

AUTOMATIC AEROSTRUCTURAL ANALYSIS AND AERODYNAMIC OPTIMIZATION FOR COLLABORATIVE AIRCRAFT DESIGN PROCESS

Massimo Mandorino

"Federico II" University of Naples

*A thesis submitted for the degree of
Master Thesis*

Naples 2020

Abstract

Analysis of scenario

Overall aircraft design is multidisciplinary by nature, it means that results coming from multidisciplinary analysis can greatly improve the final design and reduce the time cost of the design cycle. In addition, a significant improvement can be reached if high-fidelity methods are included in this kind of analysis, especially when unconventional configuration need to be considered. However, several limitations do not allow the possibility to get the benefits of high-fidelity applications during early design phases. The main thesis motivation is to illustrate how it is possible to obtain detailed design analysis in reasonable time and with appropriate accuracy, taking into consideration the coupling between aerodynamic and structural properties. The thesis is the result of a collaboration between the DAF group of the University of Naples "Federico II" and the research group of DLR institute of Hamburg in which the candidate has completed an internship within the AGILE 4.0 project.

Statement of the problem

The work concerns computational fluid dynamic and structural coupled analysis performed through automated Python-based workflows. The tools employed are open-source or provided by partners. Therefore, an appropriate connection and data transfer among modules must be provided, complying with each tool or partner requirements. The implemented workflows provide aerodynamic analysis and optimization, aero-structural analysis at different flow conditions and centralized data format geometry update.

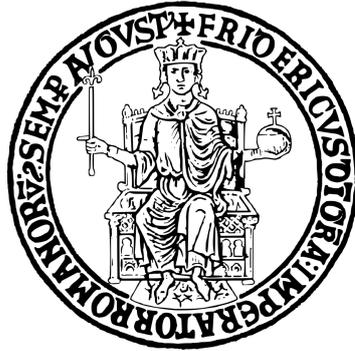
Adopted methodology

The automated aero-structural process developed is based on the DLR centralized data format CPACS. It defines the aircraft geometry and configuration and allows information sharing during the process. Pointwise and SU2 are employed respectively for mesh generation and computational aerodynamic analysis. LAGRANGE tool provides linear structural analyses. All modules are Python-based and connected to each other through RCE. SLSQP optimizer, adjoint, moving last square and FFD methods are employed.

Main results

Flexible and robust automatic workflows are implemented in a distributed design environment (RCE), provided by DLR. The input and output of the process are provided through CPACS format file. The workflows allow gradient-based shape optimization of a generic aircraft component and aero-structural analysis performed at different flow conditions. The output shape geometry is supplied through CPACS format file. An application on a UAV configuration is performed: mesh sensitivity analysis, aerodynamic constrained wing shape optimization in transonic flow condition, aero-elastic shape deformation in cruise condition and flexible polar analysis are carried out to demonstrate the workflow capabilities.

UNIVERSITÀ DEGLI STUDI DI NAPOLI
FEDERICO II
SCUOLA POLITECNICA E DELLE SCIENZE DI BASE



Department of Industrial Engineering

Master Degree Dissertation in
AEROSPACE AND ASTRONAUTIC
ENGINEERING

**Automatic Aero-Structural Analysis and
Aerodynamic Optimization for
Collaborative Aircraft Design Process**

Supervisor:
Prof. Ing. Pierluigi Della Vecchia

Co-supervisor:
Dott. Ing. Francesco Torrigiani

Candidate:
**Massimo Mandorino
M53000881**

Academic year 2019/2020

*To Martina
and my Family*

Abstract

Analysis of scenario

Overall aircraft design is multidisciplinary by nature, it means that results coming from multidisciplinary analysis can greatly improve the final design and reduce the time cost of the design cycle. In addition, a significant improvement can be reached if high-fidelity methods are included in this kind of analysis, especially when unconventional configuration need to be considered. However, several limitations do not allow the possibility to get the benefits of high-fidelity applications during early design phases. The main thesis motivation is to illustrate how it is possible to obtain detailed design analysis in reasonable time and with appropriate accuracy, taking into consideration the coupling between aerodynamic and structural properties. The thesis is the result of a collaboration between the DAF group of the University of Naples “Federico II” and the research group of DLR institute of Hamburg in which the candidate has completed an internship within the AGILE 4.0 project.

Statement of the problem

The work concerns computational fluid dynamic and structural coupled analysis performed through automated Python-based workflows. The tools employed are open-source or provided by partners. Therefore, an appropriate connection and data transfer among modules must be provided, complying with each tool or partner requirements. The implemented workflows provide aerodynamic analysis and optimization, aero-structural analysis at different flow conditions and centralized data format geometry update.

Adopted methodology

The automated aero-structural process developed is based on the DLR centralized data format CPACS. It defines the aircraft geometry and configuration and allows information sharing during the process. Pointwise and SU2 are employed respectively for mesh generation and computational aerodynamic analysis. LAGRANGE tool provides linear structural analyses. All modules are Python-based and connected to each other through RCE. SLSQP optimizer, adjoint, moving last square and FFD methods are employed.

Main results

Flexible and robust automatic workflows are implemented in a distributed design environment (RCE), provided by DLR. The input and output of the process are provided through CPACS format file. The workflows allow gradient-based shape optimization of a generic aircraft component and aero-structural analysis performed at different flow conditions. The output shape geometry is supplied through CPACS format file. An application on a UAV configuration is performed: mesh sensitivity analysis, aerodynamic constrained wing shape optimization in transonic flow condition, aero-elastic shape deformation in cruise condition and flexible polar analysis are carried out to demonstrate the workflow capabilities.

Contents

List of Figures	vi
List of Tables	xi
1 Introduction	1
1.1 Motivation	1
1.2 AGILE 4.0 Project	2
1.3 Thesis Outline	3
2 Computational Optimization for Aircraft Design	5
2.1 Computational Fluid Dynamics for Aircraft Design	5
2.1.1 Aircraft Design Benefits using CFD	5
2.1.2 Role of CFD during Aircraft Design Phases	7
2.2 Aerodynamic Optimization based on CFD	9
2.2.1 Main features of Aerodynamic Optimization	9
2.3 Multidisciplinary Design Analysis and Optimization	13
2.3.1 Collaborative Design	13
2.3.2 Multidisciplinary Design	15
2.3.3 MDO architecture	16
2.3.4 CPACS standard format	19
2.3.5 Requirements for MDO	20
2.4 Aero-Structural Analysis and Optimization for Aircraft Design	22
2.4.1 Importance of Integrated Aero-Structural Analysis and Optimization	23
2.4.2 Current High-Fidelity Method Applications	25
3 Instruments for Aerodynamic and Structural Analysis	29
3.1 Introduction	29
3.2 Automatic Mesh Generation process	30
3.2.1 Geometry Generation Module	31
3.2.2 Mesh Generation Modules	32
3.3 Stanford University Unstructured	34
3.3.1 Software architecture	34
3.3.2 Input and Output	36
3.3.3 Physical Modeling Equations	38
3.3.4 Adjoint Formulation and Sensitivity Computation	39
3.4 Automatic CSD analysis process	43
3.4.1 Structural tool LAGRANGE	44
3.4.2 Structural Process Architecture	45

4 Collaborative Aerodynamic Optimization and Aero-Structural Analysis	47
4.1 Introduction	47
4.2 Aerodynamic Analysis and Optimization	48
4.2.1 CFD Analysis Parallel Computation	48
4.2.2 Shape Optimization	49
4.2.3 Test Cases	54
4.3 Mesh Deformation	62
4.3.1 Deformation methods	62
4.3.2 FFD Mesh Deformation	63
4.3.3 Mesh Motion Points Deformation	67
4.4 Aero-Structural Analysis Workflow	69
4.4.1 Presentation of the Workflow	69
4.4.2 Input, Output and Configuration	72
4.4.3 Aero-Structural Interface Methods	74
4.4.4 Workflow Architecture	77
4.5 CPACS Interface	80
4.5.1 Need of a CPACS Interface	80
4.5.2 Aerodynamic Results Writer	81
4.5.3 CPACS Geometry Update	85
5 Application of Collaborative Aero-Structural Tools	95
5.1 Test Case: Optimale Configuration.	95
5.1.1 Aircraft configuration and TLAR	96
5.1.2 Aerodynamic Mesh definition	97
5.1.3 Structural Mesh definition	101
5.2 Aero-Structural Equilibrium in Cruise	102
5.2.1 Mesh and Input Data for Flexible Cruise	102
5.2.2 Flexible Cruise Results	104
5.3 Flexible Drag Polar	107
5.3.1 Mesh and Input Data for Flexible Wing Polar	107
5.3.2 Flexible and Rigid Wing Polar Results	108
5.3.3 Flexible and Rigid Optimale Polar	112
5.4 CPACS Output Generation	115
Conclusions and Future Developments	119
Bibliography	119

List of Figures

2.1	CFD contribution to Boeing 787 Aircraft Design.	6
2.2	Aircraft design process layout.	7
2.3	Optimization design process.	10
2.4	Possible problem decomposition: in series or in parallel.	14
2.5	Collaborative design structure.	15
2.6	Comparison between monolithic and distributed architecture.	17
2.7	CPACS schema.	19
2.8	CPACS concept.	20
2.9	Comparison between elliptical and aero-structural optimum lift distribution.	25
2.10	Integrated aero-structural optimization	26
3.1	XDSM graph of the automated mesh generation process.	30
3.2	Example of surface and volume mesh for viscous analysis on wing body configuration [26]	32
3.3	View of an FFD box with indices identifying control points [33]. . .	43
3.4	Architecture of the workflow used for structural calculation obtained from aerodynamic mesh nodes forces distribution.	45
4.1	Architecture of shape optimization tool. The arrows indicate temporal sequence of execution	51
4.2	AGILE Design Campaign 2 aircraft. Geometry visualization obtained through TiGl viewer.	55
4.3	AGILE Design Campaign 2 aircraft ParaView visualization of surface mesh surrounded by volumetric mesh.	55
4.4	AGILE Design Campaign 2 aircraft Mach number distribution in cruise condition.	56
4.5	AGILE Design Campaign 2 aircraft results from CFD analysis at low speed condition.	56
4.6	Optimale wing mesh surrounded by FFD box. Both data are provided as input to run the shape optimization.	58
4.7	Optimale wing shape optimization results. Trends of drag coefficients (expressed in drag counts) and angle of attack during optimization iterations.	59
4.8	Mach distribution comparison between original and optimized Optimale wing shape. It possible to observe the drastic reduction of the wave strength. The maximum number of mach reached is reduced from 1.5 to 1.2.	60

4.9	Pressure coefficient distribution comparison between original and optimized Optimale root section. The trend is similar for all other sections. It possible to observe the drastic reduction of the wave strength.	61
4.10	Comparison between original and deformed meshes and FFD boxes for three different magnitude of deformation of ONERA M6 wing geometry.	63
4.11	Cell Aspect Ratio distribution on mesh with 10% of deformation with respect to original one. It is possible to notice that only the 1.9% of cell is characterized by a value of the AR higher than 10.	65
4.12	ONERA M6 wing generated using Global Scale = 0.75 as input. The number of cell composing the mesh is around 29 thousand.	66
4.13	ONERA M6 wing and related FFD box deformation obtained through a symmetric displacement in z direction of FFD control points. The increase of thickness reached is around 20%.	66
4.14	Optimale wing comparison between original and deformed shape representing the typical effect of bending moment on wing in flight. A tip displacement of 1.3 meters is reached, it corresponds to approximately 8% of wing span.	68
4.15	RCE visualization of the workflow built in order to obtain Aero-Structural equilibrium of a generic configuration at different flow conditions.	70
4.16	Comparison between DC-2 configuration modal pressure coefficient distribution calculated in cruise condition through SU2 and cell forces distribution subsequently calculated.	76
4.17	DC-2 configuration force vectors. The length of the arrow depends on the magnitude of the force, all vector are oriented towards the outgoing normal.	76
4.18	Flowchart representing the aero-structural workflow architecture. Input, output and data-transmission among tools are illustrated. It is also possible to observe the sequence of tools executed.	78
4.19	Flowchart representing the general workflow which can be defined through the execution of the tools described in Chapters 3 and 4. A CPACS file with tools results can be generated in this way.	81
4.20	Comparison between CPACS 2 and 3 schema for the section which describes aircraft aerodynamic performance. It is possible to notice a slight difference between the two versions [40].	82
4.21	Illustration of CPACS, body and aerodynamic coordinate system taken into consideration by CPACS file. All the reference systems are right-handed coordinate systems [40].	84

4.22	Possible definition of the wing geometry through sections, elements and segments. Wing, section and element reference system are illustrated [40].	86
4.23	Execution order of the steps carried on by CPACS to generate a general geometry [40].	86
4.24	Architecture of CPCAS Geometry Update tool. It is possible to notice that the geometry update can be subdivided in three different steps: airfoil points reading, airfoil point modification and CPACS file update.	88
4.25	Representation of a twisted swept with non-zero dihedral wing sections modification before and after the first three normalization steps performed by CPACS geometry update tool.	91
4.26	Comparison between CPACS original and updated geometry of Optimale configuration. The right geometry represents 25 slices obtained after the wing shape optimization described in section 4.2.3.	93
4.27	Representation of points coordinates which have to be insert to generate a CPACS guide curve, a way to define a connection between two wing sections. Three coordinates for each guide curve point must be provided and a curve that interpolates the wing shape passing through these points is generated [40].	93
5.1	Optimale CPACS configuration represented through TiGl viewer.	96
5.2	Optimale surveillance mission. Representation of mission segments.	97
5.3	Lift and drag coefficients trend with reference to the initial value of mesh cell average edge length. Also the parameter "Global Scale" used as input for the generation of the mesh is indicated.	99
5.4	Optimale CPACS structural model and thickness distribution representation. The first figure represent the wing without upper skin which instead is shown in the second one.	101
5.5	Optimale complete aircraft surface mesh surrounded by volumetric mesh. This grid is used as input for the computation of aero-structural deformation in cruise condition.	103
5.6	Trend of maximum value of deformation, angle of attack and drag coefficient computed through loop iterations performed to obtain Optimale elastic cruise condition. It is possible to observe that the greatest iteration is the first, the reaming are required only to reach the desired value of convergence.	105

5.7	Comparison between rigid and flexible twist distribution of Optimale wing in cruise condition. All the section experience a reduction of twist angle that is higher for section near to the wing tip. The average difference between the two distributions corresponds to 0.24 degrees.	106
5.8	Comparison between rigid and flexible Optimale configuration shape. The maximum value of deformation is reached at wing tip, corresponding to a deflection of 0.4 meters. The remaining part of the aircraft does not experience an appreciable value of deformation. . .	106
5.9	Top view of the Optimale wing mesh employed to perform aerodynamic calculations for aero-structural equilibrium workflow. The mesh is chosen in order to both satisfy accuracy and computational time desired.	107
5.10	Comparison between Optimale wing shape obtained after aero-structural equilibrium computation at flow condition indicated in 5.6. It possible to observe deformations in both possible direction of bending deflections.	109
5.11	Trend of the maximum displacement value between the rigid and flexible Optimale wing shape at flow condition indicated in 5.6. It is possible to observe how the trend is similar to the lift trend of Fig. 5.12. Indication of the linearity of solution.	109
5.12	Trend of lift coefficient with reference to angle of attack and drag coefficient. The trend is compared between rigid and flexible calculation. It is possible to see that the lift coefficient is very similar between the two cases.	110
5.13	Comparison between pitching moment trend coefficient obtained though rigid and flexible calculation.	110
5.14	Comparison between Optimale wing twist distribution obtained through rigid and flexible analysis at different angle of attack. Only two angle of attack are considered for flexible case. The higher and the lower.	111
5.15	Comparison between rigid and flexible Optimale wing section. The incidence of this results is 4° . The y coordinates of the wing station is 0.9. The airfoil shape and the pressure coefficient distribution are shown.	112
5.16	Comparison between the flexible and rigid pithing moment coefficient trend of the entire Optimale configuration with respect to alpha. . .	113
5.17	Pitching moment coefficient breakdown of the Optimale configuration components. Wing, fuselage and horizontal plane are taken into consideration for both rigid and flexible calculation.	114

5.18	Optimale horizontal tail lift coefficient curve computed for rigid and flexible calculation. The lift coefficient is scaled with referce to the horizontal tail surface.	115
5.19	Comparison between input and output of "CPACS Geometry Update" tool. Wing shape deformation results obtained in section 5.2 are used as input.	116

List of Tables

3.1	Input parameters for Geometry Tool.	31
3.2	Input parameters for Geometry Tool.	32
3.3	Input SU2 parameters for Eureka compressible analysis.	37
4.1	Solution data available from CFD analysis performed with SU2	50
4.2	Input parameters for shape optimization tool. In addition, also parameter indicated in Tab. 3.3 must be provided.	53
4.3	Solution data available after shape optimization. In addition, solution data presented in Tab. 4.1 are provided.	54
4.4	Input data for CFD analysis of Design Campaign 2 aircraft at cruise and low speed condition.	54
4.5	Main input parameters for Optimale shape optimization tool.	57
4.6	Skewness and aspect ratio value computed for the meshes illustrated in Fig. 4.10. Also recommended value are indicated.	64
4.7	Skewness and aspect ratio value computed for the geometry meshes illustrated in Fig. 4.13. Also recommended value are indicated.	67
4.8	Skewness and aspect ratio value computed for the geometry meshes illustrated in Fig. 4.14. Also recommended value are indicated.	69
4.9	Inputs required in order to execute the aero-structural equilibrium workflow.	71
4.10	Output files and parameters provided for each iteration after the execution of the aero-structural equilibrium workflow.	73
4.11	Description of the attributes that can be added in Aeroperformance one. Both the attributes allowed for CPACS 2.3 and 3 are listed	83
5.1	OPTIMALE Top Level Aircraft Requirements for transfer and surveillance missions.	97
5.2	Input data for CFD analysis computed with different mesh in order to analyse the effect of the mesh node density on aerodynamic coefficients.	98
5.3	Number of cells and computational time spent for compute results shown in Fig. 5.3. The machine exploited posses four number of processors.	100
5.4	Main feature of aerodynamic mesh chosen for subsequent calculation. The mesh is one those used as input to obtain results plotted in Fig. 5.3.	100
5.5	Parameters used for flexible cruise condition calculation through aero-structural workflow. Data shown are not all independent from each other.	104

5.6 Parameters used for flexible cruise condition calculation through
aero-structural workflow. Data shown are not all independent from
each other. 108

List of Symbols

Symbol	Units	Description
W	kg	System Mass
g	m/s^2	Gravitational acceleration
ρ	kg/m^3	Air density
V	m/s	Flow velocity
S	m^2	Wing surface
α	Degree	Angle of attack
E	$Joule/kg$	Energy per unit of mass
P	Pascal	Static pressure
H	$Joule/kg$	Fluid enthalpy
C_p	$Joule/kgKelvin$	Specific heat at constant pressure
T	$Kelvin$	Temperature
R	$Joule/(Kelvin * mol)$	Gas constant
F	$Newton$	Force
A	m^3	Area
LE	dimensionless	Leading Edge
AR	dimensionless	Aspect Ratio
MAC	dimensionless	Mean Aerodynamic Chord
n	dimensionless	Local normal vector
p_∞	$Pascal$	Free-stream pressure
V_∞	m/s^2	Free-stream velocity
$avgDS1$	mm	Preliminary average cell edge length
C_L	dimensionless	Lift coefficient
C_D	dimensionless	Drag coefficient
C_Y	dimensionless	Lateral force coefficient
C_{F_x}	dimensionless	Force in x direction coefficient
C_{F_y}	dimensionless	Force in y direction coefficient
C_{F_z}	dimensionless	Force in z direction coefficient
C_{M_x}	dimensionless	Rolling moment coefficient
C_{M_y}	dimensionless	Pitching moment coefficient
C_{M_z}	dimensionless	Yawing moment coefficient
Re	dimensionless	Reynolds number
M	dimensionless	Mach number
CFD	dimensionless	Computational Fluid Dynamics
CSD	dimensionless	Computational Structural Dynamics
FFD	dimensionless	Free Form Deformation
DOE	dimensionless	Design Of Experiments
MDO	dimensionless	Multidisciplinary Design Optimization
$MDAO$	dimensionless	Multidisciplinary Design Analysis and Optimization
$CPACS$	dimensionless	Common Parametric Aircraft Configuration Schema

Symbol	Units	Description
<i>RANS</i>	dimensionless	Reynolds-Averaged Navier-Stokes
<i>XML</i>	dimensionless	Extensible Markup Language
<i>IGES</i>	dimensionless	Initial Graphics Exchange Specification
<i>PDE</i>	dimensionless	Partial Differential Equation
<i>MPI</i>	dimensionless	Message Passing Interface
<i>UAV</i>	dimensionless	Unmanned Aerial Vehicle
<i>MALE</i>	dimensionless	Medium Altitude Long Endurance
<i>DC2</i>	dimensionless	Design Campaign 2
<i>CSV</i>	dimensionless	Comma Separated Value
<i>TSV</i>	dimensionless	Tab-Separated Value
<i>TLAR</i>	dimensionless	Top Level Aircraft Requirements
<i>FEM</i>	dimensionless	Finite Element Method
<i>GS</i>	dimensionless	Global Scale

1

Introduction

Contents

1.1	Motivation	1
1.2	AGILE 4.0 Project	2
1.3	Thesis Outline	3

1.1 Motivation

The work presented is the results of the author’s master thesis activity developed in collaboration with Design of Aircraft and Flight technologies research group (DAF) of University of Naples Federico II and the Institute of System Architectures in Aeronautics of the German Aerospace Center (DLR) in Hamburg. The main topics of the work concern aerodynamic optimization and aero-structural coupling analysis obtained through the development of automatic Python-based workflow. Therefore, the implementation and application of flexible and robust computational fluid and structural dynamic workflows is discussed.

The development of aeronautical products is a complex multidisciplinary process with requirements and constraints on the air transport system as a whole and on all the individual components to be produced. In addition, the increasing technology level over the last decades has enabled the possibility to raise the detail knowledge of the early design phases. Furthermore, multi-fidelity approaches with growing level of fidelity have been developed during last years. Unfortunately, even current multidisciplinary design techniques are not able to incorporate high-fidelity analyses in early design phases. Despite the considerable progress made there are still

significant obstacles to be overcome in the development of numerical methods, physical modeling, and the integration of different aircraft disciplines. The aim of the presented work is to illustrate how detailed aircraft design analysis can be performed for different and distant points of the design space in a reasonable time and with appropriate accuracy. In this way, the designer can be led to the best choice for the new product configuration which can be also different from traditional design solutions.

In order to obtain a significant reduction in development costs of aircraft through the implementation of a more competitive supply chain at the early stages of design, the European AGILE project has been originated. After that, AGILE 4.0 project is born in order to significantly extend the first project scope adding manufacturing, maintenance, and certification aspects and extending the aircraft product optimization to the entire life cycle and addressing the extensive aeronautical supply chain. AGILE 4.0 project also aims to create an integrated Model Based Systems Engineering framework for aircraft applications. The thesis work, other than analyzing the aircraft configurations developed during the above mentioned projects, is basically based on AGILE paradigm.

1.2 AGILE 4.0 Project

A great challenge in the transport sector is to obtain economic growth, good standards of innovation and a high competitiveness in a compatible way with sustainability and environmental constraints. To deal with this challenge, the aviation industry needs incremental improvements of existing technologies. Highly innovative solutions, such as unconventional concepts or modifications in the development process, are required to achieve an efficient and cost-effective design process. Such a digital transformation requires a novel design paradigm to enable a fast and efficient integration of multidisciplinary models, accounting for design, manufacturing and certification requirements. These are the main features of AGILE paradigm. The AGILE 4.0 project is built upon the success of its predecessor H2020 project AGILE (Aircraft 3rd Generation MDO for Innovative Collaboration of

Heterogeneous Teams of Experts 2015-2018). The high level objective of AGILE 4.0 is to bring significant reductions in aircraft development costs and time-to-market through the implementation of an integrated cyber-physical aeronautical supply chain, increasing the competitiveness of the European aircraft industry, from integrators and suppliers to the enterprises, leading to innovative and more sustainable aircraft products.

1.3 Thesis Outline

The central goal of the thesis is to provide reliable preliminary design methods for aero-structural analysis and aerodynamic optimization in a automatic design environment. Therefore, an automated aerodynamic and structural processes have been developed in order to perform gradient-based shape optimization of a generic aircraft component and aero-structural analysis at different flow conditions. The processes are based on DLR centralized data format file CPACS and are integrated and executed through RCE (Remote Component Environment) in order to facilitate the collaborative aspect of design analysis.

Chapter 1 focuses on the context in which the work has been developed. A general description of the thesis motivation is provided and an overview of European AGILE 4.0 project is exposed.

Chapter 2 gives an overview of the role of computational fluid and structural dynamic calculation computed for simple or multidisciplinary analysis and optimizations during the different design phases. The chapter is mainly focused on the early design stage. Also an overview of the high fidelity aero-structural analysis state-of-the-art is provided.

Chapter 3 describes the already existing tools employed for the subsequent analysis. A brief description of inputs outputs and functionalities of each tool and method is provided. An automatic mesh generation process is presented and aerodynamic and structural analysis tools are described also focusing on the adopted methodologies.

Chapter 4 describes the tool and workflows developed by the author. For each one a detailed description of the architecture is provided. Possible inputs and outputs

which can be obtained for the process are described and test cases are executed in order to demonstrate the capabilities of the tool or workflow. The processes presented concern aerodynamic analysis and optimization, mesh deformation, aero-structural analysis and geometry described in common parametric language update. **Chapter 5** presents the application of the processes previously described. The test case concern the analysis and optimization of a UAV configuration. A description of the object under analysis and of the aerodynamic and structural input is provided, including a sensitivity mesh analysis in order to choose the best input grid. Then, aero-structural analysis in cruise condition and at different angle of attack is performed. In conclusion, the geometry update on common parametric language file is presented.

2

Computational Optimization for Aircraft Design

Contents

2.1	Computational Fluid Dynamics for Aircraft Design . . .	5
2.1.1	Aircraft Design Benefits using CFD	5
2.1.2	Role of CFD during Aircraft Design Phases	7
2.2	Aerodynamic Optimization based on CFD	9
2.2.1	Main features of Aerodynamic Optimization	9
2.3	Multidisciplinary Design Analysis and Optimization .	13
2.3.1	Collaborative Design	13
2.3.2	Multidisciplinary Design	15
2.3.3	MDO architecture	16
2.3.4	CPACS standard format	19
2.3.5	Requirements for MDO	20
2.4	Aero-Structural Analysis and Optimization for Aircraft Design	22
2.4.1	Importance of Integrated Aero-Structural Analysis and Optimization	23
2.4.2	Current High-Fidelity Method Applications	25

2.1 Computational Fluid Dynamics for Aircraft Design

2.1.1 Aircraft Design Benefits using CFD

The capability of simulate aerodynamic flow using CFD has greatly improved during the last decades. This is happened thanks to the enormous enhancement in calculation time and accuracy that the modern machine are able to provide. The development in computer technology has radically changed the aerospace design

process, introducing during the years an increasing number of numerical methods. Nowadays, methods such as CFD or FEM are widely used during the aircraft design process. Nevertheless, the role of CFD during design phases is not always the same, it usually changes taking into account different vehicle, type component or flight conditions. Usually, numerical methods provides a great alternative or something complementary to traditional ground-based and in-flight test. In this way the time commonly spent in wind tunnel test, rig test, engine certification test and flight test is dramatically reduced. For instance, thanks to the exploitation of CFD analysis the amount of wind tunnel test required for the design and production of aircraft such as Boeing 777 and 787 has decreased significantly in comparison with previous design. As illustrated in Fig. 2.1, during the design of the Boeing 787 analytical methods were widely used for several application. CFD was employed for the design of lifting surface, cabin, nacelle and many other components, for correction of data from wing tunnel test, for thermal analysis of engines and so on. It is possible to notice that also some applications that are emerging nowadays are mentioned: multi-disciplinary analysis (regarding aero-elasticity, aero-acoustics and aero-optics), automated optimizations and studies on unsteady flow [1].

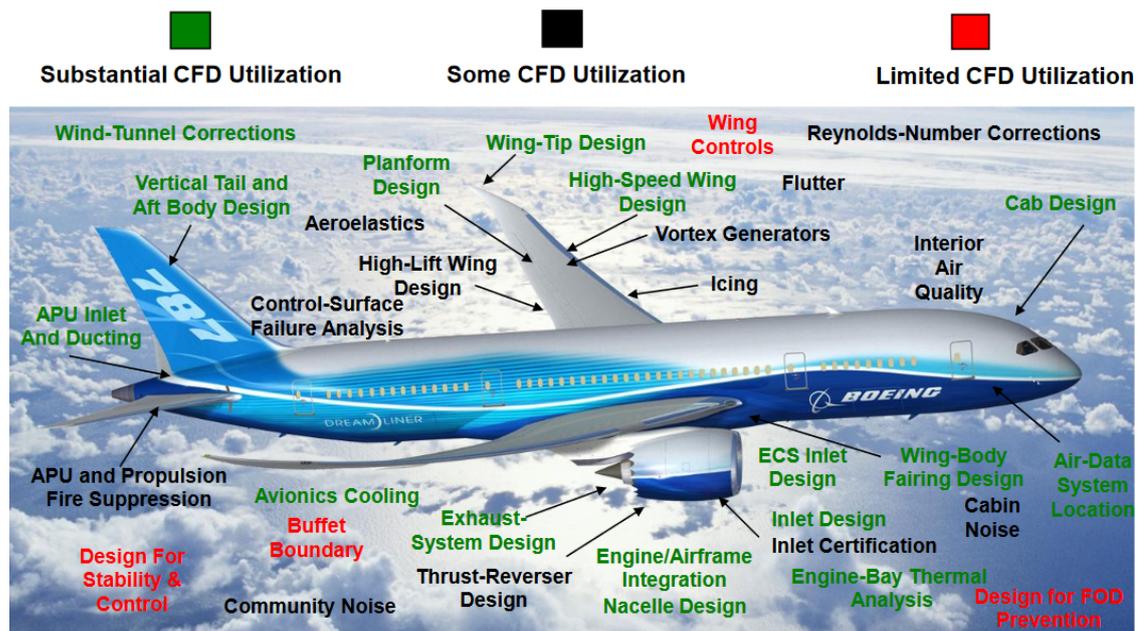


Figure 2.1: CFD contribution to Boeing 787 Aircraft Design.

Furthermore, the use of CFD with high-performance computer can allow a designer to obtain data in a time efficient way also for unconventional configuration and for all other situations in which limitations can occur. In that circumstances CFD provide a reliable source of data that restriction like model complexity, wind tunnel capability or availability and uncommon design constraints would make complicated to find through ordinary test. In addition, the analysis made with computational methods on complex or unconventional systems can give results that have the same accuracy typical of analytical theories applied on conventional configuration, in particular when non linearities occur. The consequence is that the continuous development of CFD has led not only to a reduction in design cost but has also given ad great contribution in the design of new platforms and systems. [2]

By using computers available nowadays, an Eulerian analysis for a complete aircraft configuration would last only few minutes or at most few hours, allowing to reduce cost and risk related to common test and increasing the performance of aerospace design.

2.1.2 Role of CFD during Aircraft Design Phases

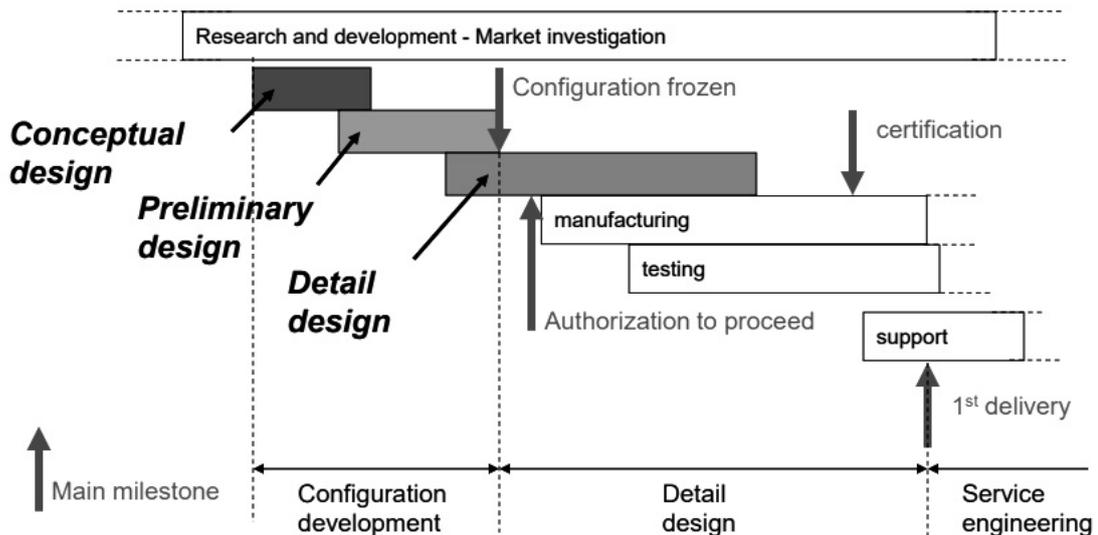


Figure 2.2: Aircraft design process layout.

The design process is typically structured in three main phases: conceptual design, preliminary design and detail design. This kind of breakdown is illustrated

in Fig. 2.2. CFD analyses can be developed during each one of these phases.

During the conceptual design, simplified models are typically used. In case of design of a not new configuration this kind of model could be obtained using data-morphing, modifying previously model used for similar application. Simplified models allow to save time for optimization process and at the same time enable to make accurate selection between drag, fuel consumption, weight, payload/range, thrust, or other performance measures. Sometime low fidelity CFD tool are used cause of the trade off between flow physics modeling accuracy and execution velocity.

When the product development program starts to be undertaken, analytical method become essential and widely used. Indeed, CFD is largely applied over the course of the preliminary and the detailed design. For commercial aircraft CFD is necessary for the design of cruise wing with wing-mounted nacelles. For military and commercial aircraft CFD is a reliable source of data for load distribution and ground effects. Other possible analysis regards the design of inlet and nozzle or the design of turbine engine components. Actually CFD is providing several data for aero and propulsion performance database. As said before, CFD become the only source of data in situation in which the possibility of develop ground test is limited or does not exist. Otherwise wind tunnel data are used only to give greater reliability and credit to CFD data.

The followings phases concern the validation and certification of product developed from the previous process. Here CFD is mostly used in order to validate the test results or to study a possible redesign of a component that could give improved performance. Computational methods can also be a useful tool used to answer all the possible questions that arise during the product testing. Indeed CFD is able to capture the required scope and physics and guide designer to the best choice for the new product configuration.

Much work has been done in the last two decades to reach a great improvement in accuracy and speed of CFD. The more these features will be enhanced the more will be possible for designer to move away form traditional design space

thanks to a greater confidence on numerical results and consequently a reduced probability to occur in problem during the tests.

2.2 Aerodynamic Optimization based on CFD

A traditional aircraft design requires that the choices to be made in order to achieve a particular design target are initially driven by similar project observation, intuition and experience. In this way, the design is strongly dependent on the experience of the designers who could restrict the design space. By consequence, it is not guarantee that the optimal design is achieved. On the other hand, considering a wing shape design process, a mathematical optimization technology can guide the designer to choose how to change the wing shape in order to obtain the required performance. In particular, high performance computing platforms and the actual development of CFD technology can made possible the use of numerical simulation since the early design phases. It is the reason why nowadays CFD analysis is not used only to predict fluid flow of interest but is exploited also in broader context such as optimization, multidisciplinary applications, automated analysis and design processes. By consequence, a CFD based aerodynamic optimization can be used to move in the wing shape design space and improve the aerodynamic performances.

2.2.1 Main features of Aerodynamic Optimization

An aerodynamic optimization requires the use of an optimization algorithms used to find the optimum in the design space. Typically, the control function (e.g. the wing shape), is parametrized with different design variable, additionally an objective function is defined. The involved equation and the constraints of the problem will express the link between the target function and the control function. After that an optimization technique is used to perform the search of the new design point, in which objective function have an improved value. As shown in Fig. 2.3 this procedure is repeated until the optimum is reached.

The ideal optimization process allows to include an high number of design variable and is able to find the optimum in an efficient way. An aerodynamic optimization

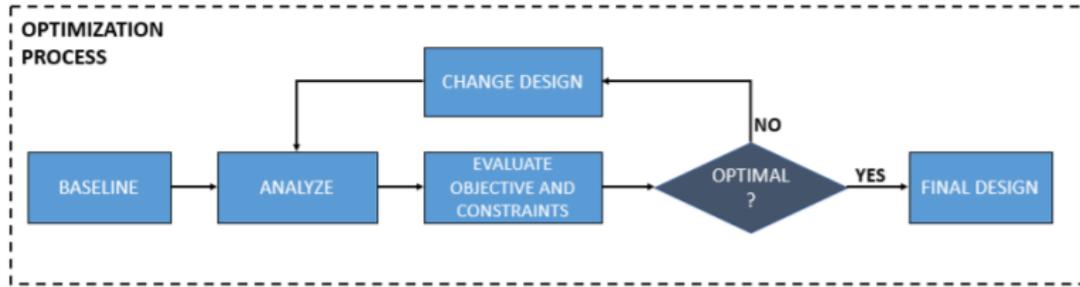


Figure 2.3: Optimization design process.

tool that use CFD is based on four main component: parameterization, optimizer, design variation generator and a CFD solver. The features of two components are described in the following [3].

- **Parameterization.** A Parameterization is tool whose job is to correlate the physical and mathematical domain. It provides a connection between design variables and the CFD computational grid. By this way the optimizer will be able to find the optimum in the mathematical domain changing the design variables.

A successful parameterization process must

1. Be automated.
2. Provide consistent geometry changes across all disciplines.
3. Provide sensitivity derivatives. The derivative of geometry with respect to design parameters should be computationally efficient and numerically accurate.
4. Fit into the product development cycle times.
5. Generate computational grids efficiently when the geometry is perturbed.
6. Generate a smooth geometry when the design variables are changed.
7. Produce a compact and effective set of design variables in order to make feasible the solution time.

Two different strategies can be followed. The grid under investigation can be parametrized and then can be deformed. Otherwise it is possible to generate a new grid for any change of a design variable. Obviously the first choice is time

efficient and does not need to perform an automatic generation of a grid; the negative aspect is that its use is limited to small geometry perturbation. On the other side, regenerate a grid during the optimization process it is a very time consuming method, especially in the situation in which the generation of a CFD grid takes a very long time. An overview of prevalent parameterization techniques is given in [4]. Some of these techniques are described in the following.

CAD approach are typically robust and time efficient, especially when the geometry deformation is large. The main difficulty occur in parametrization of existing models, unnoticed imperfection could cause problems for the grid generation. Moreover calculation of the analytical sensitivity derivatives of geometry with respect to the design variables could result in several difficulties within a commercial CAD environment.

The discrete approach use the coordinates of the grid points as design variables. The implementation of this method is very simple and it allows to avoid the regeneration of the mesh. The possible changes in the geometry are limited to the number of the points but the real negative aspect is that the shape could become less smooth and this can lead to difficulties during shape optimization. Sensitivity information are difficult to obtain by this method.

Polynomial and spline approaches can reduce the total number of design variables. In particular for two dimensional and simple three dimensional models a polynomial can describe a curve in a very compact form. In case of a complex three-dimensional problem, a great number of curves and surfaces are needed. This can lead to a great increase of number of control points and the generation of irregular or wavy geometry. The analytical sensitivities could be computed in a efficient and accurate way.

Another approach consists in using Hicks–Henne functions, a compact formulation able to parametrize airfoil sections. The approach is based on adding analytical function, which serves as a shape function, linearly to the baseline shape. The values of the functions coefficient are the design variable.

Analytical sensitivities could be computed without difficulties. This method is very effective for wing parameterization but it is not appropriate for complex geometry.

The FFD method consists in a parametrization of the possible change in the geometry instead of the geometry itself. It can provide a high-quality shape deformation considering a few values of design variables. The FFD formulation is independent of grid topology, this independence make it appropriated for different kind of analysis, from low to high fidelity tools. The analytical sensitivities can be computed and used for a gradient-based optimization. Usually the design variable composing the FFD have not physical meaning, nevertheless this method is widely and successfully implemented for aerodynamic shape optimization with analytical sensitivities for CFD codes.

- **Optimizer.** The optimizer is the tool that conducts the optimization process through the changes in the design variables. An ideal optimizer can handle a huge number of design variables with the least possible amount of CFD analysis. For each design point obtained with the optimizer, a CFD analysis is performed to evaluate the new value of the objective function. This value, together with the constrains, will let the optimizer decide if the optimum is achieved or not. If the design target is not satisfied, a new design point is given by the optimization algorithm. The process will be repeated until the optimum condition is satisfied.

Two different families of optimizer exist: gradient-based method and non gradient-based method. Gradient methods are most suitable for application in which the number of design variable is high. The main hypothesis of this method is that the objective function and constrains are smooth functions, this usually happens in aeronautical application. Obviously a gradient calculation technique is required, after using it a search method can be used to perform the research of the new design point. The primary negative aspect is the computational cost of a gradient calculation process. Different methods exists for sensitivity analysis: finite difference, complex step method, adjoint method,

and so on. At present, gradient based optimization in combination with adjoint method provides the most efficient way to solve CFD based aerodynamic optimization with a large number of design variable.

The majority of gradient based method shows some problems when the objective function or the constraints are discontinuous, non-differentiable or are more than one. The same could happen if the multiple local minima occurs, or if the design variable are discrete. When the case study have similar features, non-gradients methods are developed. The most well known gradient-free methods are Simulated Annealing, Divided Rectangles Method, Genetic Algorithms and Particle Swarm Optimization.

As can be noticed, before running an optimization process a lot of preliminary analysis should be done. Pros and cons of of the possible methodologies should be examined taking into account the number of variable and the computational time of calculation. The best way to reduce the optimization time is develop or take advantage of new optimization process, choose the appropriate solver between high and low fidelity methodology for each task and decompose the problem in several sub-problem. Actually, nowadays the calculation from each sub-system are not performed in series but the process is divided in different sub-process that can work in parallel as illustrated in Fig. 2.4 [5].

2.3 Multidisciplinary Design Analysis and Optimization

2.3.1 Collaborative Design

Nowadays, requirements coming form market competitiveness impose increasing speed, range and number of passenger at a same time under secure conditions and with environmentally friendly solutions. To achieve these targets is fundamental to take into account the interaction between several disciplines, by consequence an heterogeneous team of specialist is needed in order to provide a strong cooperation during aircraft design process. Take into consideration all involved areas during conceptual and preliminary design phases, as described in the previous sections, could

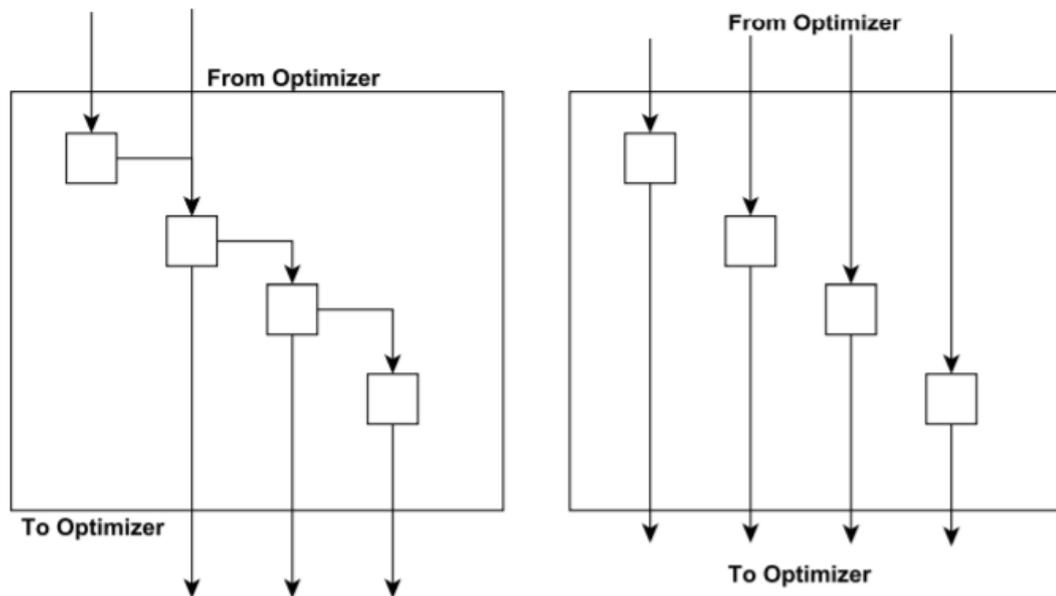


Figure 2.4: Possible problem decomposition: in series or in parallel.

typically lead to elaborate and time-consuming process. In addition, collaboration among experts could bring out several barriers such as intellectual property, resources protection, licensing and security policies about data exchanging. To deal with all this kind of obstacles, a well organized collaborative design process must be set up. This kind of approach ensures that the product is designed by experts from different backgrounds, able to give their contribution regarding the area in which they are specialized.

A typical collaborative design structure is illustrated in Fig. 2.5. As can be observed, each topic is linked with many other, including issues that belong to different design area, and so to different field of expertise. Taking as an example the design of a commercial jet, millions of design issues could be considered, hundred to thousands of expertise could be involved and hundred of design subspace could be taken into account [6]. To deal with such a big and widespread design structure, the design tasks must be distributed among different teams. Each team has the task to create tools correlated with they area of interest, such as aerodynamics, structures, costs, propulsion and control theory. In addition, each team must have the possibility to use the tools generated by other teams to obtain data needed

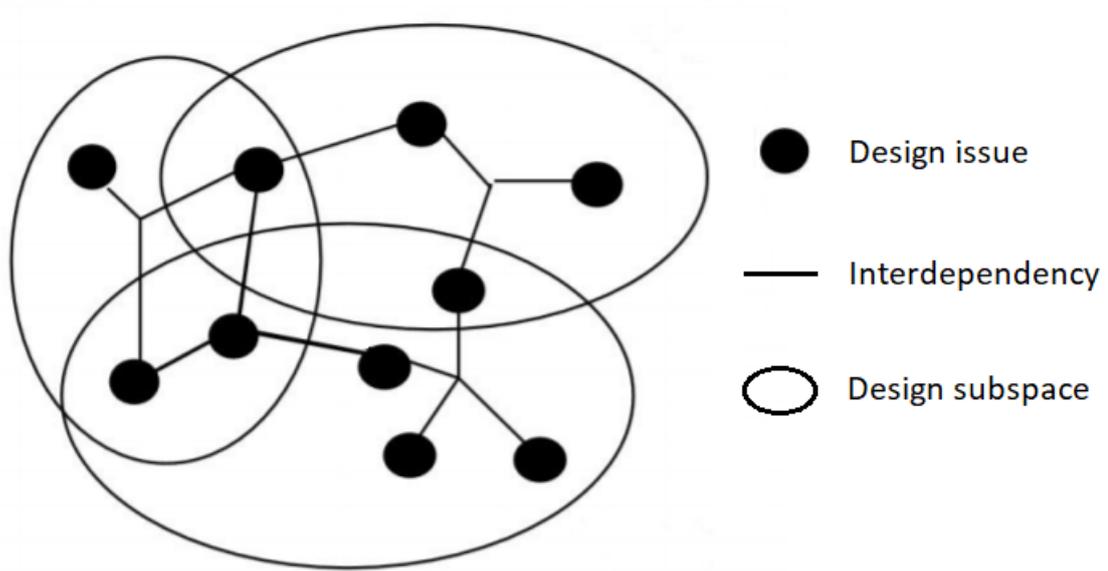


Figure 2.5: Collaborative design structure.

for their own tool. This kind of organization is distinctive of a collaborative and distributed design. Collaborative because everyone can exploit the products of other people works, distributed because each task is assigned to an individual team. The modern evolution of this way of working is the Collaborative Remote Design. It allows people from different locations to communicate and exchange their tools, experience, method and results through server connection.

This approach is the basis for Multidisciplinary Design applications.

2.3.2 Multidisciplinary Design

Overall aircraft design is multidisciplinary by nature, this means that the results of the design is strongly dependent not only on the performance of each individual discipline but also on their interactions. Multidisciplinary design analysis and optimization techniques provides instruments to examine and understand the coupling and dependencies among disciplines that can influence the development of a new product. Results coming from MDAO problems usually improve the final design and reduce time and cost of the design cycle. In particular, it is important to develop high-fidelity methods in MDAO when unconventional configuration need

to be considered and so when the designer requires to trust on the results.

Actual MDO is characterized by two main features.

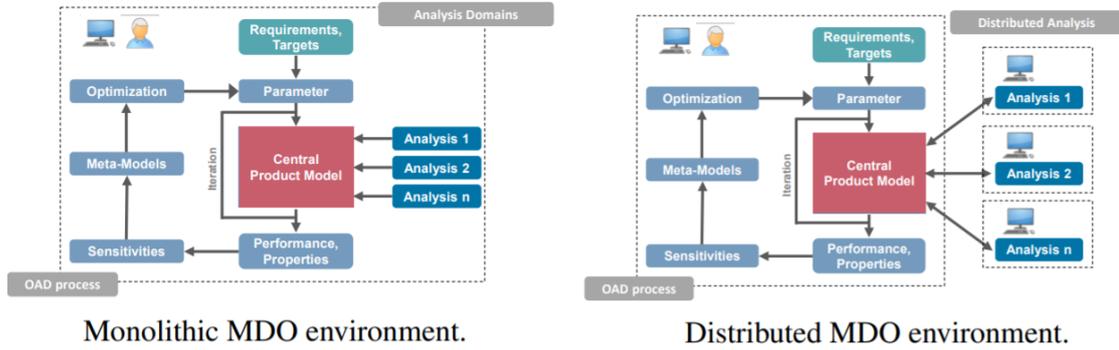
1. It is composed by an elaborate combination of tool and method that give birth to a sophisticated process that is impossible to automatize.
2. The change in one or more design variables in a subsystem will affect the results from all the system. This feature is fundamental if the aim is to reduce the design time.

An innovative method consists in the research of the optimum by optimizing each field sequentially and highlighting the interaction between teams. The optimum of simultaneous problems must be most significant than the optimum of each discipline. This methodology is extremely useful in that kind of design that includes numerous variables and a multitude of analyses and disciplines.

2.3.3 MDO architecture

A crucial aspect, that need to be taken into account during the early stage of the implementation of an MDO system is the way in which the disciplinary analysis models are organized in order to achieve an optimal design with the use of an optimization software and a proper problem formulation. This is referenced as "MDO architecture" [7]. Therefore the MDO architecture defines the coupling of different models and how the optimization problem that involves them is solved. The choice of MDO architecture depends on the design problem that must be faced and the resources available to the designer. According to the way the optimization problem is decomposed, MDO architectures can be classified as monolithic or distributed architectures. Fig. 2.6 shows the comparison between the two different architectures.

- **Monolithic architecture.** The Monolithic architecture refers to application which integrates different disciplines and the optimizer in a single coupled system. In this environment all the analysis tool are directly available to the design team lead and the multidisciplinary system can be studied with a single disciplinary optimization. The objective functions, the design variables and



Monolithic MDO environment.

Distributed MDO environment.

Figure 2.6: Comparison between monolithic and distributed architecture.

constraints had to be assigned by the optimizer. Typically coupling techniques are exploited in order to obtain the sensitivity of the overall system with respect to the design variable of interest. The monolithic approach is computationally efficient, in particular when optimization problems with strongly coupled disciplines occur, moreover the design lead has easily access to design variable and constraints. By contrast, if the coupling between disciplines is not strong this approach could be pointlessly onerous. Meanwhile, if the optimization problem is too complicated, it could be difficult to manage a big number of design variables and constraints; this happens because of the presence of a one single optimization problem. Other negative aspects regards the low flexibility to modify or update a subset of the overall system and the limitation in the scalability of the system. As more disciplines are considered for the generation of the optimization system, the integration of an additional single subsystem becomes unfeasible. Nowadays, monolithic architectures are typically set in two main contexts.

1. During the conceptual design phase, in which the design need to be fast and the coupling effect are investigated using simple models.
 2. For detailed optimization in which a few number of disciplines are involved and the coupling effects are really strong. An example is aero-structural optimization based on adjoint formulation.
- **Distributed architecture.** As described in the previous sections, an increasing level of accuracy of the analysis model is required nowadays. By

consequence, in order to handle a more sophisticated analysis tool, an increasing number of expertise are necessary to run correctly an optimization. In case of multidisciplinary analysis, it is highly possible that high-fidelity capabilities are distributed among different structures. In case like this, a distributed architecture is necessary. What distinguishes this kind of architecture is the distribution of the analysis tool among different facilities. These tools are controlled by a centralized design and optimization process, as shown in Fig. 2.6. In this way, the optimization problem is partitioned in different sub-problems containing their own variables and constraints. Such a system possess a great flexibility, indeed it possible to modify or update a single module and assemble the whole system in a easy way. It also possible to organize each computing facilities in the best way to fit with the requirements of each discipline. Designated experts are responsible for their disciplinary module, can work in isolation and can easily control their set of variable and constraints. The design lead play the only role of process integrator and central optimizer. The distributed approach allows to avoid unnecessary data exchange during the optimization process. On the other hand, the design modules need to be connected among each other and the centralized optimization component. This is the reason why a data management system and an interface between different modules is necessary.

As stated above, especially for distributed architecture, an integration framework is required. Its task consist in manage data exchange among modules and handle the optimization process, e. g. monitoring the convergence of the workflow. At present, different engineering frameworks, such as Remote Component Environment (RCE) [8], have been developed to coordinate the analysis modules. Another really helpful facility could be a centralized data format able to make easier the exchange of data among modules and to reduce the number possible interfaces. As example, in the following section CPACS standard format will be described. In addition, since the analysis tool are directly controlled by the team lead, the execution of these analysis modules must be automated and robust. In conclusion, if the overall

system includes high-fidelity analysis tool, a proper MDO strategy should be chosen, in order to avoid excessively long computation time.

2.3.4 CPACS standard format

The Common Parametric Aircraft Configuration Schema (CPACS) [9] is a centralized data structure developed in the German Aerospace Center (DLR) since 2005. It is a data format based on Extensible Markup Language (XML) technologies, in this way any kind of data is positioned below a specific tag that compose a typical XML tree in which the root element is called '/cpacs'. CPACS contains a parametric description of one or more aircraft configurations including data such as geometry, mission, airport, engine performance, landing gear and other, as illustrated in Fig. 2.7.

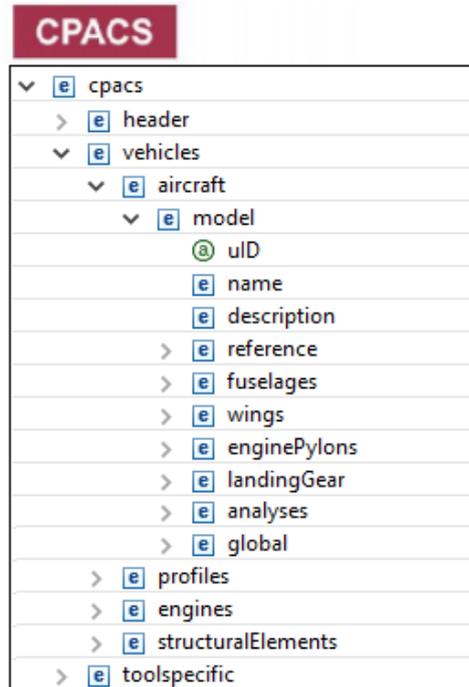


Figure 2.7: CPACS schema.

Other possible data can be used to initialize and drive the disciplinary analyses. In particular, a CPACS file contains [10]

1. Only 'exchangeable' information.
2. No redundant information.

3. Product information (Aircraft, airport, mission).

4. Tool-specific information (Option, run-time information, tools results).

CPACS can take advantage of TiGL [11] to obtain data from CPACS-described geometry that can be useful for the disciplinary analysis modules. TiGL geometry library is also able to represent the geometry of aircraft components using B-spline surfaces, it can also export this geometry to a CAD based format.

With the use of CPACS as a centralized data format, the consistency among different analysis modules and also with different fidelity level, is guaranteed. Fig. 2.8 represents the CPACS concept of a unique common data structure used for different disciplinary analysis tools.

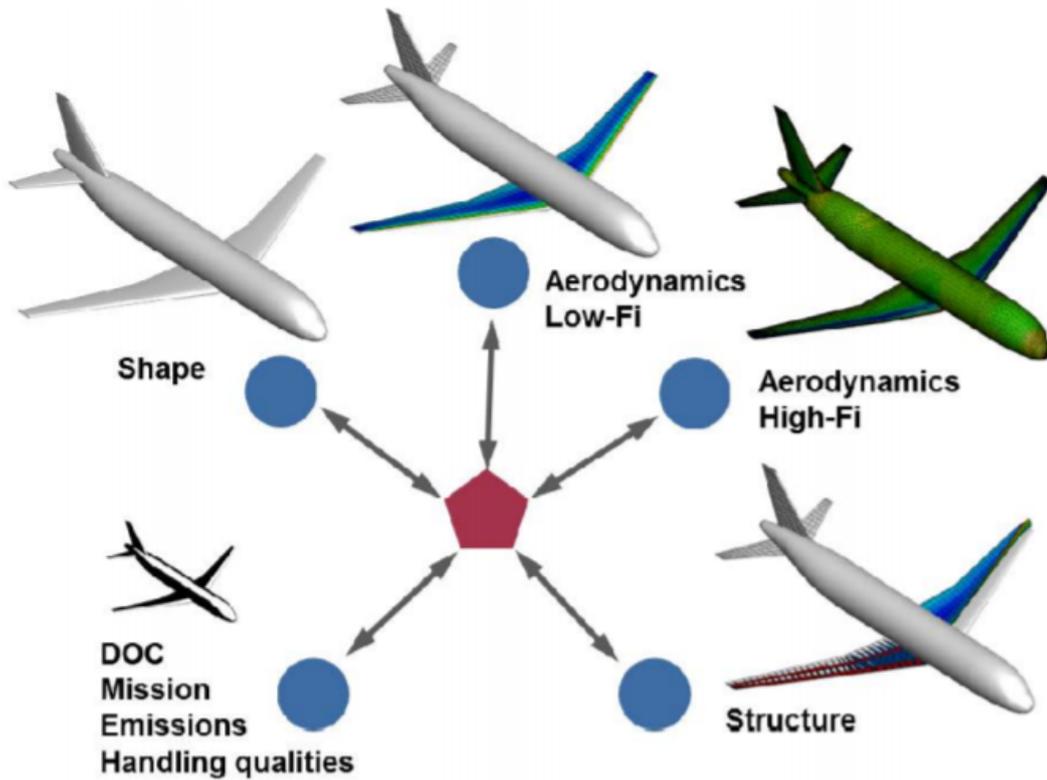


Figure 2.8: CPACS concept.

2.3.5 Requirements for MDO

In order to reach an optimal design, the choices regarding objective functions, constraints, design variable and MDO architecture are fundamental. A list of

requirements useful to achieve the most efficient compromise between the after mentioned decisions is now presented [12].

Overall system architecture

1. The structure of the system should be flexible and adaptable to different design cases, needs and design phases. A way to achieve it is using a loosely coupled modular structure.
2. The system should allow designer to integrate both internal and external design, analysis and optimization tool, data sharing and communication systems.
3. The system should allow automation of all repetitive activities characteristics of a MDO approach which requires, by nature, several iteration. Pre and post-processing of data required and generated by design and analysis tools should be provided, as well as data transfer and storage.
4. The system should employ dedicated software frameworks for the integration of analysis and design tools. The integration framework can support coordination and communication between different tools.

Analysis capability

1. The system should not have limitation on the number of tools and disciplines which can be integrated.
2. The system should allow the use of different fidelity level tools. If possible, it should allow to switch between different level; an ideal system should do it automatically according to the results of specific sensitivity analysis.

Geometry modeling

1. Also the geometry model should give the possibility to use low and high fidelity analysis tool.
2. The geometry model shall not force or lead the designer to direct the choice on conventional configuration.
3. The geometry model should be robust and automatic in order to adapt without problem to the MDO framework.

4. The parametrized geometry should be suitable with the in use CAD system and it should be possible to transfer geometry description data in a easy way among different tools.
5. It should be possible to display all the design space and not only the the single optimum point, in this way the designer can judge the robustness and the sensitivity of the actually reached design point.

2.4 Aero-Structural Analysis and Optimization for Aircraft Design

Traditionally, aero-structural analysis have been developed through an attempt based approach. Since the beginning, a designer has a general idea of the optimum load distribution. Then he tries to obtain a wing structure and shape which best fits that load distribution. This kind of approach could be considered correct only for conventional configuration, for which there is an abundant set of pre-existing case and there is a great experience. An other possible approach regards the use of statistics for the structural weight estimation, unfortunately this implies that practically all the air-frame information are absent during the conceptual design phase. This choice brings to an exclusion of aero-elastic requirements and considerations that indeed are considered subsequently during the design loop.

Nowadays, Advanced materials with improved strength and stiffness may enables the use of slender and with higher aspect ratio wings that are aerodynamically favorable, but exhibit an increased flexibility rather than conventional wings. Therefore, it is important to take into consideration aero-elastic effects even at an early stage of the design process, avoiding time-consuming redesign during the preliminary design phase and the use of weight penalties necessary to satisfy aero-elastic requirements not taken into account from the beginning. In case of new aircraft concept or not usual flight regimes the above described approaches, because of the lack of experience and the complex aero-structural interaction, can bring to designs that are far from optimal. A typical example regards the design of supersonic aircraft. In such a case, simple models can not be used to correctly describe the behaviour

of wing structures. Moreover, this aircraft usually cruise at different Mach numbers during the missions and they typically exhibit undesirable aero-elastic phenomena because of the low bending and torsional stiffness characteristic of low thickness-to-chord-ratio wings. All these phenomena can be avoided only taking into account aero-structural interaction at beginning design stage.

2.4.1 Importance of Integrated Aero-Structural Analysis and Optimization

During the years following the development of high-lift methodology in both structures and aerodynamic, a large number of numerical optimization has been applied to these two disciplines separately. On the structural side, ever more detailed finite-element analysis has been performed for wing structural optimization. This leads to an improved sizing of the structure that takes into account complex structural failure constrains and enables a greater estimation of the structural weight. In the same way, as explained in 2.2, at present is possible to obtain optimized design with respect to hundred of variables through CFD. Nevertheless, aerodynamic wing optimization alone is not enough for aircraft design. It is important to take into account the coupling effects of these two disciplines to perform the trade-off between possible design variables such as wing thickness, span and sweep. Indeed, these variables influence in different way the aerodynamic and the structural behaviour of the wing. For instance, changes in span load could lead to a reduction of induced drag but they can also penalize the structure and bring to an increase of wing weight. In the same manner an increase in the thickness-to-chord-ratio of wing sections will improve the efficiency of the structure but it may lead to an undesirable increase in wave drag. These are the reasons why aerodynamic design of wings and their internal structure have been subject of MDO studies for several time. Coupling aerodynamic and structural numerical models is essential to compute the static aero-elastic shape of aircraft flexible wings. Small changes in shape can generate a huge effect on aerodynamic performance, moreover a different shape will be reached for each flight condition. This is particularly relevant for swept wings,

in which coupling of bending and twisting effect could generate a great change in twist deformation. In addition, in recent years aero-structural coupling has become even more relevant due to the increasing trend of the wings aspect ratio which makes them more flexible. Wing flexibility results not only in static shape of the wing but also in dynamic shape, giving rise to aero-elastic phenomena such as flutter and aileron reversal. A practical example of the impact that coupled aerodynamic and structural study has on wings design is now presented.

A well-known result from subsonic aerodynamics is that the maximum efficiency for a wing is reached in presence of elliptical lift distribution. Although, aircraft designer have the necessity to optimize a more general function that reflects the overall mission of the aircraft. A possible objective function could be the maximum range of the aircraft. Eq. 2.1 represents the Breguet range formula for jet-powered aircraft.

$$Range = \frac{V}{c} \frac{C_L}{C_D} \ln \frac{W_i}{W_f} \quad (2.1)$$

V indicates the aircraft speed, c is the thrust-specific fuel consumption of the power-plant, C_L/C_D is the ratio between lift and drag, and W_i/W_f is the ratio between the initial and final fuel of the aircraft. The Breguet range equation shows the correlation between aircraft efficiency and empty weight, this is exactly why is a reasonable objective function in aircraft design. Martins [13] showed that an aero-structural optimization which aims to maximize the range for a fixed cruise weight results in the span-wise lift distribution represented in Fig. 2.9. Obviously, the optimization is performed thanks to a parameterization of the aircraft made with both aerodynamic and structural variables.

The aero-structural optimum lift distribution represents a trade-off between minimization of the drag and reduction of the loading on the wing tip. The loading in that area generates the higher contribute to the stress at the root of the wing structure, which brings to an increase of the weight of the wing. Through this lift distribution, it is possible to obtain an increase of the aircraft range, in particular if it is compared to the elliptical distribution which would bring to an higher weight ratio. Fig. 2.10 shows in a simplified manner the approach followed to obtain the

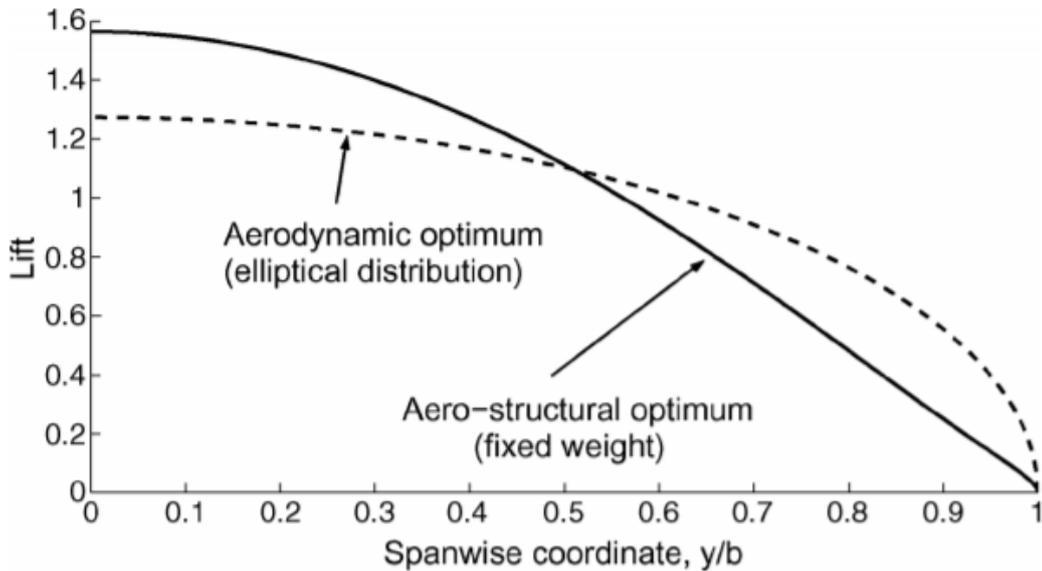


Figure 2.9: Comparison between elliptical and aero-structural optimum lift distribution.

introduced optimization. It is important to underline that the structural design and the aerodynamic shape are optimized concurrently so that the optimizer can make use of the interplay between these two closely related aspects of the design.

2.4.2 Current High-Fidelity Method Applications

Coupled aero-structural design optimization has reached a high academic level of development and now it is also widely utilized in enterprises. Complex multi-fidelity optimization frameworks have been developed to satisfy requirements of different phases of aircraft design. Typical applications include detailed wing optimization, winglet design, supersonic aircraft design and many other.

Optimization developed using low-fidelity models can help to understand the complex interaction between aerodynamic and structure. Due to their computational efficiency these methods are widely used during conceptual design to try to reach the best solution space rapidly. Several limitations do not allow the possibility to take advantage of high-fidelity methods in this phase. CFD methods require volume mesh, which are difficult to generate rapidly. In addition, mesh deformation strategies are

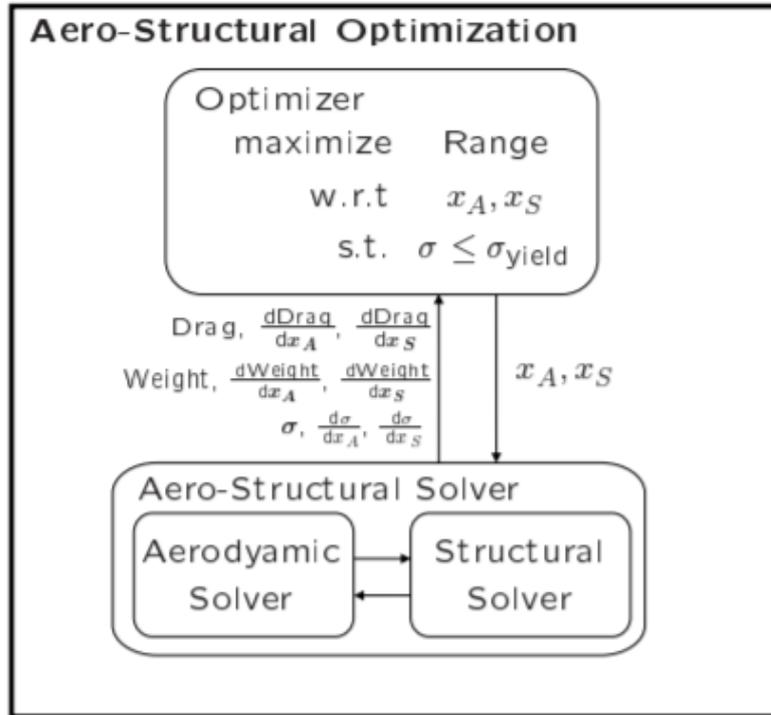


Figure 2.10: Integrated aero-structural optimization

required for optimizations that involve large scale geometry modifications. Moreover, it is necessary to transfer forces from the aerodynamic surface to the structure. These two extrapolation can be extremely difficult if there are large gaps between the aerodynamic and structural models. These requirements substantially increases the complexity of the implementation of aero-structural analysis and optimizations. This is the reason why optimization based on high-fidelity models are usually developed during detail design phase to enhance the optimization quality further. High-fidelity methods include compressible Euler or Navier Stokes equation and structural finite element method. Nowadays, the principal task is to increase the optimization efficiency, different strategies have been adopted, the most significant are illustrated in the following [14].

1. Tightly coupled solving strategy can be followed to solve the static aero-elastic problem, which can reduce the computational expense. As can be

easily figured, loosely coupled aero-elastic analyses are really simple. They usually solve sequentially the aerodynamic and the structural problem until the deformation of the wing converges. Usually not many iterations are needed to obtain the results, but a tightly coupled analysis method can be much more efficient. The flow-solid system consists of structure and flow sub-problems and they are successively solved at each time step. This kind of strategy is usually complex to implement and this is the reason why it is rarely found in commercial software packages.

2. Optimization frameworks based on various kinds of surrogate model can be developed to reduce the computational expense.
3. Gradient based optimization is used to improve the optimization efficiency. The main improvement could be achieved with an efficient technique used to compute the gradient of the objective function. Usually, this is realized through adjoint method, in which the computational effort does not depend on the number of design variables. Nevertheless, the derivation of adjoint equations in aero-structural design optimization is very complex and the convergence of the adjoint equations still needs to be improved.
4. GPGPU is adopted to speed up the optimization. Different applications have been done in computational fluid dynamics, resulting in an acceleration of twenty times compared with conventional computation based on CPU. Unfortunately, no researches have been done in coupled CFD/CSD design optimization up to now.

One of the earliest examples of aero-structural optimization was by Haftka [15] who combined a lifting-line aerodynamic model with a simple structural finite element analysis to obtain, with an iteration method, the flight shape deformation. He performed the simulation for both aluminium and composite configurations. Grossman et al. [16], compared aero-structural design optimization in which the aerodynamic and structural variables were optimized in sequence, to an integrated approach in which all design variables were optimized at one time. They showed that the integrated approach produces a design with better aerodynamic and structural

performance, and that the integrated design shows better aero-structural interactions. They successively performed an aero-structural optimization of a wing for minimum structural weight subject to a fixed range requirement [17]. Maute et al. [18] were the first who performed an high-fidelity aero-structural design optimization, they coupled a linear finite-element solver to an Euler CFD solver. Although, this application was limited to a small numbers of design variables since the authors employed the direct method. To enable high-fidelity aero-structural optimization with respect to large numbers of design variables, Martins et al. [19] proposed the use of a coupled adjoint method for aero-structural design optimization using Euler CFD and linear finite-element analysis in a two-field formulation. They showed that that the cost of computing gradients using this method could be made nearly independent of the number of design variables, indeed they were able to compute gradients with respect to an high number of variables. Then, they applied this method to the aero-structural design of a supersonic business jet with respect to 97 shape and sizing variables [20]. Maute et al. [21] also developed a coupled adjoint implementation for high-fidelity static aero-elastic simulations. In order to develop more robust aero-structural solution strategies, Barcelos et al. [22] proposed a Newton–Krylov–Schur method to solve coupled aero-structural problems. Later, Barcelos and Maute [23] developed a coupled adjoint method for the turbulent Reynolds averaged Navier–Stokes equations. More recently, Kenway et al. [24] developed a coupled adjoint for high-fidelity aero-structural optimization using the ADjoint approach (Mader et al. [25]). This framework has since been applied to the design of large transport aircraft.

3

Instruments for Aerodynamic and Structural Analysis

Contents

3.1	Introduction	29
3.2	Automatic Mesh Generation process	30
3.2.1	Geometry Generation Module	31
3.2.2	Mesh Generation Modules	32
3.3	Stanford University Unstructured	34
3.3.1	Software architecture	34
3.3.2	Input and Output	36
3.3.3	Physical Modeling Equations	38
3.3.4	Adjoint Formulation and Sensitivity Computation	39
3.4	Automatic CSD analysis process	43
3.4.1	Structural tool LAGRANGE	44
3.4.2	Structural Process Architecture	45

3.1 Introduction

In this chapter the already existing tools subsequently used are presented. In order to perform an automatic aerodynamic analysis, it is fundamental to be in possession of a flexible and robust tool for mesh generation which is able to produce a CFD grid from already existing input geometries. This tool is integrated in RCE with python based codes. The geometry is read from a CPACS file and an unstructured mesh in SU2 format is generated. Different kinds of geometries are already tested to verify the performance of this tool. Other than classical configurations, blended wing body, strut-braced wing aircraft and canard configuration were tested successfully [26]. Using the mesh obtained with the just presented tool, it is possible to perform a CFD

analysis through SU2, an open source collection of software tools able to discretize and solve problems described by partial differential equation. Both these tools are presented in the following sections. In addition, a structural solver too is presented to allow to possibility to couple aerodynamic and structural design.

3.2 Automatic Mesh Generation process

The mesh generation process is based on CPACS centralized data format described in 2.3.4 and it is integrated in RCE, a workflow-driven environment developed by the DLR. The overall process is composed by Python-based modules which generate the computational mesh that can be used for automatic viscous or inviscid analysis.

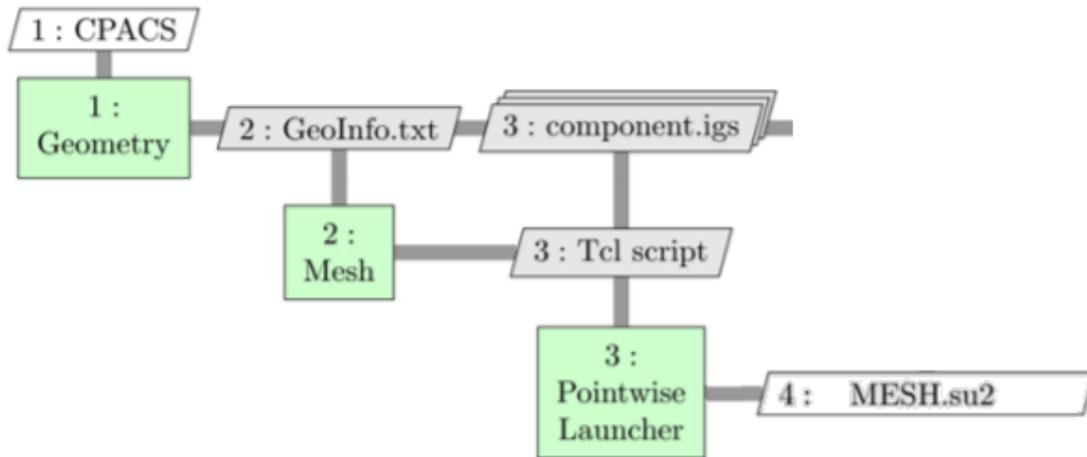


Figure 3.1: XDSM graph of the automated mesh generation process.

As shown in Fig. 3.1, three modules compose the process. The "Geometry" module generates a CAD object from a CPACS file using the TiGL library. Once the geometry is defined, a Pointwise glyph-script [27] is generated thanks to "Mesh Script" module, this script is successively executed using "Pointwise Launcher" module to generate the computational mesh. A brief description of each modules is now presented, in [26] it is possible to find a more detailed presentation.

Parameter	Description
Simulated_component	Specify the uID of each component that need to be involved in the mesh
SimulatedWingType	Specify the type of the wing insered in Simulated_component
Body1_HasRootInside_Body2	Specify the root connections through uID
Body2_HasTipInside_Body2	Specify the tip connections through uID

Table 3.1: Input parameters for Geometry Tool.

3.2.1 Geometry Generation Module

This module consists in a python script whose purpose is to generate a CAD model from the CPACS file. In Tab. 3.1 are summarized the input parameters that have to be specified in the toolspecific node in the CPACS file.

The uID of each component which have to be included in the grid is indicated using "Simulated_component". "SimulatedWingType" parameter is used as a retroactive control during the clean-up process that is performed during the generation of the mesh (Par. 3.2.2). "Body1 HasRootInside Body2" and "Body1 HasTiptInside Body2" are inserted to specify the wing connections. These two parameter are necessary to recognize the connectors which are the boundary curves of wing surfaces.

The module reads the CPACS file using TiXi library and initializes the aircraft instance of the TiGL library. In this way, it identifies the geometry topology, such as the numbers of wings and fuselages or the mean aerodynamic chord and fuselage max radius. For wings, the dimensional information such as leading edge length and trailing edge length and thickness is computed to support the meshing process later on. For the fuselage, the revolution starting point is computed to support the clean-up process. Then the complete CPACS model is decomposed into separated CPACS files, whose contain single aircraft component. After the creation of individual CPACS files, the CAD files are generated. After that, the module saves size and topology information in a dictionary subclass using Python

Parameter	Description
GlobScale	Grid factor which controls the mesh size
TotalModel	Type of external domain (Hemisphere or Sphere)
Physical_Problem	Specify the equation to solve (EULER or RANS equations)
Wall Spacing	Height of the first mesh cell off the wall
n_BL_layers	Number of an-isotropic layers for boundary layer

Table 3.2: Input parameters for Geometry Tool.

OrderedDict which save it as a text file. This file, named "GeoInfo.txt" is used as input for the following module, as shown in Fig. 3.1.

3.2.2 Mesh Generation Modules

The generation of the mesh is obtained through two Python-based modules, named "Mesh Script" and "Pointwise Launcher". The first one uses as input the IGES file and the information on components dimensions and topology such as the root and the tip connections or the component type. With these information it is able to automatically write a TCL-macros for Pointwise. The following module executes the macros in Pointwise, generating the grid. The tool produces isotropic tetrahedral meshes for inviscid flow simulations, and tetrahedral/prism meshes for viscous flow simulations. Many other parameters are also necessary to configure mesh generation, indeed they are calculated automatically during "Mesh script" module execution.

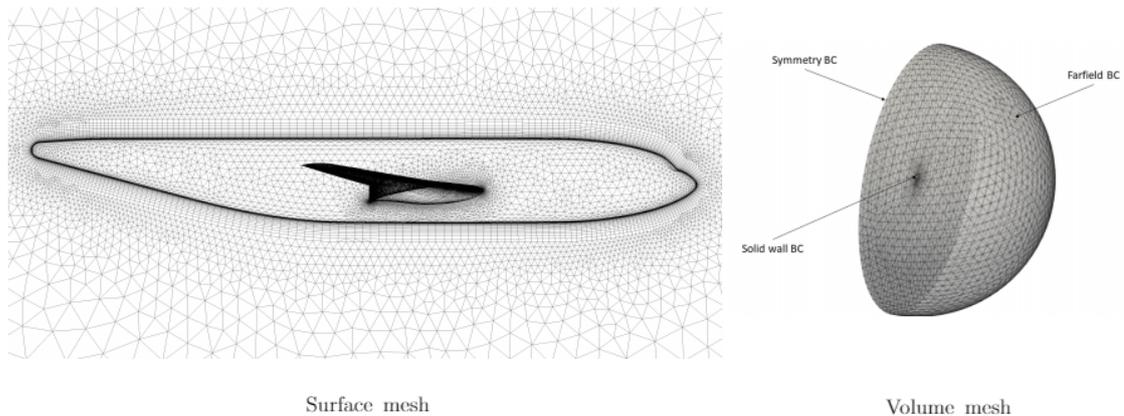


Figure 3.2: Example of surface and volume mesh for viscous analysis on wing body configuration [26]

The mesh generation process can be divided in 8 steps. A list of the main task of each step is now presented. A more detailed description can be found in [26].

1. Clean-up the geometry and to generate a fully watertight shape. The surfaces of each component are merged and a watertight geometry for each basic aircraft component is generated. An half-model of the aircraft components that are located on the symmetry plane is created.
2. The aircraft component are trimmed through a Pointwise Boolean function. Here it fundamental to know the components connections. From this point the aircraft is considered as an unique geometry model and the components appear as groups of surfaces which represent the mesh topological regions.
3. With an uniform and coarse mesh, the connectors at crucial positions are identified using a series of Pointwise procedure in order to customize and to parameterize the mesh. As instance, the connectors which are shared by both the fuselage domain and the wing surface domain are identified as "root connectors".
4. The surface mesh is improved by applying a refinement function on the connectors previously identified. The implemented refinement function takes into account the position of the connectors in order to apply a different sizing procedure.
5. The connector that are not defined during step 3 are resized. For instance, this could happens when two wings are not completely connected each other.
6. Leading and tearing edge boundaries are solved with an-isotropic triangles. This is done to provide a better control on the grid dimension for the geometry regions characterized by an high curvature such as wing trailing and leading edge.
7. A far field domain is generated according to far field length and far field surface triangle average edge length that are automatically calculated. An hemisphere domain is created at this meshing step.
8. The volume mesh is generated. For viscous applications, an-isotropic tetrahedral cells are used to simulate the boundary layer. The rest of control volume

is filled with isotropic tetrahedral cells. For inviscid applications, isotropic tetrahedral cells are generated to fill the control volume. Finally, the boundary conditions are applied and the computational grid is exported in SU2 format. An example of mesh for viscous analysis obtained from a wing body configuration is showed in Fig. 3.2.

3.3 Stanford University Unstructured

3.3.1 Software architecture

Multidisciplinary problems requires the interaction between different physical problems, usually represented by separate procedures. In order to avoid efficiency and integration problem, the modules should share the code architecture and a common set of solution algorithms. There might also be needed an infrastructure for shape design and an adjoint equations solver. SU2 is created with the purpose of make such an environment available and with a flexible architecture. Indeed, the SU2 framework is described in [28] as model with the following features.

- Portability. SU2 is able to be run on any computing platform with a C++ compile available.
- Re-usability and encapsulation. The main modules of SU2 are built in order to enable re-usability and modification of code without affecting other modules in the the suite.
- Performance. Numerical solution algorithms are exploited in order to reach and high performance in convergence of the solver.
- Gradient availability. SU2 provides adjoint solver useful to compute gradients. As matter of fact, several applications requires the possibility to calculate gradient of the computed response to variation of the largest amount of design variable.
- Open source. The choice to make it open source is made in order to allow the community to give their contribution in future development of the suite.

The seven C++ modules which make up the suite are able to perform a wide range of analysis related to PDE, such as grid adaptation and deformation, surface definition, optimization, gradient projection and CFD resolution. These modules can be executed individually or in coupling manner, in this way is possible to perform a complex activity like a design optimization. Python scripts are also provided to couple SU2 modules.

The architecture of the software can be divided in different levels of components. In order to give an explanation of how the modules are organized, the levels of optimization control architecture, respectively from the lowest to the higher, are now presented .

1. Core tools. This level contains all the binary executable code, which are the core of the suite. From input configuration files in ASCII format they generate, as output, file with results for each step of the iteration history, field data file organized in order to be plotted or deformed meshes in SU2 format.
2. Solution decomposition/composition. Different core tools can operate in parallel on a partitioned mesh. For this reason python codes are provided to perform pre-process mesh decomposition and post-process plot file merging of solution data.
3. Sensitivity analysis. The tools which compose this level perform pre and post-processing for sensitivities calculation with respect to specified design variables. As instance, in case of adjoint approach both direct and adjoint solution are obtained and then, the adjoint surface sensitivities have to be projected into the design variables through a post-process step.
4. Design evaluation. SU2 provides a design management class that make use of the previous described components only taking into account design vectors as input. It performs mesh deformation, direct solver, sensitivity analysis and returns solution data as output. Restart and plot data are saved into an organized folder structure which is useful for restart solution, make additional analysis or for debugging.

5. Design optimization. This is the highest level of architecture available. Two optimization methodologies can be performed. Gradient based optimization using SciPy's SLSQP optimizer or surrogate based optimization where a Gaussian process regression based optimizer is used.

3.3.2 Input and Output

SU2 routines requires as input only two fundamental file. The configuration file that is a simple text file in which run settings are provided and obviously the mesh file. Different output are provided. Output data are saved in specific format file in order to easily visualize or furtherly analyze the solution.

Configuration File. The configuration file is a text file with .cgf extension in which the options for SU2 are set. The syntax required to comply the insertion of an option is simple, the name of the option must be followed by the symbol “=” and then by the option value. This value can be a scalar data, a vector, a text or a more complicated structure. A comment line will start with the symbol “%” and white line are not taken into consideration. For a deeper explanation, Tab. 3.3 illustrates the main possible inputs needed to start an Eulerian compressible computation with SU2 Falcon [29].

Mesh File Native Format. SU2 provides and requires also a native mesh format, its extension is .su2. This file supports only unstructured mesh, by consequence information about node location and connectivity are both required. In addition also the types of elements and the nodes that compose each element have to be specified. Lastly, the boundaries of the mesh, named markers, and their elements connectivity need to be indicated.

The first line of the mesh file indicate if the grid is in two or three dimension. SU2 supports both type of meshes. The following lines describe the elements connectivity. For each element it has to be indicated the element type and a list of the nodes that composes the element. After connectivity information, for each node must be provided the coordinates in Cartesian space. The final section of the mesh describes all the boundaries. Their name, the node and elements that make up the boundaries

Parameter	Description
PHYSICAL_PROBLEM	Set the physical governing equations: Euler or Navier Stokes equations
MATH_PROBLEM	Define mathematical problem: direct solution or continuous adjoint formulation
REGIME_TYPE	Set the regime type: compressible or in-compressible
SYSTEM_MEASUREMENTS	Choose between international system of units and United States units
FLUID_MODEL	Ste how to model the fluid: standard air, ideal gas, constant density and other
MACH_NUMBER	Set the number of Mach
AOA	Set the angle of attack in degrees
SIDESLIP_ANGLE	Set the sideslip angle in degrees
FREESTREAM_PRESSURE	Value of the free stream pressure in N/m^2 or lbf/ft^2
FREESTREAM_TEMPERATURE	Value of the free stream temperature in Kelvin
REF_ORIGIN_MOMENT	X, Y and Z coordinates of the reference origin for moment computation
REF_LENGTH	Reference length used to compute non dimensional coefficients
REF_AREA	Reference area used to compute non dimensional coefficients
EXT_ITER	Maximum number of iterations
MESH_FILENAME	Name of the input mesh file
MARKER_EULER	Name of the Euler boundary. Coherent with the one indicated in the mesh file
MARKER_FAR	Name of the far field boundary. Coherent with the one indicated in the mesh file
MARKER_MONITORING	Name of the surface in which the the aerodynamic coefficient will be evaluated

Table 3.3: Input SU2 parameters for Eurlian compressible analysis.

are specified. For boundaries, in two dimensional grid only line elements are allowed and in three dimensional mesh triangular and rectangular boundary elements are allowed.

Solution and restart file. The output file of SU2 can usually be visualized with ParaView [30] or with Tecplot [31], indeed their format can respectively be .vtk or .plt. After each simulation, the software generates several files containing all the necessary information for plotting and restarting the solution. As instance, for a direct flow solution or for an adjoint solutions, the following files are provided [28].

- Full volume flow solution containing the flow-field data or the gradients data along all the volume mesh.
- Flow solution on specified surfaces. Contains the same information of full volume flow solution but only for a singular surface.
- Comma separated values (.csv) file containing output values on specified surfaces.
- Restart file in a native format for restarting simulations in SU2.
- File containing the convergence history information. Here also the residual value of the solution are provided.

The name of these file can be modified in the configuration file.

3.3.3 Physical Modeling Equations

The structure of SU2 is conceived to solve problem on three dimensional domains delimited by disconnected boundaries. The PDE system representing the physical model of such a problem should have the structure of Eq. 3.1 with appropriate boundary and temporal conditions which depend on the problem in analysis.

$$\partial_t U + \nabla \cdot \vec{F}^c - \nabla \cdot \vec{F}^v = Q \quad \text{in } \Omega, t > 0 \quad (3.1)$$

In this general equation, U is the vector of the state variables, \vec{F}^c and \vec{F}^v represent respectively convective and viscous fluxes, $Q(U)$ is the source term and $\Omega \subset \mathbb{R}^3$ is the domain of interest.

Different physical problems are implemented in SU2 based on Eq. 3.1. A brief

description of compressible RANS governing equations is now presented. Euler equations are considered a sub-case of this equation obtainable removing the viscosity terms.

Reynolds-averaged Navier-Stokes Equations, Compressible Formulation.

Compressible Navier-Stoker equations are classically used to represent flow field surrounding the object of interest. They describe the conservation of mass, momentum and energy in a viscous fluid. In SU2 solver, the vector of state variables is $U = (\rho, \rho v_1, \rho v_2, \rho v_3, \rho E)^\top$. ρ is the air density, E is the total energy per unit of mass and $\vec{v} = (v_1, v_2, v_3) \in \mathbb{R}^3$ represents the flow speed in Cartesian coordinate system. Furthermore, Eq. 3.2 illustrates how convective and viscous fluxes are modelled.

$$\vec{F}_i^c = \begin{pmatrix} \rho v_i \\ m_i v_1 + P \delta_{i1} \\ \rho v_i v_2 + P \delta_{i2} \\ \rho v_i v_3 + P \delta_{i3} \\ \rho v_i H \end{pmatrix}, \quad \vec{F}_i^v = \begin{pmatrix} \cdot \\ \tau_{i1} \\ \tau_{i2} \\ \tau_{i3} \\ v_j \tau_{ij} + \mu_{tot}^* C_p \partial_i T \end{pmatrix}, \quad i = 1, \dots, 3 \quad (3.2)$$

In these equations P is the static pressure, H represents the fluid enthalpy, δ_{ij} is the Kronecker function, τ_{ij} are the viscous stresses which can be written as $\tau_{ij} = \mu_{tot} \left(\partial_j v_i + \partial_i v_j - \frac{2}{3} \delta_{ij} \nabla \cdot \vec{v} \right)$, C_p is the specific heat at constant pressure, T is the temperature and R is the gas constant. In conclusion, μ_{tot}^* can be obtained through Eq. 3.3.

$$\mu_{tot}^* = \frac{\mu_{dyn}}{Pr_d} + \frac{\mu_{tur}}{Pr_t} \quad (3.3)$$

Here μ_{dyn} represents the dynamic viscosity which is assumed to satisfy Sutherland's law, μ_{tur} is turbulent viscosity, which is computed via turbulence model and Pr_d and Pr_t are respectively the dynamic and turbulent Prandtl numbers.

3.3.4 Adjoint Formulation and Sensitivity Computation

In gradient-based optimization techniques, the minimization of the objective function is achieved through an iterative process during which sensitivity derivatives

of the objective function with respect to the design variables have to be evaluated. Adjoint method is a way to compute the gradients of the objective function. It is widely spread thanks to its characteristic to compute derivatives with the same computational time required to solve the state PDEs. Adjoint methods can be divided in two different typologies. The continuous approach in which the adjoint equation are obtained from the governing PDE and then discretized and the discrete approach in which the adjoint equations are directly obtained from the discretized governing equations. The main step of the continuous adjoint methodology implemented in SU2 and applied to Navier-Stokes equation is now illustrated [28]. Then the way in which design variable are defined and a brief explanation of the optimization framework are showed.

Continuous adjoint methodology. As said before, the domain Ω in which are considered the Navier-Stokes equations is delimited by disconnected boundaries which can be divided into inlet, outlet and more wall boundaries indicated as S . For optimal shape design, it is possible to consider the objective function J defined on the solid wall S . A generic objective function depends on the flow variables U obtained from the already computed solution of the flow field equations. By consequence, the generic optimization problem can be represented by Eq. 3.4.

$$\begin{aligned} \text{find } S^{min} \in \mathcal{S}_{ad} : J(S^{min}) = \min_{S^{min} \in \mathcal{S}_{ad}} J(S) \\ \text{with } J(S) = \int_S j(\vec{f}, T, \vec{n}) ds \end{aligned} \quad (3.4)$$

\mathcal{S}_{ad} is the set of admissible boundary geometries and $j(\vec{f}, T, \vec{n})$ is a continuous function that depends on T , on \vec{n} which is the normal to S positive if inward-pointing and on \vec{f} that is equal to $P\vec{n} - \vec{\sigma} \cdot \vec{n}$ where σ is the second order tensor of viscous stresses.

As typically occurs in the adjoint approach the flow equations are included in the objective function as constrains, multiplied by a Lagrange multiplier for each equation $\Psi^T = (\psi_1, \psi_2, \psi_3, \psi_4, \psi_5)$. By this way the Lagrangian is introduced as showed in Eq. 3.5 where $R(U)$ are the Navier-Stokes equations.

$$\mathcal{J}(S) = \int_S j(\vec{f}, T, \vec{n}) ds + \int_{\Omega} (\Psi^{\top} R(U)) d\Omega \quad (3.5)$$

It is possible to consider an arbitrary small perturbation of the boundary S and without losing generality it can be assumed to be an infinitesimal deformation δS in the positive direction of the normal direction to S . Moreover, assuming a regular solution U and a smooth boundary S the change of the Lagrangian can be written as

$$\delta\mathcal{J} = \int_S \delta j(\vec{f}, T, \vec{n}) ds + \int_{\delta S} j(\vec{f}, T, \vec{n}) ds + \int_{\Omega} (\Psi^{\top} \delta R(U)) d\Omega \quad (3.6)$$

in which $\delta R(U)$ is the variation of R . After calculation showed in [28] and considering the equation in the system 3.7, the term regarding the integral domain in 3.6 can be eliminated.

$$\begin{cases} \delta R(U) = \frac{\partial R}{\partial U} \delta U = 0 & \text{in } \Omega \\ \delta \vec{v} = -\partial_n \vec{v} \delta S & \text{on } S \\ \partial_n(\delta T) = (\nabla T) \cdot \nabla_S(\delta S) - \partial_n^2 T \delta S & \text{on } S \end{cases} \quad (3.7)$$

Then with an appropriate choice of the boundary conditions it is possible to write the variation of the Lagrangian as indicated in Eq. 3.8 where h is the shape sensitivity and does not depend on the variation of the flow variable.

$$\begin{aligned} \delta\mathcal{J} &= -\int_S h \delta S ds = \int_S \left(\vec{n} \cdot \bar{\Sigma}^{\varphi} \cdot \partial_n \vec{v} - \mu_{tot}^* C_p \nabla_S \psi_5 \cdot \nabla_S T \right) \delta S ds \\ &\text{with } \bar{\Sigma}^{\varphi} = \mu_{tot} \left(\nabla \vec{\varphi} + \nabla \vec{\varphi}^T - \mathbf{I}_d \frac{2}{3} \nabla \cdot \vec{\varphi} \right) \end{aligned} \quad (3.8)$$

It possible to call $\delta\mathcal{J}$ the surface sensitivity. It provides the change of the objective function with respect to infinitesimal surface shape deformation in the direction of local surface normal. The surface sensitivity can be computed at every node of the grid with not so high computational cost.

Design variable definition. Thanks to the continuous adjoint methodology it is possible to compute the variation of the objective function with respect to an infinitesimal deformation of the shape in the direction of the local surface normal. By consequence, it could be possible to use the surface points of the mesh as design

variable in order to perform a shape optimization. This kind of approach is not usually followed. A more convenient choice could be to project the gradient of the objective function, obtained from deformation of each surface point, into a smaller set of design variables. SU2 provides two methodologies which follow this kind of approach. In case of two dimensional study, Hicks-Henne bump functions are employed for airfoil calculations. These functions can be added to the original airfoil geometry to modify the airfoil shape. In case of three dimensional shape, a FFD strategy is exploited. As already explained in 2.2.1, a box that surround the surface to be optimized have to be initialized. This box is parameterized as a Bézier solid. A set of control points on the surface of the box is defined, the number of control point will affect the order of Bernstein polynomials. The solid box is represented by Eq. 3.9.

$$X(u, v, w) = \sum_{i=0}^l \sum_{j=0}^m \sum_{k=0}^n P_{i,j,k} B_i^l(u) B_j^m(v) B_k^n(w) \quad (3.9)$$

l, m and n are the number of control point chosen respectively in x, y and z direction of Cartesian space, $u, v, w \in [0, 1]$ are the parametric coordinates in those directions, $P_{i,j,k}$ represents the coordinates of the generic control point which has indexes i, j and k , $B_i^l(u)$, $B_j^m(v)$ and $B_k^n(w)$ are the Bernstein polynomials. Therefore, the Cartesian coordinates of a generic surface point are transformed into parametric coordinates within Bézier box. In this way, the control points on the box become design variables. The box is deformed modifying the position of its control points and this leads to a smooth deformation of the surface point inside the box. In an aerospace system, FFD box allows changes in thickness, sweep, twist, dihedral, span and other. In Fig. 3.3 an example of FFD box for swept wing is illustrated. The number of control points are 11 in x direction, 9 in y direction and 2 in z direction. Indexes of some control point are also showed.

Optimization Framework. Using adjoint formulation is possible to compute the gradients of a wide range of objective functions. Some typical example could be drag minimization, lift maximization, pitching moment, quadratic deviation from a target pressure (inverse design), aerodynamic efficiency and also a combination of those

functions. In addition it is possible to set design constraints regarding flow parameter such as lift, drag of the function above-mentioned and geometrical constraints such as maximum and minimum thickness, area, volume, chord, twist, curvature, dihedral and so on. The optimizer uses the SciPy library [32] that is able to provide efficient numerical routines for non-linear constrained optimization problems.

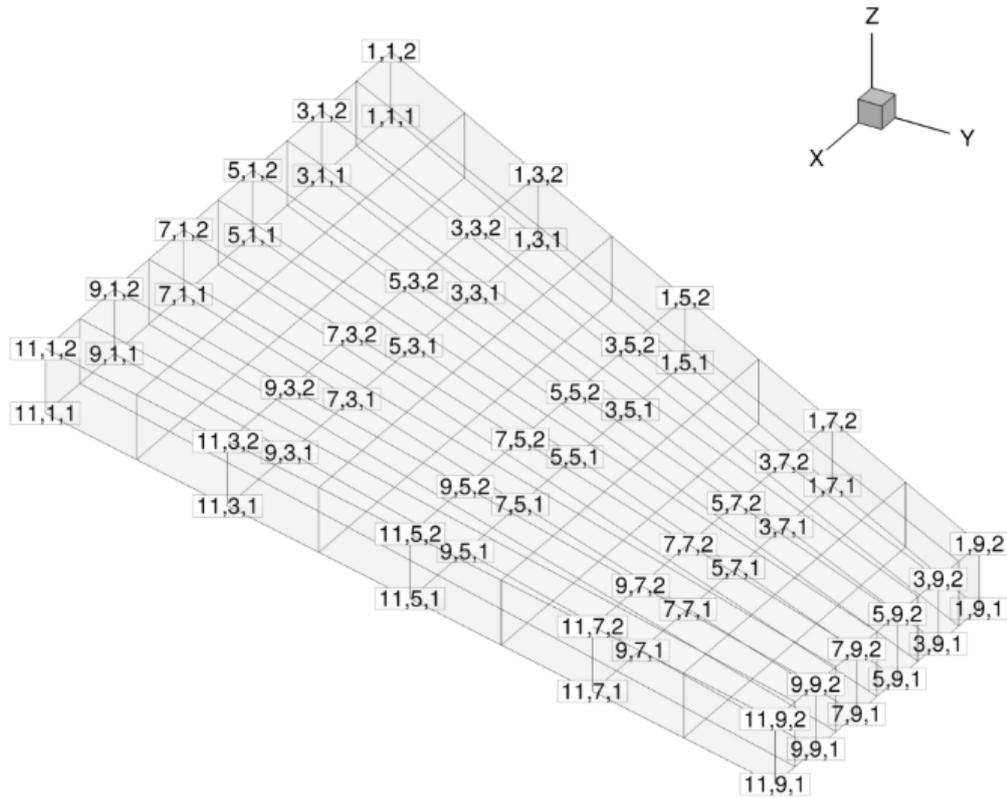


Figure 3.3: View of an FFD box with indices identifying control points [33].

3.4 Automatic CSD analysis process

An automatic workflow which allows to obtain structural computation using as input the geometry to analyze and the force distribution on a mesh which represents the object under study is provided by DLR. The geometry must be described through a CPACS file. The forces distribution need to be given as input in a file which contains the nodal points coordinates, the cell forces distribution and the barycenter of each cell. The structural computation is performed by LAGRANGE, a finite

element solver. A pre and post processing is required to generate the appropriate inputs for this tool and to obtain the output displacement of the input mesh nodes. First the structural tool is described and then, the architecture of the complete workflow is presented. The workflow used is similar to the one described in [34].

3.4.1 Structural tool LAGRANGE

LAGRANGE software consists of a finite element solver which main aims are to well suit to thin walled stiffened structures used in aerospace and provides flexible optimization algorithms [35]. Developed by Cassidian, Airbus defence & security systems department [36] in 1984, Lagrange software has been continuously improved and exploited for the design of military and civil aircraft such as Eurofighter, X-31, A400M, A380, A350, Talarion and ATLANTE as well as to future aircraft projects. LAGRANGE provides several system analysis such as linear statics, linear dynamics, linear stability, steady aero-elastics and unsteady aero-elastics. The characteristics which make this tool different than other commercial codes are the possibility to compute analytical sensitivities of a system response with respect to a given set of design variables and the linear aerodynamic analysis tool available to perform aero-elastic and loads analysis, including sensitivities studies. Different optimization algorithms are included in the program and are able to manage a large amount of design variables. Possible design variables are cross sectional areas of bars (e.g. stringers), geometrical sizes and composite lay up of various cross sections commonly used in aerospace, ply thicknesses and fibre orientations in composites, thicknesses of shell elements, shape variables and in the case of steady aero-elastic analyses, the trim variables such as angles of attack of the aircraft and control surfaces. In addition, a wide range of design criteria and displacement, dynamic or manufacturing constraints are available. In this way the multidisciplinary requirement can be easily checked and satisfied. The input of the finite element model is NASTRAN compatible, enabling the possibility to generate the input model also with NASTRAN supporting pre-processors.

3.4.2 Structural Process Architecture

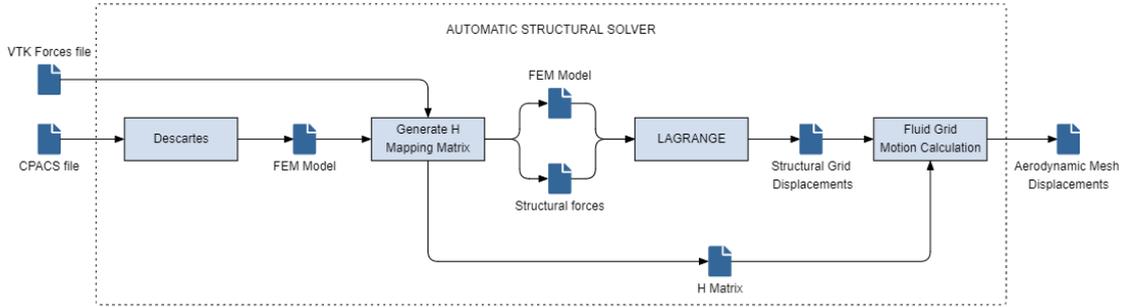


Figure 3.4: Architecture of the workflow used for structural calculation obtained from aerodynamic mesh nodes forces distribution.

Fig. 3.4 illustrates the structural workflow architecture, highlighting the data generated and transferred to each single tool. As can be noticed, the input files are the CPACS file which has the description of the geometry and a vtk file containing the aerodynamic mesh with the cell forces distribution. As previously mentioned, in order to execute LAGRANGE structural solver using a CPACS file for geometry description of configuration, a pre processing phase is required. The FEM structural model is automatically generated from Descartes [37], which provides the capabilities to generate the model from geometrical and structural description of a CPACS file. The structural model is generate in suitable LAGRANGE format. After that, two meshes are analyzed by the workflow. The structural one, which is just generated and the aerodynamic grid given as input. Obviously, the two meshes does not match each other. This means that the displacements of the structural domain must be mapped to the fluid domain and similarly, the forces on the aerodynamic domain have to be mapped to the structural one. If an high-fidelity method is exploited for aerodynamic calculation, the fluid domain is probably discretized by a much finer mesh with respect to the structural domain. In a general case, the discrete mapping equation used to compute the aerodynamic mesh displacements u_{fluid} from structural grid displacements $u_{structure}$ can be expressed as follows.

$$u_{fluid} = H^T * u_{structure} \quad (3.10)$$

H^T represents transpose of the interpolation mapping matrix. The interpolation method used to compute this matrix is based on radial basis function, described in [38]. It is a global interpolation problem which takes into consideration all model points. The mapped matrix can be expressed as follow.

$$\mathbf{H}^T = \mathbf{A}_{fluid,struct} \cdot \left[\begin{array}{c} \Phi_P \mathbf{P} \Phi_{st,st}^{-1} \\ \Phi_{st,st}^{-1} - \Phi_{st,st}^{-1} \mathbf{P}^T \Phi_P \mathbf{P} \Phi_{st,st}^{-1} \end{array} \right] \quad (3.11)$$

With

$$\mathbf{A}_{fluid,struct} = [\mathbf{P}_{fl} \quad \Phi_{fl,st}] \quad \text{and} \quad \Phi_P = \left(\mathbf{P} \Phi_{st,st}^{-1} \mathbf{P}^T \right)^{-1} \quad (3.12)$$

P_{fl} and P_{st} contain the polynomial parameters, whereas the generic $\phi_{\Gamma_1, \Gamma_2}$ represents the evaluation of the radial basis function for the Euclidian distance between two generic points of the domains Γ_1 and Γ_2 , indicate as $\mathbf{x}_{\Gamma_1, i}$ and $\mathbf{x}_{\Gamma_2, j}$ in Eq. 3.13.

$$\phi_{\Gamma_1, \Gamma_2} = \phi \left(\|\mathbf{x}_{\Gamma_1, i} - \mathbf{x}_{\Gamma_2, j}\|_2 \right) \quad (3.13)$$

The above described algorithm and also another one based on moving last squares methodology are implemented in a Python based code by DLR. Using Eq. 3.11, the tool is able to generate the mapping matrix and calculate the forces distribution on structural grid $f_{structure}$ using f_{fluid} , the forces distribution on aerodynamic mesh, given as input. This computation is executed through Eq. 3.14.

$$f_{structure} = H * f_{fluid} \quad (3.14)$$

The use of the same interpolation matrix presents in Eq. 3.10 is justified by the principle of virtual work. After this computation, all the input required for LAGRANGE tool are ready. Using the FEM model and the load distribution on structural grid, a static structural analysis is computed. In this way, the displacements of structural nodes are obtained. After transpose of mapping matrix calculation, which is already obtained, by using Eq. 3.10 the displacements of the fluid domain mesh points can be computed. These points coordinates compose the output of the described workflow.

4

Collaborative Aerodynamic Optimization and Aero-Structural Analysis

Contents

4.1	Introduction	47
4.2	Aerodynamic Analysis and Optimization	48
4.2.1	CFD Analysis Parallel Computation	48
4.2.2	Shape Optimization	49
4.2.3	Test Cases	54
4.3	Mesh Deformation	62
4.3.1	Deformation methods	62
4.3.2	FFD Mesh Deformation	63
4.3.3	Mesh Motion Points Deformation	67
4.4	Aero-Structural Analysis Workflow	69
4.4.1	Presentation of the Workflow	69
4.4.2	Input, Output and Configuration	72
4.4.3	Aero-Structural Interface Methods	74
4.4.4	Workflow Architecture	77
4.5	CPACS Interface	80
4.5.1	Need of a CPACS Interface	80
4.5.2	Aerodynamic Results Writer	81
4.5.3	CPACS Geometry Update	85

4.1 Introduction

The aim of this chapter is to describe the tool and workflow developed by the author to perform collaborative aerodynamic optimizations and aero-structural analysis. In order to ensure a robust automation, thanks to the tool described in section 3.2, it is possible to generate a mesh starting from a general geometry described through a CPACS format file. Using this CFD grid, aerodynamic analysis

and optimization can be performed. Different SU2 routines are isolated and tested to easily manage CFD calculation. In order to have the possibility to couple this kind of tool with the structural one, a mesh deformation routine is developed thanks to the SU2 methods for grid deformation. Then, a study of the capability and the range of possible deformation avoiding mesh distortion is carried out. Another fundamental feature useful for aero-structural coupling is the operation of switching from flow-field pressure to nodal or cell forces, an apposite codes is written to perform this function. The majority of these tool, together with the ones presented in chapter 3 are integrated in a workflow useful to perform aero-structural analysis remotely. Indeed, LAGRANGE tool is run from DLR facilities and the data exchange between tools installed in different work station happens through BRICS, a service made available by RCE, which aim is to enable connecting design competence across different organization. In conclusion, an interface which write SU2 results in CPACS file is provided. Not only aerodynamic results have to be written, also the geometry obtained from aero-structural analysis workflow must to be exported.

All the tool and workflow above-mentioned are presented in the following sections. Also same test case are showed in order to demonstrate their capabilities.

4.2 Aerodynamic Analysis and Optimization

An important characteristic of SU2 is its subdivision in C++ modules that can be used both individually or subsequently in order to perform high-fidelity analysis. As a consequence, coupling of these modules is easily performed through Python scripts. Two kind of this workflows are now presented. The first provide CFD analysis and the other is able to perform CFD optimization based on adjoint method with FFD strategy. The results achievable from these tool are already validated in [26].

4.2.1 CFD Analysis Parallel Computation

SU2 provides a Python script called "parallel_computation.py" which manage the setup and execution of parallel CFD computation on distributed memory

architectures through MPI. The script executes two C++ modules: SU2_CFD in parallel and SU2_SOL. They respectively perform aerodynamic analysis and generate solution output.

SU2_CFD can solve steady or unsteady direct and adjoint problems through different governing equations such as Euler or RANS. It can work with both finite volume method or finite element method with an edge-based data structure. The module has also advanced features that can be used to improve robustness and convergence such as residual smoothing, agglomeration multi-grid, or preconditioners for low-speed applications.

SU2_SOL is the module that writes volume and surface solution files in one of the available format. It can be called to generate a new set of solution files from a mesh, a configuration file and a restart file that contain the solution at each node.

The input file required for this tool are the ones described in 3.3.2. A mesh file and a configuration file with the parameters showed in Tab. 3.3 are required. The output of such an analysis consists in flow solution data on each point of the surface or volume grid, a file that can be used to restart further analysis and an history file that provides the main information for each iteration performed. Tab. 4.1 contains a list of data available in the solution file.

4.2.2 Shape Optimization

SU2 modules offer also the possibility to perform a CFD optimization based on adjoint method. Fig. 4.1 illustrates the architecture of a Python-based tool written in order to execute all the preliminary activities needed to set and subsequently perform a constrained shape optimization. As can be noticed, the tool input are the file containing the mesh and the parameters present in a CPACS file, the output consists in a set of files containing flow field, gradient and optimized geometry data. The tool employs different SU2 routines. As can be noticed in Fig. 4.1, two of that are used in a preliminary phase in order to obtain some of the input for the shape optimization. After that, the shape optimization loop is executed. A brief description of each module task is now presented.

Flow Field File Solution Data	History File Data
Density	Non-dimensional forces: C_L , C_D , C_Y ,
Energy	C_{F_x} , C_{F_y} , C_{F_z}
Mach number	Non-dimensional moments: C_{M_x} , C_{M_y} ,
Pressure	C_{M_z}
Pressure coefficient	Efficiency C_L/C_D
Temperature	Residual values of density, velocity and
X, Y and Z Momentum	energy
	Computational time
Data provided only after RANS simulation	
Eddy viscosity	Total heat flux
Heat flux	Maximum heat flux
Laminar viscosity	Residual value of nu tilde
Nu tilde	
X, Y and Z Skin friction coefficient	
Y plus	

Table 4.1: Solution data available from CFD analysis performed with SU2

SU2_GEO. SU2_GEO module evaluates geometric proprieties of the object in analysis. It is the first tool run in order to provide geometrical information about fuselage or wing. With reference to specific wing or fuselage sections indicated in the configuration file it is able to calculate a wide range of three and two dimensional data such as wing volume and area, section thickness, chord, radius, twist and other. Since in the following chapters gradient optimization are performed with possible deformation only in Z direction, the thickness distribution is stored thanks to this tool in order to impose thickness constraint on shape deformation during optimization. SU2_GEO also provide a file with the coordinates of points for each wing sections indicated in the configuration file. As can be noticed in Fig. 4.1, this module is also used as first step of the sub-module "Shape Optimization". As matter of fact, it is used to check and direct the shape deformation in order to avoid a possible violation of the imposed geometrical constraints.

SU2_DEF. The main task of SU2_DEF is the mesh deformation. It is able to compute a geometrical deformation of the surface mesh and by consequence modify the position of the surrounding nodes which compose the volumetric grid. Different technique of surface mesh deformation are available, two of these are analyzed in

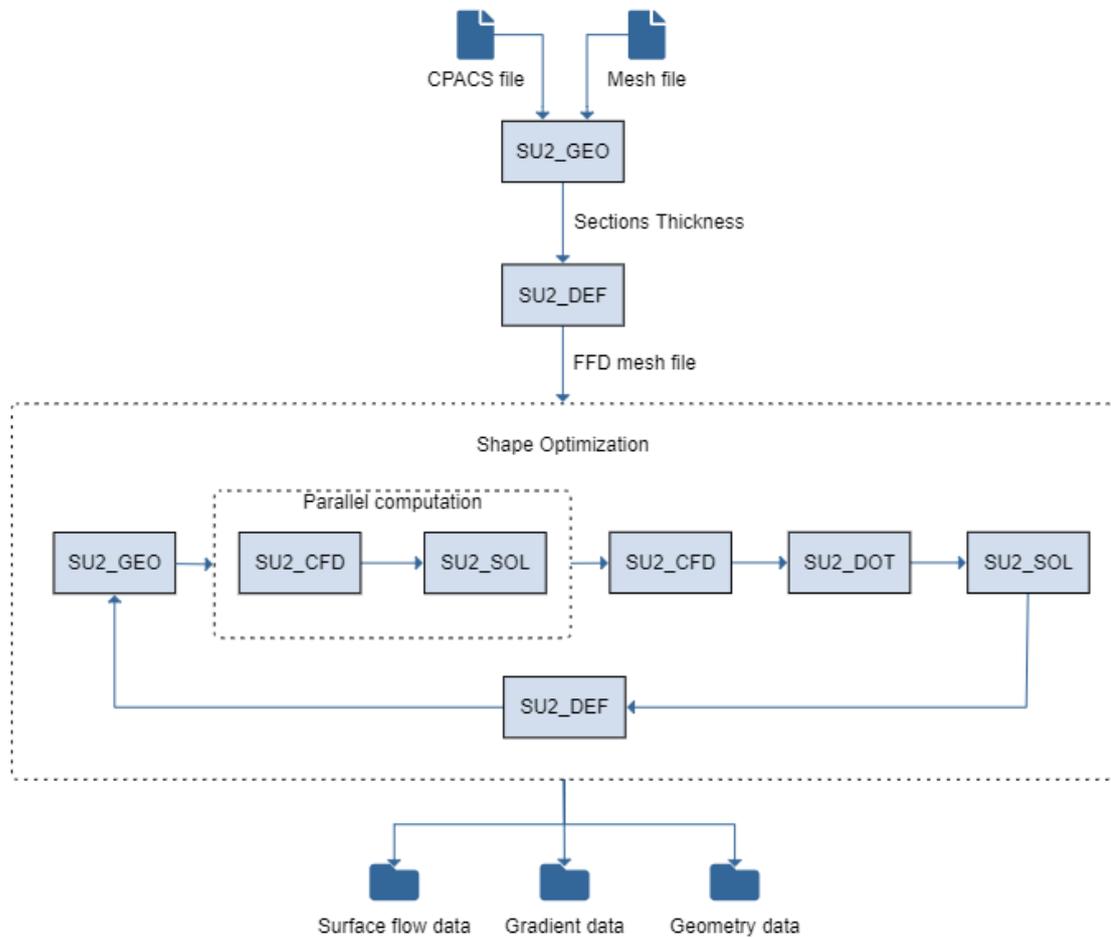


Figure 4.1: Architecture of shape optimization tool. The arrows indicate temporal sequence of execution

section 4.3. After the deformation, a method based on linear elasticity equations is exploited to deform the surrounding volume mesh. Another feature of this tool is the capability to define an FFD box inside the mesh. Indeed, in the mesh file not only the coordinates of each design point belonging to the box should be written, but also the nodes of the mesh that can be moved by a deformation of the box must be indicated. This happens for each one of the surfaces that the box can deform. In the tool represented in Fig. 4.1, SU2_DEF is firstly used only to define the FFD points in the mesh file. Then, in the optimization loop is used at the end of each iteration to deform the original shape and start a new iteration with a new mesh.

Shape Optimization Loop. The above mentioned modules are used in order to obtain some of the input for shape optimization. The other inputs are available in

the CPACS file. The optimization loop requires a mesh in which an FFD box is defined. It is also possible to add geometrical or aerodynamic constraints, in this case it is automatically inserted a thickness constraint on the section indicated as input. As instance, it is possible to impose that the thickness value can not become lower than a certain value. Since the original thickness value is calculated during preliminary phase, using a rate value inserted as input it is possible to impose as limitation a percentage value of the original one. For each iteration of the shape loop, geometry value of the analyzed surface are computed in order to control if the constraint are respected. Then a CFD analysis is performed to obtain the solution of the problem, used as input for gradient calculation. SU2_CFD is also able to solve adjoint equation and compute, as explained in 3.3.4, the gradient with reference to a shape deformation in local normal direction for each point of the grid. After that, SU2_DOT is executed. This module is able to project the gradient already calculated for each node on defined shape design variable and so compute derivative of particular objective function through a dot product operation. As explained in section 4.2.1, SU2_SOL generate the output file useful to visualize the solution for future analysis or debugging. Finally, SU2_DEF is executed in order to deform the mesh and restart a new iteration of the loop. The Python script that orchestrates all these modules, checks for each iteration if the value of the objective function is improved (typically reduced). If not, the computation of the gradient is skipped and a new shape deformation is performed in order to try to achieve an enhancement of the objective function.

Some of the possible input parameters are already presented in Tab. 3.3, they are required to set the CFD problem in which the optimization will be performed. Indeed, in Tab. 4.2 are presented the additional input parameters needed to chose proper configuration for shape optimization. In Tab. 4.3 output data available after shape optimization are presented. They are divided in two main groups: gradient computation data and geometry data. In addition, also output shown in Tab. 4.1 regarding flow field and convergence solution data are provided.

Parameter	Description
FFD_DEFINITION	Vector containing the coordinates of the points which compose the FFD box
FFD_DEGREE	Vector with the number of control points that will be defined on FFD along x, y and z axis as showed in Fig.3.3
OPT_OBJECTIVE	Objective function that will be minimized. It is possible to sum or multiply more functions and multiply each function by a scale factor
OPT_CONSTRAINT	Definition of constrains imposed as "Objective >/=/< Value". Value is a number that will be multiplied to the initial value of specified objective
CONSTRAINT_FACTOR	Weight factor that can be multiplied by each constraint
OPT_GRADIENT_FACTOR	Scale factor that multiply objective function, gradients and constraints. It is used to obtain the value of the norm of the gradient near to the recommended value of 1E-6
OPT_RELAX_FACTOR	Scale factor which allows to obtain higher value of deformation towards the minimum objective function direction
OPT_SURFACE	Name list of the mesh boundary that represent the surface that have to be optimized
GEO_BOUNDS	Positions of section in which calculate two dimensional geometry value. Y coordinates in case of wing and X coordinates in case of fuselage
GEO_NUMBER_STATIONS	Number of section in which evaluate the two dimensional geometry
DIRECTION_OF_DEFORMATION	Vector with True of False value to indicate which of x, y and z directions of optimization are allowed

Table 4.2: Input parameters for shape optimization tool. In addition, also parameter indicated in Tab. 3.3 must be provided.

Adjoint Solution Flow Data	Wing Geometry Data	Fuselage Geometry Data
Adjoint density	Volume	Volume
Adjoint energy	Area	Wetted area
Adjoint X,Y and Z momentum	Area of projection in x,y and z planes	Sections area Area of projection in x,y and z planes
Surface sensitivity	Max and min coordinates	Max and min coordinates
Angle of attack sensitivity	Thickness distribution	Width distribution
Design variable gradients	Chord distribution	Height distribution
Adjoint residuals	LE radius distribution	Max curvature
	Twist distribution	Points coordinates of the required sections
	Dihedral distribution	
	Sections area	
	Points coordinates of the required sections	

Table 4.3: Solution data available after shape optimization. In addition, solution data presented in Tab. 4.1 are provided.

4.2.3 Test Cases

Two test cases are now presented to demonstrate the capability of adapting to different geometries and to show the performance of the above-presented tools. First a CFD analysis and subsequently a shape optimization based on adjoint method are performed.

Parameter	Value in Cruise Condition	Value in Low Speed Condition
Physical problem	Euler	Euler
Regime type	Compressible	Compressible
Mach number	0.78	0.2
Angle of attack	2°	-4°, 0°, 4°, 8°, 12°, 16°, 18°, 20°
Side-slip angle	0°	0°
Altitude	12000m	Sea level

Table 4.4: Input data for CFD analysis of Design Campaign 2 aircraft at cruise and low speed condition.

DC-2. The first case concerns the reference aircraft of the so-called Design Campaign-2 (DC-2) of AGILE project. It is a regional civil low-wing aircraft with wing-mounted-engines. The configuration geometry is written in a CPACS file, and as shown in Fig. 4.2 can be visualized using TiGf viewer. From the geometry described in the CPACS file, the mesh presented in Fig. 4.3 is obtained and used

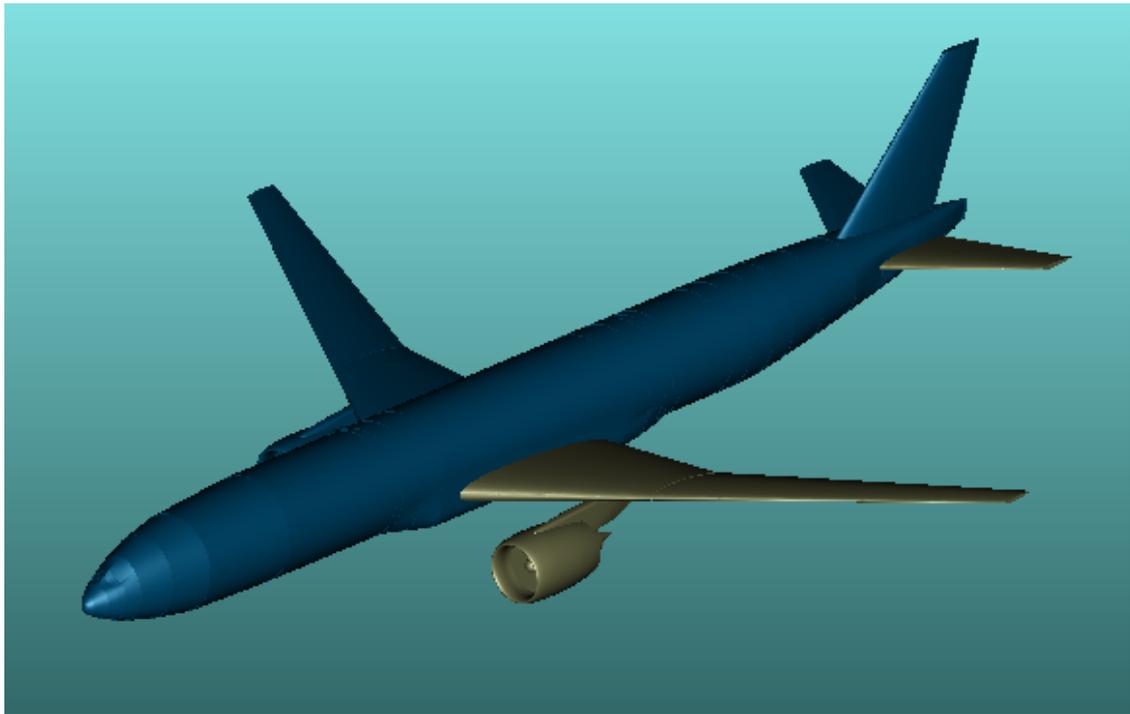


Figure 4.2: AGILE Design Campaign 2 aircraft. Geometry visualization obtained through TiGf viewer.

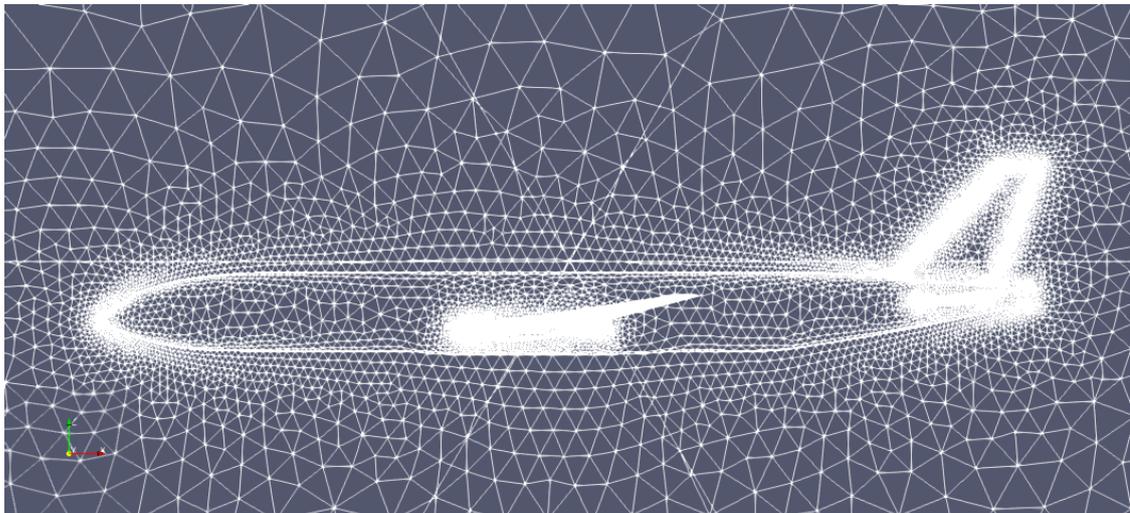


Figure 4.3: AGILE Design Campaign 2 aircraft ParaView visualization of surface mesh surrounded by volumetric mesh.

as input for aerodynamic studies. Two Eurlian compressible CFD analysis are performed. The first simulates cruise condition of such an aircraft. The second simulate low speed condition and it is performed at different angles of attack. Input data of the two analysis are presented in Tab. 4.4. The Mach number distribution in cruise condition obtained form the surface flow field solution is presented in Fig. 4.4, it possible to notice a strong shock wave along the wing span. Finally, in Fig. 4.5 drag polar and the lift coefficient with respect to the angle of attack obtained from CFD low speed analysis are illustrated. It is important to underline that the results are obtained with Eurlian calculation. By consequence the drag coefficient in Fig. 4.5 represents only the sum of induced drag and wave drag coefficients,

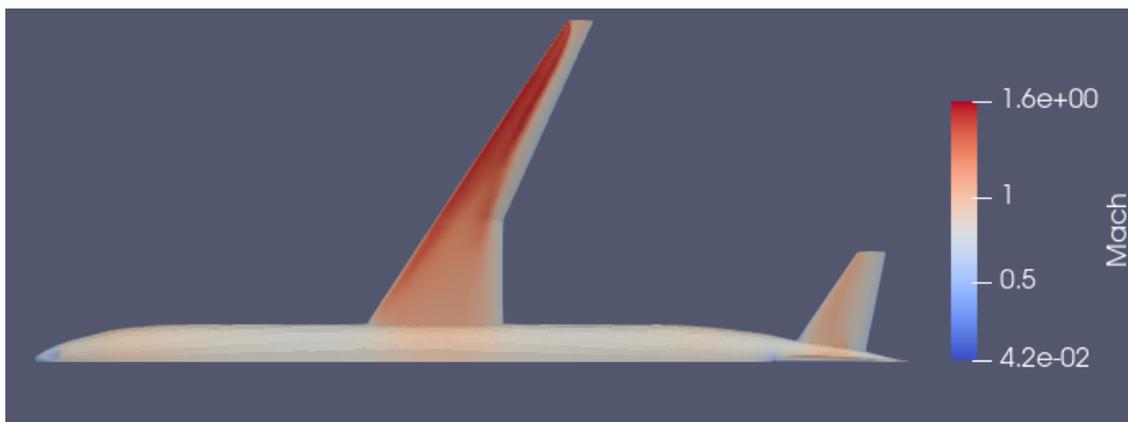


Figure 4.4: AGILE Design Campaign 2 aircraft Mach number distribution in cruise condition.

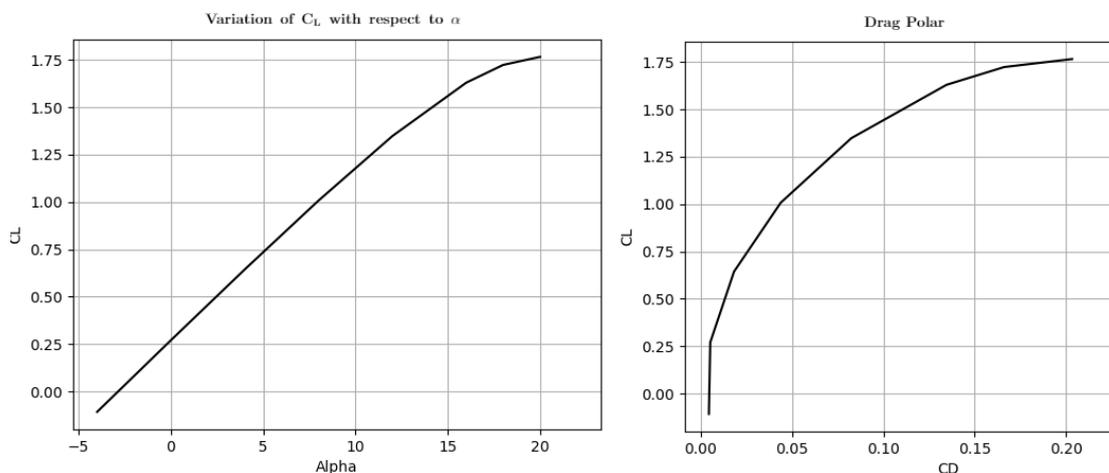


Figure 4.5: AGILE Design Campaign 2 aircraft results from CFD analysis at low speed condition.

friction drag is obviously absent. For the same reason, despite the very high value of the angle of attack reached, also the viscous stall is absent. The change in the slope of the lift-alpha curve is probably due to a wave stall. In order to see the two not found phenomena, a viscous calculation should be provided.

OPTIMALE. The second case study regards the shape optimization of a UAV configuration called MALE (Medium Altitude Long Endurance). It is a conventional low-wing and T-tail configuration with rear-mounted turbofan engines. In Fig. 5.1 is illustrated a representation of the aircraft configuration obtained through TiGL viewer. The configuration is described in CPACS file. In Tab. 4.5, the inputs used to execute the shape optimization are presented. It is chosen to simulate a cruise condition with a speed higher than the characteristic cruise speed of typical UAV

Parameter	Value
FFD_definition	Vector containing the coordinates of the FFD box corner points illustrated in Fig. 4.6
FFD_degree	8, 20, 1
Opt_objective	Drag coefficient
Opt_Constraint	Station_thickness > 0.95% original thickness
Constraint_factor	0.001
Opt_gradient_factor	1E-5
Opt_relax_factor	1e-3
Geo_bounds	[0.1, 14.8] m
Geo_number_stations	5
Direction_of_deformation	Z direction
Physical problem	Euler
Regime type	Compressible
Mach number	0.65
CL fixed	0.85
Side-slip angle	0°
Altitude	15000 m

Table 4.5: Main input parameters for Optimale shape optimization tool.

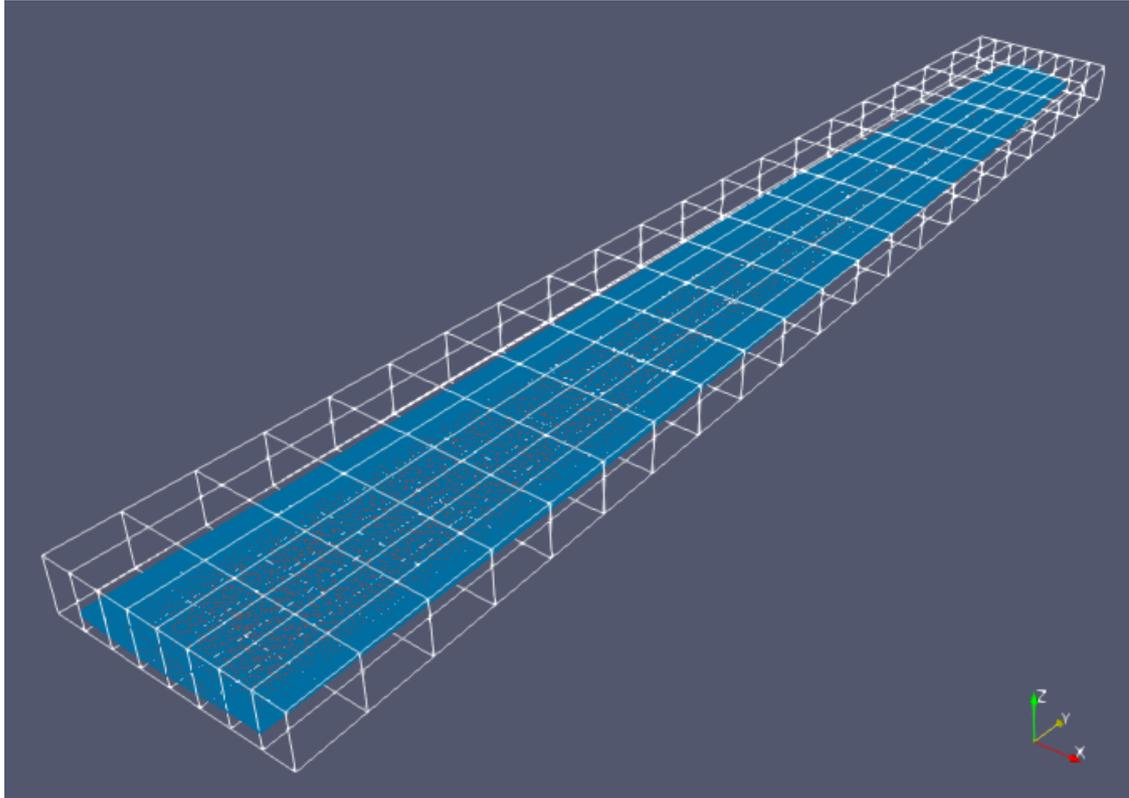


Figure 4.6: Optimal wing mesh surrounded by FFD box. Both data are provided as input to run the shape optimization.

missions. The reason is linked to the intention of performing a Eulerian compressible analysis with the presence of wave drag. In this way, it is easier for the optimizer to find a shape with a reduced value of the drag. The analysis is performed at a fixed value of C_L , indeed SU2 has inner iterative methods to find the angle of attack in which a specified value of lift is produced. In Fig. 4.6 the mesh and the FFD box surrounding the mesh are illustrated. As can be noticed, only the wing is represented by the mesh. This choice is made in order to reduce the computational cost of the optimization. In addition, as can be imagined the greatest contribution of the wing drag came from the wing. As is usually performed, the optimization is executed without the presence of winglets. Indeed, winglets are usually added in an advanced project step. Initially only a wing shape as the one presented in Fig. 4.6 is considered. The shape optimization provides that the design variable, which means all the FFD points represented in Fig. 4.6, can move in Z direction. In order to consider a structural constraint, it is imposed that the value of the thickness

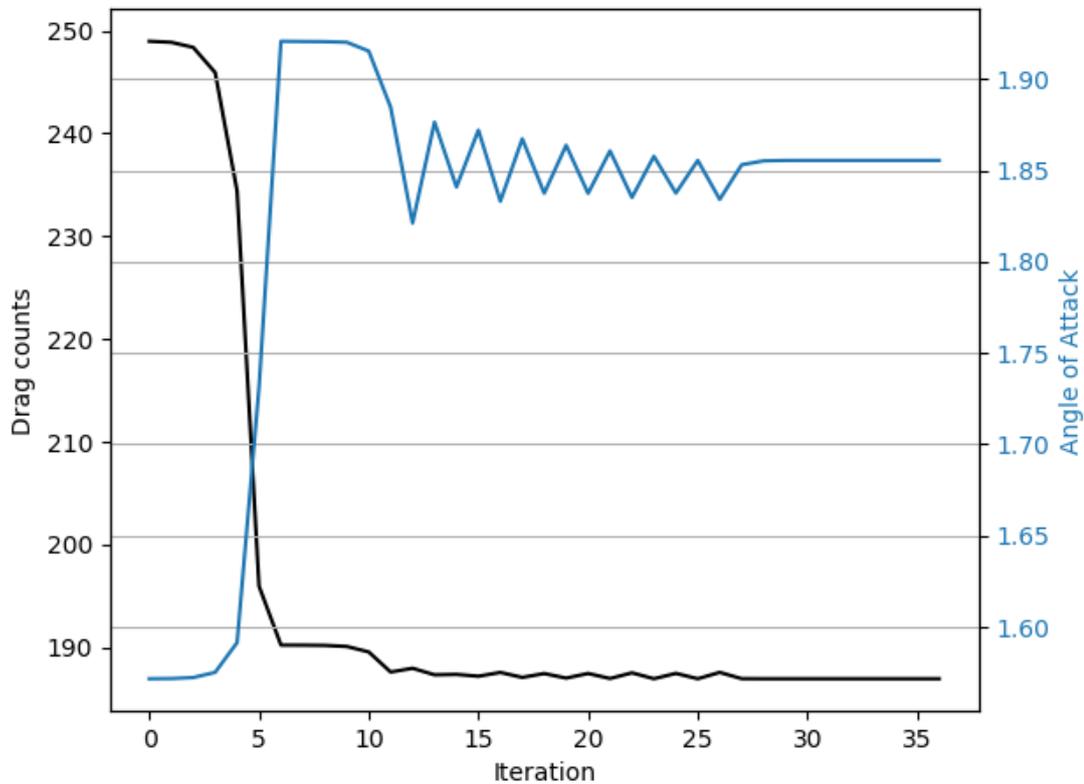


Figure 4.7: Optimale wing shape optimization results. Trends of drag coefficients (expressed in drag counts) and angle of attack during optimization iterations.

of 5 specific section along the wing span can not be reduced more than the 95% of their original value. The choice of the remaining parameter shown in Tab. 4.5, such as the scale factor, are dictated by different trial useful to understand the best configuration for shape optimization.

Several iterations are performed before reaching convergence. As already said, it is not necessary true that the objective function decrease for each iteration. It could happen that the reduction of drag is not satisfied, in that case the tool performs a new shape deformation without recalculate the gradients. This happen until the desired decrease is reached. After several iterations without a consistent decrease of the objective function the optimization is stopped cause a convergence is reached. All the above described characteristic of the results can be observed in Fig. 4.7. As can be noticed, since the calculation is at fixed lift coefficient, for each iteration a new value of the angle of attack is calculated. Between the fourth and the sixth

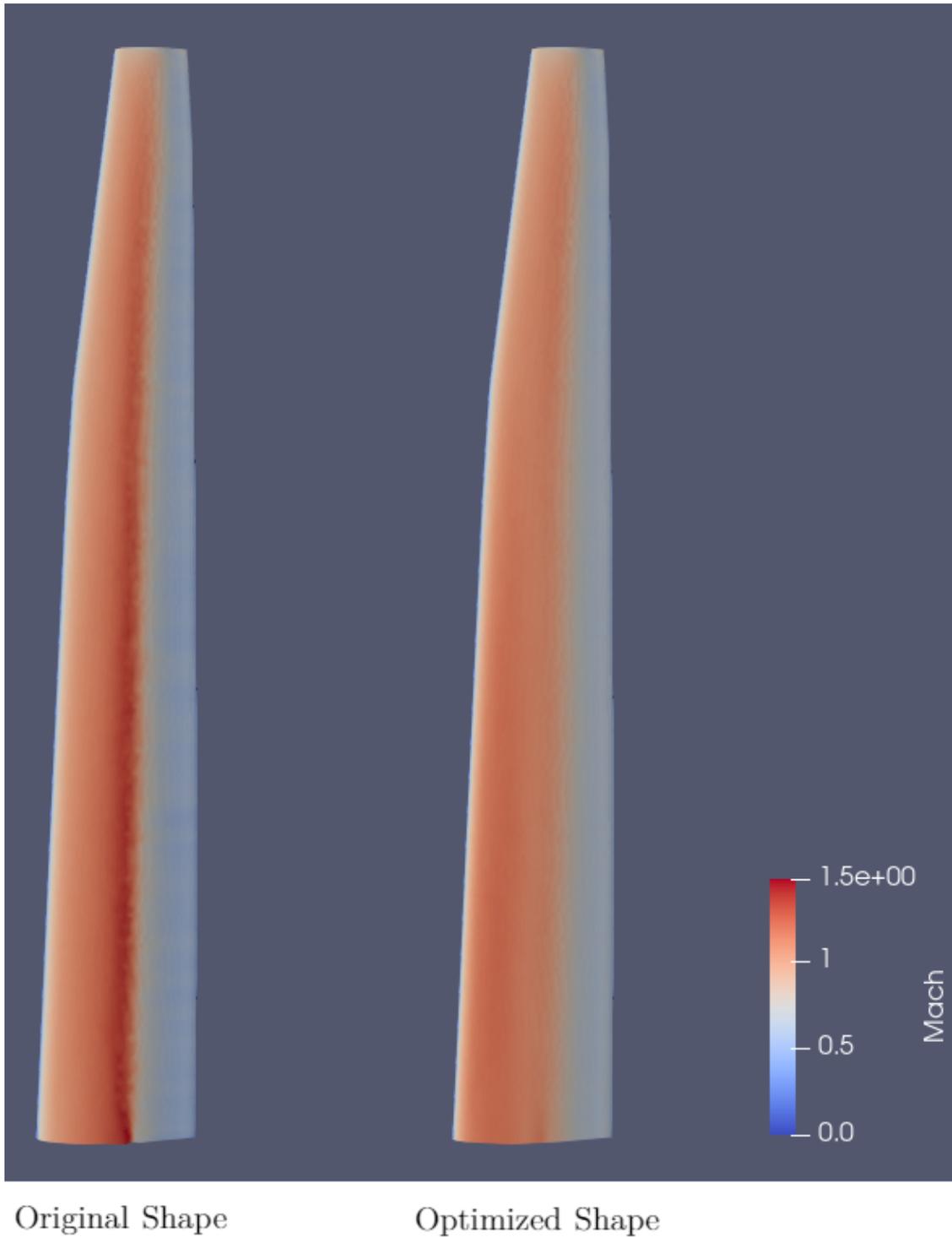


Figure 4.8: Mach distribution comparison between original and optimized Optimal wing shape. It possible to observe the drastic reduction of the wave strength. The maximum number of mach reached is reduced from 1.5 to 1.2.

iteration a huge decrease of the objective function is reached. In such a situation also the value the of angle of attach change significantly. This is due to the strong

change in wave distribution, it brings to different value not only of the drag, but also of the lift coefficient. This aspect can be noticed in Fig. 4.8 and 4.9. In the first one, the Mach number distribution along the wing span is shown. It is possible to notice a great change in the velocity distribution. In particular the maximum value of the Mach number is reduced from 1.5 to 1.2, by consequence the wave drag becomes less strong and a lowered value of drag coefficient is reached. In the second figure the pressure coefficient of the root section of both original and optimized wing shape is illustrated. Also here it is possible to notice the great decrease of the pressure gradient, another indicator of the reduced wave strength.

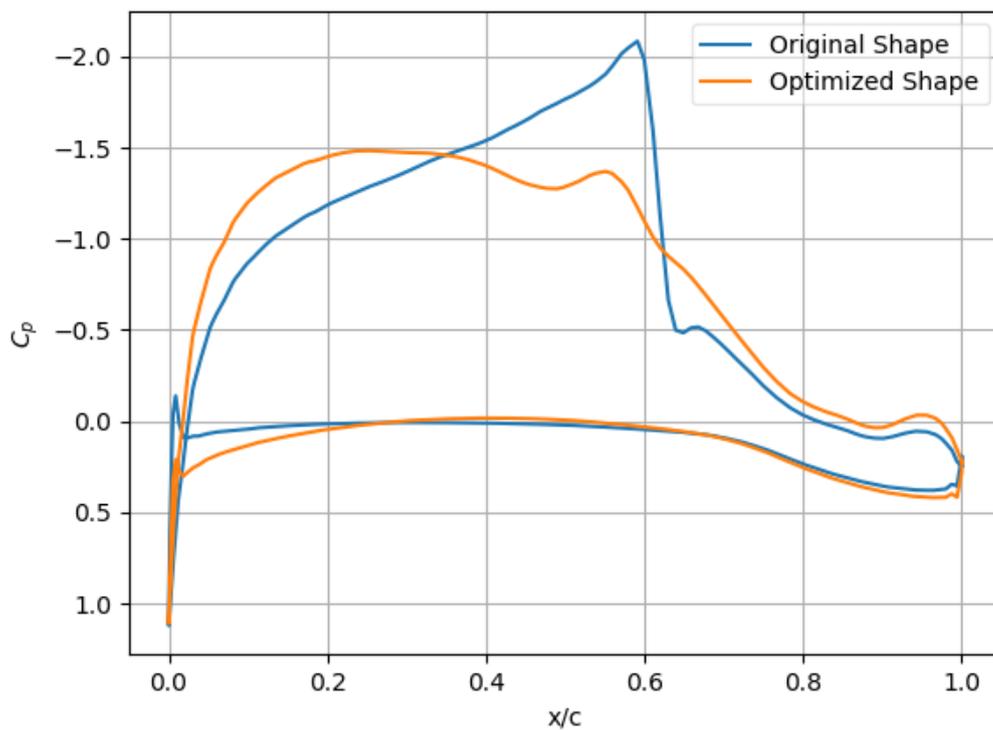


Figure 4.9: Pressure coefficient distribution comparison between original and optimized Optimale root section. The trend is similar for all other sections. It is possible to observe the drastic reduction of the wave strength.

4.3 Mesh Deformation

4.3.1 Deformation methods

SU2 provides different methods useful to deform the surface mesh. As already described in section 4.2.2, a subsequent volume mesh deformation is necessary in order to adapt the deformed grid in the surrounding volume. SU2 offers a method based on linear elasticity equations able to achieve this second task. Two main strategies are adopted to deform the surface grid. The first is based on the motion of the FFD control points. The second performs simple movement of the mesh points. As it is possible to imagine, it is easier to manage the movement of tens or hundreds of FFD design points rather than several thousand of points of an entire surface mesh. By contrast, since FFD points can be translated to every position, they do not allow every kind of deformation. They can easily modify the thickness of an airfoil or the span of a wing, but it would be almost impossible to deform the whole wing to simulate the shape deformation of a wing subject to bending moment. For a situation like this, a points displacements technique can be adopted. In this thesis both methods are adopted. The choice is linked to the trade-off between above described advantages and disadvantages. FFD points deformation is used to perform the shape optimization. In this way a lower amount of design variables is exploited allowing to perform the search of the minimum with a reduced number of gradient points computation. On the other and, if the target is to perform an aero-structural coupling, mesh points motion could be the easiest way to generate structural deformation into aerodynamic mesh. Both methods are analyzed in order to check the quality of achievable deformation.

Two parameters are evaluated in order to verify the quality of the mesh: skewness and aspect ratio. The first one is defined as the difference between the shape of the cell and the shape of an equilateral cell with the same area. A high value of skewness can get the accuracy of the solution worse and destabilize the solution. A general rule provides that the maximum cell skewness of a tetrahedral mesh should be kept below the value of 0.95 and the average value should stay below 0.33 [39].

Cell aspect ratio is a parameter that measure the stretch of the cell, a general rule for tetrahedral mesh is to avoid an high number of cells with aspect ratio higher than 10. The risk consists in the possibility to overcome the recommended value of one of these maximum and average parameters for a high number of cells. This can happen after a deformation that extends the body shape. For this reason, an FFD mesh deformation that increase the value of the span and the thickness is performed in the following. Similarly, a mesh points displacement deformation is obtained to test deformation capability of SU2 routines.

4.3.2 FFD Mesh Deformation

As already explained, SU2 gives the possibility to define an FFD box around the surface mesh of one or more objects. After using SU2_DEF to define it, the same tool can be reran to deform the mesh. In the configuration file, for each design variable the direction and the magnitude of displacement must be specified. In this way it easy to obtain deformation in a specific direction such as cord, thickness and span deformation. Regarding span and thickness deformation, an analysis and a comparison of the original and deformed meshes are performed. The geometry analyzed is the ONERA M6 wing, a popular simple test case regarding a swept

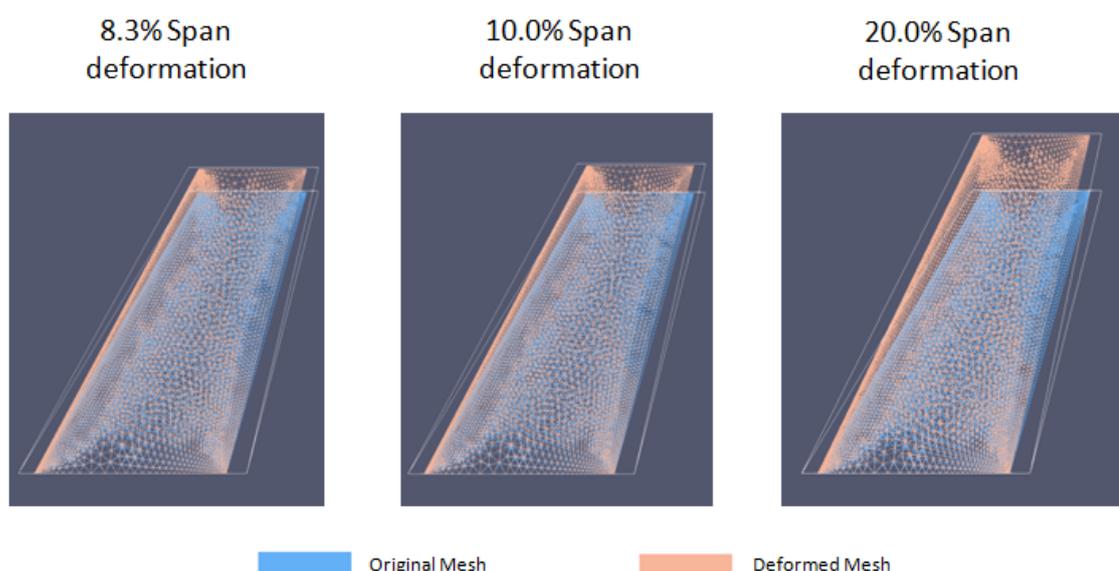


Figure 4.10: Comparison between original and deformed meshes and FFD boxes for three different magnitude of deformation of ONERA M6 wing geometry.

wing used to test CFD performance on transonic turbulent flow.

Span deformation. First, a deformation in span direction is analyzed. Three different steps with increasing magnitude of deformation are performed. Fig. 4.10 shows the comparison between original and deformed meshes and FFD boxes for three cases. An increase of approximately 8, 10 and 20 per cent with reference to original span is represented.

Tab. 4.6 represents the minimum, maximum and average value of both parameters used to analyze the meshes. It possible to observe that, except for a parameter, the limits imposed for the quality check are not exceeded. Therefore, a further analysis regarding the maximum AR, that is the only one parameter out of the recommended value, is developed. In Fig. 4.11 is showed the cell AR distribution of the mesh with

	Minimum Skewness	Maximum Skewness	Average Skewness
Original Mesh	$8.6e - 09$	0.85	0.21
8.3% span deformation	$1.26e - 5$	0.86	0.22
10% span deformation	$5.0e - 6$	0.86	0.23
20% span deformation	$1.13e - 5$	0.87	0.24
Recommended value		< 0.95	< 0.33
	Minimum AR	Maximum AR	Average AR
Original Mesh	1.0	9.98	2.15
8.3% span deformation	1.0	10.6	2.23
10% span deformation	1.0	10.7	2.25
20% span deformation	1.0	11.5	2.36
Recommended value		<10	<3

Table 4.6: Skewness and aspect ratio value computed for the meshes illustrated in Fig. 4.10. Also recommended value are indicated.

10% of deformation with respect to the original one. It is possible to notice that only 1.9 % of the cell is characterized by a value of AR higher than 10. That means that the mesh is still acceptable and usable for CFD computation. In addition, as can be noticed from the maximum value of original mesh AR, it is important to underline that the mesh under analysis is a coarse one. As demonstrated in Fig. 4.12, by increasing the number of cell, the maximum value of the aspect ratio can be dramatically reduced. The mesh in Fig. 4.12 is obtained using a reduced value of "Global Scale" as explained in section 3.2.1. The value is lowered from 1.2 to 0.75 and by consequence the number of cell is increased from 14 to 29 thousand. This lead to the possibility of generate less distorted cells and so with a reduced and acceptable value of AR.

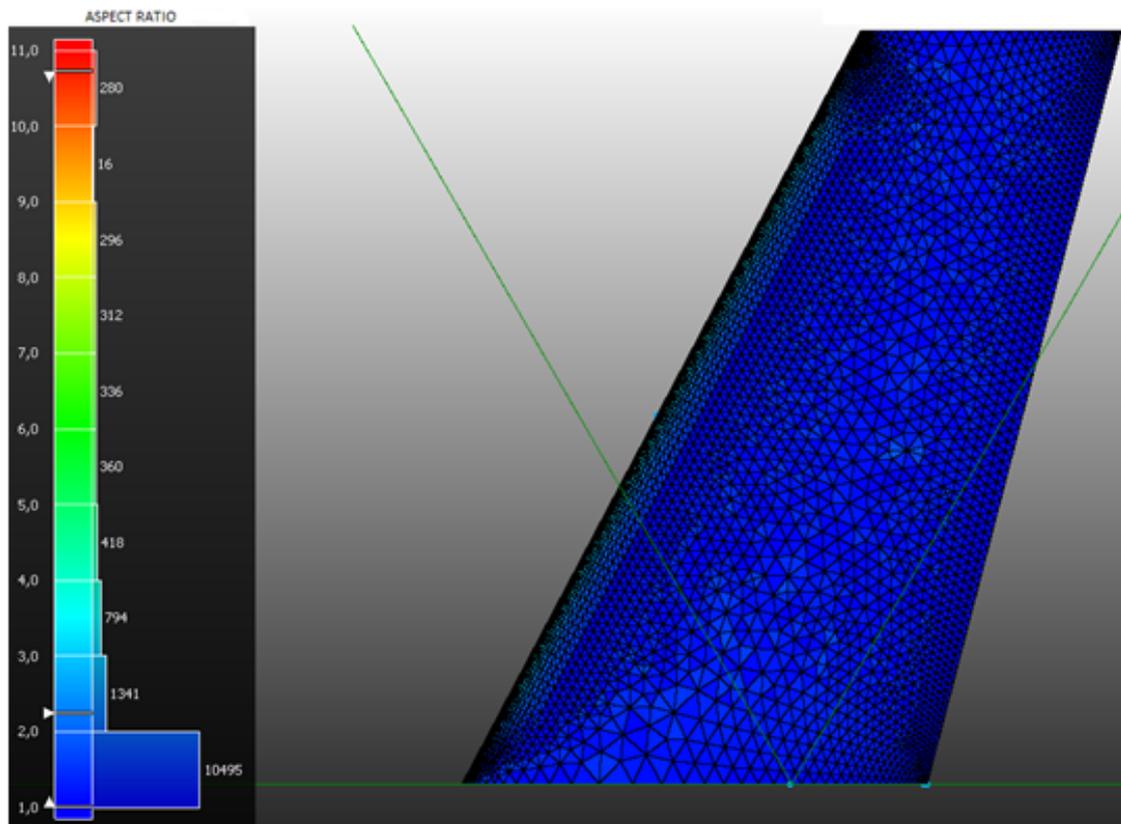


Figure 4.11: Cell Aspect Ratio distribution on mesh with 10% of deformation with respect to original one. It is possible to notice that only the 1.9% of cell is characterized by a value of the AR higher than 10.

Thickness deformation. Using the same methodology already presented for deformation in span direction, a thicker geometry mesh can be obtained. The FFD

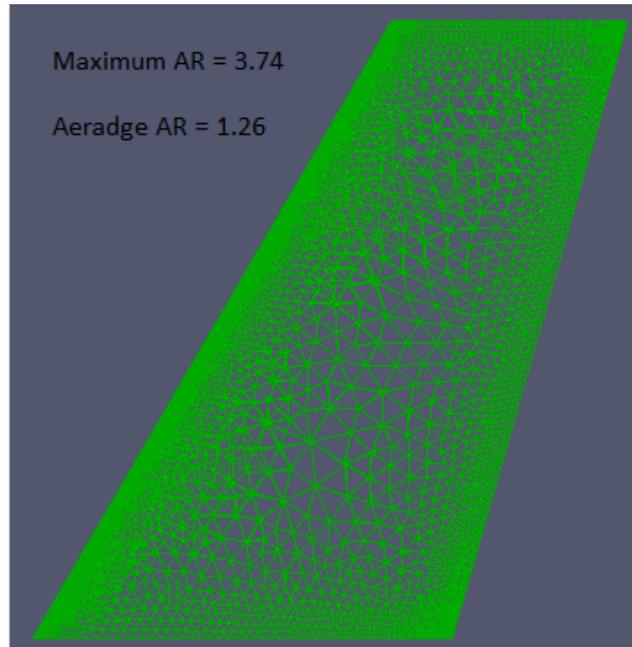


Figure 4.12: ONERA M6 wing generated using Global Scale = 0.75 as input. The number of cell composing the mesh is around 29 thousand.

21.0% Thickness deformation

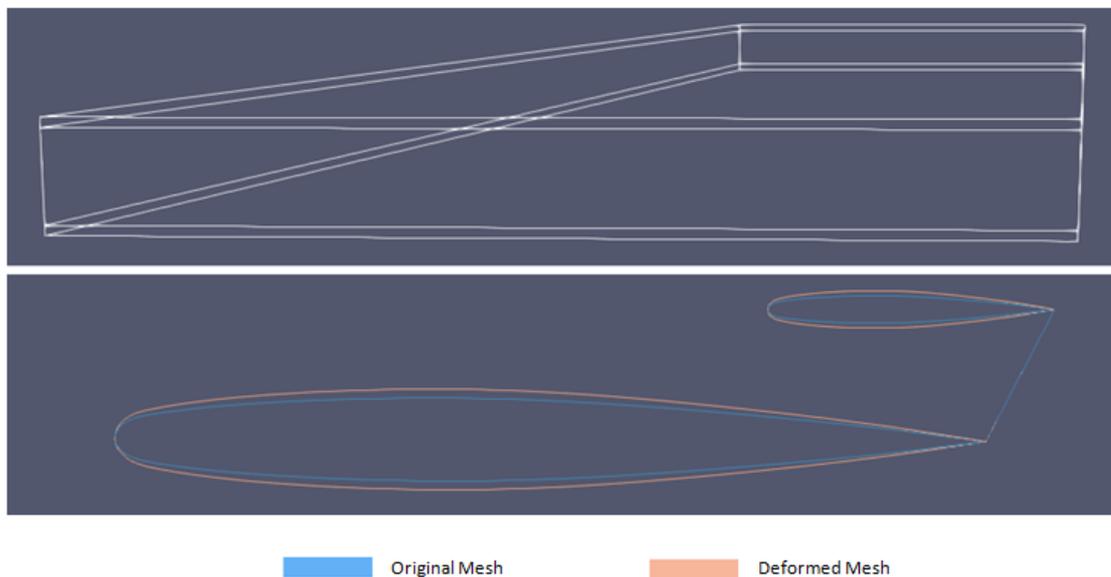


Figure 4.13: ONERA M6 wing and related FFD box deformation obtained through a symmetric displacement in z direction of FFD control points. The increase of thickness reached is around 20%.

control points are now deformed symmetrically in Z direction, generating an increase of thickness around 20% with respect to the original value. Fig. 4.13 illustrates the

deformation of root and tip airfoil shape and the translation of FFD control points. Tab.4.7 shows skewness and aspect ratio parameters of original and deformed mesh. As can be noticed, a deformation in Z direction generates a reduction in both cell skewness and aspect ratio improving the quality of the mesh.

In conclusion, it is possible to state that an value minor that 1 is required for "Global Scale" factor, leading to an higher value of mesh cell number. In this way, the mesh obtained can be defined suitable to CFD problem without generating convergence, accuracy or stability problems.

	Minimum Skewness	Maximum Skewness	Average Skewness
Original Mesh	$8.6e - 09$	0.85	0.21
21% thickness deformation	$9.6e - 10$	0.81	0.20
Recommended value		< 0.95	< 0.33
	Minimum AR	Maximum AR	Averadge AR
Original Mesh	1.0	9.98	2.15
21% thickness deformation	1.0	8.09	1.97
Recommended value		<10	<3

Table 4.7: Skewness and aspect ratio value computed for the geometry meshes illustrated in Fig. 4.13. Also recommended value are indicated.

4.3.3 Mesh Motion Points Deformation

Another deformation features that SU2 offers is the possibility to indicate for each surface point the new coordinates in which it has to be placed. As input, a simple tab-separated values file with ".dat" extension must be provided. For each line of this file, the index of the points to move and their new coordinates have to be indicated. In the configuration file is indicated the surface to which the tab-separated values file refers and to which the deformation can be performed.

Through the use of LAGRANGE tool, it is possible to deform the Optimale wing

presented in Fig 5.1 and 4.6 with the aim of representing the shape of a wing subjected to a great bending deflection. In Fig. 4.14 are presented the original and deformed surface meshes surrounded by volumetric grids. As can be noticed, a tip displacement in z direction of 1.3 meters is obtained, it corresponds to approximately 8% of wing span. So a characteristic value of wing deflection is simulated. As previously presented, in Tab. 4.8 are shown the skewness and aspect ratio value of the two surface meshes. It is possible to notice that only a slight change in aspect ratio is obtained, leading to the possibility to perform stable and accuracy computation also with deformed mesh.

In conclusion, it is possible to notice that only span deformation leads to an excessive deformation of cell mesh. In order to avoid this phenomena and obtain a mesh that satisfy the quality grid parameters only meshes with high number of cells will be computed. This can easily be applied using a "Global Scale" factor presented in section 3.2.2 minor than 1.

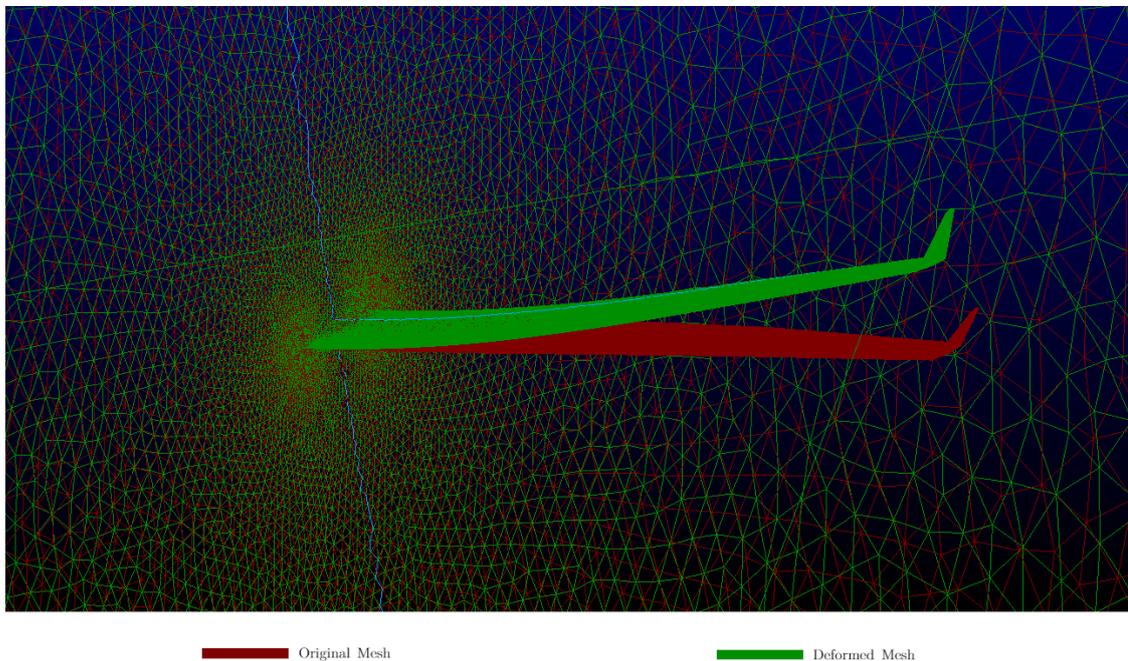


Figure 4.14: Optimale wing comparison between original and deformed shape representing the typical effect of bending moment on wing in flight. A tip displacement of 1.3 meters is reached, it corresponds to approximately 8% of wing span.

	Minimum Skewness	Maximum Skewness	Average Skewness
Original Mesh	0.0	0.90	0.23
8% tip deflection	$1.06e - 8$	0.90	0.23
Recommended value		< 0.95	< 0.33
	Minimum AR	Maximum AR	Average AR
Original Mesh	1.0	8.02	1.3
8% tip deflection	1.0	8.3	1.29
Recommended value		<10	<3

Table 4.8: Skewness and aspect ratio value computed for the geometry meshes illustrated in Fig. 4.14. Also recommended value are indicated.

4.4 Aero-Structural Analysis Workflow

4.4.1 Presentation of the Workflow

In Fig. 4.15 a workflow integrated in RCE is presented. The aim of the workflow is to obtain aero-structural equilibrium of the input configuration in one or more flow conditions. Fig. 4.15 represents the main components which make-up the workflow, a deep explanation of their task is explained in the following paragraphs. All the tools and scripts exploited in the workflow, with the exception of one which computes structural forces from aerodynamic pressure distribution, are already presented and analyzed.

As can be noticed, the equilibrium is obtained through a loop which consists in sequence of structural and aerodynamic analysis. The loops starts with the tool named "Def&Sol". It computes the first aerodynamic solution of the input mesh and from that calculates the forces distribution along the grid. After that, LAGRANGE tool must be executed. It happens through BRICS, developed at the Netherlands Aerospace Centre (NLR). It offers the possibility to share a single file on neutral server, that can be accessed by all partners. LAGRANGE tool computes the

structural deformation and gives by output a file with the new positions of all the mesh points. Because of the possibility to share a single file through BRICS, in this output file is also indicated the value of the maximum displacements between input and output meshes of LAGRANGE. "Get delta max" tool simply reads this value and passes it to the converger, together with the displacement file. After that, the converger checks if the convergence criteria, previously set, is satisfied. If not, a new iteration starts and "Def&Sol" tool deforms the aerodynamic mesh in order to rerun CFD computation with reshaped grid. When the convergence criteria is satisfied (it usually happens after three to eight iterations), a new aero-structural loop can be restarted by "Aero DOE". If it is appropriately configured, it can run subsequently multiple convergence loops with different flow field parameters. For instance, in order to compute a flexible drag polar several analysis with different angles of attack can be performed.

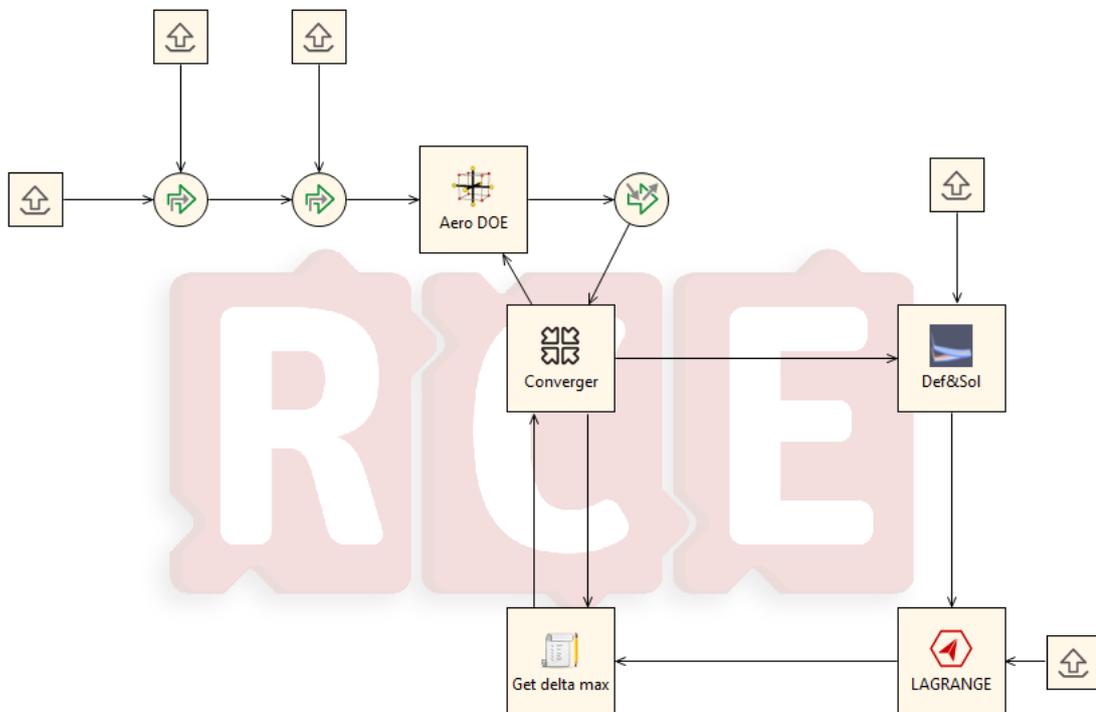


Figure 4.15: RCE visualization of the workflow built in order to obtain Aero-Structural equilibrium of a generic configuration at different flow conditions.

Input

File	Description
CPACS file	XML file that describes the geometry of the configuration under analysis. It is merged with "Toolspecific" file
Toolspecific file	XML file in which input parameters have to be inserted
Mesh file	File containing SU2 format mesh required to compute aero-structural analysis
Mesh motion file	Vtk file representing the surface mesh of the original geometry under analysis
Parameter	Description
mach_number	Set the number of mach
angle_of_attack	Set the angle of attack, in degrees
side_slip_angle	Set the sideslip angle, in degrees
free_stream_pressure	Set the value of flow-field pressure, in N/m^2
free_stream_temperature	Set the value of flow-field temperature, in Kelvin
i_origin_for_moment	X, Y and Z coordinates of the reference origin for moment computation. letter "i" must be substituted with "x", "y" and "z"
reference_length	Reference length used to compute non dimensional coefficients
reference_area	Reference area used to compute non dimensional coefficients
marker_name	Vector containing the names of the boundaries that compose the surface mesh
max_number_of_iterations	Maximum number of iteration allowed for aerodynamic solution
fixed_cl_mode	Boolean value. If True, an aerodynamic analysis at fixed value of C_L is executed
target_cl	Value of the C_L to reach in fixed C_L mode

Table 4.9: Inputs required in order to execute the aero-structural equilibrium workflow.

4.4.2 Input, Output and Configuration

In Tab. 4.9 the inputs required to execute the workflow are listed. Four files must be provided. The CPACS file, as usual, contains the geometry of the configuration under analysis and the input parameters for the tool. These input parameters are insert in the CPACS file through a merging of CAPCS and "Toolspecific", both XML file. "Toolspecific" file contains all the parameters needed to configure the aerodynamic calculation. There, it is possible to set also an analysis in C_L fixed mode useful, for instance, to compute aero-structural equilibrium in a specific flight condition such as the cruise. The mesh file and a mesh motion file of the non-deformed geometry are also required. The second input is a tab-separated values file that contains for each line the index of mesh points (coherent with mesh file) and their original Cartesian coordinates. Apart for mesh motion, all these files can be generated from the workflow presented in Fig. 3.1. This means that with the addition of a simple tool that generates the mesh motion file from the input mesh it is possible to link the workflow presented in Fig. 4.15 with the one illustrated in Fig. 3.1. In this way an automatic process that computes the aero-structural equilibrium from a CPACS file can be realized.

After the execution of the entire workflow, for each loop iteration and for each flow condition set in "Aero DOE" the output presented in Tab.4.10 are provided. Output files consist in mesh files used as input and the one generated during each iteration, also vtk and history file containing the solution data already presented in in Tab. 4.1 are provided. In addition, the vtk solution file contains data for each cell composing the mesh of that specific iteration. These data are calculated as input for LAGRANGE file. The method used for the computation is explained in section 4.4.3.

In conclusion, the two drivers of the loops can be configured in order to set the convergence parameters and the possible sequence of different aerodynamic condition for each aero-structural loop.

- The converger requires as input a control value, expressed in meters. It is referred to the maximum distance difference between the points composing the

input surface mesh and the points of the outputs surface grid for each iteration. If this value becomes lower than the control value, it means the deformation obtained during the last iteration is negligible, by consequence the inner loop is interrupted and is considered converged. Other necessary inputs are the number of subsequent iterations that must satisfy the convergence criteria and the maximum number of iteration possible for a single aero-structural equilibrium computation.

- In "Aero DOE" the number of aero-structural equilibrium computations desired must be indicated. In addition, for the majority of parameters described in Tab. 4.9 a list of multiple values can be insert in order to automatically execute more than one aero-elastic equilibrium analysis with different flow-field conditions.

Output for each iteration

File	Description
Mesh file	For each iteration, input and output meshes generated from "Def&Sol" tool are provided, they corresponds to initial and final geometries of each iteration
Surface and volume solution file	Vtk file containing the information already presented in Tab. 4.1 and the cell data listed below
History file	File containing convergence data listed in Tab. 4.1
Cell Parameter	Description
Area	Contains the area of each cell of surface mesh, in m^2
Barycentrer	Contains the geometrical barycenter Cartesian coordinates of each cell composing surface mesh, in m
Cell_Force	Contains the force vector of each cell of surface mesh in local normal direction, expressed in N
Cell_pressure_coefficient	Contains the pressure coefficient calculated for each cell of the surface mesh
Normal	Contains the directional cosines of the normal of each cell. The vector is expressed in unit norm

Table 4.10: Output files and parameters provided for each iteration after the execution of the aero-structural equilibrium workflow.

4.4.3 Aero-Structural Interface Methods

The fundamental and commonly more critical task of an aero-structural analysis is the coupling implementation. Two main tasks have to be performed. The first consists in transferring the displacements compute by structural solver to aerodynamic mesh deformation. This work can be executed thanks to the capability of the structural computation tool which gives as output the new position of each aerodynamic mesh node representing the deformed geometry. This mapping between aerodynamic and structural displacement is performed through a method based on radial basis function, described in section 3.4.2 and in [38]. In this way is possible to compute the points displacement on fluid domain u_{fluid} , knowing the structural points displacement $u_{structure}$. These two quantities are connected thanks to an interpolation matrix H , as shown in Eq. 3.10. In this way, after aero-structural grid mapping, a tab-separated value file can be generated and used as input for the mesh deformation tool described in section 4.3.3 and an aerodynamic mesh of the deformed geometry is consequently generated. The second task required to couple aerodynamic and structure analysis is the mapping of the computed pressure distribution into structural domain. The structural tool needs as input the cell parameter presented in Tab. 4.10. These data are evaluated through a Python script contained in "Def&Sol" tool of the aero-structural workflow. A description of all steps performed to compute required cell data is now presented.

1. **Get Mesh Data.** The first step consists in reading the vtk file containing the geometry and the aerodynamic solution computed. The points coordinates, the points index, the cells connectivity information and the nodal pressure coefficient are read.
2. **Get Non-Dimensional Factor.** From SU2 output the value of free-stream density and velocity are saved in order to calculate the dimensional pressure distribution on grid points. This task is executed using Eq. 4.1 where p_i represents the pressure in the grid node i and p_∞ , ρ_∞ and V_∞ correspond respectively to the free-stream pressure, density and velocity magnitude far

from the body in analysis.

$$C_{p_i} = \frac{p_i - p_\infty}{\frac{1}{2}\rho_\infty V_\infty^2} \quad (4.1)$$

3. **Get Cell Pressure Coefficient.** Then, for each cell is computed the cell pressure coefficient. This coefficient is easily calculated as the average value between all the nodal cell pressure value composing the specific cell. This can be done thanks to the connectivity information obtained in step 1. Calculation of the cell pressure through the average value of its nodes is an approximation method that lead to negligible error if the number of points is high.
4. **Get Area and Barycenter.** Through the coordinates of each cell point, the area and the geometrical barycenter can be computed.
5. **Get Cell Normal.** Next step consists in the computation of the surface norm direction for each cell. Since for surface mesh the elements can only be two-dimensional, outgoing normal is computed as the cross product between vectors representing two edges of the cell. After that, the vector is normalized dividing it by its unit norm.
6. **Get Cell Forces.** Cell forces are easily computed through Eq. 4.2 where \vec{F}_i , \vec{n}_i and A_i are respectively the force, the normal and the area of the node i . The minus sign is due to the convention for which the pressure generates positive force in the direction of the body

$$\vec{F}_i = -\vec{n}_i \cdot (p_i - p_\infty) \cdot A_i \quad (4.2)$$

7. **Write Output Data.** Finally, all data computed in the previous steps are written in order to update the vtk input file that will possess the output data presented in Tab. 4.10.

The above-described script provides also calculation for nodal force. In that case an equivalent area for each node must be computed. Nodal area is calculated as the sum of all the cell areas to which the node belongs, each one divided by number of nodes that composes the cell. In a similar manner also the normal to each node is computed. Also in this case the approximation is negligible if the mesh

is composed by an high number of nodes. Than, nodal force is easily computed as nodal pressure multiplied by nodal area and normal.

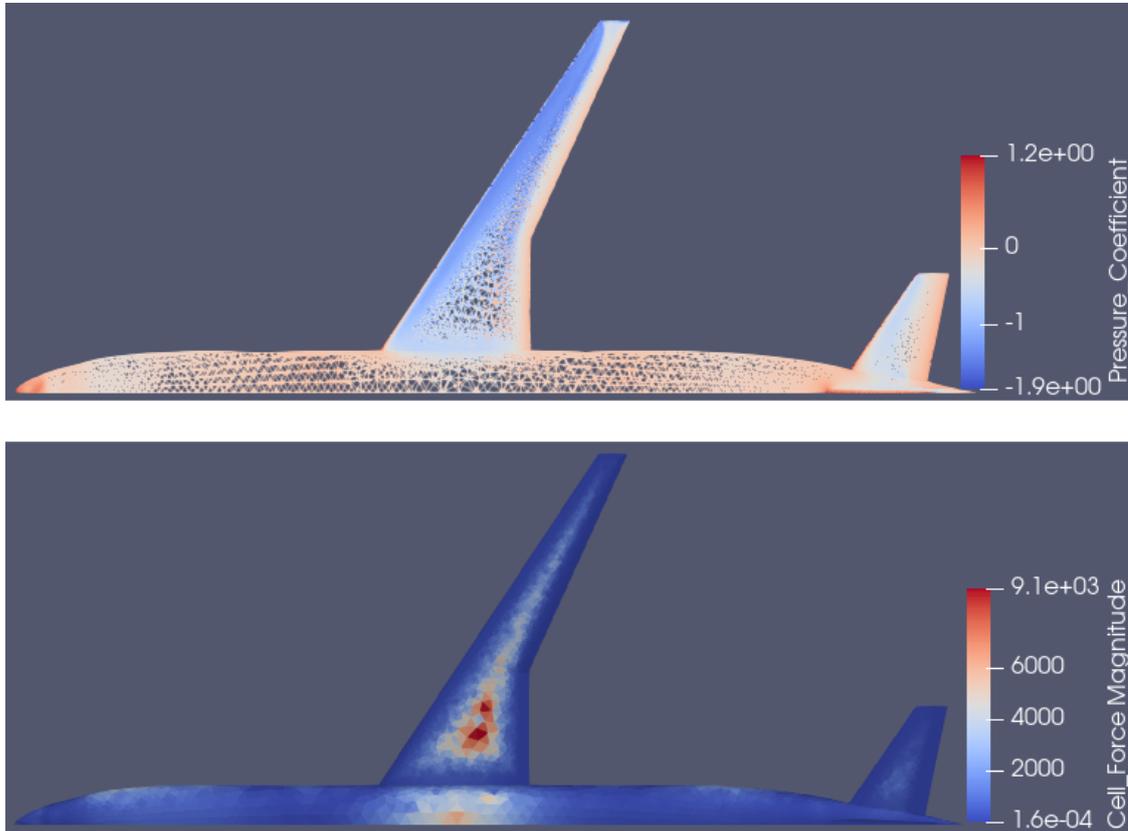


Figure 4.16: Comparison between DC-2 configuration modal pressure coefficient distribution calculated in cruise condition through SU2 and cell forces distribution subsequently calculated.

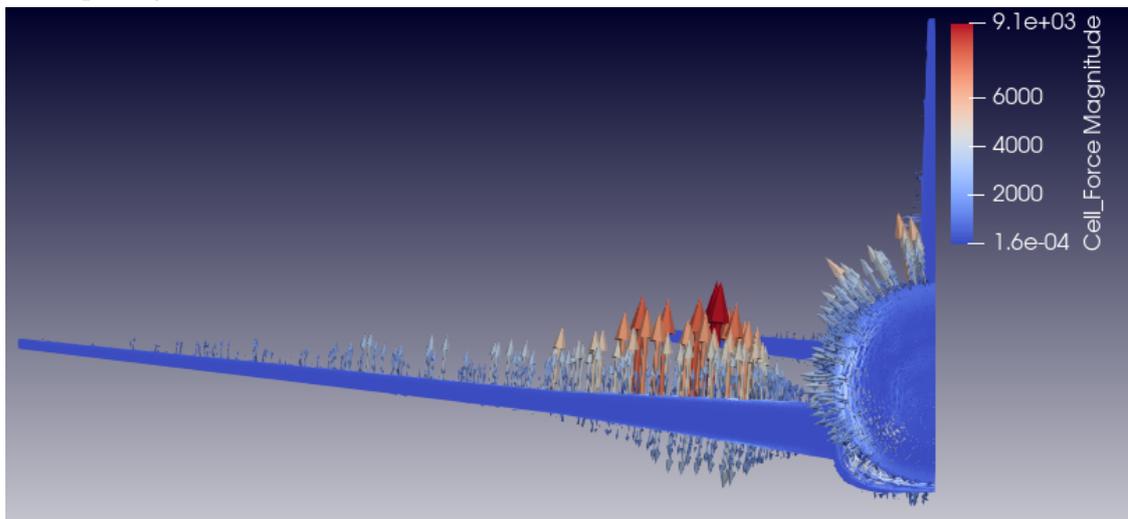


Figure 4.17: DC-2 configuration force vectors. The length of the arrow depends on the magnitude of the force, all vector are oriented towards the outgoing normal.

In Fig. 4.16 is presented an example of output provided from the already presented tool. It is possible to observe the nodal pressure coefficient distribution obtained from SU2 analysis already discussed in section 4.2.3 and the magnitude of cell forces distribution subsequently obtained from the above-described tool. As can be noticed from Fig. 4.16 and also from Eq. 4.2 the cell force value strongly depends on the area of the cell. This phenomenon does not affect the calculation of structural deformation because the total forces applied on the surface does not depend on the area of the cells. In Fig. 4.17 a sample of force vectors in surface normal direction are represented. The length of the arrows depends on the magnitude of the force. They all point toward the outgoing normal with respect to the surface.

Finally, by using the same approach that concerns Eq. 3.10, the cell forces on aerodynamic mesh can be mapped into loads at the structural model through the Eq. 3.14, justified by principle of virtual work.

4.4.4 Workflow Architecture

In Fig.4.18 the architecture of the aero-structural workflow presented in section 4.4.1 is illustrated. It is possible to observe the presence of two iterative loop, one internal with respect to the other. The internal one, illustrated with blue arrows, represents the core of the computation. It performs the coupling between aerodynamic and structural analysis. Indeed during the loop, other than the execution of CFD and CSD calculation also tool for mesh deformation, load mapping from pressure to force distribution and other coupling tasks are performed. In this way, a complete transmission of information between aerodynamic and structural mesh is provided. The external loop, illustrated with red arrows, provides the possibility to execute the just described loop several times, each one with different flow-field condition. The workflow architecture and the data-transmission among different tool is now explained.

The workflow starts with the definition of input parameters in the CPACS file. This task is performed by a XML file merger provided by RCE. It combines the input data present in LAGRANGE and aerodynamic tool specific files with the

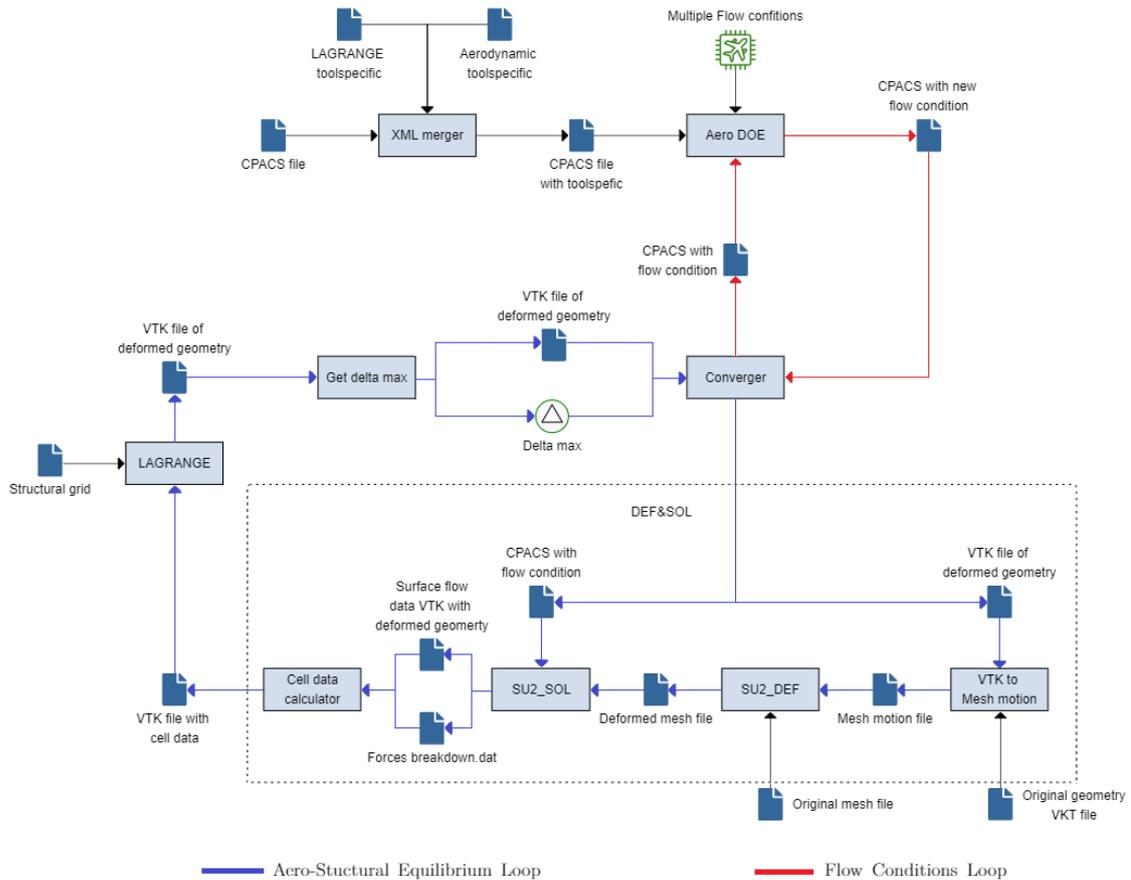


Figure 4.18: Flowchart representing the aero-structural workflow architecture. Input, output and data-transmission among tools are illustrated. It is also possible to observe the sequence of tools executed.

CPACS file of the configuration to analyze. The updated CPACS file is managed by "Aero DOE" that, thanks to another XML file merger, can update the tool specific information. Indeed, if appropriately configured, it can manage the external loop of the workflow. For each flow-field condition specified, "Aero DOE" starts the aero-structural equilibrium point research. This task performed through the internal loop which is managed by the converger. Firstly, it executes "Def&Sol" tool, providing to it the CPACS file with flow field condition to set. "Def&Sol" carries out in sequence four steps.

1. The first one, in case of the first iteration of the internal loop is performing, uses as input the VTK file that describes the original geometry. For further iteration, the VTK file is generated from LAGRANGE tool and is received as input for "Def&Sol" from the converger. From this file is generated a mesh

motion file, which is a tab-separated value file that contains the index and the Cartesian coordinates of each nodes of aerodynamic mesh.

2. Then, SU2_DEF is executed. As deeply explained in section 4.3, it is able to deform a SU2 format mesh receiving as input the tab-separated value file generated during step 1. The other fundamental input is the mesh file, that is an input of the whole workflow. SU2_DEF generates the deformed mesh containing the geometry describer in the TSV file.
3. Next step consist in CFD computation. SU2_SOL, described in section 4.2.1 is executed. As input, it requires the mesh file generated during the previous step and the flow condition. This input is provided by the converger through a CPACS file. From its tool specific node the flow parameters are extracted and then, a configuration file for SU2_SOL is generated. Among different output provided by SU2, the successively used is the VTK file containing the surface flow output data and the file containing information about the reference value used to calculate physical quantity.
4. Lastly, the tool described in section 4.4.3 calculates cell geometrical and force data need as input for LAGRANGE. The input file are the VTK with surface flow information and the file with reference value.

After the execution of "Def&Sol" tool, input file for LAGRANGE are generated. LAGRANGE tool uses as input also the structural grid of the aircraft configuration, generated and than provided as input from Descartes, as explained in section 3.4.2. After CSD computation, a VTK file is generated containing the deformed geometry under the load distribution gave as input. In this VTK file, LAGRANGE tool write also the maximum value of the distance differences between nodes composing non-deformed and deformed meshes. This value is subsequently read through "Get delta max" script that receive as input the VTK file representing the deformed geometry and returns as output the same file and the value of the maximum displacement. These two file are used as input for the converger. The value of max displacement is used to check if the desired convergence is reached. If Yes, the converger run again "Aero DOE" tool to restart a new search of aero-structural equilibrium point.

If not, a new iteration of the internal loop is performed. In the second case, the VTK file which represents the deformed geometry is proved as input for "Def&Sol" tool that can start a new CFD analysis computation.

4.5 CPACS Interface

4.5.1 Need of a CPACS Interface

As explained in section 2.3.3, a fundamental facility required in case of MDAO is a centralized data format, which aim is to simplify the exchange of data among different modules. That is the reason why each above described tool posses as input a CPACS file. Thanks to this choice, it is possible to define all the input parameters in this file that can be easily generated by external modules. By consequence, It is possible to imagine that the input for all the tools described in Chapter 4 can be output parameters coming from other tools. In the same manner, external tools can use the results generated from the modules developed for this thesis work as input for their calculation. Therefore, the generation of an interface which writes the output parameters obtained from tools described in Chapter 4 becomes necessary. Two different typologies of interface are developed. One generated with the aim to write in specific tag of the input CPACS file the aerodynamic results coming from SU2. The second, which is more complex, performs an update of the CPACS wing geometry. After deformation of the geometry, obtained for instance as results of an optimization or an aero-structural analysis, the wing geometry described in the CPACS can be partially or completely substituted in order to get hold of a CPACS file describing the new geometry. The way in which the already described tool can be subsequently executed also with a module which writes results in CPACS file is showed in Fig. 4.19. In this way, the defined workflow can be connected to different kinds of tools or workflows, giving the possibility to easily perform collaborations. Another advantage of the use of a CPACS interface is that also the tool itself can be rerun with the output CPACS file. For instance, once the geometry described by CPACS file is updated, a new mesh of that geometry can be generated. A possible use of this approach can be the generation of a gird that represent the output

geometry with different proprieties, such as the number of cells. In addition, in this way another method to generate a mesh with a modification is enabled. In the following section, the two interface tools introduced above are deeply described.

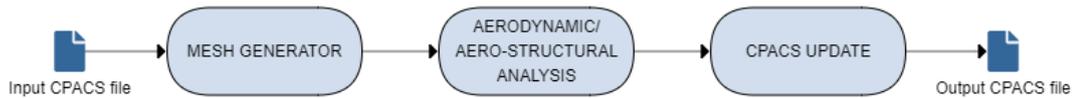


Figure 4.19: Flowchart representing the general workflow which can be defined through the execution of the tools described in Chapters 3 and 4. A CPACS file with tools results can be generated in this way.

4.5.2 Aerodynamic Results Writer

The first CPACS interface tool writes aerodynamic results obtained through SU2 in specific tag of a CPACS file. In addition, if more than one analysis are performed at different flow conditions, the tool generates several plots which shows how aerodynamic forces and moments change. These plots are generated in order to describe the behaviour of complete aircraft and also of single surfaces in which the configuration is divided. Furthermore, also a CSV file with the same information regarding the complete configuration and each single component is provided.

The scripts uses Tigl and Tixi packages in order to read and update the input CPACS file. By consequence, the first step of the tool consists in reading the version of the CPACS. As can be seen in [40], different versions of CPACS file exist. As a consequence, the proper version of Tigl and Tixi is required to read a CPACS file. After the check of the version, the right package of Tigl and Tixi is imported. Then, SU2 results are read from forces breakdown file. It is an output file generated by SU2 after aerodynamic analysis, in this file forces and moments coefficients for each component of the configuration are displayed. Obviously, this task can be performed for all possible cases: from single component to any number

of simulated components analysis. After that, a copy of the input CPACS file is generated and both flow field condition and aerodynamic coefficients of the complete configuration are written in the proper XML nodes. In Fig. 4.20 it is shown the structure of the CPACS file relative to the attribute "aeroPerformanceMap". There, aerodynamic calculation and the flow field data with which forces and moments are computed can be indicated. In Tab. 4.11 a description of each attribute inside "aeroPerformanceMap" is provided for both CPACS 2.3 and 3 version.



Figure 4.20: Comparison between CPACS 2 and 3 schema for the section which describes aircraft aerodynamic performance. It is possible to notice a slight difference between the two versions [40].

From both Fig. 4.20 and Tab. 4.11, it is possible to notice that there is a slight difference between the definition of a CPACS with version 2.3 or 3. CPACS 3 requires an additionally attribute to the path that brings to "aeroPerformanceMap". Moreover, also aerodynamic forces and momentum coefficient are defined in a different way. CPACS 2.3 schema requires that the aerodynamic forces are expressed in CPACS coordinate system, a right-handed coordinate system illustrate in Fig. 4.21. This coordinates system does not rotate with the body and not even with areodynamic flow. On the other hand, CPACS 3 structure requires that all the coefficient in "aeroPerformanceMap" attribute are related to aerodynamic system which is obtained from CPACS system through the subsequent rotation of angle of attack and angle of yaw, as illustrated in 4.21. In both cases, the coefficients

Attribute	Description	Valid for
<code>machNumber</code>	Number of Mach	CPACS 2.3 and 3
<code>angleOfAttack</code>	Angle of attack	CPACS 2.3 and 3
<code>reynoldsNumber</code>	Number of Reynolds	CPACS 2.3
<code>angleOfYaw</code>	Angle of Yaw, in degree	CPACS 2.3
<code>cfx, cfy, cfz</code>	Aerodynamic force coefficient referred to CPACS coordinate system	CPACS 2.3
<code>cmx, cmy, cmz</code>	Aerodynamic moment coefficient referred to CPACS coordinate system	CPACS 2.3
<code>altitude</code>	Altitude	CPACS 3
<code>angleOfSideslip</code>	Angle of side-slip, in degree	CPACS 3
<code>cd, cs, cl</code>	Aerodynamic force coefficient referred to aerodynamic coordinate system	CPACS 3
<code>cmd, cms, cml</code>	Aerodynamic moment coefficient referred to aerodynamic coordinate system	CPACS 3

Table 4.11: Description of the attributes that can be added in Aeroperformance one. Both the attributes allowed for CPACS 2.3 and 3 are listed

are non-dimensionalized by dynamic pressure, reference area and only for moment coefficients reference length. All these quantity, including the reference point in which the moment coefficient are computed, are specified in appropriate CPACS attribute [40]. The attribute presented in Tab. 4.11 can be insert also for multiple computation. If more than one angle of attack is insert, it means the the first value indicated in aerodynamic coefficients attribute refers to the first angle of attack and by analogy this is applied to the other. In addition, if for instance two angle of attack and two angle of yaw are insert, it means that the first two value of an aerodynamic coefficient are referred to analysis at first and second angle of attacks with the first angle of yaw. The other two coefficient will be referred to an analysis at both angle of attack with the second angle of yaw. By analogy, this criteria is applied to angle of attack, angle of yaw/side-slipe, Reynolds Number/Altitude and Number of Mach, in the order just written. In the same manner, the flow field conditions and the relative aerodynamic results can be written in another attribute. Both CPACS versions provides the possibility to generate an attribute called "toolspecific". In this way it is possible to add results without following the above described rules. As consequence, an attribute that posses more elasticity

is generated. For instance, also other aerodynamic results generated from CFD solver can be insert into this attribute.

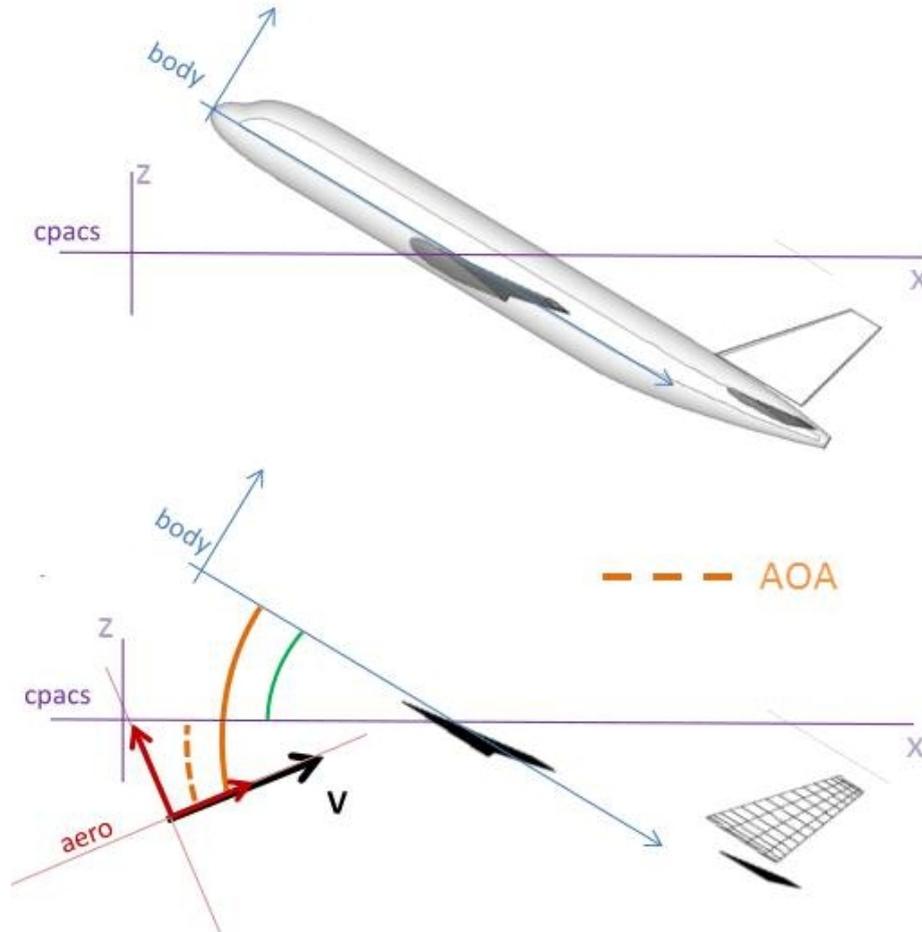


Figure 4.21: Illustration of CPACS, body and aerodynamic coordinate system taken into consideration by CPACS file. All the reference systems are right-handed coordinate systems [40].

Input and Output. Since the tool is written in order to be executed after SU2 analysis, it is able to read the parameters described in Tab. 3.3.2 using only those that are needed. They are the flow field condition indicated in Tab. 4.11 and further information regarding the wing and fuselage simulated component. In addition, the output directory of SU2 computation must be provided, otherwise it is enough only the forces breakdown file. The outputs can be divided into three typologies. Obviously, the updated CPACS file with all the addition above explained is provided. In addition, a folder named "PoltsResults" is generated. This folder contains the other two kind of output. The first is a CSV file which contains all the aerodynamic

results previously described. The first section of the file is referred to the total aircraft, the other are referred to solution for each single component of the aircraft. Each section contains the forces and moment coefficients trend with respect to the angle attack. This trend is provided for every angle of side-slip, Reynolds number and Mach number. In "PlotResults" other two folder are generated, one dedicated to the complete configuration and one to the different single components of that configurations. Each folder contains the plot of lift and pitch moment coefficient with respect to the angle of attack, the plot of the side-slip force coefficient with respect to the angle of side-slip and the polar drag plot. Each of these plots are provided for all the aerodynamic condition calculated. In conclusion, in the folder containing data referred to the single component multiple plots are provided containing the same information presents in the complete configuration folder, with a breakdown which indicates the coefficient contribution for each surface taken into consideration.

4.5.3 CPACS Geometry Update

Other than the addition of CFD results in CPACS file, useful for subsequent analysis, another important feature developed in order to completely generate a connection between the tool previously described and the CPACS file is the possibility to modify the geometry description of the input CPACS file. By consequence the entire geometry can be provided as output after an optimization or an aero-structural analysis, enabling the possibility of a complete output data transmission to other tools or partners through the centralized data format.

As explained in CPACS documentation [40], the geometry of each component of the configuration is defined through different sections that can be placed, scaled, rotated and translated in a 3 dimensional space. Usually, they are defined using at least two different slices which compose a wing or a fuselage. CPACS format gives also the possibility to define more elements in a single section. For instance, more than one element in a section can be used in case of a wing with a step or a discontinuity. For each element, a two dimensional airfoil is defined. A proper attribute contains the Cartesian coordinates of each airfoil; in the element definition there is an attribute

used to link a defined airfoil shape to the element. Also the elements can be scaled, translated and rotated. Fig. 4.22 represents a possible way of wing geometry definition. Different sections are generated and positioned in three-dimensional space, for each section at least a segment must be defined and positioned.

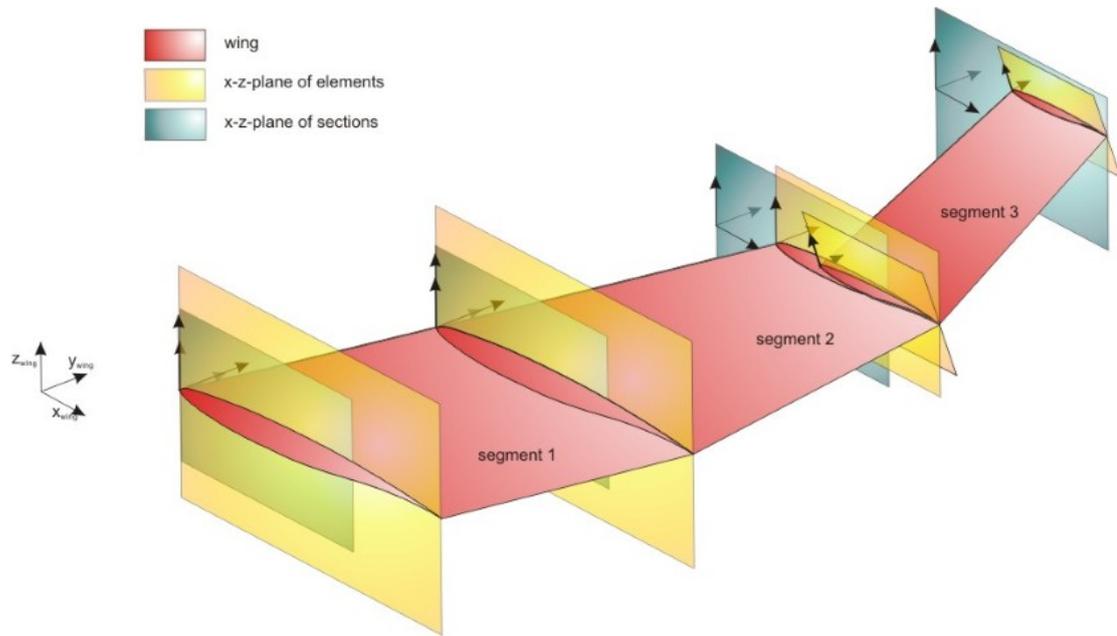


Figure 4.22: Possible definition of the wing geometry through sections, elements and segments. Wing, section and element reference system are illustrated [40].

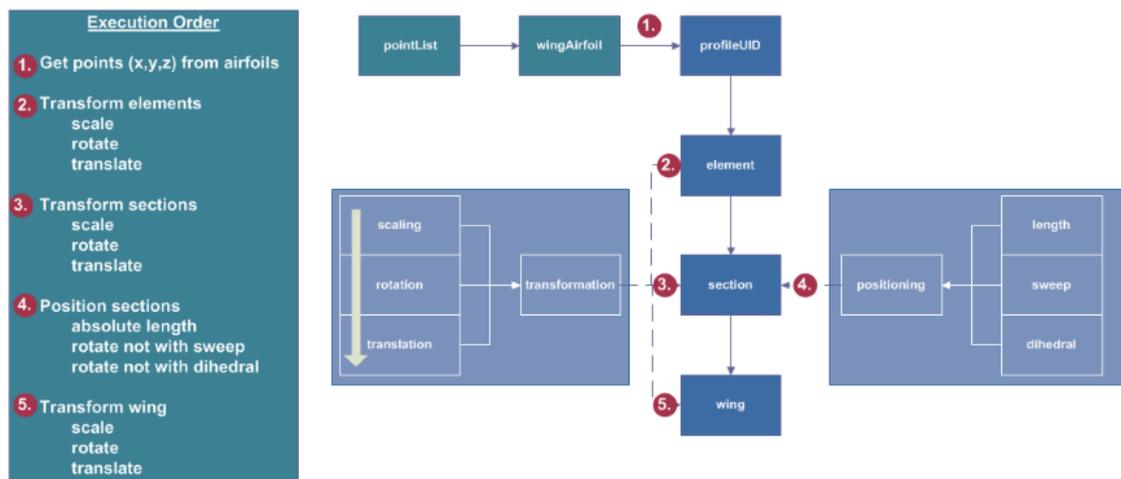


Figure 4.23: Execution order of the steps carried on by CPACS to generate a general geometry [40].

In Fig. 4.22 it is also possible to observe how different elements are linked to each other. For this task, a segment which connect two different elements must

be defined. In addition, also a link between section can be generated, it is called positioning. Actually, the positioning is a different way to locate in the space a section. Indeed, thanks to this attribute, it is possible to define how two linked sections must be connected. For instance, through the positioning attribute it is possible to define a distance between two sections and also a local sweep or a dihedral wing angle. These parameters are referred to the section reference system. Fig. 4.23 illustrates the step executed by CPACS for the definition of a generic geometry. Each profile is generated from the related two-dimensional points list. Indeed for each element, an associated airfoil must be specified. Then, the element is scaled, rotated and translated. The same happens later to the sections and then, they can be fatherly displaced through distance, sweep and dihedral parameters of the attribute positioning. In conclusion, the entire wing can be scaled, translated and rotated. It is important to understand the order of the execution in order to know if a parameter will affect or not an element or a section position.

Code Architecture. The aim of geometry update tool is to generate a CPACS file, starting from the original one, that can represent a deformed wing shape. This task is performed in three different steps. The first regards the acquisition of airfoil coordinates from input file or from SU2 output folder. The second phase regards the modification of these data in order to obtain airfoil points compatible with CPACS documentation without deform the airfoil shape. The third regards the modification of input CPACS file in order to add the wing section described only through the new original airfoil points coordinates. In Fig. 4.24 the architecture of this tool is illustrated. It is possible to notice the three different and subsequent steps to perform and also how data obtained form input file are modified and managed. In addition, input and output data are indicated.

As already said, the code is written in order to be run standalone or immediately after a wing optimization performed with SU2. For this reason, the first step consists in the research of the SU2 output file in which the wing geometry is described. As deeply explained in section 4.2.2, a generic optimization can execute number of iterations that is not known since the optimization is performed. By

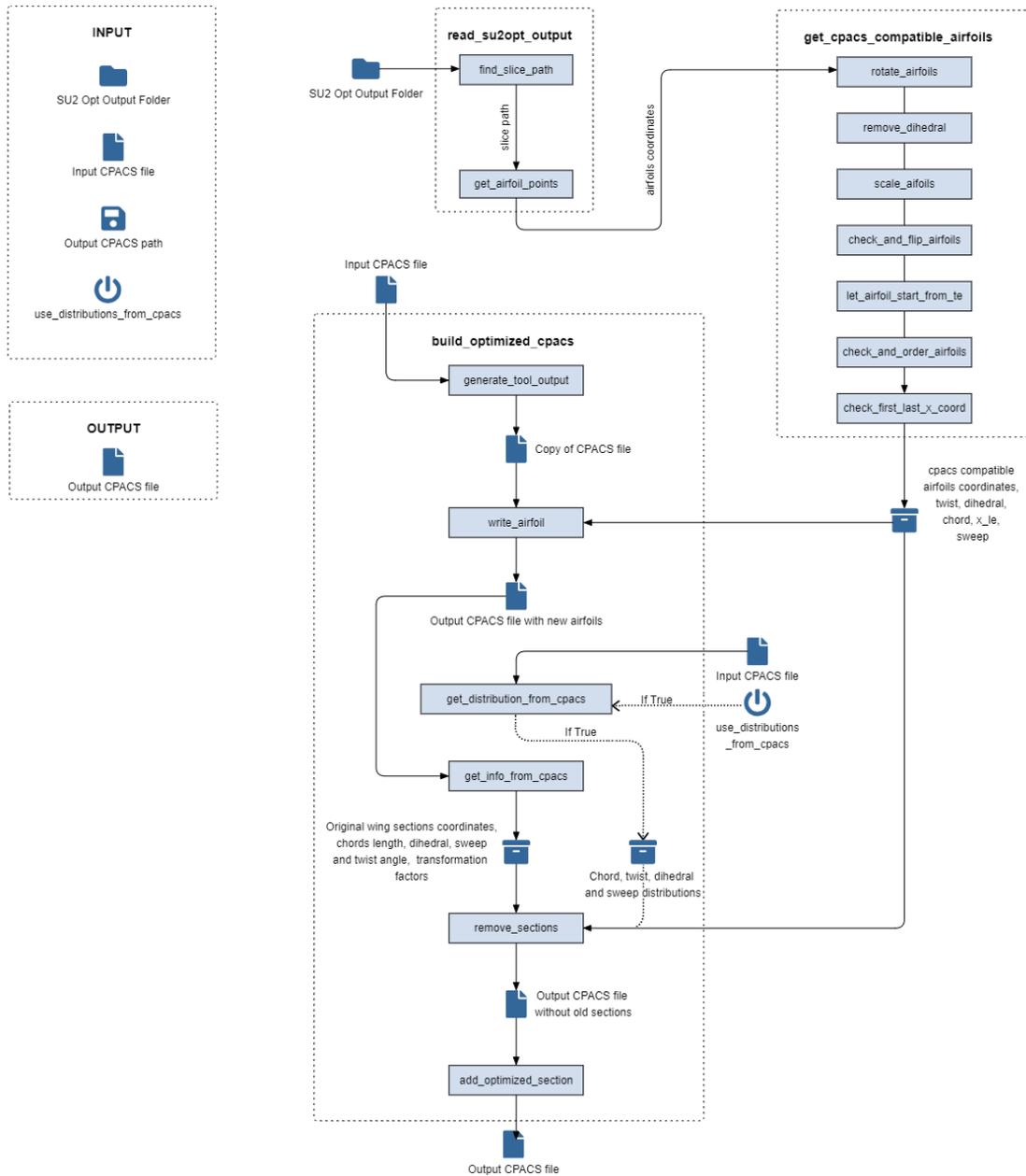


Figure 4.24: Architecture of CPCAS Geometry Update tool. It is possible to notice that the geometry update can be subdivided in three different steps: airfoil points reading, airfoil point modification and CPACS file update.

consequence the folder containing the geometry information of the last iteration must be found and then, from that folder, the file containing the points of wing slices is read. This file contains the points coordinates of different sections. These are the three-dimensional CPACS coordinates of the wing slices indicated in the SU2 configuration file. Therefore, it is also possible to generate this file after the

execution of SU2_GEO, as explained in section 4.2.2. The following step regards the adjustment of this points coordinates. As matter of fact, this points have to be added in a CPACS file. As required from CPACS documentation [40], in the point list attribute the coordinates must belong to a non-dimensional airfoil with an horizontal chord which x coordinates start from 0 and end to 1. As can be imagined, the properties of original airfoil lost during this process will be added in the wing section properties as explained in Fig. 4.23. It means that data such as sweep, twist and dihedral angle must be calculated and subsequently used for three-dimensional wing definition. In addition, also CPACS documentation provides some suggestion for the airfoil definition that are followed thanks to different check and eventual modification to which airfoil points can be subjected. In the following all the modification and check subsequently made on airfoil points coordinates are described. In Fig. 4.25, the way in which the wing shape changes before and after the execution of the first three steps is represent. A twisted swept wing with an high dihedral angle and ten sections are chosen to best illustrates the effect of these steps.

1. **Airfoil rotation.** The chord of the airfoil is made horizontal through a rigid rotation with respect to the trailing edge. During this step, the twist angle of each section is calculated and the leading edges coordinates are stored.
2. **Dihedral removal.** All the section are translated in order to have their z leading edge coordinate equal to zero. The value of dihedral angle is calculated for each couple of sections.
3. **Airfoil scaling.** The airfoils coordinates are scaled in order to have a chord with unit length. Also the z coordinates are scaled with respect to the chord because the shape of the airfoil must remain unchanged. During this phase, the chord length, the sweep angle and the x coordinates of leading edge position are calculated and stored.
4. **Airfoil flip.** CPCAS documentation provides that the airfoils must be defined first along their lower side and then along the upper one. If this does not happen, the airfoil coordinates are flipped.

5. **Starting point definition.** The first coordinates of the airfoils should belong to their leading edge points. By consequence, the script checks where the leading edge points are located and removes all the previous defined points.
6. **Airfoil ordering.** Another requirement coming from CPACS documentation is that the airfoil coordinates should be mathematically ordered. It means that the x coordinates of both the lower and upper part of the airfoil must be ordered. The script check if this requirement is satisfied and if necessary, moves the points out of order.
7. **First and last points definition.** The last step consists in checking if the first and last points have the same x coordinates. This is another requirement coming from CPACS documentation. If it is not satisfied a new point is added in the lower or upper part of the airfoil in such manner that the point will still be mathematically ordered.

Once the airfoil points coordinates are appropriately modified in a CPACS coherent way, it is possible to pass to the third phase: the CPACS update. Also this phase is composed by more steps. As illustrated in Fig. 4.24, the first step consists in the generation of a new CAPCS file. It is a copy of the original one and is the file that from now on will be modified. Then, the airfoil points are added in the CPACS file. The points coordinates taken into consideration are the one obtained after the seven steps above described. They must be two-dimensional entities, by consequence only x and z airfoils coordinates are considered. The shape of the airfoils already present in CPACS file remains unaltered, the new airfoils are added without removing the previous one. In this way both the old and the new airfoils can be used for wing definition.

At this point, a preliminary phase which aim is to obtain data from the original wing is developed. As already said, the aim of the code is to add the wing geometry represented through the wing section input points. These points can represent the entire wing or just a part of it. In the second case, it is fundamental to be aware of the the original wing geometry characteristic. Only in this way it will be possible to update only a part of a wing adding it within the old geometry. By consequence, the

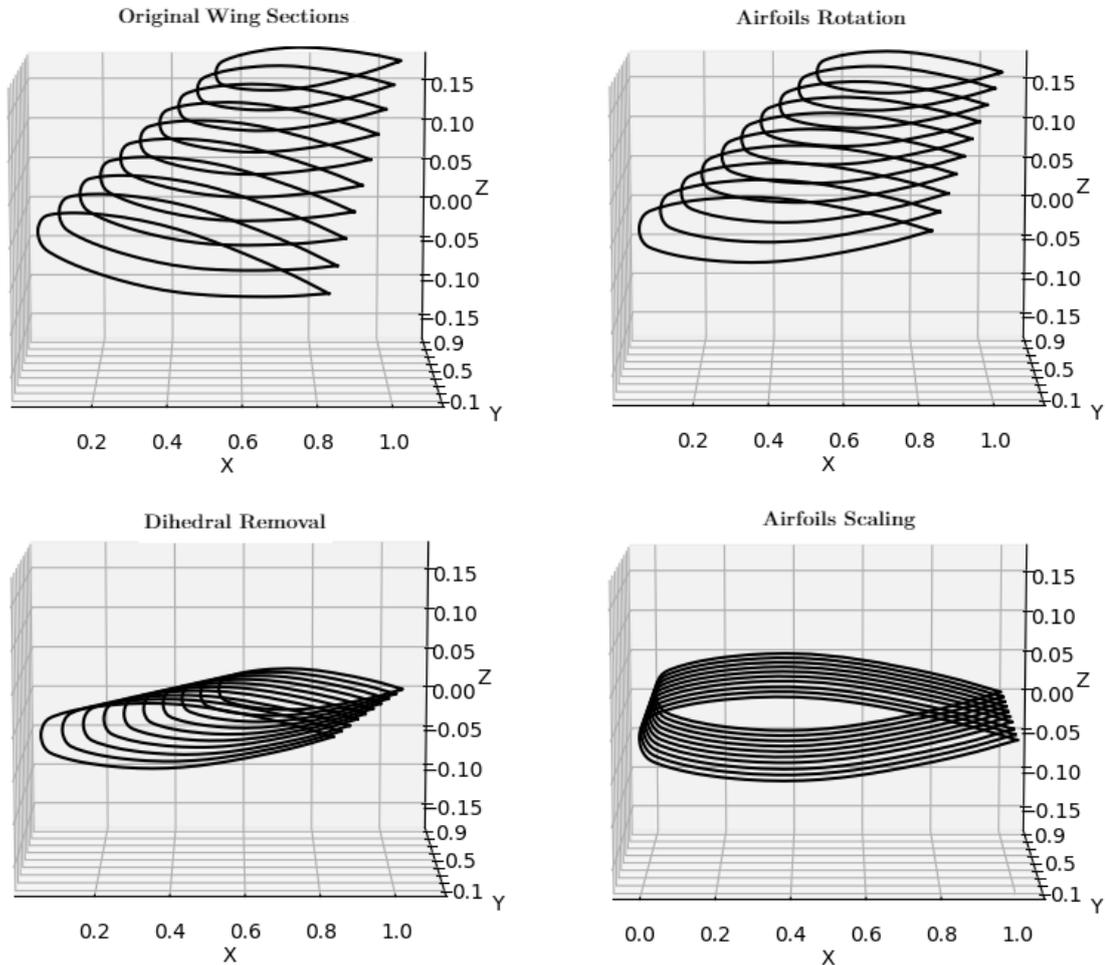


Figure 4.25: Representation of a twisted swept with non-zero dihedral wing sections modification before and after the first three normalization steps performed by CPACS geometry update tool.

next step consists in the acquisition of original wing features through the original CPACS file, this task is performed by the function "get_info_from_cpacs". The output of this function consists in different information useful for the update of the wing. They are the leading edge coordinates of the sections which compose the original wing, three-dimensional information such as the chords length, twist, dihedral and sweep angle between sections and the transformation factor of the wing segment and section (See Fig. 4.23). All data needed to update the geometry are calculated at this point. In addition, if required as input, it is possible to calculate the wing distribution already computed from the original airfoil points coordinates in another way. If specified, they can be calculated through the original CPACS

file and overwritten to the previous one. Information such as chord, dihedral, twist or sweep angle distribution could be unchanged during the wing deformation. By consequence this can be a good debugging strategy, or just a way to compare both the geometry represented by CPACS and wing slices file. Finally, the CPACS geometry update can be performed. The task is carried out through two different steps. The first one consists in removing the wing section which have to be substituted. This task is performed thanks to the information previously obtained from input CPACS and wing slice points. A comparison between the original and the new wing sections position is made, in this way the code recognize which are the original sections which are located between the minimum and maximum span coordinate of the new section. These sections and all their relative segment and positioning are removed from CPACS file. For instance, in the case illustrated in Fig. 4.26 the first and the last sections which compose the wing remain the same before and after the geometry update. This happens because the new wing geometry does not represent all the wing but only a part which starts after the first original wing section and terminates before the last one. Last step consists in the addition of the new sections. As already said, the script checks if these section have to be added within the not removed original one. If yes, a connection between original and new sections must be provided. In this case the script calculates, thanks to previous information stored, the connection information needed such as distance, sweep, dihedral angle between the two sections. After that, all the segments, sections and positioning of the new geometry are generated. The final output consists in a new CPACS file representing the new geometry. The code is able to create this output with any number of new sections provided as input. Fig. 4.26 represents two CPACS file geometry, the left one is used as input in order to obtain the right geometry. The 25 wing slices points coordinates represent the solution of the optimization performed on Optimale configuration obtained in section 4.2.3. The outline of both wing shapes is similar cause this case represents a wing airfoil optimization without big deformations. That is the reason why such an high number of slices is used. In this way, an high number of optimized airfoils is represented by the new CPACS file.

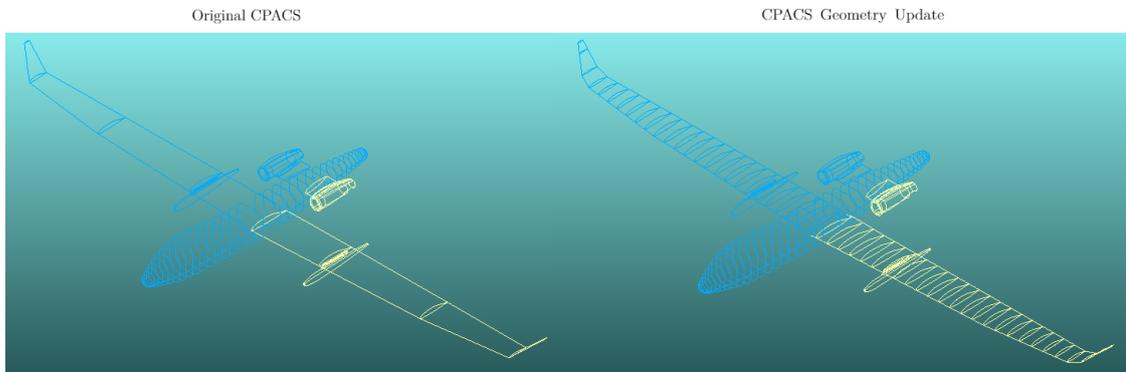


Figure 4.26: Comparison between CPACS original and updated geometry of Optimale configuration. The right geometry represents 25 slices obtained after the wing shape optimization described in section 4.2.3.

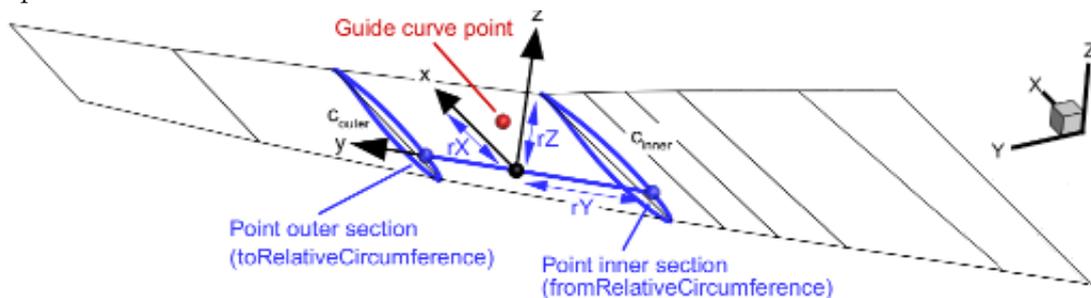


Figure 4.27: Representation of points coordinates which have to be insert to generate a CPACS guide curve, a way to define a connection between two wing sections. Three coordinates for each guide curve point must be provided and a curve that interpolates the wing shape passing through these points is generated [40].

CPACS 3.0 Update. In case that the input CPACS file version is higher or equal than 3, the workflow performed to obtain a file with the new geometry has some differences. Actually, the third version of CPACS file poses a new way of definition for three-dimensional geometries. A guide curve can be defined in order to generate a connection between two different sections. If specified, this curve can pass through one or more points with different grade of interpolation and continuity. By consequence it can generate a non-straight connection between two section, giving the possibility of generate smooth connection. In Fig. 4.27 the coordinates of the points that define this curve are illustrated. If the input

CPACS file version is equal or higher than three, the script strategies provides the use of half number of wing slices to generate the wing sections and the other half to define the guide curves points for the connection of those sections. The other steps of the code workflow remain unaltered.

5

Application of Collaborative Aero-Structural Tools

Contents

5.1	Test Case: Optimale Configuration.	95
5.1.1	Aircraft configuration and TLAR	96
5.1.2	Aerodynamic Mesh definition	97
5.1.3	Structural Mesh definition	101
5.2	Aero-Structural Equilibrium in Cruise	102
5.2.1	Mesh and Input Data for Flexible Cruise	102
5.2.2	Flexible Cruise Results	104
5.3	Flexible Drag Polar	107
5.3.1	Mesh and Input Data for Flexible Wing Polar	107
5.3.2	Flexible and Rigid Wing Polar Results	108
5.3.3	Flexible and Rigid Optimale Polar	112
5.4	CPACS Output Generation	115

5.1 Test Case: Optimale Configuration.

After the definition of tools and workflows able to perform in an automatic way mesh generation, aerodynamic analysis, CFD optimization and aero-structural analysis, a test case is now presented in order to validate the codes potential. A high AR wing aircraft configuration is taken under study in order to emphasize the aero-structural tools capability. In the following sections a presentation of the test case configuration and TLAR is provided, then a brief description of aerodynamic and structural meshes is presented and finally the results from the application of tools presented in Ch. 4 are shown.

5.1.1 Aircraft configuration and TLAR

The baseline of the analysis shown in this Chapter is the OPTIMALE configuration developed during the German AeroStruct research project. "MALE" indicates that is a Medium Altitude Long Endurance aircraft. A representation of its geometry described through CPACS file is obtained through TiG1 and shown in Fig. 5.1. Two reference missions are defined for this configuration. The first consist in a simple transfer mission which consists of takeoff, climb, cruise, descent and landing segments. The second one is a surveillance mission. In Fig. 5.2 are illustrated the segments which compose this mission. The main difference with the transfer mission is the loitering phase added during the cruise phase. In that segment the aircraft remains more or less in the same position at low altitude to gather surveillance data. Tab. 5.1 provides TLAR of this concept for both missions [41].

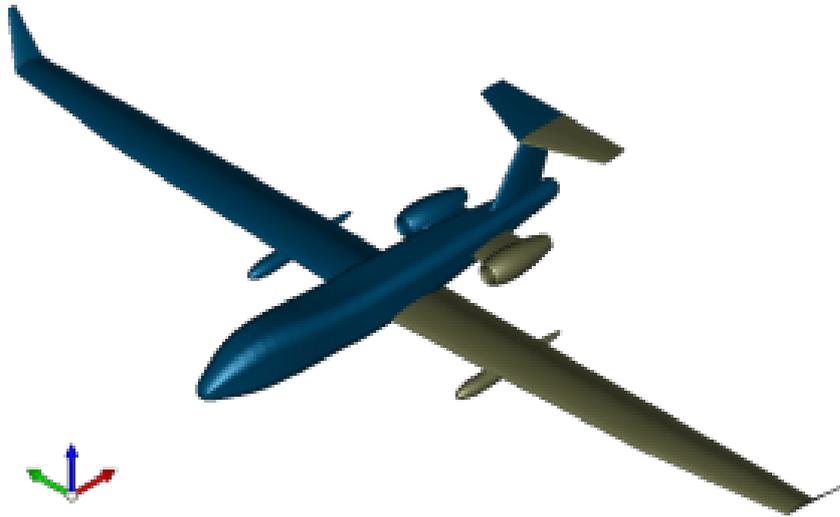


Figure 5.1: Optimale CPACS configuration represented through TiG1 viewer.

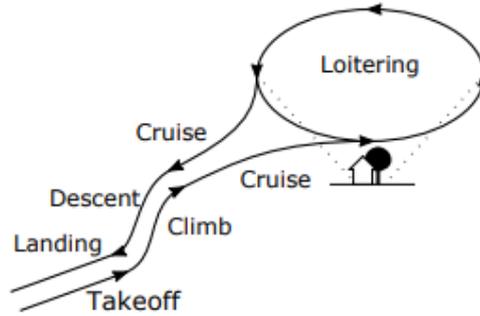


Figure 5.2: Optimale surveillance mission. Representation of mission segments.

Requirements	
Cruise altitude above civil transport	$> 15 \text{ km}$
Loitering altitude	$7500 \text{ m} - 13500 \text{ m}$
Range	$> 12\,000 \text{ km}$
Endurance	$> 20 \text{ h}$
Runways length	2500 m
Cruise speed	Mach Number 0.5 (150 m/s)
Dive speed	Mach Number 0.6 (180 m/s)
Landing speed	Mach Number 0.16 (55 m/s)
Payload weight	800 kg
Take-off weight	10000 kg
Payload volume	4 m^3
Payload power consumption	10 kW
Roll rate	60 deg/s
Rate of descent	4 m/s
Climb rate	160 m/min
Two external fuel tanks	
Electric powered hydraulic system	
SatCom Communication system	

Table 5.1: OPTIMALE Top Level Aircraft Requirements for transfer and surveillance missions.

5.1.2 Aerodynamic Mesh definition

In order to choose the input mesh for subsequent CFD analysis, a mesh sensitivity study is performed. The aim of this study is to understand the changes of aerodynamic results with reference to the number of surface mesh cells. A trade-off between computational time and accuracy is taken into account in order to obtain acceptable results in not very high time. The importance of this choice

is emphasized by the need of execute a certain number of iterations in order to reach convergence in aero-elastic computation. Usually three up to eight different CFD computation are required to obtain a great displacement convergence. If the purpose is to obtain results at different angles of attack it means that the number of required iteration becomes quite high.

The parameter taken into consideration to indicate the density of the mesh is called "avgDs1". It indicates the average cell edge length of the mesh obtained at step 3 of the mesh generation tool presented in section 3.2.2. The final average length of the cell is higher than the value of "avgDs1" but depends on it because during the following steps the mesh is refined. The strategy adopted consists in the generation of different meshes and in subsequent aerodynamic calculation with the same flow field input. In order to reduce the computational time of the test, only the isolated wing mesh is analyzed. Wing aerodynamic coefficients obtained through meshes with different node density are compared. The flow field input of CFD analysis are indicated in Tab.5.2. An analysis at low Mach number without complication such as shock waves or similar is performed. In this way a low computational time for reach convergence is obtained.

Six different meshes are obtained in order to compare the wing aerodynamic coefficients. As already explained in section 3.2.1, the input parameter for mesh generation tool used to control the size of the mesh cell is "Global Scale". Mesh

Parameter	Value
Physical problem	Euler
Regime type	Compressible
Mach number	0.2
Angle of attack	0°
Side-slip angle	0°
Pressure	11597.3 Pa (Altitude 15Km)
Temperature	216.5 K (Altitude 15Km)

Table 5.2: Input data for CFD analysis computed with different mesh in order to analyse the effect of the mesh node density on aerodynamic coefficients.

deformation analysis described in section 4.3 suggest that its value should not overcome unitary value. For this reason a range that starts from 0.4 and ends to 1.0 value of "Global Scale" is taken into consideration. Fig. 5.3 shows the lift and drag coefficients trend with reference to "avgDs1" parameter. Also "Global Scale" parameter used as input to obtain the meshes is indicated. It is possible to observe the changes of these coefficient value is very low, it corresponds to 2.6% for drag

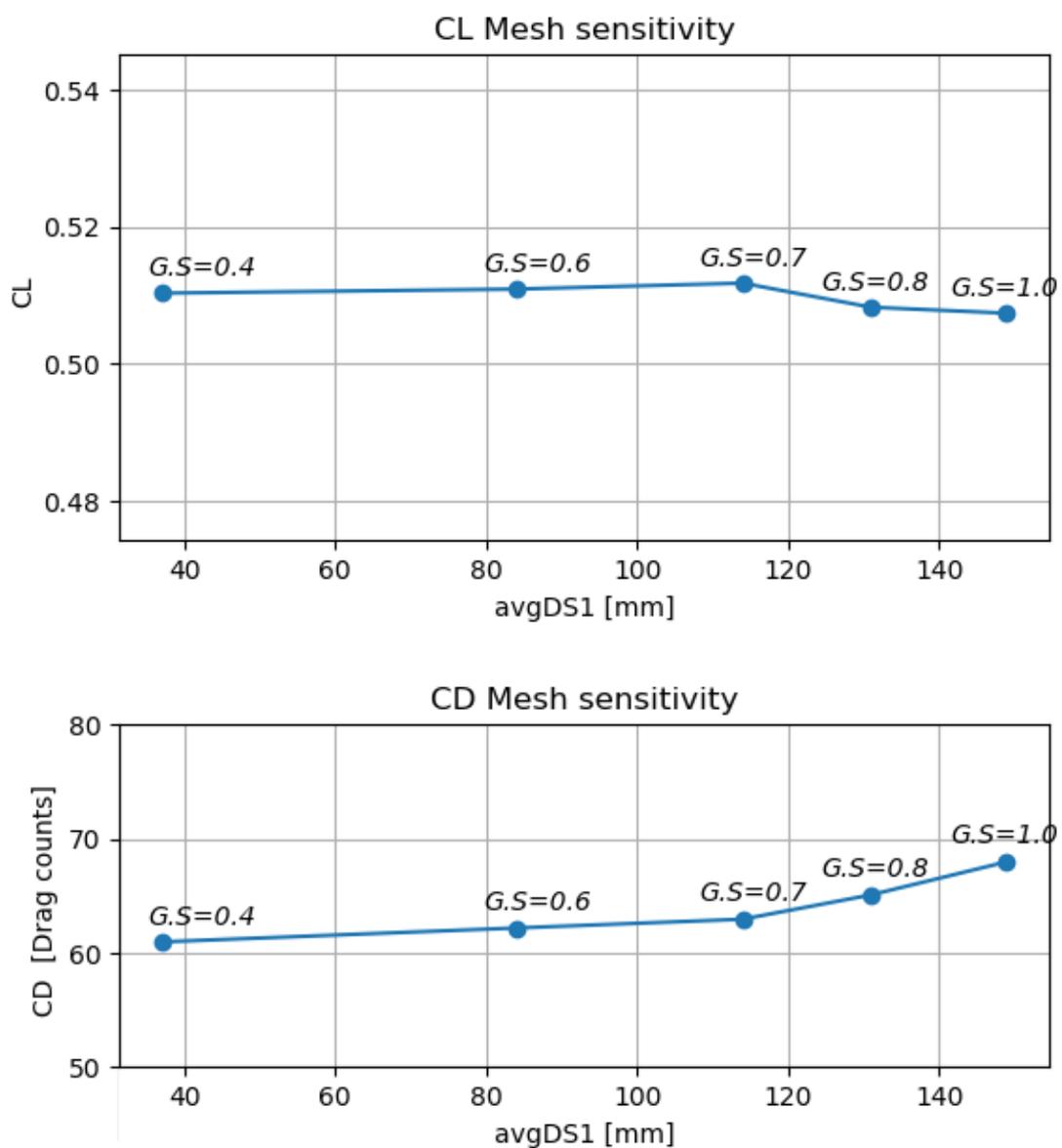


Figure 5.3: Lift and drag coefficients trend with reference to the initial value of mesh cell average edge length. Also the parameter "Global Scale" used as input for the generation of the mesh is indicated.

coefficient and 0.57% for lift coefficient computation. It is also possible to notice that the slope of the two curves is reduced when a "Global Scale" factor lower than 0.8 is used. These two aspect lead to the conclusion the a mesh with "Global Scale" equal to 0.7 has a sufficient number of cells for subsequent computation. These choice is even more justified by data illustrated in Tab. 5.4. For each mesh generated the number of cell, referred to volumetric mesh, and the computational time spent to obtain results showed in Fig. 5.3 are illustrated. The analysis is performed with a machine equipped with four processors. It is possible to see that the computational time raises to very high values in case of too high number of cells. For this reason the choice fell on the mesh characterized by a "Global Scale" parameter equal to 0.7. In conclusion, a summary of the main parameter characterizing the chosen mesh is shown in Tab. 5.4.

Global Scale	Number of cells	Computational time
1.0	1.79E6	18.5 min
0.8	1.89E6	22.48 min
0.7	2.15E6	26.54 min
0.6	2.37E6	34.3 min
0.4	3.14E6	52.24 min

Table 5.3: Number of cells and computational time spent for compute results shown in Fig. 5.3. The machine exploited posses four number of processors.

Parameter	Value
Global Scale	0.7
Volume mesh diameter	709.2m
Number or cells	2.15 millions
Number of nodes	438313
Average cell AR	1.32
Average cell skewness	0.29

Table 5.4: Main feature of aerodynamic mesh chosen for subsequent calculation. The mesh is one those used as input to obtain results plotted in Fig. 5.3.

5.1.3 Structural Mesh definition

The structural model is generated from the CPACS database and provided by DLR. The wing structure is composed by a wing-box with front, rear and also auxiliary spars located along span-wise direction. These spars are supported by several ribs in chord-wise direction. In addition, the stability of the wing-box is provided by a uniform distribution of one-dimensional stringer elements. The T-tail empennage structural configuration is basically the same of the wing. The fuselage

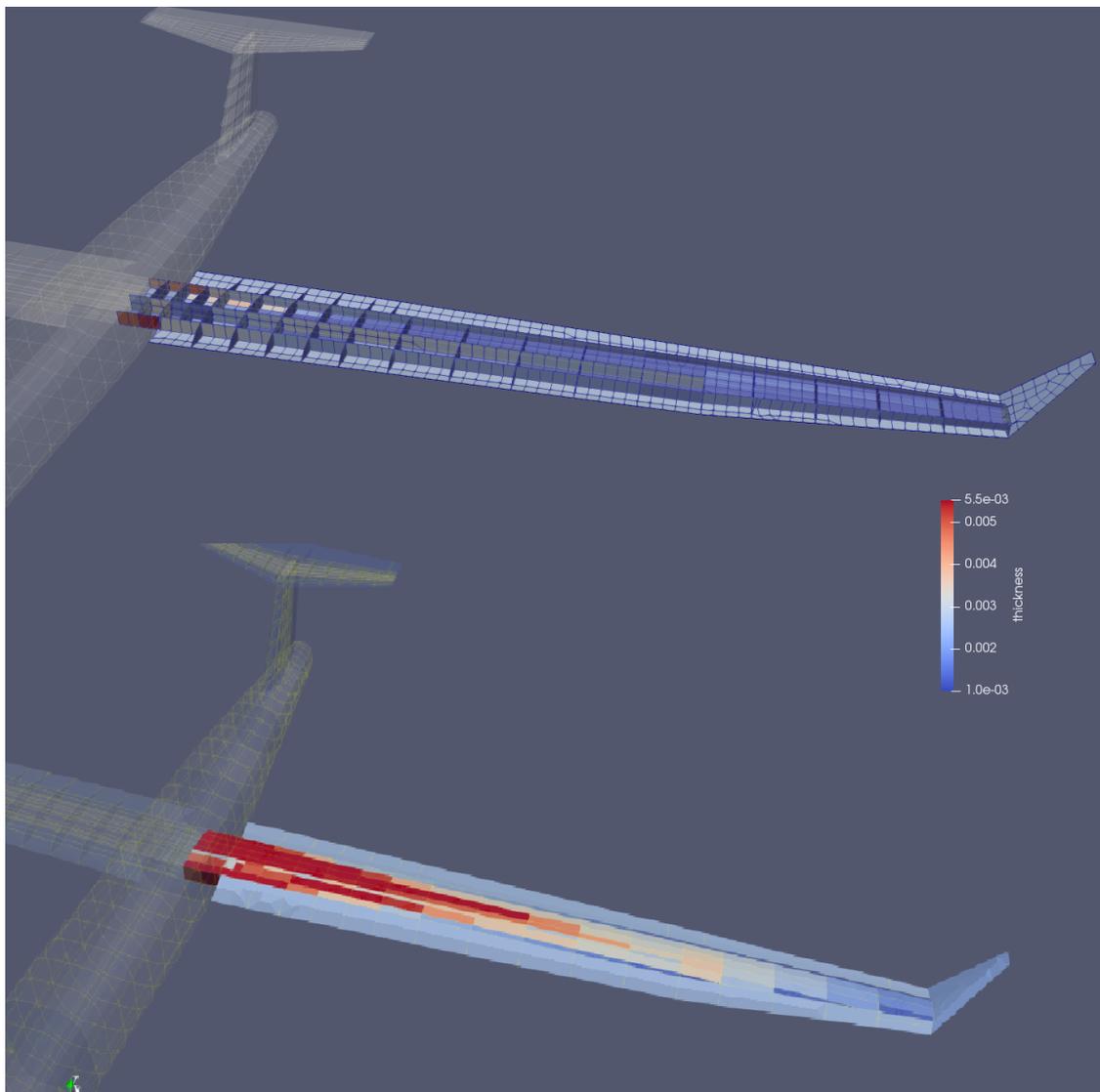


Figure 5.4: Optimal CPACS structural model and thickness distribution representation. The first figure represent the wing without upper skin which instead is shown in the second one.

structure is composed by a set of one dimensional stringers and frames on the skin. Six bulkhead elements are insert to increase the stiffness of the fuselage. Also mass elements are considered during calculation. They are obtained as result of mass model calculation. The structural model take into consideration also the masses of on-board systems, payload and fuel. In addition, also the elements structural masses are considered. Fig. 5.4 shows the structural model of the wing with and without upper skin, in order to display the internal structure. It is also possible to observe the thickness distributions of each component. This structural mesh will be used for all the elastic analysis performed to obtain shape deformation under aerodynamic load. As can be seen, the structural grid of the lower skin is different from a typical aerodynamic mesh (For instance, compared to the one in Fig. 4.6. That is the reason why to perform aero-elastic computation is necessary a load mapping, in order to obtain forces in structural nodes starting from pressure in aerodynamic nodes and a displacement mapping, in order to obtain the deformation of aerodynamic mesh starting from the structural one.

5.2 Aero-Structural Equilibrium in Cruise

5.2.1 Mesh and Input Data for Flexible Cruise

Through the use of the workflow presented in section 4.4, it is possible to execute an analysis with a single flow condition. The aerodynamic condition searched for this computation is the cruise configuration. The aim of this test is to obtain the shape deformation and load in cruise. An analysis at fixed value of CL is performed. For each aerodynamic iteration the solver looks for the value of the angle of attack needed to reach the specified lift coefficient. Once found, it performs a CFD computation at that angle of attack. After that, as already explained in section 4.4, the force distribution among the grid cells is computed starting from the nodes pressure distribution. In this way, the structural solver can compute the load on structural grid and subsequently, through a static linear analysis, the displacement of each node. In conclusion, the aerodynamic mesh is deformed in order to represent the structural shape obtained from previous calculation and the

loop restarts.

Since the flow field condition in analysis is only one, the number of aerodynamic iterations will not be too high. By consequence an aerodynamic computation of the whole aircraft configuration is performed. The aerodynamic mesh generated for this calculation is shown in Fig. 5.5. The structural mesh is the one presented in section 5.1.3.

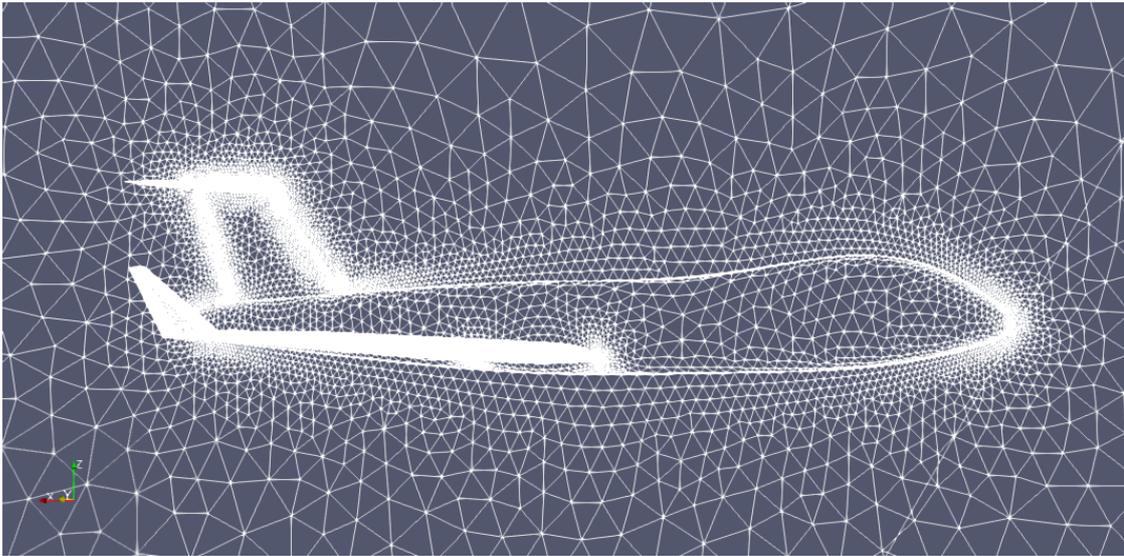


Figure 5.5: Optimal complete aircraft surface mesh surrounded by volumetric mesh. This grid is used as input for the computation of aero-structural deformation in cruise condition.

In a coherent way with the requirements data indicated in Tab. 5.1 the data assumed for the aerodynamic calculation input are illustrated in Tab. 5.5. The estimation of the weight in cruise is assumed by DLR data, adding 9200 kg of aircraft structure and fuel weight to the payload weight. The cruise speed and altitude is the one indicated by TLARS. The number of Mach related to these flight condition corresponds to 0.508. Moreover, the C_L required at cruise condition can be computed from Eq. 5.1. Where W is the mass of the aircraft taken into consideration, g is the gravitational acceleration, ρ is the air density, V the speed of the aircraft and S the wing surface.

$$C_L = \frac{W \cdot g}{\frac{1}{2}\rho V^2 S} \quad (5.1)$$

The cruise lift coefficient required at flight condition indicated in Tab. 5.5 corresponds to approximately 0.85.

Parameter	Value
Weight	10000 kg (800 kg are for payload)
Altitude	15 km
Temperature	216.65 <i>K</i>
Pressure	11597.3 <i>Pa</i>
Air density	0.186481 <i>kg/m³</i>
Speed	150 <i>m/s</i>
Side-slip angle	0°
Wing surface	55.2 <i>m²</i>
Number of Mach	0.508
Lift coefficient	0.85

Table 5.5: Parameters used for flexible cruise condition calculation through aero-structural workflow. Data shown are not all independent from each other.

5.2.2 Flexible Cruise Results

Five iterations of the aero-structural equilibrium loop are necessary to reach a convergence. The delta value set to indicate that convergence is reached is 1 millimeter of maximum displacement of the configuration shape. Fig. 5.6 shows the value assumed by this delta value, the angle of attack and the drag coefficient during the iterations. As can be noticed, after the first iteration the shape deformation obtained is already similar to the final shape. Indeed, the maximum displacement obtained between first and second deformation corresponds to 400mm. The other value are all under 10mm. The same happens for the angle of attack. For each iteration a new wing shape is analyzed and so a new value of angle of attack is reached in order to obtain the desired value of lift coefficient. As can be noticed, the value of this angle increases during the five iterations. This happens because the wing experiences twist deformation in addition to bending deflection. Sections of the wing experiences a nose down twist deformation that generates a decrease on

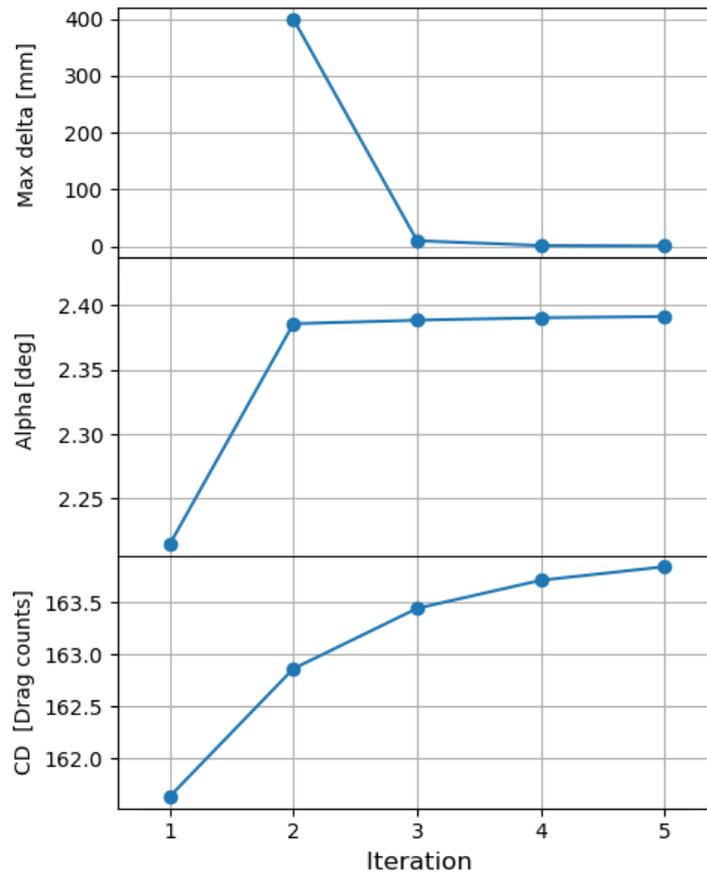


Figure 5.6: Trend of maximum value of deformation, angle of attack and drag coefficient computed through loop iterations performed to obtain Optimale elastic cruise condition. It is possible to observe that the greatest iteration is the first, the reaming are required only to reach the desired value of convergence.

the lift generated. It means that in order to reach the same value of lift coefficient obtained in rigid condition an high value of angle of attack is required for flexible condition. In Fig. 5.7 a comparison between the rigid and flexible twist distribution obtained in cruise condition for Optimale configuration is shown. It is possible to observe the above-mentioned general reduction of twist angle. The average difference between rigid and flexible twist angle corresponds to 0.24 degrees. It is also possible to observe that the difference increase moving versus the wing tip. This is due to the minor rigidity of the external wing sections. In conclusion, the reduction of twist section angle indicates that the elastic axis of the wing is always forward with respect to the aerodynamic center line. In addition, the increase of angle of attack generates an increase of the drag coefficient as can be seen from

Fig. 5.6. In conclusion, a comparison between the rigid and flexible complete aircraft shape is shown in Fig. 5.8. It is possible to observe that only the main wing experiences an appreciable value of deformation.

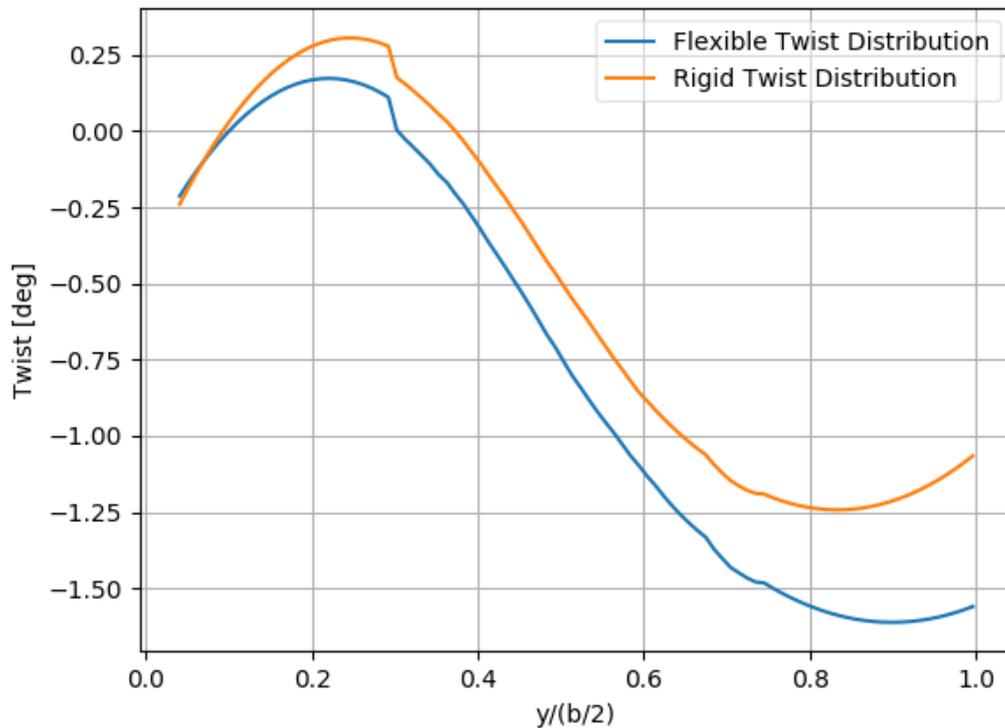


Figure 5.7: Comparison between rigid and flexible twist distribution of Optimal wing in cruise condition. All the section experience a reduction of twist angle that is higher for section near to the wing tip. The average difference between the two distributions corresponds to 0.24 degrees.

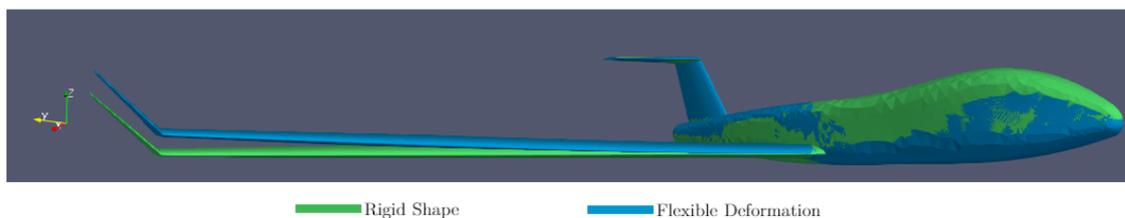


Figure 5.8: Comparison between rigid and flexible Optimal configuration shape. The maximum value of deformation is reached at wing tip, corresponding to a deflection of 0.4 meters. The remaining part of the aircraft does not experience an appreciable value of deformation.

5.3 Flexible Drag Polar

5.3.1 Mesh and Input Data for Flexible Wing Polar

The following test regards an analysis that provides a complete application of the aero-structural workflow presented in section 4.4. Indeed, in this case different flow conditions are analyzed. In particular, the angle of attack of the simulation is changed after achieving the aero-structural equilibrium and so, the convergence. In this way it is possible to compute the polar of the wing analyzed in flexible condition. In this case, only the wing mesh is taken into consideration. Computation of flexible shape at different flow conditions means that an high number of CFD analysis have to be performed. Just considering the test analyzed in previous section, five different iterations are needed to obtain the desired value of convergence. Multiplying this number by the amount of angles of attack required by input, the computational time will escalates greatly. This is the first reason that justify the analysis of isolated wing. In addition, from the previous test results, it was possible to observe that only the main wing experiences an appreciable value of deformation. By consequence the only reason to study aero-elasticity of the complete configuration could only be the possibility to see the effect on aircraft stability.

In Fig. 5.9 the mesh obtained and employed to obtain the following results is illustrated. It is obtained through a "Global Scale" parameter equal to 0.7. By consequence it satisfies the conditions selected after mesh sensitivity analysis performed in section 5.1.2.

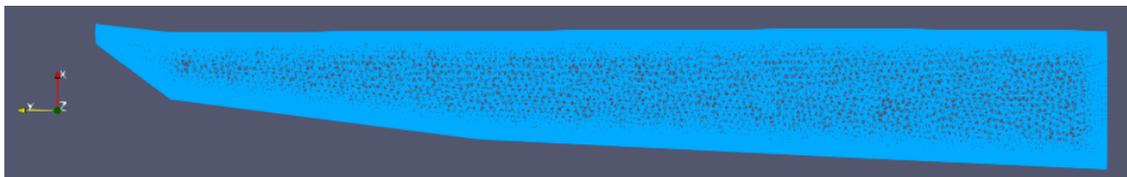


Figure 5.9: Top view of the Optimale wing mesh employed to perform aerodynamic calculations for aero-structural equilibrium workflow. The mesh is chosen in order to both satisfy accuracy and computational time desired.

Tab. 5.6 shows the input data used to compute the flexible polar though the aero-structural equilibrium workflow. They corresponds to the same aerodynamic

Parameter	Value
Weight	10000 kg (800 kg are for payload)
Angle of attack	From -8° to 8° with a step of 2°
Altitude	15 km
Temperature	216.65 K
Pressure	11597.3 Pa
Air density	0.186481 kg/m ³
Speed	150 m/s
Side-slip angle	0°
Wing surface	55.2 m ²
Number of Mach	0.508

Table 5.6: Parameters used for flexible cruise condition calculation through aero-structural workflow. Data shown are not all independent from each other.

condition used for flexible cruise calculation. Indeed, the aim of this test is to obtain a flexible polar in condition not too far from cruise condition. The angle of attack chosen corresponds to a set of angles which starts from -8 and comes to 8 degrees. One positive aspect of performing the aero-elastic analysis through an iteration method is that the first iteration does not takes into consideration the structure of the wing. By consequence, it is possible to observe the first and the last iteration in order to make a comparison between the rigid and the flexible results. That means that input data shown in Tab. 5.6 can be considered as input for both flexible and rigid calculations.

5.3.2 Flexible and Rigid Wing Polar Results

Fig. 5.10 represents the comparison between shape deformation obtained after the reach of convergence for each case analyzed through the aero-structural equilibrium workflow on Optimale wing. The flow field condition are indicated in Tab. 5.6. It is possible to notice how the wing experiences both downwards and upwards bending deflection. The case with the lower value of deformation is the one with zero angle of attack, in that condition the resulting shape is very

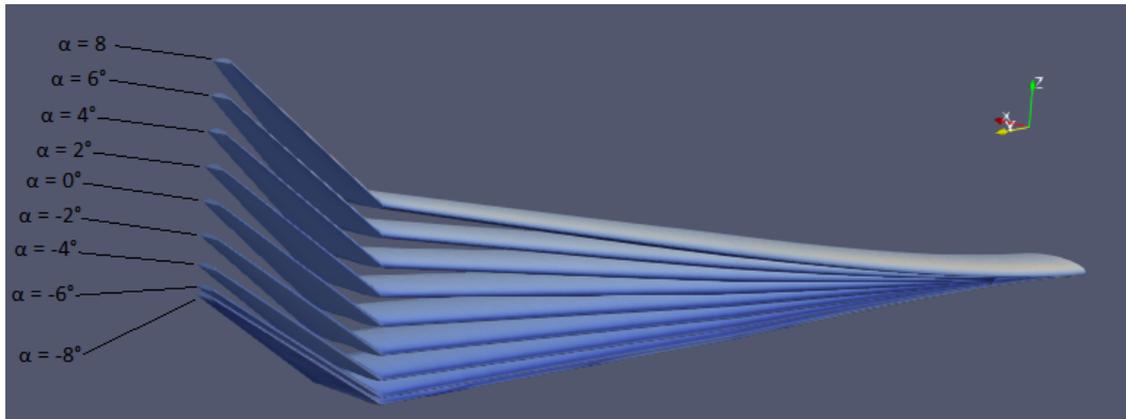


Figure 5.10: Comparison between Optimale wing shape obtained after aero-structural equilibrium computation at flow condition indicated in 5.6. It possible to observe deformations in both possible direction of bending deflections.

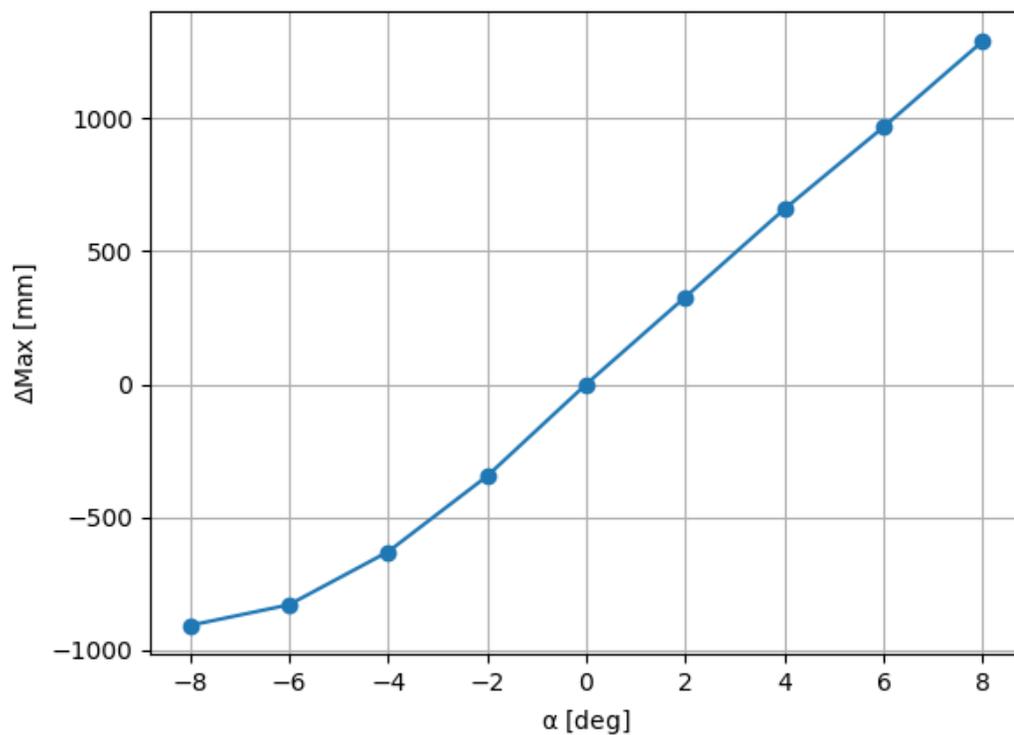


Figure 5.11: Trend of the maximum displacement value between the rigid and flexible Optimale wing shape at flow condition indicated in 5.6. It is possible to observe how the trend is similar to the lift trend of Fig. 5.12. Indication of the linearity of solution.

close to the rigid configuration. By consequence negative value of angle of attack generates downwards deflection. It happens despite the value of lift coefficient is positive, as can be seen from Fig. 5.12. The reason is the presence of wing weight

considered through LAGRANGE. The value of maximum deflection assumed by each deformation are indicated in Fig. 5.11. As can be noticed, at high incidence the value of maximum deflection overcame one meter. It is also possible to observe the linearity of the analysis. Indeed, the trend assumed by wing maximum deformation with respect to the angle of attack is the same assumed by lift coefficient as can be seen from in Fig. 5.12. This figure shows the comparison between rigid and flexible lift coefficient trend with reference to the angle of attack and the drag coefficient. It is possible to notice that the value of lift and drag forces does not experience a great modification in case of rigid or flexible analysis. A different effect can be observed

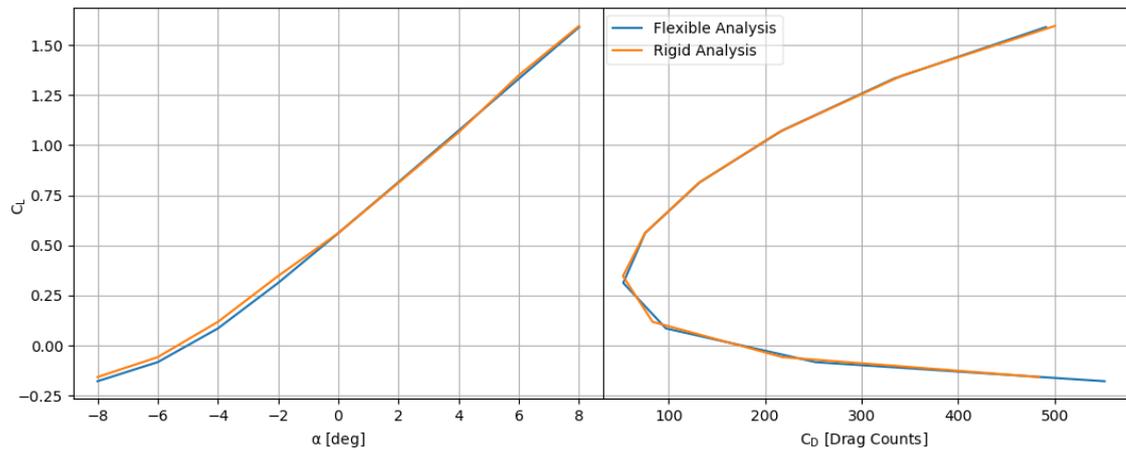


Figure 5.12: Trend of lift coefficient with reference to angle of attack and drag coefficient. The trend is compared between rigid and flexible calculation. It is possible to see that the lift coefficient is very similar between the two cases.

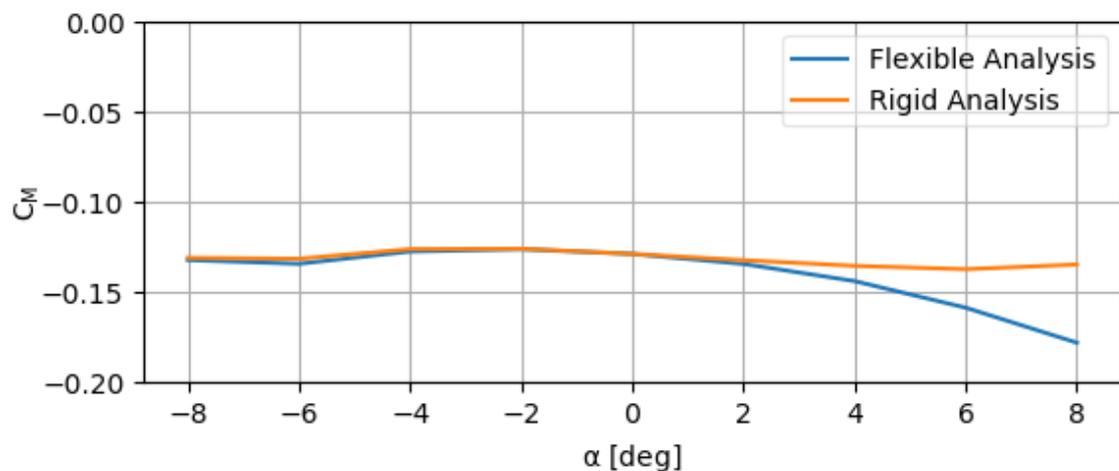


Figure 5.13: Comparison between pitching moment trend coefficient obtained through rigid and flexible calculation.

in Fig. 5.13, in which the trend of pitching moment coefficient with reference to the angle of attack is illustrated. The x origin coordinate used for pole calculation is located at 25% of the rigid wing's MAC. The pole is located in the symmetry plane of the aircraft and at the z coordinate position of the MAC. In coherence with the position of the pole, the value of the moment coefficient of rigid wing is quite constant with reference to the angle of attack. This does not happens in case of flexible wing. At positive angle of attack, as can be possible to notice in Fig. 5.10, the wing experiences an upward deflection and a nose down twist deformation. These phenomenon moves the center of pressure and also modify the inclination of the lift and drag forces. The origin for moment calculation instead, remain the same. By consequence a different value of the moment coefficient compared to the rigid case is obtained. To demonstrate what is previously described, Fig. 5.14 and

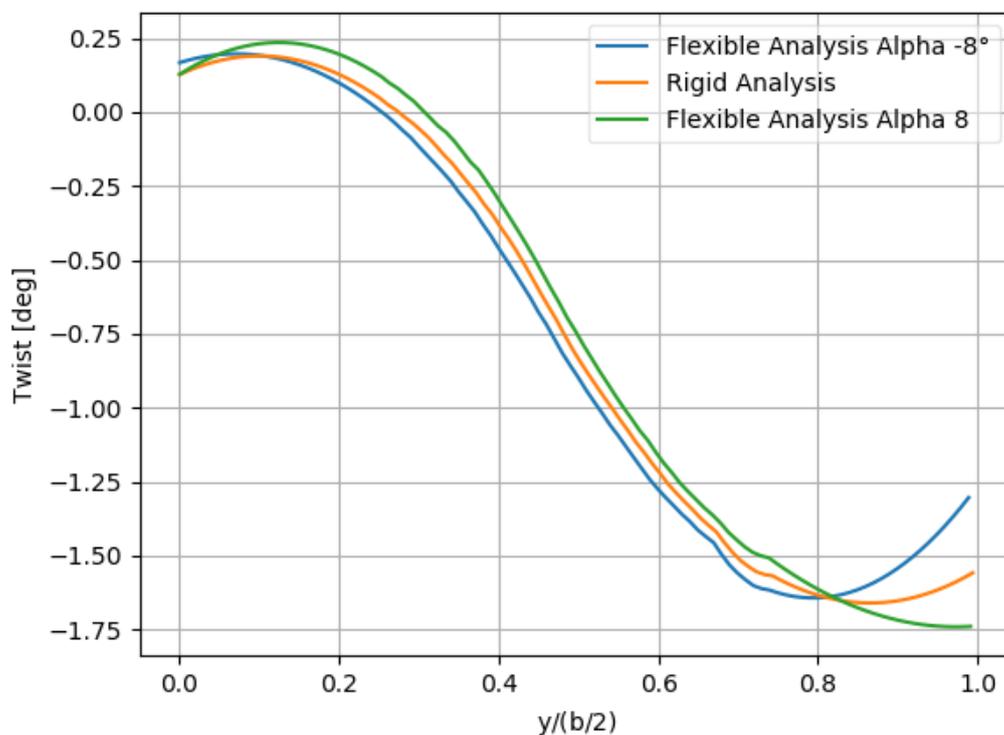


Figure 5.14: Comparison between Optimal wing twist distribution obtained through rigid and flexible analysis at different angle of attack. Only two angle of attack are considered for flexible case. The higher and the lower.

5.15 shows certain section parameters. The first figure illustrates the comparison between rigid and flexible wing twist distribution for two different incidences. As already said, a positive angle of attack generates a nose-down twist deformation. By contrast, a negative incidence brings the airfoil to twist in opposite direction. This is due to the different position of the aerodynamic center with respect to the shear center. Fig. 5.15 illustrates the airfoil shape and the relative pressure coefficient distribution of a specific wing section. Here, the angle of attack taken into consideration is 4° . A station near to the wing tip is chosen cause is representative of a high deformation case. It is possible to notice that the airfoil shape is almost the same. By consequence also the pressure distribution is the same. This indicates that the deformation are principally three-dimensional. The wing shape change as is shown in Fig. 5.10, but the airfoils shape remains basically the same.

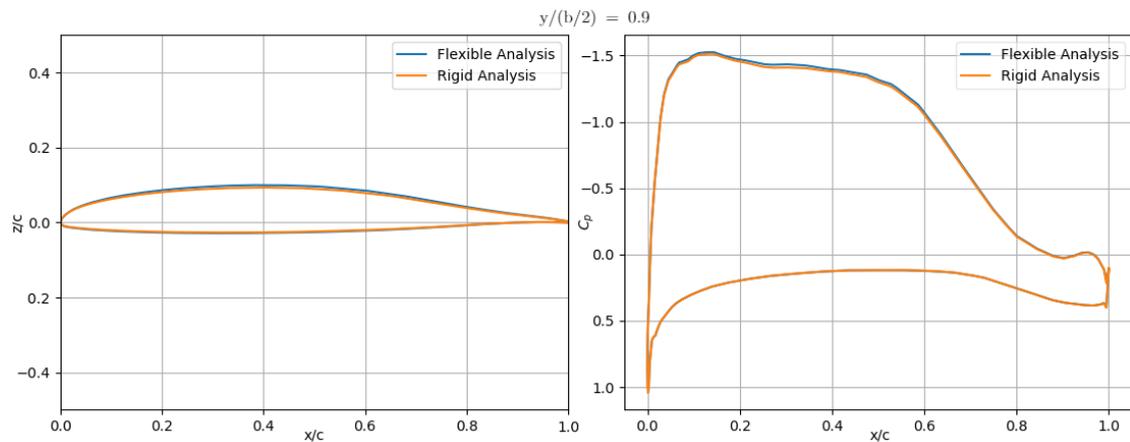


Figure 5.15: Comparison between rigid and flexible Optimal wing section. The incidence of this results is 4° . The y coordinates of the wing station is 0.9. The airfoil shape and the pressure coefficient distribution are shown.

5.3.3 Flexible and Rigid Optimal Polar

An analysis at different angles of attack of the Optimal complete configuration is performed. As already shown in Fig. 5.8, only the wing experiences an appreciable value of deformation. By consequence the wing deformation and aerodynamic results coming from this analysis are similar to the those shown in section 5.3.2. What is interesting to show is the effect of wing flexibility on the overall configuration.

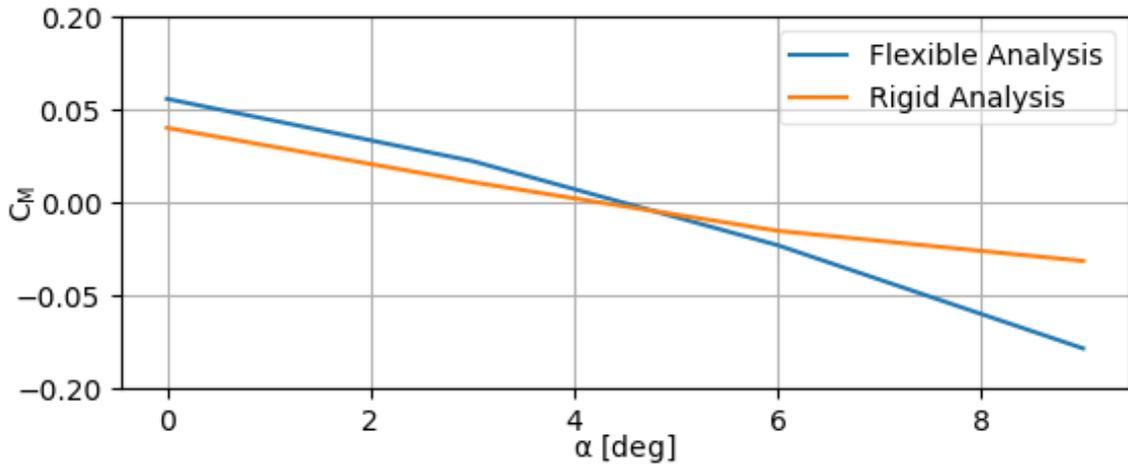


Figure 5.16: Comparison between the flexible and rigid pitching moment coefficient trend of the entire Optimale configuration with respect to alpha.

In Fig. 5.16 the pitching moment coefficient with respect to the angle of attack is illustrated. The origin point used as pole for moment computation possesses x coordinate located at 25% of the MAC. The y coordinate is in the symmetry plane of the aircraft while the z coordinate corresponds with the MAC one. As can be noticed, the moment coefficient has a negative slope if the incidence increases. It means the total aircraft is statically stable. It happens both at rigid and flexible case. The main difference is that the flexible moment coefficient possesses a lower value of the derivative, as already seen for isolated wing calculation. Fig. 5.17 shows the pitching moment contribution of wing, fuselage and horizontal tail, all calculated with reference to the wing area. The summation of all these contributes generate the total aircraft pitching moment coefficient. Two main phenomena can be noticed. The wing contributes with a more negative moment coefficient at high incidence. It happens for the same reason explained in section 5.3.2.

The wing experiences an upward deflection and a nose-down twist deformation. These two phenomena generate a displacement of the aerodynamic center and a rotation of the drag and lift forces generated by the wing. As a consequence, the moment coefficient, that is calculated with reference to the same pole used for rigid computation, decreases in value. Another relevant aspect is the change of the moment contribution which comes from the horizontal tail. This phenomenon can

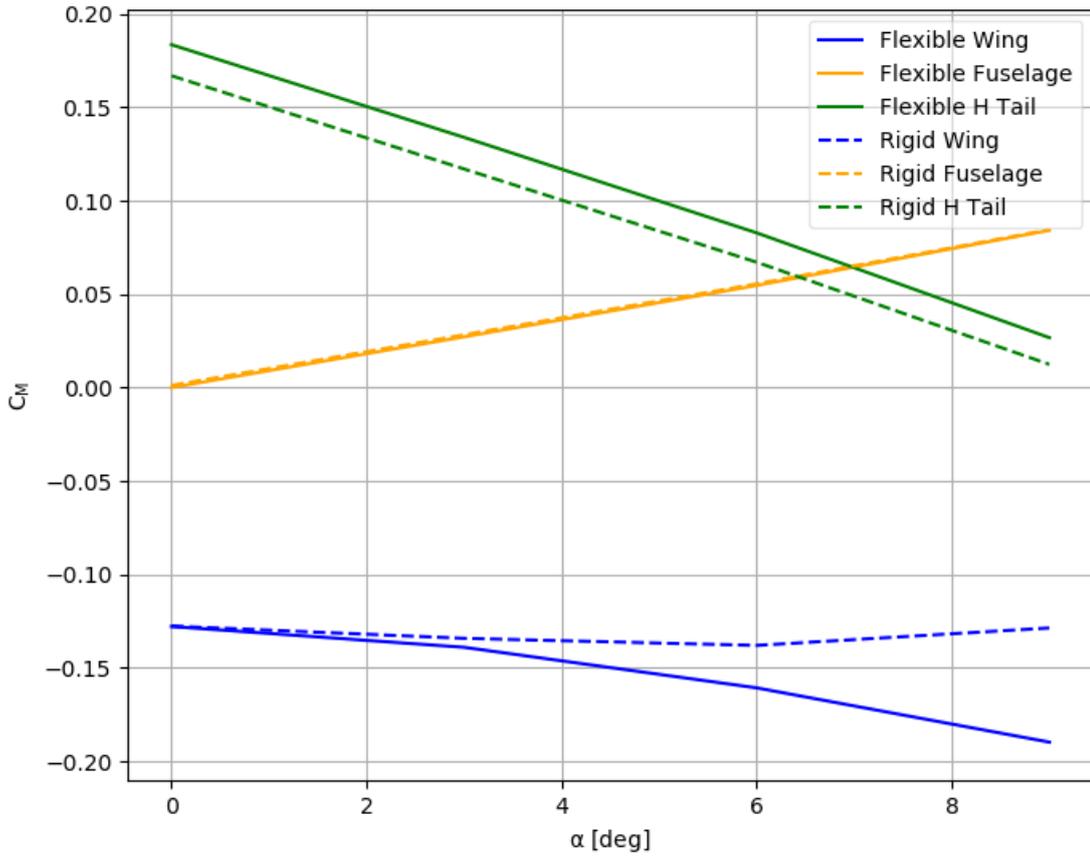


Figure 5.17: Pitching moment coefficient breakdown of the Optimale configuration components. Wing, fuselage and horizontal plane are taken into consideration for both rigid and flexible calculation.

be explained through Fig. 5.18. The lift coefficient of the horizontal tail-plane is represented, it is calculated with respect to its surface instead of the wing area. It is possible to notice that there is a translation of the lift curve in downwards direction. This happens because of a downwash effect due to a different wing shape. Since the aircraft includes a low wing and a T-tail configuration, the wing is located under the horizontal plane. It means that in case of upward deflection of the wing, the plane of the wake is moved higher, closer to the horizontal plane. As a consequence, the downwash effect on the tail-plane is slightly higher and so the angle of attack of the flow impacting the plane poses a reduced value. That is the reason why the lift coefficient curve is translated in downward direction. In conclusion, a lift drop generates an increase of pitching moment coefficient, as illustrated in Fig. 5.17. The moment generated by the fuselage is not affected by flexible calculation and

is, as usual, destabilizing. The vertical tail is not considered since its contribution to the pitching moment coefficient is almost negligible.

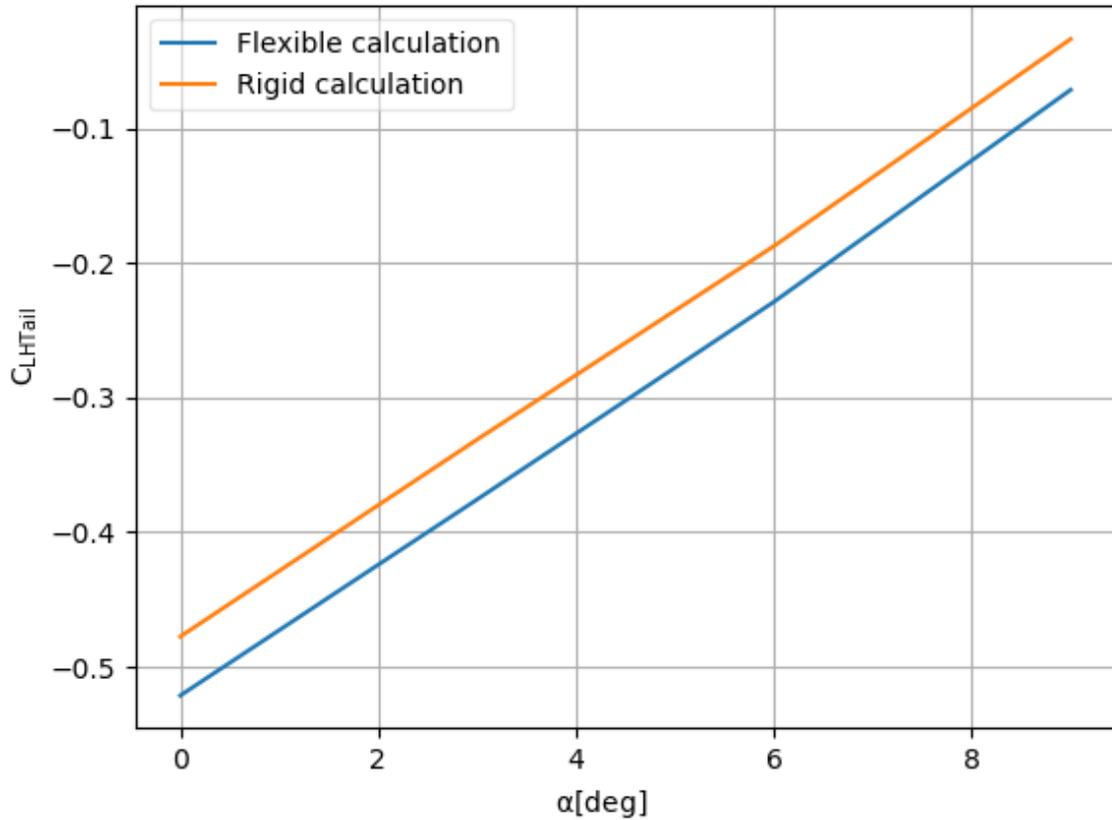


Figure 5.18: Optimal horizontal tail lift coefficient curve computed for rigid and flexible calculation. The lift coefficient is scaled with reference to the horizontal tail surface.

5.4 CPACS Output Generation

At the end of the analysis, in addition to the possibility of performing different post-processing studies, as already shown in the previous sections, it is important to generate an output suitable for the collaborative context in which the work carried out is inserted. As deeply explained in section 2.6, especially for MDO architecture, a centralized data format is required. All the exchange of data among different tools or entire workflows must be provided using files that respect the common format which is, in this specific case, a CPACS file. The main output generated after aero-structural equilibrium computation is the final shape of the aircraft. It is also important to be able to indicate the flight and structural conditions in which

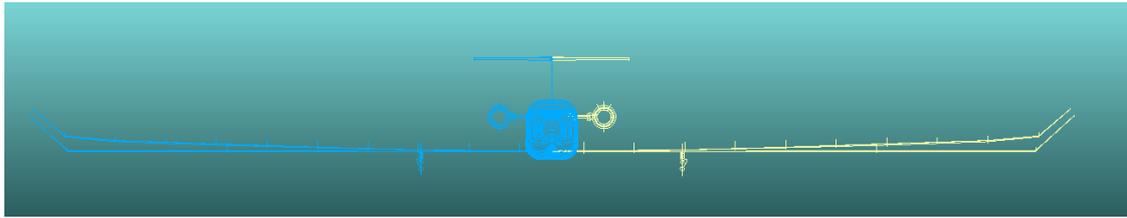


Figure 5.19: Comparison between input and output of "CPACS Geometry Update" tool. Wing shape deformation results obtained in section 5.2 are used as input.

the supplied shape geometry is obtained. An advantage of using CPACS file as input for the entire workflow consists in the possibility to be in posses of a file with all the tools input information. It means that after writing the flexible geometry, in the file the aerodynamic and structural conditions are already present. In specific tags all the aerodynamic conditions such as the angle of attack or the lift coefficient of calculation, the altitude and the parameter defined for mesh generation are provided. In addition, also the structural mesh features are already in the file, since the structural tool reads its inputs from there. Other information that can be added in this common file are the aerodynamic results of the new geometry and obviously the new aircraft shape. For these reason, the two CPACS interface tools described in section 4.5 can be executed. As already explained, the "Aerodynamic Results Writer" tool is able to write SU2 input and output in specific tag of the input CPACS file. In this way, all the information regarding forces, moment and flow field conditions of flexible configuration can be collected in a specific section. Moreover, the "CPACS Geometry Update" tool can be executed. A file containing more than one wing slice section must be provided. As already explained, this file can be obtained through the SU2 routine "SU2_GEO". After that, the CPACS wing geometry is updated. An illustration which compares the input and output CPACS of the tool is provided by Fig. 5.19. The figure shows the section and the connections between sections. The rigid geometry posses five sections used to define the span-wise wing shape. The flexible one is obtained trough twelves sections. The first and the last one are the original root and tip sections, this happens because the other ten used as input to update the wing geometry are

located between the tip and the root of the wing. By consequence, the tool does not remove the sections placed before and after the input one.

Conclusions and Future Developments

The automated aerodynamic and structural processes have been developed and tested in order to demonstrate the feasibility of applying high fidelity methods in multidisciplinary studies performed during early aircraft design phases. In order to couple the possible analysis with a generic external tool or workflow, the processes is integrated in a distributed design environment (RCE). In this way and also thanks to CPACS, the common parametric language file format, it is possible to easily exchange the information between different disciplines or modules. This is what characterizes a collaborative design structure. All the input and output of the developed process are provided through CPACS format file. As a consequence all the information generated that can be needed to a generic external tool, which share the same collaborative structure, are stored in this file.

A detailed description of each tool which composes the developed processes and a presentation of the overall architecture that characterize all the integrated workflows is provided. Also specific tests are performed to demonstrate the capability of the tools. In addition to the different cases tested to describe each described methods, an application of the overall process has been performed on a test case aircraft configuration. A mesh sensitivity analysis is executed and aero-structural analysis in cruise and at different flow field conditions are performed on a UAV configuration aircraft. After that, as already explained, all the information are included in the output CPACS format file. In this way the capabilities of the workflows are presented.

As future development, an aero-structural optimization can be performed. It is possible to use the developed tools in order to obtain aerodynamic and structural gradients. The main difficulty probably consists in computing both gradients with respect the same geometrical quantities. A solution can be the use of adjoint method to compute gradients with respect to each mesh node displacements and then project this gradient to the desired design variable that can be the section thickness, curvature, leading-edge radius or the wing span, sweep and dihedral.

Bibliography

- [1] A. Mark. Cfd: An industry perspective.” Future Directions in CFD Research: A Modeling and Simulation Conference, Hampton, Virginia. 2012.
- [2] J. Alonso D. Darmofal W. Gropp E. Lurie D. Mavriplis J. Slotnick, A. Khodadoust. CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences. *NASA/CR-2014-218178*, March 2014.
- [3] X. Gu. Application of Computational Aerodynamic Analysis and Optimization in a Multi-Fidelity Distributed Overall Aircraft Design System. August 2017.
- [4] J. A. Samareh. Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimization. *AIAA journal*, 2001.
- [5] L. Stingo. 3rd generation of collaborative MDO for more efficient aircraft preliminary design. The turboprop case within the AGILE project. January 2020.
- [6] P. Faratin M. Klein. The Dynamics of Collaborative Design: Insights From Complex Systems and Negotiation Research. *Acoustics, Speech and Signal Processing Newsletter*, October 2002.
- [7] A. B. Lambe J. R. R. A. Martins. Multidisciplinary Design Optimization: A Survey of Architectures. *AIAA Journal*, 2013.
- [8] RCE: a distributed, workflow-driven integration environment. <http://rcenvironment.de/>.
- [9] CPACS: a common language for aircraft design. <http://www.cpacs.de>.
- [10] B. Nagel L. Boggero M. Fioriti P. S. Prakasha, P. D. Ciampa. Collaborative Systems Driven Aircraft Configuration Design Optimization. *30th ICAS Conference*, September 2016.

- [11] M. Litz A. Schreiber A. Bachmann, M. Kunde. Advances in generalization and decoupling of software parts in a scientific simulation workflow system. *The Fourth International Conference on Advanced Engineering Computing and Applications in Sciences, Florence, Italy, 2010.*
- [12] M. J. L. Van Tooren G. La Rocca. Knowledge-Based Engineering Techniques to Support Aircraft Design and Optimization. *Journal of Aircraft*, November 2009.
- [13] J. R. R. A. Martins. A Coupled-Adjoint Method for High-Fidelity Aero-Structural Optimization. October 2002.
- [14] L. Fu Z. Gao G. Chen Y. Zuo, P. Chen. Advanced Aerostructural Optimization Techniques for Aircraft Design. *Journal of Aircraft*, September 2015.
- [15] R. T. Haftka. Optimization of Flexible Wing Structures Subject to Strength and Induced Drag Constraints. *AIAA Journal*, 1977.
- [16] W. H. Eppard Z. Gürdal R. T. Haftka B. Grossman, G. Strauch. “Integrated Aerodynamic/Structural Design of a Sailplane Wing. *AIAA Journal*, 1988.
- [17] P.-J. Kao D. M. Polen M. Rais-Rohani B. Grossman, R. T. Haftka. Integrated Aerodynamic-Structural Design of a Transport Wing. *Journal of Aircraft*, 1990.
- [18] C. Farhat K. Maute, M. Nikbay. Coupled analytical sensitivity analysis and optimization of three-dimensional nonlinear aeroelastic systems. *AIAA Journal*, 2001.
- [19] J. J. Reuther J. R. R. A. Martins, J. J. Alonso. Aero-Structural Wing Design Optimization Using High-Fidelity Sensitivity Analysis. *Proceedings of the CEAS Conference on Multidisciplinary Aircraft Design and Optimization*, June 2001.
- [20] J. J. Reuther J. R. R. A. Martins, J. J. Alonso. High-Fidelity Aerostructural Design Optimization of a Supersonic Business Jet. *Journal of Aircraft*, 2004.

- [21] J. J. Reuther J. R. R. A. Martins, J. J. Alonso. Sensitivity analysis and design optimization of three-dimensional non-linear aeroelastic systems by the adjoint method. 2003.
- [22] K. Maute M. Barcelos, H. Bavestrello. A Schur-Newton-Krylov solver for steady-state aeroelastic analysis and design sensitivity analysis. 2006.
- [23] K. Maute M. Barcelos. Aeroelastic design optimization for laminar and turbulent flows. *Journal of Aircraft*, 2008.
- [24] J. R. R. A. Martins G. K. W. Kenway, G. J. Kennedy. Scalable parallel approach for high-fidelity steady-state aeroelastic analysis and derivative computations. *AIAA Journal*, 2014.
- [25] J. J. Alonso E. van der Weide C. A. Mader, J. R. R. A. Martins. ADjoint: An approach for the rapid development of discrete adjoint solvers. *AIAA Journal*, 2008.
- [26] G. Maffettone. Automatic Aerodynamic Analysis and Optimization for Aircraft Design. 2019.
- [27] Pointwise Glyph. <https://www.pointwise.com/glyph2/files/Glyph/cxx/GgGlyph-cxx.html>.
- [28] A. C. Aranake A. Campos S. R. Copeland T. D. Economon A. K. Lonkar T. W. Lukaczyk T. W. R. Taylor J. J. Alonso F. Palacios, M. R. Colonno. Stanford University Unstructured (SU 2): An open-source integrated computational environment for multi-physics simulation and design. *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, January 2013.
- [29] SU2, Multiphysics Simulation and Design Software. <https://su2code.github.io/docs/home/>.

- [30] ParaView, an Open-Source, Multi-Platform Data Analysis and Visualization Application. <https://www.paraview.org/>.
- [31] Tecplot: Data Visualization & CFD Post-Processing Software. <https://www.tecplot.com/>.
- [32] P. Peterson et al. E. Jones, T. Oliphant. SciPy: Open source scientific tools for Python. 2001.
- [33] SU2, Constrained shape design of a transonic inviscid wing at a cte. CL. https://su2code.github.io/tutorials/Inviscid_3D_Constrained_ONERAM6/.
- [34] R. Maierl A. Jungo M. Zhang J.N. Walther, A.-A. Gastaldi. Integration Aspects of the Collaborative Aero-Structural Design of an Unmanned Aerial Vehicle.
- [35] Ö. Petersson M. Wagner G. Schuhmacher, F. Daoud. Multidisciplinary Airframe Design Optimisation. *28th International congress of the aeronautical sciences*, 2012.
- [36] Wikipedia Airbus Defence and Space. https://en.wikipedia.org/wiki/Airbus_Defence_and_Space#Cassidian.
- [37] F. Daoud K. Bletzinger R. Maierl, Ö Petersson. Automatic Generation of Aeroelastic Simulation Models Combined with a Knowledge Based Mass Prediction. *5th CEAS Air & Space Conference, Delft*, 2015.
- [38] H. Wendland A. Beckert. Multivariate interpolation for fluid-structure-interaction problems using radial basis functions. November 2000.
- [39] Ansys Guide for Mesh Quality Check. <https://www.afs.enea.it/project/neptunius/docs/fluent/html/ug/node167.htm>.
- [40] CPACS Documentation. <https://www.cpacs.de/pages/documentation.html>.

- [41] F. Daoud J.N. Walther P.D. Ciampa A. Jungo M. Zhang R. Maierl, A.A. Gastaldi. Aero-Structural Optimization of a Male Configuration in the Agile MDO Framework. *31st Congress of the International Council of the Aeronautical Sciences*, September 2018.
- [42] C. Kennedy Joaquim R.R.A. Martins K. A. James, J. Graeme. Concurrent aerostructural topology optimization of a wing box. 2013.
- [43] G. K. W. Kenway. A Scalable, Parallel Approach for Multi-Point, High-Fidelity Aerostructural Optimization of Aircraft Configurations. 2013.
- [44] J. R. R. A. Martins G. J. Kennedy. A parallel aerostructural optimization framework for aircraft design studies. August 2014.
- [45] D. Seider E. Baalbergen R. Lombardi R. D'Ippolito P. D. Ciampa, E. Moerland. A Collaborative Architecture supporting AGILE Design of Complex Aeronautics Products. *AIAA AVIATION Forum*, June 2017.

Acknowledgements

My heartfelt thanks go to the people who made my thesis work possible. Professors Pierluigi Della Vecchia, Fabrizio Nicolosi and Dr. Pier Davide Ciampa. They believed in me and allowed me to live such a formative and positive experience. My gratitude also goes to professor Della Vecchia and Francesco Torrigiani for their constant support. Thank you for all your suggestions, advice and for being my guide during all my thesis path. Many thanks also to Francesco for the funny experiences spent together.

A special thanks goes to Holst family. Thank you all for your delicious hospitality and for being friends to rely on for any necessity.

I would like to thank my family for enabling me to undertake this journey and to develop an enormous passion for my studies. Thank you for your moral and economical support. I would not be here celebrating without you.

Thanks to all my friends, colleagues and relatives for all the enjoyable leisure time spent together, for letting me have fun every time I needed it.

The greatest thanks, coming directly from my heart, goes to Martina. Thank you for being always on my side, thank you because you constantly supported me with all your love and your patience, during the best and the worst moments of my career. Thank you for always being the person to count on.