A Backstepping Design for Flight Path Angle Control

Ola Härkegård and S. Torkel Glad

Division of Automatic Control, Department of Electrical Engineering,
Linköpings universitet, SE-581 83 Linköping, Sweden.
E-mail: ola@isy.liu.se, torkel@isy.liu.se

Abstract

A nonlinear approach to flight path angle control is presented. Using backstepping, a globally stabilizing control law is derived. Although the nonlinear nature of the lift force is considered, the pitching moment to be produced is only linear in the measured states. Thus, the resulting control law is much simpler than if feedback linearization had been used. The free parameters that spring from the backstepping design are used to achieve a desired linear behavior around the operating point.

1 Introduction

While many conventional flight control designs assume the aircraft dynamics to be linear about some nominal flight condition, this paper deals with the problem of controlling an aircraft explicitly considering its nonlinear dynamics. Contributions along this line often use feedback linearization [9], i.e., controllers based upon dynamic inversion, to deal with the nonlinearities. The control law is designed to cancel the nonlinearities so that the closed loop system is rendered linear.

Two motivating reasons for this can be found. First, a linear system is easy to analyze with respect to stability and performance. Second, a linear system has the same dynamic properties independently of the operating point, e.g., speed, height, and angle of attack in the aircraft case. The second is true only for linear systems and cannot be competed with. We will therefore focus on the first property and present an alternative way of finding control laws that guarantee stability and can be tuned to achieve a certain closed loop behavior.

Our main tool will be backstepping [4], a Lyapunov based design method that has received a lot of attention during the last decade. Compared to feedback linearization, backstepping offers a more flexible way of dealing with nonlinearities. Using Lyapunov functions, their impact on the system can be analyzed so that stabilizing, and thus in a sense useful, nonlinearities may be kept while harmful nonlinearities can be cancelled or dominated by the control signal. Not having to cancel all nonlinearities means that the resulting control law may be much simpler than if feedback linearization had been used.

In this paper, the application is flight path angle control. Using inherent characteristics of the lift force, we will show that despite its nonlinear behavior around the stall angle, it suffices to produce a pitching moment that is linear in the measured states to stabilize the aircraft around the desired trajectory, regardless of the initial state.

The remainder of the paper is organized as follows. Section 2 presents backstepping in an informal setting. In Section 3, a nonlinear model describing the longitudinal aircraft dynamics in pure-feedback form, crucial to the backstepping design, is derived. In Section 4, the backstepping control law is derived in detail, and in Section 5 the control law is evaluated using a realistic simulation model. Section 6 contains a comparison with feedback linearization.

2 Backstepping

For backstepping [4] to be applicable, the differential equations describing the system dynamics need to have a certain *pure-feedback* structure:

$$\begin{aligned}\dot{x}_1 &= f_1(x_1, x_2)\\ &\vdots\\ \dot{x}_{n-1} &= f_{n-1}(x_1, \dots, x_{n-1}, x_n)\\ \dot{x}_n &= f_n(x_1, \dots, x_n, u)\end{aligned} \tag{1}$$

The reason for this is the way a backstepping control law, $u(x)$, is recursively constructed with the aim of bringing the state vector, x, to the origin.

To begin with, x_2 is considered a *virtual control* of the scalar $\dot{x}_1$ subsystem. A *stabilizing function*, $\phi_1(x_1)$, is determined such that $\dot{x}_1 = f_1(x_1, \phi_1(x_1))$ has the (stable) properties desired. However, x_2 is not available for control. The key property of backstepping is that it offers a constructive way of forwarding the unattainable control demand $x_2 = \phi_1(x_1)$ to a new virtual control law $x_3 = \phi_2(x_1, x_2)$. If this could be satisfied, x_1 and

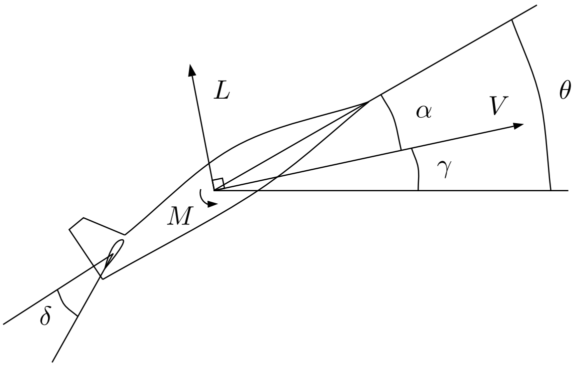

Figure 1: Illustration of longitudinal aircraft entities.

x_2 would be successfully brought to the origin. This recursive procedure is repeated until the actual control variable u is reached after n steps, whereby a stabilizing control law, $u = \phi_{n-1}(x)$, has been constructed.

Along with the control law, a Lyapunov function is constructed, proving the stability of the closed loop system.

3 Aircraft model

In this paper only control about the longitudinal axis is considered. With this restriction, the equations of motion describing the aircraft take the form [10]

$$\dot{V} = \frac{1}{m}(-D + F_T \cos\alpha - mg\sin\gamma) \tag{2}$$

$$\dot{\gamma} = \frac{1}{mV}(L + F_T \sin\alpha - mg\cos\gamma) \tag{3}$$

$$\dot{\theta} = q \tag{4}$$

$$\dot{q} = \frac{1}{I_y}(M + F_T Z_{TP}) \tag{5}$$

The state variables are V = airspeed, γ = flight path angle, θ = pitch angle, and q = pitch rate. $\alpha = \theta - \gamma$ is the angle of attack, F_T is the engine thrust force, which contributes to the pitching moment due to a thrust point offset, Z_{TP}. The aerodynamical effects on the aircraft are captured by the lift and drag forces, L and D, and the pitching moment, M. See Figure 1, where δ represents the deflections of the control surfaces that are at our disposal.

The controlled variables are V and γ. In the implemented controller, evaluated in Section 5, airspeed control is handled separately using results from [7]. Our control design will therefore focus solely on Equations (3)-(5).

To meet the structural demand in Section 2, we will neglect the dependence of the lift force on the pitch rate and the control surface deflections. The principal function of the control surfaces to produce an angular pitching moment. This will affect the angle of attack to which the lift force is strongly related. Note that this is exactly the same assumption which is needed for dynamic inversion to be applied to aircraft control [6]. Expressing L and M in terms of their aerodynamic coefficients, we have that

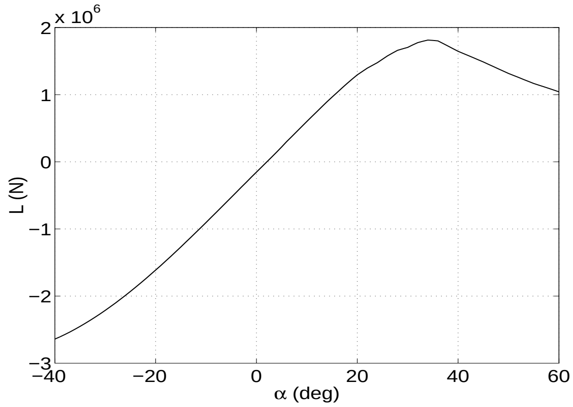

Figure 2: Typical lift force vs angle of attack relationship.

$$\begin{aligned} L &= \bar{q}SC_L(\alpha) \\ M &= \bar{q}S\bar{c}C_m(\alpha, q, \delta) \end{aligned} \tag{6}$$

where $\bar{q} = \frac{1}{2}\rho V^2$ is the dynamic pressure, ρ is the air density, S is the wing platform area, and $\bar{c}$ is the mean aerodynamic chord.

4 Design for flight path angle control

4.1 Deriving the control law

The system description consists of Equations (3)-(5), where the flight path angle γ is to be controlled. Given the reference value, γ_{ref}, the angle of attack at steady state, α_0, is defined through $\dot{\gamma} = 0$, see Equation (3). In the actual implementation, α_0 is computed on-line and updated at each time step, using current values of $\bar{q}$, δ[1], and F_T. Although this pragmatic approach to determining α_0 is not guaranteed to converge, successful simulations provide an alibi.

In Equation (3), the γ dependence of the gravitational term plays an insignificant role. Therefore we will only consider its contribution at the equilibrium, and assume the dynamics to be

$$\begin{aligned} \dot{\gamma} &= \frac{1}{mV}(L(\alpha) + F_T \sin\alpha - mg\cos\gamma_{\text{ref}}) \\ &\triangleq \varphi(\alpha - \alpha_0) \end{aligned} \tag{7}$$

where $\varphi(0) = 0$. Using the sign properties of its components, $L(\alpha)$, see Figure 2 for a typical example, and $F_T \sin\alpha$, we conclude that

$$\alpha\varphi(\alpha) > 0, \quad \alpha \neq 0 \tag{8}$$

[1] In computing α_0, the lift force dependence on δ *is* included.

for all α of practical interest[2]. This is a vital property for the backstepping control design to follow which will be done in three steps.

Step 1: First introduce the control error

$$z_1 = \gamma - \gamma_{\text{ref}}$$

whose dynamics are given by

$$\dot{z}_1 = \varphi(\alpha - \alpha_0) \tag{9}$$

considering a constant reference value. Now use the control Lyapunov function (clf)

$$V_1 = \frac{1}{2} z_1^2$$

to determine a stabilizing function, θ_{des}, considering θ as the control input of Equation (9).

$$\begin{aligned}
\dot{V}_1 &= z_1 \varphi(\alpha - \alpha_0) \\
&= z_1 \varphi(\theta - z_1 - \gamma_{\text{ref}} - \alpha_0) \\
&= z_1 \varphi(-(1+c_1)z_1 + \theta + c_1 z_1 - \gamma_{\text{ref}} - \alpha_0) \\
&= z_1 \varphi(-(1+c_1)z_1) < 0, \quad z_1 \neq 0
\end{aligned}$$

is achieved by selecting

$$\theta_{\text{des}} = -c_1 z_1 + \gamma_{\text{ref}} + \alpha_0, \quad c_1 > -1 \tag{10}$$

The fact that $c_1 = 0$ is a valid choice means that γ feedback is not necessary for the sake of stabilization. However, it provides an extra degree of freedom for tuning the closed loop performance.

Step 2: Continue by introducing the deviation from the virtual control law (10).

$$z_2 = \theta - \theta_{\text{des}} = \theta + c_1 z_1 - \gamma_{\text{ref}} - \alpha_0$$

Defining

$$\xi = -(1+c_1)z_1 + z_2$$

we have that

$$\begin{aligned}
\dot{z}_1 &= \varphi(\xi) \\
\dot{z}_2 &= q + c_1 \varphi(\xi)
\end{aligned}$$

We will also need

$$\dot{\xi} = -\varphi(\xi) + q$$

A regular backstepping design would proceed by expanding the control Lyapunov function with a term penalizing z_2. We do this, but also add a term $F(\xi)$ as an extra degree of freedom, where F is required to be positive definite and radially growing. This extension of the backstepping procedure is due to [5]. Hence,

$$V_2 = \frac{c_2}{2} z_1^2 + \frac{1}{2} z_2^2 + F(\xi), \quad c_2 > 0$$

We compute its time derivative to find a new stabilizing function, q_{des}.

$$\begin{aligned}
\dot{V}_2 &= c_2 z_1 \varphi(\xi) + z_2(q + c_1 \varphi(\xi)) + F'(\xi)(-\varphi(\xi) + q) \\
&= (c_2 z_1 + c_1 z_2 - F'(\xi))\varphi(\xi) + (z_2 + F'(\xi))q
\end{aligned}$$

Although it may not be transparent, we can again find a stabilizing function independent of φ. Choosing

$$q_{\text{des}} = -c_3 z_2, \quad c_3 > 0 \tag{11}$$

$$F'(\xi) = c_4 \varphi(\xi), \quad F(0) = 0, \; c_4 > 0 \tag{12}$$

where (12) is an implicit but perfectly valid choice of F, yields

$$\dot{V}_2 = (c_2 z_1 + (c_1 - c_3 c_4) z_2)\varphi(\xi) - c_4 \varphi(\xi)^2 - c_3 z_2^2$$

Selecting

$$c_2 = -(1+c_1)(c_1 - c_3 c_4), \quad c_3 c_4 > c_1 \tag{13}$$

gives the negative definiteness of $\dot{V}_2$ that we want since

$$\dot{V}_2 = (c_1 - c_3 c_4)\xi \varphi(\xi) - c_4 \varphi(\xi)^2 - c_3 z_2^2 < 0$$

for all $z_1, z_2 \neq 0$, again using (8). The benefit of using the extra term $F(\xi)$ shows up in Equation (13). $F(\xi) \equiv 0$ leads to $c_2 = -(1+c_1)c_1 > 0$ and the severe restriction $c_1 < 0$ implying positive feedback from z_1.

Step 3: The final backstepping iteration begins with introducing the third residual

$$z_3 = q - q_{\text{des}} = q + c_3 z_2$$

We also update the system description

$$\begin{aligned}
\dot{z}_1 &= \varphi(\xi) \\
\dot{z}_2 &= z_3 - c_3 z_2 + c_1 \varphi(\xi) \\
\dot{z}_3 &= u + c_3(z_3 - c_3 z_2 + c_1 \varphi(\xi))
\end{aligned} \tag{14}$$

where

$$u = \frac{1}{I_y}(M(\alpha, q, \delta) + F_T Z_{TP}) \tag{15}$$

is regarded as the control variable. Also,

$$\dot{\xi} = -\varphi(\xi) + z_3 - c_3 z_2$$

V_3 is constructed by adding a term penalizing z_3.

$$V_3 = c_5 V_2 + \frac{1}{2} z_3^2, \quad c_5 > 0$$

[2] Strictly speaking, there exist limiting angles of attack, α_{min} and α_{max}, beyond which either Equation (8) stops to hold or where our simple aircraft model is no longer valid. Using the Lyapunov function to be constructed the true region of attraction could, at least theoretically, be computed.

We get

$$\begin{aligned}\dot{V}_3 = & c_5\big((c_1 - c_3c_4)\xi\varphi(\xi) - c_4\varphi(\xi)^2 - c_3z_2^2 \\ & \quad + z_3(z_2 + c_4\varphi(\xi))\big) \\ & + z_3(u + c_3(z_3 - c_3z_2 + c_1\varphi(\xi))) \\ \le & - c_4c_5\varphi^2(\xi) - c_3c_5z_2^2 \\ & + z_3(u + c_3z_3 + (c_5 - c_3^2)z_2 + (c_1c_3 + c_4c_5)\varphi(\xi))\end{aligned}$$

once again using (8). Select $c_5 = c_3^2$ to cancel the z_2z_3 cross-term and try yet another linear control law.

$$u = -c_6z_3, \quad c_6 > c_3 \tag{16}$$

is a natural candidate and with this we investigate the resulting clf time derivative.

$$\begin{aligned}\dot{V}_3 \le & - c_3^2c_4\varphi^2(\xi) - c_3^3z_2^2 - (c_6 - c_3)z_3^2 \\ & + (c_1c_3 + c_3^2c_4)z_3\varphi(\xi)\end{aligned}$$

The first three terms on the right hand side are all beneficial. In order to investigate the impact of the last cross-term, we complete the squares.

$$\begin{aligned}\dot{V}_3 \le & - c_3^3z_2^2 - (c_6 - c_3)(z_3 - \frac{c_1c_3 + c_3^2c_4}{2(c_6 - c_3)}\varphi(\xi))^2 \\ & - (c_3^2c_4 - \frac{(c_1c_3 + c_3^2c_4)^2}{4(c_6 - c_3)})\varphi^2(\xi)\end{aligned}$$

$\dot{V}_3$ is negative definite provided that the $\varphi^2(\xi)$ coefficient is negative, which is true for

$$c_6 > c_3(1 + \frac{(c_1 + c_3c_4)^2}{4c_3c_4}) \tag{17}$$

We now pick c_4 to minimize this lower limit under the constraints $c_4 > 0$ and $c_3c_4 > c_1$.

For $c_1 \le 0$, we can make $c_1 + c_3c_4$ arbitrarily small whereby Equation (17) reduces to

$$c_6 > c_3$$

For $c_1 > 0$ the optimal strategy can be shown to be selecting c_4 arbitrarily close to the bound c_1/c_3. This yields

$$c_6 > c_3(1 + c_1)$$

Summary: Let us summarize our results. The initial system (3)-(5) was transformed into (14) through a backstepping design. For the latter, the linear control law (16) was shown to be globally stabilizing despite the nonlinear nature of the system. In terms of the original state variables the control law becomes

$$u = -c_6(q + c_3(\theta + c_1(\gamma - \gamma_{\text{ref}}) - \gamma_{\text{ref}} - \alpha_0)) \tag{18}$$

c_1, c_3 and c_6 are user parameters restricted by

$$\begin{aligned} & c_1 > -1 \\ & c_3 > 0 \\ & c_6 > \begin{cases} c_3 & c_1 \le 0 \\ c_3(1 + c_1) & c_1 > 0 \end{cases}\end{aligned} \tag{19}$$

4.2 Tuning the controller

The resulting control law (18) is parameterized by the three parameters c_1, c_3, and c_6. It can be rewritten as

$$\begin{aligned} u &= -kx \\ k &= \begin{pmatrix} c_1c_3c_6 & c_3c_6 & c_6 \end{pmatrix} \\ x^T &= \begin{pmatrix} \gamma - \gamma_{\text{ref}} & \theta - \gamma_{\text{ref}} - \alpha_0 & q \end{pmatrix}\end{aligned} \tag{20}$$

Closed loop stability is ensured as long as the parameter restrictions above are satisfied. What is then a suitable choice?

A natural course of action is to linearize the open loop system (3)-(5) around a proper operating point, and determine a satisfactory linear control law, $u = -kx$. Then solve (20) for c_1, c_3, and c_6 and check if these are in agreement with the given restrictions. If so, the control law (18) locally achieves the desired linear closed loop behavior, and globally guarantees stability. If the restrictions are violated, the guarantee for global stability is lost.

This trial and error strategy for finding possible closed loop systems may seem unsatisfactory. However, the reverse task of mapping the parameter restrictions above onto restrictions regarding, e.g., closed loop pole placements is not trivial and will not be dealt with here.

4.3 Realizing the controller

The derived control law (18) regards the angular pitch acceleration, $\dot{q}$, as the control variable, see Equation (15). In this section we deal with the issue of finding control surface deflections, δ, that will produce the desired pitching moment, M.

Since exact knowledge of neither the pitching moment nor the thrust force is available, we will include an extra term to capture the residual. Thus we remodel Equation (5) as

$$\dot{q} = u = \frac{1}{I_y}(\bar{q}S\bar{c}C_m(\alpha, q, \delta) + F_TZ_{TP}) + E \tag{21}$$

where C_m and F_T represent the information available to us through an identified aircraft model. To capture a constant bias, we model E as

$$\dot{E} = 0 \tag{22}$$

We note that the nonlinear first part of (21) (excluding E) depends only on known quantities. This allows us to build an observer for (21)-(22) having linear error dynamics [3], thus producing an estimate of E with an exponentially decaying error.

With E at our disposal, we solve (21) for C_m.

$$C_m(\alpha, q, \delta) = \frac{I_y(u - E) - F_TZ_{TP}}{\bar{q}S\bar{c}} \tag{23}$$

The HIRM configuration offers two ways of controlling the pitching coefficient; via the taileron at the back of the aircraft, and via the canard wings. A number of ways of how to divide the work between these control surfaces have been suggested. In this paper, however, we put no effort into optimizing this aspect but simply ignore the canard wings, leaving everything to the taileron. In the following, δ will therefore represent the taileron's angle of deflection, see Figure 1.

The pitching coefficient, C_m, is usually a complicated function of the arguments involved. The standard way of modeling it is to collect measurements for a number of different situations and create a look-up table from which intermediate values can be interpolated. Assuming such a table is available to us, it can be used to solve (23) for δ. Given measurements of α and q, the left hand side typically becomes a close to monotonic function of δ. This fits many numerical solvers well, and the Van Wijngaarden-Dekker-Brent method [1, 2] was chosen for solving (23). This method is a happy marriage between bisection, providing sureness of convergence, and inverse quadratic interpolation, providing superlinear convergence in the best-case scenario.

Naturally, there are hard bounds on the possible taileron deflections. For the HIRM, $\delta \in [-40°, 10°]$ is the possible range. In cases where (23) cannot be satisfied, i.e., when enough pitching moment cannot be produced, we saturate and choose the proper δ bound. The effects of saturation have only been studied empirically.

5 Application

In this section we investigate the properties of the derived control law, applied to High Incidence Research Model (HIRM), a realistic model of a generic fighter aircraft. The model was originally released for the GARTEUR robust flight control challenge [8].

For the simulations, a flight case where the aircraft is in level flight at Mach 0.3 and at a height of 5000 ft was chosen. Actuator and sensor dynamics, which were not considered in the preceding design, were included in the simulations. Also, a constant pitch coefficient error of -0.030 was added to C_m. The simulation results are reproduced in Figure 3.

The top picture displays the flight path angle time response to a staircase reference signal. Also plotted is the time response of the desired linear closed loop system, i.e., the control law applied to the linearization of (3)-(5). We see that for small changes in the reference, the simulated response stays very close to the linear one. For the large step in the reference, occurring after 5 seconds, the simulated response has a longer rise

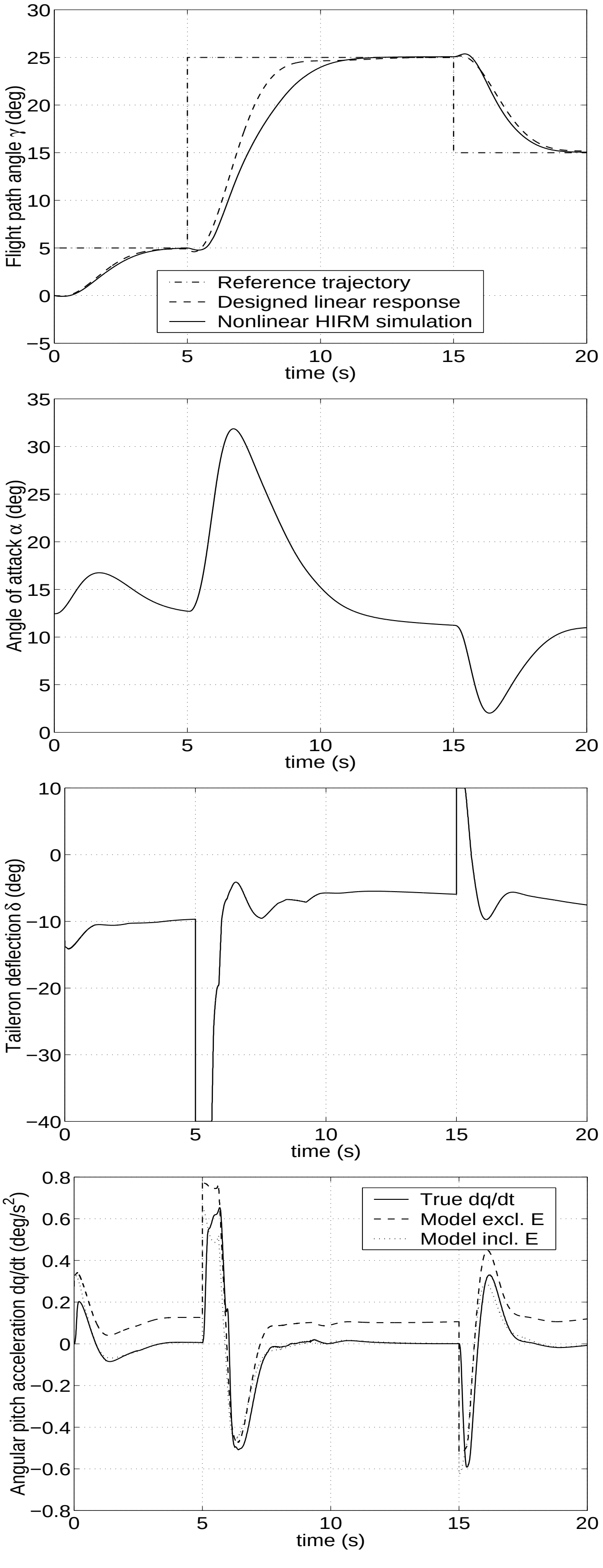

Figure 3: From top to bottom: γ time response compared with the designed linear response, α time response, taileron deflection, and finally the true $\dot{q}$ compared with Equation (21), with and without the model error estimate E.

time. This has two main causes. First, we see that the control surfaces saturate at the beginning of the step, thus not producing the desired pitching moment. This consequently slows down the step response. Second, the angle of attack becomes very large during the maneuver. When α reaches its peak after 7 seconds, the true lift force value is significantly smaller than what the linear model predicts, see Figure 2. In agreement with Equation (3), this also leads to slower γ dynamics.

In the bottom picture, the benefit of estimating the model error E can be seen. The dotted curve, representing the complete model of the angular pitch acceleration introduced in Equation (21), initially has a bias error compared to the true $\dot{q}$. This error is quickly estimated whereafter the curve closely follows the solid curve, representing the true $\dot{q}$. The latter was computed by differentiating q sensor data.

6 Backstepping vs feedback linearization

Having derived the globally stabilizing control law (18) for (3)-(5) using backstepping, it is rewarding to make a comparison with a feedback linearizing control design. Again using $z_1 = \gamma - \gamma_{\text{ref}}$, feedback linearization proceeds by defining

$$\begin{aligned}
\dot{z}_1 &= \varphi(\alpha - \alpha_0) = z_2 \\
\dot{z}_2 &= \varphi'(\alpha - \alpha_0)(q - z_2) = z_3 \\
\dot{z}_3 &= \varphi''(\alpha - \alpha_0)(q - z_2)^2 + \varphi'(\alpha - \alpha_0)(u - z_3) = \tilde{u}
\end{aligned}$$

Selecting $\tilde{u} = -\begin{pmatrix} k_1 & k_2 & k_3 \end{pmatrix}\begin{pmatrix} z_1 & z_2 & z_3 \end{pmatrix}^T$ clearly renders the closed loop system linear in the z coordinates. Solving for the angular acceleration u to be produced, defined as in (15), we get

$$u = z_3 + \frac{\tilde{u} - \varphi''(\alpha - \alpha_0)(q - z_2)^2}{\varphi'(\alpha - \alpha_0)} \tag{24}$$

Two things are worth noting about this expression. First, it depends not only φ, but also on its first and second derivatives. Recalling its definition from Equation (7), this means that the lift force L must be well known in order to accurately compute u. In the backstepping design, we relied on L only through its zero crossing and its sign switching property (8). Second, φ' is in the denominator of the expression above, implying that the control law has a singularity where $\varphi' = 0$. This occurs around the stall angle, where the lift force no longer increases with α, see Figure 2.

7 Conclusions

In this paper we have applied backstepping to flight path angle control. Regarding the pitching moment as the control variable, we have shown a linear control law to be globally stabilizing despite the nonlinear nature of the lift force with respect to the angle of attack. The fact that the control law is linear makes it easy to tune given a linearization of the open loop system. Using this the locally linear closed loop behavior can be selected.

Simulations using a realistic aircraft model including actuator and sensor dynamics, and a constant pitch coefficient error proved the control law to be robust against the approximations made during the design.

A comparison with feedback linearization showed the potential benefits of using backstepping. The backstepping control law is computationally much simpler, and is globally stabilizing while feedback linearization yields a singularity in the control law around the stall angle.

8 Acknowledgment

The authors would like to acknowledge the Defence Evaluation and Research Agency (DERA) of the United Kingdom for supplying the HIRM model.

References

[1] R.P. Brent. *Algorithms for Minimization Without Derivatives.* Prentice-Hall, 1973.

[2] G.E. Forsythe, M.A. Malcolm, and C.B. Moler. *Computer Methods for Mathematical Computations.* Prentice-Hall, 1976.

[3] A.J. Krener and A. Isidori. Linearization by output injection and nonlinear observers. *Systems & Control Letters*, 3:47–52, June 1983.

[4] M. Krstić, I. Kanellakopoulos, and P. Kokotović. *Nonlinear and Adaptive Control Design.* John Wiley & Sons, 1995.

[5] M. Krstić and P.V. Kokotović. Lean backstepping design for a jet engine compressor model. In *Proceedings of the 4th IEEE Conference on Control Applications*, pages 1047–1052, 1995.

[6] S.H. Lane and R.F. Stengel. Flight control design using non-linear inverse dynamics. *Automatica*, 24(4):471–483, 1988.

[7] T. Larsson. Linear quadratic design and μ-analysis of a fighter stability augmentation system. Master's thesis, Linköpings universitet, 1997.

[8] J-F. Magni, S. Bennani, and J. Terlouw, editors. *Robust Flight Control - A Design Challenge.* Springer, 1997.

[9] Jean-Jaques E. Slotine and Weiping Li. *Applied Nonlinear Contol.* Prentice Hall, 1991.

[10] B.L. Stevens and F.L. Lewis. *Aircraft Control and Simulation.* John Wiley & Sons, 1992.