UAS Modelling and Simulation Assignment

Natalia Chacon Rios, Msc. Automated Vehicle Control and Dynamics
student, Cranfield University, id: s428084

May 3, 2024

Keywords— Modelling, trimming, linearization, controls, propulsion

1 Modelling

The first task of the assignment is to build the simulation model of the UAV system using MATLAB/Simulink. The system considered in this assignment will include:

- The equations of motion (6-DOF equation, navigation equation, kinematic equation).
- The model of aerodynamic loads.
- The propulsion model, incl. motor model.
- The environment model (wind model and atmosphere model).
- The actuator models.

1.1 Model Conceptualization

Develop a schematic diagram of the system (conceptual model) and explain the interactions between the sub-systems:

2 The graph presented in figure 1, illustrates the interconnected subsystems of an Unmanned Aerial Vehicle (UAV) simulation model. Each component in this model plays an important role in simulating the dynamic behavior and flight characteristics of the UAV. The subsystems are designed to capture aspects such as the equations of motion, aerodynamic loads, propulsion, environment conditions, actuator responses, and external inputs.

The subsystems include:

- Actuator System: Due to the UAV has no elevator, the ailerons work as elevator by moving the actuator in the same direction. In the actuator system the deviation of the Elevator, Aileron and Throttle are taken as inputs and then after including the model of the second order system for each, and the limits, the output will be the deviation in terms of the natural frequency and damping ratio of the actuator system (controls). Interacts with the aerodynamic coefficients, propulsion model and UAS state boxes by providing its outputs of Elevator, Aileron and throttle Deviations.
- Incidence, Sideslip and Airspeed: Calculates the angles between the velocity vector and the body, and also the total airspeed from the velocity components in the body-fixed coordinate frame. The velocity components interact with the 6DOF equation block by taking the output of the Velocity in the body axes and using it to send the Incidence, sideslip and airspeed results to the Aerodynamic Coefficients calculation box.
- Dynamic Pressure: Computes dynamic pressure with air density and velocity. Interacts with 6DOF equation box by using the Velocity in the body axes and the altitude, then sends the calculated Dynamic Pressure to the Aerodynamic Forces and Moments box.
- Aerodynamic Coefficients: Calculates the Aerodynamic Coefficients and send them to the Aerodynamic Forces and Moments Box. It takes values from 6DOF equation, Actuator System and Incidence, Sideslip and Airspeed boxes.
- Aerodynamic Forces and Moments: Captures the aerodynamic forces and moments acting on the UAV based on its state information and atmospheric conditions. Takes variables from Aerodynamic Coefficients, Dynamic Pressure and 6DOF equation boxes to send the Aerodynamic forces and moment to the Total Force and Moment box.
- Propulsion Model: Models the UAV's propulsion system, taking into account atmospheric conditions, motor characteristics, and control inputs. The controls are taken from the Actuator System box and the interacts with the Total force and moments by sending the Propulsion Forces and moments.
- Total Force and Moments: Uses the Aerodynamic and Propulsion forces and moments boxes result and takes the Coordinate transformation matrix from the "6DOF equation" box to find the gravitational force.
- 6-DOF equation, Body Axes: Takes into account the object's mass properties, the forces acting on it, and the moments applied to it. Through the process of integration, it calculates how these inputs result in changes to the object's velocity, orientation, and position over time, considering both translational and rotational motion. Takes the results from the "Total forces and Moments" box and send results to "Total forces and Moments", "Aerodynamic Forces and moments", "Dynamic Pressure", "Incidence, sideslip and airspeed" and "Aerodynamic Coefficients.

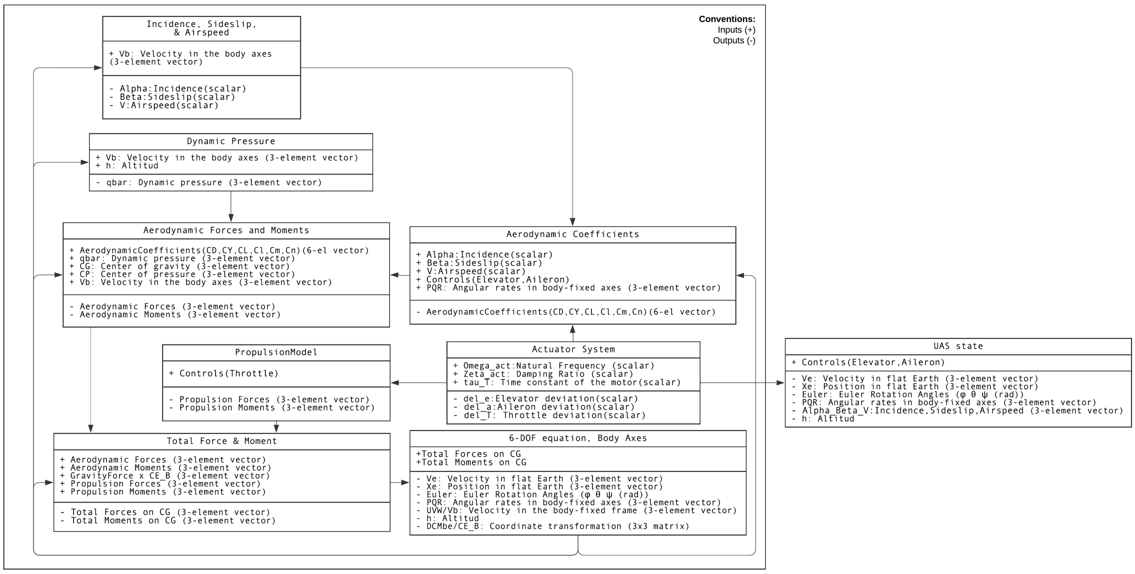

Figure 1: High-level block diagram

2.1 Flight Dynamics

Most aerospace systems use the same equations of motions including 6-DOF equation, the kinematic equation, and the navigation equation. It means that you can utilize the same model structure discussed during the module. Before development the simulation model, please, explain the equation of motion in details .

The Flat-Earth, Body-Axes 6-DOF Equations

FORCE EQUATIONS

These equations describe the translational motion of the UAV in the body-fixed coordinate system. They involve the rates of change of velocity components (U, V, W) with respect to time [1].

$$\dot{U} = RV - QW - g_D \sin\theta + \left(X_A + X_T\right)/\mathrm{m} \tag{1}$$

$$\dot{V} = -RU + PW + g_D \sin\phi\cos\theta + \left(Y_A + Y_T\right)/\mathrm{m} \tag{2}$$

$$\dot{W} = QU - PV + g_D \cos\phi\cos\theta + \left(Z_A + Z_T\right)/\mathrm{m} \tag{3}$$

Where:

- $\dot{U}$: Rate of change of velocity along the body x-axis.
- U: Velocity along the body x-axis.
- V: Velocity along the body y-axis.
- W: Velocity along the body z-axis.
- g_D: Gravitational acceleration.

- P: Roll rate.
- Q: Pitch rate.
- R: Yaw rate.
- g_D: Gravitational acceleration.
- ϕ: Roll angle.
- θ: Pitch angle.

KINEMATIC EQUATIONS

The kinematic equations relate the translational and rotational velocities to the position and orientation of the UAV. They describe how the position and orientation change over time.

The translational kinematic equations link the velocity vector to the position vector, taking into account the current orientation of the UAV.

The rotational kinematic equations relate the angular velocity vector to the orientation (attitude) of the UAV[1].

$$\dot{\phi} = P + \tan\theta(Q\sin\phi + R\cos\phi) \tag{4}$$

$$\dot{\theta} = Q\cos\phi - R\sin\phi \tag{5}$$

$$\dot{\psi} = (Q\sin\phi + R\cos\phi)/\cos\theta \tag{6}$$

MOMENT EQUATIONS

These equations describe the rotational motion of the UAV and involve the rates of change of angular velocity components (P, Q, R) with respect to time [1].

$$\Gamma\dot{P} = J_{xz}\left[J_x - J_y + J_z\right]PQ - \left[J_z\left(J_z - J_y\right) + J_{xz}^2\right]QR + J_z\ell + J_{xz}n \tag{7}$$

$$J_y\dot{Q} = \left(J_z - J_x\right)PR - J_{xz}\left(P^2 - R^2\right) + \mathrm{m} \tag{8}$$

$$\Gamma\dot{R} = \left[\left(J_x - J_y\right)J_x + J_{xz}^2\right]PQ - J_{xz}\left[J_x - J_y + J_z\right]QR + J_{xz}\ell + J_xn \cdot \Gamma = J_xJ_z - J_{xz}^2 \tag{9}$$

Where:

- $\Gamma\dot{P}$: Rate of change of roll rate.
- $J_y\dot{Q}$: Rate of change of pitch rate.
- Γ: Specific force.
- J_{xz}: Moment of inertia about the xz-axis.
- J_x, J_y, J_z: Moments of inertia about the body axes.
- ℓ, m, n: Control moments.
- m: Mass.

NAVIGATION EQUATIONS

The navigation equations are essential for determining the position and orientation of the UAV relative to a reference frame. They integrate sensor measurements and control inputs to update the state estimates.

Commonly, these equations involve integrating sensor data such as accelerometers and gyroscopes to estimate the UAV's position, velocity, and orientation[1].

$$\dot{p}_N = U\mathrm{c}\theta\mathrm{c}\psi + V(-\mathrm{c}\phi\mathrm{s}\psi + \mathrm{s}\phi\mathrm{s}\theta\mathrm{c}\psi) + W(\ \mathrm{s}\phi\mathrm{s}\psi + \mathrm{c}\phi\mathrm{s}\theta\mathrm{c}\psi) \tag{10}$$

$$\dot{p}_E = U\mathrm{c}\theta\mathrm{s}\psi + V(\mathrm{c}\phi\mathrm{c}\psi + \mathrm{s}\phi\mathrm{s}\theta\mathrm{s}\psi) + W(-\mathrm{s}\phi\mathrm{c}\psi + \mathrm{c}\phi\mathrm{s}\theta\mathrm{s}\psi) \tag{11}$$

$$\dot{h} = U\ \mathrm{s}\theta - V\ \mathrm{s}\phi\mathrm{c}\theta - W\mathrm{c}\phi\mathrm{c}\theta \tag{12}$$

Where:

- $\dot{p}_N$: Rate of change of north position.
- $\dot{p}_E$: Rate of change of east position.
- $\dot{h}$: Rate of change of altitude.
- U: Velocity along the body x-axis.

2.2 Modelling

Mathematical models of sub-systems, such as the aerodynamics model, the actuator model, the propulsion model and control for the UAV are provided in the Appendix. Using the equations of motion and provided mathematical models, please, develop the simulation model of the UAV using MATLAB/Simulink and explain it.

1. You may include all parameters needed to build this model in a separated MATLAB file named "UAV_data.m" similar to the MBD Exercises.
2. While building the model, you can utilize "Aerospace Blockset" from the Simulink Library.
3. When you compute arctangent function, please use the function named "atan2" instead of "atan".
4. Solver setting: Go to "simulation" tab -¿ "model configuration parameter" -¿ "Solver" Please set the solver as follows:
    - Type: Fixed-step
    - Solver: ode4(Runge-Kutta),
    - Fixed-step size: Step_Size (this value is already defined in "Sim_Parameters.m")
5. Initialization setting: Go to "File" tab - "Model Properties" - "Callbacks" - "InitFcn*" Please set the following initializations: clc ; clear all ; close all ; UAV_Data ; Sim_Parameters ;

2.2.1 Actuator system

In figure 2 is shown the Actuator model defined by using the equations in 13, 14 and limitations of the Aileron, Elevator and Throttle.

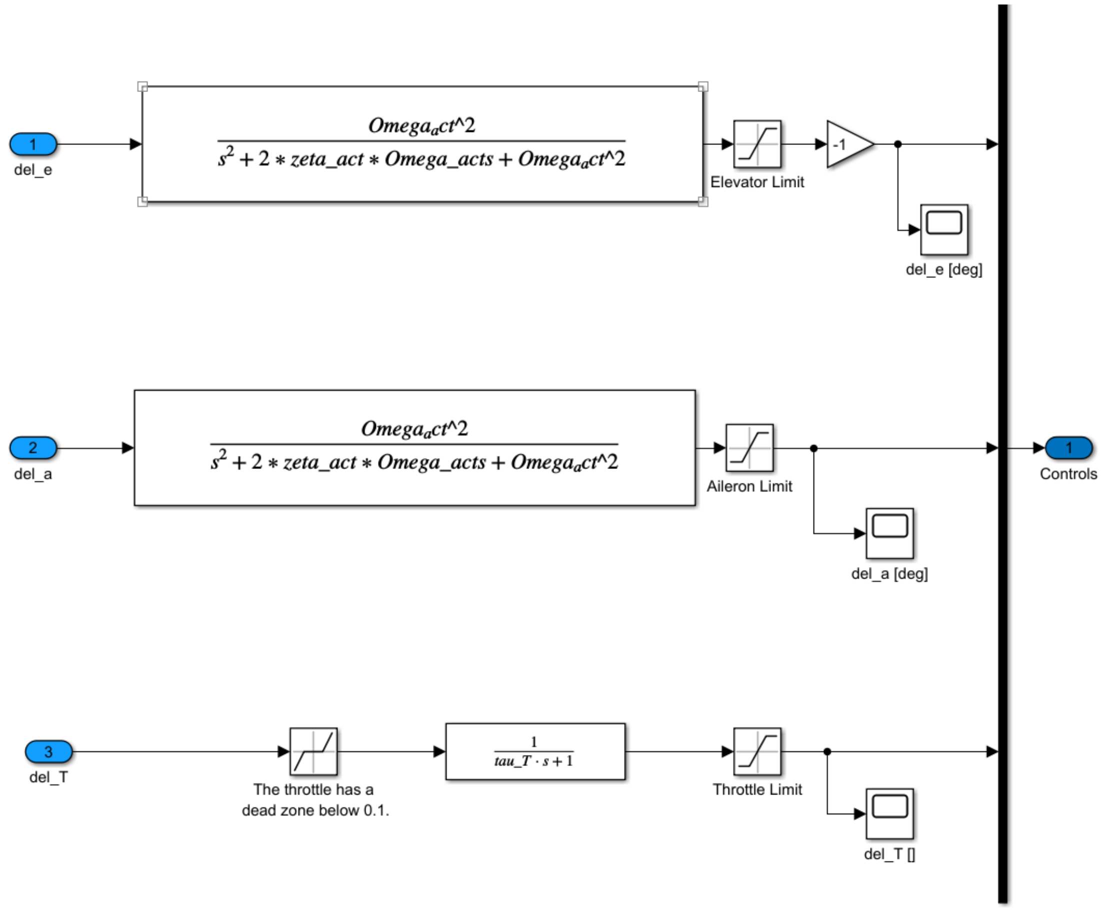

Figure 2: Actuator system

$$\frac{\delta}{\delta c} = \frac{\omega_{act}^2}{s^2 + 2 * \omega_{act} * s + \omega_{act}^2} \tag{13}$$

$$\frac{\delta}{\delta T} = \frac{1}{\tau_T + 1} \tag{14}$$

2.2.2 Altitude - ISA Analysis - Air Density

In Figure 3 is shown the section where the environmental values are calculated according to the altitude. In this case the ISA Atmosphere Model from simulink library is used to get the values of density (ρ).

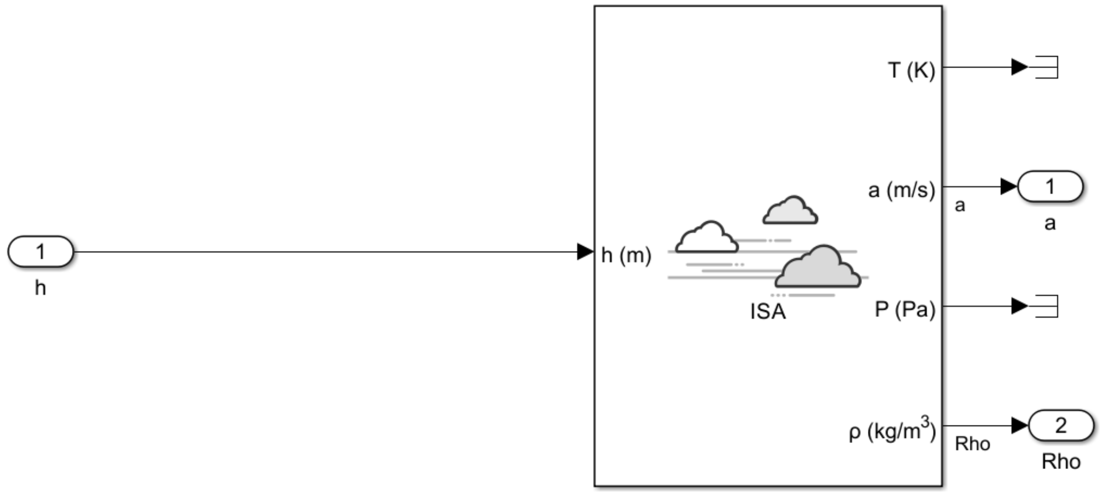

Figure 3: Air Density calculation

2.2.3 Incidence, Sideslip, and Airspeed calculation

The results shown in Figure 4 for Incidence (α), Sideslip(β), and Airspeed(V), where calculated with the equations stated in Equation 15 and 16.

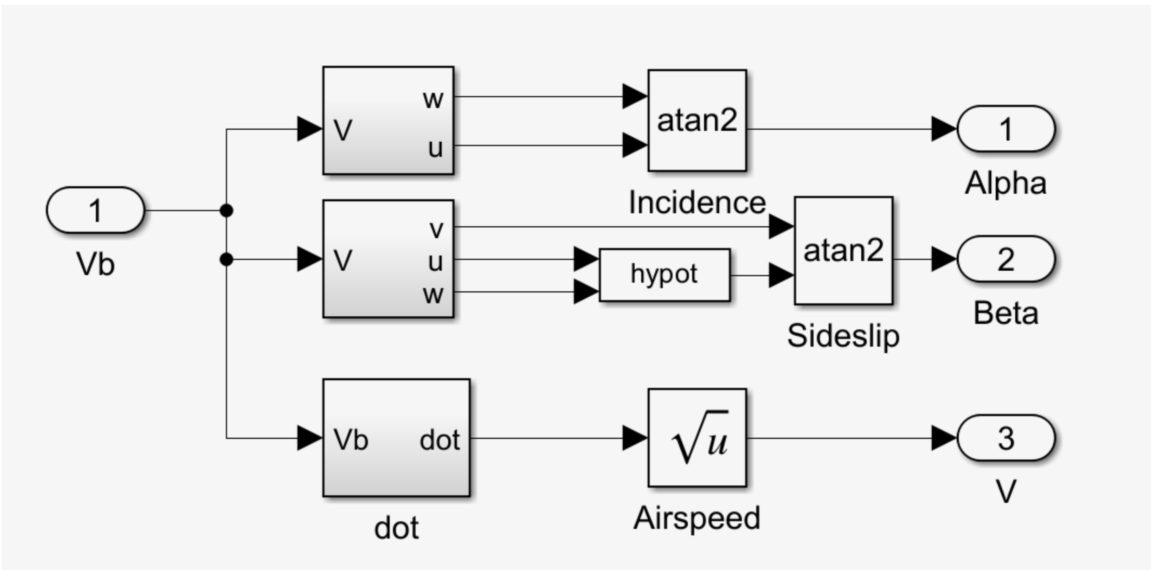

Figure 4: Definition of Incidence, sideslip and airspeed

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right), \quad \beta = \sin^{-1}\left(\frac{v}{V}\right) \tag{15}$$

$$V = \sqrt{u^2 + v^2 + w^2} \tag{16}$$

2.2.4 Dynamic Pressure

I figure 5, the dynamic pressure is calculated using the Air density calculated before.

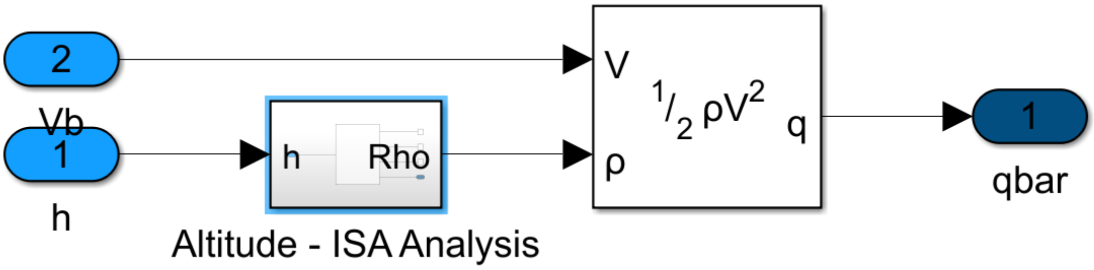

Figure 5: Dynamic Pressure Definition

2.2.5 Aerodynamic Coefficients

This section includes explanations, equations, and Simulink models illustrating aerodynamic coefficients. Simulink models for Aerodynamic force and moment coefficients are presented in Figures 6 to 11, along with corresponding equations from 17 to 22. The approach involved following the equations and constructing the model using tools such as divide, sum, gain, constant, input, output, and multiply [2].

$$C_{\mathrm{D}} = C_{\mathrm{D}_0} + C_{\mathrm{D}_\alpha}\alpha + C_{\mathrm{D}_{\alpha}2}\alpha^2 + C_{\mathrm{D}_\beta}\beta + C_{\mathrm{D}_{\beta^2}}\beta^2 + C_{\mathrm{D}_\mathrm{q}}\frac{\mathrm{c}}{2V}q + C_{\mathrm{D}_{\delta_e}}\delta_e, \tag{17}$$

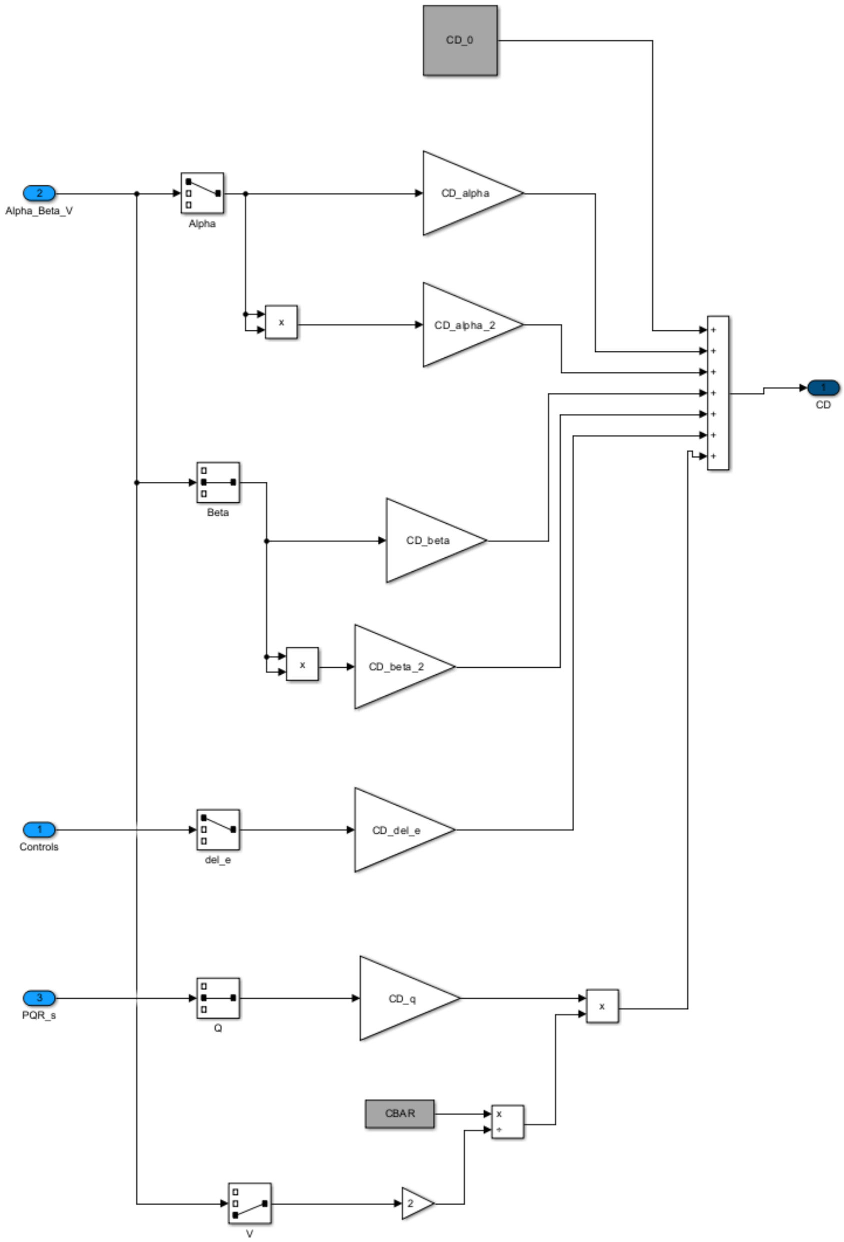

Figure 6: Drag Coefficient

$$C_{\mathrm{L}} = C_{\mathrm{L}_0} + C_{\mathrm{L}_\alpha}\alpha + C_{\mathrm{L}_\mathrm{q}}\frac{\mathrm{c}}{2V}q + C_{\mathrm{L}_{\delta_e}}\delta_e, \tag{18}$$

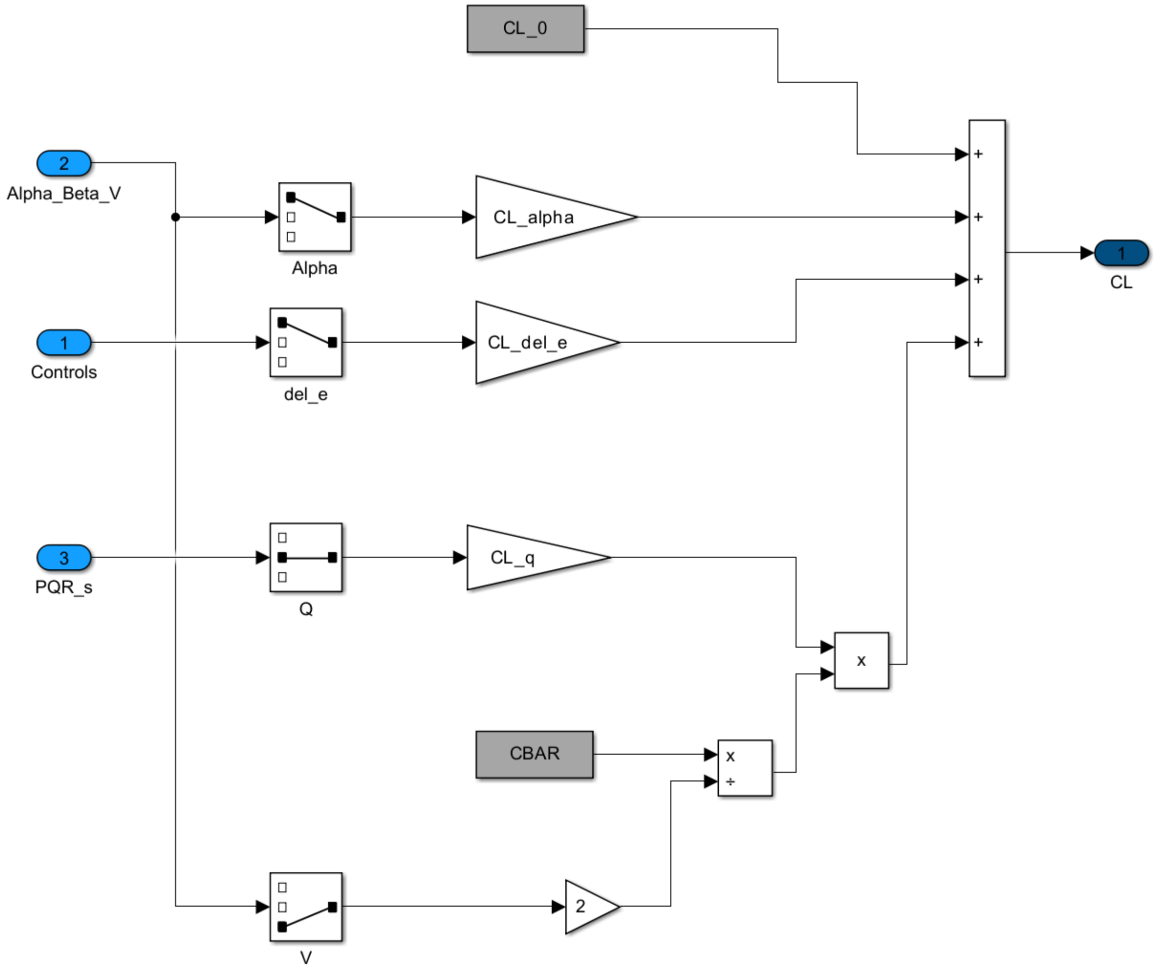

Figure 7: Lift Coefficient

$$C_{\mathrm{Y}} = C_{\mathrm{Y}_0} + C_{\mathrm{Y}_\beta}\beta + C_{\mathrm{Y}_\mathrm{p}}\frac{\mathrm{b}}{2V}p + C_{\mathrm{Y}_\mathrm{r}}\frac{\mathrm{b}}{2V}r + C_{\mathrm{Y}_{\delta_a}}\delta_a, \tag{19}$$

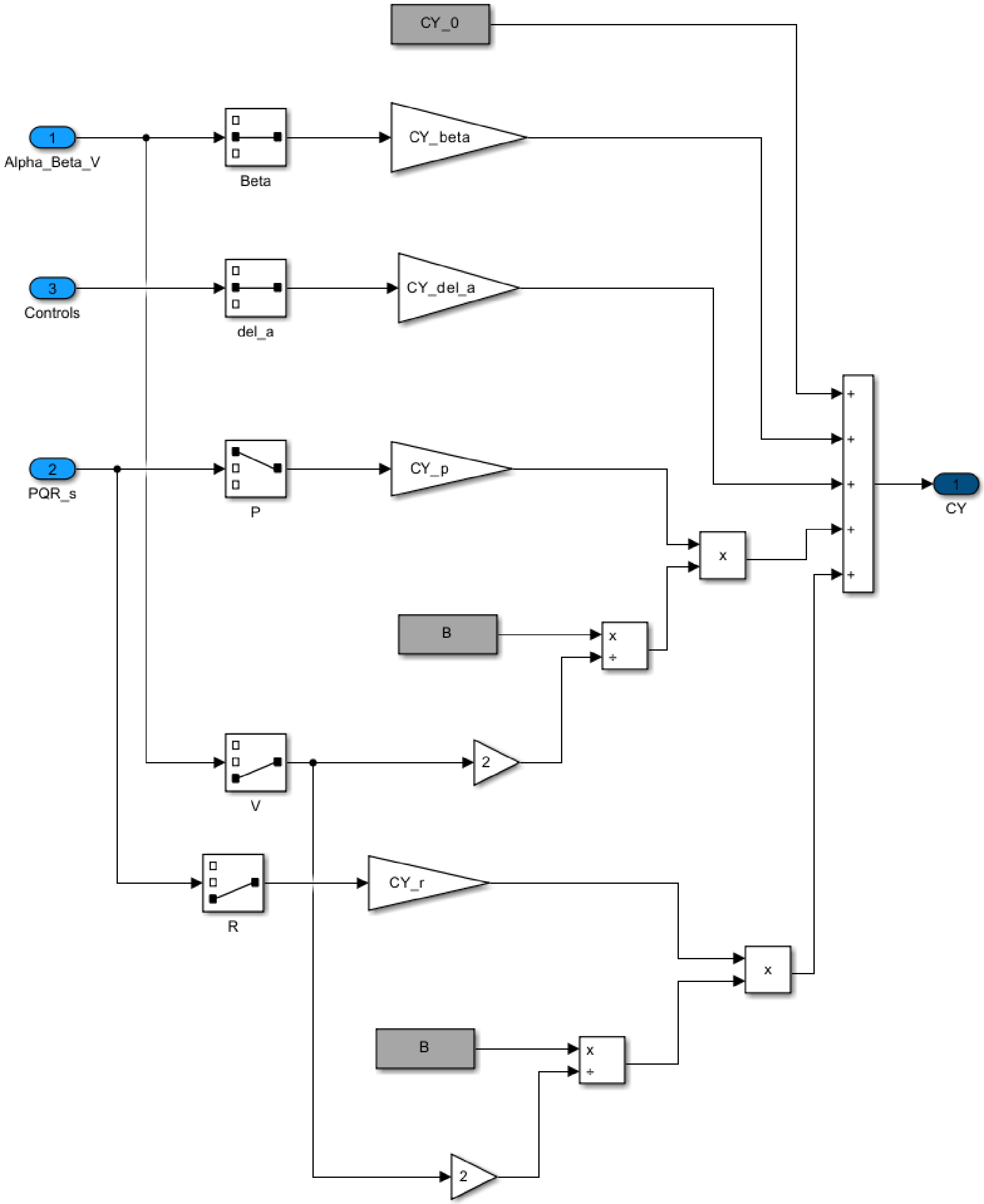

Figure 8: Sides force Coefficient

$$C_{\mathrm{m}} = C_{\mathrm{m}_0} + C_{\mathrm{m}_\alpha}\alpha + C_{\mathrm{m}_\mathrm{q}}\frac{\mathrm{c}}{2V}q + C_{\mathrm{m}_{\delta_e}}\delta_e, \tag{20}$$

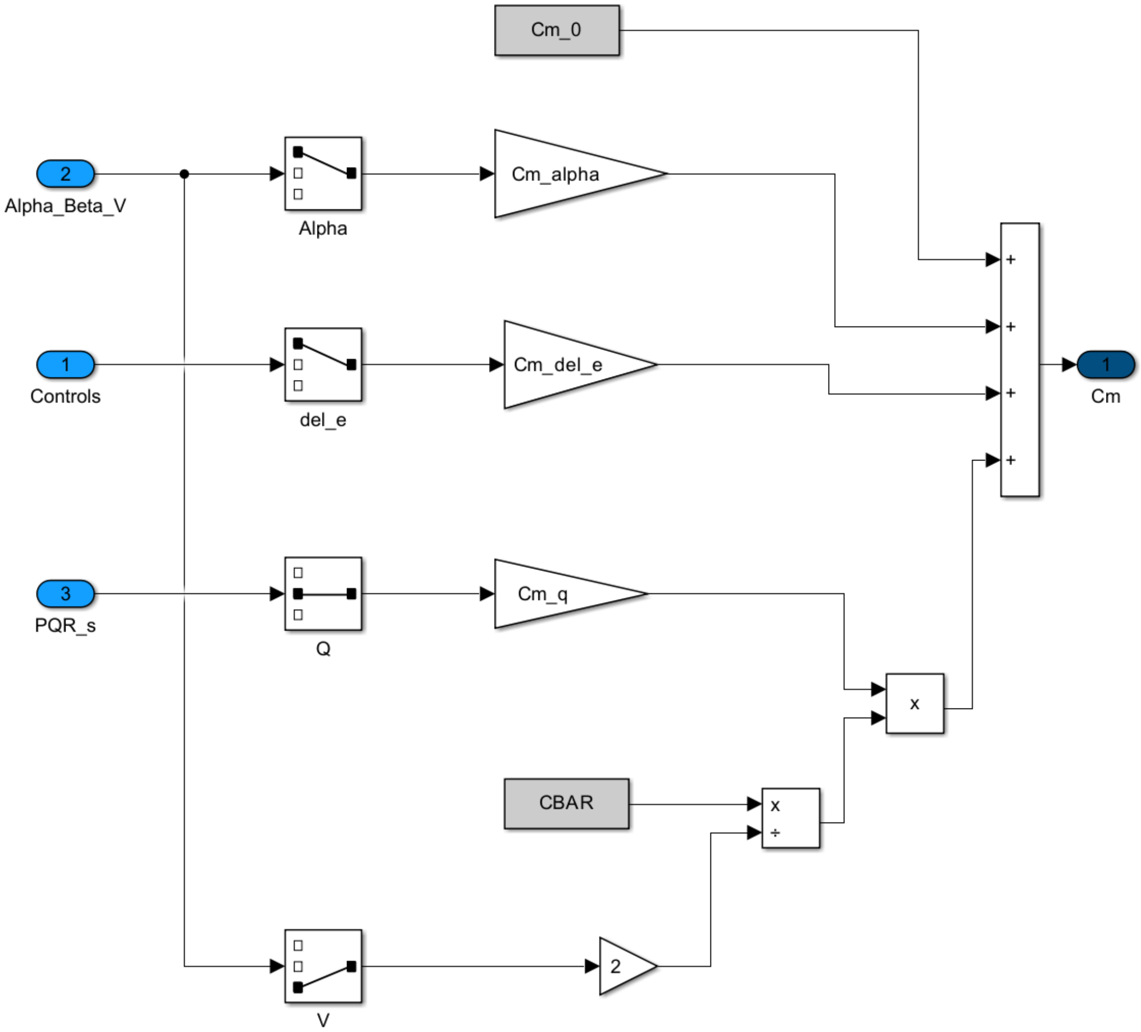

Figure 9: Pitching Moment Coefficient

$$C_{\mathrm{l}} = C_{\mathrm{l}_0} + C_{\mathrm{l}_\beta}\beta + C_{\mathrm{l}_\mathrm{p}}\frac{\mathrm{b}}{2V}p + C_{\mathrm{l}_\mathrm{r}}\frac{\mathrm{b}}{2V}r + C_{\mathrm{l}_{\delta_a}}\delta_a, \tag{21}$$

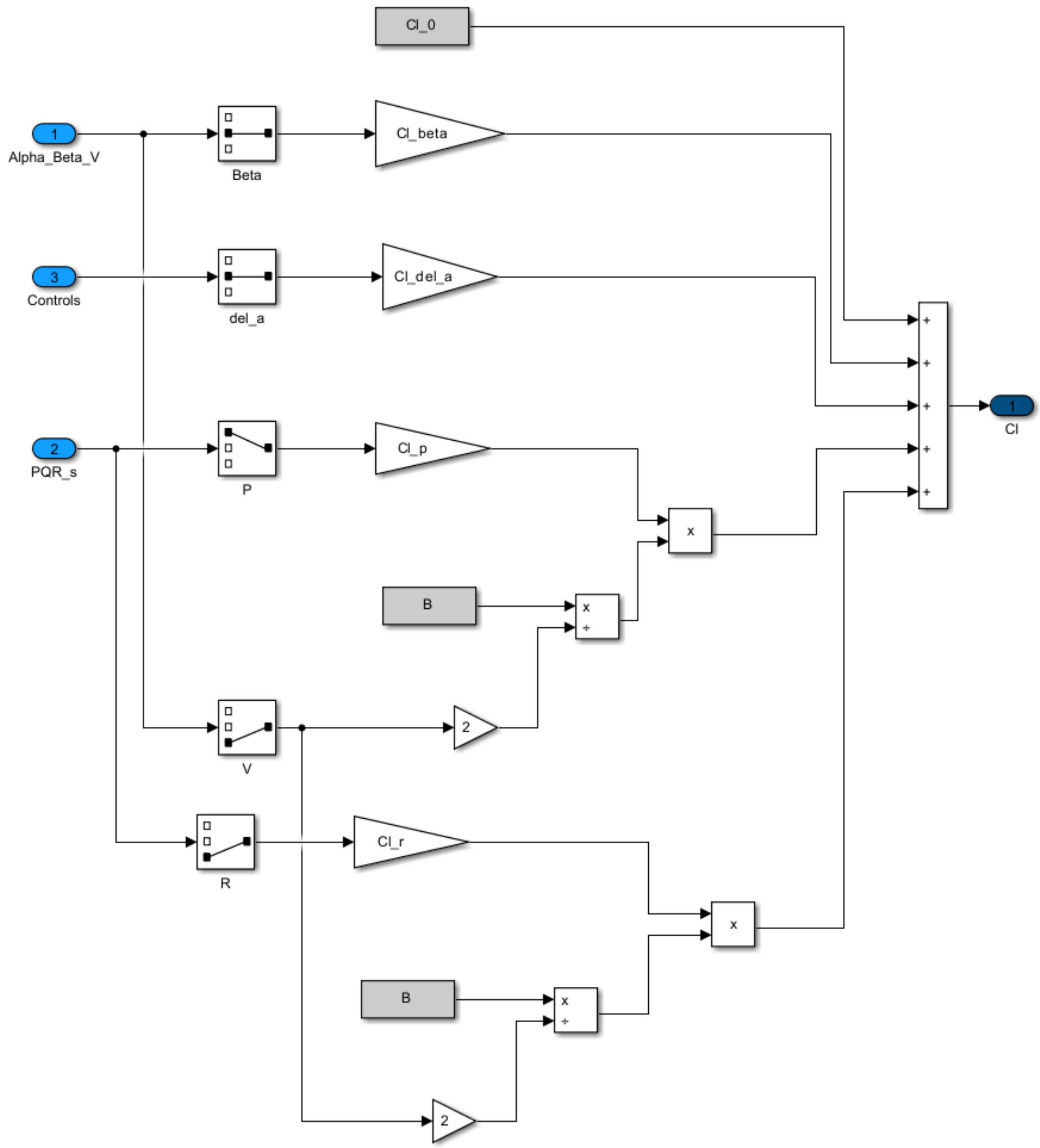

Figure 10: Rolling Moment Coefficient

$$C_{\rm n} = C_{\rm n_0} + C_{{\rm n}_\beta}\beta + C_{\rm n_p}\frac{\rm b}{2V}p + C_{\rm n_r}\frac{\rm b}{2V}r + C_{{\rm n}_{\delta_a}}\delta_a \tag{22}$$

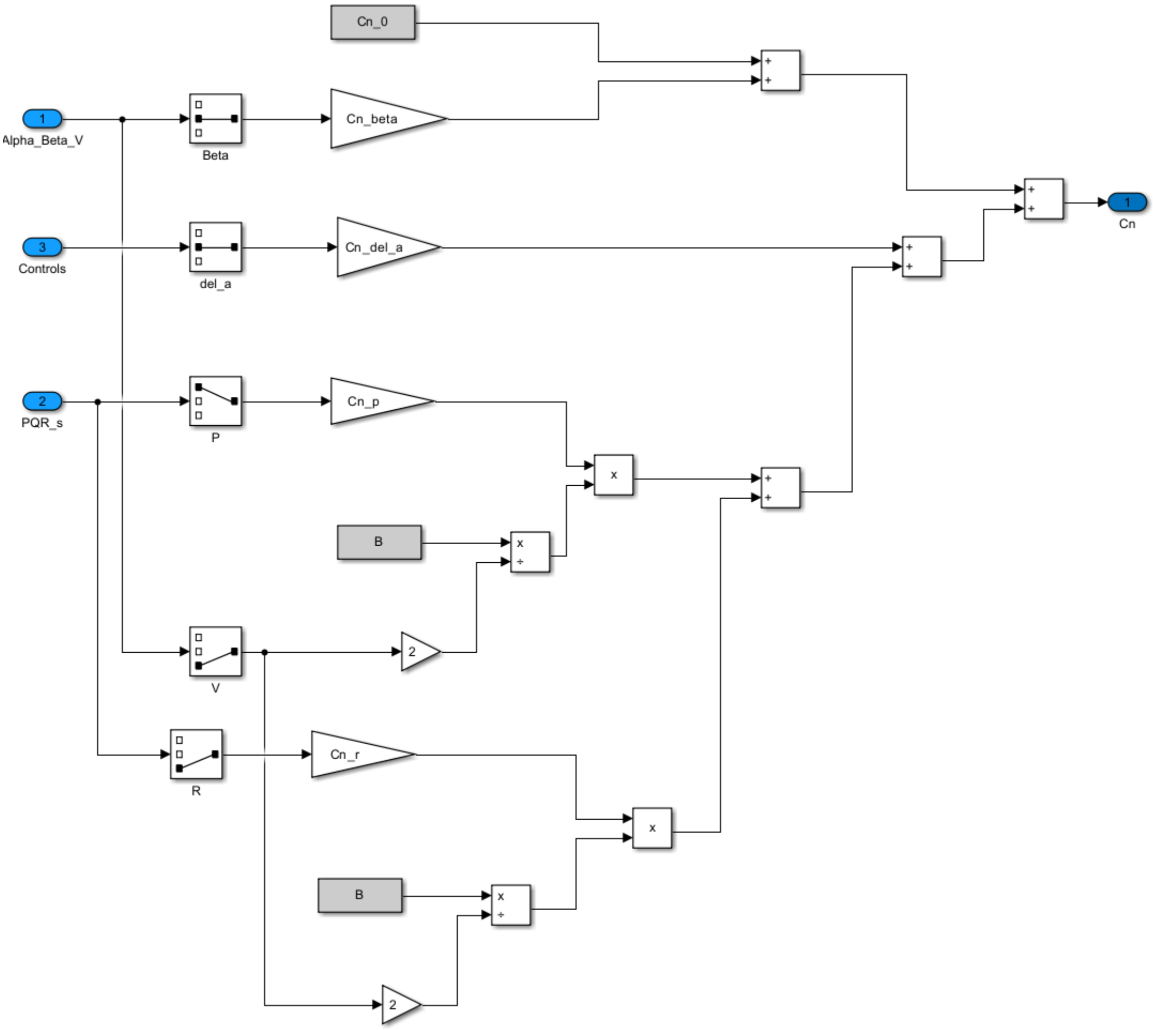

Figure 11: Yawing Moment Coefficient

2.2.6 Gravity Force

In Figure 12 is shown the gravity box in simulink. For a UAV model along the body axes, the choice of the direction cosine matrix depends on the specific convention being followed and how the orientation of the axes is defined. Generally, the direction cosine matrix C_{E_B} is used to transform vectors expressed in the body frame to the Earth frame[2].

Gravity is typically specified in the Earth frame, so it needs to be transformed to the body frame for consistency with other terms in the equations of motion expressed in the body frame.

Therefore, in Simulink, it would be used the direction cosine matrix C_{E_B} to transform the gravity vector from the Earth frame to the body frame.

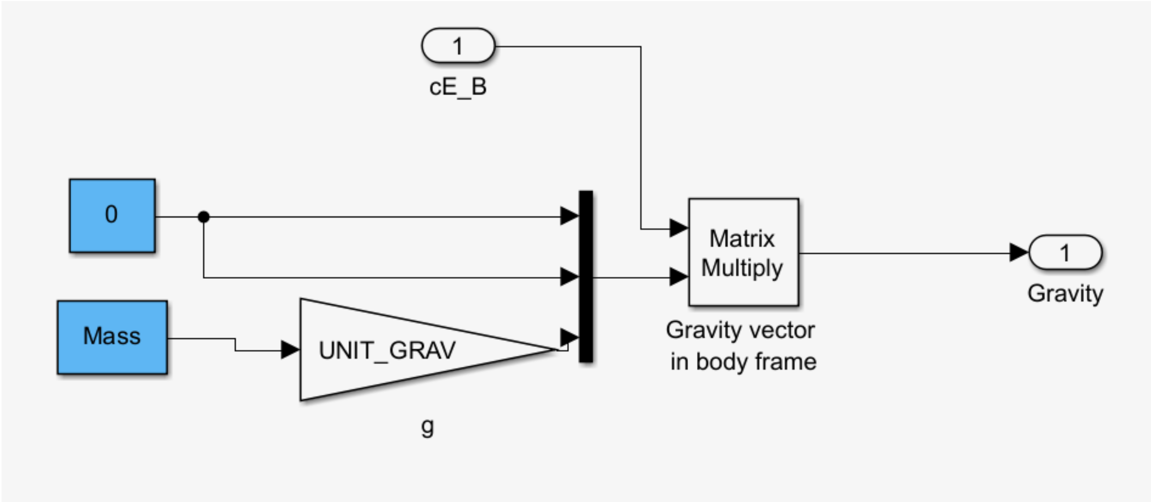

Figure 12: Gravity Box

2.2.7 Modeling Aerodynamic Forces and Moments

The Model presented in Figure 15 represents the Aerodynamic Forces and Moments simulink box and it was created by following the equations 23, 24 and 25.

$$F_D = \mathrm{QS}C_{\mathrm{D}}, F_L = \mathrm{QS}C_{\mathrm{L}}, F_Y = \mathrm{QS}C_Y \tag{23}$$

$$m = \mathrm{QSc}C_m, l = \mathrm{QSb}C_l, n = \mathrm{QSb}C_n \tag{24}$$

$$\mathrm{Q} = (1/2)\rho V^2 \tag{25}$$

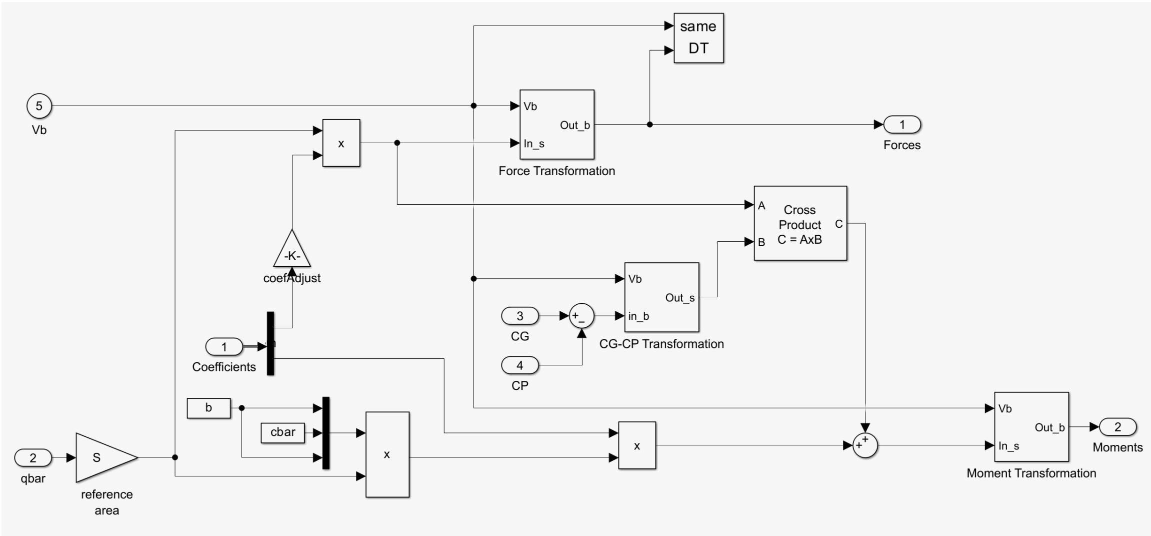

Figure 13: Aerodynamic Forces and Moments Simulink

2.2.8 Propulsive Force and Moment

The propulsive force and moment uses the equation 26 for the Motor model (Presented in Figure 14) and then converted to Revolution per second by dividing for 60. The RPS is the computed with the motor characteristics in the model 15.

$$\text{RPM} = 7000 + 20000\delta_{th} \tag{26}$$

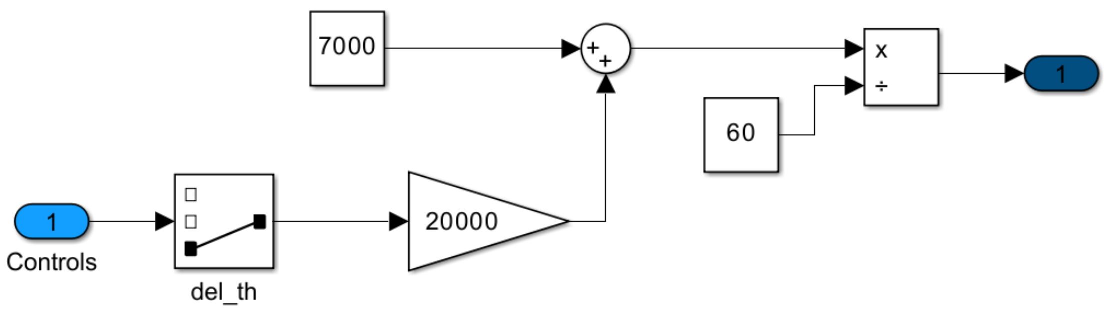

Figure 14: Motor Model

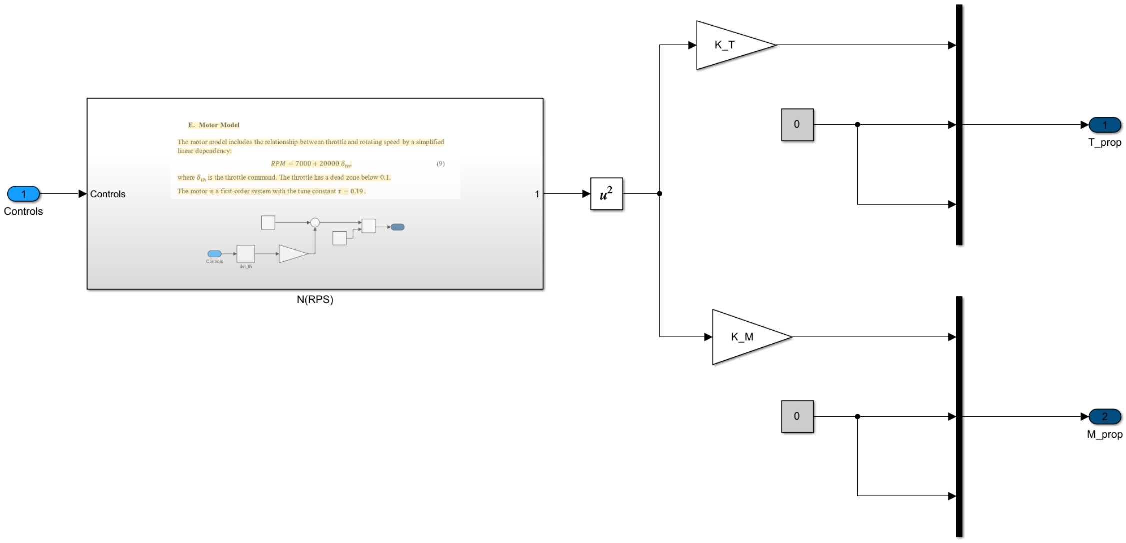

Figure 15: Propulsive Force and Moment

2.2.9 Total Force and Moment

In figure 16 is presented the model for the sum of forces and models of the system.

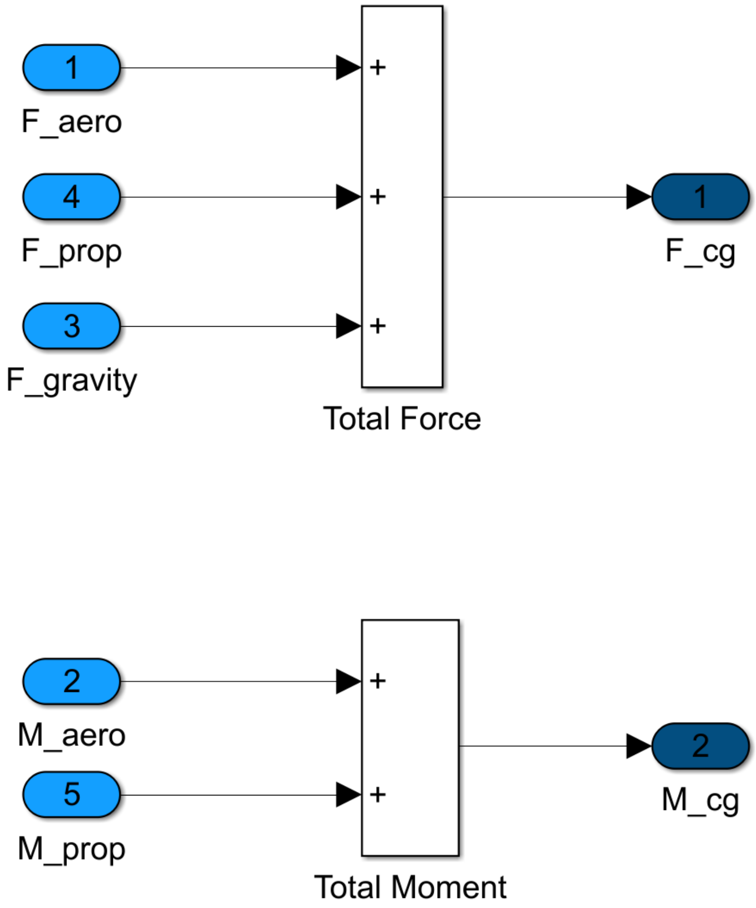

Figure 16: Total Force and Moment

2.3 Equations of motion

In Figure 17 is presented the Equation of motion models, where the results defined are the euler angles(ϕ, θ, ψ), the Velocity in the body frame (UVW), the velocity in flat eart (Ve), Position in flat earth (Xe), thee angular rates in the body-fixed frame (PQR), Altitud (h) and the Coordinate transformation (DCMbe/CE_B)

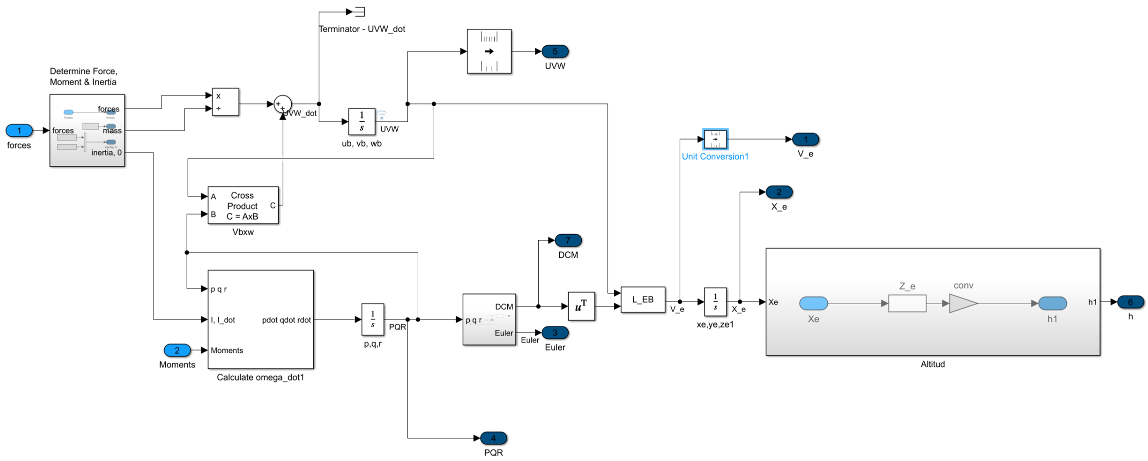

Figure 17: Equation of motion

2.4 Model Verification

After building the model, please verify the simulation model using the model verification techniques, which were introduced within the module.

Verification techniques are the following: tracing, comparing outputs of simulation models and outputs of mathematical models, checking the simulation model using prior knowledge. In particular, a simulation model using prior knowledge should be checked as follows:

1. Check whether control inputs produce the desired motion
2. Check the longitudinal and lateral modes

The model extensively utilized selectors, multiplexers (mux), buses, and well-defined block names for the purpose of clarification. This approach enhanced the comprehensibility of the file, improved visualization, and contributed to a reduction in the error rate.

2.4.1 Verification Techniques - Use of pre-programmed (or verified) blocks

Pre-programmed (verified) blocks were employed for cross-validation of the model. Simulink offers a range of pre-programmed and verified blocks, including the Aerospace Blockset. Both models yielded same results. In Figure 18 is presented the Model Designed for the assignment with the results and in Figure 19 is presented the pre-programmed blocks verification model with results.

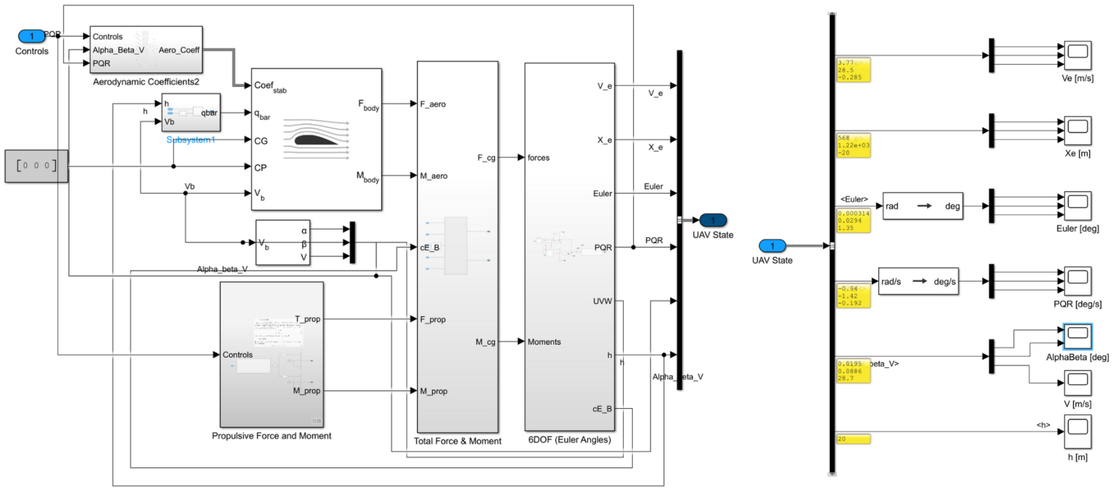

Figure 18: Model Designed

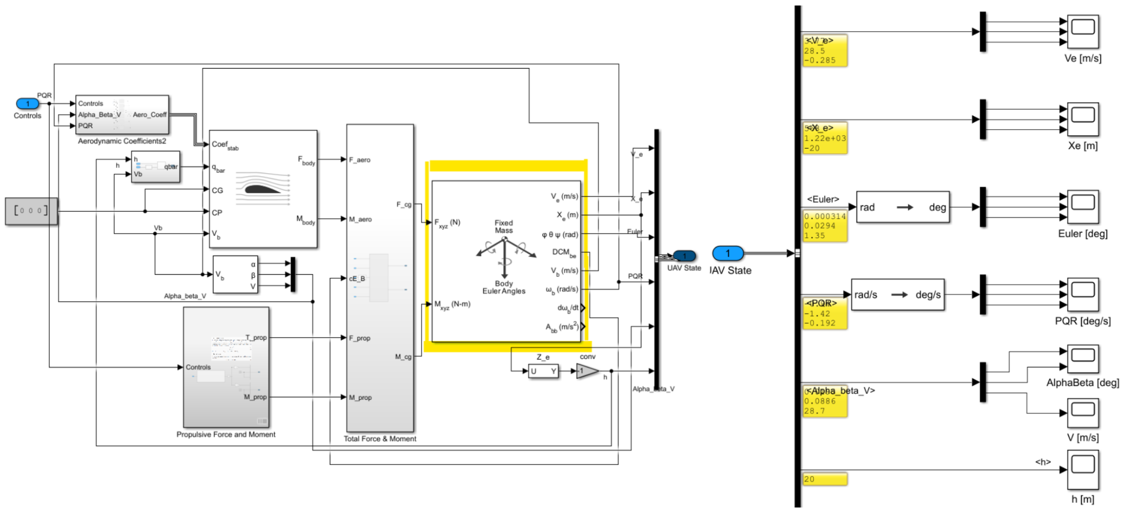

Figure 19: Pre-programmed blocks verification

Note: The model will be attached in the files shared through email, file name is Model_$Verification_Preprogrammed_Blocks.slx$.

2.5 Verification Techniques - Checking Output with Model Equations

In Figure 20 is presented the .m code created for the checking output verification technique for the Aerodynamic Coefficients. The inputs for Controls, PQR Alpha, Beta and V, is the value 3. As it can be seen in Figure 20 the results in the simulink model is the same as the ones obtained in the verification code 21

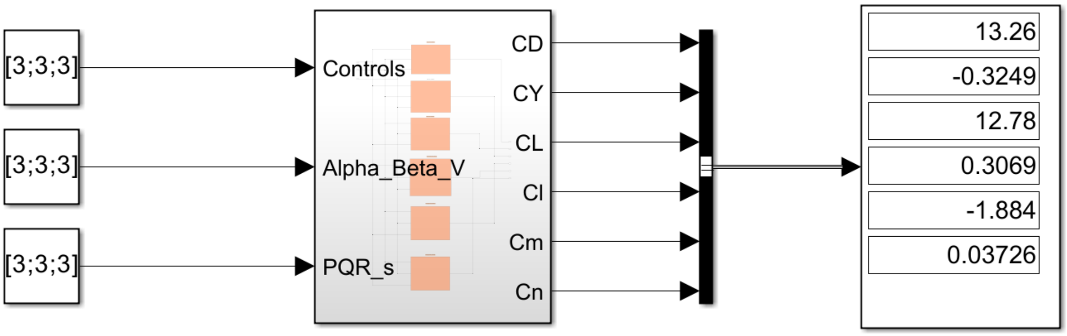

Figure 20: Checking Outputs verification- Aerodynamic Coefficients

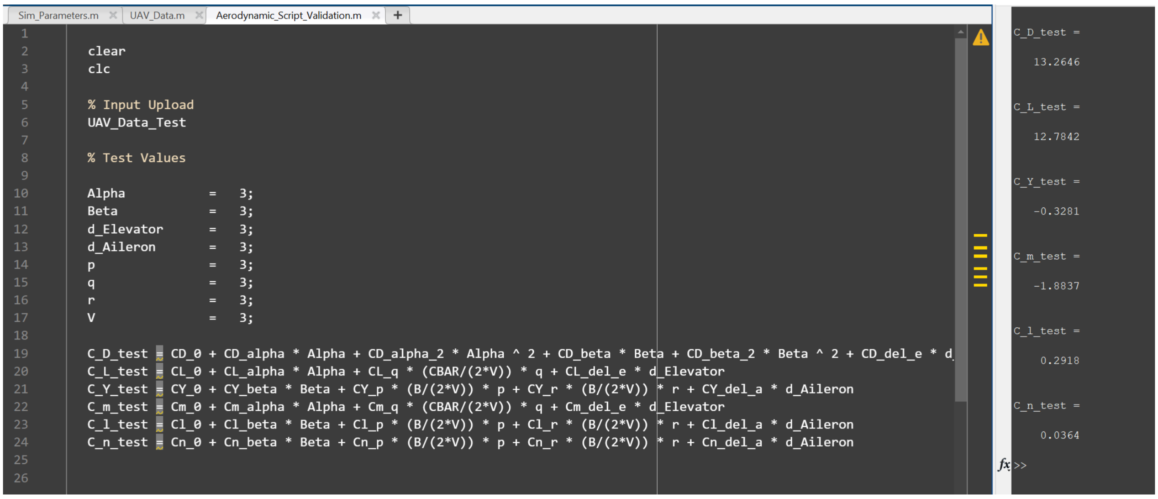

Figure 21: Checking Outputs verification- Aerodynamic Coefficients

Note: File name is the compressed file is Aerodynamic_Script_Validation.m

2.5.1 Verification Techniques - Checking Control Inputs

In order to check the control inputs, some steps should be followed. For the throttle deviation, the first step is to give an altitude and speed value to the initial conditions and leave the other initial conditions in zero. The following initial conditions were set:

Initial Conditions

Init_Pos = [0.0, 0.0, -1000] ;

Init_Vel = [5, 0.0, 0.0] ;
Init_Euler = [0.0, 0.0, 0.0] ;
Init_Rate = [0.0, 0.0, 0.0] ;
Secondly, A positive value for the throttle must be set, in this case $\delta_T = 0.3$

And finally, The correct result should be an increasing speed as shown in Figure 22.

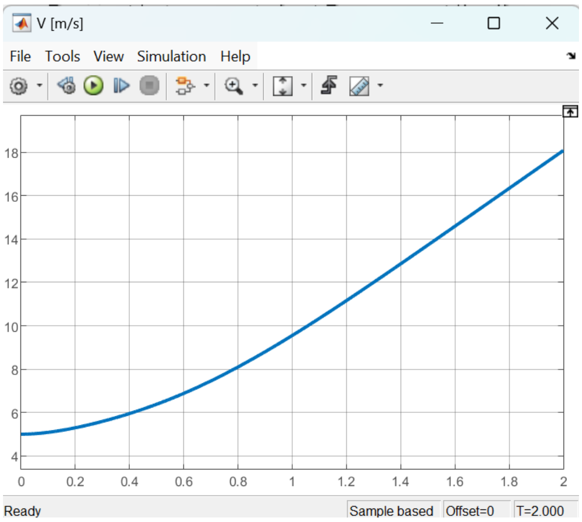

Figure 22: Speed Response with $\delta_T = 0.3$(Axis:speed[m/s]/Time[sec])

The following control input verification will be carried in the elevator deviation δ_e. First, the same initial conditions used for Throttle deviation δ_e will be used. The next step is to give a positive value to the elevator input, in this case $\delta_e = 0.5$. The positive elevator should lead to a negative pitch rate, as shown in Figure 23, due to the aerodynamic coefficient $C_{m_{\delta_e}} < 0$ in the model.

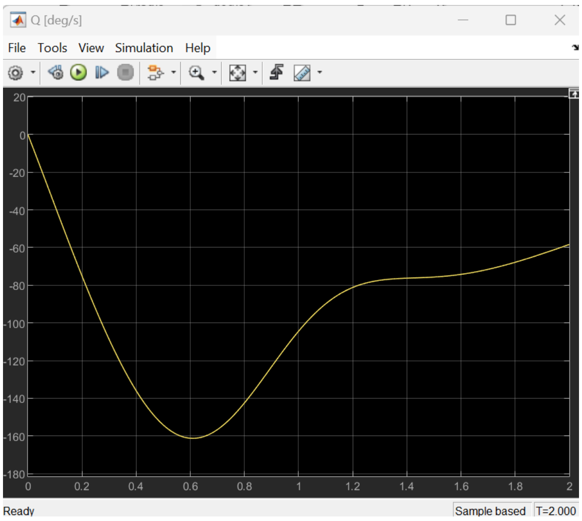

Figure 23: Pitch Rate Response with $\delta_e = 0.5$(Axis:Q[deg/s]/Time[sec])

For the Aileron Deflection δ_a, the same initial conditions are taken, the imposed positive value of δ_e will be 0.5, and finally, the positive aileron should lead to a positive roll rate, as shown in Figure 24, due to the aerodynamic coefficient $C_{l_{\delta_a}} < 0$ in the model.

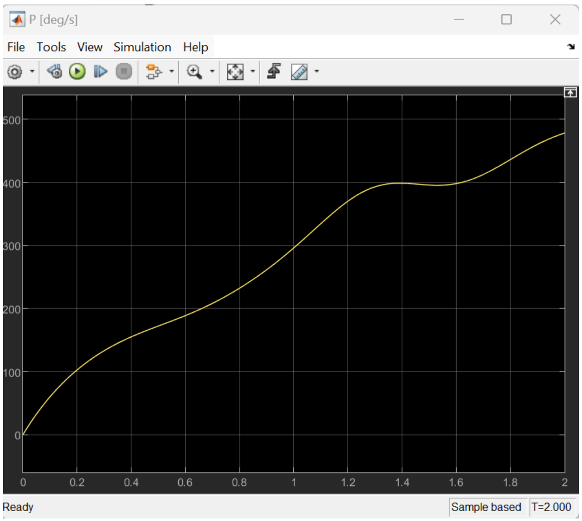

Figure 24: Roll Rate Response with $\delta_a = 0.5$(Axis:P[deg/s]/Time[sec])

Note: File name is the compressed file is Cotrol_Inputs_Verification.m

2.5.2 Verification Techniques - Checking Dynamics Modes

Steps for the Verification of Phugoid mode:

- Give some altitude and foreword speed, other initial conditions are zero. Initial Conditions
  Init_Pos = [0.0, 0.0, -1000] ;
  Init_Vel = [5, 0.0, 0.0] ;
  Init_Euler = [0.0, 0.0, 0.0] ;
  Init_Rate = [0.0, 0.0, 0.0] ;
- Exchange between potential and kinetic energy
- Imposing zero control inputs: $\delta_e = 0$, $\delta_a = 0$ and $\delta_T = 0$.
- Simulation time = 100 sec

In Figures 25 and 26 are correctly presented the phugoid behaviour for the altitude and speed response.

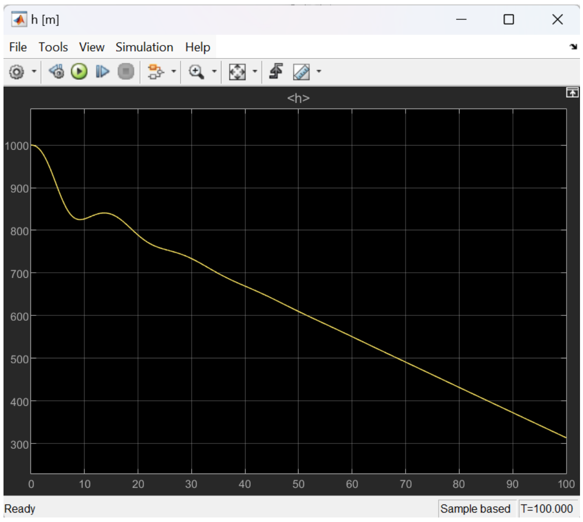

Figure 25: Altitude Response with $\delta_e = 0$, $\delta_a = 0$ and $\delta_T = 0$ (Axis:Altitude[m]/Time[sec])

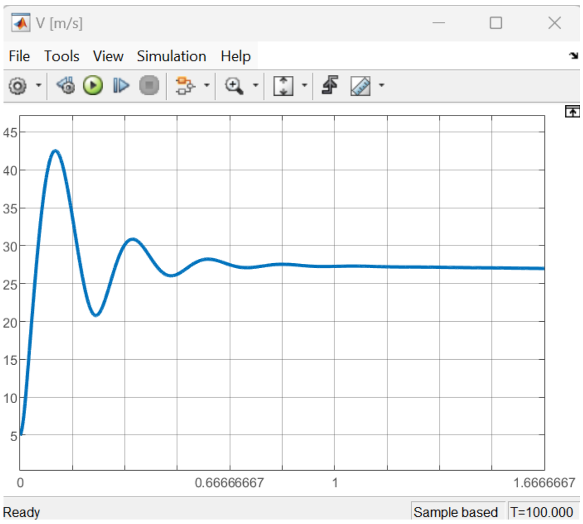

Figure 26: Speed Response with $\delta_e = 0$, $\delta_a = 0$ and $\delta_T = 0$ (Axis:Speed[m/s]/Time[sec])

Steps for the Verification of short-period mode:

- Give some altitude, foreword speed, and other initial conditions are zero. Initial Conditions
  Init_Pos = [0.0, 0.0, -1000] ;
  Init_Vel = [5, 0.0, 0.0] ;
  Init_Euler = [0.0, 0.0, 0.0] ;
  Init_Rate = [0.0, 0.0, 0.0] ;

- Imposing a elevator input only: $\delta_e = 0.3$, $\delta_a = 0$ and $\delta_T = 0$.

- Simulation time = 2 sec

In Figures 27 and 28 are correctly presented the short-period behaviour for the Pitch Rate Response and Z-Component of body Velocity response.

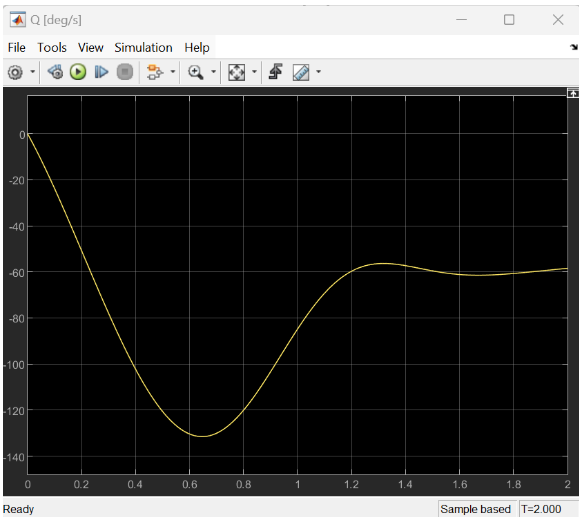

Figure 27: Pitch Rate Response with $\delta_e = 0$, $\delta_a = 0$ and $\delta_T = 0$ (Axis:Q[m]/Time[sec])

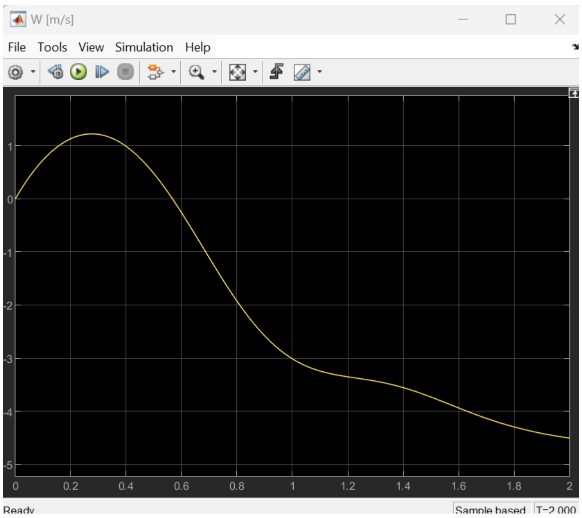

Figure 28: Z-Component of body Velocity response with $\delta_e = 0$, $\delta_a = 0$ and $\delta_T = 0$ (Axis:W[m/s]/Time[sec])

Note: File name is the compressed file is Model_Dynamic_Modes_Verification.m

3 Part 2 - Trimming/Linearization/Stability Analysis

The second task of the assignment is to find trim solutions at given operating conditions and then get linearized models at the trim conditions found. Additionally, it is required to assess the stability of the system

3.1 Process of Trimming and Linearization

Explain the process of trimming and linearization, incl. trimming constraints; discuss why we need these additional constraints:

1. What is linearization from the mathematical point of view? Linearization is a mathematical technique used to estimate the behavior of a nonlinear system near an equilibrium point. This process involves constructing a linear model by calculating the first-order Taylor series expansion of the nonlinear equations around the trim condition. This linear model offers a locally accurate representation of the system's dynamics[1].

   When the control vector is specified using the 6-degree-of-freedom (6-DOF) equations derived from the nonlinearized model, it is time to consider small perturbations deviated from the steady-state condition, which originate from deviations in Xe and Ue. This results in a set of linear constant-coefficient state equations. By expanding the nonlinear state equations through a Taylor series about the equilibrium point and considering only the first-order terms, the perturbations in the state can be identified[1].

   These perturbations, expressed as scalar products, are then organized into a vector known as the state vector. This process leads to obtaining the implicit linear state variables along with their Jacobian matrices E, A, and B [1].

2. Why do we need Trimming and Linearization? Trimming allows us to find operating points where the aircraft is in a steady state, simplifying the complex nonlinear equations governing its motion. Linearization transforms these nonlinear equations into linear ones, enabling the use of analytical and computational tools from linear control theory. This simplification is vital for stability analysis, controller design, and performance evaluation. Aerospace systems, especially aircraft, exhibit highly nonlinear behavior. Trimming and linearization provide a means to handle this complexity, making the system responsive to systematic analysis and control synthesis.

3. What is the trim point from the point of view of the governing equation? The trim point is a specific set of state and control variables at which the system is in equilibrium. In the context of the governing equations, it is the point where the time derivatives of the state variables are zero. The aircraft is not accelerating or decelerating; it is flying in a steady-state condition. The trim point represents an operational equilibrium, and deviations from this point can be analyzed using the linearized model [3] [1].

   Trim Point and Governing Equations

   The trim point in the context of an aircraft's dynamics is a specific set of state (x) and control (u) variables where the system is in equilibrium. This means that at the trim point, the aircraft is in a steady-state condition with no net change in its motion. The trim point is crucial for understanding the behavior of the system under specific operating conditions.

   Governing Equations: The dynamics of an aircraft are typically described by a set of nonlinear ordinary differential equations. These equations govern the evolution of the state variables (x) over time.

Trim Point Conditions: At the trim point, the system is in equilibrium, meaning $\dot{x} = 0$. Therefore, the trim conditions can be expressed as:

$$f(\bar{x}, \bar{u}) = 0$$

where: - $\bar{x}$ is the vector of state variables at trim, - $\bar{u}$ is the vector of control inputs at trim, and - $f(\bar{x}, \bar{u})$ is the system dynamics evaluated at the trim point.

Analyzing Deviations: Deviations from the trim point can be represented by introducing perturbations $(\Delta x, \Delta u)$ to the trim conditions [3] [1]:

$$f(\bar{x} + \Delta x, \bar{u} + \Delta u) = 0$$

These deviations can be linearized to analyze the small changes around the trim point. The linearized equations are often obtained by considering only the first-order terms in a Taylor series expansion, resulting in a linear model, as stated before in this document [3] [1].

Understanding the trim point and its deviations is fundamental for stability analysis, control design, and predicting the aircraft's behavior under different operating conditions. It provides insights into how the system responds to disturbances and inputs around the equilibrium state [3] [1].

4. Why do we need to apply constraints? Constraints are integral to the trimming process to maintain physical feasibility. Aerospace systems have limitations on control surface deflections, velocities, and other variables. Applying constraints ensures that the trim condition obtained is physically realistic and within the operational limits of the system. This is crucial for achieving meaningful results that reflect the true behavior of the aircraft under consideration.

   Without constraints, the trimming process might yield solutions that are impractical or violate physical boundaries. Constraints act as a safeguard against such unphysical configurations, allowing us to focus on trim conditions that are relevant and applicable to real-world scenarios.

3.2 Finding Trim Solutions and Trim Analysis

Build the simulation model for trimming and the simulation model for checking the obtained trim solutions

3.2.1 Find a trim solution at the operating condition airspeed V = 15 m/s and h = 0 km (Altitude). Verify the trim solution found using the simulation model for checking trim solutions.

The Trimming model was developed by copying the original model and modifying it according to class instructions. The new modeling system eliminates the autopilot and control modeling, making it dependent on input values for Elevator, Aileron, and Throttle. The UAV Model system generates outputs in Euler Angles, Rate values, Vb behavior, and the components α, β, and V.

Figures 29 and 30 represent the Trimming evaluation model and the verification model, respectively. Due to issues with the Matlab Trim package, the Simulink interactive system "Model Linearizer" will be used at its specified operating point "Trim condition" to establish trimming conditions considering the type of controls and the UAV model.

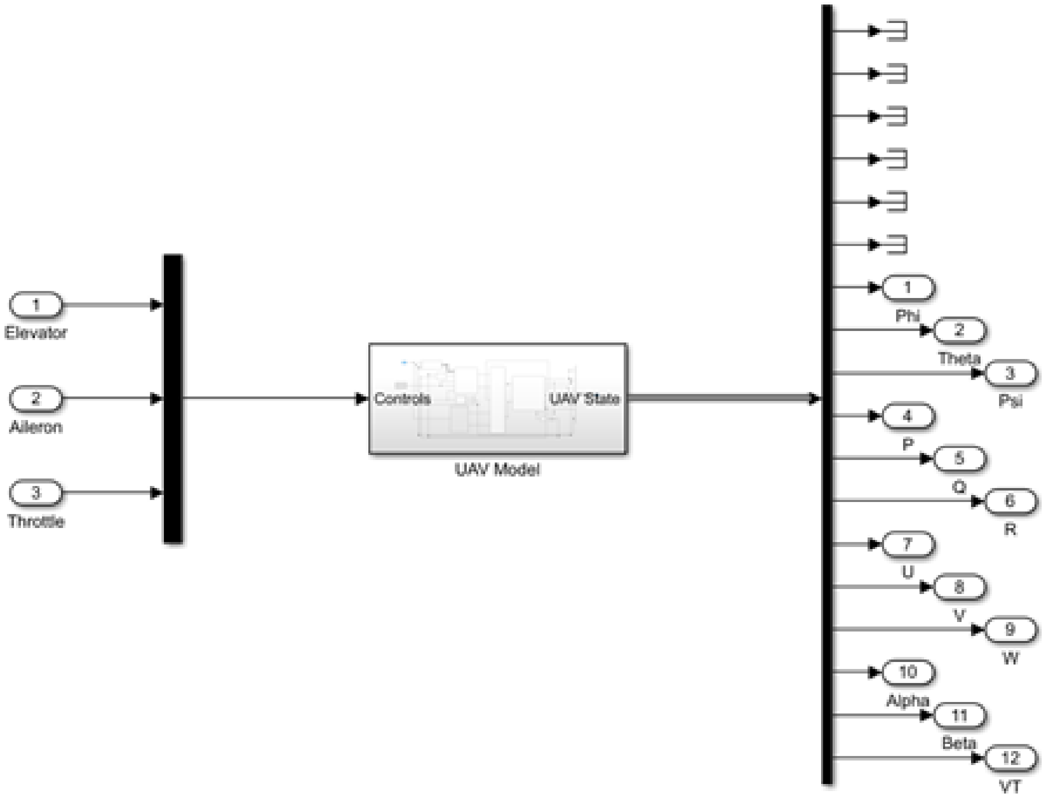

Figure 29: Trimming evaluation model

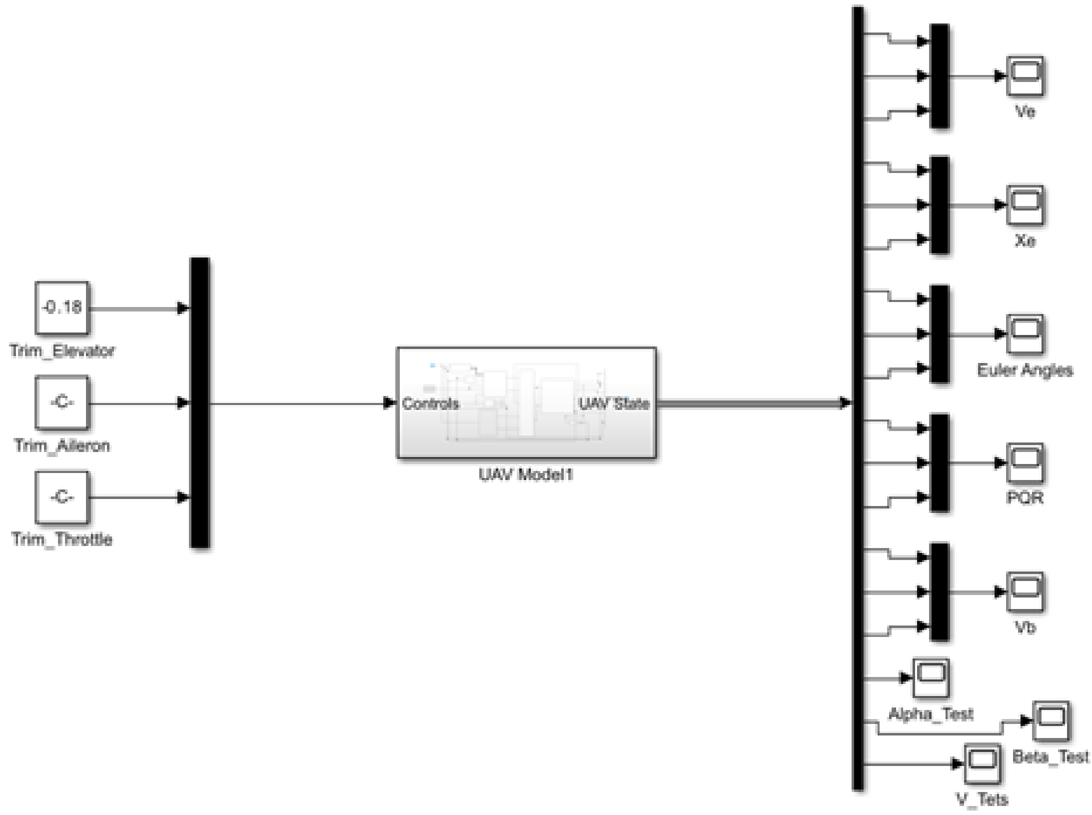

Figure 30: Trimming Verification model

According to the requested analysis order, the first case specifies airspeed conditions of V=15m/s and h=0 km. The trimming results for elevator and throttle are presented below:

$$\delta_e = -0.14843, \delta_{th} = -0.85985$$

Verification of the results shows behavior with values equal to 0 for most of the analyzed components (desired behavior) and specifies the following results in terms of Ze and Ub (Behaviors specified for the trim). Likewise, the figures shown in Figures 31 to 33 represent the validation of the Trimming.

$$Z_e = 0, u_b = 13.3356, \theta = 0.1182$$

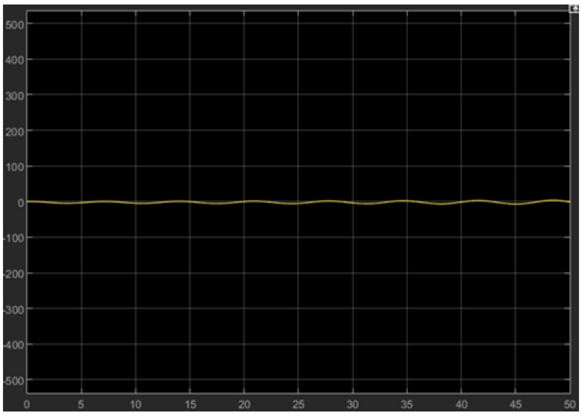

Figure 31: Trajectory demonstrating the need for the UAV to maintain a 0 km altitude km/sec

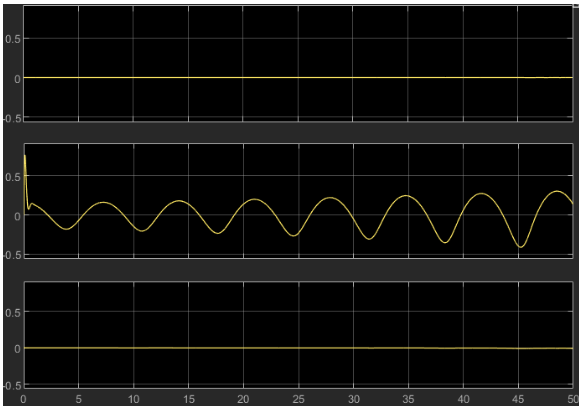

Figure 32: PQR system demonstrating the desired behavior of maintaining a 0 position

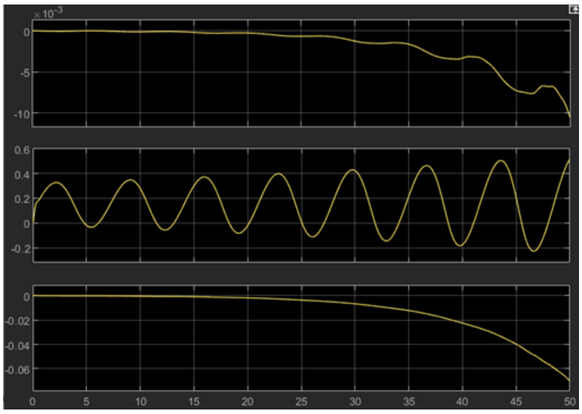

Figure 33: Euler angle system where values are maintained within a desired range for trimming conditions deg/s

3.2.2 Find trim solutions at the fixed altitude h = 0 m and various speeds V = [15, 30, 45, 60] m/s. In this case, determine the trim angle-of-attack (i.e.,(α_e= atan2(w_e,u_e), the trim pitch control elevator deflection (i.e., δ_e), and the trim throttle δ_T). Discuss patterns of these parameters as speed increases. Also, please, discuss the reason behind this behaviour.

Having verified the effectiveness of trimming, an analysis is performed for the different proposed speed conditions with a fixed h=0 for all models.

Table 1: Trim Solutions

V(m/s)	h(km)	α	δ_e	δ_{th}
15	0	0.1182	-0.14843	0.85985
30	0	0.026984	-0.064264	1.8759
45	0	0.0097772	-0.048388	3.3664
60	0	0.0037553	-0.042832	4.809

As the speed increases, the aircraft requires less angle-of-attack and elevator deflection for trim due to the increased effectiveness of aerodynamic forces at higher speeds. However, the increased drag necessitates higher engine power, reflected in the higher throttle setting required for trim.

3.2.3 Find trim solutions at the fixed airspeed V = 15 m/s and various altitudes as h = [0, 1, 2, 3, 4] km. In this case, determine the trim angle-of-attack (i.e.,(α_e= atan2(w_e,u_e), the trim pitch control elevator deflection (i.e., δ_e), and the trim throttle (δ_T). Discuss patterns of these parameters as Altitude increases. Also, please, discuss the reason behind such a behaviour.

For the following analysis, a fixed speed of V=15 m/s will be considered by varying the altitudes, and the following results were obtained:

Table 2: Trim Solutions

V(m/s)	h(km)	α	δ_e	δ_{th}
15	0	0.1182	-0.14843	0.85985
15	1000	0.13017	-0.15947	0.90061
15	2000	0.1435	-0.17177	0.95699
15	3000	0.15835	-0.18548	1.0282
15	4000	0.17489	-0.20074	1.1131

At higher altitudes, the air density decreases, resulting in reduced lift generation. To compensate for the decreased lift, the aircraft needs to increase its angle-of-attack to maintain level flight.

As the angle-of-attack increases, the aircraft's nose-up tendency also increases, requiring a larger upward deflection of the elevator to maintain level flight.

At higher altitudes, the air density decreases, leading to a decrease in engine thrust. To counteract this reduction in thrust and maintain the desired airspeed, the throttle setting must be increased.

3.3 Numerical Linearization

Build the simulation model for linearization. Perform numerical linearization at the operating condition V = 15 m/s and h= 0 m. Determine the linear time invariant (LTI) model of the longitudinal motion and the LTI model of the lateral motion.

For the application of Numerical Linearization, the Matlab function "linmod" will be used in conjunction with the data obtained from the V=15 m/s and h=0 case, which is specified below.

$$X_0 = \begin{bmatrix} X_e \\ Y_e \\ Z_e \\ P \\ Q \\ R \\ \phi \\ \theta \\ \psi \\ U_b \\ V_b \\ W_b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0.1182 \\ 0 \\ 15 \\ 0 \\ 1.7813 \end{bmatrix} ; u_0 = \begin{bmatrix} \delta_e \\ \delta_a \\ \delta_T \end{bmatrix} = \begin{bmatrix} -0.14843 \\ 0 \\ 0.859585 \end{bmatrix}$$

The LTI model of the longitudinal and lateral motion results is presented in Figgure 34

```
long_sys =

  A =
             x1        x2        x3        x4
   x1   0.01697     0.788    -1.597    -9.742
   x2   -0.4992    -6.734     13.45    -1.157
   x3    0.7876    -6.632    -5.891         0
   x4         0         0         1         0

  B =
            u1
   x1   -3.482
   x2   -22.82
   x3   -109.3
   x4        0

  C =
       x1  x2  x3  x4
   y1   1   0   0   0
   y2   0   1   0   0
   y3   0   0   1   0
   y4   0   0   0   1

  D =
       u1
   y1   0
   y2   0
   y3   0
   y4   0
```

(a) LTI model of the longitudinal motion

```
lat_sys =

  A =
             x1        x2        x3        x4
   x1   -0.2616     1.755    -15.03     9.742
   x2    -2.794    -12.66     1.115         0
   x3    0.6552     -3.23    -0.184         0
   x4         0         1    0.1188         0

  B =
            u1
   x1   0.9193
   x2    232.4
   x3    51.76
   x4        0

  C =
       x1  x2  x3  x4
   y1   1   0   0   0
   y2   0   1   0   0
   y3   0   0   1   0
   y4   0   0   0   1

  D =
       u1
   y1   0
   y2   0
   y3   0
   y4   0
```

(b) LTI model of the lateral motion

Figure 34: LTI model Results

3.3.1 Stability Analysis

Determine the dynamics modes of the longitudinal and lateral motion. Discuss the dynamic characteristics of these modes in the terms of the stability, the natural frequency, and the damping ratio (plot zero-poles map). Identify these modes in the state responses

Taking into account the stability results, the following behavior can be observed in terms of the longitudinal system.

$$\lambda_{short} = -6.3144 \pm 9.4878i, \omega_{short} = 11.3970, \zeta_{short} = 0.5540$$

$$\lambda_{phugoid} = 0.0104 \pm 0.8078i, \omega_{phugoid} = 0.8078, \zeta_{phugoid} = -0.0129$$

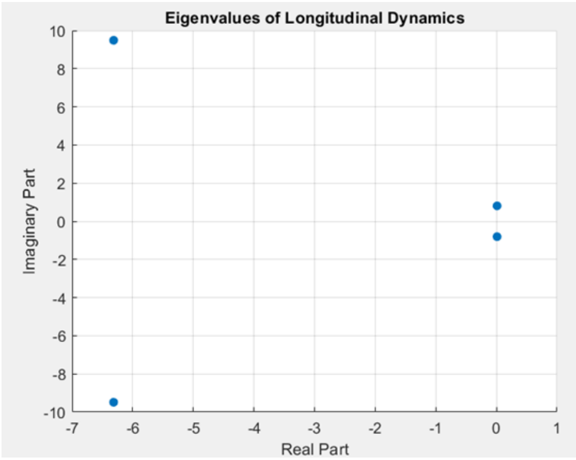

Figure 35: Dynamic Mode Longitudinal Motion

On the other hand, the lateral system adheres to the following results.

$$\lambda_{roll} = -12.9738\pm, \omega_{roll} = 12.9738, \zeta_{roll} = 1$$

$$\lambda_{Dutchroll} = -0.1042 \pm 4.7261i, \omega_{Dutchroll} = 4.7272, \zeta_{Dutchroll} = 0.0220$$

$$\lambda_{spiral} = 0.0764, \omega_{spiral} = 0.0764, \zeta_{spiral} = -1$$

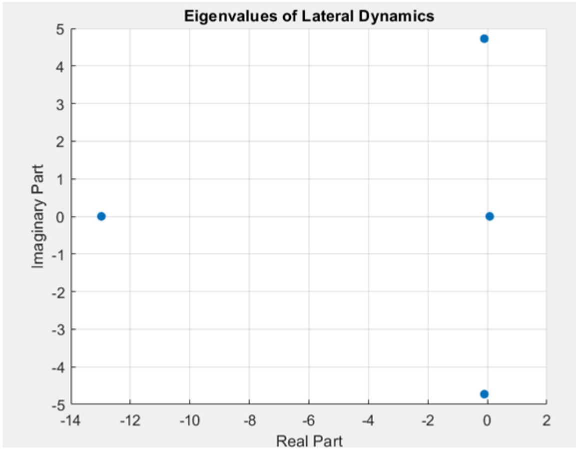

Figure 36: Dynamic Mode Lateral Motion

In terms of longitudinal motion, the short period mode exhibits stability with negative real parts of the eigenvalues, indicating a stable behavior. Additionally, the damping ratio (ζ_{short}) is 0.5540, suggesting moderately damped oscillations in pitch attitude.

Regarding the phugoid mode, the real parts of the eigenvalues are slightly positive, indicating marginal stability. The damping ratio ($\zeta_{phugoid}$) is close to zero, suggesting lightly damped oscillations in pitch altitude and airspeed.

In the context of lateral motion, the roll mode demonstrates stability with negative real parts of the eigenvalues, ensuring a stable behavior. The damping ratio (ζ_{roll}) is 1, indicating critically damped behavior in roll attitude.

Similarly, the Dutch roll mode also exhibits stability with negative real parts of the eigenvalues. The damping ratio ($\zeta_{Dutchroll}$) is 0.0220, suggesting lightly damped oscillations in both roll and yaw attitudes.

Lastly, the spiral mode shows marginal stability with a slightly positive real part of the eigenvalue. The damping ratio (ζ_{spiral}) is -1, indicating an unstable spiral motion. This mode represents a continuous, divergent turn of the aircraft.

Finally, to verify the longitudinal and lateral modes, a Simulink model is proposed as presented below:

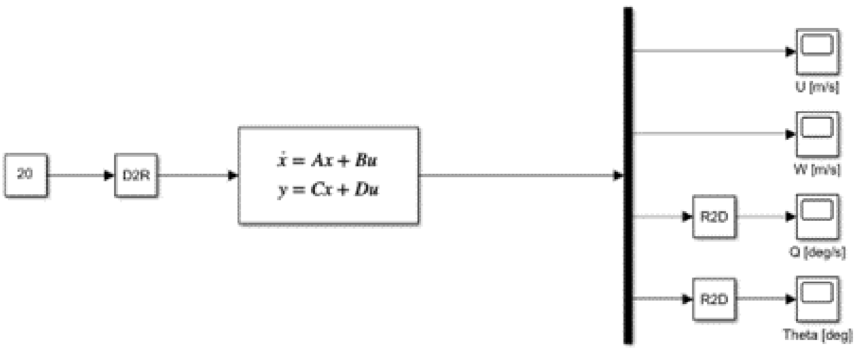

Figure 37: Longitudinal model with state-space block

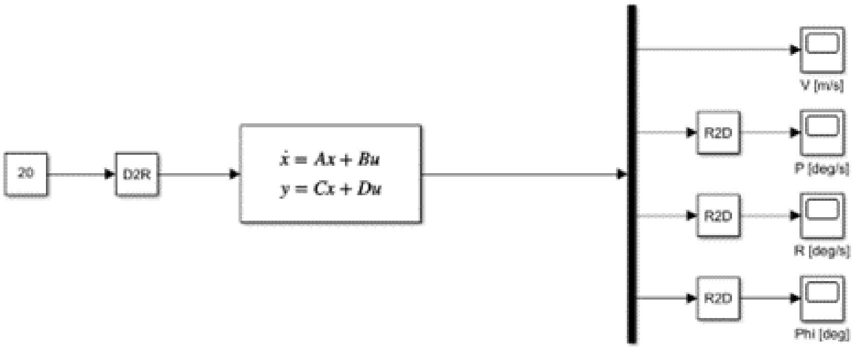

Figure 38: Lateral model with state-space block

For the longitudinal model analysis, the instructions for inputs will be followed with $\delta_e = 20$, where the following results were obtained for the Z-component of Body Velocity.

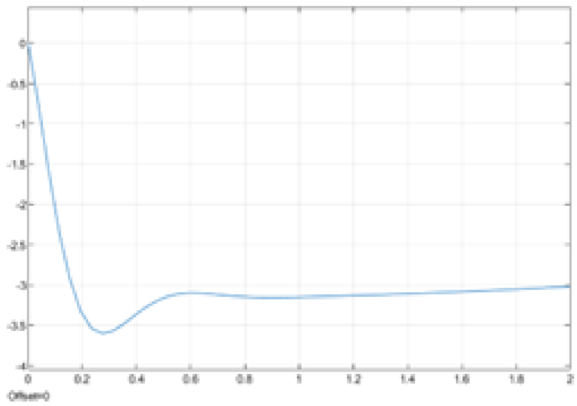

Figure 39: Z-component of Body Velocity

As observed in the previous images, the Z component represents a stable behavior for the short period.

4 Part 3 - Implementation of Control Laws

4.1 Implementing Control Law

The control laws consider the inclusion of three components: Speed Control, Pitch Control, and Roll Control. The representation of the system in SIMULINK software is shown in Figure 40.

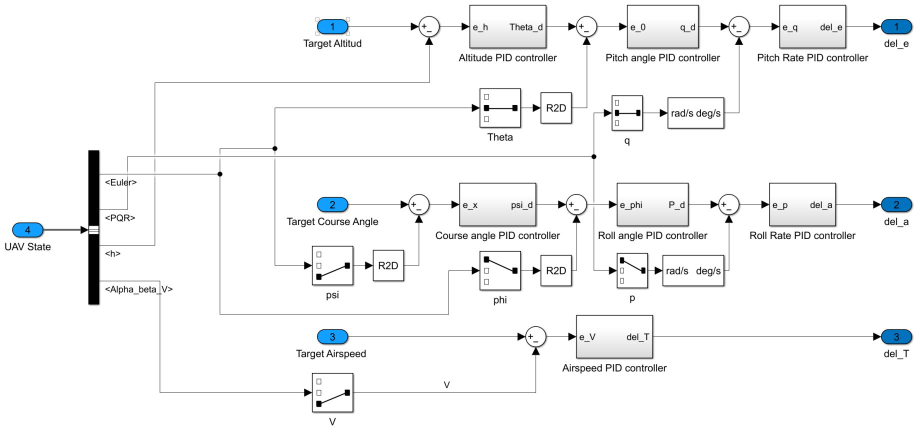

Figure 40: Autopilot Representation

To validate the Target Altiude, Course angle an Airspeed were set in h = 20, Course angle = 80 and V = 30, and then the complete model was run in order to check the results.

In Figure 41 is presented the altitude of the UAV during 50sec of simulation, as it can be seen after 20 sec the altitude is stabilized in 20m as requested in the velocity target.

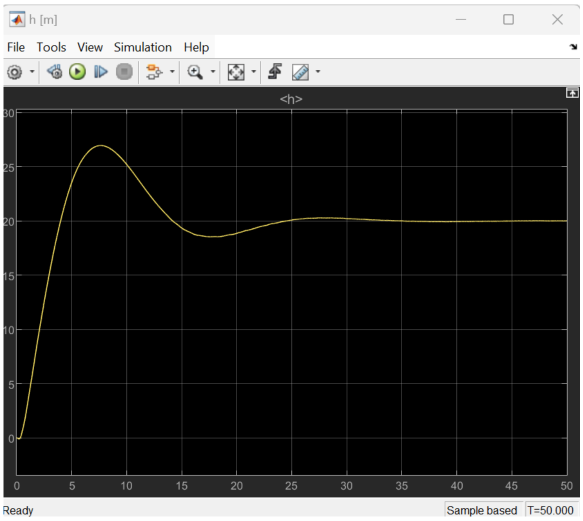

Figure 41: Altitude Target (20) m/s

The Angle target and the speed target present the correct stabilization points as shown in Figure 42 and 43.

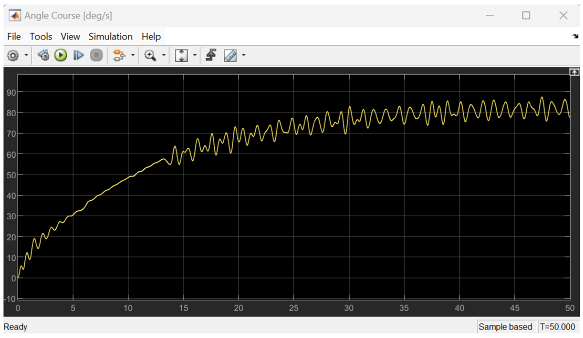

Figure 42: Angle Course Target (80) deg/s

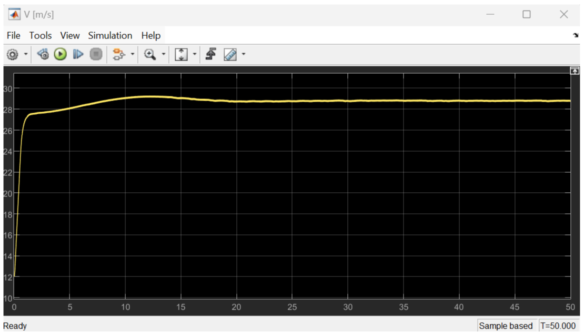

Figure 43: speed Target (30) m/s

References

[1] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, 2nd ed. wiley, 2003.

[2] M. Napolitano, *Aircraft Dynamics from modeling to simulation.* West Virginia: Wiley, 2012.

[3] M. V. Cook, *Flight Dynamics Principles.* Oxford: Elsevier, 2007.