State Estimation of In-Flight Aircraft Centre of Gravity

by

Andrew John Stanley

A doctoral thesis submitted in partial fulfilment of the requirements

for the award of Doctor of Philosophy of

Loughborough University

December, 2011

Synopsis

For all types of aircraft (civil, military, manned, unmanned) the aircraft designer specifies a safe range for the centre of gravity (cg) of the aircraft and designs the aircraft to operate safely within these limits. Changes to the cg may affect aircraft stability, performance and fuel economy so it is an important system parameter.

Changes to the in-flight cg position have traditionally been estimated by calculating fuel burn and from that calculating the change in weight and hence change in cg. Other techniques to estimate in-flight aircraft cg are included in the literature review.

The motivation for additional cg estimation techniques arise from the potential benefits they offer to a Flight Control System (FCS). These benefits include the potential for improved fault detection and an improved FCS design with better aircraft performance and fuel economy.

State estimation using Kalman filters has been used since the 1960's in many fields of application including the aerospace industry. This thesis will introduce the concept of using state estimation to detect the unexpected angular acceleration associated with a cg change. This state estimation concept is applied to a linear Phantom aircraft model and then to a complex non-linear aircraft model of a delta-canard military aircraft, called ADMIRE.

The most common state estimation approach used with non-linear systems is the Extended Kalman Filter (EKF), but an alternative approach is proposed in which the pitching and roll moment coefficient derivatives are selectively modified based upon the aircraft angle of attack, speed and altitude. Both longitudinal and lateral cg estimators are described and examples of their performance are provided and compared with an EKF version of the estimator. A discrete version of the estimator is also described and used with a hardware fuel rig. Faults are applied to the fuel rig and it is proposed that the estimator could aid the fault diagnosis.

In a real implementation the aircraft will not be precisely modelled, therefore a sequence of robustness tests are included to identify the critical aircraft parameters affecting the estimator.

The results show that a cg estimator based upon a Kalman filter, and using a selective coefficient correction approach, can satisfy the performance requirements specified by the industrial sponsor, BAE Systems.

Acknowledgements

I would like to thank my supervisor Professor Roger Goodall for his support and advice during this research.

I would also like to thank BAE Systems for making this research possible. In particular I would like to thank Dr Jonathan Irving for his expert advice and knowledge, and for providing the cg modifier model which was used in section 7. My thanks also go to Dr John Pearson for providing access to the hardware fuel rig at the SEIC (Systems Engineering Innovation Centre) in Loughborough, and allowing me to use it with the cg estimator model in section 8. I would also like to thank Brian Caldwell for the hospitality shown to me on my visits to BAE at Warton Aerodrome.

Finally, I would like to thank my wife, Dr Ana Salagean, for helping to provide the time and support needed to perform the research and write up this thesis.

Publications

Stanley, A. and Goodall, R. (2009). Estimation of aircraft centre of gravity using Kalman filters, *Proc. of European Control Conference*, Aug 23-26 2009, pp 1848–1853 (included in Appendix A)

Stanley, A. and Goodall, R. (2010). Estimation of the centre of gravity of a manoeuvring aircraft using Kalman filters and the ADMIRE aircraft model, *Proc. of 5th IFAC Symposium on Mechatronic Systems*, Sep 13-15 2010, pp 1-7 (included in Appendix B)

Table of Contents

Table of Figures

Abbreviations

cg   centre of gravity (m)

EKF   extended Kalman filter

FCS   Flight Control System

KF   Kalman filter

List of Symbols

A   state matrix

B   input matrix

b_{ref} reference wing span (m)

c   mean aerodynamic chord (m)

C   output matrix

C_{ref} mean aerodynamic chord (m)

Crm_{tot} rolling moment coefficient

Cm_{tot} pitching moment coefficient

Cym_{tot} yaw moment coefficient

D   direct matrix

dlc   left canard angle (radians)

dle   leading edge flap angle (radians)

dloe   left outer elevon angle (radians)

dlie   left inner elevon angle (radians)

dr   rudder angle (radians)

drc   right canard angle (radians)

drie   right inner elevon angle (radians)

droe   right outer elevon angle (radians)

dxcg   change in centre of gravity along x axis (m)

dycg   change in centre of gravity along y axis (m)

F_z total force in body-fixed z axis (N)

F_x total force in body-fixed x axis (N)

g   acceleration due to gravity

G   gain matrix

h   longitudinal cg position measured from leading edge wing root (m)

h_0 aerodynamic centre position on reference chord (m)

I   moment of inertia (kg m^2)

I_x moment of inertia in roll (kg m^2)

I_{xy} product of inertia about x and y axes (kg m^2)

I_{xz} product of inertia about x and z axes (kg m^2)

I_y moment of inertia in pitch (kg m^2)

I_{yz} product of inertia about y and z axes (kg m^2)

I_z moment of inertia in yaw (kg m^2)

K   Kalman gain

K_k Kalman gain at time step k

ldg   landing gear (Boolean)

L_w wing or wing-body lift (N)

L_t tailplane lift (N)

l_w distance of L_w from cg (m)

l_t distance of L_t from cg (m)

M   mach number; pitching moment

mac   mean aerodynamic chord (m)

M_0 wing moment about aerodynamic centre (N m)

M_x roll moment (N m)

M_z yaw moment (N m)

M_y pitching moment (N m)

nx   longitudinal acceleration ('g')

ny   lateral acceleration ('g')

nz	normal acceleration ('g')
p	roll rate (radians/s)
P_k^-	a priori error covariance matrix at time step k
P_k^+	a posterior error covariance matrix at time step k
$\dot{p}_b$	roll acceleration (radians/s^2)
$\dot{p}_{unex}$	unexpected roll acceleration (radians/s^2)
psa	static air pressure (mbar)
q	pitch rate (radians/s)
$\dot{q}_b$	pitch acceleration (radians/s^2)
$\dot{q}_{unex}$	unexpected pitch acceleration (radians/s^2)
$\bar{q}$	dynamic pressure (pascals)
Q	process noise covariance matrix
r	yaw rate (radians/s)
$\dot{r}_b$	yaw acceleration (radians/s^2)
R	measurement noise covariance matrix
S_{ref}	wing surface area (m^2)
t	torque (N m)
tss	throttle setting (0..1)
T_x	engine thrust along the x axis (N)
T_z	engine thrust along the z axis (N)
u	axial velocity (m/s)
U	input vector
v	measurement noise
V_T	freestream airspeed (m/s)
w	process noise; normal velocity (m/s)

W   aircraft weight through cg (kg)

x   longitudinal coordinate in axis system; state vector

x_{cg} centre of gravity along the x axis (m)

x_k state vector at time step k

$\hat{x}_k$ predicted state vector at time step k

X   axial force component (N)

y   lateral coordinate in axis system

y_k output vector at time step k

z   normal coordinate in axis system

z_{cg} centre of gravity along the z axis

Z   normal force component (N)

α   angle of attack (radians); angular acceleration (radians/s^2)

β   sideslip angle (radians)

θ   pitch angle (radians)

φ   roll angle (radians)

ψ   yaw angle (radians)

δe elevator deflection (radians)

ρ   mass density (kg/m^3)

η   elevator angle perturbation (radians)

Examples of notation

Xu   shorthand notation to denote Phantom model dimensionless derivative $\frac{\partial x}{\partial u}$

Cl_{dlc} shorthand notation to denote ADMIRE coefficient for the rolling moment derivative $\frac{\partial p}{\partial dlc}$

Cm_α shorthand notation to denote ADMIRE coefficient for the pitch moment derivative $\frac{\partial q}{\partial \alpha}$

1. Introduction

Aircraft are designed to operate within a specified centre of gravity range to ensure their safe operation. Apart from the safe operation of the aircraft the cg location affects the aircraft stability, performance and fuel consumption, so the cg value is an important piece of data for the Flight Control System (FCS).

The cg can be calculated when the aircraft is on the ground by weighing the aircraft on scales, but in flight the FCS resorts to estimation techniques to estimate the changes to the cg as fuel is consumed.

This thesis describes a new approach to estimate the in-flight aircraft cg by using state estimation techniques.

The following subsections detail the motivation for this research, the unique contribution made in this thesis to a potential new cg estimator, and the overall structure of this thesis.

This research has been sponsored by the EPSRC and BAE Systems.

1.1 Motivation

The motivation for this research are the potential benefits that an additional cg estimate may provide to the FCS. A model based cg estimator could:

a) Provide an additional method to estimate the centre of gravity when other techniques e.g. accelerometers, fuel flow measurements, have failed.

b) Provide an additional piece of information to the FCS to improve fault detection e.g confirm the failure to deploy stores, confirm the presence of a fuel leak.

c) Allow for cg position dependent FCS design 'scheduling', resulting in a more robust design, particularly at the extreme aft/lateral cg positions.

d) Allow better normalisation of aircraft handling qualities with aircraft cg. For example forward cg configurations typically tend to be more sluggish in pitch manoeuvres. In addition, lateral cg offsets induce rolling moments which need to be compensated for within the control laws to avoid intrusive bank angle motion.

It is noted in (Orgun & Flanigan, 1991) that "the practice has been to use a worst case

estimate of the center of gravity in automatic flight control situations in which center of gravity is an important parameter" which led to "less than optimal control action". Therefore a more accurate cg estimate which led to an improved worst case estimate could be used to improve the control action, and hence aircraft performance and economy.

1.2 Research Contribution

This thesis will propose the use of model based techniques to estimate aircraft cg. Modelling techniques have been used extensively in the aerospace sector (Hutchinson, 1984), but at the start of this research to the author's knowledge they had not been used with the sole purpose of estimating the in-flight aircraft cg. Recent research by Abraham and Costello (2009) does use an Extended Kalman Filter (EKF) to estimate the weight and cg of a helicopter, however their approach is different to the one proposed in this thesis.

This thesis will propose the novel method of using Kalman filters to obtain the unexpected angular acceleration caused by a change in cg, and then converting the unexpected angular acceleration into an estimate of the change in cg. The unexpected angular acceleration is the difference between the acceleration expected by the model in the estimator and the actual aircraft angular acceleration.

This thesis will propose a solution to the estimation problem applied to a non-linear aircraft model by the selective modification of the pitching or rolling moment coefficients, as opposed to the more traditional approach of using an EKF. The results of both approaches are compared and analysed.

Finally, this thesis will provide the simulation results in which the cg estimator has been used to detect changes in cg caused by faults in an experimental aircraft fuel rig, and suggest that a cg estimator could be used as an aid in fault detection.

1.3 Thesis Structure

Section 2 provides background information regarding aircraft centre of gravity, state estimation and Kalman filters, and also includes a literature review of in-flight cg estimation techniques.

Section 3 describes the performance requirements placed upon the cg estimator by the research sponsors BAE Systems. It also describes the methodology adopted to estimate the

change in cg using state augmentation to estimate the unexpected angular acceleration.

Section 4 describes the use of a linear aircraft model which is modified to enable the injection of moments representative of a cg change. It also describes a longitudinal cg estimator and provides some sample results of the estimation. The section includes results from robustness tests in which the linear aircraft model coefficients are systematically modified.

Section 5 describes the development of both a longitudinal and lateral cg estimator to a non-linear aircraft model, ADMIRE, which is a complex model of a delta-canard military aircraft.

Section 6 describes the modification of the cg estimator to remain accurate when the aircraft is manoeuvring, or is at a different speed to the estimator configuration. Robustness testing is performed to assess the accuracy of the estimator in a real system when the aircraft is not modelled perfectly.

Section 7 describes a combined longitudinal and lateral cg estimator, and provides example results using a model which incorporates a BAE developed cg modifier model. The BAE cg modifier model includes a model for the effect of aircraft acceleration on fuel sloshing and hence the cg.

Section 8 describes the implementation of a discrete version of the cg estimator used in conjunction with a hardware fuel rig. It contains examples when hardware faults are applied to the fuel rig to demonstrate the fault detection capability of the estimator.

Section 9 investigates the use of an Extended Kalman Filter (EKF) in which a linearised model of the aircraft is continuously calculated. The results from this version of an EKF are compared with those from section 6.

Section 10 summarises the results from this research and suggest future work that may be done.

The appendices include conference papers, relevant Matlab code, robustness test results, and samples of the table data used in the section 6 estimator.

2. Project Background

Section 2.1 provides background information about the aircraft cg and its effect upon aircraft stability, performance and fuel economy. It also provides background information about Kalman filters and the use of state estimation techniques.

Section 2.2 provides a literature review of current methods used to measure or estimate the aircraft centre of gravity.

Section 2.3 provides an introduction to aircraft flight dynamic equations and the terminology that will be used later in this thesis.

The literature review is summarised in section 2.4.

2.1 Background

2.1.1 Aircraft Centre of Gravity

The centre of gravity is an important piece of data about an aircraft because it is the location about which the aircraft rotates (NASA, 2006). "It is the mass center of the airplane, or the theoretical point at which the entire weight of the aircraft is assumed to be concentrated" (Federal Aviation Administration, 2004). The aircraft performance is dependent upon the cg location because all moments are derived with respect to the centre of gravity location, therefore the cg location is required in an FCS to accurately predict and control the aircraft performance.
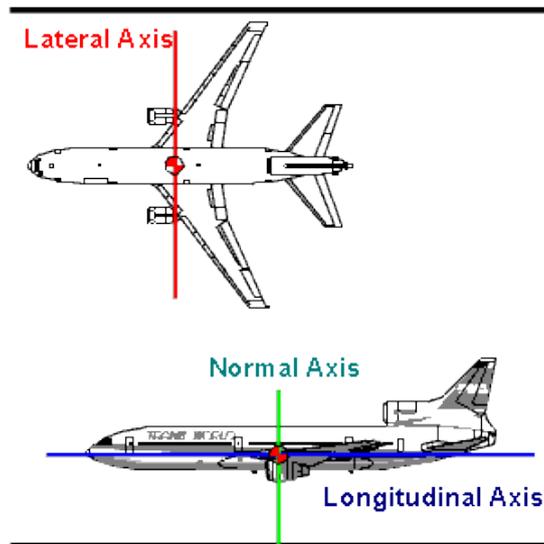
Figure 1. Aircraft axes through cg (Geistware, 2010).

Figure 1 shows the centre of gravity of an aircraft and the three rotational axes about the cg.

A stable aircraft has the capability to be trimmed and will then maintain straight and level flight without further control surface action.

Aircraft in Level Flight
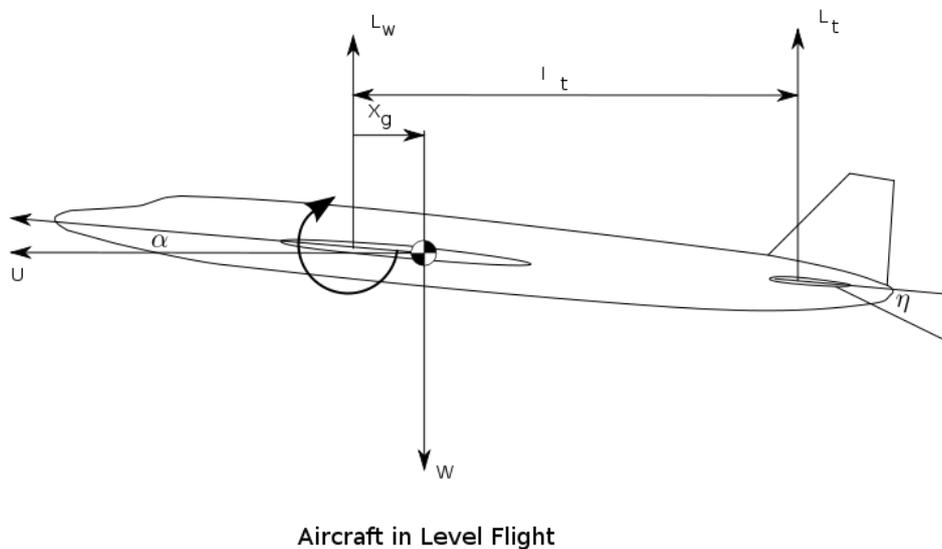
Figure 2: Aircraft in level flight (Wikipedia, 2010).

Figure 2 gives a side view of a typical aircraft. In a trimmed aircraft the lift forces L_w and L_t match the weight W which is focused through the centre of gravity. The cg is aft of the aerodynamic centre of the main wing. A more detailed analysis to calculate the aircraft cg in terms of the aircraft forces is contained in section 4.

To be statically stable any disturbance must result in the production of a restoring moment. According to Phillips (2004, p354) "A stable airplane in free flight will always seek the angle of attack that makes the pitching moment about the cg equal to zero". Therefore the location of the centre of gravity greatly affects the control stability of the aircraft. It is also noted in Phillips (2004: 366) that the further forward the cg the more stable the aircraft becomes, but the aircraft requires more negative elevator deflection for trim. In a canard configuration to maintain stability the cg must be located between the wing and the canard (Phillips, 2004: 385).

More explicitly according to (Federal Aviation Administration, 2007) in a stable aircraft if a rising gust of wind causes the aircraft nose to pitch up then the aircraft will slow down, the downward force on the tail will decrease, and the weight concentrated at the cg will pull the aircraft nose back down. Alternatively if the aircraft nose drops then the airspeed will increase and the increased tail load will bring the nose back up.

If the cg position is too far aft then the aircraft will be unstable because commanding the aircraft elevator will be unable to bring the nose back down. Conversely, if the cg is too far forward then the elevator commands the nose back up, and the aircraft will have to fly at a higher angle of attack and with increased drag.

So it can be seen that the centre of gravity position is fundamental to aircraft stability, and can be related to fuel consumption since the performance of the aircraft is affected by the attitude and trim of the aircraft. It is possible to save fuel by moving the centre of gravity to reduce the required elevator down load and hence drag. The ideal location for the centre of gravity is carefully determined by the aircraft designers and an allowable range will have been calculated for the flight control system.

The lateral cg is also important to aircraft performance, but not as important as longitudinal cg. As noted in Federal Aviation Administration (2007) "The lateral cg may be upset by uneven fuel loading or burnoff" which is corrected by using the aileron trim tab. "The deflected trim tab deflects the aileron to produce additional lift on the heavy side, but it also produces additional drag, and the airplane flies inefficiently" (Federal Aviation Administration, 2007).

The aircraft cg will vary as fuel is used up or sloshes in its tanks, or stores are deployed, or cargo is added or removed. Fuel sloshing is a particular problem in future aircraft design according to Baeten and Stern (2008), who state that fuel sloshing is recognised as one of the "limiting factors in the design of airborne tank structures and flight control systems". The high agility of combat aircraft and cruise missiles requires exact knowledge of the cg and inertia. When a fuel tank is full then the cg and inertia behaves similar to a solid, however when partly filled the residual fuel sloshes in its tanks affecting cg location, inertia and hence aircraft performance.

The aircraft cg may also vary if the FCS employs weight management to improve aircraft performance. For example it is described in Filippone (2006: 43) how fuel is pumped from the forward to the rear fuel tanks in Concorde to move the centre of gravity, because when the aircraft transitions from subsonic to supersonic flight the centre of pressure moves about 2 metres to the rear.

Unlike aircraft data such as acceleration or altitude, the aircraft centre of gravity is not something that can be directly measured during flight, therefore the aircraft centre of gravity is a dynamic value that requires estimating.

2.1.2 State Estimation of Linear Time Invariant Systems

There is a whole field of literature on the subject of estimation, be it estimation of signals, parameters or state. The process of selecting the best estimate leads to the use of statistical techniques to calculate the best estimate, and the best estimate requires some criteria of what 'best' is.

"Estimation is the process of inferring the value of a quantity of interest from indirect, inaccurate and uncertain observations." (Bar-Shalom, Rong Li, & Kirubarajan, 2001: 1)

State estimation requires that the states that cannot be measured can still be inferred from the available measurements, and this leads to the requirement that the system is observable. "A process is called observable if from the measurements of the output it is possible to determine the state of the process." (Eykhoff, 1974).

The model of the aircraft system will consist of data that is time-invariant (e.g. parameters) and data that evolves in time according to an equation (state).

The estimate of an unmeasurable state is through a system called an observer, see Figure 3.
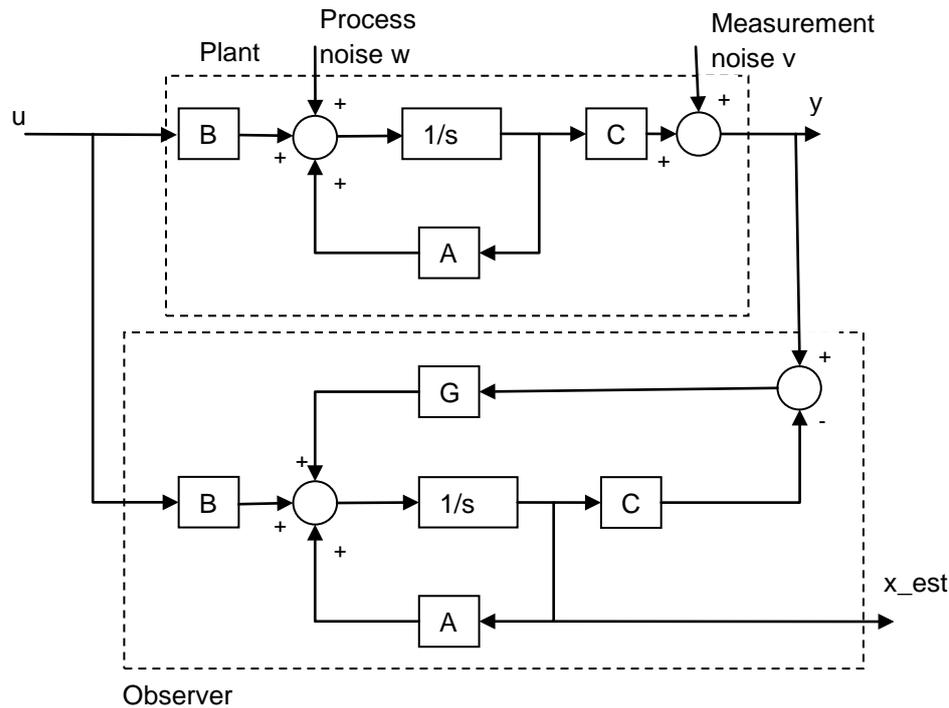
Figure 3: Full-order observer

A full-order observer estimates all of the state variables of the plant. The observer contains a state-space model to represent the physical system. In the state-space approach the state variables and state equations describe the system, where the state equations are given by:

$$\dot{x} = Ax + Bu + w \qquad (1)$$

where

        x is the state vector of dimension n_x

        u is the input vector of dimension n_u

        w is the process noise vector of dimension n_x

        A is the state matrix of dimension $n_x \times n_x$

        B is the input gain of dimension $n_x \times n_u$

The output of the system is:

$$y = Cx + v \qquad (2)$$

where

     y is the output vector of dimension n_y

     C is the measurement matrix of dimension n_y x n_x

     v is the measurement noise vector of dimension n_y

The full-order observer has an additional gain matrix G (see Figure 3) to control the dynamic performance of the observer. It can be seen that the input into G is the difference between the estimated outputs and plant measurements.

The observer design problem is to obtain a gain G that over time minimizes the error between the observer and system outputs.

$$x(t) - \hat{x}(t) =: e(t) \rightarrow 0 \tag{3}$$

It is explained in (Dutton, Thompson & Barraclough, 1997: 442) that the design equation for the observer is

$$F = A - GC \tag{4}$$

Therefore the design procedure is to choose the poles of F to give the desired dynamic performance of the observer and then use (4) to obtain the value of G.

It is recognized in control theory that the poles "may be arbitrarily assigned to desired locations if and only if (C,A) is observable" (Stevens & Lewis, 2003: 541). A good example of observer design can be found in (Carnegie Mellon University, 1997) in which a pole placement technique is used to place the poles at least five times further to the left than the dominant poles of the system. In (Stevens & Lewis, 2003: 543) and (Dutton, Thompson & Barraclough, 1997: 445) they both suggest the observer poles should be at least five to ten times faster than the fastest plant pole. One point to note is that the faster the observer is, the more noise it will pass through.

A full-order observer should give good estimates if the system is initialized to a known state and the measurements can be predicted. However in practice a model will suffer from an imperfect initial estimate and imperfectly predictable evolution. It will also suffer from noise on the process and on the measurements (as shown in Figure 3), so any estimation will have

to filter out the noise from its results. It is in such a system that the Kalman filter is preferable to the observer.

In 1960 Kalman published his paper (Kalman, 1960) which presented an optimum linear estimator as a recursive operation, and this became known as the Kalman filter. His estimator was later extended with Bucy to work in continuous form (Kalman & Bucy, 1961) and this solution became known as the Kalman-Bucy filter.

The relationship between continuous and discrete observers, and Kalman filters is illustrated in Figure 4.
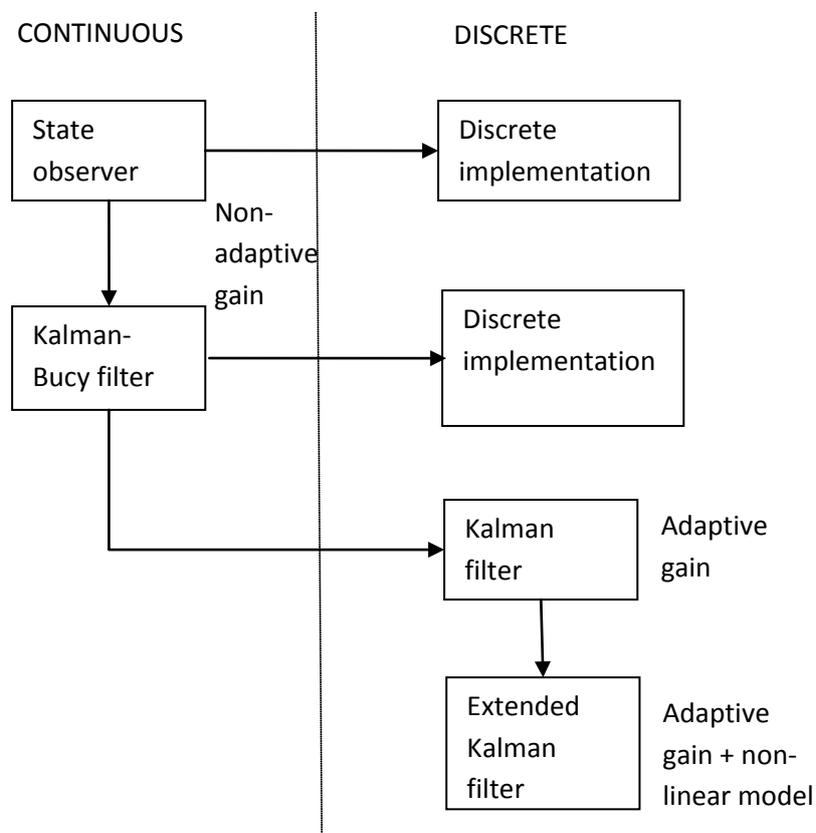
Figure 4: Observer and Kalman filter overview

As shown in Figure 4, and explained in more detail later, an observer and Kalman-Bucy filter have a fixed gain used to correct the state estimate. In contrast the discrete-time Kalman filter and Extended Kalman Filter employ an adaptive gain in which the gain is continually recalculated. The Kalman filter and Luenberger observer are identical in structure, however the Luenberger observer would be used in a deterministic system whereas the Kalman filter is

aimed towards a stochastic system and provides a design process based upon quantifying noise levels. It is explained later that the Kalman gain is continously recalculated (adaptive gain) but for linear systems the values in the gain matrix eventually converge to a constant (non-adaptive gain).

In this thesis an estimator containing a Kalman-Bucy filter with a fixed non-adaptive gain is described, and then a discrete version of this filter is also described. Finally an estimator containing an Extended Kalman Filter with a recalculated Kalman gain (adaptive gain) is described.

"In almost all realistic situations the observations made on the system under study are contaminated with random influences (disturbances, errors)". One has to use statistical methods to obtain a 'best' result by 'filtering out' the influence of the disturbances (Eykhoff, 1974).

The choice of estimation technique is dependent upon the problem. If the process and measurements are subject to zero mean Gaussian noise and the process is characterized by linear differential (or difference) equations with known coefficients, then the Kalman-Bucy filter is the optimal solution to minimize the mean square error (Bar-Shalom, Rong Li, & Kirubarajan, 2001: 355).

According to (Lewis, 1986: 68) the discrete-time Kalman filter is the best linear estimator if the initial estimate x_0, process noise w and measurement noise v have arbitrary statistics, and it is the optimal estimator if x_0, w and v are normal.

Fundamental to the Kalman filter is the Kalman gain which is applied to the difference between measured and estimated values, the same as matrix G in Figure 3. The calculation of the Kalman gain is defined in many books, for example (Dutton, Thompson, & Barraclough 1997: 484), (Lewis, 1986: 68) and (Grewal & Andrews, 2001: 116). These calculations require statistical data regarding the process noise and measurement noise and this data is defined by two matrices Q and R. The sensor noise can be measured or derived from the manufacturer's specifications and this allows the covariance data in R to be calculated. The process noise is very difficult to measure and so the covariance data in Q is obtained by tuning the filter. When the Q and R matrices have been calculated the Kalman gain may be precomputed because it does not depend upon the noise samples but on the noise statistics

(Jazwinski, 1970).

Figure 5, adapted from a diagram contained in (Dutton, Thompson, & Barraclough, 1997: 483), shows a discrete time Kalman filter connected to a plant.
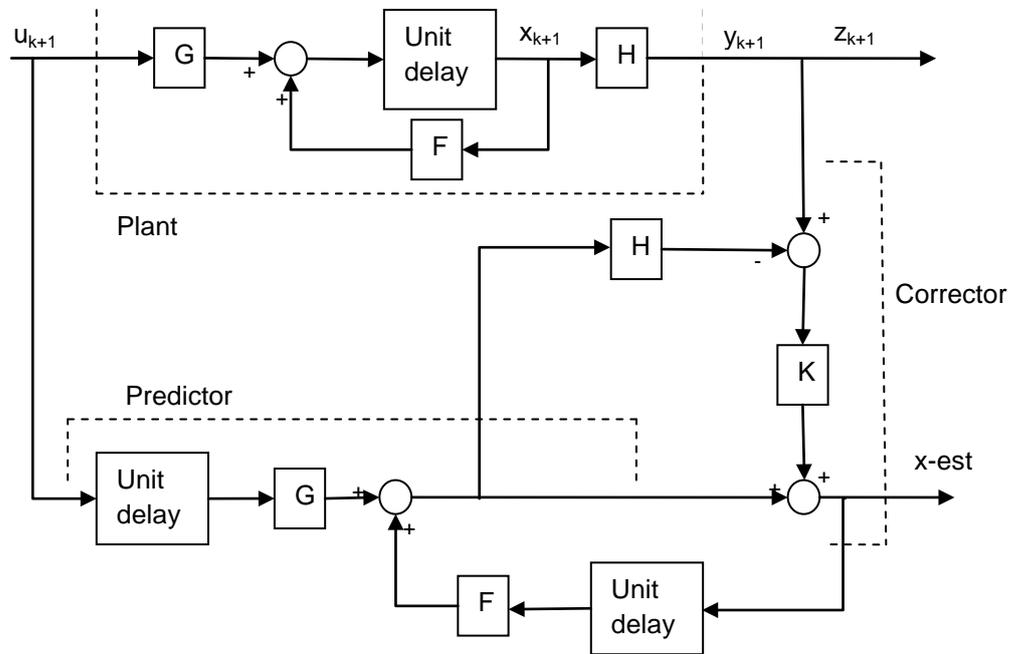
Figure 5 : Discrete-time Kalman filter

$$x(k) = Fx(k-1) + Gu(k-1) \qquad (5)$$
$$y(k) = Hx(k) \qquad (6)$$

where

       k = sample number

       x = state vector

       F = system matrix

       G = input matrix

       H = output matrix

       K = Kalman gain

       z = measurement vector

F, G and H are equivalent to A, B and C in Figure 3, but numerically different .

As explained earlier the Kalman filter is structurally identical to a Luenberger observer but is designed for a stochastic system containing noisy signals, but for simplicity the process and measurement noise have been omitted from the diagram.

The signal u is input into both the plant and Kalman filter. The Kalman filter contains a model of the plant which is used to make a prediction of the state values. The plant measurements z are also input into the Kalman filter and compared with the predicted values, and the Kalman gain K is applied to the difference and used to correct the state in the filter. Thus the Kalman filter undergoes a prediction-correction cycle: it uses the model data to make a prediction and then corrects it with the actual plant measurements.

The equations below, based on (Simon, 2006: p128), specify how the Kalman gain is calculated.

State estimate $\qquad \hat{x}_{k+1}^- = F_k \hat{x}_k + G_k u_k$ (7)

Error covariance $\qquad P_{k+1}^- = F_k P_k F_k^T + Q_k$ (8)

Kalman gain $\qquad K_{k+1} = P_{k+1}^- H_{k+1}^T (H_{k+1} P_{k+1}^- H_{k+1}^T + R_{k+1})^{-1}$ (9)

State estimate update $\qquad \hat{x}_{k+1} = \hat{x}_{k+1}^- + K_{k+1}(z_{k+1} - H_{k+1}\hat{x}_{k+1})$ (10)

Error covariance update $\qquad P_{k+1} = (I - K_{k+1}H_{k+1})P_{k+1}^-$ (11)

Where

$\hat{x}$ = state estimate

P = error covariance matrix

Q = system noise covariance matrix

R = measurement noise covariance matrix

The Kalman gain matrix must be chosen so that the estimates are optimal in some sense. The gain reflects the relative accuracy of the predicted states compared to the new observation. Low sensor noise (or high process noise) implies good measurements and / or large model uncertainty resulting in a high gain to correct the prediction. The opposite case (high sensor

13

noise or low process noise) implies poor measurements and / or small model uncertainty and therefore a lower gain is used since greater reliance is placed upon the accuracy of the model.

If the process noise w_k and measurement noise v_k are Gaussian, zero-mean, uncorrelated then the Kalman filter minimises (at each time step) the estimation error. If the process and measurement noise are not Gaussian but still zero-mean and uncorrelated then the Kalman filter is the optimal linear filter.

Figure 6 shows a simplified Kalman-Bucy filter which runs in continuous time. Its structure is identical except that the G in Figure 3 has been replaced by the stochastically determined Kalman gain K.
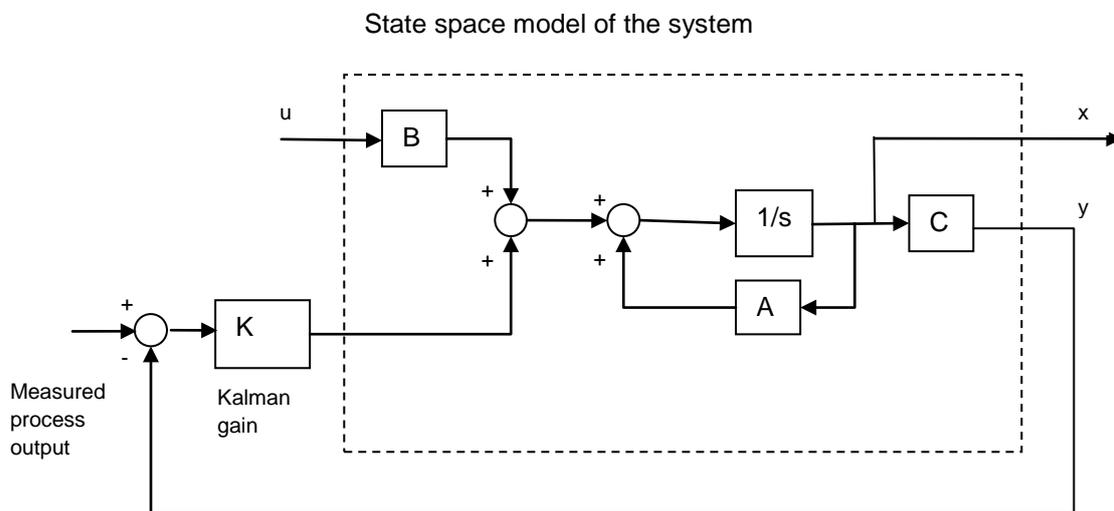
Figure 6: Continuous-time Kalman-Bucy filter

As before the inputs to the Kalman-Bucy filter are typically the plant inputs and measured outputs. The inputs u are used by the model of the plant contained in A, B and C to estimate the plant outputs. The estimated outputs are compared with the actual measured outputs and the difference is multiplied by the Kalman gain.

The equations for the continuous-time Kalman-Bucy filter (Simon, 2006: 236) are given below:

Kalman gain $\qquad K = PC^T R^{-1}$ (12)

State estimate update $\qquad \hat{x} = A\hat{x} + Bu^{\cdot} + K(y - C\hat{x})$ (13)

14

Error covariance update $\qquad \dot{P} = -PC^T R^{-1} CP + AP + PA^T + Q$ (14)

The Kalman-Bucy filter, like its discrete counterpart, is time varying but as noted in (Simon, 2006: 252) in some situations it converges to a linear time invariant filter and so a constant Kalman gain can be used.

The Kalman filter is widely used in many applications and it is impractical to detail all of its uses. It is noted in (Raol, Girija, & Singh, 2004: 66) that Kalman filtering has generated extensive application in aerospace system problems and thousands of papers have been written on Kalman filtering. For linear systems the Kalman filter is an optimal state observer for stochastically defined inputs / systems.

The evolution of the application of the Kalman filter in the aerospace industry is described in (Hutchinson, 1984). The use of the filter as a practical tool in aerospace is further described in (McGee & Schmidt, 1985).

A good practical introduction to the discrete-time Kalman Filter can be found in (Welch & Bishop, 2006).

This report will not show the derivation of the Kalman filter or Kalman-Bucy filter because the focus of the PhD is not to show the mathematics behind it but instead to show its application. There are many text books that do show the derivation of the filters and describe them in great detail, for example (Dutton, Thompson, & Barraclough, 1997: 791), (Lewis, 1986: 69) and (Grewal & Andrews, 2001: 116).

When non-linear systems are modelled the standard state estimation approach is to use an Extended Kalman Filter (EKF). According to (Simon, 2006: 396) the EKF "is undoubtedly the most widely used non-linear state estimation technique that has been applied in the past few decades". An EKF is essentially the same as a Kalman filter except that the filter is continually re-designed based upon a model that is re-linearised around the state estimate. The EKF has found use in applications such as navigation or GPS. Section 9 will describe the Extended Kalman Filter further and give an example of its implementation as a cg estimator.

2.2 Literature Review - Aircraft Centre of Gravity Estimation

The survey of aircraft centre of gravity calculation can be divided into two areas, one area will describe current methods used to obtain the cg when the aircraft is on the ground, and the

second area will describe techniques used, or under research, to estimate the cg when the aircraft is in-flight.

It is worth emphasising the importance of the weight and cg location to an aircraft. Weight and balance issues have caused numerous accidents, which primarily affect cargo flights but have still caused accidents in passenger flights. The paper by van Es (2007) provides a review of civil aircraft accidents related to weight and balance issues that have occurred since 1970, and reached the conclusion that a weight and balance related accident is 8.5 times more likely with a cargo flight compared to a passenger flight. The accidents that occurred in the period under study were caused by a variety of factors, such as errors in the load sheet, shifting of cargo, incorrect loading, etc. Van Es (2007) concluded that, although accidents due to weight and balance issues have nearly halved in the past 35 years, most of the problems could be resolved by an automatic weight and balance system. However van Es (2007) argues that the accuracy of current onboard weight and balance systems is insufficient to use them as the primary means for determining weight and balance.

In 2003 the NTSB (the US National Transportation Safety Board) recommended research into systems to give weight and balance data before flight dispatch. Following an accident in Benin of a Boeing 727 the French investigative body, the BEA (BEA, 2003), recommended that all new commercial aircraft have onboard weight and balance systems and that regulations are put in place to retrofit these systems where possible.

According to the FAA (2005) "an operator may use an onboard system to measure an aircraft weight and balance as the primary means to dispatch an aircraft provided the FAA has certified the system". The advisory circular (FAA, 2008) describes how to obtain FAA approval for an onboard weight and balance system. According to van Els (2007) the "specifications drafted for onboard weight and balance systems state that the system shall be capable of measuring the gross weight within an accuracy of 1% and the aircraft centre of gravity within 1% of the mean aerodynamic chord".

To clarify, the onboard weight and balance systems discussed in the preceding paragraphs refer to on-ground systems.

2.2.1 Aircraft Centre of Gravity Calculation on Ground

This sub-section describes the current method used to weigh an aircraft and calculate its

16

centre of gravity position, and is primarily sourced from Federal Aviation Administration (2007), a 97 page document to provide the aircraft mechanic with the method to determine the weight and centre of gravity of an empty aircraft, and also to give the aircrew information about the safe loading of aircraft.

The aircraft weight and centre of gravity is computed when the aircraft is on the ground. The aircraft manufacturer will specify the empty weight of the aircraft and its empty weight centre of gravity location. The weight in a loaded aircraft must be distributed to keep the cg within specified limits, therefore as weight is added a record is kept of the addition of new equipment, cargo etc.

The aircraft weight may be calculated on the ground by positioning the aircraft on scales. The weight of any device holding the aircraft in position needs to be subtracted and the aircraft must be in its level flight attitude. The arms (distance between the cg of the item and a datum point) of the weighing points are all specified for the aircraft. The moment is the weight multiplied by the arm and the centre of gravity is calculated as :

$$cg = total\ moment\ /\ total\ weight \tag{15}$$

The cg position is relative to the datum point.

It is also possible to calculate the change in cg when weight is shifted with :

$$\Delta cg = weight\ shifted \cdot dis\tan ce\ weight\ shifted\ /\ total\ weight \tag{16}$$

The FAA issue a Type Certificate Data Sheet for all aircraft approved by them, and this certificate will specify the safe cg range.

Example Piper PA-28-160 Cherokee

cg range (+84" to +95.9" from datum) at 1650 lb or less

(+85.9" to +95.9" from datum) at 1975 lb

(+88.2" to +95.9" from datum) at 2200 lb

maximum weight 2200 lb

datum is 78.4" forward of wing leading edge

If an aircraft mechanic adds or removes any equipment to the aircraft then the FAA dictate

that the weight and balance sheet for the aircraft must be updated.

Some large transport aircraft have on board weighing systems that, when the aircraft is on the ground, give the aircrew a continuous indication of the total aircraft weight and cg location as a percentage of the mean aerodynamic chord (mac). The system consists of strain-sensing transducers in each main wheel and nose wheel axle, a weight and balance computer, and indicators that show the gross weight, the cg location expressed as a percentage of mac.

During preflight planning the pilot must ensure that the weight and cg are within allowable limits and completes a weight and balance loading form.

2.2.2 Aircraft Centre of Gravity Estimation in Flight

The standard method to estimate the aircraft centre of gravity when the aircraft is in flight is to divide the aircraft into separate nodes with each node having its own weight and centre of gravity location. One node will specify the initial empty aircraft weight and cg location and the other nodes will be defined for the different fuel tanks or ballast. Brockman (1980) describes this approach using 18 nodes, and also describes how fuel temperature and fuel weight is used to calculate consumed fuel in each tank. The effect of flap and landing gear positions is accommodated by shifting the position of the empty aircraft weight node in the calculation. The weights are summed to obtain the total weight, the total moment is calculated and from this the centre of gravity is computed.

However according to Glover (1985) the estimation of cg through the fuel burn approach is "subject to significant errors arising out of inaccuracies in fuel burn computations and shifts in passenger and cargo locations as well as errors that may be present in the initial measurement of the center of gravity on the ground".

Blakely and Hedges (1998) describes an improvement of the approach in Brockman (1980). It adopts the same general methodology, separating the aircraft into nodes, and improves it by describing how more modern technology is exploited. In addition to this it also incorporates fuel sloshing into the calculations by including a fuel slosh time lag based upon the aircraft pitch rate.

An alternative method to estimate the cg is to use accelerometers. A patent by Sundstrand Corporation (Glover, 1985) describes a technique using two accelerometers and signal

processing to estimate the aircraft centre of gravity. The accelerometers are placed near the front and rear of the aircraft to produce signals representing the rotation of the aircraft through the cg. The accelerometer signals are combined into a signal representing both motion of the aircraft and cg, and a signal representing the motion of the aircraft. The two signals are then combined to get a signal representing the cg.

Two patents by the Boeing Company (Orgun & Flanagan, 1991) and then (Wu, 1996) describe the apparatus for the real-time estimation of aircraft centre of gravity. The first patent describes a technique to estimate cg based upon the aircraft angle of attack, flap setting and stabilizer position. The second patent is an improvement on the first by adding additional factors into the approximation including aircraft weight, the expected load factor on the aircraft, dynamic pressure on the aircraft and the reference wing area of the aircraft. When the aircraft descends through a pre-specified altitude the cg estimate is stored in memory for use during the landing phase. The patent (Wu, 1996) claims that the cg estimate "falls within 1% of mac of the measured graph over a large angle of attack range".

More recent attempts at cg estimation have focused upon using neural nets. A paper by Idan et al (2004) describes a neural net which is trained to estimate the weight and longitudinal cg of an aircraft when the aircraft is in trimmed flight, i.e. it is climbing, cruising or descending. The paper provides the theory to show that the longitudinal cg is a function of the dimensionless total pitch couple C_M and normal force C_Z and these are in turn functions of the Mach number, angle of attack α, elevator deflection δ_e, flight path angle γ and normal cg z_{cg}. It then describes how a neural net was trained to estimate the weight and cg of an aircraft using an aircraft model of a small business jet. The neural net was trained using approximately 4000 test points. Zero-mean Gaussian noise was added to the test data with standard deviations of 300 lb and 1.5% mac. The paper claims the neural net estimated the cg with an accuracy of 1.6% mac, and 99% of the points were accurate within 1% mac.

The approach taken by Idan et al (2004) was extended by Zhang et al (2009) into a data fusion approach. Zhang et al (2009) is concerned with designing an attitude controller to provide consistent performance under varying cg. It varies the cg in an aircraft model and then estimates the cg change using an adaptive weighted data fusion technique. The data fusion technique uses cg estimates from the traditional approach of estimating cg from an estimate of changes in aircraft weight, and also a cg estimate based upon the Idan et al (2004)

neural net approach. The paper does not detail the accuracy or effectiveness of this data fusion approach to estimate the cg.

Another neural net approach is described by Bi et al (2004). This paper used a neural net to estimate the gross weight and cg of an in-flight V-22 tilt-rotor aircraft. The test data used to develop the neural net were obtained from 14 tests consisting of 321 different flight "runs" to train the model, and each run lasted between 6 and 30 seconds and represented "a nominally steady flight condition at a given airspeed, gross weight and flight altitude". During the tests the cg varied with a 4.5 inch range. The authors of this paper claim the cg estimation had a root mean square (rms) error of 0.17 inches (4.32mm), and 95% of the points had an estimation error less than 0.35 inches (8.89mm).

Recent research into helicopter cg estimation by Abraham and Costello (2009) has used an Extended Kalman Filter (EKF) which is constructed with the rigid state of the helicopter and augmented with the weight and the 3 components of the cg (x, y and z axis). A different approach proposed by Cummins et al (2009) suggests that accelerometers could be used to measure the small changes in natural frequencies associated with the pitch-heave mode of vibration which would change as a result of a cg change.

2.3 Flight Dynamic Equations

This section introduces the aircraft flight dynamic equations and some terminology that will be used later in this thesis. The equations of motion for a rigid aircraft cover the six degrees of freedom and are non-linear.

The aerodynamic forces and moments which affect an aircraft depend upon its orientation with respect to the airflow. These angles are the angle of attack (α) and sideslip angle (β).

The coordinate system is defined such that the body x-axis is parallel to the fuselage, the y-axis is at 90 to starboard and the z-axis is 90 "downwards". A right-hand rule is applied to the values, for example a positive y value is to starboard and a positive z value is "downwards".

The following set of equations were obtained from (Stevens & Lewis, 2003) and (Forsell & Nilsson, 2005).

Velocity is:

$$\dot{V}_T = \frac{u_b \cdot \dot{u}_b + v_b \cdot \dot{v}_b + w_b \cdot \dot{w}_b}{V_T}$$

<div align="right">(17)</div>

Where the force equations are given by:

$$\dot{u}_b = r_b \cdot v_b - q_b \cdot w_b - g \cdot sin\theta + F_x/\text{mass} \qquad (18)$$

$$\dot{v}_b = -r_b \cdot u_b + p_b \cdot w_b + g \cdot sin\phi \cdot cos\theta + F_y/mass \qquad (19)$$

$$\dot{w}_b = q_b \cdot u_b - p_b \cdot v_b + g \cdot cos\phi \cdot cos\theta + F_z/mass \qquad (20)$$

The symbols F_x, F_y and F_z identify the forces acting on the aircraft in the x, y and z axes.

The angle of attack is calculated from:

$$\dot{\alpha} = \frac{u_b \cdot \dot{w}_b - w_b \cdot \dot{u}_b}{u_b^2 + w_b^2}$$

<div align="right">(21)</div>

The sideslip angle is calculated from:

$$\dot{\beta} = \frac{\dot{v}_b \cdot V_T - V_b \cdot \dot{V}_T}{V_T^2 \cdot cos\beta}$$

<div align="right">(22)</div>

The roll rate is:

$$\dot{p}_b = \frac{I_{xz}(I_x - I_y + I_z) \cdot p_b \cdot q_b - \left[I_z(I_z - I_y) + I_{xz}^2\right] \cdot q_b \cdot r_b + I_z \cdot M_x + I_{xz} \cdot M_z}{I_x \cdot I_z - I_{xz}^2}$$

<div align="right">(23)</div>

The pitch rate is:

$$\dot{q}_b = \frac{(I_z - I_x) \cdot p_b \cdot r_b - I_{xz}(p_b^2 - r_b^2) + M_y}{I_y}$$

The yaw rate is:

$$\dot{r}_b = \frac{\left[(I_x - I_y)I_x + I_{xz}^2\right] \cdot p_b \cdot q_r - I_{xz}(I_x - I_y + I_z) \cdot q_b \cdot r_b + I_{xz} \cdot M_x + I_x \cdot M_z}{I_x \cdot I_z - I_{xz}^2}$$

<div align="right">(25)</div>

These angular accelerations are dependent upon various moments of inertia and moments.

The moments are largely dependent upon the aircraft speed and angle of attack or sideslip angle. The main effect of aircraft speed is to change the dynamic pressure $\bar{q}$.

$$\bar{q} = {\rho \cdot V_T^2}/{2} \tag{26}$$

The angle of attack α mainly affects the pitching moment coefficient C_m whereas the sideslip angle β is equally important for the other moments. A generalisation for the angle of attack α is that as it increases then the pitching moment coefficient C_m will also increase.

$$M_x = \bar{q} \cdot S \cdot b \cdot C_n + z_{cg} \cdot F_y + y_{cg} \cdot F_z \tag{27}$$

$$M_y = \bar{q} \cdot S \cdot b \cdot C_m - x_{cg} \cdot F_z + z_{cg} \cdot F_x \tag{28}$$

$$M_z = \bar{q} \cdot S \cdot b \cdot C_l + x_{cg} \cdot F_y - y_{cg} \cdot F_x \tag{29}$$

From these equations it can be seen that changes in the centre of gravity (x_{cg}) affects the moments which in turn alter the aircraft angular acceleration.

There are also equations for the orientation of the aircraft.

A changes in the yaw angle is calculated by:

$$\dot{\psi} = (q_b \cdot sin\phi + r_b \cdot cos\phi)/cos\theta \tag{30}$$

A change in the pitch angle is calculated by:

$$\dot{\theta} = q_b \cdot cos\phi - r_b \cdot sin\phi \tag{31}$$

Finally, a change in the bank angle is calculated by:

$$\dot{\phi} = p_b + tan\theta \cdot (q_b \cdot sin\phi + r_b \cdot cos\phi) \tag{32}$$

The final set of equations define the aircraft co-ordinates in the 3 axes.

$$\dot{x}_v = cos\theta \cdot cos\varphi \cdot u_b + (sin\phi \cdot sin\theta \cdot cos\varphi - cos\phi \cdot sin\varphi) \cdot v_b + (cos\phi \cdot sin\theta \cdot cos\varphi \\ + sin\phi \cdot sin\varphi) \cdot w_b$$

$$\tag{33}$$

$$\dot{y}_v = cos\theta \cdot sin\varphi \cdot u_b + (sin\phi \cdot sin\theta \cdot sin\varphi + cos\phi \cdot cos\varphi) \cdot v_b + (cos\phi \cdot sin\theta \cdot sin\varphi \\ - sin\phi \cdot cos\varphi) \cdot w_b$$

$$\tag{34}$$

$$\dot{z}_v = -sin\theta \cdot u_b + sin\phi \cdot cos\theta \cdot v_b + cos\phi \cdot cos\theta \cdot w_b \tag{35}$$

The set of complex non-linear equations in this sub-section defines the aircraft flight dynamic behaviour.

To analyse the aircraft behaviour it is common practice to obtained linearised equations for the aircraft, and these linearised equations are obtained when the aircraft is at a known steady state.

If the aircraft is flying in a trimmed state then the aircraft is at an equilibrium point with zero acceleration.

$$\dot{p}, \dot{q}, \dot{r}, \dot{u}, \dot{v}, \dot{w} = 0 \tag{36}$$

In this thesis the ADMIRE 'trim' routines have been used to obtain an acceptable equilibrium point. The aircraft speed, altitude and angle of attack (optional) must be provided and then the trim routine executes a minimisation algorithm routine to find an acceptable equilibrium point. The trim routine provides the initial inputs into the non-linear aircraft model and the initial outputs from the aircraft model. If the aircraft is then linearised at the equilibrium point then the inputs into and the outputs from the linear model are delta values from the initial conditions.

A small perturbation from the equilibrium point is used to derive a set of linear constant-coefficient state equations called Jacobian matrices. There is a description in Cook (2007) and Stevens and Lewis (2006: section 2.6) regarding the algebraic derivation of the linear

equations. It is noted in Stevens and Lewis (2006: section 3.7) that algebraic linearisation was only "tractable under the restrictions of wings-level, non-sideslipping steady state flight" and it then goes on to describe a numerical linearisation approach. In this thesis it is necessary to generate linearised equations when the aircraft does not meet the restrictions described by Stevens and Lewis (2006). The matlab routine 'linmod' has been used to obtain the matrices of the linear aircraft model by numerical linearisation.

2.4 Summary

A limitation of the literature review is a lack of data regarding the current accuracy of cg estimates obtained by the FCS. Brockman (1980), Blakely & Hedges (1998), Glover (1985) and Zhang et al (2009) give no data for the accuracy of their approaches.

Bi et al (2004) does quote an rms error of 0.17 inches (4.32 mm) for the cg estimate, with 95% of points having an estimation error less than 0.35 inches (8.89 mm). However Bi et al (2004) does not give details of the aircraft manoeuvres in which the estimation occurred, except than the data used was obtained when the aircraft was in "a nominally stable flight condition", so it is presumed that the aircraft was not manoeuvring.

Idan et al (2004) quotes a cg estimation accuracy of 1.6% mac with 99% of points accurate within 1% mac, however the paper is only concerned with estimating the cg of a trimmed aircraft.

Wu (1996) claims an accuracy within 1% mac but the estimator is only used the specific scenario when the aircraft is in a landing phase.

The literature survey has shown papers affecting various types of aircraft: commercial passenger aircraft (Wu, 1996), tilt-rotor aircraft (Bi et al, 2004), helicopters (Abrahams & Costello, 2009) (Cummins et al, 2009), business jets (Idan et al, 2004) and it is difficult to compare the results since they use such different aircraft. The papers that do provide results use different terminology to specify them, for example (Bi et al, 2004) use the absolute measurement in inches, whereas (Idan et al, 2004) uses % mac. In addition to this another problem is the lack of precise details of the manoeuvres performed by the aircraft during the estimation.

From this survey it would seem reasonable to conclude that any future in-flight cg estimator

should aim to have an accuracy of at least 1% mac to be no worse than existing techniques such as Wu (1996), Idan et al (2004) and also to satisfy the 1% mac accuracy requirement identified in van Els (2007) for future automatic on-ground cg estimation.

There is no literature that explicitly investigates the use of Kalman filters to estimate the aircraft centre of gravity, though Abraham and Costello (2009) did use an EKF on a helicopter with encouraging results.

In this thesis it is proposed that, unlike the neural net approach, greater use is made of knowledge of the aircraft equations of motion and aircraft parameters to derive the cg estimate. Since the process is characterised by linear differential equations with process and measurement noise then Kalman filters will be used. A particular feature of this research is the application of defined aircraft manoeuvres and their effect upon the cg estimate, and the application of a coefficient correction technique to maintain accurate cg estimates across the manoeuvre range and within a specified speed, altitude and angle of attack envelope.

Fuel sloshing was identified in section 2.1.1 as a limiting factor in future aircraft design. To help test the effect of fuel sloshing upon the cg estimate a fuel sloshing model is provided by BAE Systems and is included in tests performed upon the estimator in section 7.

3. Methodology

This section specifies the requirements placed upon an in-flight cg estimator made by the project sponsors, BAE Systems. It also describes the theory which relates unexpected angular acceleration to a change in the centre of gravity position, and describes how a Kalman-Bucy filter can be augmented to obtain the unexpected angular acceleration which is then converted into a cg estimate.

3.1 Requirements

The research in this thesis was sponsored by EPSRC and BAE Systems. BAE Systems provided the performance and manoeuvre requirements for a cg estimator. The performance requirements specify an acceptable level of estimation error whereas the manoeuvre requirements specify the operational conditions under which the estimator should work correctly.

Performance requirements

1. Output updated at 10Hz
2. Maximum static error 0.3% mean aerodynamic chord (mac)
3. Maximum dynamic error 0.5% mac
4. Static error accuracy within 1 second
5. Transient exceedance of static error allowed as long as it does not exceed 20% of the static error level, the static performance is recovered within 1 second, no further exceedance of static performance requirement occur whilst the system remains undisturbed.

Manoeuvre requirements

1. Maximum roll rate +/-30 deg/s
2. Maximum lateral acceleration +/- 1.5m/s^2
3. Maximum speed acceleration +/- 0.03 Mach/s
4. Maximum pitch rate obtained from full pull up or push down command in 5 seconds

Subsequent discussions with BAE Systems further refined the scenario in which the cg estimator should operate. The cg estimator should operate in the subsonic region, up to approximately Mach 0.8, and the maximum bank angle should be +/-45°.

The accuracy requirements for this project are more restrictive than any that were specified in the literature survey. Wu (1996) obtained an accuracy of 1% mac when a large passenger aircraft was in its landing phase. Idan et al (2004) obtained an accuracy under 1% mac for 99% of points using a business jet model in trimmed flight.

3.2 Theory to relate unexpected angular acceleration with centre of gravity change

This section develops the theory which is used to estimate the longitudinal centre of gravity of an aircraft. The theory covers the scenario when the aircraft is trimmed in horizontal straight and level flight and relates pitch acceleration to a change in the centre of gravity.

Any deviation from trimmed flight is caused by an unexpected moment causing the aircraft to pitch up or down. The unexpected moment may be caused by forces external to the aircraft e.g. a strong gust of wind, but for now the unexpected moment is assumed to be caused by a movement in the longitudinal cg. The deviation from level flight will be trimmed by the FCS adjusting the control surfaces. Therefore in straight and level flight the longitudinal cg is related to the pitch angular acceleration (which is the unexpected angular acceleration) before the aircraft is trimmed, and after the trim the cg is related to the changed control commands which is again an unexpected angular acceleration.

The relationship between the change in longitudinal centre of gravity and angular acceleration is analysed in the next two sub-sections, first for a beam and then for a more representative aircraft.

3.2.1. Horizontal Beam with Two Lift Forces

Assume there is a horizontal beam supported by two lift forces L_w and L_t, and the matching opposing weight is through its centre of gravity (see Figure 7).

L_w = main wing / fuselage lift

L_t = tailplane lift

l_w = distance of L_w from cg

l_t = distance of L_t from cg

W = aircraft weight through cg
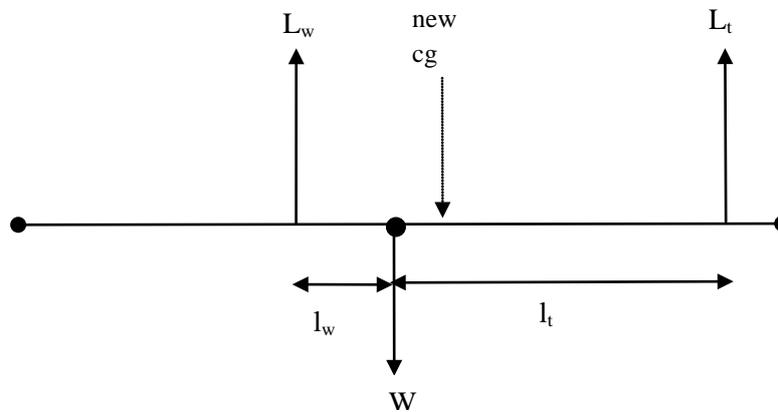
Figure 7: Horizontal beam

In this example the total lift force $L_w + L_t = W$, and in the trimmed state there are no net moments about the centre of gravity, therefore $L_w l_w - L_t l_t = 0$.

Consider the scenario where the cg moves by Δcg to create a pitching moment M, then

$$M = L_w(l_w + \Delta cg) - L_t(l_t - \Delta cg) \tag{37}$$

Equation 37 can be rearranged to obtain the change in cg:

$$\Delta cg = (M - L_w l_w + L_t l_t)/(L_w + L_t) \tag{38}$$

Since the beam was initially trimmed then $L_w l_w - L_t l_t$, and $L_t l_t - L_w l_w$, must equal zero, therefore

$$\Delta cg = {M}/{(L_w + L_t)} \tag{39}$$

Since L_w plus L_t equals the weight of the aircraft then an alternative formulation of the equation is :

$$\Delta cg = {M}/{W} \tag{40}$$

3.2.2 Horizontal Trimmed Aircraft

The theory for a horizontal beam is now extended to a traditional aircraft with a main wing and rear elevator, see Figure 8. The diagram and subsequent trim equation are based on data in (Cook, 2007: 41). This example extends the beam example by introducing moments for the wings and using standard aircraft terminology for the different forces and measurements used.
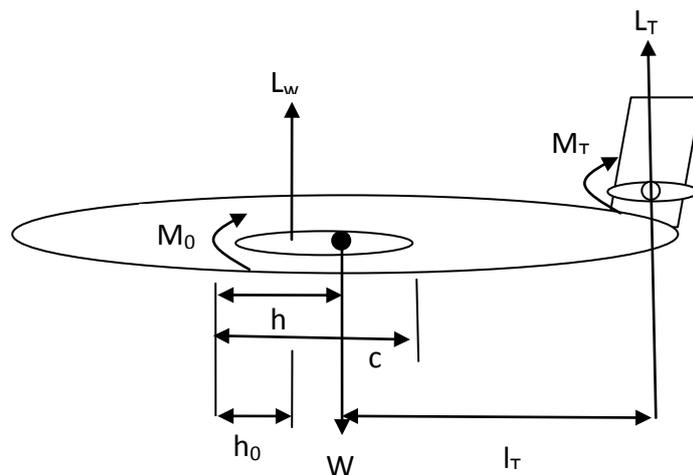
Figure 8: Trimmed aircraft in level flight

The aircraft terminology is included here for ease of reference, and is also included in the List of Symbols.

c = mean aerodynamic chord

W = aircraft weight through centre of gravity

L_w = lift force through wing

L_T = lift force through tailplane

M_0 = wing moment about aerodynamic centre

M_T = tailplane pitching moment about tailplane aerodynamic centre

h = longitudinal cg position measured from leading edge wing root

h_0 = aerodynamic centre position on reference chord

The total pitching moment M about the cg (taken from (Cook, 2007: 42)) is :

$$M = M_0 + L_w(h - h_0)c - L_T l_T + M_T \qquad (41)$$

Assume both M_0 and h_0 are constants. According to (Cook, 2007: 42) the M_0 moment is constant for subsonic flight. Also assume that the tailplane aerofoil section is symmetric so M_T is zero.

In a trimmed aircraft the pitching moment is zero therefore :

$$M_0 + L_w(h - h_0)c - L_T l_T = 0 \qquad (42)$$

This can be re-arranged to give :

$$hc = (L_w h_0 c + L_T l_T - M_0)\big/_{L_w} \qquad (43)$$

The main wing lift and centre of gravity locations are defined in terms of the mean aerodynamic chord 'c'. It is helpful if the location of the tailplane lift is also defined in terms of 'c' instead of the distance l_T.

The value of l_T depends upon the value of h, the centre of gravity position, and in the non-manoeuvring stable flight condition it can be assumed that the distance from L_T to h_0 is fixed therefore :

$$L_T l_T = L_T(D - (h - h_0)c \qquad (44)$$

where D = distance from L_T to h_0

Substituting this value for $L_T l_T$ into the previous equation and then re-arranging gives :

$$hc = \frac{L_w h_0 c + L_T h_0 c + L_T D - M_0}{L_w + L_T}$$

$$(45)$$

This defines the cg location for a trimmed horizontal aircraft.

Next consider when the aircraft is stable and trimmed but encounters a pitching moment M.

$$M = M_0 + L_w(h - h_0) - L_T l_T \qquad (46)$$

Adding in the assumption for l_T then :

$$M = M_0 + L_w(h - h_0)c - L_T(D - (h - h_0)c) \tag{47}$$

Re-arranging the equation gives :

$$hc = \left. \frac{M - M_0 + L_w h_0 c + L_T h_0 c + L_T D}{L_w + L_T} \right. \tag{48}$$

Assuming all the lifts etc are unchanged, then a comparison of (45) with (48) shows that 'h', the longitudinal centre of gravity, has moved further aft by $M/(L_w + L_T)$

i.e. $\mathbf{\Delta cg = M_{\,/}(L_w + L_T)}$

which is identical to (39).

This analysis has shown that for a trimmed horizontal aircraft the change in the longitudinal centre of gravity can be calculated if the pitching moment and total lift or aircraft weight are known.

A dependency should also exist between the cg and amount of control surfaces required to trim the aircraft. To trim the aircraft the control surfaces must cancel out any net pitching moment. For example if the aircraft is configured for trimmed flight but is pitching up at 0.005 rad/sec^2 with a pitching moment of inertia of 20000 kg m^2 then the net pitching moment is 100 Nm (calculated using $M = I_y \dot{q}$).

To trim the aircraft its control surfaces must generate an opposing moment of 100 Nm through L_w, and since this moment depends upon the altered lift L_w and the distance between L_w and cg then the longitudinal cg position can be estimated in the steady-state by the amount of trim.

3.2.3 Angular Acceleration

The previous two sections have shown the relationship between a change in the centre of gravity and the pitching moment. The final piece of theory will show the relationship between the moment and angular acceleration.

Given Newton's Second Law adapted for angular acceleration :

$$t = I\alpha \tag{49}$$

where t = torque

      I = moment of inertia

      α = angular acceleration

The aircraft pitching moment $M = I_y \dot{q}$.

Where I_y = pitching moment of inertia

      $\dot{q}$ = pitch acceleration

It was shown in the last section that $\Delta cg = M/W$

Therefore $\Delta cg = I_y \dot{q}/W$, or to be more precise since weight is the downward force which is mass multiplied by gravity, then

$$\Delta cg = {I_y \dot{q}_{unex}}\big/{mass \cdot g} \tag{50}$$

3.3 State augmentation to obtain unexpected angular acceleration

It is assumed that the unexpected changes in angular acceleration result from unexpected moments about the centre of gravity location and therefore, providing that all possible causes of the moment have been allowed for in the Kalman-Bucy filter, then there is a relationship between the unexpected angular acceleration and a cg change.

A top level view of this approach, for an aircraft model with a single elevator deflection command, is given in Figure 9.
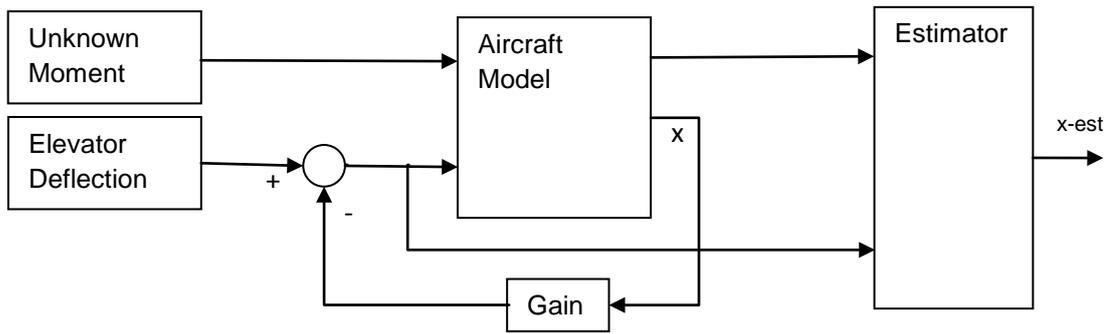
Figure 9: Aircraft model with estimator

The estimator inputs the commands into the aircraft model and uses these to predict the aircraft state, i.e. the predicted pitch rate. It also inputs the actual measurements from the aircraft to correct its prediction. Therefore the estimator has the standard prediction-correction structure described in the previous section. Since the real aircraft measurements will be subject to noise then it is proposed that the estimator will be a Kalman-Bucy filter to process the noisy aircraft measurements in an optimal way.

The estimator is required to estimate the unexpected angular acceleration, that is the difference between the expected angular acceleration in the estimator model and the actual angular acceleration. The problem is therefore one of structuring the Kalman-Bucy filter to best estimate this unexpected acceleration.

A number of different approaches were investigated to estimate the unexpected pitch acceleration $\dot{q}$, for example differentiating the measured value of q to add $\dot{q}$ as a measurement. This approach provided reliable estimates of unexpected pitch acceleration but when noise was added to the measurements it was very poor, because there was no model for $\dot{q}$ to use in preference to the noisy signal.

The adopted approach was to augment the Kaman-Bucy filter with the unexpected angular acceleration and rely upon the correction process in the Kalman filter to obtain accurate state values. This is explained in more detail below.

Let 'x' contain the aircraft state and it is augmented with the unexpected pitch acceleration $\dot{q}_{unex}$ to give x˜.

33

$$x^\sim = [x, \dot{q}_{unex}] \tag{51}$$

The state-space equation in (52) illustrates how $\dot{q}_{unex}$ is obtained. All states, apart from q, retain their original state-space A and B matrix values as shown by the '$\cdots$' to denote no change.

$$\begin{bmatrix} \dot{x}' \\ \dot{q}_{act} \\ \ddot{q}_{unex} \end{bmatrix} = \begin{bmatrix} \cdots & \cdots & 0 \\ \cdots & \cdots & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x' \\ q_{act} \\ \dot{q}_{unex} \end{bmatrix} + \begin{bmatrix} \cdots \\ \cdots \\ 0 \end{bmatrix} u \tag{52}$$

The variable q has been renamed to q_{act} because it is now the actual value of q calculated from the estimated value for $\dot{q}$ plus any unexpected $\dot{q}$.

This method of modelling the unknown acceleration is adapted from an example contained in Friedland (1986: 421), in which unknown acceleration is modelled as a random process.

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{53}$$

Equation (53) shows that the measurements for x (including q) are output but that there is no measurement for $\dot{q}_{unex}$. The additional entry in the row of the state matrix for q provides the link to $\dot{q}_{unex}$ which ensures that the new augmented state $x^\sim$ is fully observable, and this is demonstrated numerically in section 4.2 in an example with a Phantom linear aircraft model.

The Kalman gain in the filter is obtained by using the Matlab 'lqe' function on the augmented state $x^\sim$. The filter is used to obtain $\dot{q}_{unex}$ which is then scaled by the moment of inertia I_y and divided by mass, as defined in (50) to obtain the change in cg.

The remaining sections in this thesis will use this approach of augmenting the Kalman-Bucy filter to estimate the unexpected angular acceleration, which is then converted into an estimate of the change in the centre of gravity.

3.4 Summary

This section of the thesis has defined the requirements of the estimator, the relationship between angular acceleration and a change in cg, and it has also described the state augmentation approach to be used to estimate unexpected angular acceleration.

4. Application of a Centre of Gravity Estimator to a Linear Aircraft Model

This section describes the development and design of a centre of gravity estimator applied to a linear aircraft model. It applies the state augmentation approach described in the previous section to a Kalman-Bucy filter to estimate unexpected changes in angular acceleration.

It is assumed that the unexpected angular acceleration caused by an unexpected moment about the centre of gravity is caused by a change in the cg location, therefore all possible causes of a change in acceleration are accurately modelled in the Kalman-Bucy filter. The relationship between the unexpected angular acceleration and a cg change was defined in section 3.

A top level view of this approach was contained in Figure 9.

4.1 Aircraft Model

A McDonnell Douglas F4-C Phantom aircraft model is used to estimate changes in the centre of gravity. The model details were obtained from Cook (2007: 89). To allow cg changes the aircraft model is changed slightly.

The state space equation for the McDonnell F4-C Phantom aircraft model is shown in (54).

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0.007181 & 0.00457 & -29.072 & -9.678 \\ -0.0687 & -0.2953 & 174.868 & 1.601 \\ 0.00173 & -0.0105 & -0.4462 & 0.001277 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} 1.041 \\ -6.294 \\ -4.888 \\ 0 \end{bmatrix} [\delta e]$$

(54)

where

u = axial velocity

w = normal velocity

q = pitch rate

θ = pitch angle

δe = elevator deflection

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{55}$$

$$D = [0; 0; 0; 0] \tag{56}$$

Full details of the flight conditions and dimensionless longitudinal derivatives are given in Example 4.2 in Cook (2007: 80). The data required to convert the unexpected pitch acceleration $\dot{q}$ into the change in the centre of gravity is the mass of the aircraft and pitching moment of inertia. These are:

mass = 17642 kg

pitch moment of inertia I_y = 165669 kg m^2

The inputs into the aircraft model are the elevator deflection and unexpected moment. The B and D matrices in the Phantom model are amended to accept the unexpected moment as input.

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0.007181 & 0.00457 & -29.072 & -9.678 \\ -0.0687 & -0.2953 & 174.868 & 1.601 \\ 0.00173 & -0.0105 & -0.4462 & 0.001277 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} 1.041 & 0 \\ -6.294 & 0 \\ -4.888 & 6.036e-6 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta e & M_{unex} \end{bmatrix} \tag{57}$$

Equation (57) shows the modified aircraft model state-space equation. The unexpected moment M_{unex} which would be generated by a change in cg is injected into the aircraft model. This unexpected moment M_{unex} is divided by I_y in (57) to affect the pitch acceleration $\dot{q}$.

4.2 Estimation Model

The aircraft model equations are augmented to estimate the change in the centre of gravity. To obtain the change in the centre of gravity equation (50) is used to obtain the additional entry in the state-space model by converting Δcg to the pitch acceleration change.

Equation (50) is reordered to give the unexpected pitch acceleration :

$$\dot{q}_{unex} = \frac{\Delta cg \cdot mass \cdot g}{I_y}$$

(58)

Therefore the conversion factor to convert the change in cg to $\dot{q}_{unex}$ is :

$$\dot{q}_{unex} = \frac{\Delta cg \cdot 17642 \cdot 9.81}{165669}$$

$$\dot{q}_{unex} = 1.0446146 \Delta cg$$

This conversion factor is then used in the augmented state-space model to obtain a state estimate of the cg change.

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q}_{act} \\ \dot{\theta} \\ \Delta c\dot{g} \end{bmatrix} = \begin{bmatrix} 0.007181 & 0.00457 & -29.072 & -9.678 & 0 \\ -0.0687 & -0.2953 & 174.868 & -1.601 & 0 \\ 0.00173 & -0.0105 & -0.4462 & 0.001277 & 1.04466146 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q_{act} \\ \theta \\ \Delta cg \end{bmatrix} + \begin{bmatrix} 1.041 \\ -6.294 \\ -4.888 \\ 0 \\ 0 \end{bmatrix} [\delta e]$$

(59)

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

(60)

$$D = [0; 0; 0; 0]$$

(61)

The observability of the system was checked using the Matlab command 'obsv', which returns the observability matrix [C; CA; CA^2 ...], see (62).

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0.000718 & 0.0457 & -29.072 & -9.678 & 0 \\
-0.0687 & -0.2953 & 174.868 & -1.601 & 0 \\
0.00173 & -0.0105 & -0.4462 & 0.00128 & 1.0447 \\
0 & 0 & 1 & 0 & 0 \\
-0.0506 & 0.3039 & 4.0722 & -0.0514 & -30.37 \\
0.3228 & -1.7492 & -129.27 & 1.361 & 182.68 \\
-0.00005 & 0.0078 & -1.686 & -0.0005 & -0.4661 \\
0.0017 & -0.0105 & -0.4462 & 0.00128 & 1.0447 \\
-0.0139 & -0.1327 & 52.75 & 0.0084 & 4.254 \\
-0.1032 & 1.875 & -256.23 & -0.4882 & -135.04 \\
-0.0035 & 0.0154 & 2.116 & -0.0142 & -1.7613 \\
-0.00005 & 0.0078 & -1.686 & -0.0005 & -0.4661 \\
0.1004 & -0.5147 & -46.335 & 0.414 & 55.103 \\
-0.5722 & 2.1361 & 444.78 & -2.3306 & 267.67 \\
0.0026 & -0.0268 & 1.835 & 0.0115 & 2.2106 \\
-0.0035 & 0.0154 & 2.1161 & -0.0142 & -1.761
\end{bmatrix}
\qquad (62)
$$

The rank of the filter was then obtained using the Matlab 'rank' command, and a rank of 5 was obtained meaning that the system was fully observable.

The Kalman gain in the estimator was calculated by using the Matlab 'lqe' function.

4.3 Tuning the Kalman-Bucy Filter

The estimated maximum and average values of the aircraft measurements were used as initial values to develop and tune the Kalman-Bucy filter. The data in Table 1 specifies these expected maximum and average values, and also the expected measurement noise.

Table 1: Phantom Model Kalman-Bucy Filter tuning data

	u	w	q	θ
Max value	100 m/s	30 m/s	30°/s (0.524 rad/s)	30° (0.5246 rad)
Average value	20 m/s	10 m/s	5°/s (0.0873 rad/s)	5° (0.0873 rad)

Measurement Noise	0.2% rms of max (0.2 m/s)	0.2% rms of max (0.06 m/s)	0.2% rms of max (0.001047 rad/s)	0.2% rms of max (0.001047 rad)
Noise variance for R matrix	0.04 m/s^2	0.036 m/s^2	1.0966x10^{-6} rad/s^2	1.0966x10^{-6} rad^2

In Section 2.1.2 it was explained that the Kalman-Bucy filter requires data about the process and measurement noise in two matrices, the Q and R matrices, and the values in these matrices are used to calculate the Kalman Gain. The values in the R matrix are determined by the measurement noise variance which is shown in the bottom row of Table 1.

The R matrix is
$$\begin{bmatrix} 0.04 & 0 & 0 & 0 \\ 0 & 0.0036 & 0 & 0 \\ 0 & 0 & 0.0000010966 & 0 \\ 0 & 0 & 0 & 0.0000010966 \end{bmatrix}$$

It is recognised that the Q matrix values are more difficult to obtain. For example in (Dutton, Thompson & Barraclough, 1997: 486) it describes one approach in which the Q matrix values are chosen at random and then tuned in simulation studies. The tuning is performed based upon the knowledge that reducing the values in Q implies that there is less process noise, and the values in the Kalman gain are reduced to give greater emphasis to the predicted state generated by the model in the filter. Conversely, increasing the values in Q indicates greater process noise and so the measurements become more important, and hence the Kalman gain is increased to increase the correction from the measurements.

The initial value for Q was based upon the square of the estimated average values which gave

$$\begin{bmatrix} 400 & 0 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0.00076 & 0 & 0 \\ 0 & 0 & 0 & 0.00076 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 10 provides an example of the performance of this estimator. In this example there is a 0.5m change in the centre of gravity after 1 second.

39
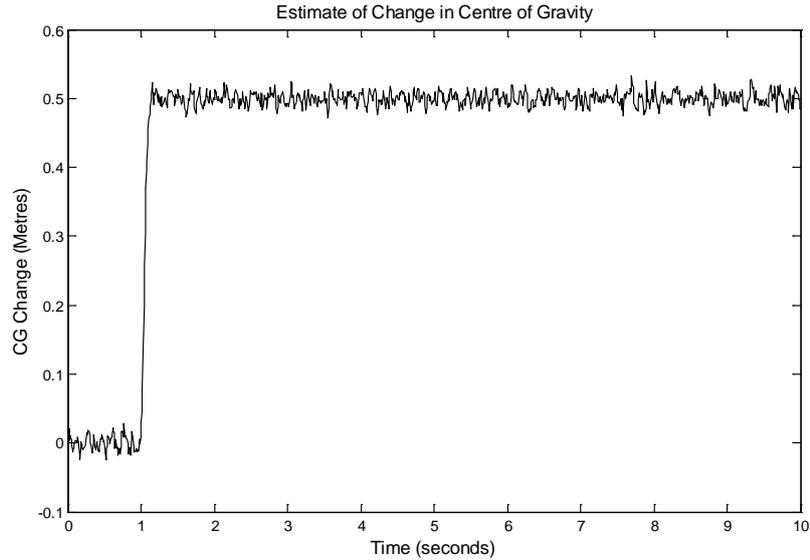
Figure 10: Phantom model (open loop) cg estimator response

It can be seen in Figure 10 that the estimator quickly estimates the correct value but the result is rather noisy.

After a number of trials the values in the Q matrix were reduced to :

$$\begin{bmatrix} 40 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0.00008 & 0 & 0 \\ 0 & 0 & 0 & 0.00008 & 0 \\ 0 & 0 & 0 & 0 & 0.001 \end{bmatrix}$$

Reducing the values in the Q matrix reduces the values in the Kalman gain matrix, and therefore there is a reduced correction from the measurements.

Figure 11 shows the results from the same test given in Figure 10, and after 10 seconds there is also a commanded elevator movement of 0.2 radians after 10 seconds. This time the cg estimate is noticeably less noisy because the reduced Kalman gain means that the noisy measurements have a reduced effect upon the estimate.

The estimate for the change in the centre of gravity was found to be most sensitive to the third and fifth row entries in the Q matrix, for q and Δcg. In general, the smaller the value for Δcg relative to the value for q in the Q matrix then the slower the response of the cg estimate, but also less noisy the estimate.
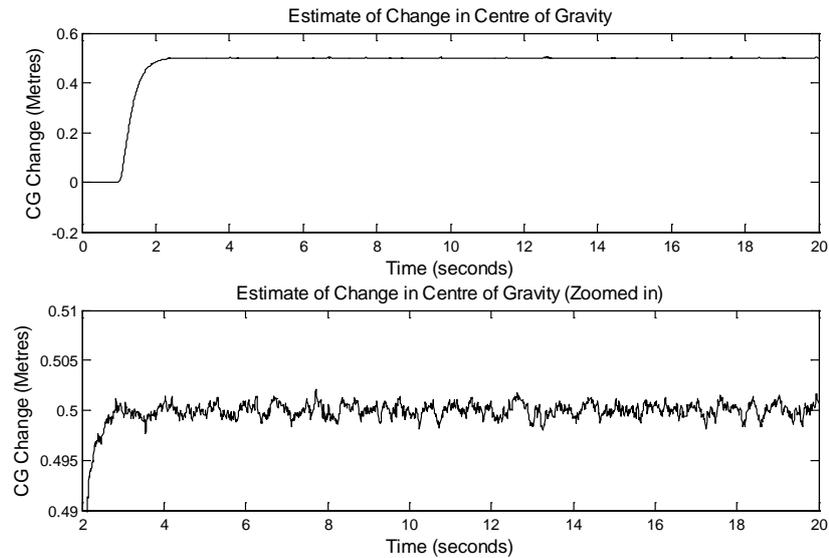
Figure 11: Phantom model (open loop) cg estimator response zoomed in

The estimate of the change in the centre of gravity can be seen to climb up to 0.5m at 1 second and is unaffected by the elevator movement at 10 seconds. The lower graph is zoomed in on the steady portion of the estimate to provide a better illustration of noise on the estimate.

4.4 Robustness Testing

In practice the Kalman-Bucy filter will not contain a model which exactly matches the real aircraft behaviour, and it is therefore necessary to perform robustness tests to examine how the estimator behaves in such a situation, and also to quantify the potential errors and to identify the key model coefficients that affect the estimator performance.

The robustness tests were performed upon the linear Phantom model described in the preceding section. The dimensionless derivatives defined in Cook (2007: 81) were increased by 10% and then the Matlab code defined in Appendix C was executed to obtain new aircraft state-space matrices. The cg estimation was then performed and the final estimate for the change in the centre of gravity of 0.5m was recorded after 20 seconds. The original estimate after 20 seconds was 0.5007m.

The Matlab code used to calculate the changed aircraft coefficients is listed in Appendix C.

Table 2: Phantom Model X and Z derivative robustness test results

Dimensionless derivative	Original Value	Test Value	Final Δcg	Final value % error
X_u	0.0076	0.00836	0.5007	0
X_w	0.0483	0.05313	0.5007	0
$X_{w\text{-}dot}$	0	0	0.5007	0
X_q	0	0	0.5007	0
X_η	0.0618	0.06798	0.5007	0
Z_u	-0.7273	-0.80003	0.5007	0
Z_w	-3.1245	-3.4365	0.5007	0
$Z_{w\text{-}dot}$	-0.3997	-0.43967	0.5007	0
Z_q	-1.2109	-1.33199	0.5007	0
Z_η	-0.3741	-0.41151	0.5007	0

X is the axial force component and Z is the normal force component. Table 2 defines the dimensionless derivatives e.g X_u is $\partial X / \partial u$. A full list of definitions is included in the List of Symbols section.

Table 2 details the results when each coefficient was increased one at a time by 10%. It can be seen that the X and Z derivatives had no effect upon the final estimate for the cg change.

This testing was repeated for the M (pitching moment) derivatives, see Table 3.

Table 3: Phantom Model M derivative robustness test results

Dimensionless derivative	Original Value	Test Value	Final Δcg	Final value % error
M_u	0.034	0.0374	0.4978	0.44
M_w	-0.2169	-0.23859	0.5379	7.58
$M_{w\text{-}dot}$	-0.591	-0.6501	0.5017	0.34
M_q	-1.2732	-1.40052	0.503	0.6
M_η	-0.5581	-0.61391	0.4069	18.62

The M derivatives did affect the final estimate, which would be expected because the M derivatives are used to directly calculate $\dot{q}$.

The final set of testing was performed using data that is used to calculate the state-space matrix coefficients using the dimensionless derivatives, and the data chosen were mass, moment of inertia and velocity, see Table 4.

Table 4: Phantom Model robustness test results

Data	Original Value	Test Value	Final Δcg	Final value % error
Velocity	178	195.8	0.3605	27.9
Mass	17642	19406.2	0.5505	10.1
Pitch Moment of inertia	165669	182235.9	0.5005	0.1

Other data such as wing area, mean aerodynamic chord and acceleration due to gravity were not varied because they either don't change (wing area) or change very little (g). For example, the gravitational constant g depends on the square of the distance from the centre of

the earth, therefore the weight of an object decreases with altitude. If an aircraft is flying at an altitude of 10km then it is approximately 6388km from the centre of the earth so the value of the gravitational constant has decreased by a ratio of the square of 6378/6388, which is about 0.9984. So a 10000kg aircraft weighs 9984kg at an altitude of 10km.

The results in Table 4 illustrate the importance of velocity to the estimator. Velocity is used to calculate m' and I'_y (see Appendix C) and is used as a multiplier to convert the dimensionless derivatives to dimensional derivatives (see Appendix 2 in Cook, 2007: 413). Therefore the velocity error causes large errors in the Kalman-Bucy filter which causes the large observed estimation error. The cg estimation error due to the mass error is proportional to the mass error, so the 10% error in mass leads to a 10% error for the estimate of the change in cg. This error can be explained by examining (58) where $\dot{q}_{unex}$ is proportional to the value for mass.

The tables do not give a complete picture of the effect of inaccurate modelling since they only provide the final estimate of the change in the centre of gravity and do not show how the estimate varies over time. The following graphs show the transient performance of the estimator when the parameters are varied one at a time.
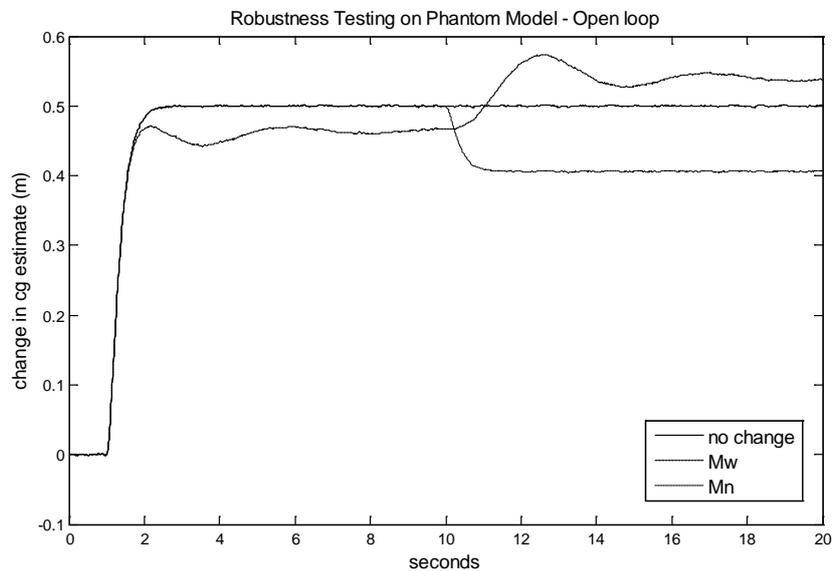
Figure 12: Phantom model robustness test – M$_w$, M$_n$
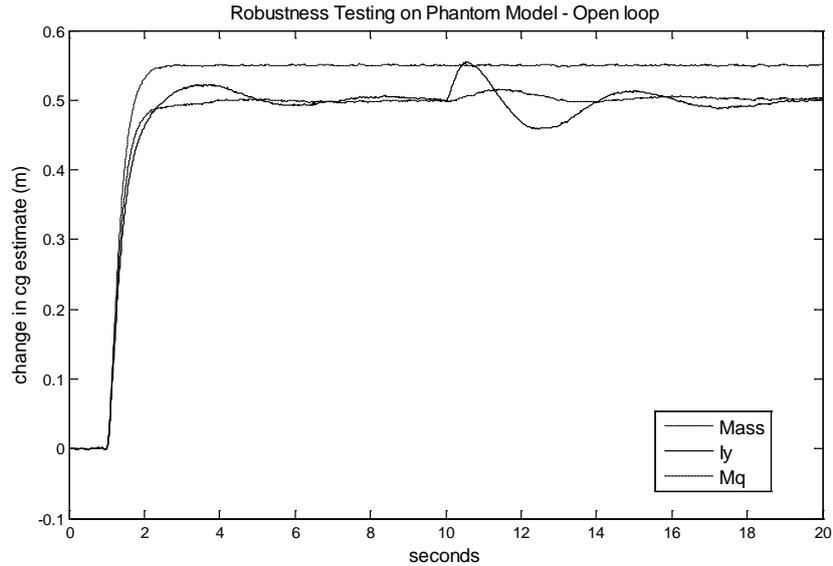
Figure 13: Phantom model robustness test – Mass, I_y, M_q

These diagrams show a number of interesting points. Figure 12 shows that the change in M_n has no effect upon the estimate until the elevator moves at 10 seconds, which is a logical result because M_n has no effect upon q until the elevator moves. Figure 13 shows that the estimate due to the change in mass is consistently approximately 10% too high. Again this result is consistent with what would be expected, because the calculation of the cg change contains a division by the aircraft mass, see (50). Note also that with the "perfect" model used for the results in Figure 10 the elevator movement had no impact, these results therefore show the "correction" action via the measurements.

The results in Table 4 indicate that the change in the moment of inertia I_y had minimal impact upon the estimation, but Figure 13 shows the transient variation in the estimate particularly when the elevator moves.

The results of the robustness tests can be summarised as:

1. The X and Z derivatives have a minimal effect upon the estimate.

2. The estimate is most sensitive to changes in velocity, M_n and mass and M_w, the other changes reduced the accuracy of the estimate by less than 1%.

3. There is a transient effect upon the estimation and this is illustrated by the effect of the pitching moment of inertia I_y

45

4.5 Pitch Rate Controller

The previous sections used the Phantom model in an open loop configuration without any form of controller. A simple pitch rate controller is added to make the results more representative of a real system. The objective of the pitch rate controller is to amend the commands into the aircraft model such that the output pitch rate q matches the command into the controller.

Note that the addition of a controller was simply to assess the performance of the estimator when a controller was added, therefore little effort was made analyzing the controller performance and no attempt was been made to check that the gains do not cause a demanded movement that may exceed physical system limits.

4.5.1 PID Controller

The use of a PID controller is shown below.
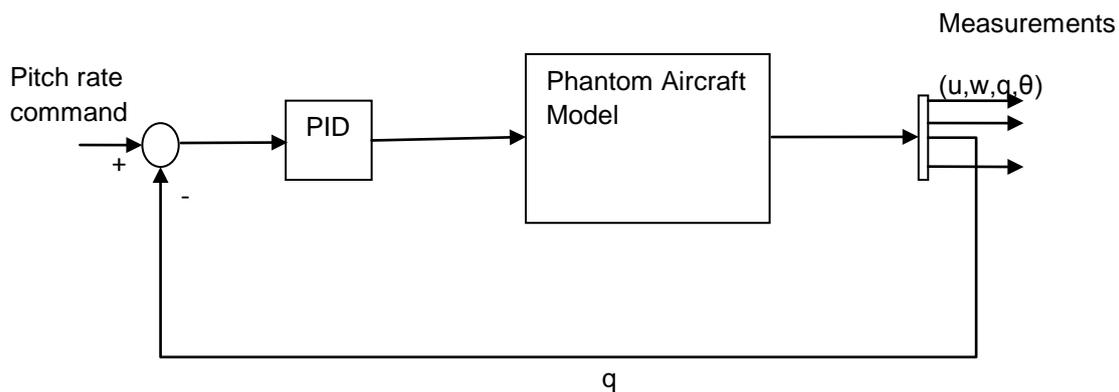
Figure 14: Phantom model with PID controller

After a number of trials a proportional gain of 50 and integral gain of 10 was selected. The performance is shown below when a pitch rate of 0.2 rad/s^2 is commanded at 1 second.
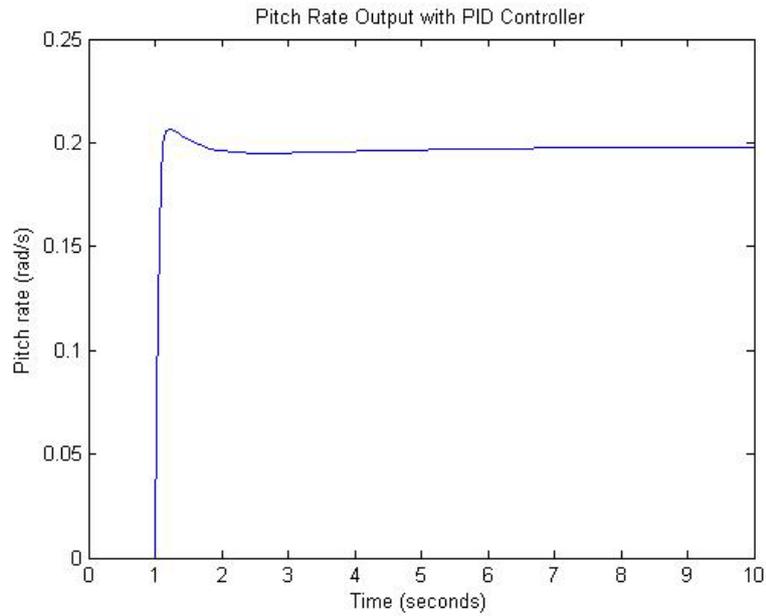
Figure 15: Phantom model with PID controller – pitch rate output

A more complex, and realistic, command sequence was next tested. At 1 second a pitch rate of 0.5 radians/s was commanded and then at 4 seconds a zero pitch rate command is issued. At 6 seconds a -0.5 radian/s pitch rate command is issued followed by a zero command at 9 seconds. Also for greater realism the pitch rate commands are steeply ramped up and down instead of being step commands.

The results are shown in Figure 16 and Figure 17.
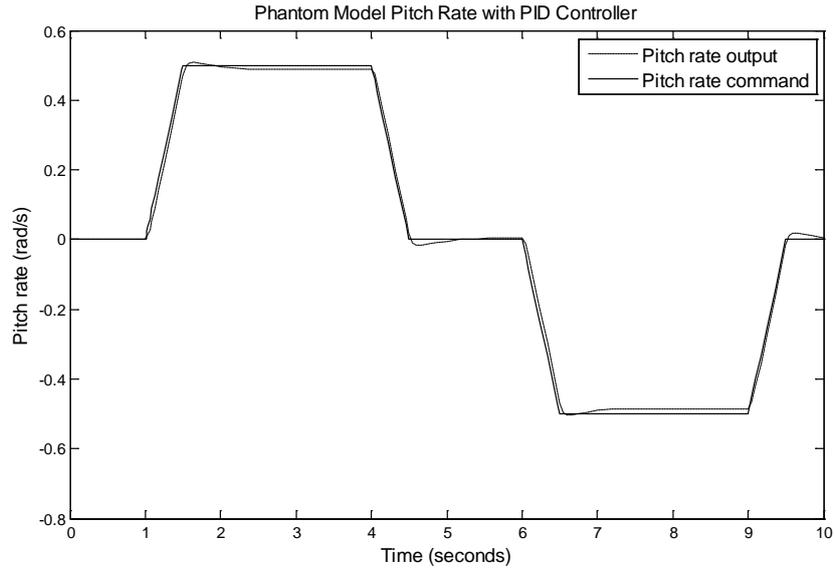
Figure 16: Phantom model with PID controller – pitch rate command and output

The final stage in this model development is to incorporate the pitch rate controller with the Phantom aircraft model, add noise to the model measurements and use a Kalman-Bucy filter to estimate the change in the centre of gravity.

Figure 17 shows the results when the pitch rate commands described above are issued, and the centre of gravity changes by 0.2m at 3 seconds.
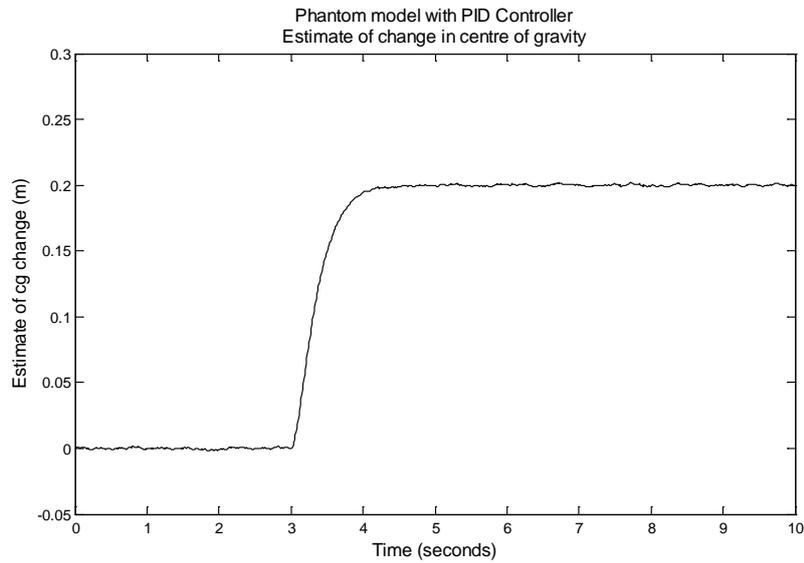
Figure 17: CG estimate for Phantom model with changing pitch rate

48

4.5.2 PID Controller plus Estimator Robustness Tests

For completeness the robustness tests were repeated on the Phantom model with the PID pitch rate controller and estimator. As before the parameter values were increased by 10%. The X and Z derivatives did not affect the estimate for the change in the centre of gravity, and therefore the results from the changed X and Z derivatives are omitted.

The change to the M derivatives did affect the estimate, and in this configuration with the PID controller M_w, M_η and then M_u had the biggest effect upon the final estimate, see Table 5. The M_w and M_u changes had a substantially increased impact upon the estimate compared to the scenario without the pitch rate controller. This can be explained by the controller causing a continuous pitch rate change whereas without it the pitch rate q would tend to damp to zero. Therefore with the controller u and w increase in value more and hence have a greater impact upon $\dot{q}$ and the change in the centre of gravity. This also leads to the conclusion that the Q matrix which was tuned for open-loop may need to be retuned in the closed loop scenario with a pitch rate controller.

Table 5: Phantom Model with PID Controller robustness testing results

Dimensionless derivative	Original Value	Test Value	Final Δcg	Final value % error
M_u	0.034	0.0374	0.4776	4.48
M_w	-0.2169	-0.23859	0.3801	23.98
$M_{w\text{-dot}}$	-0.591	-0.6501	0.4982	0.36
M_q	-1.2732	-1.40052	0.4948	1.04
M_η	-0.5581	-0.61391	0.5989	19.78

The model was also tested with changes to mass, moment of inertia and velocity as shown in Table 6.

Table 6: Phantom Model with PID Controller robustness testing results

Phantom Data	Original Value	Test Value	Final Δcg	Final value % error
Velocity	178	195.8	0.5676	13.52
Mass	17642	19406.2	0.5506	10.12
Pitch Moment of inertia	165669	182235.9	0.5029	0.58

As before velocity and mass have a major impact upon the final estimated value and a 10% increase in mass causes the estimate of the change in the centre of gravity to increase by 10%. The percentage change in cg error when velocity was changed in the closed loop model has changed from -27.9% to 13.52%. This apparent improvement is misleading because the cg estimate had not stabilized. In the closed loop model version the aircraft was continuing to pitch and the cg estimate was continuing to increase, therefore if the model had run for longer then the percentage error would have increased.

4.6 Summary

This section has described how a moment, representative of a shift in cg, is injected into a linear aircraft model, and the change in cg is then estimated by using a Kalman-Bucy filter. It has shown the relationship between the values in the Kalman-Bucy Q and R matrices and the speed of response of the estimator and the noise in the estimate. The preferred values in the Q matrix are a trade off between the speed of estimator response and noisiness of the estimate.

This section has also shown that the cg estimator is still accurate when a pitch rate controller is added to the system to make it more representative of a genuine aircraft Flight Control System, and maintains an accurate estimate when the aircraft undergoes some pitching manoeuvres. Finally this section has shown the key coefficients critical to the accuracy of the estimator with the Phantom model, the main ones being M_w and M_η.

5. Application of a Centre of Gravity Estimator to a non-linear aircraft model

This section will describe the development and design of a centre of gravity estimator applied to a non-linear aircraft model.

5.1 Non-linear Aircraft Model

The non-linear aircraft model used in the remainder of this thesis is called ADMIRE. The ADMIRE (Aero-Data Model in a Research Environment) model is a generic model of a small single-seat fighter aircraft with a delta-canard configuration, see Figure 18.
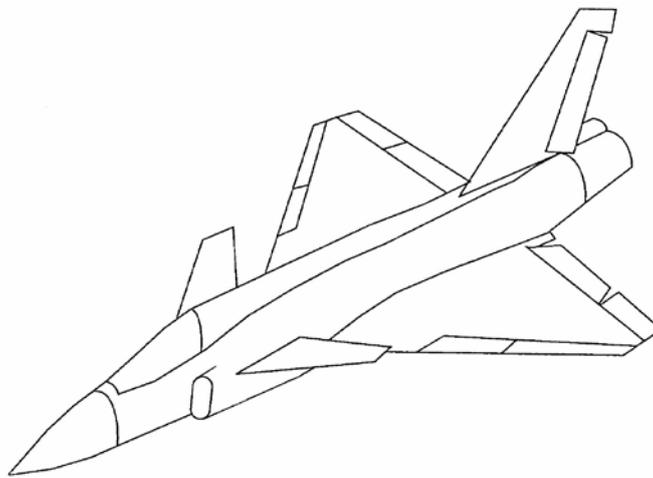
Figure 18: ADMIRE aircraft

The ADMIRE model contains twelve states : velocity (V_T), angle of attack (α), sideslip (β), roll rate (p_b), pitch rate (q_b), yaw rate (r_b), roll angle (φ), pitch angle (θ), yaw angle (ψ), longitudinal coordinate (x_v), lateral coordinate (y_v), normal coordinate (z_v) plus additional states for sensors, actuators and FCS. The model is fully described by Forsell and Nilsson (2005).

$$ADMIRE\ x = [V_T, \alpha, \beta, p, q, r, \varphi, \theta, \psi, x_v, y_v, z_v] \tag{63}$$

The ADMIRE aircraft model can model a variety of parametric uncertainties, amongst them are changes in the longitudinal cg position x_{cg} and also aircraft mass. This facility has been used to inject cg changes into the ADMIRE aircraft model.

A good description of a non-linear aircraft model and the 6-DOF (Degrees Of Freedom) equations of motion is contained in Stevens and Lewis (2003). The moment equations of

motion defined in ADMIRE are consistent with those defined in (Steven and Lewis, 2003 : 110).

The ADMIRE control system has been tuned at 30 design points, at 20m, 3000m and 6000m altitude and at Mach 0.22, 0.35, 0.45, 0.55, 0.8, 0.9, 0.95, 1.0, 1.05, 1.1, 1.2. Therefore the model should be used within these design points to obtain sensible values.

The ADMIRE model is provided with trim command files which try to trim the aircraft in straight and level flight or at a specified angle of attack. These trim commands take the aircraft speed and altitude as inputs and use a minimization routine to generate the trim. It should be noted that the trim routines generate an approximation, and at some speed and altitude combinations they are unable to generate a trimmed aircraft.

5.2 Measurement Noise

The measurement noise used with the ADMIRE non-linear aircraft model, and used to set the Kalman filter R matrix, was obtained from sample test flight data provided by BAE Systems. A sample of the test flight data is shown in Figure 19, and the other three data set diagrams are included in Appendix D.
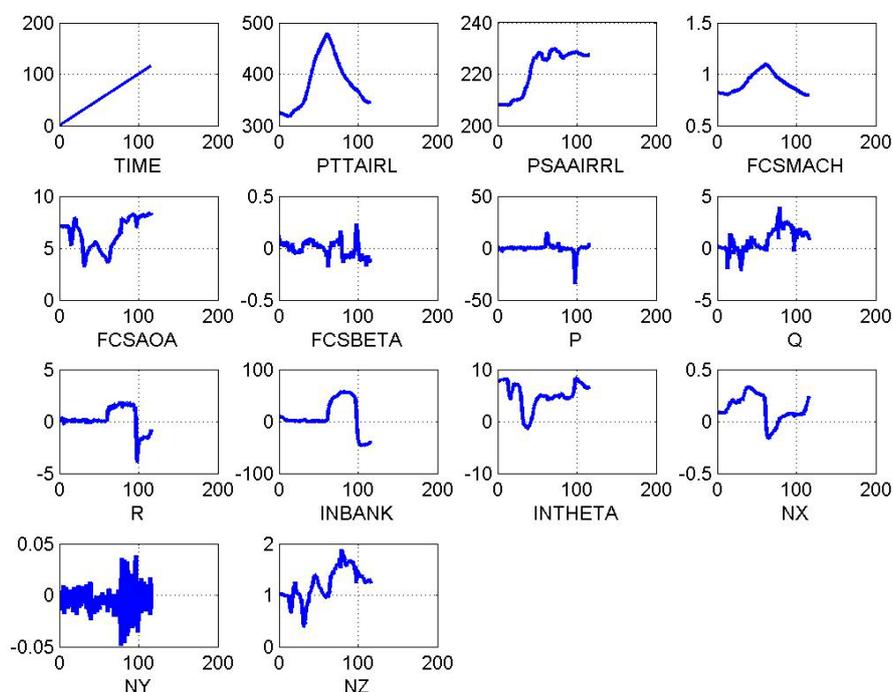
Figure 19: Sample flight test measurements used to obtain measurement noise

The graphs contained within Figure 19 are :

PTTAIRL   - total air pressure (mbar)

PSAAIRRL   - static air pressure (mbar)

FCSMACH   - Mach number

FCSAOA   - Angle of attack (°)

FCSBETA   - Beta (°)

P   - roll rate (°/s)

Q   - pitch rate (°/s)

R   - yaw rate (°/s)

INBANK   - bank angle (°)

INTHETA   - pitch angle (°)

NX   - longitudinal acceleration ('g')

NY   - lateral acceleration ('g')

NZ   - normal acceleration ('g')

To obtain a measure of the noise in these measurements the Matlab 'wden' command was used. 'wden' is a one dimensional de-noising function in the Matlab Wavelet toolbox which provides a noise free signal by thresholding the wavelet coefficients of the signal. A wavelet is a mathematical function used to cut up the data into different frequency components and then study it according to its scale. The 'wden' command used is given below:

xd = wden(x,'heursure','s','one',3,'sym8');

where:

'heursure'   - a heuristic used to select the wavelet threshold

's'   - use soft thresholding

'one'   - no rescaling used

3   - wavelet decomposition level

'sym8'   - desired wavelet name

Noise was added to the noise free signal to recreate the original signal and to obtain a measure of the RMS of the noise. The reconstituted noisy signal was visually compared with the original to spot any obvious errors. An additional check was performed by a comparison with the results of analysis at BAE Systems in which the noise was measured by taking the Fourier transform of the signal and removing the noise components, and then regenerating the 'clean' signal and subtracting from the original to leave the 'noise' part of the signal. A comparison of the results showed that the noise variance was in the same order of magnitude.

Table 7: ADMIRE Measurement Noise

	RMS noise	RMS in ADMIRE model units	Variance
Mach	0.0001 Mach	0.0001 Mach	1e-8 Mach^2
AOA	0.004°	0.0000698 rad	4.87e-9 rad^2
Beta	0.004°	0.0000698 rad	4.87e-9 rad^2
P	0.14°/s	0.00244 rad/s	5.95e-6 (rad/s)^2
Q	0.05°/s	0.00087 rad/s	7.6e-7 (rad/s)^2
R	0.05°/s	0.00087 rad/s	7.6e-7 (rad/s)^2
Bank	0.025°	0.00044 rad	1.9e-7 rad^2
Theta	0.004°	0.0000698 rad	4.87e-9 rad^2
Nx	0.0028 'g'	0.0028 'g'	7.84e-6 'g'^{2}
Ny	0.008 'g'	0.008 'g'	6.4e-5 'g'^{2}
Nz	0.008 'g'	0.008 'g'	6.4e-5 'g'^{2}
Psa	0.06 mbar	0.06 mbar	3.6e-3 mbar^2

There are some ADMIRE signals that are missing from the flight test data: velocity, phi and z.

a. Velocity

Since the speed of sound is 340.3 m/s at sea level then the RMS for Mach was converted into an RMS for m/s, giving a velocity RMS of 0.034 m/s.

b. Phi - Use same value as theta.

c. Altitude (z)

The Matlab 'atmospalt' command was used to convert the static pressure (PSAAIRRL) to altitude. At higher altitude errors in pressure have more effect due to the reduced air pressure. For example a static pressure of 220 mbar (22000 pascal) gives an altitude of 11180m and an RMS of 0.06 mbar gives an error of 1.729m. However at approximately 110m altitude the error from an RMS of 0.6 mbar falls to 0.5m. An RMS error for z of 1m was selected because this was between the two extreme values of 1.7m and 0.5m.

Table 8 contains the noise variance values which are used in the remainder of this thesis.

Table 8: ADMIRE Measurement Noise

	Chosen RMS noise	Variance
Velocity	0.034 m/s	1.156e-3 $(m/s)^2$
AOA	0.0000698 rad	4.87e-9 rad^2
Beta	0.0000698 rad	4.87e-9 rad^2
P	0.00244 rad/s	5.95e-6 $(rad/s)^2$
Q	0.00087 rad/s	7.6e-7 $(rad/s)^2$
R	0.00087 rad/s	7.6e-7 $(rad/s)^2$
Bank	0.00044 rad	1.9e-7 rad^2
Theta	0.0000698 rad	4.87e-9 rad^2
Yaw	0.0000698 rad	4.87e-9 rad^2
z	1.0 m	1.0 m^2

Nx	0.0028 'g'	7.84e-6 'g'^{2}
Ny	0.008 'g'	6.4e-5 'g'^{2}
Nz	0.008 'g'	6.4e-5 'g'^{2}

5.3 Longitudinal cg estimation

5.2.1 ADMIRE Longitudinal cg equation

This section uses the aircraft equations of motion in ADMIRE to generate the relationship between changes in longitudinal cg and unexpected changes in pitch acceleration $\dot{q}_{unex}$.

In ADMIRE the pitch acceleration equation is defined by (64).

$$\dot{q}_b = C_5 \cdot p_b \cdot r_b - C_6(p_b^2 - r_b^2) + C_7 \cdot M_y \tag{64}$$

where

$$C_5 = \frac{I_z - I_x}{I_y}$$

$$C_6 = \frac{I_{xz}}{I_y}$$

$$C_7 = \frac{1}{I_y}$$

$\dot{q}_b$ is the pitch acceleration (rad/s^2)

r_b is the yaw velocity (rad/s)

p_b is the roll velocity (rad/s)

M_y is the pitching moment (N m)

I_x is the x body moment of inertia (kg m^2)

I_y is the y body moment of inertia (kg m^2)

I_z is the z body moment of inertia (kg m^2)

I_{xz} is the x-y body axis product of inertia (kg m^2)

56

The total pitching moment equation is defined by (65).

$$M_y = \bar{q} \cdot S_{ref} \cdot C_{ref} \cdot Cm_{tot} - x_{cg} \cdot F_z + z_{cg} \cdot F_x - 0.15T_x + 5.5T_z \qquad (65)$$

where

$\bar{q}$ is the dynamic pressure (pascals)

S_{ref} is the wing surface area (m^2)

C_{ref} is the mean aerodynamic chord (m)

Cm_{tot} is the pitching moment coefficient

F_z is the total force in body-fixed z axis (N)

F_x is the total force in body-fixed x axis (N)

x_{cg} is the centre of gravity along the x axis

z_{cg} is the centre of gravity along the z axis

T_x is the engine thrust along the x axis (N)

T_z is the engine thrust along the z axis (N)

It can be seen in (65) that the effect on the pitching moment of a cg shift along the x axis is dependent upon the normal force F_z.

$$\Delta M_y = \Delta cg \cdot F_z \qquad (66)$$

The change in the moment can be calculated from the unexpected change in the pitch acceleration as :

$$\Delta M_y = I_y \cdot \dot{q}_{unex} \qquad (67)$$

Substituting (67) into (66) :

$$\Delta cg = \frac{I_y \cdot \dot{q}_{unex}}{F_z}$$

$$(68)$$

Note that although F_z is an output in the ADMIRE model n_z has been used, because ADMIRE provides a sensor model for n_z. The relationship between F_z and n_z is :

$$n_z = \frac{-F_z}{mass \cdot g}$$

<div align="right">(69)</div>

Therefore the equation used with the ADMIRE model to estimate changes in longitudinal cg is :

$$\Delta cg = \frac{I_y \cdot \dot{q}_{unex}}{-n_z \cdot mass \cdot g}$$

<div align="right">(70)</div>

5.2.2 Longitudinal CG Estimation Model

To obtain the coefficients in the Kalman-Bucy filter the aircraft is initially trimmed in straight and level flight using the 'admtrim_sl' script supplied with the ADMIRE model. The results of the trim are the initial conditions of the aircraft inputs and the aircraft outputs. The trim also incorporates the FCS which tried to maintain the trimmed aircraft state. The model can then be linearised by using the Matlab 'linmod' command to obtain its state-state equations. The normal ADMIRE state equations contain the 12 states defined in (63).

To estimate longitudinal cg the Kalman-Bucy filter needs to be augmented with $\dot{q}_{unex}$, the process to do this was described in section 3.3. The value of $\dot{q}_{unex}$ is then scaled by the pitching moment of inertia I_y, mass, gravity and nz to obtain the dxcg estimate (see (70)).

The most important state affecting pitch acceleration is the angle of attack α. Other important states affecting pitch acceleration are velocity and altitude since they define the dynamic pressure $\bar{q}$ which is used in the pitching moment equation (65).

The aircraft state stored in the Kalman-Bucy filter and used to estimate longitudinal cg is given in (71).

$$Estimator\ x = [V_T, \alpha, \beta, p, q, r, \varphi, \theta, \psi, z_v, \dot{q}_{unex}] \tag{71}$$

The 'x' and 'y' states are removed from the filter because they do not contribute to the cg estimate. An estimator focused solely upon longitudinal cg could probably still be accurate with fewer states, e.g. $[V_T, \alpha, q, z_v, \dot{q}_{unex}]$, but this approach wasn't taken because a combined longitudinal and lateral cg was later developed, see section 7.

The Kalman-Bucy filter is checked for observability by using the Matlab 'rank' and 'obsv' functions, and the filter has full observability.

The Kalman gain used in the filter is obtained by using the Matlab 'lqe' function.

The structure of the longitudinal cg estimator is shown in Figure 20.
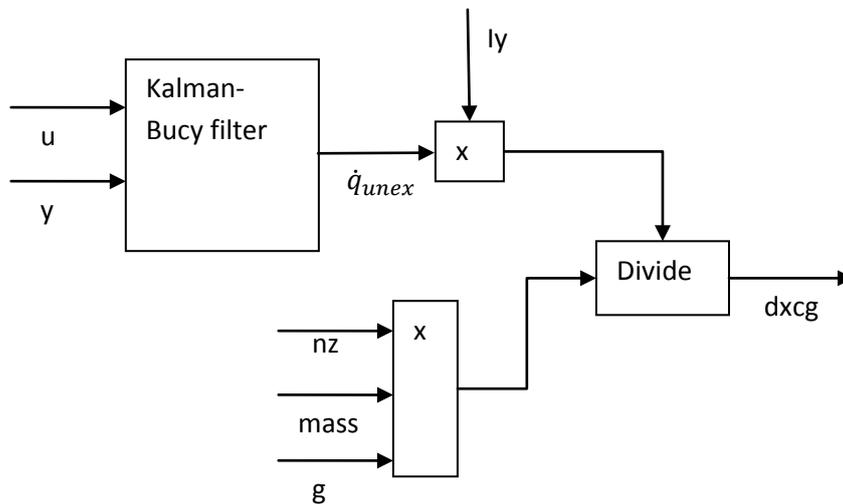
Figure 20: Structure of longitudinal cg estimator

The commands in 'u' are listed in Table 9. The aircraft measurements are listed in Table 10.

Table 9: ADMIRE commands

Abbreviation	Description
drc	Right canard angle (radians)
dlc	Left canard angle (radians)
droe	Right outer elevon angle (radians)
drie	Right inner elevon angle (radians)
dlie	Left inner elevon angle (radians)
dloe	Left outer elevon angle (radians)
dr	Rudder angle (radians)

Abbreviation	Description
dle	Leading-edge flap angle (radians)
ldg	Landing gear (Boolean)
tss	Throttle setting

Table 10: ADMIRE measurements

Abbreviation	Description
Vt	Airspeed (m/s)
alpha	Angle of attack (radians)
beta	Angle of sideslip (radians)
p	Roll angular rate (radians/second)
q	Pitch angular rate (radians/second)
r	Yaw angular rate (radians/second)
Psi (ψ)	Azimuth angle (radians)
Theta (θ)	Elevation (radians)
Phi (φ)	Bank (radians)
z	Z position (m)

The script to linearise the aircraft and create the estimator matrices is included in Appendix F.

5.2.3 Longitudinal CG Estimation Results

In a similar process to that described in section 4.3 the Q matrix in the Kalman-Bucy filter was iteratively amended to provide 'best' results, where 'best' was a trade-off between the noisiness of the estimate and speed of response. It was noted earlier in section 4.3 that the important coefficients in the Q matrix are the ones for 'q' and $\dot{q}_{unex}$ and therefore most attention was paid to these two coefficients.

The results in this section were obtained using the Q = diag([10 0.02 0.01 0.01 0.01 0.01 0.01 0.01 0.01 10 0.2]). The R matrix contained the noise variances defined in Table 8.

Figure 21 shows the estimate for a 0.1m shift in longitudinal cg after 10 seconds when the non-linear ADMIRE aircraft is trimmed in straight and level flight at Mach 0.4 and at 5000m altitude.
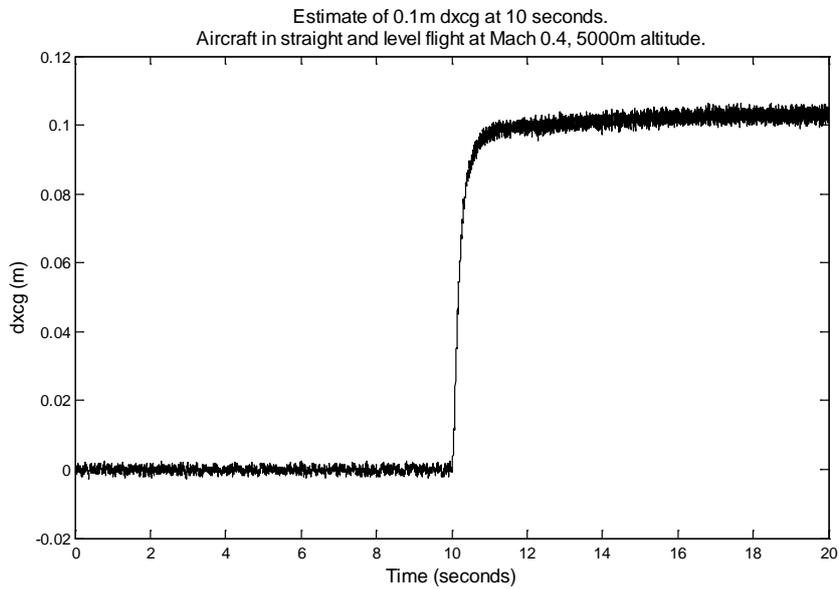
Figure 21: Estimate of 0.1m change in longitudinal cg. Aircraft trimmed at Mach 0.4, 5000m altitude.

Figure 22 shows the same scenario as Figure 21 but without measurement noise added to the measurements.
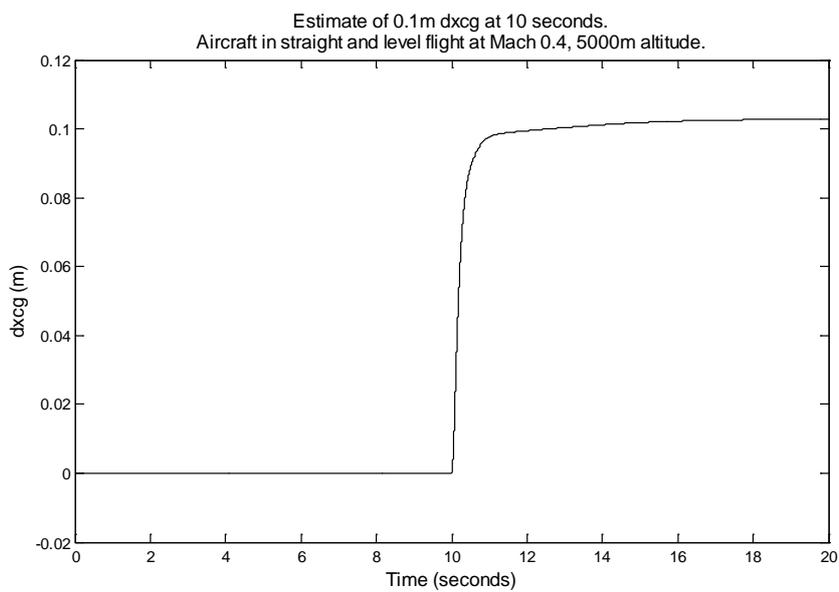
Figure 22: Estimate of 0.1m change in longitudinal cg. No measurement noise.

Figure 23 shows the estimate for a -0.05m shift in longitudinal cg after 10 seconds, this time with the aircraft trimmed at Mach 0.6 and at 3000m altitude. For clarity measurement noise has not been added.
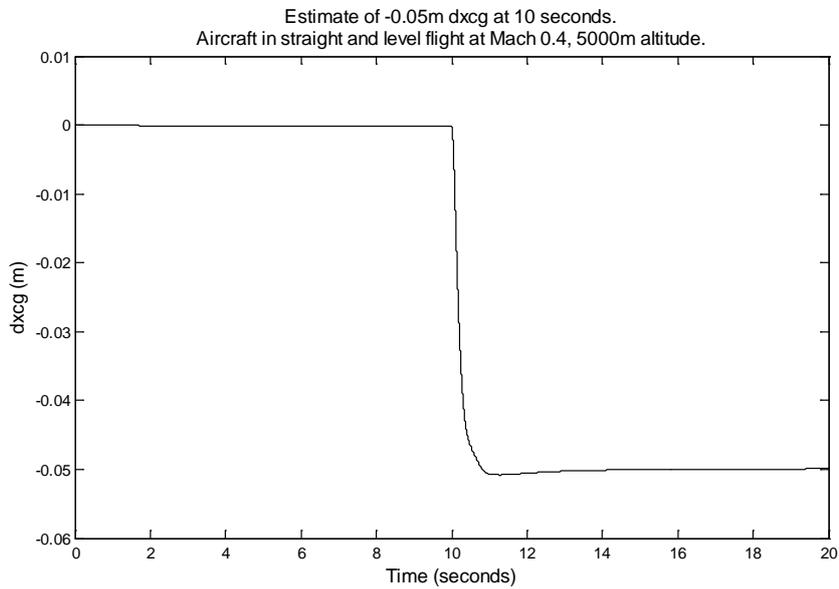
Figure 23: Estimate of -0.05m change in longitudinal cg. No measurement noise.

Figure 24 shows the estimate for a shift in longitudinal cg driven by a sine wave. The aircraft is trimmed at Mach 0.5 at 4000m altitude. As before measurement noise has not been added.
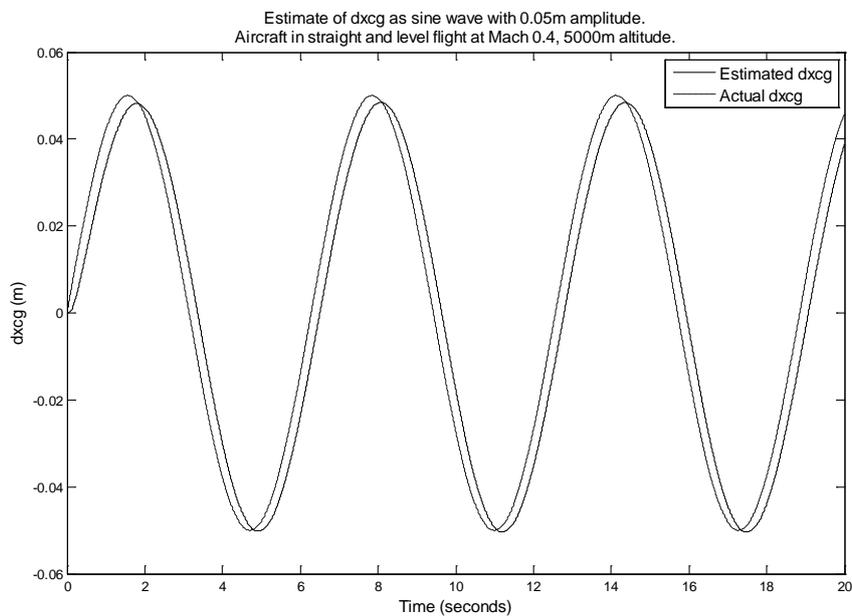
Figure 24: Estimate of sine wave longitudinal cg shift. No measurement noise.

These results show that the basic concept for cg estimation described in the previous section for the linear Phantom model also apply to the non-linear ADMIRE aircraft model. However the tests were only applied to the aircraft trimmed in straight and level flight at the Mach 0.4 and 5000m altitude. To obtain a better understanding of the cg estimator performance at different speeds the aircraft was then trimmed and linearised at different speeds (Mach 0.25, 0.35, 0.45, 0.55) at 2000m altitude. The same startup script was used as before, therefore each time the cg estimator was also configured for the same speed and altitude as the aircraft. Figure 25 shows estimates for a 0.1m shift in longitudinal cg after 10 seconds.
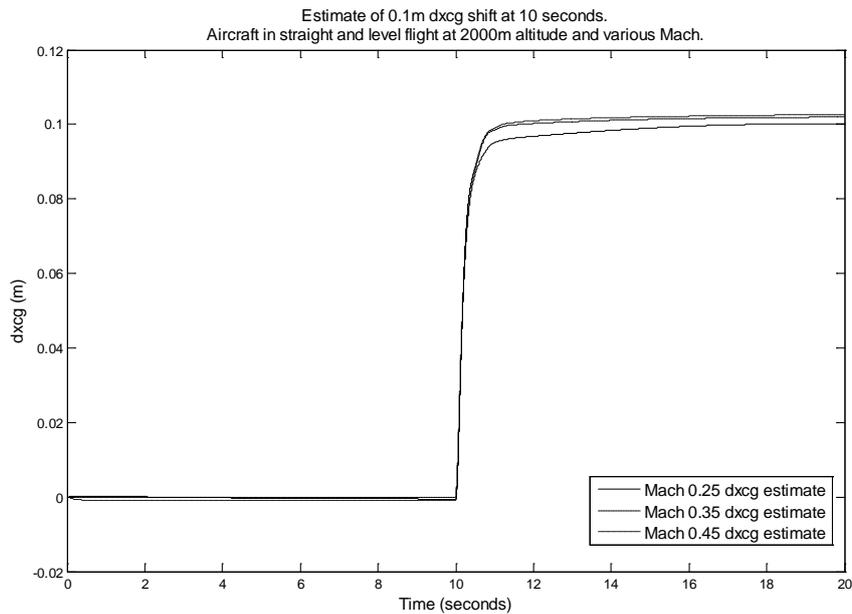
Figure 25: Estimate of longitudinal cg change at Mach 0.25, 0.35 and 0.45 at 2000m altitude. No noise added.

Figure 26 shows estimates for a 0.1m shift in longitudinal cg after 10 seconds when the aircraft is at 3000m altitude and at various Mach (0.55, 0.65, 0.75).

63

Estimate of 0.1m dxcg shift at 10 seconds.
Aircraft in straight and level flight at 3000m altitude and various Mach.
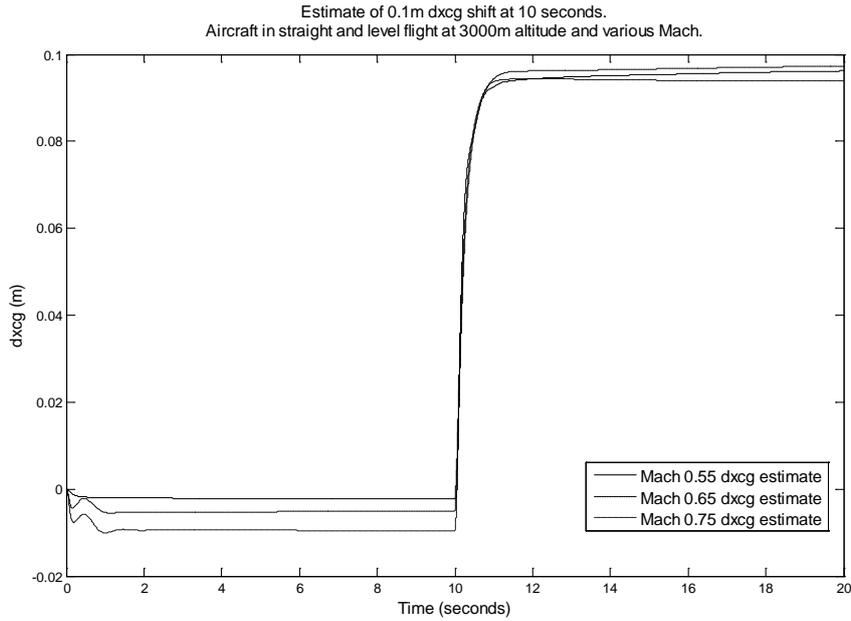
Figure 26: Estimate of longitudinal cg change at Mach 0.55, 0.65 and 0.75 at 3000m altitude. No noise added.

Figure 27 shows estimates for a 0.1m shift in longitudinal cg after 10 seconds when the aircraft is at 3000m altitude and at various Mach (0.85, 0.95, 1.05, 1.15).

Estimate of 0.1m dxcg shift at 10 seconds.
Aircraft in straight and level flight at 3000m altitude and various Mach.
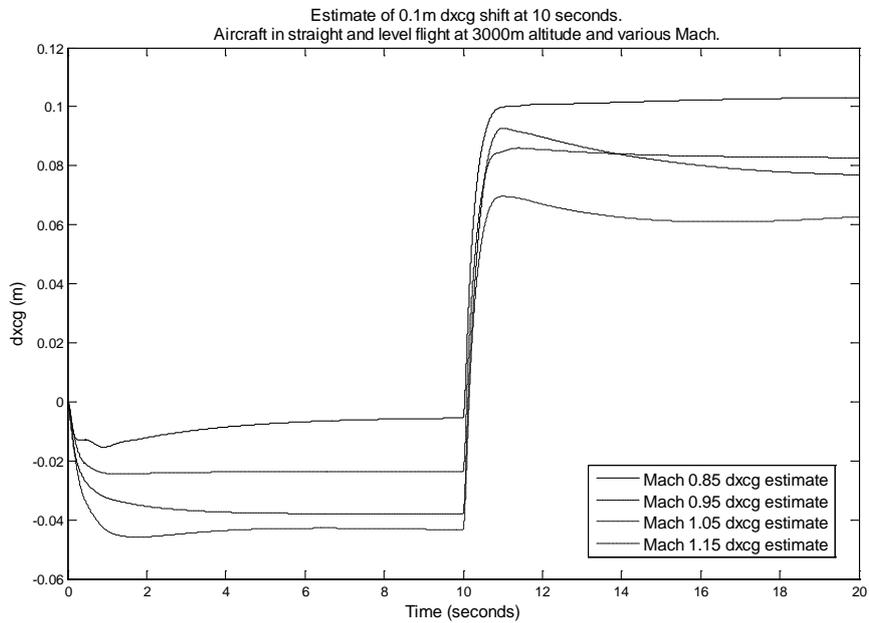
Figure 27: Estimate of longitudinal cg change at Mach 0.85, 0.95, 1.05 and 1.15 at 3000m altitude. No noise added.

64

It can be seen from Figure 27 that at speeds close to the speed of sound the cg estimate is less accurate, this is particularly noticeable at Mach 1.05.

In this sequence of tests it should be noted that the aircraft trim was not perfect and at higher speeds the requested speed and altitude were chosen to permit the trim routine to complete and just remain within the tolerance parameters of the script. In other words at high speeds it proved more difficult to obtain linearised aircraft model coefficients. At some speeds and altitude combinations the trim routine was unable to obtain a satisfactory trim.

5.3 Lateral cg estimation

5.3.1 Lateral cg equations

This section uses the aircraft equations of motion in ADMIRE to generate the relationship between changes in lateral cg and unexpected changes in roll acceleration $\dot{p}_{unex}$.

In ADMIRE the roll acceleration equation is defined in :

$$\dot{p} = (C1 \cdot r + C2 \cdot p)q + C3 \cdot M_x + C4 \cdot M_z \tag{72}$$

where

$$C1 = \frac{(I_y - I_z)I_z - I_{xz}^2}{I_x I_z - I_{xz}^2}$$

$$C2 = \frac{(I_x - I_y + I_z)I_{xz}}{I_x I_z - I_{xz}^2}$$

$$C3 = \frac{I_z}{I_x I_z - I_{xz}^2}$$

$$C4 = \frac{I_{xz}}{I_x I_z - I_{xz}^2}$$

The roll moment Mx and yaw moment Mz equations are defined in (73) and (74).

$$M_x = \bar{q} \cdot S_{ref} \cdot b_{ref} \cdot Crm_{tot} - zcg \cdot F_y + dycg \cdot F_z \tag{73}$$

$$M_z = \bar{q} \cdot S_{ref} \cdot b_{ref} \cdot Cym_{tot} - dxcg \cdot F_y - dycg \cdot F_x - 5.5T_y \tag{74}$$

where

b_{ref} is the reference span

Crm_{tot} is the rolling moment coefficient

Cym_{tot} is the yaw moment coefficient

F_y is the total force in the body-fixed y axis (N)

T_y is the engine thrust along the y axis (N)

It can be seen in (73) that the effect on the roll moment of a cg shift along the y axis depends upon the normal force F_z.

$$\Delta M_x = dycg \cdot F_z \tag{75}$$

There is also an effect upon the yaw moment from the cg shift along the y axis.

$$\Delta M_z = dycg \cdot F_x \tag{76}$$

From (72) the change in the moment can be calculated as the unexpected change in the roll acceleration:

$$\dot{p}_{unex} = C3 \cdot dycg \cdot F_z + C4 \cdot dycg \cdot F_x \tag{77}$$

Therefore :

$$dycg = \frac{\dot{p}_{unex}}{C3 \cdot F_z + C4 \cdot F_x}$$

$$\tag{78}$$

In full the equation is :

$$dycg = \frac{\dot{p}_{unex}(I_{xz} - I_{xz}^2)}{I_x \cdot F_z + I_{xz} \cdot F_x}$$

$$\tag{79}$$

As before 'nz' and 'nx' are used instead of F_z and F_x because these are measurements that would be available from a real aircraft. The equation then becomes :

$$dycg = \frac{\dot{p}_{unex}(I_x I_z - I_{xz}^2)}{-mass \cdot g(I_x \cdot nz + I_{xz} \cdot nx)}$$

(80)

5.3.2 Lateral CG Estimation Model

The important states affecting roll acceleration are sideslip β, roll acceleration p, yaw acceleration r, and the bank angle φ.

As noted earlier the ADMIRE aircraft model contains twelve states but not all of them are needed to estimate unexpected angular roll acceleration and the 'x' and 'y' states have been left out of the estimator.

The Kalman-Bucy filter is augmented with $\dot{p}_{unex}$ as described in section 3.3. The value of $\dot{p}_{unex}$ is then scaled by the moments of inertia along the x-axis I_x, moment of inertia along the z axis I_z, the deviation moment of inertia I_{xz}, the longitudinal acceleration nx and normal acceleration nz, mass and gravity to obtain the dycg estimate.

As before the aircraft is trimmed in straight and level flight using 'admtrim_sl', and then the aircraft is linearised using Matlab linmod' to obtain the coefficients to use in the Kalman-Bucy filter.

The standard ADMIRE model did not output nx which is required in (80), therefore the ADMIRE code was modified to output it.

The aircraft state stored in the Kalman-Bucy filter and used to estimate lateral cg is given in (81).

$$Estimator \; x = [V_T, \alpha, \beta, p, q, r, \varphi, \theta, \psi, z_v, \dot{p}_{unex}]$$

(81)

The Kalman-Bucy filter is checked for observability by using the Matlab 'rank' and 'obsv' functions, and the filter has full observability.

The Kalman gain used in the filter is obtained by using the Matlab 'lqe' function.

The structure of the lateral cg estimator is shown in Figure 28.

67
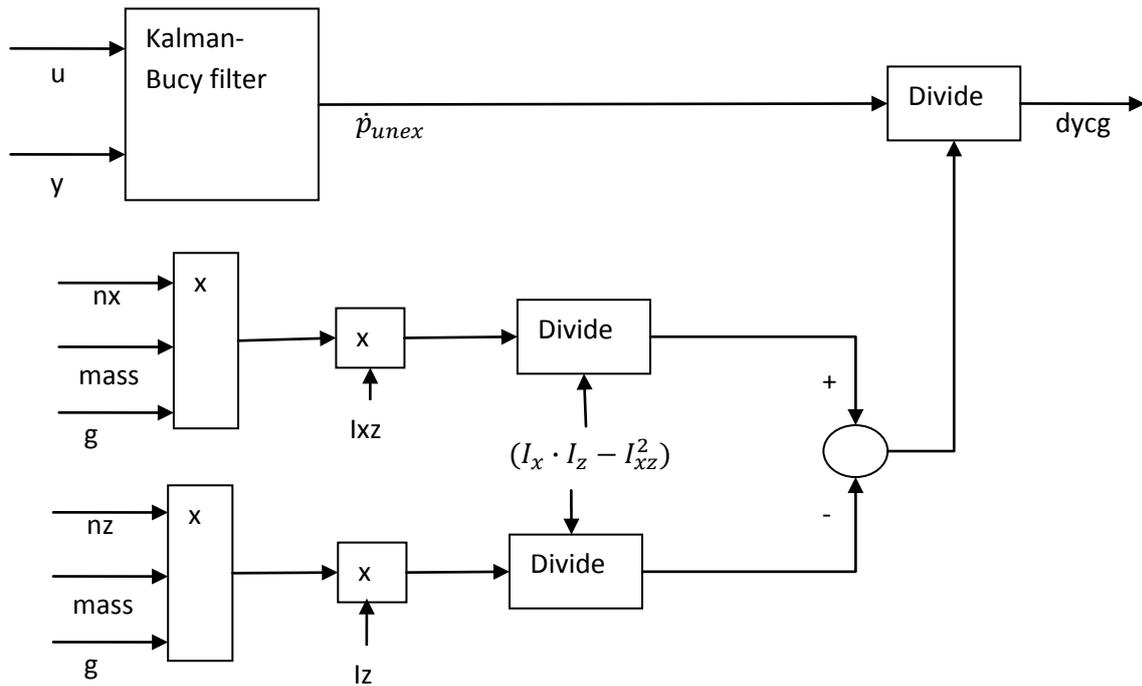
Figure 28: Structure of lateral cg estimator

The script to linearise the aircraft and create the estimator matrices is included in Appendix F.

5.3.3 Lateral CG Estimation Results

As before the Q matrix in the Kalman-Bucy filter was iteratively amended to provide 'best' results where 'best' was a trade-off between the noisiness of the estimate and speed of response. For the lateral cg estimator the important coefficients in the Q matrix are the ones for p and $\dot{p}_{unex}$.

The results in this section were obtained using Q = diag([10 0.02 0.02 0.02 0.01 0.1 0.01 0.01 0.01 10 0.2]).

Figure 29 shows the estimate for a 0.05m shift in lateral cg after 10 seconds when the ADMIRE aircraft is trimmed in straight and level flight at Mach 0.4 and at 5000m altitude.
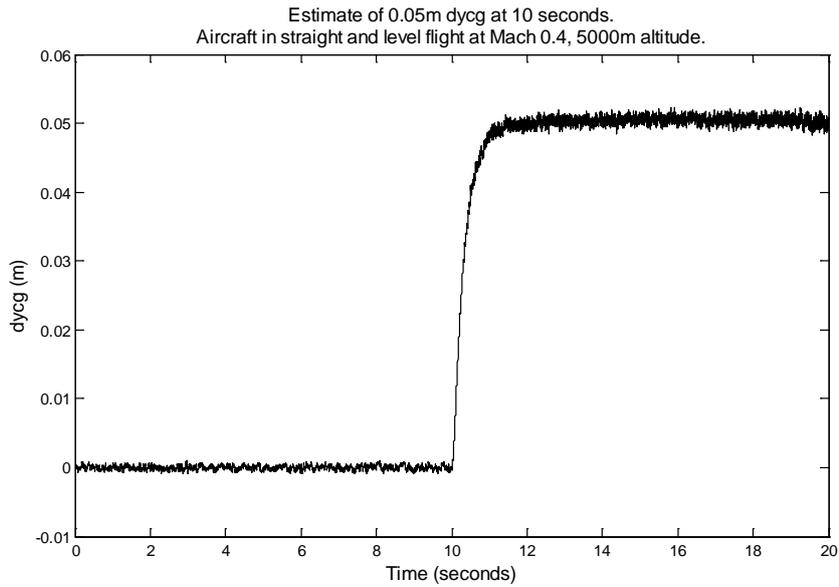
Figure 29: Estimate of 0.05m change in lateral cg

Figure 30 shows the same scenario as Figure 29 but without measurement noise added to the measurements.
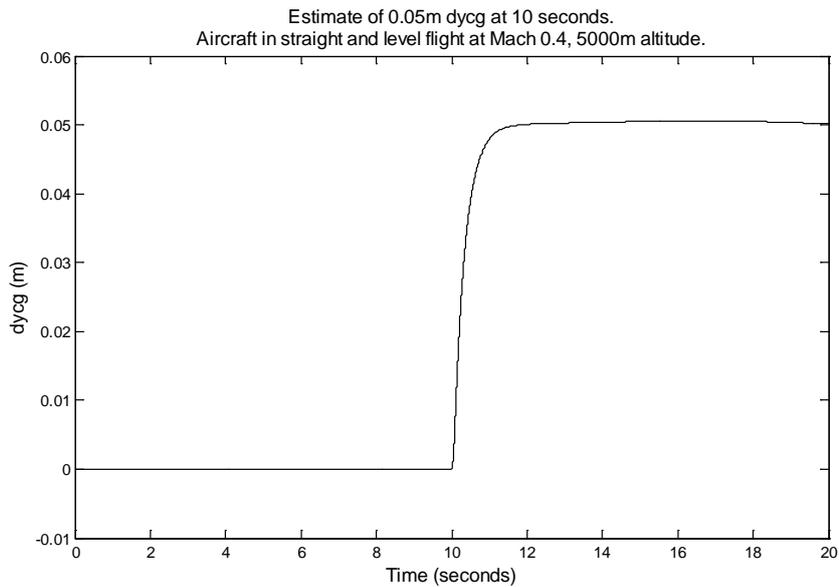
Figure 30: Estimate of 0.05m change in lateral cg. No measurement noise.

Figure 31 shows the estimate for a shift in lateral cg driven by a sine wave. The aircraft is trimmed at Mach 0.5 at 4000m altitude. As before measurement noise has not been added. The estimate of lateral cg can be seen to follow the actual change of lateral cg with a short delay.

69
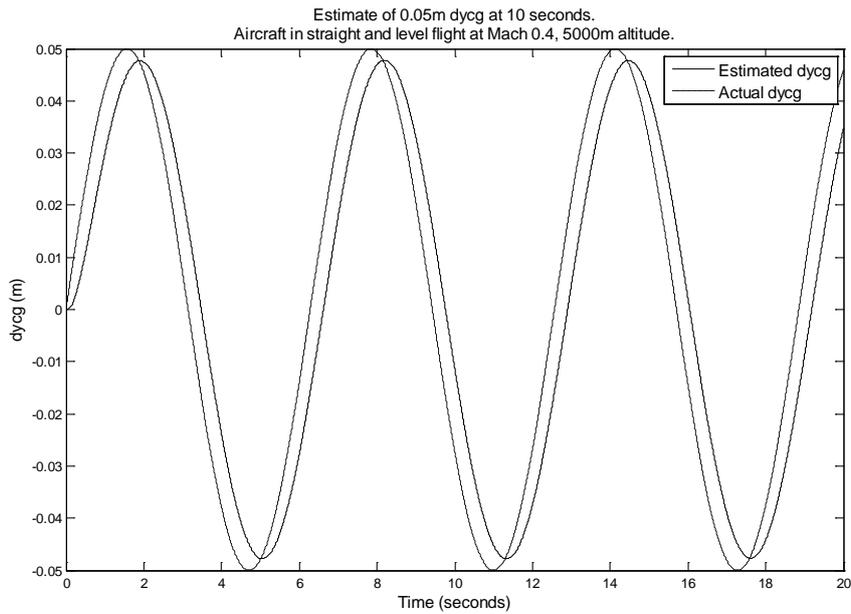
Figure 31: Estimate of sine wave lateral cg change.

The next set of diagrams provide results when the aircraft is trimmed and linearised at different speeds, and the estimator is also configured for the same speed.

Figure 32 shows estimates for a 0.05m shift in lateral cg when the aircraft is at 2000m altitude and at various Mach (0.25, 0.35, 0.45, 0.55).
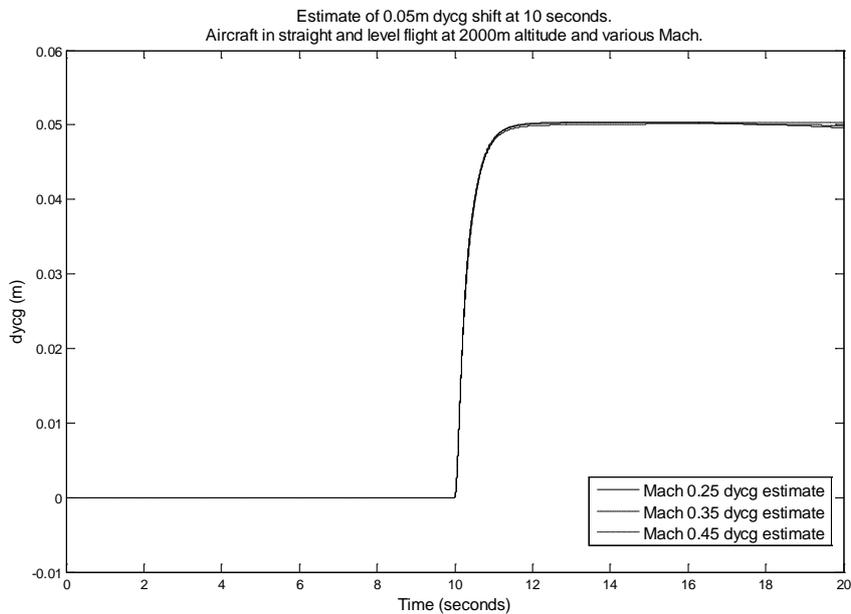
Figure 32: Estimate of lateral cg change. Aircraft at Mach 0.25, 0.35 and 0.45 at 2000m altitude.

Figure 33 shows estimates for a 0.05m shift in lateral cg when the aircraft is at 3000m altitude and at various Mach (0.55, 0.65, 0.75).
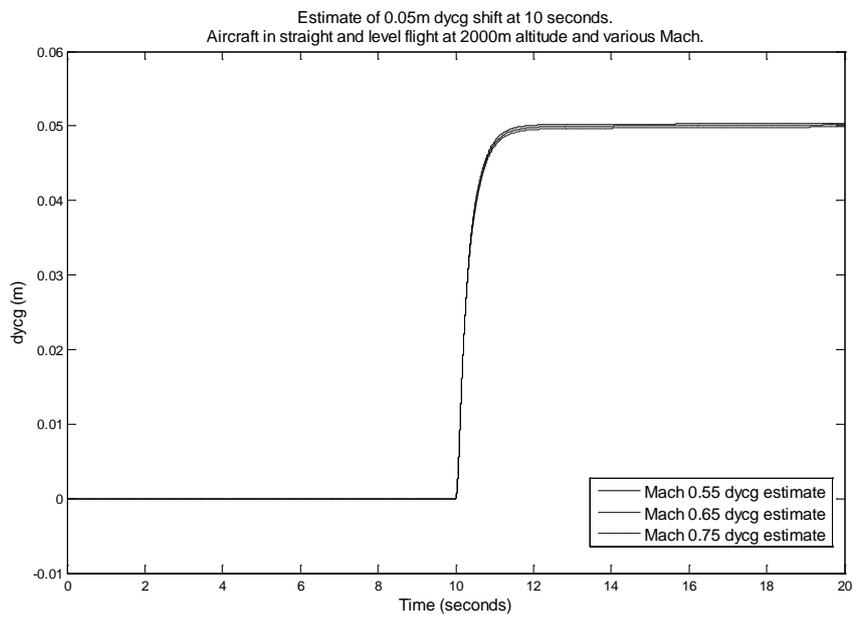
Figure 33: Estimate of lateral cg change. Aircraft at Mach 0.55, 0.65 and 0.75, at 3000m altitude.

Finally Figure 34 shows estimates for a 0.05m shift in lateral cg when the aircraft is at 3000m altitude and at high speed, Mach (0.85, 0.95, 1.05, 1.15).
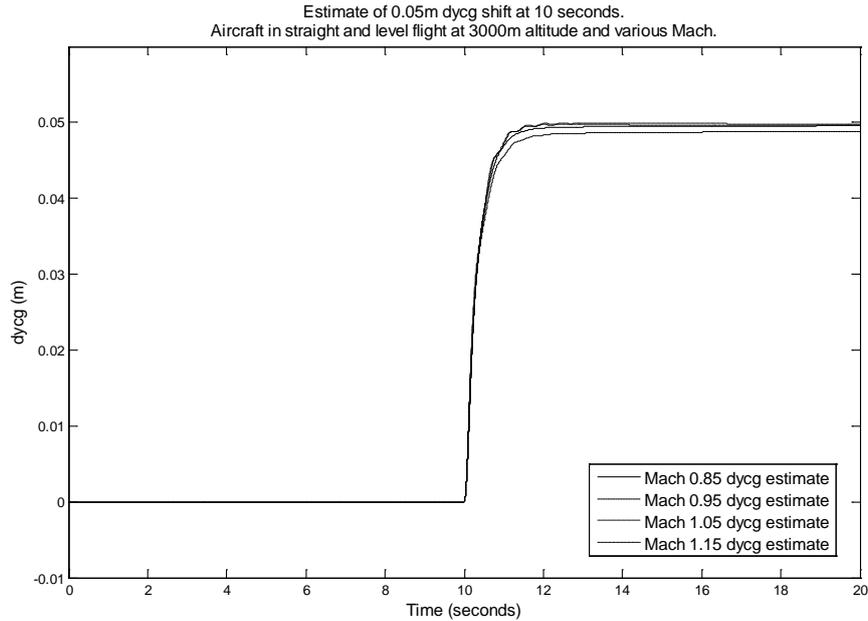
Figure 34: Estimate of lateral cg change. Aircraft at Mach 0.85, 0.95, 1.05 and 1.15, and at 3000m altitude.

The results from the lateral cg estimator have shown greater accuracy and stability at different speeds compared to the longitudinal cg estimator. This is because ADMIRE is a model of an unstable aircraft and therefore maintaining a stable pitch is a difficult process for the FCS, small changes in the aircraft angle of attack can lead to large non-linear changes in aircraft performance.

5.4 Summary

A complex non-linear aircraft model, ADMIRE, has been used with the cg estimator. The equations to estimate the cg change from unexpected angular acceleration have been developed. Apart from the addition of extra aircraft states the basic structure of the Kalman-Bucy filter is unchanged from the linear aircraft model estimator described in section 4. The overall cg estimator has been expanded to handle the addition of the aircraft forces.

The results have shown that the augmented Kalman-Bucy filter can still accurately estimate changes in angular acceleration and hence the change in cg. The use of a non-linear aircraft model and the addition of forces such as nz have had only a minor effect upon the estimator accuracy, although there is a noticeable error in longitudinal cg accuracy when the aircraft is close to the speed of sound.

When tuning the Kalman-Bucy filter the R matrix was set to the measurement noise variance whilst the Q matrix was generated on a trial and error basis starting from sensible initial values based upon the maximum expected measurement change. The key components in the Q matrix were the values for the augmented unexpected angular acceleration entry and expected angular acceleration entry i.e. the ones for 'q' and $\dot{q}_{unex}$ and 'p' and $\dot{p}_{unex}$.

However there are a number of limitations with the cg estimator developed in this section. The cg estimator has not taken into account changes in mass or moments of inertia. In each test the estimator is configured at the same speed and altitude as the aircraft so the Kalman-Bucy filter will be an accurate model of the aircraft. The aircraft is trimmed in straight and level flight and does not perform any manoeuvres. These limitations will be investigated in the next chapter.

6. Application of a Centre of Gravity Estimator to a Manoeuvring Non-Linear Aircraft Model

This section will investigate the use of the cg estimator when the aircraft manoeuvres or is at a different speed or altitude to the one that the estimator was configured for.

The estimator described in section 5 was generated from a linear aircraft model trimmed at the same speed and altitude as the actual aircraft. Section 6.1 shows that such an estimator will not be sufficiently accurate when the actual aircraft is at a different speed or altitude compared with the estimator configured speed or altitude.

One approach to correct the estimator is to scale its coefficients based upon changes in dynamic pressure. This approach is explained in more detail in section 6.2 and it greatly improves the estimator performance but it still fails to fully satisfy the requirements specified in section 3.

In section 6.3 some of the non-linearities in the longitudinal axis are investigated and then in section 6.4 an approach to selectively modify the estimator coefficients based upon angle of attack, Mach and altitude is described. The results from some testing show that this cg estimator satisfies its requirements.

The same approach is then adopted in section 6.5 for the lateral estimator; first the non-linearities are investigated and then a cg estimator is developed to selectively modify some of the rolling moment coefficients. As before some test results are provided which demonstrate the estimator satisfying its requirements.

Section 6.6 provides some examples when the aircraft manoeuvre exceeds the requirements specified in section 3.1. Section 6.7 provides robustness test results when the estimator was deliberately made inaccurate by selectively changing key coefficients, aircraft mass or inertias.

6.1. Performance of the Longitudinal CG Estimator during aircraft manoeuvres

This section provides the results when the estimator described in section 5.2 is used with an aircraft that performs different manoeuvres (pitch, roll, lateral and longitudinal acceleration), or is trimmed in straight and level flight at a different speed from that used in the estimator model.

Figure 35 shows the estimator results when it is configured for straight and level flight at Mach 0.4 5000m altitude, but the aircraft is configured at different speeds. There is a 0.1m step change in the longitudinal cg after 10 seconds.

The static upper and lower limits specified in the requirements are clearly marked on the graph. The upper limit is 0.0156m (0.3% mac) above the actual cg and the lower limit is 0.0156m below the actual cg.
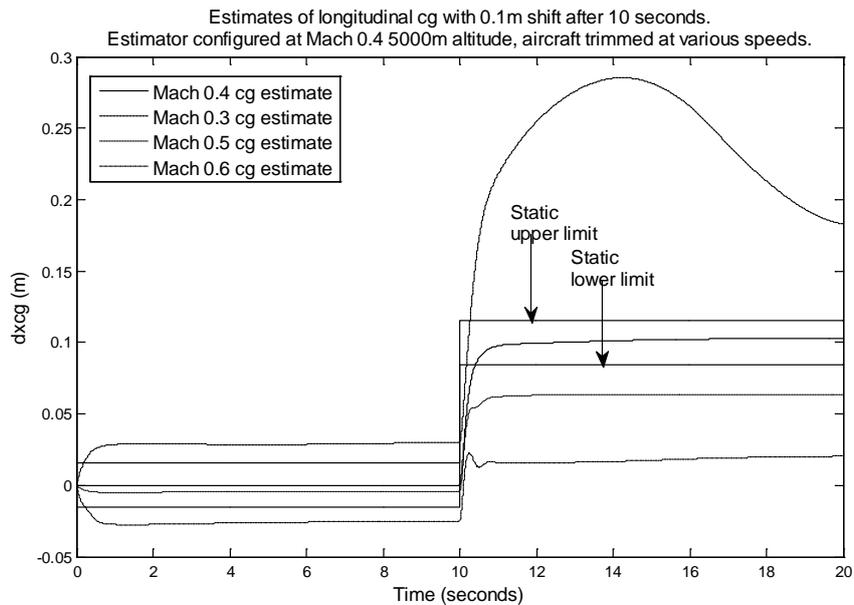
Figure 35: Estimate of change in longitudinal cg. Aircraft at various speeds, estimator configured for Mach 0.4 5000m altitude

The estimator under-estimates the cg change when the aircraft is at Mach 0.5 and Mach 0.6 but over-estimates the cg change when the aircraft is at Mach 0.3.

This is explained by the differences in dynamic pressure at the different aircraft speeds. The dynamic pressure is defined in (82) (Steven & Lewis, 2001: 63) :

$$\bar{q} = \frac{\rho}{2} V_T^2 \tag{82}$$

where

$\bar{q}$ is the dynamic pressure

V_T is the freestream airspeed

ρ is the mass density

The dynamic pressure affects the pitching moment, see (65). Therefore if the aircraft is at Mach 0.5 but the estimator is configured at Mach 0.4 then the corrective action by the control surfaces on the aircraft will be smaller because they have more effect due to the higher dynamic pressure, and therefore the estimate of dxcg will be correspondingly smaller.

For completeness the test results when the aircraft is configured at the same speed as the estimator and undergoes different manoeuvres are included in Appendix E. An example of the results from an acceleration manoeuvre are shown in Figure 36.

Estimate of 0.1m dxcg at 10 seconds. Aircraft in straight and level flight at Mach 0.4, 5000m altitude. Aircraft accelerates after 11 seconds.
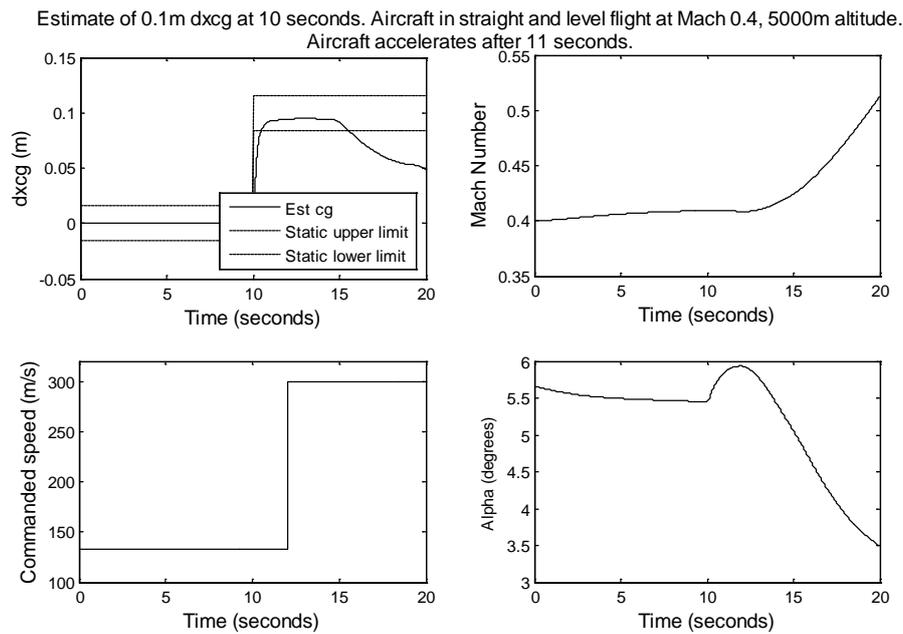
Figure 36: Uncorrected estimator. Acceleration test.

This section has shown that the estimator designed in Section 5 is inadequate when the aircraft manoeuvres or is at a different speed to the one that the estimator was configured for. The next sub-section investigates the aircraft coefficients and how the estimator may be modified to give acceptable results.

6.2. Scaled Longitudinal CG Estimator based upon Dynamic Pressure

One approach that can be used when estimating non-linear systems is to employ an Extended Kalman Filter (EKF). An EKF is essentially the same as a Kalman filter except that the filter is continuously re-linearised around the state estimate. The EKF has found use in applications such as navigation or GPS, however there are known problems with it regarding stability and divergence, see Bar-Shalom et al (2001: 385), and it is also computationally much more demanding, typically by two orders of magnitude. Since there are known difficulties

implementing an EKF, and given the restricted manoeuvre range requirements then alternative approaches were investigated to obtain an estimator that still satisfied the requirements.

The previous section suggested that the errors in the estimator at different speeds may be due to changes in the dynamic pressure $\bar{q}$. It is fairly straightforward to modify the estimator to scale its inputs based upon changes in the dynamic pressure to test such an approach. The cg estimator described in section 5 was modified to scale the inputs to the Kalman-Bucy filter by the dynamic pressure ratio, which is the actual aircraft dynamic pressure divided by the dynamic pressure that the estimator was configured at, see (83). In practice the dynamic pressure signal will be available from the aircraft's Air Data system.

$$\bar{q}\ ratio = \frac{aircraft\ \bar{q}}{estimator\ \bar{q}} \tag{83}$$

Figure 37 provides a diagrammatic representation of the modified estimator.
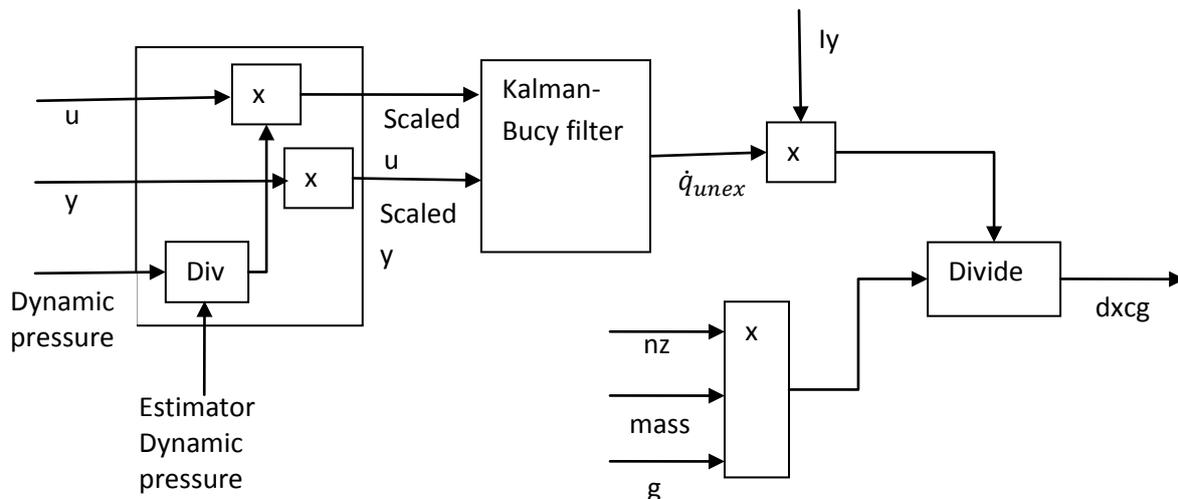
Figure 37: Longitudinal cg estimator using scaled inputs

The scaling was applied to all commands 'u', except the 'tss' thrust command because its affect upon the pitching moment is unaffected by the dynamic pressure. The scaling was also applied to the alpha measurement because the AoA has the dominant effect upon the pitching

moment. Note that the tests were also repeated using additional aircraft measurements but they had little effect upon the results.

Figure 38 illustrates the results from this approach. As before the estimator is configured for straight and level flight at Mach 0.4 5000m altitude and the aircraft is then configured in straight and level flight at different speeds. A longitudinal cg shift of 0.1m is applied after 10 seconds. The red lines on the graph denote the static upper and lower limits defined in the requirements in section 3.1.
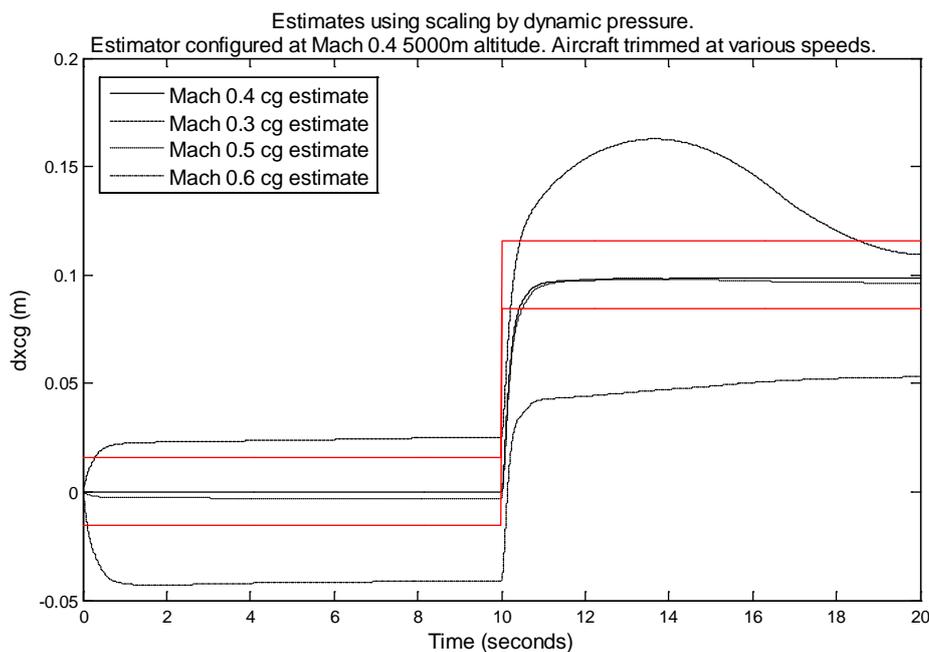
Figure 38: Estimates of dxcg with correction for changes in dynamic pressure ratio

The results show an offset in the estimate which is particularly noticeable at Mach 0.6 and Mach 0.3. Note that no correction was made for the delta measurements (i.e. measured values minus trim point values) and commands (i.e. commands minus trim point values) used in the estimator, for example at Mach 0.6 the estimator would input a large delta value for velocity V_T. As before the static upper and lower limits are shown by the red lines in the graphs.

A correction was then made to obtain more accurate delta commands and measurements with the estimator. The initial values for the commands and measurements had been obtained from trimming the aircraft at different Mach numbers, and these values were interpolated by a lookup table based upon aircraft speed. The difference between these values and the actual aircraft commands and measurements were then scaled by the dynamic pressure ratio method

described earlier. The results from this correction are shown in Figure 39. The additional correction virtually removed the estimation offsets, but at Mach 0.3 the estimate is still unacceptable because it exceeds the acceptable upper and lower limits, and at Mach 0.6 the estimate takes too long to converge within the static lower limit.
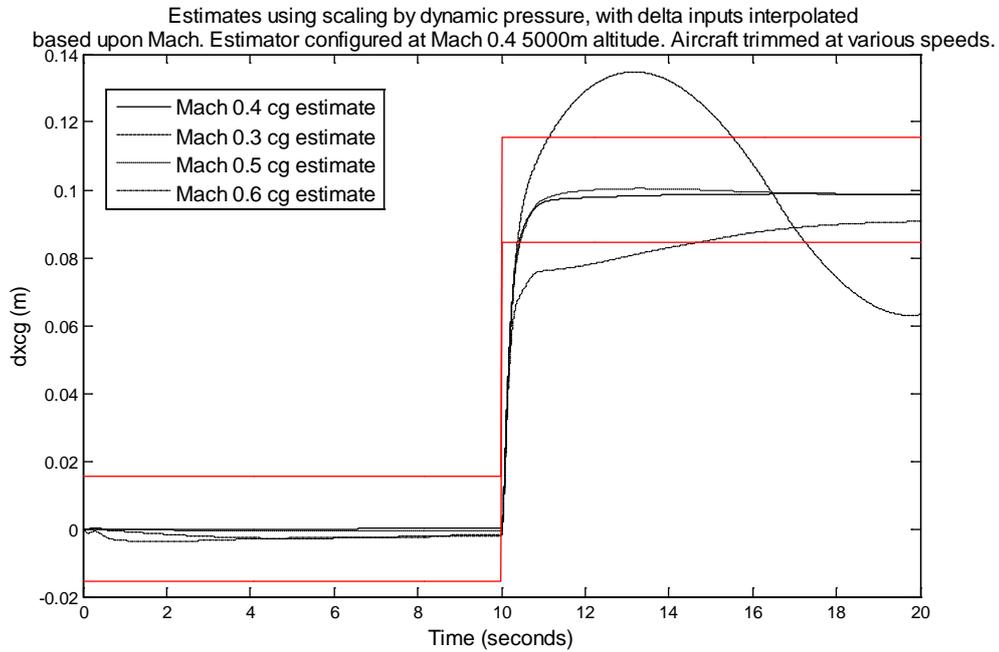
Figure 39: Estimates of dxcg with aircraft trimmed at various speeds. Estimator using interpolated trim points and scaling inputs based upon dynamic pressure ratio

The correction using dynamic pressure has greatly improved the cg estimate when the aircraft is in straight and level flight but at different speeds to the estimator configuration, but the results are still unacceptable. When the aircraft manoeuvre requirements are tested the results are worse. Figure 40 provides an example when the push-down manoeuvre is implemented, as before the top left graph provides the cg estimate.
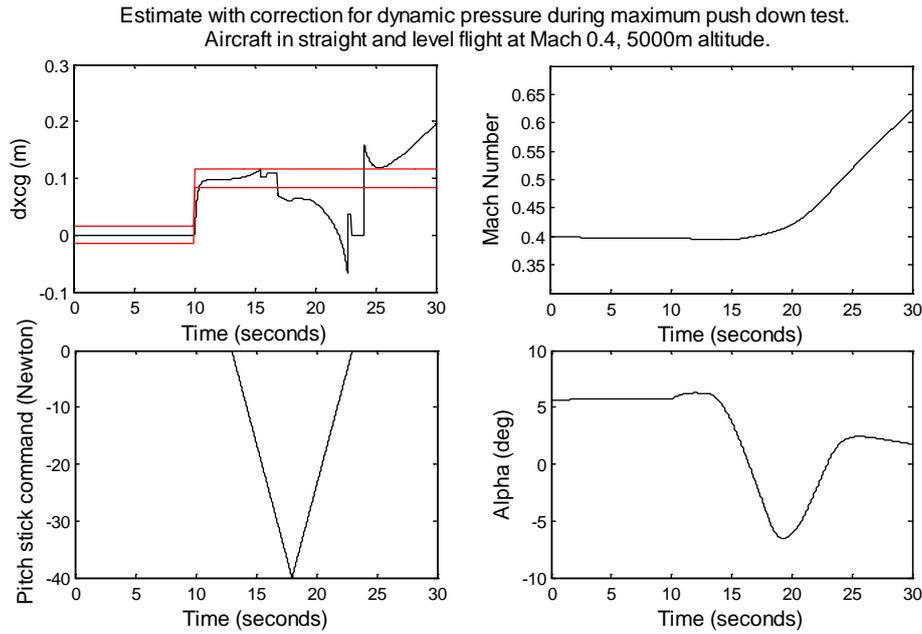
Figure 40: Longitudinal cg estimate during maximum push down manoeuvre. Estimator corrected for dynamic pressure and interpolated trim points.

The main reason the estimate in Figure 40 is so bad is due to the change in the angle of attack (AoA). The relationship between the AoA and the pitching moment is investigated in the next section.

6.3. ADMIRE Pitching Moment Non-Linearities

Figure 41 shows the value of Cm_α at various AoA when the aircraft is trimmed for flight at Mach 0.4 and at 5000m altitude. The data is obtained by trimming the aircraft at different AoA and then obtaining a linear model. For example at 8° AoA the value of Cm_α (pitching moment coefficient derivative with respect to α) is approximately 8 and at 4° AoA the value of Cm_α is approximately 4.8. In other words a change in the AoA at 8° has nearly double the effect on the pitching moment compared to when the aircraft is at 4° AoA.
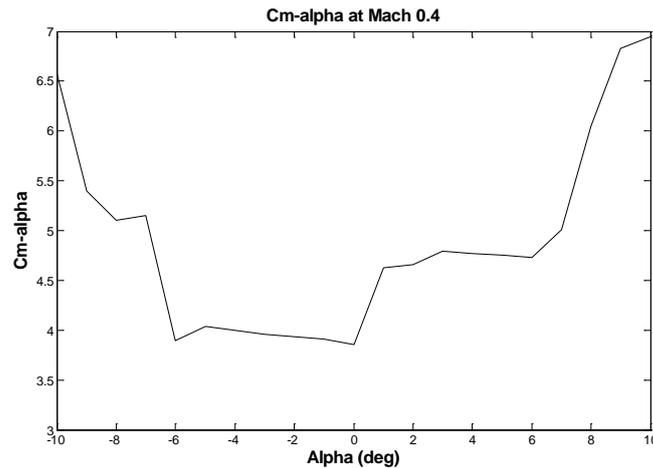
Figure 41: Cm_α at Mach 0.4 and various angles of attack

Figure 42 gives a more complete representation of some of the non-linearities in the pitching moment of the ADMIRE model, in the AoA range of +/- 12°. Identification of the non-linearities in the aircraft pitching moment coefficient values will help to determine the design of a new cg estimator.

The value of the Cm_{vt} coefficient is small and therefore although it does change for AoA the changes are unlikely to have a big effect upon the estimator. The change in Cm_{canard} is more significant but a linear interpolation between the values at -12°, 0° and 12° should provide reasonably accurate values. The coefficient Cm_q is linear (i.e. unchanged) for regions of AoA -12° to 4°, and between 4° to 8° and then 9° to 12° the change in value could be approximated by interpolation between data point set at 4°, 8° and 12°. The coefficient values for the elevons are very variable but this variability is over a small range in values, they would require data points at -12°, -10°, -7°, 2°, 3°, 6° and 12° to obtain accurate values. The effect of the elevon non-linearities on $\dot{q}$ depends upon the scale of the elevon command. For example the inner elevon coefficient derivative has the value of -2.3 at -10° AoA but at 6° AoA it has the value of approximately -2.45. Therefore the estimator could potentially contain a derivative error of -0.15 if the value is not corrected, and if the inner elevon was at 10° (0.1745 radians) then the error introduced into the estimate of the expected value of $\dot{q}$ is 0.026 radians/s^2 which will give a cg estimate error of 0.003m (calculated from (70) using the ADMIRE values for mass and I_y).

81

Aircraft coefficients at Mach 0.4, 5000m altitude in straight and level flight
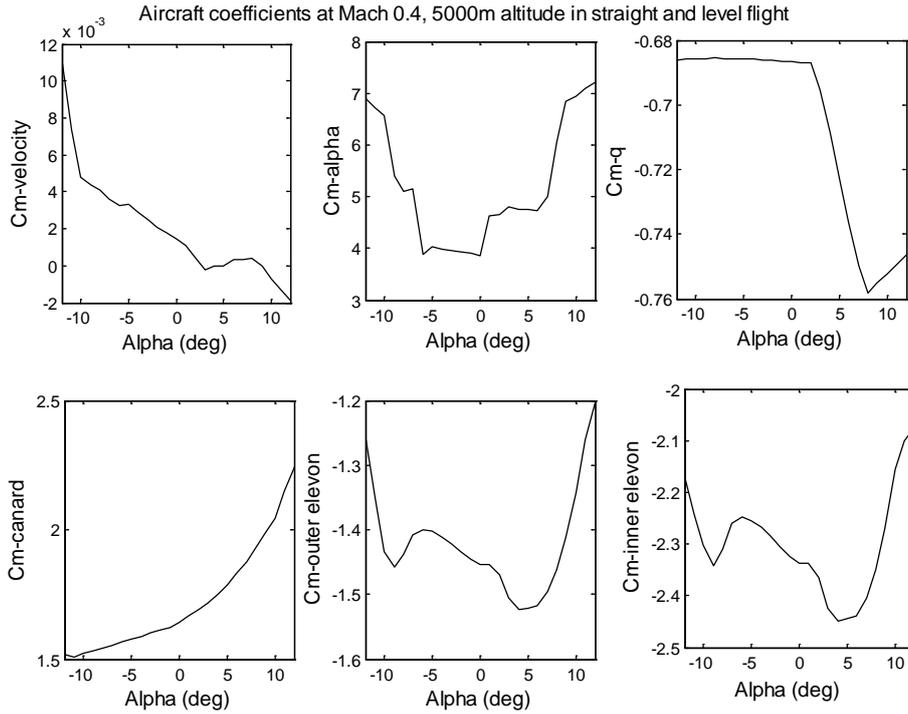
Figure 42: ADMIRE aircraft pitching moment coefficients at various AoA

The approach used in section 6.2 scaled the estimator for changes in dynamic pressure. It is now explained why this approach was insufficient.

Figure 43 shows the coefficient derivatives for Cm_α at various speeds and AoA.

Cm-alpha coefficient at various speeds when aircraft trimmed in straight and level flight at 5000m altitude
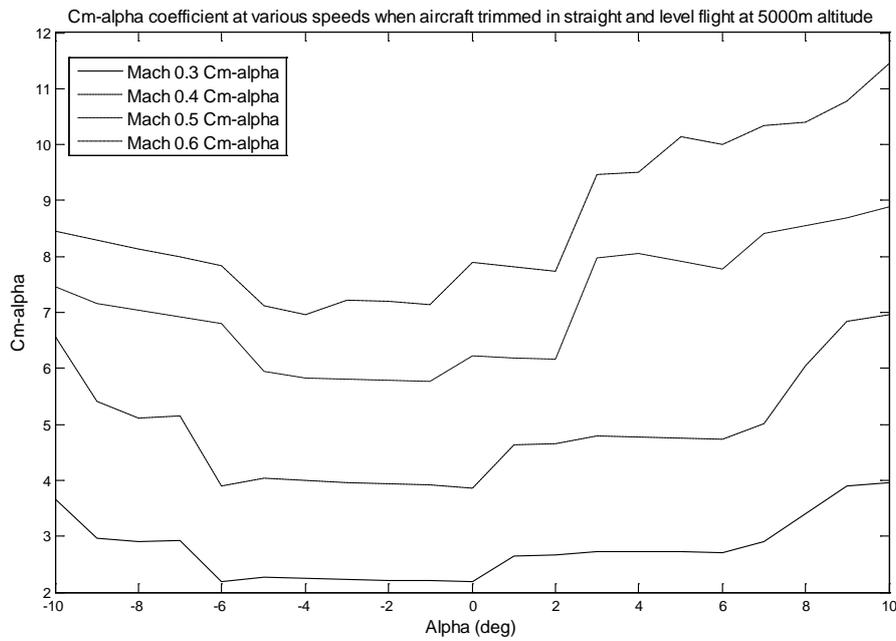
Figure 43: ADMIRE Cm_α coefficient at various Mach and Aoa

82

The changes in Cm_α for AoA at Mach 0.3 and Mach 0.4 are very similar apart from the scale of the coefficient derivative, including the fact that they share the same "kink points", -7°, -6°, 0°, 1°, 6°, 9°. However at Mach 0.5 and Mach 0.6 the "kink points" are -6°, -5°, -1°, 0°, 2°, 3°, 6°, 10°. The pitching moment is largely caused by Cm_{tot} which is scaled by the dynamic pressure $\bar{q}$ see (65), the other terms S_{ref} and C_{ref} are constant. However it is not the case that at different speeds the variation in Cm_α seen in Figure 43 is only caused by changes in dynamic pressure.

Table 11 shows the ratio of Cm_α at different speeds and AoA compared with the Mach 0.4 value when scaled for the change in dynamic pressure. For example at Mach 0.3 and at an AoA of -10° the Mach 0.3 value for Cm_α is 99.4% of the M0.4 value when scaled for the change in dynamic pressure, however at Mach 0.5 Cm_α is only 72.6% of the value it should be if the M0.4 value is scaled up by the change in dynamic pressure.

Table 11: Variation in Cm_α when scaled by dynamic pressure normalised to value at Mach 0.4

AoA (deg)	M0.3	M0.4	M0.5	M0.6
-10	0.99	1	0.73	0.57
-9	0.98	1	0.85	0.68
-8	1.01	1	0.88	0.71
-7	1.01	1	0.86	0.69
-6	1.00	1	1.12	0.89
-5	1.03	1	0.94	0.78
-4	1.00	1	0.93	0.77
-3	1.00	1	0.94	0.81
-2	1.00	1	0.94	0.81
-1	1.00	1	0.94	0.81
0	1.01	1	1.03	0.91
1	1.01	1	0.86	0.75
2	1.02	1	0.85	0.74

3	1.01	1	1.06	0.88
4	1.02	1	1.08	0.89
5	1.02	1	1.07	0.95
6	1.02	1	1.05	0.94
7	1.03	1	1.08	0.92
8	1.00	1	0.90	0.76
9	1.01	1	0.81	0.70
10	1.02	1	0.82	0.73

This analysis has shown that there is a relationship between the pitching moment, AoA and aircraft speed, but the relationship is not linear. Cm_α varies very closely with the change in dynamic pressure between Mach 0.3 and Mach 0.4, but at greater speeds the value of Cm_α is generally less than would be expected if it was scaled by the ratio of the change in dynamic pressure.

It should be noted that Cm_α will also vary at the same Mach and AoA but at different altitudes. Examples are given in Table 12 when the aircraft is configured at Mach 0.4 and at an AoA of 6°.

Table 12: Cm_α at Mach 0.4, 6° AoA at different altitudes.

Altitude	Cm_α
20	8.82
1000	7.82
3000	6.13
5000	4.73

The conclusion from this section is that modifying the estimator to correct the inputs for changes in dynamic pressure is insufficient to give reliably accurate results, and that the estimator needs to take into account changes in speed, AoA and altitude to remain accurate.

6.4 Longitudinal CG Estimator – Selective Coefficient Correction

This section develops an estimator that uses speed, AoA and altitude to modify the pitching moment coefficients in the Kalman-Bucy filter, i.e. as the aircraft manoeuvres the pitching moment coefficients are modified to more accurately represent the manoeuvring aircraft.

The calculation of the coefficients is designed to capture as many of the aircraft non-linearities as is practical. As noted earlier the standard approach to this problem would be to use an EKF, but given the known problems with an EKF then this alternative approach is investigated. Also the project sponsors BAE Systems had a preference for some sort of scheduled estimator instead of an EKF because this would simplify the process of getting the estimator accepted by the certification authorities.

For the longitudinal cg estimator Cm_α is used to select the data points since the AoA is the dominant aircraft characteristic affecting the pitching moment. The selected data points will then be reassessed depending upon estimator performance.

The key AoA data points to capture changes in Cm_α at Mach 0.3 and Mach 0.4 occur at -7°, -6°, 0°, 1°, 6°, 9° and 12° (see section 6.3).

For simplicity AoA points below -7° and above 12° are not considered. A more useful estimator would cover a far greater AoA range but the purpose here is to investigate a suitable estimation correction mechanism for changes in AoA rather than develop a complete system.

A set of derivative data was obtained at combinations of points in the flight envelope, as defined by Table 13.

Table 13: ADMIRE cg estimator data points

AoA (degrees)	[-7, -6, -1, 0, 1, 2, 3, 6, 9, 12]
Mach	[0.2 0.3 0.4 0.5 0.6]
Altitude (m)	[20 1000 3000 5000]

The data points in Table 13 were selected to reduce the effect of the non-linearities in the aircraft. For example at Mach 0.4 Cm_α is fairly constant at an AoA between 0° and -6°, and between 3° and 6°, see Figure 42, so these data points were selected. However when nz

approaches zero this magnifies any estimation error, see (70), therefore additional data points were selected at low AoA to reduce this error.

The derivative data is used to correct the Kalman-Bucy pitching moment coefficients detailed in Table 14. These coefficients were chosen because they had the largest effect upon pitch acceleration. A subset of this table data is contained in Appendix H.

Table 14: ADMIRE cg estimator corrected coefficients

Commands	Cm_{dlc} Cm_{drc} Cm_{dloe} Cm_{droe} Cm_{dlie} Cm_{drie} Cm_{tss}
Measurements	Cm_α Cm_{vt} Cm_q

The corrected coefficients are obtained by linear interpolation from a look up table using Mach, AoA and altitude as inputs. Obviously any remaining non-linearities between data points will be missed with this method. The delta inputs into the estimator are similarly obtained from the difference with an interpolated value for the trim point.

Since the estimator has been configured when the aircraft is in level flight it will not accurately estimate the effect of 'p' and 'r' on the pitch acceleration, see (64). Therefore a correction is applied to $\dot{q}_{unex}$ to compensate for this:

$$pr\_correction = C_5 \cdot p_b \cdot r_b - C_6(p_b^2 - r_b^2) \tag{84}$$

The overall structure of the estimator is shown diagrammatically in Figure 44.
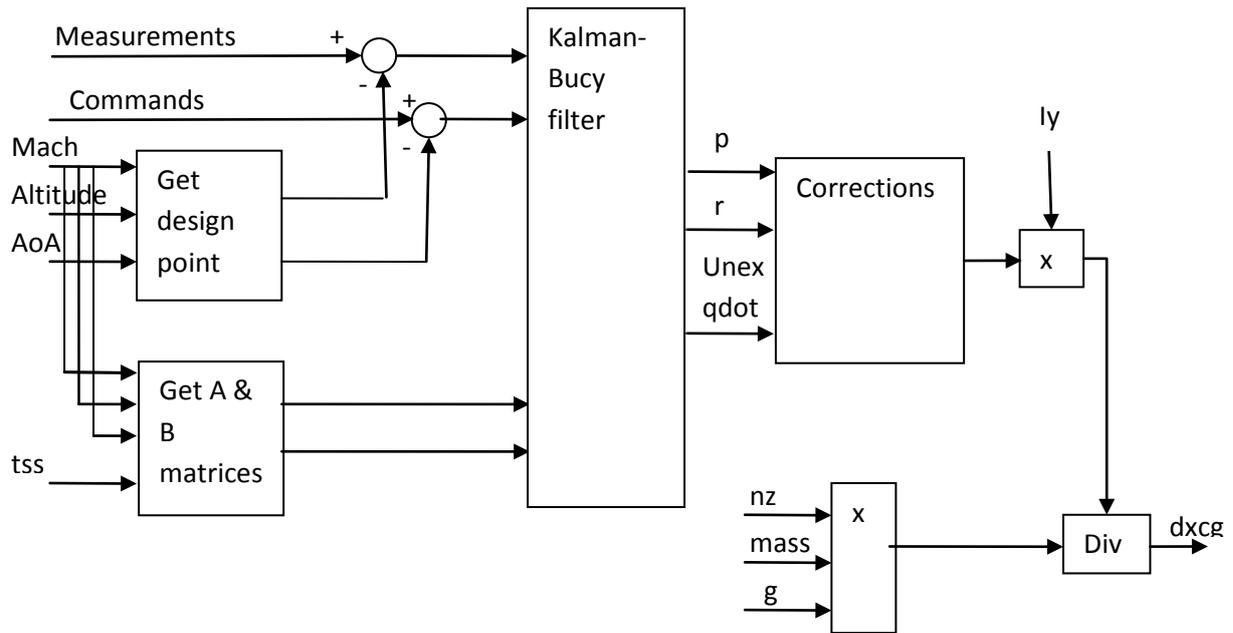
Figure 44: Longitudinal cg estimator structure

The output from the Kalman-Bucy filter is $\dot{q}_{unex}$ which is then corrected for changes in 'p' and 'r' and pitch acceleration (Figure 45), scaled by I_y and then divided by vertical acceleration nz, 'g' and the mass of the aircraft.

The "Get A & B matrices" block uses Mach number, altitude and angle of attack to linearly interpolate values for the 10 coefficients defined in Table 14. The 7 command coefficients are inserted into the B matrix in the Kalman-Bucy filter and the 3 measurement coefficients are inserted into the A matrix of the Kalman-Bucy filter. These inserted coefficients will then be used inside the filter to generate a more accurate estimate for $\dot{q}$.

The "Get design point" block in Figure 44 is used to obtain an interpolated trim point which is subtracted from the aircraft commands and measurements to get the delta values to be used in the cg estimator. However one effect of this block is that when the aircraft pitches up or down then the interpolated value for the q trim point reduces the delta value used by the estimator. The result of this is that incomplete information about the true value of pitch acceleration is received by the estimator. A correction for this error is included in the corrections block shown in Figure 44 and expanded in Figure 45. The qtrim-dot estimator estimates the pitch acceleration that is excluded from the measurements due to the use the interpolated trim points.
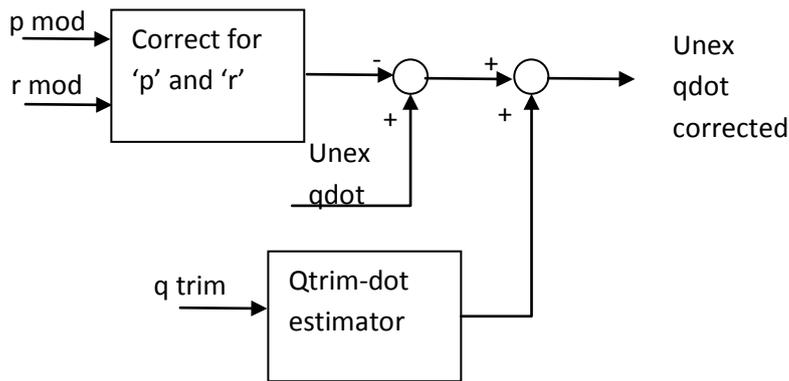
Figure 45: Longitudinal cg estimator corrections

An example is given in Figure 46 of the effect of interpolating the q trim points on the delta q value which is input into the estimator. In the example the aircraft pitches down from an AoA of 6° to -6° then back up during a push-down manoeuvre.
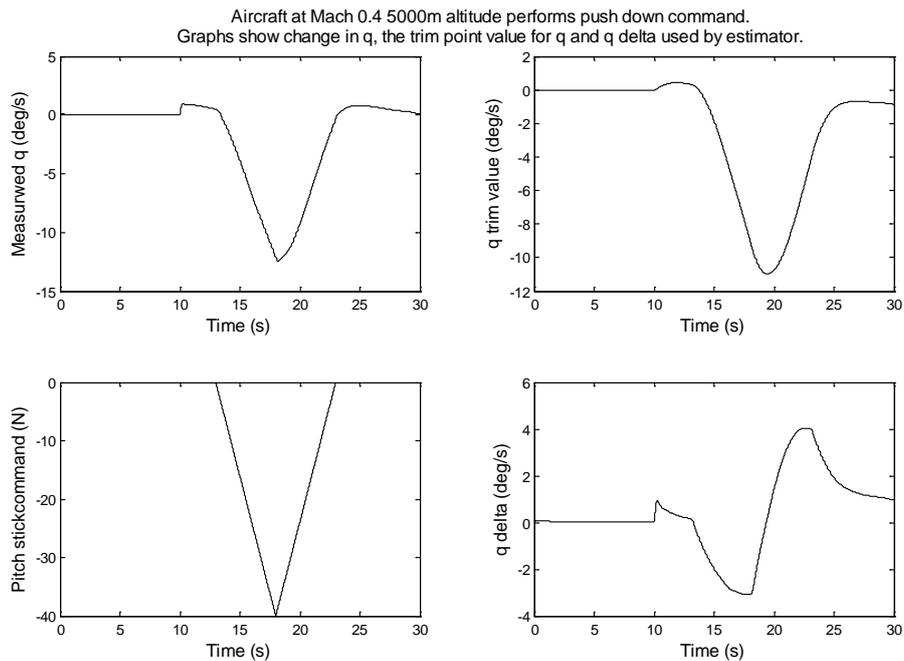
Figure 46: Effect of change in AoA on the measured 'q' and delta 'q' used by the cg estimator.

Figure 46 shows the aircraft pitch rate when it is in straight and level flight, then pitches down by about 13 °/second before returning to a zero pitch rate (see top left graph). The trim values for q (obtained from "Get design point") are shown in the top right graph and display a

similar change, therefore the delta values of q used in the estimator are relatively small (see bottom right graph in Figure 46).

When the estimator makes its estimate for $\dot{q}_{unex}$ it does not correctly estimate $\dot{q}$ due to the changing q_{trim}. Therefore the estimated $\dot{q}_{unex}$ should be the difference between the expected value for $\dot{q}$ ($\dot{q}_{exp}$) and the sum of the delta measurement input $\dot{q}_{delta}$ and $\dot{q}_{trim}$.

$$\dot{q}_{unex} = \dot{q}_{delta} + \dot{q}_{trim} - \dot{q}_{exp} \tag{85}$$

A simple Kalman-Bucy filter was used to obtain $\dot{q}_{trim}$ in a two state filter containing $[q_{trim}, \dot{q}_{trim}]$, and $\dot{q}_{trim}$ was used to correct the estimate of $\dot{q}_{unex}$.

The Q and R matrix values in the Kalman-Bucy filter to obtain $\dot{q}_{trim}$ are given in (86).

$$R_{trim} = [0.0000076]$$
$$Q_{trim} = \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix} \tag{86}$$

The values in the Q matrix were obtained by trial and error, the R matrix value matches the noise variance for q identified in Table 8.

When nz approaches zero it will increase any estimation error of $\dot{q}_{unex}$ because nz is used in the division in (70) to obtain dxcg. Therefore the cg estimator includes a check to prevent divide by zero, and holds the value of dxcg when nz is between +0.4 g and -0.4 g to reduce the magnified error. The choice of 0.4 g was obtained from experimentation. A small value causes a greater dxcg error whereas a large value leads to a greater time when the cg estimate is not updated.

Table 15: Maximum dxcg estimation error for different nz hold values

nz hold value (+/-)	Maximum dxcg error (m)
0.1	-0.0423
0.2	0.028
0.3	0.016
0.4	0.0102
0.5	0.007

Table 15 contains the maximum estimation error obtained when the maximum push down manoeuvre was performed with a change in longitudinal cg of 0.1m. The results were obtained using the coefficient correction estimator that is described later in this chapter.

Note that the thrust command (tss) is shown as a separate input in Figure 44. A simple modification to Cm_{tss} was implemented to estimate the pitching effect when the afterburners were engaged, which occurs when tss is greater than 0.8.

Figure 47 shows engine thrust generated from the thrust command tss, and shows that engine thrust increases more sharply when tss is greater than 0.8. Since the estimator is generated from trim data points when the afterburner is inactive it will underestimate thrust at large values of tss. The correction for the active afterburners in the estimator is a gain applied to the interpolated value of Cm_{tss}.
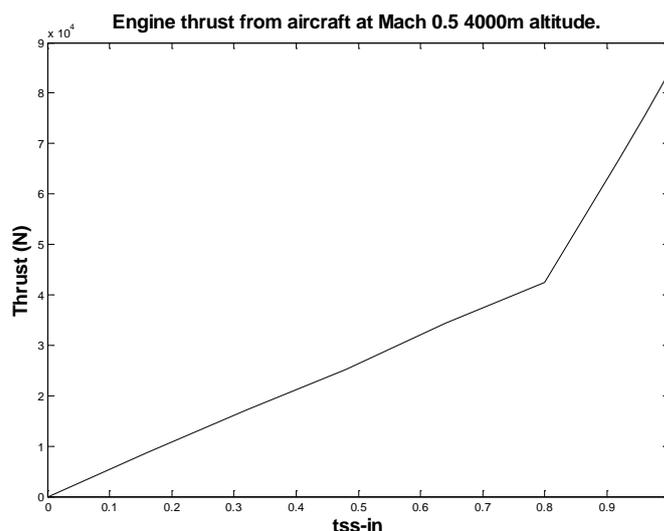
Figure 47: Engine thrust from 'tss' command. Aircraft at Mach 0.5 and 4000m altitude.

The measurement noise added to the ADMIRE aircraft model is defined in Table 8 – obtained from representative flight test data as described in section 5.2. The noise variance is used to populate the R matrix of the Kalman-Bucy filter.

The Q matrix in the Kalman-Bucy filter was obtained by trial and error from a sensible initial set of values (which were based upon the values of the average measurements) :

$$Q = diag([10\ 0.02\ 0.01\ 0.01\ 0.01\ 0.01\ 0.01\ 0.01\ 0.01\ 10\ 0.2]) \tag{87}$$

6.4.1 Modified Longitudinal CG Estimator - Results

This section displays sample results for the different aircraft manoeuvres specified in section 3.1, using the longitudinal cg estimator. Each aircraft manoeuvre occurs when the aircraft is trimmed at Mach 0.4 and at 5000m altitude, and after 10 seconds there is a cg shift of 0.1m. Note that for clarity measurement noise has not been added to the signals making the results less noisy than they would otherwise be.

In all of the figures the cg estimate is in the top left graph and the static upper and lower limits are marked on the graph. The other graphs show the aircraft commands or relevant aircraft data pertaining to the manoeuvre.

6.4.1.1 Acceleration

The BAE acceleration manoeuvre requirement (section 3.1) is that the cg estimator copes with aircraft acceleration of +/- 0.03 Mach/second.

In this test the aircraft accelerates from Mach 0.4 to Mach 0.75, see Figure 48. Note that the cg estimate drifts off at speeds above Mach 0.6 because the estimator is only configured with data up to a maximum speed of Mach 0.6, see Table 13. The cg estimate remains within its static error limit. The other graphs show the change in nz, Mach and commanded thrust. Note that the engine afterburners are activated after about 15 seconds. The rate of acceleration is about 0.02 Mach/second, which is the maximum acceleration that was achieved in straight and level flight.
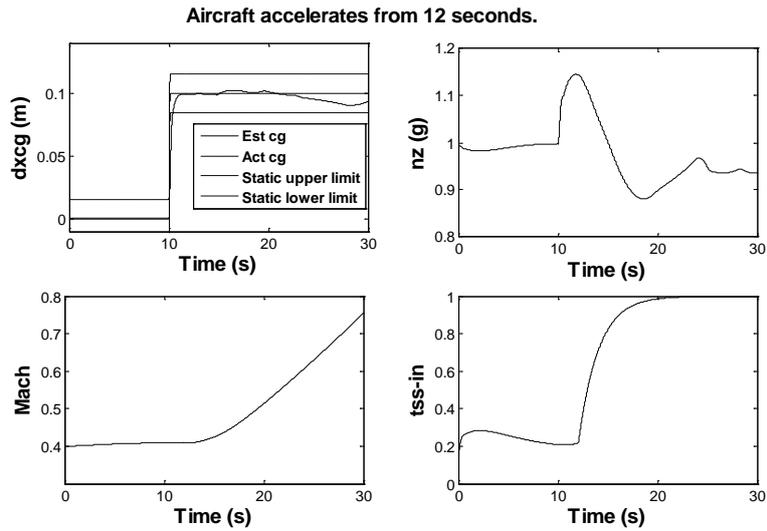
Figure 48: Acceleration test

6.4.1.2 Lateral Acceleration

The BAE lateral acceleration manoeuvre requirement (section 3.1) is that the cg estimator copes with lateral acceleration of +/- 1.5m/s^2.

In this test a rudder command occurs after 11 seconds and generates a value for vdot (lateral acceleration) of approximately -1.4 m/s^2. The effect upon the estimate of the cg is negligible, see Figure 49, and the estimate remains well within static limits.
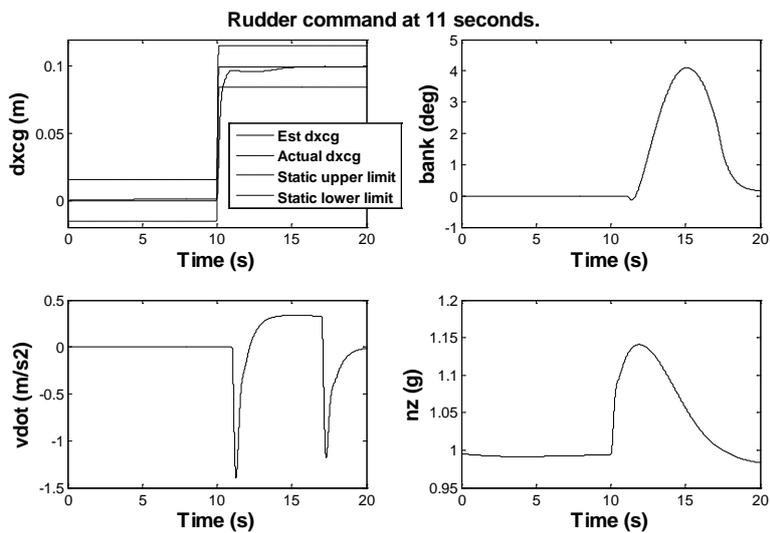
Figure 49: Lateral acceleration test

6.4.1.3 Roll

The BAE roll manoeuvre requirement is that the cg estimator copes with a roll rate of +/- 30 °/second. BAE verbally advised that a maximum bank angle of 45° should be considered.

In this test a roll is commanded after 11 seconds and the aircraft banks at a rate of approximately +21 °/second and -29 °/second, see Figure 50. The aircraft bank angle reaches about 50°. As before the cg estimate remains well within specified limits.
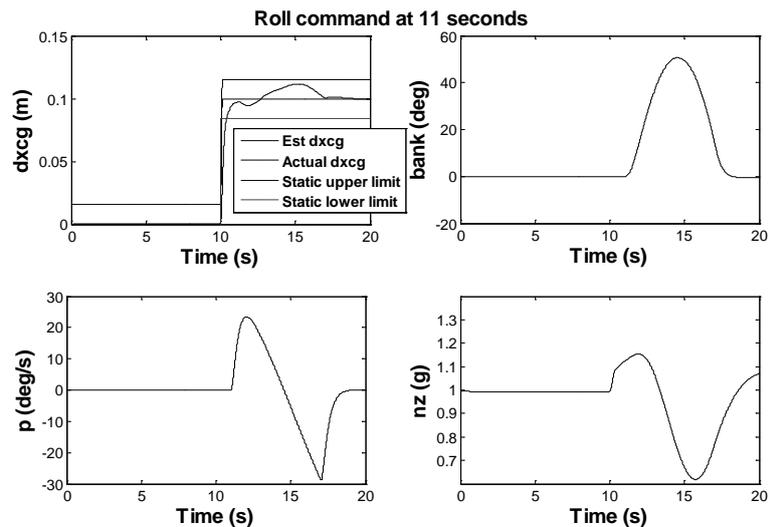
Figure 50: Roll test

6.4.1.4 Pitch down

The BAE pitching manoeuvre requirement is that the cg estimator copes with the manoeuvre when the pitch stick is pushed down to its maximum over a five second period, and then returned to its neutral position after a further five seconds.

The results from such a manoeuvre are shown in Figure 51. A maximum push down is commanded at 15.5 seconds over 5 seconds, and then the pitch stick is pulled back to level flight over another 5 seconds. This test is more demanding of the estimator because changes in the AoA have a big effect upon the pitching moment. The AoA and nz both pass through zero. Since dxcg is calculated by dividing by nz (70) then as nz decreases this magnifies any errors. The results show the cg estimate staying within its static limits.
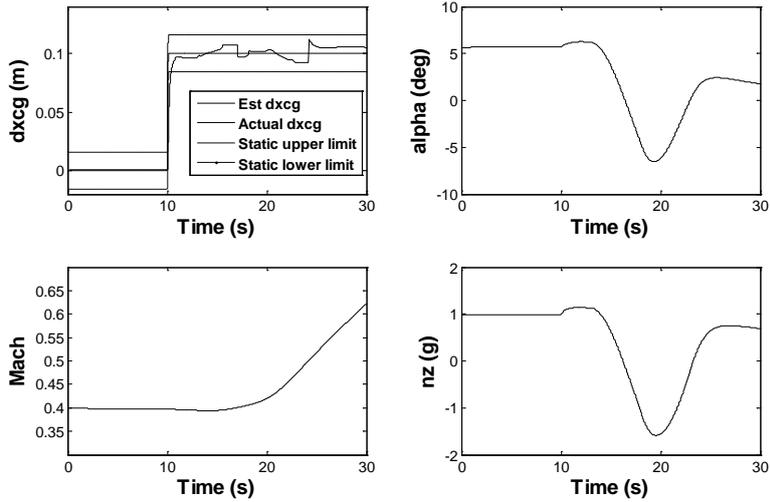
Figure 51: Pitch stick push down test

During the test the aircraft speed increases to over Mach 0.6 and altitude drops from 5000m to about 3500m (not shown in Figure 51).

The results from this section have shown that selective modification of the pitching moment coefficients in the Kalman-Bucy filter has enabled the estimator to satisfy its requirements, even when the aircraft goes through various manoeuvres.

6.4.1.5 Sample Mission

In a sample mission the aircraft accelerates from Mach 0.2 at 20m altitude to about Mach 0.5 at 4500m altitude. It then levels off and performs a couple of banked turns changing direction by about 60°, and then descends to about 700m. During this mission there are three step changes in the cg as shown in Table 16, which also shows the change in aircraft mass.

Table 16: Sample mission. Changes in cg and aircraft mass

Time (s)	dxcg (m)	Δmass (kg)
10	0.1	-400
120	-0.05	-210
180	-0.05	-210

There is a constant reduction in mass of 2kg per second to simulate fuel burn, taking no account of engine thrust. This continual reduction gives a total reduction in aircraft mass of 20% over the 500 second simulation. This is an unrealistic reduction in mass but has been used to minimise simulation time.

Figure 52 shows the cg estimate, z, mass and nz.
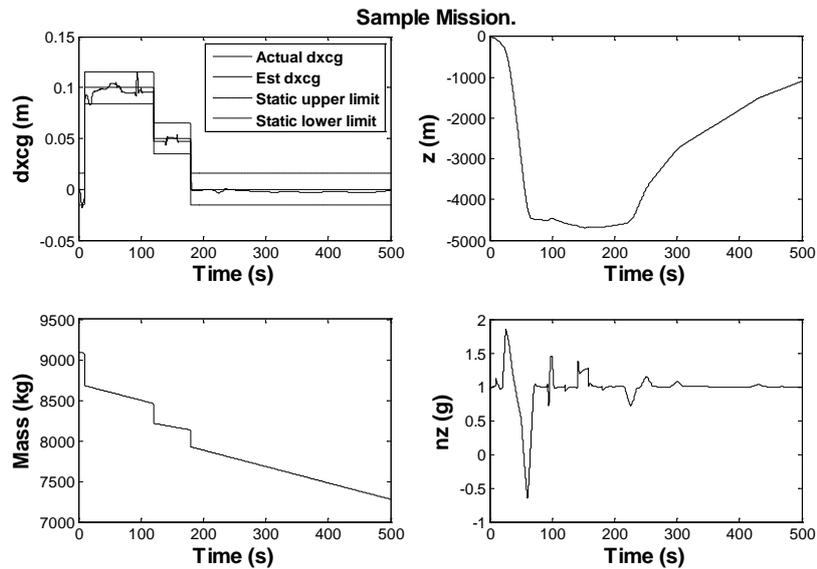
Figure 52: Sample mission (dxcg, z, mass and nz).

Figure 53 shows more aircraft data during this sample mission: Mach, AoA, yaw, bank, p and tss.
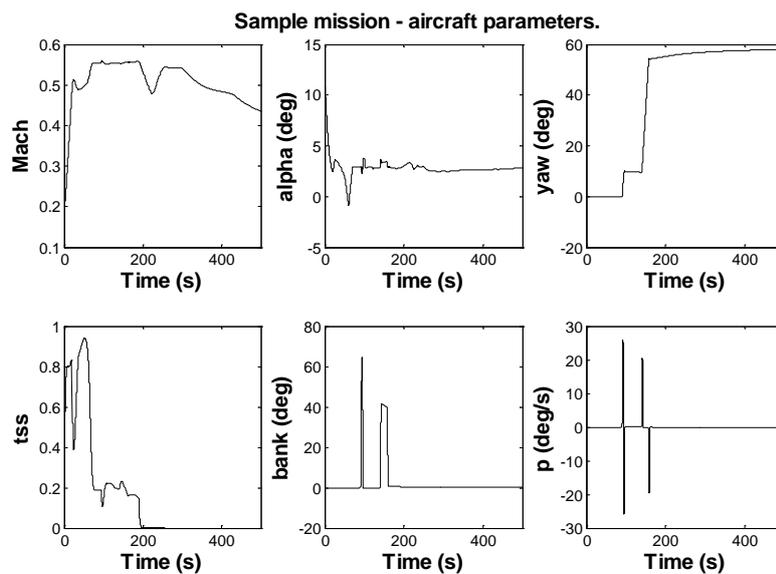
Figure 53: Sample mission (Mach, α, yaw, tss, bank, p).

At all times the cg estimate stayed within the specified dynamic limits. It strayed outside the static limits on two occasions, between 5 and 7 seconds and between 17 and 20 seconds when the aircraft was accelerating. A more accurate correction for the effect of active afterburners should improve the estimate, because in the first deviation engine thrust was increasing sharply and in the second deviation the afterburners were becoming active when tss was around 0.8.

6.5 Modified Lateral CG Estimator for Manoeuvres

6.5.1 ADMIRE Rolling Moment Non-Linearities

In this section the non-linearities in the rolling moment are investigated to see if the same selective coefficient correction approach can be used to design an accurate lateral cg estimator.

In (80) it can be seen that the unexpected roll acceleration $\dot{p}_{unex}$ needs to be estimated to calculate changes in the lateral cg, and the roll acceleration p is determined by the rolling moment and to a lesser extent by the yaw moment, see (72).

As before the coefficient data is obtained by trimming the aircraft at Mach 0.4 and 5000m altitude at different AoA and then obtaining a linear model. The coefficients affecting p are illustrated in Figure 54.
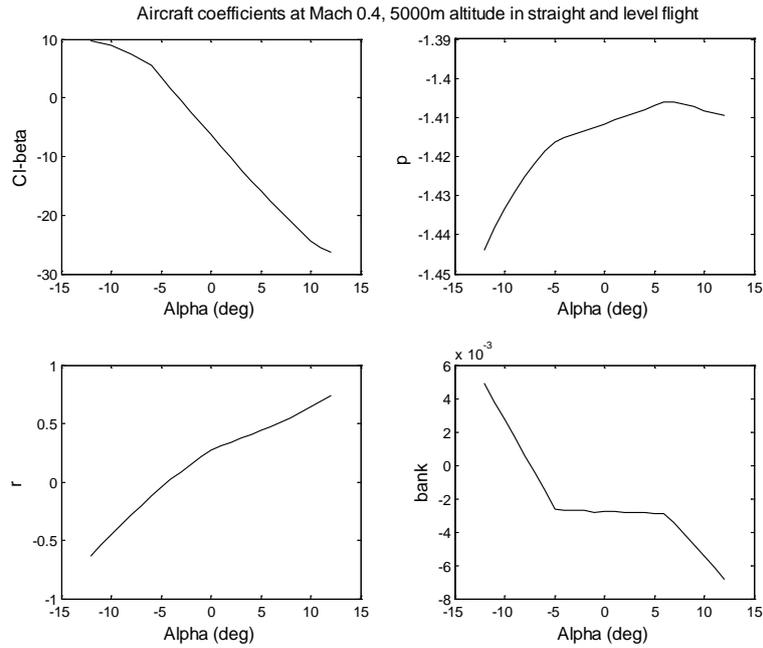
Figure 54: ADMIRE roll coefficients – measurements.

From the size of the coefficients it is clear that the key measurement coefficients affecting roll acceleration are beta (sideslip) and p (roll rate). The changes in Cl_β and Cl_r are relatively regular which means that linear interpolation between the two data end points of +/- 12° should provide sufficiently accurate coefficients. The graph of the Cl_p coefficient is curved and Cl_φ has a noticeable kink between an AoA of -6° to 6°.

The variation in the values of the Cl_p and Cl_φ coefficients should have a negligible effect upon the estimate of p for the following reasons. If two data point sets are used at 12° and -12° AoA then in the worst case at -5° AoA the coefficient for Cl_p used in the estimator will be approximately -1.43 instead of the correct value of approximately -1.415, giving a coefficient error of 0.015. Since the maximum roll rate 'p' from the manoeuvre requirements in section 3.1 is 30°/s (0.52 radians/s) then this will give a maximum error in estimated $\dot{p}$ of 0.0078 radians/s^2 (0.015 multiplied by 0.52), which would give an acceptable error in estimated lateral cg. The kink in Cl_φ will have even less effect upon the lateral cg estimate. For example at an AoA of 0° the coefficient used in the estimator will be approximately 0 but the correct value is about -0.0025 giving a coefficient error of 0.0025. At a bank angle of 45° (0.785 radians) this would give an error in the estimated $\dot{p}$ of approximately 0.002 radians/s^2.

This analysis has shown that for the measurement data then only two data points at 12° and -12° AoA should be necessary to give acceptable results.
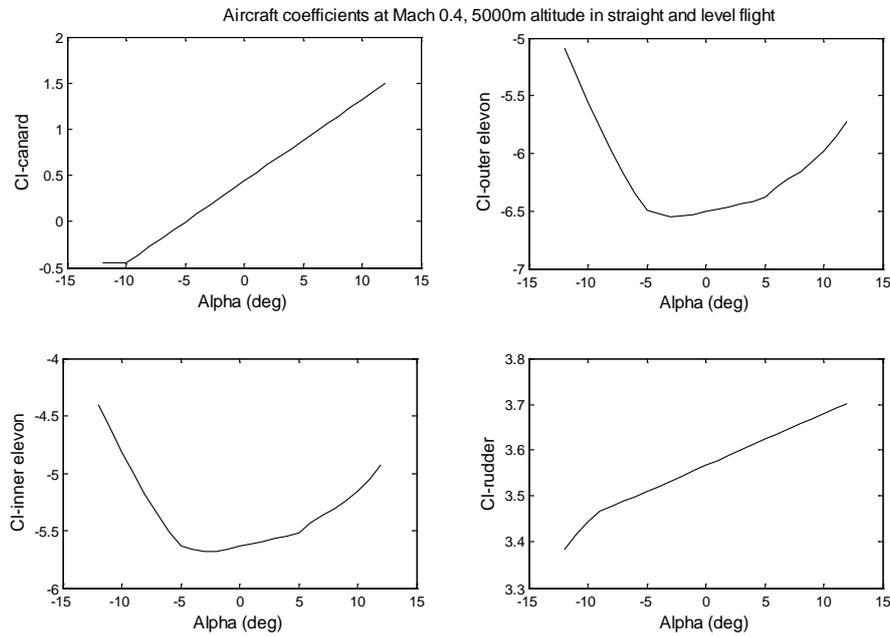
Figure 55: ADMIRE roll coefficients – commands.

Figure 55 shows the important command coefficients which affect $\dot{p}$. The change in the canard and rudder coefficients are regular down to -10° AoA, therefore the estimator should provide accurate estimates if data points were selected at -12°, -10° and 12° and linear interpolation was performed to obtain the coefficient values. The graphs for the inner and outer elevon coefficients both show a curved shape. The effect of this curve in the elevon coefficients upon the estimate of $\dot{p}$ depends upon the size of the elevon commands into the aircraft, a similar scenario for the effect of the elevon non-linearities on the pitching moment was described in section 6.3.

It should be noted that data was not obtained when the aircraft was banked because the trim routine only worked with the aircraft in level flight.

Providing that the elevon commands are not large then the results in Figure 55 suggest that fewer data points are required to estimate $\dot{p}$ with changes in AoA, compared with $\dot{q}$.

6.5.2 Lateral CG Estimator – Selective Coefficient Correction

As described in section 5.3 the lateral cg estimator is based upon using a Kalman-Bucy filter to estimate $\dot{p}_{unex}$.

A similar approach to that described in section 6.4 is applied to the lateral cg estimator, in which the coefficients used to estimate p are selectively modified when the aircraft manoeuvres. These coefficients are listed in Table 17 and were chosen because they had the largest effect upon roll acceleration.

Table 17: ADMIRE lateral cg estimator corrected coefficients

Commands	Cl_{dlc} Cl_{drc} Cl_{dloe} Cl_{droe} Cl_{dlie} Cl_{drie}
Measurements	Cl_β Cl_p Cl_r Cl_φ

A lateral cg estimator was tested with two different configurations. The first was configured using data points for AoA at 12° and -7°, the second was configured using the full set of AoA data points specified in Table 13.

The testing of the lateral cg estimator revealed that the largest error in the dycg estimate occurred during a roll manoeuvre. This is expected because the estimator is only configured with data obtained when the aircraft is in level flight. Figure 56 provides a comparison of the roll manoeuvre results when the estimator had the different configurations for the data points.
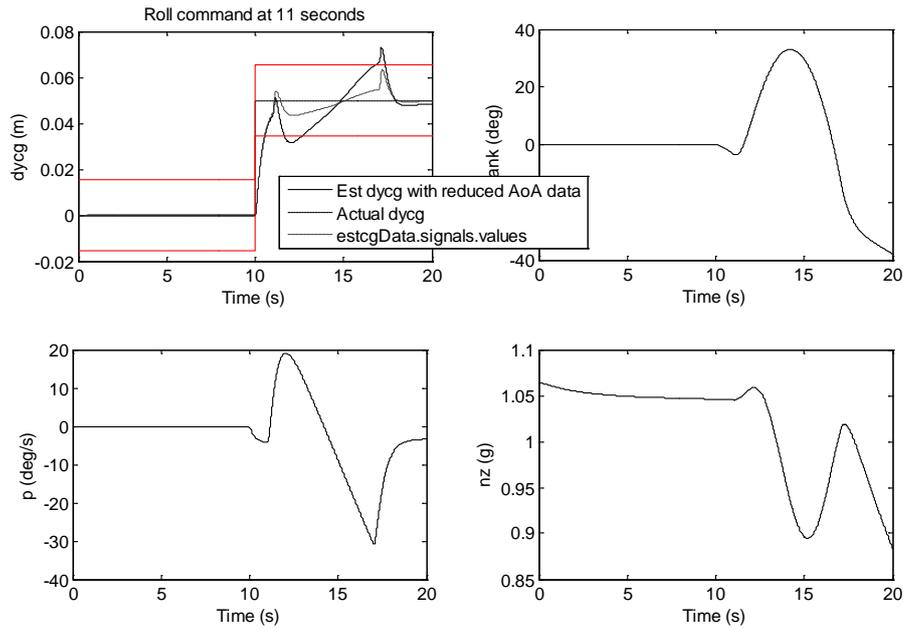
Figure 56: Estimate of lateral cg using different data point tables.

During the roll manoeuvre the larger set of data points is required to satisfy the estimator requirements and remain within the static estimation error limits. Since the estimator is only configured with data obtained when the aircraft is in level flight then, despite the analysis in section 6.5.1, the lateral cg estimator used the larger data set contained in Table 13. This larger data set provided the small improvements needed to keep the estimate within its static limits.

The structure of the lateral cg estimator is shown in Figure 57 and is similar to Figure 44. The output from the Kalman-Bucy filter is $\dot{p}_{unex}$ which is then divided by a calculation involving normal force, longitudinal force and moments of inertia to complete the dycg calculation given in (80).

100
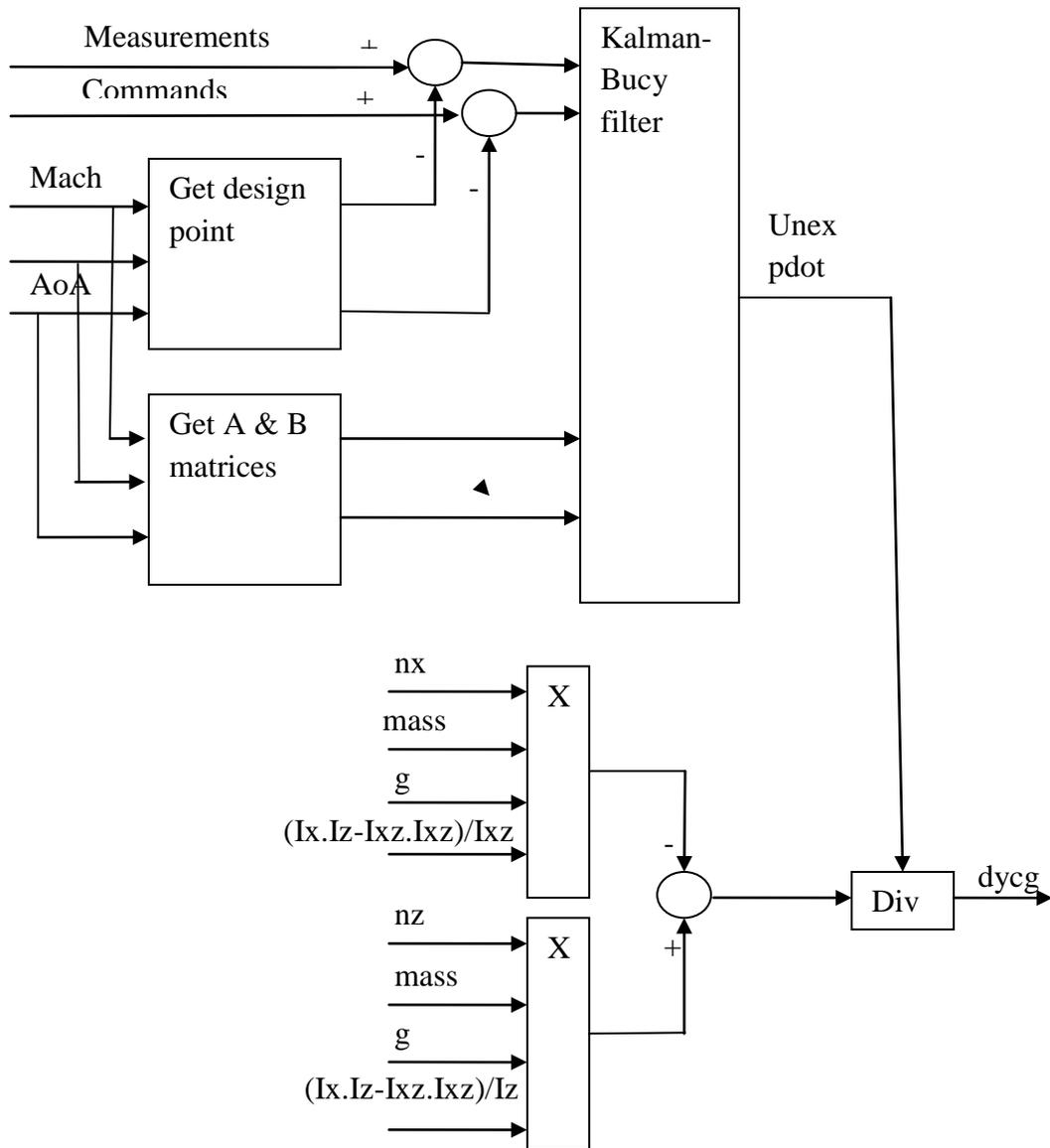
Figure 57: Lateral cg estimator structure

The "Get design point" and "Get A & B matrices" blocks operate in a similar manner to those described for the longitudinal cg estimator in section 6.4. The changed coefficients are defined in Table 17, the 6 command coefficients are inserted into the B matrix of the Kalman-Bucy filter and the 4 measurement coefficients are inserted into the A matrix in the Kalman-Bucy filter.

6.5.3 Modified Lateral CG Estimator - Results

The results in this section were obtained when the Kalman-Bucy Q matrix was populated with the values in (54). As before the R matrix was populated based upon the noise data defined in Table 8.

$$Q = diag([10\ 0.02\ 0.02\ 0.02\ 0.01\ 0.1\ 0.01\ 0.01\ 0.01\ 10\ 0.1]) \tag{88}$$

The full set of data points defined in Table 13 was used in the interpolation tables.

This section displays sample results for the different aircraft manoeuvres. The manoeuvres are identical to those in section 6.4. Each aircraft manoeuvre occurs when the aircraft is trimmed in straight and level flight at Mach 0.4 and at 5000m altitude, and after 10 seconds there is a lateral cg shift of 0.05m. Note that for clarity measurement noise has not been added to the signals making the results less noisy than they would otherwise be.

The lateral cg estimate is displayed in the top left graphs, the other graphs display aircraft characteristics which are related to the manoeuvre.

In all of the tests the lateral cg estimate stayed within its static upper and lower limits.

6.5.3.1 Acceleration
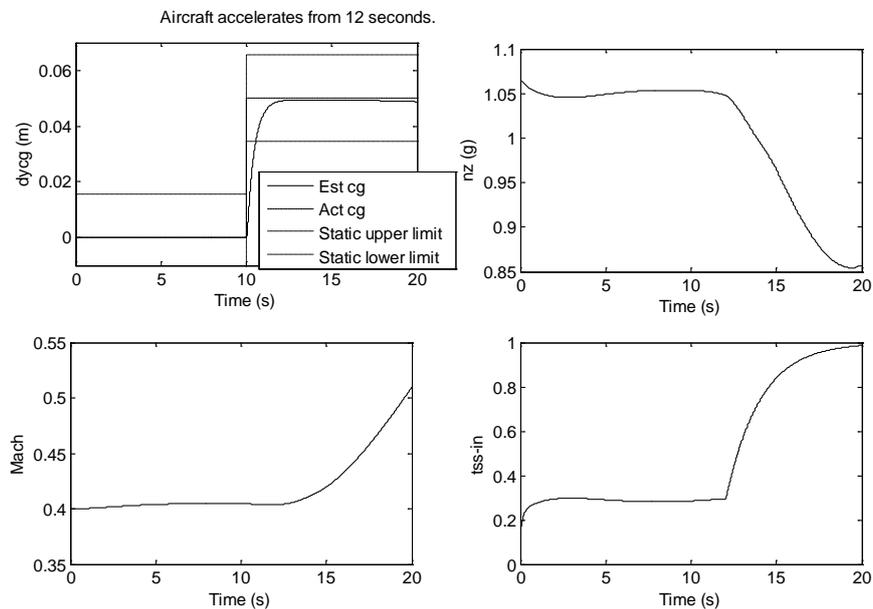
Figure 58: Acceleration test.

6.5.3.2 Lateral Acceleration
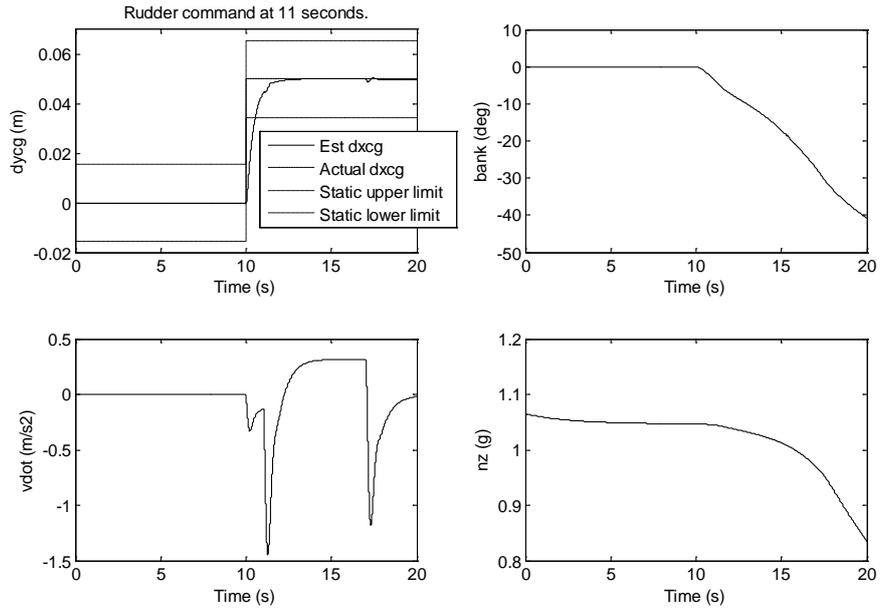
Figure 59: Lateral acceleration test.

6.5.3.3 Roll
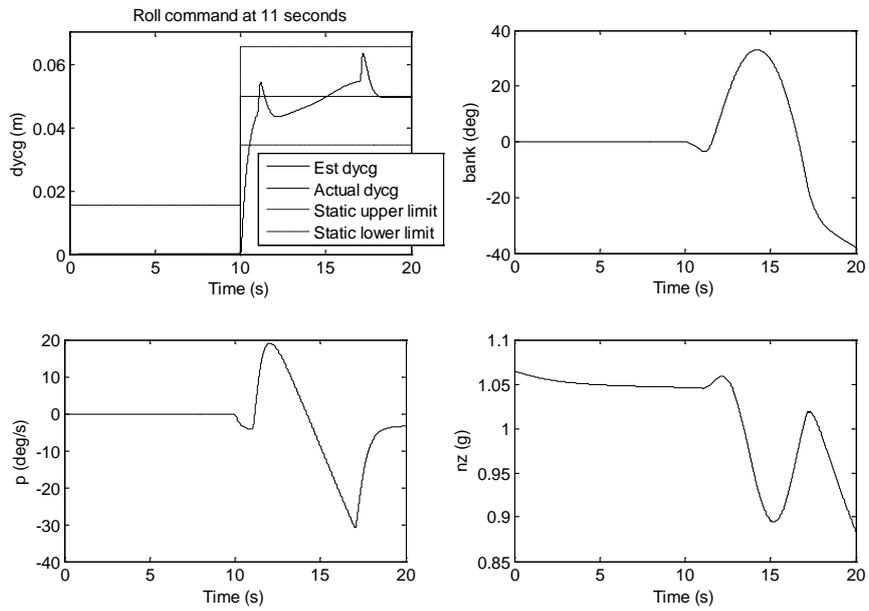
Figure 60 : Roll test.

6.5.3.4 Pitch down

103

Full push down at 15.5 seconds over 5 seconds then return to zero over 5 seconds.
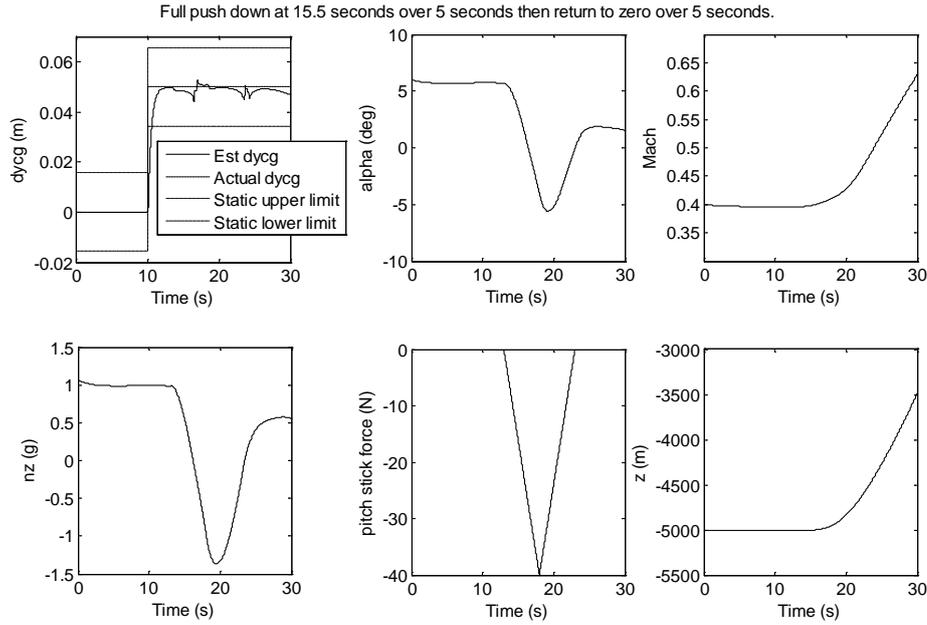
Figure 61: Push down test.

6.5.3.5 Sample Mission

In a sample mission the aircraft accelerates from Mach 0.2 at 20m altitude to about Mach 0.5 at 4500m altitude. It then levels off and performs a couple of banked turns changing direction by about 60°, and then descends to about 700m. During this mission there are four step changes in the lateral cg as shown in Table 18, which also shows the change in aircraft mass.

Table 18: Sample mission. Changes in cg and aircraft mass

Time (s)	dycg (m)	Δmass (kg)
40	0.05	-205
60	0	-205
110	-0.05	-205
120	0	-205

There is a constant reduction in mass of 2kg per second to simulate fuel burn, taking no account of engine thrust. This continual reduction gives a total reduction in aircraft mass of

20% over the 500 second simulation. As before, this unrealistic rate of reduction in mass has been done to minimise simulation time.

Figure 62 shows the cg estimate, z, mass and nz.
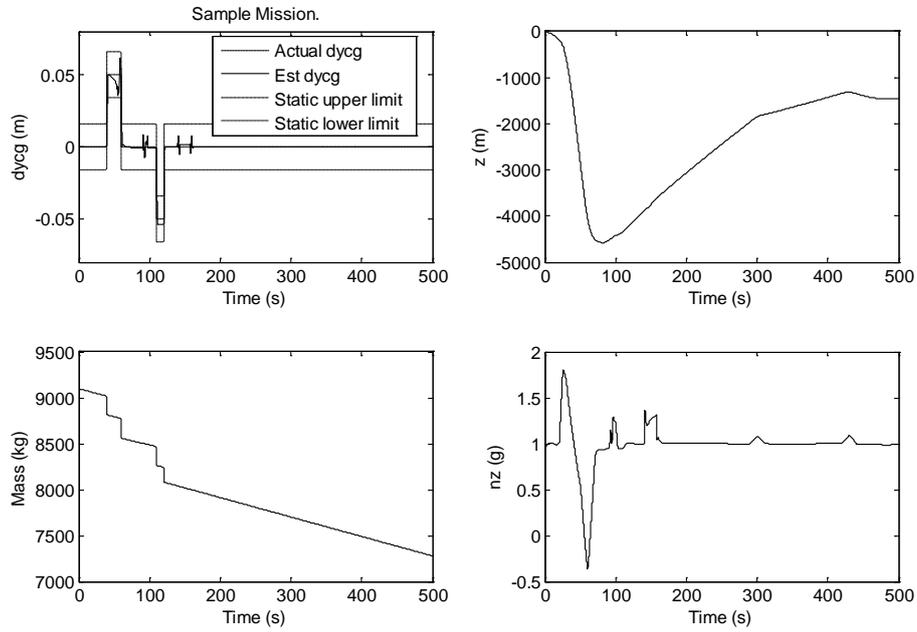
Figure 62: Sample mission (dycg, z, mass and nz).

Figure 63 shows Mach, AoA, yaw, bank, p and tss.
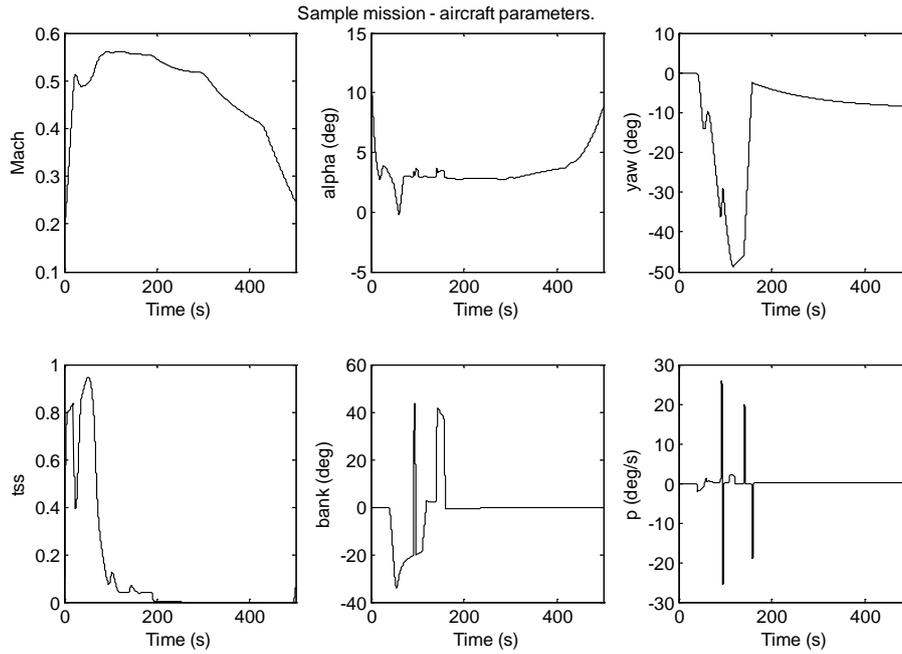
Figure 63: Sample mission (Mach, alpha, yaw, tss, bank p).

Figure 64 provides a zoomed in view of the dycg estimate in Figure 62.
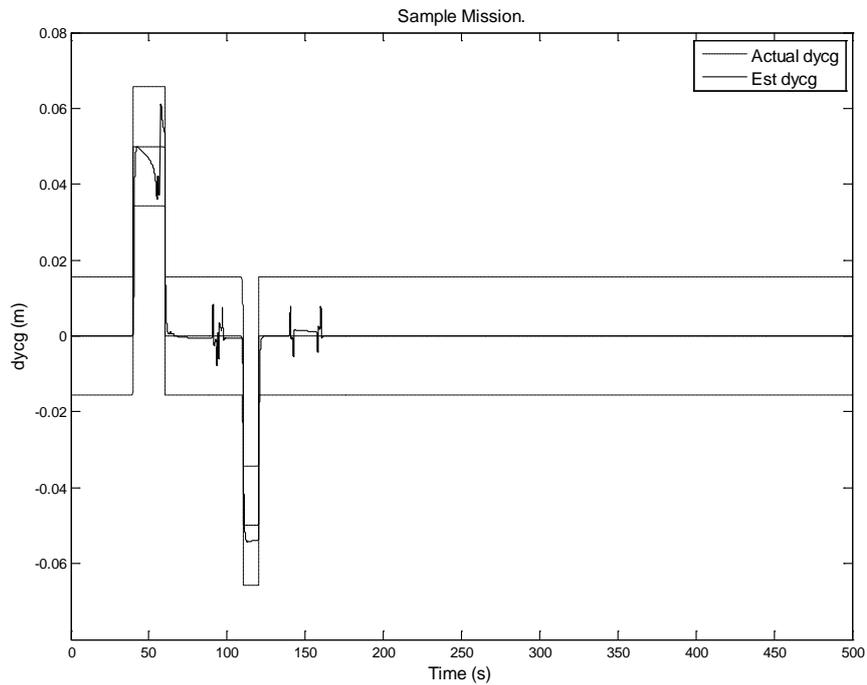
Figure 64: Sample mission dycg estimate.

During the mission, and other manoeuvre tests, the dycg estimate stayed within its static limits.

6.6 Recovery from manoeuvres exceeding specification

This section provides two examples of the performance of the cg estimator when the aircraft manoeuvre exceeds the limits specified in section 3.1. It is acceptable for the estimate to exceed its maximum dynamic error limits of 0.5% mac (26mm in the ADMIRE model) for an out of range manoeuvre, but when the aircraft returns to an acceptable manoeuvre then the estimator should then provide an accurate estimate within one second.

Figure 65 provides an example when the aircraft push down command exceeds requirements. The lower middle graph shows the pitch stick command when the maximum push down is commanded immediately after 12 seconds without a 5 second delay. After 22 seconds the pitch stick immediately returns to level, again without the 5 second delay. The cg estimate (top left graph) show the estimate exceeding the static limits but then returning within limits after about 24 seconds, when the out of range manoeuvre has ceased.
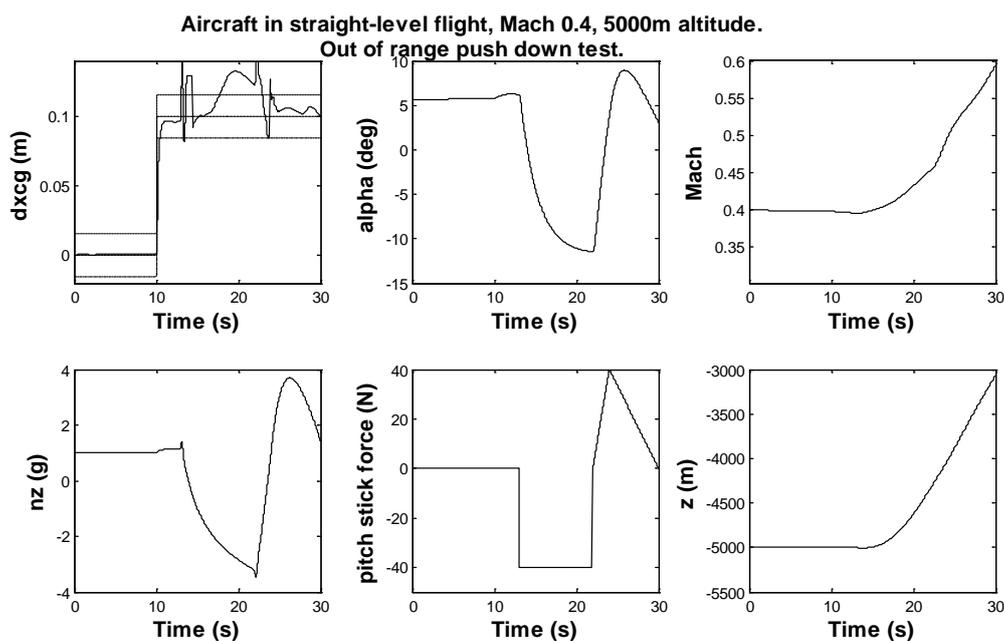
Figure 65: Out of range push down test.

Figure 66 provides an example of the results when an out of range roll acceleration manoeuvre is performed. The aircraft bank angle approaches 80° and the roll acceleration p exceeds the 30°/second limit, for example after 18 seconds p exceeds -40°/second. As before the estimate exceeds its static limit but returns within limits when the manoeuvre has completed.
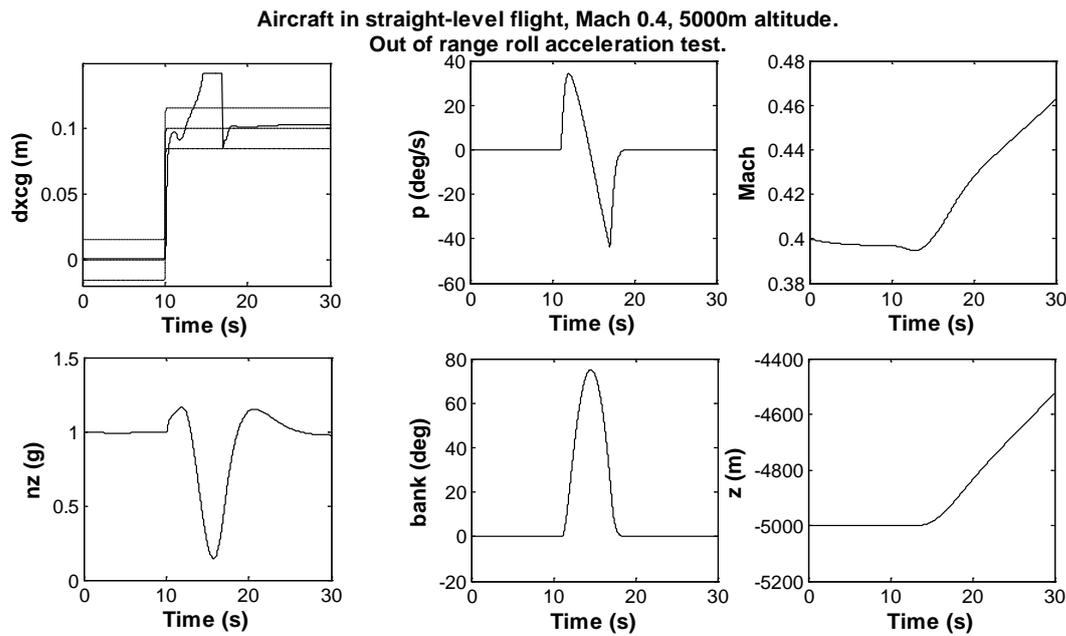
Figure 66: Out of range roll acceleration test.

6.7 Robustness Tests

All of the tests have so far assumed that the estimator is designed based on a perfect model of the aircraft. However in practice the estimator model will never be perfect so it is useful to assess the impact of inaccuracies in it. This assessment is performed by amending the aircraft parameters to differ from the estimator. The list of parameters that are assessed, and the size of the variation between aircraft and estimator, are defined in Table 19. The size of the parameter changes were decided after consultation with BAE Systems and limited by the restrictions of the ADMIRE model (Forsell and Nelson, 2005: 35).

The linear aircraft model is used for the robustness tests when the aircraft coefficients are varied, because the coefficient derivative is only valid around a small range and at higher positive/ negative AoA the contribution from the error is increased. The non-linear model is used when the inertias and mass are varied because the effect of changes in inertia and mass is unaffected by changes in AoA.

The aircraft was configured at Mach 0.4 5000m altitude in straight and level flight, and a change in dxcg of 0.1m was applied after 10 seconds. The tests for the effect upon dxcg were

performed using the longitudinal estimator described in section 6.4 and the tests for the effect upon dycg were performed using the lateral estimator described in section 6.5.

Table 19: ADMIRE robustness tests. Longitudinal cg % error.

Pitch coefficients	Change (%)	Straight-level final estimate	% error	% static tolerance level	Model
Cm_α	10	0.0987	1.3	8.3	Linear
Cm_q	10	0.1	0	0	Linear
Cm_{dne}	10	0.0991	0.9	5.8	Linear
Cm_{dei}	10	0.0955	4.5	28.8	Linear
Cm_{dey}	10	0.0972	2.8	17.9	Linear
Ix	20	0.1	0	0	Non-linear
Iy	5	0.1	0	0	Non-linear
Iz	5	0.1	0	0	Non-linear
Mass	5	0.104	4	25.6	Non-linear

Table 19 summarises the results from this testing. In straight and level flight the Cm_α coefficient has the biggest effect of the measurements upon the final estimate. For example a 10% change in Cm_α resulted in the estimate for a 0.1m cg change being in error by 0.0013m which is a 1.3% error, and uses up 8.3% of the static error tolerance defined in the BAE requirements in section 3.2. The control surface coefficients for the elevons Cm_{dey} and Cm_{dei} have a larger effect than the canard Cm_{dne} upon the estimate. The percentage error due to mass is very similar to the actual percentage error in mass, which is not surprising because in (70) there is a division by mass to convert the estimate of the unexpected pitch acceleration into an estimate of dxcg.

In straight and level flight changing the inertia values had no noticeable effect upon the estimate, therefore push-down and roll manoeuvre tests were performed when the inertias had been varied. The results of these tests are shown in a number of graphs (see Appendix G), an example is provided in Figure 67.

The results from varying the inertias reveal that during a push-down manoeuvre the inertias I_x and I_z had no visible impact upon the estimate but I_y had a transient effect. This is to be expected because during a pitch manoeuvre it is the error in the pitching moment of inertia that would affect the estimate. During a roll manoeuvre I_x and I_z have a small transient effect upon the estimate, the effect of the error by I_y is negligible.

Estimate with aircraft trimmed in straight and level flight, Mach 0.4, 5000m altitude. Iy increased by 5%.
Full pitch down commanded at 13 seconds over 5 seconds, then further 5 seconds to return to zero pitch command.
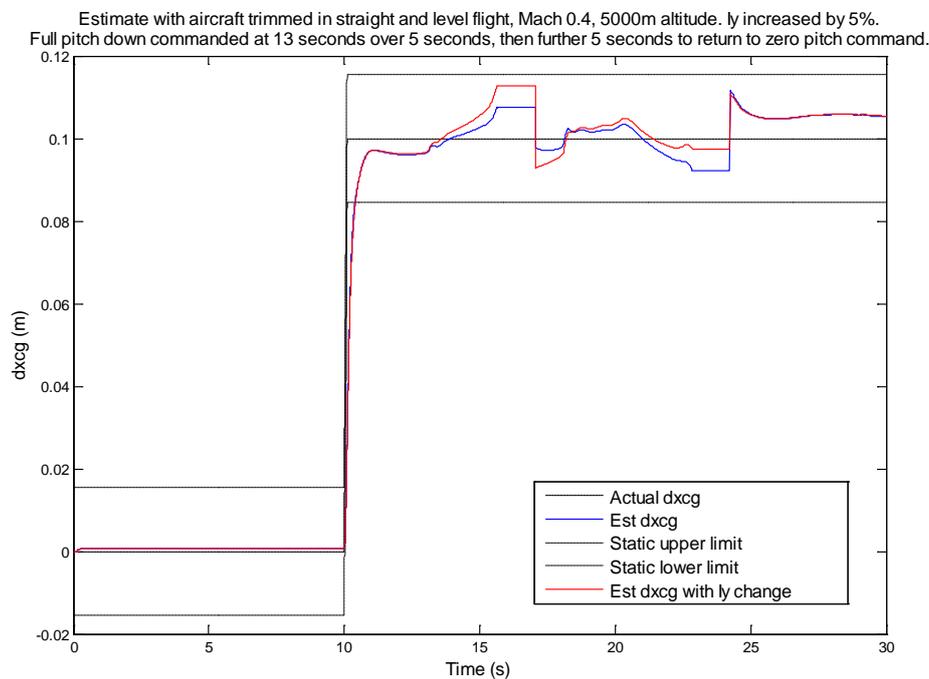
Figure 67: Estimate of longitudinal cg change of 0.1m. Aircraft in push down manoeuvre. Iy increased by 5%.

In the tests applied to the lateral cg estimator a cg change of 0.1m was injected into the linear aircraft model. The tests applied for the changed inertias and mass used the non-linear aircraft model and used a smaller lateral cg change of 0.05m. This smaller cg change was used to ensure that the non-linear aircraft model generated acceptable manoeuvres. No control surface coefficients were altered for the rolling moment because this facility is not available in ADMIRE.

Table 20 summarises the test results for the lateral cg estimator.

Table 20: ADMIRE robustness tests. Lateral cg % error

Roll coefficients	Change (%)	Straight-level final estimate	% error	% static tolerance level	Model
Cl_β	10	0.0974	2.6	16.7	Linear
Cl_p	10	0.0963	3.7	23.7	Linear
Cl_r	10	0.1012	1.2	7.7	Linear
Ix	20	0.049*	0	0	Non-linear
Iy	5	0.049*	0	0	Non-linear
Iz	5	0.049*	0	0	Non-linear
Mass	5	0.052*	6	19.2	Non-linear

* Based upon 0.05m change in lateral cg

In straight and level flight the error in the Cl_p coefficient had slightly more effect upon the estimate than Cl_β or Cl_r. For example the 10% change in Cl_p resulted in the estimate for a 0.1m cg change being in error by 0.0037m which is a 3.7% error, and uses up 23.7% of the static error tolerance defined in the BAE requirements in section 3.2.

As before the percentage error due to mass is very similar to the actual percentage error in mass, because in (80) the unexpected roll acceleration is divided by mass to convert it into an estimate of dycg.

The changes to the moments of inertias had a negligible effect upon the estimate of dycg during straight and level flight. During a push-down manoeuvre the changed inertias did affect the estimate, during a roll the errors in I_y and I_z had no noticeable effect upon the dycg estimate.

The results of these tests are shown in a number of graphs (see Appendix G), an example is provided in Figure 68.
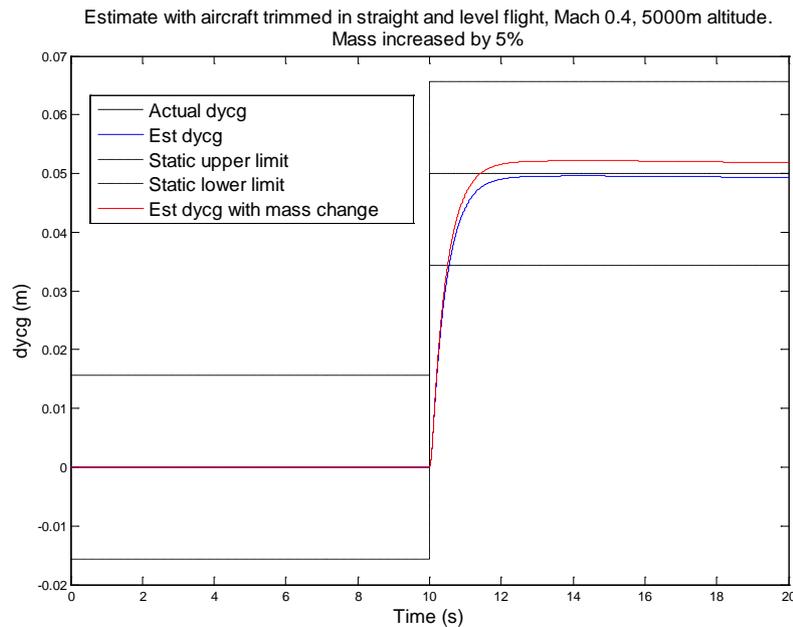
Figure 68: Estimate of 0.05m lateral cg change. Mass increased by 5%.

6.8 Summary

This section has shown that a Kalman-Bucy filter will not give accurate cg estimates when the aircraft is at a different speed to the one that the estimator was configured for. It explained that dynamic pressure $\bar{q}$ increases with speed and that the estimator could be modified to take into account changes in $\bar{q}$ but that the estimator would still not satisfy its requirements. Some of the non-linearities affecting the pitching and rolling moments were then examined and it was demonstrated that a technique to selectively modify some of the pitching and rolling moment coefficients based upon the aircraft speed, altitude and angle of attack can produce an accurate estimator.

The results from a series of tests based upon the manoeuvre requirements have been provided and the cg estimates generally stayed within its static limits. The one test failure occurred during high aircraft acceleration in the sample mission and this could be corrected by a more accurate engine model.

Some example tests were provided when the aircraft manoeuvre exceeded the requirements. The cg estimate was inaccurate during the excessive manoeuvre but quickly corrected itself when the manoeuvre completed.

A sequence of robustness tests have been performed upon the estimator by varying the aircraft parameters and then examining the results when the aircraft was in straight and level flight, or performing a pitch or roll manoeuvre. The main points from these tests were:

- Inaccurate values for the inertias only affected the cg estimate during a manoeuvre, and even then only had a relatively small effect.
- An incorrect value of mass in the estimator coefficients had a proportional effect upon the cg estimate.
- The most important coefficients affecting the longitudinal cg estimate are Cm_α, Cm_{dei} and Cm_{dey}. The most important coefficients affecting the lateral cg estimate are Cl_β, Cl_p, Cm_{dei} and Cm_{dey}.

The scaling method described in this chapter is a general approach that should be suitable for expanding to cover the full flight envelope. The coefficient data tables currently only contain data for the aircraft covering the speed, altitude and angle of attack defined in Table 13, therefore when the aircraft exceeds these limits the coefficient data is inaccurate. However there is no reason to assume that at greater speeds, angle of attack or altitude that this approach should not work providing that sufficient data points are obtained to cater for any non-linearities in the aircraft dynamics. Note that since the Kalman gain used in the estimator is fixed then the correction by the Kalman-Bucy filter is non-optimal and becomes increasingly non-optimal as the aircraft deviates from the design point for the Kalman gain.

The Kalman-Bucy filter was tuned to provide a reasonable performance for the manoeuvre requirements specified in section 3.1. The estimator was not rigorously tested when the aircraft underwent harsher manoeuvres but from the existing results it is possible to estimate the impact of more severe manoeuvres on the cg estimate.

The results show that the estimator is most sensitive to changes in pitch and roll, therefore if more stringent pitch requirements were specified then the estimator may require additional angle of attack data points. If the aircraft manoeuvre requirements permitted a greater roll angle then the estimator would probably fail to meet its performance requirements unless the data tables were restructured to include coefficient data at different roll angles. Lateral acceleration had little effect upon the accuracy of the estimate, therefore the estimate should remain accurate if greater lateral acceleration is permitted.

7. Combined Longitudinal and Lateral CG Estimator

This section describes a combined longitudinal-lateral cg estimator. To make the modelling of cg changes more accurate a BAE cg modifier model is included which models the effect of store changes and fuel sloshing on the cg position.

7.1 Combined cg estimator

Section 6.4 described the use of $\dot{q}_{unex}$ to estimate dxcg and section 6.5 described the use of $\dot{p}_{unex}$ to estimate dycg. A combined estimator for both dxcg and dycg includes both values in its state giving $[V_T, \alpha, \beta, p, q, r, \varphi, \theta, \varphi, z, \dot{q}_{unex}, \dot{p}_{unex}]$.

$$
\begin{bmatrix} \dot{x}' \\ \dot{q}_{act} \\ \dot{p}_{act} \\ \ddot{q}_{unex} \\ \ddot{p}_{unex} \end{bmatrix} = \begin{bmatrix} \cdots & \cdots & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & 1 & 0 \\ \cdots & \cdots & \cdots & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x' \\ q_{act} \\ p_{act} \\ \dot{q}_{unex} \\ \dot{p}_{unex} \end{bmatrix} + \begin{bmatrix} \cdots \\ \cdots \\ \cdots \\ 0 \\ 0 \end{bmatrix} u
\tag{89}
$$

$$
C = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
\tag{90}
$$

Equation (89) shows the augmented aircraft state-space equation to obtain estimates for $\dot{q}_{unex}$ and $\dot{p}_{unex}$. The augmented state is fully observable. As before the Kalman gain for the Kalman-Bucy filter was obtained by using the Matlab 'lqe' equation.

The value of the Q matrix used in this combined cg estimator is:

$$
Q = diag([10\ 0.02\ 0.02\ 0.02\ 0.01\ 0.1\ 0.01\ 0.01\ 0.01\ 10\ 0.1\ 0.2])
\tag{91}
$$

The noise variance of the data contained in Table 8 is used to populate the R matrix of the Kalman-Bucy filter.

7.2 BAE cg modifier

BAE Systems provided a model to set the cg value. This model contains two elements: a fuel cg model and a stores cg model.

The stores model calculates dxcg and dycg based upon the stores carried at six stations: four under fuselage and two under the wings. The store locations are defined in Table 21.

Table 21: BAE cg modifier store locations

	X position (relative to cg) m	Y position (relative to cg) m
Left under fuselage (front)	-1.5	-0.5
Right under fuselage (front)	-1.5	0.5
Left under fuselage (rear)	2	-0.5
Right under fuselage (rear)	2	0.5
Left wing tip	1.2	-4.5
Right wing tip	1.2	4.5

The store mass depends upon the type of store selected, the options are for masses of 0, 80, 160 or 500kg.

The fuel model calculates dxcg and dycg based upon the amount of fuel in five fuel tanks, two in each wing and one in the fuselage with a maximum fuel load of 2 tonnes. The fuel tank locations are defined in Table 22.

Table 22: BAE cg modifier fuel tank locations

	Full tank mass (kg)	X position (relative to cg) m	Y position (relative to cg) m
Left rear wing tank	400	-0.6	-1.5
Right rear wing tank	400	-0.6	1.5
Left forward wing tank	200	1.2	-2.7
Right forward wing tank	200	1.2	2.7
Fuselage tank	800	0	0

The BAE cg modifier model uses the aircraft acceleration to approximate fuel surface angles. The fuel angles and tank fractions are used in five 2-d lookup tables to calculate the change in cg due to fuel slosh within each tank. The acceleration signals into this model go through a one second lag filter to approximate the lag effect in the fuel movement within the tanks as the aircraft manoeuvres.
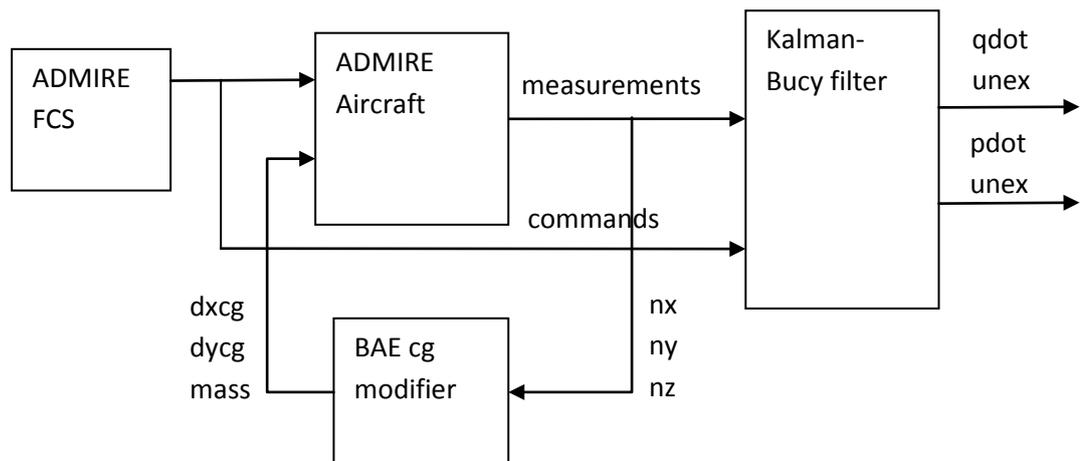
Figure 69: ADMIRE model with BAE cg modifier and estimator

Figure 69 shows the configuration with the new BAE cg modifier model using the aircraft acceleration to simulate the effect of fuel sloshing and its impact upon the cg. The stores configuration is internal to the BAE cg modifier model and therefore not shown in the diagram. The commands into the aircraft and Kalman-Bucy filter are listed in Table 9 and the measurements from the aircraft are listed in Table 10. The Kalman-Bucy filter is shown outputting the unexpected pitch and roll acceleration, the other states are output but omitted for clarity. The unexpected pitch acceleration $\dot{q}_{unex}$ and the unexpected roll acceleration $\dot{p}_{unex}$ are converted into estimates of the cg change in the same way as described in section 6, and illustrated in Figure 44 and Figure 57.

7.3 Results from combined longitudinal and lateral cg estimator with BAE cg modifier

For tests using the BAE cg modifier the following scenarios were considered:

a) Full stores held, fuel tanks half empty to generate fuel slosh, full set of manoeuvres
b) Stores deployed, fuel tanks half empty to generate fuel slosh, full set of manoeuvres

In all of the figures the top left graph contains the cg estimate, and the static upper and lower limits are marked in red.
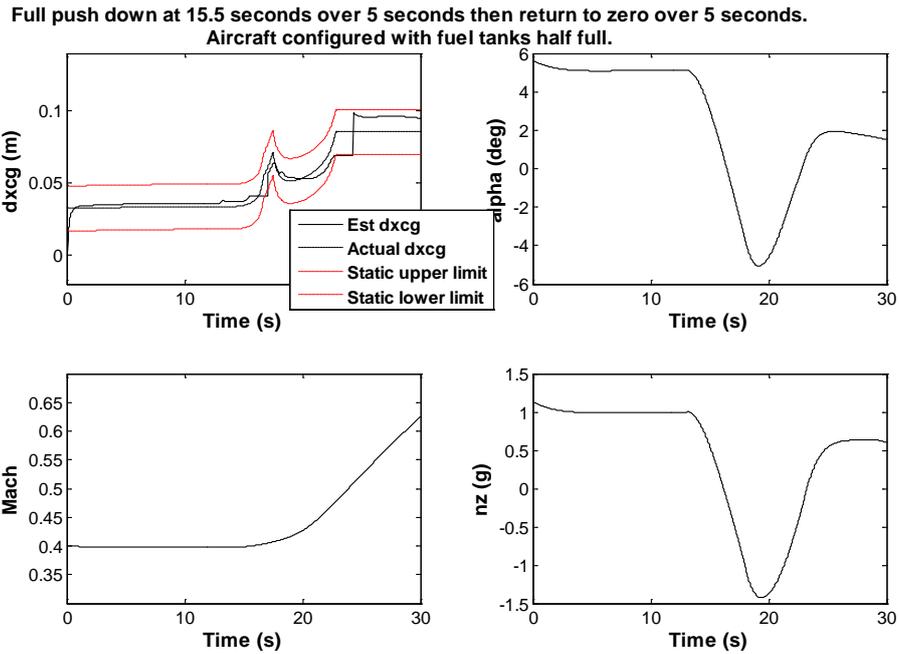
Figure 70: CG estimate during push down test using BAE cg modifier. Fuel tanks half full.

In the push down test shown in Figure 70 the cg estimate briefly exceeds its static lower limit at 17 seconds and between 22 and 24 seconds, this is when the estimator holds its last good value because nz is between plus and minus 0.4 g. These exceedances might be avoided by more table data points and / or a change to the rule when dxcg is held at its previous value e.g. hold value when nz is between +/-0.3 g instead of +/-0.4 g. The small exceedances are against the static limits, the dxcg estimate remains well within the dynamic limits.
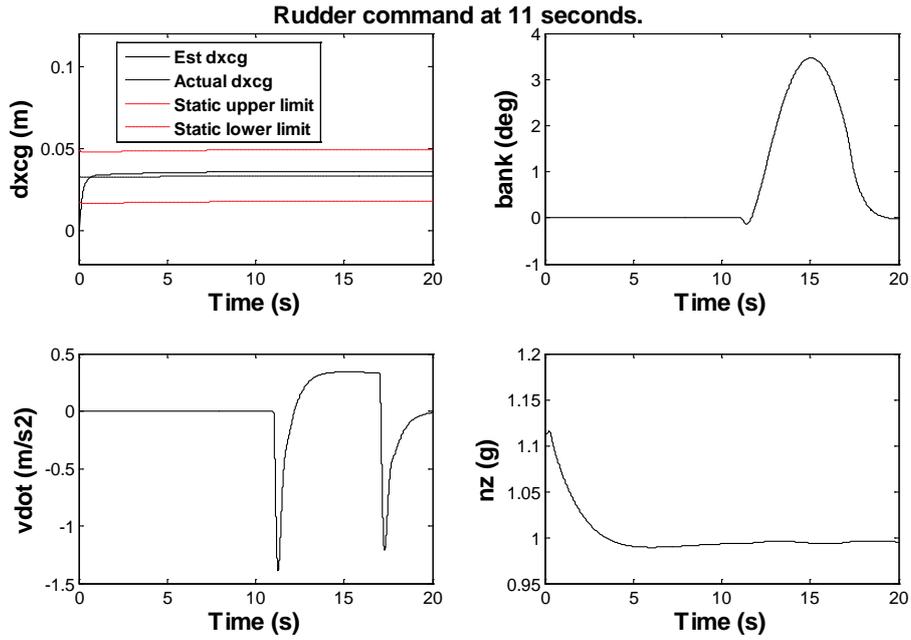
Figure 71: CG estimate during lateral acceleration using BAE cg modifier. Fuel tanks half full.
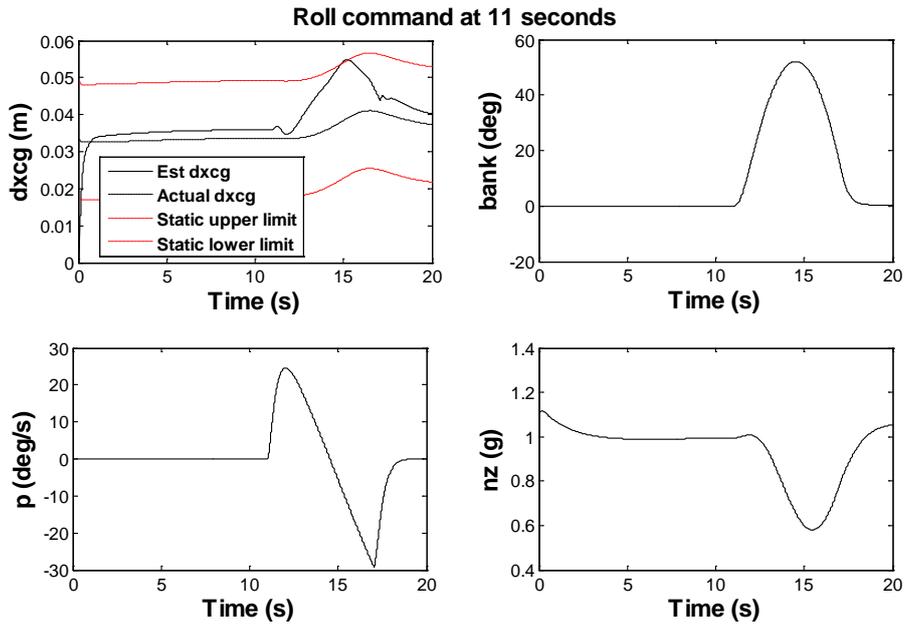
Figure 72: CG estimate during roll test using BAE cg modifier. Fuel tanks half full.

The rudder test (Figure 71) generated virtually no change in dxcg, and the estimate stays accurate.

In the roll test (Figure 72) the dxcg estimate briefly exceeds its static upper limit after about 15 seconds, but as noted for the push down test the actual cg is changing and the estimate is within the less restrictive dynamic upper limit.

In the acceleration test (Figure 73) the dxcg stays within its static limits until the aircraft is at Mach 0.8, which exceeds the table data used in the estimator which is only configured for speeds up to Mach 0.6.
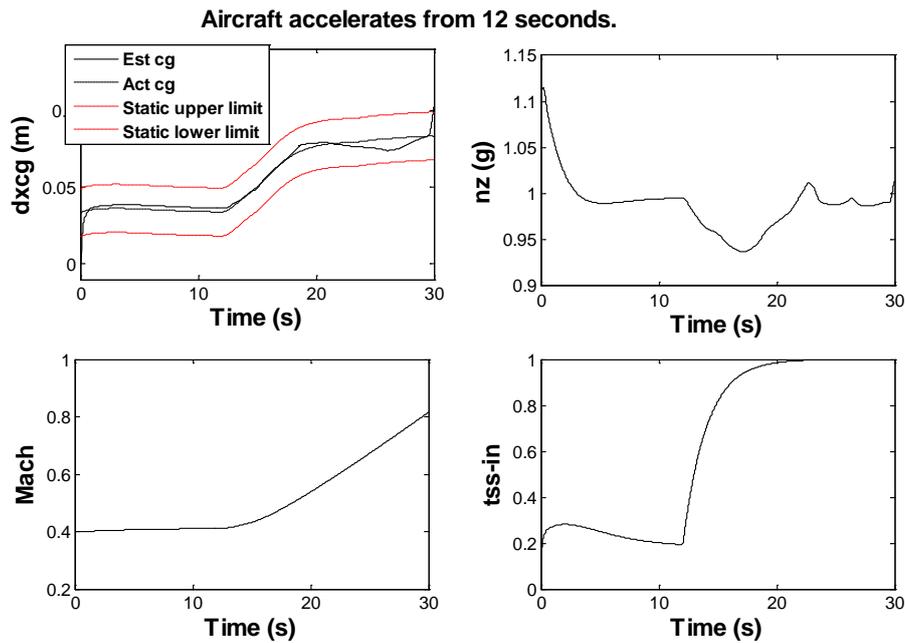
Figure 73: CG estimate during acceleration test using BAE cg modifier. Fuel tanks half full.

The tests have shown the estimate of dxcg briefly exceeding its static limits but staying within its dynamic limits at all times.

In all of the above tests, with the exception of the roll test, dycg stayed at zero, or very close to zero, and the estimate of dycg also stayed very close to zero so therefore the results have not been shown. However for the roll test the estimate of dycg exceeded its static limit, see Figure 74.
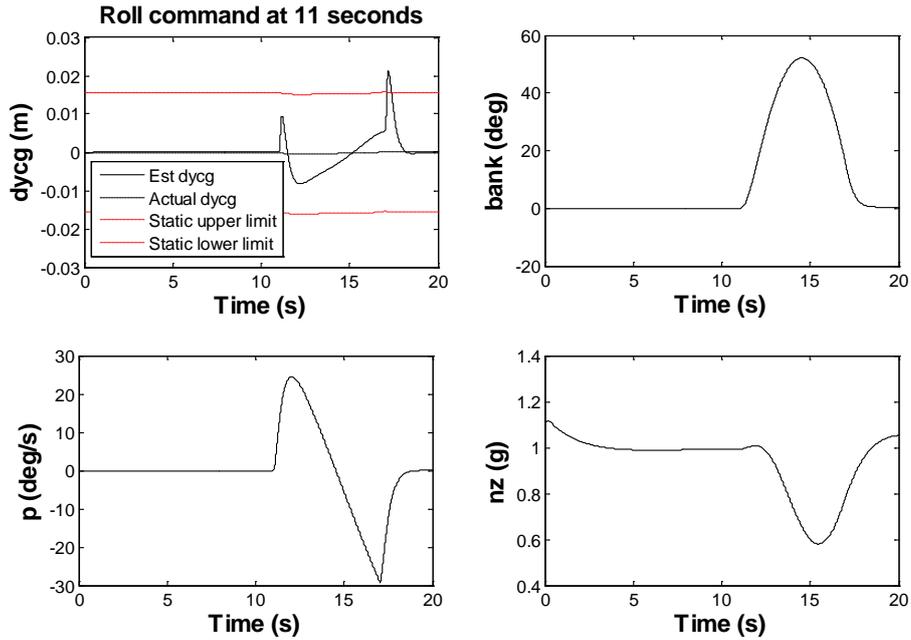
Figure 74: Lateral cg estimate during roll test using BAE cg modifier. Fuel tanks half full.

The final set of test results for the combined longitudinal-lateral cg estimator cover the scenario when stores have been deployed to effect a change in dxcg, and then deployed to change both dxcg and dycg.

Figure 75 shows the results when there is a store deployment after 10 seconds when the aircraft is in straight and level flight. The store deployment changes dxcg but dycg remains unaffected.

121

Figure 75: CG estimate with store deployment. Fuel tanks half full.

Figure 76 shows the results from a store deployment after 10 seconds when the aircraft is in straight and level flight. This store deployment affects both dxcg and dycg.

Figure 76: CG estimate with store deployment. Fuel tanks half full.

The final example using the BAE cg modifier (Figure 77) covers the scenario when stores are deployed after 10 seconds affecting dycg and dxcg, and then a full push down is commanded which causes significant fuel sloshing.

The results show the dycg estimate staying well within its static limits, and the dxcg also just stays within its static limits.
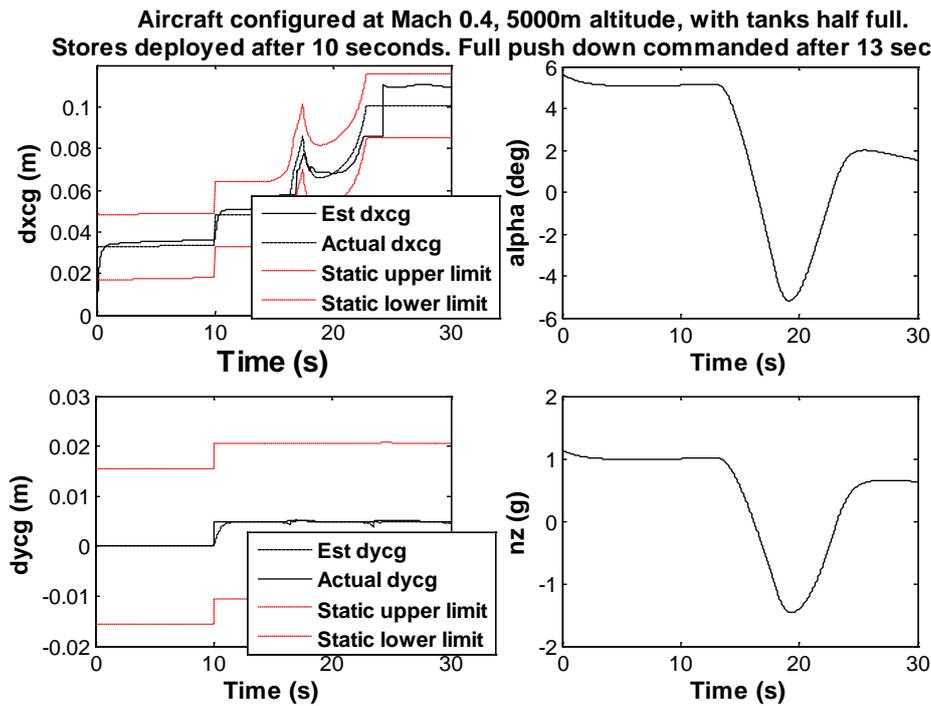
Figure 77: CG estimate with store deployment and max push down. Fuel tanks half full.

7.6 Summary

This section has described the design of a combined longitudinal and lateral cg estimator. The potential effect of fuel sloshing upon aircraft performance was modelled by using a cg modifier model supplied by BAE Systems, and the cg estimator was also tested using this model.

The results have met the performance requirements set by BAE in section 3.1 with the following exceptions:

a) During a full push down command with fuel tanks half-empty the dxcg estimate exceeded its static limit. This may be corrected by amending the estimator to either

use more data points or to freeze the estimate at smaller values of nz e.g. +/- 0.3g instead of +/- 0.4g.

b) During a roll command with fuel tanks half-empty the dycg estimate exceeded its static limit. This may be corrected by obtaining coefficient data when the aircraft is banked and including the bank angle as an input into the data tables used to correct the coefficients.

In all of the tests the cg estimate remained within the more generous dynamic accuracy limits.

8. Discrete Centre of Gravity Estimator

The previous sections have described a continuous cg estimator. This section describes a discrete version of the estimator that may be used in a hardware implementation. It also provides some sample results when this estimator is used in conjunction with a hardware fuel rig, and physical faults are applied to the rig causing changes in the cg.

8.1. Discrete CG Estimator

The discrete version of the estimator uses a digitised version of a time-invariant Kalman filter instead of a Kalman-Bucy filter. Section 2.1.2 provides a comparison between the Kalman-Bucy and Kalman filters.

A similar design process to that described in section 6 is followed to obtain the coefficient derivatives used in the lookup tables. However this time the Matlab 'c2d' function is used to convert the linear model into its discrete equivalent, using a sample time of 0.01 seconds. The set of data points contained in look up tables is defined in Table 13, and the set of coefficient derivatives that are modified is defined in Table 14.

The structure of the estimator is very similar to that shown in Figure 49 except that a Kalman filter is used instead of a Kalman-Bucy filter. The same values are used in the Q matrix (53) and R matrix (see Table 8) as were used in the continuous time version, and the Matlab 'dlqe' function is used to obtain the Kalman gain.

An 11 state estimator was created which contained the state data defined in (71).

Zero order hold blocks are added to sample and hold the inputs into the discrete estimator, see Figure 78. As before the commands are listed in Table 9 and the measurements are listed in Table 10. The discrete estimator was tested with a sample of manoeuvres and obtained the same results as the continuous estimator containing a Kalman-Bucy filter.
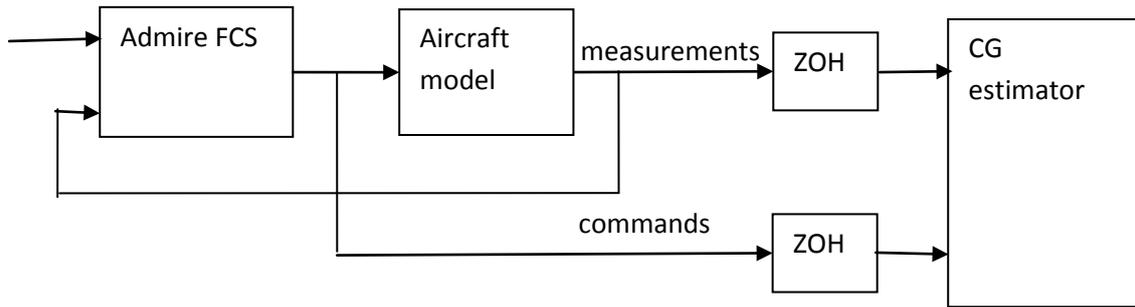
Figure 78: ADMIRE aircraft model with discrete cg estimator

BAE Systems, the project industrial sponsors, provided access to a hardware fuel rig facility to test the cg estimator (see Figure 79). The fuel rig can be set up in different configurations and can be used to test the effect of different faults in the system, its effect upon the cg, and the response of the estimator.

Figure 79: BAE hardware fuel rig.

The rig was configured to match the fuel tank configuration used in the BAE cg modifier model described in section 7. No model for stores was included in the configuration.

Table 23 defines the tank masses and locations relative to the cg.

Table 23: Fuel rig tank configuration

Fuel tank location	Fuel mass (kg)	X axis position (m)	Y axis position (m)
Left rear wing	400	-0.6	-1.5
Right rear wing	400	-0.6	1.5
Left front wing	200	1.2	-2.7
Right front wing	200	1.2	2.7
Fuselage	800	0	0

The hardware fuel rig is controlled using a fuel control system in Labview. This control facility is used to configure the rig fuel tank attributes to match Table 23.

Fuel level readings are transmitted over the LAN to a PC running Simulink which converts the fuel levels into the appropriate change in cg, which is then injected into the ADMIRE aircraft model as described in earlier sections. The aircraft model block, see Figure 80, contains the ADMIRE FCS as well as the ADMIRE aircraft model.
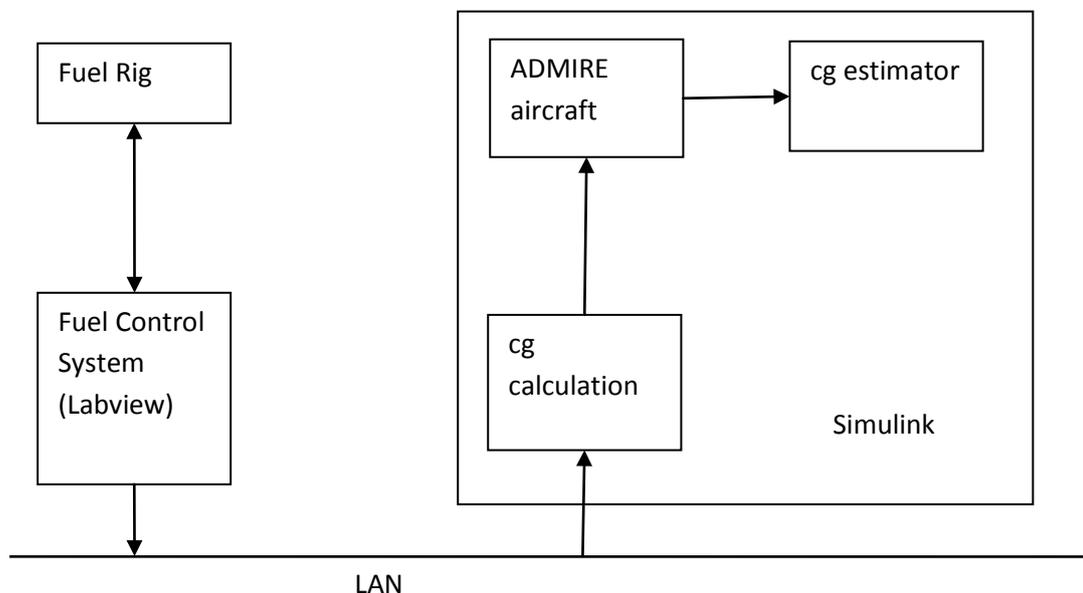
Figure 80: Configuration of BAE hardware fuel rig and cg estimator

8.2. Discrete CG Estimator Results

Figure 81 provides an example of the results from the estimator when a fuel leak was applied to the rig. Fuel was drained from a fuel tank to simulate a physical leak, however since this leak caused a very slow change in the cg then the fuel rig level, under the control of BAE personnel, was rescaled to make the change in cg more visible.

In this test the aircraft was trimmed in straight and level flight. The initial delay before an estimate is obtained is caused by the delay in obtaining fuel level measurements over the LAN.
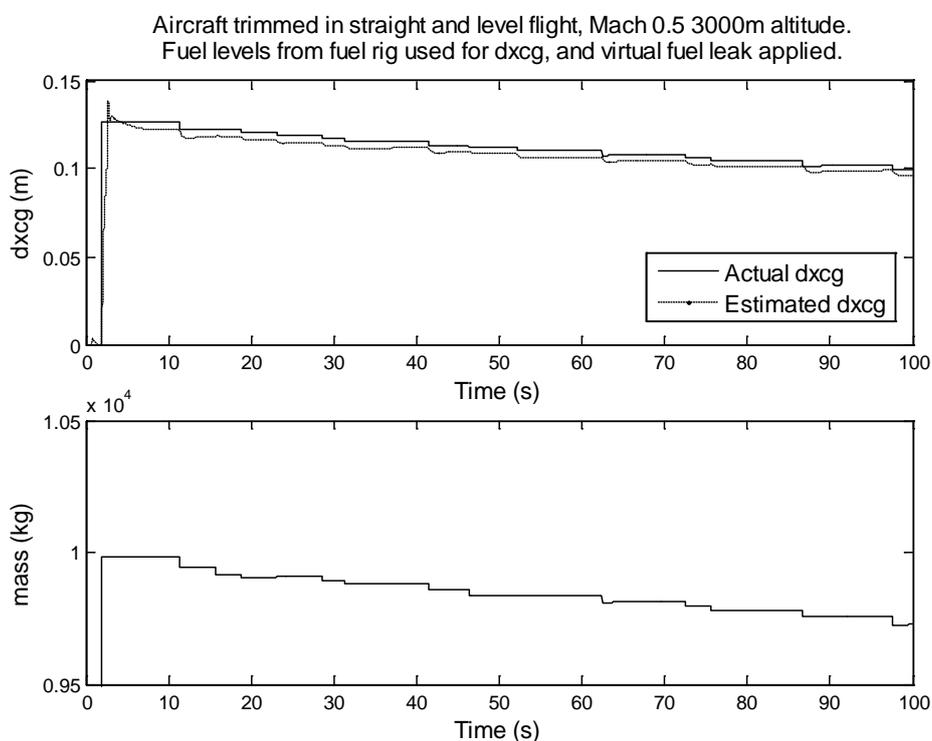
Figure 81: CG Estimate using hardware fuel rig with a 'virtual' fuel leak.

In a real aircraft fuel leak scenario the speed of the change in cg is obviously dependent upon the size of the leak and the distance of the leak from the cg.

To identify a fault condition such as a leak the cg estimate would need to be compared with the expected cg value obtained via the Fuel Control System. When the cg estimate differed from the expected cg value by a defined error tolerance level then a fault condition could be said to exist.

Other possible fault scenarios such as a faulty valve could also be detected by the cg estimator because the cg location would be different to the expected location. However different faults could cause the same error in cg position, for example a fuel leak to the rear of the aircraft would cause the cg location to be more negative than expected, and a faulty valve (less fuel flow to that expected) at the front of the aircraft would also cause a more negative than expected cg location. The cg estimator would be unable to distinguish between different types of fault without additional data, such as aircraft mass.

8.3 Summary

This section has shown that the Kalman-Bucy filter based cg estimator described in earlier sections can be converted into a discrete Kalman filter based cg estimator.

The hardware fuel rig has been used because it provides a more realistic example of the scale of centre of gravity changes that may be caused by faults in a fuel system. This part of the study also provided the opportunity to understand if any issues arise as a consequence of conversion to and implementation in discrete time. The experimentation has shown that fuel rig faults take a long time to manifest themselves in changes in the cg, and these cg changes can be quite small. Therefore in practice a cg estimator may take a long time to help confirm the presence of a fault. The study therefore helps in assessing the usefulness of a cg estimator to aid fuel system fault detection.

The general concept of detecting unexpected acceleration could be used to provide other fault detection capabilities. For example, if the estimator is modified to detect the unexpected yaw acceleration then this could be used to detect loss of engine thrust or a problem with an engine generating less thrust than it should.

A possible extension to this work would be a hardware implementation of the estimator based upon this discrete version, together with a more comprehensive analysis of its fault detection capabilities.

9. Extended Kalman Filter

The final section in this thesis considers the use of an Extended Kalman Filter (EKF) to perform the cg estimation. The EKF was proposed by Stanley F. Scmidt and has been called the "Kalman-Scmidt" filter (Grewal & Andrews, 2001: 178). As noted in section 2.1 the EKF is the most popular approach to state estimation in non-linear systems (Simon, 2006: 396).

Section 6.3 gave some examples of the non-linearities in the ADMIRE model, for example the effect of the angle of attack upon the pitching moment is non-linear, as is engine thrust when the engine afterburners are engaged. Therefore for a non-linear system such as the ADMIRE aircraft model using an EKF is a logical and standard approach that could be adopted for the state estimator in the cg estimation problem.

The main disadvantage of the EKF is "the added real-time computational cost of linearization" (Grewal & Andrew, 2001: 170). Essentially the EKF re-linearises the model on every iteration and then recalculates the Kalman gain on-line, whereas for the Kalman-Bucy and the discrete Kalman filter described previously the Kalman gain is calculated as part of the design process (i.e. off-line). The other main disadvantage with the use of an EKF in this specific research project is the additional complexity that it would add to the certification of a cg estimator in a real aircraft application.

The derivation of the EKF is not contained within this thesis but the equations of an EKF are obtained in numerous text books, e.g. (Lewis, 1986: 263), (Simon, 2006: 408), (Grewal & Andrews, 2001: 180).

The non-linear system defined below is based upon (Ribeiro, 2004) and (Grewal and Andrews, 2001: 180) :

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1} \qquad w_k \sim N(0, Q_k) \tag{92}$$

$$y_k = h_k(x_k) + v_k \qquad v_k \sim N(0, R_k) \tag{92}$$

where

x_k is the state vector at time step k

y_k is the output vector at time step k

$f(.)$ is the system function and $h(.)$ is the measurement function

w_k and v_k are white Gaussian independent random processes with zero mean and covariance matrix.

Let $F(k)$ and $H(k)$ be the Jacobian matrices of $f(.)$ and $h(.)$ denoted by:

$$F_{k-1} \approx \frac{\delta f_x}{\delta x} \text{ and } H_k \approx \frac{\delta h_x}{\delta x}$$

The following equations define the discrete time EKF.

$$\hat{x}_k^- = f_{k-1}(\hat{x}_{k-1}^+) \tag{94}$$

$$P_k^- = F_{k-1}P_{k-1}^+F_{k-1}^T + Q_{k-1} \tag{95}$$

$$K_k = P_k^-H_k^T[H_kP_k^-H_k^T + R_k]^{-1} \tag{96}$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k[y_k - H_k\hat{x}_k^-] \tag{97}$$

$$P_k^+ = (I - K_kH_k)P_k^- \tag{98}$$

In simple terms the algorithm used for the EKF has the following steps:

1. Linearise the system dynamics around the last state estimate (Matlab 'dlinmod' was used to do this). The use of the Matlab 'linmod' command is not a novel approach, it is used in EKF elsewhere, for example in (Devouassoux and Pritchett, 2001).
2. Apply the prediction step of the Kalman filter to the new linear model (i.e. calculate $\hat{x}_k^-$ and P_k^-).
3. Linearise the observation dynamics measurement equation.
4. Apply the update cycle of the Kalman filter to obtain K_k, $\hat{x}_k^+$, and P_k^+ (i.e. calculate the Kalman gain, correct the estimate by applying the Kalman gain to the residual, and then recalculate the error covariance matrix).

A full listing of the Matlab code is included in Appendix I.

The EKF that has been developed contains the 11 states defined in (71). The noise RMS used to obtain the variances used in the R matrix is defined in Table 8, and the values in the Q matrix are defined in (87).

The earlier Kalman-Bucy and Kalman filters used the Matlab 'lqe' or 'dlqe' commands to obtain the Kalman gain, this time the Kalman gain was calculated using (94) through to (98), although 'dlqe' could still have been used.

The extended Kalman filter is used to obtain an estimate for unexpected angular acceleration, which is then converted into an estimate of the cg change, see (70).

The structure of the longitudinal cg estimator is shown in Figure 82.
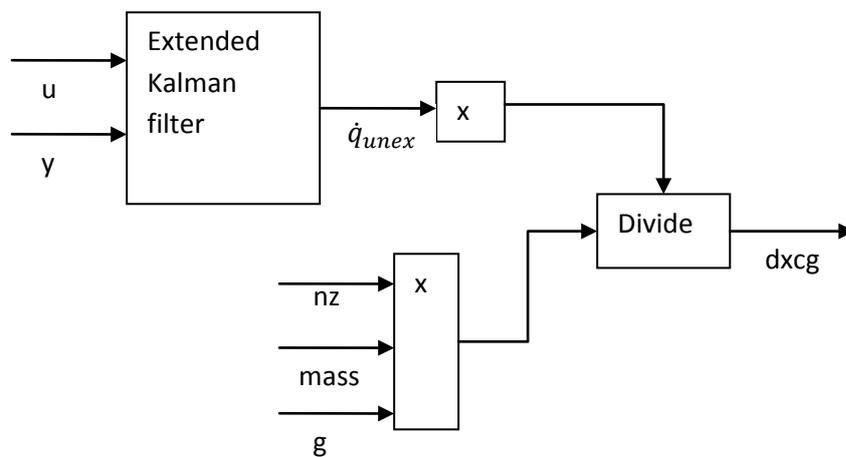
Figure 82: Structure of extended Kalman filter for longitudinal cg estimator

9.1 EKF Results

The next sequence of diagrams shows the results from the EKF. The top left graphs show the cg estimates and the static upper and lower acceptable limits specified by BAE Systems. Each aircraft manoeuvre occurs when the aircraft is trimmed at Mach 0.4 and at 5000m altitude, and after 10 seconds there is a cg shift of 0.1m, identical to the tests applied in section 6.4.1. Note that for clarity measurement noise has not been added to the signals making the results less noisy than they would otherwise be.

9.1.1 Pitch down

A maximum push down is commanded at 15.5 seconds over 5 seconds, and then the pitch stick is pulled back to level flight over another 5 seconds. The AoA and nz both pass through

zero and since dxcg is calculated by dividing by nz (70) then as nz decreases this magnifies any errors. The cg estimate stays within its static limits.

The comparison between the EKF and earlier coefficient-correction approach shows that the EKF is slightly more accurate.

Full push down at 15.5 seconds over 5 seconds then return to zero over 5 seconds.
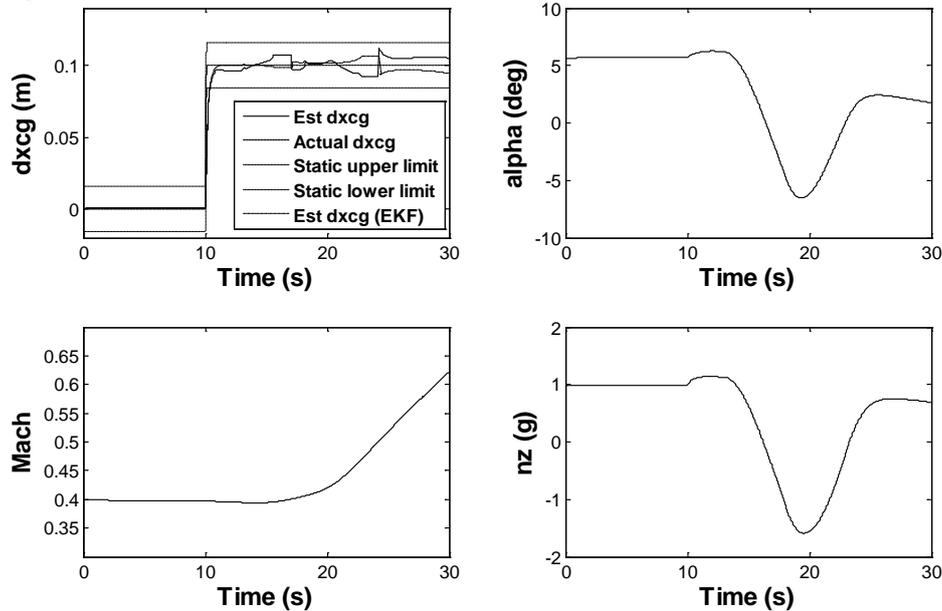
Figure 83: EKF – push down test.

9.1.2 Roll

The roll is commanded after 11 seconds and the aircraft banks at a rate of approximately 21 degrees/second and -29 degrees/second. The aircraft bank angle reaches about 50 degrees. As before the cg estimate remains well within specified limits.

The EKF estimate is very accurate compared to the coefficient-correction approach, because the EKF contains much more accurate coefficient derivatives during the bank manoeuvre.
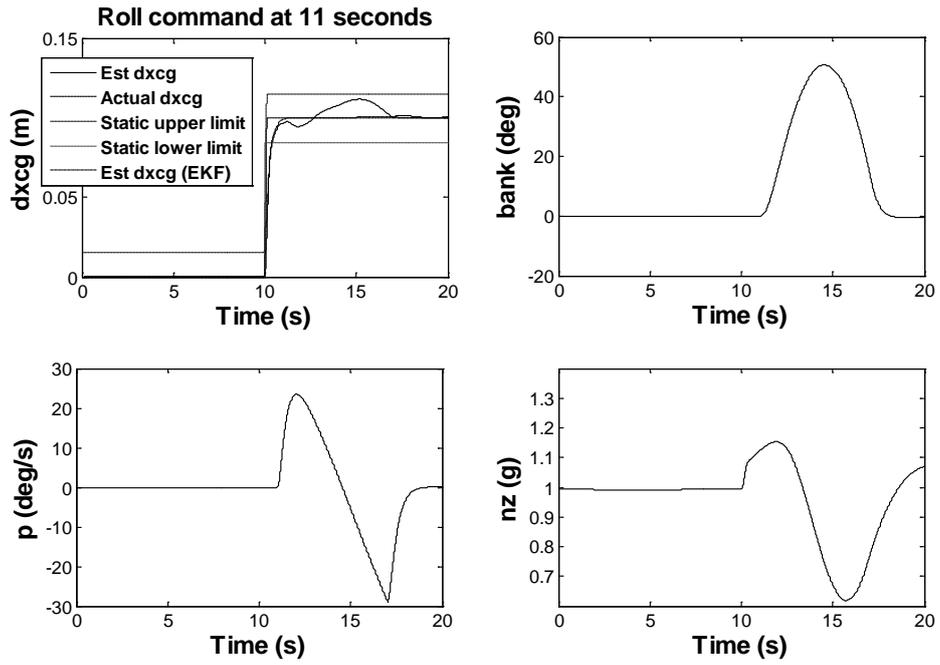
Figure 84: EKF – roll test.

9.1.3 Acceleration

The aircraft accelerates from Mach 0.4 to Mach 0.75. The cg estimate remains well within its static error limits and its estimate is superior to the coefficient-correction approach at speeds above Mach 0.6.
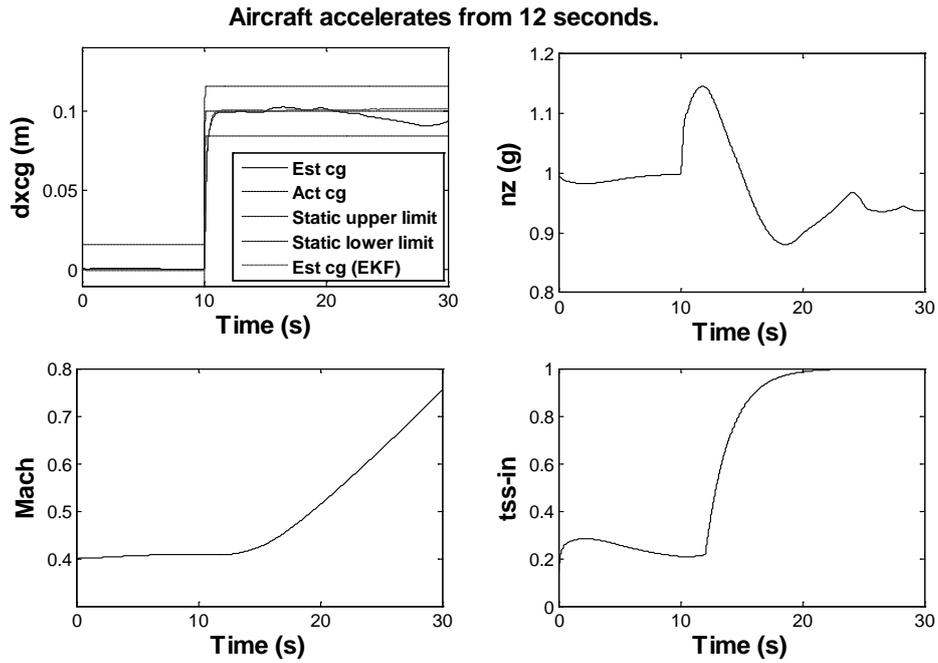
Figure 85: EKF – Acceleration test.

9.1.4 Lateral Acceleration

A rudder command occurs after 11 seconds and generates a value for vdot (lateral acceleration) of approximately -1.4 m/s^2. The effect upon the estimate of the cg is negligible, and the estimate remains well within limits.

In this scenario both the EKF and coefficient-correction method provide very good results.
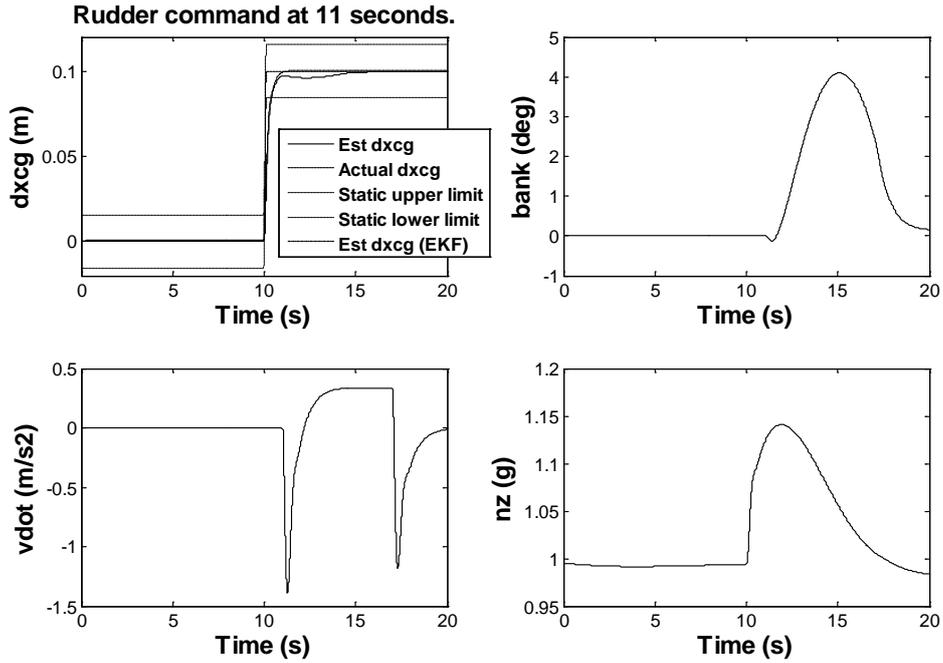
Figure 86: EKF – Lateral acceleration test.

9.2 Computational complexity comparison of EKF with coefficient correction approach

The computational load of the EKF version of the cg estimator is compared with the earlier coefficient correction approach.

It is assumed that the final EKF version of the estimator would not calculate its coefficients at run-time using a linearisation routine but would use pre-calculated data tables. The coefficient correction estimator in section 6.4 processed 10 data lookups for 7 command coefficients (B matrix) and 3 measurement coefficients (A matrix), see Table 14. The EKF version of the estimator would recalculate all of the matrices: 121 data lookups for the A matrix (11 by 11) and 110 data lookups for the B matrix (11 by 10), totalling 231 data lookups i.e. the EKF version will have approximately 23 times the number of extra data lookups to perform.

Most of the computation in both versions involves matrix operations, the following assumptions have been made about the computational complexity of matrix operations:

Square matrix multiplication of 2 nxn matrices $O(n^3)$

Rectangular matrix multiplication of nxm matrix by mxp matrix   O(mnp)

Matrix inverse of one nxn matrix $O(n^3)$

Matrix transpose : access same data but in different order

Matrix addition and subtraction has been ignored because it is low in comparison with multiplication.

These computational complexity values are based upon normal matrix multiplication. There are algorithms that provide improved matrix multiplication performance, for example for square matrix multiplication the Coppersmith-Winograd algorithm is $O(n^{2.376})$. The values for general matrix multiplication should be sufficient for a basic comparison between the two types of cg estimators.

The calculations are based upon the following sizes for the matrices:

F : nxn matrix

G : nxm matrix

H : mxn matrix

P : nxn matrix

Q : nxn matrix

R : mxm matrix

K : nxm matrix

x : nx1 vector

u : mx1 vector

y : mx1 vector

a) Coefficient correction estimator

This version uses a fixed Kalman gain, therefore only (7) and (10) of the discrete Kalman filter are applicable. Table 24 provides the equations for the discrete Kalman filter used in the selective coefficient correction version of the estimator, and its associated computational

complexity.

Table 24: Selective coefficient correction complexity

Equations	Computational complexity
$\hat{x}_{k+1}^- = F_k \hat{x}_k + G_k u_k$	$O(n^2) + O(mn)$
$\hat{x}_k^+ = \hat{x}_k^- + K_k[y_k - H_k \hat{x}_k^-]$	$2O(mn)$
Total	$O(n^2) + 3O(mn)$

b) EKF

Table 25 provides the equations for the EKF and its corresponding computational complexity.

Table 25: EKF complexity

Equations	Computational complexity
$\hat{x}_{k+1}^- = F_k \hat{x}_k + G_k u_k$	$O(n^2) + O(mn)$
$P_k^- = F_{k-1} P_{k-1}^+ F_{k-1}^T + Q_{k-1}$	$2O(n^3)$
$K_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1}$	$O(n^3) + 4O(mn^2)$
$\hat{x}_k^+ = \hat{x}_k^- + K_k[y_k - H_k \hat{x}_k^-]$	$2O(mn)$
$P_k^+ = (I - K_k H_k) P_k^-$	$2O(mn^2)$
Total	$3O(n^3) + 6O(mn^2) + 3O(mn) + O(n^2)$

In this thesis the matrix sizes have been m = 10, n = 11. For these matrix sizes the coefficient correction approach gives a computational complexity for matrix multiplication of 451 compared with the EKF value of 11704. Therefore matrix multiplication in the EKF version of the estimator is approximately 26 times more time complex compared with the coefficient correction approach.

9.3 Summary

Overall the EKF gave superior results compared to the coefficient correction approach, particularly with regard to the acceleration and roll manoeuvres. This was expected because the repeated re-linearisation in the EKF provides more accurate values for the coefficient derivatives in the estimator. This is particularly noticeable when the aircraft banks because the coefficient correction approach only contained data when the aircraft was in level flight. It is also particularly noticeable when the aircraft accelerated sharply and its afterburners were activated, again this is because the data tables in the coefficient correction approach did not contain data when the afterburners were engaged.

The main disadvantages with the EKF are the additional processing overheads involved with obtaining a linearised model at each iteration, and recalculating the Kalman gain and associated error covariances. However, in a practical implementation it is likely that the derivatives would be pre-computed and contained within tables to reduce the run-time processing overhead. An important disadvantage of the EKF for this specific research is the more complicated certification route (identified by BAE Systems) caused by the variable Kalman gain, in comparison to the coefficient-correction approach where the Kalman gain was precomputed.

The main advantage of the EKF is that it gives more accurate results and does not require the additional corrections that were applied to the coefficient-correction approach. The accuracy of the coefficient-correction approach could be improved by obtaining coefficient data when the aircraft is banked and when the afterburners are activated.

10. Conclusions and Future Work

This thesis has proposed the use of a Kalman filter to estimate the in-flight aircraft centre of gravity. The Kalman filter has been augmented to estimate unexpected aircraft acceleration and this has then been converted into an estimate of the change in longitudinal or lateral cg.

This approach has been applied to increasingly complex models and estimators. The first estimator used a linear model of the longitudinal state-space equations for a Phantom aircraft. This estimator was used to demonstrate that the augmented Kalman-Bucy filter could accurately estimate unexpected aircraft acceleration even when the aircraft went through a sequence of pitching manoeuvres.

The second model used was the more complex non-linear ADMIRE aircraft model, and the cg estimator demonstrated that it was still capable of estimating the cg despite the additional aircraft model complexity. Some of the non-linearities in the ADMIRE model were then investigated and the estimator was modified to use pre-calculated data tables to mitigate the effect of these non-linearities. The approach taken was to selectively modify some of the coefficients in the Kalman-Bucy filter based upon Mach, altitude and angle of attack, and this version of the estimator accurately estimated the cg (satisfying BAE Systems requirements) when the aircraft went through a variety of manoeuvres (apart from small deviations during high aircraft acceleration and sharp aircraft banking).

The estimator was also tested when the aircraft exceeded its specified manoeuvre limits. On this occasion the cg estimate went out of range but became accurate again when the out of range manoeuvre ended.

In practice a real cg estimator would not achieve the results shown because the model would not be an exact representation of the actual aircraft, therefore the effect of using incorrect aircraft coefficients, inertia and mass was examined to assess its effect upon the estimate.

A combined longitudinal and lateral cg estimator was then developed and used in conjunction with a BAE cg modifier model to assess its performance during the release of stores or during fuel sloshing. Again the cg estimator satisfied its performance requirements.

A discrete version of the estimator was developed and used with a hardware fuel rig to assess the performance of the estimator during various fault scenarios. The discrete cg estimator still obtained accurate estimates but it was noted that faults affecting the cg took a long time to

manifest themselves. The discrete version of the estimator could be used in a hardware implementation of the estimator.

The final version of the cg estimator used an Extended Kalman Filter. This version provided more accurate estimates compared to the earlier coefficient-correction approach but at the cost of more run-time computation and a more complex route to certification for flight use.

The literature survey did not specify the accuracy of many existing cg estimation techniques, see section 2.3, but it did indicate that any system should be accurate to at least 1% mean aerodynamic chord, and according to van Els (2007) any on-ground cg estimation system should be accurate within 1% of the mean aerodynamic chord. The requirements set by BAE Systems, see section 3.1, defined a static accuracy limit of 0.3% mac and a dynamic accuracy limit of 0.5% mac.

A key feature of this research has been the collaboration with BAE Systems who provided flight test data to obtain realistic measurement noise, and also specified the cg estimator requirements. In a military environment the deployment of stores and sloshing of fuel in its tanks can have a big effect upon the cg, therefore BAE provided a model for this which was used in the testing and development of the cg estimator.

This thesis has described the use of model based techniques to estimate aircraft cg. In particular it has described the novel method of using Kalman filters to obtain the unexpected angular acceleration caused by a change in cg, and then converting this unexpected angular acceleration into a cg estimate. The literature survey described different approaches taken to estimate the cg, with the more recent research focussing upon using neural nets, and Abraham and Costello (2009) using an extended Kalman filter to estimate the weight and cg of a helicopter. To the authors knowledge the research described in this thesis is the first research based upon using Kalman filters which have been augmented to estimate unexpected aircraft acceleration, which is then converted into an estimate of the change in longitudinal or lateral cg.

This thesis has also described a solution to the estimation problem applied to a non-linear aircraft model by selectively modifying the estimator pitch and roll moment coefficients based upon angle of attack, Mach and altitude, as opposed to the more traditional approach of using an Extended Kalman Filter (EKF). Throughout this thesis the static accuracy limit of

0.3% mac has been used in the results, and many examples have been provided. The results demonstrate the potential for a Kalman filter based cg estimator to provide accurate cg estimates for an in-flight aircraft.

10.1 Future Work

Before this technique could be implemented on an aircraft further study is required. In particular the following issues have been identified:

1. The effect of wind gusts on the estimate should be assessed by incorporating a model for the atmosphere. It is anticipated that wind gusts would affect the estimate but one approach to mitigate the effect may be to include a first order lag on the estimate.
2. A hardware implementation of the estimator should be developed to ensure that the design is practical.
3. A formal design process for the estimator should be investigated together with an understanding of the verification and validation issues that need to be satisfied so that the estimator is suitable for flight clearance.

Other potential areas of future work could involve:

1. Reducing the number of states to reduce the computation. For example the longitudinal cg estimator may be sufficiently accurate with just the α and q states, and the lateral cg estimator may be sufficiently accurate with just the β and p states.
2. Expanding the estimator to work in supersonic flight
3. Investigating the fault finding capabilities of an estimator and other potential uses from an accurate estimate of unexpected acceleration, for example estimating the unexpected yaw acceleration may be useful to detect engine problems.

References

Abraham, M. & Costello, M. (2009) In-flight estimation of helicopter gross weight and mass center location, *Journal of Aircraft*, vol. 46, No. 3, pp.1042-1049.

Baeten, A., & Stern, D. (2008) Prediction of Highly Dynamic Airborne Tank Fuel Sloshing and Impact Effects, *46th AIAA Aerospace Sciences Meeting and Exhibit*, Jan7-10, 2008.

Bar-Shalom, Y., Rong Li, X., & Kirubarajan, T. (2001) *Estimation with Application to Tracking and Navigation.* Wiley.

BEA (2003) Accident on 25 December 2003 at Cotonou Cadjehoun aerodrome (Benin) to the Boeing 727-223 registered 3X-GDO operated by UTA, *3x-o031225a.*

Bi, N., Haas, D., & McCool, K. (2004) Investigation of In-Flight Gross Weight and CG Estimation. *60th Annual Forum Proceedings - American Helicopter Society*, pp. 2244-2254.

Blakely, P., & Hedges, D. (1998) A Real-Time Application for Computing Gross Weight and Center of Gravity Measurements on Boeing Experimental Test Flights. *Proceedings of International Symposium, Instrumentation in the Aerospace Industry*, pp. 211-218.

Brockman, D. (1980) A Flight Test Real-Time CW-CG Computing System. *26th International Instrumentation Symposium*, pp. 583-589.

Carnegie Mellon University (1997) State Space Tutorial. Retrieved September 24, 2010, from http://www.engin.umich.edu/group/ctm/state/state.html.

Cook, M. (2007). *Flight Dynamic Principles.* Elsevier.

Cummins, J., Bering, A., Adams, D.E.,& Sterkenburg, R. (2009) Automated Estimation of an Aircraft's Center of Gravity Using Static and Dynamic Measurements. *Proc of the IMAC XXVI,* Feb 9-12, 2009.

Devouassoux, Y., & Pritchett, A. (2001) Application of Kalman Filtering to Pilot Detection of Failures. *Digital Avionics Systems Conference, 2001*, vol 1, pp.3D3/1-3D3/9.

Dutton, K., Thompson, S., & Barraclough, B. (1997) *The Art of Control Engineering.* Addison-Wesley.

Eykhoff, P. (1974) *System Identification.* Wiley.

Federal Aviation Administration (2004) *Airplane Flying Handbook, FAA-H-8083-3A.*

Federal Aviation Administration (2005) *Aircraft weight and balance control,* AC-120-27E.

Federal Aviation Administration. (2007) *Aircraft Weight and Balance Handbook, FAA-H-8083-1 A.*

Federal Aviation Administration (2008) *Aircraft onboard weight and balance systems*, AC-20-161.

Filippone, A. (2006) *Flight Preformance of Fuxed and Rotary Wing Aircraft.* Elsevier.

Forsell, L., & Nilsson, U. (2005) ADMIRE The aero-data model in a research environment version 4, *FOI-R-11624-SE.*

Friedland, B. (1986) *Control System Design.* McGraw-Hill.

Geistware (2010, March 19) *Aircraft axes.* Retrieved March 19, 2010, from Geistware: http://www.geistware.com/rcmodeling/articles/aerodynamics/axis_definition.gif.

Glover, J. (1985) *Patent No. 4545019.* USA.

Grewal, M., & Andrews, A. (2001) *Kalman Filtering: Theory and Practice using Matlab.* Wiley.

Hutchinson, C. (1984) The Kalman Filter Applied to Aerospace and Electronic Systems. *IEEE Trans. on Aerospace and Electronic Systems, vol 20, no. 4* , pp.500-504.

Idan, M., Iosilevskii, G., & Nazarov, S. (2004) In-Flight Weight and Balance Identification using Neural Networks. *Journal of Aircraft* , pp.137-143.

Jazwinski, A. H. (1970) *Stochastic Processes and Filtering Theory.* Academic Press.

Kalman, R. (1960) A New Approach to Linear Filtering and Prediction Problems. *Trans. ASME Journal of Basic Engineering, vol 28D* , pp.35-45.

Kalman, R., & Bucy, R. (1961) New Results in Linear Filtering and Prediction Theory. *Trans. ASME Journal of Basic Engineering, vol 83D* , pp.95-108.

Lewis, F. (1986) *Optimal Estimation.* Wiley.

McGee, L.A., & Schmidt. S.F. (1985) Discovery of the Kalman filter as a practical tool for aerospace and industry, *NASA Technical Memorandum 86847*.

NASA. (2006, March 14) *Aircraft rotations*. Retrieved June 6, 2008, from NASA Glenn Research Center: http://www.grc.nasa.gov/WWW/K-12/airplane/rotations.html.

Orgun, M., & Flannigan, S.J. (1991) *Patent No. 5034896.* USA.

Phillips, W. (2004) *Mechanics of Flight.* Wiley.

Raol, J., Girija, G., & Singh, J. (2004) *Modelling and Parameter Estimation of Dynamic Systems.* IEE Control Series 65.

Ribeiro, M.I. (2004) Kalman and Extended Kalman Filters: Concept, Derivation and Properties. Retrieved December 22, 2010, from http://users.isr.ist.utl.pt/~mir/pub/kalman.pdf.

Simon, D. (2006) *Optimal State Estimation: Kalman, H infinity, and non-linear approaches*. Wiley.

Stevens, B., & Lewis, F. (2003) *Aircraft Control and Simulation.* Wiley.

Van Es, G.W.H. (2007) Analysis of aircraft weight and balance related safety occurrences. *19th European Aviation Safety Seminar (EASS)*, NLR-TP-2007-163, National Aerospace Laboratory NLR.

Welch, G., & Bishop, G. (2006) *An Introduction to the Kalman Filter.* Department of Computer Science, University of North Carolina.

Wikipedia (2010, March 19) *Aircraft in level flight*. Retrieved March 19, 2010, from en.wikipedia.org:
http://upload.wikimedia.org/wikipedia/commons/thumb/7/7d/AirStability.svg/760px-AirStability.svg.png.

Wu, T.-H. (1996) *Patent No. 5571953.* USA.

Zhang, J., Yang, L., & Shen, G. (2009) Modeling and attitude control of aircraft with center of gravity variations, *IEEE Aerospace Conference*, vol 1, pp.3108-3118.

APPENDICES

Estimation of Aircraft Centre of Gravity using Kalman Filters

Andrew Stanley and Roger Goodall

Abstract— **The centre of gravity (cg) of an aircraft is of fundamental importance to its stability, but it is difficult to measure precisely in-flight because it will vary as the aircraft manouevres and fuel is consumed and sloshes in its tanks. This paper describes the development of a longitudinal cg estimator using Kalman filters. An example is provided using a linear Phantom F4-C aircraft model in which an unexpected moment representing a cg change is injected into the aircraft and then estimated using a Kalman filter. The work is then extended to use a more comprehensive aircraft model, ADMIRE, and the same approach is demonstrated using a linearised version of this model.**

I.   INTRODUCTION

THE aircraft centre of gravity (cg) location is an important piece of data because it is the location about which an aircraft rotates [1]. The aircraft performance is dependent upon the cg location because all moments are derived with respect to the cg. The ideal location for the centre of gravity is carefully determined by aircraft designers as it affects aircraft stability. The longitudinal cg is important due to its effect upon aircraft stability, the lateral cg is also important due to its effect upon aircraft performance. Uneven loading of fuel or cargo may cause the aircraft to be heavier on one side which requires additional lift on the heavy side of the aircraft. However this has the disadvantage of producing additional drag so the aircraft is less fuel efficient.

This paper describes the use of Kalman filters to estimate the aircraft cg location. The potential benefits of this approach are, in general, enhanced flexibility in the Flight Control System (FCS) design, and in particular :

a) It provides an additional method to estimate the cg when other methods e.g. accelerometers, fuel flow measurements, have failed.
b) It provides an additional piece of data to the FCS to improve fault detection e.g. confirm presence of a fuel leak.
c) It potentially allows for cg position dependent design scheduling, resulting in a less conservative FCS design that has to accommodate much smaller cg changes.
d) It provides the possibility to improve aircraft handling with cg, e.g. forward cg configurations are typically more sluggish in pitch manouevres.

II.   CURRENT TECHNIQUES AND RESEARCH

This section describes current techniques that are used to estimate the cg, both on the ground and in-flight.

When the aircraft is on the ground the aircraft can be weighed on scales. The arms (distance between the cg of the item and a datum point) of the weighing points are specified for the aircraft [2]. The moment is the weight multiplied by the arm. The cg is calculated as:

$$cg = total\,moment\,/\,total\,weight \qquad (1)$$

There are different systems for obtaining the weight and cg of an aircraft on-ground, for example by measuring the pressure within landing gear struts, however this paper will concentrate on in-flight cg estimation.

When the aircraft is in flight a number of different approaches have been used to estimate the cg. The simplest method involves dividing the aircraft into nodes and specifying the weight and cg location for each node. As the aircraft flies fuel is consumed which reduces the weight in the fuel tank nodes and the cg is recalculated using (1). This approach is described in [3] and then updated in [4] to make an allowance for fuel sloshing.

An alternative method for in-flight estimation is to use accelerometers at the front and rear of the aircraft [5]. The accelerometer signals undergo band-pass and low-pass filtering and are combined to obtain a signal representing the cg.

A method patented by Boeing [6] describes a cg estimation system used when a commercial aircraft is descending to land. This method uses many parameters to estimate the cg: angle of attack, flap setting, aircraft weight, elevator and stabilizer positions, load factor and dynamic pressure on aircraft, reference wing area. An obvious limitation of [6] is that it is only useful when the aircraft is in a stable configuration in its landing phase.

The most recent research in this field has focused upon using neural networks to estimate aircraft weight and cg. One paper [7] describes a neural net trained to estimate the weight and longitudinal cg for a small business jet in trimmed flight. It provides the theory to show that the longitudinal cg is a function of the dimensionless total pitch couple C_M and normal force C_Z, and these in turn are functions of the Mach number, angle of attack, elevator deflection, flight path angle, and normal cg z_{cg}. The authors claims the neural net achieved an accuracy of 1.6% of mean aerodynamic chord (mac) and 99% of the points were within 1% mac.

Alternatively [8] describes using a neural net to estimate the weight and cg in a V-22 tilt-rotor aircraft. The authors

claim the cg estimate had a root mean square (rms) error of 4.318mm (0.17 inches).

III. METHODOLOGY

It has already been noted that an aircraft rotates about its centre of gravity. The model-based method described in this paper is essentially based upon detecting unexpected angular acceleration. Providing that all possible causes of the acceleration (apart from the cg change) have been allowed for in the model then there is a relationship between the unexpected acceleration and a change in the cg.
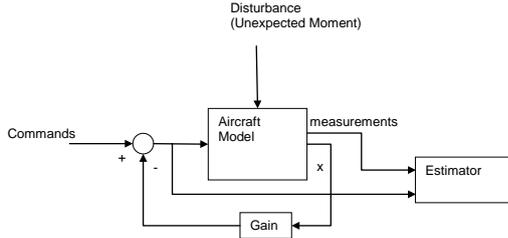

Fig. 1. Aircraft Model with model-based Estimator.

Fig. 1 shows the structure of the system.

An aircraft operates in a noisy environment i.e. its measurements will be subject to noise. A Kalman filter is the optimal estimator for linear systems with stochastic noise. The Kalman filter [9] has been used in many aerospace applications [10], [11]. The Kalman filter was extended to work in continuous systems with the Kalman-Bucy filter [12]. A Kalman-Bucy filter has been used as the estimator in this paper, and such a filter is shown in Fig. 2.
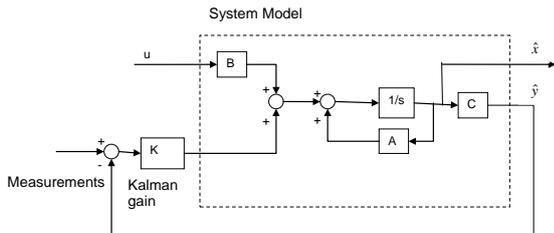

Fig. 2. Kalman-Bucy Filter.

A Kalman filter executes on a prediction-correction cycle. It contains a model of the system and uses the control inputs 'u' to make a prediction of the system state. It then uses measurements from the system to correct its state.

The Kalman gain is derived from the Q and R matrices that have been defined for the Kalman filter. A good introduction to the Kalman filter and explanation of the Q and R matrices is contained in [15]. The data in the R matrix are based upon

the variance of the measurement noise. The data in the Q matrix are more subjective and were obtained on an iterative trial and error basis, starting from sensible initial values to obtain the desired performance. The values in the Q and R matrices are used to calculate the Kalman gain: low values in Q (or high values in R) result in a small Kalman gain which gives more emphasis to the model, whereas a large gain gives greater emphasis to the measurements.

The Kalman gain used in this paper has been obtained by using the Matlab 'lqe' function.

This paper will describe the creation of the estimator and the calculation of the unexpected moment which is representative of a cg shift.

A. Theory and Assumptions

This section will develop the theory which has been used to relate a change in the longitudinal centre of gravity to unexpected pitch acceleration. The aircraft is assumed to be trimmed and in straight and level flight, therefore the forces of weight, lift, thrust and drag balance each other out. Any deviation from trimmed flight is caused by an unexpected moment. The unexpected moment may be caused by forces external to the aircraft such as a gust of wind, but for now the unexpected moment is assumed to be caused by a movement in the longitudinal cg. The effects of external forces will assessed at a later date.

Fig. 3 shows the forces and moments applied to a trimmed aircraft in straight and level flight. The variables in the figure and the pitching equation (2) are based on data
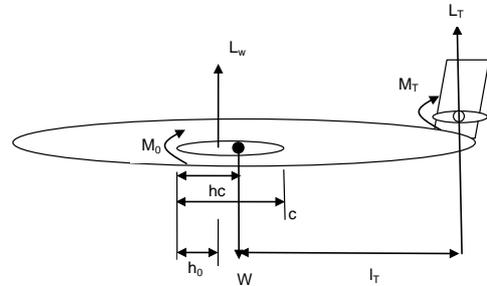
Fig. 3. Forces and Moments on Horizontal Trimmed Aircraft

contained in [13].

$$M = M_0 + L_w(h - h_0)c - L_T l_T + M_T \tag{2}$$

where
M_0 = wing pitching moment about aerodynamic centre
M_T = tailplane pitching moment
L_W = wing lift at aerodynamic centre
L_T = tailplane lift
c = mean aerodynamic chord
W = weight
$h_0 c$ = aerodynamic centre on mac
hc = cg position on mac
l_T = distance from tailplane lift to cg

The assumption is made that both M_0 and h_0 are constants. It is also assumed that the tailplane aerofoil is symmetric so M_T is zero.

In a trimmed aircraft the pitching moment is zero.

$$M_0 + L_w(h - h_0)c - L_T l_T = 0 \tag{3}$$

Equation (3) can be rearranged to give the cg location.

$$hc = h_0 c + (L_T l_T - M_0)/L_w \tag{4}$$

In the trimmed horizontal flight condition it is assumed that the distance from L_T to h_0 is fixed. Let d equal the distance from L_T to h_0 which allows the term $L_T l_T$ to be redefined.

$$L_T l_T = L_T(d - (h - h_0)c) \tag{5}$$

Equation (5) is substituted into (4) to give (6).

$$hc = (L_W h_0 c + L_T h_0 c + L_T d - M_0)/(L_w + L_T) \tag{6}$$

The aircraft is stable and trimmed but now encounters a pitching moment M due to a cg shift.

$$M_0 + (L_W(h - h_0)c - L_T l_T = M \tag{7}$$

If the definition for $L_T l_T$ in (5) is substituted into (7) it can be rearranged to give (8) to define the new cg location.

$$hc = (M - M_0 + L_W h_0 c + L_T h_0 c + L_T d)/(L_w + L_T) \tag{8}$$

If we assume that the lifts L_W, L_T and length d are unchanged then a comparison between (8) and (6) shows that the longitudinal centre of gravity has moved by $M/(L_W + L_T)$.

$$\Delta cg = M/(L_W + L_T) \tag{9}$$

In trimmed straight and level flight $L_W + L_T$ equals the weight of the aircraft which is its mass multiplied by gravity.

$$\Delta cg = M/(massg) \tag{10}$$

From Newtonian mechanics the moment can be defined in terms of angular acceleration and moment of inertia (11).

$$M = I_y \dot{q} \tag{11}$$

where
$M = $ pitching moment
$I_y = $ pitch moment of inertia
$\dot{q} = $ pitch acceleration

Equation (11) is a simplified pitching moment equation which only considers the effect of $\dot{q}$ on M. The complete pitching moment equation defined in [13] considers forces acting in all three axes and is more complex. The ADMIRE model contains the complete pitching moment equations, therefore the full equations will be analysed using the ADMIRE model.

Equation (11) is substituted into (10) to define the cg change in terms of angular acceleration, moment of inertia and mass.

$$\Delta cg = I_y \dot{q}/(massg) \tag{12}$$

Therefore, given the assumptions and simplifications that have been described, a change in the longitudinal cg can be estimated from the pitch acceleration, pitching moment of inertia and mass of an aircraft.

B. Phantom F4-C CG Estimation – Open Loop

A McDonnell F4-C Phantom aircraft model defined in [13] was used to evaluate the use of Kalman filters to detect and estimate unexpected pitch acceleration, and hence estimate the cg change. This model was chosen because the aircraft mass and pitching moment of inertia required in (12) are defined, and also the dimensionless longitudinal derivatives are defined which can be varied to test the estimator. The state-space equation for the Phantom model contained in [13] is shown in (13).

The mass of the Phantom aircraft is specified as 17642 kg, and the pitching moment of inertia is 165669 kg m^2.

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0.007181 & 0.00457 & -29.072 & -9.678 \\ -0.0687 & -0.2953 & 174.868 & 1.601 \\ 0.00173 & -0.0105 & -0.4462 & 0.001277 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} 1.041 \\ -6.294 \\ -4.888 \\ 0 \end{bmatrix} [\eta] \tag{13}$$

where
$u = $ axial velocity
$w = $ normal velocity
$q = $ pitch rate
$\theta = $ pitch angle
$\eta = $ elevator angle

The aircraft model was configured to accept the injection of an unexpected moment as shown in Fig. 1, except in this scenario the aircraft is open-loop. The unexpected moment is calculated from the cg shift and the mass of the aircraft (10).

Equation (12) is reordered to give the unexpected pitch acceleration caused by a cg shift, and the mass and pitching moment of inertia for the Phantom aircraft are used.

$$\dot{q} = 17642 \times 9.81 \Delta cg / 165669$$
$$= 1.04466146 \Delta cg \tag{14}$$

The Kalman filter contains the Phantom F4-C aircraft model and is augmented to estimate the change in the centre of gravity using (14), to convert the unexpected pitch acceleration to a change in cg. This method for modeling the unknown acceleration is adapted from an example contained in [16], in which unknown acceleration is modeled as a random process.

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \\ \Delta\dot{cg} \end{bmatrix} = \begin{bmatrix} 0.007181 & 0.00457 & -29.072 & -9.678 & 0 \\ -0.0687 & -0.2953 & 174.868 & -1.601 & 0 \\ 0.00173 & -0.0105 & -0.4462 & 0.001277 & 1.04466146 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \\ \Delta cg \end{bmatrix} + \begin{bmatrix} 1.041 \\ -6.294 \\ -4.888 \\ 0 \\ 0 \end{bmatrix} [\eta]$$

(15)

The data in Table 1 were used to tune the Kalman filter. The maximum and average values were selected as a best guess to progress the development of the filter. The measurement noise defined in Table 1 was added to the aircraft model.

TABLE 1
PHANTOM F4-C KALMAN FILTER TUNING DATA

	u (m/s)	w (m/s)	q (°/s)	θ (°)
Max value	100	30	30 (0.5236 rad/s)	30 (0.5236 rad)
Average value	20	10	5 (0.08727 rad/s)	5 (0.08727 rad)
Measurement Noise (0.2% of Maximum)	0.2	0.06	0.06 (0.0010472 rad/s)	0.06 (0.0010472 rad)
Noise variance	0.04	0.036	0.036 (1.0966x10⁻⁶ rad/s)	0.036 (1.0966x10⁻⁶ rad)

The R matrix in the Kalman filter was calculated from the noise variance of the measurement noise.

$$R = diag([0.04 \quad 0.0036 \quad 0.0000010966 \quad 0.0000010966])$$ (16)

The Q matrix in the Kalman filter was originally based upon the average values defined in Table 1 but amended in an iterative manner to improve the filter performance.

$$Q = diag([40 \, 10 \, 0.00008 \, 0.00008 \, 0.001])$$ (17)

Fig. 4 provides an example of the performance of this estimator. In this example there is a 0.5m change in the centre of gravity after 1 second, and then after 10 seconds the elevator moves by 0.2 radians.

The estimate of the change in cg can be seen to climb to 0.5m after 1 second and it is unaffected by the elevator movement at 10 seconds. Fig. 4B provides a zoomed in view of the estimate to show the effect of noise.

Robustness tests were performed to examine how the estimator behaves when the model is not an exact representation of the aircraft. The dimensionless derivatives defined in [13] were increased in turn by 10% and the estimate repeated and measured after 20 seconds.

The X and Z derivatives had no effect upon the final estimate that was obtained. However the M derivatives, and data used to calculate the matrix coefficients, namely velocity, mass and moment of inertia, can be seen to affect the estimate, as shown in Table 2.

Fig. 5 gives an example of the transient performance of the estimator when the parameters are varied one at a time.
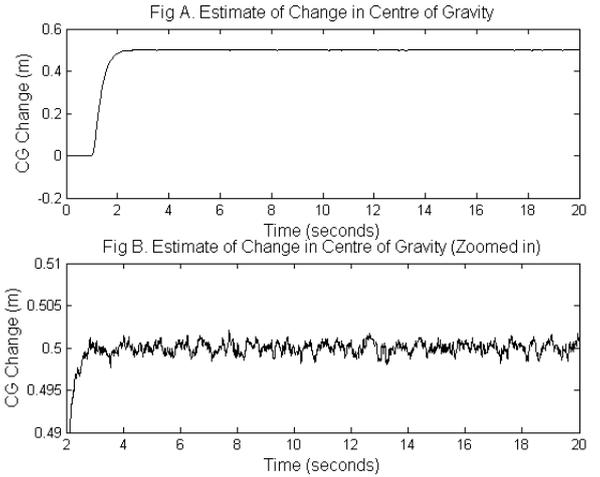
Fig. 4. Estimation of centre of gravity change from open-loop Phantom F4-C Aircraft Model

TABLE 2
PHANTOM F4-C MODEL ROBUSTNESS TESTING RESULTS

Dimensionless Derivative	Original Value	Test Value	Final Δcg	Final value % error
M_u	0.034	0.0374	0.4978	0.44
M_w	-0.2169	-0.23859	0.5379	7.58
Mw-dot	-0.591	-0.6501	0.5017	0.34
M_q	-1.2732	-1.40052	0.503	0.6
M_n	-0.5581	-0.61391	0.4069	18.62
Velocity	178	195.8	0.3606	27.9
Mass	17642	19406.2	0.5505	10.1
Moment of Inertia	165669	182235.9	0.5005	0.1

The estimate is most sensitive to changes in velocity, M_n, mass and M_w, the other changes reduced the accuracy of the estimate by less than 1%. There is a transient effect upon the estimation and this is clearly illustrated by the effect of the pitching moment of inertia I_y.
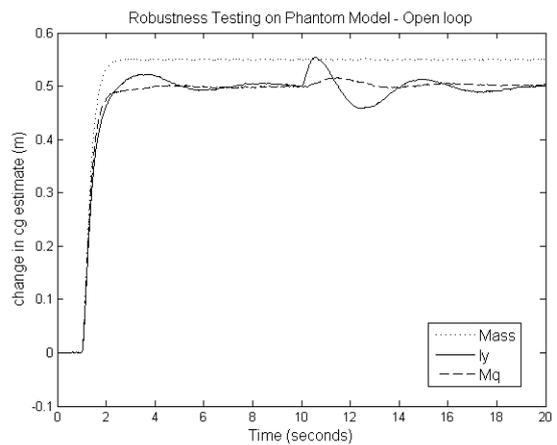
Fig. 5. Robustness tests of cg estimation on open-loop Phantom F4-C model..

The robustness tests have shown that special care should be taken to obtain accurate M derivatives, specifically M_n and M_w, and also the values for velocity and mass.

C.   Phantom F4-C CG Estimation – with control

A PID pitch rate controller was added to the model and the pitch rate was ramped up and down as shown in Fig. 6A. The Q and R matrices used to obtain the Kalman gain were the same as those defined in (16) and (17), and the noise added to the measurements is defined in Table 1.

A moment corresponding to a cg shift of 0.2m was inserted after 3 seconds, and the cg estimate is shown in Fig. 6B.

Fig. 6. Phantom F4-C pitch rate with PID controller included.

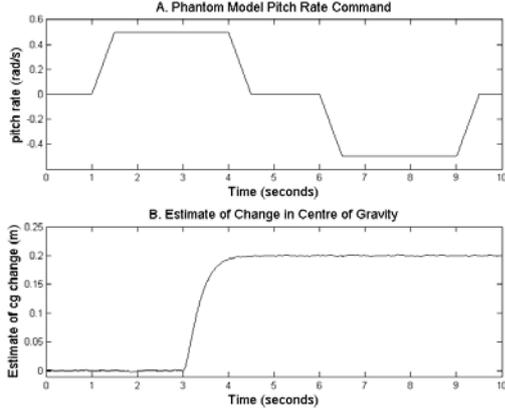The estimation results using the Phantom F4-C aircraft model have demonstrated that a Kalman-Bucy filter can be easily augmented to provide an accurate estimate for the unexpected pitch acceleration. The theory provided the logic to convert the unexpected acceleration into a cg shift. The next section will use a comprehensive aircraft model so that the effects of a cg change are more accurately modelled and incorporated into the estimator.

D.   ADMIRE

The ADMIRE [14] (Aero-Data Model In a Research Environment) aircraft model is a generic model of a small single-seat fighter aircraft with a delta-canard configuration. It contains twelve states (velocity (V_T), angle of attack (α), sideslip (β), roll rate (p_b), pitch rate (q_b), roll angle (φ), pitch angle (θ), yaw angle (ψ), longitudinal coordinate (x_v), lateral coordinate (y_v), normal coordinate (z_v)) plus additional states for sensors, actuators and FCS.

The work with the Phantom model assumed that a longitudinal cg change only affected $\dot{q}$, however a cg change will affect other state derivatives and with the ADMIRE model it will be possible to more accurately assess this impact and generate a more realistic estimator.

In the ADMIRE model the pitching moment caused by a cg shift is dependent upon the vertical force F_z, which in turn affects the angular acceleration.

$$\Delta M = \Delta cg\, F_z \tag{18}$$

Equation (18) is substituted into (11) and rearranged to give (19).

$$\Delta cg = \dot{q} I_y / F_z \tag{19}$$

The vertical force F_z is a variable in ADMIRE, therefore this equation is non-linear but can be linearised using a Taylor series approximation. Note that although F_z is an output in the ADMIRE model n_z has been used instead, because ADMIRE provides a sensor model for n_z .

The relationship between F_z and n_z is given in (20).

$$n_z = -F_z /(mass\, g) \tag{20}$$

The Taylor series approximation is in (21).

$$cx / y \approx cx_0 / y_0 + c(x - x_0) / y_0 - cx_0(y - y_0)/ y_0^{\,2} \tag{21}$$

where

$c = I_y/(mass\ g)$

$x = \dot{q}$

$x_0 = \dot{q}$ *operating point*

$y_0 = n_z$ *operating point*

For a trimmed aircraft the pitch acceleration $\dot{q}$ is zero, therefore the operating point x_0 is zero which simplifies (21) to (22).

$$cx / y \approx cx / y_0 \tag{22}$$

In terms of the cg shift this is

$$\Delta cg \approx \dot{q} I_y /(mass\, g\, nz_0) \tag{23}$$

A linear version of the ADMIRE model was used with the cg estimator, and a moment was injected into the aircraft model based upon a cg shift. The complete model contains the aircraft Flight Control System (FCS) as shown in Fig. 7.
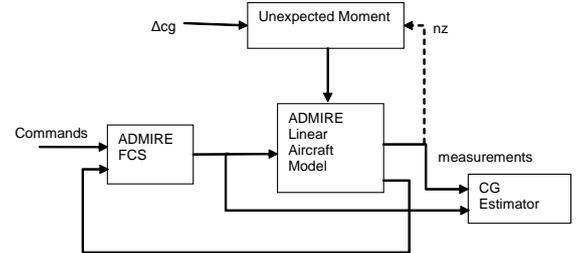

Fig. 7. Linear ADMIRE model with Injected Unexpected Moment.

The moment injected into the model is factored by n_z, as per (18) and (20).

In the following example the aircraft was first trimmed at mach 0.8 in straight and level flight at an altitude of 3000m before the model was linearised. In this scenario the

operating point used for nz was 0.9242 "g". The estimator was created by augmenting the 12 state ADMIRE model with an additional state for Δcg, in the same way as the Phantom F4-C model in (15). The new $\dot{q}_{\Delta cg}$ matrix coefficient value was calculated from (24).

$$\dot{q} \approx \Delta cg\, nz_0\, mass\, g\, /\, I_y \qquad (24)$$

where
$n_{z0} = 0.9242\ g$
$mass = 9100\ kg$
$g = 9.81\ m/s^2$
$I_y = 81000\ kg\ m^2$

The noise added to the ADMIRE aircraft model measurements is defined in Table 3. The noise variance is used to populate the R matrix of the Kalman-Bucy filter.

TABLE 3
ADMIRE NOISE DATA

	Measurement Noise (RMS)
Velocity (m/s)	0.313
α, β (rad)	0.00086
p, r (rad/s)	0.0136
q (rad/s)	0.00497
φ, θ, ψ (rad)	0.00086
x, y, z (m)	0.1732

As before the Q matrix (25) was obtained by trial and error from a sensible initial set of values.

$$Q = diag([1\ 0.05\ 0.05\ 0.01\ 0.01\ 0.01\ 0.05\ 0.05\ 0.05\ 1\ 1\ 1\ 1\ 0.1]) \qquad (25)$$

Fig. 8 shows the estimator response when an unexpected moment corresponding to a cg shift of -0.1m is injected into the linear ADMIRE aircraft model after 5 seconds.
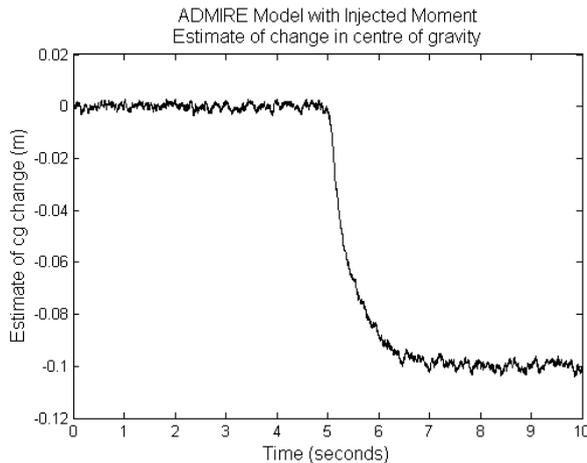

Fig. 8. CG Estimate using linear ADMIRE model and injected moment.

This section has shown that the concept of inserting an unexpected moment representative of a cg change, and then estimating the cg change from the actual and expected measurements can be applied to a complex linear aircraft model.

IV. CONCLUSIONS

This paper has presented the idea of using Kalman-Bucy filters to estimate changes in aircraft cg by measuring angular acceleration, and then comparing it with the expected angular acceleration.

It gave examples using Phantom F4-C and ADMIRE aircraft models, and described the state augmentation necessary to estimate the cg change. The Kalman-Bucy filters have shown promising results in detecting unexpected acceleration and converting this to an estimate of a cg change.

Future work will use the non-linear ADMIRE model which contains the cg as an input, thus removing the need for the artificial injection of an unexpected moment. The estimator will require modification to operate in the full flight envelope, possibly by scaling estimates based upon changes in the dynamic pressure or by upgrading to an Extended Kalman filter.

A particulare feature will be the need to accommodate the effect of aircraft trim offsets upon the accuracy of the estimation process.

Future work will also consider the sensor models in the estimator so that the measurements more realistically match those of an actual aircraft.

The final area of research will be to include an estimate for the lateral cg, and to investigate the coupling of the longitudinal and lateral axes.

REFERENCES

[1] NASA "Aircraft rotations". Retrieved June 6, 2008, from NASA Glenn Research Center:http://ww.grc.nasa.gov/WWW/K-12/airplane.html
[2] Federal Aviation Administration, Aircraft Weight and Balance Handbook, FAA-H-8083-1 A, 2007.
[3] D. Brockman, "A flight test real-time CW-CG computing system", 26th International Instrumentation Symposium, 1980, pp 583-589.
[4] P. Blakely & D. Hedges, "A real-time application for computing gross weight and center of gravity measurements on Boeing experimental test flights", Proceedings of International Symposium, Instrumentation in the Aerospace Industry, 1998, pp 211-218.
[5] J. Glover, Patent 4545019 USA, 1985.
[6] T. H. Wu, Patent 5571953 USA, 1996.
[7] M. Idan, G. Iosilevskii & S. Nazarov, "In-flight weight and balance identification using neural networks", Journal of Aircraft, 2004, pp 137-143.
[8] N. Bi, D. Haas & K. McCool, "Investigation of in-flight gross weight and cg estimation", 60th Annual Forum Proceedings – American Helicopter Society, 2004, pp 2244-2254.
[9] R. Kalman, "A new approach to linear filtering and prediction problems", Trans. ASME Journal of Basic Engineering, vol 28D, 1960, pp 35-45.

[10] C. Hutchinson, "The Kalman filter applied to aerospace and electronic systems", IEEE Trans. On Aerospace and Electronic Systems, vol 20, no 4, 1984, pp 500-504.

[11] L.A.McGhee & S.F.Schmidt, "Discovery of the Kalman filter as a potential tool for aerospace and industry", NASA-TM-86847, 1985.

[12] R. Kalman & R. Bucy, "New results in linear filtering and prediction theory", Trans. ASME of Basic Engineering, vol 83D, 1961, pp 95-108.

[13] M. Cook, "Flight Dynamics Principles", Elsevier, 2007.

[14] L. Forsell & U. Nilsson, "ADMIRE The aero-data model in a research environment version 4.0, model description", FOI-R—11624—SE, Swedish Defence Research Agency, 2005.

[15] G. Welch & G. Bishop, "An introduction to the Kalman filter", Retrieved October 21st, 2008, from University of North Carolina :http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf

[16] B. Friedland, "Control system design. An introduction to state-space methods", McGraw-Hill, 1986.

Estimation of the Centre of Gravity of a Manoeuvring Aircraft using Kalman filters and the ADMIRE aircraft model

Andrew J. Stanley, Roger M. Goodall

*Systems and Control Group, Loughborough University, LE11 3TU
UK (e-mail: A.J.Stanley@lboro.ac.uk, R.M.Goodall@lboro.ac.uk).*

Abstract: Previous work has shown that the centre of gravity (cg) of an aircraft can be estimated using a Kalman-Bucy filter. This paper will adopt a similar approach but use a non-linear aircraft model, and will demonstrate the estimation of the cg when the aircraft undergoes a series of manoeuvres. To cope with the non-linearities in the aircraft model the Kalman-Bucy filter coefficients for the pitching moment are modified when the aircraft manoeuvres. A number of examples are provided using the non-linear ADMIRE aircraft model, which is a complex model of a delta-canard military aircraft.

Keywords: aircraft control, centre of mass, estimators, Kalman filters.

1. INTRODUCTION

The aircraft centre of gravity (cg) is the location about which an aircraft rotates (NASA, 2008), and it is an important piece of data because it affects aircraft stability. The cg will vary in-flight as fuel is burnt or sloshes in its fuel tanks, or when stores are deployed.

Previous work (Stanley and Goodall, 2009) described how state augmentation could be applied to a Kalman-Bucy filter, (Kalman and Bucy, 1961) to accurately estimate unexpected angular acceleration of an aircraft, where the unexpected angular acceleration is caused by a cg shift. It also showed how the estimate of unexpected angular acceleration could be used to obtain an estimate of a change in an aircraft cg. However, Stanley and Goodall (2009) only considered estimating the cg when the aircraft was trimmed in straight and level flight, and it only gave results from using a linear aircraft model. This paper will investigate the development of a cg estimator which is used with a non-linear aircraft model when the aircraft goes through a sequence of manoeuvres. The success of the estimator will be measured against a set of pre-determined performance requirements.

It was noted in Stanley and Goodall (2009) that the potential benefits of accurately estimating the aircraft cg are, in general, enhanced flexibility in the Flight Control System (FCS) design. In particular an accurate cg estimate provides :

a) an additional method to estimate the cg when other methods have failed.
b) an additional piece of data to the FCS which may be used to improve fault detection.
c) the potential for cg position dependent design scheduling, resulting in a less conservative FCS design.
d) the possibility of improving aircraft handling with cg changes.

2. CURRENT TECHNIQUES AND RESEARCH

The previous paper (Stanley and Goodall, 2009) described a number of existing techniques used to estimate the cg, both on the ground and in-flight.

When the aircraft is on the ground the aircraft can be weighed on scales (Federal Aviation Administration, 2007). The arms (distance between the cg of the item and a datum point) of the weighing points are specified for the aircraft. The moment is the weight multiplied by the arm. The cg position is calculated as:

$$cg = total\ moment\,/\,total\ weight \qquad (1)$$

When the aircraft is in flight a number of different approaches have been used to estimate the cg. The simplest method involves dividing the aircraft into nodes and specifying the weight and cg location for each node. As the aircraft flies fuel is consumed which reduces the weight in the fuel tank nodes and the cg is recalculated using (1). This approach is described by Brockman (1980) and then updated by Blakely and Hedges (1998) to make an allowance for fuel sloshing.

An alternative method for in-flight estimation is to use accelerometers at the front and rear of the aircraft, see Glover (1985). The accelerometer signals undergo band-pass and low-pass filtering and are combined to obtain a signal representing the cg.

A method patented by Boeing in Wu (1996) describes a cg estimation system used when a commercial aircraft is descending to land. This method uses many parameters to estimate the cg. An obvious limitation of Wu (1996) is that it is only useful when the aircraft is in a stable configuration in its landing phase.

More recent research has used neural networks to estimate aircraft weight and cg. The paper by Idan et al (2004) describes a neural net trained to estimate the weight and longitudinal cg for a small business jet in trimmed flight. The authors claim the neural net achieved an accuracy of 1.6% of mean aerodynamic chord (mac) and 99% of the points were within 1% mac.

Alternatively Bi et al (2004) describe using a neural net to estimate the weight and cg in a V-22 tilt-rotor aircraft. The authors claim the cg estimate had a root mean square (rms) error of 4.318mm (0.17 inches).

Research by Abraham and Costello (2009) has used an Extended Kalman Filter (EKF) to estimate the weight and cg of a helicopter. The EKF is constructed with the rigid state of the helicopter and augmented with the weight and the 3 components of the cg (x, y and z axis).

The research described here uses an enhanced estimator structure but preserves the use of the more simplistic Kalman-Bucy filter.

3. THE ADMIRE AIRCRAFT MODEL

In this paper the ADMIRE aircraft model is used to model an aircraft and the changes to its cg position. The ADMIRE (Aero-Data Model In a Research Environment) model is a generic model of a small single-seat fighter aircraft with a delta-canard configuration. It contains twelve states : velocity (V_T), angle of attack (α), sideslip (β), roll rate (p_b), pitch rate (q_b), roll angle (φ), pitch angle (θ), yaw angle (ψ), longitudinal coordinate (x_v), lateral coordinate (y_v), normal coordinate (z_v) plus additional states for sensors, actuators and FCS. The model is fully described by Forsell and Nilsson (2005).

The ADMIRE aircraft model can model a variety of parametric uncertainties, amongst them are changes in the longitudinal cg position x_{cg} and also aircraft mass. This facility has been used to inject cg changes into the ADMIRE aircraft model.

In ADMIRE the pitch acceleration equation is defined in (2).

$$\dot{q}_b = C_5 \cdot p_b \cdot r_b - C_6(p_b^2 - r_b^2) + C_7 \cdot M_y \qquad (2)$$

where

$$C_5 = \frac{I_z - I_x}{I_y}$$

$$C_6 = \frac{I_{xz}}{I_y}$$

$$C_7 = \frac{1}{I_y}$$

$\dot{q}_b$ is the pitch acceleration

$\dot{r}_b$ is the yaw acceleration

$\dot{p}_b$ is the roll acceleration

M_y is the pitching moment

I_x is the x body moment of inertia

I_y is the y body moment of inertia

I_z is the z body moment of inertia

I_{xz} is the x-y body axis product of inertia

The total pitching moment equation is defined in (3).

$$M_y = \overline{q} \cdot S_{ref} \cdot c_{ref} \cdot Cm_{tot} - x_{cg} \cdot F_z + z_{cg} \cdot F_x \\ - 0.15T_x + 5.5T_z \qquad (3)$$

where

$\overline{q}$ is the dynamic pressure

S_{ref} is the wing surface area

c_{ref} is the mean aerodynamic chord

Cm_{tot} is the pitching moment coefficient

F_z is the total force in body-fixed z axis

F_x is the total force in body-fixed x axis

x_{cg} is the centre of gravity along the x axis

z_{cg} is the centre of gravity along the z axis

T_x is the engine thrust along the x axis

T_z is the engine thrust along the z axis

It can be seen in (3) that the effect on the pitching moment of a cg shift along the x axis is dependent upon the normal force F_z.

$$\Delta M_y = \Delta cg \cdot F_z \qquad (4)$$

The change in the moment can be calculated from the change in the pitch acceleration as :

$$\Delta M_y = I_y \cdot \Delta \dot{q} \qquad (5)$$

Substituting (5) into (4) :

$$\Delta cg = \frac{I_y \cdot \Delta \dot{q}}{F_z} \qquad (6)$$

Note that although F_z is an output in the ADMIRE model n_z has been used, because ADMIRE provides a sensor model for n_z. The relationship between F_z and n_z is :

$$n_z = -F_z / (mass \cdot g) \qquad (7)$$

Therefore the equation used with the ADMIRE model to estimate changes in longitudinal cg is :

$$\Delta cg = \frac{I_y \cdot \Delta \dot{q}}{-n_z \cdot mass \cdot g} \qquad (8)$$

4. PERFORMANCE AND MANOEUVRE REQUIREMENTS

This research is sponsored by BAE Systems who supplied the performance and manoeuvre requirements for a cg estimator. The performance requirements specify an acceptable level of estimation error whereas the manoeuvre requirements specify the operational manoeuvres under which the estimator should work correctly.

Performance requirements

1.   Output updated at 10Hz
2.   Maximum static error 0.3% mean aerodynamic chord (mac)
3.   Maximum dynamic error 0.5% mac
4.   Static error accuracy within 1 second
5.   Transient exceedance of static error allowed as long as it does not exceed 20% of the static error level, the static performance is recovered within 1 second, no further exceedance of static performance requirement occur whilst the system remains undisturbed.

Manoeuvre requirements

1.   Maximum roll rate +/-30 deg/s
5.   Maximum lateral acceleration +/- 1.5m/s^2
6.   Maximum speed acceleration +/- 0.03 Mach/s
7.   Maximum pitch rate obtained from full pull up or push down command in 5 seconds

5. METHODOLOGY

It was previously described by Stanley and Goodall (2009) how a Kalman-Bucy filter can be augmented to estimate unexpected angular acceleration, and how this can be converted into an estimate of the change in cg.

In this paper the Kalman-Bucy filter contains the 10 aircraft states in x (9), which are used in the linear differential equations of the aircraft (10).

$$x = [V_t, \alpha, \beta, p, q, r, \phi, \theta, \varphi, z] \qquad (9)$$

$$\dot{x} = Ax + Bu \qquad (10)$$

The state x has been augmented with the unexpected pitch acceleration $\dot{q}_{unex}$ to give x̃ :

$$x̃ = [x, \dot{q}_{unex}] \qquad (11)$$

The state-space equation in (12) illustrates how $\dot{q}_{unex}$ is obtained. All states, apart from q, retain their original A and B matrix values as shown by the '$\cdots$' to denote no change.

$$x' = [V_t, \alpha, \beta, p, r, \phi, \theta, \varphi, z]$$

$$\begin{bmatrix} \dot{x}' \\ \dot{q} \\ \ddot{q}_{unex} \end{bmatrix} = \begin{bmatrix} \cdots & \cdots & 0 \\ \cdots & \cdots & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x' \\ q \\ \dot{q}_{unex} \end{bmatrix} + \begin{bmatrix} \cdots \\ \cdots \\ 0 \end{bmatrix} u \qquad (12)$$

This method of modelling the unknown acceleration is adopted from an example contained in Friedland (1986: 421), in which unknown acceleration is modelled as a random process. The new augmented state x̃ is fully observable.

The Kalman gain in the filter was obtained using the Matlab 'lqe' function on the augmented state x̃. The filter is used to obtain $\dot{q}_{unex}$ which is then scaled by the moment of inertia I_y and divided by nz and mass, as defined in (8) to obtain the change in cg.

One approach that can be used when estimating non-linear systems is to employ an Extended Kalman Filter (EKF). An EKF is essentially the same as a Kalman filter except that the filter is continuously re-linearised around the state estimate. The EKF has found use in applications such as navigation or GPS, however there are known problems with it regarding stability and divergence, see Bar-Shalom et al (2001: 385). Since there are known difficulties implementing an EKF, and given the restricted manoeuvre range requirements, an alternative structure to mitigate the effects of the non-linearities was adopted.

Fig. 1 shows the structure of the estimator which uses commands into the aircraft, and measurements from the aircraft to estimate aircraft state.
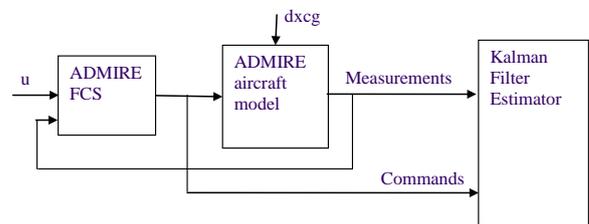

Fig. 1. Aircraft model and estimator.

Fig. 2 illustrates some of the non-linearities in the ADMIRE aircraft model and the changing effect of the angle of attack (AoA) on the pitching moment. The values were obtained

with the aircraft trimmed at different AoA. For example at 8° AoA the value of Cm_α (pitching moment coefficient derivative with respect to α) is approximately 8 and at 4° AoA the value of Cm_α is approximately 4.8. In other words a change in the AoA at 8° has nearly double the effect on the pitching moment compared to when the aircraft is at 4° AoA.
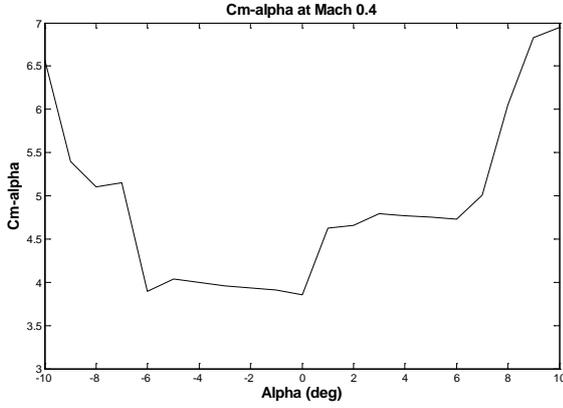

Fig. 2. CM_α at Mach 0.4 at various AoA.

Changes in speed also affect the pitching moment, and although there is a close relationship with the changing dynamic pressure $\overline{q}$ it was found that simple scaling of estimates for changes in $\overline{q}$ did not give sufficiently accurate results.

To cater for the aircraft non-linearities a set of derivative data was obtained at a number of points in the flight envelope. The data points are given in Table 1, so for example the derivative data was obtained for the linear aircraft model at Mach 0.5, at an AoA of 6° at 3000m altitude.

Table 1. ADMIRE data points

AoA (degrees)	[-7, -6, -1, 0, 1, 2, 3, 6, 9, 12]
Mach	[0.2 0.3 0.4 0.5 0.6]
Altitude (m)	[20 1000 3000 5000]

The data points in Table 1 were selected to reduce the effect of the non-linearities in the aircraft. For example at Mach 0.4 Cm_α is fairly constant at an AoA between 0° and -6°, and between 3° and 6°, see Fig. 2, so these data points were selected. However when nz approaches zero this magnifies any estimation error, see (8), therefore additional data points were selected at low AoA to reduce this error.

The derivative data is used to correct the pitching moment coefficients in the estimator, as detailed in Table 2.

Table 2. ADMIRE corrected coefficients

Commands	Cm_{dlc}, Cm_{drc}, Cm_{dloe}, Cm_{droe}, Cm_{dlie}, Cm_{drie}, Cm_{tss}
Measurements	Cm_α, Cm_{Vt}, Cm_q

The corrected coefficients are obtained by linear interpolation using Mach, AoA and altitude. Obviously any remaining non-linearities between data points will be missed with this method. The delta inputs into the estimator are similarly obtained from the difference with an interpolated value for the trim point.

Since the estimator has been configured when the aircraft is in level flight it will not accurately estimate the effect of 'p' and 'r' on the pitch acceleration, see (2). Therefore a correction is applied to $\dot{q}_{unex}$ to compensate for this :

$$pr\_correction = C_5 \cdot p_b \cdot r_b - C_6(p_b^2 - r_b^2) \qquad (13)$$

The overall structure of the estimator is shown diagrammatically in Figure 3. The output from the Kalman-Bucy filter is $\dot{q}_{unex}$ which is corrected for changes in 'p' and 'r', scaled by Iy and then divided by vertical acceleration nz, 'g' and the mass of the aircraft. The actual implementation includes a divide by zero check, and holds the value of dxcg when nz is between +0.4 g and -0.4 g.
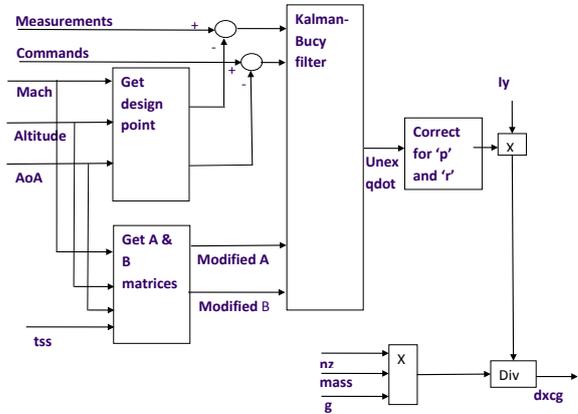
Fig. 3. CG estimator structure.

Note that the thrust command (tss) is shown as a separate input. A simple modification to Cm_{tss} was implemented to estimate the pitching effect when the afterburners were engaged, which occurs when tss is greater than 0.8.

Figure 4 shows engine thrust versus tss, in which engine thrust increases more sharply when tss is greater than 0.8. Since the estimator is generated from trim data points when the afterburner is inactive it will underestimate thrust at large values of tss. The correction for the active afterburners in the

estimator is a single value gain applied to the interpolated value of Cm$_{tss}$.
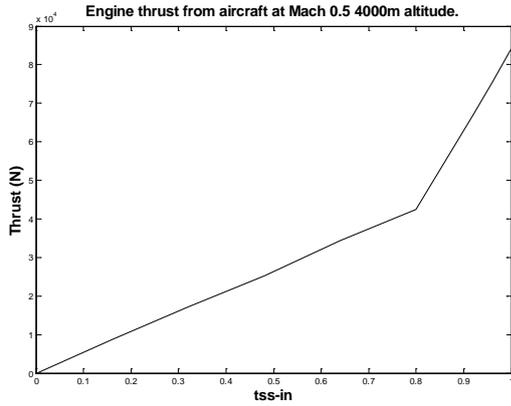

Fig. 4. Engine thrust output from tss command, aircraft at Mach 0.5 and 4000m altitude.

The measurement noise added to the ADMIRE aircraft model is defined in Table 3 – taken from typical sensor specifications. The noise variance is used to populate the R matrix of the Kalman-Bucy filter.

Table 3. ADMIRE noise data

	Measurement Noise (RMS)
Velocity (m/s)	0.034
α, β (rad)	0.0000698
p (rad/s)	0.00244
q, r (rad/s)	0.00087
φ (rad)	0.00034
θ, ψ (rad)	0.0000698
z (m)	1.0

The Q matrix in the Kalman-Bucy filter was obtained by trial and error from a sensible initial set of values (which were based upon the values of the average measurements) :

$$Q = diag([10\ 0.02\ 0.01\ 0.01\ 0.01\ 0.01\ 0.01\ 0.01\ 0.01\ 10\ 0.2]) \quad (14)$$

6. RESULTS

This section displays sample results for different aircraft manoeuvres. Each aircraft manoeuvre occurs when the aircraft is trimmed at Mach 0.4 and at 5000m altitude, and after 10 seconds there is a cg shift of 0.1m. Note that for clarity measurement noise has not been added to the signals making the results less noisy than they would otherwise be.

6.1 Acceleration

The aircraft accelerates from Mach 0.4 to Mach 0.75, see Figure 5. The top left graph shows the cg estimate, note that the cg estimate drifts off at speeds above Mach 0.6 because the estimator is only set up for a maximum speed of Mach 0.6. However the cg estimate remains within its static error limit. The other graphs show the change in nz, Mach and commanded thrust. Note that the engine afterburners are activated after about 15 seconds.
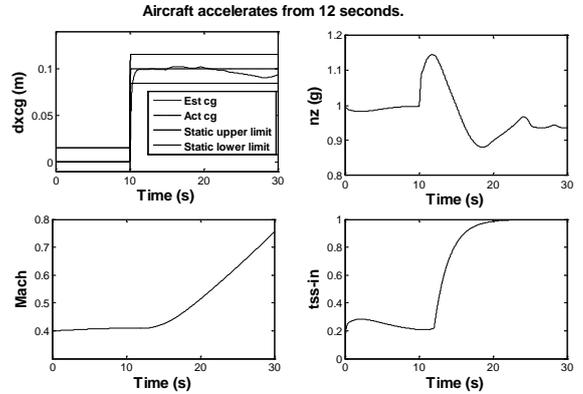
Fig. 5. Acceleration test.

6.2 Lateral Acceleration

A rudder command occurs after 11 seconds and generates a value for vdot (lateral acceleration) of approximately -1.4 m/s^2. The effect upon the estimate of the cg is negligible, see Figure 6, and the estimate remains well within limits.

Fig. 6. Lateral acceleration test.

6.3 Roll

A roll is commanded after 11 seconds and the aircraft banks at a rate of approximately 21 degrees/second and -29 degrees/second, see Figure 7. The aircraft bank angle reaches about 50 degrees. As before the cg estimate remains within specified limits.

Fig. 7. Roll test.

6.4 Pitch down

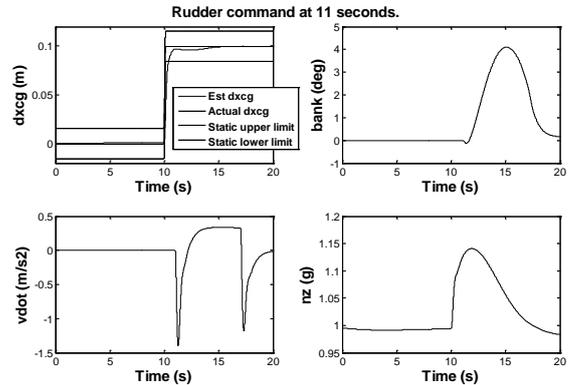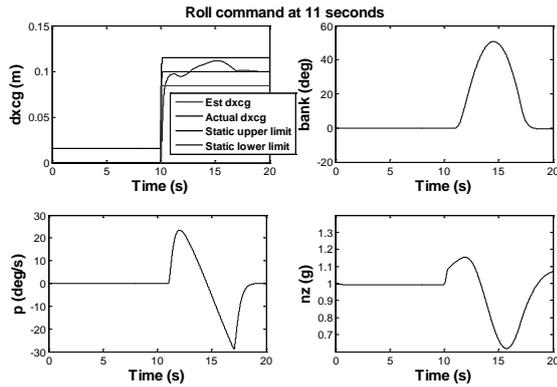A maximum push down is commanded at 15.5 seconds over 5 seconds, and then the pitch stick is pulled back to level flight over another 5 seconds. This test is more demanding of the estimator because changes in the AoA have a big effect upon the pitching moment. The AoA and nz both pass through zero. Since dxcg is calculated by dividing by nz (8) then as nz decreases this magnifies any errors. The cg estimate still stays within acceptable limits, see Figure 8.

During this test the aircraft speed increases to over Mach 0.6 and altitude drops from 5000m to about 3500m (not shown in the graph).
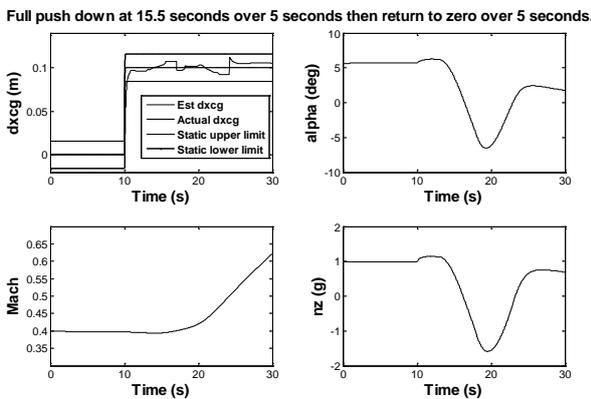

Fig. 8. Push down test.

The results from this section have shown that selective modification of the coefficients in the Kalman-Bucy filter have enabled the estimator to satisfy its requirements, even when the aircraft goes through various manoeuvres.

7. SAMPLE MISSION

In a sample mission the aircraft accelerates from Mach 0.2 at 20m altitude to about Mach 0.55 at 4500m altitude. It then levels off and performs a couple of banked turns changing direction by about 60°, and then descends to about 1000m. During this mission there are three step changes in the cg as shown in Table 4, which also shows the change in aircraft mass.

Table 4. Sample Mission.Change in cg and aircraft mass

Time (s)	Δxcg (m)	Δmass (Kg)
10	0.1	-400
120	-0.05	-210
180	-0.05	-210

There is a constant reduction in mass of 2kg per second to simulate fuel burn, taking no account of engine thrust. This continual reduction gives a total reduction in aircraft mass of 20% over the 500 second simulation. This unrealistic rate of reduction in mass has been done to minimise simulation time.

Figure 9 shows the cg estimate, z, mass and nz.
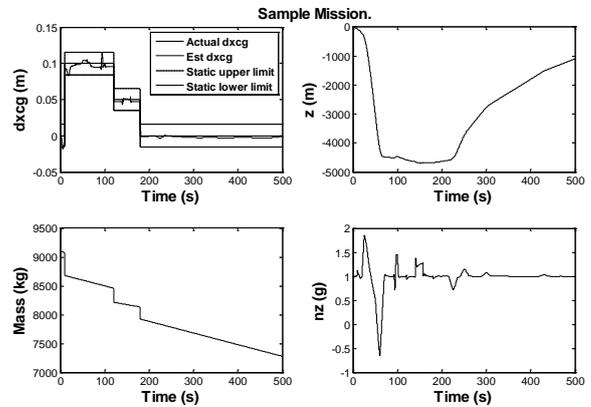
Fig. 9. Sample mission (dxcg, z, mass and nz).

Figure 10 shows Mach, AoA, yaw, bank, p and tss.
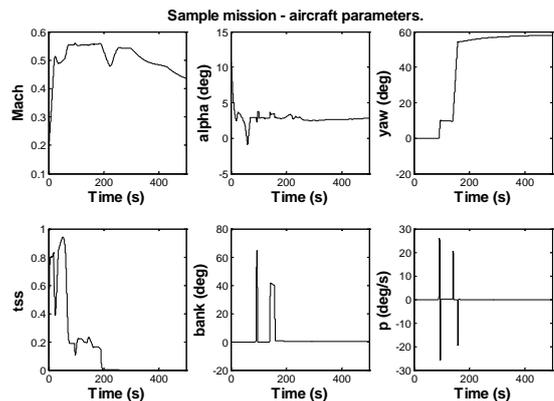
Fig. 10. Sample mission (Mach, alpha, yaw, tss, bank, p)

At all times the cg estimate stayed within the specified dynamic limits. It strayed outside the static limits on two occasions, between 5 and 7 seconds and between 17 and 20 seconds when the aircraft was accelerating. An improved engine model in the estimator should improve this because in the first deviation engine thrust was increasing sharply,

and in the second deviation the afterburners were becoming active when tss was around 0.8.

8. CONCLUSIONS

This paper has shown that a modified Kalman-Bucy filter can be used to accurately estimate the change in cg of an aircraft in manoeuvring flight. A set of performance requirements was specified and met, with the caveat that this was limited to part of the flight envelope and only limited testing has been performed.

The estimator should be extendable with a larger data set to cover more of the flight envelope. It is anticipated that different or additional modelling is required for transonic and supersonic speeds.

The estimator was found to be sensitive to changes in AoA, and therefore additional data at more AoA points should improve its performance.

It should be noted that an EKF would remove the need for the data points in Table 1, and should produce superior results since the filter is linearised on every iteration.

Future work will perform robustness tests by varying the moments of inertia, mass and pitching moment coefficients such as Cm_α. There will also be a practical validation of the estimator by using a hardware fuel rig as the source of changes in the cg.

ACKNOWLEDGEMENTS

REFERENCES

Abraham, M. and Costello, M. (2009) In-flight estimation of helicopter gross weight and mass center location, *Journal of Aircraft*, vol. 46, No. 3, pp1042-1049.

Bar-Shalom, Y. Rong-Li, W. and Kirubarajan, T. (2001) *Estimation with application to navigation and tracking*, Wiley.

Bi, N. Haas, D. and McCool, K. (2004) Investigation of in-flight gross weight and cg estimation, *60th Annual Forum Proceedings – American Helicopter Society*, pp2244-2254.

Blakely, P. and Hedges, D. (1998) A real-time application for computing gross weight and center of gravity measurements on Boeing experimental test flights, *Proc of International Symposium, Instrumentation in the Aerospace Industry*, pp211-218.

Brockman, D. (1980) A flight-test real-time CW-CG computing system, *26th International Instrumentation Symposium*, pp583-589.

Federal Aviation Administration (2007) *Aircraft Weight and Balance Handbook, FAA-H-8083-1 A*.

Forsell, L. and Nilsson, U. (2005) ADMIRE The aero-data model in a research environment version 4, *FOI-R-11624-SE*.

Friedland, B. (1986) *Control system design. An introduction to state-space methods*, McGraw-Hill.

Glover, J. (1985) *Patent 4545019* USA.

Idan, M. Iosilevskii, G. and Nazarov, S. (2004) In-flight weight and balance identification using neural networks, *Journal of Aircraft*, pp137-143.

Kalman, R. and Bucy, R. (1961) New results in linear filtering and prediction theory, *Trans. Of Basic Engineering*, vol 83D, pp95-108.

NASA (2008) Aircraft rotations, Retrieved June 6, 2008, from NASA Glenn Research Center:http://ww.grc.nasa.gov/WWW/K-12/airplane.html.

Stanley, A. and Goodall, R. (2009) Estimation of aircraft centre of gravity using Kalman filters, *Proc. of European Control Conference*, Aug 23-26 2009, pp 1848–1853.

Wu, T.H. (1996) *Patent 5571953* USA.

Appendix C – Phantom model robustness test script

This section of the appendix contains the Matlab script used to generate the cg estimator and Kalman-Bucy gain.

a) Phantom script

```
% Define the Phantom cg estimator state-space matrices
 estA = [0.0007181     0.00457     -29.072     -9.678      0;
     -0.0687     -0.2953     174.868     -1.601     0;
     0.00173     -0.0105     -0.4462     0.001277     1.04466146;
     0          0          1          0          0;
     0          0          0          0          0];

estB = [1.041;
     -6.294;
     -4.888;
     0;
     0];

estC = [1    0    0    0    0;
     0    1    0    0    0;
     0    0    1    0    0;
     0    0    0    1    0];

estD = [0;
     0;
     0;
     0];

G = eye(5);

% Initial Q matrix used to generate Kalman gain
Q = [400    0    0          0          0;
     0      10   0          0          0;
     0      0    0.00076    0          0;
     0      0    0          0.00076    0;
     0      0    0          0          1];
 % Final Q matrix used to generate Kalman gain
Q = [40    0    0          0          0;
     0     10   0          0          0;
     0     0    0.00008    0          0;
     0     0    0          0.00008    0;
     0     0    0          0          0.001];
% R matrix contains estimated noise variance
R = [0.04    0          0                0;
     0       0.0036     0                0;
     0       0          0.0000010966     0;
     0       0          0                0.0000010966];
```

```
% Matlab command 'lqe' is used to calculate Kalman gain contained within L
[L,P,E] = lqe(estA,G,estC,Q, R);
```

b) Matlab used in Phantom Robustness tests to generate revised state-space matrices

This code is based upon data contained in Cook on pages 81 and 88.

```
vel = 178;                    %velocity (m/s)
mass = 17642;                 %aircraft mass (kg)
air_d = 0.3809;               %air density (kg/m3)
S = 49.239;                   %wing area (m2)
mac = 4.889;                  %mean aerodynamic chord (m)
iy = 165669;                  %pitch moment of inertia (kg/m2)
g = 9.81;                     %acceleration due to gravity (m/s2)


mtick = mass/(air_d*vel*S/2);       %see p88 Cook 2007
Iytick = iy/(air_d*vel*S*mac/2);    %see p88 Cook 2007

% Values taken from p81 Cook 2007
Xu = 0.0076;
Xw = 0.0483;
Xwdot = 0;
Xq = 0;
Xn = 0.0618;
Zu = -0.7273;
Zw = -3.1245;
Zwdot = -0.3997;
Zq = -1.2109;
Zn = -0.3741;
Mu = 0.034;

Mw = -0.2169;
Mwdot = -0.591;
Mq = -1.2732;
Mn = -0.5581;


% Calculate values in M (p88 Cook 2007)
m1 = (Xwdot*mac/vel);
m2 = mtick - (Zwdot*mac/vel);
m3 = -Mwdot*mac/vel;

M = [mtick m1 0 0; 0 m2 0 0; 0 m3 Iytick 0; 0 0 0 1];

% Calculate values in steady symmetric flight (p88 Cook 2007)
% Body incidence is 9.4 degrees
We = vel*sind(9.4);
Ue = vel*cosd(9.4);

% Calculate derivatives of Atick
a1 = Xq*mac - mtick*We;
```

```
a2 = -mtick*g*cosd(9.4);
a3 = Zq*mac + mtick*Ue;
a4 = -mtick*g*sind(9.4);
a5 = Mq*mac;

Atick = [Xu      Xw     a1     a2;
         Zu     Zw     a3     a4;
         Mu     Mw     a5     0;
          0      0      1     0];

Btick = [vel*Xn; vel*Zn; vel*Mn; 0];

% Obtain A and B state-space matrices
A = inv(M)*Atick;
B = inv(M)*Btick;
```
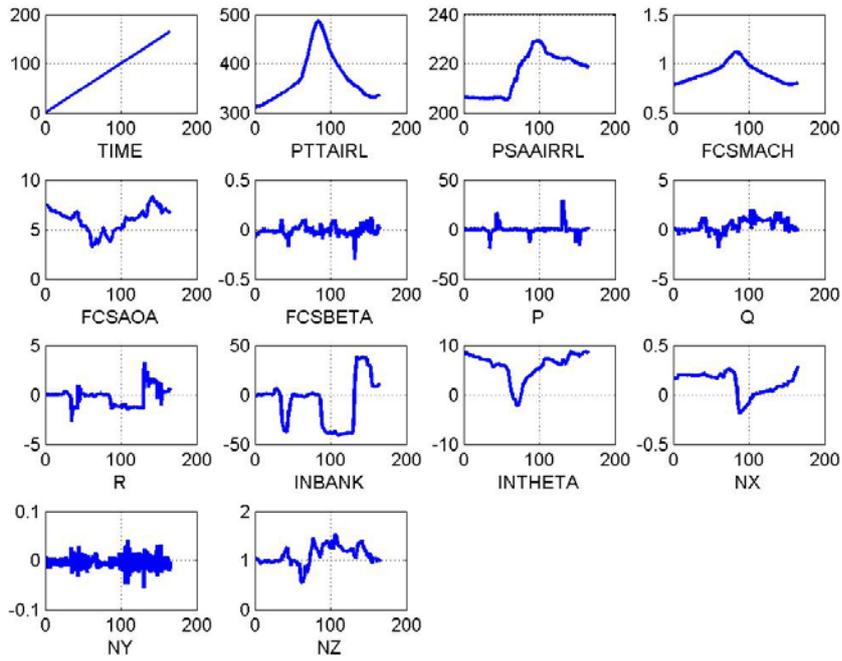
Appendix D – Flight test measurement noise data

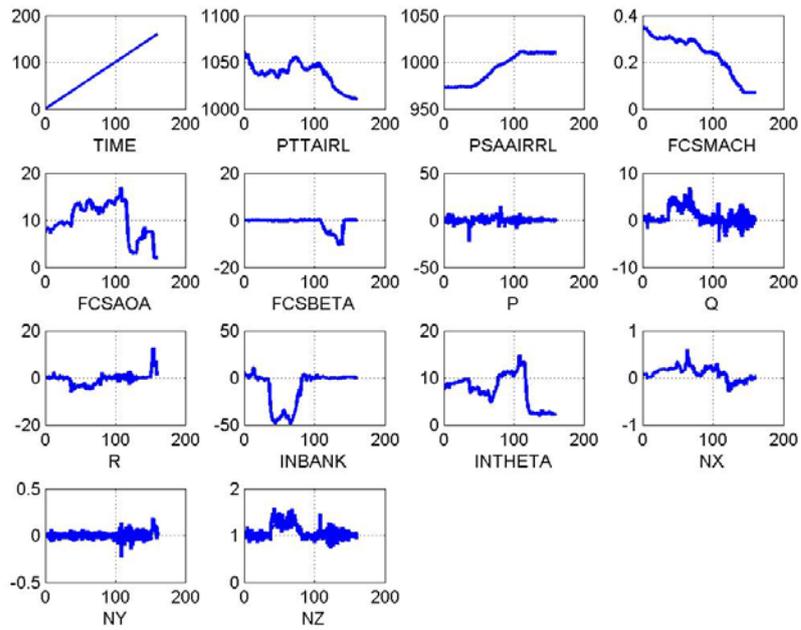**Figure 87: Flight test data used for measurement noise (2)**

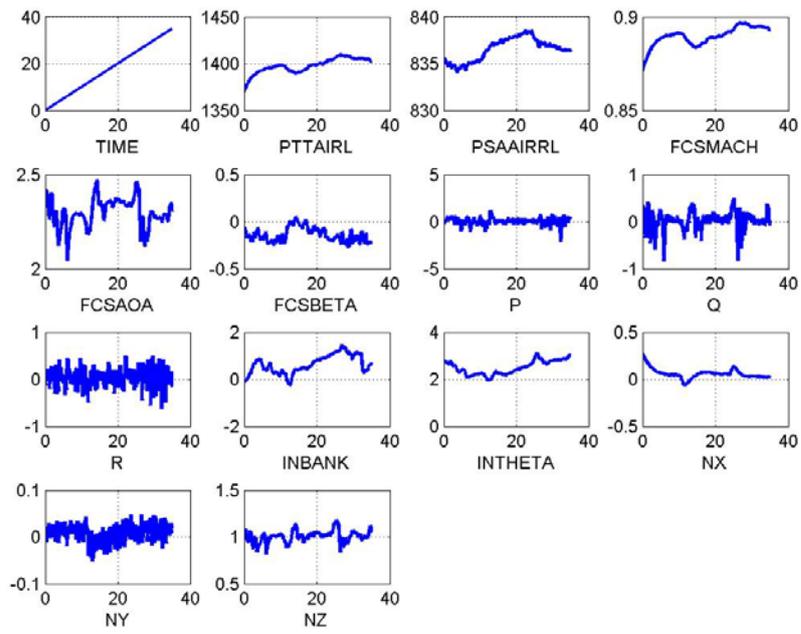**Figure 88: Flight test data used for measurement noise (3)**

Figure 89: Flight test data used for measurement noise (4)

Appendix E - Manoeuvre test results using uncorrected cg estimator

This appendix provides the manoeuvre test results for the unmodified longitudinal cg estimator defined in section 5.

Figure 90 to Figure 92 show the estimator results when the aircraft goes through different manoeuvres. In all cases the estimator is configured at the same speed and altitude as the aircraft, which is Mach 0.4 5000m altitude.

In all of the figures the cg estimate is in the top left graph and the acceptable static upper and lower limits are marked on the graph. The other graphs show the aircraft commands or relevant aircraft data pertaining to the manoeuvre.

The BAE pitching manoeuvre requirement for a push down (see section 3.1) is illustrated in Figure 90. In this manoeuvre the pitch stick is pushed down to its maximum over a five second period, and then returned to its neutral position after a further five seconds. The top left graph in Figure 90 shows that the dxcg estimate quickly becoming unacceptable. The other graphs show the pitch stick command, the increase in aircraft speed to over Mach 0.6 and the change in the AoA as the aircraft pitches down and then back up.
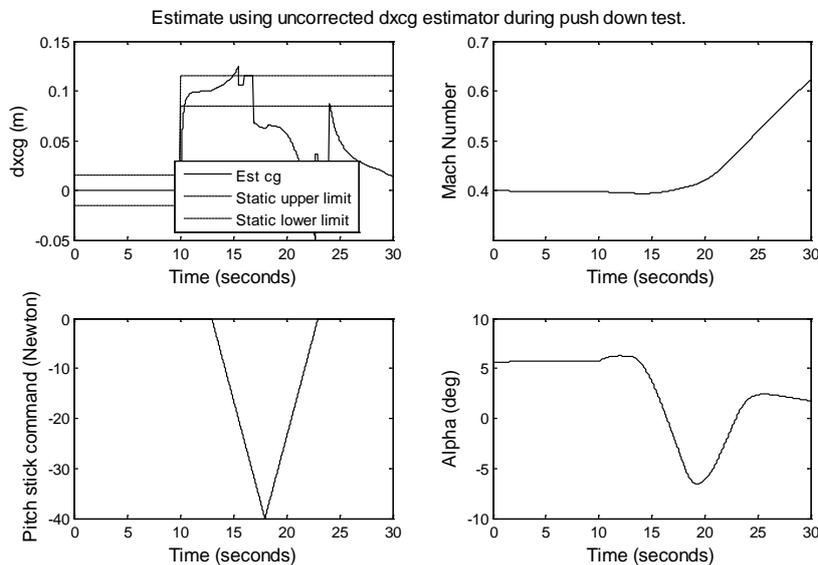

Figure 90: Uncorrected dxcg estimator. Push down test.

The BAE requirement for a roll manoeuvre is illustrated in Figure 91. BAE verbally advised that a maximum bank angle of 45° should be considered and in this test the bank angle approached 50°. There

was little change in aircraft speed during the manoeuvre. The maximum roll rate (not shown in Figure 91) was 29° per second, just under the maximum roll rate of 30° per second specified in the requirements in section 3.1. Figure 91 shows that the estimator exceeded its static upper limit for about 3 seconds during the manoeuvre.
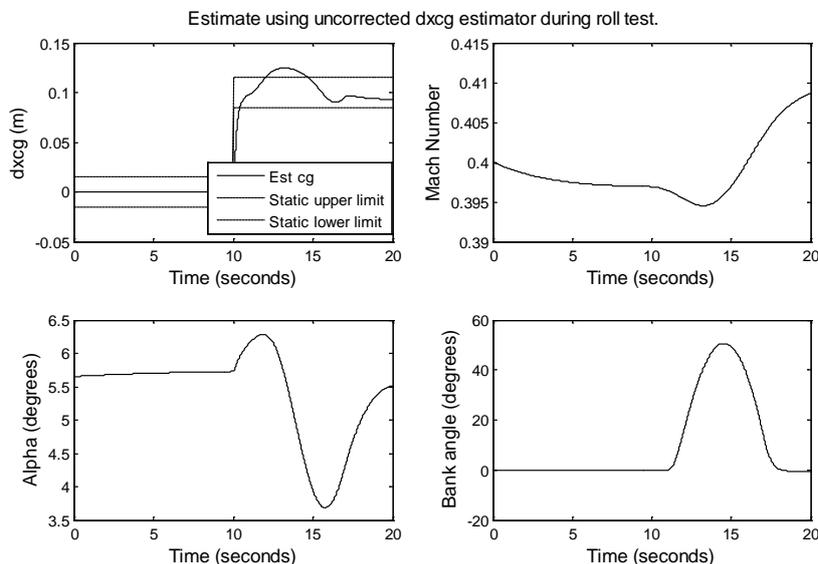
Figure 91: Uncorrected dxcg estimator. Roll test.

A lateral acceleration manoeuvre is shown in Figure 92. The requirement was to handle a maximum lateral acceleration of +/- 1.5 m/s^2, the bottom left graph shows lateral acceleration of about -1.4 m/s^2. During this manoeuvre there is little change to aircraft speed or angle of attack and the cg estimate remains well within limits.
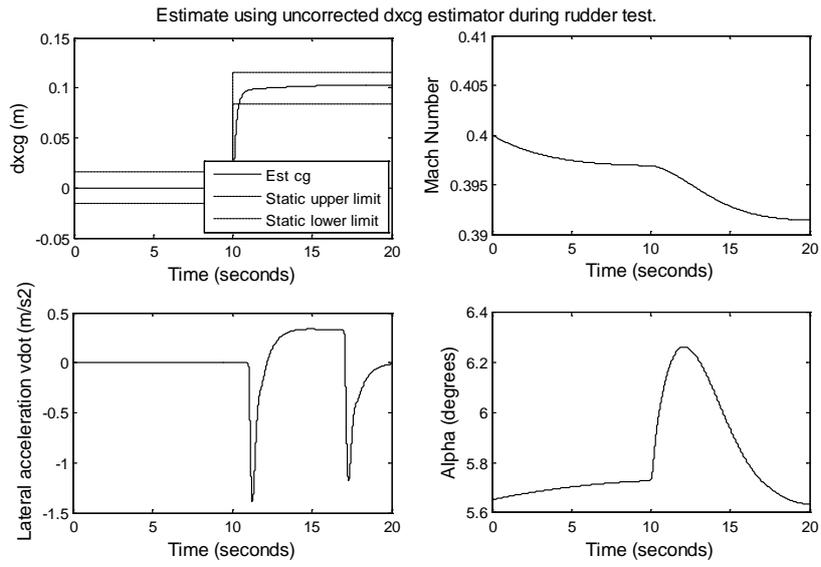
Figure 92: Uncorrected dxcg estimator. Rudder test.

Appendix F - ADMIRE trim and linearization script to generate cg estimator

The ADMIRE model used in this research was ADMIRE release 4.1.

This release contained two scripts to trim and linearise the bare aircraft and the flight control system:

Admtrim_sl.m – used to trim and linearise the aircraft in straight and level flight, takes Mach number and altitude as inputs.

Admtrim_aoa – used to trim and linearise the aircraft at a specified angle of attack, takes Mach number, altitude and angle of attack as inputs.

These trim routines calculate the initial conditions for the aircraft inputs, aircraft state and also the rate of deceleration e.g.

```
EDU>> admtrim_sl

Mach number [-] : 0.4
Altitude    [m] : 5000

Trimming ADMIRE complete trim.......... DONE!
u0new(1)  = -0.00561 deg      ( Canard deflections , + = trailing edge down )
u0new(3)  =  1.04693 deg      ( Elevon deflections , + = trailing edge down )
u0new(7)  =  0.00000 deg      ( Rudder deflection  , + = trailing edge left )
u0new(8)  =  0.00000 deg      ( Leading edge defl. , + = leading edge down  )
u0new(10) =  0.14090          ( Throttle setting                           )

After trimming for AoA:
x0(1)  = 128.21176     ( Vt    [m/s]   )
x0(2)  =  5.65143 ( AoA   [deg]   )
x0(3)  =  0.00000 ( Beta  [deg]   )
x0(4)  =  0.00000 ( P     [deg/s] )
x0(5)  =  0.00000 ( Q     [deg/s] )
x0(6)  =  0.00000 ( R     [deg/s] )
x0(7)  =  0.00000 ( Psi   [deg]   )
x0(8)  =  5.65143 ( Theta [deg]   )
x0(9)  =  0.00000 ( Phi   [deg]   )
x0(10) =  0.00000 ( X     [m]     )
x0(11) =  0.00000 ( Y     [m]     )
x0(12) = -5000.00000   ( Z     [m]     )

Decelerating:   -0.2885 m/s^2
Run simulations with 'admire_sim.mdl' and view the results with command 'trimplot'
```

After trimming the aircraft a linear model is generated using a cut down Simulink model to generate a 12 state linear aircraft model. The 12 state linear aircraft model is then modified to remove the redundant 'x' and 'y' states.

```
% Linearise aircraft to generate 12 state linear model
[Areduced,Breduced,Creduced,Dreduced] =
linmod_nnt('admire_bare_reduced',x0(10:21),zeros(16,1));

% 11 state dxcg estimator (no X or Y)
estA = Areduced;
estA(10,:) = [];  % delete row 10 X
estA(10,:) = [];  % delete row 10 Y
estA(:,10) = [];  % delete column 10 X
estA(:,10) = [];  % delete column 10 Y
for i=1:11,
    estA(i,11)=0; % Add 11th row for unexpected q-dot
end
estA(5,11) = 1;
estA(11,1:11) = 0;


estB(1:12,1:10)=Breduced(1:12,1:10);
estB(13,1:10) = 0;
estB(10,:) = [];  % remove X
estB(10,:) = [];  % remove Y


estC = Creduced;
for i=1:12,
    estC(i,13)=0;
end
estC(10,:) = []; % remove X row
estC(10,:) = []; % remove Y row
estC(:,10) = []; % remove X column
estC(:,10) = []; % remove Y column


for i=1:12,
    estD(i,1:10)=Dreduced(i,1:10);
end
estD(10,:) = []; % remove X
estD(10,:) = []; % remove Y


estG = eye(11);


estQ = [
10 0 0 0 0 0 0 0 0 0 0;        %Vt
0 0.02 0 0 0 0 0 0 0 0 0;      %alpha
0 0 0.01 0 0 0 0 0 0 0 0;      %beta
0 0 0 0.01 0 0 0 0 0 0 0;      %p
0 0 0 0 0.01 0 0 0 0 0 0;      %q
0 0 0 0 0 0.01 0 0 0 0 0;      %r
0 0 0 0 0 0 0.01 0 0 0 0;      %phi (bank)
0 0 0 0 0 0 0 0.01 0 0 0;      %theta (pitch)
0 0 0 0 0 0 0 0 0.01 0 0;      %psi (yaw)
0 0 0 0 0 0 0 0 0 10 0;        %z
0 0 0 0 0 0 0 0 0 0 0.2];      %unexpected qdot


estR = [
0.001156 0 0 0 0 0 0 0 0 0;         %Vt
0 0.00000000487 0 0 0 0 0 0 0 0;    %alpha
0 0 0.00000000487 0 0 0 0 0 0 0;    %beta
0 0 0 0.00000595 0 0 0 0 0 0;       %p
```

```
0 0 0 0 0.00000076 0 0 0 0 0;        %q
0 0 0 0 0 0.00000076 0 0 0 0;        %r
0 0 0 0 0 0 0.00000019 0 0 0;        %phi (bank)
0 0 0 0 0 0 0 0.00000000487 0 0;     %theta (pitch)
0 0 0 0 0 0 0 0 0.00000000487 0;     %psi (yaw)
0 0 0 0 0 0 0 0 0 1];                %z

% Calculate Kalman gain
[L,P,E] = lqe (estA,estG,estC,estQ,estR)

% Generate Kalman-Bucy filter to correct q-trim
estAtrim = [0 1;
            0 0];
estBtrim = [0; 0];
estCtrim = [1 0];
estDtrim = [0];


estGtrim = eye(2);


estQtrim = [1 0;
            0 10];


estRtrim = [0.00000076];             %q

[Ltrim,Ptrim,Etrim] = lqe (estAtrim,estGtrim,estCtrim,estQtrim,estRtrim)
```

The following Matlab script was used to generate the Kalman-Bucy filter for the lateral cg estimator.

```
% 11 state dxcg dyestimator (no X or Y)
dyestA = Areduced;
dyestA(10,:) = [];       % delete row 10 X
dyestA(10,:) = [];       % delete row 10 Y
dyestA(:,10) = [];       % delete column 10 X
dyestA(:,10) = [];       % delete column 10 Y
for i=1:11,
    dyestA(i,11)=0;      % Add 11th row for unex q-dot
end
dyestA(4,11) = 1;
dyestA(11,1:11) = 0;

dyestB(1:12,1:10)=Breduced(1:12,1:10);
dyestB(13,1:10) = 0;
dyestB(10,:) = [];       % remove X
dyestB(10,:) = [];       % remove Y

dyestC = Creduced;
for i=1:12,
    dyestC(i,13)=0;
end
dyestC(10,:) = [];       % remove X row
dyestC(10,:) = [];       % remove Y row
dyestC(:,10) = [];       % remove X column
dyestC(:,10) = [];       % remove Y column


for i=1:12,
    dyestD(i,1:10)=Dreduced(i,1:10);
```

```
end
dyestD(10,:) = [];        % remove X
dyestD(10,:) = [];        % remove Y


dyestG = eye(11);


dyestQ = [
10 0 0 0 0 0 0 0 0 0 0;        %Vt
0 0.02 0 0 0 0 0 0 0 0 0;       %alpha
0 0 0.02 0 0 0 0 0 0 0 0;       %beta
0 0 0 0.02 0 0 0 0 0 0 0;       %p
0 0 0 0 0.01 0 0 0 0 0 0;       %q
0 0 0 0 0 0.1 0 0 0 0 0;        %r
0 0 0 0 0 0 0.01 0 0 0 0;       %phi (bank)
0 0 0 0 0 0 0 0.01 0 0 0;       %theta (pitch)
0 0 0 0 0 0 0 0 0.01 0 0;       %psi (yaw)
0 0 0 0 0 0 0 0 0 10 0;         %z
0 0 0 0 0 0 0 0 0 0 0.1];       %unexpected pdot


dyestR = [
0.001156 0 0 0 0 0 0 0 0 0;            %Vt
0 0.00000000487 0 0 0 0 0 0 0 0;       %alpha
0 0 0.00000000487 0 0 0 0 0 0 0;       %beta
0 0 0 0.00000595 0 0 0 0 0 0;          %p
0 0 0 0 0.00000076 0 0 0 0 0;          %q
0 0 0 0 0 0.00000076 0 0 0 0;          %r
0 0 0 0 0 0 0.00000019 0 0 0;          %phi (bank)
0 0 0 0 0 0 0 0.00000000487 0 0;       %theta (pitch)
0 0 0 0 0 0 0 0 0.00000000487 0;       %psi (yaw)
0 0 0 0 0 0 0 0 0 1];                  %z


[dyL,dyP,dyE] = lqe (dyestA,dyestG,dyestC,dyestQ,dyestR)
```

Appendix G – Robustness test transient results

This appendix contains a set of graphs showing the transient behavior of the estimator when the non-linear aircraft model is changed to be different to the cg estimator configuration.

The same tests were performed on both the longitudinal and lateral cg estimator described in section 6. In all tests the aircraft was first trimmed for straight and level flight at Mach 0.4 and 5000m altitude. In straight and level flight the only graphs displayed are those for the change in mass because there was no visible effect when any moment of inertia was altered. The tests were also performed when the aircraft performed a push down and a roll manoeuvre, as described in section 6.4.1.

Table 26: ADMIRE robustness test. Changed mass and inertia values.

Aircraft parameter	% change
Ix	20
Iy	5
Iz	5
Mass	5

a) Longitudinal cg tests

Estimate in straight and level flight of dxcg with aircraft mass increased by 5%.

Figure 93: Robustness test, straight-level, dxcg, mass increased by 5%

Estimate of dxcg with Ix increased by 20%, and push down manoeuvre performed.

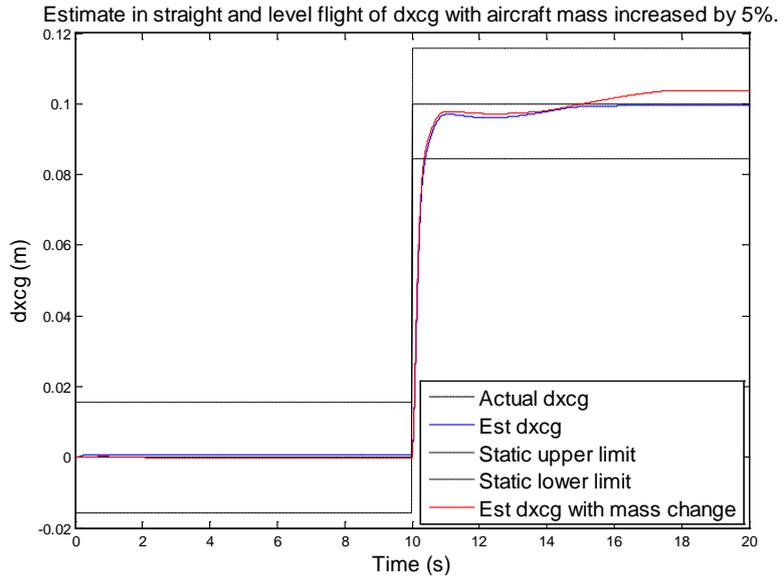**Figure 94: Robustness test, push down, dxcg, Ix increased by 20%**

Figure 95: Robustness test, push down, dxcg, Iy increased by 5%

Figure 96: Robustness test, push down, dxcg, Iz increased by 5%

Figure 97: Robustness test, push down, dxcg, mass increased by 5%

Figure 98: Robustness test, roll, dxcg, Ix increased by 20%

Figure 99: Robustness test, roll, dxcg, Iy increased by 5%

Figure 100: Robustness test, roll, dxcg, Iz increased by 5%

Figure 101: Robustness test, roll, dxcg, mass increased by 5%

b) lateral cg tests

Figure 102: Robustness test, straight-level, dycg, mass increased by 5%

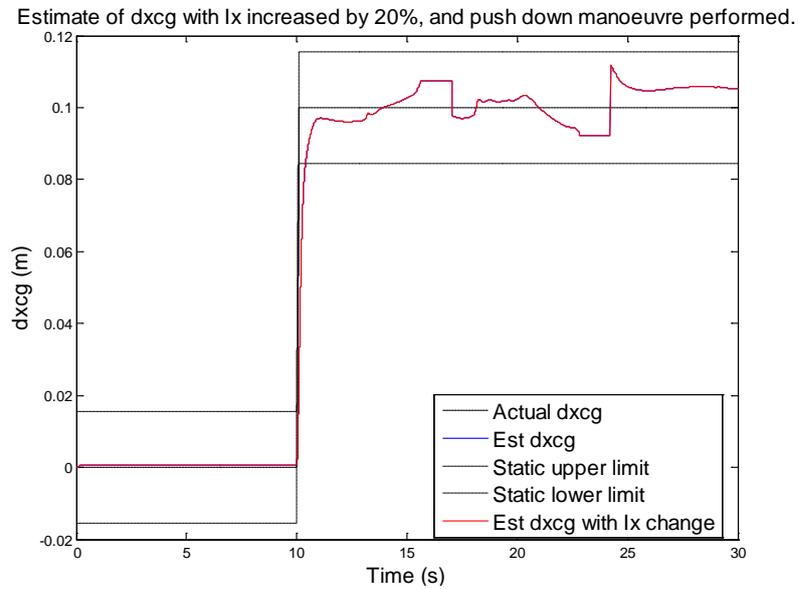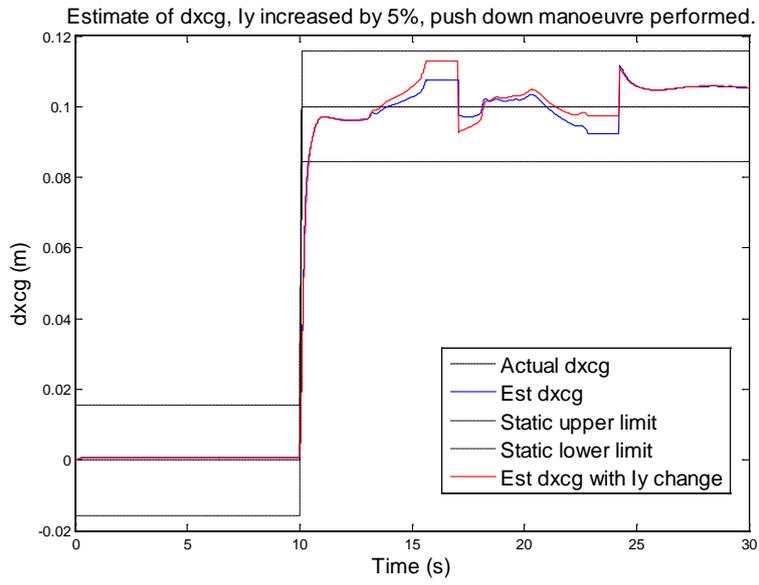**Figure 103: Robustness test, push down, dycg, Ix increased by 20%**

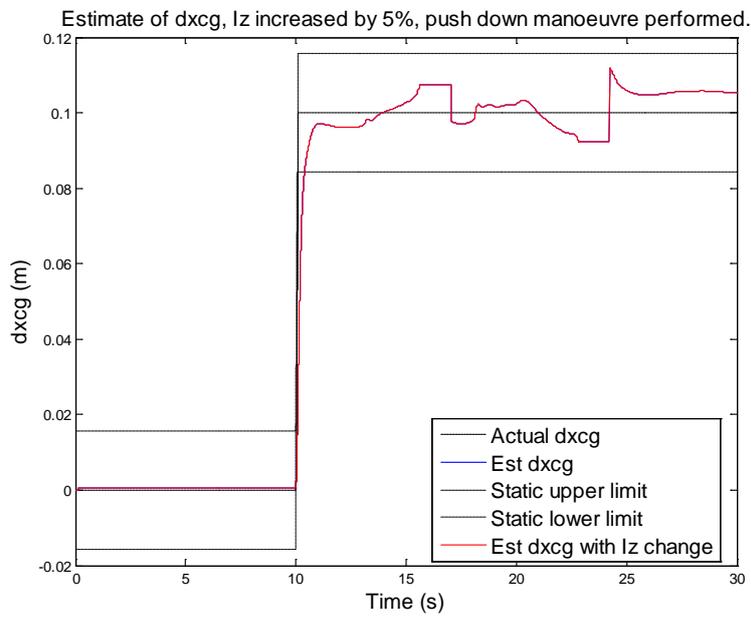**Figure 104: Robustness test, push down, dycg, Iy increased by 5%**

Estimate of dycg, Iz increased by 5%, push down manoeuvre performed.

Figure 105: Robustness test, push down, dycg, Iz increased by 5%

Estimate of dycg, mass increased by 5%, push down manoeuvre performed.
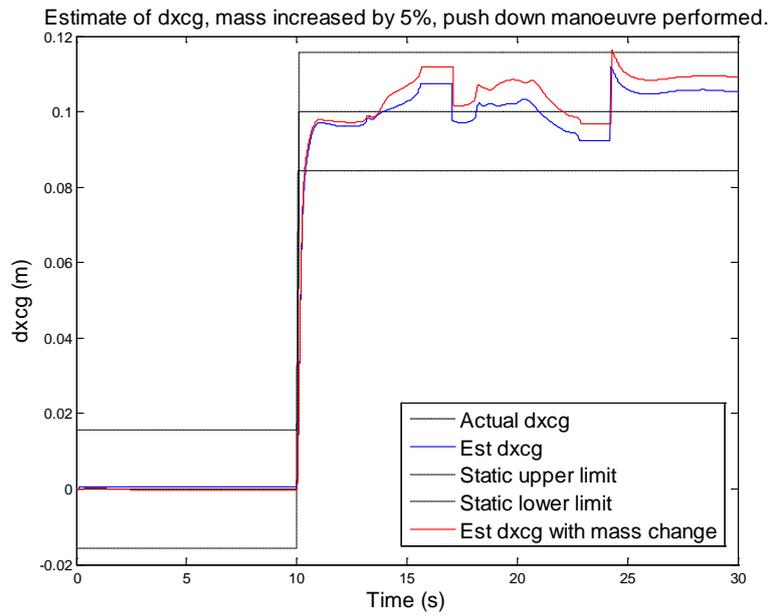

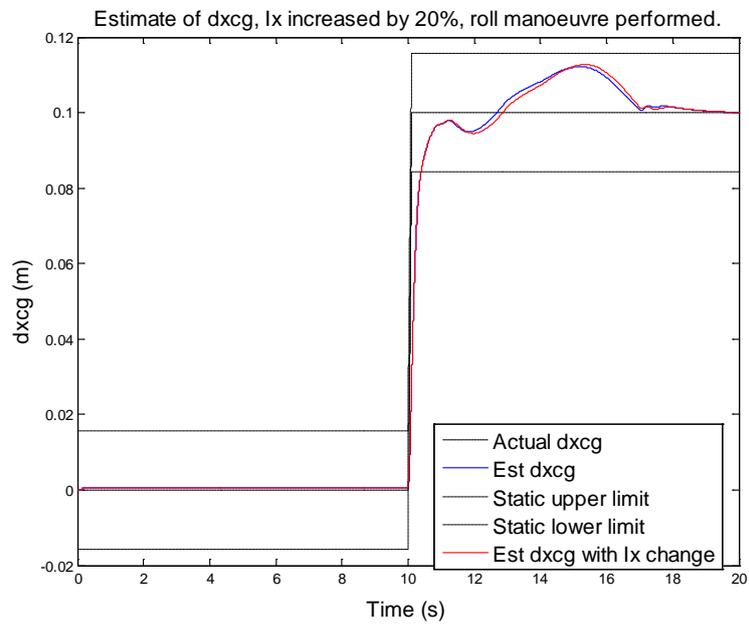**Figure 106: Robustness test, push down, dycg, mass increased by 5%**

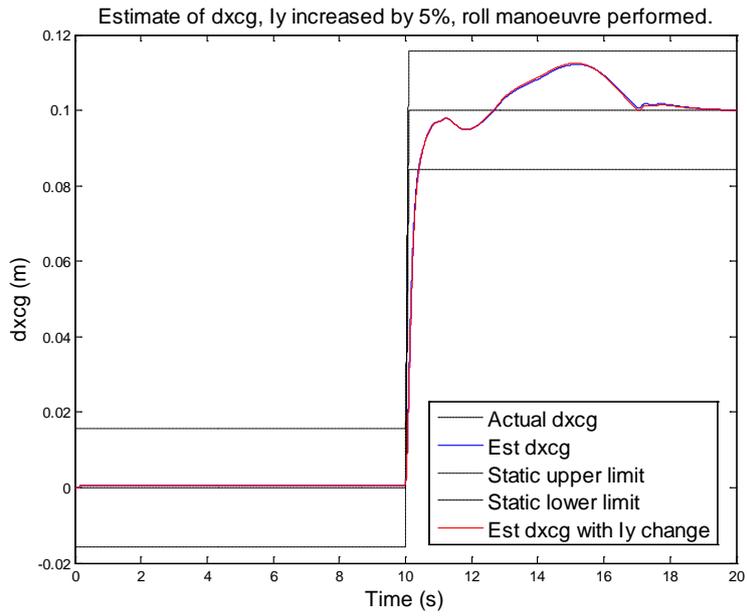**Figure 107: Robustness test, roll, dycg, Ix increased by 20%**

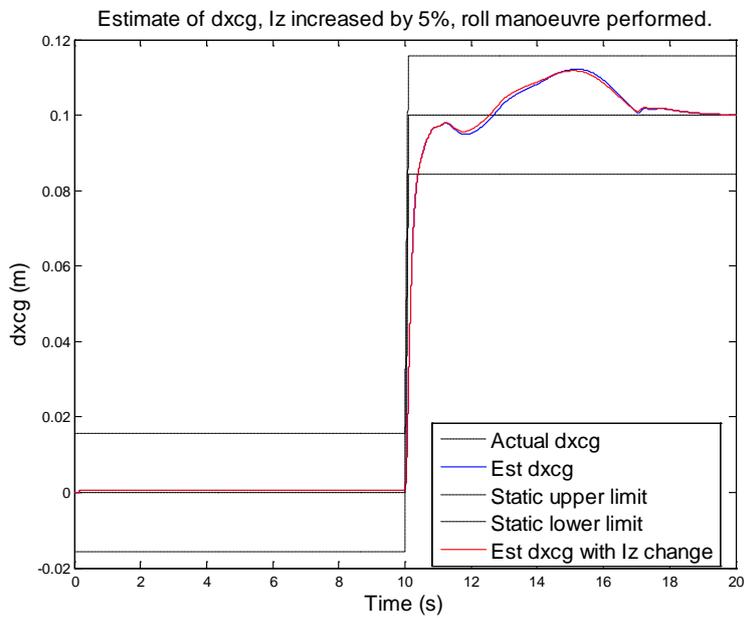**Figure 108: Robustness test, roll, dycg, Iy increased by 5%**

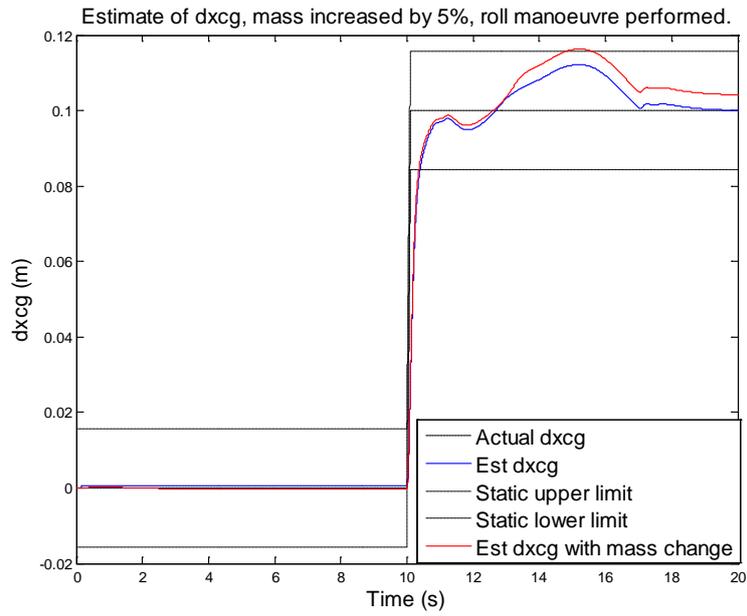**Figure 109: Robustness test, roll, dycg, Iz increased by 5%**

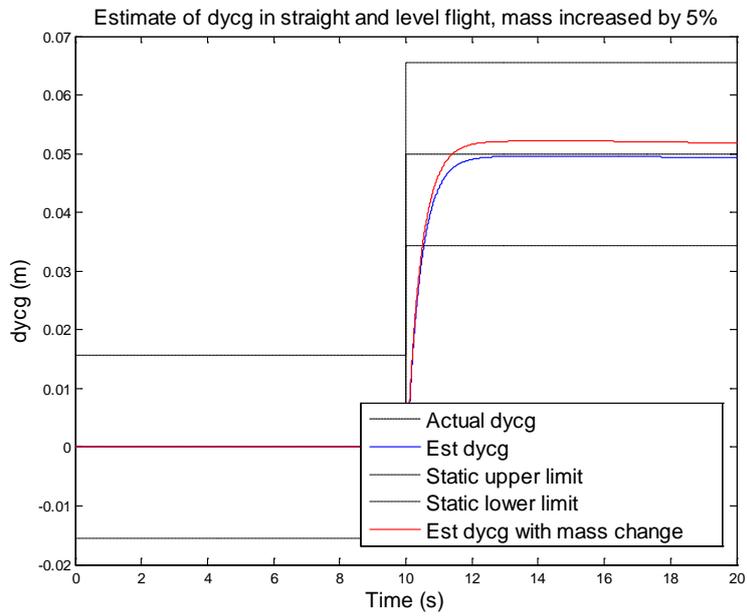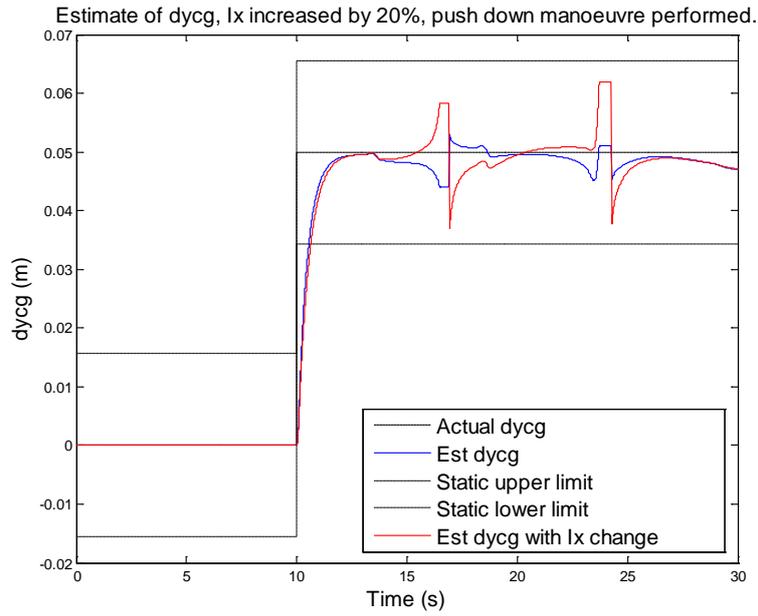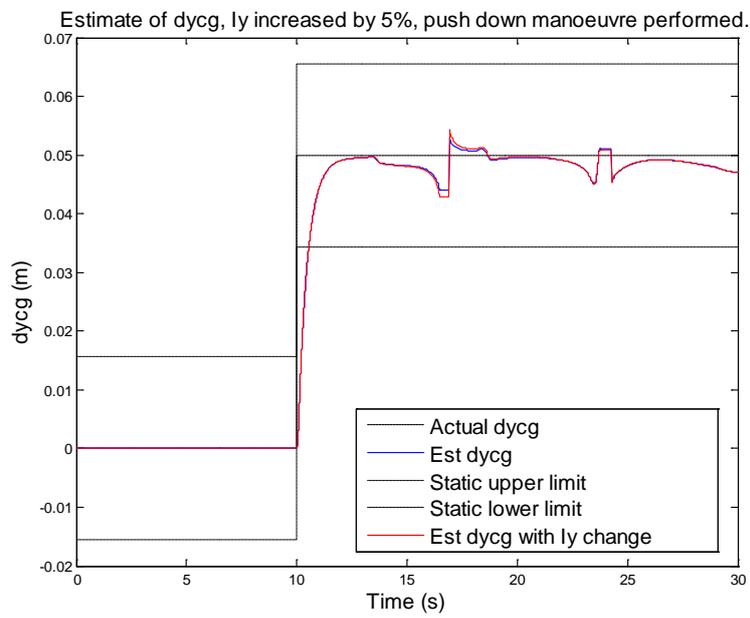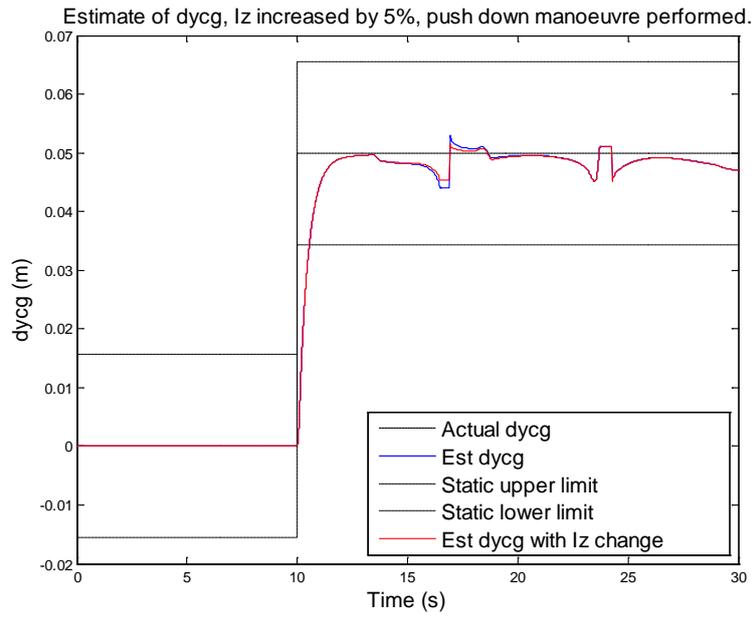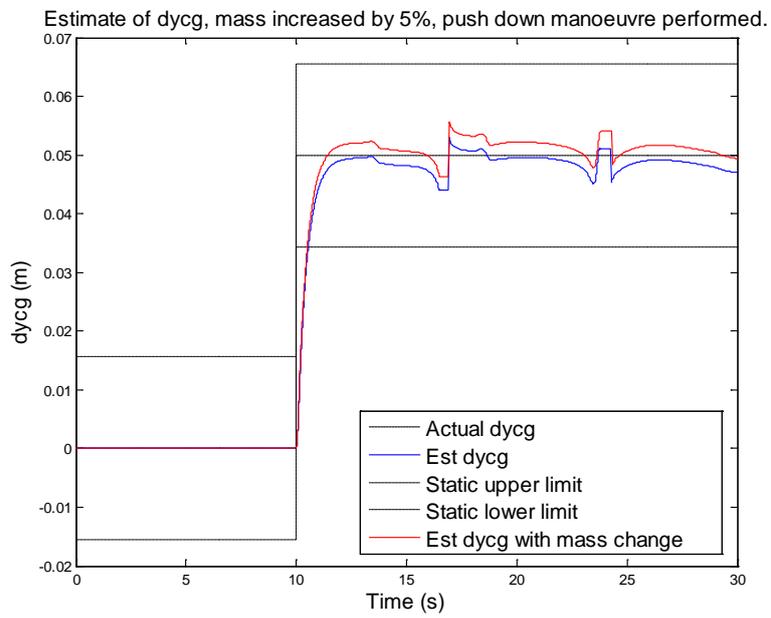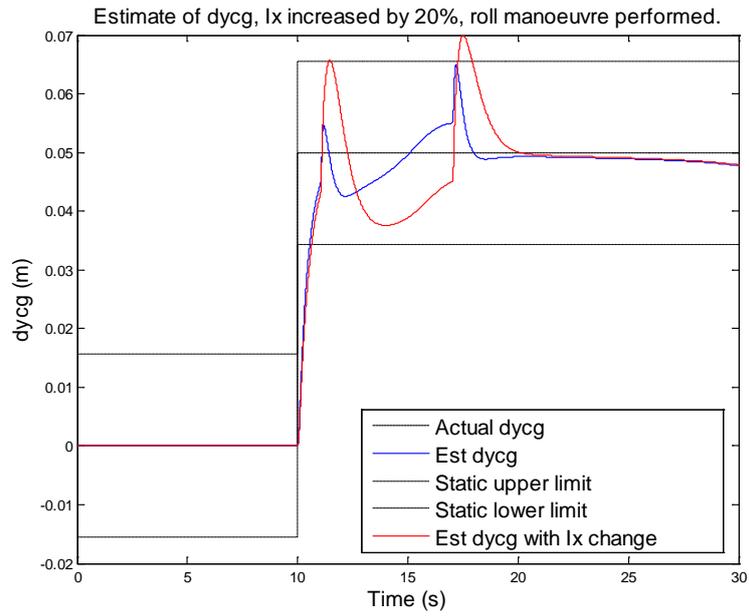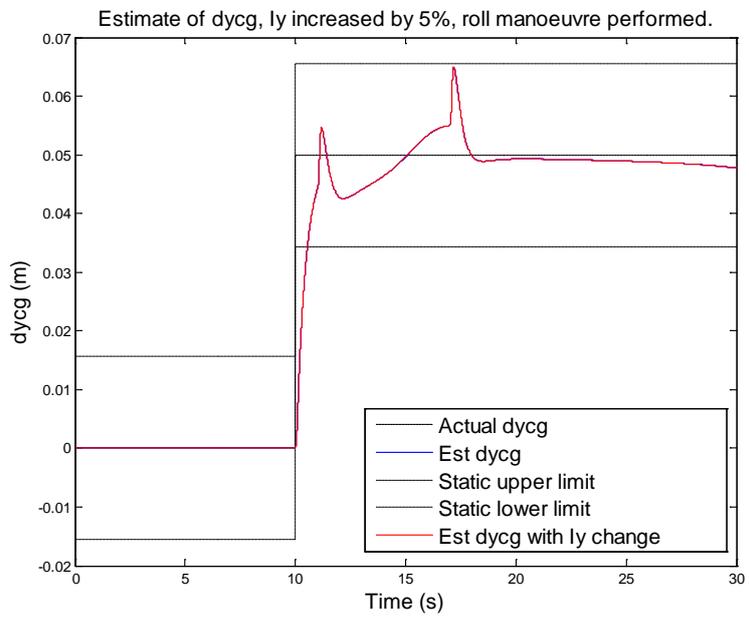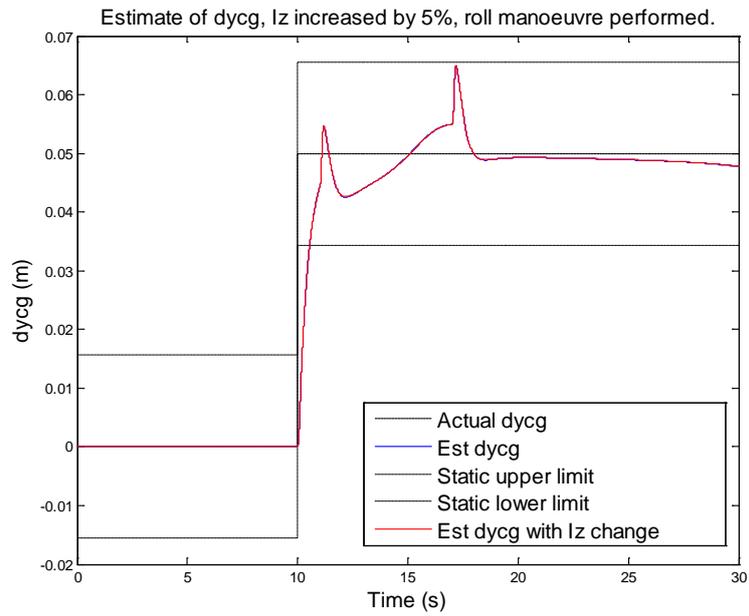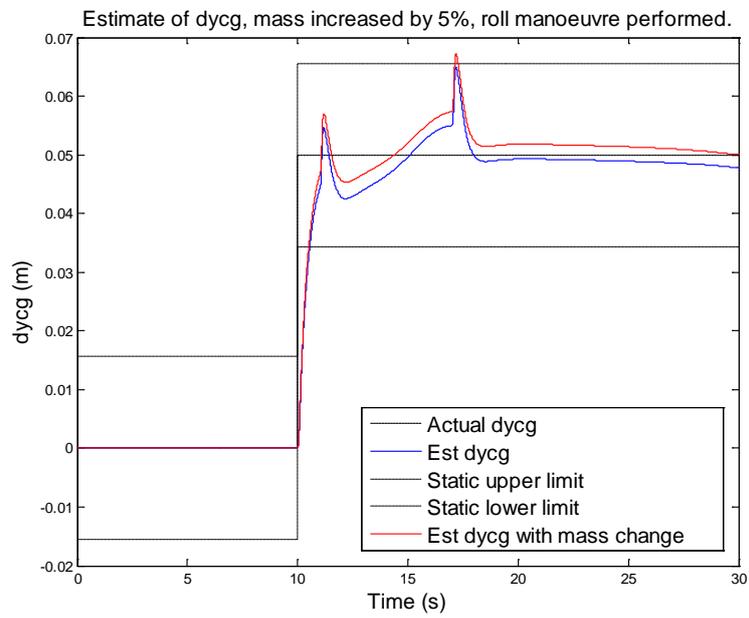**Figure 110: Robustness test, roll, dycg, mass increased by 5%**

Appendix H – ADMIRE lookup table data

The cg estimator described in sections 6 and 7 contained lookup table data to amend the coefficient derivatives depending upon Mach, altitude and angle of attack. This appendix contains examples of this data. Note that for conciseness this appendix only contains a subset of the full dataset.

The table data contains data obtained at the data points defined in Table 13. It consists of trim data which is used to obtain the delta values input into the estimator, and also the coefficient data which is used to alter the coefficients inside the Kalman-Bucy filter. The data is structured such that each block of data is for a specific Mach number, and is indexed by row for altitude and column for angle of attack. Each block of data is then concatenated together using Matlab 'cat' to generate the complete table for a specific coefficient or trim data item.

The following data is defined:

Trim data [V_T, α, q, canard (c), outer elevon (oe), inner elevon (ie), thrust (tss)]

Measurement data [Cm_{VT}, Cm_α, Cm_q, Cm_c, Cm_{oe}, Cm_{ie}, Cm_{tss}, Cm_{ldg}]

For the lateral cg estimator different tables were used for the coefficients.

Measurement data [Cl_β, Cl_p, Cl_r, Cl_φ, Cl_c, Cl_{oe}, Cl_{ie}]

1. Trim (velocity measurement)

```
vtY0bareM02 = [68.05879761  68.05879761 68.05879761 68.05879761 68.05879761
68.05879761 68.05879761 68.05879761 68.05879761 68.05879761;
67.2867943   67.2867943   67.2867943   67.2867943   67.2867943   67.2867943   67.2867943
67.2867943   67.2867943   67.2867943;
65.71558565 65.71558565 65.71558565 65.71558565 65.71558565 65.71558565 65.71558565
65.71558565 65.71558565 65.71558565;
64.10587889 64.10587889 64.10587889 64.10587889 64.10587889 64.10587889 64.10587889
64.10587889 64.10587889 64.10587889];

vtY0bareM06 = [204.1763928  204.1763928 204.1763928 204.1763928 204.1763928
204.1763928 204.1763928 204.1763928 204.1763928 204.1763928;
201.8603829 201.8603829 201.8603829 201.8603829 201.8603829 201.8603829 201.8603829
201.8603829 201.8603829 201.8603829;
197.146757   197.146757   197.146757   197.146757   197.146757   197.146757   197.146757
197.146757   197.146757   197.146757;
192.3176367 192.3176367 192.3176367 192.3176367 192.3176367 192.3176367 192.3176367
192.3176367 192.3176367 192.3176367];

vtY0bare = cat(3,vtY0bareM02, vtY0bareM03, vtY0bareM04, vtY0bareM05, vtY0bareM06);
```

2. Trim (angle of attack measurement)

```
alphaY0bare = [-0.122173048 -0.104719755    -0.017453293    0    0.017453293
0.034906585 0.052359878 0.104719755 0.157079633 0.20943951];
```

3. Trim (q measurement)

```
qY0bareM02 = [-0.244278635  -0.231068542    -0.165172178    -0.151815497    -
0.138247244    -0.124374406    -0.11051    -0.069052309    -0.026603827
0.017466593;
-0.236416977    -0.224449966    -0.164750693    -0.152649748    -0.140358682    -
0.127794343    -0.115239186    -0.07767878 -0.03922529 0.000969565;
-0.223385317    -0.213601614    -0.164787704    -0.154894558    -0.144852619    -
0.134588336    -0.124330893    -0.093625748    -0.06219826 -0.029085358;
-0.21440028 -0.206358734    -0.166226588    -0.158097298    -0.149859177    -
0.141430506    -0.132997536    -0.10776274 -0.081943553    -0.054744935];

qY0bareM06 = [-0.341758062  -0.304342091    -0.114469657    -0.076074145    -
0.035317242    0.004963484 0.044605343 0.16566644  0.286944995 0.411253377;
-0.31365455 -0.279870729    -0.108497838    -0.073842554    -0.037053156    -
0.000689519    0.035094406 0.144372096 0.253847123 0.366087259;
-0.263074908    -0.235843647    -0.097793778    -0.069875275    -0.040230247    -
0.010928094    0.017898095 0.105940915 0.194132289 0.284597744;
-0.220917127    -0.199205789    -0.089227335    -0.066984488    -0.043360695    -
0.020006602    0.002963363 0.073132599 0.143421579 0.215572308];

qY0bare = cat(3,qY0bareM02, qY0bareM03, qY0bareM04, qY0bareM05, qY0bareM06);
```

4. Trim (canard angle command)

```
dcU0newM02 = [0.045000152    0.042274485 0.03138371  0.02928043  0.026904626
0.024207887 0.021569351 0.013718949 0.004575514 -0.007956474;
0.047575981 0.044950957 0.034419854 0.032384928 0.030087934 0.027492888 0.024948639
0.01734825  0.008488121 -0.003751031;
0.054115753 0.051718272 0.042026186 0.040151175 0.038039651 0.035663565 0.033328481
0.026315617 0.018138438 0.006696128;
0.063051384 0.060789671 0.051573209 0.04978822  0.047792244 0.045525881 0.043300236
0.036631072 0.028895642 0.018056463];

dcU0newM06 = [0.010263343    0.009100928 4.15E-03    3.21E-03    2.16E-03    1.11E-03
5.69E-05    -4.38E-03    -9.06E-03    -0.014626502;
0.01078453  0.009570747 0.004395138 0.003413036 0.002312431 0.001220289 0.00011938
-0.004481121    -0.009314465    -0.015097051;
0.012521117 0.011206606 0.005583152 0.004515753 0.003324138 0.002140559 0.000947671
-0.003970774    -0.009122563    -0.015333407;
0.012767416 0.011453365 0.005823788 0.004755385 0.003566688 0.002385062 0.001194289
-0.003661821    -0.008737821    -0.014901204];

dcU0new = cat(3, dcU0newM02, dcU0newM03, dcU0newM04, dcU0newM05, dcU0newM06);
```

5. Trim (outer elevon angle command)

```
doeU0newM02 = [-0.05548964  -0.047794861    -0.017049315    -0.011111586    -
0.004404497    0.003208618 0.010657419 0.032819743 0.058632404 0.094011229;
-0.056524755    -0.048633305    -0.016974295    -0.010856824    -0.00395151
0.003849817 0.011498437 0.034347021 0.060982683 0.097776494;
-0.059780541    -0.051505634    -0.018053397    -0.011581796    -0.004293868
0.003907193 0.011966735 0.036171638 0.064395177 0.103888312;
```

```
-0.066498459    -0.057903834    -0.022880813   -1.61E-02   -0.008512951
9.93426E-05 0.008556907 0.033900067 0.063295095 0.104484525];

doeU0newM06 = [-0.056845764 -0.052027631    -0.031513528    -0.027620868    -
0.023248627    -0.018912239    -0.014540573    0.003869412 0.023238033 0.046320981;
-0.057042653    -0.052119485    -0.031126924    -0.027143463    -0.022679351    -
0.018249559    -0.013784214    0.004875677 0.024479989 0.047934481;
-0.057119355    -0.052011839    -0.030161977    -0.026014609    -0.021384606    -
0.016785822    -0.01215087 0.006959689 0.026976902 0.051109061;
-0.057658861    -0.052350256    -0.029607453    -0.025291236    -0.020489043    -
0.015715422    -0.010904844    0.008713246 0.029219668 0.054118981];

doeU0new = cat(3, doeU0newM02, doeU0newM03, doeU0newM04, doeU0newM05, doeU0newM06);
```

6. Trim (thrust command tss)

```
tssU0new = [0.102839 0.1384 0.2879 0.5186;
    0.03369 0.0442 0.1149 0.2176;
    0.02803 0.0285 0.074 0.1409;
    0.060749 0.0521 0.0764 0.1206;
    0.103866 0.0901 0.103 0.1351];
```

7. Pitch velocity coefficient (Cm_{vt})

```
vtM02 = [-9.70E-04  -8.52E-04   -2.85E-04   -1.69E-04   -5.02E-05   8.10E-05
1.94E-04    5.33E-04    9.35E-04    1.38E-03;
-6.78E-04   -5.81E-04   -1.10E-04   -3.10E-05   8.33E-05    1.86E-04    2.78E-04
5.50E-04    8.77E-04    1.24E-03;
-9.53E-05   -3.11E-05   0.000281926 0.000307991 4.10E-04    4.79E-04    0.00053812
0.000708454 0.000920091 0.001164256;
0.000553917 0.000596004 8.02E-04    8.02E-04    8.86E-04    0.000931507 0.000968799
0.001072362 1.21E-03    1.37E-03];

vtM06 = [0.007977405    0.006586224 0.001348917 0.000405191 -0.000454312    -
0.001333903    -0.002261116    -0.006524502    -0.010125671    -0.012965807;
0.007735237 0.006440713 0.001481456 0.000580465 -0.0002588  -0.001109686    -
0.002010579    -0.006068256    -0.009581964    -0.012303732;
0.006781378 0.005685649 0.001380943 0.000585768 -1.71E-04   -0.000939117    -
0.001749744    -0.005263916    -0.008415858    -0.010769277;
0.005867297 0.004967305 0.001369017 0.000696769 4.08E-05    -0.000623346    -
0.001323607    -0.004308065    -0.007064843    -0.00907646];

vtModifier = cat(3,vtM02, vtM03, vtM04, vtM05, vtM06);
```

8. Pitch α coefficient (Cm_{α})

```
alphaM02 = [2.501781893 1.831292597 1.856276232 1.846472166 2.287811051 2.268046116
2.312129337 2.322861867 3.296730007 3.474023765;
2.217211539 1.625124255 1.645769458 1.639140309 2.033090623 2.010769397 2.053093642
2.068847698 2.93642853  3.105571669;
1.720026833 1.267185239 1.280576905 1.279552557 1.585033152 1.566110445 1.604094607
1.622401548 2.297391754 2.458990323;
1.316489143 0.981658602 0.989041037 0.992230269 1.215092822 1.214329577 1.247489319
1.257067689 1.763323394 1.912934429];

alphaM06 = [14.88518477 14.65436373 13.51901275 14.98977917 14.82924721 14.65524044
17.87350778 19.09566933 20.04850979 22.24916445;
```

```
13.22221904 13.00924857 11.96169959 13.2609372  13.12075268 12.97084038 15.83268315
16.87781212 17.81584153 19.82643541;
10.32192952 10.14445455 9.278946628 10.28521293 10.17747705 10.06722075 12.30156358
13.04882029 13.92808523 15.58334358;
7.986701408 7.841070657 7.131866677 7.899930751 7.819540553 7.739320177 9.461604428
9.99272168  10.77395534 12.11732672];

alphaModifier = cat(3,alphaM02,alphaM03,alphaM04,alphaM05, alphaM06);
```

9. Pitch q coefficient (Cm_q)

```
qM02 = [-0.586630986    -0.586565729    -0.586694477    -0.586799202    -0.586937598
-0.586265983    -0.59309785 -0.627566854    -0.645594852    -0.639972958;
-0.527746597    -0.527676113    -0.527717503    -0.527794284    -0.527900254    -
0.527172476    -0.533152181    -0.563611243    -0.578980621    -0.575573243;
-0.422160424    -0.422079513    -0.421962133    -0.421988833    -0.422036844    -
0.421209774    -0.425665845    -0.448953597    -0.459581965    -0.456151414;
-0.333903293    -0.333811133    -0.333564056    -0.33355215 -0.333555976    -
0.332670841    -0.33589405 -0.353360668    -0.360352585    -0.355204426];

qM06 = [-2.084723847    -2.084673754    -2.084538741    -2.08453488 -2.097160524
-2.109793897    -2.122435008    -2.160406907    -2.198452573    -2.212008795;
-1.875941966    -1.875900903    -1.875790224    -1.875787059    -1.887169968    -
1.898559213    -1.9099548 -1.944181399    -1.978468512    -1.990628209;
-1.504623007    -1.504595977    -1.504523121    -1.504521038    -1.513675751    -
1.522834635    -1.531997694    -1.559513132    -1.587068478    -1.596819074;
-1.194732198    -1.194714777    -1.194667819    -1.194666477    -1.201957385    -
1.209250981    -1.216547268    -1.238453087    -1.260384689    -1.268130415];

qModifier = cat(3,qM02, qM03, qM04, qM05, qM06);
```

10. Pitch canard coeffiecient (Cm_c)

```
dcM02 = [0.744881832    0.746341064 0.77064322  0.777880061 0.790317544 0.802092896
0.815872958 0.862551549 0.91809184  1.045960628;
0.664273854 0.664823804 0.685633594 0.692058513 0.703115025 0.713756138 0.725941084
0.766983987 0.815125065 0.922647949;
0.522276721 0.521206164 0.531282404 0.537548067 0.548554457 0.557674545 0.567076739
0.598422537 0.634195765 0.711201241;
0.406309054 0.403860443 0.402875528 0.406240381 0.414184625 0.421211904 0.430290954
0.459968848 0.48618189  0.542679175];

dcM06 = [6.072849917    6.110596803 6.315318453 6.356227405 6.369646256 6.381151561
6.39259343  6.43630527  6.447394964 6.759861221;
5.451356816 5.48509604  5.667924201 5.704468983 5.716434526 5.726690497 5.73689532
5.776158797 5.786590268 6.061587072;
4.345848444 4.372567195 4.517068035 4.545967021 4.555343247 4.563365037 4.571354315
4.601494328 4.609694507 4.823043135;
3.420207594 3.441107501 3.553934746 3.576509195 3.583765707 3.589961774 3.59613762
3.619188171 3.624956842 3.790042883];

dcModifier = cat(3, dcM02, dcM03, dcM04, dcM05, dcM06);
```

11. Pitch outer elevon coefficient (Cm_{oe})

```
doeM02 = [-0.669701442  -0.666808412    -0.689791231    -0.690108294    -0.691546068
-0.70414391 -0.716659524    -0.71892899 -0.652262048    -0.570570606;
```

```
-0.596408566     -0.593568849     -0.614219919     -0.614331295     -0.615466717     -
0.626534699     -0.637634313     -0.639539298     -0.576181699     -0.508182287;
-0.466960451     -0.464399743     -0.480795977     -0.480615311     -0.481049738     -
0.489431078     -0.498039959     -0.49931513 -0.44510578 -0.396918215;
-0.360888313     -0.358931167     -0.371905665     -0.37168646 -0.37129463 -0.377175584
-0.383773784     -0.384610156     -0.343694315     -0.305444728];

doeM06 = [-6.318313746   -6.476353487     -6.643134627     -6.605321692     -6.552239934
-6.481059695     -6.404298745     -6.543611944     -6.529006821     -5.526005605;
-5.70822268 -5.850956517     -5.997931413     -5.96215914 -5.913621142     -5.848342107
-5.777979796     -5.911909863     -5.898472983     -4.966669563;
-4.59246123 -4.707250478     -4.818358685     -4.786979756     -4.747330409     -
4.693439637     -4.635363926     -4.756872164     -4.74592408 -3.955303067;
-3.634280928     -3.725088436     -3.809870543     -3.783345898     -3.75123481 -
3.707455049     -3.660311031     -3.764744478     -3.755986057     -3.099657146];

doeModifier = cat(3, doeM02, doeM03, doeM04, doeM05, doeM06);
```

12. Pitch inner elevon coefficient (Cm$_{ie}$)

```
dieM02 = [-1.064549985 -1.060033622     -1.098161594     -1.099559549     -1.102696237
-1.123078418     -1.142671003     -1.145325839     -1.038191094     -0.976248729;
-0.947499121     -0.943049598     -0.977386655     -0.978470489     -0.981164448     -
0.999085136     -1.016472891     -1.018718129     -0.917106982     -0.869023279;
-0.741102186     -0.737071593     -0.764326403     -0.764972447     -0.766626372     -
0.780390788     -0.793903141     -0.795440537     -0.708796056     -0.678841923;
-0.572290017     -0.569207437     -0.590231094     -0.590817763     -0.591194631     -
0.601586765     -0.611961935     -0.61299823 -0.547723003     -0.522955743];

dieM06 = [-10.64706423   -10.91343901     -11.19802056     -11.13504872     -11.04649482
-10.92753488     -10.79918245     -11.03706862     -11.01190848     -9.99251505;
-9.556553305     -9.795558608     -10.04477169     -9.985554566     -9.905093111     -
9.796675723     -9.679757934     -9.906515032     -9.883535175     -8.922718164;
-7.600325825     -7.790319855     -7.97670447 -7.925307853     -7.860318536     -
7.771802547     -7.676367925     -7.879071711     -7.860582233     -7.024140924;
-5.958179324     -6.10706351 -6.24798436 -6.204918448     -6.152762634     -6.081502965
-6.004734102     -6.176986071     -6.162346646     -5.452979824];

dieModifier = cat(3, dieM02, dieM03, dieM04, dieM05, dieM06);
```

13. Pitch thrust coefficient (Cm$_{tss}$)

```
tssM02 = [-0.133394461   -0.132560028     -0.129666877     -0.12936421 -0.129063328
-0.128839707     -0.128738995     -0.128912587     -0.129789194     -0.131354451;
-0.115765227     -0.115033521     -0.112516767     -0.112262607     -0.112038115     -
0.111910703     -0.111857418     -0.112138328     -0.113071157     -0.114642975;
-0.097539491     -0.097371578     -0.096527955     -0.096358946     -0.096189676     -
0.096020588     -0.095851609     -0.095345905     -0.094843245     -0.094345158;
-0.096334142     -0.096201446     -0.095534253     -0.095400486     -0.09526649 -
0.095132601     -0.094998768     -0.094598056     -0.094199479     -0.093798605];

tssM06 = [-0.136333652   -0.135492739     -0.131170416     -0.130284503     -0.129392947
-0.128495448     -0.127591975     -0.124851202     -0.122072043     -0.119294253;
-0.128933279     -0.12822313 -0.124549207     -0.123792143     -0.123028289     -
0.122258164     -0.121482048     -0.119121168     -0.116718707     -0.114310606;
-0.113951517     -0.113454222     -0.110867202     -0.110331563     -0.109789832     -
0.109242979     -0.108691005     -0.107006794     -0.105285324     -0.103551897;
```

```
-0.095838143     -0.095507005     -0.093790521     -0.093436195     -0.093077901     -
0.092716528     -0.092351955     -0.091240353     -0.09010474 -0.088960315];

tssModifier = cat(3, tssM02, tssM03, tssM04, tssM05, tssM06);
```

14. Pitch landing gear coefficient (Cm$_{ldg}$)

```
ldgM02 = [-0.056234024  -0.056885492     -0.060139424     -0.061836842     -0.064703658
-0.067578197     -0.070452088     -0.0790146 -0.087369745     -0.095479946;
-0.049972273     -0.050547925     -0.05342343 -0.054928922     -0.057473576     -
0.060024484     -0.062574867     -0.070173739     -0.077588579     -0.08478567;
-0.038978224     -0.039422562     -0.041642496     -0.042812618     -0.044793255     -
0.046777897     -0.048762205     -0.054674772     -0.060444143     -0.066043558;
-0.030031036     -0.030370298     -0.032065513     -0.032964298     -0.034487532     -
0.036013275     -0.03753881 -0.042084596     -0.046520287     -0.05082496];

ldgM06 = [-0.520647681  -0.526697685     -0.556903304     -0.572652724     -0.599240707
-0.625877399     -0.652505633     -0.731829704     -0.809225985     -0.884362936;
-0.461470373     -0.466801789     -0.493423083     -0.507353342     -0.530889055     -
0.554464434     -0.578032992     -0.648245485     -0.716752117     -0.783254694;
-0.358187537     -0.362282382     -0.382733868     -0.393505582     -0.411731224     -
0.429982844     -0.448230129     -0.502594458     -0.555638717     -0.607125324;
-0.274782746     -0.277895723     -0.293446472     -0.301683261     -0.315637193     -
0.329607918     -0.343575946     -0.385193887     -0.42580178 -0.465213356];

ldgModifier = cat(3, ldgM02, ldgM03, ldgM04, ldgM05, ldgM06);
```

The following data is a subset of the table data used for the lateral cg estimator. The trim data is identical to that used for the longitudinal cg estimator and is therefore omitted.

15. Roll β coefficient (Cl$_{β}$)

```
betaM02 = [-0.162027722 -0.164891702     -2.222342025     -3.138159387     -4.058392272
-4.975395177     -5.880919682     -8.349737516     -10.7027275 -12.37735004;
2.515289275 2.066351287 -1.989523176     -2.804880905     -3.624628755     -4.44053894
-5.246814041     -7.445164861     -9.541200464     -11.03393647;
1.951797839 1.599684265 -1.575007832     -2.213481466     -2.856222881     -3.496125745
-4.127759663     -5.850038555     -7.493412926     -8.667737178;
1.499671595 1.228077643 -1.223673268     -1.717266773     -2.214779827     -2.714163991
-3.202786469     -4.534769615     -5.806117385     -6.717460826];

betaM06 = [30.00993677  24.94810629 -16.97434045     -26.01113893     -35.01898888
-44.02234872     -53.02185105     -79.12864992     -99.79294125     -110.4419417;
26.7407878  22.23957465 -15.02814823     -23.06091923     -31.06808278     -39.07118122
-47.07074519     -70.25824872     -88.62081372     -98.09235147;
20.95791378 17.44369846 -11.64254567     -17.91132478     -24.16023508     -30.40586212
-36.64858432     -54.71727134     -69.03605347     -76.42310908;
16.28805672 13.57532294 -8.860705759     -13.69563921     -18.51565794     -23.33304215
-28.14809115     -42.07090451     -53.11379616     -58.82117551];

betaModifier = cat(3,betaM02, betaM03, betaM04, betaM05, betaM06);
```

16. Roll p coefficient (Cl$_{p}$)

```
pM02 = [-1.267641913    -1.26503159 -1.259530137    -1.258720666    -1.257895977
-1.25706553 -1.256244318    -1.253838371    -1.254865836    -1.256709978;
-1.141054656    -1.138705355    -1.133735706    -1.133002711    -1.132256117    -
1.131504356    -1.130760959    -1.128581713    -1.129484352    -1.131102476;
-0.914109029    -0.912224013    -0.908197162    -0.907599534    -0.906991305    -
0.906378728    -0.905772784    -0.903994689    -0.904671453    -0.905898352;
-0.72451549 -0.723012992    -0.719754188    -0.719266118    -0.718770172    -
0.718270053    -0.717774687    -0.716320413    -0.716800191    -0.717707442];

pM06 = [-3.890236326    -3.889071541    -3.883138104    -3.881933596    -3.880578745
-3.879622133    -3.879071    -3.879551316    -3.883346861    -3.910300201;
-3.519745766    -3.518689945    -3.513315147    -3.512224061    -3.510997641    -
3.510128702    -3.509624119    -3.510025662    -3.513403459    -3.537573245;
-2.845864031    -2.845007123    -2.840649377    -2.839764732    -2.838771251    -
2.838062824    -2.837645427    -2.837917985    -2.840562356    -2.859778314;
-2.271887466    -2.271199721    -2.267706765    -2.266997663    -2.266201978    -
2.265630976    -2.265289823    -2.265466975    -2.267513443    -2.282623948];

pModifier = cat(3,pM02,pM03,pM04,pM05, pM06);
```

17. Roll r coefficient (Cl_r)

```
rM02 = [-0.069670453    -0.000636213    0.285430819 0.339633474 0.368475291
0.39552178  0.422321488 0.502513176 0.608023698 0.728469684;
-0.047864865    0.014374752 0.272292045 0.321160731 0.347258895 0.371730416
0.395997185 0.468711138 0.564386472 0.674008783;
-0.008576225    0.041344135 0.248171879 0.2873584    0.308393336 0.328143564
0.347741752 0.406583337 0.484034933 0.573532901;
0.023944948 0.063353076 0.226523711 0.257431699 0.274000346 0.289636597 0.305132315
0.351620029 0.412842722 0.483977413];

rM06 = [-0.218359872    -0.076333373    0.615129709 0.752034252 0.828300333
0.90483576  0.982098559 1.297333464 1.741093628 2.112063285;
-0.191206553    -0.063260871    0.559556391 0.682850711 0.751618934 0.820633043
0.890307163 1.17423747  1.573697046 1.908288804;
-0.141058218    -0.038538845    0.460412836 0.559161511 0.614366388 0.669779977
0.725734899 0.953213424 1.27284391  1.541601771;
-0.100249397    -0.018980833    0.376376485 0.454597714 0.498432804 0.542438163
0.586880787 0.767217388 1.020282797 1.233908212];

rModifier = cat(3,rM02, rM03, rM04, rM05, rM06);
```

18. Roll φ coefficient (Cl_φ)

```
phiM02 = [-0.00074647   -0.002540531    -0.004566493    -0.004604347    -0.004640802
-0.004675824    -0.004709382    -0.004800933    -0.007920561    -0.011324908;
-0.000679027    -0.002310983    -0.00415379 -0.004188202    -0.004221342    -
0.004253178    -0.004283682    -0.004366894    -0.007204382    -0.010300757;
-0.000555804    -0.001891592    -0.003399823    -0.003427959    -0.003455052    -
0.003481079    -0.003506014    -0.003574026    -0.005896173    -0.00843007;
-0.000450357    -0.00153271 -0.002754688    -0.002777464    -0.002799395    -
0.002820462    -0.002840643    -0.002895682    -0.004776983    -0.00682975];

phiM06 = [-0.003315811  -0.003322411    -0.003340203    -0.003340712    -0.003847905
-0.00435367 -0.004857541    -0.00635318 -0.007815199    -0.014239432;
-0.00301599 -0.003021993    -0.003038176    -0.003038639    -0.003500012    -
0.00396008 -0.004418423    -0.005778892    -0.007108742    -0.012952026;
```

```
-0.00246848 -0.002473394    -0.002486639    -0.002487018    -0.00286461 -0.003241129
-0.003616229    -0.004729575    -0.005817805    -0.010599648;
-0.002000025    -0.002004007    -0.002014738    -0.002015045    -0.002320963    -
0.002626007    -0.002929897    -0.003831855    -0.004713424    -0.008587326];

phiModifier = cat(3,phiM02, phiM03, phiM04, phiM05, phiM06);
```

19. Roll canard coefficient (Cl$_c$)

```
dcM02 = [-0.085984776    -0.044417768    0.163111263 0.204600813 0.246090596
0.287580612 0.329070862 0.453543015 0.578152718 0.702842154;
-0.076409436    -0.039470959    0.144943096 0.1818113    0.218679693 0.255548275
0.292417045 0.403024488 0.513743111 0.624526176;
-0.059596497    -0.030785288    0.11304455    0.141798897 0.170553364 0.199307951
0.228062658 0.314327502 0.400663316 0.487040258;
-0.045913561    -0.02371682 0.087086602 0.109238183 0.131389839 0.15354157
0.175693376 0.242149243 0.308649407 0.37517524];

dcM06 = [1.054953604    1.054953604 1.054953604 1.054953604 1.611268021 2.167621971
2.724015462 4.178987884 5.205328929 5.907169026;
0.937898265 0.937898265 0.937898265 0.937898265 1.432135223 1.926404135 2.420705004
3.713359356 4.625343082 5.249106908;
0.731462546 0.731462546 0.731462546 0.731462546 1.116924405 1.502406661 1.887909314
2.896013978 3.60720091    4.093195743;
0.563482386 0.563482386 0.563482386 0.563482386 0.860428796 1.157387936 1.454359808
2.230933421 2.778753664 3.152826405];

dcModifier = cat(3, dcM02, dcM03, dcM04, dcM05, dcM06);
```

20. Roll outer elevon coefficient (Cl$_{oe}$)

```
doeM02 = [-3.008565944 -3.084747796    -3.116496999    -3.09835469 -3.090796884
-3.081220532    -3.071548531    -3.004259221    -2.898760233    -2.737335335;
-2.678379593    -2.747851222    -2.776041209    -2.759571143    -2.752515227    -
2.743668047    -2.734707567    -2.673527442    -2.576914971    -2.429988518;
-2.093906425    -2.150694254    -2.175413554    -2.16237016 -2.156377233    -
2.148945129    -2.141394808    -2.091621246    -2.012419335    -1.892649209;
-1.612355318    -1.657347944    -1.685304073    -1.675940355    -1.67108535 -
1.664996522    -1.658817243    -1.619299659    -1.557190402    -1.462301457];

doeM06 = [-22.69262835 -23.05253896    -24.02286489    -24.2308755 -24.38402593
-24.51464024    -24.32223088    -22.91670776    -21.21970409    -19.18297172;
-20.64589705    -20.97364893    -21.85072524    -22.03900586    -22.17648601    -
22.29328384    -22.11597938    -20.83020723    -19.27791123    -17.40349244;
-16.83892241    -17.10688389    -17.80836526    -17.96052029    -18.06948393    -
18.16133612    -18.01338814    -16.95269776    -15.67266359    -14.11037842;
-13.48779712    -13.7030106 -14.25964664    -14.38009593    -14.46502134    -
14.53603197    -14.41493374    -13.55561748    -12.51853373    -11.23960402];

doeModifier = cat(3, doeM02, doeM03, doeM04, doeM05, doeM06);
```

21. Roll inner elevon coefficient (Cl$_{ie}$)

```
dieM02 = [-2.594651561    -2.661140856    -2.685862118    -2.668144706    -2.66054783
-2.651052472    -2.641473547    -2.579692071    -2.486504388    -2.345873399;
-2.309119947    -2.369726606    -2.391587764    -2.375524103    -2.36845887 -
2.359708643    -2.350857848    -2.294693292    -2.209486845    -2.081522832;
```

```
-1.804213556    -1.853712785    -1.87304803 -1.86032639 -1.854364022    -1.847048406
-1.839626154    -1.793942859    -1.724218937    -1.619949406;
-1.388739056    -1.427913511    -1.450591352    -1.441382006    -1.436571527    -
1.430603574    -1.424552661    -1.388288074    -1.33351684 -1.250913384];

dieM06 = [-20.16305718  -20.48948104    -21.38605179    -21.57844251    -21.72175441
-21.8448421 -21.67798463    -20.43271332    -18.91317432    -17.08501185;
-18.27422956    -18.57032189    -19.37790641    -19.55145028    -19.67968715    -
19.78943839    -19.63624013    -18.50150066    -17.11687988    -15.44046638;
-14.80194763    -15.04232789    -15.68442307    -15.82381078    -15.92489029    -
16.01078704    -15.88379624    -14.95424041    -13.820344  -12.43211075;
-11.78884561    -11.98077313    -12.48765573    -12.59746089    -12.67590758    -
12.74208223    -12.63870085    -11.88997242    -10.97652888    -9.846141819];

dieModifier = cat(3, dieM02, dieM03, dieM04, dieM05, dieM06);
```

Appendix I - Extended Kalman filter Matlab code

```
function [tot_unex_qdot] = EKF(y, u, u0)
% y = measurement vector
% u = command vector
% u0 = initial command vector from trim
% tot_unex_qdot = estimate of the unexpected qdot

eml.extrinsic('dlinmod');
%
% The function returns an Extended Kalman Filter update.
%

n=11; %number of states

% Define persistent data to store between calls
persistent P
if isempty(P)
    P = eye(n);
end;

% Previous set of inputs
persistent old_u
if isempty(old_u)
    old_u = u0;
end;

% Estimate of unexpected qdot
persistent tot_qunex
if isempty(tot_qunex)
    tot_qunex = 0;
end;

% EKF state (always a delta from last linearisation point)
persistent x
if isempty(x)
    x = [0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0];
end;

% Previous state estimate stored for linearisation
persistent old_state
if isempty(old_state)
    old_state = [128.2118; 0.0986; 0; 0; 0; 0; 0; 0.0986; 0; 0; 0; -5000];
end;


G = eye(n);

Q = [
10 0 0 0 0 0 0 0 0 0 0;          %Vt
0 0.02 0 0 0 0 0 0 0 0 0;        %alpha
0 0 0.01 0 0 0 0 0 0 0 0;        %beta
0 0 0 0.01 0 0 0 0 0 0 0;        %p
0 0 0 0 0.01 0 0 0 0 0 0;        %q
0 0 0 0 0 0.01 0 0 0 0 0;        %r
0 0 0 0 0 0 0.01 0 0 0 0;        %phi (bank)
```

```
0 0 0 0 0 0 0 0.01 0 0 0;            %theta (pitch)
0 0 0 0 0 0 0 0 0.01 0 0;            %psi (heading)
0 0 0 0 0 0 0 0 0 10 0;              %z
0 0 0 0 0 0 0 0 0 0 0.2];            %unexpected qdot

R = [
0.001156 0 0 0 0 0 0 0 0 0;          %Vt
0 0.00000000487 0 0 0 0 0 0 0 0;     %alpha
0 0 0.00000000487 0 0 0 0 0 0 0;     %beta
0 0 0 0.00000595 0 0 0 0 0 0;        %p
0 0 0 0 0.00000076 0 0 0 0 0;        %q
0 0 0 0 0 0.00000076 0 0 0 0;        %r
0 0 0 0 0 0 0.00000019 0 0 0;        %phi (bank)
0 0 0 0 0 0 0 0.00000000487 0 0;     %theta (pitch)
0 0 0 0 0 0 0 0 0.00000000487 0;     %psi (yaw)
0 0 0 0 0 0 0 0 0 1];                %z


% Get data to linearise aircraft model for next iteration

% Delta from trim inputs used for linearisation
temp_u = old_u - u0;

Ared = zeros(12,12);
Bred = zeros(12,16);
Cred = zeros(12,12);
Dred = zeros(12,16);

% Linearise aircraft model at new estimated state
[Ared,Bred,Cred,Dred] = dlinmod('admire_bare_reduced',0.01,old_state(1:12),temp_u);

AdisEst = zeros(11,11);
BdisEst = zeros(11,10);
CdisEst = zeros(10,11);
DdisESt = zeros(10,10);

% Create Kalman filter state-space matrices and strip out X and Y states
% Add extra state for unexpected qdot
AdisEstA = zeros(11,11);
AdisEst(1:9,1:9) = Ared(1:9,1:9);
AdisEst(10,1:9) = Ared(12,1:9);
AdisEst(1:9,10) = Ared(1:9,12);
AdisEst(10,10) = Ared(12,12);
AdisEst(5,11) = 0.01;
AdisEst(11,11) = 1;

% Strip out X and Y states
BdisEst = zeros(11,10);
BdisEst(1:9,1:10) = Bred(1:9,1:10);
BdisEst(10,1:10) = Bred(12,1:10);

% Strip out X and Y states
CdisEst = zeros(10,11);
CdisEst(1:9,1:9) = Cred(1:9,1:9);
CdisEst(10,1:9) = Cred(12,1:9);
CdisEst(1:9,10) = Cred(1:9,12);
CdisEst(10,10) = Cred(12,12);
```

```matlab
DdisEst = zeros(10,10);

% Calculate delta measurements (difference from old state)
delta_y = zeros(10,1);

delta_y(1:9) = y(1:9) - old_state(1:9);
delta_y(10) = y(12) - old_state(12);

% Calculate delta command, difference from linearisation point command
delta_u = u(1:10) - old_u(1:10);

% Predict state
y1 = CdisEst*x;
x = AdisEst*x + BdisEst*delta_u(1:10);

% Predict plant covariance
P = AdisEst*P*AdisEst' + Q;
S = CdisEst*P*CdisEst' + R;

% Calculate Kalman gain
K = P*CdisEst'*inv(S);

% Update estimate with measurement correction
x = x+K*(delta_y-y1);

% Correct plant covariance
P = (eye(n)-K*CdisEst)*P;

% Calculate estimated aircraft state to use as linearisation point on next
iteration
ac_state = zeros(12,1);
ac_state(1:9) = old_state(1:9) + x(1:9);
ac_state(10) = 0;
ac_state(11) = 0;
ac_state(12) = old_state(12) + x(10);

old_state = ac_state;

old_u = u;

% Update estimate of unexpected qdot
tot_qunex = tot_qunex + x(11);

% Set output to unexpected qdot
tot_unex_qdot = tot_qunex;
```