CHALMERS
UNIVERSITY OF TECHNOLOGY

Streamlining the Aircraft Conceptual Design Process

From Concept to Flight: Streamlining the Aircraft Conceptual Design Process through Advanced Mass and Balance Techniques

Master's thesis in Mobility Engineering

Michael Crona
Simon Dinger

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Streamlining the Aircraft Conceptual Design Process

From Concept to Flight: Streamlining the Aircraft Conceptual Design Process through Advanced Mass and Balance Techniques

Michael Crona
Simon Dinger

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Some Subject or Technology
Division of Mechanics and Maritime sciences
Chalmers University of Technology
Gothenburg, Sweden 2024

Streamlining the Aircraft Conceptual Design Process
From Concept to Flight: Streamlining the Aircraft Conceptual Design Process through Advanced Mass and Balance Techniques
Michael Crona
Simon Dinger

Supervisor: Anderson Lindegger, Heart Aerospace
Examiner: Carlos Xisto, Department of Mechanics and Maritime sciences

Master's Thesis 2024
Department of Mechanics and Maritime sciences
Division of Fluid Mechanics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Streamlining the Aircraft Conceptual Design Process From Concept to Flight: Streamlining the Aircraft Conceptual Design Process through Advanced Mass and Balance Techniques.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Streamlining the Aircraft Design Process
From Concept to Flight: Streamlining the Aircraft Design Process through Advanced Mass and Balance Techniques
Michael Crona
Simon Dinger
Department of Mechanics and Maritime sciences
Chalmers University of Technology

Abstract

This master thesis investigates mass and balance techniques in the conceptual design phase of aircraft, with a focus on the ES-30 electric aircraft developed by Heart Aerospace. The thesis centers on developing and validating a computational tool for precise calculation of mass properties and center of gravity (CG) envelopes. This tool facilitates adjustments in aircraft design and has been applied to a case study on a conceptual liquid hydrogen (LH2) aircraft. The process aims to reduce the iterations needed in design modifications, speeding up the pre-flight testing phase and ensuring adherence to safety and performance standards. The findings demonstrate the tool's applicability in future aircraft design projects, particularly for integrating new propulsion technologies such as electric and LH2 systems.

Keywords: Aircraft Conceptual Design, Center of Gravity, Mass Properties, Computational Tools, Electric Aircraft, Liquid Hydrogen Propulsion

Acknowledgements

We would like to thank Carlos Xisto for his review and constructive feedback on our report, and acknowledge his contribution to our aerospace engineering education.
Thanks to Anderson Lindegger at Heart Aerospace for his advice on mass properties and balance for aircraft, which enhanced our understanding of the subject as we worked on our thesis.
We appreciate the flight science team at Heart Aerospace for their openness to our inquiries and for including us in their daily operations, enriching our research experience.
Heart Aerospace is acknowledged for allowing us to conduct our research and learn within their company.
Chalmers University is recognized for providing the knowledge and skills necessary to complete our thesis.
We thank Per Samuelsson and Hjalmar Strömfeldt for their detailed review and feedback on our report. A special note of thanks to Per Samuelsson for his role in developing the case study aircraft that formed the basis of our research.
Finally, we acknowledge our teachers, teaching assistants, and classmates at Chalmers for their part in our educational journey.

Michael Crona, Simon Dinger, Gothenburg, June 20, 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AOA	Angle of Attack
API	Application Programming Interface
ATA	Air Transport Association of America
A4A	Airlines for America
CAD	Computer Aided Design
CFD	Computational fluid dynamics
CG	Center of Gravity
CG-envelope	Center of Gravity envelope
COM	Component Object Model
CSA	Case Study Aircraft
CT	Collector Tank
EASA	European Union Aviation Safety Agency
EIS	Entry Into Service
GA	General Aviation
IDE	Integrated development environment
JASC	Joint Aircraft System/Component
KBE	Knowledge Based Engineering
LH_2	Liquid Hydrogen
MAC	Mean Aerodynamic Chord
MLG	Main Landing Gear
MLW	Maximum Landing Weight
MOI	Moment of Inertia
MOW	Minimum Operating Weight
MRW	Maximum Ramp Weight
MTOW	Maximum Take off Weight
MZFW	Maximum Zero Fuel Weight
NLG	Nose Landing Gear
NM	Nautical Miles
OEW	Operating Empty Weight
PAX	People
POI	Product of Inertia
SAF	Sustainable Aviation Fuel
UI	User Interface
VB	Visual Basic
ZFW	Zero Fuel Weight

Nomenclature

Below is the nomenclature of parameters that have been used throughout this thesis.

Parameters

g	Gravitational acceleration $[m \cdot s^{-2}]$
I	Inertia $[kg \cdot m^2]$
m	Mass $[kg]$
ρ	Density $[kg/m^3]$
x	x-location$[-]$
y	y-location$[-]$
z	z-location$[-]$

Contents

List of Figures

List of Tables

1

Introduction

1.1 Background

In 1903, the Wright brothers pioneered the first powered heavier-than-air aircraft, marking the beginning of its development. Initially, efforts focused on increasing speed and reducing operational costs. As a result, passenger capacity expanded, leading to a decrease in seat-mile costs. From 1928 to 1958, aircraft speed increased fivefold, reflecting significant advancements in aviation technology [1]. However, the aviation sector is facing significant changes due to the expected sharp increase in both regional and international flights in the near future. This growth in air travel is likely to accelerate the sector's contribution to global warming which today is around 3% [2], highlighting the need for significant advancements in aircraft efficiency. The engines that currently power the majority of aircraft, both turboprops and turbojets, have been the subject of extensive development over many years. However, they have now reached a point where achieving further efficiency improvements is becoming increasingly difficult. As a result, there is a clear necessity for more profound changes in the way that aircraft are propelled.

Given the traditionally slow pace of innovation within the aviation industry, the move towards adopting more sustainable propulsion technologies marks a particularly noteworthy period of progression. The shift from reliance on conventional jet fuel to the exploration of greener alternatives represents a considerable challenge. Heart Aerospace is leading the change in the transition towards electrified regional aviation with the development of their electric aircraft, the ES-30. The process of developing an aircraft is inherently complex, necessitating the optimization of mass properties through numerous iterations across various design stages. This initial conceptual phase alone can span approximately 1.5 years and may require revisiting if significant modifications are deemed necessary, highlighting the importance of streamlined and efficient development processes. In addition to this, within the aviation industry, safety is of upmost importance and because of that rigorous aircraft testing serves as the vital link between design and reality, guaranteeing that the aircraft meet safety standards and perform optimally under various conditions [3]. Hence, the development phase needs to not only be efficient but also accurate such that the flight testing can confirm the results gained from the development phase and prevent the cycle of redevelopment.

Furthermore, the selection of suitable flight maneuvers, test conditions, and acceptance criteria for test points is important for the success of flight testing. These elements are essential for gathering the data necessary to affirm the intended objectives [4].

The aircraft's center of gravity (CG) envelope is important not only for operational stability, allowing pilots to achieve precise trim and define the aircraft's limits, but also in identifying the correct test points. It is crucial for confirming or uncovering the conditions under which the aircraft remains operational.

The focus of the master thesis is to dive deeper into the aircraft development process at Heart Aerospace, specifically developing the methods used related to the weight and balance of the ES-30, presenting the CG envelope under all its operational flight conditions. It will then assess the significance of various tests to understand how changes in the fuel tank's weight and balance, influenced by different fuel volumes and pitch angles, impact the aircraft performance. In addition to this, the thesis will include a case study on how the developed methods can be used on other aircraft and in addition aid in the research and development of future liquid hydrogen (LH_2) aircraft. The Case study will be conducted on an aircraft designed during the aircraft design course (MMS236) at Chalmers University of Technology. The rationale of the case study is twofold. Mainly, it aims to test the methods' applicability across various projects, especially since the goal of the tools is to streamline the obtainment of mass properties. Therefore, it's essential that the time required to implement these methods and tools is reasonable. Secondly, analyzing the results regarding the CG envelope and other mass properties can enhance the understanding of current research topics such as hydrogen tank positioning and fuel system management.

Additionally, it involves the development of a tool to automatically update the CG calculations whenever there is a change or addition to the design. Making the process faster will enable more dynamic and responsive adjustments to the aircraft's mass properties, mainly in the conceptual design. It also allows for a smooth and reliable transition to the development phase where big design changes bring a high cost to the program (e.g. reposition the wing to mitigate aircraft CG mistake).

1.2 The ES-30 Aircraft

This thesis focuses on the ES-30, a hybrid-electric aircraft designed for regional commercial flights. The ES-30, which aims to meet the European Union Aviation Safety Agency's (EASA) Certification Specification Part 25 (CS-25) for large aircraft, offers seating for 30 passengers in a 2+1 configuration and supports 25 kg of luggage per passenger.

The aircraft's architecture features conventional design elements, including two main wings, a single vertical tail, and a horizontal stabilizer positioned on top of the vertical tail. This design ensures familiar aerodynamic performance and operational characteristics while integrating advanced hybrid-electric propulsion technology. The propulsion system is primarily powered by rechargeable batteries housed under the fuselage, which in turn energize four electric motors. In figure 1.1 a conceptual design of the ES-30 can be seen.

Figure 1.1: *Conceptual design rendering of the ES-30 provided by Heart Aerospace [5]*

To enhance its operational flexibility and safety, the ES-30 is equipped with a secondary power system consisting of two turbo-generators located in the aft of the fuselage. These generators are fueled by sustainable aviation fuel (SAF) stored in tanks within the wings, allowing the aircraft to switch from a purely electric mode, covering 200 km, to a hybrid mode that extends its range to 400 km and up to 800km with 25 passengers [5].

Scheduled to commence commercial service in 2028, the ES-30 represents a step towards more sustainable aviation practices. By integrating electric propulsion within a traditional aircraft design, it offers a promising low-emission alternative for regional air travel.

1.3 The Case Study Aircraft (LH_2 90 Pax)

The case study aircraft (CSA) is a liquid hydrogen (LH_2) aircraft with entry-into-service (EIS) 2035 and was designed during a two month course in aircraft design, MMS236 at Chalmers. It's a conceptual design for 90 PAX, 3+2 passenger configuration, range of $1100NM$, cruise altitude of $25000ft$ and turboprop with two separate LH_2 fuel tanks. It is designed by following the conceptual design steps by Raymer [6] using the Dash 8 Q400 as a reference aircraft. The aircraft has two tanks on in the front and one in the aft where the forward tank contains $9m^3$ and the aft tank contains $17m^3$ of fuel. In figure 1.2 a render of the design is shown.

Figure 1.2: *Conceptual design rendering of the LH_2 90 PAX case study aircraft*

1.4 Objective

The main objectives of this master thesis are centered around advancing the development process for electric aircraft, specifically focusing on Heart Aerospace's ES-30 weight and balance. Through detailed analysis and innovative tool development, the thesis aims to achieve the following key goals:

- Refine weight and balance methodologies for the ES-30 electric aircraft.
- Develop a weight X CG-envelope for all operational flight conditions.
- Evaluate the impact of fuel volume (consumption and refueling fuel curve) and pitch angles on aircraft performance and CG-envelope.
- Suggest testing points for which the extreme points of the envelope are present.
- Create an automated tool for dynamic CG calculations, taking into account all phases of the mission (from taxing to flight and landing).
- Contribute to optimizing electric aircraft performance for sustainable aviation.
- Perform a case study and research on how the developed tools can be applied to liquid hydrogen propelled aircraft.

1.5 Limitations

For this thesis the focus is placed on advancing the aircraft development process through innovative weight, balance, and testing methodologies, specifically targeting the ES-30 electric aircraft by Heart Aerospace. To ensure a focused and manageable scope, the study establishes clear demarcations as follows:

Scope of Technologies: The work will concentrate on the technologies directly related to mass and balance calculations, automated tool development for these calculations, and the determination of flight test points.

Type of Aircraft: The research is centered on the ES-30 electric aircraft and the case study aircraft (CSA). While the findings may have broader implications for an aircraft with an integral fuel tank, other types of fuel tank configurations than the ones optional for the ES-30 or the CSA are beyond this thesis's scope.

Flight Testing Phase: Although the thesis aims to identify crucial testing points for flight tests, it will not delve into the execution of these flight tests, including data collection and analysis. The focus remains on the pre-flight phase, specifically on weight and balance boundaries.

Environmental Impact Analysis: While the motivation for electric aviation includes reducing environmental impact, this thesis will not conduct a detailed environmental impact analysis of electric versus conventional aircraft. The study's emphasis is on the technical aspects of aircraft design and testing.

Economic Considerations: Detailed economic analysis, including cost-benefit analysis of implementing the proposed methodologies and technologies, falls outside this thesis's scope. The primary focus is on the technical feasibility and potential performance improvements.

By setting these demarcations, the thesis aims to maintain a clear focus on its primary objectives, ensuring a thorough and in-depth exploration of the specified areas within the constraints of time and resources available for master-level research.

1.6 Thesis Outline

Chapter 1 - Introduction: Sets the stage for the thesis, discussing the motivation behind focusing on electric aviation, specifically the ES-30 electric aircraft by Heart Aerospace. It highlights the advancements in electric aircraft development and outlines the objectives and scope of the thesis. This chapter emphasizes the significance of electric propulsion systems in reducing environmental impact and the technological challenges they pose.

Chapter 2 - Literature Study: This chapter reviews existing research focused on the acquisition and management of mass and balance properties in aviation. It evaluates studies that have addressed the challenges and methodologies in accurately determining these properties through current models, essential for efficient aircraft design. The review also explores the differences between current aircraft technology and future technology such as hydrogen propulsion systems. In addition to this, it touches upon the idea of automation to save valuable methods not only as employee knowledge but also through tools.

Chapter 3 - Theory: The theory chapter explores basic flight dynamics—pitch, roll, and yaw—and their importance in aircraft design. It also covers the impact of fuel storage and center of gravity management on aircraft performance. This chapter details the integration of CAD and VB.Net for enhanced mass property data collection, essential for precise design adjustments.

Chapter 4 - Methods: Methods elaborated on for the development of the mass properties tool and its integration with CAD systems. It details the methodologies used for CG envelope calculations and discusses the implementation of VB.NET for automating dynamic CG calculations. The chapter also outlines the testing protocols followed to validate the tool's effectiveness. The methods chapter also presents the Case Study Aircraft design and development procedure.

Chapter 5 - Results and Discussion: Presents the data derived from the application of the developed tools, analyzing the performance and accuracy of the CG envelope predictions under different load and flight conditions. This chapter discusses the significance of the results in the context of aircraft design optimization, including a detailed review of the case study on the LH2 aircraft. It evaluates how the tool impacts design decisions and identifies any limitations encountered during the analysis.

Chapter 6 - Conclusions: Summarizes the key findings and the impact of the developed mass properties tool on the conceptual design process of electric and LH2 aircraft. It highlights how the thesis objectives were met and discusses the implications of these findings for future aircraft design projects.

Chapter 7 - Future Work: Outlines potential areas for further research, including the refinement of the computational tool, expansion of the case study models, and exploration of additional aircraft configurations.

2

Litterature Study

2.1 Case Study, LH_2 Propelled Aircraft

Hydrogen has emerged as a highly likely energy carrier for the future of aviation. The necessity for innovation in propulsion systems has been historically met by advances in fuel technology, and hydrogen is revisited as a promising candidate amidst rising fuel costs and environmental considerations. Its potential as a zero-emission fuel was further investigated after the oil crisis in the 1970, and with the current environmental and energy sustainability challenges, research about hydrogen as a long-term solution for aviation fuel is ongoing [7].

One of the mayor advantages with liquid hydrogen compared to kerosene is that it doesn't contain any carbon which means that when burning the fuel, there is no CO_2 emissions, this is can be seen in figure 2.1

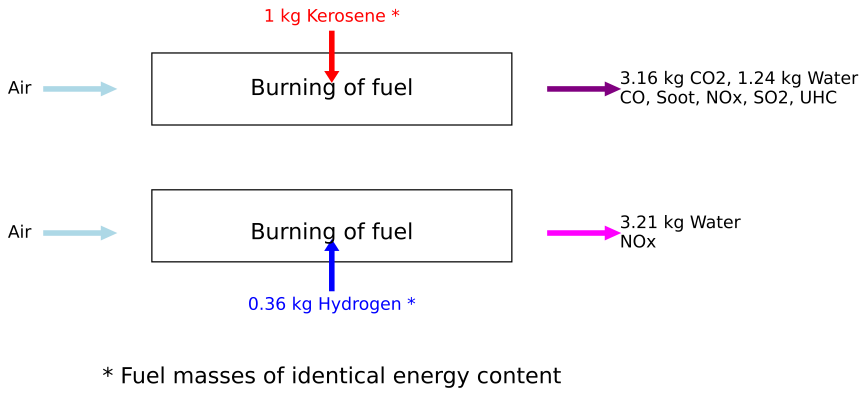


Figure 2.1: *Emissions produced and mass of fuel comparison between liquid hydrogen and kerosene*

When it comes to developing commercial LH_2 aircraft, the design and integration of LH_2 fuel tanks are major factors driving the design process [8]. Because of this, significant research has focused on fuel tank design. However, for successful development and conceptual design optimization, accurate CG location and tank placement are essential. This is where the developed tool can make a significant difference. By getting more data from the CAD model, concepts that is unfeasible can efficiently be discarded or changed.

2.1.1 Size and Weight Difference Between Conventional Kerosene Fuels and LH_2

Comparing the use of hydrogen instead of conventional kerosene fuels, hydrogen offers 2.8 times the energy per unit weight than kerosene, seen in figure 2.2. However, this advantage is partly reduced by the additional weight due to the hydrogen fuel system's complexity and larger tank size. Despite this, hydrogen is likely to facilitate an increase in payload capacity at a constant takeoff weight. Conversely, storing an equivalent energy amount in liquid hydrogen form necessitates a volume that is quadruple that of kerosene, also shown in figure 2.2. Furthermore, to satisfy insulation to be able to keep the *LH2* at approximately $-253C^{\circ}$ and minimal differential pressure requirements, the tanks must be designed in spherical or cylindrical geometries, leading to unconventional tank configurations[9].

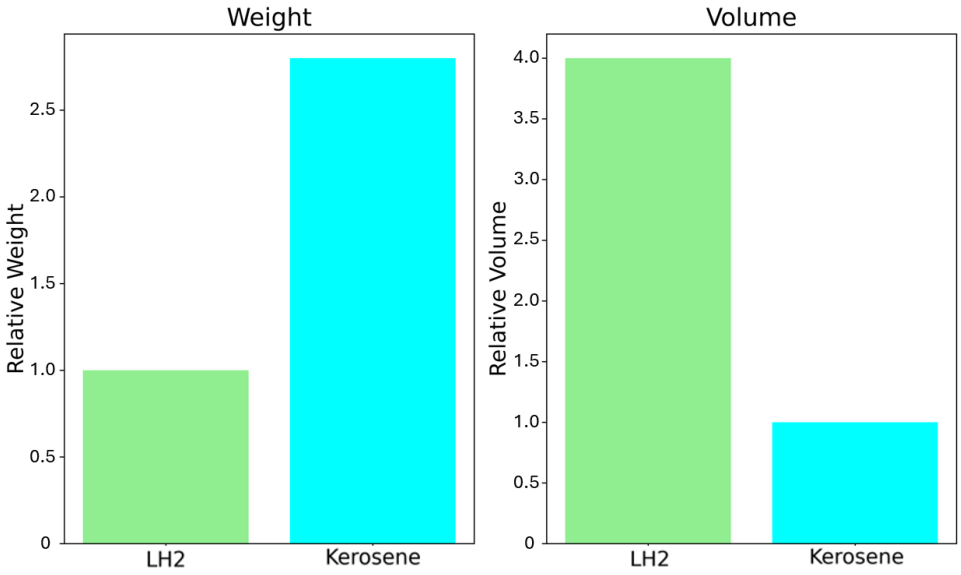


Figure 2.2: *Volume and weight differences between LH_2 and Kerosene fuels for the same amount of energy*

2.1.2 LH_2 Fuel Tank Weight and Balance Challenges

Due to the larger volume and unusual shape of liquid hydrogen tanks, modifications in aircraft design are required as opposed to the traditional setup where fuel is stored within the wings. Consequently, this has led to the development of aircraft configurations that situate the hydrogen fuel tanks within the fuselage, either towards the rear or front.

Incorporating the fuel tank into the fuselage inherently shifts the CG either forward or aft, based on the tank's placement, given that placing the tank between passenger rows is not a feasible alternative.

G. Onorato et al.[10] demonstrate that placing a trim tank at the front of the aircraft can mitigate aft CG shifts by maintaining sufficient fuel to keep the CG independent of fuel levels [10]. This approach results in lower specific energy consumption, but negatively impacts the operational empty weight (OEW) and maximum takeoff weight (MTOW).

The study by Dannet [11] explored two distinct hydrogen fuel tank configurations—one with a single tank located aft inside the fuselage, and another with two tanks positioned at both the front and the rear inside the fuselage. Their findings indicated

that the aircraft's CG remained nearly unchanged in the dual-tank configuration, while a single rear tank resulted in a CG shift of nearly half a meter [11].

Furthermore, their research also found that for the aircraft with only one tank, stable flight was dependent upon the payload distribution. The aircraft was incapable of flying with either maximum or no payload, but it required precise payload adjustment to achieve a balanced CG, underscoring the critical nature of fuel tank placement and CG management in flight operations.

2.2 Enhanced Catia Functionality for Aircraft Fuel Volume Calculation

Creating an accurate database for fuel volume characteristics is essential in aircraft design and operation. The research conducted by Gongli Tan and Chunling Zhu explores the use of Catia's analysis capabilities to quickly and accurately generate such a database for different attitude angles and fuel levels [12]. This study identifies the challenges in calculating fuel volume due to the irregular shape of aircraft fuel tanks. Previous methods, such as the slicing analysis technique derived from FEM, are noted for their complexity and potential inaccuracies due to oversimplification. In contrast, the application of Catia for analysis offers a direct method for calculating volume characteristics, which significantly improves both speed and accuracy [12].

Tan and Zhu utilize Catia's tools to simulate various fuel levels and aircraft attitudes, updating the volume characteristics accordingly. This is done by using a solid that represents the fuel volume and through automatic updating of sectioning planes simulate both different attitude and fuel volumes. This approach benefits from Catia's ability to handle complex shapes and automate calculations, representing a significant improvement over traditional finite element methods. Their method assumes a fixed position for the aircraft's body, simplifying the calculation by rotating the section-planes that simulates fuel surface through translation to match different attitudes. Since this means that Catia will base the mass properties on a leveled tank, these mathematical translations is applied for the data generated too. This simplification of using mathematical translation rather than model translation streamlines the entire process, making it more efficient and less prone to errors [12].

The paper highlights the practicality of using Catia in the aviation industry, offering a method that can be generalized to other aircraft types. The work of Tan and Zhu builds on the foundation set by earlier studies, pushing the boundaries of how 3D CAD software can be leveraged for detailed and accurate fuel volume analysis in aircraft design [13].

Their findings demonstrate the potential of using advanced software tools for improving the accuracy of measurements in aircraft design not only for static- but also for dynamic objects. By automating the generation of fuel volume characteristic databases, this research offers insight in how to generate mass data for the fuel such as CG and weight quickly which is a very valuable asset in the conceptual phase of aircraft development.

2.3 Application of Knowledge Based Engineering in the Development of Automation Tools

In the study "*Automotive Industry CAD System Automation through knowledge based engineering (KBE) Case Study*" carried out by Bahr Rogerio and Tiago Weller, they analyse the use of KBE in the automotive industry. The essence of KBE lies in its capacity to codify design knowledge, enabling the development of automation tools that are not only application-specific but also reusable across different revisions and projects. This re-usability is particularly critical in aircraft design, where similar design challenges recur across many revisions in the aircraft design process. By establishing a library of functions or tools to address these challenges, designers can significantly reduce the time spent when designing and evaluating project changes. This approach not only saves time but also ensures consistency. While the industries for which the study has been carried out differ the study is still applicable in this thesis. In product design, a field often subject to rapid changes, the integration of advanced computational techniques plays a crucial role in streamlining the development process. Among these, KBE stands out as a valuable strategy. This literature review delves into the relationship between KBE and the creation of dedicated automation tools, highlighting its capacity to enhance and streamline the development process [14].

The authors point out that, despite KBE's potential, its adoption varies significantly across their studied field, the automotive industry. Through case studies of two major automotive companies, the paper illustrates the current state of KBE implementation. It notes that while CAD software, including Catia V5, offers advanced capabilities for automation and KBE integration, not all companies fully exploit these features to enhance efficiency and innovation in product development.

3

Theory

This chapter outlines foundational theories essential for understanding the content in subsequent chapters. It starts by covering basic flight dynamics and then explains pitch, roll, and yaw, which determine an aircraft's orientation and stability. The discussion then shifts to fuel storage and management, focusing on how these systems impact the aircraft's center of gravity and balance. This section is tied into the CG-envelope, a key concept in aircraft design that sets operational and safety limits. The chapter also examines the aircraft's inertial properties, including the moment of inertia and the product of inertia. Additionally, the chapter introduces Visual Basic and CATIA, tools vital for aircraft design simulations and modeling. It also discusses the translation and shifting of coordinate systems, which will help in the understanding of the different tool methodologies.

3.1 Basic Flight Dynamics, Pitch, Roll and Yaw

Flight dynamics is a field of aerospace engineering that studies the performance, stability, and control of vehicles in flight. It involves the analysis of the forces and moments acting on a vehicle and how they affect its motion in air. The fundamental aspects of flight dynamics include the study of pitch, roll, and yaw movements, which are the primary axes of rotation for any airborne vehicle. These movements are illustrated in figure 3.1.

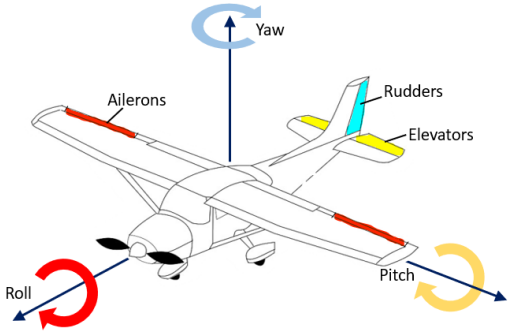


Figure 3.1: *Pitch, Roll and Yaw movements. Original aircraft image by [15], modified by the authors for illustrative purposes.*

3.1.1 Pitch

Pitch refers to the rotation of an aircraft about its lateral axis, which runs from wingtip to wingtip. This motion changes the attitude of the aircraft's nose, effectively altering its angle of attack to the oncoming air. The control of pitch is essential for maintaining

the desired flight path.

In figure 3.1 the pitch motion of an aircraft is illustrated and marked in yellow. The pitch influences the lift generated by the wings and is controlled by the elevators in the aft. An increase in pitch angle can increase lift up to a certain point; beyond this angle, the aircraft may stall due to excessive angle of attack.

3.1.2 Roll

Roll is the rotation of an aircraft about its longitudinal axis, which runs from the nose to the tail of the aircraft. This motion causes one wing to rise while the other descends, effectively banking the aircraft to the left or right.

In figure 3.1 the roll motion is marked with red. The ailerons, located at the trailing edges of the wings, are the primary control surfaces used to control roll, also marked in red. By deflecting one aileron up and the other down, a differential lift is created on the wings, inducing a roll.

3.1.3 Yaw

Yaw refers to the rotation of an aircraft about its vertical axis, which runs from the top to the bottom of the aircraft. This motion alters the horizontal direction of the aircraft's nose, enabling it to turn left or right without banking. Yaw control is crucial for directional stability and is often used in conjunction with roll during turning maneuvers.

In figure 3.1 the yaw motion is marked with blue. The rudder, located on the vertical stabilizer, is the primary control surface for yaw together with the vertical stabilizer [16]. Deflecting the rudder to the left or right creates a lateral force at the tail, causing the aircraft to yaw in the desired direction.

3.2 Fuel Storage and CG Overview

There are mainly three types of fuel tanks. Discrete tanks are containers that are separately manufactured and mounted by bolts. These are normally used for small general aviation (GA). Then there is the bladder tank, which is a thick rubber bag stuffed into a cavity in the structure. This type causes a loss of approximately 10% of the available fuel volume. It is widely used in military aircraft because of its self-sealing capabilities. Lastly, there are integral tanks, which are cavities within the airframe structure that are sealed to form a fuel tank. These are usually found in the wing box and/or the cavity between the two fuselage bulkheads. Integral tanks have a loss of about 5% of the available fuel volume due to the sealant and foam used to prevent leakage. However, with modern techniques, this risk is much less today. [17]

Conventional aircraft carries a large volume of fuel, with respect to the total volume of the aircraft, which needs to be carried aboard the aircraft to meet the required range [18]. A common fuel tank layout for a commercial transportation aircraft is to place the tank in the wing structure between the forward and aft wing spars [19] and additional tanks located between the wings. For longer range aircraft and business jets, additional tanks in the fuselage and tail do also occur. However this is not something that is used in most cases. Because of the positioning in the wings, wing sweep will affect the longitudinal CG during flight as the fuel is consumed. This change in CG can be rather big and change the aircraft static stability and handling characteristics. Optimizing the

longitudinal CG for cruise can minimize the profile drag and increase the operating range for a commercial aircraft [18].

Another aspect of the fuel system is to maintain a full feed tank (from now on called collector tank (CT)) to each engine, this is done by transferring fuel from the auxiliary tanks into the CTs. This means that the distribution of fuel will not be equal to the volume of each segment of the tank since the CT will be kept full, the auxiliary tanks will have less fuel than if the distribution was not interfered with. An example of how the layout could look for the fuel tank can be seen in figure 3.2, where segment 4, 3 and 2 represents the auxiliary tanks and the rightmost segment marked in red is the CT. For the ES-30, the effects of longitudinal CG change during flight might not be as big considering its 30 PAX capacity and that it is a hybrid electric aircraft, reducing the amount of fuel carried in the wings. The effects however, cannot be neglected and still needs to be calculated to be able to get the aircraft certified.

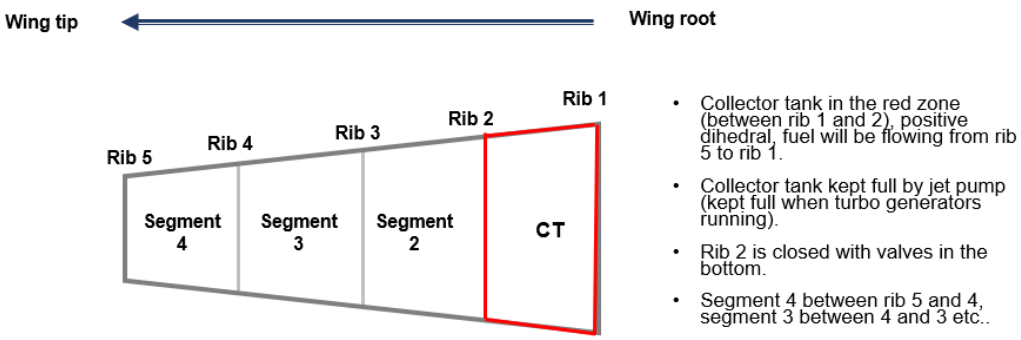


Figure 3.2: *Example of fuel tank layout, seen from above*

3.2.1 ES-30 Fuel Management and Stability During In-Flight Turbo-Generator Start-Up

A distinctive feature of the ES-30, in contrast to traditional fuel-powered aircraft that initiate engine start-up on the ground, its strategy to activate the fuel-powered generators mid-flight. Consequently, this operational choice leads to a dynamic alteration in the fuel consumption curve each time the turbo-generators are engaged and the CT remains filled. It necessitates a thorough analysis of two specific fuel management scenarios: one where the CT tank is not maintained at full capacity, and another during turbo-generator operation. Ensuring that the CG shift, triggered by engine start-up, does not compromise the aircraft's static stability is crucial.

3.2.2 Case Study Fuel Management and Stability

In figure 3.3 the tank configuration for the hydrogen powered case study aircraft can be seen. As mentioned in the introduction, the CSA has a trim tank in the front and the primary tank in the aft. A distinctive feature of the CSA compared to traditional aircraft is that the fuel tanks are placed inside the fuselage because of the need for volume. This makes the weight distribution depend a lot on where the tanks are placed and how much weight is allocated to each tank. The fuel management system needs to mitigate the CG shift during flight as much as possible.

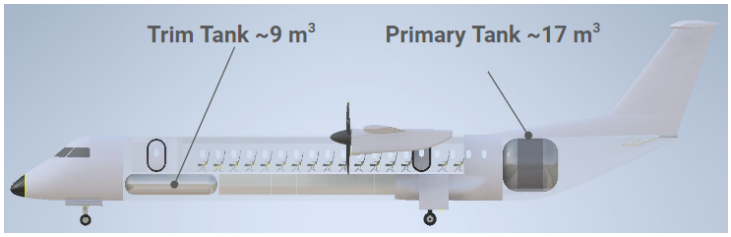


Figure 3.3: *Conceptual design tank configuration*

3.2.3 CG Shift Due to Fuel Storage and Longitudinal CG Effects

Referenced in the preceding section, the impact of fuel storage on the aircraft's longitudinal CG is crucial for aircraft stability, balance, and handling. Moreover, the effects of these factors on trim drag and consequently on the aircraft's operating range can be substantial. Below in figure 3.4 four different cases is shown for static stability and instability. The CG must be located in front of the center of lift. This means that in level flight there is a need for a pitch-up trim moment from the horizontal stabilizer to counteract the pitch down moment due to the forward CG. For this case if a pitch-up disturbance occur, the natural pitch down moment will help to get the aircraft to level flight again (statically stable). If however the CG is behind the aircraft center of lift, there will be a need for a pitch-down moment from the horizontal stabilizer at level flight and during a pitch-up disturbance, the natural pitch-up moment will increase the pitch-up attitude further (statically unstable).

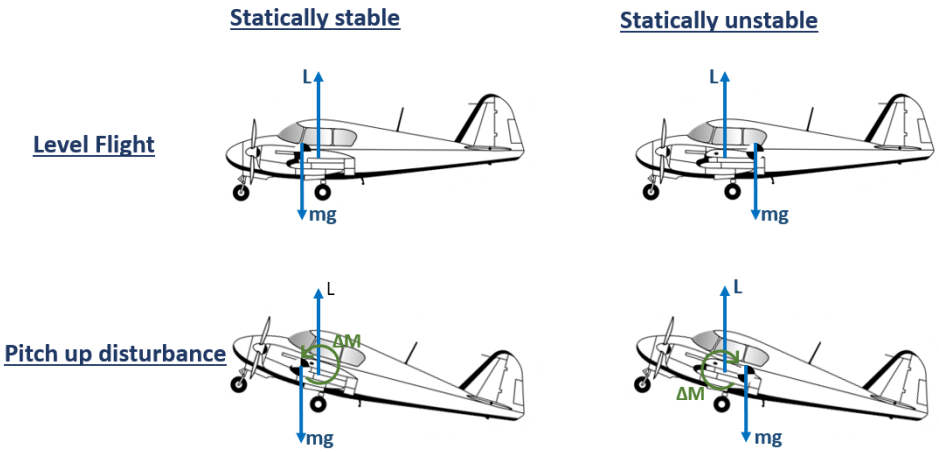


Figure 3.4: *Static stability modes. Original aircraft image by [20], modified by the authors for illustrative purposes.*

Therefore it is important to make sure that the fuel system does not have any potential failure modes that would cause an unstable aircraft CG during flight [21].

Aviation regulations dictate that all aircraft certified as transport category airplanes (14 CFR Part 25) must be both statically and dynamically stable which must be tested for all operational conditions. There is also requirements presented in (14 CFR Part 25.23, Part 25.25 and Part 25.27) referring to the loading, weight and CG limits and under which conditions the aircraft must be compliant. This implies that the CG of the

aircraft must remain in some location in space throughout the entire flight. This location (area) of which the aircraft fulfills this throughout the flight is called the center of gravity envelope (CG-envelope) [22].

To determine the CG-envelope, there is a need to look at the longitudinal, lateral and vertical CG-envelopes. In this thesis, the main focus will be on longitudinal CG-envelope and below some boundaries and effects from the CG location in the longitudinal direction is presented, gathered from [23]. A representation of CG moved to forward and too aft can be seen in figure 3.5.

CG too forward:

- **Insufficient elevator authority**: Unable to pitch up at low speeds during landing (too nose-heavy) and unable to produce enough moment to rotate the nose during take-off, which leads to the need for longer runway to attain sufficient speed.
- **Increased longitudinal stability**
- **Poor performance at any given airspeed:** Increased negative lift on the tail to resist the nose tendency to drop leads to increased angle of attack (AoA) to trim aircraft which leads to increased drag.
- **Excessive loads on the nose landing gear:** Possible damage to the airplane when landing.

CG too Aft

- **Tendency to nose up:** Nose-down elevator might be required to counter the nose-up tendency during flare when landing and at take-off, aircraft likely to nose up prematurely which leads to increased drag.
- **Decreased longitudinal stability**
- **Increased potential for a violent stall**
- **Spin recovery more difficult as the CG moves rearward**
- **Possibility of a tip over if the CG is far aft the neutral point**
- **Nose landing gear load too low, steering load requirements sets a limit**
- **Main landing gear load too high, structural loading sets a limit**

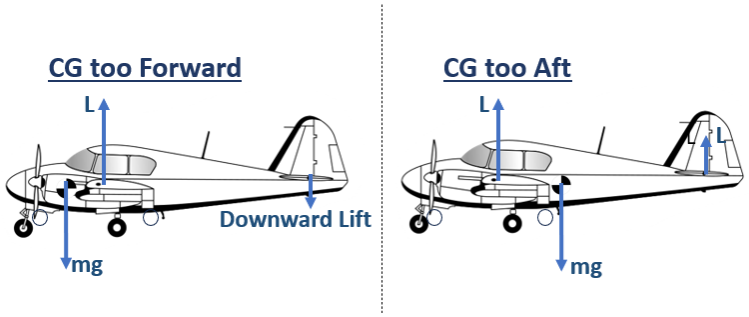


Figure 3.5: *CG too forward (left) and CG too Aft (right). Original aircraft image by [20], modified by the authors for illustrative purposes.*

3.2.4 CG-Envelope

All of the previously mentioned situations impacts the CG-envelope, and as the development of the aircraft moves on, more detail is added into the CG-envelope. In this section a description of the development of the CG-envelope will be presented as a continuation of the previous section.

The first step in establishing the CG-envelope comes from 14 CFR 25.23 which is connected to the load distribution limits, which states that [24]:

14 CFR 25.23

(a) Ranges of weights and centers of gravity within which the airplane may be safely operated must be established. If a weight and center of gravity combination is allowable only within certain load distribution limits (such as span-wise) that could be inadvertently exceeded, these limits and the corresponding weight and center of gravity combinations must be established.

(b) The load distribution limits may not exceed:

- The selected limits;
- The limits at which the structure is proven;
  or:
- The limits at which compliance with each applicable flight requirement of this subpart is shown.

Together with 14 CFR 25.27 which sets the center of gravity limits according to [25]:

14 CFR 25.27

The extreme forward and the extreme aft center of gravity limitations must be established for each practicably separable operating condition. No such limit may lie beyond:

- The extremes selected by the applicant;
- The extremes within which the structure is proven;
  or:
- The extremes within which compliance with each applicable flight requirement is shown.

To determine these limits can be tedious but often the forward limit is determined by stability and control analysis even though structural limits also needs to be considered. The forward limit is typically determined by the effectiveness of the elevator during takeoff nose wheel rotation or by pull-up or level turn at some flight condition [26]. The Forward CG limit is the most-left limit in the CG-envelope seen in figure 3.6. The aft limit is usually set by directional stability. Engine-out considerations might play a role and sometimes the aft limit needs to be moved forward due to spin entry or recovery problems, which is something that could be discovered late in the process after detailed wind tunnel, spin tunnel and flight tests [26]. The aft CG limit is the most-right limit seen in figure 3.6.

From this, by starting at the initial approximation of the operating empty weight (OEW) with different configurations of passengers and fuel loading etc. a span of conditions (points) of which it is safe to operate the aircraft can be calculated and a plotted with respect to weight and CG in percentage of the mean aerodynamic chord (MAC). An example of this plot is shown in figure 3.6 with restrictions such as minimum nose landing gear (NLG) load, constant load on the main landing gear (MLG) presented in table 3.1.

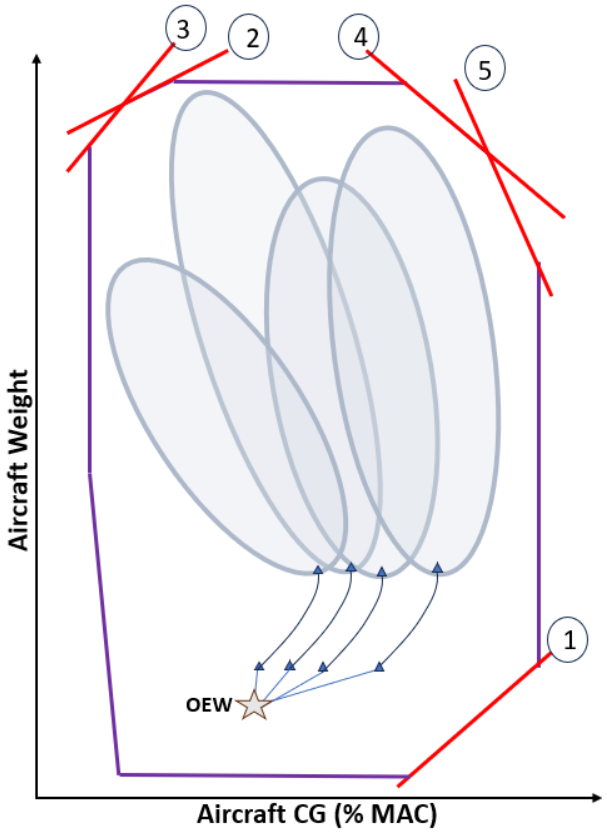


Figure 3.6: *Definition of CG-envelope*

No.	Restriction
1	Minimum NLG Load: Ground operations and steering load requirements
2	Constant NLG Load: Based on static loads, limiting gears and support structure loading
3	Horizontal stabilizer trim line (take-off): Maintains constant horizontal tail loading without having to reinforce the structure
4	Constant MLG: Based on static loads limiting gears and support structure loading
5	Fuel Vector Line: Dependent on aircraft and fuel tank configuration

Table 3.1: *Aircraft CG-envelope boundaries presented in [23]*

The second step in establishing the CG-envelope comes from 14 CFR Part 25.25 which is related to weight limits and states that [27]:

14 CFR 25.25

Maximum weights. Maximum weights corresponding to the airplane operating conditions (such as ramp, ground or water taxi, takeoff, cruise, and landing), environmental conditions (such as altitude and temperature), and loading conditions (such as zero fuel weight, center of gravity position and weight distribution) must be established so that they are not more than:

- The highest weight selected by the applicant for the particular conditions. or:
- The highest weight at which compliance with each applicable structural loading and flight requirement is shown, except that for airplanes equipped with standby power rocket engines the maximum weight must not be more than the highest weight established in accordance with appendix E of this part. or:
- The highest weight at which compliance is shown with the certification requirements of Part 36 of this chapter.

The minimum weight (the lowest weight at which compliance with each applicable requirement of this part is shown) must be established so that it is not less than:

- The lowest weight selected by the applicant;
- The design minimum weight (the lowest weight at which compliance with each structural loading condition of this part is shown); or
- The lowest weight at which compliance with each applicable flight requirement is shown.

Incorporating new regulations, the CG-envelope has been updated to include lines that represent various weight limits, as depicted in Figure 3.7. These enhancements introduce several critical weight thresholds essential for aircraft design and structural integrity:

- **Maximum Ramp Weight (MRW):** Determines the design criteria for the landing gear and supporting structures. It is the highest weight at which the aircraft is designed to operate while on the ramp before takeoff.

- **Maximum Take-Off Weight (MTOW):** Influences the design of the aircraft's wings. This is the maximum weight at which the aircraft is certified to take off.

- **Maximum Landing Weight (MLW):** A key factor in designing the landing gear, flaps, certain wing sections, the horizontal tail, and the aft fuselage. It specifies the highest weight at which the aircraft can safely land.

- **Maximum Zero Fuel Weight (MZFW):** Directs the design of the fuselage and the center wing (highest wing bending moment). This weight excludes the mass of the fuel and indicates the maximum weight of the aircraft's structure (wing-fuselage) taking into account the maximum payload (e.g. passangers, cargo etc.).

- **Minimum Operating Weight (MOW):** Represents the lowest weight at which the aircraft can operate with full operational capability, including the crew but excluding payload, passenger accommodations such as seats/galleys/lavatory and usable fuel, and other removable items.

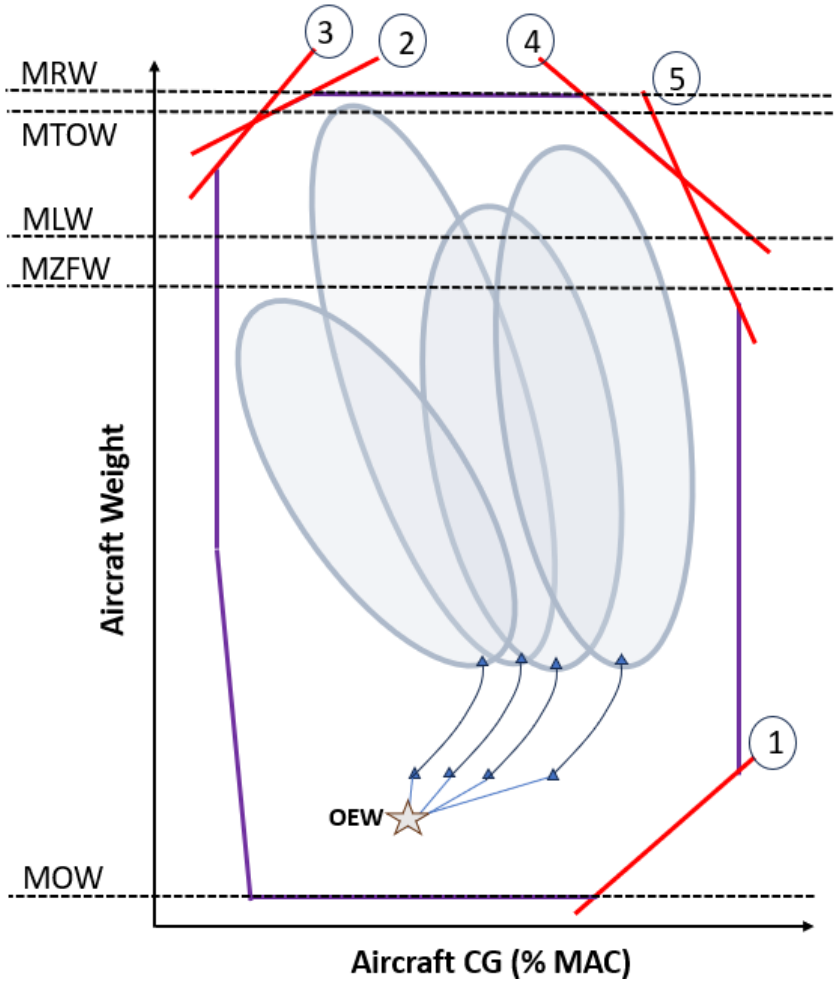


Figure 3.7: *Added weight limits into the CG-envelope*

These weight classifications are of great importance to the aircraft's design process. They ensure that each component meets safety and performance standards under various operational conditions. Starting from the initial CG-envelope calculated during the conceptual design phase, a new OEW can be derived. Each contribution is then distributed to the relevant design group, which calculates stresses, loads, and airworthiness, among other factors. A new CG-envelope is then calculated, and this process is repeated throughout the conceptual design and detailed design phases. In the detailed design phase, more precise calculations are performed until satisfactory results are achieved. Once the testing phase begins, no new CG-envelope cycles are performed unless absolutely necessary, as changes at this stage are very expensive.

3.3 Total CG, Moment of Inertia and Product of Inertia

When calculating the overall properties for a system, the CG, resulting Moment Of Inertia (MOI) and resulting Product Of Inertia (POI) for each component or part of a component needs to be calculated with respect to one coordinate system, for example the aircraft overall coordinate system. In this section the equations needed to summarize the mass properties from individual contributions into the total resulting properties are presented.

3.3.1 Total CG

In this section, we focus on calculating the total CG (CGx_{total}) along the x-axis for a system composed of multiple components or masses, necessary for later calculations.

$$CGx_{total} = \frac{\sum(m_i \cdot CGx_i)}{\sum(m_i)} \tag{3.1}$$

- $\sum(m_i \cdot CGx_i)$: Calculates the weighted sum of the x-coordinates of the center of gravity (CGx_i) for each component or mass (m_i) in the system. By multiplying each mass by its center of gravity's x-coordinate and summing these products, we obtain a measure of the total moment about the y-z plane due to the mass distribution along the x-axis.
- $\sum(m_i)$: This sums all the individual masses (m_i) in the system. It acts as a normalization factor, ensuring that the calculated total center of gravity (CGx_{total}) reflects the distribution of mass along the x-axis, independent of the system's total mass.
- The quotient of dividing the weighted sum of the x-coordinates of the center of gravity by the total mass yields CGx_{total}, the overall center of gravity of the system along the x-axis. This value indicates the average position of the system's mass along the x-axis, considering the mass distribution of all individual components and their respective distances from a reference point along this axis [28].

3.3.2 Total Moment of Inertia:

Following the assessment of the center of gravity, the discussion shifts to the moment of inertia ($I_{xx_{total}}$), a crucial property that quantifies a body's resistance to rotational acceleration around a given axis, in this case, the x-axis (roll), but the same equation could be modified for $I_{yy_{total}}$ (Pitch) and $I_{zz_{total}}$ (Yaw). This property is not only important for predicting rotational dynamics but also in designing systems that are optimized for stability and efficiency. This section shows how the distribution and orientation contribute to the total moment of inertia.

$$I_{xx_{total}} = \sum I_{xx_i} + \sum m_i \cdot CGy_i^2 + \sum m_i \cdot CGz_i^2 - m_{total} \cdot \left(CGy_{total}^2 + CGz_{total}^2\right) \tag{3.2}$$

- $\sum I_{xx_i}$: The sum of the moments of inertia of individual components around the x-axis, accounting for the inertia due to the shape and distribution of mass within each component.
- $\sum m_i \cdot CGy_i^2 + \sum m_i \cdot CGz_i^2$: These terms apply the parallel axis theorem to calculate the moment of inertia of each component about the x-axis, factoring in their mass (m_i) and their distance from the x-axis measured in the y and z directions (CGy_i and CGz_i). This adjusts the moment of inertia for the location of each component relative to a common axis.
- $m_{total} \cdot (CGy_{total}^2 + CGz_{total}^2)$: This term subtracts the moment of inertia of the entire system treated as if all its mass were concentrated at the system's overall CG from the total calculated moment of inertia. This correction is necessary to account for the overall mass distribution of the system without double-counting [29].

3.3.3 Total Product of Inertia

Lastly, the product of inertia ($I_{xy_{total}}$) is explored, which measures the distribution of mass relative to two orthogonal axes (x- and y-axes in this case) and is important for understanding the system's resistance to angular acceleration about non-principal axes. This calculation bridges the gap between the linear and rotational dynamics of a system, offering insights of forces and moments that affect the system's stability and maneuverability. This is needed to provide a deeper understanding of the system's dynamic properties and their implications for design and operation. In figure 3.8 it can be noticed that the product of inertia is zero when there is symmetry about the principal axis, to the right the positive (green) contribution is equal to the negative (red), which cannot be said about the figure to the left where there is no symmetry between the positive and negative contributions.

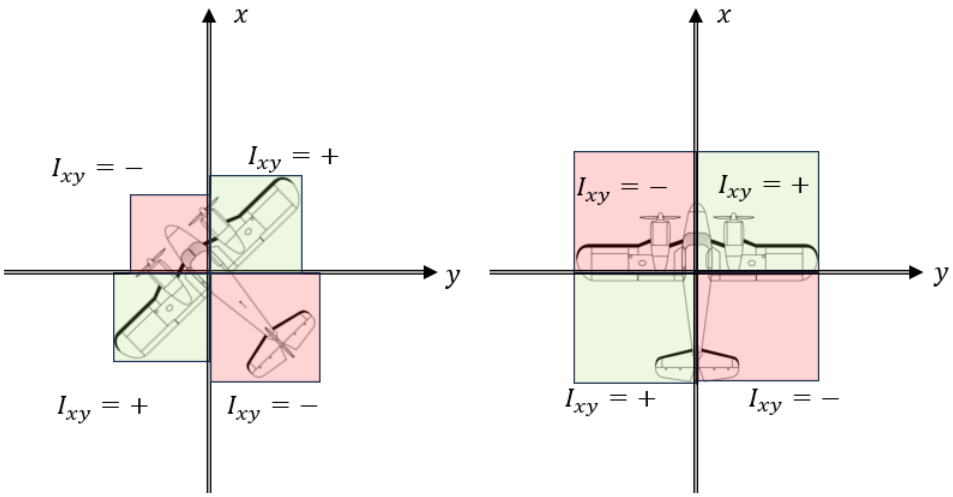


Figure 3.8: *Visualization of product of inertia. Original aircraft image by [20], modified by the authors for illustrative purposes.*

$$I_{xy_{total}} = \sum I_{xy} + \sum(m_i \cdot CGx_i \cdot CGy_i) - m_{total} \cdot CGx_{total} \cdot CGy_{total} \tag{3.3}$$

- $\sum I_{xy}$: Represents the sum of the products of inertia for individual components of a system.
- $\sum(m_i \cdot CGx_i \cdot CGy_i)$: Calculates the sum of the products of each component's mass (m_i) and its center of gravity coordinates (CGx_i and CGy_i) along the x- and y-axes, respectively. This term considers the effect of the mass distribution relative to a defined origin on the product of inertia.
- $m_{total} \cdot CGx_{total} \cdot CGy_{total}$: Subtracts the product of the system's total mass (m_{total}) and the coordinates of its overall center of gravity (CGx_{total} and CGy_{total}) in the x and y directions. This adjustment ensures that the calculation reflects the net effect of the entire system's mass distribution on the product of inertia [30].

3.4 Rotation and Translation Between Coordinate Systems and Intersection Between Lines

To ensure that the aircraft's design is evaluated across a range of operational conditions such as change in attitude, including varying pitch, roll and yaw angles, it is important to be able to shift the coordinate system accordingly. This change enables more precise obtainment of mass properties data for the aircraft during different operations without the need for altering the existing computer-aided design (CAD) models directly. Having this option is invaluable in case of some error during the calculation process since it requires no restoration of the models.

When the coordinate system is rotated to simulate these varied attitudes, the model's boundary points, specifically the maximum and minimum points shift according to the new orientation of the axes. As a result of this rotation, these extremities sketch out new boundary lines that are parallel to the orientation of the rotated axes. The intersection of these boundary lines forms a rectangular area, effectively encapsulating the model within the confines of the newly adjusted coordinate system. This can be seen in figure 3.9 where the colored (orange or blue) lines define the new box.

Moreover, there's a requirement to transition coordinate systems from the localized context of individual components and parts to the global coordinate system of the aircraft. This transition is important to make sure that all components are evaluated both accurately but mainly consistently with all parts referencing to the global aircraft coordinate system.

3.4.1 Rotation of Coordinate System

The rotation matrix for an angle α is defined as:

$$R(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \tag{3.4}$$

Given a point with coordinates (x, y), the coordinates (x', y') after a rotation by the angle α are obtained by:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = R(\alpha) \begin{bmatrix} x \\ y \end{bmatrix} \tag{3.5}$$

Expanding the matrix multiplication, the new coordinates are as follows:

$$x' = x\cos(\alpha) - y\sin(\alpha) \tag{3.6}$$
$$y' = x\sin(\alpha) + y\cos(\alpha) \tag{3.7}$$

In figure 3.9 below, an example of rotation with the angle alpha is shown and the maximum and minimum points is presented both before and after the rotation if the object is kept stationary[31].

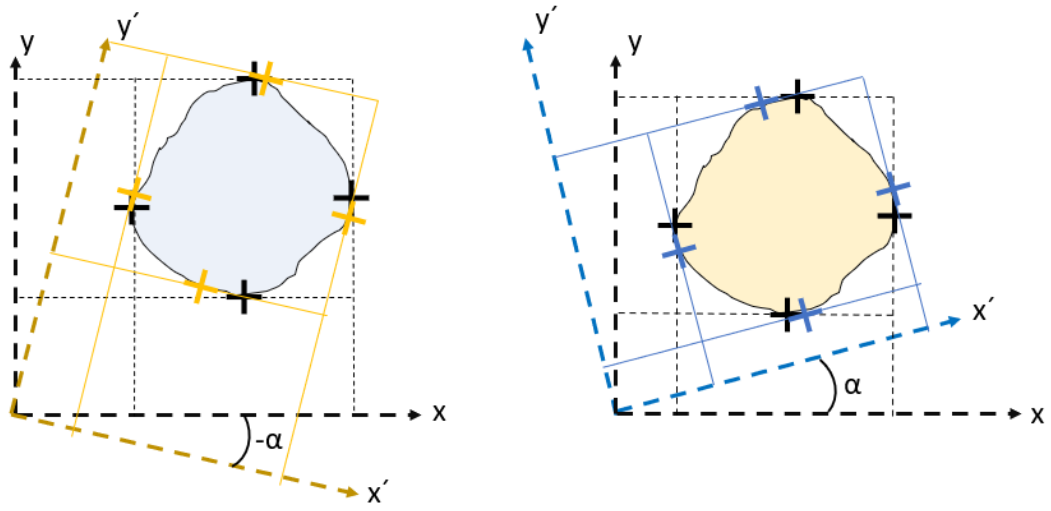


Figure 3.9: *Rotation of coordinate system*

3.4.2 Intersection Between Two Lines

To calculate the intersection point between two lines, where each line is is described by a point of origin and angle relative the local coordinate system. The calculation involves algebraic manipulation and trigonometric identities. Consider line 1 passing through point $p_1 = (x_1, y_1)$ with an angle $-\alpha$, and line 2 passing through point $p_2 = (x_2, y_2)$ with an angle $90 - \alpha$, see figure 3.10.

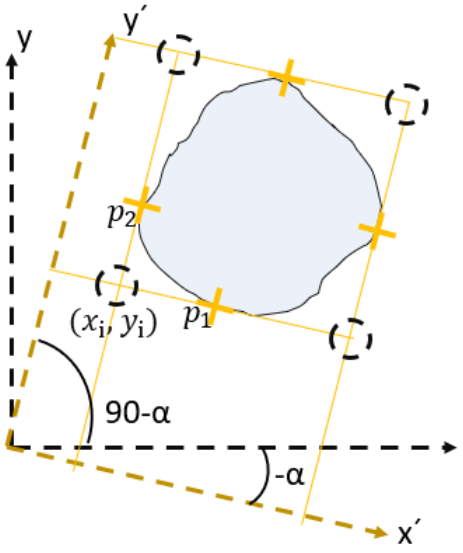


Figure 3.10: *Intersection point between two lines marked with a circle*

The slope of line 1, given its angle $-\alpha$, is $m_1 = \tan(-\alpha)$. Similarly, the slope of line 2, with its angle $90 - \alpha$, is $m_2 = \tan(90 - \alpha)$. However, since $\tan(90 - \alpha)$ is equivalent to the cotangent of α, which is $\frac{1}{\tan(\alpha)}$, the equations of the lines can be expressed as:

$$y - y_1 = \tan(-\alpha) \cdot (x - x_1) \quad \text{(equation of line 1)}$$
$$y - y_2 = \frac{1}{\tan(\alpha)} \cdot (x - x_2) \quad \text{(equation of line 2)}$$

To find the intersection point (x_i, y_i), we set the right-hand sides of these equations equal to each other and solve for x:

$$\tan(-\alpha) \cdot (x - x_1) = \frac{1}{\tan(\alpha)} \cdot (x - x_2) \tag{3.8}$$

Solving this equation for x gives the x coordinate of the intersection point (x_i). The solution is:

$$x_i = \frac{y_1 - y_2 + x_1 \tan(-\alpha) + x_2 / \tan(\alpha)}{\tan(-\alpha) + 1/\tan(\alpha)} \tag{3.9}$$

Substituting x_i back into one of the original line equations provides the y coordinate (y_i) of the intersection point. Assuming substitution into the equation of line 1:

$$y_i = y_1 + \tan(-\alpha) \cdot (x_i - x_1) \tag{3.10}$$

This method yields the coordinates of the intersection point based on the known angles and points on each line, allowing for precise geometric analysis in rotated coordinate systems.

3.5 Overview of CAD Mass Property Calculations Capabilities

CAD is widely used in aircraft design for modeling, simulation, and analysis. The complexity of aircraft systems requires a lot of precision to ensure performance, safety, and compliance with regulatory standards. Accurately matching real-world numbers to the limitations of the aircraft flight envelope is crucial, as deviations can lead to heavier components, which in turn impose stricter limitations on the envelope. This creates a vicious cycle in aircraft design, highlighting the importance of accurate measurements and assumptions [32].

While CAD's utility for measurements is limited in the initial design phase due to the need for detailed models, it becomes invaluable later. CAD allows for precise measurement and simulation of components without manufacturing, enabling cheaper iteration and optimization. This capability supports numerous design iterations that would be impractical with physical prototypes. CAD's accuracy in simulating and adjusting designs is crucial for refining aircraft efficiently, eliminating the need for constant physical prototypes and overcoming design constraints, making it essential for the iterative process of aircraft design and optimization [33].

An important consideration in using CAD, however, connects closely to the principle of "garbage in, garbage out" This means that the quality of the results is directly dependent on the quality of the models created. For an aircraft, achieving accuracy for each component can be challenging due to the vast range of parts, from screws to wing surface sections. Each part must be accurately designed with the correct material and geometrical properties [34].

3.6 Automating CAD for Enhanced Data Collection

Transitioning from traditional technical drawings and mock-ups to CAD-based development has significantly streamlined the aircraft design process [33]. However, CAD systems are not without their drawbacks, as certain modeling and analysis tasks can be labor-intensive. To enhance efficiency, one can make use of automation tools such as VB.Net for the creation of macros. These macros automate repetitive tasks, thereby reducing manual effort and expediting the design process. This integration not only streamlines the collection of data but also optimizes the overall workflow, making it an important advancement in CAD utilization. The capability to execute multiple commands swiftly with VB.Net macros in CAD software unlocks various new applications. For instance, it allows for the segmentation of multiple parts to ascertain their mass properties and the automation of exporting data into tools like Excel [35][36].

Another crucial aspect of automating Catia tools is ensuring consistent and accurate calculation of mass properties and the CG envelope. By automating these tasks, the benefits of KBE can be leveraged to maintain reliable results. The conceptual phase of aircraft design often involves rapid design iterations, making consistent mass properties and CG envelope calculations important. Especially since the differences in mass properties between revisions frequently serve as the basis for decision-making.

Integrating KBE principles into Catia tool automation allows engineers to embed design rules and domain-specific knowledge, streamlining the process. By codifying calculation methodologies, KBE enables automated tools to consistently and efficiently perform these tasks. This accelerates the design process and ensures adherence to standards and best practices.

3.7 Understanding VBA and VB.Net in CAD Integration

Visual Basic for Applications (VBA) and VB.NET are two sub-languages of the Visual Basic programming language that offer varying capabilities within different environments such as CAD software and the Office suite. VBA is an event-driven programming language from Microsoft that is predominantly used for automation within Microsoft Office applications. It is built into most Microsoft Office applications but also into the Catia environment and allows for the creation of user-defined functions, automation of tasks, and access to the application API. One advantage with VBA and the direct integration into the applications is the possibility to record different macros within the program to help understand how the API interact with certain functions [37].

VB.NET is a modern, object-oriented programming language based on VBA. In contrast to VBA, that provides a more powerful framework for software development. Designed as the successor to VBA, VB.NET runs on the .NET framework, which allows for robust application development, including enhanced security, memory management, and exception handling capabilities. It also allows for creating stand alone .exe applications thereby allowing for user interaction through UI elements [38].

The structure of both VBA and VB.NET is based on the BASIC programming language. VBA offers simplicity, suitable for straightforward scripts and macros, while VB.NET supports a comprehensive programming approach for more sophisticated applications [38]. Their clarity and structure are especially advantageous for detailed CAD

operations, where accuracy and consistency is important since an error in the code midway can leave modifications to the model not easily spotted and rectified.

3.7.1 COM Objects and Their Interaction with Catia and Excel

Component Object Model (COM) objects are an integral part of the interaction between Visual Basic and applications such as Catia and Excel. COM is a Microsoft-developed platform that enables communication between objects created in different applications or programming languages, provided they adhere to the COM standard [39].

In the context of CAD and data analysis, COM objects serve as an interface to automate and manipulate the functionalities of Catia and Excel through VBA or VB.NET. This automation is possible because both Catia and Excel expose their functionalities as COM objects, allowing a developer to create, access, and manipulate documents, execute commands, and retrieve data through code rather than using the functions manually [40].

For example, by using COM objects in VB.NET, developers can instruct Catia to generate parts, edit part-data, or extract mass properties without manual intervention, effectively communicating the desired operations via method invocations on the objects [40]. Similarly, COM objects allow VB.NET to open Excel workbooks, write data to sheets, and generate charts, thus enabling a seamless data processing and visualization workflow. The use of COM objects in this manner significantly enhances the efficiency of data collection and analysis, enabling automation of repetitive tasks, complex calculations, and sophisticated data manipulation strategies, all from within the VB.NET environment [41].

3.8 ATA-100 numbering system

The ATA-100 numbering system is an industry-standard framework that categorizes aircraft components, systems, and assemblies according to a chapter-based structure. Originally developed by the Air Transport Association (now known as Airlines for America, A4A), this system was designed to enhance consistency and efficiency in maintenance, documentation, and procurement processes across the aviation sector [42].

This structured system organizes aircraft into various chapters, each identified by a unique two-digit number, and further divides these chapters into detailed sections for more precise classifications. For instance, chapter 57 - *"Wings"* , encompasses all systems and components associated with the aircraft's wings, including structural elements, control surfaces, and associated hydraulics. For instance, in the context of an aircraft maintenance manual, the section discussing wing tips might be further identified by a unique subsection number or title within Chapter 57, such as 57-XX, where "XX" represents the specific subsection dedicated to the wing tip and its associated components, maintenance, and inspection criteria.

To mention some, as an example of chapters included within the ATA-100 system:

- Chapter 32: Landing Gear
- Chapter 52: Doors
- Chapter 57: Wing

The ATA-100 system facilitates standardization across the aviation industry, simplifying maintenance tasks, and enhancing communication among engineers, maintenance personnel, and logistics teams by providing a unified language for aircraft parts and systems.

3.8.1 Evolution of ATA Documentation Standards

The ATA-100 numbering system, as described earlier, laid the foundational framework for the classification of aircraft components, systems, and assemblies. Its chapter-based structure facilitated consistency and efficiency across the aviation sector. However, the pace of technological advancement and the increasing complexity of modern aircraft systems necessitated the development of more comprehensive documentation standards. This led to the introduction of the Joint Aircraft System/Component (JASC) Code Table and subsequently, ATA iSpec 2200. These newer standards aimed to address the limitations of ATA-100 and adapt to the evolving needs of the aerospace industry.

3.8.1.1 JASC/ATA Codes

The JASC/ATA system represents an evolution of the ATA-100 standard, incorporating the Joint Aircraft System/Component (JASC) Code Table. This system enhances the specificity and detail of aircraft system classification, allowing for a more accurate categorization of components and maintenance activities. It retains the chapter-based organization of ATA-100 but introduces additional codes to represent systems and components more precisely. This level of detail is particularly beneficial for modern aircraft, where the complexity and diversity of systems exceed what ATA-100 was originally designed to cover [43].

3.8.1.2 ATA iSpec 2200

ATA iSpec 2200 is a comprehensive standard that encompasses both the ATA-100 and JASC/ATA codes, offering a complete suite of guidelines for aircraft maintenance documentation. It covers aspects ranging from technical documentation to maintenance requirements and procedures, aiming to support the entire life-cycle of an aircraft. ATA iSpec 2200 is designed to be flexible and scaleable, accommodating the rapid technological advancements in the aerospace industry [44].

3.8.1.3 Continued Use of ATA-100 by Major Manufacturers

Despite the introduction of these advanced standards, many large aerospace companies, including industry giants like Boeing and Airbus, continue to utilize the ATA-100 system for several reasons. One primary factor is the extensive documentation and maintenance processes built around ATA-100. Transitioning to a new system entails significant changes in documentation, training, and software systems, which can be costly and time-consuming.

Moreover, ATA-100's simplicity and broad acceptance across the industry make it a practical choice for ensuring compatibility and understanding among a wide range of stakeholders, including smaller airlines, maintenance facilities, and parts suppliers who may not have the resources to quickly adapt to newer standards.

4
Methods

This chapter outlines the methods used to evaluate the mass properties of fuel tanks in aircraft during the conceptual design phase. It introduces a specialized tool that uses existing fuel tank solid models and segments it into slices for simplified analysis, allowing for quick changes without complex simulations. In addition to the fuel tool, the methodology for a more general weight evaluation tool is described. Similar to the fuel tool it can be used for quick evaluation of changes to the computer models during conceptual design. Another section of this chapter discusses a weight control tool that uses Excel for analyzing and managing aircraft weight distribution. Additionally, the chapter covers a case study aircraft designed during a previous aircraft design course, which serves as a practical application of these tools, illustrating their effectiveness in real-world scenarios.

4.1 Fuel Tank Mass Properties Tool and Post Processing of Data

When analyzing the mass properties of a fuel tank, it is important to consider various scenarios with differing fuel levels. This task becomes challenging during the conceptual design stages, as the fuel tank's design may change frequently. Therefore, the process of determining these properties should be more efficient than conducting multiple, detailed computational fluid dynamics (CFD) simulations for a tank transitioning from full to empty. As a result, after consultation with the fuel tank's design team and the mass properties engineer, the decision was made to treat the tank as a single solid entity. This method overlooks the internal reinforcements and structures, focusing solely on the fuel volume. By adopting this strategy, the tank's CAD model can be divided into slices, each approximately 10 mm thick, allowing for the simulation of gradual fuel depletion, by starting at zero slices and then add a new slice to represent more fuel.

Given the unique function of the fuel tank, and especially the collector tank (CT) which remains filled while the rest of the tank empties, sectioning must occur longitudinally to be able to isolate the contribution from the segment containing the CT. This approach ensures that when isolating the CT, the post-processing allows for the mass in the CT to be preserved, while simulating the removal of fuel (slices) from other parts of the tank, thus not distributing the mass evenly throughout.

By doing it this way, the dynamic behaviour of the fuel tank can be studied together with the difference in load distribution and CG when filling the tank (no active CT) and when running the generators (active CT).

In Figure 4.1, a flowchart depicting the operation of the fuel tank tool is introduced. This tool begins its process when the user selects a part within Catia for segmentation, in this instance, the volume of the fuel tank.

Once the tool identifies the selected object as a part, the user can determine the specifics of the segmentation process. The initial step involves deciding whether the tank needs to be rotated to simulate various pitch or roll angles. Should rotation be required, the user must also specify the axis around which the rotation is to occur. Following this, the user decides on the segmentation approach.

For example, referring to the general layout of the tank model discussed in the theoretical framework (see Figure 3.2), the model includes four segments. Assuming the wing aligns with the y-axis, this would necessitate creating four sections along the y-direction, yielding five y-coordinate values. Given a tank thickness of $50cm$, opting for 50 slices along the z-direction results in four segments, each divided into $10mm$ thick slices, cumulatively providing 200 data points regarding mass properties.

Should the initial visualization of the segmentation reveal inaccuracies in the placement of cuts, the user has the flexibility to adjust the y-coordinates to better align with the intended model. Upon finalizing these coordinates, the tool computes the CG and inertia for each segment, writing the information into an Excel document, with separate sheets dedicated to each segment. This structure ensures that the analysis is both precise enough and quick making it usable during the conceptual design process.

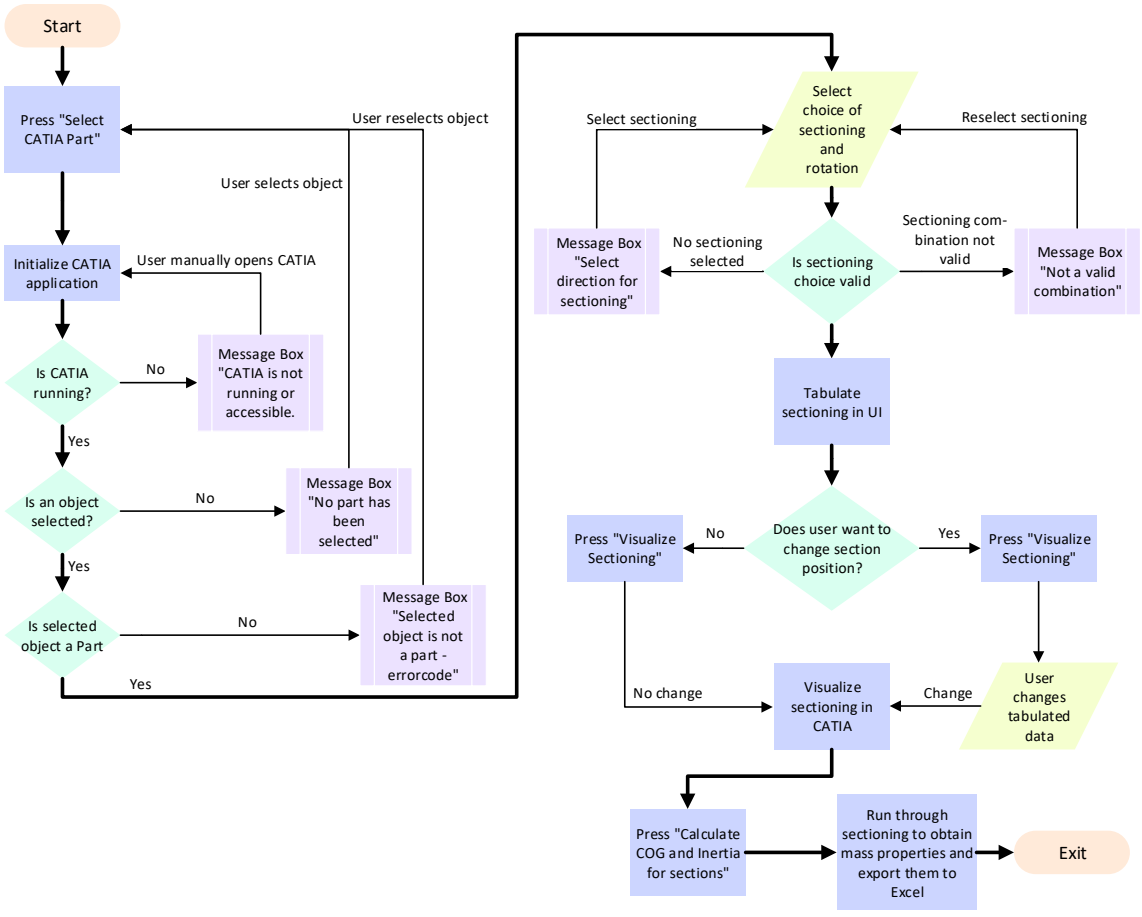


Figure 4.1: *UI flowchart for the fuel tank-tool*

4.1.1 Software Development Environment

The application was developed in a .NET framework environment, utilizing the Visual Basic (VB-net) programming language. This choice was influenced by Visual Basic's integration capabilities with the Catia V5 automation interface, allowing for direct manipulation of CAD models through the Catia API. Microsoft Visual Studio served as the primary integrated development environment (IDE), providing essential tools for code development, debugging, and testing. In addition to this it provides free templates for UI elements that is clear for the user to understand.

4.1.2 Libraries and External Tools Integration

Interaction with Catia V5 was achieved through its automation interfaces, including `INFITF`, `MECMOD`, `PARTITF`, and `HybridShapeTypeLib`, facilitating access and manipulation of parts, products, and their geometric elements. Moreover, the Microsoft Office Interop Excel library was employed for data export and analysis, enabling automated interaction with Excel spreadsheets for data entry and report generation.

4.1.3 User Interaction and Feedback

The development of the tool prioritized ease of use, focusing on a simple and intuitive interface to ensure user accessibility. The design incorporates interactive elements such as buttons, text fields, and drop-down menus, facilitating straightforward user interaction. Progress indicators, including bars and percentages, are integrated to transparently display task advancement, aiding users in monitoring the tool's progress efficiently.

A notable feature of the tool is its capability to generate visual representations within Catia for varying conditions, such as different AoA. These visual aids are crucial for users to verify the tool's results promptly, increasing confidence in its application.

The tool's layout is designed to be logical and user-friendly, adopting a conventional reading order from left to right and top to bottom, ensuring clear navigation and understanding. This layout strategy, while seemingly straightforward, plays a significant role in enhancing the tool's usability.

4.1.4 Data Handling and Export

Integration with Excel allows for the export of optimization data, including component weights, dimensions, and optimization metrics, for further analysis and reporting. This makes for an easy way of checking the generated data and also validating it with Catia values.

4.1.5 Post-processing Data

For post-processing of data, Microsoft Excel was utilized, specifically for the analysis and visualization of key metrics such as the CG along the x-axis. This method took advantage of Excel's robust functionality to organize data, interpolate missing values, and graphically depict complex datasets.

Interpolation was a crucial step in the data analysis. Since the methodology involved segmenting the tank into thin slices, direct measurements for every conceivable volume were not available. Through interpolation, the post-processing Excel were able to estimate values for these unmeasured volumes, enabling a comprehensive analysis across various fuel levels.

Visualization also played significant role in the tool development. Graphs depicting the relationship between the CG in the x-direction and other significant variables like mass and volume and inertia's provided clear insights into the dynamic properties of fuel tank during different operational scenarios, such as varying pitch angles and fuel volumes. These visual representations made complex datasets more accessible and can be used to ease the iterative design process by highlighting crucial trends and patterns not immediately apparent in the raw data gathered from the fuel tank tool.

Further, Excel was employed to calculate critical parameters for different tank segments, looking at the total volume needed and then combining the different segments of the tank such that the total height for all slices within each segment is vertically on the same level, except when the CT is active, in which case the post-process excel will not take other segments into account until the total volume is above that of the CT. The data presented from this includes the CG, total MOI and POI, and the total mass for each segment of the tank. These calculations were performed using equations 3.1, 3.2, and 3.3 and the total mass by summing the contribution from each slice for every segment that is included for the specified volume. The results were then systematically tabulated for analysis.

The Excel framework also allowed for scenario simulations, where the user could adjust variables such as the total mass and fuel density to examine specific flight conditions, with both turbo generators running and having them shut off when filling the tank to certain levels. This feature proved invaluable for assessing the distribution of mass and the movement of the center of gravity relative to the fuel volume.

Upon finalizing the Excel file for post-processing, a Python script was developed to automate the implementation of raw data generated by the fuel tank tool to the post-process Excel sheet. This automation is particularly useful when gathering data from various pitch angles, efficiently updating every dedicated Excel file with interpreted data for each pitch angle.

For each pitch angle examined, a dedicated Excel file was created. Another Python script was then created and used to compare these files, presenting a comprehensive view of how the CG shifts with fuel volume across all tested pitch angles. This level of automation significantly enhanced the ability to manage and interpret extensive datasets, streamlining the process of deriving insights from the fuel tank's performance under various conditions, displayed in a single plot.

4.2 Weight Control Tool (WCT) to Obtain CG Envelope

To leverage the advantages of KBE this study introduces an Excel-based tool designed to manage and analyze aircraft weight distribution. By creating a workbook that can be generalized for multiple projects and revisions and creating automated logic for different loading conditions the knowledge can be captured and repeated. It makes use of VBA macros within Excel to automate routine tasks. The methodology is based on a division of the aircraft into its constituent sub-assemblies such as fuselage, wing, vertical tail etc. In addition to this the division is set up for further refining by including specific bays within each sub-assembly. The idea with this is to be able to keep as much detail as possible without making the data unmanageable. In figure 4.2 an example of how the different bays can be divided is shown, focusing on the fuselage, the first number (1) indicates it is the fuselage. The subsequent numbers (01,02,03,04, etc.) indicate the position span in the longitudinal direction. This can be further detailed, for example, when examining the wing, which can be divided to include the aileron and other surfaces. Suppose the wing is divided into bays numbered 201, 202, etc. This numbering indicates the span in the lateral direction. The aileron could then be further divided into 201.1, 202.1, with the number after the comma specifying the sub-part of the wing.

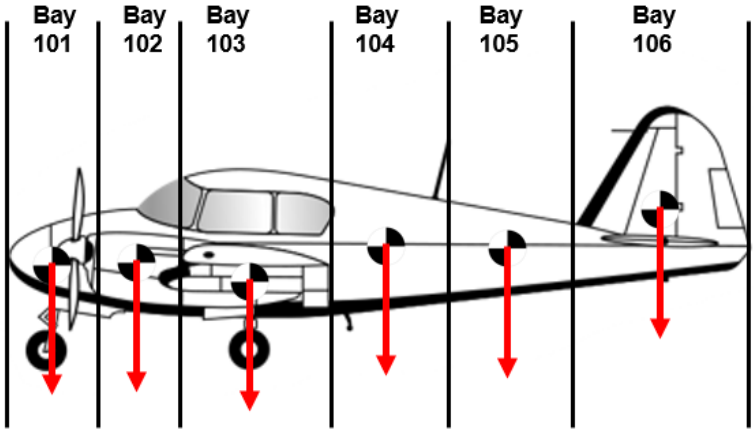


Figure 4.2: *Example showcasing different bays of the fuselage. Original aircraft image by [20], modified by the authors for illustrative purposes.*

4.2.1 Excel Spreadsheet Configuration

The foundation of the methodological approach involved the configuration of an Excel spreadsheet designed to represent the aircraft's architecture comprehensively. Each sub-assembly was allocated a dedicated sheet within the workbook. These sheets were organized to catalog each component part, providing a detailed framework for subsequent analysis. An important aspect of the spreadsheet setup was the introduction of a specific sheet for bay definitions, enabling better division and categorization of parts according to not only their function but also physical location within the aircraft. While not as important for understanding the CG of the aircraft. It is important for calculating loads on the aircraft as the loads team requires loads to be split into nodes. To get an represen-

tation, these nodes need to be summarized point loads. Thereby it is important to not only split by function but by location, achieving a balance between sufficient detail and avoiding excessive data. The loads team can later on provide limits to the CG-envelope, a short representation of this cycle is presented below in figure 4.3.

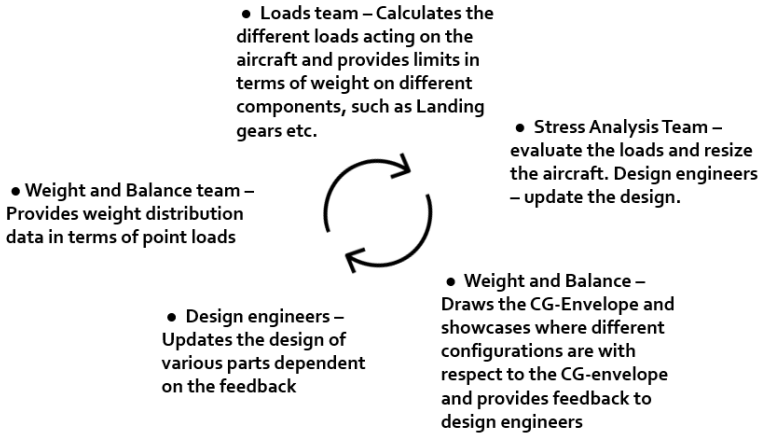


Figure 4.3: *Weight and balance, loads, CG-Envelope, design cycle example*

Another important feature of the aircraft weight management tool is its capability to monitor not just the changes in individual parts but also modifications within sub-assemblies and the entire assembly throughout the development phase. This functionality is essential for pinpointing the origins of weight and CG adjustments and observing weight trends over time. Such insight is particularly valuable during the conceptual stage of aircraft design, a phase often characterized by weight increases due to overly conservative safety margins across various sections of the aircraft. By tracking the weight across different sub-assemblies, the weight management specialist can provide feedback to other development teams regarding their progress in meeting weight goals, identify areas needing improvement, and compare actual weights against the projected targets for each segment.

4.2.2 Automation via VB.NET Macros

To mitigate the potential for extensive manual data entry and analysis, the methodology incorporates the use of VBA macros. These macros are designed to automate aspects of the weight distribution analysis process. One example of this is to more easily categorization of aircraft components into their respective bays automatically, which not only lower process time but also risk of error. This automation extends to the dynamic generation of CG envelopes, integrating logic to model various loading scenarios accurately. The addition of passenger and cargo distribution simulations within the CG envelope is necessary to get a good estimation and understand all extremes of the flight envelope.

4.2.3 Data Analysis and CG-Envelope Generation

A central part of the tool is the ability to compile and analyze the data from individual components and sub-assemblies in a more compiled overview. This was done through the creation of a comprehensive CG envelope for the aircraft. This process not only incorporates the static- but also the variable factors of the aircraft, associated with different

loading and fuel conditions. Through the execution of VBA macros, the tool is able to quickly calculate and visualize the CG-envelope, providing a valuable resource for assessing the aircraft's weight and balance. Furthermore, the tool is equipped to simulate the impact of design modifications, such as alterations to the wing's position or extensions to the fuselage, offering predictive insights into their potential effects on the aircraft's weight distribution and overall performance. In the conceptual stage this information is a good ground for design decisions.

4.3 Mass Properties Tool for Assemblies

4.3.1 Objective

The objective of the Mass Properties Tool is to gain insight in how the mass of individual parts is spread across the whole assembly and allow for section point loads to be calculated accurately and quickly. This is a valuable tool in the conceptual stage since there is a lot of changes and a lot of "customers" such as the loads and design teams that requires up-to-date mass and balance properties at different levels of accuracy. This tool is designed to interface with CAD software, using VBA and Catia V5 automation libraries for dynamic interaction with CAD models. By automating complex tasks, the tool accelerates the design and analysis process, reduces the potential for error and decreases the time, something that is important in a quickly changing development environment. It allows for an in-depth examination of how design changes affect mass distribution and product performance, providing valuable insights early in the development process. Integration with Excel for data processing and visualization enhances the tool's utility, offering a platform for more detailed data analysis.

4.3.2 Methodology

At the start, the program establishes a connection with Catia and Excel through COM objects. Initially it runs through the product structure and catalogues all parts and their respective geometrical minimum and maximum. This is important when working with large models since it is important to minimize the number of unnecessary steps. This initial step also allows for a global minimum and maximum to be created which is important when deciding the sectioning positions. In addition to this the tool also visualizes the bounding box and the sections which is an important step since it allows the user to get a graphical verification that the sectioning is correct. Since there are endless possibilities when creating parts and products this also means that the tool will not work in all cases. For example the tool is limited to working with solid parts but there might also be other limitations highlighting the importance of being able to verify the process and results.

After cataloging each part and defining the sections based on user input the program goes though each part and sorts it into the respective section. An important note here is the possibility for a product to be in more than one section. In this case since the min and max values of the part will be between two different sections this part will need to be sectioned and measured for each respective section. By having these minimum and maximum values the tool can easily sort a part based on if it is partially, fully or not in a section which saves a lot of computational steps speeding up the process. Moreover, the tool integrates with Excel which allows for post processing of the data and a good platform for displaying the data generated. By providing detailed data in a structured format, the tool can provide the user with the information needed for the design process.

A flow chart depicting the user workflow for the Mass Properties Tool can be seen below in figure 4.4.
The workflow can be summarized as follows:

- **Initialization**: Press "Select CATIA Part" to initialize CATIA. If CATIA is not running, prompt user to manually open it.
- **Object Selection**: Ensure CATIA is running and an object is selected. If no object is selected, prompt the user to select one.
- **Validation**: Check if the selected object is a valid product. If it is not, display an error message.
- **Bounding Box Calculation**: Use the "Calculate BoundingBox" button to create a temporary product and part. Iterate through objects in the product tree to calculate and update extremum points.
- **Visualization**: Visualize the bounding box as a volume in the temporary part based on the calculated global extremum.
- **Sectioning**: Select the part type and the number of sections. Create and visualize the initial sectioning. Update sectioning volumes in the user interface.
- **Mass Properties Calculation**: Calculate and store mass properties for each part. Copy parts to a temporary product, process them for sectioning, and store the properties in an Excel sheet.
- **Section Handling**: Evaluate if parts are within, outside, or partially within sections. Store properties or split parts based on sectioning volumes. Iterate through all sections and parts until the process is complete, then exit.

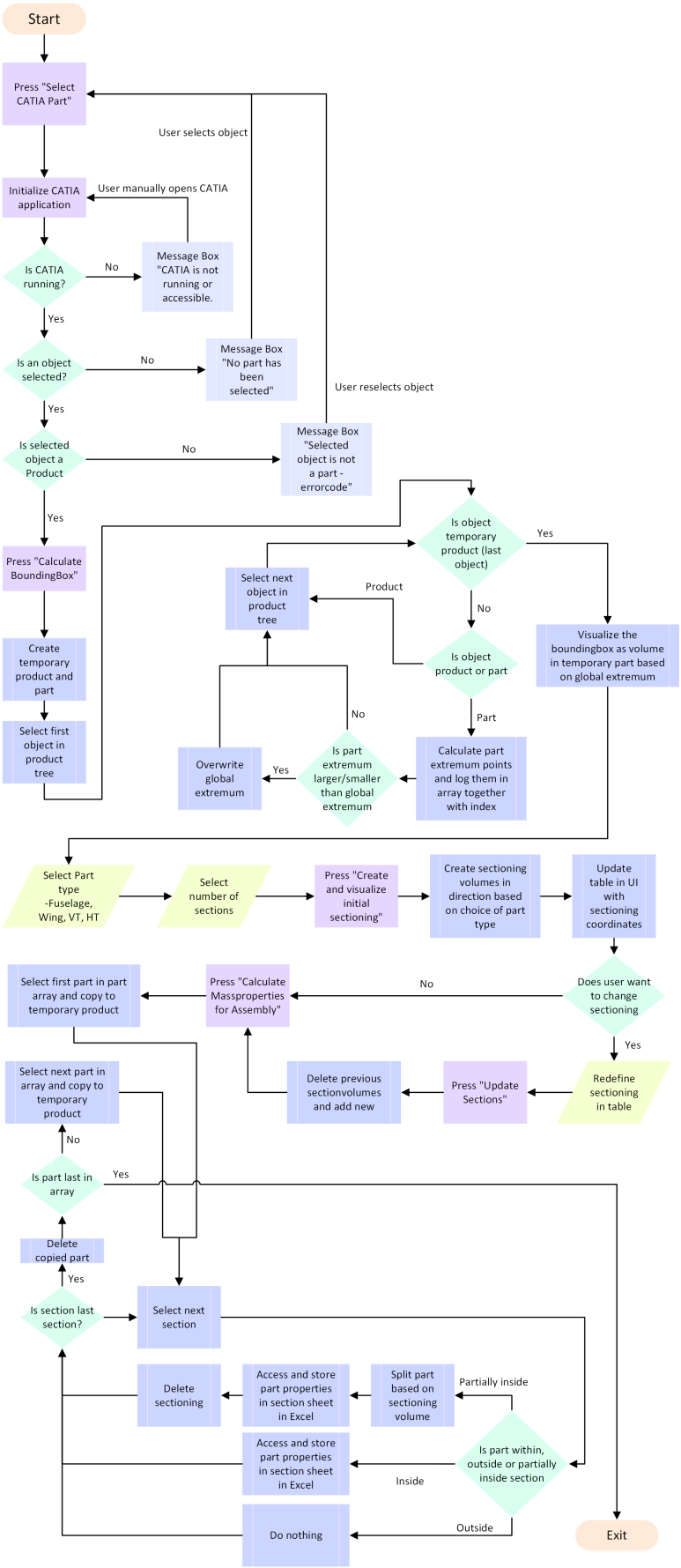


Figure 4.4: *Flow chart of mass properties tool*

4.4 CSA Design Method

The CSA, as mentioned earlier was designed during the aircraft design course, MMS236. However, here follows a brief description on how it was designed, its dimensions, weights and estimated CG.

The design weights and dimensions was determined using a Python script that performs the design steps by Raymer [6], the full code is appended in appendix A. The code starts of by defining the initial data, such as speed, altitudes, range, EIS, number of crew, pilot and so on.

Then by estimating from the reference aircraft and using tabulated data from [6], initial values for the lift and wing area are gathered. After this a value for the specific fuel consumption (SFC) is estimated by an extended trend line from data of previous years to get an estimate of a realistic SFC for 2035. This is then converted to values that represent SFC for LH_2. From this the sizing loop by Raymer [6] could be implemented. This methodology integrates core aircraft design concepts, utilizing empirical data and statistical correlations to compute a realistic MTOW for the aircraft under design.

The MTOW, (w_0) in the Python code is determined through the following sequence:

1. **Initial Guess:** The calculation starts with an initial guess for w_0, denoted as w_0_guess.
2. **Weight Fractions:** For each flight phase—including warmup, takeoff, climb, cruise, landing, and loiter—weight fractions are computed. These represent the fuel consumption as a proportion of w_0.
3. **Iterative Loop:** The code enters an iterative loop, where:
   - The total fuel fraction (w_fuel_fraction) is calculated by summing the individual weight fractions and reserves.
   - An empty weight fraction (w_empty_fraction) is obtained using a function that accounts for the aircraft's empty weight, potentially corrected for composite material usage.
   - A new estimate for w_0 is produced based on the weight of the crew, passengers (w_PAX), and the computed fuel weight (w_fuel).
   - The w_0 value is refined by averaging the new estimate with the previous guess, aiming to converge on the precise MTOW.
   - The iteration continues until the difference between consecutive w_0 estimates falls below a certain tolerance.
4. **Final Weight Calculation:** Upon convergence, w_0 is established as the final MTOW, inclusive of crew, passenger, and fuel weights. The fuel tank weight, derived from a gravimetric index, is then added to w_0 to finalize the MTOW.

After the sizing, power to weight ratios are determined, using statistical power to weight ratio with Raymers correlation [6]. Then the wing loading is calculated for the different stages during the flight. From which we get the wing reference area. Then the propeller and nacelle size is determined from looking at the reference aircraft and using statistical correlations once again.

The tail wing size was designed using Gudmundsson [45] with values for the Dash-8 Q400. Then the wheel sizing and forces acting on the wheels was determined using Raymer once more [6].

From this the initial sizing of the CSA was determined, then a few iterations was conducted to make sure that the CG doesn't move to much aft or forward depending on the payload and fuel and then a few iterations on the wing sizing after the in flight

stability analysis was conducted. Both CG and stability analysis was conducted of a model of the CSA in openVSP together with VSPAERO and parasitic drag tool, this is what was used to get the CG for different configurations during the design course.

This is also one of the steps in conceptual design where the weight management tools will be implemented to investigate how it could help the design process and increase the accuracy. One mayor thing missing from the aircraft design course is also that the CG-envelope is never investigated since the time period is to short to include everything. The tool could hopefully help with that.

4.4.1 CSA Design Fudge-factors

In addition to the statistical correlations, fudge factors were applied to several key components of the aircraft. These fudge factors are adjusted values that take into consideration improvements in materials and structures that might not be fully accounted for by the statistical models alone. The range for each of these factors was derived from [6]. When selecting fudge factors, a conservative approach was taken by favoring the middle or upper values within the given range.

- Wing: Fudge factor - 0.88
- Horizontal Tail: Fudge factor - 0.86
- Vertical Tail: Fudge factor - 0.86
- Fuselage: Fudge factor - 0.93
- Nacelle Group: Fudge factor - 0.93
- Main Landing Gear: Fudge factor - 0.97
- Nose Landing Gear: Fudge factor - 0.97
- Installed Engine: Fudge factor - 0.90

4.4.2 CSA Key Values and Weight Breakdown

In Table 4.1 a portion of the statistically gathered weights for various components are displayed, this table includes all structure components and components which has a weight that is equal to or more than 2% of the total weight. These are the components which are included in the CAD model and will be included when analyzing the design with the weight management tools.

4.4.3 Component Weight Breakdown

Table 4.1: *Smaller breakdown of aircraft component weights, including structures and weights above 2% of the total.*

Category	**Component**	**Weight (kg) / % of Total**
Capacity	**Passengers and luggage**	10354.0 kg / 34.44%
Equipment	**Furnishing**	666.61 kg / 2.22%
Operational Items	**Unit Load Device**	665 kg / 2.21%
Propulsion	**Installed engines**	1868.53 kg / 6.22%
	Tank	3334.60 kg / 11.09%
	Fuel	1795.50 kg / 5.97%
Structure	**Wing**	2253.94 kg / 7.50%
	Horizontal Tail	155.21 kg / 0.52%
	Vertical Tail	430.97 kg / 1.43%
	Fuselage	4905.76 kg / 16.32%
	Main Landing Gear	1026.86 kg / 3.42%
	Nose Landing Gear	237.33 kg / 0.79%
	Nacelle Group	571.58 kg / 1.90%
Summation	**Total**	30061 kg / 100%

Table 4.2 outlines critical dimensions and metrics of the CSA aircraft, including its overall dimensions, the CG locations for various load configurations, as well as the MAC. Additionally, the table details the CG positions expressed as a percentage of the MAC (%MAC), calculated using the formula $\%MAC = \frac{CG-IMAC}{MAC} \times 100$. Here, a %MAC of 0% corresponds to the leading edge of the MAC, called IMAC, with measurements taken from the nose of the aircraft.

Table 4.2: *Key dimensions and metrics of the CSA aircraft.*

Description	Dimension	Comment
Overall Length	36.151[m]	From nose to most aft position on tail
Wing Span	29.995[m]	Wing tip to wing tip
IMAC	19.2085[m]	Leading edge mean aerodynamic chord from nose
MAC	2.390 [m]	Mean aerodynamic chord
CG full tank and pax	19.806[m]	Measured from nose
CG empty tank and pax	19.559[m]	Measured from nose
CG full tank no pax	20.642[m]	Measured from nose
CG empty tank no pax	20.324[m]	Measured from nose
%MAC full tank and pax	25.00%	CG position as % MAC
%MAC empty tank and pax	14.66%	CG position as % MAC
%MAC full tank no pax	59.92%	CG position as % MAC
%MAC empty tank no pax	46.63%	CG position as % MAC

The full version of the statistically weight breakdown from using Raymer [6] can be found in appendix B. A drawing of the aircraft showing the size of the aircraft can be found in appendix C. In figure 4.5 the overall configuration of the CSA can be seen, with the primary tank and trim tank showcased.

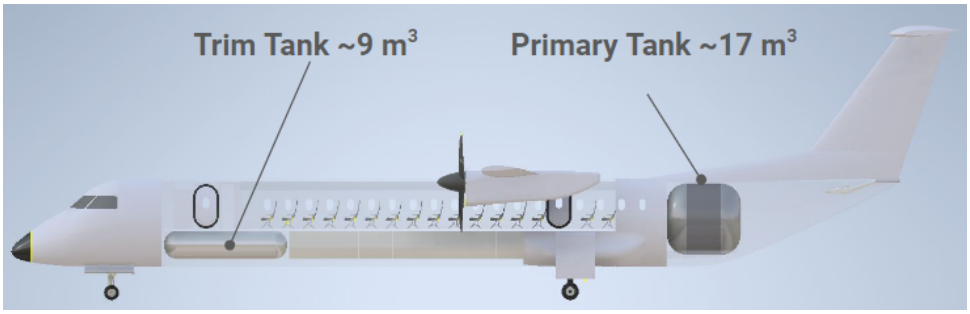


Figure 4.5: *Case study aircraft overall layout, showing the position of primary tank and trim tank.*

4.4.4 CSA CG-Envelope Limit Approximations

Due to the lack of a comprehensive weight envelope for the CSA and the time constraints associated with the master thesis project, it is necessary to approximate the CG envelope limits rather than conducting a detailed structural analysis of the landing gear combined with stability and control analysis during flight, which would yield more precise limits.

Initially, limits for the maximum allowable weights on the main landing gear (MLG) and nose landing gear (NLG) were established using data for the Dash 8 Q-400 from Bombardier [46]. Since the trim tank in the CSA adds additional weight to the front, the NLG weight limit was arbitrarily increased from the Q400's 2929kg to 5300kg. Correspondingly, the landing gear's weight was adjusted proportionally. Conversely, the MLG

limit was reduced from 13682kg to 12500kg due to the redistribution of weight to the NLG.

A minimum weight of 2000kg was set for the NLG to ensure adequate maneuverability on the ground and during takeoff, avoiding exceeding 90% of the load on the MLG.

Additionally, from the aircraft design course, it is established that the MLG must support 80-90% of the total weight, with the NLG bearing the remaining 10-20%. A safety margin of 5% was added to the upper limit of the MLG to further constrain the envelope.

Figure 4.6 illustrates the envelope with points labeled A to G and includes two vertical lines indicating the 90% weight limit for the MLG (to the right) and the 20% weight limit for the NLG (to the left). The horizontal axis represents the %MAC, extending from the forward (left) to the aft (right) positions. The vertical axis shows the weight ranging from 17500kg to 30000kg.

The limits labeled A and B in the figure are constrained by the maximum weight capacity of the NLG. Exceeding these limits would surpass the set NLG limit of 5300kg.

Limits C and D are determined by the maximum weight capacity of the MLG. However, limit D is constrained at 73%MAC, which exceeds the safety margin for the MLG, thus it is set to the 5% safety margin. The reduction in weight down to limit E is governed by the minimum weight requirement for the NLG. Between limits E and F, the minimum NLG weight becomes the limiting factor, indicating that as weight decreases, the CG must move forward if it is on the rightmost edge of the envelope.

Limit G is defined by the same %MAC as limit A and completes the envelope with a vertical line connecting limits A and G.

When it comes to which weights that was chosen for the envelope, the upper limit is set by the MTOW calculated in the aircraft design course. The rest is set by using the four different weights from having fuel, no-fuel, combined with PAX and no-PAX.

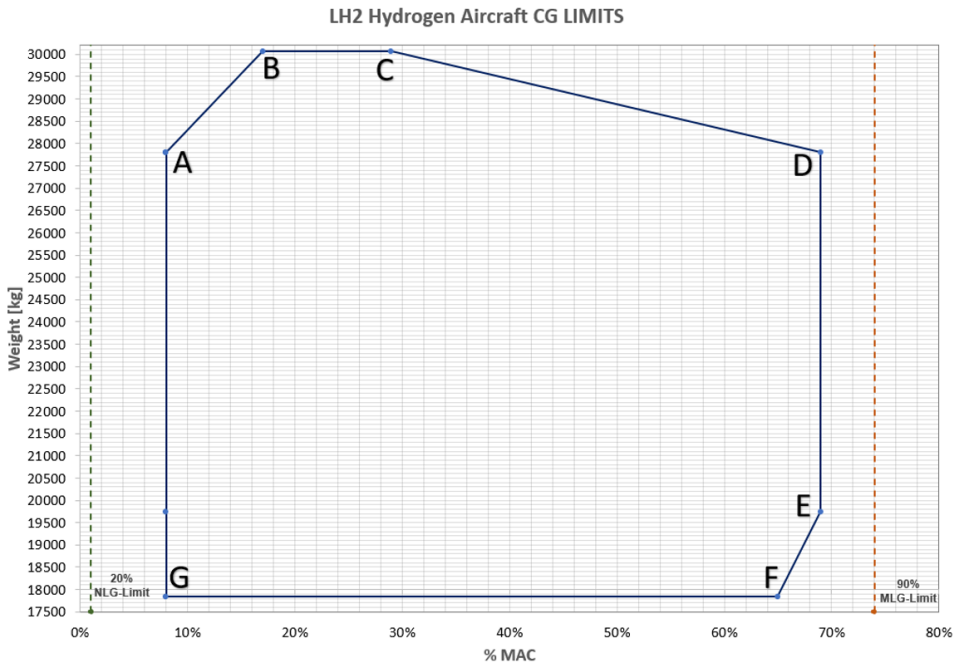


Figure 4.6: *Case study aircraft CG envelope limit approximation.*

5

Developed Method

In this section, the methods developed will be shown and explained. These methods include the fuel tank tool, which manages the mass distribution of fuel within the fuel tank. Additionally, the mass properties sectioning tool is described, which gathers mass properties from large assemblies. Finally, the weight control tool is shown, which manages all mass and balance data and presents the CG-envelope.

5.1 Fuel Tank Tool

The tool developed utilizes Visual Basic, interacting with Catia via its API. It features an easy-to-understand user interface (UI) enabling quick and straightforward analysis through Catia automation.

Figure 5.1 below, illustrates the UI with which users will interact. The user workflow is detailed as follows. Initially, the user opens the part representing the fuel volume in Catia. Then, the user opens the tool and clicks on "Select part." If the selection in Catia is correct, the part name will display in the tool. Otherwise, an error message will appear, offering an opportunity to reselect the part. The next step involves choosing the parameters for the tool to section the part, allowing for varying accuracy depending on the case. Sectioning in the "y-" and "x-" directions typically depends on the tank's design, often based on the tank ribs. In the z-direction, it determines the number of data points for volume measurement. More sections lead to higher accuracy. However, a delta level of $1cm$ is usually sufficient for detailed volume/CG data, especially when used with interpolation. Once sectioning is set, the user specifies the tank's pitch/roll to simulate different operating conditions. Following this, the user can calculate and display the bounding box. The tool then presents an initial sectioning based on the tank's minimum and maximum dimensions, with evenly set delta distances between sections. This is done in the table presented in the UI, users can then adjust these divisions as needed. Subsequently the sectioning is visualized through the creation of semitransparent planes, a important step in verifying the tool's accuracy. The next step is then for the user to start the tools' CG/inertia calculations.

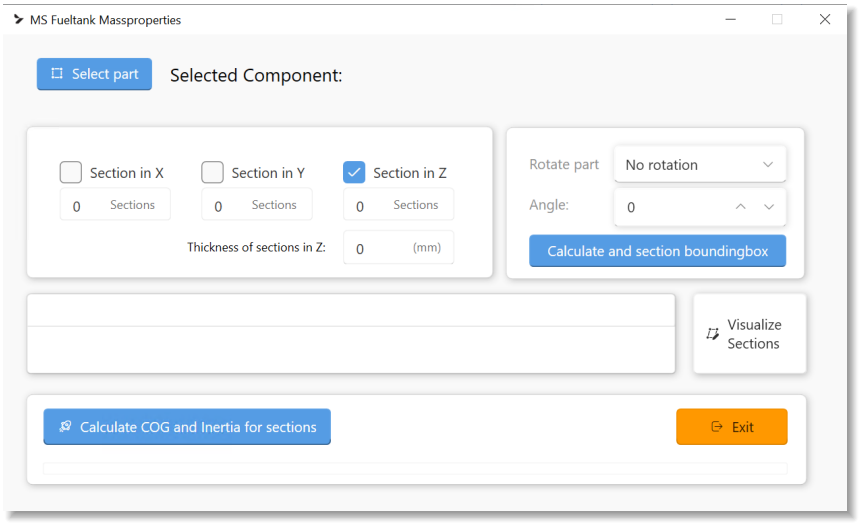


Figure 5.1: *Fuel Tank Tool GUI*

The illustration in figure 5.2 below depicts the tank at a 10-degree pitch. To minimize part alterations, the tool calculates new points for pitch representation using transformation matrices. This method allows the tank to remain unchanged while simulating pitch/roll conditions. The limitation on part modifications ensures easy reversion in case of errors during the tool calculations.

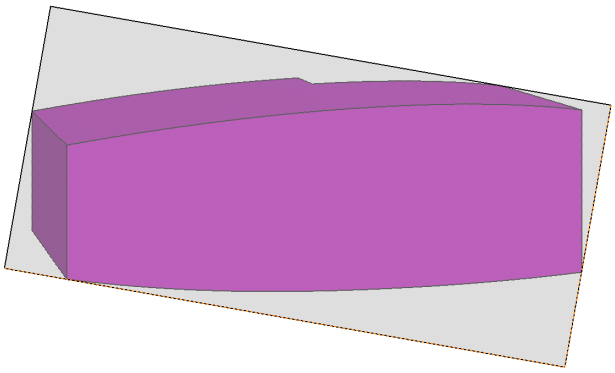


Figure 5.2: *Fuel Tank Tool bounding box for 10 degree pitch*

After the initial visualization of the bounding box, the sections can be visualized. These sections as can be seen is based on a no pitch/roll condition, but this is delibarate since it was determined that it is easier to see how the tool has sectioned the part this way. The image below represents an equal delta distance between the sections but this can be modified as the user chooses. A practical application of this feature is in the case of a collector tank (CT). As mentioned earlier, in aircraft the CT must remain filled as long as there is fuel in the main tank, necessitating a logical approach to fuel consumption. By strategically sectioning the tank to include a section representing the CT, accurate calculations can be made that incorporate this requirement.

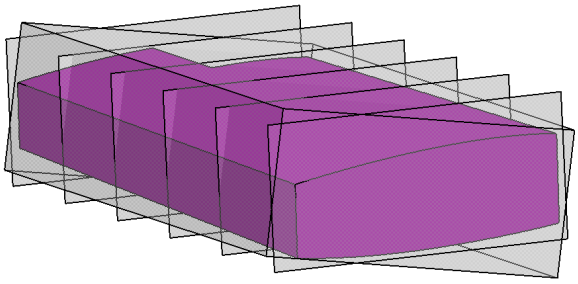


Figure 5.3: *Fuel Tank Tool sectioning into 5 segments for 10 degree pitch*

The way that the tool calculates the CG/inertia is done by utilizing sectioning planes along with Catia's built-in measurement tool. As illustrated below, for this specific case, the tool sections the tank in the "y" and "z" directions. The sectioning in the "z" direction correlates to a particular fuel volume, such as half the tank's capacity, as demonstrated in the image.

Figure 5.4: *Fuel Tank Tool section of fuel tank when running the tool for 10 degree pitch*

5.2 Mass Properties Sectioning Tool

The sectioning tool, designed with influences from the fuel tool, helps in calculating mass properties of large assemblies, efficiently summarizing them. Unlike the fuel tool which deals with individual parts, the sectioning tool processes assemblies ranging from just a few parts to thousands of parts where multiple assemblies are present. This necessitates the tool to either operate reliably for extended periods or execute quickly for frequent reruns where the user can fix potential problems arising. Given the complexity that comes from large assemblies, skins, and design methods, ensuring reliability presents challenges. For example, the tool is restricted to solids since modifying solids and surfaces differs significantly, complicating the method of sectioning.

Design approaches, whether through the use "mainbodies", geometric sets or other body types, introduce varied combinations that complicate automation a lot since it is difficult to determine the approach automatically. Thus, aiming for speed facilitates identifying and rectifying design inconsistencies through a trial and error method. The tool

provides feedback on errors, allowing users to adjust problematic designs for consistency or remove them and address them separately.

5.2.1 Program Workflow

The UI of the program is designed to efficiently guide users from the initial step of loading an assembly to the final computation of mass properties. The workflow is organized as follows:

1. **Assembly Loading:** The process begins with the user loading the assembly to be analyzed. The UI presupposes that the assembly is already open; if not, the user is prompted to open the assembly file.
2. **Linking to Catia API:** Upon successful loading, the program establishes a connection with the Catia API in order to access the assembly data required for the subsequent analysis.
3. **Initiating Bounding Box Calculations:** Users then start the bounding box calculations. This involves indexing each part within the assembly for efficient future access. During indexing, the tool records the minimum and maximum values for each part along with their indexes, aiding in the assembly's quicker sectioning later.
4. **Global Bounding Box Display:** The tool also calculates and displays a global bounding box for the entire assembly, assisting in the verification of the tool's accuracy and providing a comprehensive view of the assembly's dimensions.
5. **Specifying Assembly Type for Sectioning:** Users specify the assembly type (e.g., fuselage, wing), determining the sectioning direction (X-direction or Y-direction). This decision is based on the assembly's structural features, such as engines or fuel tanks, which influence the sectioning approach.
6. **Adjusting and Distributing Sections:** The UI facilitates the adjustment and distribution of sections according to the assembly's specifics. This step includes selecting the number of sections and their positioning to optimally encompass particular features or components. Users can reassess their section placements to ensure precision.
7. **Computing Mass Properties:** The final step involves computing the mass properties for the assembly. An Excel workbook is generated, detailing the mass properties of each section's parts. Give the preprocessing of each part this means that the parts easily can be assigned to different sections with minimum number of actions and only in the cases where part extremums goes across boundries of sections will they have to be sectioned.

This workflow is easy to follow with a left to right flow making it easier to understand, even for first time users. By also providing some feedback and offering flexible control over the sectioning process, the tool allows the user to adjust for different assemblies and conditions to be able to perform reliable calculations in shorter time.

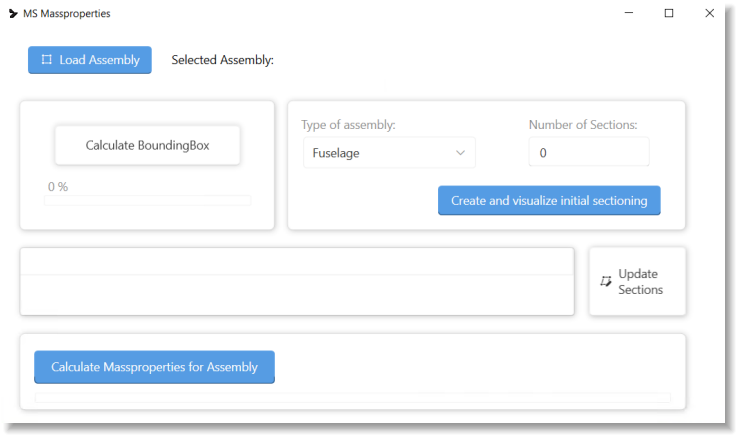


Figure 5.5: *Mass properties tool GUI*

The process of sectioning a part is largely based on previous experience of the mass properties engineer. The main idea is to center large masses such as engines or landing gears in own sections. As can be seen below the choice of sectioning has been done such that the cockpit/NLG, entry/trimtank, passenger, engines/wing, exit, main tank and HT/VT is placed in own sections. By doing this the mass can be distributed well for the loads to be calculated. Noteworthy here however is that when running a more extensive analysis the fuselage, wing, HT and VT would be run in separate assemblies to get a better distribution in all directions but due to the limitations of the case study the assembly will be run as a whole.

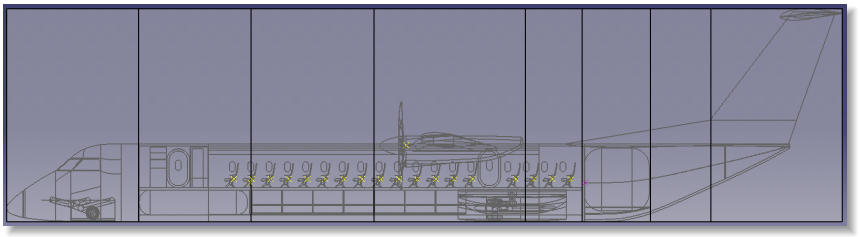


Figure 5.6: *Choice of sectioning*

5.2.2 Exported Data Structure

For each part and section the data will be exported to an Excel workbook in the following structure (See fig. 5.7).

- **1:** Since loads calculation is dependant on point masses spread out over the aircraft there is a need for generating sections that parts can be split and assigned to. These are presented as different worksheets.
- **2:** For each part in a section, the mass, volume and density is presented
- **3:** The CG is also calculated for each part, this CG is then in reference to the assembly axis system which usually is positioned on the aircraft nose cone.
- **4 & 5:** Then the MOI and POI is calculated, this is done in reference to the part CG position.
- **6:** Not seen in the image below is also a summarized worksheet where all the data from the sections are compiled. The mass and volumes can just be added but for the CG, MOI and POI there is a need for shifting which is done with equations 3.1, 3.2 and 3.3.

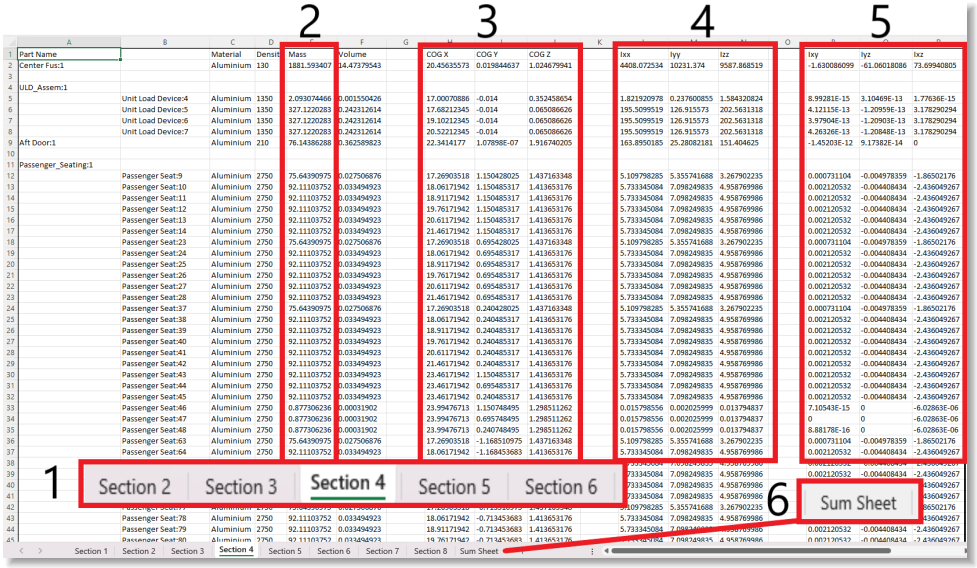


Figure 5.7: *Example data obtained from mass properties tool*

In the current CAD setup, the aircraft parts are not distinct, which may make future automation of the data challenging. Typically, parts are grouped into categories such as structural, interior, and systems. Adopting a naming system like ATA-100 or JASC/ATA could simplify the categorization of parts and assign changes to specific departments more effectively. During the initial design phase of an aircraft, tracking weight changes is crucial, both for individual parts and whole assemblies. Multiple changes might occur in one design iteration, making it difficult to identify where these changes are. A clear naming system for parts would make it easier to track these changes and to consult the appropriate department about them. These changes can then be much easier to track through the weight journal.

5.3 Weight Control Tool

The Weight Control Tool (WCT) is important for summarizing weight and balance data into a comprehensible Excel workbook. This tool, programmed with VBA, simplifies data management significantly and allows for easy visualization of the CG envelope of the aircraft.

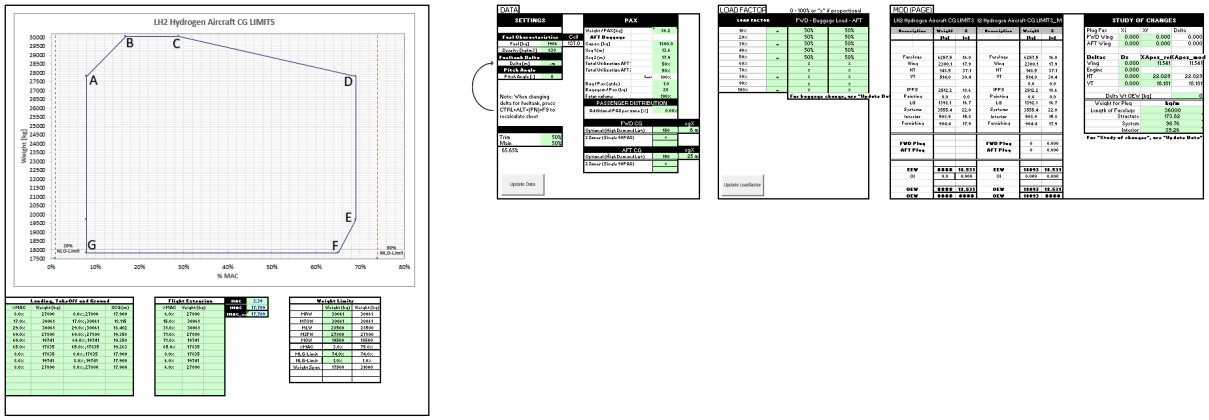


Figure 5.8: *WCT overview with CSA example data*

5.3.1 Data Management

The initial step of setting up the WCT is to obtain and export the data for the parts and the fuel. This is done in separate sheets. The part data is imported into different sheets depending on which category they belong to. For example, the mainbox of the wing will be added to the wing section while the elevators will be added to the HT. An example of this data can be seen in figure 6.5. The importation of the fuel data is easier since the data generated in the fuel tool can be imported using a VBA script provided it is in the correct directory. After importing the part and fuel data, the data for the passenger logic needs to be preprocessed. This is done by copying the position of the passenger seating to the PAX load sheet. After this the seating needs to be defined by two metrics. Firstly if it is a window seat or an aisle seat, and then which zone the seat is connected to. After this, the user can press a button in the sheet that runs a VBA script, setting up the passenger load logic automatically.

5.3.2 Logging Changes - Weight Journal

Keeping track of changes in aircraft mass properties is essential for managing the overall weight during the development process. Each subsystem of the aircraft has weight targets based on statistical estimates. To effectively monitor and understand the impact of individual changes on the total weight and compare it to these targets, changes are recorded in a weight journal.

The weight journal, when used together with the WCT, simplifies the comparison of the current aircraft weight against previous versions. This method helps identify if the weight is on track or if adjustments are needed.

Furthermore, using a structured naming scheme for parts, such as ATA100 or JASC/ATA, improves the organization of the weight journal. This makes it easier to pinpoint where changes are happening and link them to specific subsystems or categories.

This approach helps maintain accurate and organized records, making the process of managing aircraft weight more straightforward and reliable.

5.3.3 WCT Data

In figure 5.9 below, the part for which the data settings are controlled can be seen. Starting of with the selection of fuel quantity in *kg* together with the fuel density. This is to determine for which fuel volume the CG-envelope should be calculated. The use of different density also makes it possible to account for different fuels and pressures. When the desired fuel and density are selected, the fuel tank data is updated such that 25 fuel tank data points with equal distance is displayed from zero to the maximum fuel. In figure 5.10 an example of the data selected for the ES-30 fuel tank is displayed, showing the data which the tool would show for a fuel weight of $1000kg$. Then numerous other changes can be made to see the difference of using different average PAX and baggage weight and additional PAX per zone to mention some. In the cases where there are more than one CGx for the fuel tank the user can also specify how much of the fuel that should be drained from each tank.

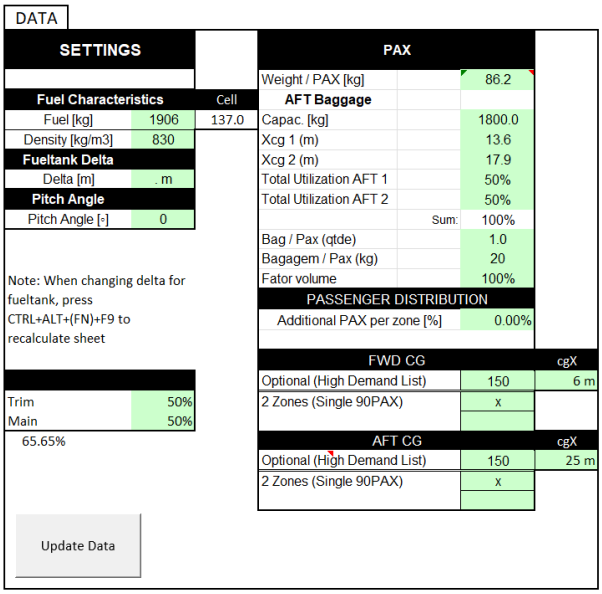


Figure 5.9: *WCT data control settings*

kg of Fuel	Xcg_0	Xcg_1	Xcg_2	Xcg_3	Xcg_4	Xcg_5	Xcg_6	Xcg_7	Xcg_8	Xcg_9	Xcg_10
0	12.967	13.054	13.134	13.209	13.278	13.331	13.371	13.403	13.426	13.442	13.453
50	12.880	12.902	12.923	12.945	12.967	12.989	13.011	13.033	13.053	13.072	13.089
100	12.870	12.881	12.892	12.903	12.914	12.925	12.937	12.948	12.959	12.970	12.981
150	12.867	12.874	12.882	12.888	12.894	12.900	12.904	12.909	12.913	12.916	12.920
200	12.846	12.850	12.853	12.856	12.858	12.860	12.861	12.862	12.863	12.864	12.865
250	12.865	12.879	12.892	12.903	12.913	12.922	12.929	12.935	12.939	12.943	12.947
300	12.876	12.892	12.908	12.923	12.936	12.947	12.957	12.965	12.973	12.979	12.985
350	12.883	12.898	12.913	12.928	12.941	12.954	12.966	12.977	12.986	12.994	13.002
400	12.887	12.901	12.914	12.928	12.941	12.954	12.966	12.978	12.989	12.998	13.007
450	12.891	12.903	12.915	12.927	12.939	12.951	12.963	12.974	12.986	12.996	13.006
500	12.893	12.904	12.915	12.926	12.937	12.948	12.958	12.969	12.980	12.991	13.001
550	12.896	12.905	12.915	12.925	12.935	12.945	12.955	12.965	12.975	12.985	12.995
600	12.897	12.906	12.915	12.924	12.934	12.943	12.952	12.961	12.970	12.979	12.988
650	12.899	12.907	12.916	12.924	12.932	12.941	12.949	12.957	12.966	12.974	12.983
700	12.900	12.908	12.916	12.924	12.931	12.939	12.947	12.955	12.962	12.970	12.978
750	12.902	12.909	12.916	12.923	12.930	12.938	12.945	12.952	12.959	12.967	12.974
800	12.903	12.909	12.916	12.923	12.930	12.936	12.943	12.950	12.957	12.963	12.969
850	12.903	12.910	12.916	12.922	12.929	12.935	12.941	12.947	12.953	12.959	12.964
900	12.904	12.910	12.916	12.922	12.928	12.934	12.939	12.944	12.949	12.953	12.957
950	12.905	12.911	12.916	12.921	12.926	12.931	12.935	12.939	12.942	12.945	12.948
1000	12.905	12.910	12.915	12.919	12.923	12.926	12.929	12.932	12.935	12.937	12.939

Figure 5.10: *WCT fuel tank data, ranging from 0 to 1000kg for pitch angle 0 to 10.*

5.3.4 WCT Load Factor

To test different PAX and baggage loading of the aircraft the WCT Load Factor management can be used. Each of the different load factors indicate a percentage of full PAX and baggage that should be showed in the CG-envelope. Together with the possibility to modify how much of the baggage that should be filled. In Figure 5.11 the case for 10% indicates that there are 10 percent of the PAX, together with a 50% filled forward baggage capacity and 50% of the aft baggage capacity, all in all 10% PAX with a total of 50% baggage. This is useful for when an aircraft is not carrying the full capacity of passengers the baggage compartment can still be used to transport goods. For the case of 80% load factor the baggage compartment are marked with an "x" this means that 80% of the forward and aft baggage compartment will be filled, as if there are no extra goods transported and all passengers carry the maximum $20kg$ luggage, which could be changed in as shown previously in the WCT data.

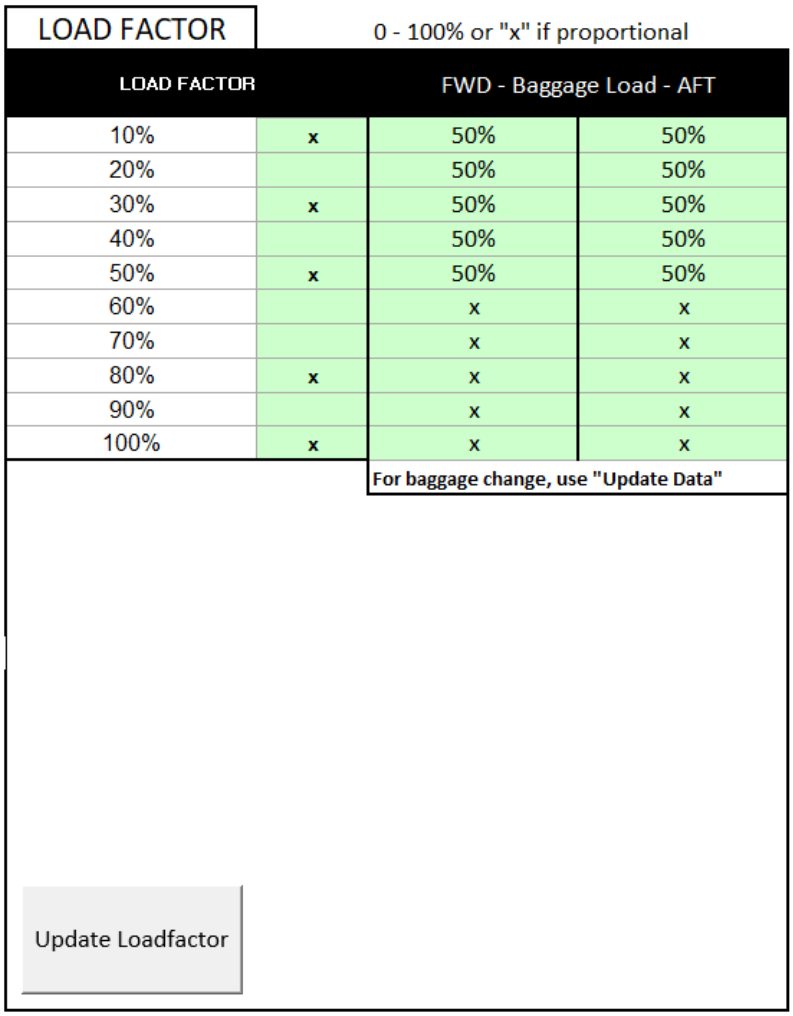


Figure 5.11: *WCT Load Factor with CSA example data*

5.3.5 WCT Modification Page

The modification page is used to test different aircraft configurations quickly. By allowing to shift parts such as the wing, tanks or adding plugs it is easy to see the impact of different changes. The modification page works such that parts that has been categories similar to the delta introduced it will shift that part. For example, a shift in the wing will also shift the engines. An example of how this can be used in the conceptual phase can be found in the case study below.

The modification page within the WCT serves as a testing ground for various aircraft configurations. By allowing shifts in components like the wing, tanks, or by adding weight plugs, the page demonstrates the potential impact of changes in real-time. It operates by applying part-specific changes within designated categories. For instance, adjusting the wing's position will proportionally adjust the weight and balance attributes linked to the wing category.

A practical application of this feature is shown in the case study below where the wing is shifted forward to reach a target CG position with a subsequent change in the CAD to compare the results.

MOD (PLUG)

LH2 Hydrogen Aircraft V4			LH2 Hydrogen Aircraft V4_Mod		
Description	Weight (kg)	X (m)	Description	Weight (kg)	X (m)
Fuselage	6257.5	16.0	Fuselage	6257.5	16.0
Wing	2300.1	17.9	Wing	2300.1	17.9
HT	148.5	37.1	HT	148.5	37.1
VT	514.0	30.4	VT	514.0	30.4
				0.0	0.0
IPPS	2512.2	18.6	IPPS	2512.2	18.6
Painting	0.0	0.0	Painting	0.0	0.0
LG	1392.1	16.7	LG	1392.1	16.7
Systems	3555.4	22.0	Systems	3555.4	22.0
Interior	508.9	15.8	Interior	508.9	15.8
Furnishing	904.4	17.9	Furnishing	904.4	17.9
FWD Plug			**FWD Plug**	0	0.000
AFT Plug			**AFT Plug**	0	0.000
EEW	**18093.1**	**18.531**	**EEW**	**18093**	**18.531**
OI	0.0	0.000	OI	0.000	0.000
OEW	**18093.1**	**18.531**	**OEW**	**18093**	**18.531**
OEW	**18093.1**	**34.40%**	**OEW**	**18093**	**34.40%**

STUDY OF CHANGES			
Plug Fus	Xi	Xf	Delta
FWD Wing	0.000	0.000	0.000
AFT Wing	0.000	0.000	0.000
Deltas	**Dx**	**XApex_ref**	**XApex_mod**
Wing	0.000	11.541	11.541
Engine	0.000		
HT	0.000	22.029	22.029
VT	0.000	18.181	18.181
Delta Wt OEW [kg]			0
Weight for Plug		**kg/m**	
Length of Fuselage		36000	
Structure		173.82	
System		98.76	
Interior		39.26	

For "Study of changes", use "Update Data"

Fueltank Delta	
Delta Primary	. m
Delta Trim	. m

Figure 5.12: *WCT Modification Page with CSA example data*

6
Results and Discussion

This section presents an analysis of the findings derived from the application of the methodologies described in the previous sections. There will be a focus on the assessment of the ES-30 electric aircraft's weight and balance, emphasizing the impact of fuel volume and pitch angles on its performance and CG-envelope. The results are categorized into several key areas: development of a CG envelope under operational flight conditions, evaluation of fuel volume and pitch angle impacts, and the effectiveness of the automated tool in dynamic CG calculations.

6.1 Evaluation and Error Calculation of Fuel Tank Tool Mass Properties Method

In this section an evaluation of the method used to calculate the total CG's, MOI and POI will be presented, that is, equations 3.1, 3.2 and 3.3 which has been used in the post-processing of the data gathered, both for the fuel tank tool and when gathering individual contributions from various parts of the aircraft and determining the overall CG of the aircraft.

In Tables 6.1, 6.2 and 6.3 the different mass properties gathered from the fuel tank tool together with the values gathered manually from Catia and the error percentage calculated from eq 6.1 is presented.

$$\left(\frac{Catia_{Value} - Tool_{Value}}{Catia_{Value}} \cdot 100\right) \tag{6.1}$$

As can be noted from the tables on the next page, the effect of sectioning and summarizing up the values for both, volume fraction ("VolFrac", which is the same as the mass when multiplied with a density), CG, MOI and POI has a negligible effect. The conclusion that can be drawn from this is that if the measuring tool in Catia gives a correct result for the model, using sections to simulate both compartments and fuel level will not affect the accuracy of the results. A similar study with 100 sections in both directions yielded similar error confirming the accuracy of the method.

The scarcity of research specifically addressing the accuracy of the measuring tool in Catia V5 may initially seem concerning. However, the widespread use of Catia V5 in various industries, from automotive to aerospace engineering, alongside the general trust in its computational reliability, speaks volumes. The mathematical foundations for calculating mass properties, such as CG, MOI, and POI, are well-established and straightforward. Given this context, the accuracy of Catia V5's measuring tools is assumed to be accurate. The main discrepancy that might be found between the real world case and the digital analysis is the fidelity of the model. However, there's quite often a natural

correlation where the required accuracy of analysis aligns well with the available fidelity of the models based on where in the development phase the product is.

Table 6.1: *Center of Gravity Error from sectioning for 5 segments and 55 slices of fuel level*

Pitch	Source	VolFrac (%)	CGx (m)	Cgy (m)	CGz (m)
0°	Tool	50,4	12,94055	-5,51324	4,65750
0°	Catia	50,4	12,94055	-5,51324	4,65750
0°	Error (%)	0,0	3,83E-07	3,31E-07	5,07E-07
4°	Tool	51,4	12,98419	-5,53999	4,66116
4°	Catia	51,4	12,98419	-5,53999	4,66116
4°	Error (%)	0,0	6,47E-07	7,08E-07	5,19E-07
8°	Tool	49,3	13,03155	-5,56107	4,66161
8°	Catia	49,3	13,03155	-5,56107	4,66161
8°	Error (%)	0,0	9,01E-07	9,70E-07	5,08E-07

Table 6.2: *Moment of Inertia Error from sectioning for 5 segments and 55 slices of fuel level*

Pitch	Source	VolFrac (%)	Ixx (kg*m2)	Iyy (kg*m2)	Izz (kg*m2)
0°	Tool	50,4	447,06659	91,55145	532,74970
0°	Catia	50,4	447,06675	91,55140	532,74982
0°	Error (%)	-	3,74E-05	-4,89E-05	2,31E-05
4°	Tool	51,4	469,72326	91,00544	554,07289
4°	Catia	51,4	469,72342	91,00538	554,07300
4°	Error (%)	-	3,38E-05	-5,92E-05	1,91E-05
8°	Tool	49,3	462,48948	81,72004	536,94949
8°	Catia	49,3	462,48964	81,71998	536,94958
8°	Error (%)	-	3,28E-05	-7,44E-05	1,71E-05

Table 6.3: *Product of Inertia Error from sectioning for 5 segments and 55 slices of fuel level*

Pitch	Source	VolFrac (%)	Ixy (kg*m2)	Ixz (kg*m2)	Iyz (kg*m2)
0°	Tool	50,4	43,96040	-0,34133	9,11089
0°	Catia	50,4	43,96043	-0,34133	9,11090
0°	Error (%)	-	6,82E-05	1,49E-04	1,21E-04
4°	Tool	51,4	46,14966	-3,62180	11,21890
4°	Catia	51,4	46,14969	-3,62180	11,21891
4°	Error (%)	-	5,84E-05	-9,63E-06	9,02E-05
8°	Tool	49,3	43,85856	-6,22244	12,58473
8°	Catia	49,3	43,85858	-6,22244	12,58474
8°	Error (%)	-	4,87E-05	-1,19E-05	7,76E-05

6.2 Fuel Tank Tool Post-Process Results

Using the data from the fuel tool, further analysis can be conducted to show how the tank's CG moves. It is important to get the CG right for the aircraft's stability and safety, not only when designing but also operating. The two different rows depicts the CG shift under refueling (turbo-generators not active) and during operation (turbo-generators active). In contrast to a normal aircraft, the ES-30 does not switch on the turbo-generators directly when starting operation. The difference this makes is that the CT does not stay full during all operating conditions but may be level with the rest of the tank during fully electric flight. This can result in a rather large shift when turning the generators on during flight as can be seen when comparing row 1 (Figure 6.1 and Figure 6.2) and row 2 (Figure 6.3 and Figure 6.4) below.

For both active and inactive turbo-generators, increasing the pitch angle moves the CG aft of the aircraft. However, with active turbo-generators, this rearward CG shift is more restrained. Notably, when the fuel volume equals that in the CT, the active generator's CG remains constant, regardless of pitch angle. This is where toggling the generator can induce a significant CG shift in the fuel tank, though its effect on overall CG is rather small due to the lower fuel weight compared to cases of near-full tank scenarios. This aspect will be elaborated upon in the total CG-envelope analysis.

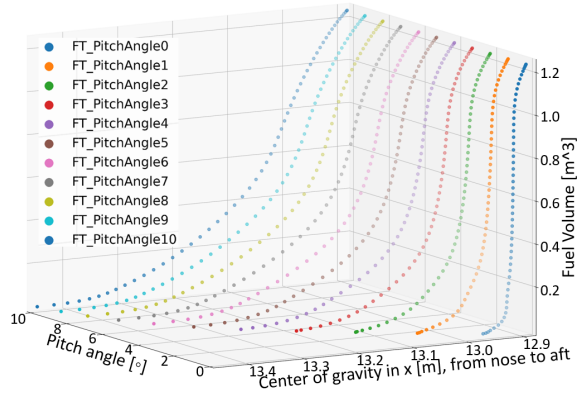


Figure 6.1: *No turbo-generators active, 3d view*

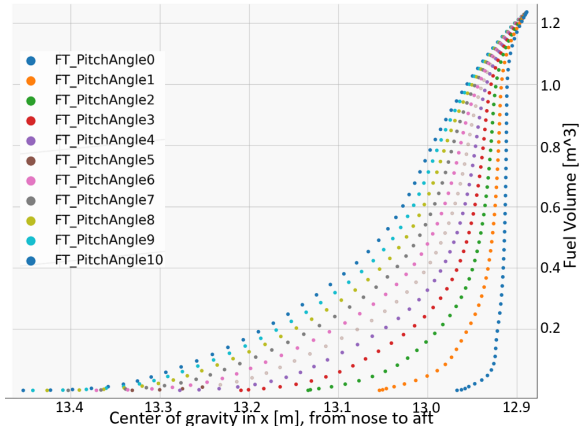


Figure 6.2: *No turbo-generators active, side view*

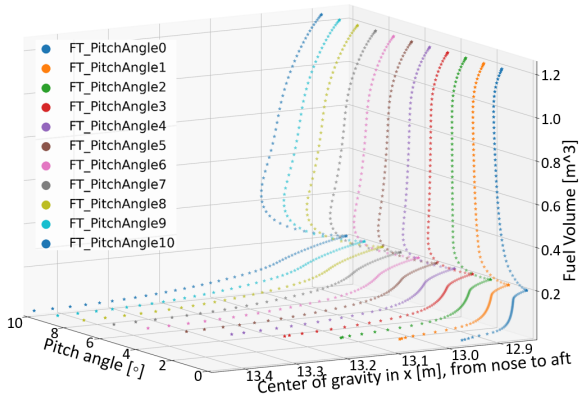


Figure 6.3: *Active turbo-generators, 3d view*

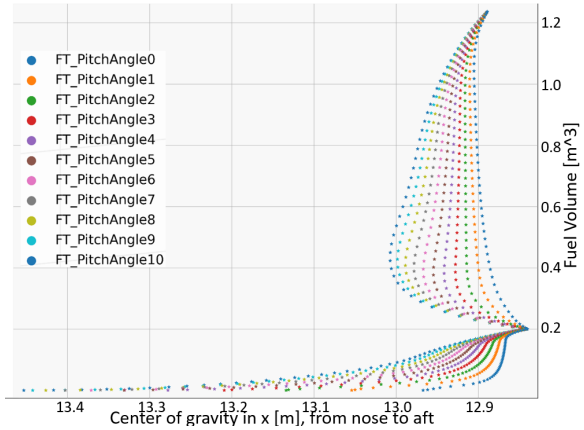
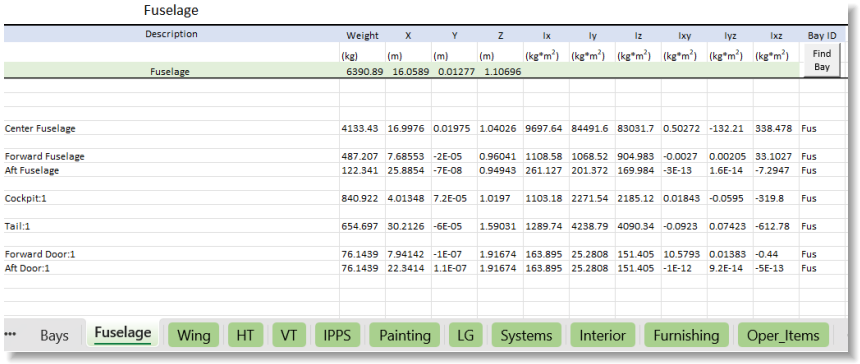


Figure 6.4: *Active turbo-generators, side view*

6.3 Case Study

For the case study as mentioned previously, a 90 PAX hydrogen aircraft was considered. The design was taken from previous work in course MMS230 which laid the foundation for the aircraft design. In the case study the tools developed was applied and modified to fit this new aircraft. This gave a unique opportunity to test the adaptability of this process in other cases than the one it was initially designed for. The first analysis that was executed was the obtainment of aircraft part initial weights through the sectioning tool. Since the sectioning tool can be used both for sectioning the aircraft and assigning parts or just obtaining individual part data this was a good fit for this step. By running the tool, each individual part mass properties could be obtained quickly and then assigned in the WCT (See fig. 6.5). This step was done for all categories of the aircraft. When completed the data could be validated by checking the summed weight with the total weight from Catia.



Fuselage

Description	Weight	X	Y	Z	Ix	Iy	Iz	Ixy	Iyz	Ixz	Bay ID
	(kg)	(m)	(m)	(m)	(kg*m²)	(kg*m²)	(kg*m²)	(kg*m²)	(kg*m²)	(kg*m²)	Find Bay
Fuselage	6390.89	16.0589	0.01277	1.10696							
Center Fuselage	4133.43	16.9976	0.01975	1.04026	9697.64	84491.6	83031.7	0.50272	-132.21	338.478	Fus
Forward Fuselage	487.207	7.68553	-2E-05	0.96041	1108.58	1068.52	904.983	-0.0027	0.00205	33.1027	Fus
Aft Fuselage	122.341	25.8854	-7E-08	0.94943	261.127	201.372	169.984	-3E-13	1.6E-14	-7.2947	Fus
Cockpit:1	840.922	4.01348	7.2E-05	1.0197	1103.18	2271.54	2185.12	0.01843	-0.0595	-319.8	Fus
Tail:1	654.697	30.2126	-6E-05	1.59031	1289.74	4238.79	4090.34	-0.0923	0.07423	-612.78	Fus
Forward Door:1	76.1439	7.94142	-1E-07	1.91674	163.895	25.2808	151.405	10.5793	0.01383	-0.44	Fus
Aft Door:1	76.1439	22.3414	1.1E-07	1.91674	163.895	25.2808	151.405	-1E-12	9.2E-14	-5E-13	Fus

Bays | Fuselage | Wing | HT | VT | IPPS | Painting | LG | Systems | Interior | Furnishing | Oper_Items

Figure 6.5: *Example of fuselage-data applied in WCT tool. In the list each part containing to fuselage is listed with their respective weight, CG, MOI and POI.*

It is essential to note that in this step, the calculated weight of the parts was defined to exclude passengers, baggage, and fuel, as these components are incorporated separately into the WCT. The next step involved obtaining data for the fuel tanks using the fuel tool introduced earlier. A notable difference in this case is the CSA's utilization of both a primary and a trim tank, meaning that the analysis had to be run once for each tank.

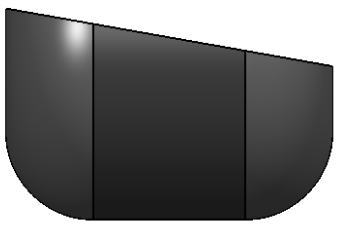


Figure 6.6: *Primary Tank Cutaway View*

Figure 6.7: *Trim Tank Cutaway View*

In this phase, the usage of the WCT required some modification since it was designed to manage a single wing tank, yet the CSA design included both a primary and a trim

tank. To accommodate this dual-tank system, a new feature was introduced allowing the user to specify the fuel volume in each tank by percentage. This solution not only resolved the discrepancy but also provided insights into the effects of fuel distribution on the CG dynamics in a dual-tank configuration. After implementing the fuel data, the passenger and baggage logic had to be edited. This is because the number of passengers and their position in the aircraft was different. By assigning PAX numbering (PAX 1A, 2A etc) and their respective CG position the previously implemented macro made the logic easy to obtain. The logic is based on the passengers being split into two zones where the first zone contains 60 of the passengers and the second zone contains the 30 remaining passengers. By assuming passengers always choose a window position instead of a corridor position the CG could be calculated for different load factors. Next the baggage had to be added, in contrast to the ES-30, the CSA makes use of 6 unit load devices in the belly of the aircraft. However making use of the initial two baggage zones seemed reasonable enough. It was assumed that each passenger will check in one $20kg$ baggage giving the aircraft a total of 1,8 tonnes of baggage fully loaded.

6.3.1 CSA - Fuel Tank CGx Shift During Flight

When conducting the analysis of the fuel tanks, a CG envelope was obtained for each specific tank. The analysis revealed a significant spread in the x-direction, ranging from 45 to 180 centimeters. Such a variation is considerable over an aircraft's operational cycle. However, the fuel's weight is relatively minor compared to the aircraft's total weight, which implies that the overall impact of the CG shift will be rather small. Below, the CG shifts are detailed for the primary and trim tanks of the hydrogen aircraft. It is observed that as the pitch increases and the fuel level approaches zero, the shift becomes more pronounced.

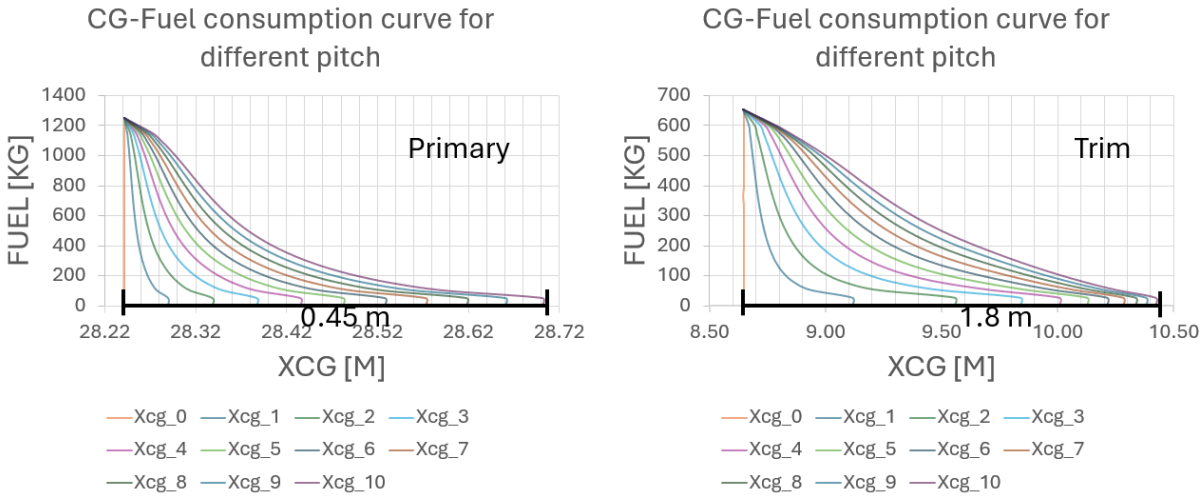


Figure 6.8: *The figure displays data for hydrogen tanks, including the primary and trim tanks. The X-axis represents the CG position in X-direction, while the Y-axis shows the fuel mass. Different series in the graph represent the pitch of the tanks, ranging from 0 pitch on the left to 10 pitch on the right. The graph includes the maximum shift values: for the main tank, there is a maximum shift of 45 cm when the pitch changes from 0 to 10 degrees at nearly empty fuel conditions. For the trim tank, the maximum shift reaches 1.8 meters under similar conditions of nearly zero fuel and a pitch change from 0 to 10 degrees.*

Next the fuel data can be used in the WCT to understand it's overall effects. By overlaying the data for 0° and 10° it can be seen that the overall effect is minimal. Also in contrast to the fuel CG curves the maximum shift is seen at half fuel conditions, this since both the weight of the fuel and the CG shift of that weight is accounted for.

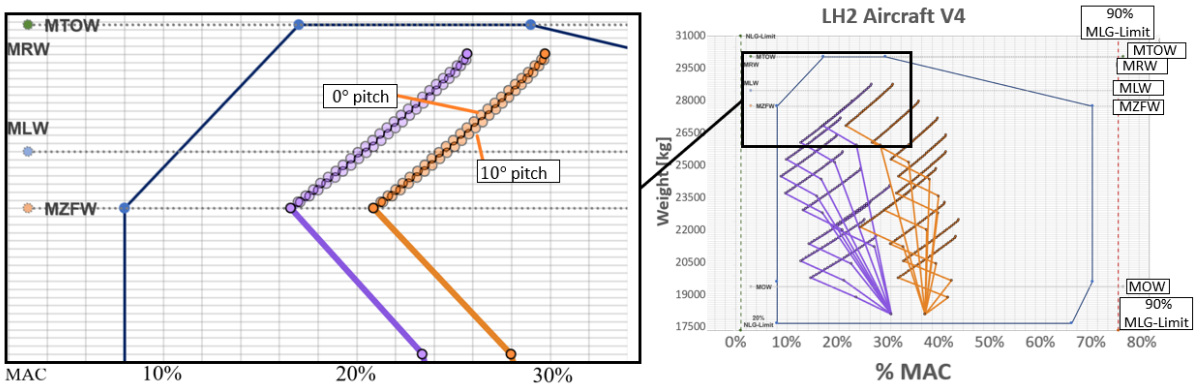


Figure 6.9: *This figure presents the total CG envelope for the aircraft, highlighting the shift in fuel position as a result of pitch changes from 0° to 10°. It visualizes how the CG shifts of the fuel impact the aircraft CG*

6.3.2 Results From CSA Original Design

6.3.2.1 CG Envelope - CSA original design

After all the data have been added to the WCT tool, parts, tank, PAX etc. the overall CG-envelope is calculated and displayed by simply pressing update on the CG-envelope sheet. The original design CG-envelope can be seen in figure 6.10, where each line showcases a different setup of number and positioning of PAX, luggage etc. The first point is the operating empty weight (OEW) then the OEW plus PAX added, then the baggage is added for the third point which is the zero fuel weight (ZFW). Finally the fuel is added which is made up by multiple points showing different fuel volumes. The difference between the blue lines and the orange lines is that within the tool the user can choose to have optional items, such as additional avionics, life vests and emergency raft for over water flights and galley items etc. In the two cases shown for optional items one has $150kg$ added to the forward CG (blue lines) and the other case (orange lines) has $150kg$ added to the aft CG.

As can be seen in the figure, the overall CG is way in front of the approximated CG-envelope eventhough the CG was calculated in OpenVSP to be within the NLG and MLG limits. That the original CG from OpenVSP wouldn't match the new CG calculated from CAD was expected even though the difference is rather big with a CG approximately one meter more forward than calculated. Because of this shift in CG the aircraft has to be modified to end up within the CG-envelope.

Note that the weight limits for MRW, MLW, MZFW and MOW is only included to show in which order they may come, they have not been calculated for the CSA due to the restricted time.

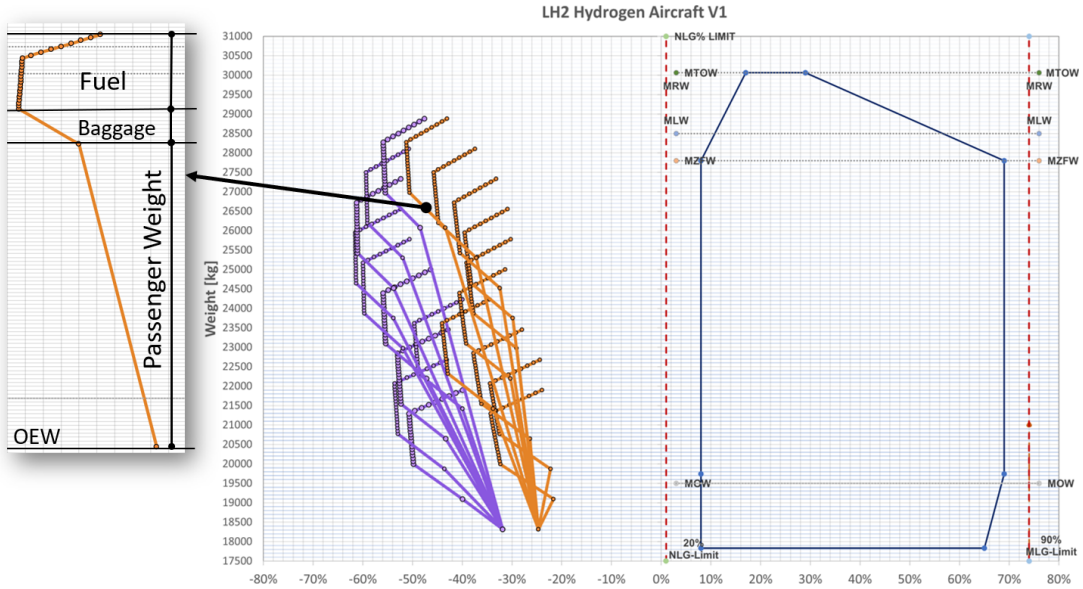


Figure 6.10: *CSA Original design CG envelope*

6.3.3 Modification of CSA Original Design

After all the aircraft data was integrated and the initial CG envelope established, it was observed that the wing placement was excessively rearward. The Mod-Tab in the tool guided a repositioning of the wing, shifting it forward by 2.27 meters to align the CG more closely with the MAC. This adjustment, while improving balance, increased the load on the NLG beyond the preferred limits. Consequently, the MLG was also moved forward to maintain the aircraft's overall balance, a change that did not significantly impact the design.

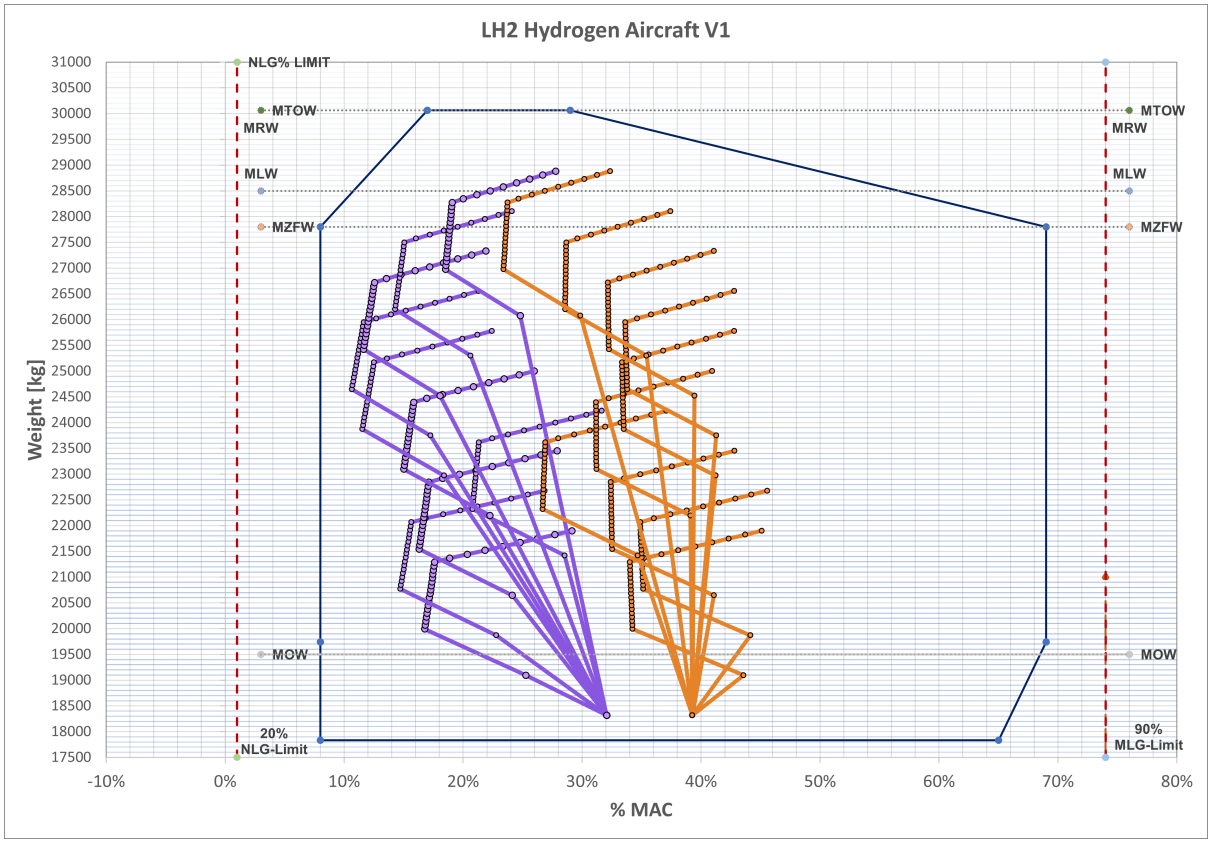


Figure 6.11: *CSA Original design CG envelope, after the use of shifting the wing mod*

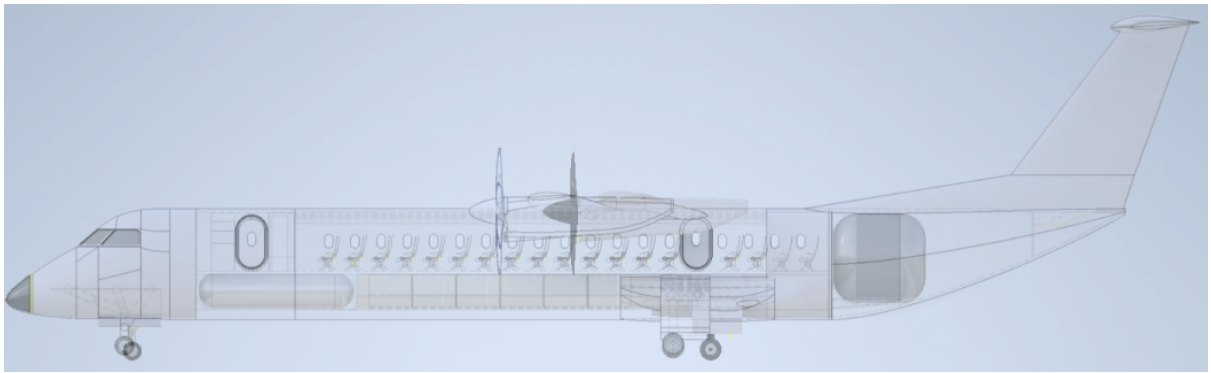


Figure 6.12: *Visualization of design changes after initial analysis using WCT. Main differences are a shift forward of the main wing and a smaller shift forward for the MLG*

6.3.4 Results From Modified CSA Design

6.3.4.1 CG-Envelope - CSA Modified Design

Post-modification, the aircraft's CG envelope displayed significant improvement. The adjustments to the wing and landing gear position resulted in the CG position for all different loading conditions to be close to or at the target of 25% of MAC but also within the envelope limitations. Note however that due to the lack of data for parameters such as stability, the envelope would have been more limited, had all aspects been able to be considered. In figure 6.13 the CG envelope for the CAD modified CSA is displayed.

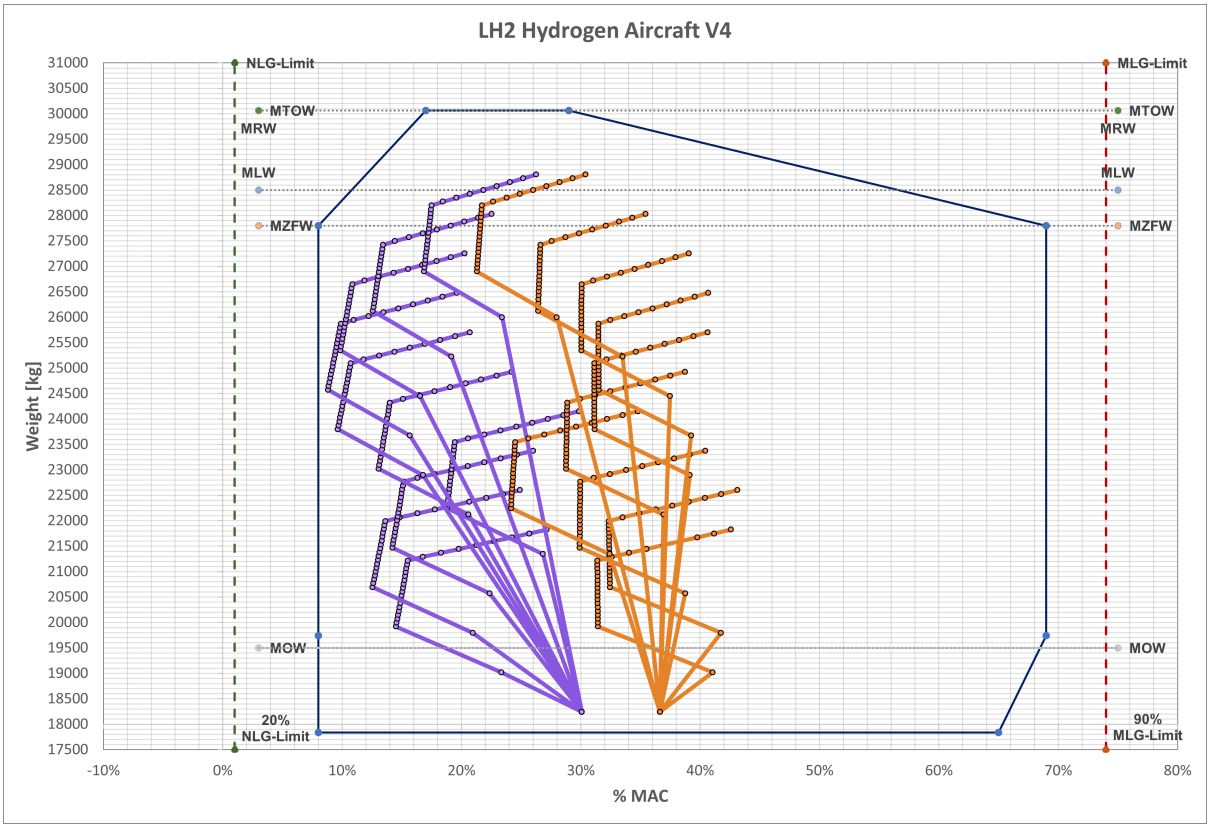


Figure 6.13: *CSA modified design CG envelope, after change of wing position in CAD*

In figure 6.14 the data from the CAD modified CSA is put on top of the data for the original CSA with modified wing position in Excel. Start-point one and three (from left to right) is the in CAD modified data and start-point two and four is for the modification done directly in the tool, the difference between changing in the CAD and using mod in the tool is approximately 2%MAC which corresponds to 5 centimeters in the overall CG. This shows that the tool modifications works really well to play around with to get an indication to what results different changes may produce.

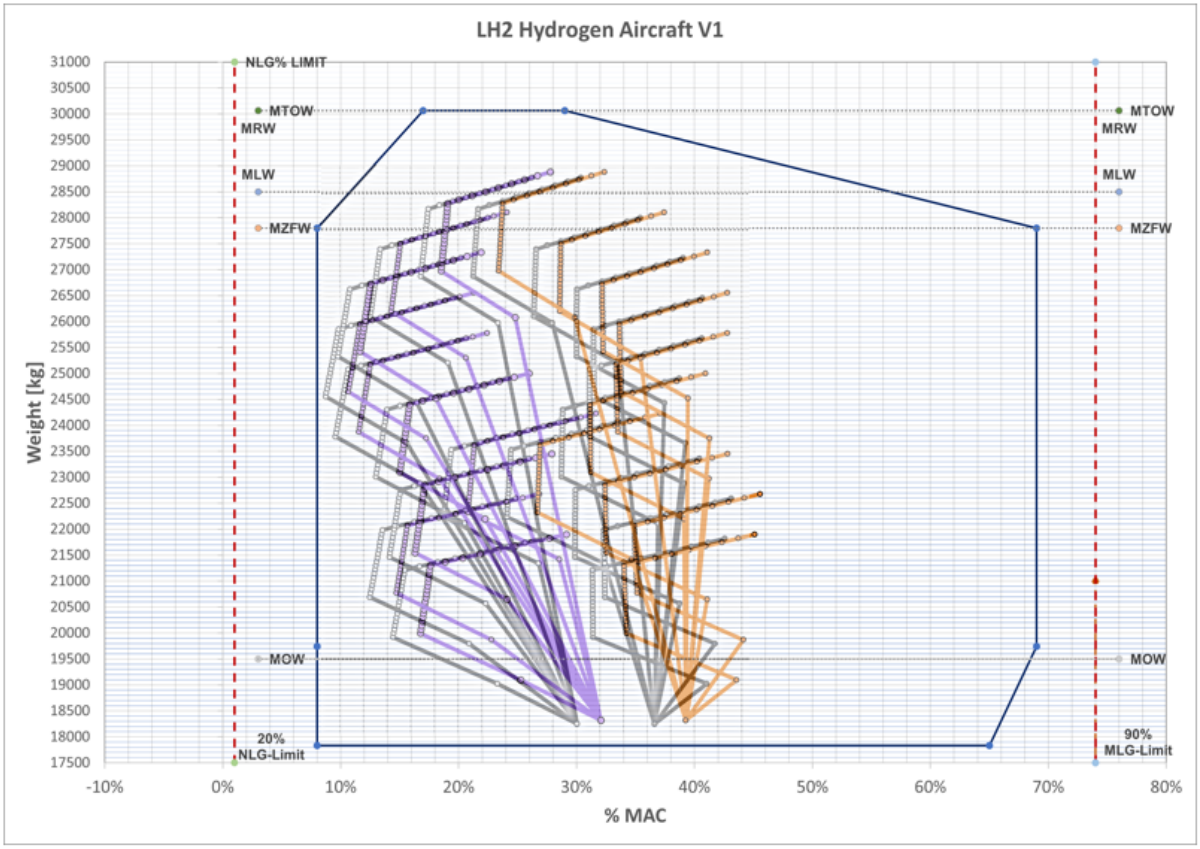


Figure 6.14: *CSA modified design CG envelope on top of original design mod in tool*

6.3.4.2 CG-envelope - Limit Optimization

Because of the margins to the CG envelope aft limit, the decision was made to limit the envelope. This means that there is more margin to the MLG limit but with less risk of making bulky parts as a result of the huge %MAC span that the current envelope has.

In figure 6.15 the new limited envelope can be seen.

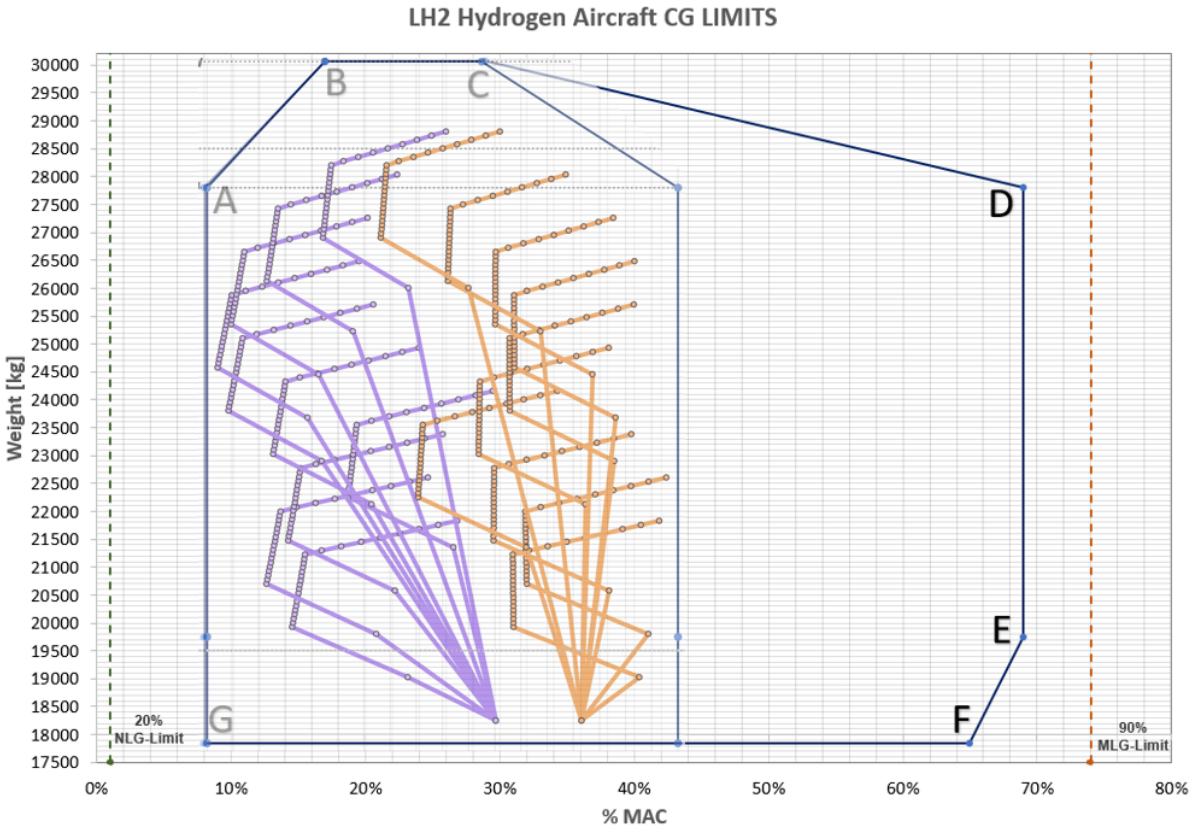


Figure 6.15: *CSA modified design with a new limited CG envelope*

6.3.5 CSA - Mitigation of CG Shift Due to Fuel Quantity

6.3.5.1 Hydrogen Tank Placement

Based on the CG-envelope, mainly the fuel tank curve, it can be seen that with the use of a trim tank, the CG shift can be mitigated. Noticeably is that when the trim tank is full and only the primary tank can be filled the CG travel is significant. This shows the efficiency of making use of a trim tank in this type of aircraft and may motivate a bigger trim tank to be able to mitigate shifts even when being close to full fuel conditions. Worth noting here however is that the increase in weight in the front carries a penalty of having to increase the strength of the NLG or shift the MLG forward. What the result for this design would have been is the implementation of two separate cargo compartments, one in front of the MLG and one behind. Also the height of the gear will have to be increased to protect against tail strike.

To further address the CG shift during flight, repositioning the trim tank aft while enlarging it, coupled with reducing and moving the primary tank forward, would extend the span of mitigated CG shift due to fuel quantity. Implementing this would however require designing a new MLG compartment to accommodate relocated baggage by placing the landing gear farther from the fuselage center-line, creating space above the current MLG location. Although this thesis lacks an aerodynamic analysis to quantify potential drag increases from such modifications, exploring these changes remains a viable option for future designs.

6.3.5.2 Primary and Trim Tank Modified Consumption

By allowing for different consumption from the two tanks, the CG curve for the fuel can be modified. For example, running a 50%/50% split of fuel between the two tanks allows for a almost non existing shift as long as 50% of the total fuel does not exceed the maximum of the trim tank. If the whole fuel amount is to be used then there will be a CG shift independent of consumption logic, in that case, running the tanks to empty at the same fraction rate gives us the most consistent and slowest shift across the flight. For this configuration that would be a 34%/66% consumption logic, i.e 34% is taken from trim and 66% from primary.

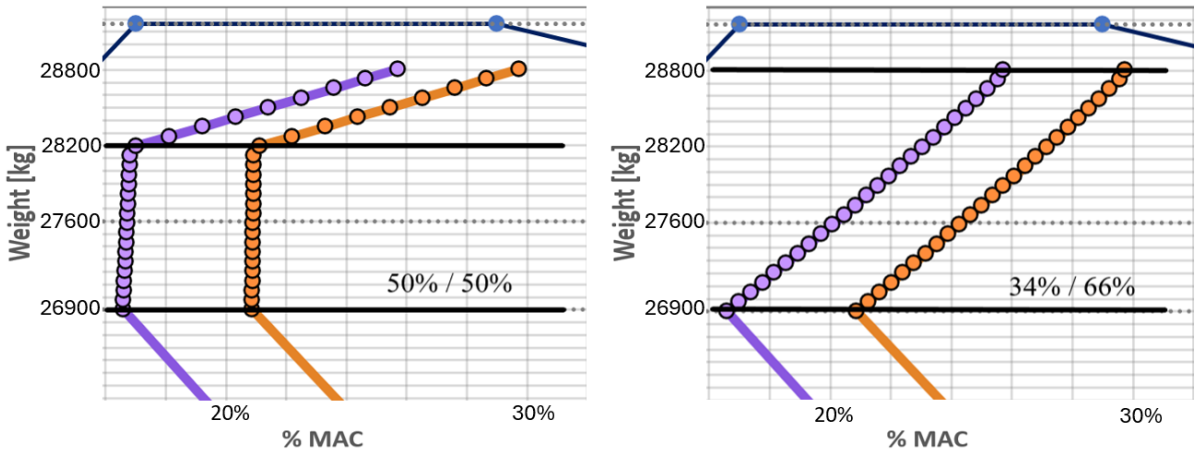


Figure 6.16: *Change in fuel consumtion logic based on needed total fuel. To the left the fuel amount needed is around 60% of total capacity allowing for zero shift CG consumption. The right image is based on 100% fuel amount, in this case it is not possible to mitigate CG shift but instead linearising it for easy prediction*

6.3.5.3 Sizing of Trim Tank

In the current configuration the fuel tanks are split such that the trim tank volume is $9.2m^3$ and the primary tank volume is $17.6m^3$. Based on the logic seen above, running a trim tank that is similar in size gives a lower shift. However the room available in the front is generally smaller than in the aft. This due to the passenger and cockpit compartments. Another way to mitigate the issue is to move the primary tank forward, this gives a smaller momentum arm around the CG point which allows for a smaller CG shift. For the current configuration, this could have been achieved by for example lowering the number of passengers to 85 and thereby removing a row. Running this analysis of tank shift was easy using the WCT and for the current CSA it resulted in issues with too forward CG for some configurations but with some modifications to the design it might be a valid solution to narrow the CG spread.

6.4 ES-30 CG-Envelope

This section presents the CG envelope generated for the ES-30. In contrast to previous case studies where detailed data was shared, the specific data for the ES-30 must remain confidential. Nevertheless, it is still possible to examine and discuss the general trends observed in the aircraft's CG envelope.

6.4.1 CG-Envelope, Turbo Generators Turned On

In figure 6.17, the CG envelope for the ES-30 is presented, including all load cases. Additionally, the flight envelope limits are delineated by solid dark-blue lines. Notably, extensions to the flight envelope are represented by dashed lines. These extensions provide margins to account for variable conditions, such as a flight attendant moving with a galley cart or passengers walking to the lavatory. Some other remarks for the envelope for the electric aircraft is for example the possibility for certain weight limits to be the same due to the electric flight. For example the MLW versus MTOW. In a conventional aircraft the landing limitation will always be lower due to the burning of fuel. One thing that can be noted however is that the placement of the tanks is close to optimal for the aircraft since the shift, independent of fuel amount is small. This since the weight aligns well with the overall CG position.

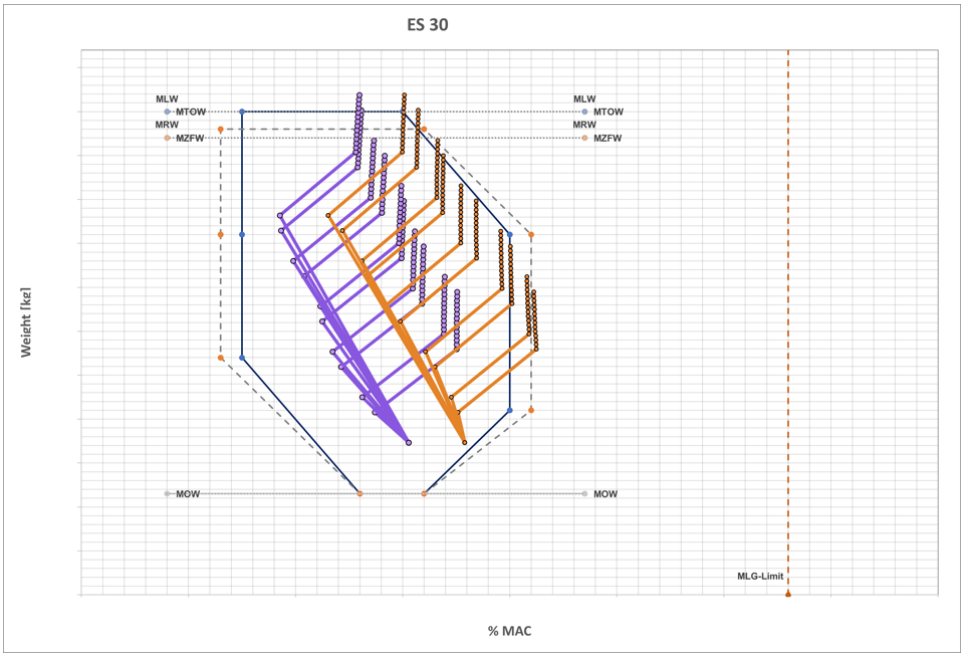


Figure 6.17: *CG-envelope for the ES-30 with pitch angle set to 0 and with turbo generator running. Weight data hidden from axis to avoid disclosure of secret data*

Given the sizes of the fuel tanks, the shift in CG as a result of pitch angle changes is very small. This is barely noticeable in Figure 6.18 below, where the points blur slightly towards the top, indicating a minimal shift. Consequently, the impact of fuel tank shift on the aircraft's flight dynamics is considered negligible.

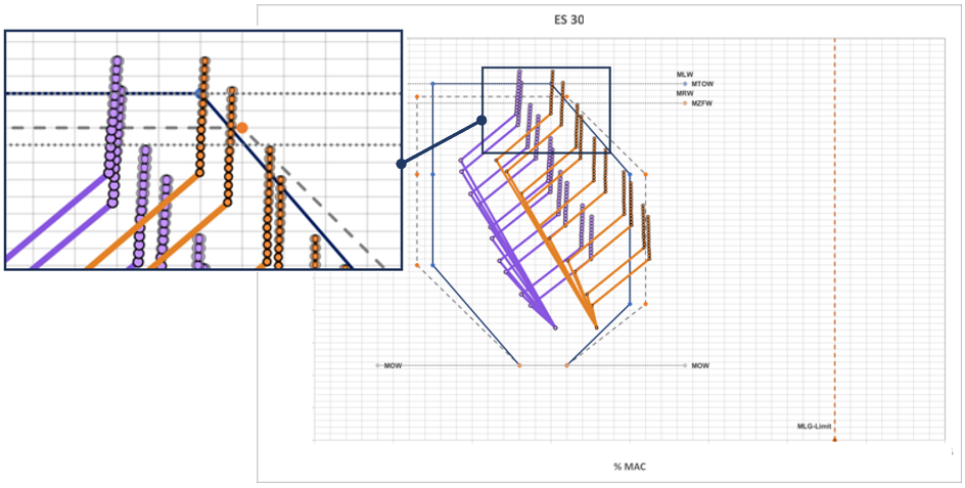


Figure 6.18: *CG-envelope comparison for the ES-30 with pitch angle set to 0 and pitch angle set to 10 when turbo generator running. The points to the right are for pitch set to 10 and the left to points represent pitch set to 0. Weight data hidden from axis to avoid disclosure of secret data*

6.4.2 CG-Envelope, Turbo Generators Turned Off

When the turbo generators are turned off, going fully electric or on ground filling the tanks the CG envelope is similar to what could be seen when running the generator. In figure 6.19 the CG-envelope is presented for pitch 0.

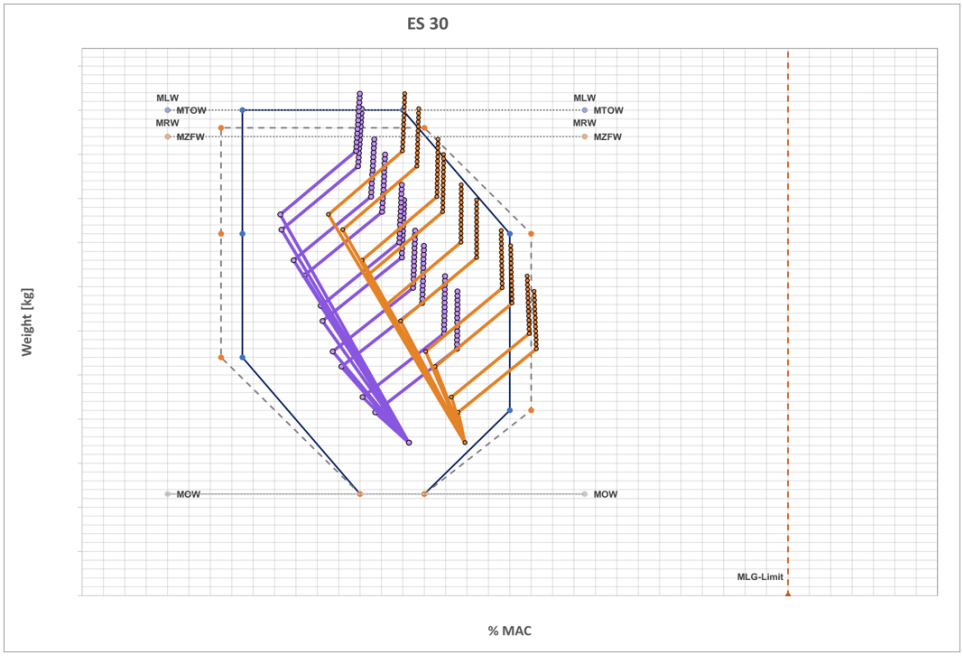


Figure 6.19: *CG-envelope for the ES-30 with pitch angle set to 0 and with turbo generator turned off. Weight data hidden from axis to avoid disclosure of secret data*

Since the turbo generators could be off during flight the pitch angle effect on the CG envelope is investigated for this case as well. In figure 6.20 a comparison between pitch 0 and pitch 10 is presented. The results are very similar to that of the turbo generators

running, showing that the CG envelope does not change significantly and that the change is negligible.

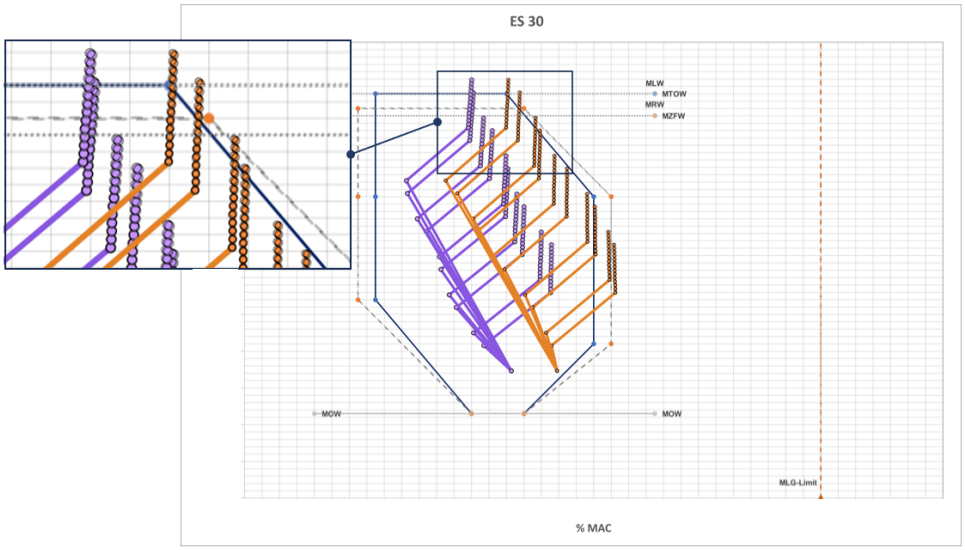


Figure 6.20: *CG-envelope comparison for the ES-30 between pitch angle set to 0 and pitch angle set to 10 when turbo generator are turned off. Data points to the left represents pitch 0 and to the right pitch 10. Weight data hidden from axis to avoid disclosure of secret data*

6.4.3 CG-Envelope, Comparing Turbo Generators Turned On/Off

In figure 6.21 the case when the turbo generator are switched on/off is showcased. The data points for having the turbo generator on is presented on top of the data points for which the turbo generator is off. In the figure the two data sets line up close to perfectly, with just a minor change when looking really close. As mentioned previously when looking at the fuel tank data there is no significant change to the overall CG when the CT tank gets filled or not. This is because the overall weight of the aircraft is much greater to that of the fuel, so even if the relative CG change of the fuel is large, to make an impact on the overall CG the fuel volume must be bigger.

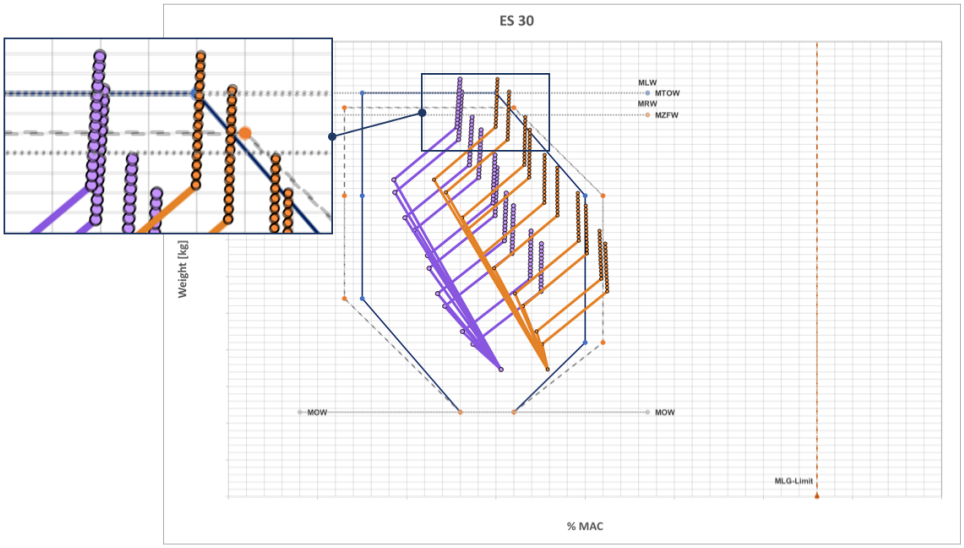


Figure 6.21: *CG-envelope comparison for the ES-30 with pitch angle set to 0 for both the turbo generator running and having it off, difference is to small to specify which data points belong to which case. Weight data hidden from axis to avoid disclosure of secret data*

As a final comment about the CG envelope and the impact from fuel CG shifts for all presented cases, the fuel volume of this electric aircraft is so small that the impact is negligible. However if the design is changed and its decided to increase the fuel volume, bigger changes will be shown. The same goes for an decrease in weight of the rest of the aircraft. This means that as new batteries develops the change might increase if the weight of the batteries decreases.

6.4.4 Flight Extension Envelope: CG Sensitivity, Passenger Moving Fwd/Aft etc..

The flight extension envelope is designed to provide an additional limited zone for flight, preventing the center of gravity from moving outside the aircraft's design limitations. This is to accommodate various shifts, such as multiple passengers moving simultaneously,the movement of a galley cart, flaps movement in flight, landing gear extension/retraction, fuel density etc.. Although the shifts from passengers or the galley cart might seem minor relative to the total weight of the aircraft, the leverage effect created by the movement across the cabin can significantly impact the aircraft's stability.

To further illustrate the critical importance of managing this shift, consider the 2010 Filair Let L-410 crash. During the flight, passengers panicked upon noticing a crocodile that had escaped from a passenger's duffel bag. In their attempt to distance themselves from the animal, the majority of passengers suddenly moved to one end of the aircraft to the other. This abrupt shift caused a significant alteration in the aircraft's center of gravity, pushing it beyond the safety limits of the flight envelope. The pilots were unable to regain control, leading to the aircraft's spiral and subsequent crash since the elevator authority no longer was sufficient for the shifted CG position. This tragic event highlights the vital necessity of ensuring that potential internal shifts are rigorously managed and kept within the designated safety margins.

7
Conclusions

This report concludes our examination of the aircraft conceptual design process, focusing on the application of advanced mass and balance techniques within the development of the ES-30 and a case study aircraft. The research developed and implemented tools and methodologies intended to enhance the speed and accuracy of weight and balance calculations. These tools facilitate rapid adjustments to design modifications, which is essential for ensuring aircraft stability and precise CG predictions.

The application of these methods to a case study involving a LH_2-propelled aircraft has allowed us to evaluate their effectiveness across different aircraft designs. This analysis helps identify both the challenges and potential of integrating liquid hydrogen fuel systems in aviation.

The findings from this study provide insights that could inform the future direction of aircraft design, particularly in terms of improving efficiency and incorporating sustainable technologies. The continued refinement of these tools and methods could help reduce the resources typically required in the design process, supporting industry efforts to meet evolving performance and environmental requirements.

1. **Refinement of Weight and Balance Methodologies for the ES-30:** The refined methodologies improved the accuracy of weight and balance calculations, leading to better predictions of the CG throughout the aircraft design process. This improvement supports iterative design and certification processes by ensuring more reliable data.
2. **Development of a CG Envelope for Operational Flight Conditions:** A comprehensive CG envelope was created to ensure aircraft stability under various conditions. This envelope helps define the operational limits and predict aircraft behavior accurately.
3. **Evaluation of Fuel Volume and Pitch Angles on Performance and CG Envelope:** The study assessed how changes in fuel volume and pitch angles affect the aircraft's CG and stability. This evaluation is essential for maintaining proper balance and safe operation during different phases of flight.
4. **Suggestion of Testing Points for Crucial CG Locations:** Specific testing points were identified to validate the aircraft's stability and safety. These points are necessary for confirming that the aircraft performs as expected under different flight conditions.
5. **Creation of an Automated Tool for Dynamic CG Calculations:** An automated tool was developed to provide real-time CG calculations, facilitating quick design modifications. This tool reduces the need for manual recalculations and speeds up the prototyping and testing phases.
6. **Contribution to the Optimization of Electric Aircraft Performance:** The tools and methods developed have improved the ES-30 aircraft's weight and balance management, supporting sustainable aviation efforts by enhancing design efficiency.

7. **Case Study on LH_2 Propelled Aircraft and Tool Applicability:** The tools were applied to a liquid hydrogen (LH2) aircraft case study, demonstrating their effectiveness and adaptability to different propulsion technologies. This shows their potential for future sustainable aircraft projects and innovation in aircraft design.

8

Future Work

8.1 Validation

Even-though the fuel tank tool has been validated with regards to the CAD model, the CAD model/solid model will not include the fluid mechanics which appear for both conventional aviation fuel and LH_2. For example the impact from sloshing is neglected completely, and the accuracy of measuring inertia and CG from CAD is assumed to be accurate. For future work there are many ways to evaluate how accurate the tool is, but we would like to estimate how it would compare to run several CFD simulations for different pitch angles to compare the results. This is because both CFD and the tool could be used during the conceptual design phase. To investigate how the results differ could indicate for which types of design changes the tool could be used, to efficiently update the data and for which changes there is a need for CFD. Another part which would be interesting is to confirm or validate the results when looking at the LH_2 tank's placement and find out if a general rule could be found which indicates for which designs the CG shift is mitigated for the hydrogen fuel tank case.

8.2 Improvements

As mentioned above, the model does not take sloshing into account and because of that its impact cannot be measured. It would be interesting to see if there is a way to make a sloshing model and try to correlate how much of an impact the worst case scenarios would have. This could then be added to the data from the tool. However since the tool is mainly aimed at making the conceptual design process faster this might not be necessary to investigate but instead focus on the systematic structure of all the data gathered. To ease the management and overall control might be of bigger interest. We also have no way of knowing how much of the fuel is trapped on internal structures which could help with the fuel tank design. This might not have an impact on the CG-envelope but to get an efficient fuel tank where the unusable fuel is minimized, will help the overall efficiency. By not carry more weight than necessary. If the model could take this into account at the same time this would help the conceptual design further and would not add any extra time since the tool is already being used.

Bibliography

[1] R. S. Shevell, "Technological development of transport aircraft - past and future," *Aerospace Research Central*, vol. 17, no. 2, Article No. 78–1530R, 1980, Accessed on: Feb. 20, 2024. DOI: 10.2514/3.57876.

[2] E. Baharozu, G. Soykan, and M. B. Ozerdem, "Future aircraft concept in terms of energy efficiency and environmental factors," *Energy*, vol. 140, no. Part 2, pp. 1368–1377, 2017, Accessed on: Feb. 20, 2024. DOI: 10.1016/j.energy.2017.09.007.

[3] B. M. Enuh. "How are aircraft tested for safety and performance?" Accessed on: Feb. 19, 2024. (Jun. 2023), [Online]. Available: https://www.azom.com/article.aspx?ArticleID=22760.

[4] K. M. Pavlock, *Aerospace Engineering Handbook*. National Aeronautics and Space Administration, 2013.

[5] *Es-30 electric airplane*, https://heartaerospace.com/es-30/, Accessed: Feb. 27, 2024, n.d.

[6] D. P. Raymer, *Aircraft Design: A Conceptual Approach*. American Institute of Aeronautics and Astronautics, 2018, ISBN: 9781624104909.

[7] B. Khandelwal, A. Karakurt, P. R. Sekaran, V. Sethi, and R. Singh, "Hydrogen powered aircraft: The future of air transport," *Progress in Aerospace Sciences*, vol. 60, pp. 45–59, 2013. DOI: 10.1016/j.paerosci.2012.12.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0376042112000887?via=ihub.

[8] A. Gomez and H. Smith, "Liquid hydrogen fuel tanks for commercial aviation: Structural sizing and stress analysis," *Aerospace Science and Technology*, vol. 95, Dec. 2019, Cranfield University, Bedfordshire, United Kingdom, MK43 0AL. DOI: 10.1016/j.ast.2019.105438.

[9] H. G. Klug and R. Faass, "Cryoplane: Hydrogen fuelled aircraft — status and challenges," *Air Space Europe*, vol. 3, no. 3-4, pp. 252–254, 2001. DOI: 10.1016/S1290-0958(01)90110-8. [Online]. Available: https://doi.org/10.1016/S1290-0958(01)90110-8.

[10] G. Onorato, P. Proesmans, and M. F. M. Hoogreef, "Assessment of hydrogen transport aircraft: Effects of fuel tank integration," *CEAS Aeronautical Journal*, vol. 13, pp. 813–845, 2022, Received: 11 February 2022 / Revised: 11 May 2022 / Accepted: 27 June 2022 / Published online: 17 September 2022. DOI: 10.1007/s13272-022-00601-6. [Online]. Available: https://doi.org/10.1007/s13272-022-00601-6.

[11] G. Dannet, "Integration of cryogenic tanks and fuel cells for future hydrogen-powered aircraft," Master's programme. Spring/Autumn 2021. LIU-IEI-TEK-A–

21/04164-SE, Master's Thesis, Linkoping University, Department of Management and Engineering, 2021.

[12] G. Tan and C. Zhu, "Volume characteristic calculation based on catia secondary development," in *2020 IEEE International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA)*, IEEE, 2020, pp. 595–598.

[13] R. Tookey, M. Spicer, and D. Diston, "Integrated design and analysis of an aircraft fuel system," in *Proceedings of the RTO AVT Symposium on Reduction of Military Vehicle Acquisition Time and Cost through Advanced Modelling and Virtual Simulation*, Available through the Defense Technical Information Center (DTIC), Compilation Part Notice ADP014159, ADA415759, NATO Research and Technology Organization, Paris, France, n.d.

[14] B. Rogerio and W. Tiago, "Automotive industry cad system automation through knowledge based engineering (kbe) case study," SAE International, Sao Paulo, Brazil, Tech. Rep. 2016-36-0372, 2016. DOI: 10.4271/2016-36-0372. [Online]. Available: https://doi.org/10.4271/2016-36-0372.

[15] Ohaju, *Airplane drawing line art*, https://www.cleanpng.com/png-airplane-drawing-line-art-clip-art-plane-1103055/, Accessed on: Feb. 27, 2024, n.d.

[16] T. Grönstedt, "Lecture 10 - stability & control," in *Aircraft Design MMS235*. Chalmers University of Technology, 2023.

[17] D. P. Raymer, "Propulsion and fuel system integration," in *Aircraft Design: A Conceptual Approach*, 6th ed. American Institute of Aeronautics and Astronautics, 2018, ch. 10, pp. 275–336, ISBN: 9781624104909.

[18] R. Langton, C. Clark, M. Hewitt, and L. Richards, "Introduction," in *Aircraft Fuel Systems*, 1st ed. John Wiley and Sons, Ltd., 2009, ch. 1, pp. 1–18.

[19] R. Langton, C. Clark, M. Hewitt, and L. Richards, "Fuel storage," in *Aircraft Fuel Systems*, 1st ed. John Wiley and Sons, Ltd., 2009, ch. 3, pp. 31–52.

[20] F. Murmann, *Piper pa-23 apache 3-view line drawing*, https://commons.wikimedia.org/wiki/File:Piper_PA-23_Apache_3-view_line_drawing.svg, Accessed on: Feb. 28, 2024, n.d.

[21] R. Langton, C. Clark, M. Hewitt, and L. Richards, "Fuel system design drivers," in *Aircraft Fuel Systems*, 1st ed. John Wiley and Sons, Ltd., 2009, ch. 2, pp. 19–30.

[22] S. Gudmundsson, "Aircraft weight analysis," in *General Aviation Aircraft Design: Applied Methods And Procedures*, 1st ed. United Kingdom: Butterworth-Heinemann, 2014, ch. 6, pp. 134–180.

[23] G. Sphati, *Aircraft cg envelopes*, https://www.sawe.ca/download/tech2011/Aircraft%20CG%20Envelopes.pdf, Sep. 2011.

[24] "14 cfr § 25.23 - load distribution limits." Accessed on 21 Feb 2024, Electronic Code of Federal Regulations. (n.d.), [Online]. Available: https://www.ecfr.gov/current/title-14/chapter-I/subchapter-C/part-25/subpart-B/subject-group-ECFR557993314ee6275/section-25.23 (visited on 02/21/2024).

[25] "14 cfr § 25.27 - center of gravity limits." Accessed on 21 Feb 2024, Electronic Code of Federal Regulations. (n.d.), [Online]. Available: https://www.ecfr.gov/

current/title-14/chapter-I/subchapter-C/part-25/subpart-B/subject-group-ECFR557993314ee6275/section-25.27 (visited on 02/21/2024).

[26] D. P. Raymer, "Weights," in *Aircraft Design: A Conceptual Approach*, 6th ed. American Institute of Aeronautics and Astronautics, 2018, ch. 15, pp. 275–336, ISBN: 9781624104909.

[27] "14 cfr § 25.25 - weight limits." Accessed on 21 Feb 2024, Electronic Code of Federal Regulations. (n.d.), [Online]. Available: https://www.ecfr.gov/current/title-14/chapter-I/subchapter-C/part-25/subpart-B/subject-group-ECFR557993314ee6275/section-25.25 (visited on 02/21/2024).

[28] S. of Allied Weight Engineers, *Weight Engineers' Handbook*, Revised May. Los Angeles, CA: Society of Allied Weight Engineers, Inc., 2011, ch. 5, pp. 5.3–5.10.

[29] S. of Allied Weight Engineers, *Weight Engineers' Handbook*, Revised May. Los Angeles, CA: Society of Allied Weight Engineers, Inc., 2011, ch. 5, pp. 5.11–5.16.

[30] S. of Allied Weight Engineers, *Weight Engineers' Handbook*, Revised May. Los Angeles, CA: Society of Allied Weight Engineers, Inc., 2011, ch. 5, pp. 5.18–5.29.

[31] E. W. Weisstein. "Rotation matrix." Provided by Wolfram MathWorld. (2024), [Online]. Available: https://mathworld.wolfram.com/RotationMatrix.html (visited on 02/25/2024).

[32] V. D. Paula and H. Rozenfeld. "Mass properties management in aircraft development process: Problems and opportunities." (2015).

[33] N. Fircroft. "The boeing 777 and how computer aided design changed the face of air travel." Accessed on: Mar. 5, 2024. (Jul. 2020), [Online]. Available: https://www.nesfircroft.com/resources/blog/the-boeing-777-and-how-computer-aided-design-changed-the-face-of-air-travel/.

[34] "How the aerospace industry is using cad design services: Drafting, modeling, and prototyping." Accessed on: Mar. 5, 2024. (Feb. 2023), [Online]. Available: https://www.cadcrowd.com/blog/how-the-aerospace-industry-is-using-cad-design-services-drafting-modeling-and-prototyping/.

[35] D. R. Ziethen, *CATIA V5: Macro Programming with Visual Basic Script*. McGraw-Hill Education, 2013, Accessed: 2024-03-20, ISBN: 9780071800020. [Online]. Available: https://www.mhprofessional.com/catia-v5-9780071800020-usa.

[36] Microsoft. "Transfer data to excel from visual basic." Accessed: 2024-03-20. (May 2022), [Online]. Available: https://learn.microsoft.com/en-us/office/troubleshoot/excel/transfer-data-to-excel-from-vb.

[37] Microsoft. "Getting started with vba in office." Accessed: 2024-03-20. (Jun. 2022), [Online]. Available: https://learn.microsoft.com/en-us/office/vba/library-reference/concepts/getting-started-with-vba-in-office.

[38] M. Jenkins. "Vba vs. vb.net - what's the difference?" Accessed: 2024-03-20. (Aug. 2020), [Online]. Available: https://software-solutions-online.com/vba-vs-vb-net/.

[39] Microsoft. "Com technical overview." Accessed: 2024-03-20. (2020), [Online]. Available: https://learn.microsoft.com/en-us/windows/win32/com/the-component-object-model.

[40] "Catia v5 automation." Accessed: 2024-03-20. (n.d.), [Online]. Available: https://catiadesign.org/_doc/V5Automation/.

[41] Microsoft. "Getting started with vba in office." Accessed: 2024-03-20. (Jun. 2022), [Online]. Available: https://learn.microsoft.com/en-us/office/vba/library-reference/concepts/getting-started-with-vba-in-office.

[42] W. contributors, *Ata 100*, https://en.wikipedia.org/wiki/ATA_100, Accessed on: Mar. 6, 2024, n.d.

[43] "Understanding ata chapter 100: The golden rules of air transport." Accessed on: Mar. 7, 2024. (2022), [Online]. Available: https://defensebridge.com/blog-home/understanding-ata-chapter-100-the-golden-rules-of-air-transport.html.

[44] A. Team. "The ata standard numbering system." Accessed on: Mar. 6, 2024. (Nov. 2023), [Online]. Available: https://www.aviationhunt.com/ata-standard-numbering-system/.

[45] S. Gudmundsson, *General Aviation Aircraft Design: Applied Methods And Procedures*. Butterworth-Heinemann, 2014.

[46] Bombardier Aerospace, *Airport planning manual*, December 5, 2014, Series 400, Airport Planning Manual for Dash 8 Series 400, Accessed: 2024-04-15, Bombardier Aerospace, Toronto, Ontario, Canada, 2014. [Online]. Available: https://customer.aero.bombardier.com/webd/BAG/CustSite/BRAD/RACSDocument.nsf/51aae8b2b3bfdf6685256c300045ff31/ec63f8639ff3ab9d85257c1500635bd8/$FILE/ATTNBEOB.pdf/D8400-APM.pdf.

A

Appendix 1

A.1 Case study aircraft design Python code

Case study Aircraft Design Python Code

Michael Crona, Simon Dinger and Per Samuelsson

October 2023

```python
# -*- coding: utf-8 -*-
"""
Created on Tue Aug 29 16:25:10 2023

@author: Michael Crona, Simon Dinger, Per Samuelsson
"""

# MMS236 Aircraft design
# Department of Mechanics and Maritime Sciences
# Division of Fluid Dynamics
# August 2023

#============================================================
# Packages needed
import numpy as np
import math as m
import numpy as np
import matplotlib.pyplot as plt
import scipy as sc
#============================================================









#
#%%
############################## UNIT CONVERTION ############
bar2Pa = 1e5               # converts bar to Pascal
ft2m = 0.3048              # converts feet to meter
m2ft = 3.28084
lb2kg = 0.45359237         # converts pounds to kilogram
kg2lb = 2.20462            # converts kilogram to pound
hp2w = 745.7               # converts horsepower to watt
```

```
w2hp = 0.00134102        # converts watt to horsepower
NM2m = 1852              # converts Nautical miles to meters
knot2ms = 0.514444       # knots to meter per second
ms2knot = 1.94384

#%%
############################## INITIAL DATA ############

gamma_air = 1.4                 # []
R_air = 287.05                  # [J/(kg*K)]
Cp_air = 1005
g_0 = 9.81                      # [m/(s^2)]
PAX=90
m_PAX = 105                     # kg
pressure_tank = 2*bar2Pa        # Pa
EIS_year = 2035                 # Technology year
N_crew = 2                      # number of crew members
N_pilots = 2                    # number of pilots
# Efficiencies

efficiency_prop = 0.8

# Take-off and landing
TOFL = 1400                             # meters, take-off length
    with MTOW, SeaLevel, ISA
TTC = 15.5*60                           # time to cruise (15 minutes,
     from 1500ft)
speed_app = 119*knot2ms                 # Approach speed (meter per
    second)
LFL = 1350                              # meters, landing length (MLW
    , ISA)

# Cruise & Reserves

altitude_cruise = 25000*ft2m  # (ISA + 10) Initial cruise altitude,
     meters
Mach_cruise = 0.5             # Initial Mach number
cruise_range = 1100*NM2m      # Range in meters

# Reserves

diversion_range = 100*NM2m        # Range of diversion in meters
loiter = 20*60                    # Time of loiter in seconds
reserve_fuel = 1.06               # 5% extra fuel in reserve, 1%
    trapped fuel

# Lift/wing area values
K_LD = 11
S_wet_S_ref = 7         #(Estimated value by comparing Dash-8 Q400
    and known wings)
aspect_ratio = 13       #(Roughly estimated from the Dash-8 Q400)
taper_ratio = 0.45      # selected from page 83
#aspect_ratio = 15.5
lift_drag_ratio_max = K_LD*np.sqrt(aspect_ratio/S_wet_S_ref)
lift_drag_ratio_cruise = lift_drag_ratio_max
lift_drag_ratio_loiter = lift_drag_ratio_max*0.866
wing_span = 30 #[m]
```

```
95
96 # Fuel tank gravimetric index
97 G_i = 0.35
98 # Composite fudge factor
99 fudge_factor_composite = 1 # Raymer suggests 0.95 if light weight
       composite construction is used.
100
101
102 #%% SFC guessing from trendline (Turboprop at take-off, Raymer)
103 SFC = [
104     [1946, 0.105],
105     [1970, 0.1],
106     [1960, 0.095],
107     [1987, 0.085],
108     [1990, 0.08],
109     [1973, 0.078],
110     [1993, 0.077],
111     [1992, 0.076],
112     [2002, 0.075],
113     [1999, 0.074],
114     [2006, 0.068],
115     [2022, 0.064],
116     [2022, 0.075]
117 ]
118
119 SFC = sorted(SFC, key=lambda x: x[0])
120 years = [x[0] for x in SFC]
121 values = [x[1] for x in SFC]
122
123 # Fit a polynomial regression model
124 coefficients = np.polyfit(years, values, 2)
125 poly = np.poly1d(coefficients)
126
127 # Create an extended range of years up to 2035
128 extended_years = np.arange(min(years), 2036)
129
130 # Calculate trendline values for the extended range
131 extended_trendline = poly(extended_years)
132
133 # Interpolate value at x = 2035
134 SFC_interpolated_value = poly(2035)
135 SFC_TO = SFC_interpolated_value # SFC at take-off, mg/Ws
136 # Append the interpolated value to the SFC list
137 SFC.append([2035, SFC_interpolated_value])
138
139 # Plotting
140 res=400
141 plt.figure(dpi=res)
142 plt.plot(years, values,'o', label='Data')
143 plt.plot(years, poly(years), color='red', label='Trendline')
144 plt.plot(extended_years, extended_trendline, color='orange',
       linestyle='--', label='Extended Trendline')
145 plt.scatter(2035, SFC_interpolated_value, color='green', label='
       Interpolated (2035)')
146 plt.xlabel('Year')
147 plt.ylabel('SFC [mg/Ws]')
148 plt.title('SFC Values Over Years with Trendline and Interpolation')
```

```
149 plt.legend()
150 plt.grid(True)
151 plt.show()
152
153 #%%
154
155
156 # Jet-A to Hydrogen
157
158 LHV_H2 = 120e6 #J/kg [Given in assignment]
159 LHV_jetA = 43.1e6 #J/kg       [wikipedia]
160 volumetric_density_hydrogen = 71 #kg/m3 [https://demaco-cryogenics.
        com/blog/energy-density-of-hydrogen/]
161 volumetric_density_JetA = 804     #kg/m3[wikipedia]
162
163 #energy_per_m3 = volumetric_energy_density_hydrogen/
        volumetric_energy_density_JetA
164
165 #Approx_energy_needed=45936e6
166 #kg_needed = Approx_energy_needed/LHV_H2
167 #volume_needed = kg_needed/volumetric_energy_density_hydrogen
168
169 SFC_power_TO = SFC_interpolated_value # SFC at take-off, mg/Ws
170
171 SFC_JetA_to_H2 = LHV_jetA/LHV_H2
172 #%%
173 ############################## GUESSES ########################
174
175 # Constant over mission
176
177 w_crew = m_PAX*(N_crew+N_pilots)   # Crew weight in kilogram
178 w_payload = m_PAX*PAX              # Payload weight in kilograms
179
180
181
182 # | TO | Climb  |  Cruise 1100 nm |Land. 1 |Loiter| Climb 2|Cruise
        100nm|Landing 2| Taxing
183
184
185 # Sizing loop
186 n = 1 # For n=1, initial sizing, n=2 "refined sizing"
187 for i in range(n):
188
189     #%% Iterate solution
190     w_PAX = m_PAX*PAX
191     tolerance = 0.001
192     difference = 1
193     w0_guess = 50000
194     while difference > tolerance:
195
196         #%% Take-off and warmup
197         w_warmup_takeoff = 1-(1-0.97)*LHV_jetA/LHV_H2   # weight
        fraction update when confirmed number  [tab 3.2 page 34]
198         mass_warmup_takeoff = (w_warmup_takeoff)*w0_guess
199         # Calculate higher value for hydrogen (>0.97)
200
201
```

```
        #%% climb 1
        if i == 0:
            w_climb1 = 1-(1-0.985)*LHV_jetA/LHV_H2     # [tab 3.2
    page 34]
        # Calculate higher value for hydrogen (>0.985)
        if i > 0:
            w_climb1 = 1-(1-(1.0065  -0.0325*0.5))*LHV_jetA/LHV_H2 #
     Updated method, p.150
        mass_climb1 = (w_climb1)*mass_warmup_takeoff
        #%% Cruise 1 1100NM

        altitude_cruise1=ft2m*25000
        dTisa = 10                                  # [K]
        T_isa_cruise = 219.9 + ((altitude_cruise1 -10500)
    /(11000-10500))*(216.65-219.9) + dTisa # [K]
        a_cruise1 = m.sqrt(gamma_air*R_air*T_isa_cruise)
        speed_cruise= a_cruise1*Mach_cruise
        SFC_power_cruise1 = SFC_power_TO*0.9 #[mg/Ws] Assumed a 10%
     better value than take-off
        SFC_cruise1 = SFC_JetA_to_H2*(SFC_power_cruise1*
    speed_cruise/efficiency_prop)/(10**6) #[kg/Ns]

        w_cruise1 = np.exp((-cruise_range*SFC_cruise1*g_0)/(
    speed_cruise*lift_drag_ratio_cruise))
        mass_cruise1 = (w_cruise1)*mass_climb1


        #%% Landing

        w_landing1 = 1-(1-0.995)*LHV_jetA/LHV_H2   # [tab 3.2 page
    34]
        # Calculate higher value for hydrogen (>0.995)
        mass_landing1 = (w_landing1)*mass_cruise1
        #%% Loiter
        altitude_loiter=ft2m*1500
        speed_loiter = 200*knot2ms
        SFC_power_loiter = 0.08 # [mg/Ws]
        SFC_loiter = SFC_JetA_to_H2*(SFC_power_loiter*speed_loiter/
    efficiency_prop)/(10**6) #[kg/Ns]
        w_loiter = np.exp(-loiter*SFC_loiter*g_0/(
    lift_drag_ratio_loiter))
        mass_loiter = (w_loiter)*mass_landing1

        #%% Climb 2

        w_climb2 = 1-(1-0.985)*LHV_jetA/LHV_H2      # [tab 3.2 page
    34]
        # Calculate higher value for hydrogen (>0.985)
        mass_climb2 = (w_climb2)*mass_loiter

        #%% Cruise 2 100NM

        w_cruise2 = np.exp((-diversion_range*SFC_cruise1*g_0)/(
    speed_cruise*lift_drag_ratio_cruise))
        mass_cruise2 = (w_cruise2)*mass_climb2

        #%% Landing 2
```

```
248
249         w_landing2 =1-(1-0.995)*LHV_jetA/LHV_H2   # [tab 3.2 page
    34]
250         # Calculate higher value for hydrogen (>0.995)
251         mass_landing2 = (w_landing2)*mass_cruise2
252
253         #%% Total fuel fraction
254         w_x = w_warmup_takeoff*w_climb1*w_cruise1*w_landing1*
    w_loiter*w_climb2*w_cruise2*w_landing2
255         w_fuel_fraction = reserve_fuel*(1-w_x)
256
257         # ================================================
258         # Empty weight fraction functions
259
260         def w_empty_weight_function(W_0):
261             w_empty_weight_fraction = fudge_factor_composite
    *((0.92)*W_0**(-0.05))
262             return  w_empty_weight_fraction
263                     # increased 0.92 with a small number in order
    to account for a slightly worse empty weight fraction due to
    more empty weight using hydrogen
264                     # Multiplying by a fudge-factor of 0.95
    assuming composite materials.
265
266         # Updated version used after additional values have been
    identified. p.
267         def w_empty_weight_function2(W_0, aspect_ratio, PW, WS, V):
268             W_0_fps = W_0*kg2lb
269             PW_fps  = PW*(w2hp)/(kg2lb)
270             WS_fps = WS*(kg2lb)/(m2ft**2)
271             V_fps = V*ms2knot
272             fraction = (0.37 + 0.09*(W_0_fps**-0.06)*(aspect_ratio
    **0.08)*(PW_fps**0.08)*(WS_fps**-0.05)*(V**0.3))
273             return fraction
274         # ================================================
275
276         if i == 0:
277             w_empty_fraction = w_empty_weight_function(w0_guess)
278         if i > 0:
279             w_empty_fraction = w_empty_weight_function2(w0,
    aspect_ratio, PW_installed, WS_stall, speed_cruise)
280         w0 = (w_crew + w_PAX )/(1-w_fuel_fraction-w_empty_fraction)
281         w_fuel = w_fuel_fraction*w0
282         w_tank = w_fuel/G_i - w_fuel
283         w0 = w0 + w_tank
284         difference = np.abs(w0-w0_guess)
285         w0_guess = (w0_guess + w0)/2
286         #print(w0)
287         w_empty_fraction = (w0_guess -(w_crew +w_fuel +w_PAX))/
    w0_guess
288     w_empty = w_empty_fraction*w0
289     volume_fuel = w_fuel/volumetric_density_hydrogen
290
291     OEW_MTOW = (w0-w_PAX-w_fuel)/w0
292
293     #%% Calculate power to weight ratio
294     #%%% Statistical power to weight ratio, Raymers correlation
```

```
295
296     PW_stat = (0.016*speed_cruise**0.5)*1000 #[W/kg}
297
298     #%%% Trust matching
299
300     # Caluclate the thrust to weight ratio and power to weight
        ratio for cruise
301     TW_cruise = 1/(lift_drag_ratio_cruise)
302     PW_cruise = TW_cruise/((efficiency_prop/speed_cruise)*(1/g_0))
        #[W/kg]
303
304     # Caluclate the thrust to weight ratio and power to weight
        ratio during climb
305     speed_vertical = (altitude_cruise-1500*ft2m)/TTC
306     TW_climb = 1/(lift_drag_ratio_cruise) + speed_vertical/(
        speed_cruise)
307     PW_climb = TW_climb/((efficiency_prop/speed_cruise)*(1/g_0)) #[
        W/kg]
308
309     # Calculate the power to weight ratio during climb
310     # For this, the mass at cruise and take-off is needed
311
312     power_cruise = PW_cruise*mass_cruise1 #[W]
313     power_takeoff = power_cruise/0.6 #[W]
314     PW_takeoff = power_takeoff/w0 #[W/kg]
315
316     PW_max = max(PW_takeoff, PW_climb, PW_cruise, PW_stat) #[W/kg]
317
318     #%% Calculate wing loading
319     airport_altitude = 0 #[m] The elevation of the landing strip
320     airport_temperature = 219.9 + ((airport_altitude-10500)
        /(11000-10500))*(216.65-219.9) + dTisa
321     airport_pressure = 74682.287  #[Pa]
322     airport_density = airport_pressure/(R_air*airport_temperature)
        #[kg/(m^3)]
323     sea_level_temperature = 288.15 +dTisa
324     sea_level_pressure = 101325
325     sea_level_density = sea_level_pressure/(R_air*
        sea_level_temperature)
326     density_ratio = airport_density/sea_level_density
327     speed_stall = speed_app/1.3
328     speed_stallQ400 = 120*knot2ms/1.3
329     WS_Q400 = 453 # Wing loading for Q400
330     C_Lmax = WS_Q400*2*g_0/(sea_level_density*speed_stallQ400**2)
331
332     # Calculate wing loading at stall
333
334     WS_stall = (1/(2*g_0))*sea_level_density*(speed_stall**2)*
        C_Lmax  # Unsure about the maximum lift factor
335
336     # Calulate wing loading for take-off
337     Takeoff_parameter = 454 # fps units (prop), see page 130
338     C_LTO = C_Lmax/1.21
339
340     # The take-off requirement is given for sea level flight, so
        sigma should be 1
341     # Create a function for the wing loading. (used later to make a
```

```
    plot)
342 def WS_takeoff_function(TOP, PW, sigma):
343     PW_fps = PW*w2hp/(kg2lb)
344     WS_takeoff_fps = TOP*sigma*C_LTO*PW_fps
345     WS_takeoff = WS_takeoff_fps*(lb2kg/(ft2m**2))
346     return WS_takeoff
347
348 WS_takeoff = WS_takeoff_function(Takeoff_parameter, PW_takeoff,
    1)
349
350 # Caluclate wing loading for landing
351
352 WS_landing = (LFL/1.43 -305)*(1*C_Lmax)/4.84 # value becomes
    very low when an altitude pressure ratio is used instead of 1,
    are the requirements set at sea level?
353 WS_landing = WS_landing/(mass_landing1/w0) # Wing loading at
    landing conditions
354 # Caluclate wing loading for climb
355
356 G = speed_vertical/speed_cruise
357 e = 0.85*0.95
358 C_D0 = 0.015+0.002
359 q = (0.5*sea_level_density*speed_cruise**2)
360
361 def WS_climb_function(PW):
362     TW = PW*((efficiency_prop/speed_cruise)*(1/g_0))
363     denominator = 2/(q*np.pi*aspect_ratio*e)
364     root_expression = np.sqrt((TW-G)**2 - (4*C_D0/(np.pi*
    aspect_ratio*e)))
365     WS_climb_plus = (TW-G)/denominator + root_expression/
    denominator
366     WS_climb_minus =  (TW-G)/denominator - root_expression/
    denominator
367     WS_climb = np.minimum(WS_climb_plus, WS_climb_minus)
368     return WS_climb
369
370 # Caluclate propeller and nacelle size
371 # Propeller size
372 ## Statistical method
373
374 # Looking at similar aircraft, the number of blades is usually
    4 or more, so this statistical correlation will be used:
375 PW_installed = max(PW_climb, PW_takeoff, PW_cruise)
376
377 prop_diameter_stat = 0.49*((PW_installed*w0)/1000/2)**(1/4) #
    Dicided P/W by two since we only have one engine
378
379 ## Method based on limiting Mach number
380 N_cruise = 850/60 # Maximum revolutions per second during
    cruise, taken from Dash-8 Q400 manual
381 N_climb = 900/60 # Maximum revolutions per second during climb,
     taken from Dash-8 Q400 manual
382 N_takeoff = 1200/60 # Maximum revolutions per second during
    climb, taken from Dash-8 Q400 manual
383 prop_diameter_cruise = np.sqrt((gamma_air*R_air*T_isa_cruise-
    speed_cruise**2)/((np.pi**2)*(N_cruise**2)))
384 prop_diameter_climb = np.sqrt((gamma_air*R_air*
```

```
    sea_level_temperature-speed_cruise**2)/((np.pi**2)*(N_climb**2)
    ))
    prop_diameter_takeoff = np.sqrt((gamma_air*R_air*
    sea_level_temperature-(speed_cruise)**2)/((np.pi**2)*(N_takeoff
    **2))) # Used cruise speed to produce very conservative value

    prop_diameter =min(prop_diameter_stat, prop_diameter_cruise,
    prop_diameter_climb, prop_diameter_takeoff) #[m]

    #Calculate engine power, and wing reference area
    P_installed = (w0/2)*(PW_climb)
    wing_reference_area = w0/WS_stall

    ## Calculate tail wing size according to Raymer
    # fuselage_length = 38 #[m] #Guessed value
    # moment_arm = 0.5*fuselage_length # 50-55% taken from Raymer (
    p.160)
    # b = 30/2
    # MAC = 2.4 #[m]# Mean aerodynamic chord
    # c_HT =0.9*0.95 # volume coefficients, taken from Raymer (p
    .160)
    # c_VT =0.08*0.95

    # vertical_tail_reference_area = c_VT*b*wing_reference_area/
    moment_arm
    # horizontal_tail_reference_area = c_HT*MAC*wing_reference_area
    /moment_arm

    # Calculate tail wing size according to Snorri, using values
    for Dash-8 Q400
    V_HT = 1.769
    V_VT = 0.1252
    c_MGC = 2.3
    moment_arm = 17.8
    horizontal_tail_reference_area = V_HT*wing_reference_area*c_MGC
    /moment_arm
    vertical_tail_reference_area = V_VT*wing_reference_area*
    wing_span/moment_arm
    # Tire sizing
    # Amount of weight on the main landing gears
    w_main_gear = 0.85*w0
    # Amount of weight on each wheel
    n_wheel_main = 4# amount of wheels on the main landing gear
    n_wheel_nose = 2# amount of wheels on nose gear
    W_wheel_main = w_main_gear/n_wheel_main
    # Increase values by 30% if we want the plane to operate on
    rough, unpaved runways
    main_wheel_diameter = 5.3*(W_wheel_main)**0.315 #[cm]
    main_wheel_width = 0.39*(W_wheel_main)**0.480 #[cm]

    # Calculate forces on the wheels using Raymer's definition
    N_a = 16.64 #[m]
    N_f = 15.6 #[m]
    M_a = 1.96 #[m]
    M_f = 3 #[m]
    B = 18.6 #[m]
    H = 2.71 #[m]
```

```
429
430     max_static_load_main = (w0/n_wheel_main)*(N_a/B)*g_0
431     max_load_main = max_static_load_main*1.07
432     max_static_load_nose = (w0/n_wheel_nose)*(M_a/B)*g_0
433     max_dynamic_load_nose = ((3*H*w0/n_wheel_nose)/B)
434     max_load_nose = (max_static_load_nose + max_dynamic_load_nose)
        *1.07
435     max_load_main_fps = max_load_main/4.448
436     max_load_nose_fps = max_load_nose/4.448
437     # Main gear: Three-part name 28x9.00-12
438     # Nose gear: Type VII 24x5.5
439
440 # Print results
441 if i == 0:
442     print("n = 1, Initial sizing method")
443 if i > 0:
444     print("n = 2, Refined sizing method")
445 print("---RESULTS---")
446 print("Maximum crew and passenger mass: " + str(round(w_crew +
        w_PAX, 4)) + " [kg]")
447 print("Total fuel mass: " + str(round(w_fuel, 4)) + " [kg]")
448 print("Tank mass: " + str(round(w_tank, 4)) + " [kg]")
449 print("Empty mass: " + str(round(w_empty, 4)) + " [kg]")
450 print("Maximum take-off weight: " + str(round(w0, 4)) + " [kg]" )
451 print("OEW/MTOW: " + str(round(OEW_MTOW, 4)))
452 print("Total fuel volume: " + str(round(volume_fuel, 4)) + " [m^3]"
      )
453 print("Maximum L/D: " + str(round(lift_drag_ratio_max, 4)))
454 print("Cruise SFC: " + str(round(SFC_cruise1*10**6, 4))+ " [mg/(Ns)
    ]")
455 print("Loiter SFC: " + str(round(SFC_loiter*10**6, 4))+ " [mg/(Ns)]
    ")
456 print("Propeller diameter " + str(round(prop_diameter, 4))+ " [m]")
457 print("W/S installed: " +str(round(WS_stall, 4))+ " [kg/m^2]")
458 print("P/W installed: " +str(round(PW_installed/1000, 4))+ " [kW/kg
    ]" )
459 print("Single engine power: " +str(round(P_installed*w2hp, 4))+ " [
    HP]" )
460 print("Wing reference area: " +str(round(wing_reference_area, 4))+
    " [m^2]" )
461 print("Horizontal tail reference area: " +str(round(
    horizontal_tail_reference_area, 4))+ " [m^2]" )
462 print("Vertical tail reference area: " +str(round(
    vertical_tail_reference_area, 4))+ " [m^2]" )
463 print("Number of main landing gear wheels: " +str(n_wheel_main))
464 print("Main landing gear wheel diameter: " +str(round(
    main_wheel_diameter/100, 4))+ " [m]" )
465 print("Main landing gear wheel width: " +str(round(main_wheel_width
    /100, 4))+ " [m]" )
466 print("Front landing gear wheel diameter: " +str(round(0.6*
    main_wheel_diameter/100, 4)) +"-"+str(round(main_wheel_diameter
    /100, 4))+ " [m]" )
467 print("Front landing gear wheel width: " +str(round(0.6*
    main_wheel_width/100, 4)) +"-"+str(round(main_wheel_width/100,
    4))+ " [m]" )
468 print("-----------------")
469
```

```
470
471 # Plot performance map
472 PW_start = 0
473 PW_stop = PW_max*1.5
474 PW_points = np.linspace(PW_start, PW_stop, num = 100)
475 WS_takeoff_array = np.zeros(len(PW_points))
476 for i in range(1, len(PW_points)):
477     WS_takeoff_array[i] =  WS_takeoff_function(Takeoff_parameter,
    PW_points[i], 1)
478
479
480
481 plt.figure(dpi = 300)
482 plt.autoscale(enable=True, axis='x', tight=True)
483 plt.autoscale(enable=True, axis='y', tight=True)
484 plt.xlabel(r'$\frac{W}{S}$ [$kg/m^2$]')
485 plt.ylabel(r'$\frac{P}{W}$ [$kW/kg$]')
486 plt.title('Constraint diagram')
487
488
489 plt.axhline(y=PW_climb/1000, color='b', linestyle='-', label= '
    Climb P/W', linewidth=3 )
490 # Fill below Climb P/W line
491 plt.fill_between([0, 1250], PW_climb/1000, color='b', alpha=0.2)
492
493
494 #plt.axhline(y=PW_takeoff/1000, color='g', linestyle='-', label = '
    Take-off P/W', linewidth=3)
495 plt.axhline(y=PW_cruise/1000, color='c', linestyle='-', label = '
    Cruise P/W', linewidth=3)
496 plt.axvline(x = WS_stall, color = 'r', linestyle='-', label = '
    Stall W/S', linewidth=3)
497 plt.axvline(x = WS_landing, color = 'y', linestyle='-', label = '
    Landing W/S', linewidth=3)
498 # Fill to the right of Stall W/S line
499 plt.fill_betweenx([0, PW_stop/1000], WS_stall, 1250, color='r',
    alpha=0.2)
500
501 plt.plot(WS_takeoff_array, PW_points/1000,'g', label = 'Take-off',
    linewidth=3)
502 plt.plot(WS_stall, PW_climb/1000, marker="X", markersize=15,
    markeredgecolor="black", markerfacecolor="red",label ='Design
    location', linestyle="None")
503 plt.legend(loc = "lower right")
```

B

Appendix 2

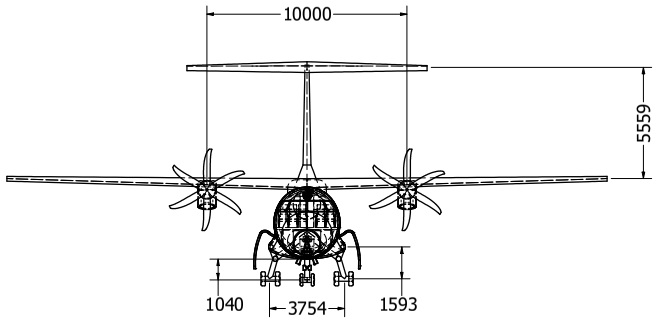
B.1 Case study aircraft Component weight breakdown

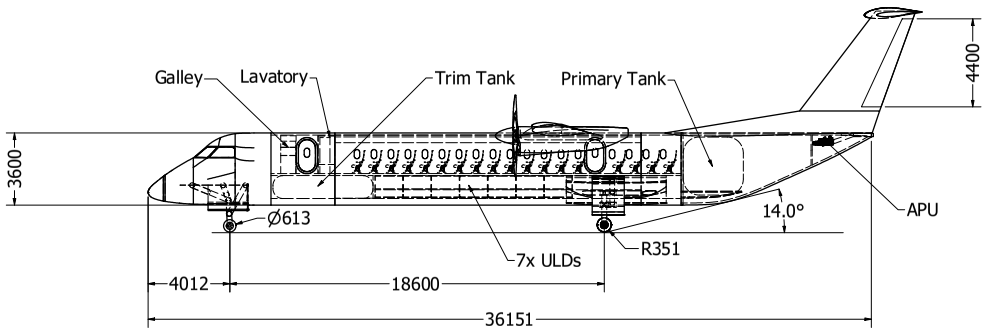
Category	Component	Weight (kg) / % of Total
Capacity	**Passengers and luggage**	10354.0 kg / 34.44%
Equipment	**Flight controls**	142.65 kg / 0.47%
	APU installed	189.20 kg / 0.63%
	Instruments	69.53 kg / 0.23%
	Hydraulics	75.00 kg / 0.25%
	Electrical	44.11 kg / 0.15%
	Avionics	560.34 kg / 1.86%
	Furnishing	666.61 kg / 2.22%
	Air conditioning	470.70 kg / 1.57%
	Anti-ice	61.05 kg / 0.20%
Operational Items	**Unit Load Device**	665 kg / 2.21%
	Lavatory	55.63 kg / 0.19%
Propulsion	**Engine controls**	37.96 kg / 0.13%
	Starter	41.56 kg / 0.14%
	Fuel system	47.38 kg / 0.16%
	Installed engine	1868.53 kg / 6.22%
	Tank	3334.60 kg / 11.09%
	Fuel	1795.50 kg / 5.97%
Structure	**Wing**	2253.94 kg / 7.50%
	Horizontal Tail	155.21 kg / 0.52%
	Vertical Tail	430.97 kg / 1.43%
	Fuselage	4905.76 kg / 16.32%
	Main Landing Gear	1026.86 kg / 3.42%
	Nose Landing Gear	237.33 kg / 0.79%
	Nacelle Group	571.58 kg / 1.90%
Summation	**Total**	30061 kg / 100%

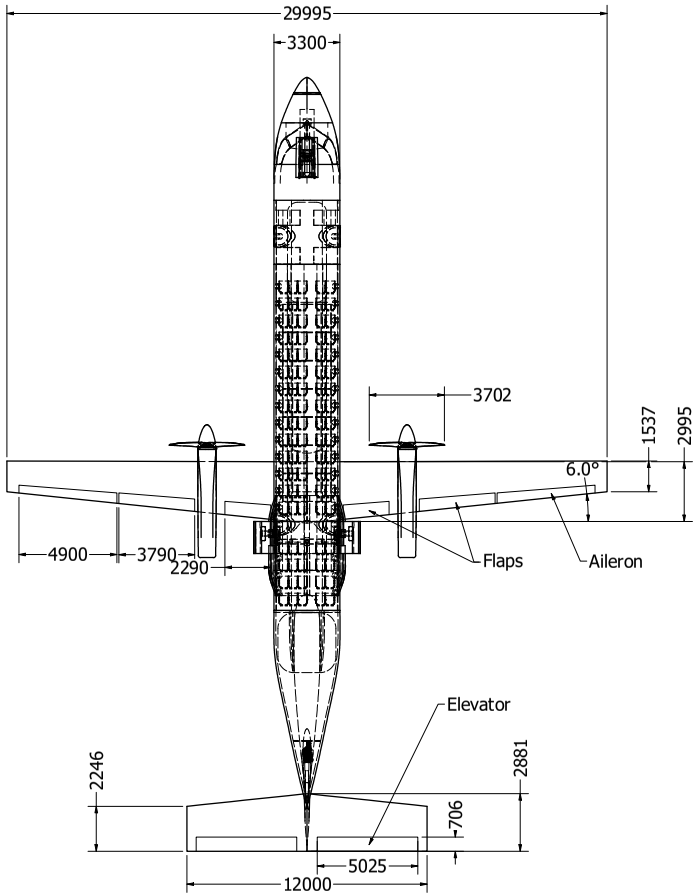
C

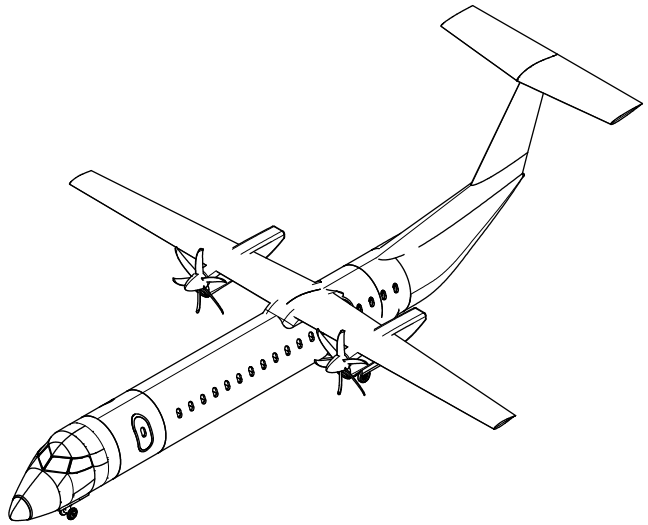
Appendix 3

C.1 Case study aircraft Drawing









DRAWN Michael Crona	17/10/2023	Chalmers
CHECKED Per Samuelsson		TITLE: Dash 1 - Group 6
QA		
MFG		
APPROVED		
		SIZE D · DWG NO 1 · REV 7
		SCALE 1 / 130 · SHEET 1 OF 1

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY