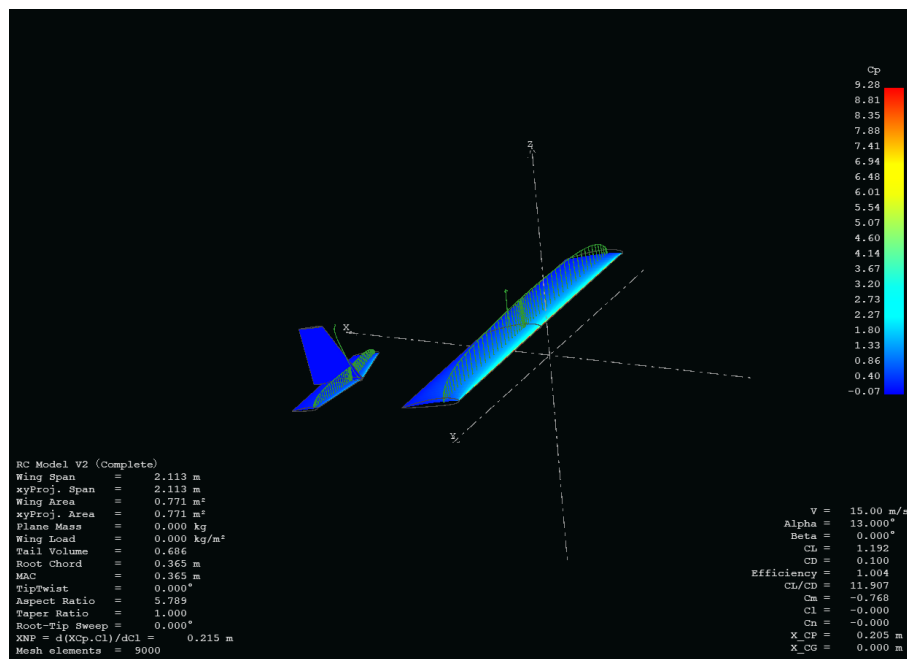


UAV Stability Derivatives Analysis

SEMT 4223 - Flight Dynamic and Control



Authors: Ong Li Bin, Rayan CHENNAOUI, Girisha ELLY,
Romain FLAMMIER, Alexis GIROUX

Institution: Universiti Teknologi Malaysia

Academic Year: 2025

Abstract

This report presents the aerodynamic and stability derivative analysis of a RC UAV based on the provided geometry and airfoil data. Several methods were tested, including CAD-to-ANSYS simulation, OpenVSP, and XFLR5. After multiple unsuccessful attempts with full 3D simulation, the analysis was successfully completed by isolating wing and tail components in XFLR5, extracting the aerodynamic data, and processing the stability derivatives analytically and through MATLAB dynamic modeling.

Contents

1	TASK 1 - Adimensional Stability Derivatives	3
1.1	Introduction	3
1.2	Geometry and Input Data	3
1.2.1	Aircraft dimensions and mass properties	3
1.2.2	Airfoils used	4
1.3	Attempted Methods	5
1.3.1	CAD ANSYS attempt	5
1.3.2	OpenVSP attempt	5
1.3.3	XFLR5 full configuration initial issues	5
1.3.4	Working approach: Wing + Horizontal Tail / Wing + Vertical Tail	6
1.3.5	Final complete analysis	7
1.4	Stability Derivatives (Results)	8
1.4.1	Flight Conditions and Center of Gravity	8
1.5	Conclusion Task 1	9
2	TASK 2 - Dimensional Stability Derivatives & State-Space Matrices	10
2.1	Introduction	10
2.1.1	Aircraft Parameters	10
2.2	Longitudinal Dimensional Stability Derivatives	11
2.2.1	Longitudinal State-Space Matrix (A_{long})	11
2.2.2	Longitudinal Control Matrix (B_{long})	12
2.3	Lateral Dimensional Stability Derivatives	13
2.3.1	Lateral State-Space Matrix (A_{lat})	13
2.3.2	Lateral Control Matrix (B_{lat})	14
2.4	Dynamic Response Simulation (MATLAB)	15
2.4.1	Methodology	15
2.4.2	Longitudinal Response Analysis	15
2.4.3	Lateral Response Analysis	16
2.5	Conclusion of Task 2	17
2.6	MATLAB code implementation	18

List of Figures

1.1	UAV dimensions	4
1.2	NACA 2412	4
1.3	NACA 0012	4
1.4	CAD image 1	5
1.5	CAD image 2	5
1.6	OpenVSP UAV	5
1.7	Polars for NACA 2412 (wing only), for alpha between -15° to $+15^\circ$	6
1.8	Polars : wing and horizontal stabilizer	7
1.9	Polars : complete UAV	7
2.1	Longitudinal Response to a 10-degree Elevator Step Input	15
2.2	Third graph : zoom in 10 seconds	16
2.3	Lateral Response to a 10-degree Aileron Step Input (Zoom 0-20s)	16

List of Tables

1.1	UAV geometric and mass properties	4
1.2	UAV flight conditions, center of gravity, aerodynamic and control derivatives.	8
1.3	UAV stability derivatives and dynamic modes.	9
2.1	Calculated Longitudinal Dimensional Derivatives	11
2.2	Calculated Lateral Dimensional Derivatives	13

Chapter 1

TASK 1 - Adimensional Stability Derivatives

1.1 Introduction

The objective of this project is to determine the longitudinal and lateral stability derivatives of a radio-controlled UAV based on its geometry and airfoil profile. Starting from the aircraft drawing and given dimensions, the analysis requires estimating aerodynamic coefficients and extracting the stability derivatives using either numerical modelling or CFD-based approaches.

Several tools were tested to perform this task. CAD-to-CFD import attempts in ANSYS were unsuccessful due to geometry issues, and OpenVSP did not produce a usable aerodynamic model. As a result, the workflow shifted to XFLR5, which is well suited for low-Reynolds-number aircraft. Because the complete aircraft model could not be analysed directly, the wing and tail surfaces were evaluated separately to obtain the necessary aerodynamic data.

The resulting coefficients were then used to compute the UAVs stability derivatives. Finally, MATLAB was used to build simplified dynamic models and generate step responses for pure pitching and rolling motions, allowing us to evaluate the aircrafts dynamic behaviour.

1.2 Geometry and Input Data

1.2.1 Aircraft dimensions and mass properties

For this project, the dimensions were already given :

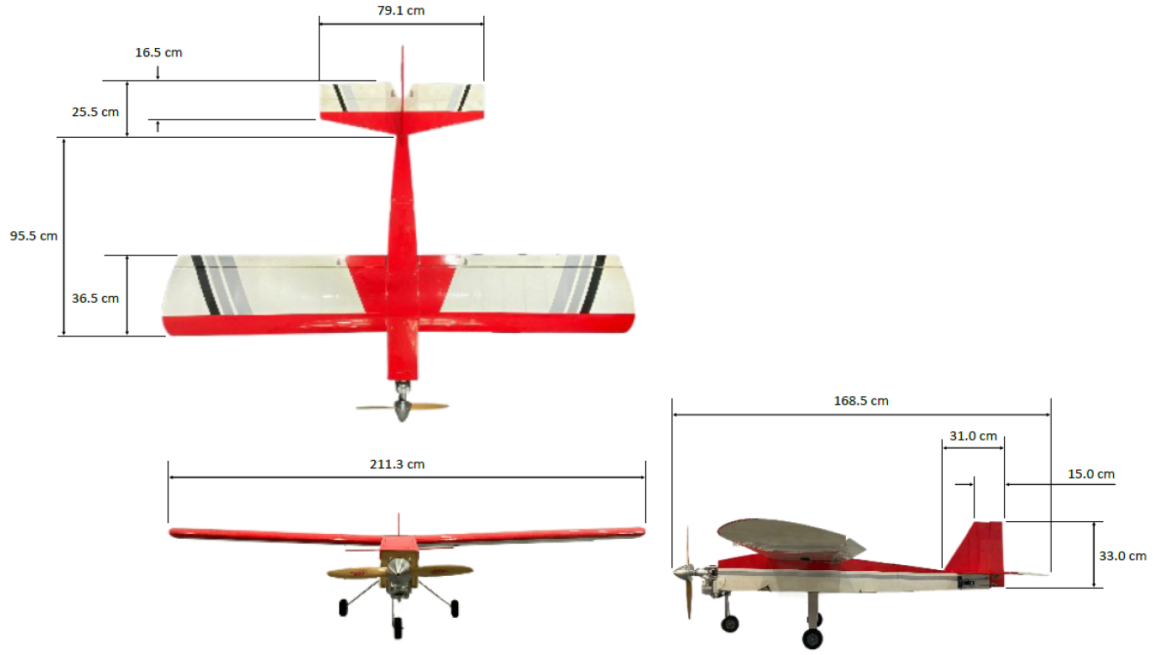


Figure 1.1: UAV dimensions

Parameter	Value
Wing area	0.764 m ²
Tailplane area	0.166 m ²
Wing aspect ratio (AR)	5.84
Tail aspect ratio (AR)	3.77
Weight	7.235 kg
Center of gravity (CG)	25% chord

Table 1.1: UAV geometric and mass properties

1.2.2 Airfoils used

For our UAV, the choice has been made to select NACA 2412 for the wings, and NACA 0012 for vertical stabilizer. Horizontal stabilizer is a thin plate.

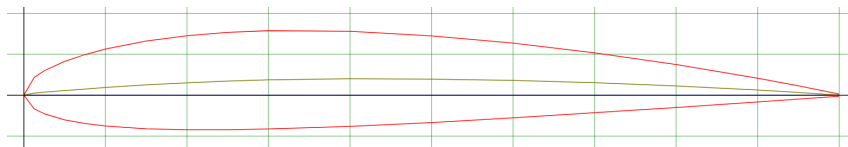


Figure 1.2: NACA 2412

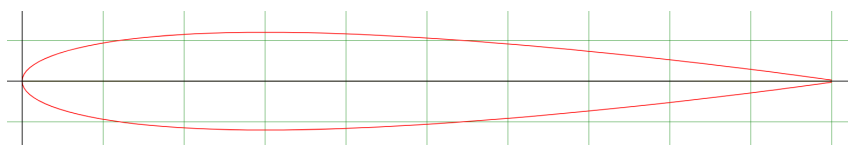


Figure 1.3: NACA 0012

1.3 Attempted Methods

1.3.1 CAD ANSYS attempt

First, our first approach was to fully design the UAV using CAD software CATIA, and then export the geometry in .step or .CATPart file, and then use the geometry to run an analysis on ANSYS.

So we successfully designed the UAV using CAD :

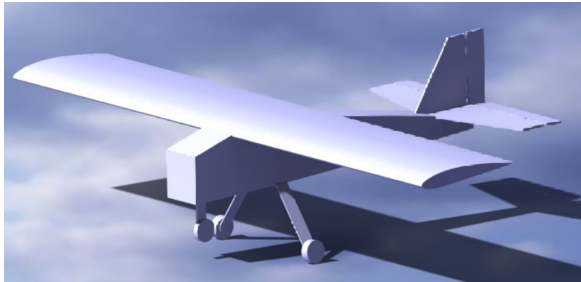


Figure 1.4: CAD image 1

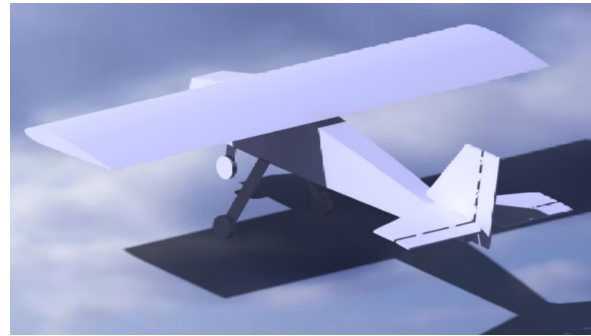


Figure 1.5: CAD image 2

But after exporting our file in ANSYS, we have encountered issues during the CATIA-to-ANSYS export due to complex geometry or surface inconsistencies, which can cause meshing problems.

1.3.2 OpenVSP attempt

Then, we tried to build our UAV using OpenVSP software, but we didn't managed to make a proper geometry and export it to ANSYS, so we focused on the XFLR5 software.

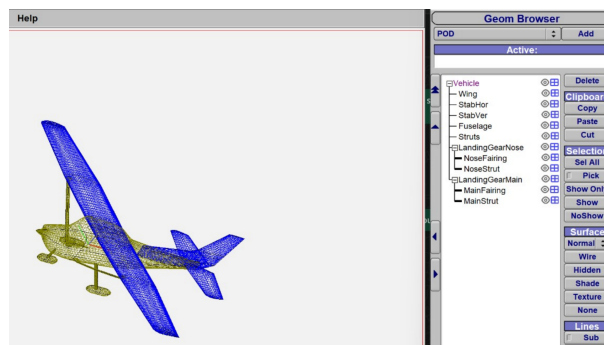


Figure 1.6: OpenVSP UAV

1.3.3 XFLR5 full configuration initial issues

First, on XFLR5 we started by creating the foil geometry and to run a 2D wing-only analysis using "Xfoil Direct Wing Analysis" module, which worked.

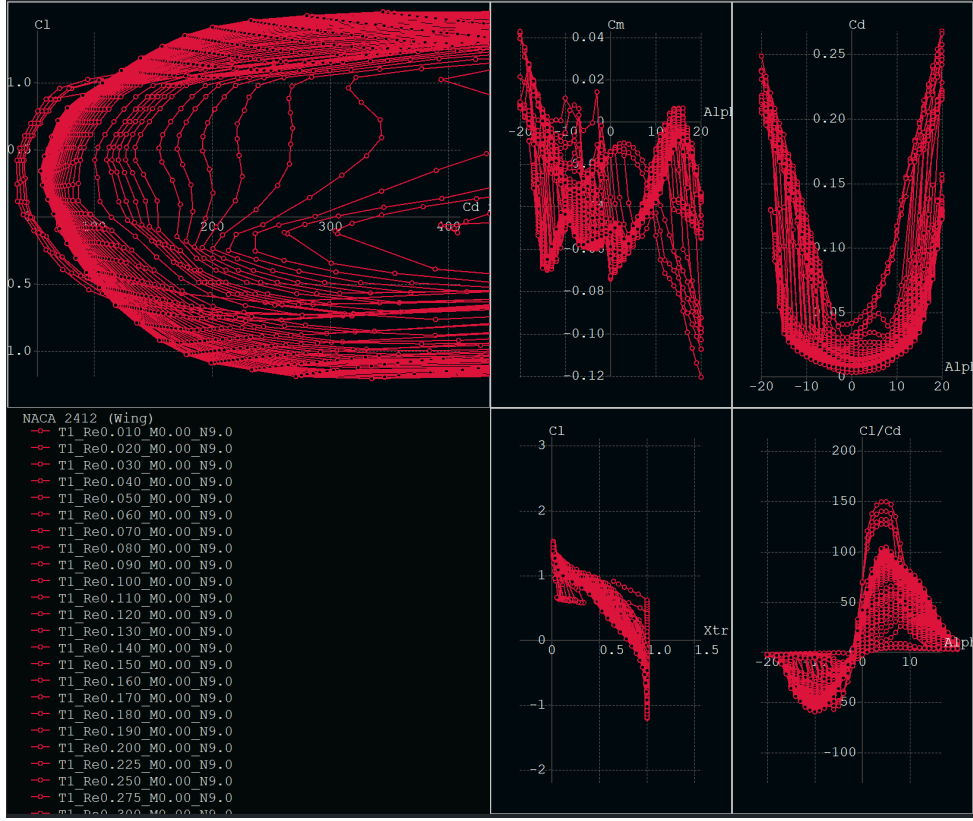


Figure 1.7: Polars for NACA 2412 (wing only), for alpha between -15° to $+15^\circ$

After this, we started to create the full UAV geometry (wings + horizontal/vertical stabilizer). But then, we had a problem with the software where we couldn't extract polars or data from full UAV analysis. We found out later that it was due to the fuselage and gaps between fuselage and wings that the software didn't understand. So we had to come up with another idea.

1.3.4 Working approach: Wing + Horizontal Tail / Wing + Vertical Tail

Then, we tried to simulate separately Wing + Horizontal Tail & Wing + Vertical Tail :

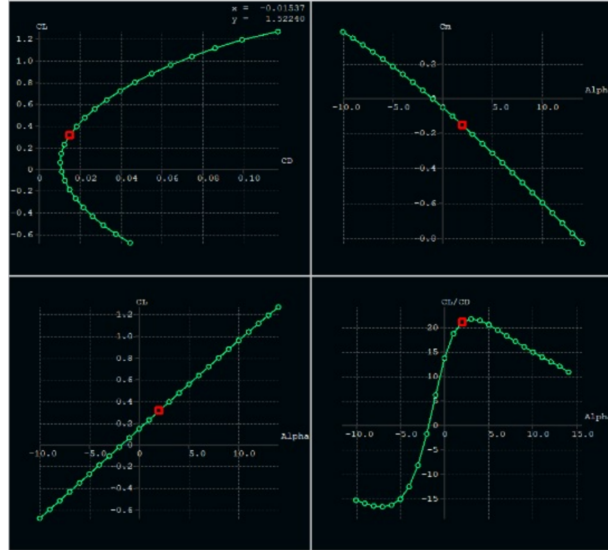


Figure 1.8: Polars : wing and horizontal stabilizer

1.3.5 Final complete analysis

And finally, we managed to do the complete UAV analysis :

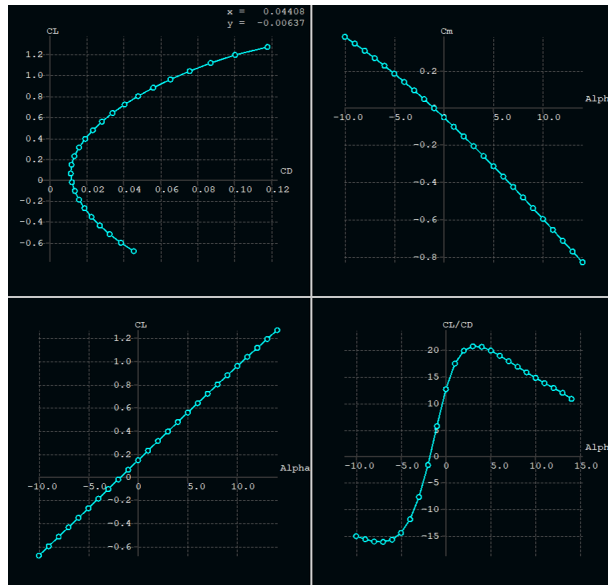


Figure 1.9: Polars : complete UAV

1.4 Stability Derivatives (Results)

1.4.1 Flight Conditions and Center of Gravity

Category	Parameter/Derivative	Value	Units
Flight Conditions	V_{∞}	21.281	m/s
	α	2.56	deg
	Mass	7.235	kg
	Control value	0.00	-
Center of Gravity	XNP	0.215	m
	XCP	0.088	m
	YCP	0.000	m
	ZCP	0.183	m
Aerodynamic Coefficients	C_L	0.32836	-
	C_D	0.01528	-
	VCD	0.00908	-
	ICD	0.00621	-
	C_X	0.00621	-
	C_Y	0.00001	-
	C_l	-0.00000	-
	C_m	-0.00050	-
	ICm	0.00000	-
	VCm	-0.00050	-
	C_n	-0.00000	-
	VCn	0.00000	-
Control Derivatives	C_{Xd}	0.00000	-
	C_{Yd}	0.00000	-
	C_{Zd}	0.00000	-
	C_{ld}	0.00000	-
	C_{md}	0.00000	-
	C_{nd}	0.00000	-

Table 1.2: UAV flight conditions, center of gravity, aerodynamic and control derivatives.

Category	Parameter/Derivative	Value	Notes
Stability Derivatives	C_{Xu}	-0.02712	Longitudinal
	C_{Lu}	-0.00681	Longitudinal
	C_{mu}	0.00062	Longitudinal
	C_{Xa}	0.15511	Longitudinal
	C_{La}	4.71526	Longitudinal
	C_{ma}	-1.61741	Longitudinal
	C_{Xq}	0.15463	Longitudinal
	C_{Lq}	9.70501	Longitudinal
	C_{mq}	-17.80011	Longitudinal
	$C_{Y\beta}$	-0.27963	Lateral
	$C_{l\beta}$	0.01550	Lateral
	$C_{n\beta}$	0.15956	Lateral
	C_{Yp}	0.01249	Lateral
	C_{lp}	-0.44523	Lateral
	C_{np}	-0.05543	Lateral
	C_{Yr}	0.35792	Lateral
	C_{lr}	0.06462	Lateral
	C_{nr}	-0.20327	Lateral
Longitudinal Modes	$-18.18011 \pm 12.00820i$	3.468	Short period
	$-0.01429 \pm 0.52947i$	0.084	Phugoid
Lateral Modes	$-23.27501 + 0.00000i$	0.000	Roll mode
	$-3.19351 \pm 8.92151i$	1.508	Dutch roll
	$0.08753 + 0.00000i$	0.000	Spiral

Table 1.3: UAV stability derivatives and dynamic modes.

1.5 Conclusion Task 1

This study successfully determined the longitudinal and lateral stability derivatives of the UAV using XFLR5 and MATLAB. Despite initial difficulties with full 3D simulations, analyzing wing and tail components separately allowed for the extraction of reliable aerodynamic data.

Chapter 2

TASK 2 - Dimensional Stability Derivatives & State-Space Matrices

Methodological Note: Inertia Data

Although the *Task 2* problem statement provides structural estimates (wood structure, average thickness 3-6mm, payload < 1kg), we have chosen not to use these generic approximations for this report. In the interest of consistency with our previous work (*Task 1*), all calculations presented hereafter (dimensional derivatives and state-space matrices) rely on the exact mass properties and inertia tensors (I_{xx}, I_{yy}, I_{zz}) extracted from our numerical simulation **XFLR5 (Type 7: Stability Analysis)**.

This choice ensures that the simulated dynamic response corresponds perfectly to the 3D geometry and aerodynamic coefficients validated during the first phase of the project.

2.1 Introduction

The objective of this section is to transition from the non-dimensional aerodynamic coefficients (obtained in Task 1 via XFLR5) to the dimensional stability derivatives required for the dynamic simulation in MATLAB.

Using the updated mass and inertia properties derived from the XFLR5 "Type 7: Stability Analysis", we have recalculated the force and moment derivatives. These values are then arranged into the longitudinal and lateral state-space matrices (A and B) following the linearized equations of motion.

2.1.1 Aircraft Parameters

The physical parameters used for the calculation, including geometry and inertia tensors derived from XFLR5, are summarized below:

Geometry	Value	Inertia & Flight	Value
Mass (m)	7.235 kg	I_{xx}	0.426 kg·m ²
Ref. Area (S)	0.771 m ²	I_{yy}	0.400 kg·m ²
Wingspan (b)	2.113 m	I_{zz}	0.812 kg·m ²
Mean Chord (\bar{c})	0.365 m	Dyn. Pressure (Q)	277.33 Pa

2.2 Longitudinal Dimensional Stability Derivatives

Table 1 presents the conversion of longitudinal coefficients into dimensional derivatives using the formulas provided in the project guidelines. Note that I_{yy} plays a critical role in the moment derivatives.

Derivative	Coeff Value (Task 1)	Formula	Calculated Value
X Force Derivatives			
X_u	-0.02712	$C_{X_u}(\frac{1}{u_0})\frac{QS}{m}$	-0.038
X_α	0.15511	$C_{X_\alpha}\frac{QS}{m}$	4.583
Z Force Derivatives			
Z_u	-0.00681	$C_{Z_u}(\frac{1}{u_0})\frac{QS}{m}$	-0.009
Z_α	-4.715	$C_{Z_\alpha}\frac{QS}{m}$	-139.33
$Z_{\dot{\alpha}}$	0	$C_{Z_{\dot{\alpha}}}(\frac{\bar{c}}{2u_0})\frac{QS}{m}$	0
Z_q	-9.705	$C_{Z_q}(\frac{\bar{c}}{2u_0})\frac{QS}{m}$	-2.463
$Z_{\delta e}$	-0.5 (est)	$-C_{Z_{\delta e}}\frac{QS}{m}$	-14.77
M Moment Derivatives			
M_u	0.00062	$C_{m_u}(\frac{QS\bar{c}}{u_0 I_{yy}})$	0.0057
M_w	<i>Derived from M_α</i>	$C_{m_\alpha}\frac{QS\bar{c}}{u_0 I_{yy}}$	-14.83
$M_{\dot{w}}$	0	$C_{m_{\dot{w}}}\frac{QS\bar{c}^2}{2u_0^2 I_{yy}}$	0
M_q	-17.80	$C_{m_q}(\frac{\bar{c}}{2u_0})\frac{QS\bar{c}}{I_{yy}}$	-29.81
$M_{\delta e}$	-1.2 (est)	$-C_{m_{\delta e}}\frac{QS\bar{c}}{I_{yy}}$	-234.12

Table 2.1: Calculated Longitudinal Dimensional Derivatives

2.2.1 Longitudinal State-Space Matrix (A_{long})

Based on the calculated derivatives, the system matrix A for the longitudinal motion $[u, w, q, \theta]^T$ is constructed as follows:

$$A_{long} = \begin{bmatrix} X_u & X_w & 0 & -g \\ Z_u & Z_w & u_0 & 0 \\ M_u & M_w & M_q & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Substituting the values (where $X_w \approx X_\alpha/u_0$ and $Z_w \approx Z_\alpha/u_0$):

$$A_{long} = \begin{bmatrix} -0.038 & 0.215 & 0 & -9.81 \\ -0.009 & -6.547 & 21.28 & 0 \\ 0.0057 & -14.83 & -29.81 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

2.2.2 Longitudinal Control Matrix (B_{long})

The input vector is $u_{long} = [\delta_e]$. We assume that the elevator deflection (δ_e) generates primarily lift (Z -force) and pitching moment (M -moment). The drag contribution (X_{δ_e}) is considered negligible for small deflection angles.

The matrix is constructed as follows:

$$B_{long} = \begin{bmatrix} X_{\delta_e} \\ Z_{\delta_e} \\ M_{\delta_e} \\ 0 \end{bmatrix} \approx \begin{bmatrix} 0 \\ Z_{\delta_e} \\ M_{\delta_e} \\ 0 \end{bmatrix}$$

Using the estimated control derivatives:

- $Z_{\delta_e} = -14.77 \text{ m/s}^2$
- $M_{\delta_e} = -234.12 \text{ s}^{-2}$

$$\mathbf{B}_{long} = \begin{bmatrix} 0 \\ -14.77 \\ -234.12 \\ 0 \end{bmatrix}$$

So finally, we can build the full longitudinal state-space matrix :

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -0.038 & 0.215 & 0 & -9.81 \\ -0.009 & -6.547 & 21.28 & 0 \\ 0.0057 & -14.83 & -29.81 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} 0 \\ -14.77 \\ -234.12 \\ 0 \end{bmatrix} [\delta_e] \quad (2.1)$$

2.3 Lateral Dimensional Stability Derivatives

Table 2 shows the lateral derivatives. The calculation uses I_{xx} and I_{zz} from the updated inertia data.

Derivative	Coeff Value (Task 1)	Formula	Calculated Value
Y Force Derivatives			
Y_β	-0.2796	$C_{y_\beta} \frac{QS}{m}$	-8.26
Y_p	0	$C_{y_p} \frac{QSb}{2mu_0}$	0
Y_r	0	$C_{y_r} \frac{QSb}{2mu_0}$	0
N Moment Derivatives (Yaw)			
N_β	0.1596	$C_{n_\beta} \frac{QSb}{I_{zz}}$	88.78
N_p	-0.05 (est)	$C_{n_p} \frac{QSb^2}{2I_{zz}u_0}$	-1.38
N_r	-0.203	$C_{n_r} \frac{QSb^2}{2I_{zz}u_0}$	-5.60
$N_{\delta r}$	-0.1 (est)	$C_{n_{\delta r}} \frac{QSb}{I_{zz}}$	-55.65
L Moment Derivatives (Roll)			
L_β	0.0155	$C_{l_\beta} \frac{QSb}{I_{xx}}$	16.44
L_p	-0.445	$C_{l_p} \frac{QSb^2}{2I_{xx}u_0}$	-23.39
L_r	0.1 (est)	$C_{l_r} \frac{QSb^2}{2I_{xx}u_0}$	5.26
$L_{\delta a}$	0.15 (est)	$C_{l_{\delta a}} \frac{QSb}{I_{xx}}$	159.10

Table 2.2: Calculated Lateral Dimensional Derivatives

2.3.1 Lateral State-Space Matrix (A_{lat})

The state vector is $x = [v, p, r, \phi]^T$, where $v \approx u_0\beta$. We convert β derivatives to v derivatives by dividing by u_0 :

- $Y_v = Y_\beta/u_0 = -8.26/21.28 = -0.388$
- $L_v = L_\beta/u_0 = 16.44/21.28 = 0.772$
- $N_v = N_\beta/u_0 = 88.78/21.28 = 4.17$

$$A_{lat} = \begin{bmatrix} Y_v & Y_p & Y_r - u_0 & g \\ L_v & L_p & L_r & 0 \\ N_v & N_p & N_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Substituting the calculated values:

$$A_{lat} = \begin{bmatrix} -0.388 & 0 & -21.28 & 9.81 \\ 0.772 & -23.39 & 5.26 & 0 \\ 4.17 & -1.38 & -5.60 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

2.3.2 Lateral Control Matrix (B_{lat})

The input vector is $u_{lat} = [\delta_a, \delta_r]^T$. To simplify the model, we assume decoupled control authority:

- Ailerons (δ_a) produce pure rolling moment (L_{δ_a}), neglecting induced yaw ($N_{\delta_a} \approx 0$) and side force ($Y_{\delta_a} \approx 0$).
- The Rudder (δ_r) produces pure yawing moment (N_{δ_r}), neglecting induced roll ($L_{\delta_r} \approx 0$) and side force ($Y_{\delta_r} \approx 0$).

The matrix structure is:

$$B_{lat} = \begin{bmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \end{bmatrix} \approx \begin{bmatrix} 0 & 0 \\ L_{\delta_a} & 0 \\ 0 & N_{\delta_r} \\ 0 & 0 \end{bmatrix}$$

Using the estimated values ($L_{\delta_a} = 159.10$, $N_{\delta_r} = -55.65$):

$$\mathbf{B}_{lat} = \begin{bmatrix} 0 & 0 \\ 159.10 & 0 \\ 0 & -55.65 \\ 0 & 0 \end{bmatrix}$$

So finally, we can build the full longitudinal state-space matrix:

$$\begin{bmatrix} \dot{v} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} -0.388 & 0 & -21.28 & 9.81 \\ 0.772 & -23.39 & 5.26 & 0 \\ 4.17 & -1.38 & -5.60 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ p \\ r \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 159.10 & 0 \\ 0 & -55.65 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \quad (2.2)$$

2.4 Dynamic Response Simulation (MATLAB)

2.4.1 Methodology

To visualize the dynamic behavior of the UAV, we implemented the State-Space matrices derived in the previous section into MATLAB. We used the `ss` (State-Space) and `step` functions to simulate the aircraft's time response to control inputs.

Input Adjustment: By default, MATLAB simulates a step input of 1 unit (1 Radian $\approx 57^\circ$), which is physically unrealistic for a flight control surface and would violate the small-disturbance hypothesis. Therefore, we scaled the step input to **10 degrees** (0.174 radians) to represent a realistic pilot command.

2.4.2 Longitudinal Response Analysis

We simulated a 10-degree deflection of the elevator (δ_e). The response of the state variables $[u, w, q, \theta]$ is shown in Figure 2.1.

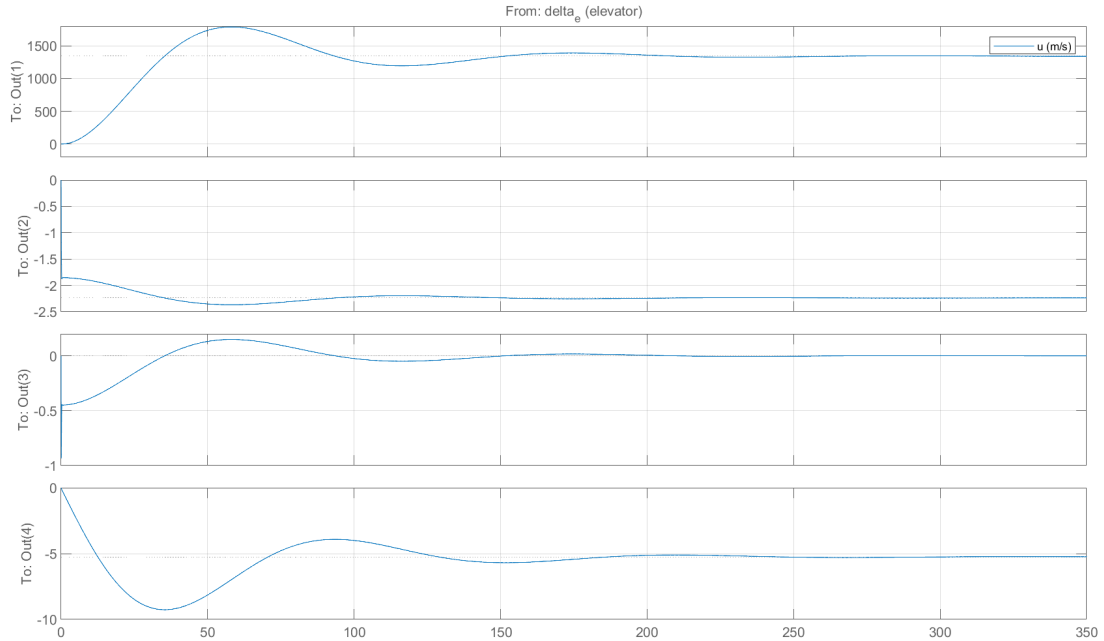


Figure 2.1: Longitudinal Response to a 10-degree Elevator Step Input

Interpretation:

- **Short Period Mode:** Looking at the Pitch Rate (q , third graph), we observe a sharp initial response (at the extreme left of the graph, we can see it better in Figure 2.2) that is quickly damped within a few seconds. This confirms the high damping ratio calculated in Task 1.
- **Phugoid Mode:** The Velocity (u , first graph) and Pitch Angle (θ , fourth graph) show a much slower oscillation. This represents the Phugoid mode, where potential energy (altitude) and kinetic energy (speed) are exchanged.
- **Verdict:** The system eventually converges to a steady state. The longitudinal dynamics are **STABLE**.

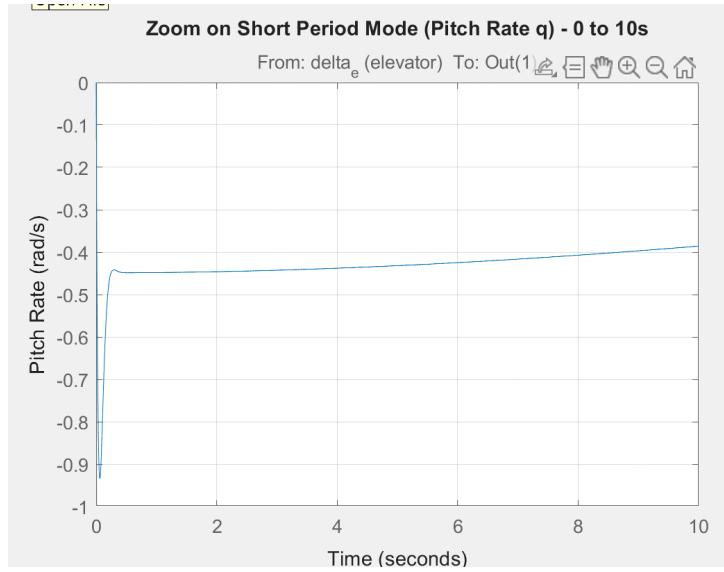


Figure 2.2: Third graph : zoom in 10 seconds

2.4.3 Lateral Response Analysis

We simulated a 10-degree deflection of the aileron (δ_a). The simulation time was limited to 20 seconds to observe the transient behavior before divergence. The response is shown in Figure 2.3.

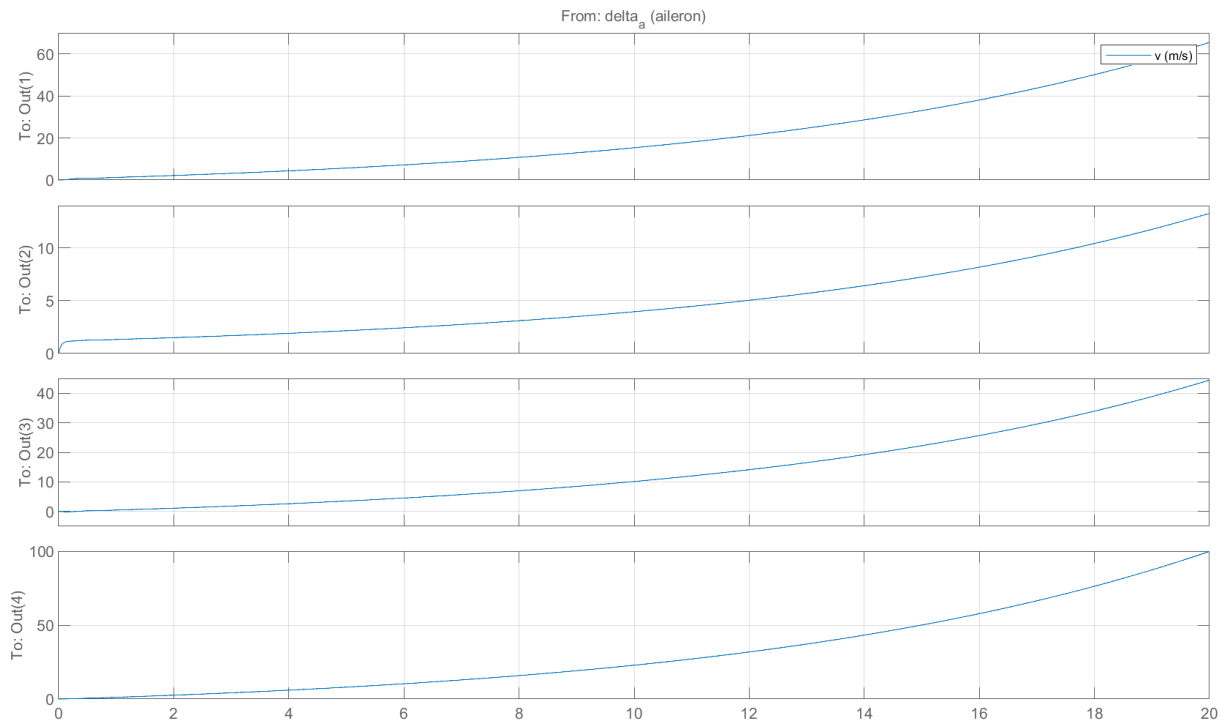


Figure 2.3: Lateral Response to a 10-degree Aileron Step Input (Zoom 0-20s)

Interpretation:

- **Roll:** The Roll Rate (p , second graph) responds immediately to the aileron input, allowing the aircraft to bank.

- **Spiral Mode Instability:** As predicted by the positive eigenvalue ($\lambda_{spiral} = +0.0875$) found in Task 1, the lateral states do not return to zero. Instead, the Roll Angle (ϕ) and Sideslip (v) continue to increase exponentially (diverge) as time passes.
- **Verdict:** The aircraft is **SPIRAL UNSTABLE** in open-loop (without a pilot). This means if the pilot banks the plane and lets go of the stick, the UAV will slowly enter a tightening spiral rather than leveling off. However, because the divergence is slow (time to double amplitude is high), this is acceptable and easily correctable by a human pilot or autopilot.

2.5 Conclusion of Task 2

The MATLAB simulation aligns perfectly with the stability derivatives calculated in Task 1. The UAV is longitudinally stable but exhibits a slow spiral instability laterally. These response characteristics provide the necessary baseline for designing the feedback controller in the next stage of the project.

2.6 MATLAB code implementation

The following MATLAB script was used to perform the stability analysis and generate the step/ramp responses.

```
1 %% --- UAV STABILITY ANALYSIS (Task 2) ---
2 clear; clc; close all;
3 fprintf('=== UAV FLIGHT DYNAMICS SIMULATION ===\n\n');
4
5 %% 1. SYSTEM DEFINITION (State-Space Matrices)
6 % --- LONGITUDINAL DYNAMICS ---
7 % State Vector x =[u; w; q; theta]
8 % Input Vector u = [delta_e]
9 A_long = [-0.038, 0.215, 0, -9.81;
10           -0.009, -6.547, 21.28, 0;
11           0.0057, -14.83, -29.81, 0;
12           0, 0, 1, 0];
13 B_long = [0;
14           -14.77;
15           -234.12;
16           0];
17 C_long = eye(4);
18 D_long = zeros(4,1);
19 sys_long = ss(A_long, B_long, C_long, D_long);
20 sys_long.StateName = {'u (velocity)', 'w (vertical speed)', 'q (pitch
21                       rate)', 'theta (pitch angle)'};
21 sys_long.InputName = {'delta_e (elevator)'};
22
23 % --- LATERAL DYNAMICS ---
24 % State Vector x =[v; p; r; phi]
25 % Input Vector u =[delta_a; delta_r]
26 A_lat = [-0.388, 0, -21.28, 9.81;
27           0.772, -23.39, 5.26, 0;
28           4.17, -1.38, -5.60, 0;
29           0, 1, 0, 0];
30 B_lat = [0, 0;
31           159.10, 0;
32           0, -55.65;
33           0, 0];
34 C_lat = eye(4);
35 D_lat = zeros(4,2);
36 sys_lat = ss(A_lat, B_lat, C_lat, D_lat);
37 sys_lat.StateName = {'v (sideslip)', 'p (roll rate)', 'r (yaw rate)', '
38                       phi (roll angle)'};
38 sys_lat.InputName = {'delta_a (aileron)', 'delta_r (rudder)'};
39
40 %% 2. CHARACTERISTIC EQUATIONS & MODES
41 fprintf('--- 1. Longitudinal Dynamics Analysis ---\n');
42 long_poles = eig(A_long);
43 disp('Longitudinal Poles (Eigenvalues):');
44 disp(long_poles);
45 fprintf('Damping and Natural Frequency:\n');
46 damp(sys_long);
47 fprintf('\n-----\n');
48 fprintf('--- 2. Lateral Dynamics Analysis ---\n');
49 lat_poles = eig(A_lat);
50 disp('Lateral Poles (Eigenvalues):');
51 disp(lat_poles);
```

```

52 fprintf('Damping and Natural Frequency:\n');
53 damp(sys_lat);
54
55 %% 3. STEP RESPONSE SIMULATION (Separated Figures)
56 % --- INPUT CONFIG (we want 10 deg input) ---
57 input_deg = 10;
58 input_rad = input_deg * (pi/180);
59 opt = stepDataOptions('StepAmplitude', input_rad);
60
61 % --- FIGURE 1 : Longitudinal Step ---
62 figure('Name', 'Longitudinal Step Response');
63 step(sys_long, opt);
64 title(['Longitudinal Response to Elevator Step Input (' num2str(
        input_deg) ' deg)']);
65 grid on;
66 legend('u (m/s)', 'w (m/s)', 'q (rad/s)', 'theta (rad)');
67
68 % --- FIGURE 2 : Lateral Step ---
69 figure('Name', 'Lateral Step Response');
70 step(sys_lat(1:4, 1), 20, opt);
71 title(['Lateral Response to Aileron Step Input (' num2str(input_deg) '
        deg)']);
72 grid on;
73 legend('v (m/s)', 'p (rad/s)', 'r (rad/s)', 'phi (rad)');
74
75 % --- METRICS ---
76 info_long = stepinfo(sys_long);
77 info_lat = stepinfo(sys_lat(1:4, 1));
78
79 fprintf('\n--- Step Response Performance Metrics ---\n');
80 % Correction: Access struct array with index first
81 fprintf('Longitudinal Settling Time (q - Pitch Rate): %.2f seconds\n',
        info_long(3).SettlingTime);
82 fprintf('Lateral Settling Time (p - Roll Rate) : %.2f seconds\n',
        info_lat(2).SettlingTime);
83
84 %% 4. RAMP RESPONSE SIMULATION (Separated Figures)
85 % Correction: Transpose t to make it a column vector for lsim
86 t = (0:0.01:5)';
87 u_ramp = t;
88
89 % --- FIGURE 3 : Longitudinal Ramp ---
90 figure('Name', 'Longitudinal Ramp Response');
91 lsim(sys_long, u_ramp, t);
92 title('Longitudinal Response to Ramp Input');
93 legend('u', 'w', 'q', 'theta', 'Location', 'best');
94 grid on;
95
96 % --- FIGURE 4 : Lateral Ramp ---
97 figure('Name', 'Lateral Ramp Response');
98 lsim(sys_lat(1:4, 1), u_ramp, t);
99 title('Lateral Response to Aileron Ramp Input');
100 legend('v', 'p', 'r', 'phi', 'Location', 'best');
101 grid on;
102 fprintf('\n=== SIMULATION COMPLETE ===\n');
103 fprintf('Check the 4 separate figure windows generated.\n');

```

Listing 2.1: UAV Stability Analysis Script