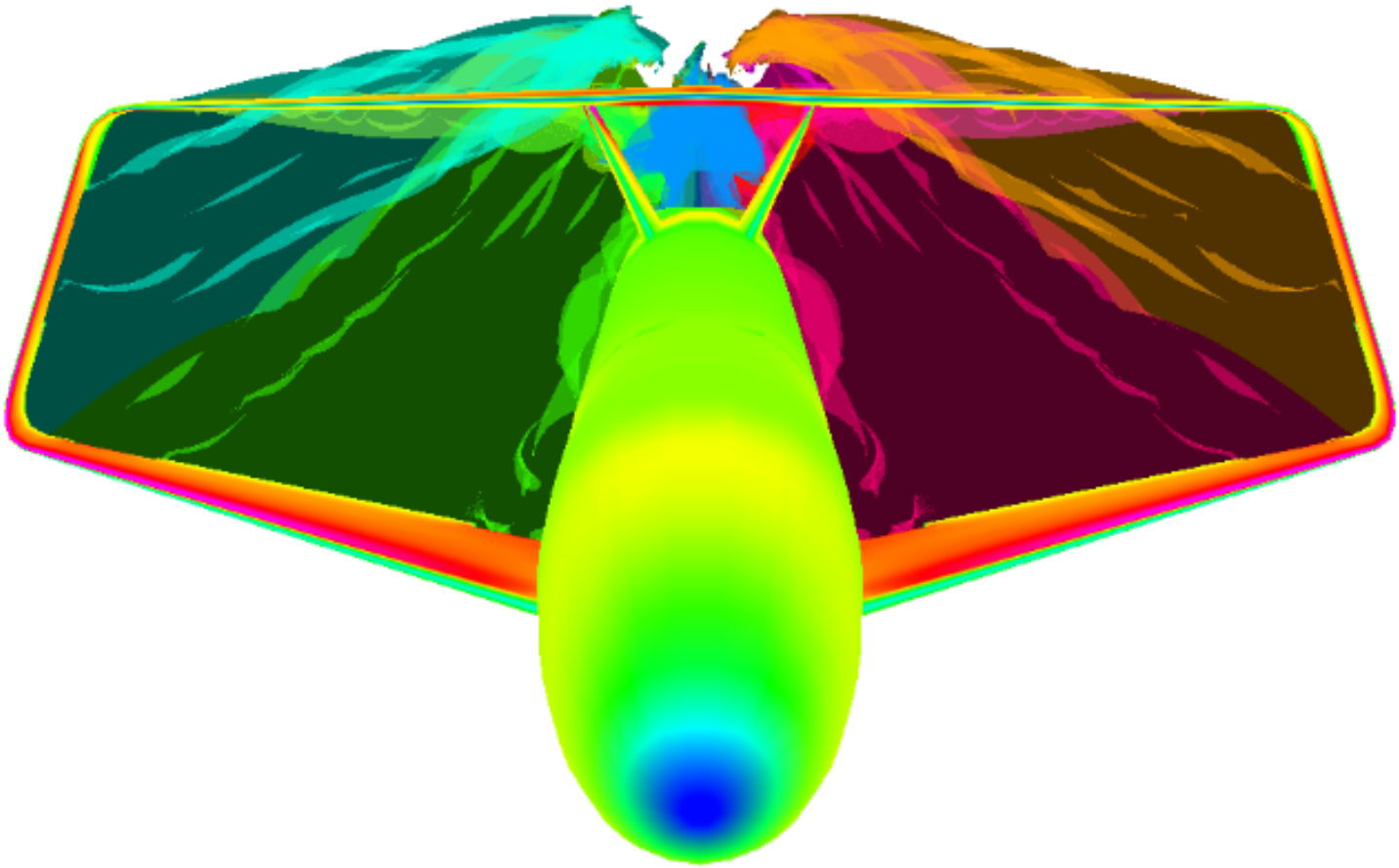
Stability & control derivatives prediction for box wing aircraft configurations

Roy Groot



Technische Universiteit Delft

TU Delft

Stability & control derivatives prediction for box wing aircraft configurations

by

Roy Groot

to obtain the degree of Master of Science in Aerospace Engineering
at the Delft University of Technology,
to be defended publicly on Tuesday October 8, 2019 at 10:00 AM.

Student number:	4097971	
Thesis committee:	Prof. dr. ir. L. L. M. Veldhuis,	TU Delft
	Dr. ir. G. La Roccca,	TU Delft, supervisor
	ir. A. Raju Kulkarni,	TU Delft, daily supervisor
	Dr. S. J. Hulshoff,	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Summary

The number of passengers using air traffic is expected to double by 2036. As a consequence, air traffic and its associated environmental impact will increase tremendously. However, the International Air Transport Association (IATA) is pushing for a 50% CO_2 emission reduction in net aviation before 2050, relative to the levels in 2005. The aviation industry is trying to cope with this issue by improving the engine efficiency. Nevertheless, these improvements will not be sufficient to meet the emission goals. Therefore, the aircraft industry is looking at unconventional aircraft configurations for more drastic emission cuts.

One of these unconventional aircraft configuration is the box wing aircraft, which features two main wings with a wing element connecting the two wings at the tips. This concept promises a lower induced drag as compared to conventional aircraft configurations, which in turn reduces fuel consumption and greenhouse emissions.

The challenge of implementing unconventional aircraft configurations, like the box wing aircraft, is that in early design stages conventional semi empirical design methods do not apply. This can have two reasons. First, unconventional aircraft lack the necessary empirical data for the methods. Second, the methods are not applicable to unconventional aircraft, because required geometrical input parameters of the methods do not exist. This increases the uncertainty of the performances of the generated design, such as its stability and controllability. Therefore, more physics based methods are required, but they tend to have long lead times compared to semi empirical methods.

A methodology is developed which supports prediction of stability and control derivatives accurately and quickly, to support flight mechanics analysis for box wing aircraft configurations using physics based methods. This methodology uses VSAero to compute the aerodynamic forces acting on the aircraft. VSAero is more accurate than a vortex lattice solver, but has lower processing time than using a finite element solver. VSAero is integrated in an in-house developed Multi Model Generator, as to further decrease preprocessing and post processing time. Before the stability and control derivatives can be computed, the trim point is calculated using an in-house non-linear flight dynamics analysis tool.

The tool is verified and validated by modelling, and analysing two different geometries in the Multi Model Generator, and comparing the results with wind tunnel test data. This showed that VSAero predicts the stability and control derivatives at least 22% more accurate for the tested aircraft models, than a Vortex Lattice method. A time study is done, from which it can be concluded that using the developed methodology decreases processing time by 96% compared to manual use of VSAero.

3 different aircraft configurations, having the same top level aircraft requirements, are processed and analysed (2 box wing aircraft and one conventional), to compute their stability and control derivatives. From this comparison it can be concluded that controllability problems can be expected for the box wing aircraft with respect to the longitudinal controllability. The large rear wing creates a large pitch down moment, which can cause problems when trying to control the pitch behaviour of the aircraft.

In conclusion, a quick and accurate method is developed that is capable of predicting stability and control derivatives using physics based methods. This opens the door for performing flight mechanics analysis on box wing aircraft configurations, which brings us one step closer to the use of boxing aircraft in commercial operations.

Preface

This thesis marks the end point as student at the TU Delft. It has been a long road and I would like to thank the people who helped me along the way.

First of all, I would like to thank my daily supervisor Akshay Raju Kulkarni for guiding me in my thesis, and helping me with many code debug sessions over the thesis period. Furthermore, I would like to thank Gianfranco La Rocca for his support and feedback on my work.

I want to thank my dad, Ko Groot, who gave me the opportunity to pursue an academic degree, and his continuous support along the way. Also I would like to thank Tara Moon Christopher, for her support and feedback while writing this report.

Special thanks go out to Anthony Mosquera and Daniel Hung for providing assistance with VSAero.

Roy Groot
Delft, September 24, 2019

Contents

Nomenclature

Roman symbols

b	=	span [m]
$\bar{c}$	=	Mean aerodynamic chord [m]
C_D	=	Aircraft drag coefficient[-]
C_L	=	Aircraft lift coefficient [-]
C_l	=	Roll moment coefficient [-]
C_m	=	Pitch moment coefficient [-]
C_n	=	Yaw moment coefficient [-]
C_X	=	Force coefficient in x direction [-]
C_Y	=	Force coefficient in y direction [-]
C_Z	=	Force coefficient in z direction [-]
p	=	Rotational velocity about x-axis [rad/s]
q	=	Rotational velocity about y-axis [rad/s]
r	=	Rotational velocity about z-axis [rad/s]
Re	=	Reynolds number [-]
S	=	Surface area [m^2]
V	=	Velocity [m/s]

Greek symbols

α	=	Angle of attack [deg]
β	=	Angle of sideslip [deg]
δ_a	=	Aileron deflection [deg]
δ_e	=	Elevator deflection [deg]
δ_r	=	Rudder deflection [deg]
ϕ	=	Velocity potential [-]

Subscripts

α	=	Derivatives w.r.t.angle of attack [1/rad]
β	=	Derivatives w.r.t.sideslip [1/rad]
p	=	Derivatives w.r.t. the angular velocity about X-axis [s/rad]
q	=	Derivatives w.r.t. the angular velocity about y-axis [s/rad]
r	=	Derivatives w.r.t. the angular velocity about z-axis [s/rad]
u	=	Velocity in the x direction of the body reference frame [m/s]
δ_a	=	Derivatives w.r.t. aileron deflection [1/rad]
δ_e	=	Derivatives w.r.t. elevator deflection [1/rad]
δ_r	=	Derivatives w.r.t. rudder deflection [1/rad]

Abbreviations

AVL	=	Athena Vortex lattice
CG	=	Center of gravity
CFD	=	Computational fluid dynamics
CPACS	=	Common parametric aircraft configuration schema
CRM	=	Common research model
DLR	=	Deutsches Zentrum für Luft- und Raumfahrt
DEE	=	Design and engineering engine
GUI	=	Graphical user interface
IATA	=	International Air Transport Association
MMG	=	Multi model Generator
NVPI	=	Number of viscous/potential iterations
NWIT	=	Number of wake shape iterations
S&C	=	Stability and control

1 Introduction

1.1. Problem statement

The aircraft industry is accountable for 3.6% of total greenhouse gas emissions in the European Union and for 13.4% of the emissions from transport in 2016 [11]. Meanwhile, air traffic is on the rise. The amount of air passengers is expected to double by 2036. [13] The effect of increased air traffic between 2005 and 2017 on CO_2 and NO_x emissions in the European Union can be seen in Figure 1.1. The International Air Transport Association (IATA) aims to reduce net aviation CO_2 emissions by 50% before 2050, relative to the levels in 2005 [14]. So far most of the advancements to lower aircraft emissions are due to improved engine efficiency. However, these changes are not sufficient to meet CO_2 emission goals. Thus novel aircraft configurations are required to lower the aviation contribution to global warming.

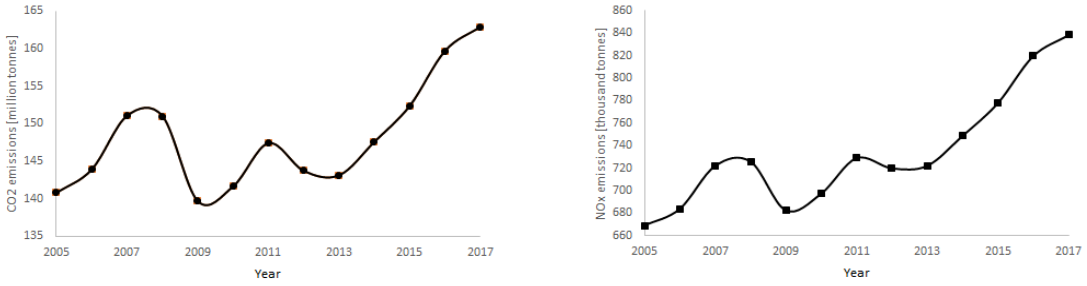


Figure 1.1: CO_2 emissions(left) and NOx emissions (right) of aircraft departing from the EU between 2005 and 2017. [11]

A novel aircraft configuration, which shows a lot of potential in reducing emissions, is found with the box wing aircraft. This configuration, which is based on the work of Ludwig Prandtl, features two main wings with a wing element connecting the two wings at the wing tips, as can be seen in Figure 1.2. [27]

An international team is collaborating in a research program, under the name of Parsifal[9], whose goal is to use the box wing design to replace the conventional aircraft for medium range distances, thereby competing with the Airbus A320 and the Boeing 737. TU Delft is one of the partners in this collaboration.

Three main advantages of box wing aircraft over the conventional aircraft can be defined. First, the box wing outperforms conventional aircraft with respect to induced drag, which accounts for about 40% of the total drag during cruise flight for transport aircraft.[18] Second, the box wing aircraft has more design space to place control surfaces.[35] By deflecting control surfaces on the front and rear wing simultaneously, direct lift can be achieved without the need to first increase the angle of attack of the aircraft in order to increase lift. By deflecting control surfaces on the connecting element between the front wing and the rear wing, a direct side force can be generated which is not possible for conventional aircraft. Finally, distributing the Lift force equally between the front and rear wing, means that the fuselage is now supported in two locations, instead of one. This reduces bending stresses on the fuselage meaning that the structure can be made lighter.

The challenge of implementing unconventional aircraft configurations, like the box wing aircraft, is that in early design stages conventional semi empirical design methods do not apply. This can have two reasons.

Figure 1.2: An example of a box wing aircraft. [5]

First, unconventional aircraft lack the necessary empirical data for the methods. Second, the methods are not applicable to unconventional aircraft, because required geometrical input parameters of the methods do not exist.

As an example to see how current design methods do not apply to box wing aircraft, the stability derivative C_{L_u} is used. Computing stability and control derivatives, henceforth referred to as S&C derivatives, is an important step in the design process of aircraft, because it forms the basis of establishing the airworthiness of the aircraft.

Specifically, C_{L_u} predicts how the generated lift of the aircraft will change as a consequence of a change in speed. A common method to determine this derivative is described by Roskam and can be seen in Equation 1.1.[33] The equation depends on the quarter chord sweep ($\Lambda_{c/4}$) of the single main wing, which in case of the box wing, does not exist because it has two main wings. Therefore, this equation cannot be used for a box wing aircraft.

$$C_{L_u} = \frac{M^2 \cos \Lambda_{c/4} C_L}{1 - M^2 \cos \Lambda_{c/4}} \tag{1.1}$$

This is only one of many details which cannot be captured by empirical methods. When empirical methods do not apply, physics based models are used. The advantage of using physics based methods is that they do not require empirical data of the aircraft, but the downside is that the computation time of these methods is often longer than using empirical methods.

A physics based approach for predicting C_{L_u} can be used by analysing the geometry for the same flight conditions except for different speeds, and then compute the difference in lift coefficient. This would increase the accuracy of the predicted results, but will also be relatively slow, because a full analysis of the geometry is required.

To benefit the development of box wing aircraft configurations it is essential to use a more physics based, higher fidelity approach for S&C derivatives prediction. The processing time needs to be relatively low, because the geometry is not yet frozen during conceptual design phase, which means that alterations in the geometry are still likely to occur. Applying physics based methods for predicting S&C derivatives will increase the confidence in novel aircraft configurations which leads to an increased chance that they will be chosen for further development after the conceptual design phase.

1.2. Research objectives

The necessity to introduce novel aircraft configurations without legacy information makes the current, semi empirical design methods inaccurate and sometimes unusable. This leads to the research objective of this report:

"Predict stability and control derivatives to support flight mechanics analysis for box wing aircraft configurations accurately and quickly using physics based methods."

The sub questions which support the research objective are stated and will be used as a guide throughout the report. They are defined as:

1. What is the best suitable aerodynamic analysis tool to compute S&C derivatives for box wing aircraft?

2. How can the preprocessing and post processing time of the chosen aerodynamic analysis tool be reduced?
3. How can a higher fidelity analysis tool be deployed within the conceptual design process?
4. What is the improvement of using the higher fidelity model compared to the current used method for box wing aircraft?
5. What are the bottlenecks in performing the higher fidelity analysis in the conceptual design phase?
6. How do the S&C derivatives of box wing aircraft configurations compare with S&C derivatives of a conventional aircraft configuration when designed for the same mission?

1.3. Report structure

Chapter 2 starts with describing in more detail the features of the box wing aircraft configuration. Then, this chapter explains how S&C derivatives can be calculated using a more physics based approach. This physics based approach requires a medium fidelity analysis tool for which a trade off matrix is created, in order to find the best suitable one. The aerodynamic tool needs to be integrated in a design methodology, of which previous work will be described in the last part of Chapter 2. An updated version is created of said design methodology, so that it can perform the tasks required for this thesis. The methodology is verified and validated in Chapter 4, followed by a sensitivity study on three different input settings in Chapter 5. The influence of the methodology on the run time is evaluated is Chapter 6. Lastly, a case study is done in Section 7 which a box wing configuration aircraft is modelled and the S&C derivatives computed. They will be compared with conventional aircraft which answers sub question 5.

2

Background

This chapter will provide background information for the thesis. Section 2.1 describes the unique characteristics of the box wing aircraft. Information on predicting stability and control derivatives using a more physics based approach is given in Section 2.2. Different aerodynamic tools which can be used to predict S&C derivatives are compared in Section 2.3. Lastly, Section 2.4 describes the design concept that will be used to perform the preprocessing and post processing. The shortcomings of this design concept will form the basis of Chapter 3.

2.1. Best wing system

The stacked wing concept used by the box wing configuration was first described in 1924 by Ludwig Prandtl, who described two wings stacked over each other as “the best wing system".[27] This section will describe the characteristics and opportunities for box wing aircraft divided over three departments:

- Aerodynamic properties
- Structural properties
- Controls and propulsion system

2.1.1. Aerodynamic properties

Drag can be divided into two sections, lift induced drag and parasite drag. As mentioned in the introduction, the lift induced drag consists of up to 40 percent of the total drag experienced during cruise by transport aircraft.[18] The relation between these drag components can be seen in Figure 2.1. The result of the paper that was published by Prandtl can be summarised in one figure, Figure 2.2. It shows that the induced drag reduces with an increasing amount of lifting surfaces, as well as increasing the gap (G) between these lifting surfaces.

Since then, many have expanded the knowledge about this aircraft configuration. Munks stagger theorem, implies that the induced drag advantage is independent of the horizontal location of the second wing relative to the first wing.[22] This means that the box wing aircraft can be used with swept wings, a concept that delays the flow over the wing from reaching the speed of sound.

Prandtl determined that, in order to fully exploit the advantage of reduced induced drag of the box wing aircraft, the lift should be divided equally between the two wings. However, in a more recent study performed by Demasi et al., it was determined through an analytical study that the load distribution can be shifted without increasing induced drag[9]. An infinite amount of solutions were obtained which consisted of a summation of an arbitrary constant to a reference optimal circulation. All these solutions have the same optimal circulation lift and optimal induced drag. This increases the design space of a box wing aircraft. The lift being generated can be partitioned unequally to favour stability and/or controllability requirements.

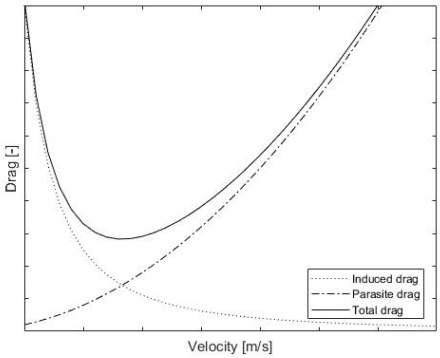


Figure 2.1: The relation between the different drag components and the velocity.

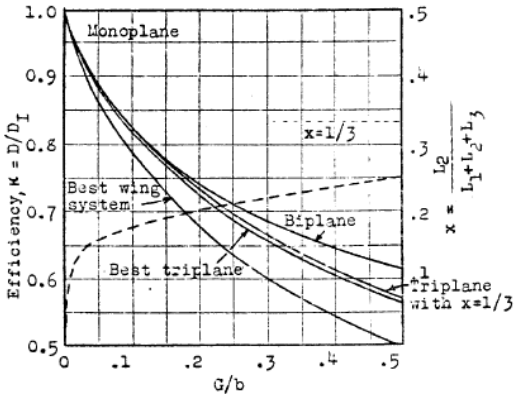


Figure 2.2: The conclusion of Prandtl in which he explains the best wing system to be the box wing.[27, p. 22]

2.1.2. Structural properties

The structural characteristics of the box wing aircraft differ on two key sections compared the the conventional aircraft.

First, the box wing configuration is overconstrained from a structural point of view. A conventional aircraft can be modelled as a simple beam, which is clamped at the root and free at the tip. For the box wing, the tip is also constrained. Therefore, more unknowns exist than equations for calculating the forces and moments in the structure. A stiffness iteration loop is also required to compute the required thicknesses of the stiffening component.

Second, the neutral axis for conventional aircraft, around which the bending moment acts, lies approximately in the middle of the wing. This means that, in case of an upward generated lift force, the top part of the wing will be compressed and the bottom part will be in tension. Chordwise moments and shear forces are often neglected. For the box wing aircraft this assumption cannot be made, as the wing system consists of two wings that are attached to each other with a connecting element, which means that these forces can become significant. According to Andrews, these secondary forces can even be higher than the out-of-plane bending moments and shear forces at certain locations along the span.[1]

In order to minimize structural weight while being able to cope with the induced moments, the material should be placed as far from the neutral line as possible. The secondary moments introduced in the box wing system cause the neutral axis to shift so that the wing box of a box wing design often feature additional material in the corners. The flange support are usually cross coupled as can be seen in Figure 2.3.[15] The cross coupling of the flange supports can save up to 13.75% compared to the symmetric case applied in most conventional aircraft according to Dal Canto et al.[8]

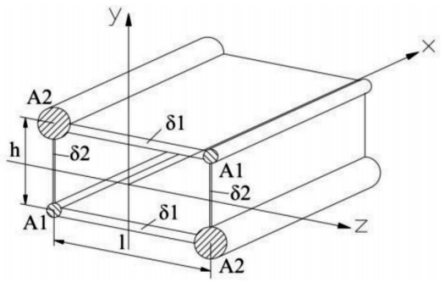


Figure 2.3: The cross coupling of the flanges for box wing aircraft.[8]

2.1.3. Controls and propulsion system

The extra wing space on the rear wing of the aircraft provides a larger design space for control surface and engine placement compared to conventional aircraft. This could potentially enable the automatic flight control system to exploit direct lift control and/or pure moment control for more efficient manoeuvring. A method has been developed and applied to the box wing aircraft to position the primary control surfaces as optimal

as possible for a specific aircraft design.[35] The optimal dimensions are shown in Figure 2.4. A similar result for the control surface placement has been obtained by Oliviero.[26] Simple models were used to evaluate the dynamic response of a box wing configuration, allowing initial sizing for control surfaces. The method has been applied to a small sized box wing aircraft, as can be seen in Figure 2.5. The control surfaces have been divided between inboard flaps/elevators on front and rear wing and outboard aileron control surfaces on both front and rear wing. The aircraft shown in Figure 2.5 is one of few box wing aircraft which has been built.

The best propulsion system for the box wing configuration, defined by Frediani, would consist of two electric turbofan engines because of its high fuel efficiency, according to a study by Voskuijl et al.[12] [38] After performing a trade off, taking into account several factors, the engines are placed at the tail of the aircraft as this proved the most promising concept. The following criteria were used in the trade-off: fuel burn, emissions, noise, compatibility with the PrandtlPlane configuration, development risk, acquisition cost, and maintenance cost.

Another advantage for placing the engine at the tail is that the engine intake will not require flattening, a measure that is usually taken for conventional aircraft with wing mounted engines with a high bypass ratio to preserve the required ground clearance.

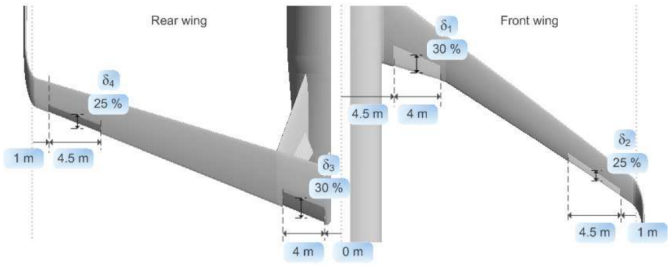


Figure 2.4: The final result of the optimal positioning study on the position of the primary movables for a certain box wing aircraft.[35]

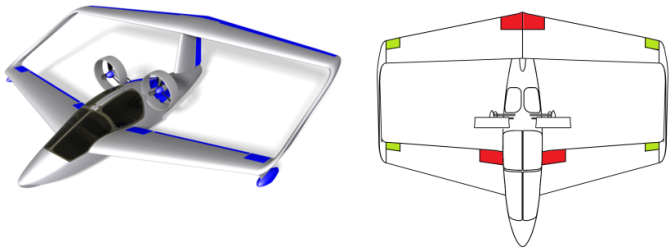


Figure 2.5: The positioning of the movables for the Idintos aircraft.[26]

2.2. Stability and control derivatives

For novel aircraft, like the box wing aircraft, many semi empirical design methods do not apply for sizing the aircraft which leads to more uncertainty in predicting the S&C derivatives. This section describes S&C derivatives in more detail.

2.2.1. Importance of stability and control derivatives

S&C derivatives are important in determining the behaviour of the aircraft at a specific flight condition. In total the effect can be analysed on 3 forces and 3 moments, caused by 9 different flight condition parameters, resulting in 48 S&C derivatives. They can be divided into longitudinal derivatives and lateral derivatives. Longitudinal derivatives are derivatives whose effect is symmetric in the x-z body plane of the aircraft (α, u, q, δ_e), while the effect of lateral derivatives is asymmetrical in the x-z body plane of the aircraft (β, p, r, δ_a, δ_r). S&C derivatives predict the effect of these parameters on the body forces and moments of the aircraft, shown in Figure 2.6. For example, the longitudinal stability derivative C_{X_q} determines what the effect is of the rotational rate around the Y-axis (i.e. pitch rate) on the force in the X-direction of the aircraft. The full list of S&C derivatives that are derived from the stability reference frame is given in Tables 2.1 and 2.2.

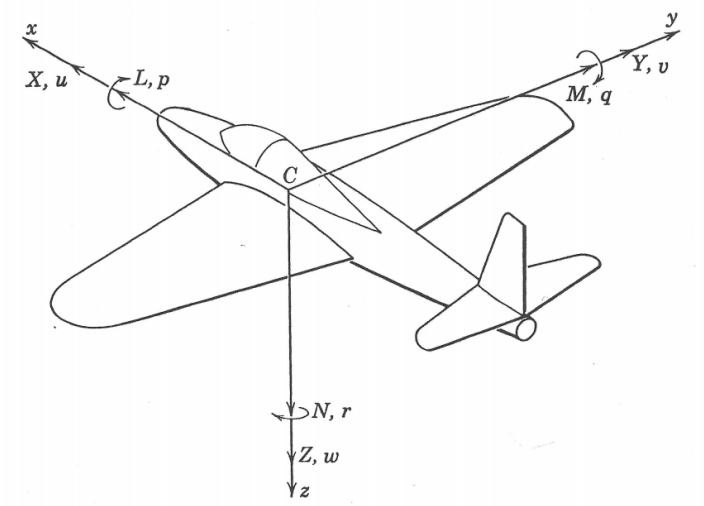


Figure 2.6: The body forces, moments, and rates of the aircraft. [7, p. 3]

Table 2.1: Longitudinal stability and control derivatives.

	Derivative	Definition
Static	C_{X_u}	$\frac{1}{\frac{1}{2}\rho VS} \cdot \frac{\partial X}{\partial u}$
	C_{Z_u}	$\frac{1}{\frac{1}{2}\rho VS} \cdot \frac{\partial Z}{\partial u}$
	C_{m_u}	$\frac{1}{\frac{1}{2}\rho VS\bar{c}} \cdot \frac{\partial M}{\partial u}$
	C_{X_α}	$\frac{1}{\frac{1}{2}\rho VS} \cdot \frac{\partial X}{\partial w}$
	C_{Z_α}	$\frac{1}{\frac{1}{2}\rho VS} \cdot \frac{\partial Z}{\partial w}$
	C_{m_α}	$\frac{1}{\frac{1}{2}\rho V\bar{c}} \cdot \frac{\partial M}{\partial w}$
Dynamic	C_{X_q}	$\frac{1}{\frac{1}{2}\rho VS\bar{c}} \cdot \frac{\partial X}{\partial q}$
	C_{Z_q}	$\frac{1}{\frac{1}{2}\rho VS\bar{c}} \cdot \frac{\partial Z}{\partial q}$
	C_{m_q}	$\frac{1}{\frac{1}{2}\rho VS\bar{c}^2} \cdot \frac{\partial M}{\partial q}$
Control	$C_{X_{\delta_e}}$	$\frac{1}{\frac{1}{2}\rho V^2 S} \cdot \frac{\partial X}{\partial \delta_e}$
	$C_{Z_{\delta_e}}$	$\frac{1}{\frac{1}{2}\rho V^2 S} \cdot \frac{\partial Z}{\partial \delta_e}$
	$C_{m_{\delta_e}}$	$\frac{1}{\frac{1}{2}\rho V^2 S\bar{c}} \cdot \frac{\partial M}{\partial \delta_e}$

Table 2.2: Lateral stability and control derivatives.

	Derivative	Definition
Static	C_{Y_β}	$\frac{1}{\frac{1}{2}\rho VS} \cdot \frac{\partial X}{\partial v}$
	C_{l_β}	$\frac{1}{\frac{1}{2}\rho VSb} \cdot \frac{\partial L}{\partial v}$
	C_{n_β}	$\frac{1}{\frac{1}{2}\rho VSb} \cdot \frac{\partial N}{\partial v}$
Dynamic	C_{Y_p}	$\frac{2}{\frac{1}{2}\rho VSb} \cdot \frac{\partial Y}{\partial p}$
	C_{l_p}	$\frac{2}{\frac{1}{2}\rho VSb^2} \cdot \frac{\partial L}{\partial p}$
	C_{n_p}	$\frac{2}{\frac{1}{2}\rho VSb^2} \cdot \frac{\partial N}{\partial p}$
	C_{Y_r}	$\frac{2}{\frac{1}{2}\rho VSb} \cdot \frac{\partial Y}{\partial r}$
	C_{l_r}	$\frac{2}{\frac{1}{2}\rho VSb^2} \cdot \frac{\partial L}{\partial r}$
	C_{n_r}	$\frac{2}{\frac{1}{2}\rho VSb^2} \cdot \frac{\partial N}{\partial r}$
Control	$C_{Y_{\delta_\alpha}}$	$\frac{1}{\frac{1}{2}\rho V^2 S} \cdot \frac{\partial Y}{\partial \delta_\alpha}$
	$C_{l_{\delta_\alpha}}$	$\frac{1}{\frac{1}{2}\rho V^2 Sb} \cdot \frac{\partial L}{\partial \delta_\alpha}$
	$C_{n_{\delta_\alpha}}$	$\frac{1}{\frac{1}{2}\rho V^2 Sb} \cdot \frac{\partial N}{\partial \delta_\alpha}$
	$C_{Y_{\delta_r}}$	$\frac{1}{\frac{1}{2}\rho V^2 S} \cdot \frac{\partial Y}{\partial \delta_r}$
	$C_{l_{\delta_r}}$	$\frac{1}{\frac{1}{2}\rho V^2 Sb} \cdot \frac{\partial L}{\partial \delta_r}$
	$C_{n_{\delta_r}}$	$\frac{1}{\frac{1}{2}\rho V^2 Sb} \cdot \frac{\partial N}{\partial \delta_r}$

S&C derivatives can be used, in combination with the aircraft inertia, to determine the behaviour of the aircraft. This is done by simulating the aircraft dynamic modes. The modes are divided into longitudinal modes (Phugoid and short period) and lateral modes (Dutch roll and spiral divergence). The dynamic modes can be determined by combining the following parameters:

- S&C derivatives
- Flight conditions
- Inertia of the aircraft
- Aircraft parameters (e.g. weight, surface area, span)

Once the flying modes have been simulated, it can be determined if the aircraft is dynamically stable and if it has good handling qualities. The ability to quickly self damp when the stick is briefly displaced in case of the short period is one of the many criteria for general aircraft certification.

2.2.2. Trim point

To compute the S&C derivatives, the trim conditions first need to be determined. Otherwise one cannot speak of stability derivatives, since dynamic stability has not yet been achieved. Trimming the aircraft means that the accelerations along the three body axis ($\dot{V}_x$, $\dot{V}_y$, $\dot{V}_z$) and the moments around the three body axis (M_x, M_y, M_z) are zero.

PHALANX (Performance, HAndling qualities and Loads ANalysis toolboX) is an in-house non-linear flight dynamics analysis tool developed in Matlab/Simulink®, which has been conceived by Voskuijl and further developed by Varriale.[37]. The following input is required for PHALANX to compute the trim conditions:

- An aerodynamic database consisting of different combinations of angle of attack (α), angle of side slip (β), Mach number(M), control surface deflection (δ), and angular rate around the three body axis x, y and z (p,q,r).
- CG of the aircraft.
- Engine thrust and location.
- Inertia of the aircraft.

Computing the aerodynamic database can be especially time consuming. As an example, when using the minimal required database size, which would consist of 2 angles of attack, 2 angles of side slip, 2 Mach numbers, 2 control surface deflections of three movables, and 2 angular rates for p, q and r, 48 cases will have to be generated (i.e. 2 α · 2 β · 2 M · 3 δ + 2 α · 2 β · 2 M · p,q,r = 48). The control surface deflections are not combined with angular rates p,q and r.

2.2.3. Physics based prediction of stability and control derivatives

When the trim point is determined, the S&C derivatives can be computed. As an example, the stability derivative C_{x_u} is computed. Analysing the forces in trim condition results in $C_{X_{trim}}$. Then the same geometry is used at a different velocity of $V_{trim} + \partial V$. The derivative can be computed using Equation 2.1.[21]

$$C_{x_u} = \frac{1}{\frac{1}{2}\rho V S} \cdot \frac{\partial X}{\partial V} \tag{2.1}$$

Where,

$$X = C_X \frac{1}{2}\rho V^2 S$$

Which can be combined to,

$$\frac{\partial X}{\partial V} = C_X \frac{1}{2}\rho V S + \frac{\partial C_X}{\partial V}\frac{1}{2}\rho V^2 S$$

And thus,

$$C_{x_u} = 2C_{X_{trim}} + \frac{\partial C_X}{\partial V} V$$

where,

$$\frac{\partial C_X}{\partial V} = \frac{C_{X_{trim}} - C_X}{V_{trim} - V}$$

In a similar way, the derivatives with respect to a change in side slip, Mach number, and accelerations around the body axis p,q and r can be computed. A schematic overview of predicting S&C derivatives can be seen in Figure 2.7. The challenge now is to find an appropriate aerodynamic analysis tool which can analyse the forces on the aircraft. This will be done in the next section.

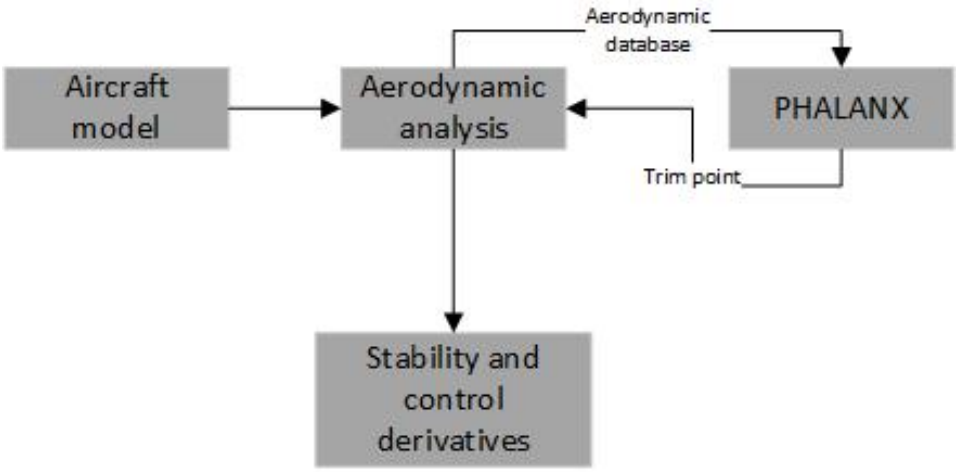


Figure 2.7: The process for predicting S&C derivatives using a physics based approach.

2.3. Aerodynamic analysis tools

The previous section described the need for an aerodynamic analysis tool that is able to generate the forces on the aircraft with sufficient accuracy and speed. A multitude of analysis programs are available to the design team to determine S&C derivatives. Each with their advantages and disadvantages. The different tools which are available to the author of this thesis are:

- Athena Vortex Lattice
- VSAero
- Ansys Fluent

2.3.1. Athena Vortex Lattice

Athena vortex lattice (AVL [10]) uses the vortex lattice method, based on solutions to Laplace's equation, to model the lifting surfaces as an infinitely thin sheet of discrete vortices to compute lift and lift induced drag (see Figure 2.8). The advantage of this method is that the computations are fast and the aircraft geometries are easy to produce. The main downsides of the vortex lattice method is that it is only able to compute linear, inviscid, and incompressible solutions. No flow separation or flow transition can be modelled using this tool.

2.3.2. VSAero

VSAero [25] uses a 3D panel method to replace a surface contour with singularity panels. Each panel is defined as a discretized surface onto which a singularity is modelled. Singularities are algebraic functions which satisfy Laplace's equation, shown in Equation 2.2. The most common functions are the point source, doublet, and vortex.

$$\nabla^2\Phi = \frac{\delta^2\Phi}{\delta x^2} + \frac{\delta^2\Phi}{\delta y^2} + \frac{\delta^2\Phi}{\delta z^2} = 0 \tag{2.2}$$

VSAero uses an advanced variant of this method and combines it with integral boundary layer equations so that it can estimate viscous effects. The advantage of a Panel method is that it has a short processing time compared to, while computing medium fidelity results. The assumption involved with the panel method is that the flow is irrotational. The user can choose between three different correction methods to account for compressibility effects in generating the results; Karmman-Tsien, Lieblein-Stockman, or Parndtl-Glauert.

2.3.3. Ansys Fluent

Ansys Fluent [2] is the most accurate computer aided analysis tool of this comparison. This comes at a price due to the fact that it also has the longest processing time. Ansys Fluent uses Navier-Stokes equations to describe the properties of the flow. These equations are the equation of conservation of mass, equation of conservation of momentum, and equation of conservation of energy. At higher Reynolds number, the flow

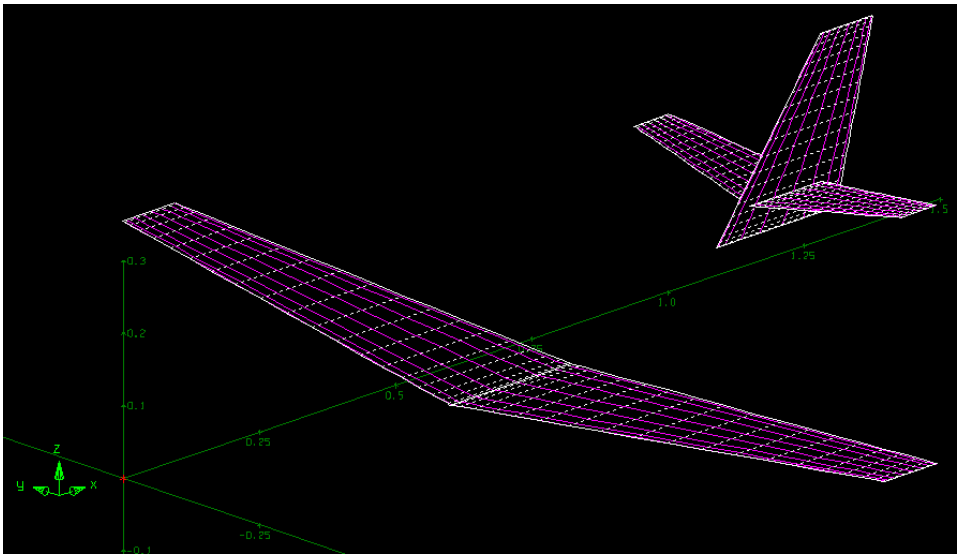


Figure 2.8: The modelling of aircraft inside AVL using infinitely thin sheets, thereby neglecting the influence of the airfoil thickness on the aerodynamic properties of the aircraft.

will become increasingly turbulent. A number of turbulence models are available in Ansys Fluent to predict the influence of the turbulence on the flow. Including these models in the computations will further increase the runtime of the analysis.

A study on using a computational fluid dynamics (CFD) solver in the conceptual design phase has been done by McCormick where it was concluded that for any CFD solver to be used in such an early design stage, an automatic mesh generator needs to be created to reduce the processing time. [20]

2.3.4. Tool comparison

A qualitative analysis is done to compare the different analysis tools with each other. What is required for an analyses tool depends on the case it is being used for (i.e. the optimal tool for the conceptual design phase of a conventional aircraft might not be the best case for an unconventional aircraft). When progressing further into the design process, the requirements will also change. Firstly, it is determined what the important rating criteria are. Secondly, a trade-off matrix is made from which the best tool for the specific box wing case is obtained.

Rating criteria

The first rating criteria is the total processing **time**. This includes the time it takes to prepare for the setup, the analysis, and the time it takes to process the results of the analysis. Especially during the conceptual design phase, when usually more than one concept is being developed and small changes to the design happen often, the processing time is very important. Processing time can usually be reduced by assigning more people to the project, but since this is a comparison between tools, it is assumed that an equal amount of people will be work on each tool to get a fair comparison.

The second rating criteria is the total **cost**, including man hours. The minimal amount of people required to run the tools analysis is assumed. For instance, it is fairly straightforward to perform the aerodynamic analysis with one person for these computer aided design tools, but it will require more man hours to preprocess Ansys Fluent than AVL.

The last rating criteria is the **accuracy** of the generated results. How well does the tool estimate the actual full scale data? The need for accuracy will increase when moving further downstream in the design process. However, as stated earlier, it is also more important for unconventional aircraft to obtain accurate analysis results at an early stage compared to conventional aircraft due to a lack of empirical data. Accuracy is coupled with the previous two in that the accuracy can be further increased by iterating the analysis (increasing cost and time) or purchasing faster and/or more accurate supportive equipment (increasing cost, decreasing time). For this qualitative analysis the average industry standard equipment is assumed.

The importance of each of these criteria depends on the stage of the design process, and on whether conventional or unconventional aircraft are being designed. For example, the later the stage in the design

process, the more important accuracy becomes, and the less important time becomes. In a general case, the importance of each of these criteria has been set equal. This means that the weight of each criteria is 1/3.

For unconventional aircraft the accuracy is more important than for conventional aircraft due to the lack of empirical data available. In order to make a proper qualitative comparison, the case needs to be defined. For this research, the case is based on an unconventional aircraft in the conceptual design stage. Therefore, in comparing the tools, the weight of the accuracy criteria will be doubled. The criteria of this case have been set to: processing time = 1, accuracy = 2 and cost = 1.

Trade-off matrix

All of the tools have their advantages and disadvantages. How they relate to each other can be seen in Table 2.3. Every method is compared with respect to how they perform on key aspects which are important during the conceptual design phase of an aircraft.

Table 2.3: Comparison of tools to compute S&C derivatives. The tools are rated between 1 (very bad) to 5 (very good). The **weighted total** takes into account the increased need for accuracy in the conceptual design phase by doubling the weight of accuracy.

	Processing time [weight=1]	Cost [weight=1]	Accuracy [weight=2]	Total	Weighted total
AVL	**5**	**5**	**1**	**11**	**11**
VSAero	**3**	**4**	**3**	**10**	**13**
Ansys Fluent	**2**	**2**	**4**	**8**	**12**

From the comparison it can be seen that, if all the rating criteria are weighted equally, AVL is the best aerodynamic solver for the conceptual design stage. However, for novel aircraft configurations it is especially important to produce higher fidelity results during the conceptual design stage. The weighted average column shows that, in that case, the VSAero analysis is a better alternative. VSAero is therefore chosen as the tool to perform the aerodynamic analysis.

2.4. The design and engineering engine

The problem in using VSAero is that it takes a lot of preprocessing time to get results. The solver requires an input file in which the geometry is segregated into patches, which are segregated into panels, which have to be ordered and orientated in a specific way.

A method has to be found to speed up this process, especially since the intent is to use this tool during early design stages, as the position of major components is not yet set, so large alterations are likely to occur. This is where the design and engineering engine (DEE) can be used. The DEE is a design system concept that has been developed at the TU Delft with the aim to improve the aircraft design process. It consists of the Initiator module, the Multi Model Generator (MMG), and various disciplinary analysis tools which can be attached to the framework.

2.4.1. Initiator

The Initiator is a conceptual design platform written in Matlab® conceived by La Rocca et al.[19]. Since its introduction, over 20 researchers have made contributions to improve the platform. It is able to synthesise both conventional and unconventional aircraft based on given set of top level requirements.

The preliminary sizing of the aircraft is done using Class I fuel fractions method of Torenbeek [34]. After the initial sizing is done, the main structure is sized using Class II.V methods, which means that the main structure is sized using physics based equations while the secondary sizing is done using empirical relations obtained from Raymer [30].

The aerodynamic properties of the aircraft are calculated using a vortex lattice method. This method is not able to calculate parasite drag and wave drag, so these are calculated using empirical relations described by Roskam [33].

2.4.2. Multi Model Generator

The MMG is a modelling tool conceived by La Rocca with the intention to support designers with "... a parametric modelling environment to define generative models of conventional and novel aircraft configurations and feeding various analysis tools with dedicated aircraft model abstractions, as required for the verification of the generated design"[19]. The MMG uses a number of functional blocks called High Level Primitives (HLP) to make up aircraft geometries. More information on HLPs can be found in Chapter 3.

The MMG was initially written on the ICAD platform. However, the ICAD platform was acquired by Dasault Systemes in 2002, after which the platform was taken off the market. This required the MMG to be newly created by van den Berg[3], Koning[17], and van Hoek[36] on a different platform: Genworks GDL. GDL is similar to ICAD because it uses the same programming language LISP. Further development on the MMG has been done by Wei, who moved the MMG from GDL to the current platform ParaPy, which is written in Python® as apposed to the outdated and hard to learn LISP language of GDL.[4, 39]

It should be noted that the MMG in itself does not represent a conceptual design methodology, however it can be used in combination with the Initiator to design the aircraft. Together with the analysis tools that the designer would like to use, the Initiator and the MMG form the DEE. A model of the DEE can be seen in Figure 2.9. A more detailed description of the MMG can be found in Chapter 3.

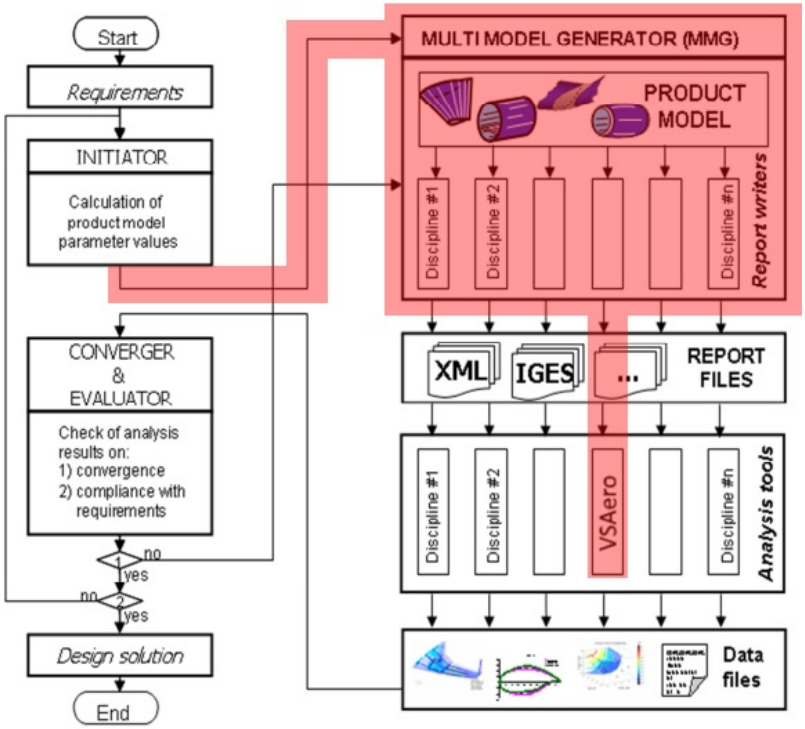


Figure 2.9: A model of the design and engineering engine. [19, p. 43] This research will focus on the development of the MMG, indicated with the red box: the MMG, the connection with Initiator, and the connection with VSAero.

2.4.3. MMG shortcomings

The MMG will be used as preprocessing and post processing tool for VSAero, which is the preferred analysis tool according to the trade-off matrix described in Chapter 2. Before the research goal can be achieved however, some changes will need to be made to the MMG.

The MMG is not able to generically model different aircraft configurations. Some knowledge is required about the aircraft configuration before it can be used. This increases the time it takes an aircraft configuration needs to be changed. For instance, when a cruciform configuration needs to be changed to a T-tail, a complete new class needs to be used. A more configuration agnostic methodology is preferred since large alterations in the configuration are still possible.

The MMG has a hybrid mesh method in place. A structured mesh is placed over the wing, and the part of the fuselage at the wing intersection has an unstructured mesh. Nathman concluded in his paper that using a fully structured mesh over the aircraft is preferable because it decreases the variation in panel size between neighbouring panels, permitting a more accurate calculation of the wake circulation. [24]

Finally, the MMG is not able to model box wing aircraft. modelling box wing aircraft poses extra challenges to the methods which are used to create the model. The class which is used to model wing segments is not able to model wings which turn 180 degrees, like the connecting element between the front and rear wing of the box wing, so a new wing class will have to be written for this. This problem will be described in more

detail in Chapter 3.

2.5. Conclusion

An aerodynamic analysis tool is needed for physics based prediction of S&C derivatives of box wing aircraft. The trade of done in this chapter showed that VSAero is the most suitable analysis tool. The downside of VSAero is the long preprocessing time because the geometry in the input file needs to be defined in detail. Long lead times (i.e. preprocessing, processing and post processing) are unfavourable in early design stages, because large design alterations are likely to occur, which leads to having to repeat the calculations.

MMG is an in house modelling tool which has the potential to automate a large part of the preprocessing of box wing aircraft, and thereby reducing the preprocessing time. However, the MMG faces a number of challenges, of which the largest one is that box wing configuration aircraft cannot be processed. This, and other challenges will be described and resolved in Chapter 3.

3

Methodology

The most significant downside of using VSAero is the long preprocessing time as discussed in Section 2.3. The MMG will be used in an effort to decrease this processing time by automating large parts of the design cycle. However, Section 2.4 marked out the shortcomings of the MMG.

This chapter gives an overview of all the steps between between the input of the MMG and the computed S&C derivatives. The steps which will serve as sections in this chapter are defined in Figure 3.1. Some parts that are described come from previous work and some have been added by the author of this report.

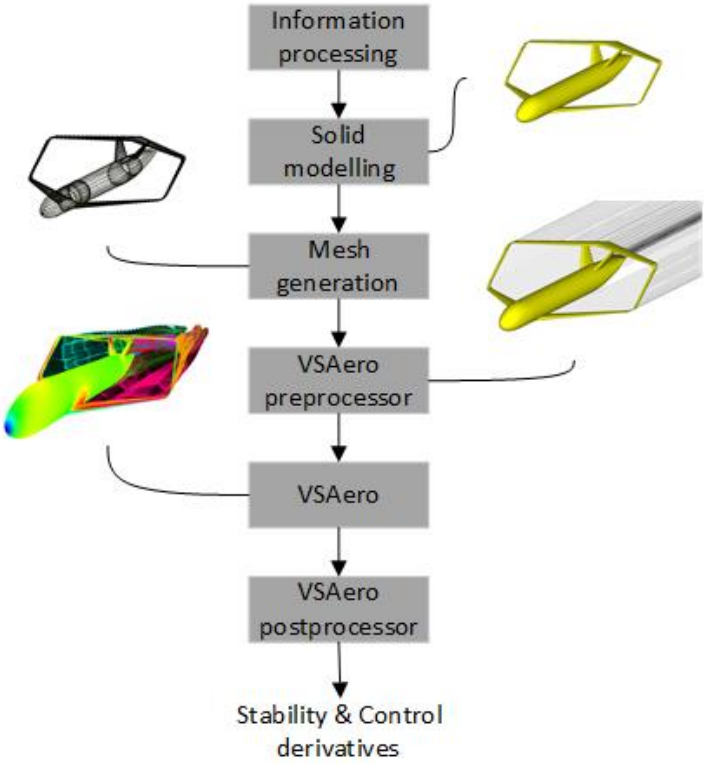


Figure 3.1: The main steps which will serve as sections for computing the stability and control derivatives.

3.1. Information processing

The MMG is not built to allow users to quickly generate aircraft configurations using top level aircraft requirements (TLAR). Instead, JSON files (JavaScript Object Notation, standard data interchange format of Python) of the wings and fuselage are written in a structure which is useful from a modelling perspective, but not necessarily from a designer perspective. The geometry is not defined by parameters like span or sweep, but by a

leading edge and trailing edge curve among others. An example of a JSON input file for a single wing can be found in Appendix A.

A stream of information needs to be passed to the MMG to be able to model the aircraft and perform the preprocessing for VSAero. The complete list of information required is:

- Wings sizing information
- Fuselage sizing information
- Settings for the aircraft mesh
- Settings for the VSAero analysis

3.1.1. Input reader class

In the MMG, the information of the fuselage and the different wing instances used to be directly fed to the different High level primitives (HLPs). HLPs are the distinct building blocks with which the different aircraft geometries are built. HLPs are described in more detail in Section 3.2. This way of information processing setup can be seen in Figure 3.2.

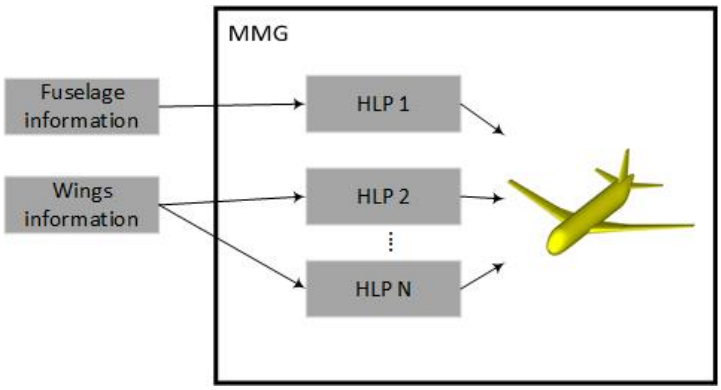


Figure 3.2: Old way of information processing for the MMG. The wings and fuselage information is directly supplied to the HLPs.

When directly connecting the input files to the HLPs, the user can only change the geometry of the aircraft by making changes to the JSON input files, increasing processing time, since the ParaPy graphical user interface (GUI, see Appendix B) has to be restarted for the changes to take effect. Processing time could be decreased if small edits (e.g. wing position, movable deflection angle) to the model could be made after the geometry is generated inside the GUI. Therefore, a separate class is created in which all the data is being read. This class, the *AircraftInputReader*, is generated alongside the geometry and is displayed inside the GUI. When certain parameters are edited in the *AircraftInputReader*, the effect will automatically be observed in the GUI, without having to regenerate the model.

Besides increasing the speed with which the model can be adapted, the *AircraftInputReader* can be used to scale elements individually, or the complete model. Scaling a model can be useful when having to convert the units from the imperial system to the metric system. A study has been done on the effect of scaled models by Kulkarni using the *AircraftInputReader* [29].

3.1.2. CPACS data

A link is created between JSON, the default file system of Python, and CPACS, which is a data structure developed by DLR (see Figure 3.3. This makes it easier to import an aircraft model in the MMG. CPACS allows for a wide range of information about an aircraft and its performance to be stored in a single xml file. The reason why CPACS was introduced is to make it easier for different design tools to exchange information about aircraft design with each other.

By enabling the MMG to read CPACS file, a design flow can be created in which multiple programs can work in series or parallel to design and optimize aircraft. Specifically, for this research, it allows the Initiator program of the TU Delft to initiate any aircraft, which can then be read and modelled by the MMG. The used CPACS files for this project originate from Initiator. However, any tool that uses CPACS can now be linked with the MMG.

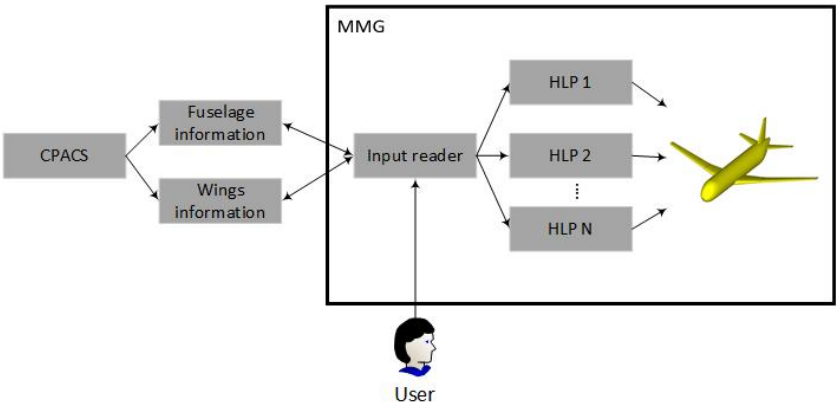


Figure 3.3: New way of information processing for the MMG. The user can directly change the geometry of the aircraft model without having to restart the GUI. CPACS files can be used to generate json input file for the wings and fuselage.

3.2. Solid modelling

After the input data is read, different HLPs are created which make up the aircraft. As mentioned in Section 2.4, HLPs are the distinct building blocks with which the different aircraft geometries are built. La Rocca conceived the terminology to make a distinction between HLPs and low level primitives used by conventional CAD systems such as surfaces, solids and splines. [19]

In total, there are 4 different HLPs which can be generated:

- Fuselage
- Wing
- Connecting element
- Wing extension

Further information on the construction of the Fuselage and Wing HLP can be found in the thesis of Wei. [39].

3.2.1. Connecting element

The class that is used to model the Wing HLP (i.e. *wingFromRails*) cannot be used to generate a connecting element. For a wing element, a leading edge and trailing edge rail is defined which is projected on a dihedral surface (see Figure 3.4). Because of the projection method, the theoretical limit of the dihedral angle is 90 degrees. A connecting element between two wings with no dihedral can be seen as a single wing with a dihedral of 180 degrees. Even when using two wings in which the front wing has an upward dihedral angle of 90 degrees and the rear wing has an anhedral angle of 90 degrees, the projections method will not work.

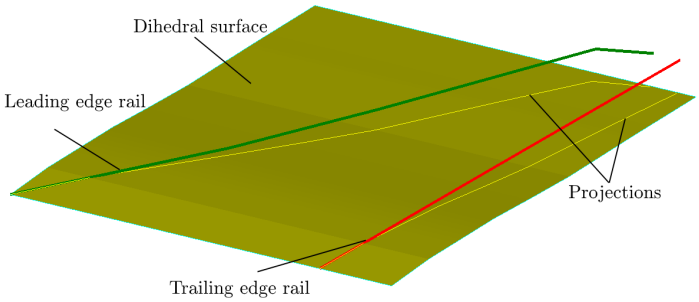


Figure 3.4: The projection of the leading edge (green) and trailing edge (red) onto a dihedral surface (yellow) shown here is an essential step in the modelling of the wing and causes the inability to model a complete connecting element.

New entries have been added to the JSON file to be able to generate a connecting element called Rotation and Scaling. The new method uses the leading edge rail to place airfoil profiles defined in the CPACS file,

which then are scaled and rotated according to these values in the input file. This way, no dihedral surface and projection is used. This new HLP is visualized in the following figures.

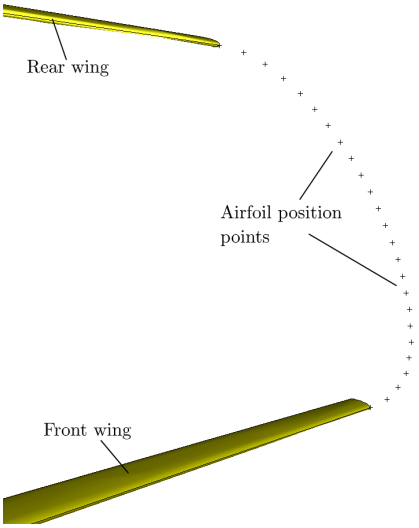


Figure 3.5: Creating points at the leading edge rail.

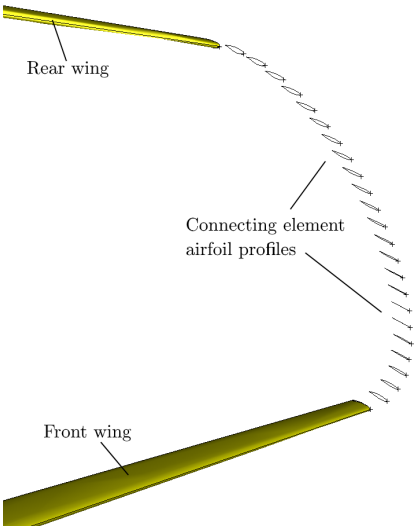


Figure 3.6: Placing the airfoil profiles at their right location and rotating them according to the Rotation input from the JSON file.

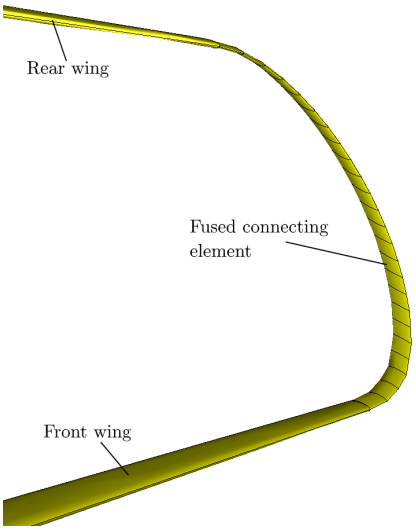


Figure 3.7: Fusing the profiles to make a connecting element solid.

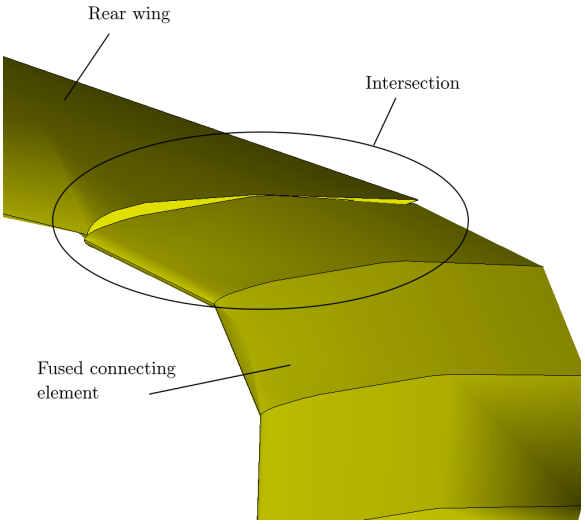


Figure 3.8: Detail of the connection between the rear wing and the connecting element.

3.2.2. Wing extension

In some cases, the accuracy with which the connecting element is defined in the input file is not high enough for medium fidelity analysis tools like VSAero (see Figure 3.8). This has been the case with the connecting elements which are defined in CPACS by Initiator.

Therefore, a new HLP is developed called the wing extension generator. The goal of the wing extension generator is to replace the connecting element which is defined in input files with a more accurately defined connecting element which is suitable for VSAero analysis.

A set of requirements has been set for the wing extension generator:

- A smooth transition from win tip to connecting element should be in place for both leading edge as trailing edge.

- Parameters should be defined which allow the user to change the shape of the connecting element.

- The wing extension generator should make it possible to dictate a maximum span of the aircraft since this is usually an important limiting factor in aircraft design.

- The wing extension generator should be able to model both winglets, as well as the connecting element between two wings.

- The wing extension generator should have the option to place movables on the straight section of the connecting element if it would exist.

Instead of creating a new HLP only capable of modelling the element between the two wings, the wing extension generator is able to model both winglets as a connecting element, thereby increasing its functionality. The HLP is enabled by defining an extension type, either 'winglet' or 'connecting element'. Depending on which of these two is passed on, the HLP will pass one or two wings into the wing extension generator.

To increase the design freedom, the HLP should be able to model a movable as well. Movables at the connecting element can create a pure side force when used as lift increasing elements, or for directional control when asymmetrically deployed.

The step wise generation of the connecting element can be seen in the following Figures.

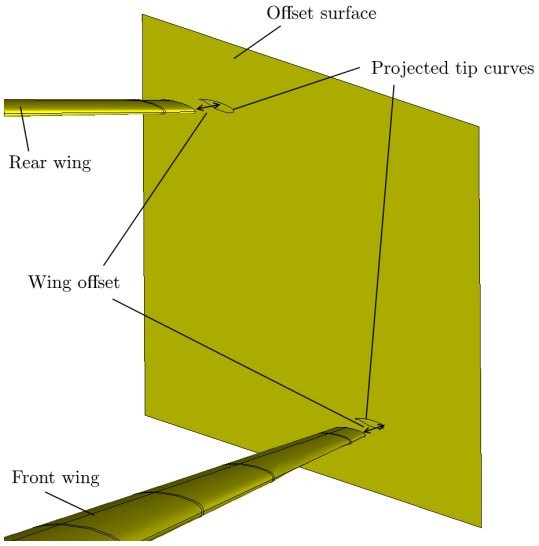


Figure 3.9: The surface is generated at input distance "wing_offset" from both wing tips. The tip profiles are projected onto this surface.

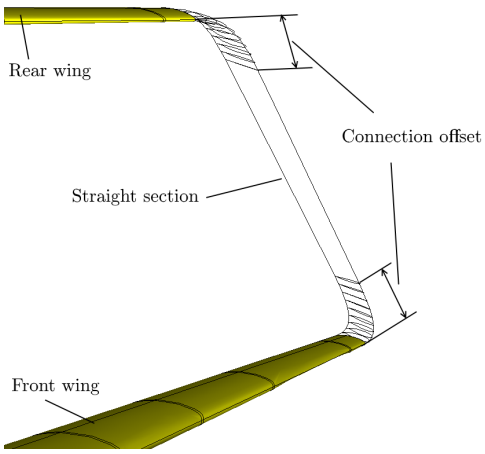


Figure 3.10: Four points are created on the surface at an input distance "connecting_offset" from the projected curves. These points are used to create the leading edge and trailing edge rails.

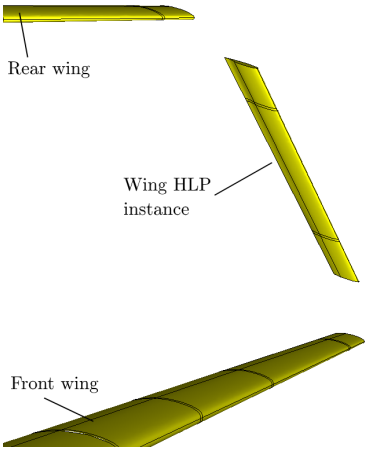


Figure 3.11: A straight *wingFromRails* instance is created between the defined connecting offset points.

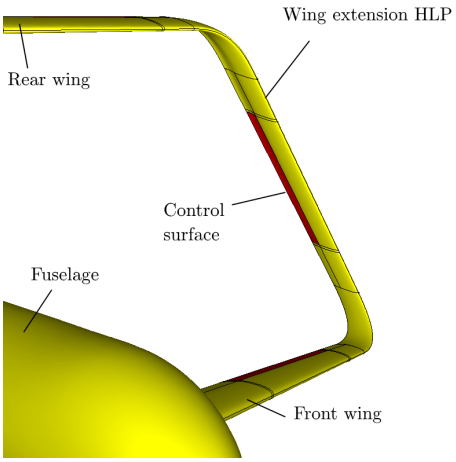


Figure 3.12: The top corner solid, bottom corner solid, and *wingFromRails* instance are fused together into one connecting element solid. In case of a winglet extension, only the bottom corner solid is generated.

Besides the functionality of creating a connecting element, the wing extension generator is able to create simple winglet elements. A winglet is initiated when the input 'wing extension type' is set to 'winglet'. The steps to create the winglet can be seen in the following figures.

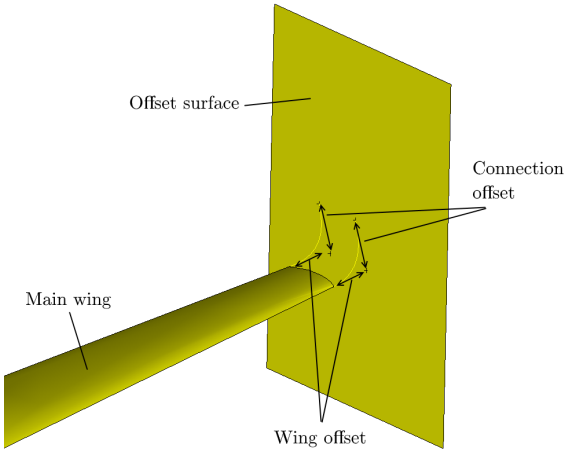


Figure 3.13: The surface is generated at input distance "wing_offset" from the wing tip. The tip profile is projected along the leading edge and trailing edge which result in the offset points. From the offset points at a distance determined by connection offset, the connection offset points are defined, which mark the tip of the winglet.

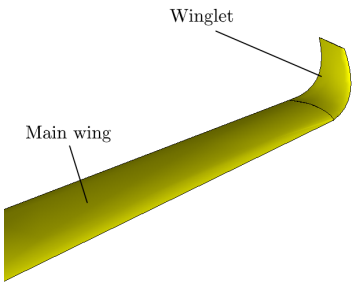


Figure 3.14: Airfoil profiles are defined along the leading edge and trailing edge which are fused to create the winglet solid.

3.2.3. Configuration agnostic fusion of elements

After all the separate elements have been defined and constructed, they need to be fused together to make one single solid (one element being an instance of an HLP), regardless of what configuration is being mod-

elled (see Figure 3.15). The fusion order is important to prevent the MMG for trying to fuse elements which do not connect. For instance with a T-tail design, the vertical tail needs to be fused before the horizontal tail. Otherwise, the horizontal tail will be floating. In the case of Figure 3.15, Element 7 cannot be fused to the main solid before either element 2 or 6.

Therefore, the elements are ordered by intersection. The first element is selected to be either the fuselage or the first element in the list of wing elements (indicated in red in Figure 3.15) and then a second element is added to the list when it intersects with any of the elements in the list.

After all the elements are properly sorted, they can be fused together. This can be done element by element, or batch by batch (a batch being a list of elements which all intersect with any of the elements in the fusion list). In case of a geometry which has an element that can be attached from two sides, it is important to fuse the geometry one by one. Otherwise, a double intersection edge might exist between two elements.

For example, when fusing the elements of a box wing aircraft using the batch by batch method. A fuselage is present so the first batch would consist of the elements connected to the fuselage: the front wing, the mirrored front wing, and the vertical tail plane(s). The second batch then consists of the elements connected to the newly generated solid: The two connecting elements and the rear wings. All the elements are now added, so the fusion process is completed. The problem here is that the rear wing is attached through the vertical tail planes, and the connecting element is attached through the front wing, meaning that they are not connected to each other. Therefore, batch by batch fusion will not work for aircraft geometries which have elements that can be attached from two sides.

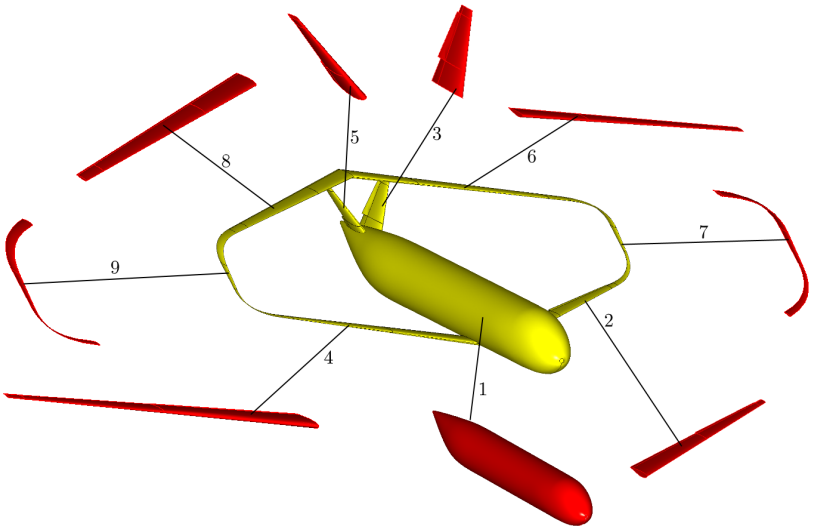


Figure 3.15: The stepwise fusion of the solid. The individual elements are indicated in red. The numbers describe the order in which the fusion is done.

3.3. Mesh generator

The solid of the aircraft needs to be discretized into smaller triangular or quad panels for the VSAero analysis. ParaPy has a built in mesher on which the mesh generator is based. This built-in mesher will be described first. The development of the mesh generator is based on previous work by Wei and Kulkarni. [29, 39]

3.3.1. ParaPy mesher

ParaPy uses a number of classes which wrap around modules from Salome to create a mesh of any solid, face, or edge.[4] The following input is required to generate a mesh over an object:

- **Shape to mesh** This is the shape from which a mesh should be made. This can either be a solid, a face or an edge.
- **Controls** The controls input is used to define the density of the mesh. For this thesis the density for the wing is controlled using the spanwise pitch and chordwise number of points. The fuselage is meshed

using equispaced points with control over the lateral pitch and longitudinal pitch.

- **Groups** The complete mesh of the shape to mesh can be divided into sub meshes using the groups input. This is useful for VSAero since the mesh of each element of the aircraft (e.g. vertical tail or fuselage) will need to be described and processed individually.

The built-in mesher is able to create both triangular panels (i.e. with 3 edges) as quad panels (i.e. with 4 edges). Quad panels are used to create a structured mesh, while triangular panels are used for unstructured meshes. A fully structured mesh is desired as input for the VSAero software because it decreases computation time. The location of each panel on a structured mesh is stored in a table, which relates to the physical location of the panel (i.e. f(x,y) = f(i,j)). In a structured mesh, it will be easier to find the neighbours of the panel ((i-1,j),(i+1,j),(i,j-1),(i,j+1)), making it more efficient with respect to computation memory.

The downside of using a fully structured mesh is that is is difficult to create a fully structured mesh on more complicated sections like the fuselage wing intersection area. Unstructured meshes also give more control over the mesh density compared to structured meshes.

Because of this increased computation speed, the goal for this thesis is to use a fully structured mesh for the wing and fuselage surfaces, with the exception of the wing tips and fuselage nose and tail face.

3.3.2. Splitter curves

The surface of the solid is split in a number of faces using splitter curves. This is needed because every face that makes up the surface of the solid has to have exactly 4 edges. This poses a challenge, especially at the wing-fuselage intersection, as seen in Figure 3.16

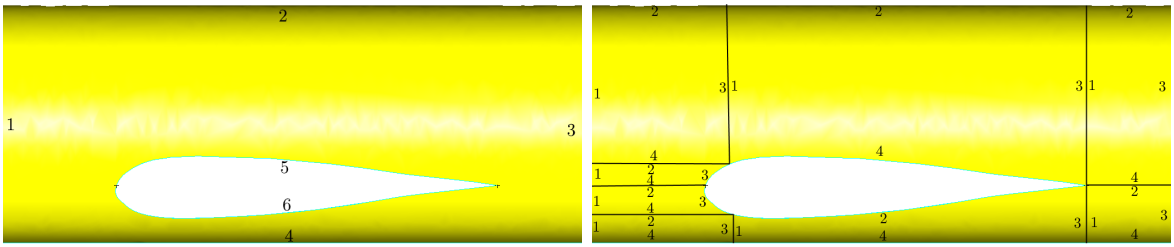


Figure 3.16: Side view of the wing fuselage intersections. Without the splitter curves (left), the intersection consists of a single face with 6 edges so no fully structured mesh is possible. The right figure shows the same face including split curves which divides the wing fuselage intersection into 8 faces which all have 4 edges.

The split curves on the fuselage face will have an effect on the wing faces because the leading edge wing splitter curve split the intersecting edge in two. This means that the wing fuselage intersection now consists of two edges instead of one. Therefore, wing splitter curves will need to be introduced. This can be seen in Figure 3.17

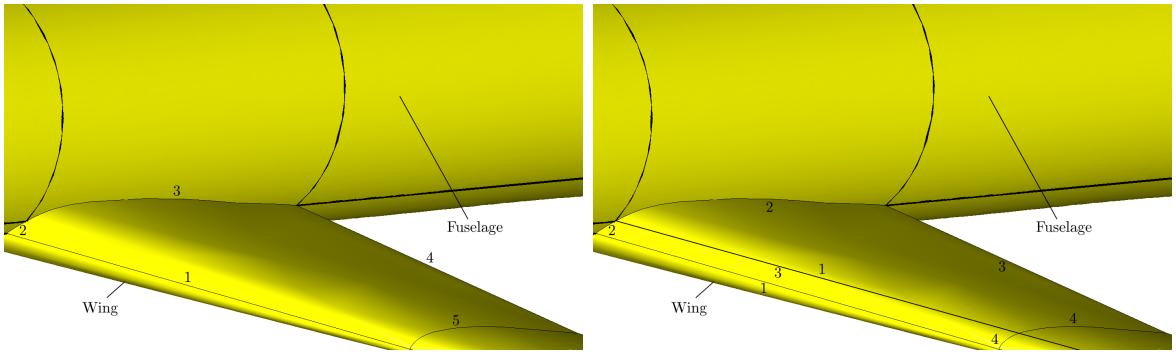


Figure 3.17: Iso view of the wing fuselage intersections. Without the splitter curves (left), the top face of the wing has 5 edges which prevents from creating a fully structured mesh. In the right figure a split curve is introduced which divides the top wing surface into two faces with both having four edges.

When all the wing splitter curves and fuselage splitter curves are drawn, the solid is divided into faces which all have four edges (quad faces) except for the fuselage nose face, tail face, and wing tips. The quad faces that are created from these splitter curves for a box wing aircraft can be seen in Figure 3.18.

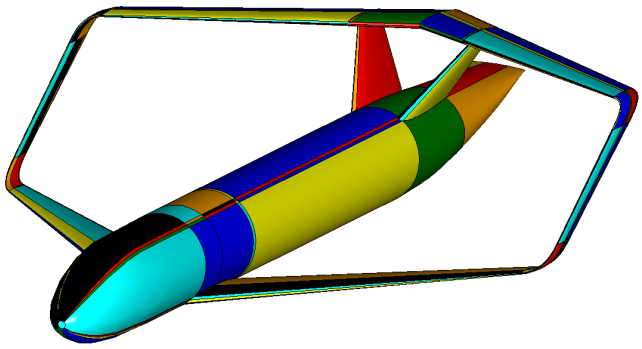


Figure 3.18: The splitter curves split the surface of the solid into quad faces.

3.3.3. Edge chains

The next step in generating a fully structured mesh is to create edge chains. All the edges are collected for each element (e.g. vertical tail plane, main wing). Any of the intersection edges are selected and it is traced on which face it lies, and then compute the opposite edge which lies on the same face.

Now there are two edges in your edge chain, the start edge and the opposite edge. For the opposite edge a check is performed which indicates if it also lies on any of the other faces in the quad faces database. If that is the case, the opposite edge is again added to the edge chain. This process repeats itself until no more unique faces or edges are found. The procedure for this is depicted in Figures 3.19 and 3.20.

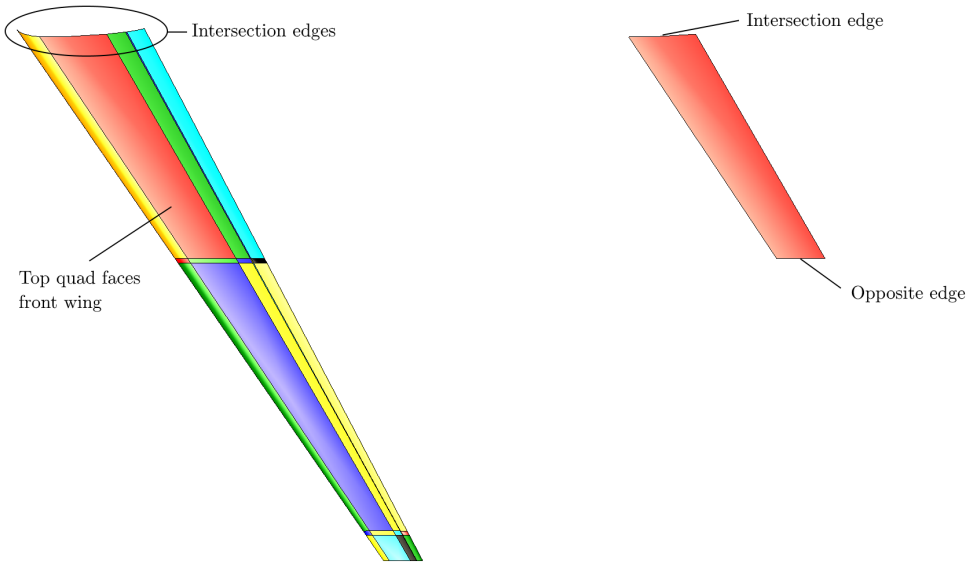


Figure 3.19: All the top quad faces for the front wing (left) and the start of the edge chain generation starting with the intersection edge (right).

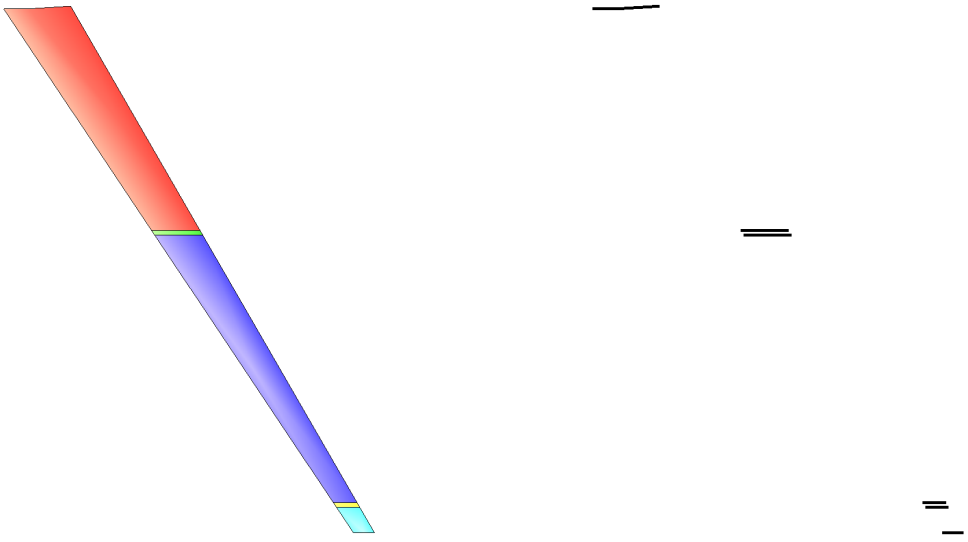


Figure 3.20: The face chain (left) and edge chain (right).

For wing element the edges and faces are separated into chordwise and spanwise edge chains. For the fuselage they are separated into longitudinal and lateral edge chains. The difference can be seen in Figure 3.21.

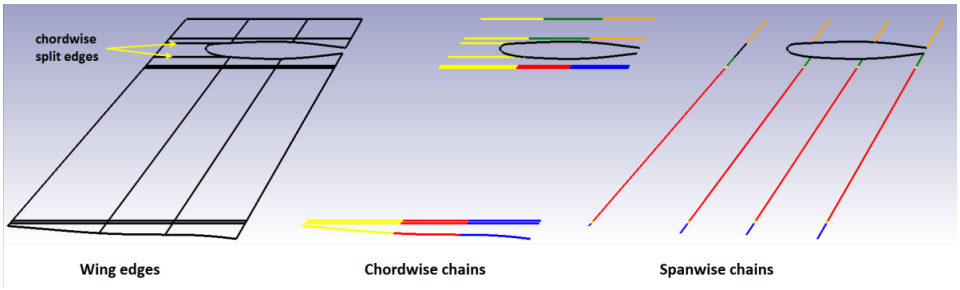


Figure 3.21: Creating edge chains for the fin element. Edges belonging to the same chain are given the same colour. [39, p. 47]

Once the chordwise and spanwise chains have been defined for all aircraft elements, they can be combined. This is done by checking if they have a common edge in their edge list. A full edge chain for a box wing aircraft which includes chains of the elements fuselage, front wing, connecting element, rear wing, rear wing mirrored, connecting element mirrored and front wing mirrored can be seen in Figure 3.22.

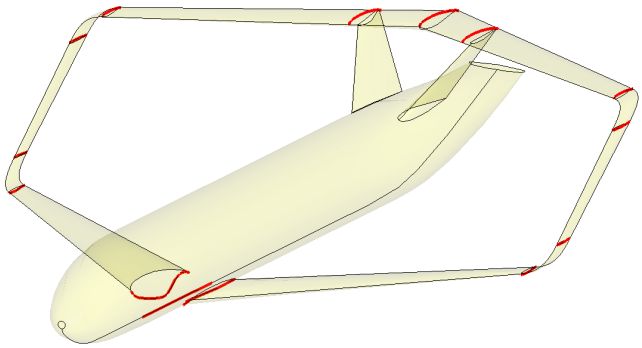


Figure 3.22: One complete edge chain, shown here in red, is created by combining the edge chains of the different HLP instances.

3.3.4. Curved leading edge splitter curves

The split method described in Section 3.3.2 uses three straight splitter curves at the leading edge of the wing fuselage intersection, and propagates them over the fuselage surface, and over the wing surface. This method has a downside in that a large number of small panels are created. A cosine distribution is used to place the mesh points along the chord of the main wing. Creating a fully structured mesh means that these points are propagated along the fuselage towards the nose of the aircraft. A higher panel density means a more accurate solution at that location, but it also increases computation time.

A new solution of splitting the leading edge is proposed by the author of this report, which uses a curved splitter curve at the leading edge(see Figure 3.23). This decreases the number of panels by 20% and thus reduces the processing time by a similar amount. Reducing the preprocessor time is one of the requirements to effectively use VSAero during the conceptual design phase.

The downside of this new method could be that a critical area directly in front of the leading edge has a lower mesh density, thus the solution might become less accurate at this location. This will be checked in Section 4. The difference between the two different meshing methods is shown in Figure 3.24.

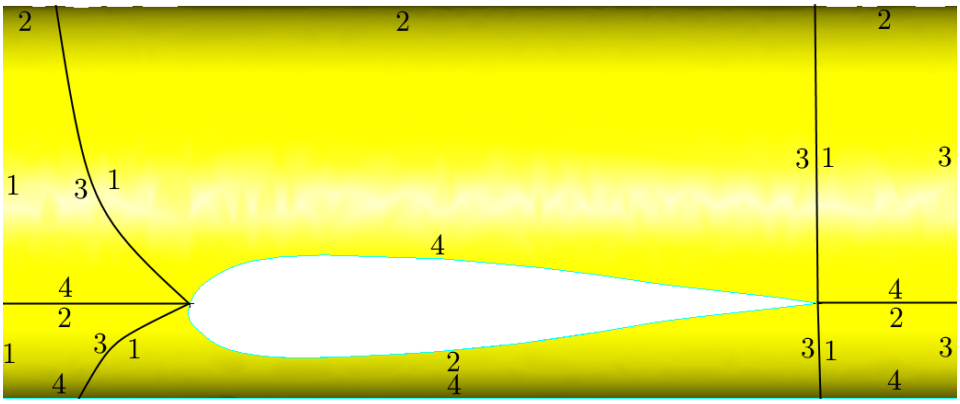


Figure 3.23: Dividing the wing fuselage intersection into quad faces using curved leading edge splitter curves.

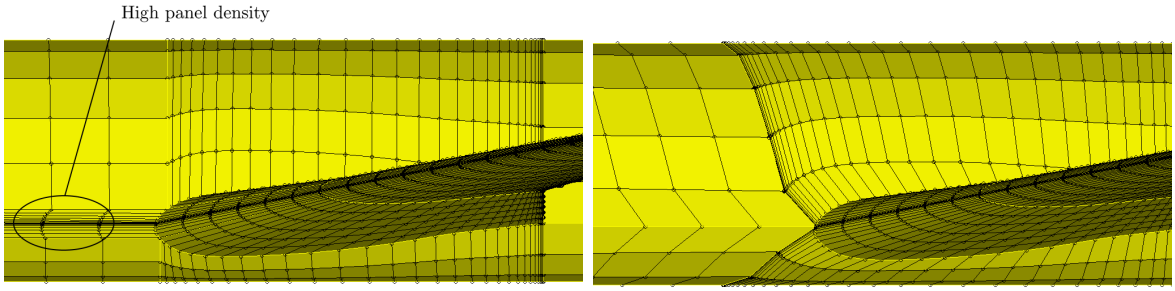


Figure 3.24: The difference between using straight leading edge splitter curves (left) and curved leading edge splitter curves (right).

3.4. VSAero preprocessor

VSAero requires a specific description of the shape that is to be analysed so that it can compute the aerodynamic results. The input is divided into components, wakes, and settings.

3.4.1. Components

The mesh that is generated is divided into different patches. Every patch consists of a number of panels. One panel consists of four mesh nodes. According to the VSAero developers, the accuracy of the analysis increases when the top and bottom side of the wing are included in one patch. Therefore, the first panel in the MMG starts at the trailing edge and moves over the top surface to the leading edge, then goes back to the trailing edge over the bottom side of the wing.

An overview of the numbering of the panels can be seen in Figure 3.25.

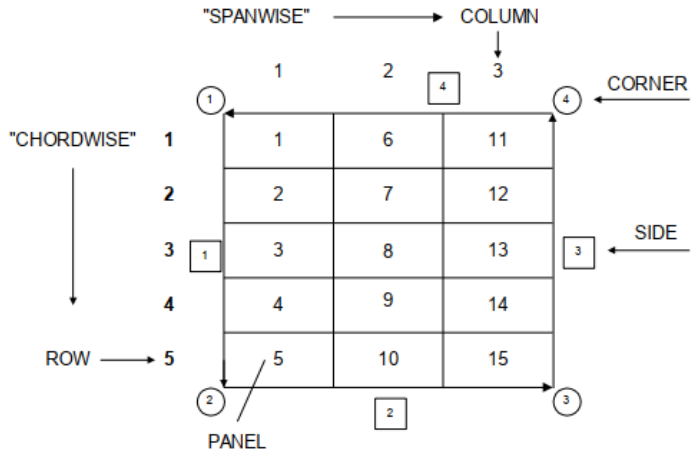


Figure 3.25: The panelling convention for each patch in the VSAero solver. [25, p. 50]

Besides the orders of the panel on each patch, it is also important that the patch is properly orientated. The normal should always face outwards of the solid which is ensured by having the sides properly defined as shown in Figure 3.25. The panelling order inside the MMG can be seen in Figure 3.26

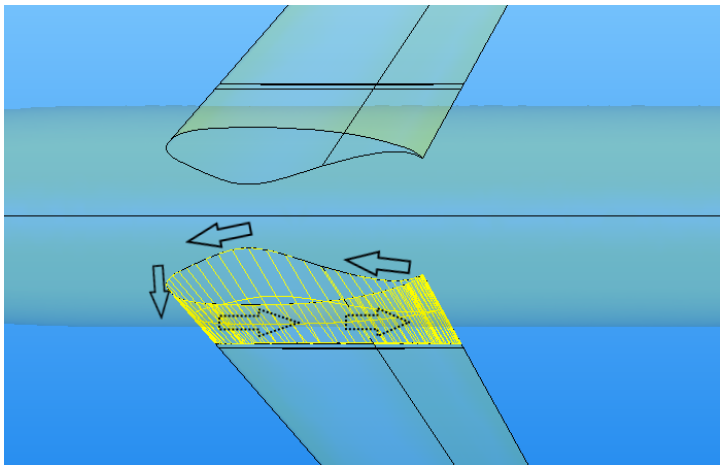


Figure 3.26: The top view of the aircraft describing the panelling order in the MMG. The first panel is located at the trailing edge, then the panels are added following the indicated arrows.

3.4.2. Wakes

At the trailing edges of the wing and at the rear of the fuselage, wake shedding panels are defined from which the wing wakes originate. The wakes represent the vorticity shed into the flow by the passage of the body. Without the wake lines, no lift or side force can be generated. The points which describe the wake lines are collected in grid planes (see Figure 3.27), which are common to all the wake lines being shed from the model.

A stitching line is required when there is an intersection between two elements. This can either be two wings, for instance in case of a cruciform tail, or a fuselage and a wing. Instead of following the path of the free stream direction when being shed from the wake panel, it follows the contour of the element that is located last in order of free stream direction. For instance, in case of fuselage-vertical tail intersection, the fuselage is placed more aft in streamwise direction, which is why the stitching line will follow the contour of the fuselage, after which it is shed from the wake panel of the fuselage.

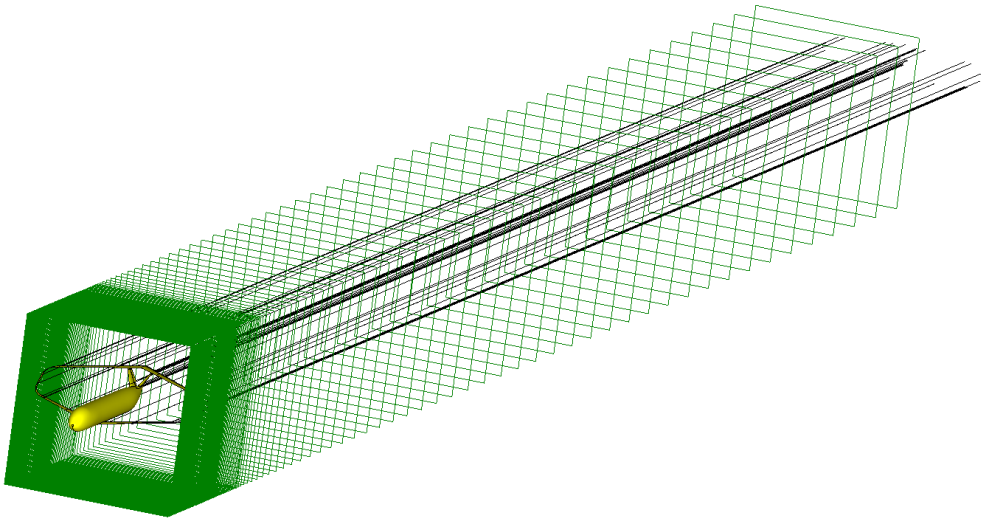


Figure 3.27: The preprocessed aircraft assembly including the aircraft solid (yellow), wake lines (black), and wake grid planes (green).

3.4.3. Settings

All the information required by VSAero is divided into a set of 35 cards. The geometry and wake information is stored in these cards, as well as the input settings.

Important cards, which the user manually have to set for every geometry that is being analyzed, are card 6, which contain the atmospheric conditions, and card 7, that contains the geometrical reference values for aerodynamic chord (used to normalize pitching moment), span (used to normalize rolling and yawing moments), surface area (used to normalize patch force coefficients), and cg location (used in determining moment coefficients).

The number of wake shape iterations and viscous/potential solution iterations can be set using card 4. The effect of theses parameters on the runtime and accuracy of the solution is analysed in Chapter 5.

3.5. VSAero post processor

VSAero generates an output file with all the information of the forces on the different patches and elements both in body directions as in wind directions for every iteration that is defined in the input file.

For visualisation purposes, a file is created which can be opened in Omni3D®. Omni3D® is an interactive 3D visualisation tool, which gives extra information on the sanity of the geometry described in the input file. Wrongly orientated patches and misplaced wake lines, which cause erroneous results, can be spotted quickly. The effect of the wake shape iteration can also easily be observed. Figure 3.28 shows the pressure distribution of a box wing aircraft visualised by Omni3D®.

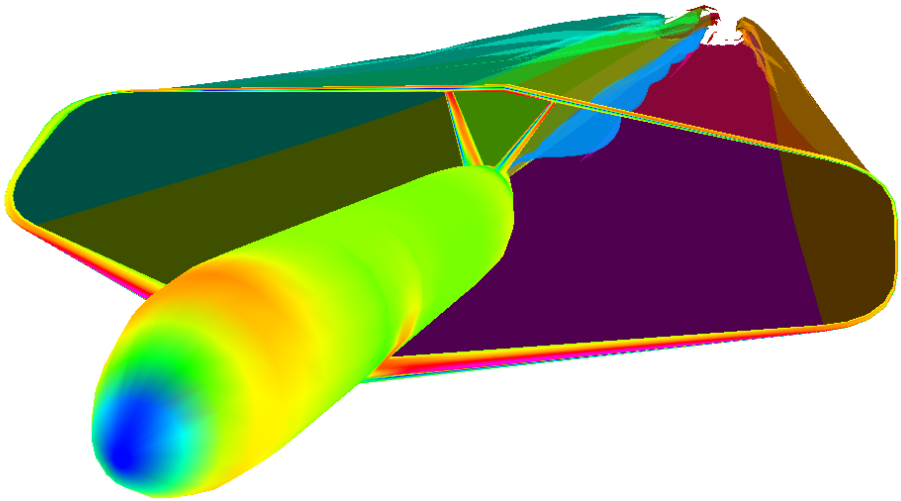


Figure 3.28: The pressure distribution over a box wing aircraft in Omni3D® which has been preprocessed by the MMG.

3.6. Conclusion

The methodology described all the steps that the aircraft model will need to make for computing the S&C derivatives. The HLP instances which make up the aircraft model can be described in CPACS or JSON files. The geometry can be edited inside the ParaPy® GUI using the *AircraftInputReader*. A wing extension can be generated and adapted using defined control inputs. The HLP instances are automatically fused, after which the solid is split into quad faces using splitter curves. The quad faces allows the model to create a fully structured mesh over the complete solid, except for the wing and fuselage end faces. This mesh is stored in an input file for VSAero, together with user defined atmospheric conditions, after which a VSAero analysis can be performed to calculate the aerodynamic characteristics along with the stability and control parameters of the aircraft. The process is summarised in Figure 3.29.

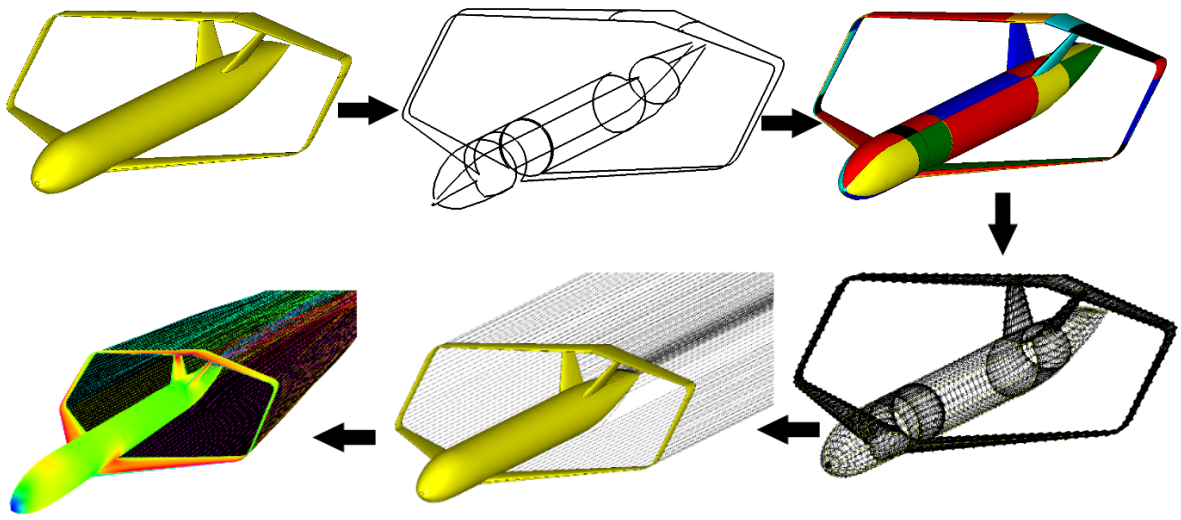


Figure 3.29: The full process to obtain the aerodynamic characteristic of the model.

4

Verification and validation

Two conventional aircraft are used to verify and validate using the MMG to preprocess aircraft models. Box wing aircraft models would be preferable, however no flight test or wind tunnel data is available to the author of this report of box wing models that can be loaded in the MMG. Besides verifying and validating the preprocessor of the MMG, this section will answer sub question 3 by comparing the results of using the MMG in combination with VSAero and the results of AVL.

The AVL analysis is done using the ParaPy platform.[4] The AVL wrappers in ParaPy make it easy to preprocess the aircraft for the AVL analysis by automatically creating input and case files from certain defined input parameters which describe the geometry of the aircraft and the run case.

4.1. VGM

VGM is a conventional aircraft model which was tested in a wind tunnel at the TU Delft, as seen in Figure 4.1. The downside of the VGM data is that the experiment is done at a Mach number of 0.13 and Reynolds number of 500.000, which is far lower than design Mach numbers for modern commercial aircraft. Also, the VGM has a simple geometry, using the minimal of 2 sections defining each wing (at the root and at the tip). The data of the wind tunnel test has been obtained from the article by Raju Kulkarni et al. [28] Important dimensions of the VGM model can be seen in Figure 4.2.

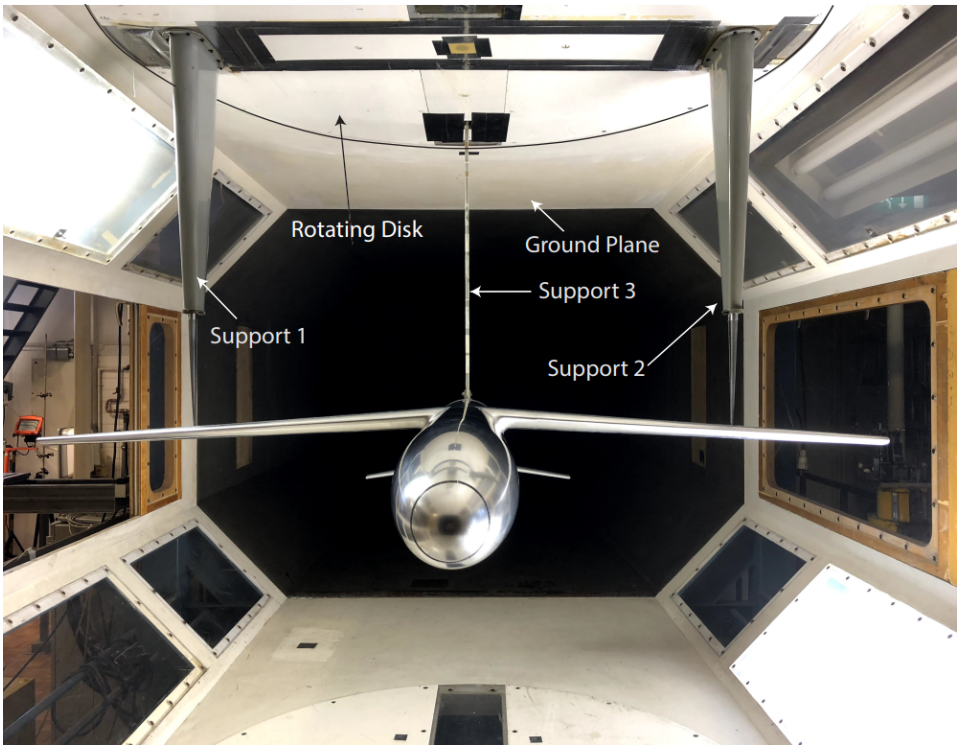


Figure 4.1: The VGM test model in the wind tunnel at the aerodynamic test facility of TU Delft.[28, p. 17]

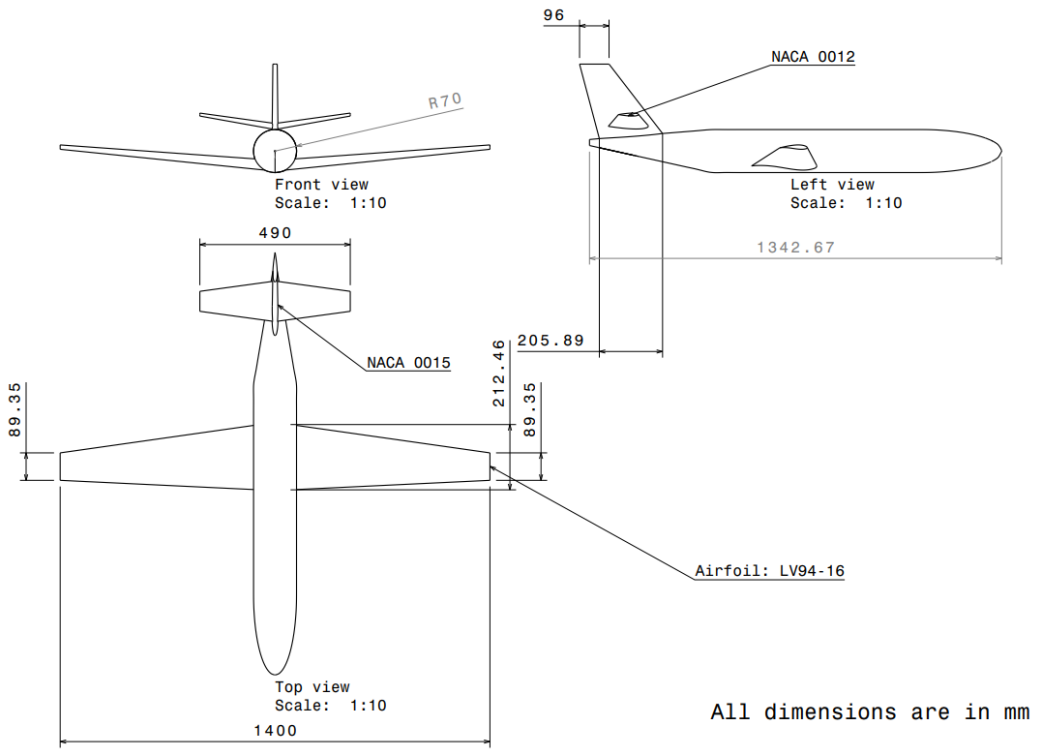


Figure 4.2: The VGM main dimensions.[28, p. 17]

4.1.1. Aerodynamic analysis

The results of the wind tunnel tests, the VSAERO analysis, and the AVL analysis for the lift is compared in Figure 4.3. Both VSAero and AVL approximate the lift curve well up until 9 degrees. Then, a decrease in

slope is observed for the wind tunnel data caused by separation of the flow from the surface of the wing. Both VSAero and AVL are not able to model separation. That is why the error increases rapidly after about 9 degrees.

The comparison of the drag curve is shown in Figure 4.4. Both AVL as VSAero significantly underestimate the drag of the VGM. Drag consists of a number of components (see Equation 4.1), some of which are not taken into account by VSAero and AVL. AVL computes inviscid solutions, meaning that only the lift induced drag is taken into account. VSAero computes some viscosity in the form of skin friction and the effects of boundary layer displacement. The difference in drag between the wind tunnel test and the VSAero and AVL analysis is because the pressure drag and interference drag are not taken into account.

$$CD = CD_{induced} + CD_{parasite} \tag{4.1}$$

In which $CD_{parasite}$ is defined as:

$$CD_{parasite} = CD_{skin} + CD_{pressure} + CD_{interference}$$

In Figure 4.5, the pitch moment coefficient curve of the wind tunnel test is compared with VSAero and AVL. It can be seen that, both VSAero and AVL match the moment coefficient quite well. VSAero performs slightly better in predicting the moment coefficient and trend line at 0 angle of attack. The drop in moment coefficient for the wind tunnel test at approximately 9 degrees matches with the drop in lift in Figure 4.3. From this it can be deduced that separation sets on the main wing earlier than on the horizontal tail plane, which is favourable in aircraft design because it ensures an additional pitch down moment when the aircraft stalls (i.e. a loss of lift due to flow separation at high angles of attack).

The key values for the aerodynamic analysis are summarised in Table4.1.

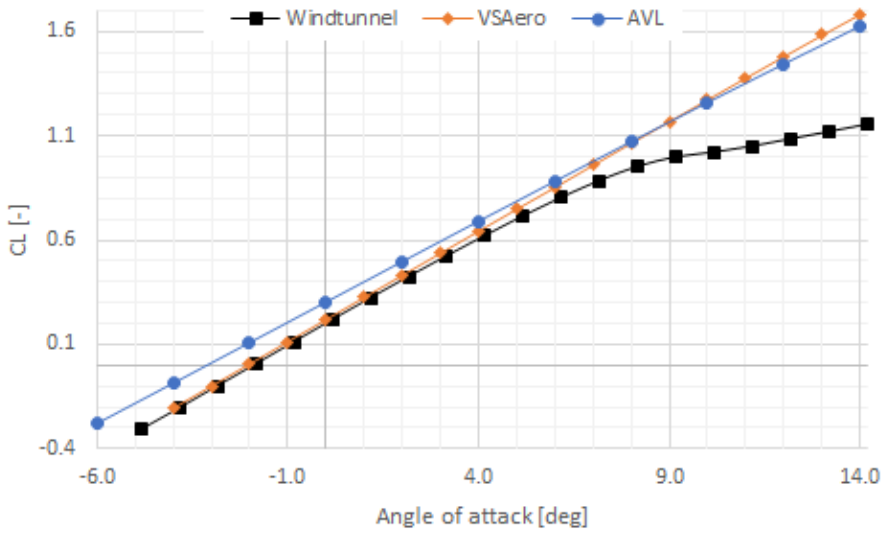


Figure 4.3: The lift curve of the VGM aircraft using different analysis tools. Re = 500,000 at Mach 0.13.

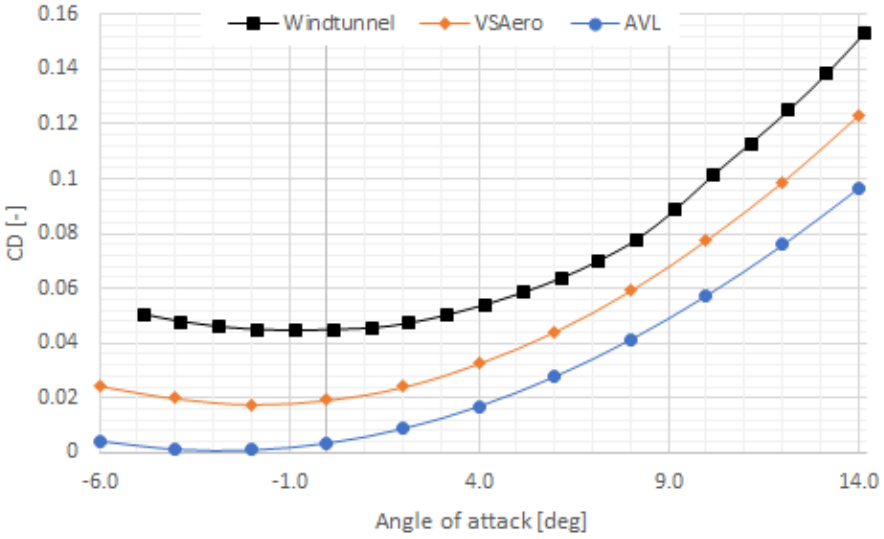


Figure 4.4: The drag curve of the VGM aircraft using different analysis tools. Re = 500,000 at Mach 0.13.

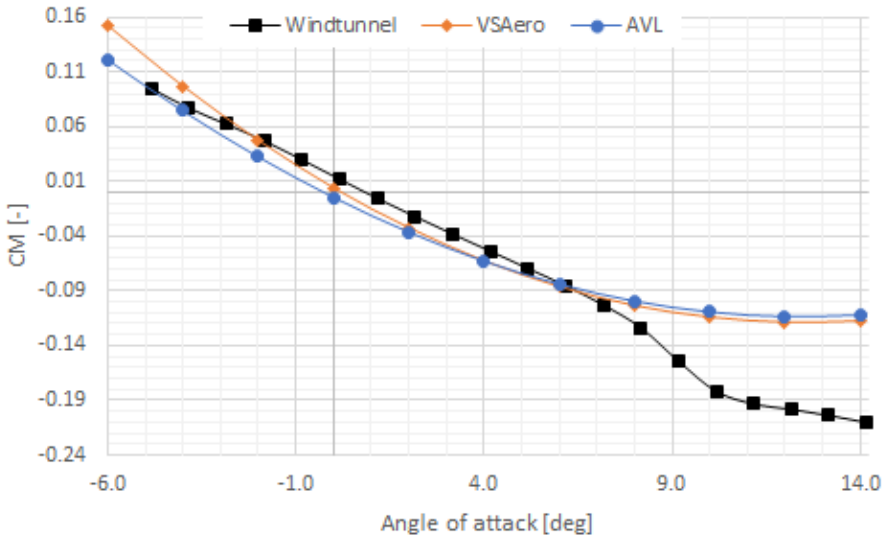


Figure 4.5: The pitch moment curve of the VGM aircraft using different analysis tools. Re = 500,000 at Mach 0.13.

Table 4.1: Comparison of key aerodynamic parameters computed by wind tunnel tests, VSAero and AVL.

	Wind tunnel	AVL	Difference [%]	VSAero	Difference [%]
CL_0	0.216	0.301	-39.4	0.215	-0.2
C_{L_α}	0.106	0.097	-8.1	0.107	1.7
C_{D_0}	0.045	0.001	-97.8	0.018	-60.2
C_{M_α}	-0.0177	-0.0160	8.7	-0.0180	2.6

4.1.2. Static longitudinal and control derivatives

A selection of S&C derivatives are computed and compared using the different aerodynamic analysis tools. The trim condition which is assumed is 2 degrees angle of attack, with no sideslip. In most cases it can be

seen that the S&C derivatives are better estimated using the VSAero solver. What stands out in this comparison is $C_{X_{\delta_e}}$, because of the large error margin of AVL. This is caused by the fact that total drag for AVL only consists of lift dependent drag. The negative deflection of the movable causes a lift reduction and thus the drag reduction, thereby causing the positive relation between C_X and δ_e.

The large error margin of C_{X_u} is caused by the drag being underestimated by both VSAero as AVL, as can be seen in Figure 4.4. C_{X_u} is calculated using Equation 1.1, which includes the term $2C_{X_0}$. Drag is proportionally related to the force in X-direction, which is why this coefficient is underestimated by both VSAero and AVL. The average error margin for VSAero is 32% less than the average error margin of AVL.

Table 4.2: Longitudinal stability and control derivative comparison for 2 degrees angle of attack and no side slip.

		Wind tunnel	AVL	Difference [%]	VSAero	Difference [%]
Static	C_{X_u}	0.053	0.0174	68	0.011	79
	C_{Z_u}	0.818	1.008	23	0.855	5
Control	$C_{X_{\delta_e}}$	-0.00037	0.0002	154	-0.00040	8
	$C_{Z_{\delta_e}}$	0.00844	0.0114	35	0.01128	34
	$C_{m_{\delta_e}}$	-0.0265	-0.0346	30	-0.0335	26
Average error margin				62		30

4.2. NASA Common Research Model

The NASA Common research model (CRM), is a model from which wind tunnel experiments have been performed at different locations around the world with the intention to create an experimental database to validate aerodynamic analysis tools. [23] An image of the model during the wind tunnel tests in the Langley wind tunnel can be seen in Figure 4.6. The experiment of which the data is used, was done at a Mach number of 0.7, Reynolds number of 3.000.000. The test model is significantly larger than the one for VGM (see Figures 4.7 and 4.8).

Many research studies have used the CRM to validate their analysis tools [6, 16]. However, it was noted that the support system used in the wind tunnel test had a larger influence on the aerodynamic data then expected. Therefore, correction factors have been used to account for the effect of the support system, which have been obtained from the report by Rivers et al. [31] In addition to this correction factor, the geometry described on the website of the CRM was described in 1G conditions. Therefore, an additional wing twist is added to the geometry of the wing.

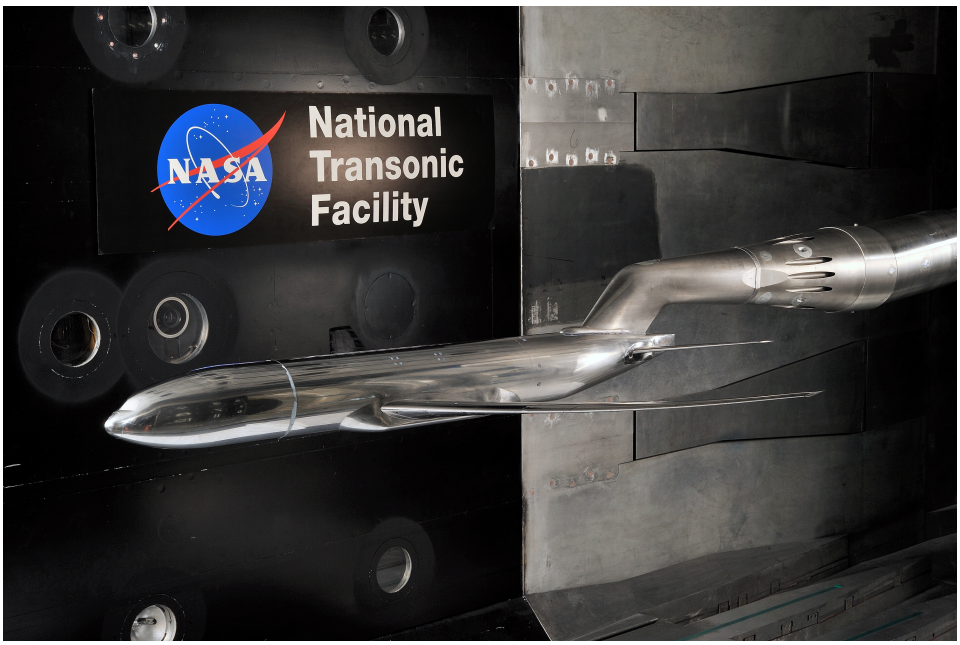


Figure 4.6: The NASA CRM in the National Transonic Facility in Hampton, Virginia.[23]

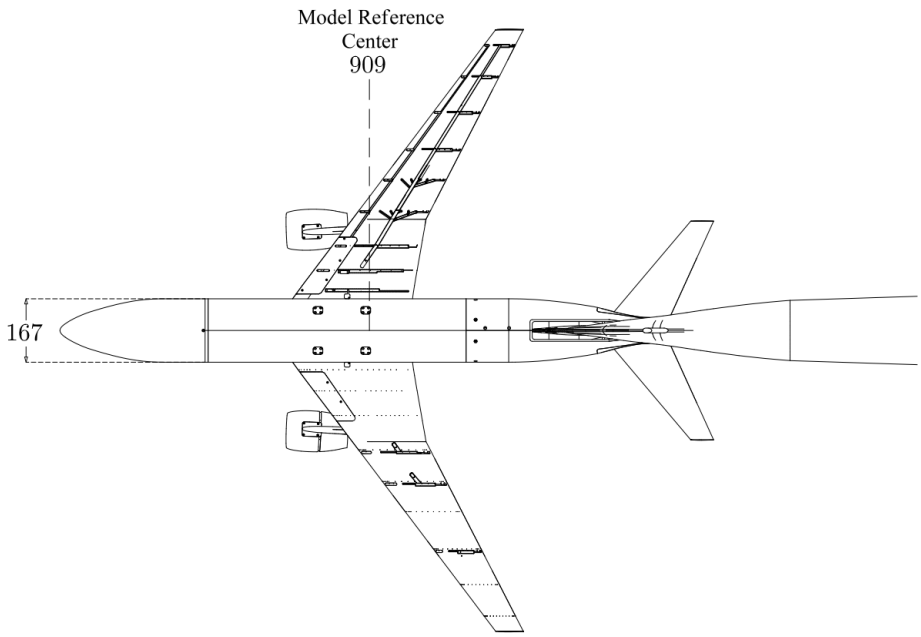


Figure 4.7: Schematic top view drawing of the NASA CRM including its main dimensions in mm. [32, p. 16]

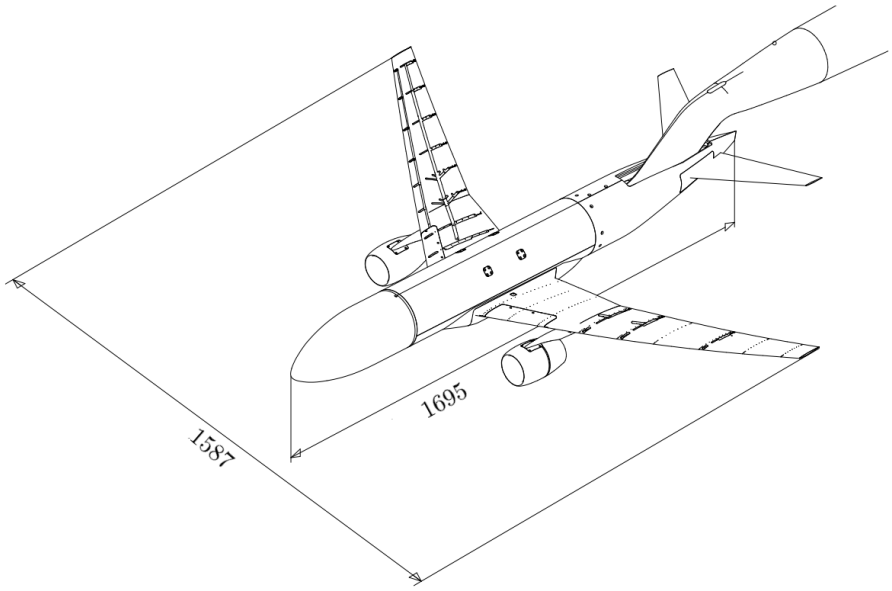


Figure 4.8: Schematic isometric view drawing of the NASA CRM including its main dimensions in mm. [32, p. 16]

4.2.1. Aerodynamic analysis

The aerodynamic results of the CRM wind tunnel tests are better predicted by VSAero which can be seen in Figure 4.9. Both VSAero as AVL overestimate the lift coefficient for the complete range of angles of attack, although the prediction of VSAero is better than of AVL. The drop in lift coefficient of the wind tunnel data at approximately 7 degrees angle of attack accounts for the onset of separation on the wing of CRM.

The drag seems to be predicted exceptionally well by VSAero, as can be seen in Figure 4.10. VSAero only takes induced drag and skin friction drag into account when computing the drag coefficient. A more significant underestimation of the drag coefficient, as shown for VGM in Figure 4.10, is expected in order to account for other viscous effects (e.g. interference drag). Since the lift coefficient matches the wind tunnel reference data well, it is concluded that the influence of skin friction drag is overestimated by VSAero.

The moment coefficient comparison is shown in Figure 4.11. As mentioned in the introduction of this

section, a correction factor has been used for computing the wind tunnel data. Despite this correction factor, VSAero still underestimates the moment coefficient. A similar result is obtained from other studies done on using CRM to verify aerodynamic analysis tools [6, 16, 32].

The predicted moment coefficient of AVL is far off. To see what caused this error, a composition curve is made of the different contributions to the moment coefficient for AVL (see Figure 4.12) and VSAero (see Figure 4.13). When comparing the two, it can be seen that AVL over estimates the moment coefficient of the horizontal tail plane. Since no error is made in computing the forces of the CRM by AVL, it is concluded that a positional error of the horizontal tail plane (i.e. moment arm) is the cause of the erroneous results of AVL.

The onset of flow separation of causes and upward curve change of the wind tunnel test results in Figure 4.11, while the wind tunnel test results of VGM showed a downward curve change (see Figure 4.5. From this observation, it can be concluded that flow separation is initiated on the main wing for VGM (causing an decreased moment), and on the horizontal tail plane of CRM (causing an increased moment).

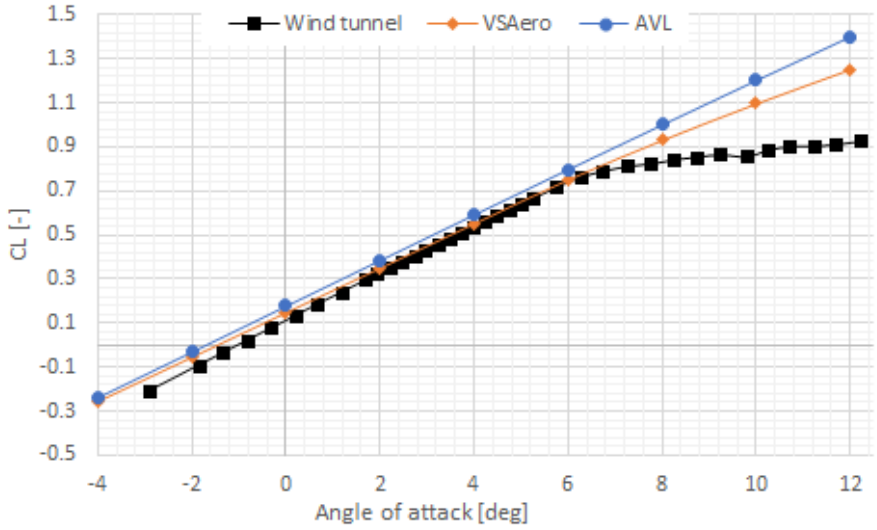


Figure 4.9: The lift curve of the CRM aircraft using different analysis tools. Re = 3.000,000 at Mach 0.7.

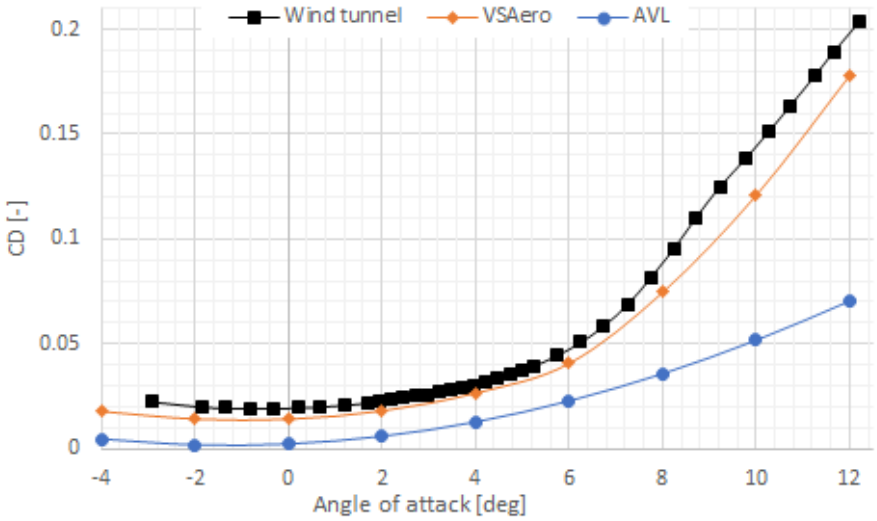


Figure 4.10: The drag curve of the CRM aircraft using different analysis tools. Re = 3.000,000 at Mach 0.7.

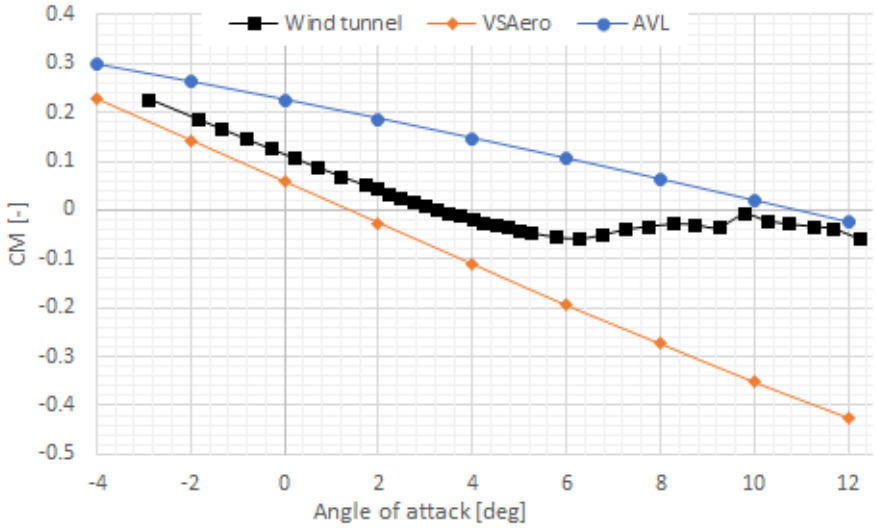


Figure 4.11: The pitch moment curve of the CRM aircraft using different analysis tools. Re = 3.000,000 at Mach 0.7.

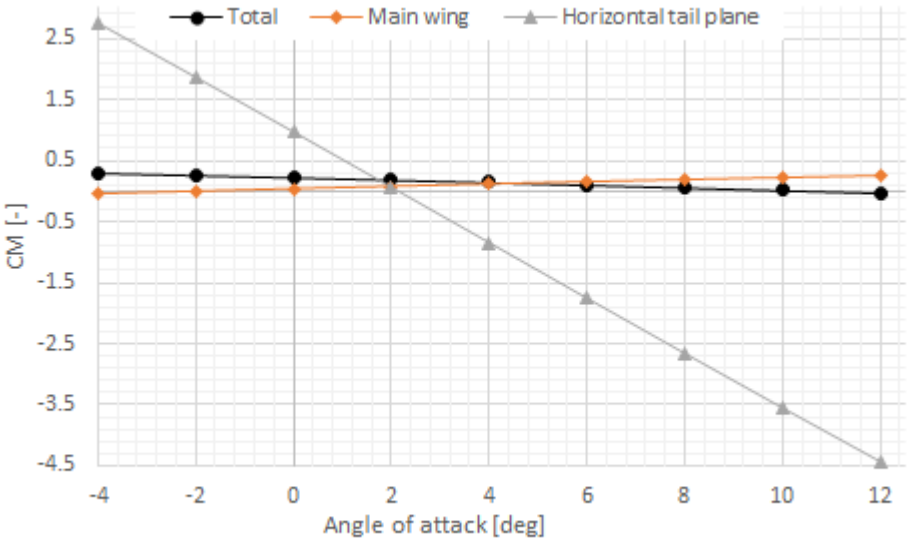


Figure 4.12: Calculated contribution by AVL of the main wing and horizontal tail plane to the moment curve C_m versus α for the CRM.

Figure 4.13: Calculated contribution by VSAero of the main wing and horizontal tail plane to the moment curve C_m versus α for the CRM.

Table 4.3: Comparison of key aerodynamic parameters computed by windtunnel tests, VSAero and AVL.

	Wind tunnel	AVL	Difference [%]	VSAero	Difference [%]
C_{L_0}	0.113	0.201	59.07	0.177	32.7
C_{L_α}	0.109	0.096	4.91	0.101	7.6
C_{D_0}	0.015	0.002	86.1	0.015	6.4
C_{M_α}	-0.037	-0.019	48.4	-0.043	12.9

4.2.2. Static longitudinal derivatives

The predicted stability derivatives by AVL, VSAero and wind tunnel for the CRM model can be seen in Table 4.4. Because the wind tunnel test of CRM was conducted without using movables, no control derivatives could be produced. Despite the increased complexity of the CRM model, and the higher Mach number, the margin of error stays similar as for the VGM.

Table 4.4: Longitudinal stability derivatives for 2 degrees angle of attack and no side slip

		Wind tunnel	AVL	Difference [%]	VSAero	Difference [%]
Static	C_{X_u}	0.017	0.007	56.1	0.021	21.4
	C_{Z_u}	0.718	0.462	38.8	0.904	25.9
Average error margin				45.9		23.6

4.3. Conclusion

VSAero reduces the error margin for predicting selected S&C derivatives by 32% compared to AVL for VGM aircraft, and by 22.3 % for CRM. The complexity of the aircraft model (i.e. number of defined airfoil sections along the span) and the Mach number has little influence on the accuracy of the aerodynamic analysis of VSAero.

There is a large error margin in the prediction of the moment coefficient curve by AVL. A composition curve is made to show all the contributions of the different elements to the moment coefficient. From this composition curve it can be concluded that this error is caused by erroneous positioning of the horizontal tail plane w.r.t. the CG.

5

Sensitivity study

This section describes the sensitivity of the VSAero results (i.e. change in lift and drag coefficient) to using different methods and VSAero input settings. The sensitivity study is applied to the VGM aircraft. The effect of the following attributes on the aerodynamic results is analysed:

- **Solid split method** - Creating a structured mesh of the surface of the solid requires splitting the geometry into quad faces using splitter curves. Two different methods have been identified in Chapter 3. The influence of the two methods on the aerodynamic results and run time are compared with each other.
- **Mesh density** - Also called mesh convergence study. An optimal number of points has to be found which results in the best combination of accuracy and run time.
- **Drag methods** - The drag of the aircraft can be computed by VSAero using three different methods: inviscid, viscid including skin friction and viscid including skin friction and boundary layer transpiration.
- **Wake and viscosity iterations** - Two loops exist in computing aerodynamic results by VSAero, one for the wake shape, and one for computing the viscous solution. The number of times that these loops will be iterated can be defined by the user.

5.1. Solid split method

A different method to split the aircraft into quad faces is introduced (see Chapter 3). This method uses curved leading edge splitter curves instead of straight ones. For clarification purposes, the two split methods are shown side by side in Figure 5.1.

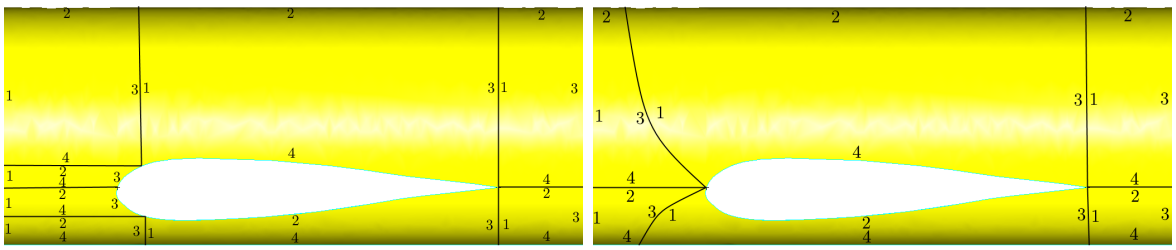


Figure 5.1: Comparison between the two split methods. The left method will henceforth be called "curved method" and the right method will be called "straight method".

Figure 5.2 compares the lift curve of the VGM aircraft when using the straight method or the curved method with the wing tunnel data. Figure 5.3 does the same for the drag curve. It can be seen that the split method has little effect on the accuracy of of the computed aerodynamic data.

The comparison result are quantified in Table 5.1. The curved method computes the results for a single run 12% faster than the straight method. This is caused by the fact that the curved method uses 15% less panels than the straight method. The difference in CL_0 error is 0.1 %, which is negligible.

From this analysis it can be concluded that the curved method is the advised meshing method for future analysis using the MMG.

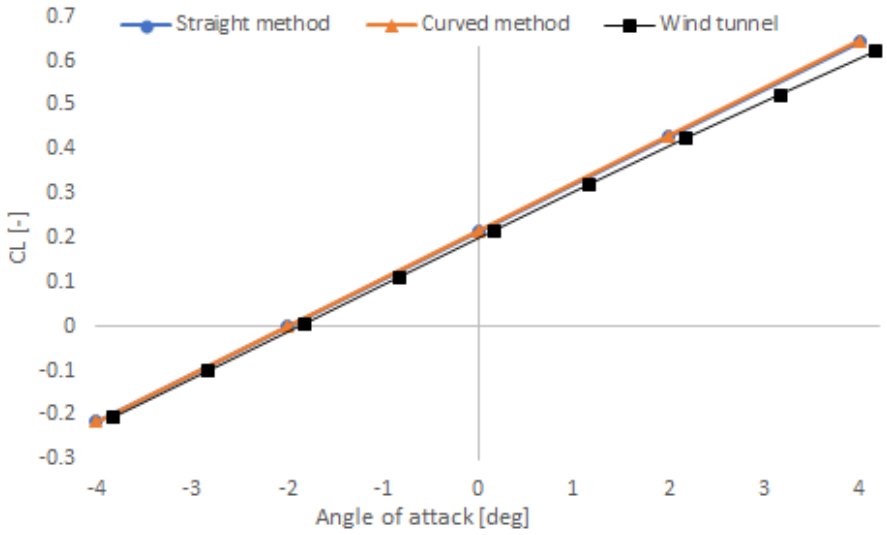


Figure 5.2: Comparison of the lift curve of the straight split method and the curved split method.

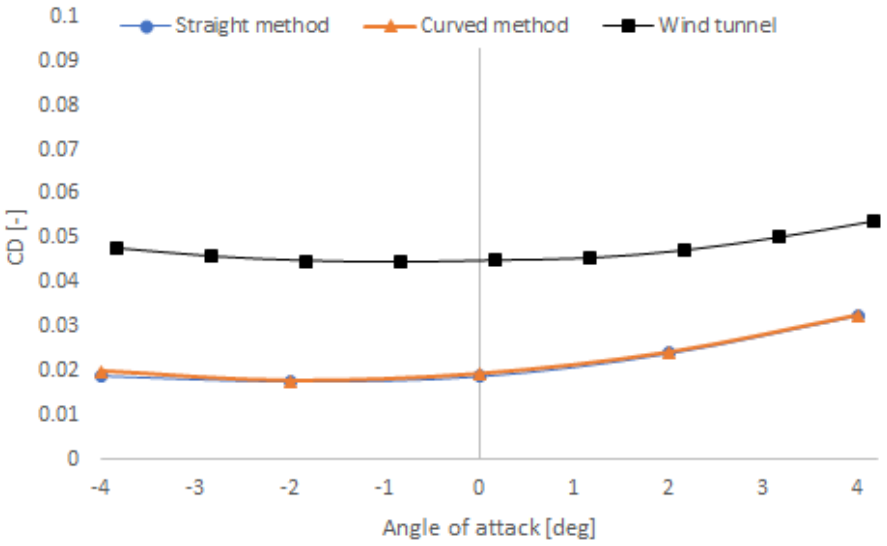


Figure 5.3: Comparison of the drag curve of the straight split method and the curved split method.

Table 5.1: Quantitative comparison between the straight method and the curved method applied on the VGM aircraft.

	Panels	Run time [s]	C_{L_0} Error [%]	CL_α Error [%]	CD_0 Error [%]
Straight method	11778	356	1.9	1.6	-91
Curved method	9968	314	2.0	1.7	-91

5.2. Mesh density

The MMG produces a fully structured mesh. The wing mesh is controlled by the spanwise pitch (i.e. maximum width between two spanwise mesh points) and chordwise number of points (see Figure 5.4). The fuselage mesh is controlled by the lateral pitch and longitudinal pitch (see Figure 5.5). The number of chordwise points has the largest influence on the results of the aerodynamic analysis because there is a larger variance in pressure in chordwise direction compared to spanwise direction.

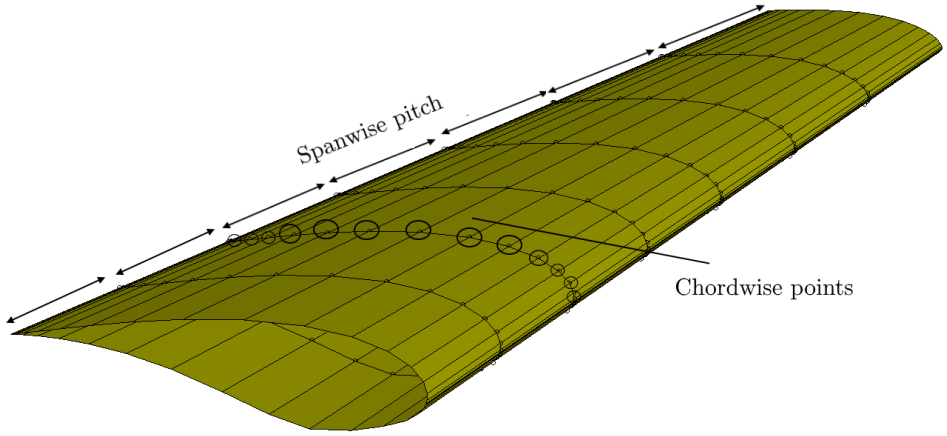


Figure 5.4: Description of chordwise number of points and spanwise pitch.

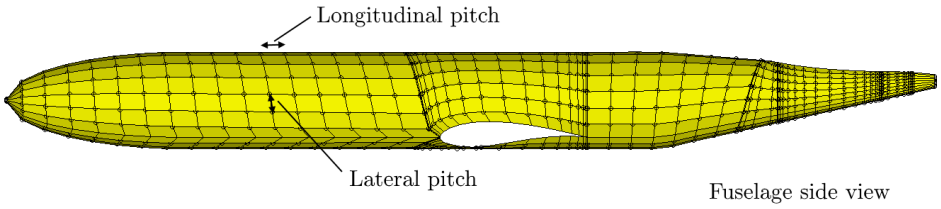


Figure 5.5: Description of longitudinal pitch and lateral pitch.

The change in lift coefficient at 0 angle of attack caused by a change in chordwise points can be seen in Figure 5.6. The lift coefficient at 0 angle of attack increases with increased mesh density. As the mesh density increases, the accuracy of the computed lift coefficient increases. The downside of increasing the chordwise number of of points is that also the run time of VSAero increases. Figure 5.7 shows the sensitivity of the drag coefficient when changing the number of chordwise points. Mesh convergence is achieved much earlier for the drag coefficient (approximately 25 points) than for the lift coefficient (approximately 40 points).

The error margin for estimating the lift coefficient at zero angle of attack only slightly decreases over 40 chordwise points. However, the run time of VSAero linearly increases when more points are used. Therefore, using 40 chordwise points is the optimal combination of a reasonable run time and accuracy the results computed by VSAero.

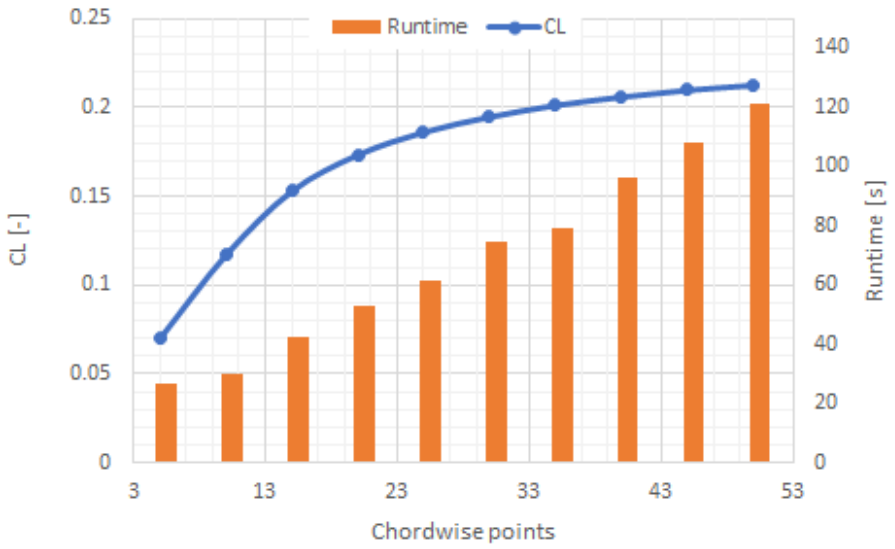


Figure 5.6: Effect of the number of chordwise points on the computed lift coefficient for VGM at 0 degrees angle of attack.

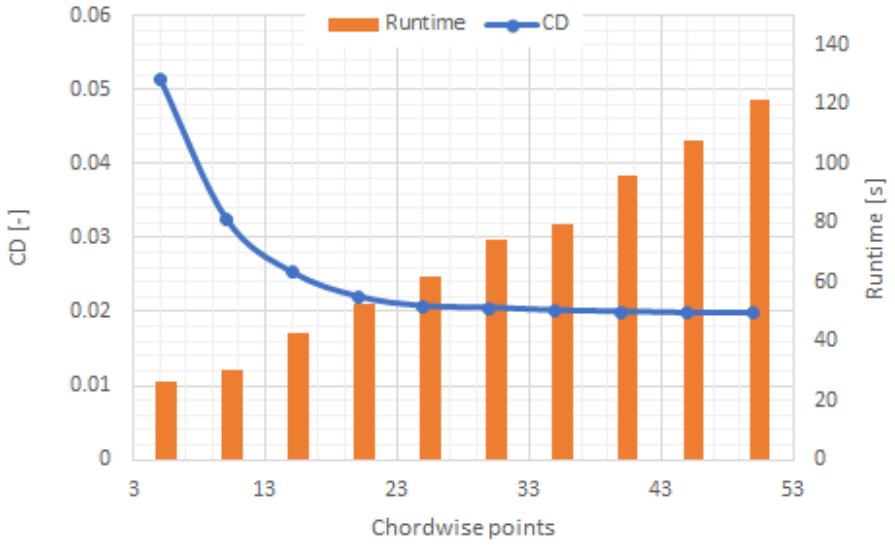


Figure 5.7: Effect of the number of chordwise points on the computed drag coefficient for VGM at 0 degrees angle of attack.

5.3. Drag prediction method

Using the results for the chordwise mesh convergence study, the effect of the drag prediction method is analysed. This is done by doing the same study, but then for three different drag prediction m in VSAero:

- Inviscid (IV): only taking induced drag into account.
- Viscous skin friction (VSF, method used in computations so far): skin friction is predicted by assuming a constant boundary layer thickness over the surface of the aircraft.
- Viscous skin friction and boundary layer transpiration (VBL): The movement of the boundary layer is calculated and adapted accordingly.

The effect of the different drag prediction methods on the lift coefficient can be seen Figure 5.8. It can be seen that there is little difference in the computed lift coefficient between IV and VSF. However, VBL method

seems to be very much off. At 45 chordwise points it seemed to have reached mesh convergence, while still being 20% off compared to reference value, after which as sudden jump in lift coefficient can be observed.

The effect on the drag coefficient can be seen in Figure 5.9. In this case, methods VSF and VBL are very similar. A decrease in computed drag coefficient can be seen when increasing the chordwise mesh density. Because the IV method shows the same trend, this is caused by the induced drag and not any of the viscous contributions. This result is unexpected since the induced drag is directly proportional to the lift coefficient.

The induced drag is calculated by integrating the contributions of all the panels over the surface area of the aircraft. More chordwise points (i.e. smaller panels) at the leading edge means that the suction peak at the leading edge is more accurately computed (see Figure 5.11). This suction peak at the leading edge rotates the resultant force vector of the airfoil to the leading edge point of the airfoil, hence reducing the computed drag.

Due to the instability of the computed lift coefficient when using the VBL method for drag prediction, for this research VSF will be used to compute the drag.

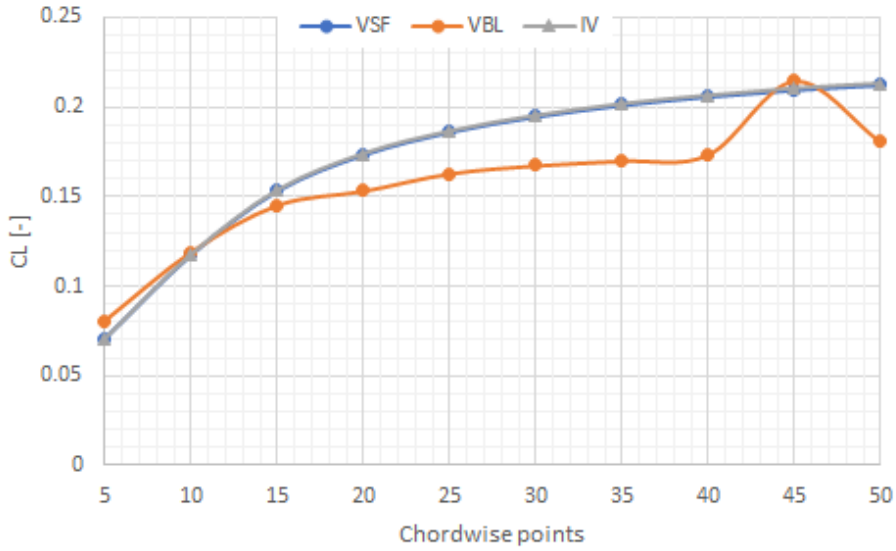


Figure 5.8: Mesh convergence study of the lift curve for three different drag prediction methods.

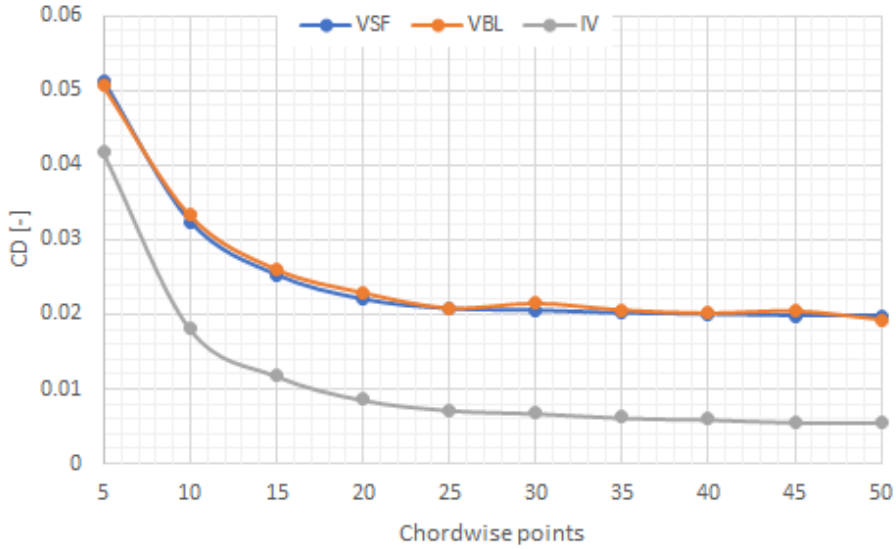


Figure 5.9: Mesh convergence study of the drag curve for three different drag prediction methods.

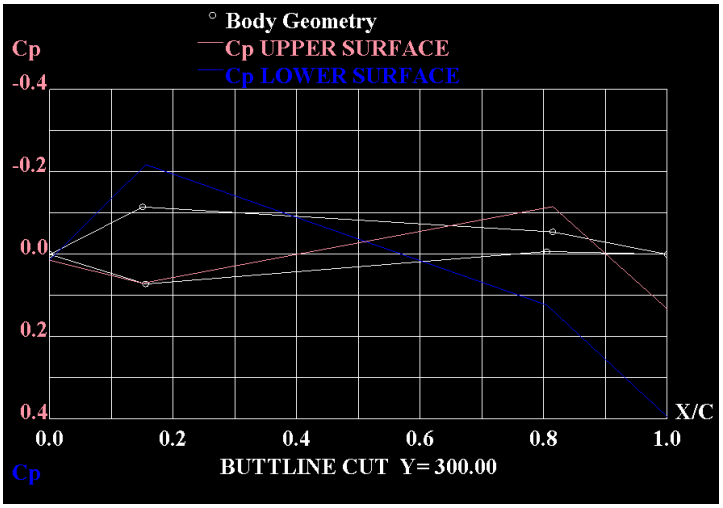


Figure 5.10: Cross sectional pressure distribution of the wing using 8 chordwise mesh points for top and bottom.

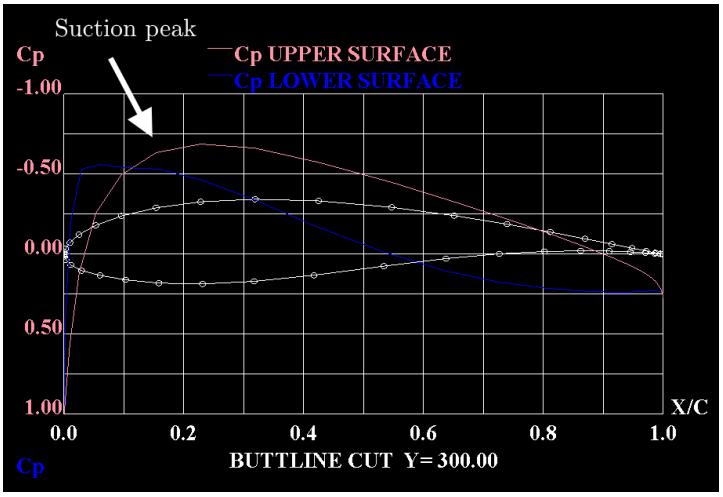


Figure 5.11: Cross sectional pressure distribution of the wing using 25 chordwise mesh points for top and bottom.

5.4. Wake and viscosity iterations

The step by step process of VSAero to compute the forces on the aircraft can be seen in Figure 5.12. The process includes two iteration loops: the viscous/potential iterations (NVPI) and the wake shape iterations (NWIT). VSAero has the option to calculate the boundary layer during the wake iteration loop (IBLTYP = 1), or after the wake iteration loop (IBLTYP = 0). To be able to see the effect of both iterations on the results, IBLTYP = 0 is chosen.

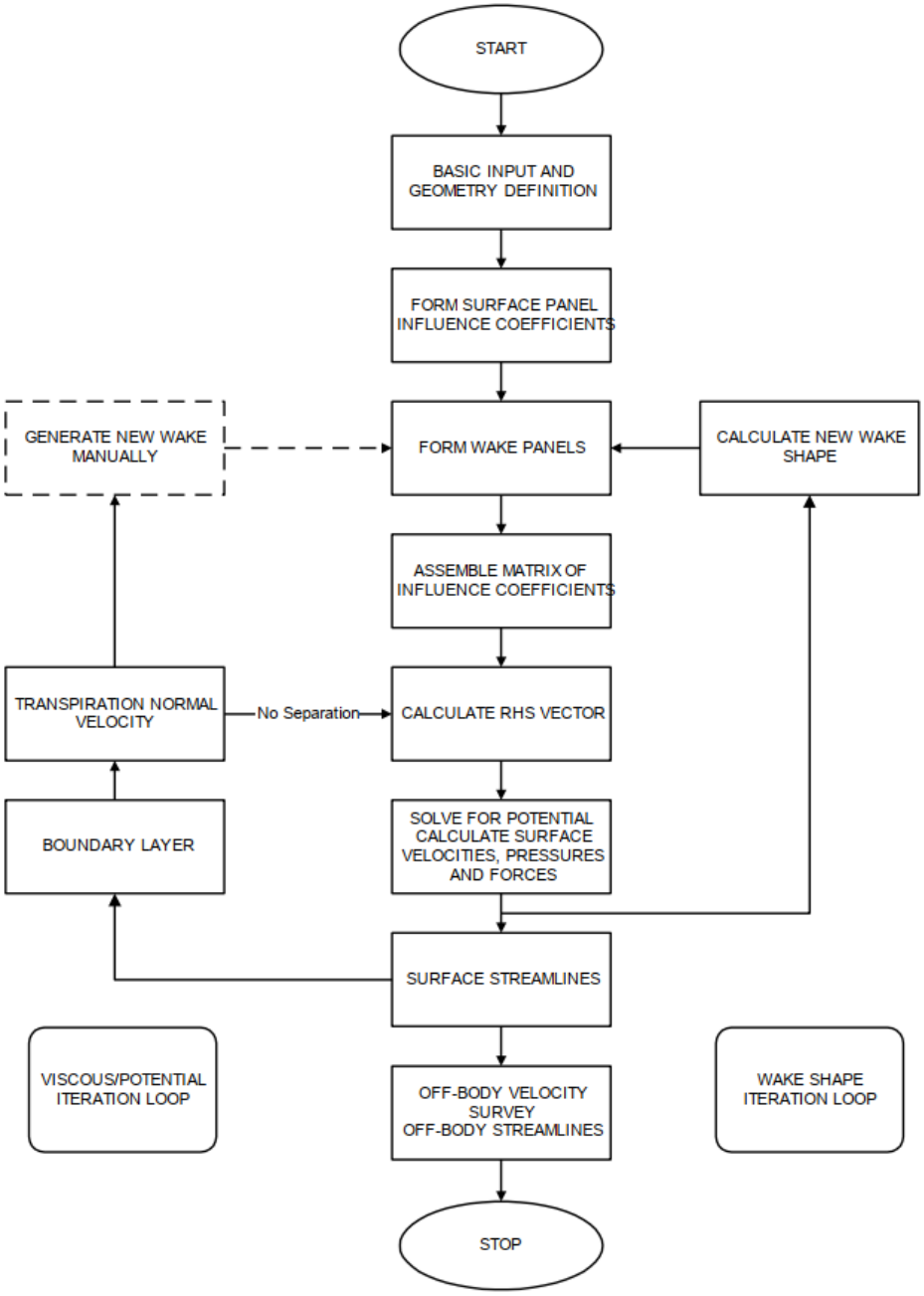


Figure 5.12: The step by step process in which VSAero computes the forces on the aircraft.[25, p. 44]

The analysis is done on the VGM aircraft, of which wind tunnel data is available. The effect of the Viscous/potential iterations (NVPI) and of the wake shape iterations (NWIT) on the lift coefficient, drag coefficient, and run time per iteration is computed.

Figures 5.13 and 5.14 show the effect of the NWIT and NVPI iterations on the computed lift and drag coefficient respectively. It can be seen that wake iterations (i.e. first 10 iterations) have little effect on the drag coefficient. The lift coefficient converges already after one wake iteration. From the iteration time it can be seen that one NWIT is almost 6 times more time consuming than one NVPI (i.e. iterations 10-30).

Figures 5.15 and 5.16 show the effect of NWIT and NVPI on the computed CL and CD for 8 degrees angle of attack to see if the angle of attack has an influence on how fast the solution convergences. It can be seen that this is not the case, Both CL and CD vary little after the first iteration.

The VSAero manual advises NVPI values of 3 or 4 to use as initial values and wake iteration values of 1, 2 or 3.[25] Because of the increased complexity of box wing aircraft (i.e. more components and intersections)

and the low effect on run time, for this thesis work, the NVPI value is chosen to be 6. The NWIT value is set at the maximum advised level of 3.

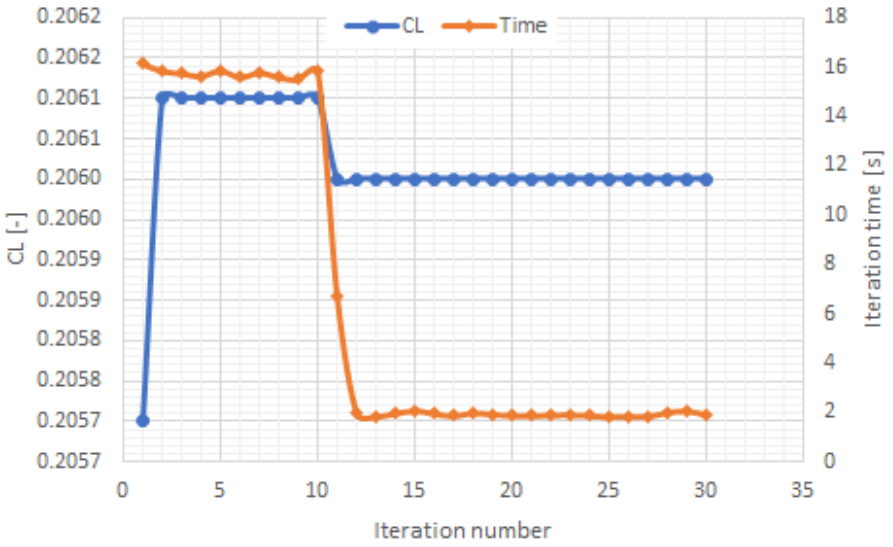


Figure 5.13: The effect of the increasing viscous/potential iterations on the lift coefficient for VGM aircraft at 0 degrees angle of attack, Mach = 0.13, Re = 500.000. The first 10 iterations are wake iterations, the last 20 are viscous iterations.

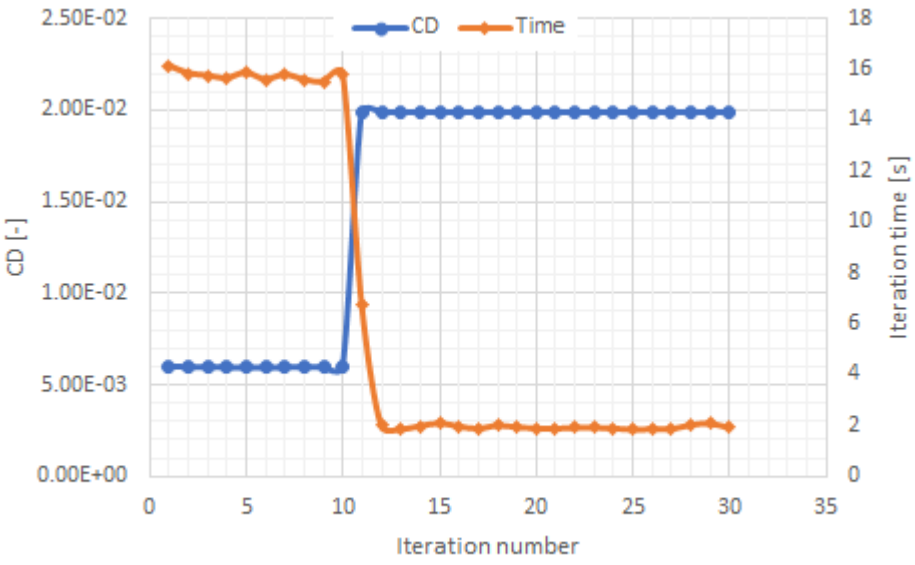


Figure 5.14: The effect of the increasing viscous/potential iterations on the drag coefficient for VGM aircraft at 0 degrees angle of attack, Mach = 0.13 and Re = 500.000. The first 10 iterations are wake iterations, the last 20 are viscous iterations.

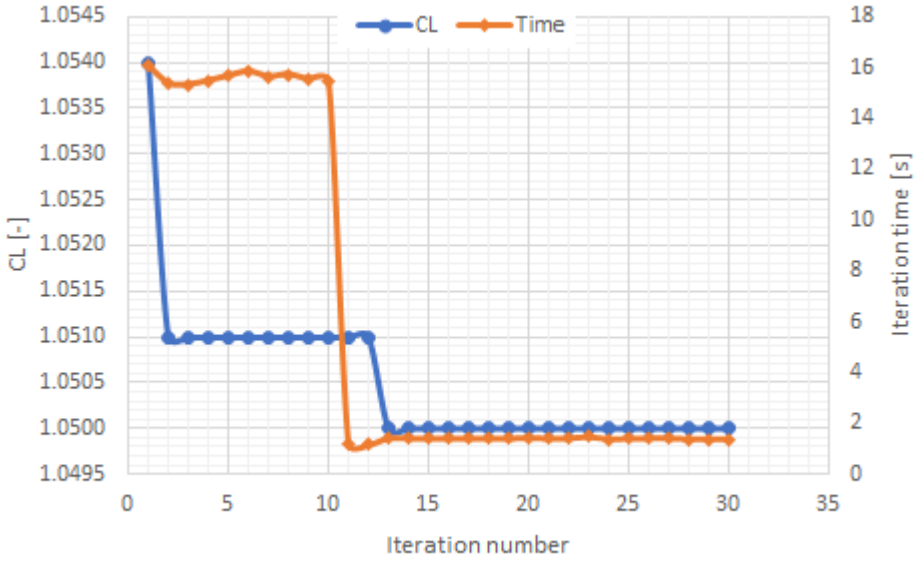


Figure 5.15: The effect of the increasing viscous/potential iterations on the lift coefficient for VGM aircraft at 8 angle of attack, Mach = 0.13 and Re = 500.000. The first 10 iterations are wake iterations, the last 20 are viscous iterations.

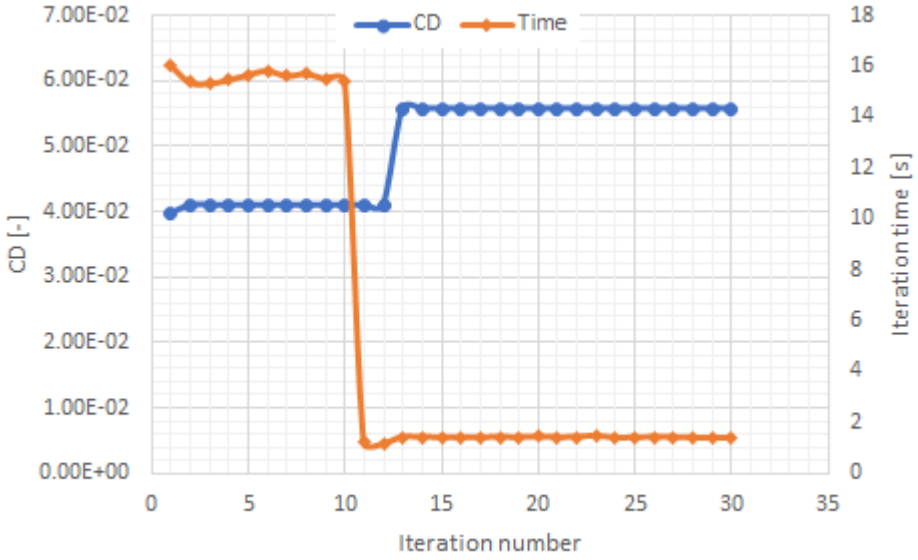


Figure 5.16: The effect of the increasing viscous/potential iterations on the lift coefficient for VGM aircraft at 8 angle of attack, Mach = 0.13 and Re = 500.000. The first 10 iterations are wake iterations, the last 20 are viscous iterations.

5.5. Conclusion

A number of settings were changed to see what the effect was on the VSAero computations, so that the optimal methods and settings can be defined. Curved leading edge split curves is the best method for splitting the surface of the solid into quad faces. A mesh density of 40 chordwise points will suffice in creating accurate results by VSAero. The drag prediction method should use the viscid solution, but exclude the effect of boundary layer transpiration because that results in unstable prediction of the lift coefficient. Finally, 3 wake shape iterations and 6 viscous iterations shall be used by VSAero to compute the aerodynamic results.

6

Time Study

This section tries to quantify the time that is saved when using the MMG instead of doing the same analysis manually. This helps to answer sub question 3. The comparison is done for a box wing aircraft with a total of 11 movables, 3 different angles of attack, 2 side slip angles, 2 Mach numbers, 3 movable deflection angles, and 2 different rotational rates around the 3 body axis. This requires an aerodynamic database of 300 cases to be able to compute trim point, from which the S&C derivatives can be derived. Of these 300 cases, there are 33 unique configurations (i.e. 11 movables deflected at 0, -10 and +10 degrees). The other cases are made up of changes in angle of attack, angle of side slip, etc of these same configuration which will take less time to manipulate.

Kulkarni estimated that it costs 30 hours to manually create an error free input file for one aircraft input file for VSAero. [29] However, this estimation is based on analysing a conventional aircraft. Since a box wing aircraft is geometrically more complex, it is assumed that it will take 35 hours to complete the input file for VSAero. For 32 cases this configuration needs to be changed. It is estimated that it takes 10 hours to change the configuration of the aircraft (e.g. deflect a movable). This comes down to a total of 320 hours. For 267 cases, the VSAero settings needs to be changed. This takes much less time than changing the configuration and is assumed to be 1 minute. For 267 cases this comes down to about 4.5 hours. In total, creating the input files for a database of 300 cases it will cost 35 + 320 + 4.5 = 359.5 hours.

It takes about 8 hours to analyse a working box wing model without any errors in the MMG. However, manipulating the aircraft for the other 32 configurations now only takes 5 minutes per configuration, which is equal to 2.7 hours in total. Hardly any time is saved in changing the input settings so this will still take about 1 minute, or 4.5 hours in total. Creating the aerodynamic database for predicting stability characteristics for a box wing aircraft using the MMG costs 8 + 2.7 + 4.5 = 15.2 hours.

This means that when comparing the two methods, a time reduction of 95.8% is achieved using the MMG. The results are summarised in Table 6.1.

Table 6.1: Comparison of the time it takes to create an aerodynamic database to estimate the stability characteristics for a box wing aircraft.

	Manual	MMG	unit
Create configuration	35	8	[hrs]
Manipulate configuration	10	0.12	[hrs]
Manipulate VSAro input	1	1	[min]
Number of cases [-]	300	300	[-]
Creation of full aerodynamic database	360	15	[hrs]

7

Case study

A case study is done to demonstrate the use of the tool. Initiator is used to perform the initial design of three different aircraft configurations with the same top level aircraft requirements (TLAR):

- Box wing with double vertical tail plane (BWD)
- Box wing with single vertical tail plane (BWS)
- Single wing aircraft with conventional tail (CON)

7.1. Generating the aircraft

The TLAR which have been used by Initiator can be seen in Table 7.1. The three aircraft can be seen in Figures 7.1, 7.2 and 7.3. BWS is sized so that the required vertical tail plane area is met, after which the rear wing is placed on top. Therefore, the position of the rear wing is higher for BWS than for BWD.

What stands out here, is that the span of CON is smaller than the span of BWD and BWS. During the design process executed by Initiator, a certain combination of thrust loading (W/T) and wing loading (W/S) is selected from the feasible design space of a performance diagram. This performance diagram includes constraints which are derived from the TLAR. The 3 aircraft have been designed from the same TLAR, but some constraints depend on properties that differ between CON, BWD, and BWS, meaning that the feasible design space can differ.

The reason why the wing size of CON is smaller than BWS and BWD is two fold. Firstly, CON has a lower estimated MTOW, so less surface area is needed to generate the same W/S. Secondly, CON is assumed to have high lift devices, something that is intentionally left out for BWD and BWS, since it is the intention of the Parsifal research team to leave them out for the box wing design. Therefore, the CL_{max} of CON is higher (2.6 for CON versus 1.8 for BWD and BWS) which affects the constraints in the performance diagram. A higher W/S can be selected for CON, so less surface area is required.

Initiator does not model control surfaces, so these have been manually added to the input files. The size of the control surfaces is based on the control surfaces used by Kulkarni [29]. The dimensions are summarised in Table 7.2.

Table 7.1: Reference parameters used to compute the Trim Point for the box wing aircraft.

Parameter	Value	Unit
Passengers	320	[-]
Cruise altitude	11000	[m]
Range	4000	[km]
Take-off distance	2200	[m]
Landing distance	1850	[m]
Approach speed	71	[m/s]
Cruise Mach number	0.78	[-]
Time to climb	35	[min]
Loiter time	30	[min]
Diverge range	500	[km]

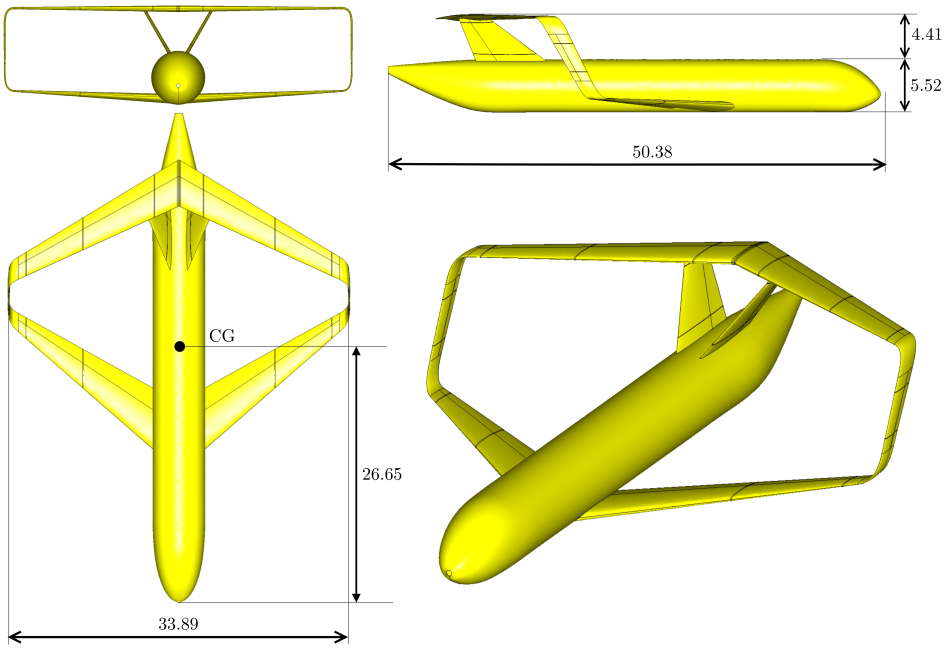


Figure 7.1: Dimensions of the BWD aircraft.

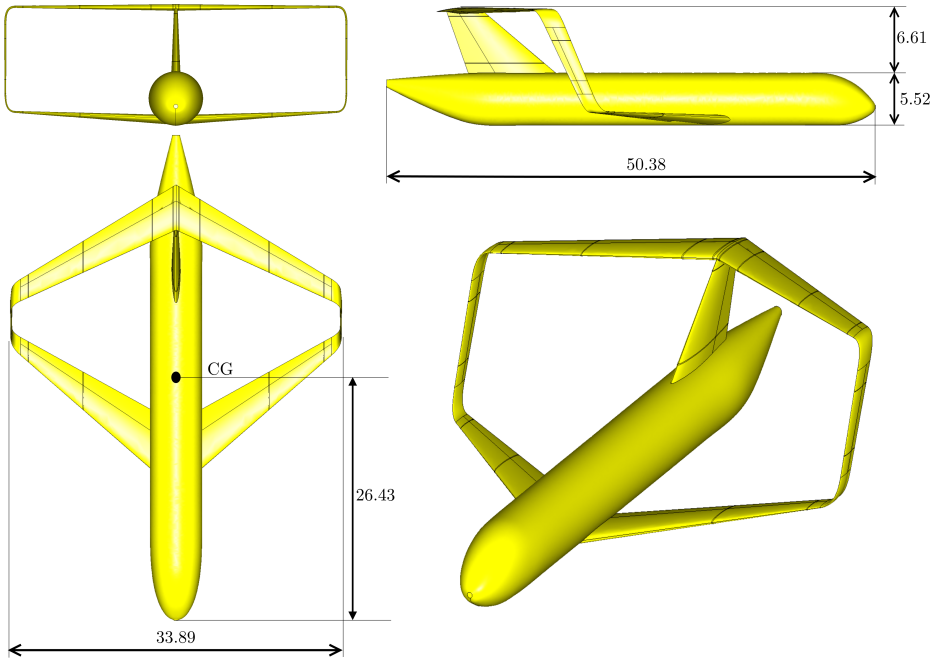


Figure 7.2: Dimensions of the BWS aircraft.

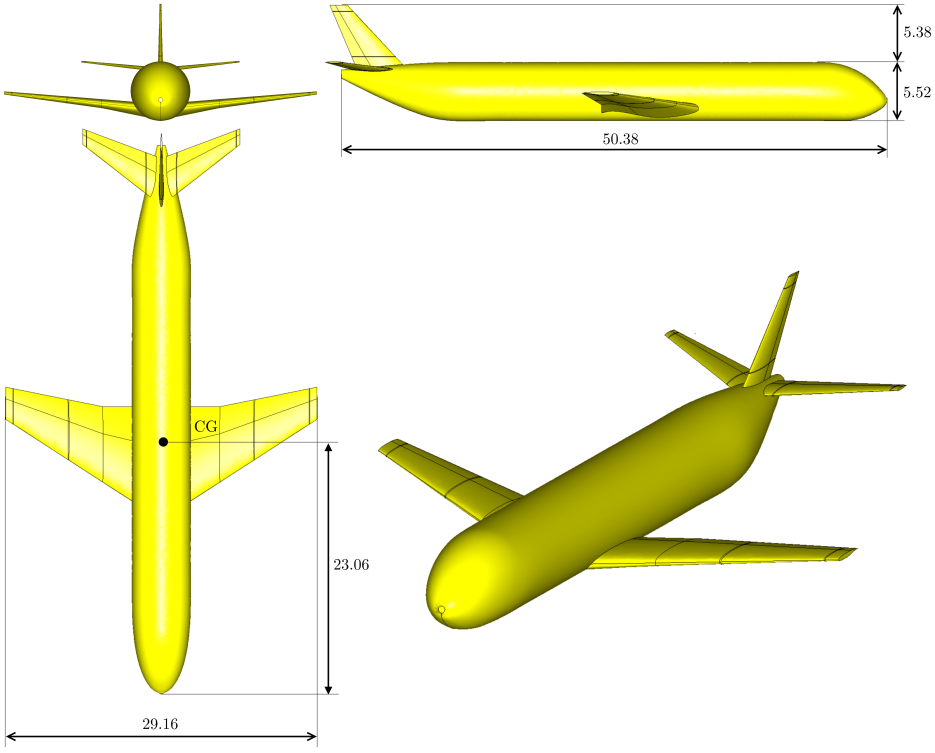


Figure 7.3: Dimensions of the CON aircraft.

7.2. Aerodynamic characteristics

The aerodynamic characteristics of the 3 aircraft are computed to see how they compare with each other. This can give more insight in the expected control deflection required for trimming the aircraft. Figure 7.4 shows

Table 7.2: Control surface information used to generate BWD, BWS and CON.

		BWD	BWS	CON	Unit
Main/front wing aileron	Chord fraction	0.35	0.35	0.35	[C]
	Spanwise start fraction	0.6	0.6	0.6	[b/2]
	Spanwise end fraction	0.95	0.95	0.95	[b/2]
	Surface area	1.11	1.11	2.52	$[m^2]$
Rear wing aileron	Chord fraction	0.35	0.35	-	[C]
	Spanwise start fraction	0.6	0.6	-	[b/2]
	Spanwise end fraction	0.95	0.95	-	[b/2]
	Surface area	1.02	1.02	-	$[m^2]$
Elevator	Chord fraction	0.35	0.35	0.35	[C]
	Spanwise start fraction	0.01	0.01	0.2	[b/2]
	Spanwise end fraction	0.15	0.15	0.95	[b/2]
	Surface area	2.68	2.68	0.76	$[m^2]$
Rudder	Chord fraction	0.3	0.3	0.3	[C]
	Spanwise start fraction	0.2	0.2	0.2	[b/2]
	Spanwise end fraction	0.9	0.9	0.9	[b/2]
	Surface area	2.03	3.22	0.73	$[m^2]$
Connecting element	Chord fraction	0.35	0.35	-	[C]
	Spanwise start fraction	0.2	0.2	-	[b/2]
	Spanwise end fraction	0.8	0.8	-	[b/2]
	Surface area	0.57	0.57	-	$[m^2]$

the lift curve comparison. There is little difference between BWD and BWS, which can be expected since they have the same lift surface area on the front and rear wing. The smaller wing of CON causes it to generate less lift. The drag curve of the three aircraft is compared in Figure 7.5. All aircraft generate comparable drag for different angles of attack.

Lastly, the moment curve is compared in Figure 7.6. The average moment coefficient for BWD and BWS is much lower than the one for CON, which is caused by the large rear wing.

A suspicious effect is observable for BWD at an angle of attack of 12 degrees, where a sudden upward data point is defined. After observing the geometries in Omni3D® (see Figures 7.7 and 7.7), it can be seen that a suction peak at high angles off attack arise at the rear bottom of the fuselage (i.e. creating an extra positive moment). This is caused by the wake lines shedding of the front wing root, which pierce the fuselage, and therefore create a low pressure region.

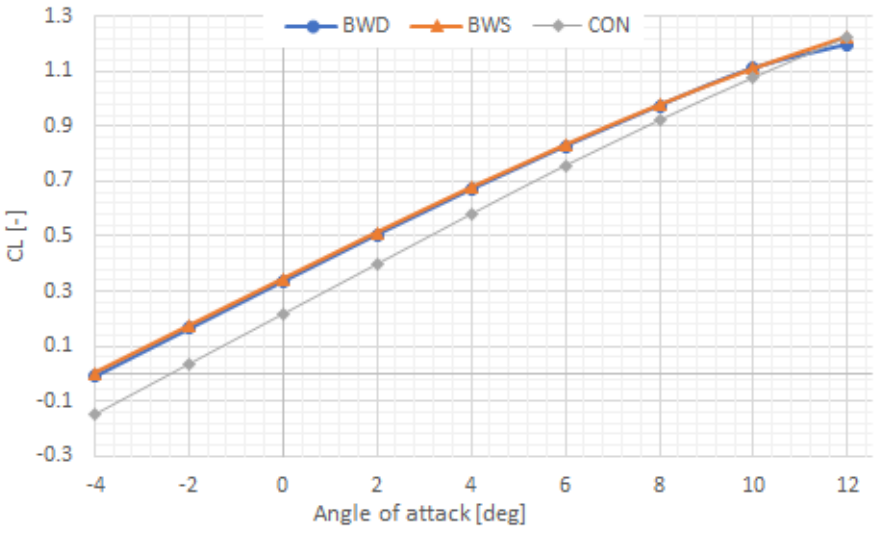


Figure 7.4: Comparison of the lift curve for the three different aircraft configurations.

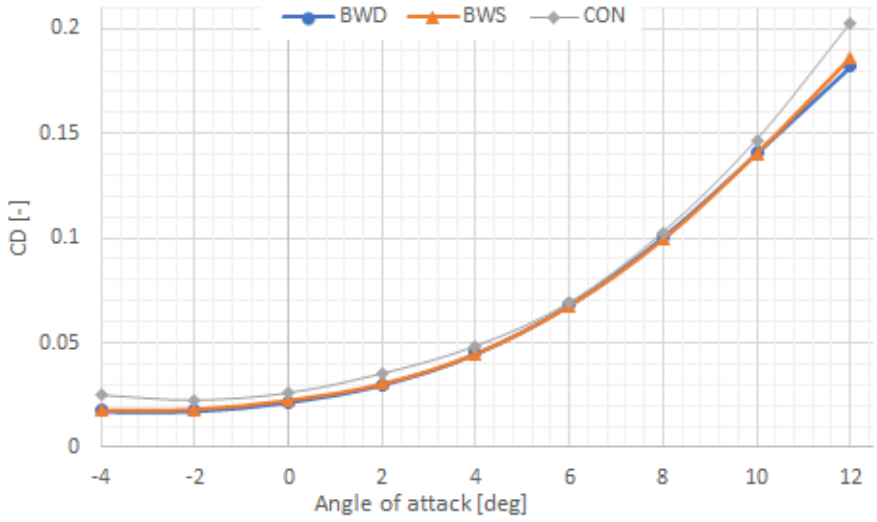


Figure 7.5: Comparison of the drag curve for the three different aircraft configurations.

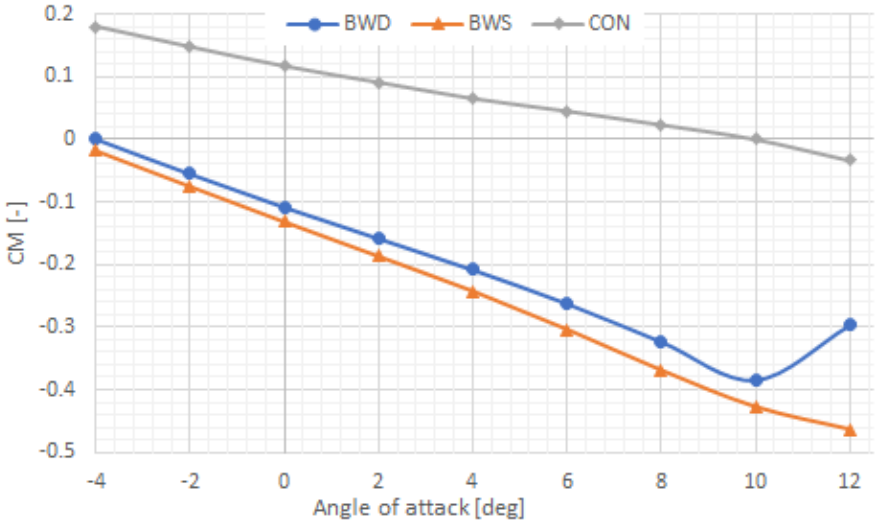


Figure 7.6: Comparison of the moment curve for the three different aircraft configurations.

7.3. Trimming the aircraft

PHALANX uses a created aerodynamic database of each aircraft in combination with reference parameters and inertial values of the aircraft(see Table 7.3), to compute the trim point. The database for BWD and BWS consists of 264 cases and the database for CON consists of 192 cases. CON needs less cases because it has less movable surfaces. An example of an aerodynamic database is shown in Appendix C. The trim conditions of the three aircraft can be seen in Table 7.4. Using these trim setting, the S&C derivatives will be computed, and can be seen in the next section.

7.4. Predicting Stability and control derivatives

The S&C derivatives are predicted for the trim settings shown in Table 7.4. The list of longitudinal computed static, dynamic, and control derivatives for BWD, BWS, and CON can be seen in Table 7.5.

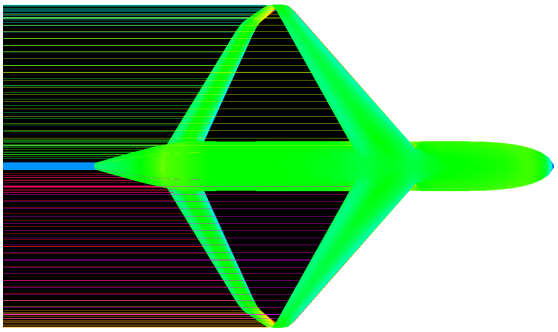


Figure 7.7: Bottom view of non deflected wake VSAero analysis of BWD, visualised by Omni3D®

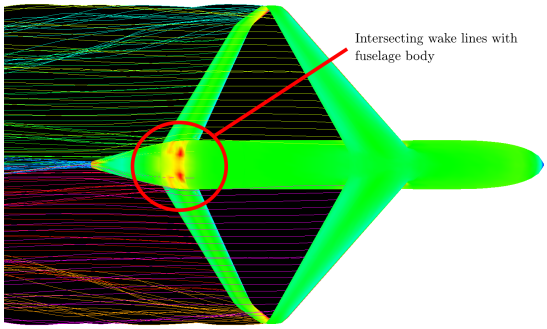


Figure 7.8: Bottom view of deflected wake VSAero analysis of BWD, visualised by Omni3D®.

Table 7.3: Reference parameters used to compute the Trim Point for the box wing aircraft in cruise conditions.

	BWD	BWS	CON	Unit
Total thrust	210	210	210	[kN]
Weight	138112	138112	131156	[kg]
CG (x,y,z)	(23.6, 0, 1.1)	(26.4, 0, 1.3)	(23.1, 0, -0.6)	[m]

Table 7.4: Trim Point for the three aircraft.

Variable	BWD	BWS	CON	Unit
α	-0.05	-0.05	0.03	[deg]
β	0	0	0	[deg]
Mach	0.78	0.78	0.78	[-]
CL	0.01	0.02	0.41	[-]
CD	0.014	0.016	0.038	[-]
δ_e	-0.12	-0.01	0.03	[deg]

Table 7.5: Longitudinal stability and control derivatives the double fin box wing, single fin box wing, and Citation I.

	Derivative	BWD	BWS	CON
Static	C_{X_u}	-0.013	-0.012	-0.018
	C_{Z_u}	-0.716	-0.737	-0.529
	C_{m_u}	-0.143	-0.180	0.219
	C_{X_α}	0.229	0.242	0.114
	C_{Z_α}	-4.92	-4.88	-5.24
	C_{m_α}	-1.44	-1.62	-0.73
Dynamic	C_{X_q}	0.29	0.41	-0.18
	C_{Z_q}	-8.9	-8.2	-8.5
	C_{m_q}	-29.5	-29.3	-18.2
Control	$C_{X_{\delta_e}}$	-0.012	-0.010	-0.020
	$C_{Z_{\delta_e}}$	-0.186	-0.153	-0.515
	$C_{m_{\delta_e}}$	-0.623	-0.511	-1.870

When comparing the longitudinal stability and control derivatives it can be seen from C_{m_α} that all aircraft are statically stable. A disturbance in angle of attack will result in a pitch down moment, which decreases the angle of attack. It is interesting to note that BWD and BWS are twice as stable as CON. This could be expected since the size of the rear wing for BWD and BWS, which generates the negative moment, is significantly larger than the size of the horizontal tail plane for CON.

The effect of the pitching velocity on the moment around the y-axis is described with the parameter C_{m_q}. It can be seen that the values for the two box wing aircraft are almost ten times as high as for the conventional aircraft. This is due to the large horizontal tail of both box wing aircraft, creating a large pitch damping force. This means that box wing aircraft might have problems in changing the pitch attitude of the aircraft.

From the control derivatives it is apparent that the effectiveness of the elevator of CON is much higher than both BWD and BWS, while the effective surface area of the elevator is the smallest (see Table 7.2. However, the effectiveness of the elevator depends on the relative lift increase when deflected. Compared to CON, the relative lift increase due to the elevator deflection is smaller for BWD and BWS because the relative size of the elevator is smaller (i.e. 0.6 of the horizontal tail plane span for CON versus 0.14 of the rear wing span for BWS and BWD).

The list of lateral computed static, dynamic, and control derivatives for BWD, BWS, and CON can be seen in Table 7.6.

Table 7.6: Lateral S&C derivatives for BWD, BWS, and CON.

	Derivative	BWD	BWS	CON
Static	C_{Y_β}	-1.17	-1.58	-0.59
	C_{l_β}	-0.13	-0.20	-0.11
	C_{n_β}	0.082	0.187	0.163
Dynamic	C_{Y_p}	-0.67	-0.98	0.40
	C_{l_p}	-1.50	-1.58	-1.10
	C_{n_p}	0.14	0.19	-0.32
	C_{Y_r}	4.63	6.34	3.65
	C_{l_r}	0.64	1.06	0.87
	C_{n_r}	-0.69	-5.15	-1.24
Control	$C_{Y_{\delta_a}}$	-0.084	-0.090	0.002
	$C_{l_{\delta_a}}$	-0.249	-0.257	-0.236
	$C_{n_{\delta_a}}$	0.019	0.020	0.024
	$C_{Y_{\delta_r}}$	0.223	0.258	0.160
	$C_{l_{\delta_r}}$	0.041	0.051	0.029
	$C_{n_{\delta_r}}$	-0.086	-0.105	-0.145

The derivative C_{n_β} is called the static directional stability. For good controllability of the aircraft it is desirable that an angle of side slip causes a positive moment around the z axis, thus reducing the side slip angle. It can be seen in Table 7.6 that all aircraft configurations are directionally stable.

The side force due to roll rate is described by the derivative C_{Y_p}. This derivative stands out because both BWD as BWS have a negative C_{Y_p} while CON has a positive C_{Y_p}. The main negative contribution to C_{Y_p} comes from the vertical tail. The local increase in angle of attack for the vertical tail creates a negative side force. The main positive contribution comes from the swept back main wing. The local decrease in angle of attack for the left wing decreases the resultant force, while on the right wing the resultant force increases (see Figure 7.9). The X and Y components of these resultant forces increases the C_{Y_p} for a back swept wing, but decreases it for a front swept wing. This is why C_{Y_p} is more negative for BWD and BWS (having one forward and backward swept wing), than for CON (having a single back swept wing). The same effect is responsible for the higher value of C_{n_p} for the BWD and BWS.

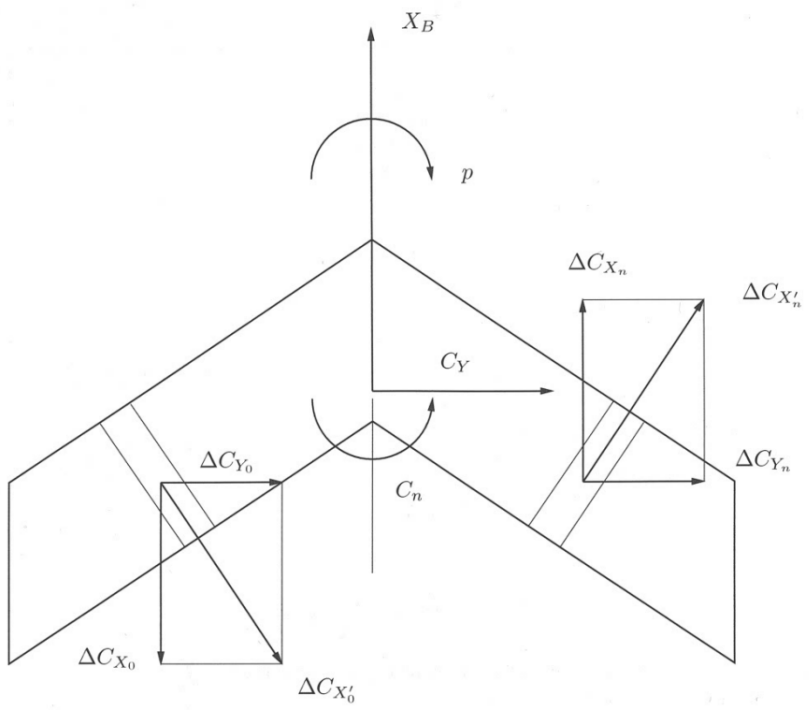


Figure 7.9: The change in resultant forces due to a roll rate for a backward swept wing.[21, p 225]

The control derivative that stands out when comparing the lateral S&C derivatives is are the derivatives w.r.t. δ_a. When doing the analysis, BWD and BWS use both their front wing aileron as their rear wing aileron deflection, while CON only deflects the aileron on the main wing.

For straight wings, $C_{Y_{\delta_a}}$ can be taken as 0, while for backward swept wings, $C_{Y_{\delta_a}} > 0$ and for forward swept wings (i.e. rear wing for BWD and BWS) $C_{Y_{\delta_a}} < 0$. The forward swept rear wing is the reason why this derivative is negative for BWD and BWS.

8

Conclusions and Recommendations

8.1. Conclusions

A methodology is developed which supports prediction of stability and control derivatives accurately and quickly, to support flight mechanics analysis for box wing aircraft configurations using physics based methods.

VSAero is chosen to be the best fit for performing the aerodynamic analysis of the box wing aircraft because of the increased accuracy of the analysis method (3D Panel method vs Vortex lattice). The error margin is reduced by 32% for VGM and 22% for CRM, when using VSAero instead of AVL to predict the S&C derivatives. The methodology that is developed reduces preprocessor time of VSAero by 96%.

A fully structured mesh of the aircraft surface is created by splitting the surface into faces with four edges using splitter curves. A new method for splitting the surface, which uses curved split curves at the leading edge, is compared with using straight split lines. This new method has little influence on the accuracy of the results while decreasing the computation time by 12%, which is why it can be concluded that this is the dominant strategy for splitting solids in the MMG.

The drag prediction method which showed the most stable results is when boundary layer transpiration is neglected. When including this in the boundary layer calculations, the lift coefficient prediction is unstable and therefore becomes unreliable. Only including skin friction by predicting a constant boundary layer thickness gives more stable results.

Using the newly developed methodology, 3 different aircraft configurations have been processed and analysed (2 box wing aircraft and one conventional, see Figure 8.1), to compute their S&C derivatives. The difference in configuration is observable when computing the S&C derivatives. From this comparison it can be concluded that controllability problems can be expected for the box wing aircraft with respect to the longitudinal controllability. The large rear wing creates a large pitch down moment, which can cause problems when trying to control the pitch behaviour of the aircraft.

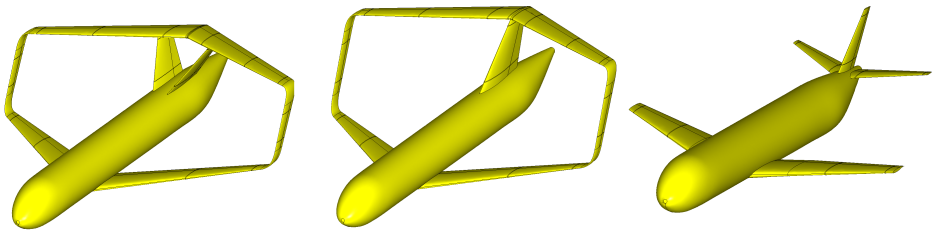


Figure 8.1: The three aircraft which have been analysed for the case study.

In conclusion, a quick and accurate method is developed that is capable of predicting stability and control derivatives using physics based methods. This opens the door for performing flight mechanics analysis on box wing aircraft configurations, which brings us one step closer to the use of boxing aircraft in commercial operations.

8.2. Recommendations

This section lists the recommendations to further develop the work that has been done in this Thesis. The recommendation are divided into three sections: General, VSAero, and MMG.

8.2.1. General

The MMG is verified using conventional aircraft. It would benefit the verification and validation of studies into box wing aircraft if more wind tunnel tests would be performed on box wing aircraft configurations. Similar as for the CRM, these tests can then be used by anyone to verify and validate their analysis tool.

Aside from predicting the trim point, PHALANX is able to compute the stability modes of aircraft. Comparing the stability modes of the different aircraft helps in establishing the airworthiness of the aircraft.

8.2.2. MMG

The MMG is currently focused on the aerodynamic analysis. It would be favourable for the development of the box wing aircraft to also include the structure of the aircraft. This can be used to improve estimations of weight prediction, which will also improve the accuracy of the S&C derivatives.

No option is available yet to model the engines or the landing gear in VSAero. These will significantly affect the CG estimation of VSAero and thus the calculation of the S&C derivatives. For instance, the location of the engine thrust will have a large influence on the trim settings.

PHALANX and VSAero are still controlled outside of the MMG environment. If these tools could be remotely controlled by the MMG, it would allow the user to create an automatic design flow for possible optimization purposes.

8.2.3. VSAero

Further research should be done on the drag estimation of VSAero. The recommended setting advises boundary layer transpiration to be included into the drag prediction, while this thesis shows that this results in unstable computations of the lift coefficient, which is why this drag contribution is left out.

The piercing of the fuselage by wake lines at high angles of attack poses a problem, because it influences the generated moment coefficient at high angles of attack. VSAero has settings which combat this effect from happening, but they seem to not work sufficiently to stop this effect from happening.

A

Example JSON input file

This appendix shows an example input file for generating a single wing of the aircraft.

```
"ExampleWing": {
        "is_mirrored": "True",
        "is_connecting_element": "False",
        "position": {
                "point": [12.7086, 0.0, -0.5957],
                "orientation": {
                        "x_vector": [-1.0, 0, 0.0],
                        "y_vector": [0, -1.0, 0.0]},
                "offset_vector":[0, -1, 0]},

        "parameters": {
                "airfoils": [
                        ["from_file","Airfoils/Profile_ExampleWing_1ID"],
                        ["from_file","Airfoils/Profile_ExampleWing_1ID"],
                        ["from_file","Airfoils/Profile_ExampleWing_1ID"]
                ],
                "kink_indices": [1],
                "span_positions": [0.0, 0.376999263096264, 1.0],
                "airfoil_thickness": [100, 100, 100],
                "airfoil_cant": [
                        ["streamwise"],
                        ["streamwise"],
                        ["streamwise"]
                ],
                "follow_dihedral":
                        [0, 1, 1]},

        "rails": {
                "rotation":
                [[0.0, 0.0, 0.0], [6.0, 0.0, 0.0], [6.0, 0.0, 0.0]],
                "scaling":
                [[7.27033, 7.27033, 7.27033],
                [3.66931, 3.66931, 3.66931],
                [1.78228, 1.78228, 1.78228]],
                "le_point_list":
                [[-0.0, 0.0, 0.0],
                [-3.601013626105942, 6.73374816264592, 0.7077454511065534],
```

```
                [-9.55177895185023, 17.86143587481365, 1.8773125583692076]],
                "le_beziers_control_point_list":
                [],
                "te_point_list":
                [[-7.27033, 0.0, 0.0],
                [-7.270323626105942, 6.73374816264592, 0.7077454511065534],
                [-11.33405895185023, 17.86143587481365, 1.8773125583692076]],
                "te_beziers_control_point_list":
                [],
                "dihedral_point_list":
                [[0.0, 0.0, 0.0],
                [0.0, 6.73374816264592, 0.7077454511065534],
                [0.0, 17.86143587481365, 1.8773125583692076]],
                "dihedral_beziers_control_point_list":
                [],
                "twist_axis_point_list":
                [[-0.0, 0.0, 0.0],
                [-3.601013626105942, 6.73374816264592, 0.7077454511065534],
                [-9.55177895185023, 17.86143587481365, 1.8773125583692076]],
                "twist_axis_beziers_control_point_list":
                [],
                "twist_points":
                [[0.0, 0.0, 0.0],
                [0.0, 6.73374816264592, 0.0],
                [0.0, 17.86143587481365, 0.0]],
                "twist_beziers_control_point_list":
                []
        }

},
```

B

ParaPy graphical user interface

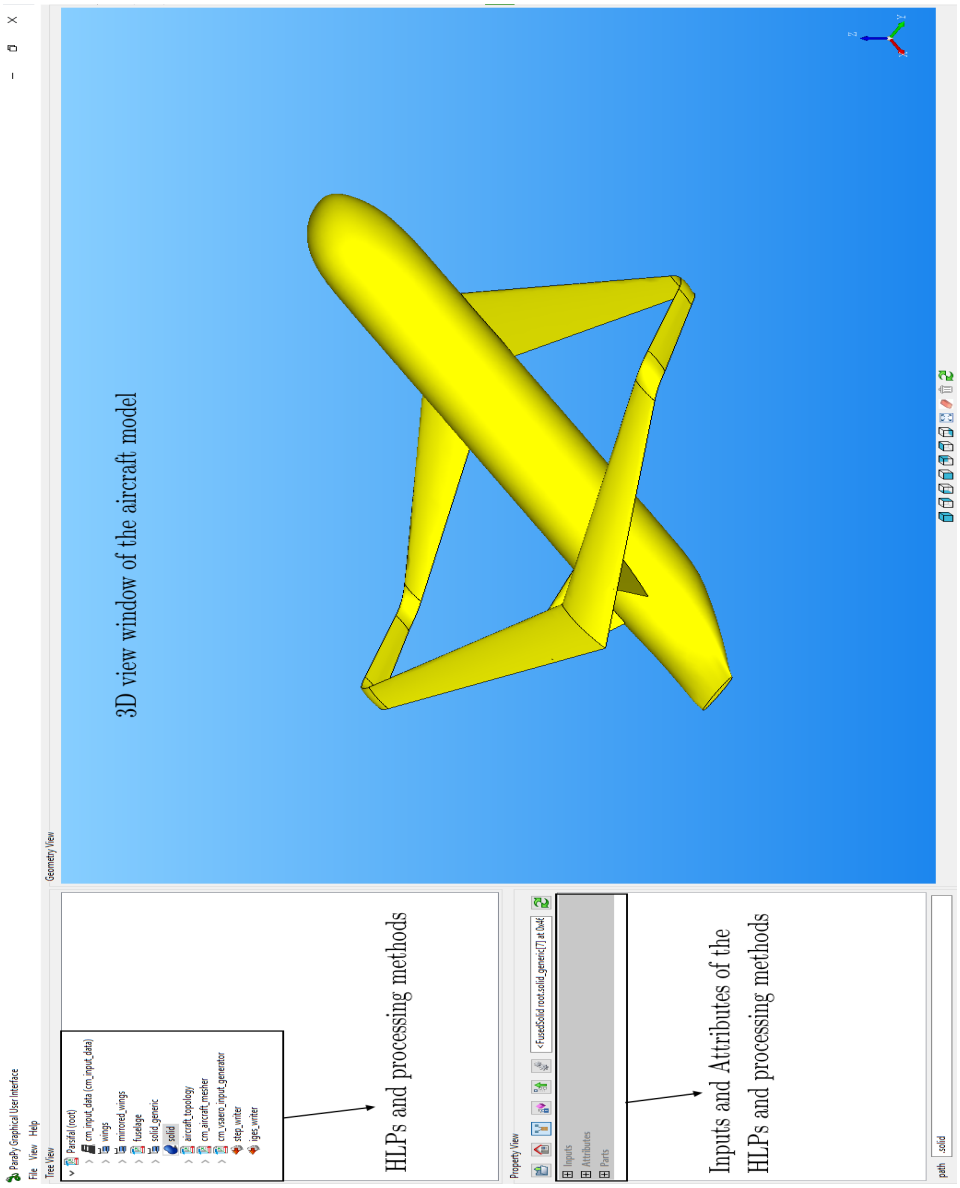


Figure B.1: The graphical user interface of ParaPy.

C

Example aerodynamic database

The appendix shows an example of an aerodynamic database which can be used by PHALANX to create the trim point of the aircraft. This particular database is made for 3 angles of attack, 3 angles of side slip, 2 Mach numbers, 2 different rates around the body axis x, y and z, and two control surface deflection angles.

```
{

"alpha": array([-4.,  0.,  4.]),
"beta": array([-3.,  0.,  3.]),
"mach": array([ 0.5,  0.7]),
"Sref": 284.206,
"cref": 4.4514,
"Pref": [26.425, 0.0, 1.313],
"bref": 32.9,

"CFX": array([[[ 0.0127  ,  0.01358 ],
        [ 0.01538 ,  0.01651 ],
        [ 0.0127  ,  0.01358 ]],

       [[ 0.01656 ,  0.0188  ],
        [ 0.01958 ,  0.0208  ],
        [ 0.01656 ,  0.0188  ]],

       [[-0.0131  , -0.006638],
        [-0.01067 , -0.004944],
        [-0.0131  , -0.006638]]]),

"CFY": array([[[ 0.0825  ,  0.08767 ],
        [ 0.002168,  0.003017],
        [-0.0825  , -0.08767 ]],

       [[ 0.08019 ,  0.0872  ],
        [ 0.002394,  0.003364],
        [-0.08019 , -0.0872  ]],

       [[ 0.07807 ,  0.08468 ],
        [ 0.002621,  0.003624],
        [-0.07807 , -0.08468 ]]])}

"CFZ": array([[[  1.10000000e-02,   4.72400000e-04],
```

```
          [   1.06200000e-02,   -9.39500000e-05],
          [   1.10000000e-02,    4.72400000e-04]],

         [[   3.27400000e-01,    3.47800000e-01],
          [   3.29900000e-01,    3.43300000e-01],
          [   3.27400000e-01,    3.47800000e-01]],

         [[   6.44300000e-01,    6.78700000e-01],
          [   6.45000000e-01,    6.77800000e-01],
          [   6.44300000e-01,    6.78700000e-01]]]),

"CMX": array([[[-0.009117 , -0.009337 ],
          [-0.0004144, -0.0005707],
          [ 0.009117 ,  0.009337 ]],

         [[-0.009814 , -0.01017  ],
          [-0.0004341, -0.0006057],
          [ 0.009814 ,  0.01017  ]],

         [[-0.01041  , -0.01005  ],
          [-0.0004474, -0.000641 ],
          [ 0.01041  ,  0.01005  ]]]),

"CMY": array([[[-0.0498 , -0.01623],
          [-0.05359, -0.01826],
          [-0.0498 , -0.01623]],

         [[-0.132  , -0.1439 ],
          [-0.1443 , -0.1314 ],
          [-0.132  , -0.1439 ]],

         [[-0.231  , -0.2445 ],
          [-0.2348 , -0.2434 ],
          [-0.231  , -0.2445 ]]]),

"CMZ": array([[[ 0.009834 ,  0.01174  ],
          [ 0.0009336,  0.001317 ],
          [-0.009834 , -0.01174  ]],

         [[ 0.00886  ,  0.01136  ],
          [ 0.00103  ,  0.00146  ],
          [-0.00886  , -0.01136  ]],

         [[ 0.008056 ,  0.01118  ],
          [ 0.001133 ,  0.001578 ],
          [-0.008056 , -0.01118  ]]]),

"CMYp": array([[[ -2.85000000e-01,    1.85000000e-01],
          [   5.00000000e-03,    2.50000000e-02],
          [  -2.85000000e-01,    1.85000000e-01]],

         [[   5.50000000e+00,   -1.50000000e+00],
          [  -0.00000000e+00,   -5.00000000e-02],
          [   5.50000000e+00,   -1.50000000e+00]],

         [[   4.50000000e-01,   -2.60000000e+00],
```

```
        [ -1.00000000e-01,  -1.00000000e-01],
        [  4.50000000e-01,  -2.60000000e+00]]]),

"CFXp": array([[[-0.105 ,  0.16  ],
        [-0.035 ,  0.065 ],
        [-0.105 ,  0.16  ]],

       [[-0.025 ,  0.02  ],
        [-0.025 , -0.015 ],
        [-0.025 ,  0.02  ]],

       [[-0.075 ,  0.0805],
        [-0.005 , -0.01  ],
        [-0.075 ,  0.0805]]]),

"CMXp": array([[[ 0.6285  ,  0.812   ],
        [ 0.61405 ,  0.78955 ],
        [ 0.6285  ,  0.812   ]],

       [[ 0.5985  ,  0.7945  ],
        [ 0.6143  ,  0.79485 ],
        [ 0.5985  ,  0.7945  ]],

       [[ 0.6085  ,  0.754   ],
        [ 0.59915 ,  0.720695],
        [ 0.6085  ,  0.754   ]]]),

"CFYp": array([[[-0.55  , -0.74  ],
        [-0.4635, -0.6045],
        [-0.55  , -0.74  ]],

       [[-0.17  , -0.51  ],
        [-0.376 , -0.504 ],
        [-0.17  , -0.51  ]],

       [[-0.265 , -0.735 ],
        [-0.232 , -0.4405],
        [-0.265 , -0.735 ]]]),

"CFZp": array([[[ 0.195 ,  0.0962],
        [-0.025 ,  0.1066],
        [ 0.195 ,  0.0962]],

       [[-1.55  ,  0.7   ],
        [-0.05  , -0.    ],
        [-1.55  ,  0.7   ]],

       [[-0.1   ,  0.8   ],
        [ 0.05  , -0.    ],
        [-0.1   ,  0.8   ]]]),

"CMZp": array([[[-0.1505 , -0.2    ],
        [-0.12045, -0.162  ],
        [-0.1505 , -0.2    ]],

       [[-0.0135 , -0.11   ],
```

```
        [-0.0669 , -0.0935 ],
        [-0.0135 , -0.11   ]],

       [[-0.0225 , -0.185  ],
        [-0.014  , -0.0815 ],
        [-0.0225 , -0.185  ]]]),

"CFXq": array([[[ 0.69  ,  0.94  ],
        [ 0.72  ,  0.97  ],
        [ 0.69  ,  0.94  ]],

       [[-0.615 , -0.305 ],
        [-0.65  , -0.7   ],
        [-0.615 , -0.305 ]],

       [[-1.845 , -1.87  ],
        [-1.8955, -1.687 ],
        [-1.845 , -1.87  ]]]),

"CMXq": array([[[-0.222  , -0.236  ],
        [ 0.0022 ,  0.00195],
        [-0.222  , -0.236  ]],

       [[-0.229  , -0.2825 ],
        [-0.001  ,  0.00165],
        [-0.229  , -0.2825 ]],

       [[-0.245  , -0.2205 ],
        [ 0.0059 , -0.0212 ],
        [-0.245  , -0.2205 ]]]),

"CFYq": array([[[ 0.695 ,  0.635 ],
        [-0.0685, -0.076 ],
        [ 0.695 ,  0.635 ]],

       [[ 0.785 ,  0.995 ],
        [-0.0495, -0.086 ],
        [ 0.785 ,  0.995 ]],

       [[ 0.97  ,  0.66  ],
        [-0.0375, -0.058 ],
        [ 0.97  ,  0.66  ]]]),

"CMYq": array([[[-38.91 , -56.255],
        [-39.855, -56.3  ],
        [-38.91 , -56.255]],

       [[-43.08 , -47.915],
        [-41.45 , -58.38 ],
        [-43.08 , -47.915]],

       [[-38.65 , -50.85 ],
        [-43.05 , -60.05 ],
        [-38.65 , -50.85 ]]]),

"CFZq": array([[[ 11.824 ,  17.22  ],
```

```
        [ 12.2925,  17.105 ],
        [ 11.824 ,  17.22  ]],

       [[ 12.25  ,  13.9   ],
        [ 11.8   ,  16.5   ],
        [ 12.25  ,  13.9   ]],

       [[ 10.3   ,  13.75  ],
        [ 11.5   ,  16.6   ],
        [ 10.3   ,  13.75  ]]]),



"CMZq": array([[[-0.0485 , -0.08   ],
        [-0.03135, -0.0355 ],
        [-0.0485 , -0.08   ]],

       [[-0.009  ,  0.075  ],
        [-0.022  , -0.0435 ],
        [-0.009  ,  0.075  ]],

       [[ 0.0495 , -0.1    ],
        [-0.0235 , -0.0355 ],
        [ 0.0495 , -0.1    ]]]),


"CFXr": array([[[ 0.12  ,  0.885 ],
        [ 0.06  ,  0.105 ],
        [ 0.12  ,  0.885 ]],

       [[ 0.15  ,  0.115 ],
        [-0.015 ,  0.03  ],
        [ 0.15  ,  0.115 ]],

       [[ 0.095 ,  0.1305],
        [ 0.1   , -0.0085],
        [ 0.095 ,  0.1305]]]),


"CMXr": array([[[-0.345   , -0.1425  ],
        [-0.334815, -0.41885 ],
        [-0.345   , -0.1425  ]],

       [[-0.4015  , -0.508   ],
        [-0.395315, -0.537855],
        [-0.4015  , -0.508   ]],

       [[-0.4725  , -0.535   ],
        [-0.461555, -0.52665 ],
        [-0.4725  , -0.535   ]]]),

"CFYr": array([[[ 2.445  ,  1.08   ],
        [ 2.385  ,  3.20555],
        [ 2.445  ,  1.08   ]],

       [[ 2.45   ,  3.33   ],
```

```
        [ 2.337   ,  3.14965],
        [ 2.45    ,  3.33   ]],

       [[ 2.49    ,  3.27   ],
        [ 2.3533  ,  3.1361 ],
        [ 2.49    ,  3.27   ]]]),

"CMYr": array([[[-0.825,  0.755],
        [ 0.05 ,  0.165],
        [-0.825,  0.755]],

       [[-2.55 ,  3.95 ],
        [-0.   , -0.05 ],
        [-2.55 ,  3.95 ]],

       [[-3.65 ,  9.85 ],
        [ 0.05 , -0.   ],
        [-3.65 ,  9.85 ]]]),

"CFZr": array([[[ 0.15   , -0.43483],
        [ 0.015  , -0.02073],
        [ 0.15   , -0.43483]],

       [[ 0.7    , -1.3    ],
        [-0.05   , -0.     ],
        [ 0.7    , -1.3    ]],

       [[ 0.9    , -2.65   ],
        [ 0.05   , -0.05   ],
        [ 0.9    , -2.65   ]]]),


"CMZr": array([[[ 0.3845 , -0.2    ],
        [ 0.36605,  0.54995],
        [ 0.3845 , -0.2    ]],

       [[ 0.393  ,  0.57   ],
        [ 0.34355,  0.5192 ],
        [ 0.393  ,  0.57   ]],

       [[ 0.405  ,  0.6    ],
        [ 0.3582 ,  0.5415 ],
        [ 0.405  ,  0.6    ]]]),
```

Bibliography

[1] S. A. Andrews, R. E. Perez, and D. Wowk. Wing weight model for conceptual design of nonplanar configurations. *Aerospace Science and Technology*, Complete(43):51–62, 2015.

[2] ANSYS. ANSYS Fluent User's Guide, November 2013.

[3] T. Berg van der. Parametric modeling and aerodynamic analysis of multi-element wing configurations. Master's thesis, Delft University of Technology, 2009.

[4] ParaPy B.V. ParaPy website, September 2019. URL `https://www.parapy.nl`.

[5] L. Cappelli, G. Costa, V. Cipolla, A. Frediani, F. Oliviero, and E. Rizzo. Aerodynamic optimization of a large PrandtlPlane configuration. *Aerotecnica Missili & Spazio*, 95(3):163–175, July 2016.

[6] A. Cartieri, D. Hue, Q. Chanzy, and O. Atinault. Experimental Investigations on the Common Research Model at ONERA-S1ma - Comparison with DPW Numerical Results. In *55th AIAA Aerospace Sciences Meeting*, Grapevine, Texas, January 2017.

[7] D. A. Caughey. Introduction to Aircraft Stability and Control Course Notes, 2011.

[8] D. Dal Canto, A. Frediani, G. Ghiringhelli, and M. Terraneo. The Lifting System of a PrandtlPlane, Part 1: Design and Analysis of a Light Alloy Structural Solution. pages 211–234. January 2012.

[9] L. Demasi, G. Monegato, E. Rizzo, R. Cavallaro, and A. Dipace. Minimum Induced Drag Theorems for Joined Wings, Closed Systems, and Generic Biwings: Results. January 2015.

[10] M. Drela and H. Youngren. AVL 3.36 User Primer, February 2017.

[11] EASA, EEA, and EUROCONTROL. European Aviation Environmental Report. Technical report, 2019.

[12] A. Frediani, V. Cipolla, and E. Rizzo. The PrandtlPlane Configuration: Overview on Possible Applications to Civil Aviation. In *Variational Analysis and Aerospace Engineering: Mathematical Challenges for Aerospace Design*, pages 179–210. Springer US, 2012.

[13] IATA. 2036 Forecast Reveals Air Passengers Will Nearly Double to 7.8 Billion, October 2017. URL `https://www.iata.org/pressroom/pr/pages/2017-10-24-01.aspx`. [Accessed August 2019].

[14] IATA. Another Strong Year for Air Travel Demand in 2016, 2017. URL `https://www.iata.org/pressroom/pr/pages/2017-02-02-01.aspx`. [Accessed February 2019].

[15] M. J. Kalinowski. Structural optimization of box wing aircraft. *Archive of Mechanical Engineering*, 12(1), 2015.

[16] B. Konig and E. Fares. Validation of a Transonic Lattice-Boltzmann Method on the NASA Common Research Model. In *54th AIAA Aerospace Sciences Meeting*, San Diego, California, USA, January 2016. American Institute of Aeronautics and Astronautics.

[17] J.H. Koning. Development of a KBE application to support aerodynamic design and analysis, towards a next-generation multi-model generator. Master's thesis, Delft University of Technology, 2010.

[18] I. Kroo. DRAG DUE TO LIFT: Concepts for Prediction and Reduction. *Annual Review of Fluid Mechanics*, 33(1):587–617, 2001.

[19] G. La Rocca. *Knowledge based engineering techniques to support aircraft design and optimization.* PhD thesis, Delft University of Technology, April 2011.

[20] D. J. McCormick. An Analysis of Using CFD in Conceptual Aircraft Design. Master's thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2002.

[21] J.A. Mulder, W.H. J. J. van Staveren, J.C. van der Vaart, E. de Weerdt, C.C. de Visser, A.C. in 't Veld, and E. Mooij. Lecture Notes AE3202 Flight Dynamics, March 2013.

[22] M. Munk. General biplane theory. Technical Report 151, National Advisory Committee for Aeronautics, 1924.

[23] NASA. NASA Common Research Model, 2018. URL `https://commonresearchmodel.larc.nasa.gov/`. [Accessed June 2019].

[24] J. Nathman and M. Matarrese. Hybrid Grid (Structured and Unstructured) Calculations with a Potential-Based Panel Method. In *22nd Applied Aerodynamics Conference and Exhibit*, Providence, Rhode Island, August 2004.

[25] J. K. Nathman. VSAERO User's Manual, September 2016.

[26] F. Oliviero, D. Zanetti, and V. Cipolla. Flight dynamics model for preliminary design of PrandtlPlane wing configuration with sizing of the control surfaces. *Aerotecnica Missili & Spazio*, 95:201–210, December 2016.

[27] L. Prandtl. Induced drag of multiplanes. Technical report 182, National Advisory Comittee for Aeronautics, March 1924.

[28] A. Raju Kulkarni, G. La Rocca, and L. L. M. Veldhuis. Degree of similitude estimation for sub-scale flight testing. In *AIAA Scitech 2019 Forum*, San Diego, California, January 2019.

[29] A. Raju Kulkarni, C. Varriale, M. Voskuijl, G. La Rocca, and L. L. M. Veldhuis. Assessment of Sub-scale Designs for Scaled Flight Testing. In *AIAA Aviation 2019 Forum*, Dallas, Texas, June 2019.

[30] D. Raymer. *Aircraft Design: A Conceptual Approach, Fifth Edition*. American Institute of Aeronautics and Astronautics, Inc., Washington, DC, August 2012.

[31] M. Rivers and C. Hunter. Support System Effects on the NASA Common Research Model. In *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Nashville, Tennessee, January 2012.

[32] M. Rivers, C. Hunter, and R. Campbell. Further Investigation of the Support System Effects and Wing Twist on the NASA Common Research Model. In *30th AIAA Applied Aerodynamics Conference*, New Orleans, Louisiana, June 2012.

[33] J. Roskam. *Airplane Design Part 1: Preliminary Sizing of Airplanes*. Roskam Aviation and Engineering Corporation, Ottawa, Kansas, 1985.

[34] E. Torenbeek. Development and application of a comprehensive, design-sensitive weight prediction method for wing structures of transport category aircraft. *Delft University of Technology, Faculty of Aerospace Engineering*, 1992.

[35] D. A. J. Van Ginneken, M. Voskuijl, M. J. L. Van Tooren, and A. Frediani. Automated Control Surface Design and Sizing for the Prandtl Plane. In *51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Orlando, USA, April 2010.

[36] M. van Hoek. Structural Design, Analysis and Optimization of a Lifting Surface in a Knowledge Based Engineering Environment. Master's thesis, Delft University of Technology, 2009.

[37] C Varriale, A Raju Kulkarni, G La Rocca, and M Voskuijl. A hybrid, configuration-agnostic approach to aircraft control surface sizing. In *Italian Association of Aeronautics and Astronautics*, page 20, Rome, Italy, September 2019.

[38] M Voskuijl, J. de Klerk, and D. van Ginneken. Flight Mechanics Modeling of the PrandtlPlane for Conceptual and Preliminary Design. January 2012.

[39] J. H. Wei. Parametric modelling for determining aircraft stability and control derivatives. Master's thesis, Delft University of Technology, 2016.