UNIVERSITY OF SOUTHAMPTON

Faculty of Engineering and Physical Sciences
Aeronautics & Astronautics
School of Engineering

Data Driven Methods for Separated Flow over Airfoils

by

Francis De Voogt

ORCiD: 0000-0002-7229-7160

A thesis for the degree of
Doctor of Philosophy

November 2023

University of Southampton

Abstract

Faculty of Engineering and Physical Sciences
Aeronautics & Astronautics
School of Engineering

Doctor of Philosophy

Data Driven Methods for Separated Flow over Airfoils

by Francis De Voogt

The investigation of separated flows over airfoils is notoriously difficult due to three-dimensional and unsteady effects. These flows require extensive experimental or computational data that can be analysed using a variety of tools. In this work, various data-driven methods have been used to examine flow over stalled wings to understand the flow physics and develop reduced-order models for predictions.

It is shown that sparsely distributed sensors in the flowfield can also predict the state of the flow. Performance of multiple data-driven reduced-order models (linear and non-linear) together with pseudo probes in the flow are used to reconstruct the separated flow. A non-linear neural network based approach is found to perform better in reconstructions across different cases.

To enhance physical interpretation of non-linear reduced-order modelling (such as autoencoders), a hierarchical approach is examined. Subnetworks are trained to rank the non-linear modes according to their contributions to the reconstruction. By forcing the latent space distributions towards a unit normal distribution, with a variational autoencoder, it becomes possible to disentangle the separate modes. It has been shown that with the proper regularisation the non-linear modes become nearly orthogonal, and offer a better reconstruction than the truncated proper orthogonal decomposition.

A large computational data set of flow over a NACA 0012 wing has been created with different types of flow ranging from attached to fully separated flow. The importance of the flow characteristics near the surface of the wing has been indicated. It is shown that surface pressure can be used to predict these flow characteristics in liaison with a data-driven stall detection model.

Finally, leveraging flow visualisation using tufts, a data driven model that estimates the unsteady lift fluctuations based on tuft motions is developed. A proof of concept is examined with computational data and subsequent wind tunnel experiments together with a neural network reduced order model to provide accurate estimates of lift and pitching moment fluctuations.

Contents

List of Figures

Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:
   Carter et al. (2021)

Signed: Francis De Voogt　　　　Date: May 2023

Acknowledgements

This work would not have been possible without the guidance from my supervisor professor Bharathram Ganapathisubramani. Important conversations have guided me to pursue the right path and have enlightened me with the necessary knowledge to complete the journey. In addition to these conversations the pub meetings with the whole research group created a very supportive close knit group. I thoroughly enjoyed all the conversations that I have had with my colleagues about both work and life. I also want to express my deepest gratitude to all the people who were geographically further removed from me because I chose to study in Southampton. The continued support that I received from these people during this endeavour, has given me a lot of motivation and made me very happy.

Chapter 1

Research Background

The airfoil shape is ubiquitous in aerodynamic applications because it is able to generate a large lift force with minimal drag. This characteristic allows for the design of efficient wings, which are used in a wide variety of applications. It is not uncommon for these applications to encounter situations that lead to flow separation over the wing. A critical challenge is to model, understand and control the behavior of separated flows over airfoils. Flow separation can cause significant loss of lift and increased drag, leading to a reduction in the overall efficiency of an aircraft or any other aerodynamic system. Despite decades of research, the precise nature of separated flows remains a challenging problem, with no clear analytical solutions or efficient modelling method available. However, advances in data driven techniques have opened up new possibilities. In this thesis, data driven methods are explored for analysing separated flows and the influence thereof on airfoil shaped wings. Specifically, the use of machine learning methods has been investigated for large scale data sets generated by both numerical simulations and experiments. This research has the potential to significantly improve the efficiency of aerospace engineering designs by offering a data driven approach for problems with no trivial analytical solution.

1.1 Separated flow over airfoils

Some of the earliest research into airfoils was conducted by the Wright brothers in a home build wind-tunnel in 1901. They measured the lift and drag at different angles of attack for a range of airfoil shapes. A NACA technical report by Jacobs et al. (1933) described for the first time the NACA 4 digit airfoil series. The report also included similar lift measurements at varying angles of attack. These initial investigations already showed that increasing the angle of attack of a wing increases the lift up to a maximum value which is observed at the stall angle of attack. Increasing the angle of attack beyond the stall angle of attack results in a lower lift coefficient. McCullough

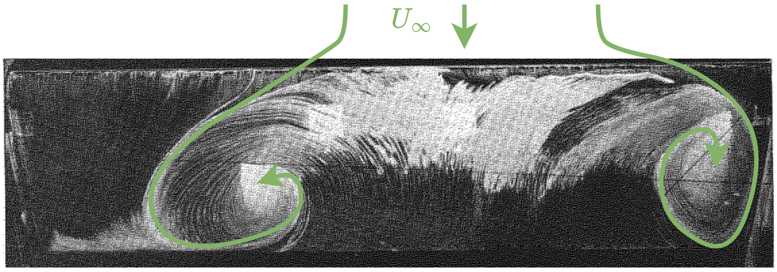


FIGURE 1.1: Suction surface view of a stalled NACA 0012 wing. A flow pattern is indicated by surface oil flow visualisation. Original image from Moss and Murdin (1968).

and Gault (1951) described the classification of three different types of stall which are distinguished by the manner in which the flow separates from the airfoil. Each type of stall was also indicated to be characteristic for different airfoil thicknesses. The relation between the shape of an airfoil and its post stall characteristics appears to be simple however the actual flow mechanisms leading to the different stall type observations are complex. The complexity of stall behaviour was further described by Moss and Murdin (1968) who found unexpected spanwise variations in the flow over a stalled wing as shown in fig. 1.1. This three dimensional separated flow is highly turbulent but still contains a coherent flow pattern which later has been referred to as a stall cell. Yon and Katz (1998) showed that stall cells tend to occur when the angle of attack is increased beyond stall. Further increases in angle of attack may result in the stall cells disappearing. Different investigations have found similar behaviour for different airfoils. Additionally Reynolds number variations at a constant angle of attack may introduce or eliminate stall cells. The coherence in stalled flow patterns has created significant research interest, which is aimed at understanding stall cell characteristics. Finding patterns in chaotic and turbulent flow offers a chance to create a more generalised understanding of the flow as a whole and estimate the influence that specific flow patterns have on the wing performance.

1.2 Data driven methods

The earliest efforts in aerodynamics research used wind-tunnels in combination with a force balance. Wind-tunnels are still an essential tool for aerodynamics research in combination with new experimental methods. These methods allow for the collection of large amounts of data that capture the flow under investigation. Experimental measurement methods in a wind-tunnel can be restricted to localised measurements of the flow characteristics, making it difficult to capture the interaction of large scale 3D flow features.

Computational fluid dynamics (CFD) simulations are able to provide information in the entire computational domain. CFD solvers use models that are analytically derived simplifications of the Navier-Stokes equations. This approach allows for general applicability, control of the accuracy and interpretable results. Unfortunately when separated flow is included in the simulated domain the accuracy requirements and computational cost increase substantially.

Data driven methods aim to create a model from data. Many different data driven methods exist and most are based on using data to obtain converged coefficients for a model. As the model is not linked directly to any known concisely written equations, the description of the model can require thousands to millions of coefficients. Due to the large number of coefficients such a model can approximate complex relations. With a limited amount of data to optimise these coefficients the model might not generalise well to new data. To prevent this issue more data can be used to obtain the converged coefficients of the model, or a model with less coefficients can be used.

Both experimental and computational methods can produce vast amounts of data. This data can be used to construct a data driven model. The proper orthogonal decomposition (POD), also referred to as principle component analysis (PCA), is a data driven method which was introduced by Payne and Lumley (1967) as a tool for the analysis of coherent structures in turbulent flow. Recent investigations such as Ceglia et al. (2023) and Liu et al. (2023) still utilise the POD for the same purpose. However, the POD is also an important tool for dimentionality reduction as illustrated by Zhang (2023) and Geibel and Bangga (2022). The POD is a popular tool as it is fast and interpretable. The implementation of POD varies in different software packages, however generally it is derived from the Singular Value Decomposition (SVD). The SVD also has varying implementations but will result in the linear decomposition of a data set into three matrices. These matrices represent the spatial modes which is a set of orthonormal multidimensional vectors, these vectors indicate directions of variance in the data set scales. The singular values are scaling factors for the spatial modes and the temporal modes which is a set of orthonormal vectors containing the coefficients corresponding to the spatial modes for each temporal sample. The multiplication of these three matrices results in the original data set (with linearly independent samples and more samples than variables in the data set). Larger singular values for a spatial mode indicate that more variance in the data set is represented by this spatial mode. This allows to capture most of the variance in the data set with a limited amount of modes which is less than the original number of parameters in the data set.

By utilising less than the total amount of modes for a reconstruction of the original data, a truncated reconstruction is obtained. Such a truncated reconstruction has an error in comparison to the original sample because not all modes from the decomposition of the original data set are included in the reconstruction.

Another method which has gained significant popularity is neural networks. This method builds a model that consists of different nodes that take several inputs and produce an output. Multiple nodes in parallel, which form a layer, can use the same set of inputs. Different layers can be connected to form a network.

The manner in which the nodes are organised within a network is referred to as the network architecture. This is an important consideration as it directly influences both the training and performance of the network. The architecture has unlimited possibilities for how different layers of nodes can be connected. This flexibility also allows to use neural networks for a wide variety of applications such as addressed in this research.

A good understanding of the problem under consideration is necessary to choose the neural network approach with the best outcome. The network should be suitable for the data that is available and the input to output mapping that is desired. The available data should be formulated optimally for the problem and neural network, in combination with the metric which guides the training of the network.

1.3 Research gaps

In some applications there can be a set of high quality data available from which patterns or relations can be learned. Afterwards we want to obtain the full data from just a sparse subset of that data. This process is called sparse reconstruction and applies to range of applications from the reconstruction of faces by Wright et al. (2008) to reconstructing sea surface temperature measurements by Manohar et al. (2018). For the investigation of separated flows it would be of interest to obtain velocity fields at a very high sampling frequency. Experimental methods are continuously evolving to expand their capabilities. It is possible to acquire PIV snapshots at a very high frequency with a dedicated setup. By using a data driven sparse reconstruction approach it is possible to take probe measurements with for example hot wire anemometry to reconstruct PIV velocity fields. This reduces the need for the continued use of a complicated experimental setup. Previous investigations have focused on flow over a flat plate, over a cylinder or within a channel (Nair and Goza (2020), Erichson et al. (2020), Guastoni et al. (2021)). A similar approach can be very useful for the investigation of separated flow fields if successful. A range of sparse reconstruction methods exist. A comparison of several of these methods on experimental data of separated flow over an airfoil is necessary as a first step towards improving experimental investigations with prior knowledge. An extensive comparison on a complex data set with many parameters, requiring many modes does not yet exist.

POD allows decomposition of a data set into linear components. The use of linear modes is sub optimal for many non-linear flow phenomena. The exploration of how non-linear modes can capture more of the flow phenomena with less modes can further improve the solution of a variety of problems which require a data driven description of the flow. The use of non-linear modes has been investigated by Saegusa et al. (2004) from a fundamental perspective. More recently the application of non-linear modes to the field of fluid dynamics has been explored by Fukami et al. (2020) and Eivazai et al. (2022). From these investigations it is clear that there are a variety of methods available with their own strengths and weaknesses. Finding the best method for the investigation of separated flows can potentially significantly improve dimensionality reduction. Novel non-linear mode decomposition methods should be compared through the use of a single data set, to clearly illustrate the differences.

Simulations with RANS are common in both industry and research but have reduced accuracy for turbulent flow. Manni et al. (2016) used both unsteady RANS simulations and delayed detached eddy simulations to investigate stall cells. Simulations which model turbulent separated flow more accurately are significantly more computationally expensive which prohibits their use for optimisations. Experiments can produce accurate data, but changes to a test object or the setup are often difficult. Some experimental methods are invasive and require careful consideration in the setup, so as to not disturb the flow to such an extent that it might become unrepresentative for the desired measurements. Different investigations such as Liu and Nishino (2018) or Zarutskaya and Arieli (2005) have computationally simulated stall cells. Previous investigations have typically used a limited set of Reynolds numbers and angles of attack. However experimental investigations such as De Voogt and Ganapathisubramani (2022) and Dell'Orso and Amitay (2018) indicate that stall cells are present in a wide range of angles of attack and Reynolds numbers. Simulations of stall cells which cover a large range of angles of attack and Reynolds numbers can produce a range of data which could not all be produced experimentally. Such a data set can provide detailed characteristics of the flow within a stall cell, while illustrating the range in which they are relevant. Experimental investigations such as Gregory and O'Reilly (1970) or more recently Dell'Orso et al. (2016) have relied on surface flow visualisation to indicate the presence stall cells. The importance of the visual identification of stall cells is also due to a lack of a method to clearly identify their presence. The detection of stall is crucial to the safety of modern aircraft. With a good knowledge of the surface flow over a range of angles of attack and Reynolds numbers it becomes possible to construct a data driven approach which relies on surface pressure inputs to detect stall. The preliminary work by Zhou et al. (2021) presents a POD based data driven approach for one dimensional stall detection for a single spanwise location. This work can be expanded to incorporate distributed pressure measurements on the entire wing surface. By including spanwise variation it becomes possible to detect the presence of stall cells.

The surface flow on an object can be visualised with tufts to provide a dynamic assessment of the sparse surface flow directions. Due to the imperfections of tufts it is intractable to formulate an analytical expression for the relation between tuft measurements and other aerodynamic parameters. Data driven analysis of tufts to provide aerodynamic parameters is relatively new and rapidly evolving. Swytink-Binnema and Johnson (2016) used tufts to obtain an estimate of the stall fraction of wind turbine blades. More recently Steinfurth et al. (2020) used tufts to obtain velocity field estimations. It is known that the surface flow over a wing is related to the surface pressure. As such the flow direction as sparsely indicated by the tufts should provide partial information related to the surface pressure. With a non-linear data driven approach it could be possible to estimate aerodynamic parameters related to the surface pressure based on sparse tuft measurements. There is currently not yet an investigation which indicates the ability to use tufts for a quantitative estimate of the aerodynamic force influence on an object.

1.4 Thesis research contributions

In the current thesis several problems associated with separated flow over airfoils will be addressed with data driven methods. In chapter 5 a data set of separated flow simulations over a NACA 0012 wing is presented. This data set provides valuable insights in the behaviour of flow separation for a range of angles of attack and Reynolds numbers. Surface pressure measurements taken experimentally show good agreement with the computational results which indicates that the computational results are representative for the performance of complex separated flow. An investigation of the flow near the wing surface indicates how the flow patterns near the surface are closely related to the wing lift coefficient. Based on the surface pressure a data driven stall detection algorithm is created. In order to optimise such a stall detection system the amount of surface pressure measurements that are required should be minimised.

In chapter 6 neural networks have been used to create a surrogate model for the lift and pitching moment of stalled wing, based on the visual observation of tufts. A complex relation between the motion of tufts and the variation in aerodynamic performance parameters exists but cannot be easily formulated analytically. Measurements of the wing performance combined with concurrent tuft observations have been used to create a neural network which approximates this relation. It has been shown on both computational and experimental data that this is possible with high accuracy. Whereas tufts are commonly used as a qualitative validation tool, they have now been illustrated to be suitable for obtaining valuable quantitative data. The ubiquitous use of tufts further underscores the importance of this method which could be adapted to obtain a wide range of aerodynamic parameters for a variety of applications.

In chapter 3 a data set of velocity fields of separated flow over a NACA 0012 wing has been used to investigate sparse reconstruction methods. A minimal set of local velocity measurements are taken as inputs for a data driven method which reconstructs the full velocity fields. The investigation compares several linear methods which estimate the coefficients for a POD reduced order model of the velocity fields. The reconstruction of the velocity fields then follows from the combination of the estimated coefficients and POD modes. In this work it is shown that by introducing a non-linear neural network the estimate of the coefficients can be significantly improved resulting in better reconstructions compared to the purely linear methods. Additionally the selection of the local measurements has also been investigated by comparing random measurement locations to measurement locations adapted to the known POD modes. The optimised measurement locations provided an improvement in the reconstruction which was larger for the full linear reconstruction method. The non-linear updating of the estimated coefficients was shown to be less sensitive to the location of the sparse velocity measurements.

The use of POD for the reduced order model provides a basis which requires a large amount of modes to capture the variance in the data set. The use of non-linear modes has the ability to capture separated flow with less modes. Several methods for non-linear modes of the separated velocity flow fields have been investigated in chapter 4. The relation between POD and neural networks with an autoencoder architecture is addressed to illustrate the similarities and differences. The use of neural networks with non-linear activation functions allows to create non-linear modes. By training successive subnetworks it becomes possible to create modes which are ranked according to the part of the data set that they are able to represent. Several different methods for obtaining non-linear modes exist, most of which have a latent space which is not interpretable. A beta variational autoencoder has been used to create modes which have a latent space which is approximately $\mathcal{N}(0, 1)$ distributed for each mode. The latent space regularisation creates a continuously representative latent space. Therefore, realistic samples can be created by sampling latent space coefficients from a unit normal distribution.

Parts of the current thesis have been published, are under consideration for publication or the manuscript is in preparation.

Published:

- “Data-driven sparse reconstruction of flow over a stalled aerofoil using experimental data” (2021). In: Data Centric Engineering.
  D.W. Carter, F. De Voogt, R. Soares and B. Ganapathisubramani
  DOI: 10.1017/dce.2021.5.

- “Unsteady Wing Loading from Tufts” (2022). In: AIAA AVIATION 2022.
  F. De Voogt and B. Ganapathisubramani
  DOI: 10.2514/6.2022-3887.

In preparation:

- “Determination of unsteady wing loading using tuft visualization”

- “Data driven stall warning from sparse pressure probes”

- “Non-linear sparse mode extraction of turbulent flow fields”

Parts of the current thesis have been presented at conferences, meetings and workshops.

- **ReCoVor** (*online, US*, Apr 2023) Presentation on the estimation of unsteady aerodynamic loading from visual tuft observations.

- **AIAA Aviation** (*Chicago, US*, Jun 2022) Presentation on the estimation of unsteady aerodynamic loading from visual tuft observations.

- **Workshop on Optical Metrology and Data Assimilation Applied to Aeroelastics** (*TU Delft, The Netherlands*, Sep 2021) Presentation on the sparse reconstruction of turbulent flow fields.

- **Machine Learning Methods for Prediction and Control of Separated Turbulent Flows (euromech 614)** (*Paris, France*, Jun 2021) Presentation on the sparse reconstruction of turbulent flow fields.

Chapter 2

Data and methods

In section 2.1 two data sets are described. These data sets are used multiple times throughout the research presented in this thesis. In section 2.2 the fundamentals of two commonly used data driven methods are explained.

2.1 Data sets

Two data sets reoccur in the thesis. The first data set, described in section 2.1.1, contains velocity fields of separated flow over a NACA 0012 wing. These velocity fields were experimentally captured with Particle Image Velocimetry (PIV). The second data set, described in section 2.1.2, contains the results from Computational Fluid Dynamics (CFD) simulations of a stalled NACA 0012 wing. This wing was simulated over a wide range of Reynolds numbers and angles of attack.

2.1.1 Separated flow PIV velocity fields

First the experimental setup for the PIV measurements is described, followed by the processing of the experimental data.

Contributions statement

The data acquisition has been performed by R. Soares and D.W. Carter, and processed by D.W. Carter. The text concerning the data acquisition and processing was initially written by D.W. Carter.

2.1.1.1 Water flume experiment

To obtain planar time-resolved velocity fields on the suction side of a separated NACA 0012 aerofoil, a PIV campaign was performed in the water flume flow facility located at the University of Southampton as illustrated in figure 2.1. A NACA 0012 airfoil model of chord length $c = 15$ cm and span $s = 70$ cm was fixed vertically in the centre of the span of the flume immediately following the contraction into the test section. The airfoil was fixed at angle of attack $\alpha = 12^\circ$ using a stepper motor attached to an overhead carriage system with precise control over the stepper motor angle. A 4 Megapixel Phantom v641 camera mounting a 105 mm Ex Sigma lens ($f = 5.6$) was directed upward from underneath the flume at a standoff distance 88 cm, resulting in a stream-wise wall-parallel field of view approximately 16 cm in the stream-wise direction x and 10 cm in the stream-normal wall-parallel direction y. The field of view was illuminated via a set of sheet-forming optics directing laser light from a Litron 527 nm Nd:YLF high-speed laser into the test section from the side of the facility at a wall-normal height of $h_L = 248$ mm from the bottom of the flume. The flume was filled until the water reached a height $h_w = 500$ mm. At this height the maximum frequency of the flume pumps yielded a free stream velocity $U_\infty = 0.5$ m/s, corresponding to a Reynolds number based on the chord length $Re_c = \frac{U_\infty c}{\nu} = 75,000$, where ν is the kinematic viscosity.

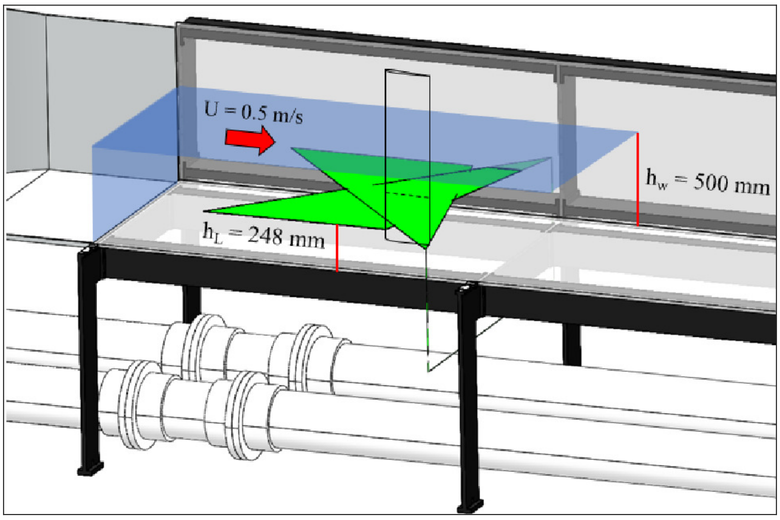


FIGURE 2.1: Illustration of the experimental setup focusing on the test section of the water flume flow facility at the University of Southampton. The NACA 0012 aerofoil is illuminated from both sides, however for this study a field of view focusing on the suction side of the aerofoil is used to capture the separation of the wake. The water level h_w and laser sheet level h_L are indicated

w

To collect the images, Davis 8.3.1 PIV software was used with a LaVision high-speed controller to ensure synchronous timing of laser and camera. The flow was seeded with Vestosint 2157 polyamide particles of nominal diameter 55 μm verified to behave as faithful flow tracers. The seeding density was iteratively adjusted until a satisfactory number of particle reflections across the field of view were observed. The images were captured at full resolution (2560 x 1600 pixels) at a frequency of 750 Hz.

In total, 13 fully time-resolved runs of 5468 sequential snapshots across 7.3 seconds each were collected for a total of 71084 snapshots. At an angle of attack $\alpha = 12°$, substantial unsteady separation from the surface of the airfoil was observed.

2.1.1.2 Data post-processing

The images were background-subtracted and low frequency noise was attenuated using a convolution with a Gaussian high-pass filter of standard deviation of 5 pixels. A mask corresponding to the location of the aerofoil in the images was stored and applied during PIV processing. The images were processed using a verified in-house PIV code based in MATLAB with iterative interrogation window stepping from 64 x 64 pixels to 32 x 32 pixels to 24 x 24 pixels with 50% overlap. Subpixel displacements were obtained via a Gaussian 3-point fit, and detected outliers were flagged and replaced via local interpolation. Interrogation windows found to overlap with the image mask by greater than 25% were discarded.

For each component of the velocity field, the velocity was decomposed into a mean and fluctuating component $u = U + u'$ and $v = V + v'$ where the the capitals here denote the mean velocity field and the prime denotes the fluctuating components of the horizontal (streamwise) u and vertical (stream normal) fields v. To reduce the random noise of the PIV fields, the filtering method based on POD described in Raiola et al. (2015) was used with the suggested value of the parameter $F = 0.999$ (the ratio of forward to backward residuals of the rank-restricted POD reconstruction). The location of the rank-order filtering truncation was found to correspond to only 5% of the energy of the velocity fluctuations. To improve the accuracy of the velocity fields, particularly in the separated region, gappy POD was employed using iterative replacement of detected outliers (Gunes et al., 2006) until satisfactory convergence was achieved. It was verified that the number of POD modes used to reduce random noise and improve the accuracy of the velocity fields was greater than the number of modes used to test the sparse reconstructions. Examples of the resulting velocity fields for the streamwise fluctuations (u') are shown in fig. 2.2.

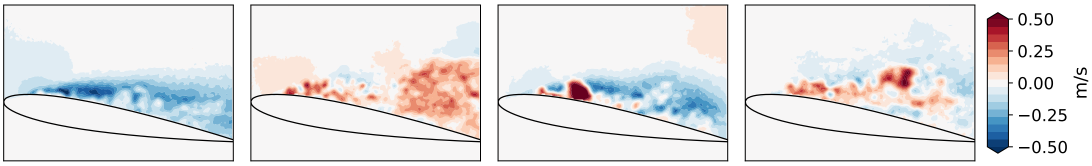


FIGURE 2.2: The streamwise velocity fluctuations (u') for four samples out of the PIV data set of separated flow over a NACA 0012 wing at Re = 75 000 and α = 12°.

2.1.2 Stalled wing CFD simulations

By utilising unsteady Reynolds averaged Navier Stokes (URANS) simulations it is possible to obtain a data set which spans many angle of attack and Reynolds number combinations within several weeks of wall clock time on a computer cluster. Other methods such as large eddy simulations (LES) or direct numerical simulations (DNS) with potentially better results for separated flow are not feasible to be executed for a large number of cases in a timely manner.

A NACA 0012 airfoil profile has been used with a one meter chord length (c), with orientations as shown in fig. 2.3. A C-shaped mesh grid has been used which extends 15 chord lengths to the circular inlet section and outlet from the leading edge of the airfoil. A 2D configuration of the NACA 0012 profile has been used to assess the required number of circumferential mesh nodes for URANS simulations. Each different mesh has been evaluated at three different angles of attack (5°, 10° and 15°) at a chord based Reynolds number of 10^6 to assess the consistency of the obtained lift coefficient for different meshes. The Reynolds number in this investigation refers to the chord based Reynolds number defined as Re = $\frac{u_\infty \cdot c}{\nu}$, where u_∞ is the freestream velocity, c is the chord length and ν is the kinematic viscosity of air at 15°C ($\nu = 14.61 \cdot 10^{-6}\ \frac{Ns}{m^2}$, $\rho = 1.225\ \frac{kg}{m^3}$). The resulting lift coefficients are shown in table 2.1.

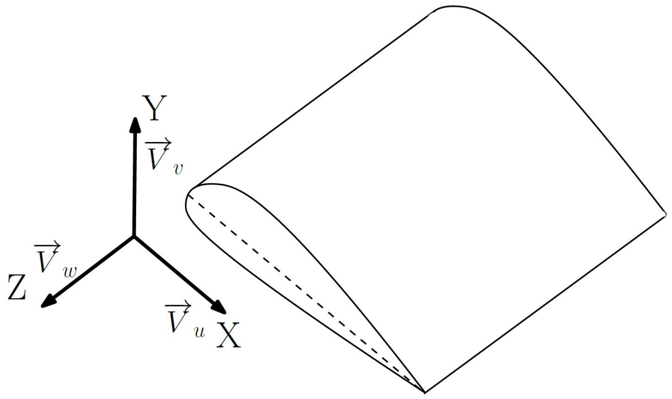


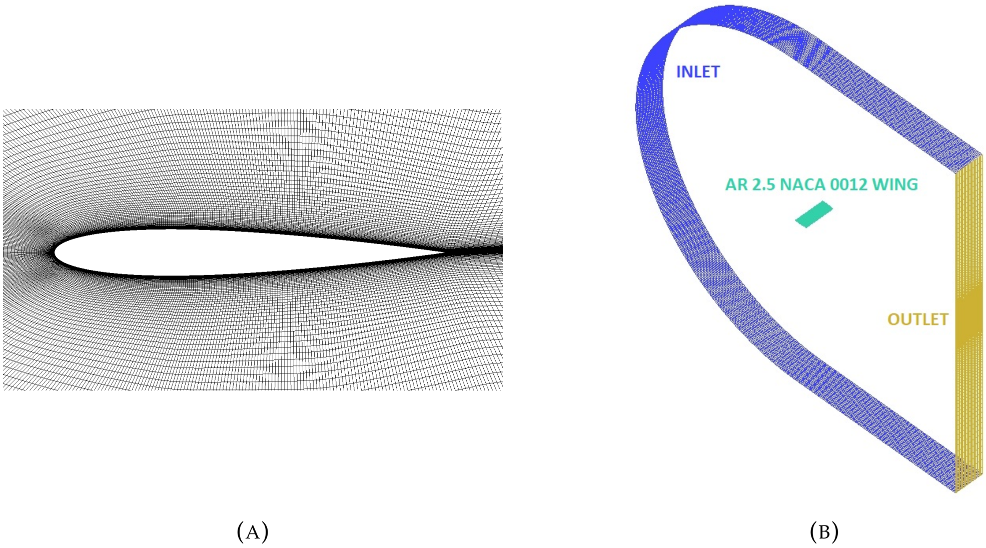
FIGURE 2.3: The orientation of the setup, with the x axis aligned with the chord and the y axis with the span. The velocity vector directions $\overrightarrow{V}_u$, $\overrightarrow{V}_v$ and $\overrightarrow{V}_w$ are indicated along the X, Y and Z directions.

The mesh convergence study showed that the mesh with 420 circumferential nodes provides a suitable balance between accuracy and computational cost for the 3D URANS simulations. A detailed view of the mesh with 420 circumferential nodes around the airfoil is shown in fig. 2.4a. The first node wall normal distance from the airfoil surface is set to $8 \cdot 10^{-6}$ m such that the $y^+ < 1$ for wall adjacent nodes at the highest Reynolds number that has been simulated, which is 10^6.

Circumferential nodes	C_L at $\alpha = 5^\circ$ / 10° / 15°
240	0.529 / 1.012 / 1.353
360	0.527 / 1.013 / 1.355
420	0.527 / 1.015 / 1.369
480	0.527 / 1.015 / 1.364
720	0.527 / 1.017 / 1.372

TABLE 2.1: 2D mesh convergence study. The resulting lift coefficient for three different angles of attack when using meshes with varying amounts of circumferential nodes on the airfoil surface.

To minimise the computational cost the spanwise domain size is kept as small as possible. For full stall cells to occur the spanwise domain should be large enough to fit at least a full stall cell with periodic boundary conditions. Based on previous investigations such as De Voogt and Ganapathisubramani (2022), Manni et al. (2016) and Zarutskaya and Arieli (2005) it has been estimated that a spanwise extrusion of 2.5 chord lengths should allow for the formation of a single full stall cell. The drawback of the minimal spanwise extension is that there is little room left for potential unsteady spanwise behaviour of a stall cell. The 2D mesh with 420 circumferential nodes has been extruded to obtain a spanwise uniform 3D wing. In fig. 2.4b the full domain is shown for the 3D configuration. The spanwise mesh spacing is 0.1 c, based on the findings of Liu and Nishino (2018) to allow for the formation of stall cells. This results in a 3D mesh with 25 spanwise sections. The wing surface has been set as a smooth no-slip wall. Periodic boundary conditions are used for the sidewalls of the domain. As such, a wing with infinite span and periodicity of 2.5 chord lengths along the span is modelled. The inlet is defined by the Cartesian velocity components for each case ($\alpha - \mathrm{Re}$ combination), and the outlet uses an average static pressure condition. Although a 5% turbulence intensity has been set at the inlet, the turbulence intensity has been verified to have fully dissipated when the flow reaches the airfoil. The fluid domain has been initialised with the freestream velocity components.



(A) (B)

FIGURE 2.4: The computational domain of the NACA 0012 wing. (a) Detail of the mesh around a NACA 0012 profile. (b) The boundaries of the 3D C-grid. Periodic boundary conditions are applied to the sides of the domain which are not shown.

A shear stress transport turbulence model has been used with automatic wall functions in ANSYS CFX 18.1. The boundary layer has been assumed to be fully turbulent. A high resolution advection scheme in combination with high resolution turbulence numerics and a second order backward Euler transient scheme was used for the URANS solver. The time step size was set to 0.001 s, which is significantly smaller than the timescale of interest for the analysis. The investigation of Liu and Nishino (2018) did not find an influence of the time step size on the results of the simulations. The convergence criterion for the RMS residuals was set to 10^{-5}, with a maximum of 15 coefficient loops per time step. In order to limit the size of the raw data output of the simulations, the results of the simulations have only been saved every 0.1 seconds. Simulations have been run for 10 to 20 seconds of simulated time, depending on the flow case, to obtain a statistically representative time frame for the flow case. As each flow case simulation takes up to two weeks on a quad-core CPU desktop, the simulations have been run on the IRIDIS high-performance computing cluster of the University of Southampton to speed up the data acquisition process of the simulations.

For validation of the results, a comparison of the time averaged lift coefficient (excluding the transient phase at the start of the simulation) is presented in fig. 2.5. The spanwise spacing of the mesh grid can influence the flow patterns that occur on the wing post stall (and thus lift), as such these are also mentioned Liu and Nishino (2018). The use of different solvers and other small differences can lead to differences in the resulting lift coefficient values, however all presented investigations in fig. 2.5 find the maximum lift coefficient at an angle of attack of 16 degrees. At higher angles of attack similar trends can be observed in the lift polar with an offset relative to each other. This is the angle of attack range for the specific Reynolds number in which stall cells occur.

At an angle of attack of 20 degrees and beyond the results appear to converge again. In this range the stall cell was observed to have disappeared and the surface flow showed fully separated flow along the entire span in the current investigation. The consistency of the trends shown in fig. 2.5 for the post stall lift polar however indicates that the results are representative of the specific flow behaviour which is consistently reproduced in different investigations.

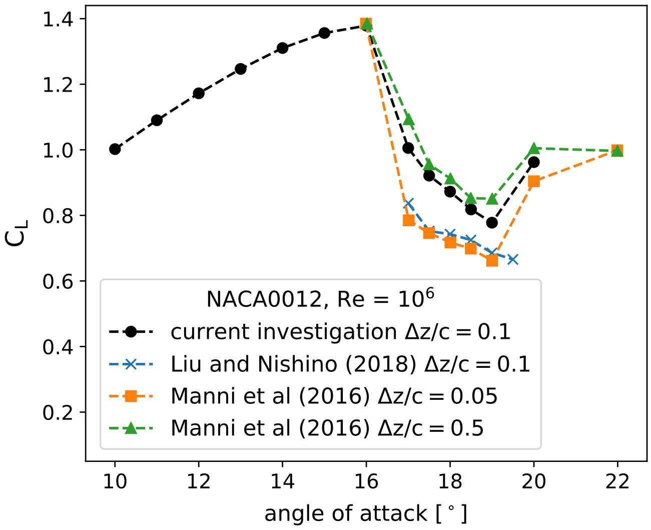


FIGURE 2.5: Comparison of the time-averaged lift coefficient obtained from CFD at a Reynolds number of one million. Validation data obtained from Manni et al. (2016) and Liu and Nishino (2018). The spanwise meshgrid spacing ($\Delta z/c$) is indicated in the legend for the different data sets.

With this computational setup a wide range of cases have been simulated as shown in fig. 2.6. The Reynolds number ranges from $3.1 \cdot 10^5$ to 10^6 and the angle of attack ranges from 14° (10° for Re = 10^6) to 20°. These unsteady simulations showed varying flow behaviours over the wing suction surface. Slight variations in computational setup can lead to significant changes in the results as shown by Manni et al. (2016) and Lehmkuhl et al. (2011). The type of flow separation that has been present most prominently for a case has been manually identified in fig. 2.6.

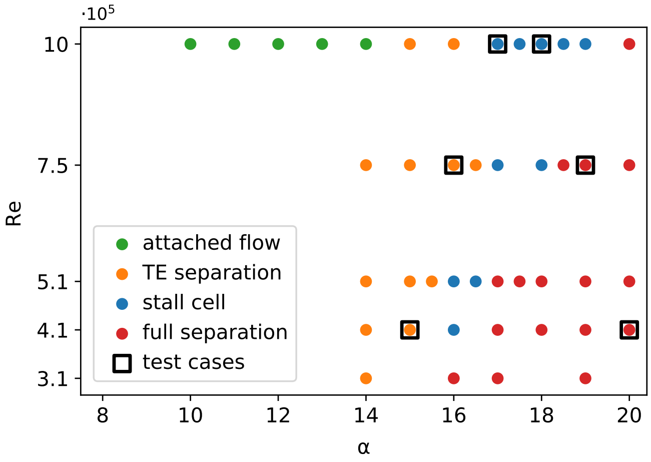


FIGURE 2.6: Identification of the flow type based on the angle of attack and Reynolds number.

2.2 Data driven methods

A succinct overview of commonly used data driven methods relevant to the research in this thesis is given. In section 2.2.1 the proper orthogonal decomposition (POD) is described. In section 2.2.2 some of the fundamental aspects of neural networks and the implementation thereof are described.

2.2.1 POD

The proper orthogonal decomposition (POD), also referred to as principal component analysis (PCA), is a data driven method which was introduced by Payne and Lumley (1967) as a tool for the analysis of coherent structures in turbulent flow. The POD is still a popular tool as it is fast, interpretable and easy to implement. The POD typically relies on the covariance matrix of a data set X with n rows for n samples and p columns for p variables:

$$\text{COV} = \frac{(X - \overline{X}^T)\ (X - \overline{X})}{n-1}, \tag{2.1}$$

with a mean $\overline{X}$ for n samples. The spectral theorem for real symmetric matrices, such as the covariance matrix, states that these matrices can be diagonalised by an orthogonal matrix W:

$$\text{COV} = W\,\Lambda\,W^{-1}, \tag{2.2}$$

where W represents the eigenvectors of the covariance matrix, with diagonal matrix Λ containing the eigenvalues. When these directions are given as unit vectors, they are often referred to a POD modes or spatial modes. POD, therefore, decomposes the data into linearly independent directions. The data can be projected onto the spatial modes to obtain the corresponding POD coefficients. The decomposition of a centred data set (X_c) into POD coefficients (A) and POD modes (Φ) is described as:

$$X_c = A\,\Phi, \tag{2.3}$$

where X_c has n rows for n samples and p columns for p variables. The rows of the Φ matrix contain the spatial modes. Where an individual spatial mode ϕ_k satisfies:

$$\phi_k \in \mathbb{R}^p, \tag{2.4}$$

for p variables in the training data. The linear decomposition is limited by the amount of available samples. When more samples than parameters are available ($n \geq p$) the linear decomposition results in an amount of modes (m) equal to the amount of parameters ($m = p$). When less samples are available than parameters ($n < p$) the number of modes that can be obtained is equal to the amount of samples ($m = n$). Matrix A in eq. (2.3) has n rows of POD coefficients for n samples, and m columns corresponding to m spatial modes.

The vector norm of the POD coefficients for a single spatial mode across all samples in the data (column of A), is referred to as the singular value for that spatial mode. POD coefficients normalised with their respective singular values, are referred to as temporal modes. The full decomposition can then be written as:

$$X_c = \Psi\Sigma\Phi, \tag{2.5}$$

where Ψ has size $[n \times m]$ for n samples and m modes. Diagonal matrix Σ with size $[m \times m]$ contains the singular values for the spatial modes. The magnitude of a singular value reflects the amount of variance that the corresponding spatial mode represents in the data set. Typically spatial modes are ordered according to their singular values. The decomposition shown in eq. (2.5) is referred to as the Singular Value Decomposition (SVD).

By using the first j spatial modes, with $j < m$, a rank truncated reconstruction of a data sample can be created. The limited set of j spatial modes can be multiplied with the respective singular values and the temporal modes of each sample, to result in a lower order representation of the original data. The truncated reconstruction (X_c') with j modes can be formulated as:

$$X'_c = \sum_{k=1}^{j} a_k \phi_k, \tag{2.6}$$

where ϕ_k represents the k^{th} spatial mode and a_k represents corresponding POD coefficients. In order to quantify the amount of modes which are necessary to achieve an accurate representation of the original data, the cumulative variance explained by a number of SVD components is a commonly used ratio. This ratio captures the variance in the data set which can be represented by the first j modes, where the modes have been ranked starting with the mode with the highest singular value and ending with the mode with the lowest corresponding singular value. The cumulative explained variance (CEV) is then calculated as:

$$\text{CEV} = \frac{\sum_{k=1}^{j} \sigma_k^2}{\sum_{k=1}^{m} \sigma_k^2}, \tag{2.7}$$

where σ_k is the singular value for the k^{th} spatial mode. This ratio can be calculated for increasing numbers of modes included. It is then possible to determine the necessary amount of modes to reconstruct a certain percentage of the variance in the original data set. The POD constructs the principal components based on the distribution of a given data set. The use of these components is thus only representative for samples belonging to a similar distribution. Samples from unrepresentative data sets may have the principal components for the variance oriented differently with different scaling, leading to the ineffective use of prior obtained principal components with differently distributed data.

2.2.2 Neural networks

Neural network nodes were inspired by biological neurons as described by McCulloch and Pitts (1943). A standard node for a neural network is constructed such as shown in fig. 2.7. In this node the output of previous nodes is the input. For each input into the current node there is a different weight. Additionally there is a constant bias value per node. The bias is added to the sum of the inputs multiplied with their respective weight. Then an activation function is applied and the result is the output of this single node.

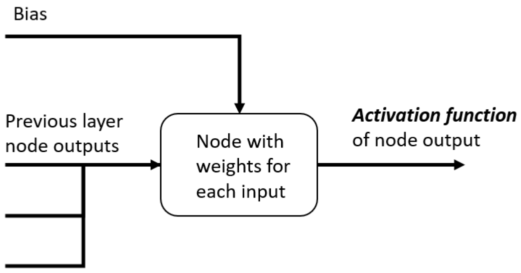

Bias
Previous layer node outputs
Node with weights for each input
Activation function of node output

FIGURE 2.7: A single node of a neural network.

The activation function is traditionally non-linear for layers which are not the input or output of a network. An example of the mathematical operation of a single layer with three nodes and four inputs is shown:

$$\begin{aligned} y_1 &= g(W_{11}x_1 + W_{12}x_2 + W_{13}x_3 + W_{14}x_4 + W_{15} \cdot 1), \\ y_2 &= g(W_{21}x_1 + W_{22}x_2 + W_{23}x_3 + W_{24}x_4 + W_{25} \cdot 1), \\ y_3 &= g(W_{31}x_1 + W_{32}x_2 + W_{33}x_3 + W_{34}x_4 + W_{35} \cdot 1), \end{aligned} \tag{2.8}$$

where the output of the layer is represented by y, the input by x, the weights of the layer by W and $g()$ is the activation function. The mapping of the input onto the output is achieved by the weights matrix W, which contains the weights for each input for each node in the layer, plus an extra weight per node for the bias. After the linear transformation the activation function is applied, to give the complete output vector y.

Two functions are commonly used as activation functions in a neural network. The first of which is the Sigmoid function:

$$S(x) = \frac{1}{1 + e^{-x}}, \tag{2.9}$$

and the function is also shown in fig. 2.8. For values smaller than -5 the Sigmoid function outputs approximately 0, and for values larger than 5 the output approximates 1. The non-linearity in the function is smooth.

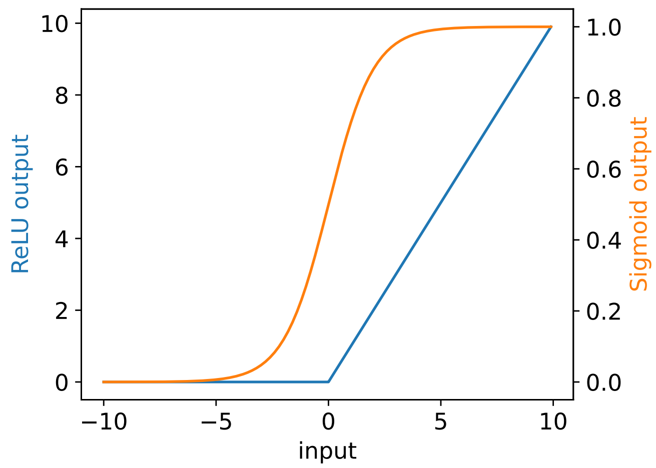


FIGURE 2.8: The Sigmoid and ReLU activation functions.

The second activation function which is used often is the rectified linear unit (ReLU) function:

$$R(x) = \max(0, x), \tag{2.10}$$

which simply sets negative input values to zero. The non-linearity is not smooth. The function is extremely easy to compute and therefore fast, both in forward and back-propagation calculations. The specific data available, network architecture and other parameters determine the optimal choice for the activation function. While not specifically considered an activation function, a unit linear activation function (or simply lack of an activation function) is used to obtain the network outputs and other operations within the network.

Ivakhnenko and Lapa (1965) introduced the method of stacking multiple layers together to create a neural network, which is still common practice today. The concise mathematical notation for a full neural network with L amount of layers, is given by:

$$Y = W_L \;\; g\Big(W_{L-1} \;\; g\big(W_{L-2} \;\; \dots \;\; g(W_1 X)\big)\Big), \tag{2.11}$$

which is a series of nested transformations with activation functions applied. Using multiple layers consecutively increases the amount of connections which allows to form more complex models.

Chapter 3

Data-driven Sparse Reconstruction of Flow over a Stalled Airfoil

Recent work has demonstrated the use of sparse sensors in combination with the proper orthogonal decomposition (POD) to produce data-driven reconstructions of the full velocity fields in a variety of flows. The present work investigates the fidelity of such techniques applied to a stalled NACA 0012 aerofoil at $Re_c = 75000$ at an angle of attack $\alpha = 12^\circ$ as measured experimentally using planar time-resolved Particle Image Velocimetry (PIV). In contrast to many previous studies, the flow is absent of any dominant shedding frequency and exhibits a broad range of singular values due to the turbulence in the separated region. Several reconstruction methodologies for linear state estimation based on classical compressed sensing and extended POD methodologies are presented as well as non-linear refinement through the use of a shallow neural network (SNN). It is found that the linear reconstructions inspired by the extended POD are inferior to the compressed sensing approach provided that the sparse sensors avoid regions of the flow with small variance across the global POD basis. Regardless of the linear method used, the non-linear SNN gives strikingly similar performance in its refinement of the reconstructions. The capability of sparse sensors to reconstruct separated turbulent flow measurements is further discussed and directions for future work suggested.

Contributions statement

The current chapter has been published as:
Data-driven sparse reconstruction of flow over a stalled aerofoil using experimental data.
by D.W. Carter, F. De Voogt, R. Soares, and B. Ganapathisubramani
in Data-Centric Engineering, volume 2, issue 5 in 2021. (Carter et al. (2021))

This work has been created in collaboration with the authors of the paper. The data acquisition has been performed by R. Soares and D.W. Carter. The idea for sparse reconstruction with POD was formulated by D.W. Carter and B. Ganapathisubramani. The linear approach for sparse reconstruction has been performed by D.W. Carter. The neural network refinement for the non-linear reconstruction has been performed by F. De Voogt. The sign correction matrix idea was formulated by F. De Voogt and worked out by D.W. Carter. The text concerning the non-linear reconstruction method and results was written by F. De Voogt. All other text was written by D.W. Carter. Some of the text has been adapted to suit the thesis format.

3.1 Introduction

Sparse reconstruction is a technique used to obtain accurate details about the full scale features of a system using a sparse subset of information (for example, a few pixels or measurements within the system) and has been the subject of interest for some decades (Donoho, 2006; Candès and et al., 2006). Applications for such state estimation problems range from reconstructing faces from limited or corrupted data (Wright et al., 2008) to deblurring and improving image resolution (Dong et al., 2011) to estimating global sea surface temperatures (Manohar et al., 2018; Callaham et al., 2019). The literature concerning state estimation and sparse reconstruction is rapidly developing. In the following we motivate and present a brief review of the literature as it relates to the present study. For a more comprehensive review, we refer the interested reader to Manohar et al. (2018) and Nair and Goza (2020).

In general, the requirements for accurate sparse reconstructions from limited data are that (i) the basis underlying the data exhibits sparsity (as will be discussed) and (ii) that full-state information for the system can be obtained or approximated *a-priori* to generate a global basis applicable to any sample or instant (Brunton and Kutz, 2019). For example, suppose a set of images of weathered ancient hieroglyphics are only partially discernible. If a library of images of known undamaged hieroglyphics is tabulated *a-priori*, the principle of sparsity can be used to estimate the weathered hieroglyphics before they were damaged (Roman-Rangel et al., 2012). Further examples of sparse reconstruction exist over a diverse range of engineering disciplines. Liu et al. (2017) demonstrated the use of sparsity to monitor and detect faults in an industrial Tennessee Eastman Process using a novel variation of principal component analysis. Bao et al. (2017) showed the use of sparse strain sensors to estimate the full stress state of a structure using a Fourier basis. Iyer et al. (2020) used recurrent neural networks and sparse observations for reconstructing and forecasting road traffic.

There remains much to be done in order to test the limitations of sparse reconstructions numerically, experimentally, and for a variety of engineering systems. For example:

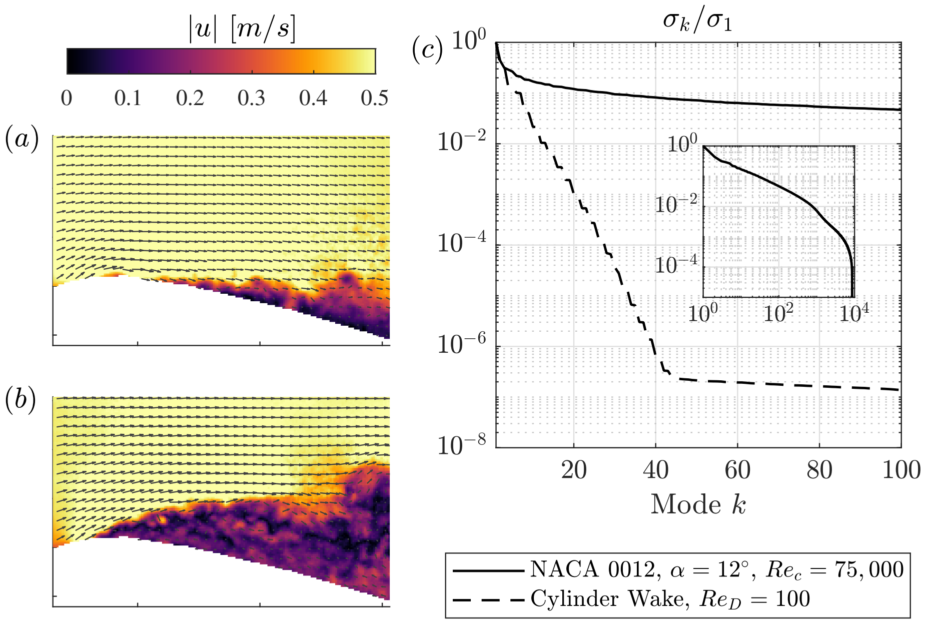


FIGURE 3.1: Instantaneous velocity fields (a,b) from the PIV data (every fifth vector shown for clarity) presented in this study at two separate instants; highlighting the variation in the size of the separated wake. The singular values are shown in (c) normalised by the first singular value (inset: up to 9000 modes). This is also shown for the laminar cylinder wake (dashed) of diameter D at $Re_D = 100$ from the DNS of Brunton and Kutz (2019) for comparison

how do variations of non-dimensional parameters that characterise a system impact the reconstruction? How many known full-state snapshots are required to generate a global basis? How many sparse sensors or probes are needed to achieve a desired reconstruction accuracy? Where should the probes be placed? These are some of the underlying questions that motivate the current study seeking to expand the application of sparse sensing to engineering problems.

The key underlying principle that allows for full reconstructions using only limited measurements is the sparsity of the representative basis. For the application of interest, a suitable basis must be chosen onto which to project the sparse signal (the limited measurements) for the full reconstruction. The basis of choice depends on the data in question but is typically taken to be either a Fourier transform (Candès and Wakin, 2008) or a data-driven basis such as the proper orthogonal decomposition (POD: Sirovich, 1987b; Berkooz et al., 1993). If the basis contains many entries that are small (near-zero) then the system is said to exhibit sparsity. For example, a Fourier transform may indicate only a few frequencies have significant amplitudes. Sparse reconstruction takes advantage of the sparsity of the basis functions to produce full reconstructions (Brunton and Kutz, 2019) that can yield surprisingly accurate full state estimations. For example, using POD as a representative basis Manohar et al. (2018) showed that using only 7 probes they could reconstruct the full vorticity field of laminar flow over a cylinder exceeding 90% accuracy.

POD is a data-driven method commonly used as a basis for reconstructions due to its attractive properties of being energy optimal and having time-independent spatial modes, though other choices are possible (Bai et al., 2015; Jayaraman et al., 2019). Manohar et al. (2018) demonstrated that the reconstruction accuracy is significantly improved if the choice of probe locations is made carefully such that it takes better advantage of the sparse basis. They demonstrated that the use of greedy algorithms to intelligently place probes within the flow improved the reconstructions greatly compared to random probe placement. However, this also depends on the complexity of the underlying system. When the analysis was applied to sea surface temperatures whose POD basis requires many more modes, the reconstruction accuracy relied more heavily on the optimal probe placement; pointing to the difficulty of reconstructing systems with a large range of spatio-temporal scales.

Several alternative methods originating in the fluid mechanics literature have been proposed for sensing turbulent flow structures and obtaining reconstructions, such as linear stochastic estimation (LSE: Adrian and Moin, 1988). In LSE, the state of the flow field is tabulated and conditioned upon the state of an event, leading to a linear map through an l_2 minimisation procedure. Several studies have demonstrated the fidelity of reconstructions based on the concept of LSE (Picard and Delville, 2000; Lasagna et al., 2013). Borée (2003) showed that LSE is equivalent to the so-called *extended* POD, where the temporal POD modes of one quantity are projected onto by the state of the flow field. This produces a set of extended modes revealing the spatial structures of the flow that are correlated with the quantity of interest. The extended POD framework is a straightforward and flexible approach offering alternative means of obtaining reconstructions using the temporal modes of the sparse information, as demonstrated by Hosseini et al. (2015). Extended POD was pushed further by Discetti et al. (2018) to obtain up-sampled time-resolved velocity fields from sparse sensors; demonstrating the flexibility of the framework.

Although the aforementioned techniques produce sparse reconstructions that are obtained entirely linearly, supervised machine learning provides a non-linear framework with additional flexibility. In fact, POD itself is a form of unsupervised machine learning (Brunton et al., 2020) and one may interpret the POD as an unsupervised neural network with a single layer and linear activation function (Milano and Koumoutsakos, 2002). The use of neural networks (NNs) to obtain sparse reconstructions has seen recent attention in the literature. Nair and Goza (2020) demonstrated the ability of NNs to outperform the linear counterparts using a network of three layers for low Reynolds number flow over a flat plate at high angles of attack. Similarly, Erichson et al. (2020) introduced a “shallow” neural network (SNN) consisting of two layers for reconstructing laminar flow over a cylinder or sea surface temperatures with high fidelity. They further applied this method to a comparatively more challenging numerical simulation of isotropic turbulence and found good results when interpolating, but difficulty in predicting future states. They discussed the use of regularisation of the loss function to avoid overfitting; a problem that can easily arise with limited training data. Intelligent choices for loss functions offers promising potential for improved neural network performance and generalisation. For example, imposing physical constraints on the loss function has been demonstrated as a successful approach for a variety of systems governed by partial differential equations (Raissi et al., 2019; Sun and Wang, 2020). More recently, convolutional neural networks have been demonstrated for flow reconstruction from wall shear measurements in a turbulent channel flow (Guastoni et al., 2021), however the inherent spatial dependence renders convolution-based approaches unattractive for the particular case of spatially sparse reconstruction. This does however represent progress in the direction of reconstructing turbulent flows, as their inherent range of spatial and temporal scales makes them significantly more challenging than the laminar case.

The present study focuses on the problem of sparse sensing motivated by the need to detect flow structures in aerospace applications. For example: to detect anomalous structures, the onset of stall over an aerofoil, or to inform control systems designed to reduce drag. Here the focus is on the situation of an aerofoil in stalled conditions. We seek to explore how a global basis may be used to predict the unseen flow about a stalled aerofoil using limited single-point sensors placed in the flow.

As of yet, data-driven sparse sensing has not been widely applied to advective turbulent flows such as the case of a stalled aerofoil due to the inherently large range of scales and difficulty in capturing a low-order representation of the dynamics. As a result, the flow requires many modes compared to e.g. the case of a laminar cylinder (figure 3.1) to capture the dynamics. This fact motivates exploration of reconstructions obtained outside the classic compressed sensing approach, which relies more heavily on optimal placement for systems with many modes. The present study seeks to explore how effectively the various linear and non-linear approaches for sparse reconstruction perform for (i) a situation which is highly turbulent and (ii) data generated via experiment and therefore contaminated by experimental noise.

The present study will be structured as follows. The experiments performed to obtain the two-dimensional time-resolved flow fields of the separated aerofoil are outlined in section 3.2. The methodology used to select probe placements and obtain the sparse reconstructions with and without neural network refinement is presented in section 3.3. The results are presented in section 3.4. Conclusions and outlook for future work are finally presented in section 3.5.

3.2 Experimental method

Particle image velocimetry (PIV) has seen widespread use in experimental fluid mechanics over the past decades due to its ability to obtain highly spatially-resolved instantaneous planar two-component velocity fields (Adrian et al., 2011). Combined with time-resolved capabilities of high-speed cameras, it is the measurement method of choice for the present data-driven analysis due to the ability to generate large spatial domains (and resulting spatial modes) as well as time information at each point in the flow, enabling the sparse reconstruction investigation via a pseudo-probe analysis. The details of the data collection and post-processing were outlined in section 2.1.1.

3.3 Sparse reconstruction methodology

3.3.1 Probe placement

The optimal placement of probes for sparse reconstruction is a challenging and ongoing subject of research (Manohar et al., 2018). As the problem involves both the number of sensors as well as the number of modes used for reconstruction, the optimisation is combinatorial in nature; making it intractable for even a modest number of possible sensor locations. The objective is to maximise the signal to noise ratio of the reconstruction by minimising the condition number of the sparse basis (Manohar et al., 2018; Jayaraman et al., 2019). A body of literature has been reported exploring heuristic greedy algorithms known as empirical interpolation methods (EIM: Barrault et al., 2004; Willcox, 2006; Yildirim et al., 2009) or discrete EIMs (DEIM: Chaturantabut and Sorensen, 2010; Drmac and Gugercin, 2016) to identify optimal locations for the probes. In addition, optimal design literature provide methods of identifying probe locations based on the moment matrix of the basis (e.g. A, C, D, E-optimal design, see Atkinson and Donev, 1992; Cox and Reid, 2000). Considerations for non-linear placement have also been explored as summarised in the recent work of Otto and Rowley (2022) and references therein.

The work of Manohar et al. (2018) shows that the optimal choice of placement (for the linear reconstruction) corresponds to the locations that contribute the maximum variance (l_2-norm) across the spatial basis. The QR decomposition with pivoting (Van Loan and Golub, 1983) is a commonly utilised heuristic for identifying such locations, resulting in an upper-triangular matrix R with entries ordered accordingly. The resulting pivot locations approximate the best sensor locations. When the number of modes used for the reconstruction is equal to the number of probes, the QR with column pivoting is applied to the spatial basis and is known as the Q-DEIM (Drmac and Gugercin,

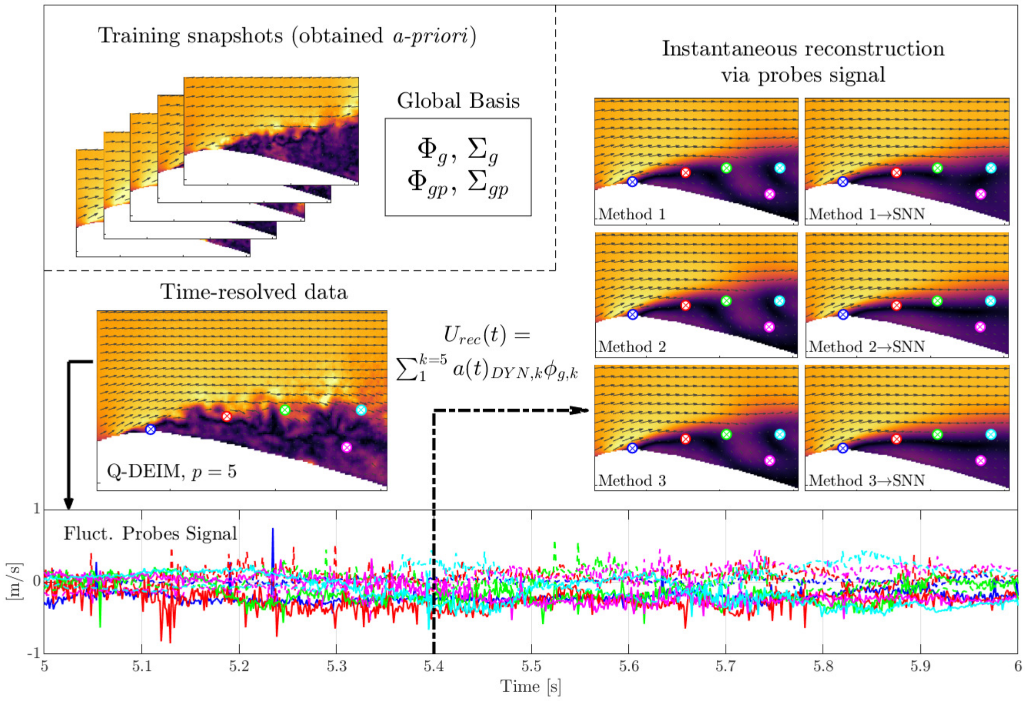


FIGURE 3.2: Conceptual illustration of the instantaneous reconstruction methodology using $p = 5$ probes and Q-DEIM placement. The global basis is obtained a-priori and the real-time probe signals are used to approximate the instantaneous fields. The probe signals are shown with a solid line for u' and dashed for v' and colour-coded according to their indicated locations. The total velocity shown in the plots are calculated by summing the mean and fluctuating fields. See sections 3.3.2 and 3.3.3 for details of the methods

2016). When the number of probes exceeds the number of modes used for reconstruction the problem is over defined and the QR-based method requires additional treatment (Manohar et al., 2018). As the present study will investigate reconstructions using non-linear methods (in addition to linear), the optimal placement is not trivial. The scope of the present study will be limited to considering probes placed randomly (the sub-optimal case) and using the Q-DEIM with pivoting (to approximate the optimal placement). For the case of randomly placed probes, one set of random locations is used (as opposed to testing multiple sets of random locations). This is due to the difficulty in generating and storing a large number of random global probe libraries, the need for which is presented in section 3.3.2.2.

3.3.2 Reconstruction

The accuracy of the reconstruction is a matter of approximating the real POD coefficients of the full velocity fields using a set of "dynamic" coefficients A_{DYN} that are estimated via the sparse probes signals U_p. For the present analysis, the probe signals are obtained via $U_p = UC$, where C is a Boolean matrix known as the *sparse matrix* of size

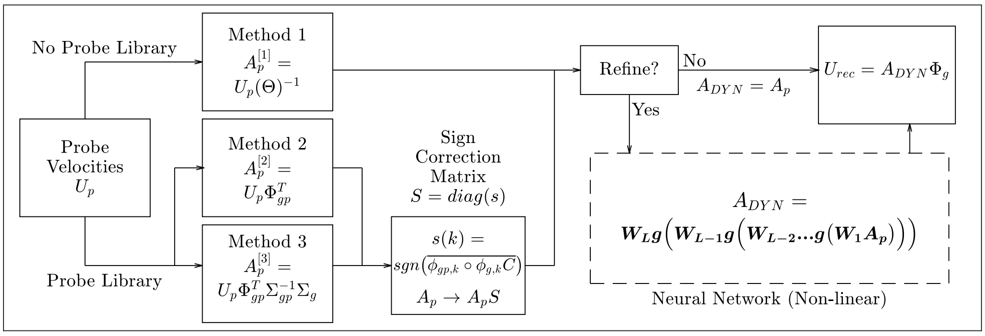

No Probe Library
Method 1
$A_p^{[1]} = U_p(\Theta)^{-1}$
Refine?
No
$A_{DYN} = A_p$
$U_{rec} = A_{DYN}\Phi_g$
Yes
Probe Velocities U_p
Method 2
$A_p^{[2]} = U_p\Phi_{gp}^T$
Sign Correction Matrix
$S = diag(s)$
$A_{DYN} =$
$\boldsymbol{W_L g\Big(W_{L-1} g\Big(W_{L-2}...g(W_1 A_p)\Big)\Big)}$
Method 3
$A_p^{[3]} = U_p\Phi_{gp}^T\Sigma_{gp}^{-1}\Sigma_g$
$s(k) =$
$sgn(\overline{\phi_{gp,k} \circ \phi_{g,k}C})$
$A_p \rightarrow A_pS$
Probe Library
Neural Network (Non-linear)

FIGURE 3.3: Block diagram of the reconstruction starting from the probe signal U_p.

$2n_x \times p$, where p is the number of probes. Each column of the sparse matrix contains a single entry equal to one and zero elsewhere; providing a map that subsamples from all spatial locations x to the sparse locations of the probes x_p (Jayaraman et al., 2019). With the POD coefficients estimated, the reconstruction is obtained via the global POD basis

$$U_{rec} = A_{DYN}\Phi_g, \tag{3.1}$$

or, more explicitly, for a up to a specified number of modes at a particular instant:

$$U_{rec}(t) = \sum_{1}^{k} a(t)_{DYN,k}\phi_{g,k} \tag{3.2}$$

where Φ_g are the *global* spatial modes from the POD performed on a set of training data U_g. There lies an implicit assumption in the application of equation 3.2 such that the external state of the system is unchanged during the reconstruction compared to the conditions under which the global basis was tabulated. The instantaneous state must remain within the bounds of the variation captured by the training data. In other words, using the aerofoil as an example, variations in the fixed free stream velocity or angle of attack are not accounted for. As such, the scope of the methodology is limited to testing the efficacy of the sparse reconstructions without external changes to the state of the system.

For the present study, the training data consists of 9000 samples. The training data is used to generate the time-independent global basis Φ_g in order to predict the complete spatio-temporal evolution of the flow based on sparse time-resolved measurements via equation 3.2. The large training data set is needed due to the high spatio-temporal variability of the turbulent separated flow, leading to a slowly decreasing set of singular values Σ_g compared to e.g. a laminar flow (figure 3.1) and therefore many modes are required within the global basis. The global POD was calculated accordingly using $n_t =$

9000 and $2n_x = 50,952$, resulting in a computation time of 418 seconds (7 minutes) on a single desktop computer (double precision with 3.6GHz CPU and 16GB RAM). It was determined that a larger set of training data was not required as the global POD library was found to be satisfactorily converged up to the maximum number of modes used for reconstructions at $k = 500$. As such the accuracy of the reconstructions depends only on how closely the dynamic coefficients A_{DYN} approximate the real ones up to the number of modes used for the reconstructions. The time-resolved reconstruction was then tested on all 5468 samples of the time-resolved data set, resulting in a training-to-validation ratio of approximately 1.65.

There are several methods used in order to approximate the coefficients that will be outlined in the following subsections. For all of the methods, provided that the global POD and optimal placement calculations are performed using a training data set *a-priori*, the calculations are possible to be performed in real-time. This is conceptually illustrated in figure 3.2. These methods are therefore highly relevant to flow-sensing and control applications. The methods for approximating the coefficients originating from the probes A_p can either be used for reconstruction immediately, or improved using a shallow neural network (figure 3.3).

3.3.2.1 Method 1: sparse recovery reconstruction

The first method used is the most common implementation of sparse recovery originating from the compressed sensing literature (Donoho, 2006; Candès and Wakin, 2008; Callaham et al., 2019). This method allows to calculate a set of coefficients that approximate the real coefficients by taking advantage of the fact that many entries of the global basis are negligibly small (c.f. Jayaraman et al., 2019). With this method, the POD coefficients are estimated by projecting the probes signal into a sparse basis corresponding to the global POD modes evaluated at the locations of the probes

$$A_p^{[1]} = U_p \Theta^{-1} \tag{3.3}$$

where the bracketed superscript denotes the method used. Here, Θ is the sparse basis formed by the product of the global basis with the sparse matrix $\Theta = \Phi_g C$. The number of rows in A_p corresponds to the number of samples (rows) of U_p, and the number of columns is the number of probe modes. As we centre our analysis around the Q-DEIM placement method described in section 3.3.1, the sparse basis Θ is a square matrix with the number of probes p equal to the number of reconstruction modes k.

As mentioned in section 3.3.1, the optimal locations of the probes are the locations within Φ_g that minimise the matrix condition number of Θ. In other words, locations within the global basis with very little variance across the modes will lead to error propagation and low signal to noise ratio (SNR) upon the inversion in equation (3.3). Therefore, the best locations will correspond to the locations with the largest variance across the modes of the global basis.

3.3.2.2 Method 2: extended POD reconstruction

Motivated by studies that seek to approximate the POD coefficients using either a different variable or a separate measurement, extended POD (Borée, 2003) offers a flexible framework with which to approximate the coefficients. Borée (2003) showed that by projecting the temporal modes of one quantity into the measurements of another, one obtains a set of extended spatial modes that can be used as a basis to reconstruct the part of the measurements that is correlated to the quantity of interest. When all modes are included, this is equivalent to the well known LSE (LSE: Adrian and Moin, 1988; Lasagna et al., 2013; Hosseini et al., 2015).

For the present study, a framework inspired by extended POD is implemented treating the probes as a quantity to be correlated to the full flow field. If the assumption is made that the coefficients of the probes are highly correlated with the coefficients of the flow field, one can use the same training data used to calculated the global POD basis Φ_g to produce a global *probe* POD basis Φ_{gp}. The coefficients are then obtained as

$$A_p^{[2]} = U_p \Phi_{gp}^T \tag{3.4}$$

where the transpose is used on the global probe basis as it is orthogonal by construction. The global probe basis is generated for all tested probe numbers and positions across all 9000 training samples used to construct the global libraries. For the sake of memory allocation, only one set of random probe locations is tested to avoid calculating and storing hundreds of additional global probe libraries. It was confirmed after testing two other sets of random placements that the results presented hereafter were qualitatively unaffected.

3.3.2.3 Method 3: quasi-orthogonal extended POD reconstruction

Building on the methodology of the previous subsection, the quasi-orthogonal extended POD reconstruction utilises all of the available information from the global POD and global probe POD libraries to produce the reconstruction. Instead of simply assuming that the coefficients produced from the global probe modes will approximate the real coefficients, instead a quasi-orthogonal basis Ψ_p is calculated using the pseudoinverse of the singular values of the global probe modes Σ_{gp}^{-1}. The term "quasi" is used to emphasise that there is no guarantee that the resulting basis will be perfectly orthogonal

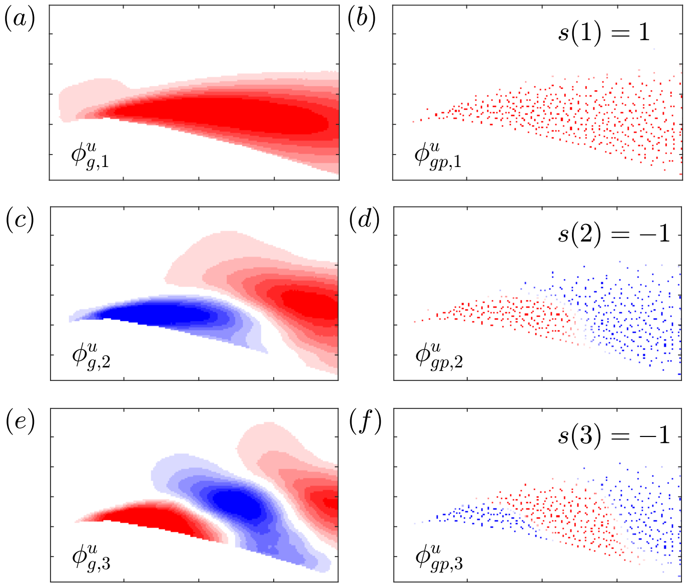


FIGURE 3.4: Global spatial modes (a,c,e) and corresponding global spatial probe modes (b,d,f) of u' mapped to the locations of the full field for 500 probes placed using the Q-DEIM. The colourbars range across $\pm 3\sigma$ of the corresponding global modes $\phi^u_{g,k}$ from blue to red. Modes with spatial locations that are in phase give a sign correction $s = 1$ (a-b) and out of phase $s = -1$ (c-f)

as it is constructed using the training data and subsequently projected into by independent time-resolved testing data. The coefficients can then be approximated using the global singular values to re-scale the quasi-orthogonal basis as

$$A_p^{[3]} = \overbrace{U_p \Phi_{gp}^T \Sigma_{gp}^{-1}}^{\Psi_p} \Sigma_g. \tag{3.5}$$

A similar principle was used by Discetti et al. (2018) to up-sample time-resolved fields using simultaneous probe signals at a higher sampling frequency.

We remark that the extended-POD methodologies presented here represent a novel departure from previous studies, e.g. Hosseini et al. (2015). In their case, the temporal information contained within the pressure probes were used to reconstruct the velocity fields using a basis determined by the extended spatial velocity modes. The present work, by contrast, uses the global POD basis for the reconstruction and estimates the coefficients using a separate POD basis tailored for the probes. The POD basis tailored for the specific sets of probes Φ_{gp} is the main novelty of the present approach. As the probes themselves are velocity measurements, the underlying assumption here is in the inherent correlation between the probe velocities, their coefficients, and the full-field velocities.

3.3.2.4 Sign correction matrix

For the extended POD based methods of the two previous subsections, a new global probe basis is used to obtain an approximation of the coefficients. The velocity autocorrelation of the probes (the underlying physical quantity used to calculate the POD) is missing information due to the sparsity of the probe signals. This may necessarily lead to spatial probe modes Φ_{gp} that, when mapped to the locations of the full field modes Φ_g, have opposite sign. Especially for modes corresponding to the largest singular values, this can lead to reconstructions that are anti-correlated with the true underlying velocity fields. This effect is illustrated in figure 3.4, showing the first three modes for the fluctuating horizontal velocity component from the global library and the global probe library side by side for the case of 500 probes placed using Q-DEIM.

To account for this effect, a sign correction matrix calculation is proposed to be calculated *a-priori* to ensure that the two spatial bases Φ_{gp} and Φ_g are not anticorrelated (or in other words, as correlated as possible):

$$s(k) = \mathrm{sgn}\left(\overline{\phi_{gp,k} \circ \phi_{g,k}C}\right), \tag{3.6}$$

where the overline denotes spatial averaging across the locations of the probes and the $\circ$ symbol denotes the Hadamard (element-wise) product. The full diagonal (and orthogonal) sign correction matrix is then

$$S = \mathrm{diag}(s) \tag{3.7}$$

where the diag function outputs a square matrix of zeros with the entries of the vector argument s along the diagonal. The coefficients are then updated through multiplication with S as shown in figure 3.3.

3.3.3 Shallow neural network refinement

As outlined in figure 3.3, the option to apply non-linear refinement via a neural network is explored. Here we briefly review literature concerning neural network size to frame the specific implementation utilised in this study. Early on, Cybenko (1989) provided proof for the universal approximation theorem via neural networks of arbitrary width with a single hidden layer and a sigmoidal activation function. A remaining issue however was in the capability to train networks with a large number of nodes in a single layer, thereby ceding computational advantage to unsupervised methods such as POD. Further generalisation of the universal approximation theorem was later shown for different activation functions and multiple layers (Hornik, 1991). In combination with

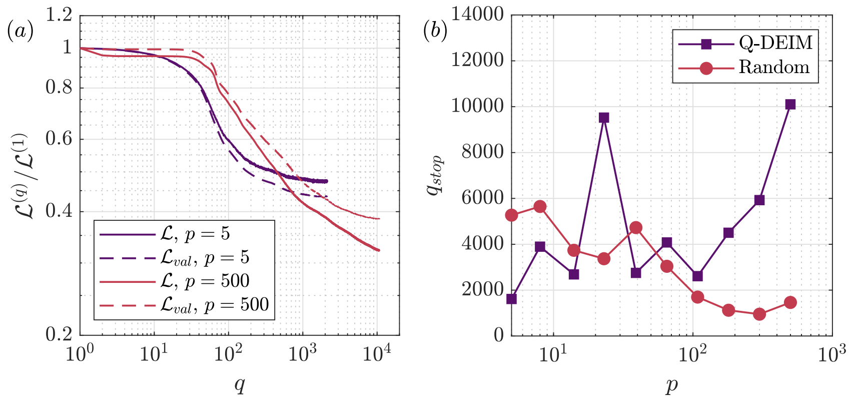


FIGURE 3.5: Normalised training and validation loss $\mathcal{L}^{(q)}/\mathcal{L}^{(1)}$ within the SNN for method 1 using 5 and 500 probes and Q-DEIM placement vs number of epochs q (a) and number of epochs before stopping q_{stop} vs number of probes for method 1 for each placement (b)

new training methods, the differences between networks of a single layer with many nodes or multiple layers with less nodes has since faded (Hinton et al., 2006). Deep neural networks with multiple layers remain commonplace, however fewer layers can provide similar results taking advantage of modern hardware and training methods to significantly reduce the required training. Although single layer neural networks are easier to interpret, it has been shown that networks with two layers provide better generalisation capabilities (Thomas et al., 2017). Networks with few layers are typically less sensitive to the specific choices of hyperparameters than their deep counterparts. As such, a shallow neural network (SNN) is adopted for the present study.

We apply an SNN architecture as outlined in the work of Erichson et al. (2020). In general, we seek a function $\mathcal{F}$ that consists of multiple fully connected or convolutional layers with associated scalar non-linear activation functions g and weights W applied to input A

$$\mathcal{F}(A;W) := W_L g(W_{L-1} g(W_{L-2} \cdots g(W_1 A))) \tag{3.8}$$

where a shallow network has a small number of layers L. For the present study $L = 3$ is used, where two layers are hidden and followed by an output layer. Due to the spatial sparsity of the inputs in the present study, we favour the fully connected layers over the convolutional layers in line with similar investigations (Erichson et al., 2020; Nair and Goza, 2020).

For the SNN architecture, shown in table 3.1, we construct a first "imagination" layer with $1.5p$ nodes followed by a $1.2p$ "refinement" layer with a 5% dropout layer in between to improve the generalisation (Erichson et al., 2020). As the network scales with the input size p, the amount of trainable parameters varies significantly for different networks considering varying amounts of probes. The training data however has a consistent amount of samples. This requires the larger networks to utilise additional generalisation. This is achieved by the 5% dropout layer (and leaves the small networks largely unaffected). The final linear layer contains the same number of inputs (number of probes/modes) as outputs. For the activation functions the rectified linear unit function (ReLU) is used for the hidden layers and a linear function for the output layer. The optimisation is performed using the ADAM adaptive moment optimisation (Kingma and Ba, 2015). The learning rate is set to 0.001, with the exponential decay rate for the first moment estimates equal to 0.9 and the exponential decay rate for the second moment estimates equal to 0.999. For numerical stability, the recommended $\widehat{\epsilon} = 10^{-7}$ value is used.

TABLE 3.1: The neural network architecture. The dropout layer is only active during training.

Layer	Layer type	Number of nodes	Activation
input	input	p	linear
hidden 1	dense	$1.5p$	ReLU
(hidden)	dropout	$0.05 \cdot 1.5p$	linear
hidden 2	dense	$1.2p$	ReLU
output	dense	p	linear

The same training data used to obtain the POD basis Φ_g is used to train each neural network $\mathcal{F}$ using 89% of the samples $n = 8000$, reserving $m = 1000$ random samples for iterative learning validation. As demonstrated by Guastoni et al. (2021), the neural network for the present study is trained to recover the POD coefficients. The global coefficients for training the outputs are obtained as $A_g = U_g \Phi_g^T$ up to $k = p$ modes (where p is the number of probes), and the sparse probe coefficients $A_{gp}^{[i]}$ using the i-th linear method applied to the training data $U_g C$ as the inputs. For each set of probe locations and for each linear method, a separate network and corresponding set of weights is trained to minimise the loss

$$W = \underset{\widetilde{W}}{\operatorname{argmin}}\, \mathcal{L}(\widetilde{W}) \tag{3.9}$$

where the loss is defined for each epoch q as

$$\mathcal{L}^{(q)}(\widetilde{W}) = \sum_{j=1}^{n} \left|\left| A_g(t_j) - \widetilde{\mathcal{F}}\left(A_{gp}^{[i]}(t_j); \widetilde{W}\right) \right|\right|_1 \tag{3.10}$$

where $A(t_j)$ indicates the vector of coefficients corresponding to the j-th training sample and the $\tilde{\cdot}$ indicates a dummy function or variable used for training. The l_1 norm is chosen in this study for evaluating the loss as it was found to outperform the commonly used l_2 norm. The use of the l_1 norm (in combination with unscaled coefficients, as will be discussed) aids the network in prioritising an error reduction on the coefficients corresponding to the modes with larger singular values. In an effort to maintain the singular value scaling of the corresponding temporal modes, the coefficient inputs and outputs are not scaled to a zero-to-one range as is often done to speed up learning. This appears to have a minor influence on the training only affecting initial optimisation epochs where the optimiser must adjust before reducing the loss (figure 3.5). The use of batch normalisation for generalisation was omitted and instead large batches of 3000 samples were used. These have the added benefit of speeding up training. Other methods such as layer activation regularisation or layer weight regularisation were explored but were found to produce less consistent results over the various neural networks without repetitive hyperparameter optimisation.

It is possible to include additional constraints on the loss function in equation 3.9 in order to keep the network more generalised (Erichson et al., 2020; Sun and Wang, 2020). Instead for this study early stopping was opted for to avoid overfitting. The trained network was evaluated on the validation subset every epoch and for every iteration where the loss decreased the weights were stored. If the network did not experience a new minimum in the validation loss within 500 epochs, the training was halted and the last stored weights adopted. The value of 500 was chosen ad-hoc for this particular study as it was found to strike a balance between "kick-starting" the network with enough initial epochs but was large enough to revert any apparent overfitting. This is shown in figure 3.5a for $p = 5$ and 500 probes via input coefficients from method 1. With this architecture and early stopping criterion, the number of epochs was on the order of a few thousand. The number of epochs increased with increasing probes to a maximum of approximately 10 thousand epochs for the Q-DEIM positioning (figure 3.5b). For the random probe positioning the number of epochs decreases from approximately 5000 for 5 probes to approximately 1500 for 500 probes.

We remark that the reconstructions use the same number of modes as probes, therefore for the present study the more probes used for the inputs the greater the performance ceiling for the SNN (up to the limit of the real coefficients truncated at p modes).

3.3.4 Quantification

Two metrics are utilised to quantify the reconstruction accuracy in the present study. To quantify the phase properties of the reconstruction, the normalised correlation is defined as

$$\rho_{u'} = \frac{\langle u'(x,t)u'_{rec}(x,t)\rangle}{\langle u'(x,t)^2\rangle^{1/2}\langle u'_{rec}(x,t)^2\rangle^{1/2}} \tag{3.11}$$

where the angled brackets $\langle \cdot \rangle$ indicated averaging over all space and time and here u'_{rec} is the reconstruction of the horizontal fluctuating velocity. The correlation takes a value of 1 for a reconstruction that is completely in phase, -1 when it is out of phase, and 0 when it is uncorrelated. To evaluate the overall difference between the original and reconstructed fields, the root-mean-square error is defined as

$$e_{u'} = \frac{\left\langle \left(u'(x,t) - u'_{rec}(x,t)\right)^2 \right\rangle^{1/2}}{\langle u'(x,t)^2 \rangle^{1/2}}. \tag{3.12}$$

The root-mean-square error describes the fraction by which the reconstruction differs from the original fields, with a perfect reconstruction at $e = 0$. Equations 3.11 and 3.12 are analogously defined for the vertical fluctuations v'.

3.4 Results

The results are presented first using the linear methods alone, followed by the non-linear SNN refinement as outlined in section 3.3 and figure 3.3. The reconstructions evaluate the ability of the linear and non-linear methods to predict instantaneous flow fields using the testing data consisting of 5468 samples. For all results, the number of modes k used in the reconstructions is equal to the number of probes p. This was chosen based on the underlying principles of the calculation for the optimal placement using the Q-DEIM (see section 3.3.1), revealing optimal probe locations for reconstruction method 1 specifically (section 3.3.2.1). These were found to correspond to locations within the separated region of the flow (figure 3.4). We remark, however, that the other linear methods (methods 2 and 3) need not necessarily use all $k = p$ modes in their reconstructions. For the present study we opt to present $k = p$ for all methods for consistency of comparison.

3.4.1 Linear reconstruction

The normalised root-mean-square error and correlations for the three linear methods are presented for each component of the fluctuating velocity in figures 3.6 and 3.7 for Q-DEIM and random placement, respectively. The best possible performance is indicated in both figures by the reconstruction calculated using the rank-truncated POD via equation 2.6. It is immediately apparent that the maximum possible performance with the maximum number of probes $p = 500$ corresponds to a normalised root-mean-square error of 16% and 24% and correlations of 0.99 and 0.96 for u' and v', respectively. This highlights the challenge of applying a reduced order model such as POD to a turbulent flow as in the present case; resulting in many required modes to capture the fluctuations.

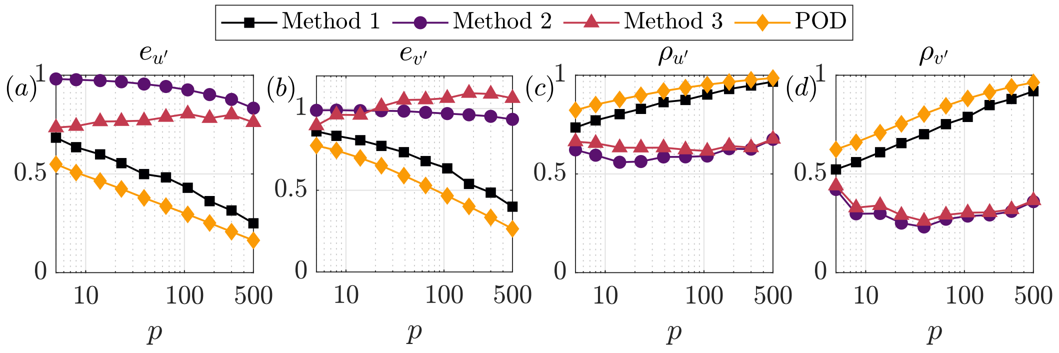


FIGURE 3.6: Normalised root mean square error (a,b) and correlations (c,d) vs number of probes for u' (a,c) and v' (b,d) using the Q-DEIM for probe placement applied to the testing data via method 1 (squares), method 2 (circles), method 3 (triangles) and POD (diamonds). The number of reconstruction modes k is equal is the number of probes used p. The POD-based reconstructions are obtained via equation 2.6 using the coefficients from projecting the full NACA 0012 velocity fields into the global basis

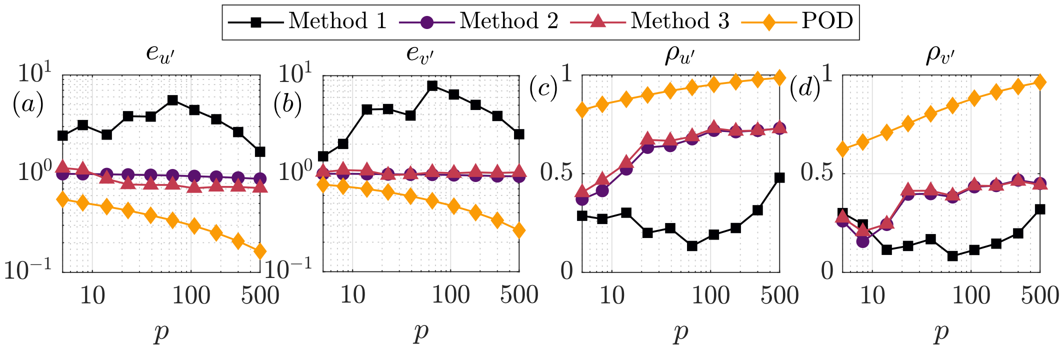


FIGURE 3.7: Normalised root mean square error (a,b) and correlations (c,d) vs number of probes for u' (a,c) and v' (b,d) using random probe placement applied to the testing data from via method 1 (squares), method 2 (circles), method 3 (triangles) and POD (diamonds). The number of reconstruction modes k is equal is the number of probes used p. The POD-based reconstructions are obtained via equation 2.6 using the coefficients from projecting the full NACA 0012 velocity fields into the global basis

The results in figure 3.6 highlight the differences in the three proposed methods when the probes are placed using Q-DEIM. For this placement, the sparse recovery via method 1 significantly outperforms the other methods. At 500 probes, method 1 was able to recover the spatio-temporal fluctuations to within 25% and 40% for u' and v', respectively, with correlations exceeding 90%. The relative superiority of method 1 is enforced by construction, as the Q-DEIM ensures that method 1 produces coefficients that most efficiently sample the POD basis. Methods 2 and 3 show mixed results. These approaches were only able to recover the fluctuations with approximately 75% error for u' and even exceeding 100% for v' with correlations in the approximate range of of 30-60%. Both methods produce reconstructions with very similar correlations. This is unsurprising, as both methods use the same global probe basis. Method 3, on the other hand, effectively contains a re-scaling of the coefficients by a factor of $\Sigma_{gp}^{-1}\Sigma_g$, resulting in notable differences in the root-mean-square error. For u', this leads to comparatively lower error using method 3. However for v', method 3 quickly compares worse to method 2 beyond 14 probes.

We remark that although the experimental data presented is inherently limited in its accuracy and contains noise, no systematic variation of noise on the fidelity of the reconstructions is presented. This is due to the underlying assumption that the measurement noise is largely confined to low-energy high-rank structure that was explicitly treated for in the data post-processing (section 2.1.1.2). The cut-off mode number for treating the random noise was found to be higher than the maximum number of modes tested for the sparse reconstructions. Nevertheless, the limited sub-pixel accuracy of the PIV (Adrian et al., 2011) should be considered in the interpretation of the results. As can be seen in figure 3.7, the vertical velocity fluctuation v' was found to be consistently more difficult to reconstruct across methods. This is likely due to the reduced signal-to-noise ratio of v' in the measurements, for which the vertical fluctuations span a smaller range of pixel values (average deviation $\sigma_{std,v} = 1.2$ pixels) than the horizontal ($\sigma_{std,u} = 1.8$ pixels) and are therefore more greatly effected by sub-pixel accuracy in the PIV.

An interesting shift in the results occurs when the probes are placed randomly as shown in figure 3.7. The sparse recovery approach of method 1 now performs significantly worse than methods 2 and 3. This is due to the fact that the random probes have some locations outside of the shear layer. These locations contribute very little variance to the global POD basis, and when they are inverted (equation 3.3) the singular values become large and amplify the noise as a result. This points to the necessity that reconstructions obtained via method 1 must avoid probe locations with very little variance across the global basis. For the case of a separated aerofoil, this means avoiding probes outside of the separated region. On the other hand, the random placement does not greatly impact the extended-POD based methods 2 and 3. Both methods perform similarly between the random and Q-DEIM placements, suggesting they may be more robust to arbitrary placement. Methods 2 and 3 do not suffer from the same effect as method 1 because they use an independent global probe POD basis that is tailored to the sets of probe locations using individual decompositions obtained *a-priori*.

The extended-POD methods appear more robust for the present PIV data of a separated aerofoil at high Reynolds number, but a natural question to ask is whether such robustness extends to the laminar case. To assess this, we use the DNS data of a laminar cylinder in crossflow at $Re_D = 100$ provided by Brunton and Kutz (2019) and perform sparse reconstructions of the vorticity as described by Manohar et al. (2018). A brief outline of the analysis is as follows. The laminar data consists of 151 snapshots for which the mean is subtracted to isolate the fluctuations. The first 100 snapshots are used to generate the global basis and the global probe basis using POD for both Q-DEIM and random placements (see figure 3.1c). The remaining 51 snapshots are used

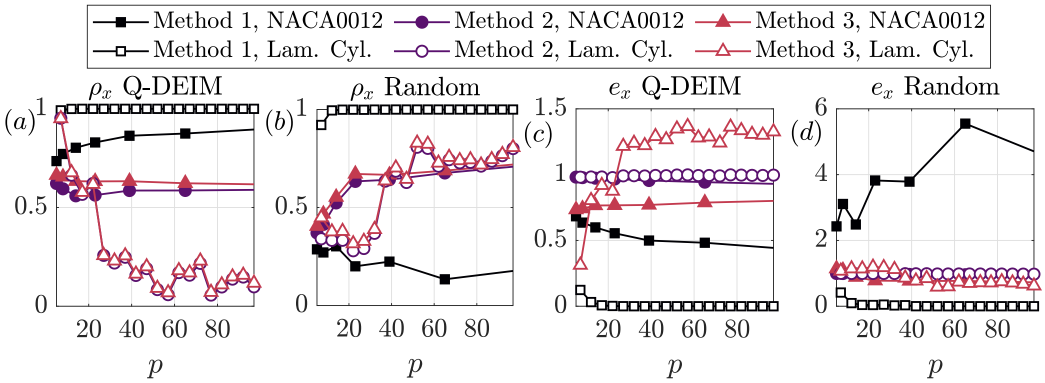


FIGURE 3.8: Comparison of normalised reconstruction correlation (a,b) and root mean square error (c,d) for the reconstruction of u' in the present case (filled symbols, $x = u'$) to the laminar cylinder at $Re_D = 100$ from the DNS of Brunton and Kutz (2019) (unfilled symbols, $x = \omega$) vs number of probes using Q-DEIM (a,c) and random placement (b,d) via method 1 (squares), method 2 (circles), and method 3 (triangles)

for assessing the reconstructions quantified using the same metrics as the present study (section 3.3.4).

The comparison to the laminar cylinder is presented in figure 3.8. The results from the laminar cylinder are shown with unfilled symbols and from the present study for u' with filled symbols using Q-DEIM and random placement. Method 1 outperforms methods 2 and 3 for reconstructing the laminar cylinder for all probe numbers and locations. Interestingly, for very few probes ($p = 7$) methods 2 and 3 correlate reasonably well for the Q-DEIM placement around the cylinder. However, as the number of probes increases the correlation rapidly decreases. It is observed that as the number of probes increases, the optimal locations for the probes shift from within the wake region to concentrating near the leading edge of the cylinder. This appears to negatively impact the ability for extended POD to capture the correlation between probes and flow. The random placement however appears to give improved correlations for methods 2 and 3 compared to the Q-DEIM; indicating that the extended POD benefits from the dispersed probe locations.

The comparison between the reconstructions in the present study and with the laminar cylinder indicates that the compressed sensing methodology is superior for systems captured by relatively few modes. This is because the compressed sensing methodology does not rely on underlying correlations between probes, but rather how their location contributes to the global basis and its corresponding sparse inversion. By contrast, the improvement in the extended POD based methods between Q-DEIM and randomly placed probes suggests that the extended POD relies more heavily on the underlying spatial correlations captured by the POD of the global probe basis. Indeed, it was confirmed (not shown) that spatially concentrated probes in the wake of the cylinder yielded significantly poorer reconstructions than spatially dispersed configurations. This effect may explain why the extended POD methods are more robust for

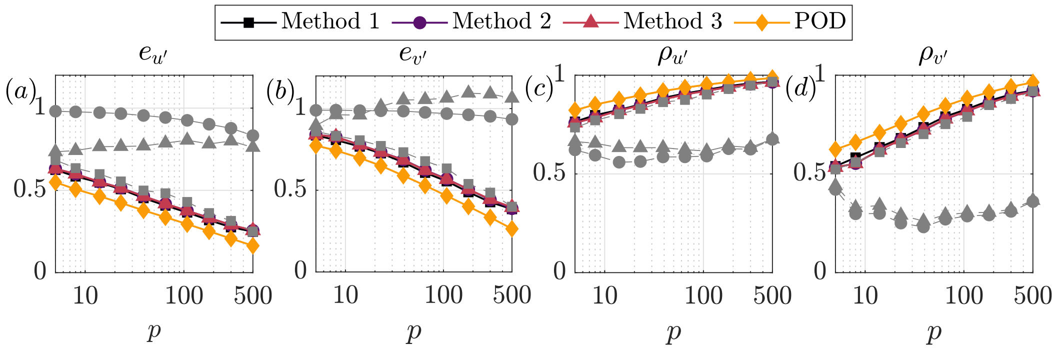


FIGURE 3.9: Normalised root mean square error (a,b) and correlations (c,d) vs number of probes for u' (a,c) and v' (b,d) using SNN refinement and the Q-DEIM for probe placement applied to the testing data from via method 1 (squares), method 2 (circles), method 3 (triangles) and POD (diamonds). The linear results are shown in gray dashed lines with corresponding symbols. The number of reconstruction modes k is equal is the number of probes used p. The POD-based reconstructions are obtained via equation 2.6 using the coefficients from projecting the full NACA 0012 velocity fields into the global basis

the separated aerofoil. The spatially dispersed probes capture a portion of the underlying spatial correlations whilst the compressed sensing approach of method 1 may be sensitive to specific regions with low variance across the global basis.

3.4.2 Non-linear reconstruction

The root-mean-square error and correlations using the non-linear SNN for refinement are presented in figures 3.9 and 3.10 for the Q-DEIM and random placements, respectively. In both figures, the grey dashed lines indicate the reconstructions without applying the SNN. Regardless of which linear method is used to supply the input for the SNN, the output coefficients perform nearly identically for each placement. The Q-DEIM placement appears to slightly outperform the random placement; pointing to the importance of the underlying correlations between the input and output coefficients on their performance of the SNN. Only for very large numbers of probes using the random placement can some small differences arise between input methods (figure 3.10). In any case, the SNN outperforms all of the linear methods both in terms of correlation and root-mean-square-error. The SNNs are trained using a loss function that is designed to minimise the difference between the known POD coefficients and the output of the SNN. It is therefore unsurprising that it outperforms the linear counterparts, although it does not fully recover the true underlying POD coefficients for either placement.

In order to understand more about how the SNN is performing compared to the linear methods, the singular values from the estimated coefficients A_{DYN} are extracted. This separates the orthogonal temporal part of the coefficients in order to reveal the *effective* singular values used for the full reconstruction. This is presented for the linear methods

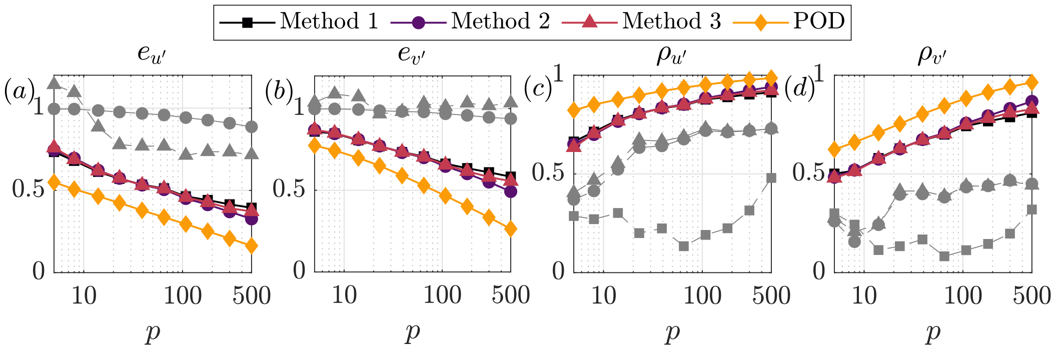


FIGURE 3.10: Normalised root mean square error (a,b) and correlations (c,d) vs number of probes for u' (a,c) and v' (b,d) using SNN refinement and random probe placement applied to the testing data from via method 1 (squares), method 2 (circles), method 3 (triangles) and POD (diamonds). The linear results are shown in gray dashed lines with corresponding symbols. The number of reconstruction modes k is equal is the number of probes used p. The POD-based reconstructions are obtained via equation 2.6 using the coefficients from projecting the full NACA 0012 velocity fields into the global basis

and with SNN refinement for the case $p = 14$ in figure 3.11 for Q-DEIM and random placement. The results are seemingly counter-intuitive. For both placements, the linear method 3 appears to most closely approximate the true singular values of the validation run, but this does not mean that it will necessarily lead to a better reconstruction. In fact, this result is unsurprising as method 3 is constructed to have a quasi-orthogonal set of temporal probe modes that are re-scaled using the global ones, and therefore will most closely resemble the true singular values upon extraction from the coefficients. The other two methods on the other hand do nothing explicit to regulate the orthogonal basis of the coefficients, and this is manifested by singular values in figure 3.11 that depart from the true ones.

Regardless of the linear method used, the singular values from the output of the SNN collapse together and consistently underestimate the true ones. It is important to note that no orthogonal regularisation was imposed on the SNN in its construction. Instead it simply attempts to minimise the loss function between the training inputs and the training data by adjusting the weights. As the root-mean-square error and correlations unilaterally improve via the SNN as shown in figures 3.9 and 3.10, the results of figure 3.11 imply that recovering the underlying temporal correlation (the phase information of the temporal modes) is inherently more important than recovering the precise singular values (at least for minimising the imposed loss function). This becomes increasingly less important as the mode number increases, as evidenced by the sharp decrease in singular values with increasing k. This shown for the case of $p = 14$, however similar results were found for all probe numbers tested.

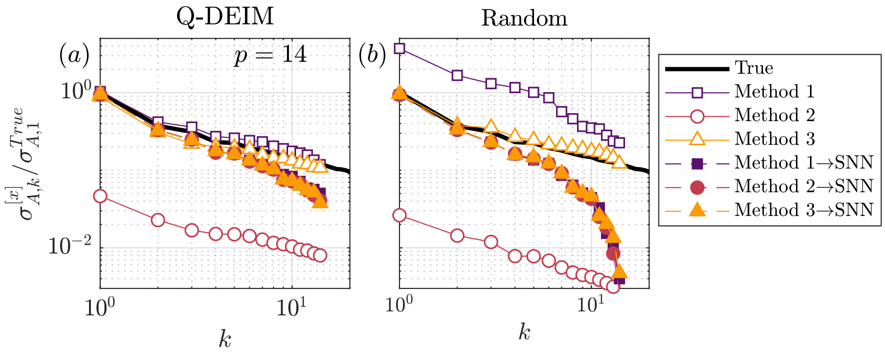


FIGURE 3.11: Singular values $\sigma_{A,k}$ extracted from the estimated coefficients A_{DYN} normalised by the leading order singular value of the true coefficients for $p = 14$ probes using linear methods (unfilled symbols) and SNN refinement (filled symbols) via Q-DEIM (a) and random placement (b)

3.5 Conclusions

We have presented three distinct linear reconstruction methodologies and a shallow neural network (SNN) for non-linear refinement to obtain state estimations of separated turbulent flow over a NACA 0012 aerofoil at $\alpha = 12°$ and $Re_c = 75000$ using time-resolved PIV from limited probes. Each methodology was trained using 9000 samples of PIV training data and instantaneous sample prediction was tested using all 5468 samples of an independent time-resolved data set. This is the first systematic investigation of sparse reconstructions for a moderately high Reynolds number advective turbulent flow obtained via experiment.

The reconstructions were first tested through the use of linear methods alone. It was found that the performance of the linear methods depended largely on the choice of placement. When the placement was chosen carefully using the Q-DEIM, the compressed-sensing approach of method 1 greatly outperformed the extended POD based approaches of methods 2 and 3 (figure 3.6). When the probes were placed randomly (figure 3.7), it was found that the extended POD methods performed move favourably than the compressed sensing approach of method 1. This was due to locations within the global POD basis outside of the shear layer with small variance causing errors in the inversion of the sparse basis for method 1. This is illustrated clearly in figure 3.12c and e, showing how the reconstruction is affected when probe locations reside in the free stream.

In order to investigate how the methods compare for the laminar case, the extended POD methods were applied to the DNS of laminar flow over a cylinder at $Re_D = 100$ of Brunton and Kutz (2019) (figure 3.8). It was found that the extended POD based methods rely on the underlying correlations of their global probe POD basis. When the probes were grouped closely together, the reconstructions performed more poorly. When the probes were more dispersed, the autocorrelation underlying the global probe

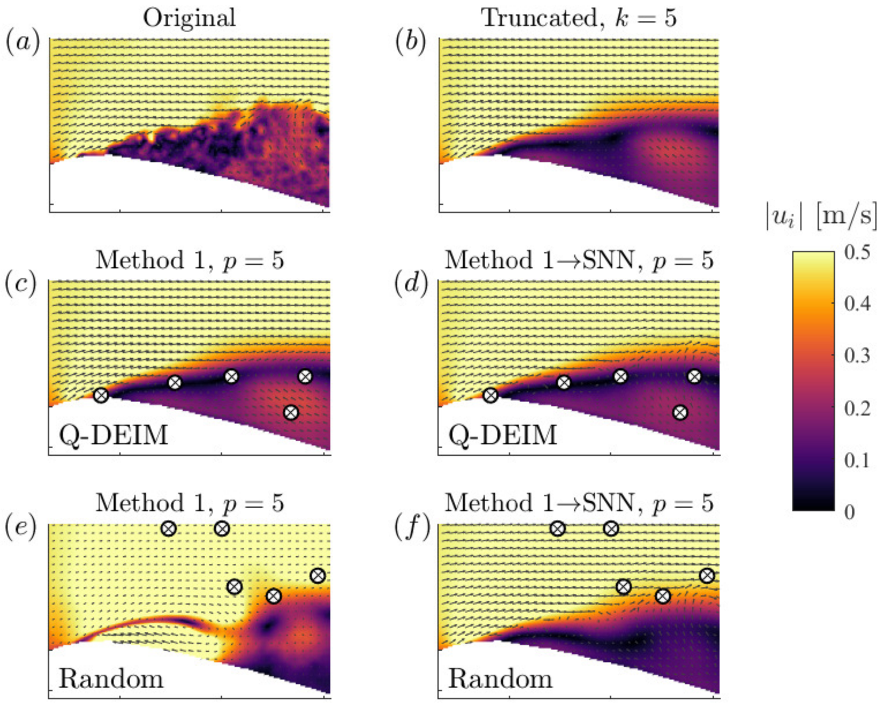


FIGURE 3.12: Original (a), POD-based (b), linear via method 1 (c,e), and non-linear (d,f) reconstructions of the total velocity at one arbitrary instant using 5 probes and Q-DEIM (c,d) and random (e,f) placement with every fifth velocity vector shown for clarity. Vectors are scaled automatically with respect to their individual fields and not across panels

POD basis was able to capture more features. Despite this, the compressed sensing approach of method 1 outperformed the extended POD approaches for both Q-DEIM and random placement in the laminar case. As the laminar data did not contain measurement noise and was captured by relatively few modes, small variances within the global basis were comparatively less problematic.

Non-linear refinement of the estimated coefficients was tested through the application of an SNN using the various linear methodologies as inputs. The performance of the SNN was found to be nearly identical regardless of the linear method used and slightly more favourable for Q-DEIM placement (figures 3.9 and 3.10). In all cases, the SNN improved the root-mean-square error and correlations of the reconstructions. Interestingly, it was found that the output coefficients of the SNN did not contain singular values that most closely matched the true singular values (figure 3.11). Instead, the SNN naturally placed more emphasis on the phase information of the underlying temporal correlations; leading to improved reconstruction performance.

The implications for sparse reconstructions in a rapidly evolving turbulent flow (the most common type of flow in engineering applications) based on the findings of the present study can be summarised as follows. Firstly, several hundred modes (and therefore probes) are required to capture the velocity fluctuations using POD as the global basis. This implies an extensive full-field data set must be tabulated *a-priori*. In this case 18 times more samples than the maximum modes used for reconstruction (9000 samples and 500 modes). Many fewer probes may be used for reconstructions depending on the desired accuracy for the application of interest (figure 3.6). Second, the placement of the probes depends on the reconstruction method of choice. The compressed sensing approach based on sparsity (method 1) is ideal but relies on chosen locations within the flow with high variance. The extended-POD approaches (methods 2 and 3) are less optimal but comparatively more robust and instead depend on being spatially dispersed. Finally, the non-linear refinement via the SNN appears to equalise the linear approaches, with only a small improvement when placed favourably. This result begs the question: why should one bother with linear methods at all? The answer again depends on the application of interest. If the objective is to reconstruct the full-field as accurately as possible our results indicate one should employ as many sensors as possible and use any linear method for the inputs to the SNN. This requires training the network with the entirety of the full-field data obtained *a-priori*. If the application instead does not require very high accuracy, the expensive task of training a neural network may be avoided in which case the differences between the linear methods become relevant once again.

This study has demonstrated the capability of linear and non-linear methods to recover instantaneous reconstructions of the full experimental velocity fields using sparse measurements. This is visualised in figure 3.12. Avenues for future work include sparse reconstructions with flow control within the sensing region (which would require a distinct global basis depending on if the flow control is active or not). A further exploration of sensitivity for variations in the system e.g. changes in the freestream velocity U_∞ or angle of attack α is warranted, as the cost of training separate global bases for variations in external conditions is undesirably high (for the present study, this implies separate training data sets on the order of thousands of samples each for each α). Future investigations may also explore variations of the SNN architecture, which might benefit from e.g. splitting the input coefficients into singular values and orthogonal modes and training the modes with orthogonal regularisation. One challenge with promising potential is to explore obtaining a greater number of outputs (modes) from the SNN than inputs, potentially achieving reconstructions that are more accurate than the rank-truncated POD itself.

Chapter 4

Non-Linear Modes of Flow over a Stalled Airfoil

The proper orthogonal decomposition has long been a popular tool to extract orthogonal modes from complex systems. Recently there has been a significant focus on finding a suitable method for extracting non-linear modes from flow fields in aerodynamics. The use of non-linear modes allows for the creation of a latent space which contains more information than the linear equivalent. In the current research the relation between the linear proper orthogonal decomposition and neural networks is shown. The use of hierarchical neural networks leads to the creation of non-linear modes which can be ranked according to their contribution to a reconstruction. In order to create orthogonal non-linear modes a variational autoencoder has been used for which the regularisation of the latent space distributions has been tuned such that the reconstruction outperforms the proper orthogonal decomposition, while the latent space distributions closely match a unit normal distribution. This allows to sample new latent space coefficients from known distributions which result in realistic reconstructions.

4.1 Introduction

The proper orthogonal decomposition (POD) has been introduced by Payne and Lumley (1967) in fluid dynamics for the study of patterns within turbulent flow. The POD allows to extract a set of linearly independent modes from a data set (Sirovich 1987a; Berkooz et al. 1993). Each mode represents a multidimensional direction in which the data set has a given variance. The POD modes are typically ordered from highest variance to lowest. The first POD mode represents the direction in the data set which spans the largest one dimensional variance. Many flow phenomena however are non-linear and could be better modelled by a non-linear representation of coherent flow structures. Autoencoders bare similarities to the POD, but are a type of architecture of non-linear neural networks. Autoencoders are a popular tool for the analysis of a wide range of aerodynamic parameters thanks to their ability to model both linear and non-linear systems efficiently as shown by Saetta et al. (2022) and Omata and Shirayama (2019). To improve the interpretability of the non-linear modes it is possible to use hierarchical networks which are trained successively as shown by Fukami et al. (2020) and Saegusa et al. (2004). As networks are trained successively, the ordering will be such that the first network represents the largest reduction in error for the reconstruction with the modes that it learns. Alternatively the use of variational autoencoders regularises the latent space by drawing the the latent space variable values from learned distributions as introduced by Kingma and Welling (2014). Variational autoencoders have been used for a range of applications in aerodynamics research from flow field predictions by Yang et al. (2022), to the design of airfoils by Yonekura and Suzuki (2021). Variational autoencoders allow to create continuous latent space variables where each combination of the latent space variables results in a realistic sample once decoded. This property is what makes this type of neural network a generative model which can create new realistic samples by sampling a latent space. The inclusion of an additional parameter, beta (β), can be used to adjust the amount of regularisation that is applied to the latent space. By tuning this coefficient it becomes possible to find a balance between the information captured in the latent space and the distribution of the latent space being close to a $\mathcal{N}(0,1)$ distribution. Eivazai et al. (2022) used β variational autoencoder networks to show the possibility of creating near orthogonal non-linear modes, which capture a larger energy content of the data set with a limited set of modes compared to the POD. In the present work different data driven reduced order models are compared to explore the advantages and disadvantages associated with each method. The different methods that are available are all closely related to each other with certain methods being better suited for specific purposes. A good understanding of the different available reduced order model methods allows to select and create the appropriate model for a data driven approach which can significantly impact the performance.

4.2 Separated flow data set

A data set compromised of processed PIV images of the flow over a wing is used as the training data. The full details about the experimental setup and the data processing of this data set are given in section 2.1.1. This data set includes the 2D velocity components in the wall normal plane above a NACA 0012 airfoil profile. For the current investigation only the fluctuations of the streamwise velocity component are used, as shown in fig. 2.2. The full data set has been split into a training and test set, the training set contains 65616 samples and the test set 5468 samples (12:1 ratio). A subset of the training set can be used as the validation set where necessary. The validation set is allowed to be evaluated during the training of several methods, whereas the test set does not have an influence on the training of any method and will only serve to evaluate the performance of a method after the training phase is completed.

4.3 Linear modes

Two advantages of linear modes are the ability to use linear superposition of modes to reconstruct the training data and the speed of such algorithms. This allows to separately interpret the modes, while the POD coefficient for each mode indicates its relative importance among the set of modes used. Similarly the singular values indicate the relative importance of the modes across a training data set. Thanks to the linearity of the modes there are a number of efficient algorithms which can be used to find a set of modes for a data set.

4.3.1 POD by singular value decomposition

Typically a large enough training set is used such that all representative occurrences are represented, which allows to utilise the learned spatial modes on new but similar samples. The Pearson correlation, between n samples of x and y, is typically given as:

$$r_{xy} = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2}}\,, \tag{4.1}$$

where x_i and y_i indicate individual samples, and $\bar{x}$ and $\bar{y}$ indicate the averages. The root mean squared error (RMSE), for n samples of x and y, is typically given as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (x_i - y_i)^2}\,. \tag{4.2}$$

In fig. 4.1 the Pearson correlation and RMSE are shown for truncated POD reconstructions in comparison with the original samples. The learned spatial modes and singular values from the training set have been used on the test set. It can be observed that the correlation quickly increases with the inclusion of more modes from one to ten modes, as more modes are included the gradient of the Pearson correlation improvement reduces. The error can be seen to reduce the most with the inclusion of the initial modes, ordered by their singular values. In fig. 4.2 the explained variance ratio is shown which represents the ratio between the variance captured by a limited set of spatial modes over the entire set of spatial modes. With the inclusion of 362 modes it is possible to reconstruct 99% of the variance in the data set.

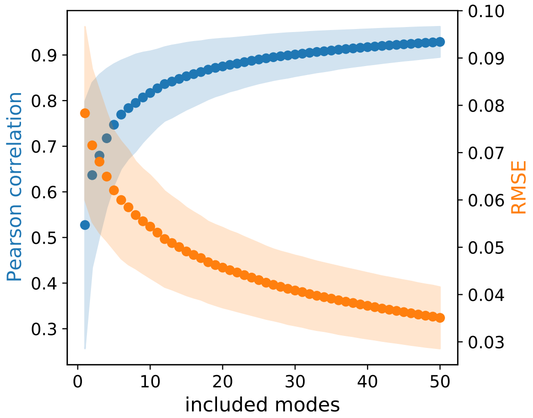


FIGURE 4.1: The Pearson correlation and the root mean squared error of the test data set for varying amounts of modes used in a truncated POD reconstruction.

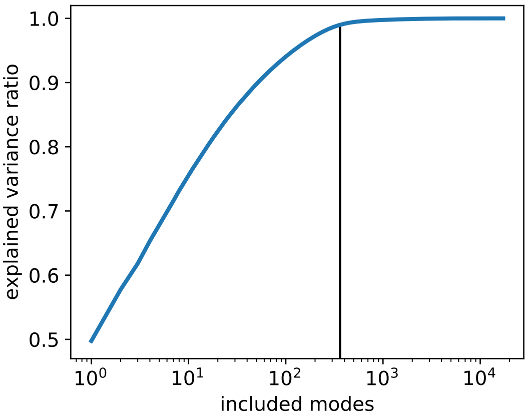


FIGURE 4.2: The cumulative explained variance ratio for an increasing amount of modes included, starting with the modes with the highest singular values. The black vertical line indicates the 99% variance threshold at 362 modes included.

In fig. 4.3 truncated reconstructions are shown of a single sample with varying amounts of spatial modes (and singular values) included. It can be seen that even up to 10 modes the flow patterns that are visible tend to be of a larger scale than represented in the original sample. When 25 modes are included in the truncated reconstruction some flow patterns are visible on similar scales to the original sample.

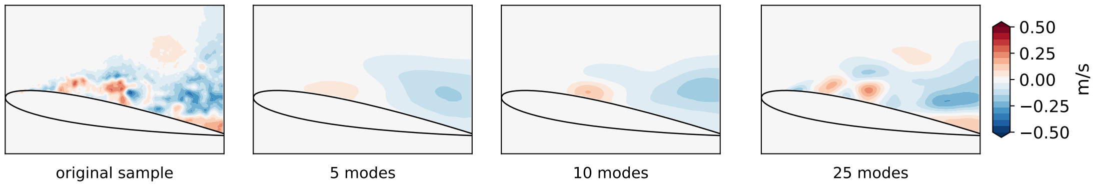


FIGURE 4.3: The reconstruction of a single sample is shown with a varying amount of POD modes included in the reconstruction.

In fig. 4.4 four of the utilised spatial modes are shown. The spatial modes with the largest singular values (lowest mode number) correspond to the modes with the largest flow structures.

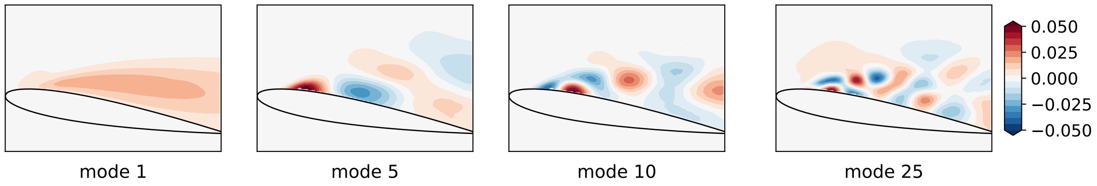


FIGURE 4.4: The spatial modes corresponding to the 1st, 5th, 10th and 25th largest singular values.

4.3.2 Hierarchical ML-POD

The use of machine learning in the form of neural networks provides unlimited variations for a desired function to be approximated through a training process. The architecture known as an autoencoder consists of several layers of nodes, where the input size matches the output size and a restricted layer is in between them. The restricted layer contains less nodes than there are variables in the original input data and resulting output data. This results in the restricted layer being forced to learn an efficient representation of the input data which allows mapping to the output with a minimal error. When only one linear layer is used between the input and output, which has less nodes than the input, the resulting network represents the same transformation as a truncated POD reconstruction as shown by Milano and Koumoutsakos (2002).

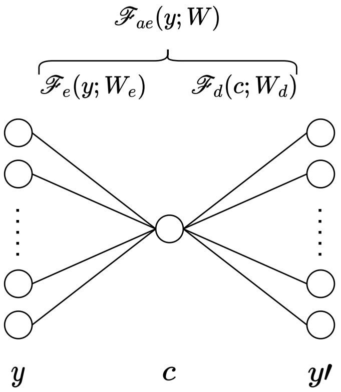


FIGURE 4.5: The general architecture of an encoder and decoder combined to form an autoencoder. Only a single node is used for the encoding of a single mode. The input y is transformed into a latent space coefficient c and this coefficient is used as an input to obtain the truncated reconstruction y‘

In order to illustrate this principle a neural network has been created with an architecture of three layers as shown in fig. 4.5. The first layer does not perform any mathematical operation and represents the inputs. The second layer consists of a single node, which represents the transformation by a single spatial mode and has an amount of trainable parameters equal to the amount of variables in the input data. The encoder is described as $\mathcal{F}_e(y;\boldsymbol{W_e}) = c$ in eq. (4.3) with input sample y, weights $\boldsymbol{W_e}$ and latent space coefficient c. The third and last layer of the network consists of an amount of nodes equal to the amount of variables in the input data, this layer is connected to the single node layer before it and produces the reconstruction of the single latent space coefficient to a full sample, the decoder is described as $\mathcal{F}_d(c,\boldsymbol{W_d}) = y'$ with weights $\boldsymbol{W_d}$ in eq. (4.3) and produces the truncated reconstruction y'. No activation function or bias is used for the second and third layer.

$$\mathcal{F}_{ae}(y;\boldsymbol{W}) = \mathcal{F}_d(\mathcal{F}_e(y;\boldsymbol{W_e});\boldsymbol{W_d}) \tag{4.3}$$

The additional requirement of the spatial modes having norm one can be explicitly prescribed in the final reconstruction layer, such as described by eq. (4.4), by scaling the weights to achieve norm one during training. The middle layer (encoder) with a single node then transforms the input into a single POD coefficient for each sample which incorporates the singular value. This allows to visualise the spatial modes obtained in the final layer of the network and to obtain the singular value found by the network as the norm of the output of the single node layer for all samples.

$$\left|\left|\boldsymbol{W}_d\right|\right|_2 = 1 \tag{4.4}$$

In combination with the original data set the optimisation of this neural network converges to the transformation that is obtained by the first spatial mode of the POD of the data set. The weights $\boldsymbol{W}_e$ and $\boldsymbol{W}_d$ are optimised by minimising the loss over multiple epochs (q) and retaining the weights which produce the lowest loss as described by eq. (4.5) where the tilde represents the state during the training phase. The loss for each epoch is the L2 norm of the difference between the original input and the output of the autoencoder compromised of the encoder and decoder, such as described by eq. (4.6) where n is the amount of samples in the data set at each epoch q.

$$\boldsymbol{W} = \underset{\widetilde{\boldsymbol{W}}}{\operatorname{argmin}} \, \mathcal{L}(\widetilde{\boldsymbol{W}}) \tag{4.5}$$

$$\mathcal{L}^{(q)}(\widetilde{\boldsymbol{W}}) = \sum_{j=1}^{n} \left|\left|\boldsymbol{y}_j - \widetilde{\mathcal{F}_{ae}}(\boldsymbol{y}_j; \widetilde{\boldsymbol{W}})\right|\right|_2 \tag{4.6}$$

In fig. 4.6 the first spatial modes as obtained with the neural network approach is shown in comparison with the spatial mode as obtained from the SVD. Only very small differences are present between the NN spatial mode or the SVD spatial mode. In fig. 4.7 the distribution is shown of the POD coefficients. These are the temporal modes from the SVD multiplied with the singular value of the first mode, and the latent space variables from the NN. It can be seen that the distributions match very well. The singular values as obtained from the SVD or the NN only differ by approximately 0.1%. This has been obtained with 10 iterations over the entire training data set in combination with the ADAM optimiser.

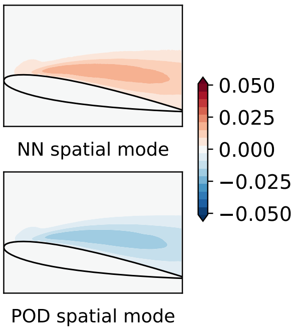


FIGURE 4.6: A comparison of the first spatial mode as found by a neural network (top) or calculated with the singular value decomposition.

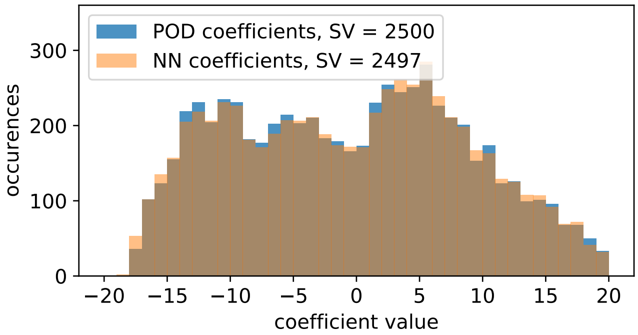


FIGURE 4.7: The distribution of the POD coefficient values and the latent variable value of the neural network for the training data set. As the spatial modes have an opposite sign, the NN coefficients have been multiplied with -1. The singular value for the first mode from the singular value decomposition is 2500, and the singular value obtained from the linear neural network is 2497.

The resulting single node autoencoder converges to the first POD mode, leaving the remaining POD modes as described by eq. (4.7) where c_k represents the POD coefficients for the k^{th} mode and ϕ_k the k^{th} spatial mode. The approximation (y') can be subtracted from the original input data (y), leaving the combination of the remaining modes. A new network with the same architecture can be used to find the second mode. In a similar fashion through recursive training the following modes can be obtained.

$$y = \mathcal{F}_{ae,\,mode\ 1} + \sum_{k=2}^{n} c_k \boldsymbol{\phi}_k \tag{4.7}$$

In fig. 4.8 the four first spatial modes are shown as obtained by the neural network approach and the SVD. The comparison shows the similarity between the spatial modes, it is possible that the modes have an opposite sign which is resolved by the corresponding coefficients in the decomposition and reconstruction.

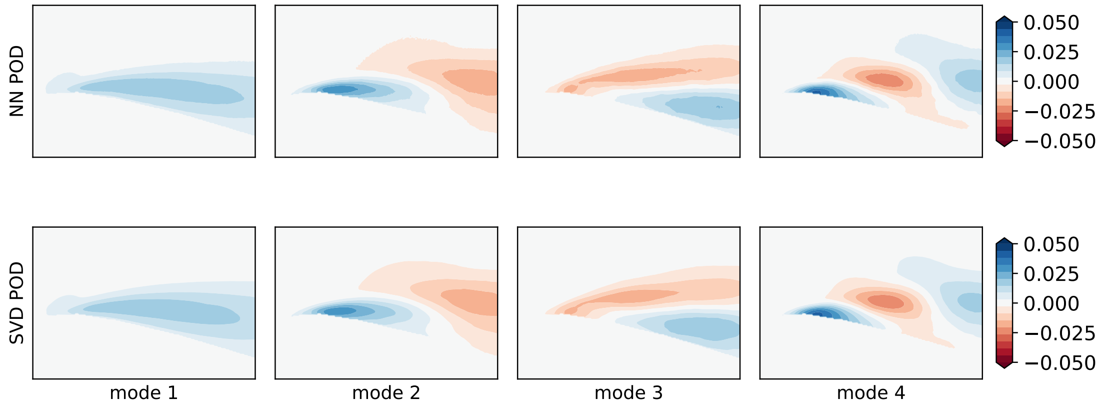


FIGURE 4.8: Comparison of the spatial modes obtained with a neural network versus the SVD for the first four modes.

4.4 Non-linear modes

Linear modes can be easily interpreted and provide a clear manner to create a truncated version of the data set. The main drawback of the linear modes is that they are not well suited to represent complex non-linear phenomena. Non-linear modes can be created which can include more information in a single mode than the linear variant. Neural networks can be used to reproduce the input data with a restricted number of nodes between the input and output. Such an autoencoder model is known to create an efficient representation of the training data. In this section some non-linear methods are explored for the use on a separated flow data set. By being able to capture more information with less modes than the linear equivalent the modes become more representative for the flow.

4.4.1 Hierarchical autoencoder

Autoencoders can exist in many variations due to the flexibility of a neural network architecture. The most restricted hidden layer of an autoencoder determines the amount of latent space variables that are used for a reconstruction. By creating a neural network with a given amount of nodes in the restricted hidden layer the accuracy of the reconstruction is also set. With multiple hidden layers and non-linear activation functions the interaction between different modes becomes more complicated to understand and cannot be further expanded to include more latent space variables after the training phase of the given neural network. A trail and error process is required to find the appropriate size of the latent space (and network architecture) for a desired level of accuracy in the reconstruction of a given data set.

The use of a regular autoencoder does not allow to create an encoded representation of the data where the latent space variables are ordered according to their importance. Saegusa et al. (2004) adapted a non-linear autoencoder approach to build successive non-linear modes ranked according to their contributions to the reconstruction of a data set. This approach relies on an autoencoder to find an efficient non-linear encoding of the data set into a single mode. For the first mode this is achieved with an autoencoder which has a single node in the middle layer and potentially multiple hidden layers. The encoder architecture is retained for finding the mapping of the input data to the latent space variable for the following modes. The structure of hierarchical autoencoders is described in more detail in Saegusa et al. (2004) and illustrated in fig. 4.9 for the first three modes. The decoder architecture is changed by increasing the number of inputs to allow for the increasing number of dimensions of the latent space vector. By successively training the encoders (and corresponding decoders) the modes are automatically ordered. This method effectively allows to train multiple networks with a limited set of available data by splitting the training into sub-networks and combining the trained networks (similarly to transfer learning). In the current investigation monotonically decreasing loss was observed during training, which allowed to use a small number of epochs (5) without a reduction in the loss larger than 10^{-5} to serve as the stopping criterion. Once the training is stopped, weights which resulted in the lowest loss during training are restored. Additionally, thanks to the transfer learning of different modes, the network which is trained for successive modes is very similar to the networks for previous modes. This allows to use the same stopping criterion for different modes.

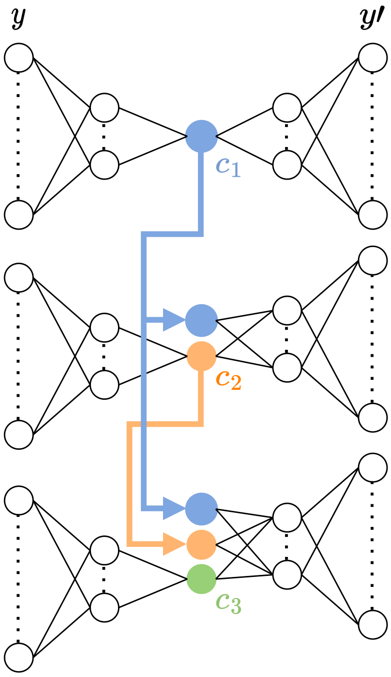


FIGURE 4.9: Hierarchical autoencoder setup. With the input y the reconstruction y' and the latent space variables c_n where n represents the n^{th} latent space variable corresponding to the n^{th} mode.

The decoder maps the latent space vector non-linearly to the reconstructed output. For the first mode it is possible to visualise the non-linear variation in the output for a range of latent space input values to the decoder. In fig. 4.10 the reconstruction is shown for four values of the latent space variable for the first non-linear mode that have been drawn from the distribution that follows from the training data set. It can be seen that the single non-linear mode obtained with the autoencoder has the ability to represent significant variation in the resulting flow patterns, similar to multiple POD modes or the combination thereof. As no additional scaling factor equivalent to the singular values in POD is used, the output directly represents the reconstruction in the velocity domain.

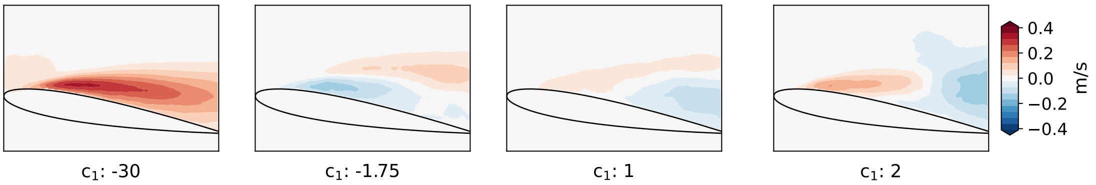


FIGURE 4.10: The reconstruction of four samples out of four values for the the latent space variable of the first mode. All other latent space variables are set equal to zero.

When using two modes with the non-linear hierarchical autoencoders the modes can not be displayed in a similar fashion as for a single mode. The resulting reconstruction is dependent on the non-linear combination of two latent space variables. The two dimensional distribution of the values for the latent space variables can be visualised as shown in fig. 4.11. It can be seen that the full range for each mode individually is not applicable over the full range of the other modes. Specific combinations of latent space variables values can lead to unrealistic reconstructions.

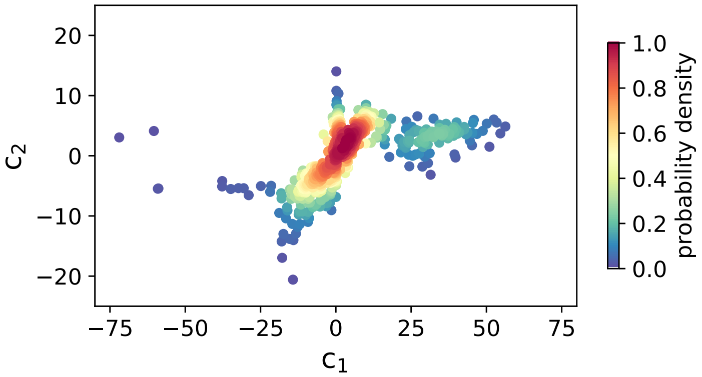


FIGURE 4.11: Two dimensional latent space vector representation as found by the first two networks of the hierarchical autoencoder. Only one out of every hundred samples is shown for clarity, the dots are coloured according to the normalised probability density.

As was shown in fig. 4.10 the hierarchical autoencoder has multiple flow patterns that can be created from a single mode. It can thus be reasoned that a single non-linear mode will perform better than a single linear mode in the reconstruction. The root mean squared error (RMSE) and the Pearson correlation have been evaluated for a reconstruction with both linear POD modes and non-linear hierarchical autoencoder modes, with up to 4 modes included in the reconstruction, as shown in fig. 4.12 and fig. 4.13. The error with a non-linear reconstruction is found to be lower when an equal amount of modes is included. Additionally the Pearson correlation for the non-linear reconstruction is significantly higher than the equivalent linear reconstruction.

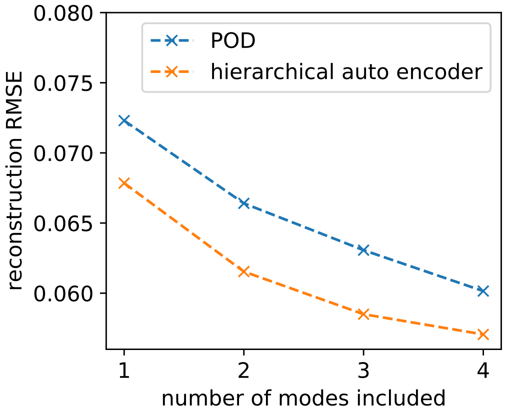


FIGURE 4.12: The root mean squared error for the truncated reconstruction with POD and a hierarchical autoencoder.

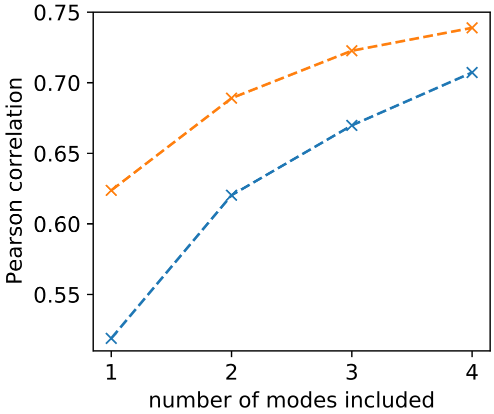


FIGURE 4.13: The Pearson correlation for the truncated reconstruction with POD and a hierarchical autoencoder.

A drawback of the non-linear approach is the complexity in the setup and the computational cost, these aspects limit the practical application of a large amount of modes. Additionally the POD is guaranteed to converge to an error of zero and a Pearson correlation of one with the inclusion of more modes. The use of non-linear modes can be beneficial when the linear modes are unrepresentative or when it is specifically desired to have a low dimensional latent space.

4.4.2 Variational autoencoder

The use of neural network based autoencoders is great to encode a given data set. The general applicability of the resulting autoencoder largely depends on whether the training data set is representative for the evaluation data or not. Additionally no samples can be trivially reconstructed from a chosen latent space vector without detailed knowledge of the multidimensional latent space distribution. These problems are tackled by using a neural network approach which aims at developing a generative model. Such a model can be created with probabilistic constraints to encourage the latent space to have a known distribution.

To this end a variational autoencoder (VAE) can be used, as originally described by Kingma and Welling (2014). The encoder of an n-dimensional VAE produces two output vectors of dimension n. The elements of the first vector represent a mean and the elements of the second vector represent a standard deviation. The n latent space variables are then sampled from the normal distributions described by the vectors. A "reparametrisation trick" allows to formulate the sampling of the distributions to be differentiable. The latent space values are then decoded by the decoder. As the encoder learns distributions for different latent space variables, these can be enforced to be similar by using the Kullback-Leibler (KL) divergence. The loss function incorporates both the reconstruction loss and the KL divergence. A beta variational autoencoder (β - VAE) uses a parameter β, to balance the contribution of the reconstruction loss and the KL divergence. In this report only the implementation of this method is shortly described. More complete analytical derivations, explanations and proof are given in Blei et al. (2017) for variational inference and in Doersch (2016) for VAEs in addition to Kingma and Welling (2014).

The encoder in this case produces two deterministic outputs, the mean μ_y and the standard deviation σ_y dependent on the input y. An additional random variable, ϵ, is introduced which is $\mathcal{N}(0, 1)$ distributed. The reparametrisation trick is used to sample a latent space, z, by incorporating the distributions produced by the encoder as shown in eq. (4.8).

$$z = \mu_y + \epsilon \cdot \sigma_y \tag{4.8}$$

The decoder uses the latent space z as the input to produce the reconstruction y'. The distribution divergence shown in eq. (4.9) is used as part of the loss which regularises the latent space. The mean squared error (MSE) reconstruction loss shown in eq. (4.10) is used to ensure the reconstruction is accurate.

$$\text{divergence loss} = \mathrm{KL}\left[\mathcal{N}\left(\mu_y, \sigma_y^2\right) \ \ || \ \ \mathcal{N}(0,1)\right] \tag{4.9}$$

$$\text{reconstruction loss} = \mathrm{MSE}(y, y') \tag{4.10}$$

The total loss as shown in eq. (4.11) is the sum of both the regularisation and reconstruction loss with a balancing parameter (β) applied.

$$\text{total loss} = \frac{1}{n}\sum_{j=1}^{n}(y_j - y_j')^2 - \frac{\beta}{2 \cdot n}\sum_{j=1}^{n}(1 + \log(\sigma_j^2) - \mu_j^2 - \sigma_j^2) \tag{4.11}$$

The β parameter serves to prioritise either the reconstruction accuracy or the latent space distribution. The KL divergence tries to match the conditional probability distribution approximation q($z|y$) to p(z), where p(z) is prescribed as $\mathcal{N}(0,1)$. As the latent space distribution is forced towards $\mathcal{N}(0,1)$ a closely matching latent space distribution will force the q($z|y$) $\approx$ p(z). This would imply that the latent space z is independent of the input y. Without the reconstruction loss term the encoder would simply sample $\mathcal{N}(0,1)$ with random y' obtained from the decoder. Without the regularisation term the variational autoencoder would not have closely matching distributions for the latent space variables. It is clear that a delicate balance between both terms is relevant and this is the purpose of the β parameter. The balance between minimising the reconstruction error and obtaining a latent space from which can easily be sampled is typically explored through trial and error. Typically the reconstruction error is observed, to not compromise the essential function of the autoencoder. At the same time the generalisation of the VAE can be monitored through the latent space distribution, or the orthogonality of the VAE modes through the determinant of the latent space correlation matrix. Regularisation of the latent space comes at the cost of reducing the reconstruction accuracy, while improving the generalisation of the decoder.

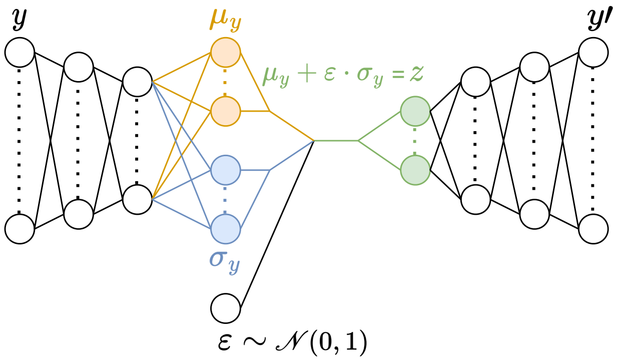

y
μ_y
$y\prime$
$\mu_y + \varepsilon \cdot \sigma_y = z$
σ_y
$\varepsilon \sim \mathcal{N}(0,1)$

FIGURE 4.14: The architecture of a VAE network.

In fig. 4.14 the detailed architecture is shown for a VAE with two shared hidden layers in the encoder, followed by the encoding layers for the mean (yellow) and standard deviation (blue). The sampling of the latent space z (green) includes the additional unit normally distributed noise parameter ϵ. The decoder has two hidden layers to decode the latent space into the reconstruction y'. The number of latent space variables is determined by the number of nodes that are used in the layers which output μ_y and σ_y. The latent space z is then sampled and decoded by the decoder with two hidden layers. In the current investigation the hidden layers of the encoder have 30 and 25 nodes, followed by 4 nodes for μ_y and also for σ_y. This encoder thus allows for the sampling of a latent space with dimension four. This latent space is then decoded by a decoder that has two successive hidden layers with 25 nodes for the first and 30 for the second layer. The final layer of the decoder has an equal amount of nodes as the amount of variables in the input. The two hidden layers in the encoder and two hidden layers in the decoder use a ReLU activation function, while all other layers use a linear activation function.

Initially the β parameter has been set to zero. This implies that only the reconstruction is included in the loss function. However unlike a regular autoencoder each variable of the latent space is determined through the combination of three parameters. Two parameters, the mean and standard deviation are calculated by the encoder, while the third parameter is random noise. The encoder can minimise the reconstruction error by minimising the impact of the noise through reducing the calculated standard deviation, this is shown in fig. 4.15. As such the calculated mean (μ_y) effectively becomes the latent space variable z.

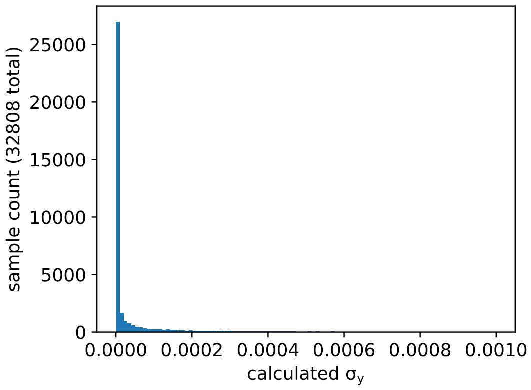


FIGURE 4.15: The calculated standard deviation by the encoder for four latent space variables for $\beta = 0$.

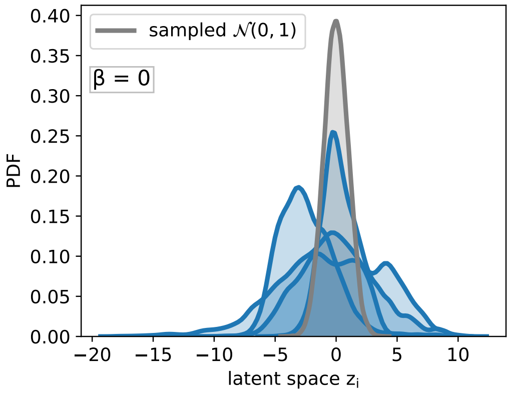


FIGURE 4.16: The distribution of four latent space variables z_i with $\beta = 0$.

In fig. 4.16 the distribution of the latent space variables is shown for each dimension of the latent space individually. These distributions are not necessarily centred around zero and do not have a consistent standard deviation close to one as the KL divergence has no impact on the loss. In fig. 4.17 the distribution is shown of the latent space values for the first and second variables. The designation of the first and second variables is arbitrary when the variables have not been ordered. From the figure it can be seen that while the distributions have significant overlap, the encoding of the known samples still follows certain directions in the latent space. Sampling from the latent space to result in new decoded samples would still require knowledge of these vectors in the latent space that provide the main variation in the distribution. This could be achieved through calculating the eigenvectors of the latent space.

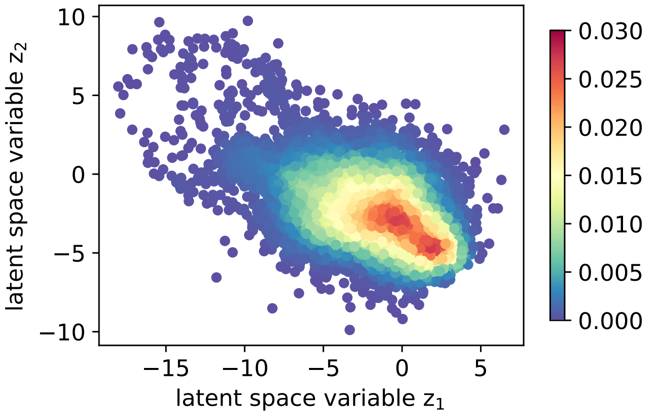


FIGURE 4.17: The distribution of the first and second latent space variables combined.

For POD we know that the modes are orthogonal and the temporal modes over the entire training data set are also orthogonal. We can illustrate the independence of the different modes with a correlation matrix of the latent space coefficients for the POD modes, as shown in fig. 4.18. It can be seen that the correlation between different modes is zero. Similarly for the non-linear modes the latent space can easily be obtained from the training set samples. The resulting correlation matrix is shown in fig. 4.19. The determinant of the correlation matrices can be interpreted as a metric of the independence of the modes, with 1 indicating that the modes are fully independent and 0 indicating that the modes are completely dependent. While the POD modes are completely independent, the modes found with the β-VAE and $\beta = 0$ are still dependent to a large extend with a determinant of 0.26 for the correlation matrix.

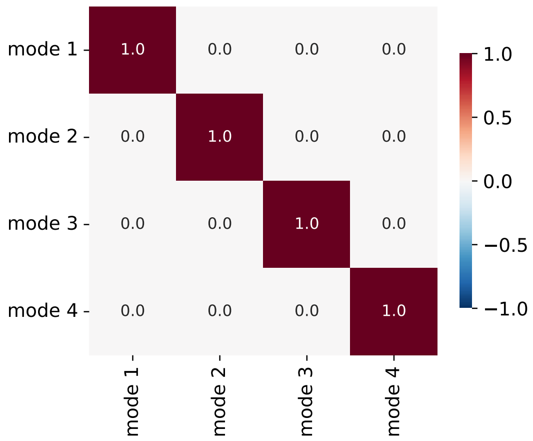


FIGURE 4.18: The correlation matrix of the training set latent space coefficients obtained with POD. The determinant is 1 when evaluated on the training data.

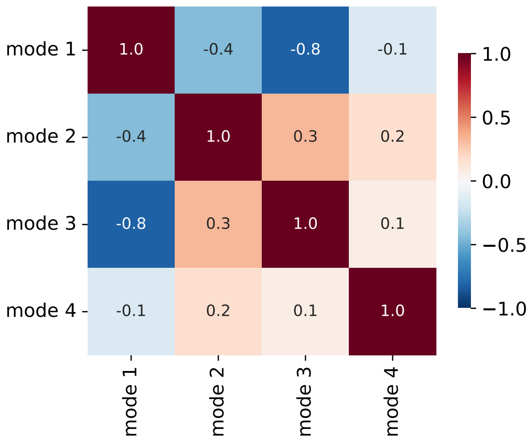


FIGURE 4.19: The correlation matrix of the training set latent space coefficients obtained with a β-VAE and $\beta = 0$. The determinant is 0.26 when evaluated on the training data.

By having the β parameter equal to zero, full priority is given to minimising the reconstruction error of the β-VAE. This results in a low reconstruction error where the orthogonality or sparsity of the modes is not prioritised. The model has also been evaluated on a test data set during training to monitor the progress. In fig. 4.20 the model metrics are shown for the training phase of the model over a 100 epochs. From the figure it can be seen that very early in the training process the reconstruction error and the Pearson correlation outperform the POD with 4 modes. Additionally it is shown that the determinant of the correlation matrix gradually increases and approximately stagnates after 40 training epochs. The different determinant value as opposed to fig. 4.19 is due to the evaluation on the test set which has no influence on the training, resulting in a lower determinant for the test data set.

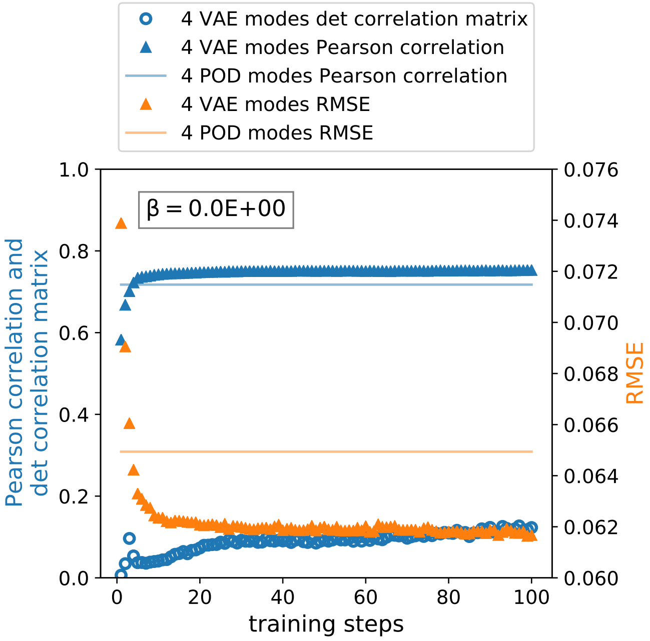


FIGURE 4.20: Training a β-VAE with $\beta = 0$. The RMSE, Pearson correlation and determinant of the correlation matrix have been evaluated on the test data set.

Lastly it is possible to illustrate the modes as decoded from latent space samples. The non-linearity of the modes and the combination thereof can not be easily visualised. It has been chosen to visualise each latent space dimension for two coefficient values while keeping all other latent space variables at zero. Each variable has been visualised at a value -2 and 2, these values fall within the distribution as obtained from the training set. The result is shown in fig. 4.21. It can be seen that due to the non-linearity the shape of the flow patterns is not consistent over the range of values that a single latent space variable can take.

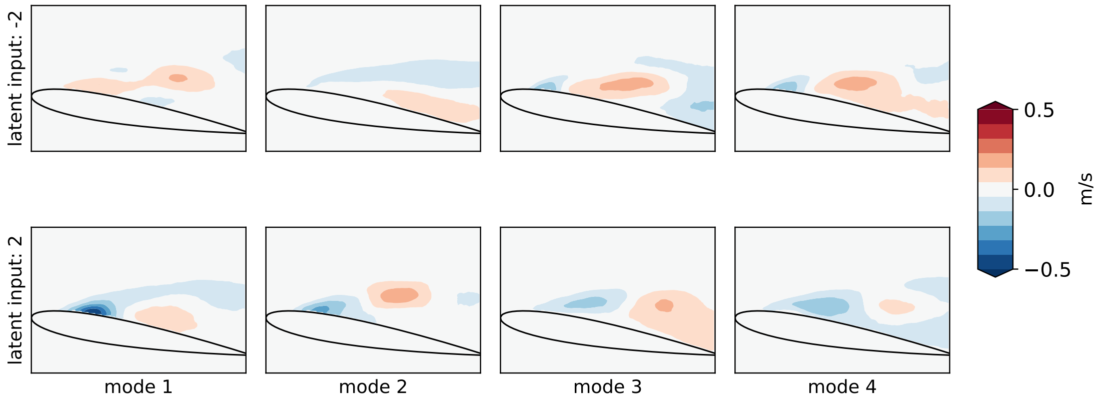


FIGURE 4.21: Visualisation of the activation of individual latent space variables with all other latent space variables set equal to zero.

It is possible to tune the β parameter to force the latent space distributions to be closer to a $\mathcal{N}(0, 1)$. By doing so the network is encouraged to keep the mean activation near zero, thereby creating sparse modes. The β parameter can be tuned to create disentangled modes which offer better interpretability for their range of activation (Higgins et al. (2017)). Each mode individually is representative for flow patterns that can occur, unlike with POD where modes with lower singular values are less likely to fully represent a realistic flow pattern. The determinant of the correlation matrix describes the independence of the different latent space variables but does not guarantee disentanglement. In the current investigation we seek to find a balance between the reconstruction offering an improvement over the POD with 4 modes, while also obtaining a high independence of the modes represented by the latent space variables. The β parameter can be increased to such an extend that the latent space becomes a random sampled $\mathcal{N}(1, 0)$ distribution which contains no information of the inputs. When the β parameter is too high, the reconstruction error will be higher and the Pearson correlation is lower than for a POD reconstruction. In fig. 4.22 the evaluation of these metrics is shown for a network with a β parameter of $2 \cdot 10^{-3}$ which results in a determinant of the correlation matrix of approximately one, indicating independent modes. At the same time the reconstruction appears to be worse than a POD reconstruction based on the RMSE and Pearson correlation.

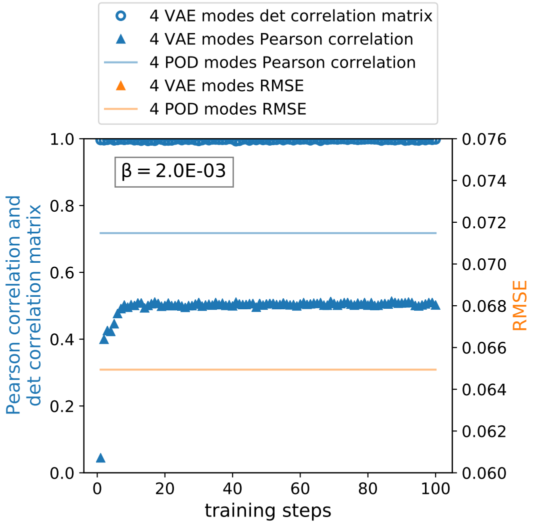


FIGURE 4.22: The RMSE, Pearson correlation and determinant of the correlation matrix during training for a β-VAE with $\beta = 2 \cdot 10^{-3}$.

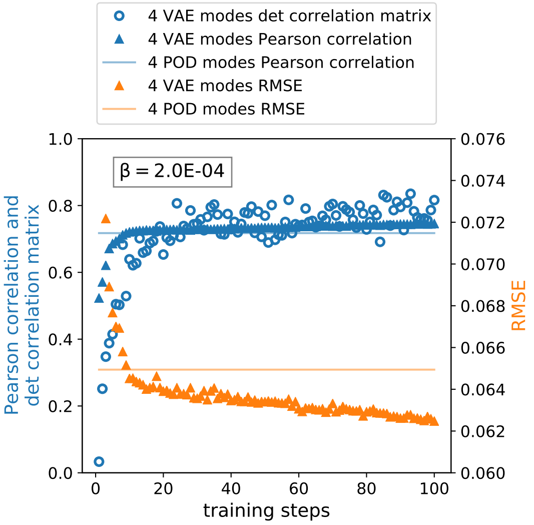


FIGURE 4.23: The RMSE, Pearson correlation and determinant of the correlation matrix during training for a β-VAE with $\beta = 2 \cdot 10^{-4}$.

In fig. 4.23 the evaluation of the reconstruction metrics and the determinant of the correlation matrix is shown for a β-VAE network with a β of $2 \cdot 10^{-4}$. The network can be seen to outperform the POD reconstruction by a small margin and also have a high determinant. This network balances the need to have an accurate reconstruction while enforcing the latent space to be $\mathcal{N}(1, 0)$ distributed.

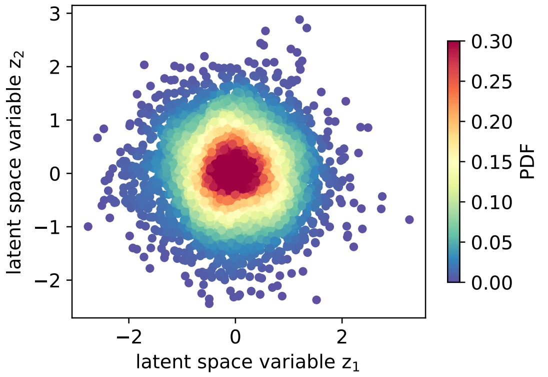
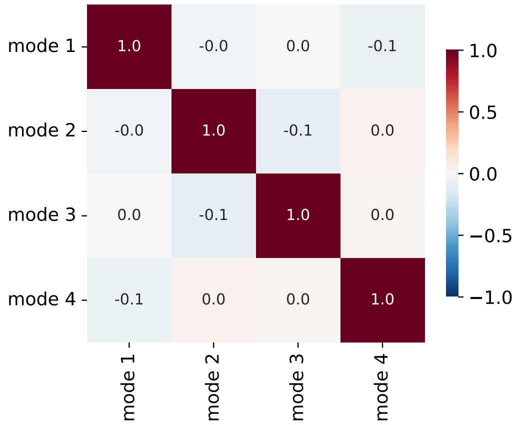


FIGURE 4.24: The distribution of the first and second latent space variables combined for a β-VAE with $\beta = 2 \cdot 10^{-4}$.



	mode 1	mode 2	mode 3	mode 4
mode 1	1.0	-0.0	0.0	-0.1
mode 2	-0.0	1.0	-0.1	0.0
mode 3	0.0	-0.1	1.0	0.0
mode 4	-0.1	0.0	0.0	1.0

FIGURE 4.25: The correlation matrix of the training set latent space coefficients obtained with a β-VAE and $\beta = 2 \cdot 10^{-4}$. The determinant is 0.98 when evaluated on the training data.

In fig. 4.24 the latent space distribution is shown of two modes of the β-VAE with $\beta = 2 \cdot 10^{-4}$. It can be seen that the latent space variable values are clustered around 0. The PDF is approximately maximal around 0 for each variable and decreases with the multidimensional distance from the origin. The generative ability of this network is better as the latent space distribution is better understood. The correlation of the latent space variables is shown in fig. 4.25. It can be seen that the latent space variables are almost completely independent. At the same time the reconstruction error is lower than POD and the Pearson correlation is higher. The modes can be considered disentangled as for this network it is likely that the reconstruction of independent latent space activations lead to realistic flow patterns.

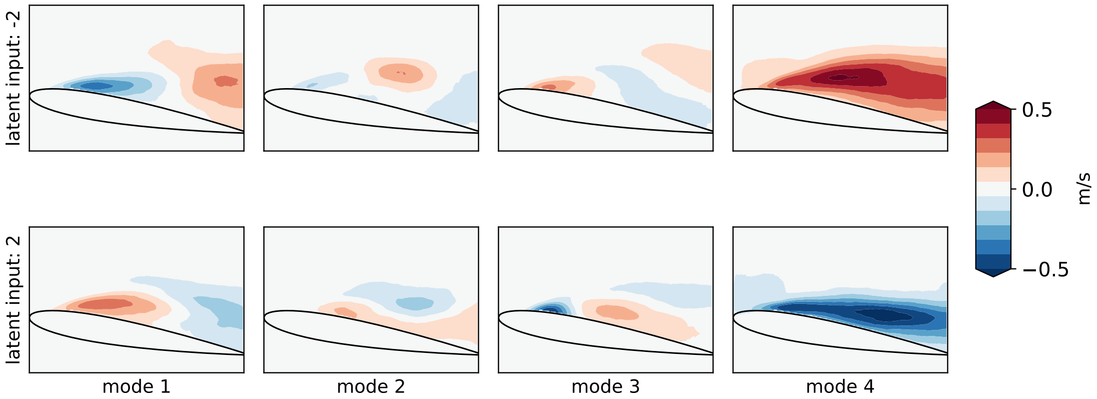


FIGURE 4.26: Visualisation of the activation of individual latent space variables with all other latent space variables set equal to zero for a β-VAE with $\beta = 2 \cdot 10^{-4}$.

Lastly it can already be observed from fig. 4.26 that the fourth VAE mode resembles the first POD mode. The Pearson correlation between the VAE activations and the POD modes can be calculated for the activation equal to -2, leading to a correlation matrix as shown in fig. 4.27. The correlation can also be visually compared with fig. 4.26 and fig. 4.4. In fig. 4.27 it is clearly indicated that the first POD modes matches the flow pattern produced by the -2 activation of the fourth VAE latent space variable. In fig. 4.28 similarly shows how the activation of 2 instead of -2 results in approximately opposite correlations. The β-VAE modes thus still resemble the POD modes while improving the generative ability and the reconstruction (or compression) with a limited latent space.

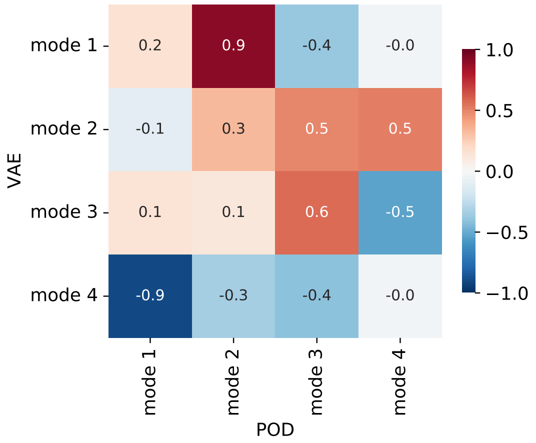


FIGURE 4.27: The Pearson correlation between the -2 activation of the VAE latent space and the POD spatial modes.

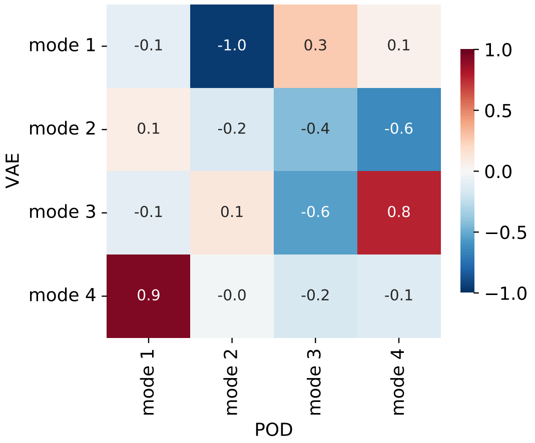


FIGURE 4.28: The Pearson correlation between the 2 activation of the VAE latent space and the POD spatial modes.

In order to further disentangle and order different modes obtained with a β-VAE it is possible to use a hierarchical approach. Training different subnetworks can be done recursively to obtain modes which address the part of the data set which has not been captured by the previously trained modes. As such the information of the data set contained in each mode will gradually decrease for additional modes.

4.5 Conclusions

A data set of flow fields over a stalled NACA 0012 wing has been used for the investigation of data driven mode extraction. The POD has been used as a benchmark which consists of a basis of linearly independent spatial modes. The root mean squared error and the Pearson correlation have been used to quantify the performance of the reconstruction with a given set of modes. The POD modes can be calculated efficiently on a large data set and provide a single set of modes for a given data set. Due to the linear nature of POD a large set of modes is required to represent most of the variance in the data set. An autoencoder neural network with linear activation function and a single hidden layer is equivalent to POD when trained to minimise the L2 norm. This has been shown through the process of training subnetworks successively which is often described as a hierarchical approach. A similar hierarchical approach can be used for non-linear autoencoders. The encoder can be trained to determine a single latent space variable. The decoder is trained to take into account the previously trained encoders and the current encoder which is being trained. As such the decoders cannot reconstruct individual modes, but the encoders can be called upon individually. A reconstruction from latent space variables requires specific knowledge about the multidimensional distribution of the latent space variables. From the reconstruction of samples in the test data set it has been shown that the use of non-linear modes does reduce the reconstruction error and improves the Pearson correlation. In order to benefit from the non-linear modes while allowing for sampling from the latent space the use of β variational autoencoders has been explored. By increasing the β parameter the distribution of the latent space variables are forced closer to a $\mathcal{N}(1,0)$ distribution, which results in a higher reconstruction error and lower correlation. Tuning the β parameter can be done such that the latent space distributions are close to $\mathcal{N}(1,0)$ while still producing a better reconstruction compared to POD. Such a variational autoencoder has a latent space which can be sampled from to generate new realistic samples. Additionally the modes of this variational autoencoder have been shown to be nearly orthogonal with a determinant of 0.98 or the correlation matrix. The β variational autoencoder which has been illustrated requires a separate evaluation method to order the modes that have been learned. An alternative approach would be to use hierarchical β variational autoencoder subnetworks to obtain a ranking of the modes according to the combination of the reconstruction error and the latent space regularisation. Further research is required to improve the interpretability of non-linear modes to complement their superior performance in comparison to POD.

Chapter 5

Surface Flow Classification

The behaviour of a boundary layer near the surface of an object is affected by many factors such as the viscosity of the air, the velocity and the pressure. The boundary layer behaviour is important because it affects the lift and drag forces on the object. An adverse pressure gradient on the object surface can cause the boundary layer to exhibit reverse flow. Separated flow on a wing surface is detrimental to the performance of the wing and can be highly unsteady and unpredictable. The high level of turbulence involved in separated flow makes it typically hard to investigate. In this investigation the near surface flow of a stalled NACA 0012 is investigated. This investigation culminates in a data driven model which predicts the surface flow type based on surface pressure.

5.1 Introduction

Separated flow over wings is an important topic of investigation as it is associated with a loss of lift and increase in drag. Additionally separated flow can result in the loss of control for aircraft as already indicated by Jacobs et al. (1933). McCullough and Gault (1951) presented three distinct types of stall based on how the wing evolves from attached flow to fully separated flow. The process of flow separation typically starts from local areas of boundary layer separation. Boundary layer separation has been a specific topic of interest for decades as shown by the overview given by Simpson (1996). The visualisation of surface flow with techniques such as surface oil flow visualisation (SOFV) or tufts relies on the flow direction close to the wing surface. More recently investigations of stall cell flow patterns have used surface flow visualisation as illustrated by Dell'Orso et al. (2016) and De Voogt and Ganapathisubramani (2022). However it remains difficult to combine the surface flow visualisation with obtaining quantitative data of the flow within a stall cell at the same time. Stall cells occur after stall as indicated by Yon and Katz (1998). They also indicated that the region within a stall cell experiences an upwash and the region outside the stall cell a downwash. The difference in effective angle of attack along the span is associated with the localised flow separation and variation in surface pressure. Different investigations such as Manni et al. (2016) have focused on how to computationally simulate stall cells. URANS simulations have been used to obtain a large data set of flow over a NACA 0012 wing with stall cells present. By using multiple angles of attack (α) and Reynolds numbers (Re) the results can be compared to assess the evolution of the surface flow in a wide operational range. This allows to understand how the surface flow is related to the surface pressure and the flow within the stall cell. Such an understanding of the flow supports the development of data driven stall detection algorithms such as that presented by Zhou et al. (2021). In the current investigation a fundamental approach for constructing such a data driven detection algorithm is presented based on pressure measurements on the wing surface.

5.2 Data and experimental setup

For the investigation of the surface flow over a NACA 0012 wing the computational data as described in section 2.1.2.

In addition to the computational data, experimental surface pressure measurements were taken on a NACA 0012 wing with aspect ratio (AR) 5.2. Two wing fences were used, which can be positioned in a desired location on the wing. The distance between the fences has been kept fixed at an AR of 2.5, allowing for the creation of a full stall cell between the fences. In some cases it was observed that two half stall cells formed rather than one whole stall cell. The experiments have been conducted in a wind-tunnel at the University of Southampton, with a test section of 2.1 m by 1.6 m and 0.02 m/s accuracy for the set velocity. The free stream turbulence intensity has been measured to be approximately between 0.5 to 1% at the location of the wing leading edge. A small piece of approximately 4 cm (0.13 c) zig-zag tape was used near the leading edge to trigger a stall cell behind it. In combination with the positioning of the fences it was possible to shift a stall cell in spanwise direction over the pressure taps. The wind-tunnel was run at a Reynolds number of $4.1 \cdot 10^5$ with the wing at an angle of attack of 14° to consistently produce a stall cell. A total of 24 pressure taps on the circumference of the wing were used in combination with a ScaniValve MPS4000 pressure transducer to record the surface pressure. The pressure was recorded at 500 Hz for a duration of 5 minutes for each run with different wing fence positioning. For each wing fence position the tufts were also recorded with a camera, to create a visual image for assessing the positioning of the stall cells relative to the pressure taps.

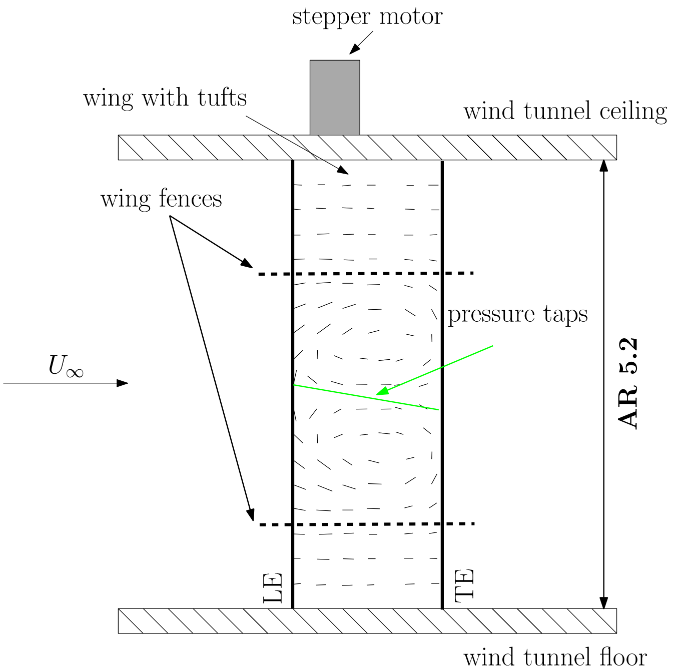


FIGURE 5.1: The NACA 0012 wing setup with tufts to indicate the presence of stall cells. A row of pressure taps (indicated in green) is in a fixed position. Movable wing fences are indicated by the dashed lines.

5.3 Flow field characterisation

In the current investigation the NACA 0012 wing has been computationally investigated with a focus on angle of attack and Reynolds number ($\alpha - \mathrm{Re}$) combinations that result in suction surface flow which includes flow separation. In fig. 5.2 different types of surface flow are indicated in lift polars for different Reynolds numbers. These surface flow types have been manually indicated by visual inspection. Different surface flow types may occur in different gradations, or even in combination with other surface flow types. This complicates the ability to create a definitive method to identify the surface flow type objectively. It has been chosen to assign a single surface flow type label to each $\alpha - \mathrm{Re}$ combination. The intermittency, size and other aspects of the surface flow patterns have been omitted for the manual classification.

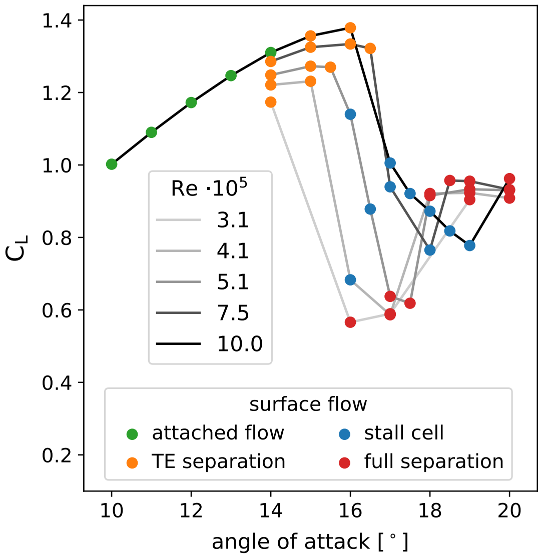


FIGURE 5.2: The time averaged lift coefficient for all the simulated cases. Different types of observed surface flow patterns indicated with markers.

The separated flow can be identified through upstream flow near the suction surface of the wing. The different types of surface flow indicated in fig. 5.2 are also illustrated by streamline plots of the flow near the wing surface in fig. 5.3, for which an offset of 0.1 % of the chord length perpendicular to the wing surface has been used. Attached flow shows no upstream flow, as shown by fig. 5.3a. The first occurrence of upstream flow when increasing the angle of attack at a constant Reynolds number occurs near the TE, as shown by fig. 5.3b. Further increases in angle of attack can in some cases result in the formation of stall cells, which include upstream flow near the wing surface and vortices on the wing surface, as shown in fig. 5.3c. When the angle of attack is increased even more, the stall cells can disappear and upstream flow occurs on almost the entire suction surface of the wing as shown in fig. 5.3d. In fig. 5.2 it can be seen that different flow types can occur for a single angle of attack at different Reynolds numbers. However the progression of flow types at a constant Reynolds number with increasing angle of attack moves consistently from attached flow to TE separation, to stall cells (if present), to finally full separation of the flow on the wing suction surface. No stall cells were observed at the lowest Reynolds number ($3.1 \cdot 10^5$) that was simulated. It is unknown whether stall cells do not occur at this Reynolds number, or that the angle of attack for the formation of stall cells at this Reynolds number was not tested. From fig. 5.2 it can also be observed that the occurrence of stall cells is limited to the cases for where the gradient $\frac{dC_L}{d\alpha}$ is negative.

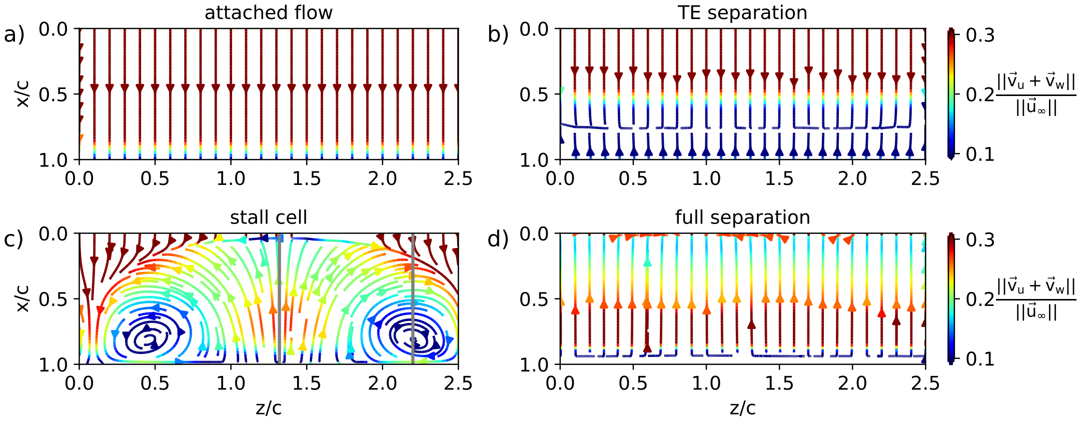


FIGURE 5.3: Flow velocity at a 0.001 c offset from the suction surface. Different types of observed surface flow patterns for the simulated cases with a Reynolds number of 10^6 at different angles of attack: a) 10°, b) 16°, c) 18° and d) 20°. The magnitude of the flow velocity is indicated relative to the freestream velocity. The vertical grey lines in c) indicate the centre of the stall cell at 1.32 z/c and the spanwise location of the vortex on the wing surface at 2.20 z/c.

The progression of surface flow patterns near the wing surface due to increasing angle of attack, as shown in fig. 5.3 is typical for wings which exhibit TE stall, and has also been observed experimentally with tufts by De Voogt and Ganapathisubramani (2022). From the CFD results it is possible to observe the flow near the surface of the wing at different chordwise locations, this allows investigating the progression of the flow as it passes over the wing surface. The surface flow patterns illustrated in fig. 5.3 have been obtained from specific $\alpha-\text{Re}$ combinations at a specific time step. The same cases have also been used to illustrate the flow above the surface with velocity profiles illustrating the local flow parallel to the wing surface as shown in fig. 5.4. At the surface of the wing the flow velocity is zero due to the viscosity of the air. When moving further away from the wing surface in a direction perpendicular to the wing surface, the momentum of the freestream velocity becomes more dominant. The layer of air between the surface of the wing and the inviscid flow away from the surface is the boundary layer. In fig. 5.4a the velocity profiles are shown for attached flow. In these velocity profiles the boundary layer can be identified as the region normal to the wing surface during which the gradient of the velocity for the wall normal distance is larger than zero. As the distance from the surface increases and the flow momentum dominates, the flow regime becomes inviscid. The thickness of the boundary layer can be estimated for attached flow in fig. 5.4a to be approximately 0.0025 c at the measurement closest to the leading edge (LE) and 0.05 c at the measurement location closest to the trailing edge (TE).

The flow near the wing surface slows down as the distance travelled along the surface increases. Both the viscosity of the air and the adverse pressure gradient contribute to slowing down the flow velocity in the boundary layer. These two effects combined can cause upstream flow to occur on the wing surface as illustrated by the TE separation in fig. 5.3b and fig. 5.4b. In comparison to the attached flow it can be seen that (at a higher angle of attack) the boundary layer thickness grows faster and bigger towards the TE. At some point along the airfoil surface the gradient of the velocity at the wing surface becomes zero and moving further towards the TE causes the gradient to become negative, indicating that the flow is moving upstream near the wing surface.

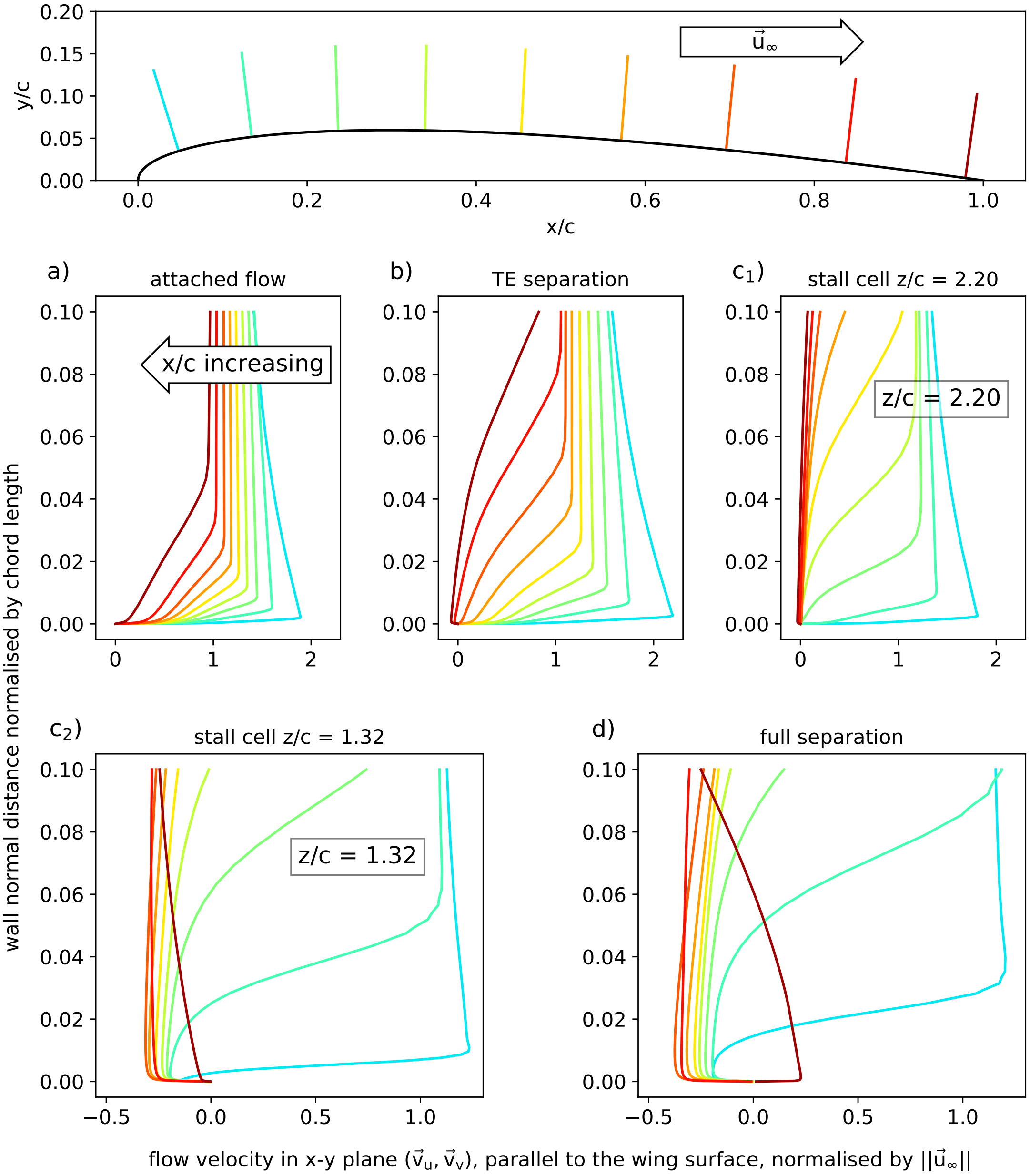


FIGURE 5.4: Velocity profiles near the wing suction surface at different chordwise positions for varying surface flow patterns. The line colors indicates the location of the measurement. The measurements are perpendicular to the wing suction surface and range between 0 to 0.1 c away from the wing surface. All velocity profile plots show results for a Reynolds number of 10^6, and varying angles of attack: a) 10°, b) 16°, c$_1$) 18°, c$_2$) 18° and d) 20°. The spanwise location is z/c = 1.25 (midspan), except for the plots c$_1$) and c$_2$) which consider a stall cell with spanwise variation in the flow.

Further increases in angle of attack increase the adverse pressure gradient on the wing surface and move the separation point upstream. It is possible that the flow near the surface suddenly transitions into a stall cell configuration with upstream flow in the centre of the stall cell and two spiral nodes near the sides of the stall cell. The vertical grey line in fig. 5.3c which crosses the stall cell vortex at $z/c = 2.2$ indicates the location at which the velocity profiles have been plotted in fig. 5.4c$_1$. Close to the surface the velocity of the flow has almost no streamwise component, apart from the measurement location near the LE which still falls outside the stall cell. Most measurement locations also contain upstream flow near the wing surface, although small changes in x/c may change the sign of the streamwise flow direction near the surface due to the locally rotating flow induced by the stall cell vortex. The stall cell vortices also introduce spanwise velocity components near the wing surface such as shown for the spanwise location $z/c = 2.2$ in fig. 5.5. It can be observed that upstream of the vortex the induced spanwise velocity component points away from the centre of the stall cell, whereas downstream of the vortex core the spanwise velocity points towards the centre of the stall cell. The induced spanwise component has significantly different velocity profiles depending on the chordwise position, indicating the complexity of the flow inside a stall cell due to the proximity of multiple vortices.

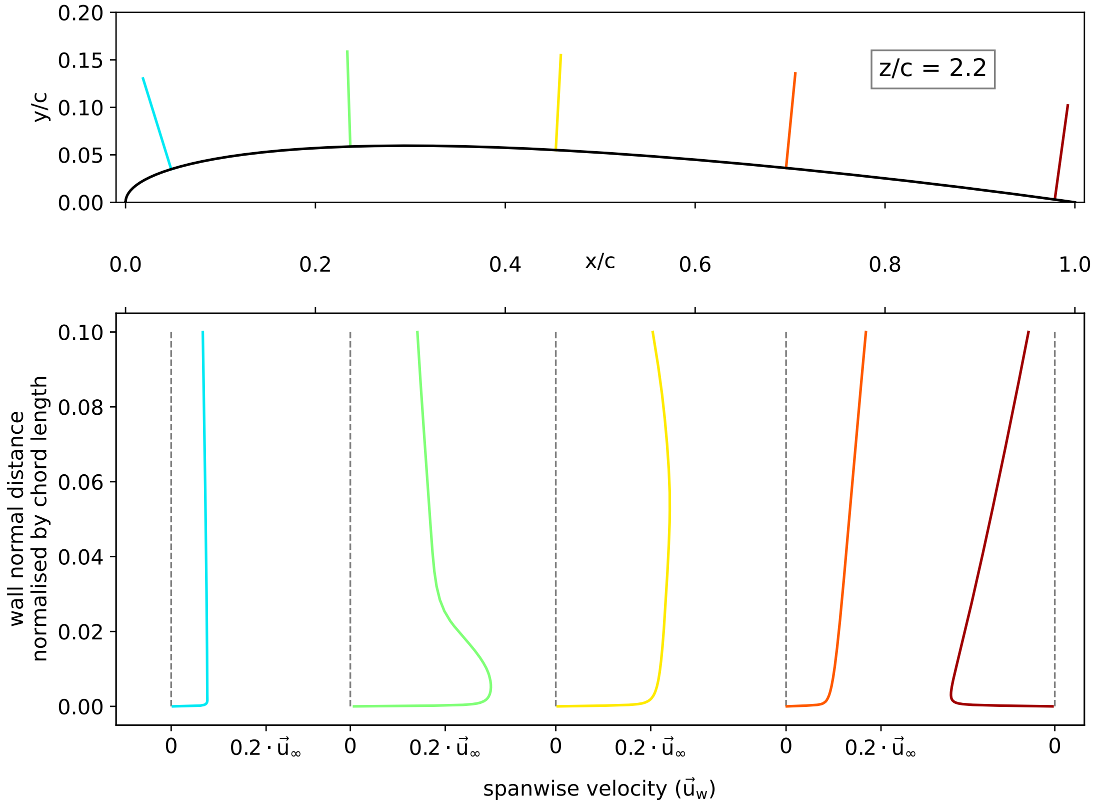


FIGURE 5.5: The spanwise velocity profiles at a spanwise location of $z/c = 2.2$, through the stall cell vortex, for a wing at an angle of attack of 18° and Reynolds number of 10^6.

The vertical grey line in fig. 5.3c which crosses the stall cellvortex at 1.32 z/c indicates the location at which the velocity profiles have been plotted in fig. 5.4c$_2$. This spanwise location indicates approximately the centre of the stall cell. At the centre of the stall cell the upstream flow near the wing surface is the largest in magnitude and spans the largest distance along the wing surface in the chordwise direction. In fig. 5.4c$_2$ it can be observed that each measurement location is contained in the stall cell near the wing surface as indicated by the upstream flow near the surface. It can also be seen that the thickness of this separated viscous region is very small near the LE and quickly grows very large when moving further towards the TE. The rate of thickness growth for the separated flow region in this location is significantly larger than observed in the cases with a smaller angle of attack, or away from the centre of the stall cell.

Finally in fig. 5.4d the velocity profiles are shown for the wing experiencing full separation along the entire span. All measurement points, except the one closest to the TE, have upstream flow near the wing surface. The growth rate of the boundary layer for this cases is the largest out of the cases considered in fig. 5.4. The last measurement point does not have upstream velocity near the surface of the wing due to a vortex located right above the TE, as shown in fig. 5.6. This vortex is a consequence of the low pressure area of separated flow over the wing suction surface and the high momentum flow along the pressure side of the wing. It can also be seen in fig. 5.6 that the separated flow above the wing surface has a size which exceeds the velocity profile measurements which extend only for 0.1 x/c away from the wing surface.

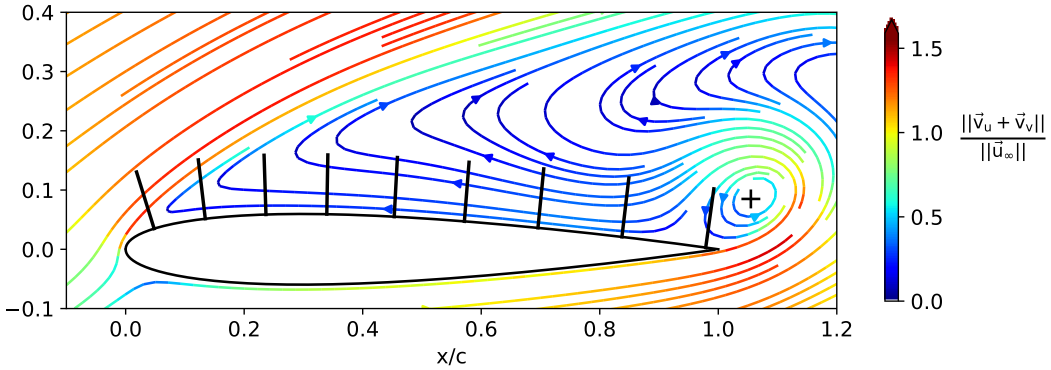


FIGURE 5.6: Streamlines at z/c = 1.25 for the case with full span separation at an angle of attack 18° and Reynolds number 10^6. The core of the TE vortex is marked with a "+" sign at x/c = 1.055 and y/c = 0.085.

5.4 Surface pressure

The flow behaviour on the wing surface can be visually identified through the use of SOFV or tufts, yet in most cases quantitative measurements are desired for a range of purposes such as design, safety or control. The ability to take high frequency data with minimal disturbance to the flow can be achieved through pressure taps. These measure the static pressure on the surface of the wing. Depending on the amount and position of the pressure taps a lot of information about the flow on the wing can be obtained. As indicated through surface flow visualisation it is only for certain combinations of angle of attack and Reynolds number that there are spanwise variations in the flow behaviour on a 2D extruded wing profile. In most cases it would thus suffice to have a single row of pressure taps along the chordwise direction to obtain representative data for the entire wingspan as presented by Zhou et al. (2021).

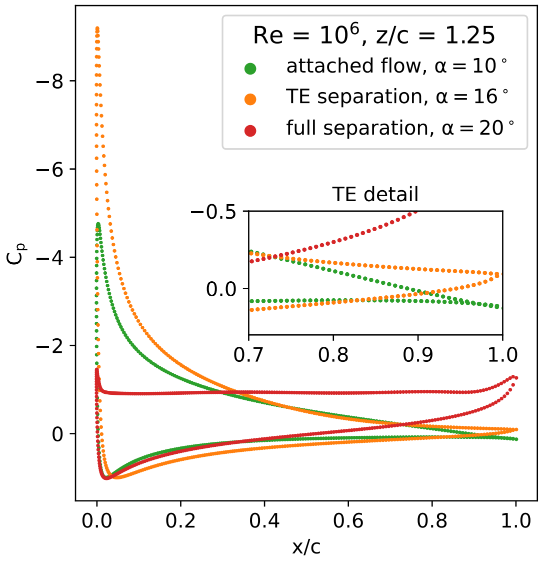


FIGURE 5.7: The pressure coefficient curves at the midspan for different surface flow variations at different angles of attack with a constant Reynolds number.

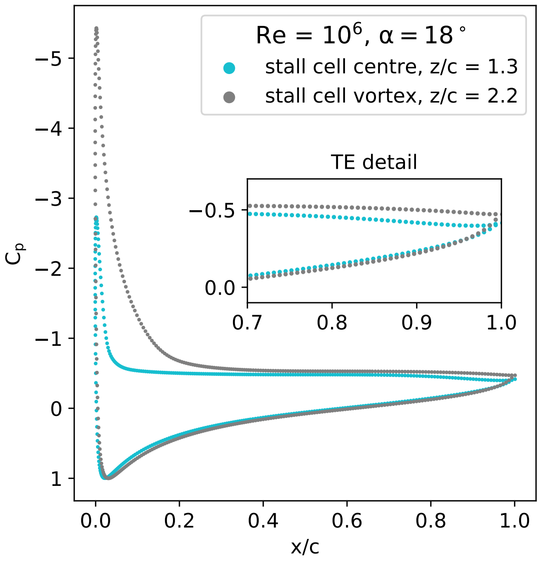


FIGURE 5.8: The pressure coefficient curves at two spanwise locations on the wing with a stall cell surface flow pattern.

In fig. 5.7 the pressure coefficient (C_p) curves along the airfoil circumference at the midspan location are shown for three types of surface flow which typically do not have any spanwise variation in flow. As such, the results shown here match the spanwise averaged results of Manni et al. (2016) very closely. Attached flow at an angle of attack of 10 degrees shows the typical characteristics of a stagnation point with a C_p of one near the LE. On the suction side of the airfoil near the LE a large (negative) pressure peak occurs, after which an adverse pressure gradient extends till the TE to create the pressure recovery on the suction side of the wing. When the angle of attack is increased the pressure peak magnitude increases, thereby increasing the magnitude of the adverse pressure gradient, which can lead to TE separation as shown in fig. 5.7 by the case with angle of attack 16 degrees. For this case the pressure gradient on the suction side near the TE becomes much closer to zero (as indicated by the nearly flat C_p in the TE detail of the curve), due to the separated flow near the TE. When the angle of attack is increased further to cause the flow to fully separate over the wing suction surface as shown in fig. 5.6 the pressure peak near the LE edge can reduce significantly in magnitude and the pressure coefficient over the suction surface becomes nearly constant as shown in fig. 5.7. In this figure it is also shown that the vortex near the TE has an effect on the pressure distribution by increasing the magnitude of the negative pressure locally. For cases with stall cells the pressure coefficient can vary significantly as shown in fig. 5.8. This spanwise variation in pressure coefficient has previously been described as related to vortices that trail downstream behind the stall cell, which cause an upwash in the centre of the stall cell and a downwash outside the stall cell as described by Yon and Katz (1998).

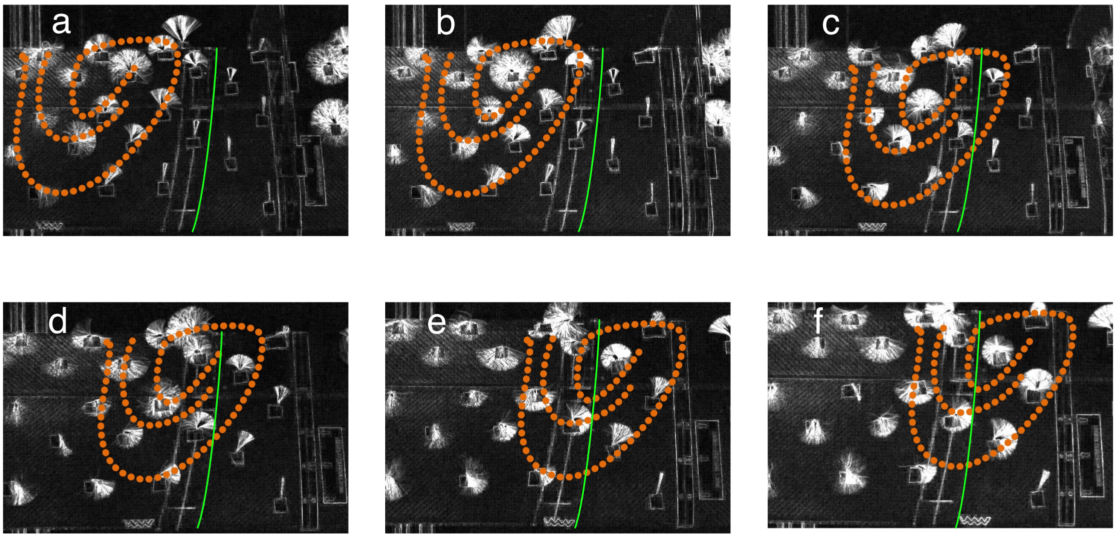


FIGURE 5.9: Images of tufts on a NACA 0012 wing at an angle of attack 14° and Reynolds number of $4.1 \cdot 10^5$. Movable wing fences and a zig-zag tripstrip have been used to move half a stall cell across a row of pressure taps. Freestream flow is from the bottom to the top of the images. The approximate layout of half the stall cell is drawn with a consistant orange dotted line pattern in each image, the location of the pressure taps is indicated with a green line. Image *a* has the pressure taps outside the stall cell, alphabetical consecutive images have the pressure taps further towards the centre of the stall cell.

In fig. 5.9 the visual results are shown of moving two fences and a tripstrip along the span of a stalled NACA 0012 wing with the goal of moving a stall cell over a (chordwise) row of pressure taps. The shape, position and behaviour of a stall cell are not perfectly controllable. As such, the different images in fig. 5.9 do not show consistent spanwise shifts of the stall cell. Both images *a* and *b* have the row of pressure taps outside the stall cell. Images *c* and *d* have the pressure taps already inside the stall cell, but still between the stall cell vortex and the stall cell outer boundary. Finally images *e* and *f* show the pressure taps on the stall cell vortex and between the stall cell vortex and the centre of the stall cell respectively. In fig. 5.10 the corresponding pressure coefficient curves are shown. From the experimental pressure measurements it can be seen that the negative pressure peak near the leading edge is the largest outside the stall cell. At these spanwise locations the flow also remains attached and shows an adverse pressure gradient till the trailing edge. For case *d* the pressure coefficient can be observed to be nearly constant for x/c between 0.3 and 0.8 in fig. 5.10, indicating that the flow is already separated. For the cases *e* and *f* the pressure coefficient is constant over nearly the entire chord. For case *f* specifically the standard deviation of the pressure fluctuations near the LE is higher, due to the zig-zag tripstrip being right at the spanwise location of the pressure taps near the leading edge.

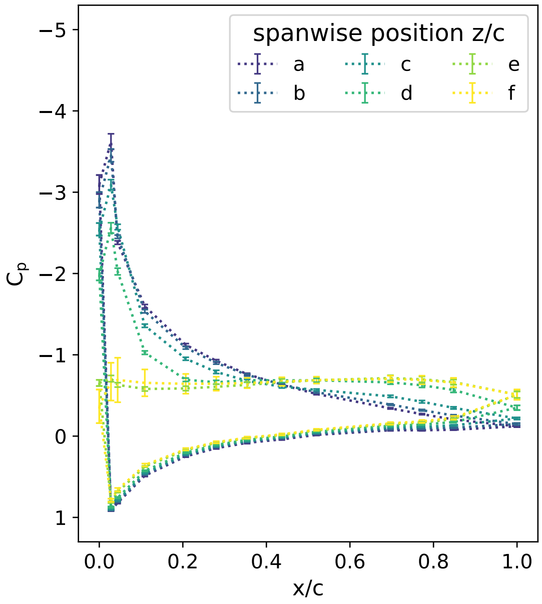


FIGURE 5.10: The experimental pressure coefficient curves at the locations such as indicated in fig. 5.9. The NACA 0012 wing is at an angle of attack of 14° and a Reynolds number of $4.1 \cdot 10^5$.

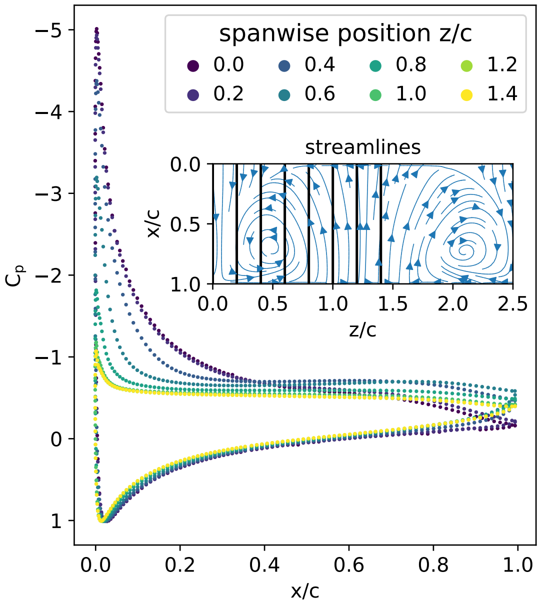
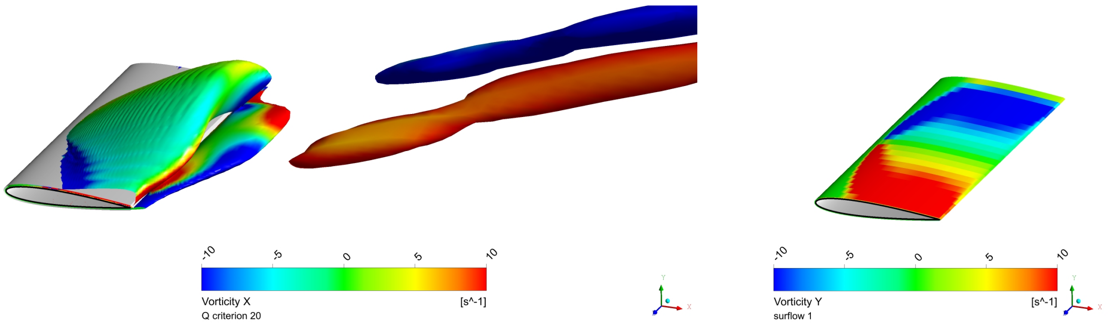


FIGURE 5.11: The computational pressure coefficient curves at the locations such as indicated in the inset. The NACA 0012 wing is at an angle of attack of 16° and a Reynolds number of $4.1 \cdot 10^5$.

In fig. 5.11 the pressure coefficient curves are shown of the computational results for a NACA 0012 wing at a Reynolds number of $4.1 \cdot 10^5$, similar to the experimental results in fig. 5.10. However in the computational results no stall cells were observed at an angle of attack of 14 degrees, only at angle of attack 16 degrees were stall cells observed computationally for the Reynolds number of $4.1 \cdot 10^5$ as shown in fig. 5.2. The streamlines shown in the inset in fig. 5.11 show the shape of the stall cell, the vertical lines indicate the locations at which the pressure coefficient curves have been obtained such as shown in the figure. Similarly to the experimental observations, the locations outside the stall cell or near the edge of the stall cell (0.0 and 0.2 z/c) show still mostly attached flow characteristics in the pressure coefficient plots. The locations contained between the stall cell vortex and the stall cell centre (0.4 - 1.4 z/c) show a mostly constant pressure coefficient on the suction side of the airfoil with a declining pressure peak near the LE when moving closer to the stall cell centre.



(A)

(B)

FIGURE 5.12: The NACA 0012 wing at angle of attack 18° and Reynolds number 10^6, with a stall cell on the suction surface of the wing. (A) Isosurfaces of the Q criterion equal to $20 s^{-2}$ are shown, which are colored by the x vorticity. (B) The y vorticity of the flow near the wing surface at an offset of 0.01 c from the wingsurface.

In fig. 5.12a the Q criterion has been used to create isosurfaces which indicate several characteristic aspects of a stall cell. The specific case shown has a stall cell on the suction surface which spans nearly the entire width of the wingspan. On the suction (top) side of the wing in the image the isosurface related to the shearlayer of the separated flow on the wing can be observed. Coming from the trailing edge another isosurface can be seen related to the trailing edge shearlayer which is a consequence of the fast flow on the pressure side and the separated flow on the suction side. Behind the wing two long tubular isosurfaces are shown which represent the stall cell vortices that trail downstream. These vortices are counter rotating as indicated by the x vorticity. The rotational orientation of the vortices can be seen to cause an upwash between the two vortices (stall cell centre) and a downwash outside the stall cell. Considering that a stall cell occurs at a post stall angle of attack, the downwash can cause the effective angle of attack to locally decrease outside the stall cell and result in regions of attached flow around a stall cell. In fig. 5.12b the y vorticity is shown near the wing surface, which indicates the vortices that are present near the surface and visible in the streamlines of the flow near the surface when stall cells are present, similar to the inset in fig. 5.11.

5.5 Lift coefficient

From the investigation of the pressure coefficient it is clear that the spanwise variation in flow, as induced by stall cells, has a significant impact on the wing lifting performance. The spanwise variation in lift coefficient is thus related to the type of stall and surface flow configuration on the wing. Additionally separated flow over the wing tends to be unsteady, which also has an effect on full span separation without spanwise variations in the flow.

5.5.1 Spanwise variation

As has been indicated in fig. 5.3, the spanwise variation in surface flow is only present for surface flow which shows stall cells. This spanwise variation in flow also has a significant effect on the lifting capacity of the wing. In fig. 5.13 the spanwise variation in lift coefficient is shown for the cases such as shown in fig. 5.3. From this figure it can clearly be seen that no spanwise variation occurs for attached flow, TE separation and even full span separation. These results have been obtained with 3D URANS simulations but suggest that 2D simulations might suffice. However it must be noted that full span separated flow rarely is fully 2D in practice. Small disturbances can have big cascading effects on the flow and URANS simulations are not suitable to investigate the required level of detail.

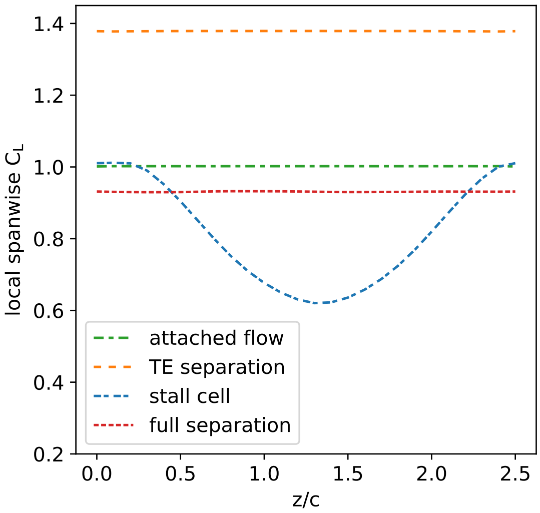


FIGURE 5.13: Spanwise variation in lift coefficient due to different surface flow types.

For cases with stall cells however a large spanwise variation in lift coefficient can be observed in fig. 5.13. The maximum local C_L along the wingspan can be observed to be approximately 1.0, whereas the local minimum along the wingspan is approximately 0.6. This local minimum thus indicates that a local reduction of 40% of the local maximum lift coefficient can occur due to the formation of stall cells. From the analysis of this specific case in fig. 5.3 and fig. 5.8 it can be concluded that the low local lift coefficient is related to the local upstream flow on the wing surface. This situation is also detrimental to the effectiveness of TE control surfaces. It is thus worrying that the positioning of stall cells is currently not understood well and might be due to arbitrary disturbances.

5.5.2 Unsteady variation

Next to the spanwise variation in lift coefficient it is also possible to have temporal variation in lift coefficient. Unsteady flow behaviour results in unsteady performance of the wing. The surface flow types which have been considered have only been investigated in a spatial manner, while the temporal aspect can be just as impactful on the wing performance. For the temporal variation investigation the time frame is given in convective cycles, defined as the chord length divided by the freestream velocity.

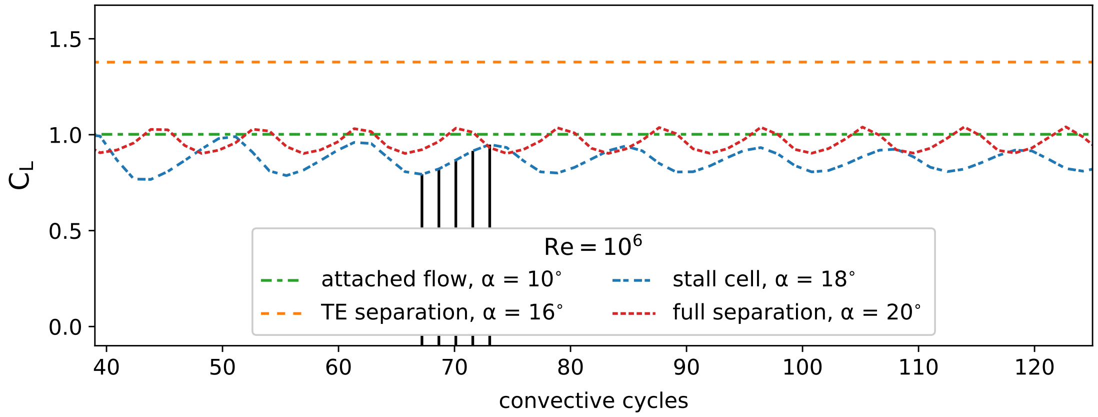


FIGURE 5.14: Lift coefficient variation over time for different surface flow types. The vertical black lines indicate the convective cycles at which the stall cell surface flow is investigated in fig. 5.15 and fig. 5.16.

In fig. 5.14 the lift coefficient of the entire wing is shown over the initial time frame of the URANS simulations. Each simulated case has a small transient time period at the start of the simulation, which is largely caused by the non-physical flow which has been set for the initialisation of the simulation. After a short amount of time (typically less than 2 simulated seconds) the flow will have adapted to the boundary conditions of the computational domain and start to represent the flow as representative for the simulated wing at a given $\alpha-$Re combination. The discussion of the wing performance variation over time does not consider this initial transient period. For the cases with attached flow and TE separation the lift coefficient is nearly constant over the entire duration of the simulation. These cases represent flow which is stable over time and thus could be solved with a more simplified method such as RANS rather than URANS. The cases with stall cells present on the wing surface result in unsteady flow, which also results in unsteady performance of the wing. This unsteady performance is easily characterised by the lift coefficient such as shown in fig. 5.14. Similar results were found by Liu and Nishino (2018). The lift coefficient for the full separation cases can be found to be steady at lower Reynolds numbers, or unsteady as shown in fig. 5.14 for higher Reynolds numbers. The unsteadiness is caused by vortices shedding from the wing into the wake which influence the pressure coefficient such has been shown by fig. 5.7.

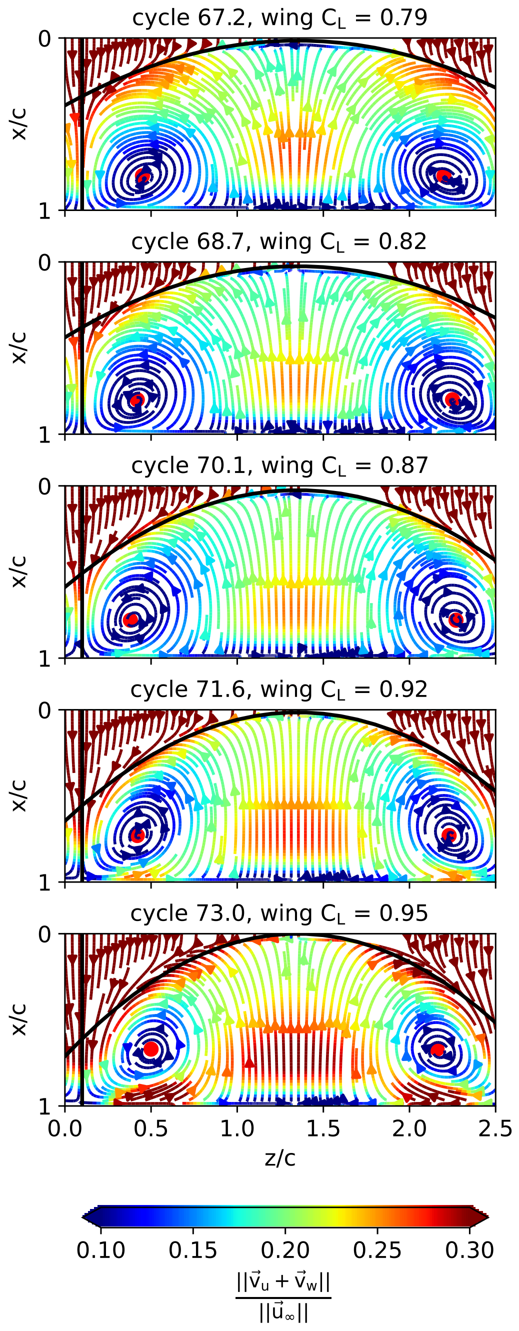


FIGURE 5.15: The surface flow at varying timesteps such as indicated by the vertical lines in fig. 5.14. The NACA 0012 wing is at an angle of attack of 18° and a Reynolds number of 10^6.

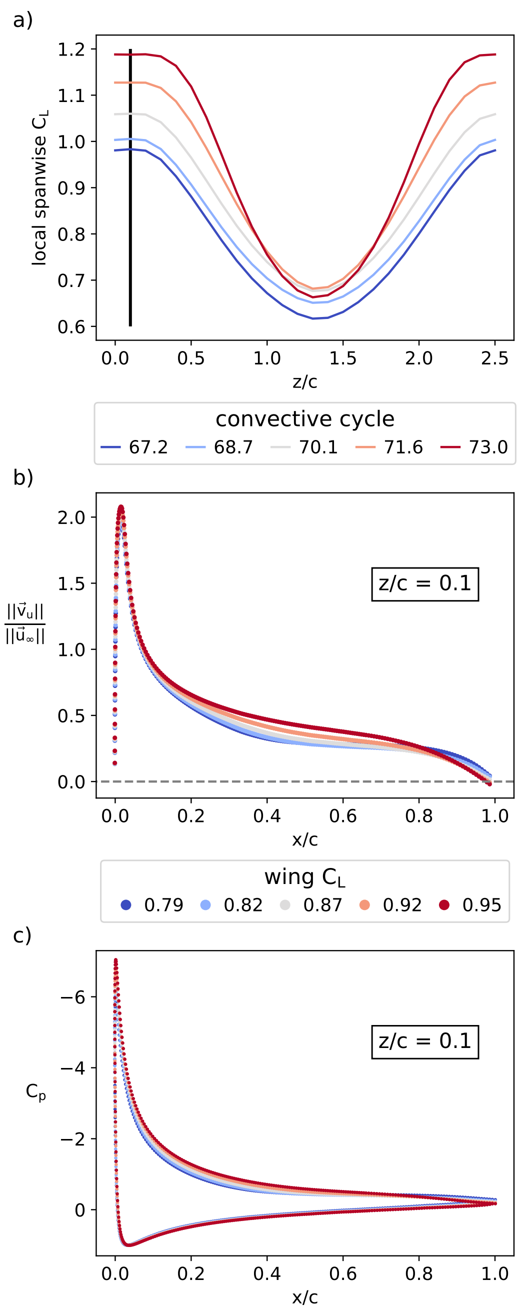


FIGURE 5.16: a) the spanwise variation in lift coefficient at different timesteps. b) the chordwise velocity magnitude along the spanwise location 0.1 z/c with mostly attached flow as indicated by the vertical lines in fig. 5.15. c) the pressure coefficient at the spanwise location 0.1 z/c with mostly attached flow.

In fig. 5.15 the surface flow patterns are shown for a single α−Re combination (as shown in fig. 5.14) at different timesteps. The surface flow patterns all clearly show a stall cell, although small differences in the exact patterns at different timesteps exist.

It can be seen that the velocity magnitude of the flow near the surface of the wing is changing over time. The shape of the stall cell can also be seen to change over time. This is also indicated by the black parabolic line which approximates the curved separation line at the front of the stall cell near the LE of the wing. However the current simulations with a wing with AR 2.5 produce stalls cells which occupy the entire span. As such, there is no opportunity for the stall cells to change in width. In fig. 5.16a the spanwise lift coefficient at the different timesteps is shown to understand how the change in integral wing lift coefficient is related to the spanwise flow variation as caused by the stall cell. It can be seen that the largest differences in spanwise lift coefficient occur in the spanwise regions between two stall cells (0.1 z/c in this case) where the flow is still mostly attached to the wing. At this spanwise location the chordwise velocity has been plotted as shown in fig. 5.16b where it can be seen that for the cases with lower wing lift coefficients the local velocity near the wing surface is also lower. While the flow velocity can be seen to decrease to approximately 0 near the trailing edge, the velocity at an offset of 0.001 c does not show any significant separation. The offset from the surface is constant at which the velocity has been measured and shown. It must be noted that this measurement is thus within a growing boundary layer. The local spanwise lift coefficient is better represented by the local pressure coefficient which integrates to the local lift coefficient. The local pressure coefficient at z/c 0.1 is shown in fig. 5.16c. The differences in pressure coefficient for different timesteps are less significant than the spanwise variation in pressure coefficient across the stall cell. For the timesteps with a lower wing lift coefficient the pressure peak near the LE is slightly lower and a minor difference persists along the chord length on the suction surface of the wing, resulting in a lower local lift coefficient.

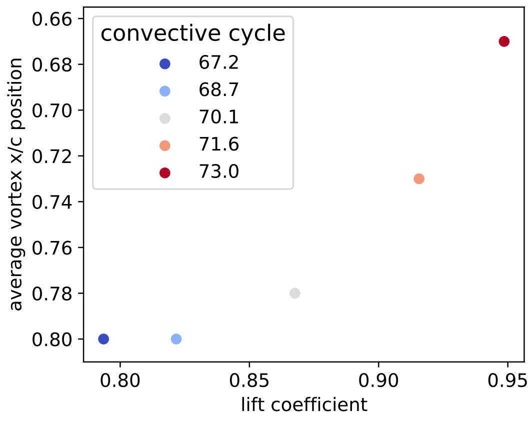


FIGURE 5.17: The average position of the stall cell vortices along the chord of the wing for varying wing lift coefficient values.

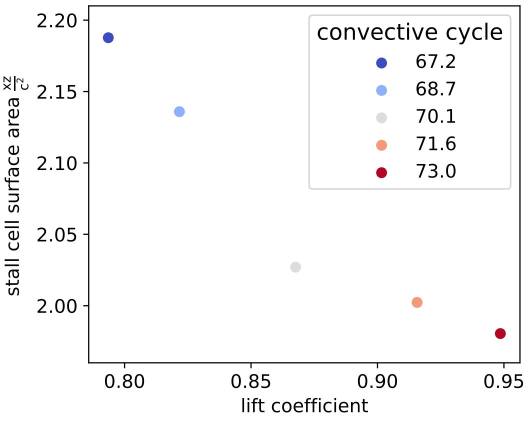


FIGURE 5.18: Approximation of the stall cell surface area based on the parabolic separation line as shown in fig. 5.15.

In fig. 5.15 the approximate location of the stall cell vortex centres (or cores) on the wing surface has been indicated with red dots. The stall cell vortices tend to be closer to the TE than the LE in almost all cases, however they do not have a fixed chordwise or spanwise location. For the case such as displayed in fig. 5.15 with a constant angle of attack and Reynolds number, the stall cell vortices have fluctuating positions over time. In fig. 5.17 the average chordwise position (of the two vortices on the wing) is shown for different lift coefficient values of the wing at different timesteps. It can be observed that the stall cell vortices are positioned closer to the LE for higher values of the wing lift coefficient. The differences in chordwise position of the stall cell vortices however are small and cannot be seen as the sole indicator of the performance variation due to stall cells. Rather the position of the stall cell vortices both chordwise and spanwise are parameters of the geometrical characterisation of stall cells. In fig. 5.15 the approximate shape of the separation line is also indicated with a parabolic curve. Integrating the area under this curve can serve as an approximation of the surface area covered by the stall cell as shown by Manolesos and Voutsinas (2014). In fig. 5.18 the variation of surface area covered by the stall cell over time is shown for the case such as the example displayed in fig. 5.15. The stall cell area is expressed as the x/c by z/c distances, as such a non-dimensional area is obtained which is relative to the wing chord and aspect ratio. From the figure it can be observed that the lower wing lift coefficients are associated with the higher stall cell areas. As the stall cell area mostly encompasses the separated flow region near the wing surface, which is associated with a smaller pressure difference between the pressure and suction side of the airfoil than attached flow, a larger stall cell will typically result in a reduced wing lift coefficient.

5.6 Stall cell 3D structure

The structure of a stall cell has been hypothesised about and investigated in the past. With the current URANS simulation results it becomes possible to investigate the flow patterns in and outside the stall cell in detail. The three dimensional layout of a stall cell has been shown to be a bubble of separated flow by Dell'Orso et al. (2016) with particle image velocimetry. In the current investigation it is possible to show streamlines in the computational results. In fig. 5.19 the three dimensional streamlines are shown which start near the stagnation point and go around the stall cell bubble. Streamlines are only shown for half the wing span, representing half of a symmetric stall cell, to improve visual clarity. From these streamlines it can be observed that the height of the bubble is the largest near the middle of the stall cell and tapers off towards the sides. The bubble starts at the separation line on the wing surface, which curves from near the LE edge at the middle of the stall cell towards the TE at the spanwise ends of the stall cell. The downwash caused by the wing in the wake is stronger near the edges of the stall cell than the middle of the stall cell. At the edges of the stall cell the flow is attached along a larger part of the wing surface in chordwise direction, than in the middle of the stall cell where the flow is separated over most of the wing surface. In fig. 5.20 the streamlines inside the stall cell are shown, which show the upstream flow near the stall cell centre. The stall cell bubble contains recirculating flow with an upstream direction near the wing surface between the spiral nodes on the wing surface. The vortex core of these spiral nodes appears to run along the span of the wing and terminates in the spiral nodes on the wing surface as indicated by the surface friction lines. Near the trailing edge a separate vortex exists with an opposite rotational direction, this vortex runs along the TE for the width of the stall cell.

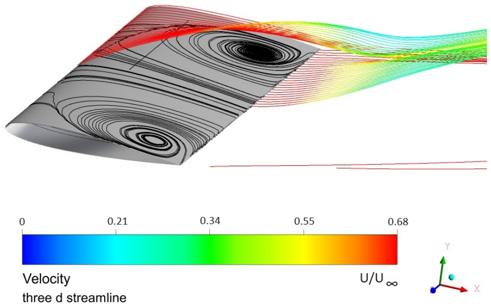


FIGURE 5.19: A snapshot image of the NACA 0012 wing at an angle of attack of 18° and a Reynolds number of 10^6. 3D streamlines which pass around the bubble of a stall cell along the span of the wing, from the middle of the stall cell to outside the stall cell.

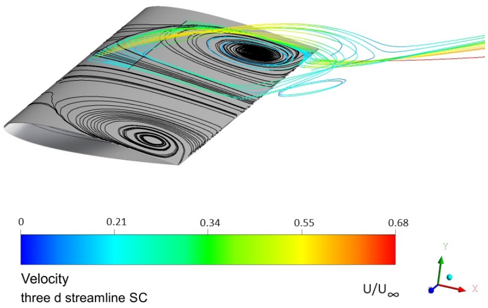


FIGURE 5.20: A snapshot image of the NACA 0012 wing at an angle of attack of 18° and a Reynolds number of 10^6. 3D streamlines inside a stall cell bubble, from the middle of the stall cell to outside the stall cell.

In fig. 5.21 the sideview is shown of streamlines inside the stall cell. The view includes streamlines along half the width of the stall cell. Two vortices can be seen, the approximate centre of these vortices (near the spanwise mid position of the stall cell) is indicated with a red dot. The first vortex is above the trailing edge and induces upstream flow near the wing surface while creating an upwash near the wing LE. The second vortex is located behind the trailing edge at the middle of the stall cell. This vortex has an opposite rotational direction compared to the first vortex and induces an upwash in the wake. These vortices only run along the width of the stall cell and therefore the upwash of the wake is reduced outside the stall cell, thereby inducing a vortex in the wake which trails downstream such as shown in fig. 5.22.

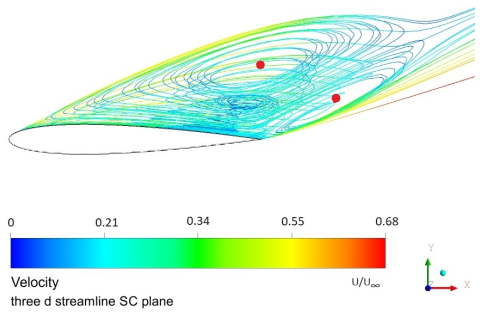


FIGURE 5.21: A snapshot image of the NACA 0012 wing at an angle of attack of 18° and a Reynolds number of 10^6. Near the middle of the stall cell two vortex cores (indicated by a red dot) can be observed with opposite rotational direction.

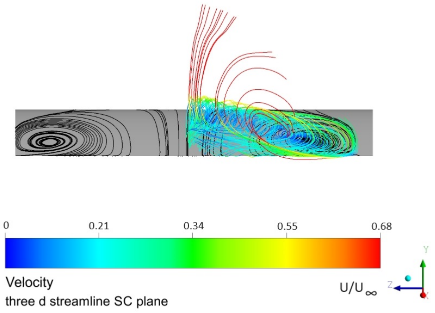


FIGURE 5.22: A snapshot image of the NACA 0012 wing at an angle of attack of 18° and a Reynolds number of 10^6. The variation in upwash in the wake along the span creates a vortex that trails downstream.

From the different perspectives of the streamlines in and around the stall cell it can be observed that outside the stall cell the incoming flow is curved more along the airfoil surface similar to attached flow. Whereas along the width of the stall cell the incoming flow is deflected less by the wing. The difference in downwash of the flow in the wake along the span induces a vortex in the wake. This process is shown in fig. 5.23, where on the left the structure is shown of the two vortices along the width of a stall cell. The configuration of these two vortices is very similar to the vortex configuration shown by Weihs and Katz (1983) during the proposed formation mechanism of a stall cell. In the current results these vortices have been observed to persist in established stall cells. The existence of these vortices along the width of the stall cell in spanwise direction still results in the commonly shown stall cell surface flow pattern. In the middle of fig. 5.23 a 2D perspective shows how these vortices have an impact on the flow which goes around the wing at a location near the middle of the stall cell (A in the figure) or a spanwise location outside the stall cell (B in the figure). It is also shown how both locations have different inflow angles as influenced by the vortex inside the stall cell, and different outflow angles as influenced by the vortex at the TE. The inflow and outflow angles can also be observed in fig. 5.19. On the right is shown how the difference in outflow angles imposes a vortex which trails downstream. This structure of the two vortices along the width is very similar to the structure as has been hypothesised by Weihs and Katz (1983). Furthermore the expected upwash near the centre of the stall cell as mentioned by Yon and Katz (1998) can be explained by both the vortex which connects the spiral nodes on the wing surface and the vortices trailing downstream.

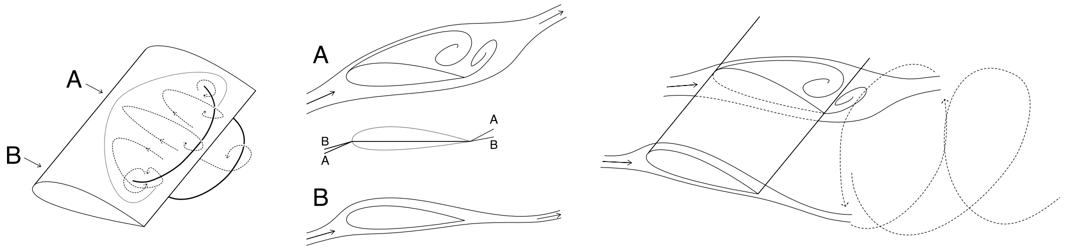


FIGURE 5.23: A scheme which represents the vortices involved in a stall cell. Left: two vortices run along the spanwise width of a stall cell. Middle: the difference in 2D incoming and outgoing flow angle near the middle of a stall cell and outside the stall cell. Right: the difference in flow angle in the wake induces a vortex which trails downstream.

5.7 Data driven modelling

The modelling of stalled flow over a wing can be done with CFD to varying degrees of accuracy. Typically the higher accuracy methods will be computationally more expensive. Such simulations may still show non-unique solutions for certain stalled flow configurations as discussed by Kamenetskiy et al. (2014) and Manolesos and Papadakis (2021). It has been proposed that the existence of non-unique solutions in the computational simulations might be due to the interaction of the separated flow with the trailing vortices Spalart (2014). Lifting line (or surface) models based on these trailing vortices have been constructed by Gross et al. (2015) and Plante et al. (2022), which showed that a negative $C_L - \alpha$ gradient is required for the trailing vortices to exist. These models further require the relevant $C_L - \alpha$ curve, filters and smoothing to reduce certain numerical effects that produce non-physical solutions. These models are interesting to help discover some of the driving factors behind the occurrence of stall cells, but are very limited in their application due to a lack of accuracy. Models with high accuracy that are also very fast would be of great interest for multiple applications. Fast and accurate physics based models typically do not exist as a fast model often relies on a range of simplifications which ignore elements that are relevant to separated flow. The blade element momentum theory is an example of a fast model based on simplified physics, which is commonly used for wind farms. The blade element momentum theory requires adaptations to take unsteady flow, yaw, tip losses and other aspects into account. More recently a lot of attention has been given to data driven models as they are able to provide very fast and accurate results, based on known data. These models are capable of interpolating known scenarios, which is both a strength and drawback. While fast and accurate, these models tend to require a large amount of known information to create such a model. When fast interpolation of known information is important such data driven models can prove valuable. Zhou et al. (2021) investigates the possibility of a one dimensional data driven stall detection model. The data driven model uses pressure measurements on the wing to infer other aerodynamic parameters. The investigation by Zhou et al. (2021) used KNN for regression and classification. The KNN algorithm is used to determine the parameters (both numerical and classes) of new samples, based on neighbouring training data samples, for which the desired parameters are known. The number of neighbouring training samples that are used in the classification of new samples has an influence on the outcome. Including too many neighbouring samples leads to the averaging of the results, whereas choosing not enough samples may lead to the inclusion of outliers. It remains challenging to choose the optimal number of neighbouring samples to use. Alternatively, KMeans can be used for clustering. The KMeans algorithm creates multidimensional clusters of samples. For new samples it can be determined which cluster they belong to. For this method it is crucial to determine the right amount of clusters to calculate from the training data. The same issue with the number of neighbours to include for KNN, is also present for the number of clusters to calculate with KMeans. By using methods such as compressive sensing as described by Bright et al. (2013) it is also possible to minimise the amount of required pressure sensors for an accurate result.

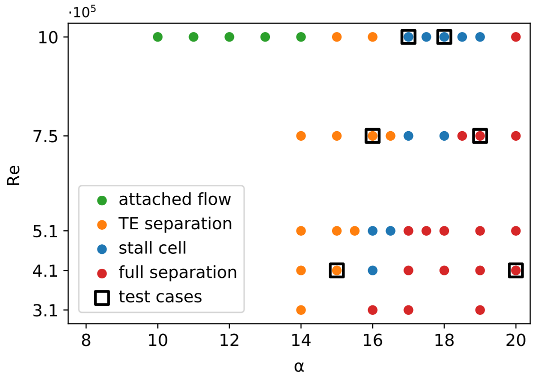


FIGURE 5.24: Identification of the flow type based on the angle of attack and Reynolds number.

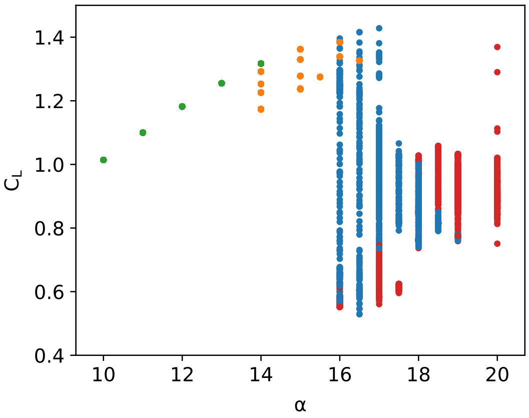


FIGURE 5.25: Identification of the flow type based on the angle of attack and lift coefficient for every individual time step of all simulated cases.

Many aircraft accidents are related to stall, where pilots or flight control systems are presented with wrong or incomplete information. The information used in such situations is typically the angle of attack and the airspeed (equivalent Reynolds number). Based on these parameters it is determined whether the wing is experiencing stall or not, and action is taken by automated systems or pilots. The recent example of the forced pitch down on the 737 MAX airplanes due to faulty angle of attack sensors (Lion Air flight 610 and Ethiopian Airlines flight 302) both resulted in a fatal accident, killing 346 people. These incidents are not unique and similar events have occurred multiple times in the past such as for the XL Airways Germany flight 888T or Air France flight 447 among others. It is essential to improve the assessment of whether the wing is stalling or not, while providing redundancy in the sensors used to assess this to reduce the number of fatal accidents. With the current methods used in airplanes the primary parameters to determine the flow type of the wing mostly rely on the Reynolds number and the angle of attack. This essentially comes down to the classification such as shown in fig. 5.24. In the simplest form curves can be estimated which form the boundaries between different flow types based on the $\alpha -$ Re parameters, which then determine the current flow type present on the wing. When the Reynolds number is desired to be replaced the lift coefficient can be used instead. While this is more complex to determine through sensors, it does give a clear indication of what flow type is present on the wing in combination with the angle of attack as shown in fig. 5.25. In fig. 5.24 and fig. 5.25 the angle of attack, Reynolds number and lift coefficient of a specific wing are shown in plots such as they are commonly presented. These three variables all have an influence on the flow type and can be combined in a 3D plot which gives a more complete overview of the relevance of each parameter such as shown in fig. 5.26. In this case it also shows how the attached flow has only been simulated at the highest Reynolds number with a low angle of attack, but this does not mean that it can only occur at this Reynolds number. The test matrix ideally should cover a range of angle of attack and Reynolds numbers that exceeds all probable combinations.

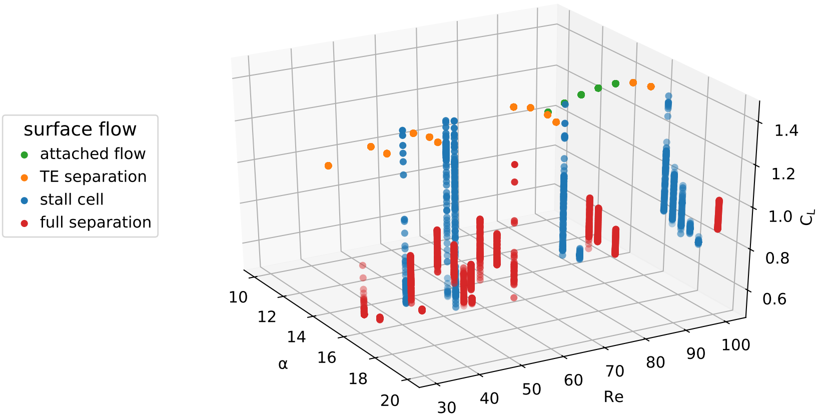


FIGURE 5.26: A 3D plot showing the relation between the angle of attack, Reynolds number, lift coefficient and the flow type.

From the airplane crash investigations it has been shown that the angle of attack sensors and pitot tubes can fail, showing that a redundant method for both would be desired. In an ideal case the entire flow on the wing surface would be known, either as the flow velocity near the surface or the pressure. As it is not practical to measure the flow velocity near the wing in operation, it has been chosen to utilise the pressure which can be measured with pressure taps and can be sampled at very high frequencies. The CFD simulations have used more than 200 grid nodes along the chordwise direction on the suction side of the airfoil. As it is desired to mimic a situation which is physically feasible and practical, the amount of grid points can be sub-sampled to a more realistic distribution for pressure taps on the physical object. Since the pressure side of the airfoil does not contribute much to the variation in pressure it also does not hold a clear relation to the flow type occurring on the suction surface. It has been chosen to sub-sample the suction surface with 15 probes. The locations of the subsampled probes along the chord are given in table 5.1. This distribution is realistic for the chordwise direction in a real life setup. The spanwise spacing is kept constant at 0.1 c which is the original grid spacing. As such there are a total of 390 (15 chordwise by 26 spanwise) probes under consideration for a wing with aspect ratio 2.5. In fig. 5.27 a comparison is shown between the available measurement points from CFD and the sub-sampled distribution by colouring the rectangular squares for which a measurement point is representative. The rectangles have an equal width in spanwise direction for both the original and the sub-sampled data, while the chordwise height of the rectangles is significantly higher for the sub-sampled measurements.

probe #	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
location [x/c]	0	0.02	0.04	0.07	0.10	0.14	0.18	0.24	0.30	0.37
probe #	**11**	**12**	**13**	**14**	**15**					
location [x/c]	0.46	0.56	0.68	0.83	1					

TABLE 5.1: Position of the subsampled pressure probes.

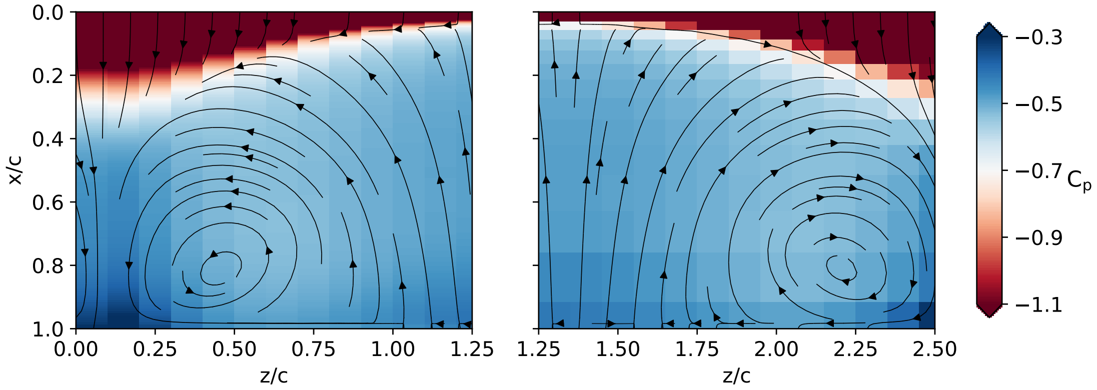


FIGURE 5.27: Comparison of the original pressure information available from the simulations (left) and the sub-sampled pressure data (right). The chordwise spacing for the sub-sampled data is much coarser than the original data. The flow pattern near the surface has been plotted to indicate the relation between the surface flow and the surface pressure, while also providing a spatial reference.

In fig. 5.28 the pressure coefficient curves are shown for the first half of the chord length. It can be observed that the subsampled spacing resolution is sufficient to approximate the pressure peak near the leading edge and the subsequent adverse pressure gradient.

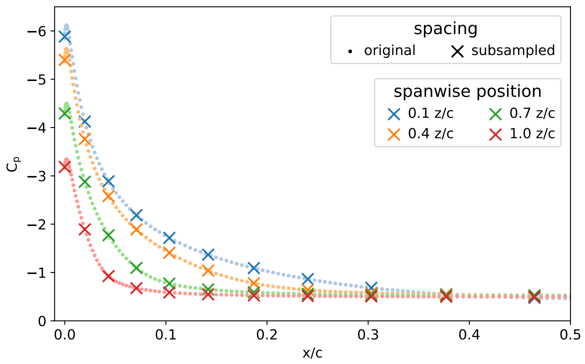


FIGURE 5.28: Pressure coefficient curves across half the width of a stall cell showing the front half of the chord length, for the original spacing of the pressure taps and the subsampled spacing.

5.7.1 POD of pressure

Machine learning methods allow for a flexible amount of inputs and outputs. They further allow for different purposes such as classification or regression. It is often chosen to use a large amount of inputs to provide a lot of information. However, this can lead to the neural network finding elements that are not relevant to the analysis but have high spurious correlations with the desired results. Preprocessing of the inputs for the neural network can help avoid the network learning undesired patterns in the data. In the current case it might be that the flow type on the wing surface can be coupled to the pressure value in a single location. It would be highly undesirable to have the classification of the flow type depend on a single sensor such as discussed above for the angle of attack sensor (of which there typically are multiple). In order to utilise the information of all pressure taps simultaneously they can be linearly combined into a new set of parameters. The mean has been subtracted from the surface pressure data set to centre the data. Then the POD has been calculated for the centred data set. In fig. 5.29 the first six spatial modes are shown as obtained from 4631 training samples with 390 pressure probes per sample. The modes have been obtained from samples across a range of angles of attack and Reynolds numbers. Most $\alpha - \text{Re}$ combinations which have been simulated did not show significant unsteady surface pressure. As such, the largest variance in the data set is caused by the changes in angle of attack or Reynolds number. The first mode can be interpreted as an approximation of the shift in mean surface pressure for different $\alpha - \text{Re}$ combinations.

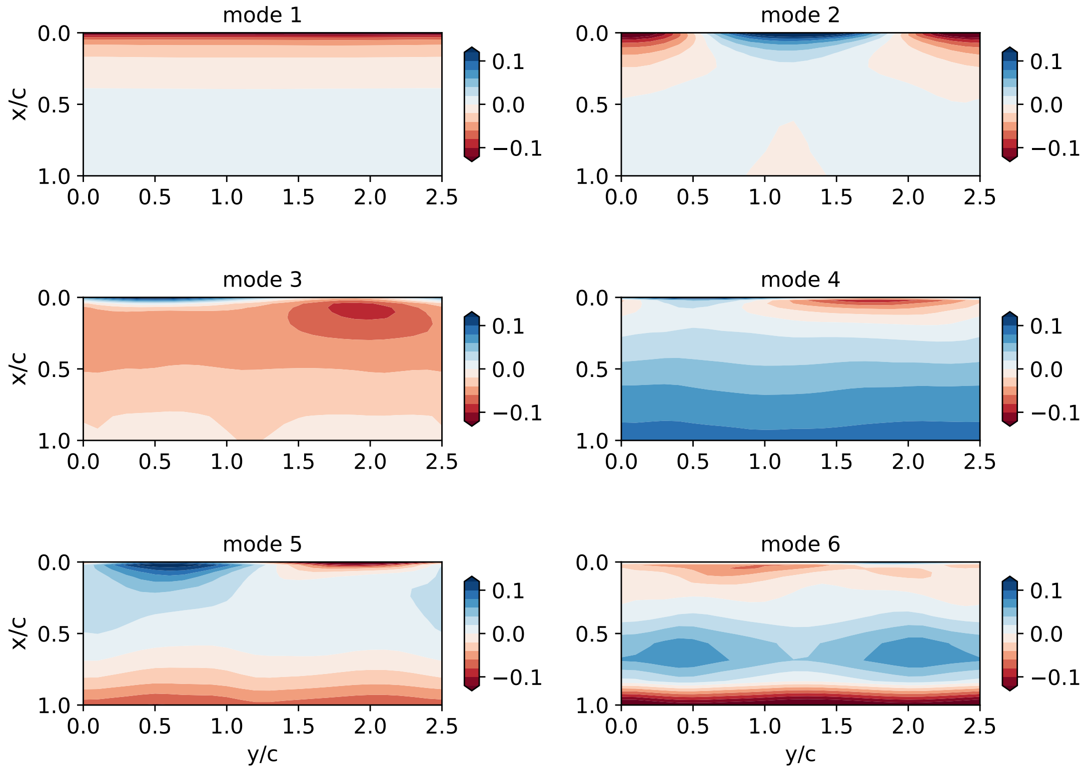


FIGURE 5.29: The first 6 spatial modes of the pressure POD.

In fig. 5.30 the singular values are shown of the spatial modes, the singular values quickly decrease by orders of magnitude. The singular values are linear coefficients and are easily understood to have a very large influence on the decomposition or reconstruction, where the first mode is roughly a 100 times more ‘important’ than the 10th mode. Due to the quick decline in the singular values for increasing spatial mode numbers only a limited amount of spatial modes are required to accurately capture the original sample. This principle can be shown by the cumulative explained variance ratio which illustrates the ratio of variance that can be captured in the training data set with a limited amount of spatial modes included, relative to the original data. This ratio is shown in fig. 5.31 where it can be seen the ratio very quickly approaches a value of one, where the value of one indicates that the complete variance in the original data set is reproduced. When the samples in the data set are linearly independent the ratio of one for the cumulative explained variance will only be obtained when all spatial modes are included. In the current case it can be seen that with only 10 modes included already a ratio of more than 99% can be achieved. This implies that a limited set of modes can accurately represent all the variations of surface pressure across all training samples.

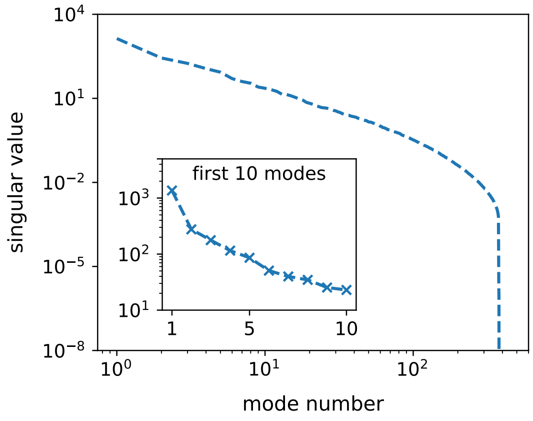


FIGURE 5.30: The singular values associated with each spatial mode of the pressure POD.

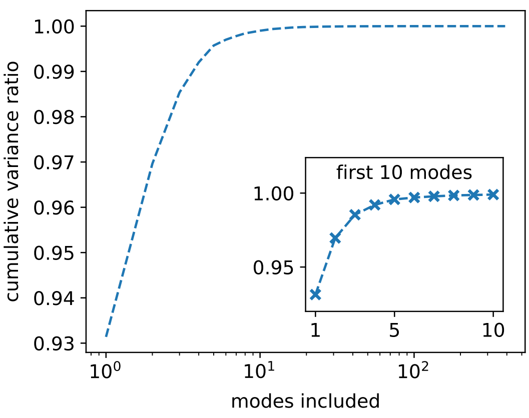


FIGURE 5.31: The cumulative explained variance ratio for an increasing amount of modes included of the pressure POD, starting with the modes with the highest singular values.

By using a limited set of trained POD spatial modes and corresponding coefficients a truncated reconstruction can be made of any sample. In fig. 5.32 the truncated reconstruction of four samples is shown. A reconstruction with just 2 modes already captures more than 95% of the variance in the data set. The truncated reconstruction with 10 modes in fig. 5.32 illustrates the accuracy of reproducing the patterns of the surface pressure within a small range of the pressure coefficient.

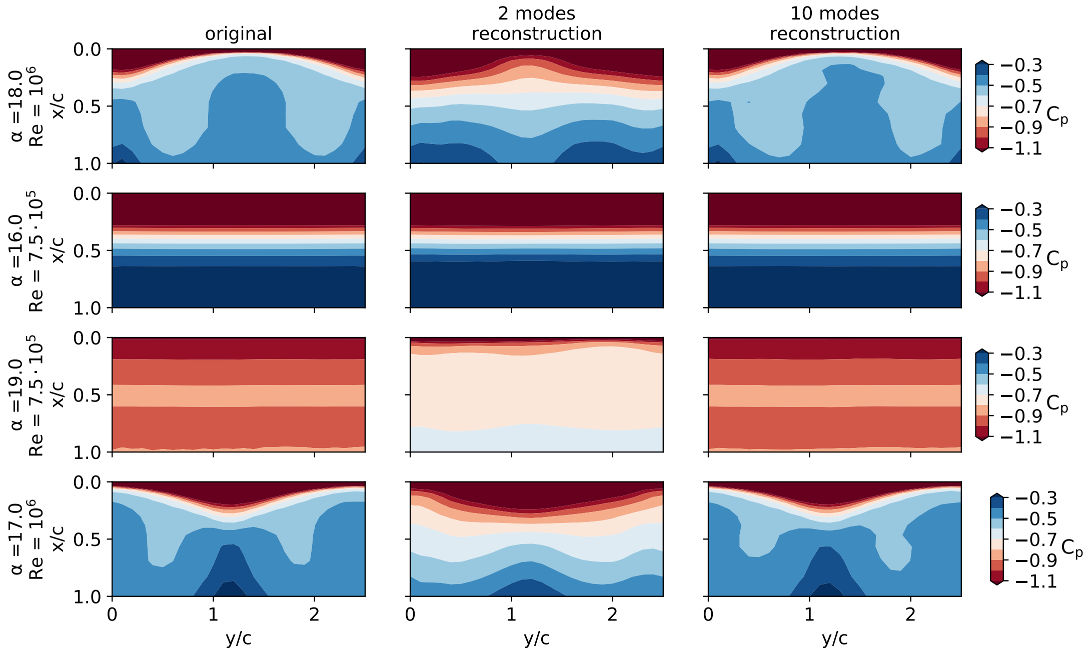


FIGURE 5.32: The surface pressure of four different samples is shown in the left column. The middle columns shows a truncated POD reconstruction with 2 modes included and the right column represents a reconstruction with 10 modes.

The spatial modes represent orthogonal vectors which thus allows to visualise the first three modes similarly to fig. 5.26 with a 3D plot of the first three POD mode coefficients for each sample, coloured by the flow type. In fig. 5.33 these POD coefficients have been plotted. It can be seen from this plot that the cases with full separation are clustered closely together with some variation possible. This likely reflects the repetitive vortex shedding at higher Reynolds number as well as the steady full separation at lower Reynolds numbers. The attached flow is represented by a single location for each separate case forming a 3D curve. This is because each separate case displays only a single constant pressure surface for the entire simulation. The TE separation surface pressure samples lie closely together with more variation per case but still structured, reflecting minimal variation per case. Lastly it can be seen that the cases with stall cells are present in large parts of the domain, spanned by all 3 POD modes.

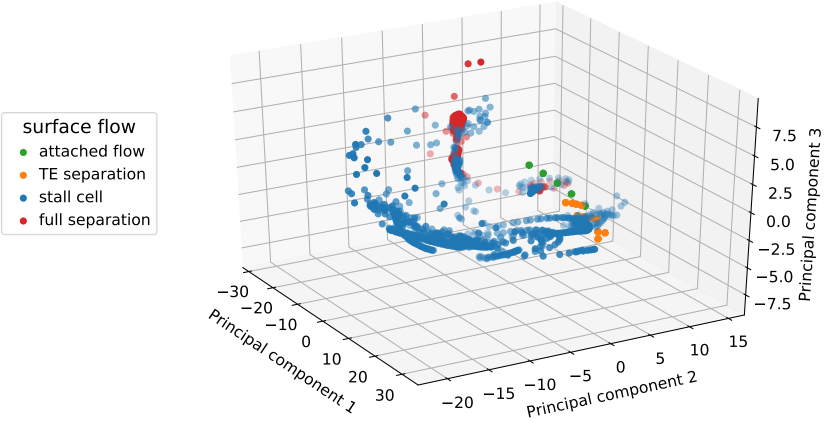


FIGURE 5.33: A 3D plot showing the relation between the POD coefficients of the first three POD modes.

In fig. 5.34 a 2D view is shown with the first and second POD modes as the axes. With this view it is clear that the surface pressure samples with attached flow or TE separation all occur within the first quadrant, this means a positive coefficient for the first and second modes. The full separation occurs exclusively with a negative coefficient for the first mode and a positive coefficient for the second (with exception of two samples). Lastly the stall cell samples can be found in multiple regions. Two of these regions are specific to certain flow configurations. The circular region around [0,12] for the [first mode, second mode] location indicates that on top of the average surface pressure coefficient map the second mode has a big influence. The positive sign of the second POD mode indicates a pressure coefficient near the center of the wing span with a lower pressure peak such as shown in fig. 5.29. This situation occurs when a single stall cell is on the wing surface such as shown by fig. 5.11. Alternatively when the second mode has a negative coefficient, it indicates the presence of a stall cell which is shifted in spanwise direction by half a span length.

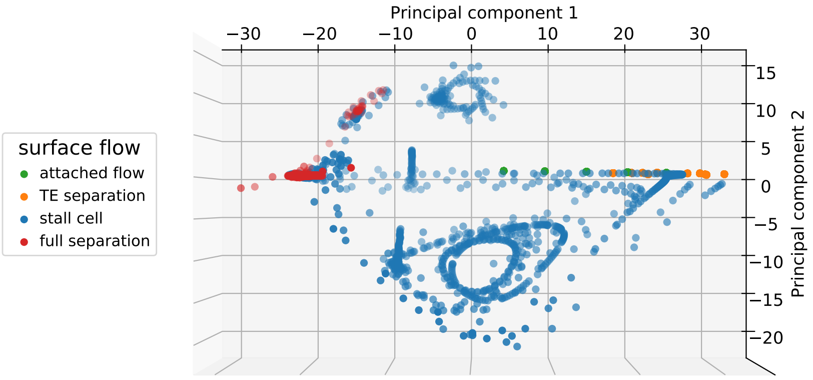


FIGURE 5.34: A 2D plot showing the relation between the POD coefficients of the first two POD modes.

The investigation with POD modes of the different types of surface flow, through the decomposition of the surface pressure, shows clear grouping of the flow types in a space spanned by the first three modes. This indicates that only a limited amount of information from the pressure taps is necessary to provide a good indication of the flow type that is present on the wing.

The POD analysis serves to reduce the dimensionality of the data set obtained with the chosen probe locations. Different positions of the probes will influence all aspects of the data driven analysis. For the analysis with the neural network that follows it is required to have a minimal number of modes necessary to identify the flow type. As such, choosing probes with high variance or important locations can lead to a lot of information captured in a minimal number of modes. The choice for the probe positioning will determine the information in the first several modes, and thereby affect the surface flow classification performance. The minimisation of the number of probes and the optimisation of their positions are interesting investigations with different considerations than the current proof of concept investigation. The current POD analysis and surface flow classification has also been evaluated with linearly spaced pressure taps, which resulted in a very similar POD basis and similar flow type classification performance by the neural network. The details of this evaluation are discussed in section 5.A.

5.7.2 Neural network for flow type

The POD modes allow to capture a large part of the information from the pressure probes into a limited set of POD coefficients. These coefficients by themselves do not allow for immediate classification of the flow type but require an additional algorithm or function to do so. A neural network can be trained based on the training data to perform the classification. It has been chosen to use just two POD modes, as the spatial modes and their coefficients can be visualised easily. This allows to illustrate the inner workings of the neural network.

The network has been chosen to have only one hidden layer and four outputs corresponding to the four relevant flow types. This network consists of 5 equations with 11 trainable parameters. The equation for the hidden layer contains three parameters, two coefficients for the two inputs and one bias. Each of the four output equations also has a coefficient and a bias, which are applied to the outcome of the hidden layer. This network can be trained on the available data set with the POD coefficients of the first two modes as inputs. In fig. 5.35 the metrics for evaluating the performance of the network are shown for the training and validation data during training. The accuracy is defined as the percentage of samples for which the predicted flow type matches the true flow type. The accuracy metric is not compatible with the optimiser and back-propagation for updating the network coefficients. As such, a cross entropy (CE) loss is used:

$$\text{CE loss} = -\sum_{n=1}^{p} y_n \cdot log(\widehat{y}_n), \tag{5.1}$$

where y_n represents the true probability of a sample belonging to class n, and $\widehat{y}_n$ the predicted probability with p possible classes. The current implementation of the cross entropy has been adapted to use integer encoding in the true probability description (sparse categorical cross entropy), in combination with the network outputting p values $\widehat{y}'_n$. These output values are scaled with:

$$\widehat{y}_n = \frac{e^{\widehat{y}'_n}}{\sum_{n=1}^{p} e^{\widehat{y}'_n}}, \tag{5.2}$$

such that $\sum_{n=1}^{p} \widehat{y}_n = 1$. The cross entropy loss, as shown in eq. (5.1), can be used as a measure of difference between the input distribution of one-hot encoded categories (a value of ’0’ for all categories, except ’1’ for the category the sample belongs to) and the predicted distribution from the network for different categories. The validation data is a random subset of 500 samples from the full training set that has been used to create the POD modes. It can be seen that the improvements in accuracy largely come in steps rather than gradual increases, and most of the accuracy improvements are achieved very early on in the training process. The training has been stopped automatically when no improvement in the loss of the validation data set is found for 100 consecutive epochs.

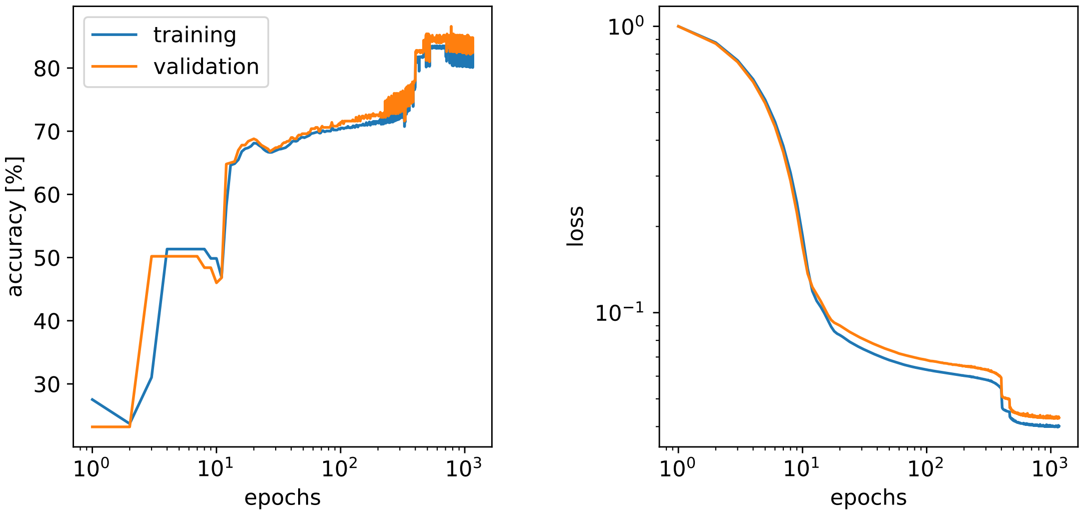
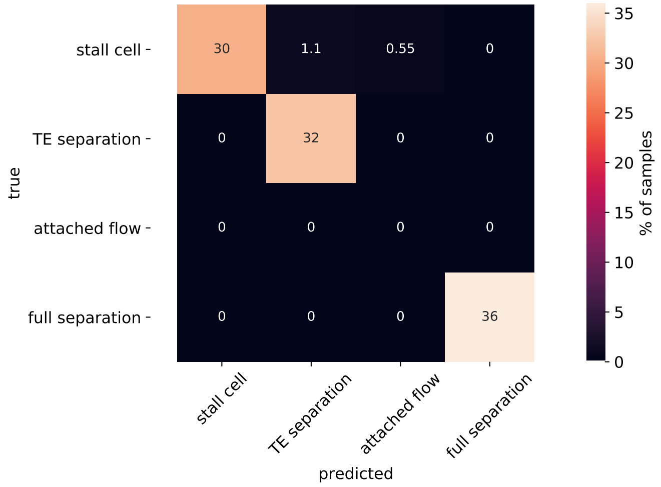


FIGURE 5.35: The accuracy and the loss (categorical cross entropy) of the model for classifying the flow type during training. The loss for both training and validation has been normalised with the respective first iteration loss.

Once the training is stopped the performance can be checked on the test data set, which has not had any influence on the training process. In fig. 5.36 a confusion matrix is shown for the evaluation of the test data set with the trained neural network. The matrix shows that when a stall cell is present a small chance exists that it is miss-classified as trailing edge separation or attached flow. Cases with trailing edge separation and full separation are correctly classified without any mistakes.



true \ predicted	stall cell	TE separation	attached flow	full separation
stall cell	30	1.1	0.55	0
TE separation	0	32	0	0
attached flow	0	0	0	0
full separation	0	0	0	36

FIGURE 5.36: Confusion matrix showing the results on the test data set for the classification of the flow type with two POD coefficients.

The results can also be visualised such as shown in fig. 5.37. From the shaded regions it can be seen that the domain spanned by two POD modes is divided by linear lines into different flow types. The angle of these lines and the offset determine the individual boundaries between two different flow types. As was shown by the confusion matrix, some stall cell samples fall in the regions where the network predicts attached flow or trailing edge separation.

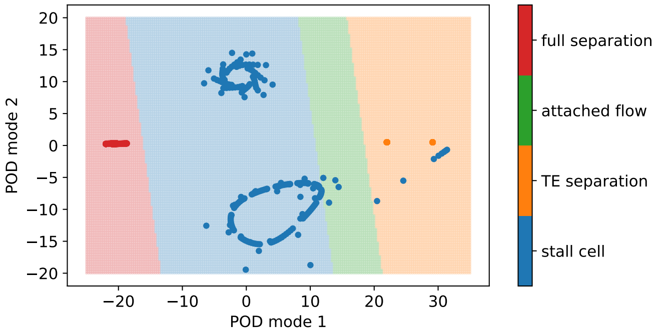


FIGURE 5.37: The results of the single hidden layer with one node neural network. The shaded area shows the interpolated predictions for the flow type based on the first and second POD coefficients. The individual dots indicate the test cases.

The equation of the single node in the single hidden layer:

$$\text{output} = \text{POD}_1 \cdot W_1 + \text{POD}_2 \cdot W_2 + B, \tag{5.3}$$

this node uses two POD coefficients (POD_1 and POD_2) which are multiplied with weights and biases is added to get the node output. An activation function which sets all negative values to zero is applied to the output. Multiple activation functions exist and could provide potentially better results. However, for the simplicity of the explanation a ReLU activation is used. A ReLU activation only finds the maximum between 0 and the input (which is the output of a node), making this very computationally efficient. A ReLU activation function is commonly used in combination with feed forward dense neural networks. The equation of the single node can be re-written into the form of a line for a fixed output:

$$\text{POD}_2 = -\frac{W_1}{W_2} \cdot \text{POD}_1 + \frac{\text{output} - B}{W_2}. \tag{5.4}$$

The equation is of the form 'y = mx + c' with m the gradient in this case determined by the negative ratio of the weights of the node and an offset determined by the fixed output value, the weight of the second POD mode and the node bias. The weights have been extracted from the trained neural network to allow to plot the outcome of the single node hidden layer. Using eq. (5.3) the output of the single node prior to the activation function has been calculated as shown in fig. 5.38. The vectors in the image indicate the direction for no change in the single node layer output and maximum change. When no change in the single node output value occurs, it also is impossible for following layers to indicate a different flow type. It can be seen that the vector of no change indicates the direction between the two clusters with stall cells, which are spaced apart in the domain. The vector indicating the direction of maximum change is approximately aligned with the line on which the full separation cases and trailing edge separation cases lie.

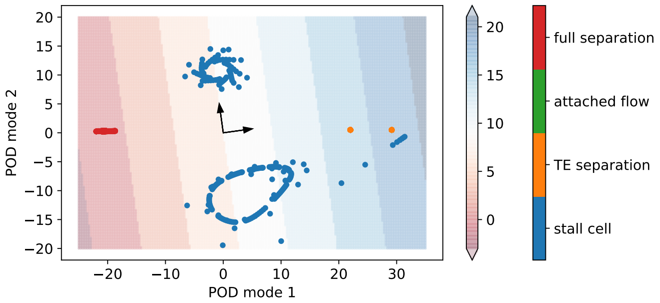


FIGURE 5.38: The shaded area is coloured according to the value of the output of the single hidden node. The vectors given represent the direction of no change in the output value and maximum change.

The output of the single node is then the singular input to four different linear equations. The output of these equations can be normalised with a softmax function, such that the outputs represent the probability, which sums to one when including all outputs. Each output also represents a given flow type, or class. Each data sample is classified as belonging to the class with the largest probability. These four equations have been visualised in fig. 5.39 for an applicable range of the single layer output values (where the normalisation function is omitted). Each line has a different gradient and offset and the local maximum value between all output ranges from 3 to 10 approximately. A negative domain is shown, however this is effectively not used by the neural net. The neural network uses an activation function to set all output values of the single node that are negative to zero. As such the equations shown in fig. 5.39 would have a constant value in the negative domain, equal to their value at zero.

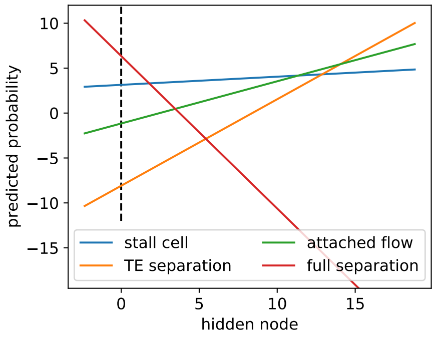


FIGURE 5.39: The different lines correspond to the equations of the different output nodes.

From the combination of the single node output (fig. 5.38) and the interpretation of that output by the final four nodes (fig. 5.39) the final classification can be found (fig. 5.37) by taking the classification with the highest output value at each position. This network although extremely simple already shows the ability to use a data driven approach to improve our understanding and the ability to estimate desired elements for which no applicable equation is available. More complex neural networks can create multidimensional criteria for the identification of the flow type, such networks ideally are also supplied with more inputs and more extensive training data.

5.8 Conclusions

Complex flow patterns such as stall cells can occur on wings. These flow patterns can be challenging to investigate experimentally due to the complexity of simultaneously detecting the presence of stall cells and acquiring quantitative data. The detection of the type of flow which is present on a wing is important for aircraft safety, control and stability. The type of flow on a wing surface of a simulated NACA 0012 wing has been extensively investigated. For a range of angles of attack and Reynolds numbers the flow near the surface has been investigated. The characteristic stall cell patterns on a wing surface are the result of the flow directions very close to the wing surface. Experimental and computational surface pressure measurements have been taken at similar locations on the wingspan relative to the stall cell surface visualisation pattern. The comparison showed good agreement between the experimental and computational results. The investigation of the computational data showed that stall cells can induce a large deviation of approximately 40% reduction in the local spanwise lift compared to the attached flow sections on the wingspan. Additionally separated flows tend to be unsteady, which in the case of stall cells is observed by the undulating movement of the stall cells. For full separation cases vortex shedding was observed which induced unsteady lift. Each of the surface flow types (attached flow, trailing edge separation, stall cell or full separation) has distinct characteristics in the surface flow patterns and also in the measurable performance such as the surface pressure. The POD has been used to obtain spatial modes of the surface pressure which capture some of the most important patterns. With just two modes more than 95% of the variance in the data set can be captured. The data projected onto the POD basis also shows some clustering. A neural network has been used to classify surface pressure measurements into different flow types based on the first two POD coefficients of the pressure map. This neural network consisted of merely 11 parameters and achieved an accuracy of 98%.

5.A Appendix: alternative pressure probe spacing

Withing the current investigation it has been mentioned that the spacing of the pressure probes, subsampled from the CFD meshgrid, has been chosen to resemble the spacing of pressure taps commonly used in practice. The spacing was chosen to be denser near the leading edge of the wing than the trailing edge. For this investigation the assumption is that the pressure on the wing is known everywhere by linearly interpolating the CFD meshgrid spacing. As such the probe locations, which will supply the necessary pressure information to predict the surface flow type, can be chosen freely. The robustness of the algorithm which classifies the surface flow type depends on the input information supplied. Variations of the input can be introduced by for example changing the number of probes and the probe spacing. In order to have redundant information a large number of probes are used, which in turn reduces the reliance on a single sensor to provide critical flight performance information.

The goal of the current investigation is also to provide an explainable algorithm which determines the flow type. This requires the neural network to be extremely small. In order to still utilise the all the information provided by the pressure probes the POD can be used. The decomposition into modes is commonly used to analyse the patterns in the data which contribute significantly to the overall variation in the data set. At the same time, it allows to compress the data into a truncated set of temporal modes in combination with the trained spatial modes. The compressed latent space then allows to reduce the number of inputs for the neural network.

Choosing (a different number of probes or) different probe locations will alter the distribution of the obtained samples that form the data set. As such, this will result in different spatial modes, singular values and temporal modes. Probes near the leading edge have a higher variance than near the mid chord position across different $\alpha-$Re cases. Probe locations near the leading edge also show spanwise significant variation in pressure for cases with stall cells. The CFD samples have been subsampled linearly in chordwise direction, with the same spanwise locations as before and without changing the number of probes. The pressure probe locations for this linear subsampling are shown in table 5.A.1.

probe #	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
location [x/c]	0	0.07	0.14	0.21	0.29	0.36	0.43	0.50	0.57	0.64
probe #	**11**	**12**	**13**	**14**	**15**					
location [x/c]	0.71	0.79	0.86	0.93	1					

TABLE 5.A.1: Position of the linearly subsampled pressure probes.

Some of the results which follow from this change in probe positioning are shown below. The linearly subsampled pressure for a single $\alpha-$Re combination at a single timestep is shown in fig. 5.A.1. Different curves show the pressure variation along the span of the wing. Compared to the non-uniform spacing which was used previously, the current linear spacing is not able to capture a similar amount of variation in C_p occurring near the leading edge pressure peak.

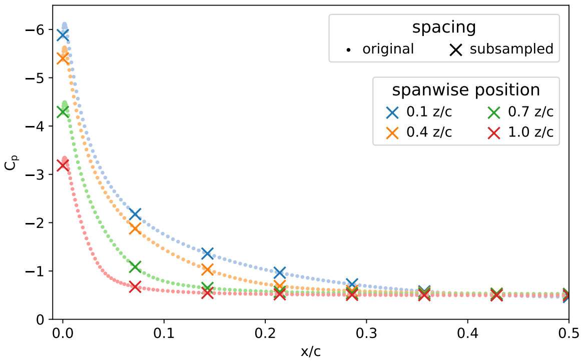


FIGURE 5.A.1: Pressure coefficient curves across half the width of a stall cell showing the front half of the chord length, for the original CFD spacing of the pressure taps and the linearly subsampled spacing.

In fig. 5.29 the first six spatial modes were shown which were obtained with non-uniform pressure probe spacing. In fig. 5.A.2 the first six POD spatial modes are shown which were obtained with linearly spaced pressure probes.

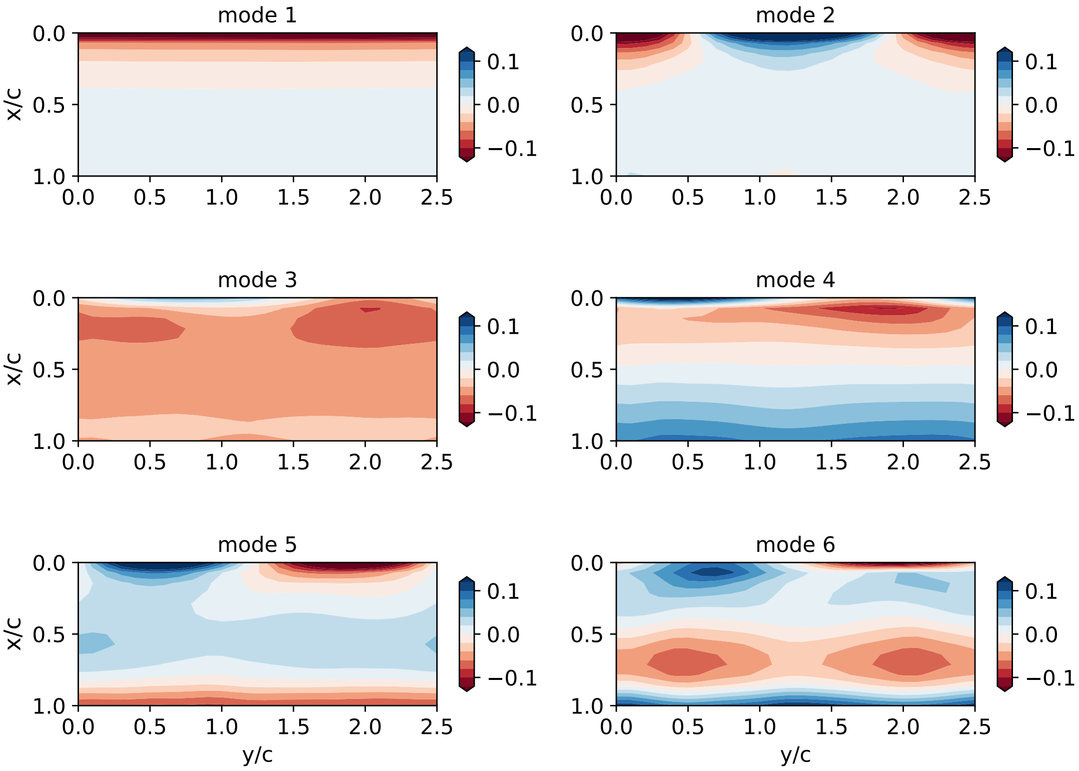


FIGURE 5.A.2: The first 6 spatial modes of the pressure POD obtained with linearly spaced pressure probes.

From the visual comparison it can be seen that the spatial modes are similar for the different spacing of the pressure probes. A more quantitative assessment can be made with the cumulative explained variance ratio, as shown in fig. 5.A.3. Similarly as for the non-uniform distributed pressure probes, the resulting POD from the linearly spaced probes captures more than 95% of the variance with just two modes.

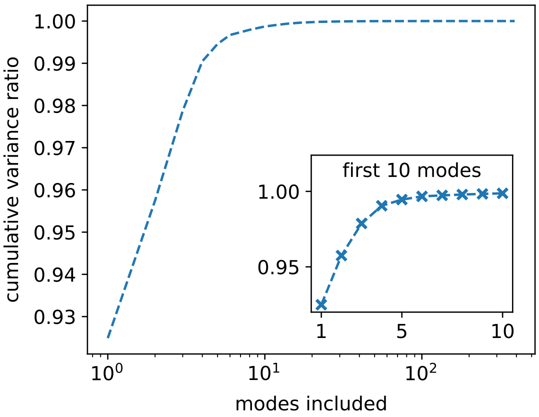


FIGURE 5.A.3: The cumulative explained variance ratio for an increasing amount of modes included of the pressure POD, starting with the modes with the highest singular values. POD obtained with linearly spaced pressure probes.

The input to the neural network which predicts the surface flow type consists of the first two temporal modes, corresponding to the first two spatial modes with the highest singular values. Similarly as in fig. 5.34 these temporal modes can be visualised for the POD obtained with linearly spaced probes, this is shown in fig. 5.A.4. Small differences between the non-uniform subsampled probe POD and the linearly subsampled probe POD exist, but the overall latent space representation of the data is very similar. This is to be expected following the similarity of the spatial modes and the cumulative variance distribution, which is related to the singular values.

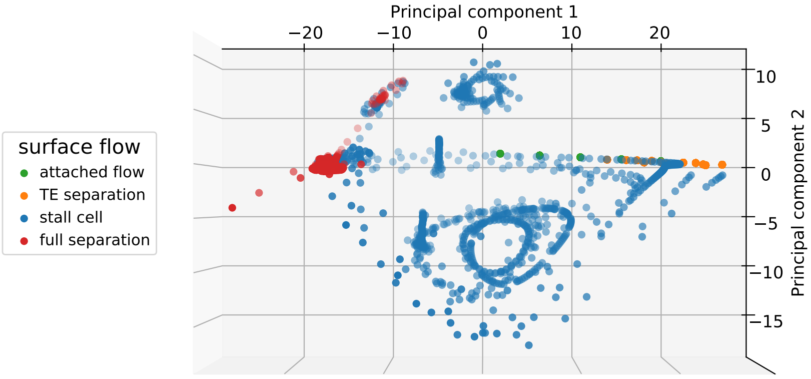


FIGURE 5.A.4: A 2D plot showing the relation between the POD coefficients of the first two POD modes obtained with linearly spaced pressure probes.

Lastly the neural network for surface flow type prediction as discussed earlier can be recreated, where everything is kept the same except the input is changed to the temporal modes obtained from the linearly spaced pressure probes POD. The confusion matrix of the trained result is shown in fig. 5.A.5. The results are very similar to the results obtained with the non-uniform spacing for the pressure probes. Both neural networks completely correctly classify all cases of full separation, trailing edge separation and attached flow. Both neural networks also have a very low error for the stall cell classification, where less than 2% of the cases are miss-classified.

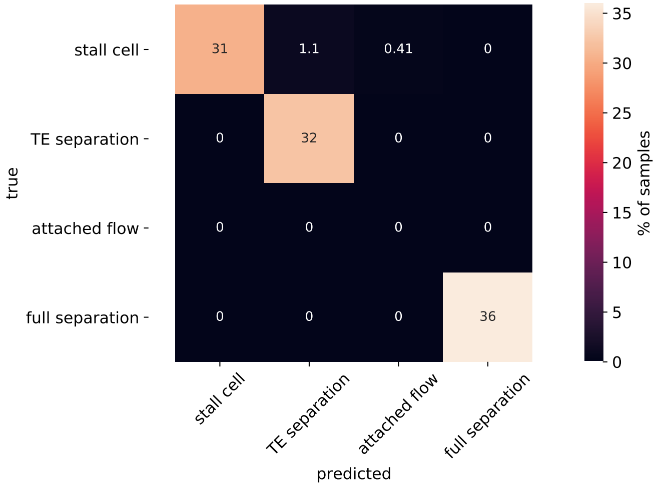


FIGURE 5.A.5: Confusion matrix showing the results on the test data set for the classification of the flow type with two POD coefficients.

With this analysis it has been shown that the current approach to predicting the surface flow type can provide consistent results with different probe positions. This mainly relies on the first two modes of the POD to be similar to the ones shown in fig. 5.29 or fig. 5.A.2 and capture a significant amount of variation in the pressure probe data set. While the first mode was shown to be indicative of changes in angle of attack and Reynolds number, the second mode includes spanwise variation in the pressure coefficient. The combination of these two aspects likely provides the necessary information for a neural network to determine a surface flow type prediction.

Chapter 6

Determination of Unsteady Wing Loading Using Tuft Visualisation

Unsteady separated flow is present on many physically large objects. Low-fidelity simulations cannot accurately estimate the influence of separated flow. The physical scale of some objects precludes full-scale testing in wind-tunnels and complicates the acquisition of performance data in operational conditions. Small-scale wind-tunnel tests can provide an estimate of the aerodynamic loading. Tufts can be used to qualitatively validate the surface flow between different investigations and the full-scale object in operational conditions. In the current investigation it is shown, with computational and experimental data, that tufts can provide a quantitative estimate of the unsteady wing loading. Unsteady 3D simulations of a NACA 0012 wing, at different angles of attack and Reynolds numbers, are used to obtain unsteady surface flow and lift coefficient data beyond stall. The computational data provides a proof-of-concept by using a linear surrogate model based on pseudo tuft orientations. This model is then extended with a non-linear component. Experimental data of a NACA 0012 wing, equipped with tufts and a force balance, has been used in combination with neural networks to infer quantitative information about the unsteady wing loading. This results in the ability to capture non-periodic lift and pitching moment fluctuations based on visual tuft observations.

6.1 Introduction

During the design phase of many aerodynamic applications both simulations and physical scale models for wind-tunnels are commonly used. Computational Fluid Dynamics (CFD) simulations can be utilised to obtain an estimate for almost any desired flow property in the simulated domain. When separated flow is present in the computational domain simulations tend to become more computationally expensive to achieve more accurate results, this can result in unrepresentative results when expensive simulations are not feasible (Ashton and Revell (2015)). Experimental investigations offer the ability to measure the flow, including separated flow, while limited by the capabilities of the experimental method and equipment. Surface flow visualisation is typically used to compare and validate simulations with experiments such as shown by Neves et al. (2020) and Cai et al. (2018). For unsteady flows the use of surface oil flow is not preferable due to the inability to provide temporal information. Tufts can be utilised for unsteady surface-flow to approximate the instantaneous local flow direction near the object surface. However tufts also have drawbacks. Each type of tufts has its own drag, inertia and stiffness which affect their specific response to the flow. Chen et al. (2019) indicated that the specific application method of tufts on a surface can also influence their ability to indicate the local flow direction. These and other shortcomings have largely precluded the use of tufts to obtain quantitative data. Some investigations have shown however that tufts can be used to obtain quantitative flow field measurements in a variety of manners for different purposes. Chen et al. (2020) used tufts for the identification of separated flow behind a backward-facing step where also the frequencies of the tuft fluctuations are considered. Similarly tufts have been used in a study by Swytink-Binnema and Johnson (2016) to characterise the flow on wind turbine blades on full scale. This study showed the ability of tufts to be combined with a video camera and video processing to achieve a reliable quantitative estimates of the stall fraction. Steinfurth et al. (2020) obtained quantitative information from tufts by using them to obtain velocity fields or identify pressure fluctuation based on the deflection of tufts. This investigation also indicated some issues associated with tufts in flow regions with a low velocity or with high frequency flow oscillations.

In the current investigation the focus is on unsteady surface flow over a wing with a NACA 0012 airfoil profile and the corresponding wing loading. For wings with airfoil profiles which exhibit trailing edge stall characteristics there is a tendency to develop coherent surface flow patterns after stall as discussed by Broeren and Bragg (2001). These characteristic flow patterns were observed by Moss and Murdin (1968). The surface flow patterns typically consist of a curved separation line along the spanwise direction which is terminated at both ends by two vortices on the wing surface, this flow structure is often referred to as a stall cell. An example of a stall cell flow configuration is shown in fig. 6.1.1, which is based on findings by Manolesos et al. (2014a,b) and Yon and Katz (1998). The investigations by De Voogt and Ganapathisubramani (2022), Dell'Orso and Amitay (2018) and Schewe (2001) indicate that the criteria for the formation of stall cells is dependent on the airfoil shape, the angle of attack and the Reynolds number. The exact formation mechanism for stall cells has not been shown yet, but has been reasoned to be related to the Crow-like instability in combination with a spanwise disturbance on the two shear layers due to the separated flow as described by Crow (1970) and Weihs and Katz (1983). The influence of a stall cell on the wing loading has been shown by Yon and Katz (1998) to be similar to a locally decreased effective angle of attack near the centre of a stall cell, where the flow near the surface has an upstream direction. An investigation by Liu and Nishino (2018) has shown that the presence of stall cells on a wing influences the performance significantly. In the research by Bartl et al. (2018) time averaged measurements of a wing with the same airfoil profile produced different results for the lift coefficient after stall, which has been attributed to multiple factors among which are stall cells. Similarly the unsteady behavior of stall cells has been shown by Liu and Nishino (2018) through CFD simulations to be highly related to the fluctuations in the lift coefficient of the wing. Stall cells can form on the surface of many aerodynamical applications, yet the implications of stall cells on the aerodynamic loading are unknown. While stall cells are identifiable through the surface flow, many other local stall patterns can also occur with characteristic surface flow and aerodynamic performance influence. The current investigation uses wings with a NACA 0012 airfoil profile, which has been extensively investigated and documented for several decades. Still, the stall behavior of this airfoil profile is not well known due to stall cells and their influence. In the current investigation it is shown that a data driven method can be used to approximate the relation between the surface flow and the unsteady aerodynamic loading through the use of tufts. The angle of attack and Reynolds number experienced by a wing can determine the aerodynamic performance regime such as attached flow or separated flow, for a simple surrogate model these input parameters are both steady and do not easily allow for an estimate of the unsteady aerodynamic loading associated with separated flow. The use of tufts as an input for a surrogate model in this investigation serves to have an unsteady input which results in a related unsteady output. Both a CFD and an experimental investigation have been conducted to assess the ability of tufts to provide an estimate for the unsteady wing loading. The computational data provides clean data which has been used to construct an interpretable proof-of-concept method which relies on the use of pseudo tufts obtained from the flow velocity near the wing surface. The experimental data represents a more complex real life situation as can be encountered in industrial applications with the use of physical tufts. The CFD investigation focused on a wide variety of cases, each with a different combination of angle of attack (α) and Reynolds number (Re), in order to estimate the lift coefficient fluctuations. The experimental investigation focused on a single combination of angle of attack and Reynolds number in order to estimate the lift and pitching moment fluctuations for highly unsteady surface flow. The use of tufts for estimating the unsteady wing loading can serve as a method for further research into unsteady surface flow or as an input to unmanned aerial vehicle control systems for example.

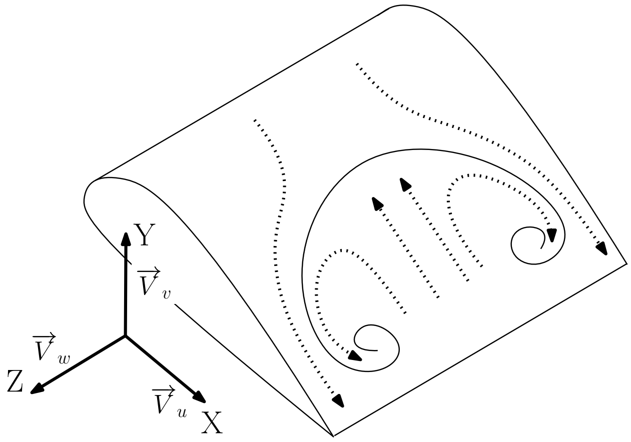


FIGURE 6.1.1: Flow direction (dotted lines) near the wing surface inside and outside of a stall cell. The x direction is aligned with the chord and the z direction is aligned with the span. The velocity vector $\overrightarrow{V}$ has its components [u,v,w] for directions [x,y,z] respectively.

6.2 Setup

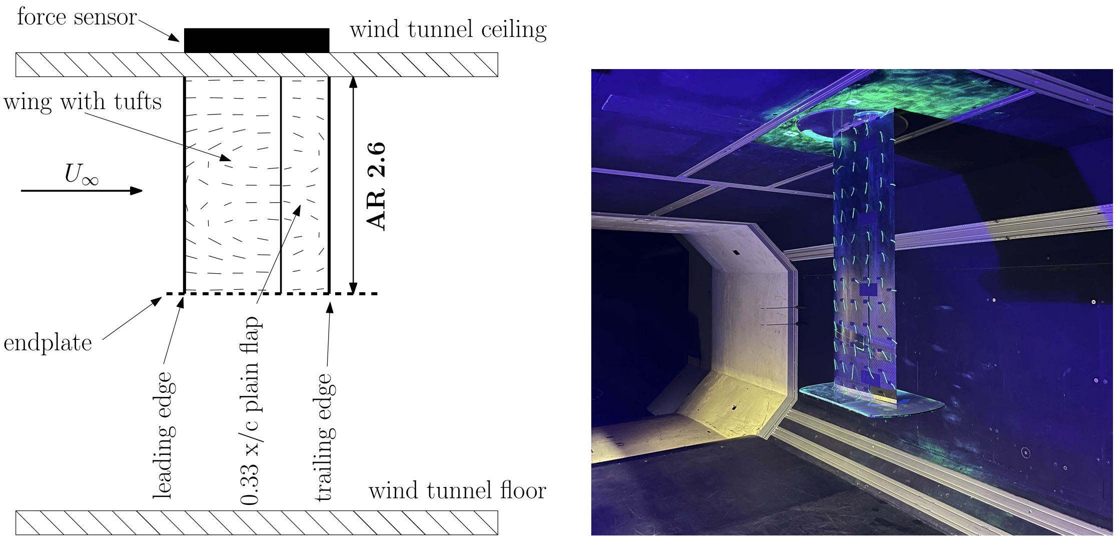
The results from the simulations described in section 2.1.2 are used for the current investigation. Additional relevant details are given in section 6.2.1. In section 6.2.2 the details of the experimental setup of a wing in the wind-tunnel are described.

6.2.1 Computations

Large scale features such as stall cells can be identified with tufts experimentally. In order to replicate this computationally, the surface flow velocity of the computational results have been used to obtain pseudo tufts. The local flow angle near the suction surface of the wing can be used to represent pseudo tufts. In order to obtain the local flow angle at desired locations near the wing surface an offset of 1 mm (0.001 x/c) has been used from the wing surface. Pseudo tufts have been obtained by linearly sampling 10 points in chordwise direction with a 0.05 x/c offset from both leading and trailing edge, the spanwise direction has been sampled at 0.1 z/c to match the original mesh grid spacing, as seen in fig. 6.4.3. The pseudo tuft and lift coefficient data obtained from the URANS simulations serves as the input and validation for the proof-of-concept method to obtain unsteady wing loading from tuft observations.

6.2.2 Experiments

The experiments have been conducted in a wind-tunnel at the University of Southampton, with a test section of 2.1 m by 1.6 m and 0.02 m/s accuracy for the set velocity. The free stream turbulence intensity has been measured to be approximately 1% near the ceiling of the tunnel at the location of the wing leading edge, and approximately 0.5% near the centre of the wind-tunnel at the location of the wing leading edge at the endplate. For the experimental setup a wing with a NACA 0012 profile, AR 2.6 and chord length of 30 cm has been used. This wing is suspended from a force balance (the model is an ATI Industrial Automation Delta IP65) outside the wind-tunnel test section at the wind-tunnel ceiling and spans approximately half the cross-section of the wind-tunnel, as shown by the schematic in fig. 6.2.1a. In order to limit the tip vortex effect an end plate has been used as shown in fig. 6.2.1b. While the endplate is only partially effective at eliminating the tip vortex, the remaining effect of the tip vortex has no influence on the analysis presented in this investigation. The forces and moments have been sampled at 6 kHz. The wing has been equipped with fluorescent tufts. The tuft spacing is based on the smallest scale flow features that are of interest. The tufts are 5 cm long and less than 1 mm thick. In chordwise direction 5 tufts have been used and in the spanwise direction 13 rows of tufts have been applied. In the analysis of the experimental data the two outer rows of tufts (near the wind-tunnel ceiling and near the endplate) have been disregarded, leading to 11 rows of 5 tufts to be analysed. The wing was illuminated with a UV floodlight, while the tufts were recorded at 120 frames per second simultaneously with the force measurements. The wing was tested at an angle of attack of 13° and a Reynolds number of $2.1 \cdot 10^5$ (10 m/s). This $\alpha-$ Re combination resulted in highly unsteady separated flow over the wing suction surface.



(A)

(B)

FIGURE 6.2.1: The experimental setup of the NACA 0012 wing equipped with tufts and a force balance. (a) A schematic of the setup. (b) The NACA 0012 wing with fluorescent tufts and an end plate installed in the 7′x 5′ wind-tunnel test section at the University of Southampton.

6.3 Methodology

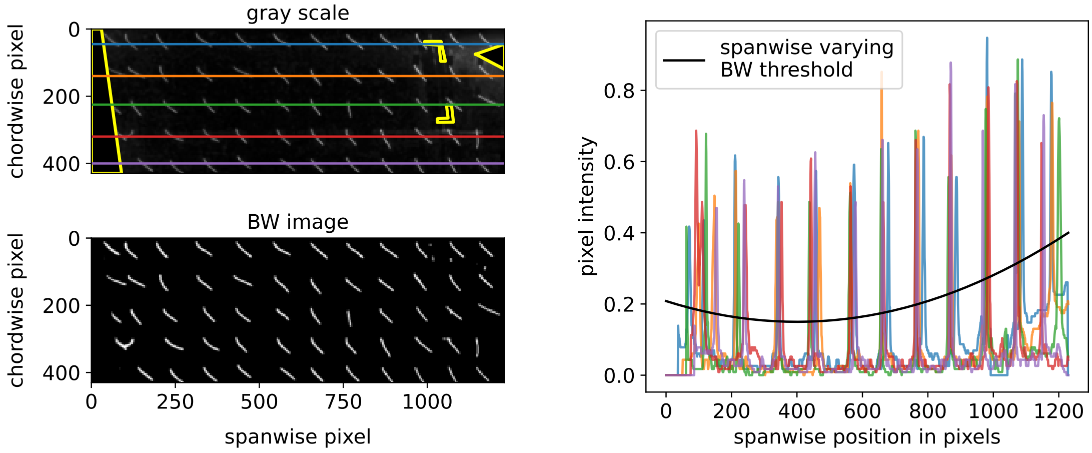
First the relevant data processing for the experimental data is presented in section 6.3.1. For the experimental data several processing steps have been described which transform the raw acquired images and measured lift and moment data into the data which has been used with data driven models. In section 6.3.2 the different approaches for data driven surrogate models which have been investigated are described.

6.3.1 Experimental data processing

The tufts provide an estimation of the local flow angle near the surface of the wing. In order to overcome the non-physical discontinuity at the zero to 2π radians boundary, each angle is converted to a sine and cosine value. Using the sine and cosine values allows two continuous values to represent any angle, where the numerical discontinuity when exceeding 2π is negated, which is beneficial for data driven methods. Additionally the cosine and sine values are individually representative for chordwise and spanwise flow directions respectively, when zero radians is aligned with the freestream direction.

6.3.1.1 Experimental tuft images

For the experimental investigation the purpose is to extract the local surface flow angles as indicated by the tufts. The procedure for doing so is dependent on the setup that has been used in combination with a wide variety of possible image processing methods, which can achieve a similar end result. In the current investigation we describe the approach that has been used for the current setup, this approach is also more generally applicable to different experimental setups. As the image processing is not the focus of the investigation, the experimental setup has been constructed to keep the required image processing to a minimum. The goal is to assign a single angle to each tuft. Curved tufts may occur due to aerodynamic forces, inertial forces or other influences. The curvature radius of the tufts usually is very large compared to the tuft length. Additionally, the tuft spacing is small enough such that a single tuft is not essential to identifying a surface flow pattern. Therefore, the curvature is ignored and tufts are assumed to be straight lines. It is possible with more complicated image processing algorithms to obtain a more accurate representation of tufts with segmented lines.



(A)

(B)

FIGURE 6.3.1: Image processing of the tuft recording. (a) Top: a single grey scale image is shown on the top, clipped to wing size, with masked areas indicated by a yellow circumference. The 5 horizontal lines crossing the tufts in spanwise direction indicate the location of the analysed the pixel intensity in part (b). Bottom: the resulting black and white image by using a spanwise varying threshold and masking. (b) The intensity of the pixels crossed in a grey scale image of the tufts by the spanwise lines as shown in the top image of part (a) of the figure.

The image processing initially serves to construct a black and white image which only contains the tufts. Irrelevant areas from the images and small areas with consistently high intensity reflections have been masked, shown in the top image of fig. 6.3.1a. The UV light intensity varies along the span of the wing, causing a varying background intensity along the span of the grey scale image. In order to eliminate the background noise of the grey scale image a black and white threshold is used which varies along the span such that it is higher than the background intensity but lower than the tuft intensity at each spanwise location as shown in fig. 6.3.1b.

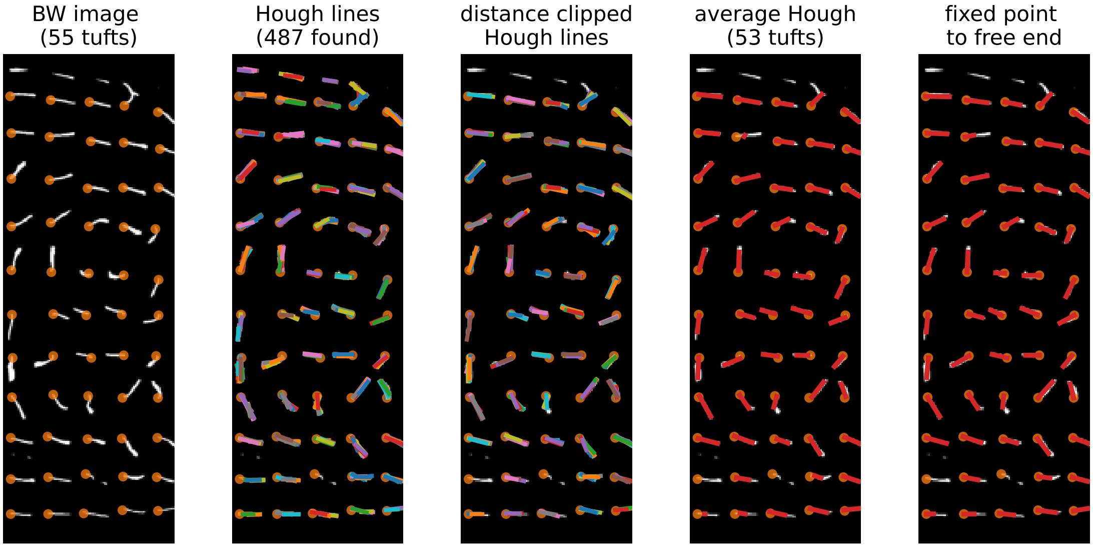


FIGURE 6.3.2: Processing of the black and white images to obtain the tufts from each snapshot image. From left to right: identification of the fixed end of the tufts (orange dots), all Hough lines found, delete lines which are far away from the tuft fixed end points, average the begin and end points of the Hough lines grouped near a tuft fixed point and as a final step the starting point of the Hough lines is replaced with the known fixed location of the tufts.

The black and white images containing only tufts can be used in combination with a probabilistic Hough transform to obtain straight lines corresponding to the tufts, as is shown in fig. 6.3.2 (Galamhos et al. (1999)). The Hough transform is implemented in several Python packages and MATLAB, although the exact implementation might vary. The Hough transform allows to set certain parameters such as the line length which should match the tuft length in the image or be slightly shorter. The initially found lines can be many more lines than tufts that are in the image and can overlap. Further processing of the initially found Hough lines is based on the known location of the tuft fixed ends on the wing. These locations can be marked on the image and used to eliminate Hough lines which are located too far away from any such known tuft fixed end location. Next the remaining Hough lines can be grouped together per tuft location to obtain a single Hough line per tuft location, by averaging the start and end points. In a final step the end point of the Hough line closest to the known tuft fixed end location is replaced by the known fixed end location. As is observable from fig. 6.3.2 the last step of the process has almost no impact on the tuft identification when the image processing is suitable for the data under consideration. Using this method, for 94% of the image samples 52 or more tufts have been found out of the 55 available tufts. From these identified tufts the local surface flow angle can be obtained and converted into the corresponding sine and cosine values. Each sample thus has 55 values for the sine and 55 for the cosine (110 total values per snapshot image), missing tufts have a value of zero for both the sine and cosine.

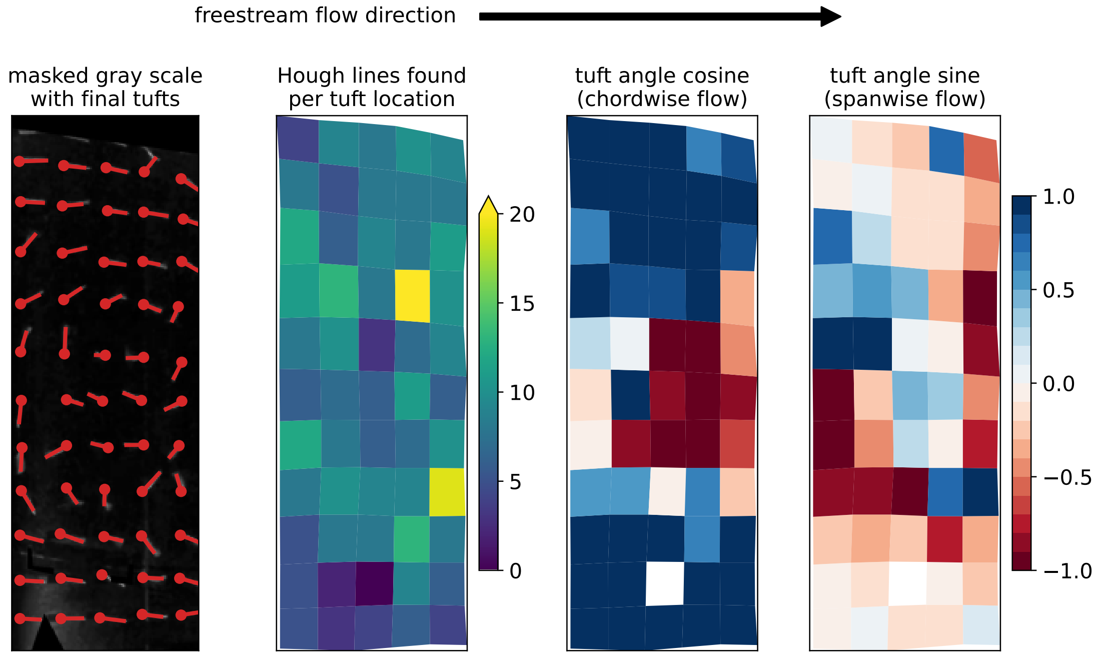


FIGURE 6.3.3: The results from the image processing, from left to right: tufts found by the processing, tracking of the Hough process, map of the cosine of the tufts and a map of the sine of the tufts. In the middle of the second tuft row from the bottom a missing tuft can be observed.

In fig. 6.3.3 the result of the image processing is shown. In the left most figure the grey scale image is shown with the masked areas blacked out. The tufts as found by the image processing are shown on top in red. The second image from the left shows the amount of Hough lines found per tuft location and can be used to assess if the parameters of the probabilistic Hough line process are appropriate for the data from the current setup. The last two images on the right illustrate the sine and cosine of the tufts that have been found for the shown sample. The sine and cosine have been obtained from the tuft angles calculated in the image frame of reference. While the calculated angles do not represent the effective flow angles near the surface relative to the surface (and chord), the difference between the image frame of reference and the local frame of reference on the wing surface is small with respect to the tuft angles. As the relative position of the camera and the wing does not change during the data acquisition, the choice of frame of reference is irrelevant. The neural network relies on the relative positioning of the tufts to each other and the freestream flow. As the angle of attack does not change, the inflow angle is automatically included in the data as a constant. For the current case it is thus not necessary to apply coordinate transformations. It would be beneficial to apply coordinate transformations of the tuft angles into a consistent frame of reference when using different angles of attack, or different viewing angles. For the application of a trained model to different objects the objects need to be geometrically similar, and the flow conditions need to be similar. Finally, the tuft positions need to be in the same frame of reference as the original data used for training the network.

This can be achieved by similar camera positioning and field of view relative to the object. Alternatively a coordinate system transformation can be applied to the new data to match the frame of reference of the training data.

6.3.1.2 Experimental force and moment measurements

The experimental data obtained with the force balance has a sampling frequency of 6 kHz, while the camera used for recording the tufts used a frame rate of 120 Hz. Consecutive images of the tufts have been verified to have only very small differences in tuft orientations. It can be checked with the force and moment data if the frame rate of the tuft recordings is sufficient to capture the unsteady aerodynamic loading. This does not guarantee that the tufts are suitable for the investigated flow as they may react insufficiently to changes in the surface flow. A frequency analysis of the lift and pitching moment signal at 6 kHz is shown in fig. 6.3.4. In order to obtain the frequency spectra 25 windows of 28.2 seconds have been used (total signal length is 705 seconds), with Hanning weighting coefficients and no overlap (Bartlett's method) (Nuttall (1981), Engelberg (2008)). The large window size allows to analyse also large timescale changes in the surface flow. The lowest frequency of 0.035 Hz corresponds to 940 convective

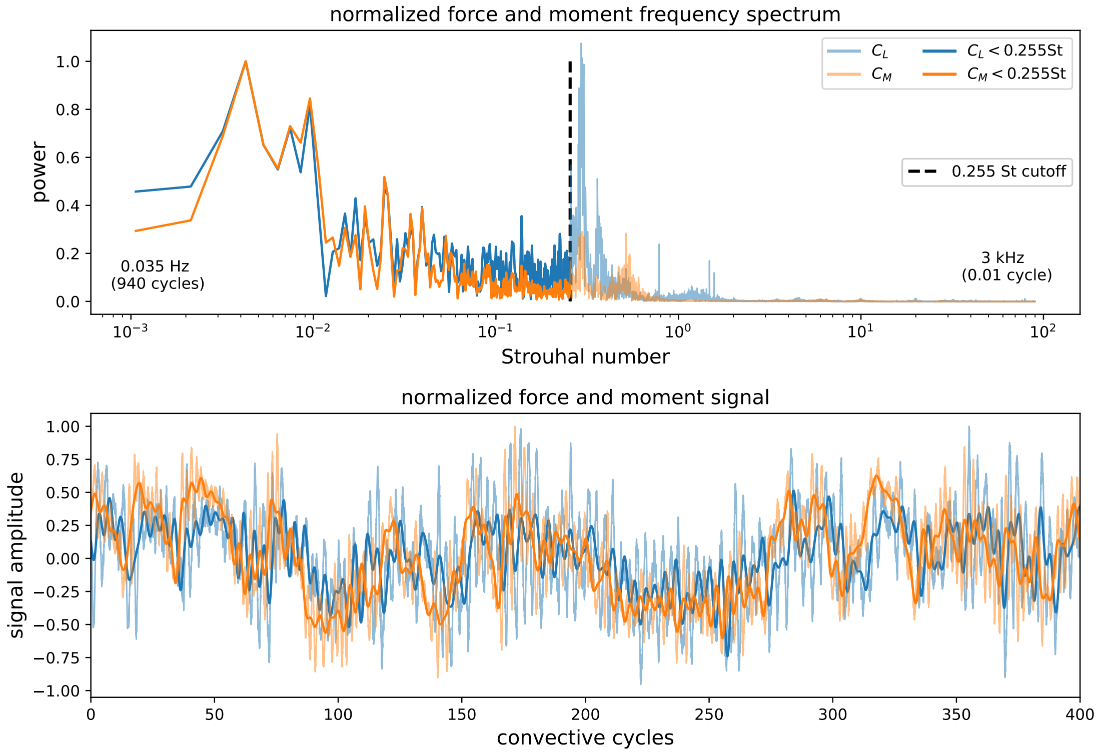


FIGURE 6.3.4: Top: the frequency spectrum of the lift and pitching moment signal, normalised by the peak value around 0.03 St. Bottom: a sample of the signal acquired for the lift and pitching moment fluctuations, normalised by the highest absolute value per signal in the sample range. Convective cycles are defined as the chord length divided by the freestream velocity.

cycles or a Strouhal number of 0.001 approximately. The highest frequency of 3 kHz corresponds to 0.01 convective cycle or a Strouhal number of approximately 90.

The frequency spectra show significant peaks at very low Strouhal numbers ranging between 0.03 and 0.1, showing the longer timescales which are relevant. These timescales represent approximately 100 - 300 convective cycles and are driven by changes in the flow over the wing. At a Strouhal number of approximately 0.3 another significant peak in the lift and moment fluctuations frequency spectra can be observed in fig. 6.3.4. It is expected that this frequency of force and moment oscillations represents the natural frequency of the suspended wing structure. This can be further verified in the future. The frequency spectra of several tuft locations has also been checked (both sine and cosine, representing spanwise and chordwise fluctuations respectively), which showed that the specific tuft location has a large influence on the frequency behavior of the tuft motions. More details are given in section 6.4.4.3. To illustrate the lift and moment signal amplitude variation without the large amplitude of the natural frequency component, also a version of the signal with all frequencies above a Strouhal number of 0.255 cut is shown in fig. 6.3.4. The signal with a 0.255 St cutoff is not used for the training of any model, but has been used for interpretation of the results. The original lift and moment signals do not contain significant frequencies above a Strouhal number of 1.8 or 60 Hz (which is the tuft recording Nyquist frequency). For the data driven model the original force and moment measurements have been down-sampled to 120 Hz from 6 kHz, such that a corresponding lift force and pitching moment are available for each snapshot image of the tufts.

6.3.2 Lift coefficient from tufts

The angle of attack and the Reynolds number are the primary parameters to determine an estimate of the steady lift coefficient of a given wing. When using for example experimental data it is common practice to take the average result from multiple instantaneous measurements at a fixed angle of attack and Reynolds number to obtain a representative steady lift coefficient result. This approach is shown by:

$$\overline{C_L}(\alpha, \mathrm{Re}) = \frac{\sum_{i=1}^{n} C_{L,i}(\alpha, \mathrm{Re})}{n} \Bigg|_{\alpha, \mathrm{Re}\, =\, \text{constant}}, \tag{6.1}$$

with n measurements for a given $\alpha - \mathrm{Re}$ combination, where the over line indicates the average for the lift coefficient. However, unsteady flow behavior results in an unsteady lift coefficient. As such, the constant parameters α and Re do not provide the necessary inputs for a simple surrogate model to produce an unsteady lift coefficient. Tufts can be used to capture the unsteady flow behavior and thereby allow for an estimation of the unsteady lift coefficient through the use of a surrogate model. The unsteady component of the lift coefficient can be represented as a deviation from the steady lift coefficient:

$$C_L = \overline{C_L}(\alpha, \mathrm{Re}) + \mathcal{F}(\mathrm{tufts}), \tag{6.2}$$

the linear combination of the steady and the unsteady lift coefficient components is shown, with the unsteady component represented as a function of the tufts.

6.3.2.1 POD and universal approximation theorem

The unsteady component of the lift coefficient, as estimated through the use of tufts, can be determined by a linear and a non-linear component:

$$\mathcal{F}(\mathrm{tufts}) = \mathcal{F}_{\mathrm{linear}}(\mathrm{tufts}) + \mathcal{F}_{\mathrm{non\text{-}linear}}(\mathrm{tufts}), \tag{6.3}$$

the linear component of the data driven model is constructed out of a linear combination of a truncated set of temporal modes of the tuft SVD (equivalent to POD coefficients), as shown by eq. (6.4) for a single sample i. A single layer neural network with a linear activation function can be used:

$$\mathcal{F}_{\mathrm{linear,i}}(\mathrm{tufts}) = \mathrm{W_L} \cdot \Psi_i + \mathrm{B_L}, \tag{6.4}$$

in eq. (6.4) the trained weights are stored in vector $\mathrm{W_L}$, the temporal modes for a set of spatial modes of a single sample are represented by vector Ψ_i, the weights and temporal modes are combined with the dot product and finally a trained bias $\mathrm{B_L}$ is added. The non-linear component of the data driven model is based on the universal approximation theorem (Cybenko (1989)). The non-linear component can be approximated using a neural network with a single hidden layer and a Sigmoid activation function, followed by a linear output node. This approach can be described for a single sample i with n nodes in the hidden layer:

$$\mathcal{F}_{\mathrm{non\text{-}linear,i}}(\mathrm{tufts}) = \sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{a_j} \cdot \sigma(\mathrm{W_{NN,j}} \cdot \mathrm{X_i} + \mathrm{B_{NN,j}}), \tag{6.5}$$

in eq. (6.5) the angles of the tufts are used as inputs rather than the SVD temporal modes. The tuft angles for a single snapshot sample are represented by vector $\mathrm{X_i}$. The tuft angles are multiplied with the weights of node j indicated by vector $\mathrm{W_{NN,j}}$ using a dot product. The bias of node j represented by $\mathrm{B_{NN,j}}$ is added. The output of each node has a Sigmoid function applied ($\sigma()$). The final outputs of all nodes are then linearly combined with coefficients $\mathrm{a_j}$ without a bias.

6.3.2.2 Neural network

The experimental data contains some noise and other deficiencies such as tufts that might get stuck and are no longer aligned with the surface flow. The tufts have initially been applied aligned with the attached flow direction. When upstream flow is present the tufts are forced in the opposite direction of their resting position, this causes further issues with determining the exact local flow angles. In order to deal with these imperfections, which are to be expected, it has been chosen to utilise a simple neural network which more easily can be trained on a large volume of data while certain shortcomings present in the data. The use of a neural network allows to illustrate the current approach of using tufts to obtain aerodynamic performance coefficients in a manner which can more easily be applied to real world data.

The previously discussed linear method based on the SVD temporal modes uses merely 31 (30 coefficients and one bias) trainable parameters, when using 30 modes as an input. The extension with the non-linear approach based on the universal approximation theorem has 18270 trainable parameters for 35 nodes in the hidden layer and 520 (sine and cosine) inputs from the tufts. The computational training data contains a high number of input parameters for the amount of training data (4.6k samples) available, which is sub optimal for data driven methods. The experimental data contains significantly less input parameters (110 sine and cosine values) for a larger amount of available training data (65k samples). This further promotes the use of a neural network where the increased complexity requires more training data than the linear or single layer equivalent data driven method.

A neural network with two hidden dense layers has been used. The first has 75 nodes and the second hidden dense layer has 10 nodes. Both hidden layers use a Sigmoid activation function. A ReLU activation was tested, which converged faster. The Sigmoid activation consistently resulted in a lower error. Two dropout layers are used, one between the input and the first hidden dense layer and a second dropout layer is used between the first and the second hidden dense layers. The full network architecture is shown in fig. 6.3.5. The network architecture with 110 inputs results in a total of 9096 trainable parameters. The inputs to the network are the sine and cosine of the tufts on the wing surface and the output is the magnitude of the instantaneous lift fluctuation. A network with the same architecture can be constructed but trained with the pitching moment as an output. The L2 norm between the known output values of the training set and the model estimation has been minimised in combination with the ADAM optimiser (Kingma and Ba (2015)).

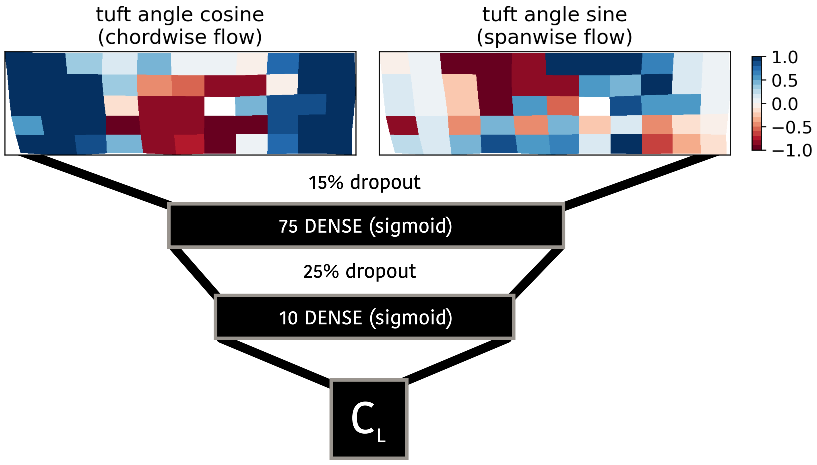


FIGURE 6.3.5: A two layer neural network architecture for determining the lift or pitching moment fluctuations from 55 tufts.

6.4 Results and discussion

The results consider initially the output of the URANS CFD simulations presented in section 6.4.1. The output of the simulations considers the surface flow and the lift coefficient experienced by the wing over time for a variety of angle of attack and Reynolds number combinations. From the computational simulations the surface flow velocity can be obtained on the entire wing suction surface, this surface flow velocity also provides the flow direction for the pseudo tufts. As such a clear comparison can be made between the detailed information from the flow velocity and the sparse information provided by the pseudo tufts. The POD analysis of the suction surface velocity obtained from the simulations is shown in section 6.4.2, followed by the POD analysis of the suction surface tufts of the simulated NACA 0012 wing in section 6.4.3. Finally the results for the computational and experimental investigation into obtaining an estimate for the unsteady wing loading from tufts are given in section 6.4.4.

6.4.1 Surface flow and wing loading of URANS computational results

The use of periodic boundary conditions allows to simulate an infinite wing. The investigation by Liu and Nishino (2018) presents in great detail the effect of the AR of the wing in combination with periodic boundary conditions. Investigations by De Voogt and Ganapathisubramani (2022), Moss and Murdin (1968), Gregory and O'Reilly (1970) and Manni et al. (2016) indicated that the NACA 0012 airfoil profile tends to produce stall cell patterns of approximately 2.5 c in spanwise width. The current investigation uses a wing with AR 2.5. By varying the angle of attack and Reynolds number over a wide range around and beyond stall, multiple flow configurations have been observed.

Most $\alpha - \text{Re}$ combinations that have been simulated contain some form of flow separation, which often results in unsteady flow behavior and by extension an unsteady lift coefficient for a constant $\alpha - \text{Re}$ combination. All 43 distinct $\alpha - \text{Re}$ combinations which have been simulated are shown in fig. 2.6.

For the highest Reynolds number under consideration, 10^6, the widest range of angles of attack has been simulated. This wide range for the angle of attack illustrates the four different primary flow configurations under consideration: attached flow, Trailing Edge (TE) separation, stall cells and full separation. The cases with attached flow typically would not require expensive 3D URANS simulations and a 2D RANS simulation could produce similar results with respect to the obtained converged lift coefficient. Furthermore it might be possible to simulate the cases with TE separation or full separation (along most of the chord length) with 2D URANS as some flow unsteadiness can occur which will still be mostly uniform along the span. Cases which contain the formation of stall cells do require 3D unsteady simulations as the flow is unsteady and not uniform along the span. In fig. 6.4.1 examples are shown for the lift coefficient result of two cases which contain stall cells at an angle of attack of 17° with two distinct Reynolds numbers. During the first seconds of the simulations it can be seen that the flow and thus lift coefficient are adjusting and the lift coefficient does not yet represent the characteristic oscillatory behavior which can be associated with unsteady stall cell flow as indicated by the investigation of Liu and Nishino (2018).

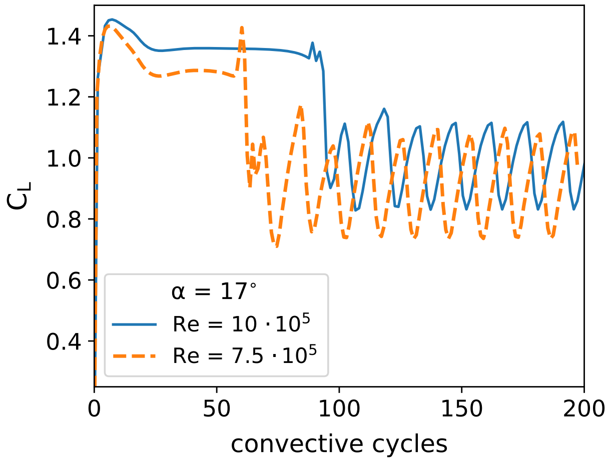


FIGURE 6.4.1: The lift coefficient as obtained with URANS simulations for two cases with angle of attack 17° and two different Reynolds numbers. Both cases exhibit stall cell surface flow behavior. Convective cycles are defined as the chord length divided by the freestream velocity.

From the current investigation of simulated $\alpha - \text{Re}$ combinations it can be observed that the angle of attack range in which stall cells occur varies across the Reynolds numbers tested. More specifically at higher Reynolds numbers the range of angles of attack in which stall cells occur can be seen to be larger and occur at higher angles of attack in fig. 2.6.

For both TE separation and full span separation the use of tufts can indicate the extent of upstream flow in chordwise direction on the wing suction surface, where the extent of upstream flow is mostly uniform along the wing span. For cases with stall cells present on the wing surface significant variation in flow direction near the wing surface along the span is induced by stall cell vortices, as shown by fig. 6.4.2. The surface flow can locally be observed to include also spanwise velocity components as opposed to almost no spanwise surface flow for cases with TE separation of fully separated flow.

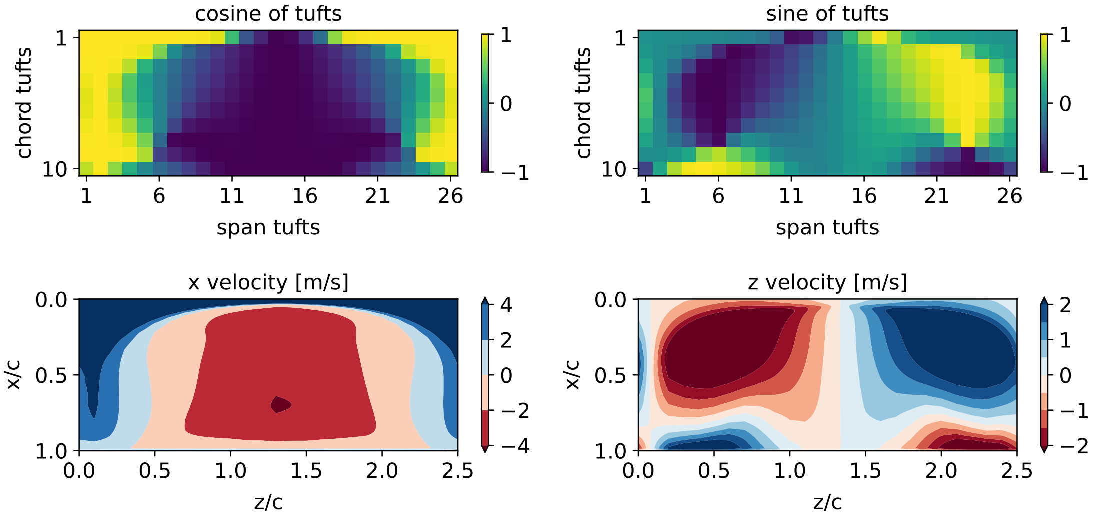


FIGURE 6.4.2: An example of a stall cell in the computational URANS results. The cosine and sine of the pseudo tufts which have been obtained from the surface flow velocity (0.001 x/c offset from the surface) on the suction side of the wing, with x/c = 0 the leading edge of the wing, for angle of attack 18° and Re = 10^6 (time = 8.5 s).

Cases with a stall cell present can also have significant variations in sectional lift coefficient along the span of the wing. In fig. 6.4.3 the variation in sectional lift coefficient can be observed. The pressure coefficient, from which the sectional (and integral wing) lift coefficient is obtained through integration, is related to the surface flow by both flow velocity and flow direction. Flow structures can be recognised from the pseudo tufts on either an integral wing scale (e.g. stall cells) or on a smaller scale (e.g. stall cell vortices).

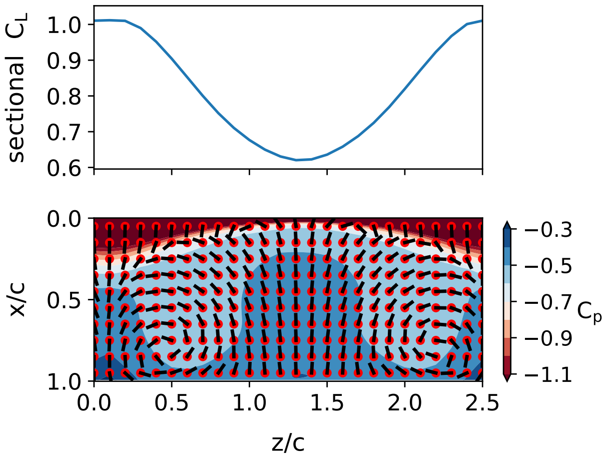


FIGURE 6.4.3: Top: sectional lift coefficient variation along the span as induced by a stall cell. Bottom: visualisation of pseudo tufts with pressure coefficient contours for angle of attack 18° and Re = 10^6 (time = 8.5 s).

For the evaluation of POD and other data driven methods, the available data obtained from simulations has been split into training data and test data. In order to assess the ability of the data driven methods to generalise to other α − Re combinations several cases have been selected to serve as test cases. These have been indicated with a black square in fig. 2.6. Among the test cases are varying angles of attack, Reynolds numbers and the three flow configurations with separated flow which are of interest for unsteady wing loading. The training and test sets both exclude the initial 25% of the simulated time to eliminate the majority of the initial transient phase of the simulation. The training data set contains 4631 snapshot samples representing 37 cases (unique α − Re combinations), while the test set contains 731 samples representing 6 cases. The training − test split is thus approximately 86 − 14 (%) respectively in terms of cases (and closely matched for sample count). For clarification with respect to the commonly used test-validation-train split in data driven methods, the test set referred to in this investigation is a data set which has no influence on the training of any method, with exception of eq. (6.1). For the other data driven aspects all data used for training is either a subset of the training data set (validation set, used for early stopping criteria of the training) or the complete training data set.

6.4.2 Surface velocity POD

The oscillations of stall cells have been linked to the oscillations of the lift coefficient by Liu and Nishino (2018). Liu and Nishino showed that the temporal modes of velocity flow fields in the wake contain a similar frequency content as the lift coefficient. In the current investigation POD can be applied to the suction surface flow velocity of the wing (sampled at a 0.001 x/c offset from the wing surface). The surface velocity has been sampled on a 26 by 204 grid, in spanwise and chordwise direction, respectively.

The surface velocity is described by both a chord- and spanwise velocity component in each grid point. A total of 10608 data points are thus available for each snapshot sample. As 4631 training samples are available for 10608 parameters, the resulting POD of the suction surface velocity results in 4631 spatial modes. The average of all snapshot samples contained in the training data set (excluding transient phase at the start of each simulated case), shown in fig. 6.4.4, has been subtracted from the samples prior to the computation of the POD basis. The mean surface velocity of the training set can be seen to include a chordwise velocity gradient which is mostly consistent along the span. The spanwise velocity component shows multiple dipoles which represent vortices.

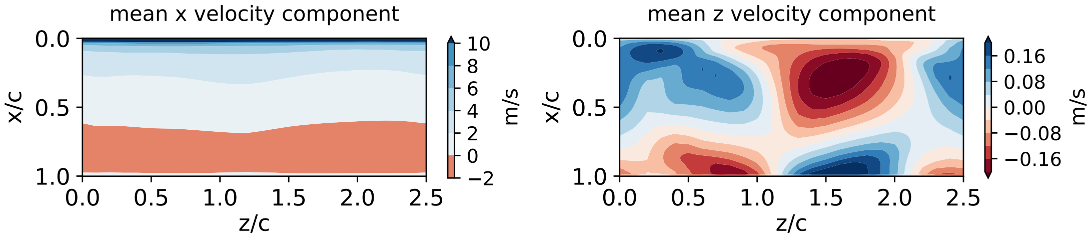


FIGURE 6.4.4: The mean velocity components of the training set. Data sampled at 0.001 x/c offset from the wing suction surface. The data set spans Reynolds numbers from $3.1 \cdot 10^5$ (4.53 m/s) to 10^6 (14.61 m/s).

In order to confirm that the training data is sufficiently large for the surface flow under investigation the cumulative explained variance ratio can be checked. This ratio represents the fraction of the variance in the original data set that can be reproduced with a rank truncated POD reconstruction with a set amount of modes. In fig. 6.4.5 this ratio is shown for an increasing rank which include a number of modes with the highest singular values. From the figure it can be seen that with 10 modes included more than 99% of the variance in the data set can be reconstructed. Several factors can contribute to the low number of modes required. One of those factors can be the similarity between samples such as periodically oscillating stall cells, or time invariant trailing edge separation. Another factor can be the chordwise distribution of the 204 measurement points, which are aligned with the computational grid and are thus closely clustered near the leading edge where the surface velocity is relatively high. Local flow velocity changes near the leading edge can thus lead to a large explained variance. Lastly, as URANS simulations have been used, a significant amount of averaging has been applied to obtain the solutions, which limits the presence of small scale features.

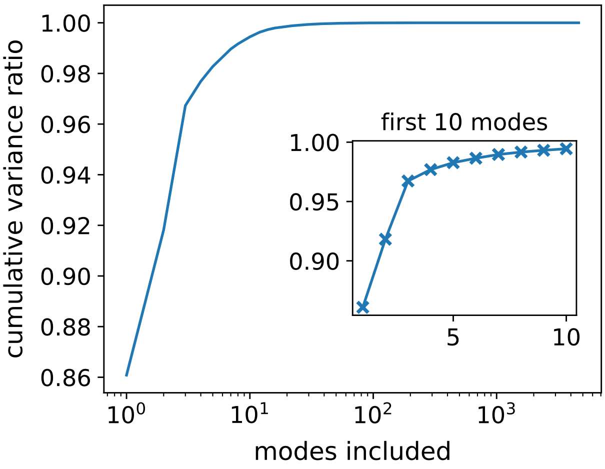


FIGURE 6.4.5: The cumulative variance ratio of the surface velocity POD.

The spatial modes of the velocity POD can be visually represented as flow patterns. These patterns can be linearly combined to reconstruct the snapshot samples of the training data set. The spatial modes are ranked according to their singular value scaling which allows to represent the relative importance of the modes. The three most important modes corresponding to the largest singular values are shown in fig. 6.4.6. The first mode can be seen to primarily affect the chordwise velocity. The velocity gradient over the suction surface with a high velocity near the leading edge which decreases towards the trailing edge is clearly represented. While the mean velocity components of the training data set also contain a significant gradient for the x velocity component, the first mode represents the variation in the magnitude of the x velocity component which can be caused by for example Reynolds number changes. The second mode can be observed to contain a structure similar to a stall cell split in the middle, with each half of a single stall cell attached to the periodic boundary. The spanwise velocity component for the second and third mode can be seen to exhibit dipole areas which are the result of the stall cell vortices. The third mode represents a regular stall cell at the centre of the wingspan. The second and third mode effectively illustrate the same surface flow pattern, which due to periodic boundary conditions, is shifted along the span. It is unsurprising that the periodic boundary conditions introduce spanwise periodicity in the data, and therefore also in the POD spatial modes. Over the wide range of cases simulated which include stall cell surface flow patterns, it has been found that the results will tend to represent either a full stall cell in the centre of the span or a stall cell split down the middle (i.e. shifted by half a span). The second and third mode thus represent a pair of modes which are of very similar importance. This is also shown in the inset of fig. 6.4.8 by the very similar singular values of the second and third mode. POD mode pairs can exist for periodic flow situations with a characteristic length scale to represent the alternating states in snap shot samples such as for a von Kármán vortex street (Mulleners et al. (2008)). The first three modes shown have a similar sign convention for upstream and downstream flow (and corresponding spanwise flow in either direction) to allow for a comparison. However when computing the POD library the signs of the modes can be arbitrary which is resolved in the decomposition of a sample with the appropriate sign for the temporal mode of the sample and corresponding spatial modes. More information with regards to the sign convention of POD modes can be found in Carter et al. (2021). Other POD spatial modes after the first three do not represent recognisable flow patterns such as presented in fig. 2.6. They rather represent a variety of differences between multiple flow configurations or unsteady flow configurations which only exist on small timescales as can be observed from the fourth and fifth mode in fig. 6.4.6.

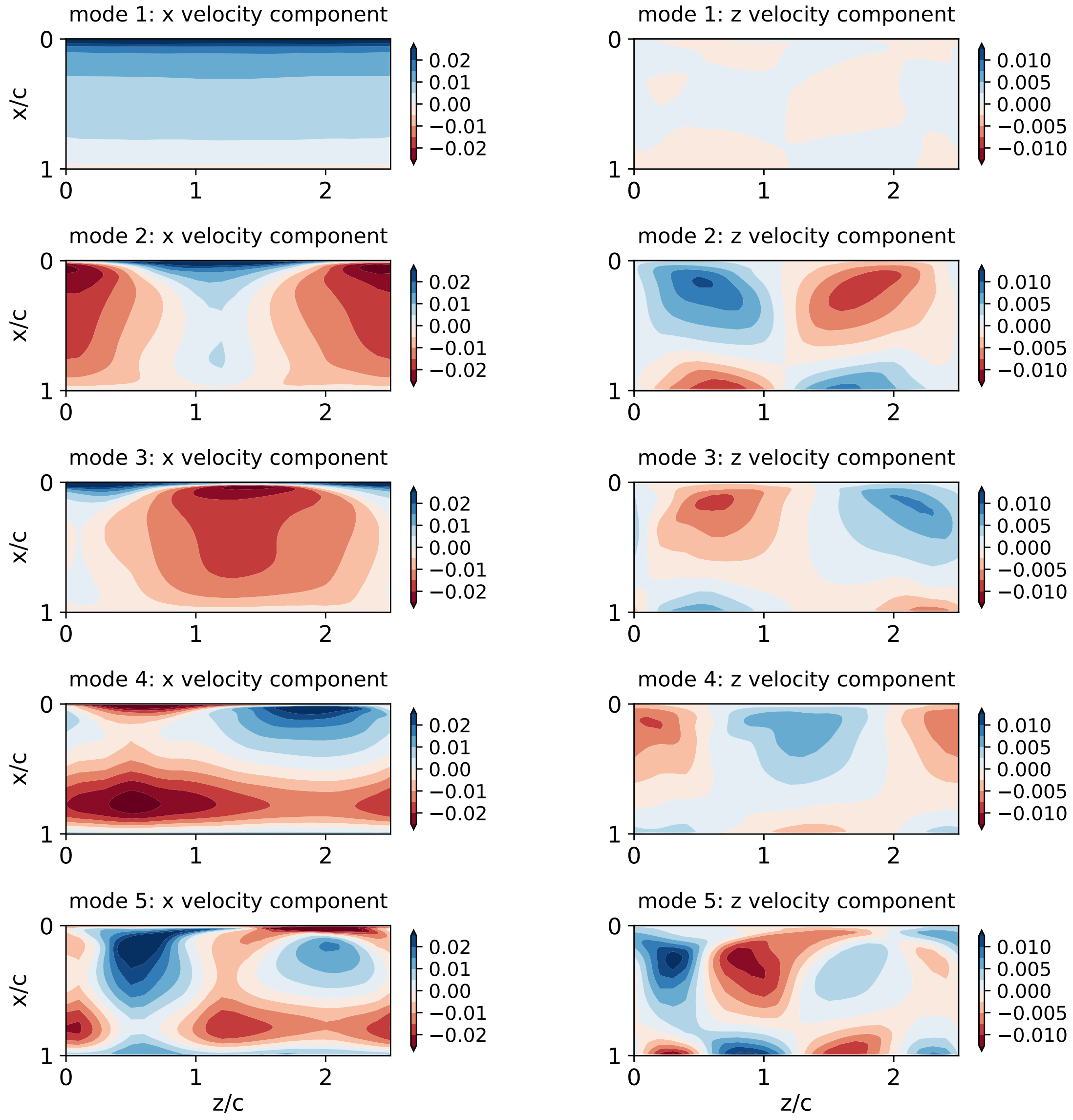


FIGURE 6.4.6: The chordwise (x) and spanwise (z) velocity components for the first five modes of the suction surface POD.

The relation between the POD modes, the flow configuration on the wing surface and the lift coefficient of the wing can be captured by the frequency spectrum of the temporal modes and the lift coefficient over the simulated time frame. In fig. 6.4.7 the lift coefficient is shown over time, where the initial transient phase (such as shown in fig. 6.4.1) has been eliminated from the analysis, together with the temporal modes (POD coefficients divided by the singular values) over time. The signals shown in the plot on the left have been normalised with the maximum of the absolute signal such that the plot allows for a comparison within the [-1,1] amplitude range, similarly the frequency spectra shown in the right plot have been normalised with their respective maximum peak value. From the pseudo tufts and the flow velocity visualisation on the wing surface it has been verified that the case with angle of attack 18° and a Reynolds number of 10^6 shows a full stall cell, as shown in fig. 6.4.3. Similarly from the temporal modes as shown in fig. 6.4.7 for the same case it can be seen that mode 1 and mode 3 retain a value close to their maximum which is positive. As mode 1 represents the general velocity gradient on the wing, this is likely present in each sample. However, mode 3 has been indicated to represent a full stall cell as shown in fig. 6.4.6 which is a specific flow configuration. Mode 2 and mode 3 of the velocity POD both show a stall cell configuration, but shifted by half a span length relative to each other. As such mode 2 and 3 have opposite rotational directions of the stall cell vortices at similar spanwise locations. This results in the temporal modes for the case shown in fig. 6.4.7 to have opposite signs for mode 2 and mode 3.

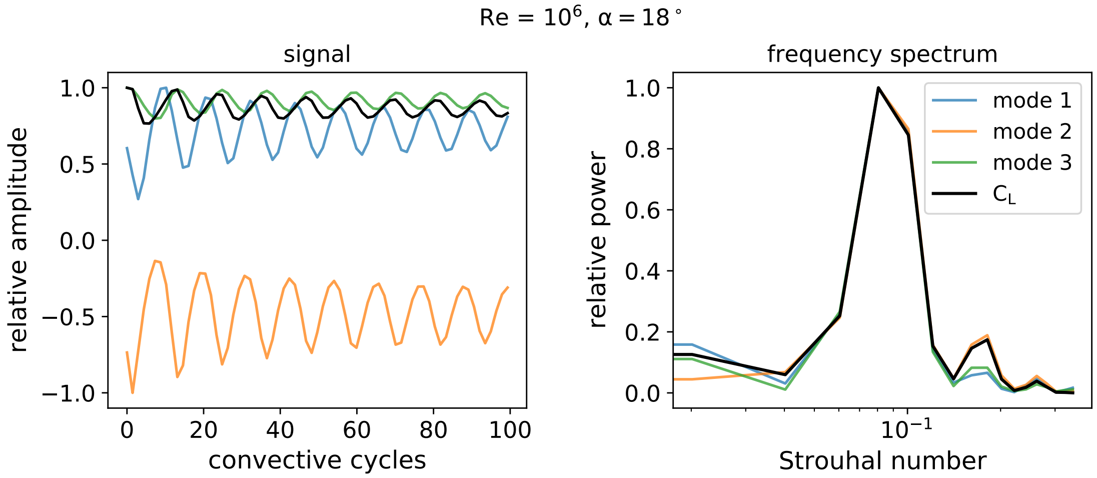


FIGURE 6.4.7: Surface flow velocity POD temporal modes frequency spectrum with lift coefficient frequency spectrum. Both signals and frequency spectra are normalised with their respective maximum of the absolute values. Left: the signal under investigation, right: the resulting frequency spectrum. Convective cycles are defined as the chord length divided by the freestream velocity.

From the frequency spectra in fig. 6.4.7 it is clear that the frequency spectrum of the SVD temporal modes matches closely with the lift coefficient frequency spectrum. Similar observations can be made for other cases with stall cells present, or cases with unsteady full span separation and oscillations in the lift coefficient. Each case may have its own characteristic frequency spectrum for the lift coefficient. The lift fluctuations follow from the unsteady flow around the wing. With POD these flow changes can be captured, which results in fluctuations of the temporal modes with similar frequencies as the lift coefficient fluctuations. The current analysis presents a clear indication that the most energetic spatial modes are able to capture unsteady flow fluctuations near the wing suction surface. Additionally the fluctuations of these spatial modes appear to be closely related to the lift coefficient fluctuations of the wing.

6.4.3 Tufts POD

The surface velocity as obtained from CFD simulations provides a lot of information in great detail as the mesh grid resolution is very fine in chordwise direction and the velocity magnitude and direction are both known. For experimental investigations obtaining the surface velocity is highly complicated due to the curved surface of the wing. Currently no experimental method can produce similar results as obtained from CFD (and used in the surface velocity POD analysis above). The use of (pseudo) tufts as illustrated by the sine and cosine of the local flow angle in fig. 6.4.2 or visually in fig. 6.4.3 eliminates the velocity magnitude while retaining the flow direction at a reduced spatial resolution. The use of tufts allows to make an equivalent assessment as with the surface velocity while eliminating the need for complex and extensive experimental measurements. The CFD data has been used to compare the surface velocity POD to the POD of pseudo tufts.

In fig. 6.4.8 the singular values of the POD of the tufts is shown. As each snapshot sample contains 260 tufts which are described by a sine and cosine. The POD basis contains 520 spatial modes, obtained from a total of 520 measurements per snapshot sample. The resulting spatial modes have singular values which follow a similar trend as the velocity POD but have a drop-off at approximately 500 modes (due to reduced dimensionality of the tuft snapshot samples). For the first 10 modes shown in the inset, again a similar trend can be observed, indicating the similarities between the POD for the full velocity fields or just the local flow angles as obtained from tufts.

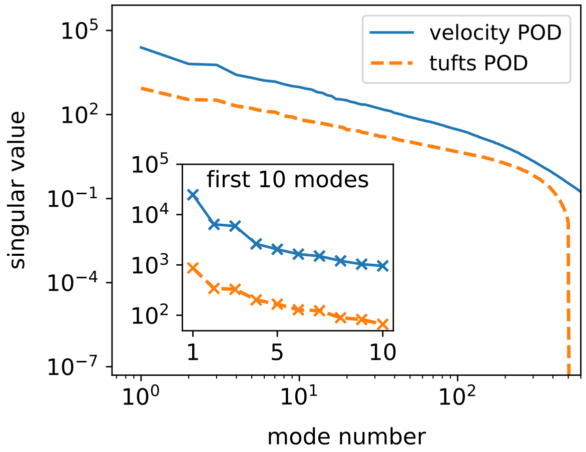


FIGURE 6.4.8: Singular values of the POD for the suction surface pseudo tufts on the wing in comparison with the singular values for the velocity POD.

The similarity between the velocity POD and tuft POD can further be illustrated through the spatial modes of the tufts as shown in fig. 6.4.9. This illustrates how the sparser information of the tufts can still capture the most important surface flow features, albeit with less detail than shown in fig. 6.4.6. The first mode in this case for the tuft POD does not represent any flow velocity but flow direction indicating downstream flow along most of the chord length. It must be noted that the sign of spatial mode is arbitrary. The temporal modes contain the sign necessary to decompose or reconstruct a sample with a given POD basis. The second and third modes of the tufts still represent similar flow scenarios as identified by the second and third velocity POD modes. It can be observed that mode 2 and mode 3 for the tufts have been switched compared to mode 2 and mode 3 for the velocity, from the singular values in fig. 6.4.8 it can be seen that the singular values for the second and third mode are almost equal, leading to the potential switch of the modes for second and third position as influenced by small differences in the training data sets for either the velocity POD or tufts POD.

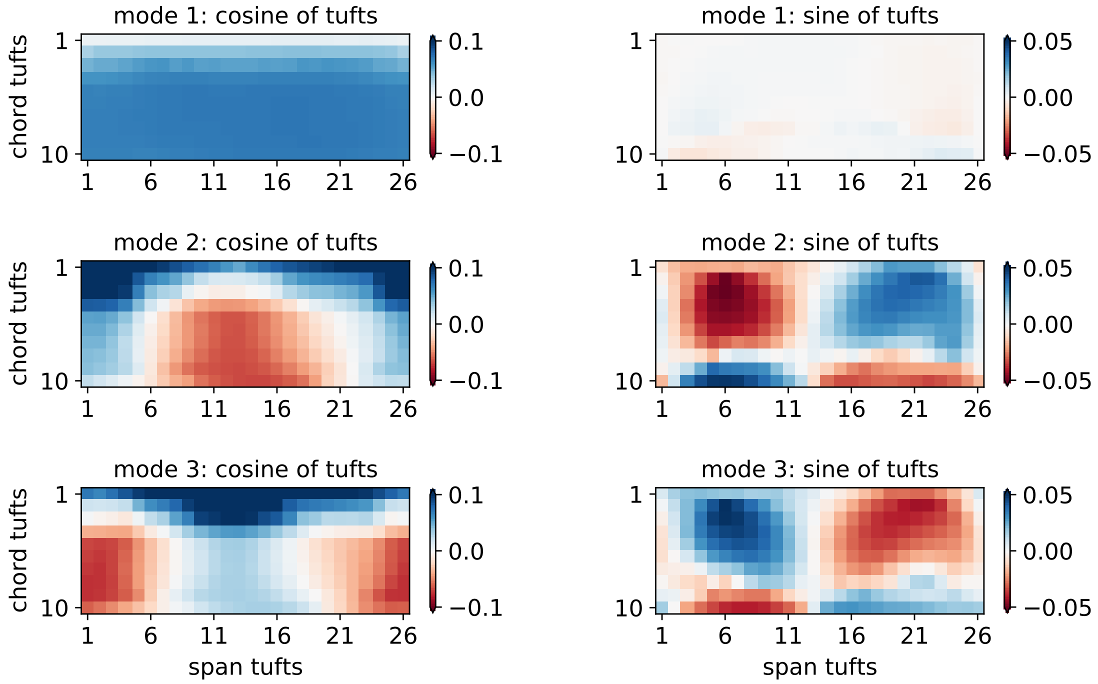


FIGURE 6.4.9: The cosine and sine of the first three spatial modes for the POD of the URANS CFD tuft data set.

Similar to the analysis with the velocity POD, the tuft POD modes can be used to investigate the same flow case as has been shown in fig. 6.4.7. For the Reynolds number of 10^6 and angle of attack 18° the temporal mode signal and frequency spectra are compared to the lift coefficient as shown in fig. 6.4.10. For the case with angle of attack 18° the second tuft POD mode represents the full stall cell. This can be observed to be the case in the temporal mode signal which is positive. However, also the negative signal of the third mode is clearly visible. As the velocity magnitude does not play a role and only the local flow direction is relevant, the second and third mode are similar but opposite when considering the sine of the tufts (spanwise flow direction). When considering the frequency spectra the results can be seen to still retain frequency peaks that match for the temporal modes and the lift coefficient, indicating that the use of the local flow directions suffices to obtain an indication of the unsteady lift coefficient oscillations due to surface flow patterns. The frequency spectra of the tuft temporal modes however do deviate more from the lift coefficient spectrum relative to the surface velocity temporal mode spectra as were shown in fig. 6.4.7.

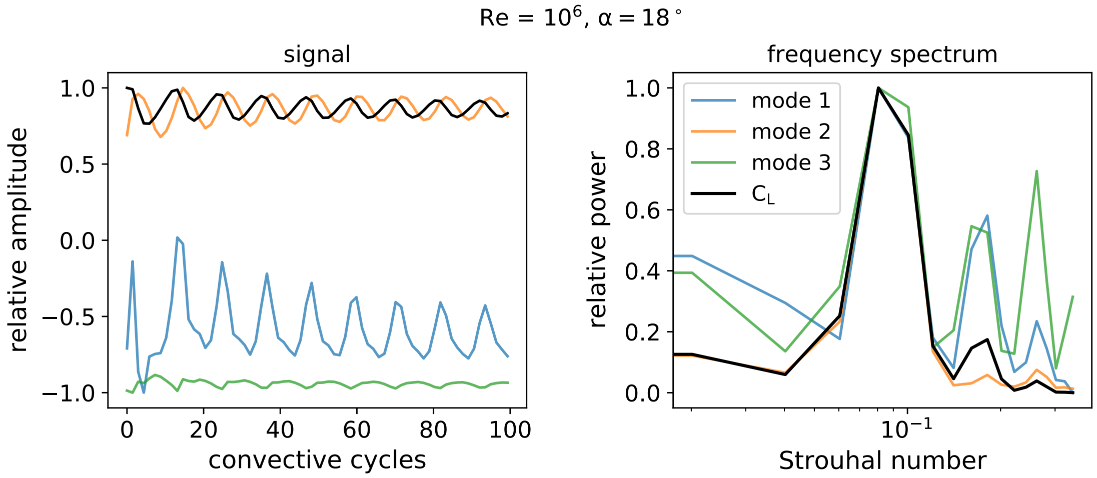


FIGURE 6.4.10: Tuft POD temporal modes frequency spectrum with lift coefficient frequency spectrum. Both signals and frequency spectra are normalised with their respective maximum of the absolute values. Left: the signal under investigation, right: the resulting frequency spectrum. Convective cycles are defined as the chord length divided by the freestream velocity.

6.4.4 Unsteady lift from tufts

6.4.4.1 Computational results

In section 6.3.2 the procedure has been described which can be followed to obtain an unsteady wing loading estimate from the pseudo tufts. For this computational approach, the first step is to obtain a steady estimate for the wing lift coefficient for each α − Re combination. In order to limit the influence of the initial transient phase in the simulation, the initial 25% of the simulated time has been ignored. The resulting average lift coefficients for each α − Re combination of the remaining 75% of the simulated time, are shown in fig. 6.4.11. For the calculation of the average lift coefficient per case also the test cases have been taken into account, as no other suitable method would lead to a reasonable estimate for the average lift coefficient. Interpolating the known average lift coefficient based on known C_L values for different α − Re combinations can lead to large errors if not enough α − Re combinations are known. Alternatively some other unrelated methods to obtain a steady estimate might exist (2D RANS, potential flow, etc.) that are computationally cheaper than 3D URANS, but come with other disadvantages such as limited accuracy for separated flow cases as under investigation here.

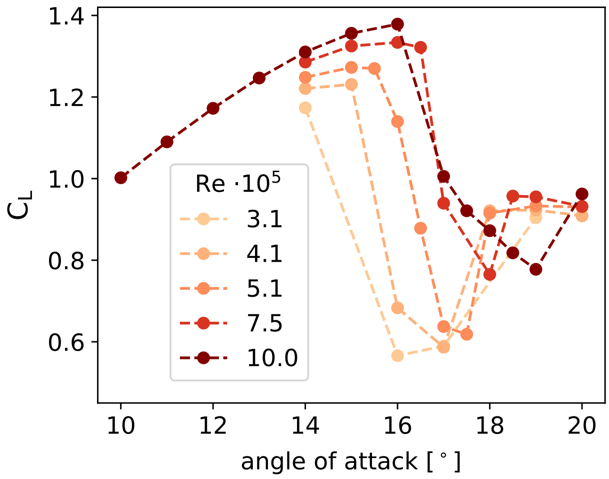


FIGURE 6.4.11: The average lift coefficient obtained from the URANS simulations for each α — Re combination. The initial 25% of the simulated time is ignored to omit the transient phase.

The equations (6.2),(6.3) and (6.4) can be combined into eq. (6.6), where the left hand side of the equation represents the unsteady component of the lift coefficient and the right hand side the optimised linear equation based on the temporal modes of the snapshot samples and trained coefficients. These coefficients can be optimised by minimising the root mean squared error (RMSE) between the measured unsteady lift coefficient and the estimated value. In the optimisation only the training data is used such that the test data represents unseen combinations of angle of attack and Reynolds number.

$$C_{L,\mathrm{i}} - \overline{C_L}(\alpha, \mathrm{Re}) = \mathrm{W_L} \cdot \Psi_\mathrm{i} + \mathrm{B_L} + \mathrm{error} \tag{6.6}$$

The unsteady component of the lift coefficient is estimated based on a number of included temporal modes with the corresponding weights and a single bias. It can be expected that the inclusion of more modes results in better performance. This has been evaluated by considering a varying amount of modes to include between 10 and 100 modes. For validation 500 random samples from the training data set were used, these samples allow for evaluation in between training iterations on the remaining training data (epochs - this is the validation data set). When the evaluation of the 500 samples did not improve for 200 iterations the optimisation was halted and the weights and bias resulting in the lowest error for the 500 samples were restored (also known as early stopping). For each varying amount of modes considered 10 different optimisations have been run, for which the resulting error and standard deviation are shown in fig. 6.4.12. From the figure it can be seen that indeed for the training data set (and the validation subset), the error decreases for increasing amounts of modes included. However, for data driven methods the tendency to overfit can occur especially when the test set represents entirely different cases with unseen angle of attack and Reynolds number combinations in the training data set. The test data set illustrates the importance of evaluating the ability of a method to generalise to different cases, which with the current data set and training – test split can be found to be close to optimal around 30 modes. It is also noticeable that the train and validation error are similar, with the training error smaller than the validation error. Both training and validation data sets contain samples of the same angle of attack and Reynolds number cases. These data sets (training and validation) are thus similarly distributed and the error is thus also similar. As the network coefficients are updated based on the training data and evaluated on the validation data, the error is lower for the training data. Lastly, the test data has been taken from angle of attack and Reynolds combination cases which do not occur in the training and validation data. As such the error for these cases can be significantly different. In the current investigation the test cases tend to contain the type of surface flow which can be predicted well due to the unsteadiness in the tufts, leading to the low error relative to the training data. Some of the $\alpha-$ Re cases in the training data, such as attached flow do not benefit from an unsteady model for performance prediction based on tufts.

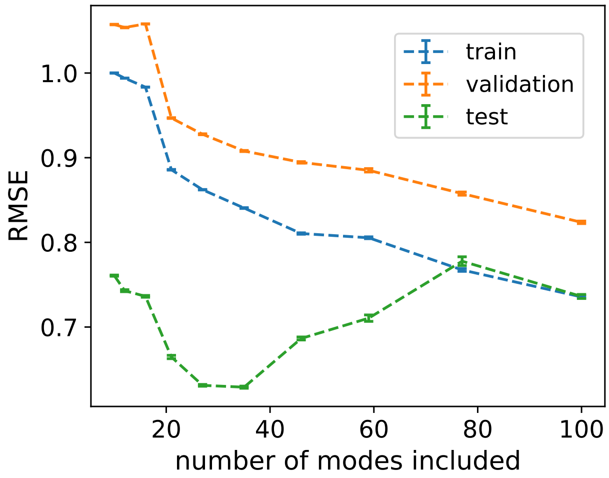


FIGURE 6.4.12: Resulting error for $\mathcal{F}_{\text{linear}}$ with increasing amounts of modes included. Root mean squared error for the training data set (train: 4131 samples), the subset for intermediate evaluation (validation: 500 samples) and the test set without any influence on the training (test: 731 samples).

A function has been created with 30 weights for 30 temporal modes and one bias. The function produces the unsteady lift coefficient component (ΔC_L in fig. 6.4.13) which has different characteristics for different types of test cases. The flow type as identified in fig. 2.6 is also indicated at the start of the respective x-axes in fig. 6.4.13. It can be noticed that for cases with stall cells the linear function can reproduce the unsteady lift with reasonable accuracy, most notably the frequency content of the test samples appears to be reproduced well for all cases. For the cases with fully separated flow over the wing suction surface there is only very little variation in the flow direction observations through tufts. Relevant information to estimate the lift coefficient fluctuations amplitude variations could be contained in the velocity. For the cases with full span separation the tufts are thus not able to provide the necessary information for the magnitude of the lift coefficient fluctuations, however it remains possible to estimate the frequency of the fluctuations based on the flow direction. Lastly for the cases with TE separation there is nearly no unsteady component. The small deviation that is introduced by the linear equation is mostly due to a combination of several modes rather than the bias (B_L) of the equation which is -0.003 (as the linear function only considers lift fluctuations).

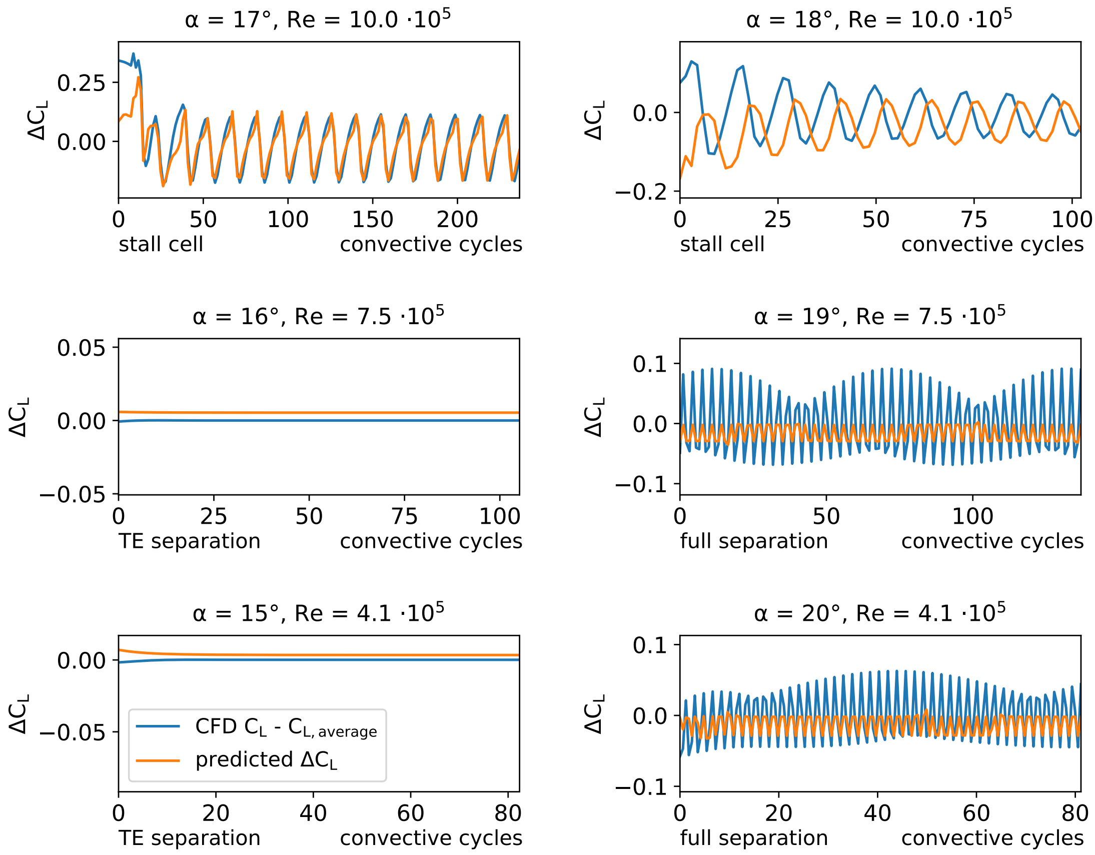


FIGURE 6.4.13: The unsteady lift coefficient component (ΔC_L), the true result from CFD and the predicted result from the linear function for the test cases.

Finally the error of the linear function, as described in eq. (6.6), can be modelled with the non-linear function based on the tuft observations as described by eq. (6.5). The use of this non-linear one layer neural network is to investigate the ability to use advanced modelling and optimisation techniques outside conventional linear methods. For neural networks the performance that can be achieved often comes at the cost of interpretability. Using an adaptation of the universal approximation theorem keeps the neural network complexity at a minimum while offering improved performance over the previously used linear method. For the linear method the full amount of tufts are implicitly used when obtaining the POD modes and the corresponding coefficients. After the POD coefficients or temporal modes have been obtained, a rank truncation can be applied to reduce the amount of parameters of the final linear combination which gives the unsteady C_L. The universal approximation theorem has been implemented as a single layer neural network with Sigmoid activation in the latent layer and a linear activation at the output node. This implementation can be interpreted similarly to the linear approach. The input layer of the non-linear network takes in all the tufts and transforms these into a latent space of predefined size similarly to the POD rank truncation. The output node provides the linear combination of the latent space coefficients, similarly to the linear combination of the temporal modes. The main difference is thus the non-linear activation function applied to the latent space. Increasing the size of the latent space of the neural network can be expected to improve the performance similarly as for the linear approach. In fig. 6.4.14 the resulting RMSE is shown for the non-linear network, with 10 optimisations per latent space size. The error can be seen to initially decrease with increasing number of latent space nodes. After 35 nodes no improvements can be observed with increasing latent space size. This is possibly a limitation caused by the amount of data required or by the limited information that is contained in a current snapshot sample. For the non-linear function the standard deviation can also be observed to be much larger compared to the optimisation of the linear function at each number of nodes or modes to include.

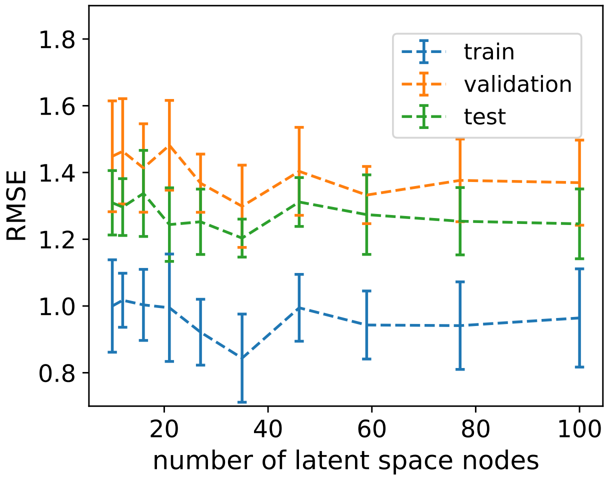


FIGURE 6.4.14: Resulting error for $\mathcal{F}_{\text{non-linear}}$ with increasing amounts of nodes in the hidden layer. Root Mean Squared Error (RMSE) for the training data set (train: 4131 samples), the subset for intermediate evaluation (validation: 500 samples) and the test set without any influence on the training (test: 731 samples).

$$C_{L,\mathrm{i}} - \overline{C_L}(\alpha, \mathrm{Re}) - (\mathrm{W_L} \cdot \Psi_\mathrm{i}) + \mathrm{B_L} = \sum_{\mathrm{j}=1}^{35} \mathrm{a_j} \cdot \sigma(\mathrm{W_{NN,j}} \cdot \mathrm{X_i} + \mathrm{B_{NN,j}}) \tag{6.7}$$

A neural network with 35 nodes has been created which attempts to optimise the coefficients on the right hand side of eq. (6.7). Where on the left hand side the 30 optimised weights for the linear function are represented by vector $\mathrm{W_L}$, on the right hand side the weights $\mathrm{W_{NN,j}}$ represent 35 vectors (layers) of 520 floating point numbers which produce the dot product with the tuft inputs ($\mathrm{U_i}$). The equation can be optimised to produce a test RMSE of 0.025. The results of the test set are shown in fig. 6.4.15, where the final resulting estimated unsteady C_L is shown through combining the steady estimate with the unsteady linear and non-linear estimate. The non-linearly estimated part of the unsteady lift coefficients makes some improvements to all the different types of surface flow cases. For cases with stall cells the lift coefficient matches the frequency, phase and magnitude slightly better than with only the linearly estimated unsteady lift coefficient. For the full separation cases the magnitude of the C_L fluctuations is better represented, however the non-linearity does not affect the lack of the ability to capture the change in amplitude of the fluctuations over time.

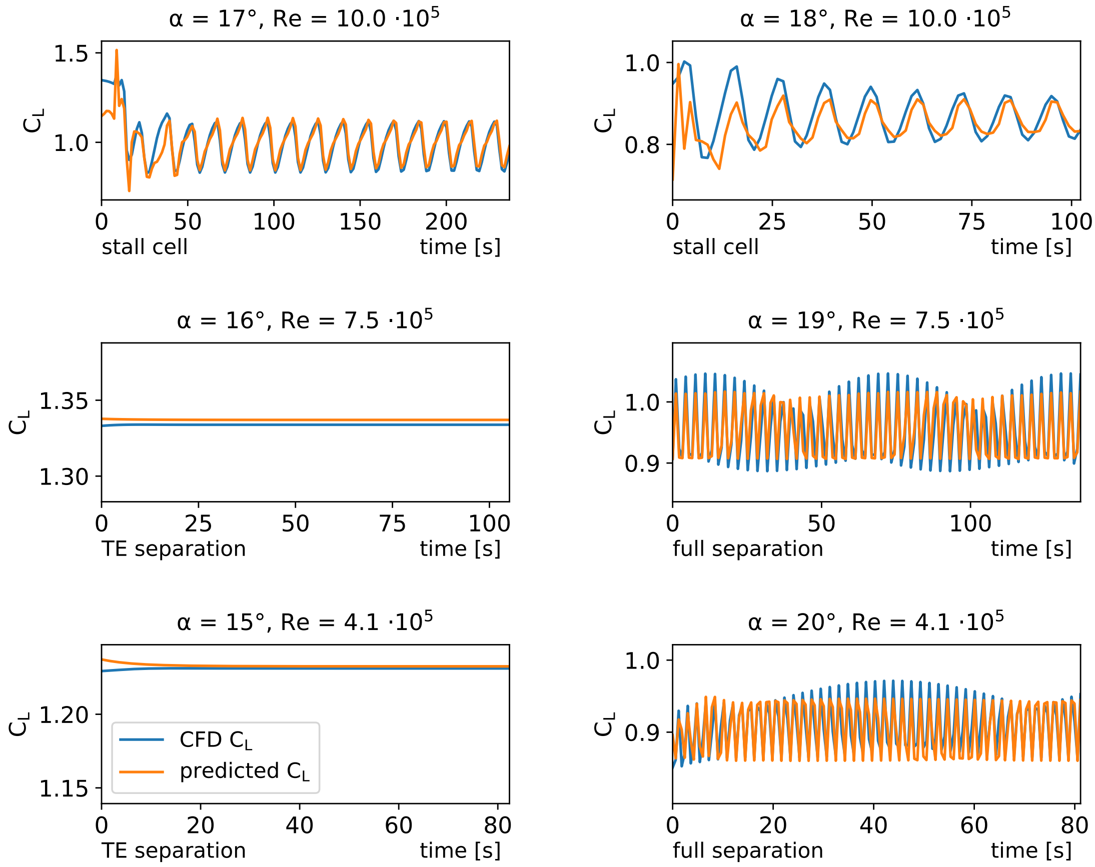


FIGURE 6.4.15: The unsteady lift coefficient, the true result from CFD and the predicted result from the linear function for the test cases. The predicted result includes the average lift coefficient, the linearly predicted unsteady component and the non-linearly predicted unsteady component.

6.4.4.2 Experimental results

While it is possible for the linear and non-linear method used in the computational investigation to explore an optimal amount of modes or nodes to include, for deep neural networks the tuning of hyper parameters is typically more challenging. Shallow two-layer neural networks have previously been demonstrated to be trainable on small data sets with good performance (Carter et al. (2021), Erichson et al. (2020)). In the current investigation a simple two-layer neural network has been used which gradually reduces the width of the network from an input size of 110 to a single output. The training data set consists of 65k samples and a separate set of 10k samples is available for validation (during training). The test set for evaluating the performance of the trained neural network also consists of 10k samples. All samples are taken from one experimental run at a constant angle of attack and Reynolds number, where the training data is the first recorded data followed by the validation data and the test data was recorded last. The validation samples are thus not an interpolation of the training samples, and the test data is also independent of the training data. Both the validation during training and the test evaluation thus represent out of sample predictions. The computational data showed highly periodic behavior for each $\alpha - \mathrm{Re}$ combination, which is not the case for the experimental data. During initial data exploration it was noticed that there appears to be a slight temporal offset between the measurements of the lift and the pitching moment. As the lift measurements appear to lag the tuft and moment measurements, the lift measurements have been shifted forward temporally in the training data by 8 time steps (0.067 seconds). The optimal temporal shift was determined by testing several different delays between five and fifteen time steps and selecting the shift which results in the lowest error for the lift coefficient surrogate model. For the pitching moment measurements the lowest error was found without any temporal shift.

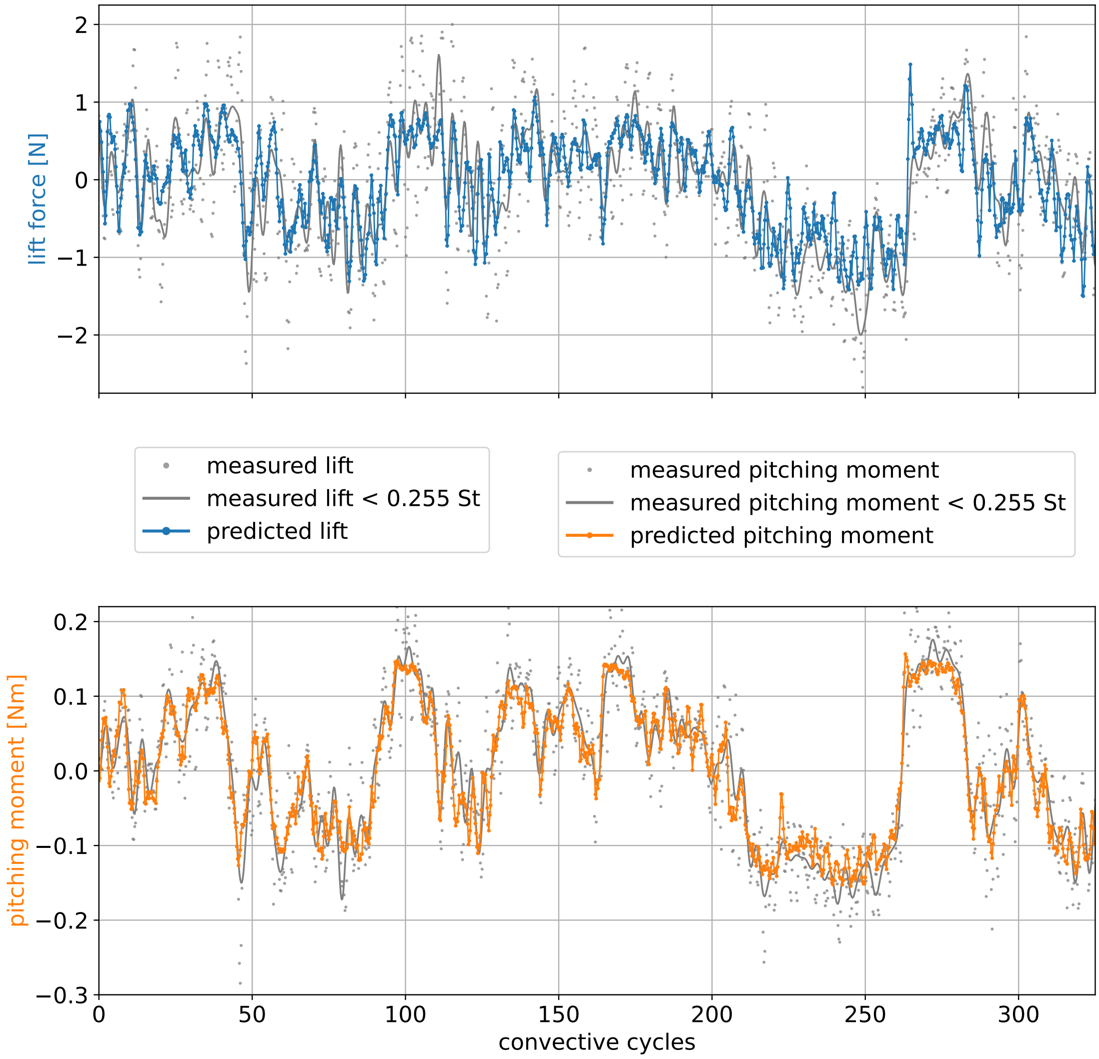


FIGURE 6.4.16: The unsteady lift and pitching moment estimated from experimental tuft recordings with a shallow neural network.

In fig. 6.4.16 the result is shown for a time frame of 10 seconds which is part of the test data set. The mean lift force measured was 11.15 N and the mean measured pitching moment was 0.24 Nm. The measured lift force fluctuations filtered at a Strouhal number of 0.255 shown in fig. 6.4.16 have a peak-to-peak amplitude of approximately 21% of the mean value, while similarly for the pitching moment this amplitude is approximately 125% of the mean value. The neural network can estimate the magnitude and phase of these fluctuations well based on the tuft recordings. The component at a Strouhal number of 0.3 of the measured fluctuations is more significant for the lift than for the pitching moment, as shown in fig. 6.4.17. In fig. 6.4.17 a comparison is shown between the unfiltered measured data (similar to training data) and the network prediction for the test data set. The match in frequency content is very good with exception of the peak at a Strouhal number of 0.3 for the lift and (also St = 0.6) for the pitching moment. Currently it can not be determined definitively what the cause is for the neural network to not capture the appropriate magnitude of the component of the fluctuations at a Strouhal number of 0.3. However, as no significant changes in tuft motion can be seen on such a small timescale it is possible that the fluctuations at these frequencies are not of aerodynamic origin, or cannot be registered properly with tufts. A short overview of the tuft motion frequency content is shown in section 6.4.4.3 this indicates that it is unlikely to obtain a significant frequency component at a Strouhal number 0.3. The higher Strouhal number range appears to still match but the significance of this Strouhal number range is very limited and close to representing noise. The networks and models used in this investigation have only been trained to process a single independent sample. It could be possible that the inclusion of multiple time steps in a model improves the results.

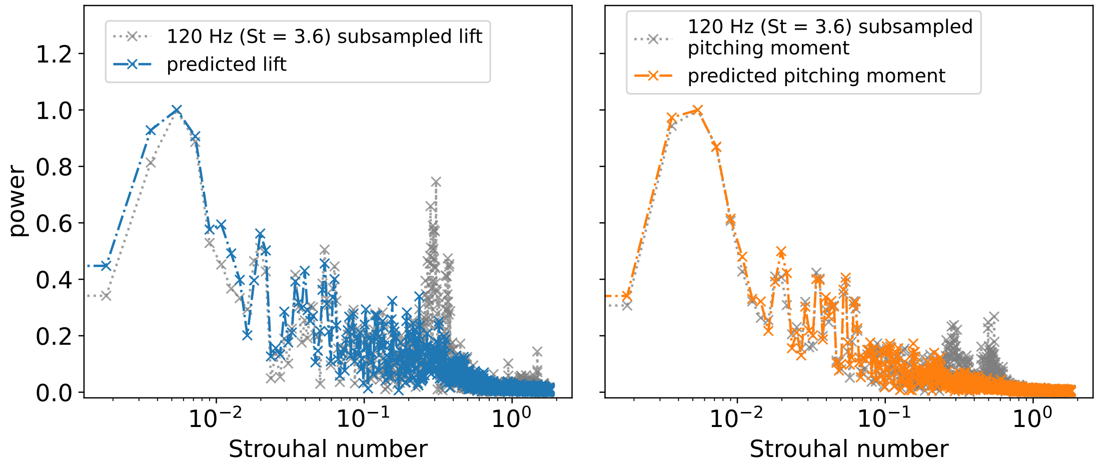


FIGURE 6.4.17: The frequency spectra, normalised by the low-frequency peak value, of the lift and pitching moment fluctuations in the test set of the data. A comparison is shown between the spectra of the measured values and the predicted values.

6.4.4.3 Time response of tufts

In fig. 6.4.18 and fig. 6.4.19 the frequency spectra are shown of 11 spanwise adjacent tufts near the centre of the chord, for the cosine and the sine of the tuft angles respectively. These tufts consider some locations which are prone to showing attached flow as well as tufts which tend to be inside a stall cell with separated flow. While the analysis of the specific individual tuft behavior in relation to the surface flow is outside of the scope of the current investigation, it remains relevant to assess the range of frequencies which can be found in the tuft data which is used as an input for the data driven method with a neural network. In fig. 6.4.18 it can be observed that the peak power of the individual tuft angle cosine frequency spectra is found between 0.1 and 1 Hz. The cosine of the tuft angle indicates the chordwise flow direction. Large changes in the cosine can thus represent the switch between attached flow aligned with the freestream or separated flow going upstream near the wing surface. In fig. 6.4.19 the frequency spectra of the sine of the tuft angles are shown, which indicate that no clear peak values are present in the frequency spectra. The sine of the tuft angle represents the change in spanwise flow direction. It can be observed that for the spanwise flow unsteadiness the relevant frequencies are an order of magnitude larger than for the chordwise flow unsteadiness (as represented by the cosine of the tuft angles). It appears that the 10 Hz frequency as found in the lift and pitching moment data is also present in the spanwise fluctuations but not with a clear distinct peak value. The analysis of the tuft fluctuations as presented through the frequency spectra of the sine and cosine of the tuft angles neglects the important aspect of the magnitude of the fluctuations, which is likely important in the physical analysis of the flow. Future investigations may focus on the complete combination of the tuft angle frequency spectra and the magnitude of the tuft angle fluctuations to improve the knowledge of separated flow structures.

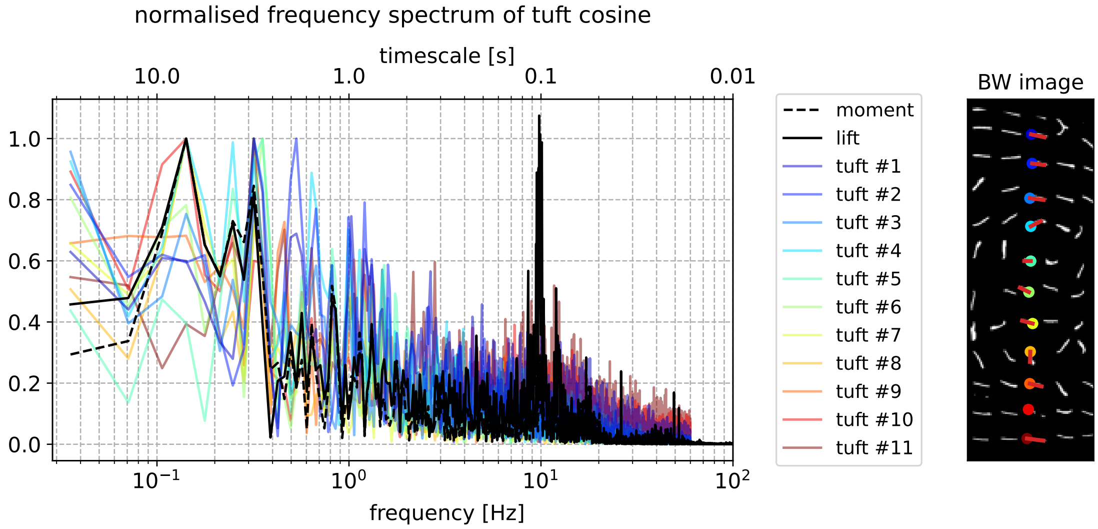


FIGURE 6.4.18: Frequency spectrum comparison of different tufts at a similar chordwise position along the span. The frequency spectrum considers the variation of the cosine of the tuft angles, which is equivalent to the tuft fluctuations in chordwise direction.

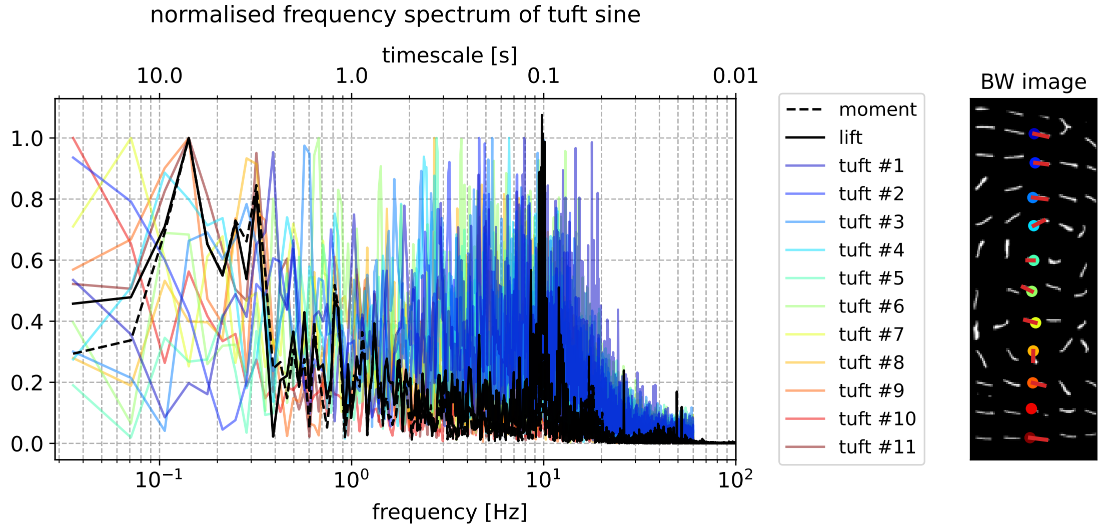


FIGURE 6.4.19: Frequency spectrum comparison of different tufts at a similar chordwise position along the span. The frequency spectrum considers the variation of the sine of the tuft angles, which is equivalent to the tuft fluctuations in spanwise direction.

6.5 Conclusions

In the current work it has been shown that, with both computational and experimental data, tufts in combination with data driven methods can provide quantitative information about the unsteady aerodynamic loading of a wing with a NACA 0012 profile.

The computational data consists of varying combinations of angle of attack and Reynolds number URANS simulations of a 3D wing with a NACA 0012 airfoil profile. Most of the simulated α − Re combinations contain unsteady surface flow with corresponding unsteady wing loading. The flow direction near the surface of the wing has been utilised to obtain pseudo tufts, which have been used as an input for a surrogate model that estimates the unsteady lift coefficient fluctuations. It has initially been shown that the surface flow velocity holds a strong relation to the lift coefficient through the use of a frequency analysis of the SVD temporal modes (equivalent to POD coefficients) of a snap shot sample. As a next step the local flow direction has been used to indicate a similar relation in terms of the frequency content of the temporal modes of the tufts. A linear combination of the temporal modes of tuft snapshot samples has been shown to be capable of estimating the unsteady lift coefficient for cases with flow separation. The remaining error after the linear estimate of the unsteady lift coefficient has further been captured through optimising the coefficients for the equation of the universal approximation theorem, which has been implemented as a single hidden layer neural network with a Sigmoid activation layer. The non-linear approximation further improves the estimation of the unsteady lift coefficient. The computational investigation showed that it is possible to use a single surrogate model which provides lift fluctuation estimates for multiple angles of attack, Reynolds numbers and surface flow configurations based on the pseudo tufts.

Unsteady separated flow tends to involve turbulent flow which is not represented well with the current URANS simulations. Therefore, an experimental investigation has been performed to assess the ability of a data driven model to function on complex flow patterns with input data which contains both noise and turbulence. The experimental investigation focused on a wing with a NACA 0012 airfoil profile, which has been investigated at a single combination of angle of attack (13°) and Reynolds number ($2.1 \cdot 10^5$) which produced highly unsteady surface flows. The forces and moments were recorded simultaneously with the tufts. With 120 frames per second for the experimental tuft recordings it is possible to acquire multiple orders of magnitude more samples than with simulations of a similar flow. The large amount of samples available with imperfect complex input data have been used to train two separate neural networks. Both neural networks have two hidden layers and a single output, one network is trained on the lift fluctuations and the other on the pitching moment fluctuations. With less than 10 minutes of sampled data at 120 Hz it has been found possible to capture the fluctuations of the lift and pitching moment with snapshot samples of the tufts. It is possible that the current results for the experimental investigation can be improved upon by tuning the network parameters or even including multiple time steps in the input.

Chapter 7

Conclusions

Research of separated flows is notoriously difficult due to the chaotic nature of the flow. Separated flow over a wing includes unsteady interaction of flow structures on a range of spatial and temporal scales. Experimental and computational methods have been essential to the research of turbulent flows. Both methods offer specific advantages and come with certain disadvantages. In the last few decades advancements in computing have helped to improve experimental and computational methods. Investigations now yield very large data sets which have become difficult to process. The advancements in computing have also helped bring data driven methods such as neural networks to the forefront. These new methods allow us to create models which capture the information contained in data sets without the need for an explicit formulation of the observed system. Both linear and non-linear methods are commonly used to capture important flow features or approximate complex systems. The current work focused on different aspects of separated flow in combination with relevant data driven methods. The initial focus has been on reducing the dimentionality of data sets with a large number of parameters. This has been explored with sparse reconstruction of particle image velocimetry velocity fields, with approximately 51 000 parameters per sample, through the use of the proper orthogonal decomposition (POD) and non-linear refinement. Recently developed methods for non-linear mode decomposition were explored next. Variations of autoencoders were shown to have particular strengths and drawbacks. The dimentionality reduction has then been applied to create a data driven stall detection model based on surface pressure measurements. The final part of the research showed that it is possible to use tufts in combination with a neural network to obtain a quantitative estimate of the wing performance during unsteady flow. The use of sparse sensors in combination with data driven methods have been shown to be a powerful tool. These tools allow to extend measurement capabilities beyond their initial purpose, and enhance our understanding of large and complex systems.

7.1 Summary

In chapter 3 a data set of two dimensional PIV flow fields has been used to investigate different sparse reconstruction methods. Sparse reconstruction methods utilise sparse measurements of a system to estimate the full state of that system. The flow fields contain turbulent separated flow over a NACA 0012 airfoil at a Reynolds number of 75000 and an angle of attack of 12°. The sparse pseudo probe measurements in this case are represented by local velocities at a point within the velocity field. Three different linear methods have been compared with either randomly positioned probes or positions obtained with Q-DEIM method. From the investigation of the linear methods it was shown that the compressed sensing approach in combination with Q-DEIM probe positioning performed the best. The extended POD and quasi orthogonal extended POD methods were preferable when the probes were positioned randomly. These linear methods estimate the POD coefficients for a POD basis which contains the POD modes of the full velocity fields. The linear estimation of these coefficients based on a limited set of measurements was shown to be improved by using a neural network to update the estimated coefficients. Such a network takes in the POD coefficients as estimated by the linear methods and updates these to reduce the reconstruction error with the POD basis of the flow fields. The neural network does not use the probe measurements as an input and thus can be interpreted as having learned the error that is introduced in the linear estimate of the POD coefficients based on the sparse measurements, allowing the neural network to correct that error. The neural network updating of the coefficients showed large improvements in the reconstruction of the flow field from randomly positioned sparse probes.

In chapter 4 non-linear modes of a separated flow field have been investigated. The popularity of POD to decompose a data set into modes has more recently sparked a lot of interest in creating non-linear modes through the use of neural networks. The PIV data set with separated flow fields has been used to compare several methods which result in non-linear modes. The principle of hierarchical subnetworks has been illustrated on a linear single layer autoencoder to show the equivalence to POD. The same process of successively training subnetworks can be applied to non-linear autoencoders to produce non-linear modes which are ranked according to the information they capture in the training set. The non-linear hierarchical autoencoders create a latent space distribution which can be hard to interpret. Variational autoencoders introduces probabilistic constraints to regularise the latent space. This results in autoencoders with a latent space that resembles a known distribution. An additional parameter (β) can be tuned to balance the reconstruction error of the autoencoder and the latent space regularisation. It was found that with the correct value for β the autoencoder learns a set of non-linear modes which are near orthogonal. These modes have a reduced reconstruction error compared to POD and an improved Pearson correlation. As the latent space distribution for the latent variables matches $\mathcal{N}(1,0)$ closely, it is possible to sample points from this distribution to create new realistic samples.

In chapter 5 the flow over a NACA 0012 wing has been simulated with unsteady Reynolds averaged Navier Stokes (URANS). Many different Reynolds number and angle of attack combinations have been simulated to obtain a representative overview of the flow on the wing ranging from attached flow to fully separated flow. The importance of the flow near the surface of the wing has been illustrated for cases which contain some form of flow separation. The surface pressure on the wing has been shown to be closely related to the surface flow patterns that are observable with surface flow visualisation techniques. Experimental and computational measurements of the surface pressure on a wing with a stall cell present showed good agreement when the location of the pressure measurements is relative to the surface flow pattern. The URANS simulations allowed to investigate the unsteadiness of the flow over a wing with a stall cell present. The fluctuations in the lift coefficient were indicated to be related to the undulations of the stall cell. Based on these observations a flow type classifier has been built which could serve as a stall detection system. The flow type classifier consists of a system of five equations to classify the surface flow into one out of four different flow types. The equations use the first two proper orthogonal decomposition (POD) modes of surface pressure measurements as inputs. This very simplistic model already provides 98% accuracy.

In chapter 6 an approach to estimate the unsteady lift coefficient from visual tuft observations has been created. Tufts have long been used as a qualitative tool to visualise the flow near the surface of an object. It has been shown that the surface flow is closely related to the surface pressure on a wing. As such, the tufts can be reasoned to contain information about the performance of a wing. It is not feasible to create an analytical expression that describes the relation between tuft observations and aerodynamic parameters. It is however possible to use a neural network to approximate this relation. An initial investigation of this approach on a computational data set showed the working principle. An experimental investigation of a NACA 0012 wing experiencing highly unsteady flow provided simultaneous images of tufts and integral force and moment measurements of the wing. This experimental data contained erratic flow and flow fluctuations. By processing the tuft images into data suitable for a neural network it was shown that this network can accurately estimate the unsteady lift and pitching moment fluctuations of the wing.

7.2 Future research

The research in this thesis has addressed several gaps in the existing literature of data driven methods for separated flows. The current research also indicates how there are still unanswered questions for this topic.

- The investigation of separated flow through computational means suffers from inherent inaccuracies introduced by the simplifications in the computational models. A thorough experimental surface flow investigation of the progression of stall including stall cells is still missing from literature. Such data can provide an important link between visually observable flow structures with tufts and the measurable properties of the flow.
- Although several methods for sparse reconstruction have been compared, there are many more possibilities for sparse reconstruction. A full non-linear approach likely performs better than the linear approach with non-linear refinement.
- The use of non-linear modes for reduced order modelling showed benefits such as a reduced reconstruction error and an improved correlation compared to the linear POD. However non-linear modes remain difficult to interpret, which is a key aspect of the decomposition into modes for flow analysis. Hierarchical methods which successively train subnetworks naturally create a ranking of the non-linear modes. Applying the hierarchical approach to β variational autoencoders can bring a clear structuring to successive modes which already have a known distribution for the latent space.
- Stall detection systems are essential for aircraft safety. In the current work a data driven approach has shown great results on a computational data set. More advanced data driven methods in combination with experimental data can create a stall detection model which performs well on chaotic real life data with just a limited set of sensors.
- The surrogate model for the lift coefficient based on tufts has been shown to be generally applicable for different angles of attack and Reynolds number for computational data. It remains to be tested if such surrogate model can be created for experimental data which works for a range of angles of attack and Reynolds numbers. Similarly when given the specific airfoil parameters as an input, it might be possible that such a surrogate model based on a neural network could further generalise to different airfoils.

References

Adrian, L., Adrian, R.J., and Westerweel, J. *Particle image velocimetry*, volume 30. Cambridge university press, 2011.

Adrian, R.J. and Moin, P. Stochastic estimation of organized turbulent structure: homogeneous shear flow. *Journal of Fluid Mechanics*, 190:531–559, 1988.

Ashton, N. and Revell, A. Comparison of RANS and DES Methods for the DrivAer Automotive Body. *SAE International*, 2015-01-1538, 2015.

Atkinson, A.C. and Donev, A.N. *Optimum experimental designs*. Clarendon Press, 1992.

Bai, Z., Wimalajeewa, T., Berger, Z., Wang, G., Glauser, M., and Varshney, P.K. Low-dimensional approach for reconstruction of airfoil data via compressive sensing. *AIAA journal*, 53(4):920–933, 2015.

Bao, Y., Shi, Z., Wang, X., and Li, H. Compressive sensing of wireless sensors based on group sparse optimization for structural health monitoring. *Structural Health Monitoring*, 17(4):823–836, 2017.

Barrault, M., Maday, Y., Nguyen, N.C., and Patera, A.T. An 'empirical interpolation'method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

Bartl, J., Sagmo, K.F., Bracchi, T., and Sætran, L. Performance of the NREL S826 airfoil at low to moderate Reynolds numbers - A reference experiment for CFD models. *European Journal of Mechanics / B Fluids*, 2018.

Berkooz, G., Holmes, P., and Lumley, J.L. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.

Blei, D., Kucukelbir, A., and McAuliffe, J.D. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112, 2017.

Borée, J. Extended proper orthogonal decomposition: a tool to analyse correlated events in turbulent flows. *Experiments in fluids*, 35(2):188–192, 2003.

Bright, I., Lin, G., and Kutz, J.N. Compressive sensing based machine learning strategy for characterizing the flow around a cylinder with limited pressure measurements. *Physics of Fluids*, 25 (12), 2013.

Broeren, A.P. and Bragg, M.B. Spanwise variation in the unsteady stalling flowfields of two-dimensional airfoil models. *AIAA Journal*, 39 (9):1641–1651, 2001.

Brunton, S.L. and Kutz, J.N. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2019.

Brunton, S.L., Noack, B.R., and Koumoutsakos, P. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52:477–508, 2020.

Cai, C., Zuo, Z., Liu, S., and Maeda, T. Effect of a single leading-edge protuberance on naca $63_4 - 021$ airfoil performance. *Journal of Fluids Engineering*, 140 (2):021108, 2018.

Callaham, J.L., Maeda, K., and Brunton, S.L. Robust flow reconstruction from limited measurements via sparse representation. *Physical Review Fluids*, 4(10):103907, 2019.

Candès, E.J. and et al. Compressive sampling. In *Proceedings of the international congress of mathematicians*, volume 3, pages 1433–1452. Madrid, Spain, 2006.

Candès, E.J. and Wakin, M.B. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008.

Carter, D.W., De Voogt, F., Soares, R., and Ganapathisubramani, B. Data-driven sparse reconstruction of flow over a stalled aerofoil using experimental data. *Data-Centric Engineering*, 2 (5), 2021.

Ceglia, G., Astarita, T., and De Gregorio, F. An experimental investigation of rotor wakes by means of time-resolved piv measurements. *Aerospace Science and Technology*, 141, 2023.

Chaturantabut, S. and Sorensen, D.C. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

Chen, L., Suzuki, T., Nonomura, T., and Asai, K. Characterization of luminescent minituft in quantitative flow visualization experiments: Surface flow analysis and modelization. *Experimental Thermal and Fluid Science*, 103:406–417, 2019.

Chen, L., Suzuki, T., Nonomura, T., and Asai, K. Flow visualization and transient behavior analysis of luminescent mini-tufts after a backward-facing step. *Flow Measurement and Instrumentation*, 71 (101657), 2020.

Cox, D.R. and Reid, N. *The theory of the design of experiments*. CRC Press, 2000.

Crow, S.C. Stability theory for a pair of trailing vortices. *AIAA Journal*, 8 (12):2172–2179, 1970.

Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

De Voogt, F. and Ganapathisubramani, B. Effects of a trailing-edge flap on stall cell characteristics of a NACA0012 wing. *Flow*, 2 (15), 2022.

Dell'Orso, H. and Amitay, M. Parametric Investigation of Stall Cell Formation on a NACA 0015 Airfoil. *AIAA Journal*, 56 (8):3216–3228, 2018.

Dell'Orso, H., Tuna, B.A., and Amitay, M. Measurement of Three-Dimensional Stall Cells on a Two-Dimensional NACA0015 Airfoil. *AIAA Journal*, 54 (12):3872–3883, 2016.

Discetti, S., Raiola, M., and Ianiro, A. Estimation of time-resolved turbulent fields through correlation of non-time-resolved field measurements and time-resolved point measurements. *Experimental Thermal and Fluid Science*, 93:119–130, 2018.

Doersch, C. Tutorial on variational autoencoders. *arXiv:1606.05908*, 2016.

Dong, W., Zhang, L., Shi, G., and Wu, X. Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Transactions on image processing*, 20(7):1838–1857, 2011.

Donoho, D.L. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

Drmac, Z. and Gugercin, S. A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions. *SIAM Journal on Scientific Computing*, 38(2):A631–A648, 2016.

Eivazai, H., Le Clainche, S., Hoyas, S., and R., Vinuesa. Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. *Expert Systems with Applications*, 202, 2022.

Engelberg, S. *Digital Signal Processing: An Experimental Approach*. Springer London, 2008.

Erichson, N.B., Mathelin, L., Yao, Z., Brunton, S.L., Mahoney, M.W., and Kutz, J.N. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proceedings of the Royal Society A*, 476 (2238), 2020.

Fukami, K., Nakamura, T., and Fukagata, K. Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data. *Physics of Fluids*, 32, 2020.

Galamhos, C., Matas, J., and Kittler, J. Progressive probabilistic hough transform for line detection. *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:554–560, 1999.

Geibel, M. and Bangga, G. Data reduction and reconstruction of wind turbine wake employing data driven approaches. *Energies*, 15 (10), 2022.

Gregory, N. and O'Reilly, C.L. Low-Speed Aerodynamic Characteristics of NACA 0012 Aerofoil Section, including the Effects of Upper-Surface Roughness Simulating Hoar Frost. *Aeronautical Research Council: Reports and Memoranda*, (3726), 1970.

Gross, A., Fasel, H. F., and Gaster, M. Criterion for Spanwise Spacing of Stall Cells. *AIAA Journal*, 53 (1):272–275, 2015.

Guastoni, L., Güemes, A., Ianiro, A., Discetti, S., Schlatter, P., Azizpour, H., and Vinuesa, R. Convolutional-network models to predict wall-bounded turbulence from wall quantities. *Journal of Fluid Mechanics*, 928, 2021.

Gunes, H., Sirisup, S., and Karniadakis, G.E. Gappy data: To krig or not to krig? *Journal of Computational Physics*, 212(1):358–382, 2006.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. *International Conference on Learning Representations*, 2017.

Hinton, G.E., Osindero, S., and Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.

Hornik, K. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4 (2):251–257, 1991.

Hosseini, Z., Martinuzzi, R.J., and Noack, B.R. Sensor-based estimation of the velocity in the wake of a low-aspect-ratio pyramid. *Experiments in Fluids*, 56(1):13, 2015.

Ivakhnenko, A. G. and Lapa, V. G. *Cybernetic Predicting Devices*. CCM Informatition Coporation, 1965.

Iyer, S. R., An, U., and Subramanian, L. Forecasting sparse traffic congestion patterns using message-passing rnns. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3772–3776, 2020.

Jacobs, E.N., Ward, K.E., and Pinkerton, R.M. The characteristics of 78 related airfoil sections from tests in the variable-density wind tunnel. *NACA Technical Report*, (460), 1933.

Jayaraman, B., Al Mamun, S.M., and Lu, C. Interplay of sensor quantity, placement and system dimension in pod-based sparse reconstruction of fluid flows. *Fluids*, 4(2):109, 2019.

Kamenetskiy, D.S., Bussoletti, J.E., Hilmes, C.L., Venkatakrishnan, V., Wigton, L.B., and Johnson, F.T. Numerical Evidence of Multiple Solutions for the Reynolds-Averaged Navier–Stokes Equations. *AIAA Journal*, 52 (8):1686–1698, 2014.

Kingma, D.P. and Ba, J. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015.

Kingma, D.P. and Welling, M. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2014.

Lasagna, D., Orazi, M., and Iuso, G. Multi-time delay, multi-point linear stochastic estimation of a cavity shear layer velocity from wall-pressure measurements. *Physics of Fluids*, 25(1):017101, 2013.

Lehmkuhl, O., Baez, A., Rodriguez, I., and Perez-Segarra, C.D. Direct numerical simulation and large-eddy simulations of the turbulent flow around a NACA-0012 airfoil. *7th International Conference on Computational Heat andMass Transfer 409*, 2011.

Liu, D. and Nishino, T. Numerical analysis on the oscillation of stall cells over a NACA 0012 aerofoil. *Computers and Fluids*, 175:246–259, 2018.

Liu, Y., Zhang, G., and Xu, B. Compressive sparse principal component analysis for process supervisory monitoring and fault detection. *Journal of Process Control*, 50: 1–10, 2017.

Liu, Y., Chu, X., Wang, W., and Weigand, B. Large-eddy simulation, convective instability, and modal causality of coaxial supersonic air-water jets considering a swirl effect. *Physics of Fluids*, 35 (6), 2023.

Manni, L., Nishino, T., and Delafin, P.L. Numerical study of airfoil stall cells using a very wide computational domain. *Computers and Fluids*, 140:260–269, 2016.

Manohar, K., Brunton, B.W., J.N., Kutz, and Brunton, S.L. Data-Driven Sparse Sensor Placement for Reconstruction: Demonstrating the Benefits of Exploiting Known Patterns. *IEEE Control Systems Magazine*, 38 (3):63–86, 2018.

Manolesos, M. and Papadakis, G. Investigation of the three-dimensional flow past a flatback wind turbine airfoil at high angles of attack. *Physics of Fluids*, 33, 2021.

Manolesos, M. and Voutsinas, S.G. Geometrical characterization of stall cells on rectangular wings. *Wind Energy*, 17 (9):1301–1314, 2014.

Manolesos, M., Papadakis, G., and Voutsinas, S.G. An experimental and numerical investigation on the formation of stall-cells on airfoils. *Journal of Physics: Conference Series*, 555 (1), 2014a.

Manolesos, M., Papadakis, G., and Voutsinas, S.G. Experimental and computational analysis of stall cells on rectangular wings. *Wind Energy*, 17 (6):939–955, 2014b.

McCulloch, W.S. and Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943. .

McCullough, G.B. and Gault, D.E. Examples of three representative types of airfoil-section stall at low speed. *NACA Technical Note*, (2502), 1951.

Milano, M. and Koumoutsakos, P. Neural network modeling for near wall turbulent flow. *Journal of Computational Physics*, 182(1):1–26, 2002.

Moss, G.F. and Murdin, P.M. Two-Dimensional Low-Speed Tunnel Tests on the NACA 00 I2 Section Including Measurements Made During Pitching Oscillations at the Stall. *Aeronautical Research Council: Current Papers*, (1145), 1968.

Mulleners, K., Henning, A., and Raffel, M. Investigations of Trailing Edge Stall on 2D Airfoils. *14th Int Symp on Applications of Laser Techniques to Fluid Mechanics*, 2008.

Nair, N.J. and Goza, A. Leveraging reduced-order models for state estimation using deep learning. *Journal of Fluid Mechanics*, 897, 2020.

Neves, A. F., Lawson, N. J., Bennett, C. J., Khanal, B., and Hoff, R. I. Unsteady aerodynamics analysis and modelling of a slingsby firefly aircraft: Detached-eddy simulation model and flight test validation. *Aerospace Science and Technology*, 106:106179, 2020.

Nuttall, A.H. Some Windows with Very Good Sidelobe Behaviour. *IEEE transactions on acoustics, speech, and signal processing*, 29 (1):84–91, 1981.

Omata, N. and Shirayama, S. A novel method of low-dimensional representation for temporal behavior of flow fields using deep autoencoder. *AIP Advances*, 9 (1), 2019.

Otto, S.E. and Rowley, C.W. Inadequacy of linear methods for minimal sensor placement and feature selection in nonlinear systems; a new approach using secants. *Journal of Nonlinear Science*, 32 (69), 2022.

Payne, F.R. and Lumley, J.L. Large eddy structure of the turbulent wake behind a circular cylinder. *Physics of Fluids*, 10 (9):S194–S196, 1967.

Picard, C. and Delville, J. Pressure velocity coupling in a subsonic round jet. *International Journal of Heat and Fluid Flow*, 21(3):359–364, 2000.

Plante, F., Laurendeau, E., and Dandois, F. Stall Cell Prediction Using a Lifting-Surface Model. *AIAA Journal*, 60 (1):213 – 223, 2022.

Raiola, M., Discetti, S., and Ianiro, A. On piv random error minimization with optimal pod-based low-order reconstruction. *Experiments in Fluids*, 56(4):75, 2015.

Raissi, M., Perdikaris, P., and Karniadakis, G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Roman-Rangel, E., Odobez, J.M., and Gatica-Perez, D. Assessing sparse coding methods for contextual shape indexing of maya hieroglyphs. *Journal of Multimedia*, 7: 179–192, 2012.

Saegusa, R., Sakano, H., and S., Hashimoto. Nonlinear principal component analysis to preserve the order of principal components. *Neurocomputing*, 61, 2004.

Saetta, E., Tognaccini, R., and Iaccarino, G. Machine learning to predict aerodynamic stall. *International Journal of Computational Fluid Dynamics*, 36 (7), 2022.

Schewe, G. Reynolds-number effects in flow around more-or-less bluff bodies. *Journal of Wind Engineering and Industrial Aerodynamics*, 89 (14-15):1267–1289, 2001.

Simpson, R.L. Aspects of turbulent boundary-layer separation. *Progress in Aerospace Sciences*, 32 (5), 1996.

Sirovich, L. Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly Applied Mathematics*, 45 (3):561–571, 1987a.

Sirovich, L. Turbulence and the dynamics of coherent structures. iii. dynamics and scaling. *Quarterly Applied Mathematics*, 45 (3):583–590, 1987b.

Spalart, P. R. Prediction of Lift Cells for Stalling Wings by Lifting-Line Theory. *AIAA Journal*, 52 (8), 2014.

Steinfurth, B., Cura, C., Gehring, J., and Weiss, J. Tuft deflection velocimetry: a simple method to extract quantitative flow field information. *Experiments in Fluids*, 61 (146), 2020.

Sun, L. and Wang, J.X. Physics-constrained bayesian neural network for fluid flow reconstruction with sparse and noisy data. *Theoretical and Applied Mechanics Letters*, 10 (3):161–169, 2020.

Swytink-Binnema, N. and Johnson, D.A. Digital tuft analysis of stall on operation wind turbines. *Wind Energy*, 19:703–715, 2016.

Thomas, A.J., Petridis, M., Walters, S.D., Gheytassi, S.M., and Morgan, R.E. Two hidden layers are usually better than one. *Communications in Computer and Information Science: Engineering Applications of Neural Networks*, 744, 2017.

Van Loan, C.F. and Golub, G.H. *Matrix computations*. Johns Hopkins University Press Baltimore, 1983.

Weihs, D. and Katz, J. Cellular patterns in poststall flow over unswept wings. *AIAA Journal*, 21 (12):1757–1759, 1983.

Willcox, K. Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Computers & fluids*, 35 (2):208–226, 2006.

Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., and Ma, Y. Robust face recognition via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 31 (2):210–227, 2008.

Yang, Y., Li, R., Zhang, Y., and Chen, H. Flowfield prediction of airfoil off-design conditions based on a modified variational autoencoder. *AIAA Journal*, 60 (10), 2022.

Yildirim, B., Chryssostomidis, C., and Karniadakis, G.E. Efficient sensor placement for ocean measurements using low-dimensional concepts. *Ocean Modelling*, 27 (3-4): 160–173, 2009.

Yon, S. A. and Katz, J. Study of the Unsteady Flow Features on a Stalled Wing. *AIAA Journal*, 36 (3):305–312, 1998.

Yonekura, K. and Suzuki, K. Data-driven design exploration method using conditional variational autoencoder for airfoil design. *Structural and Multidisciplinary Optimization*, 64:613–624, 2021.

Zarutskaya, T. and Arieli, R. On Vortical Flow Structures at Wing Stall and Beyond. *AIAA Fluid Dynamics Conference and Exhibit*, 35, 2005.

Zhang, B. Airfoil-based convolutional autoencoder and long short-term memory neural network for predicting coherent structures evolution around an airfoil. *Computers and Fluids*, 258, 2023.

Zhou, K., Zhou, L., Zhao, S., Qiang, X., Liu, Y., and Wen, X. Data-driven method for flow sensing of aerodynamic parameters using distributed pressure measurements. *AIAA Journal*, 59 (9), 2021.