

Towards the CSM-CFD modelling of membrane wings at high Reynolds numbers

J. Steiner

Master of Science Thesis

Towards the CSM-CFD modelling of membrane wings at high Reynolds numbers

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Aerospace Engineering at Delft
University of Technology

J. Steiner

January 31, 2018

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF
AEROSPACE ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of
Aerospace Engineering for acceptance a thesis entitled

TOWARDS THE CSM-CFD MODELLING OF MEMBRANE WINGS AT HIGH REYNOLDS
NUMBERS

by

J. STEINER

in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE AEROSPACE ENGINEERING

Dated: January 31, 2018

Thesis committee:

Dr.-Ing. R. Schmehl

Dr.ir. A. Viré

Dr.ir. A.H. van Zuijlen

Msc. N. Rajan

Abstract

The field of Airborne Wind Energy (AWE) is concerned with harvesting high-altitude wind power. While there exist a multitude of approaches towards harvesting the wind potential at high altitude, in the AWE group in Delft substantial research efforts have been put into accomplishing this using a kite connected to a fixed ground station equipped with a generator through its tether. Due to the high complexity that such a system ensues, it is of vital interest to have a fundamental understanding of the critical aero-elastic modes of the structure in flight for design and control of the device.

The aim of this project was to further extend on the existing research efforts in this area by coupling a high fidelity Reynolds-Averaged Navier-Stokes solver with an improved version of the existing structural solver framework and validate the model on simplified test cases using a partitioned approach. The developed methodology uses the open-source Computational Fluid Dynamics (CFD) solver foam-extend with an adapter to the coupling library preCICE as implemented in the FOAM-FSI toolbox for the fluid model. For the structure model, an in-house Python code based on a nonlinear shell element formulation is utilised.

A literature survey revealed that while there exists an abundance of publications on strongly coupled Fluid-Structure Interaction problems, the majority of them are targeted at applications set in completely different flow regimes. Thus, the most difficult part of the project was to find appropriate validation data for membrane wings at high Reynolds numbers. Nevertheless, the capabilities of the approach were demonstrated on three test cases.

First, the solver was applied to the classic FSI benchmark from Turek & Hron set in the laminar flow regime [1]. The relative error in the results for the relevant properties was at worst 10 %. The main source of inaccuracy is most likely the application of the thin shell element formulation to a comparatively thick beam. Thus, the error is expected to decrease when applied to thinner structures for which the shell model was developed for.

Finally, the solver was applied to two test cases with more realistic Reynolds numbers, where a thin, flexible material was wrapped around rigid leading and trailing edge supports and the force on the wing was measured for different membrane slack and angle of attack configurations [2, 3]. Comparison of the results from numerical models and experiments showed that the general trends for different slack lengths of the wing and angle of attack sensitivity were well captured in the numerical outcomes. However, the qualitative validation was less promising with a relative error in the mean lift and drag values of up to 15 % and 50 %, respectively. While some of the deviations can be traced back to ambiguities in the experiment description, the actual accuracy of the aeroelastic solver for membrane wings at high Reynolds numbers is yet to be determined.

Concluding, this project developed a methodology to model membrane wings at high Reynolds numbers. While the capabilities of the method have been successfully showcased on a classic FSI benchmark case, only a partial validation on benchmark cases at more realistic Reynolds numbers has been carried out. Hence going forward a full quantification of the accuracy of the method for its target application range is recommended.

Acknowledgements

First of all, I would like to thank my supervisor Axelle Viré for giving me the possibility to work on this very captivating project and supporting me throughout the process with problems related to my thesis and my career in general. Then, I would also like to extend my gratitude to Roland Schmehl for being on my thesis committee and for introducing me to the topic of aero-elastic modelling of membrane wings. A special thank you goes out to Alexander van Zuijlen for taking an interest in my thesis and aiding me with FSI related questions even though he was not supposed to be formally involved with my thesis as well as for being on my thesis committee.

Then, there were several PhD students involved in this project who generously shared their experience and time with me. Thank you, Mikko, Paul and Navi. Next, I would like to show my appreciation to all the other thesis students that worked at the Wind Energy research group for all the discussions related or also more importantly not-related to my thesis. It was a pleasure for me to work in such a multi-disciplinary group that allowed me to gain insight into a lot of other aspects of Wind Energy Engineering. Beyond that I found the entire group to be filled with passionate and kind people that were willing to help me out when asked.

Next, I am grateful for my friends outside academia back home and here for being there for me even though I have a bit of a knack for getting myself into difficult situations. Without their support, I would not have finished my thesis or made it through these past two and a half years. Lastly, I would like to thank my dad for always boundlessly supporting each and every one of my sometimes questionable decisions, especially during this past year.

Delft, University of Technology
January 31, 2018

J. Steiner

Table of Contents

Abstract	i
Acknowledgements	iii
List of Figures	ix
List of Tables	xi
Nomenclature	xiii
1 Introduction	1
2 Literature Review	3
2-1 Structural model	3
2-1-1 Governing equations	4
2-1-2 Discretisation	5
2-1-3 Relevant element formulations	6
2-2 Fluid model	9
2-2-1 Governing equations	9
2-2-2 Discretisation	10
2-2-3 Turbulence modeling	11
2-3 Immersed Boundary Treatment	13
2-4 Coupling	16
2-4-1 General classification	16
2-4-2 Different coupling schemes	17
2-4-3 Added mass effect	19
2-5 Verification	20
2-5-1 FSI benchmark cases	20
2-5-2 Membrane wings in literature	21
2-5-3 Proposed test cases	26
2-6 Available research	27

3	Structural solver	31
3-1	Shell element formulation	31
3-2	Detailed solver implementation	33
3-3	Static validation	38
3-3-1	Uniformly loaded membrane	38
3-3-2	Shell benchmark case	40
3-4	Dynamic validation	42
3-4-1	Linear spring-damper system	42
3-4-2	Linear cantilever oscillation	43
3-5	Turek & Hron Benchmark	45
4	Fluid solver	49
4-1	Model 1: Vortex particle method	49
4-2	Model 2: Immersed Boundary Computational Fluid Dynamics Approach	50
4-2-1	Implementation details	50
4-2-2	Validation	55
4-3	Model 3: Conventional Computational Fluid Dynamics Approach	62
4-3-1	Implementation details	62
4-3-2	Validation	64
5	Coupling algorithm	67
5-1	Methodology	67
5-2	Validation	69
6	Interface interpolation	71
6-1	Immersed Boundary technique	71
6-1-1	Mesh extrusion	71
6-1-2	Force projection	72
6-1-3	Velocity extrapolation	75
6-1-4	Validation	75
6-2	Body fitted mesh approach	75
6-2-1	Mesh extrusion	75
6-2-2	Force projection	76
6-2-3	Velocity extrapolation	76
6-2-4	Interface interpolation	76
7	Coupled simulations	79
7-1	Turek & Hron Benchmark	79
7-2	Greenhalgh setup	82
7-2-1	Setup and modelling simplifications	83
7-2-2	Mesh convergence	84
7-2-3	Mesh deformation	91
7-2-4	Choice of configurations and initialization of the simulation	92
7-2-5	Steady results	95
7-2-6	Unsteady results	95
7-3	Den Boer setup	101

8	Conclusions and Recommendations	105
8-1	Conclusions	105
8-2	Recommendations	107
A	Van Karman strain equations	109
B	Vortex Particle Method	111
B-1	Theory and implementation	111
B-2	Validation and convergence	112
C	Solver settings	115
C-1	OpenFOAM settings	115
C-1-1	Laminar cases	115
C-1-2	Steady RANS cases	116
C-1-3	Unsteady RANS cases	118
C-2	preCICE settings	122
D	Simulation results	125
D-1	Greenhalgh setup	125
	Bibliography	127

List of Figures

2-1	Literature Review: Membrane wrinkling illustration	7
2-2	Literature Review: FSI benchmark case	21
2-3	Literature Review: 2D clamped membrane wing	21
2-4	Literature Review: Experimental setup 2D clamped membrane wing	23
2-5	Literature Review: Experimental setup Fink	24
2-6	Literature Review: Experimental setup Den Boer	25
2-7	Literature Review: Membrane wing turbine blade	25
2-8	Literature Review: Kite Pumping cycle	28
2-9	Literature Review: Kite in flight	29
3-1	Structure model: Tiso shell element DOF	33
3-2	Structure Model Validation: Linear beam damping eigenvalue dependency	36
3-3	Structure Model: Convergence criteria	37
3-4	Structure Model Validation: Clamped membrane uniformly loaded	39
3-5	Structure Model Validation: Boundary conditions uniformly loaded membrane	39
3-6	Structure Model Validation: Pressure-Displacement plot	39
3-7	Structure Model Validation: Displacement plot	40
3-8	Structure Model Validation: Grid convergence	41
3-9	Structure Model Validation : Shell element	41
3-10	Structure Model Validation: Linear spring-damper setup	42
3-11	Structure Model Validation: One point mass system	43
3-12	Structure Model Validation: Two point mass system	43
3-13	Structure Model Validation: Linearized cantilever oscillation	44
3-14	Turek & Hron benchmark case: CSM1 mesh convergence	46
3-15	Turek & Hron benchmark case: CSM2 mesh convergence	46
3-16	Turek & Hron benchmark case: CSM3 mesh convergence	48

4-1	Turek & Hron benchmark case: CFD1 mesh convergence in Fluidity	57
4-2	Turek & Hron benchmark case: IB flow field detail in Fluidity	58
4-3	Turek & Hron benchmark case: CFD2 mesh convergence in Fluidity	60
4-4	Turek & Hron benchmark case: Time history profiles with mesh adaptivity and $CFL = 5$ in Fluidity	60
4-5	Turek & Hron benchmark case: Time history profiles with mesh adaptivity and $CFL = 0.5$ in Fluidity	61
4-6	Turek & Hron benchmark case: Mesh convergence in foam-extend	65
4-7	Turek & Hron benchmark case: CFD3 detail analysis in foam-extend	65
5-1	Coupling algorithm: Efficiency	70
5-2	Coupling algorithm verificaton: Lift hysteresis membrane wing	70
6-1	Interface interpolation: 2D extrusion mesh	72
6-2	Interface interpolation: 3D extrusion mesh	73
6-3	Interface interpolation: Shape function triangle	74
7-1	Turek & Hron benchmark case: FSI1 mesh convergence	80
7-2	Turek & Hron benchmark case: FSI3 mesh convergence	81
7-3	Turek & Hron benchmark case: FSI3 time series	82
7-4	Turek & Hron benchmark case: FSI3 flow field	83
7-5	Greenhalgh setup: Fluid mesh	85
7-6	Greenhalgh setup: Mesh convergence for round supports	87
7-7	Greenhalgh setup: Mesh convergence for oval supports	88
7-8	Greenhalgh setup: Mesh convergence without supports	88
7-9	Greenhalgh setup: Steady-state lift and drag polar for the undeformed profile	89
7-10	Greenhalgh setup: Steady-state mesh convergence	90
7-11	Greenhalgh setup: Mesh deformation	91
7-12	Greenhalgh setup: Lift and drag polar for the rigid wing	96
7-13	Greenhalgh setup: Limit cycle oscillation for a negative angle of attack	97
7-14	Greenhalgh setup: Loaded wing shape at moderate angles of attack	98
7-15	Greenhalgh setup: Unsteady coupled simulation results	99
7-16	Greenhalgh setup: Lift and drag polar	100
7-17	Den Boer setup: Fluid mesh	101
7-18	Den Boer setup: Coupled results	103
B-1	Fluid Model Validation: Steady Vortex Particle Method	113
B-2	Fluid Model Validation: Unsteady Vortex Particle Method	114
D-1	Greenhalgh setup: Flow field 1	125
D-2	Greenhalgh setup: Flow field 2	126
D-3	Greenhalgh setup: Flow field 3	126

List of Tables

2-1	Literature Review: FSI benchmark parameters	20
3-1	Structure Model Validation: Parameters	38
3-2	Structure Model Validation: Shell benchmark parameters	41
3-3	Structure Model Validation: Fixed parameters	44
3-4	Turek & Hron benchmark case: CSM Benchmark properties	45
3-5	Turek & Hron benchmark case: Summary CSM1 & CSM2	46
3-6	Turek & Hron benchmark case: Summary CSM3	47
4-1	Turek & Hron benchmark case: CFD settings	56
4-2	Turek & Hron benchmark case: CFD1 mesh adaptivity settings in Fluidity	57
4-3	Turek & Hron benchmark case: CFD2 mesh adaptivity settings in Fluidity	59
5-1	Coupling algorithm verification: Settings and properties	69
7-1	Turek & Hron benchmark case: Fluid and structure properties	80
7-2	Greenhalgh setup: Geometry	85
7-3	Greenhalgh setup: Mesh convergence	86
7-4	Greenhalgh setup: Inflow turbulence	89
7-5	Greenhalgh setup: Simulation parameters	94
7-6	Den Boer setup: Configuration	102
7-7	Den Boer setup: Rigid wing force coefficients	102

Nomenclature

Latin Symbols

$\mathbf{u}^{(f)}$	$= (u, v, w)$ Velocity of fluid particle	$[m/s]$
$\mathbf{u}^{(s)}$	$= (u, v, w)$ Structure displacement field	$[m]$
A	Area	$[m^2]$
C	Chord length	$[m]$
D	Damping matrix	$[kg/s]$
E_S	Young's Modulus	$[N/m^2]$
h_S	Shell thickness	$[m]$
K	Stiffness matrix	$[kg/s^2]$
M	Mass matrix	$[kg]$
p	Pressure	$[N/m^2]$
q	Node displacement	$[m]$
R	Residual of structural dynamic equation	$[N]$

Greek Symbols

α_s	Solid concentration IB method	$[1/m^3]$
χ_{ab}	Structure Curvature	$[-]$
ϵ_{ab}	Structure Strain	$[-]$
ϵ_{conv}	Scaling parameter for convergence of structural solver	$[-]$
ϵ_{wing}	Initial, relaxed excess length wing canopy	$[-]$

Γ	Vortex strength	$[m^2/s]$
ν_F	Kinematic viscosity fluid	$[m^2/s]$
ν_S	Poisson ratio	$[-]$
ω_a	Nth eigenmode of structure	$[rad/s]$
ρ_F	Fluid density	$[kg/m^3]$
ρ_S	Structure density	$[kg/m^3]$
ζ_n	Structural damping coefficient for mode n	$[-]$

Abbreviations

CFD	Computational Fluid Dynamics
CSM	Computational Structure Dynamics
FSI	Fluid-Structure Interaction
IB	Immersed Boundary
LEI	Leading Edge Inflated
NS	Navier-Stokes
RANS	Reynolds-Averaged Navier-Stokes
RBF	Radial Basis Function
VPM	Vortex Particle Method

a a a a a a a a

Chapter 1

Introduction

Due to the limited availability of fossil and nuclear fuels and their questionable environmental impacts, substantial research efforts have been made in the development of renewable energy systems in the past couple of decades. In particular the potential of conventional wind energy applications has been shown to be substantial as compared to the current energy demand [4, 5]. However, promising onshore wind locations are limited due to various factors and current developments are trending towards offshore applications due to higher wind speeds and less environmental constraints [6]. Alternatively, the idea to harvest energy from high altitude above $200m$ using airborne devices was first conceived by Lloyd in 1980 [7]. The increased energy potential at higher altitudes has been confirmed by various publications [5, 8, 9, 10]. Since then various Airborne Wind Energy (AWE) concepts have been conceived [11].

In the AWE group at TU Delft a substantial effort has been made towards harvesting high altitude wind power using different types of kites [12]. The chord-based Reynolds number of the kite at typical flight conditions can be placed in a range between about one and eight million. Moreover, over the course of the energy production cycle the wing is subjected to a large range of inflow angles ranging from attached to strongly separated flow. In addition to the complex flow field, the low mass and structural stiffness of the kite leads to large deformations of the wing that result in a very strongly coupled Fluid-Structure Interaction (FSI) problem. Observing the kite in flight, a few characteristic large and small scale deformation modes can be observed, some of which lead to the termination of the flight. Thus, an accurate aeroelastic model of the kite is expected to help with the understanding of its flight behaviour and help determine the onset of instabilities.

In particular, this project is focused on expanding the work previously done by Breukels (2010) [12], Bosch (2010) [13], Geschiere (2014) [14] and Berens (2015) [15] on developing an aeroelastic model of the kite in flight. Because one of the limitations of the previous setups was the low-fidelity aerodynamic model, this project aims at implementing and validating a coupled Computational Structural Model (CSM) and Computational Fluid Dynamics (CFD) model or in short a CSM-CFD solver.

However, given the complexity of the full AWE system, no attempt is made at modeling a fully realistic setup. Instead with the end aim in mind, a methodology is developed and validated step by step to satisfy the following research objective:

Further improve on the existing FSI solver by maintaining the existing structural solver methodology, coupling it to a higher fidelity CFD fluid solver, reviewing the time integration methodology of the structural solver and applying the new framework to a simplified test case. Construct the solver such that it can be used as a baseline for future projects in the AWE group; thus it should be easily extendable to more complicated test cases, and all its components are thoroughly validated.

To successfully comply with the research objective, the project is split into the following milestones. First, in chapter 2 an extensive literature review is carried out to assess existing methodologies and their suitability for this particular application, as well as to find suitable experimental and numerical reference data. Then, in chapters 3 and 4 the modeling approaches for structure and fluid are documented and briefly validated. For the structural model a nonlinear shell element formulation will be used, while three different fluid models are employed at different stages of the project. Next, chapter 5 details the coupling algorithm that links the structure and the fluid solver together. Subsequently, in chapter 6 the interface force and displacement interpolation is outlined and tested. Finally, chapter 7 puts all the previous pieces together. The full solver framework is first validated on a classical FSI benchmark case and then on two two-dimensional membrane wing profiles at more realistic Reynolds numbers. Lastly, in chapter 8 the project is concluded by summarizing the obtained results, pointing out the strong and weak points of the research and making recommendations for the future development of the solver.

Chapter 2

Literature Review

This chapter comprises a general literature review on high fidelity modelling of the Fluid-Structure Interaction of highly flexible wings. It is kept quite general, a more detailed description of the applied modelling approaches will be given at the beginning of the following chapters.

The first section 2-1 will outline general modeling approaches for the structure whereas the second section 2-2 will outline general modeling approaches for the fluid. The following section 2-3 will detail the boundary treatment for the fluid model in more detail. Then, section 2-4 will highlight suitable coupling algorithms. Subsequently, section 2-5 will showcase references from literature that can possibly be used to validate the methodology to be developed within this project. Lastly, section 2-6 will briefly summarize previous aero-elastic models that have been developed within the AWE group in Delft with the aim of pointing out potential issues and an existing body of work that can be built upon.

2-1 Structural model

This section will briefly touch upon the application of Finite Element methods to high fidelity modelling of structural dynamics. The first subsection 2-1-1 will outline the governing equations as derived from continuum mechanics. In the following, subsection 2-1-2 highlights how the equations are discretised. Lastly, in subsection 2-1-3 modelling approaches suitable for a membrane wing with and without inflated components will be described in more detail.

2-1-1 Governing equations

The governing equations for a structure in a Lagrangian framework enforcing a dynamic force equilibrium at every point in the continuum are

$$\sigma_{x,x} + \tau_{xy,y} + \tau_{xz,z} + f_x = 0 \quad (2-1a)$$

$$\sigma_{y,y} + \tau_{yx,x} + \tau_{yz,z} + f_y = 0 \quad (2-1b)$$

$$\sigma_{z,z} + \tau_{zy,y} + \tau_{zx,x} + f_z = 0 \quad (2-1c)$$

where the dynamic terms are included in the volume forcing term f_i alongside with other volume forces, e.g. gravity. In direct correspondence with the continuum equations, the Cauchy stress tensor is defined as,

$$T_{ij} = \sigma_{ij} \quad (2-2)$$

where a stress is represented relative to its present configuration. Conversely, the second Piola-Kirchhoff tensor can be introduced which expresses the stresses in a deformed configuration \mathcal{D} relative to an (undeformed) reference configuration \mathcal{R}

$$\frac{\mathcal{D}}{\mathcal{R}}S = \det FF^{(-1)}TF^{(-1)T} \quad (2-3)$$

where $\frac{\mathcal{D}}{\mathcal{R}}F_{ij} = \frac{\partial^{\mathcal{D}}x_i}{\partial^{\mathcal{R}}x_j}$ refers to the deformation gradient tensor that connects the two reference frames and T is still the Cauchy stress tensor. As a consequence, for a small deviation between the two reference frames (or small deformations) the Cauchy and the second Piola Kirchhoff tensor are identical.

The structure continuum mechanics equations are not solved for the material stresses but for the displacement field of the structure and the time derivatives thereof. Similar to the stress tensor formulation different strain tensor formulations exist. The linear Cauchy's strain tensor is given through

$$e_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) \quad (2-4)$$

whereas the nonlinear Green-Lagrange tensor with nomenclature analogous to the Piola-Kirchhoff stress tensor is defined as

$$\frac{\mathcal{D}}{\mathcal{R}}\epsilon_{ij} = \frac{1}{2} \left(\frac{\mathcal{D}}{\mathcal{R}}u_{i,j} + \frac{\mathcal{D}}{\mathcal{R}}u_{j,i} + \frac{\mathcal{D}}{\mathcal{R}}u_{k,i} \cdot \frac{\mathcal{D}}{\mathcal{R}}u_{k,j} \right) \quad (2-5)$$

where the nonlinear formulation reduces to the Cauchy strain tensor for small deformations.

The two stress and strain tensor formulations have been introduced such that the internal work in the principle of virtual work can be expressed as

$$\int_{tV} \frac{\mathcal{D}}{\mathcal{R}}\tau_{kl}\delta_{\mathcal{D}}e_{kl}d^{\mathcal{D}}V = \int_{\mathcal{R}V} \frac{\mathcal{D}}{\mathcal{R}}S_{ij}\delta_{\mathcal{R}}^{\mathcal{D}}\epsilon_{ij}d^{\mathcal{R}}V \quad (2-6)$$

where the virtual displacements are given by $\delta_{\mathcal{D}} e_{kl} d^{\mathcal{D}}$ and $\delta_{\mathcal{R}}^{\mathcal{D}} \epsilon_{ij}$, respectively.

Now the analogy between the two formulations becomes obvious. Both of these formulations were introduced such that now the Total Lagrangian framework for nonlinear problems with large deformations can be formulated. Assuming an initial reference configuration $\mathcal{R} = 0$ is given as well as the deformed configuration at time $\mathcal{D} = t$ and the determination of the stress and strain fields at time $t + \Delta t$ is of interest, the following decomposition can be made for the displacement and the stress field

$${}_{0}^{t+\Delta t} u_{i,j} = {}_{0}^t u_{i,j} + {}_{0} u_{i,j} \quad (2-7a)$$

$${}_{0}^{t+\Delta t} \epsilon_{ij} = {}_{0}^t \epsilon_{ij} + {}_{0} \epsilon_{ij} \quad (2-7b)$$

$${}_{0}^{t+\Delta t} S_{ij} = {}_{0}^t S_{ij} + {}_{0} S_{ij} \quad (2-7c)$$

Such that the principle of work for $t + \Delta t$ reduces to a variation around the configuration at time t according to

$$\int_{0V} {}_{0}^{t+\Delta t} S_{ij} \delta_{0}^{t+\Delta t} \epsilon_{ij} d^0 V = {}^{t+\Delta t} R \quad (2-8)$$

where R refers to the external virtual work.

Lastly, apart from nonlinearity due to large displacements, nonlinearity in the structural response can also arise from nonlinear material behaviour or also due to nonlinear boundary conditions or external forces. If the material behaviour is linear-elastic, the constitutive relation between stress and deformation can be expressed with the same tensor independent of the framework as

$${}^t \sigma_{ij} = C_{ijrs} {}^t e_{rs} \quad (2-9a)$$

$${}_{0}^t S_{ij} = {}_{0}^t C_{ijrs} {}_{0}^t \epsilon_{rs} = C_{ijrs} {}_{0}^t \epsilon_{rs} \quad (2-9b)$$

2-1-2 Discretisation

For the discretisation of the virtual work principle, the domain is divided into multiple elements. For each element, the virtual work principle can then be reinterpreted as a weak finite element formulation with the virtual displacements taken to be the shape functions used to distribute the nodal properties.

For each element node either translational or rotational degrees of freedom are defined. If the interpolation functions for the discretisation of the structure correspond with the interpolation functions for the degrees of freedom, then the finite element formulation is referred to as isoparametric.

If the Total Lagrangian framework is used then after assembly of the finite element matrices, the nonlinear system of equations can be summarised as

$$M^{t+\Delta t} \ddot{\mathbf{q}} + \left({}_{0}^t K_{\text{linear}} + {}_{0}^t K_{\text{nonlinear}}(\mathbf{q}) \right) \mathbf{q} = {}^{t+\Delta t} R - {}_{0}^t F \quad (2-10)$$

where the damping terms were neglected but can be added as well if desired. The forcing term t_0F on the right-hand side, as well as the linear and nonlinear stiffness matrices t_0K , are a result of the linearization around time t in the Total Lagrangian framework. For the detailed derivation of the system matrices, see Bathe (2007) [16].

For the time integration of the system, in principle any time integration scheme can be used. However, in general, the stable integration of the dynamics of a structure using an explicit time scheme, requires a very small time step, especially for stiff structures. Thus unconditionally stable implicit or conditionally stable semi-implicit methods are preferred. This holds true in particular for nonlinear problems where the natural frequencies of the structure depend on the current deformation and thus the critical time step is not known a priori. While for implicit time integration a number of schemes have been developed, some of them require the evaluation of the external forces at intermediate time steps, which is in general not the practice for FSI solvers [16].

2-1-3 Relevant element formulations

Based on the previous kite structural modelling approach by Bosch [13] and Berens [15], relevant finite element formulations are shortly elaborated on with the intent of pointing out the main characteristics and potential model shortcomings.

Membrane elements

Membrane elements are used to represent structures that have no bending stiffness and thus the stress field, and as a consequence also the strain field will be planar. Depending on the order of the element the nodal degrees of freedom either just correspond to two in-plane translations or additionally also one drilling rotation perpendicular to the element plane. A more detailed overview can be found in Felippa (2003) [17].

For curved membrane elements, a phenomenon referred to as membrane locking can occur, where the bending stiffness of the element is significantly overpredicted by the Finite Element Analysis [18]. The cause is that any change in curvature will also result in a change of the length of the midline of the element, which then results in a change of the bending deformation energy of the element.

Because a membrane has a very low bending stiffness, it will buckle under very low compressive loads. However, a simplified finite element membrane formulation where the bending stiffness is assumed to be zero will not be able to predict wrinkling (or buckling) phenomena. In fact, the model will show unnatural compressive stresses. However, if a shell element with a low bending stiffness and sufficient resolution is used wrinkling phenomena can be captured. Albeit this requires a very fine mesh resolution. See also figure 2-1 for illustration. [19]

Alternatively, coarser grids can be employed in combination with subgrid scale wrinkling models. Two main approaches can be distinguished. First, the membrane deformation gradient can be modified to incorporate the effect of wrinkling, see for example Schoop et al. (2002) [20]. Second, the constitutive relations can be modified in wrinkled regions to eliminate compressive stresses, see for example Barsotti and Ligaro (2014) [21] or Rossi et al. (2005) [22]. However, the introduction of wrinkling models can make the solution procedure unstable and thus appropriate stabilisers need to be used.

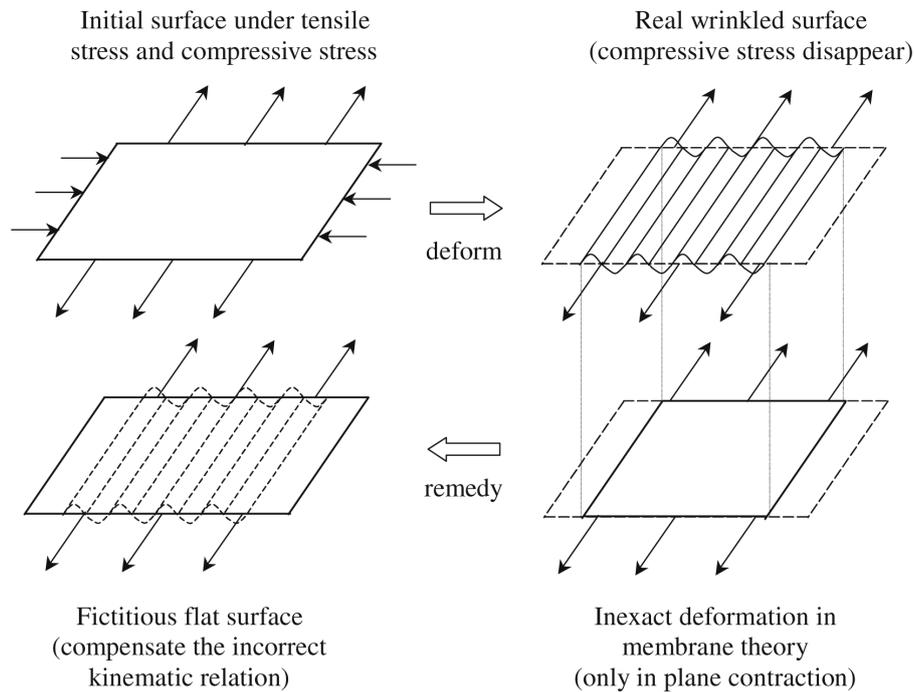


Figure 2-1: Membrane wrinkling illustration [19]

Shell elements

Shell elements are more general than plate or membrane elements since they are subjected to out-of-plane bending moments as well as in-plane shear deformations. Whereas plates only model out-of-plane deformation and membranes only model in-plane deformation. Since the derivation of shell models is very complex, here only the basic continuum mechanics concepts behind general elements as well as the main challenges occurring in the finite element implementation are touched upon.

Mathematical formulations Early approaches to shell modelling simply superimposed the stiffness matrices of plate and membrane elements, however at least initially these models did not perform well [23].

Later on, different mathematical shell models were derived partly based on kinematical considerations from plate theory. Namely the first basic assumption is plane stress in the tangent plane of the shell midsurface. This implies that particles located on a line normal to the mid-surface in the undeformed state will remain connected in this formation once the shell deforms. The second basic assumption is that the stress normal to the shell mid-surface is zero. [24]

The Naghdi shell formulation (1963) [25] is based on these assumptions, this type of shell model is also referred to as Reissner-Mindlin based because the assumptions correspond with their plate theory [26].

Alternatively, Koiter (1965) [27] derived a shell element formulation that uses the additional assumption that the material line orthogonal to the shell mid-surface remains orthogonal

throughout the deformation process. This type of shell is also referred to as Kirchhoff-Love type due to the analogy with the corresponding plate theory [26].

Lastly, an additional simplification that can be made to both of the above mentioned mathematical models, namely that the shell is shallow. This implies that the shell surface is assumed to have a constant curvature radius, this holds true if the dimensions of the shell are small compared to the curvature radius of the mid-surface [28].

Requirements for discretization Every arbitrary structural finite element formulation should fulfil the basic requirements of consistency with the underlying mathematical formulation as well as fulfil the ellipticity condition. The first condition ensures that in the limit of infinitely small elements the solution of the discretised system converges to the solution of the continuous mathematical formulation of the problem. The second condition ensures that no spurious zero energy modes are present, this means that rigid body movements do not influence the element formulation. [16]

Additionally, for a general shell element, it would also be desirable that it performs well for bending as well as for shear dominated problems. Special care needs to be taken such that shear locking can be avoided in bending dominated test cases. Shear locking refers to an underprediction of displacement and stresses for beams and shells with a small thickness. This is a consequence of a weakness in the finite element formulation where the shear deformation does not tend to zero as the element thickness tends to zero and thus the shear strain energy is overpredicted. The shear overprediction results in a locking element where the stiffness is overpredicted [23]. Bathe and Chapelle (1998) [26] state that shear locking occurs independent of the order of the degree of the interpolation functions and always occurs if pure displacement based finite element formulations are used. Although it is less pronounced if higher order elements are used [16]. Additionally, for curved shell elements as for membrane elements also membrane locking can occur if no special care is taken.

To remedy the occurrence of shear and membrane locking, different solution strategies have been proposed. Mixed or hybrid interpolation methods [29] [30], as well as introduction of drilling degrees of freedom [31] or averaging over the element dimensions [31] have shown to be successful.

Lastly, Lee and Bathe (2004) [32] also state that for triangular shell elements it is important that the element formulation be isotropic. Namely, the element stiffness matrices should be independent of the node numbering orientation.

Beam elements

Similar to shell elements the same kinematic assumptions are made with reference to the deformation of the elements along the normal to the midline of the beam [16]. If the elements on the line perpendicular to the undeformed midline remain normal to the deformed midline throughout the deformation process, Bernoulli beam theory applies. Otherwise, if the elements maintain their relative positioning along a straight line throughout the deformation process, but violate the midline normal condition due to the inclusion of shearing effects, then Timoshenko beam theory applies.

The requirements and issues with the discretisation of beam elements are similar to shell elements. Thus shear locking can occur for pure displacement based elements. Moreover, for curved elements also membrane locking can occur. In fact, shear and membrane locking are related if a curved element description is used [18]. Lastly, the same techniques as for shell elements are used to alleviate shear and membrane locking.

Pressurized beams The modelling of pressurised beams in a finite element analysis can either be done explicitly through membrane elements or implicitly through beams. For the explicit method, the tube is discretised using shell or membrane elements, and a pressure field is described as a boundary condition on the inside of the tube. For the implicit method, nonlinear beam models are used with appropriate material properties.

Veldman et al. (2004) [33] as well as Breukels (2010) [12] showed through experimental analysis that the bending behaviour of pressurised beams can be decomposed in two regimes. First, for small deflections, the material behaviour remains linear, and the tube surface remains taut. Once a certain threshold is bypassed, the material behaviour becomes nonlinear, and large-scale wrinkling effects occur. Veldman et al. define a critical collapse moment dependent on the internal pressure, size and material properties of the material that shows satisfactory agreement with their experimental results. Lastly, they also suggest based on an analytical analysis that for pressures below $25kPa$ shell elements are better suited than membrane elements to model the beam deflection behaviour.

Malm et al. (2009) [34] compared experimental and numerical results from explicit and implicit pressurised beam modelling. Their results indicate that explicit modelling is capable of capturing the nonlinear effects in the wrinkled state and that implicit modelling compares favourably with measurements as well as the unwrinkled state of the inflated beam.

Two further beam modelling approaches can be found in Apedo et al. (2010) [35] and Le Van and Wilgosz (2007) [36], they might prove to be good references for the determination of the material properties.

Lastly, Barsotti and Ligarò (2014) [21] developed a wrinkling model for the determination of the equilibrium deformation of inflated membrane elements through modification of the constitutive law. While this was only a static analysis, an elaborate four-stage loading process had to be employed to make the solution procedure stable.

2-2 Fluid model

This subsection is dedicated to the Computational Fluid Mechanics modelling approach in general. A more detailed description of the methods applied in this project will be given in chapter 4. Again as for the structural model, the first subsection 2-2-1 describes the governing equations and the second one 2-2-2 the discretisation approach. The last subsection 2-2-3 lists different turbulence models including their limitations for this particular project.

2-2-1 Governing equations

The governing equations for fluid dynamics are the continuity equation, alongside with the Navier-Stokes equation for momentum conservation and the energy conservation equation.

Assuming that the kite flies at maximally roughly $40m/s$ during typical cross-wind flight maneuvers with a wind speed of about $20m/s$ [37], then the local flow speed over the profile is still expected to lie in the incompressible range with a Mach number of close to $Ma \approx 0.15$. However, if in the future the kite will fly at higher altitudes and with increased wind and flight speeds compressibility effects might become relevant.

Nevertheless, the assumption of incompressibility is valid within the scope of this project and thus the incompressible Navier-Stokes equations can be used to model the fluid in combination with the simplified continuity equation. The energy conservation equation for incompressible flows is decoupled from the momentum equation and thus typically not relevant anymore unless heat transfer phenomena are considered as well. The governing system of equations to be solved for the fluid domain to be solved in an Eulerian framework is shown in equation 2-11.

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (2-11a)$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = - \frac{1}{\rho_f} \frac{\partial p}{\partial x_i} + \nu \cdot \frac{\partial^2 u_i}{\partial x_j \partial x_j} + f_i \quad (2-11b)$$

In case, the governing equations are solved on a domain with moving boundaries, the equations need to be rewritten in another framework. For Fluid-Structure Interaction problems, the equations are usually rewritten in an Arbitrary Lagrangian-Eulerian representation.

2-2-2 Discretisation

The most commonly used discretisation techniques for differential equations are Finite Difference, Finite Volume and Finite Element methods. The methodology behind the methods will not be elaborated here, since they are fairly standard and can be found in any textbooks on CFD.

However, the key advantages of the methods will be briefly pointed out to motivate the choice of the fluid solver for this particular project. The main general literature reference is [38].

First, Finite Difference methods are conceptually very simple, and extension to higher order schemes is quite straightforward given that structured grids are used. However, the use of structured grids significantly limits the applicability of the method for complex geometry. Nevertheless, through the introduction of IB methods more recently, the application to flows with complex geometry has been made possible. Lastly, since the differential form of the governing equations is discretised, conservation of the transported properties is not given unless special care is taken.

Second, the Finite Volume method is currently the most used discretisation technique in industrial applications of CFD due to the ease of implementation and the conservative formulation of the technique. However, as compared to Finite Different and Finite Element methods, the extension to higher order methods is more complicated, especially on unstructured grids. Because multiple approximations have to be made in the discretisation process.

Third, the Finite Element method is standard for structural mechanics applications and has more recently also gained more popularity for fluid dynamics applications. One distinct

advantage is the relatively simple implementation of higher order schemes either locally or globally, especially for Discontinuous Galerkin methods. Moreover, the discretisation is adept at handling unstructured meshes and due to this automatic mesh refinement is quite accessible as compared to Finite Volume and Finite Difference methods.

2-2-3 Turbulence modeling

While the incompressible Navier-Stokes equation provides a full, unapproximated description of the physics behind fluid dynamics, the computational effort required to solve them is significant. In fact, even with the computational resources available today solving Direct Numerical Simulation (DNS) problems is mostly limited to academic benchmark cases, because of the large ranges of scales present in high Reynolds number flows. Thus the application of turbulence models is needed to reduce the complexity of the problem.

Two main approaches to turbulence modelling can be distinguished, namely RANS (Reynolds Averaged Navier Stokes) and LES (Large Eddy Simulation). A short outline of the two different methods will be given in the following with the intent of outlining the limitations that are to be expected for the sake of computational effort.

RANS

A statistical approach to turbulence modelling is taken in the derivation of the Reynolds Averaged Navier Stokes equations, namely the mass and momentum conservation are averaged. Thus only the mean value is of interest. Ensemble averaging is used meaning that the solution of the RANS equations at a specific location and instant in time is the average between different imaginary identical experiments for the specified location and time instant. The governing equations including the definition of the averaging procedure are written out below

$$u_i(\mathbf{x}, t) = \bar{u}_i(\mathbf{x}) + u'_i(\mathbf{x}, t) \text{ where } \overline{\bar{u}_i} = \bar{u}_i, \overline{u'_i} = 0 \quad (2-12)$$

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \quad (2-13)$$

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho_f} \frac{\partial \bar{p}_i}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} + \frac{\partial \overline{u'_i u'_j}}{\partial x_i} + f_i \quad (2-14)$$

The last term on the right-hand side of the averaged momentum equation 2-17 is referred to as the Reynolds stress tensor. It arises from the non-linearity of the convection term and thus needs to be modelled through additional closure models.

The most common approach to the modelling of the Reynolds stress is the Boussinesq hypothesis in equation 2-15, that relates the shear in the mean flow to the averaged velocity fluctuation pairs through introduction of a turbulent eddy viscosity

$$-\overline{u'_i u'_j} + \frac{2}{3}k\delta_{ij} = \nu_T \left(\frac{\partial \bar{u}_i}{\partial \bar{x}_j} + \frac{\partial \bar{u}_j}{\partial \bar{x}_i} \right) = 2\nu_T \bar{S}_{ij} \quad (2-15)$$

where $k = \frac{1}{2}\overline{u'_i u'_i}$ refers to the turbulent kinetic energy. This term is added to ensure that the trace of the Reynolds stress tensor is equal to the turbulent kinetic energy.

This type of RANS turbulence models is also referred to as Linear Eddy Viscosity Models (LEVM). There are multiple implementations of LEV models available since they are most frequently used in industrial applications. According to their degree of complexity, either none or multiple partial differential equations are solved. The solutions from these transport equations can then be used to calculate the eddy viscosity.

For example, there is the one-equation Spalart-Allmaras model which solves a transport equation for an effective eddy viscosity that is then related to the prediction of the turbulent viscosity through an algebraic expression. Other commonly used two-equation models are the $k - \epsilon$ and the $k - \omega$ models that solve two model transport equations for the turbulent kinetic energy k and the rate of dissipation thereof ϵ or the specific dissipation $\omega = \frac{\epsilon}{k}$, respectively. Finally, through analysis of the units, the eddy viscosity can be calculated from the solutions of the transport equations.

However, comparison with DNS and measurement data have shown that linear eddy viscosity models have two distinct fundamental flaws, which prohibit them from performing well for specific flow situations. Namely, LEV models are in general unable to accurately predict wall normal Reynolds stresses, because the strain tensor relation can not account for the redistribution of production between neighbouring Reynolds stresses for curved surfaces. Additionally, for the same reason, the Boussinesq hypothesis also fails to predict the anisotropy of the Reynolds tensor for the flow across a flat plate. The second weakness is that the time history of the flow is not accounted for if changes in the mean strain rate occur, namely they are immediately transmitted to the eddy viscosity, which is the opposite of what happens in reality. [39]

To improve on this, also non-linear eddy viscosity models have been derived. This is done by extending the Boussinesq approximation to a polynomial of the mean stress and vorticity tensor [39]. However, this modelling approach still does not accurately model the time history of the Reynolds stresses.

The most complete RANS turbulence models solve six transport equations, so one for each component of the symmetric Reynolds stress tensor. They are referred to as Reynolds Stress Tensor (RST) models. The six transport equations can be derived directly from the momentum equation; however new unknown higher order terms appear that need to be modelled. In many situations, those models are superior to simpler linear eddy viscosity models, because they can capture the anisotropy of the Reynolds stress tensor as well as the time history thereof better [40]. Nevertheless, this comes with a significant increase in computation time, and the equations tend to be numerically stiff [40].

LES

Contrary to the RANS approach, for LES the governing system of equations is not averaged, but filtered to only resolve scales above a certain cutoff wave length. However, it should

be noted that most LES calculations do not directly filter the equations, but let the mesh implicitly filter the solution. The governing equations are shown below

$$\frac{\partial \tilde{u}_i}{\partial x_i} = 0 \quad (2-16)$$

$$\frac{\partial \tilde{u}_i}{\partial t} + \tilde{u}_j \frac{\partial \tilde{u}_i}{\partial x_j} = -\frac{1}{\rho_f} \frac{\partial \tilde{p}_i}{\partial x_i} + \nu \frac{\partial^2 \tilde{u}_i}{\partial x_j \partial x_j} - \frac{\partial}{\partial x_i} (\widetilde{u_i u_j} - \tilde{u}_i \tilde{u}_j) + \tilde{f}_i \quad (2-17)$$

where similar to the Reynolds stress tensor the non-linearity of the convection term gives rise to the residual stress tensor $T_{ij} = \widetilde{u_i u_j} - \tilde{u}_i \tilde{u}_j$. The term represents the effect of the unresolved on the resolved scales and thus a closure model - referred to as a Subgrid-scale model (SGS) - is needed to be able to solve for the resolved scales.

The most simple SGS model is the Smagorinsky model, which is based on the Boussinesq hypothesis and thus suffers from similar shortcomings as in the RANS formulation. Thus more elaborate SGS models have been developed, for an overview of the different main ideas in modelling see [41], [42].

When compared to RANS, LES offers some clear benefits in the prediction of anisotropic, unsteady turbulent flows and correct representation of large vortices as well as turbulent mixing [43]. However, this comes with a significant increase in computational requirements.

Especially in wall-bounded flows, LES requires very refined grids with refinements in three directions to be able to accurately capture the relevant scales. RANS on the other hand only needs refinement in wall normal direction [44]. To help reduce the computational requirements for LES in wall-bounded flows, hybrid approaches that use URANS for the modelling of the attached boundary layers and LES for the rest of the flow have been developed. They are referred to as Detached Eddy Simulations (DES) [45]. However, this initial approach suffered from grid-induced separation in the interface region between the two models [46]. This issue has been solved by the introduction of Delayed DES (DDES) [47]. Nevertheless, DES still remains an active area of research.

Alternatively to hybrid approaches, also wall-stress models that use analytical wall functions in the attached region have been developed as well [48].

2-3 Immersed Boundary Treatment

Conventional fluid mechanics techniques employ body-fitted meshes. In the context of Fluid-Structure Interaction problems with moving interface boundaries, the conventional approach referred to as Arbitrary Lagrangian-Eulerian (ALE) is to deform the fluid and solid mesh such that it follows the outline of the body. Such that the boundary of the fluid mesh always coincides with the boundary of the physical fluid domain and the same for the structure.

Contrary to conventional fluid-solid boundary condition methods, Immersed Boundary (IB) methods have been developed that allow the computation on non body-fitted meshes. While a multitude of implementations of IB methods exists, the main characteristics remain the same.

The solid mesh is not modified, whereas the fluid mesh is extended to cover the entire domain including the subdomain covered by the fluid. The reciprocal relationship between the fluid and the solid is either modelled through fictitious body forces or through the direct imposition of boundary conditions on the nodes close to the interface.

Mittal and Iaccarino (2005) [49], as well as Sotiropoulos and Yang (2014) [50] published review papers on the topic of IB methods in general and more specifically for FSI applications, respectively. A summary of their findings will be presented in the following. Subsequently, the interface boundary conditions as relevant for this thesis will be outlined in more detail.

Mittal and Iaccarino, in particular, distinguish between continuous and discrete forcing methods, in reference to whether the interface forcing terms were introduced before or after discretisation of the NS equations. Moreover, Sotiropoulos and Yang make an additional distinction between schemes that enforce a sharp or diffuse solid-fluid interface.

Classical IB method This method was pioneered by Peskin in 1972 [51] and was the first step towards IB modelling of fluid-solid interfaces. Counterintuitive to the usual approach in FSI modelling, the solid boundaries follow the velocity of the surrounding fluid, and in turn, the structure imposes deformation forces on the fluid. Fitting constitutive laws are used for calculation of the interface forces depending on the deformation of the solid. Since this implies a continuous forcing approach where the forces are calculated at the exact solid surface nodes, force distribution functions have to be employed to spread the force to the surrounding fluid nodes. In the original formulation of the method smoothed Delta functions are used for this purpose. The same approach is taken for the interpolation of the fluid velocity to the surface of the structure.

This method is well suited to FSI problems with elastic bodies - the original application involved the flow patterns around heart valves - where simple constitutive laws for the deformation forces can be derived. However, for stiffer or rigid bodies this approach is not well suited due to the necessity of using small time steps for numerically stable integration. Moreover, due to the use of force distribution functions the interface is also not sharp.

Direct forcing IB method In contrast to the classical IB method for direct forcing methods, the forcing term is not derived from a constitutive law but explicitly calculated from a desired interface fluid velocity \vec{u}_f . The desired velocity is derived from the structural velocity \vec{u}_s at the interface. This is more clearly illustrated in equations 2-18 when an implicit Euler formulation is used for the time stepping.

$$\frac{\vec{u}_f^{n+1} - \vec{u}_f^n}{\Delta t} = RHS + \vec{f} \rightarrow \vec{f} = \frac{\vec{u}_s^{n+1} - \vec{u}_f^n}{\Delta t} - RHS \quad (2-18)$$

Two approaches exist for the calculation of the forcing term. First, for the explicit forcing method, the forces are explicitly calculated at the solid boundary location with the extrapolated version of the right-hand side. Second, for the implicit forcing approach, a linear system of equations has to be solved for the determination of the forcing terms. Here, a coupling is established between the force and the velocity interpolation, which results in a better agreement of the structural and the fluid velocity [50].

As compared to the classical IB method the direct forcing approach is more stable, because the IB approach does not introduce any additional stiffness into the system. However, the solid interface is still diffused due to the use of forcing distribution functions.

Penalization method The forcing term is made proportional to the velocity difference at the interface, namely see equation 2-19 for the full expression. $\chi_S \in [0, 1]$ is a mask function that varies depending on the distance from the interface, since this is a smooth function the interface is again smeared. λ is called the penalization parameter, similar to the classical IB method if chosen too large the system of equations can become stiff. [50]

$$f(\vec{x}) = \lambda \chi_S (\vec{u}_s(x) - \vec{u}_f(x)) \quad (2-19)$$

Cut-cell methods Since previous methods neither satisfy the underlying conservation laws nor provide a sharp representation of the interface, cut-cell methods were developed for Finite Volume solvers. As the name suggests, cells that are split by the solid-fluid interface are reshaped to follow the solid boundary more accurately. For the cut cell interfaces, a reconstruction of the flux from the surrounding faces has to be carried out. [49]

However, once moving boundary problems are investigated, non-physical force oscillations are observed with this method. The phenomenon originates from when solid or fluid cells switch to fluid or solid cells during one time step, a spatial or temporal discontinuity takes place, respectively. Thus, special care has to be taken to mend this. Moreover, the extension of the method to 3D is also not straightforward due to the complicated geometrical nature of the problem. [50]

Hybrid Cartesian-Immersed Boundary methods Similar to cut-cell methods, the interface is sharply enforced through the imposition of boundary conditions on nodes of the fluid mesh near the immersed boundary. Depending on the method either ghost cells within the solid or background nodes within the fluid with neighbouring nodes in the solids are used for the imposition of boundary conditions. Similar to direct forcing methods, the solid velocity is known at the interface, and the goal is to approximately impose this velocity on the fluid at the interface. An interpolation scheme is then used to determine the values that need to be imposed on the ghost or background nodes to obtain the desired fluid velocity at the interface. [50]

If background nodes are used in combination with a fractional step approach, special care needs to be taken to ensure mass conservation. Since the boundary conditions are usually imposed on the intermediate velocity field. [52]

In comparison with the cut-cell method, here the Cartesian grid can be maintained, and no computationally expensive grid modifications need to be performed. However, for the same reasons as before non-physical force oscillations can occur for moving boundaries [50]. Depending on the order of the interpolation scheme, the order of the base discretisation of the solver can be maintained [53].

Inclusion of wall functions For fluids meshes that are not body conforming, it is more difficult to obtain a clustering of the cells in the boundary layer region, especially if Cartesian meshes are used. Thus if the boundary layer is to be fully resolved on the mesh, either local mesh refinement for Cartesian meshes [54] or mesh adaptivity for unstructured meshes [55] are valid options. However, combined IB and wall function implementations can be found in literature as well. Kalitzin and Iaccarino (2002) [56] implemented a direct forcing approach with an analytically derived wall function in a Finite Volume RANS solver. Kalitzin et al. (2005) [57] further improved on this by deriving a novel wall-function formulation approach, that is consistent with the used turbulence model. They obtain a turbulence model specific universal velocity profile from the numerical solution of a zero-adverse pressure gradient flow on a flat plate. Subsequently, they use a look up a table for the skin friction velocity and the turbulence quantities when wall functions are used. This makes the approach consistent with the turbulence model, that is used for the outer flow; thus the results are grid independent as compared to analytical wall functions. More recently, Bernardini et al. (2016) [46] used the same wall function approach also in combination with a direct forcing approach on a DDES finite difference solver.

2-4 Coupling

This section will discuss feasible methods to couple the fluid and the structure solver together to form a full FSI problem. To clarify, the coupling approach is mostly independent of whether an ALE or an IB approach is used for the fluid model. Moreover, the time discretisation scheme for the structure and the fluid do not necessarily have to be the same. Although certain combinations of the time integration schemes between fluid and structure model may increase or decrease the discretisation error or influence the stability of the method.

The first subsection 2-4-1 will setup basic conventions. The second section 2-4-2 will give an overview of different suitable coupling schemes. Subsequently, the third section 2-4-3 revolves around the added mass effect for highly flexible wings.

2-4-1 General classification

For this project a partitioned solver setup will be used, thus contrary to a monolithic solver setup two different solvers will be used for the structure and the fluid. This allows for better modularity of the solvers, and thus they can be ideally suited to either a particular solid or fluid situation. For this problem, in particular, this is important, because while both solvers use a finite element discretisation, the structure mesh is much coarser than the fluid mesh. Moreover, in the structural solver, the canopy will be modelled as a 2D shell element in 3D space, whereas the fluid solver in 3D space can not resolve a zero thickness element. As a consequence in an intermediate step, the structure needs to be artificially extruded. This would not be possible as easily if a monolithic solver were to be used.

However, using a partitioned setup necessitates that the two solvers are properly coupled. FSI problems can be classified into two main categories based on the underlying physics of the problem. Namely, loosely coupled problems where only a single execution of the fluid and the structure solver are necessary per time step and strongly coupled problems where sub-iterations have to be employed for stability and accuracy reasons. One example of a strongly

coupled case is highly flexible wings where small deviations in the aerodynamic forces can lead to large structural deformations. Of course, the problem of this project falls into this category, and thus most likely a loosely coupled technique will not be sufficient in this case.

Depending on the employed immersed boundary method, different boundary conditions are imposed from the structure on the fluid and vice versa. However, for IB methods with prescribed velocities at the borders, a Dirichlet-Neumann decomposition is used the same as for ALE formulations. Namely, the structure imposes certain domain boundaries and velocities on the fluid and the fluid, in turn, imposes forces on the structure at the interface. Such that the fluid and structure solver can be represented as operators \mathcal{F} and \mathcal{S} , respectively. The operators act on the interface displacements d and the interface pressure forces p .

$$d = \mathcal{S}(p) \quad (2-20a)$$

$$p = \mathcal{F}(d) \quad (2-20b)$$

2-4-2 Different coupling schemes

In the following, the ideas behind some commonly used serial coupling techniques are outlined, if they show potential for application to the problem in this project. Namely, the coupling techniques should only require access to the output of the two solvers at the interface, such that the solvers can be treated as a black box. Apart from this, of course, the coupling should be robust and require as few sub-iterations as possible.

Gauss-Seidel scheme

The most simple serial partitioning scheme, if necessary sub-iterations of the scheme can be performed. Serial coupling refers to the sequential advancement of the two models that are to be coupled to each other.

$$d^{k+1} = \mathcal{S}(p^k) \quad (2-21a)$$

$$p^{k+1} = \mathcal{F}(d^{k+1}) \quad (2-21b)$$

Gauss-Seidel with dynamic relaxation

Because for strongly coupled problems the evaluation of the fluid solver, in particular, is very expensive, classical fixed point iteration schemes are used for the Gauss-Seidel scheme to accelerate the convergence of the solution.

Namely, the problem can be rewritten as a fixed point iteration accordingly

$$\tilde{d} = \mathcal{S} \circ \mathcal{F}(d) \quad (2-22a)$$

$$\hat{\mathcal{F}}(d) = \mathcal{S} \circ \mathcal{F}(d) - d = \tilde{d} - d \quad (2-22b)$$

$$d^{k+1} = d^k + \omega \cdot (\mathcal{S} \circ \mathcal{F}(d^k) - d^k) = d^k + \omega \cdot \hat{\mathcal{F}}(d^k) \quad (2-22c)$$

From this either Aitken's underrelaxation method or the Steepest Descent (also known as Conjugate Gradient) method can be used. Both methods are implemented on an incompressible fluid and highly flexible structure FSI test case by Kuttler and Wall (2008) [58]. Comparison between these two methods and Newton-based schemes was made. The Aitken under-relaxation scheme was clearly more efficient than the Steepest Descent one. It also compared well with the Newton-based methods for this specific test case in terms of computational effort. However, this does not necessarily hold true in general because it required more iterations per time step but had lower computational requirements in total for one time step than the Newton-based methods. Because these calculations were done on a small test case, the higher iteration count did not result in a significant increase in computational time, but for more realistic problems Newton-based method can become more efficient.

Newton iterations

Newton methods are gradient based and derived from a simple Taylor expansion and truncation of second and higher order terms. The scheme can be summarised to

$$\hat{\mathcal{F}}(\mathbf{d} + \delta\mathbf{d}) = \hat{\mathcal{F}}(\mathbf{d}) + \mathcal{J}_{\hat{\mathcal{F}}} \delta\mathbf{d} + O(|\delta\mathbf{d}|^2) \quad (2-23a)$$

$$\hat{\mathcal{F}}(\mathbf{d}^{k+1}) \stackrel{!}{=} 0 \rightarrow \mathcal{J}_{\hat{\mathcal{F}}} \delta\mathbf{d} = -\hat{\mathcal{F}}(\mathbf{d}) \quad (2-23b)$$

$$\mathbf{d}^{k+1} = \mathbf{d}^k + \delta\mathbf{d} \quad (2-23c)$$

with the Jacobian $\mathcal{J}_{\hat{\mathcal{F}}} = \frac{\partial \hat{\mathcal{F}}(\mathbf{d})}{\partial \mathbf{d}}$ of the combined application of the structure and the fluid operator.

Of course, the Jacobian is not exactly known unless a symbolic expression thereof has been derived. Thus an approximation of it needs to be obtained; the method is then referred to as a quasi-Newton algorithm. In the following, one example from the relevant literature will be outlined.

Linear combination of previous sub-iteration results Degroote et al. (2006,2009) [59, 60] developed a reduced order model of the Jacobian or its inverse from linear combinations of previous sub-iteration results. They named their technique interface quasi-Newton with inverse Jacobian from a least-square model (IQN-ILS) or (IBQN-LS), respectively. In a subsequent publication, Degroote and Vierendeels (2012) [61] extend the two algorithms through use of multiple grids (or multiple levels). They refer to it as either ML-IQN-ILS or ML-IBQN-LS. For both methods, the principles remain the same, but the convergence of the method is accelerated through first converging the solution on a coarser grid for both the structure and the fluid. The idea behind the technique is that coupling instabilities in Gauss-Seidel iterations usually occur only in the low wave number range, hence the bulk of the coupling iterations can be done on the coarser grid [61], [62], [63]. The two solvers are still treated as a black box, thus the communication between grid levels and solvers is done via an interface grid that corresponds to the target finest grid level, but is also used for the coupling of the coarser meshes.

2-4-3 Added mass effect

For very small time steps the dynamic equations of the structure can be rewritten such that the effect of the pressure fluctuations in the fluid are contained in an additional mass term (Causin et al., 2005 [64]). Hence for structures with low inertia and large fluid forces, the added mass effects are strong, because the inertia effect of the fluid on the structure can be larger than the inertia effect of the structure itself. Causin et al. also showed that loosely coupled schemes and strongly coupled schemes without relaxation become unstable if the added mass of the fluid is larger than the structural mass for a 2D incompressible test case on a potential fluid and a linear elastic thin tube.

Brummelen (2009) [62] further extended on their results on a 2D model problem concerning the fluid flow on a semi-infinite domain over a flexible panel. He applied the same test case to both a compressible and an incompressible flow with a staggered coupling scheme. He showed that for compressible flows the stability of the coupling process is independent of the mass ratio given that the time step is sufficiently small and that with sub-iterations convergence to the monolithic solution is obtained. However, for incompressible flows the opposite holds true. Namely, independent of the time step sub-iterations can diverge for the staggered scheme and the order of accuracy is lower than for a monolithic solver. Further for incompressible flows he also derived that for very small time steps the viscous and convective terms are less critical for the convergence than the added mass effect. Through a frequency analysis he showed that for the limit of an infinitely small time step, the added mass term is proportional to the time step for compressible flows and constant for incompressible flows. Thus, this explains why even for very small time steps for incompressible flows explicit sub-iterations may not be stable, because the time step does not influence the added mass effect.

Kuttler and Wall (2008) [58] implemented a Gauss-Seidel scheme with Aitken under-relaxation and tested it on several incompressible test cases with highly flexible structures. They showed that a decrease in time step does not necessarily also result in a decrease in the number of necessary sub-iterations for convergence within a time step. This is in good agreement with Brummelen.

Similar to Brummelen [62], Degroote et al. (2008) [63] also performed a Fourier error analysis on a 1D-linearized test case with incompressible flow and a varying stiff structure treating both solvers as a black box. They showed that the low-frequency content of the solution can make the explicit coupling with Gauss-Seidel sub-iterations unstable. Similar to Kuttler and Wall, they also showed that a decrease in time step or stiffness of the structure can result in higher instability of the solution. Then in a second step, they tested both explicit coupling with Aitken under-relaxation and the IBQN-LS technique [59]. The results indicate that IBQN-LS clearly outperforms the Aitken under-relaxation fixed point scheme requiring roughly half the number of sub-iterations for convergence.

While all the previous publications were related to ALE implementations, the physics behind the problem remains the same. Thus it is expected that the same conclusions regarding the coupling algorithm also hold true for an IB implementation. For example, Borazjani et al. (2008) [65] apply a curvilinear immersed boundary method (CURVIB) to a hemodynamics test case with a strong added mass effect. The same conclusions with regards to the stability of explicit and implicit coupling are drawn. Additionally, also Aitken's under-relaxation is applied which can stabilise previously diverging constellations.

2-5 Verification

Since the main goal of this project is the development of a methodology that can in future projects be applied to the modelling of a Leading Edge Inflated kite in flight, the solver needs to be appropriately verified. This section outlines the findings of the literature review with regards to validation data.

The first subsection 2-5-1 lists the classical FSI benchmark cases whereas the second subsection 2-5-2 lists references specifically tied to membrane wings. Finally, the last subsection 2-5-3 concludes which test cases are most suitable as well feasible within the context of this project.

2-5-1 FSI benchmark cases

In literature, the two most commonly used benchmark cases for the implementation of a coupled CFD-CSM solver are the Turek & Hron channel flow [66, 1] and the AGARD 445.5 wing [67]. Since the second case is set in flow regimes where compressibility effects of the fluid can not be neglected, it is not relevant to this project. However, the Turek & Hron channel flow benchmark case is well suited for the validation of methodology to be developed within this report, because it models a two-dimensional strongly coupled problem.

(L, H)	$(2.5m, 0.41m)$
(C_x, C_y)	$(0.2m, 0.2m)$
r	$0.05m$
(l, h)	$(0.35m, 0.02m)$

Table 2-1: FSI benchmark parameters

The geometry of the test setup is shown in figure 2-2 and further clarified in table 2-1 modelling the flow around a rigid cylinder with an elastic flag attached to it. The height and H and length L of the channel are set. The cylinder with radius r and centre location (C_x, C_y) as well as the flag length l and thickness h are also described. The centre of the cylinder is not congruent with the centre of the channel such that the onset of flag oscillations are not dependent on numerical precision.

The initial report published by Turek and Hron [66] comprises not only of benchmark data for the coupled simulation, but also of reference results for the separate validation of the structure and fluid model. For this project, it would, of course, be useful to have a rough estimate of the error introduced through the models themselves. The second report published by Turek et al.[1] then lists benchmark results for the coupled simulations from different authors using different methodologies. Three different coupled test cases are documented, where the first and the second case correspond to a steady deformation of the flag and the third case results in an unsteady periodic oscillation of the flag tip deflection and aerodynamic forces.

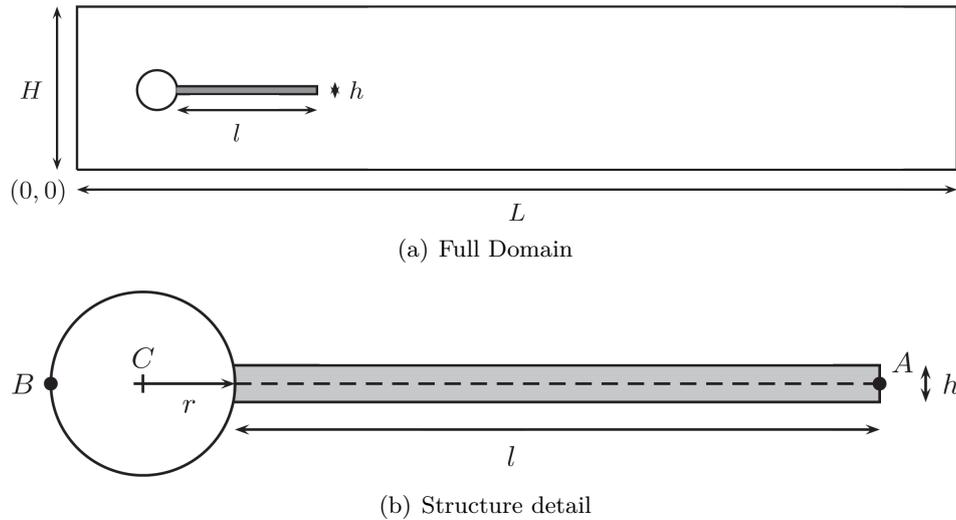


Figure 2-2: FSI benchmark case [66]

2-5-2 Membrane wings in literature

There exist various experimental references for the verification of the FSI solver applied to a clamped cantilever beam as pointed out by Berens [15] that worked on this project previously. For example, two steady cases as published by Levy in 1942 [68, 69] as well as multiple references from Dowell et al. [70, 71, 72, 73] for a cantilever beam undergoing flutter.

However, the canopy of the kite has very little bending stiffness and thus while it can be represented with shell elements, verifying the solver on test cases that are bending dominated will only give limited insight into how well the structural model performs for shear dominated applications and how efficient the coupling mechanism is in this case. Moreover, in case only membrane elements will be available in the structural solver framework, the basic flat plate benchmark case will have to deviate from a cantilever beam setup. Thus in the following, a short literature review of experimental and numerical investigations of membrane wings at different Reynolds numbers is presented with the aim of finding a more realistic benchmark case beyond the Turek-Hron case.

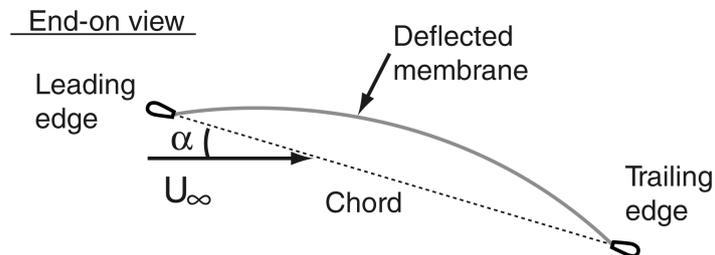


Figure 2-3: Test case clamped membrane [74]

Moderate Reynolds number The setup as shown in figure 2-3 with a fixed leading and trailing edge membrane wing is well documented in literature, however with the application

to micro air vehicles or modeling of bird wings in mind, whose Reynolds numbers are of the order of $Re \approx 10^4 - 10^5$. Given the fluid solver methodology available for this project, such test cases are out of reach, because a transition model is not available and the Reynolds number is too high for a Direct Numerical Simulation (DNS). Nevertheless, a cautious qualitative comparison might still prove useful, hence some outstanding publications are listed in the following. Song et al. (2008) [74] investigated the setup shown in figure 2-3 with a low aspect ratio of order of $AR \approx 1$ for a Reynolds number between $Re \approx (0.7 - 2.0) \cdot 10^5$ with steady inflow speed and found that the Weber number $We = \frac{\rho U^2 c_{CL}}{Et}$ characterizes the steady as well as the unsteady aero-elastic effects of the wing to some extent. Higher Weber numbers were associated with higher maximum camber of the deflected shape, of course here also the prestrain of the membrane plays an important role. For the static aerodynamic coefficients, the found as compared to a rigid plate, the flow remains attached larger angles of attack and shows a smooth stall onset. They conclude, that this is due to a decambering effect, where the growth of the laminar separation bubble with increasing angle of attack is countered with a decambering of the membrane due to the associated change in surface pressure distribution. Moreover, it should be noted that for higher inflow speeds - within a certain interval where Reynolds effects are not dominant - the decambering effect becomes stronger and thus the flow is expected to be less likely to separate from the leading edge at a higher as compared to a lower inflow speed. Apart from hysteresis effects in the stalled regime of the wing, that pertain to the same principles for rigid as well as flexible wings, the membrane wing also showed hysteresis effects around zero angle of attack. Due to the higher lift coefficients, as compared to the rigid case, the membrane wing also shows higher drag values due to downwash. Despite the steady inflow conditions, it was observed that membrane vibrations were measured for all test conditions possibly excited through vortex shedding at the leading edge. The dominant vibration frequencies were found to be close to the eigenfrequency of the membrane and increase with increasing Weber number.

Rojratsirikul et al. (2009,2011) [75, 76] further investigated the unsteady effects present in a similar setup for Reynolds number in the range $Re \approx (0.5 - 1.0) \cdot 10^5$ with a low aspect ratio wing. They showed that for this Reynolds regime, the mean maximum deflected camber of the membrane is only weakly dependent on the angle of attack, and the changes in the membrane deflection are small compared to the changes in the flow regime. For the unsteady analysis, differences in predominant mode shapes were observed for different angles of attack and inflow speeds, however, for large angles of attack, the second mode shape became dominant independent of the inflow speed which might indicate a coupling with the vortex shedding at the trailing edge. The maximum deflection amplitude increased with increasing Reynolds number, possibly due to the shear layer staying closer to the membrane surface with the same trend in inflow conditions. Comparing the unsteady flow field of the membrane wing at high angles of attack with a rigid airfoil with the mean deflection shape of the flexible membrane wing showed that the flexible membrane promotes vortex roll-up and thus the flow separates at a higher angle of attack.

In another publication, Rojratsirikul et al. (2010) [77] investigated the effect of prestrain and excess length of the membrane. The results indicated that prestrained membrane wings behave similar to rigid airfoils and membrane wings with excess length can maintain attached flow for higher angles of attack due to the increased camber.

Lastly, Arbos et al. (2013) [78, 79] investigated two different Leading and Trailing Edge

supports, namely circular and rectangular ones. They concluded that the choice of supports influences the mean deflected shape of the membrane wing, for example, circular cross-sections result in an S-shaped membrane deformation at low incidence angles. Moreover, it was also shown that especially for the rectangular supports, coupling between the natural frequencies of the membrane, the support structure and unsteady vortex shedding at high angles of attack can take place.

Low Reynolds number DNS simulations were first carried out by Gordnier (2009) [80] through coupling of a two-dimensional high-order Navier-Stokes solver with a one-dimensional nonlinear membrane model at Reynolds numbers in the range of $Re \approx 2500 - 10000$. Later on, Gordnier and Jaworski (2012) [81] extended this setup to include prescribed motions of the trailing and leading edge supports for the same Reynolds number range. Then Gordnier and Attar (2014) [82] extended the setup to be three-dimensional and applied them to the test cases of Rojratsirikul et al. (2011) [76]. In the most recent publication, Gordnier and Jaworski (2015) [83] included the effect of movement in the leading and trailing edge in their three-dimensional simulation setup.

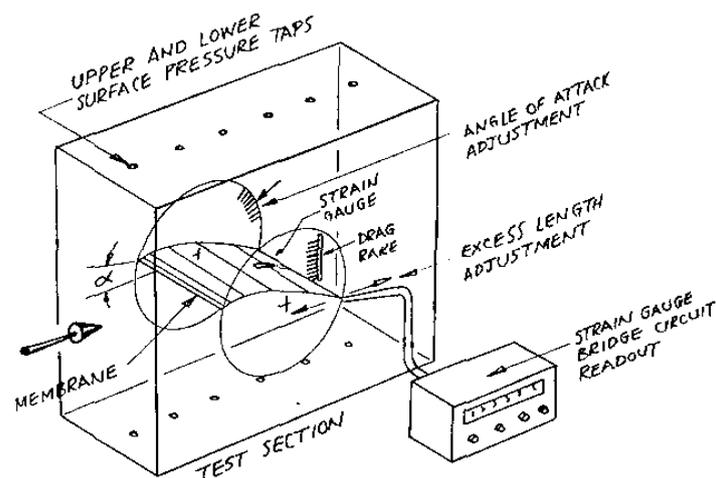


Figure 2-4: Experimental setup clamped membrane [84]

High Reynolds number Greenhalgh et al. (1984) [85, 84, 3] conducted experiments in a two-dimensional wind tunnel on a membrane wing with fixed leading and trailing edge but variable membrane excess length at a Reynolds number of $Re = 1.3 \cdot 10^6$. Greenhalgh et al. state, that the lift was determined from measuring the pressure distribution on the wind tunnel bottom and top wall, the drag was determined from wake momentum deficiency as measured with a wake rake and the membrane tension was measured with strain gauges. The results were similar to the results obtained for lower Reynolds numbers, namely higher maximum lift coefficient and attached flow up to larger angles of attack as compared to the rigid plate, benign stall behaviour, lift hysteresis around zero degrees and visible vibration of the membrane for separated flow. However, the experiment description is somewhat limited, in particular, the material properties of the membrane are not specified beyond a certain brand name. An illustration of the experimental setup is shown in figure 2-4.

Earlier experimental research by Sweeney (1961) [86] at Princeton University as well, involved the investigation of full sailplane model at Reynolds numbers up to $Re \approx 1.5 \cdot 10^6$. This was followed up by other full sailplane model tests by Fink (1967) [87], Fink (1969) [88] and Ormiston (1971) [89] at similar Reynolds numbers, see figure 2-5 for a schematic of one of the experimental setups. Later on, Maughmer (1979) [90] investigated eight different sailplanes with different cross-sections ranging from single to double membrane wings at a Reynolds number of approximately $Re \approx 0.25 \cdot 10^6$. However, it should be noted that due to the high turbulence intensity in the flow, Maughmer considered the effective Reynolds number to be at least twice as high.

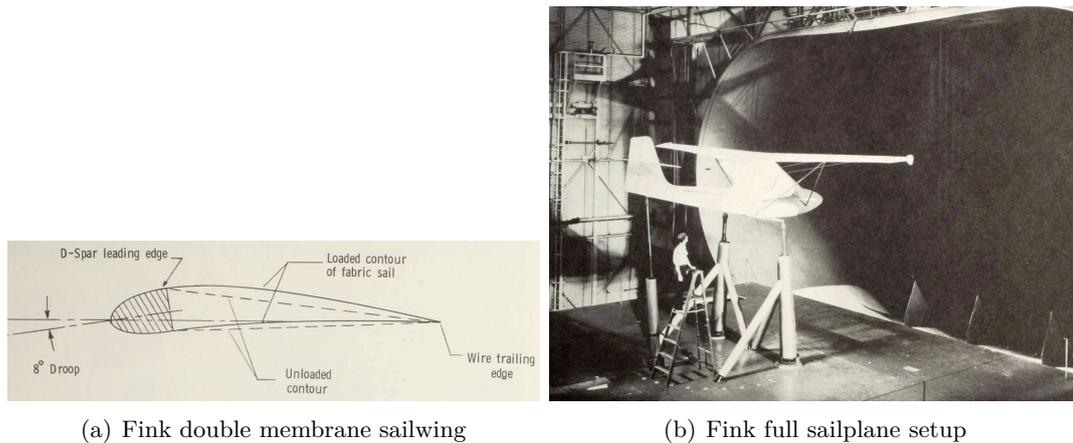


Figure 2-5: Experimental sailplane setup of Fink (1967) [87]

In Delft at the Aerospace faculty, two reports on single and double membrane sail wings with adjustable slack were published by de Boer [2, 91] in 1980 and 1982, respectively. The experiments were carried out in the low-speed low-turbulence wind tunnel, and the two-dimensional setups are shown in figure 2-6. Both wings consist of a steel leading edge, a flat plate at the trailing edge and a dacron sail material for the membrane. Lastly, both setups have a slit between the wind tunnel wall and the end of the wing on both sides, such that air can pass in and out of the midsection.

The first report from 1980 investigated single membrane airfoils with adjustable sail slack in terms of lift and drag behaviour at Reynolds numbers ranging from $Re \approx (0.3 - 0.5) \cdot 10^6$ within a limited angle of attack range as compared to the previous publications of roughly $\alpha \approx 3 - 11^\circ$. However, hysteresis effects were not considered and the sail material properties are not characterised.

The follow-up report from 1982 placed double membrane airfoils in the wind tunnel with a Reynolds number range of $Re \approx (0.5 - 0.8) \cdot 10^6$. The measured properties were lift, drag and deformed membrane shape as a function of the angle of attack α , the rotation angle of the leading edge bar δ and the sail length on suction and pressure side l_u, l_l . However, here again, the range of angles of attack is limited from $\alpha \approx 2.5^\circ - 10^\circ$. The material stiffness is noted this time.

Lastly, there also exists an old Master thesis from de Wachner (2008) [92] of a numerical and experimental investigation of a Ram-Air Wing at a Reynolds number of one million.

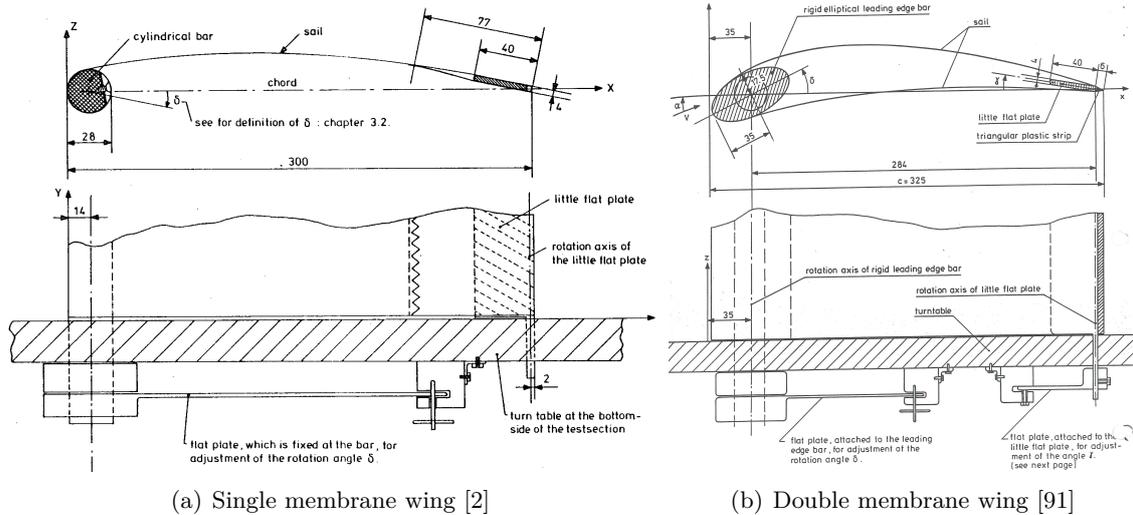


Figure 2-6: Experimental sailing setup as used by den Boer in 1980 and 1982

Other Apart from the literature on sail wings, micro air vehicles and animal wings, there is a multitude of well-documented literature available for ram-air parachutes for military applications. Two examples of full coupled high-fidelity simulations can be found in [93], [94]. While those concepts are not directly related to this project, they show that it is feasible to implement the coupling between a non-linear membrane model and a high-fidelity fluid solver.

More recently, the application of flexible membrane wings with a rigid leading edge, trailing edge and chordwise ribs periodically distribution along the blade span for wind turbines have been reexamined by Piquee and Breitsamter (2015) [95] as well as Saeedi et al. (2016) [96, 97, 98]. Figure 2-7 shows the undeformed and deformed setup of Saeedi et al. [98] for uniform inflow.

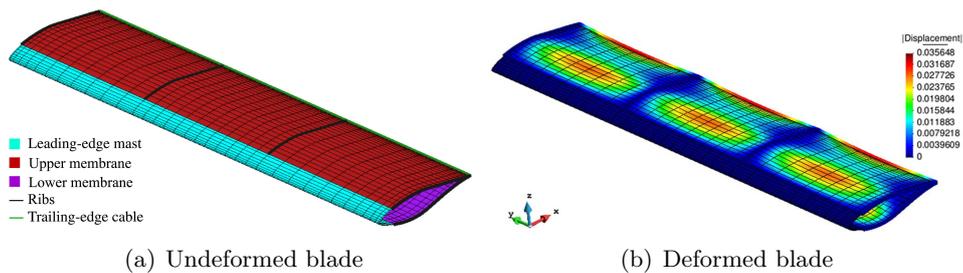


Figure 2-7: Membrane wing turbine blade as modeled by Saeedi et al. [98] for uniform inflow (Reynolds number unknown)

Saeedi et al. [96] coupled a nonlinear Finite Element structural solver with a RANS fluid solver and concluded that for their setup the semi-flexible membrane blade had several advantages over a fully rigid blade in terms of aerodynamic performance. Namely compared to the fully rigid blade the semi-flexible membrane blade was predicted to have a higher lift slope although with a lower lift coefficient at zero angle of attack, a higher maximum lift over drag ratio and as a consequence a higher maximum lift coefficient.

2-5-3 Proposed test cases

The first recommendation is to initially verify the solver on the Turek & Hron FSI benchmark case [1, 66]. It is a complex test case that does not require much computational effort because due to its low Reynolds number setting. However, it should be noted, that the structural solver may not be ideally suited to model the flag part of the setup, because it is much thicker than the canopy of a membrane wing. Moreover, the IB fluid solver may also perform better at a high Reynolds number since this was its development target range. Nevertheless, this will allow at the very least to check whether the proposed solver methodology is working as expected.

The second recommendation is to then validate the solver on a more appropriate test case with a higher Reynolds number and an actual membrane or sail wing. For the current kite design in the AWE group in Delft the Reynolds number ranges from roughly one to eight million depending on the size and flight speed. Thus, simply based on Reynolds number similarity the best setup would be the one from Greenhalgh et al. (1984) [3, 84, 85], because it has the highest Reynolds number out of all the mentioned references. However, there are some inconsistencies present within the experiment description with regards to the membrane material. Namely, no material properties are given even though the strain of the membrane is measured. The one other reference treating a single membrane airfoil at a Reynolds number close to one million is the one from den Boer (1980) [2] though it is also missing a complete sail material characterisation. The same holds for the second publication of den Boer (1982) [91].

Full sail plane or sail wing investigations as done by Fink [87, 88], Ormiston [89] and Maugmer [90] are comparatively big test cases that would most likely conflict with the timeline of this project due to the limited available computational resources. Moreover, the modelling of a double membrane flexible wing section would further complicate the setup even if the porosity of the material is assumed to be zero. Most likely, this would require a 3D simulation setup with slits at the sides of the profile.

The moderate Reynolds number cases outlined in paragraph 2-5-2 are dominated by laminar separation effects and thus are not directly comparable to the high Reynolds number flow around a LEI kite. Moreover, the simulation setup of this project has no transition model and is thus not capable of modelling transitioning flows. However, cautious qualitative comparison especially with reference to unsteady effects should still be possible. Though, it should be noted that for these cases due to the difference in Reynolds number, the stall behavior is expected to be completely different. Moreover, since a URANS turbulence model, that is only capable of reproducing mean flow unsteadiness, will be used, it is not clear how much of the unsteady fluid-structure interactions can actually be captured even for a fully turbulent case. Nevertheless, the setup should be capable of modelling the mean deflected shape of the membrane for a given angle of attack and inflow speed as well as hysteresis effects due to an unsteady inflow. Both of these effects can be used to test the solver efficiency, stability and accuracy.

The publications on low Reynolds number DNS investigations of membrane wings by Gordnier et al. [80, 81, 82, 83] would be possible to reproduce with solver setup of this project, however physically speaking this would be very far removed from the kite application and require an immense computational effort for three-dimensional calculations. Moreover, having a high

Reynolds number is also important with reference to the validation of the Immersed Boundary method, since high Reynolds number flows are more difficult to model for Immersed Boundary methods due to the larger wall normal flow gradients.

Summarizing, given the requirement for a simple and computationally limited test case and high Reynolds number, the publication from Greengalgh et al. [3] is the best validation case for the solver methodology to be implemented within this project for the mean deflection and possibly also flow hysteresis. However, given that the fluid solver is fully turbulent the laminar onset will not be accurately captured, and thus discrepancies between measurement and simulation results are to be expected. Smith and Shyy (1996) [99] numerically investigated this test case with a fully turbulent solver with Menter's SST κ - ω turbulence model as well and results from this project can thus also be compared with their numerical results. In a second step, if time allows this simple membrane setup can be extended to include a rigid leading edge and validated although only for low angles of attack with the experimental results from den Boer [2] again with discrepancies due to the fully turbulent solver. Then in a next step, the setup can be further upgraded to include an inflated leading edge although here only qualitative verification is possible with for example the lift polars obtained from Breukels [12].

2-6 Available research

This section will very briefly touch upon the research done at the AWE section in Delft, with the aim to explain what tools or results are directly available and can be re-purposed for this project.

Basic working principle In terms of aero-elastic modelling, the AWE group in Delft mainly focuses on the development of a methodology that uses a kite connected to a ground-based generator through its tether. The power production cycle or pumping cycle can be split into two phases. In the traction phase, the kite pulls the tether out of the generator, and thus the generator produces energy. In the retraction phase, the kite is reeled back in by the generator that is now used as a motor and thus uses energy. While flight path optimisation of the kite is an ongoing research topic, for this particular setup the reel-out phase usually involves figure eight flight patterns [100]. Thus during a typical power production cycle of the kite, it sees a varied range of flow inflow profiles in terms of angle as well as magnitude. Figure 2-8 illustrates the flight path of the wing.

Components of the kite Because in the following the modelling of the different components will be referred to in detail, figure 2-9(a) shows an illustration thereof. The power lines are attached to the leading edge - wing tip intersection of the kite and allow for manipulation of the angle of attack of the wing. Bridle lines, if present, are also attached at the leading edge of the kite, but they are only used to maintain a desired shape of the leading edge. The steering lines are attached to the leading edge - wing tip intersection of the kite, the turning deformation mode of the kite is shown in figure 2-9(b). Pulling the right steering line results in an increase of the angle of attack at that side and a decrease of the angle of attack on the left wing tip [12]. Because the pulling of the steering line also introduces a torsional deformation of the wing, the difference in lift results in a yawing motion that turns

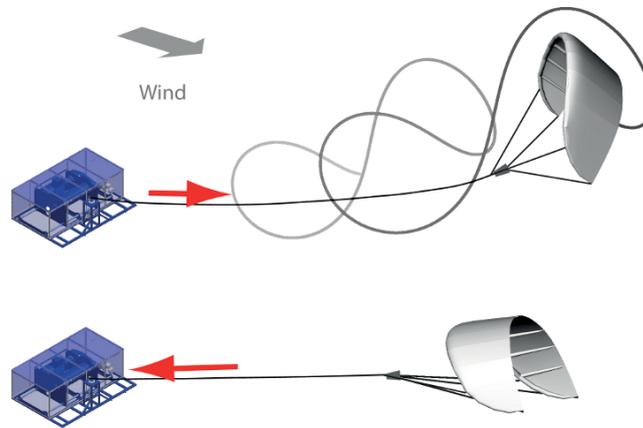
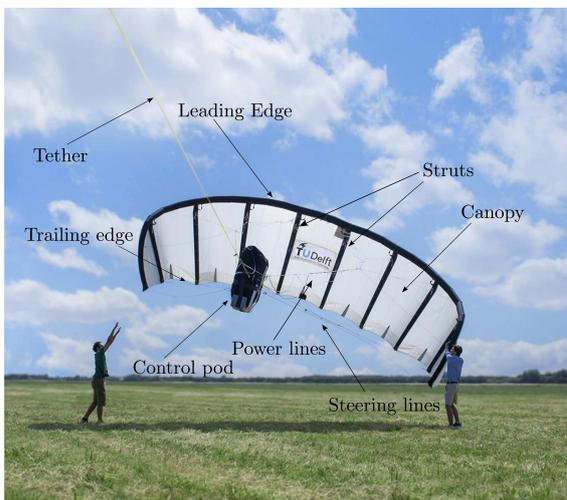


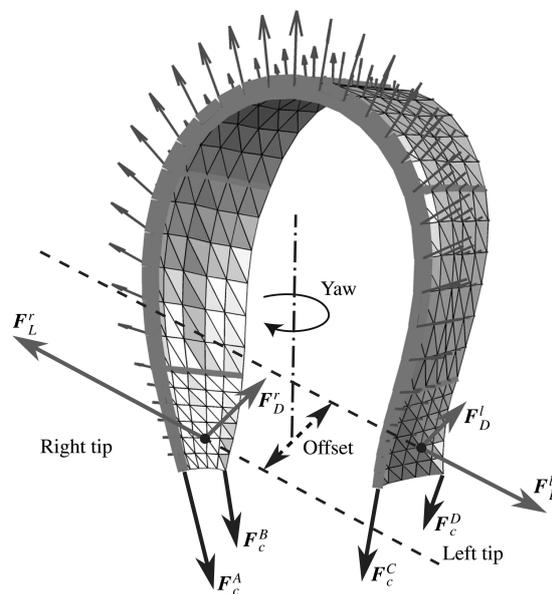
Figure 2-8: Kite pumping cycle [101]

the kite to the right [102]. Breukels refers to this deformation mode as cornering of the kite in turning manoeuvres [12]. Cornering is one example of the strong aero-elastic coupling of a kite system, which illustrates the importance of a proper aero-elastic model to explain the flight behaviour of the kite.

Existing aero-elastic models of the kite In the AWE research group at TU Delft a respectable amount of work has already been done towards modelling the flight behaviour of the kite with varying degrees of complexity. Simple dynamic particle systems are used in combination with parametric aerodynamic models and empirical correlations for control and flight path optimisation purposes [100]. Then, Breukels [12] developed a multi-body kite design toolbox which is positioned somewhere in the middle between a rigid model and a fully detailed FEM model of the kite. Schwoll [103] developed a full Finite Element model of a $25m^2$ kite. The model is detailed enough to not only capture global deformation modes of the kite, but also local deformations such as wrinkling of the canopy. More recently, Bosch [13] developed a coarse Finite Element model of a $16m^2$ kite as compared to the previously constructed model of Schwoll. The solver developed by Bosch combines a nonlinear beam model for the inflated tubes and sewn in wires, as well as a nonlinear shell element for the canopy [102]. Subsequently, Geschiere [14] and Berens [15] worked on further extending this structural model and coupling with simple aerodynamic models. Since the flow field around the kite is quite complex, two highly simplified models have been coupled with the mentioned structural model. First, Breukels [12] developed a parametric model that combines 2D CFD results with a Lifting line code; this model, however, performs quite poorly for different designs than the one it was calibrated with. Second, inviscid methods such as Lifting Line and Vortex Lattice methods show some potential for computationally affordable aero-elastic modelling of the membrane wing. Berens [15] in particular used an unsteady Lifting Line implementation. Then, Van Kappel [104] and later on also Leuthold worked on the implementation of a non-linear Vortex Lattice method for LEI kites; however deformations of the wing were not considered yet.



(a) Launch [15]



(b) Turning [102]

Figure 2-9: Leading Edge Inflatable (LEI) kite in flight during launch and during turning maneuver

Chapter 3

Structural solver

As extensively discussed in the literature review section 2-1, different modelling methodologies are expected to yield satisfactory results for this project. Thus, this chapter will first briefly outline the specific shell element formulation that will be employed in section 3-1. Second and third the implemented solver will be verified on a static test case in section 3-3 and then on a dynamic test case in section 3-4. Lastly, the results of the separate structural solver validation from the Turek & Hron FSI benchmark will be presented in section 3-5.

3-1 Shell element formulation

Given the low bending stiffness of the canopy of the kite, two different Finite Element formulations are possible. First, the least complex element would be a membrane element, that does per definition not possess any bending stiffness. As a consequence, the membrane deformation is undefined for certain load scenarios. For example, no static solution exists for flat membranes which are loaded with normal pressure. Second, a shell element formulation that models in as well as out of plane deformation can also be used if the formulation does not incur any numerical issues at very low element thickness. The advantage of using a shell over a membrane formulation is its versatility, by increasing the thickness of the element also cantilever test cases can be used to verify the solver.

Given the existing code base of a nonlinear shell element formulation in Matlab as developed by Tiso [31] and tested by Bosch [13] as well as Berens [15], this project will try to further build on the work by the mentioned authors. However, the coupling interface of the fluid solver requires the interface coupling to be done in Python, given this also the structural solver will be rewritten in Python.

Tiso's stiffness element formulation [31] is based on a simplified Green-Lagrange strain for-

mulation as outlined below for the strains ϵ_{ab}

$$\epsilon_{xx} = \frac{1}{A} \int_A u_{,x} + \frac{1}{2} (v_{,x}^2 + w_{,x}^2) dA \quad (3-1a)$$

$$\epsilon_{yy} = \frac{1}{A} \int_A v_{,y} + \frac{1}{2} (u_{,y}^2 + w_{,y}^2) dA \quad (3-1b)$$

$$\epsilon_{xy} = \frac{1}{A} \int_A \frac{1}{2} (u_{,y} + v_{,x}) + \frac{1}{2} (w_{,x}w_{,y}) dA \quad (3-1c)$$

and for the curvature χ_{ab}

$$\chi_{xx} = \frac{1}{A} \int_A w_{,xx} dA \quad (3-2a)$$

$$\chi_{yy} = \frac{1}{A} \int_A w_{,yy} dA \quad (3-2b)$$

$$\chi_{xy} = \frac{1}{A} \int_A w_{,xy} dA \quad (3-2c)$$

where the strains as well as the curvature are obtained from an average over the shell area in order to avoid shear locking [31]. Naturally, the displacements and their spatial derivatives are given by (u, v, w) and $(u_{,ab}, v_{,ab}, w_{,ab})$, respectively.

Tiso describes his implementation as a superposition of the triangular linear membrane element by Allman (1976) [105] and the triangular linear plate element also by Allman (1988) [106]. Both element formulations use cubic displacement fields, and each node k of the plate element as well as the membrane element has three degrees of freedom

$$q_{\text{membrane}} = [u_k \quad v_k \quad \theta_{k,z}] \quad \text{where } k = [1, 2, 3] \quad (3-3a)$$

$$q_{\text{plate}} = [w_k \quad \theta_{k,x} \quad \theta_{k,y}] \quad \text{where } k = [1, 2, 3] \quad (3-3b)$$

see also figure 3-1. As a result when the two models are combined each node has six degrees of freedom, such that the full shell element has 18 DOFs.

While nonlinearity in the displacements is incorporated in the analysis, the material is assumed to behave isotropic and linear, such that the constitutive relations with the Young's Modulus E_S and the Poisson ratio ν_S can be written as

$$\sigma = A_m \epsilon \quad (3-4a)$$

$$A_m = \frac{E_S h}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (3-4b)$$

where h refers to the thickness of the shell element. For dynamic analysis of the shell element also the density ρ_S of the element will be needed.

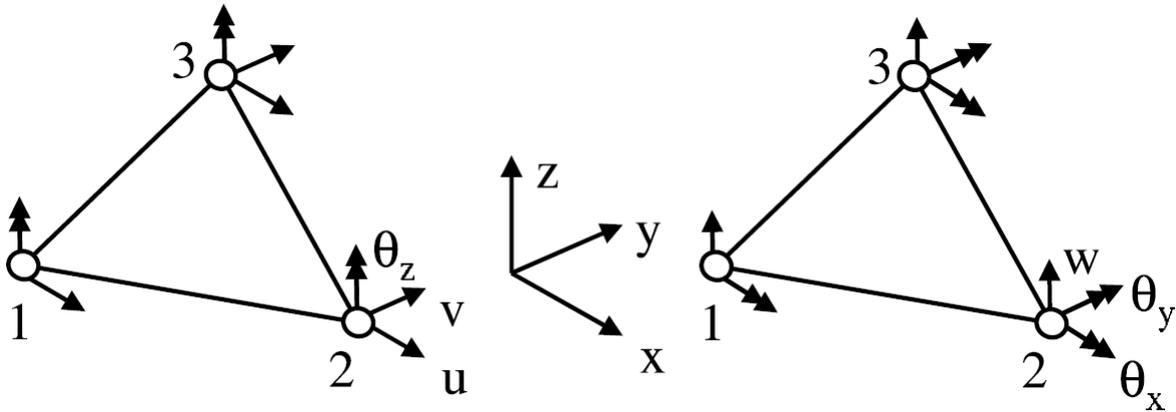


Figure 3-1: Degrees of freedom for Tiso's superimposed shell element from a plate element (left) and a membrane element (right)

The assembly of the mass matrix is done as described by Allman [107] using the same assumed displacement fields as for the stiffness matrix. Allman compares the accuracy of the eigenfrequency prediction using a consistent and a lumped approach and shows that the consistent approach shows a faster grid convergence. However, Allman also shows that using a lumped mass matrix for simple test cases does not incur significant errors in the prediction of the eigenfrequency of the system. More importantly, using a lumped matrix approach allows for the computationally efficient inversion of the mass matrix. Thus, here this will be the preferred approach.

3-2 Detailed solver implementation

A summarized version of the full structural solver algorithm is shown in the text box 1. The following paragraphs outline the different aspects of the full structural solver.

Structural framework The solver will be written in a Total Lagrangian framework where all displacements and stresses are expressed as a deviation from an initial reference configuration. The assembled system matrices then form the following nonlinear system of equations

$${}^{t+\Delta t}M\ddot{\mathbf{q}} + D\dot{\mathbf{q}} + \left({}^t_0K_{\text{linear}} + {}^t_0K_{\text{nonlinear}}(\mathbf{q}) \right) \mathbf{q} = {}^{t+\Delta t}F_{\text{ext}} - {}^t_0F \quad (3-5)$$

where the forcing term t_0F on the right hand side, as well as the linear and nonlinear stiffness matrices t_0K , are a result of the linearization around time t in the Total Lagrangian framework. For the detailed derivation of the system matrices see Bathe (2007) [16].

Boundary conditions For the enforcement of the displacement boundary conditions the full system matrices need to be reduced to the non-constrained nodes only, and the effect of the constrained nodes is moved to the RHS of the equation as illustrated below

$$K\mathbf{q} = \begin{bmatrix} K_{ff} & K_{cf} \\ K_{fc} & K_{cc} \end{bmatrix} \begin{bmatrix} \mathbf{q}_f \\ \mathbf{q}_c \end{bmatrix} = \begin{bmatrix} \mathbf{g}_f \\ \mathbf{g}_c \end{bmatrix} \rightarrow K_{ff}\mathbf{q}_f = \mathbf{g}_f - K_{cf}\mathbf{q}_c \quad (3-6)$$

where the nonlinear stiffness matrix is summarized as $K = {}^t_0 K_{\text{linear}} + {}^t_0 K_{\text{nonlinear}}(\mathbf{q})$ and for simplicity the RHS term g includes damping and inertial forces as well as the external force term.

Time discretisation Based on severe time limit restriction previously faced by Berens [15] due to the use of an explicit time integration scheme, for this project the aim is to use an implicit time integration scheme. The integration of the structure is done by artificially decomposing the system in two first order systems according to

$$\begin{bmatrix} \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -{}^{t+\Delta t}M^{-1}({}^t_0 K_{\text{linear}} + {}^t_0 K_{\text{nonlinear}}(\mathbf{q})) & -{}^{t+\Delta t}M^{-1}D \end{bmatrix} \cdot \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} 0 \\ {}^{t+\Delta t}M^{-1}({}^{t+\Delta t}R - {}^t_0 F) \end{bmatrix} \quad (3-7)$$

where I is the identity matrix sized according to context and the summarized version of the equation is written out below

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B} \quad (3-8)$$

According to Bathe [16] the trapezoidal A-stable, 2nd order integration scheme is frequently used in the time integration of structures, where no intermediate time step information is available. Because this scheme is semi-implicit, sub-iterations have to be performed. However, given the non-linearity of the stiffness matrix, this has to be done independently of the time integration scheme. Thus, the use of an implicit method which allows for larger time steps might prove to be advantageous, even though Berens [15] decided against this option in prequel projects. Applying the trapezoidal time integration scheme to the summarized version of the system as in equation 3-8 results in

$$\begin{aligned} & \frac{\mathbf{x}^{n+1} - \mathbf{x}^n}{\Delta t} - \frac{1}{2} \left(A^{n+1}\mathbf{x}^{n+1} + B^{n+1} + A^n\mathbf{x}^n + B^n \right) = \\ & \left(\frac{1}{\Delta t}I - \frac{1}{2}A^{n+1} \right) \mathbf{x}^{n+1} + \left(-\frac{1}{\Delta t}I - \frac{1}{2}A^n \right) \mathbf{x}^n + \frac{1}{2} \left(B^n + B^{n+1} \right) = 0 \end{aligned} \quad (3-9)$$

Again the equations can be further summarized to

$$M_1 \left(\mathbf{x}^{n+1} \right) \mathbf{x}^{n+1} + M_2 \left(\mathbf{x}^n \right) \mathbf{x}^n + N = 0 \quad (3-10)$$

Nonlinear solving procedure Given the nonlinearity of the time integration scheme and the finite element formulation, the system of equations has to be solved iteratively. Here this is done using a Newton-Raphson scheme as described below

$$G(\mathbf{x}) = M_1(\mathbf{x})\mathbf{x} + M_2(\mathbf{x}^n)\mathbf{x}^n + N \quad (3-11a)$$

$$G(\mathbf{x} + \delta\mathbf{x}) = G(\mathbf{x}) + \frac{\partial G}{\partial \mathbf{x}} \delta\mathbf{x} + O(\delta\mathbf{x}^2) \stackrel{!}{=} 0 \quad (3-11b)$$

$$-\frac{\partial G}{\partial \mathbf{x}} \delta\mathbf{x} = G(\mathbf{x}) \quad (3-11c)$$

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \delta\mathbf{x} \quad (3-11d)$$

where the full exact expression of the Jacobian is given by

$$J = \frac{\partial G}{\partial \mathbf{x}} = \frac{1}{\Delta t} I - \frac{1}{2} \begin{bmatrix} 0 & I \\ -{}_{-t+\Delta t}M^{-1} & {}_tK_t - {}_{-t+\Delta t}M^{-1}D \end{bmatrix} \quad (3-12)$$

the the tangential stiffness matrix K_t is accordingly given by

$$K_t = {}_tK_{\text{linear}} + \frac{\partial {}_tK_{\text{nonlinear}}(\mathbf{q})}{\partial \mathbf{q}} \mathbf{q} + {}_tK_{\text{nonlinear}}(\mathbf{q}) + \frac{\partial {}_tF}{\partial \mathbf{q}} \quad (3-13)$$

For this project full Newton-Raphson iterations are performed, this means that the system matrices and the Jacobian are updated for every sub-iteration and that the step size is not modified. Possibly looking into more sophisticated methods also in combination with the load control algorithm could lead to a more optimized solver [108]. However, this is outside of the scope of this project.

Structural damping The structural damping is determined using a Rayleigh approach where the damping is derived from the stiffness and inertial characteristics of the system. The damping matrix is simply taken to be a weighted linear superposition of the stiffness and mass matrix according to

$$D = \alpha M + \beta K \quad (3-14)$$

Berens [15] implemented the algorithm for calculation of the proportionality constants α and β according to Alipour and Zareian [109]. For Rayleigh damping given the proportionality constants, the mode specific normalized damping ζ_n of an eigenmode with frequency ω_n is given by

$$\zeta_n = \frac{\alpha}{2\omega_n} + \frac{\beta\omega_n}{2} \quad (3-15)$$

If one wants to apply a general structural damping based on the most important (or lowest) eigenmodes with frequencies $\omega_1 - \omega_N$ and not a mode-specific damping where a decoupling of

the system in its different eigenmodes is necessary, expressions for the proportionality need to be derived. For the derivation of the expression a range parameter R is introduced

$$R = \frac{\omega_N}{\omega_1} \quad (3-16)$$

Based on this the proportionality constants α and β can be calculated as

$$\alpha = 2\zeta\omega_1 \frac{2R}{1 + R + 2\sqrt{R}} \quad (3-17a)$$

$$\beta = 2\zeta \frac{1}{\omega_1} \frac{2}{1 + R + 2\sqrt{R}} \quad (3-17b)$$

where ζ is the assumed general damping coefficient [109]. For structures with small relative thickness, generally, a damping coefficient between 2 % and 5% is considered appropriate [15].

Figure 3-2 shows a cantilever oscillation linearized around the zero deflection point for a damping ratio of 10 % and different ranges of considered eigenfrequencies. The simulation settings are listed in table 3-3. In total the beam simulation has 870 unconstrained degrees of freedom, in this particular case considering roughly only the lowest 4 % of the eigenfrequency range of the beam already yields converged results in terms of the structural damping.

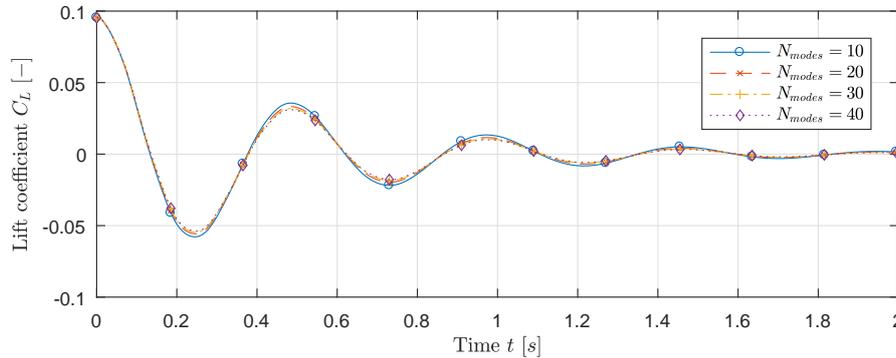


Figure 3-2: Tip displacement for linearized cantilever oscillation with a damping constant of 10 % and different ranges of considered eigenfrequencies of the linearized system

Since the shell element formulation is nonlinear the damping matrix indirectly depends on the deformation of the structure through the stiffness matrix; it needs to be updated when the stiffness matrix is updated. Thus, the eigenmode calculation of the structure needs to be carried out multiple times throughout the solving procedure in one time step.

Load control algorithm For stability and faster convergence of the structural solver especially for larger time steps, a load control algorithm is used [108]. The algorithm relaxes the external load from the previous time step to the external load from the new time step. Depending on the convergence of the structure to the newly imposed external load, the relaxation parameter is either increased or decreased. A power law is used for the expansion of the load increment; the tuning parameters were directly taken from Berens [15]. Again,

much more sophisticated methods are available and could yield an increase in efficiency and stability of the solver [108]. However, this is outside of the scope of this project and for small structural test cases also not strictly necessary.

Convergence criteria Different absolute and relative error criteria were investigated. Depending on the simulation settings and the characteristics of the test case itself some criteria are more efficient than others. The error criteria for the time step $n + 1$ is chosen based on the residual of the discretised system $\mathbf{R} = G(x^{n+1})$ and the change in displacements between sub-iterations $\delta\mathbf{q}$ according to

A Averaged absolute residual $\sqrt{\mathbf{R}^T \mathbf{R}} \leq \epsilon_{conv}$

B Averaged relative residual $\sqrt{\mathbf{R}^T \mathbf{R}} \leq \epsilon_{conv} \sqrt{\mathbf{R}_{ref}^T \mathbf{R}_{ref}}$

C Maximum residual $\max(\mathbf{R}) \leq \epsilon_{conv}$

D Averaged relative work $\delta\mathbf{q}^T \mathbf{R} \leq \epsilon_{conv} \delta\mathbf{q}_{ref}^T \mathbf{R}_{ref}$

E Relative change in node displacement $\delta\mathbf{q} < \epsilon_{conv} \mathbf{q}_{ref}$

where the reference values refer to the first sub-iteration of the structural solver for a corresponding time step. Bathe, and Cimento [110] suggest that for more rigid structures force based criteria should perform well, whereas for more flexible structures such as membranes energy-based criteria should be more efficient.

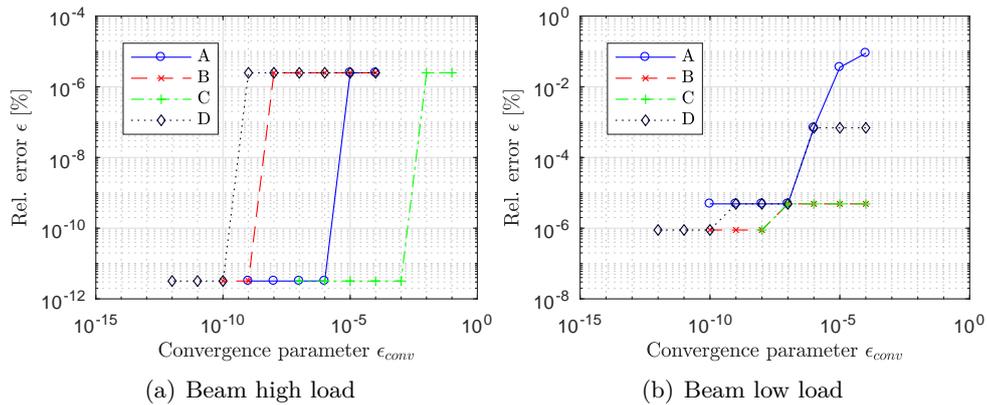


Figure 3-3: Convergence criteria

Figure 3-3 shows the dependence of the convergence on the error criteria for the test case of a uniformly loaded beam with two load magnitudes. The convergence of the solution is assessed through comparison with results checking the convergence of the tip displacement with a very strict convergence criterion. The figure suggests some reasonable values for the convergence parameter for all error criteria, while also outlining that this is test case dependent.

Algorithm 1 Structure convergence loop

Require: Initial condition for \mathbf{q}^n , $\dot{\mathbf{q}}^n$ and external load from previous and current time step \mathbf{F}_{ext}^n , \mathbf{F}_{ext}^{n+1}

Initialize structure $\mathbf{q} = \mathbf{q}^n$, $\dot{\mathbf{q}} = \dot{\mathbf{q}}^n$

while $\lambda < 1$ and $j < j_{max}$ **do**

Update load control parameter $\lambda(i, \epsilon_{NR})$ and reset counter $i = 0$

Relax external load $F_{ext} = \lambda \mathbf{F}_{ext}^{n+1} + (1 - \lambda) \mathbf{F}_{ext}^n$

Initialize residual $\mathbf{R} = F_{int}(\mathbf{q}^n, \dot{\mathbf{q}}^n, \mathbf{q}, \dot{\mathbf{q}}) - \mathbf{F}_{ext}$ and Jacobian $J(K_t(\mathbf{q}), D(\mathbf{q}, \dot{\mathbf{q}}))$

while $\epsilon_{NR} < \text{tol}$ and $i < i_{max}$ **do**

Newton Raphson iteration $\mathbf{dx} = J^{-1} \cdot \mathbf{R}$

Update structure $\mathbf{x} = \mathbf{x} + \mathbf{dx} \rightarrow \mathbf{q}, \dot{\mathbf{q}}$

Update stiffness, tangential stiffness and damping matrix $K(\mathbf{q}), K_t(\mathbf{q}), D(\mathbf{q}, \dot{\mathbf{q}})$

Update residual $\mathbf{R} = F_{int}(\mathbf{q}^n, \dot{\mathbf{q}}^n, \mathbf{q}, \dot{\mathbf{q}}) - \mathbf{F}_{ext}$ and Jacobian $J(K_t(\mathbf{q}), D(\mathbf{q}, \dot{\mathbf{q}}))$

Evaluate error $\epsilon_{NR} = \frac{\mathbf{R}^T \mathbf{R}}{\mathbf{R}_{ref}^T \mathbf{R}_{ref}}$

$i = i + 1$

end while

$j = j + 1$

end while

3-3 Static validation

For the static validation of the solver, two benchmark cases are used. Subsection 3-3-1 details the validation if the membrane is subjected to a uniform pressure load and subsection 3-3-2 uses classical shell benchmark cases as available from the literature.

3-3-1 Uniformly loaded membrane

Similar to Berens, a limited static validation analysis of the solver is carried out based on analytical results by Levy [68, 69] for the deflection of a square plate under a uniform pressure load as described by the van Karman strain equations (see appendix A). He verified his results through comparison with other approximate solutions as well as experimental results by Kaiser [111]. Two different boundary conditions, fully clamped and simple supported are investigated in the two publications. The fully clamped boundary conditions do not allow either translation or rotation at the plate boundary. The simply supported boundary conditions do not allow translation but permit rotation of the nodes at the plate boundary. Levy presented his results as a function of normalized displacements $\bar{w} = \frac{w}{h}$ and normalized load $\bar{p}_z = \frac{p_z L^4}{Eh^4}$. See figure 3-4 for illustration of the three-dimensional deformed plate and figure 3-5 for visualization of the boundary conditions.

Young's Modulus	E_S	$7e10N/m^2$
Poisson ratio	ν_S	0.316
Plate dimensions	L_x, L_y	1m

Table 3-1: Parameters

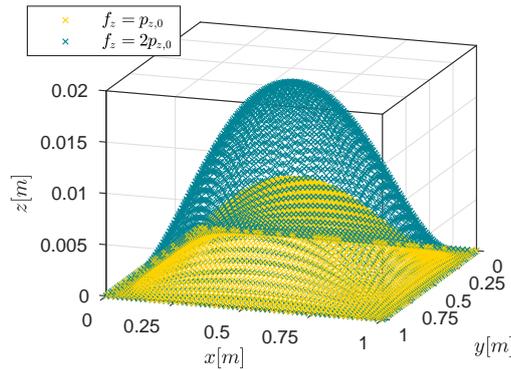


Figure 3-4: Clamped membrane uniformly loaded

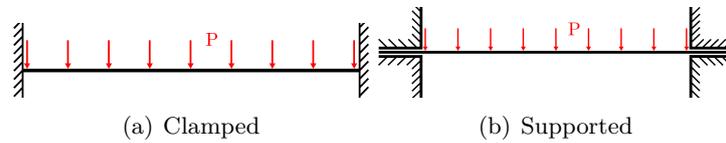


Figure 3-5: Boundary conditions uniformly loaded membrane [15]

The results from the solver implementation in this report for the parameters listed in table 3-1 are compared with the analytical solution from Levy [68, 69] and simulation results from Berens [15]. Figure 3-6 shows the normalized center displacement as a function of the normalized uniform pressure load for two different resolutions alongside with the analytic results from Levy for both of the mentioned boundary conditions. While the trend between the simulation results and the reference data agree well for both boundary conditions, however the error in the prediction of the center displacement grows as the load increases. This is also observed in the results from Berens [15].

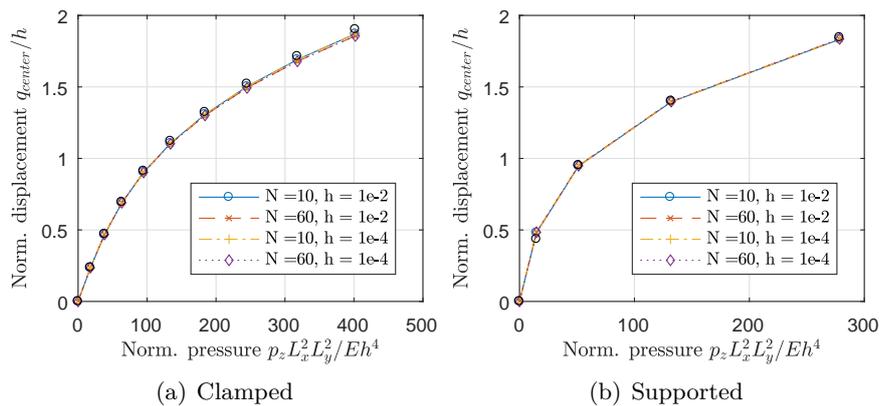


Figure 3-6: Pressure-displacement plot with the analytic solution from Levy in black

Figure 3-7 illustrates why the deviation between the center displacement results increases for higher loading of the structure, namely there seems to be a discrepancy in how the structure deforms under the load. The cause of this is most likely the shell element formulation that

differs in the assumptions made for the derivation of nonlinear plate theory in two aspects. First, the Kirchhoff condition of perpendicular transverse normals even after deformation is not enforced. Second, since it is a shell element in-plane shear deformations are also modeled which could lead to a different load response.

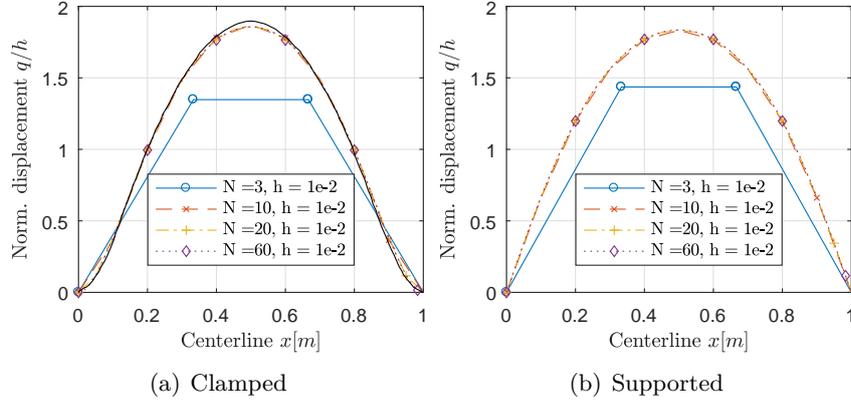


Figure 3-7: Displacement plot for $\bar{p}_z = 402.0$ in the symmetric cutting plane of the plate, analytical solution in black

However, as shown in figure 3-8 a grid refinement study shows a clear convergence of the results towards higher resolution results. The number of nodes was chosen coinciding such that no interpolation has to be carried out when comparing the results from different grid densities. The RMS error is computed according to equation 3-18. Based on figure 3-8, it can be concluded that the implemented shell element formulation is consistent for this particular test case and the discrepancy with analytical results is most likely due to different modelling assumptions. Moreover, given that for normalised load and displacements the results are not sensitive to the shell thickness, it can also be reasoned that the shell element is indeed shear locking free as claimed by Tiso [31], since a decrease in thickness does not result in a nonphysical increase of the element stiffness.

$$\epsilon = \sqrt{\frac{1}{N} \sum_N \left(\frac{w_N - w_{N_{ref}}}{w_{N_{ref}}} \right)^2} \quad (3-18)$$

3-3-2 Shell benchmark case

Sze et al. (2004) [112] list classical benchmark problems for shell elements. Tiso's shell element [31] is tested on the two simplest listed cases, namely a clamped cantilever either loaded with a shear force F_{tip} or an out of plane bending moment M_{tip} at the tip. The results are shown in figure 3-9. The details of setup are listed in table 3-2.

Little correspondence between the results is obtained, because the shell element formulation neglects transverse shear stresses. While this assumption is valid for very thin structures such as a canopy of the kite, for thicker structures as in the benchmark case undergoing large displacements this is not given. Thus, a large discrepancy between the results exists.

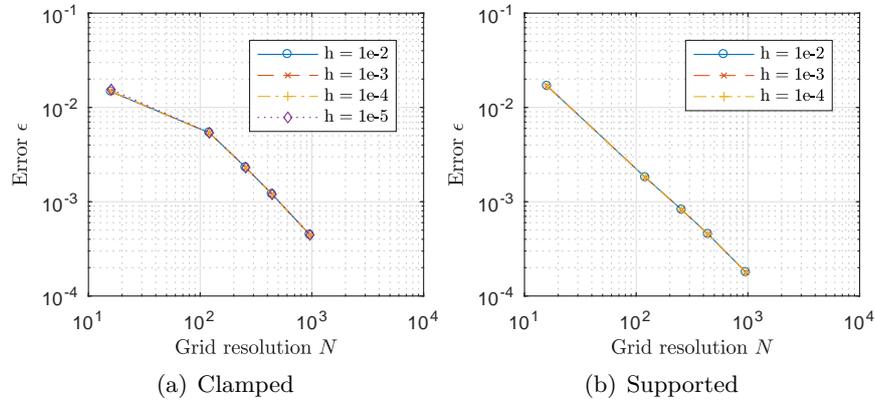


Figure 3-8: Grid convergence for $\bar{p}_z = 402.0$ where the grid resolution corresponds to the number of nodes N and the RMS error is computed with reference to a higher resolution simulation

Young's Modulus	E_S	$1.2e6 N/m^2$
Poisson ratio	ν_S	0.0
Beam dimensions	(L_x, L_y)	$(10m, 1m)$
Beam discretisation	(N_x, N_y)	$(variable, 1)$

Table 3-2: Shell benchmark parameters

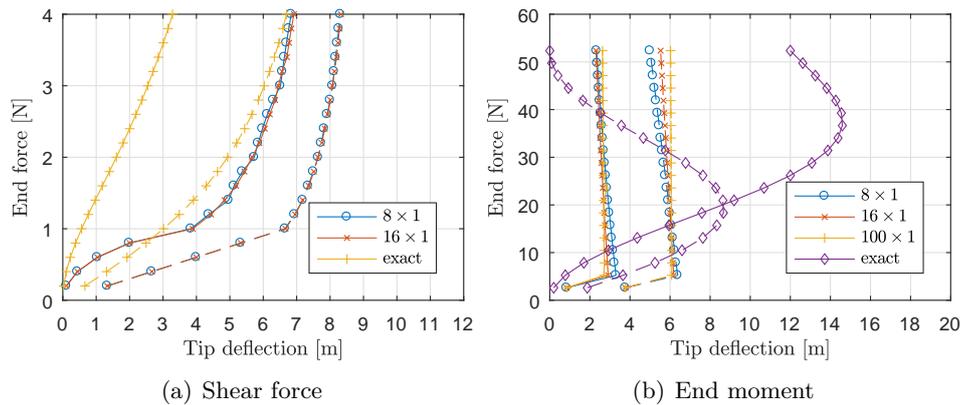


Figure 3-9: Shell element validation, displacements in the direction of the undeformed beam axis (-) and perpendicular to this axis (-)

Concluding, this benchmark case clearly showcases the limits of the sensible application range for this particular shell element formulation. Namely, it will only yield accurate results for relatively thin structures. This will be touched upon again in section 3-5.

3-4 Dynamic validation

The dynamic validation of a structural solver, in general, can be split into different subtasks, namely the validation of the implemented time integration scheme, the verification of the damping matrix assembly and the validation of the shell element formulation itself.

In the context of this project, a limited verification of all of these aspects will be carried out.

3-4-1 Linear spring-damper system

The validation of the implemented time integration scheme is done through integration of the dynamic of simple linear spring-damper systems with either one or two mass points as depicted in figure 3-10. While this may sound rather trivial, given the complexity of the code database it is essential to eliminate the possibility of implementation faults.

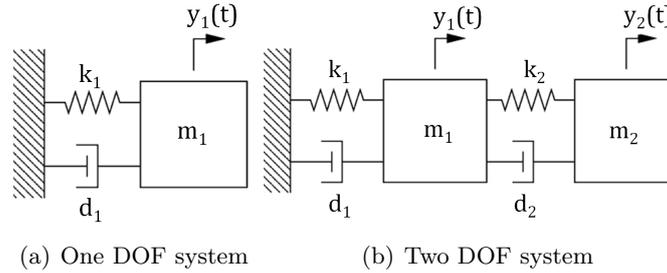


Figure 3-10: Linear spring-damper setup

For a linear, subcritically damped one degree of freedom spring-damper system with a natural frequency of $\omega_0 = \sqrt{\frac{k}{m}}$ and a damping ratio of $\zeta = \frac{d}{2} \sqrt{\frac{1}{mk}}$ without external excitation an analytic solution depending on the damping ratio exists, as written out below

$$m \cdot \ddot{y} + d \cdot \dot{y} + k \cdot y = 0 \quad (3-19a)$$

$$y(t) = \begin{cases} A \cos(\omega_0 t + \phi), & \text{for } \zeta = 0 \\ A \exp^{-\zeta \omega_0 t} \cos(\omega t + \phi), \omega^2 = \omega_0^2(1 - \zeta^2), & \text{for } 0 < \zeta < 1 \\ A \exp^{-\zeta \omega_0 t} + B t \exp^{-\zeta \omega_0 t}, & \text{for } \zeta = 1 \\ \exp^{-\zeta \omega_0 t} (A \cosh(\Psi t) + B \sinh(\Psi t)), \Psi^2 = \omega_0^2(\zeta^2 - 1), & \text{for } \zeta > 1 \end{cases} \quad (3-19b)$$

where A , B and ϕ are constants to be determined by initial conditions.

Comparison between the analytic solution, numeric results from the structural solver and the direct numerical integration are shown in figure 3-11. Direct numerical integration refers to

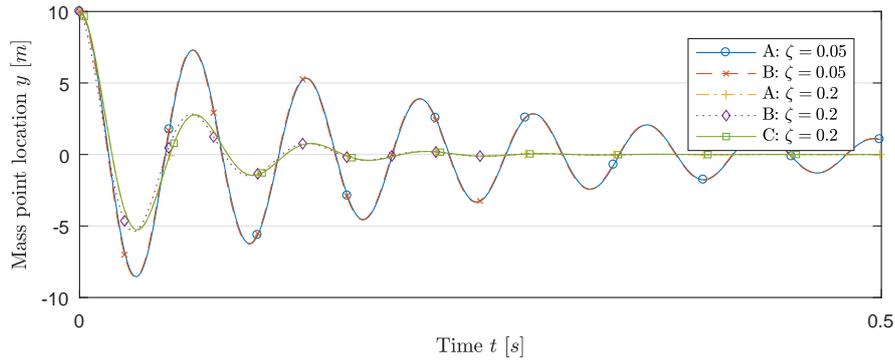


Figure 3-11: One point mass system with an initial position of $y_0 = 10m$ and a natural frequency of $\omega_0 \approx 90rad/s$, (A) refers to simulations results, (B) is the analytic solution and (C) is directly numerically integrated for a time step of $\Delta t = 1e - 4s$

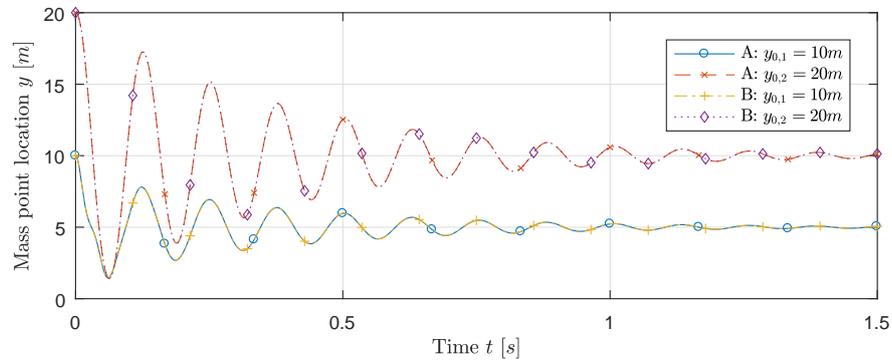


Figure 3-12: Two point mass system

directly integrating the system matrices without the surrounding structural solver framework such that the only error source is the accuracy of the time integration scheme.

For a damping ratio of $\zeta = 0.05$ the analytic solution and the simulation results agree very well, however for a higher damping ratio of $\zeta = 0.2$ some deviations are visible especially for the first oscillation. To identify the cause of this the system matrices were directly numerically integrated and those results overlap with the one from the structural solver, thus it follows that this is due to the accuracy of the time integration scheme.

Then to check whether the solver performs as expected also for a system with coupled degrees of freedom a two degrees of freedom system according to figure 3-10(b) is compared with direct numerical integration results as well, since here no analytical solution is available. The results are shown in figure 3-12 and again the results overlap.

Thus, from this verification on linear spring-damper systems, it can be concluded that the time integration scheme and the load algorithm are working as expected.

3-4-2 Linear cantilever oscillation

An unloaded cantilever clamped on one side with an initial displacement according to the static solution of a uniform pressure load p_0 is investigated. Initially, the nonlinear terms

in the stiffness matrix formulation are removed. The material parameters and simulation settings are listed in table 3-3.

E_S	$7e10N/m^2$
ν_S	0.316
ρ_S	$2780kg/m^3$
h_s	0.01m
(L_x, L_y)	(1m, 2m)
(N_x, N_y)	(5, 10)
Δt	1e-4s
p_0	$500N/m^2$

Table 3-3: Fixed parameters

If only the linear part of the stiffness matrix is considered and the damping matrix is constructed via the Rayleigh method, then an analytic solution exists. Similar to one degree of freedom spring damper system in equation 3-19 the system of equations for the linear multiple degrees of freedom system can be decoupled and for each mode the one degree of freedom analytic solution still holds. The decoupling of the system is done as follows

$$M\ddot{\mathbf{q}} + K\mathbf{q} = 0 \rightarrow \ddot{\xi}_i + \omega_{0,i}^2\xi_i = 0 \quad (3-20a)$$

$$D = \alpha M + \beta K \rightarrow \zeta_i = \frac{1}{2} \left(\frac{\alpha}{\omega_{0,i}} + \beta\omega_{0,i} \right) \quad (3-20b)$$

$$M\ddot{\mathbf{q}} + D\dot{\mathbf{q}} + K\mathbf{q} = 0 \rightarrow \ddot{\xi}_i + 2\omega_{0,i}\zeta_i\dot{\xi}_i + \omega_{0,i}^2\xi_i = 0 \quad (3-20c)$$

Figure 3-13 shows the simulation results alongside with the analytical solution and the directly numerically integrated results for different damping ratios and the settings as specified in table 3-3. Given that the results coincide, at least for the linearized solver, the damping procedure is correctly implemented.

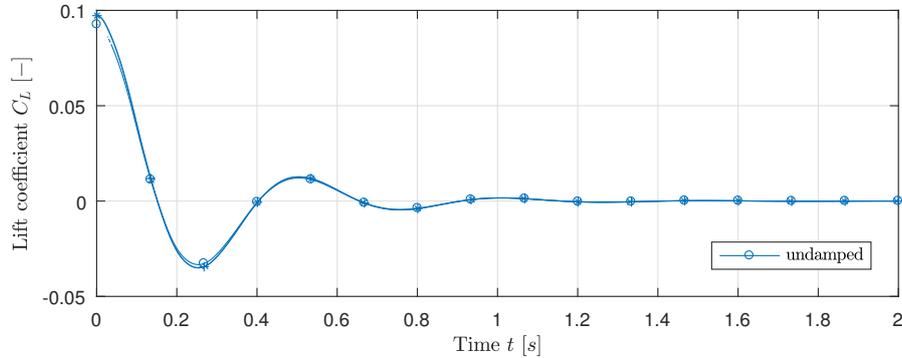


Figure 3-13: Tip displacement for linearized cantilever oscillation for different damping coefficients, analytic solution (o), simulation results (+) and direct numerical integration (x)

3-5 Turek & Hron Benchmark

The Turek benchmark case for FSI solvers was already mentioned in the literature review in chapter 2. For the separate validation of the structural solver without the presence of a fluid, reference data from Turek et al. [66] is available for three different configurations. Thus only the flexible beam part of the structure is relevant, and the only force acting on the beam is gravity with acceleration constant g . The simulation parameters are shown in table 3-4. The first and the second test case are the static deflection results, and the third test case refers to the unsteady transient.

	CSM1	CSM2	CSM3
ρ_S [kg/m^3]	10^3	10^3	10^3
ν_S [-]	0.4	0.4	0.4
E_S [N/m^2]	$1.4e6$	$5.6e6$	$1.4e6$
g [m/s^2]	2	2	2

Table 3-4: Turek CSM Benchmark

A limited grid and error criteria sensitivity study was carried out for both of the static deflection cases and the results are shown in figure 3-14 and 3-15, respectively. The convergence and accuracy of the results are assessed through comparison with the benchmark results for the tip deflection in streamwise u_x and cross-streamwise u_y direction. The relative error is calculated according to

$$\epsilon_{u_x} = \left| \frac{u_x - u_{x,Ref}}{u_{x,Ref}} \right| \quad (3-21a)$$

$$\epsilon_{u_y} = \left| \frac{u_y - u_{y,Ref}}{u_{y,Ref}} \right| \quad (3-21b)$$

The most apparent finding from the grid convergence study is that for very high mesh refinement the structure becomes artificially stiffer independent of the choice of the error criteria. Since it is not feasible to run at such a high resolution anyways, the origin of this was not further investigated. But it seems that this is related to the nonlinear shell element formulation itself. Moreover, for the CSM1 settings the relative error of the displacement normal and tangent to the inflow is at best 2% and 5%, respectively, as compared to the benchmark results from Turek et al. [66]. Since for the benchmark result the full continuum mechanics equations are solved in the domain of the beam, the benchmark results are expected to deviate from the results obtained with the nonlinear shell elements and be more accurate because the full strain formulation was used. For the derivation of the shell element, some of the components of the strain formulation were neglected for the sake of simplicity and thus this results in the observed difference in results.

For the CSM2 settings which results in a stiffer structure as compared to the CSM1 settings, the relative error as compared to the benchmark results is smaller, because the displacements are smaller and thus the neglected strain terms are smaller than for the larger deflection case.

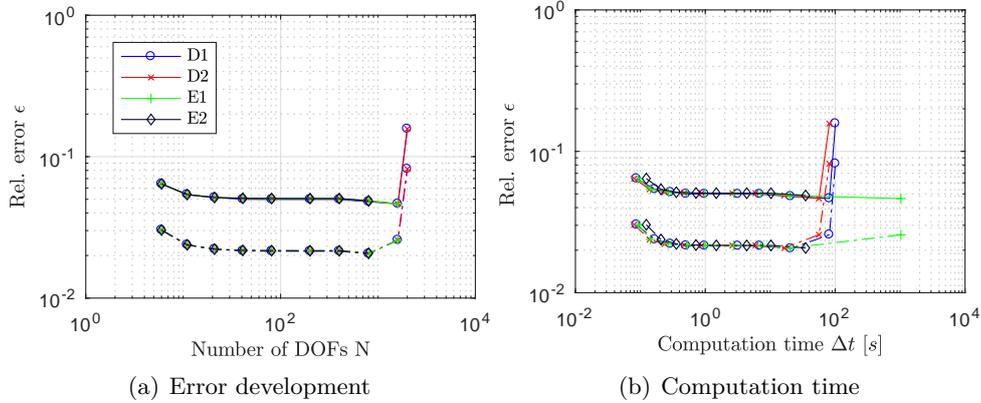


Figure 3-14: CSM1 grid convergence for four different error criteria for the streamwise (-) and the cross-streamwise (-) displacements with a work (D) and relative change in tip node displacement (E) based error criterias with convergence constants $\epsilon_{conv,D1} = 1e-5$, $\epsilon_{conv,D2} = 1e-8$, $\epsilon_{conv,E1} = 1e-5$ and $\epsilon_{conv,E2} = 1e-8$

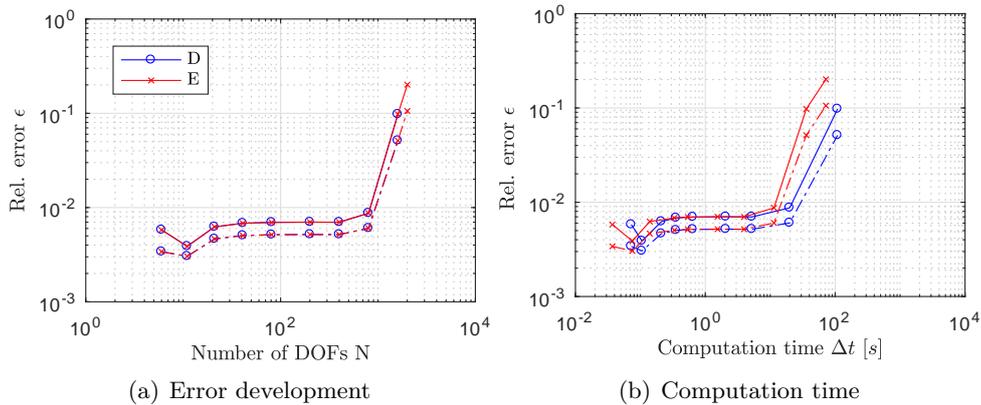


Figure 3-15: CSM2 grid convergence for four different error criteria for the streamwise (-) and the cross-streamwise (-) displacements with a work (D) and relative change in tip node displacement (E) based error criterias with constants $\epsilon_{conv,D} = 1e-5$ and $\epsilon_{conv,E} = 1e-5$

		ϵ_{u_x}	ϵ_{u_y}
CSM1	nonlinear	5.06%	2.16%
CSM1	linear	not predicted	2.16%
CSM2	nonlinear	-0.70%	-0.51%
CSM2	linear	not predicted	-0.51%

Table 3-5: Summary static deflection results

The results from the grid sensitivity study indicate that using an error criteria based on the relative change of the tip displacement (E) with a constant of $\epsilon_{conv} = 1e - 5$ seems to yield converged results at acceptable computational cost. Moreover, using 80 elements to discretise the beam are also found to be sufficient. So these convergence settings will be used for the CSM3 test case. Table 3-5 summarises the results for the CSM1 and CSM2 test cases using both the linear and nonlinear shell element formulation for the mentioned convergence criteria. The linear element formulation fails to capture the displacement in the direction tangent to the inflow because deformation along the axis of the beam due to bending is not modelled in the linear element description.

	$\epsilon_{\bar{u}_x}$	$\epsilon_{\Delta u_x}$	$\epsilon_{\bar{u}_y}$	$\epsilon_{\Delta u_y}$
$\Delta t = 0.02s$	14.31%	14.21%	6.48%	5.00%
$\Delta t = 0.01s$	14.91%	14.80%	6.56%	5.28%
$\Delta t = 0.005s$	14.59%	14.45%	6.33%	5.35%
$\Delta t = 0.001s$	14.76%	14.63%	6.44%	5.35%

Table 3-6: Summary transient deflection results

For the transient test case the mean \bar{u}_x, \bar{u}_y and the amplitude of the deflection $\Delta u_x, \Delta u_y$ over ten oscillations is used to quantify the results for comparison with the benchmark results. The transient simulations are carried out for three different time steps as for the benchmark cases and additionally also one smaller time step is used to see whether closer correspondence with the benchmark data can be obtained. The relative error as compared to the reference results is summarised in table 3-6. Only the nonlinear element formulation was used, because as already mentioned the linear formulation does not model displacements along the shell axis. As compared to the static deflection case of the beam, the relative deviation from the benchmark results increases by about a factor of three. This is not ideal for the validation of the aeroelastic solver in terms of error quantification, but given that the error mainly arises from the limitation of the element formulation within the context of this project not much can be done about this.

Figure 3-16 shows the time domain development of the tip displacements in the same format as shown in the reference paper by Turek et al. [66]. One can see that the influence of the time step on the phase and amplitude of the system response is small once the time steps become smaller than $\Delta t = 0.02s$. However, as compared to the benchmark results apart from the deviation in amplitude and mean of the tip displacement, the frequency of the oscillation is also offset by about 10 % although not explicitly quantified in the reference paper. This could be in part due to the simplified stiffness formulation of the element, but also due to the formulation of the mass matrix. In particular, for the assembly of the mass matrix mass lumping is used according to Hinton et al. [113], which can lead to slower convergence of the elements to the real inertial behaviour of the structure than using a consistent formulation. Moreover, judging by eye and also observed in the Fourier transform of the signal, the oscillation has two dominant frequencies, whereas the benchmark results seem to have only one dominant frequency.

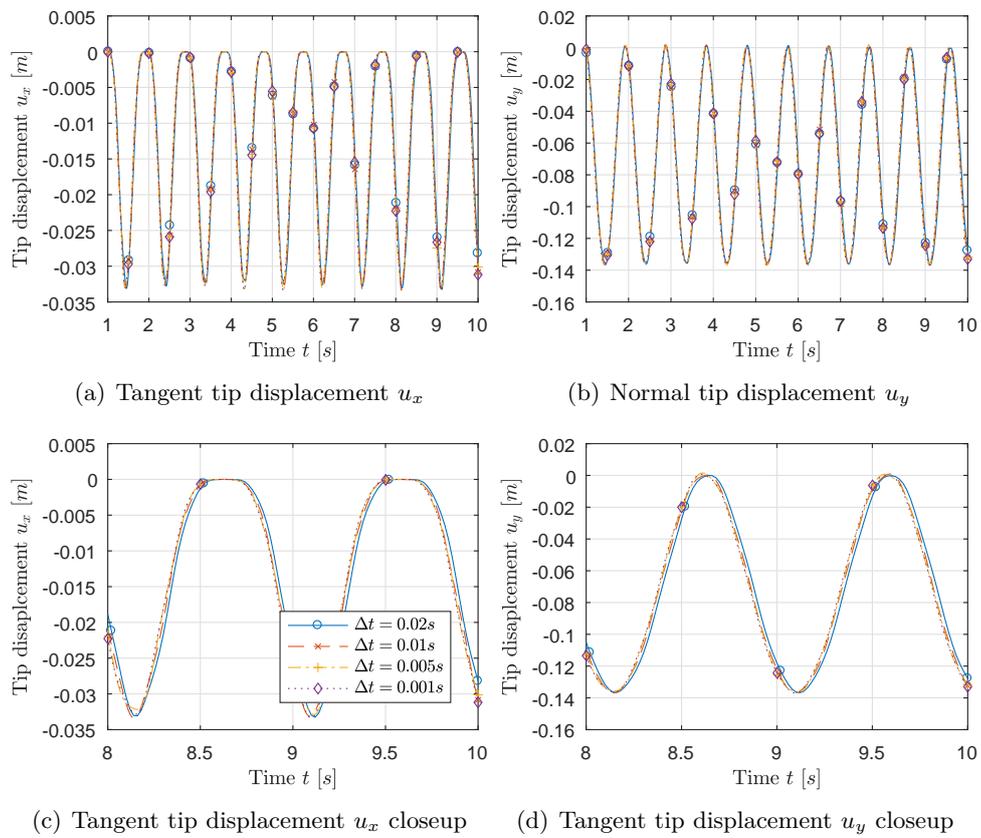


Figure 3-16: Tip displacement development of the CSM3 test case for different time steps

Chapter 4

Fluid solver

This chapter will elaborate on the implementation details of the fluid models used within the context of this thesis alongside with a limited validation on benchmark cases. First, section 4-1 briefly outlines the inviscid method that was used for the debugging and initialisation of the structure model before higher fidelity fluid models were used. Second and third, sections 4-2 and 4-3 focus on the two Computational Fluid Dynamics (CFD) approaches that were employed for this project. The second approach uses a more novel Immersed Boundary approach whereas the third one uses the conventional body-fitted mesh approach.

Because, as will be described in detail in the corresponding section, some unrecoverable issues with the Immersed Boundary approach were discovered, a more mature solver framework was then considered and used for the application to the fully coupled simulations.

4-1 Model 1: Vortex particle method

Fluid models based on inviscid methods are wildly used in engineering applications in the early design stages, because of their ease of implementation and low computational demands. However, since they do not model viscous effects they are not able to predict boundary layer effects such as transition, separation and skin friction. Consequently, this limits the applicability of the model to low incidence angles where attached flow can be expected. Further, the assumption of an inviscid and irrotational flow implies that there is no vortex decay due to viscous effects, and as such the wake vorticity of the wake will remain constant in time. [114]

For the initial configuration of the structure solver and the coupling algorithms an unsteady Vortex Particle Method is implemented in the Python solver framework. A short overview over the theory behind the method and validation of its steady and unsteady behaviour can be found in appendix B.

4-2 Model 2: Immersed Boundary Computational Fluid Dynamics Approach

For this part of the project, the transient, fully-turbulent Finite-Element solver Fluidity [115] in combination with Menter's $k-\omega$ *SST* turbulence model and an Immersed Boundary treatment is used.

First, subsection 4-2-1 will shortly outline the different implementation specific details of the fluid model. Then, subsection 4-2-2 will show the validation results on a benchmark case for laminar flow.

4-2-1 Implementation details

Mesh One of the advantages of using a Finite-Element discretisation is its performance on unstructured meshes. Moreover, Fluidity is an unstructured solver that is capable of dynamic mesh optimisation which can be used to automatically refine and coarsen the mesh in regions of high and low velocity gradients, respectively. For this thesis, this is especially convenient since it allows the automatic meshing of complicated and deforming structures. The implementation of the mesh adaptivity is briefly summarised in the following, the details of the implementation can be found in [55].

To be able to optimise the mesh, an equally weighted objective based on the element size and shape is defined. Additionally, the Hessians of solution fields - in this case, the velocity - are used to calculate an element-specific metric. In turn, this is used to evaluate the contribution of the element size to the optimisation objective. Based on an element's objective value, one or more modifications in the form of edge collapsing, edge splitting, face to edge swapping, edge swapping and node movement are performed on the elements locally. Once the optimisation procedure has been carried out, a mesh with appropriately sized and shaped elements should be obtained. Finally, the field data from the initial mesh nodes is projected to the optimised mesh nodes using a conservative Galerkin projection method.

Discretisation Because the application of Fluidity to aerodynamic flows is still in development, the implemented options are still limited. For this project, piecewise-linear finite element basis functions will be used for the weak Galerkin formulation in space and the second order accurate Crank-Nicolson scheme will be used for temporal discretisation.

The fully discretised momentum and mass conservation equations are shown below

$$\left(M + \frac{1}{2}\Delta t(A - D)\right) \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = - (A - D)\mathbf{u}^n + Cp^{n+1/2} + f^{n+1/2} \quad (4-1a)$$

$$C^T \mathbf{u}^{n+1} = 0 \quad (4-1b)$$

where the derivation of the matrices from the integration of the shape functions was omitted, moreover the gravity term is included in the force term f on the right-hand side of the momentum equation. For the detailed derivation see [116].

Solution procedure The solution method involves a two-step projection method. In the first step, the full momentum equation is solved for a prediction velocity u_\star^{n+1} using the pressure from the previous time step $p_\star^{n+1/2} = p^{n-1/2}$. In the second step, a correction is applied that adjusts the velocity and pressure field such that the prediction velocity becomes divergence free. The following two equations quantify the correction step

$$\left(M + \frac{1}{2}\Delta t A\right) \frac{\mathbf{u}_\star^{n+1} - \mathbf{u}^n}{\Delta t} = -A\mathbf{u}^n + Cp^{n+1/2} + f^{n+1/2} \quad (4-2a)$$

$$\left(C^T M C + K\right) \Delta p = -\frac{C^T \mathbf{u}_\star^{n+1}}{\Delta t} - K p_\star^{n+1/2} \quad (4-2b)$$

$$M\mathbf{u}^{n+1} = M\mathbf{u}_\star^{n+1} + \Delta t C \Delta p \quad (4-2c)$$

where the pressure stabilisation matrix K is taken according to [117], again see [116] for the details of the implementation. If desired, outer iterations on the coupled momentum-pressure equations can also be specified.

Galerkin projection Since Galerkin projection is used for both the implementation of the mesh adaptivity as well as the immersed boundary methods outlined later on, a short overview over the main advantages will be given in the following based on the description by Farrell and Maddison (2011) [118].

For the mesh adaptivity the projection method should be conservative, otherwise mass, momentum or energy is lost when the mesh is adapted. In particular, as compared to consistent interpolation, Galerkin projection has the advantage of being conservative and well defined also for discontinuous fields, i.e. when a Discontinuous Galerkin representation is used.

The theoretical background behind the projection method is relatively simple and its derivation is shown below for the projection between two overlapping meshes of the same dimension. Given a donor mesh \mathcal{T}_D with nodes \mathcal{N}_D and a target mesh \mathcal{T}_T with nodes \mathcal{N}_T , a field representation on the donor and the target mesh are given by $q_D = \sum_{i=1}^{\mathcal{N}_D} q_D^{(i)} \psi_D^{(i)}$ and $q_T = \sum_{i=1}^{\mathcal{N}_T} q_T^{(i)} \psi_T^{(i)}$, respectively.

The projection error of a field q_D on the donor mesh to a field q_T on the target mesh is optimal in the L_2 norm if

$$\|q_D - q_T\| = \min_{q \in \mathcal{V}_T} \|q_D - q\|_2 \quad (4-3)$$

where $\mathcal{V}_T = \text{span} \left\{ \psi_T^{(i)} \right\}$ for $i \in \{1, \dots, \mathcal{N}_T\}$.

Expanding the norm in terms of the basis functions and the weighting coefficients and then taking the derivative to obtain the optimal solution, results in

$$\int_{\Omega} q_D \psi_T^{(k)} dV = \int_{\Omega} q_T \psi_T^{(k)} dV, \quad \forall k \in \{1, \dots, \mathcal{N}_T\} \quad (4-4)$$

or written in terms of the mass matrix

$$M_{\mathcal{T}}q_{\mathcal{T}} = M_{\mathcal{T}\mathcal{D}}q_{\mathcal{D}} \quad (4-5a)$$

$$(M_{\mathcal{T}})_{ij} = \int_{\Omega} \psi_{\mathcal{T}}^{(i)} \psi_{\mathcal{T}}^{(j)} dV, \quad i, j \in \{1, \dots, \mathcal{N}_{\mathcal{T}}\} \quad (4-5b)$$

$$(M_{\mathcal{T}\mathcal{D}})_{ij} = \int_{\Omega} \psi_{\mathcal{T}}^{(i)} \psi_{\mathcal{D}}^{(j)} dV, \quad i \in \{1, \dots, \mathcal{N}_{\mathcal{T}}\}, j \in \{1, \dots, \mathcal{N}_{\mathcal{D}}\} \quad (4-5c)$$

The integration in eq. 4-5c requires the integration of the product of basis functions defined on two different meshes. Numerical quadrature is only expected to be accurate if the integrand is at least piece-wise continuous. Farrell et al (2009, 2011) [119], [118] constructed a supermesh that creates new elements at intersection regions of the donor and the target mesh. Thus in those intersections, both the shape function on the donor as well as on the target mesh are continuous and well-defined polynomials. The algorithm used to create the supermesh is complex, and thus the description is omitted here. However, it should be noted that the evaluation of the mass matrix integrals is executed locally, such that the algorithm can also be applied to large problems. In fact, as a consequence Farrell and Maddison [118] showed that the computational effort of the whole projection method scales linearly with the problem size. However, since mass lumping is employed, the additional numerical diffusion that is introduced leads to a modification of the bounds of the projected field [119].

Summarising, Galerkin projection is a conservative projection method for overlapping meshes of the same dimension with non-coinciding nodes and non-matching order shape functions. Thus it is ideally suited for mesh adaptivity applications, but can also be used for the interface projection in Fluid-Structure Interaction as will be done in this project.

Turbulence model Menter's $\kappa\text{-}\omega$ *SST* turbulence model will be used for this project. A brief overview will be given in the following, for a detailed description of the model the reader is referred to the original paper by Menter (1993) [120].

In this model Menter combined the already existing $\kappa\text{-}\omega$ and $\kappa\text{-}\epsilon$ two-equation RANS turbulence models and through the addition of a conditional term in the eddy viscosity relation a better representation of the flow in separated regions was achieved.

The $\kappa\text{-}\epsilon$ model by Jones and Launder (1973) [121] performs well for some flow situations but overpredicts the shear-stress in adverse pressure gradient flows. This leads to a delayed flow separation prediction. Moreover, the viscous sublayer formulation is stiff and physically incorrect boundary conditions are prescribed on the dissipation ϵ .

On the other hand, the $\kappa\text{-}\omega$ model by Wilcox (1988) [122] shows a stronger sensitivity to adverse pressure gradients and has a comparatively simple formulation in the viscous sublayer, that results in a more numerically stable formulation. Nevertheless, it also shows a strong non-physical dependency of the eddy viscosity on the freestream specific dissipation rate ω_f .

Menter thus proposed to combine the complementary advantages of the two models in one new Baseline (BSL) formulation. He suggested to apply the $\kappa\text{-}\omega$ model up until roughly 50 % of the boundary layer and then through the use of a blending function switch to the $k\text{-}\epsilon$ model. This was done by reformulating the $\kappa\text{-}\epsilon$ to a $\kappa\text{-}\omega$ formulation and then superimpose the two

transport equations and the model constants through a hyperbolic blending function. While the new blended model outperformed their separate application, separation regions were still underpredicted as compared with measurements.

Based on the main assumption of the algebraic Johnson-King (JK) model [123], that assumes that the turbulent shear-stress is proportional to the turbulence kinetic energy in the logarithmic and wake regions of the turbulent boundary layer, Menter implemented an additional modification to the newly developed Baseline model. This was done through use of a conditional statement in combination with a blending function F_2

$$\nu_t = \frac{a_1 \kappa}{\max(a_1 \omega, \Omega F_2)} \quad (4-6)$$

Here the first term refers to the eddy viscosity formulation as given in the κ - ω formulation, and the second term is taken from the JK model.

Testing the model on different, relevant flow scenarios showed that the κ - ω SST model outperformed all the other mentioned models for all the investigated configurations. However, while the prediction of the onset of the separation in adverse-pressure gradient flows was significantly better predicted, the flow recovery is underpredicted [124]. In fact, all the presented models failed to accurately capture the relaxation of the velocity profiles behind a backward-facing step flow [120].

For an LEI kite profile at a high Reynolds number, a permanent separation bubble exists on the pressure side right behind the inflated leading edge and at large angles attack also another separation bubble on the suction side [125]. For a simple membrane wing without an inflated leading edge, the flow separates at the leading edge on the suction side as well as near the trailing edge for high angles of attack. Depending on the Reynolds number either the leading edge or the trailing edge separation bubble is more dominant. Since the κ - ω SST model was developed specifically for strong adverse pressure gradient flows and has been thoroughly validated, it seems a good choice for this project. However, the literature suggests that it has a tendency to overpredict separation bubbles.

Immersed Boundary treatment Three different Immersed Boundary methods were implemented in Fluidity all of which hinge strongly on the projection of a solid density field from the mesh of the structure to the extended fluid mesh - covering both fluid and structure domain - using Galerkin projection. The first two methods use penalty forcing for the enforcement of the interface location with the first approach by Viré et al. [126] evaluating the force in the whole extended fluid domain, whereas the second approach by Viré et al. [127] only calculates the penalty forcing in a shell mesh localised at the fluid-solid boundary.

The most recent approach by Krishnan et al. [128] employs an implicit direct forcing approach where the interface is directly enforced through the imposition of velocity boundary conditions. Since this formulation will be used for this project, the methodology is outlined in more detail below.

Building on the use of the Galerkin projection in the two previously implemented IB methods in Fluidity, a unity solid concentration field α_s is projected from the structural mesh onto the entire domain of the fluid mesh. Additionally, the structural velocity is also projected onto the fluid mesh. Now two possibilities to enforce the boundary conditions arise.

First, a smeared interface can be constructed if the imposed velocity is obtained from a linear combination of the fluid dynamic equations and prescribed structural velocity weighted by the solid concentration.

Second, if a sharp interface is desired a specific cut-off value for the solid concentration can be defined. For examples, if a node has a solid concentration of above 50 %, then the solid velocity is imposed. Otherwise, no velocity boundary condition is imposed on this specific node. However, due to the nature of the Galerkin projection, in this case, the projected fields will not be conserved.

In comparison with Finite Difference and Finite Volume methods with explicitly imposed velocity boundary conditions, no extra interpolation scheme for the background or the ghost nodes needs to be implemented. The Galerkin projection already takes care of this, because it projects the desired interface velocity from the solid mesh on the nodes of the extended fluid mesh.

In more detail, the method is implemented through the weighting of the test functions with the solid concentration for the case of a smeared interface and through Boolean weighting for the case of the sharp interface

$$\psi_j = \psi_j^f + \psi_j^s = \tilde{\alpha}_{f,j} \cdot \psi_j + \tilde{\alpha}_{s,j} \cdot \psi_j, \psi_j^{s,j} = \beta_j \cdot \psi_j \quad (4-7a)$$

$$\text{Smeared interface: } \begin{cases} \tilde{\alpha}_{f,j} = 1 - \alpha_{s,j}, \tilde{\alpha}_{s,j} = \alpha_{s,j} \\ 0 < \alpha_{s,j} < 1 \rightarrow \beta_j = \alpha_{s,j} \\ \alpha_{s,j} \geq 1 \rightarrow \beta_j = 0 \end{cases} \quad (4-7b)$$

$$\text{Sharp interface: } \begin{cases} \alpha_{s,j} \geq 0.5 \rightarrow \tilde{\alpha}_{f,j} = 0, \tilde{\alpha}_{s,j} = 1, \beta_j = 1 \\ \alpha_{s,j} \leq 0.5 \rightarrow \tilde{\alpha}_{f,j} = 1, \tilde{\alpha}_{s,j} = 0, \beta_j = 0 \end{cases} \quad (4-7c)$$

where ψ_j corresponds to the test function of the element j on the fluid mesh and $\alpha_{s,j}$ is the solid concentration obtained from the Galerkin projection.

The discretised mass and momentum conservation equations as given in equation 4-1, can now be rewritten to

$$\left(M + \frac{1}{2} \Delta t (A - D) \right) \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = - (A - D) \mathbf{u}^n + C p^{n+1/2} + f^{n+1/2} + f^s \quad (4-8a)$$

$$C^T \mathbf{u}^{n+1} = - C^{s,T} \mathbf{u}^{n+1} \quad (4-8b)$$

The velocity and pressure approximation is introduced as

$$\mathbf{u} = (u, v, w)^T \approx \sum_{j=1}^N (u_j, u_{j+N}, u_{j+2N})^T \phi_j \text{ and } p \approx \sum_{j=1}^N p_j \phi_j \quad (4-9)$$

then for the formulation of IB method only the mass matrix and the forcing term need to be

modified to include the velocity prescription from the structure motion

$$\tilde{M}_{i,j} = \int_{V_e} \psi_i^f(\mathbf{x})\phi_j(\mathbf{x}) + \psi_i^s(\mathbf{x})\phi_j(\mathbf{x})dV_e \text{ for } \{i,j\} \in [1,N] \quad (4-10a)$$

$$M = \text{diag}(\tilde{M}, \tilde{M}, \tilde{M}) \quad (4-10b)$$

$$\tilde{f}_i^{s,u} = \sum_{j=1}^N \int_{V_e} \psi_j^s(\mathbf{x})\phi_i(\mathbf{x}) \frac{u_i^{s,n} - u^n}{\Delta t} dV_e \text{ for } \{i\} \in [1,N] \quad (4-11a)$$

$$f^s = [\tilde{f}^{s,u}, \tilde{f}^{s,v}, \tilde{f}^{s,w}]^T \quad (4-11b)$$

The advection and diffusion matrix remain unchanged except for being weighted with the modified test function ψ^f , instead of just the test function ψ . The extension to the fractional step method is analogue to the pure fluid equations.

Lastly, it is interesting to note how the gradient operator C^d - or the divergence operator $C^{d,T}$ - is obtained and applied.

$$\tilde{C}_{i,j}^x = \int_{V_i} \frac{\partial (\psi_j^f + \psi_j^{s,j}) (\mathbf{x})}{\partial x} \phi_j(\mathbf{x})dV_i \rightarrow C = \text{diag}(\tilde{C}^x, \tilde{C}^y, \tilde{C}^z) \quad (4-12a)$$

$$\tilde{C}_{i,j}^{s,x} = \int_{V_i} \frac{\partial \psi_j^{s,j}(\mathbf{x})}{\partial x} \phi_j(\mathbf{x})dV_i \rightarrow C^s = \text{diag}(\tilde{C}^{s,x}, \tilde{C}^{s,y}, \tilde{C}^{s,z}) \quad (4-12b)$$

$$\tilde{C}_{i,j}^{f,x} = \int_{V_i} \frac{\partial \psi_j^f(\mathbf{x})}{\partial x} \phi_j(\mathbf{x})dV_i \rightarrow C^f = \text{diag}(\tilde{C}^{f,x}, \tilde{C}^{f,y}, \tilde{C}^{f,z}) \quad (4-12c)$$

The first gradient operator C in equation 4-12 is applied to the pure fluid nodes and in case of a smeared interface also to all the nodes that lie within the smeared interface. The second C^f and the third C^s gradient operators are applied to the pure fluid and the pure structure nodes on the extended fluid domain, respectively. The distinction between those different operators is important to ensure that the pressure gradient in the interface region is well defined.

Moreover, the continuity equation 4-8b is written such that mass is conserved in the extended fluid domain, such that mass flux between the actual fluid and the structure overlapping fluid domain is possible. But because the structure velocity is prescribed in the overlapping fluid domain and because the fluid is incompressible, the mass in the actual fluid domain is approximately conserved.

4-2-2 Validation

The Turek & Hron benchmark case [66] also provides integrated force coefficients for the separate validation of the fluid solver. Thus, some time was spent on the separate validation Immersed Boundary solver on the non-moving case as well. This is done for two reasons.

First, it will help better understand deviations from the benchmark results for the coupled simulations in particular with regards to the limitations of the Immersed Boundary method. While the Immersed Boundary treatment in combination with mesh adaptivity seems ideal to treat aeroelastic problems with large deformations, the resolution of the boundary layer is expected to be less accurate as compared to a body-conforming mesh. Second, the mesh resolution requirements and the tuning parameters for the mesh adaptivity can already be tested on a computationally less expensive test case with the rigid body.

	CFD1	CFD2	CFD3
$\rho_F [kg/m^3]$	10^3	10^3	10^3
$\nu_F [m^2/s]$	0.001	0.001	0.001
$Re = \frac{\bar{U}d}{\nu_F} [-]$	10	20	200

Table 4-1: Turek CFD Benchmark settings

Again, the illustration of the benchmark case can be found in figure 2-2 in the literature review chapter. Similarly, to the structure benchmark case, for the separate fluid test case three settings with Reynolds numbers of 20, 100 and 200 are investigated. A summary of the settings is shown in table 4-1 where \bar{U} refers to the mean inflow velocity. No slip boundary conditions are imposed on the top, and the bottom wall of the channel. A parabolic velocity profile according to equation 4-13 is imposed on the inlet. A zero mean pressure boundary condition is set at the outlet.

$$u(0, y) = 0 \quad (4-13a)$$

$$v(0, y) = 1.5\bar{U} \frac{y(H-y)}{(H/2)^2} \quad (4-13b)$$

In accordance with the separate structural model validation, the CFD1 and the CFD2 settings will result in a steady solution, whereas the CFD3 test case exhibits periodic unsteady behaviour due to vortex shedding once the initial transient has decayed. For this case, to improve the convergence of the solution, the inlet boundary conditions are imposed incrementally according to the equation below

$$u(0, y, t) = 0 \quad (4-14a)$$

$$v(0, y, t) = 1.5\bar{U} \frac{y(H-y)}{(H/2)^2} \frac{1 - \cos(\pi/2t)}{2} \quad (4-14b)$$

whereas the rest of the flow field is initialised as standing still.

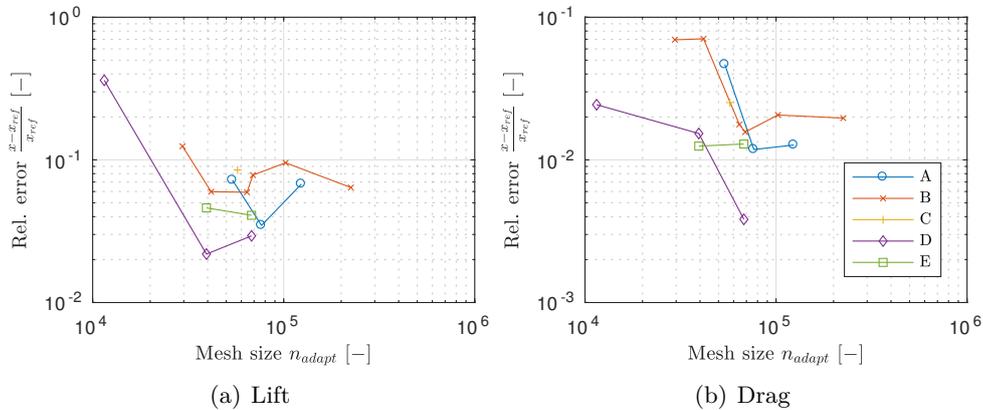
Mesh convergence

A mesh convergence study is carried out for the CFD1 test case by varying the minimum admissible edge length Δx_{min} and the interpolation error bounds of the mesh adaptivity algorithm in Fluidity [115]. For the mesh adaptivity, the solid concentration field and the velocity

Case	$\Delta x_{min}[m]$	$\epsilon_s[-]$	$\epsilon_u[m/s]$	n_{adapt}
A1	$2e-4$	0.005	0.001	123805
A2	$4e-4$	0.005	0.001	76532
A3	$8e-4$	0.005	0.001	53964
B1	$1e-4$	0.01	0.002	195858
B2	$2e-4$	0.01	0.002	102338
B3	$3e-4$	0.01	0.002	65989
B4	$4e-4$	0.01	0.002	64266
B5	$8e-4$	0.01	0.002	41715
B6	$12e-4$	0.01	0.002	29549
C	$4e-4$	0.02	0.004	57322
D1	$4e-4$	–	0.002	11499
D2	$4e-4$	–	0.001	39420
D3	$4e-4$	–	0.0005	67879
E1	$4e-4$	0.1	0.001	64621
E2	$4e-4$	0.05	0.001	72117

Table 4-2: Turek CFD1 mesh convergence

field are considered, where a certain percentage of the expected range of the corresponding field is chosen for the interpolation error bound of the algorithm ϵ_u and ϵ_s , respectively. The investigated constellations as well as the resulting number of elements n_{adapt} in the adapted mesh are summarised in table 4-2.

**Figure 4-1:** Mesh convergence Turek & Hron CFD1 case, pay attention to the different scales of the relative error in lift and drag

The benchmark data offers lift and drag values for the three test cases. Since the internal force evaluation feature in Fluidity is currently not working in combination with the Immersed Boundary Method used within this project, the force integration has to be done in a post-processing script. Since external evaluation of these properties entails interpolation of pressure and velocity gradient information, the force calculation is expected to be rather inaccurate. Nevertheless, keeping this in mind, currently, no other option is available. However, it should be noted that the calculation of the pressure forces is expected to be more accurate than the

evaluation of the skin friction because here only a pressure interpolation and a subsequent integration has to be carried out. For the evaluation of the skin friction, the velocity field first has to be differentiated, then interpolated and finally integrated which leads to relatively noisy skin friction data. Thus, the drag calculation which is dominated by the contribution from pressure drag is expected to be more accurate than the lift calculation. The results of the mesh convergence study in terms of the relative error as compared to the benchmark data from Turek and Hron [66] are shown in figure 4-1.

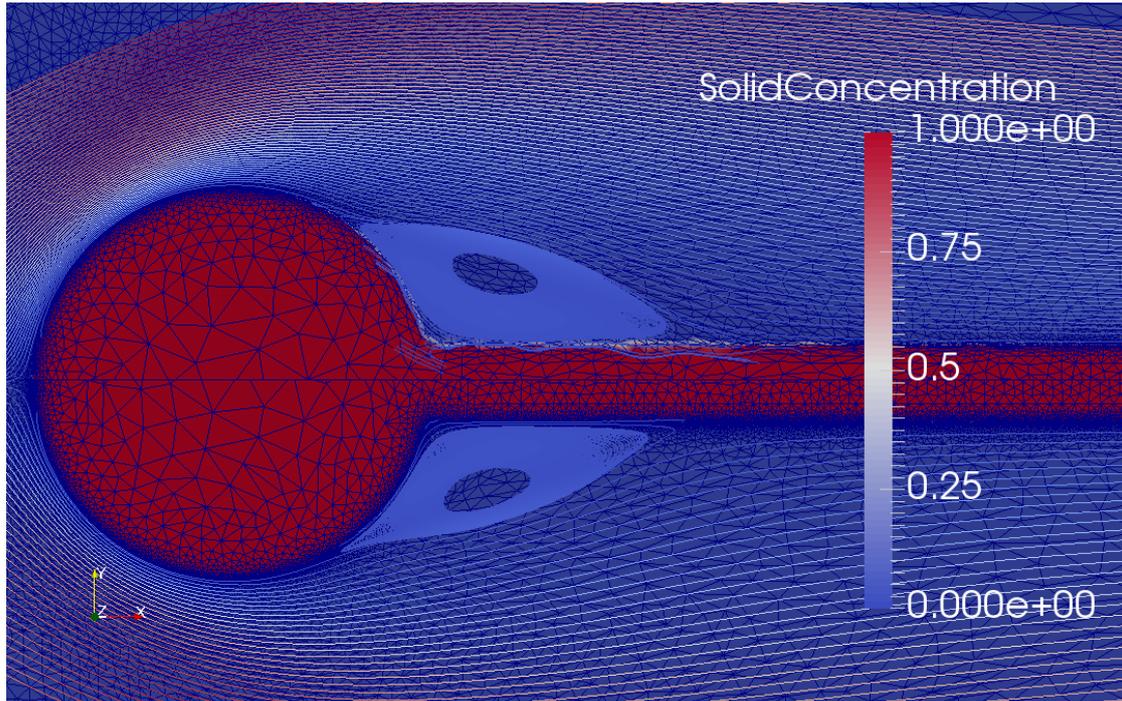


Figure 4-2: Flow field detail Turek CFD1 case, on the top the D4 and on the bottom the A2 case are shown

Because of issues related to the mesh adaptivity and the force evaluation as explained in the subsequent section, the simulations were only run with mesh adaptivity up to two seconds by which time the number of mesh elements remained approximately constant with a CFL number of 0.5. Afterwards, the simulations were restarted with a fixed mesh as obtained from the adaptivity algorithm and rerun with a CFL number of 5 until convergence of the flow field to a relative tolerance of $\epsilon = 1e - 6$ was achieved.

The results of the mesh convergence study in figure 4-1 do not yield a clear conclusion towards the mesh independence of the solution in part possibly due to the inaccuracy of the force evaluation. Nevertheless, some trends can be deduced with at least moderate certainty. Namely, first reducing the minimum edge length of the mesh Δx_{min} beyond $4e - 4m$ does not yield a clear improvement of the accuracy of the solution for the given force calculation method. Choosing the velocity interpolation error bound of the order of 0.5 % of the expected mean velocity seems to yield the best results in terms of mesh size versus accuracy of the solution. Third, in this particular case, using the solid concentration field in addition to the velocity field to adapt the mesh mainly seems to increase the size of the mesh without improving (or even decreasing) the accuracy of the simulation results.

However, a closer look at the converged flow field for the cases A2 and D3 in figure 4-2, so both cases with a minimum edge length of $\Delta x_{min} = 4e-4m$ and a small velocity interpolation error bound, but without using the solid concentration field for mesh adaptivity for the second one, reveals that this might be simply due to the inaccuracy of the force calculation method. Because as can be seen in the figure, the outline of the body is less well resolved in the case where the solid concentration field is not used for the mesh adaptivity algorithm even though the lift and drag are predicted more accurately. Thus, at least a relatively large interpolation error bound on the solid concentration field should be used given the inaccuracy of the force calculation method. However, setting the interpolation error bound too small as was seen for the cases A and B results in an unnecessarily large mesh size.

Summarising, the adaptivity settings as chosen for the E1 and E2 test case seem to yield the best results for the mesh convergence of the CFD1 case. The mesh size of roughly 70,000 elements also corresponds with what was used for the benchmark results by Turek and Hron to obtain mesh-independent results. Although it should be noted, that direct comparison is not advised, since the reference case uses a structured, body-conforming mesh. Moreover, for the reference benchmark data convergence of the results is much faster, namely with only about 15,000 elements results with approximately 1 % error can already be obtained.

Case	$\Delta x_{min}[m]$	$\epsilon_s[-]$	$\epsilon_u[m/s]$	n_{adapt}	Norm L_m
A1	$4e-4$	0.05	0.005	9503	2
A2	$4e-4$	0.05	0.001	42948	2
A3	$4e-4$	0.05	0.005	81130	∞
A4	$4e-4$	0.05	0.001	190205	∞
B1	$4e-4$	0.005	0.005	27220	2
B2	$4e-4$	0.005	0.001	54860	2
B3	$4e-4$	0.005	0.005	83354	∞
B4	$4e-4$	0.005	0.001	192283	∞
C1	$1e-4$	0.05	0.005	9596	2
C2	$1e-4$	0.05	0.001	43368	2
C3	$1e-4$	0.05	0.005	146779	∞
C4	$1e-4$	0.05	0.001	260445	∞

Table 4-3: Turek CFD2 mesh convergence

Because, extending the same methodology to the CFD2 case yielded unsatisfactory results with very noisy body force computations, the force evaluation method described above was implemented internally in Fluidity. While this does not correspond with the original approach described in chapter 6, it is more accurate than doing the evaluation externally. In fact, what was internally implemented is a consistent interpolation of the velocity gradients and pressure evaluated on the fluid mesh to the interface nodes on the structure mesh. Subsequently, these values were integrated along the exact interface location as given by the structure mesh.

Thus, another mesh convergence study was done for the CFD2 case for the settings shown in table 4-3. The results are shown in figure 4-3. Conversely, to the CFD1 case here a clear convergence of the integrated force values with mesh refinement is not distinguishable. The cause of this is unclear. However, it was decided to not spend more time on this.

In fact, the unsteady CFD3 case was briefly investigated as well based on the mesh convergence

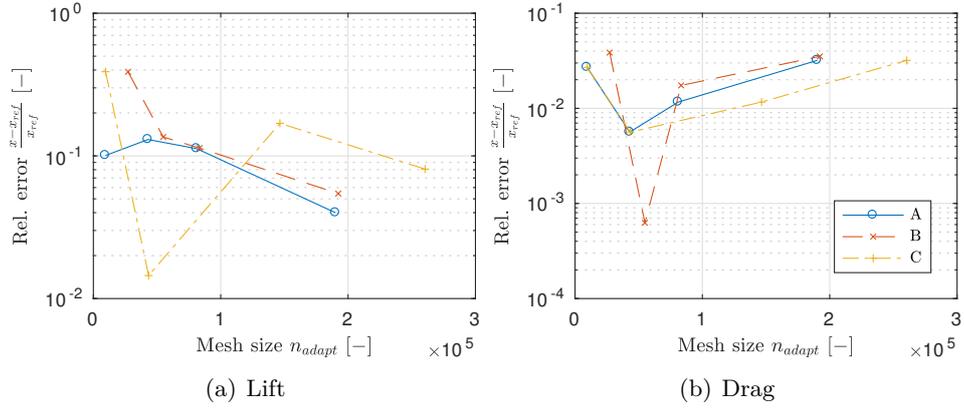


Figure 4-3: Mesh convergence CFD2 case

results for the CFD1 & CFD2 case. However, the solver was not able to capture the vortex shedding independent of the mesh resolution and the chosen time step. However, when increasing the Reynolds of the setup to $Re = 300$ a vortex shedding was observed. Thus, it can be concluded that this is due to the accuracy of the IB approach or the order of the elements used to discretise the fluid.

Mesh adaptivity and force evaluation

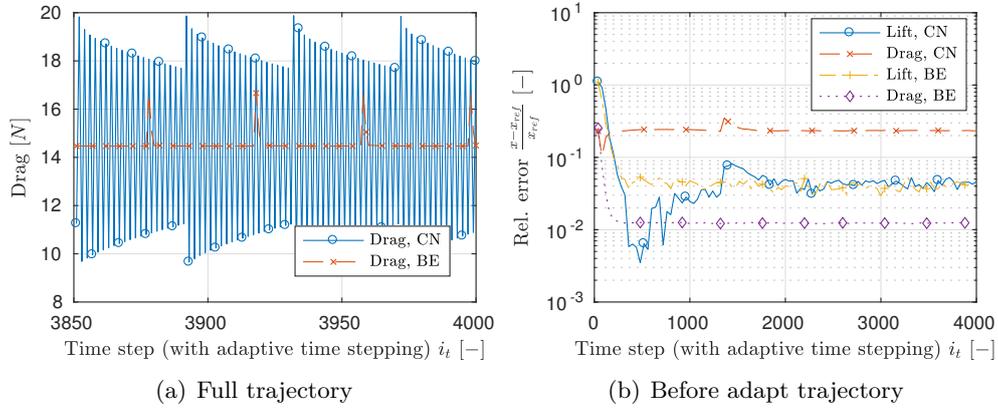


Figure 4-4: Convergence of lift and drag with mesh adaptivity for case A2 with either full trajectory for every iteration or only trajectory of values directly before the mesh is adapted run with adaptive time stepping with $CFL = 5$

Because the simulations using mesh adaptivity would not converge neither in terms of integrated force coefficients nor in terms of convergence of the flow field down to the desired tolerance when using the standard Crank-Nicolson time integration scheme, a more detailed analysis of the convergence of the transient simulation towards a steady state (for the CFD1 and CFD2 cases) was carried out. For transients with a CFL number of 0.5 as well as 5 as shown in figures 4-4 and 4-5, respectively, significant oscillations in the integrated body forces are observed for both time integration schemes. The source of this oscillations is numerical

excitation through the mesh adaptivity. While the first order accurate Backward Euler finite difference discretisation scheme damps the oscillations out within a few iterations after the mesh adaptivity algorithm is applied, the second order accurate Crank Nicolson scheme takes much longer to dampen the oscillations. In fact, the flow field will not converge because of the noise introduced through the mesh adaptivity. A truncation analysis of both schemes shows that while the Crank Nicolson method is more accurate, the leading error term is amplified by the third derivative of the variable instead of the second one as for the Backward Euler term. Thus, it introduces less numerical damping as compared to the other method. Going forward when mesh adaptivity is used, the less accurate Backward Euler time stepping scheme will be used. The numerical oscillations introduced through mesh adaptivity are better damped as compared to the Crank-Nicolson scheme such that the dynamics of the system should still be accurate.

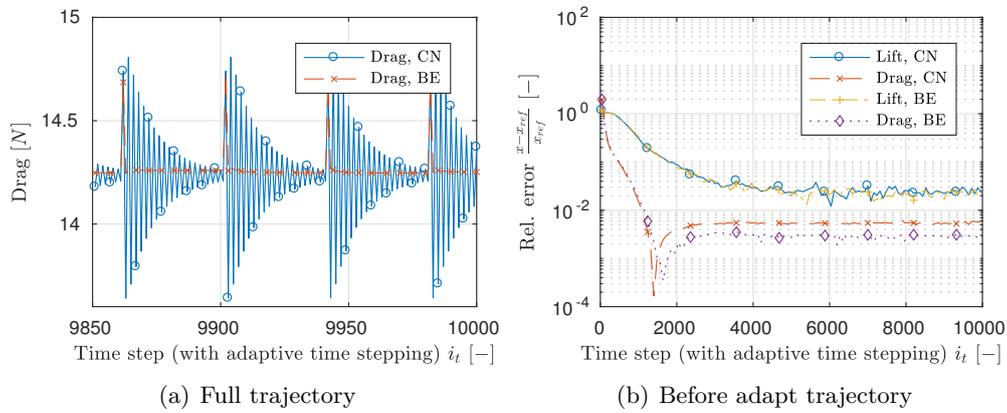


Figure 4-5: Convergence of lift and drag with mesh adaptivity for case A2 with either full trajectory for every iteration or only trajectory of values directly before the mesh is adapted run with adaptive time stepping with $CFL = 0.5$ and either the Crank-Nicolson (CN) or Backward Euler (BE) temporal discretisation

Conclusions Some inconsistencies in the results from the Immersed Boundary methodology were discovered through its application to the rigid Turek & Hron setup at Reynolds numbers set in the laminar flow regime. First, the convergence of the integrated force values with mesh refinement was not given. This could be due to the inaccurate force evaluation method, the inaccuracy of the IB method or also the lower order of the used Finite Elements. Second, the mesh adaptivity was shown to indeed influence the trajectory of the flow field variables. As a consequence, its application was only useful if used sparingly. One possible cause here is that the mesh was adapted to the solid concentration field on the fluid mesh. Since this property changes as the mesh adapts, it is not well suited for this purpose. Moreover, some preliminary results suggested that instead of adapting to the gradients in the solid concentration field, a more suitable solution would be to adapt to an exponentially decaying field that scales with the distance to the body as well as the channel walls. In addition to this, the mesh should still be adapted to the gradients in the velocity field but with a higher threshold such that vortex shedding in the wake of the body can still be resolved. Lastly, once deformations were prescribed to the flag part of the setup, the discontinuity of the pressure field between time steps would lead to divergence of the simulation because the gradient operator was not well

defined. Given, these grave limitations it was decided that within the context of this project, it is better advised to continue working with a more mature solver framework.

4-3 Model 3: Conventional Computational Fluid Dynamics Approach

As already mentioned in the previous paragraph, due to grave limitations in the Immersed Boundary solver methodology, a more mature solver framework was considered. In this case, going forwards the fluid solver foam-extend-3.2, a side branch of OpenFOAM, will be used [129]. Blom coupled this version of foam-extend to the coupling library preCICE [130]. This was ideally suited for this project because preCICE possesses Python ports. Both the fluid solver as well as the coupling library are open-source. The coupled version of foam-extend as implemented by Blom is referred to as FOAM-FSI [131]; the implementation details are documented in the dissertations from Blom (2017) [132] and Gillebart (2016) [133].

Subsection 4-3-1 will outline the implementation specific details of the fluid solver methodology also with respect to a moving interface. Then, subsection 4-3-2 documents a validation of the solver on the Turek & Hron FSI benchmark results for the rigid setup.

4-3-1 Implementation details

Discretisation foam-extend uses a Finite Volume approach for the spatial discretisation of the incompressible Navier-Stokes Equations, where various options for the discretisation of the spatial derivative are available. The time integration is done using the method of lines, where again some discretisation schemes are available. For this project, fairly standard discretisation schemes were chosen, and the settings are listed in appendix C-1.

Governing equations on a moving grid Because the boundaries of the fluid domain move to accommodate the current structure interface once the structure starts deforming the governing fluid dynamics equations 2-11 need to be rewritten in a moving framework. First, rewriting the equations in a conservative form and applying the basic assumptions related to the Finite Volume approach yields the spatially discretised Navier-Stokes equations below

$$\sum_f \mathbf{u}_f \mathbf{n}_f S_f = 0 \quad (4-15a)$$

$$V \frac{\partial \mathbf{u}}{\partial t} + \sum_f \mathbf{n}_f \cdot \mathbf{u}_f \mathbf{u}_f + \nu \sum_f (\nabla \mathbf{u})_f \mathbf{n}_f S_f = - \sum_f p_f \mathbf{n}_f S_f \quad (4-15b)$$

for a cell volume V with face areas S_f , cell centre velocity \mathbf{u} and interpolated cell face velocity \mathbf{u}_f and pressure p_f . For a moving grid, the momentum equation can now be rewritten in an Arbitrary Lagrangian-Eulerian framework as

$$V \frac{\partial \mathbf{u}}{\partial t} + \sum_f \mathbf{n}_f \cdot (\mathbf{u}_f - \mathbf{u}_{mesh}) \mathbf{u}_f + \nu \sum_f (\nabla \mathbf{u})_f \mathbf{n}_f S_f = - \sum_f p_f \mathbf{n}_f S_f \quad (4-16a)$$

where \mathbf{u}_{mesh} is the speed of the mesh. In order to ensure time consistency of the discretisation on a moving mesh the Discrete Geometric Conservation Law is used for the determination of the mesh velocity. The conservation law ensures that for a uniform flow the time integration is exact [?]. If a Finite Volume discretisation is used the semi-discretised conservation law can be written as

$$\frac{\partial V}{\partial t} = \sum_f (\mathbf{u}_{mesh} \cdot \mathbf{n}_f S_f) \quad (4-17)$$

such that the volume swept by the moving cell boundaries is equal to the change in the volume of the cell. The mesh velocity to be used in the ALE formulation of the NS equations can be obtained from this equation consistent with the temporal discretisation of the terms on the left-hand side.

Mesh Because in foam-extend the Fluid-Structure Interaction problem will be solved using an Arbitrary Lagrangian-Eulerian approach, mesh adaptivity is not needed as compared to the Immersed Boundary approach in Fluidity where the mesh needs to be adapted to follow the outline of the deforming body. Thus here a conventional unstructured mesh approach will be used. However, because Finite Volume schemes are more sensitive to skewed cells than Finite Element solvers, foam-extend has to employ non-orthogonal corrections within the solution procedure of the solver to correct for non-orthogonality of the cells. In fact, since only simple geometries are considered within the context of this project, de facto structured meshes are used that are however written in an unstructured mesh format.

Interface interpolation and mesh deformation Radial Basis Functions (RBFs) are used for mesh deformation as well as interface interpolation. The interface interpolation method is detailed in section 6-2-4. Similarly, for the mesh deformation the RBF is calibrated with the displacements at the interface, and then interpolation is used to transfer the displacements away from the interface into the rest of the mesh. Here, Thin Plate Spline functions, because they generally yield a good mesh quality for the deformed mesh [?]. For a detailed description of the methodology, see Gillebart et al. (2016) [134].

Solution procedure Similar to Fluidity, in foam-extend a number of projection based algorithms are implemented that solve the coupled pressure-momentum equations. The most basic form of the Algorithm is the SIMPLE algorithm which involves one predictor and one corrector step, where outer iterations (equal to the nonlinear iterations in Fluidity) can be specified to ensure that the continuity as well as the momentum equations are fulfilled. The predictor step solves the momentum equations for an uncorrected velocity that does in general not fulfil the continuity equation based on the pressure information from previous time steps. Subsequently, the corrector step solves a Poisson equation for the pressure based on

the uncorrected velocity and thus obtains a prediction of the pressure. Subsequently, this pressure correction is used to adjust the uncorrected velocities from the predictor loop. However, this corrected velocity does not fulfil the momentum equation anymore such that this process has to be repeated until the continuity as well as the momentum equations are fulfilled. For steady-state cases, different implementations of the basic SIMPLE, as well as the SIMPLEC algorithm, are implemented, because inherently the basic SIMPLE algorithm is not consistent and neglects some terms in the derivation of the momentum equation and thus requires underrelaxation for convergence. Due to the inclusion of an estimate of those terms the SIMPLEC algorithm generally converges faster. [135]

For transient cases, the PISO algorithm is used, which now also considers time derivative terms and uses one predictor and a specified number of pressure corrector steps. Now, differentiating between inner and outer correction loops for one time step, inner correction loops refer to the number of pressure corrector steps in the algorithm and outer iterations refer to the number of correction steps where the nonlinear advection terms (and possibly also source terms) are updated. For small time steps which fulfil the CFL condition, outer correction loops are usually not necessary because it is assumed the effect of the nonlinear terms in the momentum equation is less significant than the pressure-velocity coupling [136]. However, for large time steps as present in steady-state solvers, the opposite holds true.

Lastly, in foam-extend also the PIMPLE (or iterated PISO) algorithm is implemented which blends the SIMPLE and the PIMPLE algorithm together and thus allows transient simulations at CFL numbers larger than one. This is accomplished by applying a specified number of inner pressure correction loops as is custom for the PISO algorithm and building an outer correction loop around the PISO loop as necessary for the SIMPLE algorithm for convergence of the nonlinear terms [136, 137]. While this implicit coupling makes the transient simulations stable for larger time steps, it may not necessarily be physical depending on the importance of the neglected time scales on the global behaviour of the flow and the order of the time integration scheme.

As already mentioned in the literature review section 2-2-1, for FSI problem using an ALE approach the Navier-Stokes Equations need to be solved on a moving grid. Thus, additional complexities occur due to having to solve the equations on a moving mesh. For the specific implementation used in this project see Blom (2017) [132] with particular reference to the time-consistent implementation on moving meshes.

Turbulence model The conclusions that were drawn for Fluidity with reference to the most suitable turbulence models also hold for foam-extend. Because foam-extend is only a side branch of OpenFOAM, transition models are not available either.

4-3-2 Validation

Again, to gain some experience with the solver and its meshing requirements for the FSI benchmark case, here also first a separate validation of the fluid solver without the coupling to the structural solver is carried out based on the reference results from Turek and Hron [66]. The case setup was already documented in section 4-2-2 where the validation of Fluidity is carried out and thus will not be repeated here.

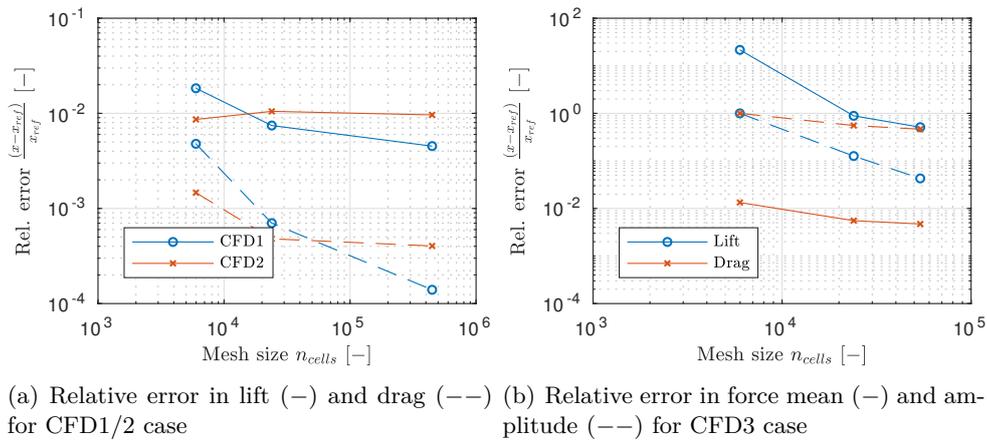


Figure 4-6: Mesh convergence Turek & Hron CFD benchmark cases

For each one of the three CFD cases, three simulations with varying mesh density were carried out. The coarsest mesh consists of roughly 6000 elements, whereas the regular and the fine mesh have close to 24,000 and 54,000 elements, respectively. The deformed fluid mesh is shown in figure 7-4. The mesh convergence results for the steady as well as the unsteady benchmark case are shown in figure 4-6. A clear convergence trend towards the benchmark results can be distinguished for all the cases. Similar to the results from Fluidity, here also drag values are more accurately predicted. However, it should be noted that especially for the unsteady cases the deviation from the reference results is not insignificant and a much finer mesh would be needed to capture the vortex shedding accurately. In fact, the coarsest investigated mesh configuration fails to capture the onset of the vortex shedding for this Reynolds number. In figure 4-7 an excerpt of the fully periodic lift and drag prediction for the CFD3 case is shown. It can be seen, that while the mean value of the lift and the amplitude of the drag oscillation are not well predicted, the qualitative behaviour agrees well with the reference results.

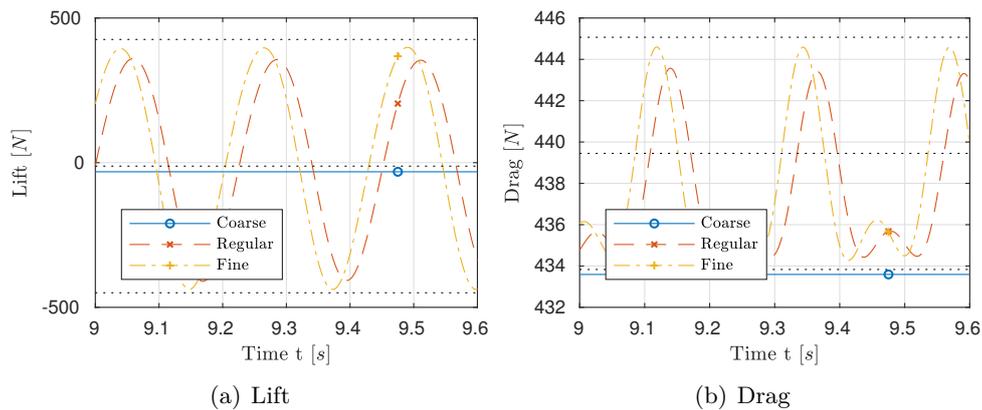


Figure 4-7: Time history of the unsteady vortex shedding for the CFD3 case alongside with the reference results in black (·)

Concluding, for the separate validation of the fluid solver with the Turek & Hron benchmark

case, the setup in foam-extend shows good convergence towards the reference results with some inaccuracy for the unsteady results when using a mesh with up to roughly 54000 elements. This is however not expected to be that important since for the FSI benchmark case the deformation of the structure will dominate over the vortex induction for the non-moving case. Thus the influence of the inaccuracy of the solver in the prediction of the onset of the vortex shedding is less significant with regards to the overall accuracy of the solver.

Coupling algorithm

As outlined in the literature review in chapter 2, two different coupling methods have been deemed useful for this project due to their efficiency gains over a standard serial coupling scheme and their negligible implementation effort. The first scheme is the standard Aitken under-relaxation scheme [58] and the second one is the IQN-ILS scheme, more recently developed by Degroote et al. [60].

A brief description of both methods is given in the following in section 5-1. Then the efficiency of the methods is compared on a simple test case through coupling of the structural solver framework with the vortex particle method in sub-section 5-2.

5-1 Methodology

The standard ALE domain composition is a Dirichlet-Neumann decomposition of the fluid and structure domain, where the structure imposes certain domain boundaries and velocities on the fluid and the fluid, in turn, imposes forces on the structure at the interface. For the immersed boundary method used in this project, the same domain decomposition is used. The fluid and structure solver can be represented as operators \mathcal{F} and \mathcal{S} , respectively. The operators act on the interface displacements \mathbf{d} and the interface forces \mathbf{f} .

$$\mathbf{d} = \mathcal{S}(\mathbf{f}) \tag{5-1a}$$

$$\mathbf{f} = \mathcal{F}(\mathbf{d}) \tag{5-1b}$$

As already mentioned two coupling schemes were implemented in the Python solver framework, the basics of the two algorithms have already been outlined in the literature review section 2-4-2 and are further elaborated on in the following. Both algorithms are also implemented in preCICE, and thus the Python implementations were only used for coupling with the VPM method.

Method 1: Serial scheme with dynamic relaxation

The relaxation parameter is calculated according to the Aitken under-relaxation method detailed in Kuttler and Wall [58].

Method 2: Newton scheme with residual based Jacobian approximation

Degroote et al. [60] developed an inverse Jacobian approximation based on previous sub-iteration results, the method is referred to as IQN-ILS and briefly outlined below.

First, the residual of an iteration step is defined according to

$$\mathbf{R}(\mathbf{d}_s) = \hat{\mathcal{F}}(\mathbf{d}_s) = \mathcal{S} \circ \mathcal{F}(\mathbf{d}) - \mathbf{d} = \tilde{\mathbf{d}} - \mathbf{d} \quad (5-2)$$

From this, two matrices are formulated at every sub-iteration step where $k \leq 2$, the matrices have to be updated at every sub-iteration step

$$V^k = [\Delta \mathbf{R}^{k-1} \dots \Delta \mathbf{R}^0] \quad \text{with } \Delta \mathbf{R}^i = \mathbf{R}^i - \mathbf{R}^k \quad (5-3a)$$

$$W^k = [\Delta \mathbf{d}^{k-1} \dots \Delta \mathbf{d}^0] \quad \text{with } \Delta \mathbf{d}^i = \mathbf{d}^i - \mathbf{d}^k \quad (5-3b)$$

Now the desired solver residual change $\Delta \mathbf{R} = \mathbf{0} - \mathbf{R}^k$ can be obtained as a linear combination of the residuals from the previous sub-iterations. The coefficients α obtained from the solution of this Least Square problem can then be used to gauge an estimate of the change in the structure displacement $\tilde{\mathbf{d}}$

$$\Delta \mathbf{R} = \mathbf{0} - \mathbf{R}^k = V^k \alpha^k \rightarrow \alpha_k = \dots \quad (5-4a)$$

$$\Delta \tilde{\mathbf{d}} = W^k \alpha^k \quad (5-4b)$$

From this now the change in displacement for the next sub-iteration can be determined. Thus the inverse of the Jacobian was implicitly calculated

$$\Delta \mathbf{d}^k = -\mathcal{J}_{\hat{\mathcal{F}}}^{(-1)} \mathbf{R}^k = W^k \alpha^k - \Delta \mathbf{R} \quad (5-5a)$$

$$\mathbf{d}^{k+1} = \mathbf{d}^k + \Delta \mathbf{d}^k \quad (5-5b)$$

When a desired residual change ΔR is orthogonal to one or multiple rows of V^k , then the corresponding coefficients are zero $\alpha_l = 0$. When the displacement input for the next sub-iteration is calculated accordingly, only the residual term is left for those basis vectors, and thus then a standard Gauss-Seidel iteration is performed in that specific direction of the solution room.

5-2 Validation

The validation of the coupling algorithm is done using the quasi-steady structural solver and the unsteady Vortex Particle Method. By choosing the grid points of the structural and the vortex particle mesh coinciding error introduction through interpolation is avoided. The test case is a membrane rigidly clamped at leading and trailing edge subjected to inflow at different angles of attack. Since the unsteady Vortex Particle Method is two-dimensional and the structural solver is three-dimensional, the loads from the aerodynamic model need to be extrapolated to create a quasi-two-dimensional load on the three-dimensional structure. The setup is chosen in close correspondence with the experimental setup of Greenhalgh et al. [84, 3]. The inflow conditions are chosen such that they correspond to a Reynolds number of $1.3e6$ at standard air conditions. The wing properties and discretisation are chosen according to table 5-1 with boundary conditions that support that quasi-two-dimensional load imposition, namely no spanwise displacements and no rotation around the axes in the spanwise plane on the side nodes of the clamped membrane. See figure 2-4 for illustration of the setup in the literature review section.

E_S	$1.25e9N/m^2$
ν_S	0.3
h_S	$1e-3m$
(N_x, N_y)	(40, 1)
(L_x, L_y)	(1m, 0.025m)

Table 5-1: Membrane properties and discretisation

The wind speed is kept constant throughout the simulation time; however, the angle of attack undergoes periodic stepwise changes as shown in figure 5-1. A time step of $\Delta t = 0.025s$ is used for the simulations, where each discontinuous change in angle of attack is followed by two sub-iterations with a rigid structure with a time step of $\Delta t_{sub} = 0.001s$. Otherwise, the simulation becomes unstable due to the time discretisation of the circulation in the Vortex Particle Method. The resulting aerodynamic forces are shown in figure 5-1(a).

Then, figure 5-1(b) shows the necessary computation time for both the Aitken under-relaxation and the IQN-ILS coupling scheme, both run with the same computational architecture. For this strongly coupled problem with up to 40 coupling iterations, the IQN-ILS algorithm outperforms the under-relaxation scheme in terms of efficiency. In fact, for the simulated time where both methods were used, the overall computation time was reduced by a factor of six when using the IQN-ILS method as compared to the simple under-relaxation algorithm. Moreover, the under-relaxation overall required twice the amount of coupling iterations, but with almost 30 times more evaluations of the fluid and structure solver.

Lastly, figure 5-2 shows the almost steady state lift coefficient versus the inflow angle for the time domain simulation results shown in figure 5-1. Despite the dynamic and damping effects being neglected in the structural model, the lift curve still shows a hysteresis effect as described by Greenhalgh et al. [3]. Due to inertial effects in the flow field, the membrane remains deflected to the same side as previous angles of attack past the zero degree inflow condition. Shortly after zero degrees, it begins to show an inflexion point and then at around two degrees it fully snaps through to the other side. This effect is more pronounced for more

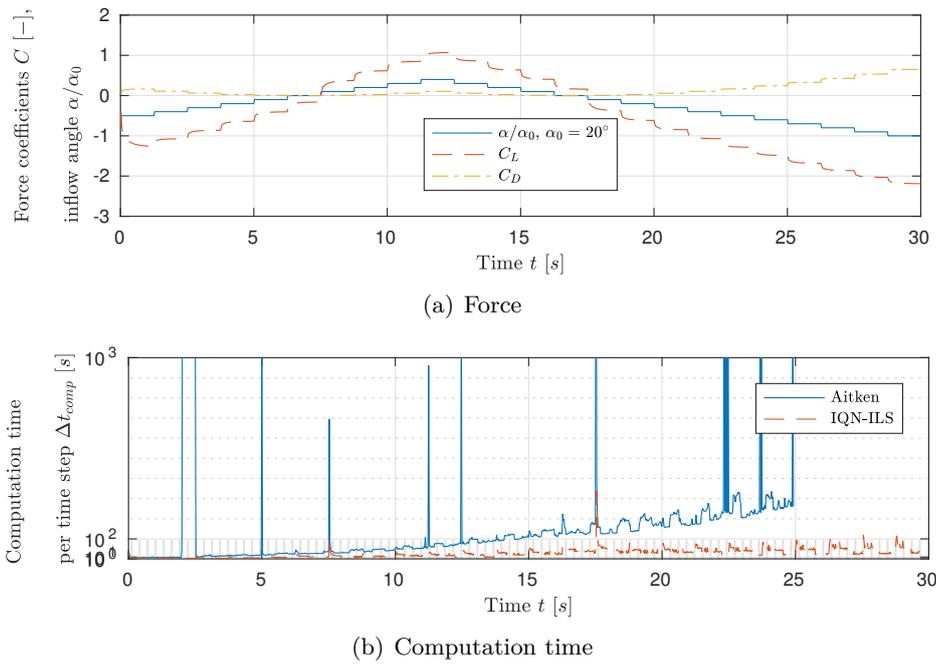


Figure 5-1: Coupling method comparison through time domain simulation with unsteady fluid solver and quasi-steady structural solver

flexible materials or for a larger excess length of the membrane as compared to the chord length [3].

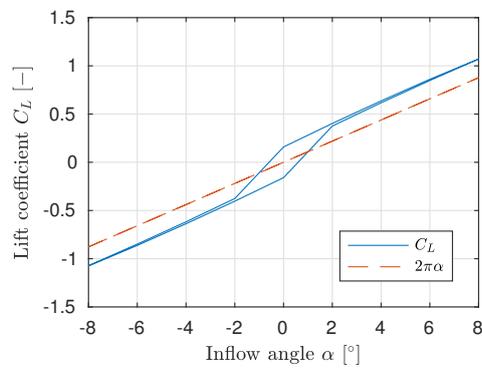


Figure 5-2: Lift hysteresis membrane wing

Interface interpolation

For the interpolation of the structure node displacements on the structure mesh and the forces on the fluid mesh, a simple methodology had to be implemented. For the two different solvers and FSI approaches, slight adjustments had to be made. This section outlines the method for the Immersed Boundary solver in detail 6-1. Then section 6-2 lists the adjustment that had to be made to apply the method to the Arbitrary Lagrangian-Eulerian approach.

6-1 Immersed Boundary technique

For the Immersed Boundary method implemented in Fluidity, the communication between the structural and the fluid model requires three steps. First, the extrusion of the zero-thickness structure mesh to a fully three-dimensional mesh. Second, the integration of the interface forces on this mesh and the subsequent projection of the force from the extruded mesh to the structure mesh. Lastly, in reverse, the velocity of the structure nodes needs to be extrapolated to the fluid mesh as well. The following subsection outlines the methodology in more detail.

6-1-1 Mesh extrusion

The two-dimensional surface mesh of the shell as modelled in the structure solver needs to be extruded to a three-dimensional body because the fluid solver is unable to work with a zero-thickness surface. The extrusion is done normal to the surface mesh with the actual zero-thickness interface location of the shell located in the middle of the extruded mesh. If possible, the thickness is chosen coinciding with the actual physical thickness of the shell elements; however, a lower limit is imposed on the thickness through the Immersed Boundary solver implementation in Fluidity. Thus for very thin structures, the thickness does not correspond with its physical dimensions.

Due to restrictions in computational time, at least initially a two-dimensional setup is investigated. Thus given that the structural solver is inherently 3D, the loads from the 2D

fluid solver need to be imposed on the structure such that the structural dynamics become essentially two-dimensional. Thus in the following, the mesh extrusion is done for the two as well as the three-dimensional case.

Due to the simple geometries investigated within the scope of this project, a simple geometrical algorithm is used to create the surface normals. Depending on the dimension of the fluid simulation space, this is done as outlined below.

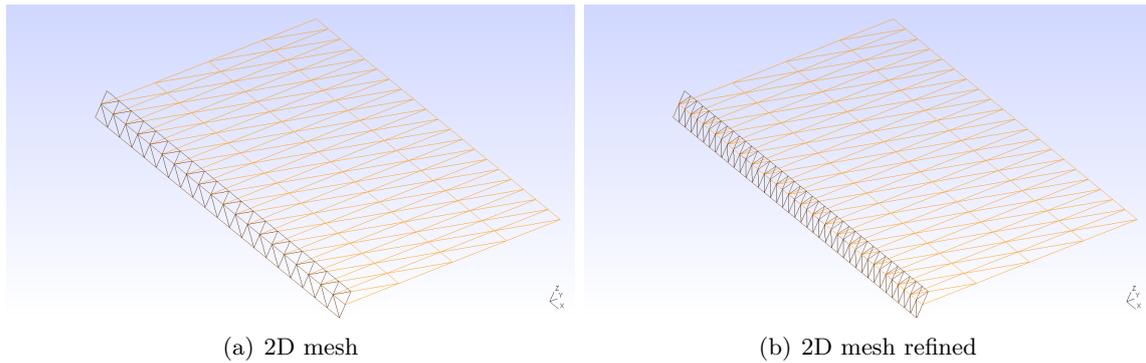


Figure 6-1: Example of a 2D extrusion mesh (black) with the underlying structure mesh (orange) in the background

2D fluid domain For each node the normals based on the two intersecting lines and the orientation of the two-dimensional simulation space are calculated, and then the mean of both normals is taken to be the normal of the corresponding node. For boundary nodes no averaging takes place. Now displacing from the structural mesh outwards along the normals an extrusion mesh is created. Because Fluidity is not able to deal with quadrilateral elements, they are split up into triangles as pictured in figure 6-1. For increased accuracy of the interpolation scheme also a refined version of the extrusion mesh is created, where between two structure nodes an additional interpolation node is introduced. This will be explained further in the next subsection.

3D fluid domain For each node all intersecting elements are identified. Then for each element, a normal is calculated based on the intersecting edge lines of the element. Subsequently, for each node the normal is obtained by averaging the normals from all the intersecting elements. The extrusion along the element normals results in a mesh consisting of triangular prisms. Due to Fluidity's inability to deal with this type of elements, the triangular prisms are split into three tetrahedrons on both sides of the original shell mesh as pictured in figure 6-2. The splitting needs to be done such that the edge lines between the elements overlap, which is straightforward for an open mesh. However, if the mesh were to for example form a cylinder, special attention should be paid to this.

6-1-2 Force projection

Once the volume forces have been projected onto the extrusion mesh, a reduction of the properties on the extrusion mesh to one dimension smaller structure mesh has to be carried

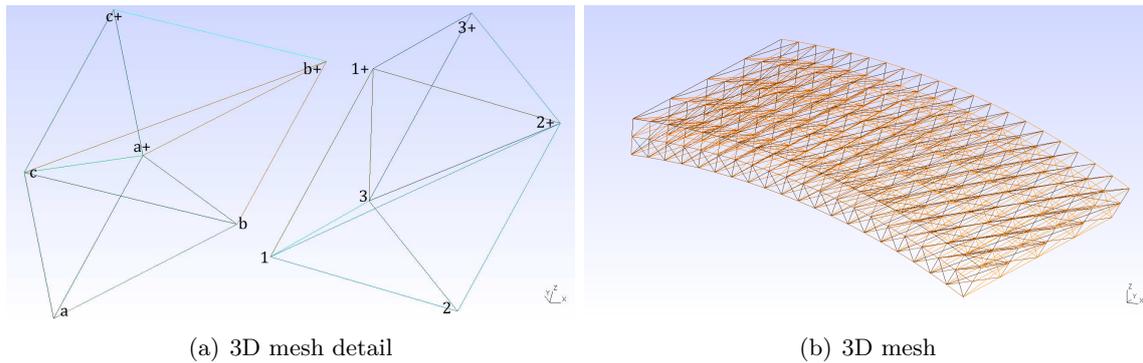


Figure 6-2: Example of a 3D extrusion mesh, the triangular prisms are split into tetrahedrons according to $(1, 2, 3, 2+), (1, 3, 1+, 2+), (3, 1+, 2+, 3+), (a, b, c, a+), (b, c, a+, b+), (c, a+, b+, c+)$

out. This can either be done exactly through the integration of the shape functions on the extrusion mesh or less accurately by averaging the values along wall normal directions. The second methodology is somewhat justified if the extrusion mesh thickness is small and a sufficient number of nodes are used in the tangential direction of the structure such that gradients are sufficiently resolved. However, such an interpolation scheme is not conservative - albeit consistent - such that momentum is either lost or created through the interpolation process which can accumulate over the duration of the simulation. A direct comparison would have to be made to quantify the effect of the less accurate interpolation scheme.

When the shape functions as defined on the extrusion mesh are used to integrate the volume forces, either exact or numerical integration can be used. For non-uniform elements, it is more practical to use numerical integration. However, in case numerical integration is used, the accuracy of the integration scheme needs to be at least as high as the accuracy of the Galerkin projection (used for the projection of the fluid properties to the extrusion mesh) and the discretisation errors expected in the structure as well as the fluid model.

For the **two-dimensional case** the numerical integration of the shape functions is easily implemented given that the extrusion mesh is refined as compared to the underlying structure mesh, such that integration over four elements will yield the force on one node as pictured in figure 6-1(b).

First, a standard isoparametric transformation - see figure 6-3 - for linear element formulations is introduced to switch between the coordinates of a quadratic triangle and an irregular triangle with vertices (ξ_i, η_i) and (x_i, y_i) , respectively, according to

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \frac{1}{(x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)} \begin{bmatrix} y_3 - y_1 & -(x_3 - x_1) \\ -(y_2 - y_1) & x_2 - x_1 \end{bmatrix} \begin{bmatrix} x - x_1 \\ y - y_1 \end{bmatrix} \quad (6-1)$$

The shape functions on the quadratic triangle are defined as

$$\phi_1 = 1 - \xi - \eta \quad (6-2a)$$

$$\phi_2 = \xi \quad (6-2b)$$

$$\phi_3 = \eta \quad (6-2c)$$

where the volume force distribution with node values a_i on the element in isoparametric coordinates is given by

$$f(\xi, \eta) = a_1\psi_1 + a_2\psi_2 + a_3\psi_3 \quad (6-3)$$

such that the integration of the force over the element can be written as

$$F^e = \int_A f(x, y) dx dy = \int_A f(\xi, \eta) |\det J| d\xi d\eta = \frac{|\det J|}{3} (a_1 + a_2 + a_3) \quad (6-4)$$

with J the Jacobian of the isoparametric transformation. The Jacobian is constant throughout the entire element because the shape functions are linear. This also means that the force integration can be done exactly as written out above. The force on a specific node is then given by summing the element forces over the four elements having an edge coinciding with this specific node.

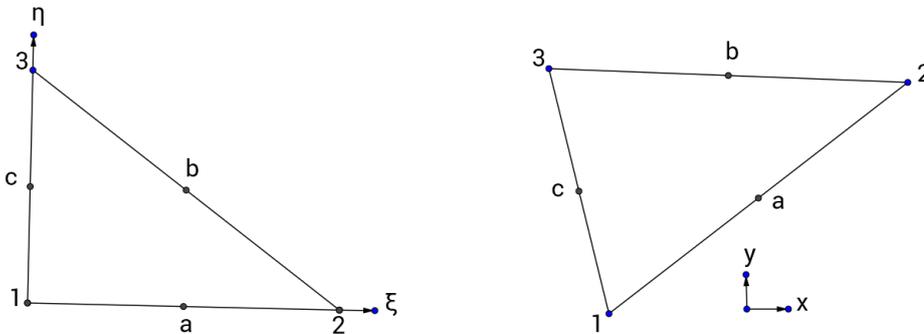


Figure 6-3: Isoparametric transformation triangle coordinates

For the **three-dimensional** case, in principle, the same integration procedure can be performed. However, this would require a more refined mesh with an additional interpolation node between the regular nodes of the structure. The mesh generation procedure would, however, become more complicated. Thus for this project, only the averaging approach is implemented. The integral of the force in one tetrahedron F^e on the positive side of the structure mesh is approximated as

$$F^e = \frac{V_E}{2} (f_a + f_{a+}) \quad (6-5)$$

where the sum of the force over all the tetrahedrons assigned to the appropriate node - in this case node a - will yield the force distribution along the structure surface. This is identical to the interpolation scheme employed by Viré et al. [127] for a very similar Immersed Boundary Method as used for this project.

6-1-3 Velocity extrapolation

Because the thickness of the shell element is assumed to be small, the rotational velocity of the element is neglected when extrapolating the velocity of the two-dimensional structure mesh to the extrusion mesh. Thus only the translational velocity of the nodes as obtained from the displacement difference between time step using forward Euler is used.

Alternatively, a more consistent way to prescribe the velocity on the extrusion mesh would be to obtain the nodal velocity on the extrusion mesh from the node movement between time steps.

6-1-4 Validation

The validation of the interpolation scheme is done in two steps. First, it is verified that the volume of the mesh is conserved by imposing a uniform force field. For deformed geometries due to the straight edge lines of the extrusion mesh elements, this is only satisfied approximately. Second, the consistency of the integration scheme is checked by imposing simple discontinuous force fields on the extrusion mesh and checking with the analytical solution. Here, the integration is fully accurate because first-order basis functions are used for the force imposition. Due to the simplicity of the interpolation scheme, the details of the validation are not documented here.

6-2 Body fitted mesh approach

For the conventional ALE approach in FOAM-FSI, a very similar interpolation approach can be taken as for the Immersed Boundary approach. The following subsections point out the subtle differences and explain the interface interpolation method in more detail.

6-2-1 Mesh extrusion

In theory, foam-extend-3.2 should be able to model zero-thickness surfaces, however within the context of this project some issues were encountered and thus the simplest solution was to actually model the thickness of the surface. Thus, the same mesh extrusion algorithm as for Fluidity was used. The only difference is that for an ALE approach only the interface node location is of interest. Moreover, because foam-extend is a Finite Volume solver, it is less sensitive to the types of element used for the surface mesh. However, since only 2D fluid simulations are run within the context of this project, this does not matter.

6-2-2 Force projection

The methodology implemented in Fluidity is based on volume forces, whereas in foam-extend the methodology is based on surface forces. Thus, the force integration is only carried out along the interface surfaces. According to standard conventions, the stress as obtained from foam-extend is calculated at the centre of the line elements. Subsequently, the stress distribution can be integrated to yield the nodal force values on the extrusion mesh. In this work, the stresses are assumed to be constant between two neighbouring centres of a line element. The force integration is validated through comparison with the internal force calculation in foam-extend. Then, the same force projection method as for the Immersed Boundary approach is used to obtain the nodal forces on the structure mesh. Again, since a 2D fluid model and a 3D structure model is used, the fluid forces are extruded and scaled to the structure mesh to obtain a quasi-3D load distribution.

6-2-3 Velocity extrapolation

The node velocity that the fluid solver uses is calculated internally within foam-extend; hence this step is not needed for this methodology.

6-2-4 Interface interpolation

Since in general, the fluid and structure mesh do not match at the interface, an interpolation scheme needs to be implemented. Hence two transfer matrices H_{sf} and H_{fs} need to be derived that interpolate the surface forces \mathbf{f} from the fluid to the structure interface and the displacements \mathbf{q} from the structure to the fluid interface as below

$$\mathbf{q}_{\text{fluid}} = H_{sf}\mathbf{q}_{\text{structure}} \quad (6-6a)$$

$$\mathbf{f}_{\text{structure}} = H_{fs}\mathbf{f}_{\text{fluid}} \quad (6-6b)$$

such that kinematic and dynamic boundary conditions can be fulfilled at the interface between fluid and structure. Similar to the Geometric Conservation Law used for the derivation of the mesh velocity in the ALE framework, the interpolation should be consistent such that constant values on one side of the interface are interpolated to constant values on the other side of the interface. Additionally, for stability and accuracy of the solution, it would be desirable that the work over the interface is conserved. This is done by establishing a relationship between the two interpolation steps such that the interpolation from the fluid to the structure and reverse are coupled through an integration of the work transmitted over the interface. However, depending on the chosen interpolation scheme the conservative interpolation method introduces unphysical pressure oscillations in the interpolated values [138].

Different approaches to interface interpolation exist based on either nearest neighbour interpolation, projection methods or spline interpolation. If the temporal and spatial discretisation error in the fluid and the structural solver is of order two or lower, de Boer et al. (2010) [138] recommend using a compact Radial Basis Function (RBF) interpolation with a high support radius and a consistent interpolation approach to avoid oscillations. If the order of the

system is higher than two, they recommend using the more expensive conservative Gauss Interpolation approach (similar to Galerkin projection in Fluidity), because in case of the RBF approach the interpolation error over the interface would be larger than the discretisation error of the system itself.

Consequently, for this project given that the system discretisation order is lower or equal than two, a consistent RBF approach will be used. For a generic spline interpolation function $w(\mathbf{x})$ with addition of a linear polynomial $q(\mathbf{x})$ that ensures consistency of the interpolation can be written as

$$w_A(\mathbf{x}) = \sum_{j=1}^{n_A} \gamma_j \phi(\|\mathbf{x} - \mathbf{x}_{Aj}\|) + q(\mathbf{x}) \quad (6-7)$$

where n_A refers to the number of interface points \mathbf{W}_{Aj} of the reference side of the interface. For obtaining a consistent transfer matrix H_{AB} two interpolation conditions have to be fulfilled

$$w_A(\mathbf{x}_{Aj}) = \mathbf{W}_{Aj} \quad \forall j \in [1, n_A] \quad (6-8a)$$

$$\sum_{j=1}^{n_A} \gamma_j s(\mathbf{x}_{Aj}) = 0 \quad (6-8b)$$

for all polynomials $s(\mathbf{x})$ with degree less than or equal to that of the polynomial $q(\mathbf{x})$. The first condition enforces that the source interpolation points are exactly recovered.

For this project, Thin Plate Spline functions with global support will be used, since this type of spline function is expected to perform well for this type of problem [139]. A thin plate spline function is given by the function below

$$\phi(\|\mathbf{x}\|) = \|\mathbf{x}\|^2 \ln \|\mathbf{x}\| \quad (6-9)$$

When using this in combination with a linear polynomial that ensures consistency of the interpolation scheme a linear system of equations can be constructed based on equations 6-8. The derivation of the transfer matrices involves the inversion of two matrices roughly of the same size as the number of the fluid and the structure interface nodes, respectively. However, since it is assumed that interface deformation is not too large, the construction of the conversion matrices is only done initially, and thus the computational effort necessary for the matrix inversion is not significant.

Lastly, it should be noted that while in preCICE different interpolation schemes are implemented, the author was not able to use the RBF interpolation in 2D. Thus, the interpolation scheme was implemented internally in the Python solver. Such that in fact, preCICE uses Nearest Neighbour interpolation.

Coupled simulations

In this chapter, a validation of the full solver methodology on three strongly coupled benchmark cases is carried out. First, in section 7-1 the model is applied to a classic FSI benchmark case positioned in the laminar flow regime. Second, in section 7-2 a two-dimensional membrane wing at a Reynolds number of above one million is modelled. Lastly, in section 7-3 a two-dimensional single membrane sail wing constellation with a rigid leading edge is investigated at a Reynolds number of roughly a third of a million.

7-1 Turek & Hron Benchmark

The results for the Turek & Hron FSI1 & FSI3 benchmark are presented below; the FSI2 case is left out because there is only limited reference data available. Moreover, the second setup contains quite large deformations and in initial attempts proved more difficult to model. Additionally, given the limitation of the shell element as described in subsection 3-5 for modelling the comparatively thick flag structure, it was decided not to investigate the FSI2 case in more detail. The structure and fluid properties are listed in table 7-1, information on the geometry of the setup have already been in listed in table 2-1. The details of the implementation on the fluid side with regards to boundary conditions and mesh requirements were already described in subsection 4-3-2. The fluid solver settings can be found in appendix C-1-1. For the coupling algorithm in preCICE the IQN-ILS method was chosen with a relatively strict coupling tolerance of $\epsilon_{coupling} = 1e-6$, because otherwise, unphysical pressure oscillations would introduce noise in the interface forces. Moreover, two mesh interpolation approaches were tested. First, the most simple setup with Nearest Neighbour interpolation and coinciding structure and fluid mesh nodes was tested. Second, consistent Thin Plate Spline Radial Basis Functions with a global support radius were used. In case, RBF functions are used for the interface interpolation, the structure is discretised using $N_s = 80$ equidistant elements. The resolution is chosen accordingly, because from the separate validation of the structural solver in subsection 3-5 on this particular setup it was determined that using this number of elements is expected to give the best results. Further increasing the number

of elements did not improve the accuracy of the solution and lead to the introduction of convergence issues. For the fluid mesh, the same three meshes as for the separate validation of the fluid solver with the number of elements ranging between roughly $N_{cells} \approx 6,000$ and $N_{cells} \approx 53,000$ are used.

	FSI1	FSI2	FSI3
$\rho_S [kg/m^3]$	10^3	10^4	10^3
$\nu_S [-]$	0.4	0.4	0.4
$E_S [N/m^2]$	$1.4e6$	$1.4e6$	$5.6e6$
$\rho_F [kg/m^3]$	10^3	10^3	10^3
$\nu_F [m^2/s]$	0.001	0.001	0.001
$Re = \frac{\bar{U}_d}{\nu_F} [-]$	10	20	200

Table 7-1: Fluid and structure properties for the benchmark case

The relative error as compared to the results from Turek & Hron [66] was evaluated by comparing the tip displacement of the beam in the lateral and transverse direction, as well as, the integrated body force exerted by the fluid so lift and drag of the body.

The mesh convergence results for the steady FSI1 case are shown in figure 7-1. When Nearest Neighbour interpolation with coinciding nodes is used, the convergence of the solution on the mesh is not given due to the restrictions of the structural solver and the convergence issues for higher resolution of the structure mesh. However, when using consistent RBF interpolation with a fixed number of structural elements and varying density fluid meshes, when going from about 24,000 to 53,000 elements the change in the interface forces and as well as the interface displacements is smaller than 1 %. Thus, for a fixed number of structural elements, the solution seems to converge on the mesh. Now looking at the accuracy of the results as compared to the benchmark results for a fixed number of structural elements with RBF interpolation. The deviation is largest for the displacements in the lateral direction with close to 25 % directly followed by the displacements in the transverse beam direction with roughly 9 %. This was expected given the shortcomings of the structural solver for this particular test case. Since the spread in results is relatively small, comparison with the other reference results for this case as given in Turek et al. [1] was omitted.

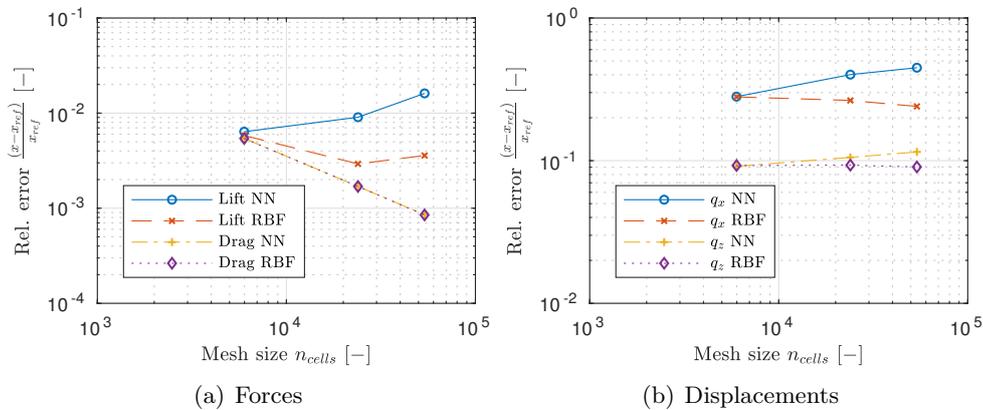


Figure 7-1: Mesh convergence of the FSI1 benchmark case

The mesh convergence results for the unsteady, periodic FSI3 case are shown in figure 7-2. Again the comparison with the results from Turek & Hron [66] is carried out. Direct comparison with the other reference results as given in Turek et al. [1] is omitted, because the spread in results is relatively small, as well as, Turek & Hron [1] obtained the most consistent convergence of the results with increasing mesh density. Again, due to the restrictions of the structural solver, coinciding meshes with Nearest Neighbour interpolation did not perform well. Thus, only the cases run with RBF interpolation are discussed in more detail. It should be noted that for the temporal discretisation of the structure the Implicit Euler scheme with a time step of $\Delta t = 0.0005$ was used for all the transient simulations. Given the difficulty of this test case, the discrepancies between simulation and reference results in the averaged mean, and amplitude values are quite high for some of the observed quantities. However, some of these parameters are hard to predict or expected to be badly predicted. First, the longitudinal beam displacements are not well modelled by the structural model. Second, the drag amplitude, as well as, the mean lift values are quite small and thus difficult to attain. For the remaining parameters, the maximum errors in the mean as well as amplitude values for the remaining parameters are at worst about 10 % and show a clear trend with the refinement of the fluid mesh.

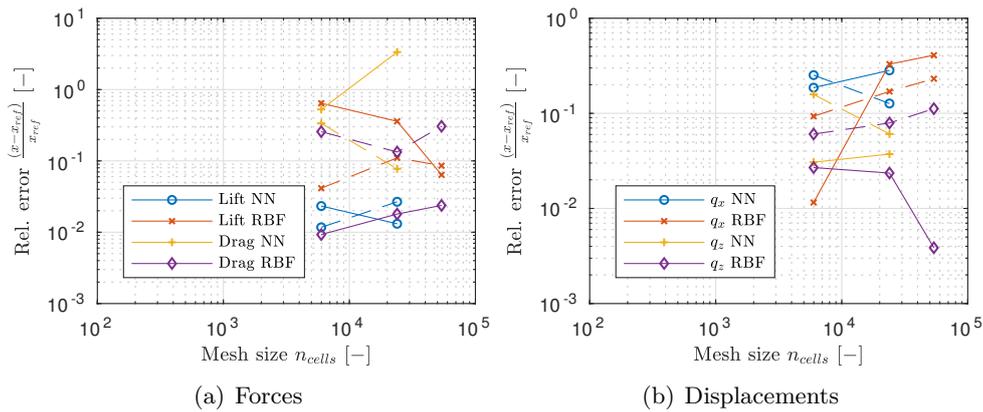


Figure 7-2: Mesh convergence of the FSI3 benchmark case with mean (—) and amplitude values (---) of the forces on the body and the tip displacement

Moving on to figure 7-3 and comparing with the time series plots from Turek & Hron [66] shows a clear correspondence between the results even though the mesh is not fine enough to indicate mesh-independence of the results. Thus, it can be concluded that qualitatively the results agree well with the reference results. However, the quantitative agreement is poor given the restriction of the structural model and possibly also the insufficient resolution of the fluid mesh as indicated by the results from the separate validation of the fluid solver for this particular setup.

To illustrate the full simulation setup, figure 7-4 shows the deformed configuration for the FSI3 case using the finest fluid mesh with roughly $n_{cells} \approx 53000$ cells including also the interface nodes of the structure.

Lastly, it should be noted that the initial simulation results showed very strong pressure oscillations even for very small time steps that eventually lead to the divergence of the coupled simulation. The origin of the oscillations was traced back to the undamped Crank-Nicolson

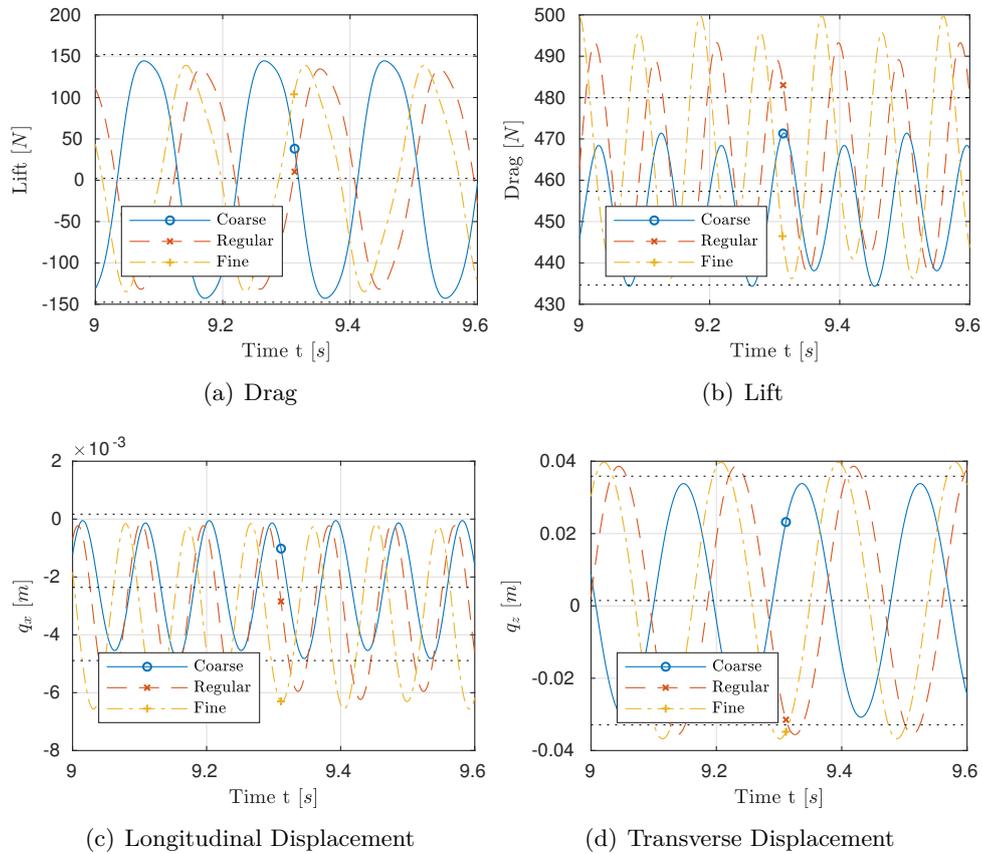


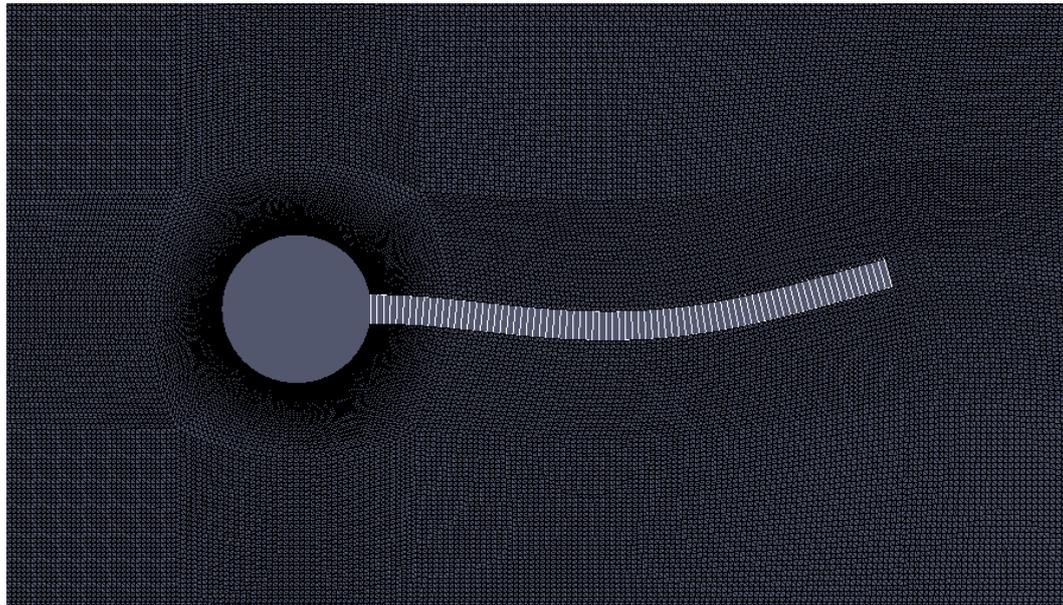
Figure 7-3: Time series results for the FSI3 case using RBF interpolation and varying density fluid meshes and the mean and extreme values of the reference results in black (:)

time integration scheme of the structure. Thus, in the following, at least for cases where no structural damping is applied, the Implicit Euler scheme is used for the integration of the structure to stabilise the coupled simulations. This diminishes the accuracy of the time integration scheme but is in this case necessary to obtain converging results.

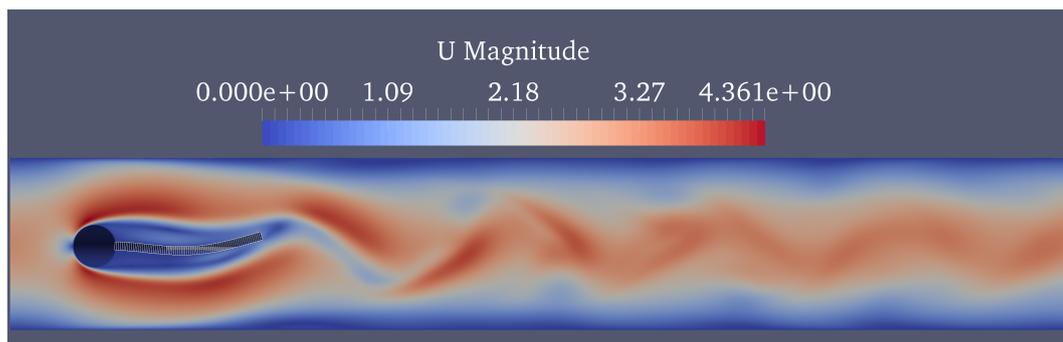
7-2 Greenhalgh setup

This section will apply the solver to a two-dimensional membrane wing constellation set in a flow regime with a Reynolds number of above one million, where experimental reference data is available from Greenhalgh et al. [3, 84, 85]. The experimental setup was already shown in figure 2-4 in the literature review section. The following subsections outline the different aspects of the case.

Sections 7-2-1, 7-2-2 and 7-2-3 outline the entire modeling process as well as the mesh convergence study. Sections 7-2-4, 7-2-5, 7-2-6 describe the results using a steady-state and unsteady model, as well as, the initialization procedure for the coupled simulations.



(a) Mesh for the solid (white) and the fluid (black)



(b) Flow field

Figure 7-4: Snapshot of the results for the FS13 benchmark case

7-2-1 Setup and modelling simplifications

Even though the experimental setup from Greenhalgh is quite elementary, the following modelling simplifications are made.

- The canopy material is wrapped around rigid, elliptical metal rods at the leading and trailing edge with a fixed chord length C . However, depending on the local inflow angle the rods are mounted such that they can freely rotate around their centre. Because this would require the implementation of unsteady boundary conditions for the structural solver, the modelling of the rotation of the supports is neglected.
- On a similar note, because in the experiment the material is wrapped around the elliptical supports and has virtually no bending stiffness at the attachment point depending on the inflow angle, the material will be tangential to the support outline. However,

this would be very difficult to model. Hence, to ensure that the membrane does not overlay with the supports a nonlinear boundary condition would have to be introduced. Thus, in the modelling process special care has to be taken to avoid crossover of the membrane into the supports. Thus, the rotation about the attachment point is set to zero in the structure model.

- Lastly, given that, in general, the order of the thickness of membrane wings is smaller than 1 millimetre, it would be easiest to model it as a zero thickness surface with regards to mesh generation and mesh deformation. However, with the external meshing tool that was used for this project the author was not able to successfully import the zero thickness surfaces (called baffles) in foam-extend-3.2. Even though, the same setup worked in OpenFOAM-3.0.0. Thus, the membrane was modelled with a finite thickness of h .

A preliminary experiment report by Greenhalgh [85] describes the investigation of the sensitivities of the lift, drag, as well as tension measurements of the setup with regards to the membrane thickness, the membrane porosity, the support geometry and the side leakage in the wind tunnel. In particular, the influence of the support radius is interesting, since a modelling simplification is made there. The report states that for a thickness below 0.5 % of the chord length, the influence of the leading edge and trailing edge radius of the support is negligible. For the given chord length this refers to $r_{LE} \approx 2.285e-3 < s_x, s_y$ such that modifications of the supports are expected to affect the accuracy of the simulation results.

The confidence placed in the absolute reference results is limited because the publications have some limitations, in particular, the canopy properties are not defined. Also, conflicting statements regarding the thickness of the membrane were found in the two different publications on the experiment. The first publication [85] states a thickness of $h = 1e-4m$ whereas the latest publication [3] states a thickness of $h = 1e-3m$. Since apart from this the same wing dimensions are used, it is concluded that a mistake was made when converting to standard units. Moreover, it should also be mentioned that wind tunnel corrections have to be applied to the measurement results to account for blockage and other boundary layer effects in the wind tunnel. Otherwise, the results are not truly two-dimensional. The application of such corrections is not mentioned in any of the reference papers. However, it is assumed that this was done.

Finally, three different simplified geometries were investigated as detailed in table 7-2 and illustrated in figure 7-11. The first model uses round supports with the membrane attached at the centreline of the supports to avoid crossover of the membrane and sensitivity to a particular inflow angle. The second model uses oval supports as described in the preliminary experiment report by Greenhalgh and the membrane is attached at the upper corner of the supports. The third model completely neglects the supports. Table 7-2 summarises the geometric properties of the different profiles.

7-2-2 Mesh convergence

In the following, the meshing methodology for the fluid as well as the structural solver is outlined. Then a steady-state mesh convergence analysis is carried out, and the sensitivity of the fluid solver results to the inflow turbulence is briefly investigated as well.

	Experiment	Round supports	Oval supports	No Supports
Chord $C[m]$	0.457	0.457	0.457	0.457
Supports $s_x [m]$	$1.27e-2$	$3.175e-3$	$1.27e-2$	$1e-3$
Supports $s_y [m]$	$3.175e-3$	$3.175e-3$	$3.175e-3$	$1e-3$
Membrane thickness $h[m]$	$1e-4$ or $1e-3$	$1e-4$	$1e-4$	$1e-3$

Table 7-2: Geometry information for the Greenhalgh setup [85] including the modelling simplifications

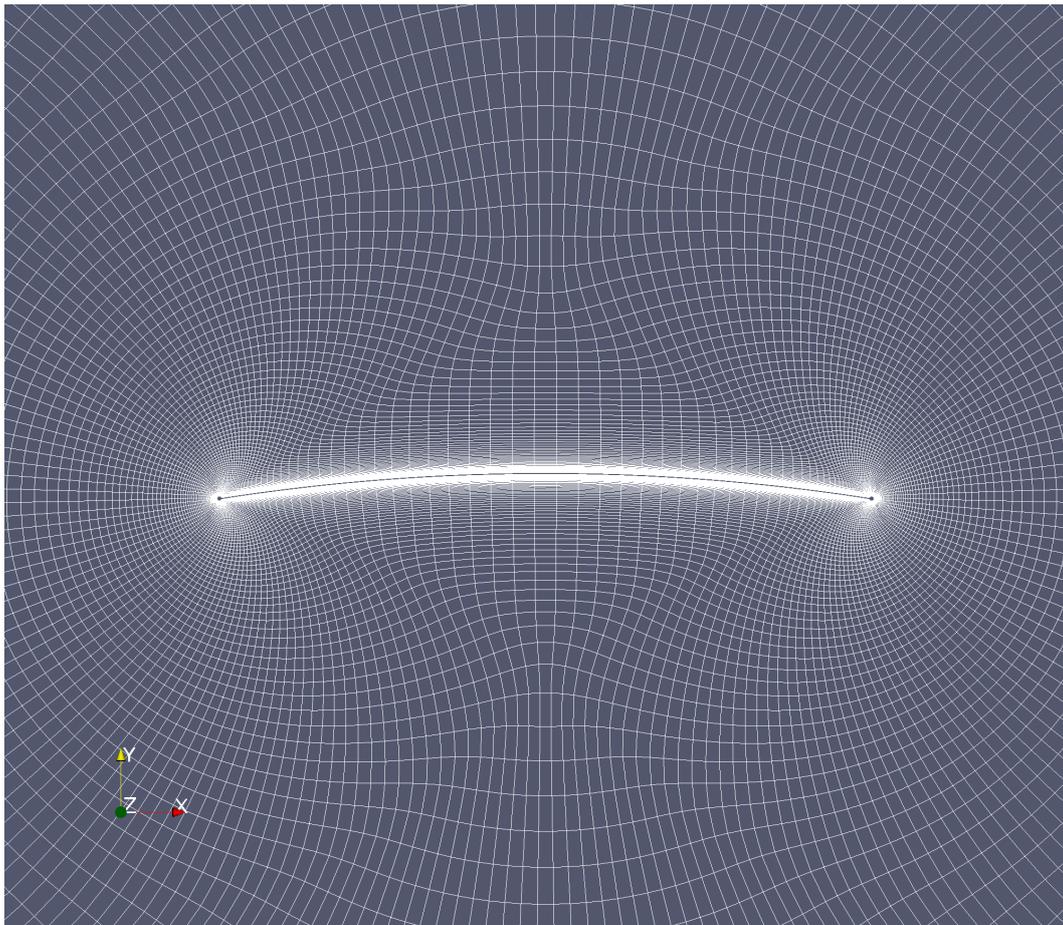


Figure 7-5: O-mesh topology of fluid mesh for Greenhalgh setup

Mesh convergence fluid solver The meshing methodology is such that only one initial mesh is created using Pointwise and membrane excess or deformed initial configurations of the setup are generated by using FOAM-FSI's mesh deformation algorithm. General meshing guidelines are adhered to for this initial mesh. The placement of the first cell away from the wall is such that normalised wall spacing fulfils $y^+ < 1$ as required by the turbulence model and the wall expansion ratio is kept below 1.2 as necessary for appropriately capturing the boundary layer [120]. The full fluid mesh with round supports is shown in figure 7-5. Lastly, it should be noted that a safety factor of about 30 % is included in the calculation of the first wall node location to account for the deformation of the setup.

	Coarse	Regular	Fine	Extra Fine
Round supports				
Nr of wall normal nodes n_{normal}	80	160	227	320
Nr of nodes on wing surface $n_{surface}$	196	376	534	756
Nr of cells n_{cells}	15680	60160	121218	241920
Wall spacing y_0 [m]	6e-6	6e-6	6e-6	6e-6
Oval supports				
Nr of wall normal nodes n_{normal}	76	149	213	300
Nr of nodes on wing surface $n_{surface}$	207	420	586	848
Nr of cells n_{cells}	15732	62580	124818	254400
Wall spacing y_0 [m]	6e-6	6e-6	6e-6	6e-6
No supports				
Nr of wall normal nodes n_{normal}	80	160	226	320
Nr of nodes on wing surface $n_{surface}$	172	348	494	700
Nr of cells n_{cells}	13760	55680	111644	224000
Wall spacing y_0 [m]	6e-6	6e-6	6e-6	6e-6

Table 7-3: Fluid solver mesh convergence for the Greenhalgh setup

The convergence of the solution on the fluid mesh is also studied by using four different meshes for each support model as detailed in table 7-3 on the rigid wing at an angle of attack of $\alpha = 5^\circ$ for a Reynolds number of $Re \approx 1.3e6$. Because the undeformed configuration of the membrane wing was considered, the setup essentially simplifies to a flat plate with an angled inflow. Since the experimental observations describe a steady behaviour of the coupled system for inflow angles smaller than 15 degrees and membrane excess lengths up to 5 %, the main objective for this project is to obtain steady-state numerical results. Thus, the mesh convergence is carried out only for the spatial discretisation and not the temporal discretisation by employing a steady-state fluid solver. The spatial discretisation settings, the solver settings as well as the utilised boundary conditions can be found in appendix C-1-2. In theory, second-order schemes are used for the spatial discretisation on the fluid mesh, however, in practice, the convergence of the error is usually slightly lower. Given the results of the lift and drag for the four different refinement levels, the order p of the leading error term of the spatial discretisation can be obtained as follows

$$f(h) = f_{converged} + C \cdot h^p \quad (7-1a)$$

$$p = \frac{\log\left(\frac{f(r^2h) - f(rh)}{f(rh) - f(h)}\right)}{\log(r)} \quad (7-1b)$$

where f is a generic functional, C is a resolution independent constant, h is a characteristic resolution length and $r = \frac{h_1}{h_2}$ is the ratio between the characteristic resolution lengths of two different meshes. For simplicity, the grid refinement ratio is chosen as $r = 2$, such that the number of nodes along the block edges is doubled when going from one level of refinement to a more refined one. Given the actual order of the spatial discretisation, Richardson extrapolation can be used to get an estimate of the asymptotic value of the functional according to

$$f_{converged} = f(rh) + \frac{f(rh) - f(r^2h)}{r^p - 1} \quad (7-2)$$

Based on this, an estimate of the error can be obtained, and a mesh refinement level which is expected to provide a reasonable trade-off between accuracy and computational effort can be picked. [140]

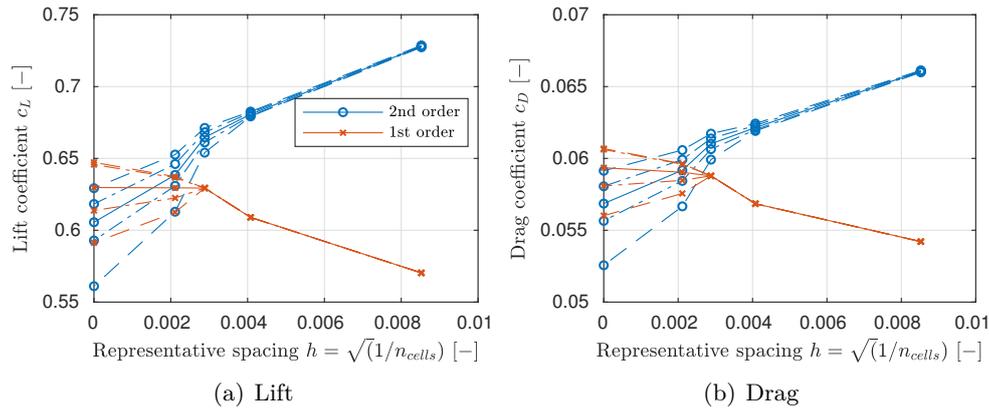


Figure 7-6: Mesh convergence for the Greenhalgh setup with round supports for $\alpha = 5^\circ$ and $Re = 1.3e6$ using a first and second order scheme to discretise the convective terms in the Navier-Stokes equations with mean (—), extreme (---) and standard deviation (---) values of the integrated force coefficients including the Richardson extrapolation thereof

However, given the low leading edge radius of the configuration, the flow around the rigid wing was found to be not entirely steady with a periodic vortex shedding occurring in particular at the leading edge and less pronounced also at the trailing edge [141]. Depending on how the supports are modelled, this is more or less amplified. Thus, the relative velocity and pressure residuals of the steady-state solver do in general not converge to the desired value of $\epsilon = 1e-5$. However, after a certain number of iterations, a periodic oscillation of the residuals occurs, indicating the presence of vortex shedding. Of course, a steady-state solver cannot capture unsteady developments in the flow; however, failure in the convergence of the steady-state solver indicates that the problem is in fact unsteady. For the mesh convergence study, an average over the last 10,000 iterations of the force coefficients from the steady-state solver is used where a total of 50,000 iterations are performed. The mesh convergence results for the three different support models are shown in figures 7-6, 7-7 and 7-8.

The setup using the round supports (figure 7-6) with the attachment point at the centreline of the supports clearly indicates the presence of a strong vortex shedding yielding unsteady results even when the highly dissipative upwind scheme is used for the discretisation of the convective terms. Depending on the resolution of the mesh, the oval supports (figure 7-7) prompt vortex shedding at the leading edge. Nevertheless, a clear trend of the results with mesh refinement is visible. Judging from the convergence of the integrated force coefficients only, the approach where modelling the supports (figure 7-8) is neglected, seems to yield the weakest vortex shedding of the three models. However, looking at the residual plots shows that a larger oscillation in the residuals is present as for the case with oval supports.

Figure 7-9 shows the lift and drag polars for the three different support models using the mesh

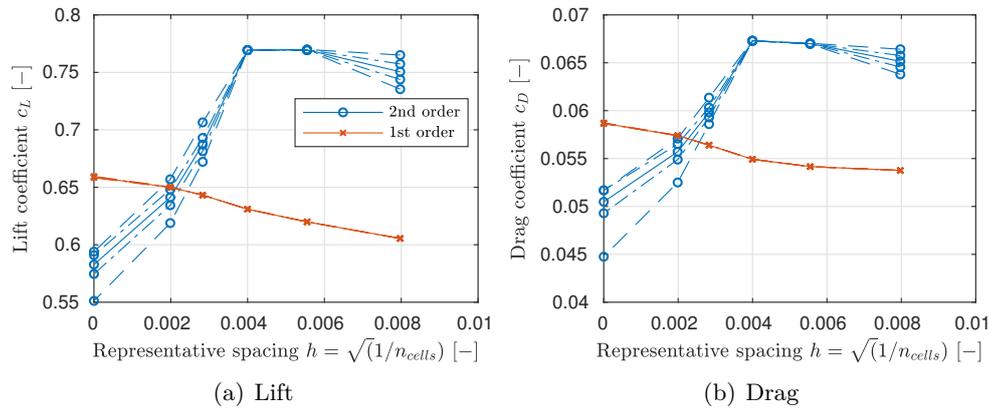


Figure 7-7: Mesh convergence for the Greenhalgh setup with oval supports for $\alpha = 5^\circ$ and $Re = 1.3e6$ using a first and second order scheme to discretise the convective terms in the Navier-Stokes equations with mean (—), extreme (---) and standard deviation (---) values of the integrated force coefficients including the Richardson extrapolation thereof

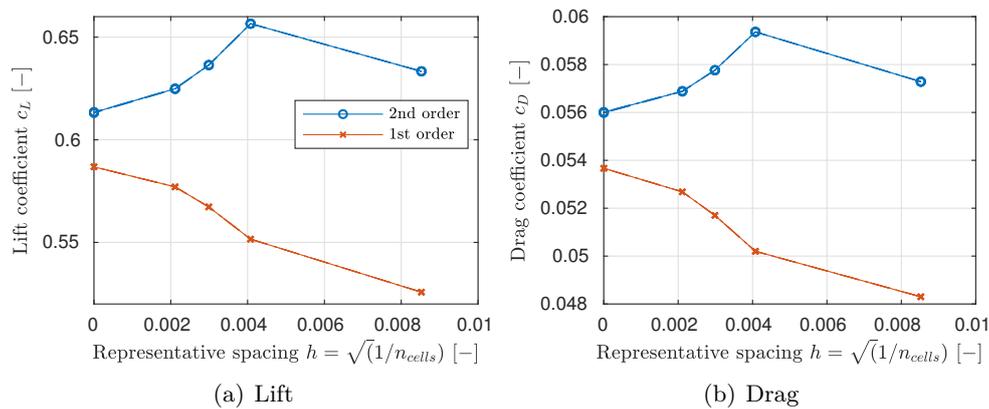


Figure 7-8: Mesh convergence for the Greenhalgh setup without supports for $\alpha = 5^\circ$ and $Re = 1.3e6$ using a first and second order scheme to discretise the convective terms in the Navier-Stokes equations with mean (—), extreme (---) and standard deviation (---) values of the integrated force coefficients including the Richardson extrapolation thereof

resolution titled as regular in table 7-3. The previous observation regarding the existence of a steady-state solution is further confirmed.

Going forward the setup using the oval supports and the one neglecting to model the supports will be further investigated in the fully coupled simulations because they correspond best with the experiment results. It should be noted, that the target mesh size of roughly $n_{cells} \approx 50,000$ cells does not yield mesh-independent results yet. However, given the computational restrictions of this project, it makes sense to at least initially not pursue a very high-resolution mesh. The size of the mesh is also in line with what is used for a publication on an airfoil benchmark case in OpenFOAM [142]. Moreover, while the experiment description presents steady lift and drag values, it is also mentioned that oscillations of the membrane are observed. Thus, given the low leading edge radius of the setup, it is entirely possible, that the experiment setup, in fact, shows some unsteady effects that were however averaged out in the final results.

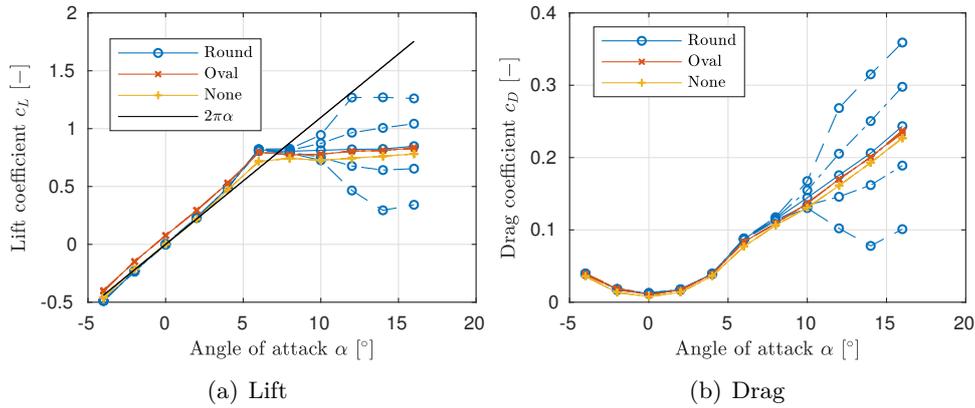


Figure 7-9: Lift and drag polars for the undeformed profiles using the three different support models for a Reynolds number of $Re = 1.3e6$ and a mesh size of roughly $n_{cells} \approx 50,000$ cells with mean (—), extreme (—) and standard deviation (—) values of the integrated force coefficients

Inflow turbulence Since the κ - ω SST turbulence model solves two transport equations for the turbulence kinetic energy κ , as well as the specific dissipation rate of the turbulence kinetic energy ω , boundary conditions need to be set for these properties. For external flows, as well as internal flows, different correlations are used. Because here an external flow is present external flow correlations are used that estimate the turbulence kinetic energy and its specific dissipation rate in the inflow based on a specified turbulence intensity, the freestream velocity and the eddy viscosity ratio [143]. Generally, for wind tunnel experiments low turbulence intensity, as well as eddy viscosity ratios, are expected. However, the reference paper from Greenhalgh [84] does not refer to the turbulence intensity in the flow. Thus, generic values are used.

The sensitivity to the inflow turbulence settings is superficially investigated as well using the steady-state solver at a Reynolds number of $Re = 1.3e6$ and an inflow angle of $\alpha = 5^\circ$ for the oval supports. The different inflow turbulence configurations, as well as the integrated force coefficients, are listed in table 7-4.

	Case 1	Case 2	Case 3	Case 4
Eddy viscosity ratio $\frac{\mu_t}{\mu}$ [-]	0.01	0.053	0.053	0.1
Turbulence intensity TI [%]	0.1	0.1	0.5	1.0
Averaged lift coefficient \bar{c}_L [-]	0.6666	0.6664	0.6515	0.5927
Standard deviation lift coefficient σ_{c_L} [-]	0.0016	0.0006	0.0053	0.0077
Averaged drag coefficient \bar{c}_D [-]	0.0561	0.0561	0.0548	0.0484
Standard deviation lift coefficient σ_{c_D} [-]	1.829e-4	1.239e-4	5.218e-4	7.961e-4

Table 7-4: Effect of inflow turbulence settings on steady-state results for the Greenhalgh setup at a Reynolds number of $Re = 1.3e6$ with oval supports

The results indicate that the influence of the turbulence intensity is stronger than the influence of the eddy viscosity ratio. But overall, at least for the first three cases, the impact of the inflow turbulence is not significant. Nevertheless, going forward it is assumed that the turbulence properties correspond to the settings from Case 1.

Mesh convergence structural solver Similarly, a mesh convergence study for the structure mesh was also carried out for this setup. Given the element formulation and the load distribution, it was not deemed necessary to use non-uniform elements, such that the only tuning parameters remained the number of elements along the membrane axis. Figure 7-10 shows the mesh convergence of the membrane excess length for a given analytical, continuous stress distribution. The stress function was chosen somewhat similar to what the aerodynamic loading is expected to look.

The stresses were imposed on the extruded fluid mesh that will be used for the coupled simulations in the following such that the RBF interface interpolation has to be applied to transfer the stresses to the extruded structure mesh. This is why the convergence of the total force on the structure is also shown in the convergence plot in figure 7-10(b). Second, the loaded excess length of the membrane under the constant load picture in figure 7-10(a) for different Young's Moduli is also shown.

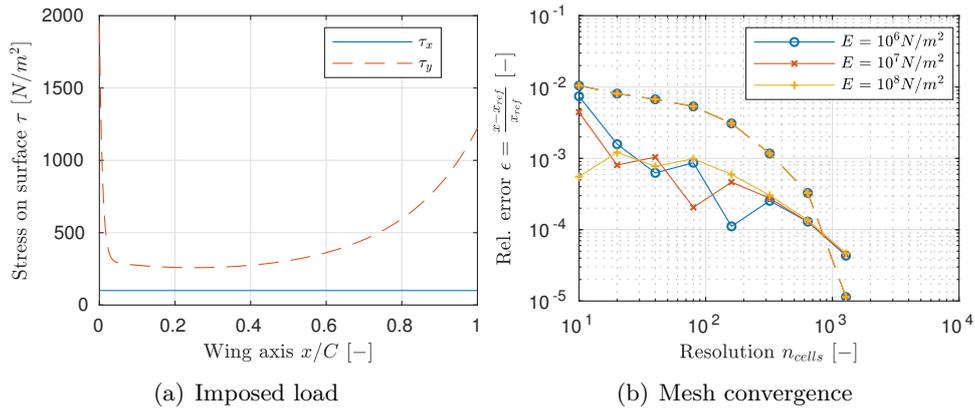


Figure 7-10: Mesh convergence structure solver for Greenhalgh setup for different Young's Moduli with deformed excess length of the membrane ranging between 3 and 14 % loaded with a fixed pressure distribution for the loaded length of the membrane (—) and the total force acting on the structure (---) as compared with reference simulation run with $n_{cells,structure} = 2560$

Two conclusions can be drawn from the convergence plot. First, the convergence occurs quite fast, such that if more than roughly $n_{cells,structure} > 100$ elements are used, the effect on the structural model is expected to be negligible, and the interpolation error seems to remain constant. Second, the convergence issues that were observed for the CSM Turek & Hron benchmark case once the element number was increased past a certain threshold are not present here. This could indicate, that the issues observed might only occur if the given shell element formulation is utilised on comparatively thick bodies, where the modelling assumptions applied in the derivation of the element formulation do not hold anymore. Thus, in the following, the number of structural elements is chosen to roughly match the number of the fluid elements along the interface of the fluid and solid mesh to reduce interpolation errors and obtain a high quality of the deformed fluid mesh.

7-2-3 Mesh deformation

As already outlined in subsection 4-3-1, given an updated location for the fluid interface nodes, Thin Plate Spline radial basis functions with adaptive mesh coarsening are used for the fluid mesh deformation algorithm in FOAM-FSI. In addition to this, Thin Plate Spline radial basis functions are also used to transfer the node displacement on the extruded structure to the fluid mesh. Given the large deformations and the small leading and trailing edge radius for this particular setup, the corner nodes of the interface had to be additionally constrained to ensure a high quality deformed mesh. Otherwise, in particular, a crossing of the grid lines led to the divergence of the solver. Conversely, this was not necessary for the Turek & Hron benchmark case given the less delicate geometry.

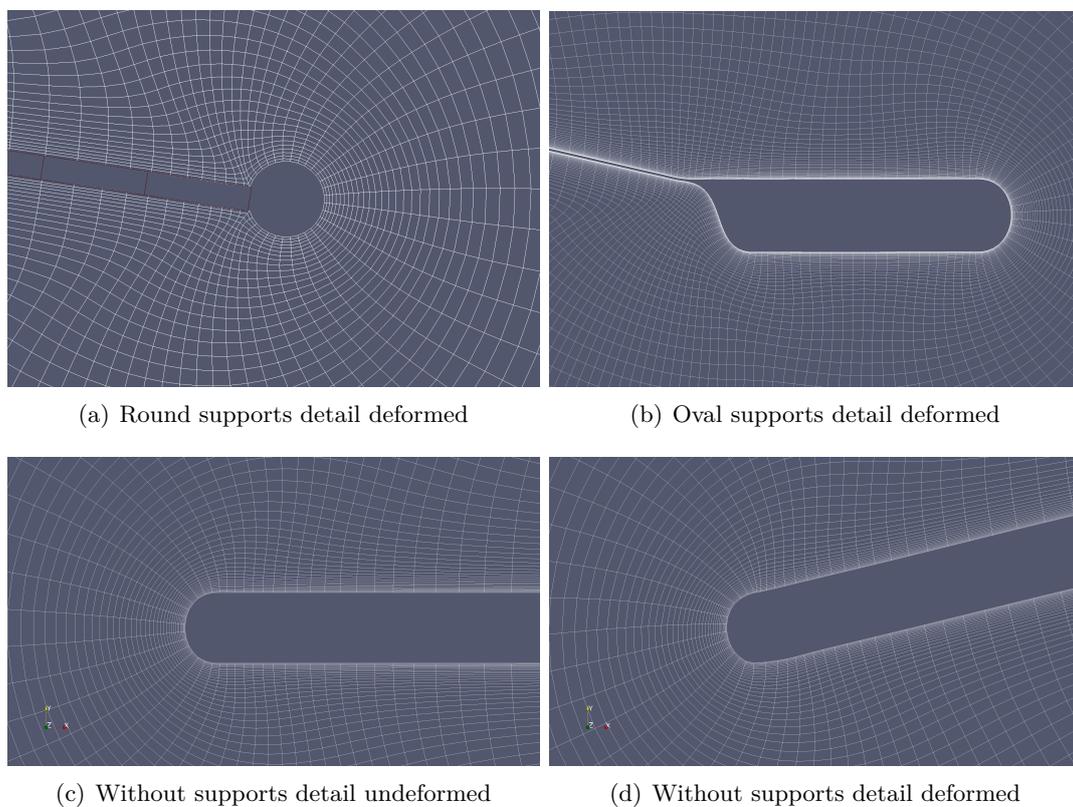


Figure 7-11: Illustration of the transition between fixed and deforming nodes of the fluid mesh for two different supports type models

Initially, different approaches were tested, however, in the end, the most successful methodology was as follows. The nodes on the extruded structure mesh were not constrained, then the consistent radial basis function approach was used to transmit the displacements from the extruded structure mesh to the fluid mesh and finally the corner interface nodes of the fluid mesh were corrected to their initial undisplaced location. This worked well for the round supports, however for the model where the supports were neglected, this resulted in the crossover of the neighbour node of the corner node over the corner node. Thus for this setup, the displacements of the corner node were set to zero and displacements of the direct neighbouring node were obtained through linear interpolation between its two neighbouring

nodes. For larger deformations, higher order interpolation might be necessary to guarantee a high mesh quality. Alternatively, the calibration of the Radial Basis Functions could also include the nodes of the supports and could thus yield a smoother transition between the fixed support and the moving interface nodes of the fluid interface mesh. Figure 7-11 shows examples of the deformed mesh for all the investigated support types.

Lastly, it should be noted that the Thin Plate Spline radial basis functions were only calibrated before the first time step of the coupled simulation. It is assumed that the relationship between the nodes remains approximately the same even in the deformed configuration. This assumption holds up if the deformations are not too large.

7-2-4 Choice of configurations and initialization of the simulation

Greenhalgh's experiment description investigates the membrane wing at a chord-based Reynolds number of roughly one million at different inflow angles and excess lengths. The excess length of the membrane is defined as the difference between the length of the membrane l_{membrane} and the chord length C as in equation 7-3. According to this definition, Greenhalgh et al. [3] investigated excess lengths between 0.21 % and 5.69 %. However, the experiment description leaves it unclear whether this refers to the loaded or the unloaded length of the membrane. Whereas the theoretical results from sail theory - that are also presented in the same publication - indicate that in fact the deformed membrane length is meant by the membrane length. Since a length constraint for the membrane is introduced in the system of equations for a generic membrane material. Nevertheless, it is difficult to enforce such a constraint in the model. Thus, an initial excess length corresponding to the experimental results was prescribed. Then, a Young's Modulus - due to lack of better reference data - corresponding to the material properties of ripstop Nylon for kites was used [144]. For comparison with the measurement results interpolation of the lift and drag coefficients according to the loaded membrane excess length was carried out.

$$\epsilon_{\text{wing}} = \frac{l_{\text{membrane}} - C}{C} \quad (7-3)$$

The detailed methodology that is used to obtain an initial condition for the membrane can be summarised as follows

1. Prescribe parabolic deformation shape of the membrane that leads to desired deformation length by solving the nonlinear equations below for the desired membrane length l_{membrane} , for the scaling parameter a leading to the transverse membrane displacements $q_{z,\text{prescribed}}$ according to

$$a = \sinh \left(2a \left(l_{\text{membrane}} - \frac{C}{2} \sqrt{1 + (ac)^2} \right) / C \right) \quad (7-4a)$$

$$q_{z,\text{prescribed}}(x) = -a (x^2 - Cx) \quad (7-4b)$$

2. Prescribe desired material and load properties

3. Find initial guess of loaded membrane shape using Vortex Particle Method
4. Initialize fluid with solution for rigid membrane shape as obtained from Vortex Particle Method using foam-extend's steady-state solver
5. Run either steady or unsteady fully coupled simulations in FOAM-FSI

From the experimental results, some general trends with respect to the excess length of the membrane can be discerned. Increasing the excess length of the membrane result in an upward shift of the lift curve due to the higher camber of the airfoil up to the point where flow separation starts to dominate the flow. However, increasing the excess length beyond 2 % resulted in a decrease of the lift slope due to increased areas of separated flow. For all excess lengths, the stall behaviour of the airfoil is benign as the loaded shape adjusts accordingly. Namely, the maximum camber point of the wing moves forward as the angle of attack grows. In turn, the impact of the trailing edge separation bubble on the integrated force on the wing is smaller as for a rigid profile. Larger excess length of the membrane was also associated with more significant vibration amplitudes of the membrane in regions of separated flow.

Given the time and computational restraints of this project, not all configurations described in the reference paper can be investigated. Moreover, since unsteady, strongly coupled CFD-CSM simulations can be very computationally expensive even in 2D, the aim is to have either steady or at least close to steady configurations. Moreover, because the fluid mesh is only around 50'000 cells, splitting the fluid domain and solving the domain in parallel is not expected to yield a large speedup of the required simulation time. Thus, two configurations with a membrane excess length of 0.21 % and 1.14 % are chosen. The details of the configuration are listed in table 7-5. Since the density of the structure is unknown and for this setup close to steady results are expected, the density was used as a tuning parameter for the convergence towards a steady-state solution. In fact, it was chosen such that the ratio between fluid and structure mass is roughly the same as for the Turek & Hron benchmark case according to the following ratio

$$\frac{\rho_{S,ref} \cdot h_{S,ref}}{\rho_{F,ref}} \approx \frac{\rho_S \cdot h_S}{\rho_F} \quad (7-5)$$

as a consequence roughly the same number of sub-iterations were necessary for convergence as for the benchmark case.

	Case 1a	Case 1b	Case 2a	Case 2b
Case specific settings				
Membrane excess length ϵ_{excess}	0.21%	1.14%	0.21%	1.14%
Support model	Oval		None	
Angles of attack α	[-4° , 16°]		[-4° , 16°]	
Structure properties				
Young's Modulus E_S	$1e8N/m^2$		$1e9N/m^2$	
Poisson ratio ν_S	0.3		0.3	
Density ρ_S	$250kg/m^3$		$25kg/m^3$	
Damping coefficient ζ	5%		5%	
Fluid properties				
Kinematic viscosity ν_F	$1.47e-5m^2/s$		$1.47e-5m^2/s$	
Density ρ_F	$1.23kg/m^3$		$1.23kg/m^3$	
Reynolds number Re	1.3e6		1.3e6	
Discretisation				
Number of structure elements $N_{cells,structure}$	150		150	
Number of modes considered for structural damping $N_{damping}$	70		70	
Size of fluid mesh $N_{cells,fluid}$	62580		55680	
Time step Δt	$1e-5s$		$1e-5s$	

Table 7-5: Simulation parameters

7-2-5 Steady results

Given that the reference experiments report steady aerodynamic forces for moderate inflow angles and excess lengths of the membrane, initially, FOAM-FSI's steady-state RANS solver was applied to the configuration. Investigating an inflow angle range ranging from $\alpha = -4^\circ$ to $\alpha = 16^\circ$ with an interval of $\Delta\alpha = 4^\circ$, the steady aeroelastic solver only delivered reasonable results for moderate inflow angles for $\alpha \geq 0^\circ$ and $\alpha \leq 8^\circ$ for some of the configurations. However, the convergence of the residuals below $\epsilon < 1e-4$ was not achieved. A closer inspection of the flow field showed again a weak vortex shedding at the leading edge for both supports types that indicated in fact weakly unsteady flow behaviour.

7-2-6 Unsteady results

Since the results from the steady-state solver indicated a periodic unsteady behaviour of the coupled system, unsteady simulations using foam-extend's unsteady RANS solver were also performed. However, unsteady simulations require very small time steps and have much more significant computational demands than steady ones. In particular, for the fluid, the Courant-Friedrichs-Levy (CFL) number has to be close to one for accuracy and for explicit methods also the stability of the method.

$$\text{CFL} = \frac{u\Delta t}{\Delta x} \leq C_{max} \quad (7-6)$$

While the fluid solver uses an implicit discretisation for the time-dependent terms in the Navier-Stokes equations, the pressure-momentum coupling is still solved explicitly if no outer iterations are performed. Thus, if the PISO algorithm is used CFL numbers smaller one impose a very strong limit on the time step for unsteady simulations especially for high Reynolds number flows where the whole boundary layer is resolved. Thus, for this particular case with regards to the external time and computational restraints, the aim is to run with a time step corresponding to a CFL number of roughly $\text{CFL} \approx 5$ using the PIMPLE algorithm. While this is still a very small time step, phenomena occurring on time scales smaller than the time step will not be resolved. It should also be noted, that convergence of the solution with the mesh and time step for unsteady simulations was not checked due to time limitations.

The detailed settings for the fluid solver with regards to discretisation and the solving algorithm can be found in appendix C-1-3.

Rigid wing In figure 7-12 the lift and drag polar obtained for the unsteady simulations with a fixed wing shape as obtained from the vortex particle method is shown for the two different support models. These results were obtained to have a frame of reference as to whether unsteady effects in the coupled system arise from unsteadiness in the flow pattern itself or from the coupling of the two systems. It should be noted that the supports are not modelled when using the Vortex Particle Method. For moderate angles of attack, convergence to steady lift and drag coefficients is observed despite not obtaining fully converged results with the steady-state solver. The cause of this is the dissipation introduced through the time integration scheme because of the large time step. In fact, the vortex shedding at the leading edge previously observed in the steady-state solver results is now visible as a fixed

separation bubble at the leading edge. For increasing angles of attack, the reattachment point of this separation bubble moves rearwards until finally the profile is fully stalled. Large-scale unsteady vortex shedding is observed for the largest simulated angle of attack $\alpha = 16^\circ$ and in one case also for $\alpha = 12^\circ$. The experimental report observed a gradual stall onset for all profiles independent of the excess length of the membrane. Thus, the wing shows a typical thin airfoil stall behaviour. As previously noted, in general, the oscillation amplitude in lift and drag is larger for the setup with oval supports as compared to the one without supports.

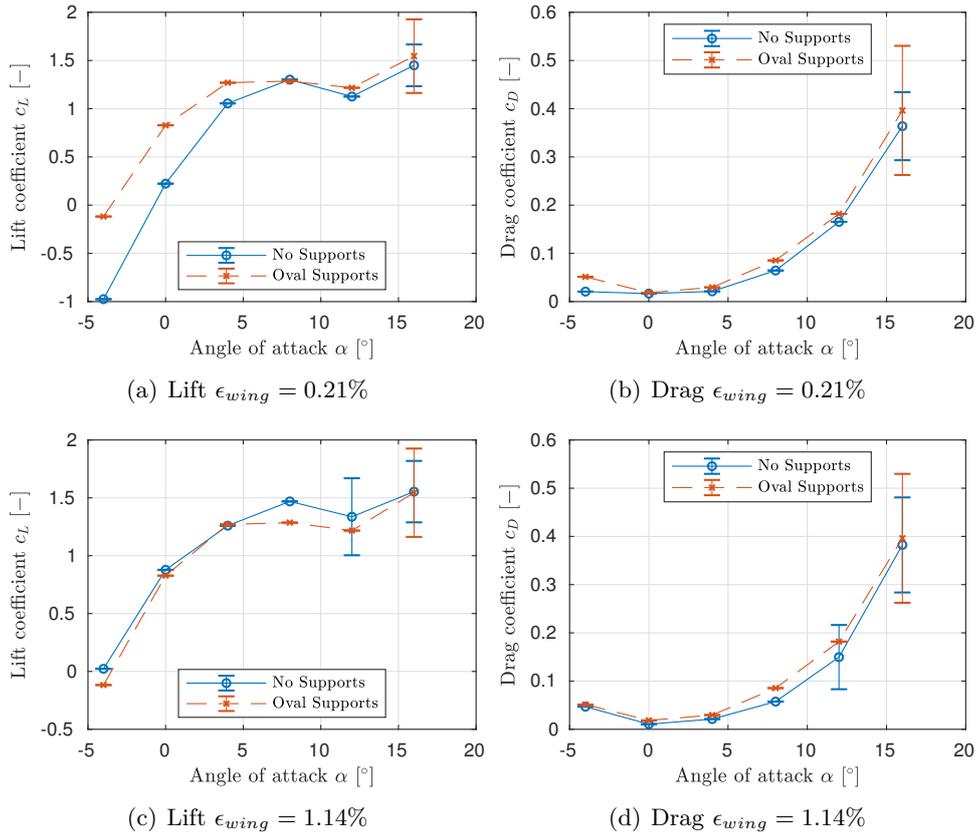


Figure 7-12: Lift and drag polar for the rigid wing without (—) supports and with oval supports (---) with the shape as obtained from the Vortex Particle Method

Flexible wing for negative angles of attack The experiment report describes a hysteresis in the wing shape near an angle of attack of zero degrees. This means that when slowly reducing the angle of attack below zero from a moderate, positive angle of attack, the wing will produce a positive lift beyond zero degrees inflow angle until at some point the membrane snaps through to the other side. The report also found the larger the excess length of the membrane, the larger the range of angles of attack at which hysteresis occurred. Moreover, an increased occurrence of limit cycle oscillations near zero degrees was also observed for configurations with large excess lengths.

Figure 7-13 shows two excerpts of the flow field for a negative angle of attack and an excess length of the membrane of roughly 1%. Because the initial, unloaded shape of the wing is

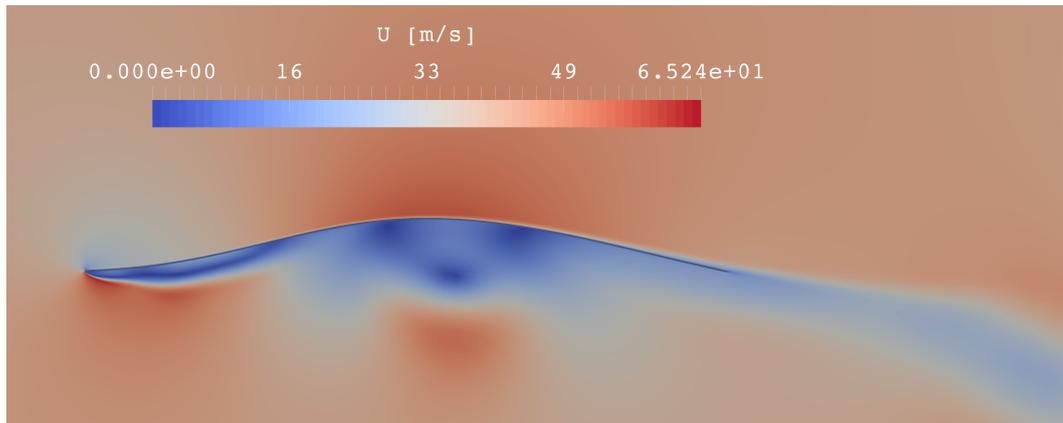
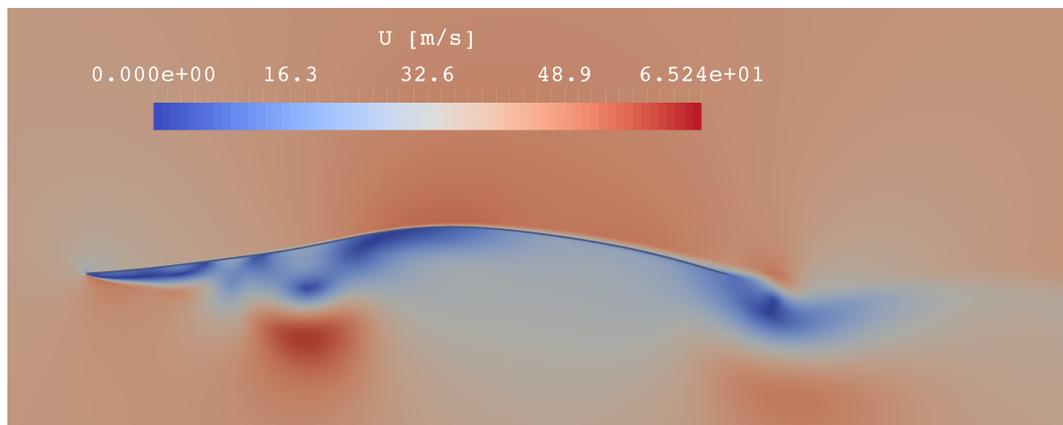
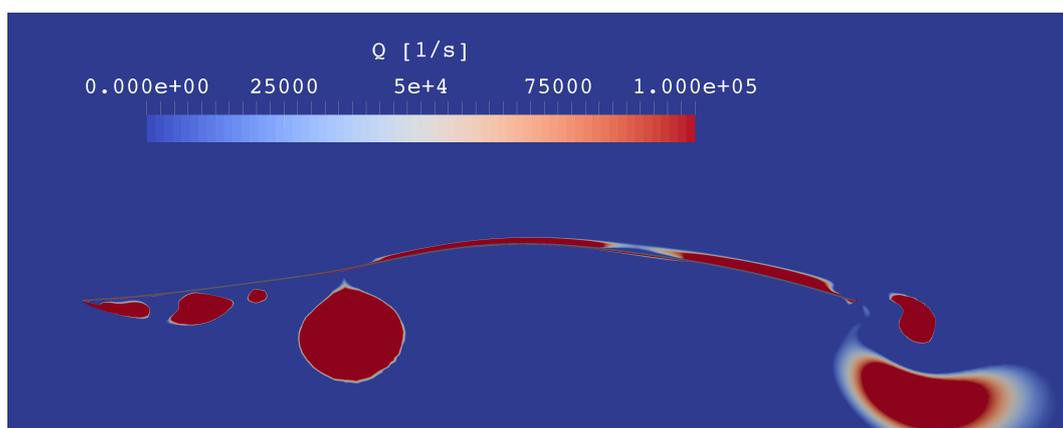
(a) Flow field at time $t = 0.62s$ (b) Flow field at time $t = 0.72s$ (c) Q-criterion at time $t = 0.72s$

Figure 7-13: Oscillatory motion of the wing for an angle of attack of $\alpha = -4^\circ$ and a relaxed excess length of $\epsilon_{wing} = 1.14\%$ without modeling the profile supports

chosen such that the maximum camber point is directed upwards, the time history of the flow ensures that this constellation remains stable instead of a snap through of the membrane to the

other side. In the first time instant in subfigure 7-13(a), a separation bubble on the pressure side is observed which then subsequently detaches from the wing surface and is convected with the freestream velocity as seen in subfigure 7-13(b). The vortex location is more clearly illustrated in subfigure 7-13(c) using the Q-criterion. It should be noted, however, that this method of vortex identification fails within the boundary layer of the wing and should only be considered further away from the surface. The profile surface moves further down as the vortex convects along the chord. The cycle then repeats itself as a new vortex is detached from the leading edge, resulting in a periodic oscillation of the wing displacement as well as the forces on the wing. Lastly, this limit cycle oscillation is not observed for the shorter excess length of close to 0.2%, where a convergence towards a steady deformed shape as well as flow field is witnessed. These tendencies are in accordance with the experimental findings from Greenhalgh.

Flexible wing at moderate angles of attack For moderate angles of attack where the reattachment of the leading edge separation bubble is in front of the mid-chord location, convergence to a steady deformed membrane shape is obtained. Figure 7-14 shows the converged deformed shape of the loaded membrane for two angles of attack, and both of the investigated support models.

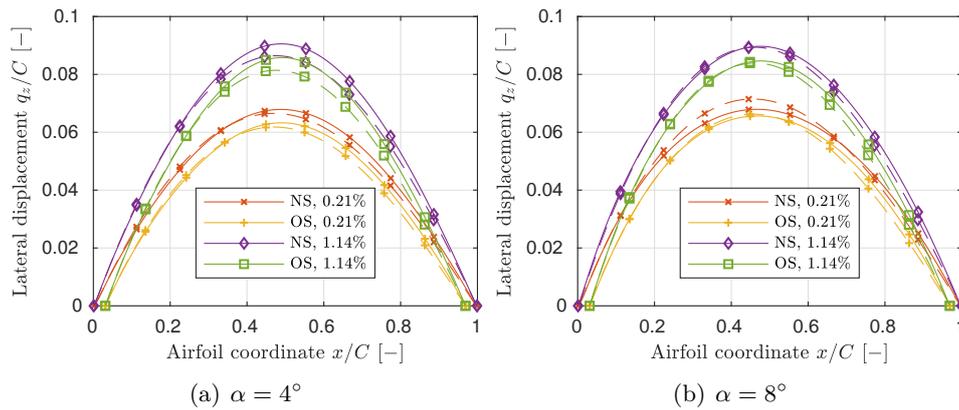


Figure 7-14: Loaded wing shape at moderate angles of attack for different initial excess lengths of the membrane with oval supports (OS) and without supports (NS) as obtained from the Vortex Particle Method (---) as well as from FOAM-FSI (—)

The figure shows both the results obtained with the PIMPLE algorithm as well as with the steady-state Vortex Particle Method when the supports are not modelled. The trends in the simulation results agree with the experiments. Namely, the maximum camber point moves towards the leading edge of the wing as the angle of attack is increased and the maximum camber decreases. Moreover, the Vortex Particle Method is also capable of capturing these tendencies. However, as the angle of attack is increased, the discrepancy between the inviscid and the viscous method is expected to grow as viscous effects become more important.

Flexible wing at large angles of attack For large angles of attack where the reattachment of the leading edge separation bubble occurs either aft of the mid-chord point or not at all, an increase in the angle of attack is associated with a drop of the slope of the lift curve. Moreover,

as previously noted even for a rigid wing large fluctuations in the aerodynamic forces are present which for a membrane wing also result in an unsteady deformation behaviour of the wing. In fact, the same unsteady stall behaviour as for a rigid wing is observed only further amplified by the low bending stiffness of the wing.

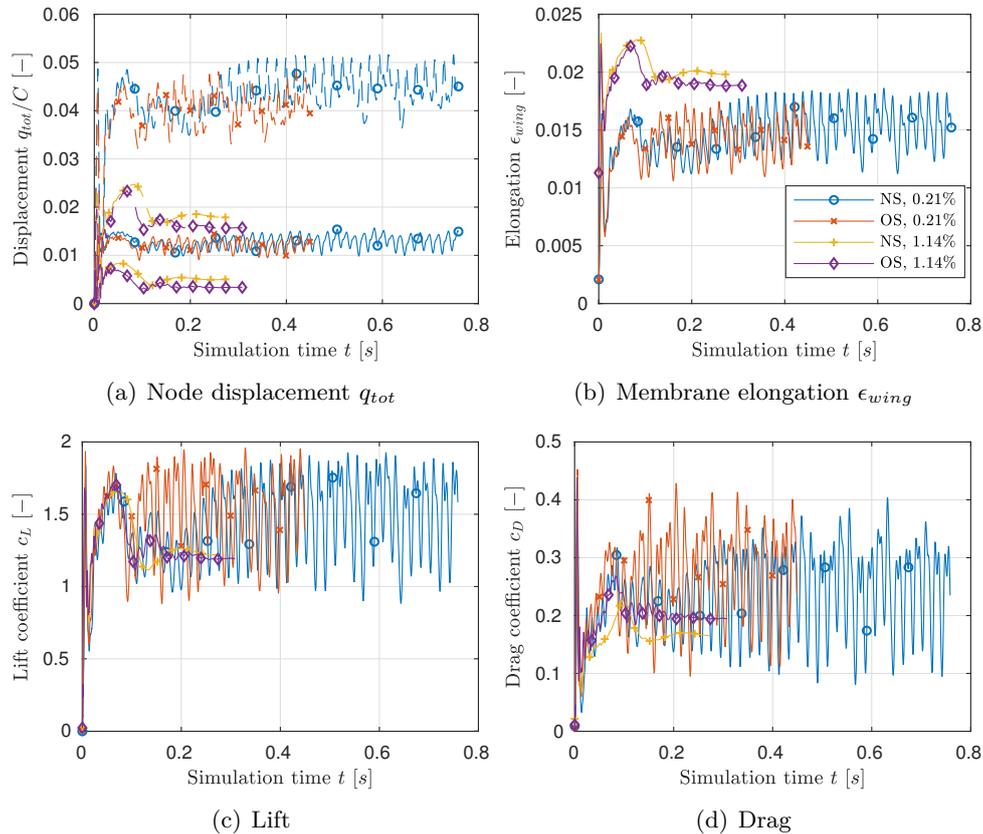


Figure 7-15: Unsteady coupled simulation results for $\alpha = 12^\circ$ and different initial excess lengths of the wing as well as with oval supports (OS) and without supports (NS), the total nodal displacements are shown at a $x/C = 0.2$ (—) and $x/C = 0.5$ (---) chord location

Figure 7-15 shows the transient of the integrated force coefficients, the wing excess length as well as the displacements of specific nodes for an angle of attack of $\alpha = 12^\circ$. Up to about $t = 0.08s$ the extension of the membrane wing from its relaxed initial state to the fully loaded state is visible. After this, a somewhat periodic vortex shedding can be observed in all the illustrated quantities. Namely, a leading edge separation bubble forms which continuously grows larger leading to a temporary increase in lift. Right before the separation bubble detaches from the wing surface, a local maximum in the lift is reached. To conserve vorticity, vortices of opposite sign are shed from the trailing edge in accordance with the shedding of a leading vortex. Simultaneously to the variation in the lift, the total membrane length increases and decreases. On close inspection, a shift between the two node locations is observed, where the one at the 20 % chord location is ahead of the one at the 50 % chord location in agreement with the growth of the leading edge separation bubble. Some examples of the flow field are shown in appendix D-1.

Polar Figure 7-16 summarises the previously described results via a lift and drag polar for the two different initial excess lengths alongside with the interpolated reference results. However, in particular, for the lowest and the largest investigated angles of attack where large fluctuations in the force coefficients were observed, the averaging period is too short to deliver fully statistically relevant predictions. The reference lift and drag coefficients were obtained via interpolation according to the deformed length of the membrane. It should also be noted that the drag values are not very accurate due to how they were documented in the experiment report.

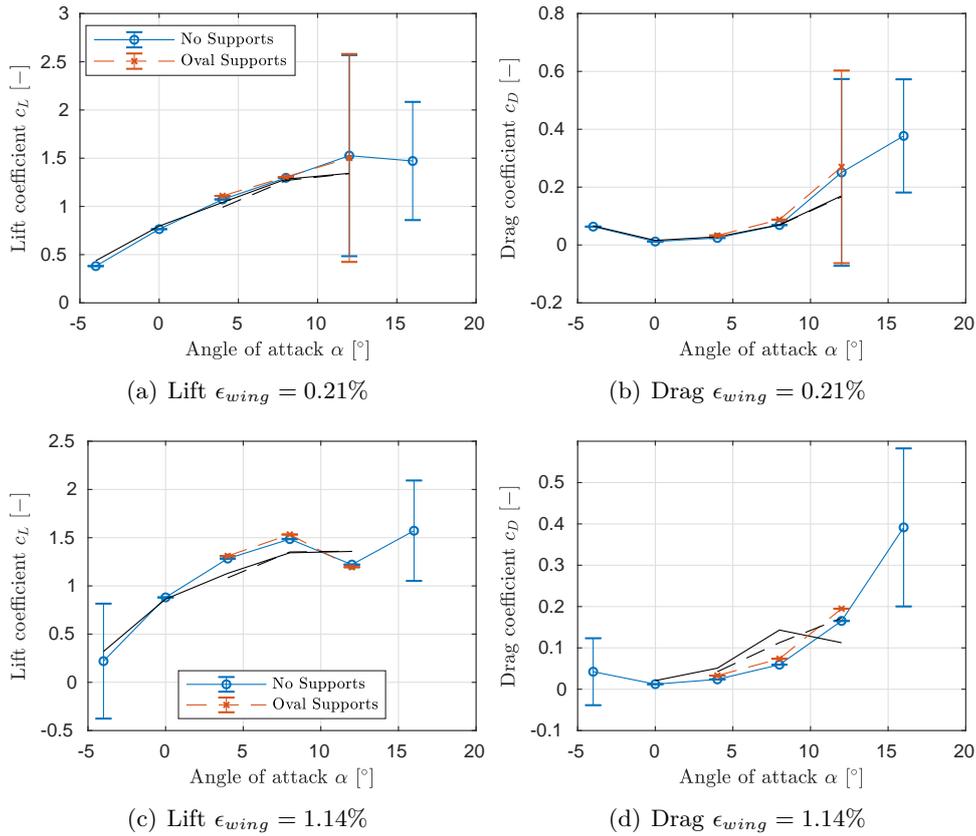


Figure 7-16: Lift and drag polar for the flexible wing with indication of the mean and extreme values of the force coefficient and the reference solution in black

For a relaxed excess length of the wing of roughly 0.21% very good agreement between experimental and numerical predictions is obtained for the lift and drag coefficients, whereas for an initial excess length of roughly 1.14% the deviation between the results increases. This highlights that there is a systematic increase in the modelling error as the excess length increases. Possibly, a larger excess length is tied with an increase in the coupled error from fluid and structure model. Furthermore, the type of support model does indeed have a non-negligible influence on the results with on average 5 % and 20 % relative deviation between lift and drag coefficients, respectively, for the two investigated support types. For the limited investigated range of angles of attack with oval supports, the drag predictions are about twice as accurate as when the supports were not modelled.

While the error in the prediction of the mean lift is still acceptable with at most roughly 15

%, the mean drag prediction for both excess lengths is up to 50 % off depending on the angle of attack and support model. This is most likely an accumulation of multiple error sources such as the mentioned modelling simplifications, insufficient averaging period, inaccuracy of the turbulence model, the assumption of fully turbulent flow as well as possibly also the large time step. In general, when CFD is used to predict lift and drag values, it is much easier to predict lift accurately than drag. Here, more investigation is necessary to see whether a more accurate prediction can be obtained. Moreover, it is unclear why an unsteady behaviour is observed for some configurations where the experiment report still describes steady results. This could be related to the support model or possibly also time-averaging in the experiment since the mean values still agree well with the mean values from the experiment report.

7-3 Den Boer setup

The single-membrane sailing profile shown in the literature review section in figure 2-6(a) was also investigated for one specific angle of attack and Reynolds number of 300,000 because here the deformed shape of the profile is available from den Boer [2]. The mesh of the setup is shown in figure 7-17.

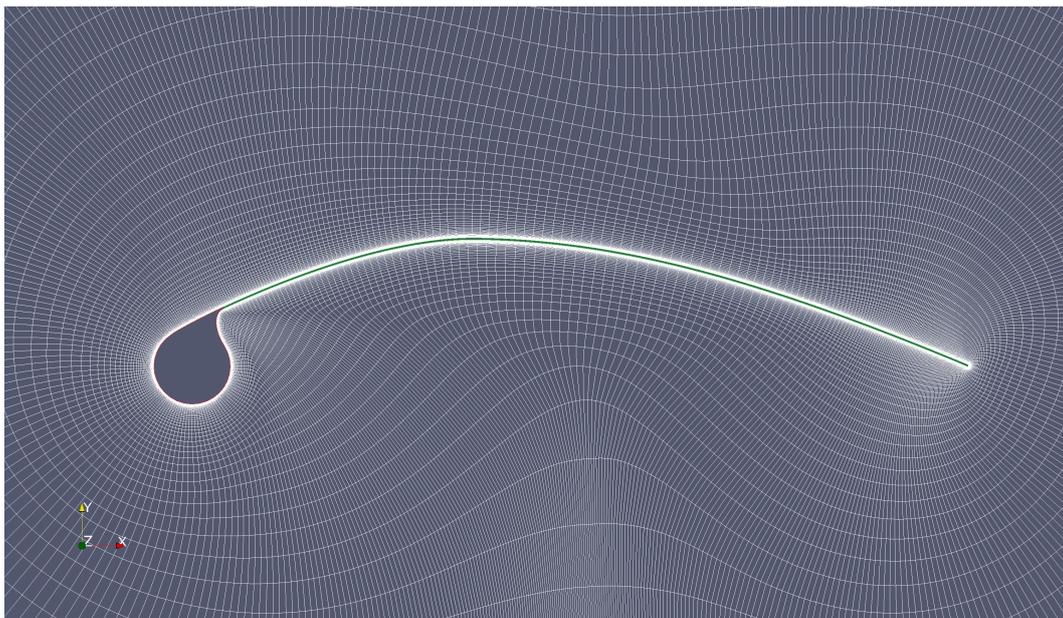


Figure 7-17: O-mesh topology for the den Boer case

The specific settings for the case are listed in table 7-6. For all other settings and modelling decisions consider the documentation of the previous test case. A mesh convergence study was also not performed since experience mesh convergence values from similar sailwings at similar Reynolds numbers were available [145].

Similar to the Greenhalgh setup where the small radius of the leading and trailing edge supports lead to unsteady results even for a rigid wing due to vortex shedding, for a single-membrane sail wing a separation bubble occurs behind the leading edge which can lead to inconsistent convergence of the steady-state fluid solver. Similarly, if a first-order dissipative

Case specific settings	
Slack angle δ	15°
Angles of attack α	7°
Structure properties	
Young's Modulus E_S	[1e5N/m ² , 5e5N/m ² , 1e6N/m ²]
Poisson ratio ν_S	0.3
Density $\bar{\rho}_S = \rho_S * h$	0.15kg/m ²
Damping coefficient ζ	5%
Fluid properties	
Kinematic viscosity ν_F	1.47e-5m ² /s
Density ρ_F	1.23kg/m ³
Reynolds number Re	2.95e5
Discretisation	
Number of structure elements $N_{cells,structure}$	175
Number of modes considered for structural damping $N_{damping}$	100
Size of fluid mesh $N_{cells,fluid}$	63872
Time step Δt	1e-4s

Table 7-6: Configuration den Boer setup

scheme is used for the discretisation of the convective terms instead of a less dissipative second order scheme, a better convergence of the results is obtained. However, this significantly impacts the accuracy of the results. Thus, for this configuration, the SIMPLE solver in foam-extend did not yield fully converged results despite steady-state reference results. This of course also holds true for steady-state results of the aeroelastic solver in FOAM-FSI. Thus, here the same methodology as already for the Greenhalgh case is applied; namely, the steady solver is used for initialisation of the coupled solver, and then the unsteady solver is used in combination with a large time step that smoothes out temporal oscillations such as vortex shedding occurring at a higher frequency than the simulation resolves.

Rigid wing Since for this particular constellation of the test case the deformed shape of the wing is given, first a simulation with a rigid profile according to the loaded shape is performed to get an estimate of the error introduced through the fluid model alone. However, the evaluation of the accuracy of the fluid model is restricted to the lift and drag values. Table 7-7 lists the reference and the newly obtained numerical results.

	Experiment	foam-extend	OF-plus [145]	OF-plus with TM [145]
Lift coefficient c_L [-]	1.255	1.229	1.123	0.755
Drag coefficient c_D [-]	0.055	0.058	0.071	0.108

Table 7-7: Force coefficients for the rigid wing as obtained from foam-extend, the reference experiment report and another reference using OpenFOAM-plus with and without a transition model

Since the fluid solver is fully turbulent and the reference experiments are set at a Reynolds number where laminar flow onset is expected, a disagreement between the force coefficients

from experiment and numerical simulation is expected. The results from foam-extend over-predict lift and drag as compared to the experiments. However, the deviation between the force coefficients is small with roughly 2% and 5% relative error for lift and drag, respectively. Hence, it seems that despite the modelling assumptions for this particular configuration foam-extend still delivers reasonable predictions of the aerodynamic properties of the profile. It should be noted though that to make a definite statement different angles of attack would have to be investigated.

Flexible wing Since the Young's Modulus of the material that is used for the membrane part of the sailing is not specified in the report from den Boer, the fully coupled simulations were run for a range of different parameters. This way no quantitative validation can be carried out, but the trends can still be checked. All coupled simulations using the PIMPLE algorithm indicated a clear convergence towards steady-state results. However, due to the time constraints of the project, full convergence of the simulations was not obtained. Nevertheless, the close to converged loaded wing shape for different Young Moduli are shown in figure 7-18.

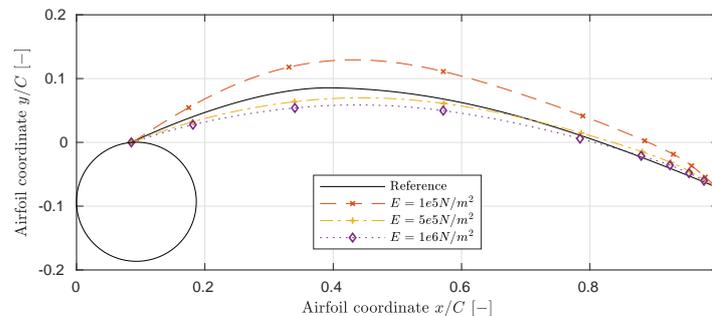


Figure 7-18: Comparison of the deformed shape of the profile between the reference experimental results and the numerical results from FOAM-FSI for different Young Moduli

A Young's Modulus of about $E \approx 5e5 N/m^2$ seems to correspond best with the reference results. But then again as illustrated before, since even for a rigid wing the deviation between the reference results and the fluid model are not negligible, no definite statements can be made. Hence, the capability of the methodology was demonstrated, but due to an insufficient experiment description and time constraints, quantitative validation of the method remains incomplete.

Conclusions and Recommendations

The CFD-CSM modelling of membrane wings at high Reynolds numbers is very challenging for two different reasons. First, the low density and bending stiffness of the wing material result in a strongly coupled FSI problem that requires multiple coupling iterations per time step. In combination with the meshing and time step constraints of the CFD model for high Reynolds numbers, this results in a computationally expensive problem. Second, as the literature survey revealed, only limited public reference data is available for the validation of new methodologies.

Nevertheless, a methodology to model membrane wings at high Reynolds numbers was developed and partially validated within the context of this project. Section 8-1 will summarise and conclude on what has been done during this project. Then section 8-2 will have a more critical look at what improvements are recommended for the existing setup and what measures would improve the confidence in the modelling outcomes.

8-1 Conclusions

Within the context of this project, a Fluid-Structure Interaction (FSI) solver for the aero-elastic modelling of membrane wings at high Reynolds numbers was developed and validated on selected test cases. This was done by essentially merging and improving on two previous aero-elastic solver implementations. First, Blom [132] and Gillebart [133] created an adapter for the open-source CFD solver foam-extend-3.2 [129] that allows communication to the coupling library preCICE [130] forming the FOAM-FSI toolbox [131]. Second, a low-fidelity aero-elastic solver developed with the aim to model a kite was implemented by Bosch [13] and Berens [15] based on the nonlinear shell element formulation derived by Tiso [31] in combination with inviscid and parametric aerodynamic models. The resulting methodology of this project comprehends the coupling of the structural solver framework from Berens and FOAM-FSI's RANS fluid models.

On the side of the fluid solver, no modifications were made since the framework in FOAM-FSI has been thoroughly validated already. Conversely, a multitude of adaptations have been

made to the structural solver framework by Berens. Namely, the structural solver was made more stable and efficient through the implementation of a more conventional time integration scheme for the coupled solver instead of a staggered linear-nonlinear coupling loop as well as the reimplementaion in a more efficient programming language. Moreover, a more thorough validation of the methodology was also carried out with the aim to rule out any implementation faults. The limitations of the shell element formulation were also outlined through application to two benchmark cases highlighting that the model should only be applied to very thin shells. This is indeed also the target application range for which it was developed [31].

Since the structural mesh does not reflect the thickness of the element due to the assumptions made for the shell element formulation, an interpolation method for the communication between the fluid and structure model that respects the thickness of the structure was implemented. The developed methodology involved the creation of an extrusion mesh from the current (deformed) structure mesh for every time step. For the imposition of the solid displacements, node location and velocity were extrapolated from the structure to the extrusion mesh. For the transmission of the fluid interface forces, the nodal forces had to be reduced from the extrusion to the structure mesh. Based on this extrusion mesh, standard interface interpolation methods developed for FSI applications can be used as was done in preCICE. The methodology was written such that it will also work for a zero-thickness element if the fluid solver of choice has this capability. Moreover, the solver was constructed to allow the coupling to the Immersed Boundary (IB) fluid solver Fluidity. A particular IB method implemented in this solver was also briefly considered within this project, however, eventually, the idea was discarded because the solver framework was not mature enough for this application.

Below the research objective is restated highlighting that while the implementation part of the objective was carefully adhered by, the methodology still needed to be validated to have confidence in its application.

Further improve on the existing FSI solver by maintaining the existing structural solver methodology, coupling it to a higher fidelity CFD fluid solver, reviewing the time integration methodology of the structural solver and applying the new framework to a simplified test case. Construct the solver such that it can be used as a baseline for future projects in the AWE group; thus it should be easily extendable to more complicated test cases, and all its components are thoroughly validated.

The subsequent application of the coupled solver on the classical rigid beam-flexible flag FSI benchmark case from Turek & Hron [1, 66] provided encouraging results regarding the accuracy of the method. Specifically, for the integrated force coefficients as well as the beam tip displacements the relative deviation from the reference results in mean and amplitude was at worst 10 % except for values with very small mean or amplitude values. Moreover, a clear trend with mesh refinement was distinguished. Two probable error sources can be identified. First, the application of the thin shell element formulation to a comparatively thick beam violated the element formulation modelling assumptions. This was already visible in the comparison with the separate validation of the structural solver on the same geometry. Second, for the pure validation of the fluid solver on the setup for the target Reynolds number of the coupled case on a comparatively large mesh of about 50,000 elements, the solution was not fully converged yet. In conclusion, the coupling of those two error sources resulted in the non-negligible total error.

Following this, the solver was applied to a two-dimensional test case with a Reynolds number of above one million. In this setup, a thin, flexible material was wrapped around small rigid leading and trailing edge supports. In this reference experiment, the force on the wing was measured for different membrane slack and angle of attack configurations. The experimental reference results were one of the few publicly available data sets with a sufficiently high Reynolds number to obtain sensitive results if the fluid solver is fully turbulent [3, 84, 85]. Comparison of the results from numerical models and experiments showed that the general trends for different slack lengths of the wing and angle of attack sensitivity were well captured in the numerical outcomes. However, the qualitative validation based on the two examined excess lengths was less promising. While the relative error in the mean lift values was at most 15 %, the mean drag results were less favourable with deviations of up to 50 %. Moreover, in some of the cases, the results were periodically unsteady, even though in the reference experiments steady-state behaviour was observed. Potential error sources are the simplifications made in the modelling process, limitations of the fluid and structure model itself as well as limitations in the experiment description and execution.

Finally, the methodology was also tested on a two-dimensional sailing configuration with membrane slack at a Reynolds number of about one-third of a million [2]. The sail material of the profile was wrapped around a rigid circular leading edge with a radius of about 5 % of the chord length and a small rotatable plate at the trailing edge. Similar to the previous test case, the sail material properties were not fully specified. However, for some of the constellations the deformed shape of the wing was given, such that the error of the fluid solver alone could be quantified for one specific moderate inflow angle to be about 2 % and 5 % for the lift and drag, respectively. Thus, highlighting that the accuracy of the fluid model for single membrane sail wing constellations that often show a stationary separation bubble on the pressure side as well as trailing edge separation at higher angles of attack is already not negligible. This error is then further amplified when coupling the fluid with the structural solver. Of course, this also depends on the Reynolds number since for this specific constellation laminar flow onset is expected in the experiments and the fluid model is fully turbulent.

Concluding, this project developed a methodology to model membrane wings at high Reynolds numbers. While the capabilities of the method have been successfully showcased on a classic FSI benchmark case, only a partial validation on benchmark cases at more realistic Reynolds numbers has been carried out due to limitations in time, computational resources as well as the availability of reliable experimental reference data.

8-2 Recommendations

The recommendations for the future development of the solver will be split into different categories according to the different submodules of the full solver.

First, for the structural solver, the efficiency and the stability could be increased by implementing a more sophisticated solving procedure for the nonlinear equation that arises due to the implicit time integration scheme and the nonlinear shell element formulation. However, as compared to a fluid model solving the full Navier-Stokes equations, the computational effort spent on the structural solver is negligible. Nevertheless, if the solver is coupled with lower fidelity aerodynamic models, the situation might reverse. Moreover, since the structural

solver is written in a Total Lagrangian framework, in case of large element rotations the implementation of follower loads for displacement dependant nonconservative loads will lead to better accuracy of the structural model [108]. Additionally, implementing moving boundary conditions in the solver could also help reduce the extent of the modelling simplifications for the high Reynolds number test case and make the solver more universally applicable.

Second, some adjustments to the interface interpolation method are also expected to make the interface communication more conservative with respect to the work transmitted over the interface. Consistent Radial Basis Function interpolation was used to transfer the stress from the fluid mesh to the structure extrusion mesh and to transfer the displacement of the extrusion mesh to the fluid mesh. Comparison between the internal interface force integration in foam-extend and the integration in the structural solver showed deviations of up to 5 % in the total force integration. While the force integration was checked for the Turek & Hron benchmark case, a close agreement was present. However, retrospectively analysing the results of the cases using a membrane wing, showed larger discrepancies, because there larger ratios between the length of the structure and fluid elements are present. Thus, to conserve the force over the interface as much as possible, a higher order force integration method should be implemented on the extrusion mesh. Currently, a first-order accurate method is used, that fails to capture large gradients in the stresses on the fluid side. Defining intermediate points along a structure mesh element and using a higher order integration method is expected to improve the accuracy of the interface force integration.

Third, the absence of transition models makes the validation of the aero-elastic solver more difficult, since the already very limited availability of benchmark data for membrane wings is mostly restricted to Reynolds numbers below one million. While the fully turbulent solver can be applied to these test cases, depending on the configuration a large penalty is already present in the fluid model which then only further amplifies the error of the coupled simulation. Thus, using OpenFOAM instead of foam-extend if an adapter to preCICE is available, would significantly simplify the validation of the solver, because OpenFOAM has transition models.

Lastly, for the modelling of a kite in Airborne Wind Energy applications during a full load cycle, in the author's opinion, it would make more sense to investigate an Integral Boundary Layer Approach with corrections for cross flow and tip effects [146, 147]. Because doing fully three-dimensional simulations of a strongly coupled FSI problem over extended periods of time is very computationally expensive and possibly only feasible for academic purposes or detail analysis.

Appendix A

Van Karman strain equations

In classical plate theory the displacement field is chosen such that it satisfies the Kirchhoff hypothesis, namely the following constraints have to be fulfilled [148]

- Straight lines remain perpendicular to the mid-surface (i.e. transverse normals) before deformation, remain straight after deformation.
- The transverse normals do not experience elongation.
- The transverse normals rotate such that they remain perpendicular to the mid-surface after deformation.

Additionally carrying out an order of magnitude analysis, one can simplify the full nonlinear strain formulations to the van Karman strains.

The governing equations for the normal displacement w and the stress function Φ are then given by equation A-1 [149]

$$\frac{\partial^4 \Phi}{\partial x^4} + \frac{\partial^4 \Phi}{\partial x^2 \partial y^2} + \frac{\partial^4 \Phi}{\partial y^4} = E \left[\left(\frac{\partial^2 w}{\partial x \partial y} \right)^2 - \frac{\partial^2 w}{\partial x^2} \frac{\partial^2 w}{\partial y^2} \right] \quad (\text{A-1a})$$

$$\frac{\partial^4 w}{\partial x^4} + \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = \frac{p_z}{D} + \frac{h}{D} \left(\frac{\partial^2 \Phi}{\partial y^2} \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 \Phi}{\partial x^2} \frac{\partial^2 w}{\partial y^2} - 2 \frac{\partial^2 \Phi}{\partial x \partial y} \frac{\partial^2 w}{\partial x \partial y} \right) \quad (\text{A-1b})$$

where E is the Young's Modulus of the material, D is the flexural rigidity given by equation A-2, p_z is the normal pressure and h is the plate thickness.

$$D = \frac{Eh^3}{12(1 - \nu^2)} \quad (\text{A-2})$$

Through the introduction of the stress function Φ , the stresses at the midplane of the plate are then given by

$$\sigma_x = \frac{\partial^2 \Phi}{\partial y^2} \quad (\text{A-3a})$$

$$\sigma_y = \frac{\partial^2 \Phi}{\partial x^2} \quad (\text{A-3b})$$

$$\tau_{xy} = -\frac{\partial^2 \Phi}{\partial x \partial y} \quad (\text{A-3c})$$

and the extreme stresses are given by

$$\sigma_x = -\frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w}{\partial x^2} + \nu \frac{\partial^2 w}{\partial y^2} \right) \quad (\text{A-4a})$$

$$\sigma_y = -\frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w}{\partial y^2} + \nu \frac{\partial^2 w}{\partial x^2} \right) \quad (\text{A-4b})$$

$$(\text{A-4c})$$

$$\tau_{xy} = -\frac{Eh}{2(1+\nu)} \left(\frac{\partial^2 w}{\partial x \partial y} \right) \quad (\text{A-4d})$$

$$(\text{A-4e})$$

Lastly, the van Karman strains are then given by

$$\epsilon_x = \frac{1}{E} \left(\frac{\partial^2 \Phi}{\partial y^2} - \nu \frac{\partial^2 \Phi}{\partial x^2} \right) \quad (\text{A-5a})$$

$$\epsilon_y = \frac{1}{E} \left(\frac{\partial^2 \Phi}{\partial x^2} - \nu \frac{\partial^2 \Phi}{\partial y^2} \right) \quad (\text{A-5b})$$

$$\gamma_{xy} = \frac{2(1+\nu)}{E} \left(\frac{\partial^2 \Phi}{\partial x \partial y} \right) \quad (\text{A-5c})$$

$$(\text{A-5d})$$

Appendix B

Vortex Particle Method

B-1 Theory and implementation

Classical inviscid flow modeling assumes that the flow around a thin structure can be modeled as an unsteady, inviscid, incompressible and irrotational flow. Thus, the flow can be characterised as a time-variant potential flow. This assumption allows for a straight forward discretisation of the flow in terms of singularity elements such as vortices, sources and sinks. In the particular case of a Vortex Particle Method, the outline of the idealised zero thickness structure is modeled using discrete vortex elements. As such on each panel of the discretised structure a vortex point is placed at the quarter chord point of the panel and the non-permeability condition is enforced at the collocation point chosen to coincide with the three quarter chord point of the panel. This particular choice of the collocation point ensures that the Kutta condition is fulfilled. The induced velocity of a vortex element Γ_j located at (x_j, y_j) at an arbitrary location (x, y) is given by

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{\Gamma_j}{2\pi r_j^2} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x - x_j \\ y - y_j \end{bmatrix} \quad (\text{B-1})$$

where $r_j = \sqrt{(x - x_j)^2 + (z - z_j)^2}$ is the distance between induction and source location.

It should be noted that since the permeability condition is only enforced at the collocation point of each panel if an insufficient amount of panels is used there may still be flow through the surface of the structure in between collocation points.

With the instantaneous position of the bound Γ_j and the free $\Gamma_{W,j}$ vortex particles as well as external inflow conditions known, the impermeability boundary condition can be specified

for each collocation point. This leads to a linear system of equations as follows

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1N} & a_{1W} \\ a_{21} & a_{22} & \dots & a_{2N} & a_{2W} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{N1} & a_{N2} & \dots & a_{NN} & a_{NW} \\ 1 & 1 & \dots & 1 & 1 \end{bmatrix} \begin{Bmatrix} \Gamma_1 \\ \Gamma_2 \\ \vdots \\ \Gamma_N \\ \Gamma_{W1} \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \\ \Gamma(t - \Delta t) \end{Bmatrix} \quad (\text{B-2a})$$

$$b_i = -[U(t) + u_W + u_{Def}, W(t) + w_W + w_{Def}]_i \cdot \vec{n}_i \quad (\text{B-2b})$$

The induced velocity of the bound vortex particles is captured through the influence coefficients a_{ij} whereas the influence of the external inflow (U, W), the wake (u_w, w_W) and the movement of the structure (u_{Def}, w_{Def}) is captured in the terms b_i on the right hands side. Finally, the last row refers to the enforcement of the Kelvin condition.

The convection of the wake particles is done at the beginning of each time step with the velocities from the previous time step; no sub-iterations are performed. The location of the first shed vortex is 25 % of the path covered by the trailing edge within on time step.

Solving the system of equations given in B-2 yields the strength of the bound vortices and the induced velocity at the location on the structure surface. From this the aerodynamic loads can be calculated through the pressure differences Δp_j across the structure according to

$$\Delta p_j = \rho \left([U(t) + u_W + u_{Def}, W(t) + w_W + w_{Def}]_j \cdot \vec{\tau}_j \frac{\Gamma_j}{\Delta l_j} + \frac{\partial}{\partial t} \sum_{k=1}^j \Gamma_k \right) \quad (\text{B-3})$$

where $\vec{\tau}_j$ refers to the tangential surface vector. The time derivative of the velocity is obtained using a first order forward Euler scheme.

The linear system of equations is reset up for every time step after the wake nodes have been convected with the information from the previous time step.

B-2 Validation and convergence

The steady state verification of the Vortex Particle Method is done for a flat plate through comparison with results from classical thin airfoil theory on symmetric airfoils, in this case, a flat plate. As such one would expect a lift slope of $C_L = 2\pi\alpha$. Figure B-1 shows the results for different panel resolutions in terms of lift B-1(a) and vorticity distribution B-1(b). It should be noted that the overall lift for the flat plate is independent of the number of panels since it is fixed by the Kutta condition. For the membrane wing, the panel nodes are chosen to coincide with the nodes of the structure mesh. The lift of the membrane wing shows a weak dependence on the resolution due to the better resolution of the deformed structure. As expected the membrane wing shows higher lift values due to the more cambered deformed shape. From the convergence of the local loading, it can be concluded that 40 panels are sufficient, a higher resolution mainly leads to an increase in the suction peak.

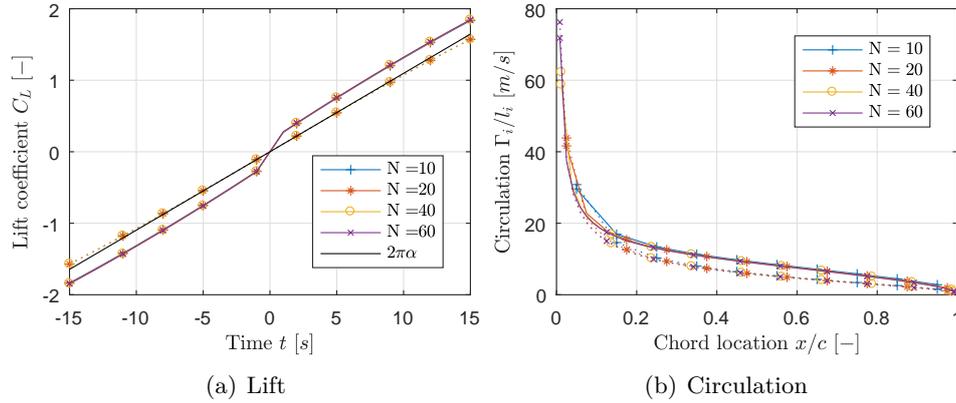


Figure B-1: Verification Steady Vortex Particle Method for a flat plate (·) and the membrane wing (-)

The unsteady verification of the Vortex Particle Method can be done through comparison with the indicial plunge response of a flat plate airfoil as analytically determined by Wanger for a flat plate in incompressible, potential flow.

$$L = 2\pi\rho_\infty U_\infty^2 \frac{c}{2} \alpha_0 \phi \left(\frac{U_\infty t}{c/2} \right) \quad (\text{B-4a})$$

$$\phi \left(t^* = \frac{U_\infty t}{c/2} \right) = 1 - 0.615 \exp^{-0.0455t^*} - 0.355 \exp^{-0.3t^*} \quad (\text{B-4b})$$

The results of the implemented Vortex Particle method for a step increase in the angle of attack is shown in figure B-2 alongside with the analytic solution from Wagner [150]. Figure B-2(a) investigates the effect of the panel density on the lift of the flat plate and the membrane wing for an inflow angle of eight degrees. Again only the membrane wing shows a weak dependency on the panel density because for the flat plate the overall lift is fixed by the Kutta condition. Comparison with the Wagner function shows that qualitatively the behaviour matches, even though quantitatively the lift slope is too steep. The peak for the first step stems from the discretisation of the time derivative of the circulation in equation B-3. For the coupled simulations of the membrane wing, after every stepwise change in the angle of attack the fluid model is advanced for two very small time steps without feedback from the structure to make the simulation stable. Afterwards, the time step is increased again, and the feedback from the structure is considered. The time step sensitivity of the lift transient is checked in figure B-2(b), the difference between the three investigated time steps is minimal and only visible for the first few time steps. Thus for the rest of this project unless otherwise noted a panel density of $N = 40$ and a time step of $\Delta t = 0.025$ will be used.

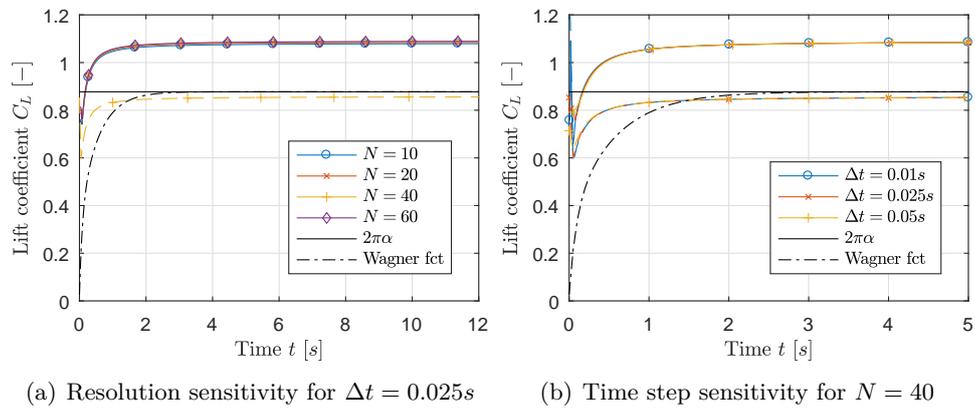


Figure B-2: Verification Unsteady Vortex Particle Method for a flat plate and the membrane wing at $\alpha = 8^\circ$

Appendix C

Solver settings

C-1 OpenFOAM settings

C-1-1 Laminar cases

Discretisation

```
1 FoamFile
2 {
3   version 2.0;
4   format ascii;
5   root "";
6   case "";
7   instance "";
8   local "";
9   class dictionary;
10  object fvSchemes;
11 }
12
13 ddtSchemes
14 {
15   default bdf2;
16 }
17
18 gradSchemes
19 {
20   default Gauss linear ;
21 }
22
23 divSchemes
24 {
25   default none;
26   div(phi,U) Gauss skewCorrected linear ;
27   div((nuEff*dev(grad(U).T())) Gauss linear ;
28   div(U) Gauss linear;
29 }
30
31 laplacianSchemes
32 {
33   default Gauss linear corrected ;
34 }
35
36 interpolationSchemes
37 {
38   default skewCorrected linear ;
```

```

39 }
40
41 snGradSchemes
42 {
43     default corrected;
44 }
45
46 fluxRequired
47 {
48     default no;
49     p /* empty */ ;
50 }

```

Solver

```

1 FoamFile
2 {
3     version 2.0;
4     format ascii;
5     root "";
6     case "";
7     instance "";
8     local "";
9     class dictionary;
10    object fvSolution;
11 }
12
13 solvers
14 {
15
16     p
17     {
18         solver          PCG;
19         preconditioner  DIC;
20         tolerance       1e-14;
21         relTol          1.0e-3;
22     }
23
24     U
25     {
26         solver BiCGStab;
27         preconditioner DILU;
28         tolerance 1e-14;
29     }
30 }
31
32 PIMPLE
33 {
34     nCorrectors 5;
35     nNonOrthogonalCorrectors 0;
36     tolerance 1.0e-13;
37     relTol 1.0e-2;
38     maxIter 15;
39     minIter 1;
40     pisoTol 1e-3;
41 }
42
43 relaxationFactors
44 {
45     U 0.9;
46 }

```

C-1-2 Steady RANS cases

Discretisation

```

1 ddtSchemes
2 {
3     default          steadyState;
4 }
5
6 gradSchemes

```

```

7 {
8     default      Gauss linear;
9 }
10
11 divSchemes
12 {
13     default      none;
14     div(phi,U)   Gauss linearUpwind Gauss linear;
15     div(phi,k)   Gauss upwind;
16     div(phi,omega) Gauss upwind;
17     div((nuEff*dev(grad(U).T()))) Gauss linear;
18 }
19
20 laplacianSchemes
21 {
22     default      Gauss linear corrected;
23 }
24
25 interpolationSchemes
26 {
27     default      linear;
28 }
29
30 snGradSchemes
31 {
32     default      corrected;
33 }
34
35 fluxRequired
36 {
37     default      no;
38     p            ;
39 }

```

Solver

```

1 solvers
2 {
3     p
4     {
5         solver      PCG;
6         tolerance   1e-8;
7         relTol      0;
8         preconditioner DIC;
9         maxIter     500;
10    }
11
12    "(U|k|omega)"
13    {
14        solver      BiCGStab;
15        preconditioner DILU;
16        nSweeps     1;
17        tolerance   1e-9;
18        relTol      0;
19        minIter     1;
20    }
21 }
22
23 PIMPLE
24 {
25     nOuterCorrectors 200;
26     nCorrectors 20;
27     nNonOrthogonalCorrectors 0;
28     tolerance 5.5e-5;
29     relTol 1e-3;
30     minIter 1;
31     maxIter 200;
32     pisoTol 1e-3;
33 }
34
35 relaxationFactors
36 {
37     p 0.7;
38     U 0.7;
39     k 0.7;

```

```

40     omega 0.7;
41 }

```

C-1-3 Unsteady RANS cases

Boundary condition

```

1  FoamFile
2  {
3      version      2.0;
4      format       ascii;
5      class        volVectorField;
6      location     "0";
7      object       U;
8  }
9  #include         "initialConditions"
10 dimensions      [0 1 -1 0 0 0 0];
11 internalField   uniform $velocity;
12 boundaryField
13 {
14     interface
15     {
16         type      fixedValue; //myMovingWallVelocity;
17         value     uniform (0 0 0);
18     }
19     supports
20     {
21         type      fixedValue;
22         value     uniform (0 0 0);
23     }
24     farfield
25     {
26         type      inletOutlet;
27         inletValue  uniform $velocity;
28     }
29     sides
30     {
31         type      empty;
32     }
33 }
34 // * * * * * //
35 FoamFile
36 {
37     version      2.0;
38     format       ascii;
39     class        volScalarField;
40     object       p;
41 }
42 #include         "initialConditions"
43 dimensions      [0 2 -2 0 0 0 0];
44 internalField   uniform 0;
45 boundaryField
46 {
47     interface
48     {
49         type      zeroGradient;
50     }
51     supports
52     {
53         type      zeroGradient;
54     }
55     farfield
56     {
57         type      outletInlet;
58         outletValue  uniform 0;
59     }
60     sides
61     {
62         type      empty;
63     }
64 }
65 }
66 // * * * * * //

```

```

67 FoamFile
68 {
69     version      2.0;
70     format       ascii;
71     class        volScalarField;
72     location     "0";
73     object       k;
74 }
75 #include        "initialConditions"
76 dimensions     [0 2 -2 0 0 0];
77 internalField  uniform $k;
78 boundaryField
79 {
80     interface
81     {
82         type      kqRWallFunction;
83         value     uniform 0;
84     }
85     supports
86     {
87         type      kqRWallFunction;
88         value     uniform 0;
89     }
90     farfield
91     {
92         type      inletOutlet;
93         inletValue  uniform $k;
94     }
95     sides
96     {
97         type      empty;
98     }
99 }
100 // * * * * *
101 FoamFile
102 {
103     version      2.0;
104     format       ascii;
105     class        volScalarField;
106     location     "0";
107     object       omega;
108 }
109 #include        "initialConditions"
110 dimensions     [0 0 -1 0 0 0];
111 internalField  uniform $omega;
112 boundaryField
113 {
114     interface
115     {
116         type      omegaWallFunction;
117         value     uniform 1e8;
118     }
119     supports
120     {
121         type      omegaWallFunction;
122         value     uniform 1e8;
123     }
124     farfield
125     {
126         type      inletOutlet;
127         inletValue  uniform $omega;
128     }
129     sides
130     {
131         type      empty;
132     }
133 }
134 // * * * * *
135 FoamFile
136 {
137     version      2.0;
138     format       ascii;
139     class        volScalarField;
140     location     "0";
141     object       omega;
142 }

```

```

143 #include      "initialConditions"
144 dimensions   [0 0 -1 0 0 0 0];
145 internalField uniform $omega;
146 boundaryField
147 {
148     interface
149     {
150         type      omegaWallFunction;
151         value     uniform 1e8;
152     }
153     supports
154     {
155         type      omegaWallFunction;
156         value     uniform 1e8;
157     }
158     farfield
159     {
160         type      inletOutlet;
161         inletValue uniform $omega;
162     }
163     sides
164     {
165         type      empty;
166     }
167 }
168 // * * * * *
169 FoamFile
170 {
171     version      2.0;
172     format       ascii;
173     class        volScalarField;
174     location     "0";
175     object       nut;
176 }
177 #include      "initialConditions"
178 dimensions   [0 2 -1 0 0 0 0];
179 internalField uniform $nut;
180 boundaryField
181 {
182     interface
183     {
184         type      nutWallFunction;
185         value     uniform 0;
186     }
187     supports
188     {
189         type      nutWallFunction;
190         value     uniform 0;
191     }
192     farfield
193     {
194         type      calculated;
195     }
196     sides
197     {
198         type      empty;
199     }
200 }

```

Discretisation

```

1  FoamFile
2  {
3      version      2.0;
4      format       ascii;
5      class        dictionary;
6      location     "system";
7      object       fvSchemes;
8  }
9
10 ddtSchemes
11 {
12     default      bdf2;
13     ddt(k)       boundedBackward;
14     ddt(omega)   boundedBackward;

```

```

15 }
16
17 gradSchemes
18 {
19     default          Gauss linear;
20 }
21
22 divSchemes
23 {
24     default          none;
25     div(phi,U)      Gauss linearUpwind Gauss linear;
26     div(phi,k)      Gauss upwind;
27     div(phi,omega)  Gauss upwind;
28     div((nuEff*dev(grad(U).T()))) Gauss linear;
29 }
30
31 laplacianSchemes
32 {
33     default          Gauss linear corrected;
34 }
35
36 interpolationSchemes
37 {
38     default          linear;
39 }
40
41 snGradSchemes
42 {
43     default          corrected;
44 }
45
46 fluxRequired
47 {
48     default          no;
49     p                ;
50 }

```

Solver

```

1  FoamFile
2  {
3      version          2.0;
4      format           ascii;
5      class            dictionary;
6      location         "system";
7      object           fvSolution;
8  }
9
10 solvers
11 {
12     p
13     {
14         solver        PCG;
15         tolerance     1e-7;
16         relTol        0.005;
17         preconditioner DIC;
18         maxIter       500;
19     }
20     pFinal
21     {
22         solver        PCG;
23         tolerance     1e-7;
24         relTol        0;
25         preconditioner DIC;
26         maxIter       500;
27     }
28     "(U|k|omega)"
29     {
30         solver        BiCGStab;
31         preconditioner DILU;
32         tolerance     1e-9;
33         relTol        0.0;
34     }
35 }
36

```

```

37 PIMPLE
38 {
39     nCorrectors 10;
40     nNonOrthogonalCorrectors 0;
41     tolerance 5.5e-5;
42     relTol 1e-3;
43     minIter 1;
44     maxIter 12;
45     pisoTol 1e-3;
46 }
47
48 relaxationFactors
49 {
50     p 0.9;
51     U 0.9;
52     k 0.9;
53     omega 0.9;
54 }

```

C-2 preCICE settings

```

1 <?xml version="1.0"?>
2
3 <precice-configuration>
4
5     <solver-interface dimensions="2">
6
7         <data:vector name="Stresses" />
8         <data:vector name="Displacements" />
9
10        <mesh name="Fluid_Nodes">
11            <use-data name="Displacements" />
12        </mesh>
13
14        <mesh name="Fluid_CellCenters">
15            <use-data name="Stresses" />
16        </mesh>
17
18        <mesh name="Structure_Nodes">
19            <use-data name="Displacements" />
20        </mesh>
21
22        <mesh name="Structure_CellCenters">
23            <use-data name="Stresses" />
24        </mesh>
25
26        <participant name="Fluid_Solver">
27            <use-mesh name="Fluid_Nodes" provide="yes" />
28            <use-mesh name="Fluid_CellCenters" provide="yes" />
29            <use-mesh name="Structure_Nodes" from="Structure_Solver" />
30            <use-mesh name="Structure_CellCenters" from="Structure_Solver" />
31            <write-data mesh="Fluid_CellCenters" name="Stresses" />
32            <read-data mesh="Fluid_Nodes" name="Displacements" />
33            <mapping:nearest-neighbor direction="write" from="Fluid_CellCenters" to="
34                Structure_CellCenters" constraint="consistent" timing="initial"/>
35            <mapping:nearest-neighbor direction="read" from="Structure_Nodes" to="Fluid_Nodes"
36                constraint="consistent" timing="initial"/>
37            <master:mpi-single />
38        </participant>
39
40        <participant name="Structure_Solver">
41            <use-mesh name="Structure_Nodes" provide="yes"/>
42            <use-mesh name="Structure_CellCenters" provide="yes"/>
43            <write-data mesh="Structure_Nodes" name="Displacements" />
44            <read-data mesh="Structure_CellCenters" name="Stresses" />
45            <!--<master:mpi-single />-->
46        </participant>
47
48        <!--<m2n:sockets exchange-directory=".." from="Fluid_Solver" to="Structure_Solver" /> -->
49        <m2n:sockets from="Fluid_Solver" to="Structure_Solver" distribution-type="gather-scatter" />
50        <!-- For serial runs -->
51
52        <coupling-scheme:serial-implicit>
53            <timestep-length value="1.0e-5" />

```

```
51     <max-timesteps value="20000" />
52     <participants first="Fluid_Solver" second="Structure_Solver" />
53     <exchange data="Stresses" from="Fluid_Solver" mesh="Structure_CellCenters" to="
54         Structure_Solver" />
55     <exchange data="Displacements" from="Structure_Solver" mesh="Structure_Nodes" to="
56         Fluid_Solver" />
57     <relative-convergence-measure limit="1.0e-6" data="Displacements" mesh="Structure_Nodes"
58         suffices="0" />
59     <max-iterations value="40" />
60     <extrapolation-order value="2" />
61     <post-processing:IQN-ILS>
62         <data mesh="Structure_Nodes" name="Displacements" />
63         <initial-relaxation value="0.001" />
64         <max-used-iterations value="200" />
65         <timesteps-reused value="20" />
66         <filter type="QR1" limit="1e-8" />
67     </post-processing:IQN-ILS>
68     </coupling-scheme:serial-implicit>
69 </solver-interface>
70 </precice-configuration>
```

Appendix D

Simulation results

D-1 Greenhalgh setup

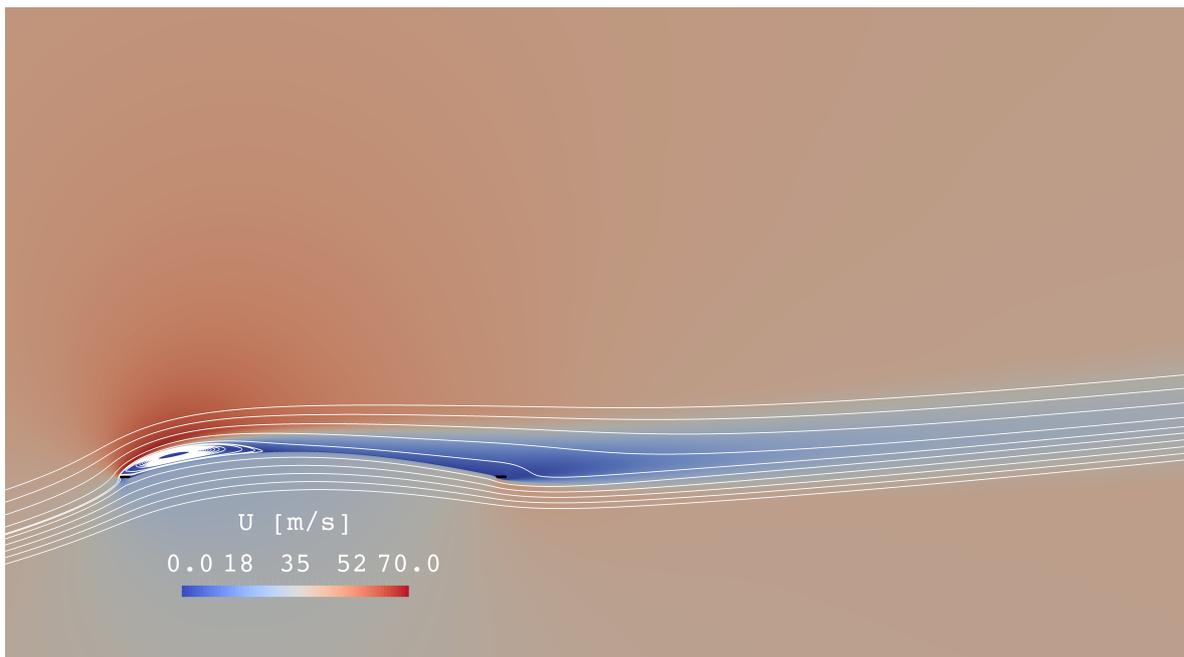


Figure D-1: Flow field for converged shape of the wing with oval supports, $\alpha = 8^\circ$ and $\epsilon_{\text{wing}} = 1.14\%$

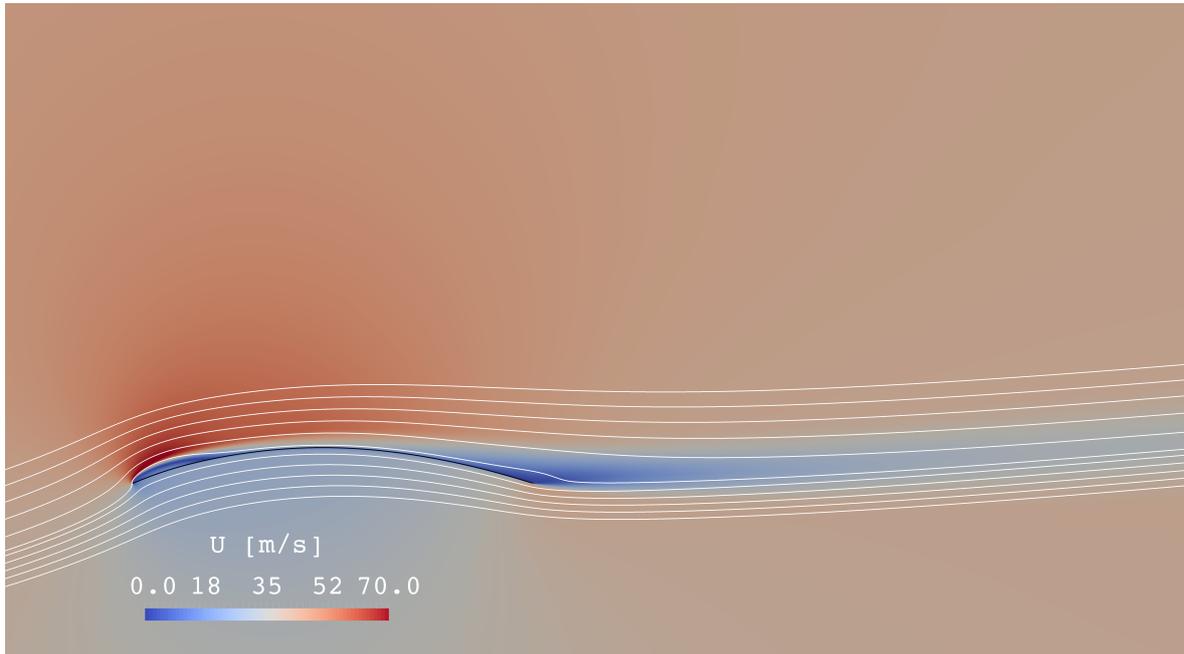


Figure D-2: Flow field for converged shape of the wing without supports, $\alpha = 8^\circ$ and $\epsilon_{\text{wing}} = 1.14\%$

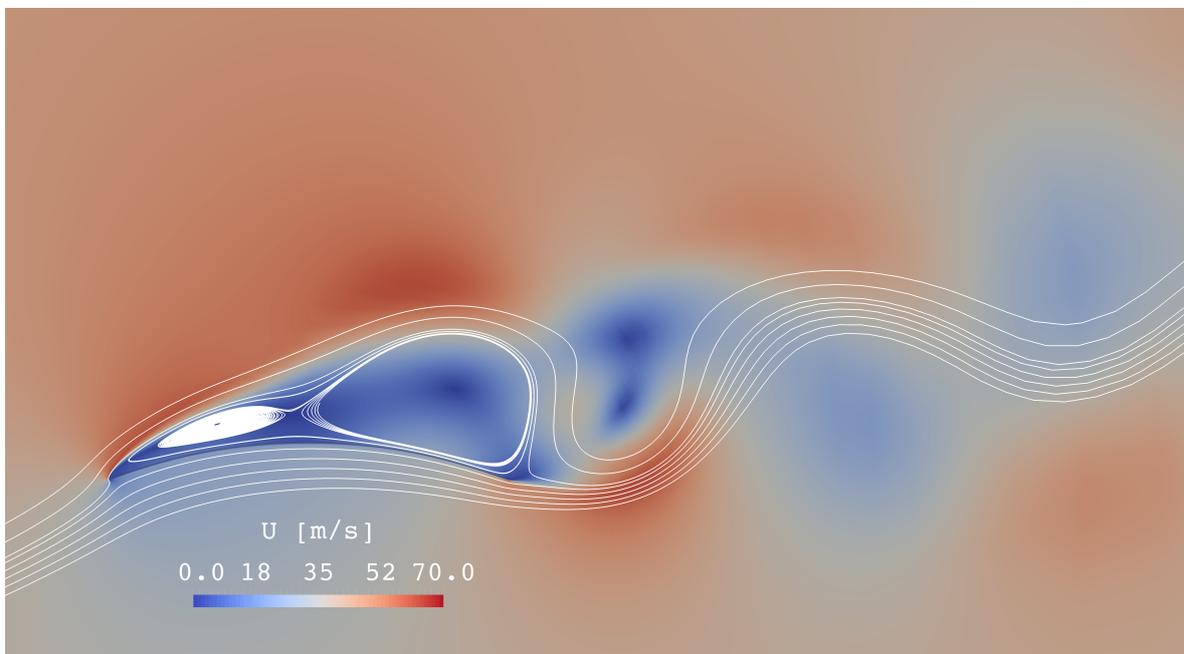


Figure D-3: Instantaneous snapshot of oscillatory flow field of the wing without supports, $\alpha = 16^\circ$ and $\epsilon_{\text{wing}} = 1.14\%$

Bibliography

- [1] S. Turek, J. Hron, R. Mudassar, W. Hilmar, and M. Schäfer, “Numerical benchmarking of fluid-structure interaction: A comparison of different discretization and solution approaches.” 2006.
- [2] R. den Boer, “Low speed aerodynamic characteristics of a two-dimensional sail wing with adjustable slack of the sail,” tech. rep., Faculty of Aerospace Engineering, TU Delft, 1980.
- [3] S. Greenhalgh, H. C. Curtiss, and Smith, “Aerodynamic properties of a two-dimensional inextensible flexible airfoil,” *AIAA Journal*, vol. 22, no. 7, pp. 865–870, 1984.
- [4] M. Z. Jacobson and C. L. Archer, “Saturation wind power potential and its implications for wind energy,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 109, no. 39, pp. 15679–15684, 2012.
- [5] K. Marvel, B. Kravitz, and K. Caldeira, “Geophysical limits to global wind power,” *Nature Clim. Change*, vol. 3, no. 2, pp. 118–121, 2013.
- [6] S. Sawyer, “Global offshore: Current status and future prospect.,” *Global Wind Energy Council*, 2012.
- [7] M. L. Loyd, “Crosswind kite power (for large-scale wind power production),” *Journal of Energy*, vol. 4, no. 3, pp. 106–111, 1980.
- [8] L. C. Archer and K. Caldeira, “Global assessment of high-altitude wind power,” *Energies*, vol. 2, no. 2, 2009.
- [9] L. M. Miller, F. Gans, and A. Kleidon, “Jet stream wind power as a renewable energy resource: Little power, big impacts,” *Earth System Dynamics*, vol. 2, no. 2, pp. 201–212, 2011.
- [10] I. Bastigkeit, P. Rohde, G. Wolken-Möhlmann, and A. Gambier, “Study on wind resources at mid-altitude,” in *6th International Airborne Wind Energy Conference*, (Delft), 2015.

- [11] A. Cherubini, A. Papini, R. Vertechy, and M. Fontana, “Airborne wind energy systems: A review of the technologies,” *Renewable and Sustainable Energy Reviews*, vol. 51, pp. 1461–1476, 2015.
- [12] J. Breukels, *An engineering methodology for kite design*. Phd, TU Delft, 2010.
- [13] H. A. Bosch, “Finite element analysis of a kite for power generation,” Master’s thesis, TU Delft, 2012.
- [14] N. H. Geschiee, “Dynamic modelling of a flexible kite for power generation,” Master’s thesis, TU Delft, 2014.
- [15] J. Berens, “Dynamic nonlinear aeroelastic behaviour of flexible wings in an airflow,” Master’s thesis, TU Delft, 2015.
- [16] K. J. Bathe, *Finite element procedures*. New Delhi :: Prentice-Hall of India, 2007.
- [17] C. A. Felippa, “A study of optimal membrane triangles with drilling freedoms,” *Computer Methods in Applied Mechanics and Engineering*, vol. 192, no. 16, pp. 2125–2168, 2003.
- [18] H. Stolarski and T. Belytschko, “Shear and membrane locking in curved c0 elements,” *Computer Methods in Applied Mechanics and Engineering*, vol. 41, no. 3, pp. 279–296, 1983.
- [19] A. Jarasjarungkiat, R. Wulchner, and K. U. Bletzinger, “A wrinkling model based on material modification for isotropic and orthotropic membranes,” *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 6-8, pp. 773–788, 2008.
- [20] H. Schoop, L. Taenzer, and J. Hornig, “Wrinkling of nonlinear membranes,” *Computational Mechanics*, vol. 29, no. 1, pp. 68–74, 2002.
- [21] R. Barsotti and S. S. Ligaro, “Static response of elastic inflated wrinkled membranes,” *Computational Mechanics : Solids, Fluids, Structures, Fluid-Structure Interactions, Biomechanics, Micromechanics, Multiscale Mechanics, Materials, Constitutive Modeling, Nonlinear Mechanics, Aerodynamics*, vol. 53, no. 5, pp. 1001–1013, 2014.
- [22] R. Rossi, M. Lazzari, R. Vitaliani, and E. Onate, “Simulation of light-weight membrane structures by wrinkling model,” *International Journal for Numerical Methods in Engineering*, vol. 62, no. 15, pp. 2127–2153, 2005.
- [23] D. Chapelle and K.-J. Bathe, *The Finite Element Analysis of Shells - Fundamentals : Fundamentals*. Computational Fluid and Solid Mechanics, Berlin :: Springer, 2011.
- [24] D. Chapelle and K. Bathe, “The mathematical shell model underlying general shell elements,” *International Journal for Numerical Methods in Engineering*, vol. 48, no. 2, pp. 289–313, 2000.
- [25] P. M. Naghdi, “Foundations of elastic shell theory,” pp. 1–90, 1963.
- [26] D. Chapelle and K. J. Bathe, “Fundamental considerations for the finite element analysis of shell structures,” *Computers & Structures*, vol. 66, no. 1, pp. 19–36, 1998.

-
- [27] W. Koiter, “On the nonlinear theory of thin elastic shells,” *Koninklijke Nederlandse Akademie van Wetenschappen, Proceedings, Series B*, vol. 69, no. 1, pp. 1–54, 1965.
- [28] J. Blaauwendraad and J. H. Hoefakker, *Structural Shell Analysis Understanding and Application*. Solid Mechanics and Its Applications, 0925-0042 ; 200; Solid Mechanics and Its Applications, 0925-0042 ; 200., Dordrecht :: Springer Netherlands, 2014.
- [29] M. L. Bucelem and K. J. Bathe, “Finite element analysis of shell structures,” *Archives of Computational Methods in Engineering*, vol. 4, no. 1, pp. 3–61, 1997.
- [30] H.-M. Jeon, Y. Lee, P.-S. Lee, and K.-J. Bathe, “The mitc3+ shell element in geometric nonlinear analysis,” *Computers and Structures*, vol. 146, pp. 91–104, 2015.
- [31] P. Tiso, *Finite element based reduction methods for static and dynamic analysis of thin-walled structures*. PhD thesis, TU Delft, Faculteit der Luchtvaart- en Ruimtevaarttechniek, 2006.
- [32] P.-S. Lee and K.-J. Bathe, “Development of mitc isotropic triangular shell finite elements,” *Computers and Structures*, vol. 82, no. 11, pp. 945–962, 2004.
- [33] S. L. Veldman, O. K. Bergsma, and A. Beukers, “Bending of anisotropic inflated cylindrical beams,” *Thin-Walled Structures*, vol. 43, no. 3, pp. 461–475, 2005.
- [34] C. G. Malm, W. G. Davids, M. L. Peterson, and A. W. Turner, “Experimental characterization and finite element analysis of inflated fabric beams,” *Construction and Building Materials*, vol. 23, no. 5, pp. 2027–2034, 2009.
- [35] K. L. Apedo, S. Ronel, E. Jacquelin, A. Bennani, and M. Massenzio, “Nonlinear finite element analysis of inflatable beams made from orthotropic woven fabric,” *International Journal of Solids and Structures*, vol. 47, no. 16, pp. 2017–2033, 2010.
- [36] A. Le Van and C. Wielgosz, “Finite element formulation for inflatable beams,” *Thin-Walled Structures*, vol. 45, no. 2, pp. 221–236, 2007.
- [37] R. Schmehl and e. al, *Airborne Wind Energy*. Springer, 2013.
- [38] J. H. Ferziger and M. Perić, *Computational methods for fluid dynamics*. Berlin :: Springer, 1996.
- [39] S. B. Pope, *Turbulent flows*. Cambridge :: Cambridge University Press, 2000.
- [40] S. Hulshoff, “Lecture notes on ae4133 cfd 2: Computation and modeling of turbulent flows.” Department of Aerospace Engineering, TU Delft, Spring 2016.
- [41] C. Meneveau and J. Katz, “Scale-invariance and turbulence models for large-eddy simulation,” *Annual Review of Fluid Mechanics*, vol. 32, no. 1, pp. 1–32, 2000.
- [42] D. Mehta, A. H. van Zuijlen, B. Koren, J. G. Holierhoek, and H. Bijl, “Large eddy simulation of wind farm aerodynamics: A review,” *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 133, pp. 1–17, 2014.

- [43] M. Breuer, N. Jovicic, and K. Mazaev, “Comparison of des, rans and les for the separated flow around a flat plate at high incidence,” *International Journal for Numerical Methods in Fluids*, vol. 41, no. 4, pp. 357–388, 2003.
- [44] B. Sanderse, S. P. Pijl, and B. Koren, “Review of computational fluid dynamics for wind turbine wake aerodynamics,” *Wind Energy*, vol. 14, no. 7, pp. 799–819, 2011.
- [45] P. R. Spalart, “Detached-eddy simulation,” *Annual Review of Fluid Mechanics*, vol. 41, no. 1, pp. 181–202, 2009.
- [46] M. Bernardini, D. Modesti, and S. Pirozzoli, “On the suitability of the immersed boundary method for the simulation of high-reynolds-number separated turbulent flows,” *Computers & Fluids*, vol. 130, pp. 84–93, 2016.
- [47] P. Spalart, S. Deck, M. Shur, K. Squires, M. Strelets, and A. Travin, “A new version of detached-eddy simulation, resistant to ambiguous grid densities,” *Theoretical and Computational Fluid Dynamics*, vol. 20, no. 3, pp. 181–195, 2006.
- [48] F. Nicoud, J. S. Baggett, P. Moin, and W. Cabot, “Large eddy simulation wall-modeling based on suboptimal control theory and linear stochastic estimation,” *Physics of Fluids*, vol. 13, no. 10, pp. 2968–2984, 2001.
- [49] R. Mittal and G. Iaccarino, “Immersed boundary methods,” *Annual Review of Fluid Mechanics*, vol. 37, no. 1, pp. 239–261, 2005.
- [50] F. Sotiropoulos and X. Yang, “Immersed boundary methods for simulating fluid–structure interaction,” *Progress in Aerospace Sciences*, vol. 65, pp. 1–21, 2014.
- [51] C. S. Peskin, “Flow patterns around heart valves: A numerical method,” *Journal of Computational Physics*, vol. 10, no. 2, pp. 252–271, 1972.
- [52] Y. H. Tseng and J. H. Ferziger, “A ghost-cell immersed boundary method for flow in complex geometry,” *Journal of Computational Physics*, vol. 192, no. 2, pp. 593–623, 2003.
- [53] N. Peller, A. Le Duc, F. Tremblay, and M. Manhart, “High-order stable interpolations for immersed boundary methods,” *International Journal for Numerical Methods in Fluids*, vol. 52, no. 11, pp. 1175–1193, 2006.
- [54] M. D. de Tullio, P. De Palma, G. Iaccarino, G. Pascazio, and M. Napolitano, “An immersed boundary method for compressible flows using local grid refinement,” *Journal of Computational Physics*, vol. 225, no. 2, pp. 2098–2117, 2007.
- [55] C. C. Pain, A. P. Umpheby, C. R. E. de Oliveira, and A. J. H. Goddard, “Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 190, no. 29–30, pp. 3771–3796, 2001.
- [56] G. Kalitzin and G. Iaccarino, “Turbulence modeling in an immersed-boundary rans method,” *Center for Turbulence Research, Annual Research Briefs*, vol. 29, no. 6, pp. 415–426, 2002.

-
- [57] G. Kalitzin, G. Medic, G. Iaccarino, and P. Durbin, “Near-wall behavior of rans turbulence models and implications for wall functions,” *Journal of Computational Physics*, vol. 204, no. 1, pp. 265–291, 2005.
- [58] U. Küttler and W. A. Wall, “Fixed-point fluid structure interaction solvers with dynamic relaxation,” *Computational Mechanics*, vol. 43, no. 1, pp. 61–72, 2008.
- [59] J. Vierendeels, L. Lanoye, J. Degroote, and P. Verdonck, “Implicit coupling of partitioned fluid-structure interaction problems with reduced order models,” *Computers and Structures*, vol. 85, no. 11, pp. 970–976, 2007.
- [60] J. Degroote, K.-J. Bathe, and J. Vierendeels, “Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction,” *Computers and Structures*, vol. 87, no. 11, pp. 793–801, 2009.
- [61] J. Degroote and J. Vierendeels, “Multi-level quasi-newton coupling algorithms for the partitioned simulation of fluid structure interaction,” *Computer Methods in Applied Mechanics and Engineering*, vol. 225-228, no. 41–43, pp. 14–27, 2012.
- [62] E. H. van Brummelen, “Added mass effects of compressible and incompressible flows in fluid-structure interaction,” *Journal of Applied Mechanics*, vol. 76, no. 2, p. 021206, 2009.
- [63] J. Degroote, P. Bruggeman, R. Haelterman, and J. Vierendeels, “Stability of a coupling technique for partitioned solvers in fsi applications,” *Computers and Structures*, vol. 86, no. 23, pp. 2224–2234, 2008.
- [64] P. Causin, J. F. Gerbeau, and F. Nobile, “Added-mass effect in the design of partitioned algorithms for fluid-structure problems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 42, pp. 4506–4527, 2005.
- [65] I. Borazjani, L. Ge, and F. Sotiropoulos, “Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3d rigid bodies,” *Journal of Computational Physics*, vol. 227, no. 16, pp. 7587–7620, 2008.
- [66] S. Turek and J. Hron, “Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow.”
- [67] E. J. Yates, “Agard standard aeroelastic configuration for dynamic response i-wing 445.6,” tech. rep., NASA TM-100942, 1987.
- [68] S. Levy, “Bending of rectangular plates with large deflections,” tech. rep., Report 737, NACA, 1942.
- [69] Levy, “Square plate with clamped edges under normal pressure producing large deflections,” tech. rep., Report 740, NACA, 1942.
- [70] E. H. Dowell, “Nonlinear oscillations of a fluttering plate,” *AIAA Journal*, vol. 4, no. 7, pp. 1267–1275, 1966.
- [71] E. H. Dowell, “Nonlinear oscillations of a fluttering plate. ii,” *AIAA Journal*, vol. 5, no. 10, pp. 1856–1862, 1967.

- [72] E. Dowell and W. Ye, "Limit cycle oscillation of a fluttering cantilever plate," *AIAA Journal*, vol. 29, no. 11, pp. 1929–1936, 1991.
- [73] D. Tang and E. H. Dowell, "Effects of angle of attack on nonlinear flutter of a delta wing," *AIAA Journal*, vol. 39, no. 1, pp. 15–21, 2001.
- [74] A. Song, X. Tian, E. Israeli, R. Galvao, K. Bishop, S. Swartz, and K. Breuer, "Aeromechanics of membrane wings with implications for animal flight," *AIAA Journal*, vol. 46, no. 8, pp. 2096–2106, 2008.
- [75] P. Rojratsirikul, Z. Wang, and I. Gursul, "Unsteady fluid-structure interactions of membrane airfoils at low reynolds numbers," *Experiments in Fluids : Experimental Methods and their Applications to Fluid Flow*, vol. 46, no. 5, pp. 859–872, 2009.
- [76] P. Rojratsirikul, M. S. Genc, Z. Wang, and I. Gursul, "Flow-induced vibrations of low aspect ratio rectangular membrane wings," *Journal of Fluids and Structures*, vol. 27, no. 8, pp. 1296–1309, 2011.
- [77] P. Rojratsirikul, Z. Wang, and I. Gursul, "Effect of pre-strain and excess length on unsteady fluid-structure interactions of membrane airfoils," *Journal of Fluids and Structures*, vol. 26, no. 3, pp. 359–376, 2010.
- [78] S. Arbos, *Aeromechanical Performance of Compliant Aerofoils*. PhD thesis, Imperial College London, 2013.
- [79] S. Arbos, B. Ganapathisubramani, and R. Palacios, "Leading- and trailing-edge effects on the aeromechanics of membrane aerofoils," *Journal of Fluids and Structures*, vol. 38, pp. 107–126, 2013.
- [80] R. E. Gordnier, "High fidelity computational simulation of a membrane wing airfoil," *Journal of Fluids and Structures*, vol. 25, no. 5, pp. 897–917, 2009.
- [81] J. W. Jaworski and R. E. Gordnier, "High-order simulations of low reynolds number membrane airfoils under prescribed motion," *Journal of Fluids and Structures*, vol. 31, pp. 49–66, 2012.
- [82] R. E. Gordnier and P. J. Attar, "Impact of flexibility on the aerodynamics of an aspect ratio two membrane wing," *Journal of Fluids and Structures*, vol. 45, pp. 138–152, 2014.
- [83] J. W. Jaworski and R. E. Gordnier, "Thrust augmentation of flapping airfoils in low reynolds number flow using a flexible membrane," *Journal of Fluids and Structures*, vol. 52, pp. 199–209, 2015.
- [84] S. Greenhalgh, J. H. Curtiss, and B. Smith, "Aerodynamic properties of a two-dimensional inextensible flexible airfoil," in *Applied Aerodynamics Conference, Fluid Dynamics and Co-located Conferences*, American Institute of Aeronautics and Astronautics, 1983.
- [85] S. Greenhalgh, "The two-dimensional inextensible lifting membrane airfoil - theory & experiment," Tech. Rep. IR No. GC123, Naval Air Development Center Warminster PA, 1983.

-
- [86] T. Sweeney, “Exploratory sail wing research at princeton,” tech. rep., 1961.
- [87] M. P. Fink, “Full-scale investigation of the aerodynamic characteristics of a model employing a sailwing concept,” tech. rep., National Aeronautics and Space Administration, 1967.
- [88] M. P. Fink, *Full-scale investigation of the aerodynamic characteristics of a sailwing of aspect ratio 5.9*. NASA technical note ; D-5047, Washington :: NASA, 1969.
- [89] R. A. Ormiston, “Theoretical and experimental aerodynamics of the sailwing,” *Journal of Aircraft*, vol. 8, no. 2, pp. 77–84, 1971.
- [90] M. Maughmer, “A comparison of the aerodynamic characteristics of eight sailwing airfoil sections,” tech. rep., 1979.
- [91] R. den Boer, “Numerical and experimental investigation of the aerodynamics of double membrane sailwing airfoil sections,” tech. rep., Faculty of Aerospace Engineering, TU Delft, 1982.
- [92] A. de Wachter, “Deformation and aerodynamic performance of a ram-air wing,” Master’s thesis, TU Delft, 2008.
- [93] K. Karagiozis, R. Kamakoti, F. Cirak, and C. Pantano, “A computational study of supersonic disk-gap-band parachutes using large-eddy simulation coupled to a structural membrane,” *Journal of Fluids and Structures*, vol. 27, no. 2, pp. 175–192, 2011.
- [94] T. E. Tezduyar, K. Takizawa, C. Moorman, S. Wright, and J. Christopher, “Space-time finite element computation of complex fluid-structure interactions,” *International Journal for Numerical Methods in Fluids*, vol. 64, no. 10-12, pp. 1201–1218, 2010.
- [95] J. Piquee and C. Breitsamter, “Numerical analysis of an elasto-flexible wing configuration using computational fluid structural simulations,” *Textile Composites and Inflatable Structures Vii*, pp. 356–367, 2015.
- [96] M. Saeedi, K. U. Bletzinger, and R. Wüchner, “Multi-fidelity fluid-structure interaction analysis of a membrane blade concept in non-rotating, uniform flow condition,” *Wind Energ. Sci.*, vol. 1, no. 2, pp. 255–269, 2016.
- [97] M. Saeedi, K.-U. Bletzinger, and R. Wüchner, “Investigation of prestress-dependent aerodynamic performance of a double membrane sailwing,” *Journal of Aircraft*, pp. 1–15, 2016.
- [98] M. Saeedi, R. Wüchner, K. U. Bletzinger, and T. S. o. M. T. f. W. T. Science of Making Torque from Wind, “Fluid-structure interaction analysis and performance evaluation of a membrane blade,” *Journal of Physics: Conference Series*, vol. 753, no. 10, 2016.
- [99] R. Smith and W. Shyy, “Computation of aerodynamic coefficients for a flexible membrane airfoil in turbulent flow: A comparison with classical theory,” *Physics of Fluids*, vol. 8, no. 12, pp. 3346–3353, 1996.
- [100] U. Fechner, *A Methodology for the Design of Kite-Power Control Systems*. PhD thesis, Delft University of Technology, 2016.

- [101] Kitepower, “Pumping cycle.” <http://kitepower.eu/technology/2-core-concepts/6-pumping-operation.html>. [Accessed 7 April 2017].
- [102] A. Bosch, R. Schmehl, P. Tiso, and D. Rixen, “Dynamic nonlinear aeroelastic model of a kite for power generation,” *Journal of guidance, control and dynamics*, vol. 37, no. 5, 2014.
- [103] J. Schwoll, “Finite element analysis of inflatable structures using uniform pressure,” msc thesis, TU Delft, 2012.
- [104] R. Van Kappel, “Aerodynamic analysis tool for dynamic leading edge inflated kite models: a nonlinear vortex lattice method,” msc thesis, TU Delft, 2012.
- [105] D. J. Allman, “A simple cubic displacement element for plate bending,” *International Journal for Numerical Methods in Engineering*, vol. 10, no. 2, pp. 263–281, 1976.
- [106] D. J. Allman, “Evaluation of the constant strain triangle with drilling rotations,” *International Journal for Numerical Methods in Engineering*, vol. 26, no. 12, pp. 2645–2655, 1988.
- [107] D. J. Allman, “Implementation of a flat facet shell finite element for applications in structural dynamics,” *Computers and Structures*, vol. 59, no. 4, pp. 657–663, 1996.
- [108] Unknown, “Lecture notes on asen6107 nonlinear finite element methods.” Department of Aerospace Engineering Sciences, University of Colorado at Boulder, Spring 2016.
- [109] A. Alipour and F. Zareian, “Study rayleigh damping in structures - uncertainties and treatments,” in *The 14th World Conference on Earthquake Engineering, October 12-17, 2008, Beijing, China*, 2008.
- [110] K. J. Bathe and A. P. Cimento, “Some practical procedures for the solution of nonlinear finite element equations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 22, no. 1, pp. 59–85, 1980.
- [111] R. Kaiser, “Rechnerische und experimentelle ermittlung der durchbiegungen und spannungen von quadratischen platten bei freier auflagerung an den rändern, gleichmässig verteilter last und grossen ausbiegungen,” *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 16(2), pp. 73–98, 1936.
- [112] K. Y. Sze, X. H. Liu, and S. H. Lo, “Popular benchmark problems for geometric nonlinear analysis of shells,” *Finite Elements in Analysis and Design*, vol. 40, no. 11, pp. 1551–1569, 2004.
- [113] E. Hinton, T. Rock, and O. C. Zienkiewicz, “A note on mass lumping and related processes in the finite element method,” *Earthquake Engineering & Structural Dynamics*, vol. 4, no. 3, pp. 245–249, 1976.
- [114] J. van den Berg, C. Jux, and J. Steiner, “Rotor/wake aerodynamics: Unsteady aerodynamics.” Assignment report for lecture AE4135, TU Delft.
- [115] “Fluidity.” <http://fluidityproject.github.io>. Accessed 01.03.2017.

-
- [116] M. D. Piggott, G. J. Gorman, C. C. Pain, P. A. Allison, A. S. Candy, B. T. Martin, and M. R. Wells, “A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes,” *International Journal for Numerical Methods in Fluids*, vol. 56, no. 8, pp. 1003–1015, 2008.
- [117] P. M. Gresho and R. L. Sani, “Incompressible flow and the finite element method : advection-diffusion and isothermal laminar flow,” *Journal of Computational Physics*, 1988.
- [118] P. E. Farrell and J. R. Maddison, “Conservative interpolation between volume meshes by local galerkin projection,” *Computer Methods in Applied Mechanics and Engineering*, vol. 200, no. 1-4, pp. 89–100, 2011.
- [119] P. E. Farrell, M. D. Piggott, C. C. Pain, G. J. Gorman, and C. R. Wilson, “Conservative interpolation between unstructured meshes via supermesh construction,” *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 33, pp. 2632–2642, 2009.
- [120] F. R. Menter, “Zonal two equation $k-\omega$ turbulence models for aerodynamic flows,” in *23rd Fluid Dynamics, Plasmadynamics, and Lasers Conference*, vol. AIAA-93-29, 1993.
- [121] W. P. Jones and B. E. Launder, “The calculation of low-reynolds-number phenomena with a two-equation model of turbulence,” *International Journal of Heat and Mass Transfer*, vol. 16, no. 6, pp. 1119–1130, 1973.
- [122] D. C. Wilcox, “Reassessment of the scale-determining equation for advanced turbulence models,” *AIAA Journal*, vol. 26, no. 11, pp. 1299–1310, 1988.
- [123] D. A. Johnson and L. S. King, “A mathematically simple turbulence closure model for attached and separated turbulent boundary layers,” *AIAA Journal*, vol. 23, no. 11, pp. 1684–1692, 1985.
- [124] F. R. Menter, M. Kuntz, and R. Langtry, “Ten years of industrial experience with the sst turbulence model,” *Turbulence Heat and Mass Transfer 4*, vol. 4, pp. 625–632, 2003.
- [125] M. Deaves, “An investigation of the non-linear 3d flow effects relevant for leading edge inflatable kites,” Master’s thesis, TU Delft, 2015.
- [126] A. Viré, J. Xiang, F. Milthaler, P. E. Farrell, M. D. Piggott, J.-P. Latham, D. Pavlidis, and C. C. Pain, “Modelling of fluid–solid interactions using an adaptive mesh fluid model coupled with a combined finite–discrete element model,” *Ocean Dynamics*, vol. 62, no. 10, pp. 1487–1501, 2012.
- [127] A. Viré, J. Xiang, and C. C. Pain, “An immersed shell method for modelling fluid structure interactions,” *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 373, no. 2035, pp. 1–14, 2015.
- [128] N. Krishnan, A. Viré, R. Schmehl, and G. van Bussel, “An immersed boundary method using finite element methods,” *Computer Methods in Applied Mechanics and Engineering*, 2017. (draft version).
- [129] “foam-extend.” <https://github.com/Unofficial-Extend-Project-Mirror/foam-extend-foam-extend-3.2.git>. Accessed 23.11.2017.

- [130] “preCICE.” <https://github.com/precice/precice.git>. Accessed 23.11.2017.
- [131] “FOAM-FSI.” <https://github.com/davidsblom/FOAM-FSI.git>. Accessed 23.11.2017.
- [132] D. Blom, *Efficient Numerical Methods for PartitiPart Fluid-Structure Interaction Simulations*. PhD thesis, TU Delft, Aerospace Faculty, 2017.
- [133] T. Gillebaart, *Towards Efficient Fluid-Structure-Control Interaction for Smart Rotors*. PhD thesis, TU Delft, Aerospace Faculty, 2016.
- [134] T. Gillebaart, D. Blom, A. van Zuijlen, and H. Bijl, “Adaptive radial basis function mesh deformation using data reduction,” *Journal of Computational Physics*, vol. 321, no. Supplement C, pp. 997 – 1025, 2016.
- [135] F. Moukalled, L. Mangani, and M. Darwish, *The finite volume method in computational fluid dynamics : An advanced introduction with openFOAM and Matlab*. Springer, 2015.
- [136] H. Jasak, *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows*. PhD thesis, Imperial College London, Department of Mechanical Engineering, 1996.
- [137] R. Issa, “Solution of the implicitly discretised fluid flow equations by operator-splitting,” *Journal of Computational Physics*, vol. 62, no. 1, pp. 40 – 65, 1986.
- [138] A. de Boer, A. H. van Zuijlen, and H. Bijl, *Radial Basis Functions for Interface Interpolation and Mesh Deformation*, pp. 143–178. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.
- [139] A. de Boer, M. van der Schoot, and H. Bijl, “Mesh deformation based on radial basis function interpolation,” *Computers & Structures*, vol. 85, no. 11, pp. 784 – 795, 2007. Fourth MIT Conference on Computational Fluid and Solid Mechanics.
- [140] J. W. Slater, “Examining spatial grid convergence.”
- [141] K. Ramesh, J. Ke, A. Gopalarathnam, and J. Edwards, “Effect of airfoil shape and reynolds number on leading edge vortex shedding in unsteady flows,” in *Fluid Dynamics and Co-located Conferences*, pp. –, American Institute of Aeronautics and Astronautics, June 2012.
- [142] N. Sørensen, B. Mendez, A. Munoz, G. Sieros, E. Jost, T. Lutz, G. Papadakis, S. Voutsinas, G. Barakos, S. Colonia, D. Baldacchino, C. Baptista, and C. Ferreira, “Cfd code comparison for 2d airfoil flows,” vol. 753, p. 082019, 09 2016.
- [143] “CFD Online. Inflow Turbulence Calculator.” <https://www.cfd-online.com/Tools/turbulence.php>. Accessed 09.11.2017.
- [144] Empa, “Biaxial test report ripstop nylon,” tech. rep., Empa materials science and technology, 2011.
- [145] M. Folkersma, “Unpublished results from phd student working on sailwings using openfoam-plus.” (Stand 11.01.2018).

- [146] R. Leloup, *Modelling approach and numerical tool developments for kite performance assessment and mechanical design; application to vessels auxiliary propulsion*. PhD thesis, Université de Bretagne, 2014.
- [147] R. C. Leuthold, “Multiple-wake vortex lattice method for membrane-wing kites,” Master’s thesis, TU Delft, 2015.
- [148] J. N. Reddy, *An introduction to nonlinear finite element analysis*. Oxford :: Oxford University Press, 2004.
- [149] S. Timoshenko, *Theory of Elastic Stability*. McGraw-Hill Book Co. Inc., 1940, 1940.
- [150] S. J. Hulshoff, “Lecture notes on ae4930 aeroelasticity.” Department of Aerospace Engineering, TU Delft, Spring 2016.

