



UNIVERSITÁ DEGLI STUDI DI GENOVA

---

SCUOLA POLITECNICA

Master's Degree in Mechanical Engineering - Energy and Aeronautics

CURRICULUM AERONAUTICS

**A Weight and Balance evaluation software for  
conventional and unconventional aircraft design**

—

MASTER'S THESIS

**SUPERVISORS:**  
**Prof. A. BOTTARO**  
**Dr. J. B. VOS**

**CO-SUPERVISORS:**  
**Eng. A. JUNGO**

**AUTHOR:**  
**STEFANO PICCINI**

ACADEMIC YEAR 2018/2019





## **Software per la valutazione di peso e bilanciamento per progettazione preliminare di velivoli convenzionali e non convenzionali**

### **Sommario**

La presente tesi si inquadra nell'ambito della progettazione concettuale e preliminare di velivoli convenzionali e non convenzionali. Il lavoro si propone di sviluppare un codice Python finalizzato alla valutazione del peso e del bilanciamento di velivoli, una componente cruciale del programma CEASIOMpy. Questo programma analizza la geometria di qualsiasi tipologia di aereo al fine di velocizzare e semplificare il processo di design preliminare e concettuale. Nello specifico, i moduli sviluppati nel contesto della presente tesi mirano a stimare le masse più rilevanti di un aeromobile partendo da una geometria semplificata. Il primo modulo analizza esclusivamente velivoli convenzionali e si basa sul metodo della regressione lineare per la valutazione dei pesi. Il modulo non convenzionale, invece, si basa su un processo iterativo ed una più dettagliata analisi delle masse per valutare aeroplani con geometrie sia convenzionali che non convenzionali. I moduli per la stima del bilanciamento sfruttano, per ogni tipologia di geometria, un metodo a masse concentrate al fine di stimare efficacemente centro di gravità e momenti di inerzia. In conclusione, il codice sviluppato comprende anche una valutazione dell'autonomia e del consumo di combustibile per ogni tipologia di aeroplano che può essere utilizzata per velocizzare le primarie fasi di progettazione o come controllo secondario per la valutazione del peso del velivolo. I moduli sono stati testati con molteplici aeroplani convenzionali e non; alcuni di questi sono velivoli commerciali ancora utilizzati oggi, altri sono stati ritirati dal commercio o rappresentano i nuovi modelli ideati durante il progetto AGILE. I moduli stimano tutte le quantità con sufficiente precisione, possono lavorare indipendentemente ed essere usati all'interno di un processo di ottimizzazione.



## **A Weight and Balance evaluation software for conventional and unconventional aircraft design**

### **Abstract**

This thesis deals with aircraft conceptual and preliminary design. It focuses on the development of aircraft Weight and Balance evaluation modules in Python, a crucial element of the CEASIOMpy software. This software aims to analyse all kind of aeroplanes geometry to simplify and speed up the conceptual and preliminary stages of the design process. In particular, the modules developed in this thesis framework aims to estimate the aircraft most crucial weights starting from the primary geometric elements. The first module analyses only conventional aeroplanes and uses a linear regression method for weight estimation. The unconventional module, instead, relies on an iteration process and a more detailed mass analysis to evaluate both conventional and unconventional geometries. The balance modules use, for all kind of geometry, a lumped masses approach to estimate the aircraft centre of gravity and moments of inertia. In the end, the software developed also comprehends a range and fuel consumption analysis for all kinds of aeroplanes and can be used to speed up the early design phases or as a secondary check for the weight estimation. The modules have been tested with several conventional and unconventional aircraft. Some of them are current or former commercial airliners, and others are new aeroplanes that are being developed alongside the AGILE project. The software modules estimate all the quantity with reasonable accuracy; they can work independently and can be used alongside an optimisation process.

# Dedication and Acknowledgements

First of all, I sincerely thank my supervisor and mentor, Jan B. Vos, for the opportunity he has granted to me to work at CFS Engineering, experimenting working life outside the university for six months away from home; he has always steered me in the right direction to carry out my thesis successfully. I would like to thank my co-supervisor, Aidan Jungo, to have bored with me, supported and taught me, day after day, with commitment; also thanks to my colleagues, Malo Drougard, Dominique Charbonnier and Alain Gehri for the time spent together at work. A special thank you to my supervisor and mentor, Alessandro Bottaro, for having guided me throughout my academic years. My thanks also goes to all professors of the Curriculum *Aeronautics* for their teachings that had a vital role in the development of my thesis.

I must express my very profound gratitude to my parents, Daniele and Eliana, and all my family that allowed my studies and experiences abroad in these years and for their unfaltering support; thanks also to my friends and colleagues, Jacopo and Matteo, who have shared with me all the happy and sad moments of this fantastic journey. Finally, a special thank you goes to my girlfriend, Martina, for all her love and support. This accomplishment would not have been possible without them.

# Contents

<b>Sommario</b>	<b>ii</b>
<b>Abstract</b>	<b>iii</b>
<b>Dedication and Acknowledgements</b>	<b>iv</b>
<b>Abbreviations</b>	<b>xii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Aircraft design process . . . . .	1
1.1.1 Conventional and unconventional aircraft differences . . . . .	3
1.2 CEASIOMpy software . . . . .	4
1.3 Purpose of the project . . . . .	6
<b>2 The physics behind</b>	<b>8</b>
2.1 Aircraft weights . . . . .	8
2.2 Centre of Gravity and Moments of Inertia . . . . .	10
2.2.1 Centre of Gravity . . . . .	10
2.2.2 Moments of Inertia . . . . .	10
2.3 Range and flight phases . . . . .	12
2.3.1 Breguet range . . . . .	13
2.3.2 Flight phases . . . . .	14
<b>3 Weight and Balance software</b>	<b>16</b>
3.1 Inputs and work-flows . . . . .	16
3.1.1 Conventional aircraft module . . . . .	17
3.1.2 Unconventional aircraft module . . . . .	21
3.1.3 Range module . . . . .	25
3.2 Geometry acquisition . . . . .	27
3.2.1 Conventional geometry . . . . .	29
3.2.2 Unconventional geometry . . . . .	30
3.2.3 Unconventional geometry without fuselage . . . . .	31
3.3 Weight analysis for conventional aircraft . . . . .	33
3.3.1 Maximum take-off mass . . . . .	33
3.3.2 Operating empty mass . . . . .	34
3.3.3 Passengers, crew members analysis . . . . .	37
3.3.4 Fuel mass . . . . .	41

3.4	Weight analysis for unconventional aircraft . . . . .	42
3.4.1	Structure, payload and fuel masses . . . . .	43
3.4.2	Crew members, engines and systems masses . . . . .	46
3.5	Range module . . . . .	49
3.5.1	Fuel Consumption . . . . .	49
3.5.2	Range evaluation . . . . .	50
3.5.3	Crew updating . . . . .	51
3.6	Balance module . . . . .	52
3.6.1	Centre of Gravity . . . . .	52
3.6.2	Moments of Inertia . . . . .	53
3.6.3	Differences in the unconventional balance module . . . . .	54
<b>4</b>	<b>Results</b>	<b>58</b>
4.1	Conventional aircraft testing . . . . .	58
4.2	Effect of the wing position on the balance analysis . . . . .	59
4.3	Conventional aircraft analysis . . . . .	65
4.3.1	ATR72® . . . . .	65
4.3.2	Boeing B777® . . . . .	70
4.3.3	WP46RM . . . . .	75
4.4	Semi-Conventional Box-wing AGILE analysis . . . . .	79
4.5	Unconventional aircraft analysis . . . . .	84
4.5.1	Blended Wing Body . . . . .	84
4.5.2	Concorde® . . . . .	87
<b>5</b>	<b>Conclusions</b>	<b>91</b>
5.1	Perspectives . . . . .	91
<b>A</b>	<b>Linear regression method</b>	<b>94</b>
<b>B</b>	<b>Weight and Balance software</b>	<b>96</b>
B.1	Aircraft Database . . . . .	96
B.2	Weight and Balance modules input data . . . . .	98
	<b>Bibliography</b>	<b>104</b>

# List of Figures

1.1	Aircraft design process work-flow [3]. . . . .	2
1.2	Benefits of the virtual product in aircraft design [5]. . . . .	3
1.3	Pictures of different aircraft configurations: conventional (1.3a), semi-conventional (1.3b) and unconventional (1.3e, 1.3c, 1.3d). . . . .	5
1.4	CEASIOMpy current development schema. . . . .	6
2.1	Drawing of the centroidal axis $I_o$ through the centre of gravity (CG) and relative remote axes $(x_1, x_2)$ for a simple geometry. . . . .	11
2.2	Example of an aircraft geometry discretized with lumped masses. . . . .	12
2.3	Sketch of an aircraft forces during a levelled flight. . . . .	13
2.4	Sketch of the common phases of a commercial flight [21]. . . . .	15
3.1	Weight (3.1a) and Balance (3.1b) conventional modules work-flow chart. . . . .	19
3.2	Unconventional weight module work-flow chart. . . . .	23
3.3	Unconventional balance module work-flow chart. . . . .	24
3.4	Range module work-flow chart. . . . .	27
3.5	Representation of the sections and segments used to build the ATR72 geometry, the yellow wire-frames are relative to the mirrored wings. . . . .	29
3.6	Cabin area polygon example. . . . .	32
3.7	Example of the linear regression results plotted for the Airbus A319 <sup>®</sup> . . . . .	35
3.8	Possible seat configuration defined by the software for an Airbus A319 <sup>®</sup> with 6 abreasts (right), and the real configuration (left). . . . .	39
3.9	Possible seat configuration defined by the software for an Airbus A330 <sup>®</sup> with 8 abreasts long (right), and a real configuration (left). . . . .	40
3.10	Possible seat disposition for the Blended Wing Body aircraft, view of half of the cabin area, cockpit on the right of the image (Blue rectangle = seats, white rectangle = empty space). . . . .	45
3.11	Engine mass variation in relation with the maximum available thrust for turbofan (left) and with the maximum output power produced for turbo-prop engines (right). . . . .	47
3.12	Example of the range vs payload plot from the range module (A319 <sup>®</sup> aircraft tested). . . . .	51
3.13	Geometry of the ATR72 aircraft (Left) with the lumped masses disposition for CG evaluation (Right). . . . .	53
3.14	Error of the evaluated moments of inertia for the test cylinder (tab.:3.8) against the number of lumped nodes used. . . . .	55



3.15	Example of the lumped masses distribution with a spacing of 2m for the B777 (upper) and the BWB aircraft (lower). . . . .	56
4.1	Centre of gravity variation in relation with the main wing position. . . . .	64
4.2	ATR72 aircraft analysis, full aircraft (4.2a), centre of gravity (4.2b) and range versus payload (4.2c). . . . .	67
4.3	B777 <sup>®</sup> aircraft analysis, full aircraft (4.3a), centre of gravity (4.3b) and range versus payload (4.3c). . . . .	72
4.4	WP46RM aircraft analysis, full aircraft (4.4a), range versus payload (4.4b, 4.4c) and centre of gravity (4.4d, 4.4e). . . . .	77
4.5	Box-Wing aircraft analysis, full aircraft (4.5a), range versus payload (4.5b, 4.5c) and centre of gravity (4.5d, 4.5e). . . . .	81
4.6	Boxwing aircraft OEM variation against the virtual thickness (4.6a), the passengers density (4.6b) and the LD coefficient (4.6c). . . . .	83
4.7	Blended Wing Body aircraft analysis, full aircraft (4.7a, 4.7b), centre of gravity (4.7c) and range versus payload (4.7d). . . . .	85
4.8	Concorde <sup>®</sup> aircraft analysis, full aircraft (4.8a), centre of gravity (4.8b) and range versus payload (4.8c). . . . .	88
A.1	Linear regression plot example. . . . .	95

# List of Tables

2.1	Aircraft masses with their respective group of belongings. . . . .	9
2.2	SFC and propeller SFC [1/hr] in cruise and loiter in comparison with the type of engines [18]. . . . .	14
2.3	Flight phases weight fractions. . . . .	15
3.1	Conventional aircraft main geometry data extracted from the CPACS file with relative unity of measure (UoM). . . . .	28
3.2	Unconventional Aircraft extra geometry data extracted. . . . .	28
3.3	List of the most common coefficients used for the evaluation of conventional aircraft OEM listed by Raymer [18]. . . . .	36
3.4	Default value for the conventional aircraft seats analysis in relation with the aircraft geometry. . . . .	38
3.5	Coefficients for the definition of the maximum fuel mass in relation with the MTOM (values related to the conventional aircraft analysis only when the CPACS geometry is not used). . . . .	42
3.6	Example of wing loading value for aircraft type [35]. . . . .	43
3.7	Passengers density variation with aircraft type and cabin area. . . . .	45
3.8	Analysis of the effect of different number of points for the evaluation of the moments of inertia [ $\text{kgm}^2$ ] of a test cylinder ( $R = 5\text{m}$ , $L = 12\text{m}$ , $M = 200\text{kg}$ ). . . . .	55
4.1	Results from the Airbus <sup>®</sup> aircraft analysis using $l_{fg}$ , $WDT_{fg}$ , $b$ , $S_w$ dimensions as data input, compared with the real value obtained from the manuals for airport planning (1 of 2). . . . .	60
4.2	Results from the Airbus <sup>®</sup> aircraft analysis using $l_{fg}$ , $WDT_{fg}$ , $b$ , $S_w$ dimensions as data input, compared with the real value obtained from the manuals for airport planning (2 of 2). . . . .	61
4.3	Result from the Boeing <sup>®</sup> aircraft analysis using $l_{fg}$ , $WDT_{fg}$ , $b$ , $S_w$ dimensions as data input, compared with the real value obtained from the manuals for airport planning. . . . .	62
4.4	Result from different aircraft analysis using $l_{fg}$ , $WDT_{fg}$ , $b$ , $S_w$ dimensions as data input, compared with the real value obtained from the manuals for airport planning. . . . .	62
4.5	NASA CRM weights estimation. . . . .	63
4.6	Moments of inertia and centre of gravity variation in relation with the main wing position. . . . .	65
4.7	CWB and UWB modules parameters defined for the ATR72 <sup>®</sup> analysis. . .	68

4.8	Conventional and unconventional balance analysis comparison for the ATR72 <sup>®</sup> aircraft. . . . .	68
4.9	Results comparison for the ATR72 <sup>®</sup> weight analysis with the conventional and unconventional WB software. . . . .	69
4.10	Results comparison for the ATR72 <sup>®</sup> range analysis with the conventional and unconventional WB software. . . . .	69
4.11	CWB and UWB modules parameters defined for the B777 <sup>®</sup> analysis. . . . .	73
4.12	Results comparison for the B777 <sup>®</sup> weight analysis with the conventional and unconventional WB software. . . . .	74
4.13	Results comparison for the B777 <sup>®</sup> range analysis with the conventional and unconventional WB software. . . . .	74
4.14	Conventional and unconventional balance analysis comparison for the B777 <sup>®</sup> aircraft. . . . .	75
4.15	CWB and UWB modules parameters defined for the WP46RM analysis. . . . .	76
4.16	WP46RM aircraft conventional and unconventional weight analysis comparison. . . . .	78
4.17	WP46RM aircraft conventional and unconventional range analysis comparison. . . . .	78
4.18	Balance analysis comparison for the WP46RM aircraft with the conventional and unconventional WB software. . . . .	79
4.19	Results comparison for the Box-Wing aircraft weight analysis with the conventional and unconventional WB software. . . . .	82
4.20	Results comparison for the Box-Wing aircraft range analysis with the conventional and unconventional WB software. . . . .	82
4.21	Balance analysis comparison for a Box-Wing aircraft with the conventional and unconventional WB software. . . . .	83
4.22	Parameters defined for the Blended Wing Body analysis with the unconventional code. . . . .	86
4.23	Results for the BWB weight and range analysis with the unconventional WB software. . . . .	86
4.24	Results for the BWB balance analysis with the unconventional WB software. . . . .	86
4.25	Parameters defined for the Concorde <sup>®</sup> analysis with the unconventional WB software. . . . .	89
4.26	Results for the Concorde <sup>®</sup> weight analysis with the unconventional WB software. . . . .	89
4.27	Results for the Concorde <sup>®</sup> range analysis with the unconventional WB software. . . . .	90
4.28	Results for the Concorde <sup>®</sup> balance analysis with the unconventional WB software. . . . .	90
B.1	Example of the aircraft database structure. . . . .	96
B.2	Input data for the balance analysis of conventional aircraft. . . . .	98
B.3	Input data for the weight analysis of conventional aircraft. . . . .	99
B.4	Input data for the unconventional weight analysis. . . . .	100
B.5	Input data for the unconventional balance analysis. . . . .	101
B.6	Input data for the engine definition. . . . .	102

B.7 Input data for the range analysis. . . . . 103

# Abbreviations

Acronyms	Description
AGILE	Aircraft 3rd Generation MDO for Innovative Collaboration of Heterogeneous Teams of Experts
APU	Auxiliary Power Unit
BWB	Blended Wing Body
CEASIOMpy	Computerised Environment for Aircraft Synthesis and Integrated Optimisation with Python
COG	Centre of Gravity
CFSE	Computational Fluid and Structure Engineering
CPACS	Common Parametric Aircraft Configuration Schema
CWB	Conventional Weight and Balance
DLR	Deutsches Zentrum für Luft und Raumfahrt (German Space Agency)
EASA	European Aviation Safety Agency
FAA	Federal Aviation Administration
ICAO	International Civil Aviation Organisation
MAC	Mean Aerodynamic chord
MDO	Multidisciplinary Optimisation
MOI	Moments of Inertia
MTOM	Maximum Take Off Mass
OEM	Operating Empty Mass
RES	Unusable fuel Reserve
SH	Single Hydraulics
SimSAC	Simulating Stability And Control Characteristics for Use in Conceptual Design
SFC	Specific Fuel Consumption
TLAR	Top Level Aircraft Requirement
TP	Turboprop
TSFC	Thrust Specific Fuel Consumption
UoM	Unity of Measure
UWB	Unconventional Weight and Balance
VP	Virtual Product
WB	Weight and balance
WL	Wing Loading
WM	Wing Mounted
WT	Winglet Type
XML	Extensible Markup Language
ZFM, ZPM	Zero Fuel Mass, Zero Payload Mass

<b>Symbols</b>	<b>Description</b>	<b>Unit</b>	<b>Subscripts</b>	<b>Description</b>
$A$	Wetted area	[m <sup>2</sup> ]	acl	After climb
$b$	Wing span	[m]	acru	After cruise
$b_c$	Cabin span	[m]	aloi	After loiter
$C$	Specific Fuel Consumption	[1/hr]	ato	After take-off
$C_{bhp}$	Propeller SFC	[lb/(hp · hr)]	c	Cabin
$CG_{x,y,z}$	Centre of gravity coordinates	[m]	cru	Cruise
$D$	Drag force	[N]	e	Engine
$g$	Acceleration of gravity	[m/s <sup>2</sup> ]	f	Fuel
$h_{cabin}$	Cabin maximum height	[m]	ffg	Fuel inside fuselage
$h_{loiter}$	Loiter duration	[min]	fg	Fuselage
$I_{xx}$	Roll moment of inertia	[kg · m <sup>2</sup> ]	fw	Fuel inside wing
$I_{yy}$	Pitch moment of inertia	[kg · m <sup>2</sup> ]	lav	Lavatory
$I_{zz}$	Yaw moment of inertia	[kg · m <sup>2</sup> ]	max	Maximum
$L$	Lift Force	[N]	mp	Maximum payload
$l$	Length	[m]	MTO	Maximum Take-Off
$LD$	Lift over drag coefficient	[-]	mVRT	Mean structure virtual thickness
$M$	Mass	[kg]	mw	Main wing
$M_{cargo}$	Cargo mass	[kg]	OE	Operating Empty
$M_{structure}$	Structure mass	[kg]	pax	Passengers
$Nb$	Number	[-]	st	Seat
$Nb_{abreasts}$	Number of abreasts	[-]	str	Structure
$Nb_{aisles}$	Number of aisles	[-]	TO	Take-Off
$Nb_{pilots}$	Number of pilots	[-]	w	Wing
$P_{max}$	Maximum output power	[kW]	wsec	Wing section
$R$	Range	[km]	x	x-coordinate
$S_{fg}$	Fuselage frontal area	[m <sup>2</sup> ]	y	y-coordinate
$S_w$	Wing planform area	[m <sup>2</sup> ]	z	z-coordinate
$T$	Thrust	[kN]	ZF	Zero Fuel
$t_{mVRT}$	Structure virtual thickness	[-]	ZP	Zero Payload
$tp$	Taper ratio	[-]		
$V_{cru}$	Cruise velocity	[m/s]		
$Vol$	Volume	[m <sup>3</sup> ]		
$W$	Weight	[N]		
$WDT$	Width	[m]		
$\rho_f$	Fuel density	[kg/m <sup>3</sup> ]		
$\rho_{lav}$	Density of passengers per lavatory	[pax/lav]		
$\rho_{mVRT}$	Structure mean virtual density	[kg/m <sup>3</sup> ]		
$\rho_{pax}$	Passengers surface density	[pax/m <sup>2</sup> ]		
$\eta_{prop}$	Propeller efficiency	[-]		



# Chapter 1

## Introduction

After a decade of near stagnation, aircraft traffic in Europe is growing again exceeding 10 million flights per year. The EUROCONTROL organisation and the ICAO (International Civil Aviation Organisation) estimate that in 2040 the number of yearly flights in Europe will reach between 16.2 and 19.5 million [1]. Globally the number of passengers will grow up to 8 billion per year by 2040, and the AIRBUS company forecasts a total market of 37000 new aircraft (including replacement of 11000 old aeroplanes) [2] to be delivered until 2040.

In this scenario, new aircraft will have to be characterised by reduced fuel consumption, operating costs, and environmental impact. Nowadays all the aircraft can be classified in one category (conventional aircraft) since they have the same known base (fuselage with wings) and the improvements that can be made to reduce fuel consumption, costs and emissions are reaching their asymptotic limit. To satisfy the requirements of reduced fuel consumption and environmental impact, and to fulfil passenger demand in the next decades new unconventional aircraft, like the Blended Wing Body (BWB) or the Prandtl-Plane, have to be built alongside the most reliable and efficient models that are leading the market today. However, the current design process is not adapted for unconventional aircraft because it uses knowledge coming from past conventional aircrafts' designs that have been translated into empirical relations.

### 1.1 Aircraft design process

The design process work-flow (fig.:1.1) consists of 3 main phases that precede the manufacturing and testing of an aeroplane [3]: the conceptual design, the preliminary design and the detailed design.

Starting from the Top Level Aircraft Requirements (TLAR, e.g. desired performance, costs, range, payload, number of passengers, cruise speed, engines) the conceptual design phase starts. This is an iterative process in which many aircraft variants are studied. For each variant the weight, balance, aerodynamics, performance are studied and the concept that fulfils the best the TLAR requirements is selected. This concept is then further refined by analysing constraints related to systems, engines, structures [4].

The preliminary design phase is also an iterative process that starts from the refined aircraft. The concept is further studied, the aerodynamic shape and main structures are defined in more detail, and reliable performance estimates are made. Design sensitivi-



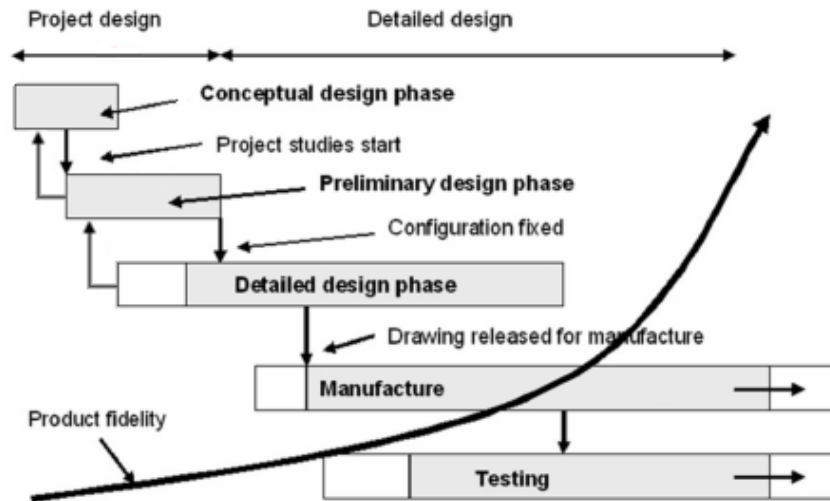


Figure 1.1: Aircraft design process work-flow [3].

ties are generated, and near the end of this phase, the concept is frozen, meaning that no significant changes in the design are expected (except when a significant problem is detected).

When the detailed design phase begins, the aircraft design receives elements such as the interior design, detailed control surfaces, safety equipment and all the other components that are designed separately with their final configuration (suspensions, landing gears, fuel tanks). In this phase, the manufacturing process is taken into account to minimise the cost and time needed to build the aircraft. The last phase allows the complete manufacturing and testing of the aeroplane, achieving the highest product fidelity. The design process reaches its end when the tested aircraft meets all the requirements both from the customer and from the safety and environmental regulations.

The aircraft design process depends to a large extent on the conceptual and preliminary design phases that can last for many years. To speed up the overall process, the design loops are becoming more interconnected with the idea to employ *Multidisciplinary Design Optimisation* (MDO) techniques in the conceptual and preliminary design phases. Coupled with this is the trend to move away from empirical or simplified approaches towards methods that better represent the physics of the problem being studied. Numerical simulation has become an essential element in the aircraft design, reducing risks, time and costs of the design process.

The *Virtual Product* (VP) is defined as a high-fidelity mathematical/numerical representation of the physical properties and the function of the product. Through increased fidelity, the VP allows improving the product earlier in the design process. As can be seen from figure 1.2, the traditional design process has the majority of the development time spent on ground facility and flight testing. Those two phases are mainly based on real product or scaled models, and each change has a significant impact on design time and costs. The VP, instead, allows having a steeper fidelity-time curve; the product reaches faster a higher fidelity level. The flight verification test, on the other hand, remains the primary tool for the design verification [5].

Several software packages are available on the market for Conceptual and Prelimi-

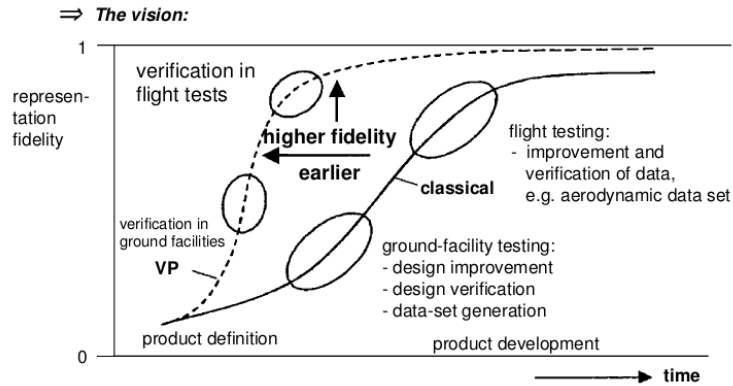


Figure 1.2: Benefits of the virtual product in aircraft design [5].

nary Aircraft design, well-known examples are AAA from Darcorporation and RDS from Dan Raymer. Both software packages use to a large extent empirical relations. The CEASIOM package was the result of the European Funded FP6 project SimSAC and was made available for free. It included modules with different levels of fidelity, for example, the aerodynamic database could be generated using empirical relations, but an Euler solver was also available if a higher fidelity was needed. Its successor is the CEASIOMpy software [6] that uses the CPACS format [7] and is written in Python. CEASIOMpy aims to simplify the transition towards new VP models and MDO algorithms for preliminary aircraft design.

### 1.1.1 Conventional and unconventional aircraft differences

In this thesis, the research study has been carried out for so-called conventional and unconventional aircraft. Since there is no general definition to distinguish those categories, it is necessary to clarify their differences.

A conventional aircraft is characterised by one fuselage and 2 main wings that respect the standard geometrical proportions <sup>1</sup>:

- $8.0 < l_{fg}/WDT_{fg} < 14.0$ ;
- $2.0\text{m} < S_w/b < 9.0\text{m}$ .

Where  $l_{fg}$  and  $WDT_{fg}$  are the fuselage length and width, while  $S_w$  and  $b$  are the main wing area and span. As an example the Airbus A320neo (fig.:1.3a) has the  $l_{fg}/WDT_{fg} \approx 9.5$  and  $S_w/b \approx 3.4\text{m}$ .

A semi-conventional aircraft has proportions of a standard configuration, but the design for the wings does not respect the standard criteria, the box wing aircraft is a classical example (fig.:1.3b).

An unconventional aircraft, instead, even if it adheres to the common aerodynamics design criteria, does not match any fuselage or wing proportion. The blended wing body aircraft is a perfect case of an unconventional aircraft (Roy H. Lange imagined some of the alternative designs for the Lockheed company in the "Review of Unconventional Aircraft Design Concepts" [8]).

<sup>1</sup>The values listed are extrapolated from the aircraft in the database

The BWB has the purpose of minimising fuel consumption through maximising the lifting surfaces and minimising the total surface. It could have in theory the same payload of a standard jetliner but with a 30% less wet surface which allows reducing skin friction drag remarkably [9]. It integrates the wing and the fuselage into a single lifting surface that, in the central part, is thick enough to accommodate the passenger cabin and the cargo bay.

Inside the unconventional category, it is possible to find also supersonic civil aircraft, like the Concorde (fig.:1.3c), or aeroplanes with multiple fuselages like the Virgin Galactic *White Knight Two* (fig.:1.3d). The Concorde is no longer flown today due to its high maintenance, fuel costs and noise but the supersonic flight will come back with global demand of daily business trips around the globe. The White Knight Two, instead, represents a new frontier for low-orbit commercial spacecraft launchers. The purpose of this aircraft is to launch the Virgin Galactic *SpaceShipTwo* from 16 kilometres to the boundary of space (100km).

## 1.2 CEASIOMpy software

CEASIOM (Computerised Environment for Aircraft Synthesis and Integrated Optimisation Methods) [3] is a support tool for engineers in the conceptual and preliminary design process. It was developed within the SimSAC project (Simulating Stability And Control Characteristics for Use in Conceptual Design), funded by the European Commission 6th Framework Program on Research, Technological Development and Demonstration. The project lasted for 3 years from 2006 to 2009 focusing on the development of CEASIOM in a MATLAB environment [11].

In 2015 began the AGILE (Aircraft 3rd Generation MDO for Innovative Collaboration of Heterogeneous Teams of Experts) innovation project, granted by the European Commission and coordinated by the DLR (German Aerospace Centre) that targets the multidisciplinary optimisation (MDO) using distributed analysis frameworks to develop the new generation of aircraft [12]. Within the AGILE project, CFS Engineering (Computational Fluid and Structure Engineering) started to develop the CEASIOMpy version entirely based on Python environment. The purpose is to realise an open-source conceptual aircraft design tool focused on multi-fidelity aerodynamics and aircraft stability [3]. Differently, from the old CEASIOM software, CEASIOMpy permits to explore new aeroplanes configurations.

The present state of the CEASIOMpy software [7] includes (fig.:1.4):

- a geometry analysis defined in the CPACS XML format [13];
- an aerodynamics analysis;
- weight balance (WB) and range modules presented in this thesis.

Currently, CFS Engineering and Airinnova are also developing:

- a propulsion module;
- a mission analysis that gathers the data from the aerodynamics and propulsion modules;



(a) Conventional aircraft (Airbus A320neo®, [www.airbus.com/aircraft/passenger-aircraft/a320-family.html](http://www.airbus.com/aircraft/passenger-aircraft/a320-family.html)).



(b) Semi-conventional aircraft (Box wing aircraft [10]).



(c) Concorde®.



(d) White Knight Two.



(e) Blended wing Body concept ([www.nasa.gov/centers/dryden/research/X-48B](http://www.nasa.gov/centers/dryden/research/X-48B)).

Figure 1.3: Pictures of different aircraft configurations: conventional (1.3a), semi-conventional (1.3b) and unconventional (1.3e, 1.3c, 1.3d).

- a structural analysis based on the WB data;
- an aeroelasticity analysis enabled by the data from both the aerodynamics and structural analysis;
- a stability analysis with all the data from the previous models.

It is important to remark that in figure 1.4 the three main modules are shown separately but in reality, all the analysis modules are interconnected and each can influence the others. The entire process is also meant to be integrated inside an optimisation cycle to achieve, automatically, the best possible design.

To conclude it is important to state that CPACS (Common Parametric Aircraft Configuration Scheme) is a data definition model for air transport system that allows ordering and centralise all the interdisciplinary information required for an aircraft design, minimising the number of interfaces between disciplines and providing an unequivocal exchange of data even with external codes [13]. For this reason, the aircraft tested with the CEASIOMpy software must be defined using the CPACS geometry format.

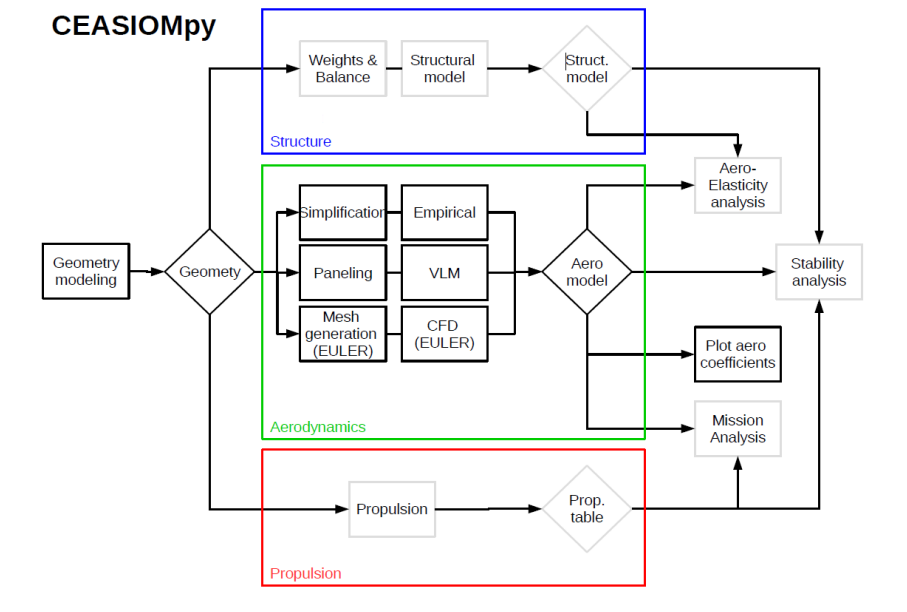


Figure 1.4: CEASIOMpy current development schema.

### 1.3 Purpose of the project

The primary purpose of this master thesis studies is the development of the Weight and Balance analysis module that includes a simple range estimation module for the CEASIOMpy software in Python environment.

The module consists of 2 parts, one for conventional aircraft design and the other both for conventional and unconventional aircraft analysis. Each part has two components that can work independently: the weight and the balance analysis. The range analysis, instead,

works for all the aircraft configurations. Both parts read the aircraft geometry defined in the CPACS format inside an XML script.

All the possible evaluations carried out by the modules developed are summarised in the following points:

- The conventional aircraft analysis module can determine:
  - the aircraft weights based on a database of existing aircraft;
  - the number of passengers, rows and abreast alongside a possible seat disposition;
  - the number of crew members and lavatory related to the passenger number;
  - the range using the Breguet equation;
  - the fuel consumption during the entire flight and for each flight phases;
  - the centre of gravity (COG) and the moment of inertia (MOI);
  - the weight and range with the wingspan, the wing area, the fuselage length and the fuselage width given directly as keyboard inputs without the entire geometry of the aeroplane.
- The general aircraft module can compute:
  - the aircraft weights;
  - the structure and systems weights;
  - the engine weight and thrust, or power in case of turboprop engine;
  - the number of passengers and crew members;
  - the range, fuel consumption, COG and MOI similarly to the conventional module;
  - all the above estimations can be carried out for conventional a novel aircraft configurations.

The conventional aircraft analysis software is meant to be a black-box code for the conceptual design of a standard aeroplane configuration. The code provides a wide range of default values valid for the majority of the aircraft and does not require their precise definition to complete the WB analysis. In the early conceptual phase, the user can define only the main dimensions of the aircraft (wingspan, the wing area, the fuselage length and the fuselage width) to have a weight and range evaluation.

The general aircraft analysis code requires a more significant number of input parameters and, for this reason, should be able to provide a more accurate analysis. The evaluation, on the other hand, is slower than the conventional aircraft analysis software and can be initiated only with the CPACS geometry file as input.

# Chapter 2

## The physics behind

In this chapter, all the main physical and practical aspects that are taken into account for the development of the weight, range and balance modules will be introduced.

### 2.1 Aircraft weights

All the different weights and masses that characterise an aircraft are listed in table 2.1. First, the Empty Mass (EM) represents the sum of all the structural elements, the systems and the fixed equipment and it indicates the weight of the aircraft as soon as it exits the factory. The operating status of an aircraft is then associated with the Operating Empty Mass and consist in the sum of the EM and the operator's items:

- engine oil and coolant, water, unusable fuel;
- furnishing, catering and emergency equipment;
- passenger seats;
- aircraft crew and their luggage.

The OEM is the mass of the aeroplane that stands in the docking position without fuel or payload. Differently, from the EM that is given by the manufacturer and is constant for each aircraft model, the OEM depends also on the customer interior choice, on the purpose of the aircraft and the regulation of the operating country (even though nowadays most country refers to the ICAO legislation).

The ZFM, Zero Fuel Mass is the sum of all the masses of the aircraft structures, systems, operating items, crew members and payload with the usable fuel neglected. During the flight, the total aircraft weight decreases progressively, but the ZFM remain constant. For stability and control purpose, the knowledge of this value permits to define the extremes of variation of the COG. Pairing the value with the one evaluated at MTOM permits to develop a control system that trim<sup>1</sup> the aircraft during the flight by changing the fuel position and thus the centre of gravity.

The ZPM (Zero Payload Mass) is the mass of the aircraft with maximum fuel and no payload boarded; it has a crucial role in the estimation of the maximum possible range of the aircraft.

---

<sup>1</sup>In a trimmed aircraft no force is required to be applied to the flight controls to maintain a levelled flight.

Table 2.1: Aircraft masses with their respective group of belongings.

Wings, fuselage, tail surfaces, landing gear, fairings, ...	Structure		EmptyMass EM	Operating Empty Mass OEM	Maximum Take-Off Mass MTOM	Ramp Mass
Engines, engine mounts and fairings, engine control.	Propulsion	Systems				
Tanks, electrical system, avionics, hydraulics, control system, ventilation, pressurisation, de-icing	Other systems					
Seats, galleys, lavatories, loading equipment	Fixed equipment					
Non-useable fuel						
In-flight meals, life vests, water	Moveable equipment					
Pilots, cabin personell	Crew					
Passengers, freight	Payload					
Oil, lubricants						
Useable fuel						
Fuel used for rolling before take-off						

The maximum payload is the maximum number of passengers that can leave the aircraft in 90 seconds [14] plus the maximum possible payload in the cargo bay. In the final aircraft configuration, considering the presence of different seat classes (economy, business and first) the actual payload is usually lower than the maximum allowed values. It is important to mention that, usually, the fuel tanks can contain more fuel than the maximum allowed with maximum payload. The possibility to modify the fuel amount with the payload gives the manufacturer and customer flexibility during the aeroplane design.

If on top of the OEM we add both the payload and fuel masses it is possible to evaluate the MTOM maximum take-off mass (also known as the maximum brake release weight<sup>2</sup>) that is the maximum weight authorised at brake release to initiate the takeoff roll.

Since the MTOM influences the takeoff distance and speed, it can change depending on the operating region and airport characteristics such as atmospheric pressure and runway length. In this thesis, only the manufacturer MTOM has been taken into account since the software is meant for conceptual and preliminary design.

The last two weights to take into account are the Maximum Landing Mass (MLM) and the Maximum Taxi Mass (MTM). The MLM is the maximum allowed weight of the aircraft in case of an emergency landing. When the aircraft is required to land with a weight higher than the MLM the pilots have to dump fuel before approaching the airport; otherwise, the aircraft may not be able to stop before the end of the runway. Since the MLM is not only defined by the manufacturer but can vary depending on each airport and weather condition, it is not considered for the preliminary design software developed in this thesis. The MTW (or Maximum Ramp Weight) is the maximum taxi weight allowed

<sup>2</sup>The aircraft weight legislation usually refers to the Weight expressing it using the pounds unit (lb); the code developed on this thesis refers to the same value defining it using the mass terms (kg), while the quantities defined as weight are expressed in Newton [N].



for manoeuvring the aircraft during the ground phases; it is lower than the maximum structural weight that the aircraft can withstand but can exceed the MTOM by the small amount of fuel burnt during the taxiing phase.

## 2.2 Centre of Gravity and Moments of Inertia

The centre of gravity and the moments of inertia are the main results evaluated with the balance analysis of the WB software. In this section, a brief explanation of the physical and mathematical background is given.

### 2.2.1 Centre of Gravity

The centre of gravity (COG) is the geometrical point that behaves as if all the masses were concentrated there and all the external forces acted at that point [15]. In aeronautics, it assumes a fundamental role since the aircraft balance and stability depends directly on it.

The COG position must be known during each phase of the flight in order to correctly modify the control inputs and to stabilise the aircraft.

The COG position ( $\mathbf{CG}$ ) is evaluated mathematically as the volume integral of the infinitesimal  $dm = \rho dV$  mass at position  $\mathbf{r}$  divided by the total mass ( $M$ ) of the object:

$$\mathbf{CG} = \frac{1}{M} \int_V \mathbf{r} dm. \quad (2.1)$$

Numerically it is possible to approximate the integral subdividing the body in a certain number ( $N$ ) of point masses that will be characterised by a fraction of the total mass in relation with their position on the object. The previous equation can be rewritten as:

$$\begin{cases} CG_x = \frac{1}{M_{\text{tot}}} \sum_{i=1}^N m_i x_i; \\ CG_y = \frac{1}{M_{\text{tot}}} \sum_{i=1}^N m_i y_i; \\ CG_z = \frac{1}{M_{\text{tot}}} \sum_{i=1}^N m_i z_i. \end{cases} \quad (2.2)$$

Where  $M_{\text{tot}}$  is the total mass of the object,  $N$  is the number of nodes and  $m_i$ ,  $x_i$ ,  $y_i$ ,  $z_i$  are respectively the mass and the coordinates of the  $i^{\text{th}}$  point mass.

### 2.2.2 Moments of Inertia

The moment of inertia is the measure of the resistance of a body to angular acceleration. The rotation of a mass  $m$  around a fixed axis is given by the sum of the resistance relative to its centroidal axis  $I_o$  and to the remote axes [16] (fig.:2.1). In general, it is possible to write:

$$I_{x1} = mr^2 + I_{o1} = m(x_2^2 + x_3^2) + I_{o1}. \quad (2.3)$$

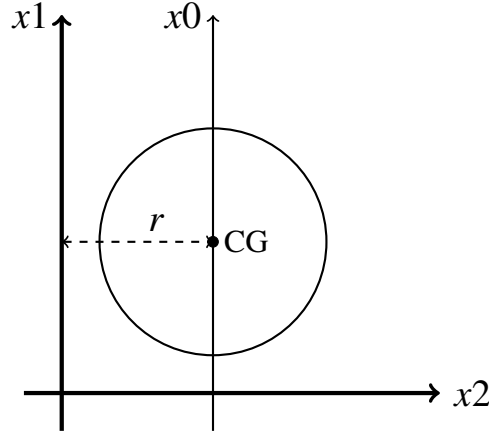


Figure 2.1: Drawing of the centroidal axis  $I_o$  through the centre of gravity (CG) and relative remote axes  $(x_1, x_2)$  for a simple geometry.

Since the definition of the  $I_o$  is:

$$I_{o_{ij}} = \int x_i x_j dm, \quad i, j = x, y, z \quad (2.4)$$

it is possible to define a nine elements matrix (eq.:2.5) with a maximum of six independent values since the matrix is symmetric.

$$\mathbf{I} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \quad (2.5)$$

If the geometry is symmetric about, for example, the x-z plane, it is possible to assume that the  $I_{xy}$  and  $I_{yz}$  are zero. For this thesis all the six independent MOI are evaluated both for conventional and unconventional aircraft and  $I_{xy}$  and  $I_{yz}$  serves as validation parameter for the code.

The stability analysis for preliminary aircraft design requires the moments of inertia referred to the centre of gravity position. For this reason, equation 2.3 becomes:

$$I_{x1} = m \left[ (x_2 - X_{2cg})^2 + (x_3 - X_{3cg})^2 \right]. \quad (2.6)$$

For conventional aircraft, the MOI evaluation can be carried out using semi-empirical equations that need to be calibrated differently for each aircraft to obtain reliable results [16]. In this thesis, instead, the more general lumped masses approach has been chosen. The lumped masses method follows the idea of approximating the integral with a discrete sum of uniformly distributed elements. The aircraft is subdivided in a certain number of equally spaced nodes (fig.:2.2) and each element receives a precise mass relative to its position assuming that the volume, approximated with a certain point, has a uniform mass distribution.

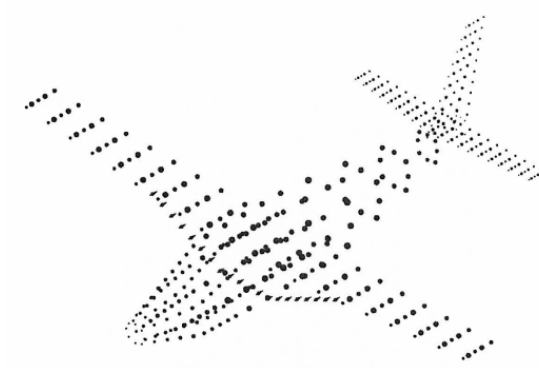


Figure 2.2: Example of an aircraft geometry discretized with lumped masses.

The equations to evaluate the moments of inertia using lumped masses, states [17]:

$$\left\{ \begin{array}{l} I_{xx} = \sum_{i=1}^N m_i [(y_i - CG_y)^2 + (z_i - CG_z)^2], \\ I_{yy} = \sum_{i=1}^N m_i [(x_i - CG_x)^2 + (z_i - CG_z)^2], \\ I_{zz} = \sum_{i=1}^N m_i [(y_i - CG_y)^2 + (x_i - CG_x)^2], \\ I_{xy} = \sum_{i=1}^N m_i [(x_i - CG_x)(y_i - CG_y)], \\ I_{yz} = \sum_{i=1}^N m_i [(y_i - CG_y)(z_i - CG_z)], \\ I_{xz} = \sum_{i=1}^N m_i [(x_i - CG_x)(z_i - CG_z)]. \end{array} \right. \quad (2.7)$$

Where  $N$  is the number of masses available,  $CG_x$ ,  $CG_y$  and  $CG_z$  are the coordinates of the global centre of gravity and  $x_i$ ,  $y_i$  and  $z_i$  are the coordinates of the  $i^{\text{th}}$  node.

Since its an approximation of the integral equation, the higher the number of nodes the more accurate the solution results. At the same time, a high number of nodes requires a high computational time and so a compromise between the error and the calculation time must be found to achieve the best possible result with the minimum computational effort. In section 3.6 the process used to calibrate the lumped method is explained in details.

## 2.3 Range and flight phases

The WB software is not performing a complete mission analysis. It uses the Breguet equation, instead, to rapidly estimate the range versus payload simplifying the optimisation process for the conceptual and early preliminary design process. Since it also evaluates the possible fuel consumption during the flight, this section proposes a brief explanation of the Breguet equation and the different flight phases.

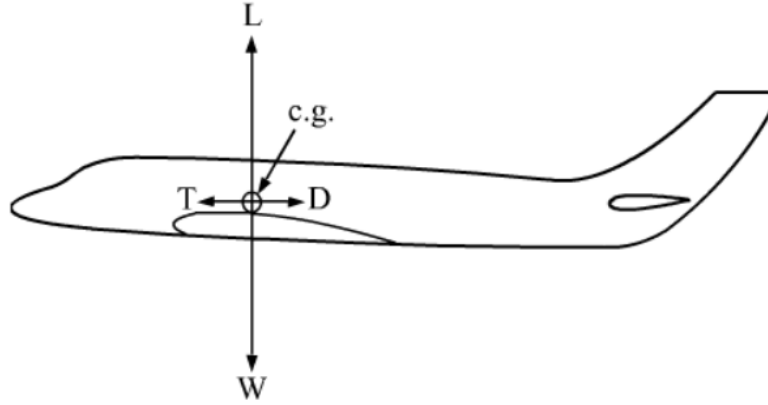


Figure 2.3: Sketch of an aircraft forces during a levelled flight.

### 2.3.1 Breguet range

With the assumption of a levelled flight at a constant speed and altitude, the cruise range ( $R$ ) can be derived considering the infinitesimal distance that the aircraft travels burning an infinitesimal percentage of fuel [18]:

$$\frac{dR}{dW} = \frac{V}{-CT} = \frac{V(LD)}{-CW}. \quad (2.8)$$

where  $V$  is the cruise speed [km/h],  $LD$  is the ratio between the lift and drag forces and  $C$  the Thrust Specific Fuel Consumption (TSFC) or the Specific Fuel Consumption (SFC) expressed with the unit of 1/hr.

Integrating over the variation of the aircraft weight due to fuel consumption it is possible to derive the Breguet range equation:

$$R = \int_{W_i}^{W_{i+1}} \frac{V(LD)}{-CW} dW = \frac{V}{C} \frac{L}{D} \ln\left(\frac{W_{i+1}}{W_i}\right), \quad (2.9)$$

where  $W_i$  and  $W_{i+1}$  are the aircraft weights before and after the flight phase. The ratio  $LD$ , during the cruise, can be evaluated with a CFD (Computational Fluid Dynamics) simulation or as the ratio between the total aircraft weight and the engine thrust. During a levelled flight it is possible to assume the lift ( $L$ ) equal to the aircraft weight ( $W$ ) and the thrust ( $T$ ) equal to the drag force ( $D$ ) (fig.:2.3). Starting from the  $LD$  coefficient at cruise condition it is also possible to determine the  $LD$  at loiter. For a turbojet aircraft, the most efficient loiter occurs at approximately the velocity relative to the maximum  $LD$  while the cruise is usually characterised by  $LD = 0.866LD_{max}$ . For a turbo or piston propeller aircraft, instead, at cruise  $LD = LD_{max}$  and during loiter  $LD = 0.866LD_{max}$  [18].

For jet engines the SFC is the ratio between the fuel consumption per hour divided by the unit of thrust produced and is usually measured as lb/(lbf · hr) or 1/hr. In case of a propeller engine, the SFC is usually defined as the fuel consumption per hour required to produce one horsepower at the propeller shaft (or one brake horsepower  $C_{bhp}$ ). Since  $bhp = 550$  (ft · lbf)/s it is possible to derive the equivalent SFC for a propeller aircraft :

$$C = C_{bhp} \frac{V}{550\eta_p}. \quad (2.10)$$

Table 2.2: SFC and propeller SFC [1/hr] in cruise and loiter in comparison with the type of engines [18].

Typical Jet SFC's	Cruise	Loiter
Pure turbojet	0.9	0.8
Low-bypass turbofan	0.8	0.7
High-bypass turbofan	0.5	0.4

Typical propeller SFC's	Cruise	Loiter
	$C_{bhp}/\eta_p$	$C_{bhp}/\eta_p$
Piston-prop (fixed pitch)	0.4/0.8	0.5/0.7
Piston-prop (variable pitch)	0.4/0.8	0.5/0.8
Turboprop	0.5/0.8	0.6/0.8

where  $V$  is the aircraft speed in feet per second and  $\eta_p$  is the propeller efficiency. To simplify the analysis Raymer proposes an average value for the SFC, propeller SFC and  $\eta_p$  shown in table 2.2.

Similarly to the range equation, it is possible to determine the aircraft endurance as:

$$E = \frac{LD}{C} \ln \frac{W_{i+1}}{W_i} \quad (2.11)$$

### 2.3.2 Flight phases

For a commercial aircraft, the entire flight consists into 6 main phases: take-off, climb, cruise, loiter, descent and landing (fig.: 2.4). The American National transport Safety Board (NTSB) provides the following definitions [19].

The take-off phase starts as soon as the position brakes are disengaged until the aircraft reaches 35 feet (10.668 meters) above the runway; once the aircraft reaches 35 feet, the climb phase starts. Between the take-off and climb there is the acceleration or transition phase, the aircraft follows a circular trajectory reaching the climb angle of attack and velocity (1.2 times the stall velocity). Since this last phase can be defined alongside both the take-off and climb stages, it is important to mention that in this thesis the climb includes the acceleration.

The cruise phases legally starts when the aircraft reaches the prescribed cruise altitude, so the climb phase ends since the rate of climb becomes zero. During the cruise, the aircraft maintains a levelled flight at constant speed.

Once the aeroplane begins to have a negative rate of climb for an extended period the descend phase initiates. This last phase also includes the approach that starts when the aircraft descends below 1000 feet and ends when the pilots execute the landing flares. The Landing phase is the last one and ends as soon as the aircraft clears the runway.

As can be seen from table 2.4, there is also the loiter phase to be considered. In the figure, the loiter is placed after the cruise but can also be identified just before the landing. In any case, it is a safety measure phase and consists of possible extra flight time at reduced speed waiting for the landing clearance or for diverting to a different airport. Following the ICAO regulation, a 30 min or 45 min loiter must be taken into account respectively if the aircraft has a turbo or a piston engines [20]. If the aircraft misses the

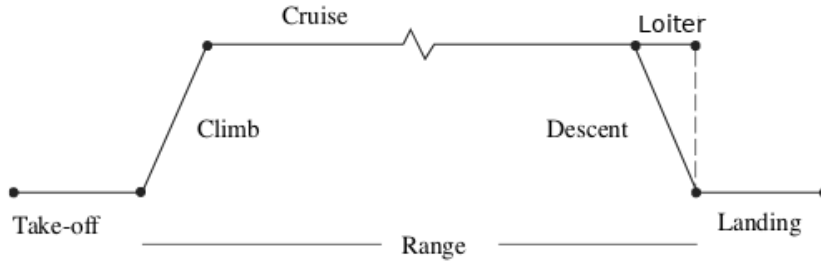


Figure 2.4: Sketch of the common phases of a commercial flight [21].

Table 2.3: Flight phases weight fractions.

	$(W_{i+1}/W_i)$
Take-off	0.985
Climb	0.975
Landing	0.995

approach and cannot safely land, it performs a go-around phase that will consist of a combination of a short climb, a loiter, a new approach and landing.

The weight fraction is the ratio between the aircraft weight before and after each phase; Raymer proposes to define it constant for the take off, climb and landing phases (table 2.3) during conceptual and preliminary design. It is important to remark that the first phase includes: engine start up, taxiing, take-off, transition and climb-out.

The fuel fraction relative to the loiter phase can be defined inverting the endurance equation 2.12, where the flight time ( $E$ ) is the loiter time ( $t_{loiter}$ ), the LD coefficient and the SFC are defined at loiter condition and  $W_{i+1}$  and  $W_i$  are respectively the weights after and before the loiter.

$$W_{i+1} = W_i e^{\left( t_{loiter} \frac{SFC_{loiter}}{LD_{loiter}} \right)} \quad (2.12)$$

In the end, the cruise weight fraction results as the ratio between the aircraft weight before loiter and after the cruise. In general, Raymer proposes that the total weight fraction can be defined, considering 6% of unusable or trapped fuel, as:

$$\frac{W_f}{MTOW} = 1.06 \left( 1 - \frac{W_{al}}{MTOW} \right) \quad (2.13)$$

where  $W_f$  is the fuel weight boarded and  $W_{al}$  is the aircraft weight after landing.

# Chapter 3

## Weight and Balance software

In this chapter, each component of the newly developed *weight and balance* modules (WB), the required inputs, the outputs and the plots produced are explained. Two software modules have been developed, the conventional WB (CWB) and the unconventional WB (UWB). The first consists in a black box function that can be used for the conceptual design of conventional aircraft; the UWB, instead, can analyse both conventional and unconventional aeroplanes for preliminary design and can be used inside an optimisation process. It was built using the knowledge obtained with the CWB software.

### 3.1 Inputs and work-flows

Both the conventional and unconventional *weight and balance* software require a large number of inputs that can be provided by the user; all the input data have a default value that can be used when the actual value is not known or provided. It is important to mention that all the modules work independently from each other. Appendix B.2 provides all the default and required quantities for each module.

The tables in the Appendix B.2 contain:

- the parameter name inside the code (first column);
- a column with yes if the relative value does not have a default quantity and must be provided;
- the default value, when defined;
- the unity of measure;
- the location inside the CPACS (Common Parametric Aircraft Configuration Schema) file;
- a short explanation of the input parameter (last column).

The coefficients for the conventional aircraft analysis are defined relative to the aeroplanes in the database. For new configurations, instead, the general equations are used only if the specific component can be considered conventional, and all the other coefficients are defined to obtain reasonable results since for aircraft like the BWB no real data exists yet.

### 3.1.1 Conventional aircraft module

#### Work-Flow

Figure 3.1a shows the work-flow of the Weight module for conventional aircraft. It starts from the input files that can be either a CPACS *XML* file containing the geometry of the aircraft or a set of input data chosen by the operator (wing area and span, fuselage length and width). The weight module gathers the entries and analyses the aeroplane fuselage and wings geometry (see ch.:3.2.1). It then evaluates the maximum take-off mass, with a linear regression based on a conventional aircraft database, the operating empty mass, the maximum number of passengers and crew members, and produces a plot of the possible seat disposition inside the aircraft (see ch.:3.3). The code will also check if the payload and fuel masses exceed the user limit and it saves the results.

If the user provides the aircraft geometry using the CPACS format [7], it is possible to use the balance module to evaluate the centre of gravity (COG) and the moments of inertia (MOI) of the aircraft (see ch.:3.6). Figure 3.1b shows the conventional balance module work-flow, the code gathers all the data saved inside the CPACS file, analyses the geometry, estimates the COG and MOI and, in the end, produces the plots of the COG and the lumped masses and it saves the results. The code evaluates both the COG and MOI for the following aircraft configurations: max payload, zero fuel, zero payloads, operating empty and a possible configuration directly defined by the user. In both cases, the input CPACS file must be placed inside the ToolInput folder and named *toolinput.xml*. Once the analysis is completed, the code saves the outputs inside the *ToolOutput* sub-folder; from the weight analysis, it is possible to find a folder with the aircraft name (NAME) that contains:

- NAME\_Aircraft\_Geometry.out, with all the geometrical information extracted from the CPACS file;
- NAME\_Seats\_disposition.out, containing a simple plot of the possible seat disposition;
- NAME\_Weight\_module.out, the output text file with all the results of the weight analysis;
- NAME\_mtomPrediction.png, the plot of the linear regression used for the maximum take-off mass estimation.

From the balance analysis inside the ToolOutput folder, the code saves a sub-folder with the aircraft name, that contains:

- NAME\_Aircraft\_Geometry.out, with all the information extracted from the CPACS file;
- NAME\_Balance\_module.out, the output text file with all the results of the balance analysis;
- NAME\_Aircraft\_COG.png, the plot of lumped nodes used for the COG analysis alongside with the COG itself;



- NAME\_Aircraft\_nodes.png, the plot of lumped nodes used for the MOI analysis.

In all modules, inside the ToolOutput folder, the code saves the updated CPACS file named *tooloutput.xml*. If the aircraft is defined without the CPACS file, the code generates a temporary *xml* file named *user\_tooloutput.xml* to be able to transfer the data to the range analysis module.

## Weight Inputs

The primary input option consists of providing a *XML* file with the geometry information generated using the CPACS format. In this case, the geometry acquisition script obtains all the structural information of the aircraft, including the nose, tail and cabin length. The conventional weight analysis can also be carried out without using the CPACS geometry format; in this case the user needs to provide:

- the fuselage width [m] (between 1 and 8 [m]);
- the fuselage length [m] (between 10 and 80 [m]);
- the main wing planform area [m<sup>2</sup>] (between 1 and 900 [m<sup>2</sup>]);
- the main wing span[m] (between 10 and 80 [m]).

The data provided must respect the conventional proportions explained in chapter 1.1.1. The constraints presented above are related to the maximum and minimum dimensions of the conventional aircraft in the database. In this case, since there is no information about the real aircraft geometry, the code roughly evaluates the nose and tail length (eq.:3.1) in relation to the fuselage width ( $WDT_{fg}$ ); the coefficients are derived from the aircraft in the database and are focused on the conventional aircraft cabin length fitting.

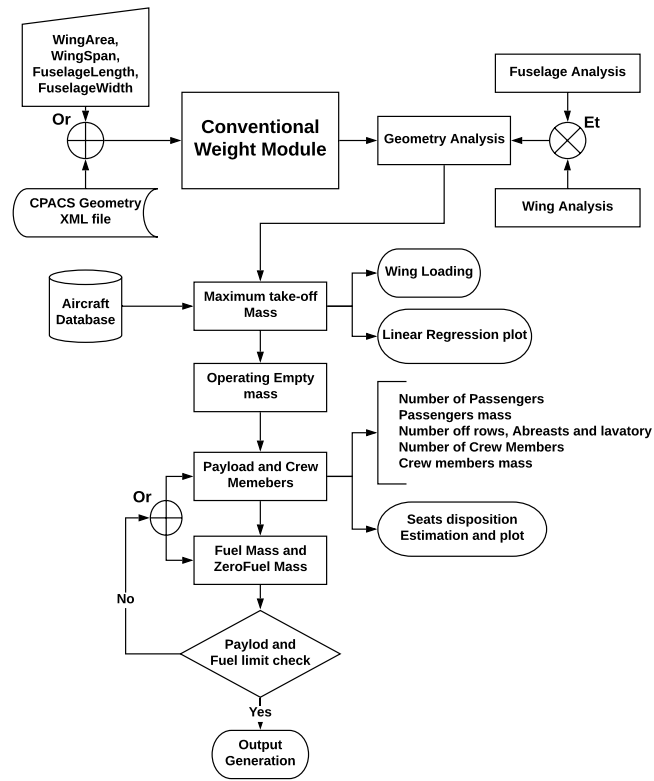
$$\begin{cases} l_{\text{nose}} = 1.3WDT_{fg} \\ l_{\text{tail}} = 1.5WDT_{fg} \end{cases} \quad (3.1)$$

The software then evaluates the cabin length as the difference between the fuselage length and the nose and tail length. If a CPACS file is provided, the code evaluates the cabin dimensions directly from the provided aircraft geometry available in this file (chapter 3.2.1). The user can define multiple parameters to specify the aircraft characteristics further<sup>1</sup> (default value in parenthesis):

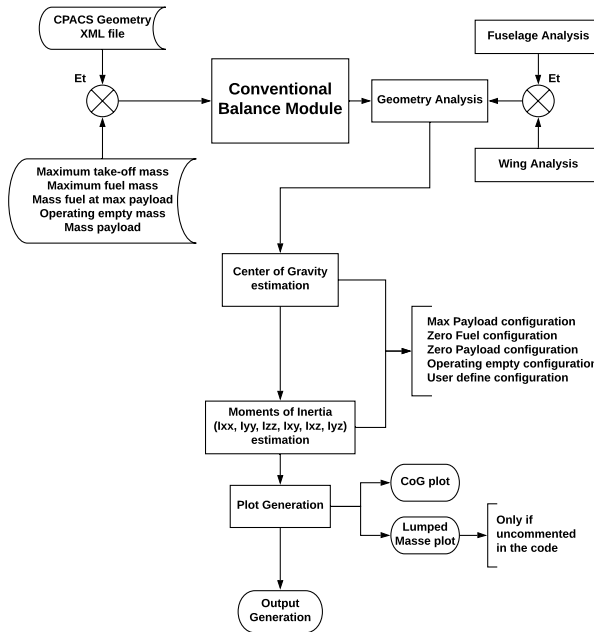
- *isDoubleFloor* option, equal to 0, 1 or 2 respectively in case of no second floor, full second floor (A380<sup>®</sup>) and half second floor (Boeing 747<sup>®</sup>);
- *turboprop* option, True if the engine is a turbo-propeller (TP);
- the number of pilots ( $Nb_{\text{pilot}}$ , 2);
- the mass of a pilot [kg] ( $M_{\text{pilot}}$ , 102 [kg]);

---

<sup>1</sup>All the parameters can be seen in appendix B.2 with the relative default value, a short explanation and the relative placement inside the CPACS file.



(a)



(b)

Figure 3.1: Weight (3.1a) and Balance (3.1b) conventional modules work-flow chart.

- the mass of a crew member [kg] ( $M_{\text{cabinCrew}}$ , 68 [kg]);
- the mass of a passenger [kg] ( $M_{\text{pax}}$ , 105 [kg]);
- the number of passengers per lavatory ( $\rho_{\text{lav}}$ , 50 [pax/lavatory]);
- the fuel density [kg/m<sup>3</sup>] ( $\rho_f$ , 800 [kg/m<sup>3</sup>]);
- the unusable fuel reserve percentage (RES, 6%);
- the extra cargo mass [kg] allowed by the user ( $M_{\text{cargo}}$ ).
- the maximum amount of fuel volume [l] allowed by the user ( $Vol_{\text{fmax}}$ );
- the maximum amount of payload [kg] allowed by the user ( $M_{\text{paymax}}$ ).

Moreover the user can modify:

- the seats width and length ( $WDT_s = 0.525$ ,  $l_{\text{st}} = 0.74$  [m]);
- the aisles width ( $WDT_{\text{aisle}} = 0.42$  [m]);
- the lavatory module length ( $l_{\text{lav}} = 1.2$  [m]);
- the fuselage thickness (as a percentage of the width,  $t_{\text{fper}} = 6.63\%$ ).

The unusable fuel reserve is a percentage of the total fuel mass and corresponds to the fuel trapped in the pumping system that can not be used [22] in flight, the default value is chosen equal to the 6% of the total fuel mass [18]. The fuel density instead is 800kg/m<sup>3</sup> in relation with the kerosene density. The number of toilets per passenger is set considering the typical disposition of lavatory in an Airbus<sup>®</sup> aircraft since there is no specific government regulation [23], and the toilet module, usually, is 1.2 meters long in the x-direction. To be able to take into account the common space areas inside the aircraft (e.g. the first class lounge bar in the Airbus A380<sup>®</sup>), the code increases the toilet module length in relation with the aircraft dimension (value chosen to fit the aircraft in the database):

- +3 m if fuselage length is > 70 m;
- +1.5 m, if fuselage length is > 65 m;
- +0.7 m, if fuselage length is < 65 m.

If the user knows precisely the lavatory module length, he/she can modify the *tltLength* option inside the CPACS file; in this case, the code does not update the lavatory module length with the fuselage length but uses directly the value defined by the user. The fuselage thickness is set as 6.63% of the fuselage width; the value was obtained by averaging the ratio between the fuselage and cabin width of the aircraft saved inside the database. The economy class seat and aisle dimensions are defined starting from the Airbus<sup>®</sup> ordinary seat dimensions [24] and then they are modified to fit different aircraft in the database. The passenger weight is considered to be the same for male and female, for each season and also includes the cabin and checked baggage weight; similarly, a default value is

chosen for the pilots and the cabin crew members. The values are defined using the EASA (European Aviation Safety Agency) *survey on standard weights of passengers and baggage* [25] and from the NASA synthesis [26]. The *turboprop* can be True or False if the aircraft has or not turboprop (or piston-prop) engines; since the majority of commercial aircraft have turbofan engines, the relative default value is False.

## Balance Inputs

The balance module requires the aircraft geometry and some of the weight analysis outputs written in the CPACS format and saved inside the ToolInput folder as *toolinput.xml*. The CPACS file must contain:

- the maximum take-off mass;
- the operating empty mass;
- the payload mass;
- the maximum fuel mass;
- the maximum fuel mass with maximum passengers.

The user can set the *wingMounted* option (WM) True or False, respectively if the engines are attached to the main wing or near the aircraft tail. The user can also define a personal case for the COG and MOI; he/she can choose the fuel and payload amount boarded inside the plane as a percentage of their respective maximum values. In this case, the user must define True the *userCase* option and the *fuelPerc* and *payloadPerc* option between 0% and 100%. It is important to mention that defining both values to 100% makes the aircraft exceed the estimated MTOM; for this reason, the code modifies the chosen percentages to respect the MTOM limit. For the MOI analysis, the code divides the aircraft in lumped masses; the user can modify the spacing between the nodes, but is recommended to use the default value (ch.:3.6):

- 0.05 m for the fuselage nodes along the x, y and z directions;
- 0.05 m for the wings along the y-direction.

The wing profile, along with the chord direction, is divided in a number of fixed nodes that the user can modify and is set 30 as default; the code does not place nodes inside the wings.

## 3.1.2 Unconventional aircraft module

### Work-Flow

The unconventional weight and balance module is the upgraded version of the conventional module, and it was built thanks to the knowledge obtained with the conventional aircraft analysis module. It can study in more detail both conventional and unconventional aircraft and is also meant to be used alongside an optimisation process. Figure 3.2 and 3.3 show the work-flow of the Weight and Balance modules for the general aircraft analysis.

It starts from the input files that must be a *XML* script containing the geometry of the aircraft in the CPACS (Common Parametric Aircraft Configuration Schema) format [7].

The weight module gathers the inputs and, if the aircraft has one or more fuselages, analyses the aeroplane fuselage and wings geometry (see ch.:3.2.2). If the aircraft instead is a Blended Wing Body (BWB), the code runs a different routine to estimate the geometry, the cabin area and the fuel tank volume (ch.:3.2.3). The code estimates the structural mass ( $M_{str}$ ), then it evaluates the passengers mass ( $M_{pax}$ ) and number ( $Nb_{pax}$ ) and fuel mass ( $M_f$ ) and checks if they exceed the user limit and, in this case, modifies their value. The software evaluates the MTOM through an iteration process that starts with a first guess of the MTOM and ZFM (eq.:3.2):

$$\begin{cases} \text{MTOM} = \text{WL} \cdot S_w, \\ \text{ZFM} = \text{MTOM} - M_f, \end{cases} \quad (3.2)$$

where WL is the wing loading [ $\text{kg}/\text{m}^2$ ] and  $S_w$  is the wing planform area [ $\text{m}^2$ ]. Inside the iteration cycle, the code evaluates the engine mass, if not already defined by the user, the crew members and system masses as a function of the MTOM and ZFM. At the convergence of the process, the code also estimates the OEM and saves all the evaluated masses. Differently, from the conventional aircraft analysis software, the general module does not suggest a possible seat disposition.

As it is possible to notice from figure 3.3, the balance module gathers all the data saved inside the CPACS file, analyses the geometry, estimates the COG and MOI and in the end produces the plots of the COG and the lumped masses and saves the results. Both the COG and MOI are evaluated with the following aircraft configurations: max payload, zero fuel, zero payloads, operating empty and user-defined option. If the user knows in advance the engine position the code also considers that information for the COG and MOI analysis.

For both the weight and balance modules the input CPACS file must be placed inside the ToolInput folder and named *toolinput.xml*. Once the analysis is completed, the code saves the outputs inside the *ToolOutput* sub-folder. From the weight analysis, it is possible to find a folder with the aircraft name (NAME) that contains:

- NAME\_Aircraft\_Geometry.out, with all the geometrical information extracted from the CPACS file;
- NAME\_Weight\_unc\_module.out, the output text file with all the results of the weight analysis.

From the balance analysis, instead, inside the ToolOutput folder the code saves a sub-folder with the aircraft name, that contains:

- NAME\_Aircraft\_Geometry.out, with all the information extracted from the CPACS file;
- NAME\_Balance\_unc\_module.out, the output text file with all the results of the balance analysis;
- NAME\_Aircraft\_COG.png, the plot of lumped nodes used for the COG analysis alongside with the COG itself;

- NAME\_Aircraft\_nodes.png, the plot of lumped nodes used for the MOI analysis (only if uncommented inside the code).

In all modules, inside the ToolOutput folder, the code saves the updated CPACS file named *tooloutput.xml*.

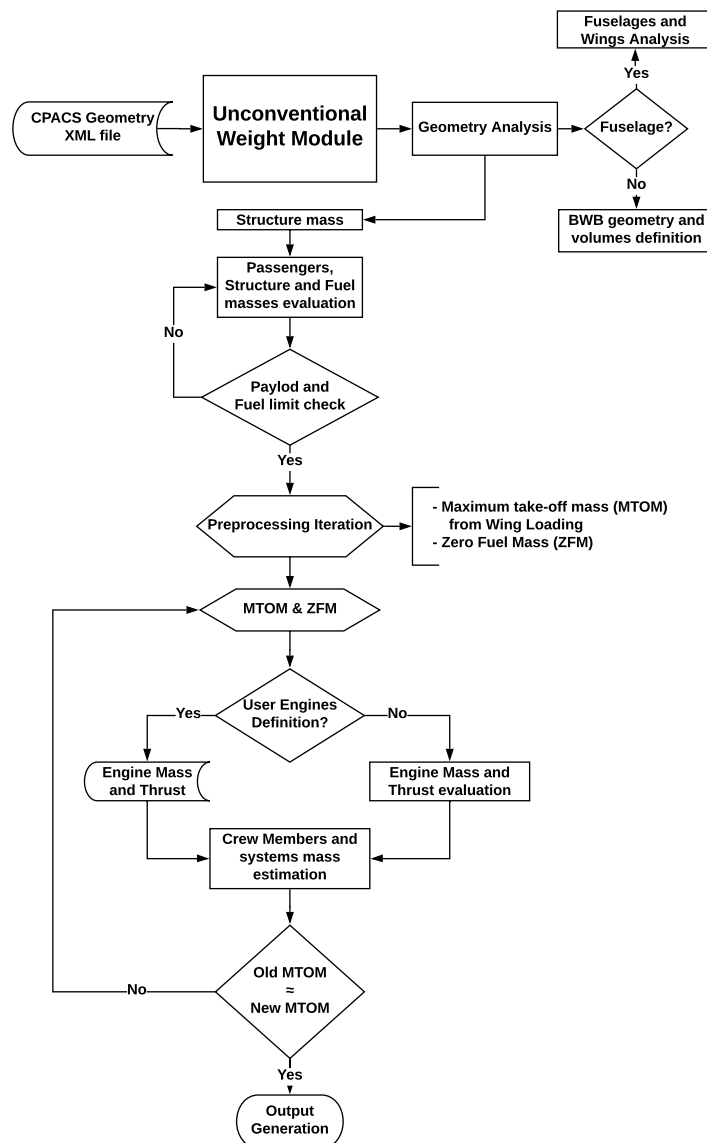


Figure 3.2: Unconventional weight module work-flow chart.

### Weight unconventional Inputs

The general aircraft weight analysis has more input parameters than the conventional module. All the inputs have a default value (tab.:B.4) that the user can modify, the code requires only the *toolinput.xml* file as input.

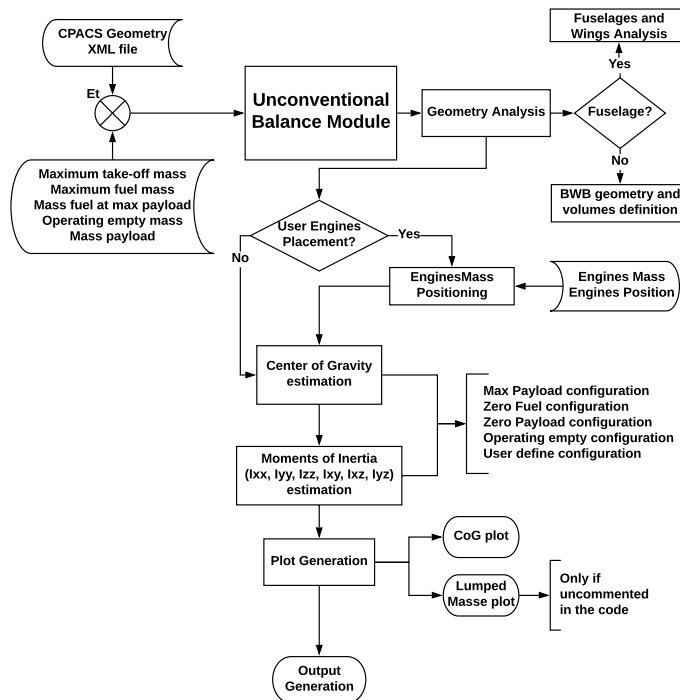


Figure 3.3: Unconventional balance module work-flow chart.

Similarly to the conventional module, the code requires the number of pilots and their mass, the cabin crew members mass, the number of passengers per lavatory and the mass of a single passenger; it also requires the unusable fuel reserve percentage (RES),  $\rho_f$  and the number of floors.

Other input values are (default value inside brackets):

- the fuselage virtual thickness and density (see chapter 3.4.1);
- the *singleHydraulics* option (SH) True or False in relation with the aircraft Hydraulic system 3.4.2);
- the cruise speed (272 [m/s], Mach = 0.8);
- the lift over drag coefficient (17);
- the cruise Specific Fuel Consumption (0.5 [1/hr]);
- the minimum cabin height (2.3 [m]);
- the wing loading starting value (600 [kg/m<sup>2</sup>]);
- the number of engines ( $Nb_e = 2$ ).

If the user knows in advance the mass ( $M_e$ ) and max thrust ( $T_{max}$ ) of each engine, it is possible to set the value inside the CPACS file. If the user then sets the *userEngine* option to True the code does not evaluate the engine characteristics but uses the user-defined values ( $M_e$  and  $T_{max}$  must be defined separately and equal for each engine). The user can

impose a maximum passenger mass, a maximum passenger number, a maximum allowed fuel volume and an extra cargo mass. The code does not check if the added cargo fits inside the aircraft but adds the cargo mass to the total payload mass, this last option should be used only when the user knows in advance the total payload mass.

The total fuel mass ( $M_{f_{max}}$ ) of an aircraft is related to the fuel tanks volume. However, usually, the maximum amount of fuel boarded with maximum payload ( $M_{f_{mp}}$ ) is lower than the maximum amount that can fill the tanks. This approach increases the aeroplane flexibility; for example, it is possible to modify the operative range during its lifetime; it is essential to mention that it actively depends on the aircraft purpose. By studying the conventional aircraft inside the database, it is possible to determine  $M_{f_{mp}}$  merely subtracting from the MTOM the ZFM. Looking then at the *manual for airport planning* of the specific aeroplane, it is possible to notice the difference between the fuel mass evaluated and the total fuel mass that can be stored. Generally it is possible to expect that the  $M_{f_{mp}}$  is roughly the 80% of  $M_{f_{max}}$  for a conventional turbofan aircraft. If the aircraft is designed for long-range missions, instead, one can expect that the fuel tanks contain almost  $M_{f_{max}}$  at max payload and the ratio is roughly 95%. This is different for turboprop aircraft that are light and used for short range flight; for this reason, the percentage usually drops to around 50%. The code uses two parameters  $FPM$  and  $FPM_T$  respectively for turbofan and turboprop aircraft (respectively 80 and 50 as default); those parameters are not defined inside the CPACS file (they are not meant for optimisation) but can be modified directly by the user inside the python script.

The last option that the user can define is *fuelOnCabin*. For the BWB this is a float number and represent the percentage of the free space inside the central portion of the main wing volume that can be used as a fuel tank (zero as default). For an aircraft with one or more fuselages the *fuelOnCabin* option is an array with one element per fuselage set zero as default. If the fuselage does not contain passengers, but it is used as fuel tank itself, the user can set a value between 0% and 80% to indicate the percentage of the volume used as the fuel tank.

### **Balance unconventional Inputs**

The unconventional aircraft balance analysis module has the same input values as the conventional one, plus the engine mass and the number of engines defined by the user or evaluated with the weight analysis. The main difference with the conventional balance code is that the user can also define the engine lumped mass position. If the engine placement is known, the user must set to True the *userEnginePlacement* option and for each engine, he/she must specify the relative lumped mass location (the aircraft should be built alongside the x-direction starting from the nose tip and on the x-y plane). If the code is run separately from the weight analysis, the user can also provide the *fuelOnCabin* option when a fuselage does not contain passengers but only fuel. Moreover, the unconventional module can perform the balance analysis for the BWB or multiple fuselages configuration.

### **3.1.3 Range module**

The range module (fig.:3.4) works for all kind of aircraft. It starts reading the information obtained from the input CPACS file inside the ToolInput folder; this module



requires:

- the maximum take-off mass;
- the operating empty mass;
- the payload mass;
- the maximum fuel mass;
- the maximum fuel mass with maximum passengers.

It also requires (default value inside brackets):

- the pilot number and mass (2 and 102 [kg]);
- the aircraft maximum lift over drag coefficient (17);
- the cruise speed (272 [m/s]);
- the Specific Fuel Consumption (SFC) at cruise and loiter (0.5 and 0.4 [1/hr]);
- the loiter time ( $h_{\text{loiter}} = 30$  [min]);
- the *turboprop* option (False);
- the *winglet* type option (0).

The winglet option can be set 0, 1 or 2 and corresponds to no winglet, medium and high efficiency winglet, it reduces the SFC by 0, 0.5 and 1.0 respectively. The code then evaluates the fuel consumption, the cruise and maximum ranges with maximum and no payload. It also proposes a crew members variation with the estimated flight time. In the end, the code produces the range VS payload plot, write the output *txt* file and saves them inside a folder with the aircraft name. Inside the ToolOutput folder, the code also saves the updated *tooloutput.xml* file.

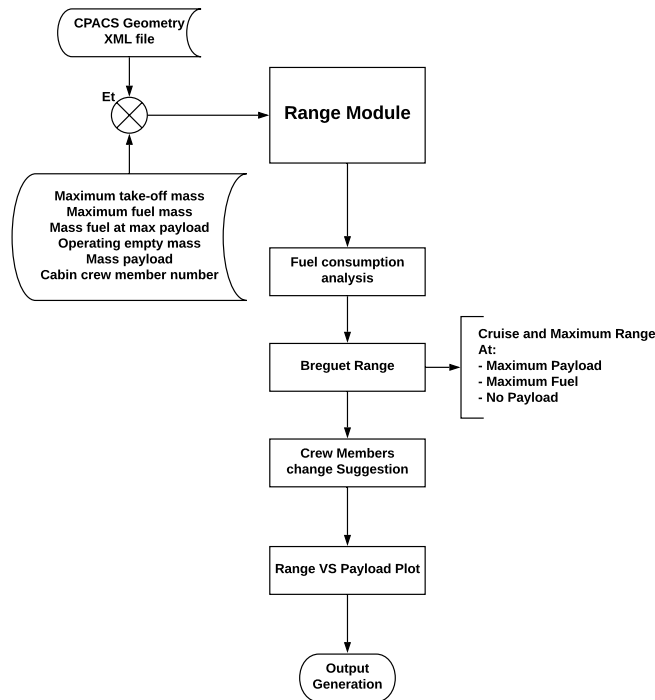


Figure 3.4: Range module work-flow chart.

## 3.2 Geometry acquisition

When the aircraft is defined in a *XML* file using the CPACS schema, the geometry analysis module evaluates the fuselage and wings dimensions. To be able to extract all the information from the *XML* file the code uses the *TiXI* [27] and *TiGL* [28] libraries. The first one allows reading all the *XML* nodes inside the CPACS file to find, extract, replace, remove and add elements. The *TiGL* geometry functions extract essential data from the CPACS file (e.g. the number of wings or fuselages, the number of sections and segments used to build the wings or the fuselage and the symmetry planes defined to construct a particular portion of the geometry [29]). The code uses the *TiGL* functions also to obtain the volume of the aircraft components as well as the wing planform area, chord, mean aerodynamic chord (MAC) and span. For the fuselage analysis, it is possible to evaluate for example the length, the perimeter of each section and many other useful features. The fuselage and the wings are built using a certain number of section profiles that are subsequently positioned and connected in pairs to create the element segments. Since the user can create a single profile defining only its main points, the code uses a *TiGL* function to interpolate between them obtaining a more detailed definition of the sections and segments. The code displays the log information, saves all the fuselage and wings data in the aircraft geometry class, exports them back to the main script and writes the *Aircraft\_Geometry.out txt* file with all the main geometry quantities extracted from the CPACS file (listed in tables 3.1 and 3.2).

Table 3.1: Conventional aircraft main geometry data extracted from the CPACS file with relative unity of measure (UoM).

<b>Conventional Aircraft Geometry</b>			
<b>Fuselage</b>		<b>Wings</b>	
<b>Name</b>	<b>UoM</b>	<b>Name</b>	<b>UoM</b>
Number of sections	[-]	Number of wings	[-]
Number of segments	[-]	Symmetry plane	[-]
Cabin segments	[-]	Number of sections	[-]
Fuselage length	[m]	Number of segments	[-]
Nose length	[m]	Span	[m]
Cabin length	[m]	MAC length	[m]
Tail length	[m]	MAC coordinates	[m]
Aircraft length	[m]	Sections thickness	[m]
Perimeter of each section	[m]	Sections mean thickness	[m]
Relative distance of each section to the first one	[m]	Segments length	[m]
Length of each segment	[m]	Max chord length	[m]
Mean width	[m]	Min chord length	[m]
Width of each section	[m]	Main wing planform area	[m <sup>2</sup> ]
Volume of each segment	[m <sup>3</sup> ]	Volume of each wing	[m <sup>3</sup> ]
Cabin volume	[m <sup>3</sup> ]	Total Volume	[m <sup>3</sup> ]
Total volume	[m <sup>3</sup> ]	Volume for fuel storage	[m <sup>3</sup> ]

Table 3.2: Unconventional Aircraft extra geometry data extracted.

<b>Unconventional Aircraft Geometry Extra Data</b>			
<b>Without Fuselage (BWB)</b>			
<b>Name</b>	<b>UoM</b>	<b>Name</b>	<b>UoM</b>
Main wing wetted area	[m <sup>2</sup> ]	Total volume of the fuselage wing	[m <sup>3</sup> ]
Tail wings wetted area	[m <sup>2</sup> ]	Volume of the wing without people	[m <sup>3</sup> ]
Main wing planform area	[m <sup>2</sup> ]	Fuel volume in the wing	[m <sup>3</sup> ]
Cabin area	[m <sup>2</sup> ]	Fuel volume in the fuselage wing	[m <sup>3</sup> ]
Cabin volume	[m <sup>3</sup> ]	Total fuel Volume	[m <sup>3</sup> ]
<b>With Fuselages</b>			
<b>Fuselages</b>		<b>Wings</b>	
<b>Name</b>	<b>UoM</b>	<b>Name</b>	<b>UoM</b>
Number of fuselages	[-]	Main wing wetted area	[m <sup>2</sup> ]
Cabin area	[m <sup>2</sup> ]	Tail wings wetted area	[m <sup>2</sup> ]
Wetted area	[m <sup>2</sup> ]	Total wings area	[m <sup>2</sup> ]
Fuel Volume in each fuselage	[m <sup>3</sup> ]	Main wing planform area	[m <sup>2</sup> ]

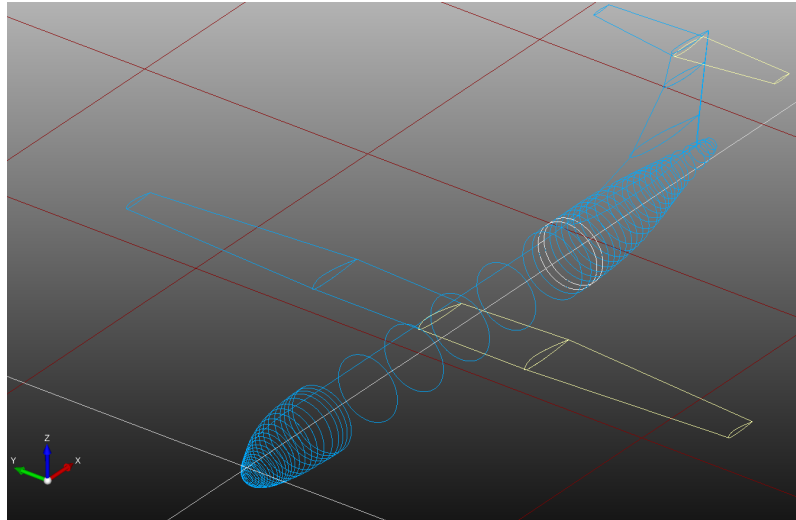


Figure 3.5: Representation of the sections and segments used to build the ATR72 geometry, the yellow wire-frames are relative to the mirrored wings.

### 3.2.1 Conventional geometry

The geometry analysis of a conventional aircraft starts with the fuselage evaluation and, more precisely, with the extraction of the number of fuselages (that must be 1 for conventional aircraft), sections and segments as well as the entire fuselage volume from the *XML* file. When an element is defined using a symmetry plane, it is mirrored to that plane, but the *TiGL* functions will not detect the extra element since there is no information in the CPACS file such as position and volume for the mirrored part. For this reason, the code checks if the structure is created using symmetry and estimate all the information of the mirrored element.

Once the code has defined the number of segments and sections, it checks the start and end section of each segment and reorders them considering that the end section of a segment is the beginning of the next one. The first section is the one that has the lowest *x* position since the aircraft is designed along the *x*-axis and on the *x*-*y* plane. The code still works when a section is defined and not used to build a segment and when the sections and segments are defined in the CPACS file with an order different from the one used to build the aircraft segments<sup>2</sup>. The system does not work if a segment is created and then not positioned because it will still count the extra part. Figure 3.5 shows an example of the sections and segments profiles used to build the ATR72<sup>®</sup> aircraft.

The next step consists in the analysis of the aircraft length, with the proper *tigl* function and includes the entire longitudinal space occupied by the aircraft. The code then computes the relative distance of each section to the first one, and the fuselage length corresponds to the difference between the *x*-coordinate of the last and first sections.

The centre point of each section is evaluated averaging the coordinates of two diametrical profile points. The same routine estimates the section circumference using the proper *GetCircumference* function, and the section width considering two points that have

<sup>2</sup>For example, the segment number 3 can be built between section 16 and 9, instead of using the section with number 3 and 4, this operation does not change the definition order of the sections and thus must be taken into account carefully.

the same z-coordinate of the central one and subtracts the y-coordinates. The sections are defined as a set of points, but the start point of each section depends only on the user choice, and thus it is not possible to know in advance where the  $n^{\text{th}}$  point or segment are placed without reading the CPACS file manually. Once the software defines all the centre points of all sections, it evaluates the position at the centre of each segment required for the CG evaluation.

In the last part of the geometry analysis, the script evaluates the nose, tail and cabin lengths. A segment is part of the cabin only when the section width is sufficiently large (eq.:3.3):

$$WDT_{wsec_i} \geq 1.25 \overline{WDT_{wsec}} \quad (3.3)$$

where  $WDT_{wsec_i}$  is the width of the  $i^{\text{th}}$  section and  $WDT_{wsec}$  is the array containing all the sections widths. The coefficient 1.25 takes into account the fuselage thickness and have cabin, nose and tail lengths that fits the real value of the aircraft in the database. The code adds the segments that do not satisfy the equation 3.3 to the nose or tail groups relative to their position. If a segment length is larger than the 65% of the fuselage length and it has the maximum width, instead, the cabin will be placed entirely and only in that section. This routine enables the user to fix the cabin placement when known in advance.

The second analysis carried out concerns the evaluation of the wings geometry. The software extracts the total number of wings using the proper *TiGL* function; it then increases the number by one each time a wing is created using a symmetry plane since the *TiGL* function does not count the mirrored wing. The software, then, evaluates the wing sections and segments number and reorders them. Once the script reorganises the segments, it evaluates the section chord, MAC (Mean Aerodynamic Chord), span, centre point and thickness. The code places the centre section and subsequently the centre segment points at the quarter chord to simulate the wing segment centre of gravity [30]. The code checks if the wing is horizontal or vertical considering that the planform area on the x-y plane must be higher than the same quantity on the x-z and y-z planes. In the conventional analysis, the code considers only the main wings suitable for the fuel tank since the majority of aircraft do not have fuel tanks on the horizontal stabiliser. The script then evaluates the wing and fuel tank volumes: (eq.:3.4):

$$Vol_f = t * Vol_{mw}; \quad (3.4)$$

where  $Vol_f$  and  $Vol_{mw}$  are respectively the fuel and the main wings volumes, while  $t$  is a coefficient equal to 0.72 or 0.5 respectively if the taper ratio (ratio between the wing tip and root chord) is greater or lower than 0.5 (value defined to fit the aircraft in the database).

For conventional aircraft both the fuselage and wings can be defined using symmetry planes but it is recommended to mirror only the wing and build the fuselage entirely.

### 3.2.2 Unconventional geometry

The unconventional aircraft code is capable of analysing different geometry configurations, as mentioned before, with multiple or without fuselages but is also able to evaluate standard (conventional) aircraft configurations.

If the aircraft has one or more fuselages (like the *WhiteKnightTwo*, fig.:1.3d) the code analyses the wings similarly to the conventional case. The idea behind is that an aircraft

with an original fuselage will still have conventional wings that provide lift and fuel storage. The fuel tank volume evaluation uses equation 3.4 since its dimensions are still a fraction of the total wing volume. Differently, from the conventional analysis, the code places fuel inside all the horizontal wings and not only in the main wing. The coefficient  $t$  of equation 3.4, for the unconventional module, results:

$$\begin{cases} t = 0.75 & \text{if } tp \cdot S_{pl_i} > 80; \\ t = 0.65 & \text{if } tp \cdot S_{pl_i} > 40; \\ t = 0.55 & \text{if } tp \cdot S_{pl_i} > 10; \\ t = 0 & \text{if } tp \cdot S_{pl_i} \leq 10. \end{cases} \quad (3.5)$$

Where  $tp$  is the taper ratio and  $S_{pl_i}$  is the planform area of the  $i^{\text{th}}$  wing (value defined to fit reasonably the conventional and unconventional aircraft tested). The product of taper ratio and wing planform area allows considering the fuel tank accurately inside secondary wings, like the horizontal tail stabilisers, that can have a high taper ratio but small planform area. The code also evaluates the wetted area of all the wings using the proper *TiGL* function and uses it later on for the evaluation of the structural mass.

The code carries out the fuselage geometry evaluation similarly to the conventional aircraft, but it also estimates the entire wetted area, the amount of fuel that can be stored or the cabin area and volume in relation with the user choice. If one or more fuselages are meant to be used as a fuel tank, the user can define for each of them the percentage of the total volume occupied by the tank. The *fuelOnCabin* option must contain a value for each fuselage, it is set automatically to zero as default, and the user can modify it considering that the first value is relative to the first fuselage, the second value to the second fuselage and so on (the number of fuselages is related to the definition order inside the CPACS file). It is essential to notice that, in this case, the fuselage will not accommodate passengers. If the fuselage (*fuelOnCabin*=0) can contain passengers, the code evaluates the cabin checking each segment width. The user can also define the minimum height that a fuselage segment must have to be considered part of the cabin. In the end, the script displays the log information on the terminal and saves the fuselage and wing data in the *Aircraft\_Unc\_Geometry.out txt* file.

### 3.2.3 Unconventional geometry without fuselage

If the aircraft does not have a fuselage, like the BWB configuration, the code performs a different routine to estimate the fuel tank and cabin volumes as well as the cabin area. In this thesis, to simplify the reading, the portion of the wing that can accommodate passengers will be called fuselage wing.

First of all the code analyses the wing sections and segments, it reorders them and saves the wetted area as well as the COG nodes (placed at quarter chord along the x-coordinate and the centre of the segment along the y and z-coordinates). Subsequently, the code subdivides the wing into multiple sections and defines them as fuselage, cabin and wing segments. The fuselage segments are defined considering that the average section thickness must be higher than the user-defined limit. The cabin segments, instead, are defined considering the portion of the wing sections that have a thickness always higher than the prescribed limit, this method allows still having the cabin included inside the

fuselage. Once the code has defined the three segment types, it estimates the fuselage and cabin planform areas and the fuselage frontal area using the Gauss's formula 3.6:

$$A = \frac{1}{2} \left| \sum_{i=1}^n x_i (y_{i+1} - y_{i-1}) \right| \quad (3.6)$$

where  $n$  is the number of points considered and  $x$  and  $y$  the coordinates of the  $i^{\text{th}}$  node. As it is possible to notice the  $z$ -coordinate is not taken into account since the cabin floor should be levelled at a constant  $z$  (an example of the area is shown in figure 3.6).

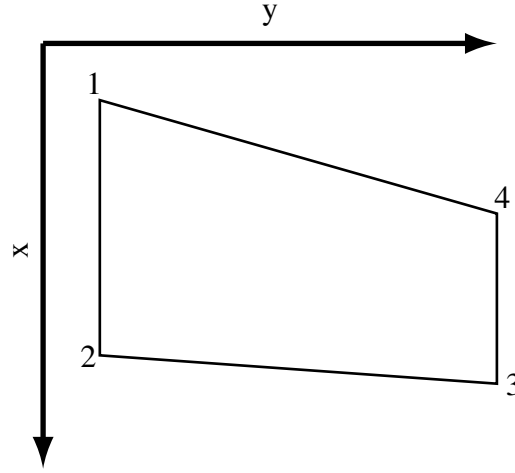


Figure 3.6: Cabin area polygon example.

Thanks to the frontal and planform areas evaluated with the Gauss's formula it is possible to use the Torenbeek's equation [31] to determine the fuselage wing volume ( $Vol_{fg}$ ):

$$Vol_{fg} = \frac{K_Q S_{fg} S_w}{\sqrt{1 + tp_c} b_c}, \quad (3.7)$$

where  $K_Q$  is a coefficient equal to 0.95 for tapered wings,  $tp_c$  is the cabin taper ratio (the ratio between the chord of the wing profile at the beginning and end of the cabin),  $S_{fg}$  is the fuselage wing frontal projection area,  $S_w$  is the wing planform area and  $b_c$  is the cabin span. Since the code defines the cabin area  $S_c$  inside the portion of the wing that has the thickness always higher than the user-defined value, the cabin volume  $Vol_c$  results (eq.:3.8):

$$Vol_c = S_c \cdot h_{lim}, \quad (3.8)$$

where  $h_{lim}$  is the height limit imposed by the user.

At this point, it is possible to evaluate the fuel tank volume inside the wing portion without passengers and inside the free volume that remains inside the fuselage. The wings fuel volume  $Vol_{fw}$  results (eq.:3.9):

$$Vol_{fw} = 0.5 \cdot (Vol_w - Vol_{fg}), \quad (3.9)$$

where  $Vol_w$  and  $Vol_{fg}$  are respectively the entire wing and the fuselage wing volumes, the 0.5 coefficient has been chosen to give reasonable results considering all the possible systems and structural components that will probably be part of the wings and will reduce the

total fuel tank volume. Since the free volume inside the fuselage wing can be substantial, the code allows the user to define a percentage ( $P$ ) of it as a fuel tank and so the fuselage fuel tank volume  $Vol_{ffg}$  results (eq.:3.8):

$$Vol_{ffg} = \frac{P}{100}(Vol_{fg} - Vol_c), \quad (3.10)$$

where  $Vol_{fg}$  is the fuselage wing volume. There is no limitation, at the moment, to the  $P$  value but it is strongly recommended to have a value not higher than 50. This limitation is related to the possible presence of systems and structural components that will limit the available free volume. In the end, the script displays the log information on the terminal and saves the fuselage and wing data in the *Aircraft\_Unc\_Geometry.out txt* file.

### 3.3 Weight analysis for conventional aircraft

The conventional aircraft weight analysis gathers all the data from the geometry analysis and from the user input parameters inside the CPACS file to evaluate:

- the maximum take-off mass (MTOM);
- the operating empty mass (OEM);
- the fuel mass;
- the payload and crew member masses;
- the zero fuel mass (ZFM).

The code evaluates all the required masses, as mentioned before, even if the user defines directly as raw inputs the wing area, the wing span, the fuselage length and the fuselage width.

#### 3.3.1 Maximum take-off mass

Since conventional aircraft have all dimensions related and the aircraft mass depends linearly on them, the code relies on a standard aircraft database for the conventional weight estimation. Inside the database, it is possible to find aeroplanes of different manufacturers that cover a wide range of dimensions, from the small private Cessna<sup>®</sup> jet to the double deck Airbus A380<sup>®</sup>. The manufacturers of the aircraft in the database are:

- Airbus<sup>®</sup>;
- ATR<sup>®</sup>;
- Boeing<sup>®</sup>,
- Bombardier<sup>®</sup>;
- Cessna<sup>®</sup>;
- Embraer<sup>®</sup>;



- Tupolev®.

The aircraft are divided by wingspan ( $b$ ), wing area ( $S_w$ ), fuselage length ( $l_{fg}$ ) and fuselage width ( $WDT_{fg}$ ). For each aircraft the MTOM and the others main characteristics written inside the database (maximum payload, maximum amount of fuel, operating empty mass, maximum range, maximum number of passengers) can be found in the EASA (European Aviation Safety Agency) and FAA (Federal Aviation Administration) reports as well as inside the *Manual for airport planning* of a specific aircraft (e.g. 777-200LR/-300ER /-Freighter Airplane Characteristics for Airport Planning [32]).

The code evaluates the MTOM of a new aircraft using the linear regression method (Appendix:A) based on the aircraft database. The software reads the four main aircraft dimensions ( $b$ ,  $S_w$ ,  $l_{fg}$ ,  $WDT_f$ ) and performs a subdivision of the database in 3 groups. The lower and upper limits of the central area are a function of the wing area and fuselage length. This partition allows focusing the linear regression. For small aircraft, the behaviour of the MTOM as a function of the geometrical parameters was found to be more linear with the wingspan than with the wing area. For this reason, when the fuselage length is lower than 40 meters the linear regression is based on the wingspan while if it is higher, the wing area is used as the primary parameter. In figure 3.7 it is possible to see the three different areas created with the relative expected value. The aircraft tested was, in this case, the Airbus A319®, and the code subdivides the database by wingspan since it has a length of 31.44m. The lower accuracy on the upper regions quantities is accepted since their value is used only to stabilise the solution, but for bigger aircraft, the code provides a more accurate evaluation on that area since it will become the central one. The linear regression has self-adjustable limits that can also subdivide the domain into only two regions if the aeroplane studied has the size close to the ends of the database.

### 3.3.2 Operating empty mass

To evaluate the OEM (operating empty mass) the program starts from the MTOM and evaluates it using the equation proposed by D.P.Raymer [18]:

$$OEM = [a \cdot KC \cdot (MTOM \cdot g)^C] \cdot MTOM; \quad (3.11)$$

where  $g$  is the acceleration of gravity,  $C$  is equal to  $-0.08$  for general aviation twin turbofan engine,  $KC$  is equal to  $1.04$  for a wing with variable sweep and must be set to one otherwise, and  $a$  strongly depends on the aircraft dimensions. All the coefficients for different aircraft proposed by Raymer came from a statistical analysis found in Jane's aircraft catalogue [33], table 3.3 provides some of the central coefficients. Nowadays due to the changes in materials and construction criteria the  $a$  coefficient differs from the one proposed by Raymer but it is possible to adapt with the aircraft size. The  $a$  coefficient, instead, is introduced to be able to fit most of the conventional aircraft OEM and changes in relation with the fuselage length ( $l_{fg}$ ) and with the wingspan ( $b$ ):

- $a = 1.45$  if  $l_{fg} < 30.00\text{m}$ ;
- $a = 1.63$  if  $l_{fg} < 35.00\text{m}$ ;
- $a = 1.57$  if  $l_{fg} < 60\text{m}$  and  $b < 61\text{m}$ ;

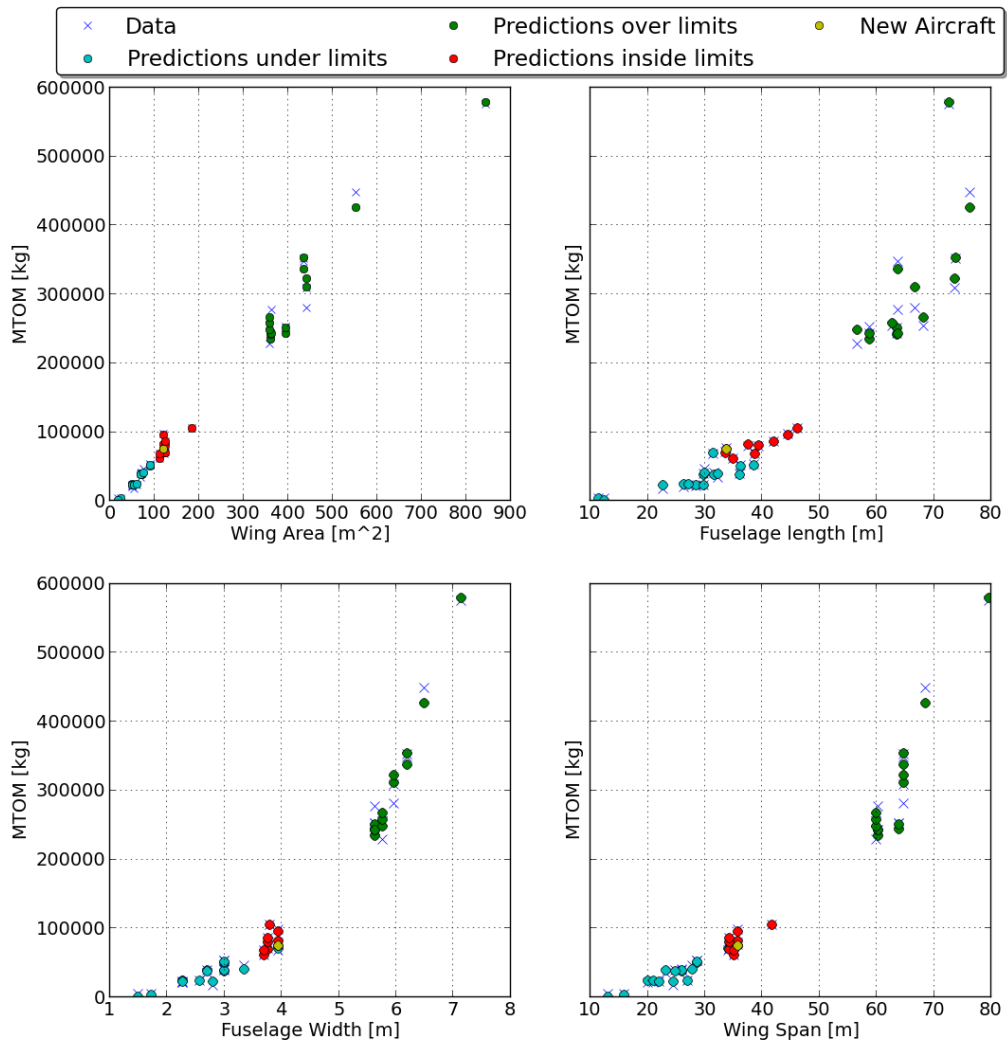


Figure 3.7: Example of the linear regression results plotted for the Airbus A319®.

- $a = 1.63$  in all the other cases.

In case of a twin turboprop aeroplane, the coefficient  $a$  values are:

- $a = 0.96$  if  $l_{fg} < 15.00\text{m}$ ;
- $a = 1.00$  if  $l_{fg} < 30.00\text{m}$ ;
- $a = 1.07$  if  $l_{fg} \geq 30.00\text{m}$ .

The  $C$  coefficient, instead, for a turboprop is fixed at  $-0.05$ .

Table 3.3: List of the most common coefficients used for the evaluation of conventional aircraft OEM listed by Raymer [18].

<b>Aircraft type</b>	<b>a</b>	<b>C</b>
Sailplane - unpowered	0.86	-0.05
Sailplane - powered	0.91	-0.05
Homebuilt - metal/wood	1.19	-0.09
Homebuilt - composite	0.99	-0.09
General aviation - single engine	2.36	-0.18
<b>General aviation - twin engine</b>	1.51	-0.08
Agricultural aircraft	0.74	-0.03
<b>Twin turboprop</b>	0.96	-0.05
Flying boat	1.09	-0.05
Jet trainer	1.59	-0.10
Jet fighter	2.34	-0.13
Military cargo/bomber	0.93	-0.07
Jet transport	1.02	-0.06

### 3.3.3 Passengers, crew members analysis

Once the cabin dimensions (width and length) are defined, the program evaluates the number of abreasts, passengers and lavatory or cabin crew work areas. The number of passengers corresponds to the maximum amount that can fit the aircraft if all the seats have economy size, this value should also correspond to the maximum certificate number of passengers allowed by the FAA and corresponds to passengers that can evacuate the aircraft in 90 seconds [14]. If the aircraft has a second floor, inside the CPACS file the *isDoubleFloor* parameter must be set to 1 in case of a full second floor and 2 in case of a partial second floor (similarly to the Boeing 747<sup>®</sup> one). In this case, the software temporarily increases the fuselage length to evaluate the number of passengers:

$$\begin{cases} l_{fg} = 1.91 \cdot l_{fg}, & \text{if } isDoubleFloor = 1; \\ l_{fg} = 1.20 \cdot l_{fg}, & \text{if } isDoubleFloor = 2. \end{cases} \quad (3.12)$$

the coefficients used fits the Airbus A380<sup>®</sup> and Boeing 747<sup>®</sup> number of passengers since these are the only double-decker aeroplanes on the market.

#### Passengers

The passenger evaluation scripts defines first, the number of aisles ( $Nb_{aisle}$ , eq.:3.14) in relation to the cabin width (eq.:3.13):

$$WDT_c = \left(1 - \frac{t_{fper}}{100}\right) \cdot WDT_f, \quad (3.13)$$

where  $t_{fper}$  is the fuselage thickness as a percentage of the fuselage width and, on average, for conventional aircraft results 6.63%;

$$\begin{cases} Nb_{aisle} = 1, & \text{if } WDT_c < 4.89\text{m}; \\ Nb_{aisle} = 2, & \text{if } WDT_c < 7.6\text{m}; \\ Nb_{aisle} = 3, & \text{if } WDT_c > 7.6\text{m}. \end{cases} \quad (3.14)$$

The first two cases cover all the conventional aircraft while the third one adds an extra aisle in case of a large cabin. Once the code knows the number of aisles, it estimates the number of abreasts ( $Nb_{abreasts}$ ) with equation 3.15 taking into account the fuselage width, thickness ( $t_{fper}$ ), the aisle number and width ( $Nb_{aisle}$ ,  $WDT_{aisle}$ ) and the seat width ( $WDT_s$ ).

$$Nb_{abreasts} = \frac{1}{WDT_s} \left( \frac{WDT_f}{1 + \frac{t_f}{100}} - Nb_{aisle} WDT_{aisle} \right) \quad (3.15)$$

In general, for safety reasons, no more than five seats can be tied up together, and a passenger must have maximum two other people at least on one of his/her left or right sides before the aisle (maximum seat configuration for conventional aircraft, 3 seats + 1 aisle + 5 seats + 1 aisle + 3 seats). The code checks the number of abreast and aisles and warns the user to increase the seat or aisle width to reduce the number of consecutive abreast; the module notifies the user if the seats are not defined symmetrically to the centre line of the aircraft (example: 2 seats on the left side, one central aisle and three seats on

the right side). It is a possible condition, but in a further design stage of the aircraft, the luggage and extra payload will have to be placed carefully to balance the plane correctly. The software, at the moment, does not take into account the distribution of the weight in case of an odd number of abreast. The code also checks if the sum of the seats and aisles width matches the cabin width and suggests the user a possible seat and aisle dimensions change if needs to. The routine to evaluate the number of rows is more complicated because it takes into account also the lavatory and free space areas. First of all the code estimates the number of rows subdividing the cabin length by the length of the seats, then it evaluates the number of passengers and calculates the number of lavatories required (as default the code adds one toilet every 50 passengers). At this stage, the code reduces the available cabin length by the lavatory length and re-evaluates the number of rows and passengers. At the convergence of the process, the code provides, as a result, the number of abreasts, rows, passengers and lavatory, it checks if cabin space remains unused and suggest a seat length change if this is the case.

Default value are provided for the seat length ( $l_{st}$ ), width ( $WDT_s$ ) aisle width ( $WDT_{as}$ ) and lavatory or common area length ( $l_{lav}$ ), but the user can modify them using the relative option in the CPACS file (tab.:3.4). As it is possible to notice the seat and lavatory default lengths depend on the fuselage length, this allows the code to be more flexible when the user does not know the actual value. If the seats and lavatory dimensions are defined inside the CPACS file, the system does not take into account the default values. If an optimisation process is built around the conventional WB software is recommended to define all the seat dimensions directly inside the CPACS file.

After the code evaluates the number of seats, a short script provides a possible seat disposition, figures 3.9 and 3.8 show examples of this output for the Airbus A330<sup>®</sup> (8 abreasts) and the Airbus A319<sup>®</sup> (6 abreasts).

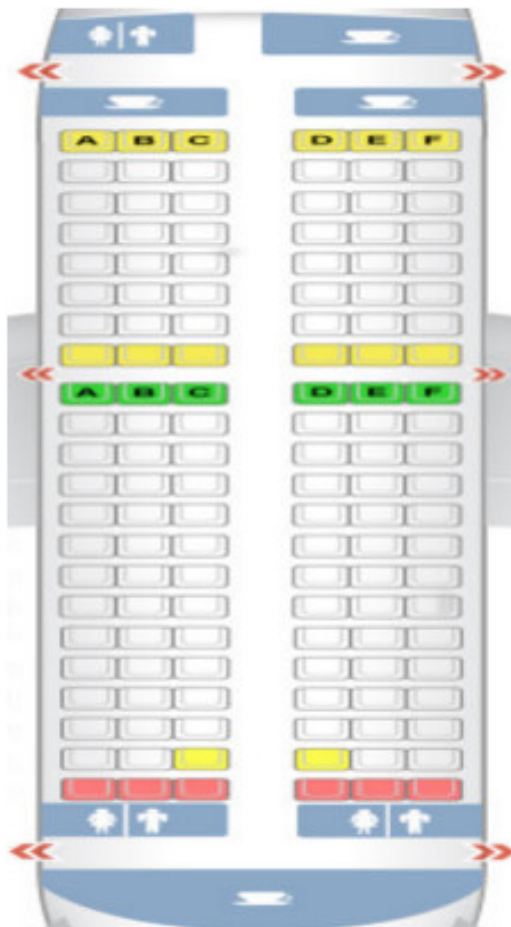
Table 3.4: Default value for the conventional aircraft seats analysis in relation with the aircraft geometry.

	CPACS option	Geometry data	Default
<b>Seat Length</b>	<i>seatLength</i>	$l_{fg} > 15m$	0.74m
		$l_{fg} < 15m$	1.4m
<b>Seta Width</b>	<i>seatWidth</i>	-	0.525m
<b>Aisle Width</b>	<i>aisleWidth</i>	-	0.42m
<b>Lavatory length</b>	<i>lavatoryLength</i>	$l_{fg} > 70m$	4.2m
		$l_{fg} > 60m$	2.7m
		$l_{fg} < 60m$	1.9m

## Crew members

The crew members evaluation follows the passengers mass analysis, and while the number of pilots is a user input (2 as default), the number of cabin crew members is evaluated using the FAA regulation [34] that states:

- For aeroplanes having a maximum payload capacity of more than 7,500 pounds (3400kg) and having a seating capacity of more than 9 but less than 51 passengers, one flight attendant;



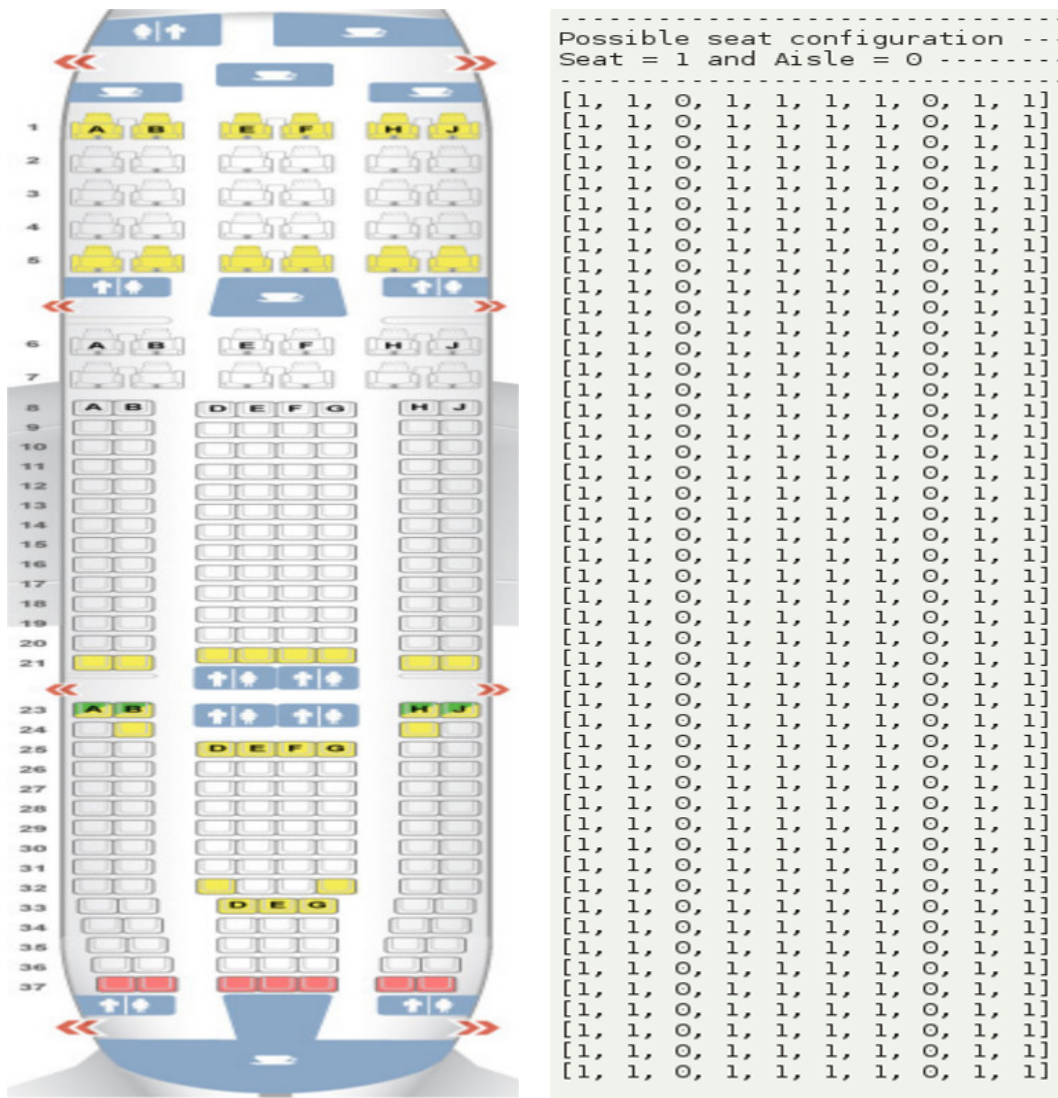
(a) Airbus A319 real seat disposition<sup>®</sup> (2014 TripAdvisor LLC<sup>®</sup>).

Possible seat configuration  
Seat = 1 and Aisle = 0 -----

[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]
[1, 1, 1, 0, 1, 1, 1]

(b) Airbus A319<sup>®</sup> seat disposition evaluated.

Figure 3.8: Possible seat configuration defined by the software for an Airbus A319<sup>®</sup> with 6 abreasts (right), and the real configuration (left).



(a) Airbus A330® real seat disposition (2014 TripAdvisor LLC©). (b) Airbus A330® seat disposition evaluated.

Figure 3.9: Possible seat configuration defined by the software for an Airbus A330® with 8 abreasts long (right), and a real configuration (left).

- For aeroplanes having a maximum payload capacity of 7,500 pounds or less and having a seating capacity of more than 19 but less than 51 passengers, one flight attendant;
- For aeroplanes having a seating capacity of more than 50 but less than 101 passengers, two flight attendants;
- For aeroplanes having a seating capacity of more than 100 passengers, two flight attendants plus one additional flight attendant for each unit (or part of a unit) of 50 passenger seats above a seating capacity of 100 passengers.

After the evaluation of the number of passengers and cabin crew members, the code calculates the total payload mass and the crew mass multiplying the number of people, pilots and cabin crew members with their respective weight defined in table B.3. It is important to mention that the crew members mass is already included inside the operating empty mass (3.3.2), and already considered with equation 3.11. The user can fix the maximum payload, and if the value defined is higher than the total passengers mass the code automatically reduces the number of passengers. Later on in the code after the evaluation of the maximum fuel mass (ch.:3.3.4), the system evaluates if extra payload mass can be added as cargo to reach the maximum take-off mass (ch.:3.3.1).

### 3.3.4 Fuel mass

After the estimation of the MTOM, OEM and the payload mass, the code evaluates the total fuel mass. If the user defines the geometrical data as raw input values, the software determines the maximum amount of fuel allowed as a fraction of the maximum take of mass, considering that for conventional aircraft the ratio between the MTOM and the maximum fuel mass ( $M_{f_{max}}$ ) is almost constant for each aircraft class 3.16:

$$\frac{M_{f_{max}}}{MTOM} = F \quad (3.16)$$

The  $F$  coefficient is derived from the analysis of the aircraft in the database, and its values can be seen in the table 3.5. It is important to state that this procedure is valid only if the CPACS geometry file is not used, and even if allows to estimate the fuel mass correctly is not suitable for optimisation. It is suggested to use the CPACS geometry file for the aircraft analysis and to use this feature only for early conceptual design to understand how the aircraft dimensions influence the different masses.

If the code receives the CPACS geometry file, it evaluates the fuel volume directly multiplying the wing volume (eq.: 3.4) for the fuel density ( $800\text{kg/m}^3$ ). The user can define, with the *maxFuelVol* option, the maximum total volume of the fuel tanks  $M_{f_{max}}$ . If the code evaluates a higher  $M_{f_{max}}$ , it reduces the fuel volume to the maximum value defined by the user.

The code estimates then the maximum fuel mass with maximum payload subtracting from the MTOM the OEM and the payload mass and checks if the fuel mass exceeds the allowed limit, and in case of positive response it reduces the fuel mass and increases the permitted payload mass. If the extra payload is higher than the maximum allowed by the user, the code reduces the MTOM keeping the OEM constant. The last mass that the code



Table 3.5: Coefficients for the definition of the maximum fuel mass in relation with the MTOM (values related to the conventional aircraft analysis only when the CPACS geometry is not used).

Aircraft	Geometry data		$F$
	$S_w$	$l_{fg}$	
Turboprop	>55	-	0.22
	<55	-	0.28
Turbofan	<90	<60	0.23
	<90	-	0.25
	<300	<35	0.28
	<300	>35	0.26
	<400	-	0.45
	<600	-	0.43
	>600	-	0.36

evaluates is the Zero Fuel Mass (ZFM), it subtracts the usable fuel mass to the maximum take-off mass. The usable fuel mass is the total fuel mass reduced by the percentage of fuel trapped inside the pumping systems, and that cannot be used to power the engine, this value is defined as default equal to 6% but can be modified by the user inside the CPACS file with the *resFuelPerc* option.

In the end, the code produces the log of all the quantities evaluated, writes the *Weight\_module.out txt* file and updates the CPACS file with the new information. If the user defines the aircraft without using the CPACS file the code creates a temporary version of the file with all the data required for the range analysis.

### 3.4 Weight analysis for unconventional aircraft

In this section, the process for the evaluation of the weights carried out by the general aircraft code is presented. It is important to notice that the original WB software can perform the analysis for both standard and novel aeroplane. The code, after the geometry evaluation (ch.:3.2.2 and ch.:3.2.3), estimates: the system mass ( $M_{sys}$ ), the passengers mass ( $M_{pax}$ ), the structure mass ( $M_{str}$ ), the fuel mass ( $M_f$ ), the engine mass ( $M_e$ ) and the crew members mass ( $M_{crew}$ ). The code also performs the estimation of the MTOM, OEM and ZFM masses.

The structure, passengers and fuel masses depends mainly on the geometrical parameter, and the code carries out their evaluation directly after the geometry analysis, as explained in the next section (ch.:3.4.1). The fuel, the engine and the crew members masses depend on the MTOM and ZFM value (see chapter 3.4.2); for this reason, the code performs an iteration process to estimate them.

The code starts guessing the MTOM, and it multiplies the wing loading (WL) for the aircraft wing planform area. The wing loading is the mass per unit of surface that the wing will need to withstand during flight, and can be directly evaluated from the lift ( $L$ ) equation (eq.:3.17).

Table 3.6: Example of wing loading value for aircraft type [35].

Aircraft	Type	MTOM [kg]	Wing Area [m <sup>2</sup> ]	WL [kg/m <sup>2</sup> ]
B777 - 200LR <sup>®</sup>	Turbofan	347452	539	645
ATR72 <sup>®</sup>	Turboprop	23000	61	377
Concorde <sup>®</sup>	Turbojet	176447	358	493

$$\frac{L}{S_w} = \frac{1}{2}\rho V^2 C_L, \quad (3.17)$$

where  $\rho$  is the air density,  $V$  is the aircraft speed and  $C_L$  is the lift coefficient. Dividing the previous equation for the gravity acceleration and considering the take-off conditions it is possible to obtain the wing loading relative to the MTOM.

For the weight analysis, at this point, the wing loading is merely a starting point, and so it is not calculated but can be directly defined by the user. The default value is 600kg/m<sup>2</sup>, and can be changed with the *wingLoading* option inside the CPACS file (at the end of the process the value is overwritten by the one evaluated with the final MTOM). A precise value of the WL is not required, and the iteration process will always converge to the same WL value; if the starting WL is too low (e.g. 200kg/m<sup>2</sup> for the B777<sup>®</sup> instead of 600kg/m<sup>2</sup>) the payload mass results higher than the MTOM and the code automatically modifies the value to be able to start the iteration process. In table 3.6 is presented the different values for the wing loading with the aircraft type.

Once the code starts the iteration process, it evaluates the systems, crew members and fuel masses and estimates the OEM ( $M_{OE}$ ), MTOM ( $M_{MTO}$ ) and ZFM ( $M_{ZF}$ ) with equation 3.18.

$$\begin{cases} M_{MTO} = M_{OE} + M_{pay} + M_{f_{mp}}, \\ M_{OE} = M_{sys} + M_{crew} + M_e + M_{str} + M_{f_{unusable}}, \\ M_{ZF} = M_{OE} + M_{pay}. \end{cases} \quad (3.18)$$

Where  $M_{f_{mp}}$  is the fuel mass with maximum passengers and  $M_{f_{unusable}}$  is the unusable fuel mass.

Once the old and new MTOM evaluated have a difference lower than 0.001 the iteration process ends and the code saves all the data including the new WL estimated inside the CPACS file. The code prints out the analysis logs and it writes all the data inside the *Weight\_unc\_module.out.txt* file and inside the CPACS file.

### 3.4.1 Structure, payload and fuel masses

In this section, all the routines that evaluate the structure, passengers and fuel masses outside the iteration cycle are explained.

#### Structure mass

From the conventional aircraft analysis, it was possible to derive an equation to relate the aircraft total wetted surface with the structure mass ( $M_{structure}$ , 3.19):

$$M_{structure} = \rho_{mvert} t_{mvert} (A_w + A_f)^k \quad (3.19)$$

where  $\rho_{\text{mvrt}}$  is the mean virtual material density (default: 2700 kg/m<sup>3</sup>),  $t_{\text{mvrt}}$  is the mean virtual thickness (default: 0.00014263), is a coefficient equal to 1.6276.  $A_w$  and  $A_f$  are respectively the wings and fuselages wetted area. The hypothesis behind this is that all the aircraft fuselage and wing structures have the same density and all the structures fill a virtual volume evaluated through the virtual thickness. The code uses the default values mentioned before, but the user can modify the *virtualDensity* and *virtualThick* options inside the CPACS file. In chapter 4.4 is presented the effect of the  $t_{\text{mvrt}}$  on the OEM. In all the aircraft tested, except for the Concorde<sup>®</sup>, the default value are used.

## Fuel mass

The fuel mass ( $M_f$ ) evaluation is just a multiplication of the fuel tank volume inside the wings and fuselages to the fuel density defined by the user (default value equal to 800kg/m<sup>3</sup>). The primary process for the fuel volume evaluation is inside the geometry analysis explained in chapter 3.2.2 or 3.2.3. The user can define a limit for the total fuel volume, and if the code evaluates a higher amount, it automatically reduces the value obtained. The code then evaluates the maximum fuel mass [kg] that can be boarded with maximum payload ( $M_{\text{fmp}}$ , eq.:3.20):

$$M_{\text{fmp}} = FPM \cdot M_f, \quad (3.20)$$

where *FPM* (fuel percentage with max payload) is a coefficient generally equal to 80 for turbofan and 50 for turboprop; those value are estimated with the aircraft in the database. Long range aircraft, like the Boeing777<sup>®</sup> or the Concorde<sup>®</sup>, usually are designed to have the fuel tanks almost full with maximum payload and so the *FPM* coefficient can be set to 95 (see chapter 3.1.2). The code automatically modifies the *FPM* value with the aircraft type while the user should change the *FPM* from 80% to 95% for long-range aircraft.

## Payload mass

The last quantity evaluated outside the iterative process is the payload mass since it depends only on the cabin geometry and on the passenger surface density; the code estimated the ( $Nb_{\text{pax}}$ , eq.:3.21) with the cabin area ( $S_c$ ) evaluated during the geometry analysis and the passenger surface density ( $\rho_{\text{pax}}$ ).

$$Nb_{\text{pax}} = S_c \cdot \rho_{\text{pax}}. \quad (3.21)$$

The passenger surface density is generally different for each aircraft but a default value of 1.66pax/m<sup>2</sup> is provided and in table 3.7 there are few values relative to the most common turbofan and turboprop aircraft as well as the passenger surface density of the Concorde<sup>®</sup> and BWB aircraft. The user can modify the  $\rho_{\text{pax}}$  using the *passDensity* option inside the CPACS file. As it is possible to notice the Concorde<sup>®</sup> has a low passenger surface density due to the presence of only business seats, in general, an aircraft with exclusively business class will have a passenger surface density of 0.5pax/m<sup>2</sup> lower than its configuration with only economy seats. The blended wing body density, instead, has been determined considering the conventional passenger seat dimensions (see tab.:B.3) and checking a possible seat layout with standard seat and aisle size to find a suitable seat disposition. From

Table 3.7: Passengers density variation with aircraft type and cabin area.

Conventional				
Aircraft	Engine type	Cabin area [m <sup>2</sup> ]	Pax number	$\rho_{\text{pax}}$ [pax/m <sup>2</sup> ]
B777 <sup>®</sup>	Turbofan	265	440	1.66
ATR72 <sup>®</sup>	Turboprop	51	70	1.39
Unconventional				
Aircraft	Engine type	Cabin area [m <sup>2</sup> ]	Pax number	$\rho_{\text{pax}}$ [pax/m <sup>2</sup> ]
Concorde <sup>®</sup>	Turboprop	103	120	1.16
BWB	Turbofan	320	540	1.69

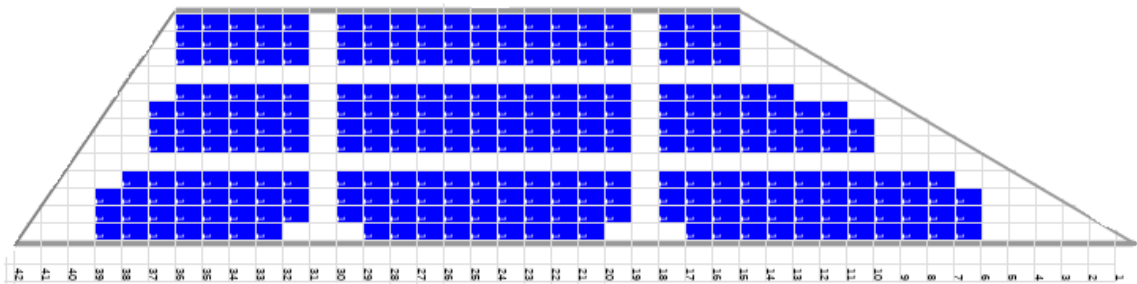


Figure 3.10: Possible seat disposition for the Blended Wing Body aircraft, view of half of the cabin area, cockpit on the right of the image (Blue rectangle = seats, white rectangle = empty space).

3.10 it was found that the density is close to the B777<sup>®</sup>, in general, is not recommended to place more than 1.8 passengers per square meter. The user can also define a *floorNumber* ( $Nb_{\text{floors}}$ ) higher than 1, the number of passenger is then multiplied by the factor ( $MF$ ) shown in equation 3.22. The code is not evaluating a second floor, but it considers that it will probably have only half of the passenger of the first floor (average value between the B747<sup>®</sup> and the A380<sup>®</sup> aircraft).

$$MF = 0.5Nb_{\text{floors}} + 0.5. \quad (3.22)$$

Subsequently the code multiplies the number of passengers for the defined mass ( $M_{\text{pass}}$ ) and saves the value obtained. The single passenger mass can be modified with the *pass-Mass* option and the default value of 105.0kg is the same used for conventional aircraft. The user can also define the cargo mass, if known, modifying the *massCargo* option inside the CPACS and the code sums the value with the passenger mass just estimated. It is important to notice that the code does not check if the cargo fits on a possible cargo bay; this last parameter should be used when the total payload mass is known in advance. In the end, the user can also define the maximum payload mass and the maximum passenger number with, respectively, the *maxPayload* and *passNb* options inside the CPACS file. The code performs an over-constrained check, but the user should be aware of the value defined.

### 3.4.2 Crew members, engines and systems masses

The crew members, engines and systems masses are evaluated inside the iteration process since they all depend on the aircraft mass. Since the cabin crew members and pilots number, as well as their respective masses, can be found in the ICAO and FAA regulations, the code uses, both for conventional and unconventional aircraft, the same routine explained in chapter 3.3.3.

#### Engine mass

The engine analysis starts from the definition of an equation that relates the engine mass ( $M_e$ ) to the maximum thrust ( $T_{\max}$ ) or power output ( $P_{\max}$ ) that a turbofan or a turboprop engine can generate. The two different equations (eq.:3.23) are plotted in figure 3.11 alongside the engines used to create the trend lines. As it is possible to notice the correlation is linear and a large number of engines is not required to have reliable results. All the information regarding the engine power, thrust and mass can be found inside the respective EASA certificate (e.g. *PW100 series engine, Type certificate data sheet* [36]).

$$\begin{cases} M_e = 16.948T_{\max} + 447.985, & \text{Turbofan} \\ M_e = 0.221P_{\max} + 80.986, & \text{Turboprop} \end{cases} \quad (3.23)$$

The maximum thrust is the take-off thrust and, generally, it can be considered four times the cruise thrust produced ( $T_{\text{cru}}$ , eq.:3.24). For a single engine results:

$$T_{\text{cru}} = \frac{\text{MTOW}}{LD} \frac{1}{1000Nb_e}; \quad (3.24)$$

where  $Nb_e$  is the number of engines, MTOW is the maximum take-off weight [kg] and the 1000 coefficient at denominator allows the conversion from N to kN.

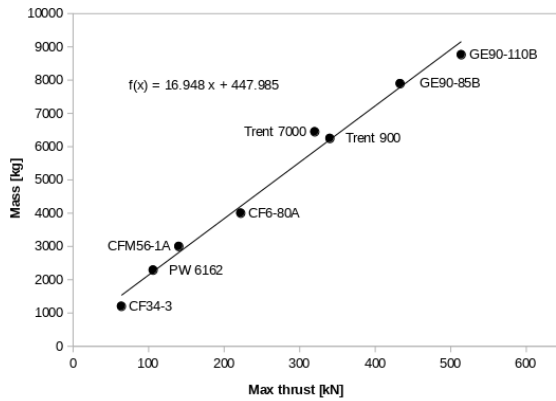
The output power [kW] produced by a turboprop engine, instead, results (eq.:3.25):

$$P_{\max} = \frac{T_{\max} V_{\text{cru}}}{\eta_{\text{prop}}}, \quad (3.25)$$

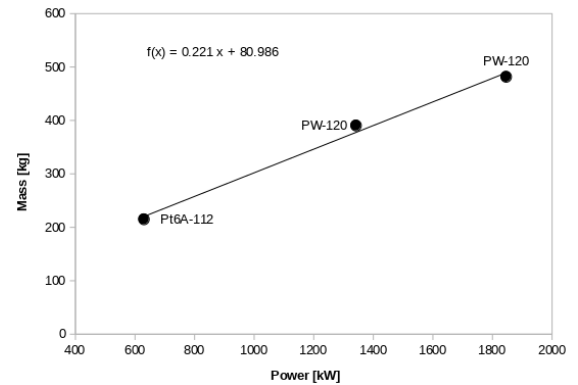
where the  $T_{\max}$  is evaluated as two times the  $T_{\text{cru}}$  (eq.:3.24), and  $\eta_{\text{prop}}$  is the propeller efficiency chosen equal to 0.85. The code calculates the total engine's mass by multiplying the value found with the number of engines.

As it is possible to notice, the code uses the MTOW instead of the actual aircraft mass in cruise conditions. Since the weight module cannot perform the mission analysis, it is not possible to have a precise evaluation of the weight losses. Using the MTOW generates an approximation error that causes the engine mass to be slightly overestimated, but the order of magnitude of this error is negligible with the aircraft MTOM.

The  $Nb_e$  and the engine type are respectively 2 and turbofan by default. The user can control those values defining in the CPACS file the *engineNumber* and the *turboprop* (False or True) options. The user can also define the engine entirely and avoid all the previous evaluation. In this last case, the user must define the engine number and the engine type and has to set the *userEngineOption* as True; moreover for each engine, separately, the user must determine the mass and the max thrust (the two value must be equal for all engines) inside the CPACS file.



(a) Turbofan engines.



(b) Turboprop engines.

Figure 3.11: Engine mass variation in relation with the maximum available thrust for turbofan (left) and with the maximum output power produced for turboprop engines (right).

### System mass

The systems in an aircraft must be reliable, efficient and with low maintenance costs and they include:

- the auxiliary power unit (APU), the pneumatic and air conditioning ( $W_{s1}$ ),
- the anti-icing ( $W_{s2}$ ),
- the flight controls and hydraulics ( $W_{s3}$ ),
- the furnishings and equipment ( $W_{s4}$ ),
- the landing gears ( $W_{s5}$ ),
- the flight instruments ( $W_{s6}$ ),
- the loads and handling ( $W_{s7}$ ),
- the avionics ( $W_{s8}$ ),
- the electrical cabling and components ( $W_{s9}$ ),
- the engine systems ( $W_{s10}$ ).

Each components is optimised for an aircraft during the detailed design phase but, especially for preliminary design, their weight is estimated with semi-empirical correlations derived from years of testing and conventional aircraft design. Since the systems technologies do not depend directly on the aircraft category but merely on the progress of the respective science and engineering fields, it is possible to assume that all aircraft have conventional systems and the equation can be used for the general aircraft analysis.

The code evaluates the system's mass with the equations proposed in the Beltramo (1977) *parametric study of aircraft transport systems* [37]<sup>3</sup>, and evaluates the furnishing

<sup>3</sup>All the equation from the Beltramo article are in imperial units, and the code performs the conversion.

and equipment mass with the equation introduced by Omran Al-Shamma (2013) *Aircraft weight estimation in interactive design process* [38].

If the aircraft has or not an APU installed, the user can set, inside the CPACS file, the *auxiliaryPowerUnit* option to True or False. The code then evaluates the  $W_{s1}$  with equation 3.26.

$$\begin{cases} W_{s1} = 13.6Nb_{\text{pax}} & \text{auxiliaryPowerUnit} = \text{False} \\ W_{s1} = 26.2Nb_{\text{pax}}^{0.944} & \text{auxiliaryPowerUnit} = \text{True} \end{cases} \quad (3.26)$$

The anti-icing systems mass depends on the engine type (turboprop or turbofan) and placement (wing mounted, or tail mounted). The user can set to True or False respectively the *turboprop* (False as default) and *wingMountedEngine* (True as default) options. For wing mounted engines the equation results (eq.:3.27):

$$\begin{cases} W_{s2} = 0.520A_w & \text{turboprop} = \text{True}, \\ W_{s2} = 0.238A_w & \text{turboprop} = \text{False} \end{cases} \quad (3.27)$$

where  $A_w$  is the main wing wetted area; for tail mounted engines both for turboprop and turbofan aircraft, the equation results (eq.:3.28):

$$W_{s2} = 0.436A_w \quad (3.28)$$

An aircraft can also be built with a single or a multi-hydraulics system, and the user can set the option *singleHydraulics* to True or False to choose between them, the default value is False since nowadays most of the aircraft uses multiple hydraulics controls. To evaluate the mass of the hydraulic system ( $W_{s3}$ ) the code uses in the first case equation 3.29. For the majority of aeroplanes that have multiple hydraulic systems  $W_{s3}$  mass also depends on the wing wetted surfaces ( $3000\text{feet}^2 \approx 279\text{m}^2$ , eq.:3.30).

$$W_{s3} = 45.0 + 0.269(A_w + 1.44A_{\text{tw}})^{1.106} \quad (3.29)$$

$$\begin{cases} W_{s3} = 45.0 + 1.318(A_w + 1.44A_t) & \text{if } (A_w + 1.44A_t) \leq 3000\text{feet}^2 \\ W_{s3} = 18.7(A_w + 1.44A_t)^{0.712} - 1620 & \text{if } (A_w + 1.44A_t) > 3000\text{feet}^2 \end{cases} \quad (3.30)$$

Where  $A_t$  is the total wetted surface of the wings except the main one (e.g. tail wing, horizontal stabiliser, winglets). It is important to notice that the previous equations take into account also the flight control system weight.

The furnishings and equipment mass [38] ( $W_{s4}$ ) depends directly on the zero fuel mass (ZFM) expressed in pounds (eq.:3.31):

$$W_{s4} = 0.196(\text{ZFM})^{0.91}. \quad (3.31)$$

The landing gear mass  $W_{s5}$  depends on the aircraft maximum take-off mass (MTOM) and results ( $108200\text{lb} \approx 49079\text{kg}$ ):

$$\begin{cases} W_{s5} = 0.0302\text{MTOM}, & \text{if } \text{MTOM} \leq 108200\text{lb} \\ W_{s5} = 0.0440\text{MTOM} - 672, & \text{if } \text{MTOM} > 108200\text{lb} \end{cases} \quad (3.32)$$

The code evaluates then all the masses of the other listed systems without the necessity of the user interaction and the equation are (eq.: 3.33).

$$\begin{cases} W_{s6} = 1.875Nb_{\text{pax}} + 0.00714M_{f_{\text{max}}} + (0.00145T_{\text{max}} + 30)Nb_e + 162, \\ W_{s7} = 50\text{lb}, \\ W_{s8} = Nb_{\text{pax}} + 370, \\ W_{s9} = 16.2Nb_{\text{pax}} + 110, \\ W_{s10} = 133Nb_e. \end{cases} \quad (3.33)$$

where  $Nb_{\text{pax}}$  is the number of passengers,  $Nb_e$  is the number of engines,  $T_{\text{max}}$  is the take off thrust and  $M_{f_{\text{max}}}$  is the maximum fuel mass loadable. At the end of the routine, the code sum all the masses evaluated and convert the value from lb to kg.

Some of the system equation just mentioned as well as the cabin crew member number and the engine mass depends directly on the aircraft MTOM and ZFM, and they are grouped inside the iteration process.

## 3.5 Range module

The range module calculates the fuel consumption during each flight phase (ch.:2.3.2), the cruise and maximum ranges. It also proposes a change in the number of crew members depending on the expected flight duration; since this last evaluation is meant to be only a recommendation, the code does not update the actual value determined during the weight analysis. The analysis does not depend directly on the aircraft geometry, and the module works for both the conventional and unconventional aircraft.

### 3.5.1 Fuel Consumption

The first analysis of the range module is the fuel consumption evaluation. The code roughly estimates the amount of fuel used for each flight phase and also evaluates if this is sufficient for the entire flight.

The analysis starts from the evaluation of the weight after landing ( $W_{\text{al}}$ ) that consists in the MTOW minus the fuel weight, except for the unusable fuel portion (RES) defined as 6% of the total fuel mass. The  $W_{\text{al}}$  results from the inverted equation 3.34, where  $W_f$  is the maximum fuel weight with maximum passengers.

$$\frac{W_f}{\text{MTOW}} = (1 + \text{RES}) \left( 1 - \frac{W_{\text{al}}}{\text{MTOW}} \right) \quad (3.34)$$

Subsequently it calculates (eq.:3.35): the weight after loiter ( $W_{\text{aloi}}$ ), the weight after cruise ( $W_{\text{acru}}$ ), the weight after take-off ( $W_{\text{ato}}$ ) and the weight after climb ( $W_{\text{acl}}$ ):

$$\begin{cases} W_{\text{aloi}} = W_{\text{al}}/0.995; \\ W_{\text{acru}} = W_{\text{aloi}} e^{(h_{\text{loi}}\text{SFC}_{\text{loi}})/(60LD_{\text{loi}})}; \\ W_{\text{ato}} = 0.995\text{MTOW}; \\ W_{\text{acl}} = 0.975W_{\text{ato}}. \end{cases} \quad (3.35)$$



Where  $h_{loi}$  is the loiter time (set equal to 30 minutes as default, ch.:2.3.2),  $SFC_{loi}$  and  $LD_{loi}$  are the specific fuel consumption and  $LD$  coefficient during loiter. The code is unable to recognise the presence of winglets directly from the geometry; for this reason, the user has the option to set their efficiency leading to a reduction of the SFC during the cruise phase (eq.:3.36):

$$SFC_{cru} = SFC_{cru} - 0.05WT, \quad (3.36)$$

where  $WT$  is the winglet type, equal to 0, 1 or 2 for respectively none, medium or high efficient winglets.

The maximum lift over drag coefficient is user input, and if not defined, the code will use 17 as default (average  $LD$  value from the aircraft in the database); the module then estimates the relative cruise and loiter  $LD$ . If the engine is a turboprop, the  $LD_{loi}$  is the 86.6% of the maximum  $LD$  while the one during the cruise  $LD_{cru}$  is the maximum  $LD$  (for turbojet and turbofan engines is the vice-versa [18]). Once all the weights have been estimated the software evaluates the mass of fuel after landing and checks if this corresponds to the unusable fuel reserve. There is a possibility that due to a lack of fuel, an excess of passengers or an inefficient engine the cruise fuel consumption becomes negative which means that the aircraft can not sustain the flight with the current settings. In this case, the code raises an exception to notify the problem.

### 3.5.2 Range evaluation

The evaluation of the range ( $R$ ) of the aircraft is entirely carried out using the Breguet equation (eq.:3.37), explained in detail in chapter 2.3.1.

$$R = \frac{3.6V_{cru}}{SFC_{cru}} LD_{cru} \ln\left(\frac{MTOW}{W_{al}}\right) \quad (3.37)$$

$V_{cru}$  is the cruise speed [m/s],  $MTOW$  and  $W_{al}$  are the maximum take off and after landing weights [N],  $SFC_{cru}$  and  $LD_{cru}$  are respectively the Specific Fuel consumption [1/hr] and the Lift over Drag coefficient evaluated at cruise conditions.

To be able to draw the payload versus range diagram (figure 3.12) the code computes the maximum possible amount of passengers considering maximum fuel boarded and evaluates the range under these conditions. In the end, the code calculates the range with the maximum amount of fuel and no payload on board (it reduces the  $MTOM$ ). It also checks if no payload is allowed when maximum fuel on board and displays a warning.

Even if the Breguet equation is meant to be used to estimate the cruise range, it can also evaluate the total flight range (from take-off to landing). It can overestimate the flight range of a percentage, usually, acceptable for the present module that is not meant to carry out a full mission analysis. Generally, the ground distance travelled during all the other flight phases is at maximum 10 or 15 percent of the total range. Figure 3.12 shows the cruise and maximum ranges versus payload diagram printed out by the program. Three straight lines compose the curve:

- the first is horizontal and represents the distance that can be travelled with maximum payload;

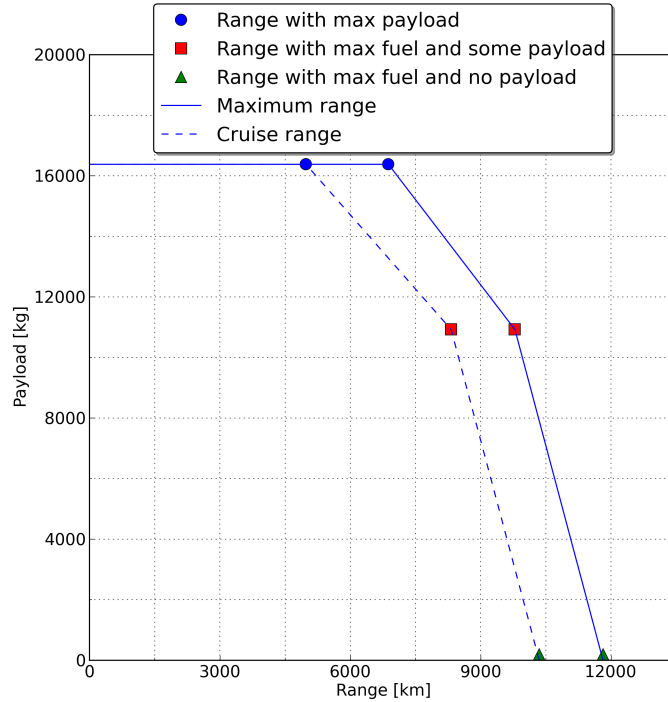


Figure 3.12: Example of the range vs payload plot from the range module (A319<sup>®</sup> aircraft tested).

- the second straight line is related to the increased range due to payload reduction and increased fuel mass;
- the last part of the curve is related to the range increment at reduced payload and with maximum fuel boarded.

The payload reduction also influences the differences between the maximum and cruise ranges, in figure 3.12 it can be seen that the two curves have a different slope that tends to become equal as the payload decrease.

### 3.5.3 Crew updating

Once the software evaluates the maximum range with maximum payload ( $r_{mp}$ ), it estimates the flight time  $t_{fl}$  [hr] and increases it by 10% to compensate for the low speed flight phases (eq.:3.38).

$$t_{fl} = 1.1 \frac{r_{mp}}{3.6V_{cru}} \quad (3.38)$$

with the cruise velocity  $V_{cru}$  expressed in [m/s].

In relation to the *Aeronautics and Space* section of the *Code of Federal Regulations* [39] the company should increase the crew members with the following minimum criteria:

- if  $t_{fl} < 8\text{hr}$  only two pilots are required;
- if  $8\text{hr} \leq t_{fl} < 12\text{hr}$  extra crew members (minimum 1) should be required;

- if  $t_{fl} \geq 12$  also an extra pilot and crew member are compulsory.

The code, in the end, shows the total crew mass increased with the extra members added inside the log file and the *Range\_module.out txt* file.

## 3.6 Balance module

The balance module computes the coordinates of the centre of gravity and the six independent moments of inertia ( $I_{xx}$ ,  $I_{yy}$ ,  $I_{zz}$ ,  $I_{xy}$ ,  $I_{yz}$ ,  $I_{xz}$ ) using the lumped masses method (chapter 2.2.2). Conceptually, it works equally for the conventional and unconventional aircraft analysis; there are different scripts for each module because for standard aircraft configurations we can always expect one fuselage and two main wings and the routines can be written more efficiently. For the unconventional aircraft analysis, the aircraft can have multiple fuselages or none at all, and the user can place the engine. This section explains the balance evaluation for the conventional case; the differences in the general analysis can be found in section 3.6.3.

Differently from the weight and range analysis, for conventional aeroplanes, the COG and MOI analysis cannot be carried out without the aircraft defined using the CPACS geometry format. It is important to mention that the mass disposition is considered uniform in each segment and the code does not take into account a possible luggage misplacement or an odd abreast number.

The code evaluates five different COG positions and MOI relative to five different possible aircraft configurations:

- no payload and fuel boarded (OEM condition);
- no fuel boarded (ZFM condition);
- no payload boarded (ZPM condition);
- maximum payload and relative fuel boarded (MTOM condition);
- user defined case with a specific fuel and payload amount, chosen as a percentage of their maximum allowed value.

If the user wants to evaluate the COG and MOI for a specific case, he/she have to set the *userCase* option True inside the CPACS file and define the *fuelPercentage* and *payload percentage*.

### 3.6.1 Centre of Gravity

The centre of gravity evaluation is the first step of the balance module. It gathers all the masses from the weight analysis and determines all the node positions. While for the fuselage each point is at the exact centre of each segment, for the wing the code locates the x-coordinate at the quarter chord and the y and z-coordinates at the centre of the segment. In figure 3.13b is possible to notice the lumped nodes used for the COG evaluation; the blue and green dots are respectively the wings and fuselage nodes, the red dot represents the COG position.

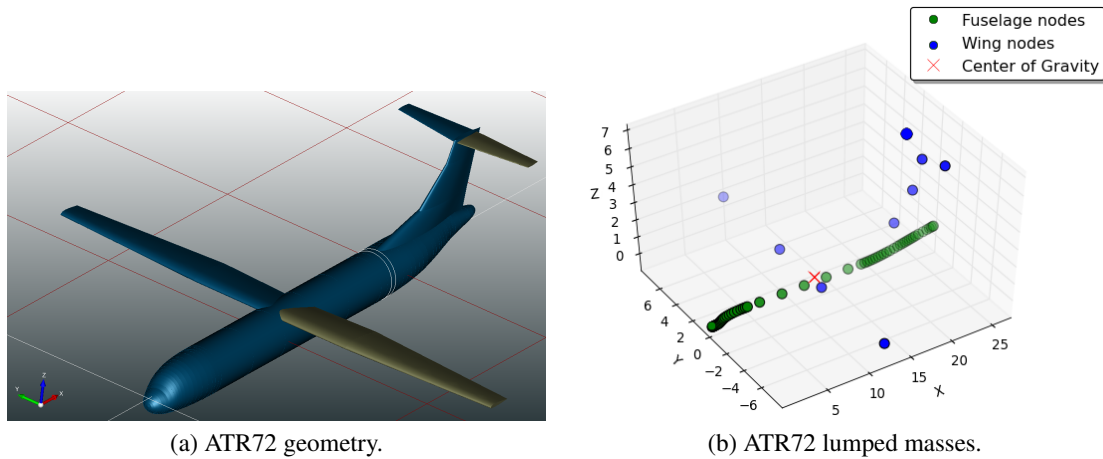


Figure 3.13: Geometry of the ATR72 aircraft (Left) with the lumped masses disposition for CG evaluation (Right).

The script then subdivides the total OEM in each segment proportionally to the relative volume, then allocates the payload into the fuselage cabin segments and, in the end, the fuel mass is placed in the horizontal main wings proportionally to their volume. The routine checks if the sum of the masses of each segment corresponds to the total MTOM and, if the response is negative, the missing mass is added respectively to the first segment of the main wing or horizontal tail stabiliser (if there are any). This last mass positioning allows to roughly simulate the engine placement under the main wing or near the fuselage tail. For this reason, in the CPACS geometry file, if the engine is near the aircraft tail, the *WingMountedEngine* must be set as False; the default value, on the contrary, is True since the majority of commercial aircraft have the engines mounted under the main wings.

Once the code has determined all the nodes and their respective masses, the centre of gravity can be calculated as explained in chapter 2.2.1. Figure 3.13b shows the lumped masses for the CoG analysis in comparison with the aircraft geometry and it also shows the position of the computed centre of gravity.

### 3.6.2 Moments of Inertia

For the analysis of the MOI, the lumped nodes defined for the evaluation of the COG are not enough and, for this reason, the code analyses the fuselage and wings geometry and subdivides them into a higher number of nodes. The system divides the fuselage segments in an equally distributed set of points and assigns to each one a fraction of the segment mass. The code places the wing nodes only on the profile perimeter, the wing sections are generally small, and the approximation of masses only on the surfaces is acceptable. The computational effort needed to set equally spaced nodes inside the wing is not paid back by the better quality of the solution. It is also important to mention that the program works for any geometry as the fuselage section, but it best fits the nodes on a circular or almost circular profile. The software equally spaces the nodes and does not check if they are symmetric to the aircraft symmetry axes. The uniform distribution of nodes is required from the assumption that each point represents the same volume fraction

of the segment to which it belongs. Thanks to this hypothesis it is possible to assign to each node the same mass fraction (eq.:3.39):

$$m_{ij} = ms_j/Nb_j, \quad (3.39)$$

where  $m_{ij}$  is the mass of the  $i^{\text{th}}$  nodes that belongs to the  $j^{\text{th}}$  segment,  $ms_j$  and  $Nb_j$  are the mass and the number of nodes of the  $j^{\text{th}}$  segment.

The user can define the node spacing (expressed in meters), but the code uses as default 0.05m to have a good solution with a small computational time. For the wing decomposition, the user can also define the number of subdivisions of the lower and upper profiles of each wing segment (default value is 30 chosen in relation to the computational time).

The optimal subdivision was obtained by analysing a cylinder of fixed radius and length, considering both the computational time and the accuracy of the solution. The cylinder has the length (L) of 12 meters, radius (R) of 4 meters and total mass (M) of 200 kg (values chosen to have a small integer MOI to perform numerous tests quickly). Its centre of gravity is at half of its length since the mass distribution is uniform.

The real inertia is evaluated using the classical equations for a solid cylinder derived from the integral one (3.40, [40]):

$$\begin{cases} I_{xx} = \frac{M}{2}R^2 = 1600 \text{ kgm}^2; \\ I_{yy} = \frac{M}{12}(3R^2 + L^2) = 3200 \text{ kgm}^2; \\ I_{zz} = I_{yy}; \\ I_{xy} = I_{yz} = I_{xz} = 0 \text{ kgm}^2. \end{cases} \quad (3.40)$$

As can be seen from table 3.8 and figure 3.14, when increasing the number of points (decreasing the node spacing) the solution converges to a value close to the analytical one. A node spacing of 0.05m gives almost the analytical solution (error lower than 0.1%) and for this reason, is defined as the default value for the code. The  $I_{xy}$  and  $I_{xz}$  moments are always zero, as expected, while the  $I_{yz}$  one tends to zero when increasing the nodes number.

Since the computational time is relevant, a spacing of 0.1m or 0.2m can also be used accepting a higher error. As it is possible to notice there are no values for a spacing smaller than 0.05m, the computational time required to complete the analysis in this case, even for a simple object, can exceed a minute without a noticeable improvement on the solution; furthermore with such a small spacing the array containing all the nodes exceeds its maximum computational dimension. Figure 3.15 shows an example of the lumped masses disposition of an entire aircraft.

### 3.6.3 Differences in the unconventional balance module

The unconventional analysis code can detect the presence of multiple fuselages and can assign their respective masses. A fuselage can be defined to carry either fuel or payload (as explained in chapter 3.2.2). Moreover, the code defines as fuel tank all the horizontal wings, differently from the conventional code that places the fuel only inside the main horizontal wings.

If the aircraft does not have any fuselage, like the BWB (fig.:1.3e), the code subdivides the wing into cabin and fuel tank segments and assign to each one a mass proportionally

Table 3.8: Analysis of the effect of different number of points for the evaluation of the moments of inertia [ $\text{kgm}^2$ ] of a test cylinder ( $R = 5\text{m}$ ,  $L = 12\text{m}$ ,  $M = 200\text{kg}$ ).

Nodes spacing [m]	Nodes number [-]	Ixx [ $\text{kgm}^2$ ]	Iyy [ $\text{kgm}^2$ ]	Izz [ $\text{kgm}^2$ ]	Iyz [ $\text{kgm}^2$ ]	Ixz=Ixy [ $\text{kgm}^2$ ]	time [s]
2.00	2.28E+02	2187.8	3460.1	3394.3	6.5	0.0	0.007
2.00	2.28E+02	2187.8	3460.1	3394.3	6.5	0.0	0.007
1.00	1.55E+03	1921.1	3336.3	3351.5	2.9	0.0	0.012
0.80	2.87E+03	1860.7	3317.5	3321.9	-5.4	0.0	0.015
0.40	2.08E+04	1728.9	3261.0	3262.6	-1.2	0.0	0.054
0.20	1.59E+05	1656.4	3227.9	3227.2	-0.4	0.0	0.343
0.10	1.24E+06	1618.9	3209.3	3209.2	-0.01	0.0	1.533
0.05	9.77E+06	1599.2	3199.6	3199.5	0	0.0	9.220
<b>Real Value</b>		1600	3200	3200	0	0	-

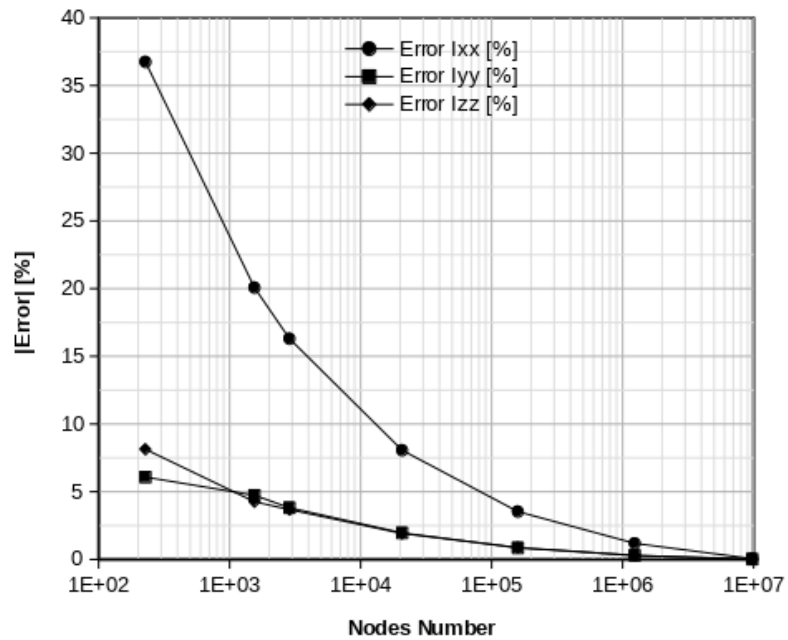
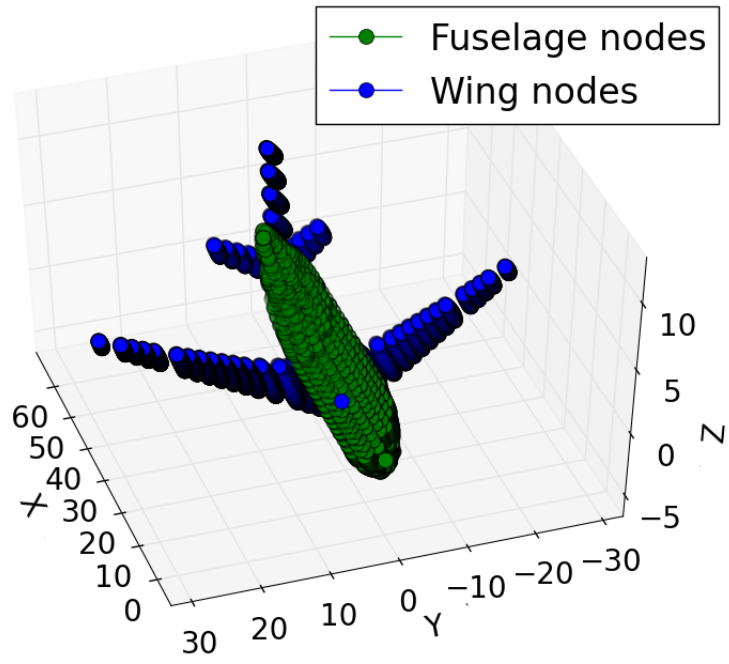
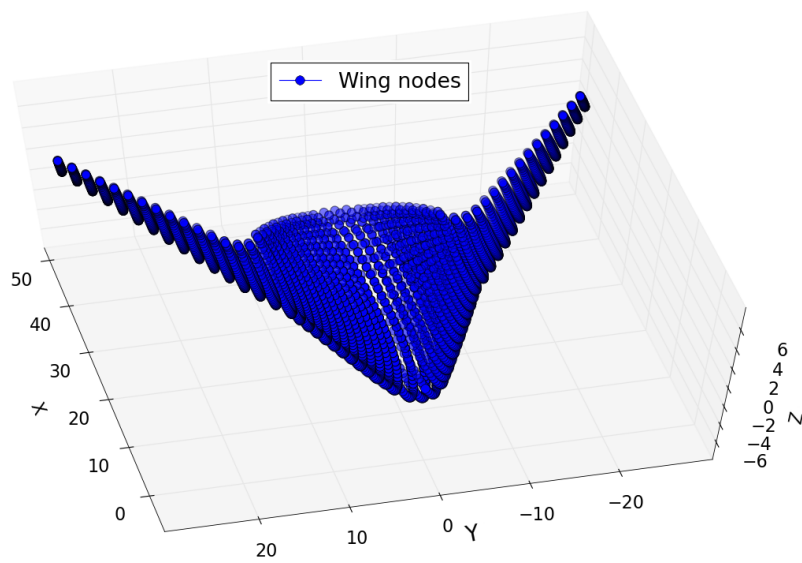


Figure 3.14: Error of the evaluated moments of inertia for the test cylinder (tab.:3.8) against the number of lumped nodes used.



(a) Boeing B777®.



(b) BWB aircraft.

Figure 3.15: Example of the lumped masses distribution with a spacing of 2m for the B777 (upper) and the BWB aircraft (lower).

to their volume (ch.:3.2.3). The code allocates the structure and system masses to each segment. The user can place fuel inside the cabin segment defining it as a percentage of the free volume left in the cabin segments (eq.:3.10). This last value, called *fuelOnCabin* inside the CPACS geometry file, is zero as default.

The MOI evaluation has the same theoretical base of the conventional analysis, and the code divides the aircraft into lumped masses. It considers multiple fuselages, if they are defined, or subdivides only the wings in the BWB case. The module subdivides the wing surface into lumped nodes (eq.:3.39).

The main feature, on the other hand, is the possibility to define precisely the positions and the mass of the engines influencing the COG and the MOI. The engine placement is not compulsory and if not specified the code distributes the propulsion systems mass in the main wing or near the aircraft tail horizontal stabiliser, in relation with the *WingMountedEngine* option. For the BWB the *WingMountedEngine* option is not considered since the plane has only one main horizontal wing. The engine mass is distributed in the central segments if the user does not define the exact placement. If the user wants to place each engine manually, he/she has to set the *userEnginePlacement* option as True inside the CPACS file and the engine x,y and z COG coordinate. The code takes into account the engine placement both for the COG and MOI evaluation.



# Chapter 4

## Results

This chapter presents results obtained by using the newly developed software on different aircraft. First, the software has been used to analyse the weight and range for a variety of different aircraft models that lead the market today. In the next step studies were made using the unconventional balance module to assess the influence of the placement of the main wing on the COG and the MOI. In a subsequent step, results obtained with the conventional WB for different aircraft are compared with the ones calculated with the unconventional WB code (note that this can only be done when the aeroplane is given as a CPACS geometry file). For the box-wing aircraft the influence on the OEM of the virtual thickness, passengers surface density and LD coefficient has been studied. And finally, using the software developed for unconventional aircraft, a BWB aircraft and the Concorde<sup>®</sup> aircraft were studied.

### 4.1 Conventional aircraft testing

The newly developed software for conventional aircraft was tested using several aircraft that are available on the market today. In this case, only the fuselage length and width, the wingspan and the wing area are needed as input for the software. All these aircraft are now included in the aircraft database, and the tests have been carried out also removing the aeroplanes from it one by one. For all the aircraft tested the code uses the default values defined in chapter 3.1.1 and listed in the appendix B.2. The range analysis is shown only for some of the Airbus<sup>®</sup> aircraft tested since if the LD coefficient and  $V_{\text{cru}}$  are appropriately defined, the software always estimates the range with an error lower than 5%.

Tables 4.1 and 4.2 present the results of the tests carried out for the majority of the Airbus<sup>®</sup> aircraft on the market. It is interesting to see that the evaluation of the MTOM, OEM and maximum fuel volume has an error always below 6%. These errors might be reduced by increasing the aircraft database and with a more complex fitting process. Table 4.3 presents the result of the analysis carried out for the Boeing<sup>®</sup> aircraft. The MTOM and OEM evaluation has an error smaller than 7% except for the B737-700<sup>®</sup> for which the fuel volume is underestimated by 9%. The number of passengers for the Boeing<sup>®</sup> aircraft shows a more significant error since the default values for the seat dimensions are based on the Airbus<sup>®</sup> aircraft values. This error can be reduced by changing the seat dimensions to the values for Boeing<sup>®</sup> aircraft. The default values are used for all aircraft to show

the code capability without user interaction. The last table (tab.:4.4) shows the results for small aircraft and the error has the same behaviour as for the previously discussed cases.

The tables show that the MTOM is strongly related to both the fuselage length and width but mostly to the wing area, and to the wingspan for small aircraft. The OEM is generally around 50% of the MTOM, and the same trend is expected for all kind of aircraft configurations. Since the fuel tanks are merely placed inside the main wings, the fuel volume result is strongly linked to the wing area and span. The passengers mass is, of course, dependent on the fuselage width and length. What is interesting to observe is that the code with the *isDoubleFloor* option (see ch.:3.1.1) estimates correctly the double-decker aircraft like the Boeing 747<sup>®</sup> and the A380<sup>®</sup>.

The code evaluates the Cessna C208<sup>®</sup> and the ATR42-300<sup>®</sup> perfectly since for small aircraft the relation between the MTOM and the aircraft dimensions is entirely linear.

The overall error is lower than 10% and, since the software is meant to be used only for conceptual design, it is considered satisfactory; but for a better and more detailed evaluation of the aeroplane characteristic, it is better to use an aircraft geometry defined in a CPACS file.

## 4.2 Effect of the wing position on the balance analysis

Before discussing the conventional and unconventional aircraft tested we present results of the MOI and COG tests carried out with the NASA CRM (Common research Model [41]) aeroplane using the unconventional WB software. The tests consist of the analysis of the variation of the centre of gravity and the moments of inertia with the main wings position. The default values are used for all the parameters in this test since the primary purpose is to demonstrate the code capability on the balance analysis and not to obtain the exact mass estimation. Table 4.5 shows that the aircraft should have an MTOM of approximately 270 tons and an OEM of 125 tons; the table also shows the zero fuel mass (ZFM) and the zero payload mass (ZPM).

Figures 4.1c, 4.1a and 4.1e show the three configuration studied. The first one represents the aircraft in its standard configuration (SC) while the second and the third one have the wing moved respectively 15 meters forward (FC) and 10 meters backwards (BC). It is important to mention that the unconventional WB software places the fuel both in the main wings and inside the horizontal tail stabiliser.

As expected, moving the main wing forwards causes the COG to shift forward while shifting the wings backwards results in a COG back shift; moreover, the COG displacement is proportional to the wing relocation; figures 4.1d, 4.1a and 4.1e show the COG position for the NASA CRM at MTOM. For the standard CRM configuration, the code estimates the centre of gravity at approximately the quarter chord of the main wing, while the FC has the COG with the same x-coordinates as the wing tip; with the BC the COG is found in front of the main wing. The y and z coordinates of the COG do not change in any configuration since the main wings are moved only by changing the x coordinate.

Table 3.8 lists all the COG and MOI that the WB software calculated with the aircraft at MTOM, ZFM, zero payload mass(ZPM) and OEM. If the fuel tanks are empty (OEM and ZFM configurations) the COG moves approximately 0.5 meters for every 10 meters of wing displacement from the initial position, and the moments of inertia are only

Table 4.1: Results from the Airbus® aircraft analysis using  $l_{fg}$ ,  $WDT_{fg}$ ,  $b$ ,  $S_w$  dimensions as data input, compared with the real value obtained from the manuals for airport planning (1 of 2).

<b>Manufacturer</b>		<b>Airbus®</b>				
<b>Type Model</b>		<b>A318 100</b>	<b>A319 neo</b>	<b>A320 neo</b>	<b>A321 neo</b>	<b>A330 200</b>
<b>Wing Area [m<sup>2</sup>]</b>		122.6	122.4	122.4	122.4	361.6
<b>Wing Span [m]</b>		34.1	35.8	35.8	35.8	60.3
<b>Fuselage Length [m]</b>		31.44	33.84	37.57	44.51	58.82
<b>Fuselage Width [m]</b>		3.95	3.95	3.95	3.95	5.64
<b>LD</b>		15	16.5	15	14	14
<b>MTOM [kg]</b>	<b>Real</b>	68000	75500	79000	97000	242000
	<b>Estimated</b>	69044	74497	81752	95475	244272
	<b>Error [%]</b>	1.54	-1.33	3.48	-1.57	0.94
<b>OEM [kg]</b>	<b>Real</b>	39500	42600	44300	50100	120600
	<b>Estimated</b>	39984	42880	44988	51891	123152
	<b>Error [%]</b>	1.23	0.66	1.55	3.57	2.12
<b>Max. Fuel Volume [l]</b>	<b>Real</b>	24210	26750	26730	32940	139090
	<b>Estimated</b>	23974	25867	26892	31406	138791
	<b>Error [%]</b>	-0.97	-3.30	0.61	-4.66	-0.21
<b>Max. Passenger</b>	<b>Real</b>	132	160	189	240	406
	<b>Estimated</b>	132	156	186	228	384
	<b>Error [%]</b>	0.00	-2.50	-1.59	-5.00	-5.42
<b>Range [km] at Max Payload</b>	<b>Real</b>	6000	4800	6300	4300	8000
	<b>Estimated</b>	5899	4849	6242	4251	8249
	<b>Error [%]</b>	-1.68	1.02	-0.92	-1.14	3.11

Table 4.2: Results from the Airbus® aircraft analysis using  $l_{fg}$ ,  $WDT_{fg}$ ,  $b$ ,  $S_w$  dimensions as data input, compared with the real value obtained from the manuals for airport planning (2 of 2).

Manufacturer		Airbus®				
Type Model		A330 300	A330 800neo	A330 900neo	A350 1000	A380 800
Wing Area [m <sup>2</sup> ]		361.6	395	395	443	845
Wing Span [m]		60.3	64	64	64.75	79.75
Fuselage Length [m]		63.67	58.82	63.66	73.78	72.72
Fuselage Width [m]		5.64	5.64	5.64	5.96	7.14
MTOM [kg]	Real	242000	251900	251000	316000	575000
	Estimated	248114	247446	252588	321666	578507
	Error [%]	2.53	-1.77	0.63	1.79	0.61
OEM [kg]	Real	129400	132000	137000	158800	277000
	Estimated	129708	129387	131858	164703	272222
	Error [%]	0.24	-1.98	-3.75	3.72	-1.72
Max. Fuel Volume [l]	Real	139090	139090	139090	158791	253983
	Estimated	140974	140595	143516	171099	258262
	Error [%]	1.35	1.08	3.18	7.75	1.68
Max. Passenger	Real	440	406	440	440	853
	Estimated	432	384	432	441	850
	Error [%]	-1.82	-5.42	-1.82	0.23	-0.35

Table 4.3: Result from the Boeing<sup>®</sup> aircraft analysis using  $l_{fg}$ ,  $WDT_{fg}$ ,  $b$ ,  $S_w$  dimensions as data input, compared with the real value obtained from the manuals for airport planning.

Manufacturer		Boeing <sup>®</sup>				
Type		<b>737</b>	<b>737</b>	<b>737</b>	<b>787</b>	<b>747</b>
Model		<b>700</b>	<b>800</b>	<b>900ER</b>	<b>8</b>	<b>8</b>
Wing Area [m <sup>2</sup> ]		125	125	125	360	554
Wing Span [m]		34.3	34.3	34.3	60	68.5
Fuselage Length [m]		33.6	39.5	42.1	56.7	76.4
Fuselage Width [m]		3.77	3.77	3.77	5.77	6.5
MTOM [kg]	Real	70000	79000	85200	227900	447700
	Estimated	68552	79346	85292	232641	426037
	Error [%]	-2.07	0.44	0.11	2.08	-4.84
OEM [kg]	Real	37648	41413	44677	119950	220128
	Estimated	39721	43769	46777	117747	205443
	Error [%]	5.51	5.69	4.70	-1.84	-6.67
Max. Fuel Volume [l]	Real	26022	26022	29666	126206	238610
	Estimated	23803	26101	28057	132182	226616
	Error [%]	-8.53	0.30	-5.42	4.74	-5.03
Max. Passenger	Real	149	189	220	381	605
	Estimated	130	170	185	360	600
	Error [%]	-13	-10	-16	-6	-0.83

Table 4.4: Result from different aircraft analysis using  $l_{fg}$ ,  $WDT_{fg}$ ,  $b$ ,  $S_w$  dimensions as data input, compared with the real value obtained from the manuals for airport planning.

Manufacturer		Embraer <sup>®</sup>	Embraer <sup>®</sup>	Bombardier <sup>®</sup>	ATR <sup>®</sup>	Cessna <sup>®</sup>
Type		<b>ERJ145</b>	<b>E190</b>	<b>CS300</b>	<b>42-300</b>	<b>C208</b>
Model		<b>LR</b>	<b>LR</b>	<b>0</b>	<b>Turboprop</b>	
Wing Area [m <sup>2</sup> ]		51.2	92.53	112.3	54.5	25.9
Wing Span [m]		22.04	28.72	35.1	24.57	15.88
Fuselage Length [m]		29.87	36.24	38.7	22.67	11.46
Fuselage Width [m]		2.28	3.01	3.7	2.8	1.73
MTOM [kg]	Real	22000	51800	67600	16900	3629
	Estimated	22709	52346	67799	16900	3629
	Error [%]	3.22	1.05	0.29	0.00	0.00
OEM [kg]	Real	12114	27837	37081	10285	2145
	Estimated	12787	29852	37872	10312	2145
	Error [%]	5.56	7.24	2.13	0.26	0.00
Max. Fuel Volume [l]	Real	6398	16250	21918	5625	1257
	Estimated	6601	17219	22302	5868	1226
	Error [%]	3.17	5.96	1.75	4.32	-2.47
Max. Passenger	Real	87	114	160	48	9
	Estimated	87	128	165	68	10
	Error [%]	0	12	3	42	10

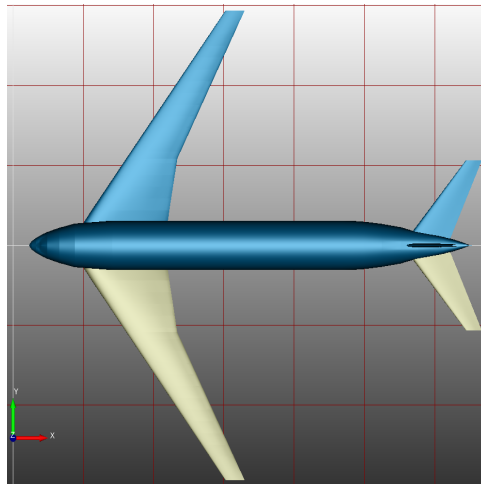
marginally affected. As soon as the tanks are filled with fuel, the COG is affected by the wing position, and it moves approximately 3 meters every 10 meters of wing displacement from the initial location. The presence of the payload inside the cabin permits to stabilise the COG position; looking at the ZPM configuration, the COG is 1 meter closer to the aircraft nose for the FC case and 1 meter further for the SC and BC cases than the COG at MTOM.

Looking at the MOI it is possible to see that the roll moment remains constant since the geometry of the aircraft is not changed relative to the x-axis, and the  $I_{xy}$  and  $I_{yz}$  are zero since the aircraft symmetry is respected. The pitch and yaw moments instead are greatly influenced by the wing and COG displacements. The yaw and pitch moments of inertia are more significant since, for the BC and FC, the overall distance of the aircraft nodes to the COG is higher than for the standard configuration. For the FC case, the increase of the MOI is substantial, while for the BC it is less pronounced. Since the wing in the BC case is behind the COG, the  $I_{xz}$  moment is, as expected, one order of magnitude lower than obtained for the SC case. At ZFM and OEM, the moments of inertia differ from the standard ones by a lower amount, but they follow the same trends as just discussed.

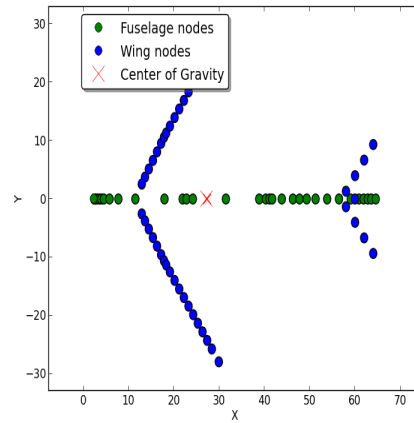
This discussion shows the importance of fuel placement and explains why having the possibility of moving fuel between tanks is crucial during flight. This test also demonstrates the balance module capability on analysing different geometries and providing reliable results.

Table 4.5: NASA CRM weights estimation.

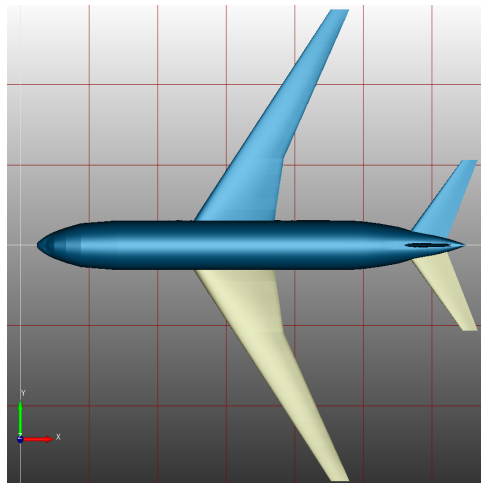
<b>Weight [kg]</b>	
MTOM	268906
ZFM	172656
ZPM	220501
OEM	124251



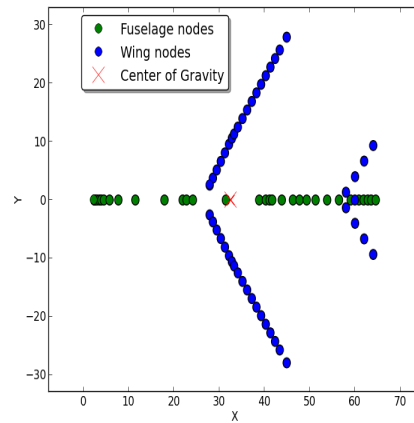
(a) NASA CRM aircraft full view (main wing front shifted).



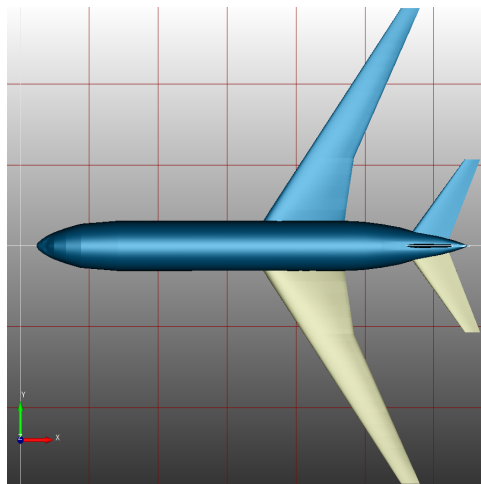
(b) NASA CRM aircraft COG (main wing front shifted)



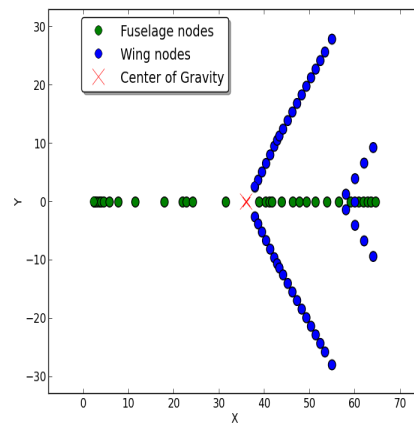
(c) NASA CRM aircraft full view.



(d) NASA CRM aircraft COG.



(e) NASA CRM aircraft full view (main wing back shifted).



(f) NASA CRM aircraft COG (main wing back shifted).

Figure 4.1: Centre of gravity variation in relation with the main wing position.

Table 4.6: Moments of inertia and centre of gravity variation in relation with the main wing position.

Max Payload (MTOM)		MOI [kgm <sup>2</sup> ]				
Wing Placement (x,y,z) [m]	COG (x,y,z) [m]	Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Front (9.5, 0, 4.4)	(27.3, 0.0, 5.3)	1.12E+07	6.07E+07	7.07E+07	0	2.20E+06
Standard (24.5, 0, 4.4)	(32.5, 0.0, 5.3)	1.12E+07	4.51E+07	5.51E+07	0	1.03E+06
Back (34.5, 0, 4.4)	(36.0, 0.0, 5.3)	1.12E+07	5.00E+07	6.00E+07	0	2.56E+05
ZFM		MOI [kgm <sup>2</sup> ]				
Wing Placement (x,y,z) [m]	COG (x,y,z) [m]	Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Front (9.5, 0.0, 4.4)	(30.6, 0.0, 5.6)	1.70E+06	3.58E+07	3.67E+07	0	6.11E+05
Standard (24.5, 0.0, 4.4)	(31.3, 0.0, 5.6)	1.70E+06	3.42E+07	3.51E+07	0	4.67E+05
Back (34.5, 0.0, 4.4)	(31.8, 0.0, 5.6)	1.70E+06	3.51E+07	3.60E+07	0	3.71E+05
ZPM		MOI [kgm <sup>2</sup> ]				
Wing Placement (x,y,z) [m]	COG (x,y,z) [m]	Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Front (9.5, 0, 4.4)	(26.2, 0.0, 5.2)	1.33E+07	5.81E+07	7.04E+07	0	2.47E+06
Standard (24.5, 0.0, 4.4)	(33.2, 0.0, 5.2)	1.33E+07	3.93E+07	5.16E+07	0	1.24E+06
Back (34.5, 0.0, 4.4)	(37.9, 0.0, 5.2)	1.33E+07	4.20E+07	5.43E+07	0	4.13E+05
OEM		MOI [kgm <sup>2</sup> ]				
Wing Placement (x,y,z) [m]	COG (x,y,z) [m]	Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Front (9.5, 0.0, 4.4)	(30.9, 0.0, 5.6)	1.48E+06	2.79E+07	2.88E+07	0	5.88E+05
Standard (24.5, 0.0, 4.4)	(31.9, 0.0, 5.6)	1.48E+06	2.62E+07	2.71E+07	0	4.43E+05
Back (34.5, 0.0, 4.4)	(32.6, 0.0, 5.6)	1.48E+06	2.70E+07	2.79E+07	0	3.46E+05

### 4.3 Conventional aircraft analysis

With the aircraft geometry defined in the CPACS format as a *XML* file, the analysis has been carried out with both the conventional and unconventional WB software. The results are obtained for the ATR72<sup>®</sup> and WP46RM turboprops and for the B777<sup>®</sup> turbofan; for each aircraft, the two codes calculate the weights, the ranges, the COG position and the MOI (the range and COG figures proposed are generated with the unconventional software). A comparison between the results obtained with the two different approaches and the real data is also provided. For each aircraft a list of the input data used is given; when a parameter is not mentioned the default value is used and can be found in chapter 3.1.1 and in the tables in appendix B.2.

#### 4.3.1 ATR72<sup>®</sup>

The ATR72<sup>®</sup> (fig.:4.2a) is the iconic European turboprop built by the *Avions de Transport Régional* manufacturer. The ATR is a joint venture between the French Airbus and the Leonardo S.p.A. manufacturers with the goal of producing a low-medium and efficient regional aircraft series. The ATR72-600<sup>®</sup>, analysed in this thesis, mounts on the main wings 2 Pratt & Whitney Canada PW127M turboprop engines, has a  $l_{fg} \approx b \approx 27\text{m}$  and the wing area is  $61\text{m}^2$ . The cabin has a maximum width of 2.5m, and the average fuselage width is  $\approx 1.83\text{m}$ . The aircraft is designed to carry 70 passengers in a single class configuration for a maximum of approximately 1500km with 5 tons of fuel boarded. The MTOM is 23 tons while the OEM is 13.2 tons.

For the ATR72-600<sup>®</sup> the code uses the input parameter defined in tables 4.7, and for the other quantities not mentioned it uses the default values. The aircraft does not have an APU, and the relative option is set to False, old ATR72 models could have a single



hydraulic system and the *singleHydraulic* option is defined True. The cruise speed is set to 141m/s and the *LD* coefficient is set to 15. The unconventional code uses a passenger surface density of 1.39 to estimate the number of passengers. For both the codes no restriction on any mass is defined.

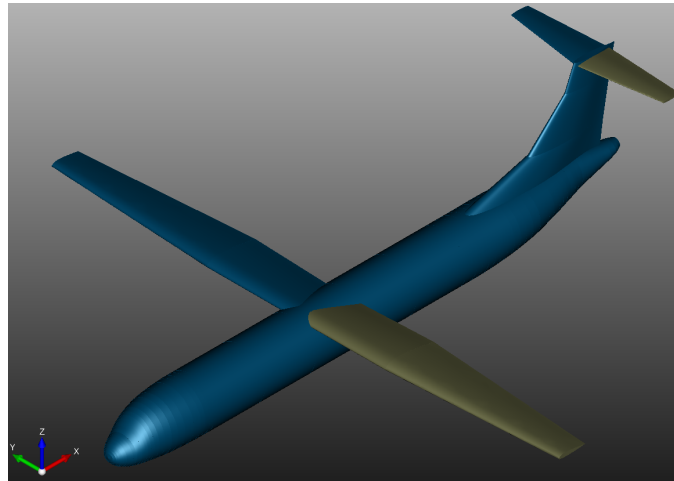
Tables 4.9 and 4.10 show all the results obtained with the conventional and unconventional WB software together with the real values [42]. First of all, one can see that the geometry analysis output matches the real aircraft dimensions. The MTOM, ZFM and the OEM match the real values with an error lower than 3%. The conventional code uses a semi-empirical equation that cannot be calibrated correctly for all aircraft and thus leads to an error of 3.4%. The CWB slightly exceeds the payload maximum since it evaluates two possible extra passengers with the given seat disposition. The unconventional WB (UWB), instead, evaluates the correct number of passengers but underestimates the maximum payload, since no cargo mass is provided as input. Both of the software agree to the crew members and the number of lavatories, and the wing loading is almost equal. It is important to state that the CWB code evaluates that the aircraft can carry a maximum of 72 passengers in 24 rows with three abreasts divided by one aisle. The real aircraft usually has longer and narrower seats, and for this reason, it has 18 rows and 4 abreasts. The number of lavatories should be 1, and the aeroplane needs four crew members (2 pilots + 2 cabin crew members). No engines are defined as default, and the UWB evaluates the total mounted engine mass; it also assesses the structure and system masses; it is interesting to see that the aircraft structure is the 60% of the OEM and one-third of the MTOM.

The range analysis provides a simple fuel consumption evaluation during the different flight phases, and the small difference in results obtained with the two software modules is merely related to the slightly different MTOM. The highest fuel consumption is during the cruise phase, and the loiter and the climb phases have almost the same fuel demand (see chapter 2.3.2). The climb last for a short time with the engines almost at maximum thrust, while the loiter phase, in this case, is characterised by a 30 minutes flight at reduced speed and thus at higher engine efficiency.

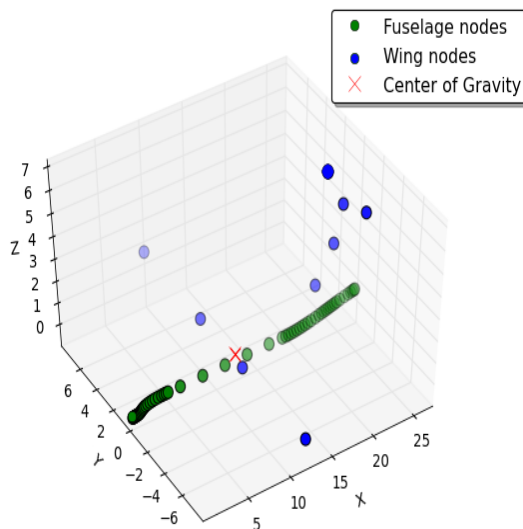
Both the code slightly underestimate the maximum range with maximum payload, but they both reproduce the range versus payload analysis reasonably. Figure 4.2c, obtained from the unconventional analysis, shows the variation of the cruise and maximum ranges in relation with the payload; it is possible to observe a range between 1500 km and 4000 km, which is a typical value for a low-mid range turboprop such as the ATR72 [42].

Figure 4.2b and table 4.8 show that the centre of gravity is evaluated between the quarter chord and the middle of the aircraft length. The CoG is placed precisely on the y-axis and has a little offset on the z-axis mainly because the wing, and so the fuel, are placed on the top of the fuselage. It is possible to notice that at OEM condition, primarily due to the rear horizontal stabiliser dimensions, the COG has the most backward position; if only fuel is added, the COG is moved to the most forward, and high position since the fuel tanks are only inside the main wings. Without fuel and with maximum payload the COG is found around the centre of the wing chord and, as expected, has the lowest z-coordinate; the COG at MTOM is placed in between the ones at ZPM and ZFM. The moments of inertia seems also be reasonable since the yaw moment is the highest one followed by the pitch and roll MOI; moreover, the  $I_{zz}$  is almost the sum of the  $I_{xx}$  and  $I_{yy}$  ones. As expected, since the aircraft is symmetric, the  $I_{xy}$  and  $I_{yz}$  MOI are zero and the  $I_{xz}$  moment as an order of magnitude less than the roll, pitch and yaw moments. Since for

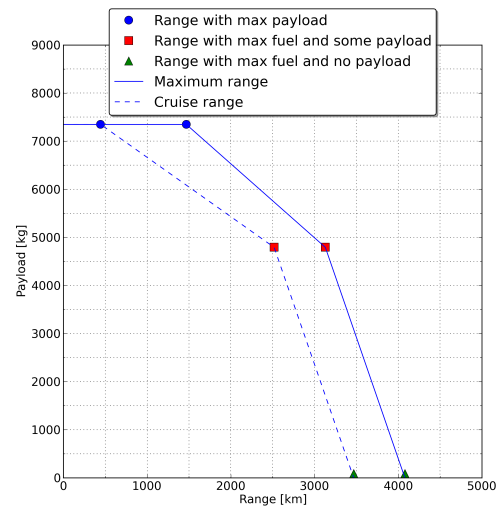
the UWB no engine placement is defined, it is interesting to see that the COG and MOI are almost equal with the one evaluated with the CWB.



(a) Full view of the ATR72<sup>®</sup> geometry.



(b) Centre of gravity and lumped nodes.



(c) Range versus payload (from Unconventional WB).

Figure 4.2: ATR72 aircraft analysis, full aircraft (4.2a), centre of gravity (4.2b) and range versus payload (4.2c).

Table 4.7: CWB and UWB modules parameters defined for the ATR72<sup>®</sup> analysis.

ATR72 <sup>®</sup> Parameters			
Unconventional WB			
Weight		Engine analysis (User Engine = False)	
Name	Value	Name	Value
$t_{mvrt}$	0.00014263	TP	True
$\rho_{mvrt}$ [kg/m <sup>3</sup> ]	2700	$Nb_e$	2
$h_{cabin}$ [m]	2	$M_e$ [kg]	-
SH	True	$T_{max}$ [kN]	-
$Nb_{floors}$	1	$TSC_{cruise}$ [1/hr]	0.7
$FPM_T$	50	APU	False
$M_{payload_{max}}, M_{cargo}$ [kg], $Nb_{pax_{max}}$ [-]	0	WM	True
LD	16	Balance (User Engine Placement = False)	
$V_{cru}$ [m/s]	141	Name	Value
$\rho_{pax}$ [pax/m <sup>2</sup> ]	1.39	1st Engine Placement	-
$Nb_{pilots}$	2	2nd Engine Placement	-
$Vol_{f_{max}}$ [m <sup>3</sup> ]	0	Balance (User Case = False)	

Weight conventional		Range	
Name	Value	Name	Value
<i>isDoubleFloor</i>	0	WINGLET	False
$V_{cru}$ [m/s]	141	$V_{cru}$ [m/s]	141
$Nb_{pax_{max}}$	0	LD	16
$Nb_{pilots}$	2	$h_{loiter}$ [min]	30
$Vol_{f_{max}}$ [m <sup>3</sup> ]	0	$TSC_{cruise}$ [1/hr]	0.7
$M_{cargo}, M_{payload_{max}}$ [kg]	0, 0	$TSC_{loiter}$ [1/hr]	0.8

Table 4.8: Conventional and unconventional balance analysis comparison for the ATR72<sup>®</sup> aircraft.

ATR72 <sup>®</sup> Balance						
Conventional Module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(12.3, 0.0, 0.2)	1.61E+05	8.68E+05	9.94E+05	0	2.59E+04
ZFM	(12.5, 0.0, 0.1)	6.70E+04	8.58E+05	8.94E+05	0	2.95E+04
ZPM	(12.3, 0.0, 0.4)	2.48E+05	6.37E+05	8.54E+05	0	2.10E+04
OEM	(12.9, 0.0, 0.2)	6.05E+04	6.15E+05	6.51E+05	0	2.76E+04

Unconventional module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(12.3, 0.0, 0.2)	1.61E+05	8.65E+05	9.90E+05	0	2.58E+04
ZFM	(12.5, 0.0, 0.1)	6.66E+04	8.55E+05	8.90E+05	0	2.94E+04
ZPM	(12.3, 0.0, 0.4)	2.48E+05	6.33E+05	8.50E+05	0	2.08E+04
OEM	(12.9, 0.0, 0.2)	6.02E+04	6.12E+05	6.47E+05	0	2.75E+04

Table 4.9: Results comparison for the ATR72<sup>®</sup> weight analysis with the conventional and unconventional WB software.

<b>ATR72<sup>®</sup></b>			
<b>Geometry</b>			
Fuselage Length [m]	27	Fuselage Width (mean) [m]	1.83
Wing Span [m]	27	Wing Area [m <sup>2</sup> ]	58.8
<b>Weight modules results</b>			
	Real value	Estimated	
		Conventional (Error [%])	Unconventional (Error [%])
MTOM [kg]	23000	22893 (-0.47)	23124 (0.54)
OEM [kg]	13500	12858 (-3.40)	13218 (-0.70)
ZFM [kg]	20800	20724 (-0.37)	20568 (-1.12)
Max. Passenger nb. [-]	70	72 (2.86)	70 (0.00)
Max. Payload tot [kg]	7500	7560 (0.80)	7350 (-2.00)
Max. Fuel mass[kg]	5000	5112 (2.24)	5112 (2.24)
Cabin Crew nb.	2	2	2
Lavatory nb.	1	1	1
Wing Loading [kg/m <sup>2</sup> ]	-	330	333
System mass [kg]	-	-	3532
Engines Mass [kg]	-	-	1202
Structure mass [kg]	-	-	7837
Abreast Nb	4	3	-
Aisle Nb	1	1	-
Rows Nb	17	24	-

Table 4.10: Results comparison for the ATR72<sup>®</sup> range analysis with the conventional and unconventional WB software.

<b>ATR72<sup>®</sup></b>					
<b>Fuel consumption [kg]</b>					
Flight Phase	Take-Off	Climb	Cruise	Loiter	Land
Conventional module	343	564	787	529	103
Unconventional module	347	569	849	533	104
<b>Ranges [km]</b>					
Configuration	Max Payload	Max Fuel	ZPM	Max Payload Real	Error [%]
Conventional module	1451	3202	4230	1500	-3.27
Unconventional module	1485	3166	4133		-1.00

### 4.3.2 Boeing B777®

The Boeing B777® (fig.:4.3a) is a twin high bypass ratio turbofan (it uses the GE90-110B / 115B), there are different versions of this aircraft but the one tested is the 200LR, where LR stands for long range. The B777-200LR® is sold since 2006 and is currently one of the commercial aircraft with the highest range and payload capability. It is characterised by a fuselage length of 63.7m, a wingspan of 64.8m, a fuselage width of maximum 6.2m and a wing area of 436.8m<sup>2</sup>. The aircraft is designed to carry 440 passengers in one class configuration for almost 14000km and, at max payload, it can board 95% of the maximum fuel allowed in the tanks (*FPM* is set to 95).

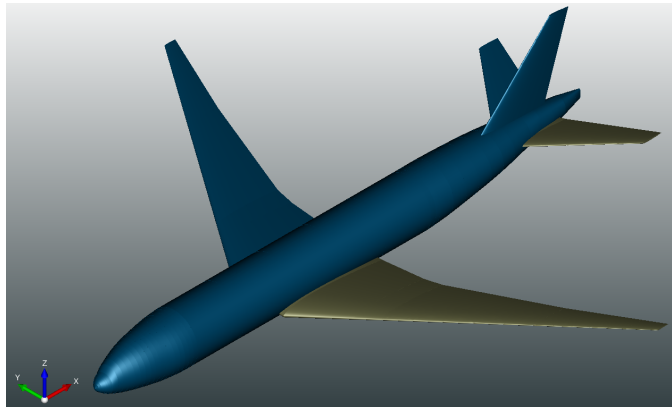
For the B777-200LR® the code uses the input parameters defined in tables 4.11, and for the other quantities not mentioned it uses the default values. In this case, the aircraft is designed with an APU engine, and multiple hydraulic systems and the relative options (*auxiliaryPowerUnit* and *singleHydraulics*) are set True and False respectively. The cruise speed is set to be 272m/s (Mach=0.8) and the LD coefficient is set to 17. Differently from the ATR72®, since the engine coefficient is much higher, the TSFC is set 0.5hr<sup>-1</sup> and 0.4hr<sup>-1</sup> respectively for the cruise and loiter phases. Seventeen tons of cargo is added to be able to correctly estimate the maximum payload mass that the aircraft can transport considering its not negligible cargo bay size. The code is not able to define precisely the cargo bay dimension; for this reason, the user must provide, when known, the cargo mass directly. In the end, the UWB uses a passenger surface density of 1.66.

Tables 4.12 and 4.13 summarize all the results obtained with the conventional and unconventional WB software alongside the real value. First of all, one can see that the geometry analysis output matches the real aircraft dimensions. The MTOM, ZFM and the OEM match the real value with an error lower than 2%. The payload mass is related to the defined cargo mass chosen by the user; to avoid an over-fitting of the aircraft parameters  $M_{\text{cargo}}$  is selected as a round number considering the value defined in the aircraft manual [32]. The CWB estimates 441 passengers in 9 abreasts and 49 rows with two aisles (3 groups of 3 abreasts separated by two corridors), while the UWB estimates exactly 440 passengers due to the  $\rho_{\text{pax}}$  defined. Both the codes agree that the number of lavatories should be 8 and the aircraft needs 11 crew members (2 pilots + 9 cabin crew members). In the end, the UWB estimates that the two engines should weight approximately 17 tons and, similarly to the previous case, the structure mass is about 2.5 times the system mass and the 25% of the MTOM.

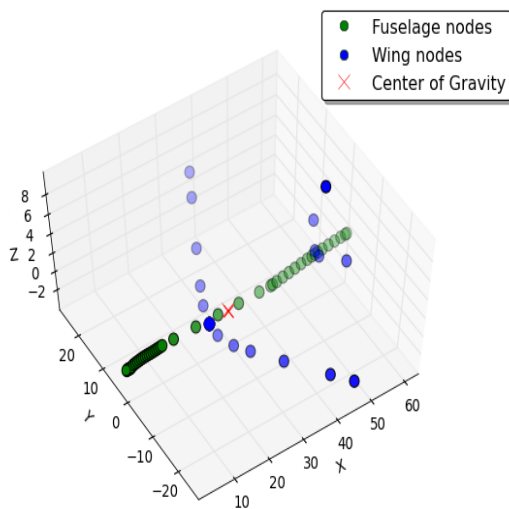
With the range analysis carried out using both the conventional and unconventional codes, it was possible to evaluate the range versus payload and fuel consumption. Figure 4.3c shows a range of 14000 km at max payload and 19000 km without payload, this last value is not far from the actual aircraft record of 21000km. Since the aircraft can carry almost the maximum amount of fuel with maximum payload, the stiffness of the range vs payload curve is higher than the one for the ATR72®. For the same reason, the range at max payload is close to the range at max fuel; in the second case, in fact, to add the remaining 5% ( $\approx 7$  tons) of fuel inside the tanks, the payload is reduced by only 7 tons. As expected for a long-range aircraft, the cruise phase has a fuel consumption that is two orders of magnitude higher than the other phases, but both codes agree on similar values. It is interesting to see that the loiter phase, differently for the ATR72® requires one-fourth of the climb phase fuel consumption. The reason behind this is related to the low TSFC

during loiter for a turbofan engine and to the aircraft weight loss between the beginning of the climb and the end of the cruise (almost 125 tons).

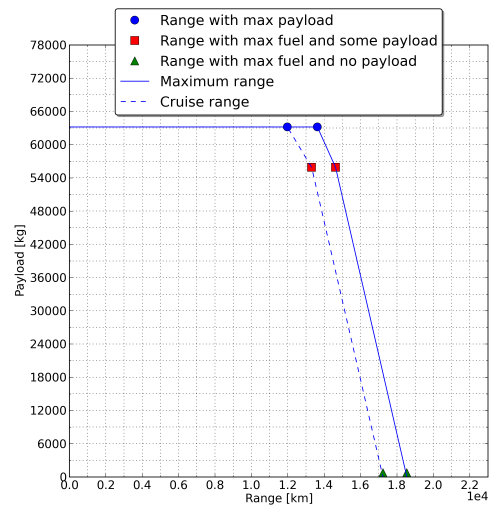
Looking at figure 4.3b and table 4.14 the centre of gravity is around the centre chord of the main wing. The CoG is placed precisely on the y-axis as expected and has a little offset on the z-axis mainly because the wing, and so the fuel, are placed at the bottom of the fuselage. In this case, the two codes evaluate similar COG estimation only with max fuel and OEM configurations. The UWB places fuel in all the horizontal wings while the CWB only in the main horizontal wings, for this reason with the UWB the COG is moved backwards with fuel on board. The cabin is placed slightly differently since the UWB takes into account the fuselage thickness and cabin height defined by the user. The COG can vary of two meters maximum between the aircraft configurations with full or empty fuel tanks. Even though both codes provide a reasonable result, in this case, the UWB mass placement, as well as the COG position, seems more accurate. Also, the MOI have reasonable results with the expected maximum value for the Yaw moment and the already discussed trend on the roll, pitch and Ixz moments. The symmetry of the aircraft is correctly taken into account since the Ixy and Iyz moments are zero. Both codes calculate the same order of magnitude for the MOI at OEM and zero fuel configurations. However, if fuel is taken into account, the results obtained are entirely different. The UWB places fuel also in the horizontal tail stabiliser and this lead to a remarkable increase of the MOI, confirming once more the importance of the fuel disposition for the aircraft balance.



(a) Full view of the B777<sup>®</sup> aircraft geometry.



(b) B777<sup>®</sup> aircraft centre of gravity.



(c) B777<sup>®</sup> aircraft range versus payload.

Figure 4.3: B777<sup>®</sup> aircraft analysis, full aircraft (4.3a), centre of gravity (4.3b) and range versus payload (4.3c).

Table 4.11: CWB and UWB modules parameters defined for the B777<sup>®</sup> analysis.

<b>B777<sup>®</sup> Parameters</b>			
<b>Unconventional WB</b>			
<b>Weight</b>		<b>Engine analysis (User Engine = False)</b>	
<b>Name</b>	<b>Value</b>	<b>Name</b>	<b>Value</b>
$t_{mvert}$	0.00014263	TP	False
$\rho_{mvert}$ [kg/m <sup>3</sup> ]	2700	$Nb_e$	2
$h_{cabin}$ [m]	2.3	$M_e$ [kg]	-
SH	False	$T_{max}$ [kN]	-
$Nb_{floors}$	1	$TSFC_{cruise}$ [1/hr]	0.5
FPM	95	APU	True
$M_{payload_{max}}$ [kg]	0	WM	True
LD	17	<b>Balance (User Engine Placement = False)</b>	
$V_{cru}$ [m/s]	272	Name	Value
$\rho_{pax}$ [pax/m <sup>2</sup> ]	1.66	1st Engine Placement	-
$Nb_{pilots}$	2	2nd Engine Placement	-
$M_{cargo}$ [kg]	17000	<b>Balance (User Case = False)</b>	
$Vol_{f_{max}}$ [m <sup>3</sup> ]	0	Name	Value
$Nb_{pax_{max}}$	0	F_PERC, P_PERC	0,0

<b>Conventional WB</b>		<b>Range</b>	
<b>Name</b>	<b>Value</b>	<b>Name</b>	<b>Value</b>
TP	False	TP	False
<i>isDoubleFloor</i>	0	WINGLET	False
$V_{cru}$ [m/s]	272	$V_{cru}$ [m/s]	272
$M_{cargo}$ [kg]	17000	LD	17
$Nb_{pilots}$	2	$h_{loiter}$ [min]	30
$Vol_{f_{max}}$ [m <sup>3</sup> ]	0	$TSFC_{cruise}$ [1/hr]	0.5
$Nb_{pax_{max}}, M_{payload_{max}}$ [kg]	0, 0	$TSFC_{loiter}$ [1/hr]	0.4



Table 4.12: Results comparison for the B777<sup>®</sup> weight analysis with the conventional and unconventional WB software.

<b>B777<sup>®</sup></b>			
<b>Geometry</b>			
Fuselage Length [m]	63.7	Fuselage Width (max) [m]	6.2
Wing Span [m]	64.8	Wing Area [m <sup>2</sup> ]	436.8
<b>Weight modules results</b>			
	<b>Real value</b>	<b>Estimated</b>	
		<b>Conventional (Error [%])</b>	<b>Unconventional (Error [%])</b>
MTOM [kg]	347814	343789 (-1.16)	347165 (-0.19)
OEM [kg]	145149	145018 (-0.09)	146306 (-0.80)
ZFM [kg]	209106	211701 (1.24)	209506 (-0.19)
Max. Passenger nb. [-]	440	441 (0.23)	440 (0.00)
Max. Payload tot [kg]	63956	63305 (-1.02)	63200 (-1.18)
Max. Fuel mass [kg]	145541	145816 (0.19)	144904 (-0.44)
Cabin Crew nb.	9	9	9
Lavatory nb.	8	8	8
Wing Loading [kg]	-	638	644
System mass [kg]	-	-	34572
Engines Mass [kg]	-	-	16568
Structure mass [kg]	-	-	85656
Aisle Nb	2	2	-
Abreast Nb	9	9	-
Rows Nb	49	49	-

Table 4.13: Results comparison for the B777<sup>®</sup> range analysis with the conventional and unconventional WB software.

<b>B777<sup>®</sup></b>					
<b>Fuel consumption [kg]</b>					
<b>Flight Phase</b>	<b>Take-Off</b>	<b>Climb</b>	<b>Cruise</b>	<b>Loiter</b>	<b>Land</b>
<b>Conventional module</b>	5157	8466	115187	2514	1062
<b>Unconventional module</b>	5207	8549	111959	2590	1094
<b>Ranges [km]</b>					
<b>Configuration</b>	<b>Max Payload</b>	<b>Max Fuel</b>	<b>No Payload</b>	<b>Max Payload Real</b>	<b>Error [%]</b>
<b>Conventional module</b>	14020	14734	18959	13890	0.94
<b>Unconventional module</b>	13447	14430	18273		-3.19

Table 4.14: Conventional and unconventional balance analysis comparison for the B777<sup>®</sup> aircraft.

B777 <sup>®</sup> Balance						
Conventional Module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(27.3, 0.0, -0.5)	3.90E+06	4.47E+07	4.71E+07	0	1.21E+06
ZFM	(29.1, 0.0, 0.1)	3.18E+06	4.15E+07	4.37E+07	0	7.94E+05
ZPM	(27.3, 0.0, -0.6)	3.61E+06	3.51E+07	3.75E+07	0	1.18E+06
OEM	(29.9, 0.0, 0.2)	2.90E+06	3.15E+07	3.37E+07	0	7.26E+05
Unconventional module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(29.2, 0.0, -0.3)	1.34E+07	5.62E+07	6.81E+07	0	1.75E+06
ZFM	(28.6, 0.0, 0.1)	2.06E+06	4.16E+07	4.26E+07	0	8.09E+05
ZPM	(29.7, 0.0, -0.3)	1.37E+07	4.69E+07	5.93E+07	0	1.81E+06
OEM	(29.2, 0.0, 0.1)	1.77E+06	3.18E+07	3.27E+07	0	7.51E+05

### 4.3.3 WP46RM

The WP46RM is a twin rear mounted (RM) turboprop aircraft; a user case designed in the AGILE project (fig.:4.4a). It is characterised by main wings mounted at the bottom of the fuselage, and the turboprop engines are placed at the tip of the horizontal tail stabiliser. The TLAR (Top Level Aircraft Requirements) state that the aircraft should be able to carry 90 passengers flying at Mach 0.56 at an altitude of approximately 7620 m. The fuel tanks should be able to carry 5 tons of fuel at maximum payload and the MTOM, and the OEM targets are respectively 37 and 23 tons.

The input parameters used for the WP46RM can be found in table 4.15. The  $LD = 15$ , the  $V_{cru} = 190\text{m/s}$  and the engine TSFC during loiter and cruise are lowered by 0.1 from the suggested turboprop default values assuming a more efficient engine. The seats length and width are set to respectively 0.6 and 0.8 meters to estimate the number of passengers correctly. The APU and SH options are set both to False since the aircraft will not have an APU engine but will include multiple hydraulic systems. Since the engines are placed on the horizontal stabilisers the *wingMounted* (WM) option is set to False; furthermore, the engine position is known, and their COG coordinates are defined inside the CPACS file. and therefore the *userEnginePlacement* option is set True.

Since the WP46RM is in a design phase, no real values for the weights are available, but the results obtained with the conventional and unconventional WB are compared. Table 4.16 shows a close agreement on the aircraft masses obtained with both codes with the conventional one exceeding the expected MTOM, OEM and ZFM by 2 tons. The CWB estimates a maximum of 2 passengers less than the UWB for a total of 9660 tons of payload; the code shows a seat disposition of 23 rows with four abreasts divided by one aisle. Since the UWB place fuel inside the tail wings, it estimates 4 tons of extra fuel. The horizontal tail stabiliser can potentially be used to store fuel, but due to the presence of the engine probably the code overestimates the tanks volume. This last result is interesting since it shows the relevance of a possible fuel tank in the tail wings. It is important to mention that only the maximum amount of fuel allowed at maximum payload is used to evaluate the MTOM and the difference between the MTOM estimated with the

two codes is due to the OEM discrepancy. To be able to fill the tanks the payload needs to be reduced completely. The UWB and CWB agree approximately 6.7 tons of fuel that can be stored at maximum payload. The structure, engine and systems masses evaluated with the UWB follows the same trend of the previous aircraft tested. Both the UWB and the CWB modules agree on the fuel consumption at maximum payload and table 4.17 shows a similar trend to the ATR72<sup>®</sup> one. Both the codes agree on the maximum range at maximum payload but, due to the difference in the maximum fuel amount, the range at ZPM obtained with the UWB is 2000km higher than the CWB one. Figure 4.4b, and 4.4c show the two range vs payload curves obtained respectively with the conventional and unconventional WB software.

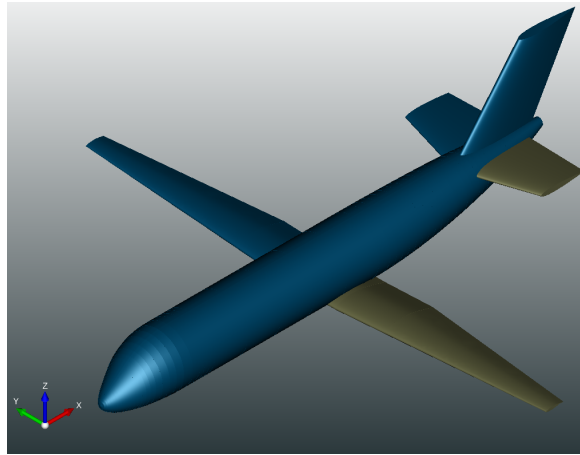
Figure 4.4d, and 4.4e show the differences between the COG position estimated with the conventional and unconventional WB software. It is interesting to see that the COG evaluated with the UWB software has a more backward position; this is due to the engines (black dots in figure 4.4e) and fuel placed in the horizontal tail stabiliser. Also, the MOI have reasonable results with the expected maximum value for the Yaw moment and the projected trend on the roll, pitch and Ixz. The symmetry of the aircraft is correctly taken into account since the Ixy and Iyz moments are zero. The engines and fuel placement influence not only the COG but, as expected, also the MOI and in particular the roll and Ixz moments.

Table 4.15: CWB and UWB modules parameters defined for the WP46RM analysis.

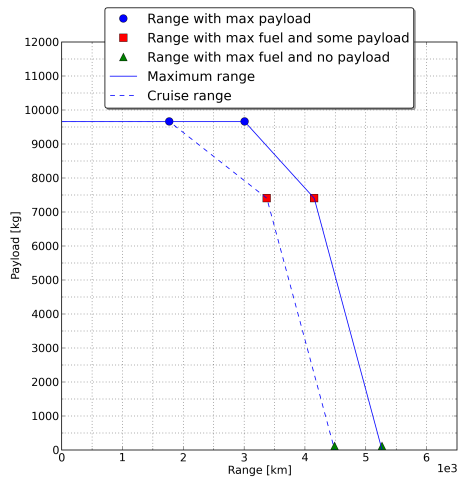
<b>WP46RM Unconventional WB Parameters</b>			
<b>Weight</b>		<b>Engine analysis (User Engine = False)</b>	
<b>Name</b>	<b>Value</b>	<b>Name</b>	<b>Value</b>
$t_{mvrt}$	0.00014263	TP	True
$\rho_{mvrt}$	2700	$Nb_e$	2
$h_{cabin}$ [m]	2.3	$M_e$ [kg]	-
SH	False	$T_{max}$ [kN]	-
$Nb_{floors}$	1	$TSC_{cruise}$ [1/hr]	0.6
$FPM_T$	0.5	APU	False
$M_{payload_{max}}$ [kg]	0	WM	False
$LD$	15	<b>Balance (User Engine Placement = True)</b>	
$V_{cru}$ [m/s]	190	<b>Name</b>	<b>Value</b>
$\rho_{pax}$ [pax/m <sup>2</sup> ]	1.39	1st Engine Placement (x, y, z) [m, m, m]	(26.0, 4.5, 2.0)
$Nb_{pilots}$	2	2nd Engine Placement (x, y, z) [m, m, m]	(26.0, -4.5, 2.0)
$M_{cargo}$ [kg]	0	<b>Balance (User Case = False)</b>	
$Vol_{f_{max}}$ [m <sup>3</sup> ]	0	<b>Name</b>	<b>Value</b>
$Nb_{pax_{max}}$	0	F_PERC, P_PERC	0,0

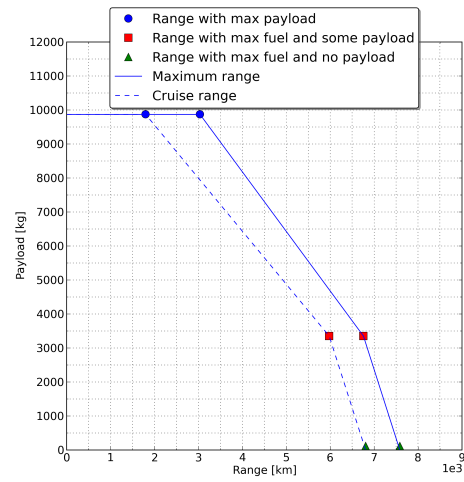
<b>Weight Conventional WB</b>		<b>Range</b>	
<b>Name</b>	<b>Value</b>	<b>Name</b>	<b>Value</b>
TP	False	TP	True
<i>isDoubleFloor</i>	0	WINGLET	False
$L_{st}$ [m]	0.8	$V_{cru}$ [m/s]	190
$WDT_{st}$ [m]	0.6	$LD$	15
$Nb_{pilots}$	2	$h_{loiter}$ [min]	30
$Vol_{f_{max}}$ [m <sup>3</sup> ]	0	$TSC_{cruise}$ [1/hr]	0.6
$M_{payload_{max}}, M_{cargo}$ [kg]	0, 0	$TSC_{loiter}$ [1/hr]	0.7



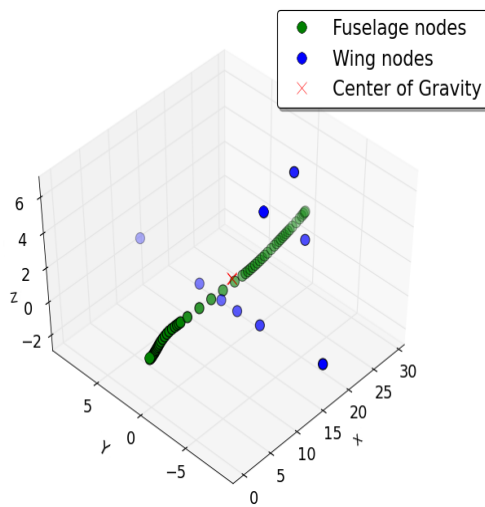
(a) Full view of the WP46RM aircraft geometry.



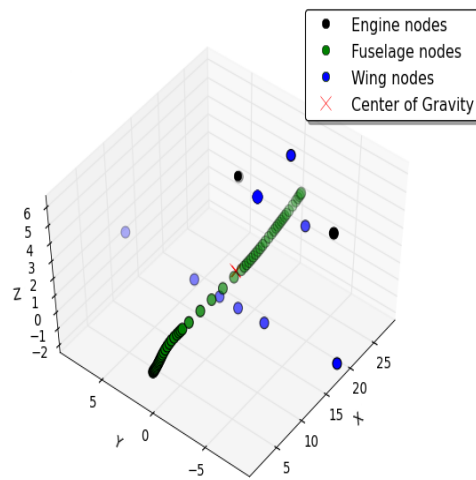
(b) Conventional module range versus payload.



(c) Unconventional module range versus payload



(d) Conventional module centre of gravity.



(e) Unconventional module centre of gravity.

Figure 4.4: WP46RM aircraft analysis, full aircraft (4.4a), range versus payload (4.4b, 4.4c) and centre of gravity (4.4d, 4.4e).

Table 4.16: WP46RM aircraft conventional and unconventional weight analysis comparison.

<b>WP46RM</b>			
<b>Geometry</b>			
Fuselage Length [m]	29.17	Fuselage Width (max) [m]	3.54
Wing Span [m]	30.6	Total Wing Area [m <sup>2</sup> ]	77.95
<b>Weight modules results</b>			
	<b>Conventional</b>	<b>Unconventional</b>	
MTOM [kg]	39636	37694	
OEM [kg]	23175	21303	
ZFM [kg]	33378	31173	
Max. Passenger nb. [-]	92	94	
Max. Payload tot [kg]	9660	9870	
Max. Fuel mass [kg]	9058	13041	
Max. Fuel at max payload [kg]	6801	6521	
Cabin Crew nb.	2	2	
Lavatory nb.	2	2	
Wing Loading [kg]	358	340	
System mass [kg]	-	6097	
Engines Mass [kg]	-	2600	
Structure mass [kg]	-	11484	
Aisle Nb	1	-	
Abreast Nb	4	-	
Rows Nb	23	-	

Table 4.17: WP46RM aircraft conventional and unconventional range analysis comparison.

<b>WP46RM</b>					
<b>Fuel consumption [kg]</b>					
<b>Flight Phase</b>	<b>Take-Off</b>	<b>Climb</b>	<b>Cruise</b>	<b>Loiter</b>	<b>Land</b>
<b>Conventional module</b>	595	976	3743	912	167
<b>Unconventional module</b>	565	928	3611	866	159
<b>Ranges [km]</b>					
<b>Configuration</b>	<b>Max Payload</b>	<b>Max Fuel</b>	<b>No Payload</b>		
<b>Conventional module</b>	3008	4152	5267		
<b>Unconventional module</b>	3035	6756	7584		

Table 4.18: Balance analysis comparison for the WP46RM aircraft with the conventional and unconventional WB software.

WP46RMBalance						
Conventional Module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(15.3, 0.0, 1.0)	2.30E+05	2.32E+06	2.46E+06	0	1.33E+05
ZFM	(13.2, 0.0, 0.9)	1.79E+05	1.47E+06	1.57E+06	0	6.44E+04
ZPM	(16.9, 0.0, 1.2)	2.31E+05	2.04E+06	2.20E+06	0	1.36E+05
OEM	(13.7, 0.0, 0.9)	1.65E+05	1.14E+06	1.24E+06	0	6.12E+04
Unconventional module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(16.1, 0.0, 1.0)	4.17E+05	2.23E+06	2.55E+06	0	3.60E+05
Max Fuel (MTOM)	(15.4, 0.0, 1.1)	1.85E+05	1.84E+06	1.95E+06	0	3.32E+05
ZPM	(18.0, 0.0, 1.0)	6.35E+05	2.15E+06	2.69E+06	0	3.88E+05
OEM	(17.0, 0.0, 1.2)	1.71E+05	1.58E+06	1.68E+06	0	3.33E+05
Engine	(26.0, 0.0, 2.0)	5.53E+04	2.56E+05	3.06E+05	0	2.54E+05

## 4.4 Semi-Conventional Box-wing AGILE analysis

The box-wing is a twin turbofan designed in the AGILE project as well as in the Parsifal project (fig.:4.5a). It is a semi-conventional aircraft (see ch.:1.1.1) characterised by a conventional fuselage and an unconventional boxed wing. One of the most significant sources of drag is the induced drag generated by the wing tip vortexes, the winglets that nowadays most of the aircraft use can reduce these vortexes and thus the induced drag. An annular wing that completes a circle around the fuselage could minimise even further the induced drag.

The box-wing aircraft analysis is carried out using both the conventional and unconventional WB software, and all the input parameters are set equal to the ones for the B777<sup>®</sup> analysis (tab.:4.11) except for the *FPM* that is left with its default value and the  $\rho_{\text{pax}} = 1.78\text{pax}/\text{m}^2$ . The box-wing aircraft is 37.57 meters long, has a maximum width of 3.95 meters, a wingspan of 26.65 meters and a wing planform area of 120 m<sup>2</sup>. The conventional WB software evaluates that the aircraft can board 180 passengers and should have six abreasts separated by one aisle and 30 rows. The unconventional code also estimates 180 passengers since the  $\rho_{\text{pax}}$  is set to 1.78; the passengers surface density is chosen to match the same passenger's number calculated with the CWB. The  $\rho_{\text{pax}}$  influences not only the total payload mass but also the OEM, as figure 4.6b shows.

Both the codes evaluate a maximum fuel mass around 14 tons, but the unconventional WB does not place fuel in the vertical portion of the box wing, and this leads to the 900 kg of difference that can be seen in table 4.19. The CWB calculates that the box-wing can carry almost the maximum amount of fuel with maximum payload, while the UWB considers the 80% of the total fuel due to the *FPM* left with its default value (80%). The two software estimate an OEM that differs from 10 tons and the same discrepancy can be found in the ZFM and MTOM. The UWB evaluates that the structure mass weights approximately 18 tons and the two turbofan engines should weight around 4 tons in total.

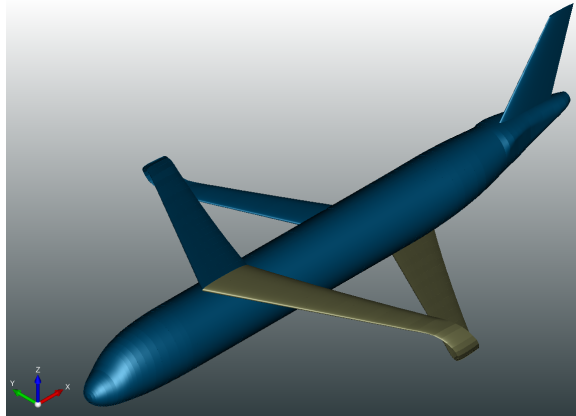
Figures 4.5b and 4.5c show that the two codes almost agree on the range at maximum

payload even if the discrepancy on the MTOM is not negligible; this is related to the difference of fuel at maximum payload (tab.: 4.17). The CWB calculates a higher MTOM, but since the fuel amount estimated is higher, the effect on the range evaluation is minimal. At max payload, both the analysis carried out with the conventional and unconventional modules agree on a 5000 km range. At zero payload calculated rangers differ by 1000 km since the aircraft evaluated with the UWB module is lighter but with the same amount of fuel as the one estimated with the CWB.

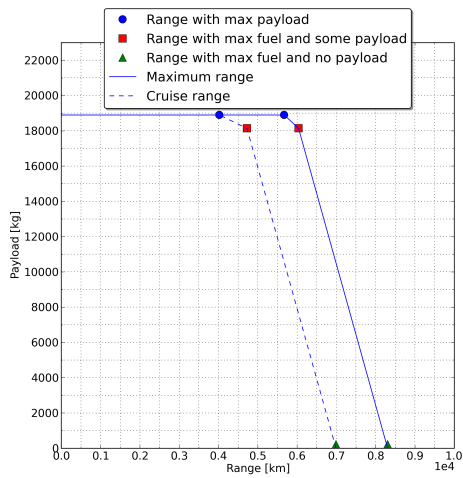
Figures 4.5d and 4.5e show the COG position at MTOM evaluated with the conventional and unconventional WB modules, while in table 4.21 all the values for the COG and MOI are listed. The two codes are in close agreement on the COG position and MOI, the differences that can be seen are related to the differences between the aircraft masses and the fuel placement. At OEM and ZFM the COG is almost at the same x-coordinate while in the other cases the one evaluated with the UWB is approximately 1 meter closer to the aircraft nose. The moments of inertia evaluated with the UWB are lower than the one assessed with the CWB due to the lower mass of the aircraft. It is interesting to notice that at MTOM and ZPM configurations the  $I_{xz}$  moments of inertia, due to the wing geometry, is negative.

As stated previously, the discrepancy on the two masses is related to the different evaluation methods for the OEM. The conventional WB evaluates the OEM using equation 3.11 while the unconventional WB calculates all the OEM components and sums them (see ch.:2.1). The unconventional module relies on the passengers surface density, the LD coefficient and the virtual thickness to be able to estimate the OEM correctly. Figure 4.6b shows that increasing the  $\rho_{pax}$  from  $1.6 \text{ pax/m}^2$  to  $1.95 \text{ pax/m}^2$  two more tons of OEM are added (value calculated for the Box-Wing Aircraft maintaining constant the cabin area). The increment is due to the higher number of crew members and furnishing. The LD coefficient has a lower impact on the OEM because it is related only with the engine mass (see ch.:3.4.2). An aircraft with a high LD requires less powerful and thus lighter engines; for this reason, increasing the LD coefficient will decrease the OEM. Figure 4.6c also shows the asymptotic trend of the OEM variation with the LD coefficient since even with a high LD the engine weight cannot be neglected. The virtual thickness is the most relevant parameter for the OEM evaluation since it influences the aeroplane structure mass directly. figure 4.6a shows that increasing the  $t_{mvrt}$  by 10% increases the Box-Wing OEM by approximately 10% and the trend line equation is linear as expected. If the user knows the actual structural mass of the aircraft he/she can modify the  $t_{mvrt}$  in the CPACS file to adjust the value estimated by the software; it is essential to state that the structure mass for the UWB is not related to the MTOM but the total wetted surface of the aircraft.

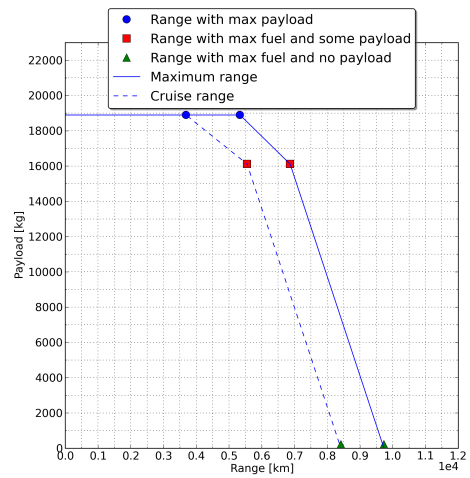
The box-wing analysis is the last one carried out with the conventional WB software, in the next section the Concorde<sup>®</sup> and Blended Wing analysis, carried out with UWB is presented.



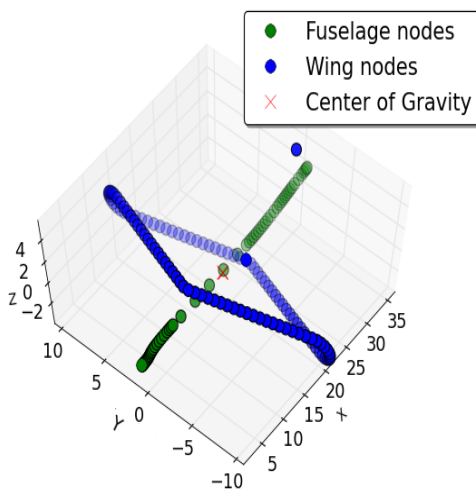
(a) Full view of the Box-Wing aircraft geometry.



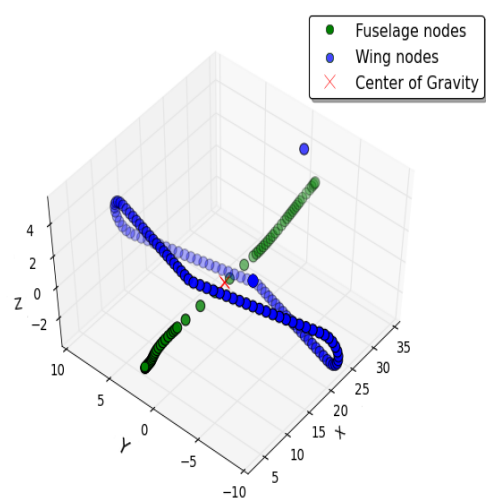
(b) Conventional module range versus payload.



(c) Unconventional module range versus payload.



(d) Conventional module centre of gravity.



(e) Unconventional module centre of gravity.

Figure 4.5: Box-Wing aircraft analysis, full aircraft (4.5a), range versus payload (4.5b, 4.5c) and centre of gravity (4.5d, 4.5e).



Table 4.19: Results comparison for the Box-Wing aircraft weight analysis with the conventional and unconventional WB software.

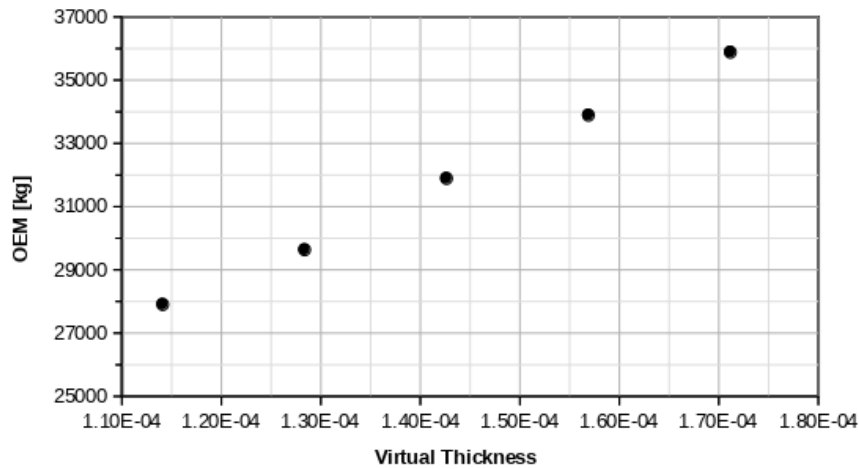
<b>Box-Wing</b>			
<b>Geometry</b>			
Fuselage Length [m]	37.57	Fuselage Width (max) [m]	3.95
Wing Span [m]	24.65	Total Wing Area [m <sup>2</sup> ]	120
<b>Weight modules results</b>			
	<b>Conventional</b>	<b>Unconventional</b>	
MTOM [kg]	74027	61916	
OEM [kg]	41062	31893	
ZFM [kg]	60582	50793	
Max. Passenger nb. [-]	180	180	
Max. Payload tot [kg]	18900	18900	
Max. Fuel mass [kg]	14824	13904	
Max. Fuel at max payload [kg]	14605	11123	
Cabin Crew nb.	4	4	
Lavatory nb.	4	4	
Wing Loading [kg]	617	516	
System mass [kg]	-	8765	
Engines Mass [kg]	-	3694	
Structure mass [kg]	-	18124	
Aisle Nb	1	-	
Abreast Nb	6	-	
Rows Nb	30	-	

Table 4.20: Results comparison for the Box-Wing aircraft range analysis with the conventional and unconventional WB software.

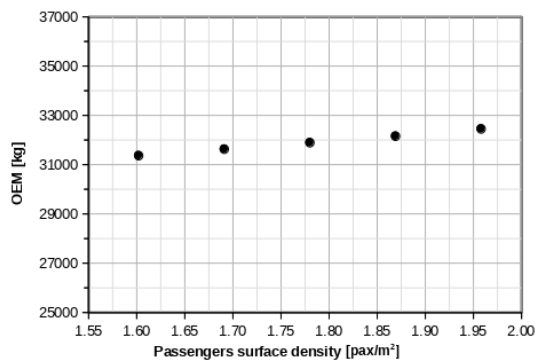
<b>Box-Wing</b>					
<b>Fuel consumption [kg]</b>					
<b>Flight Phase</b>	<b>Take-Off</b>	<b>Climb</b>	<b>Cruise</b>	<b>Loiter</b>	<b>Land</b>
<b>Conventional module</b>	1110	1823	9259	723	306
<b>Unconventional module</b>	929	1525	7132	612	259
<b>Ranges [km]</b>					
<b>Configuration</b>	<b>Max Payload</b>	<b>Max Fuel</b>	<b>No Payload</b>		
<b>Conventional module</b>	5673	6037	8304		
<b>Unconventional module</b>	5333	6864	9729		

Table 4.21: Balance analysis comparison for a Box-Wing aircraft with the conventional and unconventional WB software.

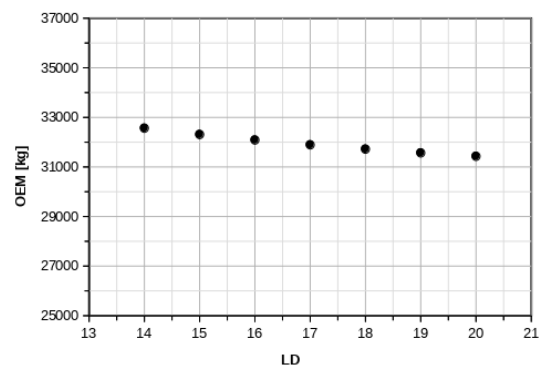
Box-Wing Balance						
Conventional Module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(18.3, 0.0, -0.4)	2.55E+05	5.18E+06	5.21E+06	0	-8.33E+04
ZFM	(16.8, 0.0, 0.0)	2.05E+05	4.44E+06	4.52E+06	0	9.85E+04
ZPM	(19.0, 0.0, -0.5)	2.17E+05	4.04E+06	4.06E+06	0	-8.07E+04
OEM	(17.1, 0.0, 0.1)	1.66E+05	3.37E+06	3.44E+06	0	9.17E+04
Unconventional module						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(17.2, 0.0, -0.1)	4.74E+05	4.06E+06	4.35E+06	0	-3.61E+04
ZFM	(17.1, 0.0, 0.0)	1.30E+05	3.69E+06	3.71E+06	0	6.72E+04
ZPM	(17.7, 0.0, -0.1)	5.22E+05	3.10E+06	3.46E+06	0	-6.60E+04
OEM	(17.4, 0.0, 0.0)	9.16E+04	2.65E+06	2.66E+06	0	6.17E+04



(a) OEM vs virtual thickness



(b) OEM vs passengers surface density



(c) OEM vs LD coefficient

Figure 4.6: Boxwing aircraft OEM variation against the virtual thickness (4.6a), the passengers density (4.6b) and the LD coefficient (4.6c).

## 4.5 Unconventional aircraft analysis

In this section the analysis of the Concorde<sup>®</sup> and BWB aircraft, carried out with the unconventional weight and balance (UWB) software, is presented. The goal is to demonstrate the capability of the UWB to estimate the aeroplane weight, balance and range even if its dimensions do not respect the conventional design criteria. The Concorde<sup>®</sup> results are compared with the real values that can be found in the relative certificate of 1979 [43]. For the BWB, no real data are available. The input parameters used for each aircraft are listed in the relative tables, but if a parameter is not mentioned it means that the default value is used (see 3.1.2 and B.2).

### 4.5.1 Blended Wing Body

The Blended Wing Body aircraft concept (figure 4.7a) consists of a hybrid shape similar to a flying wing that incorporates the fuselage and the wings frames to maximise the lift, while reducing at the same time the drag generated. The BWB design is focused on high range and high payload capabilities with reduced fuel consumption. The passengers will be placed on the central portion of the wing, loaded from the front of the aircraft, while the engine could be placed close to the aircraft tail, integrated or not with the wing profile. Placing the engines near the tail increases the overall aircraft lift minimising the turbulent flow around the wings; at the same time, the engine design must be carried out carefully since the flow at the inlet will be turbulent. The aircraft is in a conceptual design stage at the moment, NASA and Boeing are studying its capabilities.

For the test with the UWB the chosen parameters are listed in table 4.22. As one can see default values are used for most of the parameters, except for the passengers surface density that is increased to 1.69pax/m<sup>2</sup>, and the number of engines increased to 3. The engines mass and maximum thrust are unknown, but they are placed close to the wing tail following the proposal of the conceptual design.

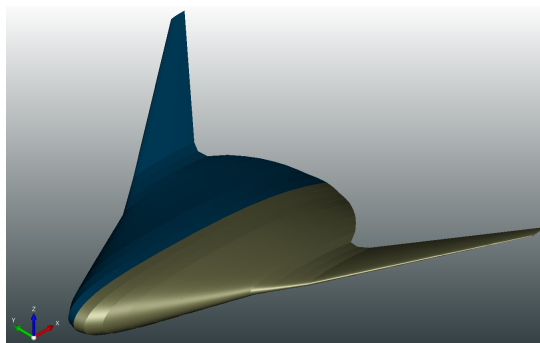
Table 4.23 shows that the aircraft could have an MTOM comparable with the B747<sup>®</sup> aircraft with almost the same amount of passengers. The B747<sup>®</sup> has an exit limit of 620 passengers while for the BWB the code evaluates 540 passengers to be conservative. For this test, only one floor is chosen, but the aircraft could be designed with two decks increasing the number of passengers to over 800. The estimated maximum fuel amount is approximately 220 tons against the 150 for the B747<sup>®</sup> allowing the BWB to fly at max payload 14000km (figure: 4.7d). The structural mass is estimated to be 57% of the OEM similarly to a conventional aircraft.

The last remarkable results obtained from the weight analysis are the aircraft dimensions that are entirely outside any conventional range. The wingspan is lower than the one of an A330<sup>®</sup> but the wing planform area is comparable to an A380<sup>®</sup> aircraft; the cabin area is higher than any other aircraft ever flown even if we do not consider a possible second floor.

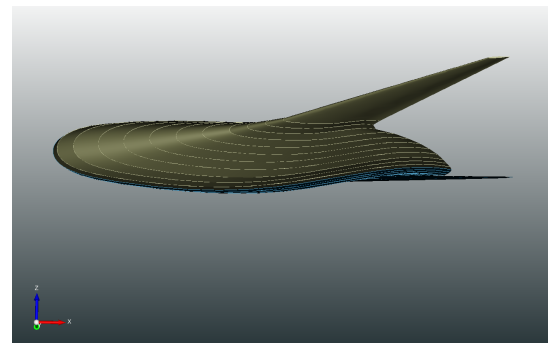
Figure 4.7c shows that the code evaluates the COG on the front part of the wing, at the centre of the cabin area. The reason for this is related to the aircraft belly dimension (figure 4.7b) that could accommodate passengers, payload, systems and even fuel. The COG can varies between a maximum of 2 meters, the furthest one from the nose corresponds to the zero payload configuration, while the closest to the nose corresponds to the zero fuel

condition. Figure 4.7b shows as black dots the engine node placement. Table 4.24 shows the aircraft MOI with different configurations and, as expected, it has a similar order of magnitude as for the B777<sup>®</sup>, since it has a comparable mass. The symmetry is respected, the  $I_{xy}$ ,  $I_{yz}$  moments are zero and the yaw moment is the most relevant one.

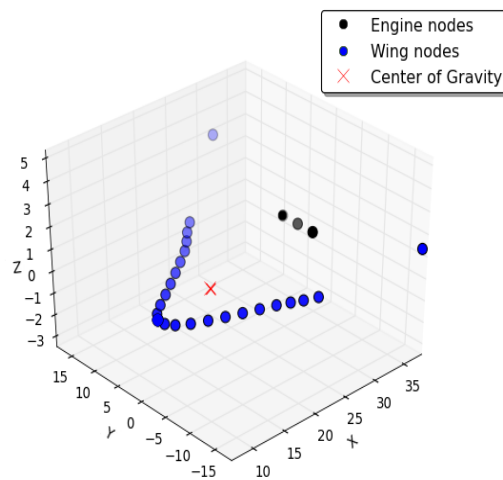
It is important to mention that the conceptual design of the BWB is also studied inside the AGILE project and a lot of energy and funding are focused its development.



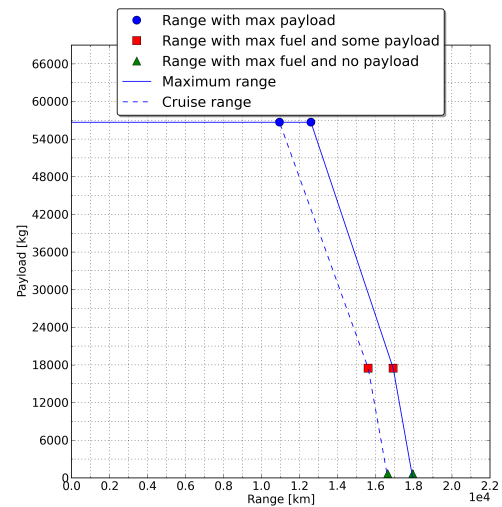
(a) Top view of the BWB geometry.



(b) Bottom view of the BWB geometry.



(c) Centre of gravity.



(d) Range versus payload.

Figure 4.7: Blended Wing Body aircraft analysis, full aircraft (4.7a, 4.7b), centre of gravity (4.7c) and range versus payload (4.7d).

Table 4.22: Parameters defined for the Blended Wing Body analysis with the unconventional code.

Blended Wing Body					
Weight		Engine analysis (User Engine = True)		Range	
Name	Value	Name	Value	Name	Value
$t_{mvert}$	0.00014263	TP	False	TP	False
$\rho_{mvert}$ [kg/m <sup>3</sup> ]	2700	$Nb_e$	3	WINGLET	False
$h_{cabin}$ [m]	2.3	$TSC_{cruise}$ [1/hr]	0.5	LD	17
SH	False	APU	True	$V_{cru}$ [m/s]	272
$Nb_{floors}$	1	WM	True	$h_{loiter}$ [min]	30
FPM	80			$TSC_{cruise}$ [1/hr]	0.5
$M_{payload_{max}}$ [kg], $Nb_{pax_{max}}$	0			$TSC_{loiter}$ [1/hr]	0.4
LD	17	Balance (User Engine Placement = True)			
$V_{cru}$ [m/s]	272	Name	Value [m]	Name	Value
$\rho_{pax}$ [pax/m <sup>2</sup> ]	1.69	1st Engine (x, y, z)	(32.0, 3, 1.0)	USER_CASE	False
$Nb_{pilots}$	2	2nd Engine (x, y, z)	(32.0, 0, 1.0)	F_PERC	0
$M_{cargo}$ [kg], $Vol_{f_{max}}$ [m <sup>3</sup> ]	0	3rd Engine (x, y, z)	(32.0, -3, 1.0)	P_PERC	0

Table 4.23: Results for the BWB weight and range analysis with the unconventional WB software.

BWB					
Weights				Geometry	
MTOM [kg]	415162	Max. Fuel Mass [kg]	215735	Wing Span [m]	56.4
OEM [kg]	185874	Max. Payload [kg]	56700	Aircraft Length [m]	39.6
ZFM [kg]	242574	Engine Mass [kg]	20103	Wing planform Area [m <sup>2</sup> ]	863.3
System mass [kg]	44420	Max. paxenger nb. [-]	540	Cabin Area [m <sup>2</sup> ]	319.2
Structure mass [kg]	107455	Cabin Crew nb.	11		
Wing Loading [kg/m <sup>2</sup> ]	481	Lavatory nb.	10		

Range and Fuel Consumption					
Flight Phase	Take-Off	Climb	Cruise	Loiter	Landing
Fuel consumed [kg]	6227	10223	141503	3008	1271

Configuration	Max Payload	Max Fuel	No Payload
Range [km]	14288	19426	20378

Table 4.24: Results for the BWB balance analysis with the unconventional WB software.

BWB Balance						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(17.7, 0.0, 0.0)	2.66E+06	5.26E+07	7.65E+07	0	4.88E+06
ZFM	(16.2, 0.0, -0.1)	7.95E+06	3.39E+07	3.99E+07	0	4.01E+06
ZPM	(18.8, 0.0, 0.1)	3.04E+07	4.88E+07	7.69E+07	0	5.11E+06
OEM	(17.5, 0.0, -0.1)	7.11E+06	2.70E+07	3.26E+07	0	4.13E+06
Engine	(32.0, 0.0, 1.0)	1.40E+05	4.12E+06	4.22E+06	0	4.10E+06

## 4.5.2 Concorde®

The Concorde® was the first and only supersonic commercial aircraft until today; thanks to its delta shape wings and its 4 Rolls-Royce turbojet engines with afterburners, it was able to reach Mach 2.04 at cruise altitude (18.3 km). It was the first class luxury jet that covered the route from Paris (Charles de Gaulle) or London (Heathrow) to New York (John F. Kennedy International) or Washington in less than four hours.

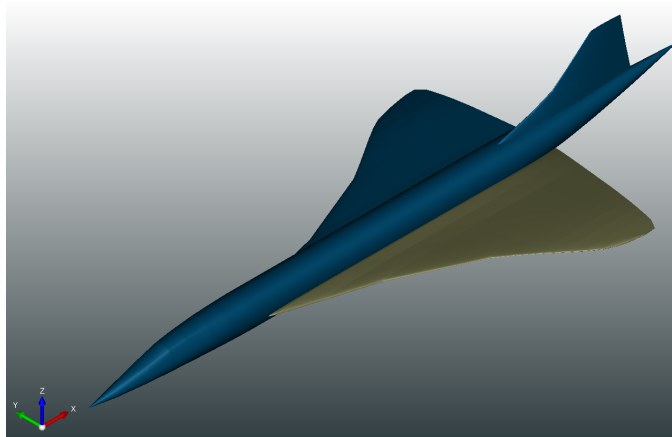
For the aeroplane analysis, the input parameters that have been changed can be found in table 4.25. First of all the virtual thickness is lowered to take into account the aircraft lighter structure. Since the Concorde® (figure 4.8a) had only first class seats, the  $\rho_{\text{pax}}$  is set to 1.29pax/m<sup>2</sup>. To be able to evaluate the aircraft crew members correctly, the  $N_{\text{pilots}}$  is increased to 3. Similarly to the B777®, the  $FPM = 95\%$  since the aircraft is realised for long-range flights at max payload. The code is not able to evaluate the supersonic flight phase, but the cruise speed is set to 600m/s and the LD coefficient to 5.5 (an average value between the cruise and minimum LD value). To test the software *userEngine* and *userEnginePlacement* options, the engines were directly defined inside the CPACS file with their respective weight, maximum thrust and placement.

The Concorde® was 61.3 meters long with a fuselage width of maximum 2.1 meters, a wingspan of 25.2 meters and a wing planform area of 414m<sup>2</sup>. Table 4.26 shows that the code evaluates all the masses with an error lower than 2% except for the payload mass since it does not take into account 1 ton of extra baggage. If the user knows the cargo mass, it is possible to use the *massCargo* option to manually add the value to the total payload mass. For this analysis, this option was left zero as default to demonstrate the UWB software capability. Even if the code underestimates the payload mass, the MTOM has an error lower than 1% since probably it slightly overestimates the maximum amount of fuel that the aircraft can carry at max payload. As it is possible to notice, even when the virtual thickness has been lowered due to the large aircraft wetted area, the structure mass is around 52% of the OEM.

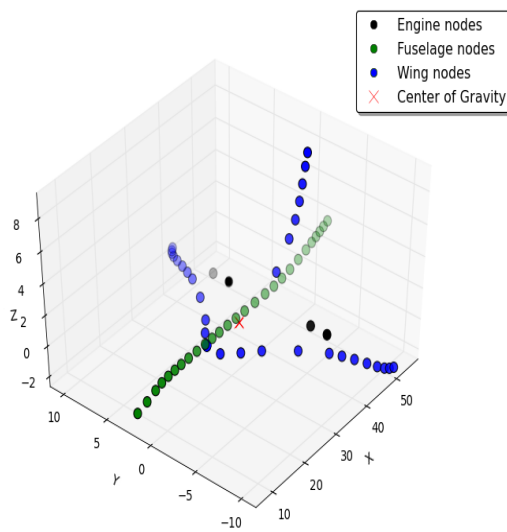
The range analysis was carried out to obtain both the fuel consumption and the range versus payload plot (figure 4.8c and table 4.27). Using the defined input values and the evaluated masses, the range at maximum payload exceeds the real value only about 2%. To be able to achieve a better range and fuel consumption evaluation, the UWB software needs to be coupled with the mission analysis module currently under development.

Table 4.28 lists all the COG and MOI values obtained with the balance analysis. At MTOM the COG is placed around the quarter chord of the delta wing (fig.:4.8b) while at OEM, due to the engine placement, it is shifted about 5 meters away from the nose. The value is reasonable since the Concorde® was trimmed by moving the fuel inside the wings. During the flight, the COG could be moved between the 50% and 60% of the aircraft length (away from nose). In the end, the MOI follows a similar trend as of all the already discussed aircraft in this chapter. The symmetry is respected, and the maximum value is relative to the yaw moment; without fuel, the roll moment is one order of magnitude lower than the one evaluated with maximum fuel.

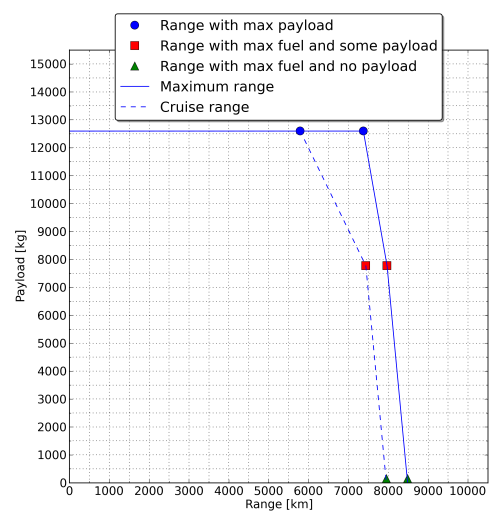
The Concorde® is no longer in service due to its high maintenance costs, the large noise emissions and also due to safety reasons; but in the foreseeable future supersonic commercial aircraft will again sail the sky.



(a) Full view of the Concorde<sup>®</sup> geometry.



(b) Centre of gravity.



(c) Range versus payload.

Figure 4.8: Concorde<sup>®</sup> aircraft analysis, full aircraft (4.8a), centre of gravity (4.8b) and range versus payload (4.8c).

Table 4.25: Parameters defined for the Concorde<sup>®</sup> analysis with the unconventional WB software.

Concorde <sup>®</sup>					
Weights		Engine analysis (User Engine = True)		Range	
Name	Value	Name	Value	Name	Value
$t_{mvert}$	0.000099	TP	False	TP	False
$\rho_{mvert}$ [kg/m <sup>3</sup> ]	2700	$Nb_e$	4	WINGLET	False
$h_{cabin}$ [m]	1.5	$M_e$ [kg]	3175	$LD$	5.5
SH	False	$T_{max}$ [kN]	169	$V_{cru}$ [m/s]	600
$Nb_{floors}$	1	$TSC_{cruise}$ [1/hr]	0.8	$h_{loiter}$ [min]	30
$FPM$	95	APU	True	$TSC_{cruise}$ [1/hr]	0.8
$Nb_{pilots}$	3	WM	True	$TSC_{loiter}$ [1/hr]	0.9
$LD$	5.5	<b>Balance (User Engine Placement = True)</b>			
$V_{cru}$ [m/s]	600	Name	Value [m]	Name	Value
$\rho_{pax}$ [pax/m <sup>2</sup> ]	1.29	1st Engine (x, y, z)	(42.0, 4.6, 0.0)	USER_CASE	False
$Vol_{f_{max}}$ [m <sup>3</sup> ]	0	2nd Engine (x, y, z)	(42.0, -4.6, 0.0)	F.PERC	0
$M_{cargo}, M_{payload_{max}}$ [kg]	0	3rd Engine (x, y, z)	(42.0, 6.4, 0.0)	P.PERC	0
$Nb_{pax_{max}}$ [-]	0	4th Engine (x, y, z)	(42.0, -6.4, 0.0)		

Table 4.26: Results for the Concorde<sup>®</sup> weight analysis with the unconventional WB software.

Concorde <sup>®</sup>			
Geometry			
Fuselage Length [m]	61.3	Fuselage Width (max) [m]	2.1
Wing Span [m]	25.2	Wing Area [m <sup>2</sup> ]	414
Weight module results			
	Real value	Estimated	Error [%]
MTOM [kg]	185000	184145	-0.46
OEM [kg]	78700	79912	1.54
ZFM [kg]	92080	92513	0.47
Max. Passenger nb. [-]	120	120	0.00
Max. Payload [kg]	13380	12600	-0.06
Max. Fuel mass [kg]	95680	96456	0.81
Max. Fuel mass with max payload [kg]	92920	91633	1.39
Cabin Crew nb.	3	3	-
Lavatory nb.	3	2	-
Wing Loading [kg]	-	445	-
System mass [kg]	-	19209	-
Structure mass [kg]	-	41706	-



Table 4.27: Results for the Concorde<sup>®</sup> range analysis with the unconventional WB software.

Range modules results					
Flight Phase	Take-Off	Climb	Cruise	Loiter	Land
Fuel mass consumed [kg]	2762	4535	69069	9276	493
Configuration	Max Payload	Max Fuel	No Payload	Max Payload Real	Error [%]
Range [km]	7373	7968	8482	7222	2.09

Table 4.28: Results for the Concorde<sup>®</sup> balance analysis with the unconventional WB software.

Concorde <sup>®</sup> Balance						
Configuration	COG (x,y,z) [m]	MOI [kgm <sup>2</sup> ]				
		Ixx	Iyy	Izz	Ixy = Iyz	Ixz
Max Payload (MTOM)	(32.5, 0.0, 0.4)	2.15E+06	2.53E+07	2.71E+07	0	1.30E+06
ZFM	(36.2, 0.0, 0.9)	7.92E+05	1.92E+07	1.98E+07	0	1.40E+06
ZPM	(32.6, 0.0, 0.4)	2.20E+06	2.42E+07	2.61E+07	0	1.31E+06
OEM	(37.4, 0.0, 0.9)	7.75E+05	1.81E+07	1.87E+07	0	1.40E+06
Engine	(42.0, 0.0, 0.0)	3.97E+05	1.16E+06	1.55E+06	0	1.16E+06

# Chapter 5

## Conclusions

The present thesis focused on the development of Weight and Balance evaluation software for conventional and unconventional aircraft design that will be integrated into the newly developed CASIOMpy software. Two modules were developed, one that can analyse conventional airliner in use today, and the other that also can test unconventional aircraft. It was shown that the developed software estimates with sufficient accuracy both conventional and unconventional aircraft masses, range, the centre of gravity and moments of inertia. The conventional module can perform the analysis with only the main aircraft dimensions, which is ideal for the early conceptual design phase when the entire aircraft geometry is not defined in details. The unconventional module can successfully estimate the weight of any kind of aircraft; it has high flexibility and can be used inside an optimisation process. This module can be calibrated precisely if some of the aircraft masses, such as the structure mass, is known a priori.

The software developed gives the best performance when the user provides the CPACS geometry file as input.

Although the modules are not meant to be used for mission analysis, they do provide a range estimation that is an additional check in the design process; furthermore, the unconventional module estimates the engines masses and the available maximum thrust.

It was shown that the software carries out the balance analysis successfully; it evaluates the centre of gravity generally between the wing quarter chord and the aircraft centre. Moreover, the centre of gravity varies with aeroplane geometry and mass placements. When using the software for unconventional aircraft, the user can directly specify the engine positioning to increase the accuracy of the results.

Both the conventional and unconventional modules provide similar results when a conventional aircraft is analysed.

### 5.1 Perspectives

Much work can still be done to improve the Weight and Balance software. For the conventional aircraft analysis, the aircraft database can be extended, and further details can be added to the database to better estimate the Weight and Balance of a more extensive range of aircraft.

The unconventional aircraft software module can be tested using new unconventional aircraft configurations, and this might lead to a reduction of the results error.

The overall code, once it will be included inside CEASIOMpy, could be speed up and all the aircraft systems and structures masses could be evaluated in more detail for design purposes. It might be useful to introduce a separate module for supersonic aircraft to differentiate subsonic and supersonic commercial aeroplanes.

Shortly, CFS Engineering and Airinnova will develop new modules of the CEASIOMpy software, allowing the complete aircraft design with new VP models and MDO algorithms. The new modules will focus on:

- the propulsion and mission analysis;
- the structural and aeroelasticity analysis,
- the stability analysis.

In the long term, the Weight and Balance software developed in this project might be included in an optimisation algorithm to completely automate the design process, focusing on the development of the novel aircraft configurations for the new age of commercial flights.



# Appendix A

## Linear regression method

The regression analysis aims to construct mathematical models that describe or explain relationships between variables [44]. Given a pair of observations  $(x_i, y_i)$  with  $i = 1, \dots, n$ ) the objective is to define a smooth curve that will intersect the majority of points. The linear regression model tries to fit the scattered distribution using a straight line. In general, it is possible to define the expected value (E) as a function of the response and explanatory variables (Y,  $X_k$ ):

$$E[Y|X_1 = x_1, \dots, X_k = x_k] = \Phi(x_1, \dots, x_k). \quad (\text{A.1})$$

Specifying the equation for the linear regression, we can expect to obtain:

$$\Phi(x_1, \dots, x_k) = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k, \quad (\text{A.2})$$

a linear equation related to the  $\beta$  coefficients and the number of explanatory variables (k). The best combination of those parameters provides the smallest approximation error defined as the divergence of the  $\Phi_k(x_k)$  point to the relative trend line one  $\Phi(x_k)$ :

$$\Delta_k = \Phi_k(x_k) - \Phi(x_k) \quad (\text{A.3})$$

The ideal trend line position is the one that minimises the root mean square error ( $\sigma$ ):

$$\sigma = \sqrt{\frac{\sum_1^n [\Delta_k]^2}{n-1}}; \quad (\text{A.4})$$

the minimum value of  $\sigma$  is obtained with the minimum of  $S = \sum_1^n [\Delta_k]^2$ . The  $\beta_k$  are, then, the solution of the system of  $\frac{\partial S}{\partial \beta_k} = 0$  equations that represent the  $S_{\min}$  condition.

In the Python environment exists the *sklearn linearRegression* model to compute the linear regression. It is a simple model that can predict data from a straight line evaluated starting from a training database. Figure A.1 shows an example of a straight line that fits an ensemble of points, the equation that characterises the line relates the expected and the explanatory variable with  $\beta_0 = 16.95$  and  $\beta_1 = 447.98$ .

It is also possible to subdivide the training data domain into multiple areas to maximise the fitting only in the region close to the test value. This strategy is successful especially when both the expected and explanatory variables are inside a wide range, and a single straight line can not fit the entire domain properly. This method minimises the error,

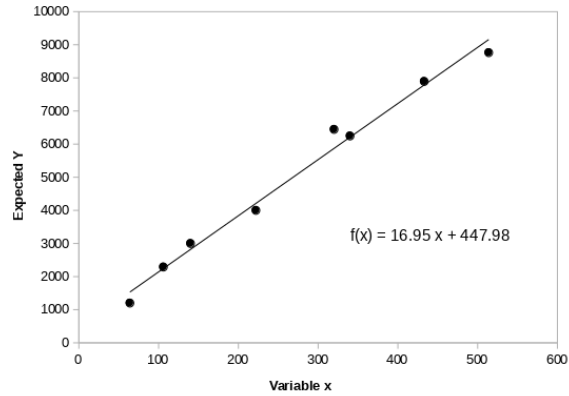


Figure A.1: Linear regression plot example.

but it might be less efficient for optimisation purposes. For this reason, the transition between the parts should have the minimum possible number (zero in the best case) of zero derivative points.

In the end, it is essential to mention that when using a linear regression method a strong correlation between the variables must exist, some examples: the aircraft maximum take-off mass and its geometrical values (wingspan, wing area, fuselage length or width) or the engine thrust and its mass. When the variables are not related there exists no straight line that can fit the value correctly.

# Appendix B

## Weight and Balance software

### B.1 Aircraft Database

Table B.1: Example of the aircraft database structure.

1	<b>Manufacturer</b>	<b>Airbus</b>	---
2	<b>Type</b>	<b>A330</b>	---
3	<b>Model</b>	<b>800neo</b>	---
4	Initial_service_date	2018	---
5	Engine_Nb	2	---
6	Seating_typical	257	---
7	Seating_1class_max	406	---
8	Abreast_Nb	8	---
9	<b>MTOM</b>	<b>251900</b>	---
10	OEM	132000	---
11	Fuel_max_litres	139090	---
12	Maximum_Payload_kg	42630	---
13	<b>Fus_Length_m</b>	<b>58.82</b>	---
14	<b>Fus_Width_m</b>	<b>5.64</b>	---
15	Cabin_Width_m	5.26	---
16	<b>Wing_Area_m2</b>	<b>395</b>	---
17	<b>Wing_Span_m</b>	<b>64</b>	---
18	Wing_AR	8.8	---
19	Max_Mach_Op_speed	0.86	---
20	Cruise_Mach_Speed	0.82	---
21	Altitude_Max_Operating_m	12634	---
22	Range_km	15094	---
23	Relative_PayloadN_PAX	257	---
24	Winglet	yes/no	---

The aircraft database is written inside a spreadsheet named *AircraftData2018\_v1* and must be saved with the *csv* format. Table B.1 shows an example of the database for the Airbus A330-800neo<sup>®</sup>, the bold values must be added since they are taken into account for the

linear regression process. The code reads line 9, 13, 14, 16, 17 and gathers, respectively, the MTOM, the fuselage length (*Fus\_Length\_m*) [m], the fuselage width (*Fus\_Width\_m*) [m], the wing area (*Wing\_Area\_m2*) [m<sup>2</sup>] and the wing span (*Wing\_Span\_m*) [m] In the first column are listed the number of the rows relative to a specific value, in the second column the name of the value, and in the third one the value of a specific aircraft. Each aircraft characteristics must be written in a single column, and additional quantities can be added after the last row. In the database there are more values than the ones required for the weight and balance analysis, it is strongly suggested when known to add all the possible information to be able to check the results afterwards.



## B.2 Weight and Balance modules input data

Table B.2: Input data for the balance analysis of conventional aircraft.

Name	Required	Default Value	UoM	Path or xPath	Description
<b>Balance Module</b>					
cpacsfile.xml	yes	-	-	TSP/balance/userBalance/userCase	Aircraft geometry in the CPACS.xml format
USER_CASE	no	False	-	TSP/balance/userBalance/fuelPercentage	Option for extra MoI and CoG analysis
F_PERC	if USER_CASE = 'True'	0	%	TSP/balance/userBalance/payloadPercentage	Fuel Percentage for user case CoG and MoI Analysis
P_PERC	if USER_CASE = 'True'	0	%	TSP/balance/userBalance/userCase	Payload Percentage for user case CoG and MoI Analysis
WING_MOUNTED	no	'True'	-	TSP/propulsion/wingMountedEngine	'True' if the engines are placed on the main wing
maximum_take_off_mass	yes	-	kg	MBD/designMasses/mTOM/mass	Maximum take off mass
mass_fuel_max	yes	-	kg	MBD/fuel/massDescription/mass	Max fuel mass
mass_fuel_maxpass	yes	-	kg	TSP/weight/passengers/fuelMassMaxpass/mass	Fuel mass with maximum payload
operating_empty_mass	yes	-	kg	MBD/mOEM/massDescription/mass	Operating empty mass
mass_payload	yes	-	kg	MBD/payload/massDescription/mass	Max payload mass
TSP = /cpacs/toolspecific/CEASIOMpy				MBD = cpacs/vehicles/aircraft/model/analysis/massBreakdown	

Table B.3: Input data for the weight analysis of conventional aircraft.

Name	Required	Default Value	UoM	Path or xPath	Description
<b>Weight Module</b>					
cpacs	no	True	-	User Input	True if cpacs.xml file as input or False if direct data input
cpacsfile.xml	yes	-	-	CEASIOMpy/lib/1Weight_module/ToolInput	Aircraft geometry in the CPACS.xml format
AircraftData2018_v1_ste.cvs	yes	-	-	CEASIOMpy/lib/1Weight_module/ToolInput	Aircraft database for conventional aircraft
Fuselage Geometry	yes	-	-	/cpacs/vehicles/aircraft/model/fuselages	Geometry in CPACS format
Wing Geometry	yes	-	-	/cpacs/vehicles/aircraft/model/wings	Geometry in CPACS format
Fuselage Profile	yes	-	-	/cpacs/vehicles/profiles/fuselageProfile	Geometry in CPACS format
Wing Airfoil	yes	-	-	/cpacs/vehicles/profiles/wingAirfoil	Geometry in CPACS format
input_data	if not cpacs	-	-	CEASIOMpy/lib/1Weight_module/Weight_main.py	Wing-area, wing-span, fuse_length, fuse_width
IS_DOUBLE_FLOOR	no	0	-	TSP/geometry/isDoubleFloor	Double floor option
SEAT_WIDTH	no	0.525	m	TSP/geometry/seatWidth	Seat width
SEAT_LENGTH	no	0.74	m	TSP/geometry/seatLength	Seat length
AISLE_WIDTH	no	0.42	m	TSP/geometry/aisleWidth	Aisle width
FUSE_THICK	no	6.3	%	TSP/geometry/fuseThick	Fuselage thickness
tl_length	no	1.2	m	TSP/geometry/ttlLength	Base lavatory length
PILOT_NB	no	2	-	TSP/weight/crew/pilots/pilotNb	Number of pilots
MASS_PILOT	no	102	kg	TSP/weight/crew/pilots/pilotMass	Pilot mass
MASS_CABIN_CREW	no	68	kg	TSP/weight/crew/cabinCrewMembers/cabinCrewMemberMass	Cabin crew mass
PASS_PER_TLT	no	50	-	TSP/weight/passengers/passPerToilet	Passengers per toilet
MASS_PASS	no	105	kg	TSP/weight/passengers/passMass	Passenger mass
MASS_CARGO	no	0	kg	MBD/payload/mCargo/massDescription	User option for extra cargo mass
MAX PAYLOAD	no	0	kg	TSP/weight/massLimits/maxPayload	Maximum allowed payload
MAX FUEL VOL	no	0	l	TSP/weight/massLimits/maxFuelVol	Maximum allowed fuel volume
TURBOPROP	no	False	-	TSP/propulsion/turboprop	'True' only if the aircraft is a turboprop
RES_FUEL_PERC	no	0.1	-	TSP/propulsion/resFuelPerc	Unusable fuel percentage
FUEL_DENSITY	no	800	kg/m <sup>3</sup>	cpacs/vehicles/fuels/fuel/density	Fuel density
TSP = /cpacs/toolspecific/CEASIOMpy				MBD = cpacs/vehicles/aircraft/model/analysis/massBreakdown	

Table B.4: Input data for the unconventional weight analysis.

Unconventional Weight Module						
Name	Required	Default Value	UoM	Path or xPath	Description	
cpacfile.xml	yes	-	-	CEASIOmPy/lib/4 Weight_unc_module/ToolInput	Aircraft geometry in the CPACS.xml format	
PILOT_NB	no	2	-	TSP/weight/crew/pilots/pilotNb	Number of pilots	
MASS_PILOT	no	102	kg	TSP/weight/crew/pilots/pilotMass	Pilot mass	
MASS_CABIN_CREW	no	68	kg	TSP/weight/crew/cabinCrewMembers/cabinCrewMemberMass	Cabin crew mass	
PASS_PER_TLT	no	50	-	TSP/weight/passengers/passPerToilet	Passengers per toilet	
MASS_PASS	no	105	kg	TSP/weight/passengers/passMass	Passenger mass	
MAX_PASS	no	0	-	'/TSP/weight/passengers/passNb'	Maximum number of passengers	
MAX_PAYLOAD	no	0	kg	TSP/weight/massLimits/maxPayload	Maximum allowed payload	
MAX_FUEL_VOL	no	0	l	TSP/weight/massLimits/maxFuelVol	Maximum allowed fuel volume	
MASS_CARGO	no	0	kg	MBD/payload/mCargo/massDescription/massCargo	Mass of the cargo inside the cargo bay.	
RES_FUEL_PERC	no	0.06	-	TSP/fuels/resFuelPerc	Unusable fuel percentage	
FUEL_DENSITY	no	800	kg/m <sup>3</sup>	cpacs/vehicles/fuels/fueldensity	Fuel density	
VRT_THICK	no	0.00014263	m	'/TSP/geometry/virtualThick'	Aircraft virtual thickness	
VRT_STR_DENSITY	no	2700	kg/m <sup>3</sup>	'/TSP/geometry/virtualDensity'	Aircraft virtual material density	
PASS_BASE_DENSITY	no	1.69	pax/m <sup>2</sup>	TSP/weight/passengers/passDensity	Passengers per square meter inside the cabin	
PASS_PER_TLT	no	50	pax/lavatory	TSP/weight/passengers/passPerToilet	Passengers per lavatory	
SINGLE_HYDRAULICS	no	False	-	TSP/propulsion/singleHydraulics	False is the aircraft will have a multiple hydraulics system	
HLLIM_CABIN	no	2.0	m	'/TSP/geometry/cabinHeight'	Minimum high of the fuselage to contain passenger	
LD	no	17.0	-	TSP/ranges/IDRatio	Lift over drag coefficient	
wing_loading	no	600.0	kg/m <sup>2</sup>	-	Wing loading first guess to initialise the iteration process	
TSP = /cpacs/toolspecific/CEASIOmPy				MBD = cpacs/vehicles/aircraft/model/analysis/massBreakdown		

Table B.5: Input data for the unconventional balance analysis.

Unconventional Balance Module						
Name	Required	Default Value	UoM	Path or xPath	Description	
cpacsfile.xml	yes	-	-	CEASIOmPy/lib/6Balance_unc_module/ToolInput	Aircraft geometry in the CPACS.xml format	
USER_CASE	no	False	-	TSP/balance/userBalance/userCase	Option for extra MoI and CoG analysis	
F_PERC	if USER_CASE = 'True'	0	%	TSP/balance/userBalance/fuelPercentage	Fuel Percentage for user case CoG and MoI Analysis	
P_PERC	if USER_CASE = 'True'	0	%	TSP/balance/userBalance/payloadPercentage	Payload Percentage for user case CoG and MoI Analysis	
maximum_take_off_mass	yes	NaN	kg	MBD/designMasses/mTOM/mass	Maximum take off mass	
mass_fuel_max	yes	NaN	kg	MBD/fuel/massDescription/mass	Max fuel mass	
mass_fuel_maxpass	yes	NaN	kg	TSP/weight/passengers/fuelMassMaxpass/mass	Fuel mass with maximum payload	
operating_empty_mass	yes	NaN	kg	MBD/mOEM/massDescription/mass	Operating empty mass	
mass_payload	yes	NaN	kg	MBD/payload/massDescription/mass	Max payload mass	
USER_EN_PLACEMENT	no	False	-	'/TSP/propulsion/userEnginePlacement'	True if the User define the Engine Placement	
WING_MOUNTED	no	False	-	TSP/propulsion/wingMountedEngine	'True' if the engines are placed in the main wing	
H_LIM_CABIN	no	2.3	m	'/TSP/geometry/cabinHeight'	minimum high of the fuselage to allow passenger	
FLOORS_NB	no	1	-	'/TSP/geometry/floorsNb'	Number of floors o the aircraft	
TSP = /cpacs/toolspecific/CEASIOmPy				MBD = cpacs/vehicles/aircraft/model/analysis/massBreakdown		

Table B.6: Input data for the engine definition.

Unconventional Module Engine Inputs						
Name	Required	Default Value	UoMt	Path or xPath	Description	
NE	no	2	-	'/TSP/propulsion/engineNumber'	Number of Engines	
en_mass	yes if USER_ENGINE = True	-	kg	'/cpacs/vehicles/engines/ engine/analysis/mass/mass'	Mass of a single mounted engine (Dry Weight)	
EN_PLACEMENT	yes if USER_EN_PLACEMENT = True	[[x1,y1,z1],[x2,y2,z2], ...]	[m,m,m]	/cpacs/vehicles/engines/ engine/analysis/mass/location' + /x, /y, /z	Placement of each engine	
EN_NAME	no	[Engine_1, Engine_2, ...]	-		Name of each engine	
TURBOPROP	no	False	-	'/TSP/propulsion/turboprop'	'True' only if the aircraft has turboprop engines	
APU	no	True	-	'/TSP/propulsion/auxiliaryPowerUnits'	'True' if the aircraft has an APU engine	
WING_MOUNTED	no	True	-	TSP/propulsion/wingMountedEngine	'True' if the engine are placed on the main wing of the aircraft	
max_thrust	yes if USER_ENGINE = True	-	kN	'/cpacs/vehicles/engines/ engine/analysis/thrust00'	Maximum aircraft thrust, evaluated by the code or defined in the CPACS file	
TURBOPROP	no	False	-	TSP/propulsion/turboprop	'True' only if the aircraft has turboprop engines	

TSP = /cpacs/tools/specific/CEASIOMpy

Table B.7: Input data for the range analysis.

Name	Required	Default Value	UoM	Path or xPath	Description
<b>Range module</b>					
cpacs.file.xml	yes	-	-	CEASIOMpy/lib/Range_module/ToolInput	Aircraft geometry in the CPACS.xml format
cabin_crew_nb	yes	-	-	TSP/weight/crew/cabinCrewMembers/cabinCrewMemberNB	Number of cabin crew members
pilot_nb	no	2	-	TSP/weight/crew/pilots/pilotNb	Number of pilots
maximum_take_off_mass	yes	-	kg	MBD/designMasses/mTOM/mass	Maximum take off mass
mass_fuel_max	yes	-	kg	MBD/fuel/massDescription/mass	Max fuel mass
mass_fuel_maxpass	yes	-	kg	TSP/weight/passengers/fuelMassMaxpass/mass	Fuel mass with maximum payload
operating_empty_mass	yes	-	kg	MBD/mOEM/massDescription/mass	Operating empty mass
mass_payload	yes	-	kg	MBD/payload/massDescription/mass	Max payload mass
TSFC_CRUISE	no	0.5	1/hr	TSP/propulsion/tsfcCruise	Thrust specific fuel consumption on cruise
TSFC_LOITER	no	0.4	1/hr	TSP/propulsion/tsfcLoiter	Thrust specific fuel consumption on loiter
WINGLET	no	0	-	TSP/geometry/winglet	Reduces the TSFC in relation with the winglet type used
TURBOPROP	no	False	-	TSP/propulsion/turboprop	'True' only if the aircraft is a turboprop
CRUISE_SPEED	no	252	m/s	TSP/ranges/cruiseSpeed	Aircraft maximum cruise speed
LD	no	17	-	TSP/ranges/IDRatio	Lift over drag coefficient
MASS_PILOT	no	102	kg	TSP/weight/crew/pilots/pilotMass	Pilot mass
MASS_CABIN_CREW	no	68	kg	TSP/weight/crew/cabinCrewMembers/cabinCrewMemberMass	Cabin crew mass
MASS_PASS	no	105	kg	TSP/weight/passengers/passMass	Passenger mass
FUEL_DENSITY	no	800	kg/m <sup>3</sup>	cpacs/vehicles/fuels/fuelDensity	Fuel density
LOITER_TIME	no	30	min	TSP/ranges/loiterTime	Loiter duration
TSP = /cpacs/toolspecific/CEASIOMpy				MBD = cpacs/vehicles/aircraft/model/analysis/massBreakdown	

# Bibliography

- [1] European Organisation for the Safety of Air Navigation (EUROCONTROL) 2018. *EUROCONTROL Flight Forecast to 2040 – Annex 1 to the Challenges of Growth 2018 report – September 2018*. 2018. URL: <https://www.eurocontrol.int/sites/default/files/content/documents/official-documents/reports/challenges-of-growth-annex-1-25092018.pdf> (visited on 02/11/2019).
- [2] Airbus. *Airbus' Global Market Forecast*. 2018. ISBN: 978-2-9554382-3-6.
- [3] A. Rizzi. "Modeling and simulating aircraft stability and control—the SimSAC project". In: *Progress in Aerospace Sciences* 47.8 (2011), pp. 573–588. doi: 10.1016/j.paerosci.2011.08.004.
- [4] U.P. Breuer. "Requirements, Development and Certification Process". In: (2016), pp. 25–44. doi: 10.1007/978-3-319-31918-6\_2.
- [5] D. Darracq E.H. Hirschel J.B. Vos A.Rizzi. "Navier-Stokes solvers in European aircraft design". In: *Progress in Aerospace Sciences* 38.8 (2002), pp. 601–697. doi: 10.1016/S0376-0421(02)00050-7.
- [6] CFS Engineering SA. *CEASIOM - computerised environment for aircraft synthesis and integrated optimization methods*. 2018. URL: <https://github.com/cfsengineering/CEASIOMpy> (visited on 09/22/2018).
- [7] DLR-SC. *CPACS - common parametric aircraft configuration schema*. 2018. URL: <https://github.com/DLR-LY/CPACS> (visited on 09/22/2018).
- [8] Roy H. Lange. "Review of unconventional aircraft design concepts". In: *Journal of Aircraft* 25.5 (1988), pp. 385–392. doi: 10.2514/3.45592.
- [9] Roberto Merino Martínez. "Design and Analysis of the Control and Stability of a Blended Wing Body Aircraft". MA thesis. Royal Institute of Technology (KTH), 2014.
- [10] M. Marino and R. Sabatini. "Advanced lightweight aircraft design configurations for green operations". In: *Proceedings of the Practical Responses to Climate Change 2014 (PRCC 2014)*. Engineers Australia. Meenakshi Arora, Geoff Sutherland, Graham Moore, 2014.
- [11] CFS Engineering SA. *CEASIOM - computerised environment for aircraft synthesis and integrated optimization methods*. 2018. URL: <https://www.ceasium.com/> (visited on 09/22/2018).
- [12] P. D. Ciampa German Aerospace Center (DLR) and B. Nagel. "The AGILE Paradigm: the next generation of collaborative MDO". In: *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference Proceedings*. Denver, Colorado, 2017.

- [13] B. Nagel D. Boehnke P. Saquet A. Rizzi M.Zhang. “Towards a unified framework using CPACS for geometry management in aircraft design”. In: *50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*. Nashville, Tennessee, 2012.
- [14] Capt. T. Buckland Capt. R.M.H. Weeks N.J. Butcher Dr.J.C. Barnet. “Emergency Evacuation of Commercial Passenger Aeroplanes”. In: (April 2018).
- [15] D. Kleppner and R. Kolenkow. *An introduction to mechanics, 2nd ed.* Cambridge University Press, 2014. ISBN: 978-0-0521-19811-0.
- [16] C. Lanham. *Inertia Calculation Procedure for Preliminary Design*. Tech. rep. ADA072679. Ohio: Aeronautical systems division Integration Division, Wright-Patter AFB, 1979.
- [17] David A. Levinson Thomas R. Kane. *Dynamics: theory and applications*. McGraw Hill, 1985. ISBN: 0-07-037846-0.
- [18] Daniel P. Raymer. *Aircraft Design: A Conceptual Approach, Fourth Edition*. AIAA American Institute of Aeronautics & Ast., 2006. ISBN: 978-1563478291.
- [19] National Transportation Safety Board. *Preliminary estimation of airplane moments of inertia using CAD solid modeling*. Annual Review NTSB/ARA-14/01. Washington, DC, 2014.
- [20] ICAO International Civil Aviation Organization. “Operation of Aircraft, Part I: International Commercial Air Transport — Aeroplane”. In: *Annex 6 to the Convention on International Civil Aviation*. 2010.
- [21] David G Hull. *Fundamentals of airplane flight mechanics*. Springer, 2007. ISBN: 9783642079870.
- [22] M. Hewitt R. Langton C. Clark and L. Richards. *Aircraft fuel systems*. John Wiley & Sons, 2009. ISBN: 978-0-470-05708-7.
- [23] European Aviation Safety Agency. “Type-certificate data sheet, aircraft: ATR 42 - ATR 72”. In: EASA.A.064.3 (22 Jan 2019).
- [24] Airbus S.A.S. “A330 aircraft characteristics airport and maintenance planning”. In: (Jul. 2018).
- [25] J.T. van den Broek-Serie F.N. Berdowski and Y.K. Jetten. “Survey on standard weights of passengers and baggage”. In: *European Aviation Safety Agency, Cologne, Germany* (2009). Ref.:EASA 2008.C.06/30800/R20090095/30800000/FBR/RLO.
- [26] Robert J. Mark. “A Rapid Empirical Method for estimating the gross takeoff weight of a high speed civil transport”. In: (1999). DOI: NASATM- 1999-209535.
- [27] DLR-SC. *TiXI: Fast and simple xml interface library*. 2018. URL: <https://github.com/DLR-SC/tixi> (visited on 09/27/2018).
- [28] DLR-SC. *The TiGL geometry library to process aircraft geometries in pre-design*. 2018. URL: <https://github.com/DLR-SC/tigl> (visited on 09/27/2018).
- [29] T. Stollenwerk-R. Maierl M. Siggel J. Kleinert. “TiGL - An Open Source Computational Geometry Library for Parametric Aircraft Design”. In: (2018).



- [30] John D. Anderson Jr. *Introduction to flight*. McGraw-Hill Education 8th edition, 1987. ISBN: 978-9814636186.
- [31] E. Torenbeek. *Advanced Aircraft Design: Conceptual Design, Analysis and Optimization of Subsonic Civil Airplanes*. John Wiley and Sons, Ltd., 2013. ISBN: 9781118568118. DOI: 10.1002/9781118568101.
- [32] Boeing Commercial Airplanes. *777-200LR/-300ER/-Freighter Airplane Characteristics for Airport Planning*. Tech. rep. D6-58329-2, REV. E. 2015.
- [33] John WR.Taylor. *Jane's All the World's Aircraft 1976*. London : Jane's yearbooks, 1976. ISBN: 978-0354005388.
- [34] OSHA Regulations and Regulatory Guidance. *Title 14 - Aeronautics and Space*. Office of the Federal Register, 2011.
- [35] H. Tennekes. *The simple science of flight: from insects to jumbo jets*. The MIT press, 2009. ISBN: 978-0-262-51313-5.
- [36] European Aviation Safety Agency. *PW100 series*. certificate data sheet for engine EASA IM.E.041 TCDS Issue 4. 1000 Marie-Victorin Longueuil Quebec, Canada, 2018.
- [37] B.W. Kimoto-D.P. Marsh M.N. Beltramo D.L. Trapp. "Parametric study of transport aircraft systems cost and weight". In: (1977). ASIN: B0007338WC.
- [38] R. Ali O. Al-Shamma. "Aircraft weight estimation in interactive design process". In: *Procs 72nd Annual International Conference on Mass Properties Engineering*. 2013.
- [39] U.S. Government. "14 CFR § 121.523". In: *Code Of Federal Regulation*. Office of the Federal Register National Archives and Records Administration, 2018.
- [40] Robert J. Beichner Raymond A. Serway and John W. Jewett. *Physics for scientists and engineers, 5th ed*. Philadelphia : Saunders College Publishing, 2000. ISBN: 0030226546.
- [41] A. Dittberner M.B. Rivers. "Experimental investigations of the NASA common research model". In: *Journal of Aircraft* 51.4 (2014), pp. 1183–1193. DOI: 10.2514/1.c032626.
- [42] European Aviation Safety Agency. "Type-certificate data sheet, aircraft: ATR 42 - ATR 72". In: EASA.A.084 (07 Dec 2017).
- [43] Federal Aviation Administration Department of Transportation. *TYPE CERTIFICATE DATA SHEET, Concorde Type 1*. Tech. rep. NO. A45EU. 1979.
- [44] G.AF.Seber and A.J.Lee. *Linear regression analysis*. John Wiley & Sons, 2012. ISBN: 0-471-41540-5.