POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Real-Time Algorithm for Deformed Shape Reconstruction of a Beam Applied to Helicopter Blades

Laurea Magistrale in Aeronautical Engineering - Ingegneria Aeronautica

Author: Mateusz Krzysztof Garbacz

Advisor: Prof. Pierangelo Masarati

Academic year: 2021-2022

1. Introduction

The main objective of the presented executive summary is to give a critical overview of the thesis covering the topic of the development of the real-time algorithm for deformed shape reconstruction of a beam with particular attention paid to the application in helicopter blades. The inspiration for the thesis was a helicopter blade model that has been developed in the laboratory of Dipartamento di Scienze e Tecnologie Aerospaziali at Politecnico di Milano. The thesis also covers the numerical modal shape functions determination using a finite element model of a helicopter blade model that has been developed in the laboratory.

A shape reconstruction problem in recent years gained more attention from many industries. It may be particularly interesting for aerospace, space, automotive, and wind turbines sectors. Information about the state of an object may be useful for the monitoring of its structural integrity and applied to the Structural Health Monitoring (SHM) systems and Health and Usage Monitoring Systems (HUMS). Both can be used to increase the safety of a machine by early malfunction detection of a structural component. Those systems may be also used to apply predictive maintenance which drives down maintenance costs for machine operators, which can increase the attractiveness of the product for the potential client [5]. In the space and aerospace industry, the knowledge about the state of an object may be used for control optimization.

In helicopter rotors, which are the focus of this thesis, the knowledge about the structural condition of the blade can be used for both; SHM/HUMS systems as well as optimization of control. In helicopters, the flight control system can be improved by the application of the Rotor State Feedback (RSF) system, which improves turbulence alleviation and handling of the machine.

A shape of an object can be reconstructed using strain measurements but there are also other methods such as photogrammetry or direct fibre optic shape sensing (DFOSS). In the thesis, only reconstruction using strain sensors is considered, due to the fact that it is the cheapest and the most available method. The modelled measurement system is based on the one used in the blade from the laboratory, there are used two types of sensors: strain gauges and optical fibres. A strain gauge rosette is used to measure an axial strain and a shear strain, while optical fibre is used to measure only axial strain [2].

2. Methodology

The global reference system of the beam is shown in fig. 1, while fig. 2 presents the reference system of the cross-section. The $Y-Z$ reference system of the cross-section is the global one, while the $\eta-\zeta$ is the local reference system that is rotated by the built-in twist angle. Both reference systems of the cross-section have origin in the position of the elastic axis of the beam.

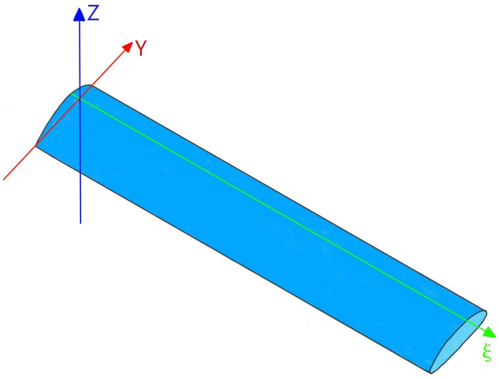


Figure 1: Global reference system

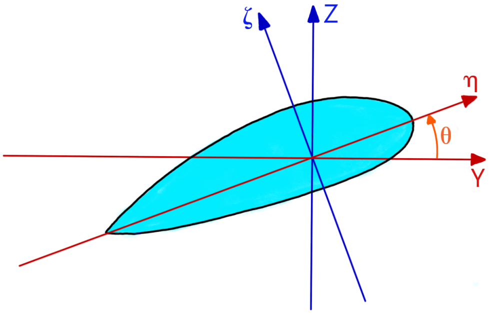


Figure 2: Cross-section's reference system

A modal shape sensing approach is proposed for the real-time shape reconstruction algorithm. It is capable to reconstruct the shape of a beam-like structure subjected to general deformations including torsion, in-plane, and out-of-plane bending and handles nonlinear terms up to 2^{nd} order. The relation between strain and displacement of the beam comes from the nonlinear model of motion for the elastic bending and torsion of twisted nonuniform rotor blades [3]. The formula of eq. (1a) presents the relation between the axial strain and displacement of the beam, while eq. (1b) and eq. (1c) show the relation between shear strain and derivative of twist angle.

$$\begin{aligned}\varepsilon_{\xi\xi} &= u' + \frac{1}{2}(v'^2 + w'^2) \\ &+(\eta^2+\zeta^2)\left(\theta'\phi' + \frac{\phi'^2}{2}\right) \\ &-v''[\eta\cos(\theta+\phi) - \zeta\sin(\theta+\phi)] \\ &-w''[\eta\sin(\theta+\phi) + \zeta\cos(\theta+\phi)] \qquad (1a)\end{aligned}$$

$$\varepsilon_{\xi\eta} = -\frac{\zeta\phi'}{2} \qquad (1b)$$

$$\varepsilon_{\xi\zeta} = \frac{\eta\phi'}{2} \qquad (1c)$$

In the proposed algorithm torsion and bending problems are decoupled; the order of solution is strictly defined, and the torsional problem must be solved as the first one.
The complete, overdetermined problem must be solved in a least-squares sense [4]. The final formulation of the torsional and bending problem is shown respectively in eq. (2) and eq. (3).

$$\boldsymbol{q} = \boldsymbol{A}_t^+ \boldsymbol{\varepsilon}_{\xi t_k} \qquad (2)$$

$$\boldsymbol{r} = \boldsymbol{A}_b^+ (\boldsymbol{\varepsilon}_{\xi\xi} - \boldsymbol{T}(\boldsymbol{q})) \qquad (3)$$

where $\boldsymbol{A}_t^+$ and $\boldsymbol{A}_b^+$ are pseudo inverted matrices related to the sensors' position and shape functions, while $\boldsymbol{q}$ and $\boldsymbol{r}$ are generalised coordinates related respectively to the torsion and bending problem. The final chordwise, flapwise and twist is computed using respectively eq. (4), eq. (5), and eq. (6), the shape functions are multiplied by generalised coordinates.

$$v(\xi,t) = \boldsymbol{n}_v(\xi)\boldsymbol{r}(t) \qquad (4)$$
$$w(\xi,t) = \boldsymbol{n}_w(\xi)\boldsymbol{r}(t) \qquad (5)$$
$$\phi(\xi,t) = \boldsymbol{n}_\phi(\xi)\boldsymbol{q}(t) \qquad (6)$$

3. Shape functions

In the thesis there were obtained two types of shape functions, analytical shape functions were obtained by pencil&paper method, while numerical shape functions were obtained by post-processing of the modal analysis results in a Matlab script.

3.1. Analytical shape functions

The analytical shape functions have been used in the Matlab script and the Simulink model. The latter has been used for co-simulation with the MBDyn models. The bending and torsion shape functions are presented in fig. 3 and fig. 4. The bending functions are the same for flapwise and chordwise deformation. Those shape functions are valid under the assumption that the beam has a uniform structure.

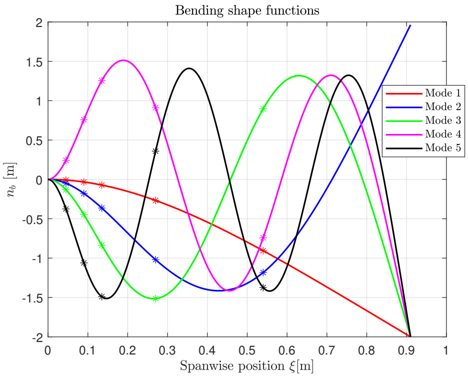


Figure 3: Analytical bending shape functions

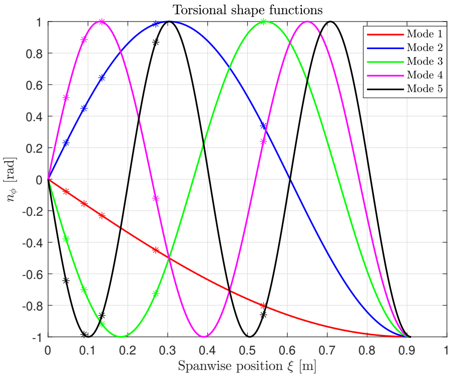


Figure 4: Analytical torsion shape functions

3.2. Numerical shape functions

The numerical shape functions have been obtained using the least-squares algorithm applied to the data obtained from the finite element method modal analysis. The model of the blade used in the analysis corresponds to the blade from the laboratory including its highly non-uniform structure. The bending shape functions in flapwise and chordwise directions are different. The shape functions are shown in fig. 5, fig. 6, and fig. 7.

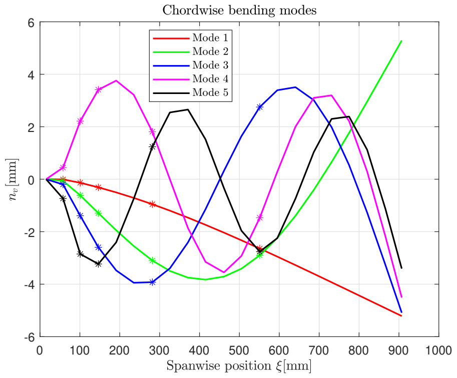


Figure 5: Numerical chordwise bending shape functions

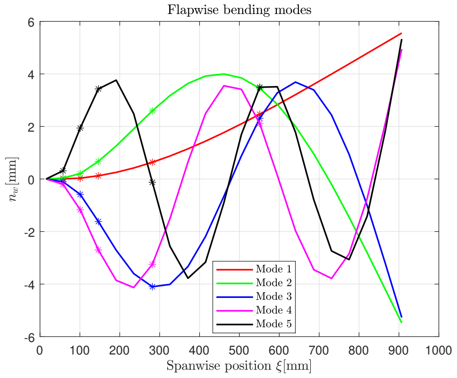


Figure 6: Numerical flapwise bending shape functions

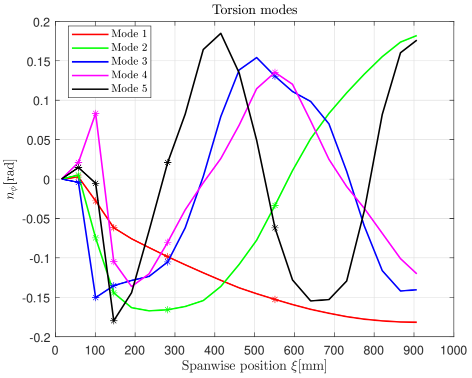


Figure 7: Numerical torsion shape functions

4. Validation using the beam with circular cross-section

The main validation of the algorithm has been performed using a beam with a circular cross-section. The beam has similar dimensions as the blade from the laboratory. Along the spanwise axis of the beam there are located 5 measuring stations, with 3 sensors on each station; 2 strain gauges and 1 optical fibre sensor.

The beam has been validated using small and large deformation. For each deformation 2 load cases have been used. In the summary, only plots for large deformation and complex load cases are shown. In this load case, two transverse forces are applied together with a torque. The results are shown in fig. 8, fig. 9, and fig. 10. The algorithm has been validated by comparison of the results obtained by the shape reconstruction with the analytical results and the deformation of the nodes from the MBDyn simulation. In the algorithm, there were used strains from the analytical formulas, as well as strains from the MBDyn simulation

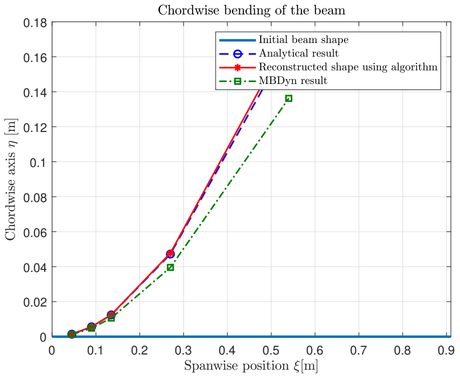


Figure 8: Chordwise deformation

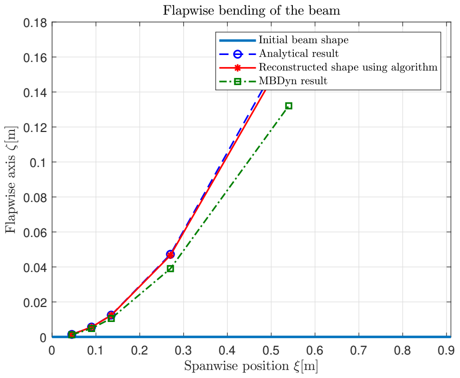


Figure 9: Flapwise deformation

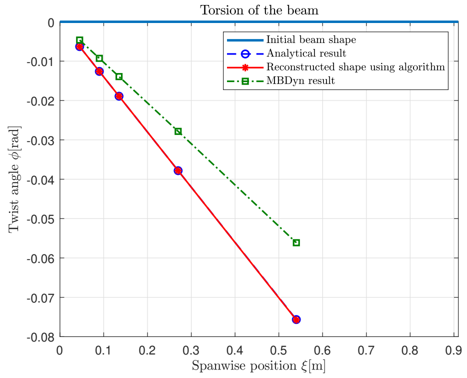


Figure 10: Twist

The algorithm has been also analysed from the linearity point of view, by applying different loads and increasing the order of load it was checked at which point the solution from the algorithm becomes nonlinear. It was also checked how the quality of the results are influenced by: the number of modes used in the shape reconstruction, the number of measuring stations along the blade, and the number of sensors on the stations.

5. Validation using the Princeton beam

The algorithm has been also verified reproducing the Princeton beam experiment from the paper that was written by Bauchal et al. [1].

In the experiment, the beam with the rectangular cross-section has been loaded with 3 different loads under different loading angles from the range $< 0^o, 90^o >$.

The results of the flapwise, chordwise and torsion reconstruction obtained through the algorithm correspond to the results of the MBDyn model of the beam that was already validated with experimental results. This proves that the algorithm correctly reproduces beam shape even if more complex load cases are applied. The results of the shape reconstruction are shown in fig. 11, fig. 12, and fig. 13.

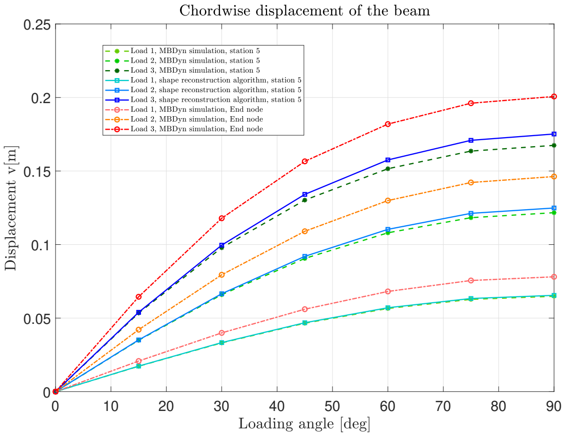


Figure 11: Chordwise deformation of the Princeton beam

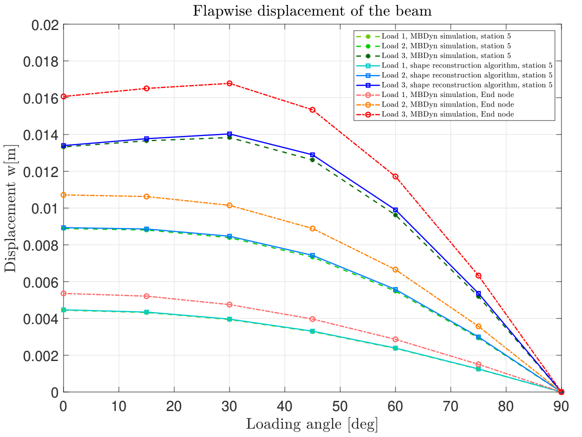


Figure 12: Flapwise deformation of the Princeton beam

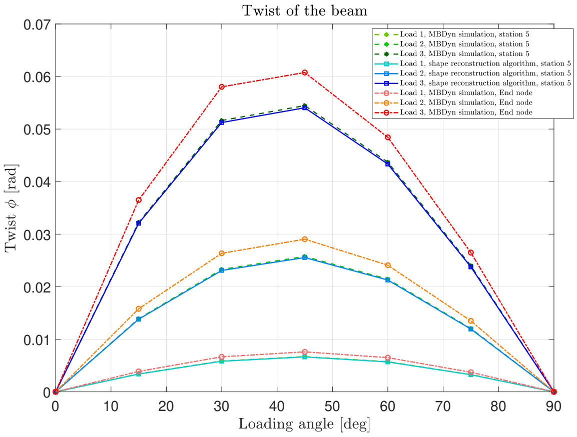


Figure 13: Twist angle of the Princeton beam

6. Application of the algorithm to the helicopter rotor model

The algorithm has been applied to the multibody helicopter model, prepared in the MBDyn software. To apply the algorithm, it was necessary to simplify the model of the rotor, as well as apply some assumptions. The main assumptions are presented below.

- The analytical shape functions are used, so the blade is assumed to be uniform, and it is assumed that the flapwise and chordwise bending shape functions are the same.
- The cross-section of the blade is assumed to be rectangular, the 4 sensors are in the middle of the sides of the cross-section.

Having those assumptions, the results are not expected to be exact but at least realistic. The results of the shape reconstruction in the flapwise and chordwise bending are presented in fig. 14 and fig. 15, while the twist is shown in fig. 16.

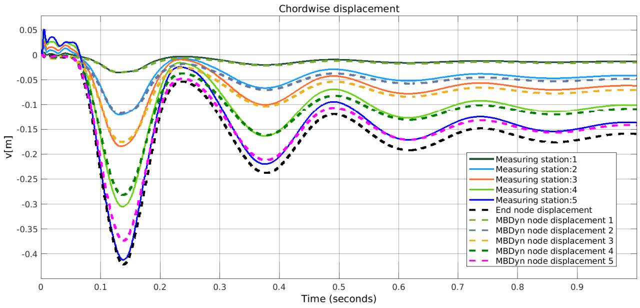


Figure 14: The real-time chordwise bending displacement of the blade

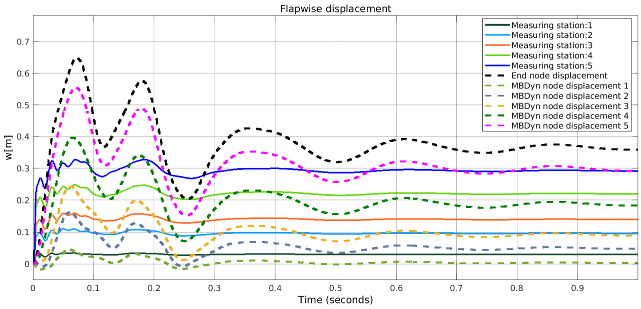


Figure 15: The real-time flapwise bending displacement of the blade

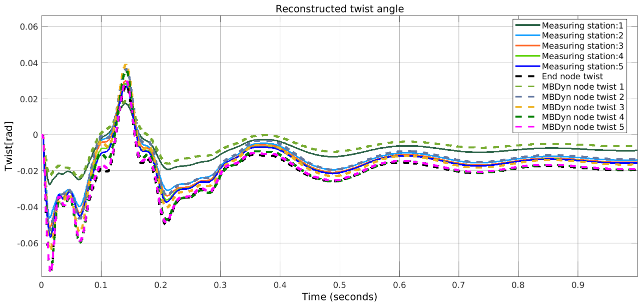


Figure 16: The real-time twist angle of the blade

7. Conclusions

The methodology proposed for the development of the real-time algorithm proved to be valid. Moreover, the procedure for the computations of the numerical shape functions from the modal FEM analysis has been proved to be correct.

The algorithm has been successfully validated using the beam with the circular cross-section, as well as using the Princeton beam with a rectangular cross-section. The results from the algorithm have been verified using the analytical results and the deformation of the nodes from the MBDyn simulation.

The real-time Simulink model has been applied to both aforementioned beams, as well as the helicopter rotor model. In the case of the latter, the MBDyn of the rotor has been simplified and the co-simulation has been performed under some assumptions. However, the obtained results are realistic, and after some improvements should be also exact.

The thesis also covered a deeper analysis of the implemented modal approach and nonlinear equations. The linearity of the solution provided by the algorithm depends on the load case and the source of strains used in the computations.

The number of modes that can be used in the simulation is strictly related to the number of measuring stations along the blade. The maximum number of modes that can be used is equal to the number of the measuring stations. The best quality of the results is provided by the maximum number of modes.

The analysis of the model also covered the addition of sensors on the cross-section of the beam. If the measurements from the sensors are affected by any kind of random error, then its influence on the final results might be decreased by additional sensors placed on the measuring stations. This has been proven for bending and torsion shape reconstruction.

To sum up, the considered model applied to the real-time algorithm can be successfully used in the shape reconstruction of a beam-like structure. It is robust and gives a broad field for adjustments to balance the quality of the results with the computational effort and complexity of the equipment used for the measurements.

8. Acknowledgements

I would like to thank Professor Pierangelo Masarati for advice and support in the development of the thesis.

I must express my profound gratitude to my parents, without them the accomplishment of the studies at Politecnico di Milano and the thesis at this university would not have been possible.

References

[1] Olivier A. Bauchau, Peter Betsch, Alberto Cardona, Johannes Gerstmayr, Ben Jonker, Pierangelo Masarati, and Valentin Sonneville. Validation of flexible multibody dynamics beam formulations using benchmark problems. *Multibody System Dynamics*, 2016.

[2] Giovanni Bernardini, Roberto Porcelli, Jacopo Serafini, and Pierangelo Masarati. Rotor blade shape reconstruction from strain measurements. *Aerospace Science and Technology*, pages 580–587, 2018.

[3] D. H. Hodges and E. H. Dowell. Nonlinear equations of motion for the elastic bending and torsion of twisted nonuniform rotor blades. Technical Report A-5711, NASA Ames Research Center, 12 1974.

[4] Donald W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 1963.

[5] Jacopo Serafini, Giovanni Bernardini, Roberto Porcelli, and Pierangelo Masarati. In-flight health monitoring of helicopter blades via differential analysis. *Aerospace Science and Technology*, pages 436–443, 2019.

POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Real-Time Algorithm for Deformed Shape Reconstruction of a Beam Applied to Helicopter Blades

Tesi di Laurea Magistrale in
Aeronautical Engineering - Ingegneria Aeronautica

Author: **Mateusz Krzysztof Garbacz**

Student ID: 925364
Advisor: Prof. Pierangelo Masarati
Academic Year: 2021-2022

Abstract

The aim of the presented Master thesis was to develop a real-time algorithm for shape reconstruction of a beam-like structure based on the measurements from strain sensors applied to helicopter blades. This report is written in order to elaborate on the achieved results as well as describe in detail the verification of the algorithm. The first part of the report is devoted to the description of the available shape reconstruction methods and applications of the aforementioned algorithms with special emphasis on applications in the aerospace industry, especially in helicopter rotor-state determination. The following part consists of the description of the formulas used to develop the algorithm, and beam models used for the verification of the algorithm. In addition, this section presents also an algorithm developed for numerical shape determination based on modal analysis of the helicopter blade. The report presents also in a graphical way the implementation of the algorithm into the Matlab script and later into the real-time Simulink model, which has been used in co-simulation with MBDyn software, using which multibody models of the beams has been prepared. Validation of the applied model has been performed using a circular beam model to which various load cases have been applied to check the behavior of the algorithm in different load conditions. In addition, the report consists of the analysis of the algorithm and considerations concerning the number of the sensors per section, the number of the measuring stations, and the number of modal shapes used in the algorithm. The algorithm was also verified using experimental results from the Princeton beam experiment. Lastly, the real-time Simulink model has been applied to a helicopter rotor model developed in MBDyn to present its application to the shape reconstruction of a helicopter blade.

Keywords: shape reconstruction; numerical modes determination; strain measurements; helicopter blade

Abstract in lingua italiana

L'obiettivo di questa tesi di Laurea Magistrale è sviluppare un algoritmo in tempo reale per la ricostruzione della forma di una struttura a trave basata sulle misurazioni dei sensori di deformazione applicati alle pale di elicottero. Questo documento elabora i risultati raggiunti e descrive in dettaglio la verifica dell'algoritmo. La prima parte del documento è dedicata alla descrizione dei metodi esistenti per la ricostruzione della forma e alle applicazioni dei suddetti algoritmi, con particolare attenzione ad applicazioni in ambito aerospaziale, in particolare alla ricostruzione dello stato di un rotore di elicottero. La parte successiva descrive le formule utilizzate per sviluppare l'algoritmo e i modelli di trave utilizzati per la verifica dell'algoritmo. Questa sezione presenta inoltre l'algoritmo sviluppato per la determinazione numerica della forma basata sull'analisi modale di una pala di elicottero. Il documento presenta anche in forma grafica l'implementazione dell'algoritmo sotto forma di script Matlab e successivamente di modello Simulink in grado di funzionare in tempo reale, che è stato utilizzato in co-simulazione con il software MBDyn, quest'ultimo utilizzato per simulare modelli multibody di travi. Il modello è stato validato utilizzando un modello di trave a sezione circolare, a cui sono state applicate varie condizioni di carico per verificare il comportamento dell'algoritmo in diverse condizioni di funzionamento. L'analisi dell'algoritmo ha richiesto anche considerazioni sul numero di sensori per sezione, di stazioni di misura e di forme modali necessarie per la ricostruzione della forma della trave. L'algoritmo è stato verificato anche utilizzando i risultati sperimentali della cosiddetta trave di Princeton. Il modello Simulink operante in tempo reale è stato infine applicato a un modello di rotore di elicottero sviluppato in MBDyn per illustrarne l'applicazione alla ricostruzione della forma di una pala di elicottero in condizioni operative.

Parole chiave: ricostruzione della forma; determinazione dei modi numerici; misura della deformazione; pale di elicottero

List of contents

Introduction

The main objective of the presented thesis is to develop a real-time algorithm for the purpose of a shape reconstruction of helicopter blades using a modal shape sensing approach. The algorithm exploits the nonlinear equations for a beam-like structure shape reconstruction that relates the flapwise and chordwise deformation with measurements of the axial strains and twist of the beam with shear strains [11]. The inspiration for the thesis was a helicopter blade model that has been developed in the laboratory of the Aerospace Science and Technology Department of Politecnico di Milano. The thesis also covers the topic of numerical modal shape functions determination using a finite element model of the aforementioned helicopter blade model.

In recent years, the shape reconstruction problem gained more attention from many industrial fields. It may be particularly interesting for the aerospace, space, automotive, and wind turbines sectors. Information about the state of an object may be useful for the monitoring of its structural integrity and applied to the Structural Health Monitoring (SHM) systems and Health and Usage Monitoring Systems (HUMS). Both can be used to increase the safety of a machine by early detection of a structural component malfunction. Those systems may also be used to implement predictive maintenance, which drives down maintenance costs for machine operators, increasing the attractiveness of the product for the potential clients. In the space and aerospace industry, the knowledge about the state of an object may be used for control optimization. In helicopter rotors, which are the focus of this thesis, the knowledge about the structural condition of the blade can be used for both; SHM/HUMS systems as well as optimization of control, through Rotor State Feedback (RSF) applied to the Flight Control System (FCS). More details about the application of shape reconstruction are given in section 1.1.

The shape of an object can be reconstructed using strain measurements but there are also other methods such as photogrammetry or direct fibre optic shape sensing (DFOSS). The shape reconstruction techniques are described in detail in section 1.2. In the thesis, only reconstruction using strain sensors is considered, due to the fact that it is the cheapest and the most available method. The modelled measurement system is based on the one used in the blade from the laboratory, there are used two types of sensors: strain gauges and optical fibres. A strain gauge rosette is used to measure an axial strain and a shear strain, while optical fibre is used to measure only axial strain.

The details of the formulas used in the thesis are presented in chapter 2. This section describes the used reference system and gives insight into models used in the development of the work. The main algorithm exploits aforementioned nonlinear equations for a shape reconstruction of a beam-like structure. Analytical beam models (that are presented in section 2.3) and multibody beam models have been used for the verification of the developed algorithm. The multibody beam models exploit the finite volume beam model, that is shown in section 2.4. The models prepared using MBDyn have been also used in co-simulation between MBDyn and the Simulink model.

The modal shape sensing approach requires the determination of the modal shape functions of an object. In the thesis, there were considered analytical and numerical shape functions. The former has been obtained by direct application of the commonly known formulas described in section 3.1, while the latter was obtained by modal analysis of the finite element model of the blade from the laboratory and post-processing of the harvested data. The analytical shape functions can be used under the assumption that the beam is uniform, they are the same for bending in flapwise and chordwise directions. The numerical shape functions have been obtained separately for chordwise, flapwise, and torsional modes. They are presented in section 3.2, while the algorithm applied to a Matlab script used for their determination is shown in section 2.5.

The application of the algorithm to the Matlab scrip and the Simulink model is briefly presented in chapter 4. This chapter also presents the Simulink model that was used in a co-simulation with the MBDyn beam models. Only analytical functions were applied to the Matlab script and the Simulink model.

Chapter 5 presents validation of the algorithm implemented into the software using a beam with a circular cross-section, together with some analysis of the applied mathematical model relating displacement to strains. The verification of the algorithm has been performed by a comparison of the obtained results with the results from analytical formulas and the nodal deformation being the output of the MBDyn simulations performed on the beam models. The algorithm was validated for small and large displacements, presented respectively in section 5.2 and section 5.3, using strains coming from analytical formulas, as well as strains harvested from the MBDyn model. In section 5.4 there is presented an analysis of the applied algorithm taking into account the linearity of the solution. This chapter of the thesis also contains the analysis of the relation between the number of the measuring stations located along the spanwise axis of the blade, as well as the influence of the number of mode shapes, used for the reconstruction of the beam deformation, on the quality of the results. Section 5.6 presents the influence of the additional sensors placed on the cross-section on the computed displacement at the measuring stations. In this case, it was assumed that the measurements are affected by some random noise. The chapter is concluded with the results of the real-time co-simulation between the Simulink model and the MBDyn model of the beam with a circular cross-section, with time varying load applied to the tip of the beam.

Another verification of the algorithm has been performed using the Princeton beam experiment. It is described in chapter 6. In this chapter, the algorithm has been applied to a beam with a rectangular cross-section subjected to tip loads placed under different loading angles. The results obtained from the algorithm have been compared with the results of the MBDyn model and experimental results from the scientific paper written by Bauchau et al [6]. This chapter is concluded with the results of the real-time co-simulation of the rectangular beam.

Chapter 7 presents the application of the developed real-time Simulink model to a helicopter blade from a multibody helicopter model of the Bölkow BO-105 machine. The model of the helicopter has been developed at the Politecnico di Milano. The real-time Simulink model has been applied using some simplifications and making assumptions that are presented in detail in section 7.1. The results that are shown in section 7.2, are the proof that the developed algorithm can be successfully used in a real-time determination of helicopter blade state.

The thesis is concluded in chapter 8 that presents the achieved objectives, together with possible future developments of the work. The thesis shows the potential of the real-time shape reconstruction algorithm with particular emphasis put on the application of this method to helicopter blades.

1 | State-of-the-Art

Nowadays, the shape reconstruction of an object gains more attention in many engineering fields but the industry that is particularly interested in this problem is the aerospace one. Shape reconstruction methods might help to reduce the cost of aircraft maintenance, increase safety, and optimize the control system of an aircraft. In the aerospace sector, the shape reconstruction technique can be applied especially to the helicopter blades and aircraft wings, which are crucial for the operational safety of a vehicle. In modern structures, blades and wings are made of composite materials with a very complex internal structure to allow significant deformation of the structure for the purpose of flight optimization and stress reduction. Due to that, the industry is also interested in a real-time technique to obtain the state of the object, which might be later used for the optimization of the control laws of the vehicle, as well as validation of the fully aeroelastic models in-flight. The real-time system can be also integrated with the already used health and usage monitoring on-board systems to enable predictive maintenance, fatigue logging, and damage detection [24]. This chapter presents the main application of the shape reconstruction techniques as well as the different methods used to fulfil this objective.

1.1. Application of shape reconstruction

In this paragraph, special attention is paid to the application of shape reconstruction in aerospace structures, mainly helicopter blades.

Blades are among the most important elements of rotary-wing aircraft because they provide the lift to counteract the weight of the vehicle and the control moments to enable execution of the mission manoeuvres; they must perform well over a very broad flight envelope. They operate in very severe conditions because they are subjected to high dynamic loads that depend on flight conditions as well as the azimuthal angle of the blade. Blades are constantly interacting with the airflow what is a source of significant and unpredictable vibrations [24]. They are flexible to reduce the vibrations passed from the blades to the airframe, but also to reduce the aerodynamic asymmetry between the advancing and retreating sides of the rotor. The variability of the aerodynamic loads and flow interaction are very complex, making it impossible to be modelled for the rotorcraft in different flight conditions [22]. Another purpose for which the flexibility of the blade is required is the control of the aircraft, flapping deformation is used to direct rotor thrust (which is normal to the tip path plane), thus generating the control moments of the aircraft [7].

Figure 1.1 presents the forces acting on an infinitesimal small blade element, where Ω is the angular velocity of the blade, d_q is the contribution of the aerodynamic forces, df_c is the centrifugal force, and df_i is the force due to the inertia of the element. The inertia and aerodynamic loads cause deflection in flapwise and lagwise directions, while the centrifugal force causes a deformation in the spanwise direction.

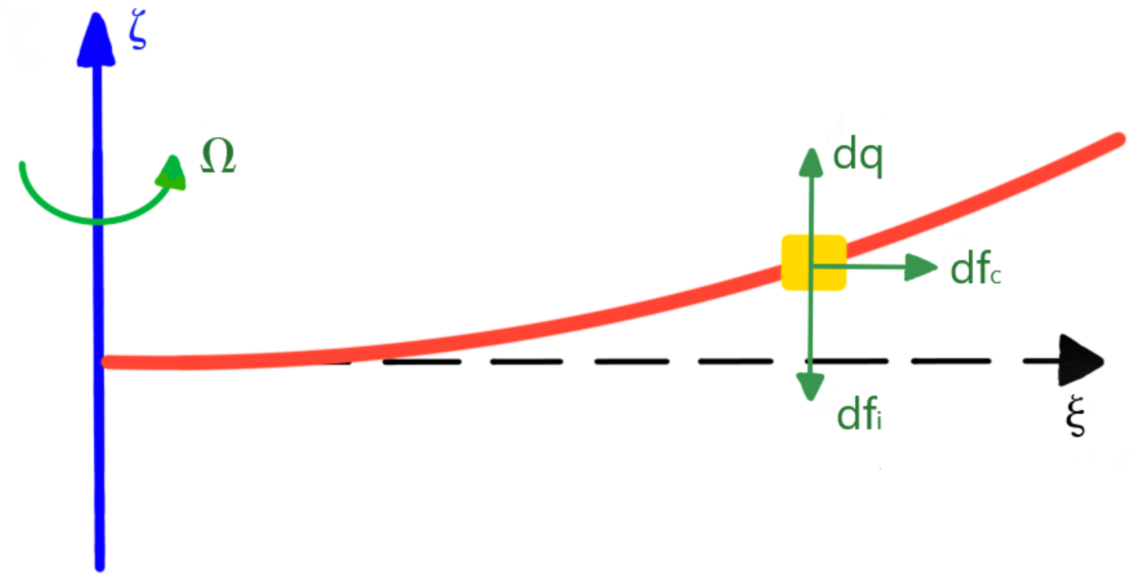


Figure 1.1: Forces acting on an infinitesimal blade element

Figure 1.2 presents the tip-path plane (TPP) of the aircraft together with a thrust vector, which is always normal to the TPP. The tip path plane in the articulated rotors is usually simplified to the plane created by the tips of the blade inclined by a flapping angle, however in other type of rotors the tip path plane depends only on the flexibility of the blades. More accurate knowledge of this plane may lead to the optimization of the control laws, especially in more difficult flight conditions [21].

Modern blades have usually very complex, non-uniform internal structure, and this also affects the process of the blade shape reconstruction making it much more complex.

Figure 1.3 presents the internal structure of a helicopter blade, the leading edge is made of titanium and steel, to make the blade strong and able to withstand the loads to which the

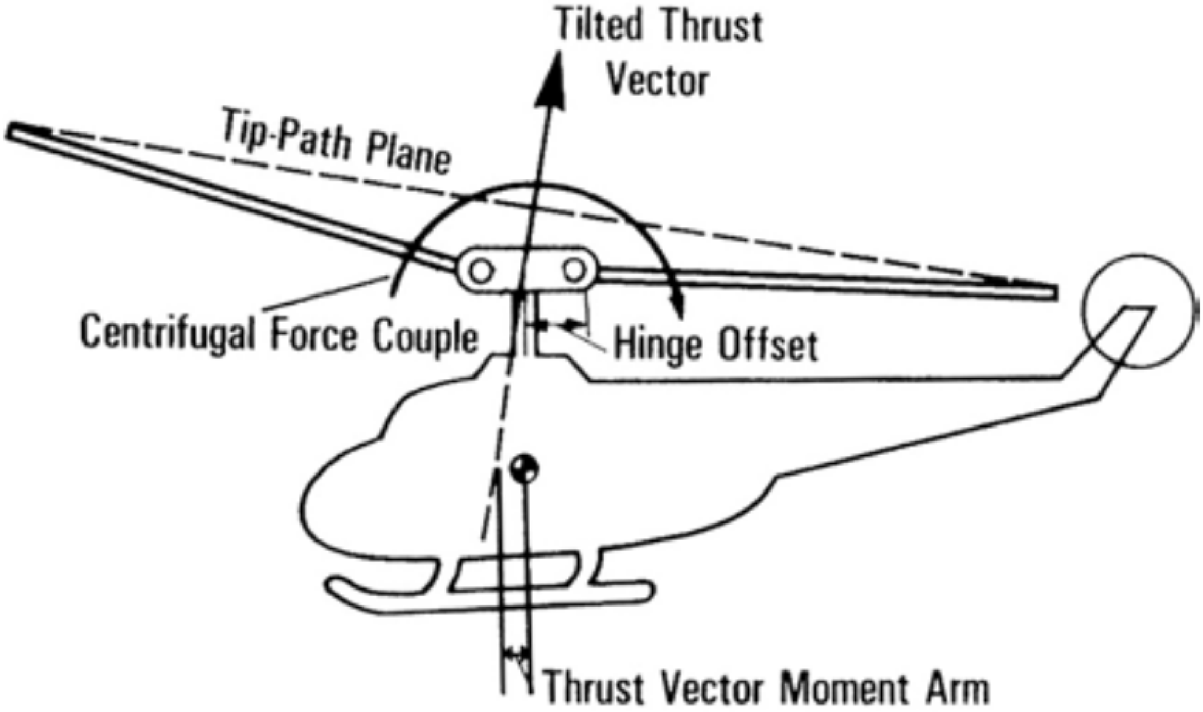


Figure 1.2: Tip-path plane in the case of articulated rotor [21]

leading part is subjected. The internal spar that provides the shape is made of fibreglass, which is stiff and lightweight material, and the trailing edge is made of fibreglass skin and core that is usually some kind of foam to provide the stiffness, shape, and reduce the mass of the entire structure [17].

In helicopters, the shape reconstruction of a blade can be used for various purposes.
First, it can be used for a structural health monitoring system (SHM), which is still in development in contrary to a simple health and usage monitoring systems (HUMS) that are already implemented in some machines. The former aims to monitor the real-time condition of the structural elements, such as blades, while the latter are based on data such as number of take-offs and landings, G-force during a landing (to determine if the landing was normal or hard), or vibrations of engines and gearboxes. Based on those data, the operator can apply predictive maintenance procedures and reduce the operational costs of the machine. SHM may drive the maintenance costs even lower, adjusting the maintenance of the blades to their real structural conditions by improving the failure detection. Any kind of blade damage may decrease the performance of the device as well as have fatal consequences (3% of the helicopter accidents were caused by a failure of a helicopter blade). Due to this fact, knowledge about the blade health condition is crucial for safety as well as predictive and optimized maintenance management [22].

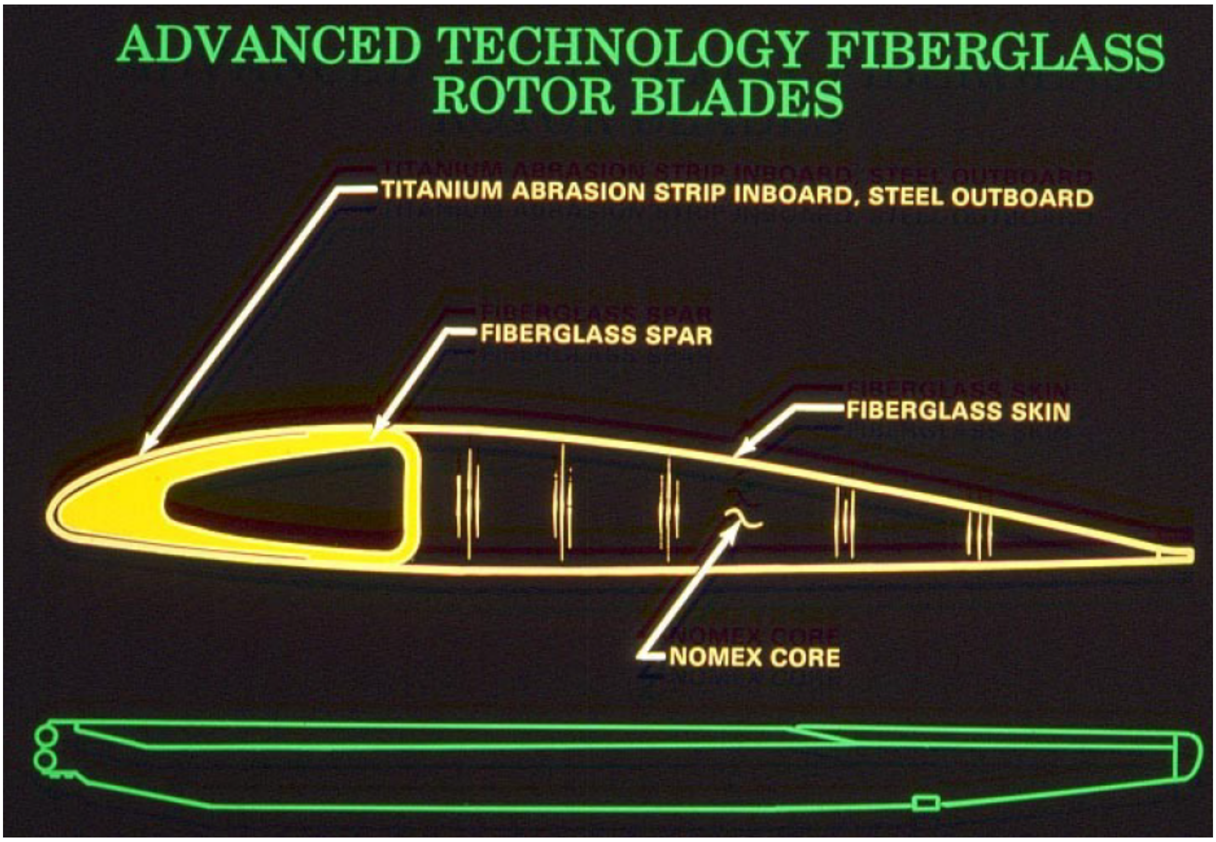

ADVANCED TECHNOLOGY FIBERGLASS ROTOR BLADES
TITANIUM ABRASION STRIP INBOARD, STEEL OUTBOARD
FIBERGLASS SPAR
FIBERGLASS SKIN
NOMEX CORE

Figure 1.3: Internal structure of a helicopter blade [17]

The most severe threats for blades are the fatigue caused by time-varying stresses; in addition, very dangerous are low-velocity impacts, and progressive damage accumulation. At present, any kind of incident related to rotor blades must be reported by pilots in the documentation of the vehicle and directed to the CAMO (Continuing Airworthiness Management Organisation) of the operator. Subsequently, the office takes a decision, based on the type of incident, to reduce the operational life of the blade, or whether it is necessary to ground the machine and immediately verify the damage. In both cases, the maintenance costs increase significantly, because the verification of the blade can only be performed after disassembly of the blade from the rotor. Later, the blade must undergo non-destructive structural inspection, which can be performed by tapping, ultrasonic methods, radiography, X-ray, or thermal field methods. All of them are usually very costly because they are performed in an external facility by experienced staff and require at least an approximate location of the damage. Application of SHM may allow for predictive maintenance of blades, reduce the downtime of a machine, and simultaneously increase the safety of operations extending the life of the component which will be inspected, repaired, or exchanged only if necessary. Such a design approach may significantly drive down the costs of the operator, making the product more attractive in the market; at present, 25% of the operational costs of the machine are related to maintenance [22].

Early warning about structural issues of the blade can be also useful to slow down crack propagation with a proper control strategy of the pilot (reduction of flight speed by 20% may reduce the crack propagation by 50% increasing the lifetime of the component) [22].

Nowadays special care is paid to extend the lifetime of machines; for example, the CH-47 Chinook is expected to reach around 100 years of machine life cycle (it is already in use since 1960s). SHM in the future will be also used for this purpose, increasing safety of operations of ageing machines. Current issues related to the SHM are caused by many false positives and negatives, also problematic is the reconstruction of the blade shape based on the limited number of sensors.

Another aim for which shape reconstruction might be used is to optimize the performance of the Flight Control System (FCS) of a helicopter, through a detailed knowledge of the rotor states [7]. The main advantage of a Rotor State Feedback application in the control system is that turbulence alleviation is improved in both high and low-frequency ranges [14, 20]. It can also improve performance of the vehicle and handling of the helicopter in precise manoeuvres [12]. For the first time, RSF system was implemented into FCS in a CH-53 and the test data showed significant improvement of gust disturbances [14]. The use of the rotor state information thus might be especially useful for helicopters operating in very difficult conditions such as SAR missions and military helicopters which must be able to perform rapid manoeuvres. The ability of RSF to improve handling of an aircraft has been proved by studies performed on real helicopters [5].

Nowadays, particular emphasis is placed on the reduction of the acoustic impact of helicopters, since those vehicles are irreplaceable in operations like SAR, Air Rescue, police operations, and noise is a factor that limits the public acceptance of rotorcraft in highly populated areas. Noise is a complex effect generated mechanically due to aerodynamic interaction between the vehicle and the air. It was discovered that it is related to three main factors which are: advance ratio, thrust coefficient, and tip path plane angle of attack. The idea is to implement a Rotor State Feedback into the control system and in this way estimate the acoustic impact on the environment and reduce it by a correct manoeuvre sequence. Deep investigation of this problem was one of the main objectives of the CleanSky MANOEUVRES project [23].

1.2. Methods of shape reconstruction

Different methods might be used in the shape reconstruction of a helicopter blade. One of the commonly used object shape reconstruction techniques is photogrammetry. This is an optical method in which a camera is used to record or take photos of an object from different angles. Markers are attached to an object at the reference points, the position of which is evaluated in the post-processing. This method gives good results and is usually cheap; however, its implementation in the shape reconstruction of a helicopter blade is very problematic, except perhaps in laboratory or wind tunnel. The main problem is that only objects that are in sight of the camera can be monitored, so the camera must be appropriately placed on the rotor hub, taking into account the fact that the markers plane must be placed at a sufficiently high angle with respect to the camera to give accurate results [7]. This method is also useless at night or in low visibility conditions such as fog. Another drawback is that, although in general this method is cheap, when applied to a helicopter blade it might be very expensive because of the cost of high-speed cameras that can withstand an environment with high vibrations. Currently, photogrammetry can be conveniently used only in the case of ground tests [9]. Figure 1.4 presents the test bench for an optical rotor blade shape reconstruction performed during ground tests.

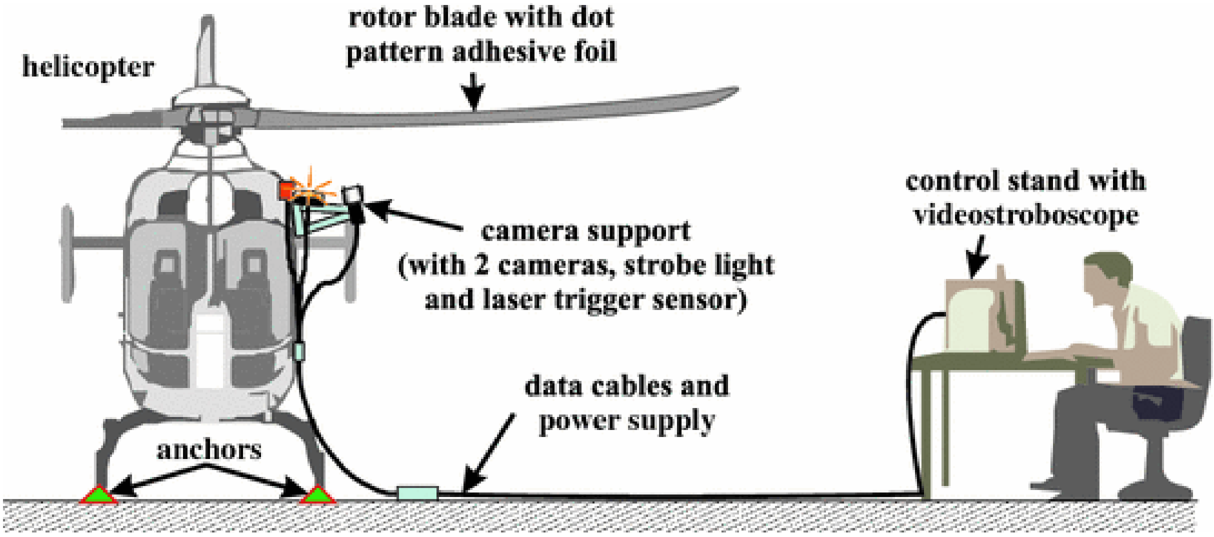

helicopter
rotor blade with dot pattern adhesive foil
control stand with videostroboscope
camera support (with 2 cameras, strobe light and laser trigger sensor)
data cables and power supply
anchors

Figure 1.4: Test bench for a photogrammetry shape reconstruction [19]

Strain-based methods are a reasonable alternative to optical techniques. They usually exploit strain gauges as well as optical fibre sensors, also known as Fibre Bragg Grating (FBG). The strain method is based on a differential strain that is measured by different sensors being placed in offset to each other. They must be mounted on the opposite surface of an object to determine its curvature. Despite being a very simple solution, strain sensors face some common issues, which are still under research [24].

The first problem is related to the number of sensors and their positioning. The more sensors are being used the resolution of the results improves, however, on the other side, they require more electrical power to be delivered and computational power to be post-processed. This brings another issue because the power and the data must be transmitted between a fixed reference frame of the vehicle and a rotating reference frame of the rotor. Conventionally, it is done by a slip ring; however, it generates a lot of noise in the measurements. The measurements may also fail due to the thermal gradient within the object. In the case of the shape reconstruction using strain measurements, it is also necessary to create a detailed structural FEM model of the object and perform its preliminary analysis to relate the strain to the deformation of a blade. The model must be detailed enough; otherwise, the final results will be corrupted [22]. On the other hand, the strain-based methods are the most developed and the cheapest, giving the best trade-off between the price and quality of the results, assuming that their application has been performed with sufficient care paid to the details described above.

The most common and the cheapest sensors are strain gauges. Figure 1.5 presents a simple arrangement of the strain gauge with its main components [1].

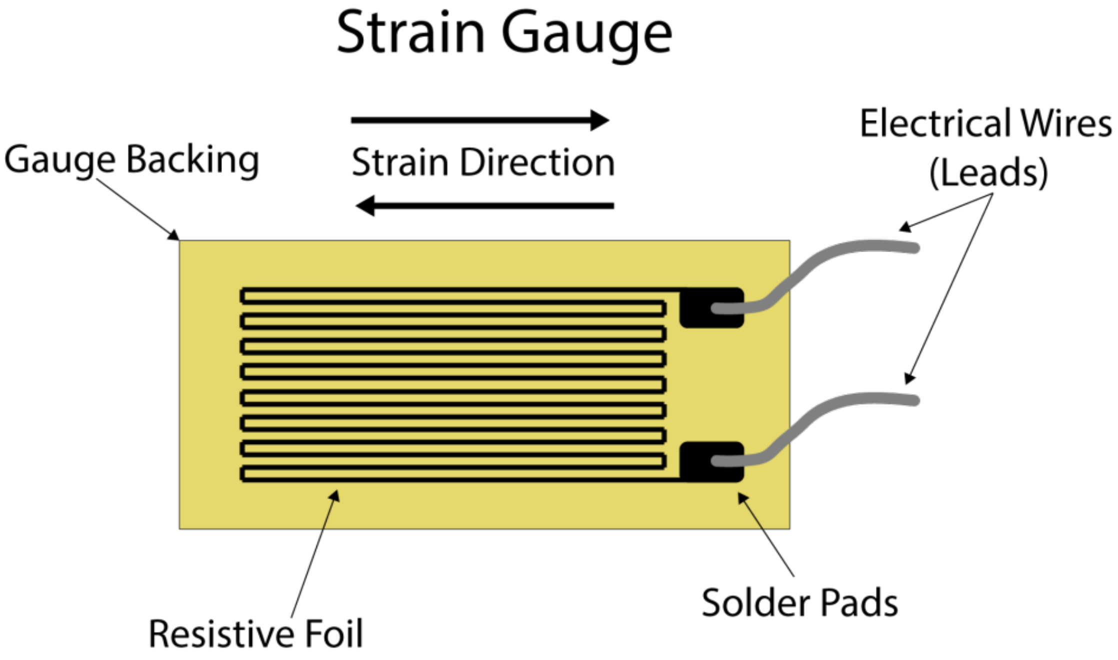


Figure 1.5: Strain gauge [4]

It is a type of electrical sensor that is glued to the measured component. When the gauge is elongated, its electric resistance changes, giving different electric output, which is later post-processed. Those sensors are small and highly sensitive, so they can measure the contraction or expansion even if it is very small. There are a lot of strain gauge types and orientations for different purposes. In blade shape reconstruction, their main advantage is that they are cheap and mounted in a way that allows for their replacement with ease.

The main disadvantage of this type of sensor is that either it affects the aerodynamic profile of a blade because of the wires that must be routed on its surface, or it is invasive if the wires are to be embedded in the structure. Also, installation and wiring takes a lot of time and is complex, error prone, and may be irreparable if wires are embedded in the structure. They are also sensitive to the environment; temperature gradients and water may affect their readings, increasing the risk of sensor's failure during the test campaign. On the other hand, if they are mounted correctly, they provide a very good quality of measurements. Strain sensors are usually organized in rosettes of 3 sensors to measure (nearly) co-located axial and shear strains [24].

Another type of sensor that measures the strain is Fibre Bragg Grating (FBG). This kind of sensor is more modern and exploits the optical fibre properties to accurately measure strains. Its principle of work is presented in Figure 1.6.

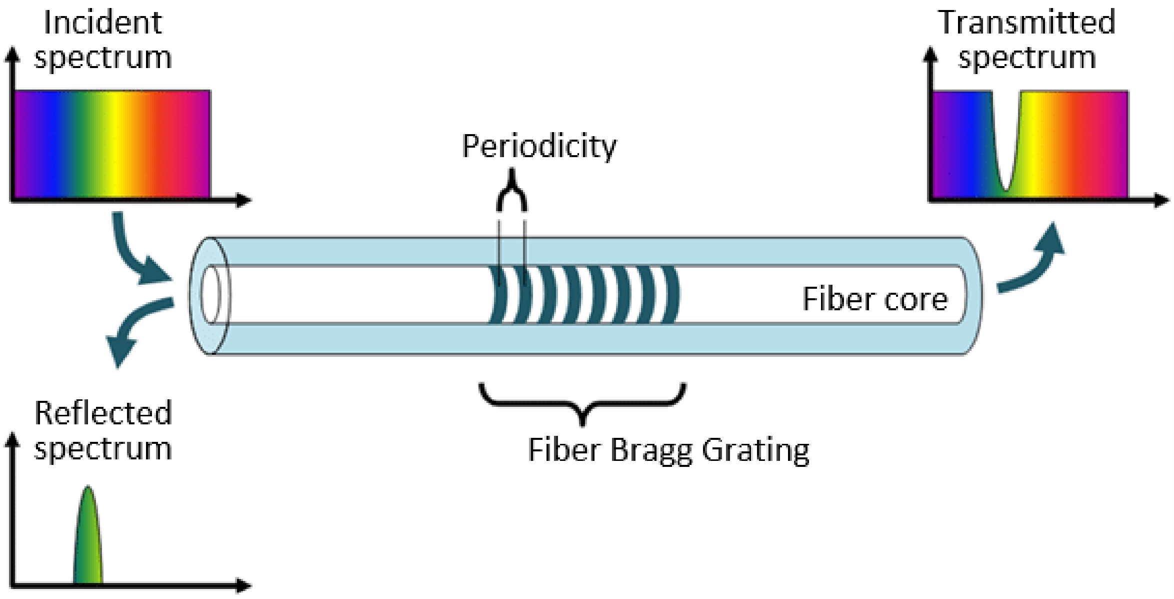


Figure 1.6: FBG sensor [2]

In this device, the core of the fibre is exposed to a periodic pattern of intense laser light with some known incidence spectrum, creating a grating in the fibre core. At the periodic refractions light is reflected and the signals combine into a large reflection at a particular wavelength. This refers to a Bragg condition and Bragg wavelength. Light signals at other wavelengths are transparent. When the sensor is strained, the reflected wavelength changes due to the changed periodicity in the sensor caused by applied elongation or contraction of the sensor [3].

FBG has a lot of advantages. The sensor itself is very flexible, small, and has low mass. Thanks to this it does not significantly affect the structural component. It may also be laminated in the structure or glued to the surface, depending on the needs. It requires also much smaller power than the conventional strain gauge. It has been already implemented in some helicopters to determine the wear of rotor head components. Due to the properties of the optical fibre, the measurements are not affected by the environment and electromagnetic interference, which increases the reliability of the results [24].

The most modern shape reconstruction technique directly exploits the measurement of the optical fibre displacement. This method is called Direct Fibre Optic Shape Sensing (DFOSS) and exploits long optical fibres that can be laminated in a structure or glued to it. The shape of an object is determined directly from the path of the deformed fibre in three dimensions, there is no dependency on strain, which simplifies post-processing of the signal and increases the reliability of the final, reconstructed shape. However, the application of this method is still under investigation and, at present, it is expensive in comparison to the strain-based techniques for shape reconstruction [24].

2 | Methodology

The following chapter presents the formulas and reference systems that have been used in the development of the presented thesis.

2.1. Reference system

The global reference system convention is shown in Figure 2.1. The ξ, Y, and Z axes are respectively in the spanwise, chordwise, and flapwise direction of the blade in the undeformed configuration and considering null built-in twist.

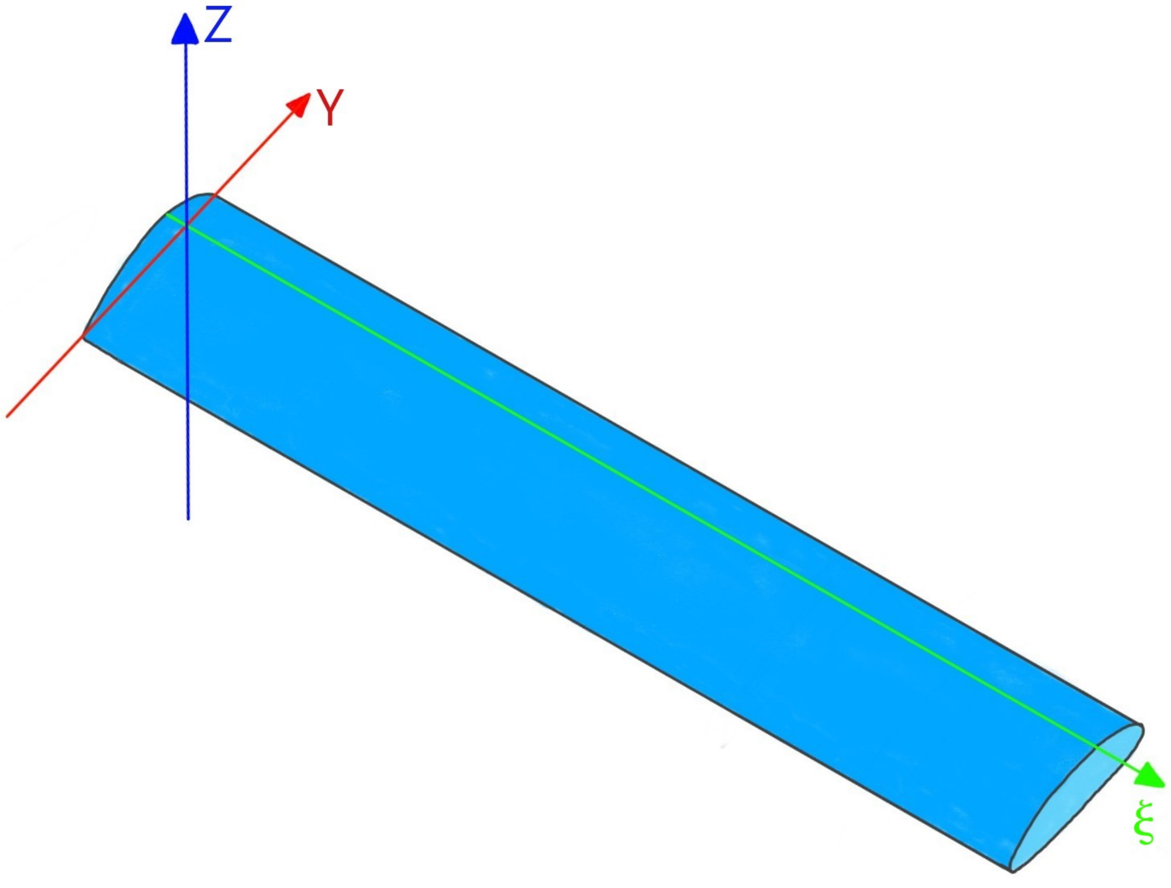


Figure 2.1: Global reference system

Figure 2.2 presents the cross-section of the blade with two reference systems, the $Y-Z$ reference system is the global one, while the $\eta-\zeta$ reference system is the local reference system of the section, that is rotated by the built-in twist angle. Both reference systems have their origin in the position of the elastic axis of the beam.

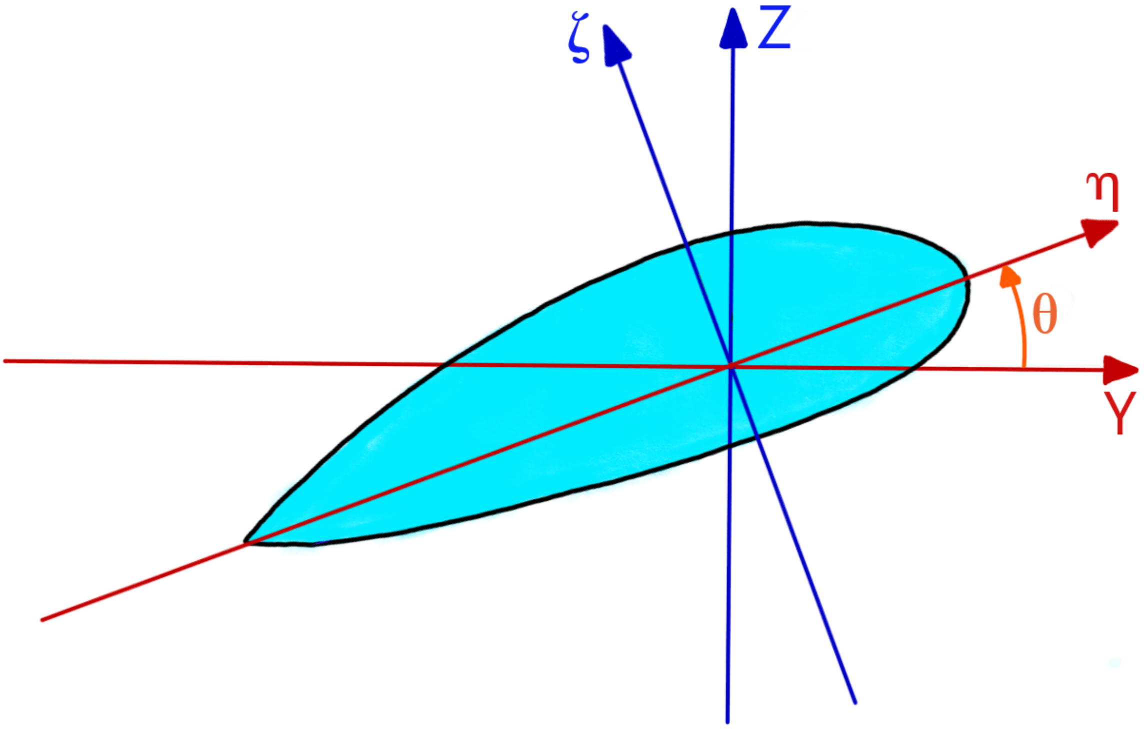


Figure 2.2: Cross-section's reference system

2.2. Nonlinear equations for a beam shape reconstruction

The main aim of this thesis is to develop a real-time algorithm for the reconstruction of the deformed beam shape using strain measurements from strain gauges and optical fiber sensors. In the considered case, the strain gauges are used to measure axial and shear strain, while optical fiber sensors are used to measure axial strain only. Those assumptions are based on the model of the blade, which is available in the laboratory of the Aerospace Science and Technology Department of Politecnico di Milano, and was an inspiration for the presented thesis.

A modal shape sensing approach is proposed in the algorithm. It is capable to reconstruct the shape of a beam-like structure subjected to general deformations including torsion, in-plane, and out-of-plane bending, and handles nonlinear terms up to 2nd order. The latter property is important for flight dynamics and aeroelastic control applications because limited knowledge about rotor kinematics may lead to incorrect results [7].

The relation between strain and displacement of the beam comes from the nonlinear model of motion for the elastic bending and torsion of twisted nonuniform rotor blades [11]. The formula of Equation (2.1) presents the relation between the axial strain and displacement of the beam, while Equation (2.2) and Equation (2.3) show the relation between shear strain and derivative of twist angle.

$$\begin{aligned}\varepsilon_{\xi\xi} &= u' + \frac{1}{2}(v'^2 + w'^2) + (\eta^2 + \zeta^2)\left(\theta'\phi' + \frac{\phi'^2}{2}\right) \\ &\quad - v''[\eta\cos(\theta+\phi) - \zeta\sin(\theta+\phi)] - w''[\eta\sin(\theta+\phi) + \zeta\cos(\theta+\phi)] \qquad (2.1)\\ \varepsilon_{\xi\eta} &= -\frac{\zeta\phi'}{2} \qquad (2.2)\\ \varepsilon_{\xi\zeta} &= \frac{\eta\phi'}{2} \qquad (2.3)\end{aligned}$$

where v and w are the in-plane and out-of-plane displacements, ϕ is the torsional rotation, and the “prime” ($'$) indicates derivation with respect to ξ.

In the modal approach, space and time coordinates separation is exploited; the equations are presented below.

$$v(\xi, t) = \boldsymbol{n}_v(\xi)\boldsymbol{r}_v(t) \qquad (2.4)$$

$$w(\xi, t) = \boldsymbol{n}_w(\xi)\boldsymbol{r}_w(t) \qquad (2.5)$$

$$\phi(\xi, t) = \boldsymbol{n}_\phi(\xi)\boldsymbol{q}(t) \qquad (2.6)$$

In the proposed algorithm, the torsion and bending problems are decoupled; the order of solution is strictly defined: the torsional problem must be considered and solved as the first one. If the chordwise and flapwise bending functions are coupled, then the chordwise and flapwise displacements are computed using the same set of generalized coordinates $\boldsymbol{r}$.

2.2.1. Torsion problem

Equation (2.7) and Equation (2.8) present the relation between the twist and shear strain along the local axes η and ζ.

$$\varepsilon_{\xi\eta_k} = -\frac{\zeta_k}{2}\boldsymbol{n}'_{\phi}(\xi_s)\boldsymbol{q}(t) \tag{2.7}$$

$$\varepsilon_{\xi\zeta_k} = \frac{\eta_k}{2}\boldsymbol{n}'_{\phi}(\xi_s)\boldsymbol{q}(t) \tag{2.8}$$

In the considered case, the measurements come from a surface strain gauge $\varepsilon_{\xi t_k}$ that gives shear strain tangent to the surface of the beam. It can be considered as a scalar product of vector $[\varepsilon_{\xi\eta}; \varepsilon_{\xi\zeta}]$ and a unit vector that is tangent to the surface of the beam at the position of the strain gauge. Taking this into account, Equation (2.7) and Equation (2.8) reduce to Equation (2.9).

$$\varepsilon_{\xi\eta_k}\boldsymbol{t}_{\eta_k} + \varepsilon_{\xi\zeta_k}\boldsymbol{t}_{\zeta_k} = \frac{-\zeta_k\boldsymbol{t}_{\eta_k} + \eta_k\boldsymbol{t}_{\zeta_k}}{2}\boldsymbol{n}'_{\phi}(\xi_s)\boldsymbol{q}(t) \tag{2.9}$$

The left-hand side of the equation can be denoted as a shear strain measured by the strain gauge $\varepsilon_{\xi t_k}$. Equation (2.10) presents the final expression:

$$\varepsilon_{\xi t_k} = \frac{-\zeta_k\boldsymbol{t}_{\eta_k} + \eta_k\boldsymbol{t}_{\zeta_k}}{2}\boldsymbol{n}'_{\phi}(\xi_s)\boldsymbol{q}(t) \tag{2.10}$$

Collecting all the shear strain measurements into a vector and all the left hand side expressions related to the sensor position and torsional shape functions, Equation (2.11) is obtained.

$$\boldsymbol{\varepsilon}_{\xi t_k} = \boldsymbol{A}_t\boldsymbol{q}(t) \tag{2.11}$$

2.2.2. Bending problem

To compute the displacements v and w, it is convenient to exploit the difference between pairs of axial strains measured at one spanwise measuring station. The axial strain is measured by both, strain gauges and optical fibres. Equation (2.12) shows that the contribution of the axial displacement derivative and the quadratic contributions in the transverse displacement derivatives simplify; the problem becomes linear in w'' and v''.

$$
\begin{aligned}
\varepsilon_{\xi\xi_{k1}} - \varepsilon_{\xi\xi_{k2}} = {} & \cancel{u'_2(\xi_s) - u'_1(\xi_s)} + \cancel{\frac{1}{2}(v'^2_2(\xi_s) + w'^2_2(\xi_s))} - \cancel{\frac{1}{2}(v'^2_1(\xi_s) + w'^2_1(\xi_s))} \\
& + [(\eta^2_{k1} + \zeta^2_{k1}) - (\eta^2_{k2} + \zeta^2_{k2})]\left(\theta'(\xi_s)\phi'(\xi_s) + \frac{\phi'^2(\xi_s)}{2}\right) \\
& - v''(\xi_s)[(\eta_{k2} - \eta_{k1})\cos(\theta(\xi_s) + \phi(\xi_s)) - (\zeta_{k2} - \zeta_{k1})\sin(\theta(\xi_s) + \phi(\xi_s))] \\
& - w''(\xi_s)[(\eta_{k2} - \eta_{k1})\sin(\theta(\xi_s) + \phi(\xi_s)) + (\zeta_{k2} - \zeta_{k1})\cos(\theta(\xi_s) + \phi(\xi_s))]
\end{aligned}
\tag{2.12}
$$

The equation can be further simplified by exploiting the matrices multiplication and by denoting the distance between the position of sensor k and the elastic axis as R_k. Equation (2.13) presents the formula of the single measuring section in a matrix form.

$$
\begin{aligned}
& \underbrace{\left\{\Delta\varepsilon_{\xi\xi_{k1k2}} - (R^2_{k1} - R^2_{k2})\left(\theta'_s\phi'_s + \frac{{\phi'_s}^2}{2}\right)\right\}}_{\boldsymbol{\Delta\varepsilon_s}} \\
& = -\underbrace{\begin{bmatrix} \Delta\eta_{k1k2} & \Delta\zeta_{k1k2} \\ \dots & \dots \end{bmatrix}}_{\boldsymbol{D_s}} \underbrace{\begin{bmatrix} \cos(\theta_s + \phi_s) & \sin(\theta_s + \phi_s) \\ -\sin(\theta_s + \phi_s) & \cos(\theta_s + \phi_s) \end{bmatrix}}_{\boldsymbol{R_s}} \underbrace{\begin{Bmatrix} \boldsymbol{n}''_v(\xi_s) \\ \boldsymbol{n}''_w(\xi_s) \end{Bmatrix}}_{\boldsymbol{N_s}} \boldsymbol{r}(t)
\end{aligned}
\tag{2.13}
$$

In the case of a limited number of sensors, it is important to make sure that matrix $\boldsymbol{D}_s$ is not ill-conditioned; for this purpose, a minimum of three sensors per section are needed. Matrix $\boldsymbol{R}_s$ is square and expresses a rigid rotation of the section about the axis of the blade by an angle that varies along the spanwise axis. Matrix $\boldsymbol{N}_s$ contains the chordwise and flapwise bending shape functions evaluated at the section. The shape functions and matrix $\boldsymbol{D}_s$ are computed once and for all, while other matrices change at every iteration.

According to theory, when the number of measurements per section exceeds a minimum number which is three, the problem becomes overdetermined. However, considering that measurements are affected by noise, and the other errors that may affect them, more measurements should guarantee a least-squares compensation of random errors and improve the quality of the final result.

Considering s spanwise stations, the equation above can be written as in the form that is presented in Equation (2.14).

$$\begin{Bmatrix} \boldsymbol{\Delta\varepsilon}_1 \\ \vdots \\ \boldsymbol{\Delta\varepsilon}_s \end{Bmatrix} = \begin{bmatrix} \boldsymbol{D}_1\boldsymbol{R}_1\boldsymbol{N}_1 \\ \vdots \\ \boldsymbol{D}_s\boldsymbol{R}_s\boldsymbol{N}_s \end{bmatrix} \boldsymbol{r}(t) = \begin{bmatrix} \boldsymbol{D}_1 & \dots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \dots & \boldsymbol{D}_s \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_1 & \dots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \dots & \boldsymbol{R}_s \end{bmatrix} \begin{bmatrix} \boldsymbol{N}_1 \\ \vdots \\ \boldsymbol{N}_s \end{bmatrix} \boldsymbol{r}(t) \tag{2.14}$$

The problem can be reformulated as it is shown in Equation (2.15). Matrix $\boldsymbol{N}$ contains the shape functions of all stations.

$$\begin{bmatrix} \boldsymbol{R}_1^T & \dots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \dots & \boldsymbol{R}_s^T \end{bmatrix} \begin{bmatrix} \boldsymbol{D}_1^+ & \dots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \dots & \boldsymbol{D}_s^+ \end{bmatrix} \begin{Bmatrix} \boldsymbol{\Delta\varepsilon}_1 \\ \vdots \\ \boldsymbol{\Delta\varepsilon}_s \end{Bmatrix} = \begin{bmatrix} \boldsymbol{N}_1 \\ \vdots \\ \boldsymbol{N}_s \end{bmatrix} \boldsymbol{r}(t) = \boldsymbol{N}\boldsymbol{r}(t) \tag{2.15}$$

The final solution of the problem is shown in Equation (2.16).

$$\boldsymbol{r}(t) = \boldsymbol{N}^+ \begin{bmatrix} \boldsymbol{R}_1^T\boldsymbol{D}_1^+\boldsymbol{\Delta}\varepsilon_1 \\ \vdots \\ \boldsymbol{R}_s^T\boldsymbol{D}_s^+\boldsymbol{\Delta}\varepsilon_s \end{bmatrix} \tag{2.16}$$

2.2.3. Minimization algorithm

The complete, overdetermined problem must be solved in a least-squares sense [15]. Equation (2.17) presents the complete formulation of the torsion problem with all shear strain measurements collected into a single vector $\boldsymbol{\varepsilon}_{\xi t_k}$.

$$\boldsymbol{e}_t = \boldsymbol{A}_t\boldsymbol{q} - \boldsymbol{\varepsilon}_{\xi t_k} \tag{2.17}$$

Equation (2.18) shows the complete bending problem with all strain measurements at all sections. Vector $\boldsymbol{T}(\boldsymbol{q})$ is quadratic in $\boldsymbol{q}$.

$$\boldsymbol{e}_b = \boldsymbol{A}_b\boldsymbol{r} - \boldsymbol{\varepsilon}_{\xi\xi} + \boldsymbol{T}(\boldsymbol{q}) \tag{2.18}$$

The problem is solved by minimizing the convex form that is presented in Equation (2.19).

$$J = \frac{1}{2}(\boldsymbol{e}_t^T \boldsymbol{W}_t \boldsymbol{e}_t^T + \boldsymbol{e}_b^T \boldsymbol{W}_b \boldsymbol{e}_b^T) \tag{2.19}$$

where $\boldsymbol{W}_t$ and $\boldsymbol{W}_b$ are symmetric, positive definite weight matrices. In the proposed approach, the sequence of the solution is strictly defined. The torsional problem is solved first; for this purpose, matrix $\boldsymbol{W}_t$ is set to $\boldsymbol{I}$ and matrix $\boldsymbol{W}_b$ is set to $\mathbf{0}$. The result is presented in Equation (2.20):

$$\boldsymbol{q} = \boldsymbol{A}_t^+ \boldsymbol{\varepsilon}_{\xi t_k} \tag{2.20}$$

where $\boldsymbol{A}_t^+$ is the Moore-Penrose pseudoinverse of matrix $\boldsymbol{A}_t$.

Then the bending problem is solved by setting $\boldsymbol{W}_b$ to $\boldsymbol{I}$, and $\boldsymbol{W}_t$ to $\mathbf{0}$. Equation (2.21) presents the result of this manipulation:

$$\boldsymbol{r} = \boldsymbol{A}_b^+ (\boldsymbol{\varepsilon}_{\xi\xi} - \boldsymbol{T}(\boldsymbol{q})) \tag{2.21}$$

with $\boldsymbol{q}$ calculated in the previous step of Equation (2.20).

2.3. Analytical beam models

The algorithm for the shape reconstruction of the beam has been validated using an analytical model presented in this section. The analytical validation of the beam has been performed using two types of beam section, a circular and a rectangular one, to which a flapwise and a chordwise load has been applied together with a torque. The axial stresses along the Y and Z axes are presented respectively in Equation (2.22) and Equation (2.23), while the shear stress is shown in Equation (2.24).

$$\sigma_y = \frac{F_y(L - \xi_k)\eta_k}{I_z} \tag{2.22}$$

$$\sigma_z = \frac{F_z(L - \xi_k)\zeta_k}{I_y} \tag{2.23}$$

$$\tau_{yz} = \frac{M_t R_k}{J} \tag{2.24}$$

The analytical axial and shear strains are used to simulate the measurements of the strain sensors located at the measuring stations. They are computed using respectively Equation (2.25) and Equation (2.26).

$$\varepsilon_{\xi\xi} = \frac{\sigma_z - \sigma_y}{E} \tag{2.25}$$

$$\varepsilon_{\xi t_k} = \frac{\tau_{yz}}{G} \tag{2.26}$$

The transverse displacement of the beam can be computed using two models: the Euler-Bernoulli and the Timoshenko beam models. For loads applied at a free end, the former is presented in Equation (2.27) and Equation (2.28), and the latter in Equation (2.29) and Equation (2.30).

$$v = \frac{F_y \xi_k^2}{6EI_z}(3L - \xi_k) \tag{2.27}$$

$$w = \frac{F_z \xi_k^2}{6EI_y}(3L - \xi_k) \tag{2.28}$$

$$v = \left(\frac{\xi_k^2}{6EI_z}(3L - \xi_k) + \frac{\xi_k}{GA_y} \right) F_y \tag{2.29}$$

$$w = \left(\frac{\xi_k^2}{6EI_y}(3L - \xi_k) + \frac{\xi_k}{GA_z} \right) F_z \tag{2.30}$$

The twist of the beam has been computed using Equation (2.31).

$$\phi = \frac{M_t \xi_k}{GJ} \tag{2.31}$$

2.4. Finite volume beam model

In the thesis, a numerical beam model has been prepared in the MBDyn software to validate the real-time algorithm using strains from the simulated sensors. MBDyn is free and open-source general-purpose Multibody Dynamics analysis software developed at the Aerospace Science and Technology Department of Politecnico di Milano [8].

The software exploits a finite volume beam model [6, 10] called `beam3` element in the syntax. The model of the beam element is presented in Figure 2.3. The beam element is composed of three nodes and two evaluation points. Considering a non-dimensional abscissa s that maps ξ along the beam element between two ends for $-1 \leq s \leq 1$, the nodes are located at points $s = -1, 0, 1$, and two (Gaussian) evaluation points are located at $s = \pm 1/\sqrt{3}$. In the considered model, the beam is split into three parts, each associated with one of the three nodes, with two cuts performed at the locations of the evaluation points. The distributed loads acting on each segment are integrated and applied to the corresponding node, as well as internal forces and moments at the cuts [18].

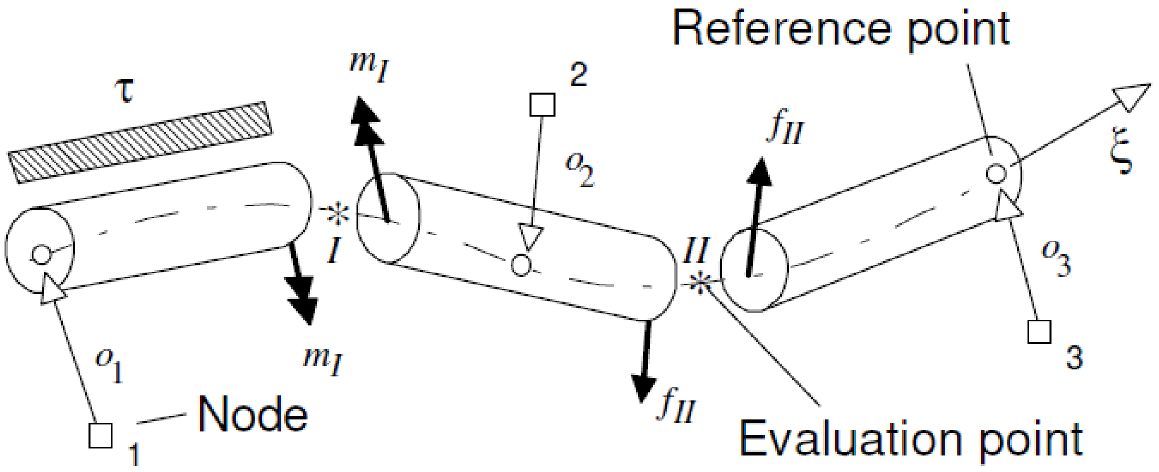


Figure 2.3: Finite volume beam model [10]

Internal forces at the boundaries of the element $\boldsymbol{f}(0), \boldsymbol{f}(l), \boldsymbol{m}(0)$, and $\boldsymbol{m}(l)$ cancel if the beam is connected with another element or are 0 in case of free end. Internal forces at the evaluation points are expressed as functions of the generalized linear and angular strains through appropriate constitutive properties. The strains are obtained by a parabolic interpolation, based on the kinematics of the nodes. It is worth mentioning that the forces and moments originating from the elements are truly energetically conjugated with the virtual displacement and rotation of the nodes to which they are applied; as a result, they have clear physical meaning [18].

In the models of the tested beams, the measuring stations are located at the position of the node that connects two adjacent beam elements. The strains measured by the sensors are computed using the algorithm that is presented below.

Initially the strains computed by the software at the evaluation point must be recomputed to simulate the axial strains and shear strains measured by a sensor k at the surface of the beam. Equation (2.32) is used to recompute the axial strain from the evaluation point to the axial strain at the sensor, while Equation (2.33) to recompute the shear strain from the evaluation point to the shear strain at the sensor. It is worth mentioning that coefficients that multiply axial strain, shear strains and curvatures have been validated only for the case of uniform beam with circular or rectangular cross-section; in the case of nonuniform beam with more complicated geometry of the cross-section they are different. This part of the algorithm is performed inside the MBDyn models; the recomputed strains are subsequently sent using an output stream connector to the Simulink model and to a text file. Each strain value is associated to a different output channel of the output stream.

$$\varepsilon_{\xi\xi_{kI}} = \varepsilon_x + \kappa_y \zeta_k - \kappa_z \eta_k \tag{2.32}$$

$$\varepsilon_{\xi t_{kI}} = \kappa_x R_k + \gamma_z \frac{\eta_k}{R_k} - \gamma_y \frac{\zeta_k}{R_k} \tag{2.33}$$

In the next step, the axial strain and shear strain at the position of the node that coincides with the measuring station, are linearly extrapolated using the strains calculated at the two evaluation points for sensor k. This operation is performed for the two adjacent beams n and $n+1$ separately. It is performed entirely by the Matlab script.

In the last step, the axial strain and shear strain of sensor k are computed using respectively Equation (2.34) and Equation (2.35).

$$\varepsilon_{\xi\xi_k} = \frac{\varepsilon_{\xi\xi_{nEX}} + \varepsilon_{\xi\xi_{n+1EX}}}{2} \tag{2.34}$$

$$\varepsilon_{\xi t_k} = \frac{\varepsilon_{\xi t_{nEX}} + \varepsilon_{\xi t_{n+1EX}}}{2} \tag{2.35}$$

Figure 2.4 presents in a graphical way the computation of the axial strain at sensor k placed on a measuring station that is located between beams n and $n+1$.

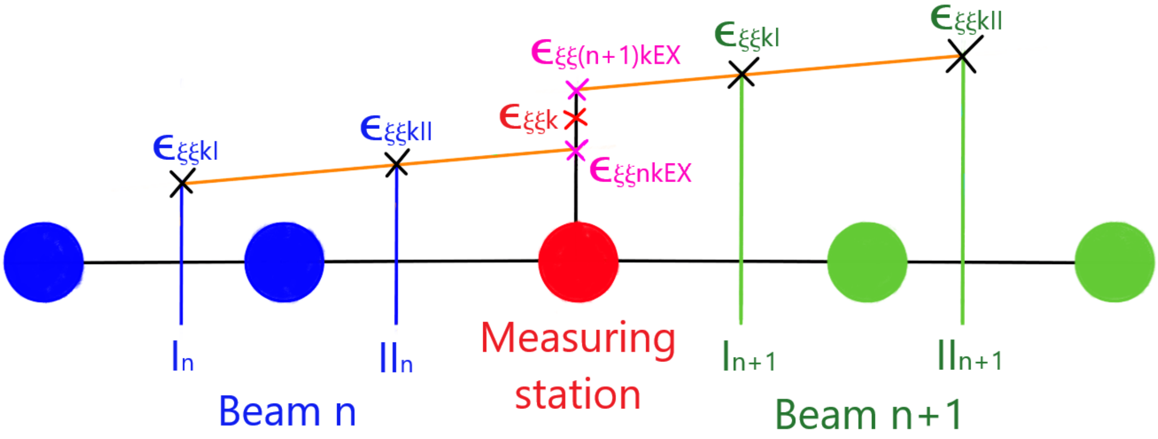


Figure 2.4: Axial strain computation scheme

2.5. Numerical modal shape function determination algorithm

This thesis also covers the calculation of the shape functions of a blade basied on the results of a modal analysis of a detailed 3D finite element model. The following section presents the algorithm used for the determination of the modal displacement at the location of the measuring stations. The algorithm exploits the least-squares method just like the algorithm used for the shape reconstruction of a beam shown in 2.2.3 [15].

Figure 2.5 presents a measuring station s. For each station, three sections are defined. Section j is located in the position of the sensor; section $j-1$ and $j+1$ are respectively located slightly before and after station j. Each station has its predefined reference point located more or less at the position of the elastic axis of the blade and in the middle of the section. Each section also is associated to a set of i nodes that belong to the FEM model. They do not need to be exactly at the same spanwise station as the reference point; they must be in the vicinity of this point in some slice of the blade.

For each section, the following computations are performed. In the beginning, the difference in position between the reference point $\boldsymbol{P}_{j_{\text{ref}}}$ and node position $\boldsymbol{P}_{ij}$ is defined as it is shown in Equation (2.36).

$$\boldsymbol{\delta}_{ij} = \boldsymbol{P}_{ij} - \boldsymbol{P}_{j_{\text{ref}}} \tag{2.36}$$

$$\boldsymbol{P}_{ij} = \begin{Bmatrix} x_i \\ y_i \\ z_i \end{Bmatrix}_j \tag{2.37}$$

$$\boldsymbol{\delta}_{ij} = \begin{Bmatrix} \delta_{xi} \\ \delta_{yi} \\ \delta_{zi} \end{Bmatrix}_j \tag{2.38}$$

In the next step, the $\boldsymbol{Z}$ matrix is computed. Every three rows of the $\boldsymbol{Z}$ matrix have the form that is presented in Equation (2.39). The matrix $\boldsymbol{Z}$ is composed of $3i$ rows and is computed for each section in the measuring station.

$$\boldsymbol{Z}_j = \Big[\boldsymbol{I}_{3\times3}, -\boldsymbol{\delta}_{ij}\times\Big]_j \tag{2.39}$$

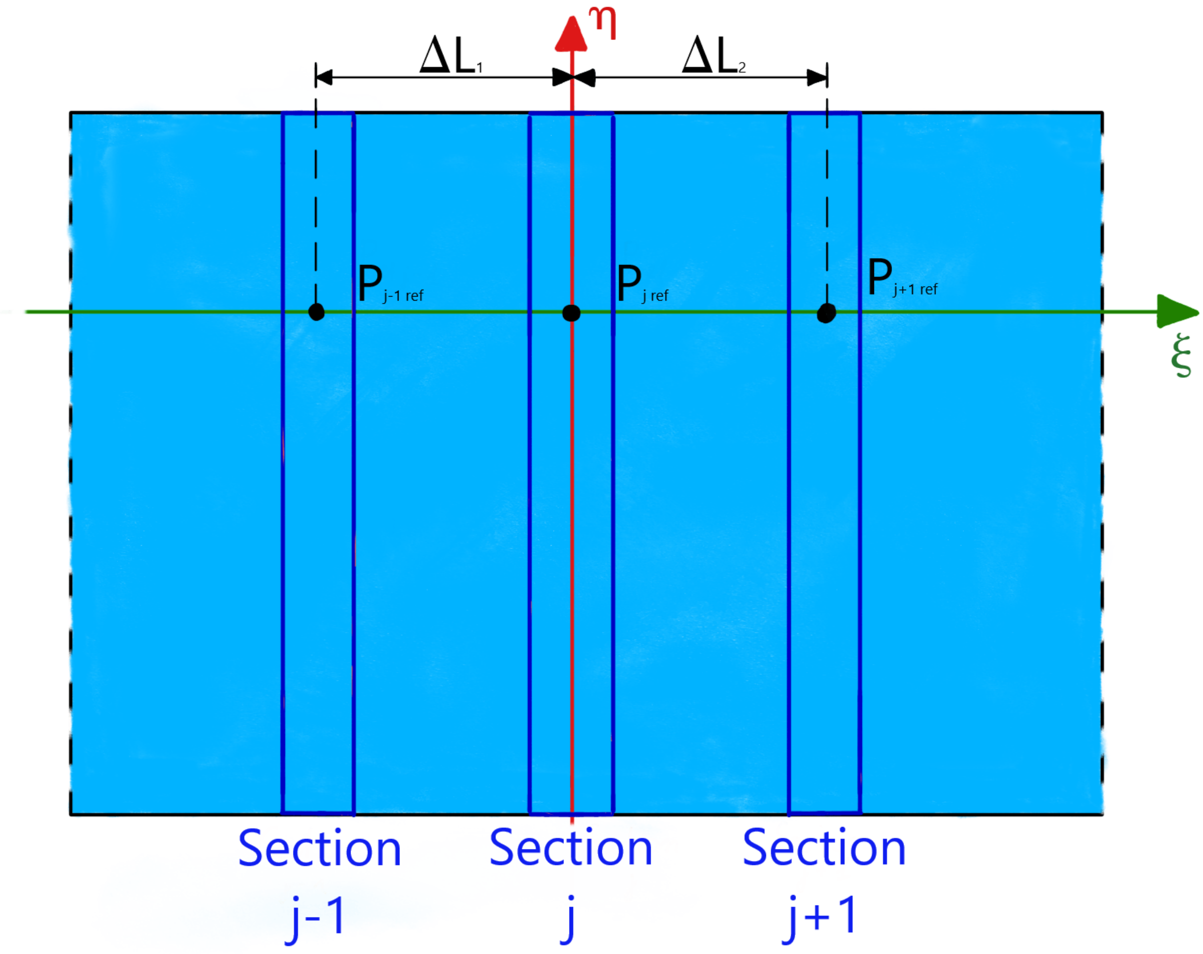


Figure 2.5: Measuring station

where $\boldsymbol{\delta}_{ij}\times$ is a vector recomputed into the matrix form presented in Equation (2.40), to perform the vector cross-product.

$$\boldsymbol{\delta}_{ij}\times = \begin{bmatrix} 0 & -\delta_{zi} & \delta_{yi} \\ \delta_{zi} & 0 & -\delta_{xi} \\ -\delta_{yi} & \delta_{xi} & 0 \end{bmatrix}_j \tag{2.40}$$

The matrix $\boldsymbol{Z}$ is computed once for all sections, as it only depends on the geometry of the model. In the next step, that is presented in Equation (2.41), the modal displacement of the reference point is computed using the mth vector of modal displacements of all the nodes, $\boldsymbol{\Phi}_j^m$.

$$\boldsymbol{d}_j^m = \boldsymbol{Z}_j^+ \boldsymbol{\Phi}_j^m \tag{2.41}$$

Vector $\boldsymbol{\Phi}_j^m$ has the form presented in Equation (2.42).

$$\boldsymbol{\Phi}_j^m = \left\{ \begin{matrix} \vdots \\ \Delta x_i \\ \Delta y_i \\ \Delta z_i \\ \Delta x_{i+1} \\ \Delta y_{i+1} \\ \Delta z_{i+1} \\ \vdots \end{matrix} \right\}_j^m \tag{2.42}$$

The modal displacement and rotation of the reference point are computed for all three sections in the measuring station.

In the next step, the modal displacement of the sections $j-1$ and $j+1$ are used to compute the first- and second-order space derivatives of the modal displacement at the reference point located at the section j, which are required in the definitions of the strains. For this purpose, the formulas of Equation (2.43) and Equation (2.44) are used.

$$\boldsymbol{d}_j'^m = \frac{\Delta L_1^2 \boldsymbol{d}_{j+1}^m + (\Delta L_2^2 - \Delta L_1^2)\boldsymbol{d}_j^m - \Delta L_2^2 \boldsymbol{d}_{j-1}^m}{\Delta L_1 \Delta L_2 (\Delta L_1 + \Delta L_2)} \tag{2.43}$$

$$\boldsymbol{d}_j''^m = \frac{2\Delta L_1 \boldsymbol{d}_{j+1}^m - 2(\Delta L_2^2 + \Delta L_1^2)\boldsymbol{d}_j^m + 2\Delta L_2 \boldsymbol{d}_{j-1}^m}{\Delta L_1 \Delta L_2 (\Delta L_1 + \Delta L_2)} \tag{2.44}$$

3 | Shape functions

This chapter of the thesis presents the shape functions obtained for the purpose of this thesis' activities. The analytical shape functions that are presented in section 3.1 have been used in the Simulink real-time model as well as in the Matlab script. The numerical shape functions that are shown in section 3.2 have been obtained using the finite element model of the blade from the laboratory.

3.1. Analytical shape functions

The analytical shape functions result from the analytical solution of the continuum dynamics of a uniform beam. They can be obtained using pencil & paper (see for example [16]). They are derived under the assumption that the beam is uniform and clamped at the root.

3.1.1. Analytical torsional shape functions

For the purpose of the shape reconstruction algorithm, it was necessary to derive the torsional shape functions and their first derivatives. The analytical formulas of the shape functions and their first derivative are presented respectively in Equation (3.1) and Equation (3.2).

$$n_t = A_t \sin\left((2n-1)\frac{\pi}{2}\frac{\xi}{l}\right) \tag{3.1}$$

$$n_t' = A_t(2n-1)\frac{\pi}{2l}\cos\left((2n-1)\frac{\pi}{2}\frac{\xi}{l}\right) \tag{3.2}$$

The coefficients of the shape function A_t and n depend directly on the shape function and on the desired normalization criterion; for shape function of mode number 1, the coefficients are equal to: $A_t = 1$, $n = 1$ for unit max rotation.

Figure 3.1 and Figure 3.2 present respectively the torsional shape functions and their derivatives, for the modes from 1 to 5 assuming that the length of the beam is equal to the length of the blade from the laboratory. The markers on the plot indicate the position of the measuring stations along the blade. Shape functions shown in Figure 3.1 are normalized for unit rotation at the blade tip.

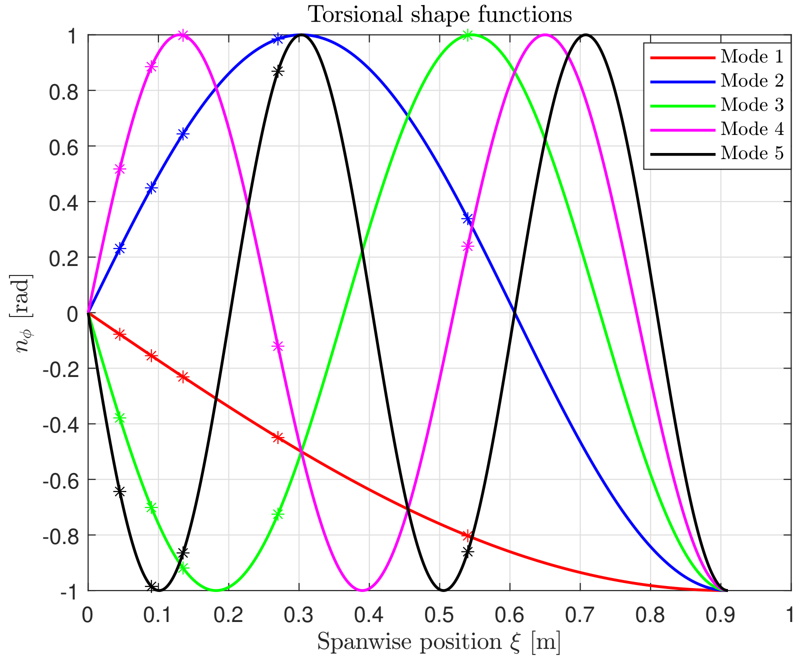


Figure 3.1: Torsional shape functions

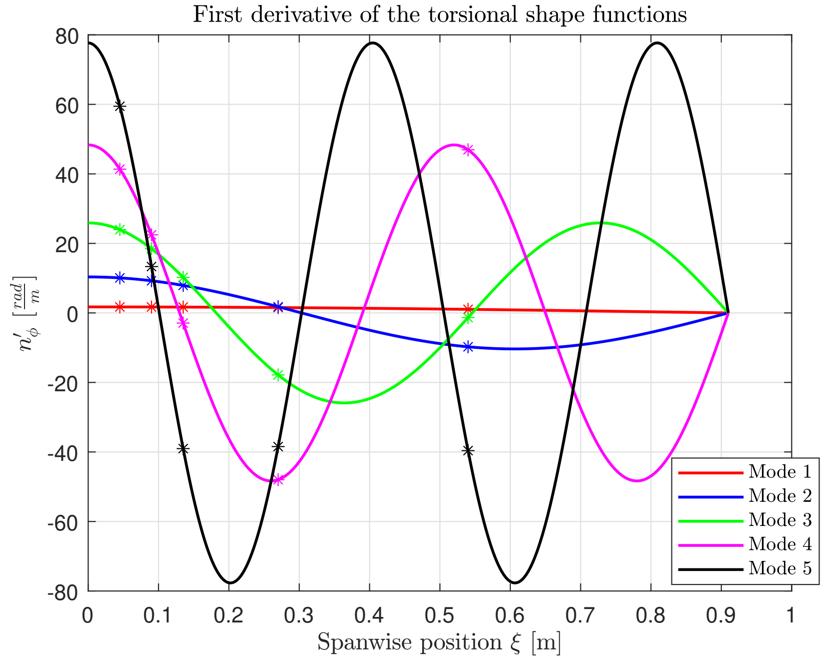


Figure 3.2: First derivatives of the torsional shape functions

3.1.2. Bending shape functions

In the case of the bending shape functions, the algorithm requires the first and second space derivatives. The analytical formulas of the shape functions and their derivatives are shown in Equation (3.3), Equation (3.4), and Equation (3.5).

$$n_b = A_{1b}\cos\left(k\frac{\xi}{l}\right) + A_{2b}\sin\left(k\frac{\xi}{l}\right) + A_{3b}\sinh\left(k\frac{\xi}{l}\right) + A_{4b}\cosh\left(k\frac{\xi}{l}\right) \tag{3.3}$$

$$n_b' = \frac{k}{l}\Bigg(- A_{1b}\sin\left(k\frac{\xi}{l}\right) + A_{2b}\cos\left(k\frac{\xi}{l}\right) + A_{3b}\cosh\left(k\frac{\xi}{l}\right) + A_{4b}\sinh\left(k\frac{\xi}{l}\right)\Bigg) \tag{3.4}$$

$$n_b'' = \left(\frac{k}{l}\right)^2\Bigg(- A_{1b}\cos\left(k\frac{\xi}{l}\right) - A_{2b}\sin\left(k\frac{\xi}{l}\right) + A_{3b}\sinh\left(k\frac{\xi}{l}\right) + A_{4b}\cosh\left(k\frac{\xi}{l}\right)\Bigg) \tag{3.5}$$

The coefficients A_{1b}, A_{2b}, A_{3b}, and A_{4b} are computed by finding the eigenvector of the matrix that corresponds to the null eigenvalue of the eigenproblem resulting from the "cantilever" boundary conditions; they are later normalized with respect to the largest coefficient. The values of the coefficients k are determined by finding the roots of the determinant of the characteristic matrix of the problem, which is a transcendental function of k.

The values of all the coefficients for the modes from 1 to 5 are shown in Table 3.1.

Mode no.	**k**	$\boldsymbol{A_{1b}}$	$\boldsymbol{A_{2b}}$	$\boldsymbol{A_{3b}}$	$\boldsymbol{A_{4b}}$
1	1.875	1.000	-0.734	0.734	-1.000
2	4.694	0.981	-1.000	-1.000	-0.981
3	7.855	1.000	-1.000	1.000	-1.000
4	10.995	-1.000	1.000	-1.000	1.000
5	14.137	1.000	-1.000	1.000	-1.000

Table 3.1: Coefficients of the bending shape functions for modes 1-5

Figure 3.3, Figure 3.4, and Figure 3.5 present respectively the shape functions, and their first and second derivatives. The bending shape functions are presented along the spanwise length of the blade and the markers indicates the location of the measuring stations along the blade.

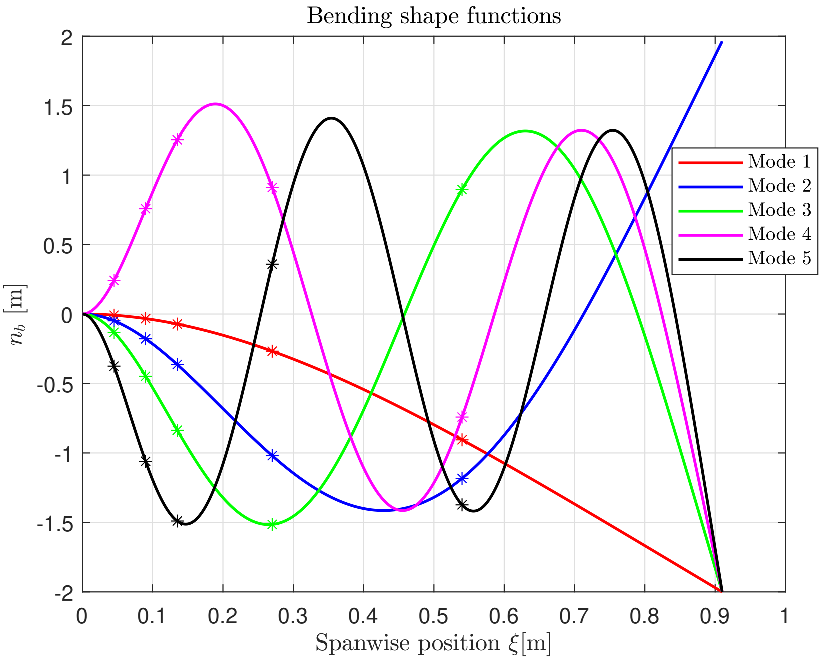


Figure 3.3: Bending shape functions

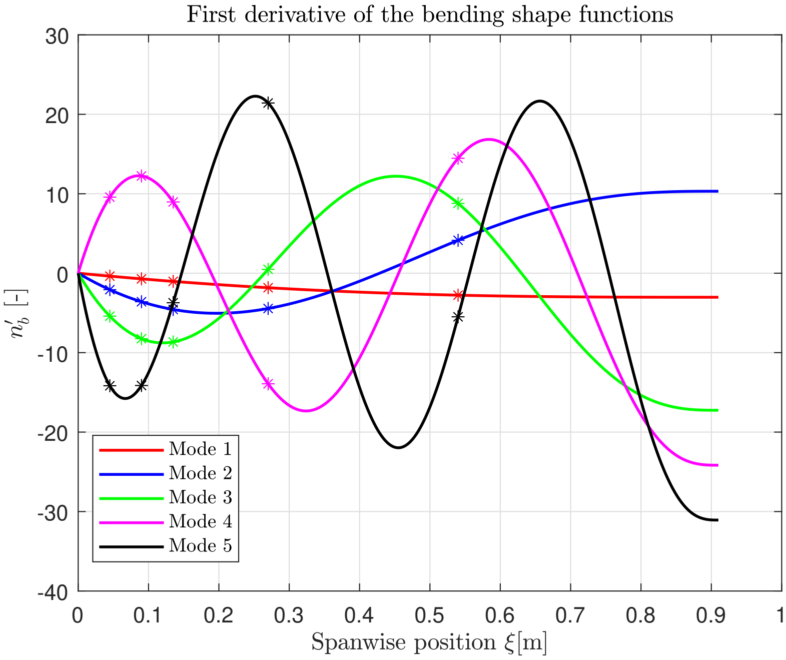


Figure 3.4: First derivatives of the bending shape functions

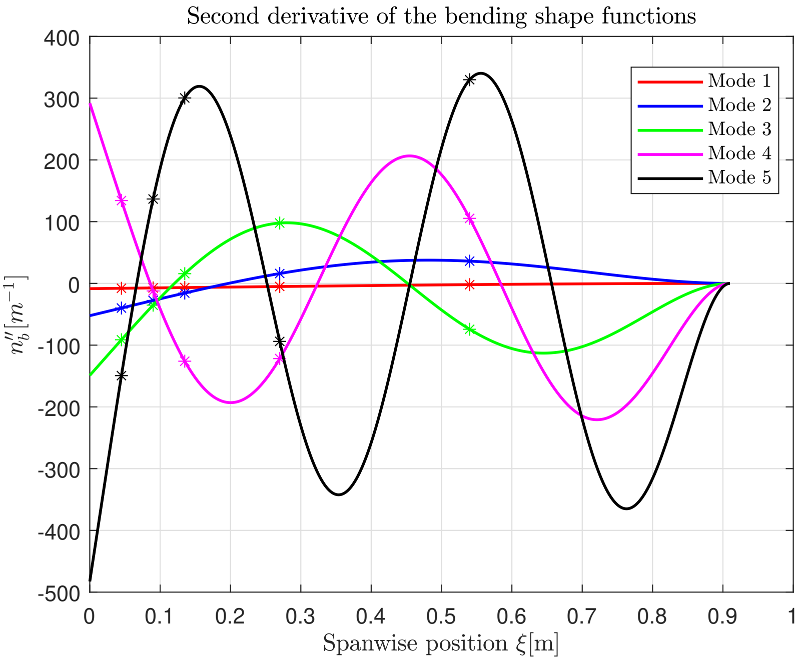


Figure 3.5: Second derivatives of the bending shape functions

3.2. Numerical shape functions

Numerical shape functions have been obtained by post-processing the results of the modal analysis, performed using the FEMAP software, using the finite element model of the blade. The model of the blade has been developed by Roberta Cumbo and shared by courtesy of Siemens Digital Industries Software, Leuven, Belgium.

3.2.1. FEM model of the blade

The blade FEM model reflects directly the structure of the real blade model from the laboratory taking into account its highly nonuniform structure. The NACA23012 airfoil is used, which is typical of many helicopter blades. The airfoil is characterized by a high maximum lift and a low profile drag, which results in a high value of the speed-range index. In addition, it has low pitching moment coefficient [13]. The airfoil is presented in Figure 3.6. The chord of the blade is 72.5 mm, while the length of the blade is 910 mm. Five measuring stations are located along the blade span; their position is reported in Table 3.2.

Measuring station no.	**Position along spanwise axis ξ[m]**
1	0.045
2	0.090
3	0.135
4	0.270
5	0.540

Table 3.2: Measuring stations along the blade

The blade is made of different composite materials. The skin of the blade is made of a glass fibre in an epoxy matrix, two plies are laminated under an angle of 45^o between each other. Two stringers are placed along the spanwise axis of the blade, both are made of the unidirectional glass fiber in the epoxy matrix. They are laminated from 4 plies under an angle of 0^o, the fibres are oriented along the spanwise axis of the blade. The last element of the blade is a foam that is placed inside the blade, it is made of Rohacell 51 WF. All of those elements have been modelled to obtain the best possible results.

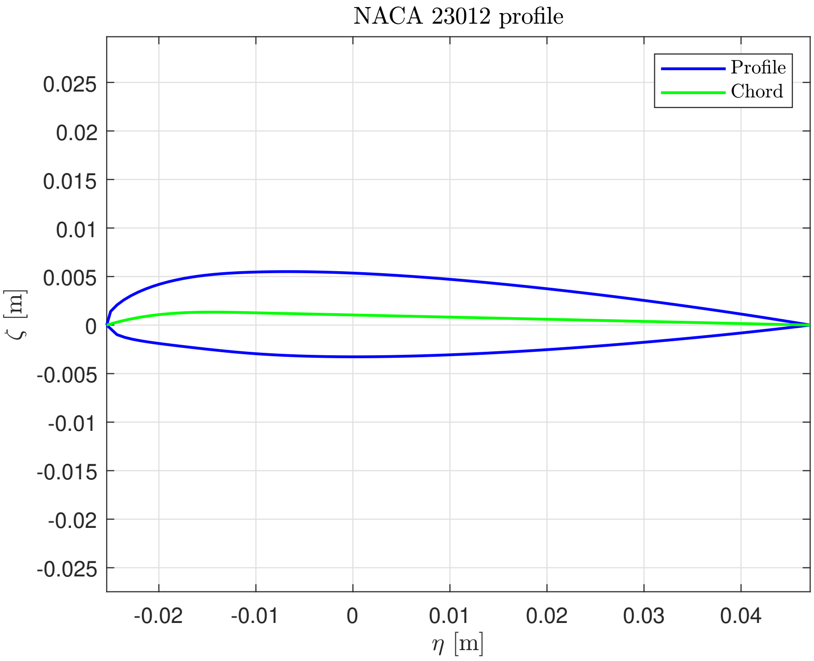


Figure 3.6: NACA 23012 profile

The model of the blade is shown in Figure 3.7. The cross-section varies with the spanwise position along the axis; at the root it is rectangular, and after the transition region it takes the shape of the airfoil. For the purpose of modal analysis, the blade is constrained at the root using a clamp that fixes the translation along all axes and rotation about all axes. The skin and stringers of the blade have been modelled using shell elements (`quad4`), while the foam has been modelled using solid elements (`brick8`). The connection between the different parts of the blade is done using rigid elements which connect the skin, the foam, and the stringers.

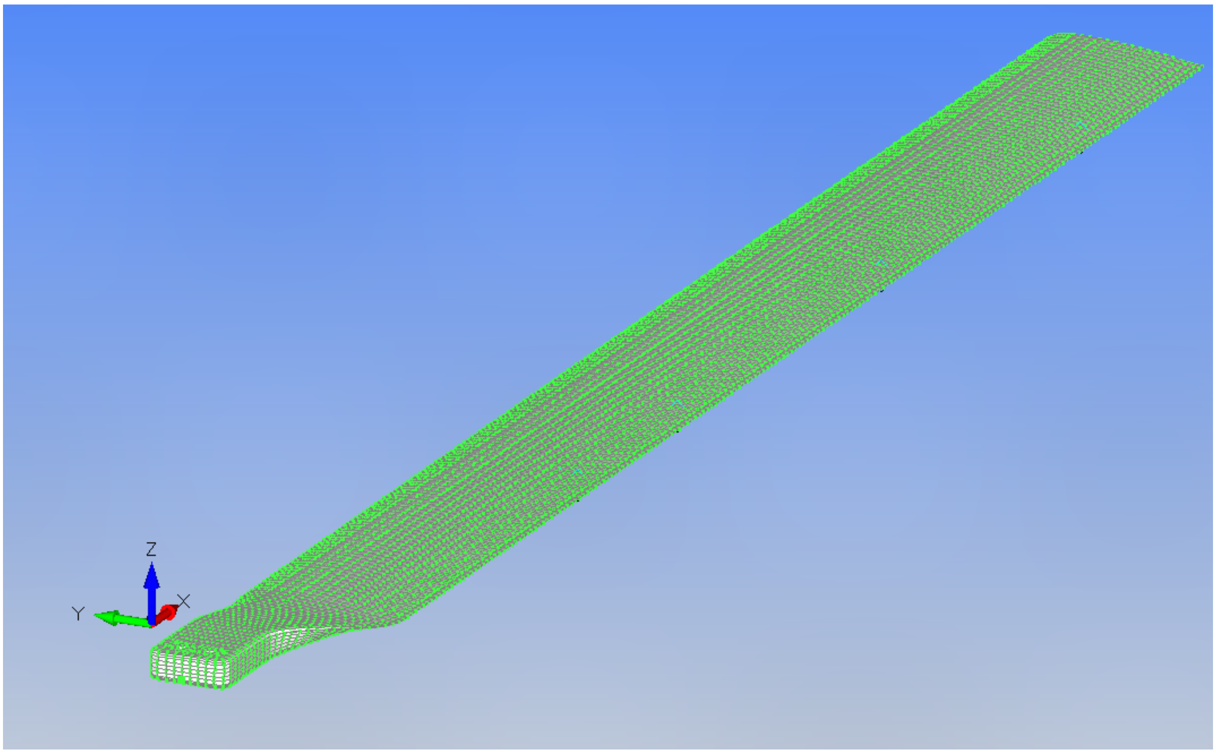


Figure 3.7: Blade FEM model

3.2.2. Matlab script for the numerical shape functions calculations

The modal shapes have been reconstructed using the algorithm presented in section 2.5, applied to the results of the modal analysis. Initially, the data of the model in the reference configuration has been loaded into the script. Using those data, 20 stations have been defined along the blade, each station consisting of 3 sections to which a subset of the nodes around the reference points have been assigned. In the next step, a simulation was performed using Nastran solver built into FEMAP. The modal method has been set to Lanczos, and it was decided that the simulation returns the first 40 modes of the blade. The output has been set to the deformation of all the nodes and directed to a `.f06` file, which has been later cleaned from the unnecessary data and loaded into the Matlab script.

In the next step, the data has been divided into the displacements assigned to particular modes and for each mode only the data related to the nodes that belongs to the sections on the measuring stations have been saved. At this point, the modal displacements of the reference points for all 40 modes have been computed using the algorithm that is presented in the section 2.5. In the last step, the shape functions in the flapwise and the chordwise directions, as well as the torsional shape functions, have been identified by inspection of the modal deformation vector. Later the results have been also validated with the visual representation of the modes given in the FEMAP post-processor.

Figure 3.8 visually presents the algorithm. The blue blocks contain the data that comes from the FEM model and FEM analysis, while the data in the yellow block is stated by the user. The green blocks contain the actions executed by the script.

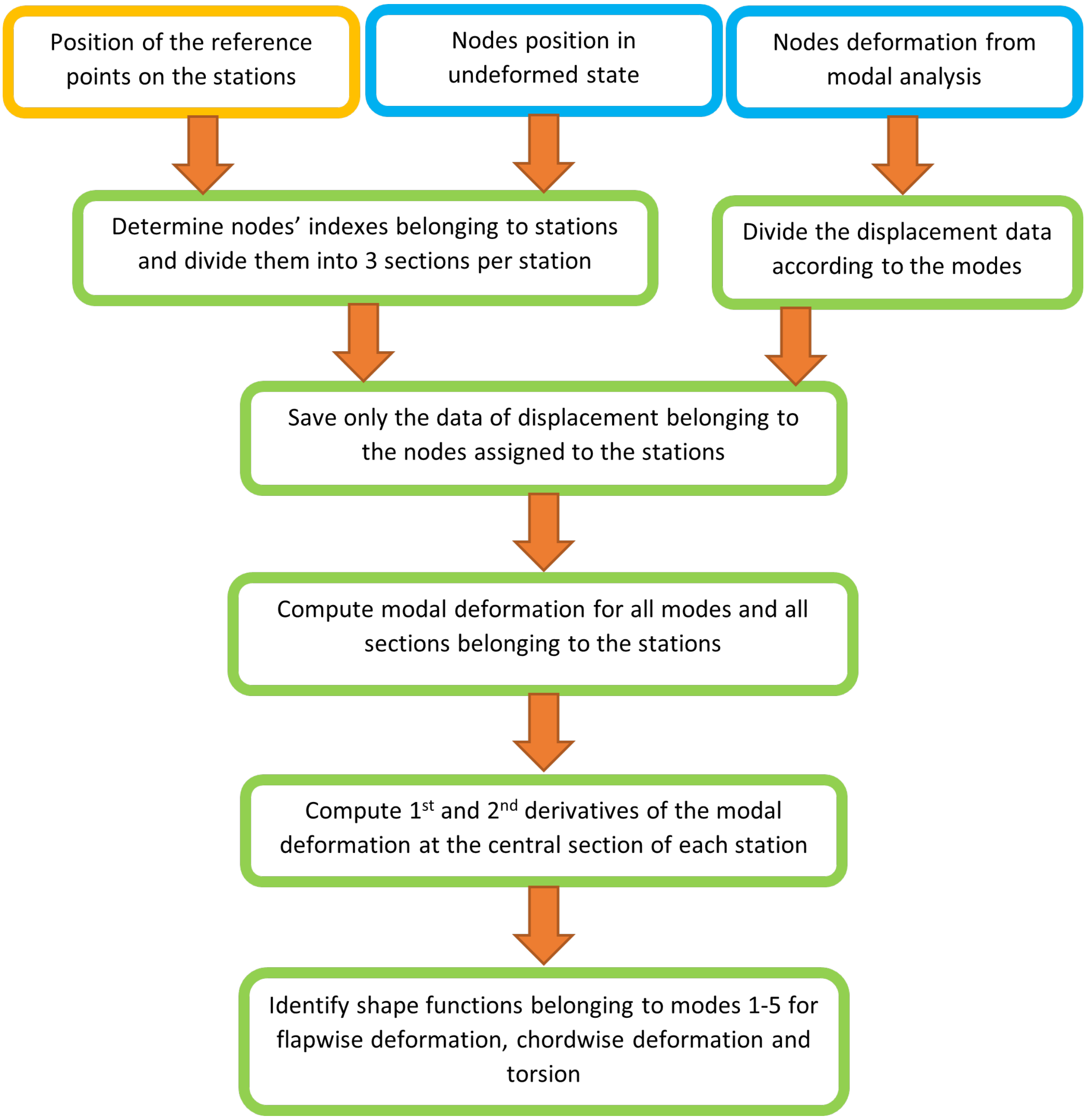

Position of the reference points on the stations
Nodes position in undeformed state
Nodes deformation from modal analysis
Determine nodes' indexes belonging to stations and divide them into 3 sections per station
Divide the displacement data according to the modes
Save only the data of displacement belonging to the nodes assigned to the stations
Compute modal deformation for all modes and all sections belonging to the stations
Compute 1st and 2nd derivatives of the modal deformation at the central section of each station
Identify shape functions belonging to modes 1-5 for flapwise deformation, chordwise deformation and torsion

Figure 3.8: Algorithm of the numerical shape functions calculations

3.2.3. Results of the numerical shape functions calculations

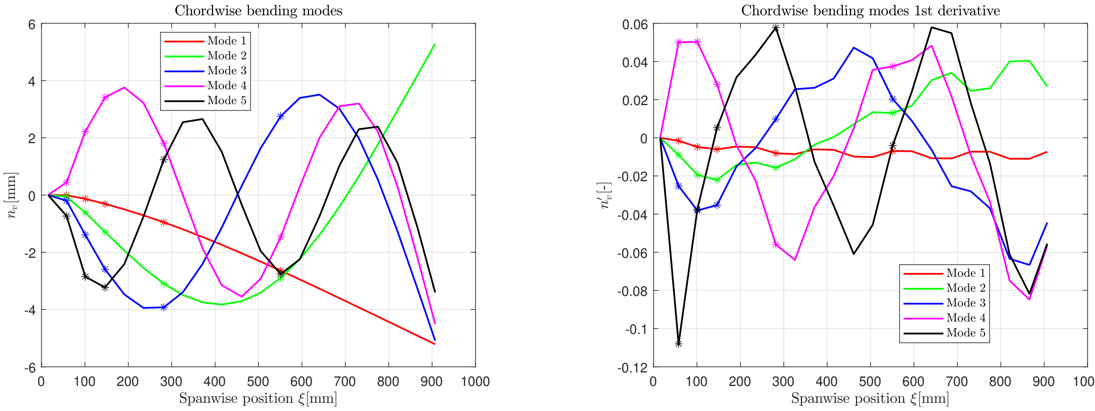
This section presents the plots of the numerical shape functions obtained from the modal analysis of the blade's FEM model. The shape functions have been plotted for 20 points located along the spanwise axis of the blade. The markers on the plots indicate the approximate position of the measuring stations.

Table 3.3 presents the chordwise, the flapwise, and the torsion shape functions for modes 1-5 together with their frequency. The flapwise modes have in general low frequency, especially in comparison to the torsional modes. The chordwise modes start at low frequency but higher modes have a very high frequency. For the implementation of the modes in the shape reconstruction algorithm, the modes with very high frequency do not have to be included.

Mode type and number	Frequency [Hz]
Chordwise 1	19.91
Chordwise 2	141.38
Chordwise 3	404.81
Chordwise 4	786.24
Chordwise 5	1244.38
Flapwise 1	4.67
Flapwise 2	29.38
Flapwise 3	81.48
Flapwise 4	158.69
Flapwise 5	255.99
Torsion 1	115.03
Torsion 2	351.09
Torsion 3	576.61
Torsion 4	600.84
Torsion 5	814.21

Table 3.3: Numerical shape functions for modes 1-5

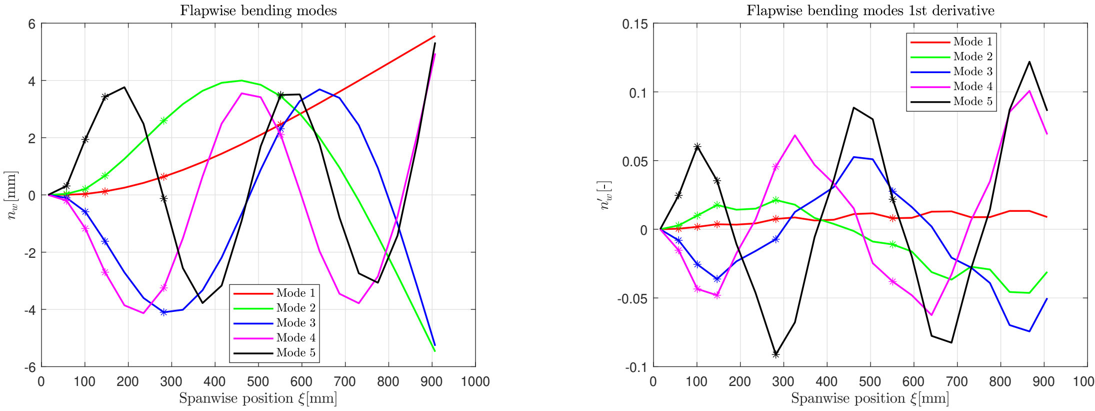
Figure 3.9, Figure 3.10, and Figure 3.11 present respectively the chordwise, flapwise and torsional shape functions for the modes 1 to 5, together with their first derivatives. The bending shape functions are reconstructed with a very good quality, while the torsional shape functions are not very well reconstructed in the area close to the root of the blade, where the cross-section changes from rectangular to that of the airfoil. This issue will be explained in the further part of this subsection. A better quality of the reconstructed shapes would be provided if more stations along the blade would be considered; however, it was decided that 20 points is sufficient and reasonable from the computational efficiency point of view. It is worth mentioning that due to the fact that each measuring station must have 3 sections, the last measuring station is not located at the tip of the blade, but slightly before.



(a) Chordwise shape functions

(b) First derivatives of the chordwise shape functions

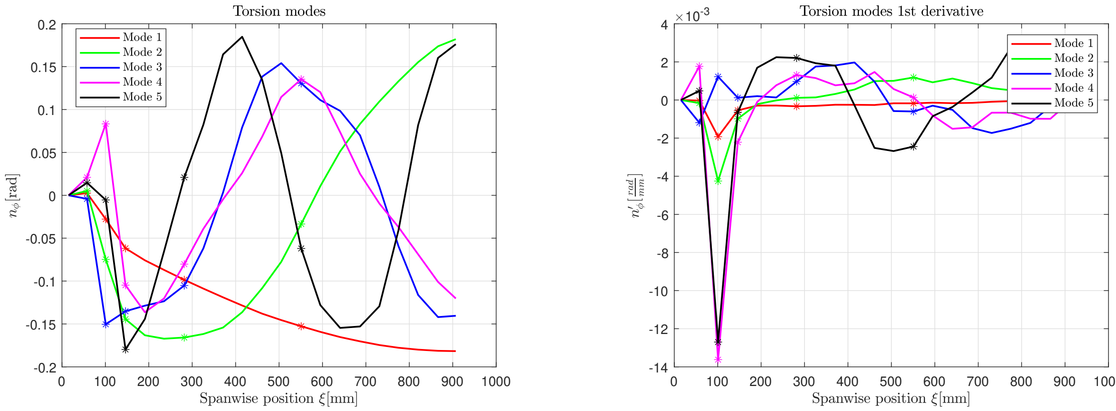
Figure 3.9: Chordwise shape functions and their first derivatives



(a) Flapwise shape functions

(b) First derivatives of the flapwise shape functions

Figure 3.10: Flapwise shape functions and their first derivatives

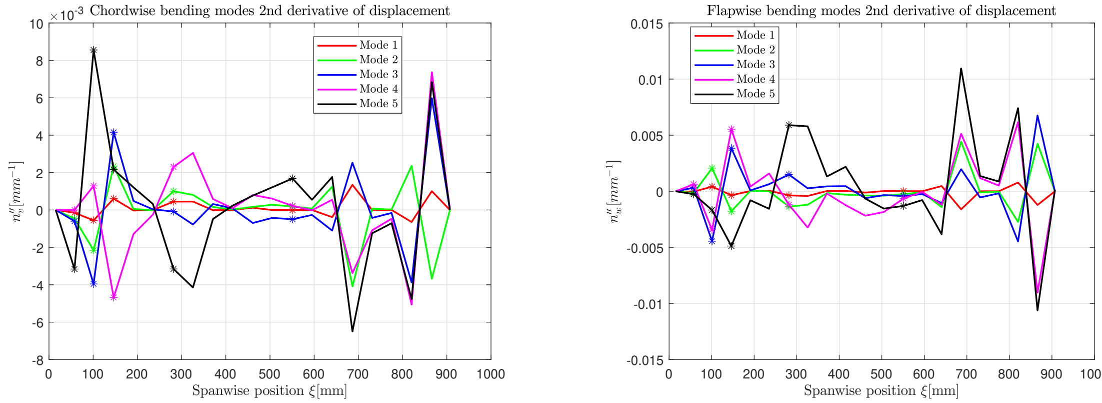


(a) Torsion shape functions

(b) First derivatives of the torsion shape functions

Figure 3.11: Torsion shape functions and their first derivatives

The second derivatives of the shape functions are necessary only in the case of bending shape functions, when the Euler-Bernoulli model is considered; they are presented in Figure 3.12.

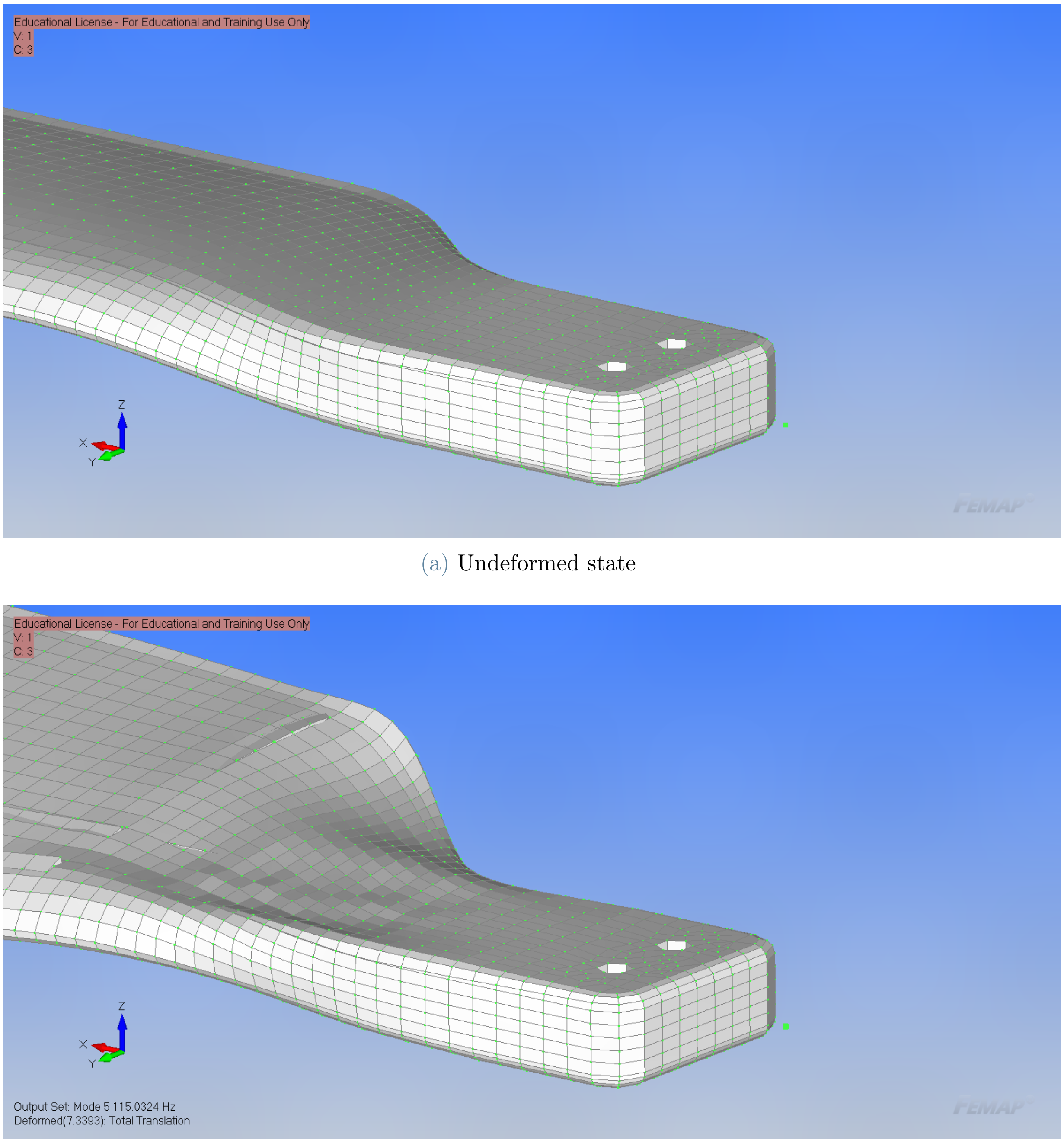


(a) Second derivatives of the chordwise shape functions

(b) Second derivatives of the flapwise shape functions

Figure 3.12: Second derivatives of the bending shape functions

Figure 3.13 presents the root area of the blade in the case of torsion mode no. 1. The blade is shown in an undeformed and a deformed state (the scale of the deformation is increased to 2 for better visualization) respectively in Figure 3.13a and Figure 3.13b. In the area of the shape transition from rectangular to that of the airfoil, the elements are additionally distorted due to the change in shape of the cross-section; their deformation is not as regular as expected. This is the possible explanation of the irregularity of the torsional shape functions close to the root, that is presented in Figure 3.12a.


Educational License - For Educational and Training Use Only
V: 1
C: 3
Z
X
Y
FEMAP
Educational License - For Educational and Training Use Only
V: 1
C: 3
Z
X
Y
Output Set: Mode 5 115.0324 Hz
Deformed(7.3393): Total Translation
FEMAP

(a) Undeformed state

(b) Deformed state

Figure 3.13: Deformation of the root area in the case of torsion mode no. 1

4 | Matlab script and Simulink model

This chapter presents the implementation of the shape reconstruction algorithm into the Matlab & Simulink environment.

The algorithm presented in section 2.2 has been implemented initially into the Matlab script, then it was validated using strains from the analytical beam model, and strains from the MBDyn model. The results given by the algorithm have been compared for validation with the analytical results and nodal deformation results given by the MBDyn models of the beams.

The structure of the algorithm is shown in Figure 4.1. It is divided into 4 main blocks. The data block contains all the information about beam, sensors, and measuring stations. This block also includes the script that computes the position of the sensors on the beam cross-section. It is not connected to any blocks in Figure 4.1 because the information that it stores is available for all the blocks of the code. The algorithm works in such a way that if the data of the reconstructed beam is changed (cross-section, measuring stations, etc.), only the data block must be changed, and the simulation must be restarted. The measurements block receives input strains from the MBDyn models, recomputes them as it was presented in section 2.4, and divides them into axial and shear strains. The torsion problem is solved first; its main output is the twist angle, which is used in the bending problem, and is also plotted at the end. In the last step, the bending problem is solved with two outputs, the chordwise and flapwise displacement components. In the presented scheme the calculation operations that are inside yellow blocks are performed only once, while the other blocks are recomputed at every iteration.

The Matlab script and the Simulink model have a very similar structure, the only difference is in the input and output to the function. The Matlab script operates on a single set of strains that are analytical and come from a text file generated by the MBDyn model; the output from the Matlab script is sent to the text file that is later used for plotting the results. The Simulink model works in co-simulation with the MBDyn models; the input are the strains from MBDyn and the output are the plots of the deformation in time. In the Simulink model, additional input contains the deformation of the nodes at the measuring stations and the displacement of the end node of the beam, which is used in the plots to present the results of the beam shape reconstruction. Figure 4.2 presents the Simulink model. The block `sfun_mbdyn_start` is used to set up a co-simulation between MBDyn and Simulink. The strains are delivered through the `sfun_mbdyn_com_read` block that has as many outputs as channels (in the presented case 103 outputs are used, 100 related to the strains, and 3 related to the end node displacement). Later, the strains are connected to the `Mux` element that combines them into a single vector that is sent to the `Main_script` block which contains the whole shape reconstruction algorithm. The block has 3 main outputs: the twist angle and the chordwise and flapwise displacement components. In the presented case there are also 2 additional outputs that show the axial and shear strains used for the calculations. The output is sent to the displays that show the final values and the scopes that generate the plots of displacement as a function of time. The scopes plot the values reconstructed using the algorithm together with the displacement of the nodes.

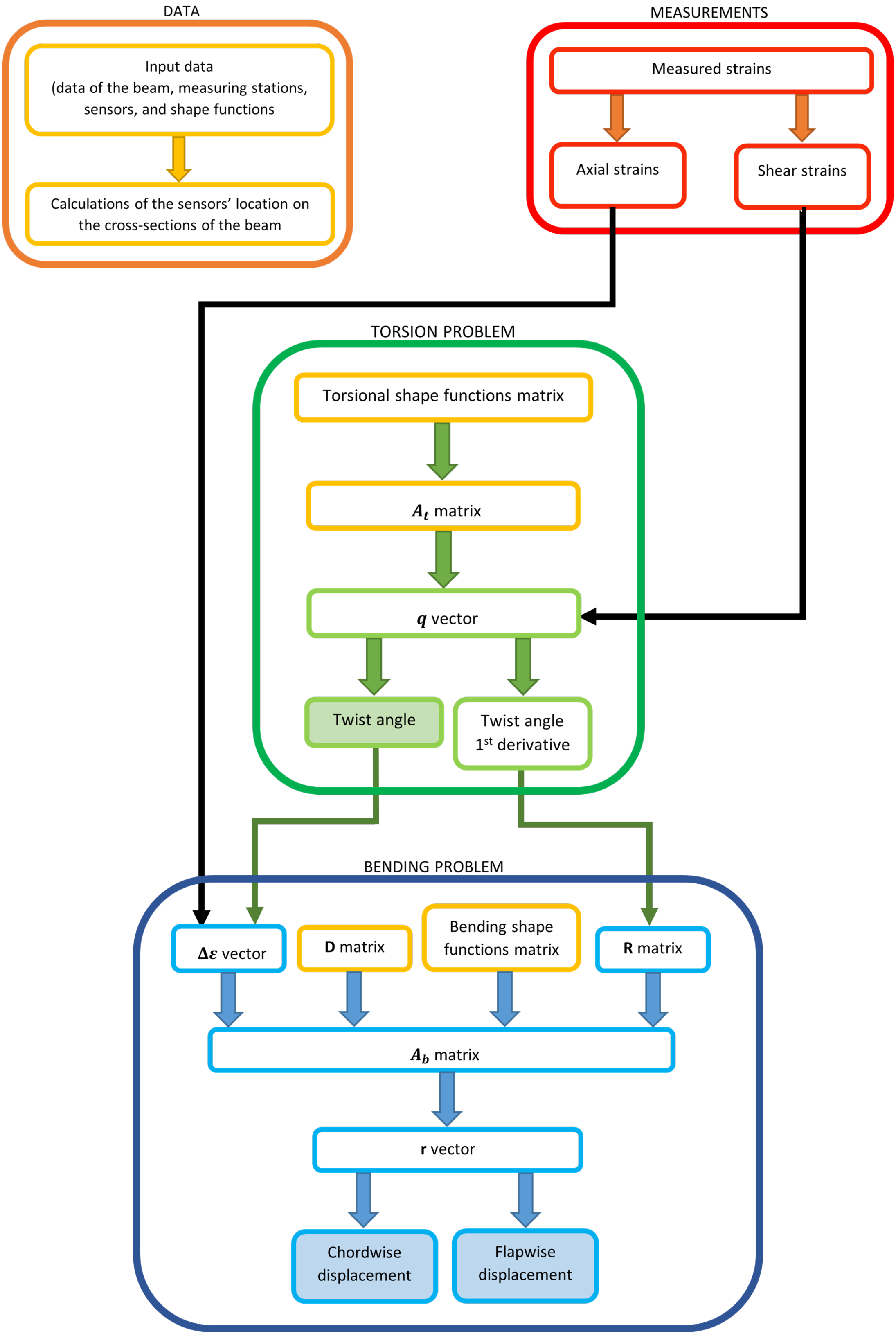

DATA
Input data
(data of the beam, measuring stations, sensors, and shape functions
Calculations of the sensors' location on the cross-sections of the beam
MEASUREMENTS
Measured strains
Axial strains
Shear strains
TORSION PROBLEM
Torsional shape functions matrix
A_t matrix
q vector
Twist angle
Twist angle 1st derivative
BENDING PROBLEM
$\Delta\varepsilon$ vector
D matrix
Bending shape functions matrix
R matrix
A_b matrix
r vector
Chordwise displacement
Flapwise displacement

Figure 4.1: Scheme of the algorithm

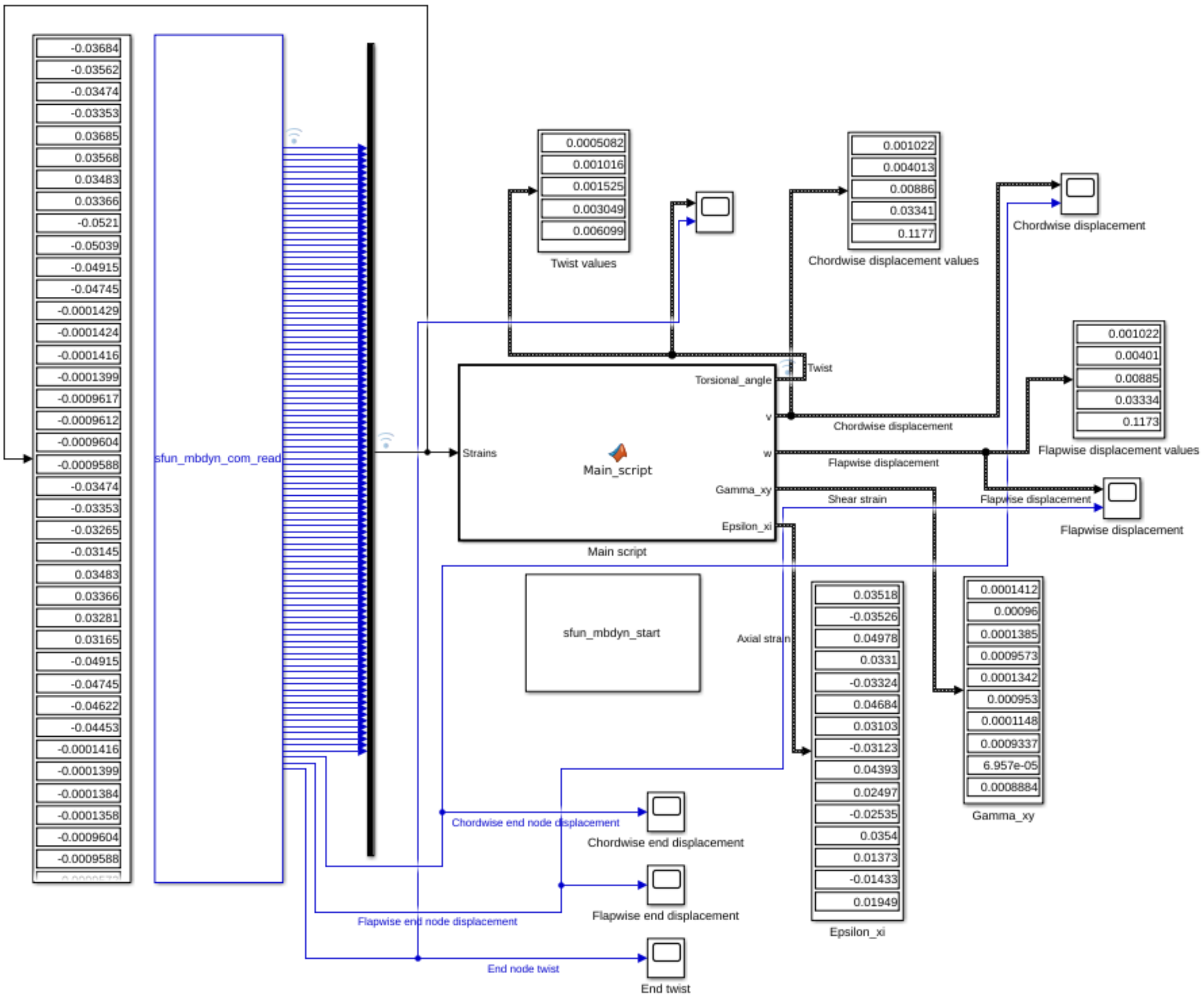


Figure 4.2: Real-time Simulink model

5 | Algorithm verification using a beam with circular cross-section

This chapter presents the verification of the algorithm shown in section 2.2, which has been implemented in a Matlab script and in a Simulink model. The validation has been performed using a beam with a circular cross-section.

5.1. Circular beam model

The beam used for verification is uniform and has a circular cross-section with a diameter equal to the chord of the blade from the laboratory, $d = 72.5$ mm, and the same length, $L = 0.91$ m. The beam used for validation does not have any built-in twist. The material is assumed to be aluminium alloy (T 7075). Five measuring stations are located along the blade, in the same locations as in the case of the original blade. The position of the measuring stations is presented in Table 3.2.

Figure 5.1 presents the plot of the beam cross-section. In the initial configuration there are three sensors placed on each measuring station; 2 strain gauges that measure the axial and shear strain, and 1 optical fibre sensor that measures only the axial strain. In the figure below two vectors are shown as well, tangent to the surface, that are required for the reconstruction of twist.

The algorithm has been validated using the analytical results as well as the deformation of the nodes taken from the MBDyn model. In the algorithm, there were used analytical strains as well as strains taken from the MBDyn models. Two MBDyn models of the beam have been created, one static and one dynamic. The former is used for the validation of the Matlab script, and the latter in the real-time co-simulation together with the Simulink model. All the models are composed of 20 `beam3` elements, whose length is adjusted such that the measuring station is always at one of the nodes. The models also

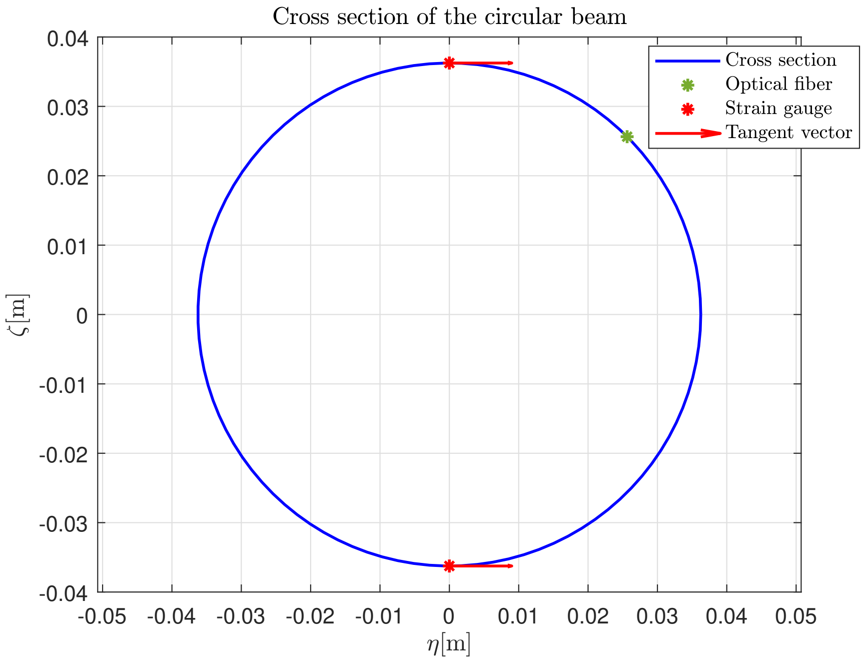


Figure 5.1: Cross-section of the circular beam with strain sensors

exploit the generation file that is created in Matlab and significantly simplifies the model development. In the Matlab script all the numerical parameters of the model, as well as references, nodes, and `beam3` elements are defined. The script generates ASCII text files which are later loaded into the main code of the model. In the main code of the model only loads and the parameters of the simulation are specified. The simulations have been performed using the `bdf` integration method with time step equal to 10^{-3} and the max iterations parameter set to 10, with the derivatives tolerance set to 10^{-5}.

In the verification process, forces in chordwise and flapwise direction, as well as torque, have been applied. All loads have been applied at the end of the beam.

5.2. Verification of the algorithm using small deformation

Small deformation of the beam is assumed to be in the range of 1%–3% of the spanwise length of the beam, whereas small rotation is in the range of 10^{-3}–10^{-2} rad. Two load cases have been defined; they are shown in Table 5.1. Load case no. 1 contains only forces along the global Z and Y axes, while load case no. 2 consists of forces and a torque. The strains have been generated using the analytical formulas presented in section 2.3 using the Euler-Bernoulli beam model. In the presented results 5 analytical modes have been used for both bending and torsional shape reconstruction.

Load case no.	F_z[N]	F_y[N]	M_t[Nm]
1	-15000	-10000	0
2	-10000	-10000	1000

Table 5.1: Load cases for verification using small deformation

5.2.1. Small deformation load case no. 1

Figure 5.2, Figure 5.3, and Figure 5.4 present the deformation of the beam subjected to load case no. 1. In the plots, there are shown: the initial (undeformed) shape of the beam and three deformed beam shapes composed of the points that are located at the measuring stations. The deformed shapes have been obtained using the analytical formulas, the nodal deformation from the MBDyn model, and the deformation computed using the algorithm.

All the methods give very similar results for the bending deformation; the plots of the deformed beam are coincident. The difference in the results is present only in the case of torsion, which is shown in Figure 5.4. The twist resulting from the MBDyn model is not null, but negligibly small. The torsion from MBDyn is most probably affected by some numerical issues that can be safely neglected.

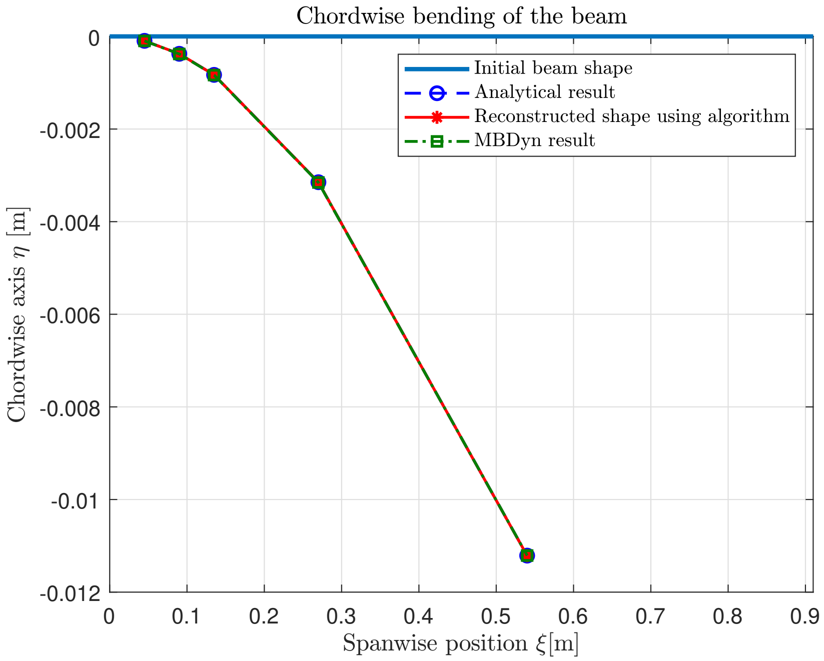


Figure 5.2: Chordwise displacement of the beam for small deformation, load case no. 1

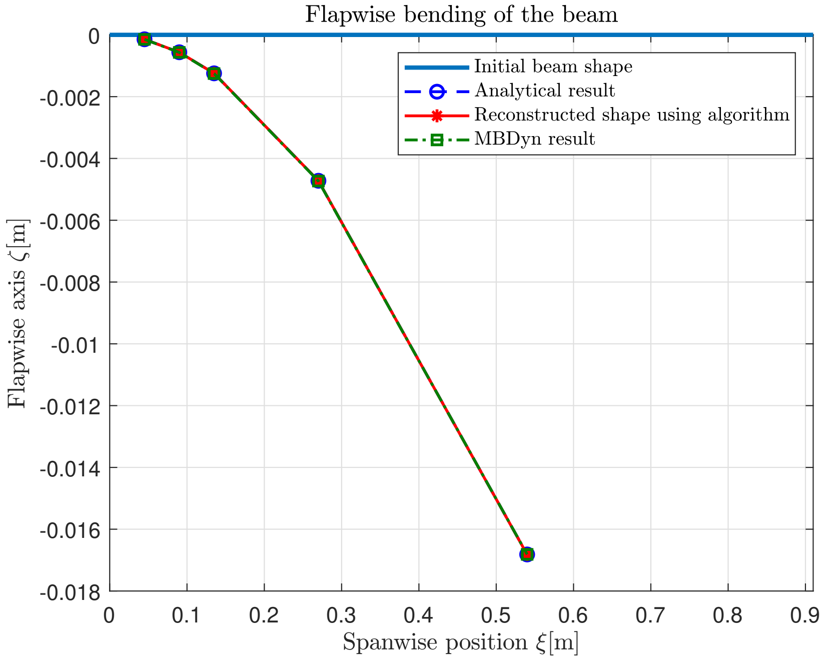


Figure 5.3: Flapwise displacement of the beam for small deformation, load case no. 1

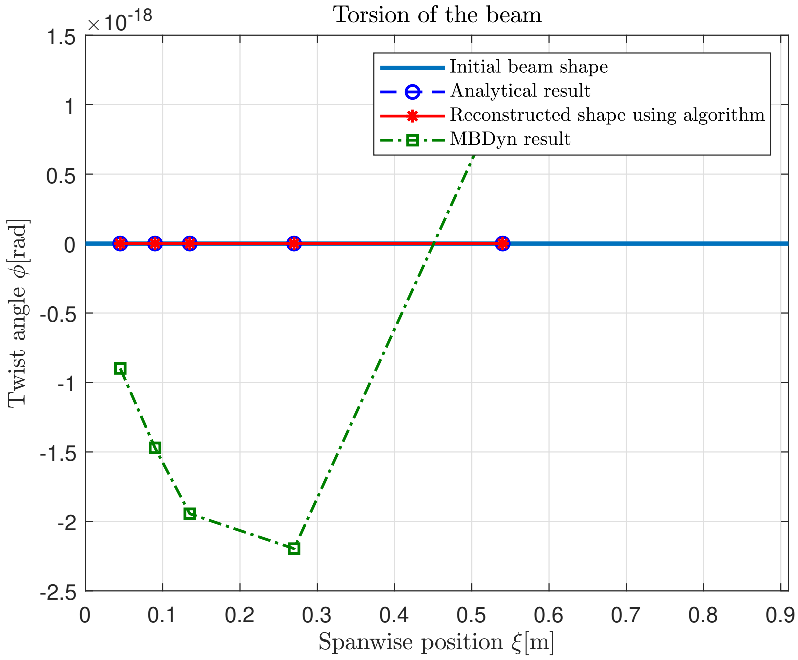
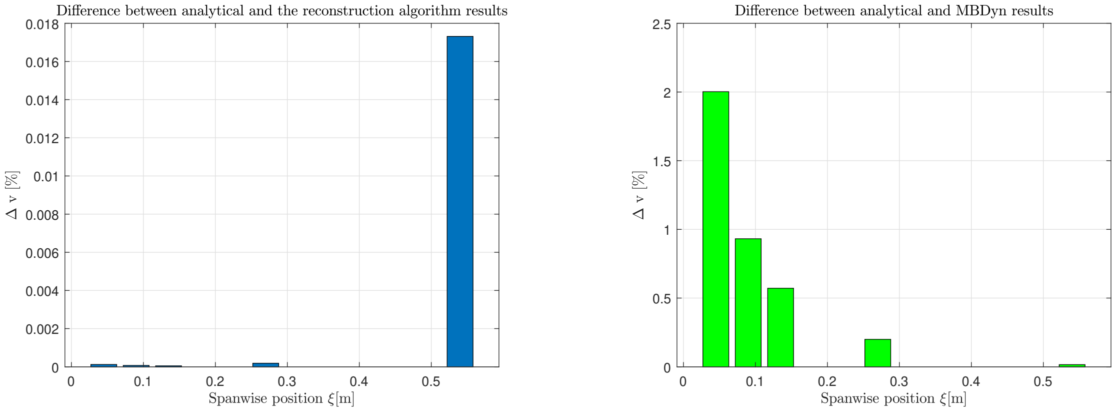


Figure 5.4: Twist of the beam for small deformation, load case no. 1

Figure 5.5 and Figure 5.5 present the difference between the shape reconstruction algorithm and the analytical results, and between the MBDyn model results and the analytical results for the chordwise and flapwise bending. The difference in flapwise and chordwise directions are the same for the algorithm and the MDByn model.

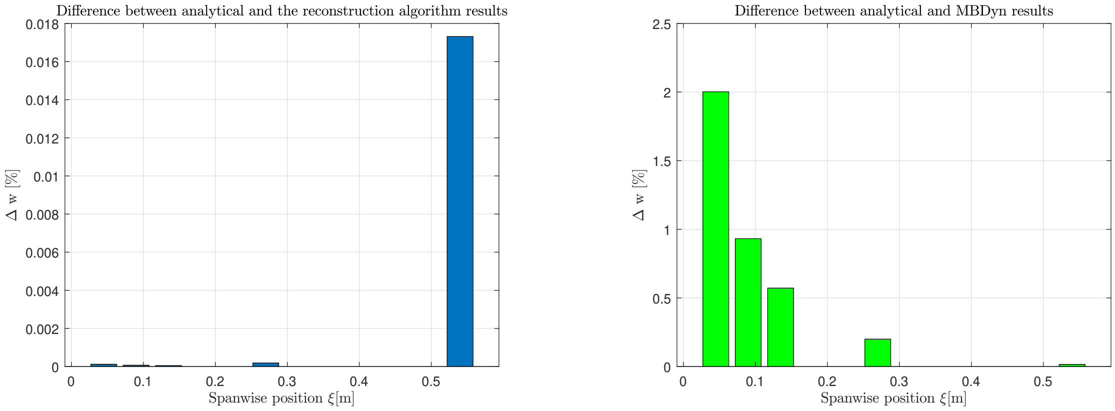
The algorithm in which analytical strains are used gives very small difference to analytical results, the largest discrepancy is given by the measuring station that is the closest to the tip of the beam. The difference between the MBDyn model and the analytical results is larger, however it does not exceed 2% and its distribution is reversed with respect to the difference between the algorithm and the results from the Euler-Bernoulli model. The largest error is given by the station that is close to the root of the blade. This effect is caused by the fact that in the analytical formulas the Euler-Bernoulli beam model is used, while MBDyn exploits Timoshenko beam model which accounts also for the shear contribution in the transverse displacement. The shear contribution increases linearly with the spanwise location (measured from the tip), whereas bending contribution increases with the third power of the spanwise location. In the Timoshenko beam model, both contributions are present, while in the Euler-Bernoulli beam model only bending contribution is used. This difference between both models is the reason of the discrepancy that is present in the figures, this conclusion is proved in section 5.2.3. Formulas of both models are presented in section 2.3.



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.5: Difference in chordwise displacement results, Euler-Bernoulli beam model, small deformation, load case no. 1



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.6: Difference in flapwise displacement results, Euler-Bernoulli beam model, small deformation, load case no. 1

5.2.2. Small deformation load case no. 2

Figure 5.7, Figure 5.8, and Figure 5.9 present the deformation of the beam for load case no. 2, thus the two forces have the same magnitude and, in addition, there is also torque applied to the end of the beam. The deformation of the beam at the measuring points obtained using all the methods are coincident also in this load case.

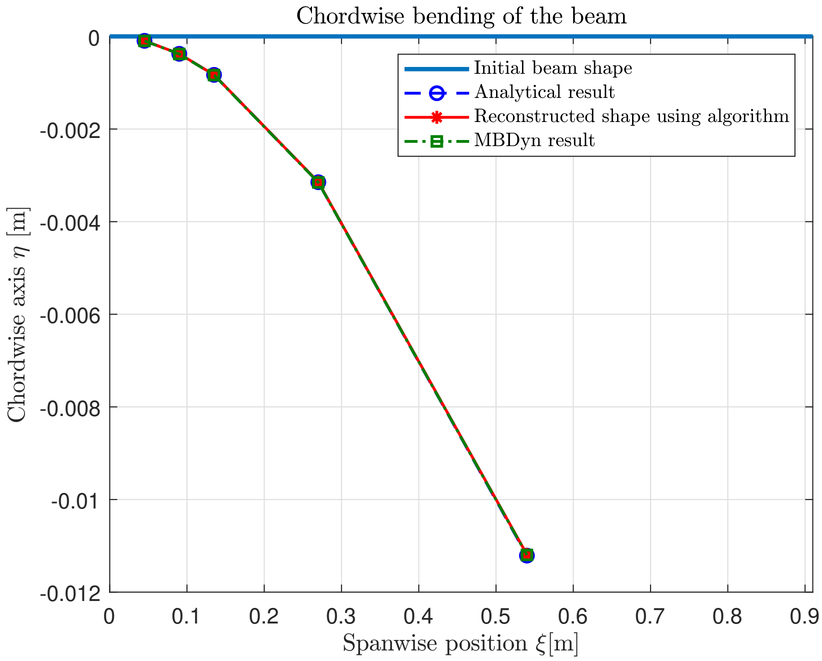


Figure 5.7: Chordwise displacement of the beam for small deformation, load case no. 2

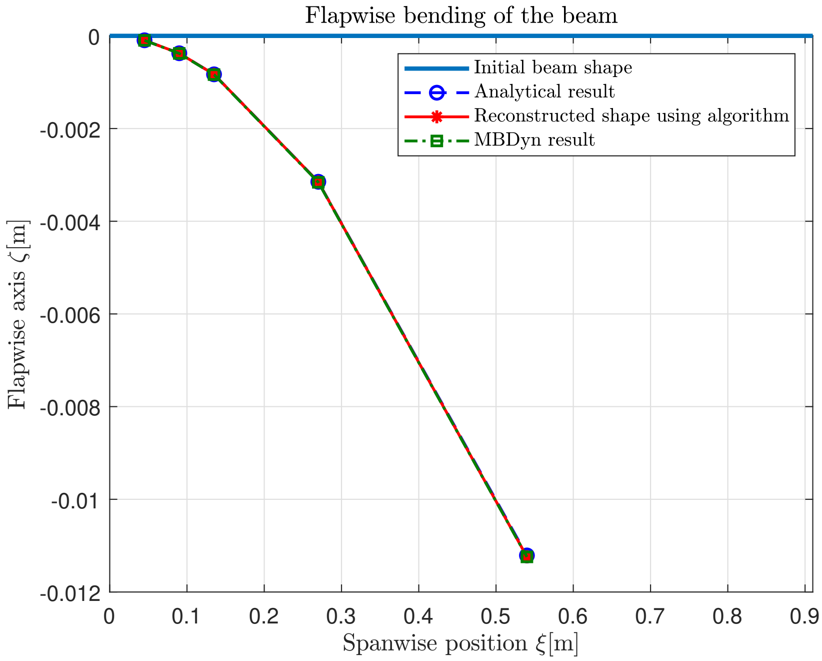


Figure 5.8: Flapwise displacement of the beam for small deformation, load case no. 2

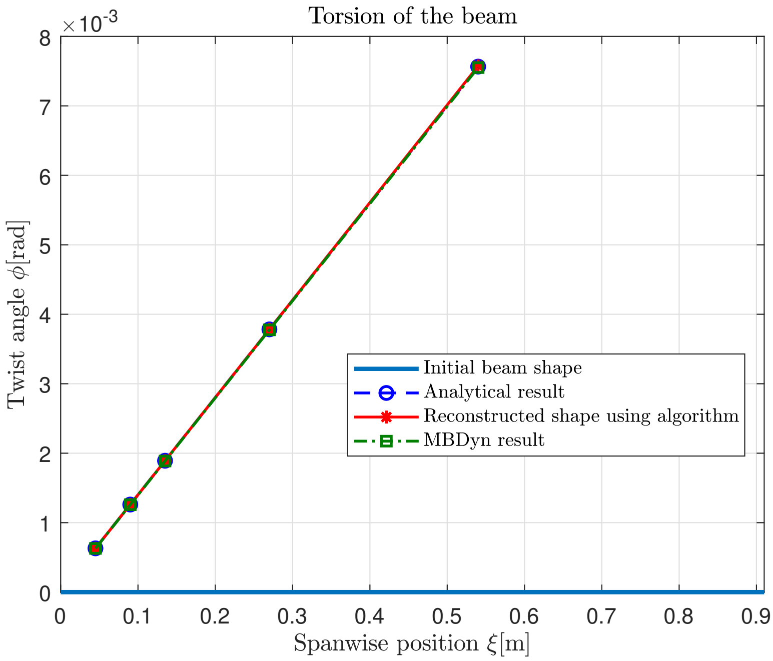
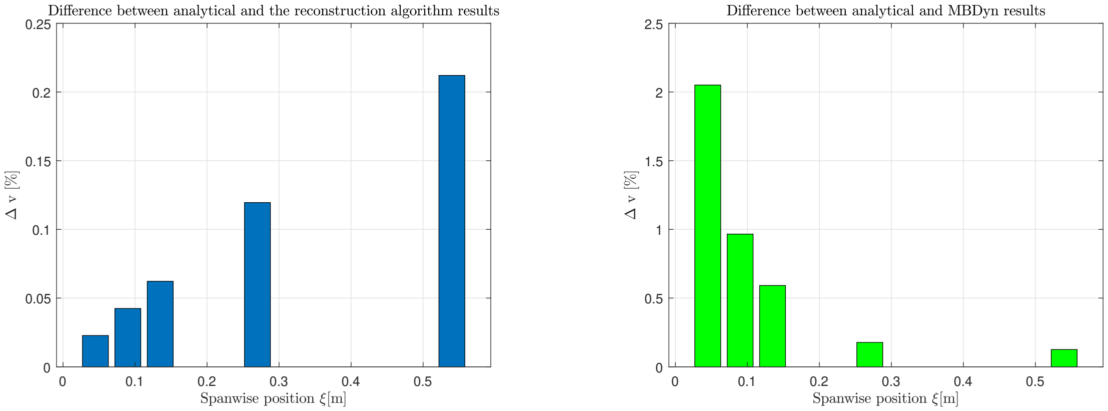


Figure 5.9: Twist of the beam for small deformation load case no. 2

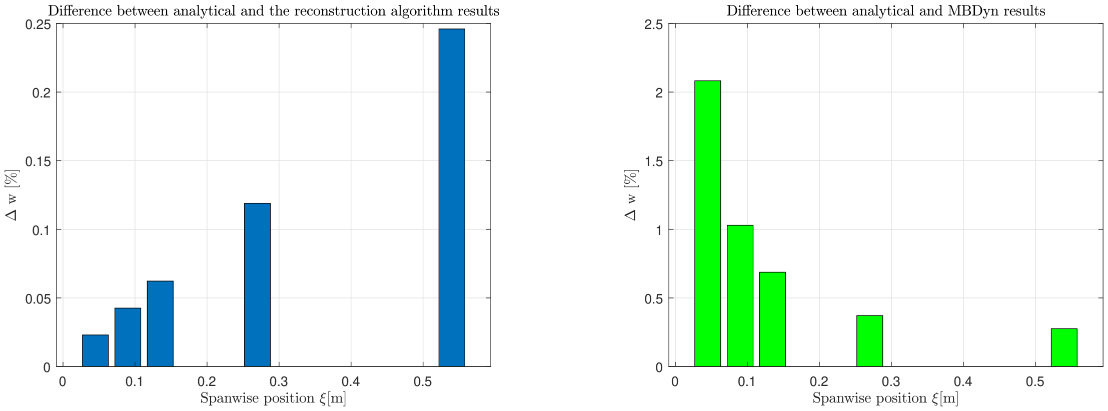
Figure 5.10 and Figure 5.11 present the difference between the reconstructed displacement and the analytical results. As shown, they are different in chordwise and flapwise direction, what is in contradiction to the results from the load case no. 1 presented in Figure 5.5 and Figure 5.6. The difference in flapwise displacement is greater than in the chordwise direction, moreover, both are larger than in the case when only forces were applied to the beam. A conclusion can be drawn that this additional difference in the results is caused by torque applied to the beam. From the shape reconstruction algorithm, which is presented in Figure 4.1, it is clear that bending is computed using also twist angle and its first derivative. The same conclusions apply also to the difference between the analytical results and the MBDyn simulation results.



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.10: Difference in chordwise displacement results, Euler-Bernoulli beam model, small deformation, load case no. 2

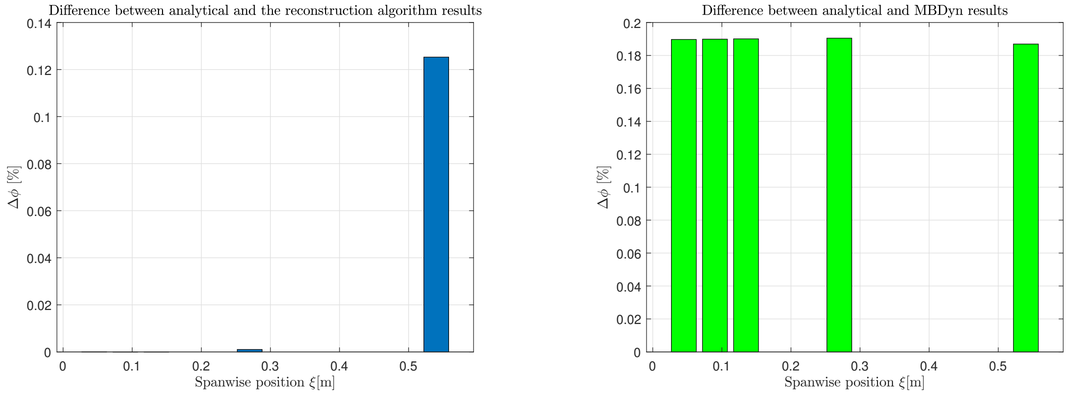


(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.11: Difference in flapwise displacement results, Euler-Bernoulli beam model, small deformation, load case no. 2

Figure 5.12 show the difference in twist. The discrepancy between the twist reconstructed using the algorithm and the analytical twist is very small, and can be considered as negligible, it is the largest in the case of the station that is the closest to the root of the blade, the other stations show a much smaller error. The error between twist from the MBDyn simulation and the analytical results is larger but still small, not exceeding 0.2%; the error is the same for all the stations.



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

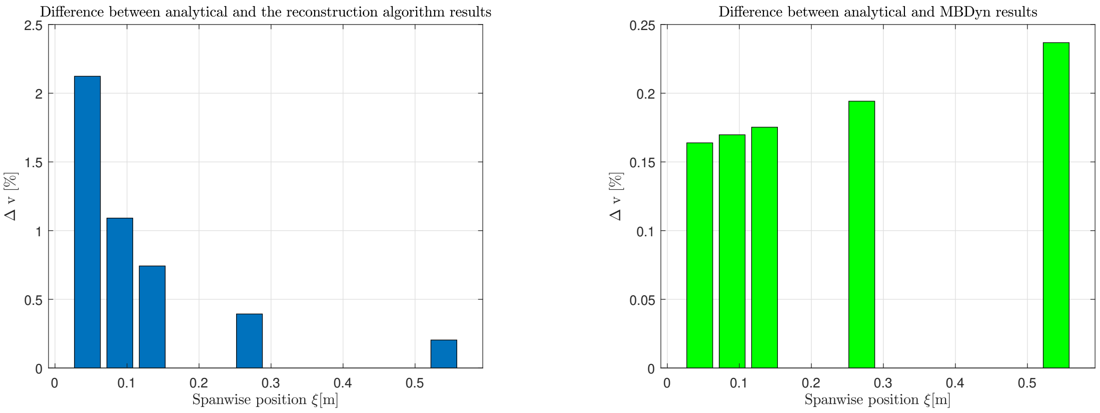
Figure 5.12: Difference in twist angle, small deformation, load case no. 2

5.2.3. Small deformation with analytical results from Timoshenko beam model

This subsection of the report presents the difference between the analytical results and respectively those from the algorithm and MBDyn simulations. However, in this case the transverse displacement has been computed using the Timoshenko beam model. In the presented results, load case no. 1 for small deformation has been used; the details of the applied forces are available in Table 5.1. Only results of difference in chordwise displacement are presented in the following, due to the fact that, as shown in section 5.2.1, in this load case the differences in displacement along both axes are the same. Moreover, the results from the algorithm used to generate Figure 5.13a and Figure 5.14a were computed respectively using the analytically computed strains and those from MBDyn.

Figure 5.13 and Figure 5.14 present the difference in chordwise displacement of the points located at the measuring stations. When the Timoshenko beam model is used, the difference between the algorithm and analytical formulas is much greater than in the case of analytical results from the Euler-Bernoulli model. In the case of analytical strains used in the algorithm (Figure 5.13a) the difference exceeds 2%, while previously it was around 0.018%. The distribution of the difference is also reversed, the highest discrepancy occurring at the station close to the root. In the case of MBDyn simulation results compared to the analytical results, the situation is reversed: the discrepancy decreased significantly, from 2% to 0.19%. The distribution is also reversed, the largest difference occurring at the point closest to the tip of the blade. Those results prove that the distribution of discrepancy between MBDyn and analytical results present in section 5.2.1 and section 5.2.2 is caused by different models used for the transverse displacement calculations. In the presented thesis, the Euler-Bernoulli model is used for analytical computations of transverse displacement, unless otherwise stated.

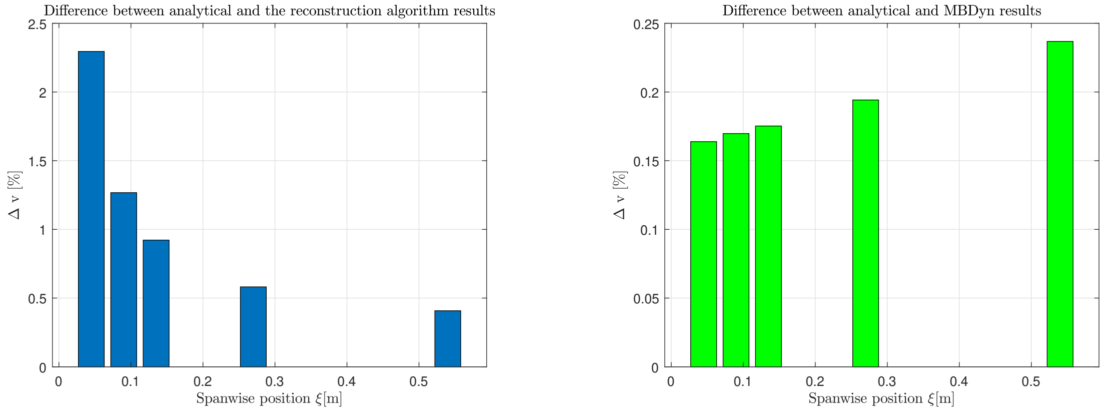
Interesting is a comparison of the difference between the analytical results and the algorithm in which strains from different sources have been used. Figure 5.13a presents the results obtained using analytical strains, whereas Figure 5.14a using MBDyn strains. The distribution of discrepancy in both figures is the same, so the greatest difference is present in the case of the measuring station at the root. This difference is slightly larger for the MBDyn strains used in the algorithm.



(a) Difference between analytical results and algorithm shape reconstruction using analytical strains

(b) Difference between analytical results and MBDyn model

Figure 5.13: Difference in chordwise displacement, Timoshenko beam model, small deformation, load case no. 1



(a) Difference between analytical results and algorithm shape reconstruction using MBDyn strains

(b) Difference between analytical results and MBDyn model

Figure 5.14: Difference in chordwise displacement, Timoshenko beam model, small deformation, load case no. 1

5.3. Validation of the algorithm using large deformation

Large deformation is assumed to be in the range of 10%–30% of the beam spanwise length, while the rotation is in the range of 0.02–0.06 rad. There two load cases, shown in Table 5.2, were simulated. One of the load cases is without the torque and with the loads applied in positive and negative direction, while in the other load case two forces are positive with negative torque. Five modes have been used for the reconstruction of the displacement of the beam at the measuring stations.

Load case no.	F_z[N]	F_y[N]	M_t[Nm]
1	120000	-200000	0
2	150000	150000	-10000

Table 5.2: Load cases for verification using large deformation

5.3.1. Large deformation load case no. 1

Figure 5.15, Figure 5.16, and Figure 5.17 present the deformations of the beam subjected to load case no. 1 for large deformation. The algorithm uses analytical strains to reconstruct the shape of the beam. In Figure 5.15 and Figure 5.16 it is shown that chordwise and flapwise deformation of the beam given by MBDyn are significantly smaller than the analytical results. Moreover, the reconstructed shape noticeably deviates from the deformation given by analytical formulas; in the case of chordwise deformation it is larger, while in the flapwise case it is smaller. The twist that is presented in Figure 5.17 is null from analytical results and the algorithm while the MBDyn simulation returns some twist, but it is negligibly small.

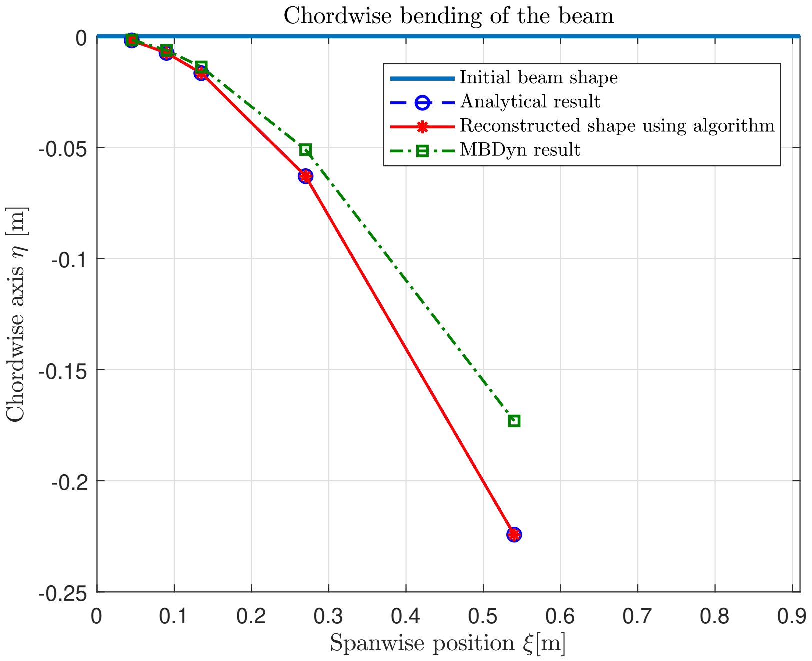


Figure 5.15: Chordwise displacement of the beam for large deformation, load case no. 1

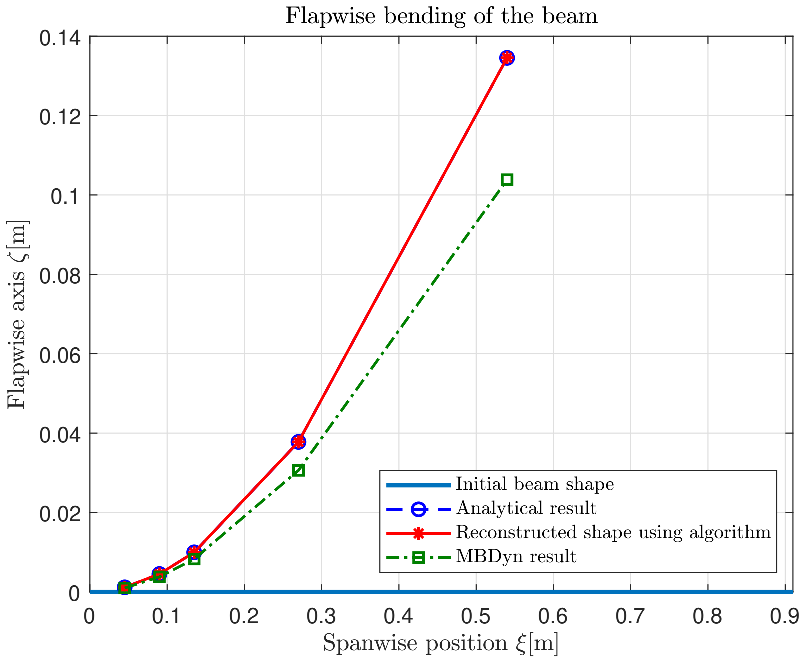


Figure 5.16: Flapwise displacement of the beam for large deformation, load case no. 1

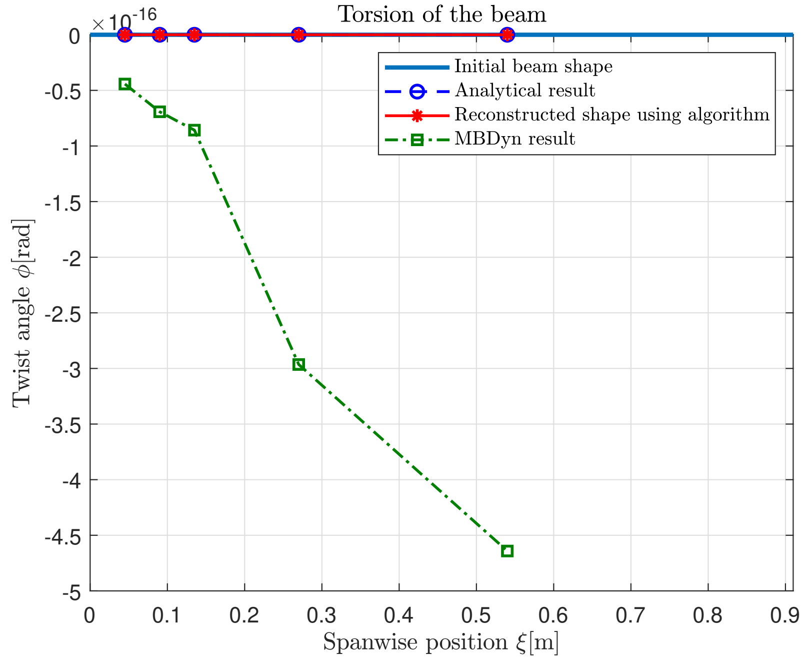
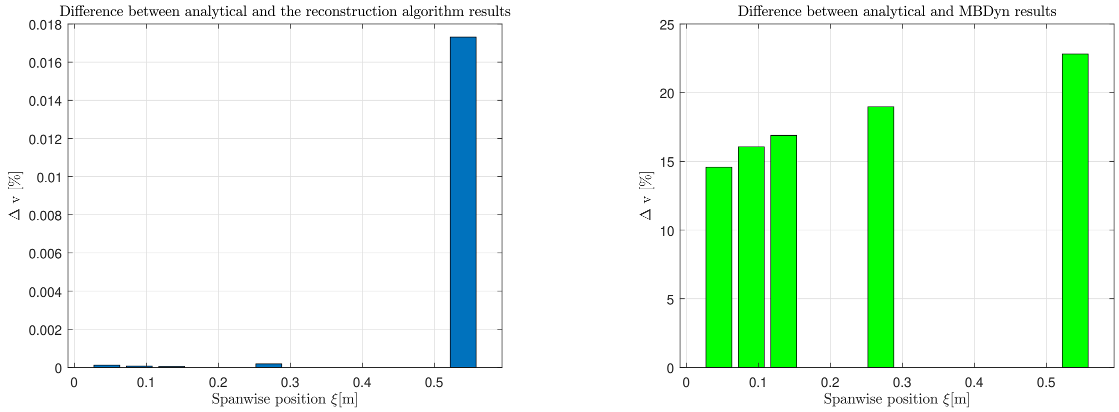


Figure 5.17: Twist of the beam for large deformation, load case no. 1

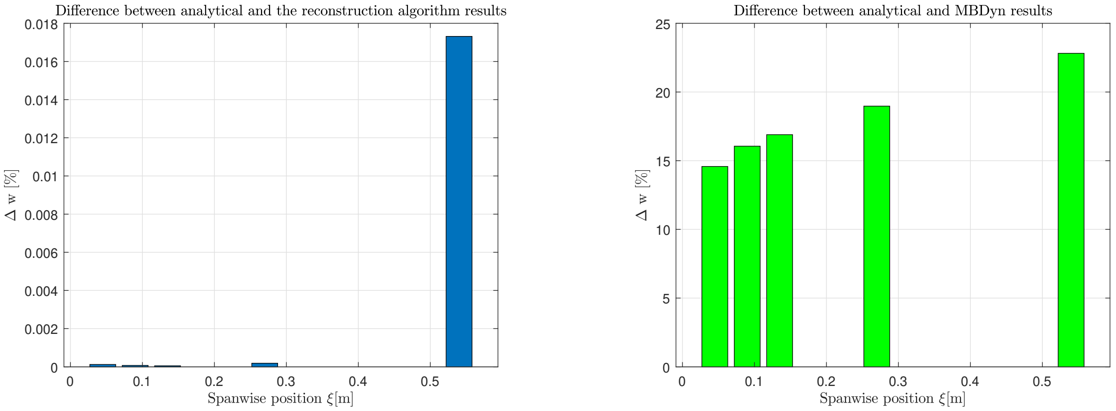
Figure 5.18 and Figure 5.19 present differences in the results from the algorithm and the MBDyn simulation with respect to the analytical results. The algorithm gives the same, negligible difference in both directions; the largest discrepancy is obtained for the station that is the closest to the tip of the blade. A much larger difference is obtained using the MBDyn results, it exceeds 20% and is the same in both directions. It is worth noticing that the distribution of discrepancy given by MBDyn in large deformation case is opposite to the discrepancy given for small deformation that is shown in Figure 5.5 and Figure 5.6. The presented difference must be driven by other factors than the difference in the Euler-Bernoulli and the Timoshenko beam model (discussed in section 5.2.3) and might be caused by the fact that if the load exceeds some magnitude, the problem becomes nonlinear. This topic is discussed in detail in section 5.4.



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.18: Difference in chordwise displacement, Euler-Bernoulli beam model, large deformation, load case no. 1



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.19: Difference in flapwise displacement, Euler-Bernoulli beam model, large deformation, load case no. 1

5.3.2. Large deformation load case no. 2

Figure 5.15, Figure 5.16, and Figure 5.17 present the deformation of the beam subjected to load case no. 2 for large deformation. The behaviour of the MBDyn results is the same as in the previous case, while the shape reconstructed using the algorithm deviates in a much stronger manner from the analytical results, but only in the case of bending results; the twist is coincident with that of the analytical beam model. The twist that is given by MBDyn significantly deviates from the other results, and is much smaller.

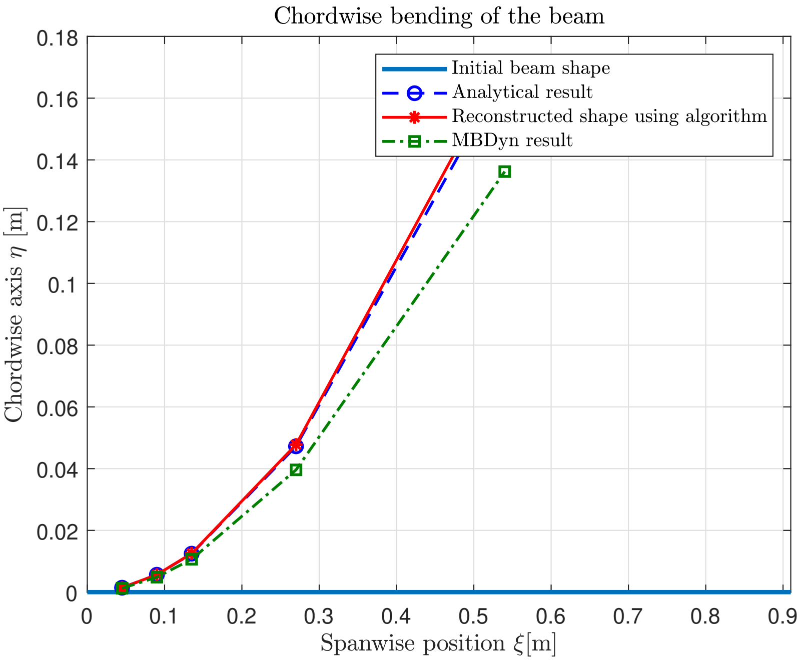


Figure 5.20: Chordwise displacement of the beam for large deformation, load case no. 2

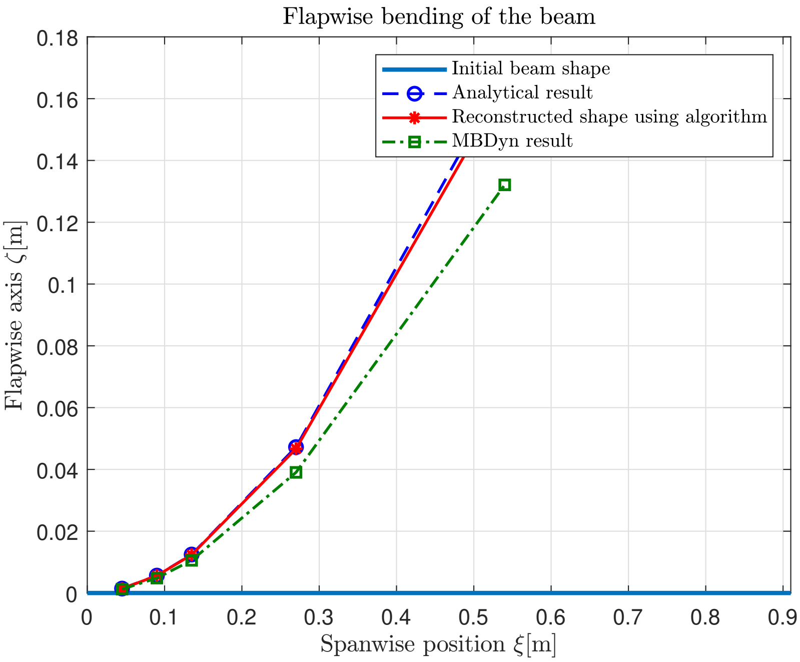


Figure 5.21: Flapwise displacement of the beam for large deformation, load case no. 2

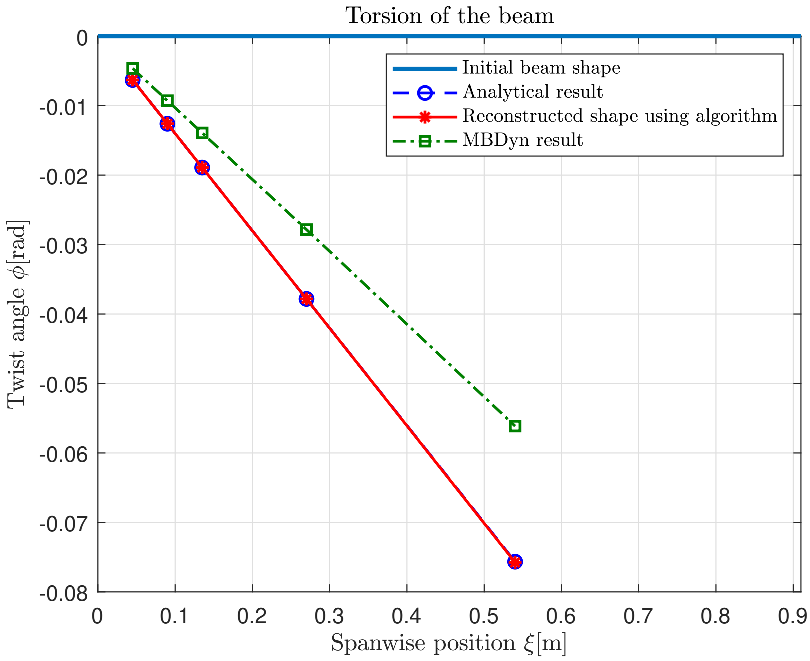
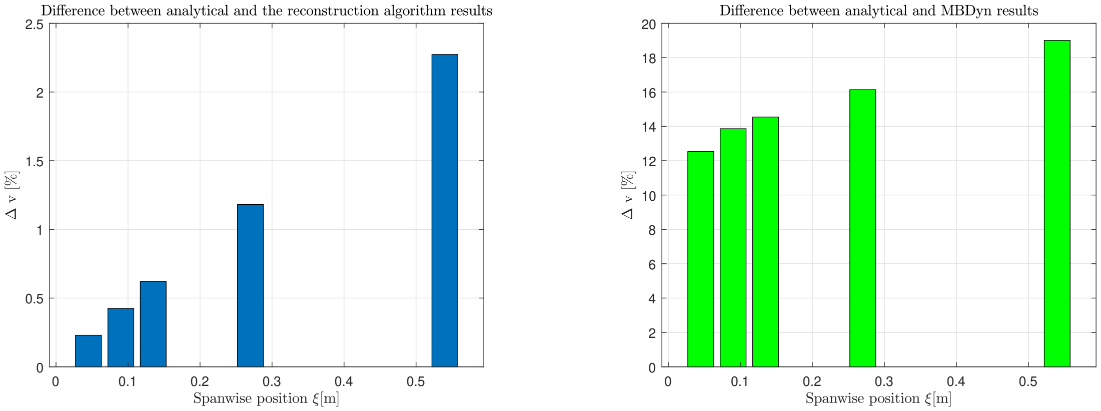


Figure 5.22: Twist of the beam for large deformation, load case no. 2

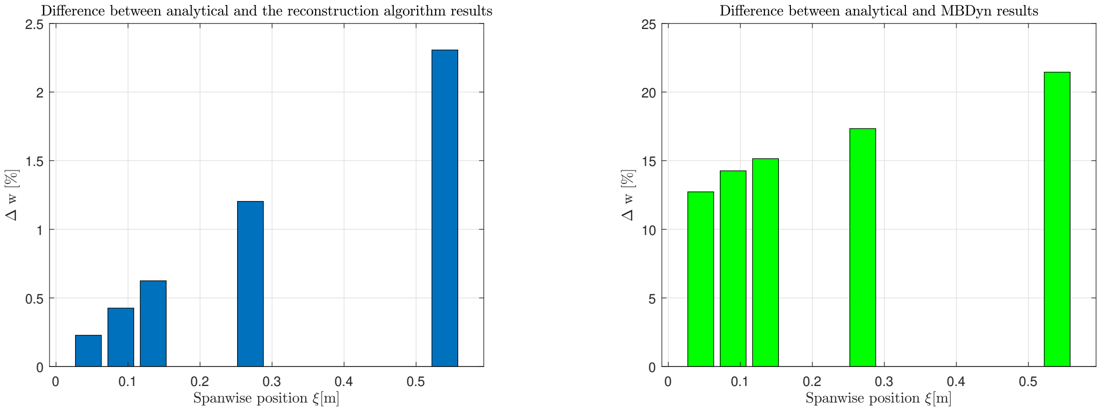
Figure 5.23 and Figure 5.24 show the deviation of the chordwise and flapwise results obtained using the algorithm and the MBDyn simulation with respect to the analytical results. The difference in chordwise and flapwise directions are not the same, the flapwise discrepancy is larger than the chordwise one. The difference is smaller in the case of the shape reconstruction algorithm than in the case of the MBDyn simulation. The algorithm gives a result that is not exceeding 2.5%; this difference is much larger than in the case of load case no. 1, without applied torque, where it was 0.018%. This confirms the conclusion that torque increases the discrepancy between the reconstructed shape and the analytical results.



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.23: Difference in chordwise displacement, Euler-Bernoulli beam model, large deformation, load case no. 2

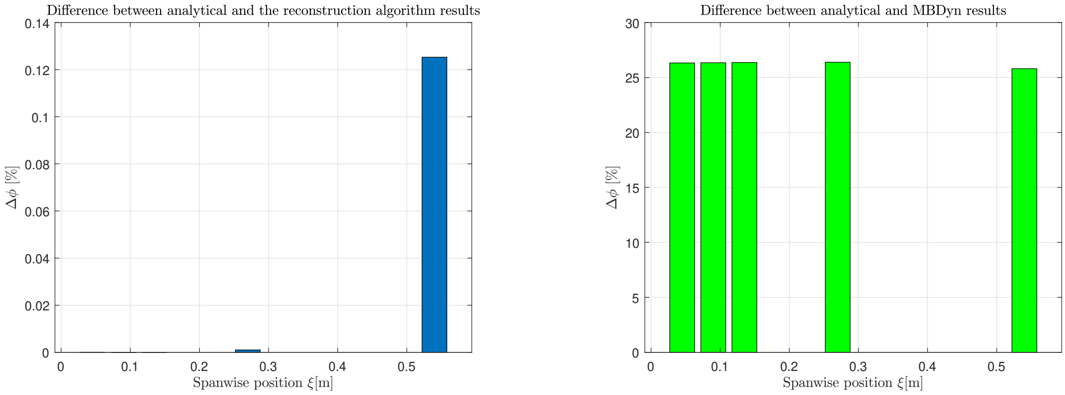


(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.24: Difference in flapwise direction Euler-Bernoulli beam model, large deformation, load case no. 2

Figure 5.25 presents the discrepancy in twist. The algorithm gives a very small deviation, which is the largest for the measuring station close to the tip, while MBDyn gives a significant difference that exceeds 25% and is more or less constant for all the stations.



(a) Difference between analytical results and algorithm shape reconstruction

(b) Difference between analytical results and MBDyn model

Figure 5.25: Difference in twist angle, large deformation, load case no. 2

5.3.3. Large deformation with strains from MBDyn simulation

The algorithm has also been validated using strains obtained from the simulations with MBDyn. For this purpose, load case no. 2 for the large deformation has been exploited, so the results presented in this section can be compared with the ones from section 5.3.2.

Figure 5.26 and Figure 5.27 present bending deformation of the beam. When the strains from MBDyn are used in the algorithm, the reconstructed shape is much closer to the results from the MBDyn simulation; however, the reconstructed values give a slightly larger deformation that is closer to that of the analytical results.

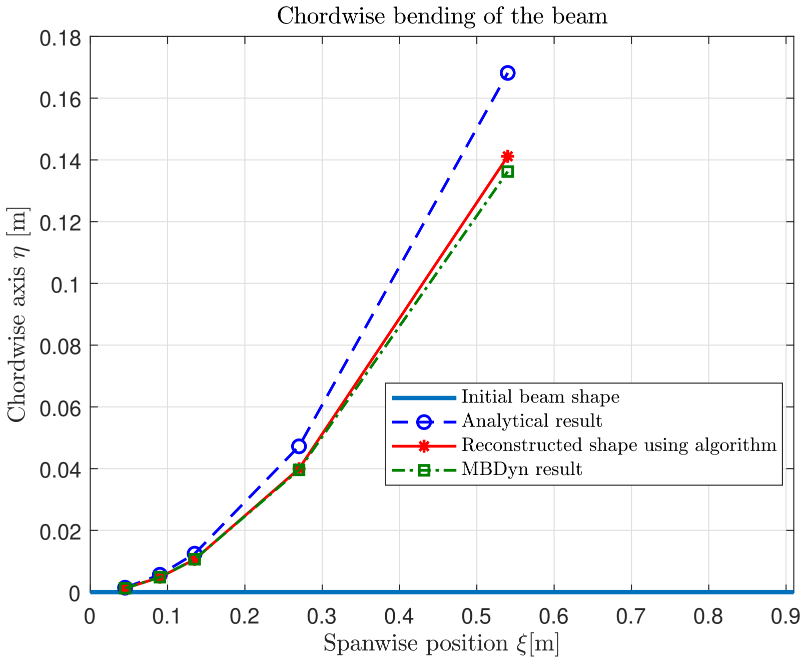


Figure 5.26: Chordwise displacement of the beam for large deformation, load case no. 2, MBDyn strains

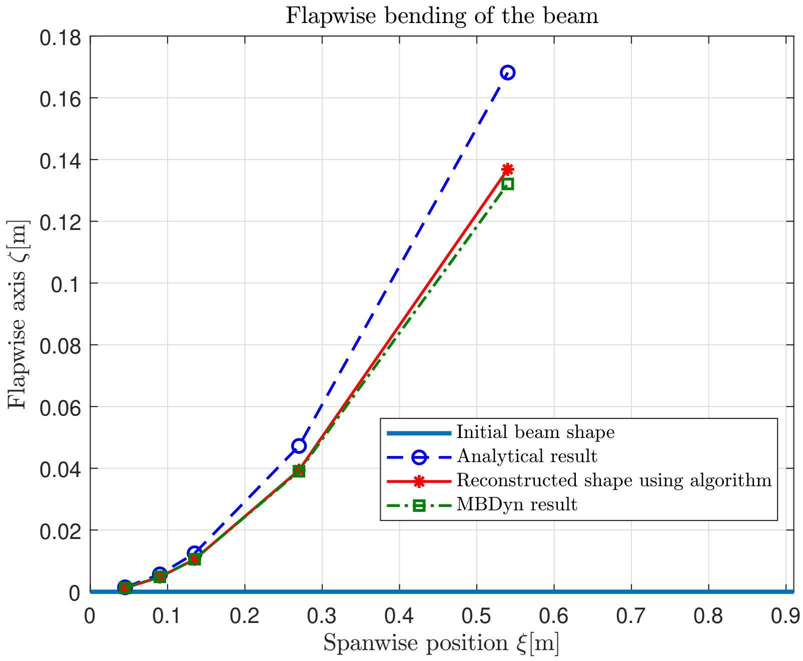


Figure 5.27: Flapwise displacement of the beam for large deformation, load case no. 2, MBDyn strains

Figure 5.28 presents the twist of the beam. The reconstructed angle is coincident with the MBDyn simulation results, it seems to be slightly smaller than what is shown by the analysis of the difference between the algorithm results and the analytical results that are shown in the further part of this subsection.

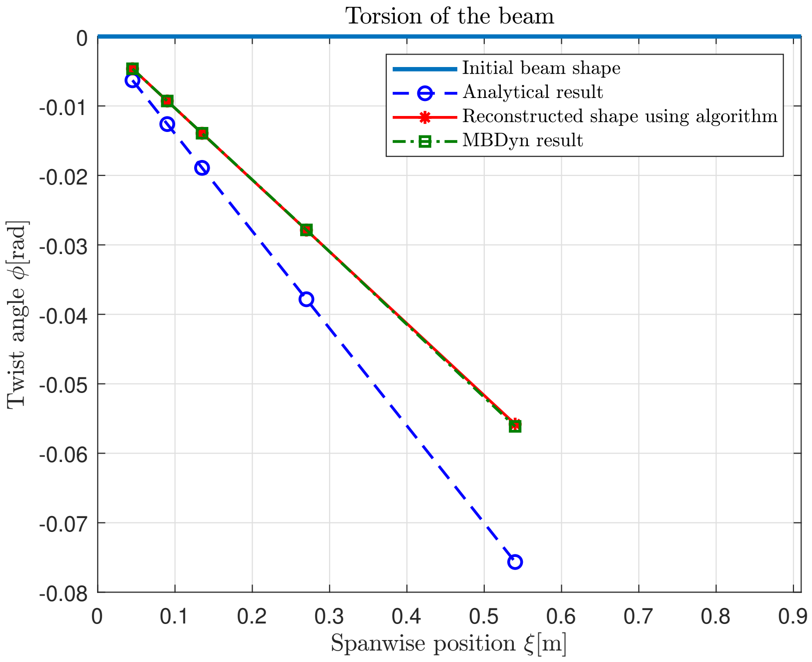


Figure 5.28: Twist of the beam for large deformation, load case no. 2, MBDyn strains

The plots below report only the difference between the analytical results and those from the algorithm in which the strains computed with MBDyn have been used, because the discrepancy between results from MBDyn and the analytical ones is already reported in section 5.3.2.

Figure 5.29 and Figure 5.30 present the difference in bending for flapwise and chordwise direction. The discrepancies in bending are much larger when strains obtained with MBDyn are used in the algorithm; they exceed 14% in both directions. The distribution of the difference is much more uniform than that resulting from the analytical strains; however, the largest discrepancy is given by the sensor that is closest to the tip of the beam.

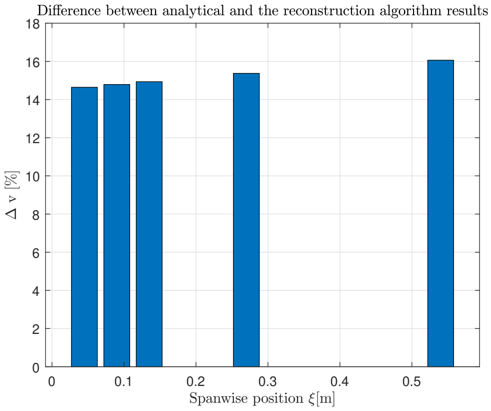


Figure 5.29: Difference in chordwise deformation, load case no. 2, MBDyn strains

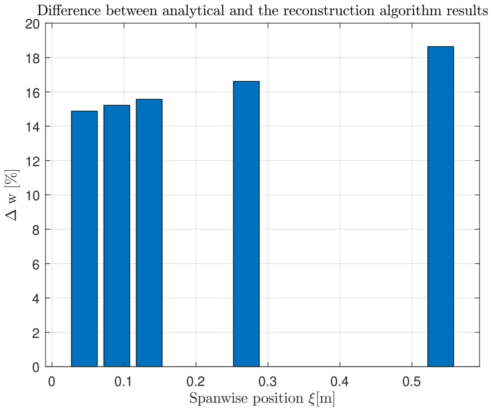


Figure 5.30: Difference in flapwise deformation, load case no. 2, MBDyn strains

Figure 5.31 presents the differences in twist angle. They are also much larger than in the case of the analytical strains used in the algorithm and exceed 26%. The distribution of the differences is more or less the same for all the measuring stations.

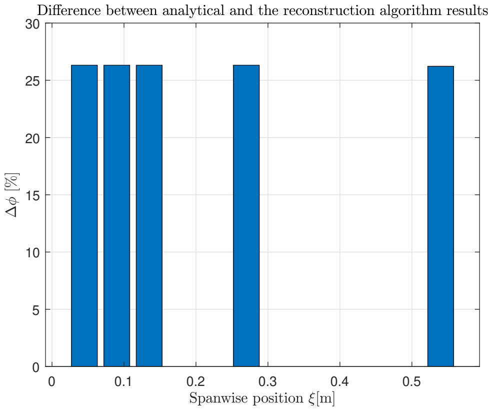


Figure 5.31: Difference in twist angle, load case no. 2, MBDyn strains

5.4. Linearity of the methods

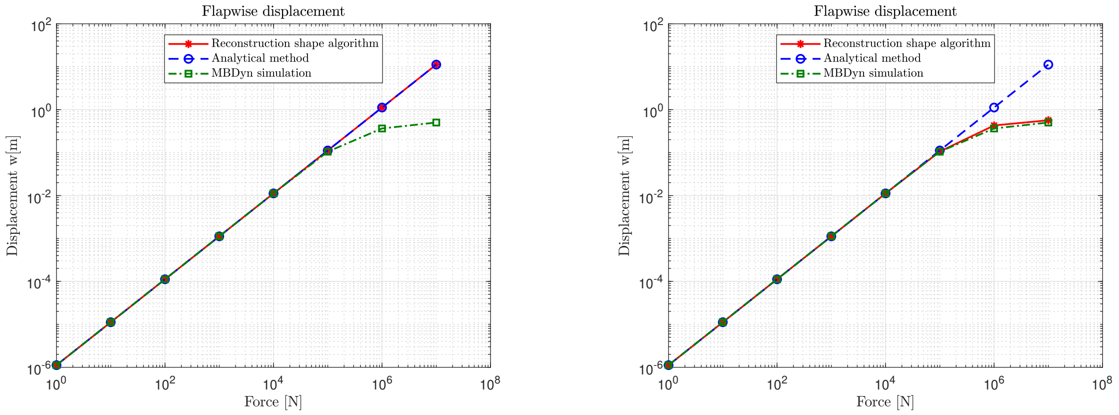
In this section, the verification of the linearity of the problem is performed. The strains used for this purpose have been generated using the analytical method and the MBDyn simulation, to check the influence of the strains on the results. The linearity has been verified for 3 load cases, that are shown in Table 5.3. The load on the beam has been increase by 1 order of magnitude at a time, up to when MBDyn solver was no longer able to converge. In load case no. 3, all the loads have been increased simultaneously. The plots that are shown in this section use a logarithmic scale on both axes.

Load case no.	$\boldsymbol{F_z}$	$\boldsymbol{F_y}$	$\boldsymbol{M_t}$
1	✓	×	×
2	×	×	✓
3	✓	✓	✓

Table 5.3: Load cases for linearity verification

5.4.1. Linearity, load case no. 1

Figure 5.32 presents the results of the linearity verification when only flapwise force is applied to the beam. The highest achieved load reaches 10^7 N. When the force magnitude is around 10^4 N, the solution departs from linearity; this becomes more and more noticeable for larger forces. When strains from MBDyn are used in the algorithm, its results follow MBDyn's solution, losing linearity for the same load values. When the analytical strains are used, instead, the algorithm is linear as the analytical model.



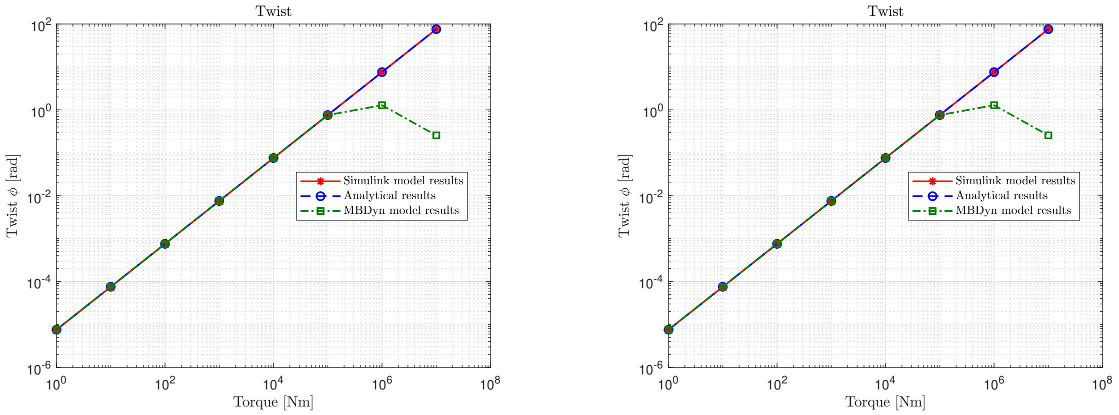
(a) Analytical strains used in the algorithm

(b) MBDyn strains used in the algorithm

Figure 5.32: Linearity plots for the load case no. 1

5.4.2. Linearity, load case no. 2

Figure 5.33 shows the results for the load case with a torque applied. The solution obtained with MBDyn becomes nonlinear above 10^4 N m. The shape reconstruction algorithm is always linear, even if the strains from MBDyn are used. Clearly, rotations greater than one radian are not physical for this problem, so departure from linearity for exaggerated values of torsional moment are not considered critical.



(a) Analytical strains used in the algorithm

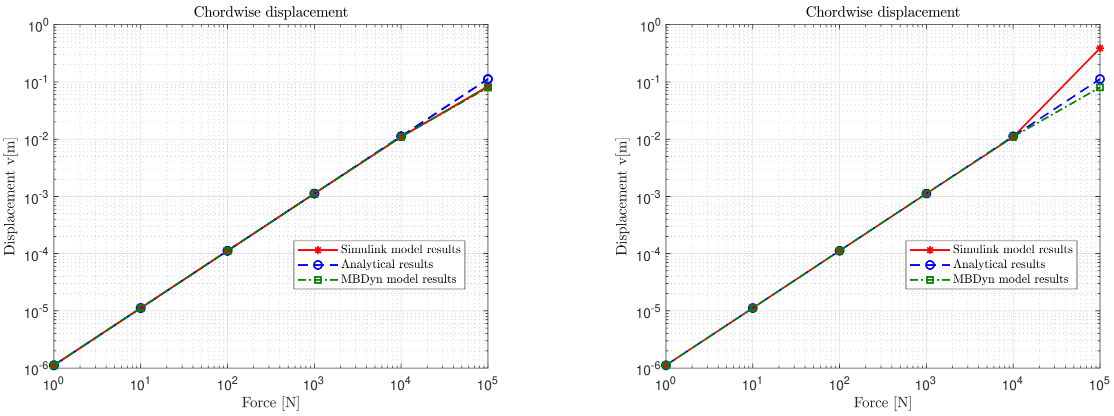
(b) MBDyn strains used in the algorithm

Figure 5.33: Linearity plots for the load case no. 2

5.4.3. Linearity, load case no. 3

This section presents the results of the linearity verification for load case no. 3, which is the most complex. The load is applied in the chordwise and flapwise axes together with torque. All loads are simultaneously increased; in this case the highest load that lead to convergence with MBDyn was of magnitude 10^5 N.

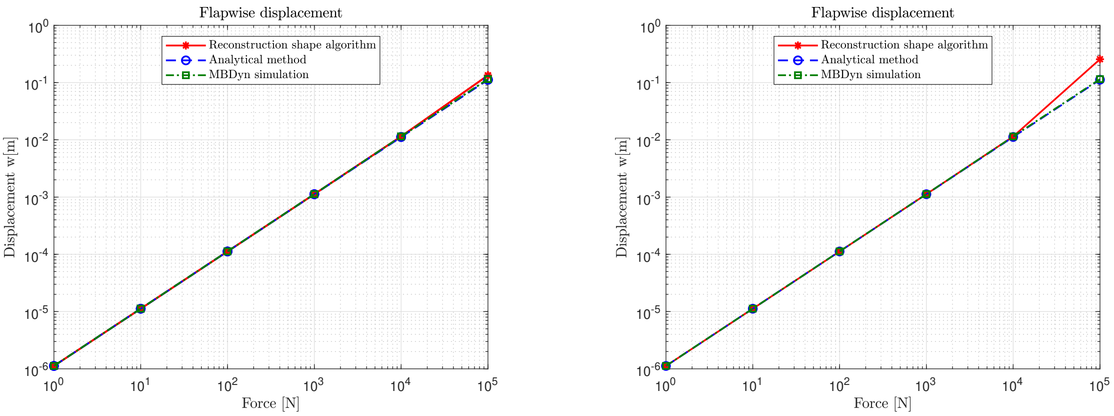
Figure 5.34 and Figure 5.35 show the linearity of the chordwise and flapwise reconstruction. The MBDyn simulation and the shape reconstruction algorithm deviate from linearity for forces higher than 10^4 N. If the analytical strains are used in the algorithm, it shows a smaller departure from linearity than MBDyn, whereas if the strains from the simulation are used, then it becomes highly nonlinear, even more than the solution obtained with MBDyn.



(a) Analytical strains used in the algorithm

(b) MBDyn strains used in the algorithm

Figure 5.34: Linearity plots for chordwise displacement, load case no. 3



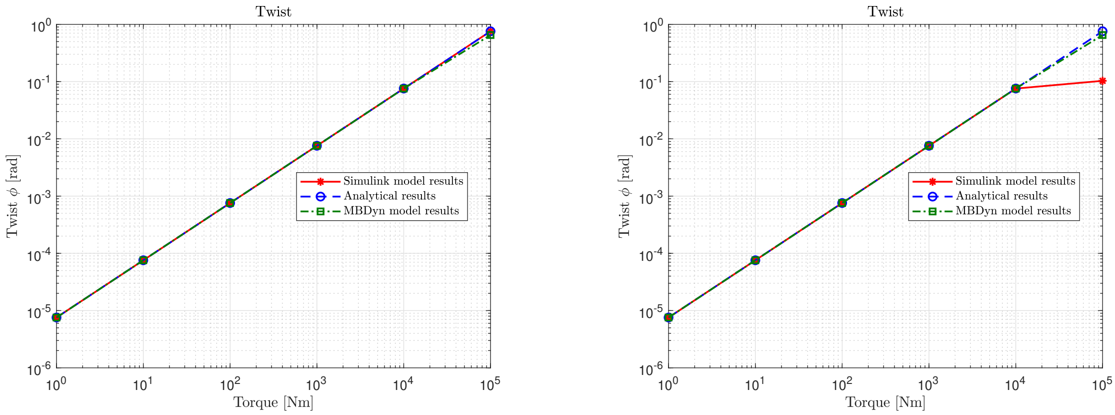
(a) Analytical strains used in the algorithm

(b) MBDyn strains used in the algorithm

Figure 5.35: Linearity plots for flapwise displacement, load case no. 3

Figure 5.36 presents the results of linearity verification for twist reconstruction. Results from MBDyn become slightly nonlinear when the torque exceeds 10^4 N m. If analytical strains are used in the algorithm, it is linear irrespective of the load magnitude, while if strains from MBDyn are used it becomes highly nonlinear above 10^4 N m. Deviation from linearity is greater than that of MBDyn's solution.

The conclusion is that the linear behaviour of the algorithm depends on the magnitude of the load and the source of the strains. The algorithm is always linear if only the torque is applied, while in the bending deformation it loses linearity if it uses nonlinear strains or if the torque is applied to the beam together with the transverse load.



(a) Analytical strains used in the algorithm

(b) MBDyn strains used in the algorithm

Figure 5.36: Linearity plots for twist angle, load case no. 3

5.5. Relation between number of modes and number of measuring stations

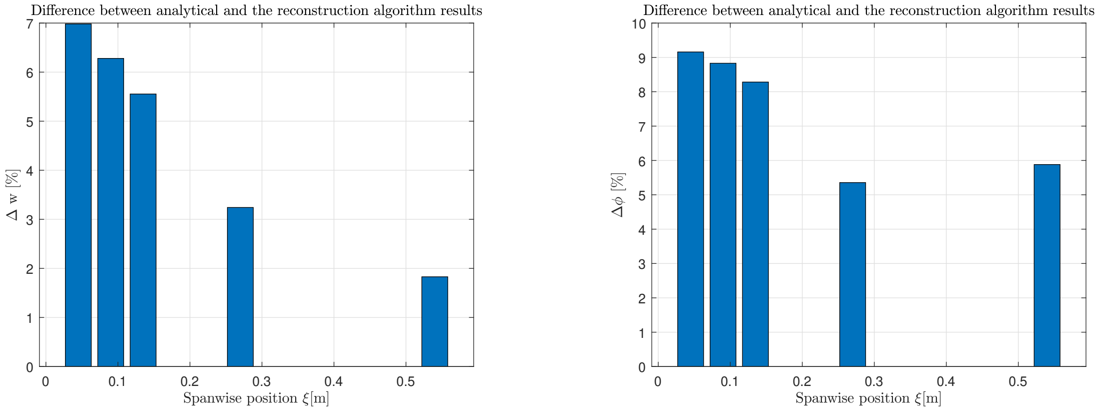
In the previously presented results, the number of bending and torsion shape functions has been set to 5. Also the number of measuring stations has been always set to 5. This section of the report presents how the reconstruction results are affected by changes in the number of modes or measuring stations. The small deformation load case no. 2 has been used to obtain the results presented in the subsequent part of this section. Details of the load case are shown in Table 5.1.

The difference between the analytical results and the shape reconstructed by the algorithm are reported in the next subsections. In the case of bending, only results in flapwise direction are reported, owing to the fact that the number of modes used in the reconstruction of flapwise and chordwise bending is always the same; so, the same conclusions apply. In the presented results, the same number of modes have been used to reconstruct bending and torsion.

5.5.1. Five measuring stations

This subsection presents the results obtained for 5 measuring stations. The modes in the shape reconstruction algorithm have been increased from 1 to 6 and simulated. The figures in this section present the difference between results given by the shape reconstruction algorithm and analytical formulas for flapwise bending and twist angle.

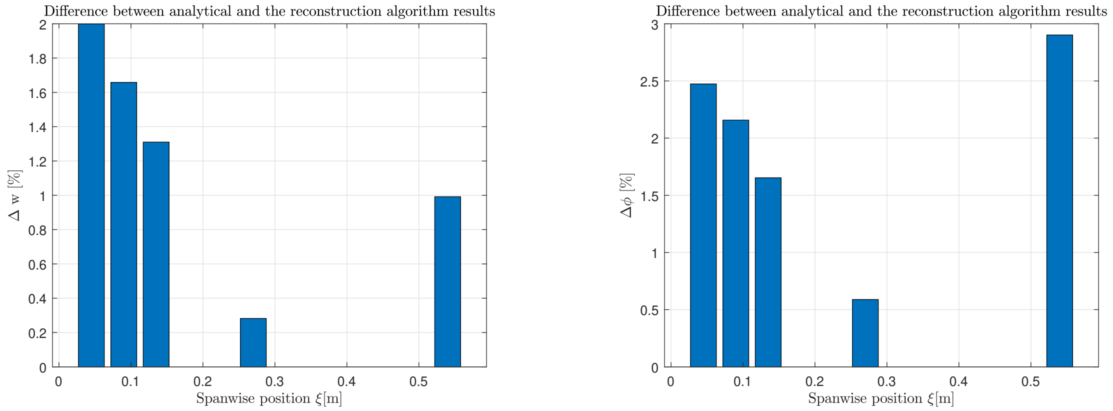
The quality of the results increases with the number of modes used for the reconstruction, something somewhat was expected. Figure 5.37 shows the results obtained using the first mode. In bending and torsion the quality of the results is poor, the difference is respectively around 7% and 9%. The discrepancy decreases significantly when 2 and 3 modes used. The best result for the shape reconstruction in both bending and torsion, is achieved with 5 modes. The results are shown in Figure 5.41. Figure 5.42 shows the results with 6 modes. It is clearly noticeable that the difference between analytical results and reconstructed shape increases significantly in both bending and torsion, reaching respectively around 21% and 18%. An explanation is that the problem becomes ill conditioned (potentially underdetermined) when the number of modes exceeds that of the measuring station.



(a) Flapwise bending

(b) Twist angle

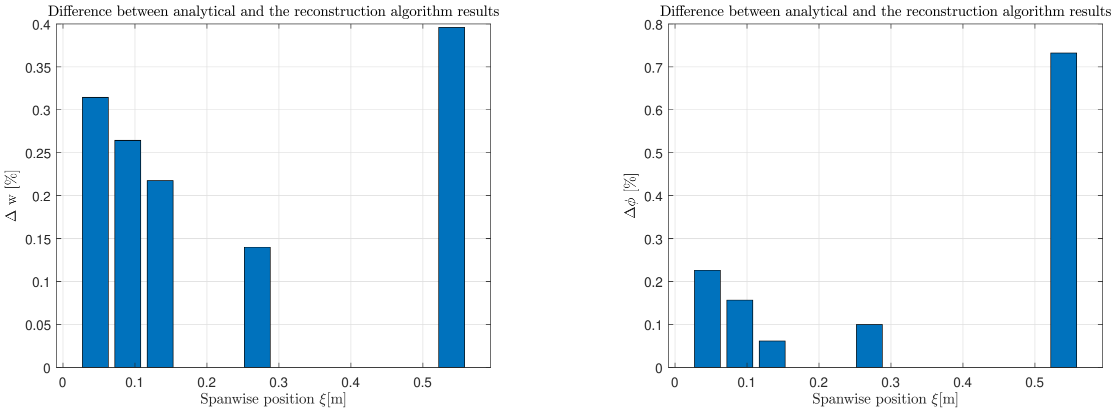
Figure 5.37: Difference between analytical results and results of the shape reconstruction algorithm for 1 mode shape used



(a) Flapwise bending

(b) Twist angle

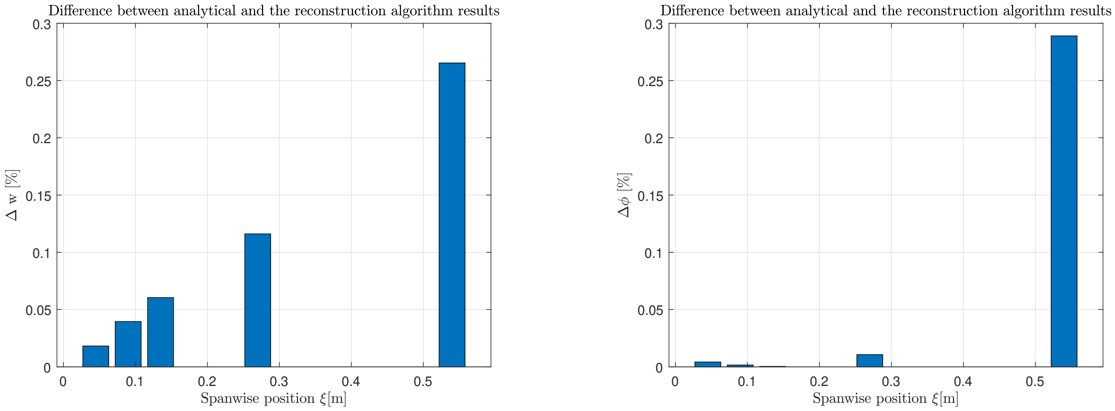
Figure 5.38: Difference between analytical results and results of the shape reconstruction algorithm for 2 mode shapes used



(a) Flapwise bending

(b) Twist angle

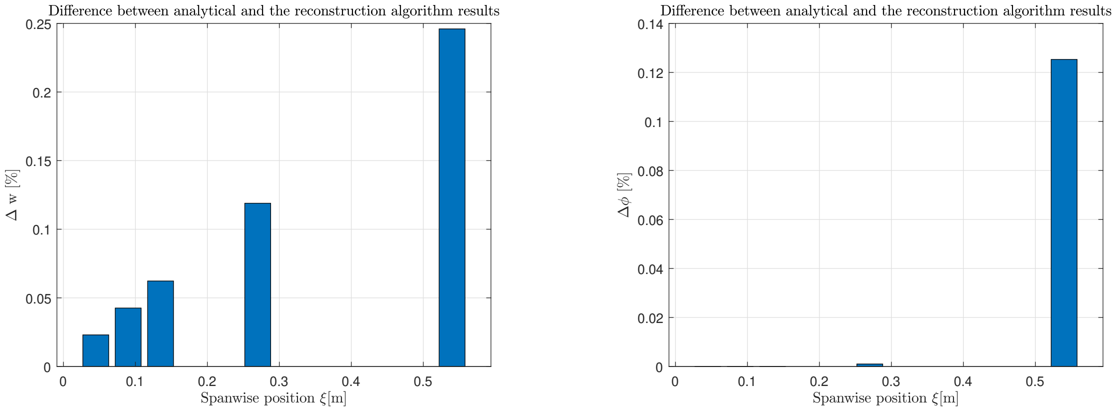
Figure 5.39: Difference between analytical results and results of the shape reconstruction algorithm for 3 mode shapes used



(a) Flapwise bending

(b) Twist angle

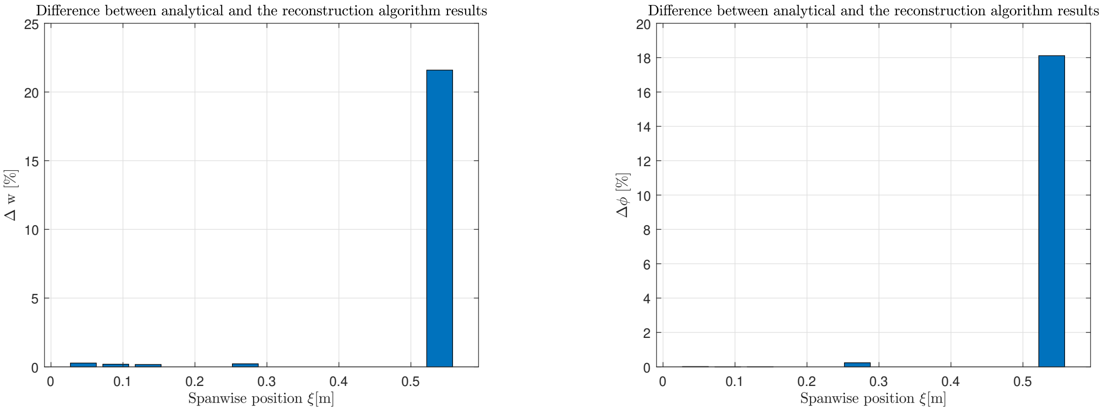
Figure 5.40: Difference between analytical results and results of the shape reconstruction algorithm for 4 mode shapes used



(a) Flapwise bending

(b) Twist angle

Figure 5.41: Difference between analytical results and results of the shape reconstruction algorithm for 5 mode shapes used



(a) Flapwise bending

(b) Twist angle

Figure 5.42: Difference between analytical results and results of the shape reconstruction algorithm for 6 mode shapes used

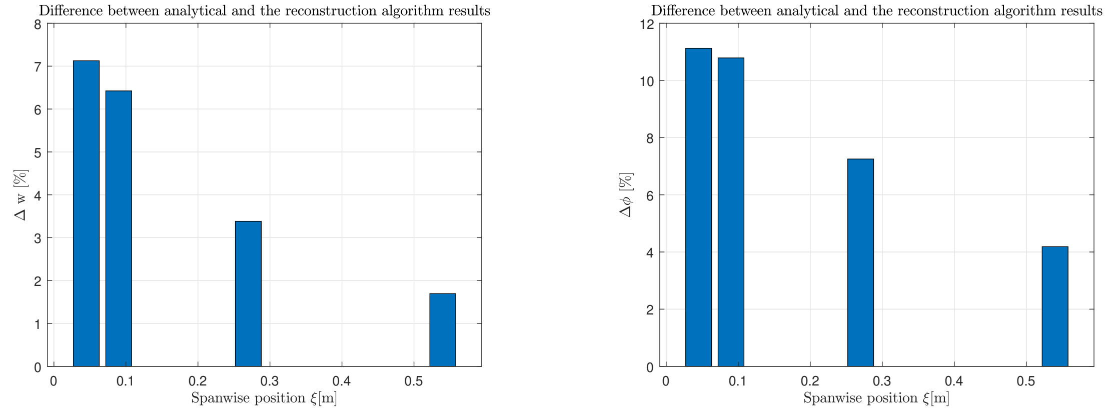
5.5.2. Four measuring stations

This subsection presents the results obtained with 4 measuring stations. The number of modes, in this case, is varied from 1 to 5. The measuring station that was removed from the results is the intermediate station no. 3, which is located in $\xi = 0.135$ m from the root of the blade.

Also with 4 measuring stations, the quality of the results increases with the number of mode shapes used for the reconstruction. Figure 5.43 shows the results with 1 mode, which once again provides the worst results. The differences are greater than with 5 measuring stations, as presented in Figure 5.37. With 4 measuring stations, the best quality of the results is obtained with 4 modes. The results are presented in Figure 5.46. However, in this case, the quality is not as good as when 5 modes were used with 5 measuring stations, as presented in Figure 5.41. This is explained by considering that the fifth mode adds a further degree of freedom for better fitting the measured strains.

When the number of modes increases to 5, i.e. exceeding the number of measuring stations, the quality of results deteriorates significantly, as presented in Figure 5.47.

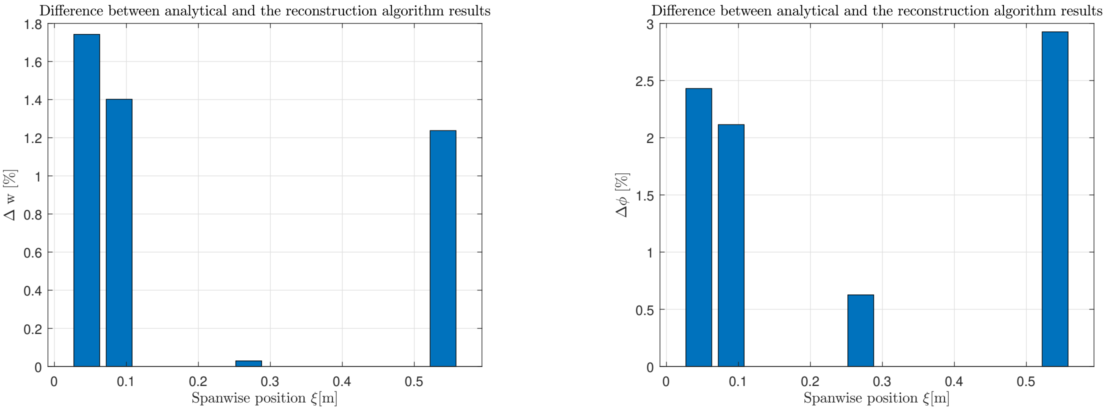
Very interesting is the fact that the results for 2 and 3 modes used for the reconstruction using 4 measuring stations (respectively Figure 5.44 and Figure 5.45) are better than in the case of 5 measuring stations (respectively Figure 5.38 and Figure 5.39). This conclusion applies to both bending and torsion. It can be explained by the fact that when 4 measuring stations are used instead of 5, 2 and 3 modes only need to fit 4 sets of measurements instead of 5, thus yielding better results.



(a) Flapwise bending

(b) Twist angle

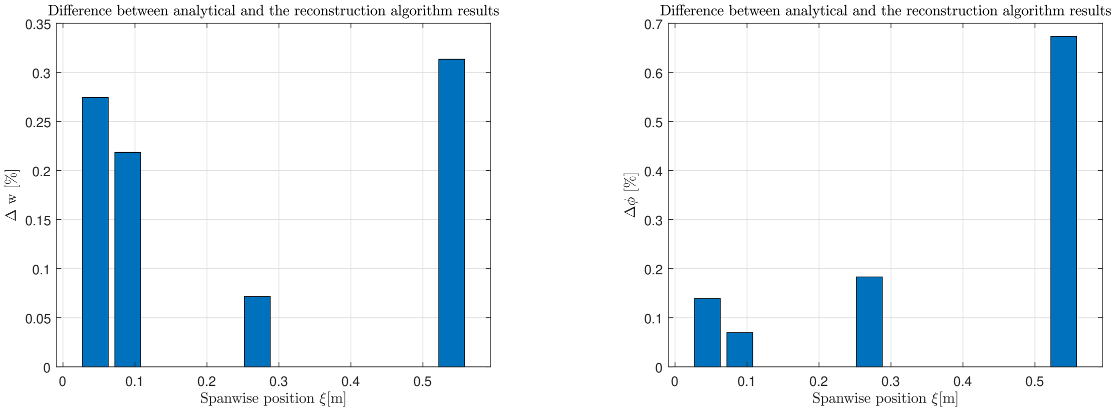
Figure 5.43: Difference between analytical results and results of the shape reconstruction algorithm for 1 mode shape used, 4 measuring stations



(a) Flapwise bending

(b) Twist angle

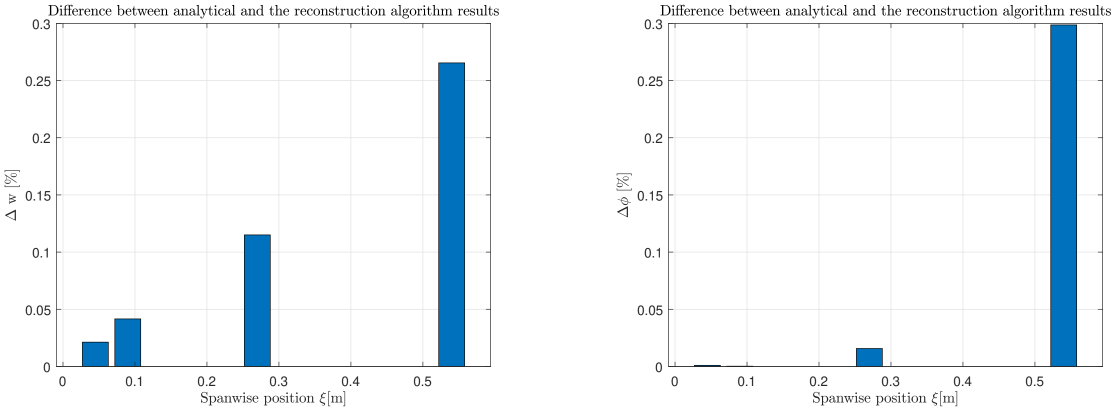
Figure 5.44: Difference between analytical results and results of the shape reconstruction algorithm for 2 mode shapes used, 4 measuring stations



(a) Flapwise bending

(b) Twist angle

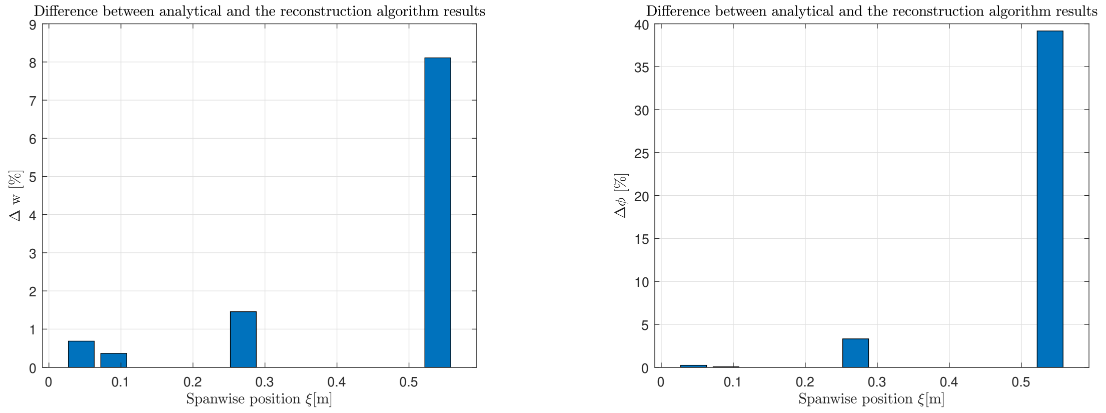
Figure 5.45: Difference between analytical results and results of the shape reconstruction algorithm for 3 mode shapes used, 4 measuring stations



(a) Flapwise bending

(b) Twist angle

Figure 5.46: Difference between analytical results and results of the shape reconstruction algorithm for 4 mode shapes used, 4 measuring stations



(a) Flapwise bending

(b) Twist angle

Figure 5.47: Difference between analytical results and results of the shape reconstruction algorithm for 5 mode shapes used, 4 measuring stations

5.6. Increased number of sensors per section

According to the theory, an increased number of the sensors should cause that the least-squares method applied to the algorithm will better alleviate potential random measuring error. To verify this assumption, a script was created that generates noise, which is added to the analytical strain measurements, assumed to be ideal. The load case used for the purpose of these simulations is load case no. 2 from a large deformation case. Details of this load case are presented in Table 5.2.

The noisy measurements are generated using the formula presented in Equation (5.1). The measurements are generated by adding to the original measurements their value multiplied by a random number drawn from a range $< -0.05, 0.05 >$. This formula is applied to all the sensors on the measuring station.

$$\varepsilon_{\text{noise}} = \varepsilon_{clean} + \varepsilon_{clean}\Bigg(\text{rand}(-0.05, 0.05)\Bigg) \tag{5.1}$$

To verify the influence of the number of sensors on the quality of the results given by the shape reconstruction algorithm, two additional sensors have been added: a strain gauge and an optical fibre sensor. In this way, one additional sensor has been used to compute the twist, and two additional sensors have been applied to the calculations of the bending deformation. The beam cross-section with additional sensors is shown in Figure 5.48. This cross-section has been used to generate the results for 5 sensors. The results for 3 sensors have been generated using the initial cross-section that is presented in Figure 5.1.

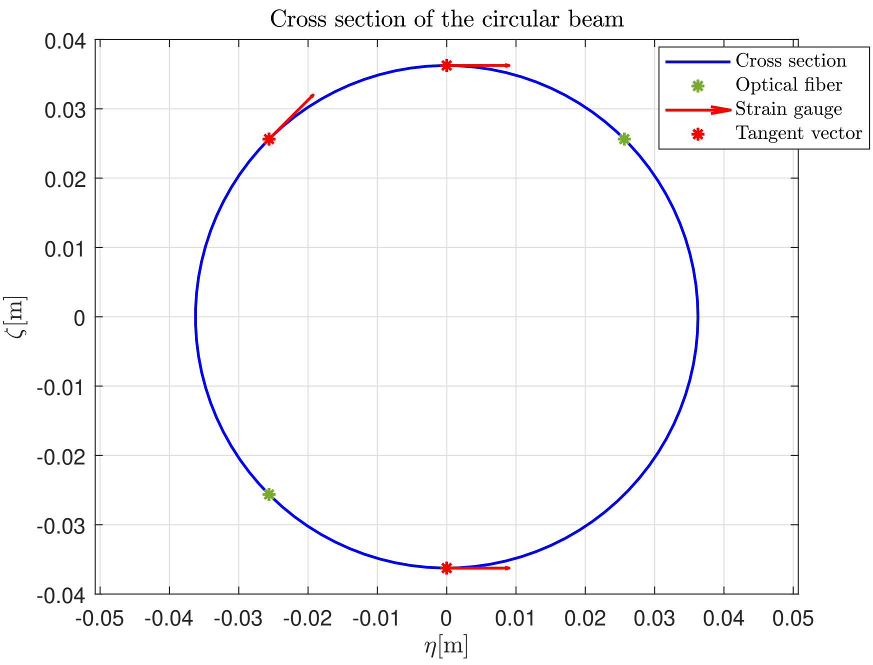
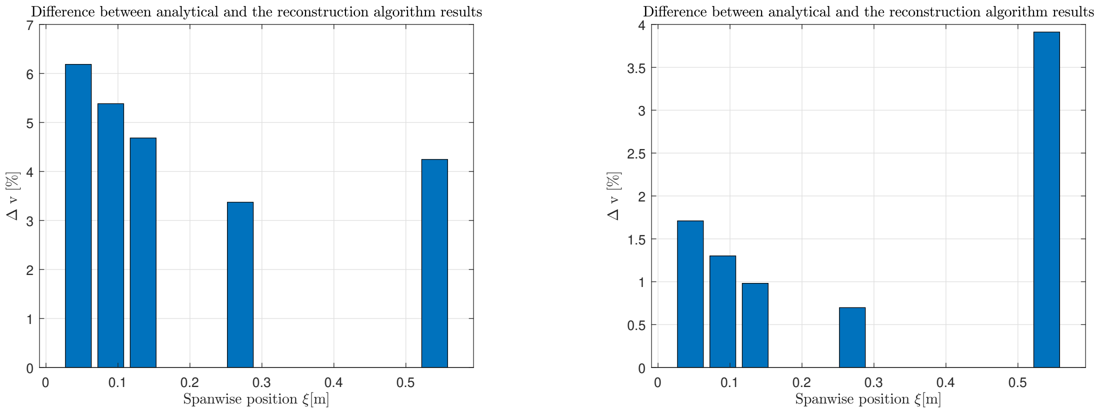


Figure 5.48: Beam cross-section with additional sensors

Figure 5.49 presents the difference in chordwise deformation between the analytical results and the results obtained using the algorithm for the 2 configurations, with 3 and 5 sensors per section. The discrepancy between the results is significantly decreased when five sensors are used on the cross-section. The distribution of the difference is random, as expected, because the noise is also a random signal.

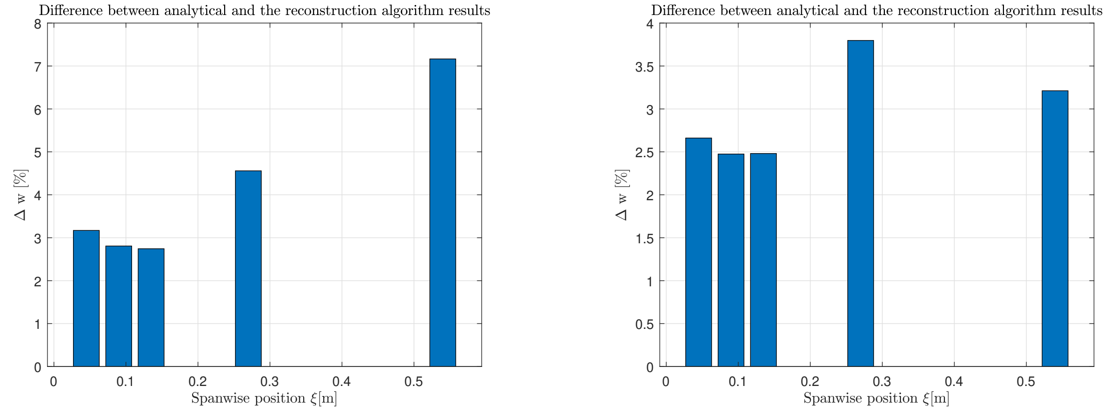


(a) Difference obtained using 3 sensors

(b) Difference obtained using 5 sensors

Figure 5.49: Difference in chordwise bending for different number of sensors

Figure 5.50 shows the difference in the case of flapwise direction. Also in this case the discrepancy significantly decreases when 5 sensors per section are used. The difference is randomly distributed.

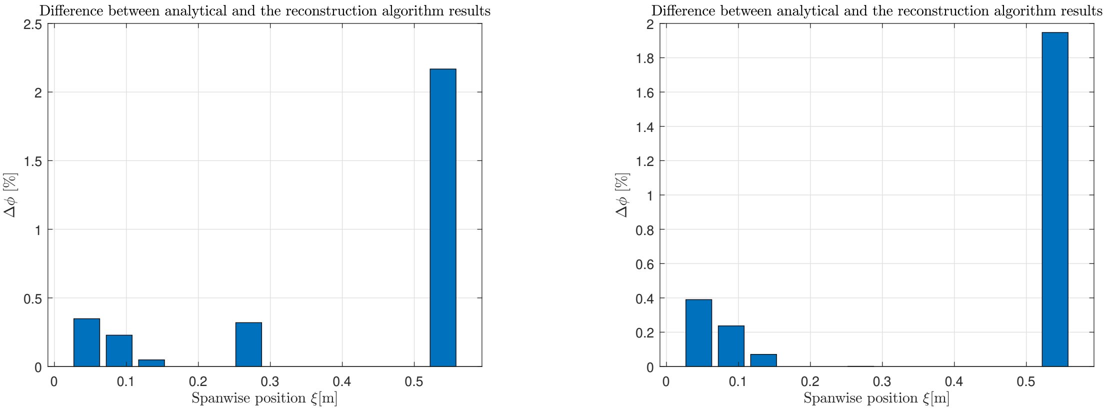


(a) Difference obtained using 3 sensors

(b) Difference obtained using 5 sensors

Figure 5.50: Difference in flapwise bending for different number of sensors

Figure 5.51 presents the discrepancy in twist. As in the other cases, the difference in twist also decreases if more sensors per section are used.



(a) Difference obtained using 3 sensors

(b) Difference obtained using 5 sensors

Figure 5.51: Difference in twist angle for different number of sensors

These results show that when random error is present in the measurements, an increased number of sensors may be considered as a method to alleviate its influence on the results. The improvement in the case of 5 sensors with respect to 3 sensors per section was noticeable for all the measuring stations and all the deformations.

5.7. Results of the real-time co-simulation

This section presents the results of the algorithm applied to the Simulink model, which has been run in co-simulation with the MBDyn dynamic model of the circular beam. The beam has been subjected to a dynamic sinusoidal load, whose amplitude has been set to the large deformation load case no. 2, presented in Table 5.2. The simulation has been run for 40 seconds. During this time the load performed two periods.

Figure 5.52, Figure 5.53, and Figure 5.54 present the real-time plots of the displacements and twist reconstructed from the deformations received from the Simulink scopes and compared with the actual displacements of the MBDyn nodes. the Figures respectively show the trajectories of chordwise and flapwise displacement, and twist angle. The presented trajectories correspond to the reconstructed deformation at the location of the measuring stations. The displacement of the end node of the beam is also displayed. The displacement of the nodes at the measuring stations and the reconstructed displacements using algorithm are nearly coincident for all the stations. This shows that the algorithm works well also in its real-time implementation.

Comparing the presented results at $t = 5$ s clearly indicates that the results given by the algorithm applied to the real-time Simulink model and Matlab script (shown in section 5.3.3) are coincident.

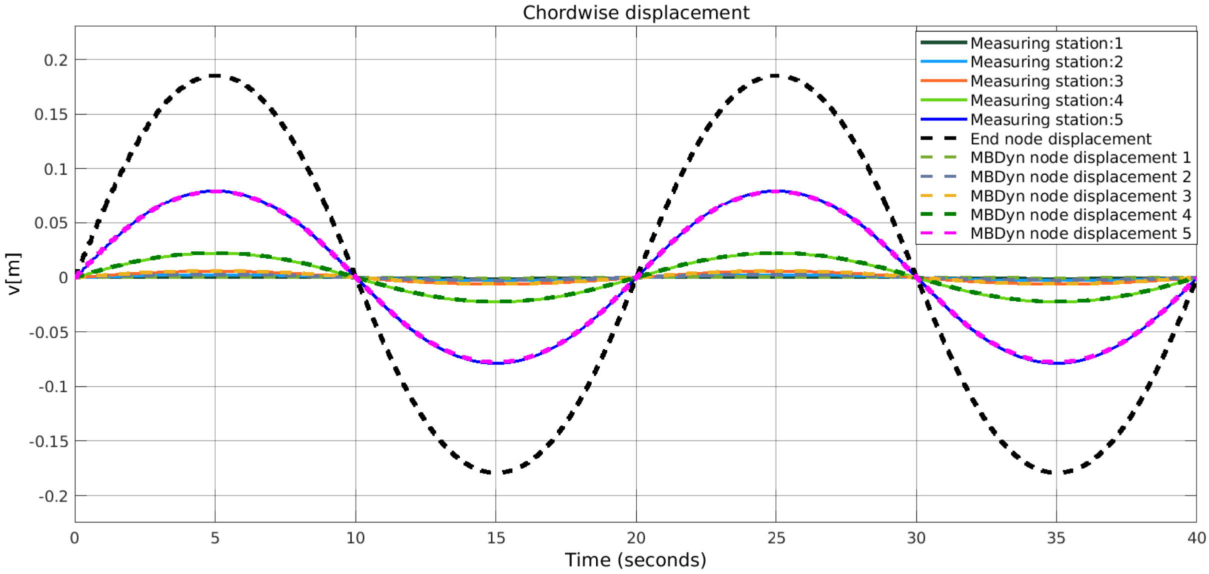


Figure 5.52: Chordwise displacement real-time results, circular beam

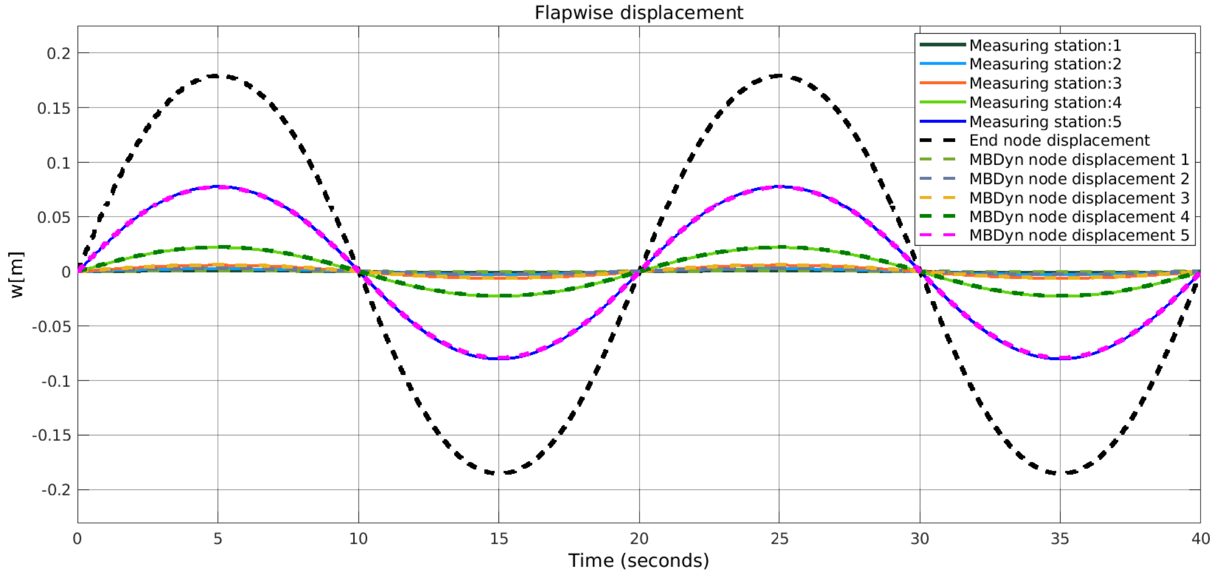


Figure 5.53: Flapwise displacement real-time results, circular beam

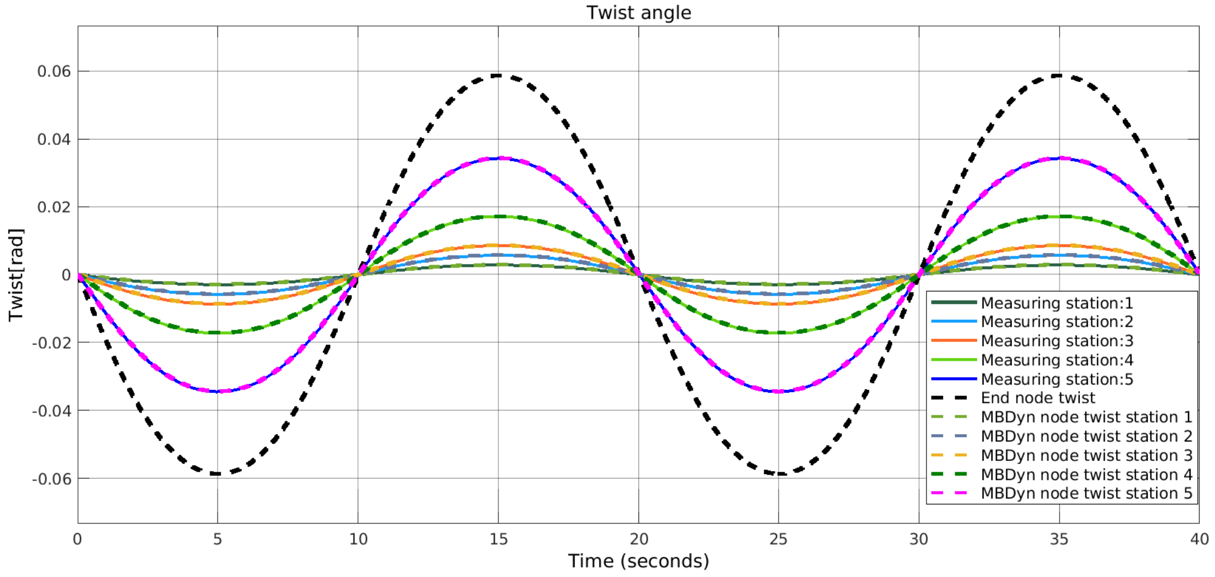


Figure 5.54: Twist angle real-time results, circular beam

6 | Algorithm validation using the Princeton beam experiment

This chapter shows the validation of the algorithm using the Princeton beam experiment described in [6]. The validation has been performed using the algorithm applied to the Matlab script and the real-time Simulink model. Two MBDyn models of the beam, a static and a dynamic one, have been also prepared and used later in co-simulation.

6.1. Description of the experiment

The Princeton beam experiment is a study of the large displacement and rotation behaviour of a simple cantilevered beam under a gravity tip load. In the prepared model, the gravity tip load has been replaced with a force applied to the tip of the blade which was changing its orientation around the spanwise axis of the beam. In the experiment, a straight beam made of aluminium alloy (T 7075) with a length $L = 0.508$ m and a rectangular cross-section with dimensions 3.175 mm × 12.7 mm was clamped at the root and subjected to three different loads applied to the tip. The loads were $P_1 = 4.448$ N, $P_2 = 8.894$ N, and $P_3 = 13.345$ N. They were applied to the beam under different loading angles in the range from 0° to 90°, the data was acquired for the loading angles $\theta_{load} = 0°$, 15°, 30°, 45°, 60°, 75°, and 90°. In the experiment, the tip end deflection along the chordwise and flapwise axes have been measured together with the twist of the tip [6].

In the presented verification model of the Princeton beam, the algorithm computes the deflection of the beam using 5 measuring stations with 3 sensors on each station; 2 strain gauges and 1 optical fibre sensor. Figure 6.1 presents the cross-section of the beam, together with sensors and tangent vectors used for the computations of the twist angle. In the results presented in this chapter, the strain from the MBDyn model have been applied to the algorithm. The MBDyn model of the beam consists 20 `beam3` elements, the measuring stations were located at the nodes located between two adjacent beam elements. Due to that, the algorithm does not reconstruct the deformation of the beam at the tip, the last station is located at the 88% of the beam spanwise length.

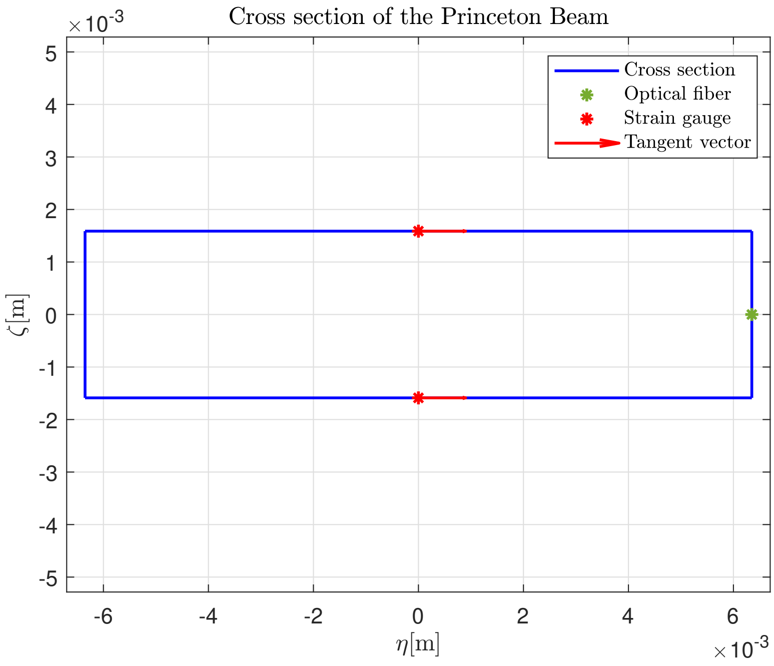


Figure 6.1: Cross-section of the Princeton beam

Along the beam a total of five measuring stations were chosen to improve the results of the shape reconstruction. The positions of the stations are shown in Table 6.1. In the results presented in section 6.2, the results from the algorithm are compared with the displacement of the nodes from the MBDyn simulation that are located at the measuring stations. The results from the MBDyn model have been validated using the Timoshenko beam model analytical formulas, for the loading angles $\theta_{load} = 0°$ and $90°$.

Measuring station no.	**Position along spanwise axis ξ[m]**
1	0.05
2	0.15
3	0.25
4	0.35
5	0.45

Table 6.1: Measuring stations along the Princeton beam

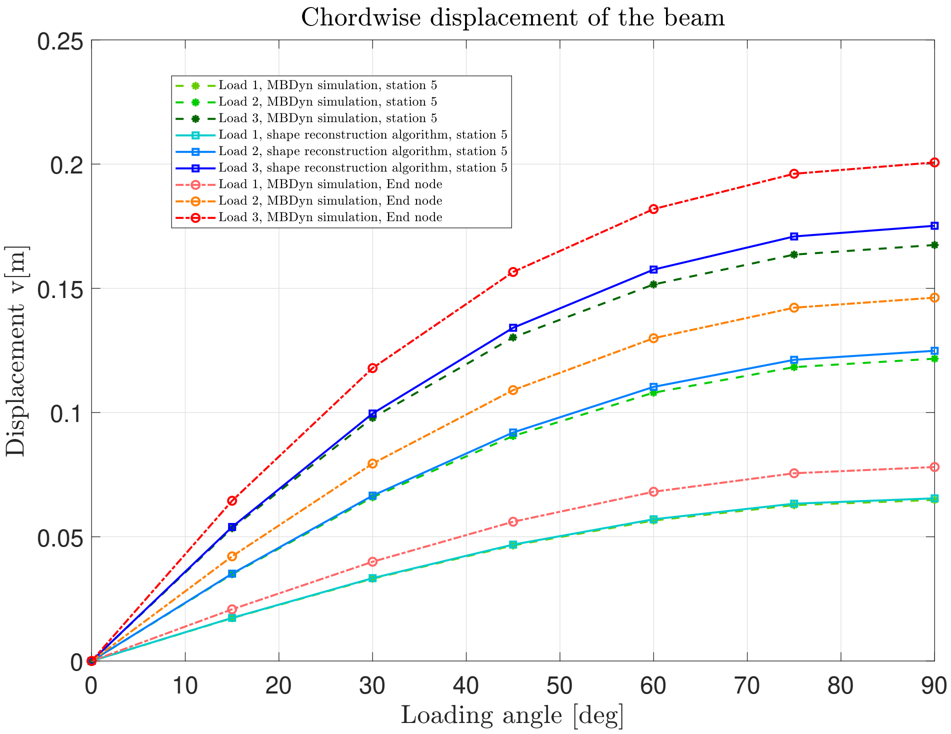
6.2. Shape reconstruction results

The comparison of the results from the algorithm, the MBDyn model, and the results from the paper are shown in this section. In the plots, there are shown results from the algorithm computed for the measuring station no. 5, together with the deformation of the node located at the same measuring station. For completeness, there is also shown the deformation of the end node of the beam.

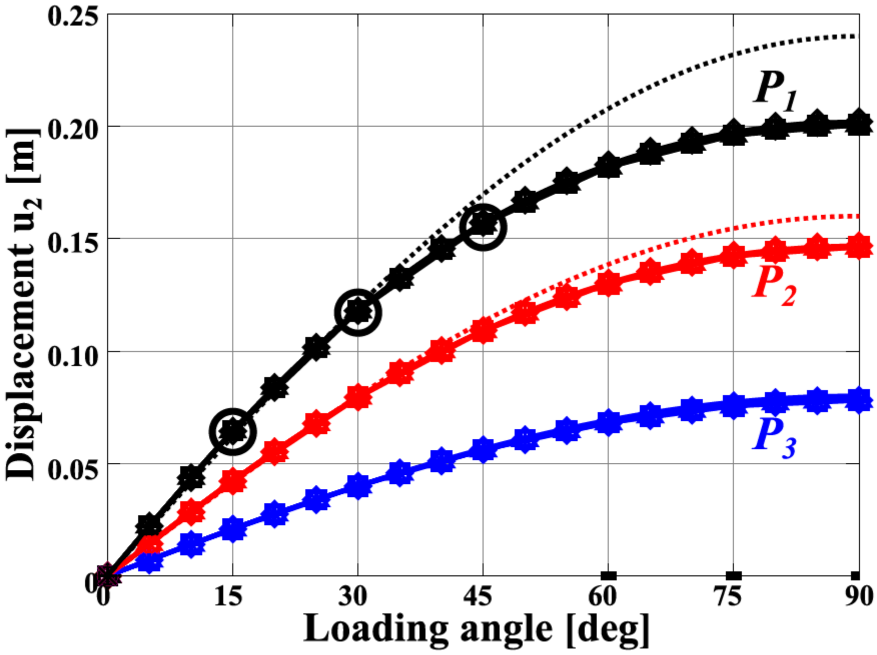
Figure 6.2 presents the results of the chordwise deformation. Comparing the results of the end node deformation shown in Figure 6.2a with the results from the paper shown in Figure 6.2b, the MBDyn model correctly reproduces the experiment. Also, the deformation at the measuring station no. 5 reconstructed using the algorithm sufficiently good reproduces the deformation of the node given by the MBDyn simulation. The algorithm for all of the loading angles gives result slightly larger than the deformation of the node. The greatest error is given by the load P_3 and loading angle 90°. The similar conclusions apply for the flapwise deformation that is presented in Figure 6.3a, however, in this case the largest discrepancy occurs for the loading angle of 45°. Nevertheless, the flapwise deformation is also correctly reproduced when compared to the results from the paper that are shown in Figure 6.3b.

Figure 6.4 presents the results of the twist angle. Also, the comparison between the end node twist, that is shown in Figure 6.4a, with the experimental results from Figure 6.4b proves that the multibody model correctly reproduces the results of the experiment. Comparison of the nodal twist at station no. 5, with twist that is reconstructed using algorithm proves that the algorithm correctly reproduces also the twist of the beam from the MBDyn model. In the case of twist angle, the results given by node twist are larger than the results from the algorithm, the greatest difference is present for load P_3 and loading angle of 45°.

The general conclusion is that having the strains from MBDyn simulation, the algorithm correctly reproduces the results of the Princeton beam experiment.

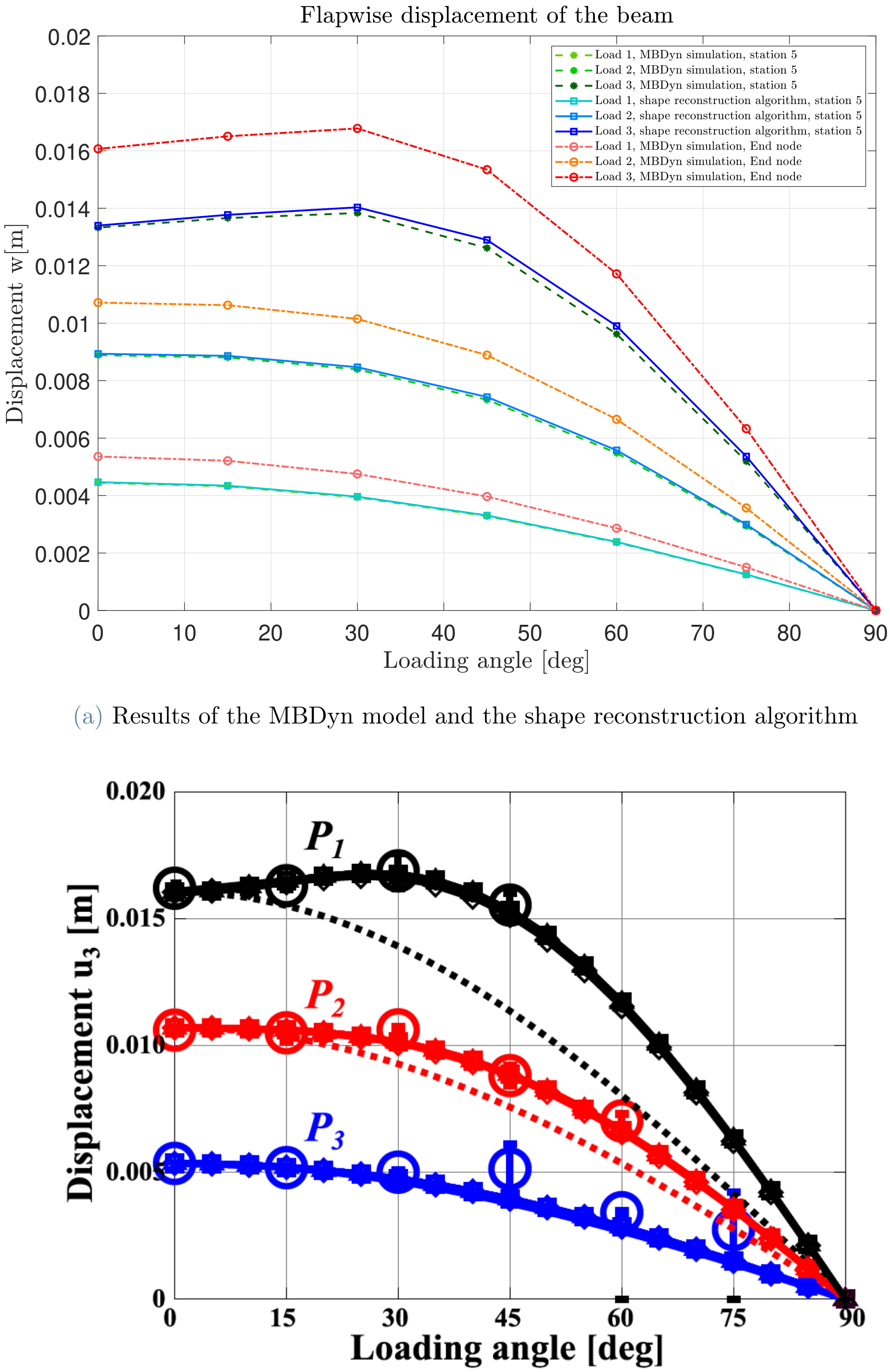


(a) Results of the MBDyn model and the shape reconstruction algorithm



(b) Deformation of the beam from the paper [6]

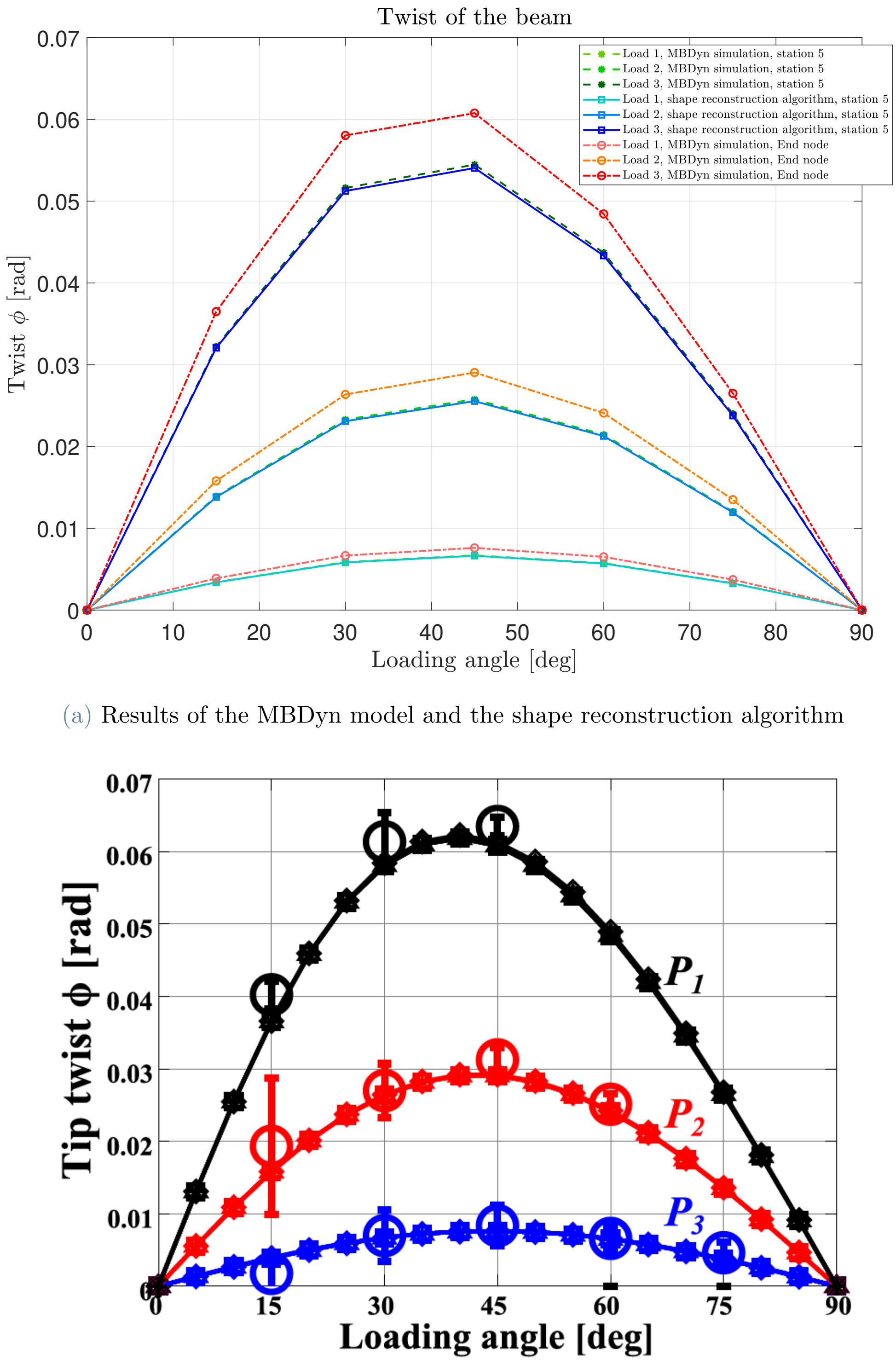
Figure 6.2: Chordwise deformation of the Princeton beam



(a) Results of the MBDyn model and the shape reconstruction algorithm

(b) Deformation from of the beam the paper [6]

Figure 6.3: Flapwise deformation of the Princeton beam

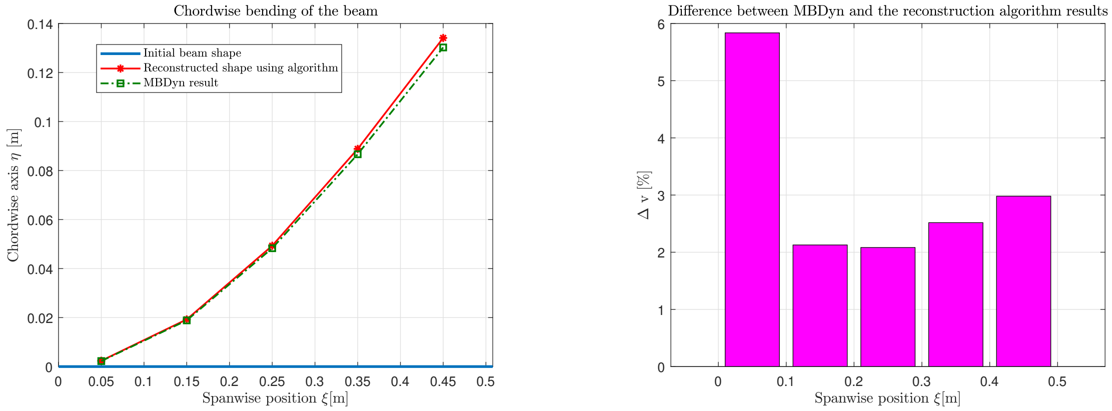


(a) Results of the MBDyn model and the shape reconstruction algorithm

(b) Deformation of the beam from the paper [6]

Figure 6.4: Twist angle of the Princeton beam

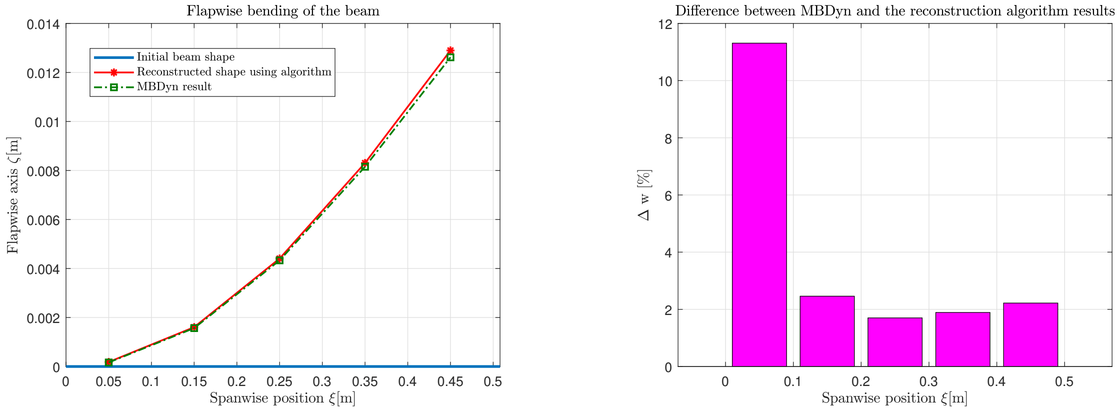
The load P_3 applied under the angle of 45° has been identified as the one that generates the largest discrepancy in results, the figures below present the beam deformation results for this case together with the difference between the MBDyn nodal deformation and the results given by the shape reconstruction algorithm. As it is shown in Figure 6.5a and Figure 6.6a, the algorithm results are slightly larger than the nodal displacement for all of the measuring stations. The discrepancies that are presented in Figure 6.5b and Figure 6.6b show that the greatest difference in results is given by the station that is close to the root of the blade, in the case of flapwise displacement it exceeds 10%, however, the discrepancy given by the station no. 5, that is close to the tip, is much smaller and does not exceed 3%.



(a) Chordwise deformation of the beam

(b) Difference between MBDyn and the shape reconstruction algorithm results

Figure 6.5: Chordwise deformation of the beam

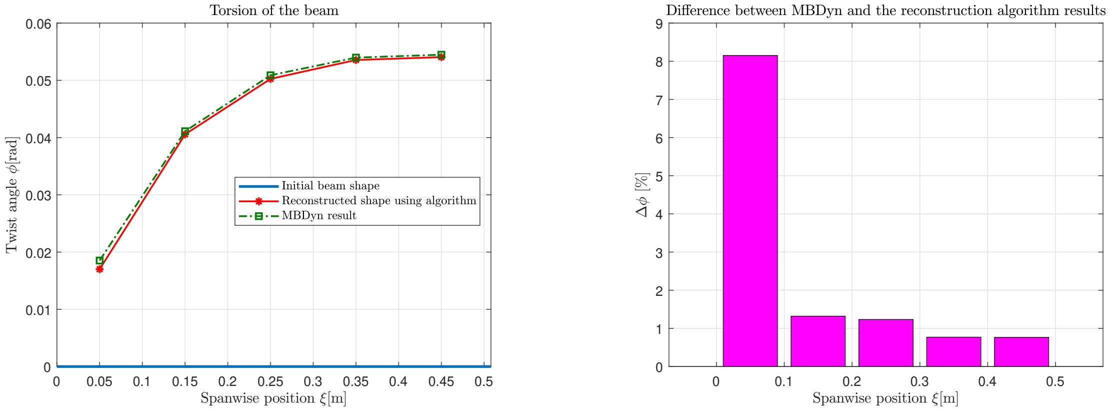


(a) Flapwise deformation of the beam

(b) Difference between MBDyn and the shape reconstruction algorithm results

Figure 6.6: Flapwise deformation of the beam

Figure 6.7 presents the twist reconstruction of the beam. As shown in Figure 6.7a the twist is not linearly distributed along the beam, also the twist given by the algorithm is smaller than the one taken directly from the node. Figure 6.7b presents the difference in the results from the algorithm and the MBDyn simulation, the smallest discrepancy is obtained for the station close to the tip (as in the case of bending) and the error at the station no. 5 does not exceed 1%.



(a) Twist angle of the beam

(b) Difference between MBDyn and the shape reconstruction algorithm results

Figure 6.7: Twist angle of the beam

6.3. Results of the real-time co-simulation

The real-time co-simulation of the Simulink model with the MBDyn dynamic model of the Princeton beam has been performed for load P_3 applied under the angle of 45°. The time duration of the simulation has been set to 10 seconds.

Figure 6.8, Figure 6.9, and Figure 6.10 show the reconstructed deformation of the beam from the measuring stations 1-5, the displacement of the nodes at the measuring stations, and end node displacement. As shown, the algorithm applied to the Simulink model correctly computes the deformation of the beam at the measuring stations, the reconstructed shape of the beam is nearly coincident with the displacement of the nodes. There is an difference between both, however, as it was discussed in section 6.2 it is not significant.

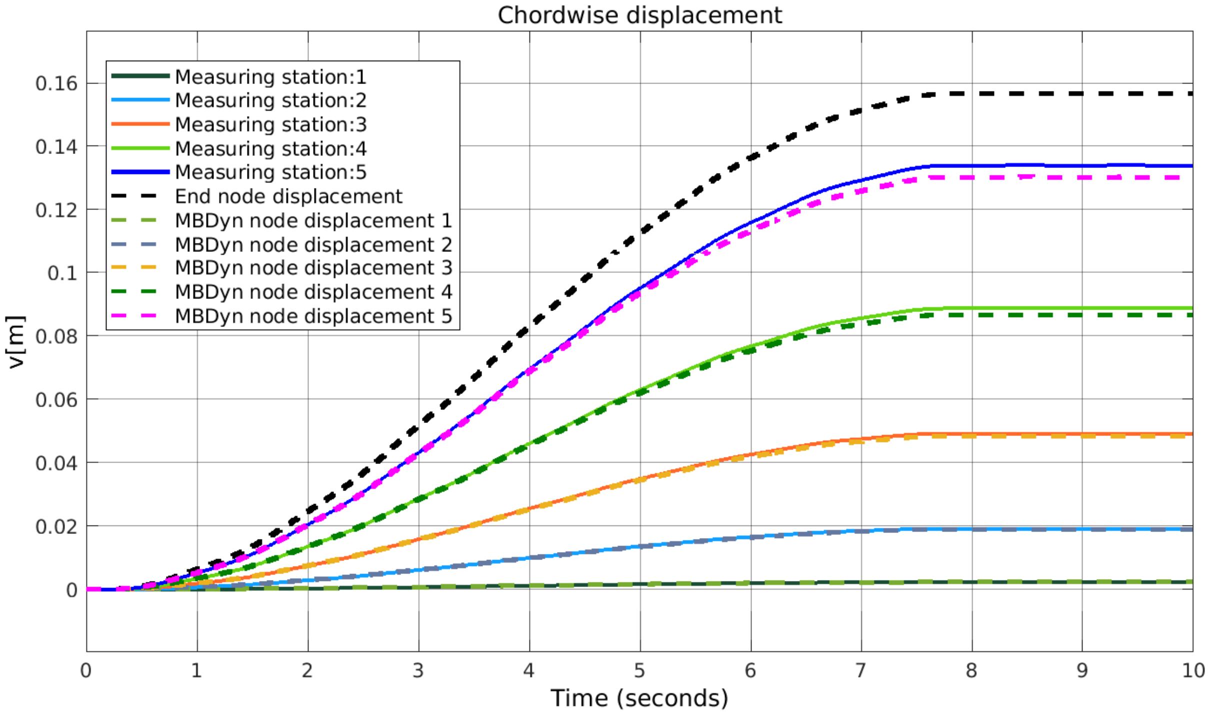


Figure 6.8: Chordwise displacement real-time result, Princeton beam

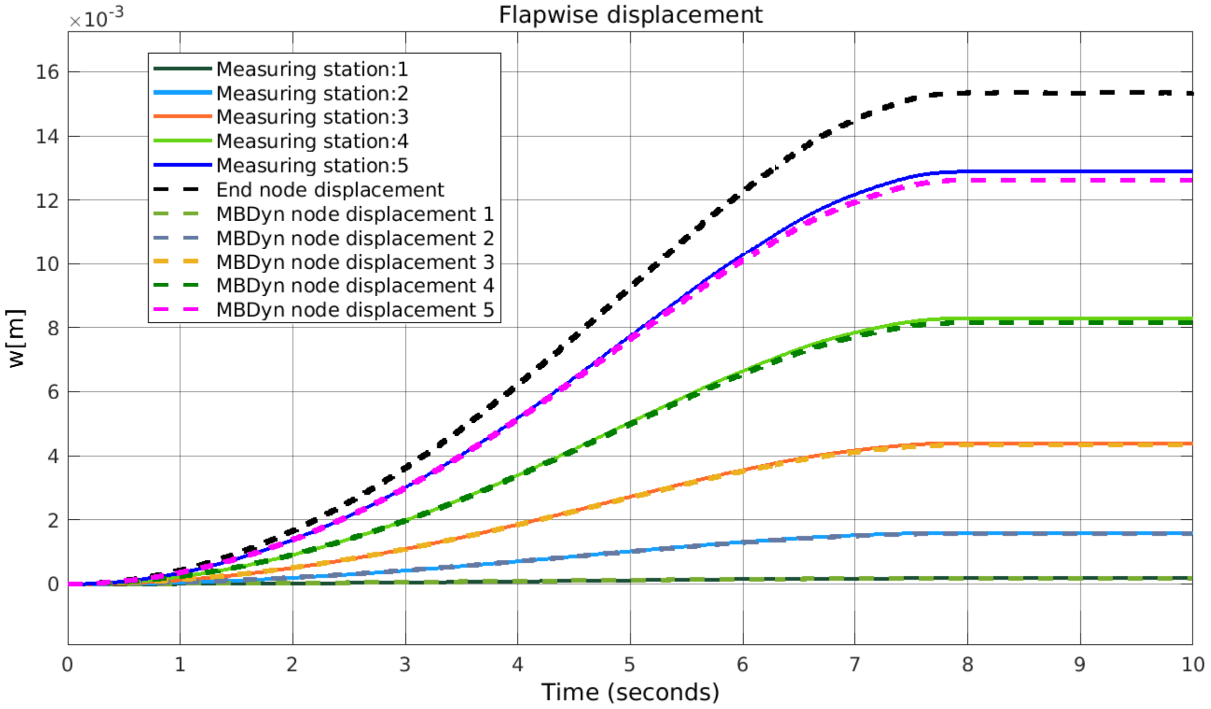


Figure 6.9: Flapwise displacement real-time result, Princeton beam

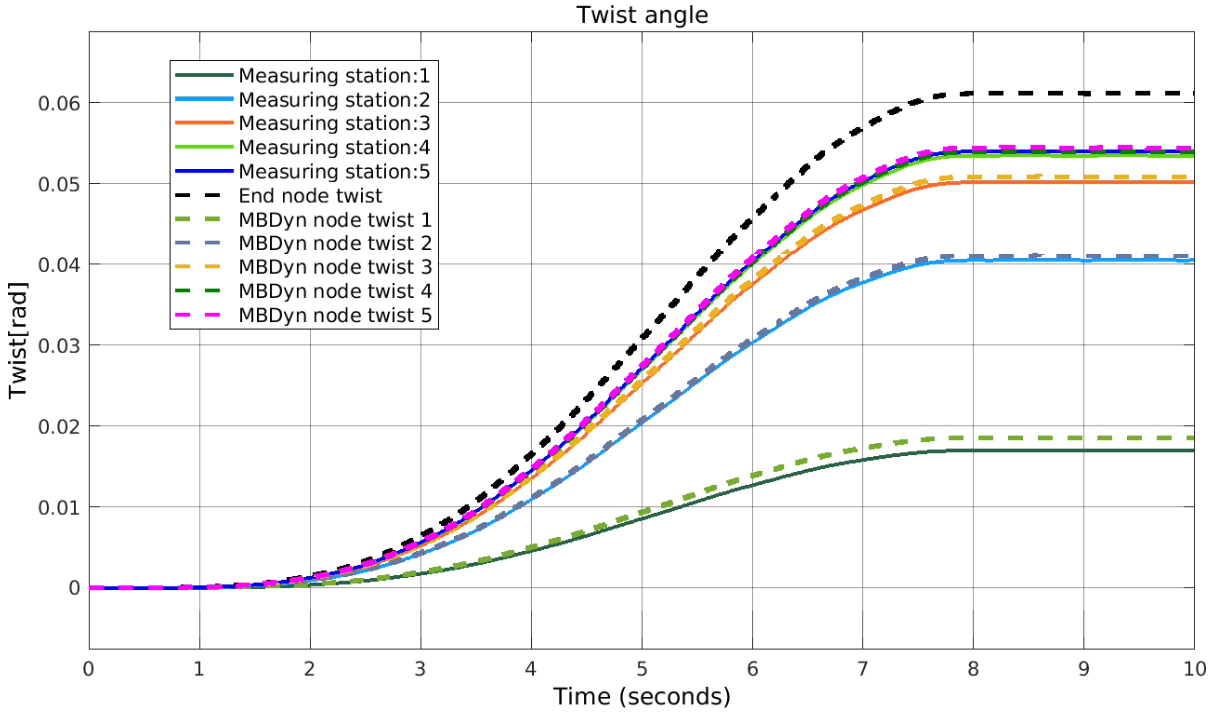


Figure 6.10: Twist angle real-time result, Princeton beam

7 | Application of the algorithm to the helicopter rotor model

This chapter presents an application of the real-time Simulink model for the shape reconstruction to a multibody model of the Messerschmitt-Bölkow-Blohm BO105. The algorithm is used to reconstruct the shape of one of the main rotor blades.

7.1. Description of the model

The multibody model of the helicopter is based on the data of the MBB BO105, which is a very interesting machine. When it was released in 1967 it was one of the most innovative helicopters. It was the first light twin-engine helicopter in the world. It also features a revolutionary hingeless design of the rotor system. In this rotor, the lead-lag motion is allowed through a friction component. Because this system generates a lot of heat, this solution has been used only in light helicopters, but it was later improved and applied to larger machines. Thanks to its innovative design, the BO105 is the first aerobatic helicopter in the world [25]. The helicopter is presented in Figure 7.1.

The multibody model used in this section has been developed for research in the field of rotorcraft-pilot coupling. It contains the multibody model of the main rotor, tail rotor, pilot, airframe, and control system. The chord of the blade is set to c = 0.27 m, and the length of the blade is L = 4.67 m. The radius of the rotor is R = 4.9 m, but the model accounts also for the radius of the rotor head. The blade is connected to the rotor head through a flex-beam; in the model, the flex-beam is considered part of the blade, just with different aerodynamic and structural properties. The rotor angular speed is Ω = 44.4 rad/s.

The model is based on data from the real helicopter, which makes it over-complicated for the purpose of the verification of the algorithm in application to a simple rotor model. Due to that, some simplifications have been applied, as listed below.

1. Twist is removed from the blades.

Figure 7.1: Messerschmitt-Bölkow-Blohm BO105 [25]

2. Gravity is removed from the model.
3. The helicopter is hovering, approximate thrust is equal to 2.2–2.4 kN.
4. The controls are simplified, and only collective control is applied.

Because till now the algorithm has been verified only using a uniform beam with a simple cross-section it was necessary to make some assumptions and simplifications also for the application of the algorithm, they are listed below.

1. The shape functions for the uniform beam are used in the algorithm. Thus, the shape functions from section 3.1 for bending and torsion are applied. In reality, the model of the helicopter blade contains a highly non-uniform model of the blade structure, with a complicated cross-section consisting of the NACA23012 airfoil.
2. The same bending shape functions are used for the chordwise and flapwise bending directions. This assumption is a significant simplification because as it is shown in section 3.2, where the numerical shape functions of the blade model were computed, the shape functions of a blade in chordwise and flapwise directions are different.
3. In the algorithm and the MBDyn model that computes strains, the cross-section of the blade is assumed to be rectangular, with dimensions $t = 0.1c$, $h = 0.5c$. The rectangle is centered at the elastic axis of the blade, located at 25% of the

chord from the blade leading edge. The real blade cross-section, together with the simplified one, is presented in Figure 7.2. The figure also shows the position of the strain gauges and the optical fibres. This assumption has been made for the reason that the algorithm has been already verified for a rectangular beam. The coefficients for re-computation of strains from the evaluation points to the location of the sensors at the cross-section are proved to be correct. In the airfoil-shaped beam, the coefficients for the formula Equation (2.32) and Equation (2.33) are not known and should be determined numerically, which is outside the scope of this thesis. This assumption affects the results of the strains from the MBDyn model and, in the algorithm, it influences the position of the sensors and tangent vectors used for the reconstruction of twist.

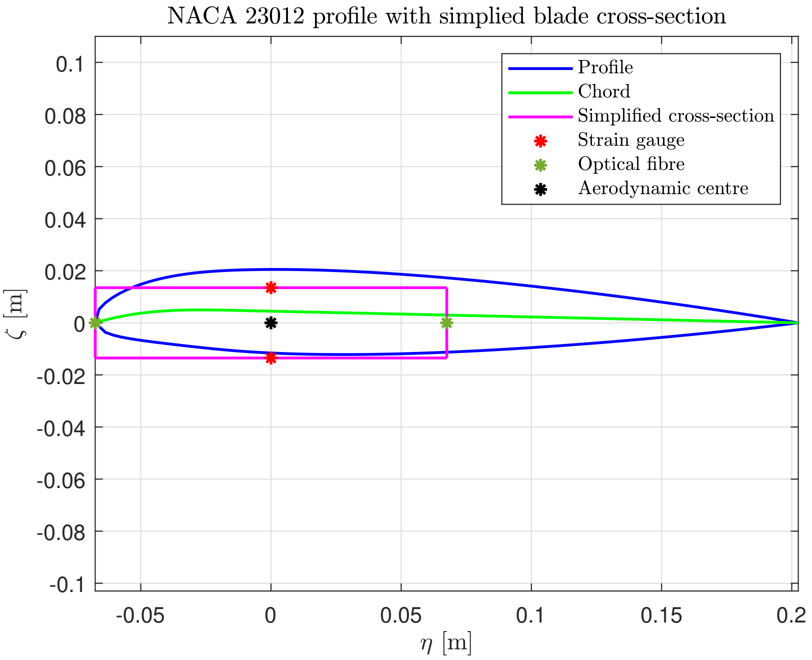


Figure 7.2: The cross-section of the modelled blade with the simplified, rectangular, cross-section

Taking into account all the simplifications and assumptions described above, the algorithm is not expected to obtain an exact reconstruction of the blade's shape. However, it is expected to obtain results that will be realistic and prove that the algorithm can be used in a real-time simulation of a sufficiently realistic rotor blade, and after removal of the assumptions will give much more accurate results.

To check that no other sources of error are present, a multibody model of the beam was created, with the same cross-section as the simplified blade and the same length as the helicopter blade. The beam was assumed to be uniform and it was simulated in real-time with an applied dynamic load. The results of the nodal displacements at the measuring stations were compared with the reconstructed displacements, proving in this way that if the assumptions are valid the algorithm correctly reproduces the bending and twist of the beam. Based on the results described in section 5.5, the number of mode shapes used in the reconstruction of the bending and torsion has been decreased to 3. This number of modes used in the reconstruction provided a sufficiently small error. The validated script was applied to the Simulink model used in the co-simulation with the BO105 multibody model.

Five measuring stations are placed along the spanwise direction of the blade. They are located on the nodes that are between two adjacent `beam3` elements. The exact locations of the measuring stations are shown in Table 7.1. At each station 4 sensors are placed; 2 strain gauges and 2 optical fibres.

Measuring station no.	**Position along spanwise axis ξ[m]**
1	0.93
2	1.91
3	2.50
4	3.49
5	4.27

Table 7.1: Measuring stations along the helicopter blade

7.2. Results of the real-time co-simulation

This section presents the results of the co-simulation between the MBDyn model of the rotor and the Simulink model of the algorithm for shape reconstruction of the blade. The algorithm has been applied to one blade; however, it can be easily extended to all 4 blades by adding additional channels to the stream output of the model. The figures below present the trajectory in time of the displacements reconstructed at the measuring stations (solid lines), as well as the displacement of the end node and of the nodes located at the stations (dashed lines).

Figure 7.3 presents the chordwise displacement results of the real-time simulation. The periodicity and the shape of the deformation are reconstructed very well; however, the magnitude of the displacements does not match; for all the stations, the computed displacement is smaller than the one from the nodes. The oscillations are as expected. After the initial phase of starting the rotor, the oscillations present at the beginning of the simulation disappear and the plots become steady.

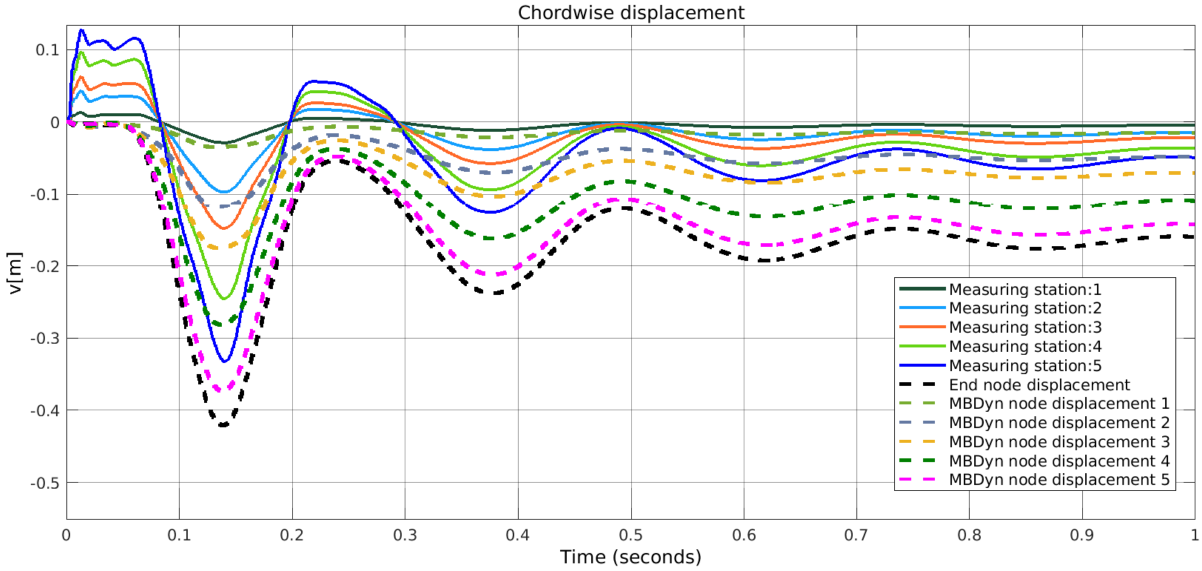


Figure 7.3: Real-time chordwise bending displacement of the blade

Figure 7.4 shows the flapwise displacement results of the real-time simulation. The periodicity of the oscillations and the shape of the displacements also in this case match. However, the displacement reconstructed for all the stations is much greater than that of the nodes. Also in this case, the oscillations present at the beginning disappear when the rotor reaches a steady condition.

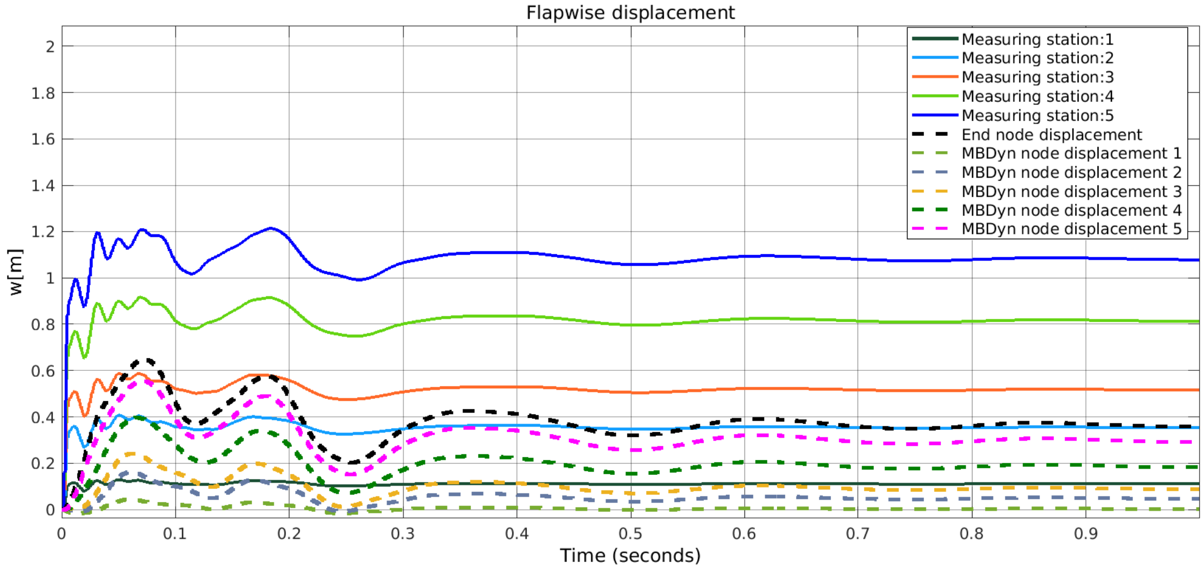


Figure 7.4: Real-time flapwise bending displacement of the blade

Figure 7.5 presents the reconstruction of the twist angle. In this case, the oscillations and the trajectory are also reconstructed with very good quality; however, the magnitudes do not match. The reconstructed twist is much greater than the twist from the nodes. This makes perfect sense because the sensors used for the reconstruction of the flapwise bending are the strain gauges, which are also used for the reconstruction of the twist. In both cases, the magnitudes of the reconstructed deformations are too large.

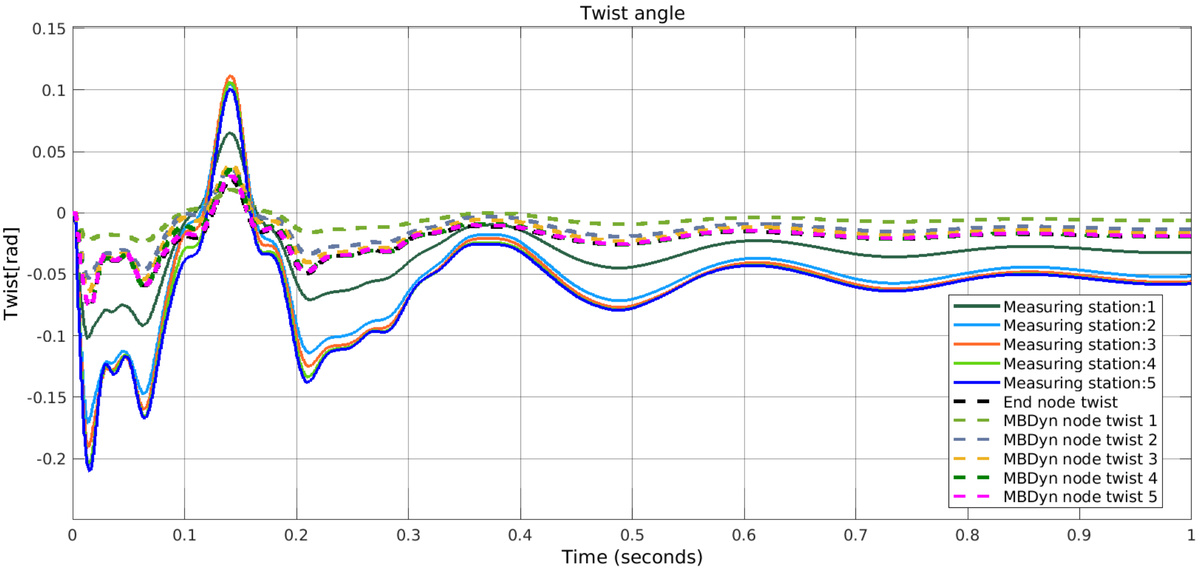


Figure 7.5: Real-time twist angle of the blade

The results show that the shape of the beam is reconstructed realistically but it is affected by a significant error, that is caused by the assumptions and simplifications described in detail in section 7.1. Especially important seems to be the simplification of the cross-section shape that affects the computations of the strains at the sensors, as well as the assumed shape functions. To mitigate those issues some coefficients have been introduced to the MBDyn model. The coefficients scale the strains computed by the software which are later sent to the Simulink model. The measurements of the optical fibers have been multiplied by a coefficient 1.3, while the measurements from the strain gauges have been multiplied by 0.48. The coefficients have been obtained by the trial and error method, and are not a solution to the problem. To get the best quality of the results, real coefficients for the re-computation of strains should be used together with proper shape functions.

Figure 7.6 presents the results of the chordwise deformation reconstruction with strain multiplied by the correction coefficient. The oscillations and shape of the trajectory are still correctly reconstructed from the strain measurements. Moreover, the magnitude of the displacements computed at the stations and nodal displacements match much better, they are nearly identical for all the stations.

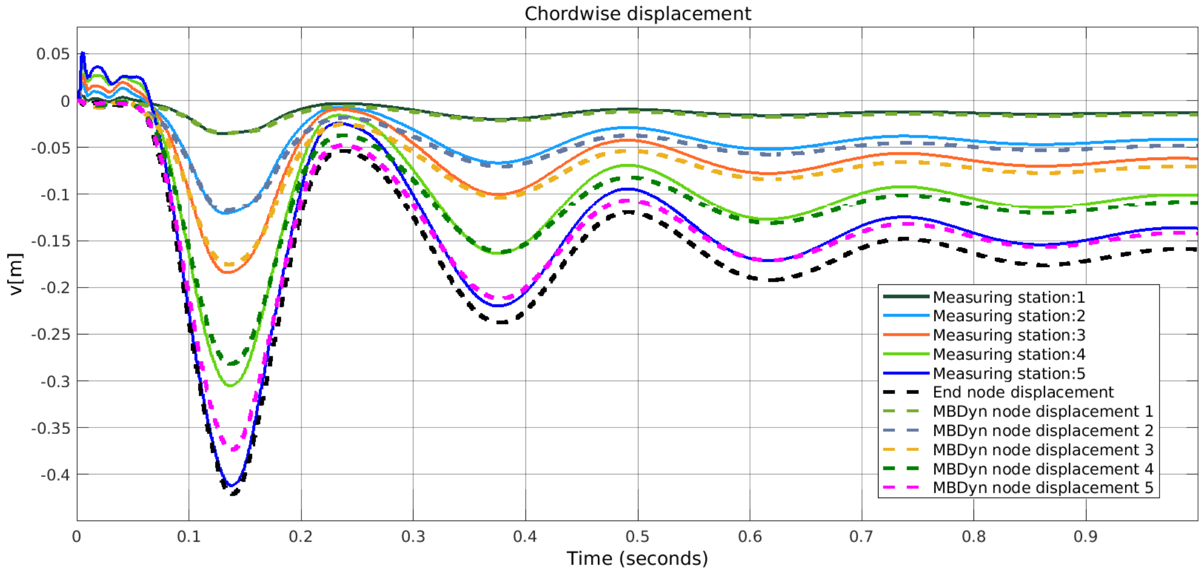


Figure 7.6: The real-time chordwise bending displacement of the blade with the correction coefficient

Figure 7.7 shows the results of the flapwise displacement reconstruction. The shape of the oscillations and periodicity are correctly reproduced. However, the magnitude of the reconstructed displacement and nodal deformation match only for the 5^{th} measuring station. In the case of the other stations the deformations from the algorithm is larger than the one taken directly from the node.

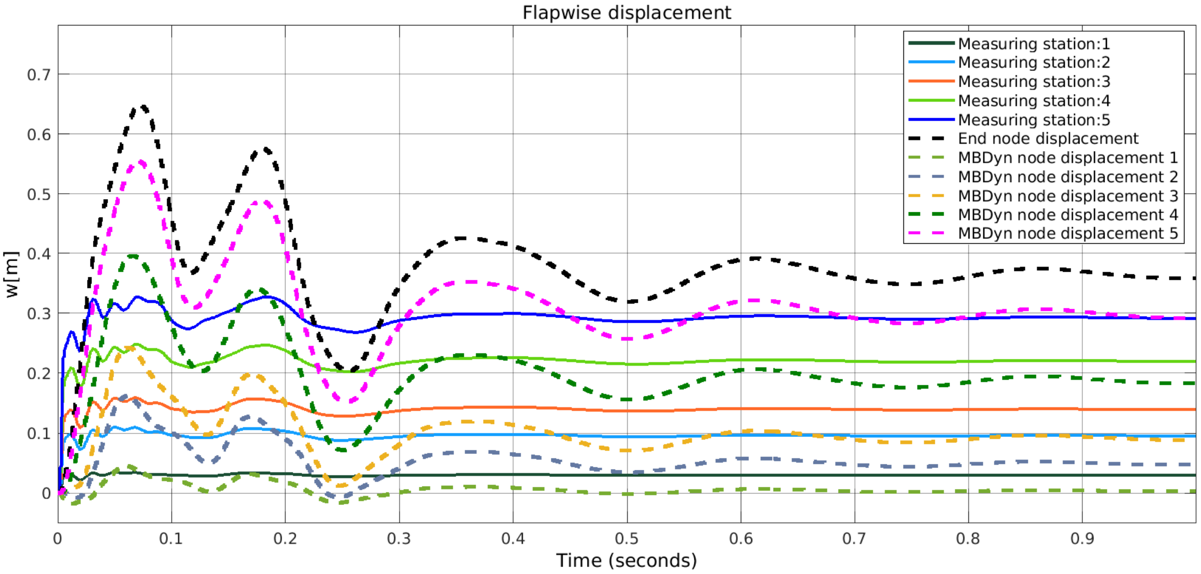


Figure 7.7: The real-time flapwise bending displacement of the blade with the correction coefficient

Figure 7.8 shows the reconstructed twist, the plot is split into subplots to make it more readable. The shape and the oscillations still match between the results from the algorithm and the nodal deformation. However, the magnitude of the reconstructed twist is lower than the twist taken directly from the simulation. This conclusion applies to all the stations.

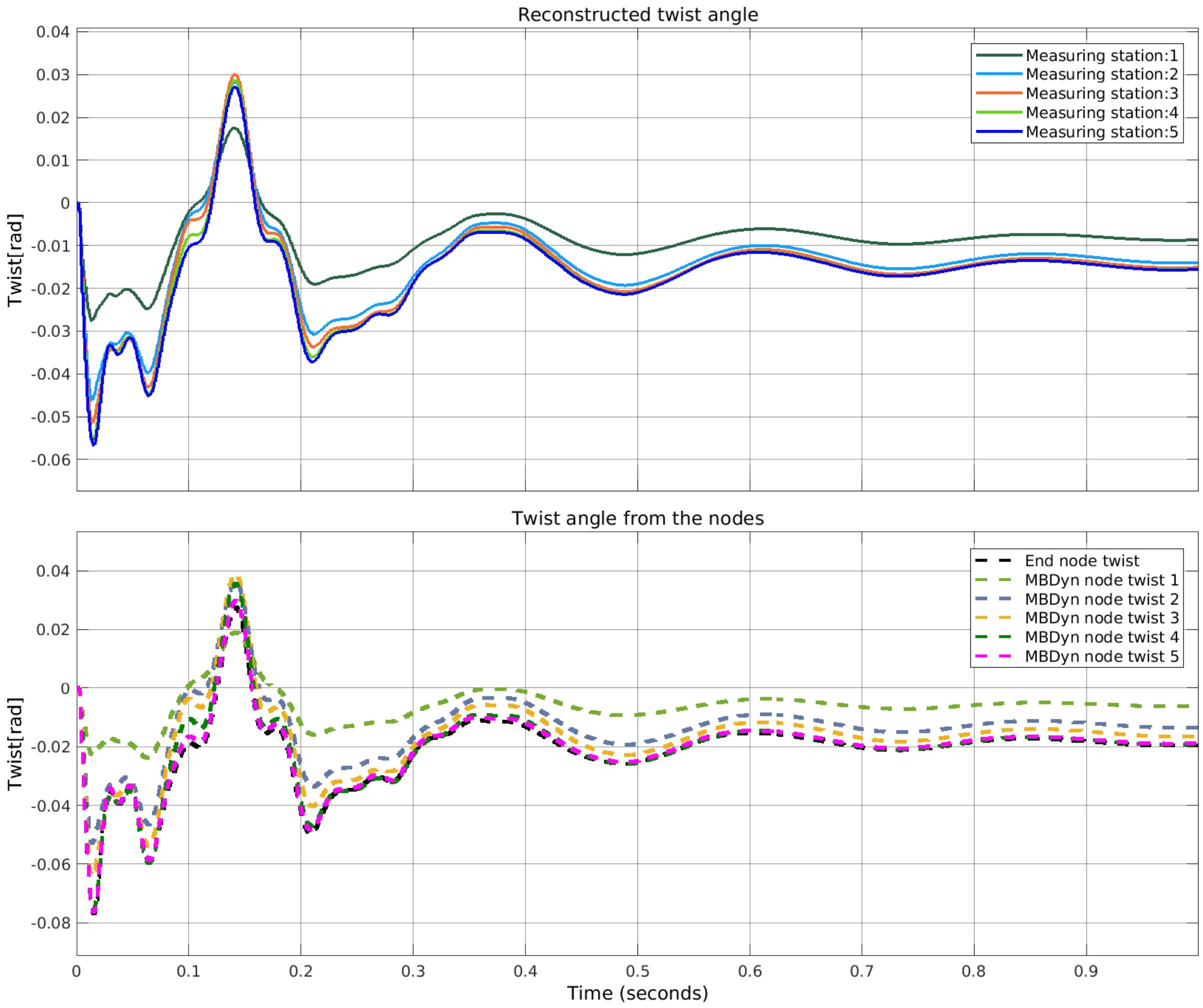


Figure 7.8: The real-time twist angle of the blade with the correction coefficient

Application of the correction coefficients to the strains taken from MBDyn has improved the matching between the reconstructed results and the nodal deformation taken from the MBDyn simulation. However, this solution is not sufficient especially in the case of the flapwise bending and twist computations. This is caused by the fact, that there are many assumptions and simplifications that result in the error, also the relation between the strains at the evaluation point and the surface of the blade is not straightforward. It involves three coefficients that multiply the strains and curvatures of the beam, all of the coefficients should be determined to obtain the best quality of the results.

8 | Conclusions and future developments

The main conclusion of the thesis is that the methodology proposed for the development of the real-time algorithm for shape reconstruction of a beam-like structure has been proved to be valid. Moreover, also the procedure for the computations of the numerical shape functions based on the results of the finite element method modal analysis proved its correctness.

The algorithm has been successfully validated using the beam with the circular-cross section. It has generated results that were similar to the reference results obtained using either the analytical formulas or simulation of the MBDyn model of the beam. The developed model has worked well irrespective of the source of strains, it has been tested using the analytical strains and the strains computed by the MBDyn software.

The algorithm has performed also very well in the case of the validation using the rectangular cross-section beam. It has correctly reproduced the results of the Princeton beam experiment using strains from the MBDyn simulation.

The real-time Simulink model, which has been used in co-simulation with the MBDyn beam models, has been also successfully validated. It has generated correct results which were compared with the nodal displacement from the MBDyn simulations.

Finally, the real-time algorithm has been applied to the multibody model of the BO-105 helicopter rotor. In this case, it has worked under significant simplifications and assumptions. Even though the obtained results have not perfectly reproduced the deformation of the blade, they can be considered as a realistic reconstruction of the blade shape. To make them more exact it is necessary to precisely determine the coefficients for strains computation, shape functions of the nonuniform blade, and use airfoil shape as a cross-section of the beam in the algorithm. The algorithm can be also easily extended to all 4 blades of the helicopter rotor model.

The thesis covered also a deeper analysis of the implemented modal approach and nonlinear equations for the reconstruction of a shape from strain measurements.

The solution provided by the algorithm has shown that its linearity depends on the source of strains and load case. The solution of the algorithm is always linear only if torque is applied to the beam, whereas in the case of application of transverse load the solution is linear only if analytical strains are used, in the case of MBDyn strains the linearity is lost at the same point as linearity of the MBDyn solution. The solution from the algorithm is linear up to the same point as the MBDyn simulation in the case of a complex load case with transverse forces and torque applied simultaneously.

Another interesting conclusion can be drawn from the relation between the quality of the results, the number of the mode shapes used in the reconstruction, and the number of the measuring stations along the blade. The best quality of the results is provided when the number of modes used in the reconstruction matches the number of the measuring stations. Moreover, the maximum number of the modes that can be used in the algorithm is equal to the number of measuring stations, more modes significantly decrease the quality of the results, making them not reliable. The more measuring stations, the more modes can be used, and better quality of the results is achieved. It is worth mentioning that in the considered results, idealised analytical strains have been used, measurement noise would additionally affect the quality of the shape reconstruction.

Considering the conclusions that are presented above and the results of the numerical shape functions determination it is possible to state that the reasonable number of modes used for the shape reconstruction should be around 3 modes. In this way, the high-frequency modes are neglected, and the quality of the results is sufficient, at the same time the computational effort is decreased.

In the thesis, it is also shown that if the measurements are affected by noise or any kind of error with a random origin, additional sensors on a cross-section might be considered as a method to alleviate the error and improve the quality of the results. This property has been proven for twist, as well as bending reconstruction.

In general, there were identified three ways to improve the quality of the results. The number of modes can be increased up to the number of the measuring stations, the number of measuring stations can be increased, or more sensors per section can be applied.

To sum up, the considered model applied to the real-time algorithm can be successfully used in the shape reconstruction of a beam-like structure. It is robust and gives a broad field for adjustments to balance the quality of the results with the computational effort and complexity of the equipment used for the measurements.

The work performed on the thesis leaves also a huge field for possible future developments.

At first, there can be developed a multibody model of the blade from the laboratory and used in a co-simulation with the Simulink real-time model of the algorithm. For this purpose, it is necessary to develop a stiffness matrix of the blade, which has a highly non-uniform structure. Exact coefficients should be found to simulate measurements of the sensors located at the surface of the blade from the axial and shear strains computed at the evaluation points of the beam element. Moreover, the numerical shape functions developed in the thesis should be used in the algorithm working in the co-simulation with the MBDyn model of the blade.

Another future work can be devoted to the improvements of the co-simulation of the Simulink model with the BO-105 rotor model. The numerical shape functions of the blade from the rotor should be developed, as well as coefficients for strains simulation at the location of the sensors. The model can be also easily extended to all 4 blades of the rotor and used for verification of how the algorithm behaves in the case of blade failure, thus in the SHM/HUMS applications.

The last possible future development that can be done based on the thesis, is related to the application of the real-time algorithm to the blade in the laboratory of Politecnico di Milano, which was the inspiration for this thesis. Many of the work is already covered, script for the determination of the sensors' position on the surface of the blade and tangent vectors is already written. As well as the numerical shape functions are already determined and presented in this thesis. The tasks that must be done are related to the optimization of the algorithm and implementation of the Simulink model to the C language so it can be used on the laboratory equipment, this can be done using the Simulink Coder package.

Summing everything up, the thesis has been successfully concluded, the algorithm has been developed and verified. The performed study also gives some insight into the behaviour of the algorithm using different parameters and settings. Moreover, there are many ways in which the job presented in this thesis can be used and developed in the future.

Bibliography

[1] Strain gauge principle, 2019. URL `https://www.variohm.com/news-media/technical-blog-archive/what-is-a-strain-gauge-`.

[2] FBG image, 2020. URL `https://hittech.com/en/portfolio-posts/noria-the-fiber-bragg-grating-manufacturing-solution/`.

[3] FBG principle, 2020. URL `https://fbgs.com/technology/fbg-principle/`.

[4] Strain gauge image, 2020. URL `https://www.michsci.com/what-is-a-strain-gauge/`.

[5] M. Alexander. Experimental flight test evaluation of the effects of rotor state measurements and feedback control on variable stability helicopters. page 15. European Rotrcraft Forum, National Research Council Canada, 9 2021.

[6] O. A. Bauchau, P. Betsch, A. Cardona, J. Gerstmayr, B. Jonker, P. Masarati, and V. Sonneville. Validation of flexible multibody dynamics beam formulations using benchmark problems. *Multibody System Dynamics*, 2016.

[7] G. Bernardini, R. Porcelli, J. Serafini, and P. Masarati. Rotor blade shape reconstruction from strain measurements. *Aerospace Science and Technology*, pages 580–587, 2018.

[8] D. di Scienze e Tecnologie Aerospaziali. MBDyn - MultiBody dynamics - homepage, 1996. URL `https://www.mbdyn.org/`.

[9] C. Enei, G. Bernardini, J. Serafini, L. Mattioni, C. Ficuciello, and V. Vezzari. Photogrammetric detection technique for rotor blades structural characterization. *Journal of Physics: Conference Series*, pages 658–673, 2015.

[10] G. L. Ghiringhelli, P. Masarati, and P. Mantegazza. Multibody implementation of finite volume c beams. *AIAA Journal*, 2000.

[11] D. H. Hodges and E. H. Dowell. Nonlinear equations of motion for the elastic bending

and torsion of twisted nonuniform rotor blades. Technical Report A-5711, NASA Ames Research Center, 12 1974.

[12] J. F. Horn, W. Guo, and G. T. Ozdemir. Use of rotor state feedback to improve closed-loop stability and handling qualities. *Journal of the American Helicopter Society*, pages 1–10, 2012.

[13] E. N. Jacobs and W. C. Clay. Characteristics of the NACA 23012 airfoil from tests in the full-scale and variable-density tunnels. Technical Report 530, National Advisory Committee for Aeronautics, 01 1935.

[14] H. Ji, R. Chen, and P. Li. Rotor-state feedback control design to improve helicopter turbulence alleviation in hover. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, pages 156–168, 2018.

[15] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 1963.

[16] P. Masarati. Dynamics and control of flexible aircraft, 2020.

[17] P. Masarati. Lecture notes for the subject: Rotorcraft design, 2020.

[18] P. Masarati. Multibody system dynamics, 2020.

[19] C. Maucher and F. Boden. Blade deformation measurements with IPCT on an EC 135 helicopter rotor. *Advanced In-Flight Measurement Techniques*, 2013.

[20] S. Panza and M. Lovera. Rotor state feedback in the design of rotorcraft attitude control laws. *Advances in Aerospace Guidance, Navigation and Control*, pages 205–225, 2015.

[21] R. W. Prouty. *Helicopter aerodynamics. 1 1*. Lulu.com, 2009. ISBN 978-0-557-08991-8.

[22] J. Serafini, G. Bernardini, R. Porcelli, and P. Masarati. In-flight health monitoring of helicopter blades via differential analysis. *Aerospace Science and Technology*, pages 436–443, 2019.

[23] L. Trainelli, M. Gennaretti, G. Bernardini, A. Rolando, C. E. D. Riboldi, M. Redaelli, L. Riviello, and A. Scandroglio. Innovative helicopter in-flight noise monitoring systems enabled by rotor-state measurements. *Noise Mapping*, pages 190–215, 2016.

[24] S. Weber, T. Kissinger, E. Chehura, S. Staines, J. Barrington, K. Mullaney, L. Z. Fragonara, I. Petrunin, S. James, M. Lone, and R. Tatam. Application of fibre optic

sensing systems to measure rotor blade structural dynamics. *Mechanical Systems and Signal Processing*, pages 107758–107774, 2021.

[25] C. work. Bolkow bo-105, 2021. URL `https://en.wikipedia.org/wiki/MBB_Bo_105`.

List of Figures

List of Tables

List of Symbols

Variable	Description	SI unit
$\boldsymbol{\xi}$	coordinate along elastic axis of the beam	m
$\boldsymbol{\eta}$	chordwise coordinate of the section	m
$\boldsymbol{\zeta}$	flapwise coordinate of the section	m
t	time coordinate	s
$\boldsymbol{X}$	spanwise coordinate(FEM model)	m
$\boldsymbol{Y}$	global chordwise coordinate	m
$\boldsymbol{Z}$	global flapwise coordinate	m
$\boldsymbol{\theta}$	built-in twist of the beam	rad
$\boldsymbol{v}$	chordwise displacement of the beam	m
$\boldsymbol{w}$	flapwise displacement of the beam	m
$\boldsymbol{\phi}$	twist of the blade	rad
$\boldsymbol{r}_v$	chordwise generalised coordinate	-
$\boldsymbol{r}_w$	flapwise generalised coordinate	-
$\boldsymbol{q}$	twist generalised coordinate	-
$\boldsymbol{n}_v$	matrix of chordwise shape function	m
$\boldsymbol{n}_w$	matrix of flapwise shape function	m
$\boldsymbol{n}_\phi$	matrix of torsional shape function	rad
$\boldsymbol{n}_b$	analytical bending shape functions	m
$\boldsymbol{n}_t$	analytical torsional shape functions	rad
$\boldsymbol{A_t, n}$	torsional shape function coefficients	-
$\boldsymbol{A_b, k}$	bending shape function coefficients	-
$\varepsilon_{\xi\xi_k}$	axial strain at the sensor k	-
$\varepsilon_{\xi\eta_k}$	shear strain along η axis at the sensor k	-
$\varepsilon_{\xi\zeta_k}$	shear strain along ζ axis at the sensor k	-
$\varepsilon_{\xi t_k}$	shear strain tangent to the beam surface	-

Variable	Description	SI unit
$\boldsymbol{\varepsilon}_x$	axial strain at the evaluation point of beam3 element	-
$\boldsymbol{\varepsilon}_{\xi\xi_{kI}}$	axial strain at the sensor k and evaluation point of beam3 element	-
$\boldsymbol{\gamma}_y$	shear strain at the evaluation point of beam3 element in Y direction	-
$\boldsymbol{\gamma}_z$	shear strain at the evaluation point of beam3 element in Z direction	-
$\boldsymbol{\varepsilon}_{\xi t_{kI}}$	shear strain at the sensor k and evaluation point of beam3 element	-
$\boldsymbol{\kappa}_x$	curvature about axis X at the evaluation point of beam3 element	-
$\boldsymbol{\kappa}_y$	curvature about axis Y at the evaluation point of beam3 element	-
$\boldsymbol{\kappa}_z$	curvature about axis Z at the evaluation point of beam3 element	-
$\boldsymbol{t}$	tangent vector to the cross section surface	-
$\boldsymbol{R}$	distance between strain sensor and elastic axis of the beam	m
$\boldsymbol{\Omega}$	angular speed of a rotor	$\frac{rad}{s}$
$\boldsymbol{df}_i$	inertia force acting on a blade element	N
$\boldsymbol{dq}$	aerodynamic force acting on a blade element	N
$\boldsymbol{df}_c$	centrifugal force acting on a blade element	N
$\boldsymbol{F}_z$	force acting along Z axis	N
$\boldsymbol{F}_y$	force acting along Y axis	N
$\boldsymbol{M}_t$	torque	Nm
$\boldsymbol{\sigma}$	axial stress	Pa
$\boldsymbol{\tau}$	torsional stress	Pa
$\boldsymbol{E}$	Young modulus of elasticity	Pa
$\boldsymbol{G}$	Shear modulus	Pa
$\boldsymbol{A}$	area of the cross section	m^2
$\boldsymbol{I_x}, \boldsymbol{I_y}$	moments of inertia	m^4
$\boldsymbol{J}$	polar moment of inertia	m^4
$\boldsymbol{L}$	length of a beam	m
$\boldsymbol{P}$	position vector	m
$\boldsymbol{\delta}$	difference between node and reference point position	m
$\boldsymbol{d}$	modal displacement vector of the reference point	m
$\boldsymbol{\Phi}$	modal displacement vector of the nodes	m
$\boldsymbol{\lambda}$	warping function	-
$()'$	first space derivative	m^{-1}
$()''$	second space derivative	m^{-2}

Variable	Description	SI unit
$()^+$	Moore-Penrose pseudoinverse	-
$()^m$	superscript of the mode number	-
$()_s$	subscript of the measuring station number along the beam	-
$()_k$	subscript of the strain gauge number	-
$()_j$	subscript of the section number	-
$()_i$	subscript of the node on the section j	-
$()_{jref}$	subscript of the reference point on section j	-
$()_n$	subscript of the beam element number	-
$()_{EX}$	subscript of the extrapolated value	-

Acknowledgements

I would like to express my special thanks and gratitude to my thesis supervisor Professor Pierangelo Masarati who gave me the opportunity to do this thesis on a project related to the shape reconstruction of helicopter blades. The door to the professor's office was always open whenever I run into trouble or had questions about the performed work. He consistently allowed this paper to be my own work but steered me in the right direction whenever I needed it.

I must express my profound gratitude to my parents, without them the accomplishment of the studies at Politecnico di Milano and the thesis at this excellent university would not have been possible. I would like to also thank the rest of my family and friends from Poland and Italy for providing me unfailing support and encouragement throughout the years of study.

Last but not least, I would like to express my recognition to the teachers and employees of Politecnico di Milano for sharing their knowledge and experience in order to let us, students, grow and become great specialists that will lead the world of the aerospace industry in the next decades.

Grazie!
Dziękuję!
Thank you!

Mateusz Krzysztof Garbacz

In the end, I would like to include a quote that was accompanying me through the process of the presented thesis development.

"Gli ostacoli sono troppo alti solo per chi non ha ambizioni abbastanza forti"
Stefano Pioli, allenatore AC Milan