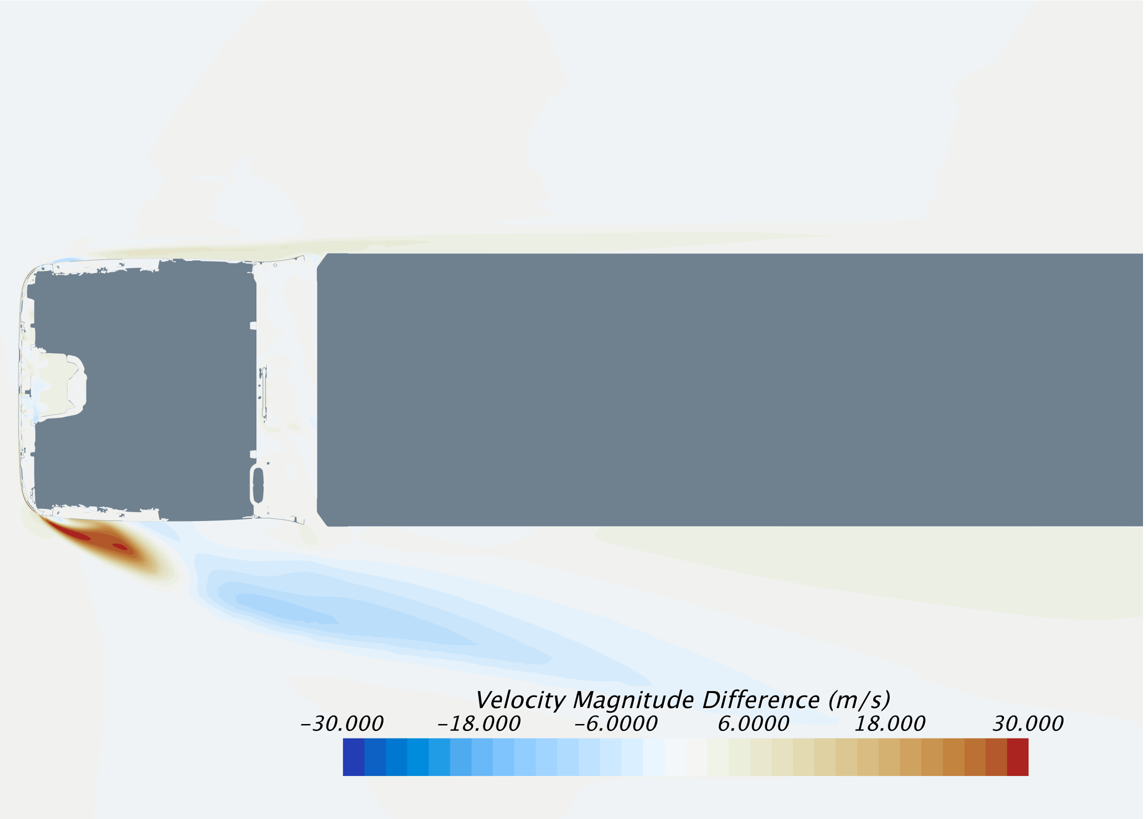
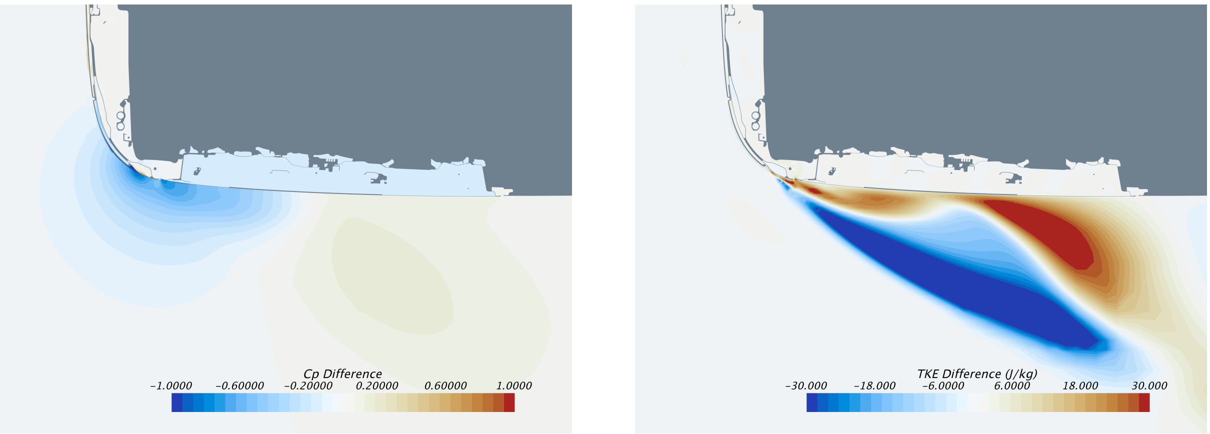


Figure 4.2: *Difference in velocity magnitude between the two modes, $V_{Mode\ 1} - V_{Mode\ 2}$. The flow conditions are yawed, with the side wind coming from the right side of the truck, or top in the image.*



(a) *Pressure coefficient.*

(b) *TKE.*

Figure 4.3: *Zoomed in view of differences in pressure and in turbulence between the modes. Mode 1 - Mode 2.*

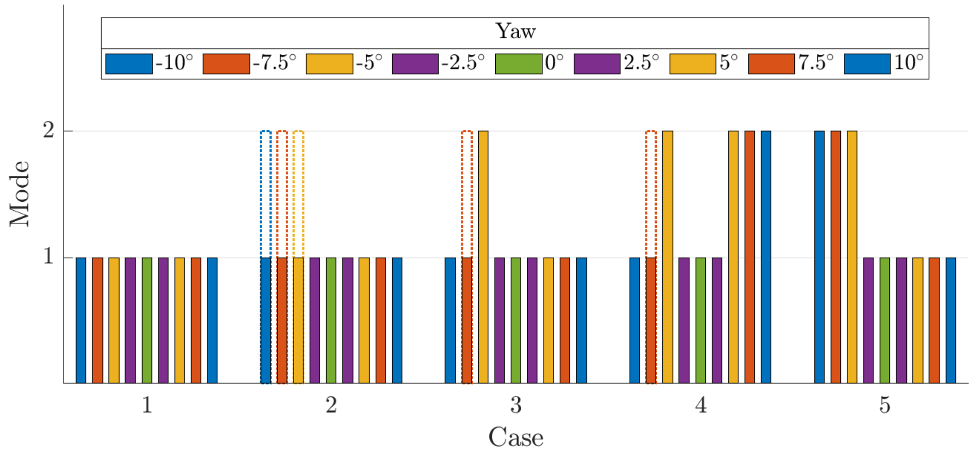


Figure 4.4: *The recreated modes for the chosen cases, from $-10°$ yaw to the left to $+10°$ yaw to the right. The dotted columns in Case 2 signify that different resulting mode was different because of a negligible difference in the mesh. The dotted columns in Case 3 and 4 signify that mode differences appeared between different clusters.*

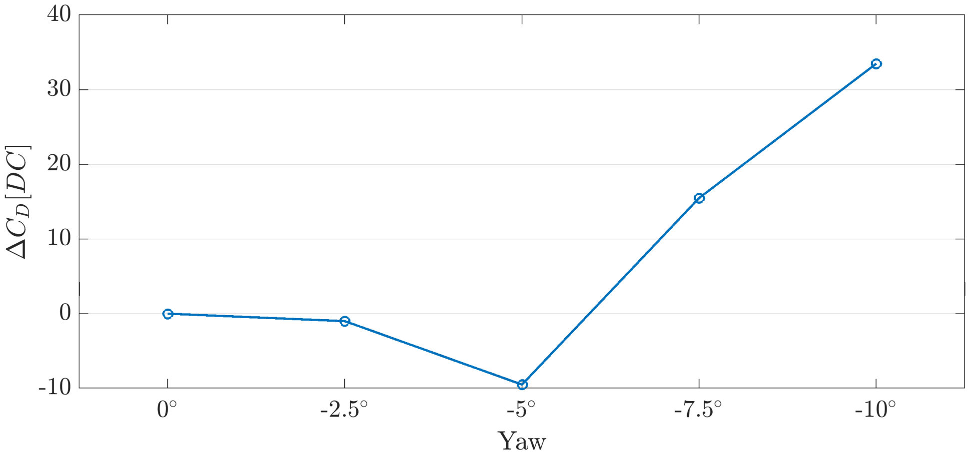


Figure 4.5: *The impact of the modes on C_D. Mode 2 minus Mode 1 C_D estimated by the mean difference in C_D between the negative yaw angles and positive yaw angles for Case 2 and 5.*

very early in the solution process. It is considered possible that Mode 2 is the result of the solution locking into the initial flow state.

In order to avoid the splash effect and maybe avoid erroneously locking into Mode 2, it was tried to gradually increase the inlet velocity in the beginning of the simulation. The results from this are shown in Figure 4.7a. The 100 iteration inlet velocity ramp did not have any effect on the final mode, whereas the 1000 iteration inlet velocity ramp changed the final solution significantly. For both cases an extreme version of Mode 2 was entered. It is assumed that the gradual increase over 1000 iterations interfered with the solution process in some way, and the final results are not considered to be correct. The gradual increase of inlet velocity therefore did not have the intended effect. It was also investigated if changing the inlet velocity would affect the modes. As seen in Figure 4.7a this did however not change the mode for the tested cases.

Using a solution from a lower and a higher yaw angle as initial condition was also tested. The resulting modes from this investigation are shown Figure 4.7b. As can be seen, changing the initial conditions can have an effect on the final mode. However, the 0.0° yaw initial resulted in a new mode, which was mostly similar to Mode 2, but still not the same. It can also be seen that using an initial solution is not effective for selecting

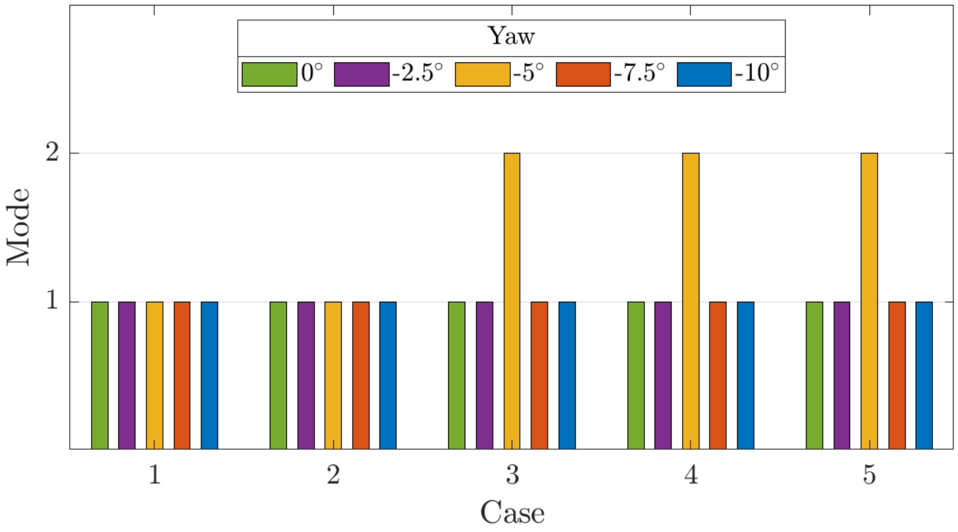
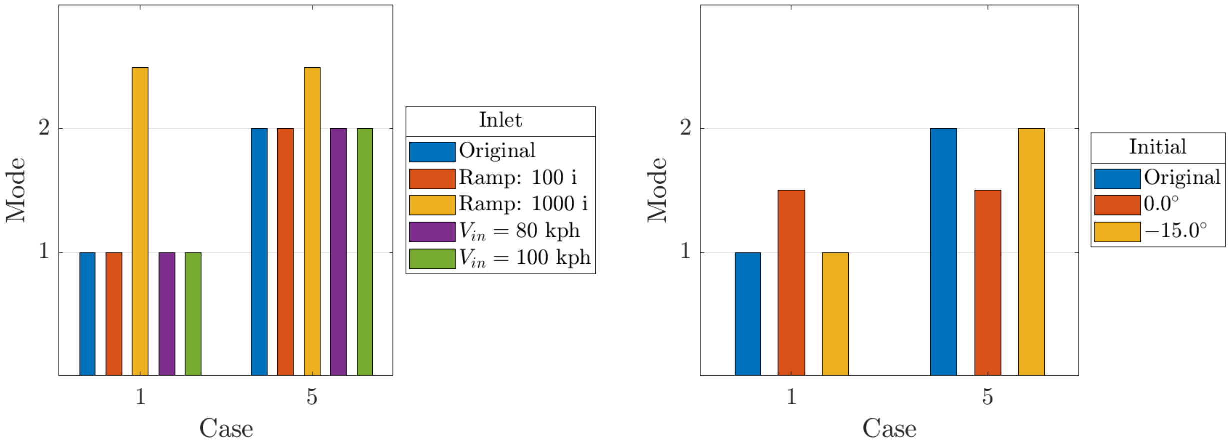


Figure 4.6: *The modes resulting from simulations in Star-2 for the selected cases.*



(a) *Changes to initial conditions.*

(b) *Modes resulting from initial fields from* 0.0° *and* −15.0° *yaw simulations.*

Figure 4.7: *Modes resulting from changes to inlet and initial conditions,* −7.5° *yaw.*

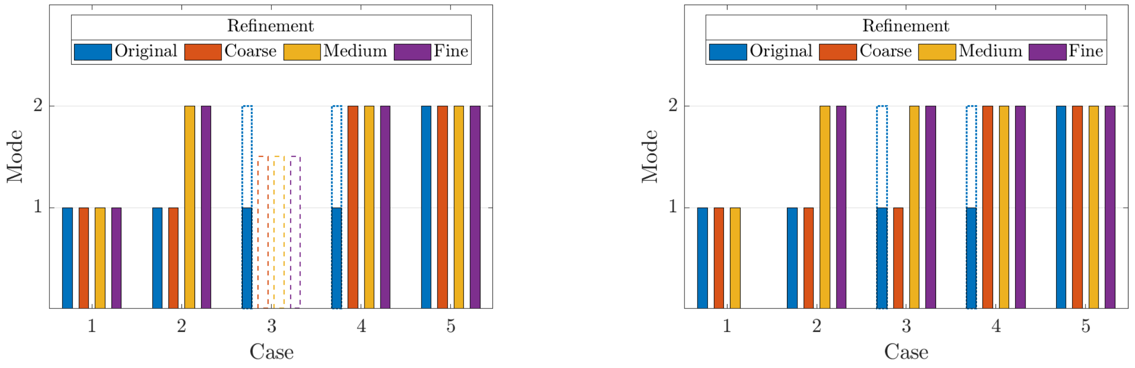
which mode to enter, as the −15° yaw initial resulted in different modes for the tested cases. Because of this, and because of the risk of locking in new, unwanted modes as seen for the 0.0° yaw initial, this solution can not be used.

4.2.3 Effect of Sealing Door Split-line

Sealing the split-line between the door and the front panel proved to eliminate the separation bubble by the mirror foot entirely, and thereby prevents Mode 2 for all cases. This is of course not a solution to the simulation problem, but it further proves that the split-line plays an important role in the phenomenon. It would however be beneficial to seal the split-line in future truck models, to remove ambiguity of results.

4.2.4 Effects of Mesh Refinement

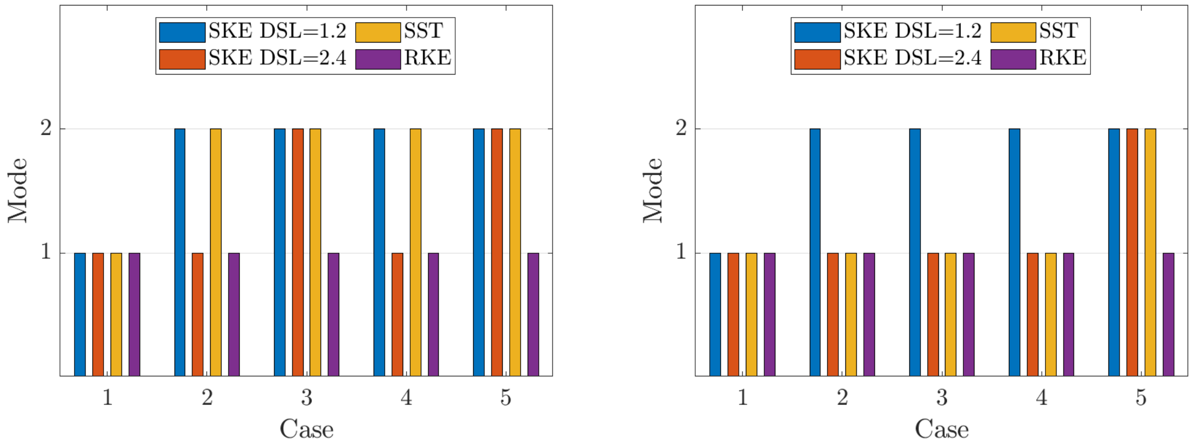
The modes resulting from the mesh study are shown in Figure 4.8. The dotted columns again signify cases where the solution is in such a sensitive position that different modes were entered using different clusters. The dashed columns for the small refinement region signify that those simulations did not converge. The reason for



(a) *Small box refinements. Case 3 becomes unstable with a highly oscillating flow field.*

(b) *Large box refinements. Case 1 fine refinement crashes in meshing.*

Figure 4.8: *The modes resulting from the mesh refinements.*



(a) *Yaw* = $-5.0°$.

(b) *Yaw* = $-7.5°$.

Figure 4.9: *The modes resulting from the turbulence model study.*

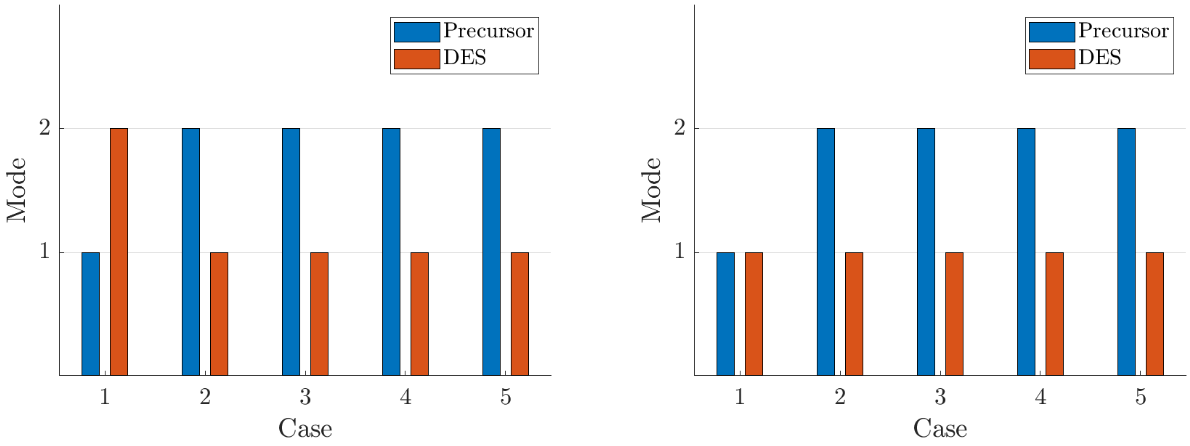
this was not found. Case 1 can be seen to remain in Mode 1 for all refinements, while Case 2 to 5 seem to converge to Mode 2 as the mesh is refined. This shows that the problem is very mesh sensitive, which was also seen in Part 1, Chapter 3. The fact that both modes appear for the finest mesh indicates that neither is caused by discretization error.

4.2.5 Effects of Turbulence Modeling

The modes resulting from the turbulence model study are shown in Figure 4.9. It can be seen that Case 1 remains in Mode 1 for all models. For the SKE model with DSL=1.2, however Mode 2 is entered for all the other cases. Increasing the DSL suppresses the separation in some of the cases, resulting in Mode 1, but not for all cases. This is caused by an increased build-up of TKE just in front of the front panel, suppressing the separation. The SST model predicts Mode 2 for all cases at $-5.0°$ yaw, but only the extreme case for $-7.5°$ yaw. The RKE model only predicts Mode 1.

The fact that only the SKE and the SST models, which is closely related to the SKE, predict Mode 2, whereas RKE does not indicates that Mode 2 is caused by modeling error. Something in the SKE model, that also partly affect the SST model, causes the solution to enter Mode 2.

Because no test data is available, the physicality of the modes cannot be verified. The next best approach is to



(a) *Yaw* = -5.0°.

(b) *Yaw* = -7.5°.

Figure 4.10: *The modes resulting from the DES study, and the mode of the initial field from the precursor simulations.*

compare the RANS results with more accurate simulations. The results of the DES simulations are shown in Figure 4.10, together with the mode of the initial field from the SKE precursor. As can be seen, the initial flow field from the precursor is again in Mode 1 only for Case 1, and Mode 2 for the other cases. As the DES develops, however, the solution switches to Mode 1 for all simulations, except Case 1 -5.0° yaw. Note that the flow field shape of the DES is somewhat different compared to the RANS simulations, but that the modes can still be clearly identified. The DES separation in the region of interest had the general shape of the corresponding RANS mode, but with small high-frequency oscillations about that shape. The longer simulations are Case 1 -5.0° yaw, which was run for 30 s, and Case 5 -7.5° yaw, which was run for 60 s. No mode switches were seen in either of these two cases, but maybe that would have required even longer simulation times.

DES simulations, although more accurate than RANS, are not perfect physical representations, so it is possible that in reality the flow oscillates between Mode 1 and Mode 2, and that the DES simulation locks in on one of the modes. If this was the case, one would however expect that it would lock in on the mode present in the precursor. The fact that this does not happen is further indication that Mode 1 is the physical normal behavior, and that Mode 2 only occurs rarely, or not at all.

The mode preference of the DES may also be caused by the model used in the RANS layer, in this case SST. The SST model was shown to favor Mode 1 for -7.5° yaw and Mode 2 for -5.0° yaw however, so the DES simulations would be expected to predict Mode 2 more for -5.0° yaw. As this is not the case, it seems like the DES is not biased by the RANS layer model employed. It would however be interesting to see how the behavior changed if the RANS layer model was changed.

All in all, judging by the results from the DES simulations, Mode 1 appears to best reflect the true behavior, and Mode 2 does only occur rarely, for lower yaw angles, or not at all. The RANS model that performs best in this case is the RKE model. This is not surprising since RKE is formulated to predict separation better. Changing to the RKE model would however need some more model validation. The SST model performs better than the SKE, and may not be such a big change in practice compared to the SKE. Adjusting the DSL for the SKE may also improve performance, but at the risk of worse prediction in other areas of the flow as the realizability constraint is relaxed. If it can be afforded the best way may however be to use DES simulations, which should improve accuracy in other areas as well. Ideally, wind tunnel tests would be performed to determine the true behavior. However, most wind tunnels too small to test a full-size truck without significantly affecting the flow field. Blockage effects may suppress larger separations, so it is possible that the bi-stability will not appear in a test on a full-size truck. A scale-model can be used to avoid some of the blockage effects, but other problems will appear instead. For example difficulties with the scaling of the velocity and capturing important geometrical details. Because measurements in experiments are limited to probes there may also be difficulties in detecting the flow states, although techniques such as PIV may improve this situation. Considering the challenges of experimentally investigating the flow, it may be advantageous to extend the turbulence model study. Firstly it would be interesting to perform an LES, which has minimal modeling, to get an simulation as accurate as possible. DES with other models in the RANS layer would also be of interest, to avoid any bias from the SST model. Other RANS models may also give some insight into the modeling aspect. For example models with more or fewer equations for the turbulence closure, to significantly change the modeling.

4.3 Conclusions

The conclusion of the second part of this project is that the bi-stable modes are mostly similar, but with a significant difference that appear by the upper front corner. Here Mode 1 has a small separation bubble, which in Mode 2 is much larger. The difference in C_D between the modes is significant and varies with drag. For 5.0° yaw, Mode 2 predicts about 9 DC lower than Mode 1, whereas for 7.5° and 10.0° yaw, Mode 2 predicts about 15 DC and 35 DC higher drag respectively. A part of the explanation to the bi-stability seems to be a strong interaction with front door split-line. A complete explanation for the bi-stability has not been found.

The mesh study shows that the modes are not caused by discretization error. The turbulence model study shows that Mode 1 is most probably physical. The physicality of Mode 2 could not be determined with certainty, as there is indications that it may be caused by modeling error in the SKE model. There is weak support for Mode 2 in the DES study that indicates that Mode 2 is physical, but that Mode 1 definitely is the dominant mode. Since some uncertainty remains as to the physicality of Mode 2, further investigations are recommended, for example LES.

In the meantime the best solution to minimize the bi-stability problem is to use DES if it can be afforded. Otherwise the RKE model is recommended, since it's predictions are most consistent with the DES.

5 Part 3: Effects of Geometrical Variation on Drag

There will always be some variation in the geometry of real trucks. Some of the variation occur in the production of components and in the different stages of assembly of the vehicle. This can appear as variation in split-line gap and flush compared to the nominal geometry.Gap and flush deviation is generally regulated by tolerance requirements, often for functionality, safety and appearance reasons.

There is also reason to suspect that variation in gap and flush may cause variation in drag, because of the role split-lines play in triggering transition to turbulence and separation. CFD simulations are however almost exlusively made with nominal CAD geometry, so the variation in drag is often unknown. This part of the project will therefore investigate the impact of gap and flush variation on drag, both to quantify the effect and to determine if there are some areas where tolerance requirements can be restricted in order to reduce drag.

5.1 Method

An initial investigation of the sensitivity of the drag with respect to the geometry was performed to identify which areas had the greatest impact on drag. This was followed by discussion with geometrical engineers, to learn more about geometrical variation and tolerances. A study visit was also made to the Tuve assembly plant, to learn more about geometrical variation in production. Based on these preparatory investigations, as well as experience from the aerodynamics engineers at Volvo, the tolerance areas to be investigated were chosen. The geometrical changes were implemented in ANSA. An initial investigation was then conducted to estimate the impact on drag. A more thorough DOE investigation was then performed to identify which areas affect drag and how large the effects are.

All simulations in this part of the project were performed in Star-2. The investigations were performed on the truck in Figure 2.3a. Note however that the trailer used in the simulations was longer, with three wheel-pairs, and not the one shown in the figure. All investigations were performed on five yaw angles, from 0.0° to −10.0°.

5.1.1 Adjoint Investigation of Surface Sensitivity

An adjoint analysis was performed using the surface sensitivity model in Star-CCM+, to determine which areas were important for drag. In order to run an adjoint analysis in Star-CCM+ a well converged, coupled, primal solution is required. The coupled implicit solver was used with the standard $k-\varepsilon$ model with coefficients recommended for vehicle aerodynamics. The drag force was set as the adjoint cost function. The settings in Table 5.1 were found to yield a converged adjoint solution, but may not be the most efficient settings. Other settings were also tried out, such as 2nd order discretization, higher CFL numbers, the Restarted GMRES accelerator, fewer Krylov dimensions and fewer correction sweeps were tried out to yield a more accurate solution (discretization) and faster convergence (CFL number and accelerator settings) or lower computational cost (accelerator settings).

When the adjoint solution had converged, a surface sensitivity analysis was run for the boundaries corresponding to external cab surfaces. A 2% (0.02) sensitivity smoothing factor was used to avoid spikes in the result, as recommended by the documentation, [32].

Table 5.1: Adjoint settings.

Discretization	CFL number	Accelerator	Krylov dimensions	Correction sweeps	Iterations
1st order	40	Flexible GMRES	150	10	15

5.1.2 Selection or Areas and Tolerances

Discussions were held with engineers from Geometrical Architecture, Geometrical Assurance and lead engineer for air deflectors, to decide which areas are of interest, what tolerances apply there and what the actual geometrical deviations are. A study visit to Tuve assembly plant was also made to hear the perspective from quality engineering, assembly and audit, and learn more about geometrical deviation in production. The tolerance requirements and measurements of geometrical deviations on real trucks were studied, and the relevant data was extracted. The deviations from the measurements were almost completely within tolerance. The measurements were performed on pre-production vehicles, when tools and processes are still being tuned and practiced. Trucks in ordinary production are therefore expected to deviate even less. Because of this it was decided to use the tolerance requirements for the preliminary study. Based on the engineering experience, surface sensitivity results and the tolerance requirements, the areas to investigate were decided.

The four areas that were decided upon are shown in Figure 5.1 with the selected split-lines marked out. The lower front (LF) and upper front (UF) were chosen because the flow separates in these areas, so they were anticipated to have a large impact on drag by affecting the point of separation. The door split-line was chosen for the same reason, although it is slightly more shielded from the flow. The air deflectors were chosen because of their importance as aerodynamic devices for reducing drag, and because the tolerances are larger for these parts. Because of the large number of split-lines, and because both gap and flush were considered, studying the individual effects of the split-lines would require a much larger number of tests than could be performed in this project. Therefore it was decided to study the effects of the complete areas instead.

5.1.3 Implementation of Geometry Changes in ANSA

The geometry variations were implemented in ANSA version 19.1.1, by moving geometry parts in the CAD file using the "`Transform>>Move`" function. In order to reduce complexity, it was decided to implement the split-line deviations by rigid body translations of the components forming the split-line. Furthermore, all deviations were made symmetrical about the xz-center plane, so that a component on both the left hand and right hand side was moved in a mirrored fashion.

For the preliminary investigation of the impact on drag the transformations were done manually. For the DOE investigation the operations were automated using an ANSA script written in Python for the purpose, which moved the parts to positions determined in an Excel sheet.

5.1.4 Estimation of Impact on Drag

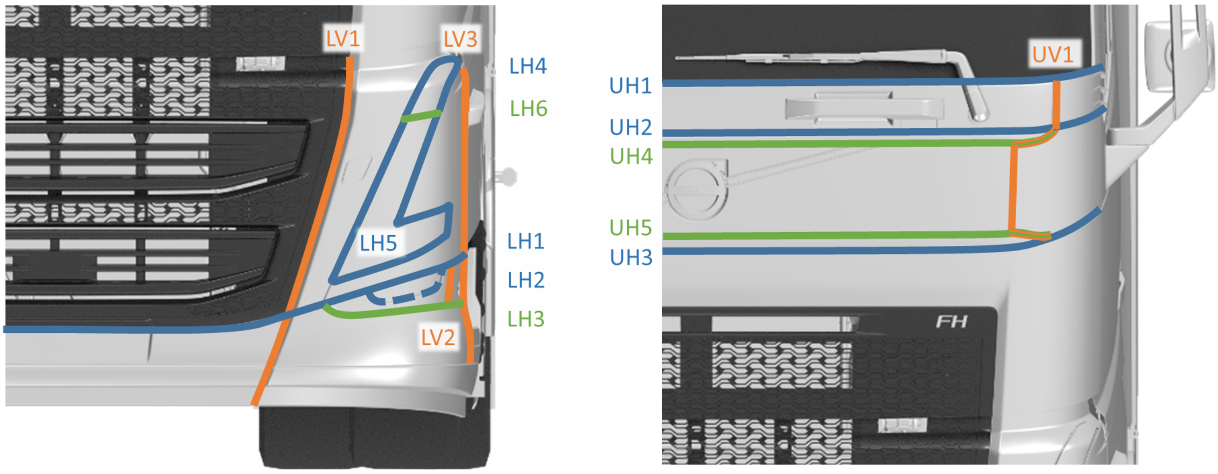
A smaller, preliminary study was performed to get an estimation of how large the impact of tolerance deviations on drag is. If the impact was significant, the following investigation would also use the tolerance deviations. Therefore, for each area a combination of gap and flush deviations was composed to give a large impact on drag. This meant maximum gap and over-flush where possible. Six tests were then performed in which all areas except the listed had nominal positions:

1. LF
2. UF
3. DO
4. AD
5. All areas

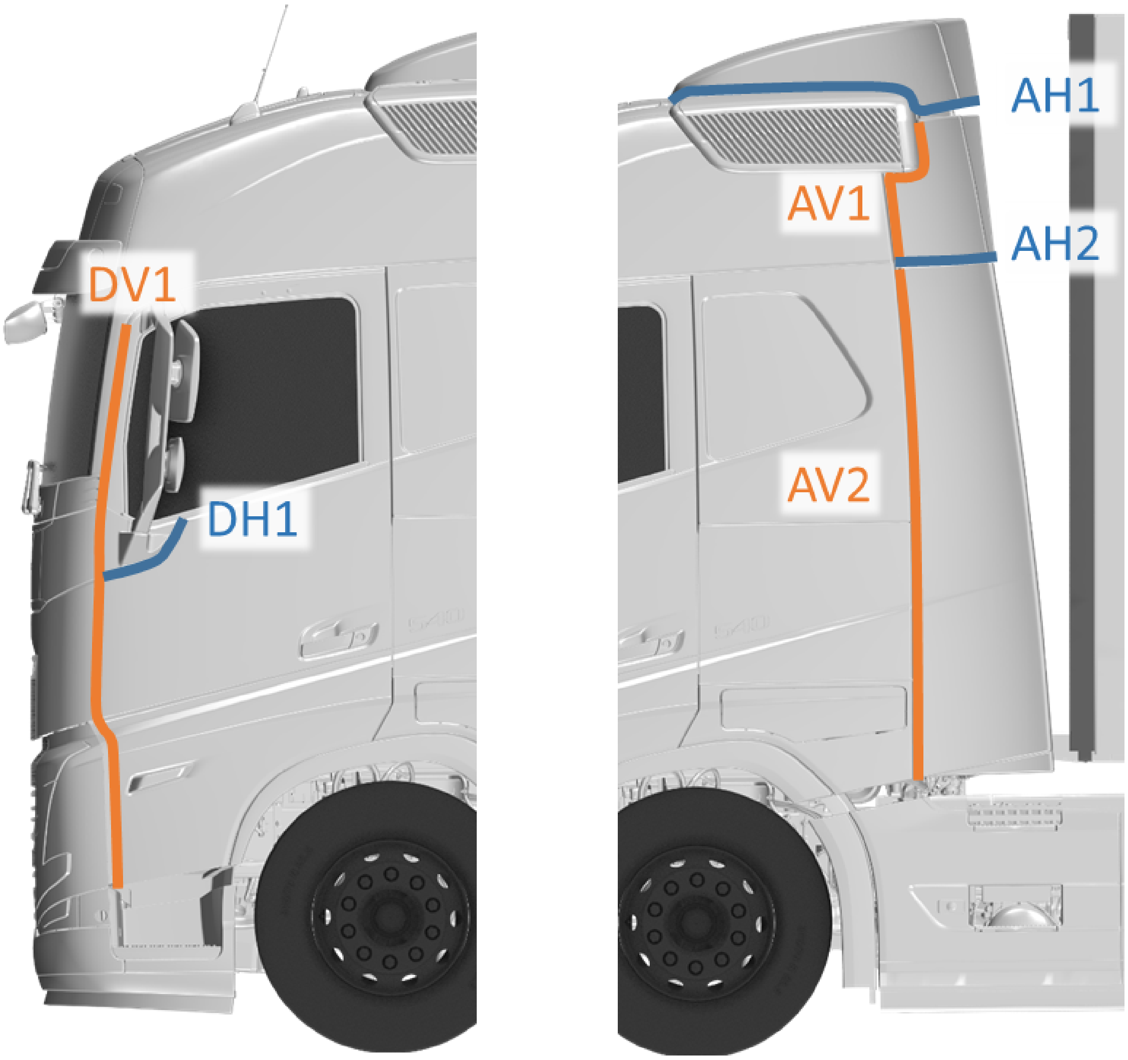
5.1.5 DOE investigation

A factorial design investigation was conducted to investigate the effects of gap and flush variation in the considered areas. The factors considered were therefore gap and flush variation respectively for each area, and a 2^{8-4}_{IV} RFA design with 16 tests was chosen, [18]. The factors that were considered were:

- GL & FL: Gap & Flush lower front



(a) *Lower front split-lines* **(b)** *Upper front split-lines*



(c) *Door split-lines* **(d)** *Air deflector split-lines*

Figure 5.1: *The split-lines of the areas investigated. Vertical split-lines are denoted by "V" and marked in orange; horizontal split-lines are denoted by "H" and marked in blue or green.*

Table 5.2: Design table for the 2^{8-4}_{IV} design. The design was selected from the NIST e-Handbook of Statistical Methods [19, 18].

Test no.	GL 1	FL 2	GU 3	FU 4	GD 5=234	FD 6=134	GA 7=123	FA 8=124
1	-	-	-	-	-	-	-	-
2	+	-	-	-	-	+	+	+
3	-	+	-	-	+	-	+	+
4	+	+	-	-	+	+	-	-
5	-	-	+	-	+	+	+	-
6	+	-	+	-	+	-	-	+
7	-	+	+	-	-	+	-	+
8	+	+	+	-	-	-	+	-
9	-	-	-	+	+	+	-	+
10	+	-	-	+	+	-	+	-
11	-	+	-	+	-	+	+	-
12	+	+	-	+	-	-	-	+
13	-	-	+	+	-	-	+	+
14	+	-	+	+	-	+	-	-
15	-	+	+	+	+	-	-	-
16	+	+	+	+	+	+	+	+

- GU & FU: Gap & Flush upper front
- GD & FD: Gap & Flush door
- GA & FA: Gap & Flush air deflectors

In the chosen design 2-factor interactions were confounded with other 2-factor interactions, in groups of four. For example GL×FL is confounded with GU×GA, FU×FA and GD×FD. In order to avoid confounding of 2-factor interactions a 64 test design would be required, which could not be afforded.

Because the preliminary study did show a significant impact on drag for tolerance based deviations, the levels of this investigation were also based on the tolerance values. The levels for the gap factors were therefore chosen to be maximum and minimum possible gap according to tolerances. The levels for the flush factors were chosen to be maximum possible over-flush and under-flush according to tolerances. Because of the implementation by rigid body motion, some combinations of maximum or minimum gap and flush for different split lines were not possible to achieve while adhering to the specified tolerance values. For example, the horizontal split-lines in the upper front cannot all be over-flush, or the flush deviation of the UH1 split-line will be too large. Instead, some split-lines were prioritized. For the upper front factors alternative combinations were composed. The split-lines that were not controlled directly are marked in green in Figure 5.1.

5.2 Results and Discussion

Here the results are presented and discussed.

5.2.1 Surface sensitivity

The surface sensitivity analysis did show that no split-line areas are particularly sensitive to drag at 0.0° yaw. As the yaw angle increases, however, so does the surface sensitivity. The shape of the lee-side corner has great impact on drag. Particularly the lower front area, but the outer part of the upper front area and the front part of the door also impacts drag significantly. The air deflectors have a moderate impact, mainly on the windward side-deflector and on the top of the roof deflector.

It is expected that the front corner has a large impact on drag, because the flow separates here. The velocity is also high and there are a lot of surface roughness in the form of split-lines, so that friction drag should be high also. The door is more shielded from the flow, so it is expected to give a lower contribution. Also the air deflectors are more shielded, except for the windward side-deflector in yawed conditions and the roof deflector. The results of this analysis therefore supports the selection of the investigated areas.

5.2.2 Estimation of Impact on Drag

The impact on C_D from the preliminary tests are shown in Figure 5.2, as the difference in C_D compared to the baseline. Here it can be seen that the impact on drag is dependent on both area and yaw angle. The areas with the largest impact are the lower front and the upper front, both with an YWA impact of about +3 DC. Both the door and the air deflectors have a smaller impact at less than +1 DC. If all areas are modified, the impact is larger, at about +5 DC. The larger impact is expected, because the flow is disturbed in more areas.

5.2.3 DOE Results

In Figure 5.3 the impact on C_D for the DOE investigation is shown. The impact can be seen to be almost centered about the baseline value, ranging from circa -0.1 DC for 0.0° yaw to $+1.4$ DC at $-10.0°$ yaw. This results in a small bias towards increased drag, by about 0.4 DC for the YWA. The total variation is similar for all angles, about ±3 DC for YWA, which is significant.

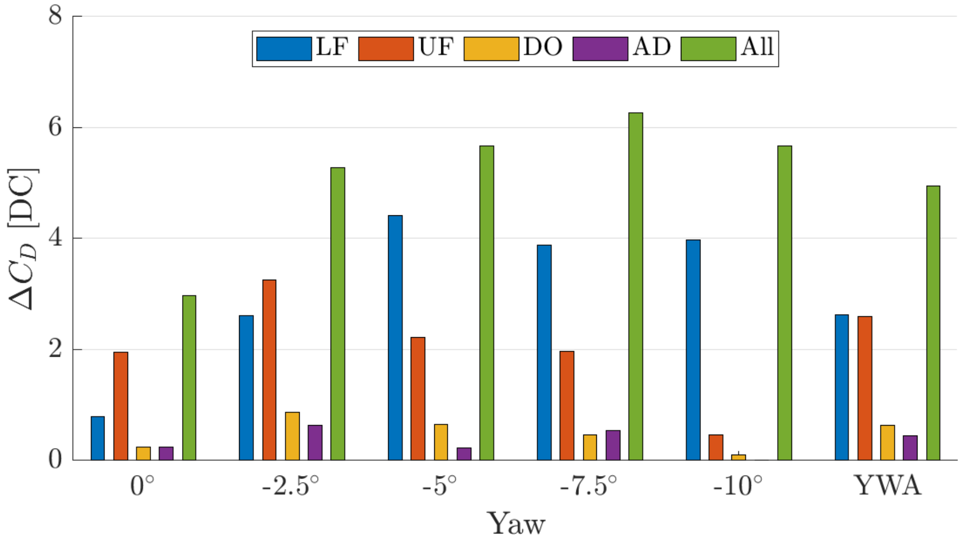


Figure 5.2: *The impact on drag in the preliminary investigation, measured as the difference in C_D from the baseline for each experiment and yaw angle.*

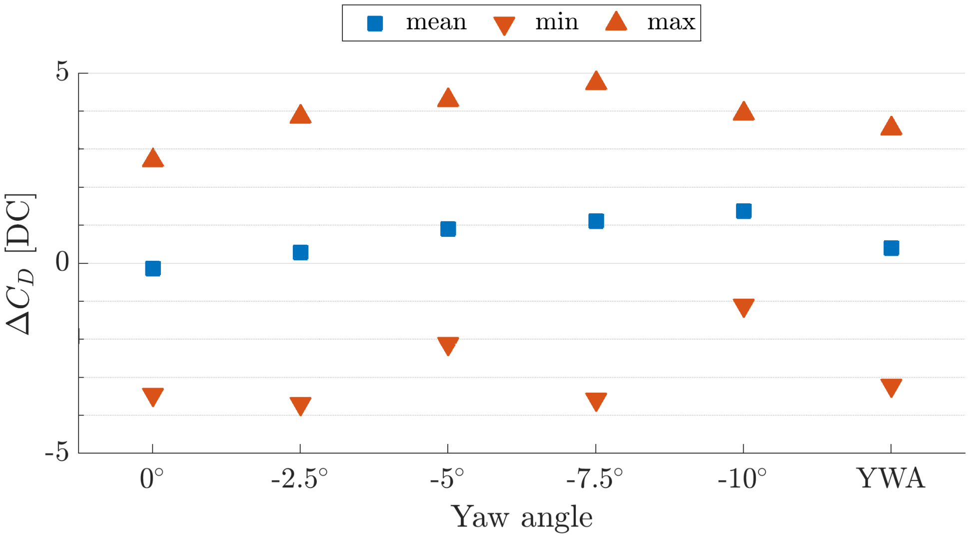


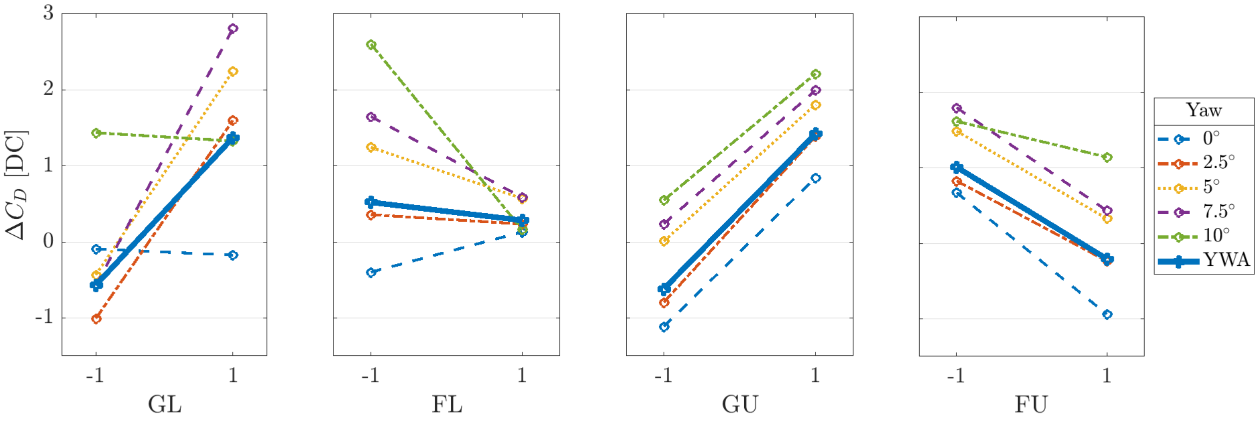
Figure 5.3: *The impact on drag for the DOE investigation, measured by the difference in C_D from the baseline. The mean, minimum and maximum impact of all tests for each yaw angle.*

As seen from the results in the previous section, the impact on drag can be larger than ±3 DC. A value of +5 DC was reported. This study furthermore only covered split-line variation in some areas, so the impact could be larger if more areas are included. Based on engineering experience and the surface sensitivity study, the most important areas were selected however, so the added impact should not be too large. Furthermore, most trucks are expected to be within the tolerance requirements, so it is very unlikely that any truck should have deviations at either minimum or maximum tolerances for all split-lines. Therefore it is expected that most trucks will be within the ±3 DC bounds, but to cover extreme cases a more conservative limit of ±5 DC could also be considered.

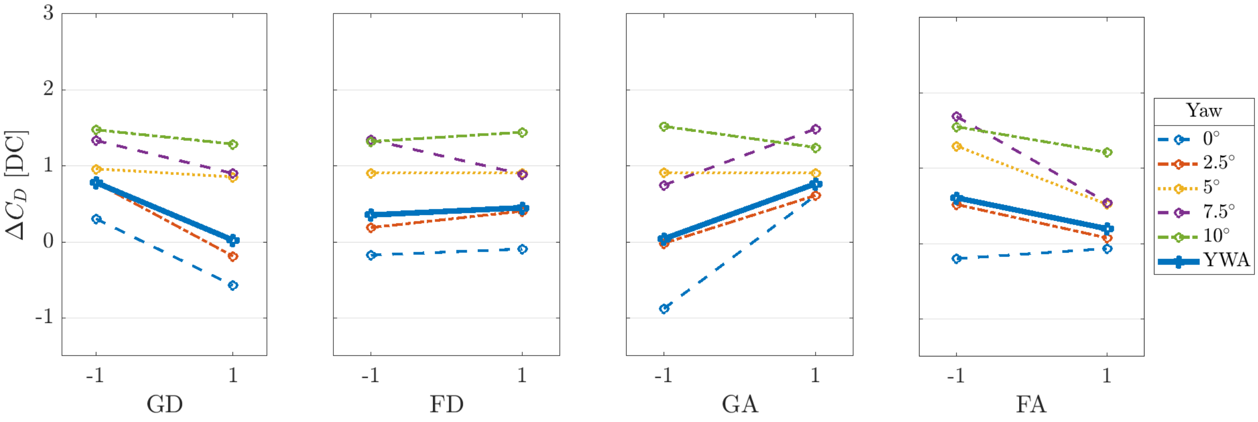
The main effects of the factors in the DOE, gap and flush for the considered areas, are shown in Figure 5.4. The most important factors are lower front gap, Figure 5.4a and upper front gap, Figure 5.4c. Both can be seen to go from -0.5 YWA DC for minimum gap to 1.5 YWA DC at maximum gap, a 2 DC difference. For the lower front it is the medium yaw angles that increase drag the most, while for upper front all yaw angles contribute to the increase. The next most important effect is upper front flush, shown in Figure 5.4d, which shows a decrease from 1 DC to about -0.2 DC going from the low to the high level. Door gap also shows a decrease, Figure 5.4e, from 0.8 to 0 DC going from minimum to maximum gap, whereas air deflector gap goes from 0 to 0.7 DC, shown in Figure 5.4g. All other factors, lower front flush in Figure 5.4b, door flush in Figure 5.4f and air deflector flush in Figure 5.4h, show negligible effects on YWA C_D.

The main effects results show that the most important areas are the upper and lower front. This is expected, and supported by the results of the previous investigations. In the front corner the air velocity is high, making this region more sensitive. There are also more split-lines interfering with the flow, causing disturbances that can affect turbulence and separation. Furthermore gap is shown to have a greater impact than flush. This was not entirely expected, because flush are generally thought to upset the flow more than gap. This may however be explained by the fact that increased gap causes more leakage behind the exterior component, which may cause more disturbance.

Interaction effects between factors could not be independently determined in this investigation, because the design had too few tests. The combined effects can however be determined, to set a bound for the effect of the interactions. This is shown in Figure 5.5. As can be seen, the combined effect of $\mathrm{GL}\times\mathrm{FL}+\mathrm{GU}\times\mathrm{GA}+\mathrm{FU}\times\mathrm{FA}+\mathrm{GD}\times\mathrm{FD}$ is -1.2 DC. The individual interaction effects of any of these will therefore be less than this, which is still less than the most important main effect. The next largest combined interaction effect is 0.3 drag counts, which is insignificant in comparison with the more important effects. No interaction effects are therefore more important



(a) *Lower front gap* **(b)** *LF Flush* **(c)** *UF Gap* **(d)** *UF Flush*



(e) *DO Gap* **(f)** *DO Flush* **(g)** *AD Gap* **(h)** *Air deflectors flush*

Figure 5.4: *Effects of the factors in the DOE investigation. The average value of ΔCD plotted for the high and the low level of the factor. The difference is the main effect.*

than the most important main effects.

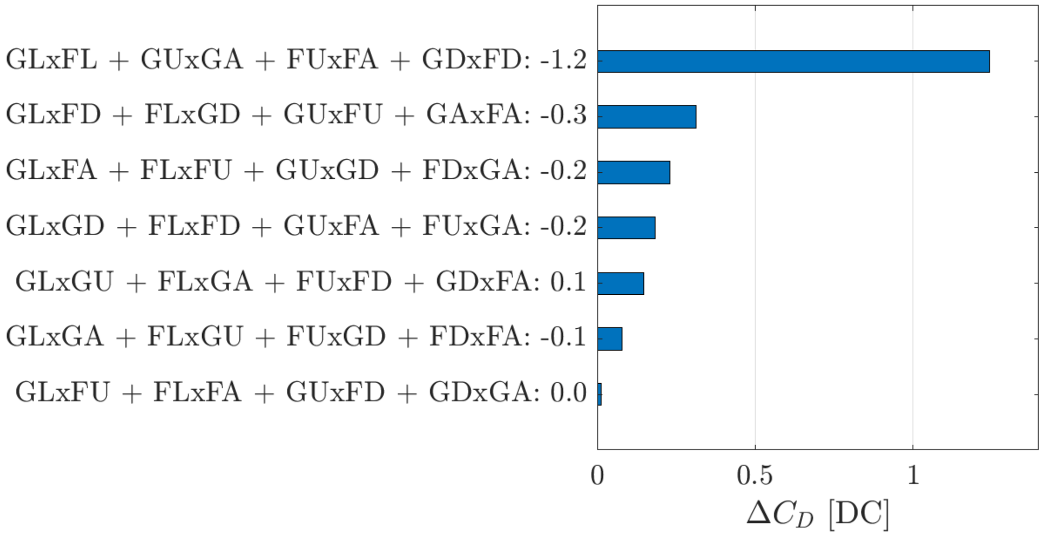


Figure 5.5: *The 2-factor interaction effects of the DOE investigation, ordered by magnitude. Because of confounding in the design no 2-factor effect can be determined independently. For example, the largest effect here is the combined effect of the interactions* $GL\times FL$,$GU\times GA$,$FU\times FA$,$GD\times FD$, *that is the sum of their individual contributions, as shown by the y-axis labels.*

5.3 Conclusions

The variation of drag caused by split-line variation was seen to be almost centered about the baseline value, and for most cases to be in the ±3 DC range. For more extreme cases the impact was seen to be larger however, so a conservative limit of ±5 can also be considered. Because most trucks are expected to be within tolerance this limit should cover the majority of the produced vehicles.

The areas that are most important for drag are the upper and lower front corner, where the effect is an increase of 2 DC between minimum gap and maximum gap. The door and air deflectors are less important. Gap was found to be more important than flush in all investigated areas.

6 Conclusion

The problem with differences in steady simulation results because of parallelization was shown to be caused by an underlying instability in the solution, combined with the AMG-solver not being run to convergence at each solver iteration. The solution to this is using a constant order for MPI reduction operations, to avoid differences between clusters, and to use a standard number of processors for simulations. The best option is however to address the underlying instability.

The underlying instability appears in the shape of two bi-stable modes, that are mostly similar, but with a significant difference in the separation shape by the front corner. This difference in separation also produces a significant difference in drag, which grows with the yaw angle. Neither Mode 1 or Mode 2 was found to be caused by discretization error, but Mode 2 could possibly be caused by modeling error from the SKE turbulence model. The fact that Mode 2 appears in one out of 10 DES simulations complicates this conclusion, indicating that Mode 2 may be physical. In any case there is considered to be sufficient evidence to conclude that Mode 1 is physical and the dominant mode. If Mode 2 should be physical, as indicated by the DES simulations, it seems to appear only rarely. More testing is needed to determine the physicality of Mode 2 with greater certainty. With the current information it is recommended to use DES it if can be afforded, as it is a more accurate model. If not the RKE model is recommended, as it is most consistent with the DES predictions among the tested RANS models.

The variation on drag caused by geometrical variation of split-lines was also investigated. It was found that the maximum bound on effects on C_D was ±5 DC, and that most trucks in production likely will fall within ±3 DC from the nominal value. The areas with greatest effect are split-line gap on the upper and lower front corner and split-line flush on the upper front corner. To decrease drag it is therefore recommended to decrease gap in these areas and make the upper front panels as flush as possible.

6.1 Future Work

Because the differences in results that appeared for some parallel simulations was caused by an underlying problem in the simulation cases, it is considered that no further investigations are necessary for this issue.

Regarding bi-stable flow states, it would of course be best if the correct flow behavior could be determined experimentally. There are however some challenges that must be considered for such a experiment. For example the possibility of suppression of separation due to blockage effects, and difficulties in detecting different flow states. Therefore it would also be interesting to investigate some more turbulence models. For example LES, with minimal modeling, DES with different RANS layer models, to avoid bias from the SST model. It may also be interesting to test other RANS models, that are more different than the SKE and RKE models.

On a more general note it should be investigated further how to predict and represent bi-stability in steady simulations. Vehicle aerodynamics are clearly subject to bi-stabilities of different scales. Because of the high computational cost required to capture especially the longer scales in transient simulations, a method for predicting such behavior in steady simulations may be attractive. This is of course no easy task, and toes the border between transient and steady simulations.

Regarding the effect of geometrical variation on flow properties, geometrical variation may appear in many other areas than in split-lines. For example deformed parts, such as warped and skewed body panels may affect the flow by changing the curvature of the surface and causing more leakage. Different mirror settings, roof deflector settings, riding height may also have significant effects. It is furthermore popular among some truck drivers to fit their trucks with exterior ornamentation. It could be of interest to investigate how much that affects drag, and in extension fuel consumption and CO_2 emissions.

References

[1] R. H. Barnard. *Road Vehicle Aerodynamic Design: An Introduction.* 3rd ed. St Albans, UK: MechAero Publishing, 2009. ISBN: 9 780954 073473.

[2] T. J. Barth. "A Brief Introduction to the Design and Implementation of Parallel Numerical Simulation Codes". *Lecture Notes in Computer Science.* Vector and Parallel Processing – VECPAR'98 (Porto, 1998). Ed. by V. Hernández, J. M. L. M. Palma, and J. Dongarra. Vol. 1573. Berlin: Springer, 1999, pp. 171–175. ISBN: 978-3-540-48516-2. DOI: `https://doi.org/10.1007/10703040_15`.

[3] T. J. Barth, T. F. Chan, and W.-P. Tang. "Parallel Domain-Decomposition Preconditioning for Computational Fluid Dynamics". *Lecture Notes in Computer Science.* Vector and Parallel Processing – VECPAR'98 (Porto, 1998). Ed. by V. Hernández, J. M. L. M. Palma, and J. Dongarra. Vol. 1573. Berlin: Springer, 1999, pp. 176–202. ISBN: 978-3-540-48516-2. DOI: `https://doi.org/10.1007/10703040_15`.

[4] P. K. Chang. *Separation of Flow.* 8th ed. Oxford, UK: Pergamon, 1970. ISBN: 0080134416.

[5] L. Davidson. *Fluid mechanics, turbulent flow and turbulence modeling.* Gothenburg, Sweden: Department of Mechanics and Maritime Sciences, Chalmers University of Technology, 2020. URL: `http://www.tfd.chalmers.se/~lada/postscript_files/solids-and-fluids_turbulent-flow_turbulence-modelling.pdf` (visited on 04/17/2020).

[6] P. A. Durbin. On the k-3 stagnation point anomaly. *International Journal of Heat and Fluid Flow* **17**.1 (1996), 89–90. DOI: `https://doi.org/10.1016/0142-727X(95)00073-Y`.

[7] European Commision. *Reducing CO2 emissions from heavy-duty vehicles.* URL: `https://ec.europa.eu/clima/policies/transport/vehicles/heavy_en` (visited on 05/04/2020).

[8] J. H. Ferziger and M. Perić. *Computational Methods for Fluid Dynamics.* 3rd ed. Berilin: Springer, 2002. ISBN: 978-3-540-42074-3. DOI: `https://doi.org/10.1007/978-3-642-56026-2`.

[9] M. Grandemange, M. Gohlke, and O. Cadot. Turbulent wake past a three-dimensional blunt body. Part 1. Global modes and bi-stability. *Journal of Fluid Mechanics* **722** (2013), 51–84. DOI: `https://doi.org/10.1017/jfm.2013.83`.

[10] G. Hager and G. Wellein. *Introduction to High Performance Computing for Scientists and Engineers.* Boca Raton, FL: CRC Press, 2011. ISBN: 978-1-4398-1192-4.

[11] W.-H. Hucho, ed. *Aerodynamics of road vehicles: from fluid mechanics to vehicle engineering.* 1st ed. London, UK: Butterworth-Heinemann, 1987. ISBN: 0-408-01422-9.

[12] G. K. Kenway et al. Effective adjoint approaches for computational fluid dynamics. *Progress in Aerospace Sciences* **110** (2019). DOI: `https://doi.org/10.1016/j.paerosci.2019.05.002`.

[13] E. Ljungskog. *Siemens STAR-CCM+ support.* Private communication. Siemens PLM Software, Apr. 28, 2020.

[14] D. McArthur et al. An experimental characterisation of the wake of a detailed heavy vehicle in cross-wind. *Journal of Wind Engineering & Industrial Aerodynamics* **175** (1996), 364–375. DOI: `https://doi.org/10.1016/j.jweia.2018.01.033`.

[15] W. Meile et al. Non-symmetric bi-stable flow around the Ahmed body. *International Journal of Heat and Fluid Flow* **57** (2016), 34–47. DOI: `https://doi.org/10.1016/j.ijheatfluidflow.2015.11.002`.

[16] D. C. Montgomery. *Design and Analysis of Experiments.* 5th ed. New York, NY, USA: John Wiley & Sons, 2001. ISBN: 0-471-31649-0.

[17] NIST. *NIST/SEMATECH e-Handbook of Statistical Methods: 2**(5-1) FRACTIONAL FACTORIAL DESIGN.* 2013. URL: `https://www.itl.nist.gov/div898/handbook/pri/section3/eqns/2to5m1.txt` (visited on 04/23/2020).

[18] NIST. *NIST/SEMATECH e-Handbook of Statistical Methods: 2**(8-4) FRACTIONAL FACTORIAL DESIGN.* 2013. URL: `https://www.itl.nist.gov/div898/handbook/pri/section3/eqns/2to8m4.txt` (visited on 04/23/2020).

[19] NIST. *NIST/SEMATECH e-Handbook of Statistical Methods: Summary tables of useful fractional factorial designs.* 2013. URL: `https://www.itl.nist.gov/div898/handbook/pri/section3/pri3347.htm#TABLE` (visited on 04/23/2020).

[20] G. Pavia, M. Passmore, and C. Sardu. Evolution of the bi-stable wake of a square-back automotive shape. *Experiments in Fluids* **59** (2018). DOI: `https://doi.org/10.1007/s00348-017-2473-0`.

[21] A.-K. Perry, M. Almond, and M. Passmore. The Study of a Bi-Stable Wake Region of a Generic Squareback Vehicle using Tomographic PIV. *SAE International Journal of Passenger Cars - Mechanical Systems* **9**.2 (2016), 743–753. DOI: `https://doi.org/10.4271/2016-01-1610`.

[22] A. N. Rao et al. An LES Investigation of the Near-Wake Flow Topology of a Simplified Heavy Vehicle. *Flow, Turbulence and Combustion* **102** (2019), 389–415. DOI: `https://doi.org/10.1007/s10494-018-9959-6`.

[23] C. Rumsey. *The Menter Shear Stress Transport Turbulence Model.* NASA. 2015. URL: `https://turbmodels.larc.nasa.gov/sst.html` (visited on 04/20/2020).

[24] Y. Saad. *Iterative Methods for Sparse Linear Systems.* 2nd ed. Society for Industrial and Applied Mathematics, 2003. ISBN: 978-0-89871-534-7. DOI: `https://doi.org/10.1137/1.9780898718003`.

[25] J. Sacks et al. Design and Analysis of Computer Experiments. *Statistical Science* **4**.4 (1989), 409–423. URL: `https://www.jstor.org/stable/2245858`.

[26] T.-H. Shih et al. *A New K-epsilon Eddy Viscosity Model for High Reynolds Number Turbulent Flows: Model Development and Validation.* Technical report. NASA, 1994. URL: `https://ntrs.nasa.gov/search.jsp?R=19950005029`.

[27] Siemens PLM Software. *Best Practices for Vehicle Aerodynamics.* 2020. URL: `https://thesteveportal.plm.automation.siemens.com/articles/en_US/FAQ/Best-Practices-for-Vehicle-Aerodynamics`.

[28] Siemens PLM Software. *How do the number of cores I use to run a simulation influence results and especially acoustics results.* 2018. URL: `https://thesteveportal.plm.automation.siemens.com/articles/en_US/FAQ/How-do-the-number-of-cores-I-use-to-run-a-simulation-influence-results-and-especially-acoustics-results`.

[29] Siemens PLM Software. *Spotlight on... Adjoint Method.* 2020. URL: `https://thesteveportal.plm.automation.siemens.com/articles/en_US/FAQ/Spotlight-on-Adjoint`.

[30] Siemens PLM Software. *Why do I get different cell count with parallel meshing?* 2017. URL: `https://thesteveportal.plm.automation.siemens.com/articles/en_US/FAQ/Why-do-I-get-different-cell-count-with-parallel-meshing`.

[31] Siemens PLM Software. *Why do I get different results depending on the number of cores used in parallel?* 2017. URL: `https://thesteveportal.plm.automation.siemens.com/articles/en_US/FAQ/Why-do-I-get-different-results-depending-on-the-number-of-processors-the-case-is-run-on`.

[32] *Simcenter STAR-CCM+ User Guide.* Version 2019.2 (14.02). Siemens PLM Software, 2019.

[33] J. W. Slater. *Uncertainty and Error in CFD Simulations.* NASA. 2008. URL: `https://www.grc.nasa.gov/WWW/wind/valid/tutorial/errors.html` (visited on 05/02/2020).

[34] H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method.* 2nd ed. Harlow, UK: Pearson Education, 2007. ISBN: 978-0-13-127498-3.

[35] F. M. White. *Fluid Mechanics: In SI Units.* 8th ed. New York, NY, USA: McGraw-Hill Education, 2016. ISBN: 978-9-814-72017-5.