

POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Aerospaziale



**Politecnico
di Torino**

Tesi di Laurea Magistrale

**OPTICAL SENSORS NETWORK FOR STRUCTURAL
HEALTH MONITORING**



Relatore

Prof. Paolo Maggiore

Correlatori

Ing. Matteo Dalla Vedova

Ing. Gaetano Quattrocchi

Candidato

Liborio Puccio

Aprile 2022

Ringraziamenti

*Al mio Relatore, Prof. Paolo Maggiore,
Per i suoi preziosi insegnamenti durante tutto il corso di Laurea e per aver
suscitato in me un grande interesse verso il settore dei sistemi.
Ai miei Correlatori, l'Ing. Matteo Dalla Vedova e l'Ing. Gaetano Quattrocchi,
Per i loro utili consigli e per il supporto che mi hanno dato durante il mio lavoro
di tesi.
Ai ragazzi di ICARUS, in particolare ad Emanuele, Alessandro e Lorenzo,
Per la fantastica esperienza nel team e per la loro disponibilità.*

*A Mamma e Papà,
Per avermi cresciuto nel migliore dei modi, per la loro determinazione che è
stata d'esempio e per la fiducia e il supporto, anche nei momenti più difficili.
A Gabriele,
Per l'energia e la felicità che è sempre in grado di trasmettermi
Ai nonni e agli zii,
Per essere stati sempre presenti e su cui potrò sempre contare.
A Claudio,
Per aver riportato nella nostra famiglia una serenità che non immaginavo
potesse ritornare.
Agli amici di sempre, ad Angelo, Vincenzo e Rosario,
Per i bei momenti che continuiamo a passare insieme, nonostante le strade
diverse che abbiamo scelto.*

Sommario

L'obiettivo principale di questa tesi è sviluppare una metodologia per raccogliere e analizzare un gran numero di dati provenienti da simulazioni CFD e FEM, al fine di costruire una rete di sensori ottici il cui scopo è di monitorare i carichi agenti sulla struttura alare per espletare funzioni di diagnostica e monitoraggio real-time e per future applicazioni prognostiche.

Nel caso in esame, il metodo sarà applicato ad un prototipo di UAV alimentato da pannelli solari sviluppato dal Team ICARUS del Politecnico di Torino, che si presta bene a scopi sperimentali e potrebbe trarre grandi benefici da questa metodologia, non solo per il monitoraggio strutturale ma anche per il monitoraggio indiretto su altri sistemi.

Le Smart Structures offrono molti vantaggi nella manutenzione e nella sua programmazione, sia in termini di tempi di ispezione necessari che di costi associati alle procedure manutentive.

Sensori di tipo Fiber Bragg Grating (FBG) saranno utilizzati per questa applicazione, in collaborazione con Photonext Team.

Le simulazioni aerodinamiche (CFD) sono state realizzate utilizzando il software Siemens Star CCM+, che consente il calcolo dei carichi aerodinamici in diverse condizioni di volo.

I carichi ottenuti vengono quindi applicati al modello FEM strutturale dell'ala e l'analisi sarà condotta utilizzando Altair Hyperworks. La post-elaborazione e l'ottimizzazione dei dati verrà fatta tramite MATLAB.

Una rete neurale di tipo feedforward verrà in seguito addestrata utilizzando le deformazioni misurate dai sensori posti all'interno della struttura alare.

Al fine di ottenere una rete ben addestrata, verranno eseguite diverse simulazioni per costruire un set di dati significativo.

Abstract

The main goal of this thesis is to develop a methodology to collect and analyze large number of data coming from CFD and FEM simulations, in order to build an optical sensors network whose purpose is to monitor structural loads for future prognostic applications and for real-time monitoring.

In this case study, the method will be applied to a solar panel powered UAV prototype developed by the ICARUS Team, which is very suitable for experimental purposes and could greatly benefit from this methodology, not only for structural health monitoring but also for indirect monitoring on other systems.

Smart structures provide many advantages in maintenance, both in terms of inspection time needed and costs.

Fiber Bragg Grating (FBG) sensors will be used for this application, produced in collaboration with Photonext Team.

Aerodynamics CFD simulations are made using the software Siemens Star CCM+, allowing the computation of aerodynamic loads in different flight conditions.

The loads are then applied to the FEM structural model and the analysis will be conducted using Altair Hyperworks. Data post-processing and optimization is done with MATLAB.

A feedforward neural network is then trained using strains measured by the sensors placed inside the wing structure.

In order to achieve a well trained network, several simulations will be carried out to build a significant dataset.

Contents

1	Introduction	6
1.1	Thesis objectives and structure	7
2	Fiber Bragg Grating (FBG) sensors	8
2.1	Optical fiber	8
2.1.1	Optical fiber characteristics	8
2.1.2	Operating principles	10
2.1.3	Signal Attenuation	11
2.1.4	Main advantages	12
2.1.5	Connectivity	13
2.2	Fiber Bragg Grating sensors	13
2.2.1	FBG Operating principles	14
2.2.2	Fiber Bragg Grating (FBG) strain dependence	16
2.2.3	Fiber Bragg Grating (FBG) temperature dependence	17
2.2.4	FBG Classification	18
2.2.5	FBG Fabrication	20
2.2.6	Temperature effect compensation	22
2.2.7	Data acquisition system	25
3	ICARUS Team and project RA	27
3.1	ICARUS Team	27
3.2	Project RA	28
3.2.1	Aircraft Description	28
3.2.2	Wing aerodynamic and geometry	30
3.2.3	Wing structure	31
4	Flight conditions, FEM and CFD models	36
4.1	CFD model	36
4.1.1	Introduction to Computational Fluid Dynamics	36
4.1.2	Model geometry	38
4.1.3	Simulation parameters and boundary conditions	39
4.1.4	Mesh	40
4.1.5	Limitations and assumptions of the model	42

4.1.6	CFD analysis input and output	43
4.2	FEM model	45
4.2.1	Introduction to Finite Element Method	45
4.2.2	FEM analysis overview	46
4.2.3	Model geometry	48
4.2.4	Mesh	49
4.2.5	Materials	51
4.2.6	Loads and Boundary Conditions	53
4.2.7	Limitations and assumptions of the model	57
4.2.8	Results and data post-processing	58
4.3	Flight conditions	59
5	Sensors placement and configuration	61
5.1	Requirements and constraints	61
5.2	Final configuration	62
6	Artificial Neural Network	66
6.1	Introduction	66
6.2	Classification	69
6.3	Learning	70
6.4	Training and Validation	72
6.5	Neural network implementation	74
6.6	Results	78
7	Conclusions	81

List of Figures

2.1	Optical fiber structure	9
2.2	Optical fiber signal propagation	10
2.3	Optical fiber signal attenuation vs wavelength	11
2.4	Optical fiber bending attenuation	12
2.5	FBG sensor	15
2.6	FBG sensor	15
2.7	Reflected power vs FBG sensor length	16
2.8	Thermo-optic coefficient vs Temperature	17
2.9	Uniform FBG grating	18
2.10	Apodized FBG grating	18
2.11	Phase-shifted FBG grating	19
2.12	Chirped FBG grating	19
2.13	Tilted FBG grating	19
2.14	Interfering beam method	20
2.15	Phase mask technology	21
2.16	Point by point technique	22
2.17	Mechanical insulation by lead-through capillary pipe	23
2.18	Mechanical insulation by non-lead-through capillary pipe	23
2.19	Thermal expansion coefficient modification	24
2.20	Overwritten FBGs	24
2.21	Data acquisition system layout (SmartScan interrogator)	25
2.22	SmartScan SBI	26
3.1	Render of RA aircraft	28
3.2	CAD model of RA aircraft	29
3.3	CAD model (top view) of the wing	30
3.4	Airfoil - Selig 3002-099-83	30
3.5	Wing structure CAD	32
3.6	Wing structure detailed CAD	33
3.7	Wing joint detail	34
3.8	Flap mechanism (Sample from ANUBI (2017) model)	35
4.1	RA - Wing vorticity distribution ($\alpha = 10^\circ$)	37
4.2	RA - Longitudinal symmetry plane velocity distribution	38

4.3	Wing pressure distribution	40
4.4	CFD mesh around wing profile	41
4.5	CFD wing mesh	41
4.6	Prismatic layers detail	42
4.7	Pressure distribution ($V = 15m/s$)	43
4.8	Velocity distribution around wing profile ($V = 15m/s$)	44
4.9	CAD model imported for FEM analysis	48
4.10	FEM model	49
4.11	FEM bias setting towards the center hole (example)	50
4.12	FEM ribs mesh	51
4.13	FEM skin mesh	51
4.14	FEM model view	53
4.15	Pressure loads and concentrated loads	54
4.16	Concentrated loads detail	55
4.17	Battery pack distributed loads (in red)	56
4.18	Constraints	56
4.19	Sandwich ribs	57
4.20	Regions where FBG sensors installation is possible	58
4.21	Composite strains on skin ($V = 15\text{ m/s}$, $\alpha = 1^\circ$)	59
5.1	FBG sensors configuration - Top View	63
5.2	FBG sensors configuration - Bottom View	63
5.3	FBG sensors configuration - Isometric View	64
6.1	Artificial Neural Network diagram	67
6.2	Neuron model (Perceptron)	68
6.3	Training algorithms comparison	73
6.4	Feedforward Neural Network scheme	74
6.5	Common activation functions	76
6.6	Network Performance vs Number of neurons	78
6.7	Network Performance vs Number of neurons	79
6.8	Training Performance - One hidden layer	80
6.9	Training Performance - Two hidden layers	80

List of Tables

2.1	SmartScan SBI specifics	26
3.1	General characteristics (RA aircraft)	29
3.2	Airfoil characteristics	31
3.3	Aerodynamic coefficients ($V = 15m/s$)	31
4.1	TeXtreme Carbon fabric characteristics	52
4.2	SX10 EVO epoxy characteristics at 20 °C	52
4.3	Estimated carbon fiber laminate's characteristics	52
4.4	Battery packs weight	55
4.5	Cruise conditions	59
4.6	Flight conditions	60
5.1	FBG sensors details	65

Chapter 1

Introduction

The high complexity and costs of aircraft structures combined with their high operational reliability and safety needs results in increasing interest in structural health monitoring systems. Fibre Bragg Grating (FBG) sensors are an interesting tool for these applications.

The advantages of FBG sensors such as their small size, high multiplexing capabilities, corrosion resistance and good compatibility with the most advanced composite materials exploited in the aeronautics and aerospace fields made them a good tool for measuring strain within structures using a non-destructive approach.

The development of integrated sensor networks for non-destructive testing (NDT) is an essential part of the design and manufacture of composite aerospace structures in aerospace, military, and civil applications.

The use of composite components requires regular maintenance and the use of classical NDT techniques, such as ultrasonics, infrared thermography, laser velocimetry, etc. These techniques evaluate the condition of the parts and document the occurrence and in-service growth of defects in the lifetime of the parts, and this allows a prediction of the useful lifetime of the part. However, these inspections are time-consuming and require both qualified personnel and appropriate test equipment. In addition, there are difficulties in using traditional NDT techniques such as the inability to inspect hard-to-reach components, and the fact that in-service testing may be impossible.

Traditional strain-gauge sensors requires a dedicated electrical connection for each sensor making the implementation difficult and are also susceptible to electromagnetic interference, corrosion and losses.

Real-time monitoring offers many advantages, such as the possibility to change maintenance plan, replacing preventive maintenance for certain structures with on-condition maintenance, allowing reduction in costs and operational downtime. This will also increase safety as less maintenance operations are needed thus reducing the possibility of human errors.

Real-time monitoring also allows a fast identification of structural damage or wear and enable an optimized use of the structures by measuring real-time loads.

This thesis is partly based on a previous thesis job written by Ing. Emanuele Frediani, see bibliography for reference.

Fiber Bragg Grating sensors have been previously tested by the ICARUS Team on ANUBI, a competition model aircraft developed in 2017. This made it possible to acquire the necessary experience and carry out many tests that showed the feasibility and robustness of this type of system on a real context. The team thus decided to develop and optimize a methodology both numerical and experimental to implement a sensors network capable of doing real-time structural health monitoring and also to develop future prognostic applications for flight control actuators.

1.1 Thesis objectives and structure

The main goal of this thesis is to develop a methodology to collect and analyze large number of data coming from CFD and FEM simulations, in order to build an optical sensors network whose purpose is to monitor structural loads for future prognostic applications and for real-time monitoring.

The structure of the thesis is described below:

- **Chapter 2:** In this chapter optical fibers and Fiber Bragg Grating sensors will be introduced, from a physical point of view and also from a technological point of view
- **Chapter 3:** Brief description of the aircraft used for the simulations (Project RA)
- **Chapter 4:** Overview of the CFD and FEM models used for aerodynamic simulations and for finite elements structural analysis. In this chapter the analyzed flight conditions will also be discussed
- **Chapter 5:** In this chapter the configuration of the sensors and their positioning inside the wing box will be briefly analyzed
- **Chapter 6:** Introduction to Artificial Neural Networks (ANNs) and application to this case study
- **Chapter 7:** Conclusions and possible future developments

Chapter 2

Fiber Bragg Grating (FBG) sensors

2.1 Optical fiber

The advent of low-loss optical fibers in the early 1970s was a major event in the field of communication in the third quarter of the twentieth century. Since the first low-loss optical fiber with less than 20 dB/km at 632.8 nm in 1970, optical fiber loss has continued to evolve toward lower levels, their design and fabrication techniques have shown exceptionally rapid progress, reaching 0.142 dB/km at 1560 nm as of today.

Engineers now believe that in the distant future the optical fiber will change dramatically the entire aspect of communication.

In this chapter the theoretical basis of transmission characteristics of optical fibers will be described as well as the fabrication technologies to assist the comprehension of Fiber Bragg Grating (FBG) sensors.

2.1.1 Optical fiber characteristics

An optical fiber is a cylindrical dielectric waveguide made of low-loss materials such as silica glass. It has a central core in which the light is guided, embedded in an outer cladding of slightly lower refractive index. Light rays incident on the core-cladding boundary at angles greater than the critical angle undergo total internal reflection and are guided through the core without refraction. Rays of greater inclination to the fiber axis lose part of their power into the cladding at each reflection and are not guided. The fiber also has an external protective coating.

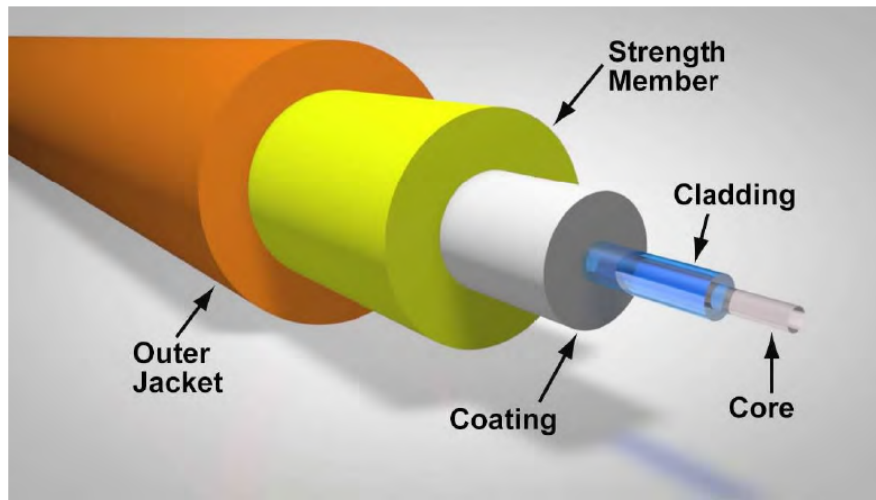


Figure 2.1: Optical fiber structure

- **Core:** This central section, made of silica or doped silica, is the light transmitting region of the fiber.
- **Cladding:** Around the core, this is the first layer. It is made of silica as well, but not in the same way as the core. By total internal reflection at the core-cladding contact, an optical waveguide is created, which confines light in the core.
- **Coating:** Around the cladding, the coating is the first non-optical layer. The coating is usually made up of one or more polymer layers that protect the silica structure from physical and environmental degradation. When the fiber is connectorized or fusion spliced, the coating is peeled off.

The strength member and outer jacket are situated outside of the coating and help protect the fiber from breaking during installation and termination.

Depending on the application, single mode and multimode fiber structures are employed in fiber optic communications. Light passes through multimode fiber in different "modes," which are different light routes. Only one path is propagated via the fiber in single mode fiber.

For this thesis single mode fibers will be used as there is no need to use multimode.

2.1.2 Operating principles

Optical fibers work on the principle of total internal reflection. Light reaching the boundary between two materials is reflected such that it never leaves the first material. In the case of fiber optics, light is reflected from the optical fiber core-cladding interface in such a way that it propagates down the core of the fiber. This can be explained by a brief discussion of Snell's law of refraction and law of reflection.

$$n_1 \sin(\theta_1) = n_2 \sin(\theta_2) \quad (2.1)$$

In our case n_1 is the refraction index of the core and n_2 is the refraction index of the cladding, with $n_1 > n_2$.

Light hitting the boundary or interface at angles greater than or equal to the critical angle would never pass into the second material, but would rather undergo total internal reflection, without losses.

$$\theta_c = \arcsin(n_2/n_1) \quad (2.2)$$

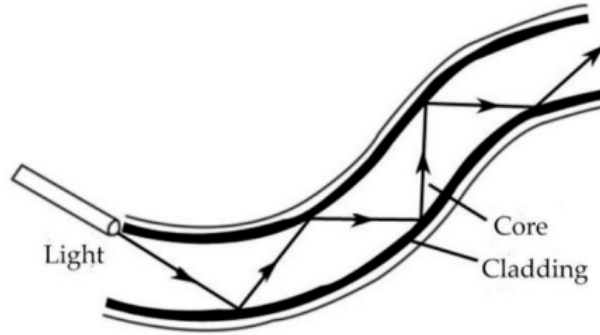


Figure 2.2: Optical fiber signal propagation

Another phenomenon to consider is Fresnel reflection, it occurs at the air-glass interfaces at the entrance and exit ends of an optical fiber. Resultant transmission losses, on the order of 4% per interface, can be reduced considerably by the use of index-matching materials. The coefficient of reflection depends upon the refractive index difference and the polarization of the incident radiation. For a normal ray, the fraction of reflected incident power is given by the following law:

$$R = (n_1 - n_2)^2 / (n_1 + n_2)^2 \quad (2.3)$$

2.1.3 Signal Attenuation

Light is gradually attenuated when it travels through fiber. The attenuation value is expressed in dB/km (decibel per kilometer). Attenuation is a function of the wavelength (λ) of the light.

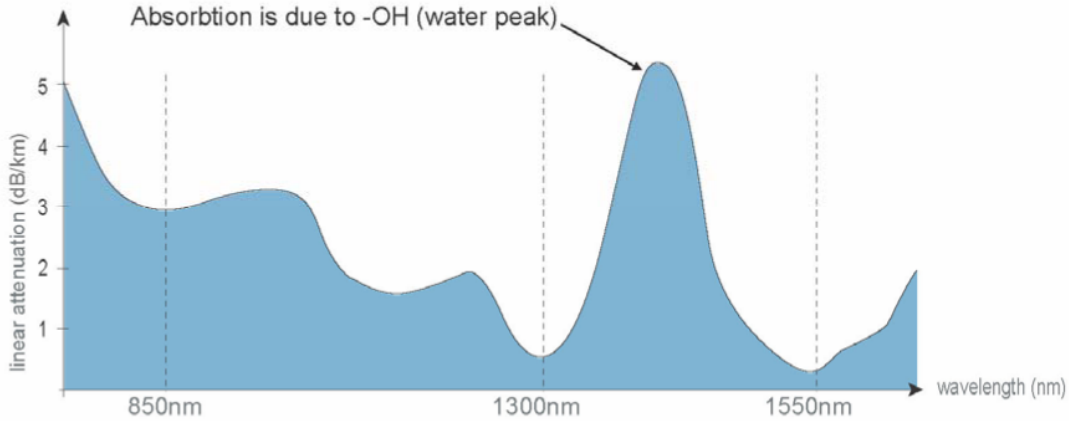


Figure 2.3: Optical fiber signal attenuation vs wavelength

Attenuation can be measured by the following coefficient:

$$\alpha = -\frac{10}{z[km]} \log \left(\frac{P(z)}{P(0)} \right) \quad (2.4)$$

Where $P(z)$ is the power measured in z position and $P(0)$ is the source power.

There are several causes of attenuation in an optical fiber:

- **Absorption:** it is caused by the absorption of the light and conversion to heat by molecules in the glass. Primary absorbers are residual OH- and dopants used to modify the refractive index of the glass. This absorption occurs at discrete wavelengths, determined by the elements absorbing the light. The OH- absorption is predominant, and occurs most strongly around 1000 nm, 1400 nm and above 1600 nm.
- **Scattering:** occurs when light collides with individual atoms in the glass and is anisotropic. Light that is scattered at angles outside the numerical aperture of the fiber will be absorbed into the cladding or transmitted back toward the source. Scattering is also a function of wavelength.
- **Macro-bending:** it is the attenuation associated with bending or wrapping the fiber. When the fiber is bent a percentage of the light is refracted out of the waveguide, depending on the bending radius. This is caused by the incidence angle being lower than the critical angle needed for total reflection.

- **Micro-bending:** it is an attenuation increase caused by high frequency longitudinal perturbations to the waveguide. The perturbations are normally considered as a set of very small radius bends of the fiber core. Physical irregularities on the surface in contact with the fiber perturb it into small radius bends which can cause microbending.

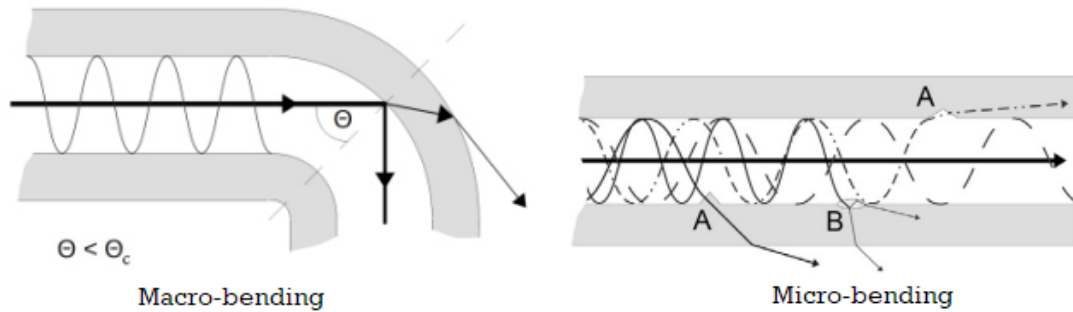


Figure 2.4: Optical fiber bending attenuation

2.1.4 Main advantages

Fiber optics continues to be used in more and more applications due to its inherent advantages over copper conductors.

- **Increased bandwidth:** The high signal bandwidth of optical fibers provides significantly greater information carrying capacity. Typical bandwidths for multimode (MM) fibers are between 200 and 600 MHz-km and more than 10 GHz-km for single mode (SM) fibers. Typical values for electrical conductors are 10 to 25 MHz-km.
- **Electromagnetic Interference:** Optical fibers are immune to electromagnetic interference and emit no radiation.
- **Decreased cost, size and weight:** Compared to copper conductors of equivalent signal carrying capacity, fiber optic cables are easier to install, require less duct space, weigh 10 to 15 times less and cost less than copper.
- **Lower loss:** Optical fiber has lower attenuation (loss of signal intensity) than copper conductors, allowing longer cable runs and fewer repeaters.
- **No sparks or shorts:** Fiber optics do not emit sparks or cause short circuits, which is important in explosive gas or flammable environments.

- **Security:** Since fiber optic systems do not emit RF signals, they are difficult to tap into without being detected.
- **Grounding:** Fiber optic cables do not have any metal conductors; consequently, they do not pose the shock hazards inherent in copper cables.
- **Electrical Isolation:** Fiber optics allow transmission between two points without regard to the electrical potential between them.

2.1.5 Connectivity

Fiber optic links require a method to connect the transmitter to the fiber optic cable and the fiber optic cable to the receiver. In general, there are two methods to link optical fibers together.

- **Fusion Splice:** This operation consists of directly linking two fibers by welding with an electric arc or a fusion splicer. The advantages of this approach are that the linking method is fast and simple and there is very little insertion loss. The disadvantages are that the link is relatively fragile, is permanent, and the initial cost (of the fusion splicer) is high.
- **Connectors:** The second method involves the uses of fiber optic connectors. A connector terminates the optical fiber inside a ceramic ferrule, using epoxy to hold the fiber in place. The connectors can be mated and unmated at any time. The advantages of this approach are that the connection is robust, the connector can be chosen according to the application.

2.2 Fiber Bragg Grating sensors

The introduction of fiber-optic communication has transformed practically every element of communication technology.

Optical fiber sensors have been extensively researched and used for a variety of applications in recent decades, including strain measurement, refractive index measurement, vibration of structures and machines, electric current, voltage, impedance measurement, temperature, pressure, humidity measurement, and so on. Although there has been substantial progress in this sector, the integration of optical mirrors, partial reflectors, and wavelength filters remains a difficulty since it adds to the complexity and cost of the system.

This challenge has been solved by using a fiber Bragg grating, which can perform all of the basic functions of reflection, dispersion, and filtering, making it ideal for sensing applications.

Photosensitivity is the process of optical absorption of ultraviolet (UV) light that affects the refractive index of the fiber core, resulting in a fiber Bragg grating (FBG).

Fiber Bragg grating was first discovered by Ken Hill in 1978 at Communication Research Centre, Canada. Ever since its discovery, grating structures have gained wide attention in the area of optical sensing due to their inherent advantages, such as low cost, small size, real-time response, high accuracy, high sensitivity, and immunity to electromagnetic interference.

There is great potential of sensing various parameters, such as temperature, pressure, stress, and refractive index using grating-based devices.

Such measurements are very important for example in the field of smart structures where sensors have a key role to monitor the conditions and supply the information to the control system that processes it and generate a useful effect, thanks to the actuators, to vary or restore the working conditions. Another advantage of using optical fibre sensors is that they don't need electrical power to work. Power source is required to the expensive equipment needed to process the measurement, but it can be installed far away from the sensor in remote places. Optical sensors are integral part of the fibre, so they are very small and light weighted, and it's possible to equip a single fibre with multiple sensors.

Current applications of fiber Bragg gratings are found in high temperature sensors, health and biomedical devices, structural engineering, industries, biochemical applications, radioactive environment, aerospace, maritime and civil engineering, and many other fields.

2.2.1 FBG Operating principles

Fiber Bragg grating sensors (FBG) consist of a series of reticular fringes, made by photoengraving in the core of the fiber using UV light. The segments of fiber that undergo this treatment behave intrinsically as sensors, maintaining the signal transmission function.

Photoengraving cause a periodic modulation of the index of refraction of the fiber core along the longitudinal direction, this allows the use of diffraction grating principle. By grating, it is meant that there is a periodic change in the core's refractive index. When the light travels inside the grating structure, some portion of light gets reflected back from each grating plane, so the FBG behaves like a filter. Only a specific wavelength is reflected to the source, denominated Bragg wavelength λ_B which corresponds to a peak in the reflected spectrum.

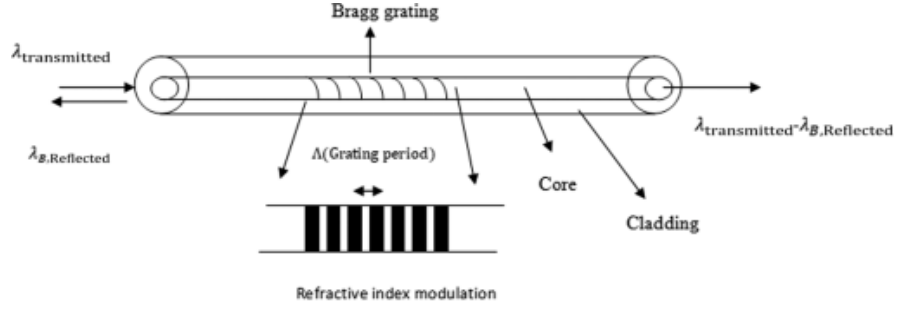


Figure 2.5: FBG sensor

The value of λ_B can be estimated with the following equation:

$$\lambda_B = 2\Lambda n_{eff} \quad (2.5)$$

where Λ is the grating period that forms the distance between two adjacent grating planes, n_{eff} is the effective core refractive index.

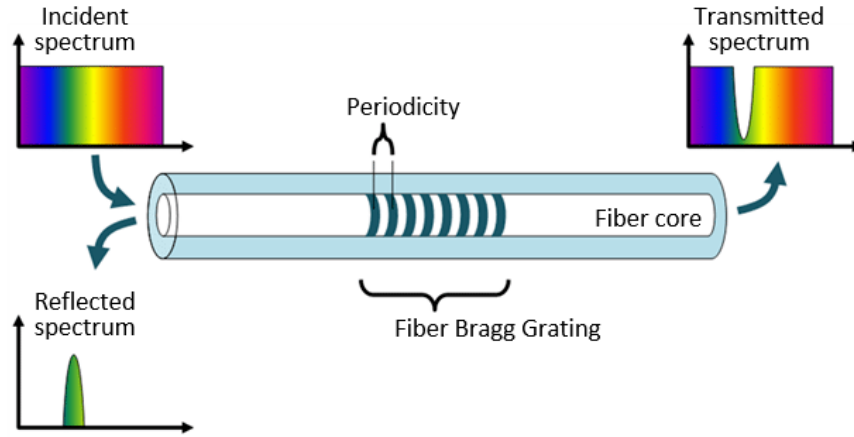


Figure 2.6: FBG sensor

When external conditions cause the material to expand or compress, the grating step length changes and consequentially the reflected wavelength change too, allowing the measurements of relative deformations and temperature.

$$\frac{\Delta\lambda_B}{\lambda_B} = k_\epsilon \Delta\epsilon + k_T \Delta T \quad (2.6)$$

Where the first term represents the strain effect on the fiber and the second term represents the effect of temperature.

The coefficient k_ϵ is determined by the physical elongation of the grating pitch and strain-optic coefficient of the fiber while the coefficient k_T is determined by the thermal expansion coefficient and the thermal-optic coefficient of the fiber.

The intensity of the sensor response, ie the intensity of the peak reflected by the grating depends on the length of the FBG itself, as each fringe reflects a small part of the incident signal which add up between them. Increasing the length of the sensor increases the number of fringes and consequently the intensity of the reflected wavelength.

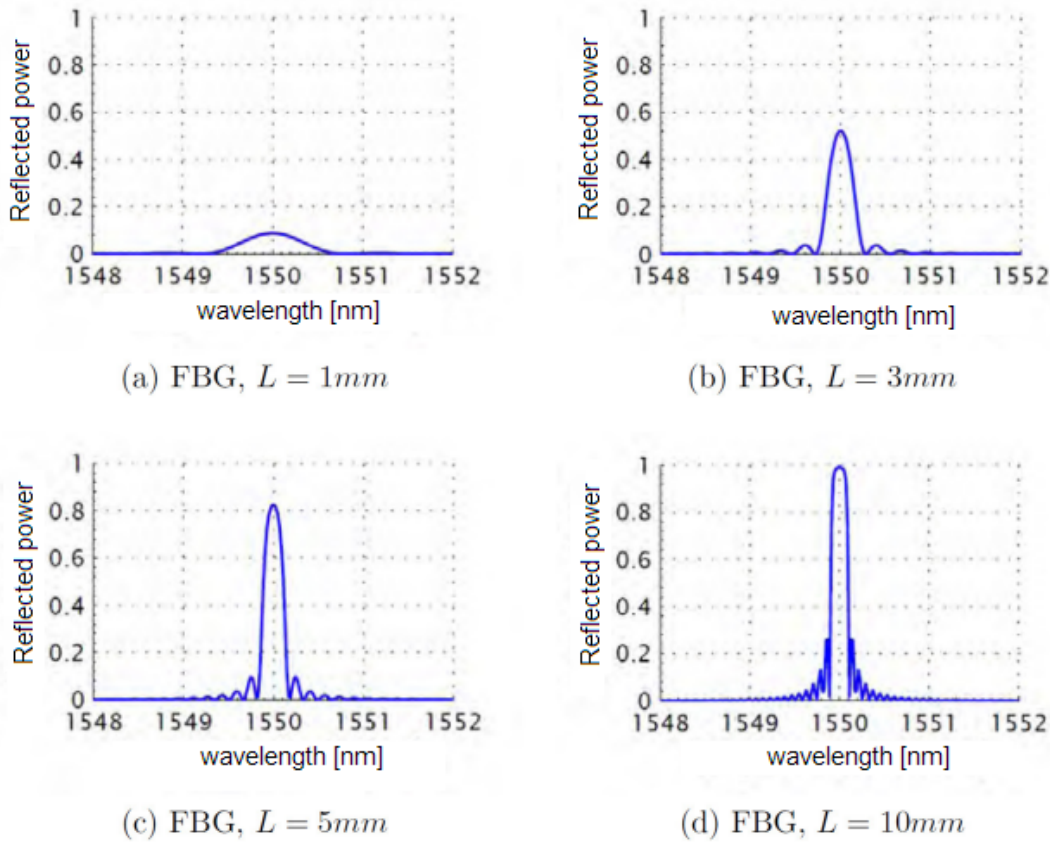


Figure 2.7: Reflected power vs FBG sensor length

2.2.2 Fiber Bragg Grating (FBG) strain dependence

The strain dependence of a fiber Bragg grating can be determined by differentiating the wavelength:

$$\frac{\Delta\lambda_B}{\lambda_B} = k_e\Delta\epsilon = (1 + p_e)\Delta\epsilon \quad (2.7)$$

Where p_e is the photoelastic constant (variation of index of refraction with axial tension). For optical fiber $p_e \approx -0.21$, meaning that strain Sensitivity of a FBG is given by the expression:

$$\frac{\Delta\lambda_B}{\Delta\epsilon} = k_e\lambda_B = 0.79\lambda_B \quad (2.8)$$

2.2.3 Fiber Bragg Grating (FBG) temperature dependence

Similarly to the strain dependence of a fiber Bragg grating, the temperature dependence can be determined by differentiating the wavelength expression:

$$\frac{\Delta\lambda_B}{\lambda_B} = k_T\Delta T = (\alpha + \zeta)\Delta T \quad (2.9)$$

Where α is the coefficient of thermal expansion of the fiber and ζ is the thermo-optic coefficient (dependence of the index of refraction on temperature).

For a temperature sensitivity approximation, we can assume that these values are constant for the temperature range: $\alpha = 0.55 \cdot 10^{-6}/^\circ C$ and $\zeta = 5.77 \cdot 10^{-6}/^\circ C$. Meaning that the approximate thermal sensitivity is given by:

$$\frac{\Delta\lambda_B}{\Delta T} = k_T\lambda_B = 6.3 \cdot 10^{-6}\lambda_B \quad (2.10)$$

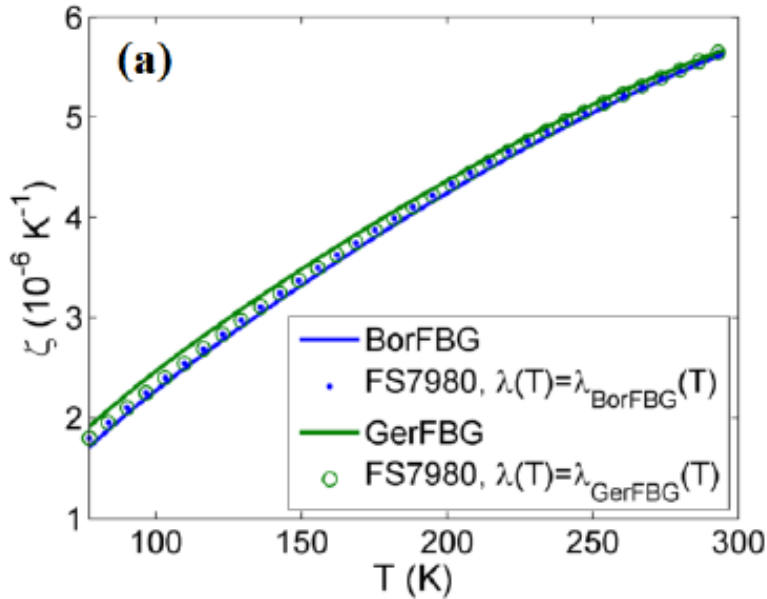


Figure 2.8: Thermo-optic coefficient vs Temperature

2.2.4 FBG Classification

The structure of the FBG can vary via the refractive index, or the grating period. The grating period can be uniform or graded, and either localized or distributed in a superstructure. The common structures for FBGs are the following:

- **Uniform gratings:** the modulation of the refractive index is constant in amplitude and period. This type of FBGs will be used for this thesis.

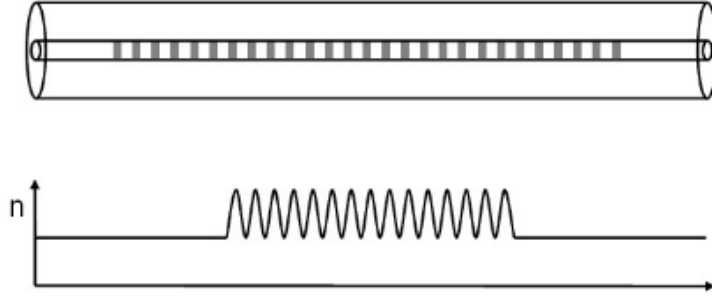


Figure 2.9: Uniform FBG grating

- **Apodized gratings:** the refractive index modulation is defined by a certain function, typically a Gaussian function or a cosine function. Apodized gratings offer significant improvement in side-lobe suppression while maintaining reflectivity and a narrow bandwidth.

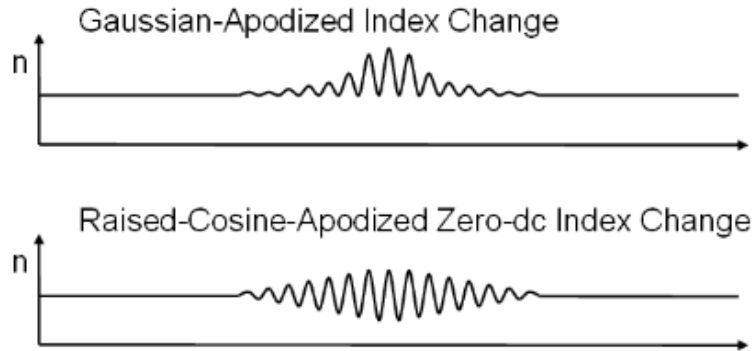


Figure 2.10: Apodized FBG grating

- **Phase-shifted gratings:** the refractive index modulation has a phase-shift (π) in the center of the FBG's length. Phase-shifted FBGs have interesting applications in optical communications and sensing due to their special filtering characteristics

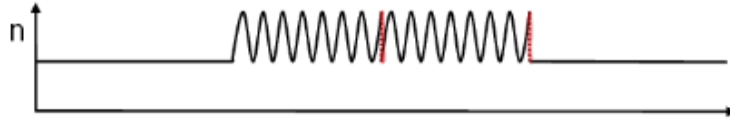


Figure 2.11: Phase-shifted FBG grating

- **Chirped gratings:** The amplitude of the refractive index modulation is constant, but the period is variable along the length. The reflected wavelength changes with the grating period, broadening the reflected spectrum.

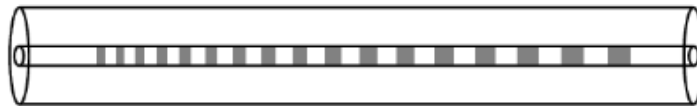


Figure 2.12: Chirped FBG grating

- **Tilted gratings:** The refractive index modulation is tilted by a certain angle to the optical axis. The angle of tilt in a TFBG has an effect on the reflected wavelength, and bandwidth.

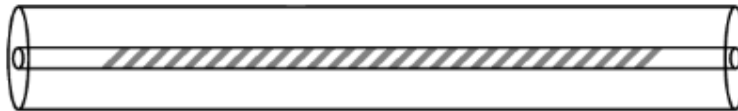


Figure 2.13: Tilted FBG grating

2.2.5 FBG Fabrication

Grating inside the fiber can be formed using various methods. The writing process is the process of creating gratings in the fiber core. Internal and external writing are the two main types of writing. Hill was the first to demonstrate internal writing with an argon ion laser, exposing the photosensitive fiber by coupling light in the fiber core. Because the change in refractive index noticed is so slight, this style of writing is ineffective and consequently rarely employed. The most frequent approach is externally inscribed writing, which can be done in a variety of ways, including the interfering beam method, phase mask technology, point-by-point technique, and line-by-line technique.

- **Interfering beam method:** A prism or a beam splitter is used to divide the light from a laser into two distinct beams. The two beams propagate until they come into contact with two mirrors, which reflect them onto the fiber. The writing process is carried out as the interference pattern on the fiber is being generated. The wavelength of light and the half angle between the interfering beams determine the period of the interference pattern. The main disadvantage of this technology is that it is difficult to keep the laser beam aligned for greater coherency.

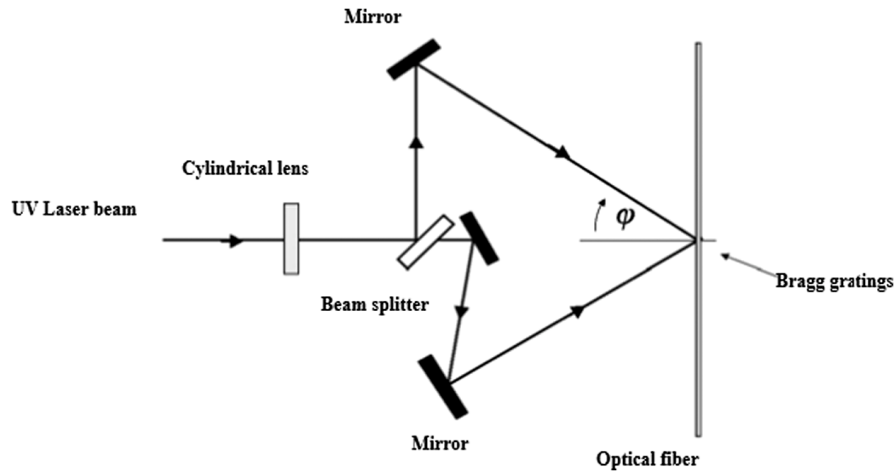


Figure 2.14: Interfering beam method

- **Phase mask technology:** a laser is used as a source to the phase mask layer, which is made using holographic technology or electron beam lithography. The phase mask's beam is diffracted into a number of different beams. The interference pattern is created on the fiber that is near the phase mask layer. A Krypton fluoride (KrF) excimer laser with a wavelength of 248 nm is employed as a light source in phase mask technique for writing because

UV radiation in the 228 to 253 nm spectral region is thought to be extremely effective in causing change in refractive index of fiber.

Phase Mask FBG Writing Schematic

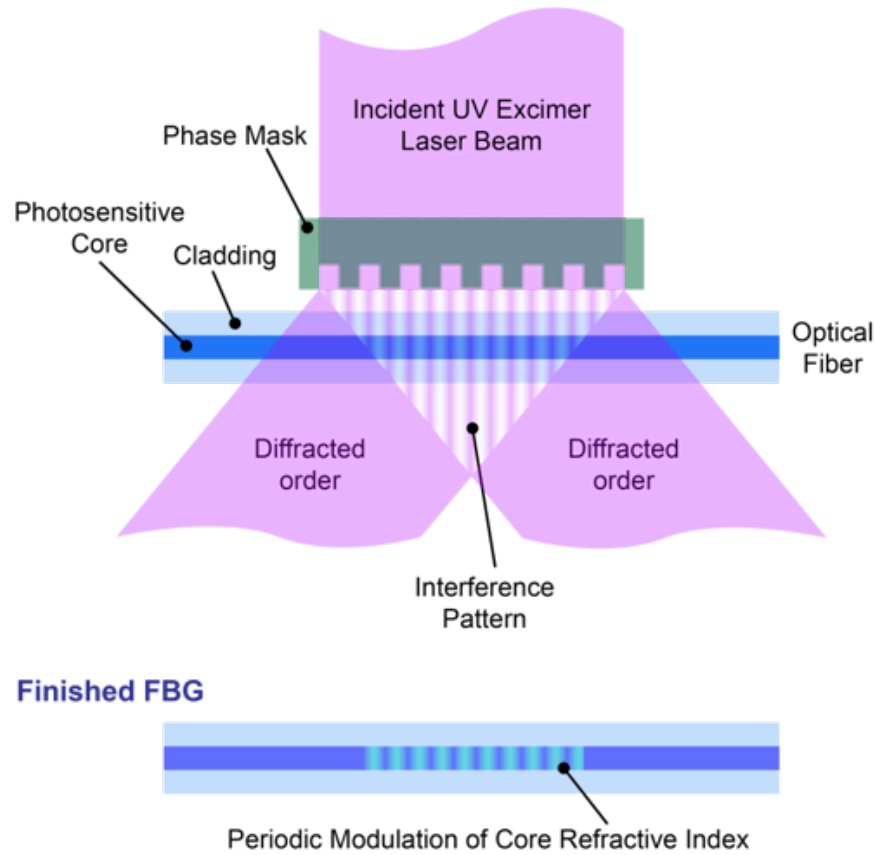


Figure 2.15: Phase mask technology

- **Point by point technique:** This technique uses a highly focused femtosecond laser system (FSL). The FSL sends highly energetic and high-precision pulses directly into the optical fiber core. The narrow duration of the pulse ensures that the permanent index change in the core remains confined. To form grating, a mechanically controlled translation stage is involved that repeats the same process every time.

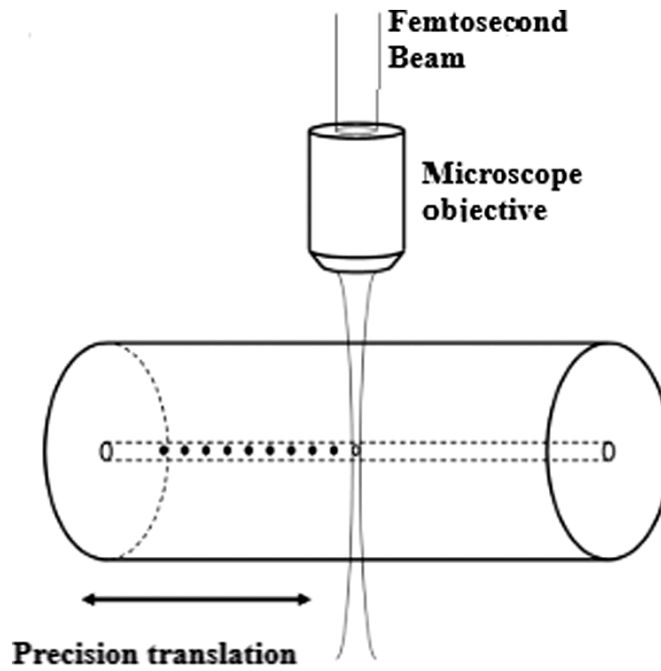


Figure 2.16: Point by point technique

2.2.6 Temperature effect compensation

FBG sensors measure both mechanical deformation and thermal expansion at the same time; it is therefore necessary to decouple the contributions that the two effects produce on the wavelength variation reflected.

Here are some methods for compensating for the temperature effect:

- **Mechanical insulation by lead-through capillary pipe:** It consists of inserting two FBG sensors close together, inside a rigid pipe. The first sensor is bonded to the pipe using epoxy, so as not to undergo deformation and to only detect the thermal expansion. The second sensor, on the other hand, is free to deform with the structure under examination, so as to detect both effects. By comparing the measurements of the two sensors it is possible to isolate the effect of deformation.

This method involves some problems, such as high intrusiveness (which undermines the ease of integration into composite materials), and the relative difficulty in creating the rigid bond between the first FBG and the walls of the capillary tube.

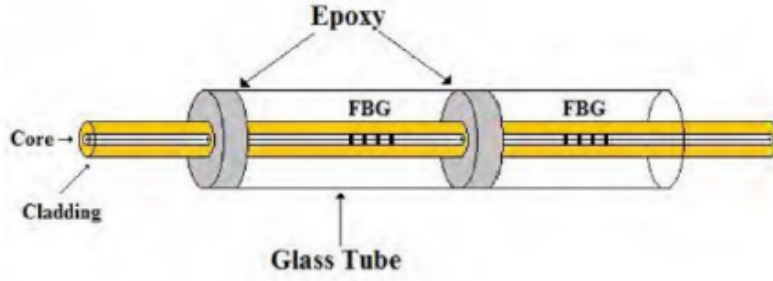


Figure 2.17: Mechanical insulation by lead-through capillary pipe

- **Mechanical insulation by non-lead-through capillary pipe:** Compared to the previous method, in this case the capillary pipe is located at the end of the fiber and a single sensor is placed inside of it. Similarly to the previous case, this sensor is constrained to the walls of the pipe using epoxy, so as to only detect the thermal expansion. A second FBG, positioned along the fiber, measures the two coupled contributions. From the comparison of the two measurements, the deformation can be isolated.

The capillary pipe is still intrusive but this method is easier to be integrated into a composite structure, although it may involve a non-optimal arrangement of the fiber.

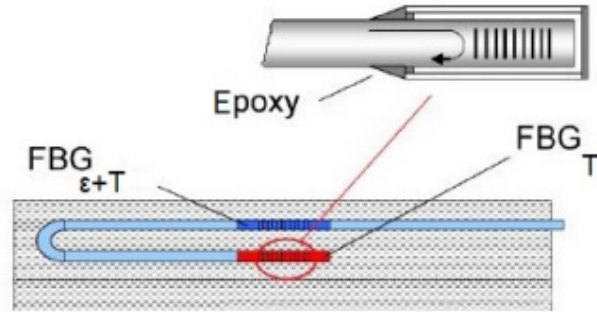


Figure 2.18: Mechanical insulation by non-lead-through capillary pipe

- **Thermal expansion coefficient modification:** This method involves a direct compensation of temperature effect. This can be achieved by the elimination of the k_T coefficient in equation 2.9, thus making the grating of the fiber temperature invariant. To obtain this compensation, the FBG sensor has to be incorporated in a crystalline liquid polymeric material with a thermal expansion coefficient equal and opposite to the one of the fiber.

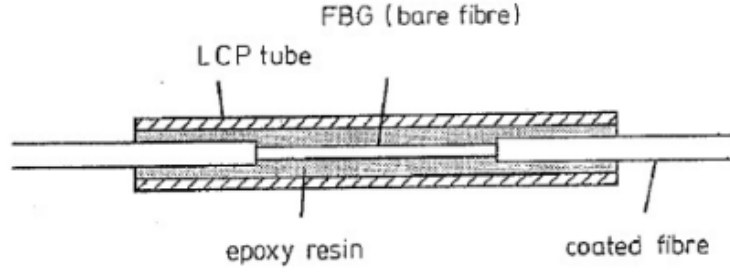


Figure 2.19: Thermal expansion coefficient modification

- **Overwritten FBGs:** This method consists of writing two different gratings with different λ_B on the same segment of the fiber, in order to obtain different values of k_T and k_ϵ (equations 2.7 and 2.9).

The wavelengths of the two gratings should be sufficiently spaced ($\Delta\lambda_B > 200nm$) to obtain a well-posed system.

$$\begin{Bmatrix} \Delta\lambda_{B1} \\ \Delta\lambda_{B2} \end{Bmatrix} = \begin{bmatrix} k_{\epsilon 1} & k_{T1} \\ k_{\epsilon 2} & k_{T2} \end{bmatrix} \begin{Bmatrix} \Delta\epsilon \\ \Delta T \end{Bmatrix} \quad (2.11)$$

The downside of this method is that two light sources are necessary to cover both wavelengths, increasing the costs.

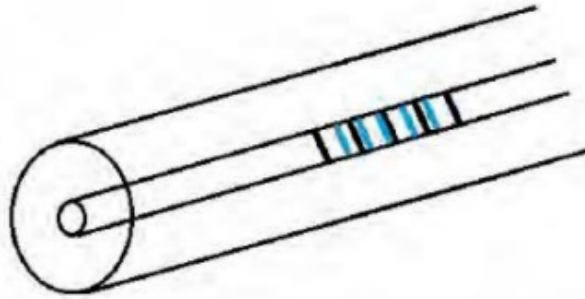


Figure 2.20: Overwritten FBGs

2.2.7 Data acquisition system

In order for the sensors to work properly a data acquisition system is necessary, which comprises an optical interrogator (such as a Single Board Interrogator SBI), a processing module (PM), a power supply and acquisition software.

The core component for the data acquisition system is the Optical FBG Interrogator which is capable of emitting light signals inside the fiber and also receives the reflected signals coming from the sensors and turns them into current.

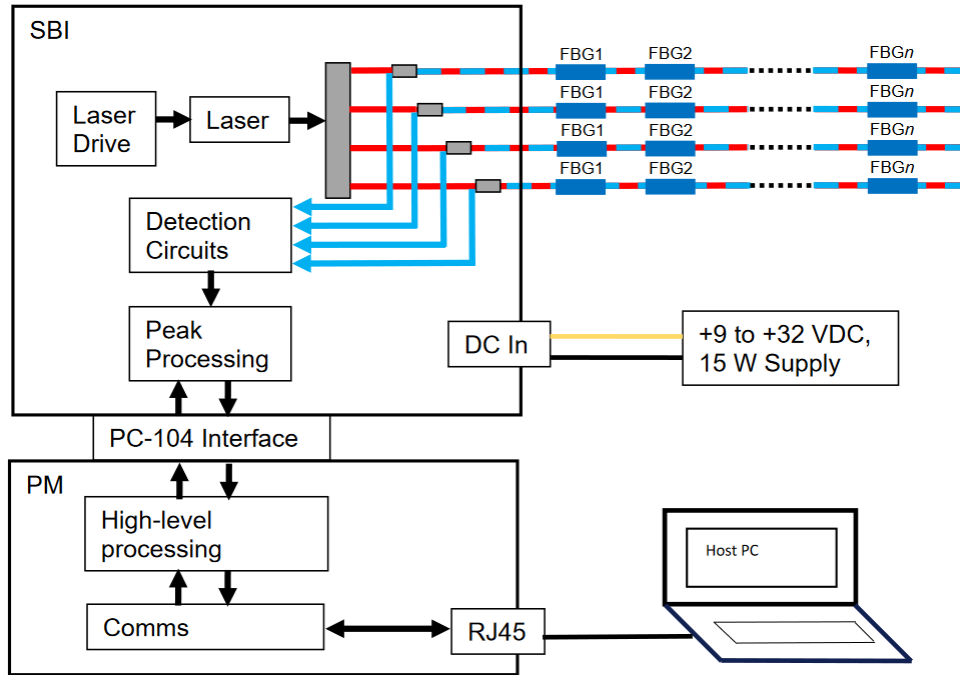


Figure 2.21: Data acquisition system layout (SmartScan interrogator)

The optical interrogator is made by the following parts:

- **Light source:** which can be either LED (light emitted diode), SLED (superluminescent LED) or LD (laser diode).
- **Wavelength filters:** used to select the specified wavelength
- **Optical isolator:** it is an optical component which allows the transmission of light in only one direction. It is used to prevent unwanted feedback.
- **Photodetector:** a p-n junction that converts light photons into current.
- **Circulators:** used to direct the source and the reflected signals to the right ports.

As said before, we can place multiple sensors on the same fiber channel as long as every sensor has a different characteristic wavelength λ_B , in order for the measurement to be associated to the right sensor. It is thus important to have wavelength filters for this purpose.

During the set-up phase, the interrogator perform a tuning process, identifying the correct λ_B for each sensor on each channel.

This allows the interrogator to select a specific FBG sensor by its wavelength.

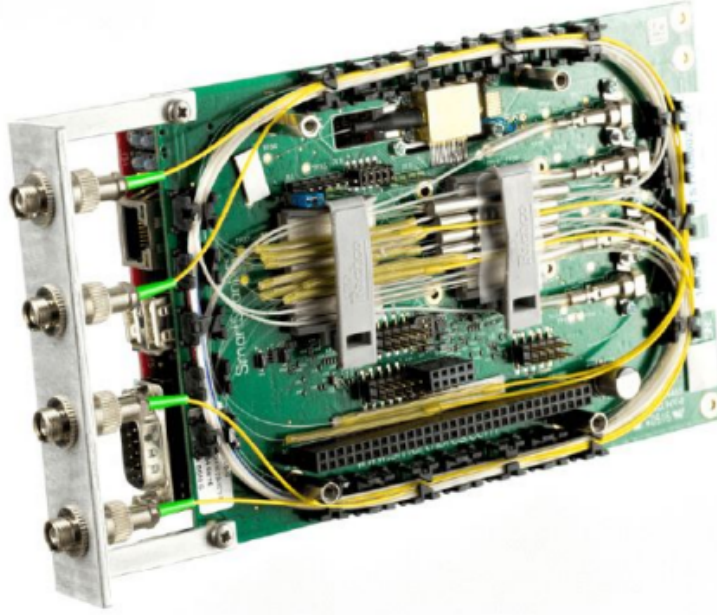


Figure 2.22: SmartScan SBI

The aircraft equips a SmartScan SBI interrogator with the following specifics:

Wavelength Range	Number of Optical Channels	Max Number of Sensors per Channel
1528-1568 nm	4	16

Scan Frequency (all sensors simultaneously)	Scan Frequency (per each sensor in turn)
2.5 kHz	25 kHz

Table 2.1: SmartScan SBI specifics

Chapter 3

ICARUS Team and project RA

3.1 ICARUS Team

ICARUS PoliTO is a student team of the Polytechnic of Turin. Of a purely aerospace nature, the ICARUS Team has more than 90 members, working on 3 different projects in 5 areas.

- **ACC:** design and construction of a competition UAV for a two-year university competition (Air Cargo Challenge) organized in Europe.

The ACC Team division develops and builds an UAV with specific characteristics to race in the competition, which develops on kilometer and hourly efficiency, in addition to the transport of a defined payload. During the last edition (Stuttgart 2019) the Team placed 6th;

- **DART:** design and construction of a rocket model with solid propellant. The ultimate goal of the project is to break the record of 2500m of apogee for an amateur carrier. DART is equipped with a solid propellant engine consisting of 65% of KNO₃ and 35% of sorbitol, capable of generating up to 1500 [N] with a maximum chamber pressure of 10 [Mpa].
- **RA:** Design and construction of a UAV powered solely by solar panels, allowing a virtually unlimited range. The ultimate goal of this project is to complete a trans-oceanic flight in complete autonomy.

For this thesis the sensors will be applied to project RA inside the wing structure, it is thus important to introduce the aircraft, paying particular attention to the wing geometry and structure.

3.2 Project RA



Figure 3.1: Render of RA aircraft

Record Aircraft (RA), born in 2017, is an ambitious project that is part of the current industrial and energy development. Its ultimate goal is to create a UAV entirely powered by solar energy.

The aircraft will be equipped with solar panels with virtually unlimited autonomy. The long-term goal is to fly across the Atlantic Ocean.

The aircraft, therefore, shall not be bound to meet particular speed requirements neither the ability to perform complicated maneuvers, but, rather, must have high aerodynamic efficiency, simple controllability and possibly the ability to carry small payloads of various types (cameras, sensors, telecommunication systems ...). It will therefore have the characteristics and appearance of a glider.

3.2.1 Aircraft Description

The aircraft will have a conventional high-wing configuration, with a span of approximately 5 meters, tear drop-shaped fuselage, just over two meters long and a T-tail about half a meter high.

The main landing gear is a single wheel on the fuselage centerline, with unsprung tailwheel and dolly system for taxiing phases.

A brushless electric engine will be mounted on the nose, equipped with a propeller of 50 centimeters in diameter.

Access to the internal compartment is possible through an opening on the top of the fuselage.

The external surfaces and the main structural components will be built in carbon fiber composite material and will be manufactured internally by the team.

The battery pack (Li-ion batteries) will be allocated inside the wing structure.

The solar panels will be mounted on the top of the wing surface. Gallium arsenide (GaA) panels will probably be used with a power density of approximately 250 W/m^2 and a power-to-weight ratio of 1 W/g (estimated).

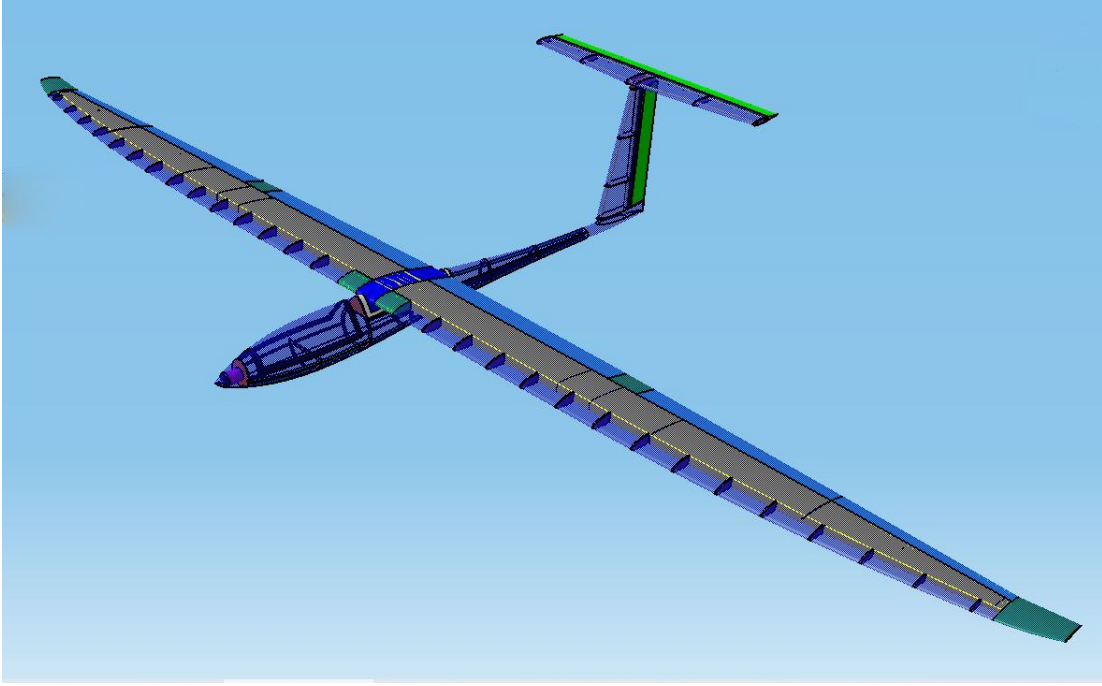


Figure 3.2: CAD model of RA aircraft

Estimated mass	15 [kg]
Cruise speed	15 [m/s]
Maximum aerodynamic efficiency	25
Wing surface	1.5 [m ²]
Wing span	5 [m]
Mean aerodynamic chord	0.3 [m]
Fuselage length	2 [m]
Engine power	1650 [W]
Propeller diameter	46 [cm]

Table 3.1: General characteristics (RA aircraft)

3.2.2 Wing aerodynamic and geometry

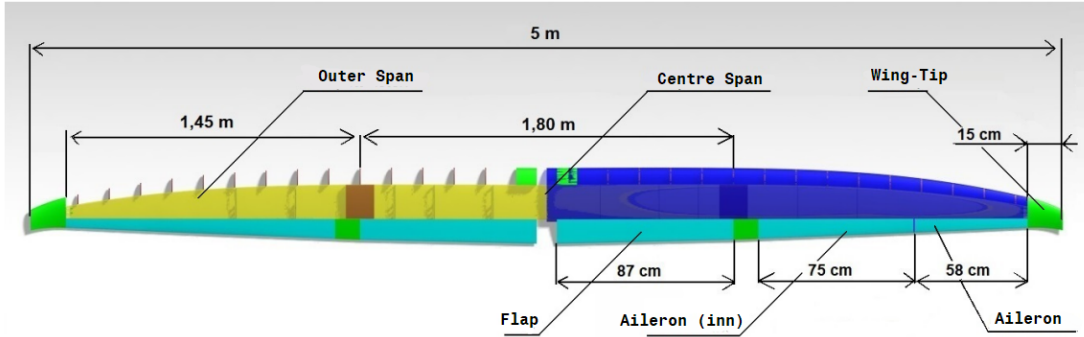


Figure 3.3: CAD model (top view) of the wing

In order to complete an endurance mission, such as that of the Record Aircraft (RA), it is strictly necessary that the aircraft soars in the air as efficiently as possible; because of this reason, the aerodynamic aspect is certainly among the top priorities of the project.

The planform of the wing is a derivation of Schumann's wing, widely used for gliders for its aerodynamic efficiency, allowing a drastic reduction in induced drag. Changes have been made to the sweep angle along the span, in order to optimize lift distribution and obtain an almost elliptical configuration. The wing also has a bit of twist near the tip region while dihedral angle is zero.

The wing can also mount wing-tip devices in order to minimize the interaction between upper and lower part of the wing to improve efficiency.

The chosen airfoil is the *Selig 3002-099-83*, a profile for low Reynolds number with a limited camber on the top in order for the solar panels to work efficiently, capturing solar rays at an ideal angle, without compromising aerodynamic efficiency.

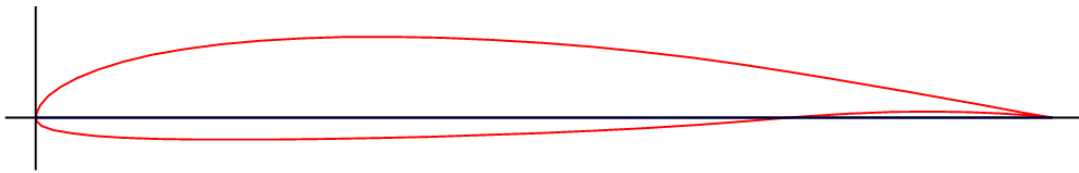


Figure 3.4: Airfoil - Selig 3002-099-83

The following tables contain the characteristics of the airfoil and the main coefficients obtained through CFD analysis.

Max thickness	Max camber	Root chord	Tip chord
9.9% at 30.6% chord	2.3% at 49.8% chord	0.372 [m]	0.146 [m]

Table 3.2: Airfoil characteristics

$C_{L_{max}}$	C_{L_0}	C_{L_α}	C_{M_0}	C_{D_0}	$C_{D_{min}}$
1.24	0.34	5.39	-0.078	0.016	0.014

Table 3.3: Aerodynamic coefficients ($V = 15m/s$)

The mobile surfaces are on the trailing edge: flaps near the fuselage and two ailerons per each wing.

3.2.3 Wing structure

Unlike the conventional half-shell configuration with spars along the span and ribs, for RA aircraft a different configuration has been chosen which is more similar to a shell configuration.

This choice has been made in order to obtain as much free volume as possible inside the wing to allow the allocation of the batteries and other subsystems and also to reduce structural weight.

Since the battery packs are the heaviest elements on-board, their placement on the wing is fundamental for flight mechanics. In this configuration it is possible to obtain a high moment of inertia on the roll axis, making the response to commands or disturbances slower, and therefore predictable and easily manageable.

The wing structure has been designed to transfer almost the entire load on a closed section that traces the shape of the airfoil, cutting off leading edge and trailing edge area.

An external skin is then glued to the wing box to restore the airfoil shape and fulfil the aerodynamic functions.

This type of structure was feasible thanks to the high aspect ratio of the wing and the small size.

The structure will be manufactured using carbon fiber composite materials.

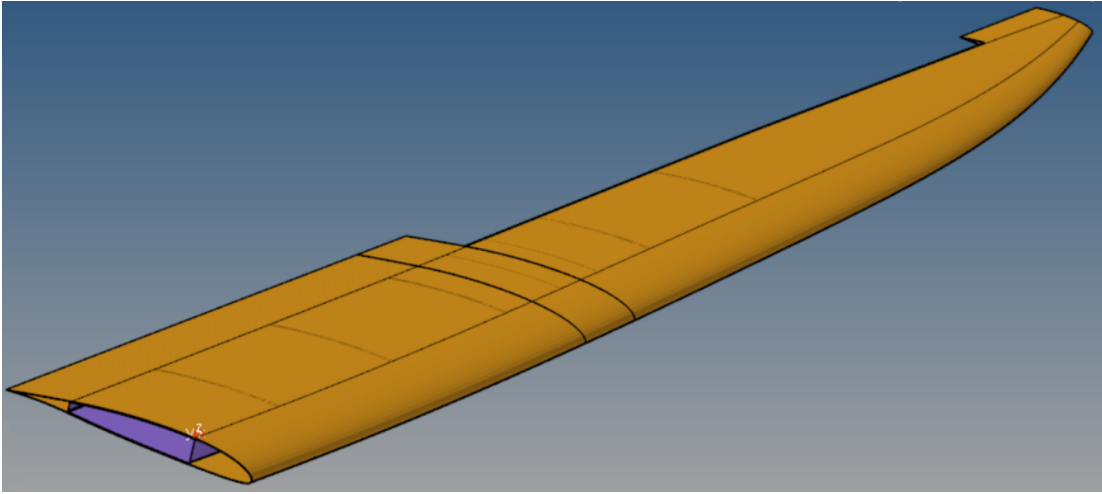


Figure 3.5: Wing structure CAD

The structure consists of these main components:

- **Wing box:** This element contributes to the bending and torsional stiffness of the wing. The vertical faces are positioned approximately at the fore-quarter and hindquarter of the profile and virtually fulfil the spar's function while the upper and lower faces traces the airfoil shape and close the load paths. The component is realized through in-mold lamination and vacuum forming with two layers of carbon fiber oriented $\pm 90^\circ$.
- **Skin:** The external skin allows to maintain the right aerodynamic shape. Foam is used to fill areas subject to high aerodynamic loads or exposed to the risk of accidental damages. The component is realized through in-mold lamination and vacuum forming with two layers of carbon fiber oriented $\pm 45^\circ$.
- **Sandwich ribs:** Sandwich ribs helps against contact pressure loads and contributes to the torsional stiffness of the wing. They also support the battery packs. The ribs are realized with three layers of carbon fiber and Rohacell foam core. A total of five ribs are placed along the span for each half-wing.

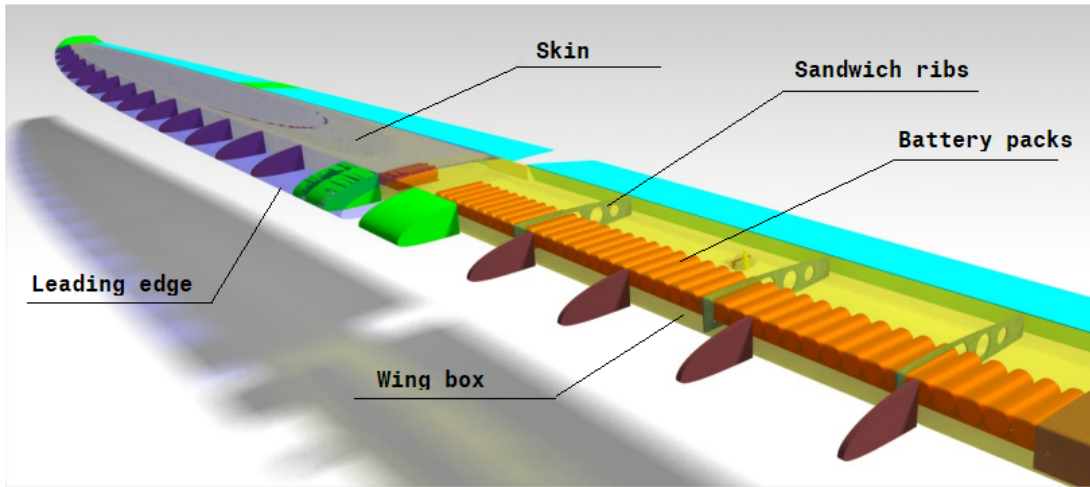


Figure 3.6: Wing structure detailed CAD

FBG sensors will be integrated inside the wing box during the lamination process, allowing a direct monitoring of strain and protecting the fibers against accidental damage.

The wing has been divided into three parts in order to easily transport and disassemble the aircraft. This requires a joint between the external and the central parts. The joint will be realized using a bayonet mechanism, a reinforcement integrated inside the central wing box is inserted in the external wing box.

It is important to note that the presence of the joint interrupt the optical fiber lines, it is thus necessary to have connectors to ensure the continuity of the fiber lines.

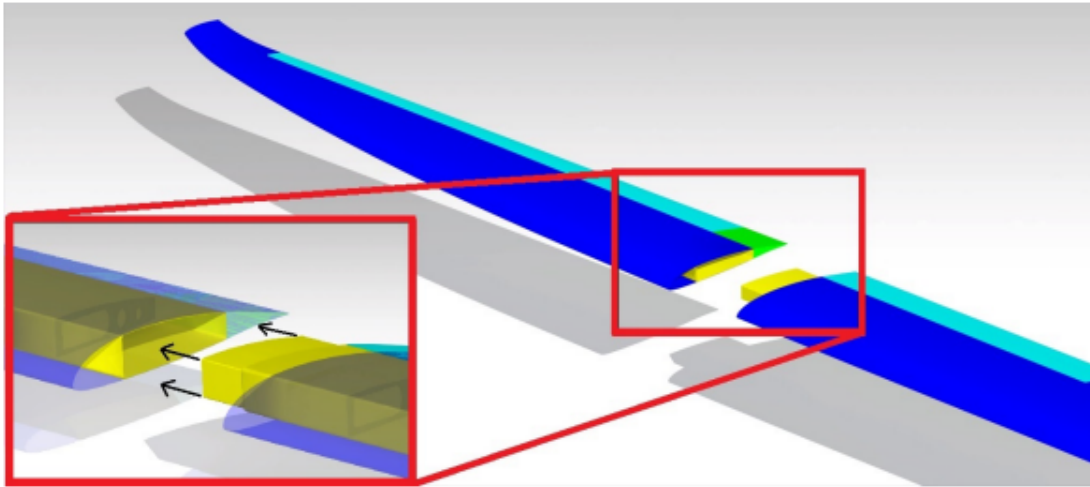


Figure 3.7: Wing joint detail

The mobile surfaces are installed on the trailing edge of the wing by cutting the rear part of the skin from the rest of the structure. This part of the skin will rotate and act as a mobile surface.

There are no hinges that connect the mobile surfaces to the structure, duct tape will be used to allow the rotation of the parts.

This architecture is simple but it's important to note that aerodynamic loads will be transferred to the structure exclusively through the servo-actuators connected to the mobile surfaces.

Each half-wing has a flap on the inner part near the fuselage and two ailerons.

For each mobile surface there is a single servo-actuator connected in the centre of it and fixed to the wing box.

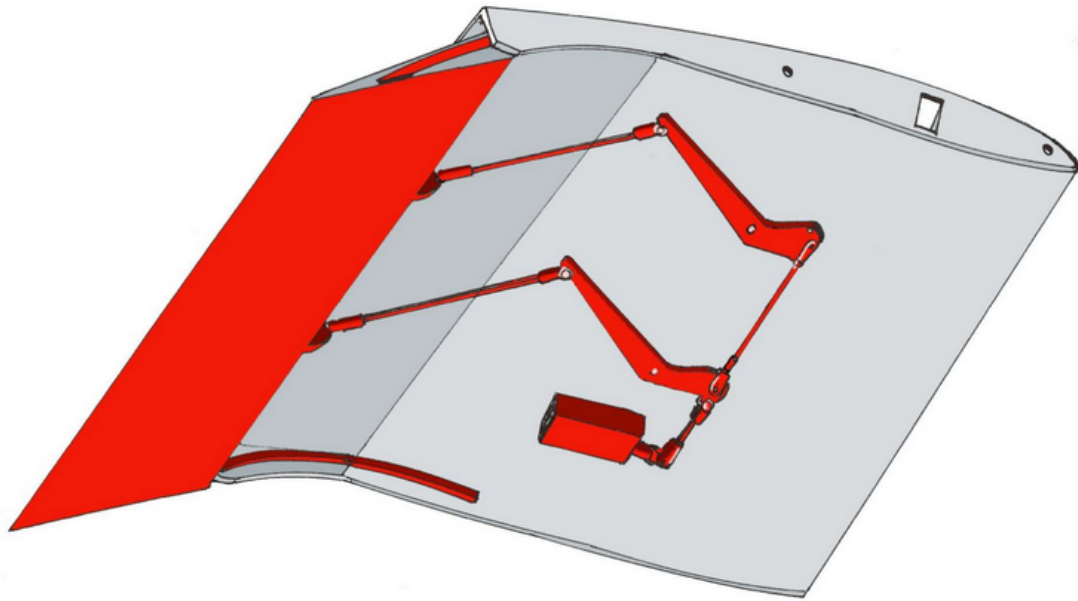


Figure 3.8: Flap mechanism (Sample from ANUBI (2017) model)

Chapter 4

Flight conditions, FEM and CFD models

4.1 CFD model

For this thesis a CFD model developed by ICARUS has been used in order to compute aerodynamic loads and hinges moment to varying of flight conditions and mobile surfaces deflection.

CFD analysis are based on highly non linear equations that are solved by high performance computing. The software used for the solution is Siemens Star ccm+.

4.1.1 Introduction to Computational Fluid Dynamics

The technique of mathematically modeling a physical event involving fluid flow and solving it numerically utilizing computing power is known as computational fluid dynamics (CFD).

The investigation of fluid flow in terms of physical parameters such as velocity, pressure, temperature, density, and viscosity is carried out in a CFD software analysis. Those features must be examined simultaneously in order to effectively develop an accurate solution for a physical phenomenon linked with fluid flow.

To study the fluid flow, a CFD software tool employs a mathematical model of the physical case and a numerical algorithm.

The mathematical model must be verified in order to establish an accurate case for solving the problem. Furthermore, determining appropriate numerical methods is essential for generating a solid result.

The main framework of the thermo-fluids analysis is guided by governing equations based on the conservation law of fluid physical characteristics. The basic equations are the three laws of conservation:

- Conservation of Mass: Continuity Equation

- Conservation of Momentum: Newton's Second Law
- Conservation of Energy: First Law of Thermodynamics or Energy Equation

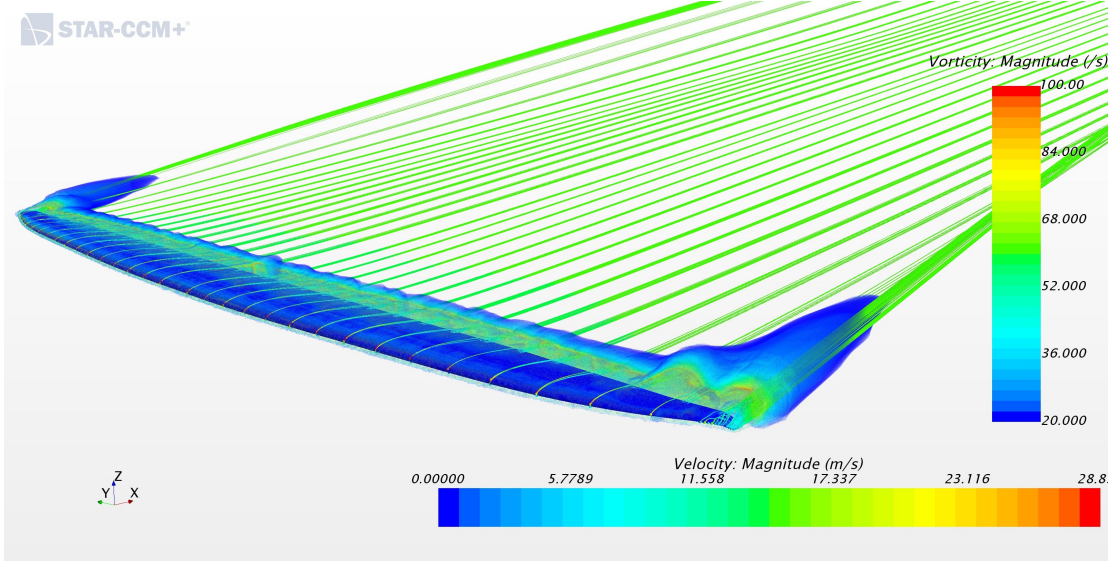


Figure 4.1: RA - Wing vorticity distribution ($\alpha = 10^\circ$)

The equation for the Conservation of Mass is specified as:

$$\frac{D\rho}{Dt} + \rho(\nabla \cdot \vec{v}) = 0 \quad (4.1)$$

where ρ is the density, v the velocity and ∇ the gradient operator. Conservation of momentum is given by:

$$\frac{\partial(\rho\vec{v})}{\partial t} + \nabla \cdot (\rho\vec{v}\vec{v}) = -\nabla p + \nabla \cdot (\bar{\tau}) + \rho\vec{g} \quad (4.2)$$

where p is the static pressure, $\bar{\tau}$ is the viscous stress tensor and $\rho\vec{g}$ is the gravitational force per unit volume.

Conservation of Energy is the first law of thermodynamics which states that the sum of the work and heat added to the system will result in the increase of the energy in the system:

$$dE_t = dQ + dW \quad (4.3)$$

where dQ is the heat added to the system, dW is the work done on the system and dE_t is the increment in the total energy of the system.

4.1.2 Model geometry

A CAD model has been prepared for the analysis and then imported on star ccm+. The CAD model consists of the right half of the aircraft, cut from the longitudinal plane of symmetry, this allows to obtain the results using half of the processing power.

The geometry has been lightly modified in order for the analysis to run smoothly. Trailing edges of the wing and tail and some other sharp edges have been rounded a little, so as not to cause any problem during the solution phase. Surfaces are then meshed inside star ccm+ and the values for flight speed, angle of attack and mobile surfaces deflection are set for every flight condition analyzed.

The simulation domain is a semi-sphere of 20 m radius with the aircraft positioned on the centre.

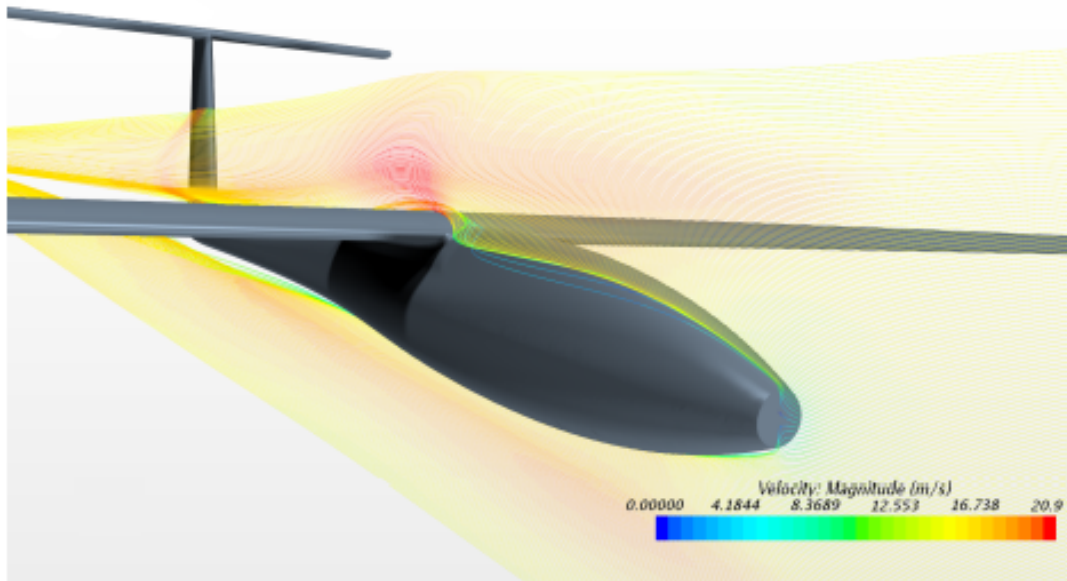


Figure 4.2: RA - Longitudinal symmetry plane velocity distribution

4.1.3 Simulation parameters and boundary conditions

The flow model used for the simulations has the following characteristics:

- **Steady:** Steady-state flow refers to the condition where the fluid properties at a point in the system do not change over time.
- **Compressible:** Density is considered a function of temperature and pressure.
- **Turbulent:** Flow above a certain Reynolds number is considered as turbulent. Turbulent flow is characterized by recirculation, eddies, and apparent randomness.

The solution is obtained through separated flows model, which uses decoupled equations to calculate velocity and pressure. This is possible only with low Mach numbers, to reduce simulation time and complexity, maintaining accurate results.

Boundary conditions are defined as follows:

- **Free stream condition:** on the domain edges
- **No-slip condition:** on the aircraft surfaces
- **Symmetry condition:** on the xz plane

Turbulence is simulated using *SST k -omega* method with $\gamma - Re_\theta$ transition model, this method uses different equations depending on the distance of the flow from the walls, allowing an appropriate modeling for the whole domain.

The following set of equations are then used to run the simulations:

- **Reynolds-Averaged Navier-Stokes equations (RANS):** are time-averaged equations of motion for fluid flow. The idea behind the equations is *Reynolds decomposition*, whereby an instantaneous quantity is decomposed into its time-averaged and fluctuating quantities.
- **Partial differential equations (PDE) for turbulence:** either *Wilcox k -omega* (boundary layer) or *k -epsilon* (free stream). Both are composed by two PDE and are used to simulate turbulence by the rate of dissipation of turbulent kinetic energy.
- **Blending function:** it allows to use the correct turbulence model depending on the distance of the flow from the walls.

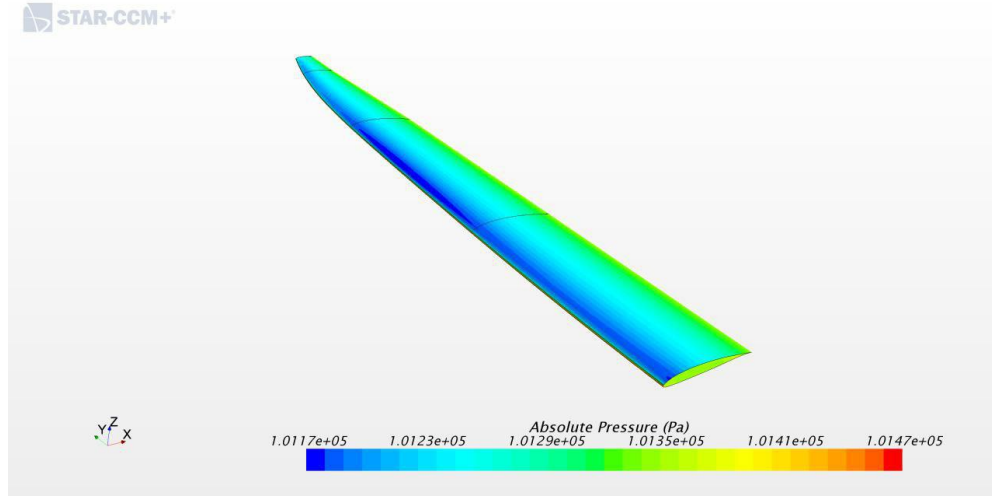


Figure 4.3: Wing pressure distribution

4.1.4 Mesh

The solution domain is divided into several sub-domains, known as cells, in order to conduct an analysis. Mesh refers to the combination of these cells in the computational framework.

Mesh is the technique of breaking down a domain into small cells or parts in order to apply a mathematical model based on the premise that each cell is linear. This means that within each cell, the behaviour of the variables that need to be solved must be assumed to be linear. This necessitates the use of a finer mesh in places where the physical attributes to be predicted are thought to be extremely variable.

Mesh structure errors are a common problem that leads to simulation failure. This could be because the mesh is too coarse, and instead of covering all effects that occur in a single cell element one by one, it covers numerous effects that change as the mesh gets finer. The mesh structure has a huge impact on the precision of the solution.

For the model used in this thesis, mesh is composed by polyhedral cells.

Polyhedral cells have multiple exchange interfaces between them, and are therefore more suitable for analyses in which it is not easy to predict the direction of the flow.

Considering the complexity of the model, polyhedral cells allows a reduction in the iterations needed to reach convergence, thus decreasing the simulation time and also obtaining lower values of residuals and a more precise solution. As said before it is necessary to have different cells size for areas where the physical properties are suspected to be highly variable.

For this reason larger cells are found in the periphery of the domain and a

finer mesh is adopted in areas with rapid changes in flow characteristics, such as the aircraft surfaces, the wake region and even smaller cells on the wing leading edge and trailing edge.

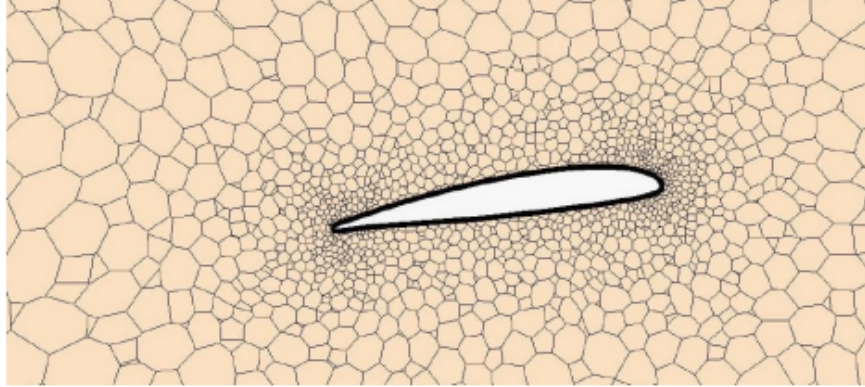


Figure 4.4: CFD mesh around wing profile

The mobile surfaces were placed in regions of space separated from the rest of the domain via interface elements.

In this way they can be rotated in the pre-processing phase and, then, divided into cells with a dedicated mesh, which interfaces with the global mesh, so that the entire mesh does not have to be recalculated at each simulation.

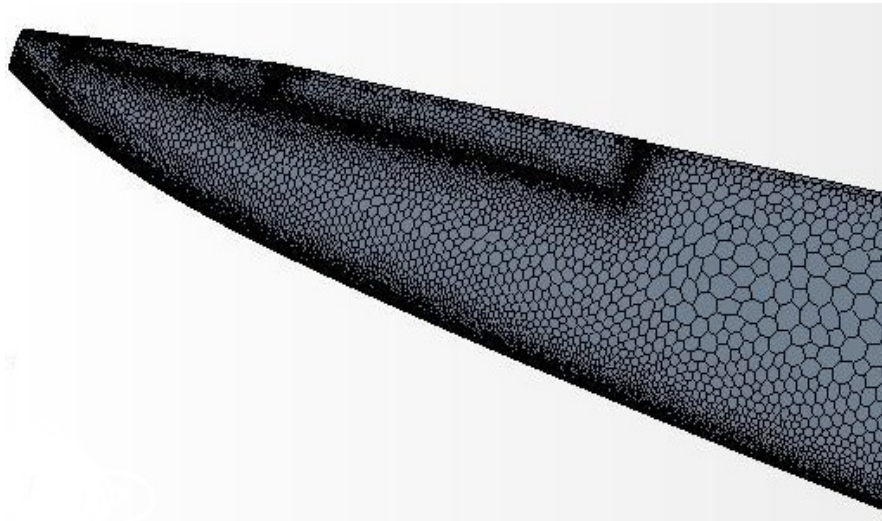


Figure 4.5: CFD wing mesh

In order to properly simulate the boundary layer region, twenty-five prismatic cells layers have been generated parallel to the surfaces. The height of the cells increases with the distance from the wall.

Inside the boundary layer the effects of viscosity prevail over inertial forces.

The thickness of this layer is estimated through preliminary simulations of a hundred iterations, and then be set correctly in the final mesh.

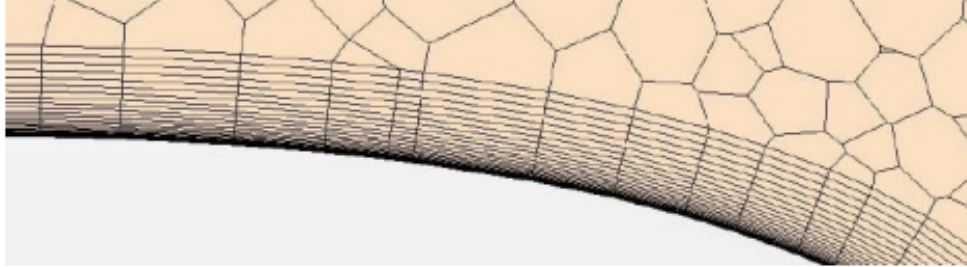


Figure 4.6: Prismatic layers detail

4.1.5 Limitations and assumptions of the model

As in every modeling problem, assumptions have to be made in order for the model to be as simple as possible, representing the physical phenomenon with the desired level of accuracy.

The separated flows model loses accuracy for high Mach numbers or high density, the flight conditions analyzed in this thesis never exceed those limits.

For the CFD model of this thesis, two main assumptions are made:

- **Steady-state flow:** in reality the flow around the aircraft is not steady, the physical quantities of the flow changes over time and interact with the aircraft dynamically. This means that the aerodynamic loads calculated with the analysis refers to an equilibrium condition and are therefore static loads.

This represents the main limitation of the model, even though the aircraft will perform very slow maneuvers in most of the cases.

For the objective of this thesis, that is building a sensors network, working with static loads is acceptable, even though for a more precise positioning of the sensors a dynamic model shall be used in the future.

- **Lack of propeller:** As said before, the aircraft is equipped with a propeller driven by an electric motor to provide the necessary thrust. To reduce the complexity of the model and the resources needed for the simulation the propeller has not been considered. In reality the propeller adds a rotational component to the flow which is highly asymmetric, leading to a different angle of attack between the two half-wings and between the right and left side of the elevator.

Protruding elements such as landing gear, antennas, external sensors are not considered by the model, neither do superficial defections and other small openings. These elements adds complexity to the model without significantly affecting the results.

4.1.6 CFD analysis input and output

Before running each simulation, the model receives a set of inputs that contain the value for velocity, angle of attack, mobile surfaces defections (ailerons, elevator) and environmental conditions.

The simulations were run with the same environmental conditions to have data consistency in order for the sensors network to be developed properly.

ISA (International Standard Atmosphere) model at sea-level was used:

- Temperature: $15\text{ }^{\circ}\text{C}$
- Pressure: 101325 Pa
- Density: 1.225 kg/m^3

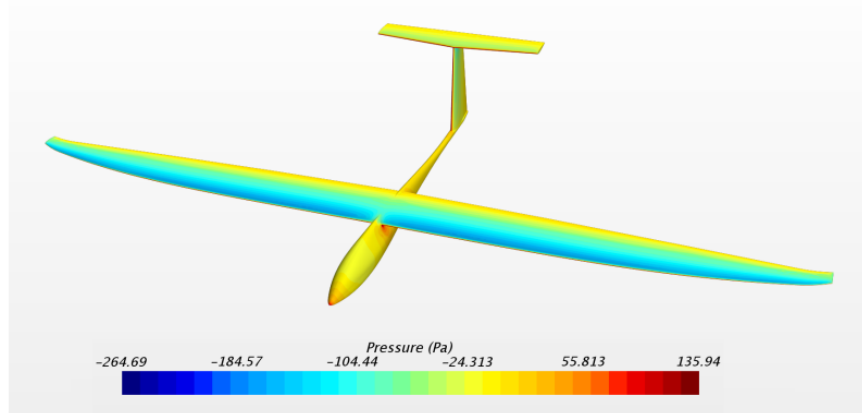


Figure 4.7: Pressure distribution ($V = 15\text{ m/s}$)

The velocity can be modified in the *physics* section of the software.

The angle of attack can be changed by modifying the free stream direction using three unit vectors. This is made possible by the use of a spherical domain and does not require mesh recalculation after angle of attack changes,

Surfaces defection can be modified in the *geometry* section of the software, mesh recalculation is necessary after the changes.

The mobile surfaces were placed in regions of space separated from the rest of the domain via interface elements.

In this way they can be rotated in the pre-processing phase and, then, divided into cells with a dedicated mesh, which interfaces with the global mesh, so that the entire mesh does not have to be recalculated at each simulation.

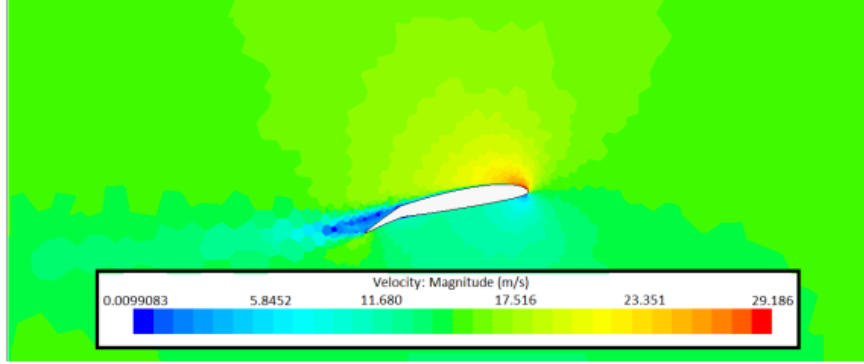


Figure 4.8: Velocity distribution around wing profile ($V = 15m/s$)

The output of the simulation contains the following information:

- Pressure distribution on the wing
- Pressure distribution on the mobile surfaces (inner ailerons, outer ailerons, elevator)
- Pressure distribution on the fuselage and tail
- Hinges moment on the inner and outer ailerons
- Forces and moment coefficients (c_x, c_z, c_m) needed for the calculation of aerodynamic derivatives on the longitudinal plane
- Forces and moment coefficients (c_y, c_n, c_r) needed for the calculation of aerodynamic derivatives on the lateral-directional plane
- Various graphic representations

For this thesis only some of the outputs are needed, in particular wing pressure distribution (static pressure) and hinges moments that will be applied to the FEM model.

4.2 FEM model

The aerodynamic loads obtained through CFD analysis are then applied to a FEM model developed internally in order to perform the structural analysis on the wing, obtaining stress and strain distribution that will be used by the sensor network.

The software used for the FEM analysis is Altair Hyperworks suite that contains Hypermesh to handle the geometry and Optistruct which is the solver part of the suite.

4.2.1 Introduction to Finite Element Method

The finite element method (FEM) is a numerical approach for performing finite element analysis (FEA) on any physical phenomenon.

To fully comprehend and quantify any physical phenomenon, such as structural or fluid behaviour, thermal transfer, wave propagation, and biological cell growth, mathematics is required. Partial differential equations are used to describe the majority of these processes (PDEs). Numerical algorithms have been developed over the last few decades to allow a computer to solve these PDEs, and one of the most popular today is the finite element method.

The finite element approach looked promising in the modelling of a variety of mechanical applications in aerospace and civil engineering at the outset. The finite element method's applications are only now beginning to realise their full potential.

The FEM approach is based on PDEs such as elliptic, hyperbolic, and parabolic PDEs. Boundary and/or initial conditions must be specified while solving these differential equations. The required inputs can be determined based on the type of PDE.

Finite difference methods (FDM) and variational (or energy) methods are the two basic approaches to solving elliptic PDEs. FEM belongs to the second group. The notion of energy minimization underpins most variational techniques.

The principle of energy minimization is the primary backbone of the finite element method; it states that when a boundary condition (such as displacement or force) is applied, only the configuration with the least total energy is chosen out of the numerous possible configurations that the body can take.

The domain is divided into small parts known as "elements," with each element's corner point referred to as a "node." At the nodal positions, the unknown functional $u(x)$ is determined. For each element to interpolate, nodal values are used to construct interpolation functions for values within the element. Shape functions are another name for these interpolation functions.

The equations are supplied to a solver to solve the system of equations once the matrix equations have been established. Direct or iterative solvers are commonly utilised, depending on the nature of problem.

4.2.2 FEM analysis overview

When importing a CAD model for a FEM structural analysis a few steps are required before the model is ready:

- **Geometry cleanup:** This step can be done either through a CAD software or directly inside a FEM pre-processor software like HyperMesh. In many cases, the imported geometry is not ready for meshing. It is important to remove "broken" surfaces, redundant surfaces, surfaces which are not stitched together and surfaces which are too small to be meshed in a reasonable way. In many cases it is also useful to remove features that are not necessary for the FEM analysis, like antennas, sensors and everything that gives little to no contribution to the results.
- **Meshing:** Once the geometry is in an appropriate state, a mesh will be created to approximate the geometry. Either a beam mesh (1-D), shell mesh (2-D) or a solid mesh (3-D) will be created. This meshing step is crucial to the finite element analysis as the quality of the mesh directly reflects on the quality of the results generated. At the same time the number of elements (number of nodes) affects the computation time but also the accuracy of the results, that is why choosing the right element size is important.
- **Reference frame:** It is important to define a global reference frame as well as check that local reference frames created for the various elements are coherent.
- **Defining materials and properties:** After meshing is completed, material (e.g. Young's Modulus, Poisson coefficient) and property information (e.g. thickness values) are assigned to the elements. The material can be defined as isotropic, orthotropic or anisotropic and laminates or composites can also be defined in this step.
- **Loads and boundary conditions:** Various loads and constraints can be added to the model to represent the loading conditions the part is subjected to. Different load cases can be defined to represent different loading conditions on the same model.

Connectivity matrices that links nodes to Degree of Freedoms (DOFs) for each element are computed.

Rotation matrix are then calculated in order to link the local reference frames to the global reference frame.

$$\{q^{L(e)}\} = [\Lambda^{(e)}] \{q^{G(e)}\} \quad (4.4)$$

Where $q^{L(e)}$ is the local DOFs vector, $\Lambda^{(e)}$ is the rotation matrix and $q^{G(e)}$ is the global DOFs vector.

The same process is repeated for stiffness matrices $K^{(e)}$ and loads and constraints vectors $F^{(e)}$.

$$[K^{G(e)}] = [\Lambda^{(e)}]^T [K^{L(e)}] [\Lambda^{(e)}] \quad (4.5)$$

$$\{F^{L(e)}\} = [\Lambda^{(e)}] \{F^{G(e)}\} \quad (4.6)$$

Matrices and vectors are then expanded and assembled in order for them to contain all the DOFs of the structure.

Assembling is the same as imposing the following conditions:

- Congruence of nodal displacements
- Balance of nodal forces

Finally a matrix system is obtained, which can be written as follows:

$$[K^{G(e)}] \{q^{G(e)}\} = \{F^{G(e)}\} \quad (4.7)$$

The output requested by the FEM analysis in this thesis are stresses and strains.

Von Mises criterion is used for the analysis. Von Mises stress is a value used to determine if a given material will yield or fracture. It is mostly used for ductile materials, such as metals. The von Mises yield criterion states that if the von Mises stress of a material under load is equal or greater than the yield limit of the same material under simple tension then the material will yield.

The maximum distortion energy theory originated when it was observed that materials, especially ductile ones, behaved differently when a non-simple tension or non-uniaxial stress was applied, exhibiting resistance values that are much larger than the ones observed during simple tension experiments. A theory involving the full stress tensor was therefore developed.

$$\frac{1}{6}[(\tau_{11} - \tau_{22})^2 + (\tau_{22} - \tau_{33})^2 + (\tau_{33} - \tau_{11})^2 + 6(\tau_{12}^2 + \tau_{23}^2 + \tau_{13}^2)] = \frac{S_y^2}{3} \quad (4.8)$$

Where τ is the stress tensor and S_y is the tension elastic limit.

4.2.3 Model geometry

The geometry of the model consists of the CAD model of the wing, with some modifications in order to eliminate all unnecessary features. Only the right half-wing is contained in the model to take advantage of symmetry, allowing to halve the computational cost.

Aerodynamic loads will be applied as a *pressure field* while hinges moment are modeled as concentrated loads.

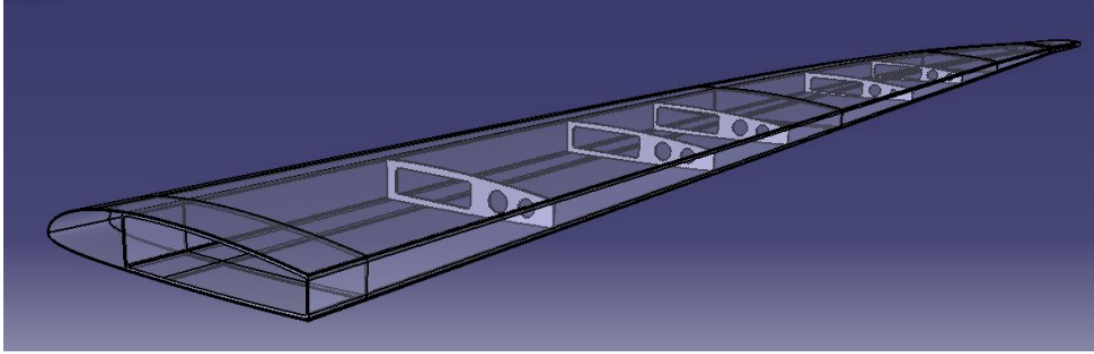


Figure 4.9: CAD model imported for FEM analysis

CAD model is composed by the wing box, 5 ribs and the skin.

To reduce computational costs the mesh is made using bidimensional elements, either quad or tria. Thickness is then added with the material properties through lamination where each layer of carbon fiber has a certain thickness value. These parts are imported into Hyperworks as bidimensional surfaces, passing through the mean plane of each component.

As said before some features have been removed from the original CAD model either because they are not necessary for the FEM analysis or add superfluous complexity:

- Filling foam (Rohacell) has been removed as its contribution to structural strength is negligible
- Openings to the various systems have been removed
- Batteries and servo-actuators have been removed
- The joint between the two parts of each half-wing has been removed
- Mobile surfaces have been removed as they do not contribute to structural strength. Loads exchanged with mobile surfaces will be added through hinges moment.

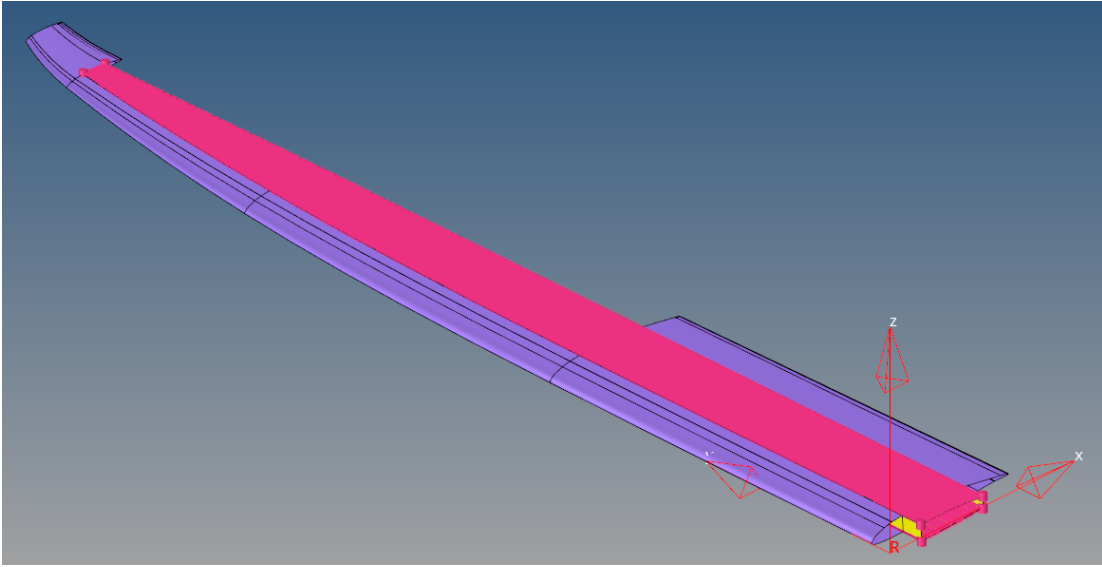


Figure 4.10: FEM model

4.2.4 Mesh

Meshing the wing in an appropriate way is an important step to obtain accurate results, the mesh has to be smooth and regular and has to use the simplest element type suited for the problem. The element size is a key parameter as coarse meshes require less computing resources but can sometimes return inaccurate results while a finer mesh is more accurate but is not always needed.

For the present case a bidimensional mesh has been chosen, using *quad* or *tria* elements. 2D elements are used when two of the dimensions are very large in comparison with the third dimension.

Quite often the geometry of thin walled 3D structures is simplified to a geometric model with lower dimensionality. This is typically called a mid-surface model. The mid-surface model is then meshed with 2D elements. Thus, there is no need for a detailed volume mesh as the thickness of the geometry is virtually assigned to the 2D elements. Mathematically, the element thickness is assigned with half in the $+Z$ direction (element top) and the other half in the $-Z$ direction (element bottom).

Using the software Hypermesh, *size and bias* mode has been used. This mode allows to set a base element dimension (3 mm in this model) and then the *biasing* function allows to obtain a finer mesh near the edges.

This setting is particularly useful for the leading edge and trailing edge and also near the tip region where the curvature of the skin requires a finer mesh to accurately follow the geometry.

The bias factor can be expressed as the ratio of the maximum element length

divided by the minimum element length.

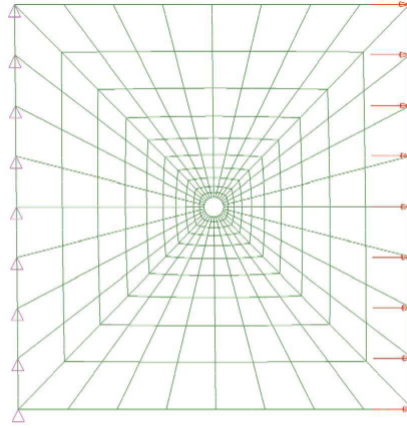


Figure 4.11: FEM bias setting towards the center hole (example)

As said before, the mesh uses both *quad* and *tria* elements:

- **Quad elements:** The strain in the x-direction is a linear function of its y-value. Quad elements produce more accurate results but meshing complex geometry can be difficult.
- **Tri elements:** The triangular element is called as a constant strain triangle (CST) element. Value of the stress tensor in the whole element will be constant. This may not be true in the real situation and may produce inaccurate results. The advantage of triangular elements is that they can better approximate complex geometry.

The combination of both elements allows better mesh flow, helps with mesh transition from a coarse mesh to a finer mesh and helps with meshing sharp areas while maintaining good accuracy.

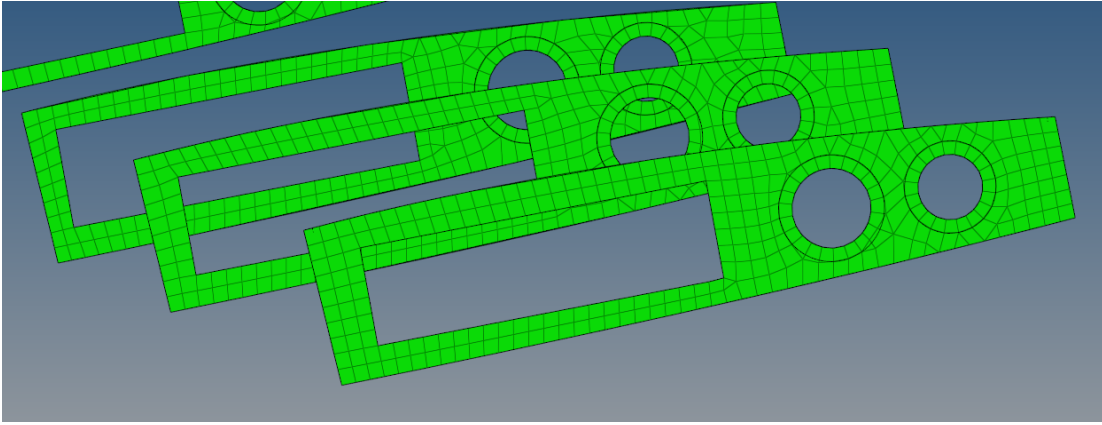


Figure 4.12: FEM ribs mesh



Figure 4.13: FEM skin mesh

4.2.5 Materials

Materials characterization has been made by ICARUS team recently with a series of tensile, compression and bending tests on various specimens. However, when Ing. Emanuele Frediani did his thesis job these data were not available and a different approach was used for his simulations.

The objective of this thesis is to develop a sensors network and increase the available dataset of simulations in order for the network to work properly.

For this reason, to have consistency in the simulations between the already analyzed flight conditions and the new ones that will be simulated in this thesis, it has been chosen to use the same approach used by Ing. Emanuele Frediani in his thesis job.

Otherwise all of the simulations had to be made again with the updated materials properties with little to no gain regarding the pursued objective.

In a future job, when the implementation of the system starts, it might be useful to update the FEM model with experimental materials data.

The approach used to estimate the material properties is based on literature data combined with the software Altair Multiscale Designer.

Multiscale Designer is a comprehensive framework for the development of highly predictive and computationally efficient material models for all material behaviors ranging from homogeneous isotropic to heterogeneous anisotropic. Multiscale Designer reduces the time and cost associated with the development of highly predictive material models with minimal experimental data requirements.

The fabric used for the wing is *TeXtreme Carbon fabric* $80g/m^2$ that will be combined with epoxy resin *SX10 EVO*, with the following proportions: 40% epoxy and 60% carbon fiber.

Density	Fiber diameter	Young's Modulus	Tensile Strength
1.79 [g/cm^3]	7 [μm]	240 [GPa]	4800 [MPa]

Table 4.1: TeXtreme Carbon fabric characteristics

Viscosity	Density	Young's Modulus	Glass Transition
1200 [$mPas$]	1.1 [g/cm^3]	3000 [N/mm^2]	55 [$^{\circ}C$]

Table 4.2: SX10 EVO epoxy characteristics at 20 $^{\circ}C$

The carbon fiber laminate's characteristics obtained through Multiscale Designer are shown in the table below.

E_x	E_y	G_{xy}
$8.5 \times 10^4 [MPa]$	$8.5 \times 10^4 [MPa]$	$1.156 \times 10^4 [MPa]$

Density	Kinematic Viscosity
$1.55 \times 10^{-3} [g/mm^3]$	4.268×10^{-2}

Table 4.3: Estimated carbon fiber laminate's characteristics

The lamination sequence that will be used for the various components is as follows:

- **Wing box:** 2 layers with $\pm 90^{\circ}$ orientation
- **Skin:** 2 layers with $\pm 45^{\circ}$ orientation
- **Ribs:** 8 layers each, 4 of them at $+45^{\circ}$ orientation and the other 4 at -45° orientation

The FEM model also requires a connection between the skin and the wing box, this has been modeled using rigid elements (RBE) to simplify the model.

In reality a layer of glue (approximately 200g, with a density of 1150 kg/m^3) will be used between the skin and the wing box.

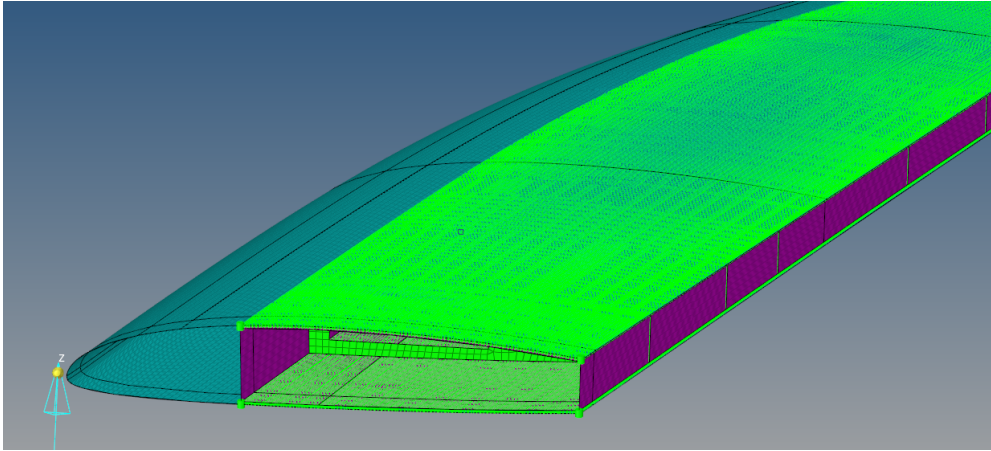


Figure 4.14: FEM model view

4.2.6 Loads and Boundary Conditions

The last step before running the analysis is to assign loads and constraints to the model, either applying them to the geometry or to the mesh.

- **Aerodynamic Loads:** Aerodynamic loads are obtained through CFD analysis described earlier. They consist of a pressure field applied to the wing and concentrated loads due to the forces exchanged between the ailerons and the structure.
- **Inertial Loads:** the weight of the structure is automatically calculated by Hypermesh using the materials data. Battery pack's weight is then added manually.
- **Constraints:** The half-wing contained in the model is integral with the other half-wing and is fixed to the fuselage body using an aluminium insert.

Pressure loads are saved by the CFD analysis in the form of coordinates (x, y, z) in mm and the corresponding pressure in that particular point in MPa. They're then applied to the model using a pressure field. The software computes and applies a normal vector to the surface of each element of the mesh, averaging its value to respect the relationships between the coordinates of the field and the position of each element.

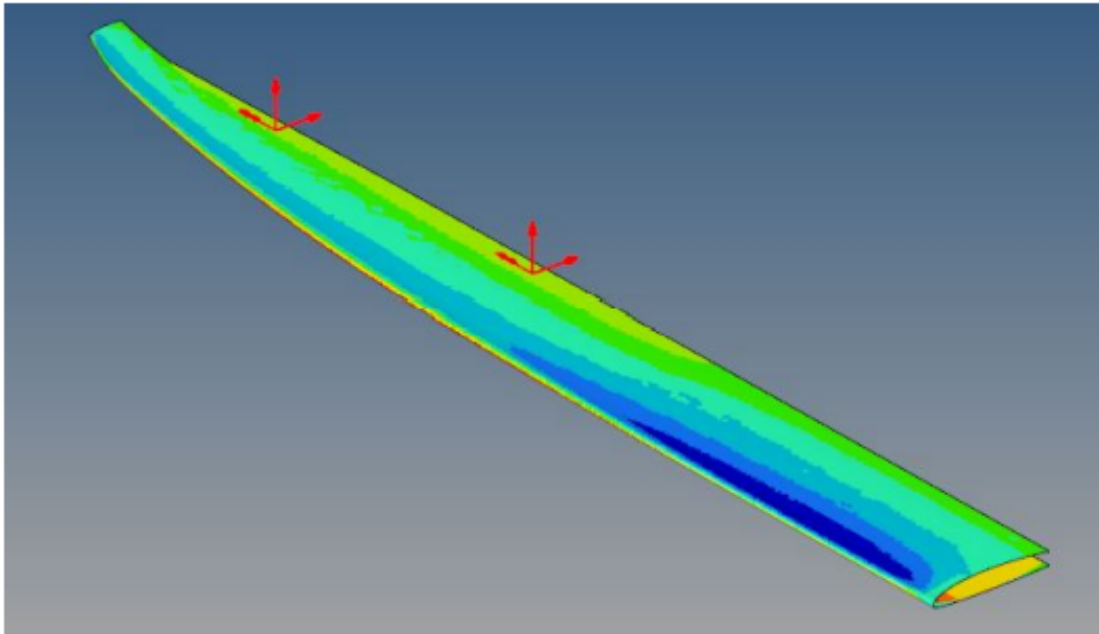


Figure 4.15: Pressure loads and concentrated loads

Forces exchanged with the mobile surfaces are applied using concentrated loads. In fact, there are no hinges connecting the ailerons to the wing box and loads are transferred only through the servo-actuators arms fixed to the wing structure. For this reason it has been decided to apply two concentrated loads in correspondence of the center of each aileron.

The servo-actuator arms are considered as rigid beams.

In the figure below the orange arrows on the right represents the hinges moment computed by the CFD analysis and they are then transposed and applied to the servo-actuator as shown.

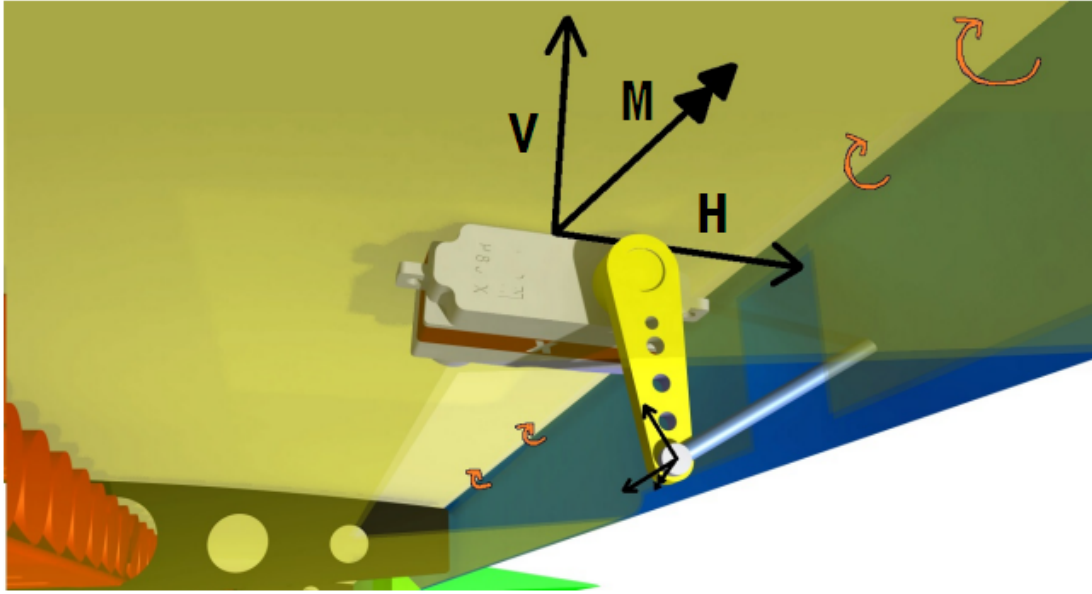


Figure 4.16: Concentrated loads detail

Regarding the battery packs, they're allocated inside the wing box and a linearly distributed load was applied to simulate their weight. The table below shows the applied weights in Maximum Take-off Weight (MTOW) condition.

	Mass per unit length	Battery pack length	Total weight
Internal pack	$2.5 [Kg/m]$	$0.8 [m]$	$2 [Kg]$
External pack	$2.5 [Kg/m]$	$0.65 [m]$	$1.625 [Kg]$

Table 4.4: Battery packs weight

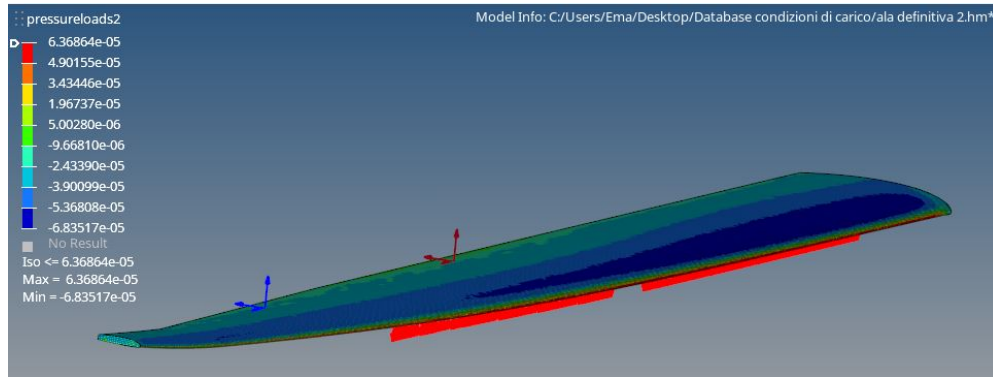


Figure 4.17: Battery pack distributed loads (in red)

Continuity between the two half wings is granted by fixed joint applied on each node that lays on the symmetry plane (x-z plane). Constraints are applied to the nodes of both the skin and the wing box.

Wing-fuselage joint is modeled by fixing the nodes where the aluminium insert described before will be located, which are the first 5 cm along the y direction as is visible in the image below.

The insert is quite thick compared to the rest of the structure, for this reason fixed nodes had been used, considering an infinite stiffness of this area.

The aluminium insert is integrated inside the wing box and is necessary to joint the wing to the fuselage as it is not possible to use screws or bolts with carbon fiber composites.

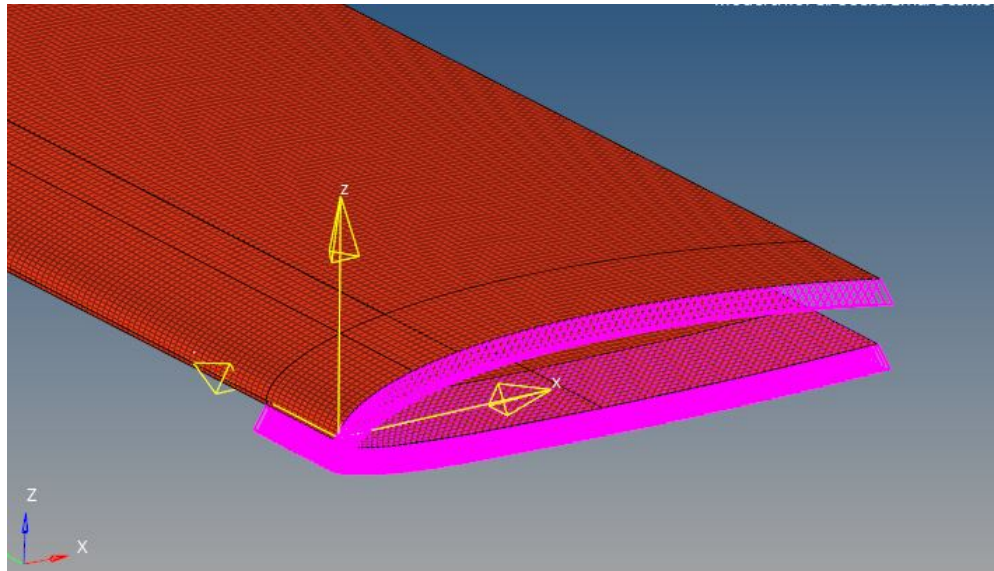


Figure 4.18: Constraints

4.2.7 Limitations and assumptions of the model

The FEM model has many simplifications and assumptions, some of them have been made to simplify the model from features that do not influence the results that much and others have been made because some phenomenons are very complex to model and require much more time than what is available for this thesis.

In future works many of the aspects addressed in this section could be developed and implemented in the model.

Accessibility openings on the wing are not considered by the model, these can slightly lower bending stiffness and torsional stiffness and most importantly fiber optic lines can not pass in these areas so no sensors can be placed there.

One of the most important limitations is the joint between the two sections of each half-wing, in fact the wing has been divided into three parts in order to easily transport and disassemble the aircraft. This requires a joint between the external and the central parts. The joint will be realized using a bayonet mechanism, a reinforcement integrated inside the central wing box is inserted in the external wing box and fixed in place.

The FEM model does not consider this joint, as its modeling requires time and also tests are needed to verify that the joint model sets out the real situation. The joint model is being developed by the team internally.

Another difference is the ribs modeling, their shape is identical to the real case but the real ribs will be realized using a sandwich structure to reduce weight, while in the FEM they are modeled as laminates.

The model is valid in static conditions and can't be used for dynamic analyses.

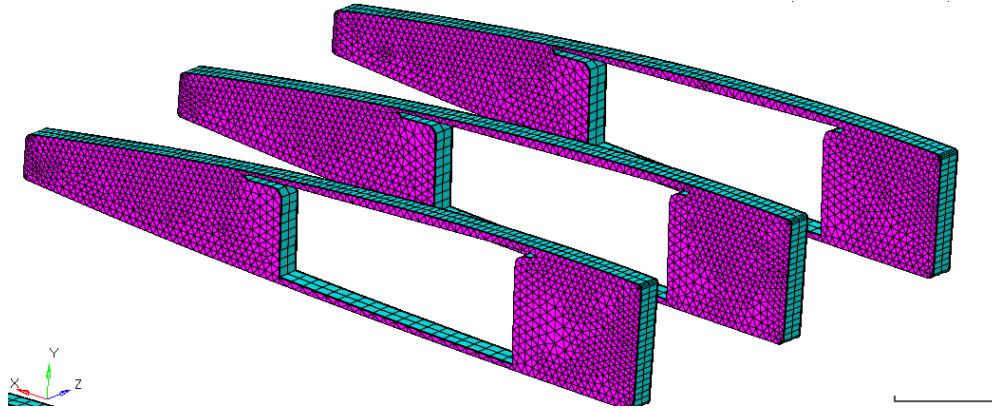


Figure 4.19: Sandwich ribs

4.2.8 Results and data post-processing

The solution is obtained through Optistruct software and data post-processing has been made with the software Hyperview.

Optistruct output a .h3d file that contains the results and is then opened through Hyperview to plot strain, stresses and displacements and export them for later use.

FBG sensors will be placed only in specific regions of the wing box, as shown in the image below, it is thus important to export the desired results only in those regions.

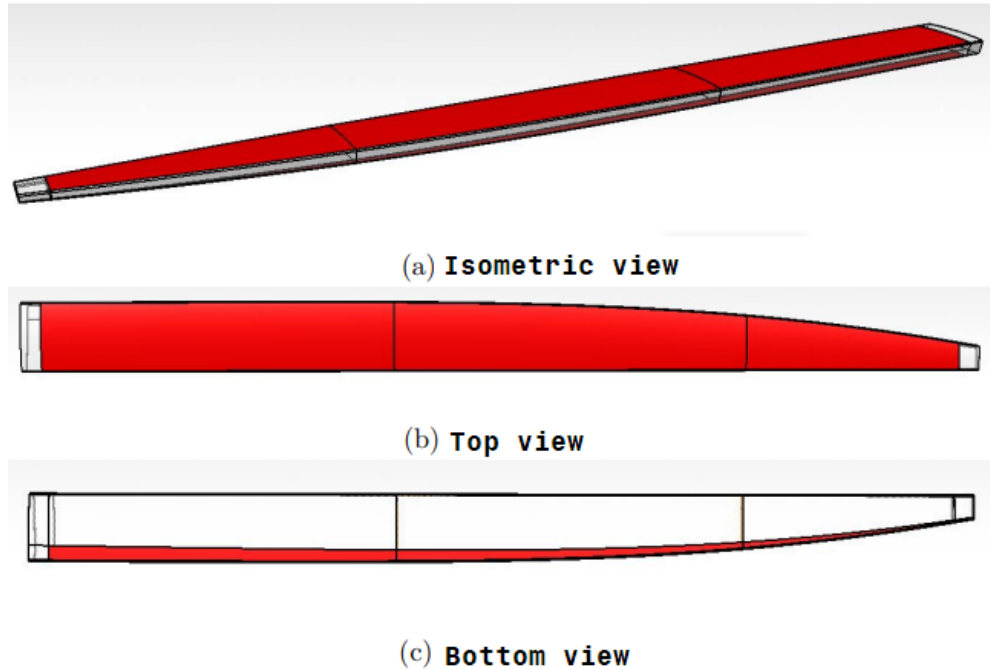


Figure 4.20: Regions where FBG sensors installation is possible

It has been chosen to export elements strain in combination with elements ID and elements centroid coordinates to have all the information needed for the sensors network.

- Element ID: integer number that identifies the element inside the mesh
- Element centroid coordinates: x,y,z coordinates of each element
- Composite strain: strain of each element using Von Mises criterion

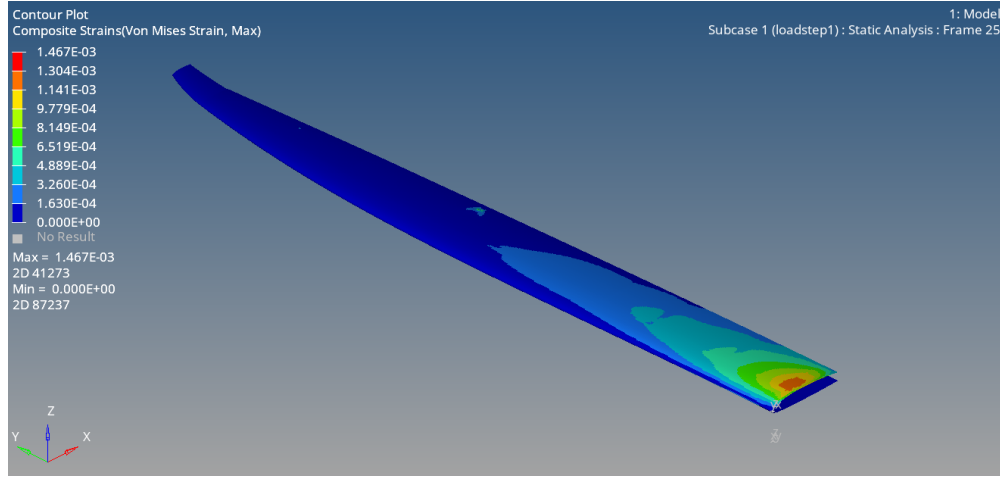


Figure 4.21: Composite strains on skin ($V = 15 \text{ m/s}$, $\alpha = 1^\circ$)

Strains distribution depends on the pressure field obtained through CFD analysis and shows higher strains near the root of the wing, as expected.

Concentrated loads due to the mobile surfaces also cause high strain on the contact points.

4.3 Flight conditions

CFD and FEM analysis are quite burdensome in terms of computing performance and thus require time and resources.

It is important to choose the right flight conditions that allows a good training of the network, minimizing the requested resources.

For this thesis 45 flight conditions have been analyzed, 12 of them had already been studied by Ing. Emanuele Frediani in his thesis.

ISA standard model has been used for all the analyses with the following values, referring to sea level conditions:

- Pressure: 1 atm
- Temperature: 15°C
- Air density: 1.225 kg/m^3

Cruise conditions for the aircraft analyzed in this thesis are shown below.

Velocity	Angle of attack	Ailerons deflection
15 [m/s]	3 [deg]	0 [deg]

Table 4.5: Cruise conditions

In order to build a useful and robust sensors network it has been chosen to pick flight conditions near the cruise design point.

The choice is also justified by the fact that this type of aircraft will rarely perform maneuvers that are far from the cruise point.

Flaps deflection will not be considered as they are only used during take-off and landing.

Ailerons (inner and outer) will be operated with the same deflection.

The table below shows the analyzed flight conditions:

Velocity	10 12.5 15 [<i>m/s</i>]
Angle of attack	1 2 3 4 5 [<i>deg</i>]
Ailerons deflection	-5 0 +5 [<i>deg</i>]

Table 4.6: Flight conditions

Ailerons deflections for the aircraft object of this analysis have been studied in Ing. Francesco Bellelli's thesis, in which the typical mission profile for RA led to defining turn requirements.

In his work, the hardest manoeuvre at 15 m/s requires a turn radius of 100 m in 0.25 s time, which is obtainable with a deflection angle of 18°.

However a deflection of -5° to +5° has been chosen because it represents the majority of the maneuvers that the aircraft will make during its mission and is a good interval for the sensors network validation.

Chapter 5

Sensors placement and configuration

In order to obtain a sensors system capable of performing real-time structural health monitoring and possible prognostic applications, it is important to define a proper configuration and placement of the sensors and optical fibers inside the wing box structure.

The configuration chosen has been extensively studied by Ing. Emanuele Frediani in his thesis (see reference in bibliography) and will be briefly illustrated in this chapter as it is a mandatory step for the introduction of the neural network in the next chapter.

FEM data will be used for the evaluation of the best possible configuration as well as important considerations about the structure behaviour, the prototype characteristics and other design constraints.

5.1 Requirements and constraints

As said before, the sensors system will be used for structural health monitoring, prognostic applications on ailerons actuators and possibly aeroelastic effects monitoring and attitude optimization.

The design of the sensors configuration follows specific requirements and constraints as described below.

Requirements:

- Sensitivity to hinge moment variation
- Sensitivity to global strain field and torsional modes
- Reliability of the system
- Preserving the integrity of the wing structure

- Temperature effect compensation
- Preventing signal attenuation

Sensitivity requirements are satisfied by placing the sensors in high strain variation points, among the analyzed flight conditions. Torsional sensitivity is obtained by placing some sensors far from the torsion axis.

Reliability of the system is obtained by using high quality sensors and a relatively low number of them, minimizing the probability of faults while maintaining an acceptable accuracy overall.

As described in Chapter 2, signal attenuation is avoided by choosing the right wavelength and by avoiding macro-bending, while temperature compensation can be achieved with the use of dedicated sensors (see Chapter 2).

Constraints:

- Openings along the wing structure
- Wing joints

As said before when describing the FEM model, only some areas of the wing box are eligible for the placement of the sensors, this is due to the presence of openings, especially on the bottom side of the wing, to access other systems. The vertical faces of the wing box are also excluded as they provide an important contribution to structural stiffness and fiber integration could result problematic.

Wing joints also impose the interruption of optical fiber lines and the use of connectors at joint intersection.

5.2 Final configuration

Final configuration is obtained through a Matlab script that divides the eligible wing box surface into 100 subdomains and compares elements strain between the different flight conditions, finding the element with the higher strain variation in each subdomain, to satisfy the sensitivity requirements.

The results are then filtered in order to find a limited number of sensors, in order to fulfill the reliability requirement.

As expected, a certain number of sensors should be placed near the ailerons actuators and near the ribs, as there is a high strain variation in those regions.

Ailerons actuators are directly connected to the wing box and exert a force on a relatively small area due to the aerodynamic hinge moment and thus are subjected to higher strains. Placing sensor near the actuators is also important for future prognostic applications.

Ribs act as a constraint for the wing box carbon fibers causing high strain variations, especially on the corners.

Wing joint area has been excluded to prevent placing sensors in this area which is not present in the FEM model and should be studied separately.

Further filtering to the eligible points has been made to consider that optical fiber lines can't have high turning angles as they would cause macro-bendings and signal attenuation.

The final configuration consists of 18 sensors for each half wing, arranged in 3 optical fibers lines, two on the top and one on the bottom of the wing box, as shown in the image below.

Coordinate system is centered on the wing root, in correspondence of the leading edge, with x axis towards the tail, y axis towards the outer wing and z axis pointing upward.

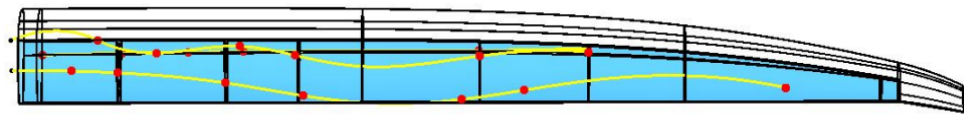


Figure 5.1: FBG sensors configuration - Top View

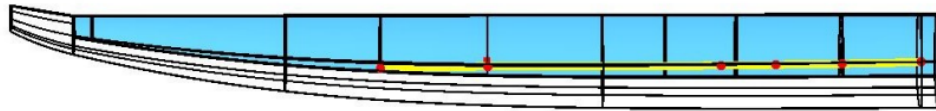


Figure 5.2: FBG sensors configuration - Bottom View

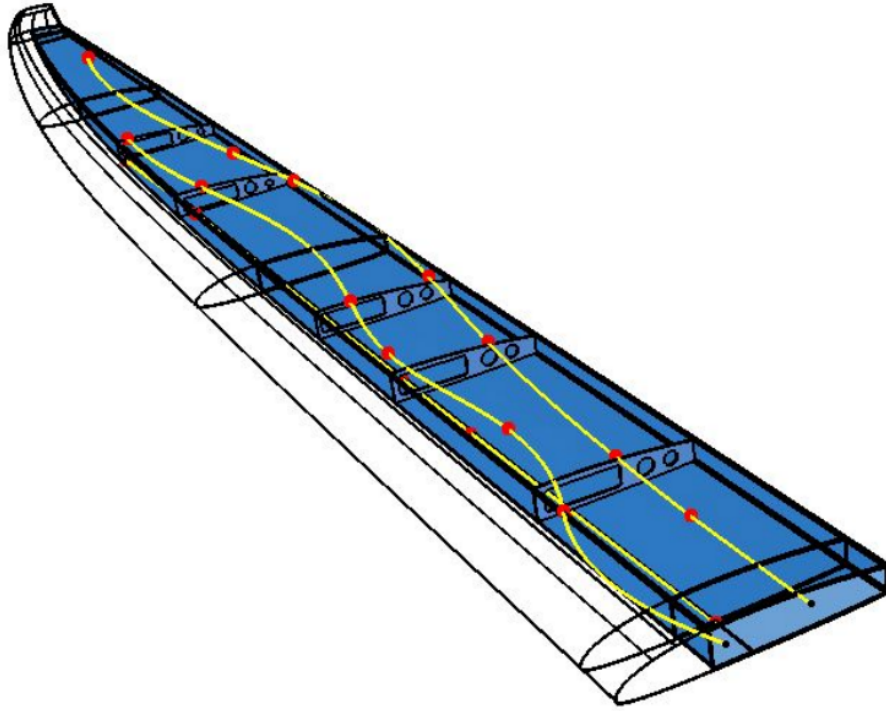


Figure 5.3: FBG sensors configuration - Isometric View

Top rear optical fiber line (Channel A)					
Sensor	Element ID	X [mm]	Y [mm]	Z [mm]	$\Delta_{Strain} [\mu\epsilon]$
1A	169770	165.7	143.4	29.11	241.8
2A	175150	170.48	266.75	28.59	314.3
3A	126600	197.37	556.74	25.96	239.6
4A	101779	231.6	765.89	21.93	219.1
5A	219579	241.05	1192.97	19.17	150.8
6A	222104	216.51	1361.18	21.13	13741
7A	106565	210	2067.62	15.85	697.5

Top front optical fiber line (Channel B)					
Sensor	Element ID	X [mm]	Y [mm]	Z [mm]	$\Delta_{Strain} [\mu\epsilon]$
1B	171021	84.17	213.89	28.41	315.7
2B	174731	118.1	371.57	29.48	319.5
3B	127402	100.01	595.04	28.64	279
4B	128597	124.15	744.78	28.52	234.9
5B	223152	125.7	1241.76	26.46	183.7
6B	224760	115.31	1533.24	24.39	142

Bottom optical fiber line (Channel C)					
Sensor	Element ID	X [mm]	Y [mm]	Z [mm]	$\Delta_{Strain} [\mu\epsilon]$
1C	157536	118.32	266.75	-6.67	255.4
2C	157032	115.68	448.43	-6.54	234
3C	141938	113.81	598.53	-6.46	203.5
4C	161011	110.99	1238.26	-6.46	107.2
5C	167499	105.32	1533.24	-6.75	101.7

Table 5.1: FBG sensors details

As described in Chapter 2, the interrogator used for this project has 4 channels. It has been decided to use 3 channels and leave the other channel for the temperature effect compensation.

Chapter 6

Artificial Neural Network

6.1 Introduction

Artificial Neural Networks (ANN) are one of the most rapidly-evolving artificial intelligence (AI) systems, driving the growth of nearly every industry. Engineering is experiencing additional opportunities as well as challenges as a result of neural networks and big data. Given the large number of complex problems encountered in engineering applications, it is critical to develop a tool that can effectively replace the large number of complicated computations required by existing approaches. Researchers, on the other hand, hope to uncover more valuable or intriguing information hidden beneath the physical phenomenon and experimental data. As a most promising data-driven technique, ANN is of significant interest, and it has been shown to be successful and reliable in a variety of academic areas and projects.

An Artificial Neural Network is a parallel computational model inspired in the performance of the human brain that consists in an arrangement of interconnected information processing units (called neurons), used for tasks such as function approximation, filtering, control, time series analysis, signal processing, and pattern association or recognition. To accomplish these assignments, ANNs have to perform a process of learning (training) in which they adapt the characteristics of their interconnections to attain a desired objective.

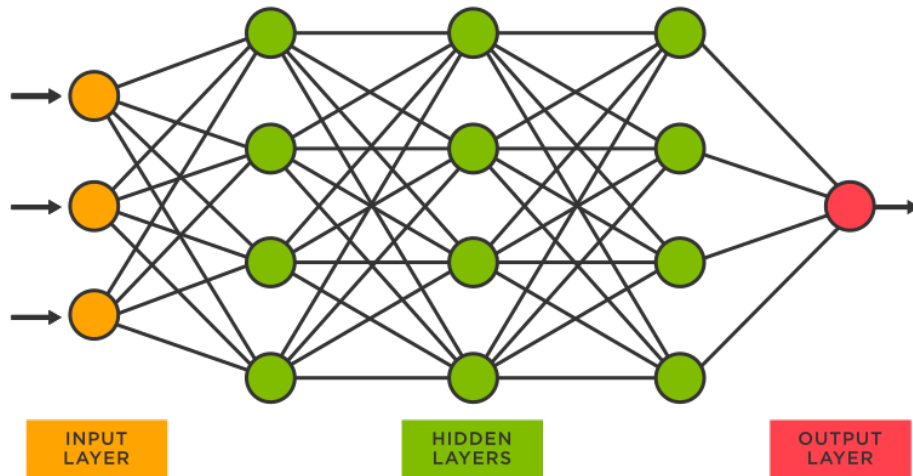


Figure 6.1: Artificial Neural Network diagram

The human brain is capable of quickly recognising complex non-linear problems. Three simple layers are responsible for the computation's speed. The first phase involves obtaining information, which can be done through sight or other senses. This data is subsequently passed on to the neurons that have been trained. The activation of a trained conglomeration of neurons facilitates the use of stored algorithms, allowing recognition of, for example, the shape of a letter or number, in stage three.

Artificial neural networks follow the same three-layer logic. The artificial neural network's initial layer is the input layer, where known data is sent into the system. The proper input variables are chosen based on the parameters to be predicted. After data is entered into the input layer, it is sent to the hidden layer, which is the algorithm-storing counterpart of the human brain. Certain nodes become active as a result of the network's learning in response to the input layer nodes. Depending on the amount of data and the quality of the prediction, the hidden layer might be made up of numerous layers of nodes.

The last layer is the output layer, which predicts the desired variable based on the input variables as well as the hidden layer containing the trained knowledge.

The internal structure of an ANN is an array of neurons. These neurons are grouped in layers depending on the type of network and its components are:

- Inputs signals s_i : They are the inputs of the problem or the outputs of neurons in a previous layer.
- Synaptic weights w_{ki} : They measure the contribution of each input to the neuron and might take positive or negative values.

- Summing junction Σ : It constitutes a linear combiner of the weighted input signals $u_k = \sum_{i=1}^n s_i \cdot w_{ki}$ where i indicates the corresponding input or synaptic weight of the k neuron.
- Bias b_k : They increase or lower the input of the activation function, resulting $v_k = u_k + b_k$
- Activation function φ : It limits the amplitude of the output of the neuron described by $y_k = \varphi(v_k)$
- Output y_k : It can be an output or input of the model for the following layer of neurons.

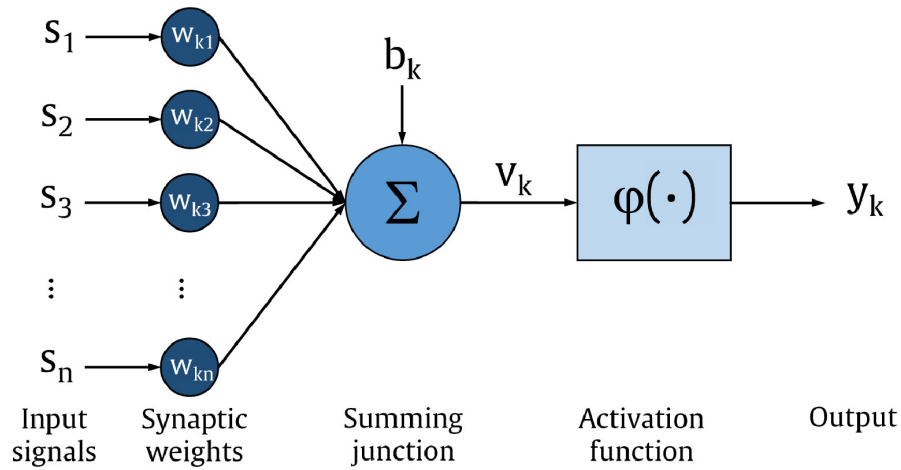


Figure 6.2: Neuron model (Perceptron)

The model in the figure above is also known as 'Perceptron model' which is the basis of the early Artificial Neural Networks.

The training of an ANN model requires:

- **Training function:** which depends on the type and characteristics of the network
- **Learning rate:** that indicates how fast the weights w_{ki} and biases b_k change in each iteration
- **Performance parameter:** that measures the precision of the ANN model and stops the training when the desired tolerance is reached.

6.2 Classification

According to network architecture, ANNs can be classified into the following three major types:

- **Multi-layer Perceptron Neural Network (MLPNN):** it consists of one input layer, at least one hidden layer and one output layer. The MLPNNs involving more than two hidden layers are usually called 'Deep Neural Networks'. It has been established that increasing the number of hidden layers improves the ability of the network (in terms of capturing the non-linear dependencies between the inputs and outputs), especially in the case when there is a large number of network inputs. The nodes in each layer are fully connected to the nodes in the adjacent layers through weights. The training approach will update the weights between the nodes until the model error is as low as possible. For efficient training, there are a variety of training methods to choose from, including the gradient descent algorithm, stochastic gradient descent algorithm, Levenberg-Marquardt algorithm, and others. The difference between the target and prediction can be assessed using the loss function during the training process. Different loss functions, such as mean-squared loss, cross-entropy loss, and dice coefficient loss, can be used to fit the target for different prediction tasks. The output of the MLPNN model can be described by a specific mathematical formula once the MLPNN model has been trained.
- **Recurrent Neural Network (RNN):** MLPNNs can explain complex non-linear relationships between inputs and outputs, but they neglect interdependencies among input variables. RNN was created with this goal in mind, and it was created to solve time series problems like weather forecasting and text processing. The neurons in the hidden layer of an RNN not only receive the current input information, but they are also influenced by previous information. Each layer of RNN unrolling has the same weight values, which are trained at different times with time-related inputs.
- **Convolutional Neural Network (CNN):** It was developed to extract features from two-dimensional (2D) data, such as images, efficiently. To build feature maps that reflect the main information in the images, CNNs use a sequence of convolution and down-sampling layers. CNNs have made significant progress in the fields of computer vision and image identification over the last decade.

The benefits that may accrue via the use of multi-layer feedforward artificial neural networks (MLPNN) include the following:

- With the availability of data, feedforward networks could be used for modeling a broad spectrum of systems.
- Feedforward neural networks could be useful for applications where analytical methods are yet to be discovered.
- Using a suitable neural network size, it is possible to capture the non-linear dependencies between the input and outputs to a reasonable degree of accuracy.

6.3 Learning

There are three major learning paradigms: supervised learning, unsupervised learning and reinforcement learning.

Usually they can be employed by any given type of artificial neural network architecture. Each learning paradigm has many training algorithms.

- **Supervised learning:** it is a methodology in data science that creates a model to predict an outcome based on labeled data. Labeled data contains a collection of variables (inputs) and a specific output that we are trying to predict. The task is to produce a correct output given a particular input. This paradigm is effective when dealing with problem related to classification, regression and function fitting.

The assumption behind supervised learning is that future data would behave similarly to previous data. The algorithms "learn" from a given dataset, fitting a model based on previous behaviours and labels. When these models are exposed to new data, they may not perform as well. When this happens, the model is said to be "overfit," or overly tuned to the past data.

- **Unsupervised learning:** it is a technique that determines patterns and associations in unlabeled data. This technique is often used to create groups and clusters. In unsupervised learning we seek to determine how the data is organized, as no information label is present.

Unsupervised learning is frequently used for exploratory analysis and anomaly identification since it allows users to observe how data segments relate to one another and what trends may exist. They can be used to preprocess data before using a supervised learning algorithm or other AI approaches.

- **Reinforcement learning:** it is a technique that provides training feedback using a reward mechanism. The learning process occurs as a machine, or Agent, that interacts with an environment and tries a variety of methods to reach an outcome. When the Agent reaches a desirable or undesirable state, it is rewarded or penalised. The Agent learns which conditions produce positive results and which should be avoided. So that the Agent can iteratively learn to acquire a better score, success is quantified using a score (denoted as Q , so reinforcement learning is sometimes referred to as Q -learning).

No dataset is usually given, it is generated by interactions with the environment. Some applications of reinforcement learning are in the field of robotics and autonomous driving.

In order to better understand how the learning process work it is important to introduce some key concepts:

- **Cost function:** A cost function (also called 'Loss function') is a measure of the error in prediction committed by an algorithm. It indicates the difference between the predicted and the actual values for a given dataset. Closer the predicted value to the actual value, the smaller the difference and lower the value of the cost function. Lower the value of the cost function, the better the predictive capability of the model.
- **Learning rate:** it is used to control the rate at which an algorithm updates the parameter estimates or learns the values of the parameters. Higher values of learning rate means a faster learning process, however using high values can cause low accuracy as the cost function could saturate at a value higher than the minimum value.
- **Backpropagation:** this learning algorithm is popular with supervised learning paradigms and has proven to be useful for solving a spectrum of practical problems. The algorithm involves the forward propagation of input signals, computing the cost function and backpropagating the error until the cost function is minimized for a number of training iterations or epochs. The algorithm minimizes the sum of squared approximation errors using a gradient descent technique.

6.4 Training and Validation

Training is the process of computing the appropriate weights of the neural connections inside the ANN, by receiving feedback from the backpropagation described in the previous section.

Some of the most used training algorithms are:

- **Gradient descent (GD):** it is a first order method and is the most straightforward training algorithm. The parameters are updated at each epoch in the direction of the negative gradient of the cost function. The learning rate can be constant or updated after each iteration.
- **Newton's method (NM):** it is a second-order algorithm because it uses the Hessian matrix. This method's objective is to find better training directions by using the second derivatives of the cost function. Newton's method requires fewer steps than gradient descent to find the minimum value of the cost function, however, Newton's method has the difficulty that the exact evaluation of the Hessian and its inverse are pretty expensive in computational terms.
- **Conjugate gradient (CG):** in this training algorithm, the search is performed along with conjugate directions. They generally produce faster convergence than gradient descent directions. These training directions are conjugated concerning the Hessian matrix. It can be regarded as something intermediate between gradient descent and Newton's method. It is motivated to accelerate the typically slow convergence associated with gradient descent. This method also avoids the information requirements associated with the Hessian matrix's storage, evaluation, and inversion, as Newton's method requires.
- **Levenberg-Marquardt (LMA):** it is designed to work specifically with cost functions which take the form of a sum of squared errors. It works without computing the exact Hessian matrix but has the speed of a second-order method. Instead, it works with the gradient vector and the Jacobian matrix.

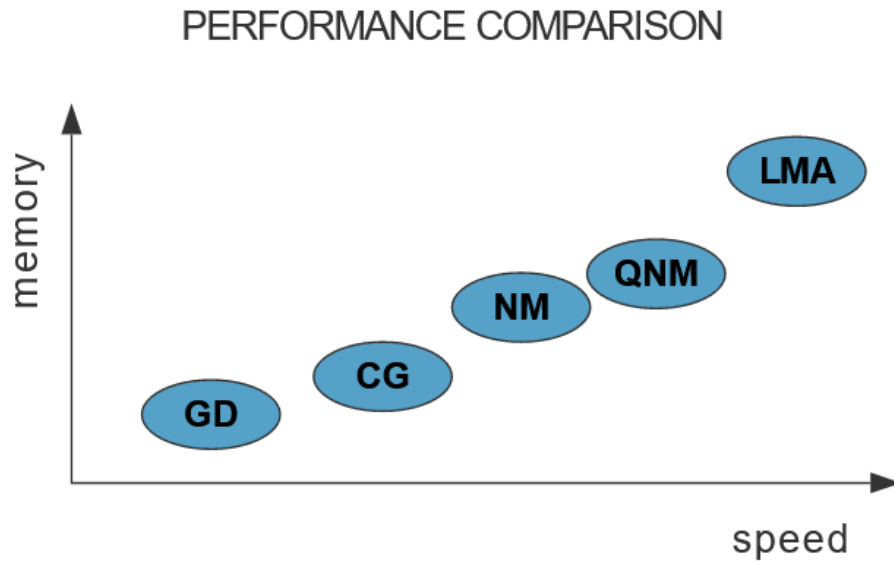


Figure 6.3: Training algorithms comparison

Validation

In order to test the neural network a small sample of data is extracted from the training dataset and used for the validation to assess the accuracy of the network and avoid overfitting.

Overfitting describes the situation in which the model is over-optimized to accurately predict the training set, at the expense of generalizing to unknown data. This can happen because the model greatly twists itself to perfectly conform to the training set, even capturing its underlying noise.

The simplest solution is to split the dataset into a training set and a test set. The training set is used for the training procedure as described above, but the accuracy of our model is evaluated by forwarding the test set to the trained model and measuring its accuracy. During training, it is possible to monitor the accuracy of the model on the training set and test set.

Regularization

Regularization is a set of strategies used in Machine Learning to reduce the generalization error, aiming at reducing overfitting and keep, at the same time, the training error as low as possible.

One way overfitting occurs is when the magnitude of the weights grows too large, one way to regularize is to modify our objective function by adding an additional term which penalizes large weights (L1-L2 regularization).

Other effective regularization methods are Dropout and Bayesian regularization. Dropout consists of randomly deactivating some neurons in a certain layer, in order to reduce the network's tendency to come to over-depend on some neurons. Bayesian regularization can be used to convert a nonlinear regression into a "well-posed" statistical problem in the manner of a ridge regression.

6.5 Neural network implementation

Neural Network implementation has been done in MATLAB using the Deep Learning Toolbox.

It has been decided to use a feedforward neural network, as it is adequate for the complexity of the problem; using a more advanced network does not bring any substantial benefit for this case study.

Network Topology

The neural network topology is defined using the function `feedforwardnet([n])`, where `n` is a vector defining the number of neurons for each hidden layer.

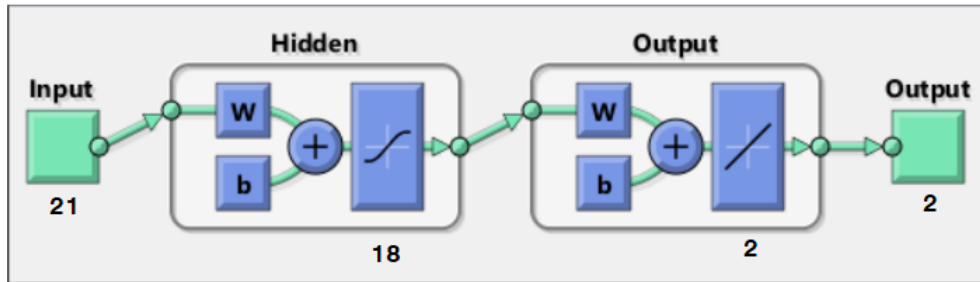


Figure 6.4: Feedforward Neural Network scheme

- **Input:** 18 FBG sensors as described in previous chapter and 3 flight conditions (velocity, angle of attack and ailerons deflection), for a total of 21 inputs.
- **Output:** Ailerons hinge moment, for both outer and inner ailerons.

Different configurations have been tried, using a single hidden layer or two hidden layers and changing the number of neurons for each layer. Results will be shown later in this chapter.

Training Function

Using the parameter `net.trainFcn` it is possible to define the training function, choosing between the ones available in the Deep Learning Toolbox.

trainlm: This function uses the Levenberg-Marquardt algorithm. It works without computing the exact Hessian matrix but has the speed of a second-order method. Instead, it works with the gradient vector and the Jacobian matrix, expressed as follows:

$$H = J^T J \qquad g = J^T e \qquad (6.1)$$

where e is a vector containing network errors and J is the Jacobian matrix that contains the derivatives of the network errors calculated in respect to the weights and biases.

Weights and biases are then updated after each iteration using a Newton-like method:

$$x_{k+1} = x_k - [H + \mu I]^{-1} g \qquad (6.2)$$

The difference between the Newton method is the parameter μ , when it is equal to zero the method is equal to the classic Newton method, whereas when μ is large the method is similar to a gradient descent.

The goal is to lower μ after each successful iteration, as Newton method is generally faster, while having the possibility to increase it when error increase.

traibr: this training function uses Bayesian regularization. It is based on Levenberg-Marquardt algorithm but aims at reducing the generalization error, reducing overfitting and keeping, at the same time, the training error as low as possible. Bayesian regularization minimizes a linear combination of squared errors and weights.

Bayesian regularization can help with reducing overfitting especially in networks with a low number of samples.

Activation Function

The activation function limits the amplitude of the output of the neuron and is defined using the parameter `transferFcn`. It is possible to define a different activation function for each layer.

The most used activation functions are 'saturated symmetrical linear' (**satlins**) and 'hyperbolic tangent sigmoid' (**tansig**), shown in the image below.

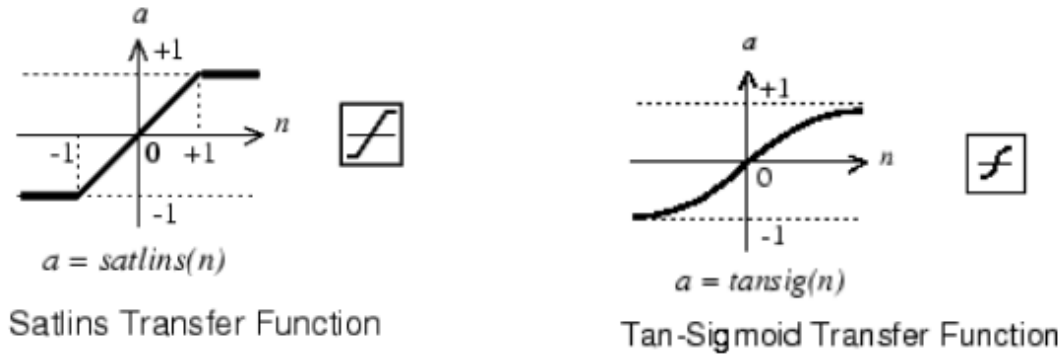


Figure 6.5: Common activation functions

Performance Function

It is defined using `performFcn`. Common performance functions are 'mean squared error' (`mse`) and 'mean absolute error' (`mae`).

For this network mean squared error has been used. It calculates the average squared difference between the estimated values and the actual value and has the advantage of not working with absolute values that can be problematic when working with data with different magnitude.

Network Parameters

- **Training Goal:** it defines the overall accuracy of the network, it is set using `trainParam.Goal`. For this network it has been set to $1e^{-8}$ as this gives very good accuracy without impacting performance as the dataset is not large.
- **Epochs:** One epoch is when an entire dataset is passed forward and backward through the neural network once. It is defined with `trainParam.Epochs` and for this network it has been set to $1e^2$ as it is not necessary to use a greater number to reach the desired accuracy.
- **Maximum validation checks:** it is the maximum number of validation checks before training is stopped. It is basically a stopping criteria. It is set by `trainParam.max_fail`.

MATLAB code

```
1 %% Neural network creation
2 net = feedforwardnet([13]); format short;
3
4 %% Training
5 net.trainFcn = 'trainlm';
6 net.trainParam.showCommandLine = true;
7 net.trainParam.epochs = 1e2;
8 net.trainParam.goal=1e-9;
9 net.trainParam.max_fail = 10; % Validation checks
10 net.trainParam.lr = 0.001; % Learning rate
11
12 net.layers{:}.transferFcn = 'satlins';
13
14 net.performFcn='mse';
15
16 net = train(net, input_rete', H', 'useParallel', 'yes', '
    showResources', 'yes');
17
18 %% Verification
19
20 verificationOut = net(valid_net')
21
22 abserr = (valid_H - verificationOut)
23 format bank
24 relerr = 100*(valid_H - verificationOut)./valid_H
25 format short
26 perf = perform(net, valid_H, verificationOut)
27 moderr = abserr.*relerr;
```

6.6 Results

Different network configurations have been tested in order to find the optimal setup.

Most of the configurations tested are feedforward networks with one hidden layer. A two hidden layer network (Deep) has also been tested and results will be described later in this section, although from first analyses it emerged that is not necessary for the examined problem to have more than one hidden layer.

As said before, the network has 21 inputs and 2 outputs, so the first parameter that needs to be chosen and optimized is the number of neurons in the hidden layer.

For this type of network, usually, the number of neurons is between the number of inputs and the number of outputs. A few tests have been run in order to find the best value.

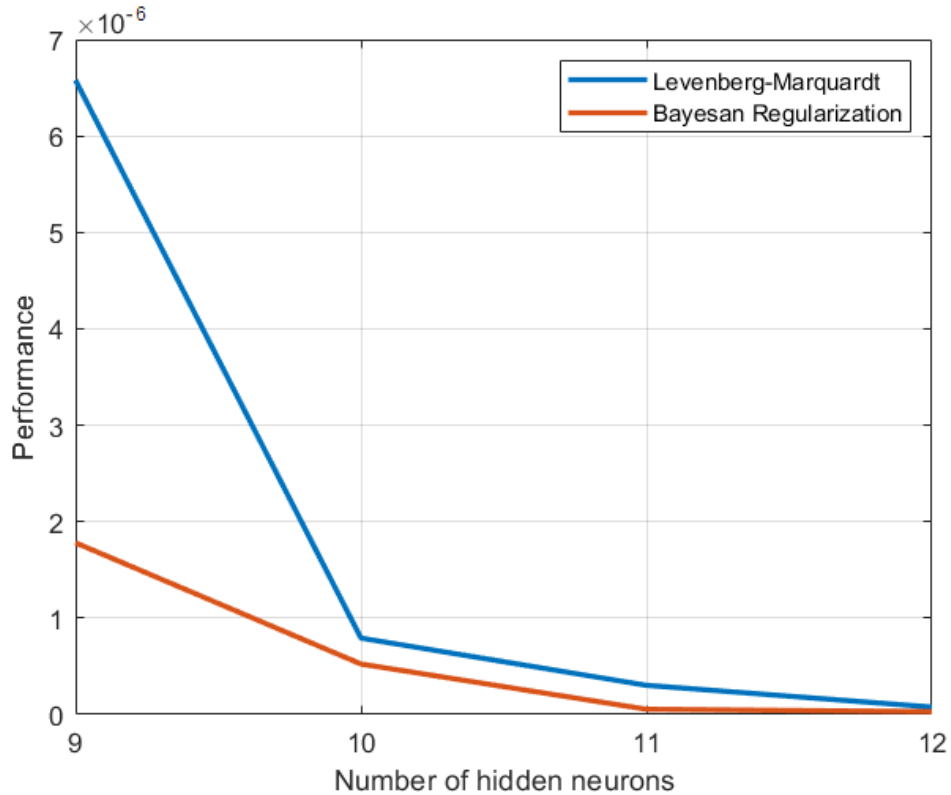


Figure 6.6: Network Performance vs Number of neurons

The image above shows network performance for a number of neurons in the hidden layer between 9 and 12 for two different training algorithms.

With a lower number of neurons we can see that the Bayesian regularization algorithm performs better than the Levenberg-Marquardt.

Increasing the number of neurons reduce this performance difference and for a value of 12-13 the best performance index is reached as shown in the image below. Further increasing the number of nodes does not offer any significant advantage in terms of network performance and errors.

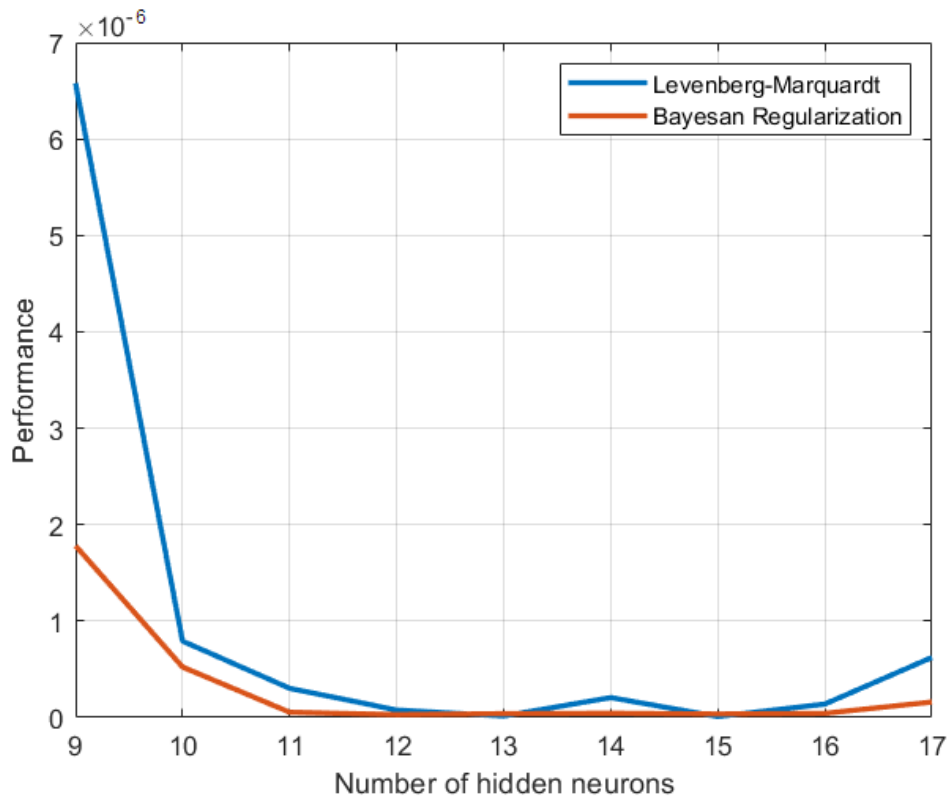


Figure 6.7: Network Performance vs Number of neurons

The next two images show the training performance for a single hidden layer network with 13 neurons and for a two hidden layers network with 13 neurons in the first layer and 5 on the second one. Progressively reducing the number of neurons is important to gradually reach the output layer configuration that has fewer nodes.

There is a slight improvement both in performance and training times using two layers but considering also the high accuracy already obtained with the single layer configuration it has been decided to use the latter one.

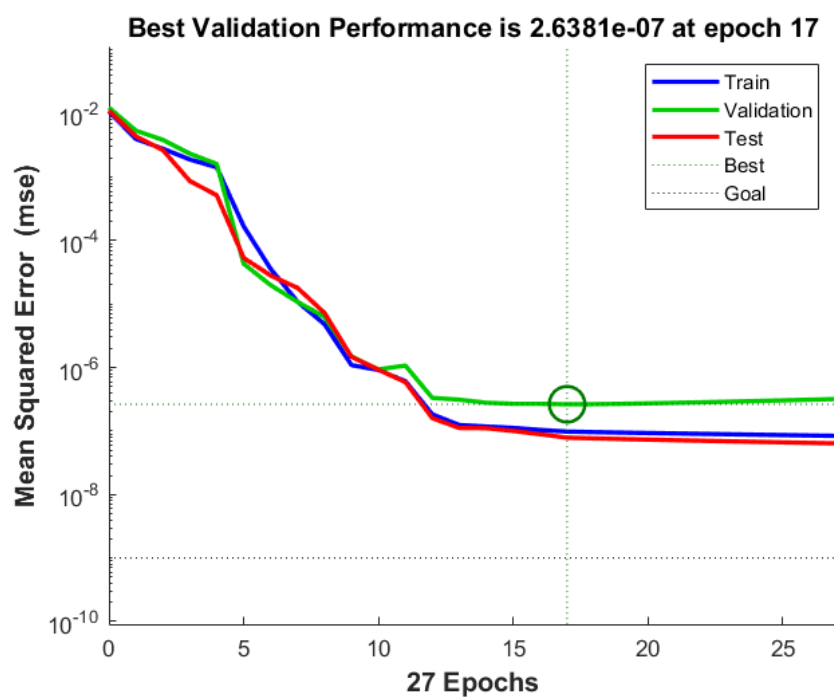


Figure 6.8: Training Performance - One hidden layer

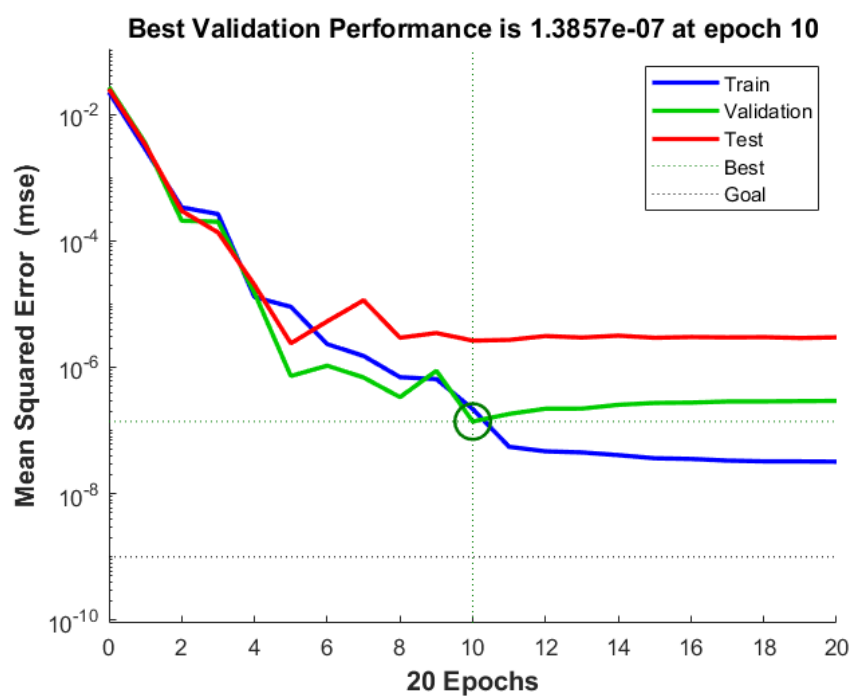


Figure 6.9: Training Performance - Two hidden layers

Chapter 7

Conclusions

In this work, an Artificial Neural Network for structural health monitoring purposes has been applied to a sensors network integrated in the wing structure. The obtained results shows the feasibility and effectiveness of this concept.

In order to simulate the real strains that would be measured by the sensors a Computation Fluid Dynamics (CFD) model has been first used to compute the aerodynamic loads for different flight conditions.

These loads are then applied to a Finite Element model (FEM) that computes the elastic response of the structure and from which strains are derived.

For this reason the accuracy and complexity of the models has to be reviewed with real data to validate the results.

As of today, project RA prototype is still in development and many tests have to be carried out to validate the aerodynamic model (in-flight tests or wind tunnel) and the structural model (materials characterization and static tests).

Sensors network is then defined using the configuration described in Chapter 5. CFD and FEM data is post-processed and organized to be inserted in the neural network.

A feedforward Neural Network has been trained using the strains simulated in 18 Fiber Bragg Grating (FBG) sensors distributed inside the wing structure and in various flight conditions near the design point as well as ailerons hinge moment derived from CFD simulations.

Many different configurations and algorithms have been tested as better described in Chapter 6.

The trained network is capable of computing ailerons hinge moment starting from wing box strains, with a relatively high accuracy.

These data allows real-time structural health monitoring, as well as enhanced diagnostic.

Future prognostic applications on other systems are also possible and interesting, such as prognostic on Flight Control System actuators, positioned near the ailerons and attached to the wing box structure.

The next step would be the validation of CFD and FEM models with real data and their correction and update following the development of the prototype.

CFD model lack of automatic hinge axis correction when changing angle of attack. FEM model also needs to be reviewed to add materials real data and to model the joint in each half-wing as well as adding accessibility openings.

An other interesting study would be to analyze the behaviour of the network to different atmosphere conditions, introducing altitude, pressure and temperature variation to the parameters.

Regarding the FBG sensors, temperature compensation effect should also be implemented as described in Chapter 2.

Bibliography

- [1] Emanuele Frediani (2020), Studio di una configurazione di sensori ottici per monitoraggio strutturale e applicazioni prognostiche
- [2] Magdalena Mieloszyk et al 2011 J. Phys.: Conf. Ser. 305 012066, An application of neural network for Structural Health Monitoring of an adaptive wing with an array of FBG sensors
- [3] Oleg Budadin · Vladimir Vavilov · Mikhail Fedotov · Sergei Vasiliev · Pavel Gnusin · Sofia Kozelskaya1 · Marina Kuimova (2021), Theoretical and Experimental Studies of Structural Health Monitoring of Carbon Composites with Integrated Optical Fiber Sensors Based on Fiber Bragg Gratings
- [4] Takanori Okoshi (1982), University of Tokyo, Optical Fibers
- [5] Bahaa E. A. Saleh, Malvin Carl Teich (1991), Fundamentals of Photonics
- [6] Mark Curran, Brian Shirk, Basics of Fiber Optics
- [7] Fiber Optics - The Principles Behind Fiber Optics - Light, Angle, Material, and Interface - JRank Articles
- [8] Cosimo Corsi (2019) - Strumentazione e Flight testing con sensori ottici FBG
- [9] Jasjot K. Sahota, Neena Gupta, Divya Dhawan (2020) - Fiber Bragg grating sensors for monitoring of physical parameters: a comprehensive review
- [10] Riccardo Raballo (2020) - Design and Development of Optical Sensors Based on FBGs Technology
- [11] Wei Jin, Y. Zhou (2006) - Structural Strain and Temperature Measurements Using Fiber Bragg Grating Sensors
- [12] Giuseppe Spampinato (2019) - Innovative optical sensors for prognostic techniques in the aerospace field: design and development of a test bench for studying the frequency response of FBG sensors
- [13] SmartScan SBI Interrogator Product Manual (2020)

- [14] SimScale documentation - Computational Fluid Dynamics (2021)
- [15] SimScale documentation - Finite Element Method (2020)
- [16] Altair University - 2D Meshing (2014)
- [17] Gaetano Quattrocchi (2019), Development of innovative prognostic methods for EMAs
- [18] Xu Yang, Jinchao Guan, Ling Ding, Zhanping You, Vincent C.S.Lee, Mohd Rosli, Mohd Hasan, Xiaoyun Cheng (2021), Research and applications of artificial neural network in pavement engineering: A state-of-the-art review
- [19] S. Haykin (2008), Neural networks: a comprehensive foundation
- [20] S.E. Shanmuganathan, S.E. Samarasinghe, Springerlink & Springerlink Ebooks Engineering, C. 2016. Artificial Neural Network Modelling
- [21] Vivian W.Y.Tam, Anthony Butera, Khoa N.Le, Luis C.F. Da Silva, Ana C.J.Evangelista (2022), A prediction model for compressive strength of CO2 concrete using regression analysis and artificial neural networks
- [22] Nadia D.Roman, Facundo Bre, Victor D.Fachinotti, Roberto Lamberts (2020), Application and characterization of metamodels based on artificial neural networks for building performance simulation: A systematic review
- [23] Richard Fiifi Turkson, Fuwu Yan, Mohamed Kamal Ahmed Ali, Jie Hu (2016), Artificial neural network applications in the calibration of spark-ignition engines: An overview
- [24] Magdi Zakaria, Mabrouka AL-Shebany, Shahenda Sarhan (2014), Artificial Neural Network : A Brief Overview