A Direct Comparison of Small Aircraft Dynamics between Wind Tunnel and Flight Tests

Kai Lehmkühler

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy

Faculty of Engineering - The University of Sydney - 2017
Statement of Originality
This is to certify that the content of this thesis is my own work. This thesis has not been submitted for any degree or other purposes. I certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged.

Previous Publications
The text and the images of part IV on the determination of the inertial properties of the test aircraft has been published in the ‘Aeronautical Journal’, managed by Cambridge University Press on behalf of the Royal Aeronautical Society [1]. The text of the paper is reproduced here with minor changes to fit into the overall thesis.

Special Thanks to
Matt, KC, Dries, Kim, Dan and JAG, as well as my family and my friends outside the academic world.

Copyright © 2017 Kai Lehmkühler

Final Version. February 2017
Abstract

The miniaturization of embedded electronics and sensors driven by the rapid development of mobile devices has enabled powerful avionics systems for very small aircraft. This enables a potential step forward in accurate flight data gathering for vehicles weighing 5 kg or less. Being able to flight test a small platform like this also allows the comparison of the results with reference data from ground testing in a standard sized wind tunnel of an identical airframe. With this process, the following questions can be answered: Firstly, would such a system then be able to collect accurate flight data for system identification (ID)? Is it possible at all to fly a small, remotely piloted aircraft precisely enough to record the required data, given its sensitivity to atmospheric turbulence, airframe noise, limitations of the remote piloting and so on? And secondly, if accurate data has been obtained, how well do the two experiments match? The small scale might potentially result in previously unknown or at least insignificant physical phenomena, which need to be taken into account when flight testing such a small platform. The changes in the inertial properties of the platform due to the added mass effect is one of these phenomena, which can typically be ignored for full scale aircraft. However, this has proven to be critically important for the successful analysis and comparison of the flight- and wind tunnel data obtained throughout this project.

The avionics suite designed for this research was developed in house, since the weight restrictions of the small platform excluded any commercially available flight data recording packages. The suite features an lightweight airdata probe, control surface feedback sensors, a custom designed GPS receiver and many other advanced components previously not possible at this scale. A commercial reference INS was used to benchmark the system. The UAVmainframe also provides basic flight control functionality to aid the pilot in obtaining the required trim conditions and turbulence mitigation. Extensive data compatibility analysis and calibrations were performed on the recorded data using an Extended Kalman Filter (EKF) and various other methods to ensure the best possible data quality.

The inertial properties of the test aircraft were determined by swing tests. The significance of the added mass contributions was discovered during these tests, which added up to 25% onto the ‘true’ airframe inertial properties. In an effort to estimate these added mass terms, it has been found that the methods presented in literature to determine the corrections for full scale aircraft do not give the correct results for the small scale aircraft under consideration. Swing tests of a flat plate model of the test aircraft also did not capture the magnitude of the phenomenon correctly, which led to swing tests with a geometrically similar 3-d object of known inertial properties to successfully estimate the added mass corrections.

Static derivatives were obtained from conventional wind tunnel testing, in conjunction with a high fidelity three dimensional inviscid solution using the PanAir code. A dynamic test rig was used in the wind tunnel to determine the dynamic derivatives. It allowed the instrumented airframe to rotate freely on a three axis gimbal, essentially ‘fly’ in the tunnel. The aerodynamic derivatives from these 3 DoF tests were estimated by performing system ID on the recorded data, where the model structures were modified for the reduced set of motion variables.
Extensive flight testing was performed at the university's flight test centre. These tests showed the difficulty of testing such a small and light airframe due to wind and airframe noise, as well as the limitations due to lack of feedback received by the remote pilot. The pilot was aided by the flight control system to achieve a good trim condition, and pre-recorded input sequences, similar to the dynamic wind tunnel tests, were used to excite the longitudinal and lateral dynamics of the aircraft. One particular finding during the test campaign was that there is no such thing as totally calm conditions for this scale of airframe. Other findings include a high correlation between the pitch damping term and the pitching moment due to elevator, making it impossible to determine both at the same time, and that in flight the inertial properties of the test aircraft change to the values that include the added mass components, as compared to the dynamic wind tunnel tests, where the ‘true’ inertias are used. By including these findings in the data processing, close agreement between flight and ground test data has been achieved.
## Contents

### I  Introduction

1  Background and Research Proposal .................. 3
  1.1 Introduction ..................................... 3
  1.2 Literature Review ................................ 4
  1.3 Initial Research Proposal ......................... 6
  1.4 Initial Research Plan .............................. 7
  1.5 Revised Research Proposal ......................... 8
  1.6 Background on Added Mass ....................... 9
  1.7 Thesis Structure .................................. 11
  1.8 Airframe Overview ................................ 13
  1.9 Mathematical Background ......................... 16

### II  The UAVmainframe

2  Motivation ............................................ 23
  2.1 Introduction ....................................... 23
  2.2 Market Review ..................................... 26
  2.3 The UAVmainframe Concept ....................... 27
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Lateral Combined Manoeuvre</td>
<td>195</td>
</tr>
<tr>
<td>13.1</td>
<td>Model Structure</td>
<td>195</td>
</tr>
<tr>
<td>13.2</td>
<td>Input Design and Verification</td>
<td>196</td>
</tr>
<tr>
<td>13.3</td>
<td>Results</td>
<td>200</td>
</tr>
<tr>
<td>13.4</td>
<td>Lateral Combined Summary</td>
<td>207</td>
</tr>
<tr>
<td>VI</td>
<td>Reference Data: Collected Results</td>
<td></td>
</tr>
<tr>
<td>VI.1</td>
<td>Reference Data Summary</td>
<td>211</td>
</tr>
<tr>
<td>VII</td>
<td>Flight Testing</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Flight Operations</td>
<td>217</td>
</tr>
<tr>
<td>14.1</td>
<td>Requirements</td>
<td>217</td>
</tr>
<tr>
<td>14.2</td>
<td>A Typical Research Flight</td>
<td>218</td>
</tr>
<tr>
<td>14.3</td>
<td>Lessons Learned</td>
<td>221</td>
</tr>
<tr>
<td>14.4</td>
<td>Summary</td>
<td>228</td>
</tr>
<tr>
<td>15</td>
<td>Flight Data Preparation</td>
<td>229</td>
</tr>
<tr>
<td>15.1</td>
<td>Data Compatibility Check</td>
<td>229</td>
</tr>
<tr>
<td>15.2</td>
<td>Properties of Selected Flight Data Sets</td>
<td>231</td>
</tr>
<tr>
<td>15.3</td>
<td>Summary</td>
<td>241</td>
</tr>
<tr>
<td>16</td>
<td>Longitudinal Results</td>
<td>243</td>
</tr>
<tr>
<td>16.1</td>
<td>Introduction</td>
<td>243</td>
</tr>
<tr>
<td>16.2</td>
<td>Model Structure</td>
<td>244</td>
</tr>
<tr>
<td>16.3</td>
<td>Input Design and Verification</td>
<td>246</td>
</tr>
<tr>
<td>16.4</td>
<td>Multiple Input Results</td>
<td>253</td>
</tr>
<tr>
<td>16.5</td>
<td>Summary</td>
<td>262</td>
</tr>
<tr>
<td>17</td>
<td>Lateral Flight Results</td>
<td>263</td>
</tr>
<tr>
<td>17.1</td>
<td>Dutch Roll Mode</td>
<td>264</td>
</tr>
<tr>
<td>17.2</td>
<td>Roll Mode</td>
<td>268</td>
</tr>
<tr>
<td>17.3</td>
<td>Lateral Combined Manoeuvre</td>
<td>272</td>
</tr>
<tr>
<td>17.4</td>
<td>Summary</td>
<td>277</td>
</tr>
</tbody>
</table>
Future Work and Conclusion

18 Future Work

18.1 The Mysterious Nature of Added Mass
18.2 Sensors and Equipment
18.3 Experiment Design

19 Conclusion

References

Appendices

A The 7x5 Feet Low Speed Wind Tunnel
A.1 Facility Overview
A.2 Test Section Survey

B Wind Tunnel Wall Corrections
B.1 Balance Calibrations
B.2 Blockage, Tares and Interference
B.3 Wall Interference

C Virtual Wind Tunnel
C.1 Solver Selection
C.2 Workflow
C.3 Modelling the Wind Tunnel Environment
C.4 Aircraft Models
C.5 Simulation Benchmarking
C.6 Numerical Wind Tunnel Wall Corrections
C.7 Summary

D AVL Dynamic Derivatives

E Tailplane Downwash Coefficient

F Comment on Hobby Grade Equipment
1 Background and Research Proposal . . . 3
1.1 Introduction
1.2 Literature Review
1.3 Initial Research Proposal
1.4 Initial Research Plan
1.5 Revised Research Proposal
1.6 Background on Added Mass
1.7 Thesis Structure
1.8 Airframe Overview
1.9 Mathematical Background
1. Background and Research Proposal

1.1 Introduction

Aircraft flight testing is one of the most difficult, but also most exciting disciplines in engineering. It is the flight test engineers and pilots who take a new design to the air for the first time to evaluate the flight performance and handling qualities of the new aircraft. The process of flight testing finally relates all the engineering work during the design phase with the response to the physical airframe during flight, and is therefore a very good activity for understanding the physics of flight. Unfortunately, up until recently, this activity was limited to a small group of selected individuals, who would qualify to flight test a full scale aircraft. Since this is a expensive endeavour, only the absolutely necessary amount of flight testing was and is done for full scale aircraft.

With the advent of small scale unmanned aircraft (UAV) and the mind boggling progress in miniaturisation of electronic systems, it is now possible to perform a flight test programme for university research at a fraction of the previous cost and manpower necessary. What was a supercomputer 15 years ago can now be mounted into a 4 kg remotely piloted aircraft to record and process data in flight in real time. This progress allows to measure the dynamic properties of such a small aircraft in flight and to compare the data to other sources like wind tunnel testing or computer simulations for the purpose of education, design verification and many more, making aircraft flight testing much more accessible to anyone interested in the field.

While it is true that one can get an aircraft instrumented with basic sensors and flying for less than $1000, the question is whether the recorded data will be correct and whether the results are comparable to available reference data. There are many factors that need to be considered for successful flight testing, including the quality of the data acquisition system, disturbances from atmospheric turbulence and engine
vibrations, both potentially much more severe for a small scale airframe. Furthermore, how to generate good inputs to excite the modes of motion of the aircraft while standing hundreds of metres away? And, most interestingly, what steps are required to be able to compare wind tunnel data to the flight test results? Maybe the dynamics of flight of a small platform are changed by some effect due to the small scale, like the very low Reynolds numbers or the tiny inertial properties of such a small platform? \(^1\)

All these issues need careful consideration to generate confidence in the flight test results. But, as will be discussed in the next section, there is only a very small body of literature on small aircraft flight testing available, and none of these publications demonstrates the direct comparison between ground- and flight test data. Hence, in order to advance the knowledge in this field, in this thesis a small, fixed wing aircraft of standard configuration will be extensively tested in the wind tunnel and an identical airframe will be flight tested, using a custom designed high performance avionics system that takes full advantage of the possibilities of miniaturised sensor and computing devices. Before going into more detail, a review of other flight test projects with small scale aircraft is given in the following section.

### 1.2 Literature Review

A limited number of publications on flight testing of small scale aircraft is available. Most, however, simply present their system and a general overview of the work, while no mention is made if their systems actually work and no results are presented [2, 3, 4, 5, 6, 7]. None of these publications report on any calibrations or data compatibility checks nor do they present flight data.

A few notable exceptions are discussed next. Reference [8] reports on flight tests of a blended wing body airframe with 3.3m span and 25kg MTOW. The publication only lists preliminary data that does not match the reference data well, as noted by the authors. No information on sensor calibration or data compatibility is given. No follow up publication of this project is available, so it remains unclear if the results could be improved and why only a poor match was achieved in the first place.

Two papers are available on flight test efforts at the Georgia Institute of Technology [9, 10]. The first reference reports on a small scale aircraft of unknown dimensions for educational purposes. It contains some useful information on system calibration and sensor fusion, as well as some limited flight data. The longitudinal results match their reference data of unknown origin reasonably well, while the lateral results have limited accuracy. The second reference is on real time system identification of a very small aircraft with 1.2m span and 1.5 kg MTOW. No details about calibrations or data compatibility are given, but the flight data is identified with reasonable accuracy for this small scale aircraft. Verification is done only by prediction tests of other flight data and no independent reference data is used.

The largest body of literature covers the NASA project AIRSTAR [11]. As high budget project it covers the flight test operations of a dynamically scaled jet transport with 2m

---

\(^1\)This is symbolised by the chapter title image, which shows the typical research student, including this author, at the start of the project: Very keen, but utterly clueless about what he/she got himself/herself into...
1.2 Literature Review

wingspan and 20kg MTOW. Reference [11] covers the development of the airframe and is the only reference available that describes tests for the added mass contribution to the inertial properties. In the reference, only a flat plate representation of the aircraft is used and the reported added mass contribution is small. As shown later in this work, only a fully three-dimensional model fully captures the added mass contribution, which may have lead to inaccurate inertia estimates for the AIRSTAR airframe. An indication is the table of flight data presented in reference [12], where the match between the pitch stability derivative $C_{m_{\alpha}}$ between wind tunnel and flight data is off by 35%, a similar order of magnitude as discovered during this project before correctly including the added mass effect. Unfortunately, there is no publication of a direct benchmark between flight and ground test data for the AIRSTAR project, so this could not be investigated further. Most references on this project deal with developing new methods of system identification [13, 14, 15, 16], and give only limited comparisons with reference data.

Another NASA project is FASER [17]. This work is concerned with an aircraft of similar size to this project. The publication reports on the system development and as a rare exception mentions the data compatibility requirement. Reference [18] reports on some dynamic wind tunnel tests of the airframe, but no flight data is available.

A corresponding effort is the work of the UAV test group an the Uni of Minnesota [19], which uses the same aircraft as the NASA FASER project mentioned before. Their work appears to be of high quality, with reports of a sound system design and test procedures. However, the resulting match between the NASA provided reference data and the flight data is poor, as noted by the authors. A reason may be the use dimensional of derivatives, which are not valid for this class of vehicles due to the large variation of airspeed during manoeuvres, as will be shown later in this thesis.

The recent PTERA aircraft [20] is a promising effort, but it requires much more work, as the authors state themselves. Some preliminary flight data were presented, but only with rough estimates of the inertial properties of the airframe. Also, no benchmark for the reference data is given. This aircraft was substantially bigger (4m span and 100kg MTOW) than the size of aircraft under consideration for this project.

Another study, focussing mainly on control surface failures, is reported in reference [21]. The aircraft size is again considerably larger with 4.8m span and 37kg MTOW. Some flight test results are given and are compared to DATCOM and AAA, showing poor agreement, which is probably caused by the limitations of the methods, rather than the flight test data. However, no information on calibration or data compatibility is given.

In summary, there is no publication, or set of publications, that reports on the full cycle of generating high quality reference data and performing the required steps to compare that data to high quality flight data for a single, small scale airframe to directly compare the resulting accuracy and to identify any issues that may arise. Since this is required to gain confidence in the data acquisition system, the flight test procedures and the data analysis, the following initial research proposal was formulated to address this gap in the literature.
1.3 Initial Research Proposal

The initial research proposal was a very general statement that reflects the identified gap in the literature discussed in the previous section:

*Do the latest advances in miniature embedded computing and sensor technology enable the collection of high quality flight test data from a very small, fixed wing aircraft? What is required to design such a system for accurate, reliable and safe operation and how can it be benchmarked? What are the limitations of such a small flight platform? And finally, what data quality can be achieved?*

To achieve the research objectives the following steps will be required:

1. Development of a sensor system suitable for the given aircraft scale that can perform the required measurements accurately and reliably
2. Develop flight test methods and procedures to collect the necessary data safely and efficiently, identify limitations and issues that might restrict the accuracy of the measured data.
3. Perform suitable experiments / computations for benchmarking of the data

To meet these research objectives, several success criteria have been defined. The list is ordered by increasing accuracy and therefore usefulness of the flight data but also by increasing difficulty as the requirements for successful completion of the respective topic become more demanding:

1. Ability to predict independent flight manoeuvre from flight data alone;
2. Match of mode frequency and damping characteristics between experiments;
3. Match of aerodynamic derivatives between experiments; and
4. Ability to use a 6 DoF flight simulation to predict aircraft dynamic motion.

The ability to predict independent flight manoeuvres from flight data alone is a standard method of testing system identification methods [22]. Here, the identified system model is used to predict an independently recorded data series by time-integrating the model from known initial values using the control inputs of the independent manoeuvre and comparing the result to the actual response of the aircraft. A close match validates the identified system and its model structure. But the data may well still be corrupted by sensor- and other systematic errors present in both data sets. Therefore this method is not suitable to state that the identified system is representing the aircraft motion, it only represents the aircraft motion as recorded by the sensor system.

Using an independent experiment such as wind tunnel testing gives a data set that is (ideally) not affected by the sensor errors mentioned above. Matching the flight data with such an experiment increases the confidence in the data quality dramatically. For this work two different stages are used for this benchmark. In the first step, only the flight mode characteristics (natural frequency and damping) are compared. As will be discussed later, these key figures are much more insensitive to errors than the aerodynamic derivatives, which will be used for comparison in the second step.

Finally, feeding all results into a full 6 DoF flight simulation code, it will be tested if this simulation is accurate enough to predict the aircraft response to a given input. This
is similar to the first method with the big difference that the response is computed completely independent of the flight data processing step as only the identified aerodynamic derivatives and mass properties are used for the simulation. If this step is completed successfully, the data would then be good enough to be used for control system design, flight simulation and other purposes.

1.4 Initial Research Plan

The test aircraft was chosen from a list of commercially available model aircraft such that it would fit into the University of Sydney’s 7x5ft wind tunnel to obtain reference data of an identical airframe to avoid scaling issues. This limited the maximum span to 1.5m to keep the necessary wall corrections at a reasonable level. An aircraft with a maximum span of 1.5m has a typical MTOW of 3-5kg, with an available payload capacity of 500g or less. Since no suitable avionics and flight data acquisition system was (and still is) commercially available for this scale, this system had to be developed from scratch. The design and implementation is described in Part II.

Flight test methods and procedures for an aircraft of this scale are very different from a full scale aircraft due to the remote piloting, as described in Part VII. Some of the previously mentioned references included some best practises and advice that was included in the development of the procedures. Through iterative tests and improvements, suitable methods have been found and implemented, as discussed in Part VII.

The reference data required for this project included all geometry, mass and inertial properties, as well as a full set of aerodynamic stability and control derivatives. Some of these derivatives can be obtained from standard static wind tunnel tests, but others cannot, since they depend on dynamic motion of the airframe. These derivatives can either be obtained from forced oscillation tests [18], or through a more recent development allowing to simulate 3 DoF flight in the wind tunnel [23]. The latter method is much simpler to implement, if a high quality sensor system is available that fits into the wind tunnel model. Since such a system was to be developed for the flight tests, this method promised to be able to obtain all but a few dynamic derivatives, while also allowing to test and verify the avionics system at the same time. Once the wind tunnel data was available, all required corrections for the presence of the walls had to be determined and benchmarked to enable a comparison with the flight data. Since such data was not available for the 7x5 ft. wind tunnel, the wind tunnel environment was simulated in the high order panel code PanAir to obtain these corrections. The inertial properties were estimated by swing tests, which is a standard method to determine these properties experimentally. During the literature review of publications on test methods for the inertial properties, this author was first introduced to the added mass phenomenon, which is generally not included in any aircraft related literature or teaching [24]. This phenomenon became more and more important during the process of this project and therefore it was decided to modify the research proposal to include the significance of added mass on the flight dynamics of small scale, fixed wing aircraft.
1.5 Revised Research Proposal

Even though the literature on swing tests for the determination of the inertial properties reports on the significance of added mass during the pendulum motion, this contribution was initially not included into the measured inertias of the test aircraft, since reference [24] states that the added mass is only important for vehicles where the displaced surrounding air mass is similar to the weight of the vehicle, such as airships. This displaced mass ratio is similar for the test aircraft and the Boeing 747 used in the book. Hence, one would expect no significant contribution due to added mass for the small aircraft tested for this project. During preliminary testing of the methods, however, it was discovered that there was significant disagreement between the stability derivatives that could be obtained from both static and dynamic tests. This issue led to a revisit of the inertial testing methods, which are required for the data analysis of the dynamic wind tunnel tests. As discussed in Part IV, significant contributions of added mass to the inertial properties do occur, and require special treatment. Since the added mass is caused by accelerating the aircraft with respect to the surrounding air, the question arose then, whether the dynamics in flight would also be affected by the added mass phenomenon. This was also indicated by initial flight data, which showed large discrepancies to the ground test data. Since the added mass contributions are usually ignored for aircraft [24], only very limited knowledge about their effect on the flight characteristics of small aircraft is known. This, together with the magnitude of corrections required, lead to a re-formulated research proposal as follows:

Do the latest advances in miniature embedded computing and sensor technology enable the collection of high quality flight test data from a very small (≈ 4 kg), fixed wing aircraft? What is required to design such a system for accurate, reliable and safe operation and how to benchmark it? What are the limitations of such a small flight platform? Additionally, are the added mass contributions affecting flight and are they therefore identifiable in the flight data? And finally, can they be directly transferred from the swing tests with their quite different motion pattern?

The modified research proposal led to some additions in the work required and some shifts in priorities. More research into the added mass issue had to be done, and the added mass components of the test aircraft had to be determined as accurate as possible, which is rather difficult for a complex shape like a full airframe. The experimental method that was developed for this task is discussed in Part IV. The inclusion of the added mass effects will result in potentially different inertial properties between wind tunnel and flight tests. This makes is more difficult to compare the natural frequency and damping ratios of the modes of motion, since these are dimensional parameters depending on the inertias of the aircraft. Hence, less attention was spent on matching these properties and all effort was re-directed to the comparison of the aerodynamic derivatives. Before moving on to the organisation of the thesis, a brief introduction and review of the added mass contribution to the aircraft mass and inertial properties is added to the general literature review from above.
1.6 Background on Added Mass

There are three aerodynamic effects that can influence the mass- and inertial properties of a body:

1. buoyancy of the body in the surrounding air;
2. enclosed air mass in the (hollow) body; and
3. added mass due to acceleration of the body while immersed in a fluid

The influence of buoyancy and enclosed air mass actually results in a change in weight of the test article, where buoyancy reduces the measured weight and the enclosed air adds to the measured weight of the body, respectively. The added mass due to the inertia of the fluid accelerated by the bodies’ motion is caused by a momentum change of the surrounding fluid due to an acceleration of the pendulum. It has the same form as a mass term, hence the name [25].

For full scale aircraft, the corrections due to buoyancy typically amount to 3% of the measured inertias. The enclosed air adds around 5% in the X-axis and is negligible in the Y-axis. Another 20% are added in the X-axis due to the added mass phenomenon, while errors in Y and Z are about 5% [26]. For small scale fixed wing aircraft, the enclosed air mass and buoyancy are negligible, because the airframes only have small internal volumes and the volume of their structures is also small. Hence, the changes due to these two effects will only add the equivalent of a few grams of weight to a 2-5 kg airframe and therefore these two corrections can be neglected. The added mass correction, however, is very significant for small scale fixed wing aircraft and thus requires careful consideration. For example, the correction due to added mass in the X-axis is 25.1% for the aircraft used for this thesis.

Reference [27] from 1941 reports on a comparison between flight tests of a full scale aircraft and free flight tests of a model in the Langley free flight tunnel. In the conclusion, the authors report a similar change in longitudinal stability as observed in this project, but fail to reach the conclusion of the significance of the added mass on the small model.

But what is added mass? As explained by Brennan [25], a body moving through a fluid adds a certain amount of kinetic energy to the fluid. That kinetic energy $T$ can be written for steady and rectilinear motion as

$$T = \frac{1}{2} m_{\text{fluid}} V^2$$

or, if incompressible flow is assumed,

$$T = \frac{1}{2} \rho \zeta V^2$$

where

$$\zeta = \int_V \frac{v_x v_y v_z}{V^2} \, dV$$

and $V$ is the velocity of the body and the integral $\zeta$ is a measure of the volume of fluid affected by the motion of the body inside the entire fluid domain $V$. The resulting differences in velocity relative to $V$ in the flow field are denoted $v_i$. The product $\rho \zeta$ is then the mass of fluid affected by the motion of the object. The integral $\zeta$ is constant for constant velocity $V$.

When the body accelerates or decelerates, as it constantly does during pendulum motion or during a flight manoeuvre, the velocity $V$ of the body changes and with that the kinetic energy $T$ imparted on the fluid. This requires additional work to be done
by the body, which is simply $dT/dt$. The rate of work done can then be expressed as $F_D V$, where $F_D$ is an additional drag force. Assuming that $\zeta$ is constant, that is, the flow pattern does not change, the added drag, $F_D$, is

$$F_D = \frac{1}{V} \frac{dT}{dt} = \rho \zeta \frac{dV}{dt} = m \frac{dV}{dt}$$  \hspace{1cm} (1.3)$$

where the sign of the force depends on whether the body accelerates or decelerates. The added drag force $F_D$ has the same form and sign as a force required to accelerate or decelerate the mass $m$ of the body. Therefore, the term $\rho \zeta$ can be interpreted as an additional mass $m_f$ of fluid that is being accelerated or decelerated by the body. This added mass $m_f$ has an inertia about the axis of rotation and hence the inertia measured is

$$I_{meas} = I_{TA} + I_{m_f}$$  \hspace{1cm} (1.4)$$

It should be observed that this added drag force is different from the ‘conventional’ drag force, which is proportional to the square of the velocity of the body. The added drag described here is proportional to the acceleration of the body. Given the direct dependency on the fluid density $\rho$ and the high density of water, this effect is very important especially for hydrodynamic problems[25, 28]. However, it is also critical for the correct determination of airframe inertial properties as previously discovered [26]. Methods have previously been developed to experimentally estimate $I_{m_f}$ [26, 29, 30]. These methods are based on test data for flat plates and ellipsoids to model the airframe’s shape from these basic bodies. The corrections for the full test articles are then assumed to be the sum of the corrections for the separate components, ignoring potential interference effects between the parts. For full scale aeroplanes, this appears to work quite well, with resulting inertia estimates within 2.5% or less of the true value [29].

Applying the previously published methods to the given aircraft geometry, results in corrections that are an order of magnitude too small. It is unclear why this happens, since the datasets in [29, 30] are missing crucial information to reproduce the findings. Alternatively, Lin & Liao [28] handled the same problem using modern fluid-structure interaction solvers. Brennan [25] and Reference [31] introduce methods for estimating the corrections based on results of a potential flow solver. Both methods are very difficult to use in practise and were beyond the scope of this project. Instead, geometrically similar bodies with known inertial properties, together with additional swing tests, were used to determine the corrections due to $I_{m_f}$ for the given aircraft. The potential flow method is an interesting topic for future work, where the PanAir solver results from appendix C may potentially be used to achieve similar results as in the references. This, however, requires changes to the PanAir output to obtain the required data, which is not a straight forward task.

For this thesis, the added mass matrix $I_{m_f}$ is assumed to be a simple, additive 3x3 matrix with only three terms on the main diagonal, which is in line with the literature on the added mass contributions for aircraft cited above. During presentation of the findings in the upcoming chapters it will become more and more clear that this assumption is not fully valid. Yet, it gives reasonable results and shows the need of awareness in the
1.7 Thesis Structure

Anyone familiar with an experimental project like this will appreciate that the work takes place over several iterations until the required data quality is achieved and the research objectives can be met. This was no different for this work and it is not easy to present it in a linear manner, since some of the work is motivated by the findings of previous iterations. Therefore, in this presentation of the thesis structure the interactions between the different parts will be included to explain some of the references between the following parts of the thesis. The remainder of this part will present information on the selected test airframe and the mathematical background that will be used throughout. To better comply with the required word limit, some aspects of the work were moved into the appendix and only the results of these chapters is used in the main text. Yet, especially the chapter on the numerical simulation of the wind tunnel environment represent considerable work, which should be of interest to the reader.

Part II discusses the development of the UAVmainframe, as the newly developed avionics system is called. After presenting the concept for the hard- and software, the implementation of the system is introduced. This is followed by a chapter on the software that is used for the data analysis, including a brief introduction to the system identification methods that are used for this thesis. Significant work was spent on an Extended Kalman Filter (EKF) for sensor error and data compatibility analysis to obtain high quality data from the system. Results of the filter runs are discussed later in Part VII on the flight data analysis to achieve better flow of the presentation. The part concludes with a discussion of all sensor calibration methods that were used.

Part III presents the work on the static wind tunnel tests. All static stability and control derivatives were measured on an newly designed balance. The data was then corrected for the wall interference effects, using correction factors obtained from a model of the wind tunnel environment in the PanAir solver. The results of this part were then used as benchmark during the dynamic wind tunnel tests and the flight tests. Appendix A contains description of the all-new wind tunnel balance that was designed and constructed for this project. A survey of the wind tunnel data quality follows. Appendix B and C contain a large block of work on developing the wind tunnel wall corrections using two independent methods. First, the well known correction method,
based on a horseshoe vortex and the method of images, that can be found in several reference books (termed classical method in this work) is applied to the given problem. Secondly, the wind tunnel environment is modelled in the high order panel code PanAir to verify and improve on the previously obtained correction factors. PanAir models the true geometry of the test aircraft in the tunnel test section and is therefore expected to yield more accurate correction factors than the more basic classical methods.

The following parts IV and V discuss the inertial property measurements and the dynamic wind tunnel tests. Both tasks are presented on their own, but they are tightly interconnected and also make use of the UAVmainframe system and the static test results from Part III to demonstrate how the added mass components of the test airframe were first discovered and then determined. The inertial properties are measured with two separate swing tests. Initially, the added mass contribution was estimated using a flat plate simulator of the aircraft, as done in [11]. When these results were used in the system ID process of the motion data in part V, and the results were then compared to the corresponding static test results, they did not match by a large margin. During further iterative testing, it was then discovered that the added mass components can only be measured accurately with a simulator of similar volume and surface area as the test aircraft.

The identification of the dynamic derivatives from the wind tunnel motion experiment then uses the corrected inertial properties from Part IV and the static test data from Part III to estimate all possible longitudinal and lateral parameters from the 3-DoF response of the aircraft in the wind tunnel. The three most often used system ID methods, the equation error (EQN), the output error (OEM) and the filter error method (FEM) are all used and compared during the identification of the aerodynamic model parameters to establish their performance. Several problems with data correlation and identifiability due to the small scale are discussed.

Part VI is a summary of all reference data collected so far to present a short and comprehensive data set for comparison and benchmarking of the flight test data.

Part VII then discusses the flight test operations and the data processing, starting with the issues of the flight operations of a small scale aircraft and the solutions used for this project are presented. This is followed by the results of the data compatibility checks using the EKF, as well as some general remarks on the flight data quality that was achieved. The part ends with the system identification results of the flight data, which clearly show the importance of the added mass contributions to the aircraft inertial properties in flight. All three system ID methods are compared as before. Since the added mass contributions are most visible in the longitudinal data, most time and effort was spent on this topic. The inputs and the flight procedures for the lateral axis are therefore not yet as optimised as the longitudinal axis and only some preliminary data is available at the time of writing, which is presented in appendix 17. Even this preliminary data, however, shows clearly that the estimated added mass components are correct for all three axes of motion.

The thesis ends with suggestions for future work and a summary of all findings in Part VIII, which also includes an introduction to the more general theory of added mass
used to design ships, submarines and also airships to illustrate the need for further research into the added mass phenomenon for small, fixed wing aircraft.

1.8 Airframe Overview

The aircraft chosen for this research is a (at the time) commercially available model aircraft kit of a standard, low wing general aviation aircraft configuration. It was selected for its standard configuration, the 1.5m wingspan compatible with the University’s 7x5 ft. wind tunnel and the large fuselage with ample room for all the required equipment. The standard configuration ensures stable, benign and predictable flying characteristics, which are important for the development of the flight test methods and procedures. The compatibility with the wind tunnel allows for identical airframes to be tested in flight and on the ground, which removes all issues with scaling. And the roomy fuselage was perfect for all the additional electronic circuit boards and a large battery for long flight durations.

In this section a 3-view of the aircraft is shown, together with some general information. Detailed measurements and other information is given throughout the document, where required. The sensor installation is covered in Part II. The aircraft are made from laser-cut plywood and balsa wood structures, which is covered with a heat shrinking film for lightweight construction.

Significant modifications from the factory standard were necessary to improve flight safety and robustness of the aircraft. The tailplane was replaced by a foam cut version with additional fibreglass strengthening and a multi-hinged full span elevator. The vertical fin was reinforced with a carbon rod and new balsa wood skins to remove the weakness of the original structure. The main canopy was replaced by a custom made version of the same shape but only half the weight. The scaled interior of the original kit was completely removed to save weight and free up space.

All servo motors used are metal gear, high speed servos to obtain high speed control surface motion for the required control inputs. The propulsion system is a high voltage, low current system, running off a five-cell LiPo battery with 4200mAh capacity to drive a large 14x7” propeller. This highly efficient system allowed a maximum flight time of nearly 20 minutes with a new battery. The final flying weight of the flight model was 4100 grams, while the wind tunnel model was slightly lighter, since no landing gear was installed on that airframe.

Figure 1.1 shows a top view of the wind tunnel airframe with the main canopy and engine cover removed. Listed are the reference quantities used, as well as the three CG locations used during the project.

The wing of the wind tunnel model is specific for the ground test, since it was modified to hold the tunnel mount and the motion gimbal, as discussed in Part V. The fuselages are in principal exchangeable between the two aircraft, but this was done only once to confirm that power effects were insignificant in the wind tunnel (Section 7.4). The engine in the wind tunnel model is just a dummy to account for the weight. No powered tests were done with the wind tunnel airframe.
Figure 1.1 shows the side and front views with the fuselage reference line and the coordinate system origin. The wing dihedral shown in the front view is approximately 3 degrees and the aerofoil is 16% thick. The exact wing section is unknown, since the documented NACA2416 aerofoil is notably different than the actual wing section of the aircraft. Since the wing section geometry does only have a minimal effect on the stability and control derivatives, no attempt was made to measure the true aerofoil geometry. The coordinate origin on the table surface made it easy to determine all z-dimensions from a common reference, a task that otherwise would be very difficult.

The aircraft has proven quite sturdy and has survived several accidents during the
flight tests. It was tested up to 4.5g in the wind tunnel without any problem. The wooden structure made it easy to repair damage and to install all the custom equipment.

Figure 1.2: Front- and side view of the wind tunnel test airframe with reference axes
1.9 Mathematical Background

The mathematical background for this work are the standard equations of motion for a rigid body undergoing six degree of freedom motion. The standard symbols, reference frames and the derivation of these equations can be found in any textbook on flight mechanics, for example [22, 38, 39], and will not be repeated here. In this thesis, the widely used conventions for reference frames and axes definitions as shown in Figure 1.3, are used in all equations. There:

\[ X, Y, Z = \text{Body axes coordinates} \]

\[ L, M, N = \text{Moments about body axes} \]

\[ u, v, w = \text{Body axes velocities} \]

\[ p, q, r = \text{Rotation rates about body axes} \]

\[ \delta e, \delta a, \delta r = \text{Control surface deflections} \]

\[ V, \alpha, \beta = \text{Airspeed, angle of attack, angle of sideslip} \]

Specialised derivations will be done where required across the thesis. Since it was found during this project that on this small scale, wind is nearly always present and needs to be accounted for, the following sections state the assumptions used to generate the process model for the Kalman filter and the model formulations for the system identification tasks.

Figure 1.3: Aircraft body axes with definitions
1.9 Mathematical Background

1.9.1 Assumptions

The following general assumptions are used throughout the thesis. These are generally accepted to be valid for flight dynamic problems, where manoeuvre duration and flight distances involved are short [22, 39]:

<table>
<thead>
<tr>
<th>General Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The aircraft is a rigid body</td>
</tr>
<tr>
<td>2. Weight and inertias are constant</td>
</tr>
<tr>
<td>3. The aircraft is considered symmetric about the XZ plane of the body axes</td>
</tr>
<tr>
<td>4. The earth is fixed in inertial space</td>
</tr>
<tr>
<td>5. The earth’s surface can be approximated as flat</td>
</tr>
<tr>
<td>6. Uniform gravitational acceleration</td>
</tr>
<tr>
<td>7. Thrust forces are along the x body axis only</td>
</tr>
<tr>
<td>8. Gyroscopic moments due to the propulsion system are negligible (verified by calculations)</td>
</tr>
<tr>
<td>9. The added mass matrix can be written as a simple 3x3 matrix with non-zero terms only on the diagonal</td>
</tr>
</tbody>
</table>

These assumptions may appear restrictive, but most are no serious limitations because a small scale aircraft flying small amplitude manoeuvres will approximate them quite well. The rigid body assumption (1) holds well for a low aspect ratio airframe at small load factors. Remaining wing flex and fuselage torsion is expected to be insignificant for this project. As an all electric plane, its weight and inertias are constant during a flight (2). By design, the utilised airframe is symmetric about the XZ plane. The only asymmetry is the protruding airdata probe, which has been balanced inertia-neutrally with ballast weights inside the opposite wing (3). The following three assumptions (4-6) about the inertial frame can also be considered true, since a small aircraft does not cover large distances. The next two assumptions about the propulsion system (7,8) are also true by design and were verified by calculations. As mentioned before, the assumption about the form of the added mass matrix is in line with the aeronautical literature. Yet, throughout this thesis it will become obvious that although reasonable results were achieved using this form, some results cannot be explained and further research is required to determine the true form of the added mass matrix for a small fixed wing aircraft. Some additional assumptions are required regarding the omnipresent wind, as discussed next.

1.9.2 Effects of an Unsteady Atmosphere

The presence of an atmosphere in motion has several implications on the equations of motion of an aircraft. Firstly it is necessary to define the terms wind, wind shear and turbulence in relation to small scale aircraft flight:
Wind: Wind is defined as the large scale motion of the atmosphere with respect to the earth frame, which has time- and length scales much larger and slower than the UAVs flight dynamics. This results in a constant local wind velocity distribution of the air surrounding the UAV. Wind speed and direction can be measured with ground instrumentation or be deducted from flight data by using the difference in measured speed and heading relative to the ground (GPS) and the air (airdata).

Wind Shear: For full scale aircraft, wind shear is rapid change in either wind direction of velocity, which can have sharp spacial gradients, like a thunderstorm front. Wind shear causes significant wind gradients across the aircraft, which can severely influence the stability and control of an aircraft. For a small scale UAV, wind shear is not considered because firstly, the UAV is typically too small for these large wind gradients to develop and secondly, it would not be able to fly in conditions that produce wind shear anyway.

Turbulence: Turbulence is air mass motion on a smaller and faster scale than the aircraft size and dynamics. Similar to wind shear this flow field variation causes airspeed gradients across an airframe, but on a much smaller scale. Turbulence causes airframe vibration but is seldom so severe that stability and control problems occur. Turbulence is very difficult to measure directly, but several well tested model based on the stochastic properties of typical atmospheric turbulence exist.

For this thesis only wind and turbulence are considered. Wind will be included in the equation of motion and turbulence is modelled as random noise in the flight data processing. Hence, the following additional assumptions are made with respect to the motion of the atmosphere to address those simplifications:

**Additional Assumptions for the Unsteady Atmosphere**

- The atmospheric wind gradients are large compared to the size of the UAV
- Wind direction changes are slow compared to the UAV dynamics
- Turbulence levels are low enough to not change the stability and control properties in a significant manner, which would require the inclusion into the equations of motion

The following sections briefly discuss the effects of the wind on the various aircraft states. Some of them are affected by the motion of the atmosphere and others not at all. The results of this discussion were then used to formulate the EKF process model, as listed in section 5.2.2.

Effects of Wind on Aircraft Translation

If the atmosphere is translating relative to the earth, an aircraft will translate together with the atmosphere, while also translating relative to the air mass. Any change in the speed or direction of the wind results in additional terms in the state equations for the translation and in the measurement equations for the translational sensors, the accelerometers as well as the measurement equations for velocity and position. These
terms describe a change of motion of the atmosphere frame $F_A$ relative to the earth frame $F_E$. Based on the above assumptions, these wind changes will unlikely be affecting the system ID test points because of their short duration, but they need to be included in the flight path reconstruction tasks.

**Effects of Wind on Aircraft Rotation and Attitude**

The second assumption above, that wind direction changes are slow compared to the aircraft dynamics, states that there will be only very slow effects on rotation rates and aircraft attitude relative to the earth frame. These effects are in fact small enough to be entirely ignored. Hence, the state equations for the rotation rates and attitude as well as the measurement equations for the gyroscopes and magnetometers are not affected by the presence of wind and can be used in their standard form.

**Effects of Wind on Airdata Measurements**

The airdata sensors measure speed and attitude of the aircraft relative to the atmosphere. They will not be affected by the presence of wind, because a translating air mass does not affect the relative speed between it and an immersed object. The only step necessary will be to ensure that the measurement equations for the airdata sensors are written using the body axes velocities relative to the air $V^A_B$ and not relative to the ground $V^E_B$. The difference between the two body axes velocities needs to be established during the flight path reconstruction by applying the wind estimates to the flight data.
2 Motivation ........................................... 23
  2.1 Introduction
  2.2 Market Review
  2.3 The UAVmainframe Concept

3 Flight Hardware ................................. 29
  3.1 Main Computer
  3.2 Power Distribution Board
  3.3 Control Signal Generator Card
  3.4 Sensor Cards
  3.5 Airdata Probe
  3.6 Other Hardware

4 Flight Software ................................. 41
  4.1 Data Acquisition
  4.2 User Interface
  4.3 Flight Control and Input Sequence Generation
  4.4 Fault tolerance
  4.5 Groundstation Software

5 Data Processing Software ................. 53
  5.1 Data Management
  5.2 The Extended Kalman Filter
  5.3 System Identification Software

6 System Calibration ............................ 73
  6.1 Sensor Error Sources
  6.2 Timing Accuracy
  6.3 Inertial Sensor Calibration
  6.4 Airdata
  6.5 Control Surface Feedback
2. Motivation

2.1 Introduction

To perform the planned aircraft system identification tasks from flight data requires accurate measurements of the motion of the aircraft, while the aircraft executes suitable manoeuvres to excite its dynamics. The required data for stability and control analysis are the translational accelerations, the rotation rates, the aircraft attitude, the aerodynamic inflow angles and the dynamic pressure. Further data is required for navigation and flight management, as shown in Table 2.1. For full scale aircraft testing, where the flight crew manages the flight path, it is therefore sufficient to install a data acquisition system for the required parameters (or use the flight instrumentation already present) and have the pilot fly the required manoeuvres, while the data is recorded. The pilot has information on all the other required states for controlled flight, like position, ground speed, heading and remaining fuel available on his instruments. The pilot also has direct feedback about the aircraft’s motion from his own senses and uses all available information to ensure safe an controlled flight.

For a small, remotely piloted aircraft considered for this project, there are several additional requirements for successful data gathering. Since the pilot is not aboard the aircraft and flies based on visual contact only, information about air- and ground speed, attitude and remaining fuel (or battery power) are not readily available. Flying remotely also removes any direct feedback of the aircraft’s response to control inputs. Additionally, the small scale of the airframe does not allow for highly accurate sensor systems, which would simply be too large and too heavy. One has to utilise lower grade MEMS sensors that due to their small size, low weight and cost can be integrated into a small airframe, at the expense of less accurate raw data. In order to collect useful flight data, the onboard avionics must do much more than just record the required parameters listed in the table.
Chapter 2. Motivation

It needs to be able to send telemetry information to the pilot and operator on the ground, must accept commands being send to it, must be able to generate the required control surface sequences, which are very difficult to manually input from the ground far away, and ideally have some kind of flight control assistance to the pilot to help with keeping the correct attitude during the flight test manoeuvres.

<table>
<thead>
<tr>
<th>States</th>
<th>Nav. &amp; Control</th>
<th>+Wind</th>
<th>SysID (stab./ctrl)</th>
<th>SysID full</th>
<th>Fault tolerance</th>
<th>Flight Mgmt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Velocity</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Acceleration</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Attitude</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Attitude rates</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Altitude</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Airspeed</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>AoA</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Sideslip</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Ctrl Feedbk</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Thrust</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>(✓)</td>
<td>(✓)</td>
<td>-</td>
</tr>
<tr>
<td>Fuel flow</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Energy status</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Redundant</td>
<td>-</td>
<td>-</td>
<td>(√)</td>
<td>(√)</td>
<td>√</td>
<td>-</td>
</tr>
<tr>
<td>Sensors</td>
<td>(benchmark/ noise mitig.)</td>
<td>(benchmark/ noise mitig.)</td>
<td>√</td>
<td>-</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.1: Aircraft states required for several data processing applications. States in [] are not yet integrated into the UAVmainframe

Based on these considerations and work done during this project, a detailed list of requirements for a small aircraft avionics system for flight testing can be compiled as follows:

**Data Acquisition**

- Sampling frequency of 100 Hz or better
- Timing accuracy better than 5 msec across all sensors
- Redundant sensor suite for cross-checking and potential augmentation
- Performance benchmarked by a wide variety of experiments

The sampling frequency is determined by the expected maximum speed of the various modes of motion of the test aircraft. Due to the small size of the airframe considered for this project, these frequencies will be much higher than for a full scale aircraft. At the start of the project, when the inertial properties of the aircraft were yet to be determined,
some initial estimates indicated that 4 Hz would be a reasonable upper bound for the mode frequencies. Reference [22] suggests that the sampling frequency of the data acquisition system should be 25 times the fastest frequency to be measured to obtain a good resolution. Hence, a sampling frequency of 100 Hz is required for the proposed data acquisition system.

Experience during the project has shown that all sensors should be synchronously sampled in the first half of the 10ms interval. The control surface feedback sensors are especially critical. It can be shown from test data that a delay of 10ms (or a single time frame) on these sensors causes a significant change in the estimated parameters.

Redundant sensors allow for cross checking of the results, and to select the best sensor based on noise levels and other errors. Measuring the same quantity with different principles will allow to use sensor fusion methods to correct for sensor errors. Both considerations allow to quickly benchmark the newly developed data acquisition system and to isolate errors quickly. Confidence in the results can only be gained by careful benchmarking of the data against independent data sources.

**Flight Management and Control**

- Designed for flight safety without compromise
- Flight stabilisation for safety and improved test data quality
- Flexible onboard manoeuvre generation, adjustable in flight from the ground

The proposed avionics system will need to perform some form of closed loop control to aid the remote pilot. This turns the system into a safety critical component and it needs to be designed for this task. Even though the test aircraft is unmanned, an accident or crash will set back the project for a long time and therefore the avionics system needs to be reliable and safe in operation.

The control surface input sequences required to excite the aircraft dynamics are ideally also generated by the avionics system to achieve reliable and repeatable manoeuvres. Methods to pre-program these sequences are therefore required and the pilot and flight test engineer need to be able to engage these sequences from the ground station in a predictive and safe manner.

**User Interface / Telemetry**

- Real time telemetry and control interface
- Short range/high rate and long range/low rate datalinks
- Utilize the established MavLink telemetry protocol [40]
- Interface with the Qgroundcontrol ground station software [41]

The entire user interface needs to take place across a radio link. Real time data must be examined on the ground for quality and error checking. Commands must be initiated on the ground and then executed on board. This link must be stable and fast enough for the data rate and latency requirements.
General Requirements

- Very powerful, yet small, lightweight and relatively low cost
- Fully modular design to accommodate many different requirements of the experiments used
- Must be expandable and upgradeable for future expansion and additions of new technology

All the previous requirements need to fit inside a very small and light package to be able to be flown on the small aircraft under consideration. At most, 500 grams would be allowed, which includes all hardware as well as the wiring, which can add considerable weight. The system should also be fully modular to allow for rapid re-configuration to adapt between the planned wind tunnel and flight experiments, as well as the inertia swing tests. Technology rapidly advances in this field and therefore components specified at the start of the project might already be outdated, or worse out of production near the end. Modularity ensures these components can quickly and easily be replaced by new parts.

The following market and literature review was done to identify potentially available systems (that also have to fit the budget requirements) and to learn about other approaches to the issue of flight data acquisition and management.

2.2 Market Review

Currently available, non-military, non-classified flight control/management systems of this scale are rare and typically do not have the built-in redundancy to mitigate equipment failures. While they all offer some kind of data-logging capability, it is often not possible to calibrate individual sensors to achieve the required accuracy and compatibility of the recorded data to analyse the flying qualities of the test aircraft. Commercial, conceptually similar data acquisition solutions do exist, for example [42], but those focus on the data recording part and do not feature any of the control systems and hardware redundancies required to safely control an unmanned system. Most of those systems are still too big and heavy to fly on a small scale aircraft.

The combined (data acquisition and control) systems of this scale are predominantly research systems developed by universities around the world. It seems that there is not yet much commercial interest in this sector. The scientific literature can be divided roughly between publications published before 2010 [43, 44, 45, 46, 47] and others from 2010 till today (2015) [20, 48, 49, 50, 51]. The reason for this division has been the rapid advancement of technology driven by the mobile device revolution. Any system designed before 2010 is most likely obsolete today. None of the recent systems can be compared to the UAVmainframe’s size, capabilities, flexibility and safety. Systems with some comparable features are [17, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61], but none fulfils all criteria and none is commercially available to this author’s knowledge.

The most widely used system for an aircraft of this scale at the time of writing (2016) is the Pixhawk flight computer [62], designed as an autopilot for aircraft, multirotors and ground vehicles, depending on the software used. The unit is now available at low
cost, but it did not yet exist when this project was started. Being designed as an flight controller, the focus of the design and the software is control. The system is capable of logging data for debugging, but this logging functionality has only very low priority and is done only if the computational timing allows. This results in an unevenly spaced time vector, which would require serious work to prepare the data for the system ID requirements. The Pixhawk also only allows for rudimentary sensor calibrations and has a limited number of sensor channels.

2.3 The UAVmainframe Concept

Since no suitable avionics system was (and still is) not commercially available, it was decided to develop the UAVmainframe, an avionics system designed to the specifications listed above, with the data acquisition task a central priority. The system will be based on a small, single board computer running the embedded Linux operating system. This provides plenty of computing power in a small package. The embedded Linux operating system provides all the required programming interfaces and connectivity and is compatible to the standard desktop version of Linux, which simplifies the operation and learning curve. Linux is, however, not a real time operating system and not designed to adhere to strict timing requirements. The concept of the UAVmainframe addresses this through the addition of dedicated sub-processors for communication with the sensor units, as well as advanced programming methods. A second, unique, feature of the UAVmainframe concept is shown in Figure 2.1. Inertial sensors are distributed across the aircraft, with the main INS located at the CG and auxiliary accelerometers and magnetometers at the wing tip and the tail. This fulfils the requirement of redundant sensors and provides a large amount of data to be used for analysis. Control surface feedback sensors based on the Hall effect principle are measuring the surface deflection angles.

The remainder of this part of the thesis describes the system and the accompanying software in more detail, before being put to use to demonstrate the performance and accuracy of the design.
Figure 2.1: UAVmainframe distributed sensor locations
3. Flight Hardware

The hardware concept of the UAV mainframe is shown in Figure 3.1. It consists of four main blocks, the main computer, the sensor cards, the control signal generator and the power supply with integrated sensors. The main computer is a commercially available, credit card sized embedded Linux computer, which runs the main flight software. It communicates with expansion cards that connect the system to the sensors, making it easily re-configurable. A separate and redundant control signal generator connects the system to the flight controls of the aircraft. The system is powered by the power distribution board, which in turn also provides accurate sensor data about all power systems of the aircraft.

The UAV mainframe concept is currently implemented (and was used for this thesis) in a prototype with most of the intended capabilities as shown in Figure 3.2. The prototype measures 100x80x60 millimeters and weights 250 grams including all wiring. In addition to the system shown in Figure 3.2, there are a number of auxiliary components like the power distribution board, to complete the system. These external components will be discussed later on.

All hardware of the UAV mainframe prototypes (four sets were built) was assembled by hand, using commercially made printed circuit boards. All component soldering was done in house using a simple SMT reflow oven. While this may sound cumbersome, it eventually is much faster than sending the boards off for assembly (at least for such a small number of sets built) and all quality control can be done in house as well. This ensures fully functional and well tested components to be installed on the aircraft and enables highly productive and safe flight testing. No hardware has ever failed during this project, neither in the air or in the wind tunnel.
3.1 Main Computer

The main computer of the UAVmainframe is a credit card sized embedded Linux computer (BeagleBone Black [63]) running a single core mobile phone application processor. The CPU can be clocked up to 1GHz which provides ample computing power for the data processing and storage, the user interface, the sensor interface and the flight controller. Multiple hardware interfaces provide connections to the various parts of the system: The user interface connects through a wired or wireless network in the laboratory or a long range serial data radio in flight to the ground station laptop computer. The sensor cards are linked via various data buses that can operate in parallel to transfer the sensor data to the main CPU. A sync line ensures correct timing between the cards and the main CPU. The flight control interface generates the required actuator commands based on
the sensor data. These commands are sent to the control signal generator for further processing and output to the servo motors and engine throttle.

The components of the main computer are shown in Figure 3.3. The assembly consists of 3 circuit boards: The BeagleBone Black, the support ‘Cape’ board and the main-board with the card sockets. The support cape contains all components to support the BeagleBone Black. It features the battery buffered real time clock, as well as the required voltage level shifters to connect the 3.3V BeagleBone Black to the 5V sensor cards. The main-board is simply an interconnect between the main computer and the card slots. It does not contain any electrical components except the connectors. The intermediate support cape was necessary to be able to route the main-board circuitry on a double layer circuit board. Otherwise, the parts located on the support cape could have been placed on the main-board to save a circuit board and the associated weight. A further iteration of the hardware will most likely be designed without the support cape and using a multi-layer main circuit board.

### 3.2 Power Distribution Board

The power distribution and monitoring board, as shown in Figure 3.4, is an integral part of the fail-safe concept of the UAVmainframe. By generating clean supply voltages for the system and monitoring all voltages and currents, the power systems can be assessed at all times and problems can be detected early. The power supply of the aircraft consists of two batteries, one for main engine power, the RC receiver and the servo motors and the other one for the supply of the UAVmainframe. The idea behind this split of battery supplies is that in the event of the UAVmainframe battery failing, the UAVmainframe will stop operating but the aircraft is still controllable by the safety pilot, because its
systems (propulsion and control surfaces) are still powered from the main propulsion battery. Even in the event that this large battery runs low, the control system will still work because the main motor will stop running long before the battery runs completely flat. This allows for at least a glide to save the aircraft in the unlikely event of both supply batteries being drained unnoticed. The power board also features a ground power port, where a second, larger battery can be connected without removing the flight battery. This allows for the UAVmainframe to stay powered up during battery change on the ground between flights, which saves time and reduces the chances for errors. There have been test days, where the system was powered up all day without interruption. This also keeps the sensors at a constant temperature and reduces bias drift due to changes in sensor temperature.

The sensor card on the power distribution board is an 8 channel 16-bit A/D converter to monitor all supply voltages and currents as well as the motor and speed controller temperature. Currents are measured with precision shunt resistors and amplifiers, with the main power shunt capable of currents up to 100 Amperes using four shunt resistors in parallel, as shown on the upper right of Figure 3.4. The shunts for the supply currents for the servo motors and the UAVmainframe are optimised for maximum current resolution in the range of 0-2 Amperes. The A/D converter features third order Butterworth [64] low pass filters with a 10 Hz cut-off frequency for anti-aliasing purposes. All power channels are recorded at 100 Hz, which results in very high resolution of the power measurements.
3.3 Control Signal Generator Card

The control signal card is shown in the top centre of Figure 3.5. It features an ATMEGA 32U4 microcontroller [65], which can generate 6 channels of PWM in hardware to command the servo motors and the throttle of the aircraft. The signals from the RC receiver are routed through the relays on the card, which allows to switch between the control signals from the RC pilot or the signals generated by the UAVmainframe as shown in Figure 3.1. The safety switch of the remote pilot is implemented using a separate channel on the RC receiver. Depending on the state of that channel (and several other criteria like valid communication with the main computer) the card CPU switches between the control signal sources. The relays are wired in a way that the automatically connect the RC pilot to the controls in case of a power failure or any other malfunction. This ensures controllability of the aircraft at all times and is an unique feature of the UAVmainframe not seen on any comparable system. This safety concept has worked very well during the project and there was never a situation where the aircraft became un-responsive due to an onboard failure. The ability to take over control at any time also gives the test pilots much more confidence in the system, especially during flight controller tuning and the initial input sequences, where the full response of the aircraft could be surprising at first.

3.4 Sensor Cards

Figure 3.5 shows a selection of sensor cards of the UAVmainframe. All feature a common 26-pin connector that supplies power and connects the data buses to the card CPU. Some cards also act as breakouts for the buses to be able to connect the external sensors cards that are located away from the main-board. The system is designed in such a way that each card can be plugged into any slot on the main board and the software will
recognize it automatically. Most cards are based on modified versions of the popular ARDUINO platform [66], which allows for quick and relatively easy code development. A advantageous feature of this distributed computing approach is that each component in itself is fairly simple and thus easy to design and to maintain. All card functionalities are grouped by sensor type, the control feedback card reads all control surface feedback sensors, the airdata card all airdata related sensors and more. Some of the cards with notable features will now be described in more detail.

### 3.4.1 Airdata Card

The airdata card is the interface to the airdata probe. It contains an absolute and a differential pressure sensor to read the ports on the pitot tube. The absolute pressure sensor is connected to the static port of the pitot probe with a silicone hose to eliminate potential errors from differences in cabin pressure compared to the ambient static pressure. The pressure sensors use their own, separate power supply, feeding from the sensor power rail, for maximum accuracy. The card also has inputs for the position encoders that are mounted to the air vanes on the probe to measure the angle of attack and angle of sideslip. Ambient temperature is obtained from the pressure sensors to calculate the air density. The card connects to the system via I2C bus.

### 3.4.2 GPS Card

The GPS card uses a state of the art Venus8 GPS module from Skytraq [67]. This module is, to the knowledge of this author, the only one on the market capable of outputting position estimates at 50 Hz. Most other modules are limited to 5-10 Hz. The Skytraq module uses a DSP processor to calculate 50 position estimates per second. It does not interpolate between slower measurements as confirmed by Skytraq. This high update rate is a major advantage over other solutions during the post-processing of the data.
3.4 Sensor Cards

The card has a slot for a backup battery to minimise the GPS startup time by buffering the real time clock of the GPS module. In practice, these modern modules require so little time (less than 30 sec.) to acquire the GPS satellite signals that the backup battery was never used.

Considerable time was spent to optimise GPS reception by choosing the best antenna configuration and placement. During initial testing the system always had issues with GPS reception, especially in turns, where the antenna is no longer pointing straight up due to the roll angle of the aircraft. Studying the datasheets of the available antennas, revealed that all small, commercially available GPS antennas are actually designed for automotive navigation, where the metal body of the car acts as a ground plane for the GPS antenna. No such ground plane existed in the UAV, which degraded the antenna performance significantly. A solution was found by placing the GPS antenna onto a balsa wood board, mounted above the UAVmainframe with clear view of the sky. The board was covered in aluminium foil and connected to the antenna ground. The dimensions of the board were chosen to be at least 5cm around the antenna, which is a quarter wavelength of the 1.575 GHz GPS signal. The assembly is shown in Figure 3.6. The GPS performance with that ground plane improved considerably and the rate of signal dropouts reduced to less than 10 frames during a typical 10 minute flight (30,000 frames). The second antenna on the board is the GPS antenna of the reference INS, which will be discussed later on.

3.4.3 RC-Interface Card

The remote control (RC) interface card reads the command data stream from the commercial RC hardware for logging purposes. The data is also used to implement flight stabilisation methods by modifying the pilot commands using PID control software and

Figure 3.6: UAVmainframe GPS antenna assembly
Chapter 3. Flight Hardware

sending these modified commands to the servo motors via the control signal generator. The RC-interface card can be used to read signals from different brands of RC gear by modifying the software on the card. The interface to the UAVmainframe is not affected by changing the RC hardware in the aircraft. This increases the flexibility of the UAVmainframe, since it is not dependent on a particular RC equipment.

3.4.4 Reference INS Card

The INS card, shown in Figure 3.7, supports the stamp sized VectorNav VN100 and VN200 INS [68]. The VN-100 is a 9 DoF sensor with integrated attitude estimator, and was used for the dynamic wind tunnel tests, where only rotations and no translations occur. The VN-200 is a 9 DoF inertial sensor plus GPS for position information and air pressure for altitude measurement. It contains a state estimator to combine this information to a full rigid body navigation solution, including positions, velocities, attitude, accelerations and rotation rates. The VN-200 is used for the flight tests and its filter solution is used as a benchmark for the UAVmainframe sensor fusion algorithm. Both sensors are factory calibrated by the manufacturer for sensor bias, scale factors and axis alignments. The card features a USB to serial converter for INS configuration and a super capacitor to buffer the VN-200 GPS real time clock, similarly to the GPS card mentioned above. The INS communicates via a high speed (8MHz) SPI connection, which is required due to the large amount of data to be transferred between INS and the UAVmainframe.

![Figure 3.7: UAVmainframe IMU card with VN-200](image)

3.4.5 Auxiliary Magnetometer/Accelerometer Card

The auxiliary magnetometer/accelerometer card is located inside the left wing tip, far away from any magnetic disturbance. The card features a 3-axis accelerometer and a 3-axis magnetometer and is the most miniaturised card of the system, measuring only 35x20mm as shown in Figure 3.8. It features the best 3-axis magnetometer chip available
3.4 Sensor Cards

3.4.6 Telemetry Radio

The long range telemetry radio is a RF Design RFD900 [72], as shown in Figure 3.9, compatible with the 3DR radio standard [73], which supports MAVlink diagnostic data about the radio link to be received by the UAVmainframe and the ground station. The radio features two antenna ports for antenna diversity. Two 900MHz dipole antennas were custom made to save weight as shown in Figure 3.10 and placed in a perpendicular location in the UAV. This ensures attitude independent reception and is a great improvement over single antenna radios used during testing of the system. The MAVlink diagnostic data of the radio received by the UAVmainframe contains information about the link quality and the transmission buffer state. This allows the UAVmainframe to dynamically adjust the amount of telemetry data to be transmitted, based on the buffer status. The data rate is reduced following a pre-determined schedule, which ensures that the most important telemetry data is always preferred and will get transmitted to the ground station. Less important diagnostic data, for example, is simply reduced in frequency to achieve a lower data rate.
3.5 Airdata Probe

The airdata probe, shown in Figure 3.11, is a custom design because commercial products are too large and too heavy for the test aircraft. The probe is made from a 1.5mm plywood frame with balsa wood fillings. The frame houses the two Hall effect angle encoders for the angle of attack and sideslip vanes. The vanes itself are constructed from a carbon rod with a depron fin and a brass counter weight and typically weigh three grams. The pitot tube is also custom made from two thin walled aluminium tubes that are mounted inside each other. The inner tube is the dynamic pressure port and the outer tube has holes around its circumference for the static pressure. The tip of the probe is sealed with epoxy resin and shaped into an ellipsoid with sandpaper. The end of the pitot tube inside the probe has two 2mm ports to connect to the hoses running to the pressure sensors of the UAVmainframe. These two tubes are also sealed with epoxy resin inside the respective parts of the pitot tube. The probe also contains a 3-axis accelerometer in its body to judge vibration levels. The wiring from the encoders, the tubing and the wiring for the integrated accelerometer are fed through a carbon fibre tube into the main wing of the aircraft. This carbon tube also acts as the mount of the probe and is fixed with clamps to the wing structure of the aircraft. The probe weights 55g in total. The calibration of the probe is discussed in a later section.
3.6 Other Hardware

There are several other components which complete the UAVmainframe as shown in Figure 2.1, but will not be discussed here. One notable exception is the wiring of the system itself. All analogue signals from the accelerometers require shielded cables to reduce noise. However, commercially available five wire shielded cable is exceedingly heavy and cannot be used on this scale. It was therefore required to custom make all shielded cables for the system in house. These cables are constructed from standard, unshielded ribbon cable, which is wrapped in aluminium foil. A bare strand of wire is wrapped into the shield and later connected to the signal ground to ground the entire shield. The assembly is then fed into a tube of heatshrink and shrunk to form a custom cable less than a third of the weight of commercial alternatives, but with the same performance. Constructing the cables required for a full UAVmainframe assembly took about three days.
The flight software of the UAVmainframe represents a very significant effort of this research project. It contains some 30,000 lines of code and took at least 1.5 man years to develop. Long hours of testing ensured that everything was working error free, since faults or software crashes in flight can be catastrophic. The code can be roughly divided into three large blocks, listed here in terms of priority levels for this project.

- Data acquisition
- User interface
- Flight control and input sequence generation

The highest priority item is the accurate data recording of all sensors of the UAVmainframe. In order to perform productive flight testing, the user interface with the ground station needs to be reliable and easy to use. The flight control and input sequence generation has become invaluable during the project, but in theory, the flight tests could have been done without it, albeit at much lower quality and considerably increased risk. Hence, all three blocks of the flight code are important for a successful flight test programme for a small, remotely piloted aircraft. This chapter will introduce some of the concepts and strategies used in the code to achieve all these objectives. A further requirement of a flight software is a high level of fault tolerance and some of the issues related to that requirement are presented as well.

The software concept of the UAVmainframe prototype is based on the open source Linux operating system (OS). Linux is an Unix based desktop operating system (OS), which offers a large set of features for multitasking and connectivity. It is also free to use and requires no special knowledge other than standard Unix familiarity. Linux has been ported to many hardware architectures and is the system of choice for the small, embedded computing platforms used for UAVs. Linux is not a real-time operating system, but features several methods to achieve good timing accuracy. The UAVmainframe uses
many of these features as will be discussed later on. The software that implements the UVAmainframe functionality runs as a program on top of the Linux operating system and interacts with the hardware through system calls to the Linux OS kernel.

4.1 Data Acquisition

The flight data acquisition requires reliable and accurate sensor readings at a constant time interval. As mentioned before, achieving this with a desktop operating system like Linux requires careful system design in hard- and software. The hardware concept with the multiple, parallel microcontrollers on the sensor cards was presented in the previous chapter. Here, the corresponding software design is introduced. Both work hand in hand to achieve the stringent timing requirements for high quality data acquisition.

4.1.1 General Code Layout

Figure 4.1 shows a simplified overview of the UAVmainframe software architecture in the context of the operating system environment. In any multitasking OS, multiple tasks require CPU time but only one can execute at a given time. The access to the CPU is governed by the task scheduler. It has a list of processes and threads with their respective priorities and grants access to the CPU based on that list. It is important to note that the OS kernel is also just a process that runs in parallel with all other tasks. A high priority of the kernel task ensures CPU time as required to provide all OS functionality. But it is possible to have a user program task running with higher priority than the kernel task to ensure reliable timing of that task. In that case, the user task has to ensure that it releases the CPU for the kernel and all other tasks to execute. This feature allows the UAVmainframe to achieve near real-time performance with a non real-time OS (together with some other functionality described later). The Linux scheduler also has different policies which dictate how the list of tasks is prioritised. The standard policy (Round Robin) grants every task a certain amount of CPU time and switches to the next task if the time is up, regardless of whether the task has finished its work. This is obviously not a good approach for reliable timing accuracy. The policy used for the UAVmainframe is FIFO (first in, first out). This policy grants CPU access based on task priorities until the task finishes or releases the CPU. If the user task does not release the CPU the system will stop because nothing else can execute, including the kernel task. The software developer has to ensure that the user code releases the CPU at times when the user task does not need it.

A process can be divided into separate threads, which can execute certain tasks of the process by encapsulating the functionality into a separate task that competes for CPU time. This allows for a thread to access a slow hardware interface and wait for this access to complete without blocking the rest of the program flow. The blocking task is simply taken of the CPU by the scheduler and another thread of the application can execute while the first thread waits for its request to finish. Threads can have priorities assigned to them in the same way as processes to govern the CPU access of each thread. As shown in Figure 4.1, the UAVmainframe application is multi-threaded, with the timing loop placed in a thread that has the highest priority of the entire system assigned to it.
The timing loop has very short execution times but it requires to run exactly at the right time. Having the top priority in the system ensures that the timing loop runs, no matter what else is using the CPU. The timing loop generates the sync pulse for the cards and acts as a sub-scheduler for the other tasks of the UAVmainframe, calling the threads accessing the sensor cards, processing and recording the data and interacting with the user via the telemetry link. Those threads have an identical but lower priority than the main timing loop and compete for CPU time. Since all the sensor card interactions are actually quite slow (in terms of main CPU speed), those threads send a single hardware
system call to the kernel requesting the transaction and then wait for completion. The Linux scheduler recognizes this waiting (blocking in OS terms) and assigns the next thread to the CPU, which again just requests a transaction on a separate interface and then blocks (waits). This repeats several times and results in virtually parallel execution of these threads. This architecture allows the UAVmainframe to transfer large amounts of data between the cards and the main CPU in a short amount of time, using several separate hardware interfaces in parallel.

4.1.2 Event Sequence

Figure 4.2 shows a diagram of the events being executed during each time step across the UAVmainframe. The loop runs at 100 Hz, resulting in 10 ms timestep length. The main timing loop generates a 300 Hz clock, which divides these 10 ms into three parts to execute events in the correct order. At $t=0$, a rising edge on the sync line is created to trigger all cards to initiate their tasks. The cards sample their sensors or execute other functionality, and on completion wait for the main CPU to request the data transfer. For the fast cards this request is sent out in the second half of the first sub-step and data transfer commences across all interfaces. On reception by the main CPU all data is run through the respective calibration methods and added to the sensor data memory structure.

At $t=3.33$ ms, the next clock tick occurs. All slow cards initiate the data transfer. The INS sensor readings have been transferred already, so a data processing thread can perform certain operations using this data, if required. At $t=6.67$ ms the third clock tick occurs. All data has been transferred and calibrated and is ready to be recorded. Then flight control thread (see below) does all calculations to generate the required commands for the control surfaces. On completion this data is transferred to the control signal generator, which on reception generates the PWM signal for the servo motors. All data of the current time step, including the updated control surface commands, is then recorded to the disk for later processing on the ground and real time telemetry information is sent out. On completion of all tasks of the current sub-step the system releases the CPU and waits for the next timestep. User interface operations are processed asynchronously as they occur, based on software interrupts.

This architecture ensures that all data is sampled synchronously among the cards and is then transmitted to the main CPU a quickly as possible. Highly parallel data transfers ensure large amounts of data can be exchanged across the system efficiently. Together, this enables the UAVmainframe to be an accurate, high performance data acquisition platform, while still containing all the powerful functionality of a modern desktop operating system.
4.2 User Interface

The user interface of the UAVmainframe consists of several options. As long as it is connected via a network link, the user can start the code from the command line with a debug option. This prints the state of each sensor at one second intervals into the terminal window and allows to check for any errors in the system. This debug output, however, is very resource intensive and should not be used during normal operation.
The main interaction of the user with the UAVmainframe is facilitated via the widely used MAVlink protocol [40]. This popular message based open source protocol allows to send telemetry data over a serial port or network link, and features a parameter interface to configure and change settings of the flight code in real time. The MAVlink protocol is read by the ground station software Qgroundcontrol [41], which is presented in a subsequent chapter. During ground tests and in the wind tunnel, a wireless network link enables very high data rates and therefore a detailed picture of the state of the UAVmainframe. This is enabled by the use of the Linux operating system, which provides facilities for system wide network links. During flight, a long range serial radio link is used for the communication. The data rate over this radio link is much slower, but with the use of dynamic load balancing a highly reliable communication between ground and aircraft is possible.

The MAVlink protocol allows for three main modes of communication. One is the telemetry data link to display data on the ground station screen for inspection and situational awareness of the operator. The second is to send commands, which are linked to buttons in the ground station, to change flight modes, engage the input generator, among others. And finally, the parameter interface allows to change parameters in the flight code from the ground in real time. These include the tuning parameters of the PID flight controllers, the properties of the control surface input sequences, the limits for aborting a manoeuvre and many more. The functionality of all these features depends on the implementation in the flight code and therefore much time was spent to integrate all important commands and parameters in a intuitive manner to allow for total control of the system during flight. With the final version of the flight code, it was possible to have the pilot switch the system into manual flight mode and totally re-configure the flight control logic to account for higher turbulence levels, for example, without requiring to land the aircraft. After that, the pilot would simply switch back to stabilised mode, and all new setting came into effect immediately. This flexibility greatly improves the productivity and safety of the flight test programme.

The parameters of the UAVmainframe that do not have to change in flight, like sensor calibrations, are configured through a simple configuration file. This file is read by the code on start up and configures all parameters specified in the file to the required values. Each system has its own configuration file with the individual calibration values and the user has to select the appropriate file prior to operation. In the future, it is planned to transfer all those parameters into the MAVlink interface to avoid the possibility of loading the wrong file on start up.

The input sequences for the control surfaces are defined by a second type of configuration file as listed below. These files are read on commencement of an input sequence and prescribe the motion of the control surface during the input sequence. The logic of these files is such that each time a time step defined in the file is reached (relative to the start of the manoeuvre), the state of the control surface changes to the programmed level.
4.3 Flight Control and Input Sequence Generation

The UAVmainframe has three flight modes: manual, stabilise and auto. In manual mode the pilot has full control, with the commands of the RC transmitter directly connected to the servo motors and the engine throttle via hardware. The UAVmainframe is recording the flight data passively in this mode. In stabilise mode, the pilot is aided by the flight controllers implemented in the flight code, but still has control over all axes through the stick commands. In auto mode the UAVmainframe executes a pre-recorded input sequence on one or more control surfaces. In order to determine the open loop dynamics
of the aircraft, the flight controllers are disengaged in the axis of the manoeuvre (lateral or longitudinal) but stay active in the other axis to retain the trim condition in that axis.

As shown in Figure 4.2, the flight control code runs in the third timeslot of the 10ms frame after all sensor data has been acquired by the main computer. This allows to use any of that data during the control loop calculations. In manual mode, the control inputs are fed through the flight controllers without alteration to be recorded as commands for the controls. In stabilise mode all of the control loops shown in Figs. 4.3 and 4.4 are engaged and the RC command inputs are fed into the PID loops as the input commands.

Figure 4.3: UAV mainframe lateral attitude controllers

Figure 4.4: UAV mainframe longitudinal attitude controllers

These PID controllers deliver full pitch and roll angle control, as well as a simple yaw damper. The controllers use the gyroscope rotation rate data and the attitude estimate of the reference INS as the feedback information. The gyroscope biases are removed by the estimator of the INS prior to the use in the control system. In the roll axis the bank angle is limited to a value that can be set by a parameter from the ground station, typically 60 degrees. The bank angle range is then linearly mapped onto the pilot command. This
results in a highly predictable response for the pilot, since if a full stick command is sent, the aircraft will always bank to the pre-defined angle and stay there. This allows the long distance circuits discussed in Chapter 14, which under manual flight control would be far to dangerous to fly for a remote pilot. The yaw damper implementation is very simple by commanding a zero yaw rate. The code currently does not include any feed forward in a turn, so the yaw damper will also try to correct a yaw rate during a heading change, resulting in an un-coordinated turn. Since the rudder is very ineffective on this aircraft and very low gains on the yaw damper yield the required response during straight and level flight, no further improvement was necessary so far. This shortcoming, however, has to be addressed in the future to enable full heading angle control for navigation purposes. The pitch controller is the latest addition to the control system to improve the longitudinal trim condition at the start of the manoeuvres. Its structure is very similar to the roll angle controller.

During auto mode, the controls of the selected axis are initially held fixed at the trim condition and routed around the flight controllers. The other axis remains in stabilised mode. At each time step of the main loop, the input generator checks against the list of time steps from the input definition file shown above, and if there is a match the output command for that surface will be modified to the specified value from the file. Additionally, the system checks the attitude estimates from the INS at each time step. If a violation of the specified maximum attitude parameters occurs, the input sequence is immediately aborted and the flight controller are re-engaged to return to the trim attitude. At the end of the input sequence, the system automatically returns to fully stabilised mode.

The control commands are recorded after the modifications from either the flight controllers or the input generator to be able to debug exactly what was sent out to the servo motors.

4.4 Fault tolerance

A software in control of an aircraft needs to be fault tolerant and able to handle errors without user interaction. Otherwise the results may be catastrophic. The UAVmainframe was designed from ground up with this philosophy. As presented before, the hardware features manual override and other safety features by design, such that the pilot can always take over if a problem occurs. Yet, the flight software in itself needs to be designed such that this is very unlikely to happen. Otherwise nobody will be able to gain some trust in the system, since a fault is to be expected at any time. This would severely constrain the productivity of the flight tests.

The UAVmainframe code was designed carefully to deal with any conceivable error sources. It does not contain any code where it would enter an infinite loop where there is no exit. This would be as bad as a complete crash or power loss. Similarly, the code is designed to check all data for consistency and to disregard all faulty data. This is done via extensive use of checksums and range checks. If any data is regarded as faulty, the system re-uses the previous data, instead of writing a zero. This is very important to
keep a consistent set of measurements. The PID controllers are quite robust and are able
to handle a data dropout for a short time, if the previous value is kept and the next valid
measurement is reasonably close to the old state (this ensures the ‘jump’ in the data is
small). If one would write the invalid data to zero instead, the large step in the data even
for a single measurement can create a large and potentially dangerous response in the
integral and differential parts of the PID controllers.

So far the UAVmainframe code has been extremely robust, with no crashes or signifi-
cant malfunctions during any flight or ground test. The only time the manual override
functions were used were during flight controller tuning and when a bad trim condition
at the start of a manoeuvre resulted in a dangerous attitude. With the new automatic
attitude check, this is rarely necessary any more.

4.5 Groundstation Software

The ground station software used is Qgroundcontrol [41], an open source project origi-
inating at the ETH Zürich. It was chosen because unlike most other codes available,
it can be customised to operate with any flight code and not just the particular one it
was designed for. The ground station connects to the aircraft via a network link on the
ground or via a long range serial radio in flight and allows to monitor extensive telemetry
data as well as to uplink commands to the UAVmainframe via the MAVlink protocol [40].

Figure 4.5 shows the flight view screen of Qgroundcontrol, customised for the
UAVmainframe. At the centre is the flight display, modelled after a typical artificial
horizon instrument. At the upper right is the map display with real time position data
superimposed on it. This is useful to find the small aircraft in the sky when looking up
from the screen. Three windows with telemetry data readout are used to monitor engine
performance, control surface positions, GPS data quality and several other parameter
that are important for safe and productive flight testing. At the bottom is the UAVmain-
frame control interface, which allows to start and stop the code, change its flight modes
and to send commands for the control surface sequences. There are two similar panels
(not shown) for longitudinal and lateral tuning of the flight controllers. These have input
fields for the various PID gains and options to activate various test procedures helpful
during controller configuration.

The real time telemetry plotting window shown in Figure 4.6 allows to plot any
telemetry data channel in real time for inspection. The Figure shows the data from the
longitudinal sensors during an elevator input. The telemetry rate is 10 Hz, so some of
the fast changes in states appear distorted but this view is nevertheless very helpful to
quickly find problems, whether it is a invalid calibration, a sensor failure or any other
discrepancy of the readings from the expected values. Based on this information, the
decision whether to continue or abort the flight can be made quickly. The software also
allows to replay the recorded telemetry data stream to analyse the data after landing to
isolate problems more easily.
Figure 4.5: Qgroundcontrol flight screen

Figure 4.6: Qgroundcontrol plot screen
5. Data Processing Software

After each flight, the recorded data is downloaded from the aircraft via network and then requires several processing steps before the desired aerodynamic parameters are available. This is done with three pieces of code, which are the topic of this chapter. There is the data management tool to prepare and edit the data, the Extended Kalman Filter (EKF) for error correction and data compatibility analysis and finally the parameter ID software to do the actual identification of the model parameters. The EKF was developed from scratch by this author, whereas the data management and the parameter ID code are modified and extended versions of the SIDPAC toolbox from reference [22] and the example software included in reference [74].

5.1 Data Management

The UAVmainframe saves data in a space saving, binary format, which is spread over several individual files for the separate sensor groups. This allows for the quick customisation of the UAVmainframe and reduces the data rate of the in-flight recording. Matlab, in contrast, works best with all data in a single struct in double precision floating point format. On modern desktop computers memory is not an issue, so it is not necessary to optimize for storage space. The Matlab file format for this project was chosen to be the \texttt{fdata} structure which is pre-defined in SIDPAC, and all functions are written to work with this format. The \texttt{fdata} structure was extended considerably to account for the extra data channels of the UAVmainframe.

SIDPAC comes with a GUI for data editing and inspection. This GUI was extended with many new functionalities for this project. The extended GUI (called KSID) has three main windows for data import, edit and analysis. Figure 5.1 shows the main window for data import and initial analysis, as well as manoeuvre extraction. The UAVmainframe...
binary data structure can be imported and saved in the \textit{fdata} format. Due to the slow performance of Matlab executing for-loops this step takes about 6 min per 10 minute flight. Once the data is in the Matlab format, the subsequent handling is much faster. The list on the right shows all available data channels. A click on each channel plots it in the graph window.

![KSID software main screen](image)

Figure 5.1: KSID software main screen

The list on the bottom shows the timestamps of the detected manoeuvres as extracted from the data, using a special index channel in the flight data, which has a defined state corresponding to the particular manoeuvre being executed. Clicking on the timestamp plots all relevant data channels as shown in the Figure for an elevator doublet. This allows for a quick inspection of the data quality and noise levels. A button allows to automatically cut down and save the manoeuvres into separate \textit{fdata} files for analysis.

The edit screen, shown in Figure 5.2, allows to edit and cut the raw data. This is used, for example, to cut of the unwanted data recorded during taxiing on the ground. The screen allows to select a channel as indicator where to cut and then to simply click into the graph window to define the in and out points. Saving the cut updates the internal data structure with a new time vector adjusted for the new data length. Errors can be corrected by using an undo function.

The third screen is shown in Figure 5.3. It is the data analysis window. It allows to select a data channel and a manoeuvre and plots the raw data and its power spectral density (PSD) distribution over a selected frequency range. Several low-pass filters can be applied to the data to judge their ability to reduce noise. The PSD is a good tool to compare the commanded input spectrum with the resulting motion of the UAV to judge the effectiveness of the input shape in exiting the required airframe frequencies for the
5.2 The Extended Kalman Filter

The UAV mainframe uses many different sensors to measure the state of the aircraft. Most of those are separate units at various locations across the UAV. All of them are miniaturized parts to be able to fit into the UAV. For good results from the parameter ID, all these separate data streams have to be compatible, that is conform to the equations of motion. Being compatible means that the data needs to be aligned with the chosen axes system, must be measured about the CG of the airframe and must be free of distortions like bias- and scale factors. For example, a measured acceleration will result in a velocity change along that axis. If the data is compatible, the measured acceleration, integrated over time, will match the measured velocity change. If not, sensor errors are present and

Figure 5.2: KSID software data edit screen

system ID process. Shown in the plot is the pitch rate response to an elevator input. The frequency response nicely covers the entire spectrum around the natural frequency of 1.5 Hz of the mode and the input is therefore suitable to excite the short period mode. The data analysis window also has a function to judge the timing accuracy of the data and produces the plots shown in Figure 6.3.

The KSID tool has proven very effective and valuable during the experiments, especially during the flight tests, because it allows to quickly judge the data quality recorded during the flights. Problems can be identified readily and rectified for the next departure. During post processing, the tool allows to quickly manage large amounts of data to be prepared for the system ID process.
must be accounted for. For the UAV mainframe, this results in corrections for installation and alignment errors, wind, but also in specific calibrations for the characteristics of the miniaturized sensing units, like temperature dependency, bias factors and noise levels. A final issue to consider is that the test aircraft is not a static flight platform, that can be set up and calibrated once for the rest of the test campaign. For transport, the wings have to be removed to save space and therefore their alignment is not exactly reproducible, the aircraft is made from balsa wood, which is affected by humidity, and so on. Repairs from damages during flight operations occur frequently at this scale of aircraft. All this requires constant re-calibration and evaluation of the system status to obtain the best possible results. An Extended Kalman Filter was developed, inspired by [75], to estimate the sensor errors that could not be eliminated by prior calibration and to correct the data to be fully compatible with the equations of motion. This chapter will discuss the filter and all related sensor calibration efforts to result in a corrected, compatible flight dataset for the system ID analysis.

The Kalman filter is an algorithm for data filtering and sensor fusion for optimal estimation of a system’s state, with [76, 77, 78, 79, 80] examples of a large body of literature. It is a recursive algorithm, which means that past measurements influence the current state. The principle of the Kalman filter has been summarised comprehensively by Newman [75]:

The aim of the Kalman filtering process is to find a stabilising feedback control which works to drive towards a minimum the error between a ‘true’ but unknown model state and an estimated model state. It operates under constraints based on the characteristics of the process noise covariance, the
The Extended Kalman filter is an extension of the linear Kalman filter to deal with physical models that are non-linear. This covers a large range of physical problems. The extended Kalman filter cannot be proven to be optimal, like the linear version, but it has been used successfully during numerous, high profile engineering applications, such as the Apollo guidance computer used for the moon landings [81] and the Mars rover missions [82]. It is also the basis of the GPS global positioning system and the base of almost every aerospace guidance and navigation system. For this project, the filter is used to estimate the unknown sensor errors of the UAV mainframe, as proposed in [22] and to check and correct any issues with the data compatibility of the flight data.

As with most other feedback control methods, the Kalman filter requires tuning by an experienced engineer to yield the optimum performance. This is a very difficult process, because typically the ‘true’ answer to the problem is unknown (otherwise there would be no need for a filter). The following sections describe the filter formulation used for this project and how the filter was tuned. Example results are discussed later in Part VII.

The EKF system model presented in this thesis is a reduced model which uses only the standard flight instrumentation suite (GPS, INS at the CG and airdata). An advanced filter formulation, which makes use of all the auxiliary sensors of the UAV mainframe is still work in progress. As presented in the following chapters, the standard filter performance is already very good, and thus the completion of the advanced filter was added to the list of future work.

5.2.1 EKF Algorithm

The EKF algorithm consists of two discrete steps: A time update (or prediction step) and a measurement update (or correction step). Each iteration of the filter runs both steps in sequence, updating the state estimates, the Kalman gain and the error covariance matrices. The filter is started with an initial estimate of the state vector and the error covariance matrix $P_0$. These are calculated from the available measurements and represent the best possible, but not necessarily correct, estimate of the system state at $t = 0$. The filter will converge to the near-optimal estimate of the system state during the run, if set up correctly.

The process at time step $k$ is modelled by a system of differential equations in state space form

$$\dot{x}_k = f(x_k, u_k, \omega_k)$$  (5.1)

$$y_k = h(x_k, u_k, \nu_k)$$  (5.2)

where $f$ are the state equations and $h$ the measurement equations, which depend on the current state $x_k$, a forcing function $u_k$ and process noise $\omega_k$, as well as measurement noise $\nu_k$. Both noise terms are assumed to be zero mean, random Gaussian noise. This assumption is most likely violated in a real world system, but it appears to be a good approximation of reality and has been used with good results in many applications.
The prediction step, where the filter predicts the state $x$ at the current time step ($k$), based on the information available at the last time step $k-1$, consists of a time integration of the state equations without the process noise $f(x_{k-1}, u_{k-1}, 0)$

$$x_k = x_{k-1} + \int_{t_{k-1}}^{t_k} f(x_{k-1}, u_{k-1}, 0) dt$$

(5.3)

Then, the error covariance matrix for the current time step $k$ can be predicted by

$$P_k = (I + F dt) P_{k-1} (I + F dt)^T + dt^2 G Q G^T$$

(5.4)

where $F$ and $G$ are the Jacobians

$$F = \frac{\delta f}{\delta x} \quad \text{and} \quad G = \frac{\delta f}{\delta \omega}$$

(5.5)

with $\omega$ being the process noise vector defined in section 5.2.4 and $Q$ is the process noise covariance matrix. The term $(I + F dt)$ is a simple Euler time integration of $F$. Since it is only used for the covariance matrix propagation and not for the state integration, this simple integration method is sufficient [80]. The time integration of the state equations in Eq. (5.3) is done with a fourth order Runge-Kutta method.

The correction step begins with the linearisation of the measurement equations $h$ about the current system state

$$H = \frac{\delta h}{\delta x}$$

(5.6)

Then the measurement equations $y_k = h(x_k, u_k, 0)$ are used to calculate the predicted outputs $y_{k}$, based on the state estimate from the prediction step. Next, the Kalman gain matrix is calculated from

$$K_k = P_k H^T (HP_k H^T + R)^{-1}$$

(5.7)

where $R$ is the measurement noise covariance matrix. Using that result, the system state $x_{k+1}$ is updated by multiplying the measurement error (or residual) $z_k - y_k$ by the Kalman gain

$$x_{k+1} = x + K_k (z_k - y_k)$$

(5.8)

where $z_k$ is the vector of sensor measurements. This represents an independent update of the state based on sensor measurements. The final equation is then used to correct, hence the name, the covariance estimate of the filter by

$$P_{k+1} = P_k - K_k H P_k$$

(5.9)

Then the process starts over and the feedback loop is closed by using the updated estimates of $x_{k+1}$ and $P_{k+1}$ in the next prediction step, where they become $x_{k-1}$ and $P_{k-1}$, respectively.

GPS data is only available at 50 Hz and the airspeed and flow angles are not valid below 5m/s airspeed. Any other sensor might experience dropouts during the flight. Therefore the EKF was designed to detect sensor data updates and only use the measurements in Eq. (5.8) that were updated at the current time step for the calculations. The following code fragment shows the Matlab implementation of Eqs. (5.7) to (5.9) used for the EKF. The $idx$ variable encodes the rows of updated measurements for the current time step.
Only the filter residuals $M_{res}$ with updated measurements are used to calculate $X_{new}$ and $P_{new}$, corresponding to $x_{k+1}$ and $P_{k+1}$. In order to have the correct matrix dimensions, only the corresponding rows of $H$ and corresponding rows and columns of $R$ are used. This mechanism can also be used to turn on and off measurements during the tuning process.

```matlab
% select measurements
H = H(idx==1,:);
R = R(idx==1,idx==1);

%M error in measurements
Mres = Z - Y;

% kalman gain $K$
K = P*H' / (H*P*H' + R);

Xnew = X + K*(Mres(idx==1));
Pnew = P - K*H*P;
```

### 5.2.2 EKF Non Linear Process Model

The EKF for this project is based on a model of general rigid body, six degree of freedom motion through Earth’s atmosphere. The following assumptions were applied during the development of the process model (adopted from [75, 83]):

1. Earth’s geometry is described in the earth fixed, earth centred (ECEF) coordinate system
2. Constant gravitational acceleration of $g = 9.80665 \text{ m/s}^2$
3. Position and attitude are reported relative to a local flat Earth at each data point
4. Earth’s translation and rotation are neglected. This is valid, because the UAV moves slowly and within visual range of the safety pilot.
5. The aircraft is a rigid body and does not deform during the test flights. Weight and inertial properties are constant.
6. Air velocity is constant across the flow field surrounding the aircraft.
7. Local air velocity changes (wind) are small with respect to the aircraft dynamic properties
8. Air pressure and density is assumed constant during a test flight, which is reasonable for a flight duration of approximately 10 minutes
9. The simple sensor model $x_{true} = (x_m - b_x)(1 + \lambda_x)$ is appropriate to model the sensor errors [22].

The process model of the EKF is described by the following state equations, formulated in state space form for the time-integration step. The state vector consists of the position $P$ in NED axes, the body axes velocities $V$, the attitude expressed in quaternions $q$, the wind vector $W$ in NED axes and a series of bias and scale factor states. Those are included to estimate the sensor errors. These states include bias values for the airdata,
accelerometers and gyroscopes, as well as scale factors for all previous sensors except the altimeter. Bias states were also added for the GPS measurements of position and velocity, as well as for the magnetometer and IMU attitude measurements. The use of these measurement biases will be discussed further below. The final state vector contains 44 states and can be written as

\[
X = [P \ V \ q \ W \ b_{\text{air}} \ \lambda_{\text{air}} \ b_{\text{acc}} \ \lambda_{\text{acc}} \ b_{\text{gyro}} \ \lambda_{\text{gyro}} \ b_{\text{pos}} \ b_{\text{vel}} \ b_{\text{mag}} \ b_{\text{att}}]^T
\] (5.10)

The state rate equations \( f(x_k, u_k, 0) \) link the derivatives of the states at the current time step to the current values of the states. This is used to predict the state of the aircraft at the next time step by numerical time integration. The state rate equations use the measurements of the accelerations and rotation rates as forcing functions \( u_k \), in addition to the gravitational acceleration. The rate of changes of the wind-, bias and scale states are unknown. They are therefore modelled as random walks, which is driven by the process noise [83].

The state rate equations \( f(x_k, u_k, 0) \) for the motion- and wind states of this process model, without the process noise (added later), can be written as:

**Theorem 5.2.1 — EKF Process Model.** Non-linear process model excluding the bias and scale factor states

Positions in NED axes:

\[
\dot{P} = L_{eb} V_B
\] (5.11)

Velocities in body axes:

\[
\dot{V}_B = -\Omega_C V_B + L_{bec} g + a
\] (5.12)

Attitude in body axes:

\[
\dot{q} = -\frac{1}{2} \Omega_q q
\] (5.13)

Wind in NED axes:

\[
\dot{W} = 0
\] (5.14)

All error state equations have the form \( \dot{X} = 0 \). The NED to body axes rotation matrix is

\[
L_{be} = L_{eb}^T = \\
\begin{bmatrix}
q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 + q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\
2(q_1 q_2 - q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 + q_0 q_1) \\
2(q_1 q_3 + q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2
\end{bmatrix}
\] (5.15)

The Coriolis matrix and the attitude propagation matrix, both using the measured rotation rates in body axes with bias and scale factors removed:

\[
\Omega_C = \begin{bmatrix}
0 & -r & q \\
r & 0 & -p \\
-q & p & 0
\end{bmatrix} \quad \Omega_q = \begin{bmatrix}
0 & p & q & r \\
-p & 0 & -r & q \\
-q & r & 0 & -p \\
-r & -q & p & 0
\end{bmatrix}
\] (5.16)
The measured body axes accelerations and rotation rates, with bias and scale factors removed, are:

\[
\begin{bmatrix}
    a_x \\
    a_y \\
    a_z
\end{bmatrix}
= \begin{bmatrix}
    (a_{x_m} - b_{a_x})/(1 - \lambda_{a_x}) \\
    (a_{y_m} - b_{a_y})/(1 - \lambda_{a_y}) \\
    (a_{z_m} - b_{a_z})/(1 - \lambda_{a_z})
\end{bmatrix}
\]

\[
\begin{bmatrix}
    \omega_x \\
    \omega_y \\
    \omega_z
\end{bmatrix}
= \begin{bmatrix}
    (\omega_{x_m} - b_{\omega_x})/(1 - \lambda_{\omega_x}) \\
    (\omega_{y_m} - b_{\omega_y})/(1 - \lambda_{\omega_y}) \\
    (\omega_{z_m} - b_{\omega_z})/(1 - \lambda_{\omega_z})
\end{bmatrix}
\]  

(5.17)

and finally, the gravity vector in NED axes

\[
g = [0 \ 0 \ 9.80665]^T
\]

(5.18)

### 5.2.3 EKF Measurement Equations

The measurement equations \( h(x_k, u_k, 0) \) calculate the predicted sensor measurements based on the current state vector, or in other words, the measurement equations are used to calculate what the EKF expects to measure with the installed sensors. This is then used to compare the predicted measurements with the actual state of the system and to update the state vector, as well as the state covariance matrix.

The measurements used for this EKF formulation consist of the position and velocity from the GPS receiver converted to NED axes, the airdata measurements airspeed, angle of attack, sideslip and pressure altitude and finally an attitude measurement, either the magnetic field vector rotated into the body axes, or the IMU estimate from the reference IMU. On modern GPS receivers the position and velocity are calculated by two different methods which makes them independent. The position is calculated from the satellite ranging method, whereas the velocity is determined from the doppler shift in the satellite signals. Hence, both measurements present independent information and can be used separately in the EKF.

The airdata measurements include the wind components, which need to be added to the body axes velocities for the measurement equations. The inflow angle measurements are affected by the off-CG location of the airdata probe. This will be discussed in the next chapter on sensor calibrations, but it has proven beneficial to do this correction inside the EKF instead of pre-calibrating the airdata measurements. For the sideslip measurements one has to distinguish between the definition of sideslip and the measurement of the corresponding vane. What is measured is the flank angle [22], which is slightly different from the definition of the sideslip, especially at higher angles of attack. Therefore, in the measurement equations the definition for the flank angle is used, and in the reconstruction of the sideslip angle during post processing the definition of the sideslip angle is used.

In all flight software used before 2016 there was an error in the recording of the magnetometer data that made it impossible to calibrate the data during post processing. The EKF measurement equations were therefore formulated to use either the magnetometer or the IMU attitude estimate for the attitude measurement. The selective update mechanism described in Section 5.2.1 then allows for either set of measurements to be
selected and the other to be turned off. The vector of measurements then becomes

\[ Y = \begin{bmatrix} P_N \ P_E \ P_D \ V_N \ V_E \ V_D \ V_{air} \ \alpha \ \beta \ \text{Alt} \ \ B_x \ \ B_y \ \ B_z \ \ \phi \ \theta \ \psi \end{bmatrix}^T \]  

(5.19)

and the measurement equations \( h \) can then be written as a function of the states as follows:

**Theorem 5.2.2 — EKF Measurement Equations.**

GPS position and velocities in NED axes:

\[ \mathbf{P}_{m,NED} = \mathbf{P} + \mathbf{b}_{pos}; \quad \mathbf{V}_{m,NED} = \mathbf{L}_{eb} \mathbf{V} + \mathbf{b}_{vel} \]  

(5.20)

Airdata measurements including bias and scale factors:

\[ V_{air_m} = \left( \sqrt{u_A^2 + v_A^2 + w_A^2} \right) (1 + \lambda_{V_A}) + b_{V_A} \]  

(5.21)

\[ \alpha_m = \left( \tan^{-1} \left( \frac{u_A}{v_A} \right) \right) (1 + \lambda_{\alpha}) + b_{\alpha} \]  

(5.22)

\[ \beta_m = \left( \tan^{-1} \left( \frac{v_A}{w_A} \right) \right) (1 + \lambda_{\beta}) + b_{\beta} \]  

(5.23)

with airspeed components, including the wind estimate and the correction for the off-CG location of the airdata probe:

\[ \begin{bmatrix} u_A \\ v_A \\ w_A \end{bmatrix}^T = \mathbf{V}_B + \mathbf{L}_{be} \mathbf{W} + \mathbf{x}_{air} \]  

(5.24)

Pressure altitude relative to starting altitude including bias value:

\[ \text{Alt}_m = -P_z + b_{Alt} \]  

(5.25)

Magnetic field reference vector \( \mathbf{B} \) in body axes:

\[ \mathbf{B}_m = \mathbf{L}_{be} \mathbf{B}_{ref} + \mathbf{b}_{mag} \]  

(5.26)

Attitude in Euler angles

\[ \phi_m = \tan^{-1} \left( \frac{2(q_2 q_3 + q_0 q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2} \right) + b_{\phi} \]  

(5.27)

\[ \theta_m = -\sin^{-1}(2(q_1 q_3 - q_0 q_2)) + b_{\theta} \]  

(5.28)

\[ \psi_m = \tan^{-1} \left( \frac{2(q_1 q_2 + q_0 q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2} \right) + b_{\psi} \]  

(5.29)

where \( \mathbf{L}_{be} \) is the rotation from NED to body axes as defined in Eq. 5.15, and \( \mathbf{x}_{air} \) is the location of the airdata probe with respect to the CG.
5.2 The Extended Kalman Filter

5.2.4 EKF Process Noise

The process noise terms $\omega_k$ in the state equations,

$$\dot{x}_k = f(x_k, u_k, \omega_k) \quad (5.30)$$

which have not been treated yet, are the topic of this section. In this system model, the forcing functions $a$ and $\omega$ are sensor readings of the accelerometers and the gyroscopes, respectively. These readings contain sensor noise, that enters the system in the state equations. This sensor noise is not distinguishable from the process noise terms $\omega_k$, and feeds through several states during the filter run due to off-diagonal terms in the process noise matrix $G$. Therefore the process noise vector includes the sensor noise from the forcing functions, together with linearly additive noise for the remaining states. The base process noise vector used for this filter formulation is

$$\omega_k = [\omega_P \, \omega_{Acc} \, \omega_{Gyro} \, \omega_W \, \omega_{Air} \, \omega_{Acc_b} \, \omega_{Acc_{b'}} \, \omega_{Gyro_b} \, \omega_{Gyro_{b'}}]^T \quad (5.31)$$

with process noise corresponding to positions, wind and bias states, together with the measurement noise from the accelerometers and gyroscopes. Depending on how many bias states are included in the model, the length of the process noise vector will vary.

5.2.5 EKF Multi-Pass

In order to correctly identify all sensor error and wind states, the EKF was set up for multiple passes. This significantly improves the performance, because the error and wind states are initialised with the results of the previous pass. The initial error covariance matrix $P_0$ is also re-initialised with the diagonal of the previous run. Typically two to three passes were run, depending on the noise levels in the data. This strategy yielded very stable sensor error estimates, as long as the state was observable, and relaxes the requirement for precise initial conditions.

5.2.6 EKF Input Pre-Processing

Before the filter run, the GPS and magnetometer measurement data has to be processed to be in the correct format for the filter. The GPS data is recorded in the ECEF frame and needs to be transformed into the NED frame used by the filter. The transformation of the ECEF GPS data into the NED frame is defined in [84] as

$$P_{m,NED} = L_{ned\rightarrow ecef} [P_{ECEF} - P_{ECEF_{ref}}] \quad (5.32)$$

and

$$V_{m,NED} = L_{ned\rightarrow ecef} [V_{ECEF}] \quad (5.33)$$

where $P_{ECEF_{ref}}$ is the origin of the local NED frame in ECEF coordinates, typically the starting location of the take off run of the UAV, and

$$L_{ned\rightarrow ecef} = \begin{bmatrix} \sin(\phi) \cos(\lambda) & -\sin(\phi) \sin(\lambda) & \cos(\phi) \\ -\sin(\lambda) & \cos(\lambda) & 0 \\ -\cos(\phi) \cos(\lambda) & -\cos(\phi) \sin(\lambda) & -\sin(\phi) \end{bmatrix} \quad (5.34)$$
where $\lambda$ is the longitude to the NED origin and $\phi$ the latitude of the NED origin, typically the start of the take off run of the UAV, both in longitude, latitude and altitude (LLA) coordinates, as before.

The raw magnetometer data is calibrated for hard- and soft iron and sensor orientation before the filter run with the methods developed in the next chapter. All other data can be used in the filter as recorded by the UAV mainframe, with only some unit conversions from degree into radians and $g$ into $m s^{-2}$ required.

5.2.7 EKF Result Processing

After the EKF run, further processing is required to obtain the data used for the system ID process later on. Rotation rates, accelerations and the airdata require post-processing, because those variables are not directly estimated as states. The state rate- and measurement equations defined in section 5.2.2 can be used to calculate these parameters from the state estimates as follows.

Rotation Rates

To obtain the estimates for the body axes rotation rates, firstly the quaternion states have to be differentiated. Then Eq. (5.35) can be used to yield the final estimates for the corrected gyroscope data $\omega_m$. Differentiating the quaternion states can be noisy. The final rotation rate estimates are therefore low-pass filtered with a frequency domain filter (no phase shift) with a cut-off frequency of 10 Hz.

$$\omega_{ekf} = 2 \begin{bmatrix} -q_1 & q_0 & q_3 & -q_2 \\ -q_2 & -q_3 & q_0 & q_1 \\ -q_3 & q_2 & -q_1 & q_0 \end{bmatrix} \frac{\delta}{\delta t} (q)$$ \hspace{1cm} (5.35)

Accelerations

Similarly to the rotation rates, the body axes accelerations can be obtained by differentiating the body axes velocities and re-arranging Eq. (5.12) to yield the estimated accelerometer readings $A_m$. The resulting accelerations are also low pass filtered to reduce the noise generated by the differentiation of the velocities.

$$A_{ekf} = \frac{\delta}{\delta t} (V_B) + \Omega_B V_B - L_{be} g$$ \hspace{1cm} (5.36)

Airdata

The airdata estimates are calculated from a combination of the body axes velocities and the wind velocities using Eqs. (5.21) to (5.24) with all bias states set to zero. For the sideslip the definition of the true sideslip angle is used. The airspeed estimate is used to calculate the dynamic pressure during each test point. Because the aircraft flies at low subsonic speeds, the density recorded during flight can be used without further processing for that calculation.
\[ V_{\text{air,ekf}} = \sqrt{u_A^2 + v_A^2 + w_A^2} \]  
(5.37)

\[ \alpha_{\text{ekf}} = \tan^{-1}\left(\frac{w_A}{u_A}\right) \]  
(5.38)

\[ \beta_{\text{ekf}} = \sin^{-1}\left(\frac{v_A}{u_A}\right) \]  
(5.39)

with airspeed components, including the wind estimate:

\[ [u_A \ v_A \ w_A]^T = V_B + L_{be} W \]  
(5.40)

The dynamic pressure \( \bar{q} \) is updated from the airspeed estimate

\[ \bar{q}_{\text{ekf}} = 0.5 \rho_m V_{\text{air,ekf}}^2 \]  
(5.41)

with the air density \( \rho \) used from the original flight data.

**Attitude**

The attitude quaternions are converted to Euler angles for the final EKF output using the relationships

\[ \phi_{\text{ekf}} = \tan^{-1}\left(\frac{2(q_1q_2 + q_0q_3)}{q_3^2 + q_1^2 + q_2^2 + q_3^2}\right) \]  
(5.42)

\[ \theta_{\text{ekf}} = -\sin^{-1}(2(q_1q_3 - q_0q_2)) \]  
(5.43)

\[ \psi_{\text{ekf}} = \tan^{-1}\left(\frac{2(q_2q_3 + q_0q_1)}{q_3^2 + q_1^2 + q_2^2 + q_3^2}\right) \]  
(5.44)

### 5.2.8 EKF Initialisation and Tuning

The EKF is initialised with a sequence of steps as follows. The flight data file is edited to begin at steady, level flight. A long segment of straight and level trimmed flight is part of each flight plan. After the file is opened by the EKF, the time vector is extracted and the time step length is verified. If, for some reason, it is not the required 10ms, the filter will notice the user and quit. The pre-flight calibrations are applied to the magnetometer data. The accelerometer calibrations are applied during flight. The airdata calibrations are estimated during the filter run, which results in a more stable EKF output. Only for very noisy data sets the airdata calibrations are fixed to known values from a previous flight. Then the data is converted into SI units where required, and low-pass filtered with a cut-off frequency of typically 8 Hz. This removes all high frequency noise from the data before entering the filter. The final step is the filter initialisation, where the various required noise matrices are defined and the initial condition \( X_0 \) is set. Then the filter loop is started.

The initial conditions, being the state of the system at loop commencement, must be calculated carefully from the available sensor data. A bad initial condition introduces high levels of instability into the filter and it can take a long time for it to recover. Sometimes the algorithm even diverges. Hence, data from the reference IMU is used to initialise the filter as follows.
Chapter 5. Data Processing Software

**Position**

The ECEF data from the GPS is converted into the local NED coordinates using Eq. (5.32). Ten samples are averaged to reduce noise. This location is relative to the origin of the local NED coordinate system, which is either set to the pilot’s position on the ground. This allows for easy comparison between flight data sets.

**Velocities**

The GPS receiver reports ECEF axes velocities. These values have to be converted into body axes to be used as initial conditions. The attitude estimate of the reference IMU is used to generate the required rotation matrices. 10 samples of all data used is averaged to reduce noise.

**Attitude**

The reported attitude of the reference IMU is used as the initial attitude after being converted to quaternions. Again, ten samples are averaged for noise reduction.

**Wind**

The wind components are initialised as zero, unless previous runs with the same dataset or a run with data from a flight before or after the current one have yielded reliable information on the wind conditions at the time of flight. Then this information is used to initialise the wind states. Wrong initial conditions of the wind states are a major contribution to initial filter oscillations, because the airdata measurements will not be compatible with the inertial measurements if wind is present. A good wind estimate at the start ensures a smooth filter startup.

**Bias States**

The bias states are typically initialised as zero, unless reliable information about those numbers are known. These initial conditions can then be iteratively improved during filter tuning. This method will also identify non-observable error states, which will show up in the results as non-steady estimates with large standard deviations. These states can then be deactivated or set to a constant value, as it is the case for the angle of attack scale factor, which can be identified correctly in clean data but not in noisy data.

**Tuning Procedure**

The following general procedure has worked well for tuning the EKF to most data sets:

1. Set Q and R to the starting values derived from static ground tests. Set all bias and scale factor initial values to zero.
2. Disable all airdata measurements and associated sensor error states. This removes the influence of the wind and the additional noise from the airdata measurements and leaves just the inertial ‘subsystem’ operational. Also disable the magnetometers.
3. Iteratively estimate the observable sensor errors for the gyroscopes and accelerometers, using just the GPS measurements. Repeat this until the estimates are stable. If an error state estimate is very noisy or close to zero, it is best to set this state to zero and turn it off. This improves the stability of the remaining system.
4. Disable GPS measurements. This effectively turns the EKF into a dead reckoning system by integrating the state equations without the measurement feedback. Then test the dead reckoning performance of the inertial sensors. If the errors have been estimated correctly, the time before the state estimates for position, velocity and attitude diverge fully from the measurements should improve significantly. This means that the inertial part of the filter operates at optimum performance with the smallest possible error introduced into the filter by the inertial sensors.

5. Set all sensor error states for gyroscopes and accelerometers to the estimates values and turn the estimation of these states off.

6. Enable the airdata and magnetometers and associated bias states. Set the wind process noise to a high value to allow for large changes.

7. Estimate the airdata bias values and set them to constant when finished.

8. Reduce wind process noise until errors show up in other states to smoothe the wind estimates.

9. Re-run the filter with all sensor error states set constant and the wind process noise reduced as much as possible for the final data output.

During all these steps, the filter residual, that is the difference between the estimated measurements and the actual sensor data, and the time history of the error covariance matrix diagonal can be inspected to isolate problems. The filter residual should show measurement errors that are constrained and un-biased. If the errors become large, either a problem with the sensor exists, or the filter is diverging. The error covariances should always decrease when measurements are available [85]. In practise, the error covariances stabilise at a low level and oscillate slowly about the mean value. This is probably caused by the remaining uncertainty due to un-modelled errors, like structural vibrations, and imperfect sensors.

A final method to confirm the filter results is to run some preliminary parameter ID on the results and compare the results with the expected values. Care must be taken not to tune the filter to the expected results. Nevertheless, this is a helpful procedure, especially at the beginning of the tuning process, where completely wrong filter settings can quickly be identified from the resulting errors between the expected parameter values and the identified results. Once the order of magnitudes of the sensor variances were established, the filter was tuned without attempting any further parameter ID to avoid the ‘tune to result’ issue.

This procedure, as well as some example results will be discussed in chapter 15 using the recorded flight data. The wind tunnel data does not require EKF treatment, since the data is coming mainly from the calibrated reference IMU, which will be checked, together will all other sensors, in the sensor calibration chapter below.

5.3 System Identification Software

The toolbox for system ID developed for this project consists of various tools and algorithms sourced mainly from SIDPAC [22] and reference [74]. Three methods for system ID were used for this project: The equation error method (EQN) in the frequency domain,
Chapter 5. Data Processing Software

the output error method (OEM) in the time domain and the filter error method (FEM) in the time domain. These methods will be briefly introduced in this section, with the in-depth theory available in the two reference books as well as a wide body of literature. All methods have been set up such that they can operate with the standard \texttt{fdata} structure from SIDPAC and the same model definitions, which makes the results directly comparable. All time integrations are done with a fourth order Runge-Kutta methods for maximum accuracy. Any low pass filtering is done with the frequency domain methods from SIDPAC to avoid any time delays due to the filtering.

5.3.1 Equation Error Method

The term equation error method relates to a class of system ID methods which use the direct input-output relation of a dynamic system for the cost function minimisation. The most common method of this class is the least squares technique, which is also used in this thesis. Some references liberally mix the terms equation error and least squares methods and therefore for this work they refer to the same thing. The following description is limited to linear modelling, but there are extensions of the method for non-linear problems [74].

The least squares method is mathematically simple and non-iterative. This allows for fast processing times, which makes this method ideal for semi-real time applications [ ]. On the downside, the method uses only the input-output relation for the parameter estimates, which results in a direct sensitivity to measurement and modelling errors.

Assumptions

The least squares method assumes that

- Dependent or output variables measured with white noise only
- Independent or regressor variables are measured exactly
- Errors or residuals are uncorrelated with the independent variables and can be modelled as white noise

The requirement of perfect measurements of the regressors is very difficult to satisfy in reality, and the resulting errors are discussed below.

Method

As explained in reference [74], consider the simplest case of a vector of measurements of an output variable \( Y \) depending only on measurements of a single regressor \( X \). The regression equation for \( N \) measurements is then in matrix form

\[
Y = \theta X + \epsilon
\]  

where \( \theta \) is the unknown, constant parameter and \( \epsilon \) is the equation error caused by inevitable measurement and/or modelling errors.

The cost function to be minimised is the sum of the square of the residuals for each measurement \( k \), given as

\[
J(\theta) = \frac{1}{2} \sum_{k=1}^{N} \epsilon^2 = \frac{1}{2} (Y^T - \theta X^T) (Y - \theta X)
\]  

(5.46)
with the gradient with respect to the parameter $\theta$

$$\frac{\delta J(\theta)}{\delta \theta} = -Y^T X + \theta (X^T X)$$ \hspace{1cm} (5.47)

The minimum of the cost function can be found by setting the gradient to zero, which yields an estimate for the unknown parameter $\hat{\theta}$

$$\hat{\theta} = (X^T X)^{-1} X^T Y$$ \hspace{1cm} (5.48)

if the inverse of $(X^T X)$ exists. Hence, an estimate for the unknown parameter vector $\theta$ is obtained by a simple algebraic matrix inversion, if the problem is well conditioned.

**Properties**

In the case of error free measurements of the independent variables, the parameter estimates of the least squares method are unbiased. Since this is unrealistic in practical applications, reference [74] presents some investigations based on the bias and covariance of the equation error $\epsilon$. The result is that for systematic sensor errors the parameter estimates become biased, depending on the errors and noise in the regressors only. The variance of the estimates is affected by noise both on the regressors and the output variable. Turbulence takes the form of process noise and if this process noise is assumed white and no sensor errors are present, there is no effect on the parameter estimates. Hence, the equation error method can deal with atmospheric turbulence, as long as its spectrum approaches white noise, but will yield biased estimates due to measurement errors. How well this immunity to process noise works in practise probably depends on the manner of how the input and output variables are affected by the process noise. As long as the output contains the response of the system to the noise in the regressors, this might work well, but this needs to be established on a case to case basis.

A variation of the equation error method is to use the frequency domain. This allows for easier separation of noise and rigid body dynamics and is known to work better than using the time domain for imperfect data [1]. This method is the one used for this work.

A fundamental issue with the standard formulation of the least squares method is that Eq. (5.45) does not allow for constant pre-defined parameter values except zero, which are frequently required during this project due to parameter correlations. This limits the usability of the method for system ID of small scale aircraft, unless the model structure or the algorithm formulation is adapted to allow for these constant parameter values. Nevertheless the results of the equation error results will be included in the results where possible for comparison.

### 5.3.2 Output Error Method

The output error method (as well as the filter error method discussed next) belong to the class of maximum likelihood methods, which is based on probability theory and therefore deals with the statistical properties of the input and output data, as well as the residuals. The output error method is a subclass of the maximum likelihood methods, where it is assumed that there is negligible process noise. It is by far the most widely used method for aircraft parameter ID and is featured in a wide body of literature [1]. The output error
method is capable of linear and non-linear dynamic models and the latter is discussed in this section.

Assumptions

The assumptions for the output error method are ([74]):

- The input sequence is independent of the output. This excludes feedback control systems on potentially unstable aircraft.
- The measurement residuals are statistically independent and have zero mean and a covariance matrix $R$.
- The measured data contains only measurement noise and no process noise.
- The dynamic system must be excited such that the desired parameters are identifiable. This usually requires to excite the various modes of motion of the aircraft sufficiently.

Method

The output error method uses a formulation of the (potentially non-linear) dynamic system in state space form

$$\dot{x}(t) = f(x(t), u(t), b)$$

$$y(t) = g(x(t), u(t), b)$$  \hspace{1cm} (5.49a)

where $b$ are unknown bias parameters accounting for modelling errors and other unmodelled discrepancies, and

$$z(t) = y(t) + G\nu(t)$$

$$x(t_0) = x_0$$  \hspace{1cm} (5.50)

where $z(t)$ are the measured system outputs and $G\nu$ is the measurement noise. The maximum likelihood cost function to be minimised is derived in reference [74] as

$$J(\Theta, R) = \frac{1}{2} \sum_{k=1}^{N} (z(t_k) - y(t_k))^T R^{-1} (z(t_k) - y(t_k)) + \frac{N}{2} \ln(det(R)) + \frac{N n_y}{2} \ln(2\pi)$$  \hspace{1cm} (5.52)

The last term is constant for a given sample size $N$ and a constant parameter number $n_y$ and can be neglected during the optimisation of the cost function. The evaluation of the cost function requires the calculation of the estimated system outputs $y(t_k)$, which is obtained from integrating the system model in time using Eqs. (5.49) and the current parameter vector $\Theta$.

Typically the measurement covariance matrix $R$ is unknown and has to be estimated in conjunction with the parameter vector. This is usually done as follows: In a first step, a maximum likelihood estimate for $R$ can be obtained from

$$R = \frac{1}{2} \sum_{k=1}^{N} (z(t_k) - y(t_k))(z(t_k) - y(t_k))^T$$  \hspace{1cm} (5.53)

which can be substituted into the cost function as

$$J(\Theta) = \frac{N n_y}{2} + \frac{N}{2} \ln(det(R)) + \frac{N n_y}{2} \ln(2\pi)$$  \hspace{1cm} (5.54)
Since the first and last term of the cost function are constant for the assumed model structure, they can be neglected. Hence, the cost function reduces to

\[ J(\Theta) = \text{det}(R) \]  

(5.55)

The task of the output error method is now to optimise the parameter vector \( \Theta \) such that \( J \) is minimised. This is typically done with a Gauss-Newton algorithm, but there are many other methods to perform this optimisation, depending on the problem at hand [74]. In most cases the matrix \( R \) is assumed diagonal, which simplifies the problem. As with any non-linear optimisation problem, a set of initial estimates of the unknown parameters is required, as well as the initial conditions \( x_0 \) of the chosen state variables. These can be found by trial and error, by using results of other experiments like wind tunnel data or they can be estimated with the equation error method from above. The equation error method is limited to linear models, but for the low angle of attack manoeuvres used in this project, its parameter estimates will be close enough to the true values to start up an non-linear output error estimation.

**Properties**

The output error method has several favourable properties [74]. It can be shown that its parameter estimates are unbiased and they converge in probability to the true value. The estimates from multiple datasets are asymptotically normally distributed and their theoretically achievable best accuracy is described by the lower Cramer-Rao bounds, which are an useful tool to judge the confidence in the estimated results.

For this project with its small scale fixed wing aircraft, the assumption of no process noise (or turbulence) is almost always violated to some extend, and it remains to be seen how much the estimated parameters are affected by this fact. Therefore the data will also be analysed with the final and most capable method, the filter error method, to judge the performance of the output error method being applied to flight test data of a very small aircraft.

### 5.3.3 Filter Error Method

The filter error method is an extension of the output error method, which was developed to be able to account for process noise (atmospheric turbulence) during the parameter ID process. The method still belongs to the same class of maximum likelihood methods, but due to the allowance for the process noise, it is considerably more involved than the output error method discussed before. The presence of process noise turns the system model into a stochastic process, which requires a suitable state estimator to calculate the system outputs required for the cost function, which is the same as Eq. (5.52) for the output error method.

**Assumptions**

The filter error method operates under the same assumptions as the output error method, except it allows for the presence of process- and measurement noise. Also, as a consequence of the algorithm formulation, the filter error method is capable of estimating parameters of unstable aircraft, while the output error method may fail in this case.
Method

The filter error method used for this project is the general formulation for non-linear system models, as derived in reference [74]. Due to the presence of the process noise the system model of Eqs. (5.49) becomes a stochastic process by adding this term to the state rate equation

\[
\dot{x}(t) = f(x(t), u(t), b) + Fw(t) \tag{5.56a}
\]

\[
y(t) = g(x(t), u(t), b) \tag{5.56b}
\]

where \(b\) are unknown bias parameters accounting for modelling errors and other unmodelled discrepancies as before, and

\[
z(t) = y(t) + G\nu(t) \tag{5.57}
\]

and \(x(t_0) = x_0 \tag{5.58}\)

where \(F\) and \(G\) are the process- and measurement noise distribution matrices, respectively. For the output error method the system model could simply be integrated in time to obtain the system outputs \(y(t)\). Here, this is not possible any more, and a suitable state estimator must be used. For the non-linear case this is typically done with an extended Kalman filter (EKF), while a linear Kalman filter can be used for linear systems. In addition to the mathematical complexity of an EKF, the identification problem is further complicated by the required, but unknown process- and measurement noise covariance matrices, as well as the Kalman gain matrix \(K\). The only viable solution appears to be to use an estimator for \(R\) similar to Eq. (5.53) and then include the entries of the process noise distribution matrix \(F\), which is typically assumed diagonal, into the parameter vector to be estimated. With this theoretical groundwork it is then possible to formulate an iterative algorithm functionally similar to the output error method, but the result is much more complicated. The details of the mathematics can be found in reference [74]. In practise, it is now required to specify initial conditions for the unknown parameters, the noise covariance matrix \(R\) and the diagonal entries of \(F\) and experience has shown that especially the initial conditions for \(F\) are critical for good convergence. Bad starting values there lead to divergence of the method in most cases.

Properties

The filter error method generally produces a good fit of the estimated system outputs to the measured data. This can be quite deceptive, since there is no way of telling whether this good agreement is caused by the process noise or the parameter estimates. The filter error method is also much more sensitive to the initial parameter estimates, as discussed above, and convergence problems occur often. This, together with significantly increased computational burden, leads to much longer processing times. Another observation at this stage is that the filter error method does not seem to work well with the output data from the EKF. Convergence and stability are reduced, if compared to runs on the raw data. This might be caused by the process noise of the data already being taken care of by the EKF run, which is, however, required to ensure data compatibility.
6. System Calibration

Each sensor of the UAV mainframe requires careful calibration to deliver optimum performance. Some of those calibrations could be done before flight in the lab and others were left to the EKF run during post processing. This chapter discusses all pre-flight calibration methods used for this project. No specialised equipment for these tasks was available, so much of the work was concerned with developing methods that would yield acceptable results with the resources available.

6.1 Sensor Error Sources

For stability and control system ID, the equations of motion describe the change in acceleration and attitude of the aircraft due to a control surface input. The primary data required for this task are the airdata-, acceleration- and rotation rate measurements as well as the control surface deflections. For the data compatibility checking and instrumentation error determination, using the EKF, as described in Section 5.2, additional data is required. Those are the position and velocity of the aircraft, as well as the orientation of the magnetic field vector with respect to the aircraft.

Tables 6.1 and 6.2 list the variables measured by the UAV mainframe, together with their units, expected error sources and calibration methods. The control surface deflections are not used in the EKF. The aircraft attitude is not directly measured, but calculated by the EKF from the input data. The accuracy of this estimate depends on the quality of the EKF performance and it is not possible to name specific errors for the attitude estimates.

The errors in the airdata measurements occur mainly due to the location of the probe close to the wing leading edge, where the sensors are affected by the wing upwash, and due to the fact that the probe is disassembled for transport each time, which
### Table 6.1: System ID variables

<table>
<thead>
<tr>
<th>Input</th>
<th>Symbol</th>
<th>Units FDR</th>
<th>Units EKF</th>
<th>Expected Errors</th>
<th>Ground cal.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alt</td>
<td>alt</td>
<td>m</td>
<td>m</td>
<td>bias, scale</td>
<td>-</td>
</tr>
<tr>
<td>Airspeed</td>
<td>$V_T$</td>
<td>m/s</td>
<td>m/s</td>
<td>bias, scale, inflow angle</td>
<td>Wind tunnel</td>
</tr>
<tr>
<td>AoA</td>
<td>$\alpha$</td>
<td>deg</td>
<td>rad</td>
<td>alignment, bias, scale</td>
<td>Wind tunnel</td>
</tr>
<tr>
<td>Sideslip</td>
<td>$\beta$</td>
<td>deg</td>
<td>rad</td>
<td>alignment, bias, scale</td>
<td>Wind tunnel</td>
</tr>
<tr>
<td>Accel X</td>
<td>$a_x$</td>
<td>g</td>
<td>m/s$^2$</td>
<td>alignment, bias, scale</td>
<td>Lab</td>
</tr>
<tr>
<td>Accel Y</td>
<td>$a_y$</td>
<td>g</td>
<td>m/s$^2$</td>
<td>alignment, bias, scale</td>
<td>Lab</td>
</tr>
<tr>
<td>Accel Z</td>
<td>$a_z$</td>
<td>g</td>
<td>m/s$^2$</td>
<td>alignment, bias, scale</td>
<td>Lab</td>
</tr>
<tr>
<td>Gyro X</td>
<td>$\omega_x$</td>
<td>deg/s</td>
<td>rad/s</td>
<td>alignment, bias, scale</td>
<td>Check only</td>
</tr>
<tr>
<td>Gyro Y</td>
<td>$\omega_y$</td>
<td>deg/s</td>
<td>rad/s</td>
<td>alignment, bias, scale</td>
<td>Check only</td>
</tr>
<tr>
<td>Gyro Z</td>
<td>$\omega_z$</td>
<td>deg/s</td>
<td>rad/s</td>
<td>alignment, bias, scale</td>
<td>Check only</td>
</tr>
<tr>
<td>Roll</td>
<td>$\phi$</td>
<td>deg</td>
<td>rad</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Pitch</td>
<td>$\theta$</td>
<td>deg</td>
<td>rad</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Yaw</td>
<td>$\psi$</td>
<td>deg</td>
<td>rad</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Elevator</td>
<td>$\delta_e$</td>
<td>deg</td>
<td>-</td>
<td>bias, scale</td>
<td>Lab</td>
</tr>
<tr>
<td>Aileron</td>
<td>$\delta_a$</td>
<td>deg</td>
<td>-</td>
<td>bias, scale</td>
<td>Lab</td>
</tr>
<tr>
<td>Rudder</td>
<td>$\delta_r$</td>
<td>deg</td>
<td>-</td>
<td>bias, scale</td>
<td>Lab</td>
</tr>
</tbody>
</table>

### Table 6.2: Additional EKF input variables

<table>
<thead>
<tr>
<th>Input</th>
<th>Symbol</th>
<th>Units FDR</th>
<th>Units EKF</th>
<th>Expected Errors</th>
<th>Ground cal.</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPS lat</td>
<td>lat</td>
<td>deg</td>
<td>m</td>
<td>GPS limitations</td>
<td>-</td>
</tr>
<tr>
<td>GPS lon</td>
<td>lon</td>
<td>deg</td>
<td>m</td>
<td>GPS limitations</td>
<td>-</td>
</tr>
<tr>
<td>GPS alt</td>
<td>$\text{gpsalt}$</td>
<td>m</td>
<td>m</td>
<td>GPS limitations</td>
<td>-</td>
</tr>
<tr>
<td>GPS Vx</td>
<td>$V_x$</td>
<td>m/s</td>
<td>m/s</td>
<td>GPS limitations</td>
<td>-</td>
</tr>
<tr>
<td>GPSVy</td>
<td>$V_y$</td>
<td>m/s</td>
<td>m/s</td>
<td>GPS limitations</td>
<td>-</td>
</tr>
<tr>
<td>GPS Vz</td>
<td>$V_z$</td>
<td>m/s</td>
<td>m/s</td>
<td>GPS limitations</td>
<td>-</td>
</tr>
<tr>
<td>Mag x</td>
<td>$B_x$</td>
<td>Gauss</td>
<td>Gauss</td>
<td>alignment, hard- and soft iron</td>
<td>Lab</td>
</tr>
<tr>
<td>Mag y</td>
<td>$B_y$</td>
<td>Gauss</td>
<td>Gauss</td>
<td>alignment, hard- and soft iron</td>
<td>Lab</td>
</tr>
<tr>
<td>Mag z</td>
<td>$B_z$</td>
<td>Gauss</td>
<td>Gauss</td>
<td>alignment, hard- and soft iron</td>
<td>Lab</td>
</tr>
</tbody>
</table>
potentially introduces bias errors in the flow angle data. Calibration methods used for
the airdata probe include wind tunnel tests and numerical studies. The accelerometers
are mainly affected by alignment errors as well as bias and scale factors due to the
limited accuracy of these low cost sensors. MEMS accelerometers are also affected by
temperature changes, which cause these bias and scale factors to drift. The MEMS
gyrosopes suffer from similar error sources as the accelerometers, as well as a typically
large bias error (up to 360 degrees per minute, depending on the chip quality) due to
drift of the sensor. One advantage of this large bias error is that it is easily observable
and corrected with the EKF. On the other hand, raw gyroscope data without treatment is
unusable for any analysis. The bias value is temperature dependent and varies over time.
Hence, just reading the bias at the start of an experiment and using that value without
an EKF to correct the gyroscope bias will lead to error. The feedback sensors o the
control surfaces will have a bias error due to the assembly orientation and potential scale
factors caused by the linkage to the surface. These can be easily corrected by ground
measurements. GPS measurements are subject to the limitations of the GPS system. The
accuracy depends on the quality of the receiver as well as the satellite constellation at the
time of the flight. No calibration is possible, but using the manufacturer provided error
statistics in the EKF will correct these measurements. The magnetometers are the most
difficult sensors to calibrate. They are distorted by magnetic fields and nearby magnetic
materials, as well as alignment errors. Magnetic fields inside the aircraft are strongly
dependant on the electrical environment and the motor throttle setting, all of which may
be time variant. It is therefore beneficial to place the magnetometer as far away from any
disturbance as possible and then apply the correction methods discussed in this chapter.

Another source of error that is specific to the use of small MEMS sensors it depicted in
Figure 6.1. Due to possible variations in the amount of solder applied during the circuit
board manufacturing process, these small integrated circuits can become misaligned
with the board as shown in the Figure. This adds to the sensor alignment error and
has to be taken into account. Simply mounting the circuit board in a known orientation
does not guarantee correct sensor alignment. During this project up to 5 degrees offset
were observed for some components. This would clearly degrade the quality of the data
from this sensor and therefore has to be treated. The elaborate method to determine the
alignment of the wing tip magnetometer, as discussed in this chapter, is a result of this
alignment problem.

![Figure 6.1: Sensor chip orientation error due to uneven solder distribution](image-url)
Chapter 6. System Calibration

The final source of error is the sensor sampling timing. All methods used for data processing require a constant time step and all sensors should be sampled at the same time. This was discussed in the requirements of the UAVmainframe and has been a key driver for the concept and design of the system. A verification of the outcomes of this effort is presented next.

6.2 Timing Accuracy

The EKF and all other processing methods expect a constant time step length of 10ms for the sensor data sampling. This is the rate at which the state equations are integrated. Achieving this constant time step length with a non-realtime operating system was a major area of development during this project. The entire design of the UAVmainframe is based on methods to achieve this constant sample rate. This section evaluates the results of this effort.

Figures 6.2 and 6.3 show a comparison between the timing accuracy with and without all system optimisations for real-time operations. Figure 6.2 shows the timestep error accumulation over time and the actual length of each timestep for operation without optimisations. It can be clearly seen that, if the UAVmainframe process competes with the rest of the system for CPU time, the timing accuracy is not very good. Some time steps take several seconds instead of 10 ms due to the process being blocked from executing by the Linux scheduler. The accumulated error is almost a minute over a 6 minute flight. This is clearly not acceptable for a flight data logger or control system.

Figure 6.3 shows the timing behaviour with the multi-threaded UAVmainframe software, running the timing loop with the highest system priority. The timestep length is now nearly constant at 10 ms with only a single small outlier during a 9 minute flight. As a consequence the accumulated error is very close to zero. This is quite a dramatic improvement.
6.2 Timing Accuracy

Figure 6.3: UAVmainframe timing result optimised

improvement over the non-optimised code and enables the UAVmainframe to be used as planned. It should be noted that these improvements were all achieved using standard techniques available in the Linux OS and no additional software was required.

A verification of the sensor timing accuracy can be done by using the precision current monitoring capabilities of the UAVmainframe. Figure 6.4 shows the rudder response to a command and some selected sensor readings. There is some significant delay between the command and the rudder actually moving, which is caused by the servo motor processing the input signal.

To establish when the servo motor starts moving, the current drawn by it is plotted in the Figure. As soon as the current rises, the servo starts moving the rudder and the

Figure 6.4: UAVmainframe sensor data timing, with a detailed view of (A) on the right
aircraft starts yawing. In the detailed view on the right, it can be observed that current, surface feedback, yaw rate and wingtip acceleration in the x-axis all start moving at the same time step. The same can be done for the other control surfaces with similar results. This confirms the excellent timing accuracy between the sensor readings of the UAVmainframe. Due to the parallel processing architecture of the system this level of accuracy is not affected by the number of sensors sampled. This allows the system to be expanded to potentially several hundred channels if required. The timing accuracy of the parallel design also ensures that all sensors are read within the first 2ms of the 10ms frame, which effectively increases the time resolution even further and allows for higher sample rates in the future, if required.

6.3 Inertial Sensor Calibration

6.3.1 Accelerometers

The various accelerometers of the UAVmainframe all require calibration for bias and scale errors, as well as corrections for alignment with the body axes. These errors can be calibrated prior to flight. Temperature drift and other factors causing bias shifts are treated by the EKF for each individual flight during post-processing. The accelerometer calibration uses readings in all six main directions and a least squares calculation to derive the correction matrices. The assumed sensor model is

\[
A_{\text{cal}} = \begin{bmatrix} AR_{11} & AR_{12} & AR_{12} \\ AR_{21} & AR_{22} & AR_{23} \\ AR_{31} & AR_{32} & AR_{33} \end{bmatrix} \times \begin{bmatrix} ASC_{11} & 0 & 0 \\ 0 & ASC_{22} & 0 \\ 0 & 0 & ASC_{33} \end{bmatrix} \times \begin{bmatrix} A_{mx} - A_{bx} \\ A_{my} - A_{by} \\ A_{mz} - A_{bz} \end{bmatrix} \]

(6.1)

where \( AR \) are the elements of the rotation matrix, \( ASC \) the scale factors and \( A_b \) the bias parameters. The sensor model can be rewritten as:

\[
A_{\text{cal}} = \begin{bmatrix} ACC_{11} & ACC_{12} & ACC_{12} \\ ACC_{21} & ACC_{22} & ACC_{23} \\ ACC_{31} & ACC_{32} & ACC_{33} \end{bmatrix} \times \begin{bmatrix} A_{mx} + AB_1 \\ A_{my} + AB_2 \\ A_{mz} + AB_3 \end{bmatrix} \]

(6.2)

or

\[
\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} = \begin{bmatrix} A_{mx} \\ A_{my} \\ A_{mz} \end{bmatrix} + \begin{bmatrix} AB_1 \\ AB_2 \\ AB_3 \end{bmatrix} \]

(6.3)

or in matrix form

\[
Y = m \times X \]

(6.5)

where \( Y \) is a matrix of the expected values for all six orientations and \( m \) a matrix of acceleration component measurements at each of the six orientations, combined into a
single matrix and a column of ones added as per definition. Then the unknown matrix $X$ can then be found from the least square method:

$$X = (m^T m)^{-1} \times m^T \times Y$$ (6.6)

A typical dataset is shown in Table 6.3, with the calibration performed in $mg$. The method is very flexible because it automatically matches the desired output units to the input units. For the analogue accelerometers, for example, the input data was specified directly in Volts as measured by the A/D converter. This calibrates the sensor together with the A/D converter in a single step. This sensor is mounted vertically inside the wingtip on the main spar web. The data in the Table shows that there is considerable misalignment between the sensor axes and the body axes of the UAV. There are also scaling errors, where the measurement of $g$ is larger or smaller than the expected $1000mg$.

### Table 6.3: Accelerometer calibration input data

<table>
<thead>
<tr>
<th>Orientation</th>
<th>Y [mg]</th>
<th>m [mg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z down (normal pos.)</td>
<td>[0 0 -1000]</td>
<td>[-968.7 -29.3 -75.5 1]</td>
</tr>
<tr>
<td>Z up (upside down)</td>
<td>[0 0 1000]</td>
<td>[1056.8 -31.8 -58.3 1]</td>
</tr>
<tr>
<td>Y down (right side)</td>
<td>[0 -1000 0]</td>
<td>[49.2 -34.0 -1074.6 1]</td>
</tr>
<tr>
<td>Y up (left side)</td>
<td>[0 1000 0]</td>
<td>[22.1 1.2 946.7 1]</td>
</tr>
<tr>
<td>X down (nose down)</td>
<td>[-1000 0 0]</td>
<td>[53.7 -1040.7 -60.4 1]</td>
</tr>
<tr>
<td>X up (tail down)</td>
<td>[1000 0 0]</td>
<td>[54.1 979.9 -80.7 1]</td>
</tr>
</tbody>
</table>

The results of this particular calibration is listed in Eq. (6.8). The rotation matrix $\text{ACC}$ contains the large rotation of the sensor into the body axes, visible from the large entries in the matrix not being on the diagonal, and some small corrections for cross-coupling in the off-diagonal terms. The bias vector $\text{AB}$ contains entries of considerable size, which would introduce sizeable errors if not treated properly. These matrices are written into the configuration file of the UAVmainframe and are applied automatically during flight. Hence, the flight test engineer can observe calibrated accelerations in real time via the telemetry link.

$$\text{A}_{\text{cal}} = \text{ACC} \times \text{A}_m + \text{AB}$$ (6.7)

$$\begin{bmatrix}
0.0015 & 0.9895 & -0.0173 \\
-0.0083 & 0.0099 & 0.9892 \\
0.9871 & 0.0001 & 0.0133
\end{bmatrix} = \begin{bmatrix}
24.2881 \\
67.0312 \\
-43.0666
\end{bmatrix}$$ (6.8)

A visualisation of the above calibration vs. the raw data is shown in Figure 6.5, for a different sensor, by plotting the elements of $m$ against the elements of $\text{A}_{\text{cal}}$. Bias and scale factors, as well as miss-alignments are clearly visible in the raw data. The
calibrated data shows no such errors. There is a remaining error of a few $mg$ on some axes, where the least square method was not able to fit the data perfectly. These errors are treated within the EKF after the flight, together with temperature induced bias shifts. Overall the data quality of these tiny and cheap MEMS accelerometers is very good and very stable. Even after a year, there were only minor changes to the calibration matrices during a re-calibration.

Figure 6.5: Accelerometer calibration results
6.3.2 Gyroscopes

Gyroscopes are typically calibrated on a rate table, which subjects the sensor to a known rotation rate. Such a device was not available for this project, so no gyroscope calibration was possible. But since the gyroscopes data was taken from the reference IMU, which was factory calibrated, this is not expected to be a major problem. In order to test the data quality, the gyroscopes can be checked against the wind tunnel gimbal angle encoders of the dynamic test rig, described in Part V. The data from a lateral test session is plotted in Figure 6.6. The angle encoder data was differentiated and plotted against the reference IMU gyroscope data recorded simultaneously. The Figure shows that the residuals are unbiased, meaning that the bias estimation of the reference IMU is correct, and they form a random noise floor of small magnitude, compared to the input signal. One reason for this noise floor is the limited resolution of the angle encoders of 12 bit or 0.08 degrees. There are a few larger errors in the pitch axis, most likely caused by small errors in the pitch gyro bias estimate due to insufficient pitching motion of the IMU. This is unlikely to happen during flight. Overall this is a good result, and the gyroscopes of the reference IMU can be used with good confidence for this project.

Figure 6.6: Gyroscope rotation rates vs. gimbal rates
6.3.3 Magnetometers

The magnetometer calibration is the most involved method of them all, especially since no reference field generator was available. Calibrating the magnetometers well is also very important, because they are the only sensor that can generate independent data about the attitude of the aircraft. These measurements therefore help stabilising the rotational part of the EKF process, which otherwise would mainly depend on the correct integration of the rotation rates.

The magnetometers measure the size and direction of the earth’s magnetic field, which is a relatively weak signal to measure and therefore data is easily affected by the following factors:

- Hard iron distortion due to the presence of external magnetic fields and hard iron materials near the sensor.
- Soft iron distortion due to the presence of soft iron materials like batteries near the sensor.
- Sensor installation attitude with respect to the body frame of the aircraft.

These error sources result in the following magnetometer sensor model:

\[
M_{\text{cal}} = M_R \times M_{\text{SC}} \times M_{SI} \times [M_m - M_B] \tag{6.9}
\]

where \( M_R \) is the sensor rotation matrix, \( M_{\text{SC}} \) a scaling matrix, \( M_{SI} \) the soft iron matrix and \( M_H \) the hard iron or bias vector. Expanded, Eq. (6.9) becomes

\[
M_{\text{cal}} = \begin{bmatrix}
M_{R11} & M_{R12} & M_{R12} \\
M_{R21} & M_{R22} & M_{R23} \\
M_{R31} & M_{R32} & M_{R33}
\end{bmatrix}
\times
\begin{bmatrix}
MSC_{11} & 0 & 0 \\
0 & MSC_{22} & 0 \\
0 & 0 & MSC_{33}
\end{bmatrix}
\times
\begin{bmatrix}
MSI_{11} & MSI_{12} & MSI_{12} \\
MSI_{21} & MSI_{22} & MSI_{23} \\
MSI_{31} & MSI_{32} & MSI_{33}
\end{bmatrix}
\times
\begin{bmatrix}
M_{mx} - M_{bx} \\
M_{my} - M_{by} \\
M_{mz} - M_{bz}
\end{bmatrix} \tag{6.10}
\]

The magnetometer calibration method has two distinct parts, the hard- and soft iron calibration to determine \( M_{SI} \) and \( M_H \) as well as \( M_{SC} \), and then the remaining alignment with the body axes of the aircraft to estimate \( M_R \).

Hard- and Soft-Iron Calibration

The measurements of a three axis magnetometer represent the size and orientation of the magnetic field vector with respect to the sensor. If the ideal sensor readings are normalised by the local magnetic field strength and the sensor is rotated, all measurements should lie on an unit circle. The presence of additional magnetic fields and ferro-magnetic materials near the sensor distort that unit circle into an ellipse and offset its centre from the origin. The task of the hard- and soft-iron calibration is to determine a mapping of that ellipse back onto the unit circle, which can then be applied to the raw measurement data [86, 87]. The two references describe an algorithm to do that, mapping an ellipse to the data using a least square method and then use the Eigenvalues of that ellipse to determine the transformation matrix for the mapping onto a sphere.
To calibrate a magnetic sensor, data needs to be recorded during rotations of the sensor about all three axes. This creates measurements which are distributed all over the ellipse and constrain its shape and dimensions. Ideally, the sensor is installed in its correct location and the environment around it is as close as possible to the flight condition to account for all occurring disturbances. In order to demonstrate the calibration method, a dataset with very large magnetic distortions was prepared. This was achieved by mounting a magnet close to the sensor to create hard iron disturbances and by placing a large, non-magnetic steel bolt near the sensor for the soft iron distortions. Figure 6.7 shows the resulting data with the ellipse fitted to it, as well as the distorted ellipse with respect to the desired unit circle. Note that in order to plot the two objects into one figure in Matlab, it was necessary to reverse the offset of the ellipse. It is placed at the origin and the unit circle is located at the offset coordinates of the sensor. The figure shows a large offset of the ellipse from the circle, caused by the magnet near the sensor. The shape of the ellipse is the result of the presence of the steel bolt near the sensor.

The result of the calibration is shown in Figure 6.8, where the data is mapped onto the unit sphere. Due to the magnitude of the distortions in this test case, this is not perfect. The colour coding of the data, which illustrates the distance from the origin of each data point, shows that there is a remaining error of 5% or less for most data points. Realistic data has a much smaller remaining fitting error, as will be shown later.

Figure 6.9(left) shows the ellipse fitted to the raw data from wing tip sensor without artificial disturbances. Comparing the plot to Figure 6.7, the ellipse is much closer in shape to the unit circle and the hard iron offset is much smaller. Figure 6.9(right) illustrates the corrected data mapped onto the unit sphere. The colours indicate that the remaining calibration error is smaller than the error from the large distortion test.
Chapter 6. System Calibration

Figure 6.8: Corrected data of the large distortion experiment

Figure 6.9: Raw (left) and calibrated (right) data for the undisturbed wing tip magnetometer with most data points within 2-3% of the target magnitude of one. The remaining error is caused by sensor noise and non-linearities, but its magnitude is acceptably small and therefore the matrices generated can be used to correct the raw data of the wing tip magnetometer.
A different visualisation method is plotting the data against time and to include the expected measurement to the plot to illustrate the remaining sensor errors. The expected orientation of the magnetic field vector in body axes can be obtained from rotating the local field vector in NED axes into the body axes of the aircraft, using the rotation matrix obtained from the attitude estimation of the reference IMU.

$$B_{\text{exp}} = L_{bc}B_{\text{ref}}$$ (6.11)

Figure 6.10 shows the raw data of the calibration test against time. In the residuals on the right, the reference IMU’s magnetometer (blue) appears to be well calibrated, at least compared to the other two sensors. Those show significant hard- and soft iron distortions, which require treatment. Figure 6.11 shows the same sensor data with the hard- and soft iron calibrations applied. The residuals of the two external sensors have reduced considerably, but are not as small as the reference IMU. This is an indication of additional alignment errors of the sensor boards and the sensor chips itself with the body axes of the UAV. Especially the wing tip sensor shows a large, remaining bias error in $Y$ after the hard- and soft iron calibration. In these tests the IMU magnetometer performs very well, so the question might be why it is not the preferred sensor over the other two. The reason is that the IMU is located close to the battery and propulsion system, with its changing disturbances based on engine throttle setting. This cannot be tested on the ground because rotating the aircraft with the propeller going at high speed is too dangerous. During the EKF runs with flight data, the wing tip sensor appears to perform much better than the IMU because it is located far away from the large magnetic disturbances inside the fuselage. Therefore it is beneficial to complete the calibration of the other two sensors by determining the installation attitude of them.
Chapter 6. System Calibration

Figure 6.11: Magnetometer data with hard- and soft-iron corrections applied

Installation Attitude

Figure 6.11 indicated that there is a remaining alignment error of the two external magnetometers that requires further treatment. Typically, this would be done with a reference magnetic field generated by a Helmholtz coil, where a known field direction is used to calibrate the measurements. Such a device was not available, so a different method was devised. This method uses the expected field orientation that was generated before, which is based on the attitude estimate of the IMU. Since there are only pure rotations involved, it can be expected that these attitude estimates are close to the truth and this method will therefore generate a valid calibration. Data for this final step was recorded on the flight test site, far away from any metallic structure for best accuracy.

To perform the alignment estimation, equation 6.11 can be written in its three components. This results in three equations of the form

\[ B_x = L_{x1}B_{mx} + L_{x2}B_{my} + L_{x3}B_{mz} \]  

where \( R_{xx} \) are the entries of a row of \( L_{be} \). The measured magnetic field data of each sensor can then be fitted to the expected field direction by a least square method and the rotation matrix assembled row by row. Applying the generated rotation matrix to the sensor measurements yields the results plotted in Figure 6.12.

After the axis alignment, the match between the three sensors and the expected readings is nearly perfect, with only some small remaining errors. These errors are probably caused by a remaining imperfections in the hard- and soft iron calibration. It can also be seen that the magnetometer of the IMU is much noisier than the one located at the wing tip. Figure 6.13 shows a different data set from an actual flight with the
same calibrations applied to the magnetometers. Figure 6.14 shows that the calibration appears to be universally valid, with the fit quality equally good as the reference data set of Figure 6.12.

This concludes the magnetometer calibration. It should be noted that the results depend on the quality of the attitude estimates of the reference IMU. The data compatibility EKF, discussed later, will allow to judge the overall alignment of the magnetic sensor data by inspecting the noise levels on these measurements. Finally, Table 6.4 lists the sensor orientations for all three magnetometers of the UAVmainframe. The large angle in $X$ for the wing tip sensor is a combination of the dihedral of the wing (3 degrees) and a misalignment of the sensor chip with the circuit board due to uneven amounts of solder, as discussed before.

![Figure 6.12: Magnetometer data with all calibrations applied vs. expected reading](image)

<table>
<thead>
<tr>
<th></th>
<th>$\Delta X$ [deg]</th>
<th>$\Delta Y$ [deg]</th>
<th>$\Delta Z$ [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>CG</td>
<td>0.25</td>
<td>-0.37</td>
<td>-0.39</td>
</tr>
<tr>
<td>Wingtip</td>
<td>-6.32</td>
<td>2.11</td>
<td>-0.16</td>
</tr>
<tr>
<td>Tail</td>
<td>1.43</td>
<td>0.21</td>
<td>0.47</td>
</tr>
</tbody>
</table>
Figure 6.13: Raw magnetometer readings from flight data

Figure 6.14: Calibration verification using the flight data set from Figure 6.13
6.4 Airdata

The airdata measurements are static pressure, dynamic pressure, angle of attack and angle of sideslip. The altitude and airspeed are derived from static and dynamic pressure measurements of the pitot tube installed on the airdata probe shown in Figure 6.15. The two angles are measured directly using wind vanes installed perpendicular to each other on the airdata probe. The Figure also shows the airdata sensor card installed on the UAVmainframe, together with the associated plumbing. The static pressure is connected to the altimeter and the differential pressure sensor via a Y-splitter, as shown in the Figure, to obtain the correct pressure from the static port of the pitot tube instead of using the cabin pressure for the absolute pressure reading. The cabin pressure can be quite different from the ambient static pressure due to the flow around and through the body and this would introduce errors that cannot be corrected for.

![Figure 6.15: UAVmainframe airdata probe and sensors with plumbing](image)

6.4.1 Absolute Pressure and Air Density

The air density, used in the dynamic pressure calculations for the airspeed and in the aerodynamic coefficients, can be determined by the relationship

\[
\rho = \frac{P}{RT}
\]

(6.13)

where \( P \) is the static pressure, \( R \) the specific gas constant for air (287.05 J/(Kg K)) and \( T \) the air temperature in Kelvin. Both required measurements, \( P \) and \( T \), are performed by the absolute pressure sensor. The datasheet gives an accuracy of this sensor as \( \pm 1.5 \text{ Pa} \) and \( \pm 0.8 \text{ deg. C} \). In flight, due to the aerodynamic noise, the absolute pressure measurement accuracy is approximately \( \pm 10 \text{ Pa} \). The temperature is measured in the cabin and is therefore assumed to be slightly higher than ambient due to sun radiation. No reference data was available, but it is believed that the measured temperature is with \( 5 \text{ deg. C} \) of the correct value, because all flights take place in the early morning where
the sun intensity is still low. This leads to an uncertainty in the estimated air density of approximately 2%. Given the small size of this uncertainty as compared to the other errors expected due to the small scale of the aircraft and the grade of sensors used, and the difficulty in improving the air density estimate, no further action was taken. In the future, it is planned to extend the EKF process model with an atmospheric model to estimate the air density that way.

6.4.2 Altitude

The altitude above the airfield can be calculated from the equation given in the sensor data sheet as

\[ \text{Alt} = 44330 \left(1 - \left(\frac{P}{P_{\text{ref}}}\right)^{1/5.255}\right) \]  \hspace{1cm} (6.14)

with \( P_{\text{ref}} \) being the reference pressure at ground level, typically the surface air pressure at the airfield. This altitude is used only for flight monitoring and in the EKF, but not for the system ID process. It will be shown later that the altitude estimates match the other data quite well.

6.4.3 Dynamic Pressure and Airspeed

The airspeed can be calculated by

\[ V_{\text{air}} = \sqrt{\frac{2q}{\rho}} \]  \hspace{1cm} (6.15)

where \( q \) is the dynamic pressure measured by the pitot tube. The pitot tube was calibrated in the department’s very clean 4x3 wind tunnel against a water manometer. Despite the big lump of the probe closely behind the static ports there is virtually no error in the measurements. This is probably due to the slow flight speeds and the resulting low dynamic pressure changes from that lump. Under these conditions the back pressure from the lump is probably too small to cause an error in the readings. Figure 6.16 shows the pitot probe calibration curve and its dependency on angle of attack. There is only 0.1m/s error in the speed measurement at the reference flight speed of 20m/s. The dependency on angle of attack is only 0.2 m/s in the angle range used for the flight test inputs. Hence, the measurement errors are most likely smaller than the sensor noise in flight, and no calibrations were applied to the airspeed measurements. The remaining bias error, caused mostly by sensor drift due to vibrations will be corrected by the EKF.

6.4.4 Angle of Attack

The custom designed airdata probe weights only 60 grams, yet that is enough to interfere with the structure of the aircraft. Originally the probe was mounted much further forward to reduce wing interference effects. This led to vibrations due to twist of the entire wing caused by the cantilevered mass out front. To reduce these vibrations, the probe was brought closer to the wing leading edge, with the angle of attack vane now less than half a chord length away from the leading edge. This arrangement requires extensive corrections but is necessary from a practical point of view. Another reason for this arrangement is that a longer probe is more likely to be damaged during an imperfect landing.
The angle of attack vane is influenced by three effects during flight. Firstly, because of the off CG position, there will be errors caused by the roll- and pitching motion of the aircraft \cite{22}. and secondly, as shown in Figure 6.17, the angle of attack vane is very close to the wing leading edge and will be strongly affected by the wing upwash. Corrections for these errors will be developed separately and tested against the dynamic wind tunnel data, where a reference is available in form of the gimbal attitude angle data. A linear combination of the separate corrections will then give the true angle of attack from the test data. Based on several test runs, it turns out that the results of the EKF are better if these steps are integrated into the measurement equations listed in chapter 5.2.2, instead of performing the calibration beforehand on the raw data. Nevertheless, the results of the wind tunnel runs are important to judge the EKF performance and to correct very noisy data, where the EKF fails to estimate the calibration factors correctly.

Rotation Rates
As shown in Figure 6.17, the probe is mounted on the right wing less than half a chord length in front of the leading edge. Pitching and rolling motion will influence the measured angle of attack as the vane is displaced by those motions. For example, a positive roll rate (right wing going down) will increase the measured angle of attack. An approximation for the required correction is given in reference \cite{22} as

$$\alpha_{\text{true}} = \alpha_{\text{meas}} + \frac{qx_a}{V_{\text{air}}} - \frac{py_a}{V_{\text{air}}}$$

(6.16)

where $x_a$ and $y_a$ are the distances of the vane from the aircraft CG and $V_{\text{air}}$ the true airspeed. This simplified correction is valid only for small angles and slow rates. Both assumptions are true for this project and airframe as will be shown using real test data later on. The distance in $x$ of the vane in front of the CG is only 0.12m, so the influence of the pitch rate is expected to be relatively small. The $y$ distance is larger with 0.475m and
even though the roll rates are slower than the pitch rate this correction will be significant during rapid rolling motion during aileron doublets.

It is important that the corrections from Eq. (6.16) are applied first to remove all influences of the aircraft motion from the data. Otherwise the correction due to the wing upwash will also be applied to the component of the measurement due to the motion of the aircraft, which would lead to the wrong answer.

Wing Upwash

The angle of attack vane is affected by the wing upwash caused by the flow field around the wing as shown in Fig. 6.18. This effect is dependent on the lift coefficient and therefore on the angle of attack itself. At subsonic flight speeds the perturbations of the streamlines extend far in front of the wing. Typically, an airdata probe would have to be mounted two chord lengths in front of the leading edge to measure the free stream conditions. This is not possible for this probe as discussed above. The figure shows the location of the angle of attack vane inside the flow field at (a) the zero lift angle of attack and (b) at 5 degree angle of attack.

At the zero lift angle of attack, the slow around the wing should be symmetrical and thus the effect on the vane should be zero. This close to the leading edge this is not true, because the blockage of the wing will split up the flow and thus cause local displacements of the streamlines. This will show as a small offset of the correction curve at the zero angle of attack. At 5 degrees angle of attack the upwash effect is quite severe. The measured angle of attack is about twice the true angle. It is therefore very important to carefully develop this correction as errors can be large. The small rectangle around
For the free air case with an elliptical wing loading the correction can be written as

$$\alpha_{meas} = \alpha_{true} + \alpha_i$$  \hspace{1cm} (6.17)

$$\alpha_{meas} = \alpha_{true} + \frac{C_L}{\pi AR} + \alpha_{i,0}$$  \hspace{1cm} (6.18)

where $\alpha_{i,0}$ is the component at the zero lift angle of attack as discussed above. Then using $C_L = C_{L,0} + C_{L,\alpha} \times \alpha_{true}$ and simplifying gives

$$\alpha_{meas} = \left(1 + \frac{C_{L,\alpha}}{\pi AR}\right) \alpha_{true} + \frac{C_{L,0}}{\pi AR} + \alpha_{i,0}$$  \hspace{1cm} (6.19)

This result is the equation of a line. The given rectangular wing with its non-elliptical loading is also expected to be fairly linear, at least at small angles of attack, since the differences in wing loading from the elliptical case are relatively small.

In the wind tunnel the correction is also dependent on the upwash due to the wind tunnel walls

$$\alpha_{meas} = \alpha_{true} + \alpha_i + \alpha_{walls}$$  \hspace{1cm} (6.20)

where the wall correction $\alpha_{walls}$ also depends on the wing $C_L$ and thus has the same form as the induced angle of attack. The upwash due to the walls can be significant (see
Figure 6.19: PanAir solution for the wing upwash. Red: vane at the true AoA (5 deg), Black: vane at measured AoA (aligned with the flow)

appendix C.6), so two corrections are required: one for the free air case and one for the wind tunnel case.

The wind tunnel case correction can be obtained by experiment, comparing the measured angle of attack against a reference data source, such as the static balance angle read-out or the gimbal pitch angle measurement. Correcting for the wall interference to get the free air correction is more difficult as it requires the knowledge of the local changes in upwash due to the walls. Given the good results of the simulation of the wind tunnel environment (see appendix C), this simulation should give an accurate estimate of the free air correction once the method is benchmarked against the wind tunnel experimental data. The virtual wind tunnel also allows to take into account any geometry differences between the two airframes. A solution from the virtual wind tunnel, using the PanAir solver to calculate a grid of off-body points around the alpha vane, is shown in Figure 6.19. It represents a detailed view of the small rectangle in Figure 6.18(b).

The PanAir solution for off-body points gives the velocity components at each point. This information can then be used to generate stream lines as shown in the figure. The flow curvature is not constant along the vane length this close to the wing, so it is important to calculate the deflection at precisely the right point on the vane. Here the quarter chord of the vane (at about x=0.05m) was chosen. Due to the changes in flow curvature it is then necessary to iterate to obtain the correct vane deflection that is aligned with the flow. Doing this for a number of angles of attack with and without the walls present results in the correction curves plotted in Fig. 6.20. The free air correction is smaller than the wind tunnel correction as expected. The corrections are also precisely linear despite the non-elliptical wing loading. The figure also shows that at the zero lift angle of attack the correction is not zero as indicated by the black dot.

To obtain the correction for the angle of attack measurements experimentally, the
fully instrumented wind tunnel airframe was mounted on the static wind tunnel balance (Figure A.3). The balance was then driven in steps with increasing angle of attack while the data from the probe was recorded by the instrumentation as shown in Figure 6.21 (left). To locate each step in the data, the index channel feature of the UAVmainframe was used to put a marker at the beginning and the end of each step as shown in the figure. This allows to automatically locate the required data during processing. Using the

Figure 6.20: PanAir solutions for the wing upwash corrections. The dot represents the zero lift angle of attack

Figure 6.21: Airdata probe: Angle of attack vane experiment and PanAir numerical results. left: vane raw data, right: resulting corrections from the experiment and the calculations
recordings from the wind tunnel balance for the true angle of attack a plot of measured AoA vs true AoA can be made (Figure 6.21 right). The resulting correction equation from the wind tunnel experiment is (in degrees)

\[ \alpha_{\text{true}} = 0.62 \alpha_{\text{meas}} - 1.27 \]  

(6.21)

From the PanAir solution the correction is (in degrees)

\[ \alpha_{\text{true}} = 0.61 \alpha_{\text{meas}} - 1.24 \]  

(6.22)

The figure shows a very good match between the numerical solution and the experiment. Slope and offset of the data is in close agreement. This shows that the virtual wind tunnel can predict the corrections for the angle of attack vane very well and therefore it can be assumed that the solution for the free air case without the walls present is of similar precision. Hence, the EKF estimates for the angle of attack scale factor should match these predictions, if this state is sufficiently observable. This is true for clean flight data as shown in chapter 15. The free air correction is calculated in PanAir from the same aircraft model without the walls, as described in appendix C:

\[ \alpha_{\text{true}} = 0.66 \alpha_{\text{meas}} - 0.94 \]  

(6.23)

The change in slope is about 5% and the difference in offset is 25% due to the influence of the wind tunnel walls. This results in a measured difference of 1.2 degrees at a true angle of attack of 5 degrees between the cases, which illustrates the importance of distinguishing the two cases. The next sections will benchmark these corrections against some dynamic wind tunnel test data to demonstrate the magnitude of corrections required.

**Verification of the Results**

The buildup of the corrections for the angle of attack are demonstrated in Figures 6.22 for a longitudinal manoeuvre and 6.23 for a lateral manoeuvre. The data is from the dynamic wind tunnel tests presented in part V. Longitudinal and lateral motion is cleanly separated in those tests except for the noise due to the turbulence in the test section which excites all axes in each test. During the dynamic wind tunnel tests the attitude sensors of the gimbal provide a precise reference for the aerodynamic angles as the gimbal pitch angle equals the angle of attack and the gimbal yaw angle equals the negative of the sideslip angle. These tests typically use perturbation data only, so all biases have been removed during the data processing.

The response to an elevator input in Figure 6.22(a) shows a large error between the raw angle of attack and pitch angle measurements during the input as expected. The smaller oscillations in the first 0.5 seconds are due to turbulence. The errors are plotted in 6.22(d) for the raw data and the final corrected data. The corrections due to the different causes (pitch rate, roll rate and upwash) are plotted in 6.22(b). The largest correction is due to the upwash, followed by the pitch rate correction. There is only a small amount of roll rate correction due to the wing rock caused by the turbulence. This component is not significant for the longitudinal case.
Applying the corrections to the raw data in the correct order, as mentioned above, gives the final corrected timeseries for alpha as shown in 6.22(c) vs. the true pitch angle. Very good agreement has been achieved, with the residuals reduced to oscillations of ±0.5 deg caused by the turbulence that excites the light vane but not the entire airframe. There are small remaining errors at 1.1 sec and 1.9 sec which are also caused by the turbulence interfering with the vane during the manoeuvre. These errors cannot be corrected, but their small magnitude is not expected to make a significant difference during the system identification.

The lateral case in Fig. 6.23 shows a response to an aileron input preceded by some oscillation in roll caused by a rudder input and the subsequent dutch roll motion. The rudder input is not shown for clarity. Part (a) plots the raw data against the true pitch angle. There is quite a bit of noise due to turbulence on the sensor, made more visible by the smaller scale of the plot and the longer time frame needed for the slower lateral motion as compared to the fast longitudinal motion in Fig. 6.22. The only significant correction is the one due to the roll rate as shown in (b). The upwash correction acts only on the remaining noise after the roll rate component was removed. This shows the significance of the order in which the corrections are applied. No visible pitching motion can be observed.

The corrected alpha time series and the corresponding residuals in (c) and (d) contain only noise. This can be seen at the beginning and after 4 sec where there is no difference
between the raw and corrected residuals as the noise cannot be corrected for since there is no corresponding airframe motion. Only the vanes flutter in the unsteady flow due to their low inertia.

Summarising, the corrections for the alpha vane developed in this section yield very good agreement with the reference pitch angle in both longitudinal and lateral cases and therefore give the correct angle of attack when applied to the raw sensor data. The corrections used for the flight data has to be adjusted for the wall interference in the upwash correction but otherwise will be valid without further modification. This section shows that correcting the alpha vane is critically important to obtain the correct results. Neglecting this step will lead to significant errors later in the processing.

### 6.4.5 Angle of Sideslip

Similarly to the angle of attack, a correction for the sideslip vane due to airframe rotations can be derived as [22]

\[
\beta_{true} = \beta_{meas} - \frac{r x_{\beta}}{V_{air}} + \frac{p z_{\beta}}{V_{air}} \tag{6.24}
\]

where \( x_{\beta} \) and \( z_{\beta} \) are the distances of the vane from the aircraft CG in the respective axes and \( V_{air} \) is the true airspeed.

The sideslip vane does not require any corrections in the longitudinal case due to upwash. Pure pitching motion will be perpendicular to the vane and thus have no effect.
The only issue might arise from the fact that the vane in this case is mounted on top of the bulky sensor housing as opposed to the preferred mount below. For this probe, a vane below the case would be very close to the ground on the flight airframe and therefore prone to damage. On the other hand, a vane on top of the probe could be affected by the wake of the housing at higher angles of attack. For this project, the alpha range is small and the advantages of the bigger ground clearance prevail.

To verify the performance of the beta vane in the longitudinal case, raw sideslip data has been plotted against the gimbal yaw angle in Fig. 6.24, together with the residuals. The dataset is the longitudinal input used in Fig. 6.22. As can be seen in the figure, there is no signal in the data except turbulence noise. Thus the assumption of zero corrections in the longitudinal case appears correct for the given alpha range.

The response to a rudder input is shown in Figure 6.25. The raw sideslip data is plotted against the true yaw angle in (a). Residuals are shown in (d) as before. There are only small differences, mainly due to noise. The correction due to the yaw rate is shown in (b). No other correction is significant, as the z-distance of the probe from the CG close to zero. The yaw rate correction is small because the the distance to the CG is small and the motion (e.g., the yaw rate) is slow. The corrected beta time series vs. the true yaw angle is shown in (c) and the resulting residuals in (d). During the dutch roll yawing motion the residuals are slightly improved, but are still dominated by the noise. The noise causes oscillations of the magnitude ±1 degree, which is much more than the alpha vane experienced. Overall, the beta corrections are small for this probe and installation. They could probably be omitted for the wind tunnel case as noise dominates the errors. For the flight case the yaw corrections can be used without modifications.
6.5 Control Surface Feedback

The only group of sensors required for the system ID and not treated by the EKF are the control surface positions. Calibration of these sensors is straightforward and can be done with several methods. The easiest method to calibrate the control surfaces is to attach a two or three axis accelerometer to the control surface and to record its data while moving the surface with the remote control transmitter. The angle of the accelerometer can then be calculated from the components of the gravity vector in the different accelerometer channels by simple trigonometry. A least squares method can then be used to determine the coefficients of the calibration polynomial.

An example is shown in Figure 6.26 for the elevator. The ailerons can be treated in the same way, while for the rudder it is required to place the aircraft on its side to obtain useful accelerometer data. The two graphs at the top of the Figure show the raw data as measured by the accelerometer and the control surface feedback sensor for two different inputs. It appears that the calibration polynomial is simply a linear scale factor between the two measurements. This is confirmed by the least squares solution shown at the bottom of the Figure, where both sensors match quite well except for some transients on the sharp elevator doublet. The determined scale factors are 0.767 and 0.777, respectively.

The bias value, that is the travel midpoint, was set by taking the feedback sensor reading with the surface aligned with the wing aerofoil. These two values are then used...
in the sensor model
\[
\delta_e^{(true)} = scale \times (\delta_e^{(meas.)} - bias)
\] (6.25)
which is automatically applied to the raw control surface position measurements by the UAV mainframe in flight. This simple calibration is not valid at the end of the deflection range due to non-linearities in the linkages at large angles. This is not a problem for this project because all test points are small angle perturbation manoeuvres around trimmed, level flight with expected maximum control deflections of \( \pm 5 \) deg or less.

This concludes the pre-flight calibrations. The remaining errors are treated by the EKF processing step after the flight. It will be shown in Part VII that the combination of the calibrations developed in this chapter, together with the EKF result in highly accurate and repeatable flight data that allows detailed analysis of the flight dynamics of this small aircraft.
### 7 Static Wind Tunnel Tests

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Overview and Test Requirements</td>
</tr>
<tr>
<td>7.2</td>
<td>Measurement Uncertainties</td>
</tr>
<tr>
<td>7.3</td>
<td>Initial Polar</td>
</tr>
<tr>
<td>7.4</td>
<td>Powered Test</td>
</tr>
<tr>
<td>7.5</td>
<td>Results with Wall Interference</td>
</tr>
<tr>
<td>7.6</td>
<td>Wall Interference Corrected Results</td>
</tr>
<tr>
<td>7.7</td>
<td>Static Wind Tunnel Test Summary</td>
</tr>
</tbody>
</table>

Reference Data: Static Derivatives
7. Static Wind Tunnel Tests

7.1 Overview and Test Requirements

The most reliable source of aerodynamic data is obtained from static wind tunnel testing, which has been used since the earliest days of aircraft development. These tests are performed in a controlled and known environment and therefore should produce high quality reference data to benchmark other data sources. For this to happen the wind tunnel must be equipped with reliable instrumentation and its properties known by careful calibration.

The wind tunnel used for this project originally did not have a suitable balance to do tests in all six degrees of freedom. Therefore, a new balance was designed and constructed. It is described in appendix A, together with a thorough evaluation of the highly turbulent test section flow quality and the accuracy of the newly installed systems. To compare the wind tunnel test data to the flight data, it has to be corrected for the wall interference effects. Corrections were developed using the methods in the classic reference text [88] and by simulating the wind tunnel environment with the PanAir solver. This work is presented in appendix B and C, respectively. The resulting wall corrections are then applied to the test data obtained in this chapter to yield the required reference data for the flight tests.

The static wind tunnel tests for this project are used to obtain reference values for the static stability and control derivatives of the test aircraft. These depend only on steady state variations of the forces and moments with angle of attack and sideslip and are not time dependent. The derivatives are typically written in non-dimensional coefficient form as

\[ C_{X\alpha} = \frac{\delta C_X}{\delta \alpha} \]  

(7.1)
and

\[ C_{X\beta} = \frac{\delta C_X}{\delta \beta} \] (7.2)

where \( X \) can be any of the three forces (lift, drag and sideforce) and moments (roll, pitch and yaw) measured at the reference point (typically the aircraft CG).

The aerodynamic forces and moments are non-dimensionalised into coefficient form by

\[ C_F = \frac{F}{qS} \] (7.3)

or

\[ C_M = \frac{M}{qSl} \] (7.4)

where \( q \) is the dynamic pressure, \( S \) the reference area and \( l \) a reference length, chosen according to the derivative calculated.

The derivatives are typically linear for a conventional aircraft shape at small inflow angles, where no flow separation occurs. They can be obtained by fitting a line of the form \( y = ax + b \) to a sequence of wind tunnel test results, where \( x \) is either the angle of attack or the sideslip and \( a \) is the value of the derivative as defined above. Therefore care must be taken to measure this gradient \( a \) with best possible accuracy, whereas the constant offset \( b \) is typically not important for the flying properties of an aircraft, unless it is large. This allows for some simplifications for the design of the static wind tunnel tests for this project:

- It is important to measure the change in the aerodynamic angles between test points accurately. Small offset errors, caused by the model installation, can be ignored as they would introduce only a constant offset \( b \). This relaxes the requirements for the model installation onto the balance, which does not have to be calibrated for accurate zero angle alignment.

- The dynamic pressure \( q \) must be measured with constant bias and scale factors at each test point but the exact value is not important because each data point is normalised by the current dynamic pressure. Constant errors on the measurement of \( q \) will therefore cancel out during the line fit.

- The dynamic pressure can vary between test points, because all measurements are normalised by \( q \), as long as the speed variations are small enough such that the flow properties, such as the Reynolds number, do not significantly change. This relaxes the requirements for exact control of the tunnel airspeed, which is not an easy task, given the long time constant of the system and the ancient equipment of this wind tunnel.

- The reference values can in theory be arbitrarily chosen, but using values conforming to the standard conventions enables comparison of the data with other publications.

- The forces and moments need to be measured as accurately as possible at each test point, but a constant bias may be acceptable because it won’t change the slope of the line fit.

- All reference quantities must be kept constant between tests.
The main focus of this project were the stability and control derivatives. Drag measurements are given for reference only and no effort was made to verify or correct those. The high turbulence levels in the test section will not be comparable to calm weather flight data anyway.

The loadcell reference point is at \(X = 0.227\, m\), \(Y = 0\, m\) and \(Z = 0.207\, m\). Moments are given at the load cell reference point and transferred to the gimbal/flight CG at \(X = 0.240\, m\), \(Y = 0\, m\) and \(Z = 0.225\, m\).

Firstly, the measurement uncertainties will be discussed. Then the general longitudinal aerodynamic properties of the test aircraft are introduced with a standard lift-drag- and pitching moment polar. Briefly, the matter of power effects on the stability derivatives are then discussed. Finally, the results of the static wind tunnel tests will be presented together with the data corrected for the wall interference effects.

### 7.2 Measurement Uncertainties

To begin this section, a selection of sample data sets are presented to illustrate the data quality achieved with the new wind tunnel balance. A calibrated test article to benchmark the accuracy of the results was not available, hence this section concentrates on repeatability and data noise.

Figure 7.1 shows the results for three runs for the lift- and drag force coefficients. The repeatability of the lift is excellent, with virtually no difference between the test points and identical slopes of the fitted line. The data listed in Table 7.1 confirms this result. As mentioned before, each test point is the mean of 2000 individual samples. This appears to be a sufficiently large number to even out the spread between the individual measurements.

The drag shows somewhat more spread between the runs, especially near the minimum. But the quadratic fit to obtain the Oswald span efficiency \(e\) is again very repeatable, as also confirmed by Table 7.1.

The pitching moment coefficients plotted in Figure 7.2 are perfectly repeatable, similar to the lift data. Table 7.2 confirms this with the results for the slopes of the curve fits for each run. The rolling moment is most affected by the turbulent flow inside the test section. This shows in the test data in Figure 7.2, where, despite using 5000 samples, there is more spread between the runs and the data points have larger residuals to the curve fit. The the rolling moment coefficient slope is nevertheless repeatable to a high degree of accuracy as shown in Table 7.2.

<table>
<thead>
<tr>
<th>Run</th>
<th>(C_{L_{\alpha}})</th>
<th>(e)</th>
<th>(C_{Y_{0}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.170</td>
<td>0.761</td>
<td>-0.51</td>
</tr>
<tr>
<td>2</td>
<td>5.153</td>
<td>0.766</td>
<td>-0.51</td>
</tr>
<tr>
<td>3</td>
<td>5.178</td>
<td>0.771</td>
<td>-0.51</td>
</tr>
</tbody>
</table>
Chapter 7. Static Wind Tunnel Tests

Figure 7.1: Lift and drag force coefficient measurements at 20m/s for three separate runs

Figure 7.2: Pitching and rolling moment coefficient measurements at 20m/s for three separate runs

Despite the highly repeatable sample results presented, quantifying the measurement uncertainties for the static wind tunnel tests is not an easy task because no test article with known properties was available, and therefore the absolute correctness of the measured forces and moments is unknown (some basic tests with lead weights were done in appendix A, but these were only for small loads). Using the standard deviations of each measurement is also not a good representation because of the noisy environment and the heavy averaging used. For example the standard deviation of the lift force is $\pm 3 - 4N$ independent of the measured force. This is due to the noise on the small signal strain.
7.3 Initial Polar

This section contains an initial longitudinal polar run to assess the general characteristics of the airframe. Using this data, obtained early during the project, the test parameters for the stability and control runs were determined. The data was also used to determine the static margin for the flight CG location.

Table 7.2: Sample results for moment coefficients

<table>
<thead>
<tr>
<th>Run</th>
<th>$C_{m_{\alpha}}$</th>
<th>$C_{l_{\beta}}$</th>
<th>$C_{n_{\beta}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.188</td>
<td>-0.0667</td>
<td>0.0912</td>
</tr>
<tr>
<td>2</td>
<td>-1.190</td>
<td>-0.0669</td>
<td>0.0913</td>
</tr>
<tr>
<td>3</td>
<td>-1.190</td>
<td>-0.0660</td>
<td>0.0911</td>
</tr>
</tbody>
</table>

Gauges inside the load cell. Yet, the force measurement are repeatable with accuracies of $< 1\%$, if sufficient numbers of samples are averaged. The line fit for the final result, the $C_{L_{\alpha}}$ derivative, is a further averaging process.

Therefore, the repeatability of the derivative measurements and the level of agreement with the virtual wind tunnel results gives a better estimate for the uncertainties of the static wind tunnel tests. Table 7.3 shows the results of tests done to quantify the 95% confidence interval ($2\sigma$) for repeated tests of the aerodynamic derivatives. All quantities are repeatable to an accuracy of less than one percent over multiple runs, as mentioned in appendix A.

The table also contains the differences of the wind tunnel tests to the PanAir results from the virtual wind tunnel simulations. As discussed in appendix C, the longitudinal data matches to an accuracy of less than two percent. The lateral data agrees less well, but is was theorized in appendix C that this is mostly due to un-modelled flow separation over the fuselage and geometry modelling errors of the fin shape. Given the excellent repeatability of the wind tunnel data and the agreement in the longitudinal axis, it would be overly conservative to use the lateral differences to the PanAir data as the uncertainties for the wind tunnel measurements. Further research is required to quantify these measurement errors, but for this project it has been decided to use $\pm 2\%$ for the longitudinal axis and $\pm 5\%$ for the lateral axis. It is felt that this is a conservative assumption that is justified by the preceding discussion.

Table 7.3: Wind tunnel uncertainties

<table>
<thead>
<tr>
<th></th>
<th>$C_{L_{\alpha}}$</th>
<th>$C_{m_{\alpha}}$</th>
<th>$C_{y_{\beta}}$</th>
<th>$C_{l_{\beta}}$</th>
<th>$C_{n_{\beta}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeatability</td>
<td>$&lt; 1%$</td>
<td>$&lt; 1%$</td>
<td>$&lt; 1%$</td>
<td>$&lt; 1%$</td>
<td>$&lt; 1%$</td>
</tr>
<tr>
<td>spread ($2\sigma$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Panair $\Delta$</td>
<td>$&lt; 1%$</td>
<td>1.6%</td>
<td>15%</td>
<td>7.2%</td>
<td>5%</td>
</tr>
</tbody>
</table>
Chapter 7. Static Wind Tunnel Tests

Table 7.4: Wind tunnel uncertainties final

<table>
<thead>
<tr>
<th>Assumed uncertainties</th>
<th>$C_{L_{\alpha}}$</th>
<th>$C_{m_{\alpha}}$</th>
<th>$C_{y_{\alpha}}$</th>
<th>$C_{l_{\beta}}$</th>
<th>$C_{n_{\beta}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2%</td>
<td>2%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
</tr>
</tbody>
</table>

Figure 7.3 indicates that the linear region of this aircraft is between -5 deg and 5 deg angle of attack (AoA). Due to the large wing incidence the angle of attack for the cruise $C_L \approx 0.4$ is at zero degrees, keeping the fuselage at level. Above 5 degrees separation starts, with lift and pithing moment starting to deviate from the linear fit. But this separation is benign until 10 degrees AoA, where a sharp change in all curves indicates a major change in the flow field. Visual observations of vibrations and the kink in the moment curve suggest a stall of the tailplane at this point. Full stall of the wing occurs at 13 degree AoA.

The drag polar shows a fairly high $C_{D_0} = 0.043$, which is expected with a bulky airframe like this one at low Reynolds numbers. The Oswald efficiency $e = 0.77$ is in line with the theory for an aspect ratio of 5.5. The resulting low glide ratio $L/D_{max} = 13.6$ is significant during emergencies like engine failures and during glide tests to remove engine noise, because the altitude loss during glide is large and therefore not much distance can be covered back to the runway. This was a limitation that had to be considered during the flight tests and has led to at least one landing accident during the project.

The pitching moment curve has a negative slope, indicating stable pitch trim at the load cell CG. The static margin is about 23%, which indicates that the flight static margin will be between 15-20%, because the flight CG is a short distance behind the load cell mount.

![Graphs showing CL, Cm, CD, and L/D](image_url)

Figure 7.3: Initial polar about the load cell reference point (section 7.1) at 20m/s
Power effects can have a large effect on the characteristics of an aeroplane in the wind tunnel [88], especially if scaled power loadings are used to simulate a full scale aircraft. To investigate if the power effects for the airframe used in this project are significant when comparing the data to the flight tests, the flight fuselage was mounted onto the wind tunnel wings and tests were run with varying power settings. The results for the lift coefficient and the pitching moment is shown in Fig. 7.4. The lift curve slope $C_{\text{L}}$ increases with rising power due to the wings being blown by the propeller as expected. At full power it is 10% higher than without power. On the other hand, even at full power, there is no significant difference in $C_{\text{m}}$ to the un-powered run.

All test points for this project were flown at cruise power. At that setting the power effects are small and thus have not been considered in this project. This can probably be explained by the fact that the power loading of the small scale test airframe is less than half than the airframe used in the reference book. There may be some effects at very low airspeeds but those are not within the flight regime considered in this project. For slow speed test data, a more detailed study in the wind tunnel would be required.

### 7.5 Results with Wall Interference

This section contains the uncorrected results from the longitudinal and lateral tests as obtained directly from the balance recordings. The dynamic tests are run under similar conditions in the same tunnel, so they can be directly compared to these uncorrected static test results. No wall corrections are required in this case.
7.5.1 Stability Derivatives

Results are given for the stability and control derivatives in the linear region between $-5\,\text{deg} < \alpha < 5\,\text{deg}$. Each run was repeated three times at $V = 20\,\text{m/s}$ with 2000 samples at each data point over 2 seconds. The coefficient slopes were determined by line fits to the data. The repeatability of the results is very good and hence it was decided that three repeats are sufficient. The applied moments are given at the load cell CG and are transferred to the flight CG for use during the dynamic tests (See section 7.1 for CG locations). Tables 7.5 to 7.9 contain the results for the lift, pitching moment, side force, rolling moment and yawing moment coefficient, respectively. The moment results indicate that the test aircraft is stable in all three axis about the load cell CG, as well as the flight CG.

Table 7.5: Lift coefficient results for three runs at $V = 20\,\text{m/s}$

<table>
<thead>
<tr>
<th>Run</th>
<th>$C_{L,\alpha}$</th>
<th>$C_{L(\alpha=0)}$</th>
<th>$\alpha_{(C_{L}=0)}$ [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.170</td>
<td>0.372</td>
<td>-4.125</td>
</tr>
<tr>
<td>2</td>
<td>5.153</td>
<td>0.369</td>
<td>-4.102</td>
</tr>
<tr>
<td>3</td>
<td>5.178</td>
<td>0.369</td>
<td>-4.09</td>
</tr>
<tr>
<td>Mean</td>
<td><strong>5.167</strong></td>
<td><strong>0.370</strong></td>
<td><strong>-4.106</strong></td>
</tr>
</tbody>
</table>

Table 7.6: Pitching moment results for three runs at $V = 20\,\text{m/s}$ with moment transfer to the gimbal/flight CG

<table>
<thead>
<tr>
<th>Run</th>
<th>$C_{M,\alpha}$ [at Loadcell]</th>
<th>$C_{M,\alpha}$ [at Gimbal]</th>
<th>$X_{NP}$ [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.190</td>
<td>-0.947</td>
<td>0.291</td>
</tr>
<tr>
<td>2</td>
<td>-1.190</td>
<td>-0.949</td>
<td>0.291</td>
</tr>
<tr>
<td>3</td>
<td>-1.188</td>
<td>-0.948</td>
<td>0.291</td>
</tr>
<tr>
<td>Mean</td>
<td><strong>-1.189</strong></td>
<td><strong>-0.948</strong></td>
<td><strong>0.291</strong></td>
</tr>
</tbody>
</table>

Table 7.7: Side force coefficient results for three runs at $V = 20\,\text{m/s}$

<table>
<thead>
<tr>
<th>Run</th>
<th>$C_{k,s}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.51</td>
</tr>
<tr>
<td>2</td>
<td>-0.51</td>
</tr>
<tr>
<td>3</td>
<td>-0.51</td>
</tr>
<tr>
<td>Mean</td>
<td><strong>-0.51</strong></td>
</tr>
</tbody>
</table>
### Table 7.8: Rolling moment coefficient results for three runs at $V = 20\text{m/s}$

<table>
<thead>
<tr>
<th>Run</th>
<th>$C_{l_5}$ [at Loadcell]</th>
<th>$C_{l_5}$ [at Gimbal]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.067</td>
<td>-0.061</td>
</tr>
<tr>
<td>2</td>
<td>-0.067</td>
<td>-0.061</td>
</tr>
<tr>
<td>3</td>
<td>-0.066</td>
<td>-0.06</td>
</tr>
<tr>
<td>Mean</td>
<td><strong>-0.067</strong></td>
<td><strong>-0.061</strong></td>
</tr>
</tbody>
</table>

### Table 7.9: Yawing moment coefficient results for three runs at $V = 20\text{m/s}$

<table>
<thead>
<tr>
<th>Run</th>
<th>$C_{n_5}$ [at Loadcell]</th>
<th>$C_{n_5}$ [at Gimbal]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.091</td>
<td>0.087</td>
</tr>
<tr>
<td>2</td>
<td>0.091</td>
<td>0.087</td>
</tr>
<tr>
<td>3</td>
<td>0.091</td>
<td>0.087</td>
</tr>
<tr>
<td>Mean</td>
<td><strong>0.091</strong></td>
<td><strong>0.087</strong></td>
</tr>
</tbody>
</table>

#### 7.5.2 Elevator derivatives

The elevator derivatives were determined by deflecting the elevator on the otherwise static airframe and recording the changes in lift and pitching moment. The airframe was kept at zero angle of attack on the balance.

The elevator angles were read from the calibrated feedback sensor while the surface was deflected by the servo motors using the trim switches on the remote control. That way a repeatable deflection could be set by simply counting the 'clicks' on the RC transmitter. The forces and moments were recorded on the static balance system and later correlated with the recorded elevator deflections that were read of the UAVmainframe data.

This method cannot account for flexing in the surface, as the calibration of the angle encoder could only be done in an unloaded condition. Excessive flex should show up in the test data as a deviation from the expected straight line for these derivatives.

Figure 7.5 shows the test data and the curve fit for the two elevator derivatives $C_{L_{5e}}$ and $C_{m_{5e}}$. Both lines provide a good fit to the data, which confirms that the angle sensor calibrations work fine. The results for the pitching moment have to be corrected to be valid at the gimbal/flight CG position. Wall corrections were not considered for the elevator, since those are expected to be small. The results for the longitudinal elevator derivatives are:

<table>
<thead>
<tr>
<th>Derivative</th>
<th>Value [at Loadcell]</th>
<th>Value [at Gimbal]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_{5e}}$</td>
<td>-0.511</td>
<td><strong>-0.511</strong></td>
</tr>
<tr>
<td>$C_{m_{5e}}$</td>
<td>-1.169</td>
<td><strong>-1.143</strong></td>
</tr>
</tbody>
</table>
7.5.3 Aileron derivatives

The aileron derivatives were tested similarly to the elevator. The data for $C_{l_{\delta a}}$ is not as clean as the elevator data, which might be caused by the larger influence of the test section turbulence on the roll axis. The $C_{n_{\delta a}}$ cross-derivative is a very small number and not surprisingly the data is noisy. Both values were transferred to the gimbal/flight CG for further use, even though the differences are very small, because only a vertical change in the CG location applies. The results for the lateral aileron derivatives are:

<table>
<thead>
<tr>
<th>Derivative</th>
<th>Value [at Loadcell]</th>
<th>Value [at Gimbal]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l_{\delta a}}$</td>
<td>-0.179</td>
<td>-0.178</td>
</tr>
<tr>
<td>$C_{n_{\delta a}}$</td>
<td>0.009</td>
<td>0.008</td>
</tr>
</tbody>
</table>

Figure 7.5: Elevator derivatives test data and curve fit

Figure 7.6: Aileron derivatives test data and curve fit
7.5 Results with Wall Interference

7.5.4 Rudder derivatives

There are three derivatives due to a rudder deflection, side force, rolling moment and yawing moment. Figure 7.7 shows the data for those. Side force and yawing moment are the large derivatives in the main direction of the rudder effectiveness, showing linear trends as expected. The rolling moment derivative is very small and therefore noisy. All data has been transferred to the flight CG as before.

![Graphs showing rudder derivatives test data and curve fit](image)

Figure 7.7: Rudder derivatives test data and curve fit

The results for the lateral rudder derivatives are:

<table>
<thead>
<tr>
<th>Derivative</th>
<th>Value [at Loadcell]</th>
<th>Value [at Gimbal]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{y_{\delta r}}$</td>
<td>0.169</td>
<td>0.169</td>
</tr>
<tr>
<td>$C_{l_{\delta r}}$</td>
<td>0.009</td>
<td>0.007</td>
</tr>
<tr>
<td>$C_{n_{\delta r}}$</td>
<td>-0.065</td>
<td>-0.063</td>
</tr>
</tbody>
</table>

7.5.5 Static Test Result Summary

This section collects all uncorrected results from the static wind tunnel experiment, transferred to the gimbal/flight CG. These can be directly compared to the results of the dynamic tests discussed in part V. These values are also used for the preliminary definition of the aircraft in the flight simulator code, where they take precedence over any data obtained from the dynamic tests due to their higher accuracy. The table also gives the 95% or two-sigma confidence interval, using the uncertainties of 2% for the longitudinal data and 5% for the lateral data as discussed before. The lateral cross derivatives of the rudder and the aileron will be set to zero because they are too small to be significant and thus very difficult to identify from the dynamic data.
Table 7.10: Summary of uncorrected static wind tunnel derivatives about the gimbal/flight CG at $V = 20\text{m/s}$

<table>
<thead>
<tr>
<th>Deriv.</th>
<th>Value</th>
<th>Error</th>
<th>95% Confidence</th>
<th>Deriv.</th>
<th>Value</th>
<th>Error</th>
<th>95% Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_{\alpha}}$</td>
<td>5.167</td>
<td>2%</td>
<td>[5.06...5.27]</td>
<td>$C_{y_{\alpha}}$</td>
<td>-0.510</td>
<td>5%</td>
<td>[-0.536...-0.484]</td>
</tr>
<tr>
<td>$C_{m_{\alpha}}$</td>
<td>-0.948</td>
<td>2%</td>
<td>[-0.967...-0.929]</td>
<td>$C_l_{\beta}$</td>
<td>-0.061</td>
<td>5%</td>
<td>[-0.064...-0.058]</td>
</tr>
<tr>
<td>$C_{m_{\delta_e}}$</td>
<td>-0.511</td>
<td>2%</td>
<td>[-0.521...-0.501]</td>
<td>$C_{n_{\delta_e}}$</td>
<td>0.087</td>
<td>5%</td>
<td>[0.083...0.091]</td>
</tr>
<tr>
<td>$C_{m_{\delta_e}}$</td>
<td>-1.143</td>
<td>2%</td>
<td>[-1.166...-1.120]</td>
<td>$C_{l_{\delta_e}}$</td>
<td>-0.178</td>
<td>5%</td>
<td>[-0.187...-0.169]</td>
</tr>
<tr>
<td>$C_{m_{\delta_r}}$</td>
<td>0.169</td>
<td>5%</td>
<td>[0.161...0.177]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$C_{n_{\delta_r}}$</td>
<td>0.007</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$C_{n_{\delta_r}}$</td>
<td>-0.063</td>
<td>5%</td>
<td>[-0.066...-0.060]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7.6 Wall Interference Corrected Results

This section contains the static test results with the wall corrections applied to the longitudinal data. Figure 7.8 shows the simulated test aircraft in the test section of the wind tunnel. The full development of the simulation and its benchmarking is presented in appendix C. As shown in Section C.6, the corrections obtained from the virtual wind tunnel are a better match to the data than the corrections based on the method of images. Therefore the PanAir results will be used from now on and have been repeated in Table 7.11 for reference. The corrections have been applied to the longitudinal data in Table 7.12. The lateral data requires no correction and the correction for the elevator derivatives is assumed small, hence the data in Table 7.10 is valid for those parameters in flight.

Table 7.11: Virtual wind tunnel wall interference results.

<table>
<thead>
<tr>
<th>PanAir Wall Correction</th>
<th>$C_{L_{\alpha}}$</th>
<th>$C_{m_{\alpha}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-10%</td>
<td>-18%</td>
<td></td>
</tr>
</tbody>
</table>

Table 7.12: Corrected static wind tunnel derivatives about the gimbal/flight CG at $V = 20\text{m/s}$

<table>
<thead>
<tr>
<th>Deriv.</th>
<th>Value</th>
<th>Error</th>
<th>95% Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_{\alpha}}$</td>
<td>4.540</td>
<td>2%</td>
<td>[4.449...4.631]</td>
</tr>
<tr>
<td>$C_{m_{\alpha}}$</td>
<td>-0.768</td>
<td>2%</td>
<td>[-0.783...-0.753]</td>
</tr>
</tbody>
</table>
7.7 Static Wind Tunnel Test Summary

This concludes the treatment of the static derivatives of the test aircraft. After designing and constructing a entirely new wind tunnel balance, a thorough validation of the wind tunnel systems was performed. Considerable work was spent on developing a numerical simulation of the wind tunnel environment to compare the results from the physical tests and to develop the corrections for the presence of the wind tunnel walls. These corrections were determined analytically and using the numerical solution. The numerical method gave better results due to the ability to model more details of the problem geometry. Results for all static derivatives were determined by wind tunnel tests and confirmed by the numerical solutions. The data was then corrected for the CG location and the wall interference effects and is now ready for use to benchmark the dynamic wind tunnel tests and the flight test data.
Reference Data: Mass Moments of Inertia

8 Mass Moments of Inertia .............. 121
8.1 Background and Theory
8.2 Methodology
8.3 Results
8.4 Summary
8. Mass Moments of Inertia

The text and the images of this part on the determination of the inertial properties of the test aircraft has been published in the ‘Aeronautical Journal’, managed by Cambridge University Press on behalf of the Royal Aeronautical Society [1]. The text is reproduced here with minor changes to fit into the overall thesis.

8.1 Background and Theory

Airframe inertial properties have been of interest since the early days of aviation because of their importance in characterising the handling qualities of aircraft. Their importance has now further increased due to the use of automated flight control systems. Modern model-based control systems require an accurate model of the airframe which includes the mass and inertial properties. The same is true for flight simulation applications. With the growing importance of UAVs, it is becoming crucial to obtain estimates for these properties for small scale flight vehicles.

To perform system identification (ID) on the experimental data obtained during this project, the inertial properties of the UAVs had to be determined accurately. On new aircraft designs, usually a sufficiently detailed CAD model would be used to determine the inertial properties of the airframe. This was not possible for the existing, commercial, small aircraft design under consideration here, because the airframe structure is too complex to draw up precisely in CAD without destructively dismantling the aircraft, particularly since no engineering drawings were available. The only possible method to estimate the inertial properties, in a time and cost effective manner, was by experiment.

In this part, the common single degree of freedom (1DoF) pendulum method and a novel, three degree of freedom (3DoF) pendulum method are tested and compared. The 3DoF method requires only a single swing test to obtain the entire inertia tensor at once.
This is potentially easier and less time consuming, because the test article has to be mounted onto and aligned with the test rig only once, compared to the three orientations required for the 1DoF method (four, if $I_{XZ}$ is required as well). The 3DoF method has not previously been applied to small fixed wing aircraft, leading to accuracy and usefulness of the method being investigated for this case.

The added mass phenomenon was discovered during the work for this part of the thesis. As mentioned before, the initial estimates of the inertial properties led to a mismatch between the static wind tunnel derivatives and the values from the dynamic tests (discussed in the next part). Re-visiting the literature on the topic led to this mismatch being identified as the added mass contributions. Since the exact magnitude of these contributions had to be determined, methods were developed and tested and are described in the following chapters.

8.1.1 Background

Reference [89] presents a review of available inertia measurement methods for various purposes. The main difference between the various methods is the use of either forced or free oscillations. Forced oscillation methods use an apparatus to force the test specimen into either translational or torsional oscillations [90] with the inertial properties being measured indirectly by the force or moment required to move the test article. These methods can be very accurate [91] but require a complicated and expensive apparatus that is prohibitive for typical low-budget university research.

Free oscillation methods use some sort of translational or torsional pendulum. These will oscillate freely under the influence of gravity alone, once displaced from rest and released. Pendulum designs have been used with a single suspension wire as the simplest method or using multiple suspension wires to create either a translational or torsional pendulum [92, 93]. The latter is more suitable for aircraft applications, because a multi-wire pendulum will hold the test article in a defined attitude.

The most widely used pendulum swing experiment for aircraft is the two-wire translational pendulum, used since the 1930s [94]. An extension of the basic method [26] discusses the aerodynamic effects affecting the pendulum motion due to the geometric properties of fixed wing aircraft. During the swing experiments, the airframe is immersed in a fluid (the surrounding air) and therefore the measured inertial properties will be affected by added mass due to the inertia of the air being accelerated by the pendulum [25, 26, 95]. Hence, the measured inertia $I_{\text{meas}}$ will be different from the test article inertia $I_{TA}$ measured in a vacuum. Reference [26] gives empirical methods to estimate the magnitude of those corrections, based on the geometric properties of typical aircraft components. These aerodynamic effects can add up to 20% on top of the true airframe inertias and are therefore very significant [26].

A further extension of the pendulum method [96] from 1948, indicates that a major source of difficulty of all previously mentioned methods is the requirement to accurately determine the pendulum length, that is the distance from the pivot to the pendulum CG. The difficulty lies with how the vertical CG position of the aircraft with respect to the pendulum pivot is to be determined. This method proposes to swing the pendulum with
two different arbitrary lengths, and use the two results to simultaneously solve for the pendulum lengths. This is potentially more accurate than measuring this quantity. A different method was developed in 1950 [97] to enable inertia measurements for large and heavy airframes. The pendulum method is impractical for such large airframes and it was replaced by a ground based spring support. Otherwise this method is similar to the pendulum method.

A novel method that uses multi-degree of freedom motion together with system identification [90], uses an apparatus which simultaneously allows rotation about the pitch and yaw axes and translation along the roll and pitch axes. The method does not require the measurement of the CG position of the test article, compared to the previously mentioned techniques. On the other hand, the required test apparatus is very complex.

There have been some existing references dealing specifically with small scale aircraft and UAVs [32, 33, 34, 35, 36]. Reference [32] compares different pendulum methods. The other references all use some form of the standard bifilar pendulum apparatus. On the very small scale, reference [98] reports on a test of a small quad-rotor flight vehicle. The paper also performs a comparison with a CAD modelling method and concludes that considerable errors between physical experiments and CAD modelling are possible, unless extreme care in the modelling is taken. Another very promising, novel method, [99], is based on a three degree of freedom pendulum, suspended from a three axis gimbal. It uses system identification of the pendulum motion to estimate the entire inertia tensor at once. This method is the basis for the second technique to be presented.

The results of the swinging tests of aircraft require substantial corrections due to aerodynamic effects. A very interesting report regarding these corrections looks into all kinds of error sources for a clock pendulum [95]. Some of these errors are very relevant for the current project and are indeed the same as reported by Soule and Miller [26]. Other error sources, such as the change in gravitational acceleration due to the position of the moon with respect to earth, are affecting only the long term stability of a clock pendulum and do not need to be accounted for during the short experiment durations of the airframe swing tests.

Using system ID of the pendulum motion instead of timing the oscillation periods [26, 96], has proven more reliable for the small inertias involved, because the system ID method uses the full motion data to estimate frequency and damping instead of characterising the motion from a few single data points from a timer. It is possible to obtain reasonably accurate data using a millisecond precision timer, but exploratory tests have shown that this requires many more experiment repeats than using the system ID method. Most recent implementations of inertia experiments use system ID for the data processing [90, 98, 99], and a similar approach has been taken for the presented work.

8.1.2 Theory of Pendulum Motion

1 DoF Pendulum

The equations of motion for a rigid body, single degree of freedom pendulum can be developed from the free body diagram in Figure 8.1. The test article with mass $m$, inertia $I$ and frontal area $S$ swings about pivot point $O$. The pendulum length or the distance between $O$ and the CG of the test article is $l$. 
From Euler’s second law, the rotational equation of motion for the rigid body pendulum can be written as

\[ M = I_O \dot{\omega} \tag{8.1} \]

where \( M \) is the applied moment about \( O \), \( I_O \) the inertia of the pendulum about \( O \) and \( \dot{\omega} \) the rotational acceleration of the pendulum.

For the system identification algorithm, the equations of motion need to be expressed in state space form, depending on the state derivative vector \( \dot{X} \) and the measurement vector \( Y \). The state vector for the pendulum is \( X = [\theta \ \omega]^T \), where \( \omega \) is the rotation rate of the pendulum and \( \theta \) the attitude angle. Using Eq. (8.1), the state rate equations for the rigid body pendulum become

\[ \dot{X} = \begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \omega \\ \frac{M}{I_O} \end{bmatrix} \tag{8.2} \]

The applied moment is the sum of the moment due to the gravitational acceleration, which is driving the motion, and the opposing moments due to aerodynamic drag and bearing friction. The moment due to the gravitational acceleration about point \( O \) is

\[ M_g = -mgl \sin \theta \tag{8.3} \]

The moment due to the drag about point \( O \) opposes the motion according to

\[ M_D = -\bar{q}SC_D \times l \times \text{sign}(\omega) \tag{8.4} \]

\[ = -0.5\rho SC_D \omega^2 l^3 \text{sign}(\omega) \tag{8.5} \]

with the dynamic pressure \( \bar{q} = 0.5\rho V^2 \) and \( V = \omega l \). The term \( \text{sign}(\omega) \) ensures that the drag force is always opposing the direction of the motion. Friction in the bearings is small compared to the drag of the test article and will not be modelled separately. It has a similar effect on the motion as the drag and therefore the identified drag coefficient will be slightly higher due to that friction.
So far, the equations of motion for the pendulum include the inertia of the pendulum about the pivot $O$. If the inertia of the test article about its CG is desired, the parallel axis theorem can be used to express the inertia $I_O$ of the pendulum as

$$I_O = I + ml^2$$  (8.6)

The measurement or output equations $Y$ consist of the two states $\theta$ and $\omega$ that will be measured directly by the rig instrumentation. Using the above derivations, the final equations of motion can be written as

$$\dot{X} = \begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \omega \\ \frac{M_\theta + M_D}{I + ml^2} \end{bmatrix}$$  (8.7)

$$Y = \begin{bmatrix} \theta \\ \omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ \omega \end{bmatrix}$$  (8.8)

Inspecting Eq. (8.7), some important implications for the experimental methodology can be immediately deducted. Firstly, because the expected inertias $I$ of the test article are going to be small, it is paramount to make the pendulum length $l$ as short as possible. For a small fixed wing aircraft, the parallel axis component of the total inertia of the pendulum, $ml^2$, will always be large compared to the test article inertia $I$, and will dominate the magnitude of the denominator in Eq. (8.7). If the pendulum is long, this will be even more significant. A long pendulum, therefore, makes it very difficult to identify the small inertia $I$ with good precision. This problem has also been referred to in [26] and [99].

Secondly, using knowledge of the system identification algorithm, it is also beneficial to estimate the full inertia of the pendulum about the pivot ($I_O = I + ml^2$) instead of estimating the inertia of the test article $I$ in isolation. The parallel axis theorem can be applied after the estimate for $I_O$ has been determined. The system ID algorithm perturbs each parameter by a small amount and calculates the significance of this perturbation onto the fit of the estimated system response to the measured data. Now, if Eq. (8.7) is used in its stated form, a small perturbation of the parameter $I$ will not have a significant effect on the magnitude of the denominator because $ml^2$ is always much larger in magnitude. Perturbing $I + ml^2$ instead gives a more robust response and the algorithm converges much faster and more accurately. Carefully considering those two issues during the experimental design is the first step to an accurate estimate of small scale fixed wing aircraft inertial properties.

To determine the pendulum length $l$, the vertical CG location of the test article must be known precisely with respect to a reference point [94]. This can be achieved by suspending the test article from a point longitudinally and vertically offset from the CG as shown in Figure 8.2(a). A vertical line from the suspension point to the longitudinal CG location will indicate the vertical CG position as shown in the Figure. This can then be measured from any convenient reference line on the test article as indicated. The length $l$ is then the distance from the pivot to this reference line plus the above measurement to the vertical CG of the test article. For a combination of objects, such as the test article
mounted on a support frame, as shown in Figure 8.2(b), the pendulum length $l$ to the combined CG can be found from a moment balance about point $O$

$$l = \frac{m_F l_F + m_{TA} l_{TA}}{m_F + m_{TA}}$$  \hspace{1cm} (8.9)

where the subscript $F$ denotes the properties of the support frame and $TA$ the properties of the test article.

The pendulum uses a support frame to hold the test article. This removes the need for any wire attachment points on the test article. Hence, after the swing test results are obtained, a final step is necessary to extract the inertial properties of the aircraft about its CG. This step is the removal of the support frame inertias from the results. The support frame properties have to be determined beforehand by a separate experiment or other means such as CAD modelling. A simple way of doing this would be to just subtract the inertias of the frame from the combined result, but this would lead to error because the inertia of the frame is typically measured about its own CG, which can be quite different from the combined CG location, as shown in Figure 8.2(b). In addition, the CG location of the aircraft is unlikely to be identical to the combined CG position as illustrated in the Figure. To separate the inertial properties of the frame and the aircraft accurately, it is therefore necessary to apply the parallel axis theorem, using the masses and dimensions given in Figure 8.2(b). This leads to the equation for the test article inertia $I_{TA}$ about its CG

$$I_{TA} = -[I_F + m_F \times l_F^2] - m_{TA} \times (l_1)^2 + I_{meas}$$  \hspace{1cm} (8.10)

where $I_F$ is the frame inertia about its CG, $m_F$ the frame mass, $m_{TA}$ the test article mass, $I_{meas}$ the combined inertia result from the system ID and the dimensions $l_i$ as defined in Figure 8.2(b). The resulting inertia $I_{TA}$ for a body with significant added mass is then comprised of the sum of the true or vacuum inertia of the body and the added mass term about that axis. If the true inertia is desired, the added mass term needs to be determined and subtracted from $I_{TA}$. 

Figure 8.2: Pendulum CG definitions
3 DoF Pendulum

The three-dimensional rigid body pendulum, similar to that introduced in [99], is an extension of the single degree of freedom pendulum from the previous section. Figure 8.3 shows a sketch of the support frame with the axes definitions. The figure also contains an image of the three degree of freedom gimbal used to suspend the pendulum.

In [99], all equations were developed in the body frame of the pendulum with origin at the CG of the pendulum. As mentioned above, this is not ideal because it leads to numerical issues during the system identification. Hence all equations are developed in the earth fixed frame with origin at the gimbal pivot point $O$. This increases the stability of the system ID procedure. The body axes inertia tensor about the pendulum CG is computed using the parallel axes theorem as a second step.

The derivation method of the equations of motion for the 3 DoF rigid body pendulum is similar to the single axis pendulum described before, as well as the rotational part of the full aircraft equations of motion developed earlier. Euler’s second law, now in three dimensions, describes the oscillatory motion of the rigid body as

$$\mathbf{M} = \mathbf{J} \ddot{\omega} + \omega \times \mathbf{J} \omega$$  \hspace{1cm} (8.11)

with $\mathbf{M}$ being the applied moment vector about the pivot point, $\mathbf{J}$ the inertia tensor of the pendulum about the pivot and $\omega$ the vector of angular velocities $p$, $q$ and $r$ of the pendulum. For a typical aircraft with symmetry about the $xz$ plane, the inertia tensor simplifies to

$$\mathbf{J} = \begin{bmatrix} I_x & 0 & I_{xz} \\ 0 & I_y & 0 \\ I_{xz} & 0 & I_z \end{bmatrix}$$  \hspace{1cm} (8.12)

The state space form of Eq. (8.11) with states $\omega = [p \quad q \quad r]^T$, as required for the system
ID algorithm, can be written as
\[
\dot{\omega} = J^{-1}(-\omega \times J\omega) + J^{-1}M
\] (8.13)

Similar to the 1 DoF pendulum, the applied moment is the sum of the moment due to gravity and the moments caused by bearing friction and aerodynamic drag. The gravitational force vector acting at the CG is \( F_g = mg \). The moment due to gravity can then be obtained by defining a vector \( \mathbf{R}_{O,CG} \) from the pivot point to the pendulum CG and taking the cross product
\[
\mathbf{M}_G = \mathbf{R}_{O,CG} \times \mathbf{F}_G
\] (8.14)

The components of the vector \( \mathbf{R}_{O,CG} \) in the earth fixed frame can be found by rotating the body axes vector \( \mathbf{R}_{O,CG} = [0 \ 0 \ l]^T \) between the pivot point and the pendulum CG (as shown in Figure 8.3) into the earth fixed axes using standard orthogonal transforms.

The damping terms are the moments due to bearing friction and the moment of the aerodynamic drag of the pendulum about the pivot point. The drag vector is the opposite of the velocity vector, which due to the complicated motion of the 3 DoF pendulum constantly changes direction. Also, the drag will be different about each axis because the aircraft shape. Therefore, the drag is treated in component form for each axis, similarly to the bearing friction. No other aerodynamic force is expected to create a significant moment about the pivot point to influence the motion. Drag and bearing friction have a similar effect on the motion and can be combined into a single vector \( \mathbf{M}_D = [M_{D,x} \ M_{D,y} \ M_{D,z}] \), unless their separate numerical values are of interest. For this project this was not the case, so the applied moment becomes
\[
\mathbf{M} = \mathbf{R}_{O,CG} \times \mathbf{F} - \mathbf{M}_D
\] (8.15)

Expanding \( \mathbf{M} \) into its components gives
\[
\begin{bmatrix}
    M_x \\
    M_y \\
    M_z
\end{bmatrix} = l \begin{bmatrix}
    \cos \phi \sin \theta \\
    -\sin \phi \\
    \cos \phi \cos \theta
\end{bmatrix} \times mg \begin{bmatrix}
    0 \\
    0 \\
    1
\end{bmatrix} - \begin{bmatrix}
    M_{D,x} \\
    M_{D,y} \\
    M_{D,z}
\end{bmatrix}
\] (8.16)

\[
= mgl \begin{bmatrix}
    -\sin(\phi) - M_{D,X} \\
    -\cos(\phi)\sin(\theta) - M_{D,Y} \\
    -M_{D,Z}
\end{bmatrix}
\] (8.17)

Above equations require the attitude angles \( \phi, \theta \) and \( \psi \) of the pendulum with respect to the earth fixed frame. These states can be added to yield the final state vector
\[
\mathbf{X} = [\phi \ \theta \ \psi \ p \ q \ r]^T
\] (8.18)

where the angles \( \phi, \theta \) and \( \psi \) are the attitude angles of the pendulum in the \( X, Y \) and \( Z \) axis, respectively.

The state rate equations for the Euler attitude angles have been developed earlier in this thesis. These, together with Eqs. (8.13) and (8.17) give the final equations of motion
for the 3 DoF rigid body pendulum

\[
\dot{\phi} = p + \tan(\theta)(q \sin(\phi) + r \cos(\phi)) \tag{8.19}
\]

\[
\dot{\theta} = q \cos(\phi) - r \sin(\phi) \tag{8.20}
\]

\[
\dot{\psi} = \frac{[q \sin(\phi) + r \cos(\phi)]}{\cos(\theta)} \tag{8.21}
\]

\[
\dot{p} = I_{xz} \Gamma[q(I_{xz}p + I_{xz}r) - I_{y}pq] + I_{z} \Gamma[q(I_{xz}p + I_{xz}r) - I_{y}qr] - I_{z}M_{x} \Gamma + I_{xz}M_{x}\Gamma \tag{8.22}
\]

\[
\dot{q} = \frac{p(I_{xz}p + I_{xz}r) - r(I_{xz}p + I_{xz}r)]/I_{y} + M_{y}/I_{y}} \tag{8.23}
\]

\[
\dot{r} = -I_{x} \Gamma[q(I_{xz}p + I_{xz}r) - I_{y}pq] - I_{xz} \Gamma[q(I_{xz}p + I_{xz}r) - I_{y}qr] + I_{xz}M_{z} \Gamma - I_{z}M_{z}\Gamma \tag{8.24}
\]

where \( \Gamma = 1/(I_{xz}^2 - I_{x}I_{z}) \). The measurement equation \( Y \) is

\[
Y = I \times [\begin{bmatrix} \phi & \theta & \psi & p & q & r \end{bmatrix}]^T \tag{8.25}
\]

where \( I \) is the identity matrix. The identified inertias about the pivot can then be transferred to the test article CG using the parallel axis theorem as before. The only terms affected are \( I_{x} \) and \( I_{y} \), while for \( I_{z} \) and \( I_{xz} \) the parallel axis theorem component is zero. The procedure of removing the support frame inertias from the solution is identical to the single axis pendulum method. Each of the inertia results contains the added mass contribution, similarly to the 1 DoF case. Since it was assumed that the added mass matrix is diagonal and all off-diagonal terms are zero, each result is the sum of the added mass and the vacuum inertia about the respective axis. If the assumption is correct, the true inertias can be found by subtracting the added mass terms from each result as before.

## 8.2 Methodology

The two pendulum experiments require an apparatus, which allows the test article to be mounted onto a support frame that can swing about a single axis for the 1 DoF case and about three axis for the 3 DoF case. Initially, the traditional knife edge suspension design [26] was used for the 1 DoF case and the three axis motion gimbal for the 3 DoF case. Testing has shown, however, that there is very little difference in the results if the gimbal is used for the 1 DoF case as well. One simply has to be careful to initiate the oscillations about a single axis only. Consequently only the gimbal suspension rig was used for this work, which allowed a common apparatus between the methods. The axes definitions for both experiments follow the standard flight mechanics conventions with \( X \) forward, \( Y \) out to the right wing and \( Z \) down.

In addition to the test aircraft, several other test articles were used to test and verify the method. These were generally simple shapes, where the true inertias could be derived from a CAD model. This allowed to benchmark the methods with a known object, and also to generate estimates for the added mass corrections required.
Chapter 8. Mass Moments of Inertia

8.2.1 Experiment Setup

The experimental setups for the 1 DoF and 3 DoF case are quite similar. The only difference is the design of the support frame. The 1 DoF frame allows the airframe to be placed on its side for the Z-axis measurements. To keep the pendulum length short, the frame has a bay for the test article to be placed in, as shown in Figure 8.4b. The 3 DoF frame, as shown in Figure 8.5, is flat because the test article can remain in a single position during the tests. The support frame is suspended from a three axis gimbal on four steel cables to form the pendulum. The gimbal contains high quality ball bearings to keep friction as low as possible. The gimbal is mounted onto a rigid cantilever beam framework as shown in Figure 8.4.

The data acquisition system can be seen at the bottom of the frame in Figure 8.4(a). It consists of the core components of the UAVmainframe with the main CPU and the reference IMU, since the only data required for the swing tests are the rotation rates and the attitude angles of the frame. This usage shows that the UAVmainframe can also be used as a versatile data acquisition system on the ground, where the ground station program is used as a real time data display application.

Further data required from the experiments are the mass $m$ of the pendulum, the gravitational acceleration $g$ and the distance from the pivot to the test article CG $l$. Gusev [100] reported on highly accurate measurements for the gravitational acceleration at the University of Sydney, where this work was conducted. For the pendulum length, the vertical CG position of the frame and test article was measured using the method of off-CG suspension illustrated in Figure 8.2(a). As discussed during the derivation of the equations of motion, the pendulum must be as short as possible for good results. The limiting factor is the wing span of the test airframe, which can be seen in Figure 8.4(b).
would be possible to use a shorter length for the $X$ and $Y$ cases, but during testing it was shown that the increases in accuracy did not warrant the extra effort in using two separate pendulum lengths.

1 DoF Pendulum

The support frame for the 1 DoF case was designed to accept the aircraft horizontally for the $X$ and $Y$ axes and and lying on its side for the $Z$ axis as shown in Figure 8.4. This way, all axes could be tested using the bifilar pendulum, avoiding the difficult trifilar pendulum method typically used for the $Z$ axis [26, 94, 96]. The difficulty of the trifilar pendulum is the requirement to excite the pendulum in pure rotation about the CG. In practice, it is very difficult, if not impossible, to set the pendulum in motion without causing translations. In addition, the aerodynamic damping is large due to the vertical fin of the airframe. This makes it difficult to obtain a sufficient number of oscillations for a valid measurement.

In each case, the aircraft CG was aligned precisely below the pivot point (along the $Z$ axis). An advantage of the additional degrees of freedom of the gimbal is that the aircraft does not have to be moved for the two horizontal cases, which minimises errors due to CG alignment. Each test was initiated from rest by manually deflecting the frame in one axis by about 5 degrees and letting go. Using only very small initial deflections reduces the damping due to the drag. The oscillation was allowed a few seconds to settle from the disturbance of the release into pure unforced oscillation before the data recording was started. Each recording was about 40 seconds and was repeated at least 5 times.

3 DoF Pendulum

The support frame for the 3 DoF experiment is shown in Figure 8.5. The test article has to be placed on the frame in a single, horizontal orientation only. This enables a shorter pendulum as shown in the figure. The frame is essentially a flat version of the previous frame, which makes it easier to locate its vertical CG location. It also reduces the inertias of the frame itself, which have to be removed from the test results. The data acquisition system is located in the plane of the frame and is identical to the 1 DoF case.

The tests were initiated manually as before. As will be discussed in the next section, the starting attitude is very critical to obtain good results with this method. It is also necessary to vary the initial deflections to obtain multiple solutions for a better judgement of the data quality. To achieve this, the pendulum was deflected between 5 and 10 degrees about $X$ and $Y$ and released. This was repeated multiple times with varying attitudes. Each recording was 50 seconds on average.

8.2.2 Data Processing

The recorded data was then prepared for use with the output error system identification algorithm [22]. Due to the long data sets and the low frequency content the sample rate was reduced by a third from the nominal rate of 100Hz without loss of information. The down-sampling was done by low pass filtering the data with a 10Hz cut-off and then re-sampling it at 33.3 Hz. This reduces the runtime of the algorithm considerably without affecting the accuracy of the parameter estimates.
Chapter 8. Mass Moments of Inertia

1 DoF Pendulum

The output error method for the 1 DoF case uses the equations of motion of Eq. 8.7. Parameters to estimate are the inertia $I_O$ and the drag coefficient $C_D$. Initial values for the parameters were generated by trial and error, based on available information like previous test results or CAD results for the reference bodies. Figure 8.6 shows an example dataset with the model output $Y$ and the experimental data $Z$ plotted on top of each other for comparison. The figure also contains plots of the residuals $Z - Y$.

The Figure shows excellent agreement between the data and the model in both frequency and damping of the oscillation. The residuals for the pitch rate $q$ are purely random noise with less than 1% of the signal amplitude. The pitch angle $\theta$ residuals still contain some sinusoidal component. This is caused by a small phase error between the model and the experimental data. This phase error is most likely the result of a small error in the initial value for $\theta$ and is of no consequence for the accuracy of the results. The quality of the model fit is similar across all three axes and it is repeatable to a very high degree of accuracy as will be shown later. Therefore, the model of the pendulum motion as developed above is correctly describing the data and can be used to identify the inertia of the pendulum with high accuracy.

3 DoF Pendulum

The system ID for the 3 DoF case uses the model equations Eqs. (8.24). Estimated parameters are the four inertia terms $I_x$, $I_y$, $I_z$ and $I_{xz}$, together with the three damping terms. Figure 8.7 shows a typical result of the process, plotting the model over the measured data and showing the residuals similarly to the 1 DoF case. The model fit to the data is excellent, which proves that Eqs. (8.24) describe the motion correctly. The assumption of any aerodynamic force other than drag being insignificant appears correct. There are some minor phase differences in the residuals which are most likely caused
by an imperfect pendulum release but these have no influence on the final results. On release, the pendulum probably may have not been fully at rest or the disturbance of the release may not have completely dissipated throughout the recording.

The Z-axis rotation is unforced and purely the result of the cross-coupling between the axes in the equations of motion. The $I_{ZZ}$ component is also the largest element in the tensor for a typical aircraft. Figure 8.7 shows that the $Z$ axis, denoted $\psi$ in the Figure, has very low information content due to these physical properties. Over the 20 seconds duration plotted, only a single oscillation occurs, compared to 12 oscillations in the $Y$ axis.

This low information content in the $Z$ axis has several implications for the experiment execution as well as for the processing of the data. During execution, it is important to choose starting attitudes which maximises the $Z$ oscillations. This requires considerable trial and error, because these starting attitudes depend on the ratio between the inertias and the damping in the $Z$-axis of the test article. They are therefore different for each body tested.

During processing, the system ID algorithm is essentially attempting to establish the frequency and damping of the motion of the pendulum. Attempting to do this with a single oscillation exceeding a period of 20 seconds will not be very accurate. This shows up in the uncertainties reported by the algorithm and is illustrated in Figure 8.8. As shown, the uncertainties for the $Z$ axis are an order of magnitude larger than for the $X$ and $Y$ axes. Naturally, this leads to a larger deviation between repeats, which in turn,
requires more runs than the 1 DoF pendulum to achieve similar accuracy. Another effect of the low frequency in the Z axis is that the time window chosen for the system ID can make a large difference in the results. This applies to the length of the window as well as to the location. For example, a 20 second window of a particular dataset can yield an estimate for $I_z$ of 0.6 or 0.56, which equals a 7% difference, depending on the starting point. The same dataset returns an $I_z$ estimate of 0.6 or 0.62 for a 20 second and a 30
second data window, respectively. This equals a 3% difference. Finally, some datasets with particular starting attitudes do not converge at all. The best results were obtained from starting attitudes that result in at least a full period of oscillation in the Z axis. Depending on the test article and its damping properties, this may not be possible at all and a careful selection of the data window is required. This is the case for the aircraft under consideration, where the period of the Z oscillation can exceed 40 seconds.

To obtain a trustworthy result for $I_z$ with the 3 DoF method, it is therefore necessary to repeat the experiment often and use critical engineering judgement when interpreting and selecting the results. For this project, the final tests were repeated 10 times. Each dataset was then processed with a 20 and 30 second window and the final results were averaged. Obvious outliers with unrealistic parameter estimates were discarded in the process.
8.2.3 Verification

As the next step, it is necessary to investigate the magnitudes of corrections $I_{mf}$ required to extract the true inertial properties of the test aircraft. As mentioned above, the empirical formulas developed in [29] did not predict accurate values for $I_{mf}$ for the test aircraft. For example, using the method in [29], $I_{mf}$ for the X-axis was calculated be 0.02 kg m$^2$. The correct value, as determined experimentally, is 0.12 kg m$^2$. This is a difference of 83%. The testing has also revealed that the corrections are not the same for the two swinging methods. It is therefore necessary to develop a two-step verification and calibration process to estimate $I_{mf}$ for the test aircraft.

The first step of verification is unique to this project. The aircraft was tested extensively in the wind tunnel to determine its aerodynamic properties. This was done using conventional static tests as well as more involved dynamic tests, as presented in part V. Some of the stability derivatives can be estimated from either static or dynamic tests and the results should be identical [101]. The dynamic test results contain the inertias of the aircraft while the static tests do not. The inertia estimates for the test aircraft could therefore be verified in a unique way by matching the wind tunnel test results using the inertial properties from the swing tests and evaluating the differences.

To estimate the value of $I_{mf}$ experimentally, a known body was used as a benchmark. As shown in Figure 8.9, this body initially was simply a flat plate model of the test aircraft. It was made from particle board with approximately uniform density and had similar weight and inertia properties as the test aircraft. The flat plate was modelled in a CAD package to determine its inertias with high accuracy.

Based on the derivations of the correction methods in [26, 29], the corrections for the test aircraft were calculated from the flat plate tests by using the differences in the results to the CAD model. Preliminary testing, using the wind tunnel data, showed that
the flat plate corrections were too small. The corrections, however, were closer to the empirical values in reference [29]. The experimentally determined correction for the flat plate model were $0.05\text{kg m}^2$, versus $0.02\text{kg m}^2$ from the calculations. Since the expected value for the full aircraft was $0.12\text{kg m}^2$, the flat plate model does not capture the full extent of the added mass quantity for the test aircraft. The reason for this must be the three dimensional shape of the fuselage and vertical fin of the test aircraft, since the wings are essentially a thick flat plate (although streamlined). During swinging motion about the X-axis, the flow is perpendicular to the fuselage walls and there will be significant displacement of air. This might lead to a larger body of air being affected by the pendulum and hence $I_{mf}$ will be increased.

The solution to the problem is shown in Figure 8.10. A foam simulator of the test aircraft was constructed to have similar volume and surface area. It was then placed on the support frame and loaded up with aluminium bars to have similar inertial properties as the test aircraft. The arrangement is shown in Figure 8.10(b). Figure 8.11 illustrates the CAD model for the simulator configuration used for the 3 DoF case. Note the different location of the aluminium bars compared to the 1 DoF case (Figure 8.10). Because the 3 DoF motion is more violent, the bars had to be secured to the frame more rigidly than in the 1 DoF case, resulting in the given configuration.

Since the foam density and dimensions, as well as the properties of the aluminium bars were known, the simulator could be modelled in the CAD package relatively accurately and its inertial properties determined this way. Given the geometric similarity of the simulator to the test aircraft, it is expected that the values for $I_{mf}$, obtained from testing this simulator will be fairly close to the corrections required for the test aircraft.

The mass $m$ and the pendulum length $l$ were measured with a precision of $\pm 1\text{g}$ and $\pm 1\text{mm}$, respectively. To judge whether this accuracy is sufficient, a brief sensitivity study
was performed using the 3 DoF method. Firstly, a pendulum length change of 2mm during the system ID resulted in an error in the inertia estimate of 0.7% or less. Then a run was performed with the mass increased by 5g. This lead to errors of less than 0.5%. It is not expected to be able to estimate the size of the errors due to the aerodynamic effects with similar or better accuracy and therefore the precision of the measurements of $m$ and $l$ was considered sufficient.

8.3 Results

The first dataset presented and discussed in this section is for the flat plate, then for the foam test aircraft simulator, and finally for the test aircraft itself. All datasets include the support frame. All tables list the identified results for the respective inertias together with the uncertainty of the parameter estimate as reported by the output error algorithm for multiple runs. The mean and standard deviation of the results are calculated and compared to the CAD results. The differences are listed as an absolute value and as a percentage.

Because the support frame is different between the 1 DoF and the 3 DoF cases, the estimated inertias of the full pendulum cannot be directly compared between the methods. This is not a serious limitation, because corresponding CAD models were used to determine the corrections required.

8.3.1 Flat Plate

The 1 DoF results for the flat plate from Figure 8.9 are presented in Table 8.1. The repeatability is excellent and the reported uncertainties are small over five runs in all axes. The comparison with the CAD results shows that the inertias of the flat plate are over predicted in all axes, as expected. This is caused by the added mass inertia $I_{mf}$. Contrary to the corrections from [26], where the X-axis required the largest correction
The Z-axis correction is also more than twice as big as the result computed with the method in [29], as discussed before.

The 3 DoF inertia estimates for the flat plate are listed in Table 8.2. The flat plate is symmetrical about the \(xz\) and \(xy\) plane. Therefore the product of inertia \(I_{xz}\) of the plate is zero and the total \(I_{zz}\) of the plate on the frame is negligibly small. As a result the system ID algorithm was unable to identify this parameter reliably, because it is not sufficiently observable. It was therefore decided to remove the \(I_{xz}\) from the parameter vector and just identify \(I_x\), \(I_y\) and \(I_z\), while setting \(I_{xz}\) to zero. This also reduced the uncertainties of \(I_z\).

Similarly to the 1 DoF cases, the repeatability of the \(X\) and \(Y\) axes are very good, with small uncertainties. The \(Z\) axis, however, has uncertainties an order of magnitude larger than the two other axes. The estimates for \(I_z\) vary up to 10% between the runs. Comparing the differences between 3 DoF experimental results and the CAD data to the 1 DoF case in Table 8.1, there are similar magnitudes of corrections for the \(X\) and \(Y\) axes. The \(Z\) axis, however, has a significantly smaller correction due to the added mass effect in the 3 DoF case. This will be even more pronounced for the 3-dimensional

### Table 8.1: 1 DoF inertia test results of the flat plate with support frame

<table>
<thead>
<tr>
<th>Run</th>
<th>(I_x) [(kg m^2)]</th>
<th>(I_y) [(kg m^2)]</th>
<th>(I_z) [(kg m^2)]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.651 ± 0.0002 0.01%</td>
<td>0.344 ± 0.0003 0.02%</td>
<td>0.835 ± 0.0007 0.02%</td>
</tr>
<tr>
<td>2</td>
<td>0.654 ± 0.0006 0.02%</td>
<td>0.343 ± 0.0002 0.01%</td>
<td>0.829 ± 0.0004 0.01%</td>
</tr>
<tr>
<td>3</td>
<td>0.650 ± 0.0003 0.01%</td>
<td>0.346 ± 0.0002 0.01%</td>
<td>0.829 ± 0.0003 0.01%</td>
</tr>
<tr>
<td>4</td>
<td>0.648 ± 0.0003 0.01%</td>
<td>0.344 ± 0.0001 0.01%</td>
<td>0.830 ± 0.0003 0.01%</td>
</tr>
<tr>
<td>5</td>
<td>0.649 ± 0.0002 0.01%</td>
<td>0.344 ± 0.0002 0.01%</td>
<td>0.829 ± 0.0005 0.01%</td>
</tr>
<tr>
<td>mean</td>
<td>0.650 ± 0.002</td>
<td>0.344 ± 0.001</td>
<td>0.831 ± 0.003</td>
</tr>
<tr>
<td>CAD</td>
<td>0.597</td>
<td>0.298</td>
<td>0.778</td>
</tr>
<tr>
<td>(\Delta)</td>
<td>0.053 8.2%</td>
<td>0.046 13.4%</td>
<td>0.053 6.4%</td>
</tr>
</tbody>
</table>

by a factor of 2 or more, the flat plate here has the highest correction in the Y-axis. The X-axis correction is also more than twice as big as the result computed with the method in [29], as discussed before.

### Table 8.2: 3 DoF inertia test results for the flat plate with the support frame

<table>
<thead>
<tr>
<th>Run</th>
<th>(I_x) [(kg m^2)]</th>
<th>(I_y) [(kg m^2)]</th>
<th>(I_z) [(kg m^2)]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.577 ± 0.0003 0.1%</td>
<td>0.279 ± 0.0003 0.1%</td>
<td>0.787 ± 0.013 0.8%</td>
</tr>
<tr>
<td>2</td>
<td>0.574 ± 0.0004 0.1%</td>
<td>0.275 ± 0.0003 0.1%</td>
<td>0.746 ± 0.025 1.7%</td>
</tr>
<tr>
<td>3</td>
<td>0.587 ± 0.0003 0.1%</td>
<td>0.279 ± 0.0002 0.1%</td>
<td>0.806 ± 0.021 1.3%</td>
</tr>
<tr>
<td>4</td>
<td>0.582 ± 0.0005 0.2%</td>
<td>0.285 ± 0.0003 0.1%</td>
<td>0.818 ± 0.027 1.6%</td>
</tr>
<tr>
<td>5</td>
<td>0.581 ± 0.0005 0.2%</td>
<td>0.285 ± 0.0002 0.1%</td>
<td>0.830 ± 0.051 3.1%</td>
</tr>
<tr>
<td>mean</td>
<td>0.580 ± 0.0009</td>
<td>0.281 ± 0.0008</td>
<td>0.797 ± 0.059</td>
</tr>
<tr>
<td>CAD</td>
<td>0.541</td>
<td>0.2425</td>
<td>0.7825</td>
</tr>
<tr>
<td>(\Delta)</td>
<td>0.039 6.7%</td>
<td>0.039 14%</td>
<td>0.015 1.9%</td>
</tr>
</tbody>
</table>
bodies tested next. For both cases, the corrections due to the added mass are too small to satisfy the verification test using the aerodynamic derivatives. Clearly, the flat plate model of the test aircraft is not a suitable simulator for the actual aircraft.

8.3.2 Foam Simulator

To estimate the magnitude of corrections required due to the added mass effect \( I_m \), the test aircraft foam simulator from Figure 8.10 was tested next. The 1 DoF results are listed in Table 8.3. Again the repeatability is excellent with very small uncertainties, leading to high confidence in the results. Comparing the mean values for the estimated inertias to the CAD predictions shows significant deviations for all axes due to the added mass. The additional inertias are 25% for the \( X \) axis, 18% for the \( Y \) axis and 15% for the \( Z \) axis, considerably more than for the flat plate and the full scale airframe in [26]. The \( I_{XZ} \) term is too small to be estimated reliably and no attempt has been made to test for it in the 1 DoF experiments.

The 3 DoF test results are listed in Table 8.4. Each run was processed with a 20 second and 30 second data window, as discussed above. The results for the \( X \) and \( Y \) axis perfectly repeatable, similarly to the flat plate and they do not depend on the data window. The \( Z \) axis shows more spread across the runs and the results also vary depending on the data window.

Comparing the added-mass corrections between the two cases, significant differences can be seen between the two methods. The \( X \) and \( Y \) corrections of the 3 DoF case are about half of the 1 DoF experiment. The correction for the \( Z \) axis is even negative for the 3 DoF case. Both corrections are, however, correct for their respective methods, as will be shown in the next section. It appears that the added mass corrections depend on the type of motion the pendulum performs. A possible explanation may be that the assumption of constant integral \( \zeta \), that is the volume of affected fluid in the derivation of the added mass in Section 1.6, is violated by the larger, more complex motion pattern of the 3 DoF pendulum. More research with different bodies would be required to fully explain this phenomenon. Another explanation may be that the added mass matrix takes

<table>
<thead>
<tr>
<th>Run</th>
<th>( I_X [kg\cdot m^2] )</th>
<th>( I_Y [kg\cdot m^2] )</th>
<th>( I_Z [kg\cdot m^2] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.481 ± 0.0002</td>
<td>0.492 ± 0.0006</td>
<td>0.744 ± 0.0004</td>
</tr>
<tr>
<td>2</td>
<td>0.481 ± 0.0003</td>
<td>0.491 ± 0.0004</td>
<td>0.745 ± 0.0004</td>
</tr>
<tr>
<td>3</td>
<td>0.482 ± 0.0002</td>
<td>0.492 ± 0.0004</td>
<td>0.745 ± 0.0005</td>
</tr>
<tr>
<td>4</td>
<td>0.482 ± 0.0004</td>
<td>0.488 ± 0.0002</td>
<td>0.745 ± 0.0004</td>
</tr>
<tr>
<td>5</td>
<td>0.486 ± 0.0005</td>
<td>0.486 ± 0.0003</td>
<td>0.744 ± 0.0003</td>
</tr>
<tr>
<td>mean</td>
<td>0.482 ± 0.002</td>
<td>0.490 ± 0.003</td>
<td>0.745 ± 0.001</td>
</tr>
<tr>
<td>CAD</td>
<td>0.361</td>
<td>0.404</td>
<td>0.631</td>
</tr>
<tr>
<td>( \Delta )</td>
<td>\textbf{0.12}</td>
<td>\textbf{0.09}</td>
<td>\textbf{0.11}</td>
</tr>
<tr>
<td></td>
<td>25.1%</td>
<td>17.6%</td>
<td>15.3%</td>
</tr>
</tbody>
</table>
8.3 Results

Table 8.4: 3 DoF inertia test results for the foam simulator with the support frame (each run reported for 20 sec and 30 sec data)

<table>
<thead>
<tr>
<th>Run</th>
<th>$I_X$ [kg m$^2$]</th>
<th>$I_Y$ [kg m$^2$]</th>
<th>$I_Z$ [kg m$^2$]</th>
<th>$I_{XZ}$ [kg m$^2$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.279 ± 0.001</td>
<td>0.432 ± 0.001</td>
<td>0.506 ± 0.024</td>
<td>0.002 ± 0.003</td>
</tr>
<tr>
<td>2</td>
<td>0.535 ± 0.012</td>
<td>0.520 ± 0.010</td>
<td>0.548 ± 0.012</td>
<td>0.004 ± 0.002</td>
</tr>
<tr>
<td>3</td>
<td>0.530 ± 0.012</td>
<td>0.513 ± 0.001</td>
<td>0.547 ± 0.014</td>
<td>0.003 ± 0.003</td>
</tr>
<tr>
<td>4</td>
<td>0.528 ± 0.001</td>
<td>0.435 ± 0.001</td>
<td>0.513 ± 0.001</td>
<td>0.002 ± 0.004</td>
</tr>
</tbody>
</table>

mean | 0.282 ± 0.005 | 0.433 ± 0.002 | 0.533 ± 0.037 | 0.004 ± 0.005 |

CAD | 0.2147 | 0.3857 | 0.5914 | 0.0038 |

$\Delta$ | 0.067 | 23.8% | 0.047 | 10.9% | -0.058 | -10.9% | -

The foam simulator is geometrically similar to the test aircraft and was set up to have comparable inertial properties. Therefore, the deviations between the experimental data and the CAD results for the respective methods are expected to be valid to correct the experimental results for the unknown inertial properties of the test aircraft, as will be discussed next.

8.3.3 Test Aircraft

The 1 DoF test results for the test aircraft are listed in Table 8.6. As before, the test results are fully repeatable over five runs and the reported uncertainties on the estimated inertias are very low. No test was performed for $I_{XZ}$ in the 1 DoF case. The 3 DoF tests were run 20 times to improve the accuracy of the $Z$ axis. The individual results are plotted in Figure 8.7 and a summary is listed in Table 8.5. The standard deviations for the 3 DoF case are at least an order of magnitude larger than for the 1 DoF case. The mean values were then corrected using the factors determined with the foam simulator for the respective method above. It should be noted that due to the different support frames used for each method, these results are not directly comparable.

Table 8.5: 3 DoF inertia test results including the support frame of the test aircraft

<table>
<thead>
<tr>
<th></th>
<th>$I_X$ [kg m$^2$]</th>
<th>$I_Y$ [kg m$^2$]</th>
<th>$I_Z$ [kg m$^2$]</th>
<th>$I_{XZ}$ [kg m$^2$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean</td>
<td>0.340 ± 0.013</td>
<td>0.449 ± 0.006</td>
<td>0.550 ± 0.080</td>
<td>-0.011 ± 0.006</td>
</tr>
<tr>
<td>Corr.</td>
<td>-0.067</td>
<td>-0.047</td>
<td>+0.058</td>
<td></td>
</tr>
<tr>
<td>Final</td>
<td>0.27</td>
<td>0.40</td>
<td>0.61</td>
<td>-0.011</td>
</tr>
</tbody>
</table>
Table 8.6: 1 DoF inertia test results of the test aircraft including the support frame

<table>
<thead>
<tr>
<th>Run</th>
<th>$I_X$ [$kg,m^2$]</th>
<th>$I_Y$ [$kg,m^2$]</th>
<th>$I_Z$ [$kg,m^2$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.499 ± 0.0002</td>
<td>0.518 ± 0.0002</td>
<td>0.750 ± 0.0005</td>
</tr>
<tr>
<td>2</td>
<td>0.498 ± 0.0002</td>
<td>0.516 ± 0.0003</td>
<td>0.745 ± 0.0003</td>
</tr>
<tr>
<td>3</td>
<td>0.499 ± 0.0002</td>
<td>0.517 ± 0.0004</td>
<td>0.746 ± 0.0003</td>
</tr>
<tr>
<td>4</td>
<td>0.497 ± 0.0002</td>
<td>0.518 ± 0.0002</td>
<td>0.746 ± 0.0003</td>
</tr>
<tr>
<td>5</td>
<td>0.499 ± 0.0001</td>
<td>0.518 ± 0.0002</td>
<td>0.746 ± 0.0004</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$mean$</th>
<th>$Correction$</th>
<th>$Final$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_X$</td>
<td>0.498 ± 0.001</td>
<td>-0.12</td>
<td>0.378</td>
</tr>
<tr>
<td>$I_Y$</td>
<td>0.517 ± 0.001</td>
<td>-0.09</td>
<td>0.427</td>
</tr>
<tr>
<td>$I_Z$</td>
<td>0.747 ± 0.002</td>
<td>-0.11</td>
<td>0.637</td>
</tr>
</tbody>
</table>

Table 8.7 lists the final results for the test aircraft inertial properties. The support frame contributions were removed using Eq. 8.10. The agreement between the two methods is good, only the $X$ axis shows a difference above 10%. Given the discussed issues with the 3 DoF method and the added mass corrections, this is a good result. More research, however, should be done to improve the 3 DoF method before it can be preferred over the 1 DoF method for small fixed wing aircraft. Once perfected, the 3DoF method will be easier and quicker than the traditional 1 DoF method, because the aircraft only needs to be swung in a single orientation to obtain the entire inertia tensor. For the current project, the 1 DoF results were used for the subsequent flight dynamic analysis. Only the $I_{XZ}$ value was used from the 3 DoF experiment. The final results also confirm that the added mass corrections are specific to the test article and the experiment used. For each test article it will be necessary to construct a reference body of known inertial properties and determine the added mass corrections with the described methods. At this stage, no method exists to transfer the corrections between test articles or the two experiments. This will require a large database of different airframes shapes to be tested, similarly to the design data available for full scale aircraft [103], and an improved understanding of the reasons behind the differences in added mass corrections between the two experiments.

Table 8.7: Final test results without the support frame [$kg\,m^2$]

<table>
<thead>
<tr>
<th>Axis</th>
<th>1 DoF</th>
<th>3 DoF</th>
<th>$\Delta$ (in %)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_X$</td>
<td>0.22</td>
<td>0.19</td>
<td>-13.6%</td>
</tr>
<tr>
<td>$I_Y$</td>
<td>0.31</td>
<td>0.31</td>
<td>-</td>
</tr>
<tr>
<td>$I_Z$</td>
<td>0.51</td>
<td>0.48</td>
<td>-5.9%</td>
</tr>
<tr>
<td>$I_{XZ}$</td>
<td>-</td>
<td>-0.01</td>
<td>-</td>
</tr>
</tbody>
</table>
Table 8.8: Stability derivative benchmark using the 1 DoF results

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>$C_{l_{x}}$</td>
<td>-0.061</td>
<td>-0.039</td>
<td>-36%</td>
<td>-0.061</td>
</tr>
<tr>
<td>Y</td>
<td>$C_{m_{y}}$</td>
<td>-0.948</td>
<td>-0.74</td>
<td>-22%</td>
<td>-0.956</td>
</tr>
<tr>
<td>Z</td>
<td>$C_{n_{z}}$</td>
<td>0.087</td>
<td>0.072</td>
<td>-17%</td>
<td>0.089</td>
</tr>
</tbody>
</table>

Table 8.8 lists the results of the aerodynamic derivatives from the wind tunnel tests (as a preview of the next part). The static test data is used as the reference. It should be matched by the results from the dynamic tests, which include the inertial properties in the identified model parameters. The table shows that the obtained derivatives have large deviations when determined using the uncorrected inertial properties of the test aircraft. With the corrected data the match to the static data is excellent. In the Z-axis, the difference is higher than for the other two axes, which is most likely caused by either a small remaining error in the added mass correction for that axis, or other errors in the parameter identification of the Dutch roll mode caused by the turbulence in the wind tunnel. These results demonstrate that the devised test- and correction methods are working correctly. The table also emphasises that without performing these corrections, the estimates for the inertial properties of the test aircraft will be in error by a significant amount.

As required for the processing of the flight test data, Table 8.9 lists the inertial properties of the test airframe including the added mass terms and with the support frame inertias removed. The reasons and corresponding results are discussed in Part VII. Due to the yet to be explained nature of the added mass correction for the 3 DoF value for $I_z$ there is a large discrepancy between the results of the two methods for that value. At this stage it is assumed that the 1 DoF results will be correct, which is confirmed by flight test data. The added mass contributions for the 3 DoF experiment requires further research to fully explain the findings, which will be included in the discussion at the end of the thesis.

Table 8.9: Final results including the added mass components [kg m^2]

<table>
<thead>
<tr>
<th>Axis</th>
<th>1 DoF</th>
<th>3 DoF</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_X$</td>
<td>0.34</td>
<td>0.29</td>
</tr>
<tr>
<td>$I_Y$</td>
<td>0.40</td>
<td>0.36</td>
</tr>
<tr>
<td>$I_Z$</td>
<td>0.62</td>
<td>0.42</td>
</tr>
<tr>
<td>$I_{XZ}$</td>
<td>-</td>
<td>-0.01</td>
</tr>
</tbody>
</table>
8.4 Summary

Two methods have been tested and compared for the determination of the inertial properties of a small, fixed wing aircraft. The first method uses the standard single degree of freedom pendulum method. The second experiment implemented a novel and potentially easier three degree of freedom pendulum method, which yields the entire inertia tensor from a single swing test. Both methods used system identification of the pendulum motion to estimate the inertial properties of the test aircraft.

Substantial corrections due to the effect of added mass, caused by the acceleration of the pendulum immersed in the surrounding air, need to be applied to the results to obtain the correct values for the inertial properties of the test aircraft. It has been found that the methods presented in previous literature to determine the corrections for full scale aircraft do not give the correct results for the small scale aircraft. At this stage, the only feasible method to generate these corrections are swing tests with a geometrically similar object of known inertial properties. It has also been found that the corrections are specific for the type of experiment and cannot be transferred between the 1 DoF and 3 DoF case. This has to be kept in mind when applying one of the methods.

Several benchmarking methods, including the innovative use of results obtained from static and dynamic wind tunnel tests, have been utilised to prove the accuracy of the results obtained with both swinging methods. Very good agreement between the experiments and the reference data was achieved. Both methods agree well, but there is more research required before the three degree of freedom pendulum method can be used with equally high confidence to determine inertial properties of small, fixed wing aircraft.
## Dynamic Wind Tunnel Testing

9.1 Background
9.2 Experiment Design

## Short Period Mode

10.1 Model Structure
10.2 Elevator Input Design
10.3 Verification
10.4 Results
10.5 Variation of CG Location
10.6 Short Period Mode Summary

## Dutch Roll Approximation

11.1 Model Structure
11.2 Input Design and Verification
11.3 Results
11.4 Dutch Roll Mode Summary

## Roll Mode

12.1 Model Structure
12.2 Input Design and Verification
12.3 Results
12.4 Roll Mode Summary

## Lateral Combined Manoeuvre

13.1 Model Structure
13.2 Input Design and Verification
13.3 Results
13.4 Lateral Combined Summary

Reference Data: Dynamic Derivatives
9. Dynamic Wind Tunnel Testing

In order to fully benchmark the flight test results, the dynamic derivatives of the test aircraft are required together with the results of the static tests and the inertial properties determined in the previous chapters. Dynamic (or unsteady) derivatives are the properties of an aircraft depending on the rotation rates \( p, q \) and \( r \), as well as the rates of change in angle of attack \( \dot{\alpha} \) and sideslip \( \dot{\beta} \). These parameters are much more difficult to determine due to the dependence on a moving airframe. Several methods exist, however, and after a review of these methods this part of the thesis will present the method and the results of using a three axis motion gimbal to simulate flight in three degrees of freedom (3DoF) in the wind tunnel.

This part is organised as follows: The experimental method is introduced first, together with some background. Then the format for the result presentation is introduced. This format will be used for all dynamic data in this thesis. This is followed by a detailed presentation of the four input patterns tested. Elevator inputs to excite the short period mode come first and then in the lateral axis rudder inputs for dutch roll motion, aileron inputs for the roll mode and finally a lateral combined input with rudder and aileron moving both during an input sequence. At the end, all results are collected and compared across the different experiments to select the best data as a reference for the flight tests.

9.1 Background

The most basic methods to determine the dynamic derivatives of an aircraft are the empirical equations given in many textbooks on aircraft design and flight mechanics [39]. These are first pass methods for use during preliminary design or for classroom exercises and typically only include the effects of the most important lifting surfaces on the parameters. Databases of historical data, such as DATCOM [104], contain the
properties of many previous aircraft designs and use curve fitting methods to estimate the properties of a particular aircraft geometry. These methods give reasonable results for full scale aircraft with sufficiently high Reynold’s number flow but their performance for small scale aircraft has not yet been thoroughly investigated.

Numerical methods exist in the form of panel codes or Navier-Stokes solutions. Both allow a more detailed model of the aircraft geometry and the flow conditions. Panel methods for unsteady derivatives are AVL [105] and TORNADO [106] as well as some other commercial products. The two mentioned codes are planar methods which use only the camber line of the lifting surfaces and have limited ability to model fuselages, unlike PanAir, which was used in the previous chapters but cannot do dynamic derivatives. Example computations with AVL will be included in the results of this chapter for comparison. Full Navier-Stokes solutions allow detailed models of the airframe and the flow conditions, but are still very time- and resource intensive, especially for an unsteady solution of a full aircraft configurations. Nevertheless, these methods have been used successfully for full scale aircraft [107, 108, 109, 110]. On the other hand, no literature is yet available on a benchmarked Navier-Stokes solution of a small scale aircraft with its more challenging flow conditions due to the low Reynolds numbers involved.

Experimental methods used to determine dynamic derivatives require some kind of motion rig inside a wind tunnel to measure the forces and moments acting on the test article while moving in the flow. Reference [111] gives a good overview of the various experimental methods used for that purpose. Most of these methods use some form of forced oscillation wind tunnel balances, where the test aircraft is mounted rigidly onto the balance and is then oscillated by some form of actuator. The forces and moments acting on the test article are measured and the dynamic derivatives can be extracted from the data. These balances for forced oscillation are very complicated and expensive [112, 113, 114, 115, 116, 117] and not many wind tunnels have that capability. The forced oscillation method, however, is the only method capable of determining the (\(\dot{\alpha}\)) and (\(\dot{\beta}\)) parameters, since these can only be estimated from pure translational motion of the airframe. Any simultaneous rotation will cause the inflow angle rate parameters to become correlated with the rotational parameters (\(C_{m_{\alpha}}\) and \(C_{m_{\beta}}\) for example) and they cannot be separated.

Another experimental method to determine the dynamic derivatives of an aircraft is the wind tunnel flight method, which was used for this project as well. There, the aircraft is mounted on a motion gimbal and can freely rotate about the three axes, but usually not translate. The aircraft can then be ‘flown’ in the wind tunnel, using its control surfaces for control via a radio transmitter. This allows to execute similar manoeuvres as in flight and parameter ID methods can be used, in conjunction with adapted models, to estimate the required derivatives [118, 119, 120, 121, 122, 123, 124, 125]. The drawback of this method is the requirement of having high quality sensors and data acquisition systems installed inside the test airframe, a requirement that has only recently become practical due to the miniaturisation of electronic components. Any wiring connecting the aircraft with external systems will interfere with the motion and potentially void the results. For this project, the UAVmainframe was designed for that purpose and since the wind tunnel
and flight test aircraft were identical, this wind tunnel flight method proved to be an ideal test case for the avionics system before using it in free flight.

9.2 Experiment Design

9.2.1 Setup

The wind tunnel rig for the dynamic tests uses the mast of the static balance and replaces the balance head with a 3DoF motion gimbal. The gimbal was installed inside the aircraft at the same CG location that was used for the flight tests to avoid the requirement of moment transfers between the wind tunnel and flight test data. The gimbal was designed and manufactured in house and is shown in Figure 9.1. It allows for 25 degree motion in pitch and roll and unlimited motion in yaw. Low friction ball bearings provide a common pivot for all three axes. Each axis is instrumented with a 12 bit absolute angle encoder that can be read by the UAV mainframe for additional attitude information. The encoders are frictionless and do not influence the motion in any way. This gimbal is the same as the one used for the inertia swing tests in the previous chapter. There a different centre-piece was used to provide the pendulum pivot instead of the mount to the wind tunnel mast.

![3 axis motion gimbal](image)

Figure 9.1: 3 axis motion gimbal

The test aircraft is a fully instrumented and operational model aircraft, identical to the flight test airframe except no landing gear and propulsion system was installed on the wind tunnel model. The aircraft is controlled with a standard radio control transmitter, allowing to set the trim or to manually ‘fly’ the aircraft inside the wind tunnel using the normal control surfaces. The gimbal is mounted inside the aircraft through a large cut-out at the centre of the wing to allow for sufficient rotation of the airframe
without impacting on the gimbal. This large cut-out required the replacement of the main spar of the wing with a structure to transfer the loads around the opening. This structure was made from 6mm plywood and integrated into the wing structure. Since this reinforcement is very close to the CG of the airframe it does not have an impact on the inertial properties, despite being quite heavy duty to take the required loads. The airframe was then balanced with lead balls to have the same CG position and inertial properties as the flight aircraft. Figure 9.2 shows the gimbal installed inside the aircraft, as well as the instrumentation and the battery to supply the power to the avionics and the flight controls.

Figure 9.2: The UAV mainframe installed in the wind tunnel test airframe. The 3-axis gimbal can be seen at the centre below the IMU.

The instrumentation of the aircraft is a exact copy of the UAV mainframe used in flight, with an additional sensor card for the gimbal position encoders. It communicates via a WIFI network with the groundstation laptop to control and observe the experiments. All data is recorded onboard and downloaded after or during the wind tunnel run via the network. This avoids any time delays due to the wireless connection [119].

9.2.2 Data Acquisition

For the data recording sessions, the wind tunnel was set to the nominal flight condition of $V_{\text{air}} = 20 \text{ m/s}$. The flight controller were configured for a trim condition at zero angle of attack and sideslip. When a pre-defined input sequence was initiated from the ground station, the flight controllers in the axis of the input (longitudinal or lateral) would disengage to execute the open loop input sequence. Due to the high turbulence in the wind tunnel and the issue of limited roll stability, as discussed below, it was required to
keep the flight controllers of the other axis engaged to keep the aircraft in trim about that axis. After the input sequence was executed and a programmed hold time to let the input response die out was over, the flight controllers would fully re-engage in all axes to return the aircraft to the trim condition. Particularly for lateral inputs it was required to re-trim each time before another input sequence could be started. The WIFI network, together with the Linux operating system and the real time input re-shaping capability of the UAVmainframe, allowed to modify the input sequence while the aircraft was flying in the tunnel. Over the network the definition file for the input sequence could be edited on the harddisk of the UAVmainframe and then the required direction and timing of the input could be set from the ground station. The recorded data could also be downloaded from the aircraft while flying to quickly inspect the recorded data of an input sequence. This allowed for rapid exploration and comparison of different input designs. Once the input sequence design was finalised, it was run repeatedly at least 10 times in a single recording session to generate the data for the repeatability tests discussed later in the chapter.

9.2.3 Roll Axis Stability Issue

The reduced set of equations of motion for the wind tunnel gimbal causes one issue that requires some work to rectify. Aircraft in flight obtain their roll stability from a restoring moment due to side-slip, the $C_{l\beta}$ effect. If the aircraft is disturbed in bank, it will start to turn and this creates a sideslip. The restoring rolling moment due to that sideslip rolls the aircraft back into level flight. It is therefore stable in the roll axis (assuming that $C_{l\beta}$ has the correct value). On the wind tunnel gimbal, if the aircraft banks there is no turn because there is no net force on the aircraft. Hence, no sideslip is generated and no restoring rolling moment exists. This makes the aircraft neutrally stable in roll. This can be easily tested in the wind tunnel. If the aircraft is set up in trim and then grabbed by hand, one can clearly feel the resistance in pitch and yaw. In roll the aircraft rotates freely and can be set to any bank angle, where it will stay. This neutral stability, together with the inevitability remaining small weight imbalance in the roll axes and the high turbulence levels in the tunnel cause the aircraft to drift in roll until it hits a stop on the gimbal. This motion is very slow, but still results in an uncontrollable trim attitude unless active control is used for the roll axis. The UAVmainframe provides this as discussed in Section 4.3.

9.2.4 Result Reporting

The results of the parameter ID of the dynamic wind tunnel data (and from the flight tests) will be reported in a standardised format, presented in this section. This removes the need to repeatedly explain all plots and tables in great detail.

Figure 9.3 shows an example of a single manoeuvre result plot. This particular one is illustrating the longitudinal short period response. At the top, the left plot shows the measured dynamic pressure from the airdata probe installed on the aircraft. Therefore some dependency of the measurement on the motion of the aircraft can be expected, since the probe shows small variations with inflow angle (see Section 6.4). The top
right plot gives the frequency response of the control input and of the response of the states. These spectra are normalised to be of similar magnitude. This allows for better comparison. The middle rows (the number depends on the number of model states) each show a measured state and the model fit on the left and the residual, that is the difference between data and model, on the right. The scale of the residual axes is kept similar in magnitude to easily compare the quality of the model fit. The bottom row shows the input time series on the left, which usually contains the measured control surface angle, but can include other measured states if the model structures requires it. The bottom right shows the other axis motion, here the lateral motion for a longitudinal manoeuvre. This allows to inspect the motion caused by the flight controllers stabilising that axis during the input. If that motion becomes large, possibly during a wind gust
Table 9.1: Single manoeuvre result table example (short period mode shown)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m\alpha}$</td>
<td>-0.948</td>
<td>-0.949</td>
<td>± 0.010 (0.003)</td>
<td>1.09 (0.28)</td>
<td>[-0.970 -0.929]</td>
</tr>
<tr>
<td>$C_{mq}'$</td>
<td>-11.492</td>
<td>± 0.276 (0.071)</td>
<td>2.40 (0.62)</td>
<td>[-12.045 -10.940]</td>
<td></td>
</tr>
<tr>
<td>$C_{m\delta e}$</td>
<td>-1.143</td>
<td>-1.110</td>
<td>± 0.021 (0.006)</td>
<td>1.90 (0.51)</td>
<td>[-1.153 -1.068]</td>
</tr>
</tbody>
</table>

$\omega_n = 1.47$ Hz, $\zeta = 0.41$

R2 for Output 1: 99.49 , R2 for Output 2: 99.48

or from other causes, there may be cross-coupling between the longitudinal and lateral axes and the model structures will not be valid. These manoeuvres can then be sorted out. The motion of the other axis is also a quick indicator of how turbulent the air is during that particular input.

Table 9.1 lists an example result of parameter estimates (this one corresponding to Figure 9.3). Listed are the estimated results for each model parameter together with some statistical information and, if available, the reference test result for that parameter. The columns with the standard deviations contain the original uncertainty estimates reported by the identification algorithms in brackets and the results after the correction for the coloured residuals [22]. This allows to easily identify the order of magnitude of growth in the parameter estimation uncertainty due to the correction. The bottom rows of the Table list some properties of the identified mode, as well as the $R^2$ value of the model fit.

If a series of manoeuvres is analysed sequentially, a plot like Figure 9.4 is generated. It shows the estimated parameter value for each manoeuvre together with the uncertainty as errorbars. Since three different algorithms (equation error (EQN), output error (OEM) and filter error (FEM) are used in parallel, there are three markers slightly offset to show the result of each method for this particular input. The vertical bars at the left and right of the figure give the colours corresponding to each method as well as the standard deviation of the spread of the series of results as the size of the bar. The mean value for each parameter is indicated as a horizontal line for each method in its colour. In the example, the mean values are very similar, but this is not always the case. As a general note, one would typically expect in a plot like this that the uncertainty of the estimate of each value would span a confidence band in which all estimates are located. This is rarely the case for the wind tunnel data, which indicates that despite of the correction for the residual colouring, the reported uncertainties for the parameter estimates from the algorithms are to small.

Table 9.2, which corresponds to Figure 9.4, lists the mean results for each parameter from the repeated runs, together with the 95% confidence interval, for the three identification methods. Where available, the expected value will also be given, as well as the mean values for the dynamic properties of the identified mode. In this example, the results of the three methods are fairly similar, but his is not always the case. There
Figure 9.4: Short period mode parameter estimation results for a single, repeated input (corresponding to Figure 10.3)

have been quite a few surprises about the capabilities of some of the methods during the course of this project, which will be discussed in the upcoming chapters.

Table 9.2: Short period mode parameter estimation results for a single, repeated input

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean ±2σ</th>
<th>OEM Mean ±2σ</th>
<th>FEM Mean ±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m_a}$</td>
<td></td>
<td>-0.948 ± 0.098</td>
<td>-0.886 ± 0.104</td>
<td>-0.889 ± 0.098</td>
</tr>
<tr>
<td>$C'_{m_a}$</td>
<td>-</td>
<td>-11.474 ± 2.77</td>
<td>-11.696 ± 1.75</td>
<td>-11.523 ± 3.04</td>
</tr>
<tr>
<td>$C_{m_{dc}}$</td>
<td>-1.143</td>
<td>-1.043 ± 0.158</td>
<td>-1.060 ± 0.166</td>
<td>-1.047 ± 0.166</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>1.455 ± 0.086</td>
<td>1.454 ± 0.092</td>
<td>1.456 ± 0.088</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.421 ± 0.1</td>
<td>0.429 ± 0.068</td>
<td>0.422 ± 0.108</td>
</tr>
</tbody>
</table>
10. Short Period Mode

The short period mode is the only longitudinal mode of motion of the 3DoF dynamic wind tunnel experiment. In this chapter the model structure and the input design for the elevator will be discussed and the results of the parameter ID process will be presented. Since this is the first 'real' data processing step, more details about the process will be discussed, whereas for the upcoming lateral modes the basic structure is simply repeated with only small modifications.

10.1 Model Structure

The standard model for the short period motion for the 6 DoF case can be found in references like [22] or [39]. The derivation of the 3 DoF version for the wind tunnel experiment from that model is very simple. One simply removes all of the force terms, which are reacted by the wind tunnel mount, from the 6 DoF model. This removes the $A_z$ equation and the $C_L$ components in the $\dot{\alpha}$ equation, as well as most of the $\dot{\alpha}$ terms in the pitching moment equation. The remaining model is fairly simple with only three parameters in the pitch equation. It is, however, not an approximation like in the 6 DoF case, but the full representation of the longitudinal equations of motion of the aircraft on the three axis gimbal. It is therefore expected that the model will be a good match to the test data, unless the assumption of the linear aerodynamic model or the negligible bearing friction turn out to be incorrect. To make the model suitable for the OE algorithm, each equation requires a bias parameter [22]. These were added to the $B$ matrix. The full model for the short period motion in 3 DoF is summarised below. Except for $C_m_{\alpha}$, all parameters are directly comparable to the static wind tunnel tests, which makes it easy to verify those and also to judge the correctness of the estimated value for $I_y$. 
Theorem 10.1.1 — Short Period Mode. The state space form of the short period mode on the wind tunnel gimbal is

\[
\begin{bmatrix}
\dot{\alpha} \\
\dot{q}
\end{bmatrix} =
\begin{bmatrix}
0 & \frac{\bar{q} Sc}{I_y} C_{m\alpha} & \frac{\bar{q} Sc^2}{2V_{air} I_y} C'_{mq} \\
0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
\alpha \\
q
\end{bmatrix} +
\begin{bmatrix}
0 & b_{\alpha} \\
\frac{\bar{q} Sc}{I_y} C_{m_{\delta_e}} & b_q
\end{bmatrix}
\begin{bmatrix}
\delta_c
\end{bmatrix}
\] (10.1a)

\[
\begin{bmatrix}
\alpha \\
q
\end{bmatrix} =
\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}
\begin{bmatrix}
\alpha \\
q
\end{bmatrix} +
\begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
\delta_c
\end{bmatrix}
\] (10.1b)

with potentially identifiable parameters \(C_{m\alpha}, C'_{mq}\) and \(C_{m_{\delta_e}}\) where \(C'_{mq} = C_{mq} + C_{m\dot{\alpha}}\)

The short period mode in 3 DoF has the properties

\[
\omega_{n,SP,wt} = \sqrt{-\bar{q} Sc/I_y \times C_{m\alpha}} \quad \text{and} \quad \zeta_{SP,wt} = -\frac{\bar{q} Sc/I_y \times C'_{mq} \times c/(2V_{air})}{2\omega_n}
\] (10.2)

10.2 Elevator Input Design

The elevator input used to excite the short period mode needs to put suitable energy into the pitching motion at or near the natural frequency of the mode to achieve a good signal to noise ratio and therefore a good identifiability of the parameters in question [22]. On the other hand, the pitch angle needs to be constrained between \(\pm 4 \text{deg}\) to stay well within the linear aerodynamic region of this aircraft. The easiest way to achieve this is a multistep input. From preliminary testing the expected natural frequency of the short period mode at \(V_{air} = 20 \text{m/s}\) is approximately \(1.5 \text{Hz}\). Given the guidelines in [22], a 3-2-1-1 multistep input was developed as shown in Figure 10.1(a) with the resulting power spectral density (PSD) in Figure 10.1(b). The figure also shows the actual elevator response to the commanded input which reveals the transfer function of the servo motor used. The difference between command and elevator response highlights the advantage of using a physical control surface feedback sensor.

The energy spectrum of the input is well distributed around the natural frequency at \(1.5 \text{Hz}\). The amplitudes of the input steps were shaped to achieve a mostly constant energy across the range. This was possible because the input is computer controlled and perfectly repeatable. The finite speed of the servo motor rounds of the edges of the command sequence but the PSD shows that this does not cause much of a difference in terms of frequency content. During the actual experiments, this input will be used in various forms, in order to eliminate any bias of the long step at the beginning. Using the computer control, the input can be inverted in time and direction, as well as tuned to a different frequency or size in real time from the groundstation. This enables rapid testing of various variations of this input sequence to obtain the best possible data from the experiment.
10.3 Verification

Figure 10.2 shows the recorded data and the fit of the identified model, corresponding to Eqs. (10.1). The output error method was used for this analysis. The pitch rate is taken from the IMU readings. Since it is a perturbation manoeuvre, the gyro bias does not have to be treated. The angle of attack can be obtained from the airvane or the pitch angle gimbal sensor, as well as from the IMU attitude estimate. Because of the turbulence, the airvane is noisy, so the gimbal pitch angle measurement was chosen, because it is a direct measurement of the state. Frequency domain smoothing was applied with a cut-off at $5\text{Hz}$. During the longitudinal input the lateral flight controller remain enabled to hold the wings level. The resulting motion in the lateral axis can be seen in Figure 10.2 at the lower right.

The resulting model fit is very good with $R^2$ values above 99% as listed in Table 10.1. Only small errors are visible in the residuals, caused by the turbulence in the test section. The estimates for $C_{m_{\alpha}}$ and $C_{m_{\delta_e}}$ are within 1% and 3% of the static test results, respectively. This is an initial confirmation that the model is indeed correctly describing the motion in the longitudinal axis in response to an elevator input. The excellent model fit also demonstrates that the UAVmainframe delivers high quality data from its sensors. In the next section, it will be investigated how well the parameter ID results can be repeated and what strategy of inputs will result in the best possible reference data for the longitudinal axis.
Figure 10.2: 3 DoF short period response to an elevator input

Table 10.1: Parameter ID results for a single elevator input

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m\alpha}$</td>
<td>-0.948</td>
<td>-0.949</td>
<td>± 0.010 (0.003)</td>
<td>1.09 (0.28)</td>
<td>[-0.970 -0.929]</td>
</tr>
<tr>
<td>$C_{m\delta}$</td>
<td>-11.492</td>
<td>-11.492</td>
<td>± 0.276 (0.071)</td>
<td>2.40 (0.62)</td>
<td>[-12.045 -10.940]</td>
</tr>
<tr>
<td>$C_{m\delta_e}$</td>
<td>-1.143</td>
<td>-1.110</td>
<td>± 0.021 (0.006)</td>
<td>1.90 (0.51)</td>
<td>[-1.153 -1.068]</td>
</tr>
</tbody>
</table>

$\omega_n = 1.47$ Hz, $\zeta = 0.41$

R2 for Output 1: 99.49 , R2 for Output 2: 99.48
10.4 Results

This section presents the results for the short period mode tests in the wind tunnel. The longitudinal axis is the only one relatively unaffected by the turbulence in the tunnel. Hence, high quality, repeatable results will minimal spread are expected. The short period mode is also the only mode, where the aircraft returns to the trim position by itself, allowing for rapid repeats of the inputs. These two properties will be used to demonstrate the accuracy achievable with this experimental method and also to show the data quality of the UAVmainframe. At the beginning, a single input is repeated several times and each is then run separately through the parameter estimation procedure. Next, several inputs are recorded in quick succession and a number of groups is then identified. Finally, a long series of inputs is identified as a whole, giving the algorithms a large amount of data to work with. All data was run through the equation error, output error and filter error methods, using identical models, to compare the results.

10.4.1 Repeated Single Inputs

The single inputs are precisely the ones described in the input design section above. As shown in Figure 10.3, the input was repeated five times and then mirrored in time and direction for another five times each. The Figure shows the times series of the in- and outputs, as recorded by the UAVmainframe. The angle of attack in this case is the data from the gimbal encoders to avoid the noisy data of the air vanes. The repeatability of the elevator input appears to be very good in magnitude, but shows some variation in the time axis. This is caused by the servo motor itself by having variable delays in

![Figure 10.3: Time series of repeated elevator inputs and aircraft response.](image-url)
responding to a command input. The feedback sensor on the elevator picks this up and shows the actual motion of the surface. The aircraft response repeatability is affected by this variation in the elevator timing, but this is no problem, because the corresponding input is known. Variation in magnitude of the response must probably be attributed to the turbulent conditions in the test section, causing variations in the dynamic pressure during the manoeuvre. That said, the variations in the input responses appear to be fairly small and it is expected that there will be only a small spread in the identified parameters. Figure 10.4 shows the frequency spectra (PSD) of the 20 elevator inputs in Figure 10.3. The variations in the timing of the inputs appears in the magnitudes of the PSD’s, showing small fluctuations in the energy levels across the frequency range. This plot confirms that all inputs excite the aircraft in a similar manner, leading to the good repeatability shown in Figure 10.3.

Figure 10.5 and Table 10.2 show the parameter identification results from the 20 inputs processed individually with the three methods. The first observation is that none of the four input shapes appears to produce consistently different results. The first

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean ±2σ</th>
<th>OEM Mean ±2σ</th>
<th>FEM Mean ±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m_a}$</td>
<td></td>
<td>-0.948</td>
<td>-0.887 ± 0.049</td>
<td>-0.886 ± 0.056</td>
</tr>
<tr>
<td>$C'_{m_q}$</td>
<td>-</td>
<td>-11.474 ± 1.39</td>
<td>-11.696 ± 0.88</td>
<td>-11.523 ± 1.52</td>
</tr>
<tr>
<td>$C_{m_{sc}}$</td>
<td></td>
<td>-1.143</td>
<td>-1.043 ± 0.079</td>
<td>-1.060 ± 0.084</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>1.455 ± 0.043</td>
<td>1.454 ± 0.046</td>
<td>1.456 ± 0.044</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.421 ± 0.05</td>
<td>0.429 ± 0.034</td>
<td>0.422 ± 0.054</td>
</tr>
</tbody>
</table>
five inputs show a slightly higher value for $C_{m_{\alpha}}$ but $C'_{m_q}$ and $C_{m_{\delta e}}$ are in line with the other three inputs. In general, all parameter values appear to be randomly spread. This confirms that the aircraft response is not affected by inverting the input sequence either in time or direction, as long as the frequency spectrum stays constant. The second observation is that none of the parameter ID algorithms has any advantage over the others for this data. In $C'_{m_q}$ the output error method has a slightly smaller standard deviation, but for the other two parameters all three methods show similar uncertainties. Since the model is essentially a single equation model and the process (turbulence) noise levels are low, this result is not surprising. All results, however, show significant spread, which is much larger than the estimation uncertainties indicated in the error bars. For $C_{m_{\alpha}}$ the confidence interval is about 10%, for $C'_{m_q}$ it is 20%-25% and for $C_{m_{\delta e}}$ the confidence interval is about 15%. This is a common problem in aircraft parameter ID [22, 38], but since all time domain uncertainties were treated for the coloured residuals with the method included in SIDPAC, this large spread is somewhat surprising. The final observation is that the estimate for $C_{m_{\alpha}}$ differs from the static test result by 6.4% and the estimate for $C_{m_{\delta e}}$ is off by 8%. This appears to be a somewhat large error, given the good fit of the model to the data.
Chapter 10. Short Period Mode

On the other hand inspecting the dynamic properties of the resulting system matrices, the natural frequency $\omega_n$ and damping ratio $\zeta$, it can be seen that there is very little difference and uncertainty on these values. It appears that the 3DoF short period model can exhibit similar properties for a large number of parameter combinations, which makes them difficult to observe during the parameter ID process. This may be an issue of insufficient information content in the response to a single input, or it may be a property of the model itself. To investigate this, the single input will now be replaced by a series of inputs to increase the information content in the data.

10.4.2 Repeated Series of Multiple Inputs

Figure 10.6 shows the input sequence that was used to increase the information content in the data. It consists of four separate inputs placed far enough apart to act as individual

![Figure 10.6: Extended elevator input sequence with aircraft response (The PSD in the upper right is affected by the four inputs not being exactly in phase. This is not a problem since each input is spaced to act as a single input in the time domain.)](image-url)
Figure 10.7: Short period mode parameter estimation results for multiple, repeated inputs (corresponding to Figure 10.6)

inputs. All four sequences have the same timing as the single input, but the magnitude and direction of some of the pulses was altered slightly to provide some variation. The new input sequence was run 10 times and the parameters of the system model were identified with the three methods as before. The model fit to the data in Figure 10.6 is excellent, with only small residuals for both states. These are caused by the turbulence levels as before. An interesting observation is the plot of the lateral motion in the Figure down right. It shows how hard the flight controllers have to work to hold the wings level in these flow conditions.

Table 10.3: Short period mode parameter estimation results for multiple, repeated inputs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{ma}$</td>
<td>-0.948</td>
<td>$-0.932 \pm 0.043$</td>
<td>$-0.936 \pm 0.037$</td>
<td>$-0.935 \pm 0.043$</td>
</tr>
<tr>
<td>$C'_{mq}$</td>
<td>-</td>
<td>$-11.695 \pm 1.41$</td>
<td>$-12.072 \pm 0.89$</td>
<td>$-11.693 \pm 1.57$</td>
</tr>
<tr>
<td>$C_{ms_e}$</td>
<td>-1.143</td>
<td>$-1.085 \pm 0.037$</td>
<td>$-1.110 \pm 0.047$</td>
<td>$-1.089 \pm 0.05$</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>$1.452 \pm 0.034$</td>
<td>$1.456 \pm 0.029$</td>
<td>$1.455 \pm 0.033$</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>$0.419 \pm 0.043$</td>
<td>$0.431 \pm 0.027$</td>
<td>$0.418 \pm 0.049$</td>
</tr>
</tbody>
</table>
The results of the parameter ID for the multiple input series are listed in Table 10.3 and plotted in Figure 10.7. Despite providing much more information content, the spread in the results has not changed significantly. The output error method uncertainties improved slightly, while the other two methods even deteriorated on some parameters. The mean values for $C_{m\alpha}$ and $C_{m\delta e}$, however, show a better fit to the static data now, so there is some benefit of the longer inputs. The system matrix properties are almost identical to the single input, reinforcing the theory of the low observability of the system parameters in this model. On the other hand, the constant system properties, together with better estimates for $C_{m\alpha}$ and $C_{m\delta e}$ should yield a better estimate for $C'_{mq}$, which is the main target of this experiment.

### 10.4.3 Single Long Series of Multiple Inputs

As a final step, now 10 input series of four were recorded in quick succession in a single file. This represents over two minutes or 12000 samples of data and should yield the best possible accuracy obtainable from this experiment. Figure 10.8 shows the result of such a run, modelled with the output error method. All residuals approach white noise, which is the ideal case for this method.

Tables 10.4 and 10.5 show the identified parameters for a single run using the OE and FEM method, respectively. As expected, these results are very good, with almost negligible uncertainties. The OE results are slightly closer to the static test results, but for both methods the remaining difference is well within the expected tolerance. The repeatability between runs, as shown in Figure 10.9 and Table 10.6, has also improved.

<table>
<thead>
<tr>
<th>Table 10.4: OE results for a single long series of elevator inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>$C_{m\alpha}$</td>
</tr>
<tr>
<td>$C'_{mq}$</td>
</tr>
<tr>
<td>$C_{m\delta e}$</td>
</tr>
</tbody>
</table>

$\omega_n = 1.46$ Hz, $\zeta = 0.43$

R2 for Output 1: 98.49, R2 for Output 2: 98.76

<table>
<thead>
<tr>
<th>Table 10.5: FEM results for a single long series of elevator inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>$C_{m\alpha}$</td>
</tr>
<tr>
<td>$C'_{mq}$</td>
</tr>
<tr>
<td>$C_{m\delta e}$</td>
</tr>
</tbody>
</table>

$\omega_n = 1.45$ Hz, $\zeta = 0.42$

R2 for Output 1: 100.00, R2 for Output 2: 99.96
Figure 10.8: A long series of elevator inputs, identified with the OE method

for all parameters, bringing down the uncertainty on all parameters by at least a factor of two. Most peculiarly, did the spread for the natural frequency increase by the same factor.

These long input sequences allowed the identification with the best possible precision in the trying conditions of the turbulent wind tunnel flow. The results match the static test data well, which confirms those results, as well as the measurements of the pitch inertia of the aircraft. In the next section, a brief attempt will be made to explain the observability issues of the 3DoF short period model, before the results of this chapter are summarised at the end.
Table 10.6: Parameter ID results for multiple long series of elevator inputs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m_a}$</td>
<td>-0.948</td>
<td>-0.924 ± 0.019</td>
<td>-0.930 ± 0.016</td>
<td>-0.925 ± 0.013</td>
</tr>
<tr>
<td>$C'_{m_q}$</td>
<td>-</td>
<td>-11.789 ± 0.294</td>
<td>-12.099 ± 0.193</td>
<td>-11.812 ± 0.355</td>
</tr>
<tr>
<td>$C_{m_s}$</td>
<td>-1.143</td>
<td>-1.087 ± 0.029</td>
<td>-1.113 ± 0.021</td>
<td>-1.091 ± 0.032</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>1.378 ± 0.059</td>
<td>1.383 ± 0.059</td>
<td>1.379 ± 0.061</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.425 ± 0.005</td>
<td>0.434 ± 0.004</td>
<td>0.425 ± 0.005</td>
</tr>
</tbody>
</table>

10.4.4 Notes on Short Period Model Observability

In order to investigate the observability issues of the 3DoF short period model, the first question to investigate is whether the parameter ID algorithms can actually estimate error free parameters from perfect, noise free data. To test this, a dataset was prepared by time time-integrating the model with the parameters set to the values in Table 10.7 (these were preliminary results during the experiment development). The resulting simulated response was then used to identify the three parameters with the OE algorithm.
result, shown in Figure 10.10 is a perfect fit, with zero error between the estimates and the true parameter values. Only a small error in the initial values exists because the algorithm would not run properly if the data and the model are exactly the same. Therefore the OE algorithm is capable of returning the true parameter values from perfect, noise free data.

The next step is to test the OE algorithm with a simulated response from a full, non-linear flight simulation, which was adapted to the 3DoF equations of motion. Since the 3DoF short period model is kinetically exact (no approximation as with the flight case), the only possible reasons for the identifiability issues are the simple, linear aerodynamic model of the applied moment $C_m$, or the properties of the model with the parameters for this particular aircraft itself, allowing for many different parameter combinations to yield similar properties. If the aerodynamic model was the cause, the fit of the model to the simulated data would show significant residuals, which would account for the

![Figure 10.10: Results of parameter ID using data from integrating the second order short period mode model](image)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m_{\alpha}}$</td>
<td>-0.977</td>
</tr>
<tr>
<td>$C_m^{\prime}$</td>
<td>-11.99</td>
</tr>
<tr>
<td>$C_{m_{\delta_e}}$</td>
<td>-1.17</td>
</tr>
</tbody>
</table>
Figure 10.11: Results of parameter ID using data from a 3 DoF flight simulation

un-modelled terms. Figure 10.11, however, shows an excellent fit of the identified model to the simulated data from the non-linear equations of motion. Yet, Table 10.8 lists differences of 3-4% between the parameter values used for the simulation and the estimated results. This points to the speculated insensitivity in the model and can be explained as follows:

The cost function, which is minimised during the parameter estimation process (or the least square fit for the equation error method) must have a shallow minimum as depicted in Figure 10.12, at least for this particular aircraft. The Figure shows a sketch of the cost function, reduced to a single dimension for illustration purposes, with a shallow valley and several local minima. These correspond to the varying solutions for the parameters that were seen throughout this chapter. The algorithms find the large valley, where the cost function is small, but fail to identify the correct local minimum, since there is no strong differentiation between them in terms of model fit and properties.

Table 10.8: Parameter estimates from simulated data (corresponding to Figure 10.11)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>True value</th>
<th>Identified value</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m_\alpha}$</td>
<td>-0.948</td>
<td>$-0.977 \pm 0.007$</td>
<td>3%</td>
</tr>
<tr>
<td>$C'_{m_q}$</td>
<td>-12.5</td>
<td>$-11.99 \pm 0.211$</td>
<td>4%</td>
</tr>
<tr>
<td>$C_{m_\delta_e}$</td>
<td>-1.143</td>
<td>$-1.17 \pm 0.017$</td>
<td>2.6%</td>
</tr>
</tbody>
</table>
Hence one algorithm might settle for a different local minimum than the others, giving slightly different parameter estimates, but similar system properties. It may require a different parametrisation of the model or a completely different approach to improve this. For this project, however, the accuracy of the estimated parameters of the 3DoF short period motion was acceptable and no further attempt in improving the results were made.

10.4.5 Input Sensitivity Tests

During flight testing of a remotely piloted aircraft, the airspeed cannot be expected to be kept exactly at the value that was used to design the elevator input shape. Therefore the frequency content of that input sequence will potentially be different from the ideal spectrum, since this depends on the dynamic pressure. So the question is: How sensitive is the short period response to the frequency content of the elevator input?

This question can be easily answered using the dynamic wind tunnel test capabilities. The computer controlled input generator of the UAV mainframe allows easy modifications of the inputs. The size can be controlled in terms of percentage of the nominal input and the frequency can be changed similarly. When changing the frequency, the computer attempts to re-size the input such that the total energy stays constant.

A series of inputs with a natural frequency range of 0.75-3 Hz is plotted in Figure 10.13(a). The corresponding PSDs in Figure 10.13(b) show the changing spectra of the inputs. The area under these curves is the total energy, which is held close to constant. The Figure shows every second input of the series of 11 for clarity. Figure 10.13(c) then shows the parameter estimate for $C_{m\alpha}$ for each input as an example.

There appears to be no major pattern of errors or any other feature linking the input frequency to the quality of the parameter estimation. This is rather surprising, especially since the slowest inputs virtually do not contain any frequencies at the natural frequency.
Figure 10.13: Elevator inputs (frequency range 0.75-3 Hz) with PSD and parameter estimates
of the mode or above. Given this data, the input shape does not seem to matter at all.

Similarly, a sequence of inputs with constant shape but different sizes as shown in Figure 10.14(a) does not seem to have any influence on the results of the estimates for $C_{\text{mq}}'''$ in Figure 10.14(b). Even the smallest input still yields a usable result.

Concluding, the 3-2-1-1 input used for exciting the short period mode is well suited for the task in terms of frequency range. Further analysis showed that, counter intuitively, the input shape and size does not matter much and any input within the tested range would yield usable parameter estimates. This relaxes the accuracy required during the flight tests, if this finding holds for the 6DoF motion as well.
10.5 Variation of CG Location

A final run was made with a aft CG location to obtain some data on the changes to the longitudinal dynamics due to the reduced stability. The gimbal was mounted 25 mm aft of the nominal CG location, which approximately represents a reduction in static margin by half. The pitch stiffness derivative $C_{m_\alpha}$ is directly proportional to the static margin and therefore a value of -0.474 is expected. The aircraft was then re-trimmed by moving the main battery and some other weights to achieve a CG location coincident with the gimbal pivot. Due to the re-trim the longitudinal inertia $I_y$ is reduced by an unknown amount. Since this is just an exploratory test, the pitch inertia has not been re-measured for the new weight distribution. Instead, the estimate for $C_{m_\alpha}$ can be used to correct for the reduced inertia by comparing the expected result with the estimate and adjusting $I_y$ until good agreement has been achieved. This is possible since the value for $C_{m_\alpha}$ is estimated with high confidence, which was demonstrated before. Table 10.9 lists the results for the aft CG parameter estimates. As expected from discussions in reference [39], the natural frequency of the short period mode reduces and the damping increases. The value for $C'_{m_q}$ is slightly smaller than for the nominal CG location and the control derivative $C_{m_\delta e}$ is also smaller due to the shorter moment arm of the elevator. These results will become useful when interpreting the flight test results later on.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{m_\alpha}$</td>
<td>-0.474</td>
<td>-0.476</td>
<td>$\pm 0.004$ (0.001)</td>
<td>0.74 (0.12)</td>
<td>$[-0.483, -0.469]$</td>
</tr>
<tr>
<td>$C'_{m_q}$</td>
<td>-11.288</td>
<td>$\pm 0.146$ (0.024)</td>
<td>1.29 (0.21)</td>
<td>$[-11.579, -10.996]$</td>
<td></td>
</tr>
<tr>
<td>$C_{m_\delta e}$</td>
<td>-1.031</td>
<td>$\pm 0.011$ (0.002)</td>
<td>1.10 (0.18)</td>
<td>$[-1.053, -1.008]$</td>
<td></td>
</tr>
</tbody>
</table>

$\omega_n = 1.08$ Hz, $\zeta = 0.6$

R2 for Output 1: 97.23, R2 for Output 2: 97.61
10.6 Short Period Mode Summary

Table 10.10 lists the final results for the longitudinal dynamic tests. The results of the output error method from Table 10.6 were chosen, because they have the smallest overall uncertainties of the three methods. Yet, all three methods delivered results that are well within the accuracy expectations for this experiment under the turbulent conditions in the wind tunnel. On the other hand, none of the methods seems to have a clear advantage. Especially the filter error method performs on par with the output error method and does not provide any improvements for this experiment.

Table 10.10: Result summary for the short period mode from dynamic wind tunnel tests

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value±2σ</th>
<th>% Uncert.</th>
<th>Static Test</th>
<th>% Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cmα</td>
<td>−0.930 ± 0.032</td>
<td>3.4%</td>
<td>−0.948</td>
<td>1.8%</td>
</tr>
<tr>
<td>C'mq</td>
<td>−12.099 ± 0.386</td>
<td>3.1%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>C'mδe</td>
<td>−1.113 ± 0.042</td>
<td>3.7%</td>
<td>−1.143</td>
<td>2.6%</td>
</tr>
</tbody>
</table>

Using the estimate for δε/δα from appendix E, the result for C'mq can be used to calculate Cṁ and Cmα, as follows (see appendix E):

\[
C_{mq} = \frac{C'_{m_{qs}}}{(1 + \delta \epsilon/\delta \alpha)} = \frac{-12.099}{(1 + 0.481)} = -8.17
\]

and

\[
C_{mα} = C_{mq} \frac{\delta \epsilon}{\delta \alpha} = -8.17 \times 0.481 = -3.93
\]
The next mode of motion tested was the dutch roll mode. The aircraft was excited by a rudder doublet to start the motion. The second order approximation for the dutch roll, which consists of the yawing moment and sideslip equation was used to identify the parameters of the yawing moment equation.

11.1 Model Structure

The model for the dutch roll motion on the wind tunnel gimbal can be developed from the 6 DoF model in a similar manner to the previous short period model. In this case the side force is reacted by the gimbal and therefore all force terms drop out of the $\dot{\beta}$ equation, together with the entire $a_y$ output equation. The yawing moment due to roll rate parameter $C_{n_p}$ was inserted into the $\dot{r}$ equation as a pseudo input, using the measured roll rate $p$. This may potentially allow to identify this parameter as well, which would complete the yawing moment equation. Similarly to the short period model, bias terms were added to the model as required for a linear model.
Chapter 11. Dutch Roll Approximation

Theorem 11.1.1 — Dutch Roll Mode. The state space form of the dutch roll mode approximation on the wind tunnel gimbal is

\[
\begin{bmatrix}
\dot{\beta} \\
\dot{r}
\end{bmatrix} =
\begin{bmatrix}
0 & -1 \\
k_1 C_{n\beta} & k_1 k_2 C_{n\gamma}
\end{bmatrix}
\begin{bmatrix}
\beta \\
r
\end{bmatrix} +
\begin{bmatrix}
0 & 0 & b_{\beta} \\
k_1 C_{n\gamma} & k_1 k_2 C_{n\phi} & b_r
\end{bmatrix}
\begin{bmatrix}
\delta_r \\
p_m \\
1
\end{bmatrix}
\tag{11.1a}
\]

\[
\begin{bmatrix}
\beta \\
r
\end{bmatrix} =
\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}
\begin{bmatrix}
\beta \\
r
\end{bmatrix} +
\begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}
\delta_r
\tag{11.1b}
\]

with

\[
k_1 = \frac{q S b}{I_{zz}} \quad k_2 = \frac{b}{2V_{air}}
\]

and the potentially identifiable derivatives

\[
C_{n\beta}, C_{n\phi}, C_{n\gamma} \text{ and } C_{n\delta}
\]

where \(C_{n\beta}\) and \(C_{n\delta}\) should match the static test data. The 3 DoF dutch roll approximation has the properties:

\[
\omega_{n,DR,wt} = \sqrt{k_1 C_{n\beta}} \quad \text{and} \quad \zeta_{DR,wt} = \frac{k_1 k_2 C_{n\gamma}}{2\omega_n}
\tag{11.2}
\]

11.2 Input Design and Verification

The rudder input that was used for the dutch roll experiments is shown in Figure 11.1, together with an example response and the resulting model fit. The input is a four step input with identical timing and sizes, which resembles two repeats of a normal doublet input. In the turbulent conditions in the wind tunnel, this yields a better result because the aircraft moves in direct response to the rudder motion for longer. With this strategy, the information content dominated by the input sequence is increased before the free, lightly damped dutch roll oscillation starts. The light damping (\(\zeta \approx 0.1\) for this aircraft) requires a long hold after the input to identify the motion parameters correctly. During this hold the turbulence causes significant disturbances, which can be seen by the increasing residuals in the Figure. The frequency spectrum of the input and response is also shown in the Figure, centred nicely at 0.8Hz. Due to the tolerances of the servo motor, the two doublets are not fully identical and the input spectrum becomes slightly noisy.

It turns out that \(C_{n\phi}\) and \(C_{n\gamma}\) are highly correlated (> 90%) for this aircraft and input sequence. This makes it impossible to identify both at the same time. When doing so, the result for \(C_{n\gamma}\) degrades. Attempts to find a more suitable input sequence were not successful. Identifying \(C_{n\beta}\) with \(C_{n\phi}\) set to zero and turned off and then holding \(C_{n\beta}\) at the identified value while trying to identify \(C_{n\phi}\) in a second run were also not successful. Therefore \(C_{n\phi}\) was set to zero and the estimation of this parameter was turned off for the rest of this experiment.
The data for the dutch roll identification comes from the yaw gyro and the gimbal yaw angle encoder. The IMU yaw estimate critically depends on its magnetometer readings, which are severely disturbed in the wind tunnel test section with its steel floor and frame. Therefore the heading solution of the IMU is not usable and the gimbal position encoder is the data source of choice. The $\beta$ vane is equally noisy as the $\alpha$ vane, which would lead to degradation of the results. The pitch angle controller remain active during lateral inputs and the resulting motion can be seen in the Figure. The dynamic pressure measurement of the airdata probe, as depicted in the Figure at the top left, shows more sensitivity to yawing motion than to pitching motion (see Figure 10.2). The variation in dynamic pressure equals $0.5 \text{ m/s}$ as compared to $0.25 \text{ m/s}$ in pitch. Since all models used

![Figure 11.1: Aircraft response to a rudder input](image-url)
Table 11.1: Sample results for the dutch roll parameters from the output error method

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n\beta}$</td>
<td>0.087</td>
<td>0.088 ± 0.004 (0.000)</td>
<td>4.83 (0.46)</td>
<td>[0.080 0.097]</td>
<td></td>
</tr>
<tr>
<td>$C_{nc}$</td>
<td>-</td>
<td>-0.075 ± 0.003 (0.000)</td>
<td>4.65 (0.57)</td>
<td>[-0.082 -0.068]</td>
<td></td>
</tr>
<tr>
<td>$C_{nr\delta}$</td>
<td>-0.063</td>
<td>-0.066 ± 0.002 (0.000)</td>
<td>3.76 (0.46)</td>
<td>[-0.071 -0.061]</td>
<td></td>
</tr>
</tbody>
</table>

$\omega_n = 0.81$ Hz, $\zeta = 0.09$

R2 for Output 1: 98.54, R2 for Output 2: 98.50

for this thesis use the measured dynamic pressure for each individual sample and not the mean value. This larger variation of dynamic pressure during the lateral inputs is of no consequence and can be ignored.

Table 11.1 lists the results of the parameter identification of the aircraft response from Figure 11.1, as determined with the output error method. The results show excellent agreement with the static test results and all uncertainties are below 5%. The damping ratio is very low with $\zeta = 0.09$, resulting in a slowly diminishing oscillation in the yaw axis that would be quite uncomfortable for a manned aircraft. For a remotely piloted aircraft this results in constant swaying in yaw, making the flight path quite unsteady. The accuracy of the model fit to the measured data as well as the agreement of the identified parameters with the static test results indicate that the dutch roll model is a good description of the motion of the aircraft in response to a rudder input. Similarly to the short period model, in the next section some series of inputs will be processed with the three methods to generate the reference values for the yaw axis parameters except $C_{n_p}$, which will be estimated in a later chapter using inputs with combined rudder and aileron motion.

11.3 Results

To generate reliable reference data for the yaw axis under the turbulent conditions, it is necessary to repeat the experiment often. As for the short period mode, it is better to identify the parameters from a long dataset with multiple inputs than to run each input separately. Nevertheless, to get an idea what kind of accuracy single inputs will achieve, this is done first. During flight testing there will only be a single rudder input per dataset, so it is a good idea to gain some experience on what to expect in this case.

11.3.1 Repeated Single Inputs

Ten separate rudder inputs were used for this section. The results of the parameter ID, using the three methods as before, are listed in Table 11.2 and are plotted in Figure 11.2. Even though the mean value for $C_{n\beta}$ and $C_{nr\delta}$ are very close to the results from the static tests, the uncertainties are quite large. Especially the control derivative confidence interval is large with $\pm 25\%$. $C_{n\beta}$ appears to be predicted fairly reliably, with the error bars (or confidence interval) on each estimate being larger than the spread between the results, at least for the OE and FEM methods. The yaw damping derivative $C_{nr\delta}$ also has a large confidence interval, and the results from the equation error and filter error methods are markedly different from the output error results (-0.067 vs. -0.079). The
### Table 11.2: Parameter ID results for the dutch roll mode from 10 separate inputs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean $\pm 2\sigma$</th>
<th>OEM Mean $\pm 2\sigma$</th>
<th>FEM Mean $\pm 2\sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n,3}$</td>
<td>0.087</td>
<td>0.089 $\pm$ 0.003</td>
<td>0.089 $\pm$ 0.003</td>
<td>0.087 $\pm$ 0.003</td>
</tr>
<tr>
<td>$C_{n,r}$</td>
<td>-</td>
<td>$-0.067 \pm 0.013$</td>
<td>$-0.079 \pm 0.013$</td>
<td>$-0.067 \pm 0.010$</td>
</tr>
<tr>
<td>$C_{n,\delta r}$</td>
<td>-0.063</td>
<td>$-0.064 \pm 0.008$</td>
<td>$-0.064 \pm 0.008$</td>
<td>$-0.063 \pm 0.005$</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>0.818 $\pm$ 0.016</td>
<td>0.809 $\pm$ 0.000</td>
<td>0.809 $\pm$ 0.000</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.078 $\pm$ 0.014</td>
<td>0.079 $\pm$ 0.000</td>
<td>0.079 $\pm$ 0.000</td>
</tr>
</tbody>
</table>

Observability of this parameter is limited by the very low damping in yaw of this aircraft. Since the motion is so lightly damped, it takes a long time to die out. During this time all lateral controls are held fixed and therefore the motion is vulnerable to the turbulence in the wind tunnel, as seen in the residuals in Figure 11.1. Also, the equation for the damping ratio (Eq. (11.2)) is not very sensitive to the value of $C_{n,r}$ for light damping. These two issues cause difficulties for the parameter ID algorithms to estimate this value precisely. The example of the shallow cost function of the short period mode also applies here. Similarly to the longitudinal case, a sequence with multiple inputs will increase the information content and hopefully reduce the uncertainty of the results. For the flight data, however, the result for $C_{n,r}$ will most likely retain a large uncertainty.

Figure 11.2: Parameter ID results for the dutch roll mode from 10 individual inputs (Figure key: Section 9.2.4)
11.3.2 Sequence of Multiple Inputs

For the long sequence of inputs the same file as for the previous separate inputs was used. But instead of cutting it down into separate files, it was run through the parameter ID as a single dataset. The results of the output- and filter error methods are listed in Tables 11.3 and 11.4. The data is plotted in Figure 11.3 for the filter error method, which can be identified by the virtually zero residuals. As before, the estimates for $C_{n,\beta}$ and $C_{n,\delta_r}$ from both methods match the static test data very well, and the uncertainties have reduced considerably. The values for $C_{n,\alpha}$ are still quite different (-0.077 vs. -0.066) between the methods but the confidence interval has reduce from 25% to below 5%. An interesting detail of this comparison between output- and filter error method is that while...

Figure 11.3: Aircraft response to a repeated rudder input, with a model fit from FEM
Table 11.3: OE results for a sequence of ten rudder inputs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n,\beta}$</td>
<td>0.087</td>
<td>0.089</td>
<td>± 0.001 (0.000)</td>
<td>1.43 (0.10)</td>
<td>[0.086, 0.091]</td>
</tr>
<tr>
<td>$C_{n,r}$</td>
<td>-</td>
<td>-0.077</td>
<td>± 0.002 (0.000)</td>
<td>2.32 (0.24)</td>
<td>[-0.081, -0.074]</td>
</tr>
<tr>
<td>$C_{n,\delta r}$</td>
<td>-0.063</td>
<td>-0.063</td>
<td>± 0.001 (0.000)</td>
<td>1.58 (0.19)</td>
<td>[-0.065, -0.061]</td>
</tr>
</tbody>
</table>

$\omega_n = 0.82$ Hz, $\zeta = 0.09$

R2 for Output 1: 97.12, R2 for Output 2: 97.05

Table 11.4: FEM results for a sequence of ten rudder inputs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n,\beta}$</td>
<td>0.087</td>
<td>0.087</td>
<td>± 0.001 (0.000)</td>
<td>1.64 (0.12)</td>
<td>[0.084, 0.090]</td>
</tr>
<tr>
<td>$C_{n,r}$</td>
<td>-</td>
<td>-0.066</td>
<td>± 0.003 (0.000)</td>
<td>4.85 (0.28)</td>
<td>[-0.072, -0.059]</td>
</tr>
<tr>
<td>$C_{n,\delta r}$</td>
<td>-0.063</td>
<td>-0.063</td>
<td>± 0.002 (0.000)</td>
<td>3.08 (0.22)</td>
<td>[-0.067, -0.059]</td>
</tr>
</tbody>
</table>

$\omega_n = 0.81$ Hz, $\zeta = 0.08$

R2 for Output 1: 100.00, R2 for Output 2: 100.00

The filter error matches the data exactly, its uncertainties of the parameter estimates are actually larger than for the output error method. This deceptive behaviour has been reported in various publications [74]. It is therefore a good idea to always use the two methods in combination and then carefully select the best results.

11.3.3 Repeated Series of Inputs

As a final step, five long sequences of ten inputs each were run through the three methods. The results are plotted in Figure 11.4 and listed in Table 11.5. The results are similar to the previous cases, with $C_{n,\beta}$ and $C_{n,\delta r}$ matching the static test data well, although the mean value for $C_{n,\beta}$ is slightly higher than before. The result for $C_{n,r}$ is again different for the output error method. Similarly to the short period mode, this test seems to show the limitations of the identifiability of the dutch roll model in turbulent conditions.
Table 11.5: Results of multiple sequences with ten rudder inputs each

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>0.087</td>
<td>0.089 ± 0.002</td>
<td>0.090 ± 0.002</td>
<td>0.088 ± 0.002</td>
</tr>
<tr>
<td>$C_{n_{r}}$</td>
<td>-</td>
<td>-0.064 ± 0.005</td>
<td>-0.076 ± 0.004</td>
<td>-0.064 ± 0.003</td>
</tr>
<tr>
<td>$C_{n_{\delta r}}$</td>
<td>-0.063</td>
<td>-0.063 ± 0.004</td>
<td>-0.063 ± 0.003</td>
<td>-0.064 ± 0.004</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>0.822 ± 0.007</td>
<td>0.823 ± 0.000</td>
<td>0.823 ± 0.000</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.075 ± 0.006</td>
<td>0.079 ± 0.000</td>
<td>0.079 ± 0.000</td>
</tr>
</tbody>
</table>

Figure 11.4: Results of multiple sequences with ten rudder inputs each
11.4 Dutch Roll Mode Summary

The final results for the 3 DoF dutch roll mode are listed in Table 11.6. The results for the dynamic tests are the average values from Table 11.5, as those represent the best possible estimates for this mode of motion. The results of the static derivatives $C_{n\beta}$ and $C_{n\delta r}$ match the estimates from the dutch roll motion very well. This confirms the model structure and the estimate for $I_Z$. The result for the yaw damping derivative shows a larger uncertainty due to the low damping and model sensitivity. The final uncertainty of $C_{n\delta r}$ is also quite large, which is probably caused by the low rudder effectiveness due to its small size, together with the turbulent conditions in the wind tunnel. The dutch roll mode of this aircraft is very lightly damped, which normally does not present a significant issue for a remotely piloted aircraft, as it is hardly noticeable from the ground. But for parameter ID flight testing it requires an active yaw damper to stabilise the dutch roll mode for successful data collection.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value±2σ</th>
<th>% Uncert.</th>
<th>Static Test</th>
<th>% Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n\beta}$</td>
<td>0.089 ± 0.004</td>
<td>4.49%</td>
<td>0.087</td>
<td>2.3%</td>
</tr>
<tr>
<td>$C_{n\gamma}$</td>
<td>-0.068 ± 0.01</td>
<td>15.6%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{n\delta r}$</td>
<td>-0.063 ± 0.008</td>
<td>12.7%</td>
<td>-0.063</td>
<td>-</td>
</tr>
</tbody>
</table>
The roll mode is an exponentially converging, non-oscillatory mode, which means it is characterized by a single transition between two steady states. This behavior is much harder to characterize in a turbulent environment, where the airframe is rocking in roll due to uneven flow over the wings, since there is no repeated, oscillatory motion where the system ID algorithm can attempt to determine the correct frequency and damping over a longer dataset. Figure 12.1 illustrates the issue, showing a simulated response and a measured response to the same aileron input. The figure shows heavy disturbances in the wind tunnel response, especially large overshoots at the end of each transition, which would indicate an underdamped, oscillatory mode. This result would be wrong, of course. So the main challenge in this chapter will be to construct an input sequence that minimizes the influence of the wind tunnel turbulence to obtain sensible results for the roll mode parameters. In any case, the results from this experiment are not expected to be as accurate as for the pitch and yaw axes.

12.1 Model Structure

The roll motion on the wind tunnel gimbal can be modeled using the full roll rate equation, which takes into account the secondary adverse jaw effect and the additional rolling motion due to excitation of the dutch roll mode. The roll rate equation is

\[
\dot{p} = \frac{q S b}{I_x} \left( C_{1\beta} \dot{\beta} + C_{1p} \frac{pb}{2V_0} + C_{1r} \frac{rb}{2V_0} + C_{1\delta a} \delta a + C_{1\delta r} \delta r \right)
\]

(12.1)

if the small cross inertia \(I_{xz}\) is ignored. In order to use this single equation to estimate the four parameters with the parameter ID algorithms, it has to be written in state space form. With single state \(p\), the contributions of the sideslip and yaw rate are modeled as pseudo inputs, using measured data, together with the aileron deflection. The contribution of
the rudder to the rolling moment is negligible and can be ignored for this aircraft. Since this is an aileron only manoeuvre, there won’t be any rudder deflection anyway and this term can be deleted from the model. The resulting model, including the bias parameter becomes:

**Theorem 12.1.1 — Roll Mode.** The state space form of the full roll mode equation on the wind tunnel gimbal is

\[
\dot{\phi} = [k_1 k_2 C_{l_p}] \phi + [k_1 C_{l_{\delta a}} \delta_a + k_1 C_{l_\beta} \beta + k_1 k_2 C_{l_r} r + b_p]
\]

(12.2)

with

\[
k_1 = \frac{\bar{\alpha} S b}{T_{xx}} \quad k_2 = \frac{b}{2V_{air}}
\]

and the potentially identifiable derivatives

\[
C_{l_p}, C_{l_{\delta a}}, C_{l_\beta} \text{ and } C_{l_{\delta a}}
\]

where \(C_{l_p}\) is related to the time constant \(\tau\) of the roll mode by

\[
\tau = -\frac{1}{k_1 k_2 C_{l_p}}
\]

(12.3)

and \(C_{l_{\delta a}}\) and \(C_{l_{\delta a}}\) should match the static test data.
The time constant $\tau$ of the roll mode depends only on the single derivative $C_{l \alpha}$, so correct identification of the roll mode properties depends critically on the correctness of the estimate for this single parameter. This is potentially much more difficult in the noisy test environment than estimating the parameters in a larger matrix of an oscillatory mode.

12.2 Input Design and Verification

The input used for the identification of the roll mode is a fast, repeated aileron doublet as shown in Figure 12.2. The recipe for success in the turbulence is to keep the aircraft responding to control surface motion as much as possible and to limit the unforced

Figure 12.2: Typical roll mode response to an aileron input sequence
motion. This reduces the issues illustrated in Figure 12.1. The input sequence shown in Figure 12.2 is so fast that the roll rate never settles. Due to the stability issues in the roll axis, as discussed in Section 9.2.3, the input also needs to start immediately after the control system is released to avoid drifting away from the trim condition. For the identification of the roll parameters it appears also to be beneficial to run the input sequence over the entire duration of the manoeuvre as shown in the Figure.

During initial testing it was discovered that $C_{l\beta}$ and $C_{l\delta a}$ are highly correlated for this aircraft and input sequence and cannot be estimated simultaneously. This was the case for any aileron input tested. Since the control derivatives for elevator and rudder from the static wind tunnel tests were shown in the previous sections to be of high quality, it was decided to keep $C_{l\delta a}$ fixed at its pre-determined result of $-0.178$ and to turn off the estimation of this parameter for this experiment. The resulting model fit plotted in the Figure is from the output error method and matches the data quite well, with residuals of less than 10%. The control inputs in the Figure depicts the three inputs to the model, with sideslip and yaw rate activated through the dutch roll mode. The data sources are similar to the dutch roll mode experiment, with $p$ and $r$ measured by the gyros and the sideslip angle coming from the yaw angle gimbal sensor.

Table 12.1 lists the identified parameters from the manoeuvre using the output error method. $C_{l\beta}$ matches the expected value from the static tests well. The two dynamic derivatives $C_{l\delta p}$ and $C_{l\delta r}$ are estimated with confidence intervals of 5% and 10%, respectively. This is expected to improve with more repeats of the experiment. This initial result shows that this model can be used to identify roll axis parameters with acceptable accuracy in the turbulent conditions.

### Table 12.1: Parameter ID results for a typical roll response to an aileron input. $C_{l\delta a} = -0.178$ fixed

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l\beta}$</td>
<td>-0.061</td>
<td>-0.062</td>
<td>$\pm$ 0.009 (0.005)</td>
<td>13.83 (8.29)</td>
<td>[ -0.079 -0.045 ]</td>
</tr>
<tr>
<td>$C_{l\delta p}$</td>
<td>-</td>
<td>-0.411</td>
<td>$\pm$ 0.019 (0.007)</td>
<td>4.60 (1.79)</td>
<td>[ -0.449 -0.373 ]</td>
</tr>
<tr>
<td>$C_{l\delta r}$</td>
<td>-</td>
<td>0.384</td>
<td>$\pm$ 0.041 (0.019)</td>
<td>10.59 (5.07)</td>
<td>[ 0.303 0.466 ]</td>
</tr>
</tbody>
</table>

$\tau = 0.55$ sec

R2 for Output 1: 99.52

12.3 Results

This section presents the results for the roll mode experiments. As before, a series of single inputs and a single long input sequence will be discussed, together with four repeats of the long series with slightly modified inputs.

The correlation of $C_{l\beta}$ and $C_{l\delta a}$ requires $C_{l\delta a}$ to be set to a fixed value to estimate $C_{l\beta}$. This cannot be done with the equation error function $\text{lesq}().\text{m}$ from SIDPAC, where one
can only add or remove regressor time series but cannot include a constant parameter (at least not to this author’s knowledge). To resolve the issue, the rolling moment equation was modified for the equation error method as follows:

\[ C_l - (C_{l_{\delta a}} \delta a) = C_{l_{\beta}} \beta + C_{l_p} \frac{p_b}{2V_{air}} + C_{l_r} \frac{r_b}{2V_{air}} \]  

(12.4)

with \( C_{l_{\delta a}} \) set to the reference value of \(-0.178\). This is equivalent to setting the parameter fixed in the other two methods and is expected to yield the correct results.

### 12.3.1 Repeated Single Inputs

Table 12.2 and Figure 12.3 show the results of identifying 10 input responses separately. The filter error method appears to have issues with this task and shows by far the largest uncertainties on each data point. Its results also do not match the static test results and the estimates of the other two methods. The equation- and output error method do a reasonable job and produce nearly identical results. The estimates for \( C_{l_{\beta}} \) are slightly on the low side with very large uncertainties. The characteristic parameter for the roll mode, \( C_{l_p} \), is estimated with high confidence, as is the cross derivative \( C_{l_r} \). The next section will use a long series of inputs to try and improve on the results for \( C_{l_{\beta}} \).

![Figure 12.3: Parameter ID results for 10 repeated aileron inputs with \( C_{l_{\delta a}} \) fixed at -0.178](image-url)
Table 12.2: Parameter ID results for 10 repeated aileron inputs with $C_{l_{\delta a}}$ fixed at -0.178

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-0.061</td>
<td>-0.057 ± 0.049</td>
<td>-0.056 ± 0.046</td>
<td>-0.045 ± 0.033</td>
</tr>
<tr>
<td>$C_{l_{p}}$</td>
<td>-</td>
<td>-0.420 ± 0.026</td>
<td>-0.420 ± 0.022</td>
<td>-0.315 ± 0.024</td>
</tr>
<tr>
<td>$C_{l_{c}}$</td>
<td>-</td>
<td>0.374 ± 0.092</td>
<td>0.369 ± 0.087</td>
<td>1.354 ± 0.216</td>
</tr>
</tbody>
</table>

12.3.2 Single Series of Multiple Inputs

A single series of ten aileron inputs is shown in Figure 12.4. After each input sequence, the flight controllers briefly re-engage to restore the trim condition. This ensures each input is started from wings level attitude. This re-trimming adds motion to all axes of the aircraft, most of which is not modelled by the roll mode model. This may increase the

Figure 12.4: OE results for a series of ten aileron inputs
uncertainties on some of the parameters, but cannot be avoided. The results of the filter error, despite a good fit to the data, are useless as before and are omitted. The equation error results are similar to the previous section and show no improvement for the long series.

Table 12.3 lists the output error results for the series of aileron inputs. Compared to Table 12.2 all uncertainties are reduced significantly. \( C_l_\beta \) was already predicted with good confidence and this has been confirmed by the current run. The estimate for \( C_l_\beta \) is within 6% of the static test result, which is a decent outcome, given the difficult conditions. The estimate for \( C_l_r \) is significantly smaller for this dataset, which may be caused by the restoration of the trim condition between the inputs, and it is hard to judge which estimate is correct. The combined lateral inputs in the next chapter may be better suited to determine the cross derivatives. As a final step, now four series of long input sequences will be tested to judge the repeatability of the results in Table 12.3.

### 12.3.3 Repeated Long Input Sequences

Table 12.4 and Figure 12.5 present the results of repeating the long input series four times. Each run used a small variation of \( \pm 75 \text{ msec} \) in the input timing to provide some variation. Again the filter error method shows the largest uncertainties on each estimate and its results do not agree with the static tests and the other two methods. It is therefore not suitable to identify the roll model, possibly due to its nature of being a single state model with only limited freedom to deal with the high noise levels. The equation- and output error methods deliver fairly consistent results. The characteristic derivative of the roll mode, \( C_l_\beta \), is estimated consistently with the previous results and with small

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_l_\beta )</td>
<td>-0.061</td>
<td>-0.065</td>
<td>( \pm 0.012 \ (0.001) )</td>
<td>18.15 (2.04)</td>
<td>[ -0.089 -0.042 ]</td>
</tr>
<tr>
<td>( C_l_\beta )</td>
<td>-</td>
<td>-0.424</td>
<td>( \pm 0.017 \ (0.002) )</td>
<td>3.98 (0.58)</td>
<td>[ -0.458 -0.390 ]</td>
</tr>
<tr>
<td>( C_l_r )</td>
<td>-</td>
<td>0.265</td>
<td>( \pm 0.041 \ (0.005) )</td>
<td>15.49 (1.93)</td>
<td>[ 0.183 0.348 ]</td>
</tr>
</tbody>
</table>

\( \tau = 0.53 \text{ sec} \)

R2 for Output 1: 97.57
uncertainty. The cross derivative $C_l$ is predicted consistently with the single long series, yet the difference to the repeated single inputs remains. The $C_{l\beta}$ derivative seems to be suffering from observability issues in these roll mode experiments. The value is however estimated with the correct order of magnitude. Overall, this is not a bad result in the difficult conditions.
12.4 Roll Mode Summary

This chapter reported the result of the roll axis test performed on the wind tunnel gimbal. The dynamic properties of the motion and the high noise levels presented a challenging environment. Despite these difficulties, the output- and equation error method were able to give reasonable parameter estimates for the coefficients in the roll equation. These final results are listed in Table 12.5, which are the results from the output error method in Table 12.4.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value±2σ</th>
<th>% Uncert.</th>
<th>Static Test</th>
<th>% Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{ls}$</td>
<td>-0.067 ± 0.011</td>
<td>14.4%</td>
<td>-0.061</td>
<td>8.9%</td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.416 ± 0.023</td>
<td>5.5%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{lr}$</td>
<td>0.222 ± 0.058</td>
<td>26%</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

The final step in this part of the thesis will be tests of a lateral combined manoeuvre, with combined rudder and aileron motion. This is hoped to confirm the results from the dutch roll and improve the roll mode results. These combined inputs were also used in flight, so their properties are of interest in the preparation of the flight data processing.
13. Lateral Combined Manoeuvre

In this final wind tunnel experiment, the parameters in the full lateral equation of motion will be estimated from responses to a combined rudder and aileron input. By doing this, it is expected to be able to capture the cross-coupling between the roll- and yaw axis better than with the separate inputs. These combined inputs will be longer and richer in information content, so the results, especially for the roll axis, may improve with this final experiment.

13.1 Model Structure

The model structure is derived from the lateral 6DoF equations of motion by removing all side force terms and the $a_y$ output equation as before. For this model the cross inertia $I_{xz}$ is kept, which complicates the model structure. Bias terms were added as before. The final model for the lateral 3DoF motion is given below:

**Theorem 13.1.1 — Lateral Equations of Motion.** The state space form of the linearised lateral equations on the wind tunnel gimbal is

\[
\begin{align*}
\dot{\beta} &= -r + b_\beta \\
\dot{p} &= c_3 L + c_4 N \\
\dot{r} &= c_4 L + c_9 N \\
\dot{\phi} &= p + \tan \theta_0 r + b_\phi
\end{align*}
\]
where

\[ L = \dot{q} S b \left( C_{l \beta} \beta + \frac{p b}{2V_{air}} + C_{l \delta a} \delta a + b_p \right) \]

\[ N = \dot{q} S b \left( C_{n \beta} \beta + \frac{p b}{2V_{air}} + C_{n \delta a} \delta a + b_r \right) \]

\[ \Gamma = I_x - I_z \frac{I_{xx}}{\Gamma} \quad c_3 = I_x / \Gamma \quad c_4 = I_{xx} / \Gamma \quad c_9 = I_x / \Gamma \]

and the potentially identifiable derivatives

\[ C_{l \beta}, C_{l \delta a}, C_{l a}, C_{l \delta r}, C_{n \beta}, C_{n \delta r}, C_{n a}, C_{n \delta a} \text{ and } C_{n b_r} \]

where \( C_{l \beta}, C_{n \beta}, C_{l a} \text{ and } C_{n b_r} \) should match the static test data.

The roll angle equation is used only for the filter error method, where reference [38] states that the filter error method is stabilised by its use due to the extra information in the observation equations. Because the roll angle has no effect on the other states in this model, there is no benefit to the output error method, as stated in the reference and a three state model is used for that method.

The data sources for this experiment are the roll and yaw gyroscope measurements as well as the gimbal yaw angle sensor for the sideslip. For the four state model the gimbal roll angle measurement is added.

### 13.2 Input Design and Verification

Two difference input sequences were used for this experiment. The first is a time skewed input, where the rudder doublet comes first to start the slower dutch roll mode and is then followed by an aileron input about one period of the dutch roll mode later. The second input sequence is an orthogonal square wave input as suggested in reference [22], where rudder and aileron move at the same time in opposite directions and the aileron at twice the speed of the rudder.

Figure 13.1 shows the time skewed input sequence the aircraft response, as well as the identified model. The individual input shapes are essentially the same as in the two previous chapters. The spacing of the two inputs was chosen to generate some kind of disruption to the dutch roll mode, where on commencement of the aileron input a sharp change in the motion occurs. This can be seen in the Figure in the yaw rate and to some extend in the side slip angle. This disruption helps to reduce the correlation between \( C_{l \beta} \) and \( C_{l a} \) and as a result it is much lower (0.81 vs. 0.95) than for the roll mode tests before. It is then possible to estimate the aileron control derivative, whereas before it had to be held fixed at the value from the static tests. The roll rate measurements are again much noisier than the rest of the data due to the wind tunnel turbulence and the model fit is less accurate in this axis.

Table 13.1 lists the identified parameters for this particular input. The results compare well with the static test data, with the first time estimate for \( C_{l a} \) being within 6% of the static result and the other derivatives better than that. For the dynamic derivatives, the
uncertainty of the cross coupling parameters $C_{l_r}$ and $C_{n_p}$ is larger than for the damping parameters of each axis. This is to be expected, as for a conventional airframe these cross-coupling terms are less important and therefore more difficult to observe. The roll mode derivative $C_{l_p}$ agrees well with the result from the roll mode experiment, as does the yaw damping derivative $C_{n_r}$ with the result from the dutch roll experiment. Overall, this input sequence seems to work well in the noisy conditions. These inputs will also be used in the flight tests, so the following tests for repeatability will also give an insight into the quality of data that can be expected from the flight tests.
Table 13.1: Parameter ID results from a time skewed lateral input, corresponding to Figure 13.1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_l\beta$</td>
<td>-0.061</td>
<td>-0.058 ± 0.005 (0.001)</td>
<td>9.43 (2.40)</td>
<td>[ -0.069 -0.047 ]</td>
<td></td>
</tr>
<tr>
<td>$C_l\rho$</td>
<td>-</td>
<td>-0.428 ± 0.033 (0.007)</td>
<td>7.72 (1.61)</td>
<td>[ -0.494 -0.362 ]</td>
<td></td>
</tr>
<tr>
<td>$C_l\varphi$</td>
<td>-</td>
<td>0.147 ± 0.020 (0.006)</td>
<td>13.79 (3.93)</td>
<td>[ 0.106 0.187 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{li_a}$</td>
<td>-0.178</td>
<td>-0.189 ± 0.012 (0.003)</td>
<td>6.44 (1.36)</td>
<td>[ -0.214 -0.165 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\beta}$</td>
<td>0.087</td>
<td>0.083 ± 0.002 (0.000)</td>
<td>2.32 (0.37)</td>
<td>[ 0.079 0.087 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\rho}$</td>
<td>-</td>
<td>-0.061 ± 0.015 (0.002)</td>
<td>24.90 (3.35)</td>
<td>[ -0.091 -0.031 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\varphi}$</td>
<td>-</td>
<td>-0.075 ± 0.005 (0.001)</td>
<td>7.14 (1.36)</td>
<td>[ -0.085 -0.064 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\delta_r}$</td>
<td>-0.063</td>
<td>-0.066 ± 0.004 (0.001)</td>
<td>5.42 (0.79)</td>
<td>[ -0.073 -0.058 ]</td>
<td></td>
</tr>
</tbody>
</table>

Dutch roll $\omega_n = 0.83$ Hz, $\zeta = 0.12$

R2 for Output 1: 96.92, R2 for Output 2: 96.40, R2 for Output 3: 96.41

The second input sequence, is shown in Figure 13.2. It features an orthogonal input sequence with simultaneous motion of rudder and aileron. The aileron is moving twice as fast and in the opposite direction. In order to increase the information content in the data, the input is run twice in short succession. This input fully removes the correlation.

Table 13.2: Parameter ID results from an orthogonal lateral input, corresponding to Figure 13.2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_l\beta$</td>
<td>-0.061</td>
<td>-0.059 ± 0.004 (0.001)</td>
<td>7.40 (2.06)</td>
<td>[ -0.068 -0.050 ]</td>
<td></td>
</tr>
<tr>
<td>$C_l\rho$</td>
<td>-</td>
<td>-0.420 ± 0.024 (0.007)</td>
<td>5.74 (1.69)</td>
<td>[ -0.468 -0.372 ]</td>
<td></td>
</tr>
<tr>
<td>$C_l\varphi$</td>
<td>-</td>
<td>0.109 ± 0.023 (0.005)</td>
<td>20.89 (4.53)</td>
<td>[ 0.063 0.154 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{li_a}$</td>
<td>-0.178</td>
<td>-0.188 ± 0.008 (0.002)</td>
<td>4.05 (1.26)</td>
<td>[ -0.204 -0.173 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\beta}$</td>
<td>0.087</td>
<td>0.078 ± 0.003 (0.001)</td>
<td>3.69 (0.68)</td>
<td>[ 0.072 0.084 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\rho}$</td>
<td>-</td>
<td>-0.064 ± 0.023 (0.004)</td>
<td>36.29 (6.35)</td>
<td>[ -0.110 -0.017 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\varphi}$</td>
<td>-</td>
<td>-0.063 ± 0.007 (0.001)</td>
<td>11.48 (1.40)</td>
<td>[ -0.077 -0.048 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\delta_r}$</td>
<td>-0.063</td>
<td>-0.058 ± 0.005 (0.001)</td>
<td>8.24 (1.13)</td>
<td>[ -0.068 -0.049 ]</td>
<td></td>
</tr>
</tbody>
</table>

Dutch roll $\omega_n = 0.81$ Hz, $\zeta = 0.12$

R2 for Output 1: 98.15, R2 for Output 2: 95.99, R2 for Output 3: 98.09
between $C_{l\beta}$ and $C_{lh\alpha}$, and produces an interesting, disruptive pattern in the roll rate. Table 13.2 lists the identified parameters from this input sequences. Most estimated values are very similar to the previous input sequence, just $C_{n\alpha}$ is a bit too low. On the other hand, most uncertainties of the primary derivatives are lower than for the time skewed input sequence. The repeated tests in the next section will show if this orthogonal input has an advantage over the time skewed one.
13.3 Results

In this section, the results for the lateral combined inputs are presented. Both input shapes will be compared to obtain the best possible reference data for the lateral derivatives of the test aircraft. As usual, the three identification methods will be used and compared on this more complicated model.

13.3.1 Repeated Single Input

Several time skewed inputs were run separately and the results are listed in Table 13.3. These results were obtained with the estimation of $C_{l_{\delta a}}$ enabled. The data is visualised in Figure 13.3. This appears to be the first time, where the filter error method gives better results than the output error and equation error methods. The filter error uncertainties are the lowest for all parameters and they are the most consistent with the results of the static and previous dynamic tests. The output error method seems to have problems with the remaining correlation between $C_{l_{\beta}}$ and $C_{l_{\delta a}}$, since its estimates for the roll axis parameters are not very good. A second run was made with the estimation of $C_{l_{\delta a}}$ turned off, as listed in Table 13.4. Here the output error results are much better, while the filter error method shows no difference between the two runs. The equation error estimates for $C_{l_{\beta}}$ are improved, and the result for $C_{l_{\beta}}$ has increased markedly. $C_{l_{\delta a}}$ shows a poor result with very large uncertainties for the equation error method.

Comparing these results with the roll mode results from the previous section, $C_{l_{\beta}}$ is predicted markedly lower by 22%, while $C_{l_{\beta}}$ comes out the same. The result for $C_{l_{\beta}}$ is also significantly lower than estimated from the roll mode (0.153 vs. 0.222), at least for the output and filter error methods. The equation error result for that parameter is in line with the results from the roll mode. In the yaw axis, $C_{n_{\beta}}$ is also estimated lower

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Static</th>
<th>EQN Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-0.061</td>
<td>$-0.049 \pm 0.046$</td>
<td>$-0.044 \pm 0.029$</td>
<td>$-0.052 \pm 0.005$</td>
</tr>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-</td>
<td>$-0.447 \pm 0.041$</td>
<td>$-0.327 \pm 0.190$</td>
<td>$-0.400 \pm 0.033$</td>
</tr>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-</td>
<td>$0.193 \pm 0.146$</td>
<td>$0.152 \pm 0.062$</td>
<td>$0.153 \pm 0.047$</td>
</tr>
<tr>
<td>$C_{l_{\delta a}}$</td>
<td>-0.178</td>
<td>$-0.199 \pm 0.032$</td>
<td>$-0.148 \pm 0.070$</td>
<td>$-0.178 \pm 0.011$</td>
</tr>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>0.087</td>
<td>$0.081 \pm 0.004$</td>
<td>$0.080 \pm 0.006$</td>
<td>$0.080 \pm 0.003$</td>
</tr>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>-</td>
<td>$-0.060 \pm 0.025$</td>
<td>$-0.067 \pm 0.016$</td>
<td>$-0.059 \pm 0.004$</td>
</tr>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>-</td>
<td>$-0.073 \pm 0.018$</td>
<td>$-0.067 \pm 0.016$</td>
<td>$-0.052 \pm 0.017$</td>
</tr>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>-0.063</td>
<td>$-0.069 \pm 0.010$</td>
<td>$-0.066 \pm 0.011$</td>
<td>$-0.064 \pm 0.008$</td>
</tr>
<tr>
<td>$\omega_n$ DR [Hz]</td>
<td>-</td>
<td>$0.806 \pm 0.048$</td>
<td>$0.825 \pm 0.021$</td>
<td>$0.814 \pm 0.017$</td>
</tr>
<tr>
<td>$\zeta$ DR</td>
<td>-</td>
<td>$0.129 \pm 0.051$</td>
<td>$0.123 \pm 0.038$</td>
<td>$0.102 \pm 0.030$</td>
</tr>
</tbody>
</table>
Figure 13.3: Parameter estimates for multiple time skewed input sequences, including the estimation of $C_{i_{kn}}$
Table 13.4: Parameter estimates for multiple time skewed input sequences without \( C_{ls} \)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2( \sigma )</th>
<th>OEM Mean±2( \sigma )</th>
<th>FEM Mean±2( \sigma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_{l\beta} )</td>
<td>-0.061</td>
<td>-0.043 ± 0.050</td>
<td>-0.054 ± 0.006</td>
<td>-0.052 ± 0.005</td>
</tr>
<tr>
<td>( C_{l\rho} )</td>
<td>-</td>
<td>-0.401 ± 0.051</td>
<td>-0.396 ± 0.033</td>
<td>-0.400 ± 0.019</td>
</tr>
<tr>
<td>( C_{l\tau} )</td>
<td>-</td>
<td>0.253 ± 0.064</td>
<td>0.147 ± 0.045</td>
<td>0.153 ± 0.047</td>
</tr>
<tr>
<td>( C_{n\beta} )</td>
<td>0.087</td>
<td>0.081 ± 0.004</td>
<td>0.081 ± 0.005</td>
<td>0.080 ± 0.003</td>
</tr>
<tr>
<td>( C_{n\rho} )</td>
<td>-</td>
<td>-0.060 ± 0.025</td>
<td>-0.062 ± 0.007</td>
<td>-0.058 ± 0.004</td>
</tr>
<tr>
<td>( C_{n\tau} )</td>
<td>-</td>
<td>-0.073 ± 0.018</td>
<td>-0.071 ± 0.015</td>
<td>-0.052 ± 0.017</td>
</tr>
<tr>
<td>( C_{n\delta} )</td>
<td>-0.063</td>
<td>-0.069 ± 0.010</td>
<td>-0.066 ± 0.011</td>
<td>-0.064 ± 0.007</td>
</tr>
</tbody>
</table>

than during the dutch roll runs by 10\%, similar to \( C_{n\tau} \), which is estimated lower by 24\% by the filter error method. The output and equation error methods, however, give a good result for this parameter. The rudder control derivative matches well between the experiments. The final result of this test is a somewhat mixed picture. The filter error method appears to give better estimates for all parameters but \( C_{n\tau} \), where the other two methods are much better. Since the yaw damping is a very important parameter, a more consistent result is needed. This leads to the repeated test of the orthogonal input sequence.

The results for multiple, separate runs with the orthogonal input sequence are shown in Figure 13.4 and are listed in Table 13.5. The estimation of \( C_{ls} \) was enabled and did not cause any issues as expected. The results show a much better consistency between output and filter error methods, while the equation error method has large uncertainties on many parameters, although its resulting estimates are reasonably close to the other methods.

Table 13.5: Parameter estimates for multiple orthogonal input sequences, including the estimation of \( C_{ls} \)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2( \sigma )</th>
<th>OEM Mean±2( \sigma )</th>
<th>FEM Mean±2( \sigma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_{l\beta} )</td>
<td>-0.061</td>
<td>-0.070 ± 0.107</td>
<td>-0.061 ± 0.008</td>
<td>-0.059 ± 0.007</td>
</tr>
<tr>
<td>( C_{l\rho} )</td>
<td>-</td>
<td>-0.449 ± 0.042</td>
<td>-0.423 ± 0.057</td>
<td>-0.403 ± 0.039</td>
</tr>
<tr>
<td>( C_{l\tau} )</td>
<td>-</td>
<td>0.169 ± 0.193</td>
<td>0.119 ± 0.041</td>
<td>0.125 ± 0.035</td>
</tr>
<tr>
<td>( C_{l\delta} )</td>
<td>-0.178</td>
<td>-0.194 ± 0.015</td>
<td>-0.185 ± 0.020</td>
<td>-0.183 ± 0.012</td>
</tr>
<tr>
<td>( C_{n\beta} )</td>
<td>0.087</td>
<td>0.081 ± 0.015</td>
<td>0.078 ± 0.003</td>
<td>0.079 ± 0.002</td>
</tr>
<tr>
<td>( C_{n\rho} )</td>
<td>-</td>
<td>-0.049 ± 0.117</td>
<td>-0.067 ± 0.020</td>
<td>-0.056 ± 0.007</td>
</tr>
<tr>
<td>( C_{n\tau} )</td>
<td>-</td>
<td>-0.076 ± 0.032</td>
<td>-0.078 ± 0.027</td>
<td>-0.067 ± 0.020</td>
</tr>
<tr>
<td>( C_{n\delta} )</td>
<td>-0.063</td>
<td>-0.071 ± 0.030</td>
<td>-0.066 ± 0.014</td>
<td>-0.067 ± 0.007</td>
</tr>
</tbody>
</table>
Figure 13.4: Parameter estimates for multiple orthogonal input sequences, including the estimation of $C_{l_{\delta a}}$

The output- and filter error estimates for the roll axis now match the static tests well, similarly to the value for $C_{l_{p}}$ agreeing well with the roll mode results. The estimate for
$C_{l_{e}}$ is still only half of what was found from the roll mode. The roll mode result for this parameter is probably not very good and the estimates from the combined inputs should be used for it. In the yaw axis, $C_{n_{\beta}}$ is still estimated low by about 10%, while the rudder control derivative matches well. The results for the yaw damping parameter $C_{n_{r}}$ from the filter error method now match the dutch roll results well, while the output error result is high by 10%. The estimation of this parameter is probably still affected by low sensitivity of the model for this low damped mode, similarly to the discussion in the dutch roll mode section. Overall, this orthogonal input seems to work better with the output- and filter error methods, allowing to estimate all model parameters and giving more consistent results. Some inconsistency with the static test results remain, however. In an attempt to improve this, now a series of inputs will be run as a long sequence to provide more information to the algorithms.

### 13.3.2 Sequence of Multiple Inputs

A series of ten time skewed inputs was run through the output and filter error methods. The results are listed in Tables 13.6 and 13.7, respectively. Both sets of results are very similar in value of the parameter estimate and it uncertainty. As before, the stability derivatives $C_{l_{\beta}}$ and $C_{n_{\beta}}$ are predicted about 10% smaller than the static test results. The yaw damping derivative $C_{n_{r}}$ is estimated smaller by the filter error method but matched the dutch roll experiment well, when the output error method is used. This is very similar to the repeated single inputs for this input shape. Therefore running a single, long sequence instead of the individual inputs does not have any advantage for the time skewed input. One explanation might be that during the re-trimming between the inputs the dynamics of the aircraft change significantly as the closed loop controllers take over. This different motion is part of the estimation process and may void any advantage of the larger information content of the long dataset. For all previous tests, this change in dynamics occurred in a similar manner, but must have been less significant, as the long series usually improved the results of the parameter ID. A similar picture emerges from the results of the long series of orthogonal inputs as listed in Tables 13.8 and 13.9. There is no improvement over the sequence of single inputs for the output error method and only small improvements for the filter error results. Nevertheless, these filter error results represent the best estimate for the lateral derivatives, matching the static test data well in the control derivatives, as well as the characteristic derivatives of the roll mode $C_{l_{p}}$ and the yaw damping $C_{n_{r}}$. The estimate for $C_{l_{\beta}}$ is also close to the static test result. The stability derivative $C_{n_{\beta}}$ continues to be predicted lower by about 10%. Since this is a consistent result across all the lateral combined inputs, this might point to problems with the other experiments, although the estimate for this parameter from the dutch roll mode also had a small uncertainty. Given the presence of the strong turbulence, however, this cannot be verified any further. It would need a comparative test in a clean wind tunnel to pinpoint the reason for the under-prediction of the weathercock stability derivative from the lateral combined inputs.
### Table 13.6: Output error results for a single series of time skewed inputs, not estimating $\delta a$

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-0.061</td>
<td>-0.053 ± 0.003 (0.000)</td>
<td>5.22 (0.87)</td>
<td>[ -0.059 -0.048 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{l_{p}}$</td>
<td>-0.397</td>
<td>± 0.013 (0.002)</td>
<td>3.39 (0.59)</td>
<td>[ -0.424 -0.370 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{l_{r}}$</td>
<td>0.163</td>
<td>± 0.011 (0.002)</td>
<td>6.85 (1.10)</td>
<td>[ 0.141 0.186 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>0.087</td>
<td>0.080 ± 0.001 (0.000)</td>
<td>1.25 (0.12)</td>
<td>[ 0.078 0.082 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{p}}$</td>
<td>-0.060</td>
<td>± 0.006 (0.001)</td>
<td>10.38 (0.98)</td>
<td>[ -0.072 -0.047 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{r}}$</td>
<td>-0.060</td>
<td>± 0.003 (0.000)</td>
<td>4.79 (0.54)</td>
<td>[ -0.066 -0.054 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{\delta r}}$</td>
<td>-0.063</td>
<td>-0.061 ± 0.001 (0.000)</td>
<td>2.37 (0.27)</td>
<td>[ -0.064 -0.058 ]</td>
<td></td>
</tr>
</tbody>
</table>

### Table 13.7: Filter error results for a single series of time skewed inputs, estimating all parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-0.061</td>
<td>-0.051 ± 0.003 (0.000)</td>
<td>6.11 (0.92)</td>
<td>[ -0.058 -0.045 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{l_{p}}$</td>
<td>-0.396</td>
<td>± 0.015 (0.002)</td>
<td>3.73 (0.62)</td>
<td>[ -0.426 -0.367 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{l_{r}}$</td>
<td>0.173</td>
<td>± 0.012 (0.002)</td>
<td>6.99 (0.98)</td>
<td>[ 0.149 0.197 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{l_{\delta a}}$</td>
<td>-0.178</td>
<td>-0.176 ± 0.005 (0.001)</td>
<td>3.05 (0.52)</td>
<td>[ -0.187 -0.166 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>0.087</td>
<td>0.080 ± 0.001 (0.000)</td>
<td>1.42 (0.12)</td>
<td>[ 0.077 0.082 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{p}}$</td>
<td>-0.057</td>
<td>± 0.007 (0.001)</td>
<td>12.37 (1.08)</td>
<td>[ -0.071 -0.043 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{r}}$</td>
<td>-0.049</td>
<td>± 0.003 (0.000)</td>
<td>6.66 (0.59)</td>
<td>[ -0.056 -0.043 ]</td>
<td></td>
</tr>
<tr>
<td>$C_{n_{\delta r}}$</td>
<td>-0.063</td>
<td>-0.063 ± 0.002 (0.000)</td>
<td>3.09 (0.28)</td>
<td>[ -0.067 -0.059 ]</td>
<td></td>
</tr>
</tbody>
</table>
Table 13.8: Output error results for a single series of orthogonal inputs, estimating all parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l\beta}$</td>
<td>-0.061</td>
<td>-0.064 ± 0.003 (0.001)</td>
<td>4.73 (0.87)</td>
<td>[-0.070 -0.058]</td>
<td></td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.445</td>
<td>± 0.014 (0.003)</td>
<td>3.11 (0.73)</td>
<td>[-0.472 -0.417]</td>
<td></td>
</tr>
<tr>
<td>$C_{lr}$</td>
<td>0.123</td>
<td>± 0.015 (0.002)</td>
<td>11.90 (1.78)</td>
<td>[0.094 0.152]</td>
<td></td>
</tr>
<tr>
<td>$C_{ls}$</td>
<td>-0.178</td>
<td>-0.188 ± 0.004 (0.001)</td>
<td>2.11 (0.56)</td>
<td>[-0.195 -0.180]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\beta}$</td>
<td>0.087</td>
<td>0.077 ± 0.002 (0.000)</td>
<td>2.01 (0.31)</td>
<td>[0.074 0.081]</td>
<td></td>
</tr>
<tr>
<td>$C_{np}$</td>
<td>-0.074</td>
<td>± 0.011 (0.002)</td>
<td>15.22 (2.30)</td>
<td>[-0.097 -0.052]</td>
<td></td>
</tr>
<tr>
<td>$C_{nr}$</td>
<td>-0.073</td>
<td>± 0.003 (0.000)</td>
<td>4.59 (0.54)</td>
<td>[-0.079 -0.066]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\delta}$</td>
<td>-0.063</td>
<td>-0.063 ± 0.002 (0.000)</td>
<td>3.08 (0.40)</td>
<td>[-0.067 -0.059]</td>
<td></td>
</tr>
</tbody>
</table>

Table 13.9: Filter error results for a single series of orthogonal inputs, estimating all parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est. Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l\beta}$</td>
<td>-0.061</td>
<td>-0.058 ± 0.003 (0.001)</td>
<td>5.75 (0.91)</td>
<td>[-0.064 -0.051]</td>
<td></td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.395</td>
<td>± 0.016 (0.003)</td>
<td>4.03 (0.78)</td>
<td>[-0.426 -0.363]</td>
<td></td>
</tr>
<tr>
<td>$C_{lr}$</td>
<td>0.129</td>
<td>± 0.015 (0.002)</td>
<td>11.37 (1.52)</td>
<td>[0.100 0.159]</td>
<td></td>
</tr>
<tr>
<td>$C_{ls}$</td>
<td>-0.178</td>
<td>-0.178 ± 0.005 (0.001)</td>
<td>2.57 (0.57)</td>
<td>[-0.187 -0.168]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\beta}$</td>
<td>0.087</td>
<td>0.078 ± 0.002 (0.000)</td>
<td>2.77 (0.32)</td>
<td>[0.074 0.082]</td>
<td></td>
</tr>
<tr>
<td>$C_{np}$</td>
<td>-0.056</td>
<td>± 0.015 (0.002)</td>
<td>27.27 (3.18)</td>
<td>[-0.086 -0.025]</td>
<td></td>
</tr>
<tr>
<td>$C_{nr}$</td>
<td>-0.064</td>
<td>± 0.004 (0.000)</td>
<td>5.97 (0.52)</td>
<td>[-0.071 -0.056]</td>
<td></td>
</tr>
<tr>
<td>$C_{n\delta}$</td>
<td>-0.063</td>
<td>-0.065 ± 0.003 (0.000)</td>
<td>4.54 (0.40)</td>
<td>[-0.071 -0.059]</td>
<td></td>
</tr>
</tbody>
</table>
13.4 Lateral Combined Summary

Summarising, the lateral combined input, using the orthogonal input sequence, gave the best estimates for the lateral parameters, especially when using the filter error method. For the output error method it is better to use single inputs separately and average the results. The filter error method is slightly better with the single, continuous dataset. Table 13.10 list the final results from this chapter for reference. These are taken from the filter error estimates of Table 13.9.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value±2σ</th>
<th>% Uncert.</th>
<th>Static Test</th>
<th>% Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l_s}$</td>
<td>$-0.058 \pm 0.003$</td>
<td>5.75%</td>
<td>-0.061</td>
<td>4.9%</td>
</tr>
<tr>
<td>$C_{l_p}$</td>
<td>$-0.395 \pm 0.016$</td>
<td>4.03%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{l_r}$</td>
<td>$0.129 \pm 0.015$</td>
<td>11.4%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{l_{bs}}$</td>
<td>$-0.178 \pm 0.005$</td>
<td>2.57%</td>
<td>-0.178</td>
<td>-</td>
</tr>
<tr>
<td>$C_{n_s}$</td>
<td>$0.078 \pm 0.002$</td>
<td>2.77%</td>
<td>0.087</td>
<td>10%</td>
</tr>
<tr>
<td>$C_{n_p}$</td>
<td>$-0.056 \pm 0.015$</td>
<td>27.3%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{n_r}$</td>
<td>$-0.064 \pm 0.004$</td>
<td>5.97%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{n_{bs}}$</td>
<td>$-0.065 \pm 0.003$</td>
<td>4.54%</td>
<td>-0.063</td>
<td>3%</td>
</tr>
</tbody>
</table>
Reference Data: Collected Results
Longitudinal Derivatives

The final results from the static and dynamic tests in the longitudinal axis are collected in Table 13.11. The results of the calculations to extract $C_{m\alpha}$ are listed in Table 13.12. Since there is only a single dynamic mode to test, the selection of the best estimate for $C_{mq}'$ is trivial. For all other derivatives the static test results are considered the most accurate. The data selected for the final reference results are highlighted in bold.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Dynamic WT ±2σ</th>
<th>% Uncert.</th>
<th>Static WT ±2σ</th>
<th>% Diff.</th>
<th>PanAir</th>
<th>% Diff.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L\alpha}$</td>
<td>-</td>
<td>-</td>
<td>5.167 ± 0.103</td>
<td>-</td>
<td>5.16</td>
<td>-</td>
</tr>
<tr>
<td>$C_{L\delta e}$</td>
<td>-</td>
<td>-</td>
<td>−0.511 ± 0.010</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{m\alpha}$</td>
<td>−0.930 ± 0.032</td>
<td>3.4%</td>
<td>−0.948 ± 0.019</td>
<td>−0.18%</td>
<td>−0.964</td>
<td>−3.5%</td>
</tr>
<tr>
<td>$C_{mq}'$</td>
<td>−12.099 ± 0.386</td>
<td>3.1%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{m\delta e}$</td>
<td>−1.113 ± 0.042</td>
<td>3.7%</td>
<td>−1.143 ± 0.023</td>
<td>2.6%</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 13.12: Derived longitudinal results

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{mq}$</td>
<td>−8.17</td>
</tr>
<tr>
<td>$C_{m\alpha}$</td>
<td>−3.93</td>
</tr>
</tbody>
</table>

Lateral Derivatives

The final results of the three lateral dynamic tests are listed in Tables 13.13, 13.14 and 13.15. Table 13.16 contains further lateral data from the static tests that could not be verified during the dynamic tests due to the limitations of the 3DoF motion. These values, however, will also be obtained from the flight tests and can be compared there.

The results of the lateral combined input appear to be the most consistent estimates for the lateral derivatives, except for $C_{n\beta}$, which is estimated with good confidence but yet comes out 10% lower than the static test result and the dutch roll estimate. The reason for this is unknown at this time and it would need tests in a clean wind tunnel to further investigate the issue. The data selected for the final reference results from the three experiments and the static tests are highlighted in bold.
Table 13.13: Lateral results from dutch roll mode

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Dynamic WT ±2σ</th>
<th>% Uncert.</th>
<th>Static WT ±2σ</th>
<th>% Diff.</th>
<th>PanAir</th>
<th>% Diff.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{n\beta}$</td>
<td>0.089 ± 0.004</td>
<td>4.5%</td>
<td>0.087 ± 0.002</td>
<td>2.3%</td>
<td>0.0911</td>
<td>1.2%</td>
</tr>
<tr>
<td>$C_{nr}$</td>
<td>-0.068 ± 0.01</td>
<td>15.6%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{n\delta r}$</td>
<td>-0.063 ± 0.008</td>
<td>12.7%</td>
<td>-0.063 ± 0.001</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 13.14: Lateral results from roll mode

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Dynamic WT ±2σ</th>
<th>% Error</th>
<th>Static WT ±2σ</th>
<th>% Diff.</th>
<th>PanAir</th>
<th>% Diff.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l\beta}$</td>
<td>-0.067 ± 0.011</td>
<td>14.4%</td>
<td>-0.061 ± 0.001</td>
<td>8.9%</td>
<td>-0.065</td>
<td>3.1%</td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.416 ± 0.023</td>
<td>5.5%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$Cl_r$</td>
<td>0.222 ± 0.058</td>
<td>26%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 13.15: Lateral results from a combined rudder and aileron input

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Dynamic WT ±2σ</th>
<th>% Uncert.</th>
<th>Reference</th>
<th>Diff.</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l\beta}$</td>
<td>-0.058 ± 0.003</td>
<td>5.75%</td>
<td>-0.061</td>
<td>4.9%</td>
<td>Static WT</td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.395 ± 0.016</td>
<td>4.03%</td>
<td>-0.416</td>
<td>5%</td>
<td>Roll mode</td>
</tr>
<tr>
<td>$Cl_r$</td>
<td>0.129 ± 0.015</td>
<td>11.4%</td>
<td>0.222</td>
<td>41%</td>
<td>Roll mode</td>
</tr>
<tr>
<td>$C_{la}$</td>
<td>0.178 ± 0.005</td>
<td>2.57%</td>
<td>0.178</td>
<td>-</td>
<td>Static WT</td>
</tr>
<tr>
<td>$C_{n\beta}$</td>
<td>0.078 ± 0.002</td>
<td>2.77%</td>
<td>0.087</td>
<td>10%</td>
<td>Static WT</td>
</tr>
<tr>
<td>$C_{np}$</td>
<td>-0.056 ± 0.015</td>
<td>27.3%</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$C_{nr}$</td>
<td>-0.064 ± 0.004</td>
<td>5.97%</td>
<td>-0.068</td>
<td>6.25%</td>
<td>Dutch Roll</td>
</tr>
<tr>
<td>$C_{n\delta r}$</td>
<td>-0.065 ± 0.003</td>
<td>4.54%</td>
<td>-0.063</td>
<td>3.1%</td>
<td>Static WT</td>
</tr>
</tbody>
</table>

Table 13.16: Additional lateral results from static testing

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value ±2σ</th>
<th>% Uncert.</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{y\beta}$</td>
<td>-0.51 ± 0.01</td>
<td>N/A</td>
<td>Static WT</td>
</tr>
<tr>
<td>$C_{y\theta}$</td>
<td>N/A</td>
<td>-</td>
<td>Not possible</td>
</tr>
<tr>
<td>$C_{y\phi}$</td>
<td>N/A</td>
<td>-</td>
<td>Not possible</td>
</tr>
<tr>
<td>$C_{y\alpha_a}$</td>
<td>0</td>
<td>-</td>
<td>Static WT</td>
</tr>
<tr>
<td>$C_{y\alpha_r}$</td>
<td>0.169 ± 0.003</td>
<td>N/A</td>
<td>Static WT</td>
</tr>
<tr>
<td>$C_{n\delta a}$</td>
<td>≈ 0</td>
<td>-</td>
<td>Static WT</td>
</tr>
<tr>
<td>$C_{l\delta r}$</td>
<td>≈ 0</td>
<td>-</td>
<td>Static WT</td>
</tr>
</tbody>
</table>
Flight Test Reference Card

A summary of all data collected from wind tunnel testing and numerical simulations to be used as a reference to benchmark the flight test results.

Reference Geometry/Mass Properties

Reference geometry, CG location and mass properties for all reference data in this part.

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_{ref}$</td>
<td>0.425</td>
<td>$m^2$</td>
</tr>
<tr>
<td>$C_{ref}$</td>
<td>0.278</td>
<td>$m$</td>
</tr>
<tr>
<td>$B_{ref}$</td>
<td>1.530</td>
<td>$m$</td>
</tr>
<tr>
<td>$AR$</td>
<td>5.5</td>
<td>-</td>
</tr>
<tr>
<td>$X_{CG}$</td>
<td>0.240</td>
<td>$m$</td>
</tr>
<tr>
<td>$Y_{CG}$</td>
<td>0.0</td>
<td>$m$</td>
</tr>
<tr>
<td>$Z_{CG}$</td>
<td>0.225</td>
<td>$m$</td>
</tr>
</tbody>
</table>

Table 13.17: Final results including the added mass components [$kg \cdot m^2$]

<table>
<thead>
<tr>
<th>Axis</th>
<th>Value [kg m²]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{XX}$</td>
<td>0.22</td>
</tr>
<tr>
<td>$I_{YY}$</td>
<td>0.31</td>
</tr>
<tr>
<td>$I_{ZZ}$</td>
<td>0.51</td>
</tr>
<tr>
<td>$I_{XZ}$</td>
<td>-0.01</td>
</tr>
</tbody>
</table>

Longitudinal Derivatives (Free Air)

Longitudinal reference data corrected for wall interference, to be used to compare against flight data. Since $C_{L_0}$ is assumed small and no wall corrections apply, the result for $C'_{m_q}$ in the short period mode model is expected to be identical in flight and in the wind tunnel.

The expected value for $C'_{m_\alpha}$ in the 6 DoF short period mode model can be calculated from

$$C'_{m_\alpha} = C_{m_\alpha} - \frac{\rho S c}{4m} m_{\alpha} C_{L_0}$$

(13.2)

$$= -0.626$$

Due to the additional uncertainty of the estimates for $C_{m_\alpha}$ and $C_{L_0}$, the confidence interval for $C'_{m_\alpha}$ will also be larger and is assumed to be around 5%, which is more than twice the uncertainty of $C_{m_\alpha}$. The expected value for $C'_{m_{\delta e}}$ is found from

$$C'_{m_{\delta e}} = C_{m_{\delta e}} - \frac{\rho S c}{4m} m_{\delta e} C_{L_{\delta e}}$$

(13.3)

$$= -1.159$$

(13.4)
Similar adjustments to the uncertainty as with $C_{m_{\alpha}}$ were made. This completes the parameters to be expected during the flight tests in the longitudinal axis and they are summarised in the table below.

### Table 13.18: Expected values for the longitudinal derivatives in flight

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>% Uncertainty</th>
<th>95% Confidence Interval</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_{\alpha}}$</td>
<td>4.540</td>
<td>2%</td>
<td>[4.450 4.631]</td>
<td>Numerical Correction</td>
</tr>
<tr>
<td>$C_{L_{\delta q}}$</td>
<td>$\approx 0$</td>
<td>-</td>
<td>-</td>
<td>N/A</td>
</tr>
<tr>
<td>$C_{L_{s_{\alpha}}}$</td>
<td>-0.511</td>
<td>2%</td>
<td>[-0.521 - 0.501]</td>
<td>Corr. assumed small</td>
</tr>
<tr>
<td>$C_{'m_{\alpha}}$</td>
<td>-0.626</td>
<td>5%</td>
<td>[-0.595 - 0.657]</td>
<td>Numerical correction</td>
</tr>
<tr>
<td>$C_{'m_{q}}$</td>
<td>-12.099</td>
<td>3.1%</td>
<td>[-13.092 - 11.606]</td>
<td>No correction necessary</td>
</tr>
<tr>
<td>$C_{'m_{s_{\alpha}}}$</td>
<td>-1.159</td>
<td>5%</td>
<td>[-1.219 - 1.11]</td>
<td>Corr. assumed small</td>
</tr>
</tbody>
</table>

### Lateral Derivatives

Lateral reference data compiled from all ground test results, to be used to compare against flight data. No wall corrections are required for lateral derivatives. All static data was verified during dynamic testing.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>% Uncertainty</th>
<th>95% Confidence Interval</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{y_{\beta}}$</td>
<td>-0.510</td>
<td>5%</td>
<td>[-0.520 - 0.500]</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{y_{p}}$</td>
<td>N/A</td>
<td>-</td>
<td>-</td>
<td>Flight only</td>
</tr>
<tr>
<td>$C_{y_{r}}$</td>
<td>N/A</td>
<td>-</td>
<td>-</td>
<td>Flight only</td>
</tr>
<tr>
<td>$C_{y_{s_{\alpha}}}$</td>
<td>$\approx 0$</td>
<td>-</td>
<td>-</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{y_{s_{r}}}$</td>
<td>0.169</td>
<td>5%</td>
<td>[0.166 0.172]</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-0.061</td>
<td>5%</td>
<td>[-0.062 - 0.059]</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{l_{p}}$</td>
<td>-0.395</td>
<td>4.03%</td>
<td>[-0.379 - 0.411]</td>
<td>Roll mode</td>
</tr>
<tr>
<td>$C_{l_{r}}$</td>
<td>0.129</td>
<td>11.4%</td>
<td>[0.114 0.144]</td>
<td>Lat. Combined</td>
</tr>
<tr>
<td>$C_{l_{s_{\alpha}}}$</td>
<td>-0.178</td>
<td>5%</td>
<td>[-0.182 - 0.174]</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{l_{s_{r}}}$</td>
<td>$\approx 0$</td>
<td>-</td>
<td>-</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{n_{\beta}}$</td>
<td>0.087</td>
<td>5%</td>
<td>[0.085 0.089]</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{n_{p}}$</td>
<td>-0.056</td>
<td>27.3%</td>
<td>[-0.071 - 0.041]</td>
<td>Lat. Combined</td>
</tr>
<tr>
<td>$C_{n_{r}}$</td>
<td>-0.064</td>
<td>5.97%</td>
<td>[-0.069 - 0.060]</td>
<td>Dutch roll</td>
</tr>
<tr>
<td>$C_{n_{s_{\alpha}}}$</td>
<td>$\approx 0$</td>
<td>-</td>
<td>-</td>
<td>Static</td>
</tr>
<tr>
<td>$C_{n_{s_{r}}}$</td>
<td>-0.063</td>
<td>5%</td>
<td>[-0.064 - 0.062]</td>
<td>Static</td>
</tr>
</tbody>
</table>

Table 13.19: Average parameter estimates with 95% confidence interval.
14 Flight Operations ................. 217
14.1 Requirements
14.2 A Typical Research Flight
14.3 Lessons Learned
14.4 Summary

15 Flight Data Preparation ............ 229
15.1 Data Compatibility Check
15.2 Properties of Selected Flight Data Sets
15.3 Summary

16 Longitudinal Results ............... 243
16.1 Introduction
16.2 Model Structure
16.3 Input Design and Verification
16.4 Multiple Input Results
16.5 Summary

17 Lateral Flight Results ............... 263
17.1 Dutch Roll Mode
17.2 Roll Mode
17.3 Lateral Combined Manoeuvre
17.4 Summary
14. Flight Operations

Flight testing of a small, remotely piloted aircraft presents its own set of challenges. In this chapter some of these difficulties will be discussed and the solutions used for this project will be presented.

14.1 Requirements

The requirements for a flight test project are the same and independent from the aircraft size. What differs are the methods used to meet these requirements. The most important items are listed below:

- Be safe!
- Be productive.
- Be able to react quickly to findings, issues and circumstances

The first topic is self explanatory. During full scale testing, there are lives at stake. That is not the case for a small, remotely piloted aircraft, but any damage to the, usually unique, test airframe will cause major delays of the project and large amounts of additional workload. So it is vital to do things carefully and methodically. If the conditions are not right, just don’t fly. Otherwise accidents will happen.

The second topic is vital to the progress of the research project. In any flight test programme there is only limited time to fly. So every bit of flight time needs to be used to gather useful data. But there will inevitably be problems, especially with a developmental prototype, like the UAVmainframe. In order to be productive, one therefore needs to satisfy topic three by being able to react quickly to problems and adjust the test schedule accordingly. But it is also vital to have some time buffer to be able to fix larger issues and come back for another flight later on.

Satisfying the last two requirements during a small, low budget university flight test
Chapter 14. Flight Operations

programme, is difficult due to limitations in man-power, budget and, particularly for this project, the fact that all hardware used during the flight tests was a brand new prototype of flight control and data acquisition system with very complicated software. Another limitation, that is overlooked often, is the lack of experience of the involved parties. As a PhD student, almost everything is done for the first time and the learning curve for successful flight tests is incredibly steep. This is seldom acknowledged in academia and leads to high levels of stress during the project. During the five years duration of this project, almost 50 days were spend at the flying field, with more than 150 flights. Only on the very last two days everything finally worked at a fully satisfactory level. But on the last day the weather was bad...

14.2 A Typical Research Flight

A typical research flight to gather data, as opposed to a checkout or system tuning flight, has several distinct phases, which are the same for all of them. The procedures have been developed from the experience of many test flights and worked well for this project.

All flights took place at the University of Sydney’s flight test centre located near the town of Marulan in the southern highlands of New South Wales, about 2.5 hours south of Sydney. The facility is located on a research farm owned by the university. Due to the remote location, it has been cleared by the Civil Aviation Safety Authority (CASA) for flight testing of experimental vehicles including autonomous operations, which was up until recently a quite unique feature. The facility has a 300m tarmac runway, a service shed for aircraft maintenance and provides accommodation. An aerial view is shown in Figure 14.1. Being on a remote location like this requires careful planning, because anything one forgets to bring is at least a five hour drive away. Any damage to the test aircraft that cannot be repaired on location immediately ends the flight test excursion. As a result, this author and his colleagues have become very skilled at on the spot repairs with any kind of material available.

Figure 14.1: Aerial view of the Marulan flight test centre
Before the flight, the aircraft fitted with charged batteries and a visual inspection is performed to identify any damage, especially to the landing gear, motor and propeller and the airdata probe. The UAVmainframe is then booted up, being connected to ground power and Ethernet. The feature of the UAVmainframe to be able to be powered from a ground and a flight battery with uninterrupted switch over saves a lot of time and ensures a fully charged flight battery, no matter how long the system is up and running before the flight. This uninterrupted power supply also allows for all sensors to be powered constantly, which helps with the temperature dependency of the MEMS sensors. Once the system is up and connected to Qgroundcontrol, a check of all sensor readings is done, as well as a verification of all control parameters. As these are initialised to the values of the previous run of the code, some modifications maybe required, especially if the input definitions were changed between flights. It is also good practise to inspect the PID gain parameters to catch any errors in these settings before flight.

When it is time to fly, the pilot checks all control surfaces and the motor response, while the sensor readings of these checks are monitored on the ground station. After confirming that the UAVmainframe is in ‘manual’ flight mode, the pilot taxis out to the runway and lines up for take off. After final checks of the sensor data, the UAVmainframe is ‘armed’ to start the data recording and to enable the other flight modes. Sometimes during that check there were large bias errors on the air pressure sensors that required a re-bias. These errors have been traced to the hard vibrations of the airframe while taxiing. On this scale a tarmac runway is actually quite rough, with small rocks and other dirt causing further bumps. This appears to shake the membranes inside the pressure sensors so that they de-bias. Mounting the UAVmainframe on a rubber base and monitoring the airdata bias values has improved the issue.

After take off, the pilot does a few manual manoeuvres to confirm everything is working well. Then the UAVmainframe is switched to ‘stabilise’ mode, which engages the flight controllers and stabilises the aircraft in all three axis. From there on the pilot only has to do minimal control of the attitude and can fully concentrate on the flight path. To initialise the EKF, a straight and level segment is flown with the pilots hands-off the controls. This yields a segment of data undisturbed by any control inputs and helps with the initial convergence of the filter during the data analysis. The straight and level segment is followed by a constant bank angle figure eight to obtain data to stabilise the wind estimate early in the EKF run. If all is well, then the circuits for running the input sequences can begin. These initial manoeuvres take about 3 minutes or a quarter to a third of the entire flight time, depending on the battery state, which is constantly monitored on the ground station.

The manoeuvre circuits are flown into and away from the wind to limit any sideways disturbances due to the wind, which will is especially important for the lateral inputs. Because even low speed winds are typically a quarter or more of the airspeed, lateral gusts just blow the little aircraft over in roll and causes a quickly diverging spiral during the long hold times after the lateral inputs. Since the UAVmainframe at this stage does not have an online wind estimator, the wind direction is determined by the pilot on the ground, which is prone to error. This led to many aborted and therefore mostly useless lateral inputs.
Due to the reliable flight controllers of the UAVmainframe, the pilot was confident to fly the aircraft much further away than it would normally be safe because the aircraft becomes very hard to see in the distance against the sky. Since the pilot was able to rely on the attitude control and a predictable response to control inputs, flying the turns for the next circuit in the distance without proper visual attitude feedback was acceptable and enabled two to three inputs per leg of the circuit, compared to a single input under pure visual flight rules. Gaining this confidence took many flights to open the distance envelope. Any failure of the UAVmainframe during these distant turns would have presented significant danger to the aircraft. In the end this reliable performance enabled much more productive flights with typically more than 20 manoeuvres flown in a single flight. Figure 14.2 shows a typical flight path, with the EKF init leg and the figure eight at the beginning. The aircraft, having just 1.5m wingspan, is up to 300m away from the pilot’s position during the turns of the circuits.

The procedure to fly an input sequence is for the pilot to set the aircraft up wings level at the correct airspeed, which is called out from the ground station. Then the pilot would call out to transfer control and the groundstation operator will engage an input sequence by sending a command to the UAVmainframe via the telemetry link. The
aircraft then switches to ‘auto’ mode and executes the recorded command sequence. At the end of the sequence, the UAVmainframe will automatically return to ‘stabilised’ mode, which is followed by an audio message of the groundstation. On hearing the message the pilot knows that he is back in control. At any time during the manoeuvre, the pilot can override and switch back to ‘manual’ mode with a switch on the RC transmitter, which will hand over full control to the pilot and immediately terminates the input sequence. A new addition to the flight code is a self check, where the UAVmainframe by itself checks the attitude during the manoeuvre execution. If pre-programmed limits in pitch and roll angle are violated, the sequence is immediately terminated and the flight mode changes back to ‘stabilised’ with an audible ABORT message. The flight mode change will then quickly return the aircraft to trimmed, wings level flight and the pilot can take over for the next attempt. This feature works exceedingly well and it is a impressive sight to watch the aircraft stabilise itself from sometimes quite dangerous flight attitudes. This process usually takes less than a second and greatly relieves the pilot from saving the aircraft during the manoeuvres.

Once the battery runs low the ground station will call out a warning and the pilot will start the landing procedures, where the aircraft is brought back to the airfield and set up for approach. The pilot then switches back to ‘manual’ mode and lands the aircraft on the runway. Once landed, the aircraft taxies back to the ground station, where ground power and networking is connected and the flight data is downloaded. A quick inspection for damage is done during the download time. If conditions are good, a new battery can be installed and the aircraft is ready to fly in less than 10 minutes for a repeat of the entire procedure. Because the UAVmainframe does not have to be turned off during the battery changes, only a quick data check is required to establish flight readiness. This allows to make the most of the short periods of good and calm weather encountered and greatly improves the productivity during the flight tests.

14.3 Lessons Learned

The previous section described a perfect research flight. Anyone familiar with the matter will agree that those are very rare. There are many things that can and will go wrong. This section focuses on some of the issues specific to flight testing a small scale aircraft and presents some solutions developed over the duration of this project.

14.3.1 Weather

The biggest limitation of flying a small aircraft is the weather. Slight breezes, which are no issue for larger aircraft, must be counted as serious wind on this scale and quickly becomes an issue corrupting the flight data. Experience has shown, however, that there can be a steady wind that is very uniform. One of the cleanest flights to date had a $5 \text{ m/s}$ wind at flying altitude, which caused no issue once taken into account by the EKF. On the other hand, during the day in summer, thermals occur even at very low winds and will result in a very noisy flight. As a result, the best flying times are in the early morning or just before sunset. This is well known in the flight test community. For a remotely piloted aircraft, however, this presents another issue. The runway at Marulan airfield is
roughly in east-west direction, resulting in the sun rising and setting over one end of
the runway or the other. If the sun is too low, the pilot cannot see the aircraft properly
and take-off and landing become dangerous. After suffering a bad landing accident
due to low sun position, the acceptable flight times on a day were severely restricted to
a combination of low wind and acceptable sun position. As a result there was only a
window of about 45 min each morning and evening each day, unless the weather was
particularly good and not too hot all day. These limited time windows allowed for typically
two flights, significantly rising the pressure on system availability and made the above
mentioned productivity increases per flight mandatory. Several days of beautiful weather
were lost due to minor technical issues that could not be fixed before either the wind
started up or the sun got too low. These restrictions need to be accounted for during
planning, as it is not guaranteed to obtain any useful data from a test day.

Even on a calm day dark cloud cover can severely restrict the visibility of the aircraft
against the sky. This leads to range restrictions and the number of manoeuvres possible
per flight significantly reduces. Further, cold temperatures in winter are very good for
engine cooling but have a strong effect on battery life. A cold battery can lose up to
a third of the normally available capacity, which in turn restricts the available flight
time. The discharge characteristics of the LiPo batteries also changes, which requires
even more caution not to run the battery to low. The final issue requiring attention is
the visibility of computer displays in bright sunlight. The flight test procedures require
direct contact between pilot and ground station operator and it has proven very helpful if
the operator is able to watch the aircraft move in the sky. Hence, the usual solution for
remotely piloted operations to sit inside a darkened room watching only the displays does
not work and it is required for the groundstation to be located outside. On the other hand,
reading the ground station display is crucial and therefore the monitor used needs to be
readable outside in bright sunlight. Initially, a standard laptop computer was used and
the display was shaded with covers. Near the end of the project it then became possible
to purchase a sunlight readable screen, designed for trade shows and exhibitions. These
screens can be easily read in bright sunlight and improve work with the groundstation
immensely. This author can only recommend to invest in such a monitor if a similar
flight test project is planned. A photograph of the groundstation is shown in Figure 14.3,
with the two antenna, long range telemetry radio in the background.

Many of the restrictions due to visibility are caused by the nature of remotely piloting
a small aircraft, which also leads to other issues as discussed in the next section.

14.3.2 Remote Piloting and Aircraft Size

Remotely piloting a small aircraft during a low budget project is typically done by a pilot
standing near the airfield runway with a handheld RC transmitter, flying the aircraft
visually around his position with no direct feedback about the aircraft’s attitude or state
other than what he can see and to an extend hear. The fully featured ground control
units with visual and telemetry feedback trough radio links, as used by the military
or commercial operations are far beyond the budget capabilities of a typical university
research project.
Flying an aircraft remotely removes the direct feedback a pilot in the aircraft normally receives. It is a very different skill, as experienced by this author after taking some flying lessons in a full scale aircraft and later trying to learn to fly a remotely piloted aircraft. Relying on visual feedback only for attitude detection makes it very difficult to obtain precise trim conditions, for example, because the small, remaining errors in attitude are simply not observable from the ground. RC pilots also tend to use a single trim setting at a convenient cruise speed and adjust any divergence manually. Since this trim condition is airspeed dependant, the aircraft will drift away over time at other speeds, even if the trim was set perfectly.

This flying technique results in the above mentioned issues with visibility, limiting safe flight to certain weather conditions and distances away from the pilot’s position on the ground. There is nothing one can do about the weather limitations, but the distance requirements have been overcome to an extend by the automatic flight control system, as discussed before. The flight control system also addresses the issues of imperfect trim which improves the data considerably. Most of these limitations can potentially be completely removed by fully autonomous flight, which is an interesting project for future work.

A unique problem faced during this project is a direct result of the remote piloting technique in combination with the automatic flight controls that are required for successful test flights. As the input sequences are open-loop to determine the unaltered dynamics of the aircraft, how does one do the transition from the closed loop controlled flight into the open loop manoeuvre, while maintaining trimmed flight? Initially one might think that this is quite easy, just switch of the flight controllers and hold the control surfaces at their last position. This approach, however, will fail in all but absolutely dead calm conditions (which do not exist in this author’s experience). Consider a normal flight

Figure 14.3: Flight test ground station with telemetry radio and sunlight readable screen
under closed loop control, where the flight controllers move the control surfaces to keep straight and level flight in an unsteady atmosphere. Depending on the size of the gust disturbances, the control surface may at any time be deflected considerably to reduce an attitude error. If at this moment the command to transition to open loop is received and the controls are simply locked to their current position, this will result in a potentially quick and dangerous departure from the trim condition. It is therefore necessary to perform some kind of blending between the closed loop control and the open loop surface trim position. The UAVmainframe does this by time averaging the closed loop control commands, which during random turbulence are assumed to move the control surface
around a mean value that represents the trim condition. The difficulty lies in determining the time taken for this averaging. If the time is too short, the determined mean might still contain some element of the control response to a gust and the open loop trim will be wrong. On the other hand, if the time is too long, the ability of quickly commanding an input sequence after a turn, for example, will result in the control commands for the turn to become part of the determined mean value and the resulting open loop trim position will be wrong, as shown in Figure 14.5. Here the aircraft is coming out of a steep turn with the elevator up for extra lift. During levelling out the control system actually commands elevator down from the trim condition most of the time to obtain level flight. Since the averaging time in this case was 3 seconds, the elevator motion during the transition is still part of the average. As soon as the input sequence is started, the aircraft pitches down away from the trim condition. Lots of testing during various turbulence levels were required to optimise this time span to yield a good open loop trim position most of the time. Values between 1-2 seconds (representing 100-200 samples), depending on the turbulence level, have worked well during most conditions.

14.3.3 Airframe Vibrations

A general problem for any kind of data acquisition in flight are airframe vibrations caused by the propulsion system or aerodynamic effects like flow separation. For small aircraft
this becomes especially problematic, because the propulsion systems represent a large mass relative to the overall airframe weight and the motors and propellers typically turn very fast. The low airframe weight, together with the typical construction methods on this scale results in very low damping of such vibrations. On top of that, as it was the case for this test aircraft, the structural modes of the airframe become very slow and overlap with the frequencies of the aerodynamic modes, making the system ID of these modes difficult. It is therefore required to carefully balance and isolate the propulsion system to reduce these vibrations. Aerodynamic noise adds to the overall airframe noise due to the separated flow on the landing gear, the propeller slipstream impacting the airframe and other flow disturbances. Nothing really can be done about that part of the noise but experience from glide tests has also shown that the aerodynamic noise is much less problematic than the propulsion system vibrations.

The first step to reduce the motor vibrations is to carefully balance the propeller. This is typically done by mounting the propeller in a shaft supported by very low friction bearings. The heavier blade of the prop will rotate down due to gravity. The easiest way of applying a counter weight on the other blade is to use a strip of heavy tape as shown in Figure 14.6. By trial and error, the position and amount of tape can be optimised such that the propeller will stay in a horizontal position on the balancing shaft. This method is used frequently for these small propellers but due to the remaining friction in the shaft bearings and the tiny masses involved (the 14” propeller used on this aircraft weights less than 30g), there will be a remaining imbalance that cannot be improved upon, unless some specialised equipment is used.

Initially, the motor was mounted directly onto the firewall of the test aircraft without any vibration isolators, as it is typical practice for this scale of aircraft. Even with a balanced propeller, the resulting airframe vibrations were still reading up to ±1g on the accelerometer, depending on the motor RPM. This was clearly unacceptable, given that the expected accelerations due to the aircraft motion are only 0.5g or less. The motor had to be insulated from the airframe structure. This is actually quite a difficult task, because it requires to design a mount that still transmits the thrust force and torque but acts as a low pass filter to filter out vibrations. The mount also has to be structurally sound because a failure of the motor mount will be very dangerous. Since this author did not have any experience in this field (and not many people have on this small scale) a custom made vibration mount was purchased. This mount was tuned to the exact motor specifications and resulted in a fivefold reduction of the vibrations on the accelerometers. However, as Figure 14.7(a) shows, there is still about 200mg noise on the y axis accelerometer at certain RPM. This is still a problem during lateral manoeuvres, where the sideforce typically only results in a 100mg acceleration.

The next step is then to balance the motor itself. One might expect that the motor would come balanced from the factory, but since these are hobby parts (as discussed in appendix F) the results vary by a large amount between identical motors. An easy way of balancing a brushless motor is to use cable ties as shown in Figure 14.6. These can be fitted around the motor bell and rotated until the best balance is achieved. For this project, a special motor mount with an integrated accelerometer was built to be able to
judge the changes in motor vibrations with variation in cable tie orientation. Once the optimal position is found, the cable ties can be glued into place.

The final step is to dynamically balance the propeller-motor combination. With the motor mounted on the instrumented mount from above, the propeller is rotated with respect to the motor in small steps until the position is found where the remaining imbalance of propeller and motor cancel out as good as possible. The orientation of the propeller to the motor is then marked on both parts to be able to reproduce the correct position. With this technique, the accelerations in the test motor mount could
be reduced from 300mg to about 30mg. Together with the vibration mount on the plane, this results in no distinguishable engine vibrations on the sensors as shown in Figure 14.7(b) for a similar RPM range. The downside of this procedure is that each motor-propeller combination requires separate balancing, which is quite time consuming. For this project, two motors with three propellers each were balanced to be able to switch parts quickly in case of damage.

14.4 Summary

Flight operations on this small scale present their own, unique challenges. The limitations due to weather and remote piloting have been discussed and solutions developed during the course of this project were presented. The requirement of a flight control system was identified and its effectiveness in reducing pilot workload and improvement of the trim attitude and overall productiveness were discussed and will be further demonstrated during the data processing stage.

Flight testing one’s own flight hard-and software is liberating and very scary at the same time. One always waits for the last bug in the system to present itself in an inconvenient manner. Yet, the performance of the UAVmainframe was flawless during all flight tests, with no issues ever presenting a danger to the aircraft.

The next chapters will describe the data processing and parameter ID of the flight data recorded during the flight operations. Even though it is presented here in a linear fashion, this is of course a highly iterative process with many trials and repeats until all works correctly during good weather.
15. Flight Data Preparation

After downloading the raw flight data from the aircraft after landing, the data needs to be processed and evaluated before it can be used to run the parameter identification. This process is done with the EKF. The theory and formulation of the EKF was discussed in section 5.2. In this chapter it will be put to use to evaluate and correct the raw flight data.

15.1 Data Compatibility Check

The quality of the flight data recorded by the UAVmainframe is the subject of this section. Several results of the EKF runs can be used to judge how well the sensor system operates and whether the recorded data is consistent. This is called data compatibility check. The result of this check will determine the level of confidence in the measured data actually representing the true motion of the aircraft during flight. The parameter identification that follows in the next chapter will then show how well the aerodynamic models of the previous chapters will match the aircraft’s response to control inputs during flight.

As discussed in the EKF development chapter, its model of the aircraft does not rely on any aerodynamic parameters, but solely uses the sensor measurements of the UAVmainframe to drive a kinematic model of a rigid body undergoing six degree of freedom motion. Based on the measurement inputs (the derivative of the translations accelerations and the rotational accelerations), the EKF integrates the equations of motion of the rigid body and then uses further sensor data to refine or correct its model of the rigid body motion. Since the UAVmainframe uses many sensors in different locations of the aircraft, the kinematic relationships between the rigid body motion and the sensor readings (the measurement equations) can be used to evaluate the measurements. If the predicted reading of the EKF and the actual reading agree well, then there is a high
chance of the EKF predicting the correct motion, based on all the other measurements, and that the sensor readings in question are of good quality. If there are discrepancies, the statistical process of the EKF will output a corrected dataset, which fits the process model optimally in a least squares sense.

In order to judge the quality of the flight data and the performance of the EKF, some metrics were defined. Some of these are based on the results of the EKF runs, like the final confidence of the algorithm in the state estimates and the measurement residuals, as well as the quality of the estimates for the error states (These should be constant throughout a flight). Others are based on the ability of the filter to predict known bias and scale factors and the filter performance in turbulent air, where the noise levels of the sensor measurements increase. In particular, the following quality metrics were used extensively for the tuning of the EKF:

**Known bias and scale factors:** The bias values of the gyroscopes can be determined from the start of the data recording, where the aircraft sits on the ground, waiting to take off. The bias values are fairly constant and are not expected to change significantly during a flight. The initial bias values can then be compared to the estimates of the EKF for the gyro bias states, where a close match is expected for good data. Similarly, the calibration factor for the angle of attack change due to the wing upwash are known from the calibration procedure in section 6.4. These should match the EKF estimate for the angle of attack scale factor. The bias values for the air vanes should also be predicted as constant but their values depend on the assembly of the airdata probe at the airfield and are not known beforehand.

**Plausibility of the wind estimates:** The filter estimates for the wind are expected to be fairly constant during the flight duration of typically 10-12 minutes and the resulting wind direction should agree with the ground measurements, especially for stronger winds. That said, there have been several flights where the wind at an altitude of 100m above the airfield was considerably different than on the ground. Since there is no way of knowing the true wind speed and direction, the filter estimates have to be taken for granted. In general, the EKF appears to work best if the wind estimates are allowed to fluctuate somewhat (by increasing the process noise for the wind states). Engineering judgement was used to evaluate the overall wind estimates of the EKF.

**State variances:** The diagonal of the error covariance matrix $P$ contains the variance, or the level of confidence, of the filter estimates for each state. For a well working EKF, this variance is expected to be better than the variance of a particular sensor measuring that state. That means that the filter is more confident in its state estimate of velocity for example, which is based on multiple sensor readings, than the GPS receiver by itself. The state variances should also be fairly constant during a flight.

**Filter residual and sensor output predictions:** The filter residuals are the difference between the sensor measurements and the predicted measurements based on the state of the aircraft. A well calibrated sensor system will have only small residuals, that is each sensor measures accurately and is aligned well with the rest of the
Further, the filter can be used to predict a sensor reading that is not used for the correction step. Comparing this estimate with the actual reading can also yield some answers on the quality of the state estimates and the sensor performance.

**Filter stability in turbulence:** With increasing turbulence the level of noise on most sensors increases, especially on the high order sensors like accelerometers. A well functioning EKF should be able to absorb this extra noise (or uncertainty) without much deterioration of the result. This will work up to a point where the filter will start to diverge. On very noisy data sets, higher levels of process noise may be necessary for filter stability. The extra uncertainty also limits the ability of the EKF to estimate the error states. It has been found that the filter works best if for noisy flights the error state estimation is turned off and all those error states are initialised with the best available estimate from a previous flight. With that strategy no filter instability has occurred on any flight data sets evaluated.

These metrics will now be used to evaluate selected flight data sets and the results are presented in the next section, together with some remarks on properties and issues of those data sets.

### 15.2 Properties of Selected Flight Data Sets

Data from two flight test sessions were processed for further use at this stage. The first set is from November 2014 and is named with the prefix 14fxx, where xx is the flight number. The second set is the most recent flight from April 2016, labelled 16fxx. This section discusses some of the features and issues of these data sets. The EKF formulation has proven to be relatively insensitive with respect to the tuning parameters such that for all reasonably clean flight data the same set of parameters can be used. These will be discussed in this section. For the very turbulent flights the EKF still works acceptably with these parameters, but loses the capability of estimating the error states as mentioned before. In any case, for these noisy flights the intricate corrections of the EKF to the flight data are fully masked out by the aerodynamic noise and it is better to just use the filter error method on the raw data if one really has to use this noisy data. For this thesis, only reasonably clean data is considered and the analysis of the noisy data is left for a future project.

#### 15.2.1 EKF Tuning and Common Remarks

The EKF requires a set of initial conditions of the state vector and the magnetic reference vector to start up. The tuning of the filter requires the entries in the initial error covariance matrix \( P_0 \), in the process noise matrix \( Q \) and the entries in the measurement noise matrix \( R \). Some general remarks on how to generate these values were already given in section 5.2.8, and in this section the system specific methods are presented.

The input- and measurement data channels for the EKF are the same for all flights except for the magnetometers, which did not work correctly during the 14fxx flights as discussed before. For those flights, the magnetometer measurement was replaced with the attitude estimate of the reference IMU. For the 16fxx flights the wing tip magnetometer
was used, after applying the calibrations from section 6.3.3. The acceleration and rotation rate input measurements come from the raw measurements of the reference IMU. The airdata is the raw data from the airdata probe, with all calibrations for off-CG location of the probe and wing upwash being done by the EKF. GPS position and velocity data is supplied in the ECEF frame at 50 Hz and is fused at every second time step by the EKF.

During initial testing, it was established that several states of the EKF process model are not clearly observable or had minimal influence on the result. These states were disabled for all subsequent runs. The vertical wind is one of these states. It is highly correlated with the angle of attack measurement and can be assumed small in any case for weather conditions suitable for small aircraft flight. The airspeed scale factor is another non-observable state, which has been shown to be negligible during the pitot tube calibration in section 6.4. Accelerometer and gyroscope scale factors appear to be insignificant, too. The accelerometers were calibrated sufficiently and to observe gyroscope scale factors, higher order dynamics are required. And finally, the GPS velocity measurement bias factors were repeatedly predicted close to zero, so those states were disabled as well to remove unwanted freedom from the process model.

**Initial Conditions**

The EKF requires several initial conditions that depend on the location of the flight. These are the reference vectors for gravity and the magnetic field. While the vector of gravitational acceleration can be assumed to be \( g = [0 0 9.80665]^T \) without introducing much error (unless one flies in an alpine region or over the ocean), the magnetic field vector variations are significant and need to be determined correctly for good results. Models for the magnetic field strength and orientation variation exist [126] and allow to predict the magnetic reference vector prior to flight. This website has been used to determined the magnetic field for Marulan, NSW, Australia as

|        | X   | Y   | Z   | |B| |
|--------|-----|-----|-----|---|---|
| Marulan| 23.692 µT | 5.199 µT | -52.230 µT | 57.588 µT |

The initial conditions for the motion states are determined from the start of the flight data set. Most error states are initialised as zero, unless the value is known. This is the case for the angle of attack scale factor, which can be initialised with the calibration factor determined in section 6.4. Due to the multi-pass feature of the filter, the initial values for the error states are not that important, because the first run of the filter will produce good estimates to start the second run in most cases. The wind states are initialised to the values of a previous test run and need to be reasonably close to the actually values for optimal performance at the beginning of the dataset.

**Error Covariance Matrix** \( P_0 \)

The initial error covariance matrix \( P_0 \) determines how much the filter relies on the sensor measurement during initial convergence. Larger values increase sensor use. Settings that are too large can cause the filter to diverge. If a state has a zero entry in \( P \), the filter assumes the state is perfectly known, and the filter will not change it. This can be used to turn off the estimation of certain states during tuning, and to set a state to a
constant value, as done with all the sensor error states during the final runs. For the UAVmainframe flight data, values of 0.1 for the motion states and 0.0001 for the bias and scale factor states have worked well.

**Process noise $Q$**

The entries on the main diagonal of the process noise matrix $Q$ represent the square root of how much the state is allowed to change per timestep, effectively low pass filtering the state. Therefore the tuning process involves finding a level of process noise for each state which leads to sufficient noise smoothing of the data, while preserving the desired aircraft responses due to control inputs. A value that is too small will lead to smoothing of the fast parts of the airframe short period and roll mode responses. On the other hand, for example, a large value for the wind states will lead to non-physical wind estimates, as the filter ‘unloads’ all the non-modelled noise from the airdata system into the wind states to enforce a kinematically compatible dataset.

Using trial and error and the quality metrics introduced above, the initial process noise values for the motion states were determined as listed in Table 15.1. The gyroscopes use the noise levels obtained from a static ground run, while the noise for the accelerometers was doubled to allow for any remaining calibration errors. The values for the position and wind states seem to work well, although no physical derivation is possible. All enabled bias and scale factor states were set to $1 \times 10^{-8}$ to obtain highly smoothed estimates, as these states are expected to be constant during a flight.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.0002</td>
<td>0.0001</td>
<td>0.002</td>
<td>0.00132</td>
<td>0.00133</td>
<td>0.00218</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>State</th>
<th>Pn [m]</th>
<th>Pe [m]</th>
<th>Pd [m]</th>
<th>Wn [m/s]</th>
<th>We [m/s]</th>
<th>Wd [m/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 15.1: Initial process noise variances for the motion states

**Measurement Noise $R$**

The measurement covariance matrix was initialised with the values listed in Table 15.2 and remains constant during the EKF run. The variances for the GPS measurements of position and velocity are the values from the data sheet of the receiver module. The airdata and magnetometer noise values were determined from the flight data and sensor resolution considerations. For the older data sets, where no magnetometers were available, the attitude estimation accuracy from the data sheet of the reference INS were used for the uncertainty values listed in the table.
## Chapter 15. Flight Data Preparation

<table>
<thead>
<tr>
<th>Meas.</th>
<th>Pn [m]</th>
<th>Pe [m]</th>
<th>Pd [m]</th>
<th>u [m/s]</th>
<th>v [m/s]</th>
<th>w [m/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>9</td>
<td>9</td>
<td>36</td>
<td>0.0625</td>
<td>0.06251</td>
<td>0.0625</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Meas.</th>
<th>$V_{air}$ [m/s]</th>
<th>AoA [deg]</th>
<th>Sideslip [deg]</th>
<th>Alt [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.125$^2$</td>
<td>0.25$^2$</td>
<td>0.25$^2$</td>
<td>0.25</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.025$^2$</td>
<td>0.025$^2$</td>
<td>0.025$^2$</td>
<td>0.04</td>
<td>0.04</td>
<td>1</td>
</tr>
</tbody>
</table>

### Table 15.2: Initial measurement noise variances

#### 15.2.2 April 2016 EKF Sample Results

The flights in April 2016 were performed with the most advanced flight software running on the UAVmainframe. Prior to these flights, it gained a new control architecture with added pitch control and the input sequence abort feature, which greatly improved the trim condition at the start of the manoeuvre and the safety of the flights. The ground station setup was also significantly improved, with the addition of the sunlight readable screen and a better layout of the Qgroundcontrol software. This enabled better awareness and control by the flight test engineer during the flight. All new input sequences were tested in the wind tunnel before the flight which improved the data considerably, as will be shown later. On the downside, the flights took place during several days of windy and wet weather, which left only two data sets usable for further analysis. These were flown during a brief calm period just before the sun dropped too low in the afternoon and represent the best data recorded yet. The first flight of the two (16f5) is the only one during the entire project with zero wind for the first half of the flight. In the second half the wind started to pick up again, which is nicely estimated by the EKF. Interestingly, this flight was quite bumpy despite the calm conditions, showing the difference between wind and turbulence due to rising air. The second flight, 16f6, which took place about 15 minutes later, has a reasonably constant wind of 3m/s and very low turbulence levels. This flight data is used as the reference to illustrate the accuracy that was achieved with the UAVmainframe. The full EKF results span over 100 plots, so only the most interesting ones were selected for this section.

### Magnetometer residual

Figure 15.1 shows the calibrated sensor data of the wing tip magnetometer against the EKF prediction, as well as the filter residual of the magnetometer. The fit is very good, confirming the successful calibration of the sensor in section 6.3.3. This shows that that a magnetometer, if treated correctly, is actually a good attitude sensor. Including the magnetometer measurements in the filter greatly improves the variance on the attitude estimate because otherwise the filter would otherwise mainly rely on the integration of the gyroscopes alone, which can be biased due to the unknown integration constant. This is especially important for the low grade MEMS sensors used in the UAVmainframe.
15.2 Properties of Selected Flight Data Sets

**Figure 15.1: Magnetometer**

**Airdata bias and scale factors**

Figure 15.2 shows the estimated airdata readings vs. the actual sensor readings. Constant bias values can be clearly seen for the airspeed and the inflow angles, while the altitude appears to match well. All biases appear to be constant as expected.

This is confirmed in Figure 15.3, which shows the bias and scale factors for the

**Figure 15.2: EKF airdata estimates vs. raw data**
airdata measurements that were estimated by the EKF. Altitude bias and sideslip scale factors are estimated as zero and one, respectively, and therefore could be turned off if desired. The bias values for the airspeed and the two air vanes are estimated as nearly constant, with only a small amount of noise in the angle of attack bias. This is probably caused by uncertainties in the wind estimates. The angle of attack scale factor estimate is nearly perfectly stable at 1.51. The scale factor is the inverse of the calibration factor for the wing upwash from section 6.4. The value estimated there was 0.655. The EKF estimate from the flight data is 0.663, which amounts to a difference of just 1.2%. This result therefore confirms the validity of the PanAir free air prediction of the angle of attack vane calibration of section 6.4. As mentioned before, on noisier data sets the ability of the EKF to observe this scale factor diminishes. But since its value was confirmed during this flight, it is now possible without loss of confidence to set this state to the calibrated value and turn the estimation off.

It should also be noted that all these bias and scale factors for the airdata probe appear only in the measurement equations of the EKF. They have no direct influence on the final result, but they improve the overall system uncertainty by allowing the filter to better match the measured data. Without the bias and scale factors, the overall process noise levels would be higher, which can lead to a less optimal filter output.

Velocity and Wind estimates

The EKF wind and velocity estimates are shown in Figs. 15.4 and 15.5, respectively. The wind speed during this flight was about 2.5m/s from the south-east. One peculiar feature of most wind estimates from this EKF is the variation in direction vs. the variation in speed. One would expect that the wind direction is usually constant over a time frame of 10 min, while the wind speed might vary with gusts during this time frame. Quite the opposite is predicted by the EKF for this data set and also for most other flights. The
wind direction appears to fluctuate, while the speed is nearly constant. Much time was spent on investigating this behaviour, but no conclusive answer could be found, simply because the true wind at the time of flight is unknown and there is no way of measuring it without installing weather stations at the actual flight altitude. The airfield is located in hilly terrain, and the wind direction changes might as well be caused by the topography of the land. Therefore, the very high confidence of the algorithm in the wind estimates, as shown in the figure, must be taken for granted.

![Figure 15.4: EKF wind estimates](image)

Figure 15.4: EKF wind estimates

![Figure 15.5: EKF air- and ground speed estimates](image)

Figure 15.5: EKF air- and ground speed estimates
The EKF body axes velocities and the corresponding airspeed components also appear to be estimated with very high confidence, as shown in the confidence intervals of Figure 15.5. All speeds are expected to be accurate within 0.02m/s or better, which is equal to 0.1% of the typical cruise speed. This kind of accuracy should be sufficient for high quality parameter estimates, and is a proof of the accuracy of the UAVmainframe.

**Inertial bias estimates**

Figure 15.6 show the results for the bias estimates of the inertial sensors. All biases appear to be well observable and are estimated as (nearly) constant across the entire flight. The position biases are small, but the filter returns lower overall uncertainties if they remain enabled. The Magnetometer biases account for the remaining calibration error, while the accelerometer biases are mainly caused by temperature changes from the calibration conditions. The gyroscope bias estimates closely match the data from a ground run, and are relatively stable between flights.

**Dataset 16f6 Summary**

The results of the EKF run presented in the previous sections clearly show that the filter is working very well and that the measured data from the UAVmainframe is of high quality. The filter confidence levels for all state estimates (including the ones not shown in these examples) are very high, and therefore the design goals for the UAVmainframe in terms of accuracy and consistency have been met. The next section will briefly demonstrate the second example flight, 14f9, which produces similar data quality during much windier conditions.
15.2.3 November 2014 EKF Sample Results

Figures 15.7 and 15.8 show the velocity and wind estimates for flight 14f9. The wind is much stronger during this flight with an average speed of 5.5 m/s from the north-west. Incidentally, this was a flight, where it was dead calm on the ground, which shows that wind measurements on the ground will not be useful for a wind estimate at flight altitude. Despite the higher wind speed, the confidence intervals on the state estimates are of similar magnitude as for the previous flight. This shows that a higher wind speed is not a problem and does not deteriorate the EKF performance.

Figure 15.7: 14f9 EKF velocity estimates

Figure 15.8: 14f9 EKF wind estimates
Figure 15.9: 14f9 airdata

Figure 15.9 show the airdata estimates vs. the raw sensor measurements. Immediately visible is the higher noise level in the air vanes, especially in angle of attack, due to the higher winds. The example nicely shows the ability of the EKF to filter this noise and to produce an angle of attack output with significantly lower noise levels. It is also noteworthy that the noise levels increase during the flight, which shows that even during a 10 minute flight the atmospheric conditions cannot be assumed constant. A zero bias on the sideslip vane is estimated, showing that for this flight the vane was installed perfectly on the sensor. The bias value on the airspeed is constant as before.

Figure 15.10 shows the attitude estimate of the EKF vs. the data from the reference INS. Its estimate is based on its internal sensors and a separate GPS receiver, but without airdata. The Figure shows differences of up to ±1 deg in the attitude estimates between the INS and the UAVmainframe EKF. The confidence intervals for the attitude estimates are difficult to derive from the quaternion formulation, such that a direct comparison of the data is not possible. But, based on the quality of all other estimates presented in this chapter, this author is convinced that the UAVmainframe EKF produces a higher quality result than the US$2000 INS, which is listed as being accurate within 0.25 degrees for the attitude estimates.
15.3 Summary

After the EKF treatment, the flight data is ready for the parameter estimation. The EKF and the UAV mainframe have demonstrated excellent data quality. All the quality metrics show that the motion of the aircraft was recorded accurately and consistently at a level not previously seen (to this author’s knowledge) in an aircraft of this size. Some flight data might be noisy due to atmospheric turbulence, but, and this is important for the next steps, the aircraft’s motion and response to the turbulence has been measured and recorded accurately (within the limitations of the underlying assumptions of a rigid airframe) and it is now a matter of the parameter ID methods to deal with the remaining noise, which is the topic of the next chapter.
16. Longitudinal Results

16.1 Introduction

The system- and parameter ID for the longitudinal motion in flight has been surprisingly difficult. While the standard models for the short period mode fit the data well, it was much more difficult to obtain a good match between expected and measured parameters in these models. After many trials and flight tests to exclude all possible error sources, the issue was finally traced to the properties of the motion itself. Firstly, the inertial properties of the aircraft are changed by the added mass. Hence, the pitch inertia including the added mass component from Table 13.17 must be used. Secondly, for this small aircraft some of the parameters in the pitching moment equation become highly correlated and cannot be identified simultaneously. Any attempt in doing so will lead to the large errors observed during the project. In order to demonstrate the issues, the chapter starts with attempts of identifying the parameters in the longitudinal model without the added mass contributions and ignoring the correlation problem. By doing this, it can be clearly shown how the results improve with the addition of the two phenomena.

This chapter is structured similarly to the dynamic wind tunnel analysis, starting with the model structure used, then describing the input sequences flown, and finally presenting the findings in full detail. The data that is analysed in this chapter is the output data from the EKF, with all corrections and calibrations applied. Only data from two particularly clean flights will be analysed here to clearly show that the above mentioned properties are not the result of noisy flight data, but a fundamental characteristic of the longitudinal motion of this small aircraft.
Chapter 16. Longitudinal Results

16.2 Model Structure

For the parameter ID of the flight short period mode, a model structure using linearisation through measured data presented by Iliff and Maine [38] has been found to give more consistent results than the traditional linear perturbation model of the short period mode given in reference [22]. The linearisation through measured data allows for a larger range of flight conditions during the manoeuvre and does not require perfect trim. This class of models is therefore ideally suited for the flight data from the small remotely piloted vehicle tested for this thesis, where, as discussed before, achieving perfect trim is very difficult. In addition, this modelling technique also allows for some lateral motion, which is always present due to the lateral flight stabilisation. The model structure of this longitudinal model is essentially an augmented version of the short period mode approximation, using the same states $\dot{\alpha}$ and $q$. The aerodynamic model is also the same, with linear expansions for $C_L$ and $C_m$. The model used here is valid only for low angles of attack and assumes that $C_L \approx C_N$. The parameters $C_{L_0}$ and $C_{m_0}$ act as the bias parameters but have a physical meaning for this model. They are the lift and moment coefficient offsets at zero angle of attack and can be compared to the wind tunnel test results. This adds two more parameter that can be used to validate the results, although this was not used for this project because these bias parameters require a much more detailed calibration of the wind tunnel balance and the elevator setting than what was necessary for the stability and control derivatives. The results for these bias parameters will, however, be included in the results, where they serve as an indication of the repeatability of the experiments. The input data into this model is therefore not perturbation quantities but the full measurements provided by the EKF. The $\dot{\alpha}$ contributions are included in the moment derivatives in a similar manner as in the short period approximation. Hence these parameters carry a dash to distinguish them from the ‘true’ moment derivatives.

Theorem 16.2.1 — Linearised longitudinal model using measured data. (called LLM model from now on)

State rates:

\[ \dot{\alpha} = -\frac{\bar{q}S}{mV_{\text{air}}} (C_L + b_\alpha) + q + \frac{g}{V_{\text{air}}} (\cos \phi \cos \theta \cos \alpha_m + \sin \theta \sin \alpha_m) \quad (16.1) \]

\[ \dot{q} = \frac{\bar{q}Sc}{I_y} C_m \quad (16.2) \]

\[ [\dot{\theta} = q] \quad (16.3) \]

Measurements:

\[ \alpha = \alpha \quad q = q \quad a_n = \frac{\bar{q}S}{mg} C_n \quad (16.4) \]
with the aerodynamic expansions:

\[ C_L \approx C_{L0} + C_{L\alpha} \alpha + C_{L\delta e} \delta_e \]  \hspace{1cm} (16.5)

\[ C_m = C'_{m0} + C'_{m\alpha} \alpha + C'_{mq} \frac{q_c}{V_{air}} + C'_{md} \delta e \]  \hspace{1cm} (16.6)

with potentially identifiable parameters

\[ C_{L0}, C_{L\alpha}, C_{L\delta e}, C'_{m0}, C'_{m\alpha}, C'_{mq} \text{ and } C'_{md} \]

where the moment derivatives contain the contributions of the \( \dot{\alpha} \) terms as discussed in the text. The variables \( \dot{q}, V, \phi, \theta \) and \( \alpha_m \) are measured data applied on a per sample basis and are not averaged.

The \( \dot{\theta} \) equation is used only for the filter error method, where it acts as a stabilising element, according to reference [38]. The output error method uses only the first two state equations.

Reference [38] only states that the pitching moment derivatives contain the \( \dot{\alpha} \) contributions but does not say how these might be related to derivatives measured in a wind tunnel for example. In order to compare the results to the reference data for the aircraft, this information is required for this project. By inspection of the model structure and comparing it to the short period approximation model, it appears as if the additional term containing the gravitational acceleration has the form of a (pseudo) input to the \( \dot{\alpha} \) equation and does not affect the aerodynamic model at all. Therefore, it may be assumed that the contributions of \( \dot{\alpha} \) to the moment derivatives in the LLM model is similar to the short period approximation. In order to confirm this assumption, a response to a real elevator input was simulated in a non-linear flight simulation, using the reference data for the aircraft in the model definition. The simulation calculates \( \dot{\alpha} \) separately and takes in the values for \( C'_{m\alpha}, C_{mq}, C'_{mq} \) and \( C_{md} \) as listed in Table 16.1.

Table 16.1: Longitudinal parameters used for the simulation

<table>
<thead>
<tr>
<th>Param.</th>
<th>( C_{L0} )</th>
<th>( C_{L\alpha} )</th>
<th>( C_{L\delta e} )</th>
<th>( C_{mq} )</th>
<th>( C_{m0} )</th>
<th>( C_{m\alpha} )</th>
<th>( C_{md} )</th>
<th>( C_{mq} )</th>
<th>( C_{md} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.3328</td>
<td>4.5400</td>
<td>0</td>
<td>0</td>
<td>-0.5110</td>
<td>-0.0690</td>
<td>-0.7680</td>
<td>-4.0598</td>
<td>-8.4402</td>
</tr>
</tbody>
</table>

The simulated time series were used to identify the parameters in the standard short period approximation and in the LLM model. The results are listed in Tables 16.2 and 16.3, respectively. The expected values for the \( C'_{m\alpha} \) derivatives were calculated in Section VI and are included in the tables.

The estimates of the two model structures agree well, which shows that the assumption about the \( \dot{\alpha} \) contribution to the moment derivatives is correct and the same form as with the short period approximation can be used. The tables also show that the linearised longitudinal model delivers better estimates than the short period model, even though the confidence intervals are somewhat larger. Both match the simulated response well,
Table 16.2: Parameter estimates using the Short period approximation model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>4.54</td>
<td>4.826</td>
<td>$\pm 0.128$ (0.033)</td>
<td>2.66 (0.69)</td>
<td>[4.569 5.082]</td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-0.626</td>
<td>-0.608</td>
<td>$\pm 0.006$ (0.004)</td>
<td>0.95 (0.73)</td>
<td>[-0.619 -0.596]</td>
</tr>
<tr>
<td>$C_{m_q}$</td>
<td>-12.099</td>
<td>-12.398</td>
<td>$\pm 0.195$ (0.177)</td>
<td>1.58 (1.43)</td>
<td>[-12.788 -12.007]</td>
</tr>
<tr>
<td>$C_{m_{\delta e}}$</td>
<td>-1.159</td>
<td>-1.162</td>
<td>$\pm 0.010$ (0.009)</td>
<td>0.85 (0.77)</td>
<td>[-1.182 -1.142]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 99.27, R2 for Output 2: 98.75, R2 for Output 3: 96.67

Table 16.3: Parameter estimates using the LLM model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>4.54</td>
<td>4.603</td>
<td>$\pm 0.018$ (0.008)</td>
<td>0.38 (0.17)</td>
<td>[4.568 4.639]</td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-0.626</td>
<td>-0.628</td>
<td>$\pm 0.009$ (0.001)</td>
<td>1.41 (0.18)</td>
<td>[-0.645 -0.610]</td>
</tr>
<tr>
<td>$C_{m_q}$</td>
<td>-12.099</td>
<td>-11.978</td>
<td>$\pm 0.363$ (0.041)</td>
<td>3.03 (0.34)</td>
<td>[-12.705 -11.251]</td>
</tr>
<tr>
<td>$C_{m_{\delta e}}$</td>
<td>-1.159</td>
<td>-1.142</td>
<td>$\pm 0.015$ (0.002)</td>
<td>1.31 (0.18)</td>
<td>[-1.172 -1.113]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 99.87, R2 for Output 2: 99.95, R2 for Output 3: 99.90

as shown in the $R^2$ values, with the linearised longitudinal model being slightly better. This observation has been confirmed during many comparisons of the two models with real flight data of varying levels of turbulence and trim conditions, and therefore from now on only the LLM model will be used. As a first indication that the response of this small aircraft to an elevator input will be different from a full scale aircraft shows in a correlation of 88% between $C_{m_q}$ and $C_{m_{\delta e}}$ even for this perfectly clean simulator data. The parameters, however, are still identified correctly, so this is not a concern yet. Another indication is that $C_{L_{\delta e}}$ is not identifiable from the response. There is no significant difference in the model fit if this parameter is set to zero or its reference value from the wind tunnel and any attempts of identifying it from the data yield very small results with large uncertainties, a clear indication of non-observability of this value. The flight data analysis will show that this is true for all longitudinal data analysed for this project.

### 16.3 Input Design and Verification

The initial flight tests used the same 3-2-1-1 input sequence as the wind tunnel experiments. Figure 16.1 shows a particularly clean example obtained on one of these rare flights with very low turbulence. The size of the input is quite large, because at the
time it was thought it would be best to maximise the signal to noise ratio, especially for more noisy data. The final inputs, which are discussed later on, are smaller to avoid any non-linearities. The fit of the LLM model to the data is excellent, with barely any difference visible between the flight data and the model response. The control derivative $C_{L_{\delta e}}$ is still not identifiable and does not appear to have any significant effect on the pitching motion. The model fit is actually best if this parameter is turned off and set to zero instead of setting it to the value from the wind tunnel. The results for the moment parameters (using the true inertia of the aircraft) are listed in Table 16.4. They show large differences to the expected values, with errors of $\approx 30\%$ for $C_{m_{\alpha}}$ and $C_{m_{q}}$ and 17%
Table 16.4: OE results for the input sequence of Figure 16.1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>-</td>
<td>0.340</td>
<td>± 0.002 (0.001)</td>
<td>0.72 (0.30)</td>
<td>[0.335 0.345]</td>
</tr>
<tr>
<td>$C_{L_a}$</td>
<td>4.54</td>
<td>4.660</td>
<td>± 0.091 (0.023)</td>
<td>1.96 (0.49)</td>
<td>[4.478 4.843]</td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-</td>
<td>-0.025</td>
<td>± 0.001 (0.000)</td>
<td>2.41 (0.80)</td>
<td>[-0.026 -0.024]</td>
</tr>
<tr>
<td>$C'_{m_0}$</td>
<td>-0.626</td>
<td>-0.429</td>
<td>± 0.021 (0.004)</td>
<td>4.79 (0.95)</td>
<td>[-0.470 -0.388]</td>
</tr>
<tr>
<td>$C'_{m_q}$</td>
<td>-12.099</td>
<td>-17.351</td>
<td>± 0.610 (0.175)</td>
<td>3.52 (1.01)</td>
<td>[-18.572 -16.130]</td>
</tr>
<tr>
<td>$C'_{m_d}$</td>
<td>-1.159</td>
<td>-1.296</td>
<td>± 0.028 (0.008)</td>
<td>2.17 (0.62)</td>
<td>[-1.353 -1.240]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 99.25, R2 for Output 2: 99.72, R2 for Output 3: 99.74

for $C_{m_q}$. The parameter correlation between $C_{m_q}$ and $C_{m_d}$ is now 93%. But more about that later, for the input design stage it is only important to show that the short period mode has been adequately excited by the elevator sequence and that the model fits the data well. Both criteria are true for the single 3-2-1-1 input, as shown in Figure 16.1.

This input sequence was flown in many different variations by inverting it in time and direction, similarly to the wind tunnel tests. However, these single inputs do not contain a large amount of information and are easily distorted by turbulence. They are also not the best use of the finite flight time, since only 2-3 manoeuvres can be flown per circuit. In order to improve on these shortcomings, a longer input sequence was devised. The new, and final input sequence is shown in Figure 16.2, again with a very clean example. Table 16.5 lists the corresponding parameter results.

The sequence simply consists of two single elevator 3-2-1-1 inputs in short succession, with sufficient spacing to ensure the response of the first input is fully damped out. The

Table 16.5: OE results for the input sequence of Figure 16.2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>-</td>
<td>0.293</td>
<td>± 0.003 (0.001)</td>
<td>0.86 (0.31)</td>
<td>[0.288 0.298]</td>
</tr>
<tr>
<td>$C_{L_a}$</td>
<td>4.54</td>
<td>4.693</td>
<td>± 0.076 (0.023)</td>
<td>1.62 (0.50)</td>
<td>[4.540 4.845]</td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-</td>
<td>-0.028</td>
<td>± 0.002 (0.000)</td>
<td>6.66 (0.94)</td>
<td>[-0.032 -0.024]</td>
</tr>
<tr>
<td>$C'_{m_0}$</td>
<td>-0.626</td>
<td>-0.403</td>
<td>± 0.029 (0.003)</td>
<td>7.24 (0.86)</td>
<td>[-0.461 -0.345]</td>
</tr>
<tr>
<td>$C'_{m_q}$</td>
<td>-12.099</td>
<td>-15.784</td>
<td>± 0.971 (0.150)</td>
<td>6.15 (0.95)</td>
<td>[-17.725 -13.842]</td>
</tr>
<tr>
<td>$C'_{m_d}$</td>
<td>-1.159</td>
<td>-1.224</td>
<td>± 0.043 (0.007)</td>
<td>3.50 (0.57)</td>
<td>[-1.310 -1.138]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 99.20, R2 for Output 2: 99.58, R2 for Output 3: 98.63
second input will occur at an off-trim condition, based on the attitude of the aircraft after the first input. This is acceptable due to the capability of the LLM model to deal with off-trim conditions. The response now contains twice the data, and as it was shown for the wind tunnel tests, this improves the parameter ID substantially, especially under noisy conditions. It also makes much better use of the available flight time. The frequency response shown in Figure 16.2 is distorted by the fact that the two inputs are spaced out of phase. In the time domain, there is a clear and flat segment between the two inputs, where the elevator is returned to the trim condition and no change in attitude occurs. The two inputs are therefore independent. The fit of the model to the data is again very good, with only minimal residuals due to turbulence. The parameter estimates listed in
Table 16.6: Longitudinal derivatives without the added mass contribution and $C_{m_α}^\prime$ fixed at -12.1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L0}$</td>
<td>-</td>
<td>0.278</td>
<td>± 0.004 (0.001)</td>
<td>1.46 (0.42)</td>
<td>[0.270 0.286]</td>
</tr>
<tr>
<td>$C_{L\alpha}$</td>
<td>4.54</td>
<td>4.625</td>
<td>± 0.124 (0.030)</td>
<td>2.68 (0.65)</td>
<td>[4.377 4.873]</td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-</td>
<td>-0.022</td>
<td>± 0.002 (0.000)</td>
<td>8.11 (1.35)</td>
<td>[-0.025 -0.018]</td>
</tr>
<tr>
<td>$C_{m_\alpha}^\prime$</td>
<td>-0.626</td>
<td>-0.474</td>
<td>± 0.028 (0.004)</td>
<td>5.80 (0.89)</td>
<td>[-0.529 -0.419]</td>
</tr>
<tr>
<td>$C_{m_\delta e}^\prime$</td>
<td>-1.159</td>
<td>-1.072</td>
<td>± 0.045 (0.007)</td>
<td>4.20 (0.70)</td>
<td>[-1.162 -0.982]</td>
</tr>
</tbody>
</table>

$\omega_n = 1.47$ Hz, $\zeta = 0.70$

R2 for Output 1: 98.43, R2 for Output 2: 99.20, R2 for Output 3: 98.26

Table 16.5, again using the true pitch inertia, still have large differences to the expected reference values and $C_{L_\delta e}$ continues to be unidentifiable. The large correlation between $C_{m_q}$ and $C_{m_\delta e}$ also remains. The reason why the solution to the problem took such a long time is the fact that the estimates for $C_{m_q}$ and $C_{m_\delta e}$ are constantly too large and the results for $C_{m_q}^\prime$ are too small. Therefore using the inertia with the added mass component alone does not solve the problem. And since there is no literature where such strong (or any) correlations are reported in the longitudinal axis, this was initially not considered a problem. Only after all conceivable error sources, such as wind, measurement errors, dynamic pressure variation during manoeuvres and so on, were eliminated, was the possibility of the correlation being the reason for the difficulties in matching the model parameters considered. Therefore, one must use a multi step approach, where in the first step $C_{m_q}$ is held fixed at the expected value and is removed from the list of parameters to be estimated. This eliminates the correlation issue and results in the parameter estimates listed in Table 16.6.

As Table 16.6 shows, even with the removal of the correlation did the results for $C_{m_q}^\prime$ and $C_{m_\delta e}^\prime$ not match the expected data. But, and this led to the final solution of the problem, both errors are now in the same direction, whereas previously $C_{m_q}^\prime$ was too large and $C_{m_\delta e}^\prime$ was too small. In the new results, both values are too large, indicating a reduced pitch stability. This can be caused by two different issues: Either the flight CG was behind the nominal value or the pitch inertia $I_y$ is too small. The required CG shift for this error is about 11 mm backwards, which does not sound like much, but for an aircraft of this scale it is. A CG position error of this magnitude is clearly detectable during the weight and balance checks and it would have not been missed during the three years of flight testing the aircraft. So the reason must be a difference in pitch inertia between the wind tunnel test and the free flight. The two airframes were virtually identical, so the only explanation is the added mass effect. In flight, it appears, the added mass needs to be included to be able to correctly estimate the pitching moment parameters of this small aircraft. Table 16.7 lists the results of the parameter ID with
Table 16.7: Longitudinal derivatives with corrected inertias and $C_{mq}$ fixed at -12.1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>-</td>
<td>0.273 ± 0.005 (0.001)</td>
<td>1.81 (0.52)</td>
<td>[0.263, 0.283]</td>
<td></td>
</tr>
<tr>
<td>$C_{L_\alpha}$</td>
<td>4.54</td>
<td>4.796 ± 0.156 (0.037)</td>
<td>3.26 (0.76)</td>
<td>[4.484, 5.109]</td>
<td></td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-</td>
<td>-0.022 ± 0.003 (0.000)</td>
<td>12.86 (1.93)</td>
<td>[-0.028, -0.016]</td>
<td></td>
</tr>
<tr>
<td>$C'<em>{m</em>\alpha}$</td>
<td>-0.626</td>
<td>-0.662 ± 0.044 (0.006)</td>
<td>6.71 (0.96)</td>
<td>[-0.751, -0.573]</td>
<td></td>
</tr>
<tr>
<td>$C'<em>{m</em>{\delta e}}$</td>
<td>-1.159</td>
<td>-1.222 ± 0.073 (0.010)</td>
<td>6.01 (0.86)</td>
<td>[-1.369, -1.075]</td>
<td></td>
</tr>
</tbody>
</table>

$\omega_n = 1.43 \text{ Hz}, \zeta = 0.64$

R2 for Output 1: 97.56, R2 for Output 2: 98.44, R2 for Output 3: 97.98

$C'_{mq}$ held fixed and with the inertial properties set to the values that include the added mass from Table 13.17. Figure 16.3 shows the corresponding model fit, which is slightly worse than the model fit from Figure 16.2, especially in the pitch rate $q$. Since the data contains some remaining noise, it is, however, difficult to judge whether this difference in the model fit is caused by that noise or remaining modelling errors. Both pitching moment derivatives are now estimated within 5% of the expected values, which shows that the reasoning is indeed correct. The remaining difference of 5% is a very good result, considering the number, magnitude and uncertainties of the corrections required to compare wind tunnel and flight data. These were the estimation of the added mass components, the estimate for $C'_{m_\alpha}$ and the wind tunnel wall corrections, all quantities not directly measurable.

The second step is now to try and confirm $C'_{mq}$ by setting $C'_{m_{\delta e}}$ to its identified value and disable its estimation. The result for $C'_{m_\alpha}$ should be similar to the previous result if this method is valid. The results are listed in Table 16.8. As expected are the estimates

Table 16.8: Longitudinal derivatives with corrected inertias and with $C_{m_{\delta e}}$ fixed

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Value</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>-</td>
<td>0.275 ± 0.004 (0.001)</td>
<td>1.59 (0.52)</td>
<td>[0.266, 0.283]</td>
<td></td>
</tr>
<tr>
<td>$C_{L_\alpha}$</td>
<td>4.54</td>
<td>4.738 ± 0.143 (0.037)</td>
<td>3.03 (0.79)</td>
<td>[4.451, 5.025]</td>
<td></td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-</td>
<td>-0.022 ± 0.003 (0.000)</td>
<td>13.36 (1.92)</td>
<td>[-0.028, -0.016]</td>
<td></td>
</tr>
<tr>
<td>$C'<em>{m</em>\alpha}$</td>
<td>-0.626</td>
<td>-0.618 ± 0.045 (0.007)</td>
<td>7.28 (1.06)</td>
<td>[-0.708, -0.528]</td>
<td></td>
</tr>
<tr>
<td>$C'<em>{m</em>{\delta e}}$</td>
<td>-12.099</td>
<td>-12.917 ± 1.549 (0.230)</td>
<td>11.99 (1.78)</td>
<td>[-16.016, -9.819]</td>
<td></td>
</tr>
</tbody>
</table>

$\omega_n = 1.42 \text{ Hz}, \zeta = 0.67$

R2 for Output 1: 97.39, R2 for Output 2: 98.61, R2 for Output 3: 98.04
for $C'_{m_\alpha}$ very similar to the results in Table 16.7, which shows that it is estimated correctly and independently of the other two pitching moment parameters. The estimate for $C'_{m_q}$ is very close to the reference value that was used in the first step, confirming that the reference data is matching the flight results very well.

Questions now arise about the physical explanation of the added mass effect being significant in flight and not in the wind tunnel. In Part IV it was assumed that the added mass causes an additional inertia that is added to the ‘vacuum’ inertia matrix. For the wind tunnel tests these additional inertias were simply subtracted and this gave the correct result for the wind tunnel parameters. But when the physical explanation of the added mass effect, namely the energy transfer due to acceleration, is considered, this cannot be the full explanation. The pendulum motion is a rotation, although with a very small angle change, and the wind tunnel motion is also purely rotational. During the short period motion in flight, the main components of the motion are also rotational. So
where is the difference, which causes the inertial properties to change as they apparently do? At this stage the answer is unknown, but must be related to the assumption of the added mass matrix being diagonal with no off-diagonal terms.

The next section will present averaged results from the identification of multiple manoeuvres using the two step procedure. Additionally, an explanation will be attempted on the nature of the strong correlation between the parameters and the insignificance of the $C_{L_{e}}$ parameter, which appears to be a distinct feature of these very small vehicles which has not been reported on larger aircraft of standard configuration.

### 16.4 Multiple Input Results

Similarly to the wind tunnel tests, this section shows the results of repeated parameter ID of multiple input sequences. In flight it is not possible to repeat the manoeuvres quickly after each other to generate a long sequence, which gave the best results for the wind tunnel data. There are extensions of the output error method that allow it to use multiple manoeuvres for a single run [74] but these have not yet been tested. Here each input response is identified separately and the results are averaged. The data of two very clean flights was used for this section, one flight (14f9) with the single elevator inputs and another (16f6) with the sequence of two inputs. The 14f9 flight data is affected by the large size of the inputs and the fact that the UAVmainframe at the time did not yet have pitch control. Hence, the trim conditions at the start of the manoeuvre vary more than in the 16f6 flight. This, together with the shorter input sequence is expected to yield larger scatter for the 14f9 flight. As before, the initial parameter ID runs were made with $C'_{m_q}$ fixed to the reference value.

#### 16.4.1 Fixed $C'_{m_q}$

The parameter estimates presented in this section were obtained from the three ID methods with a fixed value for $C'_{m_q} = -12.1$. Eight input sequences were available from the 14f9 flight and six from the 16f6 flight. The results are listed in Tables 16.9 and 16.10 with corresponding plots in Figs. 16.4 and 16.5.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>0.275</td>
<td>-</td>
<td>0.323 ± 0.014</td>
<td>0.328 ± 0.013</td>
</tr>
<tr>
<td>$C_{L_{a}}$</td>
<td>4.738</td>
<td>4.619 ± 0.304</td>
<td>4.900 ± 0.208</td>
<td>4.423 ± 0.162</td>
</tr>
<tr>
<td>$C'<em>{m</em>{a}}$</td>
<td>-0.022</td>
<td>-</td>
<td>-0.024 ± 0.003</td>
<td>-0.022 ± 0.002</td>
</tr>
<tr>
<td>$C'<em>{m</em>{e_{a}}}$</td>
<td>-0.618</td>
<td>-0.694 ± 0.035</td>
<td>-0.714 ± 0.054</td>
<td>-0.699 ± 0.044</td>
</tr>
<tr>
<td>$C'<em>{m</em>{k_e}}$</td>
<td>-1.159</td>
<td>-1.250 ± 0.075</td>
<td>-1.245 ± 0.058</td>
<td>-1.169 ± 0.068</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>1.462 ± 0.177</td>
<td>1.491 ± 0.153</td>
<td>1.453 ± 0.153</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.619 ± 0.011</td>
<td>0.624 ± 0.009</td>
<td>0.609 ± 0.012</td>
</tr>
</tbody>
</table>
Table 16.10: Results for repeated longitudinal inputs of 16f6 with $C'_{m_q}$ fixed

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EIQ Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_o}$</td>
<td>0.275</td>
<td>–</td>
<td>0.280 ± 0.009</td>
<td>0.296 ± 0.005</td>
</tr>
<tr>
<td>$C_{L_o}$</td>
<td>4.738</td>
<td>4.916 ± 0.167</td>
<td>4.841 ± 0.149</td>
<td>4.750 ± 0.109</td>
</tr>
<tr>
<td>$C_{m_o}$</td>
<td>-0.022</td>
<td>–</td>
<td>-0.023 ± 0.001</td>
<td>-0.023 ± 0.001</td>
</tr>
<tr>
<td>$C'_{m_o}$</td>
<td>-0.618</td>
<td>-0.659 ± 0.020</td>
<td>-0.678 ± 0.023</td>
<td>-0.665 ± 0.012</td>
</tr>
<tr>
<td>$C'<em>{m</em>{k_e}}$</td>
<td>-1.159</td>
<td>-1.230 ± 0.061</td>
<td>-1.228 ± 0.047</td>
<td>-1.222 ± 0.033</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>1.512 ± 0.099</td>
<td>1.520 ± 0.119</td>
<td>1.505 ± 0.113</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.647 ± 0.003</td>
<td>0.639 ± 0.004</td>
<td>0.639 ± 0.004</td>
</tr>
</tbody>
</table>

Figure 16.4: Results for repeated longitudinal inputs of 14f9 with $C'_{m_q}$ fixed (Figure key: section 9.2.4)
The lift curve slope $C_{L\alpha}$ for 14f9 is predicted best by the equation error method, albeit with the largest uncertainty. The output error method produces a value that is too large, while the estimate of the filter error method is too small. For the longer sequences of 16f6, the filter error method performs best in terms of the mean value and the spread. The output error method comes second and the equation error algorithm under predicts by about 5%, again with the largest uncertainty. $C_{m\alpha}$ is underpredicted by all methods for 14f9, whereas for 16f6 the filter and equation error method produce estimates within 6% of the expected value. $C_{m\alpha}$ is quite consistently underpredicted by 5% in both data sets, except for the filter error method in 14f9. The uncertainties for the 16f6 data set are expectedly smaller than for the 14f9 flight. The 16f6 data is highly consistent between the methods and between runs and gives excellent results with small uncertainties. This confirms that the longer inputs in combination with the longitudinal flight controllers deliver much better data than what was achieved on earlier flights. The remaining difference between the reference values and the flight data is 5% for $C_{L\alpha}$ and 6% for $C_{m\alpha}$. Given the almost perfect repeatability of the flight results, these errors are most likely caused by the multiple corrections that were required to obtain comparable data, as
discussed above.

Inspecting Tables 16.9 and 16.10 and Figure 16.4, one can observe much smaller error bars on the results of the equation error method. This is caused by the fact that this method does not require the treatment for the coloured residuals, since it is operating in the frequency domain. The spread of the results between the different inputs is however no better, or even larger than for the two time domain methods, which were treated for the coloured residuals. The figure also shows that the spread of the time domain results lies within the bounds of the error bars, but the equation error results do not. It is therefore unlikely that the untreated frequency domain uncertainties represent the true spread in the data and need some form of treatment similarly to the time domain methods. In the next section, the same data will be analysed again with $C'_{m_{\delta e}}$ set to the mean estimated value of -1.22 from Table 16.10 to confirm the reference value for $C'_{m_q}$.

### 16.4.2 Fixed $C'_{m_{\delta e}}$

Tables 16.11 and 16.12 list the parameter estimates for the longitudinal input sequences with $C'_{m_{\delta e}}$ fixed at the previously estimated value to -1.22. The corresponding plots of the results are shown in Figure 16.6 and 16.7. The picture is very similar to the previous section, where the data from flight 16f6 is estimated with smaller uncertainties. $C'_{m_q}$ is estimated slightly smaller than the expected value for both data sets and all methods, but this difference is hardly worth mentioning. This small change in $C'_{m_q}$, however, improves the results for $C'_{m_{\alpha}}$ even further, which are now very close to the expected value of -0.618 for the 16f6 data and slightly smaller for the 14f9 data. Looking at the overall uncertainties in the plots, the same issue of the too small error bars for the equation error results is evident. On the other hand, the two time domain methods deliver very similar performance.

The quality of these results show that when the added mass contributions and the pitching moment correlations are treated correctly, it is possible to confirm the reference data obtained from the ground tests within 5% and better for a small aircraft like this. Further improvement may be possible by repeating the input sequence more often during

<table>
<thead>
<tr>
<th>Table 16.11: Results for repeated longitudinal inputs of 14f9 with $C'<em>{m</em>{\delta e}}$ fixed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>$C_{L_0}$</td>
</tr>
<tr>
<td>$C_{L_\alpha}$</td>
</tr>
<tr>
<td>$C'_{m_0}$</td>
</tr>
<tr>
<td>$C'<em>{m</em>{\alpha}}$</td>
</tr>
<tr>
<td>$C'_{m_q}$</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
</tr>
<tr>
<td>$\zeta$</td>
</tr>
</tbody>
</table>
16.4 Multiple Input Results

Figure 16.6: Results for repeated longitudinal inputs of 14f9 with $C'_{m_{\delta e}}$ fixed

Table 16.12: Results for repeated longitudinal inputs of 16f6 with $C'_{m_{\delta e}}$ fixed

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>EQN Mean±2σ</th>
<th>OEM Mean±2σ</th>
<th>FEM Mean±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_{0}}$</td>
<td>0.275</td>
<td>–</td>
<td>0.281 ± 0.008</td>
<td>0.296 ± 0.006</td>
</tr>
<tr>
<td>$C_{L_{a}}$</td>
<td>4.738</td>
<td>4.916 ± 0.167</td>
<td>4.800 ± 0.191</td>
<td>4.738 ± 0.101</td>
</tr>
<tr>
<td>$C_{m_{0}}$</td>
<td>-0.022</td>
<td>–</td>
<td>-0.023 ± 0.001</td>
<td>-0.023 ± 0.001</td>
</tr>
<tr>
<td>$C'<em>{m</em>{a}}$</td>
<td>-0.618</td>
<td>-0.618 ± 0.031</td>
<td>-0.631 ± 0.024</td>
<td>-0.623 ± 0.005</td>
</tr>
<tr>
<td>$C'<em>{m</em>{q}}$</td>
<td>-12.917</td>
<td>-13.029 ± 1.173</td>
<td>-13.151 ± 0.814</td>
<td>-13.134 ± 0.540</td>
</tr>
<tr>
<td>$\omega_{n}$ [Hz]</td>
<td>–</td>
<td>1.509 ± 0.109</td>
<td>1.513 ± 0.126</td>
<td>1.503 ± 0.118</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>–</td>
<td>0.673 ± 0.021</td>
<td>0.667 ± 0.014</td>
<td>0.667 ± 0.008</td>
</tr>
</tbody>
</table>

A single flight and to improve upon the added mass estimate required for the magnitude of the pitch inertia $I_{y}$, as well as by confirming the longitudinal CG with sub-millimetre accuracy. The reference values for $C'_{m_{\delta e}}$ appear to be slightly too small and a revisit of these results may be worthwhile as well. The results also show that the model structure
Chapter 16. Longitudinal Results

Figure 16.7: Results for repeated longitudinal inputs of 16f6 with $C_{m_{\delta_e}}$ fixed

with the measured data works very well, especially in combination with the EKF corrected data. Further, all three methods yield good parameter estimates, with the filter error method only slightly better than the other two. Yet the issue with the uncertainties of the frequency domain methods need a revisit and until then the two time domain methods are preferable, unless a non-iterative solution is required.

16.4.3 Interpretation of Results

The question is then, what causes the high correlation between the model parameters and why is $C_{L_{\delta_e}}$ insignificant for the motion, despite having a measurable value in the wind tunnel? The answer is most likely the way the ratio of mass and moment of inertia scales down to this small scale. Mass scales with the power of 3 and the inertia with the power of 5. That means that for this small aircraft, the inertia in pitch is tiny compared to the inertia in lift because the mass in $F = ma$ has a similar effect as the inertia in $M = L\dot{\omega})$. Also, the contribution of the tailplane to the overall lift of the aircraft is small. During a change in elevator angle, the rotational motion reacts much faster than the translation. This results in most of the lift change on the tailplane being converted into a
16.4 Multiple Input Results

Pitch rate and almost none into a change in overall lift. Therefore $C_{m_q}$ and $C_{m_{\delta_e}}$ have both a very similar effect on the motion and become correlated. This also explains why $C_{L_{\delta_e}}$ has virtually no effect on the motion. This author is unaware of any other publication where this correlation occurs and poses a serious issue during the parameter ID process. It is therefore an interesting topic for future work to establish the size of aircraft for which this problem exists, and if there is any way to avoid the issue to be able to identify all pitching moment parameters in a single run.

16.4.4 EKF Test

The excellent quality of the longitudinal flight data also allows to demonstrate the effect of the EKF corrections. Tables 16.13 and 16.14 list results of runs with the raw data of flight 16f6, calibrated only for the airdata probe errors and accelerometer alignment. Included in the tables are the mean estimates from the corrected data from the previous section. As before one run was made with $C'_{m_q}$ fixed and another with $C'_{m_{\delta_e}}$ fixed. The lift curve slope $C_{L_0}$ is only moderately affected by the non-compatible data, only the output error method in Table 16.14 is considerably off. The estimates for $C'_{m_{\alpha}}$ show larger errors but are still within acceptable ranges. The results for $C'_{m_{\delta_e}}$ and $C'_{m_q}$, however, are completely wrong and totally useless with errors up to 55%. Notable is also that in all previous results the zero moment coefficient $C_{m_0}$ was estimated reasonably well, but becomes completely wrong as well in Table 16.14.

These results show clearly that the data compatibility check and correction is vital to be able to obtain useful flight data with the grade of sensors used and the limitations in sensor alignments in a small aircraft. It is surprising that almost no other publication appears to be concerned about this matter, or maybe these steps are seen as trivial and are not reported. Since this author has spent about a third of his time on the calibration and EKF related work, this topic should be mandatory in any publication on aircraft flight testing, similarly to the mesh size studies required for any publication on computational fluid dynamics (CFD).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>with EKF</th>
<th>OEM Mean value±2σ</th>
<th>FEM Mean value±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_0}$</td>
<td>0.29</td>
<td>0.242 ± 0.012</td>
<td>0.256 ± 0.004</td>
</tr>
<tr>
<td>$C_{L_{\alpha}}$</td>
<td>4.8</td>
<td>5.178 ± 0.203</td>
<td>4.960 ± 0.070</td>
</tr>
<tr>
<td>$C_{m_0}$</td>
<td>-0.023</td>
<td>-0.019 ± 0.001</td>
<td>-0.020 ± 0.001</td>
</tr>
<tr>
<td>$C'<em>{m</em>{\alpha}}$</td>
<td>-0.62</td>
<td>-0.716 ± 0.020</td>
<td>-0.646 ± 0.076</td>
</tr>
<tr>
<td>$C'<em>{m</em>{\delta_e}}$</td>
<td>-1.22</td>
<td>-1.694 ± 0.053</td>
<td>-1.650 ± 0.060</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>1.519 ± 0.095</td>
<td>1.461 ± 0.129</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.642 ± 0.004</td>
<td>0.653 ± 0.021</td>
</tr>
</tbody>
</table>
Table 16.14: 16f6 estimates from raw data with $C_{m_{data}}'$ fixed

<table>
<thead>
<tr>
<th>Parameter</th>
<th>with EKF</th>
<th>OEM Mean value±2σ</th>
<th>FEM Mean value±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L0}$</td>
<td>0.29</td>
<td>0.224 ± 0.019</td>
<td>0.255 ± 0.006</td>
</tr>
<tr>
<td>$C_{L\alpha}$</td>
<td>4.8</td>
<td>5.853 ± 0.190</td>
<td>4.973 ± 0.082</td>
</tr>
<tr>
<td>$C_{m0}$</td>
<td>-0.023</td>
<td>-0.007 ± 0.004</td>
<td>-0.009 ± 0.003</td>
</tr>
<tr>
<td>$C_{m\alpha}'$</td>
<td>-0.62</td>
<td>-0.802 ± 0.034</td>
<td>-0.710 ± 0.033</td>
</tr>
<tr>
<td>$C_{m\alpha}$</td>
<td>-13.1</td>
<td>-5.922 ± 0.820</td>
<td>-6.603 ± 0.391</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>1.431 ± 0.106</td>
<td>1.358 ± 0.098</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.559 ± 0.019</td>
<td>0.545 ± 0.011</td>
</tr>
</tbody>
</table>

### 16.4.5 Predictive Capability

Another demonstration of the accuracy achieved with the UAVmainframe can be done by predicting a manoeuvre from a different flight with the results of this chapter. This is done in Figure 16.8, using the data from before and the technique explained in Reference [22]. The model is integrated in time with the estimated parameters from this chapter, but the bias values still need to be re-calculated for the prediction data set. With the model structures used for this project this is easy. All known parameters are set to their respective values and the estimation of these parameters is turned off. That results in a normal run of the output error method, which now simply estimates the remaining bias parameters. The result is shown in Figure 16.8 for a manoeuvre of flight 16f5, which took place some time before the 16f6 flight. This data is quite noisy due to higher turbulence levels at the time. The data also shows significant changes in dynamic pressure during the manoeuvre, which indicates a bad trim condition at the start, possibly caused by a gust hitting just before the start of the sequence. Figure 16.8 shows that the motion is predicted very well by the model and the estimated parameters. This exercise has also been repeated with data from the 14fxx block, which was recorded over a year before with a very different flight software and input sequence. The results are of the same quality as shown in Figure 16.8, which indicates that the determined model parameters capture the dynamics of the test aircraft in response to an elevator input very well.
Figure 16.8: 16f5 predicted manoeuvre
16.5 Summary

This concludes the presentation of the longitudinal results from the flight tests. Several, so far unknown features of the flight dynamics of an aircraft of this scale were discovered. Firstly, the inertial properties are significantly affected by the added mass phenomenon and secondly, strong correlations between the moment derivatives prohibit their estimation in a single run. This has some serious implications for the usefulness of the flight test results, because in order to estimate all parameters in the model, it is necessary to know either $C'_{m_q}$ or $C'_{m_{\delta_e}}$. Otherwise it cannot be done. One might use an iterative process if neither of these values is known, but this will be tedious. The only feasible method of decorrelating the moment parameters is to design and use input sequence which achieves this. Maybe the more advanced orthogonal sine inputs that are being used will be better in this case. This topic, however, must be added to the long list of suggested future work at the end of the thesis.
After successfully estimating the parameters in the longitudinal motion, it is now time to address the lateral manoeuvres. The results presented in this chapter are preliminary and have not yet the same high quality as the longitudinal results. Most time and effort was spent on working out the problems with the longitudinal data and only limited effort was put into the lateral input design until the very last flight test in April 2016. As mentioned previously, the longitudinal controllers were also only added for this last test opportunity. As will be shown in this chapter, the roll mode is strongly affected by large airspeed variations caused by a bad trim condition. This renders most old data useless. Unfortunately, the weather was not favourable during most of this last excursion and only a single flight with good lateral data is available, the rest being too noisy due to high turbulence levels, thus limiting the opportunity to optimise the input sequences. The limited data shown in this chapter, however, demonstrates that the methods are on the right track and will yield good results once fully optimised. The data presented also shows that the inertial properties with the added mass components give the correct results, similarly to the longitudinal axis.

The chapter starts with the dutch roll mode, then continues with the roll mode and finishes with the lateral combined inputs. The dutch roll mode has enough data to show the analysis for repeated inputs, similarly to the wind tunnel data from part V. The other two modes will be discussed showing good and poor examples and pointing out the issues as well as planned improvements to increase the data quality. As for the longitudinal axis, all data used in this chapter is the output from the EKF.
17.1 Dutch Roll Mode

17.1.1 Model Structure

The dutch roll approximation can be developed from the full lateral equations of motion by assuming the motion affects only yaw and sideslip and thus removing the rolling moment equation [39]. The Y-axis acceleration is added to the output equations.

**Theorem 17.1.1 — Dutch Roll Approximation.** Two equation dutch roll model with roll rate \( p \) as pseudo-input, ignoring the cross inertia contribution

**States:**

\[
\dot{\beta} = \frac{\ddot{q}S}{mV_{air}} C_y + \sin(\alpha_m)p_m + (-\cos(\alpha_m))r + b_{\beta} \quad (17.1)
\]

\[
\dot{r} = \frac{\ddot{q}Sb}{I_z} C_n + b_r \quad (17.2)
\]

**Measurements:**

\[
\beta = \beta \quad r = r \quad a_y = \frac{\ddot{q}S}{mg} C_y + b_{a_y} \quad (17.3)
\]

with aerodynamic expansions

\[
C_y = C_{y_{y\beta}} \beta + C_{y_{yr}} \frac{rb}{2V_{air}} + C_{y_{r\beta}} \delta_r \quad (17.4)
\]

\[
C_n = C_{n_{y\beta}} \beta + C_{n_{np}} \frac{p_m b}{2V_{air}} + C_{n_{nr}} \frac{rb}{2V_{air}} + C_{n_{r\beta}} \delta_r \quad (17.5)
\]

with potentially identifiable parameters

\[C_{y_{y\beta}} \quad C_{y_{yr}} \quad C_{y_{r\beta}} \quad C_{n_{np}} \quad C_{n_{nr}} \quad C_{n_{r\beta}}\]

The roll rate \( p \) is used as a pseudo input in the yawing moment equation to potentially be able to identify the cross-coupling term \( C_{np} \). Reference [38] omits the \( C_{yr} \) derivative as typically insignificant, but is has been kept in the model to test this assumption. Unidentifiable parameters can always be turned off for the final runs.

17.1.2 Input Design and Verification

The input design for the dutch roll mode is a simple rudder doublet as shown in Figure 17.1. Longer rudder sequences, similar to the wind tunnel inputs have been tested, but there is not yet enough data to show any improvement over the single doublet sequence. The response shown in Figure 17.1 is a typical example at reasonably low turbulence levels, but there is significantly more noise in the data than in the longitudinal axis of the same flight. These noise levels make it difficult to identify the very low damping of the mode correctly because the oscillation is distorted by the noise as soon as the amplitude becomes small. The Y-axis accelerometer is affected by airframe vibrations from the propulsion system, despite all efforts of damping the motor. But the accelerations due to the sideforce and quite small compared to the Z-axis accelerations in the longitudinal case and thus even small vibrations become more significant.
The corresponding parameter estimates are listed in Table 17.1. The derivatives $C_{n_p}$ and $C_{y_r}$ are not identifiable from this data and have been turned off. Most other derivatives are estimated close to the expected values, only the sideforce due to rudder is significantly lower than expected. This value also has a very high uncertainty, which indicates that it is not sufficiently observable in this particular data set. The yaw damping term $C_{n_r}$ also has a large uncertainty, especially after being corrected for the coloured residuals, even though in this example the estimated value is very close to the expected result. It suffers from the observability issues due to noise, as mentioned before.

Figure 17.1: Rudder input sequence with identified model
Chapter 17. Lateral Flight Results

Table 17.1: Parameter estimates corresponding to the dutch roll manoeuvre in Figure 17.1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{y,\beta}$</td>
<td>-0.51</td>
<td>-0.501</td>
<td>± 0.036 (0.013)</td>
<td>7.15 (2.62)</td>
<td>[-0.572 -0.429 ]</td>
</tr>
<tr>
<td>$C_{y,\delta r}$</td>
<td>0.169</td>
<td>0.080</td>
<td>± 0.046 (0.017)</td>
<td>58.09 (21.37)</td>
<td>[-0.013 0.172 ]</td>
</tr>
<tr>
<td>$C_{n,\beta}$</td>
<td>0.087</td>
<td>0.081</td>
<td>± 0.006 (0.001)</td>
<td>7.19 (1.16)</td>
<td>[ 0.070 0.093 ]</td>
</tr>
<tr>
<td>$C_{n,r}$</td>
<td>-0.064</td>
<td>-0.063</td>
<td>± 0.027 (0.004)</td>
<td>42.41 (5.92)</td>
<td>[-0.117 -0.010 ]</td>
</tr>
<tr>
<td>$C_{n,\delta r}$</td>
<td>-0.063</td>
<td>-0.056</td>
<td>± 0.010 (0.001)</td>
<td>17.48 (2.34)</td>
<td>[-0.075 -0.036 ]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 86.43, R2 for Output 2: 73.88, R2 for Output 3: 78.73

17.1.3 Multiple Inputs

Five manoeuvres from the same flight have been analysed with the output- and filter error methods. The results are shown in Figure 17.2 and are listed in Table 17.2. The first observation is that all results are spread more or less within the confidence intervals, which shows that the correction for the coloured residuals works well.

The results of both methods are quite similar, with the filter error method showing slightly smaller standard deviations. In terms of the mean parameter estimates, the output error method gives a better result for $C_{n,\beta}$, while the filter error method is better for $C_{n,r}$. The remaining results are similar and both methods estimate $C_{y,\delta r}$ significantly too small and with large uncertainties, which confirms the low observability of this parameter. The small vertical tail does not create a notable sideforce and the small rudder does not change this sideforce by a large amount. However, the rudder effectiveness on the yaw axis $C_{n,\delta r}$ is estimated without error and mostly very small uncertainties. Due to the small inertia of the aircraft in the Z-axis, even the little amount of sideforce generated by the rudder is sufficient to generate a easily identifiable yawing moment. The sideforce due to sideslip $C_{y,\beta}$ and the weathercock stability parameter $C_{n,\beta}$ are also estimated accurately.

Table 17.2: Results of multiple dutch roll manoeuvres

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>OEM Mean value±2σ</th>
<th>FEM Mean value±2σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{y,\beta}$</td>
<td>-0.51</td>
<td>-0.508±0.029</td>
<td>-0.509±0.016</td>
</tr>
<tr>
<td>$C_{y,\delta r}$</td>
<td>0.169</td>
<td>0.087±0.081</td>
<td>0.098±0.038</td>
</tr>
<tr>
<td>$C_{n,\beta}$</td>
<td>0.087</td>
<td>0.088±0.010</td>
<td>0.081±0.010</td>
</tr>
<tr>
<td>$C_{n,r}$</td>
<td>-0.064</td>
<td>-0.081±0.038</td>
<td>-0.074±0.045</td>
</tr>
<tr>
<td>$C_{n,\delta r}$</td>
<td>-0.063</td>
<td>-0.063±0.009</td>
<td>-0.063±0.007</td>
</tr>
<tr>
<td>$\omega_n$ [Hz]</td>
<td>-</td>
<td>0.809±0.114</td>
<td>0.776±0.099</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>-</td>
<td>0.143±0.037</td>
<td>0.141±0.042</td>
</tr>
</tbody>
</table>
with the filter error method giving a smaller (10%) value for $C_{n_}\beta$. The yaw damping parameter $C_{n_}\gamma$ is over predicted by both methods and suffers again from low observability due to turbulence, as shown in the large uncertainties for this value.

Overall, these results are of acceptable accuracy and confirm that the Z-axis inertia with added mass contribution is correct for this axis. The low damping in the yaw axis will require some special treatment to improve the results, although at this stage a solution which will allow to identify this parameter in noisy conditions is not yet known. The low rudder effectiveness on the sideforce appears to have only limited impact on the dutch roll motion. Maybe it will be better observable during the lateral combined inputs later on. There is no clear advantage of the filter error method over the output error method for this model, where both algorithms deliver comparable results.
17.2 Roll Mode

17.2.1 Model Structure

The model structure for the roll mode in flight is identical to the model used during the wind tunnel tests. It consists only of the rolling moment equation, with the sideslip angle and the yaw rate included from measured data.

\[ \dot{\beta} = \frac{qSb}{I_{x}}C_{l} + b_{p} \]  
(17.6)

Measurements:

\[ \beta = \beta \]  
(17.7)

and the aerodynamic rolling moment

\[ C_{l} = C_{l_{p}}\beta_{m} + C_{l_{p}}\frac{pb}{2V_{air}} + C_{l_{r}}\frac{r_{m}b}{2V_{air}} + C_{l_{a}}\delta_{a} \]  
(17.8)

and potentially identifiable parameters

\[ C_{l_{p}}, C_{l_{r}}, C_{l_{a}} \]

where \( C_{l_{p}} \) and \( C_{l_{a}} \) can again be expected to be correlated to a high level, since this model is identical to the 3 DoF model used for the wind tunnel experiments. The time constant of the exponentially converting roll mode is given by

\[ \tau = \frac{1}{L_{p}} = \frac{I_{x}2V_{air}}{qSb^{2}C_{l_{p}}} \]  
(17.9)

17.2.2 Input Design and Verification

The input for the roll mode is a repeated aileron doublet as shown in Figure 17.3. Two responses and the resulting parameter estimates are discussed. A good one in Figure 17.3 and a poor one in Figure 17.4. There is not yet enough data to fully explain the differences. Similarly to the wind tunnel experiments, it is not possible to estimate \( C_{l_{p}} \) and \( C_{l_{a}} \) at the same time due to correlation. For the following analysis the aileron derivative was set to the reference value and \( C_{l_{a}} \) is estimated. Given the good results for the rudder derivative in the previous section, this approach appears justified. The filter error method does not converge well for this model. This is potentially caused by the limited degrees of freedom of the process noise matrix, which reduces to a single value for this model. Hence, all results presented here are obtained from the output error method.

The good response and the corresponding results, as listed in Table 17.3, show acceptable accuracy. The roll stability parameter \( C_{l_{a}} \) is estimated close to the expected value, and the time constant of the roll mode within 15% of the wind tunnel result. The cross coupling parameter \( C_{l_{r}} \) is estimated very high and with large uncertainties and thus not accurately observable in the data. This parameter is however required in the model, otherwise all other results become completely wrong. The model fit to the data
Table 17.3: Roll mode good example with $C_{\delta a}$ set to -0.178

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_I\beta$</td>
<td>-0.061</td>
<td>-0.063</td>
<td>$\pm$ 0.011</td>
<td>(0.002)</td>
<td>17.95 (3.57)</td>
</tr>
<tr>
<td>$C_Ip$</td>
<td>-0.395</td>
<td>-0.461</td>
<td>$\pm$ 0.086</td>
<td>(0.013)</td>
<td>18.64 (2.75)</td>
</tr>
<tr>
<td>$C_Ir$</td>
<td>0.129</td>
<td>0.234</td>
<td>$\pm$ 0.070</td>
<td>(0.014)</td>
<td>30.14 (5.84)</td>
</tr>
</tbody>
</table>

R2 for Output 1: 97.84

Figure 17.3: Good roll mode response
Table 17.4: Roll mode poor example with $C_{l_{\delta a}}$ set to -0.178

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{l_{\beta}}$</td>
<td>-0.061</td>
<td>-0.096</td>
<td>$\pm$ 0.015 (0.004)</td>
<td>15.27 (3.99)</td>
<td>[-0.126 -0.067]</td>
</tr>
<tr>
<td>$C_{l_{p}}$</td>
<td>-0.395</td>
<td>-0.527</td>
<td>$\pm$ 0.153 (0.022)</td>
<td>29.00 (4.23)</td>
<td>[-0.832 -0.221]</td>
</tr>
<tr>
<td>$C_{l_{r}}$</td>
<td>0.129</td>
<td>0.052</td>
<td>$\pm$ 0.146 (0.022)</td>
<td>279.27 (42.14)</td>
<td>[-0.240 0.344]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 92.24

Figure 17.4: Poor roll mode response
is reasonably good, but there are considerable overshoots in the roll rate, especially at the beginning. These overshoots are significantly worse in the poor response of Figure 17.4. It is not yet clear what the reason for these overshoots may be, since only a few of the manoeuvres were flown so far. One factor may be a relatively strong dependency of the $C_{l}$ derivative on the dynamic pressure. Some of the responses to this aileron input with larger speed changes during the manoeuvres show significantly worse parameter estimates than the ones discussed here. The same issue is also observed for the lateral combined inputs later on. Another observation in Figure 17.4 is the unsymmetrical yaw rate $\gamma$, which indicates that the aircraft has entered the spiral mode. Maybe this contribution increases the errors seen for that particular manoeuvre. It will require more work to optimise this input sequence to understand these issues before any definitive results can be presented. In any case, the lateral combined manoeuvre appears to deliver much better results for the roll mode parameters than the pure aileron input, as will be shown in the next section.
17.3 Lateral Combined Manoeuvre

17.3.1 Model Structure

The model structure for the lateral combined input in flight has been presented in reference [38]. Similarly to the longitudinal model, measured values are used to linearise the equation of motion. This allows for a wider range of trim conditions and this model has worked well so far, although the advantages of this model over the standard state space lateral model given in reference [22] are less significant than for the longitudinal model used before.

Theorem 17.3.1 — Lateral Model. Full lateral model using measured data [38]

States:
\[
\dot{\beta} = \frac{\bar{q}S}{mV_{\text{air}}} (C_y \beta + b \beta) + p \sin(\alpha_m) - r \cos(\alpha_m) + \\
g/V_{\text{air}} \sin(\phi_m) \cos(\theta_m) + g/V_{\text{air}} \cos(\phi_m) \cos(\phi_m) (\phi - \phi_m) 
\]
(17.10)
\[
\dot{p} = (c_1 r + c_2 p) q_m + \bar{q} S (c_3 C_l + c_4 C_n) 
\]
(17.11)
\[
\dot{r} = (c_8 p - c_2 r) q_m + \bar{q} S (c_9 C_n + c_4 C_l) 
\]
(17.12)
\[
\dot{\phi} = p + r \tan(\theta_m) \sin(\phi_m) + q_m \tan(\theta_m) \cos(\phi_m) 
\]
(17.13)

Measurements:
\[
\beta = \beta \quad p = p \quad r = r \quad \theta = \theta \quad a_y = \frac{\bar{q} S}{mg} C_y + b a_y 
\]
(17.14)

with aerodynamic expansions
\[
C_y = C_{y_0} \beta + C_{y_{\beta}} \beta + \frac{r b}{2V_{\text{air}}} + C_{y_{\delta r}} \delta r 
\]
(17.15)
\[
C_l = C_{l_0} + C_{l_{\beta}} \beta + C_{l_{p}} \frac{p b}{2V_{\text{air}}} + C_{l_{r}} \frac{r b}{2V_{\text{air}}} + C_{l_{\delta a}} \delta a 
\]
(17.16)
\[
C_n = C_{n_0} + C_{n_{\beta}} \beta + C_{n_{p}} \frac{p b}{2V_{\text{air}}} + C_{n_{r}} \frac{r b}{2V_{\text{air}}} + C_{n_{\delta r}} \delta r 
\]
(17.17)

with potentially identifiable parameters
\[
C_{y_{0}}, C_{y_{\beta}}, C_{y_{\delta r}}, C_{l_{0}}, C_{l_{\beta}}, C_{l_{p}}, C_{l_{r}}, C_{l_{\delta a}}, C_{n_{0}}, C_{n_{\beta}}, C_{n_{p}}, C_{n_{r}}, C_{n_{\delta r}} 
\]

where the usual correlation between $C_{l_p}$ and $C_{l_{\delta a}}$ in the roll axis is expected.

17.3.2 Input Design and Verification

The input design for the lateral combined input is similar to the wind tunnel version of section 13.2. It consists of an initial rudder input, followed by an aileron input. Using this input in flight is much more difficult than in the wind tunnel for the following reasons: The spacing between the rudder and aileron inputs is strongly airspeed dependent,
because the dutch roll frequency changes with airspeed. Placing the aileron input at the correct phase of the dutch roll is critical for observability of the cross derivatives $C_{np}$ and $C_{lr}$. Since the sequence is currently hard coded in the UAVmainframe, many of the manoeuvres flown have to be discarded due to airspeed. This can be avoided by expanding the input generator to account for airspeed, but this has not yet been implemented. The next difficulty is the long tail after the aileron input, which is required to identify the dutch roll decay correctly. Since all lateral controls stay fixed during this section of the manoeuvre, it is critical to exit the rolling motion due to the aileron input with the wings level. On a full scale aircraft this would be done by the pilot instinctively during the aileron input, but for this small aircraft the onboard systems have to achieve this without manual input. The aileron input sequence of the manoeuvre is shaped such that the last leg of the input returns the aircraft back to wings level, if the airspeed is correct. If the attitude is incorrect after the aileron input, the aircraft will roll into a spiral dive and the pilot or the automatic attitude check will have to abort the manoeuvre. These two issues have so far limited the available number of useful data for this input sequence. As with the roll mode, in the following sections a good and a poor manoeuvre will be presented to illustrate the state of the work. The correlation in the roll axis again prevents the estimation of $C_{lp}$ and $C_{ls_a}$ simultaneously. Therefore the control derivative has been fixed to its reference value.

Table 17.5: Parameter estimates for a good lateral combined input with $C_{ls_a}$ fixed

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{ya}$</td>
<td>0</td>
<td>0.005</td>
<td>± 0.001 (0.000)</td>
<td>18.27 (4.92)</td>
<td>[ 0.003 0.006 ]</td>
</tr>
<tr>
<td>$C_{y3}$</td>
<td>-0.510</td>
<td>-0.538</td>
<td>± 0.016 (0.005)</td>
<td>2.89 (0.91)</td>
<td>[ -0.569 -0.507 ]</td>
</tr>
<tr>
<td>$C_{yr}$</td>
<td>N/A</td>
<td>0.616</td>
<td>± 0.075 (0.027)</td>
<td>12.21 (4.40)</td>
<td>[ 0.466 0.767 ]</td>
</tr>
<tr>
<td>$C_{yb}$</td>
<td>-0.169</td>
<td>0.173</td>
<td>± 0.026 (0.007)</td>
<td>15.04 (3.88)</td>
<td>[ 0.121 0.225 ]</td>
</tr>
<tr>
<td>$C_{l0}$</td>
<td>0</td>
<td>-0.000</td>
<td>± 0.000 (0.000)</td>
<td>123.50 (13.31)</td>
<td>[ -0.001 0.000 ]</td>
</tr>
<tr>
<td>$C_{l3}$</td>
<td>-0.061</td>
<td>-0.041</td>
<td>± 0.013 (0.001)</td>
<td>31.61 (3.28)</td>
<td>[ -0.066 -0.015 ]</td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.395</td>
<td>-0.405</td>
<td>± 0.036 (0.007)</td>
<td>8.93 (1.69)</td>
<td>[ -0.477 -0.332 ]</td>
</tr>
<tr>
<td>$C_{ls}$</td>
<td>0.129</td>
<td>0.306</td>
<td>± 0.044 (0.006)</td>
<td>14.37 (1.98)</td>
<td>[ 0.218 0.393 ]</td>
</tr>
<tr>
<td>$C_{n0}$</td>
<td>0</td>
<td>-0.001</td>
<td>± 0.000 (0.000)</td>
<td>8.58 (1.43)</td>
<td>[ -0.002 -0.001 ]</td>
</tr>
<tr>
<td>$C_{n3}$</td>
<td>-0.087</td>
<td>0.083</td>
<td>± 0.003 (0.000)</td>
<td>3.16 (0.39)</td>
<td>[ 0.078 0.088 ]</td>
</tr>
<tr>
<td>$C_{np}$</td>
<td>-0.056</td>
<td>-0.065</td>
<td>± 0.013 (0.002)</td>
<td>19.75 (2.68)</td>
<td>[ -0.091 -0.039 ]</td>
</tr>
<tr>
<td>$C_{nr}$</td>
<td>-0.064</td>
<td>-0.059</td>
<td>± 0.013 (0.002)</td>
<td>21.14 (3.23)</td>
<td>[ -0.085 -0.034 ]</td>
</tr>
<tr>
<td>$C_{ns}$</td>
<td>-0.063</td>
<td>-0.062</td>
<td>± 0.004 (0.000)</td>
<td>6.57 (0.70)</td>
<td>[ -0.070 -0.054 ]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 97.72, R2 for Output 2: 93.55, R2 for Output 3: 97.42
R2 for Output 4: 95.60, R2 for Output 5: 95.82
Table 17.5 and Figure 17.5 show the results of a very good example of the lateral combined manoeuvre. The model fit is excellent and most estimated parameters are very close to the expected values. The y-axis acceleration is still quite noisy, but the signal to noise ratio is improved over the dutch roll response discussed before. The tail of the sequence is a bit short with only 2 seconds, which results in a large uncertainty of the yaw damping term. The roll angle at the end of the manoeuvre is about 15 degrees, which is not perfect but still low enough to prevent the dangerous roll into the spiralling motion. This input sequence could probably be elongated by another second to improve the $C_{m_\alpha}$ estimate. The roll rate shows only small overshoots when compared to the model and the resulting estimate for $C_{l_p}$ is very good. The roll stability term $C_{l_{\beta}}$ does not match...
the expected value well and it has a large uncertainty. It appears that this parameter is not very observable in this manoeuvre. The roll rate due to yaw is estimated about twice the expected value. In the yaw axis all parameters agree well with the expected values, although the uncertainties for $C_{n_p}$ and $C_{n_r}$ are quite large. The side force derivatives also agree well with the wind tunnel results.

Figure 17.6 and Table 17.6, in comparison show an example of a poor lateral combined input. There is a large increase in airspeed during the manoeuvre, which indicates a bad trim condition for this particular airspeed. As a result, the estimates for the roll axis parameters deteriorate considerably, which is also illustrated with the large overshoots in the roll rate. This appears to confirm that despite using non-dimensional derivatives, the change in airspeed is too large for the assumption of constant aerodynamic derivatives to hold. The yaw axis parameter estimates are less affected, although there is now a much larger error in the yaw damping term. At this airspeed the aileron sequence does not roll the aircraft back to wings level and the roll angle quickly deteriorates at the end of the manoeuvre. The automatic abort would have taken place probably less than a second after the current end of the sequence. This illustrates the difficulty of performing this lateral combined input with a remotely piloted vehicle. To increase reliability and productivity, it seems to be necessary to perform these manoeuvres with the autopilot engaged, and to reduce all gains to keep the flight path stable without interfering with the input sequence.

Table 17.6: Parameter estimates for a poor lateral combined input

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Ref.</th>
<th>Est.</th>
<th>Standard Dev.</th>
<th>Std. Dev. in %</th>
<th>95% conf. interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{y_0}$</td>
<td>0</td>
<td>0.007</td>
<td>$\pm$ 0.002 (0.000)</td>
<td>23.14 (4.57)</td>
<td>[0.004 0.010]</td>
</tr>
<tr>
<td>$C_{y_3}$</td>
<td>-0.510</td>
<td>-0.546</td>
<td>$\pm$ 0.030 (0.007)</td>
<td>5.51 (1.28)</td>
<td>[-0.607 -0.486]</td>
</tr>
<tr>
<td>$C_{y_r}$</td>
<td>N/A</td>
<td>0.436</td>
<td>$\pm$ 0.196 (0.038)</td>
<td>44.90 (8.62)</td>
<td>[0.045 0.828]</td>
</tr>
<tr>
<td>$C_{y_{av}}$</td>
<td>-0.169</td>
<td>0.092</td>
<td>$\pm$ 0.039 (0.009)</td>
<td>42.01 (9.77)</td>
<td>[0.015 0.170]</td>
</tr>
<tr>
<td>$C_{l_0}$</td>
<td>-0.061</td>
<td>-0.017</td>
<td>$\pm$ 0.024 (0.003)</td>
<td>140.18 (17.08)</td>
<td>[-0.065 0.031]</td>
</tr>
<tr>
<td>$C_{l_p}$</td>
<td>-0.395</td>
<td>-0.507</td>
<td>$\pm$ 0.117 (0.018)</td>
<td>23.05 (3.64)</td>
<td>[-0.741 -0.273]</td>
</tr>
<tr>
<td>$C_{l_r}$</td>
<td>0.129</td>
<td>0.184</td>
<td>$\pm$ 0.095 (0.013)</td>
<td>51.94 (6.88)</td>
<td>[-0.007 0.374]</td>
</tr>
<tr>
<td>$C_{n_0}$</td>
<td>0</td>
<td>-0.002</td>
<td>$\pm$ 0.000 (0.000)</td>
<td>16.11 (1.83)</td>
<td>[-0.002 -0.001]</td>
</tr>
<tr>
<td>$C_{n_3}$</td>
<td>0.087</td>
<td>0.090</td>
<td>$\pm$ 0.005 (0.001)</td>
<td>5.96 (0.64)</td>
<td>[0.079 0.100]</td>
</tr>
<tr>
<td>$C_{n_p}$</td>
<td>-0.056</td>
<td>-0.074</td>
<td>$\pm$ 0.021 (0.004)</td>
<td>28.73 (4.87)</td>
<td>[-0.117 -0.032]</td>
</tr>
<tr>
<td>$C_{n_r}$</td>
<td>-0.064</td>
<td>-0.108</td>
<td>$\pm$ 0.022 (0.003)</td>
<td>20.36 (3.11)</td>
<td>[-0.153 -0.064]</td>
</tr>
<tr>
<td>$C_{n_{av}}$</td>
<td>-0.063</td>
<td>-0.067</td>
<td>$\pm$ 0.007 (0.001)</td>
<td>10.71 (1.21)</td>
<td>[-0.082 -0.053]</td>
</tr>
</tbody>
</table>

R2 for Output 1: 94.14, R2 for Output 2: 88.32, R2 for Output 3: 96.93
R2 for Output 4: 90.91, R2 for Output 5: 93.71

The first example of the lateral combined input demonstrated that if the manoeuvre is
performed well it is possible to obtain very good results for most of the lateral parameters. The given results confirm the roll and yaw inertial properties with the included added mass components. Once more data of this quality is available, it is expected to gain further confidence in the results by the averaging process. The identified issue with the airspeed dependence of the lateral combined input can be relatively easy improved by scaling the time axis of the input sequence by the dynamic pressure in the UAV mainframe input generator. The problem of the roll attitude at the end of the manoeuvre is more difficult to solve, but scaling with airspeed should help with this as well. Another option to try is the orthogonal input sequence, which was used in the wind tunnel with good success. This might improve the observability of $C_{\beta\alpha}$, as well as the uncertainties on some of the other parameters.
17.4 Summary

This concludes the presentation of the preliminary results of the flight test data in the lateral axis. Acceptable results were obtained from the dutch roll response to a rudder input, which confirmed the Z-axis inertia and the expected parameter values from the wind tunnel tests. Manoeuvres including the roll axis need further work to be able to generate repeatable data. The first impressions, however, also appear to confirm the reference values and the X-axis inertia. Since reasons for the issues with the roll manoeuvres were identified, it is now simply a matter of tweaking the system\textsuperscript{1} and to fly again.

\textsuperscript{1}At the time of writing the UAVmainframe has now gained the capability to generate the orthogonal multi-sine inputs used with success in the NASA AIRSTAR project [127]. These give much better results in the wind tunnel and are expected to work well in flight as well. Unfortunately this could not be tested before the due date of this thesis.
18. Future Work

18.1 The Mysterious Nature of Added Mass

Throughout his thesis it was demonstrated that the added mass effect plays a critical role for the flight dynamics of the small aircraft under consideration. It was, however, also found that there were several inconsistencies about the nature of the added mass. In summary:

- The 1 DoF small amplitude swing tests performed for part IV resulted in measured inertias that included the added mass effect. The swinging motion is a purely rotational motion, although the rotation rates and the deflection angles are small. As shown in Figure 8.6, the angular rates during these swing tests are about 10 degrees per second and the deflection is only three degrees. The motion represents constant rotational acceleration and deceleration with respect to the surrounding air.
- The 3 DoF medium amplitude swing tests performed for part IV were also affected by the added mass, but distinctively different to the 1 DoF motion. The 3DoF motion is also purely rotational, but with much larger rates and deflections than the 1DoF swing tests, as shown in Figure 8.7. Again, this motion represents constant rotational acceleration and deceleration with respect to the surrounding air.
- The 3 DoF motion of the aircraft in the wind tunnel is not affected by the added mass. During this experiment the aircraft rotates with respect to the surrounding air, but does not translate. The air in the wind tunnel, however, translates with respect to the aircraft. This motion also represents constant rotational acceleration and deceleration with respect to the surrounding air.
- The 6 DoF free flight tests showed that the motion is affected by the added mass, and that these contributions have the same magnitude as the added mass components
of the 1 DoF swinging motion. During flight the aircraft translates and rotates with respect to the surrounding air. In all performed manoeuvres the rotational accelerations dominate, while only limited translational acceleration occurs.

All four experiments are dominated by rotational motion. This would lead to the assumption that the added mass components should be the same for all experiments, since added mass only depends on the geometry of the vehicle [24, 31]. But this is not true, as the above list clearly shows. The difference in added mass for the two swinging tests indicates that the assumption of simply adding the added mass components to the main diagonal of the inertia tensor is not correct. And the only difference between the wind tunnel and flight tests is the added translation during flight, unless the motion of the air with respect to the aircraft in the wind tunnel results in a change of the added mass properties. These arguments lead to the conclusion that the added mass matrix must contain other elements, which allow for the described phenomena to happen. The only source of information in the aeronautical literature is the topic of airship design [24], but even those references treat this topic very briefly. Airships do not accelerate quickly and therefore the simplified form of the added mass matrix seems to be valid. No literature on the added mass of an fixed wing aircraft in flight exists to this author’s knowledge.

In ocean engineering the added mass is much more commonly required to be treated in more detail. Water has a much higher density than air and therefore ships and submarines are considerably affected by the added mass. In the literature for marine vehicles, a much more general added mass matrix is used. The full, 6 DoF added mass matrix for vehicles with a single plane of symmetry, which would describe the test aircraft, is developed in reference [25] and [102] as

$$ I_{mf} = \begin{bmatrix} X_u & X_v & 0 & 0 & 0 & X_r \\ Y_v & 0 & 0 & 0 & Y_r \\ Z_w & Z_p & Z_q & 0 \\ L_p & L_q & 0 \\ M_q & 0 \\ \text{sym} & N_r \end{bmatrix} \tag{18.1} $$

where the first three rows correspond to the added mass components in the force terms in the equations of motion and the last three rows correspond to the inertia terms in the moment equations. All terms can be interpreted similarly to aerodynamic derivatives, except that the added mass derivatives all describe properties caused by acceleration, as indicated with the notation. Therefore the assumed added mass matrix $I_{mf}$ for this project is a subset of Eq. (18.1), namely the terms $L_p$, $M_q$ and $N_r$. It is not clear, however, how the extra terms in Eq.(18.1) will make any difference to the rotationally dominated motion seen throughout this project and thus the question of the true form of the added mass matrix for small, fixed wing aircraft has to be left open. It is hoped that the work presented in this thesis will lead to further research into the matter, given the magnitude of the effects that were demonstrated. Determination of the
entries of Eq.(18.1) for a complex shape like an aircraft requires either the numerical methods [25, 31] or testing in a water tunnel. An unsteady CFD solver might also enable simulations with isolated translations or rotations to potentially solve the issues described here. Other suggested areas of future work include:

## 18.2 Sensors and Equipment

Technology and developments in this sector move so fast that the current UAVmainframe prototype is already outdated. Upgrades to the system can potentially further improve accuracy and reliability:

- At the time of writing, GPS RTK systems with centimetre accuracy and 10 km range start to become affordable. These can potentially be used for attitude determination as well, which would remove the need to the difficult magnetometer sensor.
- Integrated sensor chips with up to 9 degrees of freedom (accelerometer, gyroscope and magnetometer) are becoming regularly available at decent quality. Using these gives an aligned accelerometer in each sensor package which allows to determine the sensor orientation much easier than the complicated procedures that were required for this work (Chapter 6).
- Re-formulate the EKF to enable all remaining sensors of the UAVmainframe and establish whether any improvement of the results is possible by doing that.
- Integrate the EKF into the UAVmainframe to enable real time state estimation for enhanced flight control and real time wind estimates.
- Enable engine thrust measurements to be able to do drag estimates
- What other sensing options are available for the aerodynamic angles? The airdata probe is too complicated and fragile for regular use. Options are 5-hole probes and flush airdata probes. Both need extensive calibrations. Can that be improved somehow?

## 18.3 Experiment Design

- Add a load cell to the wind tunnel gimbal to enable force measurements through the accelerometer output equations. This would then potentially provide the full set of derivatives from the dynamic wind tunnel tests.
- Research advanced input sequences to potentially de-correlate the pitching moment equation, and to produce more reliable lateral flight test results.
- Add better longitudinal control loops to further improve pitch trim and the blending between closed- and open loop flight phases.
- Or, remove the need for open loop flight altogether by implementing a control strategy which allows parameter ID, while maintaining a controlled flight path.
- And now that the process of small aircraft system ID has been shown to work, other airframes with advanced geometries can be flight tested and characterised.
19. Conclusion

This concludes the presentation of the work carried out to demonstrate the feasibility and potential of flight testing a small scale, fixed wing aircraft and to identify the aerodynamic stability and control parameters of the test aircraft accurately and reliably. The successful completion of the project required work in many different areas and can be summarised as follows:

- The current state of miniaturised electronics and computing hardware allows to create a flight data acquisition and management system that is small and light enough for a small scale aircraft with less than 5kg take-off weight. The UAVmainframe was realised with standard, commercial off the shelf equipment only and runs on an unmodified version of the Linux operating system, while performing all the time critical tasks this work required.
- A small aircraft cannot be expected to carry calibrated instrumentation of similar quality to a full scale aircraft. Size, weight and budget limit the grade of sensors available, as well as the precision of installation and alignment with the aircraft body axes. To ensure high quality, kinematically compatible flight data, an extended Kalman filter was designed to analyse and correct the raw data for the inevitably remaining errors.
- The EKF works best if all sensors are calibrated as good as possible to reduce the amount of uncertainty of the system. Procedures and methods to calibrate all sensors of the UAVmainframe were developed, including a successful procedure to determine the magnetometer installation attitude, which greatly improved the attitude estimates of the EKF.
- The custom designed airdata probe was extensively calibrated for wing upwash due to its location close to the leading edge of the wing. The PanAir solver was used to determine the wind tunnel wall interference corrections for this calibration to yield
highly accurate angle of attack measurements in flight.

- The UAV mainframe features physical control surface feedback sensors, which is a novelty on this scale. This data significantly improved the precision of the recorded control surface motion used for the parameter identification.

- The UAV mainframe also acts as a flight controller to aid the remote pilot, allowing for greater flight distances, better trim conditions and overall better productivity and safety of the flight tests.

- The test aircraft was tested on a newly designed 6 DoF, static wind tunnel balance. The new balance and the test results were benchmarked with a simulation of the wind tunnel environment using the PanAir solver. This simulation was also used to generate the wall interference corrections to enable a comparison of wind tunnel and flight data.

- Dynamic wind tunnel tests, using a 3 DoF gimbal to enable reduced model flight tests inside the wind tunnel, were used to determine the dynamic properties of the test aircraft instrumented with the UAV mainframe. System identification of the recorded 3 DoF motion was used to determine the parameters in the reduced models. The wind tunnel had very high turbulence levels, which limited the accuracy of the parameter estimates particularly in the roll axis.

- The inertial properties of the test aircraft were determined with swing tests. Significant corrections due to added mass were discovered in the process. These were identified to be caused by the volume of the test article and cannot be determined by swinging a flat plate of similar outline as the test article. Only a body with similar surface area and volume yields the correct added mass corrections.

- More than 150 flight tests were flown with the fully instrumented test aircraft to work out procedures and input designs and to collect flight data. Issues relating to the small scale of the test aircraft were the limitations on direct pilot feedback due to the remote piloting, which required the addition of the flight control system to the UAVmainframe, as well as the sensitivity of the small airframe to wind and turbulence. It was found that significant wind is almost constantly present on this scale, which resulted in the integration of a wind estimator into the EKF. Wind and turbulence levels also represent a significant challenge for the parameter identification of the flight data.

- The three most widely used parameter identification algorithms, the equation error method, the output error method and the filter error methods were all used to identify the model parameters throughout the project. Quite surprisingly did none of the methods stand out as particularly good or bad. Except for a few notable exceptions all methods resulted in similar quality of estimated parameters. The complicated filter error method did not result in better results most of the time.

- Significant correlation was found in the longitudinal axis between the $C_{m_q}$ and $C_{m_{\delta e}}$ derivatives. This prevented the parameter identification of the short period mode, unless one of the parameters was held fixed at its known value.
• Significant correlation also existed between $C_{l_p}$ and $C_{l_d}$ in the roll axis, but with adjusted input designs this issue could be mitigated.

• The results of the flight tests closely agree with the wind tunnel test data, if the added mass is added to the inertial properties of the aircraft, that is the in-flight inertias of the aircraft are larger than in the wind tunnel. This finding has important implications design and simulation of these small aircraft, with apparent changes to the longitudinal stability of 25% or more if the added mass contributions are not included.


[40] Lorenz Meier and team. “MAVLink micro air vehicle marshalling / communication library”. In: (accessed 01/07/2016). URL: https://github.com/mavlink/mavlink (cited on pages 25, 46, 50).


A. The 7x5 Feet Low Speed Wind Tunnel

The 7x5 ft. low speed wind tunnel at the University of Sydney is a closed circuit tunnel. It can reach speeds up to 35m/s inside the test section. The wind tunnel is quite old (constructed in the 1940s) and has been neglected for some years due to lack of projects and funds. The existing balance originates from the 1950s. It is a mechanical device, which offers only limited accuracy. It also requires the model to be suspended upside down from the ceiling which interfered with the needs of this project. Embedded into the test section floor, there were the remains of another balance (a rear sting type) that was ill conceived and never really worked due to overly large flow blockage and other issues.

Given this situation, it was decided to design and build a new balance with a vertical sting rising from the floor. It would hold a modern six axis load cell on top of the sting to be mounted as an internal sensor inside the model. The old mechanism in the floor would be adapted to provide actuators for pitch and yawing motion of the model. An entirely new control- and data reduction software was developed for the new system.

After the construction of the balance was complete, a detailed survey and calibration of the system was performed to gain confidence in the results. During this process, severe issues with the flow quality inside the test section were discovered. Despite a large and time consuming effort, these could not be resolved and the results continue to be affected by high turbulence levels inside the tunnel.

A.1 Facility Overview

In this section, the test section of the 7x5 ft. wind tunnel and its instrumentation is briefly introduced. Aspects discussed include the test section dimensions, the newly designed balance and its control software, the instrumentation and finally the model installation.
A.1.1 Test Section Geometry

The test section of the 7x5 ft wind tunnel is a rectangular section with chamfered edges as shown in Figure A.1. The dimensions and the position of the sting balance with reference to the walls are indicated in the Figure. The cross-sectional area of the test section is $2.99\text{m}^2$. In the Figure the equivalent elliptical section is indicated as a dashed line. This will be used for the development of the wall interference corrections in appendix B.

A.1.2 Balance

A CAD image of the new sting balance is shown in Figure A.2. The sting or main mast is a solid steel bar with stepwise increasing cross-sections and it is mounted to the tunnel floor, which is part of the turntable for the sideslip motion. The headpiece of the sting holds the 6-axis load cell and can pivot in pitch. The pitch angle is controlled by the pitch actuator. It is driven by a four-bar mechanism of the original rear-sting balance. The headpiece of the balance can be removed to install the motion gimbal for dynamic testing as discussed later. A fairing, made from foam profiles with plywood ribs, covers the mast. The pitch actuator and additionally serves as a wire channel. For powered tests, the high current wires for the electric motor can be routed away from the small signal loadcell cable in separate channels of the fairing. The fairing can rotate around the sting and is fixed to the stationary part of the tunnel floor with a bar. That way, the fairing always stays aligned with the inflow, even if the sting rotates in yaw. The final assembly is also shown in Figure A.3, with the wind tunnel test aircraft installed.
To control the balance and to perform the data acquisition and reduction, a graphical user interface (GUI) was created in Matlab as shown in Figure A.4. The GUI interfaces with the balance drive motors and the wind tunnel instrumentation via the Matlab data acquisition toolbox. A timer-loop in the background realises PID feedback controllers to be able to drive the balance to the desired angles. End switches and an emergency stop button states are read by the timer loop and appropriate action is performed depending on the state of the switches. Test points can be recorded individually or in a sequence.
mode to acquire polars. The sample rate and duration for each test point can be selected to adjust for noise and other distortions. The GUI performs the calibrations of the raw load cell data and angle transformations from the load cell axes to the wind tunnel axes and writes the results into a text file. Data recorded are the actual forces and moments, as well as the coefficients calculated from the reference data and the dynamic pressure. For each data channel, statistical information, in the form of standard deviations of the individual samples from the calculated mean value, is recorded as well.

### A.1.3 Wind Tunnel Instrumentation

The wind tunnel uses a National Instruments NIDAQ USB analog to digital interface with 16 analogue, differential channels as well as digital inputs and analogue outputs for the motor drives. Its input range is $\pm 10V$.

The load cell used for the static wind tunnel tests is a *ATI MINI 45* six axis force and moment transducer. It delivers an analogue voltage between -10 and +10 volts for each of the six channels. It comes with a factory supplied calibration matrix which has been integrated into the wind tunnel control software. The measurement ranges and resolutions for each channel are given on the webpage\(^1\) as:

\(^1\)http://www.ati-ia.com/products/ft/ft_models.aspx?id=Mini45
### A.1.4 Model Installation

The model installation of the static test airframe is shown in Figure A.3. On the floor in front of the mount the pitot tube for the reference airspeed measurements can be seen. It was removed for the final test runs.

A detailed view of the model installation is shown in Figure A.5. The main wing of the airframe has been modified to accommodate mounting points for the load cell and the motion gimbal used this project. This modification required the removal of the main spar of the wing. To replace it, two large plywood spars were installed in front and at the back of the opening and integrated into the wing structure. The modified wing structure has been tested up to loads of 4g without problems.

The model has a span of 1.5m, which corresponds to approximately 70% of the tunnel width. Significant corrections for wall interference effects are expected for this large model [88]. Since the same model is used for the dynamic tests, it is fully functional with servo motors, RC control gear and instrumentation installed. The static control surface tests have been performed by deflecting the controls using the servo motors and using the angle feedback information from the onboard instrumentation for the data processing.
A.2 Test Section Survey

As mentioned above, a survey of the test section flow quality and accuracy of the measurement instrumentation was necessary because no data was readily available from any previous test and the entire balance was new. The following sections discuss the flow quality, the load cell calibration, the airspeed measurement and finally, the balance angle measurements.

A.2.1 Flow Quality

From the start of the project it became clear that the flow inside the test section was very turbulent. This showed in the model rocking in roll on the balance due to unsymmetrical flow over the wings and heavy noise on the airdata measurements of the tunnel and the UAVmainframe sensors. In order to characterize the turbulence level and distribution across the test section, airspeed data was collected at 100 points as shown in Figure A.6. Usually, a hot wire probe would be used to gather airspeed and flow direction information [88], but such a device was not available. Measurements of the turbulence levels using drag measurements of a sphere [88] were not conclusive, because the drag of the sphere was too low for the load cell resolution, such that the transition speed could not be determined accurately. The test indicated, however, that the turbulence intensity in the test section was well above 0.5%, which is not very good for an aeronautical wind tunnel. The high turbulence levels do not affect static tests as much due to averaging of many individual samples for each test point. They are more important for dynamic tests performed for this project, especially in the lateral directions.

The grid of airspeed readings was collected using a pitot tube with the calibrated reference sensor at a nominal speed of 16m/s, as indicated on the control GUI. Faster airspeeds would cause vibrations of the pitot tube, which would have caused errors in the data. At each point, samples were taken at 10Hz for 30 seconds each. The data was mapped onto a volume grid and visualisations are plotted in Figs. A.7 and A.8.
Figure A.6: Locations for airspeed survey. Inflow is from the left. The balance location is indicated.

The mean airspeed distribution in Figure A.7 shows large variation across the test section. The color range of the Figure was selected to show more detail near the nominal speed of 16m/s. The dark blue color indicates speeds 0.5m/s and more below the nominal speed. The speed at the centre of the jet at the test section entry is actually only 13m/s. The very low speeds behind the balance are influenced by the wake of the assembly. In general, the airspeeds near the roof of the test section are higher than near the floor. Along the test section, the core of the jet at the entry is very slow, but the flow becomes more uniform near the balance location and then deteriorates to the back of the test section.

The standard deviation of the 300 individual samples at each location across the flow field is plotted in Figure A.8. This is the best indicator of how much the airspeed fluctuates and therefore the turbulence levels. At the entry and exit of the test section there are fluctuations exceeding 2m/s, that is more than 12%. Near the balance the situation is somewhat better with variations around 0.75m/s or 5%. Nevertheless, having up to 1.5m/s differential airspeed over the left and right wing of the model causes large rolling moments and this shows in the rocking of the model in roll and also in the much higher noise level for the rolling moment measurements. These have to be averaged over a longer time span to yield acceptable results. One can imagine what these flow conditions do to a model on a motion gimbal where it is free to rotate. This will be discussed later on. Originally, the UAVmainframe was to be fitted out with a five hole probe for flow angle measurements because this would be much lighter and smaller than the air vanes. But it was impossible to calibrate such a probe under these flow conditions. The noise on the
readings reduced the resolution of the probe to more than 1.5 degrees. Smaller angle changes could simply not be detected. These results indicate that the single screen in the reservoir of the tunnel is not capable of straightening the turbulent flow coming from the drive fan across two sets of guide vanes. In order to improve on this, the tunnel requires a completely new system of flow straighteners, including a honeycomb and multiple, fine screens.
Figure A.8: Standard deviation in m/s of airspeed measurements at $V_m = 16\text{m/s}$. Sampled 100 times at 10 Hz. The inflow is from the left.

A.2.2 Loadcell

The load cell had been in storage for some time. In order to test the validity of the supplied calibration matrix, tests were run in the $F_z$ and $M_y$ axes using weights. The results are shown in Figure A.9. For both axes the error between the expected reading and the actual result is less than 1%. It is expected that the other axes have a similar precision. They could not be tested without manufacturing a specialized, more complicated test rig. The remaining 1% error is much smaller than the errors introduced by other factors like the flow turbulence so no further action was taken to attempt an improvement.
The analogue load cell signals are heavily amplified signal of strain gauges and are quite noisy even outside the wind tunnel during bench testing. To obtain the data quality described above, every test point is actually an average of 2000 to 5000 samples taken at 1000 Hz. The standard deviation of five test points, using an average of 2000 samples each is about 0.05 N for a 9.81 N reading. This is better than the load cell resolution specified by the manufacturer, which was 0.125 N. Based on these test results, the standard settings used for the static wind tunnel tests were 2000 samples at 1kHz for the longitudinal cases and 5000 samples at 1kHz for the lateral cases.

### A.2.3 Dynamic Pressure Measurement

The tunnel airdata system is based on the pressure difference between the test section entrance and the reservoir. The system has been calibrated against a water manometer using a pitot tube in the test section. During the test runs the airspeed readings of the tunnel system was also checked against the calibrated airdata probe of the UAVmainframe. Given the turbulent conditions in the tunnel, (the standard deviation of the dynamic pressure reading at 230 Pa or 20 m/s is about ±6Pa or ≈ 0.5m/s) there was no apparent difference between those readings. Because the exact absolute value for the dynamic pressure is not important for the stability and control derivatives, no further action seemed necessary.

### A.2.4 Balance Angles Measurement

The aerodynamic angles $\alpha$ and $\beta$ of the balance are measured with precision potentiometers as shown in Figure A.10. The sensor for the angle of attack is located below the tunnel floor. These readings have proven highly stable and reliable.

Inside the wind tunnel control software, the voltage readings are converted into angles using a polynomial curve fit. These have been obtained by driving the balance to 15
Figure A.10: Potentiometer and linkage for $\beta$ feedback

different angles, then measuring the actual angle and the voltage at each point. Then a fourth order polynomial has been fitted to that data. A higher order curve did not improve the accuracy. Naturally, this curve fit is not perfect and small errors remain as discussed below.

The calibration was then checked by triangulation. The screen inside the reservoir of the tunnel is about 5.5m away from the balance. A laser pointer was fitted to the balance. Then the balance was driven to test points in 0.5 degrees intervals between -5 and 5 degrees on each axis. By measuring the distance the laser point travelled on the screen the angle change of the balance can be calculated. As the absolute error on the angles is not that important, only the angle differences between the test points were considered.

A first result is that the repeatability is very good. During 5 runs it was typically better than 1mm on the 5483mm distance the screen which translates to 0.01 degree.

While positions could be repeated accurately, the actual angle intervals are not that precise as shown in Figs A.11 and A.12. The figures show the error from the ideal 0.5 degree interval as commanded for each step between -5 and 5 degrees. The error is the sum of the actual position error as measured by the laser pointer and the drive controller error. As the balance consists of heavy machinery, the drive controller needs to have a dead-band of 0.05 degrees to avoid oscillations around the target point due to the inertia of the mechanism. Therefore the targeted angle is never reached perfectly.

Figure A.11 shows that for the angle of attack, the balance can be positioned with a mean close to 0.5 degree step size and a standard deviation of about 0.05 degree. The accuracy deteriorates somewhat as the angles become larger. For the sideslip in Figure A.12, the result is similar, but the mean is slightly further away from the 0.5 degree target and the data spread is larger and evenly distributed through the angle range. All these errors are very repeatable so they cannot be caused by random noise on the system. It is more likely that the following systematic errors are responsible for this behaviour:

1. Limitations of the curve fit: As mentioned above the curve fit to convert the potentiometer voltage readings to angles has a limited accuracy which could not be improved without significant effort generating lookup tables.
Figure A.11: \( \alpha \) positioning error per 0.5 degree interval

Figure A.12: \( \beta \) positioning error per 0.5 degree interval

2. The controller dead-band is necessary for reliable function of the drive mechanism. It has been chosen as small as possible but there remains a finite interval that limits the drive precision.

3. The potentiometer sample rate and size is different during drive and measuring a test point due to the fact that during the drive the controller needs constant updating of the actual position to function properly. Therefore only a smaller sample size can be used in that mode. Residual noise on the sensor might be more pronounced in that smaller sample size, affecting the drive accuracy. The angle reading during the test point sample might be slightly different due to the larger sample size used there. It has not been possible to quantify this error but it is thought to be small. Nevertheless it has been included for completeness.
Even though the angle positioning accuracy contains some errors, these have not been treated any further. Firstly, most angle interval errors are smaller than 0.05 degree and not biased. And secondly, each aerodynamic derivative itself is a curve fit to 20 or more separate data points. This reduces the error due to the balance positioning accuracy even further as the curve fit is basically an averaging process. With a mean position error close to zero, it should disappear during that curve fit.
In order to compare wind tunnel test data with free flight data, it is necessary to correct the wind tunnel data for the presence of the walls around the test section. The walls limit the extension of the flow field created by the aircraft and this changes the measured aerodynamic characteristics compared to the free flight case where no such boundaries exist [88].

There are two approaches to develop these corrections: Firstly, the classical or analytical method using horseshoe vortices and their mirror images at the walls and secondly using a potential flow solver. The first method is straight forward and all required information can be found in [88]. The second method requires the modelling of the entire aircraft inside the wind tunnel and solving for the aerodynamic coefficients with and without the walls present. The difference between the two solutions are the wall corrections that can be applied to the wind tunnel test data.

In this chapter, the classical method is used to derive the corrections and in the next chapter the numerical method is applied. The classical method is expected to be less accurate than the numerical method because it does not take into account the fully detailed geometry of the problem. It is necessary, however, to generate the classical corrections to benchmark the numerical method and to get an idea of the magnitude of corrections required.

B.1 Balance Calibrations

The first three steps of the data reduction involve calibration of the measured data as follows:

**Step 1: Generate true balance loads.** Calibrate and correct the strain gauge readout voltages to Forces [N] and Moments [Nm]. This is done by using the manufacturer’s
Chapter B. Wind Tunnel Wall Corrections

calibration matrix during the data recording process by the wind tunnel control software.

**Step 2: Calibrate** $\alpha$ and $\beta$. Calibrate and correct the voltages from the potentiometers into angles in [deg], using the generated calibration curves for the balance. This is again done during data recording.

**Step 3: Apply dynamic pressure corrections.** Apply tunnel airspeed calibrations from external pitot tube measurements. This calibration is also applied by the control software.

At this stage the calibrated coefficients for the force and moment measurements have been generated. Now the corrections due to the flow field interaction with the tunnel walls have to be determined.

### B.2 Blockage, Tares and Interference

Blockage and Tare corrections are caused by the presence of a model inside the test section and do not depend on the generation of any lifting forces. Blockage is the change in tunnel cross-section with the model installed and the Tares are due to measurement errors of the balance. Interference corrections are caused by the interaction of the flow around the test article with parts of the balance, where, for example, wakes might interfere with a fairing.

#### B.2.1 Blockage corrections

The dynamic pressure in the tunnel is typically calibrated with an empty test section. Installing a model, especially if it is large, will change the cross-sectional area of the test section. This, in turn, will influence the dynamic pressure over the model, as the flow will speed up due to the reduced area. Therefore the measured dynamic pressure of the wind tunnel instrumentation will be different from the actual dynamic pressure over the model and this must be corrected for. To investigate whether blockage corrections are significant, a first order approximation for the blockage is given in [88] as

$$\epsilon_t = \frac{1}{4} \frac{A_m}{A_{WT}} \quad (B.1)$$

where $A_m$ is the model frontal area and $A_{WT}$ is the test-section area. In this case $A_m \approx 0.096 \, m^2$ and $A_{WT} = 2.99 \, m^2$. Thus,

$$\epsilon_t = \frac{0.096 \, m^2}{4 \times 2.99 \, m^2} = 0.008 \quad (B.2)$$

The correction for the dynamic pressure is given as

$$q_c = q_m (1 + \epsilon_t)^2 \quad (B.3)$$

where $q_c$ is the corrected dynamic pressure and $q_m$ is the measured dynamic pressure. Using the result for $\epsilon_t$ the correction factor is 1.016 or 1.6%. Due to the turbulence in the tunnel the uncertainty of the dynamic pressure measurements in this tunnel is about 5-10%, depending on the speed. There will be no gain in accuracy applying the blockage correction and therefore it will be ignored.
B.3 Wall Interference

B.2.2 Weight Tares

The load cell reference point is offset vertically from the CG location of the model. As the load cell rotates with the model about the pitch axis, the gravity vector in earth axis is split into a component in the $F_x$ and $F_z$ body axes measurements of the load cell. This mainly distorts the drag readings. To remove this, a wind-off pitch run for the appropriate angle range is saved and subtracted from the wind-on measurements by the control software.

B.2.3 Interference

It has not been possible to perform interference runs with the model installed upside-down as suggested in reference [88] due to cost and time constraints. These corrections are believed to be small for the stability and control derivatives and therefore have not been treated any further.

B.3 Wall Interference

The wall interference causes a change in angle of attack seen at the model main wing leading edge as well as a change in tail angle of attack. The first effect influences the value for $C_{L\alpha}$ and $C_{m\alpha}$, while the change in tail inflow angle only affects $C_{m\alpha}$.

B.3.1 Wall Interference Corrections for Angle of Attack

The correction of the angle of attack due to the wind tunnel walls is given in reference [88] as

\[ \alpha_c = \alpha_m + \Delta \alpha_w \] (B.4)

\[ = \alpha_m + \delta \left( \frac{S}{C} \right) C_{LW} \] (B.5)

\[ = \alpha_m + C_1 \times C_{LW} \] (B.6)

where $\alpha_c$ is the corrected (real) angle of attack in radians, $\alpha_m$ the measured angle in radians, $\delta$ the wall correction factor, $S$ the model wing area, $C$ the tunnel cross-sectional area and $C_{LW}$ the measured wing-body (tail off) lift coefficient. $C_1$ is shorthand for the angle of attack correction factor to be used in the GEODEUDE software for post processing and later in the pitching moment correction.

The value of $\delta$ is determined from charts in reference [88] and the geometric data in Table B.1 as follows. Because of the rectangular wing, the true wing loading of the test aircraft is not elliptical and its true shape is unknown. The reference suggests to use charts for uniform loading with a correction to the wing span to account for the vortex rollup behind the wing. This effective span $b_e$ can be found from

\[ b_e = \frac{b + b_v}{2} \] (B.7)
<table>
<thead>
<tr>
<th>Variable</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>model span</td>
<td>$b$</td>
<td>1.53</td>
</tr>
<tr>
<td>model wing area</td>
<td>$S$</td>
<td>0.425</td>
</tr>
<tr>
<td>wing AR</td>
<td>$AR$</td>
<td>5.5</td>
</tr>
<tr>
<td>taper ratio</td>
<td>$\lambda_w$</td>
<td>1.0</td>
</tr>
<tr>
<td>tunnel area</td>
<td>$C$</td>
<td>2.99</td>
</tr>
<tr>
<td>tunnel h/w</td>
<td>$\lambda$</td>
<td>0.714</td>
</tr>
</tbody>
</table>

Table B.1: Data required to determine $\delta$

The value of $b_v$ can be read off Fig 10.11 in reference [88] with the data in Table B.1 to be

$$\frac{b_v}{b} = 0.87 \quad (B.8)$$

Thus, the effective span is

$$b_e = \frac{b + 0.87b}{2} = 1.43 \text{m} \quad (B.9)$$

The value of $\delta$ can be found from charts for an uniformly loaded wing inside a elliptical or rectangular tunnel. The correct choice is not clear from the book. As shown in Figure A.1, this tunnel test section is a rectangle with fillets in the corners to form an octagonal. This brings it closer to the elliptical shape. But is is not known, which dimensions of the fillets are required to make the elliptical test section shape valid. Thus, both shapes will be calculated and then compared to make a choice. For the elliptical test section Fig 10.28 in reference [88], using a ratio of effective span/jet width $k = 0.671$ and a ratio of tunnel height/width $\lambda = 0.714$, yields:

$$\delta_{\text{ellipt}} = 0.105 \quad (B.10)$$

$$\Delta \alpha_{w,\text{ellipt}} = 0.0149C_{LW} \quad (B.11)$$

for angles in radians. Similarly, for the rectangular test section using Fig. 10.17 in reference [88], with the same $k$ and $\lambda$, gives:

$$\delta_{\text{rect}} = 0.123 \quad (B.12)$$

$$\Delta \alpha_{w,\text{rect}} = 0.0175C_{LW} \quad (B.13)$$

for angles in radians. This is a significant difference of about 15%. In the text an example is given for a $8x12$ ft tunnel with 1.5 ft fillets. It is treated as a rectangular tunnel. This tunnel is $7x5$ ft with approximate 1 ft fillets, which is more significant to the tunnel shape. Using preliminary test data, the two corrections in the end cause only 1% difference in the corrected lift curve slope. This is less that the measurement uncertainty.
As a compromise, lacking better data, the mean between the two tunnel shapes will be used, as the tunnel is clearly a blend between them. So the final correction for the angle of attack is

$$\delta_{7x5} = 0.114$$  \hspace{1cm} (B.14)

$$\Rightarrow C_1 = 0.0162$$  \hspace{1cm} (B.15)

$$\Rightarrow \Delta \alpha_{\mu,7x5} = 0.0162 C_{LW}$$  \hspace{1cm} (B.16)

### B.3.2 Wall interference corrections for $C_m$

In addition to the change in angle of attack, the modification of the streamlines by the tunnel walls also cause an additional tailplane inflow angle change. This affects the lift generated by the tail and therefore the magnitude of the pitching moment. The correction to the pitching moment due to the wall interference is given in reference [88] as:

$$C_{m,CG,c} = C_{m,CG,meas} - \Delta C_{m,CG}$$  \hspace{1cm} (B.17)

$$= C_{m,CG,meas} - \delta \left( \frac{S}{C} \right) \left( \frac{\delta C_{m}}{\delta \epsilon} \right) \tau_2 (57.3) C_{LW}$$  \hspace{1cm} (B.18)

$$= C_{m,CG,meas} - C_1 \times C_2 \times C_{LW}$$  \hspace{1cm} (B.19)

where $C_{m,CG,c}$ is the corrected (real) pitching moment about the CG, $C_{m,CG,meas}$ the measured pitching moment, $\delta C_{m}/\delta \epsilon$ the change in pitching moment with elevator deflection on a tail-on model, $\delta$ the wall correction factor from above, $\tau_2$ the downwash correction factor, $S$ the model wing area, $C$ the tunnel cross-sectional area and $C_{LW}$ the measured wing-body (tail off) lift coefficient. $C_1$ is the angle of attack correction factor from above and $C_2$ the abbreviation for the pitching moment specific terms to be used in GEODUDE.

To determine $\delta C_{m}/\delta \epsilon$, pitching moment curves have been measured at five different elevator settings over the standard angle of attack range [-5..5] degrees as shown in Figure B.1. The change in pitching moment $C_m$ due to the change in elevator setting $\delta \epsilon$ has been evaluated at five different angles of attack by averaging the change between the five elevator settings. The result is

<table>
<thead>
<tr>
<th>AoA [deg]</th>
<th>-4</th>
<th>-2</th>
<th>0</th>
<th>2</th>
<th>5</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{\delta C_{m}}{\delta \epsilon}$ [deg]</td>
<td>-0.0209</td>
<td>-0.0204</td>
<td>-0.0199</td>
<td>-0.0194</td>
<td>-0.0187</td>
<td><strong>-0.0198</strong></td>
</tr>
</tbody>
</table>

The downwash correction factor $\tau_2$ can then be obtained from charts in reference [88]. The required data is

- Tail length $l_t$, the distance between the two aerodynamic centres, here $l_t = 0.665m$
- The tail height with respect to the wing, which is 0.081m or 0.066e above the wing

Again, there is the choice between elliptical and rectangular cross-sections. For the elliptical tunnel with a closed test section and a tail length to tunnel width ratio $l_t/B = 0.31$, Figure 10.38 in reference [88] gives
Chapter B. Wind Tunnel Wall Corrections

![Graph showing test data for C_m wall corrections with varying elevator settings](image)

Figure B.1: Test data for $C_m$ wall corrections with varying elevator settings

$\tau_{2,\text{ellipt}} = 0.77$  \hspace{1cm} (B.20)

$\Delta C_{m,CG,\text{ellipt}} = -0.874C_{LW}$  \hspace{1cm} (B.21)

For the rectangular, closed test section an interpolation between Figures 10.39 and 10.40 in the reference is used because the tail is $0.06b$ above the wing and charts are only available for the tail in line with the wing and $0.1b$ above. Using the same values for $k$, $\lambda$ and $l_t/B$ as before:

$\tau_{2,\text{rect}} = 0.75$  \hspace{1cm} (B.22)

$\Delta C_{m,CG,\text{rect}} = -0.817C_{LW}$  \hspace{1cm} (B.23)

As with the angle of attack correction, the mean between the two values will be used for this tunnel:

$\tau_{2,7x5} = 0.76$  \hspace{1cm} (B.24)

$\Rightarrow C_2 = -0.8623$  \hspace{1cm} (B.25)

$\Rightarrow \Delta C_{m,CG,7x5} = -0.014C_{LW}$  \hspace{1cm} (B.26)
The wall corrections for the test aircraft inside the 7x5 ft wind tunnel were determined in this chapter using the classical method based on mirrored images of a horseshoe vortex. The results are repeated below for reference.

\[ \Delta \alpha_{w,7x5} = 0.0162C_{LW} \]

\[ \Delta C_{m,CG,7x5} = -0.014C_{LW} \]

These wall corrections will be compared to the corrections obtained from the virtual wind tunnel in appendix C and then they will be applied to the wind tunnel test data to enable a comparison with the flight test data later on.
This chapter is about the numerical simulation of the wind tunnel environment developed for this project. The development of this virtual wind tunnel (VWT) was motivated by the need to benchmark the all new wind tunnel instrumentation as well as to perform some investigations that are not possible or very difficult to do in reality. These include the flow field in front of the leading edge of the wing for the airdata calibrations as well as investigations into the wing downwash coefficient at the tailplane for the $\dot{\alpha}$ derivatives. The most important application, however, was the ability to calculate the interference effects of the wind tunnel walls on the test data to enable comparison with the flight data later on.

The chapter starts with the selection of the numerical solver to be used, based on the requirements formulated in this chapter. The primary code selected is PanAir. Most of the remaining chapter will cover this solver. The AVL solver is mentioned as a secondary tool. Its contribution is based on the capability of estimating dynamic derivatives and it will be used to show that there is no need to develop any wall corrections for these parameters. This is done by simple comparison and a very minimal model, so the coverage of this part will be brief. The workflow used to interact with the solver and how the results are processed is described. Further on, the modelling of the wind tunnel in the code is introduced, followed by the models of the test aircraft itself. The PanAir results are then benchmarked against wind tunnel test data and finally the wall corrections for the static derivatives obtained from the computations are presented. Finally, the AVL solver is used to show that there is no significant difference in the dynamic derivatives between the wind tunnel and free air case and no further action is required. All findings are summarised at the end of the chapter.
C.1 Solver Selection

The VWT is based on the PanAir [129] high order panel code which uses linear potential flow theory. The PanAir solver was selected based on the following criteria: its ability to model internal flow (the wind tunnel is essentially a duct flow problem); ability to report off-body flow field properties for airdata probe corrections; ability to use a detailed fuselage model; ability to easily modify the geometry for perturbation tests; expected accuracy for the given problem based on literature review and own test cases; short running time; availability on an affordable research license in Australia; and good documentation. Panel codes have been used to model the wind tunnel environment since they became readily available with increasing computing capabilities [88], mainly to determine wall interference corrections. Since they allow the modelling of the actual geometry of the wind tunnel and the test article more precisely, they potentially offer a better accuracy than the traditional methods based on a single horseshoe vortex [88]. Early studies, [88], used relatively simple, planar methods to investigate this new method of generating wall interference corrections.

Further development enabled the next generation of codes, such as PanAir [129], allowing for more complex three dimensional geometry modelling. Available publications on PanAir used for wind tunnel problems include a general description of an automotive application with a race car model surrounded by a wind tunnel [130], an investigation on how to best model a wind tunnel duct in PanAir, covering the geometry meshing and the boundary conditions [131] and finally a study of a fighter jet in a wind tunnel at NASA Ames Research Centre [132]. This last paper reported on generating wall corrections for a detailed wind tunnel model and presented the results, something the other references did not do in a complete fashion. The major difference of NASA Ames case to the 7x5 wind tunnel used for the current work is that the tunnel at NASA Ames is much larger compared to the model size. Therefore the resulting corrections are smaller than what is expected for the 7x5 wind tunnel. Reference [132] also contains a test case to benchmark the wall modelling which will be used later on. Another wind tunnel modelling case is reported in [133], but this reference focused mainly on vented tunnel walls and their treatment in PanAir. It demonstrates that very complex cases can be investigated with solvers like PanAir. Other codes were used in [134, 135, 136, 137, 138], demonstrating that the panel code method in general is useful to generate data for the wall interference correction. None of the cases, however, involved a relatively large, standard configuration airframe in a low speed wind tunnel.

The choice of the PanAir code was based on reference [139], which contains a list of codes available at its time of writing (1990). Most are not currently available on a research license, with only Panair (A502) [140] and VSAERO [141] obtainable. VSAERO would be an interesting choice as it comes with a mesh generator and is the only code capable of predicting the wake roll-up behind the model [139]. Unfortunately, it was not available to the author at the time of this project.

Since 1990, most development effort has concentrated on CFD solvers so there is no new advanced panel method publicly available. The only new entries are AVL [105] and Tornado [106] [142], which are planar vortex lattice methods not capable of modelling
complex 3d geometry. AVL has recently gained the capability to model wind tunnel walls [143], so it is still useful as a second opinion because it is very easy to use. AVL, however, struggles with the prediction of the pitching moments caused by the fuselage. These are important for the aircraft under investigation, so AVL was not suitable as the main solver for the VWT. Tornado cannot model wind tunnel walls. Neither of the two vortex lattice methods can solve for off-body points required for the airdata probe calibration.

This left the PanAir code as the only viable option fulfilling all the specified requirements (except for the ease of use part). The solver code is freely available [140] with many support documents [129, 144, 145]. On the other hand, it requires considerable work as no pre- or post processing tools are readily available.

It should be mentioned that the PanAir solver was chosen carefully and its use for this case can be justified over the use of a Navier-Stokes CFD solver. Whilst there have been significant developments of CFD solvers in recent times, a number of drawbacks still remain, which favours the use of panel methods for this application. For example, the mesh generation for a full aircraft model in a CFD analysis is time consuming and difficult (especially as a new mesh is required for every angle of attack change considered in the simulation which then also has consequences in terms of benchmarking the solution), the solution times on typical hardware are still in the magnitude of hours or days for the complex case required here, and every problem setup needs to be benchmarked against known experimental data which makes gaining confidence in the CFD solution a difficult process. Some of these drawbacks may be less critical for this problem because one could probably run a inviscid solution that does not require detailed boundary layer and turbulence modelling, but this has not been tested. The mesh generation effort is still prohibitive as is the solution convergence time. As the model is required to rotate inside the tunnel, each new angle requires a new mesh (or a complicated multi zone mesh), instead of just an inflow angle change for an external flow problem. This was considered too time consuming for the required investigations.

**C.2 Workflow**

The PanAir solver code is a command line executable reading in a plain text input file and returning a very large output file. All options have to be set up in the input file which also contains the mesh of the geometry to be analysed. PanAir is written in Fortran77 and the input file has very stringent formatting requirements originating from the era of punch cards, making it very error prone when edited by hand. A single character in the wrong column will cause the solver to fail.

A graphical application called ‘GEOdude’ was developed in Matlab to generate the PanAir input mesh and to process the output files. The application was inspired by [146], a very impressive work which was developed under an industry contract and is not available elsewhere. Some of the user interface is shown in Figures C.1, C.2 and C.3. GEOdude can also generate input files for the AVL solver from the same geometry definition to run cases quickly in different codes. All output files are processed and translated into the same file format that is used for the newly developed wind tunnel
balance, leading to results being easily plotted together for comparison.

The workflow in Geodude is as follows: A new project is started by choosing the required aerofoil sections for the geometry definition. The aerofoils are prepared in Xfoil \cite{147} and saved in the standard coordinate file format. These are imported into Geodude and then processed and made available across the application. Future work will include basic aerofoil manipulation tools to change thickness and camber and to add a flap deflection. This will enable most common modifications directly in Geodude. The next step is to open the geometry editor in Geodude, shown in Figure C.1. The lifting surfaces of the model are defined using the AVL input format. This format is easily adapted to a tabular representation. The AVL format uses a stack of aerofoil sections to define a lifting surface. Each aerofoil is placed by specifying the leading edge location, together with the chord length and the twist angle. The lifting surface is the lofted linearly between those aerofoils as shown in the Figure. For the 3d PanAir case, an endplate is automatically added (shown in green) to close off the wing volume. Surface normal vectors can be plotted to check the correct orientation of the mesh. Fuselage geometry is currently only supported as hard coded scripts. It was beyond the scope of this project to write a user interface that can handle all conceivable fuselage geometries. A geometry script can be selected to add the test aircraft fuselage for example, and another to add the 7x5 ft. wind tunnel geometry. The wake geometry is also specified in those scripts, taking into account the various geometry intersections between wing wakes and aircraft body.

The solver interaction window, shown in Figure C.2 for the PanAir solver, uses the mesh generated by the geometry processor, including the wakes, and features input fields for the solver configuration, like the inflow angles and reference geometry data. For
a longitudinal case, the geometry can be defined as a half model to reduce computing
time of the solver. For lateral cases the full aircraft mesh is required, which is generated
by mirroring the input mesh about the x-z plane. All data is then written into the PanAir
input file format, ready to be run by the solver.

Once the solver has produced an output file, it can be read back into the Geodude
software and processed in several ways. The application reads in all the surface pressures
from the solution and maps it onto the input mesh for visualisation as shown in Figure
C.3. Each PanAir output file contains up to four solutions at different inflow angles,
depending on the setup. Each solution can be selected and visualized. The data can then
be saved in the standard Geodude data format, which is identical to the wind tunnel
balance data format. That way experimental data and numerical results can directly be
plotted together using the same code. Geodude allows to plot up to three data sources
simultaneously, which has been extensively used to compare AVL and PanAir results to
the wind tunnel test data.

Figure C.2: Geodude case input panel with PanAir mesh and wakes
C.3 Modelling the Wind Tunnel Environment

Modelling the wind tunnel environment requires the simulation of an internal duct flow, where the test article is placed inside the duct (the test-section). An internal flow problem usually uses a closed control volume. One wall is used as an inlet with an adequate boundary condition that provides the required mass flow into the control volume. Another wall is configured as the exit. This is in contrast to the typical aircraft aerodynamics problem, which requires a solution of an external flow case. Here the test article is placed in an uniform onset flow field without boundaries.

To simulate the internal flow inside the tunnel test section with an external flow solver such as PanAir, an open duct, modelled by two-dimensional surface panels, is placed into the onset flow [131, 132]. The duct is open at both ends such that the flow can go through it. There is usually a small disturbance at the entry where the flow field adjusts to the presence of the duct, so the inlet needs to be long enough for these disturbances to die out. After that there will be similar conditions in the duct as if an inlet wall was used. Conservation of mass requires that the air that enters the duct at the front has to exit it at the end. Changes in cross-sectional area of the duct thus influence the airspeed along the length of the duct, similar to a real wind tunnel. A challenge for panel codes is leaking of the walls or loss of mass flow along the duct which depends on the code formulation and the boundary condition used for the duct/tunnel walls [132].
The duct needs to extend forward into the flow for about 20 chord lengths [132], for the reasons mentioned above. The length of the duct behind the model should be even longer to capture the influence of the walls on the wake of the aircraft. A wake is attached to the duct walls to enforce the Kutta condition [132], although this is probably not necessary if the duct is long enough (≥ 40 chord lengths) behind the test article.

PanAir has the capability of measuring the airspeed in the off-body flow field at specified locations. This enables the verification of the duct flow modelling as described next.

### C.3.1 Verification of Internal Flow Modelling

To verify the model of the wind tunnel walls, a 4:1 converging duct has been modelled as suggested in [132]. Figure C.4 shows the input mesh for the duct. The walls are modelled as two-dimensional panels with increased panel density in the contracting region. A wake has been attached to the end of the duct. The duct is immersed in the uniform inflow coming from the left in the Figure. The flow on the symmetry plane of the duct is expected to be approximately two-dimensional. Points for flow field property evaluations have been inserted along the centre line of the duct to investigate the flow velocities along the duct.

![Figure C.4: Panair input network for the 4:1 duct with exit wake](image)

Figures C.5 and C.6 show the results of the simulation. The pressure distribution in Fig. C.5 indicates high pressure at the inlet of the duct and ambient pressure after the contraction section. From Bernoulli’s law this corresponds to slower than ambient airspeed at the inlet and equal to ambient airspeed at the exit. As the flow is incompressible (M=0), this is required to fulfil the conservation of mass law. Since the area of the duct is four times larger at the inlet than at the exit, theory predicts a velocity drop from $V_0$ to $0.25V_0$ at the inlet and an acceleration to ambient airspeed in the
Chapter C. Virtual Wind Tunnel

contraction region. The closer the solution will be to these expected values the smaller
the wall leaking in the simulation. For PanAir, with its high order panel formulation, this
is expected to be minimal.

![Diagram](image)

Figure C.5: Pressure coefficient $C_p$ results for simulated duct flow

Figure C.6 confirms the expectations from the pressure distribution. The flow deceler-
ates to $0.25V_0$ in front of and inside the inlet and accelerates in the contraction region to
ambient speed. It is interesting, how far in front of the duct the airspeed is influenced
by the obstacle. At minus 14 meters the speed starts to drop, initially slowly but with
increasing rate until at minus 2m, where the main deceleration takes place until reaching
a stable speed 2m into the duct. The return to ambient speed is complete less than 1m
after the contraction stops, so the rest of the duct makes no difference to the flow. In
this case the flow at the exit will be perfectly aligned with the ambient flow direction
(this is inviscid flow without boundary layer) and the wall wake is not necessary. It has
nevertheless been kept in the model.

This verifies the modelling of the wind tunnel walls, since the pressure and velocity
distribution matches the theoretical predictions accurately. There are no mass flow
losses along the duct and the flow field is simulated as expected. Streamlines as shown
in [132] have not been computed due to time constraints. They would show the same
shape as in the reference, given the velocity distribution inside the duct matches the one
calculated in the paper.

C.3.2 Model of the 7x5 Tunnel

The model of the 7x5 ft. wind tunnel consists of four regions as shown in Fig. C.7:
The Inlet on the left, followed by the Test section with the balance fairing, the outlet or
C.3 Modelling the Wind Tunnel Environment

The cross-section of the duct was kept constant using the dimensions of the test section. Since PanAir works in non-dimensional coefficients and inviscid flow, there is no reason to model the conditions in the real contraction- and diffusion sections of the tunnel to simulate the correct airspeed in the test section.

The dimensions of the test section were modelled including the corner fillets to obtain the correct cross-sectional area for the blockage estimate. The model support with its fairing was also included to take into account possible interference effects. The fairing is a symmetrical NACA 0024 section and always aligned with the flow. Therefore it will not generate any lift and a wake is not necessary. This removes the possibility of the model wake overlapping the fairing wake at high angle of attack, which would have caused problems in the case setup. Also, to use a half model for the longitudinal cases, there cannot be a wake network on the symmetry plane. It would be duplicated onto itself during solver start-up and cause the solution to fail.

The length of the inlet is 2 m, the test section with higher density panelling is 3 m and the outlet length is 5 m. Wakes were attached to the exit. In practice, their presence did not change the solution as the tunnel outlet is long enough. The flow solution of the empty tunnel in Fig. C.7 verifies the correctness of the model. The walls display zero pressure all the way and no significant disturbances at the inlet and outlet of the duct. The pressure distribution on the fairing is symmetric and the resulting forces are zero. The presence of the fairing causes some flow disturbances on the tunnel floor. The tunnel mesh consists of 1653 panels (including the wake).

PanAir normally sets the flow angles by adjusting the onset flow field direction. This approach does not work with the wind tunnel duct because the flow direction always has...
Chapter C. Virtual Wind Tunnel

Figure C.7: 7x5 wind tunnel mesh and flow solution without model, inflow is from the left (Cp indicated)

to be aligned with the walls. There are two ways of setting the angle of attack on the model while keeping the flow aligned with the tunnel walls. One could rotate the model inside the duct and keep the onset flow at a constant angle (as done in the real tunnel) or set the onset flow at an angle, keep the model horizontally fixed and rotate the tunnel walls around the model to align with the flow. Both methods are equivalent but the later approach has advantages for the grid generation code used here. It is much easier to rotate just the few wall networks than the complex model of the test aircraft.

Figure C.8 shows the wind tunnel model used in AVL to test the wall influence onto the dynamic derivatives, which is not possible in PanAir. The wind tunnel panelling is based on an example provided with AVL adjusted for the 7x5 tunnel dimensions. It does not seem to require a high panel density around the model, if the example is correct. Since this was used only for a short test, no further work was done on this model. Wakes are automatically added in AVL and don not have to be defined in the input mesh as it is the case with PanAir.

In the case definitions for both solvers the resulting forces and moments on the wind tunnel model were excluded from the final summation. This allows to compare the model forces and moments between the tunnel and free air scenarios.
C.4 Aircraft Models

The physical wind tunnel model of the test aircraft has a modular structure. This enabled a component build up test strategy to investigate the influence of the different aircraft components onto the final forces and moments. The PanAir models of the aircraft have been designed to reproduce that modularity. This allows to compare the test results for the individual components with the simulation to benchmark the outcomes and to identify at which stage errors might occur. Before this is described further, however, the general meshing strategy is discussed next.

C.4.1 Meshing Strategy

PanAir works with a rectangular panel mesh. While this is quite straightforward to generate for a simple wing or a body alone, a wing-body intersection is difficult due to the complex geometry involved. Therefore the meshing capabilities in Geodude for a wing alone were developed first. As mentioned above, the geometry definition in the AVL format is used to define lifting surfaces for PanAir as well. This enables quick tests to be run in AVL before the more time consuming PanAir runs are done.

A PanAir wing mesh consists of a three networks, a top and bottom surface plus an endplate to close off the 3-D volume. The mesh is required to be ‘watertight’, with no gaps allowed. PanAir is fairly robust with respect to mesh quality as demonstrated in Figure C.9 where a randomly panelled sphere was used to calculate the velocity distribution around it. There is good agreement between the results and the theoretical velocity distribution despite the highly distorted panels on the sphere. This robustness is a result...
Figure C.9: PanAir meshing study: a randomly panelled sphere with calculated velocity distribution vs. theory. Source: [148]

of the high order panel formulation, where each panel is divided into several sub-panels [139, 145]. This feature ensures continuity across the panel boundaries, a property required mainly for the supersonic flow solutions. As a side effect, this results in the demonstrated meshing quality robustness and also in the non-existing leakage in the duct walls in Section C.3.1. The high order formulation also makes the code more robust against wake-surface interactions. In a vortex lattice method like AVL, a wake from the main wing needs to be of similar spacing as the tailplane panelling if the wake passes closely to the tailplane. Otherwise the solution will fail [143]. In PanAir, this wake spacing has no effect on the solution as demonstrated during preliminary testing.

Brief testing was conducted on the wing meshes to determine the best compromise for the panel density of the wing and the tail. In the chordwise direction the minimum panel number is 11 on each side and an exponential distribution is applied to obtain close leading edge spacing. On a coarser mesh the results change with panel number. Above 11 panels, only the curve offsets of the derivatives \( \left( C_{L_0}, C_{m_0} \right) \) vary while the slopes stay constant. So, for the purpose of this project, the chordwise mesh density can be regarded as converged with 11 panels or more on each side of the wing. Spanwise, a sinusoidal distribution was used with increasing panel density at the wing tips as shown in Fig. C.12. A higher panel density was found to make no difference to the results.

The fuselage mesh was hard coded in a script, starting with the wing-fuselage intersection. This is shown in Figure C.10. The script picks up the coordinates of the inboard edge of the wing mesh. These edges are then used as a starting edge for the intersection mesh. The intersection mesh is designed to fill the space between the wing root and the body. The body mesh connects to the intersection along straight lines which are relatively easy to handle. As shown in the Figure, the test aircraft wing has a low wing which forms parts of the body floor. This part is also generated by the intersection code, filling in the gap near the leading edge and extending the horizontal mesh up to the centre-line. No detailed studies into the fuselage mesh density were performed as
this was too time consuming. The fuselage panelling was mainly dictated by the manual meshing procedure. There are a few spots where there are large jumps in panel size and aspect ratio, but these could not be avoided without major re-work. Here the robustness of the code with respect to mesh quality was of great advantage.

The tailplane is a mid-wing at an incidence of two degrees. The intersection mesh, as plotted in Figure C.11, is symmetrical above and below the tailplane and again interfacing with the body mesh along straight lines. PanAir allows for the mesh to be discontinuous along straight lines, which means that the panel density along a straight line can be different, as long as the edge still forms that exact straight line to achieve the watertight mesh. This fact has been used extensively for the two intersection meshes and considerably reduces the amount of ‘bookkeeping’ required during the meshing process. As discussed in the next section, this simplification does not seem to have a detrimental effect on the solution quality.

The body mesh was created by lofting across several cross-sections taken from the physical model and picking up the edges of the intersection meshes. The nose section and endplate on the tail of the body close off the fuselage volume. The fin was meshed using 2-d camber line panels suitable for modelling thin aerofoils.

Wakes from the lifting surfaces were modelled as straight filaments along the x-axis. The body base wake models a region of separated flow behind the body base, using the appropriate boundary conditions. The base wake is connected to the tailplane wake as required. A quite complicated piece of geometry is the wing-body intersection wake. Its mesh is required to connect to an edge of the body mesh as well as to the edge of the wing wake. To create it, a body edge was traced up to the body base and network edge coordinates collected. Then, using those edge coordinates, the body mesh was extended to intersect with the wing wake. The resulting network forms the intersection wake as shown in Figure C.11. Figure C.12 shows a half model of the completed mesh of the full test aircraft.
C.4.2 Geometry Modifications

Some of the test aircraft components required some modifications to keep the workload for the mesh generation reasonable. These will be discussed in this section, while comparing a photo of the real object with the PanAir mesh used for each. All the component tests except the full aircraft involved only longitudinal tests for benchmarking the PanAir
solutions. The lateral effects of these modifications were assumed small and have not been tested.

**Wing Only (W)**

The main wing by itself was difficult to mount on the wind tunnel balance due to the large cutout for the mount in the centre of the wing as well as due to all the wiring coming out of the wing root from the sensors installed in the wing. These wires had to be taped down and faired during the tunnel runs. Note the cardboard fairing in Fig. C.13(a). This fairing was far from ideal but time constraints did not allow a better solution. The wing was also tested with the airdata probe installed.

![Wing Only (W)](image)

**Figure C.13: Wing only real and model**

The PanAir model does not have the mid wing cutout. This might lead to errors in the lift data as the model has effectively a larger area. The aerofoil of the centre wing is also the ideal one, not matching the shape of the fairing on the real wing. The real wing has the angled Hoerner wing tips which were replaced with square wing tips on the model. This removed the difficulty of rotating the tip aerofoil about the x-axis. There is no significant difference between the two wing tip shapes anyway as demonstrated in [149], so this modification is not expected to cause any error. The ailerons were set and fixed into the neutral position and have not been modelled in PanAir yet.

**Wing with Tail (WT)**

For wing and tail without the fuselage, shown in Fig. C.14, a temporary beam was mounted between the two lifting surfaces which was not modelled. The beam will have no effect on the lift or pitching moment or the wing/tail combination. The wing opening and the sensor wiring was covered as before with cardboard strips. The position of the tail in the wing downwash was reasonably correct, with some tolerance due to the beam flexing under load. The elevator on the real tail was locked into the neutral position. No control surfaces were modelled in PanAir.
Wing with Fuselage (WB)

The main wing with the fuselage alone and no tail was easy to realise on the model as the tail is removable for transport. The openings in the rear of the fuselage were covered up to clean up the geometry. In the PanAir model the tailplane intersection was meshed with a filler to close the fuselage geometry. In Fig. C.15 shows that the PanAir model does not have a fin since it does not have any influence on the longitudinal tests.

Full Aircraft (WBT)

Figure C.16 shows the tail section of the test aircraft and the corresponding PanAir model. The main difference between the two is the fuselage rear section, which was changed to make the meshing process easier. On the physical model, the fuselage is wedge shaped.
and ends in front of the single elevator, which extends all the way across the tailplane. In the PanAir model, this would have required for the fuselage base network to be embedded within the tailplane and for the base wake to extend across the elevator. The difficulty then would be to align all the mesh edges (or abutments in PanAir terms) of the base plane and base wake with the tailplane mesh. To avoid this, the fuselage was extended to the trailing edge of the tailplane and the cross-section was kept constant as shown in Fig. C.16. This modification simplified the fuselage-tail intersection significantly and allowed for the fuselage base section to be meshed independently. The tailplane span was adjusted to correct for the area lost at the root. The fin was meshed using two-dimensional surface panels typically used for thin aerofoils with less than 12% thickness[145]. The fin of the physical model is a flat plate, so this simplified panelling method is adequate. Because the fin is located on the symmetry plane, it has to be removed when running longitudinal cases with a half model. Otherwise the fin would be duplicated onto itself and PanAir will fail.

Figure C.16: Full A/c real and model
Full Aircraft (WBT) in the 7x5 Tunnel

Figure C.17 shows a half model the Piper UAV model inside the 7x5 wind tunnel model. The tunnel is rotated to generate an angle of attack of 5 degrees. The rotation point is the load cell mount such that the plane position within the tunnel is correct. The wake system of the UAV model is extended automatically by PanAir along the inflow direction. This ensures that the wake filament stays aligned with the tunnel walls. The final mesh for the half model consists of 3936 panels (including wakes) and takes about 5 min to run on a 2014 standard desktop computer.

Figure C.17: PanAir Wind tunnel simulation: Input network
Figure C.18 shows a full solution with the surface pressure coefficients mapped back onto the mesh after processing the PanAir output file. The discontinuities in the interpolation are caused by network boundaries not treated properly by the post processing code and do not affect the resulting forces and moments. It is interesting to observe the large pressure differences at the front of the canopy and how the spanwise low pressure region is continuous across the fuselage. The fuselage of this UAV has a large influence on the pitching moment in particular as will be shown later. The visualisation of the surface pressures helps understanding these findings.

Figure C.18: PanAir pressure coefficient $C_p$ solution for the full aircraft ($\alpha = 0^\circ$ and $\beta = 2^\circ$)

To verify the Panair solution, the results were compared to longitudinal static wind tunnel data. This was done in four steps. Firstly the main wing in isolation, secondly the wing with the horizontal tail and with the fuselage replaced by an aluminium beam, thirdly the main wing with the body but no tail and finally the full Piper UAV. This method allows to judge the influences of the separate components on the final forces and moments. It also helps to isolate the cause of potential errors.

During preliminary testing there was always a large discrepancy in the lift curve slope between the numerical solutions and the test data. This was identified to be caused by
the low Reynolds number flow over the model in the wind tunnel as discussed next.

### C.5.1 Inviscid vs. Low Reynolds Number Flow

PanAir is an inviscid solver. It does not handle the effects of low Reynolds numbers because those occur inside the boundary layer which is not simulated by PanAir [139, 149]. As shown in [149], the presence of the boundary layer effectively changes the shape of the aerofoil, increasing the thickness and reducing the camber. This will change the lift and pitching moment of the aerofoil for a given angle of attack.

![Aerofoil shape change due to boundary layer displacement](Source: [139])

This needs to be taken into account when comparing wind tunnel results at low speeds with numerical data, even though reference[88] states that ‘the effect of Reynolds number on the lift curve slope is typically small’ and demonstrates it with some graphs between Reynolds numbers of 8 million and 350,000 (approximately the Reynolds number over the main wing at \( V = 20 \text{ m/s} \)). It will be shown here that there are significant changes to the lift curve slope across that Reynolds number range.

![Lift curves at various re](Source: [88])

The program Xfoil [147] is a two-dimensional panel code for aerofoil design that is capable of including viscosity into the solution by modelling the boundary layer. The program was used to estimate the effects of Reynolds number on the NACA 2416 aerofoil that is used on the test aircraft. Figure C.21 shows the lift curves for an inviscid case and three viscous solutions. One with a Reynolds number of 8 million and two with \( \text{Re}=350,000 \) and varying turbulence levels. The ‘Ncrit’ parameter of Xfoil is used to set
the turbulence level. A value of 9 represents normal wind tunnel turbulence levels, whereas Ncrit=4 represents a 'dirty' wind tunnel with high turbulence levels [147]. The wind tunnel used for this project is more likely in the last category, as shown in appendix A. The ‘clean’ tunnel case shows some non-linearities at higher angles of attack due to the boundary layer transition from laminar to turbulent flow. The ‘dirty’ wind tunnel is almost linear across the test range. There is a notable difference in slope for the three Reynolds numbers. The inviscid case has the steepest slope and the slope decreases with reducing Reynolds number. The step from inviscid flow to Re=8 million is almost of the same magnitude as going from Re=8 million to Re=350,000.

The results are tabulated in Table C.1, where the the lift curve slopes for a three-dimensional wing $C_{L_{\alpha,3d}}$ were estimated from the two-dimensional aerofoil $C_{L_{\alpha,2d}}$ using the lifting line theory

$$ C_{L_{\alpha,3d}} = C_{L_{\alpha,2d}} \left( 1 + \frac{C_{L_{\alpha,2d}}}{\pi e AR} \right) \tag{C.1} $$

where $e$ is the Oswald span efficiency obtainable from the drag polar and $AR$ the wing aspect ratio. Table C.1 shows that, within the limitations of this method, PanAir will over predict the lift curve slope by about 0.5/rad, which is about 10%. This is certainly a

Table C.1: Xfoil lift curve slopes for inviscid and low Reynolds number flow

<table>
<thead>
<tr>
<th></th>
<th>Aerofoil [1/rad]</th>
<th>Wing [1/rad]</th>
</tr>
</thead>
<tbody>
<tr>
<td>inviscid</td>
<td>7.13</td>
<td>4.64</td>
</tr>
<tr>
<td>Re=8,000,000</td>
<td>6.56</td>
<td>4.39</td>
</tr>
<tr>
<td>Re=350,000</td>
<td>6.08</td>
<td>4.17</td>
</tr>
</tbody>
</table>

Figure C.21: Xfoil solutions for inviscid and viscous aerofoil lift curve slopes.
significant effect. The results for the pitching moment are in good agreement therefore it was not necessary to apply any corrections.

For the lateral cases it is not possible to estimate the low Reynolds number effects in a simple fashion as the flow around the fuselage is much more complicated. For this, a known reference model or a good Navier-Stokes solution is required.

C.5.2 Lift Curve

The first benchmark is the lift curve slope $C_{L, \alpha}$. The results for the component build-up are compared separately, starting with the wing in isolation (W), then the wing-tail combination (WT), next the wing and body (WB) and finally the full aircraft with wing, fuselage and tailplane (WBT). For each case, a corresponding PanAir model was used. Table C.2 shows the results of the comparison of the PanAir results with and without the viscous correction against the wind tunnel data obtained at $V=20\text{m/s}$.

Table C.2: Lift curve slopes for the different configurations compared to the wind tunnel data

<table>
<thead>
<tr>
<th>$C_{L, \alpha}$ [1/rad]</th>
<th>W</th>
<th>WT</th>
<th>WB</th>
<th>WBT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PanAir (inviscid)</td>
<td>4.82</td>
<td>5.41</td>
<td>5.02</td>
<td>5.66</td>
</tr>
<tr>
<td>PanAir (viscous correction)</td>
<td>4.32</td>
<td>4.91</td>
<td>4.52</td>
<td>5.16</td>
</tr>
<tr>
<td>Wind tunnel</td>
<td>4.06</td>
<td>4.93</td>
<td>4.58</td>
<td>5.17</td>
</tr>
<tr>
<td>Difference %</td>
<td>6.4%</td>
<td>&lt;1%</td>
<td>1.3%</td>
<td>&lt;1%</td>
</tr>
</tbody>
</table>

Except for the isolated wing, the match of wind tunnel data and corrected PanAir results is excellent, with errors of 1.3% or less. This shows that the viscous correction is necessary and correct for this aircraft model and test conditions. The lift of the wing in isolation was affected by the exposed mount to the load cell, which is normally located inside the fuselage, hence leading to the large discrepancy.

C.5.3 Pitching Moment Curve

The second benchmark is the pitching moment measured about the loadcell attachment point. The same component build up is used as before. Table C.3 shows the results for these tests.

Table C.3: Pitching moment slopes for the different configurations compared to the wind tunnel data

<table>
<thead>
<tr>
<th>$C_{m, \alpha}$ [1/rad]</th>
<th>W</th>
<th>WT</th>
<th>WB</th>
<th>WBT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PanAir</td>
<td>-0.12</td>
<td>-1.43</td>
<td>0.055</td>
<td>-1.21</td>
</tr>
<tr>
<td>Wind tunnel</td>
<td>-0.16</td>
<td>-1.54</td>
<td>0.081</td>
<td>-1.19</td>
</tr>
<tr>
<td>Difference %</td>
<td>33%</td>
<td>7.6%</td>
<td>47%</td>
<td>1.6%</td>
</tr>
</tbody>
</table>
The match for the full aircraft (WBT) is very good. The other cases are not as close, with the largest errors for the tailless configurations. These might be explained by several factors: The pitching moments itself and the coefficient slopes in these cases are very small because the loadcell mount is close to the centre of pressure of the wing aerofoil. In the wind tunnel, this causes issues with loadcell accuracy and resolution and in PanAir there are potentially numerical errors due to the finite resolution of the chordwise pressure distribution caused by the wing panelling. The results for the wing-tail combination are probably also affected by flexing under load of the aluminium beam that replaced the fuselage. With 7.6% the error is still acceptable as a reasonable match.

Another notable result is the magnitude of the fuselage contribution to the total pitching moment. The fuselage is destabilizing and reduces the pitching moment slope by 22%. For an aircraft configuration like this, it is therefore important to include the pitching moment of the fuselage into all calculations. Simulating just the lifting surfaces, which is standard practice in simpler vortex lattice codes, will lead to large errors in the results for the static margin of the aircraft.

C.5.4 Lateral Data
The lateral derivatives for benchmarking are the sideforce and the rolling- and yawing moment due to sideslip. These were tested for the full configuration only, as the lateral cases are mainly affected by the flow over the wing-body combination together with the vertical fin. Table C.4 contains the results for these tests. The lateral results from PanAir are much more sensitive to modelling errors of the fuselage and the fin than the longitudinal cases. Great care was taken reproduce all dimensions correctly, but the model is not a perfect image of reality. The shape of the fin is also only an approximation, mainly due to the changes to the fuselage rear geometry and the shape of the ventral fin. Small changes to the fin size have large effects on $C_{n\beta}$ and $C_{y\beta}$. As mentioned before, no effects of the low Reynolds numbers could be estimated which may cause additional errors. Another source of error has been discussed in reference [150]. Due to the non-streamlined shape of the fuselage, if viewed from the front or back, flow separation will occur at very small angles of sideslip. Flow separation causes areas of low pressure next to the body, which in turn creates a sideforce that is not modelled in the inviscid PanAir simulation. The resulting differences between the lateral parameter measurements and simulations are larger than for the longitudinal cases. The side force derivative is most affected, most likely by the flow separation discussed above, while the other two have errors of 7% and 4% respectively. This is not a bad result given the limitations of the modelling and the wind tunnel noise levels, as discussed above.
Table C.4: Comparison of lateral results between experimental and PanAir results

<table>
<thead>
<tr>
<th>Panair</th>
<th>Wind tunnel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load cell CG</td>
<td>Flight/Gimbal CG</td>
</tr>
<tr>
<td>$C_{y,\beta}$</td>
<td>-0.43</td>
</tr>
<tr>
<td>$C_{l,\beta}$</td>
<td>-0.0704</td>
</tr>
<tr>
<td>$C_{n,\beta}$</td>
<td>0.0947</td>
</tr>
</tbody>
</table>

C.5.5 Summary

This concludes the verification of the virtual wind tunnel. The agreement between the test data and the simulations for the full Piper UAV airframe is excellent, which gives good confidence in the capabilities of the simulation. The method will now be used to generate a solution for the wall interference corrections of the wind tunnel test results.

C.6 Numerical Wind Tunnel Wall Corrections

The PanAir corrections are found by obtaining a solution with and without the surrounding tunnel walls and subtracting the results from each other. Only the longitudinal case was investigated, so a half model was used to save time. Similarly to the classical wall corrections, as discussed in appendix B, a correction for the angle of attack and the pitching moment due to the change of the tail angle of attack is expected. Since only the linear region and the coefficient slopes are of interest, the angle of attack correction can be expressed as a correction for the $C_{L,\alpha}$ slope and the pitching moment correction as an adjustment of the $C_{m,\alpha}$ slope.

A visualisation of the extensive wall interference with the aircraft’s flow field is shown in Figure C.22. The wing tip vortex impacting the wall extends back all the way to the end of the duct and the wall pressure changes near the model cover the entire test section. Measuring these wall pressure changes is another method of calculating the interference corrections in practise [136].

The computed wall corrections for the lift and moment curve slopes due to angle of attack changes at the wing and the tail are listed in Table C.5. It has been found that expressing the corrections as a percentage compared to absolute values works most accurately. These results can be directly applied to the wind tunnel derivatives. A transformation to the actual angle of attack offsets is not necessary as no absolute value is required for any single data point.

Table C.5: Virtual wind tunnel wall interference results

<table>
<thead>
<tr>
<th>PanAir Wall Correction</th>
<th>$C_{L,\alpha}$</th>
<th>$C_{m,\alpha}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-10%</td>
<td>-18%</td>
<td></td>
</tr>
</tbody>
</table>
In the following sections, the numerical results are compared with the classical corrections by applying them to the wind tunnel test data and comparing the result with the PanAir free air solutions of the aircraft component build up. Given the excellent agreement of the PanAir results with the wind tunnel test data, it has been assumed that the free air solution of PanAir will match free flight results of the Piper UAV equally well.

### C.6.1 Lift Curve Correction

Table C.6 lists the results for the lift curve slope with the wall corrections applied versus the numerical results. The correction for the lift curve slope is large because the UAV

<table>
<thead>
<tr>
<th>$C_{L_{\alpha}}$</th>
<th>WBT</th>
<th>WB</th>
<th>WT</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>PanAir free air (inviscid)</td>
<td>5.0</td>
<td>4.5</td>
<td>4.87</td>
<td>4.38</td>
</tr>
<tr>
<td>PanAir free air (viscous correction)</td>
<td>4.5</td>
<td>4.0</td>
<td>4.37</td>
<td>3.88</td>
</tr>
<tr>
<td>Wind tunnel w/ classic correction</td>
<td>4.76</td>
<td>4.26</td>
<td>4.57</td>
<td>3.8</td>
</tr>
<tr>
<td>Difference %</td>
<td>5.7%</td>
<td>6.5%</td>
<td>4.6%</td>
<td>2.1%</td>
</tr>
<tr>
<td>Wind tunnel w/ PanAir correction</td>
<td>4.54</td>
<td>4.12</td>
<td>4.44</td>
<td>3.78</td>
</tr>
<tr>
<td>Difference %</td>
<td>&lt; 1%</td>
<td>2.9%</td>
<td>1.6%</td>
<td>2.6%</td>
</tr>
</tbody>
</table>
Table C.7: Pitching moment slope wall interference corrections compared to PanAir free air solutions

<table>
<thead>
<tr>
<th></th>
<th>WBT</th>
<th>WT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PanAir free air</td>
<td>-0.99</td>
<td>-1.19</td>
</tr>
<tr>
<td>Wind tunnel w/ classic correction</td>
<td>-1.03</td>
<td>-1.36</td>
</tr>
<tr>
<td>Difference %</td>
<td>4.4%</td>
<td>14%</td>
</tr>
<tr>
<td>Wind tunnel w/ PanAir correction</td>
<td>-0.98</td>
<td>-1.26</td>
</tr>
<tr>
<td>Difference %</td>
<td>1%</td>
<td>5.8%</td>
</tr>
</tbody>
</table>

model wing span is almost 70% of the wind tunnel width. The PanAir data was calculated with the walls removed to obtain a free air solution. Then the viscous corrections were applied as before. Both corrections, the classical and the numerical are listed in the table to compare their agreement with the free air solution.

The classical correction for the lift curve is smaller than the numerical results. Inspecting the table, the PanAir correction is a better fit across all components with errors below 3% and better than 1% for the full aircraft. This is a very good result, showing that the more detailed modelling in the virtual wind tunnel gives a better accuracy than the traditional correction methods.

C.6.2 Pitching Moment Curve Correction

Table C.7 shows the results, again comparing the numerical corrections against the classic method. As before the PanAir corrections are larger than the classical results and fit the wind tunnel data better for the two tail-on configurations. The models without tail suffer from the same problems as in the pitching moment benchmarking in Table C.3 and have not been included in the table. The full aircraft configuration has a difference of only 1%, demonstrating again the accuracy of the PanAir solution for the wall corrections.

The calculated wind tunnel wall interference corrections from the virtual wind tunnel method agree very well with the experimental data. This is another successful benchmark for the method and gives good confidence in using these results to match the wind tunnel test data with the flight test data.

C.6.3 AVL Dynamic Derivatives

AVL is capable of estimating the dynamic derivatives for a configuration. It was used to investigate whether there is a need for correcting any of the dynamic derivatives obtained in Part 5 for comparison with the flight data. Correcting unsteady wind tunnel data is extremely complicated and AVL is the only accessible tool to attempt it. As shown in Fig C.8, the wind tunnel was modelled according to an example given in the AVL manual [143]. Only the lifting surfaces of the test aircraft were included in the model. As with PanAir, the corrections are simply the difference between a solution with and without the walls. Table C.8 lists the results for the most important dynamic parameters, the pitch damping, roll damping and yaw damping.
The data shows that there is negligible difference between the solutions within the limitations of the AVL solver. Therefore it seems to be not necessary to apply any corrections to the dynamic parameters obtained in the wind tunnel.

### C.7 Summary

This chapter introduced the virtual wind tunnel developed for this project. The PanAir solver was used to simulate the 7x5 wind tunnel environment to obtain data for the wall interference corrections and other data not easily obtainable from the wind tunnel tests. Considerable work has been put into the pre- and post processing tools for PanAir. The results is a very good match between the wind tunnel test data and the numerical simulation. This gives good confidence that the results obtained from the VWT are close to reality and can be used throughout the rest of the project.
D. AVL Dynamic Derivatives

For comparison of these experimental results with a numerical estimate of the dynamic derivatives, a simple model of the aircraft for the AVL code was created as shown in Figure D.1. Attempts in modelling the fuselage did not yield any usable results. The resulting dynamic derivatives are compared to the experimental results in Table D.1. The results for the roll mode parameter $C_{lp}$ and the yaw damping parameter $C_{nc}$ closely match the experimental results. For $C_{m_q}$ it is not clear if the estimate includes the $\dot{\alpha}$ component, similarly to the result for $C_{m_{\theta}}$. If so, this would also be a good match, but the manual is not clear on this issue. The cross derivative $C_{np}$ is not predicted very well, but this value is also very uncertain in the experimental results and is therefore not overly important for the dynamic response of this aircraft. The properties of the dynamic dutch roll- and the roll mode derivatives are estimated with good precision by AVL. This allows it to be used for preliminary design studies, which is the purpose this code was designed for. Together with the PanAir static results the properties of the primary modes of motion can be predicted quite successfully (at least for this standard configuration), which is the reason why both methods were integrated in the GEODUDE code introduced in Part 4.
Figure D.1: AVL model for estimation of the dynamic derivatives

Table D.1: AVL estimates against experimental data for selected dynamic derivatives

<table>
<thead>
<tr>
<th>Parameter</th>
<th>AVL</th>
<th>Experiment</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clp</td>
<td>-0.402</td>
<td>-0.395</td>
<td>1.7%</td>
</tr>
<tr>
<td>Clr</td>
<td>0.106</td>
<td>0.129</td>
<td>17.8%</td>
</tr>
<tr>
<td>Cmq</td>
<td>-11.21</td>
<td>-8.17</td>
<td>27.1%</td>
</tr>
<tr>
<td>Cnp</td>
<td>-0.016</td>
<td>-0.056</td>
<td>71%</td>
</tr>
<tr>
<td>Cnr</td>
<td>-0.062</td>
<td>-0.064</td>
<td>3.1%</td>
</tr>
</tbody>
</table>
An approximate method to determine $C_{m_\dot{\alpha}}$ has been presented in [151]. It is based on an example given in [39] for approximating the $C_{m_\dot{\alpha}}$ contribution of the tailplane, which causes most of the $C_{m_\alpha}$ effect.

Approximations for $C_{mq}$ and $C_{m_\dot{\alpha}}$ of the tailplane can be developed as

$$C_{mq} = -2a_t V_H \frac{l_t}{c} \quad \text{and} \quad C_{m_\dot{\alpha}} = -2a_t V_H \frac{l_t}{c} \frac{\delta \epsilon}{\delta \alpha_w}$$

(E.1)

It follows that

$$C_{m_\dot{\alpha}} = C_{mq} \frac{\delta \epsilon}{\delta \alpha_w}$$

(E.2)

where $C_{mq}$ has to be obtained from flight tests, dynamic wind tunnel experiments or numerical analysis. For this project, dynamic wind tunnel testing was used.

The above equations require the tail downwash factor $\frac{\delta \epsilon}{\delta \alpha}$ to be known. This factor can be estimated by running wind tunnel tests with different tail incidence angles and another run with the tail removed [88]. The moment curves for each run can then be plotted as shown in Figure E.1. The intersections of the moment curves of the tail-on runs with the tail-off curve give the wing angle of attack that results in zero tail lift ($\alpha_t = 0$ for a symmetric aerofoil).

The wing downwash at the tail $\epsilon_w$ can then be calculated from

$$\alpha_t = \alpha_w + i_t - \epsilon_w = 0$$

(E.3)

$$\Rightarrow \epsilon_w = \alpha_w + i_t$$

(E.4)

where $\alpha_w$ is the wing angle of attack and $i_t$ is the tail incidence. For multiple tail incidences, the downwash $\epsilon_w$ can be found with its corresponding wing angle of attack $\alpha_w$ and $\frac{\delta \epsilon}{\delta \alpha_w}$ can be calculated.
Figure E.1: Zero lift angle with several tail incidences (generated with PanAir)

For this project, these wind tunnel tests could not be performed because the tail cannot be rotated on the models without major rework and the tail-off moment curve is very shallow as the reference point is very close to the aerodynamic centre of the wing. The loadcell is not very accurate under these conditions. The same tests, however, can easily be done in the virtual wind tunnel, where it is simple to change the tail geometry of the model. Figure E.1 shows the results for five different tail angles and one tail-off run. Table E.1 list the resulting zero lift angles of attack and the wing downwash factor for each tail incidence angle. Plotting $\epsilon_w$ against $\alpha_w$ in Figure E.2 yields the result for $\frac{\delta \epsilon}{\delta \alpha}$.

<table>
<thead>
<tr>
<th>Tail incidence [deg]</th>
<th>0</th>
<th>1</th>
<th>2.25</th>
<th>3</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zero lift AoA [deg]</td>
<td>6.02</td>
<td>4.36</td>
<td>1.67</td>
<td>0.225</td>
<td>-3.48</td>
</tr>
<tr>
<td>$\epsilon_w$ [deg]</td>
<td>6.02</td>
<td>5.36</td>
<td>3.92</td>
<td>3.225</td>
<td>1.52</td>
</tr>
</tbody>
</table>

Figure E.2: Tail downwash $\epsilon_w$ estimation against angle of attack
\[ \frac{\delta \epsilon}{\delta \alpha_{\text{PanAir}}} = 0.481 \quad (E.5) \]

As a benchmark for this result, the USAF DATCOM method given in [39] gives for the test aircraft geometry

\[ \frac{\delta \epsilon}{\delta \alpha_{\text{DATCOM}}} = 0.432 \quad (E.6) \]

which is within 10\% of the virtual wind tunnel result. The DATCOM method is approximate and for full scale aircraft and does not include scaling effects due to low Reynolds numbers. Therefore it is not expected to be fully accurate for this case. Preliminary flight test results show that the PanAir results for the tail down wash factor are indeed of the correct magnitude.
F. Comment on Hobby Grade Equipment

An additional issue causing trouble during a flight test programme on this small scale is that there are virtually no professional grade components available, at least not for an affordable price. This results in all critical aircraft components being hobby grade parts with widely varying quality standards. From this author's experience paying more does not necessarily provide better quality in this sector. ¹

The main issues with using hobby parts for a flight test project are the lifespan of these products and the limited availability of correct specifications and data. This project, for example, went through five motors, with two inflight failures resulting in considerable damage to the aircraft. None of the batteries actually deliver the capacity that is printed on them, one is lucky if one gets about 80% out of a new one. This requires careful battery state monitoring and ground tests to avoid accidents. And finally, none of the parts have any way of monitoring their health or to perform maintenance other than a visual inspection. This leads to unpredictable failure of servo motors, motor speed controllers and other vital components. It is therefore necessary to plan crashes and accidents into the project plan. They will inevitably happen and cause potentially large delays in the project.

Another potentially catastrophic issue is the quality of the radio link from the pilot’s transmitter to the aircraft. With all the extra electronics installed in the plane and the powerful telemetry radio, it is quite possible to interfere with the control radio link, especially if a lower grade product is used. During this project this was the case until the radio gear was replaced with a top of the line product. Intermittent dropouts in the link can cause dangerous situations, and are hard to trace since most products do not have a method of monitoring signal strength. From this authors experience, it is not worth

¹This author was present at a talk, where the high loss rate of military UAS was attributed to the simple statement: 'They all use toy parts in a $100k UAS...'
saving money on this vital equipment. The cost of an accident or time lost due to an unreliable radio link easily outweighs the higher purchase price.