Robust and Gain Scheduled Flight Control of Fixed-wing UAVs in Icing Conditions

Ruben Kleiven

17/12/2020

Contents

Abstract

Inspired by the growing field of application of unmanned aerial vehicles(UAVs), this thesis investigates UAV-control in icing conditions. Specifically, a single robust controller and a gain scheduled controller is implemented using an $\mathcal{H}_\infty$ control approach in order to mitigate the aerodynamic effects of icing on the UAV. The control methods are tested through simulations in MATLAB/Simulink. From this, it is concluded that both the single robust controller and the gain scheduled controller gives satisfactory results in terms of stability and robustness. By comparing the simulation results using the two controllers, the control performance of gain scheduled control approach gives the best results.

Preface

The work presented in this thesis is performed at the Department of Engineering Cybernetics, NTNU. The starting point of this task was to investigate control of an UAV that was subject to icing conditions, where the aerodynamic model of the Skywalker X8 was given from previous work and analysis. I would like to my main supervisor thank professor Tor Arne Johansen and Co-supervisor Kristoffer Gryte for their help and guidance throughout this project.

Chapter 1

Introduction

In aviation, icing conditions are potentially menacing and has been the source of numerous accidents. Icing can occur both when an aircraft is on ground and airborne. For airborne aircraft, in-flight airframe icing, often referred to as atmospheric icing, occurs when supercooled water droplets in clouds strike the aircraft surface and freeze and are the type of icing that is considered in this thesis. How the icing affects aircraft performance has been a subject of research throughout the last decades([1–3]). Generally, these effects include increased drag and weight, decreased lift and instability due to adverse moments.

According to Thomas et al. [4], the airborne anti-icing and de-icing methods that are in use can be divided into three groups, freezing point depressants, thermal melting and surface deformation. Where anti-icing methods refer to methods which prevent any ice from building up, and de-icing refers to methods which remove ice from the surfaces. When looking at these techniques, it is essential to distinguish between large aircraft, where they are commonly in use, and a much smaller unmanned aerial vehicle(UAV). Even if any of these techniques would be applicable for UAVs in theory, they have some disadvantages, such as high energy requirements and additional weight, which makes them more impractical for small UAVs than for large aircraft.

In this thesis, different control strategies are implemented and tested in a MATLAB/Simulink based simulation environment of a fixed-wing UAV which is subject to in-flight airframe icing. The specific UAV that is used is the flying wings Skywalker X8, where the simulation model is based on the previous work of Gryte et al. [5] and Winter et al. [6]. The objective of these controllers is to mitigate the effects of icing while achieving desirable performance. Two control approaches are used in this thesis;

- A single robust controller which gives sufficient performance for a range of different levels of icing
- A gain scheduled controller, which uses different controller gains depending on the state of level and type of icing. This method requires a way to detect the level of icing so the appropriate gain schedule can be used.

The idea with robust control is to have a control system that is capable of regulating systems that contain some uncertainties. The amount of robustness required depends on how good the model of the control system. In general, mathematical models are approximations of the physical world. Therefore, some robustness is always needed to some extent when designing a control system. There are several ways to design a robust control system for aircraft. Many approaches are mentioned in Magni et al [7]. In this thesis, a H_∞- approach is used because of how it handles plant uncertainties in the frequency domain.

1.1 Structure of Thesis

The thesis is divided into 6 chapters.

- Chapter 2 presents the theory used in the subsequent chapters. Here, the UAV dynamics and control techniques are presented.
- Chapter 3 presents the model for the Skywalker X8 and the controller implementations.
- Chapter 4 shows the simulation results for the implemented controllers.
- The two final chapters, 5 and 6, contains the discussions and conclusions

Chapter 2

Theory

This chapter will present the basic theory and background information that is needed to comprehend the material that is introduced throughout this thesis.

2.1 Modelling the UAV

This section will give the outlines of the derivation of the model, which is later used in the simulations and to develop the controllers.

2.1.1 Coordinate Frames

The mathematical model used in this thesis is using different coordinate frames in order to describe the motion of the UAV. The equations of motions are easiest expressed with respect to a body-fixed coordinate frame, where a space fixed coordinate frame is more suitable for longer-term guidance. A short description of the coordinates frames that are used follows.

World Reference Frame

An external reference frame using the North-East-Down(NED) convention. This frame has its origin at a fixed point on earth and does not move. The x-direction of the frame points north, y-direction points east, and the z-direction points down into the centre of the earth. This frame is denoted as $\mathcal{F}^i$.

Body Frame

The body coordinate frame is denoted $\mathcal{F}^b$, and has its origin at the centre of mass of the vehicle. Here, the x-axis points out of the nose of the aircraft, the y-axis points along the right-wing, and the z-axis points out of the belly of the aircraft. By denoting a frame with origin at the vehicle centre of mass and axes aligned with the inertial frame as $\mathcal{F}^v$, the body frame can be obtained by the rotational

matrix in equation 2.1. Here, the ϕ, θ and ψ are the roll, pitch and yaw angles respectively.

$$\mathbf{R}_{v}^{b}(\phi,\theta,\psi) = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & s_\phi c_\theta \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\phi c_\theta \end{bmatrix} \tag{2.1}$$

Where $sin(x)$ and $cos(x)$ are written on a compact form s_x and c_x. The body frame coordinate frame is illustrated in figure 2.1.

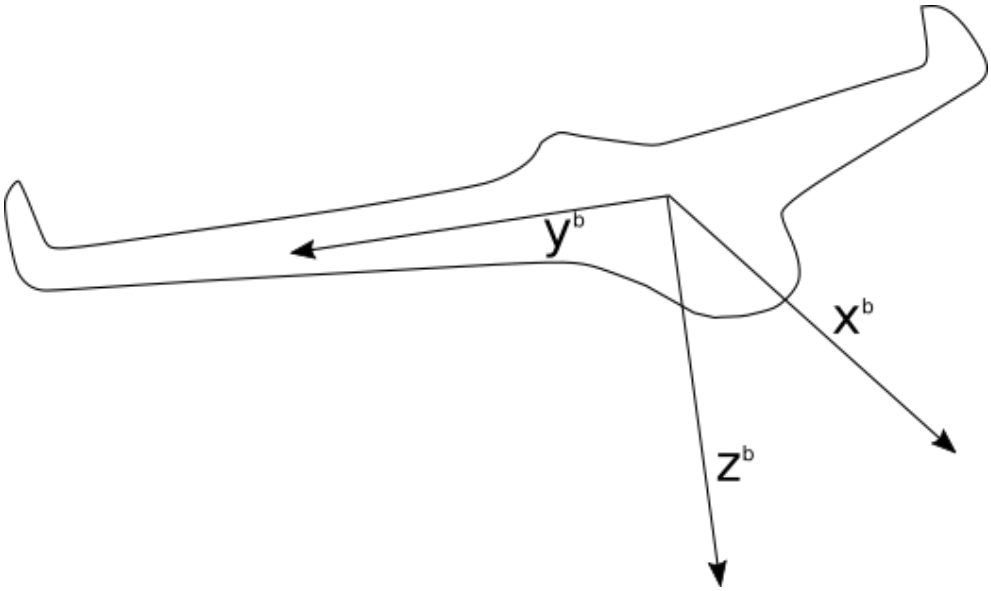


Figure 2.1: Illustration of the body frame coordinate frame, which follows the aircraft. The x-axis denoted x^b points out the aircraft nose, the y-axis denoted y^b points along the right wing of the aircraft, and the z-axis, denoted z^b is pointing out of the belly of the aircraft

Wind Frame

The wind coordinate frame, denoted $\mathcal{F}^w$, is a coordinate frame with its origin at the centre of mass of the aircraft, and the axes are oriented based on the aircraft's motion relative to its surroundings. This frame can be obtained by rotating the body frame by the rotation described by the rotational matrix in equation 2.2. This rotation corresponds to rotating the x_b and z_b axes the AOA, α, about the y_b axis. This gives a coordinate often referred to as the stability frame. In this frame, the x-axis points along the projection if airspeed-vector into the x_b-z_b plane, the y-axis still points along y_b axis. The wind frame is then obtained by rotating the stability frame axis x_s and y_s, a right-handed rotation of the AOS, β around the

z_s. Now, the x_w axis points in the direction of the airspeed, V_a.

$$\mathbf{R}_b^w(\alpha) = \begin{bmatrix} cos(\beta)cos(\alpha) & -sin(\beta)cos(\alpha) & -sin(\alpha) \\ sin(\beta) & cos(\beta) & -sin(\beta)sin(\alpha) \\ -cos(\beta)sin(\alpha) & 0 & cos(\alpha) \end{bmatrix} \tag{2.2}$$

The longitudinal aerodynamic forces are usually expressed in this frame.

2.1.2 Mathematical Model

A twelve state 6 degrees of freedom(DOF) model is used to describe the motion of the UAV. The inertial position, $\mathbf{P} = [p_n, p_e, h]$, is the UAV north, east and altitude coordinates in the inertial coordinate frame. $\mathbf{V} = [u, v, w]$ is the velocity along the x, y and z axes in the body frame. $\omega = [p, q, r]$ is the angular velocity about the same axes. The Euler angles describing the UAV attitude is $\hat{} = [\phi, \theta, \psi]$. Where ϕ is the roll angle, θ is the pitch angle, and ψ is the yaw angle.

Kinematic Equations

The positional and attitudinal states are given in another coordinate frame than the velocity and angular velocity states, where the kinematic equations describe this relation. The relationship between the derivative of **P** given in the inertial frame, and the velocity, **V**, in the body frame can be found from equation 2.1, and is shown in equation 2.3.

$$\dot{\mathbf{P}} = \mathbf{R}_b^v(\phi, \theta, \psi)\mathbf{V} = \mathbf{R}_v^b(\phi, \theta, \psi)^T\mathbf{V} \tag{2.3}$$

The relationship between the Euler angles, Θ, and the angular rates in body frame, ω, is given by equation 2.4

$$\dot{\Theta} = \begin{bmatrix} 1 & sin(\phi)tan(\theta) & cos(\phi)tan(\theta) \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi)sec(\theta) & cos(\phi)sec(\theta) \end{bmatrix} \omega \tag{2.4}$$

Dynamic Equations

General six-degrees of freedom dynamic equations for a rigid body can be obtained through a Newtonian approach. By assuming a flat earth model, and denoting the sum of all external forces acting on the rigid body as **F**, and the sum of all external moments about the centre of mass as **M**, we get the following equations

$$\begin{aligned} m(\dot{\mathbf{V}} + \omega \times \mathbf{V}) &= \mathbf{F} \\ \mathbf{I}(\dot{\omega} + \omega \times \mathbf{V}) &= \mathbf{M} \end{aligned} \tag{2.5}$$

Where **I** is the inertia matrix. In order to describe these forces and moments, it is common to assume that the forces and moments are primarily from three

sources [8], namely gravitation, propulsion and aerodynamic. Hence, the forces and moments from equation 2.5 can be written as;

$$\begin{aligned} \mathbf{F} &= \mathbf{F}_a + \mathbf{F}_g + \mathbf{F}_p \\ \mathbf{M} &= \mathbf{M}_a + \mathbf{M}_p + \mathbf{M}_g \end{aligned} \tag{2.6}$$

Where subscript p, a and g denotes propulsion, aerodynamic and gravitational forces respectively.

Gravitational Forces and Moments

The gravitational moment about the center of mass is 0. Hence $\mathbf{M}_g = 0$. The gravitational force can be written as $\mathbf{F}_g^v = [0, 0, mg]$ in the vehicle carried coordinate frame, denoted $\mathcal{F}^v$, with axes aligned with the inertial frame. The gravitational force in body frame is therefore given as;

$$\mathbf{F}_g^b = \mathcal{R}_{\mathrm{v}}^b \mathbf{F}_g^v \tag{2.7}$$

Where $\mathcal{R}_{\mathrm{v}}^b$ is the rotational matrix mapping from $\mathcal{F}^v$ to $\mathcal{F}^b$ described in 2.1.1.

Aerodynamic Forces and Moments

This section will present a general mathematical model for aerodynamic forces and moments. Here, it is assumed that Mach effects are negligible for the operating speeds of the UAV, the body is rigid, and the Reynolds number effects are small. Another common simplification is to separate the aerodynamic equations into longitudinal ($\mathbf{x_b}$-$\mathbf{z_b}$ plane) and lateral direction ($\mathbf{x_b}$-$\mathbf{y_b}$- plane) [8]. By denoting V_a as the airspeed of the aircraft, S as the area of the wing, b as the wingspan of the aircraft and c as the mean chord of the wing, the aerodynamic forces and moments in the longitudinal direction can be expressed as in equation 2.8.

$$\begin{bmatrix} D \\ L \\ m \end{bmatrix} = \frac{1}{2}\rho V_a^2 S \begin{bmatrix} C_D(\alpha, q, \delta_e) \\ C_L(\alpha, q, \delta_e) \\ cC_{\mathrm{m}}(\alpha, q, \delta_e) \end{bmatrix} \tag{2.8}$$

The lateral direction aerodynamic forces and moments can be expressed as in equation 2.9

$$\begin{bmatrix} Y \\ l \\ n \end{bmatrix} = \frac{1}{2}\rho V_a^2 S \begin{bmatrix} C_Y(\beta, p, r, \delta_a, \delta_r) \\ bC_{\mathrm{l}}(\beta, p, r, \delta_a, \delta_r) \\ bC_{\mathrm{n}}(\beta, p, r, \delta_a, \delta_r) \end{bmatrix} \tag{2.9}$$

In equations 2.8 and 2.9, D, Y and L are the drag, side and lift aerodynamic forces along the wind-frame x, y and z axes respectively and l, m, n are the aerodynamic moments about body-frame x,y and z axes. C_D, C_Y and C_L, are non-dimensional drag, lift and side force coefficients, and C_l, C_m and C_n are non-dimensional are rolling, pitching and heading moment coefficients. α and β are

the AOA and AOS. The control inputs that affects the aerodynamic forces and moments are δ_e, δ_a and δ_r, which is the deflection angle of the elevator, ailerons and rudder. By denoting the aerodynamic forces along the body-frame axes as F_x , F_y and F_z, they can be expressed as in equation 2.10.

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \mathbf{R}_w^b(\alpha) \begin{bmatrix} -D \\ Y \\ -L \end{bmatrix} \tag{2.10}$$

Where $\mathbf{R}_s^b(\alpha)$ is the rotational matrix from $\mathcal{F}^w$ to $\mathcal{F}^b$. The drag force and lift force are acting in the negative x and z direction, which explains the choice of signs in equation 2.10. Hence, the summarized equations of aerodynamic forces and moments are gicen in equation 2.11

$$\begin{aligned} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} &= \mathbf{R}_w^b(\alpha) \frac{1}{2} \rho V_a^2 S \begin{bmatrix} C_D(\alpha, q, \delta_e) \\ C_Y(\beta, p, r, \delta_a, \delta_r) \\ C_L(\alpha, q, \delta_e) \end{bmatrix} \\ \begin{bmatrix} l \\ m \\ n \end{bmatrix} &= \frac{1}{2} \rho V_a^2 S \begin{bmatrix} bC_{\mathrm{l}}(\beta, p, r, \delta_a, \delta_r) \\ cC_{\mathrm{m}}(\alpha, q, \delta_e) \\ bC_{\mathrm{n}}(\beta, p, r, \delta_a, \delta_r) \end{bmatrix} \end{aligned} \tag{2.11}$$

Propulsion Forces and Moments

A simple model for thrust force can be obtained by applying Bernoulli's equation to the air in front of and behind the propeller [8]. This model assumes that the relation between throttle to exit velocity is linear, and that the propeller efficiency is constant. It also assumes that the thrust force generated is directly along the x-axis of the body frame. By denoting S_{prop} as the area swept out by the propeller, and C_{prop} be an efficiency factor, the propulsion force is given in equation 2.12

$$\mathbf{F_p} = \frac{1}{2} \rho S_{\text{prop}} C_{\text{prop}} \begin{bmatrix} (k_{\text{motor}} \delta_t)^2 - V_a^2 \\ 0 \\ 0 \end{bmatrix} \tag{2.12}$$

2.2 $\mathcal{H}_\infty$ control

The effect of icing can be looked at as uncertain plant model As briefly mentioned in the introduction, the H_∞ robust control approach is used because of how it handles plant uncertainties in the frequency domain. In order to apply $\mathcal{H}_\infty$ control method, the closed-loop system needs to be written as a generalized plant shown in the block diagram of this form is shown in figure 2.2. Here the process block, P has two inputs; The control input **u** and the exogenous inputs **w**, including disturbances, sensor noise and reference signals. P also has two outputs; the measured output signal used by the controller, **y**, and the weighted error signals, **z**. The error signals are a collection of variables to be regulated.

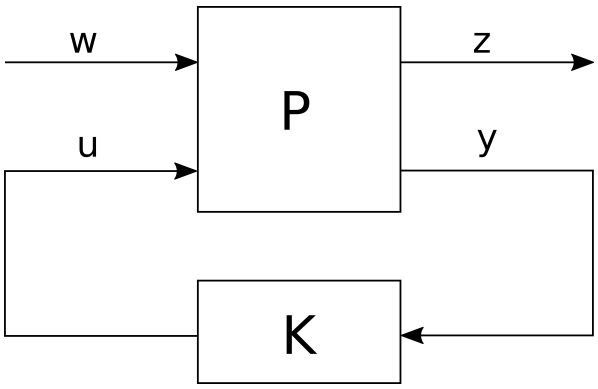


Figure 2.2: H_∞ closed loop block diagram formulation. Where P is the process, $\mathbf{w}$ is the exogenous inputs, $\mathbf{z}$ is the generalized error signals, $\mathbf{y}$ is the measured output signal used by the controller C.

Problem Formulation

For a linearized process, we can write the state-space equations as in 2.13.

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \\ \mathbf{z} &= \mathbf{C}_1\mathbf{x} + \mathbf{D}_2\mathbf{u} + \mathbf{D}_1\mathbf{w} \\ \mathbf{y} &= \mathbf{C}_2\mathbf{x} + \mathbf{D}_4\mathbf{u} + \mathbf{D}_3\mathbf{w} \end{aligned} \tag{2.13}$$

Where $\mathbf{x}$ is the system states. From equation 2.13, P in figure 2.2 can be found as shown in equation 2.14.

$$\mathbf{P}(s) = \left[\begin{array}{c|c} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \hline \mathbf{P}_{21} & \mathbf{P}_{22} \end{array}\right] = \left[\begin{array}{c|cc} \mathbf{A} & \mathbf{E} & \mathbf{B} \\ \hline \mathbf{C}_1 & \mathbf{D}_1 & \mathbf{D}_2 \\ \mathbf{C}_2 & \mathbf{D}_3 & \mathbf{D}_4 \end{array}\right] \tag{2.14}$$

For the linear controller, $\mathbf{K}$, connected from y to u as shown in figure 2.2, the closed loop transfer function from exogenous inputs to the weighted error signal can be written as a function of the process and the controller, as shown in 2.15 [9].

$$\frac{\mathbf{z}}{\mathbf{w}}(s) = \mathcal{F}(\mathbf{P},\mathbf{K})(s) = \mathbf{P}_{11} + \mathbf{P}_{12}\mathbf{K}(\mathbf{I} - \mathbf{P}_{22}\mathbf{K})^{-1}\mathbf{P}_{21} \tag{2.15}$$

The controller design goal is to minimize the weighted error-signals $\mathbf{z}$, i.e. such that the gain of the transfer function in equation 2.15 is as small as possible, while the system remains internally stable. In other word, this means a controller that minimizes the worst case effects of the exogenous inputs $\mathbf{w}$ on the weighted error signal $\mathbf{z}$. The H_∞-norm gives a measure of transfer function gain. Hence, the controller design can be obtained by minimizing the infinity norm of $\mathcal{F}(\mathbf{P},\mathbf{K})(s)$, where the standard infinity norm is formulated as in equation 2.16

$$||\mathcal{F}(\mathbf{P},\mathbf{K})||_\infty = \sup_\omega \bar{\sigma}(\mathcal{F}(\mathbf{P},\mathbf{K})(j\omega)) \tag{2.16}$$

Where $\bar{\sigma}$ is the maximum singular value.

Weighting Filters

This algorithm uses the weighting filters as tuning parameters to get a controller meeting the desired frequency requirements. By denoting the model plant as $G(s)$, the open loop transfer function is given as $L(s) = G(s)K(s)$. The sensitivity function $S(s)$ and complementary sensitivity function, $T(s)$, can then be written as in equation 2.17

$$\begin{aligned} S(s) &= (I + L(s))^{-1} \\ T(s) &= L(I + L(s))^{-1} \end{aligned} \tag{2.17}$$

Where I is the identity matrix of the same order as $L(s)$. From the block diagram in figure 2.3, we can write the the transfer function from reference r to error, $e(s)$ and measured output $y(s)$ as:

$$\begin{aligned} e(s) &= S(s)r(s) \\ y(s) &= T(s)r(s) \end{aligned} \tag{2.18}$$

From the same block diagram, we now see that the error-signals z_1 and z_3 are the weighted sensitivity and complementary sensitivity, and z_2 is the weighted control activity. Generally, you want to have high loop-gain at low frequencies to have good reference tracking, but the loop gain should roll-off at high frequencies in order to achieve good noise attenuation and robustness. This is equivalent to have a small $S(s)$ at low frequencies(inside the bandwidth), and a small $T(s)$ at high frequencies (outside the bandwidth). There are several approaches to design weighting filters. Two specific methods are discussed in Lundstrøm et al [10], namely a signal approach used in Doyle et al [11] and loop-shaping. The signal approach considers the response to sinusoidal signals, and are more appropriate for multi-variable problems[10]. Hence, this thesis uses a loop-shaping approach, which is described in Lavertsky et al. [12]. The weighting filters can be used to design the desired $S(s)$ and $T(s)$. Here, $S(s)$ can be used to reach acceptable bandwidth and achieve desired disturbance rejection and reference tracking, while $T(s)$ can be used to avoid amplification of high-frequency noise. This is a trial-error procedure. The weighting filters should be selected to be of minimum order in order to avoid adding unnecessary complexity to the controller. The shape of the weighting filter should be the inverse of the desired shape of the unweighted function. By using the sensitivity function $S(s)$ as an example, the W_S should be the inverse of the desired shape of $S(s)$, such that when the H_∞ norm of z_1, e.i. $||W_S S||_\infty$, is minimized, it will shape $S(s)$.

Output Feedback H_∞ Control

In order to obtain an output feedback controller, a common approach is γ-iteration. This method is thoroughly described in Lavertsky et al. [12]. By demanding the

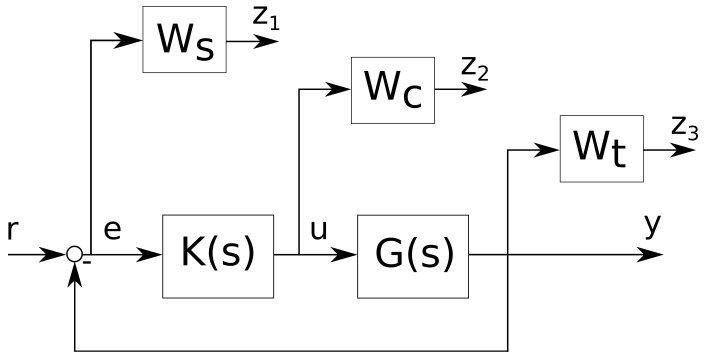


Figure 2.3: Block diagram where W_S, W_C and W_T are weighting filters. z_1, z_2 and z_3 are the weighted sensitivity, control activity and complementary sensitivity.

infinity norm of $\mathcal{F}(\mathbf{P},\mathbf{K})(s)$ to be less then some positive and predefined peak $||\mathcal{F}(\mathbf{P},\mathbf{K})||_\infty$ value γ, we can look at the problem of finding all stabilizing sub-optimal controllers such that equation 2.19 is fulfilled.

$$||\mathcal{F}(\mathbf{P},\mathbf{K})(s)||_\infty < \gamma \tag{2.19}$$

There are several numerical methods for solving this problem. One method is to solve algebraic riccati equations, and then find the optimal controller by iterating on γ.

Strong robustness results related to the $\mathcal{H}_\infty$ controller. By introducing the Vinnicombe metric, often referred to as the ν-gap metric, which is a distance measure between the closed-loop systems of two plants P_1 and P_2. A thorough description of this metric is given in Vinnicombe [13]. As we have the plant model for both clean and iced case from Winter et al. [6], this metric can be used as a measure of plant uncertainty. By letting P_1 and P_2 be two lateral scalar transfer functions that satisfies the winding number conditions given in equation 2.20, the ν-gap metric is defined as in equation 2.21.

$$(1+P_2P_1)(j\omega) \neq 0 \forall \omega \tag{2.20a}$$

$$\text{wno}(1+P_2P_1) + \eta(P_1) - \eta(P_2) = 0 \tag{2.20b}$$

Where wno is the winding number on the standard Nyquist contour. of the and $\eta(P_i)$ denotes the number of right half plane poles of P_i. Note that since the plants that are being considered later in this thesis are scalar, both equations 2.20 and 2.21 are simplified to yield scalar transfer functions. General multivariable plants are given in

$$\delta_\nu(P_1,P_2) = \sup_\omega \frac{|P_1(i\omega) - P_2(i\omega|}{\sqrt{(1+|P_1(i\omega)|)^2(1+|P_2(i\omega)|)^2}} \tag{2.21}$$

According to Åstrom and Murray[14], from this definition it follows that a H_∞ controller with performance level γ for plant P_1 stabilizes all plants with ν-gap distance less than $\frac{1}{\gamma}$ from P_1.

2.3 Gain Scheduled Controller

A gain scheduled approach is a method where a there is a set of controllers for designed for different operating points in the operational area, and a selector mechanism is implemented to select the controller suitable for the current state of the system. This is a practical approach to control non-linear systems where the process gain is changing considerably at different operating points [15]. Gain scheduling is commonly used when a single set of gains does not provide the desired performance and stability requirements throughout the area of operating conditions. In order for this method to be implemented, there have to exist measurable variables that correlate with the change in process gain.

Chapter 3

Methods

3.1 Simulation Model

A model of Skywalker x8 is used in this thesis. All parameters in this section are from Gryte [16] unless other is stated. The complete simulation model is obtained by using the theory from section 2.1 and the parameter values are given this section. The air density $\rho = 1.2250$ kg/m^3 and gravitational constant $g = 9.8066$ m/s^2 are assumed constant during the simulations.

Elevons

Skywalker X8 is a flying wing UAV, and it is depicted in figure 2.1. For a flying wing UAV, the control surfaces ailerons and rudder are combined into a pair of elevons. By respectively denoting the left and right elevon deflection as δ_{el} and δ_{er}, the mathematical conversion between elevons to aileron-elevator signals is shown in equation 3.1 [8].

$$\begin{bmatrix} \delta_e \\ \delta_a \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \delta_{er} \\ \delta_{el} \end{bmatrix} \tag{3.1}$$

Hence, the standard derivation using aileron and elevator signals in equation 2.11 can be used. Another characteristic feature with the flying wing X8 is that it has no tail or rudder.

Aerodynamic Coefficients

The aircraft lift, drag, side force, pitching, rolling and heading moment coefficients from equation 2.11 are in general non-linear equations of the system states. A common simplification of the forces and moments in equation this can be found by using first order Taylor series expansions [8]. This is shown in equation 3.2, where the partial derivative of a state is denoted with subtext. I.e. C_{i_x} is the partial

derivative of C_i with respect to x.

$$\begin{bmatrix} C_D(\alpha,q,\delta_e) \\ C_Y(\beta,p,r,\delta_a) \\ C_L(\alpha,q,\delta_e) \\ C_{\mathrm{l}}(\beta,p,r,\delta_a) \\ C_{\mathrm{m}}(\alpha,q,\delta_e) \\ C_{\mathrm{n}}(\beta,p,r,\delta_a) \end{bmatrix} = \begin{bmatrix} C_D(\alpha) + \frac{c}{2V_a} C_{D_q} q + C_{D_{\delta_e}} \delta_e \\ C_{Y_0} + C_{Y_\beta}\beta + \frac{b}{2V_a} C_{Y_p} p + \frac{b}{2V_a} C_{Y_r} r + C_{Y_{\delta_a}} \delta_a \\ C_L(\alpha) + \frac{c}{2V_a} C_{L_r} r + C_{L_{\delta_e}} \delta_e \\ C_{\mathrm{l}_0} + C_{\mathrm{l}_\beta}\beta + \frac{b}{2V_a} C_{\mathrm{l}_p} p + \frac{b}{2V_a} C_{\mathrm{l}_q} q + C_{\mathrm{l}_{\delta_a}} \delta_a \\ C_{\mathrm{m}}(\alpha) + \frac{c}{2V_a} C_{\mathrm{m}_q} q + C_{\mathrm{m}_{\delta_e}} \delta_e \\ C_{\mathrm{n}_0} + C_{\mathrm{n}_\beta}\beta + \frac{b}{2V_a} C_{\mathrm{n}_p} p + \frac{b}{2V_a} C_{\mathrm{n}_r} r + C_{\mathrm{n}_{\delta_a}} \delta_a \end{bmatrix} \tag{3.2}$$

Here, the $\frac{c}{2V_a}$ and $\frac{b}{2V_a}$ are standard factors used for nondimensionalization. Since the model is of a UAV with no rudder, δ_r is removed from the equation set. Due to the nonlinearities of C_D, C_L and C_m in α, these parameters are kept as a nonlinear function of α to give a more accurate simulation model. $C_{D_{\delta_e}}$, $C_{Y_{\delta_a}}$, $C_{L_{\delta_e}}$, $C_{\mathrm{l}_{\delta_a}}$, $C_{\mathrm{m}_{\delta_e}}$ and $C_{\mathrm{n}_{\delta_a}}$ are denoted control derivatives.

The aerodynamic coefficient model can be extended to include the element of icing. An icing variable ζ is used to describe the level of icing. As emphasized by Winter et al. in [6], the atmospheric icing will primarily affect the leading edge of the aircraft. Hence, the assumption that the control derivatives is not affected by icing during the simulations. The aerodynamic coefficient with icing cab then be mathematically expressed as in equation 3.3

$$\begin{aligned}
C_D(\alpha,q,\delta_e,\zeta) &= C_D(\alpha,\zeta) + C_{D_q}(\zeta)\frac{c}{2V_a}q + C_{D_{\delta_e}}\delta_e \\
C_Y(\beta,p,r,\delta_a,\zeta) &= C_{Y_0}(\zeta) + C_{Y_\beta}(\zeta)\beta + C_{Y_p}(\zeta)\frac{b}{2V_a}p + C_{Y_r}(\zeta)\frac{b}{2V_a}r + C_{Y_{\delta_a}}\delta_a \\
C_L(\alpha,q,\delta_e,\zeta) &= C_L(\alpha,\zeta) + C_{L_q}(\zeta)\frac{c}{2V_a}q + C_{L_{\delta_e}}\delta_e \\
C_{\mathrm{l}}(\beta,p,r,\delta_a,\zeta) &= C_{\mathrm{l}_0}(\zeta) + C_{\mathrm{l}_\beta}(\zeta)\beta + C_{\mathrm{l}_p}(\zeta)\frac{b}{2V_a}p + C_{\mathrm{l}_r}(\zeta)\frac{b}{2V_a}r + C_{\mathrm{l}_{\delta_a}}\delta_a \\
C_{\mathrm{m}}(\alpha,q,\delta_e,\zeta) &= C_{\mathrm{m}}(\alpha,\zeta) + C_{\mathrm{m}_q}(\zeta)\frac{c}{2V_a}q + C_{\mathrm{m}_{\delta_e}}\delta_e \\
C_{\mathrm{n}}(\beta,p,r,\delta_a,\zeta) &= C_{\mathrm{n}_0}(\zeta) + C_{\mathrm{n}_\beta}(\zeta)\beta + C_{\mathrm{n}_p}(\zeta)\frac{b}{2V_a}p + C_{\mathrm{n}_r}(\zeta)\frac{b}{2V_a}r + C_{\mathrm{n}_{\delta_a}}\delta_a
\end{aligned} \tag{3.3}$$

The icing variable ζ takes values between 0 and 1, where 0 indicates no ice, and 1 indicates a worst-case level of ice (mixed ice). Throughout this section, no ice is referred to as clean, and worst-case level of ice is referred to as iced. In the simulations, parameter values are collected from the work done in [5, 6]. The effects of icing is implemented to the simulation environment by using linear interpolation between the model values with no ice and mixed ice. This implementation technique assumes that the effects of icing is linearly evolving with the level of ice, which is a simplification.

The value of control derivatives used in the simulations is given in table 3.1.

Table 3.1: Control Derivatives

Control Derivative	Value
$C_{D_{\delta_e}}$	0.8461
$C_{Y_{\delta_a}}$	−0.0696
$C_{L_{\delta_e}}$	0.5872
$C_{\mathrm{l}_{\delta_a}}$	0.2987
$C_{\mathrm{m}_{\delta_e}}$	−0.4857
$C_{\mathrm{n}_{\delta_a}}$	0.0076

The parameters forming the aerodynamic coefficients are collected from Winter [17]. As previously mentioned, $C_D(\alpha,\zeta)$, $C_D(\alpha,\zeta)$ and $C_{\mathrm{m}}(\alpha,\zeta)$ are kept as non-linear function during the simulations. Plots of these functions are shown in figure **??**.

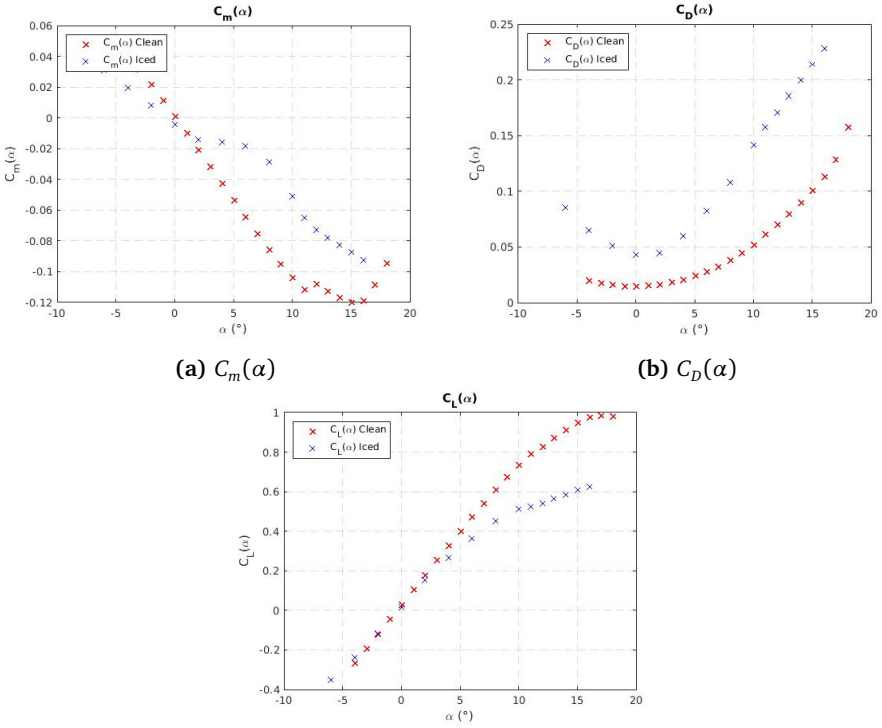
As shown in equation 3.3, C_n, C_Y and C_l are all linearized in beta in the simulation model. The values of the constant and linear factor for each parameter are given in table 3.2.

Table 3.2: Shows the values of C_n, C_Y and C_l when linearized with respect to β.

Parameter	Clean	Iced
C_{n_0}	0.0000	0.0000
C_{n_β}	0.0297	0.0348
C_{l_0}	0.0000	0.0000
C_{l_β}	−0.1010	−0.0861
C_{Y_0}	0.0000	0.0000
C_{Y_β}	−0.2700	−0.2300

The final derivatives in equation 3.3 are the quasi-static derivatives, which are the ones depending on the roll, pitch and yaw rates p, q and r. These derivatives are damping coefficients. The lateral derivatives are shown in table 3.3. These values are primarily collected from Winter [17]. However, the sign of C_{n_p} is inverted. C_{n_p} contributes to adverse yawing when negative. This is in line with the work of Gryte et al. [5]. C_{n_p} is in other words, essential when describing the coupling between yaw and roll. It is important to emphasize that the damping coefficients are particularly hard to find precise values.

The influence of pitch rate on drag is C_{D_q} is usually neglected, and are therefore set equal to zero during the simulations. The other two longitudinal damping coefficients, namely C_{L_q} and C_{m_q} are given as functions of angle of attack(AOA), α, in Winter et al. [6]. It is measured at low (2°) and high (8°) for clean and iced case. The changes in values with respect to AOA is quite small. Here, the coefficients are assumed constant, and the values the values that are used are



(a) $C_m(\alpha)$ **(b)** $C_D(\alpha)$

(c) $C_L(\alpha)$

Figure 3.1: Shows the C_m, C_D and C_L as function of AOA, α

Table 3.3: Shows the values of the lateral Quasi-static derivatives C_{Y_p}, C_{Y_r}, C_{l_p}, C_{l_r}, C_{n_p} and C_{n_r}

Parameter	Clean	Iced
C_{Y_p}	−0.185	−0.034
C_{Y_r}	0.005	0.002
C_{l_p}	−0.409	−0.407
C_{l_r}	0.039	0.158
C_{n_p}	−0.027	−0.017
C_{n_r}	−0.022	−0.049

found by linear interpolation between the values for low and high AOA, and use the value for $\alpha = 4°$. The values are given in table 3.4.

Table 3.4: Shows the values of the longitudinal damping coefficients C_{L_q} and C_{m_q}

Parameter	Clean	Iced
C_{L_q}	4.63	−3.38
C_{m_q}	−1.99	−2.06

Actuator Model

For physical actuator components, there are both static and dynamic limitations. In order to implement this in the simulator, there are both rate limits and minimum/maximum saturation applied to the control signals. For the elevons, the rate limit and saturation of servos are set to ±3.4907 rad/s and $(-30°, 35°)$ respectively. These values collected from the work of K.Gryte [16], where the rate limit is based on a typical delay for a mini servo, and the angular deflection limit is based on measurements on the Skywalker X8.

The motor dynamics is modeled as a simple first order model as in equation 3.4

$$\frac{n}{n_{\text{command}}} = \frac{\frac{1}{\tau}}{1 + \frac{1}{\tau}s} \tag{3.4}$$

Where n is the motor speed and n_{command} is the commanded motor speed. During the simulations, time constant $\tau = 0.2$ is used.

Skywalker X8 Model Parameters

Table 3.5 shows the total mass, moments of inertia and products of inertia to the UAV. Here , $I_{xy} = I_{yx}$ and $I_{yz} = I_{zy}$ are assumed negligible. The values moments of inertia are based on experimental data from Gryte [18], which was performed after the report done in Gryte [16]. The new data seemed more realistic than the ones calculated from the first experiment. Hence, these are the values that are used in the simulations and stated in table 3.5.

Table 3.6 shows parameters used during simulations.

3.2 Trim Condition

A common first step to develop linear aircraft controllers is to specify an operating point where trim conditions are calculated. Mathematically, trim conditions can be found by solving a set of nonlinear algebraic equations. This is explained in greater detail in Beard and McLain [8]. In this thesis, the Matlab function trim is used to calculate the trim conditions. This function starts from a given initial

Table 3.5: Skywalker X8 mass, moments of inertia and products of inertia

Parameter	Clean	
m	3.3650	kg
I_{xx}	0.340	kgm^2
I_{yy}	0.165	kgm^2
I_{zz}	0.400	kgm^2
I_{xy}	0.000	kgm^2
I_{xz}	−0.031	kgm^2
I_{yz}	0.000	kgm^2

Table 3.6: Skywalker X8 physical measures

Parameter	Symbol	Iced	
Wing span	b	2.1	m
Mean aerodynamic chord	c_M	0.3571	m
Wing area	S_w	0.75	m^2
Area swept by propeller	S_p	0.1018	m^2
Motor constant	k_m	40	
Motor efficiency factor	C_{prop}	1	

point and then uses a sequential quadratic programming algorithm to find the nearest trim point of a dynamical system. The initial starting point used for the simulations in this thesis is given by table 3.7.

Table 3.7: Shows initial values used to find a trim point

Parameter	Symbol	Value	
Airspeed	V_a	20.0	m/s
Angle of attack	α	2.5	(deg)
Sideslip angle	β	0.0	(deg)
Height above ground	h	50.0	m

3.3 Linearization

The controllers that are implemented in this thesis are linear controllers, and is based on a linearized system. The system is linearized about the trim condition found in section 3.2. The linearized model can be written as in equation 3.5.

$$\Delta\dot{\mathbf{x}} = \frac{df}{d\mathbf{x}}\Delta\mathbf{x} + \frac{df}{d\mathbf{u}}\Delta\mathbf{u} \tag{3.5}$$

Where $\mathbf{x}$ is the state vector containing the twelve system state, and f is defined as in equation 3.6. $\frac{df}{d\mathbf{x}}$ and $\frac{df}{d\mathbf{u}}$ are the jacobian matrices of $\mathbf{f}$ with respect to $\mathbf{x}$ and $\mathbf{u}$ respectively.

$$\mathbf{f} = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \vdots \\ \dot{x_{12}} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{12} \end{bmatrix} \tag{3.6}$$

The system is linearized by using a finite difference method to calculate the system Jacobian matrix. This is shown in equation 3.7.

$$\frac{df_i}{\partial x_j} = \frac{f_i(\mathbf{x} + \mathbf{e}_j \Delta x) - f(\mathbf{x})}{\Delta x} \tag{3.7}$$

Where e_j is a vector of the same length as $\mathbf{x}$, where the j'th element is 1, and all other elements are 0. Δx is a small value chosen equal to 10^{-3}.

3.4 Lateral and Longitudinal Decomposition

The controller designs used in this thesis are based on decoupling the longitudinal and lateral flight dynamics into two separate control systems. The lateral controller controls the ailerons, and the longitudinal controller regulates throttle and elevator. Since this is a flying wing and ailerons and elevator are elevons, they can still be set separately using the relationship from equation 3.1. An important note here is that this type of lateral and longitudinal decoupling only is appropriate for gentle manoeuvring, and not appropriate for cases where there is a high degree of lateral-longitudinal cross-coupling nor for aggressive manoeuvres.

3.5 Lateral Control

The inner loop lateral controller uses the elevons and the relationship in equation 3.1 to calculate the desired control signal δ_a based on the desired roll angle ϕ_c. A new lateral state vector and input is defined as in equation 3.8.

$$\mathbf{x}_{\text{lat}} = \begin{bmatrix} v \\ p \\ r \\ \phi \end{bmatrix} \tag{3.8a}$$

$$\mathbf{u}_{\text{lat}} = \delta_a \tag{3.8b}$$

The lateral state equations are extracted from the linearized model in section 3.3, and used to obtain a lateral linear state space model as in equation 3.9.

$$\Delta \dot{\mathbf{x}}_{\text{lat}} = \frac{df_{\text{lat}}}{d\mathbf{x}_{\text{lat}}} \Delta \mathbf{x}_{\text{lat}} + \frac{df_{\text{lat}}}{d\mathbf{u}_{\text{lat}}} \Delta \mathbf{u}_{\text{lat}} \tag{3.9}$$

.

The lateral dynamics of the UAV includes the roll rate damping mode, a spiral mode and the dutch roll mode. These modes are obtained by from the eigenvalues to the lateral state matrix from equation 3.9, and listed in table 3.8. By looking at the open-loop transfer function from input ϕ_c to output ϕ, there is a right half plane(RHP) zero at 0.7961. The presence of a RHP zero, i.e. the system is a non-minimum phase gives an upper boundary to the bandwidth frequency and hence gives a fundamental limit to performance. Generally, these systems are inherently difficult to control. In order to meet the robustness requirements for such systems, the common approach is to reduce the bandwidth, which reduces the performance.

Table 3.8: Values of lateral dynamics modes

Mode	Clean	Iced
Spiral mode	-0.2517+0.0000i	-0.8666 + 0.0000i
Roll damping mode	-26.6236 + 0.0000i	-25.3899 + 0.0000i
Dutch-roll mode	-0.6360± 3.2100i	-1.2132± 2.1586i

3.6 Longitudinal Control

The inner loop longitudinal control calculates the desired control signals throttle, δ_t, and elevator, δ_e based on the desired airspeed V_{ac} and desired pitch angle θ_c. This control problem is complicated since the airspeed and pitch angle are two strongly coupled quantities that both need to be controlled. There are several ways to approach this control problem. One way is to use two single input single output(SISO) controllers and introduce a state-machine with modes which are used to decide how δ_t and δ_e are calculated. Typical state-machine modes of operation are Descend mode, Climb mode and Level flight mode. This approach is described in Beard and McLain [8]. The controllers in this thesis are based on approach, but only the level flight mode controller is considered. It is important to notice that there are some drawbacks using this method. Generally, by controlling one variable using one controller, the other states controlled by the other decoupled controller may be driven away from its equilibrium point. For instance, if airspeed is being kept constant by the throttle controller, the altitude responds rapidly to even small changes in aircraft pitch. More complex control techniques, such as Total energy control system(TECS) (described in Faleiro and Lambregts [19]) have later been introduced to cope with these problems.

A new longitudinal state vector and inputs are defined as in equation 3.10

$$\mathbf{x}_{\text{lon}} = \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} \tag{3.10a}$$

$$\mathbf{u}_{\text{lon}} = \begin{bmatrix} \delta_e \\ \delta_t \end{bmatrix} \tag{3.10b}$$

By the using the same method as for the lateral system in section 3.5, the longitudinal state equations can can be extracted from the linearized model in section 3.3, which gives the longitudinal linear state space model in equation 3.11.

$$\Delta\dot{\mathbf{x}}_{\text{lon}} = \frac{df_{\text{lon}}}{d\mathbf{x}_{\text{lon}}}\Delta\mathbf{x}_{\text{lon}} + \frac{df_{\text{lon}}}{d\mathbf{u}_{\text{lon}}}\Delta\mathbf{u}_{\text{lon}} \tag{3.11}$$

The longitudinal dynamics includes the short period modes and the phugoid modes. These modes are obtained from the longitudnal state matrix and listed in table 3.9.

Table 3.9: Values of longitudinal dynamics modes

Mode	Clean	Iced
Short period modes	-9.9357 ± 15.6017i	-8.5264 ± 4.0733i
Phugoid modes	-0.4389 ± 0.3774i	-0.5409 ± 0.4132i

3.7 Implementing Controllers

In aircraft control, it is quite common to distinguish between inner and outer loop control. The task of the low-level inner loop control system is to stabilize the body-frame states using measurements and actuators. In other words, the inner loop keeps the aircraft flying. The tasks of the higher level outer loop control is to implement guidance. This thesis looks at the inner loop control.

3.7.1 Airspeed Control

The main effect of icing on the SISO-airspeed controller is increased drag. This means that more throttle is needed to maintain the airspeed. This will also lower the maximum airspeed possible to achieve for the UAV. For considering in-flight airspeed control well below the maximum achievable airspeed, a single PI airspeed controller is found and used for all cases in the simulation. The output of the airspeed controller is given by equation 3.12

$$\delta_t = \tilde{\delta}_t + (K_p + \frac{K_i}{s})(V_a^c - V_a) \tag{3.12}$$

Where $\tilde{\delta}_t$ is the calculated trim-value, and K_p and K_i are tuning parameters. Values K_p and K_i are found by trial and error. By demanding an overshoot less than 15% and trying to get as short as possible settling time used to reach an acceptable error as possible, the values of $K_p = 0.124$ and $K_i = 0.123$ are obtained. The step response of the airspeed controller for the iced and clan cases are shown in figure 3.2, with a step of 1 m/s.

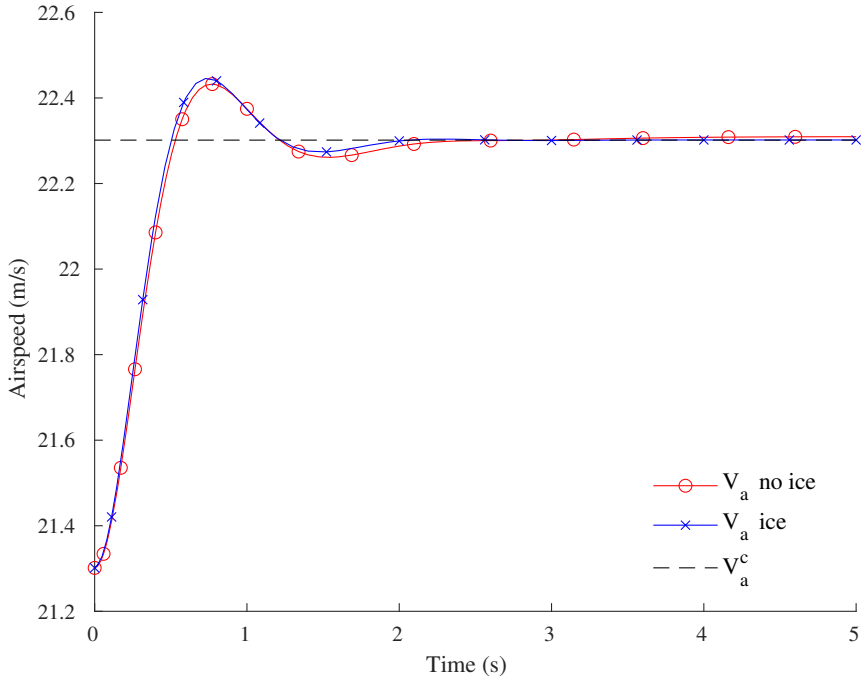


Figure 3.2: Shows the airspeed step-response from the trimmed value with a step size of 1 m/s

3.7.2 Robust Control

A robust controller is obtained using output feedback an H_∞ control algorithm as described in section 2.2. The weighting filters are designed based on arguments given in that section. A starting point for the filters are given in equation 3.13.

$$W_S = \frac{s/M + 1)}{s + \omega_0 A} \tag{3.13a}$$

$$W_C = \text{Constant} \tag{3.13b}$$

$$W_T = \frac{s + \omega_0/M}{As + \omega_0} \tag{3.13c}$$

Where ω_0 is the desired controller bandwidth, M is limiting the maximum sensitivity peak, A gives limits the low frequency gain. As mentioned in chapter

2, W_T are usually formed to get sufficient noise attenuation, and robustness to uncertain high-frequency dynamics. Icing is can bee looked at as a low frequency disturbance and the simplification of excluding measurement noise is made during the simulations. Hence this W_T is simply chosen as a high pass filter symmetric W_s around the bandwidth as shown in equation 3.13. However, in order to not get unrealistically high bandwidth, a maximum crossover frequency at 2.5 Hz is chosen based on the control choices made in Lavertsky [12]. Hence, this is the noise attenuation requirement. This system differs from the ones mentioned in Lavertsky [12]. However, the noise attenuation requirements are comparable.

Control activity is limited by the $|W_C|^{-1}$. By choosing W_C as a constant, all control activity is penalized equally at all frequencies. If this constant is chosen to small (e.i. control activity does not get a high penalty), it can lead to oscillatory responses. If the constant is chosen too large (e.i. control activity gets a high penalty), it can reduce the bandwidth capability of the controller. This is shown in figure 3.3, where a control variable y is controlled using a high and low value of W_C, while other filter parameters are constant.

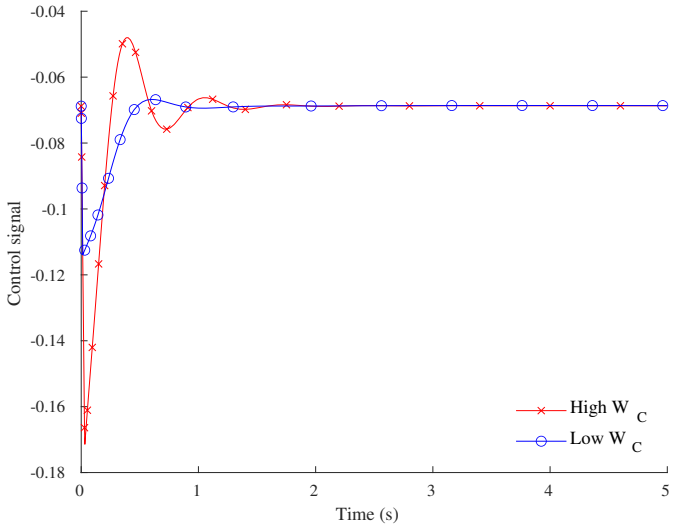


Figure 3.3: Shows the change in control signals of the controlled stat, y, with high and low W_C

Closed Loop Specifications

The control objective is to have controllers with good reference tracking and reasonable gain, phase stability margins for both the clean and iced case. When tuning the $\mathcal{H}_\infty$-controller, stability, robustness and performance requirements are considered. The stability requirement is formed by using the ν-gap introduced in chapter 2 to express the plant uncertainty when the UAV is subject to icing

conditions. The winding condition in 2.20 is satisfied for both the lateral and longitudinal case. The ν-gap between the two plants is calculated using the gapmetric function in MATLAB. The $\mathcal{H}_\infty$-controller is based on a nominal plant. This nominal plant is chosen as the plant with the approximately same ν-gap distance to the clean and iced plant. By denoting the clean plant as P_{clean}, the iced plant as P_{iced}), and the nominal plant dependant on the icing level ζ as $P_{\text{nominal}}(\zeta)$, the stability requirement is expressed in 3.14

$$\frac{1}{\gamma} > \text{Max}(\delta_\nu(P_{\text{clean}}, P_{\text{nominal}}), \delta_\nu(P_{\text{nominal}}, P_{\text{iced}})) \tag{3.14}$$

Where γ upper limit to the acceptable $\mathbf{H}_\infty$-norm of $\mathcal{F}(\mathbf{P}, \mathbf{K})$ as introduced 2. This requirement sets an upper limit for γ. Due to the general relationship in equation 3.15, this choice of nominal plant relaxes this requirement.

$$\text{Max}(\delta_\nu(P_{\text{clean}}, P_{\text{nominal}}), \delta_\nu(P_{\text{nominal}}, P_{\text{iced}})) \leq \delta_\nu(P_{\text{clean}}, P_{\text{iced}}) \tag{3.15}$$

If the requirement in equation 3.14 is full-filled, in addition to the winding number condition in equation 2.20, it follows that the controller will stabilize both the clean and iced plant.

The closed-loop robustness requirements are chosen based on Astrom and Murray [14], which states that reasonable closed-loop values for margins are

- Minimum phase margin $\phi_m = 30°$
- Minimum gain margin $g_m = 2$
- Minimum stability margin $s_m = 0.5$

Here, the stability margin is the smallest distance from the Nyquist contour of a plant to the critical point at $-1 + 0\text{i}$. For SISO systems, this is equivalent to the maximum peak of the sensitivity function $S(s)$.

The performance are measured in terms of maximum closed-loop overshoot and settling time when the system is subject to a step in reference signal. Acceptable value of settling time and overshoot is are dependant on the on the controlled variable. Hence, they will be mentioned at each specific controller implementation. Since the controller goal is to have satisfactory performance for different types of plants, there will always be some sort of trade-of between performances for different plants. The tuning method that is used to find the

Longitudinal Robust Controller

The first step is to find the nominal longitudinal plant which will be used for implementing the $\mathcal{H}_\infty$ controller. This is found by iterating through the icing level ζ, and finding $P_{\text{nominal}}(\zeta)$ that minimizes $\text{Max}(\delta_\nu(P_{\text{clean}}, P_{\text{nominal}}), \delta_\nu(P_{\text{nominal}}, P_{\text{iced}}))$. This gives $\zeta = 0.6500$ and $\text{Max}(\delta_\nu(P_{\text{clean}}, P_{\text{nominal}}), \delta_\nu(P_{\text{nominal}}, P_{\text{iced}})) = 0.1929$.

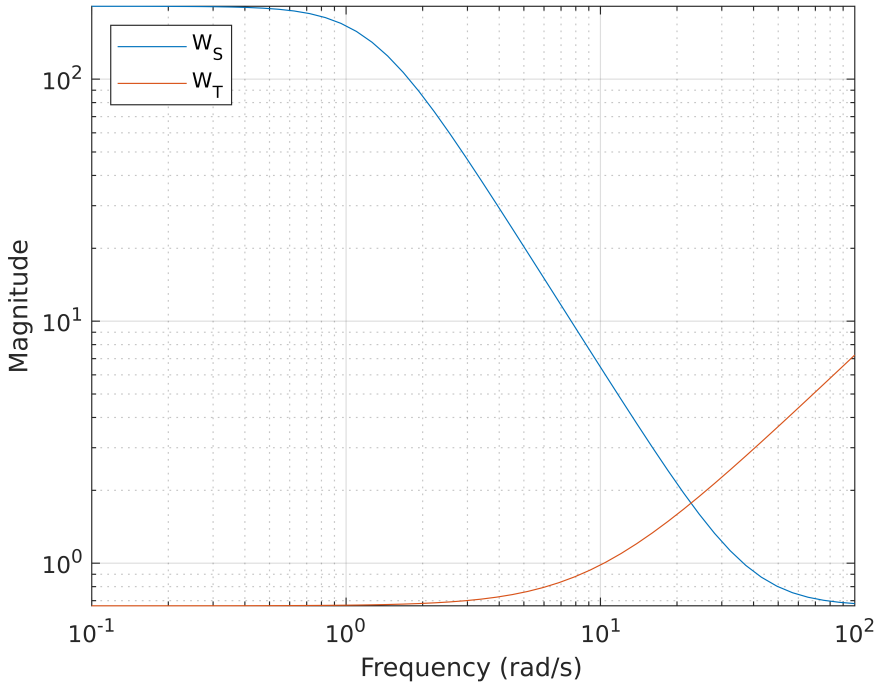
By using the nominal plant as reference plant for tuning the H_∞ - controller, guarantee stability requirements for iced and clean case are met if the $\mathcal{H}_\infty$-controller achieved $\gamma_{\text{lon}^*} < \frac{1}{0.1929} = 5.1840$. The controller is tuned by in an iter-

ative process by increasing the bandwidth while the closed loop requirements in previously mentioned is met.

The weighting filters parameters needs to be chosen. the constant A from weighting filters in equation 3.13, decides the gain at low frequencies, and therefore decides the maximum allowed steady state offset. Different values were tested, and the final value is chosen equal to 0.001. M limits the maximum peak of the sensitivity function $S(s)$. Since the stability margin s_m is the given as the maximum peak of $S(s)$ for SISO systems, the M in equation 3.13 can be chosen based the minimum stability margin $s_m = 0.5$. Here, $M = 2$ is chosen. The value $W_C = 8.4$ is finally chosen to avoid oscillatory response as previously discussed, and to get the satisfactory performance for both clean and iced case.

The filter frequency response are shown in figure 3.4. The initial value for ω_b is 1 rad/s. The final value for ω_b given the chosen parameters, and where all requirements are met is 13.56 rad/s. The achieved γ^*_{lon} - value is 5.1748. The final controller values are summarized in table 3.10.

Figure 3.4: Frequency response of longitudinal the tuning filters W_S and W_T



The tuning procedure that is used can be summarized in the following steps;

- Find the icing level ζ of the nominal plant which minimizes $\text{Max}(\delta_\nu(P_{\text{clean}}, P_{\text{nominal}}), \delta_\nu(P_{\text{nominal}}, P_{\text{iced}}))$, and use this value to set the stability requirement.
- Choose appropriate weighting filter constants A and M based on the desired

low frequency gain and stability margin s_m.

- Set a suitable starting point for W_C and ω_b. Increase ω_b until either the stability requirement or robustness requirement for iced and clean case are no longer satisfied.
- Repeat the last step by adjusting W_C such that good performances are achieved for both the iced and clean case.

Table 3.10: Longitudinal Robust Controller Summary

Performance parameter	Value
ω_b	13.56 rad/s
γ^*_{lon}	5.1748
Filter parameter	Value
M	2
A	0.001
W_C	8.4

Lateral Robust Control

A similar method tuning method is used in the lateral case. $P_{\text{nominal}}(\zeta)$ is found. $\text{Max}(\delta_\nu(P_{\text{clean}}, P_{\text{nominal}}), \delta_\nu(P_{\text{nominal}}, P_{\text{iced}})) = 0.2433$ at $\zeta = 0.5700$. Hence, the lateral stability requirements for the H_∞ controller is met if $\gamma^* < \frac{1}{0.2433} = 4.110$. $A = 0.005$ and $M = 2$ are chosen for the same reasoning as in the longitudinal case. The initial value for ω_b is 0.1. $W_C = 0.3$ is chosen by doing the iteration as before. The largest value for ω_b given the chosen parameters, and where all requirements are met is 0.4150 rad/s. The controller is summarized in table 3.11. The filter frequency response are shown in figure 3.5. While achieved γ^*_{lat}-value is 2.0371. Here, it is the general stability margin $s_m > 0.5$ for the iced case that stops the iteration.

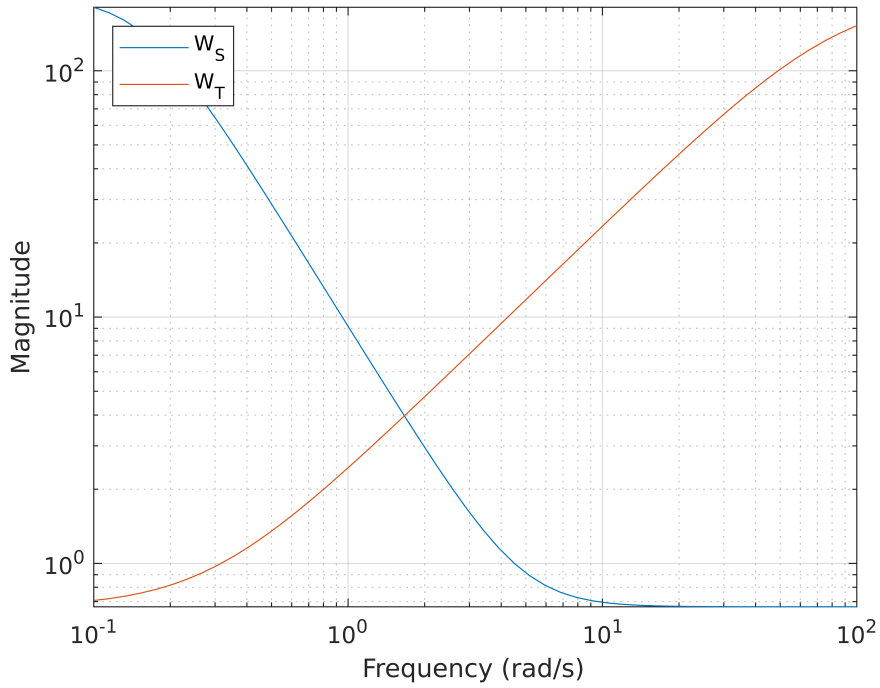
Table 3.11: Lateral Robust Controller Summary

Performance parameter	Value
ω_b	0.4150 rad/s
γ^*_{lat}	2.0371
Filter parameter	Value
M	2
A	0.005
W_C	0.3

3.7.3 Gain Scheduled Controller

The gain scheduled approach is using the exact same tuning method as above. Here, a set of $\mathcal{H}_\infty$ controllers are made for using different levels of icing as nom-

Figure 3.5: Frequency response of lateral the tuning filters W_S and W_T



inal plants. Generally, this gives a much lower ν-gap between the the plants that is used for the controller synthesis and the plants that needs to be stabilized, which in turn relaxes the stabilizing γ-requirement. Here, it is chosen to use the ζ-values of 0.1, 0.3, 0.5, 0.7 and 0.9 as nominal plants, where the controller selected if the icing level is $\zeta \in [\zeta_{\text{nominal}} - 0.1, \text{nominal} + 0.1)$. The results presented in 4 are simulations made for ζ-values 0, 0.3 and 1. Hence these controllers are summarized in the tables below. Lateral controllers for ζ-values 0, 0.3 and 1 are summarized in table 3.12. Here, it was the robustness requirement of $s_m > 0.5$ which was the limiting factor when tuning. Longitudinal controllers for ζ-values 0, 0.3 and 1 are summarized in tables 3.13. Here it was the maximum frequency of 2.5 Hz to preserve general noise attenuation capabilities which was the limiting factor when tuning.

Table 3.12: Lateral gain scheduled controllers with $\zeta = 0.1$, 0.3 and 0.9

Performance parameters	$\zeta = 0.1$	$\zeta = 0.3$	$\zeta = 0.9$
ω_b	0.7150 rad/s	0.6500 rad/s	0.3800
γ^*_{lat}	2.1514	2.2393	2.3184
Filter parameters			
M	2	2	2
A	0.001	0.001	0.001
W_C	0.3	0.3	0.3

Table 3.13: Longitudinal gain scheduled controllers with icing level value ζ

Performance parameters	$\zeta = 0.1$	$\zeta = 0.3$	$\zeta = 0.9$
ω_b	15.7 rad/s	15.7 rad/s	15.7
γ^*_{lat}	2.54	5.2861	6.6221
Filter parameters			
M	2	2	2
A	0.001	0.001	0.0003
W_C	1.3	6.2	14.6

Chapter 4

Results

The simulation results are represented by looking at the controlled variables responses from a step in commanded roll angle, ϕ_c, and commanded pitch angle, θ_c for each of the controllers in 3. Each case is simulated with no ice ($\zeta = 0$), some ice ($\zeta = 0.3$) and full ice ($\zeta = 1$). The steps in ϕ_c, θ_c are of size 4 °. In each simulation, one control variable is tested per simulation. The step occurs after 10s, and a negative step of the same magnitude is applied after 60 seconds. The total simulation time is 90 seconds. In other words, the testing sequence is a step at 10s, hold the angle for 50 seconds, then a step back to the initial value and hold this angle for the last 30s. The initial values are the calculated trimmed values from section 3.2

4.1 $\mathcal{H}_\infty$ Robust Controller

This section will present the simulation results for the roll and pitch testing sequences using the $\mathcal{H}_\infty$ Robust Controller.

4.1.1 Roll testing sequence

The H_∞ robust controller tuning parameters are summarized in tables 3.11 and 3.10 for the lateral and longitudinal respectively. The roll angle testing sequence simulation results are shown in figure 4.1. The variables presented for this test sequence is roll angle, pitch angle, aileron deflection and elevator deflection. Figure 4.1 a) shows the roll angle response. The settling time (measured from the step) increases as with the ν-gap distance from the nominal plant. For the worst-case scenario of icing, the settling time is 21.31 seconds, with no overshoot. However, there is a small, steady-state offset, but this is less than the acceptable error band of ±2% of the final step value. For the clean case, the settling time is about 18.8s. The clean response has an overshoot of 26.5%. For the icing level $\zeta = 0.3$, the settling time is 13.9s, which is faster slower than for the clean case. There is an overshoot of 11.5 %. The undershoot caused by the RHP zero is also increasing with ice. The undershoot of icing levels $\zeta = 0$, $\zeta = 0.3$ and $\zeta = 0.9$ are 24.7%,

27.3%, and 45.3 % respectively. Figure 4.1 b) shows the aileron deflection. The control effort is increased with ice. In other words, a larger aileron deflection is needed to hold the reference roll in icing conditions. Figure 4.1 c) shows how the pitch angle deflects when a step in roll is applied. The deviation from the steady angle in the iced case is not as large as in the clean case. The pitch angle has a steady state offset from the trimmed-value, which increases with ice. Figure 4.1 d) shows the elevator deflection during the simulation. The required elevator deflection to fly in level-flight with constant pitch decreases for a higher level of icing.

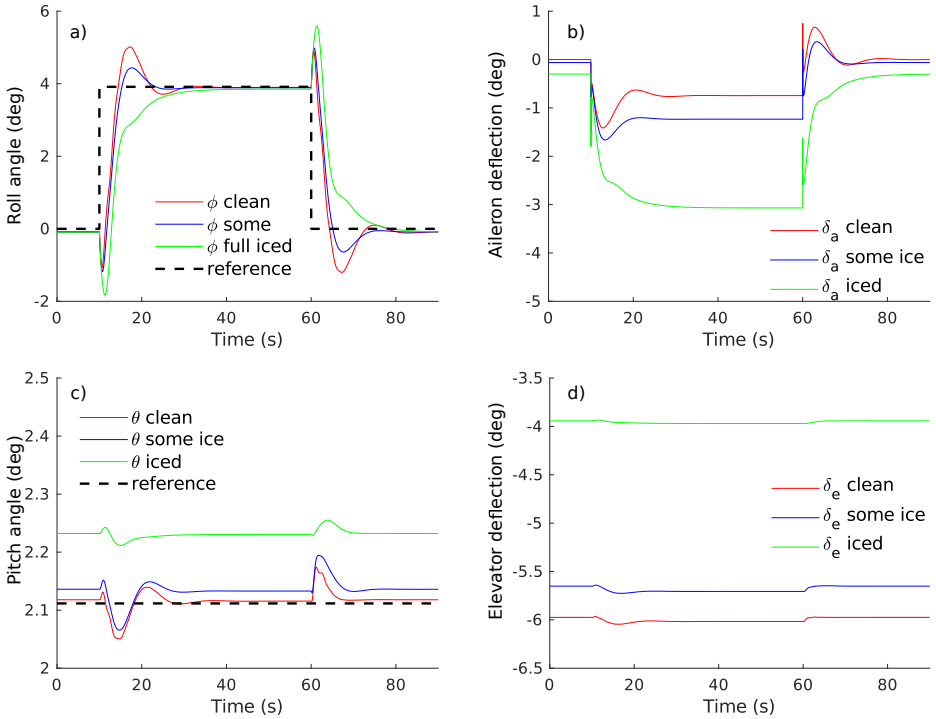


Figure 4.1: Shows the robust controller response to a step in commanded roll angle ϕ_c. a) shows the roll angle. b) shows the aileron deflection. c) shows the pitch angle and d) shows the elevator deflection.

4.1.2 Pitch testing sequence

The simulation results for the testing sequence of pitch angle using the robust controller are shown in figure 4.2. The variables presented here are pitch angle, roll angle airspeed, elevator deflection, aileron deflection and throttle. The pitch angle are shown in figure 4.2 a). The settling time is quite similar for different levels of ice. For the iced case with $\zeta = 1$, there is a steady-state offset of 0.12° as previously mentioned. For the worst-case scenario of icing, there is also some quick oscillations before it settles. The response settles at this value after 9.8s after

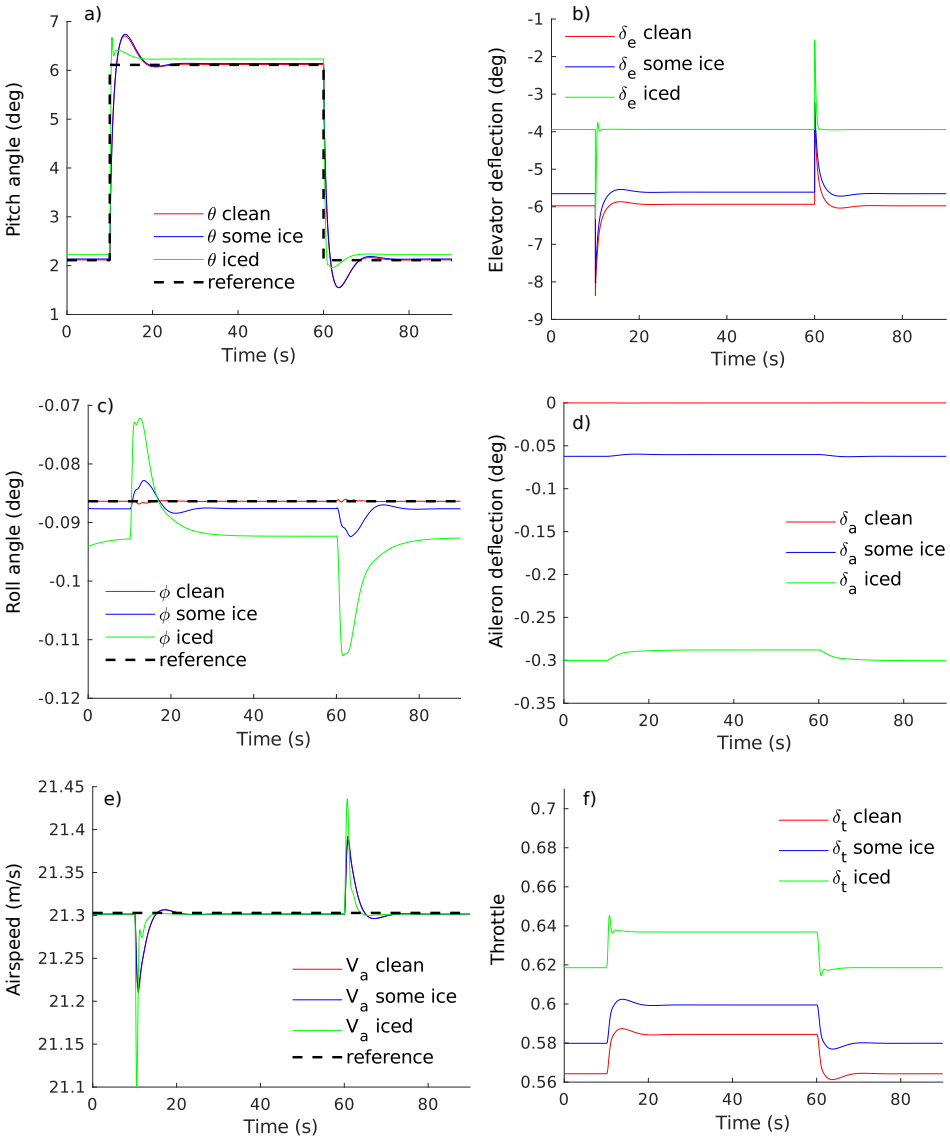


Figure 4.2: Shows the robust controller response to a step in commanded pitch angle θ_c. a) shows the pitch angle. b) shows the elevator deflection. c) shows the roll angle and d) shows the aileron deflection. e) shows the airspeed. f) shows the throttle

the step. The responses for $\zeta = 0$ and $\zeta = 0.3$ are quite similar. The overshoots are both 10.7 %. The settling times are 7.56s and 7.8s for $\zeta = 0$ and $\zeta = 0.3$ respectively. Figure 4.2 b) shows the elevator deflection. On the iced case, there is an oscillatory behaviour on the step after 10s, as described for the pitch angle. 4.2 c) shows the roll angle. This emphasize the previously mentioned statement about longitudinal and lateral coupling for aggressive manoeuvres. 4.2 d) shows the aileron input. The airspeed response is shown in figure 4.2 e). As mentioned

earlier, the airspeed and pitch angle variables are strongly related. This is shown here, where an increase in pitch leads to a drop in airspeed. 4.2 f) shows the throttle. This increases with icing as a result of the increased drag.

4.2 Gain Scheduled Controller

This section will present the simulation results for the roll and pitch testing sequences using the gain scheduled controller.

4.2.1 Roll testing sequence

The gain scheduled controller tuning parameters for the lateral and longitudinal controllers is summarized in table 3.12 and 3.13 respectively. The roll angle testing sequence simulation results are shown in figure 4.3. Figure 4.3 a) shows the roll angle. Here, the iced cases are considerably better compared to the robust controller. For icing level $\zeta = 0.3$, the settling time is 4.9s with the previously used acceptable error-band of ±2% of the final step value. The overshoot for this case is 2.6%. For icing level $\zeta = 1$, the settling time is 9.8s without any overshoot. For the clean case, the settling time is 9.36s with an overshoot of 5.3%. The undershoot of icing levels $\zeta = 0$, $\zeta = 0.3$ and $\zeta 0.9$ are 27.3%, 39.7%, and 47.2 % respectively. More icing moves the RHP zero closer to 0, meaning that the zero gets slower, which again will give a stricter limitation on controller bandwidth. This can be seen in table 3.12, where the achieved bandwidth for the gain scheduled controllers are decreasing when ζ is increasing. This can also be seen in figure 4.3 a), where the rise-time of the step response for $\zeta = 1$ is slower than the other two curves. figure 4.3 b) shows the aileron deflection. Here, it can be seen that the control signal is a more aggressive than in the single robust $\mathcal{H}_\infty$ controller. This can also be seen in figures 4.3 c) and d), which shows a greater dependency between the pitch angle and the steps in roll angle. The magnitude of these effects is still less than 0.1 °, which is a small deviation.

4.2.2 Pitch testing sequence

The simulation results are shown in figure 4.3. Figure 4.3 a) shows the roll angle. Here, the offset from the robust controller is effectively eliminated by reducing A for the worst-case icing scenario. The responses are now quite similar for all levels of icing. The maximum overshoot is 9.67 %, and worst-case settling time is 7.16s with the error bandwidth of 2%. Figure 4.3 b) shows that as a result of reducing the steady-state error in pitch angle, the magnitude of elevator deflection in the iced case is increased. It is still smaller than in the clean case. The responses for roll angle, aileron deflection, airspeed and throttle in figures **??** c), d), e) and f) respectively, are quite similar to the response for the robust control design. Small improvements still occur, such as the roll angle steady-state offset is reduced.

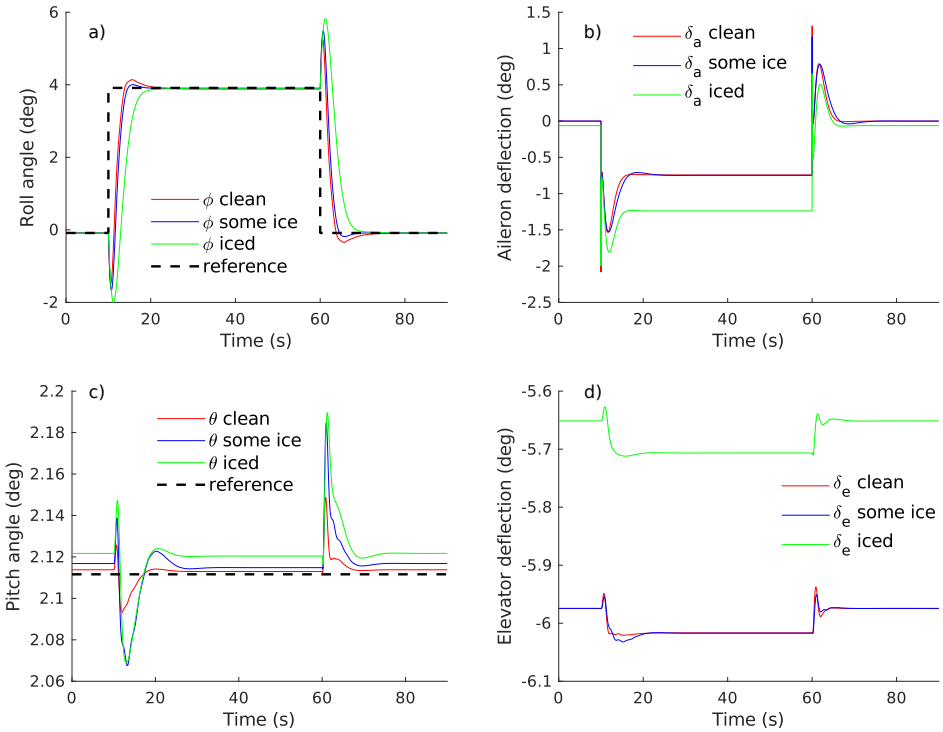


Figure 4.3: Shows the gain scheduled controller response to a step in commanded roll angle ϕ_c. a) shows the roll angle. b) shows the aileron deflection. c) shows the pitch angle and d) shows the elevator deflection.

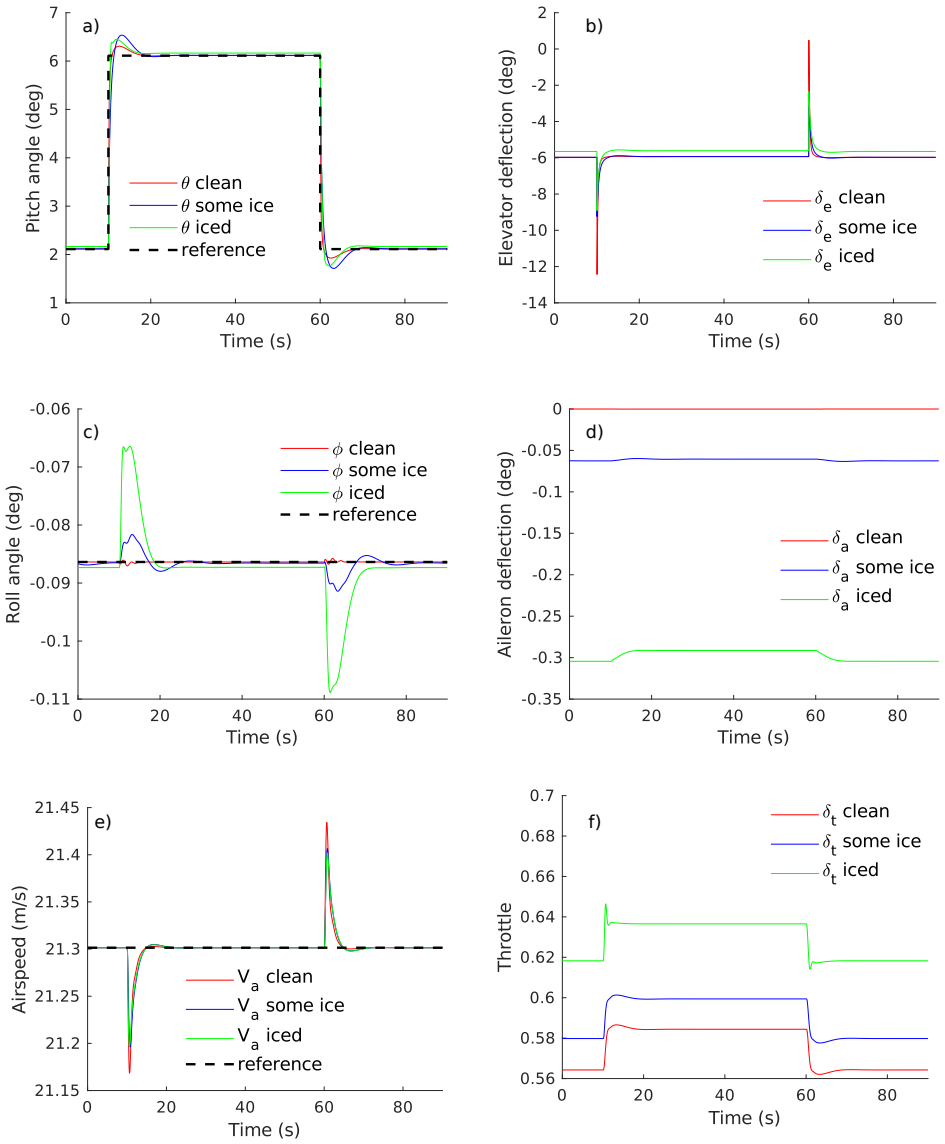


Figure 4.4: Shows the gain scheduled controller response to a step in commanded pitch angle θ_c. a) shows the pitch angle. b) shows the elevator deflection. c) shows the roll angle and d) shows the aileron deflection. e) shows the airspeed. f) shows the throttle

Chapter 5

Discussion

The results showed the performances of the implemented controllers. This section will discuss some of the results and look at the challenges and limitations of the implemented methods.

5.1 Tuning Approach

As mentioned in chapter 3, when designing the tuning filters for the $\mathcal{H}_\infty$-controller is there are trade-offs that need to be made. The fact that the controller needs to take the performance and robustness of different plants into account make the tuning problem complex. Here, there is not only the general trade-off between performance and robustness for tuning filters for a single plant but also a trade-off between performances of different plants for different levels of icing, $\zeta \in [0, 1]$. A mathematical model is always an approximation of the real system, and hence all the plants should have a minimum level of robustness. The tuning approach that was used here was that an absolute lower limit of closed-loop robustness and stabilization requirements was set, and then the performance of the plants of clean and worst-case ice was improved through an iterative process when the controller bandwidth was increased, and the control signals limitations were adjusted. There are several factors that are not included in the icing model [17]. This includes surface quality of the wings, thermal conductivity, the structural composition of the wing, etc. This emphasizes the importance of robustness to model uncertainties. The robustness requirements were chosen based on the literature, and not compared to the magnificence of unmodeled dynamics in this specific model.

5.2 Longitudinal and Lateral Decoupling

The controller method implemented this based on a linearized model where the longitudinal and lateral dynamics are decoupled. The results show that there is a weak coupling, which is illustrated by investigating the roll angle for a step in pitch angle and the pitch angle for a step in roll angle. The results indicate that

this coupling increases for more aggressive manoeuvres. Note that during this test sequence, the coupling is quite weak (small deviations in reference values to the investigated angle variables), but it illustrates the general concept that increased longitudinal and lateral coupling for aggressive manoeuvres.

5.3 Controlling Right Half Plane Zeros

The lateral dynamics has an RHP zero. This is illustrated in the simulation result with the characteristic non-minimum phase response to a step input. As previously mentioned, this gives fundamental limits to achievable performance and robustness. As seen in the results, the RHP zero is moving closer as the icing level is increasing, which gives stricter limitations to performance and robustness. The performance measures in this thesis are primarily settling time and overshoot to step response. Hence, the attempt to minimize the undesirable non-minimum phase undershoot has not been a control objective. However, the usual method of reducing the undershoot is to reduce the bandwidth, which again reduces the rise time and increases the settling time.

5.4 Single Robust Controller

There are some drawbacks by using the single robust controller for plants with large uncertainties. Generally, this method assumes the worst-case scenario at all times, which again gives stricter robustness requirements and worse performance than necessary. This is best illustrated by comparing the performance of the roll test sequence for the single robust and the gain scheduled controllers. In a scenario where the icing is absent, or there is low level of icing during the entire flight, the performance is using this method could have been a lot better by using a different method. On the other hand, this quite simple method that gives a guarantee of stability during the entire flight while being in the region of the trim point, without needing any information of the actual level of icing.

The effect of icing is a function with AOA. The trimmed point used during the simulations shown in section 3.2, has a quite low AOA. By increasing the AOA, the maximum Vinnicombe distance between the new linearized plants for the clean and iced case is likely to change, and therefore change stability requirement for the controller tuning method. If the Vinnicombe distance gets to large, it will make it difficult/impossible to achieve satisfactory results in terms of robustness and performance[14].

5.5 Gain Scheduled Controller

The gain scheduled controller improves the performance compared to the single robust controller. During the simulations, the icing level is assumed known. The

main drawback of this controller is that it requires a method to measure the level and type of ice. In general, this is a quite difficult quantity to measure.

Chapter 6

Conclusion

This thesis has investigated the control of an in-flight UAV that is subject to icing conditions. Two different control techniques were used to create an inner-loop controller that ensures stability and robustness to model uncertainty in order to mitigate the icing effects in the UAV. The results shows that both the single robust controller and the gain scheduled controller gave satisfactory results in terms of robustness and stability. The results also shows that by increasing the available information available and assuming that the level of icing is known, the performance in terms of settling time and overshoot can be significantly improved by using a gain scheduled controller and using the icing signal as selector variable.

Bibliography

[1] R. W. Gent, N. P. Dart and J. T. Cansdale, 'Aircraft icing', *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 358, no. 1776, pp. 2873–2911, 2000.

[2] T. Cebeci and F. Kafyeke, 'Aircraft icing', *Annual review of fluid mechanics*, vol. 35, no. 1, pp. 11–21, 2003.

[3] R. Hann, A. Wenz, K. Gryte and T. A. Johansen, 'Impact of atmospheric icing on uav aerodynamic performance', in *2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*, 2017, pp. 66–71. DOI: 10.1109/RED-UAS.2017.8101645.

[4] S. K. Thomas, R. P. Cassoni and C. D. MacArthur, 'Aircraft anti-icing and deicing techniques and modeling', *Journal of aircraft*, vol. 33, no. 5, pp. 841–854, 1996.

[5] K. Gryte, R. Hann, M. Alam, J. Rohac, T. A. Johansen and T. I. Fossen, 'Aerodynamic modeling of the Skywalker X8 Fixed-Wing Unmanned Aerial Vehicle', *2018 International Conference on Unmanned Aircraft Systems, ICUAS 2018*, pp. 826–835, 2018. DOI: 10.1109/ICUAS.2018.8453370.

[6] A. Winter, R. Hann, A. Wenz, K. Gryte and T. A. Johansen, 'Stability of a flying wing uav in icing conditions', in *8th European Conference for Aeronautics and Space Sciences (EUCASS), Madrid*, 2019.

[7] J.-F. Magni, S. Bennani and J. Terlouw, *Robust flight control: a design challenge*. Springer, 1997, vol. 110.

[8] R. W. Beard and T. W. McLain, *Small unmanned aircraft: Theory and practice*. Princeton university press, 2012.

[9] K. Glover and J. C. Doyle, 'State-space formulae for all stabilizing controllers that satisfy an h-norm bound and relations to relations to risk sensitivity', *Systems Control Letters*, vol. 11, no. 3, pp. 167–172, 1988, ISSN: 0167-6911. DOI: https://doi.org/10.1016/0167-6911(88)90055-2. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0167691188900552.

[10] P. Lundstrom, S. Skogstad and Z. Wang, 'Weight selection for h-infinity and mu-control methods-insights and examples from process control', in *Symposium on Robust control System Design using H infinity and related methods*, 1991.

[11] J. Doyle, K. Lenz and A. Packard, 'Design examples using μ-synthesis: Space shuttle lateral axis fcs during reentry', in *Modelling, Robustness and Sensitivity Reduction in Control Systems*, Springer, 1987, pp. 127–154.

[12] E. Lavretsky and K. A. Wise, 'Robust adaptive control', in *Robust and adaptive control*, Springer, 2013, pp. 317–353.

[13] G. Vinnicombe, *Uncertainty and Feedback: H [infinity] Loop-shaping and the [nu]-gap Metric*. World Scientific, 2001.

[14] K. J. Åström and R. M. Murray, *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.

[15] K. J. Åström, 'Adaptive control', in *Mathematical System Theory: The Influence of R. E. Kalman*, A. C. Antoulas, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, pp. 437–450, ISBN: 978-3-662-08546-2. DOI: 10.1007/978-3-662-08546-2_24. [Online]. Available: https://doi.org/10.1007/978-3-662-08546-2_24.

[16] K. Gryte, 'High angle of attack landing of an unmanned aerial vehicle', Master's thesis, NTNU, 2015.

[17] A. Winter, R. Hann, A. Wenz and T. A. Johansen, 'Stability of a Flying Wing UAV in Icing Conditions', *8th European Conference for Aeronautics and Space Sciences (EUCASS)*, 2019. DOI: 10.13009/EUCASS2019-906.

[18] K. Gryte, 'Precision control of fixed-wing uav and robust navigation in gnss-denied environments', 2020.

[19] L. Faleiro and A. Lambregts, 'Analysis and tuning of a 'total energy control system'control law using eigenstructure assignment', *Aerospace science and technology*, vol. 3, no. 3, pp. 127–140, 1999.