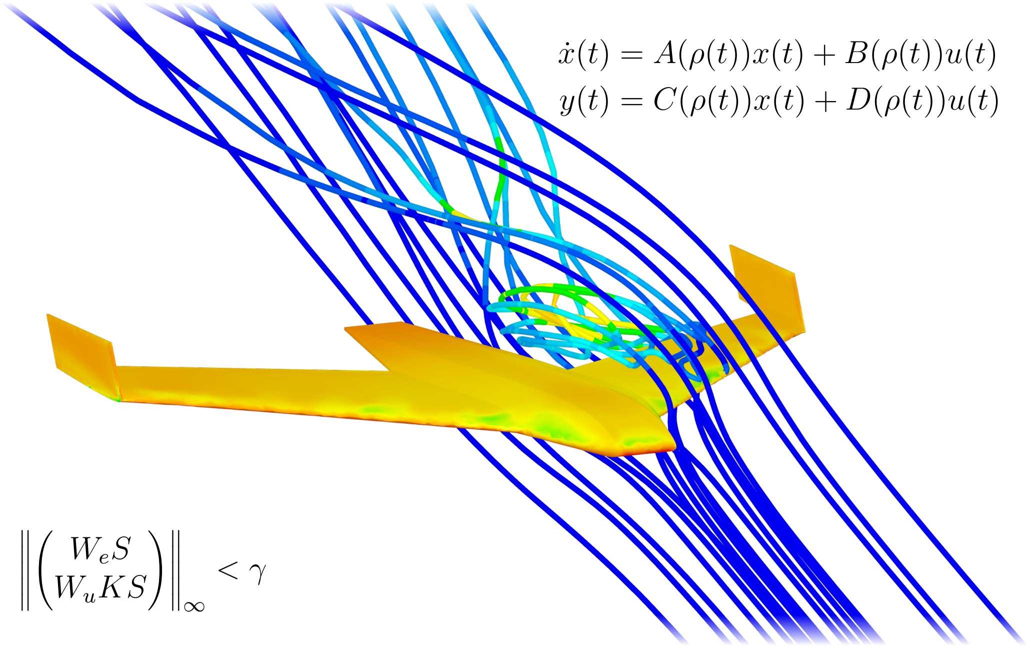
CHALMERS
UNIVERSITY OF TECHNOLOGY



Model-Based Robust Control of an Ultralight Fixed-Wing Tailsitter

Aerodynamic Modelling, LPV Control Synthesis and Simulation of a VTOL UAV

Master's thesis in Systems, Control and Mechatronics

Gunnar Edman
Robin Ohlsson

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

Master 's thesis 2026

Model-Based Robust Control of an Ultralight Fixed-Wing Tailsitter

Aerodynamic Modelling, LPV Control Synthesis and Simulation of a VTOL UAV

Gunnar Edman
Robin Ohlsson

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2026

Model-Based Robust Control of an Ultralight Fixed-Wing Tailsitter
Aerodynamic Modelling, LPV Control Synthesis and Simulation of a VTOL UAV
Gunnar Edman
Robin Ohlsson

Supervisor: Fredrik Falkman, Remote aero
Examiner: Balázs Kulcsár, Department of Electrical Engineering

Master's Thesis 2026
Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Illustration showing CFD particle trace over Tailsitter drone, LPV equations and H_∞ mixed sensitivity optimization criterion.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2026

Model-Based Robust Control of an Ultralight Fixed-Wing Tailsitter
Aerodynamic Modelling, LPV Control Synthesis and Simulation of a VTOL UAV
Gunnar Edman, Robin Ohlsson
Department of Electrical Engineering
Chalmers University of Technology

Abstract

Development of fixed-wing VTOL UAVs is an active area of research, enabling increased range and top-speed far surpassing conventional quad-rotors, while preserving the landing flexibility. A subset of this research is the fixed-wing tailsitter, which achieves vertical landing by pitching the entire airframe up into hover. This thesis investigates model-based robust control of an ultralight 250 gram tailsitter UAV which aims to autonomously land vertically in real-world conditions. A 6-DOF model adapted for the unique challenges of tailsitters at hover was developed, extracting aerodynamic coefficients through CFD simulation. By incorporating propwash dynamics, the model ensures control surfaces maintain authority as airspeed approaches zero, enabling continuous simulation from cruise to hover. To account for the significant nonlinearities during transition, an LPV model was developed, scheduled over a grid of airspeed and angle of attack to cover the flight envelope. From this, a MIMO H_∞ controller was developed to stabilize the vehicle's coupled dynamics. Simulation results demonstrate successful trajectory tracking through transition and descent, showcasing that the LPV captures the nonlinear dynamics. Furthermore, wind gust simulations utilizing the Dryden wind model were used to evaluate the operational limits of the aircraft. These tests revealed that the gain-scheduled LQR baseline demonstrated higher overall performance and superior disturbance rejection. Ultimately, the results indicate the aircraft can only handle low to moderate wind conditions, which is attributed to its low inertia and control authority.

Keywords: Tailsitter, VTOL, LPV, H_∞, CFD, Robust Control, Gain Scheduling

Acknowledgements

We would like to first show sincere gratitude to our examiner Balázs Kulcsár, Full Professor at the department of Systems and Control for his continuous guidance, insightful feedback and encouragement. We also thank Fredrik Falkman for the opportunity, faith and academic freedom to pursue this work.

Gunnar Edman & Robin Ohlsson, Gothenburg, June 2026

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
ESC	Electronic Speed Controller
FF	Feedforward
FRD	Forward Right Down
GNSS	Global Navigation Satellite Systems
LHP	Left Half Plane
LPV	Linear Parameter Varying
LTI	Linear time invariant
LQR	Linear Quadratic Regulator
NED	North East Down (inertial frame)
NL	Nonlinear
PBH	Popov-Belevitch-Hautus
RHP	Right Half Plane
SS	Steady-state
UAV	Unmanned Aerial Vehicle
VTOL	Vertical Take-Off and Landing

Nomenclature

Below is the nomenclature of indices, sets, parameters, and variables that have been used throughout this thesis.

V_b	Body-frame velocity vector [m/s]
V_a	Airspeed [m/s]
q_x, q_y, q_z	Roll, Pitch, Yaw [°]
p, q, r	Body-frame angular rates [rad/s]
α, β	Angle of attack, sideslip angle [°]
δ	Elevon deflection angle [°]
C_k	Aerodynamic force/moment coefficients
$k \in$	$\{D, Y, L, l, m, n\}$
ρ	Scheduling parameters or air density (context-dependent)
$K(\rho)$	Gain-scheduled LQR controller
W_e, W_u, W_p, W_{oM}	H_∞ weighting functions
γ	H_∞ performance bound
η	Coefficient blending function
g	Gravitational acceleration (9.81 m/s^2)

Contents

List of Figures

List of Tables

1

Introduction

Every year the Swedish Sea Rescue Society receives approximately 13000 emergency calls and about 1200 of them are classified as life-dependent [1]. This thesis is to help improve the project called *Eye in the sky*, that provides help with search and rescue missions. The project in development still has challenges that remain unsolved, this thesis is to investigate one of them, namely the landing procedure.

1.1 Background

Remote Aero has developed an ultralight tailsitter drone seen in Fig. 1.1, to help sea-rescue operations by providing visual data. Currently, the drone is outfitted with two rotors, two elevons and an ArduPilot flight computer board which includes a microcontroller, IMU, barometer, airspeed sensor and two ESCs. It is also equipped with a Raspberry Pi together with a GNSS antenna for navigation and a gimbal camera for visual tasks. At 250 grams, 0.6 m wingspan with limited control inputs, the platform is under-actuated and sensitive to wind disturbances. Today the landing procedure is done in a conventional aircraft configuration requiring open areas which limits landing flexibility and forces manual retrieval further from the launch pad.

The current implementation utilizes the default implementation in ArduPilot [2], consisting of PID controllers tuned for only hover and forward flight. The transition is handled by blending these two controllers when transitioning, which entirely ignores the nonlinearities of the transition dynamics.

---

[1] Sjöräddningssällskapet, “Sjöräddningssällskapets Årsbok 2025,” *Sjöräddningssällskapet*, 2025. [Online]. Available: [Online PDF], [Accessed: Jun. 7, 2026].

[2] ArduPilot Project, “ArduPilot Autopilot Software,” 2026. Available: [Online].

Figure 1.1: Fixed-wing tailsitter UAV developed by Remote Aero

1.2 Purpose

The main task is to investigate the landing procedure by developing a system that enables robust precision landing. Vertical take-off and landing (VTOL) provides the optimal landing procedure, where the airframe is tilted into the hover orientation for a controlled descent. As this specific UAV is 250 grams, that is a challenge in windy conditions. The primary purpose is to investigate the landing procedure by performing transitions and disturbance rejection. Additionally this expands the operational domains of the system by eliminating the need for a large open area, minimizing wear and especially damage.

1.3 Objective

The objective of the thesis is to enable robust, autonomous precision landing of the existing fixed-wing drone by implementing new control algorithms. The thesis aims to answer the following research questions:

RQ1: Can a model-based, robust controller stabilize the tailsitter through transition from forward flight to hover and stabilize it in hover?

RQ2: What are the operational limits of the implemented controllers in terms of static and gusty wind conditions?

1.4 Limitations

This thesis is limited to simulation only. No flight data was provided meaning the model is limited to approximations and therefore, no thorough validation of the nonlinear model was performed. The aerodynamic model is derived from CFD simulations and not verified against wind tunnel tests or flight telemetry. Full state feedback and ideal sensors are assumed throughout the entire thesis.

1.5 Method

The main focus of the project will be model-based control. The work consists of four main parts.

- Developing a 6-DOF model
- Obtaining aerodynamic coefficients using CFD simulation and capturing the slipstream effects
- Development of a Linear-Parameter-Varying (LPV) model by linearizing a grid of trimpoints with appropriate scheduling parameters.
- Synthesising two controllers, a robust H_∞ LPV controller and a gain-scheduled LQR.

The designed controllers will be evaluated in simulation, comparing performance and disturbance rejection throughout the flight envelope from forward flight to hover.

1.6 Ethics and security

Current regulations dictate that drones remain under 250 grams to avoid pilot license requirements. Respecting this strict weight limit is crucial for the project, as Remote Aero is currently working to obtain special permits for Beyond Visual Line of Sight (BVLOS) operations. The main advantage of a tailsitter drone over a conventional quadrotor is that it generates lift using its wings in forward flight, making it far more energy efficient. This extends the operational time and range significantly, enabling missions further out at sea. Investigating a robust landing procedure is a necessary step toward eventually making the tailsitter drone a practical tool for search and rescue operations.

1.7 Thesis outline

The thesis starts with Chapter 2 where relevant theory is presented. The first parts cover coordinate frames, orientations, aerodynamic coefficients and computational fluid dynamics. The second half of theory outlines control theory, LPV and robust control. Chapter 3 Method begins on the modelling and CFD analysis. It then goes into LPV and control synthesis such as the weight tuning. This chapter also explains the simulation setup.

The results are presented in Chapter 4 that shows flight transitions, disturbance rejection and a comprehensive comparison between two controllers, a gain scheduled LQR with feedforward and the H_∞ LPV. Analysis of the results and the reasoning behind them are discussed in Chapter 5, its focus is on why. A comprehensive conclusion of the thesis work in written in Chapter 6 to tie everything together. This is completed with a section on future work where multiple points are brought up about our work and the current VTOL.

2

Theory

2.1 Modelling

2.1.1 Quaternions and Kinematics

Quaternions are widely used in aerospace applications to avoid gimbal lock, which is a kinematic singularity where axis align in the Euler angle represenation and one degree of freedom is lost. A unit quaternion describes 3D orientation in a 4D space and is defined as the state vector:

$$\mathbf{q} = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix}^T \tag{2.1}$$

This is thanks to the unit-norm constraint for quaternions $\sqrt{q_w^2 + q_x^2 + q_y^2 + q_z^2} = 1$. More specifically q_w is representing the radial distance in a 4D sphere rather than an orientation degree of freedom. The orientation of a body frame relative to an inertial frame is described by a rotation matrix (Direction Cosine Matrix) $\mathbf{R}$, which is mapped directly from the quaternion elements.
The rate of change of the quaternion, driven by the angular velocity $\boldsymbol{\omega}$, is:

$$\dot{\mathbf{q}} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega})\mathbf{q} \tag{2.2}$$

where $\boldsymbol{\Omega}(\boldsymbol{\omega})$ is a skew-symmetric matrix of the angular rates. The body-frame velocity $\mathbf{v}$ is mapped to the inertial position rate $\dot{\mathbf{P}}$ using the rotation matrix $\mathbf{R}$:

$$\dot{\mathbf{P}} = \mathbf{R}\mathbf{v} \tag{2.3}$$

2.1.2 Newton-Euler equations

The 6-DOF dynamics of a rigid body in a rotating reference frame are governed by the Newton-Euler equations. For a body with mass m and inertia tensor $\mathbf{J}$, the translational and rotational equations of motion are:

$$\dot{\mathbf{v}} = \frac{1}{m}\mathbf{F} - \boldsymbol{\omega} \times \mathbf{v} \tag{2.4}$$

$$\dot{\boldsymbol{\omega}} = \mathbf{J}^{-1}\left(\mathbf{M} - \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega})\right) \tag{2.5}$$

where $\mathbf{F}$ and $\mathbf{M}$ are the total forces and moments, $\mathbf{v}$ is the translational velocity, and $\boldsymbol{\omega}$ is the angular velocity.

2.1.3 Angle of attack and sideslip angle

The orientation of the aircraft relative to the incoming airflow is defined by the angle of attack (α) and the sideslip angle (β), shown in Fig 2.1.

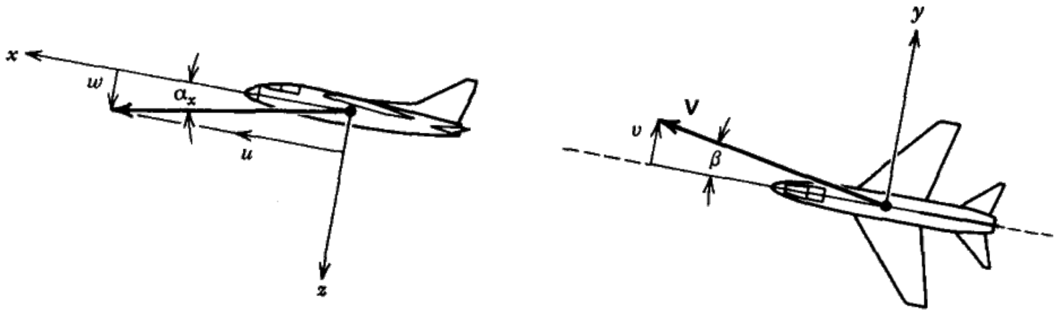


Figure 2.1: Attack and sideslip angles visualized

They are obtained, using the body-frame velocity components (u, v, w) and the total velocity magnitude V, as:

$$\alpha = \operatorname{atan2}(w, u) \qquad \beta = \arcsin\left(\frac{v}{V}\right) \tag{2.6}$$

2.1.4 Aerodynamic Coefficients

Aerodynamic forces and moments are non-dimensionalized to scale across varying flight conditions. The aerospace conventions dictate that this non-dimensionalization is performed using dynamic pressure $\bar{q}$, platform reference area S_{ref} (from above), and a reference length C_{ref} (such as the mean aerodynamic chord):

$$C_F = \frac{F}{\bar{q}S_{ref}}, \quad C_M = \frac{M}{\bar{q}S_{ref}C_{ref}} \tag{2.7}$$

The resulting coefficients are defined relative to the apparent wind vector:

- C_D, C_Y, C_L: Drag, Sideforce, and Lift force coefficients (associated with the x, y, and z axes).
- C_l, C_m, C_n: Roll, Pitch, and Yaw moment coefficients (rotations about the x, y, and z axes).

2.2 Computational Fluid Dynamics

Computational Fluid Dynamics (CFD) is a numerical method used to approximate solutions to the Navier-Stokes equations, which govern continuous fluid motion. Analytical solutions to these nonlinear differential equations are generally impossible for complex aerodynamic shapes, necessitating numerical approximations.

2.2.1 Spatial Discretization and Meshing

The fluid domain is divided into a finite number of discrete control volumes, or cells, forming a mesh. The continuous fluid equations are integrated over these cells to form a solvable system of algebraic equations. Mesh quality dictates solution accuracy and stability; a primary metric is non-orthogonality, which measures the angular deviation between the vector connecting adjacent cell centers and the normal vector of their shared face. Excessive non-orthogonality degrades the numerical discretization and induces errors.

2.2.2 Reynolds Number

The Reynolds number (Re) is a dimensionless quantity representing the ratio of inertial forces to viscous forces within a fluid flow:

$$Re = \frac{\rho V c}{\mu} \tag{2.8}$$

where ρ is fluid density, V is macroscopic velocity, c is a characteristic reference length, and μ is dynamic viscosity. It is the primary parameter for scaling fluid dynamics and determining whether flow remains laminar or becomes turbulent.

2.2.3 Boundary Layers and y^+

y+ is a dimensionless distance metric used in CFD to measure how well the mesh resolves the boundary layer adjacent to a wall. It is mathematically defined as:

$$y^+ = \frac{u_\tau y}{\nu} \tag{2.9}$$

where u_τ is the friction velocity, y is the absolute distance from the wall to the first mesh cell center, and ν is the local kinematic viscosity.

The boundary layer is divided into distinct flow regions based on this distance, as illustrated in Fig. 2.2. The y^+ value of the first mesh cell adjacent to the wall dictates the simulation strategy. Either, the mesh must directly try to capture the viscous sublayer ($y^+ \approx 1$) requiring a significant number of cells. Or, it must utilize wall functions that target the log-law region (represented by the linear "log law" curve plotted in the region where $y^+ > 30$ in Fig. 2.2) and avoid the buffer layer. This in turn reduces the mesh resolution requirements.

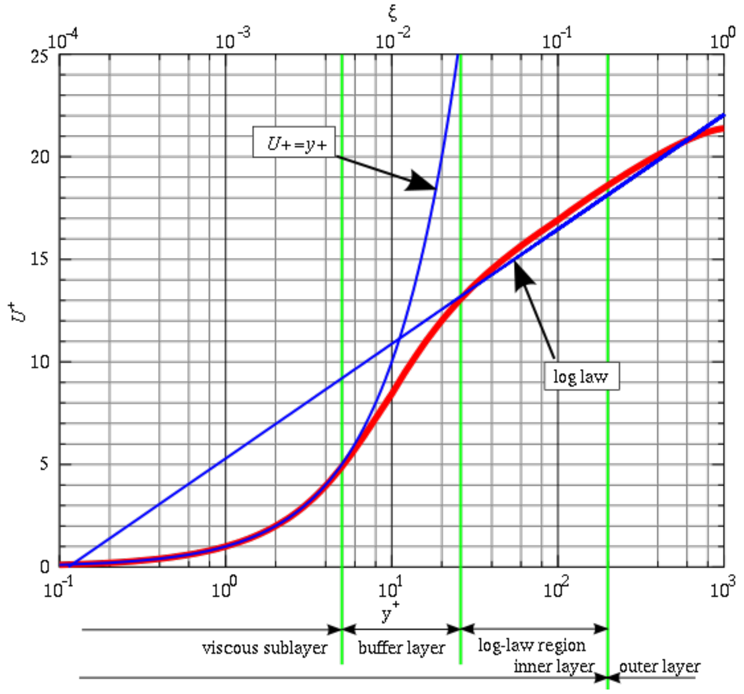


Figure 2.2: Dimensionless velocity profile U^+ as a function of y^+, highlighting the viscous sublayer, buffer layer, and log-law region. Image courtesy of SimScale.

2.2.4 Boundary Conditions

Boundary conditions mathematically constrain the computational domain, defining the fluid state at the edges of the mesh. Standard applications include defined freestream inlets, pressure outlets, and solid no-slip walls. These constraints are strictly required to yield a unique solution to the Navier-Stokes equations.

2.3 Scaling and Conditioning

One should generally normalize or scale the matrices inputs and states of a system so they are dimensionless and of comparable magnitude. This is critical because it allows for meaningful statements to be made about the system's input-output controllability and expected performance [1]. One should be careful with scaling as there is little point in scaling on the basis of making the mathematical problem well conditioned, it needs to reflect the sensitivity of the physical problem [2].

$$x = D_x x_n, \qquad u = D_u u_n \tag{2.10}$$

$$\dot{x}_n = \underbrace{D_x^{-1} A_{\text{lin}}\, D_x}_{A_n} x_n + \underbrace{D_x^{-1} B_{\text{lin}}\, D_u}_{B_n} u_n \tag{2.11}$$

The condition number κ is the relationship between the largest and smallest singular value and acts as an indicator of how well conditioned a system is.

$$\kappa(Co) = \frac{\bar{\sigma}}{\underline{\sigma}} \tag{2.12}$$

2.4 Stabilizability and PBH-test

A system is stabilizable if for all unstable modes, controllable. Equivalently stabilizable if for all uncontrollable modes, naturally stable.
Popov-Belevitch-Hautus (PBH) test is an algebraic method to verify stabilizability of a system. If for all unstable eigenvalues $Re(\lambda) \geq 0$ it is off full rank n (size of state vector), it is stabilizable [3].

$$\operatorname{rank}\left(\begin{bmatrix}\lambda I - A & B\end{bmatrix}\right) = n \tag{2.13}$$

2.5 Linear Parameter-Varying Systems

Linear Parameter Varying models are generally built upon the LTI framework by expanding their region of validity. The LPV system is (instead of static matrices) built with parameter varying model with the following equations.

$$\dot{x} = A(\rho(t))x + B(\rho(t))u$$

$$y = C(\rho(t))x + D(\rho(t))u$$

where $\rho(t) \in \mathcal{P} \subset \mathbb{R}^{n_\rho}$ is a parameter vector (real-time available) belonging to a compact, convex parameter polytope $\mathcal{P}$. A note about LPV is that there is no single standard method when it comes to developing a LPV model. The theory in this section is based on [4].

2.5.1 LPV stability

Lyapunov functions for LPV models are complex, stability is usually analysed using sufficient conditions.

Parameter-dependent stability

The LPV model is asymptotically stable if there exists a smooth $P(\rho) = P^\top(\rho) \succ 0$ and $c > 0$ such that:

$$\frac{\partial P(\rho)}{\partial \rho}\dot{\rho} + P(\rho)A(\rho) + A^\top(\rho)P(\rho) + cP(\rho) \prec 0 \tag{2.14}$$

Quadratic stability

If P is parameter-independent, the condition simplifies to:

$$PA(\rho) + A^\top(\rho)P + cP \prec 0 \tag{2.15}$$

For affine or polytopic $A(\rho)$ over a convex $\mathcal{P}$, it suffices to satisfy this at the vertices of $\mathcal{P}$ only.

2.6 Robust Control

Robust control is designed to stabilize and perform in the presence of model uncertainty and disturbances. Robust control allows for explicitly defining uncertainty into the synthesis, while other controllers such as LQR have implicit robustness properties. The theory in this section is based on [4]

2.6.1 H_∞ norm

For an asymptotically stable, strictly proper transfer function G(s) / g(t), the H_∞-norm is the peak value of the transfer function magnitude plot. Also known as the largest singular value. Interpreted as a peak amplification (or peak induced 2 norm).

$$\|G(s)\|_\infty = \|G(i\omega)\|_\infty = \max_{\omega \in \mathbb{R}} \bar{\sigma}\Big(G(i\omega)\Big) \tag{2.16}$$

2.6.2 Generalized Plant

Below is the general feedback-loop block diagram for robust control.

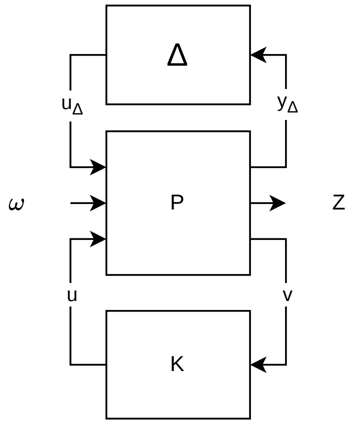


Figure 2.3: P-K-Δ Structure

Symbol	Description
P	Generalized plant
K	Controller
Δ	Uncertainty block
ω	Exogenous inputs
Z	Regulated outputs
v	Measured outputs to controller
u	Control inputs
u_Δ	Uncertainty input
y_Δ	Uncertainty output

Table 2.1: P-K-Δ signal definitions

The generalized plant P contains weights, the plant model G and the interconnected structure. Exogenous inputs ω refers to reference, noise and disturbances. Δ represents uncertainty and is unknown other than that it is of diagonal structure and is magnitude bound $\|\Delta\|_\infty \leq 1$. The control law is $u = Kv$.

2.6.3 H_∞ synthesis

H_∞ synthesis can be interpreted as a minmax optimisation problem by minimizing the effect of the worst case (max) of the H_∞-norm from ω to Z. This optimisation is done with by solving two coupled Riccati equations. This worst case optimisation is what gives the controller explicit robustness properties.

2.6.4 Weight shaping

The approach is known as mixed sensitivity, where multiple closed-loop transfer functions are implemented simultaneously through weight shaping. There are three main weights that handle the base of weight shaping in the frequency plane, the W_e, W_p and W_u weight.

$$Z_e = W_e \cdot e \tag{2.17}$$

$$Z_u = W_u \cdot u \tag{2.18}$$

$$Z_p = W_p \cdot y \tag{2.19}$$

Error tracking weight W_e, control effort weight W_u and performance weight W_p.

2.6.5 Mixed sensitivity and Gamma γ

Sensitivity S is the closed loop from disturbance to output, Complementary sensitivity $T = I - S$ is the closed loop from exogenous input to output. $S + T = I$ is their trade-off relationship. KS refers to the closed loop from output disturbance to control input. The mixed sensitivity optimisation problem is then:

$$\min_K \left\| \begin{pmatrix} W_e S \\ W_u KS \\ W_p T \end{pmatrix} \right\|_\infty < \gamma \tag{2.20}$$

Objective is to make γ as low as possible and a value of $\gamma < 1$, indicates that all weighted closed-loop transfer functions are simultaneously bounded.

2.6.6 Uncertainty

In modelling, there are dynamics that are left out or are not accurate. Such as unmodeled actuator dynamics, changes in air density, vehicle elasticity and much more. These are often defined as unstructured or structured. In the robust control toolbox, this can be explicitly included. Multiplicative uncertainty is defined as uncertainty on the output of the dynamics in a multiplicative manner. This is unstructured (not targetting a specific parameter) and signifies that the assumptions and simplifications in modelling will differ from reality [5].

$$\tilde{y} = (I + W_{oM} \cdot \Delta) y \tag{2.21}$$

2.7 LQR, feedforward and gain scheduling

Linear Quadratic Regulator, is a controller that optimizes the feedback gain

$$u = -Kx \tag{2.22}$$

for a given linear system with the quadratic cost-function of the state weighted cost Q and input weighted cost R.

$$J = \int_0^\infty \left(x^T Q x + u^T R u \right) dt \tag{2.23}$$

LQR is designed to minimize errors from their operating point, they don't include an offset for where that operating point is. This is not a limitation of LQR but a feature of it, feedforward addresses this with the updated control law.

$$u = -Kx + \bar{N}r \tag{2.24}$$

LQR are not designed to control for points that are not their operating point. This is addressed by gain scheduling where the controllers gains are interpolated [6].

$$u = -K(s)y + \bar{N}(s)r \tag{2.25}$$

3

Methods

A number of methodologies spanning from aircraft modelling, aerodynamics, CFD, system analysis, controller designs and eventual simulation test manoeuvrers have been explored, which this chapter aims at explaining. The time parameter t for functions is neglected throughout the chapter to avoid clutter.

3.1 Modelling Framework Overview

Model based control is entirely dependent on accurate modelling of the system, which is particularly time intensive and complex for aircraft due to aerodynamic uncertainties. While simplified modelling approaches suffice for standard fixed-wing flight, they fail to capture the complex dynamics of tailsitter aircraft, especially during the transition phase. Hence, an elaborate set of modelling approaches were adapted to capture the fundamental principles that tailsitters operate on. Among a few key concepts are:

- Segmentation of the airframe into freestream and propeller slipstream segments, to capture elevon control authority as airspeed is near zero in hover.
- The aircraft is symmetric around the XZ-plane which enables decoupling of slipstream and elevon deflection coefficients from sideslip. This in turn enables mirroring of obtained coefficients from the right side to the left. This is a strict requirement for combinatoric reasons of all the parameters affecting aerodynamic coefficients.

These derivations and assumptions are further detailed in Section 3.2.4.

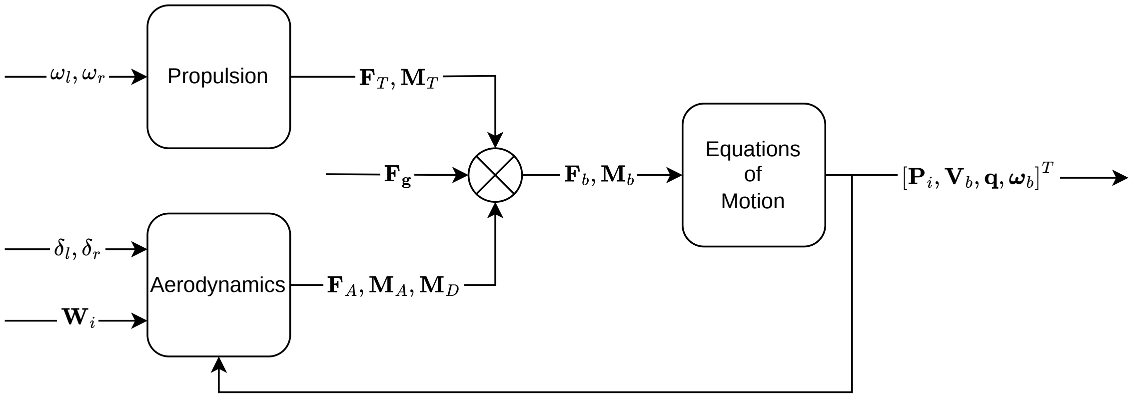


Figure 3.1: Model signal path from system inputs, including wind $\mathbf{W_i}$, to resulting states.

Fig. 3.1 details the signal paths in the overall model. The aerodynamics block consists of coefficient look-up tables, detailed in Section 3.4, derived via CFD simulation which is explained in Section 3.3.

3.2 Aircraft Modelling

This section details the model dynamics of the aircraft, including equations of motion, propulsion and aerodynamic forces and moments.

3.2.1 Platform Specification

The aircraft investigated is a 250 gram, fixed wing, dual-rotor UAV, made of primarily styrofoam. The dimensions and rotor position are shown in Fig. 3.2.

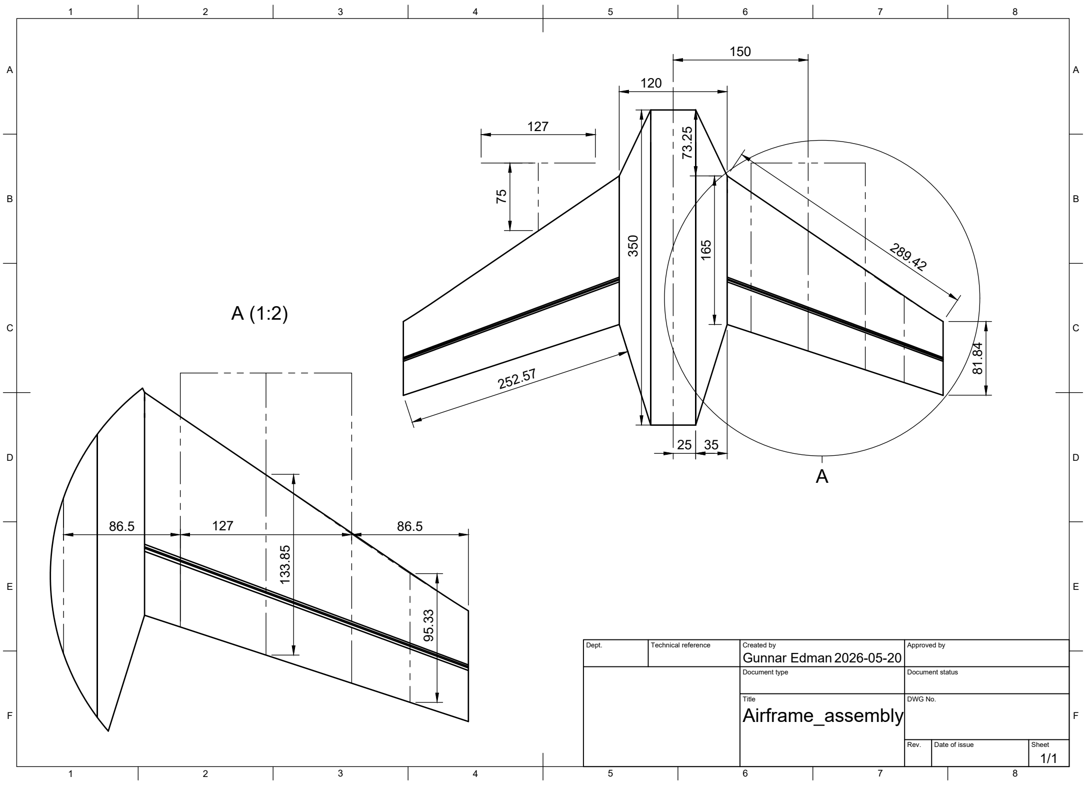


Figure 3.2: Tailsitter airframe drawing (mm)

The moment of inertia was approximated based on the longitudinal and lateral placement of the major components:

- Battery
- Avionics
- Payload (such as camera)
- Propulsion left and right (main contributors to roll and yaw inertia)
- Homogeneous styrofoam frame, inertia obtained in CAD

Utilising the XZ-plane symmetry and adding the inertia matrices of the mass points to the styrofoam CAD inertia matrix, the following total inertia matrix was obtained:

$$J_{total} = \begin{bmatrix} 0.001494 & 0.000000 & 0.000003 \\ 0.000000 & 0.000679 & 0.000000 \\ 0.000003 & 0.000000 & 0.002155 \end{bmatrix} \mathrm{kg} \cdot \mathrm{m}^2 \tag{3.1}$$

Comparing the approximation with another master thesis [7], which evaluated a similar platform of approximately the same mass, confirms the approximation is reasonable. The aircraft in his research has landing gear mounted at the wingtips, explaining the larger roll and yaw inertias:

$$J_{chap} = \begin{bmatrix} 0.002986 & -0.000022 & -0.000012 \\ -0.000022 & 0.000637 & 0.000001 \\ -0.000012 & 0.000001 & 0.003487 \end{bmatrix} \mathrm{kg} \cdot \mathrm{m}^2 \tag{3.2}$$

Parameter	**Symbol**	**Value**	**Unit**
Total Mass	m	0.25	kg
Total Wingspan	b	0.60	m
Propeller Radius	r_p	0.0635	m
Rotor Lateral Offset	l	0.150	m
Geometric Segments			
Center Span	b_{cent}	0.120	m
Slipstream Span	b_{slip}	0.127	m
Tip Span	b_{tip}	0.0856	m
Center Area	S_{cent}	44.02×10^{-3}	m^2
Slipstream Area	S_{slip}	16.93×10^{-3}	m^2
Tip Area	S_{tip}	8.27×10^{-3}	m^2

Table 3.1: Physical and geometric parameters of the tailsitter platform

The total reference area S_{ref} and the area-weighted Mean Aerodynamic Chord (MAC) $\bar{c}$ are derived from the discretized geometric segments:

$$S_{ref} = \sum S_i, \quad \bar{c} = \frac{\sum(S_i c_i)}{S_{ref}} \tag{3.3}$$

where $c_i = S_i/b_i$ is the local mean chord for each segment.
In order to stay consistent, standard aerospace reference frames were used in this thesis, with the body frame being located in the CG of the aircraft:

- Inertial frame (I): North East Down (NED)
- Body frame (b): Forward Right Down (FRD)

As shown in Fig. 3.3, the body frame is preserved regardless of flight mode. Rotations about the respective body axes are defined as $(x_b, y_b, z_b) \mapsto (\text{roll}, \text{pitch}, \text{yaw})$.

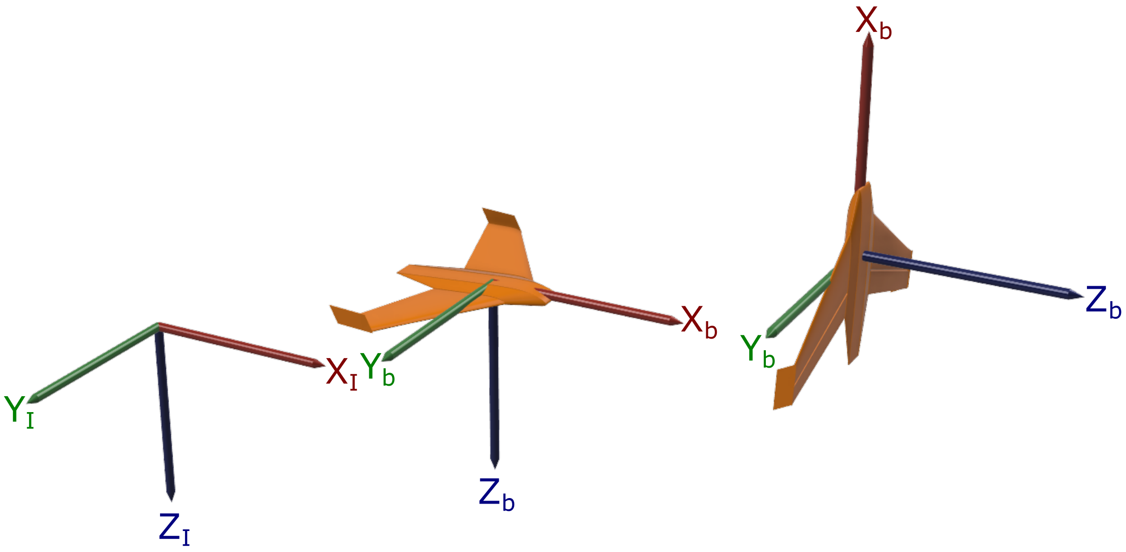


Figure 3.3: Reference frames

3.2.2 Equations of Motion

This section provides the overall equations of motion of the aircraft and individual forces and moments are derived in subsequent section. The total forces $\mathbf{F}_b$ and moments $\mathbf{M}_b$ acting on the aircraft in the body frame are expressed as:

$$\mathbf{F}_b = \mathbf{F}_A + \mathbf{F}_T + \mathbf{F}_g \tag{3.4}$$

$$\mathbf{M}_b = \mathbf{M}_A + \mathbf{M}_T + \mathbf{M}_D \tag{3.5}$$

where:

- $\mathbf{F}_A, \mathbf{M}_A$ are the steady-state aerodynamic forces and moments.
- $\mathbf{F}_T, \mathbf{M}_T$ are the forces and moments generated by the thrusters.
- $\mathbf{M}_D$ is the aerodynamic damping moments.
- $\mathbf{F}_g = \mathbf{R}_{bi}^T \begin{bmatrix} 0 & 0 & mg \end{bmatrix}^T$

A unit quaternion is used to represent the aircraft's attitude, yielding the vector:

$$\mathbf{q} = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix}^T \tag{3.6}$$

The transformation from body frame to inertial frame is given by the rotation matrix $\mathbf{R}_{bi}$:

$$\mathbf{R}_{bi} = \begin{bmatrix} 1 - 2(q_y^2 + q_z^2) & 2(q_x q_y - q_w q_z) & 2(q_x q_z + q_w q_y) \\ 2(q_x q_y + q_w q_z) & 1 - 2(q_x^2 + q_z^2) & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z - q_w q_y) & 2(q_y q_z + q_w q_x) & 1 - 2(q_x^2 + q_y^2) \end{bmatrix} \tag{3.7}$$

The relationship between the body-frame angular rates $\boldsymbol{\omega}_b$ and the quaternion derivative $\dot{\mathbf{q}}$ is defined by the skew-symmetric kinematic rate matrix $\boldsymbol{\Omega}(\boldsymbol{\omega}_b)$, as defined in equation 3.8.

$$\boldsymbol{\Omega}(\boldsymbol{\omega}_b) = \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \tag{3.8}$$

The complete equations of motion are finally formulated as:

$$\begin{cases} \dot{\mathbf{P}}_i = \mathbf{R}_{bi}\mathbf{V}_b \\ \dot{\mathbf{V}}_b = \dfrac{1}{m}\mathbf{F}_b - \boldsymbol{\omega}_b \times \mathbf{V}_b \\ \dot{\mathbf{q}} = \dfrac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_b)\mathbf{q} \\ \dot{\boldsymbol{\omega}}_b = \mathbf{J}^{-1}\left(\mathbf{M}_b - \boldsymbol{\omega}_b \times (\mathbf{J}\boldsymbol{\omega}_b)\right) \end{cases} \tag{3.9}$$

3.2.3 Propulsion

Since exact thruster specifications were unavailable, the rotor dynamics are approximated using generic rotor coefficients from [7]. As previously established, this provides a reasonable baseline due to the similarities between the evaluated aircraft. For real-world implementation, physical tests measuring thruster force should be performed to establish exact thruster coefficients. Additionally, modelling propeller advance ratio was omitted due to the absence of specific wind tunnel data. This remains an acceptable simplification for this scope, with the resulting aerodynamic mismatches treated as unmodelled plant dynamics. The resulting force T_i and torque Q_i are obtained, using momentum theory, as:

$$T_i = \frac{4}{\pi^2}\rho\omega_i^2 r_p^4 C_t \tag{3.10}$$

$$Q_i = \frac{4}{\pi^3}\rho\omega_i^2 r_p^5 C_p \tag{3.11}$$

where ρ is the air density. The thrust force is aligned with the chord line of the aircraft and hence, all the force is directed along the x-axis pushing the aircraft in the positive x-direction:

$$\mathbf{F}_T = \begin{bmatrix} T_l + T_r \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{M}_T = \begin{bmatrix} Q_l - Q_r \\ 0 \\ l(T_l - T_r) \end{bmatrix} \tag{3.12}$$

The thrusters induce moments around two main axis. Firstly from the aerodynamic torque differential between the left and right rotors, which induces a roll moment about the x-axis. Secondly the thrust differential acting over the lateral lever arm l, which induces a yaw moment about the z-axis. The left and right contribution signs are determined from rotor directions, where left rotor is counter clockwise and the right rotor is clockwise, seen from behind.

3.2.4 Aerodynamic Forces and Moments

The most extensive forces and moments to model are the contributions from aerodynamic forces and moments caused by airflow over the airframe. The flight envelope of the aircraft extends way beyond stall of a conventional fixed wing aircraft, all the way into stationary hover at approximately 90 degrees pitch angle. In a stationary hover, the freestream airspeed velocity over the airframe approaches zero. Subsequently, if the aerodynamic forces and moments were to be derived purely from free-stream airspeed, all forces and moments would approach zero in hover meaning the elevons would have no control authority and the aircraft would be uncontrollable. To maintain controllability in hover, the tailsitter relies entirely on prop-wash which causes an increased airspeed in the slipstream of the rotors. The rotor slipstreams are assumed roughly equivalent to the propeller radius. This means the airframe experiences localized airspeeds over different segments of the airframe, which are visualized in Fig. 3.4. This segmentation method is utilized in similar research such as [7].

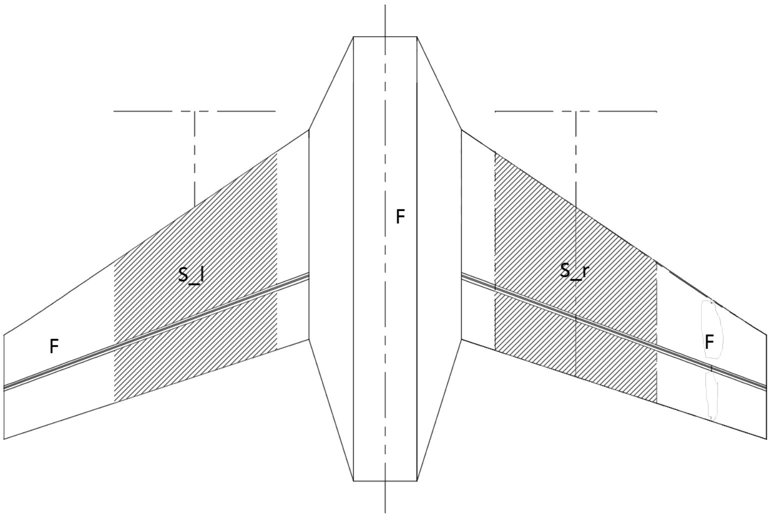


Figure 3.4: Airframe segments

To obtain accurate aerodynamic forces and moments, apparent airspeed accounting for translational velocity $\mathbf{V_b}$ and velocity of wind $\mathbf{W}_i$ is computed as:

$$\mathbf{V}_{a,b} = \mathbf{V}_b - \mathbf{R}_{ib}\mathbf{W}_i = \begin{bmatrix} u_a \\ v_a \\ w_a \end{bmatrix} \tag{3.13}$$

The lateral v_a and vertical w_a components are assumed uniform across the entire airframe. The longitudinal component however must account for prop-wash in the slipstream segments by augmenting the speed by the propeller thrust T_i. This yields the local longitudinal airspeed $u_{a,i}$:

$$u_{a,i} = \begin{cases} u_a & \text{for } i = F \\ \sqrt{u_a^2 + \frac{2T_i}{\rho \pi r_p^2}} & \text{for } i \in \{S_r, S_l\} \end{cases} \tag{3.14}$$

Using this localized longitudinal component, the magnitude of the local airspeed $\|\mathbf{V}_{a,i}\|$, local angle of attack α_i and sideslip angle β_i are evaluated for each segment i:

$$\|\mathbf{V}_{a,i}\| = \sqrt{u_{a,i}^2 + v_a^2 + w_a^2} \tag{3.15}$$

$$\alpha_i = \operatorname{atan2}(w_a, u_{a,i}), \quad \beta_i = \arcsin\left(\frac{v_a}{\|\mathbf{V}_{a,i}\|}\right) \tag{3.16}$$

Subsequently, the local dynamic pressure $\bar{q}_i$ acting on each segment is computed as:

$$\bar{q}_i = \frac{1}{2}\rho\|\mathbf{V}_{a,i}\|^2 \tag{3.17}$$

To model forces and moments, a number of aerodynamic coefficients are utilized. C_k, $k \in \{D, Y, L, l, m, n\}$, represents the standard 6-DOF force and moment coefficients (see section 2.1.4) which are unique for the freestream and slipstream segments respectively. The primary set of coefficients belonging to segment F is a function of α and β:

$$C_{k,F}(\alpha, \beta) \tag{3.18}$$

Segments $\{S_r, S_l\}$ share another set of coefficients to capture local aerodynamic effects in the model. Since the rotors always provide thrust in the flight envelope of interest, prop-wash T_i increases the local longitudinal velocity $u_{a,i}$ in the slipstream segments. Hence, the denominator for β in Eq. 3.16 becomes significantly larger than the contribution from realistic crosswinds in v_a, driving the fraction close to zero. This motivates a small angle assumption in the slipstream segments where $\arcsin(\approx 0) \approx 0°$. This combined with the fact that the airframe is symmetric around the XZ-plane (see Fig. 3.3) enables a single set of coefficients to be derived from S_r and to be mirrored to S_l,

$$C_{k,S}(\alpha) \tag{3.19}$$

The primary control authority of the tailsitter, aside from differential thrust, is provided by the elevons. The deflection δ of such a control surfaces not only induce moments, it also fundamentally changes the shape of the air-foil, thereby affecting all aerodynamic forces and moments. To model this, elevon deflection effects are evaluated as isolated increments ΔC that are added to the base coefficients. Decoupling ΔC from β in the slipstream segments aligns with the $\beta \approx 0°$ assumption established previously. For the freestream, this decoupling is a strict computational necessity to prevent an unmanageable number of parameter combinations with α, β, δ_r, and δ_l (further detailed in Section 3.4). By evaluating ΔC exclusively at $\beta = 0°$, the aerodynamic increments derived for the right elevon deflections δ_r can be mirrored to the left δ_l via the XZ-plane symmetry, yielding the final set of deflection coefficients:

$$\Delta C_{k,F}(\alpha, \delta), \quad \Delta C_{k,S}(\alpha, \delta) \tag{3.20}$$

The outlined aerodynamic model is numerically stable as long as $V_{a,i}$ does not approach zero. This condition breaks when the aircraft approaches stationary hover and the freestream $V_{a,F}$ approaches zero which gives rise to kinematic singularities in eq. 3.16. When stochastic wind gust disturbances are added to the simulation (see section 3.13.2), α and β changes rapidly with the noisy wind directions. This transfers directly into instantly changing $C_{k,F}$. This introduces numerical stiffness in the simulation. In particular, the resulting torque spikes combined with the exceptionally small inertia of the evaluated aircraft generate unrealistically fast angular accelerations. A solution to resolve the hover dynamics, inspired by [8], is to blend these coefficients toward zero with a cubic Hermite blending function, shown in Fig. 3.5:

$$\begin{aligned}
\mu &= \min\left(\max\left(\frac{V_{a,F} - V_{lower}}{V_{upper} - V_{lower}}, 0\right), 1\right) \\
\eta &= 3\mu^2 - 2\mu^3
\end{aligned} \tag{3.21}$$

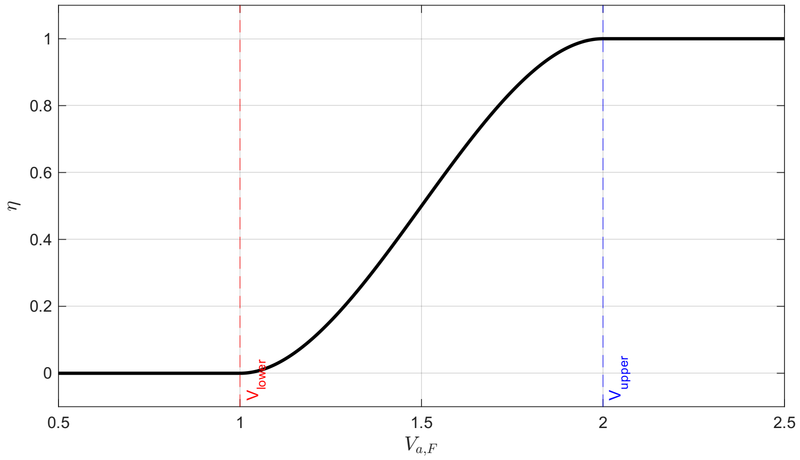


Figure 3.5: Cubic Hermite blending function

To construct the full set of coefficients, the geometric symmetry of the airframe across the XZ-plane (see Fig. 3.4) is utilized to mirror the aerodynamic data. This maps the previously defined sets into the final equations for the freestream F, right slipstream S_R, and left slipstream S_L according to two physical principles:

- **Sideslip symmetry (β direction):** The longitudinal forces and pitch moment $C_{\{D,L,m\}}$ are independent of the crosswind direction, relying only on the magnitude $|\beta|$. Conversely, the lateral-directional terms $C_{\{Y,l,n\}}$ are anti-symmetric and flip direction based on the crosswind, requiring the signum function $\text{sgn}(\beta)$.
- **Left-Side Mirroring:** Deflecting the left elevon δ_l yields the exact same longitudinal effects $C_{\{D,L,m\}}$ as the right elevon δ_r. However, the lateral-directional effects $C_{\{Y,l,n\}}$ generated by the left elevon and the left slipstream are inverted relative to the right side. Consequently, these terms are negated when applied to the left side of the aircraft.

Applying these symmetry rules and blending function η yields the final coefficient equations for the three segments:

$$\begin{cases} C_{D,F} &= \quad C_{D,F}(\alpha,|\beta|) & + \Delta C_{D,F}(\alpha,\delta_r) & + \Delta C_{D,F}(\alpha,\delta_l) \\ C_{Y,F} &= \quad C_{Y,F}(\alpha,|\beta|)\mathrm{sgn}(\beta) & + \Delta C_{Y,F}(\alpha,\delta_r) & - \Delta C_{Y,F}(\alpha,\delta_l) \\ C_{L,F} &= \quad C_{L,F}(\alpha,|\beta|) & + \Delta C_{L,F}(\alpha,\delta_r) & + \Delta C_{L,F}(\alpha,\delta_l) \\ C_{l,F} &= \eta\Big(C_{l,F}(\alpha,|\beta|)\mathrm{sgn}(\beta) & + \Delta C_{l,F}(\alpha,\delta_r) & - \Delta C_{l,F}(\alpha,\delta_l)\Big) \\ C_{m,F} &= \eta\Big(C_{m,F}(\alpha,|\beta|) & + \Delta C_{m,F}(\alpha,\delta_r) & + \Delta C_{m,F}(\alpha,\delta_l)\Big) \\ C_{n,F} &= \eta\Big(C_{n,F}(\alpha,|\beta|)\mathrm{sgn}(\beta) & + \Delta C_{n,F}(\alpha,\delta_r) & - \Delta C_{n,F}(\alpha,\delta_l)\Big) \end{cases} \tag{3.22}$$

$$\begin{cases} C_{D,S_R} &= C_{D,S}(\alpha) & + \Delta C_{D,S}(\alpha,\delta_r) \\ C_{Y,S_R} &= C_{Y,S}(\alpha) & + \Delta C_{Y,S}(\alpha,\delta_r) \\ C_{L,S_R} &= C_{L,S}(\alpha) & + \Delta C_{L,S}(\alpha,\delta_r) \\ C_{l,S_R} &= C_{l,S}(\alpha) & + \Delta C_{l,S}(\alpha,\delta_r) \\ C_{m,S_R} &= C_{m,S}(\alpha) & + \Delta C_{m,S}(\alpha,\delta_r) \\ C_{n,S_R} &= C_{n,S}(\alpha) & + \Delta C_{n,S}(\alpha,\delta_r) \end{cases} \tag{3.23}$$

$$\begin{cases} C_{D,S_L} &= \quad C_{D,S}(\alpha) & + \Delta C_{D,S}(\alpha,\delta_l) \\ C_{Y,S_L} &= -C_{Y,S}(\alpha) & - \Delta C_{Y,S}(\alpha,\delta_l) \\ C_{L,S_L} &= \quad C_{L,S}(\alpha) & + \Delta C_{L,S}(\alpha,\delta_l) \\ C_{l,S_L} &= -C_{l,S}(\alpha) & - \Delta C_{l,S}(\alpha,\delta_l) \\ C_{m,S_L} &= \quad C_{m,S}(\alpha) & + \Delta C_{m,S}(\alpha,\delta_l) \\ C_{n,S_L} &= -C_{n,S}(\alpha) & - \Delta C_{n,S}(\alpha,\delta_l) \end{cases} \tag{3.24}$$

Finally, the aerodynamic forces and moments are obtained as a sum of the localized contributions in segments $i \in \{F, S_r, S_l\}$, utilizing the reference area S_{ref} defined in Section 3.2.1:

$$\mathbf{F}_A = \sum_i \mathbf{R}_{wb,i} \begin{bmatrix} -\bar{q}_i S_{\mathrm{ref}} C_{D,i} \\ \bar{q}_i S_{\mathrm{ref}} C_{Y,i} \\ -\bar{q}_i S_{\mathrm{ref}} C_{L,i} \end{bmatrix} \tag{3.25}$$

$$\mathbf{M}_A = \sum_i \begin{bmatrix} \bar{q}_i S_{\mathrm{ref}} b C_{l,i} \\ \bar{q}_i S_{\mathrm{ref}} \bar{c} C_{m,i} \\ \bar{q}_i S_{\mathrm{ref}} b C_{n,i} \end{bmatrix} \tag{3.26}$$

R_{ab} defines the rotation matrix from wind frame to body frame:

$$\mathbf{R}_{ab} = \begin{bmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{bmatrix} \tag{3.27}$$

3.2.5 Aerodynamic Damping Coefficients

Initial attempts to analytically derive localized damping coefficients resulted in numerical singularities as the aircraft transitioned into hover (detailed in Appendix 1). Without access to wind tunnel testing for dynamic oscillation data, constant aerodynamic damping derivatives were manually tuned to provide baseline attitude stability in the simulation. The empirically selected values were $C_{l_p} = -0.4$ for roll, $C_{m_q} = -1.0$ for pitch, and $C_{n_r} = -0.15$ for yaw, yielding the moment damping vector:

$$\mathbf{M}_D = \bar{q} S_{ref} \begin{bmatrix} b C_{l_p} \frac{pb}{2||\mathbf{V}_b||} \\ \bar{c} C_{m_q} \frac{q\bar{c}}{2||\mathbf{V}_b||} \\ b C_{n_r} \frac{rb}{2||\mathbf{V}_b||} \end{bmatrix}, \qquad \bar{q} = \frac{1}{2}\rho ||\mathbf{V}_b||^2 \tag{3.28}$$

where $\bar{q}$ is computed purely from ground-speed to prevent excessive oscillations.

3.3 Aerodynamic Coefficients from CFD

Aerodynamic coefficients in 6-DOF can be obtained through several means, some of which are wind-tunnel tests, CFD simulations and utilizing airfoil databases in combination with analytical approaches. The aircraft investigated has low mass in relation to surface area and hence, its characteristics are highly influenced by these coefficients. Since a wind-tunnel were not easily accessible, CFD simulation was selected, providing higher fidelity then purely analytical approaches such as flat-plate theory used in [7]. Specifically, 3D incompressible CFD using the cloud-based program SimScale [1] were utilized. Fig. 3.6 outlines the full chain from CAD to coefficients being used in the dynamical model.

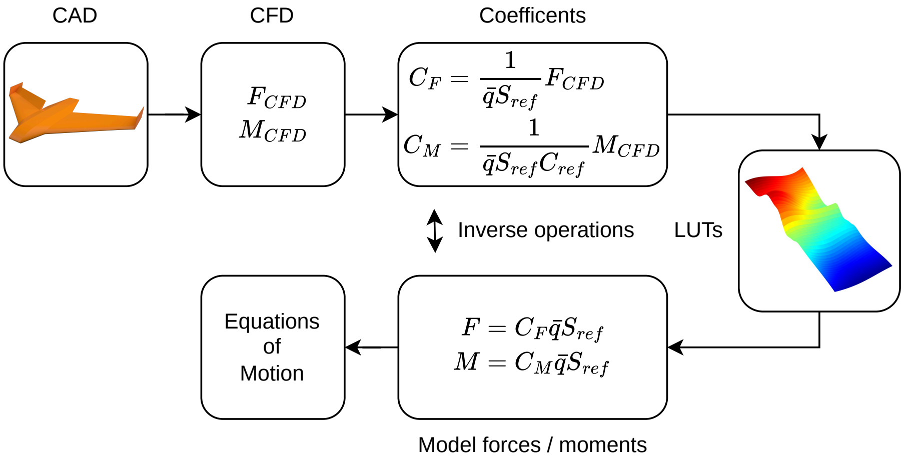


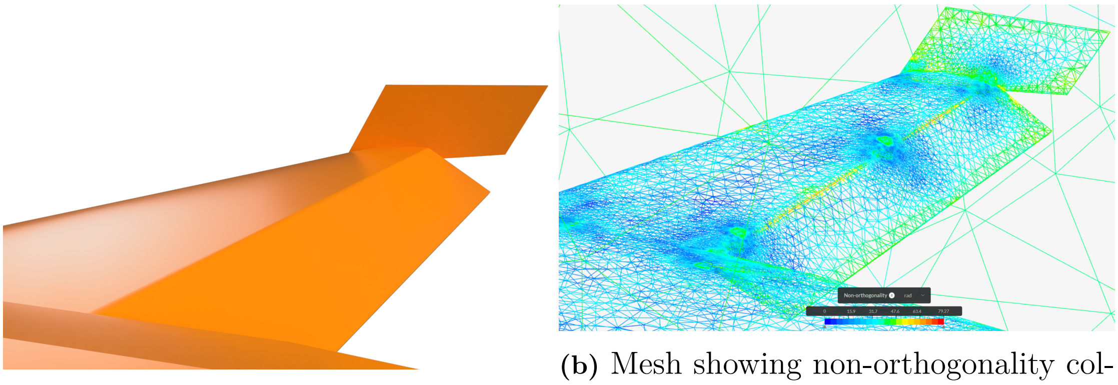
Figure 3.6: CFD coefficients pipeline

---

[1] SimScale GmbH, Munich, Germany. Available at: `https://simscale.com`

3.3.1 CAD and Meshing

Only the clean airframe was analysed when obtaining aerodynamic coefficients. Adding complex shapes of motors, rotors and other components has diminishing returns since the number of cells required to obtain a valid mesh increases significantly, while resulting coefficients are not enhanced to the same degree. The CAD model provided by Remote Aero also had to be slightly modified and cleaned up in numerous iterations before the non-orthogonality of the mesh was within acceptable ranges. This does not refer to modifying the airframe but rather ensuring the surfaces are perfectly closed and no cavities are present. In particular, elevon hinges that remain closed, smooth surfaces for all deflection angles had to be modelled in CAD, using sweep surfaces parametrized on deflection angle δ_r. Fig. 3.7 shows the elevon hinge in CAD and the resulting successful mesh.



(a) CAD render

(b) Mesh showing non-orthogonality colormap

Figure 3.7: Comparison between CAD and resulting mesh, showcasing parametric elevon hinge surface being meshed without excessive non orthogonality (shown in red).

The meshes were generated using the SimScale standard mesh algorithm, which provides high-level adjustments shown in Table 3.2. The global fineness level dictates the cell size across the entire volume, including all ambient air. It was therefore kept at default value, while surface refinement was applied to the entire airframe. Extra high refinment were required at the leading edge, highlighted in Fig. 3.8, to capture its tight radius of curvature and since the leading edge is highly influential on the simulation results.

Area	Setting	Value
Global	Fineness	Level 5
Airframe	Surface refinement	Level 7
Leading edge	Surface refinement	Level 9
Airframe	Inflate boundary layer	– 3 layers – First-layer size: 1.425×10^{-3} m – Expansion ratio: 1.3
Leading edge	Inflate boundary layer	– 3 layers – First-layer size: 0.57×10^{-3} m – Expansion ratio: 1.1

Table 3.2: SimScale mesh generation parameters

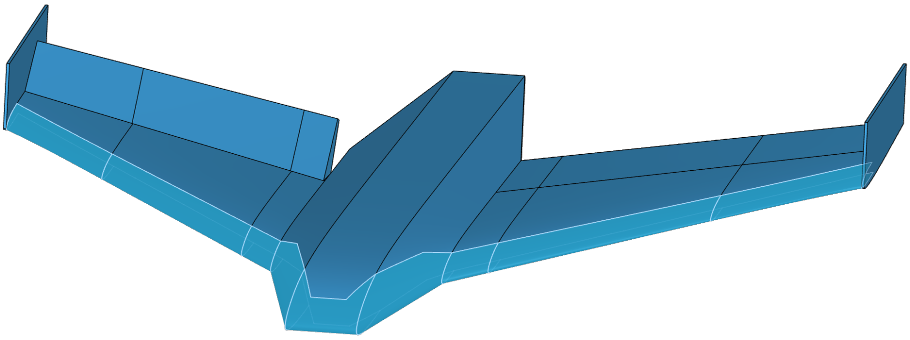


Figure 3.8: Highlighted leading edge surfaces, subject to increased mesh resolution

In addition to general cell size, a mesh feature of great importance is the boundary layers which model the attached flow over the airframe surfaces. For the purposes of controller design and simulation in this thesis, only approximate aerodynamic coefficients over a wide range of α are feasible given the computation and time budget available. Hence, targeting $y^+ > 30$ under nominal conditions at α around zero is the obvious choice since the alternative of $y^+ < 5$ would be extremely computationally expensive. Approximately 15 iterations of empirical tuning and test simulations resulted in the parameters outlined in Table 3.2. Every discrete elevon deflection δ_r evaluated, requires a new mesh. The final mesh obtained for $\delta_r = 0$, along with resulting y^+ from test simulation, is shown in Fig. 3.9, having 1.4 million cells across the entire domain of air. A major trade-off between mesh resolution and target $y^+ > 30$ had to be made, explaining the sub-optimal y^+ along the leading edge and other areas.

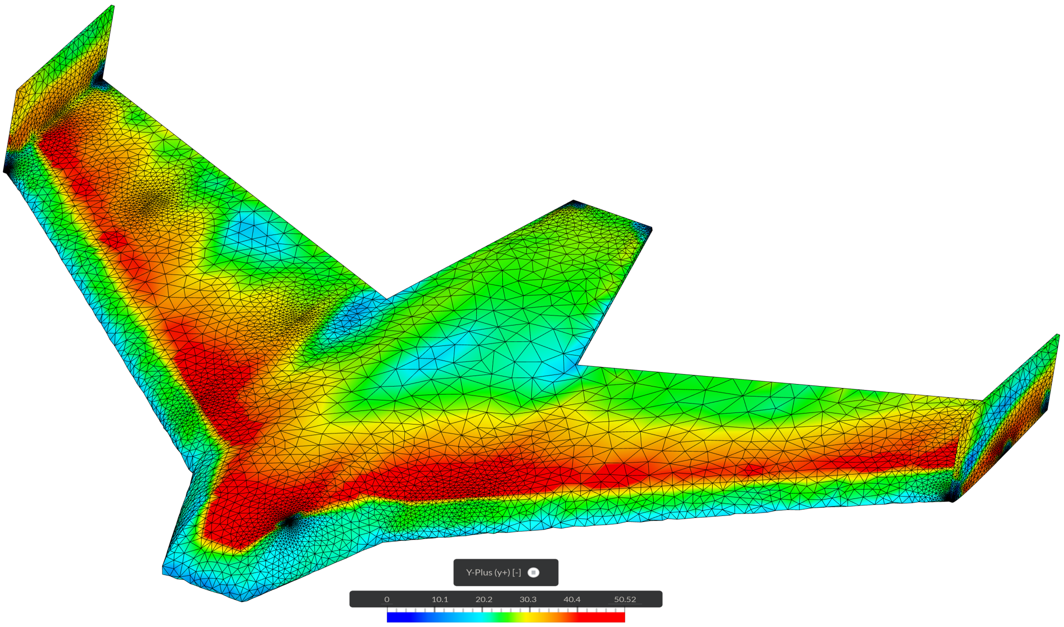


Figure 3.9: Airframe surface mesh contours and colormap highlighting y^+ from simulation with $(\alpha, \beta, \delta_r) = (0°, 0°, 0°)$.

The aircraft operates in the low Reynolds number regime, calculated for the nominal cruise speed ($V \approx 15$ m/s) and mean aerodynamic chord ($c \approx 0.127$ m) as:

$$Re = \frac{\rho V c}{\mu} \approx 10^5 \tag{3.29}$$

At this scale, the physical boundary layer is thick relative to the airframe [9, Ch. 2, Sec. 2.2]. Increasing the boundary layers to satisfy y^+ however, drastically degrades the mesh quality, not properly capturing the geometries and most importantly the leading edge. This in turn breaks the non-orthogonality requirements suggested by SimScale 2.2.1. Consequently, thinner layers were utilized, pushing y^+ within the buffer zone in some areas of the airframe. However, the $k - \omega$ SST turbulence model used in SimScale natively accounts for this. It utilizes blending functions that interpolate across the buffer zone, preserving the validity of the simulation results [10]. Furthermore, as increasing angle of attack induce flow separation [11, Ch. 4, Sec. 4.12.4], pressure drag heavily dominates over friction drag, diminishing the overall impact of strict y^+ requirements.

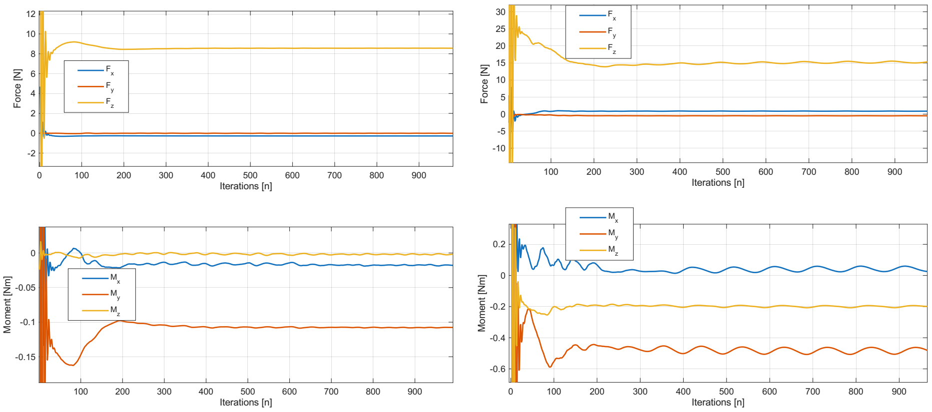
3.3.2 CFD Simulation

CFD simulation was performed using SimScale standard incompressible, steady-state flow using the SIMPLE algorithm and the $k - \omega$ SST turbulence model. A nominal cruise velocity magnitude of 15 m/s was utilized across all simulations. To enable simulation across the full ranges of α and β, a custom freestream boundary condition, shown in Table 3.3, was assigned to the domain box faces. This setup functions as a bidirectional inlet-outlet, allowing flow to enter or exit any face depending on the specified wind direction.

Boundary Parameter	**Value**
Gauge Pressure	0 Pa
Turbulent kinetic energy (k)	8.5×10^{-3} m^2/s^2
Specific dissipation rate (ω)	57 s^{-1}

Table 3.3: Inlet-outlet boundary condition parameters

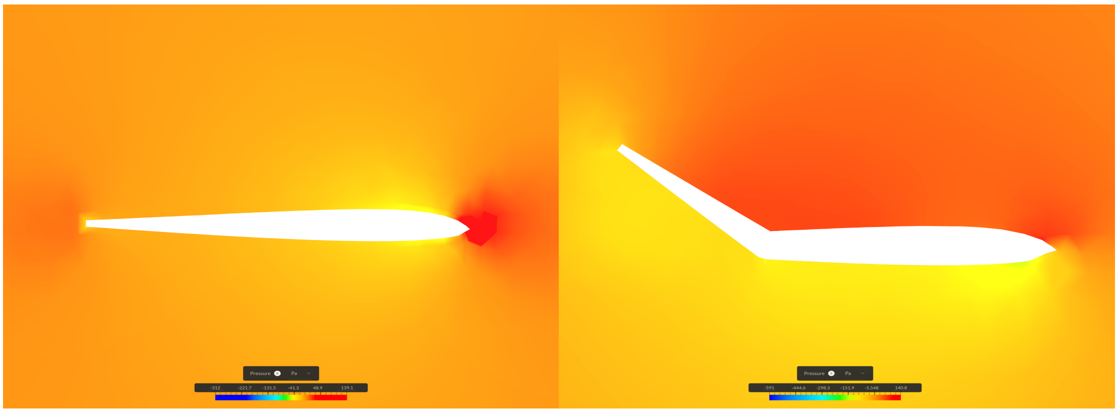
Convergence was ensured by running all simulations for 1000 iterations. Results from one conservative case, and one extreme case are shown in Fig. 3.10. In the conservative case, clear steady-state convergence occurs. For the extreme case, oscillations occur, attributed to the significant flow separation and resulting turbulence in in the wake shown in Fig. 3.12 and 3.13. Moments across all cases are more prone to oscillations around convergence-points due to the simulation-noise being amplified by the lever arms to CG across the airframe.



(a) $(\alpha, \beta, \delta_r) = (10°, 0°, 0°)$ **(b)** $(\alpha, \beta, \delta_r) = (45°, 0°, 35°)$

Figure 3.10: CFD simulation comparison comparing convergence for a moderate attack angle on the left and extreme on the right, highlighting increased oscillations.

The simulation results constitutes the steady state simulation values at the end of simulation. One result of interest is Fig. 3.11 which clearly highlights the massive pressure difference changes induced by elevon deflections, highlighting the need for Δ coefficients outlined in section 3.2.4.



(a) $\delta_r = 0°$ **(b)** $\delta_r = -35°$

Figure 3.11: 2D cross-section view of wing showing pressure gradient with $(\alpha, \beta) = (0°, 0°)$, showcasing significant increase in pressure difference with elevon deflection.

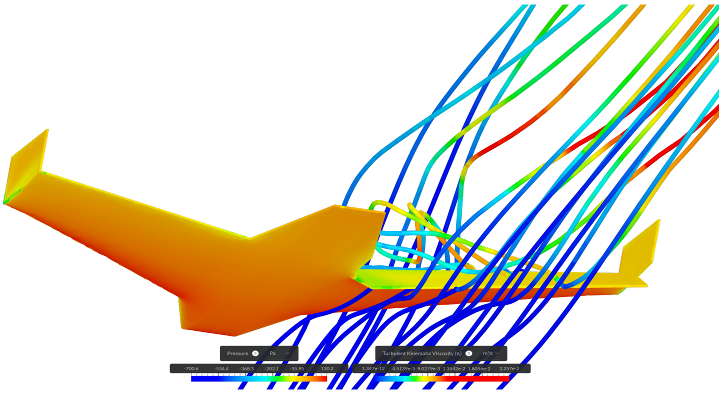


Figure 3.12: Particle trace showcasing flow separation and resulting turbulence above elevon with $\delta_r = 35°$

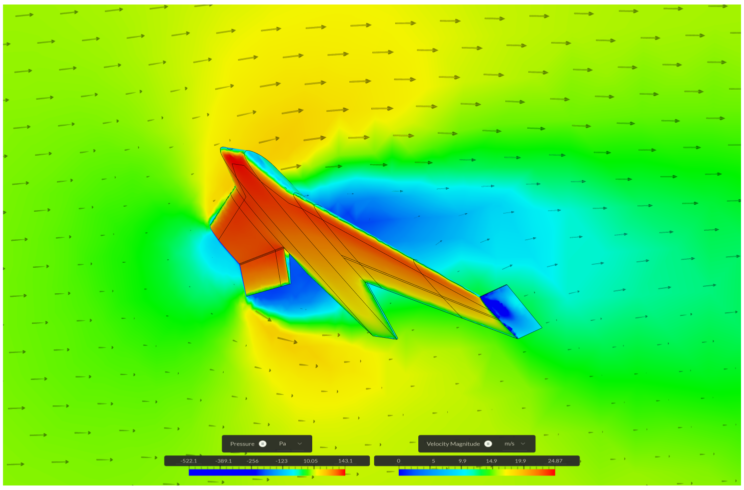


Figure 3.13: Cutting plane showing effects around deflected elevon with $(\alpha, \beta, \delta_r) = (0°, 0°, 0°)$.

3.3.3 Coefficient extraction

Coefficients are computed by averaging the last $N = 50$ iterations of the simulation, handling the oscillations shown in Fig. 3.10. As explained in Section 2.1.4 and shown in Fig. 3.6, dimensionless force C_F and moments C_M coefficients are obtained as:

$$C_F = \frac{1}{\bar{q} S_{ref}} F_{CFD}, \qquad C_M = \frac{1}{\bar{q} S_{ref} C_{ref}} M_{CFD} \tag{3.30}$$

To obtain the aerodynamic coefficients across the relevant flight envelope, sweeps of the parameters $(\alpha, \beta, \delta_r)$ were performed utilizing the SimScale Python API to automate the execution of simulations and extraction of data. As explained in Section 3.2.4, the ΔC coefficients are decoupled from β. This is a strict requirement since not doing so would far exceed the computational budget provided by SimScale.

- α is sampled non uniformly with enhanced resolution in critical regions around 0° and stall region around 38° where C_L drops.
- δ_r is sampled uniformly, more sparsely then α to respect budget constraints.
- β is only sampled at 0° and sparsely at 20° due to the computational constraints.

The distribution across the parameters are shown as discrete points in the figures following, for instance Fig. 3.14.

3.4 Coefficient Lookup Tables

To bridge the gap between the obtained, discrete sparse coefficient samples and the dynamics model, lookup tables (LUTs) are used. They return coefficients based on input parameters α, β, δ and also handle interpolation between the samples. To simplify the trim search process in Section 3.6.2 and ensure smooth derivatives in the linearization process, spline interpolation is utilized. Assuming consistent CFD coverage is provided in the regions of interest, it successfully approximates the real world aerodynamics sufficiently for the purposes of this thesis. Linear interpolation provides a safer interpolation method without the risk of overfitting on outlier samples, but ultimately fails in the trim process making spline the method of choice. The resulting

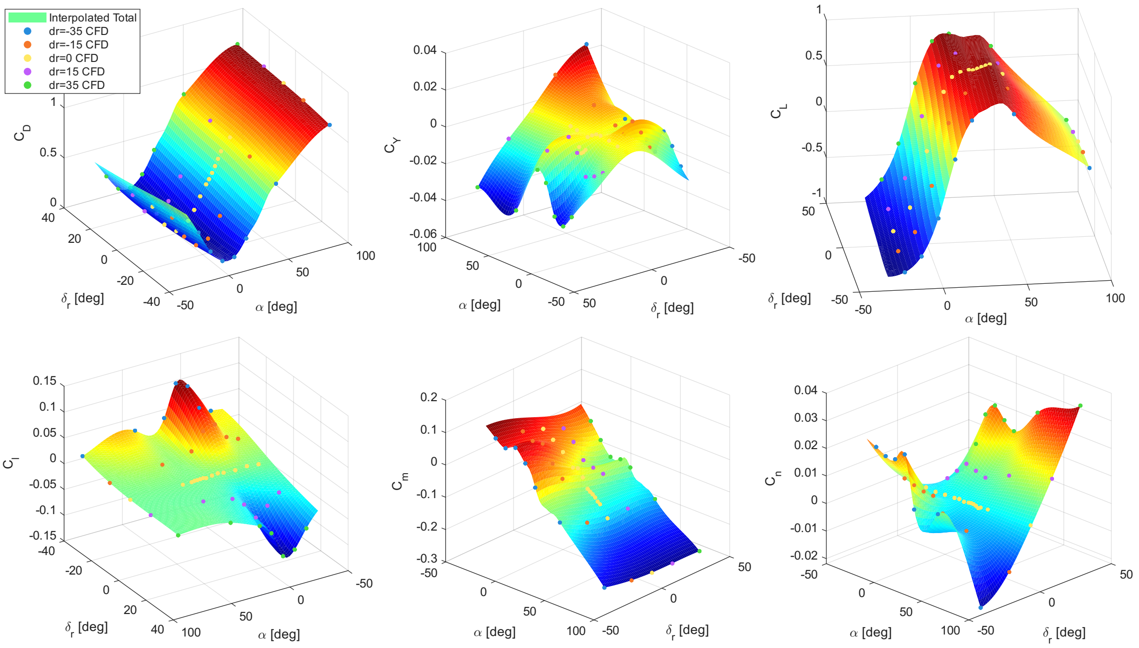


Figure 3.14: 3D LUTs plot showing total aircraft coefficients $\sum_i C_{k,i}$ with $\beta = 0$ and right elevon actuated over zoomed in domain of interest.

3.5 Model limitations

In some areas of the modelling, approximations had to be made in lack of data and experimental tests. The limitations of the model are:

- Exact thruster specification unknown. Bench tests using a digital scale to obtain a thrust curve would have enhanced the model further. This however only scales the required control signals across operating points slightly and does not fundamentally change all of the dynamics.
- Thruster advance ratio was ignored in the absence of propeller specifications or wind tunnel tests.
- Decoupling elevon deflection Δ coefficients and slipstream segment coefficients from sideslip angle β.
- Lack of sideslip samples from CFD, which limits the ability to test severe crosswind disturbances
- Limited scope of hover model. Either more CFD-runs in β or a different flat-plate theory model could have been utilized to enhance accuracy of freestream segments model in hover.
- Gyroscopic precession from rotor inertias not modelled. Since the rotors are extremely small and no exact inertias were available, this was a reasonable simplification.

3.6 Flight Envelope and Scheduling Parameter Selection

The aircraft's dynamics operate across four distinct flight regimes, namely cruise, transition, hover and hover-descent. The system dynamics vary significantly across these flight conditions. In order to capture these different dynamics, a parametrisation grid is established using appropriate scheduling parameters.

3.6.1 Scheduling parameters

To capture the nonlinearities, the scheduling parameters needs to be chosen with care. The main parameter exhibiting nonlinearities is α, as can be seen in Fig. 3.14, spanning all of the aerodynamic force and moment effects within the flight envelope. Additionally, since the aerodynamic forces and moments are a function of dynamic pressure $\bar{q}$, in turn being a function of apparent airspeed $|V_a(t)|$, it also represents a strong candidate for scheduling. Using inertial speed can potentially provide a more robust observable measurement in real the real world, but it fails to capture the effects of for instance a non-zero mean headwind. As such, airspeed was prioritized over inertial speed yielding the final scheduling vector:

$$\boldsymbol{\rho}(t) = \begin{bmatrix} |\mathbf{V}_a| & \alpha \end{bmatrix}^T \tag{3.31}$$

3.6.2 Grid generation

A trim-point is a point in where the chosen states reach a steady state by locking some states and letting the solver search for if any combination of control inputs solves it. The conditions such that there are no accelerations and no angular rates. The two crucial trimpoint are cruise and hover, they are solved independently to ensure reliable building blocks to build the grid from.

Operating Mode	v_{bx} [m/s]	v_{bz} [m/s]	Pitch θ [°]	$\omega_{l,r}$ [rad/s]	$\delta_{l,r}$ [°]
Cruise	15	-0.2	≈ 0	700	-5
Hover	0	0	90	1000	0

Table 3.4: Initial guesses supplied to the trim solver.

Aerodynamics implausible regions are excluded for the search as no physical trim should exist there. The search algorithm is optimised by utilises the previous found trimpoint as the initial guess for the next one. If the solver fails to find a valid trimpoint at a given α, it retries with small perturbations ($\pm 2°$, $\pm 5°$). Failed trimpoints are excluded from the valid grid.

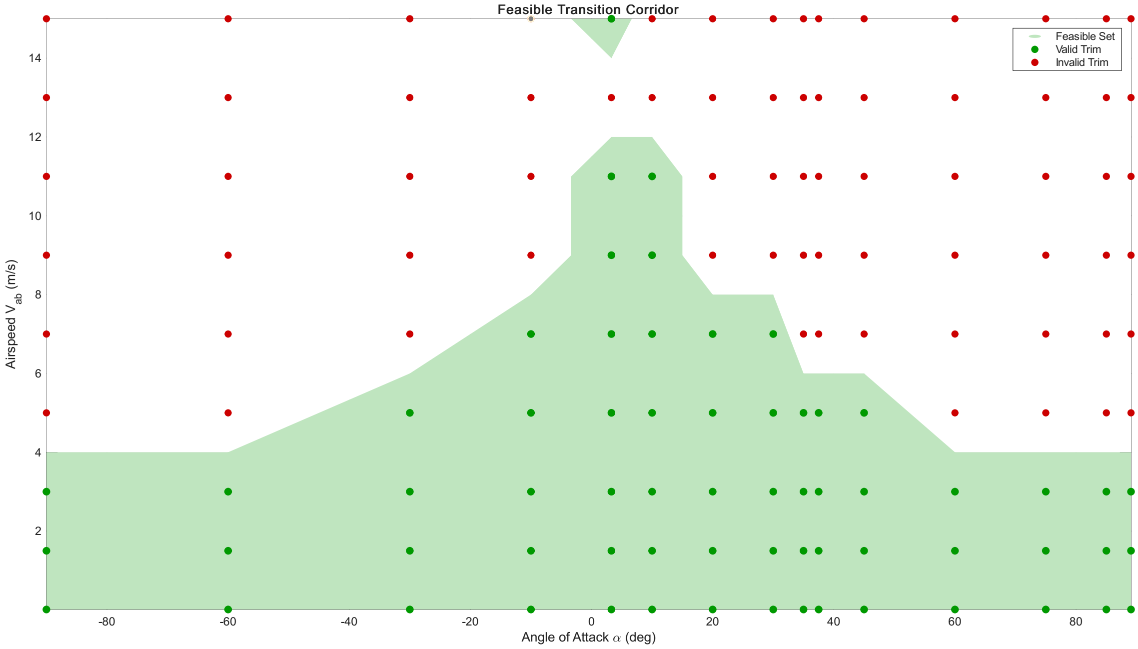


Figure 3.15: Trimgrid

3.7 LTI Model Linearization and Pre-processing

3.7.1 State reduction and scaling

The states are first defined as inertial positions, body velocities, quaternions and angular rates.

$$X = \begin{bmatrix} x & y & z & v_{bx} & v_{by} & v_{bz} & q_w & q_x & q_y & q_z & p & q & r \end{bmatrix}^T$$

State reduction is applied for the controller design. The position states are removed as linearization in later steps are incompatible with position dependant dynamics. As discussed in section **??**, the state q_w is a constraint of quaternions and is also cut. Therefore a reduced state-space is the following:

$$X = \begin{bmatrix} v_{bx} & v_{by} & v_{bz} & q_x & q_y & q_z & p & q & r \end{bmatrix}^T$$

Additionally, states and inputs are scaled with regards to their corresponding maximums, meaning the elements in the matrices will be of magnitude 1 and unit-less. This mitigates drastic differences in scale between the different states, making the off-diagonal cross-couplings more meaningful.

$$D_x = \text{diag}\left(\begin{bmatrix} 15_{1\times3} & 1_{1\times4} & 2\pi_{1\times3} \end{bmatrix}^T\right) \tag{3.32}$$

$$D_u = \text{diag}\left(\begin{bmatrix} 2000 & 2000 & 35 & 35 \end{bmatrix}^T\right) \tag{3.33}$$

3.7.2 Condition number and stabilizability

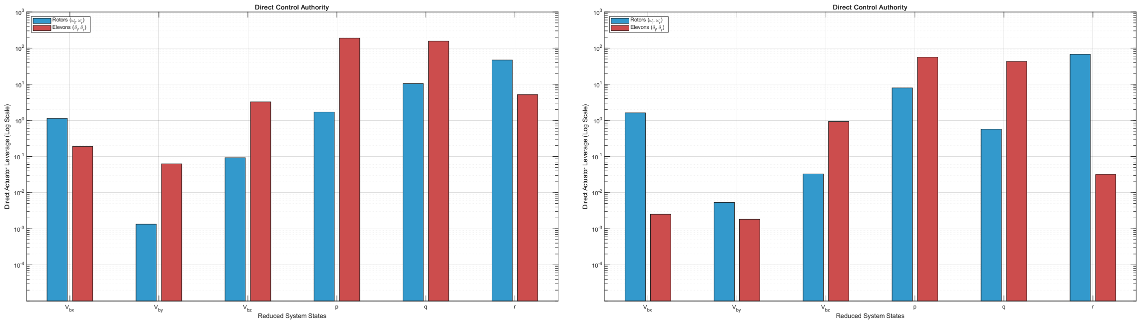
The controllability matrix for the 9 states is defined as:

$$\mathcal{C} = \begin{bmatrix} B & AB & \cdots & A^8B \end{bmatrix}$$

An eigenvalue in the A matrix of -33 becomes $(-33)^8$, this leads to extreme differences between singular values σ, also known as the condition number. For this case condition number of 10^{13}. It is used to tell if a MIMO system is ill-conditioned, however this value of 10^{13} is more of an artefact of scaling rather than an accurate indication of the system.
With a system that includes RHP poles, one cannot and should not look at the stability but instead stabilizability. A PBH test is implemented as described in section 2.4 that confirmed stabilizability.

3.7.3 Control authority over the flight envelope

Control authority is essential for understanding how the aircraft can be stabilized and controlled across the flight envelope. For a under-actuated system with 4 control inputs commanding 9 states this is needed as, states cannot be independently controlled as the dynamics are strongly coupled. The figure 3.16 quantifies to what degree the states are affected by calculating the magnitude of the B matrix elements.



(a) Cruise flight **(b)** Hover flight

Figure 3.16: Control authority

One important note is that the y-axis is in log scale starts from 10^{-5} meaning several of the states posses control authority below 10^0. The key observations from the figure are:

- Angles are pure integrators of the attitude rates and the control signals do not map directly to angles. Therefore, they are not plotted.
- No significant control authority over state V_{by} at either cruise or hover.
- Attitude rates maintain high authority in both cruise and hover, indicating that the propeller slipstream modelling works.
- Elevons have larger authority over velocities in cruise compared to in hover.
- Yaw rate is primarily controlled by the rotors, as expended from differential thrust.

3.8 Open Loop Analysis

While all valid grid points are used for the LPV model and subsequent control synthesis, the frequency response visualization is limited to six points of interest. These span the intended flight profile including cruise, transition and hover regimes, to illustrate the full range of dynamics across the envelope without cluttering the figures.

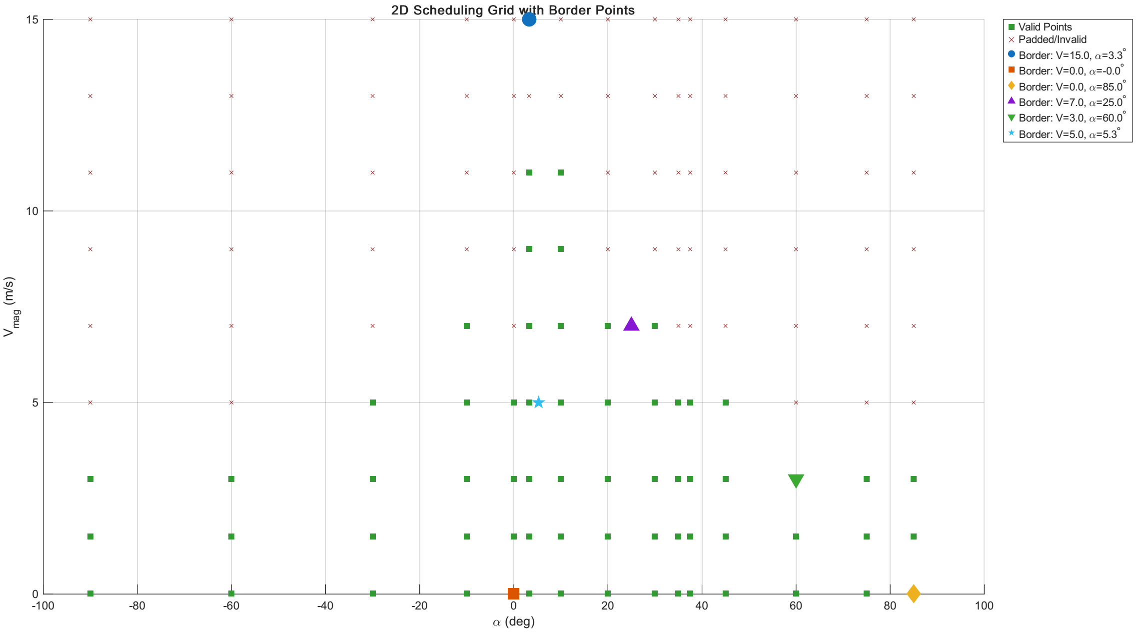


Figure 3.17: Locations of the six selected trim-points of interest in the trim-grid

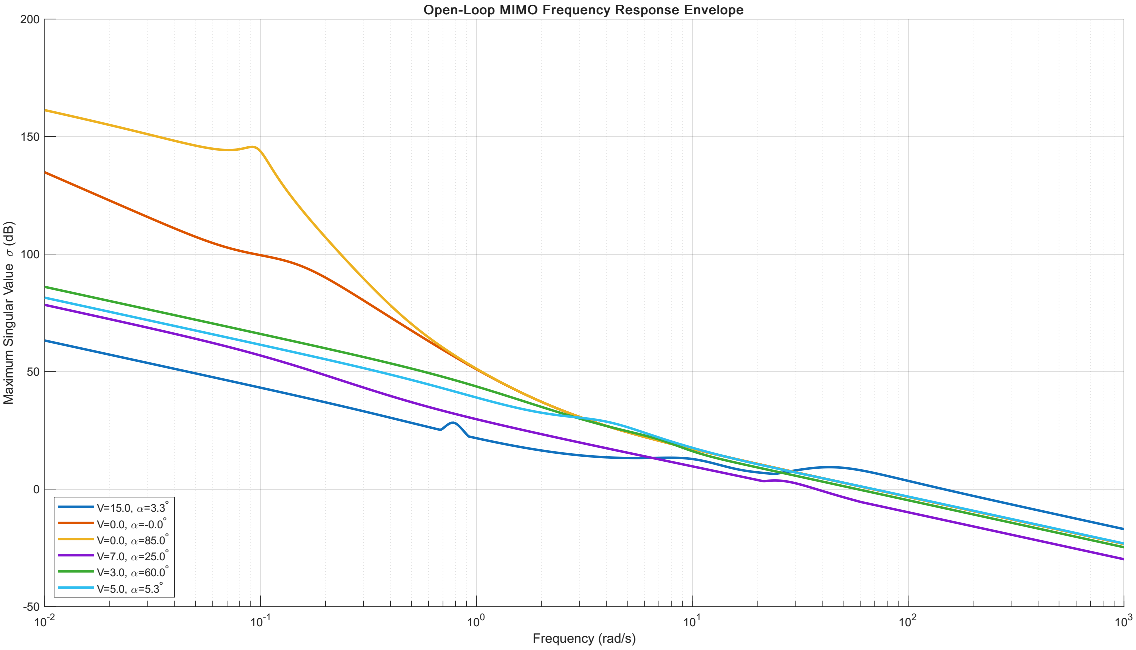


Figure 3.18: LPV corners open-loop MIMO frequency response

As seen in Fig. 3.18, there is a 100 dB spread of the singular values at the low frequencies depending on the trim condition. There is also a convergence at the approximate frequency of 1-30, where the inertia and actuators dynamics start to dominate. This confirms that a single LTI controller is insufficient to meet performance requirements across the full envelope. The high frequency natural rollof is significant and will be expanded upon in section 3.10.1.

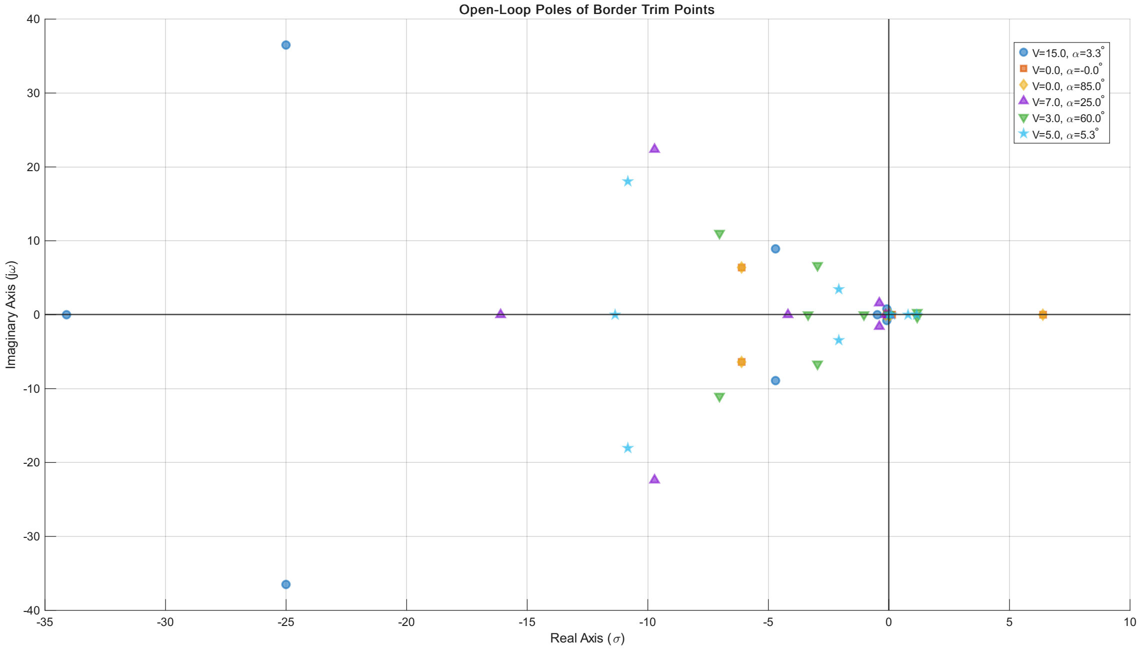


Figure 3.19: Pole locations of border points

Key takeaways from the pole locations. At higher velocities the dynamics are naturally stable and as α grows, poles moves in a positive direction one the real axis, meaning they grow faster and more unstable.

3.9 LPV Model Development

A grid-based Linear Parameter Varying (LPV) is developed to capture the changing dynamics across the flight envelope with the scheduling parameters chosen in section 3.6.1.

$$\boldsymbol{\rho}(t) = \Big[|\mathbf{V}_a| \quad \alpha\Big]^T$$

Adapted for the chosen structure, the grid-based LPV has the following equations.

$$\dot{x} = \dot{x}_0(\rho) + A(\rho)\Big(x - x_0(\rho)\Big) + B(\rho)\Big(u - u_0(\rho)\Big) \tag{3.34}$$

$$y = y_0(\rho) + C(\rho)\Big(x - x_0(\rho)\Big) + D(\rho)\Big(u - u_0(\rho)\Big) \tag{3.35}$$

$$x(0) = x_{\text{init}} \tag{3.36}$$

Figure 3.20: NL (yellow), LPV (blue) LTI (orange) - Initial validation of the LPV performed by flying with a joystick.

3.9.1 Scheduling Parameter Filtering

When aerodynamic disturbances and wind gusts are introduced to the simulation environment, the resulting high-frequency fluctuations in velocity and angle of attack cause rapid switching across the LPV polytope. To prevent the subsequent failure due to rapid switching of operatign points, low-pass filters are applied to smooth the raw scheduling parameters before they enter the gain-scheduling matrix:

$$\begin{aligned} G_{|V_a|}(s) &= \frac{1}{0.1s+1} \\ G_\alpha(s) &= \frac{1}{s+1} \end{aligned} \tag{3.37}$$

The scheduling parameter α is blended to 0 as velocity approaches hover. This is in order to avoid singularities and unnecessary interpolation between controllers with unstable poles seen in figure 3.19. The chosen values and blending strategy were determined through iterative tuning.

$$\alpha_{\text{scheduled}} = \begin{cases} 0 & \text{if } |V_a| \leq 0.7 \\ \frac{|V_a|-0.7}{0.8} \cdot \alpha_{\text{aero}} & \text{if } 0.7 < |V_a| < 1.5 \\ \alpha_{\text{aero}} & \text{if } |V_a| \geq 1.5 \end{cases}$$

3.10 Robust Control Synthesis

The H_∞ synthesis follows the mixed sensitivity framework presented in section 2.6.5, extended to include output multiplicative uncertainty. The regulated outputs are the following:

$$\left\| \begin{pmatrix} W_e S \\ W_u K S \\ W_p T \\ W_{oM} \end{pmatrix} \right\|_\infty < \gamma \tag{3.38}$$

The resulting signal structure can be seen here:

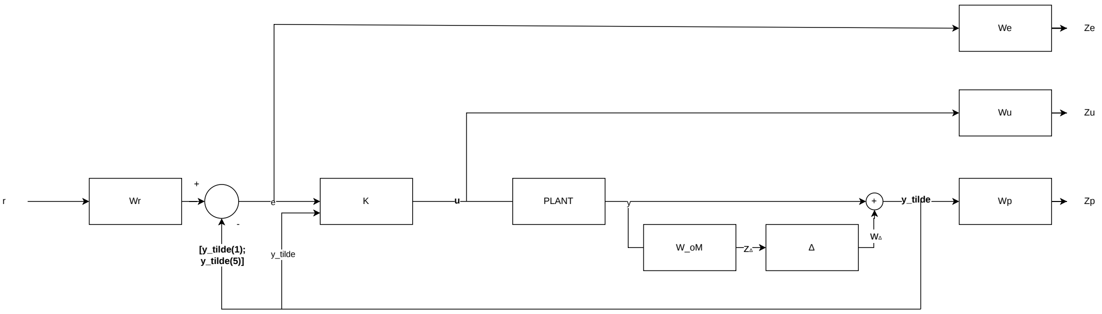


Figure 3.21: Block Diagram

3.10.1 We - Error tracking weight

The error tracking weight, targets the low frequencies to allow the controller to stabilize. The two tracked signals are forward velocity (V_{bx}) and the pitch angle (q_y). Higher DC gain for pitch angle to allow the controller to have some room to work in.

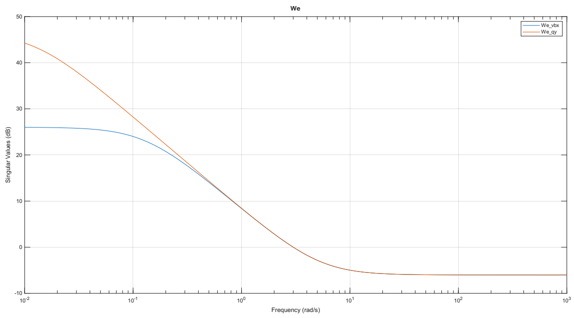


Figure 3.22: Caption

3.10.2 Wu - control effort weight

The control weight Wu penalizes control effort across frequency to prevent actuator-saturation. It is built with two separate weights for the dynamics of motors and elevons. Both weights share a crossover frequency of 20 rad/s. Below this, control effort is cheap and hence, the controller is free to use the actuators for tracking.

Above it, effort becomes expensive, preventing high-frequency oscillations from actuator activity. The high-frequency gain is set above 1 to ensure the synthesis does not attempt control action beyond the actuator bandwidth.

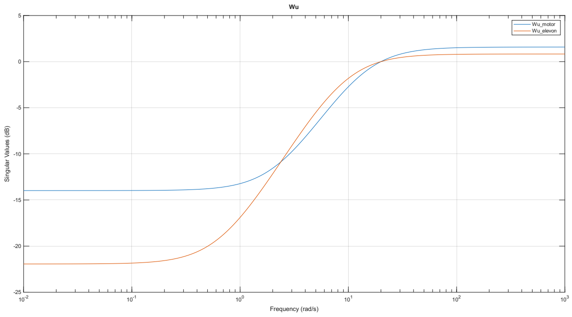


Figure 3.23: Wu

3.10.3 Wp - Performance Weight

The performance weight W_p helps prevent the controller from using unwanted states by cross-coupling. The structure of W_p for V_{by} is a low-pass structure, penalizing the steady-state drift. This allows the controller to, when a sideslip velocity is induced, ignore it in order to stabilize first before then addressing the lateral drift. High-pass structure on the angular rates was tuned to penalize high-frequency behaviour, slowing down rotations to retain stability.

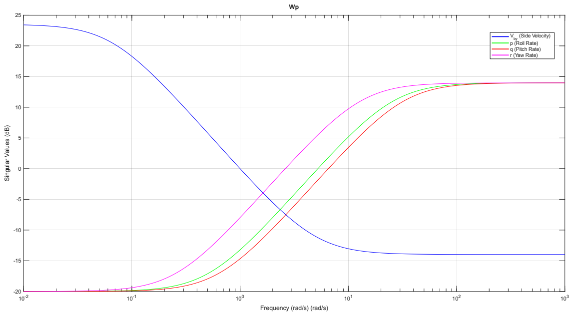


Figure 3.24: Wp

3.10.4 $\mathbf{W_{oM}}$ - Output Multiplicative Uncertainty Weight

As observed in 3.19, the unstable poles of the trimgrid grew faster with higher angle of attack α. This is directly tied to the coefficients related to aerodynamic moments. An output multiplicative uncertainty was then chosen to reflect these unstable poles as highpass, growing linearly with α on the angular rates p, q, r.

$$W_{oM} = W_{oM,base} \cdot (1 + \alpha)$$

A benefit of this is that the outline of the trimgrid has the highest uncertainty, meaning if the scheduling parameters go outside the valid region the nearest valid linearization will have high uncertainty.

3.10.5 Summary

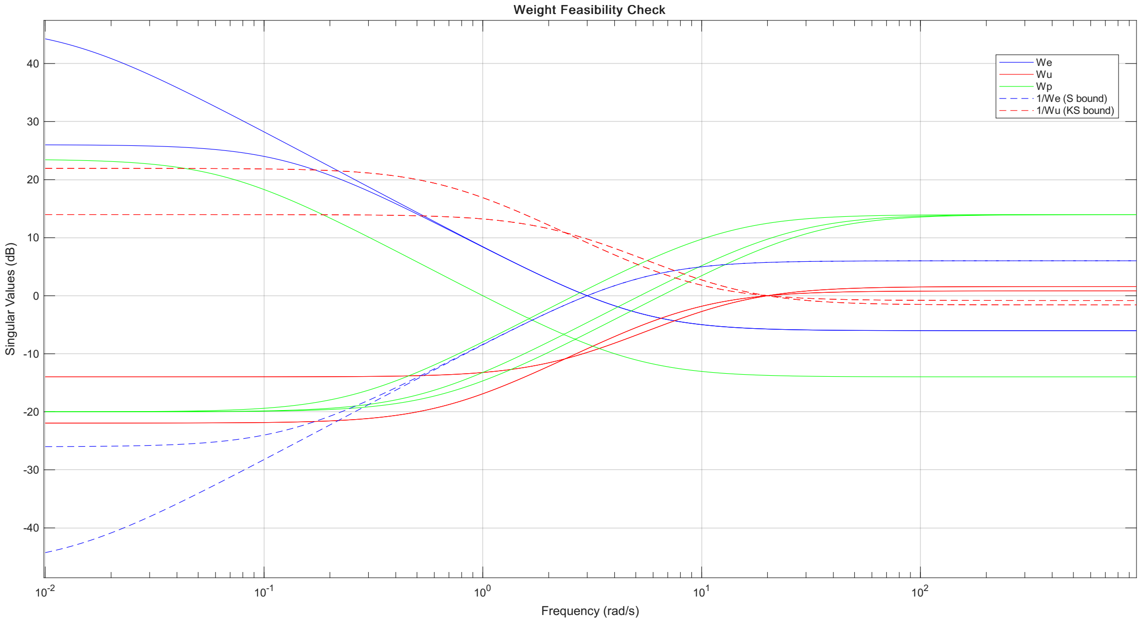


Figure 3.25: Weight Feasibility

Table 3.5: H_∞ synthesis weight parameters

Filter	**Description**	**DC gain**	**Crossover (rad/s)**	**HF gain**
Reference model — W_r				
$W_{r,v}$	Velocity reference	1	3	0
$W_{r,q}$	Pitch reference	1	3	0
Error weight — W_e				
$W_{e,v}$	Velocity error	20	3	0.5
$W_{e,q}$	Pitch error	210	3	0.5
Input weight — W_u				
$W_{u,\text{motor}}$	Motor authority	0.20	20	1.2
$W_{u,\text{elev}}$	Elevon authority	0.08	20	1.1
Performance weight — W_p				
$W_{p,vby}$	Side velocity	15	1.0	0.2
$W_{p,p}$	Roll rate	0.1	2.0	5
$W_{p,q}$	Pitch rate	0.1	2.5	5
$W_{p,r}$	Yaw rate	0.1	1.0	5
Output multiplicative uncertainty — W_{oM}				
W_{oM}	Rate channel uncertainty	0.01	5.0	$0.30 + 0.8\,\lvert\alpha\rvert/90^\circ$

3.10.6 Synthesis & Verification

Figure 3.26 showcases the gamma values of each synthesised controller. Gamma values below 1 as mentioned in section 2.6.5 indicate that the weight tuning is achievable for the controller. Values higher than 1 are near the edges of the polytope, reflecting both the stall regions where control authority is lower and the uncertainty tuned. There are three clear outliers with high gamma and another four that are empty. These points represent a velocity and flight direction where the controller does not have a large enough authority to meet the requirements specified in the tuning. Other points fail completely indicating unfeasibility.

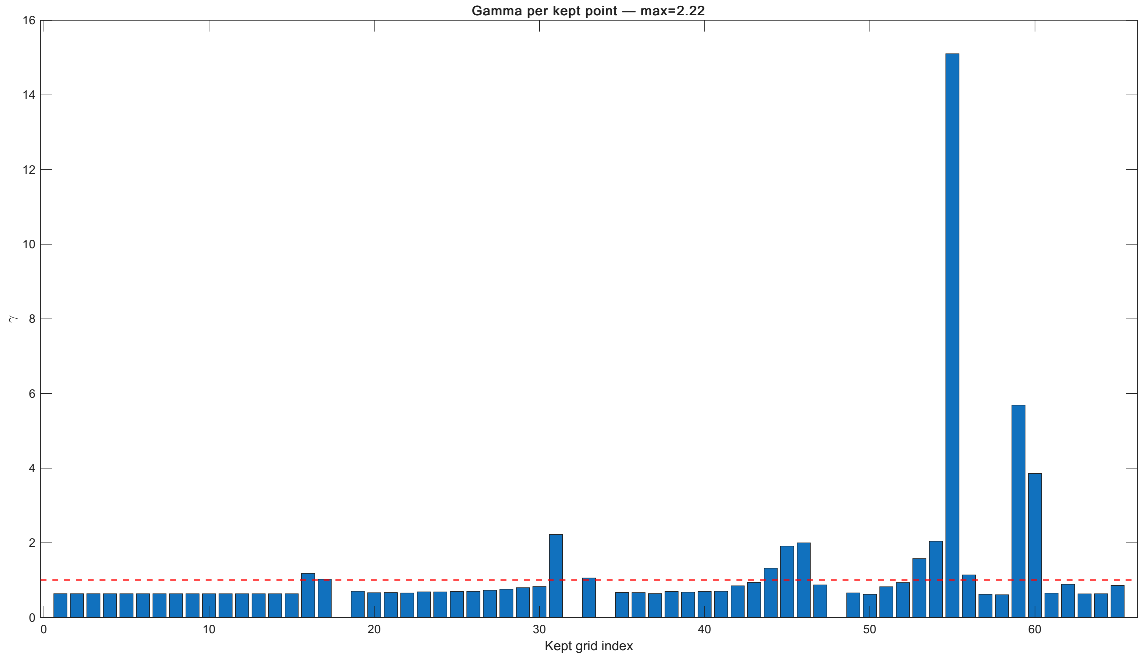


Figure 3.26: Controllers - outliers

These points are removed from the controller grid and are showcased here in Fig. 3.27.

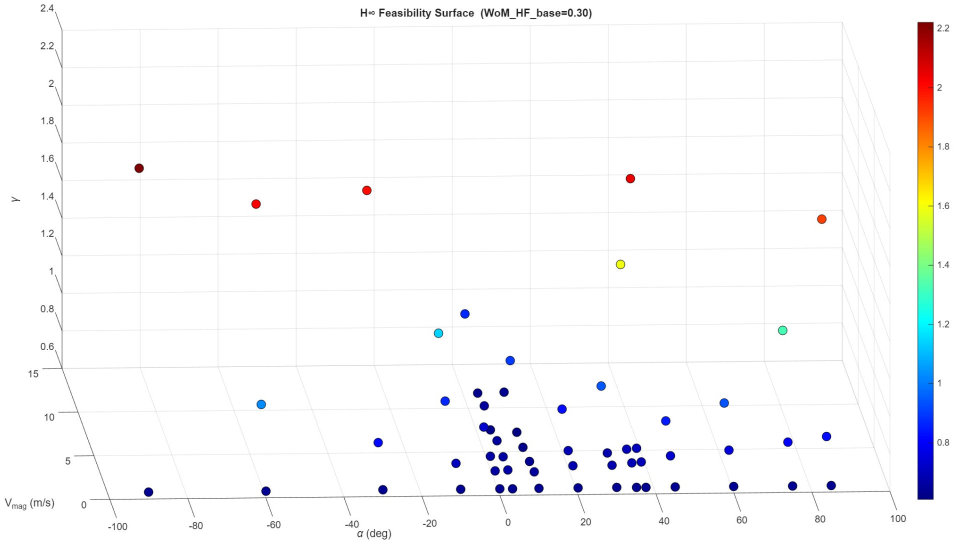


Figure 3.27: H_∞ feasibility surface

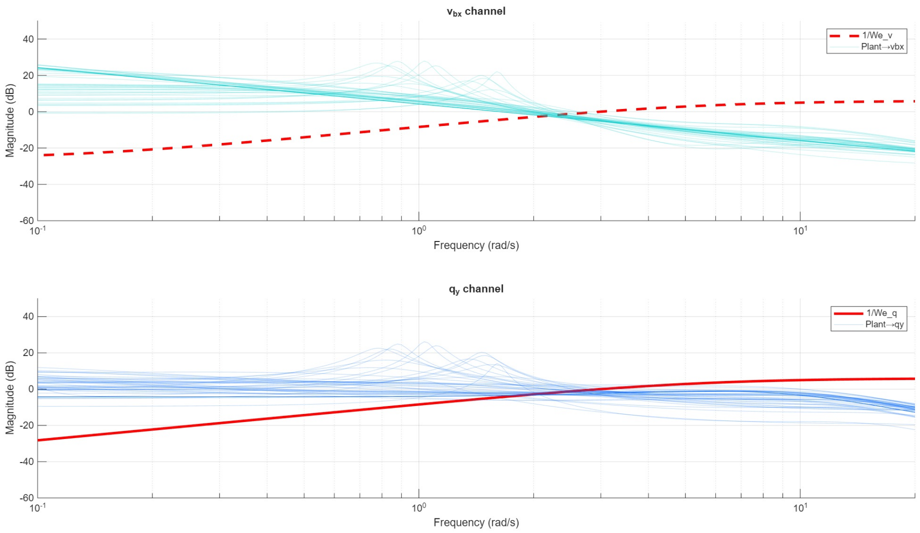
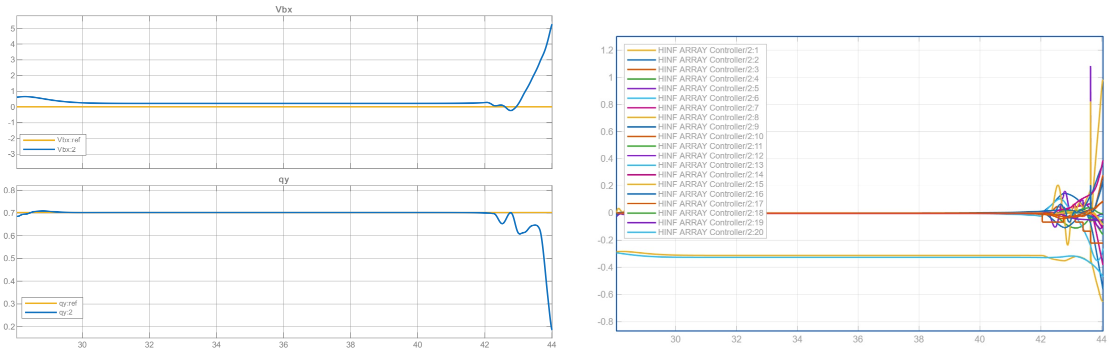


Figure 3.28: Plant gain vs performance bounds (all kept grid points)

Figure 3.28 shows the plant gain envelope across all retained grid points against $1/W_e$. At low frequencies the plant gains spread significantly between operating points, reflecting different dynamics across the scheduling grid. The gains converge at higher frequencies. The crossover frequency of W_e is placed in this convergence region, ensuring the performance bound is representative of all operating points simultaneously rather than over constraining any single one.

3.10.7 Anti-windup

The observed controller divergence seen in Fig. 3.29 was observed to be caused a controller state K(s) having a slow but constant drift. It was tracked down to the the filter W_e which had a near-integrator pole of $s = -0.0434$ for state V_{bx}. This allowed a small but constant offset to occur. By increasing the low frequency DC value for the filter W_e, the pole was moved closer to the imaginary axis, $s = -0.0017$. This yields a stronger integrating pole which in turn eliminates the steady-state error. As a result the controllers state drift is bounded for the simulation.



(a) Reference tracking **(b)** Controller states

Figure 3.29: W_e error windup explosion

3.11 Gain scheduled LQR with feedforward

An LQR controller with feedforward and gain scheduling was developed as a baseline controller for validation. Index k refers to the gridpoint index. Implementation uses a state vector of 9 and 4 inputs. The steady-state condition is solved at each grid point. The equations in 2.7 becomes.

$$M = \begin{bmatrix} A_k & B_k \\ C_{\text{track}} & 0 \end{bmatrix}, \qquad \begin{bmatrix} x_{ss} \\ u_{ss} \end{bmatrix} = M^{\dagger} \begin{bmatrix} 0 \\ I \end{bmatrix} \tag{3.39}$$

Feedforward gain calculated as:

$$\bar{N}_k = N_u + K_k N_x \tag{3.40}$$

The state cost matrix Q and input cost matrix R are tuned such that the tracking penalty for pitch q_y is the highest in Q. Any use of v_{by} is penalized to suppress sideslip effects, as they are not accurately captured at the linearized points. Elevons are penalized more heavily in R than rotors to avoid oscillations and limit saturation.

$$Q = \operatorname{diag}\Big(\begin{bmatrix} 5 & 100 & 5 & 10 & 250 & 10 & 30 & 50 & 30 \end{bmatrix}\Big) \tag{3.41}$$

$$R = \operatorname{diag}\Big(\begin{bmatrix} 1.5 & 1.5 & 8.0 & 8.0 \end{bmatrix}\Big) \tag{3.42}$$

The closed-loop dynamics at grid point k become:

$$\dot{x} = (A_k - B_k K_k)\, x + B_k \bar{N}_k\, r \tag{3.43}$$

The control law across the grid becomes:

$$u = -K(s)\, x + \bar{N}(s)\, r \tag{3.44}$$

Here $s = (V_{\text{mag}}, \alpha)$ matches the LPV scheduling parameters to ensure consistent gain interpolation. The grid that the controllers are synthesised on is the valid grid established in Fig. 3.27. This is motivated by the fact that the outliers have the same properties for H_∞ synthesis as they do for LQR, it also allows for direct comparisons between the different interpolation methods between gain scheduling and LPV.

3.12 Interpolation and Extrapolation

Both the H_∞ and LQR controllers are interpolated between the trim-points, using linear interpolation. In terms of extrapolation, when scheduling parameters go outside of the valid grid, neither of the two controllers extrapolates beyond it but instead clips the interpolation and uses the closest valid interpolated value from the valid trim grid.

3.13 Simulation setup

3.13.1 Handling Stiff Dynamics

Initial open-loop analysis of the obtained dynamics signalled significant stiffness, which is attributed to the low inertia in relation to surface area of the aircraft. To enable simulation without unrealistic actuator stiffness, actuator dynamics and saturation were employed during simulation, shown in Equation 3.45. Elevon servos and motor spool-up dynamics were modelled as first order transfer-functions with approximative bandwidths that represent typical time constants of small-scale brushless motors and micro-servos.

$$\begin{aligned} G_{\text{rotor}}(s) &= \frac{1}{0.02s+1}, \qquad \omega \in [0,\ 2000]\ \text{rad/s} \\ G_{\text{elevon}}(s) &= \frac{1}{0.05s+1}, \qquad \delta \in [-35^\circ,\ 35^\circ] \end{aligned} \tag{3.45}$$

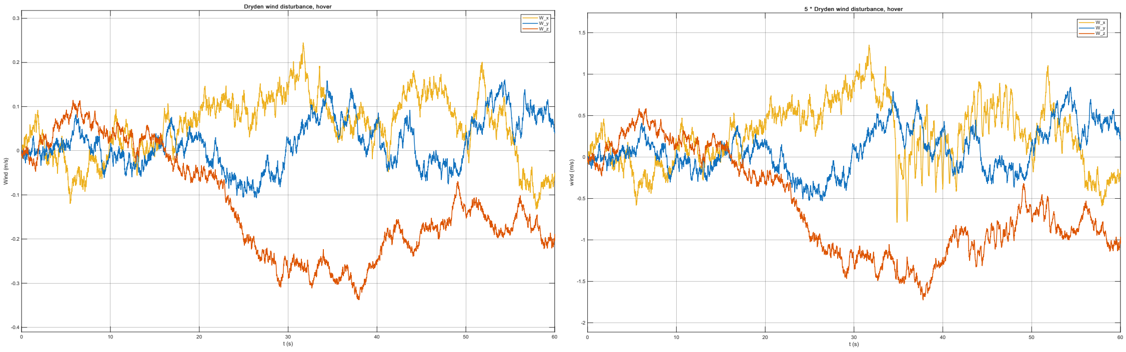
Additionally, Simulink's variable-step ODE23t solver was used throughout all evaluations to prevent numerical failure. This approach was chosen over fixed-step alternatives to prioritize accuracy over real-time simulation capabilities.

3.13.2 Wind Disturbance Modelling

In order to model realistic conditions, the Dryden Wind Gust Model is utilized. The continuous low-altitude Dryden turbulence model however assumes the vehicle's airspeed is significantly larger than the turbulent wind velocity, which causes numerical instability at near-zero speeds in hover. To mitigate this, a surrogate constant airspeed of 5 m/s was applied to the Dryden model inputs, avoiding the singularity. This approach follows methodology for rotorcraft simulation used in [12].

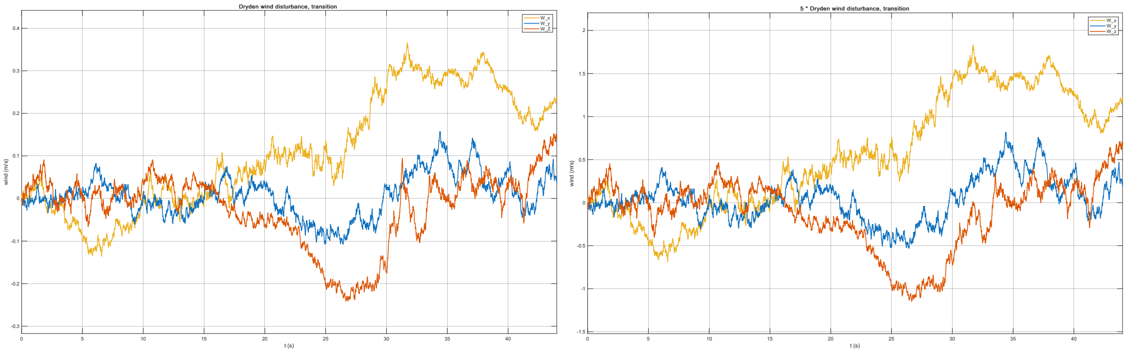
While the established litzerature utilizes both translational and rotational Dryden disturbances, the rotational rate components were omitted because their inclusion caused solver divergence during simulations. Consequently, only translational components (W_x, W_y, W_z) were utilized.

The resulting wind disturbance signals for a few selected scenarios are shown here in Fig. 3.30 and 3.31. Each simulation uses the same random generator seed for the wind. The obtained signal differs between each run since the Dryden model is a function of aircraft attitude and the obtained wind magnitude is scaled by a `wind_gain` to test different wind conditions.



(a) `wind_gain` 1 (b) `wind_gain` 5

Figure 3.30: Hover wind



(a) `wind_gain` 1 (b) `wind_gain` 5

Figure 3.31: Transition wind

3.14 Testing scenarios

3.14.1 Transition Trajectories

Since the transition maneuver from forward cruise to hover operates across a wide aerodynamic envelope, reference trajectories must be carefully generated. To ensure the controllers maintain performance guarantees, the transition maneuver must strictly remain within the valid scheduling polytope. By selecting a set of waypoints in the polytope, reasonable transition trajectories that satisfy this can be generated. This in turn enables controller evaluation without the need to develop a cascaded guidance loop.

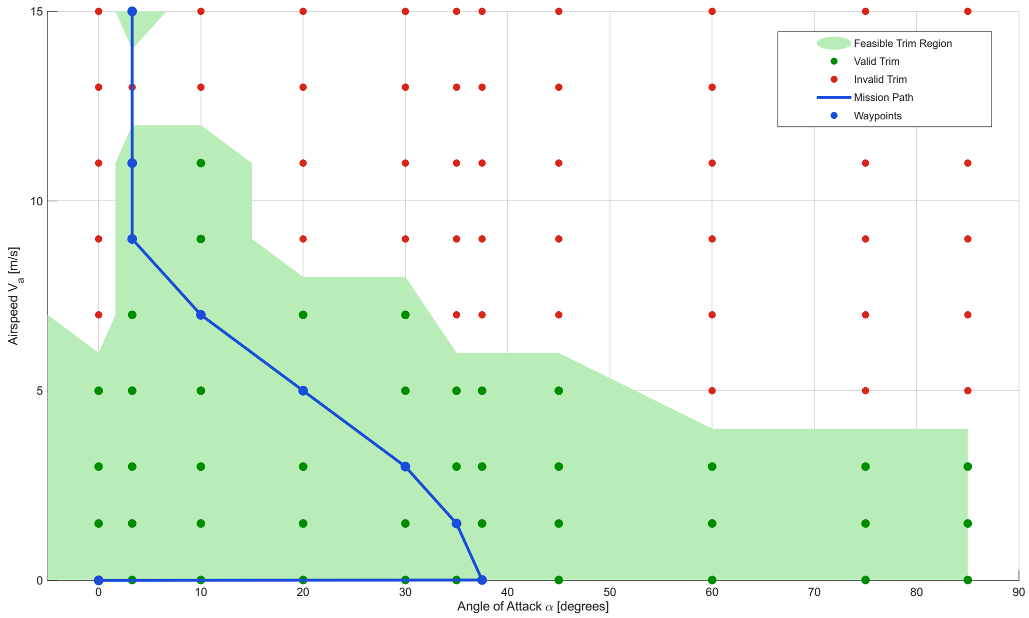
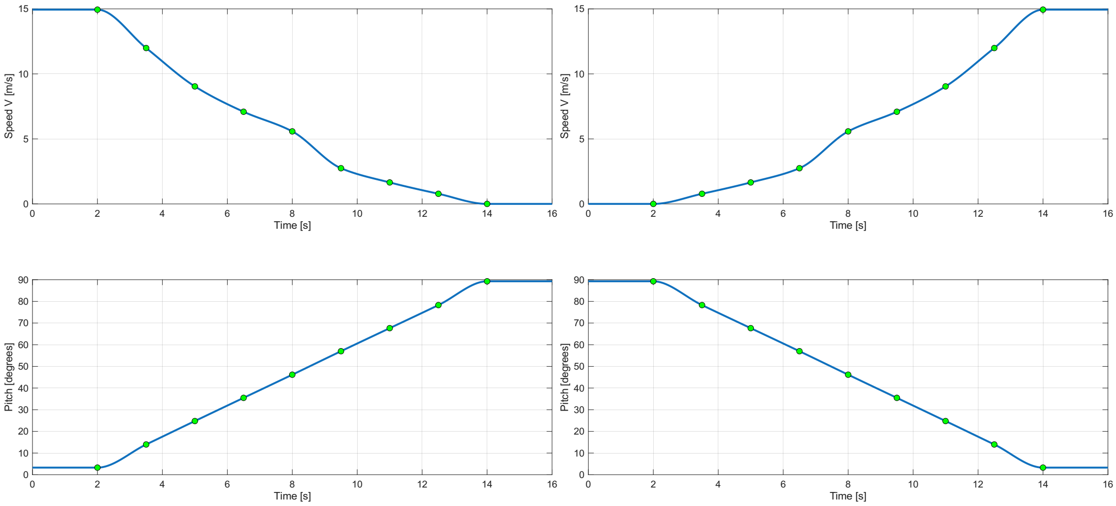


Figure 3.32: Transition trajectory path through LPV polytope scheduling space.

The reference generator extracts discrete waypoints from the LPV grid by pairing speed increments with specific angle-of-attack targets. These α targets are tuned to balance unwanted altitude climb during transition against robust controller performance. To achieve a constant pitch rate during transition, these waypoints are spaced in time according to their respective pitch angles. Smooth continuous reference signals are subsequently generated using Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) between these waypoints, shown in Fig. 3.33.



(a) Cruise → Hover **(b)** Hover → Cruise

Figure 3.33: State reference trajectories generated via PCHIP interpolation for transition maneuvers.

4

Results

Throughout this chapter, units for pitch has been converted to degrees to improve readability. Tables are presented to support the figures, not all data is displayed to reduce clutter. Some simulations and step-responses have been moved to Appendix B in order to have a more concise result chapter. The results are separated into three sections: first, validation of controllers and interpolation. The other two sections test the performance of the controller under disturbances. They directly addresses the research questions stated in section 1.3, which are:

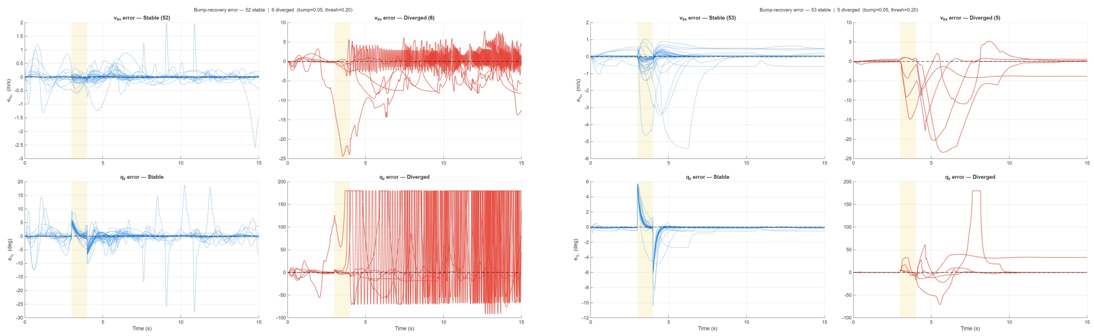
RQ1: Can a model-based, robust controller stabilize the tailsitter through transition from forward flight to hover and stabilize it in hover?

RQ2: What are the operational limits of the implemented controllers in terms of static and gusty wind conditions?

4.1 Validation tests (no disturbance)

Validation of the interpolation is separated into two complementary tests. A bumptest that assesses instability of the controller grid followed by a nominal transition to test if the interpolation in LPV and gain scheduling cause instability. Together, these establishes that the simulation setup is correct and showcases a baseline performance before adding introducing wind disturbances.

4.1.1 Bumptest



(a) H_∞ batch **(b)** LQR batch

Figure 4.1: Bumptest

Interpolation for both the H_∞ LPV and gain-scheduled LQR controllers is functioning correctly, and the control syntheses are valid. It showcases that five H_∞ and six LQR controllers are diverging, while the remaining 52-53 controllers are valid.

4.1.2 Nominal transition

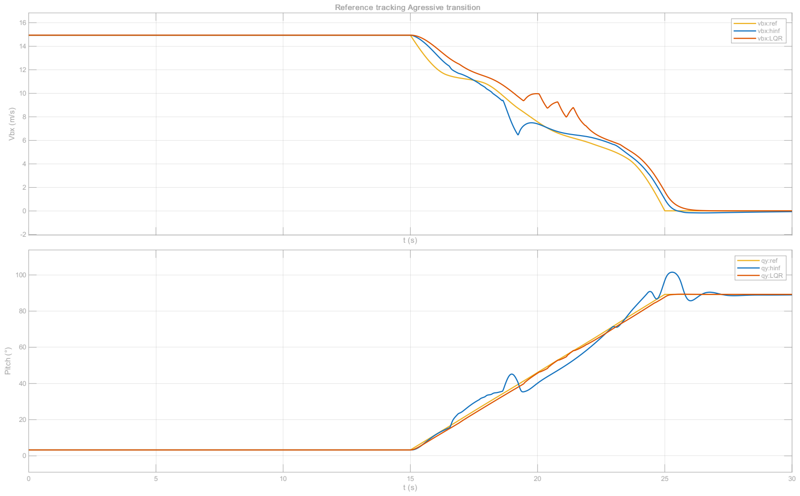


Figure 4.2: Aggressive transition

Transition from cruise to hover. This confirms trimgrid validity and that the simulation setup is working correctly. Observe that in nominal conditions, both controllers accurately track the reference trajectory.

4.2 Transition performance with wind

This section addresses **RQ1** directly with a transition of (Cruise → Hover → Cruise) with disturbance.

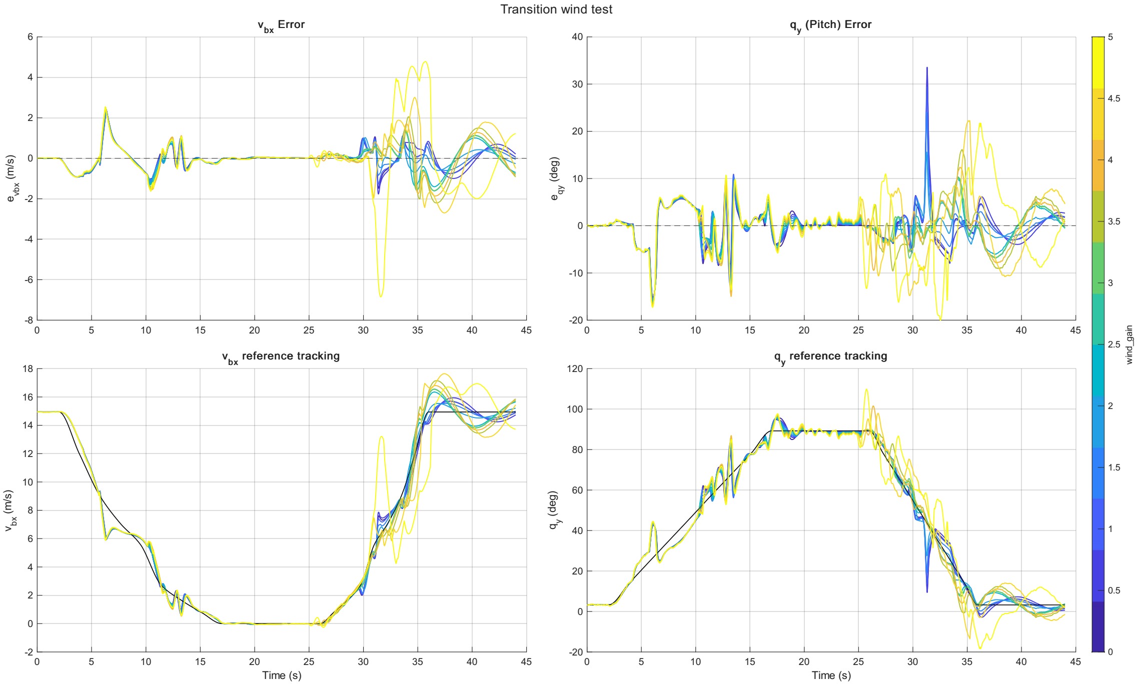


Figure 4.3: H_∞ transition

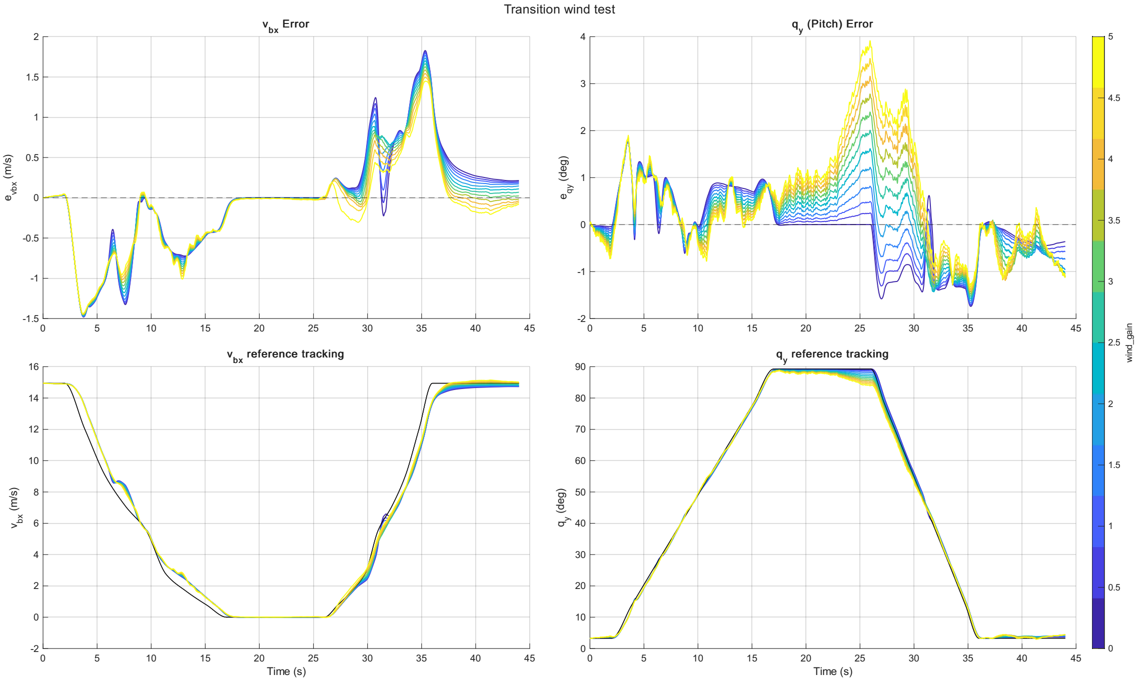


Figure 4.4: LQR transition

Table 4.1: Transition wind test — H_∞ vs. LQR

	$\mathbf{H_\infty}$		**LQR**	
Wind gain	**Peak** $\|e_q\|$ **(°)**	**Bias** e_q **(°)**	**Peak** $\|e_q\|$ **(°)**	**Bias** e_q **(°)**
0.00	32.91	−0.06	1.76	−0.38
0.30	33.57	+0.11	1.76	−0.40
0.60	29.37	−0.07	1.77	−0.41
1.00	24.02	−0.18	1.78	−0.46
1.50	15.88	−0.32	1.79	−0.50
2.00	16.10	−0.42	1.81	−0.52
2.50	16.31	−0.50	2.01	−0.54
3.00	16.52	−0.47	2.40	−0.55
3.50	16.72	−0.43	2.78	−0.54
4.00	16.91	−0.43	3.16	−0.52
4.50	22.25	+0.46	3.54	−0.49
5.00	21.74	+3.37	3.92	−0.46
5.50	DIVERGED		DIVERGED	

4.3 Hover performance with wind

This section addresses **RQ2** directly with a stay in hover reference with disturbance.

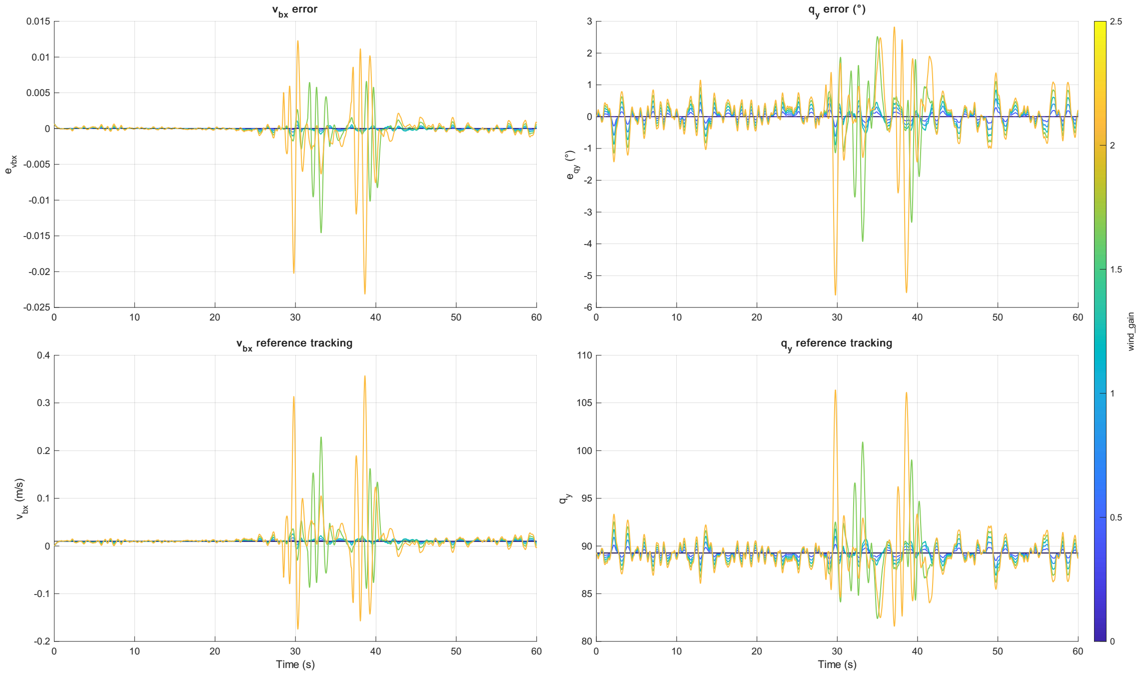


Figure 4.5: H_∞ hover

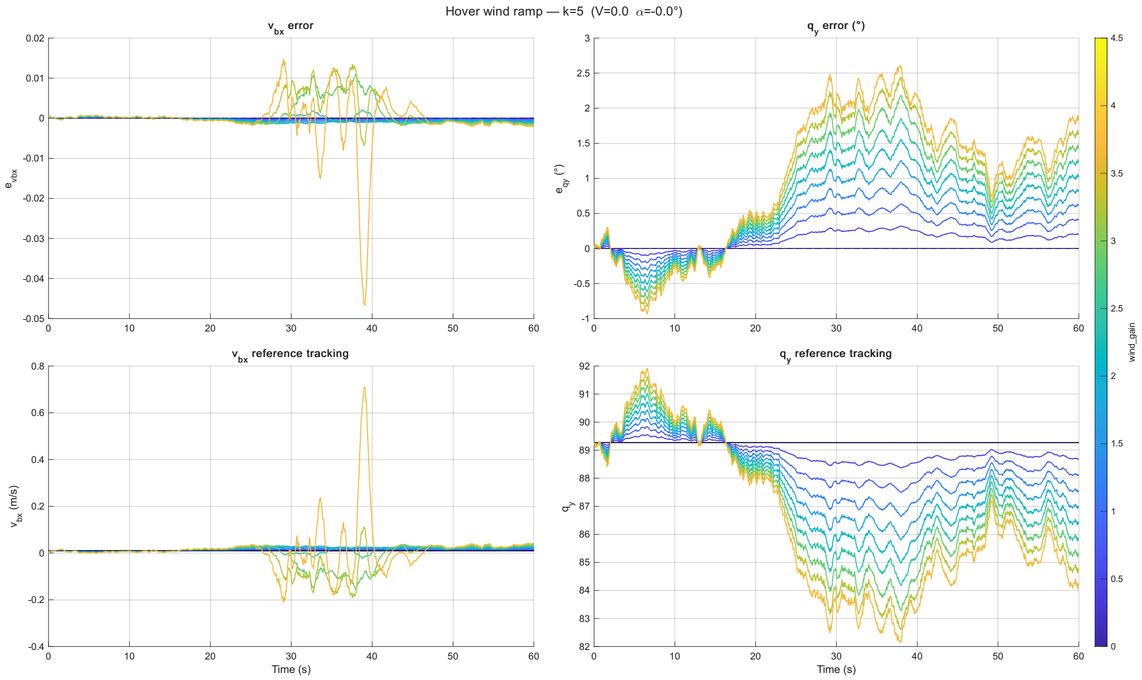


Figure 4.6: LQR hover

Table 4.2: Hover wind ramp — H_∞ vs. LQR

	H_∞		LQR	
Wind gain	**Peak $\lvert e_q \rvert$ (°)**	**Bias e_q (°)**	**Peak $\lvert e_q \rvert$ (°)**	**Bias e_q (°)**
0.0	0.00	+0.00	0.00	+0.00
0.5	0.28	+0.01	0.21	+0.15
1.0	0.56	+0.01	0.42	+0.31
1.5	0.84	+0.02	0.63	+0.46
2.0	1.11	+0.02	0.84	+0.61
2.5	1.38	+0.02	1.06	+0.76
3.0	DIVERGED		1.27	+0.91
3.5			1.48	+1.06
4.0			1.69	+1.20
4.5			1.89	+1.35
5.0			6.94	+0.35
5.5			DIVERGED	

5

Discussion

This chapter discusses the results obtained when simulating the nonlinear model and the two synthesised controllers.

5.1 Validation and interpolation stability

Outliers were removed with regard to gamma values and stability as outlined in Section 3.10.6. Results of the nominal bumptest revealed that several grid points (H_∞ 6, LQR 5) showed divergence, requiring removal.

The most important detail from bumptest is the instability behaviour. LQR's lack of integral action, together with the cost function $J = \int (x^T Q x + u^T R u) dt$ being valid regardless of initial condition, make it stabilize, although the divergence can cause it to stabilize in the wrong operating point.

Nominal simulation showcased the successful aggressive transition into hover then descent. There was little to no drift in y-position or angles in roll and yaw, along with an altitude gain of 12 meters. The simulation showcases that interpolation works and the validity of the reference trajectory.

5.2 Transition Performance In Wind

Both controllers successfully transition through the flight envelope. The result shows an aggressive angle of attack transition from cruise to hover then back to cruise over a 45 s simulation.

Both cases of H_∞ and LQR achieved successful simulations up to a wind gain of 5 before divergence. This indicates that for transitions, the limiting factor is the plant itself rather than controller limited performance. The nature of our ultralight UAV has an operational limit of 5 in wind gain.

During transition, while H_∞ maintains lower bias, it exhibits larger peak errors than LQR, a tendency to overshoot.

The error plots for both controllers have peaks at 25-35 s, corresponding to the hover and re-transition phase.

5.3 Hover Performance In Wind

The hovertest in section 4.3 reveals that the LQR controller outperforms the H_∞ controller by managing twice the amplitude of disturbance (5.5 vs 3.0) before diverging.

LQR lacks integral action, resulting in steady-state errors. H_∞ also lacks an integrator property but as discussed in 3.29, the high DC gain in the error tracking filter results in integrator properties, enabling stable hover around the operating point. On a weight-constrained platform with limited control authority, this memoryless aggressive response is critical for disturbance rejection.

This is not testing the scheduling robustness; it is primarily testing disturbance rejection at the hover operating points. The LQR has a structural advantage here because as a static gain, it carries no internal dynamics. The H_∞ controller, made conservative through the uncertainty weight W_{oM}, destabilises at a lower wind amplitude. This showcases the trade-off between robustness and performance.

Why? The conservatism required to achieve robustness (W_{oM}) resulted in slower control responses. This slow response allowed errors to accumulate before correction, which then demanded larger control corrections, causing overshoot and instability. At hover, the pursuit of robustness paradoxically reduced practical robustness.

5.4 Overarching Limitations and Trade-offs

5.4.1 Grid based controller architecture

The H_∞ controllers mostly meet the condition $\gamma < 1$ with only some controllers having a higher value. This indicates that all weighted closed-loop transfer functions are bounded at their corresponding operating condition. Despite this, the H_∞ LPV controller failed at a higher rate and had less tracking performance.

The frozen point H_∞ synthesis does not provide any guarantee of stability for time varying $\rho(t)$ outside of operating points. The full stability condition states that the term $\frac{\partial P(\rho)}{\partial \rho}\dot{\rho}$ is bounded but it is not included in the point-wise Riccati equation.

Gain-scheduled LQR is built with static gains $K(\rho)$; it does not have internal dynamics. This means no controller states are affected by time varying systems. This is a drawback but also as in this case, an advantage. Scheduling only modifies $A(\rho)$ instantaneously, leaving no accumulated controller states to destabilize.

5.4.2 Cases where H_∞ is better suited

While not covered by the simulation scenarios, if the model parameters such as the inertia matrix or centre of gravity deviate from the real platform, H_∞ robustness could in theory provide advantages when it comes to handling these model uncertainties. In order to assess this robustness properly against the LQR, model uncertainties should have been modelled and added during simulation.

6

Conclusion

The main object of this thesis was to investigate robust, autonomous landing of an ultralight 250 gram tailsitter UAV. This was pursued by employing model based robust control, requiring extensive modelling to be performed, extracting aerodynamic coefficients from CFD simulations. Two controllers were synthesized and evaluated being one H_∞ LPV controller and one gain scheduled LQR controller with feedforward. The results were obtained through simulation to answer the two research questions in Section 1.3.

6.1 RQ 1: Stable transition

Can a model-based, robust controller stabilize the tailsitter through transition from forward flight to hover and stabilize it in hover?

Yes, the two controllers successfully transition through the dangerous transition envelope from cruise to hover. The gain-scheduled architectures effectively capture the nonlinearities throughout the flight envelope demonstrated by the 100 dB spread in singular values in Fig. 3.18. Simulation validation confirmed stability at different operating points and when tracking the transition trajectories.

6.2 RQ2: Operational limits

What are the operational limits of the implemented controllers in terms of static and gusty wind conditions?

Both controllers achieve successful transitions during windy conditions up to a limit of the wind gain 5.5. This indicates a plant based limit rather than a controller tuning limitation. In hover the operational limit is approximately 5.5 as well, achieved by the LQR control architecture. LQR manages to be twice as effective in handling disturbances in hover compared to H_∞, indicating that its fast responsive structure is beneficial. A summary on the obtained limits are shown here in table 6.1

Table 6.1: Controller Comparison Summary

Metric	H_∞	**LQR**
Transition wind limit	5.5	5.5
Hover wind limit	3.0	5.5
Peak error (hover, max stable)	1.38°	6.94°
Mean bias (hover, max stable)	+0.02°	+0.35°
Response	Slow	Fast
Primary strength	Steady-state	Disturbance rejection

6.3 Implications

Marine environments common in the intended sea rescue application, require high wind rejection abilities. The concrete results show that the LQR handles a Dryden gain of 5.5 while the H_∞ handles up to 3. This indicates that even when simulating on an ideal, nominal, nonlinear model, the aircraft only handles low to moderate wind disturbances. It is primarily in or near hover, when the wind is perpendicular to the big platform surface area, that failure occurs. Subsequently, neither of the controllers can perfectly handle the realistic environments during the landing procedure.

For practical flight controller implementation, the gain-scheduled LQR architecture is more suitable. It only requires interpolating a static gain matrix across the 2D scheduling grid. In contrast, the LPV H_∞ controller requires interpolating full state-space matrices which increases the computational load and dimensionality.

Ultimately, the fundamental limitation is the physical platform rather than the control design. The ultralight aircraft has an inherently low mass-to-area ratio, making it highly sensitive to wind. In hover, wind disturbances disrupt the slipstream over the elevons, causing a severe loss of control authority. Possible solutions to this are covered in the next section.

6.4 Future work

There is a number of things that can be improved and considered in future work on this project. Among a few are:

- Adding more CFD samples to better model sideslip in hover. This would provide more accurate evaluation of the yaw stability in hover.
- Validating the aerodynamic model through wind tunnel testing. Enables capturing currently unmodelled effects such as propeller swirl effects, propeller aerodynamics (running the rotors while measuring in wind tunnel) and more sideslip effects among a few things.
- Making further modifications to the Dryden wind model which currently does not provide accurate turbulence modelling as speed approaches zero in hover. One approach would be to tune a lower bound threshold instead of constant airspeed input to the generator.

- Introduce model uncertainty in simulation to evaluate H_∞ robustness against the LQR.
- Implementing outer loop position tracking controller in conjunction with guidance algorithms that provide feasible trajectories which respect the valid trim envelope. In a ramping headwind the expected behaviour is that the drone tilts into the wind in hover in order to hold position. This is made possible by the densely populated grid of operation points from cruise to hover.
- The outer loop must also include heading angle tracking to minimize sideslip to ensure the aircraft has a headwind in during final approach for transition and landing, since crosswinds could severely degrade the elevon authority
- Addressing the singularity in α as speed approaches zero in hover, by investigating other scheduling parameters or fine tuning the workaround that blends the parameter toward zero in hover.
- Evaluating controller performance in conjunction with sensors and state estimators such as the Extended Kalman Filter and the effects of the resulting phase shifts in the scheduling parameters.
- Instead of removing the outlier controllers along the envelope edge, separate scheduled tuning might be more appropriate. The hover test specifically indicates the need for a dedicated retune.
- Investigating a tilt rotor design such as that of [13], where the entire rotor-assembly can be rotated around the pitch axis using a servo mechanism. This enhances the control authority substantially, reducing the sensitivity to wind gusts and significantly increasing the operational wind limits.

Bibliography

[1] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 2nd ed. Chichester, U.K.: John Wiley & Sons, 2005. [Online]. Available: `https://skoge.folk.ntnu.no/book/ps/bookall.pdf`.

[2] C. Paige, "Properties of numerical algorithms related to computing controllability," *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 130–138, 1981. DOI: `10.1109/TAC.1981.1102563`.

[3] M. M. Peet, *MMAE 543: Advanced Control Systems – Lecture 11: Stability of LTI Systems, Eigenvalues, and Lyapunov Functions*, Course Lecture Notes, Illinois Institute of Technology, Available at `https://control.asu.edu/Classes/MMAE543/543Lecture11.pdf`, 2013.

[4] B. Kulcsár, *EEN050: Robust and nonlinear control – Course Lecture Notes*, Lecture Notes, Department of Electrical Engineering, Chalmers University of Technology, Unpublished internal course material, Gothenburg, Sweden, 2026.

[5] Honeywell Technology Center, Lockheed Martin Skunk Works, and Lockheed Martin Tactical Aircraft Systems, "Application of multivariable control theory to aircraft control laws. final report: Multivariable control design guidelines," Wright Laboratory, Flight Dynamics Directorate, Air Force Materiel Command, Wright-Patterson AFB, OH, Tech. Rep. WL-TR-96-3099, May 1996, Approved for public release; distribution unlimited.

[6] T. Glad and L. Ljung, *Control Theory: Multivariable and Nonlinear Methods.* London: Taylor & Francis, 2000.

[7] R. Chiappinelli, "Modeling and control of a flying wing tailsitter unmanned aerial vehicle," M.S. thesis, McGill University, Montreal, Quebec, 2018.

[8] E. Enenakpogbe, J. F. Whidborne, and L. Lu, "Full envelope control of over-actuated fixed-wing vectored thrust eVTOL," *Aerospace*, vol. 11, no. 12, p. 979, 2024. DOI: `10.3390/aerospace11120979`.

[9] H. Schlichting and K. Gersten, *Boundary-Layer Theory*, 9th. Berlin, Heidelberg: Springer, 2017.

[10] F. R. Menter, "Two-equation eddy-viscosity turbulence models for engineering applications," *AIAA Journal*, vol. 32, no. 8, pp. 1598–1605, 1994.

[11] J. D. Anderson, *Fundamentals of Aerodynamics*, 5th. New York: McGraw-Hill Education, 2010, Specifically Chapter 4.12.4 on Flow Separation.

[12] J. Aires, S. Withrow-Maser, A. Ruan, C. Malpica, and S. Schuet, "Analysis of handling qualities and power consumption for urban air mobility (UAM) eVTOL quadrotors with degraded heave disturbance rejection and control response," in *Proceedings of the Vertical Flight Society's 78th Annual Forum & Technology Display*, Fort Worth, TX, USA, May 2022.

[13] Z. Ma, E. J. Smeur, and G. C. de Croon, “Design and control of a tilt-rotor tail-sitter aircraft with pivoting vtol capability,” *IEEE Robotics and Automation Letters*, vol. 10, no. 6, pp. 5911–5918, 2025. DOI: `10.1109/LRA.2025.3563821`.

[14] R. C. Nelson, *Flight Stability and Automatic Control*, 2nd. McGraw-Hill, 1998.

[15] J. D. Anderson, *Fundamentals of Aerodynamics*, 5th. McGraw-Hill, 2010.

[16] D. P. Raymer, *Aircraft Design: A Conceptual Approach* (AIAA Education Series), 5th. Reston, VA: American Institute of Aeronautics and Astronautics, 2012, ISBN: 978-1600869112.

A

Semi-Empirical Derivation of Aerodynamic Damping Coefficients

In an attempt to derive more accurate aerodynamic damping coefficients, a set of semi-empirical methods based on strip theory, adapted from Nelson [14] were utilized. The standard empennage volume equations were modified to account for the specific tailless, blended-wing geometry of the UAV. However, these methods assume a forward flight and some of the assumptions used break as the tailsitter approach hover. This created extensive numerical instabilities and therefore, the vastly simplified set of arbitrary coefficients outlined in section 3.2.5 were instead used in all control design work and simulations.

A.1 Geometric Discretization

To calculate the localized area moments and lever arms required for the rotational damping derivatives, the blended-wing platform (aircraft viewed from above) is discretized into five distinct trapezoidal segments: the center rectangle, transition segment, wing root, slipstream (propwash region), and wing tip.

For a given segment with span b, root chord c_{root}, and tip chord c_{tip}, the localized geometric properties are defined analytically. The segment platform area A (trapezoid area) and taper ratio λ are:

$$A = \frac{1}{2}(c_{root} + c_{tip})b, \quad \lambda = \frac{c_{tip}}{c_{root}} \tag{A.1}$$

The local Mean Aerodynamic Chord (MAC), denoted $\bar{c}$, is calculated using the standard geometric integration for a tapered wing [15]:

$$\bar{c} = \frac{2}{3}c_{root}\left(\frac{1 + \lambda + \lambda^2}{1 + \lambda}\right) \tag{A.2}$$

To determine the exact longitudinal moment arms for pitch damping, the aerodynamic center (x_{AC}) of each segment must be localized. According to subsonic thin-airfoil theory [15], the theoretical aerodynamic center lies approximately at the quarter-chord point ($0.25c$). Applying this to the local MAC and accounting for the leading-edge sweep, the x-coordinate of the local aerodynamic centre relative to the nose is:

$$x_{AC} = x_{LE} + 0.25\bar{c} \tag{A.3}$$

where x_{LE} is the longitudinal coordinate of the segment's leading edge evaluated at the spanwise location of the local MAC.

A.2 Roll Damping (C_{l_p})

Following the continuous strip theory formulation in [14], the global non-dimensional roll damping derivative for a symmetric wing is:

$$C_{l_p} = -\frac{4C_{L_\alpha}}{S_{ref}b^2}\int_0^{b/2} y^2 c(y)dy \tag{A.4}$$

where S_{ref} is the reference area, b is the wingspan, and C_{L_α} is the global lift-curve slope. Since $dA = c(y)dy$, the continuous integral evaluates the planform's second moment of area. For the discontinuous blended-wing geometry, this is approximated as a discrete Riemann sum over n segments:

$$C_{l_p} \approx -\frac{4C_{L_\alpha}}{S_{ref}b^2}\sum_{i=1}^{n} A_i y_i^2 \tag{A.5}$$

To accurately model hovering and low-speed flight where the global freestream velocity approaches zero ($V_\infty \to 0$), the global non-dimensionalization breaks down due to a mathematical singularity. To resolve this, the damping formulation is dimensionalized and adapted to utilize local flow conditions, the same way as freestream and slipstream segments are locally evaluated for static aerodynamic coefficients. The total dimensional rolling moment M_p induced by roll rate p is explicitly split into freestream (fs) and slipstream ($slip$) contributions:

$$M_p = \sum_{i\in fs}\left(-q_{fs}C_{L_{\alpha,fs}}A_i\frac{py_i^2}{V_{fs}}\right) + \sum_{i\in slip}\left(-q_{slip}C_{L_{\alpha,slip}}A_i\frac{py_i^2}{V_{slip}}\right) \tag{A.6}$$

Simplifies to:

$$M_p = -\frac{1}{2}\rho\left(\frac{p}{V_{fs}}C_{L_{\alpha,fs}}V_{fs}^2\sum_{i\in fs}S_i y_i^2\right) - \frac{1}{2}\rho\left(\frac{p}{V_{slip}}C_{L_{\alpha,slip}}V_{slip}^2\sum_{i\in slip}S_i y_i^2\right) \tag{A.7}$$

Cancelling squared velocities with divisions, it simplifies to:

$$M_p = -\frac{1}{2}\rho p\left(V_{fs}C_{L_{\alpha,fs}}\sum_{i\in fs}S_i y_i^2 + V_{slip}C_{L_{\alpha,slip}}\sum_{i\in slip}S_i y_i^2\right) \tag{A.8}$$

The local lift-curve slopes are extracted dynamically via central difference from the CFD lookup tables. Since the aerodynamic database is indexed in degrees, the derivative is converted to rad^{-1}:

$$C_{L_\alpha} = \frac{C_L(\alpha+\Delta\alpha) - C_L(\alpha-\Delta\alpha)}{2\Delta\alpha}\cdot\frac{180}{\pi} \tag{A.9}$$

where $\Delta\alpha = 1°$. This isolated evaluation ensures slipstream segments maintain continuous roll damping authority regardless of the global airspeed.

A.3 Pitch Damping (C_{m_q})

For a tailless blended-wing geometry, classical tail-volume derivations are inapplicable as the platform lacks a distinct horizontal stabilizer. Instead, pitch damping is generated by the chordwise distribution of lift across the entire swept planform. The restorative moment is evaluated using longitudinal strip theory integration [15]. When the aircraft experiences a pitch rate q, a wing segment located at a longitudinal distance $\Delta x = x_{ac,local} - l_{cg}$ from the center of gravity experiences an induced vertical velocity $V_z = q\Delta x$. This induces a local change in the angle of attack:

$$\Delta\alpha = \frac{q\Delta x}{V} \tag{A.10}$$

The resulting induced lift on that specific segment is:

$$\Delta L = \frac{1}{2}\rho V^2 S_{local} C_{L_\alpha} \left(\frac{q\Delta x}{V}\right) \tag{A.11}$$

This localized lift acts at the moment arm Δx to create a restoring pitching moment. Because the kinematic derivation squares the lever arm (Δx^2), all surfaces contribute to stabilizing aerodynamic damping regardless of whether their local aerodynamic center is positioned cruise or aft of the global center of gravity. The total dimensional pitch damping moment is the summation of these geometric contributions:

$$M_{pitch} = -\frac{1}{2}\rho V C_{L_\alpha} q \sum \left[S_{local}(\Delta x)^2\right] \tag{A.12}$$

where C_{L_α} is extracted dynamically via central difference from the computational fluid dynamics (CFD) lookup tables.

A.4 Yaw Damping (C_{n_r})

Yaw damping is primarily generated by the side-force of the tip-mounted winglets, which act as the geometric substitute for a conventional vertical tail. Following Nelson's directional damping formulation [14]:

$$C_{n_r} = -2\left(\frac{S_v}{S_{ref}}\right)\left(\frac{\Delta x_w}{b}\right)^2 C_{L_{\alpha,w}} \tag{A.13}$$

where S_v is the combined reference area of both winglets, b is the wingspan, $C_{L_{\alpha,w}}$ is the winglet 3D lift-curve slope, and Δx_w is the longitudinal lever arm from the center of gravity (CG) to the winglet aerodynamic center (AC).
From subsonic thin airfoil theory [15], the AC resides at 25% of the local chord ($0.25c$). For the swept parallelogram geometry, the AC is offset further aft by the geometric tilt, defining the moment arm as:

$$x_{ac,w} = x_{tip} + 0.25c_{tip} + \Delta x_{tilt} \tag{A.14}$$

$$\Delta x_w = x_{ac,w} - l_{cg} \tag{A.15}$$

For standard attached flow, the 2D lift-curve slope is theoretically 2π rad^{-1}. However, the winglets possess a very low aspect ratio ($AR_w < 1$). Applying the 2D assumption overestimates the restoring moment because lifting efficiency is severely diminished by dominant tip vortices rather than bound circulation. To provide a realistic analytical bound, the 3D lift-curve slope $C_{L_{\alpha,w}}$ is scaled down using Helmbold's equation, which corrects for low aspect ratio and leading-edge sweep Λ [16]:

$$AR_w = \frac{b_w^2}{S_v/2} \tag{A.16}$$

$$C_{L_{\alpha,w}} = \frac{2\pi AR_w}{2 + \sqrt{\left(\frac{AR_w}{\cos\Lambda}\right)^2 + 4}} \tag{A.17}$$

where b_w is the true vertical span of a single winglet. This formulation improves how the damping derivative approximates the 3D aerodynamic inefficiency of the parallelogram platform. It is however only an empirical estimate and real wind-tunnel testing with oscillation tests would be needed to improve estimates.

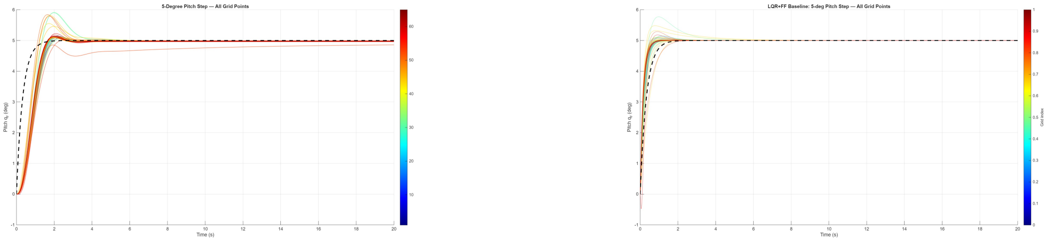
B

Extended simulation results

Numerous additional simulation results were omitted in the results sections, a few of which are shown in this appendix.

B.1 Step Responses

Confirms synthesis, each controller is stable and tracks reference.

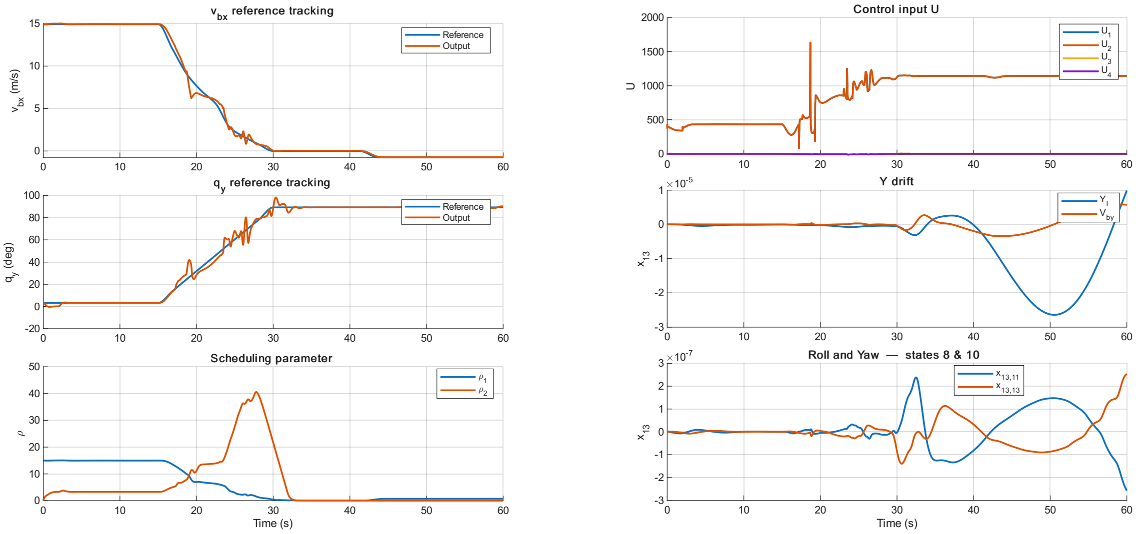


(a) H_∞ **(b)** LQR

Figure B.1: Step responses, 5 degrees in pitch

B.2 H_∞ Transitions - full simulations

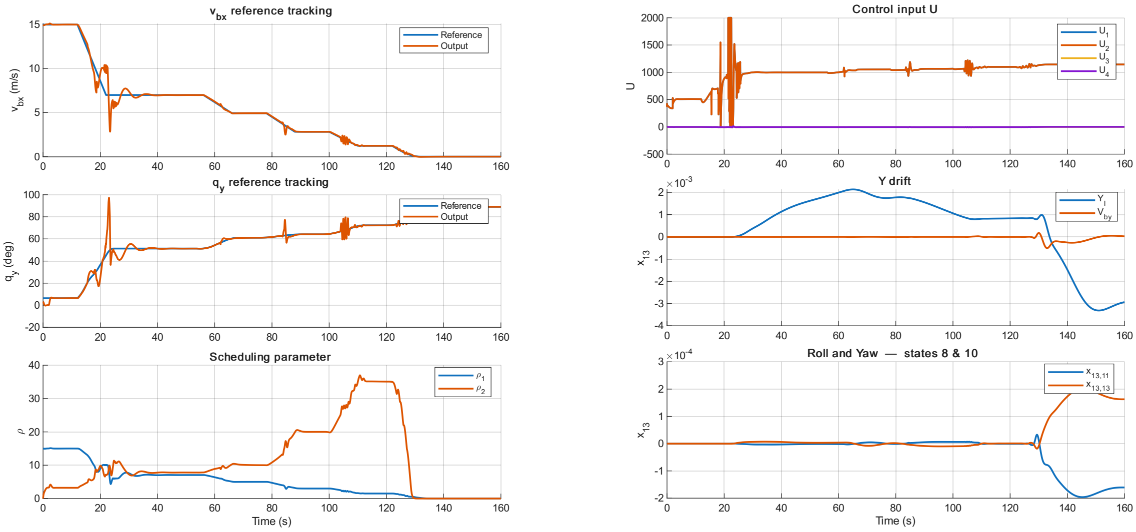
Confirms multiple objectives. The nature and validity of the H_∞ controller, that the LPV captures nonlinearities and that interpolation works well.



(a) Tracking **(b)** Internals

Figure B.2: Aggressive cruise-hover-descent

Confirms stable operating points, confirms controller handles, saturation and drift.



(a) Tracking **(b)** Internals

Figure B.3: Trimpoint to trimpoint transition

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY