Modelling of a Hybrid UAV Using Test Flight Data

E.J.J. Smeur*, Q.P. Chu, G.C.H.E. de Croon, B. Remes, C. De Wagter, E. van der Horst
Delft University of Technology, Kluyverweg 1, Netherlands

ABSTRACT

The concept of an aircraft capable of both hover as well as fast forward flight (hybrid) has recently been implemented on unmanned aerial vehicles (UAV). Hybrid UAVs combine hover capability with long range and endurance. As UAVs are often required to operate without human intervention, there is a call for autonomous guidance of hybrid UAVs. Because the dynamics in the transition region from hover to forward flight are not well known, this research focusses on the development of a longitudinal model for a hybrid UAV based on test flight data. The same approach can be used for other types of airframes as well, allowing cheap and easy modelling. Sensors were logged for nine different flights and a Kalman filter was used for state estimation. The system was excited by doublet inputs on the commanded pitch and thrust. From the input-output response a piecewise linear model was estimated. This model was verified by comparing the measured doublet input response to the simulated response based on the model, yielding a very similar result.

1 INTRODUCTION

Now that Unmanned Aerial Vehicles (UAV) are adopted on a large scale, the number of applications increases as well. Many of these applications require take-off in an urbanised or otherwise obstructed environment as well as a long flight time and range. Traditional fixed wing UAVs require an open space and equipment for take-off and landing, while rotorcraft do not have as much range and endurance. A vehicle that combines the good properties of both is a *Hybrid* UAV. It can hover and transition into a fast forward flight. An example of such a UAV is the recently developed Quadshot [1].

Autonomy is a key concept for UAV aircraft, as this reduces operator workload, makes more complicated missions possible and it allows for operation beyond line of sight. Moreover, autonomy enables deployment of multiple UAVs at the same time (swarming). To accurately control the altitude and velocity of a hybrid vehicle during the transition from hover to forward flight and back, a model of the vehicle aerodynamics is necessary. There are several ways to obtain a model, a common way being wind tunnel measurements. This approach has been used for hybrids as well [2, 3]. However, wind tunnel time is scarce and expensive. An alternative is identification based on test flight data, which has been done for fixed wing aircraft before [4], but not for a hybrid, which flies at a regular angle of attack in forward flight, but at very high angles of attack in the transition phase.

This paper for the first time discusses identification of a model for a hybrid aircraft based on test flights. Though this is difficult due to the poor sensor quality in a low-cost UAV and the presence of turbulence, it is a very cheap way of identifying a model. Also, it can easily be redone when the airframe is changed.

Since in the end the goal is to control altitude and velocity, a longitudinal model is needed. The inputs available are the thrust setting and the pitch command, with the assumption that the attitude is sufficiently controlled by a gain scheduled PID controller. These two inputs are sufficient for longitudinal modelling and control [5].

The test flights are performed with a Quadshot with additional sensors, which are all logged for later analysis. From this data a state estimate is calculated for each moment in time by means of an Extended Kalman Filter. From the state estimates and the recorded inputs, a piecewise linear model is constructed by least squares fitting. The obtained model is verified by analysis of the modelling error and simulations. The simulations are compared to the measured data to assess how well they represent reality.

2 EXPERIMENT SET-UP

In order to perform structured test flights, a basic autonomous flight controller was implemented in the Paparazzi open source autopilot system based on previous efforts [6]. The inner loop consist of a gain scheduled PID controller based on the transition phase. For the outer loop two cases are defined: the hover case and the forward flight case. In hover a PID controller is used for position control with thrust vectoring and the altitude is controlled using the thrust. In forward flight the position is controlled with coordinated turns and the altitude is controlled with the pitch angle. In between these flight regimes the control inputs of both controllers are interpolated.

This controller provides decent performance, but the model is expected to improve on this. The way the experiment is conducted is by flying back and forth between two waypoints. The waypoints are more than 100 m apart, so that in flying from one to the other, the Quadshot spends enough time in forward flight that the system can be excited to obtain an input response.

*Email address: e.j.j.smeur@tudelft.nl

The Quadshot carries a 2200 mAh battery, which allows it to fly for about ten minutes. This is why the measurement procedure is optimised as much as possible. When the operator gives the launch command, the Quadshot autonomously takes off and flies to an altitude of 40 m. Upon reaching this altitude, it will proceed to the first waypoint. When it has arrived there, it waits for the command from the operator to fly back and forth to the second waypoint. Before giving this command, the operator specifies the pitch angle the Quadshot will fly at when flying forward. This ensures enough coverage of the relevant flight envelope.

The system has to be excited to make the dynamics visible. The available inputs are the thrust setting and the pitch setpoint. Doublet inputs are used on one of these inputs or both at the same time. Typically, the operator will select different doublet inputs during a flight. Finally, when the battery voltage gets low, the operator gives the command to land.

2.1 Apparatus

To reconstruct the state at each point in time after a test flight, sensors are necessary. The Quadshot is by default equipped with accelerometers, gyroscopes, magnetometers and a barometer. Additionally, a GPS receiver is added to provide information about the speed of the Quadshot. The GPS module is the Sparkfun GS407 with helical antenna, which delivers position information at 4 Hz with a typical accuracy of 4-5 m.

Next to that, the Quadshot is modified with a sensor boom. This sensor boom holds a Pitot tube (Eagletree airspeed v3) as well as an angle of attack sensor. This sensor is custom made from a wooden vane and an angle sensor with ball bearings (US Digital MA3 Miniature Absolute Magnetic Shaft Encoder). The vane is mass-balanced with a piece of lead on the tip. The boom extends 18 cm in front of the wing in order to reduce the effect of the rotors on the measurements.

The sensor boom is mounted under an angle of 33 degrees to optimize the airspeed measurement. The Quadshot is assumed to fly with angles of attack of 5 up to 55 degrees in the airspeed region of interest (transition regime). The expected mean of these angles is therefore 30 degrees. Obviously, the airspeed measurement is best if the angle of attack with respect to the sensor is zero. This is why the sensor boom is mounted under the angle that minimizes the mean.

2.2 System Excitation

The model parameters will be identified by flying at equilibrium conditions and then exciting the system through a control input. The input will be on one of the two controls, pitch or thrust, while the other is kept constant at the equilibrium value. Also, inputs will be given on both controls at the same time to determine the cross coupling. The control input is chosen to be a doublet, which is an often used input for model identification purposes [7]. A nice property of the doublet is that the deviation from the equilibrium flight path will not be large, since the input is applied both ways. Each part of the doublet is 2 seconds. The magnitude of the input on the pitch is 10 degrees and for the thrust it is 300 on a scale from 0 to 9600. Both up-down as well as down-up doublets are used to provide a varied excitation of the system.

The pitch angle is taken as an actuator and not a state, because it is assumed to be adequately controlled by a PID controller and the pitch control is assumed to be a lot faster than the airspeed and altitude dynamics.

2.3 Axis Conventions

Before going into the various states, inputs and system matrices, it is important to define the axis conventions. Two reference frames will be used in this work, the body fixed frame and the North East Down (NED) reference frame.

The body frame has the origin in the center of mass of the Quadshot, which is assumed to be coinciding with position of the IMU. The X axis is defined perpendicular to the wing, in the direction of the pylons that are far apart. The body Z axis points parallel to the wing in the direction that the propellers blow. Finally, the system is completed by the right hand rule, according to which the Y axis should be parallel to the wing pointing to the right. The body axis system is shown in Figure 1. Because of these body axes, the pitch angle at hover is zero degrees and will go to -90 degrees for forward flight. However, the alpha is still defined as the incidence angle of the wing, so $\alpha = \theta - \gamma + 90°$.

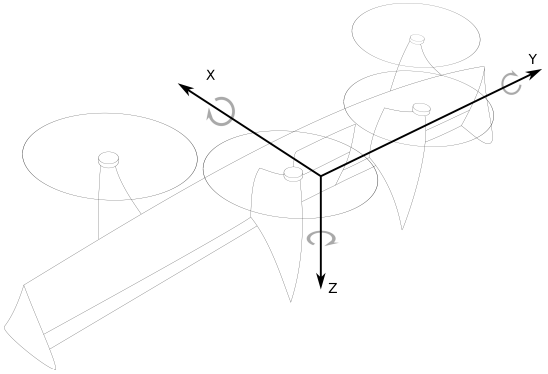

Figure 1: Body axes definition.

The origin of the NED frame is the starting point on the earth surface. The X axis points north, the Y axis points east and the Z axis points to the earth.

3 State Estimation

The extended Kalman filter used for the state estimation is based on the extended Kalman filter published by Simon [8]. The Kalman filter consists of two stages: prediction and innovation. In the prediction stage, a prediction of the state is calculated. This is done using the state from the previous iteration $\hat{\boldsymbol{x}}_{k-1|k-1}$, inputs to the system $\boldsymbol{u}_{k-1}$, and the state equations from a system model in Eq. (1).

$$\hat{\boldsymbol{x}}_{k|k-1} = f(\hat{\boldsymbol{x}}_{k-1|k-1}, \boldsymbol{u}_{k-1}) \tag{1}$$

The calculation of the measurement residual is given by Eq. (2), where h is a non-linear mapping of the state to the measurement vector.

$$\tilde{\boldsymbol{y}}_k = \boldsymbol{z}_k - h(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k) \tag{2}$$

The rest of the EKF equations are quite general and will not be discussed here. The state vector used in this case is described by Eq. (3). Here $\boldsymbol{q}$ is the attitude quaternion vector and $\boldsymbol{v}$ is the velocity vector in the body frame. $\boldsymbol{\lambda}_a$ and $\boldsymbol{\lambda}_\omega$ are the biases on the accelerometer and gyroscope respectively. Finally, $\boldsymbol{W}$ denotes the wind velocity vector in the NED frame.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{q} & \boldsymbol{v} & \boldsymbol{\lambda}_a & \boldsymbol{\lambda}_\omega & \boldsymbol{W} \end{bmatrix}^T \tag{3}$$

The input vector $\boldsymbol{u}_k$ is given by Eq. (4). The accelerometer measurements $\boldsymbol{a}_a$ and the gyroscope measurements $\boldsymbol{\omega}_g$ are used in Eq. (1) as inputs to the model. The other inputs are used as additional information in the calculation of $h(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k)$. The derivative of the GPS speed measurements gives the acceleration in the NED X and Y axes. Derivation of GPS measurements to obtain the linear accelerations is also discussed by Lopes et al. [9]. Derivation of the speed from the barometer yields the vertical acceleration along the NED Z axis. Together these accelerations form $\boldsymbol{a}_{\text{NED}}$. Finally q_{filtered} is a filtered version of the gyroscope measurement of q. For the latter four inputs no variance is taken into account, so they are filtered to remove high frequency noise. The pitch rate is filtered with a Butterworth filter with a cut-off frequency of 40 Hz to produce q_{filtered}. The acceleration from GPS and barometer are filtered with a cut-off frequency of four Hz.

$$\boldsymbol{u}_k = \begin{bmatrix} \boldsymbol{a}_a & \boldsymbol{\omega}_g & q_{\text{filtered}} & \boldsymbol{a}_{\text{NED}} \end{bmatrix}^T \tag{4}$$

The measurement vector $\boldsymbol{z}_k$ is given by Eq. (5). The measured magnetic field is denoted by $\boldsymbol{B}_m$, the measured accelerations by $\boldsymbol{a}_a$ and the combination of GPS horizontal speed and barometer derived vertical speed by $\boldsymbol{V}_{\text{NED}}$. V_p is the airspeed measured by the Pitot tube and α is the measured angle of attack. The GPS, airspeed sensor and angle of attack sensor have lower data rates than the IMU. To make use of every sample from the IMU, the other sensors have to be interpolated in between data samples, so that there is a data sample of every sensor at every k.

$$\boldsymbol{z}_k = \begin{bmatrix} \boldsymbol{B}_m & \boldsymbol{a}_a & \boldsymbol{V}_{\text{NED}} & V_p & \alpha \end{bmatrix}^T \tag{5}$$

The mapping of the state to the measurement domain is given by Eq. (6). In this calculation the rotation matrices $\boldsymbol{T}_{b0_{k|k-1}}$ and $\boldsymbol{T}_{0b_{k|k-1}}$ are used, which rotate a vector from the NED frame to the body reference frame and vice versa respectively. They are constructed using the quaternion from the state $\hat{\boldsymbol{x}}_{k|k-1}$. Since $\boldsymbol{T}_{0b_{k|k-1}}$ is the opposite rotation, $\boldsymbol{T}_{0b_{k|k-1}} = \boldsymbol{T}_{b0_{k|k-1}}{}^T$.

$$h(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k) = \begin{bmatrix} \boldsymbol{T}_{b0} \cdot \boldsymbol{B}_m \\ \boldsymbol{T}_{b0} \cdot \boldsymbol{g} + \boldsymbol{T}_{b0} \cdot \boldsymbol{a}_{\text{NED}} \\ \boldsymbol{T}_{0b} \cdot \boldsymbol{v}_{\text{b}} \\ \sin(33^\circ) u_a - \cos(33^\circ) w_a \\ \tan^{-1}\left(\frac{u_a - q_{\text{filtered}} d}{-w_a}\right) \end{bmatrix} \tag{6}$$

$$\begin{bmatrix} u_a & v_a & w_a \end{bmatrix}^T = \boldsymbol{v} - \boldsymbol{T}_{b0} \cdot \boldsymbol{W} \tag{7}$$

In Eq. (6) u_a and w_a are the body airspeeds with the wind taken into account as shown in Eq. (7). In the equation for α, u_a is corrected for the pitch rate, because the angle of attack sensor is located a distance d away from the center of gravity.

The state prediction equation is shown in Eq. (8). It shows the calculation of the derivatives and the integration of the state. The first line of the derivative calculation shows the quaternion derivative, where $\boldsymbol{\Omega}$ is the rate matrix as defined in Eq. (10). Second is the derivative of the body velocity. It is composed out of the measured accelerations minus the accelerometer biases. From this measured acceleration the gravity vector rotated to the body frame is subtracted. Finally, since u, v, and w are in a moving reference frame, a correction term is present to correct for rotations. The other derivatives are zero, since their derivatives are unpredictable based on the states and inputs.

$$f(\hat{\boldsymbol{x}}_{k-1|k-1}, \boldsymbol{u}_{k-1}) = \boldsymbol{x}_{k-1|k-1} + dt \cdot \begin{bmatrix} \frac{1}{2}\boldsymbol{\Omega}\ \boldsymbol{q} \\ \boldsymbol{a} - \boldsymbol{\lambda}_a + \boldsymbol{T}_{b0_{k|k-1}} \boldsymbol{g} + \boldsymbol{\omega} \times \boldsymbol{v}_{\text{b}} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{8}$$

$$\boldsymbol{\omega} = \begin{bmatrix} p - \lambda_p & q - \lambda_q & r - \lambda_r \end{bmatrix}^T \tag{9}$$

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \tag{10}$$

3.1 Modifications

The angle of attack measurement is found to be only valid when the norm of the airspeed is larger than 6 m/s. Therefore, the measurement covariance of the angle of attack is set to a high value when the norm of the airspeed is small. Similarly, the measurement covariance of the airspeed measurement is set to a high value when the measurement value is low, because the signal to noise ratio is very bad in that region.

Finally, the calculation of $\boldsymbol{a}_{\text{NED}}$ for Eq. (6) showed a delay of the GPS speed data. To find the right delay, the covariance of the GPS acceleration with the acceleration from the accelerometer (rotated to the NED frame) was calculated

for delays from 0 - 1 sec. For each flight, the delay that maximised the covariance was used to correct the GPS speed measurement data.

4 MODEL

The model that will be estimated is a piecewise linear aerodynamic model. Since the goal of the project is controlling airspeed and altitude, only the longitudinal dynamics are considered. Furthermore, the control of the attitude is sufficient as is, which is why the aerodynamic moment is not considered. Therefore, only the lift and drag coefficients will have to be estimated.

4.1 Model structure

According to Newton's second law of physics, the force equations in body axes are shown in Eq. (11) and Eq. (12). The aerodynamic forces are unknown functions f and g of their respective parameters. In these functions, δ_e is the input to the elevon and is assumed to be proportional to its deflection.

$$F_x = m(\dot{u} + qw - rv) = -mg\sin\theta + f\left(u, w, q, \delta_e\right) \quad (11)$$

$$F_z = m(\dot{w} + pv - qu) = mg\cos\phi\cos\theta + g\left(u, w, q, \delta_e, T\right) \quad (12)$$

The functions f and g are expected to be non-linear. To still be able to use simple but powerful linear theory, linear models will be estimated for different trim conditions. Together, these linear models will form a piecewise linear model. Doing the linearisation, the above equations can be written down as shown in Eq. (13) and Eq. (14). The acceleration in the body frame and the gravity term have been replaced by the measured acceleration. Note that $\theta, u, w, q, \delta_e$ and T will from now on be deviations from their respective equilibrium values.

$$ma_x = X_0 + X_u u + X_w w + X_q q + X_{\delta_e}\delta_e \quad (13)$$

$$ma_z = Z_0 + Z_u u + Z_w w + Z_q q + Z_{\delta_e}\delta_e + Z_T T \quad (14)$$

The model parameters will be estimated for a number of equilibrium flight conditions with different airspeeds. Between these conditions, the model parameters can be interpolated to provide a continuous model over the flight envelope.

4.2 Parameter Estimation

Eq. (13) and Eq. (14) should hold for every moment in time, so for every measurement sample. If the left hand side is written as a vector $\boldsymbol{y}$, the coefficients on the right hand side as a vector $\boldsymbol{\beta}$, and the variables on the right hand side as matrix $\boldsymbol{A}$, it can be written as in Eq. (15). From this equation a least squares solution is calculated.

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\beta} \quad (15)$$

4.3 Bins

Model coefficients are calculated for every doublet as described in the previous subsection. This results in a lot of different models for every airspeed. To reduce the number of models, the doublets are grouped into five bins of equal size ordered by the trim airspeed. For each of these bins one set of model coefficients is calculated. This way, random occurrences are averaged out and the identifiability is increased, as doublets on both inputs are taken together. For each entire bin, only one set of coefficients will be estimated, except for X_0 and Z_0. The latter two are still estimated for each doublet, because they represent the the trim condition which can differ per doublet.

If there are dependencies between the columns of $\boldsymbol{A}$, the matrix might be singular and the inverse does not exist. This can happen if one of the inputs is not used or certain dynamics are not excited. This problem is solved by taking different doublets together in the bins. In the calculation of the bin coefficients, 75% of the doublets are used. The remainder is saved for verification purposes.

4.4 Data Selection and Filtering

The data that will be used for the parameter estimation will be the data during the doublet only. This is because during the doublet the inputs are fixed to be either constant or the doublet input. This results in five seconds of data per doublet: one second with constant inputs, followed by a doublet with an up and a down time of both two seconds.

The accelerometer measurements that are used for the modelling contain a lot of noise from vibrations and are filtered with a Butterworth filter with a cut-off frequency of 40 Hz. The pitch rate measurement and the elevator command are filtered with a Butterworth filter with a cut-off frequency of 10 Hz. Finally, first order actuator dynamics are taken into account for the thrust. The coefficients of the first order filter were determined such that it gave a good fit with the data.

4.5 Simulation

Because in Eq. (13) and Eq. (14) the inertial acceleration is used, some rewriting is necessary to be able to do a simulation in the body reference frame. The inertial accelerations are replaced by the body accelerations along with the gravity term, as shown in Eq. (16) and Eq. (17).

$$\begin{aligned} m(\dot{u} + qw - rv) + mg\sin\theta = \\ X_0 + X_u u + X_w w + X_q q + X_{\delta_e}\delta_e \end{aligned} \quad (16)$$

$$\begin{aligned} m(\dot{w} + pv - qu) - mg\cos\phi\cos\theta = \\ Z_0 + Z_u u + Z_w w + Z_q q + Z_{\delta_e}\delta_e + Z_T T \end{aligned} \quad (17)$$

Because symmetrical flight is assumed, the terms rv, pv and ϕ are assumed to be zero. Furthermore, the gravity terms are linearised and brought to the right hand side. Then the equations are divided by m and the qw and qu terms are linearised, with the assumption that $q_0 = 0$. Then these terms

are brought to the right hand side as well. This results in Eq. (18) and Eq. (19).

$$\dot{u} = -g\cos\theta_0\theta + x_0 + x_u u + x_w w + (x_q - w_0)\, q + x_{\delta_e}\delta_e \tag{18}$$

$$\dot{w} = -g\sin\theta_0\theta + z_0 + z_u u + z_w w + (z_q + u_0)\, q + z_{\delta_e}\delta_e + z_T T \tag{19}$$

Where $x_* = \frac{X_*}{m}$ and $z_* = \frac{Z_*}{m}$. In state space form:

$$\begin{bmatrix} \dot{u} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} x_u & x_w \\ z_u & z_w \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix} + \begin{bmatrix} -g\cos\theta_0 & x_q - w_0 & 0 & x_{\delta_e} \\ -g\sin\theta_0 & z_q - u_0 & z_T & z_{\delta_e} \end{bmatrix} \begin{bmatrix} \theta \\ q \\ T \\ \delta_e \end{bmatrix} \tag{20}$$

It can be observed from Eq. (20) that θ and q are considered as inputs. These parameters are closely related and are controlled by the attitude controller. For the performed flights, the values are determined with the Extended Kalman Filter in order to simulate the same manoeuvre as performed in the real flight. If instead this model would be used to simulate the response to an arbitrary manoeuvre, of which no θ and q data are available, the controller dynamics have to be taken into account, as well as the relation between θ and q.

5 Results

The test flight experiment was conducted as described in Section 2. In total, nine flights provided useful data. This data was put through the Kalman Filter described in Section 3, of which results are shown in Section 5.1. The state estimates are used for the model parameter estimation as was described in Section 4, of which the results are shown in Section 5.2.

5.1 Kalman Filter

The Kalman filter developed in Section 3 provides stable state estimates for every performed flight. The estimated pitch angle is shown in Figure 2. Although the Kalman Filter uses quaternions, the attitude is converted to Euler angles afterwards for use in the model and for ease of perception. The figure shows the pitch setpoint angle, the estimate from Paparazzi and the estimate from the Kalman Filter. During this time segment, two doublet inputs are performed: one starts at $t = 292$ s and one at $t = 300$ s.

5.2 Model Estimation

In total, nine of these flights provided useful data. In these flights, 91 doublet inputs were given on thrust (11), pitch (37), or both (43). These inputs were given for different trim conditions. Figure 3 shows the trim pitch angle plotted against the trim airspeed. It also shows on which input a doublet was applied. From this figure a relation between pitch angle and airspeed for equilibrium flight can be observed.

The least squares parameter estimation as described in Section 4 is used on the data obtained from the Kalman filter for every doublet. This results in different model coefficients for every doublet. Plots of these coefficients are shown in Figure 4 and 5. Note that some of the coefficients x_T are zero. These are derived from the datasets in which only a doublet was given on pitch and the Δx_T was zero for the entire duration of the measurement.

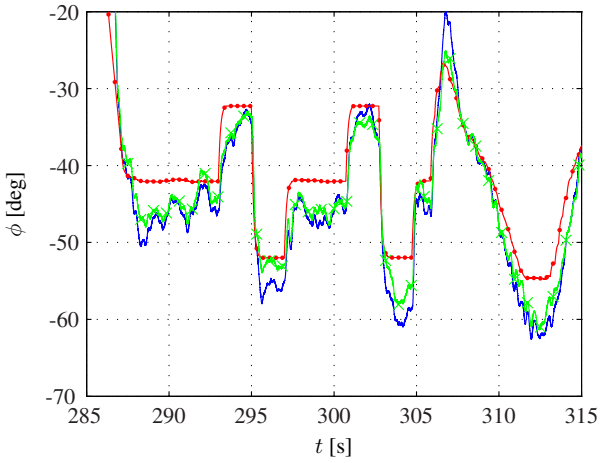

Figure 2: Pitch angle versus time. The red dotted line is the setpoint, the normal blue line is the Kalman Filter estimate and the green line with crosses is the estimate from the Paparazzi autopilot.

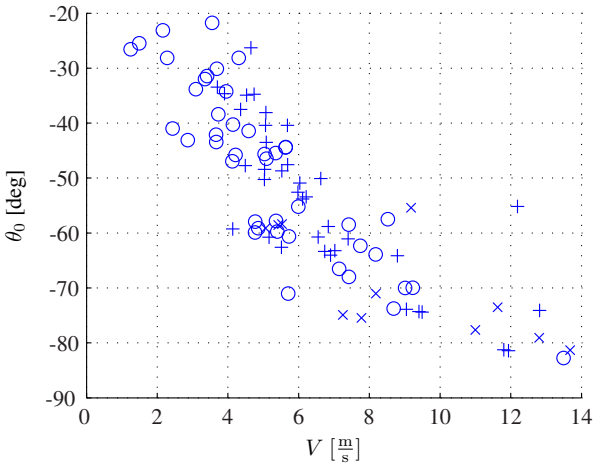

Figure 3: Doublet trim conditions. A + indicates a doublet on pitch, an x is a doublet on thrust and a ∘ is a doublet on both.

6 Analysis of the Results

6.1 Kalman Filter

Figure 2 shows that the estimate from the Kalman Filter is close to the estimate from Paparazzi. This provides confirmation of the performance of both. Additionally, it can be seen that the estimate sometimes deviates from the setpoint more than five degrees or overshoots. This indicates that an improvement of the attitude controller might be possible. Especially, it seems that the integrator gain could be higher, since around $t = 290$ s and $t = 300$ s the same error persists for multiple seconds.

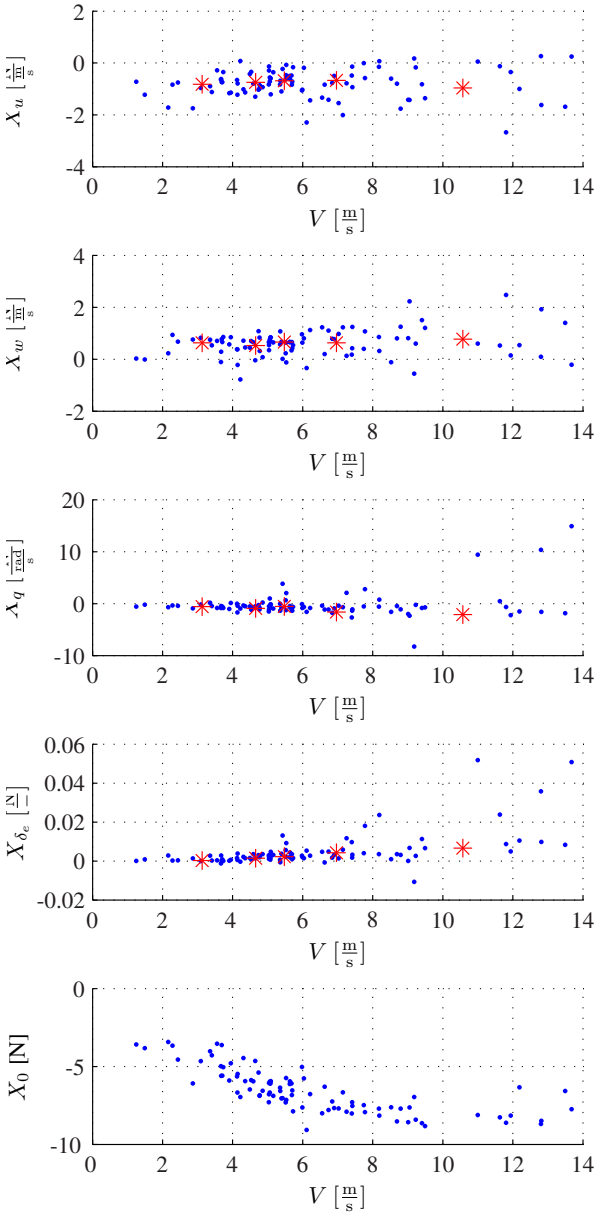

Figure 4: X axis model coefficients for every doublet. The red stars are bin coefficients.

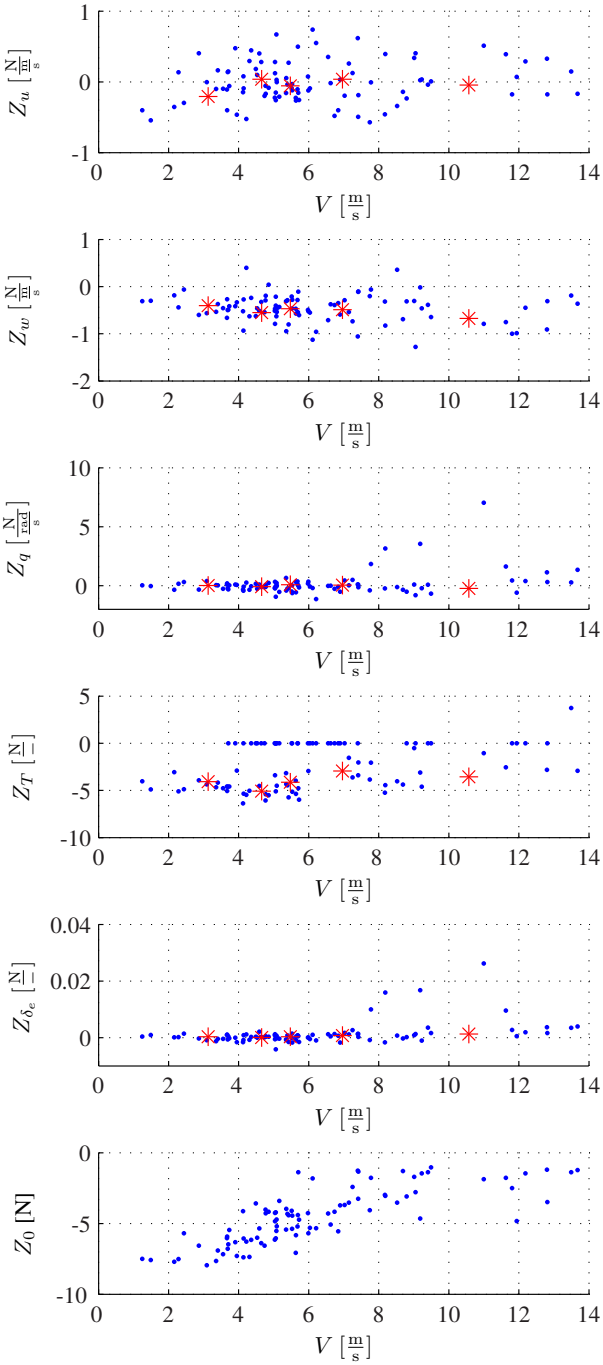

Figure 5: Z axis model coefficients for every doublet. The red stars are bin coefficients.

6.2 *Dynamic Pressure*

Theoretically, aerodynamic phenomena scale with the dynamic pressure $\frac{1}{2}\rho V^2$ When observing Figure 4 and 5, the effect of the dynamic pressure, a quadratic trend with V, is not visible. Especially the u and w coefficients are quite linear. A possible explanation may lie in the poor accuracy of the estimated body velocities. Another possibility is that in a lot of the cases with high angle of attack the wing was actually stalled, yielding different results than expected.

6.3 *Trends*

When looking at Figure 4 and 5, one can make some observations. Nearly for all doublets the x_u coefficient is negative. This is expected, because an increase in airspeed in the x axis will cause an increase in drag along this axis. The same holds for z_w. On the other hand, x_w is mostly positive. This can be explained as when w increases, the airspeed decreases

and less lift is produced. Depending on the pitch angle, this will generate an acceleration along the x axis.

6.4 Modelling Error

The modelling error is the mismatch between the actual acceleration and the acceleration calculated from the model, on the same dataset as used for the modelling. Figure 6 shows a plot of the measured acceleration (grey), the acceleration from the model based on just this doublet (red line with circles) and the acceleration from the model based on the second bin (blue line with crosses). From these figures it can be observed that all the main trends are modelled, but that the high frequency noise is not modelled. In this case, there is little difference between the model made specifically for this doublet and the model of the bin.

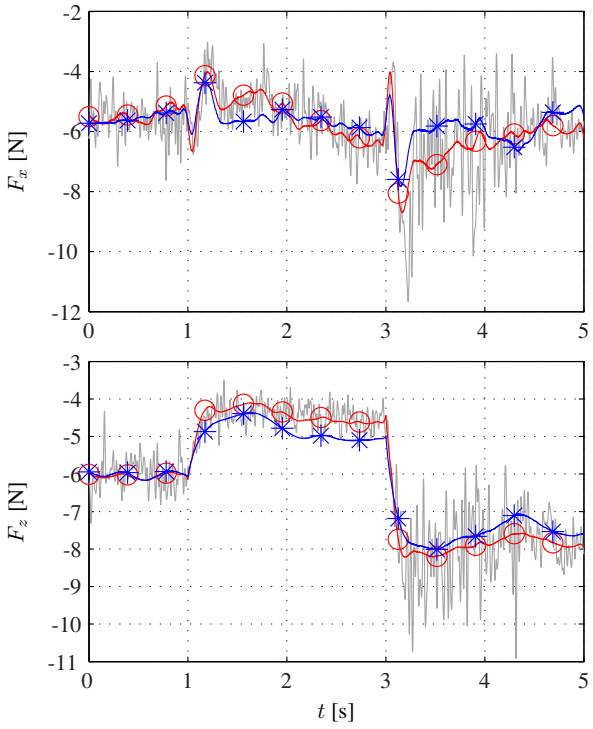

Figure 6: Model fit for an up-down pitch doublet. The grey line is the measured acceleration, the red line with circles is the best fit for this individual doublet and the blue line with crosses is the fit using the bin coefficients.

6.5 Simulation

The modelling error provides some verification, but the final test is a full simulation. This means calculation of the accelerations using the model coefficients and integrating these to get the body airspeeds. If this is done using inputs from one of the doublets, the simulated airspeeds can be compared to the recorded airspeeds.

When the linear coefficients are estimated with just one doublet, in 16% of the cases the state matrix has positive eigenvalues and an unstable model is produced. However, the models estimated with the bins are all stable, so these are used for simulation. The way the simulations are performed is by taking the input vector in Eq. (20), $\begin{bmatrix} \theta & q & T & \delta_e \end{bmatrix}$, from the actual test flight data for every time sample. The model coefficients are taken from the bin the verification doublet belongs to.

Four examples of such a simulation are shown in Figure 7. These simulations are selected because of their comparable trim speeds: 2.6 m/s $< u_{\text{trim}} <$ 4.1 m/s and -6.9 m/s $< w_{\text{trim}} <$ -6.5 m/s. Furthermore, the same inputs are applied, a down-up pitch along with an up-down thrust doublet (the doublet goes one way at t=1 s and the other way at t=3 s). The figure shows the simulated airspeed as well as the airspeed estimated by the Kalman Filter for u and w. Note again that w is positive in the direction of the 'tail' of the Quadshot. Because of visibility reasons, all simulations are depicted in blue and all measured airspeeds are depicted with dashed red lines, which means they can not be matched. It can be seen that the measurements are very noisy, but that the simulations show the same general trend as the measurements.

Clearly visible in both the simulation as well as the state estimate are the pitch changes, which cause the velocity to go from the X to the -Z axis and back. Because of the pitch down rotation w increases and u decreases. The thrust increase causes w to increase even more over time. The pitch up rotation causes velocity to shift axes from -Z to X again. After this w rapidly decreases. This can be explained because less thrust is given and a pitch up motion will increase lift and drag and therefore cause a deceleration along the -Z axis.

From a heuristic point of view, the simulation does the right thing. But the comparison with the real data is better for some simulations than for others. This might be because the accuracy of the simulation depends on the accuracy of the equilibrium condition. If the Quadshot is not in equilibrium when the doublet starts, or if the initial estimate of the trim airspeeds is incorrect, a steady state error will arise.

Because showing every verification plot is not feasible and we would like to have one number representing the accuracy of the model, a performance metric is needed. It is chosen to use the root mean squared error (RMSE) over all verification simulations. To give this number some meaning, it is compared to not having a model, resulting in simulated states $u = w = 0$ for every moment in time. In other words, the root mean square (RMS) of the u and w states from the Kalman Filter. This results in the table below.

	All doublets		Figure 7	
state variable	RMSE	RMS	RMSE	RMS
u	0.9862	1.1286	1.0557	1.1159
w	0.8432	1.1984	1.2443	1.5336

As can be seen from this table, the RMSE value is lower than the RMS value for both w and u. This means that the performance of the model is slightly better than without any model. The relatively large RMSE error might be caused by the start

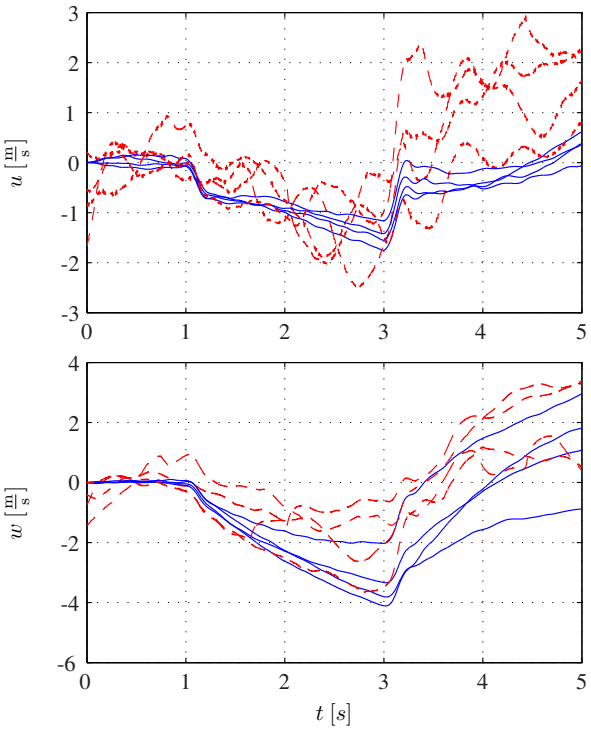

Figure 7: Four simulations of down-up pitch and up-down thrust doublets (continuous line) along with state estimates from the Kalman Filter (dashed line). 2.6 m/s $< u_{\text{trim}} <$ 4.1 m/s and -6.9 m/s $< w_{\text{trim}} <$ -6.5 m/s

of the doublet not being in an equilibrium condition, or an erroneous (initial) estimation of u and w.

7 Conclusion

Concluding, it has been shown that it is possible to identify the model of a low-cost UAV based on test flight data, avoiding the need for expensive equipment. To identify a model for the relevant flight envelope of the Quadshot, a measurement campaign was set up and carried out to obtain input response data for nine different flights. Autopilot code was written for the open source project Paparazzi to facilitate autonomous flights from take-off until landing and automatisation of the measurement process. A Kalman Filter was designed for the specific sensor set-up of the test Quadshot. This filter outputs a state estimate based on the sensor data. Doublet inputs were used to excite the system, resulting in 91 datasets. Five linear models were derived from the input response data, each for a different trim speed. Combined, these models provide a piecewise linear model of the Quadshot dynamics.

The model fitting was verified using analysis modelling error of the acceleration. Furthermore, it was observed that all five linear models were stable. Simulations with doublet inputs showed plausible responses, and comparison with recorded doublet responses proved the significance of the model.

Acknowledgements

This work was supported by the Delphi Consortium.

References

[1] Pranay Sinha, Piotr Esden-Tempski, Christopher Forrette, Jeffrey Gibboney, and Gregory Horn. Versatile, modular, extensible vtol aerial platform with autonomous flight mode transitions. IEEE Aerospace Conference, 2012.

[2] M. Itasse and J.M. Moschetta. Equilibrium transition study for a hybrid mav. Proceedings of the International Micro Air Vehicles conference 2011 summer edition, 2011.

[3] Michiel J. Van Nieuwstadt and Richard M. Murray. Rapid hover-to-forward-flight transitions for a thrust-vectored aircraft. Journal of Guidance, Control, and Dynamics, Volume 21, Issue 1, pp. 93-100, 1998.

[4] Dongwon Jung and Panagiotis Tsiotras. Modeling and hardware-in-the-loop simulation for a small unmanned aerial vehicle. AIAA Infotech at Aerospace, 2007.

[5] Nathan B. Knoebel, Stephen R. Osborne, Deryl O. Snyder, Timothy W. McLain, Randal W. Beard, and Andrew M. Eldredge. Preliminary modeling, control, and trajectory design for miniature autonomous tailsitters. AIAA Guidance, Navigation, and Control Conference and Exhibit, 2006.

[6] C. De Wagter, D. Dokter, G. de Croon, and B. Remes. Multi-lifting-device uav autonomous flight at any transition percentage. Proceeding of: EuroGNC 2013, 2013.

[7] Andrei Dorobantu, Austin M. Murch, Bernie Mettler, and Gary J. Balas. Frequency domain system identification for a small, low-cost, fixed-wing uav. Proceedings of AIAA Guidance, Navigation, and Control Conference, 2011.

[8] Dan Simon. Optimal State Estimation, page 409. John Wiley and Sons Inc., 2006.

[9] H.D. Lopes, E. Kampen, and Q.P. Chu. Attitude determination of highly dynamic fixed-wing uavs with gps/mems-ahrs integration. AIAA Guidance, Navigation, and Control Conference, 2012.