



Norwegian University of
Science and Technology

System Identification of Unmanned Aerial Vehicles

Thomas Ingebretsen

Master of Science in Physics and Mathematics

Submission date: Januar 2012

Supervisor: Ingve Simonsen, IFY

Co-supervisor: Thor Inge Fossen, ITK



MASTER THESIS FOR

THOMAS INGEBRETSEN

Master of Science in Technology, Applied Physics and Mathematics,
Study Direction Technical Physics

“System Identification of Unmanned Aerial Vehicles .”
“Systemidentifikasjon av Ubemannede Fly..”

This work has been carried out at Department of Physics, under the supervision of Professor Ingve Simonsen and Professor Thor I. Fossen

Trondheim, 22/01/2011

Ingve Simonsen

Professor at Department of Physics

Thor I. Fossen

Professor at Department of Engineering Cybernetics



MSC THESIS DESCRIPTION SHEET

Name: Thomas Ingebretsen
Department: Department of Physics, NTNU
Thesis title (Norwegian): Systemidentifikasjon av Ubemannede fly
Thesis title (English): System Identification of Unmanned Aerial Vehicles

Thesis Description: The purpose of the thesis is to investigate different methods for system identification, parameter estimation of unmanned aerial vehicles (UAVs). This includes literature study and computer simulations of an UAV system.

The following items should be considered:

1. Literature study on system identification. Give an overview of different methods found in the literature including recursive and off-line algorithms. Define the scope of the thesis and clarify what your contributions are.
2. Parameterize the UAV model and identify the parameters that need to be estimated. Show how parameter estimation techniques can be used to accurately calculate the model parameters using realistic measurements. Different maneuvers should be considered. A discussion regarding persistent excitation and parameter convergence should be included.
3. Conclude your results

Start date: 2011-08-22
Due date: 2012-01-22

Thesis performed at: Department of Physics, NTNU
Supervisor: Professor Ingve Simonsen, Dept. of Physics, NTNU
Co-Supervisor: Professor Thor I. Fossen, Dept. of Eng. Cybernetics, NTNU

Abstract

The least squares method has been applied to estimate parameters in an aerodynamic model of a simulated aircraft, using data that can be expected to be available from sensors on an Unmanned Aerial Vehicle. A combination of two non-linear state observers have been implemented to estimate wind data such as angle of attack, sideslip and dynamic pressure. Simulations have confirmed that the observers are able to estimate the wind data using noisy sensor measurements. Parameter estimation has been demonstrated with both measured and estimated wind data.

Sammendrag

Miste kvadraters metode har blitt brukt til å estimere parametere i en aerodynamisk model til et simulert fly, ved bruk av sensordata som kan forventes å finnes på et ubemannet fly. En kombinasjon av to ulinære observere har blitt implementert for å estimere vinddata som angrepsvinkel, sideslipp og dynamisk trykk. Simuleringer har bekreftet at estimering av vinddata kan gjøres ved bruk av støyete sensormålinger. Parameterestimering har blitt demonstrert ved bruk av både målte og estimerte vinddata.

Contents

Abstract	i
1. Introduction	3
2. Theory	7
2.1. Mathematical preliminaries	7
2.2. Coordinate Systems	8
2.2.1. Rotation Matrices	9
2.2.2. Quaternions	10
2.2.3. Inertial and body axes	11
2.2.4. Stability and wind axes	12
2.3. Rigid body equations of motions	13
2.4. Aircraft configuration	15
2.4.1. Physical properties	15
2.4.2. Inertia	16
2.5. Aircraft aerodynamics	17
2.6. Aircraft sensors	20
2.6.1. Global Navigation Satellite Systems	20
2.6.2. Rate gyro	21
2.6.3. Accelerometer	21
2.6.4. Magnetometer	22
2.6.5. Pitot-static tube	22
2.6.6. Wind vanes	23
2.7. Obtaining aerodynamical forces and moments	24
2.8. Vehicle state estimation	25
2.8.1. Attitude estimation	26
2.8.2. Velocity estimation	27
2.8.3. Dynamic pressure, alpha and beta estimates	28
2.9. System identification	29
2.9.1. Least squares estimator	29

2.9.2. Coefficient of Determination	31
2.9.3. Experiment design	32
2.10. Moving Polynomial Filter	32
3. Implementation	33
3.1. Simulator	33
3.2. Non-linear state observers	34
4. Results	35
4.1. Aircraft configuration	35
4.1.1. Inertia properties	35
4.2. Simulation results	36
4.2.1. Case 1 - LS with perfect data	37
4.2.2. Case 2 - LS with perfect instrumentation	39
4.2.3. Case 3 - LSE using noisy measurements	46
4.2.4. Case 4 - Non-linear observers	52
5. Discussion	65
5.1. Parameter estimation using least squares	65
5.2. Wind vanes	67
5.3. Coefficient of determination	67
5.4. Note on air data estimation	67
5.5. Input design	68
5.6. Inertia	68
5.7. Aerodynamical model	69
6. Conclusion	71
6.1. Future work	71
Appendices	77
A. Derivations	77
A.1. Physical pendulum	77
A.2. Explicit Complementary Filter	78
B. Matlab figures	79

C. Code implementation	81
C.1. Least Squares	81
C.2. Explicit Complementary Filter	81
C.3. Hua 2010	83
D. Digital attachment	85

List of Figures

2.1. Aircraft coordinate systems and configuration.	9
2.2. Overview of vehicle state estimation.	26
3.1. Simulator structure overview.	33
4.1. Estimated aerodynamic forces (a) and moments (b) during simulation in Case 1.	38
4.2. Input sequences are shown in (a). Wind angles α , β and rotation rates p , q and r are plotted in (b).	39
4.3. Angular accelerations obtained by numerical differentiation in Case 2.	43
4.4. Measured and corrected wind angles in Case 2.	44
4.5. Figure shows fitted, calculated and true aerodynamic forces in Case 2. Fitted curve was obtained using LSE with the calculated data as input.	45
4.6. Figure shows fitted, calculated and true aerodynamic moments in Case 2. Fitted curve was obtained using LSE with the calculated data as input.	46
4.7. Measured and true wind angles during simulation in Case 3.	49
4.8. Measured and true angular velocities during simulation in Case 3.	50
4.9. Model prediction and calculated aerodynamic forces in Case 3. Fitted curve was obtained using LSE with the calculated data as input.	51
4.10. Model prediction and calculated aerodynamic moments in Case 3. Fitted curve was obtained using LSE with the calculated data as input.	52
4.11. Estimated roll, pitch and yaw obtained by filtering sensor data compared to the true values.	54

List of Figures

4.12. Estimated bias with and without acceleration corrections. True bias values are all 0.	55
4.13. Estimated specific force in NED using proposed observer (Hua 2010)	56
4.14. Estimated NED velocity using proposed observer (Hua 2010).	57
4.15. Estimated velocity using proposed observers expressed in body coordinates.	58
4.16. Estimated wind angles calculated from estimated body ve- locities in Case 4	59
4.17. Estimated dynamic pressure \hat{q} calculated from estimated velocity in Case 4.	60
4.18. Estimated aerodynamic force coefficients in Case 4, using estimated α , β , and \bar{q} from filtering results.	62
4.19. Estimated aerodynamic moment coefficients in Case 4, us- ing estimated α , β , and \bar{q} from filtering results.	63
B.1. Overall simulator system.	79
B.2. Aircraft dynamics implemented in SIMULINK.	80
B.3. Rigid body kinetics.	80

List of Tables

2.1. Notation and brief description of common aircraft coordinate systems.	8
2.2. Some physical properties of an aircraft	16
2.4. Description of parameters included in the aerodynamical model used in simulations.	19
4.1. Aircraft and rig measured parameters.	35
4.2. Measured pendulum oscillation periods.	36
4.3. Measured distances from center of gravity to the axis of rotation.	36
4.4. Calculated moments of inertia.	36
4.5. Case studies using simulated aircraft dynamics.	37
4.6. Sensors used in Case 2.	40
4.7. Coefficients of determination of identified model in Case 2.	41
4.8. Estimated aerodynamic parameters in Case 2 using perfect measurements.	42
4.9. Instrumentation used in Case 3.	47
4.10. Coefficients of determination of identified model in Case 3.	47
4.11. Estimated aerodynamic parameters in Case 3 using noisy measurements.	48
4.12. Observer parameters used for filtering data in Case 4.	53
4.13. Instrumentation used in Case 4.	54
4.14. Estimated aerodynamic parameters in Case 4 using noisy measurements and estimated α , β and \bar{q}	61
4.15. Coefficients of determination of identified model in Case 4.	61
D.1. Digital attachment.	85
D.2. Files in digital attachment.	86

Abbreviations

ECF - Explicit Complementary Filter

EKF - Extended Kalman Filter

IMU - Internal Measurement Unit

GNSS - Global Navigation Satellite System

GPS - Global Positioning System

CFD - Computational Fluid Dynamics

PE - Persistent Excitation

COD - Coefficient Of Determination

MEMS - Micro Electro-Mechanical Systems

1. Introduction

System identification is the process of determining a mathematical description of a system based on empirical data from experiments. This includes input signals and measured system behaviour. In some cases it can be enough to build representations to mimic the output behaviour of the system in question, excluding the physical description of the underlying system. These methods are called non-parametric methods of system identification, and many such approaches are found in literature [6]. However, if more insight into the physics and details of the process is required, the identification problem is to identify parameters in a postulated model, linking the applied input to the observed output. Several methods exist [8] and in the context of aircraft system identification particularly the book *Aircraft System Identification: Theory and Practice*[9] gives a great introduction to the theory, with examples of application of different methods using real aircraft data.

Historically the least squares (LS) method has been applied to many technical problems [19] and is the method chosen to be demonstrated in this thesis. It may not be the best choice for identification of aircraft systems if highly accurate unbiased results are required, but the method is a good start for obtaining initial estimates [9]. Different extensions and modifications to the LS method have been developed over the years such as the weighted least squares (WLS) and instrumental variables method[19], but only ordinary least squares will be looked into.

One of the challenges of system identification is to obtain accurate measurements of the variables required for identifying the model in question. Ordinary equipment for data collection in commercial and military aircrafts is most probable expensive state of the art proprietary sensor systems, not available to be used in small unmanned aerial vehicles (UAVs) for student projects. However, recent development of microelectromechanical (MEMS) based inertial sensors [2] and non-linear observers for system state estimation[11][12] looks promising for obtaining reliable data

1. Introduction

for light weight airbourne vehicles at a lower cost. This approach will be investigated in this thesis.

As a part of an Unmanned Aerial Vehicle project, which involves several students, PhDs, Post Docs. and Professors at NTNU, this thesis will give a demonstration of system identification applied small UAVs with limited sensor systems. Using system identification provides a good alternative for developing mathematical descriptions of airborne systems without using expensive equipment such as wind tunnels and computational fluid dynamics (CFD). A good mathematical description is helpful for developing flight control systems and for tuning regulators. Such models can potentially reduce the need for experimental flying. Less flying leads to reduced risk, faster and more cost effective development. Being able to use a realistically simulated aircraft also allows the developers to experiment with ideas that might not be approved for testing using real aircrafts.

This thesis should be considered as an introduction to the field of aircraft system identification for small aircrafts using inexpensive instruments typically found in UAV flight control systems. The following list gives an overview of the chapters in this thesis:

- Chapter 2: The theory required for a rigid body simulation of an aircraft is introduced. Modelling of typical aircraft sensors with descriptions of their strenghts and weeknesses. A combination of two unlinear observers is proposed as an alternative to some of the complex sensors important for system identification. The LS method is explained.
- Chapter 3: An implementation of a rigid-body aircraft simulator using SIMULINK for experimenting with system identification of a known system. Implementation of unlinear filters for state estimation.
- Chapter 4: Simulation results demonstrating the use of LS for system identification and a verification of the possibility of using non-linear observers for replacing air relative physical sensors.
- Chapter 5: Discussion of key results and some important observations.
- Chapter 6: Conclusion and suggestions for future topics.

- Appendix A: Derivations
- Appendix B: Matlab figures of the simulator implementation.
- Appendix C: Code implementations.
- Appendix D: Digital Attachment.

2. Theory

2.1. Mathematical preliminaries

Skew symmetric

A matrix \mathbf{S} is said to be skew-symmetric if it has the property [1]

$$\mathbf{S} = -\mathbf{S}^T. \quad (2.1)$$

For any skew-symmetric matrix \mathbf{S} given as

$$\mathbf{S}(\Omega) = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix}, \quad (2.2)$$

the inverse operation is denoted $\text{vex}(\cdot)$ and is defined to be

$$\text{vex}(\mathbf{S}(\Omega)) := \Omega = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix}. \quad (2.3)$$

Linearly independent

Nonzero vectors \mathbf{x} and \mathbf{y} are linearly independent if and only if it does not exist any constant a so that $a\mathbf{x} = \mathbf{y}$. If such a constant exists, \mathbf{x} and \mathbf{y} are said to be linearly dependent [16].

Uncorrelated

Vectors \mathbf{x} and \mathbf{y} are said to be uncorrelated if the inner product $(\mathbf{x} - \bar{x}\mathbf{1})^T(\mathbf{y} - \bar{y}\mathbf{1})$ is equal to zero, where $\mathbf{1}$ is a vector with ones, and \bar{x} and \bar{y} are the means of elements in x and y respectively. Uncorrelated vectors are a special case of linearly independent vectors [16].

2. Theory

Orthogonal

Vectors \mathbf{x} and \mathbf{y} are orthogonal if and only if their inner product $\mathbf{x}^T \mathbf{y}$ is equal to zero. Orthogonal vectors are a special case of linearly independent vectors. Vectors can be both uncorrelated and orthogonal [16].

Persistent excitation

A signal u is considered to have the property of persistent excitation (PE) if there exists positive constants $\bar{\alpha}_0$ and T_0 so that the following condition is satisfied [7]:

$$\int_t^{t+T_0} u^2(t) dt \geq \bar{\alpha}_0 T_0 \quad (2.4)$$

$\bar{\alpha}_0$ is called the *level of excitation*.

2.2. Coordinate Systems

Four coordinate systems are convenient to introduce with respect to aircraft physics: body, stability, wind and an inertial reference coordinate system that is usually fixed to earth for local navigation [1]. Table 2.1 defines some notations for coordinate systems used in this thesis and Figure 2.1 gives a graphical representation of the axes and aircraft configuration.

Table 2.1.: Notation and brief description of common aircraft coordinate systems.

Coordinate system	Axes	Description
Inertial	X_I, Y_I, Z_I	Inertial non-accelerating coordinate system.
Body	X_b, Y_b, Z_b	Aircraft-fixed coordinate system.
Stability	X_s, Y_s, Z_s	Body system rotated around Y_b by the angle of attack to make Z_s perpendicular to the incoming wind.
Wind	X_w, Y_w, Z_w	Stability system rotated around Z_b by the side-slip angle making X_w to point into the incoming wind.

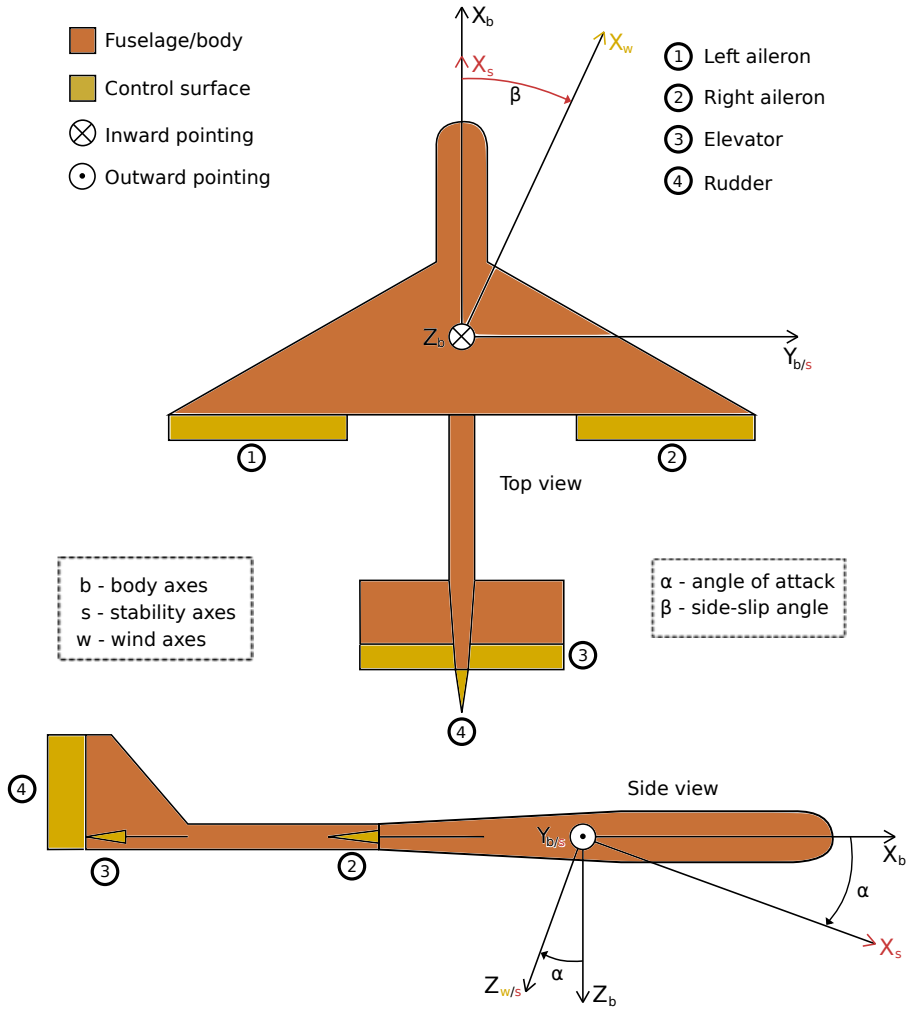


Figure 2.1.: Aircraft coordinate systems and configuration.

2.2.1. Rotation Matrices

Rotation matrices can be used to transform vectors from one reference frame to another. This is accomplished by the matrix-vector product [1]:

$$\mathbf{v}^b = \mathbf{R}_n^b \mathbf{v}^n. \quad (2.5)$$

The notation is such that the subscript denotes the source coordinate

2. Theory

system, while the superscript symbolizes the target coordinate system. Equation 2.5 is read: “ \mathbf{v}^b is equal to \mathbf{v}^n converted from n coordinates and into coordinates of b ”.

Rotation matrices are orthogonal and the inverse can be obtained by the transpose operation [1]:

$$(\mathbf{R}_n^b)^{-1} = (\mathbf{R}_n^b)^T = \mathbf{R}_b^n. \quad (2.6)$$

Inverse transformations using the relation from equation (2.6) can be written as:

$$\mathbf{v}^n = \mathbf{R}_b^n \mathbf{v}^b. \quad (2.7)$$

2.2.2. Quaternions

Quaternions can be used as an alternative representation of one coordinate system’s orientation with respect to another. Unlike the euler representation, quaternions do not have any singularities for particular orientations. A unit quaternion can be represented as a scalar and a three dimensional vector:

$$\mathbf{q} = \eta + \epsilon_1 \hat{i} + \epsilon_2 \hat{j} + \epsilon_3 \hat{k}, \quad (2.8)$$

with the constraint

$$\eta^2 + \epsilon_1^2 + \epsilon_2^2 + \epsilon_3^2 = 1. \quad (2.9)$$

Given an angular velocity $\boldsymbol{\omega}$ of the coordinate system with orientation represented by the unit quaternion \mathbf{q} , the rate of change of \mathbf{q} is given by the transformation

$$\dot{\mathbf{q}} = \frac{1}{2} \mathbf{T}_q(\mathbf{q}) \boldsymbol{\omega}, \quad (2.10)$$

where $\mathbf{T}_q(\mathbf{q})$ is the transformation matrix [1]

$$\mathbf{T}_q(\mathbf{q}) = \begin{bmatrix} -\epsilon_1 & -\epsilon_2 & -\epsilon_3 \\ \eta & -\epsilon_3 & \epsilon_2 \\ \epsilon_3 & \eta & -\epsilon_1 \\ -\epsilon_2 & \epsilon_1 & \eta \end{bmatrix}. \quad (2.11)$$

The rotation matrix can be found from the unit quaternion \mathbf{q} : [1]

$$\mathbf{R}_n^b(\mathbf{q}) = \begin{bmatrix} \eta^2 + \epsilon_1^2 - \epsilon_2^2 - \epsilon_3^2 & 2(\epsilon_1\epsilon_2 + \eta\epsilon_3) & 2(\epsilon_1\epsilon_3 - \eta\epsilon_2) \\ 2(\epsilon_1\epsilon_2 - \eta\epsilon_3) & \eta^2 - \epsilon_1^2 + \epsilon_2^2 - \epsilon_3^2 & 2(\epsilon_2\epsilon_3 + \eta\epsilon_1) \\ 2(\epsilon_1\epsilon_3 + \eta\epsilon_2) & 2(\epsilon_2\epsilon_3 - \eta\epsilon_1) & \eta^2 - \epsilon_1^2 - \epsilon_2^2 + \epsilon_3^2 \end{bmatrix}, \quad (2.12)$$

This is the transformation matrix that transforms vectors from the earth fixed coordinate system NED to vectors in the body coordinate system that is oriented according to NED with the unit quaternion \mathbf{q} .

2.2.3. Inertial and body axes

Body axes are aligned with the aircraft's principle axes. Origo is placed in the center of mass (CM) to eliminate any torque due to gravity, since gravity is assumed to be acting at CM. X_b is pointing towards the nose of the aircraft, Y_b along the right wing and Z_b is pointing straight down. This is illustrated in Figure 2.1.

The body reference frame is related to the inertial frame using the rotation matrix \mathbf{R}_n^b . Using euler angles this rotation matrix can be expressed as [1]:

$$\mathbf{R}_n^b(\phi, \theta, \psi) = \begin{bmatrix} c\psi c\theta & s\psi c\theta & -s\theta \\ -s\psi c\phi + c\psi s\theta s\phi & c\psi c\phi + s\phi s\theta s\psi & c\theta s\phi \\ s\psi s\phi + c\psi c\phi s\theta & -c\psi s\phi + s\theta s\psi c\phi & c\theta c\phi \end{bmatrix}, \quad (2.13)$$

where ϕ (roll), θ (pitch) and ψ (yaw) are the euler angles relating the body coordinate system to NED (North East Down). c is representing $\cos(\cdot)$ while s represents $\sin(\cdot)$.

2. Theory

The inertial frame should be a non-accelerated and non-rotating coordinate system, but for the purpose of this thesis a local earth surface reference system is used with its axes pointing north, east and down (NED). This is a good approximation of an inertial system for vehicles traveling at low velocities [1]. The NED reference system is denoted by using the letter n .

2.2.4. Stability and wind axes

The stability reference frame is obtained by rotating the body frame by the angle of attack α around the body y-axis Y_b [18]:

$$\mathbf{R}_b^s(\alpha) = \mathbf{R}_y(\alpha) = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix}. \quad (2.14)$$

The stability x-axis X_s is aligned with the relative wind vector so that the z-axis Z_s is perpendicular to the relative wind.

To obtain the wind reference frame the stability frame is rotated around Z_s by the side slip angle β [18]:

$$\mathbf{R}_s^w(\beta) = \mathbf{R}_z(\beta) = \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.15)$$

Now the wind x-axis X_w is parallel and opposite to the incoming wind vector. Then there is no component of the wind vector along Y_w and Z_w , such that the velocity expressed in wind coordinates becomes:

$$\mathbf{v}^w = \begin{bmatrix} V_T \\ 0 \\ 0 \end{bmatrix} \quad (2.16)$$

If the air mass is not moving relative to the earth, V_T will be equal to the body velocity $\|\mathbf{v}^b\|$.

The combined rotation from body-axis to wind-axis can be expressed as the matrix product of \mathbf{R}_b^s and \mathbf{R}_s^w :

$$\mathbf{R}_b^w(\alpha, \beta) = \mathbf{R}_s^w(\beta)\mathbf{R}_b^s(\alpha) = \begin{bmatrix} \cos \alpha \cos \beta & \sin \beta & \sin \alpha \cos \beta \\ -\cos \alpha \sin \beta & \cos \beta & -\sin \alpha \sin \beta \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix}. \quad (2.17)$$

Assuming no earth relative wind, the following three wind axis angles can be found using body velocity components $\mathbf{v}^b = [u \ v \ w]^T$ [18][9]:

Angle of attack:

$$\alpha = \tan^{-1} \left(\frac{w}{u} \right). \quad (2.18)$$

Sideslip angle:

$$\beta = \sin^{-1} \left(\frac{v}{V_T} \right). \quad (2.19)$$

Flank angle:

$$\beta_f = \tan^{-1} \left(\frac{v}{u} \right). \quad (2.20)$$

The sideslip angle is related to the flank angle by the relation [9]

$$\beta = \tan^{-1}(\tan \beta_f \cos \alpha) \quad (2.21)$$

that is obtained by inserting expressions for v and u using V_T , α and β , and then solving for β .

2.3. Rigid body equations of motions

The equations of motions are derived from Newton's second law for translation and rotation [9].

$$\mathbf{f}^b = \frac{d}{dt} (m\mathbf{v}^b) \quad (2.22)$$

$$\boldsymbol{\tau}^b = \frac{d}{dt} (\mathbf{I}\boldsymbol{\omega}) \quad (2.23)$$

$\mathbf{f}^b = [f_x \ f_y \ f_z]^T$ and $\boldsymbol{\tau}^b = [\tau_x \ \tau_y \ \tau_z]^T$ are the sum of all forces and torques acting on the rigid body respectively, expressed in body coordinates. \mathbf{v}^b is the body velocity and $\boldsymbol{\omega}$ is the angular velocity of the body, both expressed in body coordinates. \mathbf{I} is given as:

2. Theory

$$\mathbf{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix}, \quad (2.24)$$

where the inertia matrix elements are given by the volume integral [9]:

$$I_{ij} = \int_V ij dm. \quad (2.25)$$

Note that $I_{ij} = I_{ji}$, and for a rigid body that is symmetric with respect to the XZ-plane, like an aircraft, the cross terms I_{xy} and I_{yz} will be zero:

$$\mathbf{I} = \begin{bmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix}. \quad (2.26)$$

Assuming that the mass and inertia properties are constant, the translational and rotational equations of motion for the rigid body (2.27) and (2.28), can be found by differentiating the right hand side of equation (2.22) and (2.23) with respect to the body reference frame.

$$\mathbf{f}^b = m\dot{\mathbf{v}}^b + \boldsymbol{\omega} \times m\mathbf{v}^b \quad (2.27)$$

$$\boldsymbol{\tau}^b = \mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \quad (2.28)$$

The velocity of the body relative to the inertial frame is given by the rotated body velocity:

$$\dot{\mathbf{p}} = \mathbf{v}^n = \mathbf{R}_b^n \mathbf{v}^b \quad (2.29)$$

Combining equations (2.27), (2.28), (2.29) and the kinematics of quaternions (2.11) give the necessary differential equations for time-domain simulation:

$$\dot{\mathbf{p}} = \mathbf{R}_b^n \mathbf{v}^b \quad (2.30)$$

$$\dot{\mathbf{v}}^b = \frac{1}{m} (\mathbf{f}^b - \boldsymbol{\omega} \times m \mathbf{v}^b) \quad (2.31)$$

$$\dot{\mathbf{q}} = \mathbf{T}(\mathbf{q}) \boldsymbol{\omega} \quad (2.32)$$

$$\dot{\boldsymbol{\omega}} = \mathbf{I}^{-1} (\boldsymbol{\tau}^b - \boldsymbol{\omega} \times \mathbf{I} \boldsymbol{\omega}) \quad (2.33)$$

Expanding equation (2.31) and (2.33) gives:

$$\dot{u} = \frac{f_x}{m} - qw + vr \quad (2.34)$$

$$\dot{v} = \frac{f_y}{m} - rv + wp \quad (2.35)$$

$$\dot{w} = \frac{f_z}{m} - pv + uq, \quad (2.36)$$

and

$$I_{xx}\dot{p} - I_{xz}\dot{r} = \tau_x + rq(I_{zz} - I_{yy}) - I_{xz}pq \quad (2.37)$$

$$I_{yy}\dot{q} = \tau_y + pr(I_{xx} - I_{zz}) + I_{xz}(p^2 - r^2) \quad (2.38)$$

$$I_{zz}\dot{r} - I_{xz}\dot{p} = \tau_z + pq(I_{yy} - I_{xx}) + I_{xz}rq \quad (2.39)$$

2.4. Aircraft configuration

2.4.1. Physical properties

Physical properties used to describe the aerodynamical forces and moments are listed in Table 2.2 [18].

2. Theory

Table 2.2.: Some physical properties of an aircraft

Property	Symbol	Description
Wing span	b	Length of main wing, from wing tip to wing tip.
Wing surface area	S	Area of the main wing.
Mean aerodynamic chord (MAC)	\bar{c}	The chord line is the straight line between the leading edge and trailing edge of the wing. MAC is the mean length of this line along the wing span.
Mass	m	The total mass of the aircraft.
Inertia	$I_{xx}, I_{yy}, I_{zz}, I_{xz}$	Moments of inertia and product of inertia.

2.4.2. Inertia

Mass and inertia are two of the physical properties of the aircraft that need to be determined in order to find an aerodynamical description of the vehicle based on in-flight data recordings. The principle of a physical pendulum and the parallel axis theorem can be used to set up an experiment for determination of inertia properties. A physical pendulum assuming small angles of deflection has an oscillatory period approximated to be:

$$T \approx 2\pi \sqrt{\frac{I}{\hat{m}gR}}, \quad (2.40)$$

where R is the length between the pivoting point and the center of mass, and \hat{m} is the total pendulum mass, including the rig mass. A derivation of the above formula is found in A.1. This will hold for angles where

$\theta \approx \sin \theta$. The moment of inertia around the pivoting point can then be found by solving Equation (2.40) for the moment of inertia:

$$I_{pivot} = \hat{m}gR \left(\frac{T}{2\pi} \right)^2 - I_{rig}. \quad (2.41)$$

Here the moment of inertia due to the suspension rig I_{rig} is subtracted. Using the theorem ($I_R = I_{cm} + mR^2$)[4], the moment of inertia around the center of mass is expressed as:

$$I_{cm} = \hat{m}R \left(\frac{gT^2}{4\pi^2} - R \right) - I_{rig}. \quad (2.42)$$

Using the above equation requires measuring R and I_{rig} and a good experimental setup to obtain the period T . This can then be done for each of the axes to find all the inertia properties of the rigid body. In order to find the cross inertia term I_{xz} the above method is used to determine the moment of inertia of an intermediate axis in the XZ-plane (I_ξ), at an angle ξ to the x-axis. The following formula can then be used to find the product of inertia I_{xz} [9]:

$$I_{xz} = \frac{I_\xi - I_{xx} \cos^2 \xi - I_{zz} \sin^2 \xi}{\sin 2\xi} \quad (2.43)$$

2.5. Aircraft aerodynamics

Aerodynamic forces \mathbf{f}_A^b and moments $\boldsymbol{\tau}_A^b$ for a rigid body aircraft can be expressed using nondimensional coefficients [18]:

$$\mathbf{f}_A^n = \bar{q}S \begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} = \bar{q}S \mathbf{R}_w^b \begin{bmatrix} C_D \\ C_C \\ C_L \end{bmatrix} \quad (2.44)$$

$$\boldsymbol{\tau}_A^b = \bar{q}S \begin{bmatrix} bC_l \\ \bar{c}C_m \\ bC_n \end{bmatrix} \quad (2.45)$$

2. Theory

Where \bar{q} is the dynamic pressure given by the air pressure ρ and free airstream velocity V_T :

$$\bar{q} = \frac{1}{2}\rho V_T^2 \quad (2.46)$$

Aerodynamical coefficients (C_D , C_C , C_L , C_l , C_m and C_n) can be a function of several variables such as the control input, wind angles, angular velocity, dynamic pressure, Reynolds number, wing flexing, altitude, temperature and the time histories of all of them [18].

Assumptions are made to the aerodynamical model used in this thesis. Some of the assumptions are related to the purpose of using the model for system identification of small UAV's, while others are introduced to limit the complexity of going beyond what is needed for demonstration purposes in this thesis. The forces are given in the wind reference frame

It is assumed that

- ... the coefficients depend only on the instant value of the depending variables and not their time histories. This is the same as assuming steady flow [9].
- ... the aircraft simulated is considered to be a rigid body, neglecting effects from any flexing structures.
- ... variables such as dynamic pressure, Reynolds number, altitude and temperature are relatively constant during data collection and are therefore omitted from the model.
- ... the remaining variables are assumed to have a linear dependence on the coefficients. Polynomials of the depending variables can be used to model non-linear effects, but is not considered in this thesis.

The purpose of simulations is not to mimic exact aircraft dynamics, but to explore different methods for identification of aircraft-like models based on measurements most likely to be available in a real identification application. The following aerodynamical parameters is used unless otherwise specified in this thesis:

Table 2.4.: Description of parameters included in the aerodynamical model used in simulations.

Parameter	Description
C_{D_0}	Drag at zero angle of attack and sideslip.
C_{D_α}	Drag due to angle of attack. Angle of attack will increase the area facing into the wind and therefore increase the drag.
C_{D_β}	Drag due to sideslip. Sideslip will increase the aero facing into the wind and therefore increase the drag.
C_{C_β}	Cross-wind force due to sideslip. Sideslip will expose one side of the fuselage to the wind causing a net force in the Y_w direction.
C_{L_0}	Lift at zero angle of attack. This will be non-zero for asymmetric airfoils [18].
C_{L_α}	Lift due to increased angle of attack. The airfoil will generate greater lift as the angle of attack increases until the wing <i>stalls</i> [18].
C_{l_β}	Restoring roll moment due to sideslip. Mainly caused by the <i>dihedral angle</i> of the wing and the vertical stabilizer [18].
C_{l_p}	Damping from angular velocity. Mainly caused by the main wing when rolling.
$C_{l_{\delta_a}}$	Roll moment due to aileron control surface deflection.
C_{m_α}	Restoring pitch moment due to angle of attack. The tail horizontal surface will be the main contributor to this moment as it will hit the wind with an angle and thus generate the restoring moment.
C_{m_q}	Damping from angular velocity. Mainly caused by the horizontal stabilizer when pitching.
$C_{m_{\delta_e}}$	Pitch moment due to elevator surface deflection.
C_{n_β}	Restoring yaw moment due to side slip. Vertical stabilizer at the tail will generate an restoring moment because of the exposed area while side slipping.
C_{n_r}	Damping from angular velocity. Mainly caused by the vertical stabilizer when yawing.
$C_{n_{\delta_r}}$	Yaw moment due to rudder surface deflection.

2. Theory

Note that it is common to keep these coefficients non-dimensional for comparison between different aircrafts and between wind tunnel testing and parametric system identification. The scaling/non-dimensionalization is not explicitly shown here, but can be assumed to be included in the parameters themselves, since no such comparison will be given in this report. Equations (2.47)-(2.52) specify the aerodynamical coefficients used in simulations and for parameter estimation.

$$C_D = C_{D_0} + C_{D_\alpha}|\alpha| + C_{D_\beta}|\beta| \quad (2.47)$$

$$C_C = C_{C_\beta}\beta \quad (2.48)$$

$$C_L = C_{L_0} + C_{L_\alpha}\alpha \quad (2.49)$$

$$C_l = C_{l_0} + C_{l_\beta}\beta + C_{l_{\delta_a}}\delta_a + C_{l_p}p \quad (2.50)$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_{\delta_e}}\delta_e + C_{m_q}q \quad (2.51)$$

$$C_n = C_{n_0} + C_{n_\beta}\beta + C_{n_{\delta_r}}\delta_r + C_{n_r}r \quad (2.52)$$

Where α and β are the wind relative angles. δ_a , δ_e and δ_r are the control surface deflection expressed in radians of ailerons, elevator and rudder respectively. Control surfaces are shown in Figure 2.1. p , q and r are the angular velocities of the body with respect to the inertial frame of reference expressed in body coordinates. C_{l_0} , C_{m_0} and C_{n_0} are considered to be zero, assuming the aircraft is trimmed so that all moments are zero when controls are in the center position with α , β and angular velocities are zero.

2.6. Aircraft sensors

2.6.1. Global Navigation Satellite Systems

Global Navigation Satellite Systems (GNSS) [1] can be used to measure position and velocity in an earth fixed coordinate system. Several such systems exist today, although GPS (Global Positioning System) is the most used and established system [1]. In this thesis velocity measurements \mathbf{v}_E^n will be modelled as bias free and with added white noise $\boldsymbol{\nu}_v$:

$$\mathbf{v}_E^n = \dot{\mathbf{p}}^n + \boldsymbol{\nu}_v \quad (2.53)$$

2.6.2. Rate gyro

A rate gyro measures angular velocities of the body coordinate system relative to the inertial frame. Rate gyros are affected by environmental parameters such as temperature and input voltage [17]. Consequently it can not be assumed that the gyro measurements are bias free, and the bias can change with time, caused by a change in the environment. The measured angular velocity using a rate gyro can be expressed as [11]:

$$\boldsymbol{\omega}_E = \boldsymbol{\omega} + \mathbf{b}_\omega + \boldsymbol{\nu}_\omega, \quad (2.54)$$

where $\boldsymbol{\omega}_E$ and $\boldsymbol{\omega}$ are the actual measured and true angular velocities respectively, \mathbf{b}_ω is the time varying sensor bias, and $\boldsymbol{\nu}_\omega$ represents the measurement noise.

2.6.3. Accelerometer

Accelerometers give the immediate acceleration of the body reference system minus the gravitational component [18].

$$\mathbf{a}^b = \frac{1}{m} (\mathbf{f}^b - \mathbf{f}_g^b) \quad (2.55)$$

Assuming that the total forces acting on the aircraft \mathbf{f}^b is the aerodynamical forces \mathbf{f}_A^b from equation (2.44) and propulsion $\mathbf{f}_T^b = [T \ 0 \ 0]^T$, the accelerometer output can be expressed as follows [9]:

$$\mathbf{a}^b = \begin{bmatrix} a_x^b \\ a_y^b \\ a_z^b \end{bmatrix} = \frac{1}{m} \begin{bmatrix} \bar{q}SC_X + T \\ \bar{q}SC_Y \\ \bar{q}SC_Z \end{bmatrix} \quad (2.56)$$

The measured acceleration obtained by using an on-board accelerometer can be modelled by a time-varying bias \mathbf{b}_a and measurement noise $\boldsymbol{\nu}_a$ [1]:

$$\mathbf{a}_E = \mathbf{a}^b + \mathbf{b}_a + \boldsymbol{\nu}_a \quad (2.57)$$

2.6.4. Magnetometer

A magnetometer measures the magnetic field vector relative to the sensor orientation. Since the earth has its own magnetic field, the measurement can be used as vector measurement to relate the body coordinate system to an earth fixed coordinate system [11]. Let \mathbf{m}^n be the magnetic field vector given in the NED frame of reference, then an expression for the measured magnetic field given in body coordinates can be expressed as [11]:

$$\mathbf{m}_E = \mathbf{R}_n^b \mathbf{m}^n + \mathbf{B}_m + \boldsymbol{\nu}_m. \quad (2.58)$$

where \mathbf{B}_m represents the static magnetic deviation due to body fixed magnetic objects. $\boldsymbol{\nu}_m$ is the measurement noise. During local navigation it can be assumed that \mathbf{m}^n does not depend on the position \mathbf{p}^n . Just as with accelerometer vectors, magnetometer measurements do not provide any information about the rotation around the axis \mathbf{m}^n . Without an additional vector measurement, that is not parallel with \mathbf{m}^n , the total attitude will not be observable.

2.6.5. Pitot-static tube

A pitot-static tube measures the dynamic pressure caused by the relative wind. At the tip of the pitot tube the pressure will be the sum of the static and the dynamic pressure q . Subtracting the static pressure gives the dynamic pressure that can be calculated using Bernoulli's equation assuming the air is an incompressible fluid [13]:

$$\bar{q} = \frac{1}{2} \rho V_T^2 \quad (2.59)$$

$$(2.60)$$

Here ρ is the air density and is a function of altitude and temperature. Since the dynamic pressure is responsible for the magnitude of all of the aerodynamical forces and moments it is important to have good measurements of it for doing system identification. The measured dynamic

pressure is assumed to be bias-free and with added band limited white measurement noise:

$$\bar{q}_E = \bar{q} + \nu_{\bar{q}} \quad (2.61)$$

Great care should be taken when placing the pitot-static tube of the aircraft. It should be placed so that it will measure the undisturbed stream of air, unaffected by the fuselage and wing [13].

2.6.6. Wind vanes

Wind vanes can be used to measure the angle of attack α and flank angle β_f [9]. Since they cannot be placed at the center of gravity, the relative wind at the sensor location will be a function of both α , β , the angular rates p , q , r , and the sensor positions $\mathbf{r}_\alpha = [x_\alpha \ y_\alpha \ z_\alpha]$ and $\mathbf{r}_\beta = [x_\beta \ y_\beta \ z_\beta]$. \mathbf{r}_α and \mathbf{r}_β define the vectors from center of gravity to the location of the alpha and beta vanes respectively. Other errors due to local air disturbances should be avoided by correct placement of the vanes and wind tunnel calibration.

Given an arbitrary body position $\mathbf{r} = [x \ y \ z]$, relative to the center of gravity, the additional wind velocity due to rotation is provided by the cross product:

$$\mathbf{v}_r = \begin{bmatrix} u_r \\ v_r \\ w_r \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} + \boldsymbol{\omega} \times \mathbf{r} = \begin{bmatrix} u + qz - yr \\ v + rx - zp \\ w + py - xq \end{bmatrix} \quad (2.62)$$

Where \mathbf{v}_r is the total wind velocity at position \mathbf{r} . This is because the wind will be perpendicular to both the rotation axis and the displacement \mathbf{r} . Measured values obtained by the vane sensors can then be expressed using equations (2.18) and (2.19):

$$\tan \alpha_E = \frac{w + py_\alpha - x_\alpha q}{u + qz_\alpha - y_\alpha r} \quad (2.63)$$

$$\sin \beta_{fE} = \frac{v + rx_\beta - z_\beta p}{u + qz_\beta - y_\beta r} \quad (2.64)$$

Simplifications can be made by noting that $u_R \simeq u$, as the rotation contribution can be assumed to be small compared to the value of u :

2. Theory

$$\tan \alpha_E \simeq \frac{w + py_\alpha - x_\alpha q}{u} \quad (2.65)$$

$$\sin \beta_{fE} \simeq \frac{v + rx_\beta - z_\beta p}{u} \quad (2.66)$$

Further assumptions of small wind angles and rotation velocities give a formula for correcting the measured angles to obtain values for angle of attack and sideslip [9]:

$$\alpha_C = \alpha_E + \frac{qx_\alpha}{V_T} - \frac{py_\alpha}{V_T} \quad (2.67)$$

$$\beta_C = \beta_{fE} - \frac{rx_\beta}{V_T} + \frac{pz_\beta}{V_T} \quad (2.68)$$

2.7. Obtaining aerodynamical forces and moments

Since there is no sensor that can directly measure the aerodynamical forces and moments they need to be calculated from available sensor data. Starting with the equations (2.37)-(2.39) describing the rigid body rotational kinetics and inserting the aerodynamical moments from equation (2.45), solving for C_l , C_m and C_n gives:

$$C_l = \frac{1}{\bar{q}Sb} [I_{xx}\dot{p} - I_{xz}(pq + \dot{r}) + (I_{zz} - I_{yy})qr] \quad (2.69)$$

$$C_m = \frac{1}{\bar{q}S\bar{c}} [I_{yy}\dot{q} + (I_{xx} - I_{zz})pr + I_{xz}(p^2 - r^2)] \quad (2.70)$$

$$C_n = \frac{1}{\bar{q}Sb} [I_{zz}\dot{r} - I_{xz}(\dot{p} - qr) + (I_{yy} - I_{xx})pq] \quad (2.71)$$

The same procedure can be done with the accelerometer output equation (2.55), which together with the aerodynamical forces in equation (2.44) gives a fairly direct measure of C_X , C_Y and C_Z :

$$C_X = \frac{1}{\bar{q}S} (ma_x^b - T) \quad (2.72)$$

$$C_Y = \frac{1}{\bar{q}S} (ma_y^b) \quad (2.73)$$

$$C_Z = \frac{1}{\bar{q}S} (ma_z^b) \quad (2.74)$$

To obtain the same aerodynamical forces relative to the wind coordinate system a rotation shown in equation (2.44) is applied:

$$\begin{bmatrix} C_D \\ C_C \\ C_L \end{bmatrix} = R_b^w \begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} \quad (2.75)$$

2.8. Vehicle state estimation

To be able to estimate the necessary variables two observers [11][12] are put together. There will be given no proofs for stability by joining these two filters, apart from simulation results that show the performance under the specified conditions. Figure 2.2 gives an overview of their workings and how they are related. The three sections that follow give a more detailed description of the filtering process and how the estimates can be used.

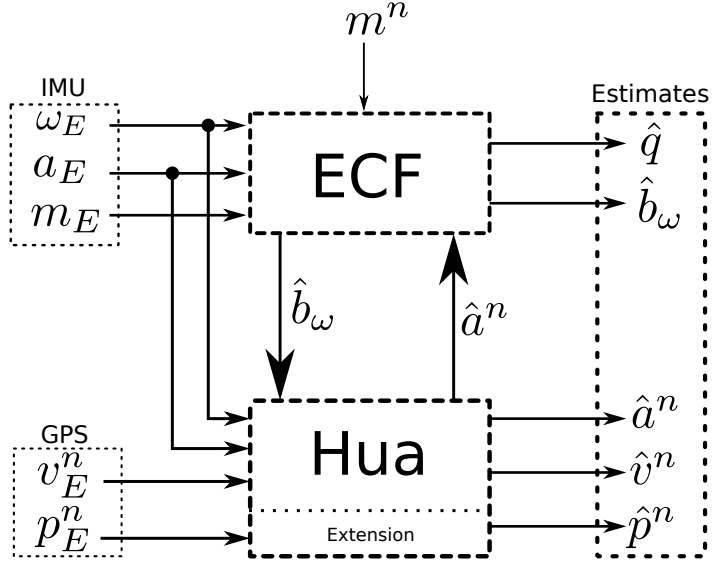


Figure 2.2.: Overview of vehicle state estimation.

2.8.1. Attitude estimation

The Explicit Complementary Filter (ECF) can be expressed as [11]

$$\boldsymbol{\omega}_{mes} = -\text{vex} \left\{ \frac{1}{2} \sum_{i=1}^n k_i (\mathbf{v}_i \hat{\mathbf{v}}_i^T - \hat{\mathbf{v}}_i \mathbf{v}_i^T) \right\} \quad (2.76)$$

$$\dot{\hat{\mathbf{q}}} = \mathbf{T}_q(\hat{\mathbf{q}}) (\boldsymbol{\omega}_E - \hat{\mathbf{b}}_\omega + k_P \boldsymbol{\omega}_{mes}) \quad (2.77)$$

$$\dot{\hat{\mathbf{b}}}_\omega = -k_I \boldsymbol{\omega}_{mes}, \quad (2.78)$$

where $\boldsymbol{\omega}_E$ is the angular velocity measured with respect to the body reference frame and \mathbf{T}_q is the transformation matrix from Equation (2.11). Measurements of the angular velocity will typically be done using a rate gyro and $\hat{\mathbf{b}}_\omega$ is the estimated gyro measurement bias. k_P and k_I are tuning parameters that are chosen to set a desirable convergence rate. k_i is a weighting factor for each of the n vector measurements. \mathbf{v}_i and $\hat{\mathbf{v}}_i$ are the measured and predicted vectors respectively. These vectors are arbitrary and any vectors that can be measured in NED and in the body reference frame can be used. Vectors available from an inertial measurement unit

are typically acceleration and magnetic field vectors. Equation (2.76), using only accelerometer (\mathbf{a}_E) and magnetometer (\mathbf{m}_E) vectors, can then be written:

$$\boldsymbol{\omega}_{mes} = -\frac{1}{2} \left[k_a \left(\frac{\mathbf{a}_E}{\|\mathbf{a}_E\|} \times \mathbf{R}_n^b(\hat{\mathbf{q}}) \hat{\mathbf{a}}^n \right) + k_m \left(\frac{\mathbf{m}_E}{\|\mathbf{m}_E\|} \times \mathbf{R}_n^b(\hat{\mathbf{q}}) \hat{\mathbf{m}}^n \right) \right], \quad (2.79)$$

where k_a and k_m are the filter tuning parameters, weighting the impact of the measurements respectively. These parameters can be used to dampen the effects of bad measurements if they are subject to high noise levels or are considered unreliable. See A.2 for the proof for how $\boldsymbol{\omega}_{mes}$ can be written using the cross product. Vector $\hat{\mathbf{a}}^n$ is an estimate of the current acceleration expressed in NED. \mathbf{m}^n is the local magnetic field vector expressed in NED obtained by measurements at site or using a lookup table. Note that all vectors are normalized to make the weighting parameters dimensionless and thus comparable. $\hat{\mathbf{q}}$ is the current attitude estimate expressed in quaternions. How to obtain the acceleration reference vector expressed in NED coordinates ($\hat{\mathbf{a}}^n$) is considered in the next section. If the sensor platform is suspended with no forces acting besides the normal force the reference vector will be reduced to $\hat{\mathbf{a}}^n = [0 \ 0 \ -g]^T$.

2.8.2. Velocity estimation

The following observer [12] is considered for estimation of the linear accelerations in NED coordinates. It requires measurements of $\boldsymbol{\omega}$, \mathbf{a}^b and NED velocity \mathbf{v}^n typically obtained by using a GPS receiver.

$$\dot{\hat{\mathbf{v}}}^n = k_1 (\mathbf{v}_E^n - \hat{\mathbf{v}}^n) + \mathbf{g}^n + \mathbf{Q} \mathbf{a}_E \quad (2.80)$$

$$\dot{\hat{\mathbf{Q}}} = \mathbf{Q} \mathbf{S} \left(\boldsymbol{\omega}_E - \hat{\mathbf{b}}_\omega \right) + k_v (\mathbf{v}_E^n - \hat{\mathbf{v}}^n) (\mathbf{a}_E)^T \quad (2.81)$$

k_1 and k_v are positive constant gains, which can be considered to be diagonal matrices if individual gains for the three axes is desirable. \mathbf{Q} is a 3×3 rotation-like matrix. Bias estimate $\hat{\mathbf{b}}_\omega$ obtained by ECF observer, is used to improve gyro measurements. \mathbf{v}_E^n is the actual measured velocity

2. Theory

from for example a GPS receiver. The Frobenius norm $\|\cdot\|_F$ is used to ensure the boundeness of \mathbf{Q} as suggested in [12].

$$\mathbf{Q} = \frac{\mathbf{Q}}{\|\mathbf{Q}\|_F} \sqrt{3} \quad (2.82)$$

An estimation of the acceleration in NED to be used as the reference vector ($\hat{\mathbf{a}}^n$) in Equation (2.79) is given by [12]:

$$\hat{\mathbf{a}}^n = \dot{\hat{\mathbf{v}}} - \mathbf{g}^n. \quad (2.83)$$

With perfect measurements and under the conditions that $\boldsymbol{\omega}$, $\dot{\mathbf{v}}^n$ and $\ddot{\mathbf{v}}^n$ are bounded signals, $\hat{\mathbf{v}}^n$ and $\mathbf{Q}\mathbf{a}^b$ will converge to \mathbf{v}^n and \mathbf{a}^n respectively [12]. If the system is under a constant non-zero acceleration the observer will be globally exponentially stable [12].

2.8.3. Dynamic pressure, alpha and beta estimates

The estimated body velocities

$$\hat{\mathbf{v}}^b = \begin{bmatrix} \hat{u} \\ \hat{v} \\ \hat{w} \end{bmatrix} = \mathbf{R}_n^b(\hat{\mathbf{q}})\hat{\mathbf{v}}^n \quad (2.84)$$

obtained by this observer can now be used to construct estimates for dynamic pressure \bar{q} , angle of attack α and side-slip angle β using equations (2.59), (2.18) and (2.19):

$$\hat{q} = \frac{1}{2}\rho(\hat{u}^2 + \hat{v}^2 + \hat{w}^2) \quad (2.85)$$

Note that the air density ρ is a function of altitude.

Estimate of angle of attack:

$$\hat{\alpha} = \tan^{-1}\left(\frac{\hat{w}}{\hat{u}}\right). \quad (2.86)$$

Estimate of side-slip angle:

$$\hat{\beta} = \sin^{-1}\left(\frac{\hat{v}}{\sqrt{\hat{u}^2 + \hat{v}^2 + \hat{w}^2}}\right). \quad (2.87)$$

It is important to note that these estimates assume that the local air-mass is not moving during the experiment. This means only very calm days should be considered for flying when using this method of estimating wind data.

2.9. System identification

2.9.1. Least squares estimator

The following derivation of the least squares estimator is from [10].

Given a parametric model on the form

$$y(t) = \boldsymbol{\theta}^{*T} \boldsymbol{\phi}(t) + v(t) \quad (2.88)$$

where $y(t)$ is the measured system output, $\boldsymbol{\phi}(t)$ is a collection of known signals and $\boldsymbol{\theta}^*$ is a vector of model constants. v represents the measurement error.

An estimate of the model parameters $\hat{\boldsymbol{\theta}}$ can be used to calculate an estimation error at any given time by subtracting the estimated output from the actual output:

$$e(t|\hat{\boldsymbol{\theta}}) = y(t) - \hat{\boldsymbol{\theta}}^T \boldsymbol{\phi}(t) \quad (2.89)$$

Given a set of N measurements of the output $y(t)$ and regressors $\boldsymbol{\phi}(t)$ the least squares estimate is the vector of model parameters $\hat{\boldsymbol{\theta}}_N^{LS}$ that minimized the sum of the squared estimation errors given in the equation above.

$$\hat{\boldsymbol{\theta}}_N^{LS} = \min_{\boldsymbol{\theta}} \left\{ \frac{1}{N} \sum_{t=1}^N [e(t|\boldsymbol{\theta})]^2 \right\} = \min_{\boldsymbol{\theta}} \left\{ \frac{1}{N} \sum_{t=1}^N [y(t) - \boldsymbol{\theta}^T \boldsymbol{\phi}(t)]^2 \right\} \quad (2.90)$$

Since the cost function is convex downward with respect to $\boldsymbol{\theta}$ the minimization problem can be solved by setting the derivative equal to zero:

2. Theory

$$\frac{d}{d\boldsymbol{\theta}} \frac{1}{N} \sum_{t=1}^N [y(t) - \boldsymbol{\theta}^T \boldsymbol{\phi}(t)]^2 = 0 \quad (2.91)$$

$$\frac{1}{N} \sum_{t=1}^N \boldsymbol{\phi}(t)y(t) = \frac{1}{N} \sum_{t=1}^N \boldsymbol{\phi}(t)\boldsymbol{\phi}^T(t)\hat{\boldsymbol{\theta}}_N^{LS} \quad (2.92)$$

The least squares estimate can then be solved analytically:

$$\hat{\boldsymbol{\theta}}_N^{LS} = \left[\sum_{t=0}^N \boldsymbol{\phi}(t)\boldsymbol{\phi}^T(t) \right]^{-1} \sum_{t=1}^N \boldsymbol{\phi}(t)y(t). \quad (2.93)$$

Using equation 2.93 in a recursive manner can be useful to be able to observe the convergence of parameters as more data is gradually included in the calculation.

Matrix notation

Using matrix notation with $\boldsymbol{\phi}$ as an n times N matrix where n is the number of regressors and N is the number of data points the least squares estimate can be written:

$$\hat{\boldsymbol{\theta}}_N^{LS} = (\boldsymbol{\phi}^T \boldsymbol{\phi})^{-1} \boldsymbol{y} \boldsymbol{\phi} \quad (2.94)$$

where \boldsymbol{y} is vector of length N containing the model outputs.

Properties

Parameter error due to the measurement error v can be found by inserting $y(t)$ and then subtracting the true parameter vector $\boldsymbol{\theta}^*$ [10]:

$$\begin{aligned} \tilde{\boldsymbol{\theta}}_N &= \hat{\boldsymbol{\theta}}_N^{LS} - \boldsymbol{\theta}^* \\ &= \left[\frac{1}{N} \sum_{t=0}^N \boldsymbol{\phi}(t)\boldsymbol{\phi}^T(t) \right]^{-1} \left[\frac{1}{N} \sum_{t=0}^N \boldsymbol{\phi}(t)\boldsymbol{\phi}^T(t)\boldsymbol{\theta}^* + \frac{1}{N} \sum_{t=0}^N \boldsymbol{\phi}(t)v(t) \right] - \boldsymbol{\theta}^* \\ &= \left[\frac{1}{N} \sum_{t=0}^N \boldsymbol{\phi}(t)\boldsymbol{\phi}^T(t) \right]^{-1} \left[\frac{1}{N} \sum_{t=0}^N \boldsymbol{\phi}(t)v(t) \right] \\ &= (\mathbf{R}_N)^{-1} \cdot \mathbf{f}_N \end{aligned} \quad (2.95)$$

If ϕ is considered as a collection of deterministic signals and the measurement noise $v(t)$ have a zero mean, the expectancy of the parameter error stated by the equation above can be shown to be zero [10]. However, as the regressors in ϕ are often measured system outputs, and thus stochastic variables, this is a bad assumption. Allowing stochastic quasi-stationary signals in ϕ , convergence of θ^{LS} to the correct value θ^* will happen as N approaches infinity if [10]:

- $v(t)$ is quasi-stationary and with zero mean.
- $\lim_{N \rightarrow \infty} \mathbf{R}^N$ is non-singular.
- Measurement error $v(t)$ is white or independent of any regressor in ϕ

Assuring that \mathbf{R} do not become singular is a problem related to the *persistent excitation* property of the regressors [18].

2.9.2. Coefficient of Determination

As a measure of how well the model is able to describe the measured data the coefficient of determination can be used. It is defined as the ratio of the regression sum of squares SS_R and the total sum of squares SS_T and can then be expressed as [9]:

$$R^2 = \frac{SS_R}{SS_T} = \frac{\sum_{t=1}^N (\hat{y}(t) - \bar{y})^2}{\sum_{t=1}^N (y(t) - \bar{y})^2} \quad (2.96)$$

where \bar{y} is the mean value of $y(t)$ defined as:

$$\bar{y} = \frac{1}{N} \sum_{t=1}^N y(t) \quad (2.97)$$

2.9.3. Experiment design

From any estimator point of view it is clear that it would be difficult to identify correct parameters if the regressors are correlated. For example if two regressors are linearly dependent there would be no information in the sample data to distinguish between their dependency on the observed output. Because of this it would be optimal if the system inputs are designed so that the regressors will be uncorrelated to each other. Even better would be to ensure that the regressors are orthogonal, decoupling the minimization problem in the least squares method [9]. System inputs appearing as regressors, that are subject to design, can be made orthogonal by selection. That is if there are no restrictions of the input signals that conflict with them being orthogonal. Such restrictions could for example be a desire to maintain stable flight during the experiment. To design an optimal input to make regressors linearly independent a system description would be needed beforehand and thus eliminating the need for conducting a system identification in the first place. Despite this any a priori information about the system could be used to help maximizing the orthogonality between model regressors. This will not be investigated any further in this thesis.

2.10. Moving Polynomial Filter

Differentiation can be performed by fitting a polynomial of degree k to a window of points and then evaluating the derivative of the polynomial [15]. The size of the window is defined by the number of points to the left and right of the center point. By moving the window over the entire data set it is then possible to find an approximation of the functions derivative. Interpolation can also be performed by evaluating the polynomial at an arbitrary distance from the center point. If the filter is applied to a series of equally spaced data values it is possible to express such a filter as a *finite impulse response* filter, using the least squares method [15]. To specify the parameters used the notation $\text{mpoly}(k, n_L, n_R)$ will be used, where k is the polynomial degree, while n_L and n_R are the number of points to the left and right of the center point. An implementation of the filter can be found in the digital attachment of this thesis.

3. Implementation

3.1. Simulator

Simulink was used to implement a rigid body aircraft simulator based on equations (2.31) and (2.28). It consists of four main components; input design, aircraft system, sensor emulation and logging, as shown in Figure 3.1.

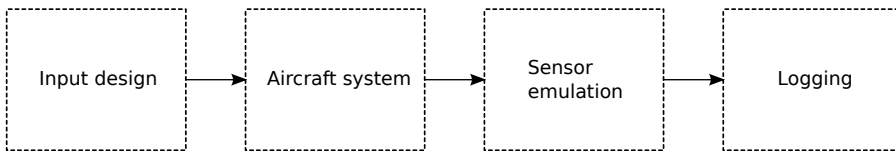


Figure 3.1.: Simulator structure overview.

The “Input design” component outputs the selected control input for the maneuver to be simulated. “Aircraft system” handles the rigid body simulation and calculates forces and moments based on the provided input and current aircraft state. The aircraft state is passed to “Sensor emulation” to be sampled as different on-board sensors would be able to, adding noise and biases to the true values. Sensor data is then given to the “Logging” component to be saved for post processing and system identification. Figure B.1 shows the implementation of the overall system in SIMULINK. Figure B.2 shows the content of the “Aircraft system” block that connects the aerodynamical forces (2.44)-(2.45) to the rigid body equations (2.34)-(2.39). The “6DOF Rigid body” block implements equations (2.30)-(2.32) and can be seen in Figure B.3.

3.2. Non-linear state observers

The implementation of ECF and Hua’s observer are divided into two and three MATLAB functions respectively. Filter state variables are kept in a structure and initialized with the function denoted with a `_init` ending. These functions set up the necessary initial filter states before the filtering process begins. The explanation of the function arguments are provided in the code listings found in C.2 and C.3. This implementation allows several instances of the filters to be used simultaneously.

ECF has only one function for updating the filter state. This function is called `ecf_update` and updates the filter according to equations (2.79), (2.77) and (2.78). This is convenient for using internal measurement units (IMU’s) that includes rate gyros, accelerometers and magnetometers in one package and providing the samples of these sensors at the same rate. Functions `e2q` and `q2rot` are converting euler angles to quaternions and quaternions to rotations matrices respectively. Both are provided in the digital attachment found in Appendix D. `q2rot` is an implementation of Equation (2.12), while function `Tq` is an implementation of Equation (2.11).

Hua’s filter is implemented with two functions to separate between updates of new IMU and GPS data, since they are not assumed to be sampled at the same rate. `hua_update` uses new GPS velocity measurements to correct $\hat{\mathbf{v}}^n$ and \mathbf{Q} estimates according to equations (2.80) and (2.81), but omitting $\mathbf{QS}(\boldsymbol{\omega}_E - \hat{\mathbf{b}}_\omega)$ and $\mathbf{g}^n + \mathbf{Q}\mathbf{a}_E$. The most recent measured acceleration is used as \mathbf{a}_E in this function.

For every sample of the IMU `hua_predict` is called to predict the filter states by using equations (2.80) and (2.81). Contributions involving \mathbf{v}_E^n are omitted from the equations as these are handled by the `hua_update` function as described above.

4. Results

4.1. Aircraft configuration

4.1.1. Inertia properties

The method for determining inertia properties in section 2.4.2 was applied to a model aircraft with physical properties listed in Table 4.1.

Table 4.1.: Aircraft and rig measured parameters.

Description	Symbol	Value
Aircraft mass	m	2.657 kg
Wing span	b	1.58 m
Mean chord	\bar{c}	0.22 m
: Wing area	S	0.348 m ²
Fuselage length	l	1 m
Suspension rig mass	m_{rig}	0.164 kg
Suspension rig moment of inertia	I_{rig}	0.000005 kgm ²
Total pendelum mass	$\hat{m} = m + m_{rig}$	2.821 kg

A total of 20 periods were measured for each of the three axes for determination of the pendulum oscillatory period. The results are listed in Table 4.2.

4. Results

Table 4.2.: Measured pendulum oscillation periods.

Experiment (#)	T_x (s)	T_y (s)	T_z (s)
1	1.455	1.425	1.675
2	1.455	1.420	1.690
3	1.460	1.420	1.690
4	1.460	1.425	1.675
Mean	1.458	1.423	1.683

Table 4.3.: Measured distances from center of gravity to the axis of rotation.

R_{yz} (m)	R_{xz} (m)	R_{yz} (m)
0.1456	0.1672	0.1898

Together with the lengths between center of gravity and the axis of rotation listed in Table 4.3 moments of inertia were found by using equation (2.42). The results are listed in Table 4.4.

Table 4.4.: Calculated moments of inertia.

I_x (kg · m ²)	I_y (kg · m ²)	I_z (kg · m ²)
0.157	0.158	0.275

These aircraft properties are used during simulations in this thesis.

4.2. Simulation results

Using the developed simulator different case studies have been performed to learn more about the validity and accuracy of parameter estimation using least squares. Table 4.5 show the situations that have been considered.

Table 4.5.: Case studies using simulated aircraft dynamics.

Case #	Description
1	Least squares method using perfect time series of aerodynamic coefficients. During simulation the coefficients are stored at every time step. This is the most perfect data time series that can be obtained.
2	Least squares method using perfect aircraft instrumentation data. Now the measurements are constrained to the instrumentation typically available for system identification of aircraft systems. Measurements are still perfect in the sense that no measurement noise or systematic errors are added.
3	Least squares method using noisy aircraft instrumentation data. This case is the same as Case 2 with measurement noise added to emulate a real aircraft sensory system.
4	Using non-linear observers to estimate \bar{q} , α , β , linear acceleration and gyro bias b , fixing gyro bias and eliminating the need for extra sensors.

4.2.1. Case 1 - LS with perfect data

The method is applied directly to the perfect data obtained from simulation. Time series of the aerodynamic coefficients in equations (2.47) to (2.52) are used directly together with the true values of the model regressors (α , β , δ_a , δ_e , δ_r , p , q and r). As expected this results in parameters that are identical to the ones that generated the data during simulation. Figure 4.1 shows the time series of both estimated and true aerodynamic forces and moments during simulation. Figure 4.2 shows input and output regressors.

4. Results

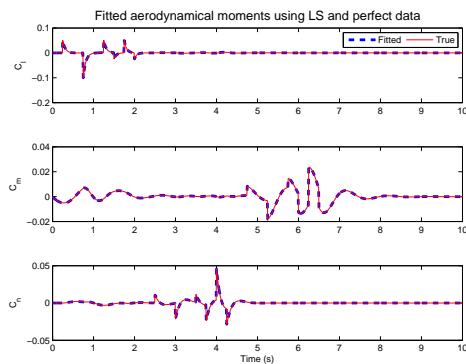
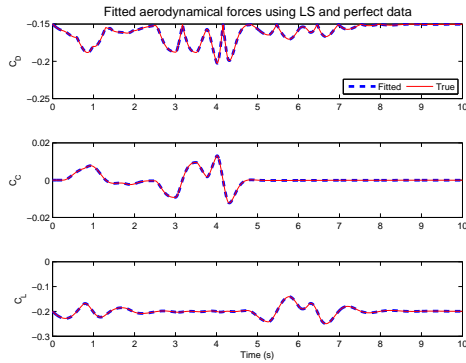


Figure 4.1.: Estimated aerodynamic forces (a) and moments (b) during simulation in Case 1.

This case is not very realistic and is provided only for demonstration of using the least squares method with aircraft data. In real world applications of parametric estimation of aircraft aerodynamics the challenge will be to obtain good measurements of the aerodynamical forces and having a good parametric model of the system. Some of the problems will be illustrated in the following cases.

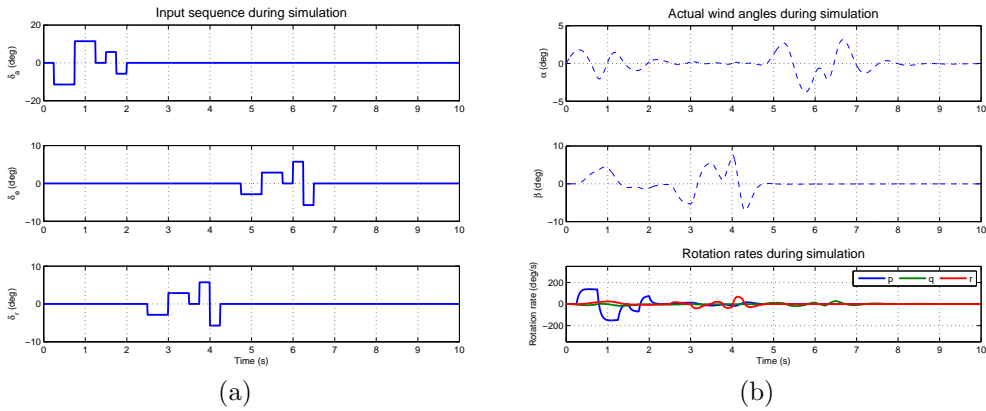


Figure 4.2.: Input sequences are shown in (a). Wind angles α , β and rotation rates p , q and r are plotted in (b).

4.2.2. Case 2 - LS with perfect instrumentation

In this case study the actual time series of the aerodynamic coefficients are not directly obtained from the simulator, but calculated based on typical aircraft instrumentation. The instruments are in this case assumed to provide noise and bias free perfect measurements. This does not mean that systematic errors are not included and simulations will demonstrate how the corrections affect the results. Table 4.6 shows aircraft sensors used.

4. Results

Table 4.6.: Sensors used in Case 2.

Sensor	Measured value	Description
Accelerometer	\mathbf{a}^b	Acceleration in body coordinates.
Rate gyro	$\boldsymbol{\omega}$	Angular velocity in body coordinates.
Alpha vane	α	Angle of attack sensor.
Beta vane	β_f	Measures the flank angle.
Pitot-static tube	\bar{q}	Measures dynamic pressure.
Aileron sensor	δ_a	Aileron deflection angle.
Elevator sensor	δ_e	Elevator deflection angle.
Rudder sensor	δ_r	Rudder deflection angle.

The accelerometer and pitot-static tube are used to calculate the aerodynamic forces C_X , C_Y and C_Z according to equations (2.72)-(2.74). An estimate of the thrust during the maneuver is also needed and in this case it was assumed to be constant and known. Wind angles α and β are calculated by using wind vanes. Corrections are applied to remove the errors due to angular velocities as described in 2.6.6. The alpha vane is placed two chord lengths outside the left wing tip, while the beta vane is placed two chord lengths in front of the leading edge of the left wing. Figure 4.4 show the results for α and β corrections during simulation, and it is clear that the corrections have improved the measurements.

There is no noise or bias introduced in any of the signals involved in the calculations. Thus it is expected that the calculated curves should be very close to the true ones when systematic errors are removed. Figure 4.5 shows the calculated and true aerodynamic forces, which overlap as expected, together with the identified model prediction.

Calculation of moments using sensor data is not as trivial since no sensors to measure the angular accelerations are provided. Angular accelerations are needed in equations (2.69)-(2.71) to be able to calculate the aerodynamic moment coefficients. This data can be obtained by differentiating the angular velocity measured by the rate gyros. This was done using the moving polynomial described in section 2.10, with $\text{mpoly}(1,1,1)$ for p and r , and $\text{mpoly}(1,5,5)$ for q . The results from this differentiation are given in Figure 4.3.

Table 4.7 gives the coefficients of determination (Eq. 2.96) describing how well the model is able to account for the variations in the data. It indicates a close fit of the model. However, by looking at Table 4.8 it is clear from the parameters identified that some errors were introduced by the necessary corrections.

Table 4.7.: Coefficients of determination of identified model in Case 2.

C_D	C_C	C_D	C_l	C_m	C_n
100.00%	100.00%	99.95%	92.63%	98.46%	95.35%

Data in this experiment comes from the same maneuver as in Case 1 and input regressors shown in Figure 4.2 therefore applies to this case as well. Figure 4.5 shows the true aerodynamic forces together with the results from calculation and predictions using the identified model.

4. Results

Table 4.8.: Estimated aerodynamic parameters in Case 2 using perfect measurements.

Parameter	True value	Estimated value	Error (%)
C_{D_0}	-0.150	-0.150	0.0%
C_{D_α}	-0.300	-0.301	-0.2%
C_{D_β}	-0.400	-0.400	-0.1%
C_{C_0}	0.000	0.000	0.0%
C_{C_β}	0.100	0.100	-0.0%
C_{L_0}	-0.200	-0.200	-0.0%
C_{L_α}	-0.900	-0.903	-0.3%
C_{l_0}	0.000	0.000	0.0%
C_{l_β}	-0.050	-0.045	11.0%
C_{l_p}	-0.020	-0.017	16.4%
$C_{l_{\delta_a}}$	-0.250	-0.213	14.7%
C_{m_0}	0.000	0.000	0.0%
C_{m_α}	-0.200	-0.198	1.1%
C_{m_q}	-0.010	-0.009	12.7%
$C_{m_{\delta_e}}$	-0.150	-0.144	4.3%
C_{n_0}	0.000	-0.000	0.0%
C_{n_β}	0.120	0.109	9.4%
C_{n_r}	-0.020	-0.017	13.6%
$C_{n_{\delta_r}}$	-0.210	-0.184	12.2%

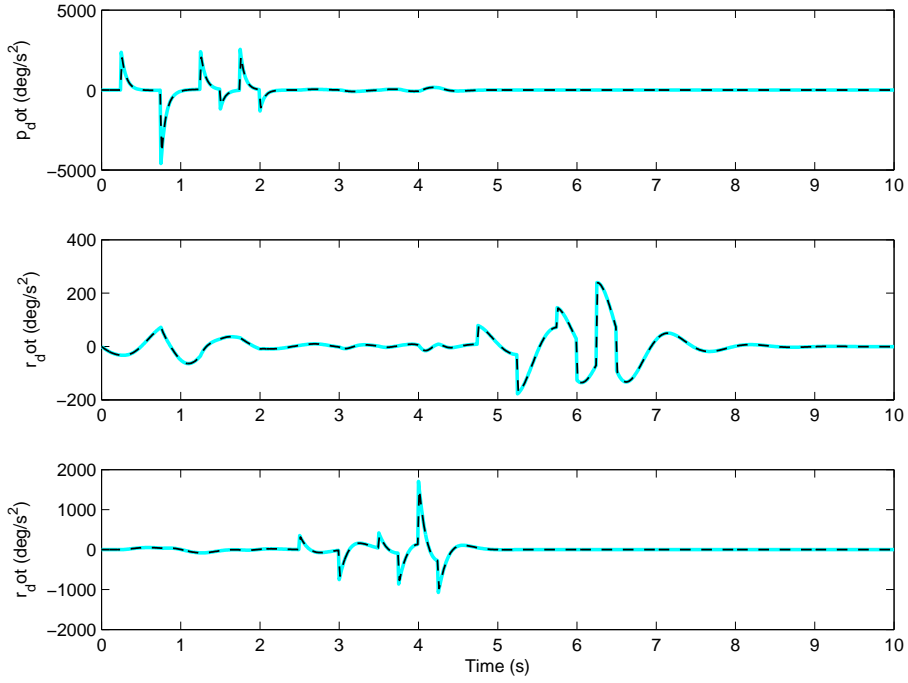


Figure 4.3.: Angular accelerations obtained by numerical differentiation in Case 2.

4. Results

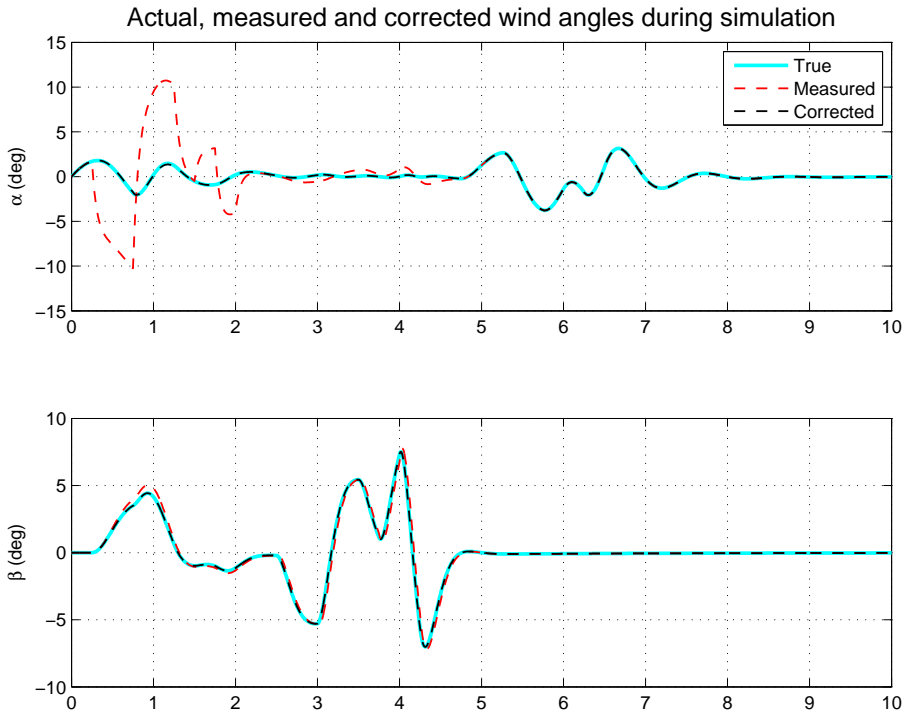


Figure 4.4.: Measured and corrected wind angles in Case 2.

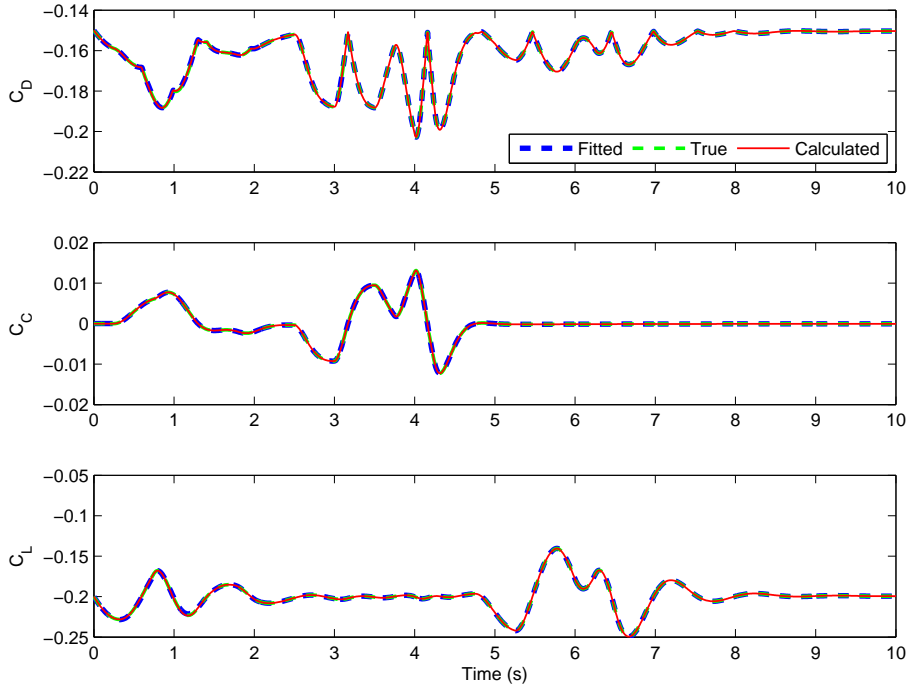


Figure 4.5.: Figure shows fitted, calculated and true aerodynamic forces in Case 2. Fitted curve was obtained using LSE with the calculated data as input.

4. Results

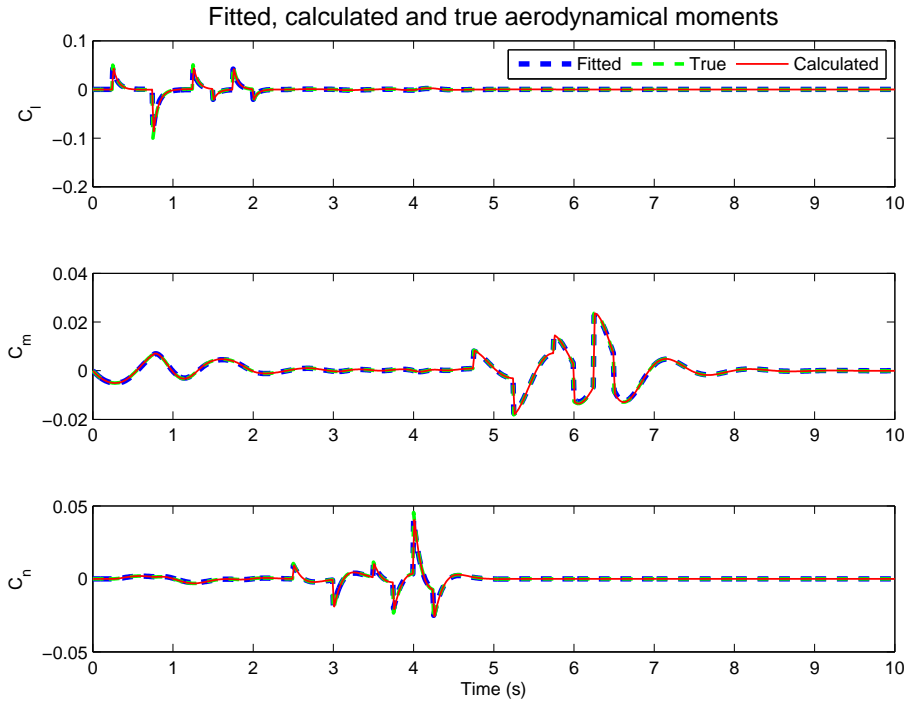


Figure 4.6.: Figure shows fitted, calculated and true aerodynamic moments in Case 2. Fitted curve was obtained using LSE with the calculated data as input.

4.2.3. Case 3 - LSE using noisy measurements

More realistic measurements of the aircraft sensors are used to see their impact on the identified parameters. The simulation setup is the same as in Case 2, but with added noise shown in Table 4.9. All sensors in Table 4.9 are sampled at 100Hz in simulations.

Sensor	Measured value	Noise density
Accelerometer	$\mathbf{a}^b + \boldsymbol{\nu}_a$	$3 \cdot 0.00981 \text{ ms}^{-1} \text{ Hz}^{-\frac{1}{2}}$
Rate gyro	$\boldsymbol{\omega} + \boldsymbol{\nu}_\omega$	$3 \cdot 0.5 \cdot \frac{\pi}{180} \frac{\text{rad}}{\text{s}} \text{ Hz}^{-\frac{1}{2}}$
Alpha vane	$\alpha + \nu_\alpha$	$0.05 \cdot \frac{\pi}{180} \frac{\text{rad}}{\text{s}} \text{ Hz}^{-\frac{1}{2}}$
Beta vane	$\beta_f + \nu_\beta$	$0.05 \cdot \frac{\pi}{180} \frac{\text{rad}}{\text{s}} \text{ Hz}^{-\frac{1}{2}}$
Pitot-static tube	$\bar{q} + \nu_{\bar{q}}$	$2.0 \text{ Pa Hz}^{-\frac{1}{2}}$
Aileron sensor	δ_a	0.0
Elevator sensor	δ_e	0.0
Rudder sensor	δ_r	0.0

Table 4.9.: Instrumentation used in Case 3.

Accelerometer and rate gyro noise densities are set to 3 times the densities given in the datasheet for Analog Devices ADIS16405 [5], which is a low cost internal measurement unit. Alpha and beta noise densities are set to roughly 20 times of the uncertainties given for a general aviation airplane in [9].

The estimated aerodynamic parameters from this simulation can be found in Table 4.11. It shows that estimates for linear force parameters are quite accurate, and that moment parameters have greater relative errors. Coefficients of determination calculated for the identified model can be seen in Table 4.10.

Table 4.10.: Coefficients of determination of identified model in Case 3.

C_D	C_C	C_D	C_l	C_m	C_n
49.22%	52.79%	47.85%	85.07%	74.13%	55.24%

4. Results

Table 4.11.: Estimated aerodynamic parameters in Case 3 using noisy measurements.

Parameter	True value	Estimated value	Error
C_{D_0}	-0.150	-0.149	0.9%
C_{D_α}	-0.300	-0.279	7.1%
C_{D_β}	-0.400	-0.411	-2.7%
C_{C_0}	0.000	0.000	0.0%
C_{C_β}	0.100	0.102	-2.4%
C_{L_0}	-0.200	-0.201	-0.5%
C_{L_α}	-0.900	-0.712	20.9%
C_{l_0}	0.000	-0.000	0.0%
C_{l_β}	-0.050	-0.044	12.9%
C_{l_p}	-0.020	-0.017	14.3%
$C_{l_{\delta_a}}$	-0.250	-0.219	12.6%
C_{m_0}	0.000	0.000	0.0%
C_{m_α}	-0.200	-0.164	18.1%
C_{m_q}	-0.010	-0.008	23.2%
$C_{m_{\delta_e}}$	-0.150	-0.138	7.8%
C_{n_0}	0.000	-0.000	0.0%
C_{n_β}	0.120	0.100	16.8%
C_{n_r}	-0.020	-0.016	19.7%
$C_{n_{\delta_r}}$	-0.210	-0.171	18.4%

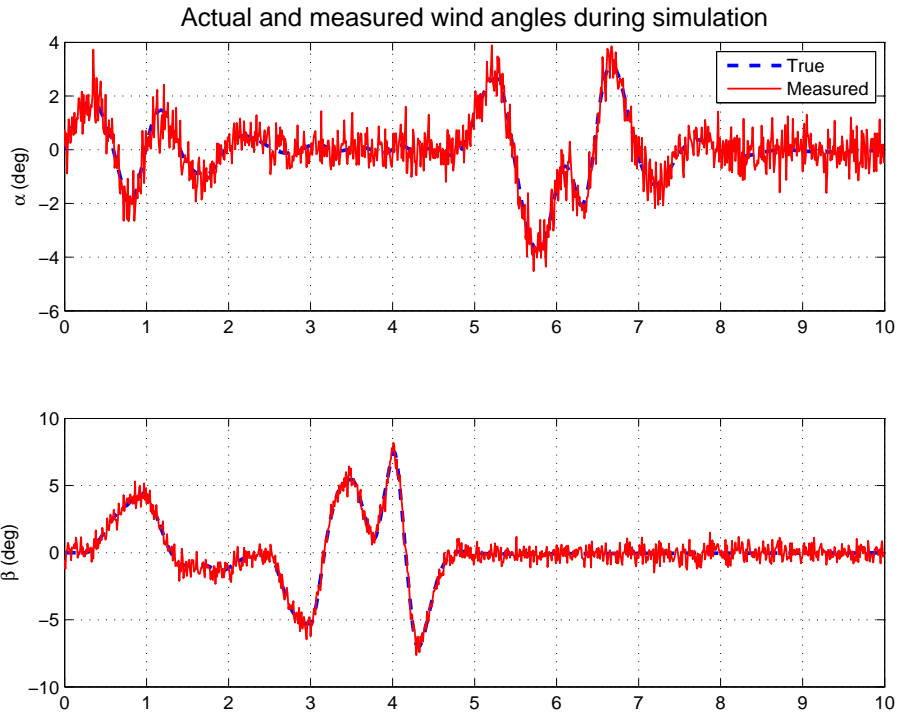


Figure 4.7.: Measured and true wind angles during simulation in Case 3.

4. Results

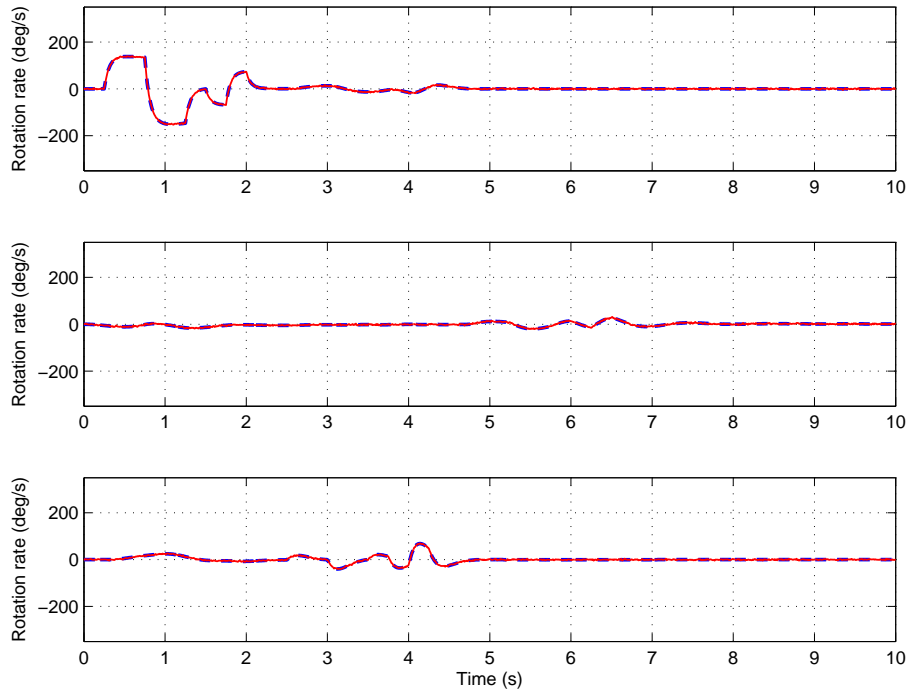


Figure 4.8.: Measured and true angular velocities during simulation in Case 3.

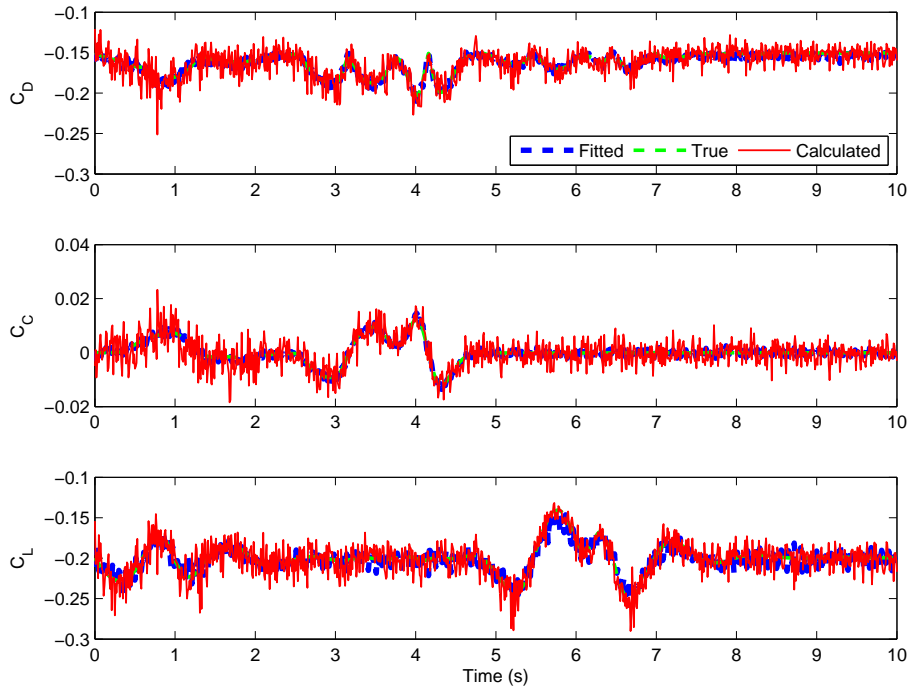


Figure 4.9.: Model prediction and calculated aerodynamic forces in Case 3. Fitted curve was obtained using LSE with the calculated data as input.

4. Results

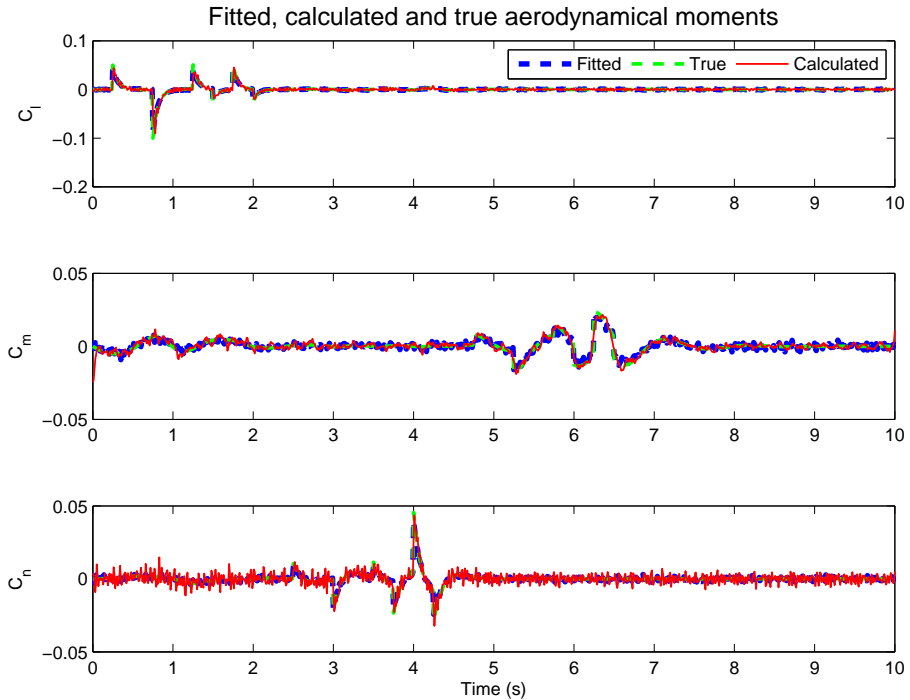


Figure 4.10.: Model prediction and calculated aerodynamic moments in Case 3. Fitted curve was obtained using LSE with the calculated data as input.

4.2.4. Case 4 - Non-linear observers

Applying the Explicit Complementary Filter expressed in equations (2.76)-(2.78) together with the observer from (Hua 2010) in equations (2.80) and (2.81) estimates of several important quantities can be found. Figure 2.2 gives an overview of the input/output variables and shows the interconnection between the two observers. This allows estimation of relative wind velocity and angles at the cost of adding a GNSS such as a GPS receiver. Local air mass has to be assumed to not be move during the experiment, requiring calm weather for data collection. GPS receivers are often already present in flight control equipment, and thus removing the vane sensors and the pitot-static tube could make the instrumental setup less complex and cheaper for doing system identification.

Table 4.13 shows the new setup used in Case 4. Vanes and the pitot-

static tube are removed and GPS velocity measurements are added. No other changes are made to the sensors from Case 3. Figure 4.11 shows the estimated euler angles using the proposed observer system. It also compares the results with attitude estimates using only ECF without adopting the specific force estimates from Hua's filter. A significant improvement is achieved with specific force corrections both to attitude and bias estimation, as seen in Figure 4.11 and 4.12. Note that the gyro biases are assumed to be constant during data collection and the bias observer is initially set to the correct value, simulating that bias estimates have converged prior to the start of the maneuver. Observer parameters used for generating the results presented in this section are given in Table 4.12.

Table 4.12.: Observer parameters used for filtering data in Case 4.

k_a	k_m	k_p	k_i	k_1	k_v
10.0	5.0	1.9	0.05	0.5	0.001

Velocity estimates obtained by using Hua's observer can be seen in Figure 4.14. The corresponding BODY velocity estimates are displayed in Figure 4.15 and are calculated by using the estimated attitude found by the filtering process. Estimated body velocities are then used to produce estimates for aircraft alignment to the relative wind given in Figure 4.16. Dynamic pressure calculated by the estimated total air relative speed can be seen in Figure 4.17.

4. Results

Table 4.13.: Instrumentation used in Case 4.

Sensor	Measured value	Noise density
Accelerometer x-axis	$a_x + \nu_a$	$5 \cdot 0.00981$
Accelerometer y-axis	$a_y + \nu_a$	$5 \cdot 0.00981$
Accelerometer z-axis	$a_z + \nu_a$	$5 \cdot 0.00981$
Gyro x-axis	$p + \nu_\omega$	$0.5 \cdot \frac{\pi}{180}$
Gyro y-axis	$q + \nu_\omega$	$0.5 \cdot \frac{\pi}{180}$
Gyro z-axis	$r + \nu_\omega$	$0.5 \cdot \frac{\pi}{180}$
Aileron sensor	δ_a	0.0
Elevator sensor	δ_e	0.0
Rudder sensor	δ_r	0.0
GPS velocity	$v^n + \nu_v^{gps}$	0.1

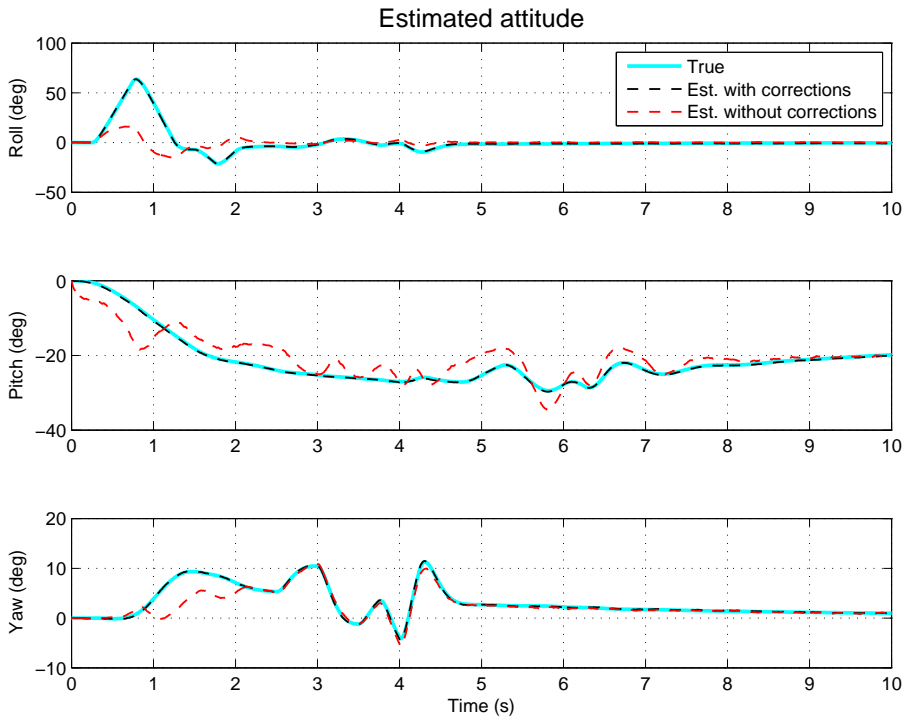


Figure 4.11.: Estimated roll, pitch and yaw obtained by filtering sensor data compared to the true values.

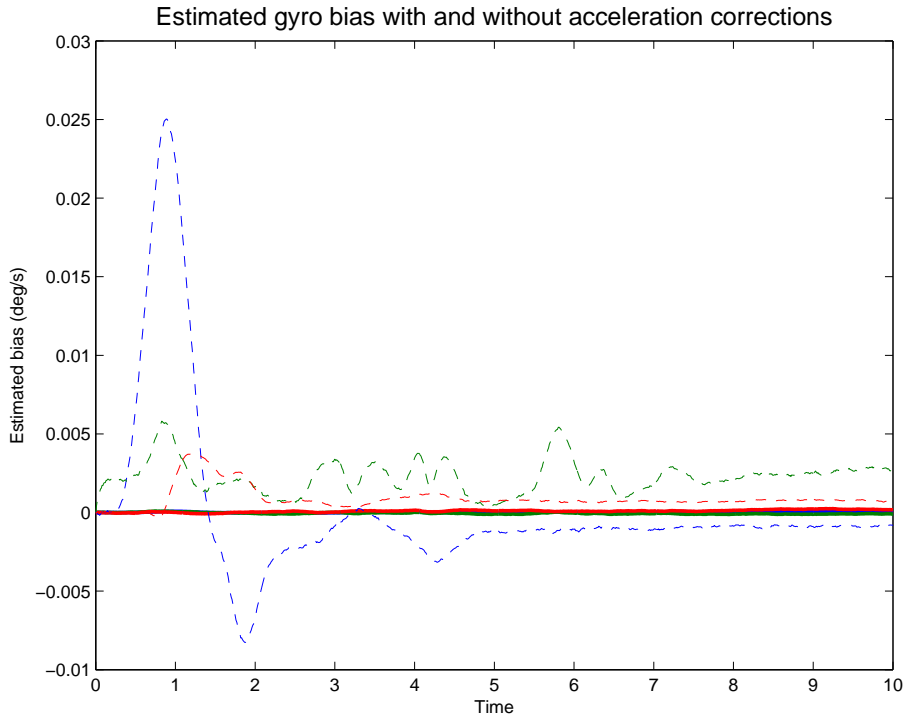


Figure 4.12.: Estimated bias with and without acceleration corrections. True bias values are all 0.

4. Results

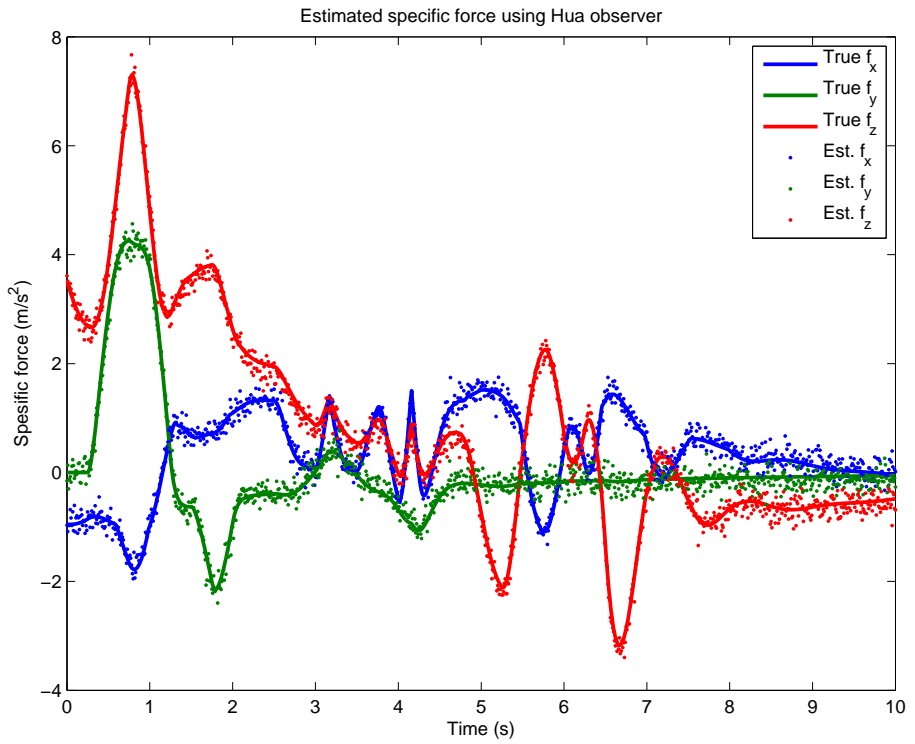


Figure 4.13.: Estimated specific force in NED using proposed observer (Hua 2010)

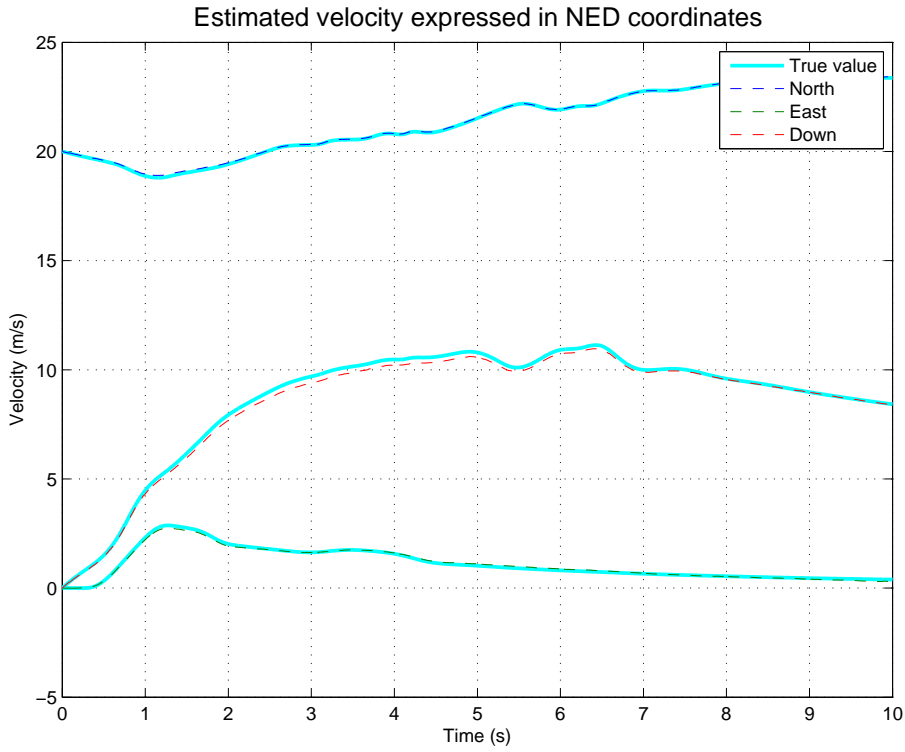


Figure 4.14.: Estimated NED velocity using proposed observer (Hua 2010).

4. Results

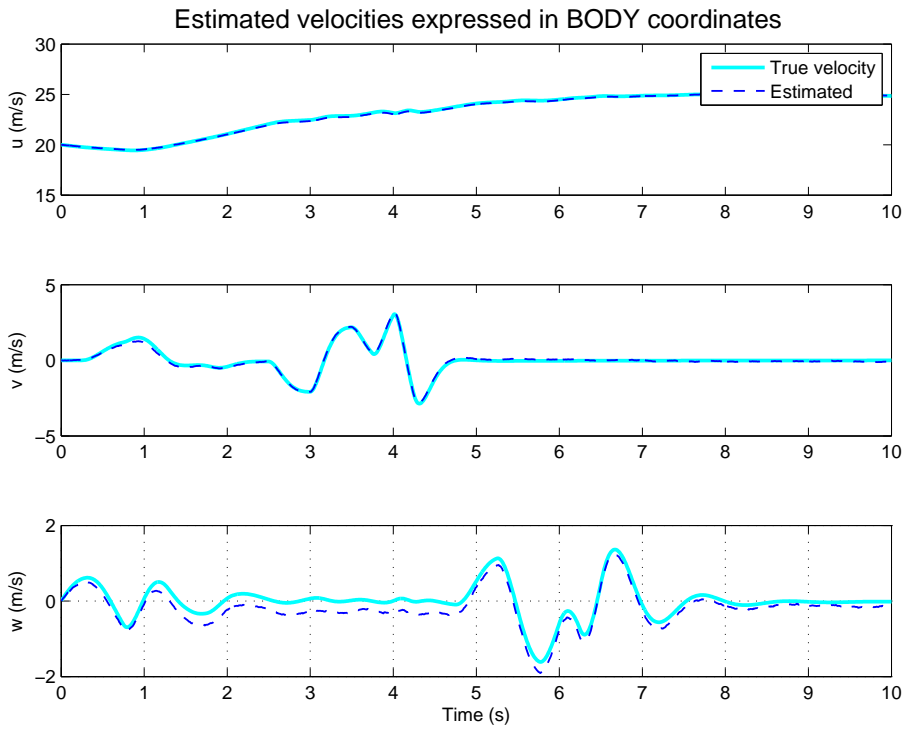


Figure 4.15.: Estimated velocity using proposed observers expressed in body coordinates.

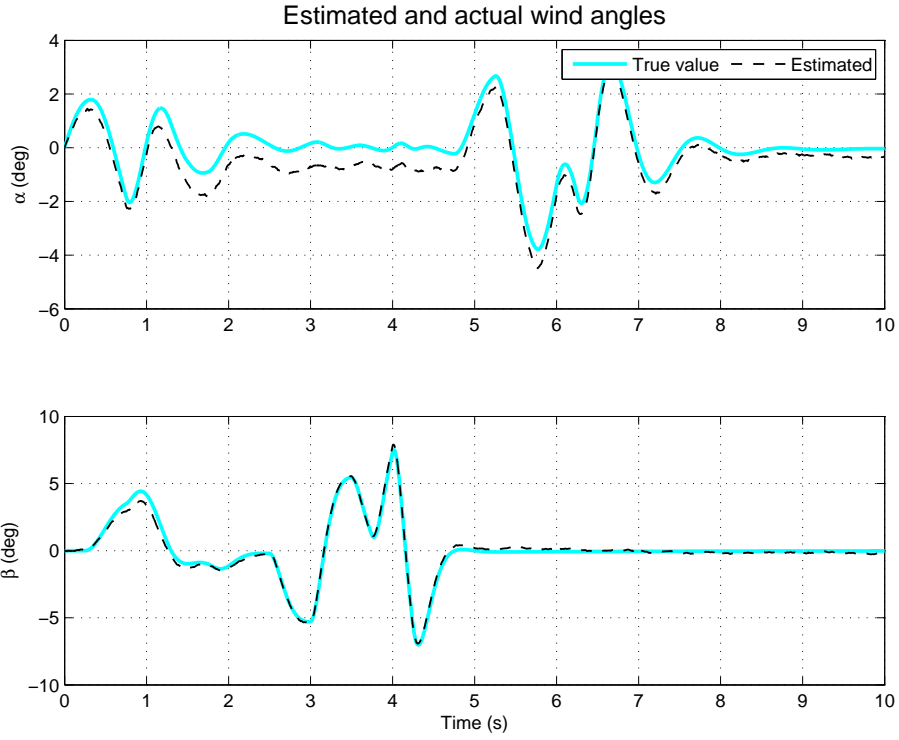


Figure 4.16.: Estimated wind angles calculated from estimated body velocities in Case 4

4. Results

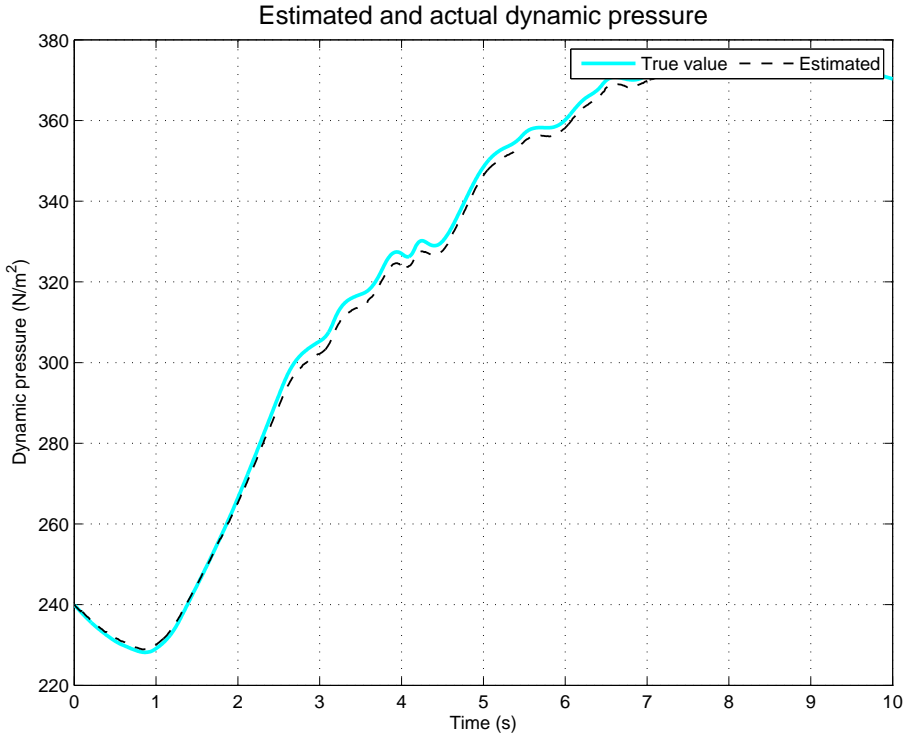


Figure 4.17.: Estimated dynamic pressure \hat{q} calculated from estimated velocity in Case 4.

Least squares estimates using the estimated α , β and \bar{q} instead of the measured ones as in Case 3 yields the results shown in Table 4.14 and 4.15. Model predictions for forces and moments can be seen in Figure 4.18 and figure 4.19 respectively.

Table 4.14.: Estimated aerodynamic parameters in Case 4 using noisy measurements and estimated α , β and \bar{q} .

Parameter	True value	Estimated value	Error
C_{D_0}	-0.150	-0.148	1.7%
C_{D_α}	-0.300	-0.289	3.7%
C_{D_β}	-0.400	-0.381	4.9%
C_{C_0}	0.000	-0.000	0.0%
C_{C_β}	0.100	0.096	4.0%
C_{L_0}	-0.200	-0.211	-5.4%
C_{L_α}	-0.900	-0.825	8.4%
C_{l_0}	0.000	-0.000	0.0%
C_{l_β}	-0.050	-0.044	12.1%
C_{l_p}	-0.020	-0.017	17.2%
$C_{l_{\delta_a}}$	-0.250	-0.213	15.0%
C_{m_0}	0.000	-0.002	0.0%
C_{m_α}	-0.200	-0.191	4.5%
C_{m_q}	-0.010	-0.006	35.7%
$C_{m_{\delta_e}}$	-0.150	-0.143	4.9%
C_{n_0}	0.000	0.000	0.0%
C_{n_β}	0.120	0.108	10.1%
C_{n_r}	-0.020	-0.016	22.0%
$C_{n_{\delta_r}}$	-0.210	-0.177	15.5%

Table 4.15.: Coefficients of determination of identified model in Case 4.

C_D	C_C	C_D	C_l	C_m	C_n
86.09%	47.43%	90.04%	88.06%	22.15%	71.15%

4. Results

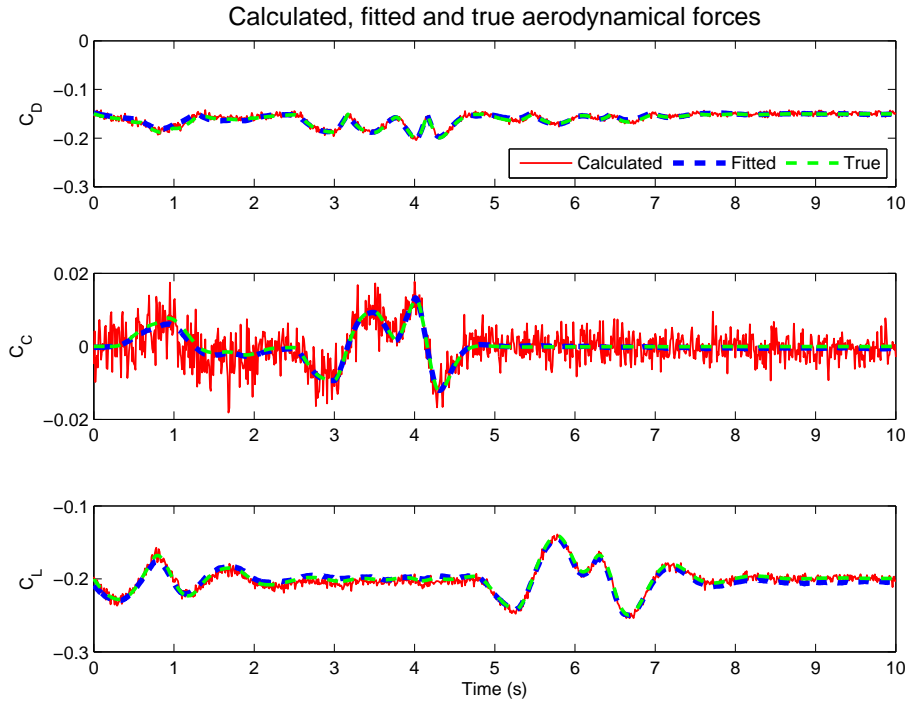


Figure 4.18.: Estimated aerodynamic force coefficients in Case 4, using estimated α , β , and \bar{q} from filtering results.

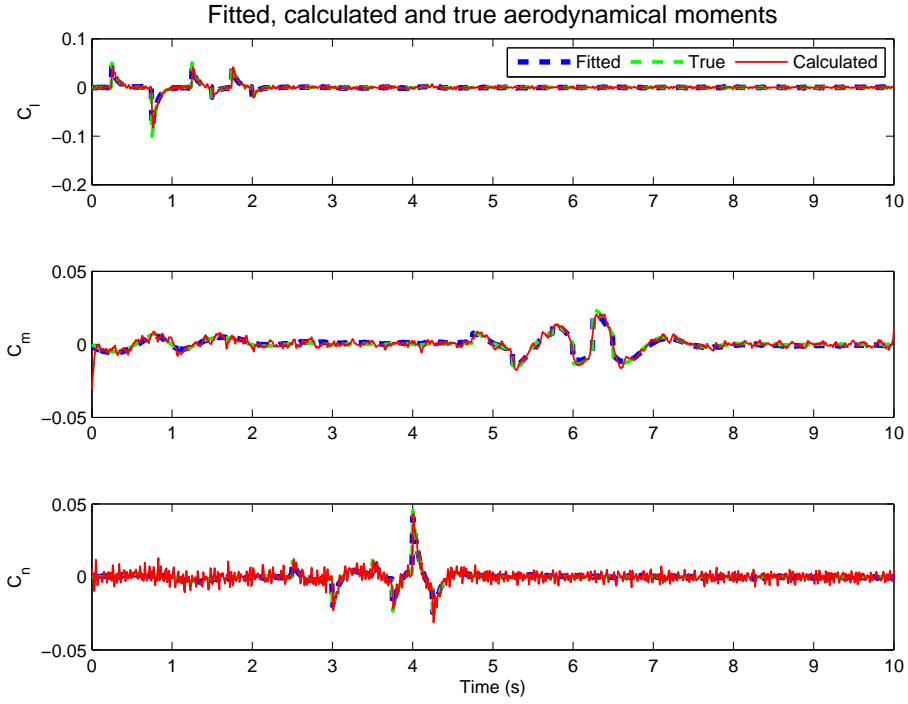


Figure 4.19.: Estimated aerodynamic moment coefficients in Case 4, using estimated α , β , and \bar{q} from filtering results.

5. Discussion

5.1. Parameter estimation using least squares

Case 1 demonstrates how the LS estimator is able to identify all the coefficients when the measurement error will be zero. From equation (2.95) it is clear that with $v = 0$ the parameter error is zero, given that R_N is invertible. R_N will be invertible as long as there is enough information in the gathered data to be able to distinguish between the dependency of the different parameters in the model. As described in 2.9 this means that non of the regressors can be linearly dependent, since it would result in a singular R_N . The intuitive explanation of this would be that if two regressors have the exact same variation it would be impossible to determine which of them are causing the variation in the measured data. This is therefore a problem that must be avoided when applying any method for system identification and not just in the case of the LS method. Having as accurate data as used in Case 1 is of course not very realistic based on the instrumentation considered in this thesis.

Using available sensors for the calculation of the aerodynamic coefficients is considered in Case 2. Even when perfect measurements of these sensors are assumed, some issues due to systematic errors occur. Angular accelerations are not measured directly and need to be calculated using numerical differentiation. As proposed in [9] a moving polynomial fitting filter is applied to estimate the first derivative of the angular velocities. Wind vanes also have systematic errors that are discussed in 5.2. The results from Case 2 look promising as reflected in the coefficients of determination in Table 4.7. Force coefficient parameters are estimated to very high precision, only limited by the errors in the wind angles due to the corrections applied to remove the systematic errors in the measurements. Moment parameters have larger errors caused by their dependency on the estimated angular acceleration, as seen in equations (2.69)-(2.71), caus-

5. Discussion

ing measurement errors and thus biases in parameter estimates as seen in Equation (2.95).

Case 3 considers sensor setup as in Case 2, but with added measurement noise. COD values listed in Table 4.10 seem not to be a good indication of how well the identified model reflects the measured data in this case, explained in more detail in 5.3. The increase in noise levels have increased the estimation error of the parameters.

In Case 4, estimates using ECF and Hua's non-linear filters are replacing measurements of α , β and \bar{q} . This eliminates the need for complicated instruments such as wind vanes and pitot-tube, subject to position errors if not calibrated correctly [13]. The instrumental setup is therefore becoming simpler, more compact and possible cheaper since the GPS receiver, which is needed to accomplish the estimation, is probably already present in the instrumental setup of the aircraft. Figure 4.11 shows how good the attitude estimation is, despite the noisy sensor readings and accelerated aircraft body. The figure also shows the plots estimating the attitude without the acceleration corrections when assuming the specific force is equal to $-\mathbf{g}$, which corresponds to a suspended aircraft only affected by the normal force. The estimated accelerations can be seen in Figure 4.13 and significantly improve estimated attitude during high accelerations. Gyro bias estimates are close to the correct value of zero during the experiment, as seen in Figure 4.12. It is clear that the acceleration estimates provided by Hua improves the estimation of the bias. Velocity estimates given in NED (Figure 4.14) are rotated by using the estimated attitude into BODY coordinates that can be viewed in Figure 4.15. It shows some errors in the z-component (w) and this is transferred into errors in the estimated dynamics pressure, angle of attack and sideslip angle seen in Figures 4.17 and 4.16. The errors might be the result of bad tuning, since correct tuning of these filters has not been looked into. Parameters used for the observers are listed in Table 4.12 and are found by trial and error until the results were reasonably good for demonstrating correct filter behaviour. Figures 4.18 and 4.19 show the model predictions identified using the estimated values obtained from filtering. Table 4.14 lists the estimated parameters and the coefficients of determination can be seen in Table 4.15.

5.2. Wind vanes

In Case 2 wind vane sensors were introduced to measure the aircraft alignment to the relative wind. While the alpha vane is able to directly measure the angle of attack, the beta vane is actually measuring the flank angle. This is in principle not a big issue since there exists an analytical relationship between flank and beta angle expressed in Equation (2.21), given that the angle of attack is known. However, since wind vanes cannot be positioned at the center of gravity additional relative velocities due to angular velocities will corrupt the measurements. Assuming small angles of attack, sideslip and angular velocities, corrections (Eq. (2.67) & (2.68)) for these systematic errors were tested. Corrections are undoubtedly effective under the conditions simulated as seen in Figure 4.4.

5.3. Coefficient of determination

Noisy measurement seems to make the coefficient of determination give a pessimistic measure of the fitted model when applied directly as seen in tables 4.10 and 4.15. The values are of course correct in the sense that they describe the proportion of the variation in the measured signal that is predicted by the model, but some of the variation is due to signal noise and not the deterministic variation that is the primary goal to predict. Despite this, bad COD values due to measurement noise will be an indication of low signal to noise ratio, and better instrumentation or bigger excitations of the signals in question should be considered to improve the results. This can be observed in Figure 4.9, 4.10, 4.18 and 4.19, which visually indicates, for the linear forces, that low signal to noise ratio will give lower COD values as seen in Table 4.10 and 4.15. Even when the fitting seems to be reasonably accurate and the parameter error is low compared to other parameters producing better COD values (see Table 4.11). This means that some care should be used when comparing COD values.

5.4. Note on air data estimation

Since the estimated velocity obtained by using the proposed observers is relative to the earth, it will not give the correct air relative velocities

5. Discussion

if the air is moving earth relative. Gust and wind affecting the aircraft while flying will thus not be included in these estimations, but errors are expected to be small when flying on calm days [9]. Despite this, using non-linear observers for estimating wind data can be preferable since no extra sensors need to be installed on the aircraft. Installing sensors like wind vanes is not trivial since the flow of air around the aircraft is disturbed by the aircraft itself, and this would lead to *position errors* in the sensors readings. The best placement of such sensors is usually found by using advanced CFD simulations, and wind tunnels are used for calibration [13].

5.5. Input design

There are several aspects concerning the correlation between measurement error v and regressors ϕ . If v is white noise with zero mean it will by definition be uncorrelated to ϕ and thus LS should give an unbiased identification of θ^* as N approaches infinity. While some of the regressors in ϕ are based on system outputs, others are system inputs and can be chosen. In system identification of aircrafts input regressors are typically surface deflections. Then if a closed loop feedback controller is used during data collection the inputs will be correlated with the system outputs, and thus resulting in a biased estimate. Using open loop control of the aircraft while collecting data is therefore preferable. This can either be done by predefining some input sequence to be executed or by flying manually. Correlation of output regressors with measurement noise is not as trivial to correct for, since they come from the same sensors.

5.6. Inertia

Inertia properties of a model aircraft were found by using the principle of a physical pendulum. Using low amplitude oscillations reduces the effects of the nonlinearity of the pendulum force, yielding consistent measurements of the oscillation period. Another method for finding inertia properties could be to record samples of pendulum angles during the experiment and applying the method of least squares for identifying the system. Then damping could be added as well, and greater amplitudes could be used.

5.7. Aerodynamical model

Several assumptions were introduced to the aerodynamical model used in this thesis, and when identifying the parameters it was assumed that the model was known exactly. This is an ideal situation that is not very realistic. Much is known about the dynamics of aircrafts, but to have an exact representation as in the simulated cases can not be assumed when using real aircraft data. When applying system identification to real aircraft data the choice of model will depend on the aircraft used and the content of the measured data. If the rudder was not used during data collection, the regression model used should not depend on this variable either. If only small excitations of the angle of attack are included in the data set, a linear term of angle of attack maybe enough to explain its dependency, and it will be preferable to skip the heigher order terms.

Different methods for determining if a regressor should be included or not could be used[14][3][9]. This is called stepwise regression. The idea is to define a stopping rule or a test for significance of the proposed regressor, to be able to decide if it should be included in the model or not. Stepwise regression can in this way be used to determine the model structure by testing several common aerodynamical dependencies found in litterature. An automatic approach to this can then be implemented to both find the best matching structure and identify the model parameters.

6. Conclusion

In this thesis a rigid-body simulator capable of simulating aircraft behaviour has been developed and used to investigate the least squares method of system identification. Simulations confirms several methods for correcting sensor data for systematic errors. Wind vanes can be corrected for local winds caused by angular velocity and non-linear filters can estimate biases in gyro measurements under the conditions simulated. Non-linear observers have also been proposed to replace wind vanes and pitot-static tube, and simulations indicate that they might be useful for estimating the angle of attack, sideslip angle and dynamic pressure, provided that there is no wind or gust during flying. This would simplify the instrumental setup and provide a faster and cheaper way of gathering data for system identification. UAV control systems might already have the necessary hardware and logging capabilities to provide a useful data set for model building. Ordinary least squares method for identification of aircraft dynamics is not ideal since the solution is expected to be biased due to nonlinearities even when assuming white sensor noise. Despite this, under the circumstances simulated biases are relative small and it can be argued that the results are useful. At least for initial estimation. Coefficient of determination has been applied to explain how well the identified model is able to account for the variations of the measured data. Some issues concerning the validity of these coefficients when used on noisy data have been discussed.

6.1. Future work

- Weighting of data measurement in LS algorithm based on confidence in sensor data. The LS estimation can be done without assigning equal weight to the data points. Data points known to be of less quality can thus be weighted less, possibly improving estimation re-

6. Conclusion

sults.

- Establish models for servo dynamics for surface deflection estimation without having to explicitly measure the deflection angle. Wind tunnel testing can be used to investigate the impact of aerodynamic forces acting on the surface while flying.
- Investigate different models for propeller thrust and engine dynamics. Develop good estimators for predicting thrust based on incoming air velocity, RPM and commanded power. Testing and verification of these models could be done using wind tunnels with the thrust system mounted on the aircraft as the aircraft shape could have an effect on the resulting thrust.
- Estimating aerodynamic forces without using any thrust during the maneuvers. Then the engine model can be identified in maneuvers with variable engine input.
- Identifying aerodynamics without using motor thrust. This removes the problem with thrust estimation. After the drag model is identified maneuvers with excited engine power at different flying speeds can be used to identify the propeller dynamics.
- Identifying both moments of inertia and angular damping by using time series of angle during pendulum experiments, by applying LS method for estimating parameters in the pendulum model.
- Stepwise regression for model building and determination of which regressor should be included, based on the information content in the sampled data. This could be automated by specifying a pool of regressors from aerodynamic literature that the algorithm can choose from.

Bibliography

- [1] T. I. Fossen. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons Ltd, 2011.
- [2] N. Barbour and G. Schmidt. Inertial sensor technology trends. *Sensors Journal, IEEE*, 1(4):332 –339, dec 2001.
- [3] Robert B. Bendel and A. A. Affi. Comparison of stopping rules in forward "stepwise" regression. *Journal of the American Statistical Association*, 72(357):pp. 46–53, 1977.
- [4] Bruce Denardo. Demonstration of the parallel-axis theorem. *The Physics Teacher*, 36(1):56–57, 1998.
- [5] Analog Devices. Tri-axis inertial sensor with magnetometer adis16405. Technical report, Analog Devices, Inc., 2009.
- [6] Peter E. and Wellstead. Non-parametric methods of system identification. *Automatica*, 17(1):55 – 69, 1981.
- [7] Petros A. Ioannou and Jing Sun. *Robust adaptive control*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1995.
- [8] P.N. James, P. Souter, and D.C. Dixon. A comparison of parameter estimation algorithms for discrete systems. *Chemical Engineering Science*, 29(2):539 – 547, 1974.
- [9] V. Klein and E.A. Morelli. *Aircraft system identification: theory and practice*. AIAA education series. American Institute of Aeronautics and Astronautics, 2006.
- [10] Lennart Ljung. *System Identification: Theory for the User (2nd Edition)*. Prentice Hall, 2 edition, 1999.

Bibliography

- [11] R. Mahony, T. Hamel, and J.-M. Pflimlin. Nonlinear complementary filters on the special orthogonal group. *Automatic Control, IEEE Transactions on*, 53(5):1203 –1218, june 2008.
- [12] Minh-Duc and Hua. Attitude estimation for accelerated vehicles using gps/ins measurements. *Control Engineering Practice*, 18(7):723 – 732, 2010. `|ce:title|Special Issue on Aerial Robotics|/ce:title|`.
- [13] R.C. Nelson. *Flight stability and automatic control*. WCB/McGraw Hill, 1998.
- [14] P. T. Pope and J. T. Webster. The use of an f-statistic in stepwise regression procedures. *Technometrics*, 14(2):pp. 327–340, 1972.
- [15] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3 edition, 2007.
- [16] Joseph Lee Rodgers, W. Alan Nicewander, and Larry Toothaker. Linearly independent, orthogonal, and uncorrelated variables. *The American Statistician*, 38(2):pp. 133–134, 1984.
- [17] K. Shcheglov, C. Evans, R. Gutierrez, and T.K. Tang. Temperature dependent characteristics of the jpl silicon mems gyroscope. In *Aerospace Conference Proceedings, 2000 IEEE*, volume 1, pages 403 –411 vol.1, 2000.
- [18] B.L. Stevens and F.L. Lewis. *Aircraft control and simulation*. John Wiley, 2003.
- [19] V. and Strejc. Least squares parameter estimation. *Automatica*, 16(5):535 – 550, 1980.

Appendices

A. Derivations

A.1. Physical pendulum

Assume an object with mass m , moment of inertia I_p around the pivoting axis that is a distance l from the center of mass. Gravity is acting on the center of mass and thus the total torque can be found by the product of the gravitational force and the moment arm:

$$\tau = mg \cdot l \sin \theta \quad (\text{A.1})$$

Using Newton's second law for angular momentum:

$$I_p \ddot{\theta} + mgl \sin \theta = 0 \quad (\text{A.2})$$

For small deflection angles $\sin(x)$ can be approximated with x . The above equation then becomes an ordinary second order differential equation:

$$I_p \ddot{\theta} + mgl \theta = 0 \quad (\text{A.3})$$

with the known solution:

$$\theta(t) = \theta_0 \cos \omega^2 t \quad (\text{A.4})$$

where $\omega^2 = \frac{mgl}{I_p}$. The pendulum period T can then be found:

$$T = \frac{2\pi}{\omega} = 2\pi \sqrt{\frac{I_p}{mgl}} \quad (\text{A.5})$$

A.2. Explicit Complementary Filter

By expanding the inner parentheses of equation (2.76),

$$\begin{aligned}
 v_i \hat{v}_i^T - \hat{v}_i v_i^T &= \begin{bmatrix} v_{i,x} \\ v_{i,y} \\ v_{i,z} \end{bmatrix} \begin{bmatrix} \hat{v}_{i,x} & \hat{v}_{i,y} & \hat{v}_{i,z} \end{bmatrix} - \begin{bmatrix} \hat{v}_{i,x} \\ \hat{v}_{i,y} \\ \hat{v}_{i,z} \end{bmatrix} \begin{bmatrix} v_{i,x} & v_{i,y} & v_{i,z} \end{bmatrix} \\
 &= \begin{bmatrix} v_{i,x} \hat{v}_{i,x} & v_{i,x} \hat{v}_{i,y} & v_{i,x} \hat{v}_{i,z} \\ v_{i,y} \hat{v}_{i,x} & v_{i,y} \hat{v}_{i,y} & v_{i,y} \hat{v}_{i,z} \\ v_{i,z} \hat{v}_{i,x} & v_{i,z} \hat{v}_{i,y} & v_{i,z} \hat{v}_{i,z} \end{bmatrix} - \begin{bmatrix} \hat{v}_{i,x} v_{i,x} & \hat{v}_{i,x} v_{i,y} & \hat{v}_{i,x} v_{i,z} \\ \hat{v}_{i,y} v_{i,x} & \hat{v}_{i,y} v_{i,y} & \hat{v}_{i,y} v_{i,z} \\ \hat{v}_{i,z} v_{i,x} & \hat{v}_{i,z} v_{i,y} & \hat{v}_{i,z} v_{i,z} \end{bmatrix} \\
 &= \begin{bmatrix} 0 & v_{i,x} \hat{v}_{i,y} - \hat{v}_{i,x} v_{i,y} & v_{i,x} \hat{v}_{i,z} - \hat{v}_{i,x} v_{i,z} \\ v_{i,y} \hat{v}_{i,x} - \hat{v}_{i,y} v_{i,x} & 0 & v_{i,y} \hat{v}_{i,z} - \hat{v}_{i,y} v_{i,z} \\ v_{i,z} \hat{v}_{i,x} - \hat{v}_{i,z} v_{i,x} & v_{i,z} \hat{v}_{i,y} - \hat{v}_{i,z} v_{i,y} & 0 \end{bmatrix}, \tag{A.6}
 \end{aligned}$$

it is clear that this is the skew-symmetric matrix $\mathbf{S}(v_i \times \hat{v}_i)$. This proves that equation (2.76) can be written as

$$\omega_{mes} = -\text{vex} \left(\frac{1}{2} \sum_{i=1}^n k_i \mathbf{S}(v_i \times \hat{v}_i) \right). \tag{A.7}$$

Now by using the definition of the vex operator (2.3) ω_{mes} can be expressed as:

$$\begin{aligned}
 \omega_{mes} &= -\frac{1}{2} \sum_{i=1}^n k_i \text{vex}(\mathbf{S}(v_i \times \hat{v}_i)) \\
 &= -\frac{1}{2} \sum_{i=1}^n k_i (v_i \times \hat{v}_i). \tag{A.8}
 \end{aligned}$$

ω_{mes} is then a weighted sum of n cross products of v and \hat{v} . v is the actual measured vector relative to the BODY frame, while \hat{v} is the known NED vector rotated to BODY using the current estimate of the attitude. The cross product will give an axis of rotation for correcting the error. The rotation magnitude will be scaled by the length of the cross product and the factor $\frac{k_i}{2}$.

B. Matlab figures

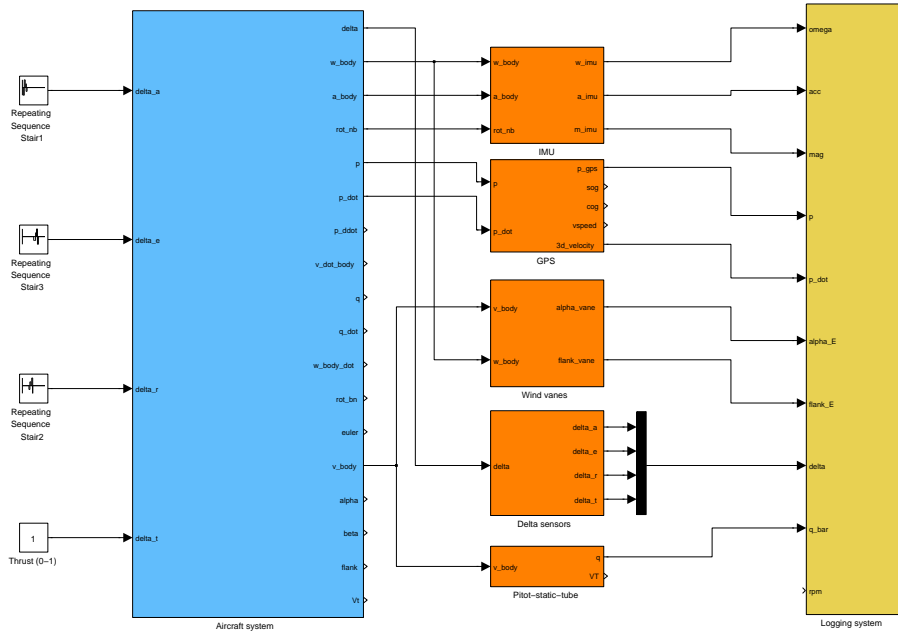


Figure B.1.: Overall simulator system.

C. Code implementation

C.1. Least Squares

C.2. Explicit Complementary Filter

```
%=====
% ecf_init.m – Setup initial variables in filter state
%                structure.
% k_p – tuning parameter for vector measurement feedback.
% k_i – tuning parameter for bias convergence.
% euler – initial euler angles.
% gyr_bias – initial gyro bias estimate.
%=====
function [x] = ecf_init(k_p, k_i, euler, gyr_bias)
x.k_p = k_p;
x.k_i = k_i;
x.q_hat = e2q(euler);
x.gyr_bias_hat = gyr_bias;
x.R = q2rot(x.q_hat);

%=====
% ecf_update.m – Update filter estimate with new IMU
%                measurements.
%
% x – filter state structure
% gyr – gyro angular velocity measurement
% k_acc – tuning parameter deciding impact of accelerometer
%          measurement
% acc – accelerometer measurement in body coordinates.
% acc_ref – acceleration reference vector in NED.
```

C. Code implementation

```
% k_mag - tuning parameter deciding impact of magnetometer
%           measurement
% mag - magnetometer measurement in body coordinates.
% mag_ref - magnetic field reference vector in NED.
% dt - time since last update.
%=====
function [x] = ecf_update(x, gyr, k_acc, acc, acc_ref, ...
                        k_mag, mag, mag_ref, dt)

%Rotate reference vectors to body coordinates using the
%current estimate of the attitude:
acc_ref = x.R*acc_ref';
mag_ref = x.R*mag_ref';

%Normalize vectors:
acc = acc / norm(acc);
acc_ref = acc_ref / norm(acc_ref);
mag = mag / norm(mag);
mag_ref = mag_ref / norm(mag_ref);

%Construct w_mes:
w_mes = 0.5*(k_acc*cross(acc, acc_ref) + ...
            k_mag*cross(mag, mag_ref));

%Calculate derivatives:
q_hat_dot = T_q(x.q_hat)*(gyr - x.gyr_bias_hat + ...
                        x.k_p*w_mes)';
gyr_bias_hat_dot = -x.k_i*w_mes;

%Integrate states using forward euler:
x.q_hat = x.q_hat + dt*q_hat_dot';
x.gyr_bias_hat = x.gyr_bias_hat + dt*gyr_bias_hat_dot;

%Normalize quaternion:
x.q_hat = x.q_hat / norm(x.q_hat);

%Calculate rotation matrix:
```

```
x.R = q2rot(x.q_hat);
```

C.3. Hua 2010

```
%
% hua_init.m – Initialize a new structure for holding
%                information used in Hua’s filter.
% vel – Initial velocity
%
```

```
function [x] = hua_init(vel)
x.Q_hat = eye(3);
x.v_hat = vel;
```

```
%
% hua_update.m – Update filter state after a specified
%                time dt when a velocity measurement from
%                GPS is available
% x – filter state structure
% k_1 – filter parameter
% k_v – filter parameter
% a – acceleration from an accelerometer
% vel – velocity measurement from GPS.
% dt – time step since last update.
%
```

```
function [x] = hua_update(x, k_1, k_v, vel, a, dt)
%Find error between current estimate and measurement:
v_err = vel - x.v_hat;

%Update v_hat:
x.v_hat = x.v_hat + dt * k_1 * v_err;

%Update Q:
x.Q_hat = x.Q_hat + dt * k_v * v_err * a';

%Normalize Q:
x.Q_hat = x.Q_hat / norm(x.Q_hat, 'fro') * sqrt(3);
```

C. Code implementation

```
%=====
% hua_predict.m – Predict filter state after a specified
%                   time dt using gyro and accelerometer
%                   measurements.
%
% x – filter state structure
% w – bias corrected angular velocity from gyro
% a – accelration from an accelerometer
% dt – time step
%=====
function [x] = hua_predict(x, w, a, dt)
x.v_hat_dot = [0 0 9.81]' + x.Q_hat*a;
x.v_hat = x.v_hat + dt * x.v_hat_dot;
x.Q_hat = x.Q_hat + dt * x.Q_hat*skew(w);
```


D. Digital attachment

A digital attachment is provided, containing the files used to generate the results in this thesis. The files are listed in Table D.2.

Table D.1.: Digital attachment.

Filename	Size (bytes)	MD5
appendix_d.zip	41559	00c256ac6b578e16246a07536bd1dc5b

It can be downloaded from http://dl.dropbox.com/u/58476755/appendix_d.zip.

Table D.2.: Files in digital attachment.

Filename	Description
case2.m	Script for generating data and displaying results for Case 2
case4.m	Script for generating data and displaying results for Case 4 after filtering.
e2q.m	Convert euler angles to quaternions.
ecf_update.m	Update ECF with new IMU data.
hua_predict.m	Predict filter states using IMU data.
lse_conv.m	Show convergence of least squares estimates.
mpoly.m	Moving polynomial filter implementation.
rot2e.m	Finds euler angles from a rotation matrix.
runsim.m	Initialize simulation settings.
skew.m	Calculates the skew-symmetric matrix given a vector.
vex.m	Inverse of skew.m.
case1.m	Script for generating data and display results for Case 1.
case3.m	Script for generating data and display results for Case 3.
cod.m	Calculate the coefficient of determination.
ecf_init.m	Initialize ECF state structure prior to filtering.
filtering.m	Uses ECF and Hua's filter for estimation of state variables using sensor data from simulation.
hua_init.m	Initialize state structure for Hua's filter prior to filtering.
hua_update.m	Update Hua's filter with GNSS velocity measurements.
lse.m	Ordinary Least Squares method.
q2rot.m	Create rotation matrix from quaternions.
rot_bw.m	Create rotation matrix from body to wind coordinates given angle of attack and sideslip.
simulate.mdl	SIMULINK simulation model.
T_q.m	Quaternion transformation matrix.