



Budapest University of Technology and Economics
Faculty of Mechanical Engineering
Department of Fluid Mechanics

Data-driven aerodynamic models for reduced-order aeroelastic simulations

TDK Thesis

Name: HORVÁTH DÁVID ANDRÁS

Neptun: YJEQDB

Supervisors:

Lelkes János Lendvai Bálint

PhD student PhD student

Last modification: November 7, 2021, Budapest

Acknowledgement

I would like to thank my leading advisor Lelkes János for his persistent support in writing this thesis.

I would also like to acknowledge my secondary advisor Lendvai Bálint for making his CFD simulations available.

Budapest, 2021

Horváth Dávid

Absztrakt

Áramlásba helyezett, karcsú rugalmas szerkezetek esetén számos módszer létezik az aerodinamikai erők időbeli meghatározására. Az aerodinamikai erők számítására analitikus, szemi-empirikus, CFD szimulációs és különböző redukált modelleket alkalmazhatunk. Ezen modellek leggyakrabban csak kis kitérésekre alkalmazhatóak, nagy kitérések esetén az aerodinamikai erők számítása rendkívül idő- és számításigényes. Jelen dolgozatban az aerodinamikai erők modellezésére egy adatalapú identifikációs módszert alkalmazunk. Ezen modell legfőbb előnye, hogy nagy állásszögek esetén is pontos, illetve az adatokra történő illesztést követően rendkívül gyors futásidejű szimulációk végezhetőek a segítségével. Az adatalapú modell megalkotásához kényszermozgásos, nagy pontosságú, validált CFD szimulációkat alkalmazunk. A modell megalkotása során az úgynevezett SINDy (nemlineáris rendszerek együtttható-identifikációja) algoritmust alkalmazzuk. A modell illesztéshez többféle optimalizációs módszert kipróbálunk, pl. LASSO (legkisebb zsugorodás és kiválasztás operátor), és STLSQ (szekvenciális küszöbértékű legkisebb négyzetek módszere). A dolgozat eredményeképpen nemcsak egy adatalapú aerodinamikai modellt kapunk, hanem egy kompakt eljárást, amellyel tetszőleges, periodikusan mozgó szerkezetek aerodinamikai terhelése meghatározható időtartományban.

Abstract

Numerous methods exist for calculating the time-dependent aerodynamic loads of thin elastic structures subjected to airflow. For the calculation of the aerodynamic loads, analytical, semi-empirical, CFD-based, and various reduced-order models exist. These models are usually only applicable for small deformations; in the case of large deformations, calculating aerodynamic loads is computationally costly. In this thesis, we apply a data-based identification method for the calculation of the aerodynamic loads. The most significant advantage of this method is that it is also accurate for large angles of attack. After the initial data fitting process, the simulations run very quickly. For the creation of the data-based model, we used high precision validated CFD simulations. For the model construction, we use the SINDy (Sparse Identification of Nonlinear Dynamics) algorithm. We tried multiple optimization routines to fit the model, e.g., LASSO (Least Absolute Shrinkage and Selection Operator), STLSQ (Sequentially Thresholded Least Squares). As the result of the thesis, we get not only a data-based aerodynamic model for a single configuration but also a compact process, with which the aerodynamic loading of arbitrary moving structures can be calculated in the time domain.

Contents

Nomenclature	3
1 Introduction	5
1.1 Aeroelasticity	6
1.2 Sparse Identification of Nonlinear Dynamical Systems	9
1.3 The aim of this thesis	11
2 Data-driven aerodynamic models	13
2.1 CFD simulation	13
2.2 Identified models	17
3 Application of the aerodynamic model on a the two-degree-of-freedom system	34
3.1 Model $M_{0.5,5}$	36
3.2 Model $M_{0.5,10}$	37
3.3 Model $M_{0.1,5}$	38
3.4 Model $M_{0.1,10}$	39
4 Summary	41
Bibliography	43

Nomenclature

CFD	Computational Fluid Dynamics
DOF	Degrees Of Freedom
FEM	Finite Element Method
FSI	Fluid-Structure Interaction
LASSO	Least Absolute Shrinkage and Selection Operator
MSE	Mean Squared Error
SINDy	Sparse Identification of Nonlinear Dynamics
STLSQ	Sequentially Thresholded Least Squares
b	Semi-chord
c_h	Viscous damping plunge DOF
c_i	Coefficients of the identified models
C_l	Lift coefficient
c_α	Viscous damping pitch DOF
f_h	Natural frequency for the plunge DOF
f_α	Natural frequency for the pitch DOF
h	Plunge DOF displacement
H_i, A_i	Flutter derivatives
I_α	Moment of inertia around pitching axis
K	Scalan's reduced frequency
k	Reduced frequency
k_h	Spring constant plunge DOF
k_α	Spring constant pitch DOF
L	Lift force
M	Aerodynamic moment
m	Mass of the wing
$M_{k,\Delta\alpha}$	Identified model
p	Pressure
Re	Reynolds number
S_α	Mass unbalance
U	Flow velocity
k_{α^3}	Nonlinear stiffness coefficient

$\Delta\alpha$	Amplitude of the constrained motion for the training set
Ξ	Vector of weights
Θ	Feature library
α	Pitch DOF displacement
λ	Regularization parameter
ξ_h	Damping factor for the plunge DOF
ξ_α	Damping factor for the pitch DOF
ρ	Air density
ω	Angular velocity
ω_h	Angular natural frequency for the plunge DOF
ω_α	Angular natural frequency for the pitch DOF

Chapter 1

Introduction

Obtaining accurate and efficient aerodynamic models has been a fundamental goal of research efforts in aeronautics over the past century [1]. Aerodynamic models are essential for designing aircraft, evaluating static and dynamic aeroelastic stability, and developing feedback control laws. Closed-form solutions for the attached incompressible unsteady flow problem around a two-dimensional (2D) airfoil exist in both the frequency and time domains [2].

Wagner [3] developed a model for the unsteady lift on a two-dimensional flat plate for arbitrary small-amplitude pitching motion. He computed the effect of idealized planar wake vorticity on the circulation around the plate in response to a step in the angle of attack analytically. After this, the response to arbitrary motion could be constructed by convolution with this indicial response. Ten years later, Theodorsen [4] derived a complementary model to study the aeroelastic problem of flutter instability in the frequency domain. Wagner’s and Theodorsen’s theories were derived analytically for an idealized two-dimensional flat plate moving through an inviscid, incompressible fluid. The motion of the flat plate is assumed to be infinitesimal, leaving an idealized planar wake.

The finite state flow model offers state equations for the induced flow field itself [5]. The governing equations of the finite state flow model were derived directly from the potential flow equations (either velocity or acceleration potential). Thus, no intermediate steps were invoked in which restrictions were placed on airfoil motions. The theory is an arbitrary-motion theory from the outset. In contrast to Computational Fluid Dynamics (CFD) and vortex lattice methods, the states represent induced flow expansion fields rather than velocities at discrete nodes. As a result, the states are hierarchical, and the equation coefficients are known in closed form. No numerical fitting of frequency-response or indicial functions is needed to apply the finite state flow model.

The dynamic stall phenomenon and its importance for load calculations and aeroelastic simulations are well known. Different models exist to model the effect of the dynamic stall. For dynamic stall, the physics of the flow separation and stall development differs fundamentally from the stall mechanisms observed for the same airfoil under static (quasi-steady) conditions [6].

The effect of dynamic stall can be defined as a delay in the stall onset: stall occurs at a higher angle of attack than for the static stall case. A strong vortex is formed at the leading edge that separates and is convected along the suction surface of the airfoil. This event begins with a rapid increase in lift and ends with complete flow separation and catastrophic loss of lift as the vortex disturbance is convected past the trailing edge of the airfoil. This behavior can produce hysteresis loops in the force coefficients, producing cyclic pressure loads that are not predicted by conventional lift and drag data obtained at steady angles of attack [1].

The ONERA semi-empirical model [7] describes the unsteady airfoil behavior employing a set of nonlinear differential equations. A first-order linear differential equation describes the inviscid (attached flow) aerodynamic contribution, and a second-order differential equation describes the nonlinear viscous effects associated with the stall.

The Beddoes–Leishman method [8] is a dynamic stall model where the emphasis is on a more accurate and complete physical representation of the overall unsteady aerodynamic problem but still in a form that keeps the complexity of analysis down to minimize computational cost. In this way, it attempts to overcome the limitations of other models where many empirically determined coefficients limit the method’s applicability. The Beddoes–Leishman model essentially consists of four subsystems:

1. an attached flow model for the unsteady (linear) aerodynamic forces based on Duhamel superposition,
2. a separated flow model for the nonlinear aerodynamic forces,
3. a dynamic stall onset model,
4. a dynamic stall model for the vortex-induced aerodynamic forces.

The nonlinear, second-order dynamic stall model developed by H. Snel [9, 10] is an example of a modern semi-empirical engineering model used to include dynamic stall effects in aeroelastic response codes for wind turbines. This model uses no airfoil-specific parameters in its modeling equations but still can predict the dynamic stall with the same accuracy as models that require such input. This characteristic makes Snel’s dynamic stall model desirable for application in an aeroelastic design code where numerous airfoil profiles need to be evaluated, negating the need for any parameter identification from dynamic wind tunnel tests.

1.1 Aeroelasticity

The application of accurate reduced aerodynamic models is essential in the field of aeroelastic engineering structures. Aeroelasticity is the interaction of aerodynamic, elastic, and inertial forces [11]. We can distinguish between four types of aeroelastic phenomena

depending on which of the interaction forces are dominant. This can be illustrated using the Collar's force triangle [12], see Figure 1.1.

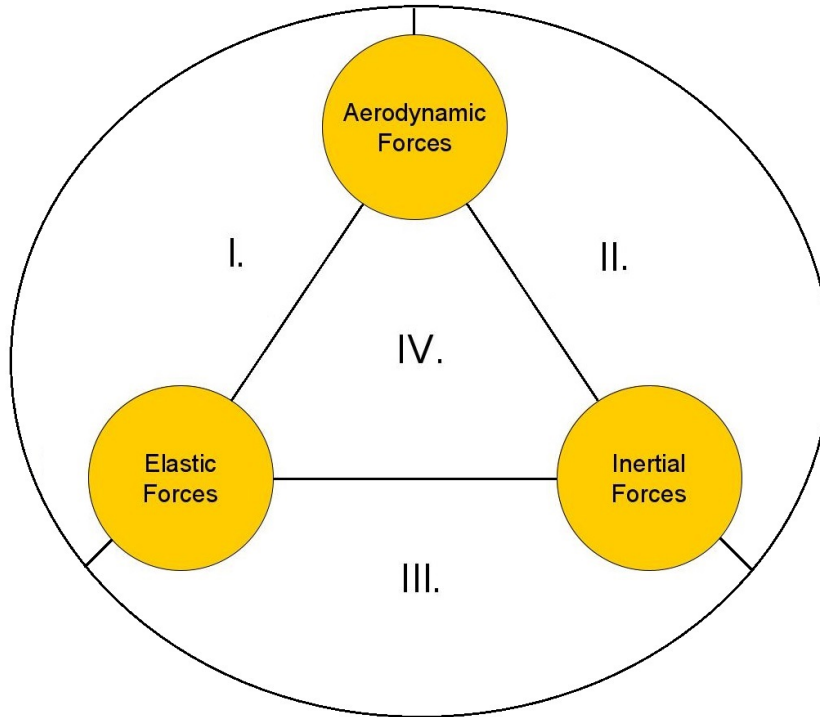


Figure 1.1. Collar's force triangle [12]

The four regions (I - IV.) in Figure 1.1 represent the different types of aeroelastic interactions:

- I. When the aerodynamic and elastic forces are dominant, aperiodic aeroelastic phenomena can be observed, for example, divergence, or reversal.
- II. In this region, the main forces involved are the aerodynamic and inertial forces. The rigid airplane is a classical example of this type of interaction.
- III. This region corresponds to mechanical vibrations because the elastic and inertial forces are dominant. An example of this interaction is the vibrations of the landing gear suspension.
- IV. When the aerodynamic, elastic and inertial forces are equally dominant, periodic aeroelastic phenomena can occur, like flutter, galloping, or buffeting [13].

1.1.1 Aeroelastic Instabilities

Aeroelastic instability occurs when a flexible structure loses its stability caused by an airflow faster than the structure's critical wind velocity [14]. The most common aeroelastic instabilities are divergence, reversal, and flutter.

Divergence is a static aeroelastic phenomenon when the wing twists with respect to the body of the airplane. The aerodynamic forces become larger than the structure can handle, which causes the failure of the system.

Reversal instability occurs when the control surfaces become ineffective. This happens when the twist of the elastic wing negates the effect of the control surface.

Flutter is a dynamic aeroelastic instability when periodical twisting and bending motion of the wing is combined. This type of instability is illustrated in Figure 1.2. The upper part of the figure depicts the so-called wing-flutter. On the lower part of the figure, the so-called control surface flutter is illustrated. During control surface flutter, the wing and the control surface both bend, but only the control surface twists [13].

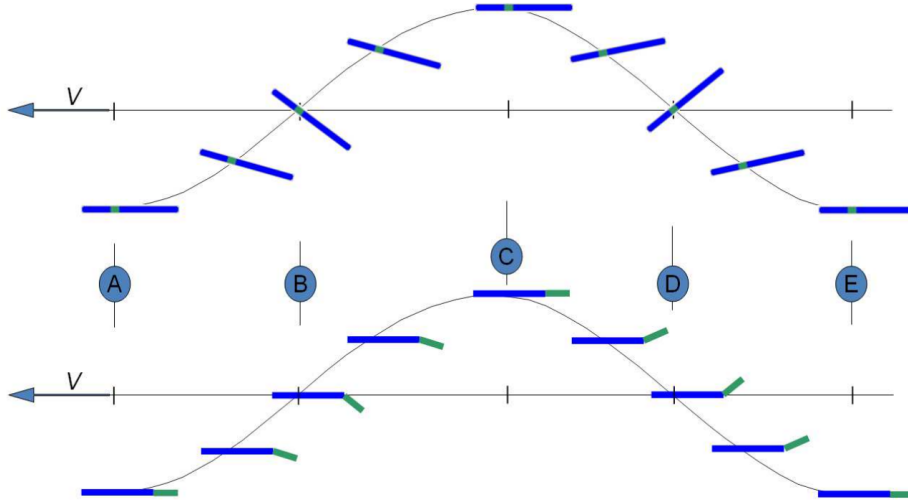


Figure 1.2. Characteristic motion during flutter oscillation [13]

A famous example of the flutter instability is the vibrations and structural failure of the Tacoma Narrows bridge (see Figure 1.3). Shortly after the construction, it became evident that even small winds cause large oscillations of the bridge deck. Even though efforts were made to control the oscillations, it proved ineffective. Wind tunnel tests were carried out to find a solution. However, before the results could be implemented, the bridge collapsed [15, 16].



Figure 1.3. Oscillations and the collapse of Tacoma Narrows bridge [17]

Another impressive example of flutter is the tail flutter of the Piper PA-38 plane, see Figure 1.4



Figure 1.4. Flutter of the Piper PA-38 plane [18]

1.2 Sparse Identification of Nonlinear Dynamical Systems

Traditionally, dynamical systems are modeled by first principles, such as Newton’s second law. However, this approach can be time-consuming while requiring expert knowledge, and the resulting models can be too simplistic to capture the real-world dynamics accurately. With the advent of powerful computers and efficient machine learning algorithms, modeling based on real-world measurement data becomes possible.

In this thesis, we use the Sparse Identification of Nonlinear Dynamical systems (SINDy) method, introduced by Brunton et al. [19, 20], and later refined in the work of Champion et al. [21]. An overview of the method, as well as the description of a Python package that is used in our thesis, is given by Silva et al. [22]. Several alternate versions of this method have been proposed. Kaheman et al. [23] introduced SINDy-PI, which is a parallel version that can be used to identify implicit dynamics while being robust to noise. Kaptanoglu et al. [24] proposed a modification of the SINDy algorithm, which is useful for identifying globally stable models. The SINDy method can also be modified to be applicable to various boundary value problems, as shown by Shea et al. [25].

An overview of data-driven methods in aerospace engineering is given by Brunton et al. [26]. In the work of Sun et al. and Pohl et al. [27, 28] the SINDy method is used to derive polynomial models for the lift of an airfoil.

Here, a brief introduction to SINDy is given; more information can be found in the articles mentioned above. Our aim is to determine a model for a dynamical system in the following form:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, t), \tag{1.1}$$

where $\mathbf{x}$ is the vector of the state variables, and t is the time variable. We want to

determine the function $\mathbf{f}$. The idea of the SINDy method is that usually $\mathbf{f}$ has a sparse representation in the space of the possible functions. Due to this, a sparse regression is used to discover $\mathbf{f}$. Let us denote

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^T(t_1) \\ \vdots \\ \mathbf{x}^T(t_n) \end{bmatrix}, \quad (1.2)$$

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{\mathbf{x}}^T(t_1) \\ \vdots \\ \dot{\mathbf{x}}^T(t_n) \end{bmatrix}, \quad (1.3)$$

where we measure the state vector $\mathbf{x}$ at $t_1, \dots, t_n$ time points, and either also measure the derivative of the state vector, or calculate it numerically from $\mathbf{x}$. The feature library is denoted by $\Theta(\mathbf{X})$, for example, for a polynomial library:

$$\Theta(\mathbf{X}) = \begin{bmatrix} \mathbf{1} & \mathbf{X} & \mathbf{X}^{P_2} \dots \end{bmatrix}, \quad (1.4)$$

where $\mathbf{X}^{P_2}$ denotes the second order polynomials

$$\mathbf{X}^{P_2} = \begin{bmatrix} x_1^2(t_1) & x_1(t_1)x_2(t_1) & \dots & x_2^2(t_1) & \dots & x_m^2(t_1) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_1^2(t_n) & x_1(t_n)x_2(t_n) & \dots & x_2^2(t_n) & \dots & x_m^2(t_n) \end{bmatrix}. \quad (1.5)$$

We are searching for the vector $\Xi = [\xi_1 \dots \xi_m]$:

$$\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi. \quad (1.6)$$

Once this is found, a model can be constructed as follows:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) = \Theta(\mathbf{x}^T)\Xi. \quad (1.7)$$

For real-world data, usually $\mathbf{X}$, and $\dot{\mathbf{X}}$ are contaminated with noise, so instead of equation (1.6) we have

$$\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi + \eta\mathbf{Z}, \quad (1.8)$$

where noise is modelled as a Gaussian distribution with zero mean, and η noise magnitude. There are numerous objective functions, that can be used for the sparse regression, we have mostly used the sequentially thresholded least squares (STLSQ), and the least absolute shrinkage and selection operator (LASSO), which are respectively l_2 , and l_1 regularized regression promoting sparsity [19].

1.2.1 SINDy optimizers

As we wrote previously, the SINDy method learns the model by a regression algorithm. Three main optimizers are used for this: the sequentially thresholded least squares (STLSQ), the least absolute shrinkage and selection operator (LASSO), and the SR3 algorithm. In our study, we have used the former 2, so those will be detailed in this section.

The LASSO optimizer is a simple l_1 regularized regression with the following objective function

$$\frac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \lambda \|\mathbf{w}\|_1, \quad (1.9)$$

where $\mathbf{y}$ is the training data $\dot{\mathbf{X}}$, $\mathbf{X}$ is the function library $\Theta(\mathbf{X})$, $\mathbf{w}$ is the vector of weights Ξ , and λ is the regularization parameter. The norms are defined the following way

$$\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|, \quad (1.10)$$

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}, \quad (1.11)$$

where $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]^T$. The other optimizer, that we have used is the STLSQ, which was originally proposed to be used with the SINDy algorithm in [19]. It has the following objective function

$$\frac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \lambda \|\mathbf{w}\|_2^2. \quad (1.12)$$

It can be observed that this is very similar to the objective function that LASSO uses. The only difference is that STLSQ utilizes an l_2 regularization instead of an l_1 . However, there is one more important difference. The algorithm works by finding the optimum weights for the objective function, then masking out every coefficient that is lower than a set threshold. Afterward, it iterates these steps for the non-masked-out coefficients. This iteration procedure is useful to promote sparsity in the learned model.

1.3 The aim of this thesis

The Thoedorsen model can accurately describe the unsteady aerodynamics of an airfoil at small angles of attack. However it fails, when considering large amplitude vibrations due to stall. For these cases, CFD simulations can accurately calculate the loads on a wing, but that is computationally expensive.

This thesis aims to provide a procedure based on machine learning methods, which can be used to obtain reduced-order models of elastic structures subjected airflow for large vibrations, using a small amount of measurement or simulation data. This approach is used to model the air loads of the 2 degrees of freedom (2-DOF) model of a flat

plate subjected to airflow. This simplified model is used to demonstrate the feasibility of the approach. Some of the identified models show auspicious results, which leads us to conclude that this approach could also be applied to more complicated problems.

Chapter 2

Data-driven aerodynamic models

Recently, data-driven modeling of aerodynamics has been a trending topic for researchers. Li et al. [29] implemented a method based on sparse regression with time-varying coefficients for the flutter of long-span suspension bridges. This method with static coefficients has been used to model the lift coefficient of an airfoil using CFD simulation data by Sun et al. [27]. In the work of Fonzi et al. [30] a 3-dimensional, coupled FEM-CFD model of morphing wings has been studied. The dynamic mode decomposition (DMD) method was extended to algebraic-differential equations and then applied to capture the dynamics of morphing wings. Model predictive control (MPC) based on the learned method has also been applied for gust load alleviation and lift force tracking.

2.1 CFD simulation

We use the simulation results of Lendvai and Lelkes [31, 32] as measurement data for the SINDy algorithm. In this section, the method is briefly summarized; more details can be found in the articles mentioned earlier.

The simulation has been carried out using the ANSYS Fluent 19.0 software, utilizing the unsteady Reynolds averaged Navier-Stokes solver with turbulence models (URANS). The turbulence model used was the shear-stress transport (SST) $k - \omega$ model, which uses the $k - \omega$ turbulence model close to the walls, and the $k - \varepsilon$ further away. This is done because the $k - \varepsilon$ model is more accurate further away from the walls, and the $k - \omega$ model has higher accuracy in the boundary layers. At the surface of the airfoil, a wall-type boundary condition was prescribed. In the circular region around the airfoil, constant inlet velocity in the x direction has been defined with 5% turbulence intensity, and the hydraulic diameter was set to be equal to the chord length. The velocity-pressure coupling has been set to coupled mode, the other schemes to second order, and double-precision numbers were utilized.

To analyze flutter, long simulation times are needed because near the critical wind velocity, it can take longer for the system to settle or blow up. Also, the maximal time step is determined by the flutter frequency, so a mesh with lower elements is practical. In

particular, the mesh contained 15700 elements, which is illustrated by Figure 2.1.

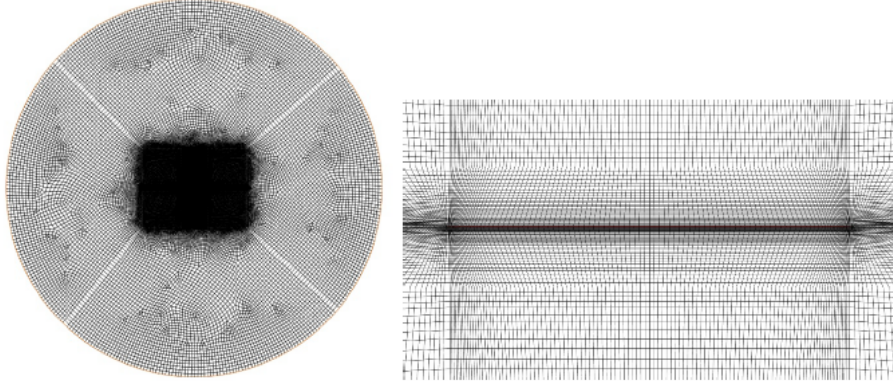


Figure 2.1. Mesh for CFD simulation

The ANSYS Fluent software provides the possibility to define user-defined functions (UDF), which can be utilized to augment the default algorithms. This feature has been used to provide the fluid-structure interaction (FSI). This has been done by implementing the equations of motion for the airfoil using explicit integration. The aerodynamic forces have been calculated from the pressure, which resulted in neglecting the shearing forces, but these were under 1% of the total forces, so the overall results have not been altered.

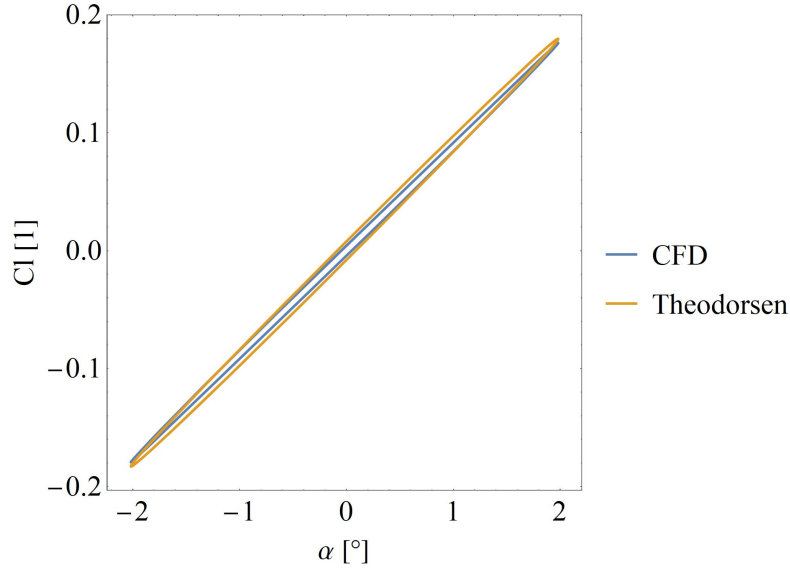
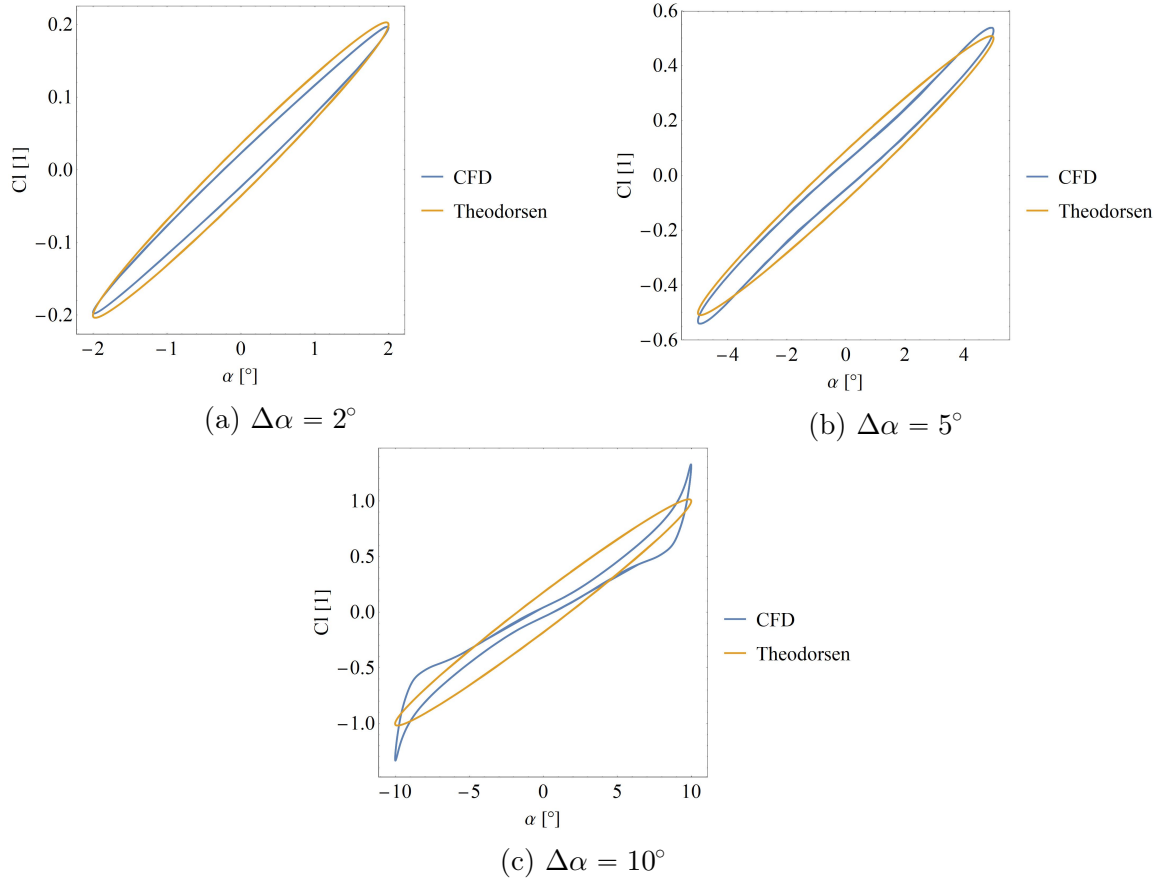
Because the motion of the airfoil can be divided into translational and rotational motion, it is also possible to handle them separately in the simulation. The translational motion is easily described by a moving reference frame. For the rotational motion, a moving mesh technique was employed.

Usually, the validation of a CFD simulation is done by examining the steady-state results. However, the examined simulation is dynamic in nature. Thus a dynamic validation is necessary. This has been done for a flat plate using the Theodorsen model, which numerous independent measurements have validated [33, 34, 35].

The simulations have been run at 10^6 Reynolds number, for this, the viscosity has been set to the required value while wind velocity and chord length were set to unity. For every period of the motion, 1000 timesteps were used. During the course of the simulations five reduced frequencies were investigated: $k \in \{0.01, 0.1, 0.2, 0.5\}$, and for the amplitude of the rotation three variations were investigated: $\Delta\alpha \in \{2^\circ, 5^\circ, 10^\circ\}$.

In Figure 2.2 the lift coefficient is plotted versus the angle of attack for $k = 0.01$, and $\Delta\alpha = 2^\circ$. It can be seen that for slow oscillation and small angles of attack, the simulation, and the Theodorsen model give the same result.

In Figure 2.3 the lift coefficient is plotted versus the angle of attack for $k = 0.1$, and varying $\Delta\alpha$ values. It can be observed that for small angles of attack, the simulation agrees with the Theodorsen model very nicely. However, a dynamic stall occurs for large angles of attack, and the Theodorsen model cannot describe this phenomenon.


 Figure 2.2. Lift coefficient for $\text{Re} = 10^6$, $U = 1$ m/s, $k = 0.01$, $\Delta\alpha = 2^\circ$

 Figure 2.3. Lift coefficient for $\text{Re} = 10^6$, $U = 1$ m/s, $k = 0.1$

When we start increasing the reduced frequency, the Theodorsen model overestimates the hysteresis, as seen in Figure 2.4. For lower angles of attack, the overall shape of the plots is about the same. However, for higher angles of attack, it can be observed that when we start decreasing the angle of attack, the lift increases by a small amount.

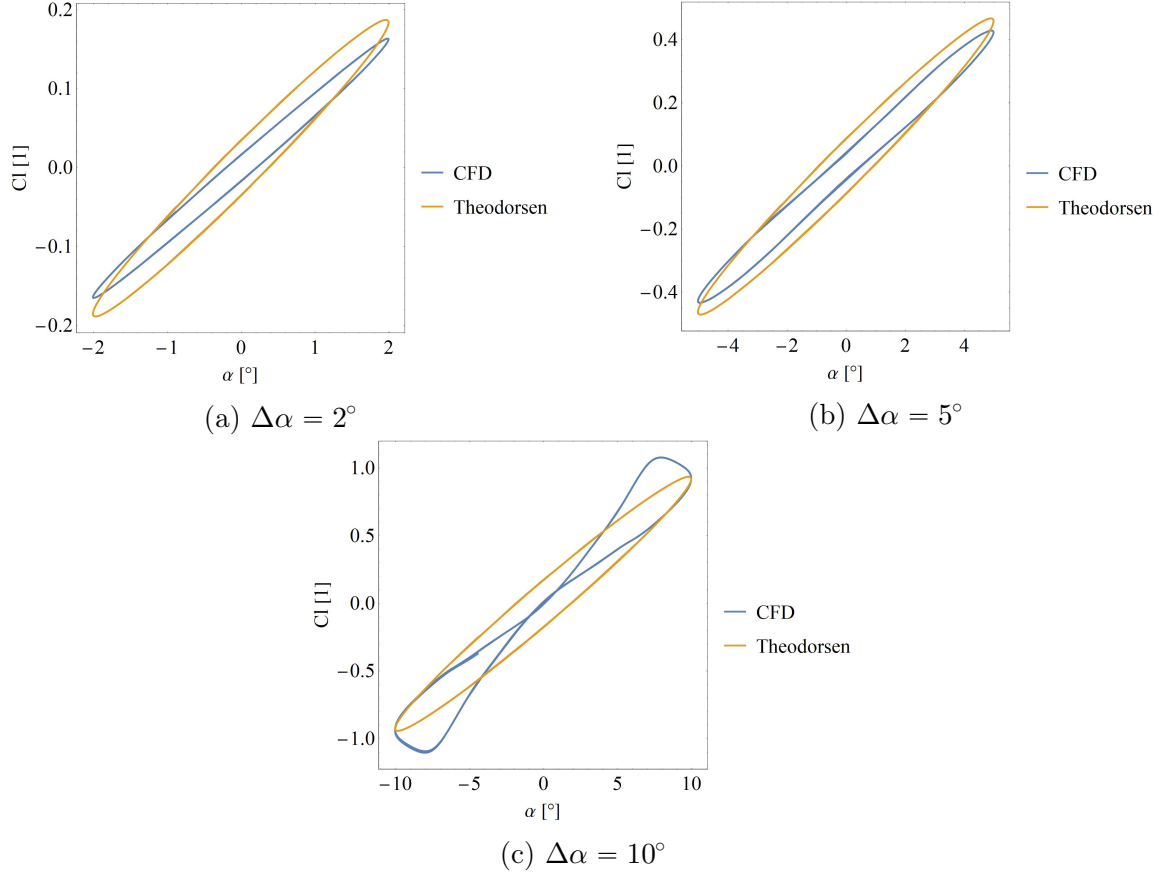


Figure 2.4. Lift coefficient for $\text{Re} = 10^6$, $U = 1 \text{ m/s}$, $k = 0.2$

If the reduced frequency is further increased to $k = 0.5$, the hysteresis in the Theodorsen model is much less, than in the CFD simulation, as evident from Figure 2.5. However, the amplitude of the lift is similar in the CFD simulation to the Theodorsen model.

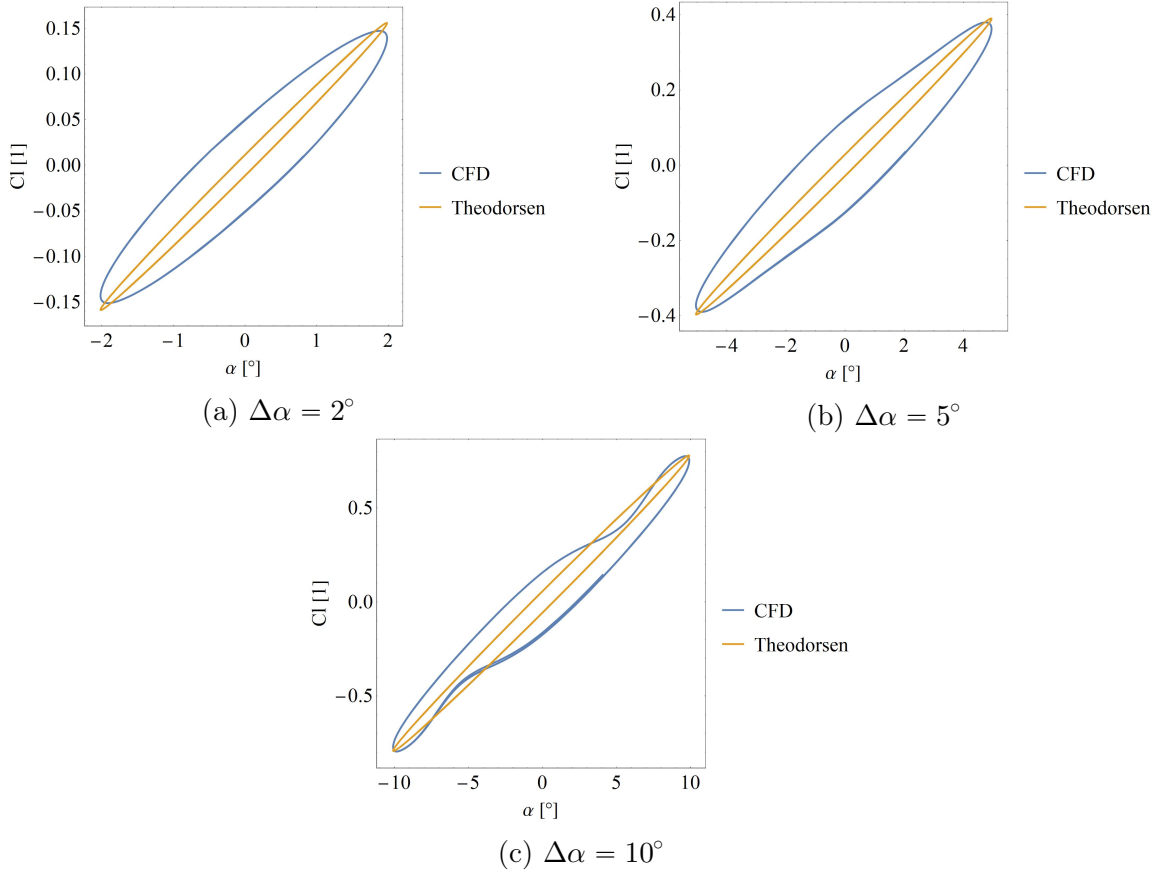


Figure 2.5. Lift coefficient for $\text{Re} = 10^6$, $U = 1 \text{ m/s}$, $k = 0.5$

2.2 Identified models

We have applied the model identification process for varying k values, and defined the following constrained motion for α

$$\alpha(t) = \Delta\alpha \cos(\omega t), \quad (2.1)$$

where $\omega = \frac{kU}{b} = 0.2$ for $\Delta\alpha = 2, 5, 10^\circ$, and $k = 0.1, 0.2, 0.5$ values, and also for $\Delta\alpha = 2^\circ$, and $k = 0.01$. In order to identify the models, we have mainly followed the process outlined in [27]. This means, that we have used SINDyc (SINDy with control), having α as the input parameter, and identified an ODE for the lift coefficient. Firstly, we used the SINDy algorithm with the LASSO, or STLSQ optimization routine to find the active coefficients in the ODE. For this we normalized the data applying the following transformation for $\alpha, \dot{\alpha}, C_l$:

$$f(x) = \frac{x}{\max(|x|)}. \quad (2.2)$$

Then we gathered the terms for which the coefficient was at least 1% of the maximum coefficient. Using these terms, we applied a least-squares regression using the non-regularized data to get the final model.

The approach requires us to find the value of the regularization parameter λ in the

objective function, which balances the model complexity with the accuracy. The smaller the value, the more accurate the model will be. However, complexity also increases, which may lead to overfitting. So it is necessary to find a value for each case, which leads to an accurate model, that can generalize. We used a simple approach to plot the number of terms and the logarithm of the mean squared error (MSE). In the following, we will call this the error-term graph. This graph helped us choose a value that would avoid overfitting while giving an accurate model. In order to help the understanding of this method, the graph for $k = 0.5$, $\Delta\alpha = 10^\circ$ is shown in Figure 2.6. The blue bars show the number of terms in the normalized model on the left-hand side, while the red line shows the logarithm of the MSE on the right. It can be observed that accuracy worsens, as λ increases, while the model gets simpler. The Figure also shows that increasing λ more than about 11 does not result in any changes to the identified model.

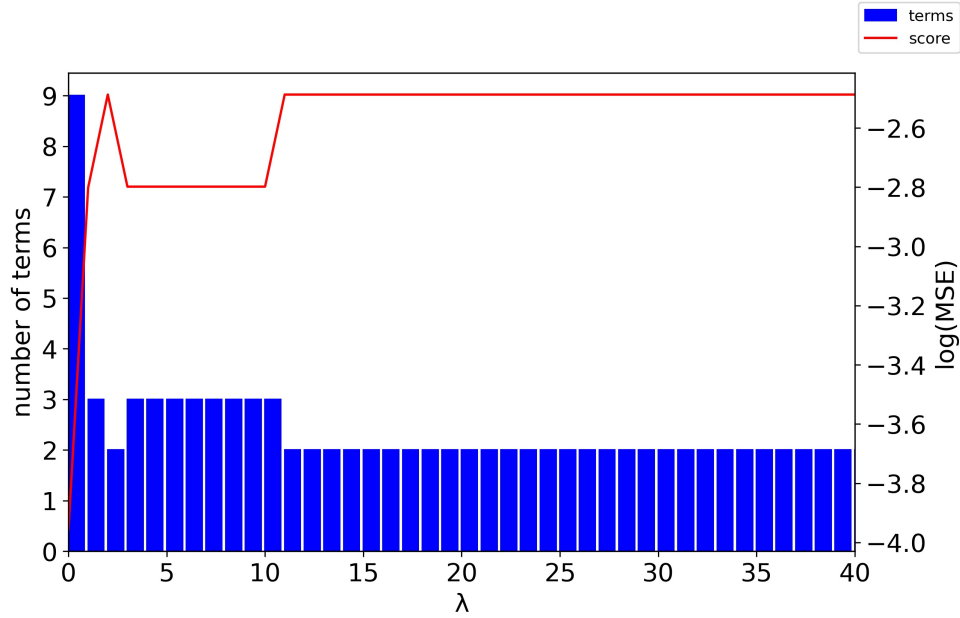


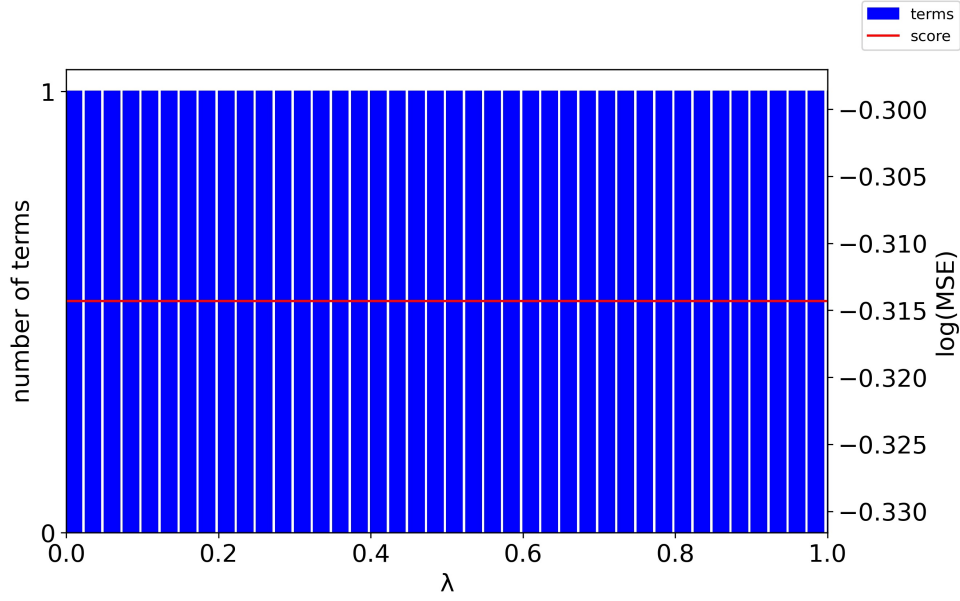
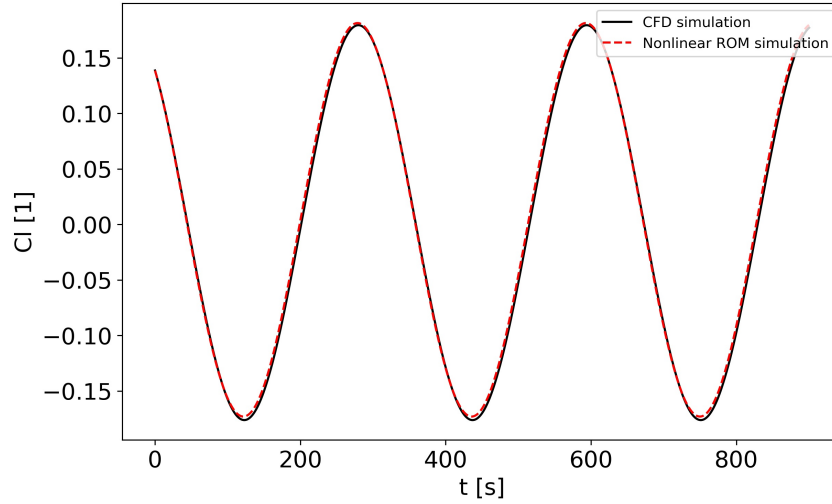
Figure 2.6. Error-term graph for $k = 0.5$ and $\Delta\alpha = 5^\circ$

We have written the model equations using a lexicographic ordering, using the following order of the state variables: α , $\dot{\alpha}$, Cl . Thus the general form of the model can be written as

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta) = c_1(k, \Delta\alpha)\alpha + c_2(k, \Delta\alpha)\dot{\alpha} + c_3(k, \Delta\alpha)C_l + c_4(k, \Delta\alpha)\alpha^2 + c_5(k, \Delta\alpha)\alpha\dot{\alpha} \dots \quad (2.3)$$

In order to be able to reference the models, they have been named as $M_{k,\Delta\alpha}$, so for example the identified model for $k = 0.5$, $\Delta\alpha = 10^\circ$ is referred to by $M_{0.5,10}$

For the $k = 0.01$, and $\Delta\alpha = 2^\circ$ case, we have used polynomials up to 3rd degree without interaction, and used the STLSQ optimizer with a threshold of 0.01. The error-term graph is illustrated in Figure 2.7. We have chosen $\lambda = 0.2$, but as seen in the graph,


 Figure 2.7. Error-term graph for $k = 0.01$ and $\Delta\alpha = 2^\circ$

 Figure 2.8. Absolute model for $k = 0.01$ and $\Delta\alpha = 2^\circ$

any value between 0 and 1 would result in the following simple model

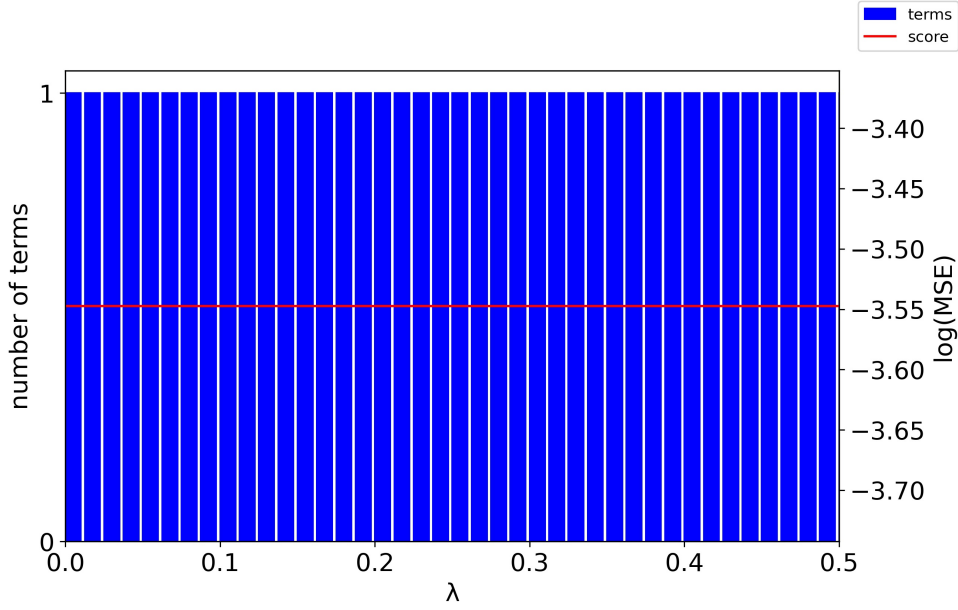
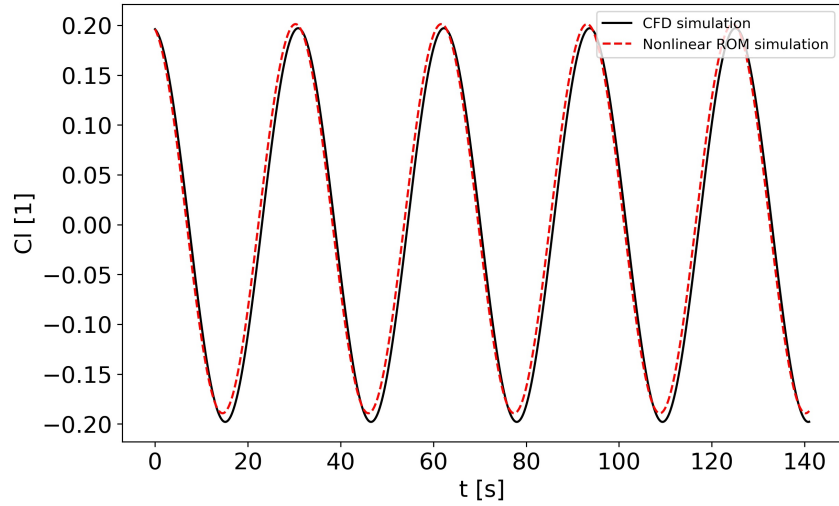
$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.4)$$

This model is shown in Figure 2.8

When $k = 0.1$, for the $\Delta\alpha = 2^\circ$ case, we have used polynomials up to 3rd degree without interaction, and used the STLSQ optimizer with a threshold of 0.1. The error-term graph is illustrated in Figure 2.9. We have chosen $\lambda = 0.2$, but any λ value from the $[0, 0.5]$ interval would have resulted in the following simple model

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.5)$$

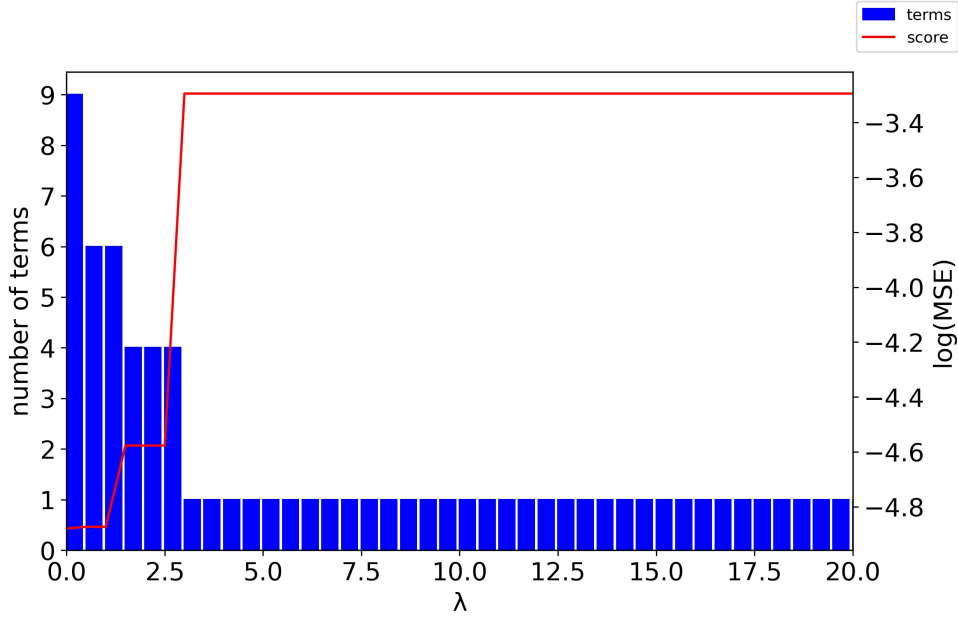
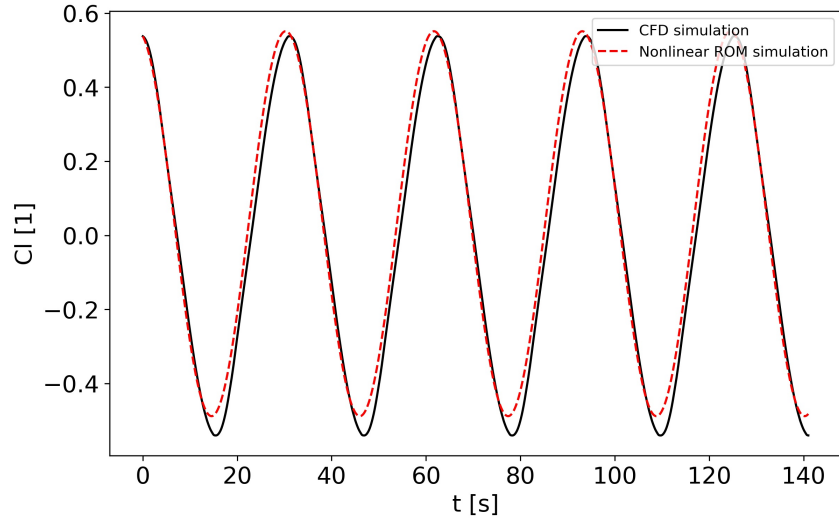
This model is shown in Figure 2.10


 Figure 2.9. Error-term graph for $k = 0.1$ and $\Delta\alpha = 2^\circ$

 Figure 2.10. Absolute model for $k = 0.1$ and $\Delta\alpha = 2^\circ$

For $\Delta\alpha = 5^\circ$, the same feature library, and optimizer has been used as in the previous case. The error-term graph is shown in Figure 2.11. From that graph we have chosen $\lambda = 15$, as that resulted in the following simple model, that is very similar to the previous case.

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.6)$$

It can be seen in Figure 2.12, that this model agrees very well with the CFD simulation.


 Figure 2.11. Error-term graph for $k = 0.1$ and $\Delta\alpha = 5^\circ$

 Figure 2.12. Absolute model for $k = 0.1$ and $\Delta\alpha = 5^\circ$

For $\Delta\alpha = 10^\circ$, we also used the STLSQ optimizer, but with a threshold of 0.2 as we got better results this way, and also included polynomials up to 5th degree in the feature library. This was necessary due to the complexity of the model. We plot the error-term graph in Figure 2.13. We wanted to find the simplest model, that would accurately capture the dynamics because that would be able to generalize the most. So we have chosen $\lambda = 6$, and got the following 8-term model:

$$\begin{aligned} \dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = & c_1(k, \Delta\alpha)\alpha + c_{10}(k, \Delta\alpha)\alpha^3 + c_{35}(k, \Delta\alpha)\alpha^5 + \\ & + c_2(k, \Delta\alpha)\dot{\alpha} + c_{15}(k, \Delta\alpha)\dot{\alpha}^3 + c_{49}(k, \Delta\alpha)\dot{\alpha}^5 + c_3(k, \Delta\alpha)C_l + c_{54}(k, \Delta\alpha)C_l^5. \end{aligned} \quad (2.7)$$

We plot this model in in Figure 2.14, it can be observed that the CFD simulation has very sharp turnarounds at the top and the bottom, and also a dip in the middle, this is

the reason, that a more complex model is necessary in this case.

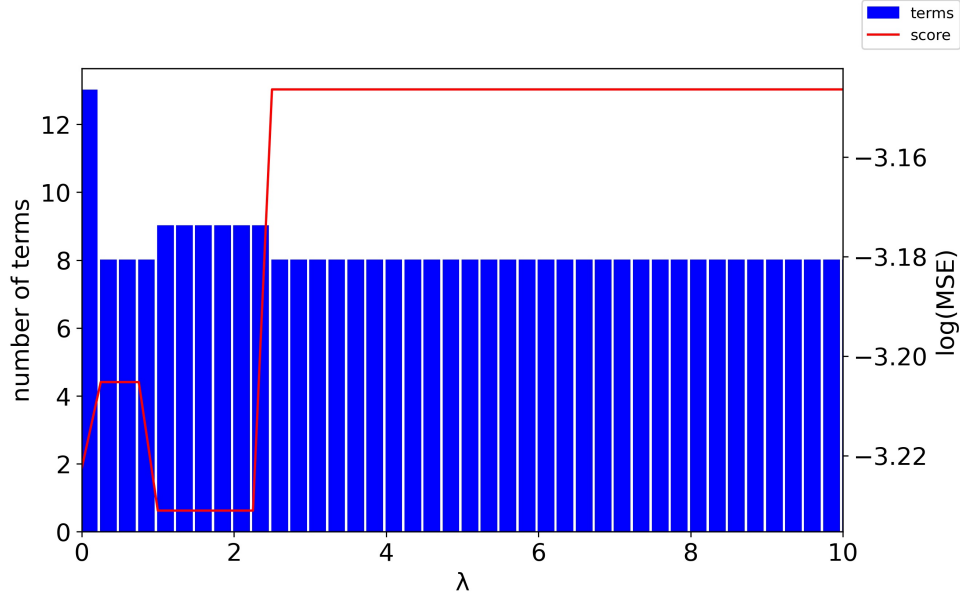


Figure 2.13. Error-term graph for $k = 0.1$ and $\Delta\alpha = 10^\circ$

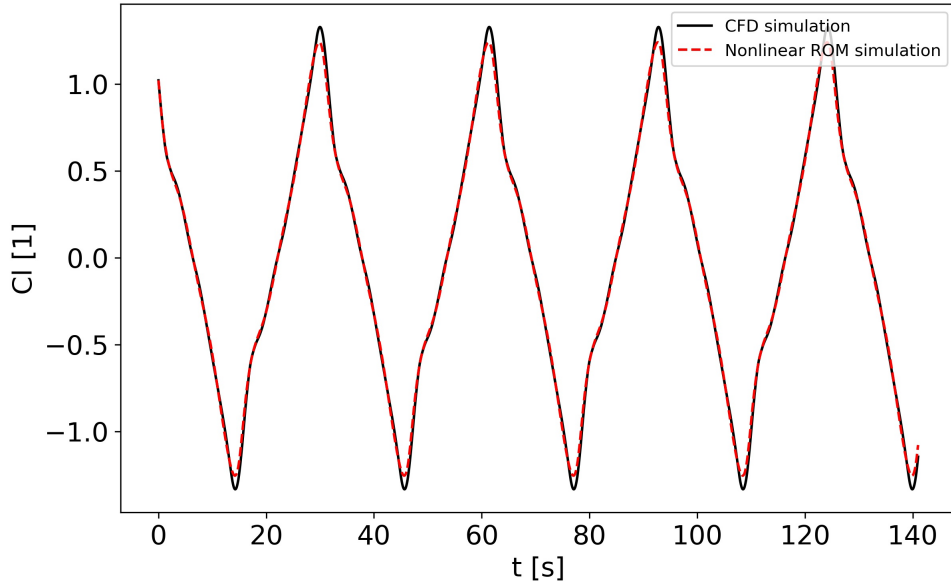
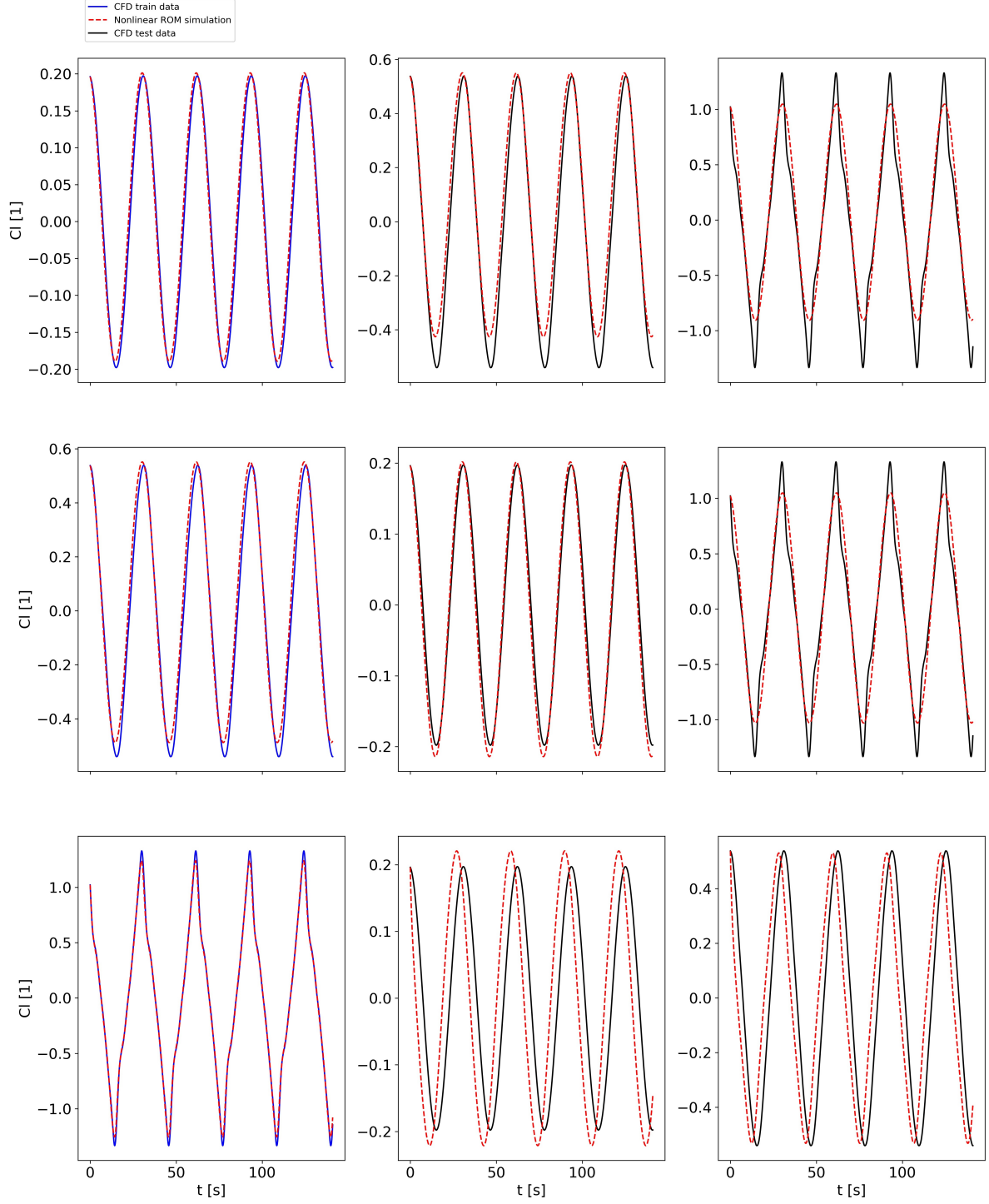


Figure 2.14. Absolute model for $k = 0.1$ and $\Delta\alpha = 10^\circ$

A comparison of these three models for the other two $\Delta\alpha$ values was made. It is illustrated in Figure 2.15. In the first column, the training data can be seen, while in the next two columns we show the application of the model for the two other $\Delta\alpha$ values in ascending order. It can be observed that the model trained on the lower $\Delta\alpha$ values does not provide a good fit for the higher values. However the model trained on $\Delta\alpha = 10^\circ$ has a poorer fit on the two lower $\Delta\alpha$ values, but generally still fits the data best.


 Figure 2.15. Comparisons of the three models for $k = 0.1$

When $k = 0.2$ for the $\Delta\alpha = 2^\circ$ case, we have used polynomials up to 3rd degree without interaction and used the STLSQ optimizer with a threshold of 0.2. The error-term graph is illustrated in Figure 2.16. We have chosen $\lambda = 0.2$ because that resulted in a pretty simple model, which accurately captured the dynamics of the system. We got the absolute model in the following form:

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.8)$$

This model is shown in Figure 2.17

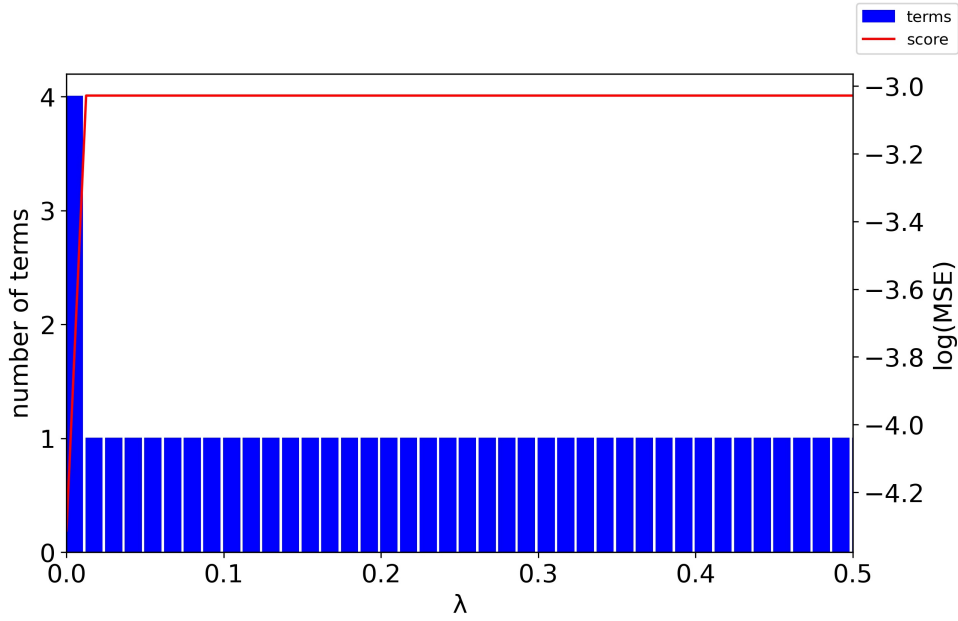


Figure 2.16. Error-term graph for $k = 0.2$ and $\Delta\alpha = 2^\circ$

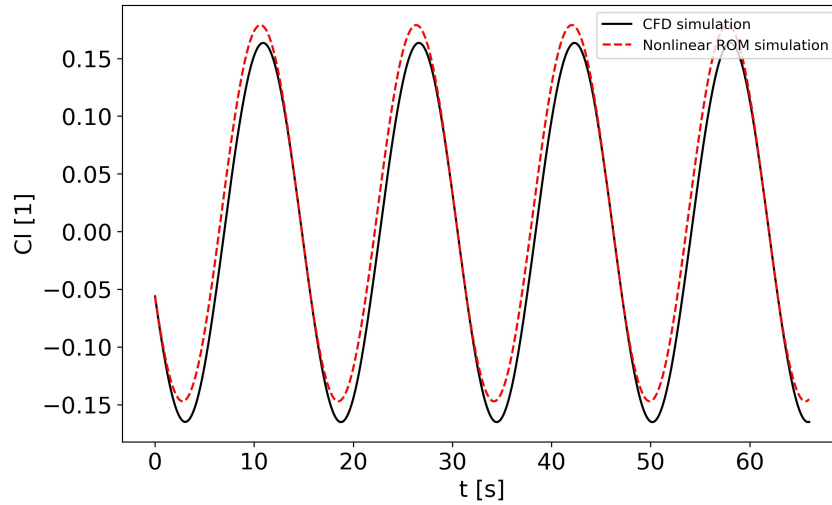
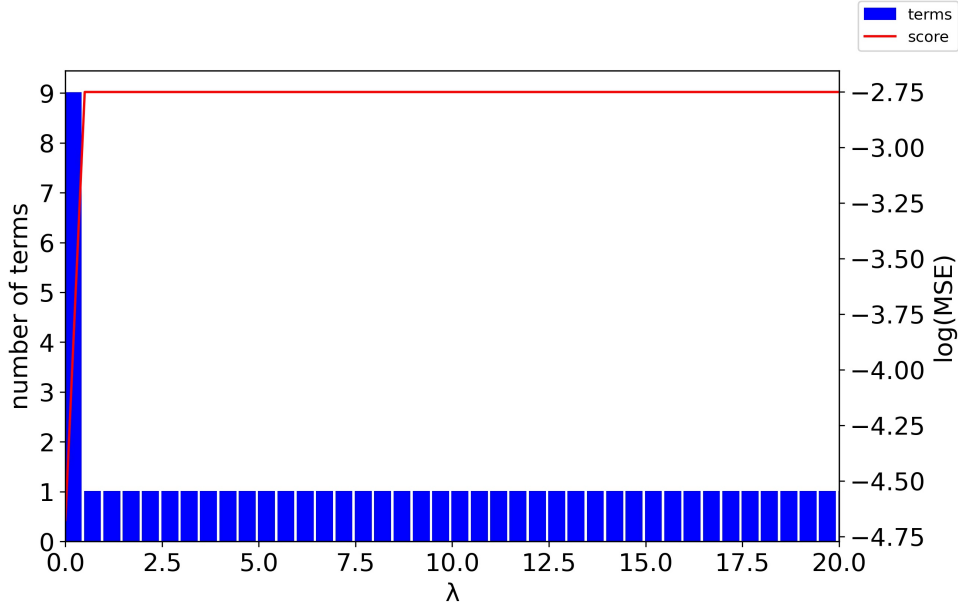
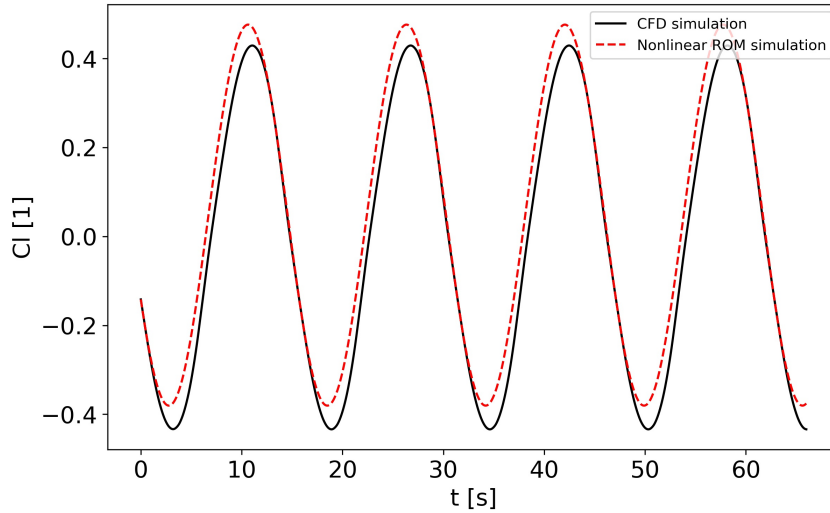


Figure 2.17. Absolute model for $k = 0.2$ and $\Delta\alpha = 2^\circ$

For $\Delta\alpha = 5^\circ$, the same feature library, and optimizer has been used as in the previous case. The error-term graph is shown in Figure 2.18. From that graph we have chosen $\lambda = 15$, but as seen from the graph could have chosen any value that is a bit larger than 0, and still get the following simple model

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.9)$$

It can be seen in Figure 2.19, that this model agrees very well with the CFD simulation.


 Figure 2.18. Error-term graph for $k = 0.2$ and $\Delta\alpha = 5^\circ$

 Figure 2.19. Absolute model for $k = 0.2$ and $\Delta\alpha = 5^\circ$

It can be observed, that the identified model overpredicts the CFD results for positive angles of attack, and underpredicts by about the same amount for negative angles of attack. The cause of this is that as the identified model only contains the coefficient of $\dot{\alpha}$, it cannot describe hysteresis, as shown in Figure 2.20. However, apart from that, it is a perfect fit for the CFD data.

For $\Delta\alpha = 10^\circ$, the STLSQ optimizer didn't give us satisfactory results, so we have instead used the LASSO optimizer and included interaction in the feature library. This was necessary due to the complexity of the model. We plot the error-term graph in Figure 2.21. We wanted to find the simplest model, that would accurately capture the dynamics because that would be able to generalize the most. So we have chosen $\lambda = 0.03$, and got

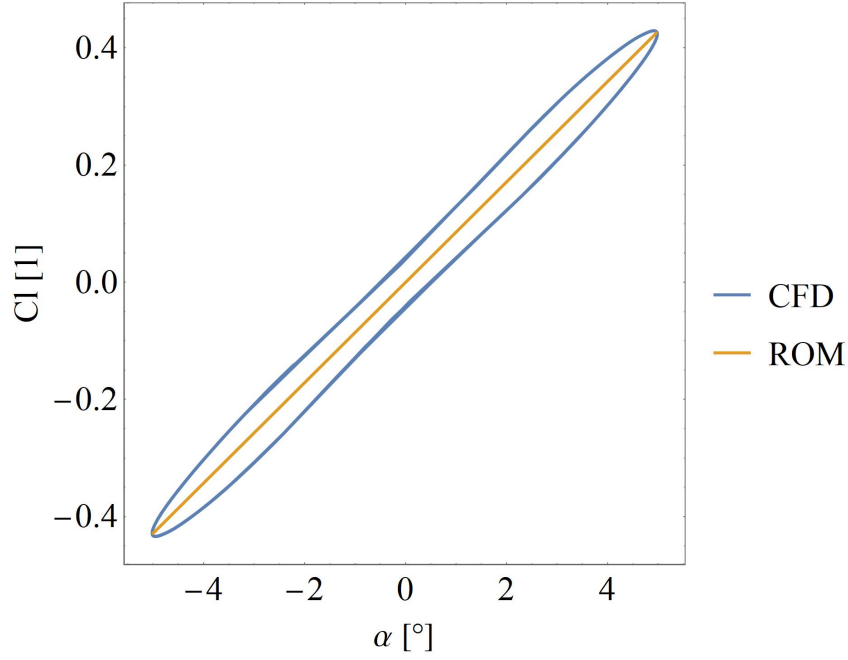


Figure 2.20. Comparison of the identified model with the CFD simulation for $k = 0.2$ and $\Delta\alpha = 5^\circ$

the following 2-term model:

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_{10}(k, \Delta\alpha)\alpha^3 + c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.10)$$

It can be seen in Figure 2.22, that the agreement with the CFD simulation is lower, especially at the extrema, where the derivatives are large, than in the previous two cases. However, considering the complexity of this case, it can still be regarded as a very accurate model.

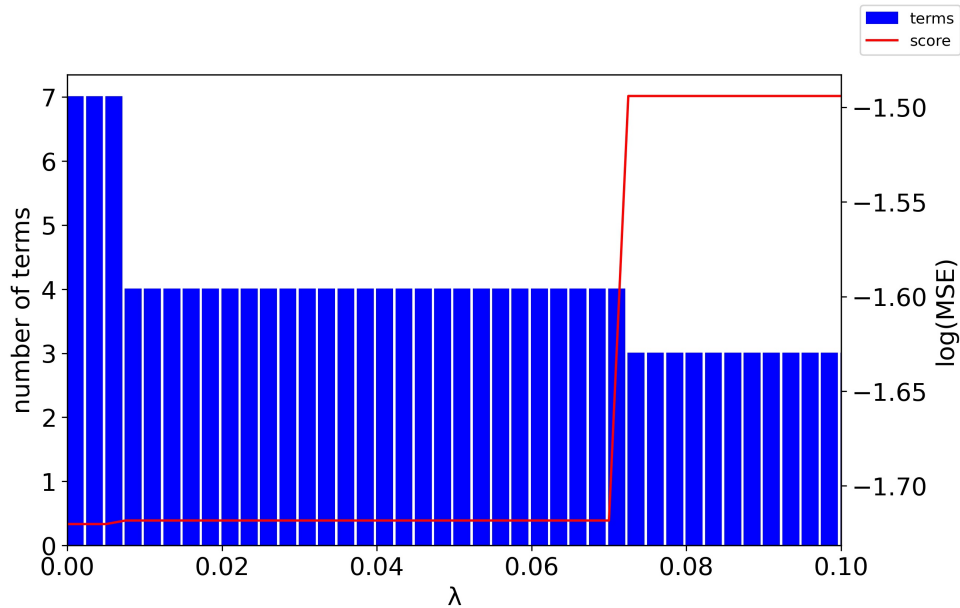


Figure 2.21. Error-term graph for $k = 0.2$ and $\Delta\alpha = 10^\circ$

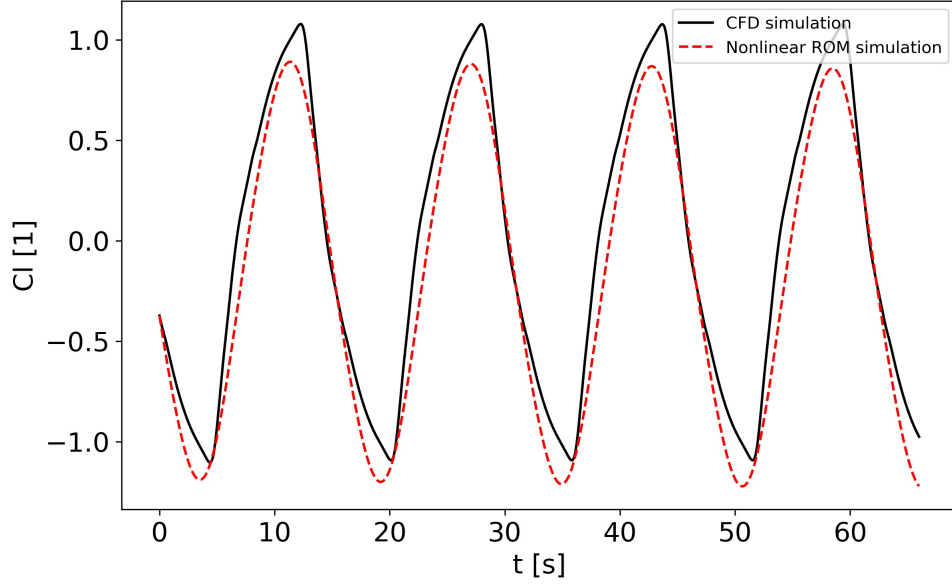
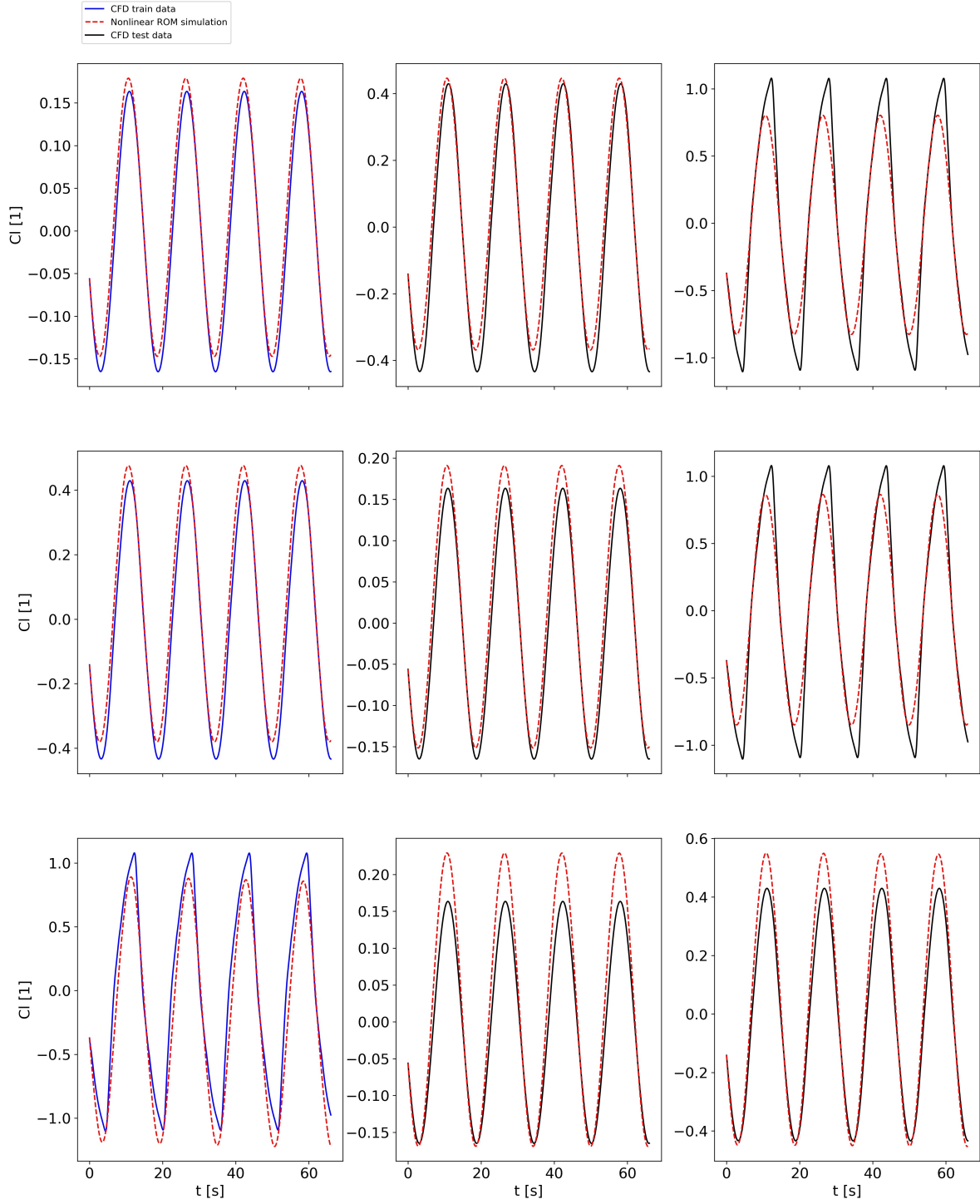


Figure 2.22. Absolute model for $k = 0.2$ and $\Delta\alpha = 10^\circ$

A comparison of these three models for the other two $\Delta\alpha$ values was made. This is illustrated in Figure 2.23. In the first column, the training data can be seen, while in the next two columns we show the application of the model for the two other $\Delta\alpha$ values in ascending order. It can be observed that if we apply the model trained on the lower two $\Delta\alpha$ values to the value of the other, we can observe a good fit. However the model trained on $\Delta\alpha = 2^\circ$ does not provide a good fit for $\Delta\alpha = 10^\circ$. In contrast, the middle model provides a reasonable fit for that case also. The model trained on $\Delta\alpha = 10^\circ$ has a poorer fit on the two lower $\Delta\alpha$ values, but it does not fit its training data perfectly, so it is debatable, whether the middle or the lower model has the best overall fit.


 Figure 2.23. Comparisons of the three models for $k = 0.2$

When $k = 0.5$ for the $\Delta\alpha = 2^\circ$ case, we have used polynomials up to 3rd degree without interaction and used the STLSQ optimizer with a threshold of 0.2. The error-term graph is illustrated in Figure 2.24. We have chosen $\lambda = 0.2$ because that resulted in a pretty simple model, which accurately captured the dynamics of the system. We got the absolute model in the following form:

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_1(k, \Delta\alpha)\alpha + c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.11)$$

This model is shown in Figure 2.25

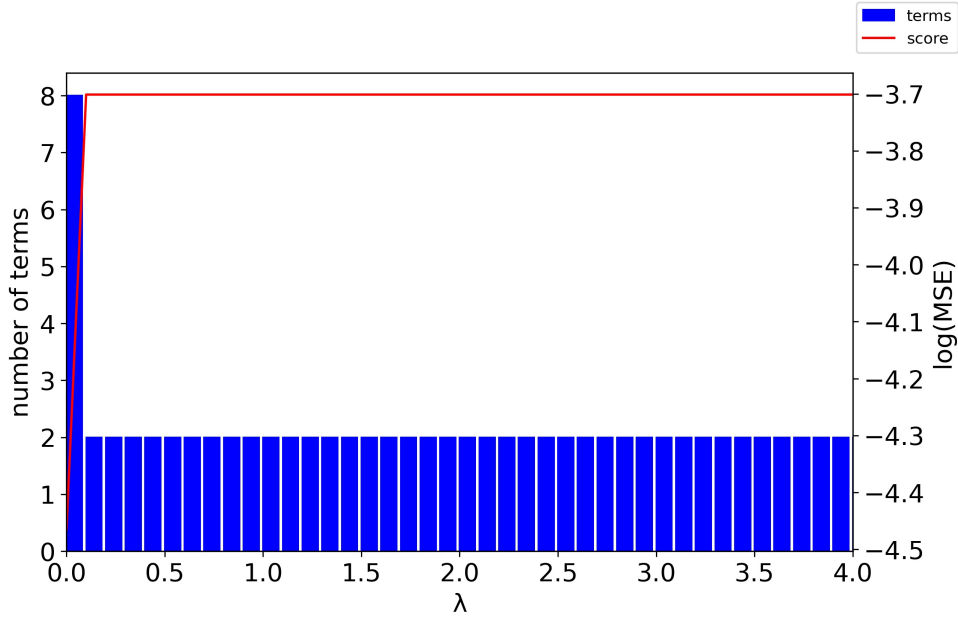


Figure 2.24. Error-term graph for $k = 0.5$ and $\Delta\alpha = 2^\circ$

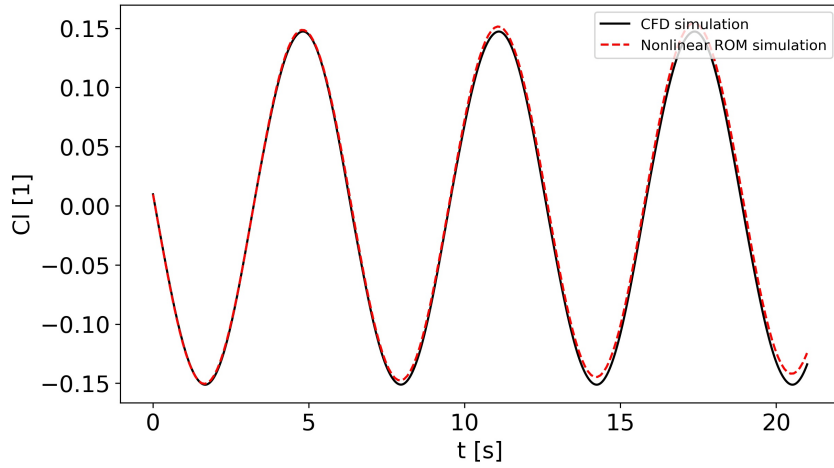
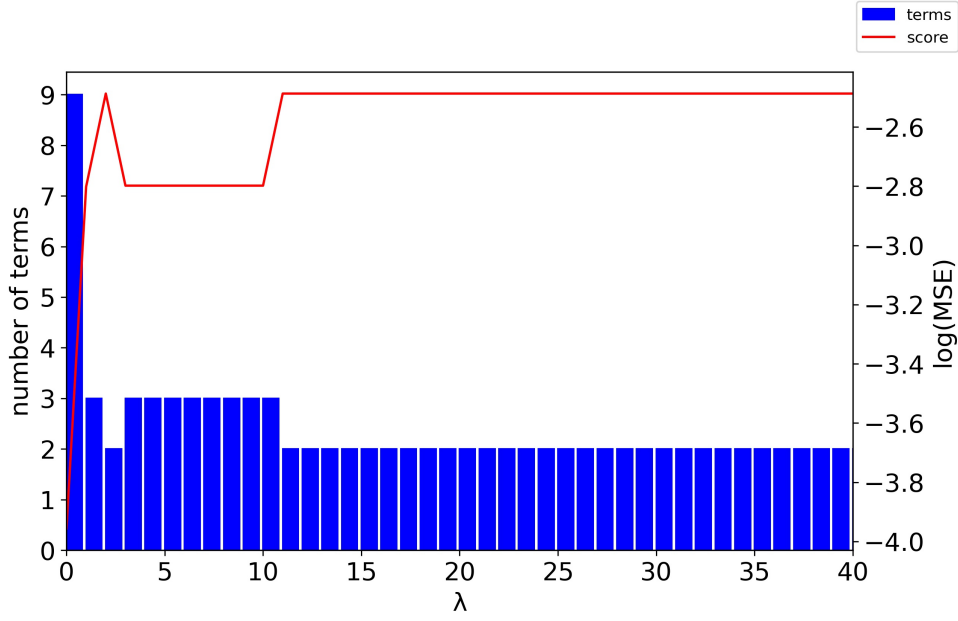
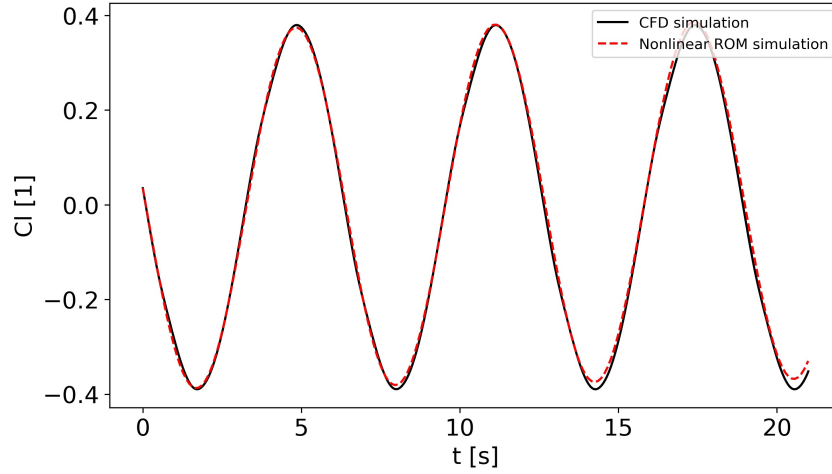


Figure 2.25. Absolute model for $k = 0.5$ and $\Delta\alpha = 2^\circ$

For $\Delta\alpha = 5^\circ$, the same feature library, and optimizer has been used as in the previous case. The error-term graph is shown in Figure 2.26. From that graph we have chosen $\lambda = 15$, as that resulted in the following simple model, that is very similar to the previous case.

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_1(k, \Delta\alpha)\alpha + c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.12)$$

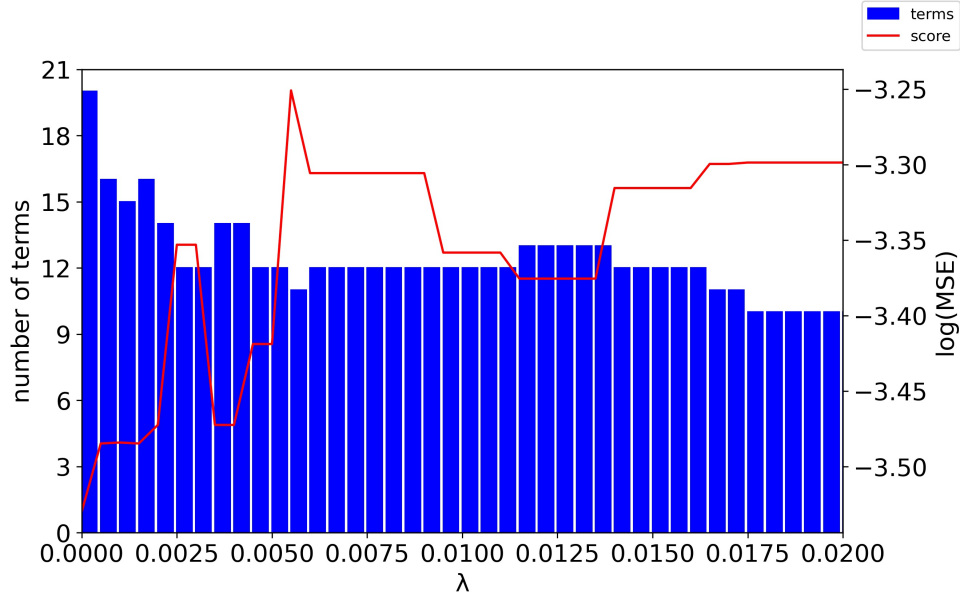
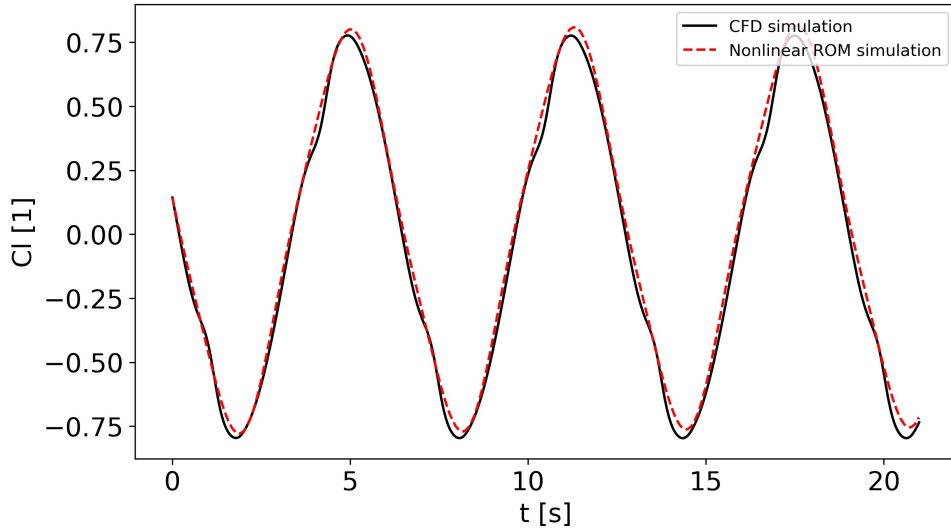
It can be seen in Figure 2.27, that this model agrees very well with the CFD simulation.


 Figure 2.26. Error-term graph for $k = 0.5$ and $\Delta\alpha = 5^\circ$

 Figure 2.27. Absolute model for $k = 0.5$ and $\Delta\alpha = 5^\circ$

For $\Delta\alpha = 10^\circ$, the STLSQ optimizer didn't give us satisfactory results, so we have instead used the LASSO optimizer and included interaction in the feature library. This was necessary due to the complexity of the model. We plot the error-term graph in Figure 2.28. We wanted to find the simplest model, that would accurately capture the dynamics because that would be able to generalize the most. So we have chosen $\lambda = 0.015$, and got the following 3-term model:

$$\dot{C}_l(\alpha, \dot{\alpha}, C_l, k, \Delta\alpha) = c_1(k, \Delta\alpha)\alpha + c_{11}(k, \Delta\alpha)\alpha^2\dot{\alpha} + c_2(k, \Delta\alpha)\dot{\alpha}. \quad (2.13)$$

It can be seen in Figure 2.29, that although the agreement with the CFD simulation is lower, than in the previous two cases, considering the complexity of this case, it can still be regarded as a very accurate model.

Figure 2.28. Error-term graph for $k = 0.5$ and $\Delta\alpha = 10^\circ$ Figure 2.29. Absolute model for $k = 0.5$ and $\Delta\alpha = 10^\circ$

We have also done a comparison of these three models for the other two $\Delta\alpha$ values. This is illustrated in Figure 2.30. In the first column, the training data can be seen, while in the next two columns we show the application of the model for the two other $\Delta\alpha$ values in ascending order. It can be observed that if we apply the model trained on the lower two $\Delta\alpha$ values to the value of the other, we can observe a good fit. However these models don't provide a good fit for $\Delta\alpha = 10^\circ$. The model trained on $\Delta\alpha = 10^\circ$ has a poorer fit on the two lower $\Delta\alpha$ values, but it still has the best overall fit. For this value of k , models trained on multiple time series were also investigated, but did not perform better, than the model trained for $\Delta\alpha = 10^\circ$.

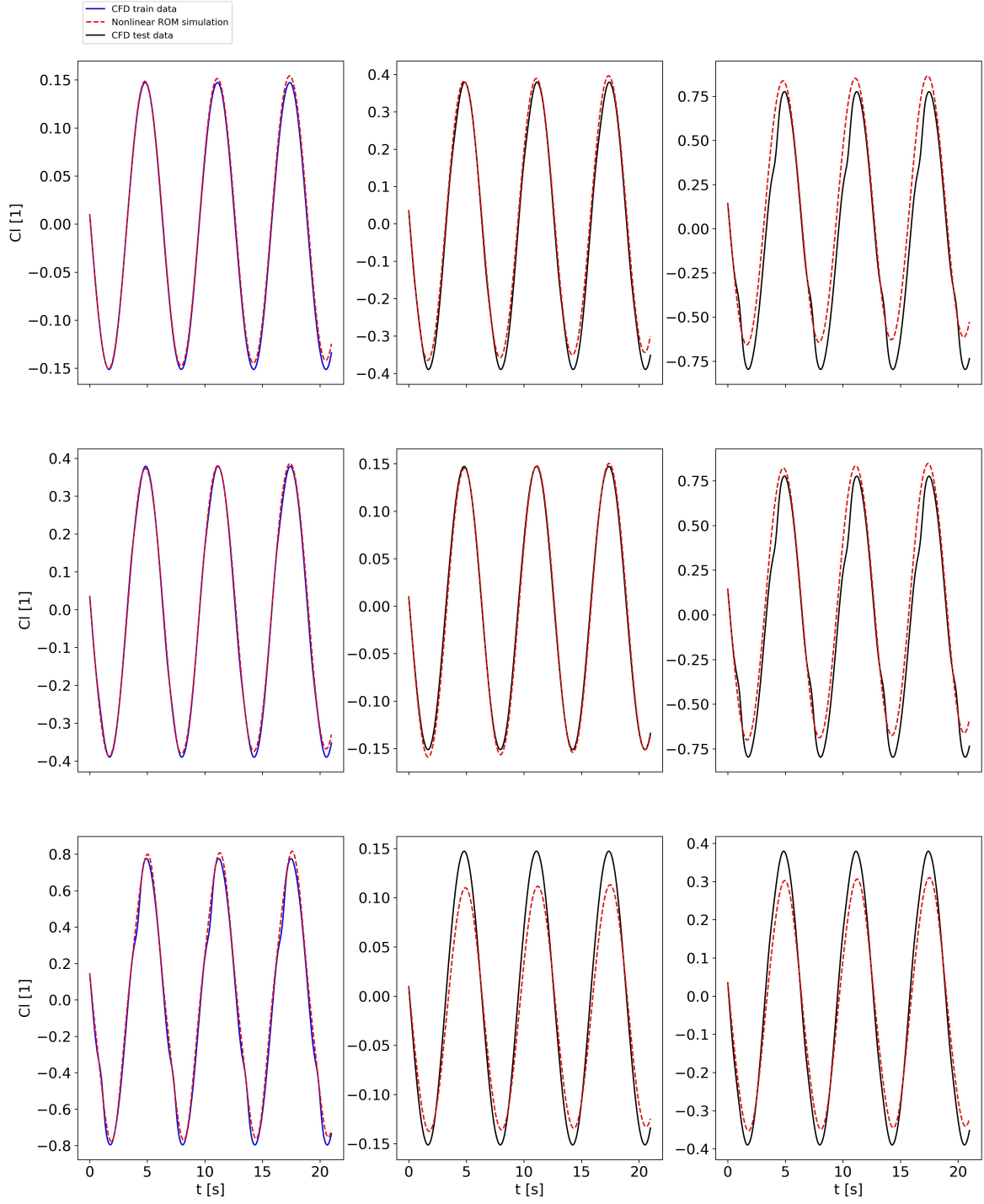


Figure 2.30. Comparisons of the three models for $k = 0.5$

k	$\Delta\alpha$	c_1	c_2	c_3	c_{10}	c_{11}	c_{15}	c_{35}	c_{49}	c_{54}
0.01	2°	0	-5.08	0	0	0	0	0	0	0
0.1	2°	0	5.59	0	0	0	0	0	0	0
0.1	5°	0	5.95	0	0	0	0	0	0	0
0.1	10°	6.26	36.2	1.47	80.23	0	-54757	-1746.12	$2.5 \cdot 10^7$	0.166
0.2	2°	0	4.67	0	0	0	0	0	0	0
0.2	5°	0	4.91	0	0	0	0	0	0	0
0.2	10°	0	5.7	0	24.2	0	0	0	0	0
0.5	2°	-1.44	4.01	0	0	0	0	0	0	0
0.5	5°	-1.31	4.14	0	0	0	0	0	0	0
0.5	10°	-0.79	3.4	0	0	104.44	0	0	0	0

Table 2.1. Coefficients of the identified models

The coefficients used to express the models can be found in Table 2.1.

Chapter 3

Application of the aerodynamic model on a the two-degree-of-freedom system

In this chapter, we apply the identified aerodynamic models to a 2-DOF aeroelastic system. In the model, α describes the pitch (positive clockwise), while h is the plunge displacement (positive downward). The mass is m , the moment of inertia around the pitching axis is I_α , the semichord of the wing is b . The spring, and the damping coefficient for the plunge DOF is k_h, c_h , respectively, while these for the pitch are k_α, c_α , and S_α is the mass unbalance [36].

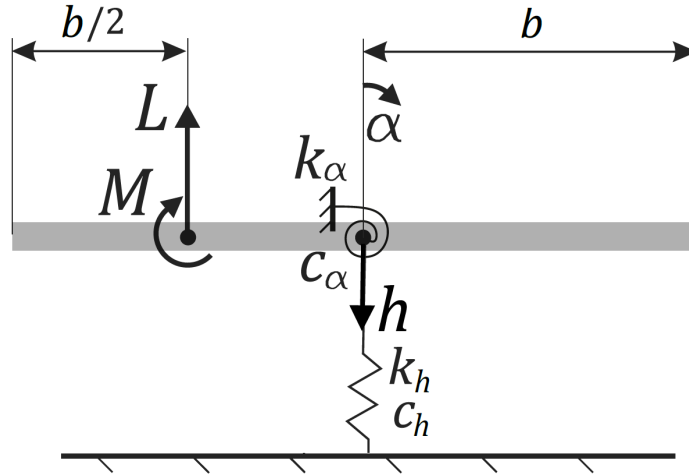


Figure 3.1. 2-DOF flat plate

Equations of motion for a simple 2-DOF flat plate (see Figure 3.1) can be written generally as [11]:

$$m\ddot{h} + c_h\dot{h} + k_h h + S_\alpha \ddot{\alpha} = -L, \quad (3.1)$$

$$S_\alpha \ddot{h} + I_\alpha \ddot{\alpha} + c_\alpha \dot{\alpha} + k_\alpha \alpha = M, \quad (3.2)$$

where L and M are the aerodynamic the lift and moment, respectively. They can be

calculated from the pressure distribution on the surface, i.e.

$$L = \int p d\mathbf{A}, \quad (3.3)$$

$$M = \int p x d\mathbf{A}, \quad (3.4)$$

where p is the pressure distribution, x is the distance from the point where the lift is applied.

Some of the identified models are now applied to predict flutter for similar parameter values as in the training data. We used Eq. (3.2) in combination with the identified models described in the previous chapters to simulate flutter. The time series were then compared to CFD simulations. It needs to be noted that the lift in this model also depends on the plunge state, but when training the SINDy models, we did not take this into account. Due to this, we used Theodorsen's lift theory to model the plunge parts of the lifts.

We have also added a cubic spring to the angular DOF to keep the vibrations inside the applicability region of the model and the better approximate reality in the process. Thus we can transform (3.1), (3.2) into the following form:

$$\begin{bmatrix} m & 0 \\ 0 & I_\alpha \end{bmatrix} \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{X}''(t) + \begin{bmatrix} 2\omega_h \xi_h & 0 \\ 0 & 2\omega_\alpha \xi_\alpha \end{bmatrix} \mathbf{X}'(t) + \begin{bmatrix} \omega_h^2 & 0 \\ 0 & \omega_\alpha^2 + k_{\alpha^3} \alpha(t)^2 \end{bmatrix} \mathbf{X}(t) \right) = \begin{bmatrix} -L \\ M \end{bmatrix}, \quad (3.5)$$

where the state vector $\mathbf{X}(t)$ is

$$\mathbf{X}(t) = \begin{bmatrix} h(t) \\ \alpha(t) \end{bmatrix}. \quad (3.6)$$

k_{α^3} is the stiffness coefficient of the cubic spring, $\xi_h = \xi_m = \xi$ is the damping factor, and ω_h, ω_α are the angular natural frequencies of the h , and α degrees of freedom. We write

$$\begin{bmatrix} -L \\ M \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \rho U^2 b \left(C_l(t) + K \frac{H_1(K)}{U} h'(t) + \frac{K^2 H_4(K)}{B} h(t) \right) \\ \frac{1}{2} \rho U^2 b^2 \left(\frac{C_l(t)}{4} + K \frac{A_1(K)}{U} h'(t) + \frac{K^2 A_4(K)}{B} h(t) \right) \end{bmatrix}, \quad (3.7)$$

where $K = 2k$, H_1, H_4, A_1, A_4 come from the Theodorsen function, and $C_l(t)$ is obtained by adding the identified model from SINDy to (3.5). We solve this system of differential equations using the Mathematica software and show plots of the time series for increasing U values, and also a bifurcation diagram. For the numerical solution, we used the following initial conditions

$$\begin{bmatrix} h(0) \\ \alpha(0) \\ C_l(0) \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} \dot{h}(0) \\ \dot{\alpha}(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \quad (3.8)$$

3.1 Model $M_{0.5,5}$

First, we apply the $M_{0.5,5}$ model. Our goal was to apply the model in similar conditions to the training set, and determine whether they can predict flutter. For this we set the natural frequency of the heave mode $f_h = 0.1$ [Hz], and for the torsional mode $f_\alpha = 0.12$ [Hz]. This results in a reduced frequency of $K = 0.7$, which is pretty close to 0.5, so we can test the generalization capability of our model. We also set $k_{\alpha^3} = 75$, and $\xi = 0.89$. The bifurcation diagram can be seen in Figure 3.2, where we show the vibration amplitude with respect to the wind velocity.

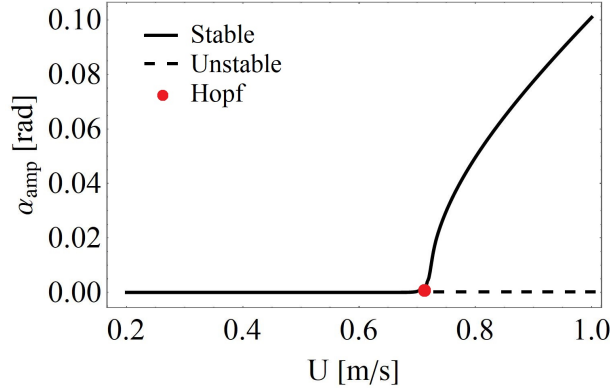


Figure 3.2. Bifurcation diagram for $M_{0.5,5}$

For wind velocities between $U = 0.7$, and $U = 0.9$ the time series for α can be seen in Figures 3.3. For $U = 1$ m/s, the time series for both DOFs were plotted in Figure 3.4. It can be seen in the bifurcation diagram, as well as in the time series, that a supercritical Hopf bifurcation occurs, thus flutter develops as the velocity is increased. We conclude that this model is able to describe flutter.

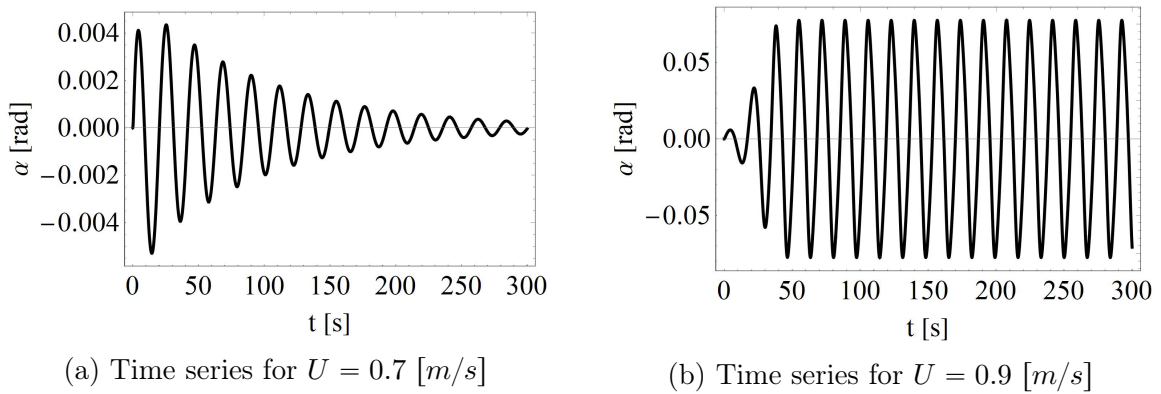
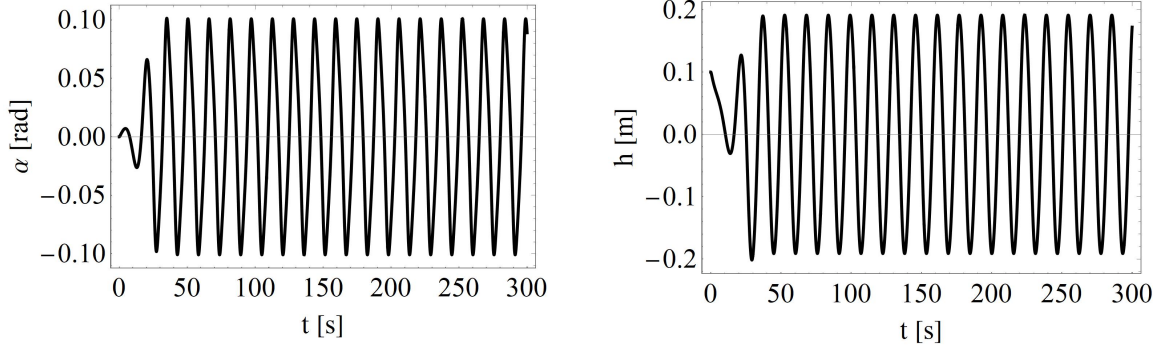
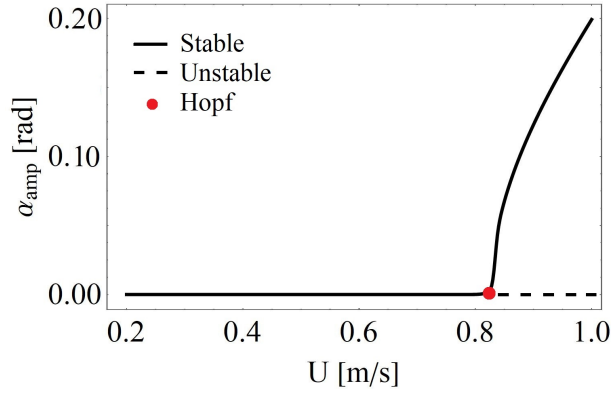


Figure 3.3. Time series of $\alpha(t)$ for $M_{0.5,5}$

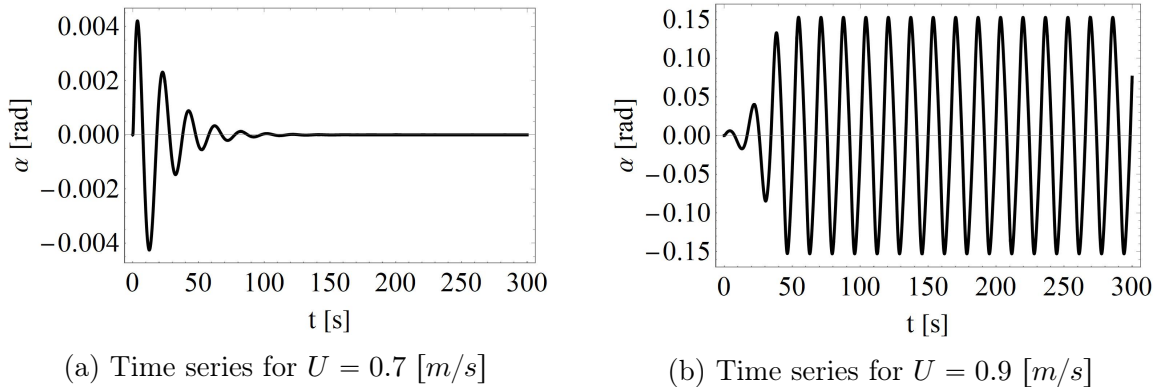

 Figure 3.4. Time series of $h(t)$, $\alpha(t)$ for $M_{0.5,5}$, $U = 1$ [m/s]

3.2 Model $M_{0.5,10}$

Now we show the application of $M_{0.5,10}$. The bifurcation diagram can be seen in Figure 3.5, where we show the vibration amplitude with respect to the wind velocity.


 Figure 3.5. Bifurcation diagram for $M_{0.5,10}$

For wind velocities between $U = 0.7$, and $U = 0.9$ the time series for α can be seen in Figures 3.6. For $U = 1$ m/s, the time series for both DOFs were plotted in Figure 3.7. It can be seen in the bifurcation diagram, as well as in the time series, that a supercritical Hopf bifurcation occurs, thus flutter develops as the velocity is increased. We conclude that this model is able to describe flutter.


 Figure 3.6. Time series of $\alpha(t)$ for $M_{0.5,10}$

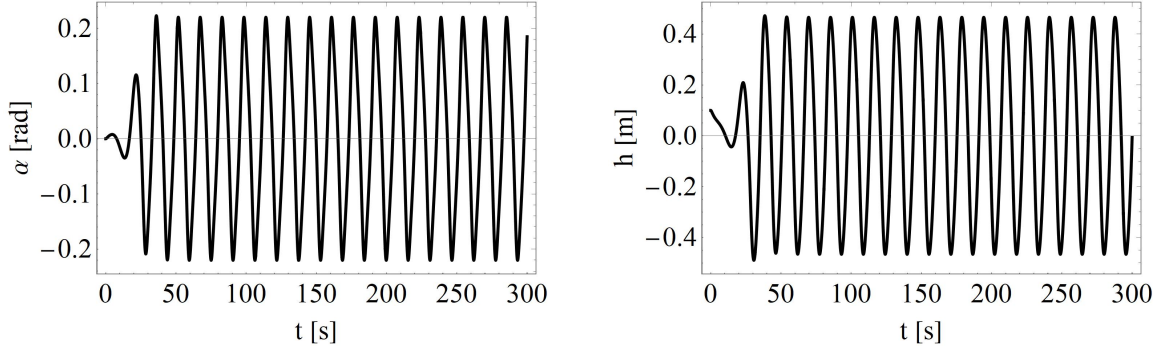


Figure 3.7. Time series of $h(t)$, $\alpha(t)$ for $M_{0.5,10}$, $U = 1$ [m/s]

3.3 Model $M_{0.1,5}$

Next we apply $M_{0.1,5}$, for this we set the natural frequency of the heave mode $f_h = 0.02$ [Hz], and for the torsional mode $f_\alpha = 0.024$ [Hz]. This results in a reduced frequency of $K = 0.14$, which is pretty close to 0.1, so we can test the generalization capability of our model. We also set $k_{\alpha^3} = 150$, and $\xi = 0.89$. The bifurcation diagram can be seen in Figure 3.8, where we show the fixed points respect to wind velocity.

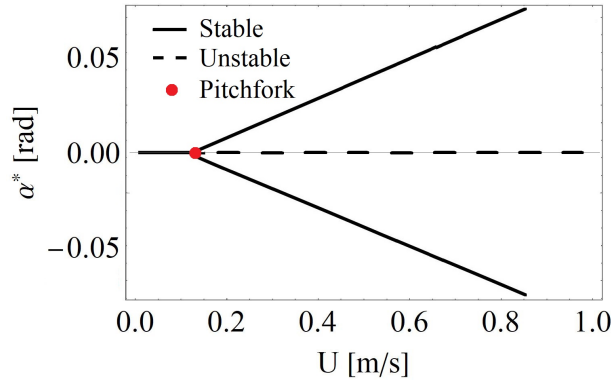
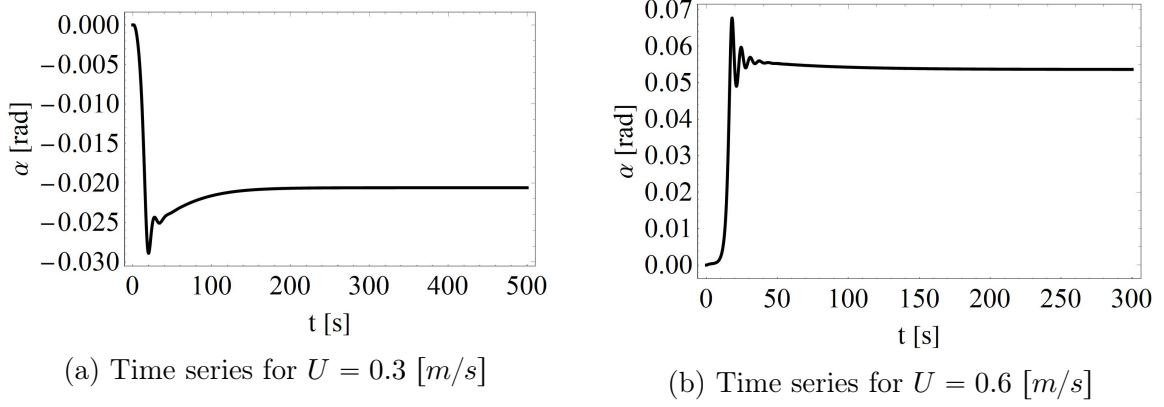
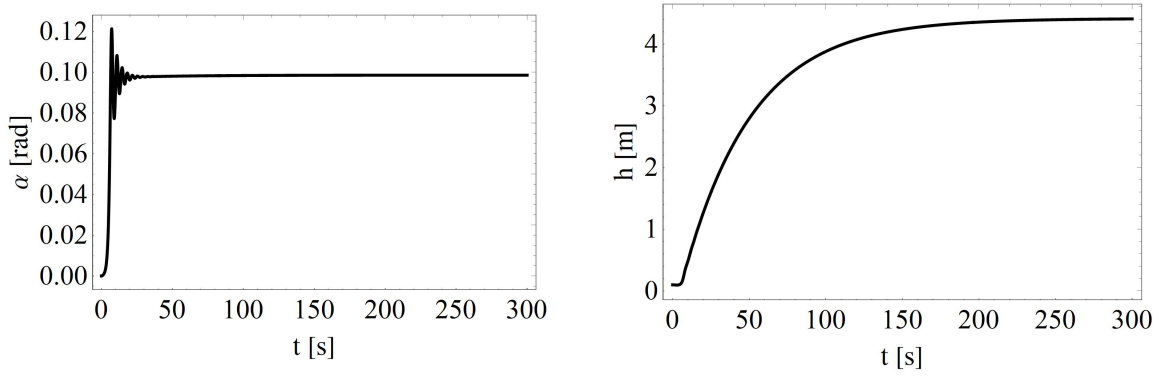


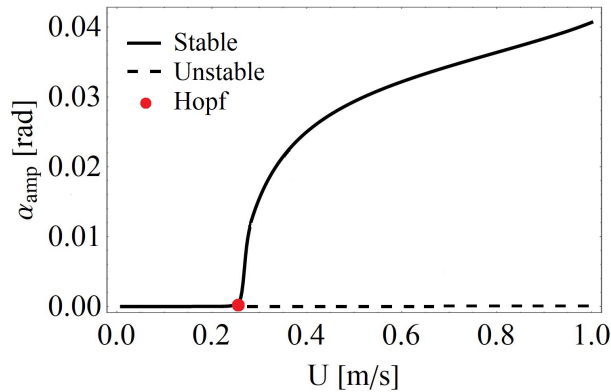
Figure 3.8. Bifurcation diagram for $M_{0.1,5}$

It can be observed on the bifurcation diagram, as well as on the time series in Figures 3.9, and 3.10, that a supercritical pitchfork bifurcation occurs, thus divergence develops as the wind velocity is increased. We can conclude that the identified model cannot describe flutter.


 Figure 3.9. Time series of $\alpha(t)$ for $M_{0.1,5}$

 Figure 3.10. Time series of $h(t)$, $\alpha(t)$ for $M_{0.1,5}$, $U = 1$ [m/s]

3.4 Model $M_{0.1,10}$

Lastly we apply $M_{0.1,10}$, for this we set the natural frequency of the heave mode $f_h = 0.02$ [Hz], and for the torsional mode $f_\alpha = 0.024$ [Hz]. This results in a reduced frequency of $K = 0.14$, which is pretty close to 0.1, so we can test the generalization capability of our model. We also set $k_{\alpha^3} = 2000$, and $\xi = 1$. The nonlinear stiffness coefficient needs to be set so high for the solutions not to diverge. The bifurcation diagram can be seen in Figure 3.11, where we show the vibration amplitude with respect to the wind velocity.


 Figure 3.11. Bifurcation diagram for $M_{0.1,10}$

It can be observed on the bifurcation diagram, as well as on the time series in Figure 3.12, that a supercritical Hopf bifurcation occurs, thus flutter develops at a relatively low wind velocity. As we increase the wind velocity even further, due to the very high nonlinear stiffness, the amplitude remains relatively small. What can be observed in this model is overfitting, which means that the model is so complex that it performs poorly outside of training data. This is also the reason for the divergent solutions in the case of smaller nonlinear stiffness values.

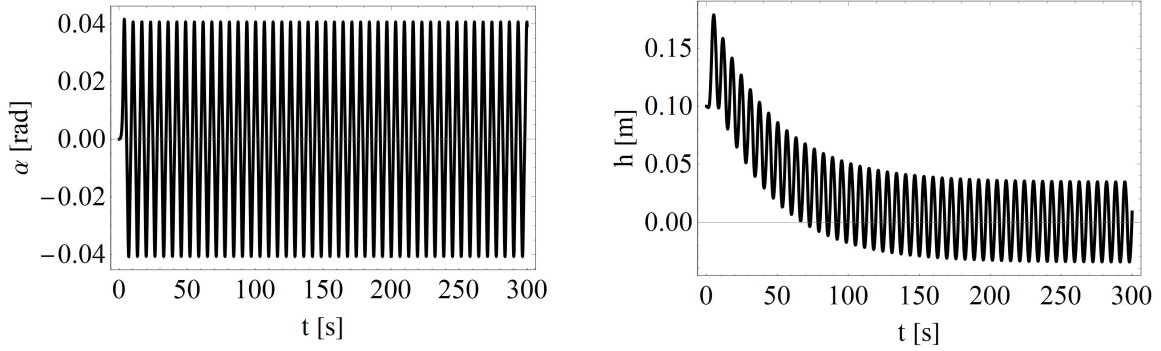


Figure 3.12. Time series of $h(t)$, $\alpha(t)$ for $M_{0.1,10}$, $U = 1$ [m/s]

We conclude that the models identified for $k = 0.5$ were able to describe the flutter phenomenon, but the others failed in this aspect except for the most complicated model, which was identified for $k = 0.1$ and $\Delta\alpha = 10^\circ$. That model, however, had such a large complexity that it could not go beyond about 2° of amplitude. Otherwise, the solution would diverge. Both models contain the coefficient for α and also for $\dot{\alpha}$. However, in the model for $k = 0.1$, and $\Delta\alpha = 5^\circ$, and also in the case of other simple models, the coefficient of α is zero. This results in these models, although being simple, but are not able to reproduce flutter. So in future works aiming to develop reduced-order models for flutter based on the procedure outlined in this thesis, it is necessary to identify simple models with low coefficient values, but they should be complex enough to contain at least the coefficients for α , and $\dot{\alpha}$.

Chapter 4

Summary

In this thesis, analytical aerodynamic models are summarized, and also the field of aeroelasticity. Afterward, data-driven modeling methods based on measurement or simulation data are introduced. Then the methodology of the CFD simulations, which form the basis of the modeling approach, is described.

A modeling method called Sparse Identification of Nonlinear Dynamics (SINDy) is chosen to model the lift of a 2-degrees-of-freedom (2-DOF) flat plate. This method is based on identifying the right-hand side of the governing differential equations of the dynamical system using sparse regression. Several optimizers are also introduced. Afterward, multiple optimization routines are used to fit models using the CFD simulation data for several different oscillation regimes.

At the end of the thesis, the identified models are applied to predict flutter. Some of the models show very promising results. Based on the resulting time series and bifurcation diagrams, several important conclusions are made corresponding to the structure of the identified models, which need to be taken into account in the modeling process.

The result of this thesis is a compact method that results in a fast to compute model, which accurately describes the aeroelastic phenomena by using a small amount of measurement data. Although it is demonstrated that these types of models can describe the aerodynamics of large amplitude vibrations, more research is necessary before application to the design of real-world aircraft is possible. In this thesis, the models are applied only for similar parameters to the training dataset. Before real-world applications, it would be essential to show that the models identified using this approach are applicable to the whole flight regime of the aircraft. It is suspected that the best models would be able to generalize because they only compute the lift coefficient, but this needs to be shown using simulation data.

Another research direction would be applying this procedure to transonic flows, which are very important to modern transport airplanes. Several different measurements have been made using 2-DOF models for different airfoils [37, 38, 39]. A data-driven approach has also been used to create a reduced-order model using a neural network [40]. These two data-driven methodologies could be compared.

It also could be taken into account that a real-world airplane has 3-dimensional flexible wings, and the vibrations are much more complex than those described by the 2-DOF model. Several measurement data is available using this approach [41, 42, 43]. CFD simulations have also been carried out by coupling a Finite-Element Method (FEM) to the CFD simulation [44, 45].

Bibliography

- [1] J. Holierhoek, J. De Vaal, A. Van Zuijlen, and H. Bijl, “Comparing different dynamic stall models,” *Wind Energy*, vol. 16, no. 1, pp. 139–158, 2013.
- [2] J. Boutet and G. Dimitriadis, “Unsteady lifting line theory using the wagner function for the aerodynamic and aeroelastic modeling of 3d wings,” *Aerospace*, vol. 5, no. 3, p. 92, 2018.
- [3] H. Wagner, “Über die entstehung des dynamischen auftriebes von tragflügeln,” 1924.
- [4] T. Theodorsen, “Report no. 496, general theory of aerodynamic instability and the mechanism of flutter,” *Journal of the Franklin Institute*, vol. 219, no. 6, p. 766–767, 1935.
- [5] D. A. Peters, S. Karunamoorthy, and W.-M. Cao, “Finite state induced flow models. i-two-dimensional thin airfoil,” *Journal of aircraft*, vol. 32, no. 2, pp. 313–322, 1995.
- [6] G. J. Leishman, *Principles of helicopter aerodynamics with CD extra*. Cambridge university press, 2006.
- [7] C. Tran and D. Petot, “Semi-empirical model for the dynamic stall of airfoils in view of the application to the calculation of responses of a helicopter blade in forward flight,” 1980.
- [8] J. G. Leishman and T. Beddoes, “A semi-empirical model for dynamic stall,” *Journal of the American Helicopter society*, vol. 34, no. 3, pp. 3–17, 1989.
- [9] H. Snel and J. Schepers, “Joint investigation of dynamic inflow effects and implementation of an engineering method,” 1995.
- [10] H. Snel, “Heuristic modelling of dynamic stall characteristics,” in *EWEC-CONFERENCE*-, pp. 429–433, Bookshop for Scientific Publications, 1997.
- [11] E. H. Dowell, H. C. Curtiss, R. H. Scanlan, and F. Sisto, *A modern course in aeroelasticity*, vol. 3. Springer, 1989.
- [12] A. Collar, “The expanding domain of aeroelasticity,” *The Aeronautical Journal*, vol. 50, no. 428, pp. 613–636, 1946.

- [13] D. G. Tamás, *Aeroelasztikus jelenségek és dinamikai terhelés*. 2015.
- [14] R. L. Bisplinghoff, H. Ashley, and R. L. Halfman, *Aeroelasticity*. Courier Corporation, 2013.
- [15] A. Harish, “Why the Tacoma Narrows bridge collapsed: An engineering analysis.” <https://www.simscale.com/blog/2018/07/tacoma-narrows-bridge-collapse/>. Accessed: 2020-07-10.
- [16] “Tacoma narrows bridge (1940).” [https://en.wikipedia.org/wiki/Tacoma_Narrows_Bridge_\(1940\)](https://en.wikipedia.org/wiki/Tacoma_Narrows_Bridge_(1940)). Accessed: 2020-07-10.
- [17] “Tacoma bridge collapse: The wobbliest bridge in the world? (1940) | british pathé.” <https://www.youtube.com/watch?v=XggxeuFDaDU>.
- [18] D. Cenciotti, “David Cenciotti on twitter.” <https://twitter.com/cencio4/status/947928399384907776>. Accessed: 2020-07-10.
- [19] S. L. Brunton, J. L. Proctor, and J. N. Kutz, “Discovering governing equations from data by sparse identification of nonlinear dynamical systems,” *Proceedings of the National Academy of Sciences*, vol. 113, no. 15, p. 3932–3937, 2016.
- [20] S. L. Brunton, J. L. Proctor, and J. N. Kutz, “Sparse identification of nonlinear dynamics with control (sindyc),” *IFAC-PapersOnLine*, vol. 49, no. 18, p. 710–715, 2016.
- [21] K. Champion, P. Zheng, A. Y. Aravkin, S. L. Brunton, and J. N. Kutz, “A unified sparse optimization framework to learn parsimonious physics-informed models from data,” *IEEE Access*, vol. 8, p. 169259–169271, 2020.
- [22] B. D. Silva, K. Champion, M. Quade, J.-C. Loiseau, J. Kutz, and S. Brunton, “Pysindy: A python package for the sparse identification of nonlinear dynamical systems from data,” *Journal of Open Source Software*, vol. 5, no. 49, p. 2104, 2020.
- [23] K. Kaheman, J. N. Kutz, and S. L. Brunton, “Sindy-pi: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 476, no. 2242, p. 20200279, 2020.
- [24] A. A. Kaptanoglu, J. L. Callahan, A. Aravkin, C. J. Hansen, and S. L. Brunton, “Promoting global stability in data-driven models of quadratic nonlinear dynamics,” *Physical Review Fluids*, vol. 6, no. 9, 2021.
- [25] D. E. Shea, S. L. Brunton, and J. N. Kutz, “Sindy-bvp: Sparse identification of nonlinear dynamics for boundary value problems,” *Physical Review Research*, vol. 3, no. 2, 2021.

- [26] S. L. Brunton, J. N. Kutz, K. Manohar, A. Y. Aravkin, K. Morgansen, J. Klemisch, N. Goebel, J. Buttrick, J. Poskin, A. W. Blom-Schieber, and et al., “Data-driven aerospace engineering: Reframing the industry with machine learning,” *AIAA Journal*, p. 1–26, 2021.
- [27] C. Sun, T. Tian, X. Zhu, and Z. Du, “Sparse identification of nonlinear unsteady aerodynamics of the oscillating airfoil,” *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 235, no. 7, p. 809–824, 2020.
- [28] J. Pohl, R. Semaan, and A. R. Jones, “Dynamic lift measurements on an airfoil with periodic flap motion at high reynolds number,” *AIAA Scitech 2019 Forum*, 2019.
- [29] S. Li, E. Kaiser, S. Laima, H. Li, S. L. Brunton, and J. N. Kutz, “Discovering time-varying aerodynamics of a prototype bridge by sparse identification of nonlinear dynamical systems,” *Physical Review E*, vol. 100, no. 2, p. 022220, 2019.
- [30] N. Fonzi, S. L. Brunton, and U. Fasel, “Data-driven nonlinear aeroelastic models of morphing wings for control,” *Proceedings of the Royal Society A*, vol. 476, no. 2239, p. 20200079, 2020.
- [31] L. Bálint and L. János, “Aeroelasztikus szárnymodell numerikus vizsgálata : Numerical analysis of aeroelastic wing model,” *Nemzetközi Gépészeti Konferencia – OGÉT*, vol. 29, p. 48–51, Apr. 2021.
- [32] L. Bálint and L. János, “Aeroelasztikus szárnymodell analitikus és numerikus vizsgálata,” *TDK Thesis*, Oct. 2020.
- [33] R. L. Halfman, *Experimental aerodynamic derivatives of a sinusoidally oscillating airfoil in two-dimensional flow*. National Advisory Committee for Aeronautics, 1951.
- [34] R. I. Windsor, “Measurement of aerodynamic forces on an oscillating airfoil,” 1970.
- [35] J. Ertveldt, R. Pintelon, and S. Vanlanduit, “Identification of unsteady aerodynamic forces from forced motion wind tunnel experiments,” *AIAA Journal*, vol. 54, no. 10, p. 3265–3273, 2016.
- [36] J. Lelkes and T. Kalmár-Nagy, “Analysis of a piecewise linear aeroelastic system with and without tuned vibration absorber,” *Nonlinear Dynamics*, 2020.
- [37] K. Isogai, “On the transonic-dip mechanism of flutter of a sweptback wing,” *AIAA Journal*, vol. 17, no. 7, p. 793–795, 1979.
- [38] K. Isogai, “Transonic dip mechanism of flutter of a sweptback wing. ii,” *AIAA Journal*, vol. 19, no. 9, p. 1240–1242, 1981.

- [39] J. Heeg, P. Chwalowski, D. E. Raveh, M. J. Dalenbring, and A. Jirasek, “Plans and example results for the 2nd aiaa aeroelastic prediction workshop,” *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2015.
- [40] W. Li, X. Gao, and H. Liu, “Efficient prediction of transonic flutter boundaries for varying mach number and angle of attack via lstm network,” *Aerospace Science and Technology*, vol. 110, p. 106451, 2021.
- [41] E. C. Yates, “Agard standard aeroelastic configurations for dynamic response i-wing 445.6,” *AGARD Report 765, North Atlantic Treaty Organization, Group for Aerospace Research and Development*, 1988.
- [42] T. R. Brooks, G. K. W. Kenway, and J. R. R. A. Martins, “Benchmark aerostructural models for the study of transonic aircraft wings,” *AIAA Journal*, vol. 56, no. 7, p. 2840–2855, 2018.
- [43] M. Rivers and A. Dittberner, “Experimental investigations of the nasa common research model in the nasa langley national transonic facility and nasa ames 11-ft transonic wind tunnel (invited),” *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2011.
- [44] B. Zhang, W. Ding, S. Ji, and J. Zhang, “Transonic flutter analysis of an agard 445.6 wing in the frequency domain using the euler method,” *Engineering Applications of Computational Fluid Mechanics*, vol. 10, no. 1, p. 243–254, 2016.
- [45] B. Stanford and S. J. Massey, “Uncertainty quantification of the fun3d-predicted flutter boundary on the nasa crm,” *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2017.