

# Control of a Non-linear UAV Model in Longitudinal Motion

Allison D. Ryan

May 19, 2004

## Abstract

A simplified non-linear model of a fixed-wing UAV in longitudinal flight is analyzed, controlled, and observed using non-linear techniques. Its stability about a trim condition can be concluded using Lyapunov's linearization method, although it is not stable in the case of zero inputs. After checking the accessibility of the system, a sliding mode controller is designed to control forward airspeed and pitch angle. Because direct sliding mode altitude control results in unstable internal dynamics, a PI altitude control is used to select the desired pitch angle for the sliding mode controller. This results in accurate altitude tracking, with the robustness properties of the sliding mode control. Next, a Lyapunov-type observer is implemented. The observer is shown to have locally stable error dynamics, and is used in the feedback loop with the sliding mode controller.

## Contents

<b>1</b>	<b>Model Development</b>	<b>2</b>
<b>2</b>	<b>Open Loop Simulations</b>	<b>3</b>
<b>3</b>	<b>Stability Analysis</b>	<b>6</b>
<b>4</b>	<b>Accessibility Analysis</b>	<b>6</b>
<b>5</b>	<b>Sliding Mode Controller Design</b>	<b>7</b>
5.1	Internal dynamics for $z = [u \ y]$ . . . . .	7
5.2	Sliding mode control design for $z = [u \ \theta]$ . . . . .	8
5.3	Discrete sliding control . . . . .	9
5.4	Smooth robust sliding control . . . . .	12
5.5	PI control of altitude . . . . .	14
<b>6</b>	<b>Lyapunov-based observer design</b>	<b>16</b>
6.1	Observability requirements . . . . .	16
6.2	Observer design based on Thau's theorem . . . . .	17
<b>7</b>	<b>Conclusions</b>	<b>20</b>
<b>8</b>	<b>Appendix: UAV and controller parameters</b>	<b>21</b>

# 1 Model Development

The model used for control design and simulation is a non-linear fixed wing aircraft in longitudinal motion. As developed in any aerodynamics textbook, an aircraft's orientation in space can be described by the 3-2-1 Euler angle transformation, where the Euler angles are  $\psi$ ,  $\theta$ , and  $\phi$ , which represent yaw, pitch, and roll, respectively. The aircraft's velocity is represented as components parallel to each of the Euler axes, such that

$$V = u + v + w \quad (1)$$

where  $u$  is parallel to the roll axis,  $v$  is parallel to the pitch axis, and  $w$  is parallel to the yaw axis.

The assumption of longitudinal motion leads to the following simplifications

$$\begin{aligned} \psi &= 0 \\ \phi &= 0 \\ v &= 0 \end{aligned} \quad (2)$$

The system can now be described by four state variables: the velocities  $u$  and  $w$ , the pitch angle  $\theta$ , and its derivative  $q$ . State equations are derived from a balance of linear and angular momentum. The following simplifications are made in order to require fewer physical parameters specific to the aircraft, and are reasonable for a UAV such as the SigRascal, which is not a high-performance aircraft.

1. Thrust force is aligned with first velocity component ( $u$ )
2. Lift forces (from elevator and wing) are aligned with third velocity component ( $w$ )
3. Drag force is opposed to thrust force

These simplifications result in state equations of the following form

$$\begin{aligned} m\dot{u} &= u_T - Drag - mgsin(\theta) \\ m(\dot{w} - uq) &= -Lift + ELift + mgcos(\theta) \\ \dot{\theta} &= q \\ J\dot{q} &= -x_L * Lift + x_E * ELift \end{aligned} \quad (3)$$

$$\begin{aligned} u_T &= \text{thrust force (system input)} \\ m &= \text{aircraft mass} \\ g &= \text{accelaration due to gravity} \\ J &= \text{aircraft moment about pitching axis} \\ Lift &= \text{Lift force due to wings} \\ ELift &= \text{Lift force due to elevators} \\ x_L &= \text{moment arm from Lift to center of mass (CM)} \\ x_E &= \text{moment arm from ELift to CM} \end{aligned}$$

The aerodynamic forces (lifts and drag) are related to aircraft configuration, flight conditions, and control surface positions and are highly nonlinear. However, for a standard fixed-wing small aircraft in non-aggressive flight maneuvers, they will be approximated by their dependence on the state variables and inputs listed above. For

more rigorous simulation, the constant aerodynamic coefficients used here could be replaced by the appropriate stability derivatives[1]. Henceforth, aerodynamic forces will be approximated as follows:

$$\begin{aligned} Drag &= C_D u^2 \\ Lift &= C_L \alpha u^2 \\ Elift &= C_E (u_E - \alpha) u^2 \end{aligned} \quad (4)$$

$$\begin{aligned} u_E &= \text{elevator deflection (system input)} \\ C_D &= \text{drag coefficient} \\ C_L &= \text{wing lift coefficient} \\ C_E &= \text{tail lift coefficient} \\ \alpha &= \text{angle of attack} \end{aligned} \quad (5)$$

Over the range of angle of attack likely to be experienced by a SigRascal in autopiloted flight, the approximation

$$\tan(\alpha) = \alpha = \frac{w}{u} \quad (6)$$

is valid, resulting in the following state equations from the combination of all preceding equations.

$$\begin{aligned} m\dot{u} &= u_T - C_D u^2 - mg \sin(\theta) \\ m(\dot{w} - uq) &= -C_L u w + C_E (u_E - \frac{w}{u}) u^2 + mg \cos(\theta) \\ \dot{\theta} &= q \\ J\dot{q} &= -x_L C_L u w + x_E C_E (u_E - \frac{w}{u}) u^2 \end{aligned} \quad (7)$$

## 2 Open Loop Simulations

The open loop system was simulated with fixed control inputs in order to examine its stability about trim condition. With the control inputs fixed at their trim values, the UAV remains in trim if it begins at the equilibrium condition. Plots are not included, because all parameters remain constant.

With initial conditions displaced slightly from equilibrium (with control inputs fixed at trim) the system goes into damped oscillations and returns to the equilibrium point, as shown below for displacements of  $u$  and  $\theta$ . The system is also self-stabilizing in response to changes in the inputs from their trim values.

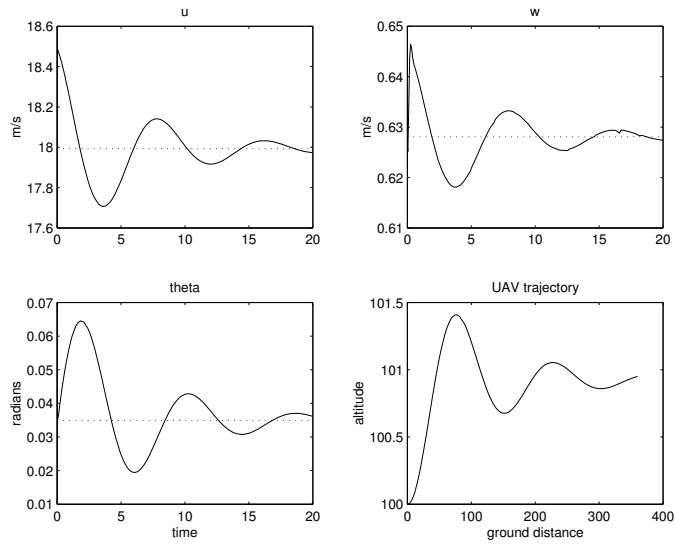


Figure 1: Open loop response with  $u$  initially displaced from equilibrium value

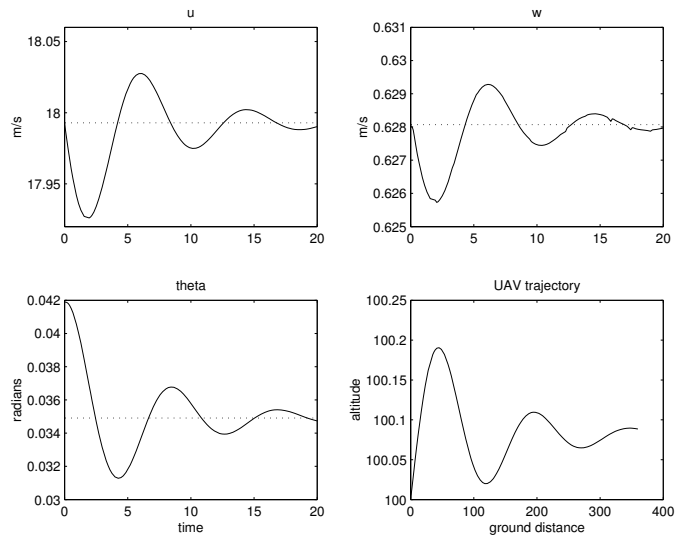


Figure 2: Open loop response with  $\theta$  initially displaced from equilibrium value

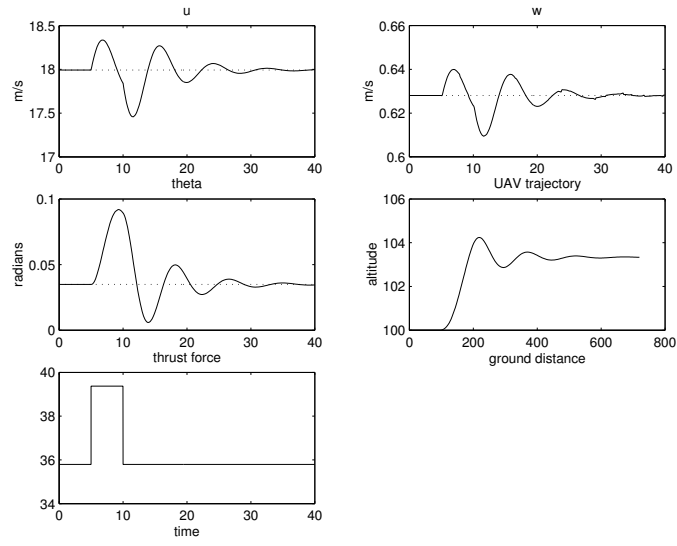


Figure 3: Open loop response with thrust temporarily displaced from equilibrium value

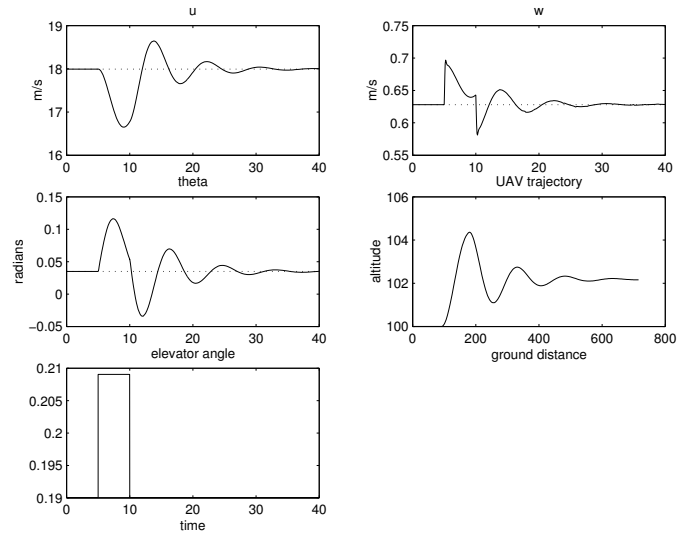


Figure 4: Open loop response with elevator deflection temporarily displaced from equilibrium value

### 3 Stability Analysis

Using Lyapunov's first theorem, the stability of an equilibrium point of a non-linear system can be inferred from its linearization about that equilibrium point. This is contingent on the condition that the non-linear terms do not grow unboundedly when approaching the equilibrium point. This system has no such terms, so Lyapunov's first theorem applies. For a system of the form  $\dot{x} = f(x) + g(x) * u$ , stability analysis is usually performed assuming that  $u = 0$ . Unfortunately, the UAV model has no equilibrium point in this case, because when unpowered, it simply falls.

A more interesting analysis can be performed by selecting a constant value of  $u_{ep} = [u_{T0} \ u_{E0}]$  that will result in a trim condition or equilibrium point  $x_{ep} = [u_0 \ w_0 \ \theta_0 \ q_0]$ . Now the system can be linearized about the equilibrium point as

$$\dot{x} \approx \left( \frac{\partial f}{\partial x} \Big|_{x_{ep}} + \frac{\partial g}{\partial x} \Big|_{x_{ep}} * u_{ep} \right) * \Delta x = A \Delta x \quad (8)$$

and the equilibrium point's stability can be inferred from the eigenvalues of the A matrix.

$$A = \begin{bmatrix} -2C_D u_0 / m & 0 & -g \cos(\theta_0) & 0 \\ \frac{-w_0}{m} (C_L + C_E) + \frac{2C_E u_{E0} u_0}{m} & \frac{-u_0}{m} (C_L + C_E) & -g \sin(\theta_0) & u_0 \\ 0 & 0 & 0 & 1 \\ \frac{-C_L x_L w_0}{J} + \frac{C_E x_E}{J} (2u_{E0} u_0 - w_0) & \frac{-u_0}{J} (C_L x_L + C_E x_E) & 0 & 0 \end{bmatrix} \quad (9)$$

For trim conditions similar to a SigRascal in level flight, the eigenvalues of the A matrix are stable. For example, for a trim condition of level flight,  $u = 18 \frac{m}{s}$ , and  $\alpha = 3^\circ$ , the eigenvalues of A are  $-19, -1, -0.2 \pm 0.5i$ . These concur with the open loop simulations, which show the UAV stable but oscillatory.

### 4 Accessibility Analysis

Generally, if a system's linearization about an equilibrium point is controllable, the non-linear system will be accessible about that point, so the observability of the linearization will be examined first.

As was done for stability analysis, the system is linearized about the trim condition  $(x_{ep}, u_{ep})$ , resulting in the following.

$$\begin{aligned} \dot{x} &\approx \left( \frac{\partial f}{\partial x} \Big|_{x_{ep}} + \frac{\partial g}{\partial x} \Big|_{x_{ep}} * u_{ep} \right) * \Delta x + g(x_{ep}) \Delta u \\ &= A \Delta x + B \Delta u \end{aligned} \quad (10)$$

A from previous equation

$$B = \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & \frac{C_E u_0^2}{m} \\ 0 & 0 \\ 0 & \frac{C_E x_E u_0^2}{J} \end{bmatrix}$$

A linear system is controllable if its controllability matrix  $[B \ AB \ \dots \ A^N B]$  is full rank. When the equilibrium values of  $x_{ep}$  and  $u_{ep}$  are substituted into the matrices A and

B, the controllability matrix is full rank, so the linearized system is controllable. This suggests that the non-linear system will be accessible.

A non-linear system is accessible at an equilibrium point if its accessibility matrix is full rank when evaluated at that point. The accessibility matrix of this system, where  $g = [g_1 \ g_2]$  is  $C = [g_1 \ g_2 \ [g_1, g_2] \ [f, g_1] \ [f, g_2]]$ .

For this system,  $g_1$  and  $g_2$  are the columns of matrix B, and the other terms of the controllability matrix are shown below. It is full rank at the trim condition investigated, and will be in general when  $u$ ,  $w$ , and  $\theta$  are characteristic of low-performance fixed-wing flight. Thus, the UAV can initiate motion in any direction from trim condition, and it is feasible to design a controller.

$$[g_1, g_2] = \begin{bmatrix} 0 \\ \frac{2uC_E}{m^2} \\ 0 \\ \frac{2uC_E x_E}{mJ} \end{bmatrix} \quad (11)$$

$$[f, g_1] = \begin{bmatrix} \frac{2C_D u}{m^2} \\ \frac{w}{m^2}(C_E + C_L) + \frac{q}{m} \\ 0 \\ \frac{w}{mJ}(C_E x_E + C_L x_L) \end{bmatrix} \quad (12)$$

$$[f, g_2] = \begin{bmatrix} 0 \\ \frac{-u^3 C_E}{m^2}(C_E + C_L) + \frac{C_E x_E u^3}{J} \\ \frac{C_E x_E u^2}{J} \\ \frac{-C_E u^3}{mJ}(C_E x_E + C_L x_L) \end{bmatrix} \quad (13)$$

## 5 Sliding Mode Controller Design

A MIMO sliding mode approach to control is appropriate for the UAV system because both modeling uncertainties and disturbances will be encountered. The approximations of the aerodynamic forces given in section 1 will lead to unmodeled dynamics, and wind gusts cause disturbance forces. Because sliding mode control is a form of input/output linearization, certain choices for the output  $z$  can result in unstable internal dynamics in the controlled system. The original choice of  $z = [u \ y]$  would allow direct altitude control (which is desired), but resulted in unstable  $\theta$  dynamics. Instead,  $z = [u \ \theta]$  was selected.

### 5.1 Internal dynamics for $z = [u \ y]$

Two sliding surfaces are defined for the outputs  $u$  and  $y$  (altitude), where the control objective will be to drive these surfaces to zero using a Lyapunov formulation to guarantee convergence. A sliding mode control is chosen so that

$$\begin{aligned} \begin{bmatrix} \dot{s}_1 \\ \dot{s}_2 \end{bmatrix} &= \begin{bmatrix} L_1(x) \\ L_2(x) \end{bmatrix} + \begin{bmatrix} \frac{1}{m} & 0 \\ \frac{1}{m} \sin(\theta) & \frac{-C_E u^2}{m} \cos(\theta) \end{bmatrix} \begin{bmatrix} u_T \\ u_E \end{bmatrix} \\ &= \begin{bmatrix} -\tilde{f}_1(s_1) \\ -\tilde{f}_2(s_2) \end{bmatrix} \end{aligned} \quad (14)$$

where the sliding surfaces are the following.

$$\begin{aligned} s_1 &= u - u_d \\ s_2 &= \dot{y} - \dot{y}_d + \beta(y - y_d) \end{aligned} \quad (15)$$

It can be seen from the system state equations that the internal dynamics of the system are

$$\begin{aligned} \dot{\theta} &= q \\ \dot{q} &= \frac{-uw}{J}(C_{ExE} + C_{LxL}) + \frac{C_{ExE}u^2}{J}u_E \end{aligned} \quad (16)$$

because  $u$  is controlled directly, and  $w$  is closely coupled with  $\dot{y}$ . When the system is being controlled,  $u_E$  will be given by the equations for  $\dot{s}$ ,  $\tilde{f}(s) = 0$ , and  $w$  can be calculated by using the fact that  $\dot{y} = \dot{y}_d$ .

$$w = \frac{1}{\cos(\theta)}(u \sin(\theta) - \dot{y}_d) \quad (17)$$

Plugging the control law and the substitution into the internal dynamics results in the following dynamic for  $\theta$ , where  $B_i$  are all bounded and positive when the controller is active.

$$\ddot{\theta} - B_1(\cos(\theta) - \dot{u})\dot{\theta} + B_2 \tan(\theta) - \frac{B_3}{\cos(\theta)} + B_4 = 0 \quad (18)$$

As  $\theta$  increases toward  $\frac{\pi}{2}$ , the theta dynamic approaches

$$\ddot{\theta} - C_1\dot{\theta} + \infty = 0 \quad (19)$$

and the internal dynamics are unstable. Therefore, a sliding mode controller should not be designed for the output  $z = [u \ y]$ .

## 5.2 Sliding mode control design for $z = [u \ \theta]$

By choosing  $\theta$  as one of the controlled outputs, the problem presented in the previous section is avoided. Thus, a sliding mode control is designed for  $z = [u \ \theta]$  by defining the following sliding surfaces.

$$\begin{aligned} s_1 &= u - u_d \\ s_2 &= \dot{\theta} - \dot{\theta}_d + \beta(\theta - \theta_d) \end{aligned} \quad (20)$$

The control  $[u_T \ u_E]^T$  will be chosen so that  $\dot{s}_i = -\tilde{f}_i(s)$ , where  $\tilde{f}_i$  will be chosen to guarantee the sliding condition. With  $\dot{s}$  of the form

$$\begin{bmatrix} \dot{s}_1 \\ \dot{s}_2 \end{bmatrix} = \begin{bmatrix} L_1(x) \\ L_2(x) \end{bmatrix} + \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & \frac{-C_{ExE}u^2}{J} \end{bmatrix} \begin{bmatrix} u_T \\ u_E \end{bmatrix} \quad (21)$$

the control is given by

$$\begin{aligned} \begin{bmatrix} u_T \\ u_E \end{bmatrix} &= \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & \frac{-C_{ExE}u^2}{J} \end{bmatrix}^{-1} * \begin{bmatrix} -\tilde{f}_1(s_1) - L_1 \\ -\tilde{f}_2(s_2) - L_2 \end{bmatrix} \\ &= \begin{bmatrix} m(-\tilde{f}_1(s_1) + \frac{C_D u^2}{m} + g \sin(\theta) + \dot{u}_d) \\ \frac{J}{C_{ExE}u^2}(-\tilde{f}_2(s_2) + (C_{LxL} + C_{ExE})\frac{uw}{J} + \ddot{\theta}_d - \beta(q - \dot{\theta}_d)) \end{bmatrix} \end{aligned} \quad (22)$$

The effect of this control is to replace the non-linear dynamics of  $\dot{s}$  with the desired dynamics  $\tilde{f}$ .

Sliding mode control is based on a Lyapunov analysis with the Lyapunov function

$$V = \frac{1}{2}s^2 \quad (23)$$

which will have a negative definite derivative when  $\tilde{f}$  is chosen to satisfy the sliding condition on  $s$ . The sliding condition,

$$s\dot{s} \leq 0 \quad (24)$$

can be satisfied by a number of different forms for  $\tilde{f}$ . When  $\dot{V}$  is negative definite,  $V$  is a Lyapunov function for  $s$ , and by Lyapunov's second method, the  $s$  dynamics are globally asymptotically stable. This implies that  $s_1$  and  $s_2$  will decay to zero at a rate determined by  $\tilde{f}$ . Because the sliding surfaces were defined as the error between states and their desired values, the tracking error will be zero when  $s$  converges to zero.

### 5.3 Discrete sliding control

A basic form of sliding mode control defines  $\tilde{f}(s) = -\eta \text{sgn}(s)$  and results in  $s$  going to zero very quickly. In the absence of modeling error and disturbances, this results in a very aggressive control. When the sliding surface  $s = 0$  is reached, chattering will occur, where the system vibrates with high frequency and low amplitude about the sliding surface. This can be seen in simulation in the following figures. Also, robustness terms have not yet been included, so this control fails in the presence of modeling error as shown in figures 7 and 8.

Chattering can be very problematic in many applications, especially if it excites unmodeled vibration modes. To remedy this and to add robustness to disturbance or modeling error, another variation of sliding control is shown in the next section. Simulation results of the discrete sliding mode controller are shown below.

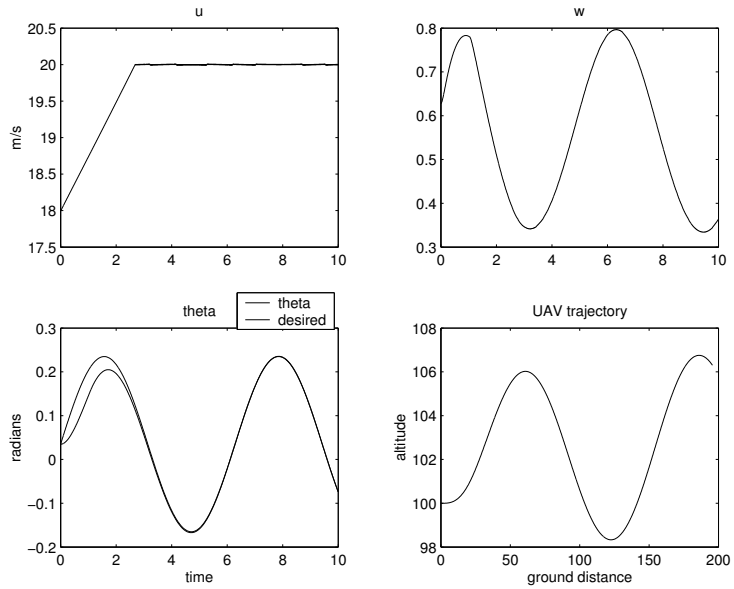


Figure 5: UAV states with discrete sliding mode control tracking  $\theta$  and  $u$  with no modeling error or disturbance

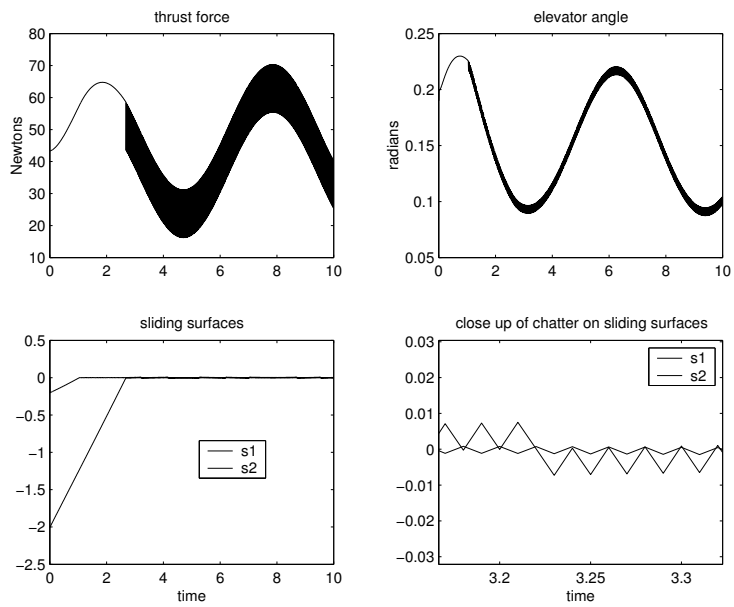


Figure 6: Actuation and sliding surfaces for discrete sliding control, no modeling error or disturbance

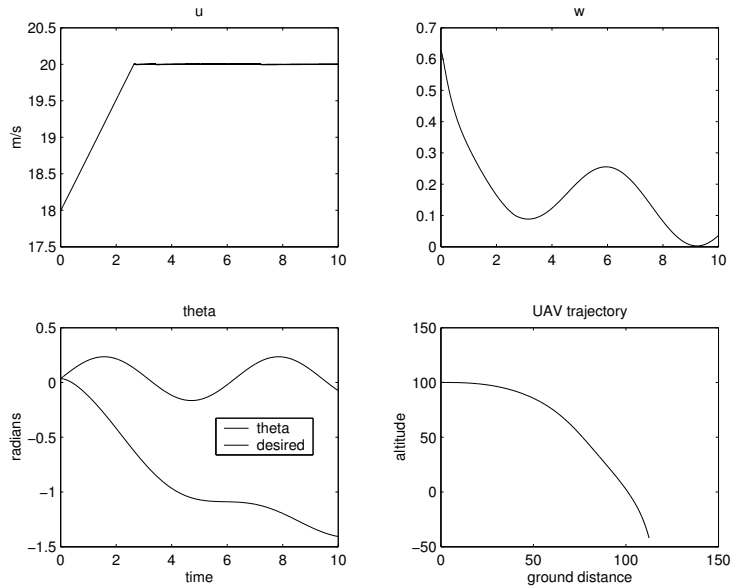


Figure 7: UAV state with discrete sliding mode control tracking  $\theta$  and  $u$ . 15 % modeling error in  $C_E$ . Note that controller has failed to track  $\theta$ .

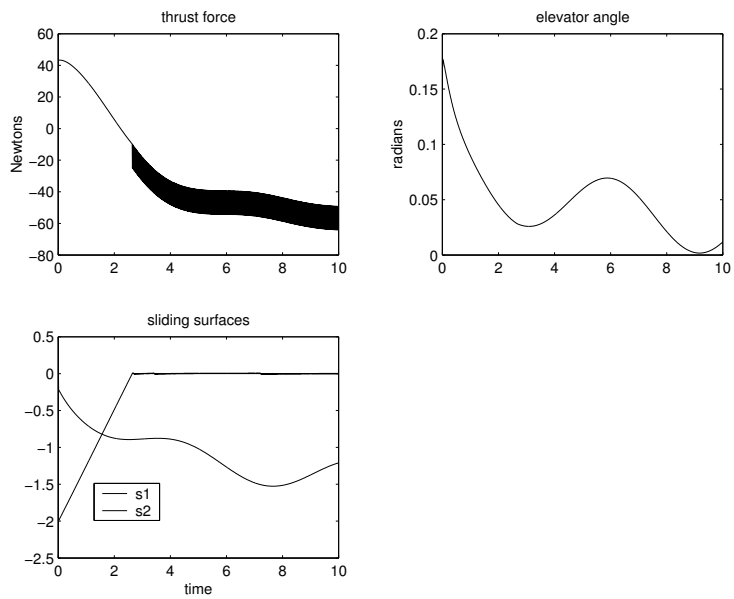


Figure 8: Actuation and sliding surfaces for discrete sliding control with modeling error. Note that  $s_2$  does not converge to zero.

## 5.4 Smooth robust sliding control

The two problems associated with the discrete control presented in the previous section are chatter and lack of robustness to parameter error and disturbance. The chatter problem will be solved by using a smooth rather than discrete function  $\tilde{f}$ . As a result,  $s$  will converge to within a boundary layer on either side of the sliding surface, rather than converging to zero. The robustness problem will be resolved by adding an extra term into  $\tilde{f}$  to overcome modeling error and disturbances. This requires that the parameter errors and disturbance have bounded magnitude, where the bounds will determine the size of the robustness term in the control, shown below.

$$\tilde{f}_i(s_i) = (K_i + \Delta_i) \text{sat}\left(\frac{s_i}{\Phi_i}\right) \quad (25)$$

$K_i$  determines the rate of convergence of  $s_i$ ,  $\Delta_i$  is the robustness term, and  $\Phi_i$  is the width of the boundary layer to which  $s_i$  will converge.  $K_i$  and  $\Phi_i$  can be tuned based on desired performance, but  $\Delta_i$  must be calculated to ensure that it overcomes the uncertainty and disturbance. In an aircraft model, the aerodynamic parameters can be the most difficult to calculate, so it will be assumed that the true parameters  $C_D$ ,  $C_L$  and  $C_E$  differ from their assumed values ( $\hat{C}_D$ ,  $\hat{C}_L$  and  $\hat{C}_E$ ) by the parameter estimate errors  $\tilde{C}_D$ ,  $\tilde{C}_L$  and  $\tilde{C}_E$ . As a result,  $\dot{s}$  will include terms other than  $\tilde{f}$ . For example,

$$\dot{s}_1 = -\tilde{f}_1(s_1) - \tilde{C}_D \frac{u^2}{m} \quad (26)$$

$\Delta_i$  must be calculated to ensure  $\dot{s}_i$  is always negative, based on the bounds of the parameter uncertainties. For the simulation shown, it was assumed that the parameter estimate errors were less than fifteen percent, and that there was no disturbance. As a result,  $\Delta_1 = 0.8$  and  $\Delta_2 = 2$ .

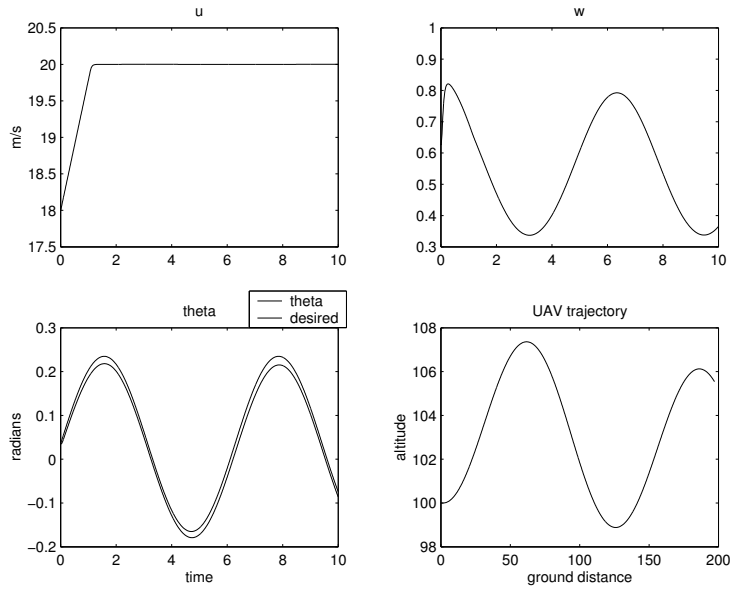


Figure 9: UAV state with smooth sliding mode control tracking  $\theta$  and  $u$  15 % modeling error in  $C_E$

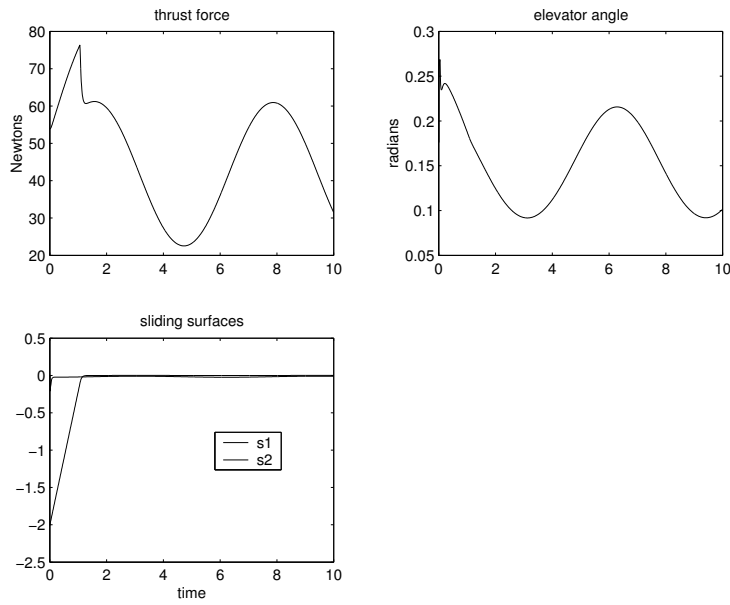


Figure 10: Actuation and sliding surfaces for smoothed robust sliding mode control

## 5.5 PI control of altitude

Although direct sliding mode altitude control was shown to lead to unstable internal dynamics, one would still like to control the aircraft's altitude rather than pitch angle. One possibility would be to calculate in advance a desired trajectory for  $\theta$  based on a desired trajectory for altitude. However, this would be computationally difficult, and not suitable for real time control.

Instead, the combination of UAV and sliding mode controller is treated as a plant inside a PI loop, as shown in the schematic below. The PI controller outputs  $[\theta_d \ q_d \ \dot{q}_d]$ , which are the required inputs for sliding mode control of  $\theta$ . The PI control follows the standard formulation,

$$\theta_d(t) = K_i \int_0^t (y_d(\tau) - y(\tau)) d\tau + K_p (y_d(t) - y(t)) \quad (27)$$

which is passed through saturations to ensure that an unreasonable climb rate or pitch angle cannot be commanded. Altitude tracking with the combination PI and SMC control is shown on the following page.

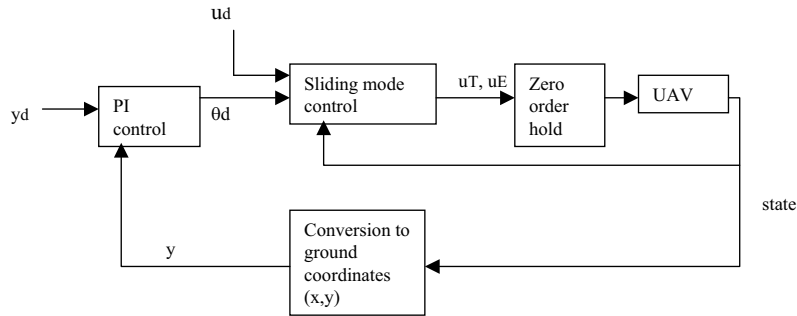


Figure 11: Schematic drawing of sliding mode controller and UAV inside PI loop

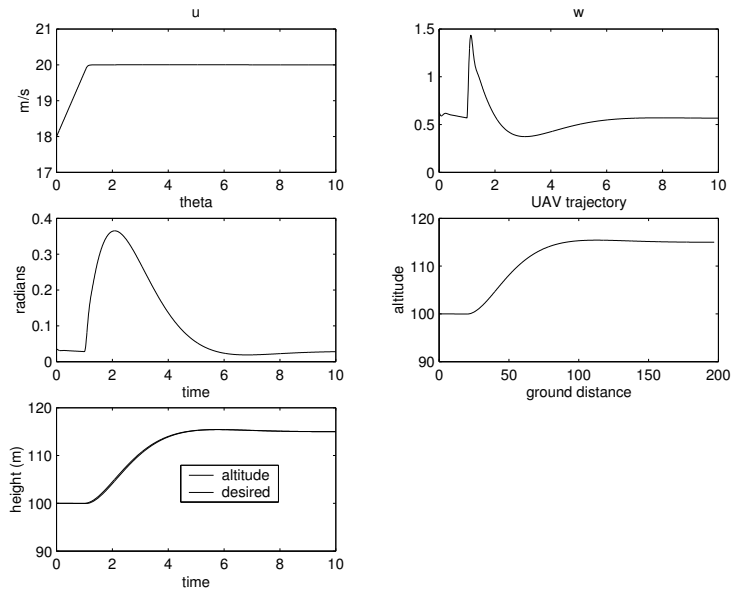


Figure 12: Tracking of filtered altitude step using PI loop combined with sliding mode control. Sliding mode control is smoothed and has modeling error as before.

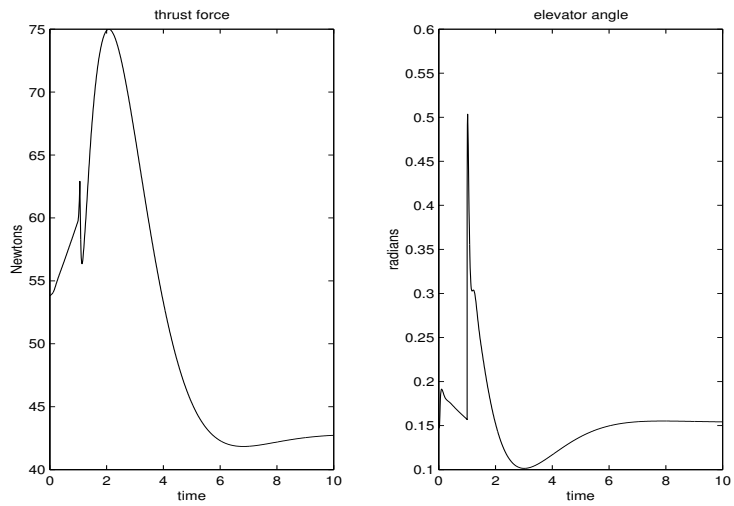


Figure 13: Actuation and sliding surfaces for PI/SMC altitude control

## 6 Lyapunov-based observer design

The sliding mode controllers described in the previous chapter all assume full knowledge of the aircraft state. In reality, knowledge of the aircraft state comes from limited and imperfect sensors. The forward airspeed  $u$  is calculated from the dynamic pressure using a pitot sensor, and the pitch rate  $q$  comes from a gyroscope. However, integrating  $q$  to obtain  $\theta$  is usually unwise due to sensor inaccuracies, so  $\theta$  and  $w$  must be estimated. This will be done using a Lyapunov-based observer, which has the following state estimation dynamics.

$$\dot{\hat{x}} = A\hat{x} + g(u, z) + f(\hat{x}) + L(z - M\hat{x}) \quad (28)$$

$$\begin{aligned} A &= \text{Linear terms from state equations} \\ M\hat{x} &= \text{estimation of outputs} \\ z &= \text{outputs of sensors} \\ g(u, z) &= \text{known nonlinear terms} \\ f(\hat{x}) &= \text{nonlinear function of estimated states} \\ L &= \text{observer gain matrix} \end{aligned}$$

resulting in the estimation error dynamics shown below. It is required that the matrices  $A$  and  $M$  satisfy the linear conditions for observability, and that  $f(x)$  be globally Lipschitz (or locally Lipschitz for local convergence).

$$\dot{\tilde{x}} = (A - LM)\tilde{x} + f(x) - f(\hat{x}) \quad (29)$$

If a Lyapunov function of  $\tilde{x}$  can be found, then the state estimates must converge to their true values. Thau's theorem proves that

$$\dot{V} = \tilde{x}^T P \tilde{x} \quad (30)$$

is a Lyapunov function for  $\tilde{x}$  when  $P$  and  $L$  are chosen according to certain conditions to be detailed in the section on observer design.

### 6.1 Observability requirements

Before an observer can be designed, the system must be written in a form

$$\begin{aligned} \dot{x} &= Ax + f(x) + g(u, z) \\ z &= Mx \end{aligned} \quad (31)$$

such that the set of  $A$  and  $M$  is observable and  $f(x)$  is at least locally Lipschitz. The UAV state equations only have one linear term, so the matrix  $A$  would be almost empty and the observability requirement could not be met. To remedy this, the original state equations  $\dot{x} = f(x) + g(x) * u$  are expanded below in a two-term Taylor series about the trim state and input.

$$\dot{x} \approx \left. \frac{\partial f}{\partial x} \right|_{ep} * \Delta x + \left. \frac{\partial g}{\partial x} \right|_{ep} * \Delta x * u_{ep} + \frac{1}{2} \left. \frac{\partial^2 x}{\partial x^2} \right|_{ep} * (\Delta x)^2 + g(x_{ep}) * \Delta u \quad (32)$$

Letting  $A = \left. \frac{\partial f}{\partial x} \right|_{ep} + \left. \frac{\partial g}{\partial x} \right|_{ep} * u_{ep}$  and writing out  $\frac{1}{2} \left. \frac{\partial^2 x}{\partial x^2} \right|_{ep}$ , the (approximated) state equations are as follows.

$$\dot{x} \approx A\Delta x + \begin{bmatrix} -C_D(\Delta u)^2/m + \frac{g}{2}\sin(\theta_0)(\Delta\theta)^2 \\ \frac{-g}{2}\cos(\theta_0)(\Delta\theta)^2 \\ 0 \\ 0 \end{bmatrix} + g(x_{ep}) * \Delta u \quad (33)$$

Now the matrices A and M form a full-rank observability matrix. This is almost the desired form, but the term  $C_D(\Delta u)^2/m$  will not satisfy the Lipschitz condition. However, because u is one of the sensor outputs, this term is entirely known, and can be included in the term  $g(u,z)$  in the desired state equation form. Thus, the system can be written as  $\dot{x} = Ax + f_1(x) + g_1(u, z)$  as desired, where

$$A \quad \text{is given in stability analysis section}$$

$$f_1 = \begin{bmatrix} \frac{g}{2} \sin(\theta_0) (\Delta\theta)^2 \\ \frac{-g}{2} \cos(\theta_0) (\Delta\theta)^2 \\ 0 \\ 0 \end{bmatrix}$$

$$g_1 = \begin{bmatrix} \frac{-C_D(\Delta u)^2}{m} + \frac{u_T}{m} \\ \frac{C_E u_0^2}{m} u_E \\ 0 \\ \frac{C_E x_E u_0^2}{J} u_E \end{bmatrix}$$

Now the Lipschitz condition must be checked for  $f_1(x)$  and a Lipschitz constant  $\gamma$  must be found. Because the UAV was linearized about a trim condition,  $\theta_0$  will be small, so the small angle approximations can be used for the  $\sin(\theta_0)$  and  $\cos(\theta_0)$  terms in  $f_1$ , with the following result.

$$\begin{aligned} \|f_1(x) - f_1(\hat{x})\| &\approx \frac{g}{2} \left\| \begin{bmatrix} \theta_0((\Delta\theta)^2 - (\Delta\hat{\theta})^2) \\ -((\Delta\theta)^2 - (\Delta\hat{\theta})^2) \end{bmatrix} \right\| \\ &= \frac{g}{2} \|(\Delta\theta)^2 - (\Delta\hat{\theta})^2\| \left\| \begin{bmatrix} \theta_0 \\ -1 \end{bmatrix} \right\| \end{aligned} \quad (34)$$

Again assuming that the trim value  $\theta_0$  is small, the magnitude of the vector above is approximately 1, and the Lipschitz condition can be written as follows.

$$\begin{aligned} \gamma \|x - \hat{x}\| &\leq \|f_1(x) - f_1(\hat{x})\| \\ &\approx \frac{g}{2} \|\theta + \hat{\theta} - 2\theta_0\| \|\theta - \hat{\theta}\| \end{aligned} \quad (35)$$

Since  $\|x - \hat{x}\| \geq \|\theta - \hat{\theta}\|$ , this shows that  $\|f_1(x) - f_1(\hat{x})\| \leq \frac{g}{2} \|\theta + \hat{\theta} - 2\theta_0\| \|x - \hat{x}\|$ . Because  $\theta$  will be small for controlled UAV flight, a Lipschitz constant  $\gamma = 10$  will satisfy the Lipschitz condition.

## 6.2 Observer design based on Thau's theorem

Thau's theorem states that  $V = \tilde{x}^T P \tilde{x}$  is a Lyapunov function for the observer error dynamics under the following conditions.

- P is a solution to the Lyapunov equation  $(A - LM)^T P + P(A - LM) = -Q$
- Lipschitz constant  $\gamma < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)}$

It can be shown that the ratio of eigenvalues above is maximized when  $Q = I$ , so Q will be set equal to I, and Raghavan's iterative approach will be used to select L.

Briefly, Raghavan's iterative approach uses the following algebraic Ricatti equation, which is derived from the derivative of the Lyapunov function candidate,  $\dot{V}$ .

$$AP + PA^T + P(\gamma^2 I - \frac{1}{\epsilon} M^T M)P + (1 + \epsilon)I = 0 \quad (36)$$

The iterative process is to pick a positive  $\epsilon$  and then solve for P. If P is not positive definite, epsilon is decreased. When P is symmetric and positive definite, L is as follows.

$$L = \frac{PM^T}{2\epsilon} \quad (37)$$

By this method, an observer was designed such that  $\gamma = 10$  and  $\epsilon = 0.01$ . The first set of figures shows the observer tracking the actual states, but not connected in the feedback loop. The initial state estimates are zero, except for the forward airspeed, which is initially estimated at 15 *m/s*. All estimates converge to the true parameters in less than a half second.

The second set of figures shows the same observer connected in the feedback loop, along with the PI altitude control and the sliding mode control. The system is tracking the same altitude step as in previous sections, with the same initial parameter estimates given above.

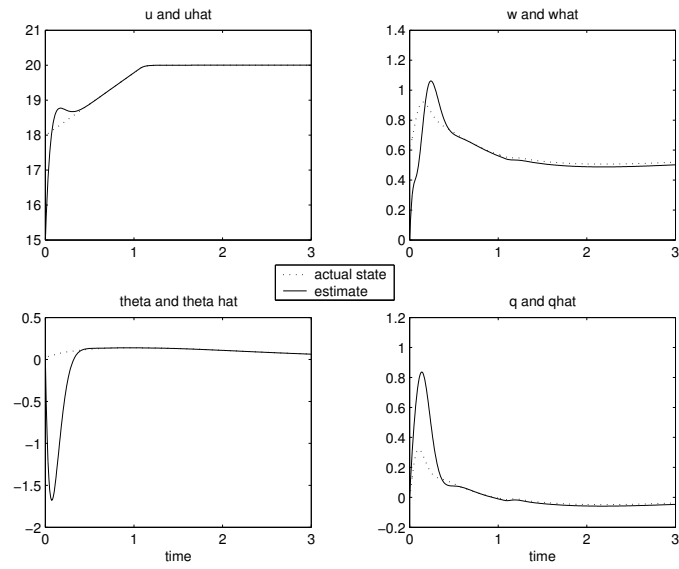


Figure 14: Parameter estimation convergence using Lyapunov-type observer (note decreased time scale)

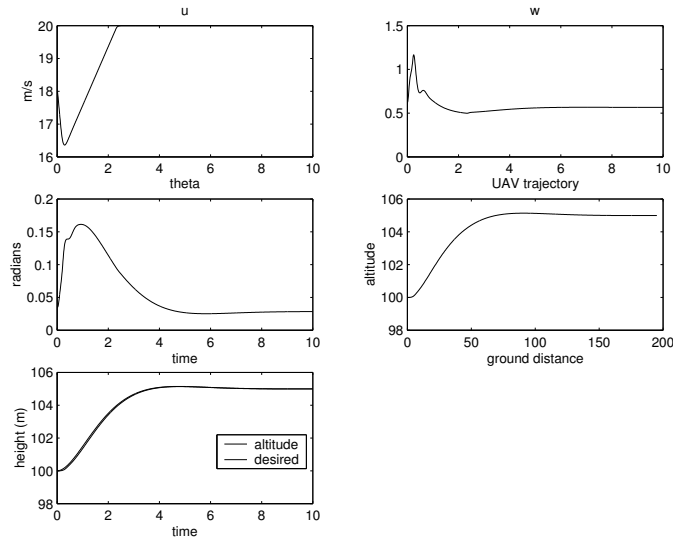


Figure 15: UAV state and altitude tracking with observer in control loop

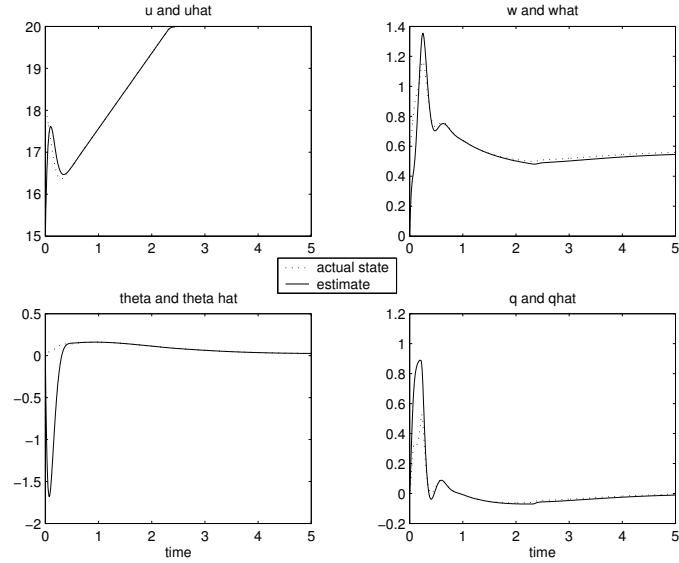


Figure 16: Parameter estimation convergence with Lyapunov-type observer in control loop (note decreased time scale)

## 7 Conclusions

The fixed wing UAV model is a good candidate for non-linear control due to its significant non-linearities and its lack of a true equilibrium point when linearized with no control input. This model incorporates simplifications in the aerodynamic forces lift and drag in order to reduce the number of plant parameters. The simplified aerodynamic forces capture the general flight characteristics of a fixed wing aircraft in conservative flight.

A sliding mode controller was selected due to the assumption of bounded modeling error as a result of simplified aerodynamic forces, as well as the possibility of varying wind force disturbances. Although a discrete sliding mode control provides superior tracking performance, a smooth sliding control is better suited to the application because of the reduced control actuation requirements. By incorporating robustness terms, the smooth sliding controller was shown to function in spite of 15 % modeling error. Robustness to larger errors or disturbance could be included by redesigning larger robustness terms.

For real-world applications, altitude control is preferable to pitch angle control. Unfortunately, it was shown that sliding mode altitude control causes the pitch angle to become an unstable internal dynamic. Other combinations of outputs such as  $[x \ y]$ ,  $[\dot{x} \ \dot{y}]$ , and  $[x \ \dot{y}]$  were also checked, although these were not included because the results were the same and lengthy. In order to allow altitude control while preventing unstable internal dynamics, a sliding mode control of  $[u \ \theta]$  was combined with a PI control of altitude.

When the sliding mode control is active, constant forward airspeed is accurately maintained, and the rate of altitude change can be completely determined by controlling

$\theta$ . Thus, a PI control was designed to output  $\theta$  based on the altitude tracking error. This can be understood intuitively because when the aircraft needs to increase altitude, the PI control simply aims it up. The very small integral term in the PI control also has the effect of negating the slight steady state tracking error that would otherwise be caused by the boundary layer of the smoothed sliding control. It is important to note that the PI control includes saturation terms to prevent it from commanding an unreasonable pitch angle or rate; otherwise large altitude change commands could result in instabilities.

A Lyapunov-type observer was designed to estimate the aircraft states based on the measurements  $u$  and  $q$ , which can be obtained from a pitot pressure tube and a gyroscope. The state equations as derived in chapter one do not contain enough linear terms to satisfy the observability requirement for this type of controller, so the system is replaced by its Taylor expansion about a trim condition. Thus, the linear terms come from the first term of the Taylor expansion, and the non-linear terms come from the second term of the Taylor expansion. These non-linear terms must satisfy the Lipschitz condition, which could only be shown by assuming that the pitch angle and its estimate remain small. Therefore, only the local Lipschitz requirement was met, and the observer would not function for large errors in pitch angle. However, in the generally conservative flight characteristic of an autopiloted UAV,  $\theta$  and  $\hat{\theta}$  will always be small, so the observer is appropriate.

Future work on this system should include further observer development, especially for robustness to sensor noise and modeling error. A sliding mode observer may be useful in this case. A more rigorous investigation of the stability of the combined PI and SMC control may be possible as well.

Extension of this model to include lateral dynamics would be very interesting, including design of a similar sliding mode controller and observer. If this project could be expanded to a master's thesis, I would like to compare the performance of my model to the full-state simulink model in the Vehicle Dynamics Lab, and then test my controllers on the full simulink model to see if they might be of use for the AINS project for improved low-level control.

## 8 Appendix: UAV and controller parameters

UAV Parameters:

$$C_E = 0.3 \quad xE = 1.5$$

$$C_L = 10 \quad x_L = 0.2$$

$$m = 10 \quad g = 9.8$$

$$C_D = 0.1 \quad J = 3.3$$

Sliding mode control parameters:

$$K_1 = 1 \quad K_2 = 2$$

$$\Phi_1 = 0.1 \quad \Phi_2 = 0.1$$

$$\beta = 1$$

PI control parameters:  $K_p = 1 \quad K_i = 0.05$

$$\text{Observer gain matrix: } \begin{bmatrix} 23.2 & 0.33 \\ -0.08 & 1.32 \\ -19.5 & 0.02 \\ 0.33 & 3.6 \end{bmatrix}$$

## References

- [1] Etkin, Bernard. *Second Edition: Dynamics of Flight- Stability and Control*. John Wiley and Sons, New York, 1982.
- [2] Hedrick, J.K, *Lecture notes from Mechanical Engineering 237*, given at UC Berkeley, 2004.
- [3] Raghavan, S. and Hedrick, J.K., "Observer design for a class of nonlinear systems", *International Journal of Control*, 1994, vol. 59, no. 2
- [4] Slotine, J.J., and Li. W.P., *Applied Nonlinear Control*, Prentice-Hall, 1991.