Table of Content

Part 1 – Modelling

Model Conceptualization

Figure 1 shows the general model concept of the nonlinear missile model.

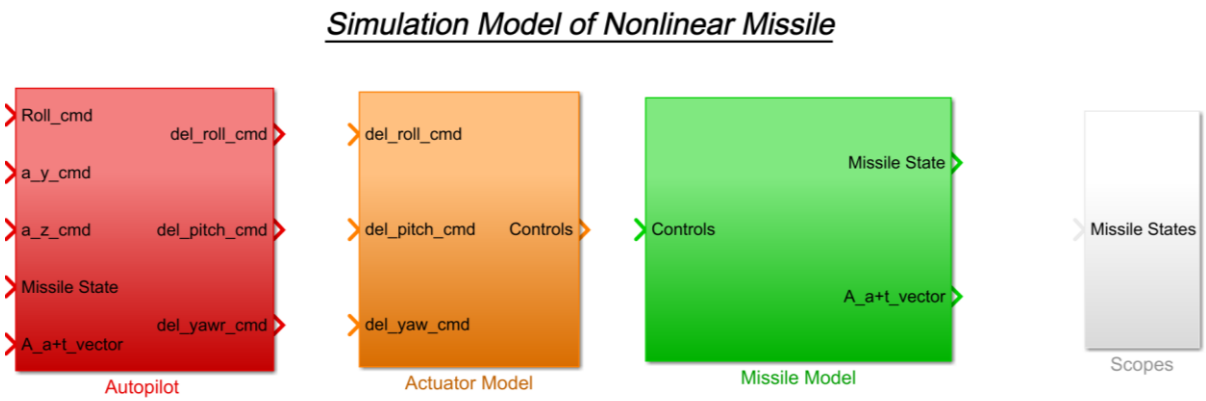

Figure 1 – Concept Model

Concept model of nonlinear missile has three main models and a scope block for outputs:

- Autopilot Model
- Actuator Model
- Missile Model

Initial conditions set by user are received by the autopilot and it calculates the control commands. PID controllers in the actuator block tolerate the system to have a better stability and calculate the control parameters once again by using the given transfer function. Missile model contains the aerodynamic and flight dynamic outputs of the missile such as Euler Rates, Euler Angles, Alpha, Beta and VT. Moreover, it calculates acceleration on y and z axes and missile states by using subsystems. These subsystems, aerodynamic model, external forces and equations of motion, are shown in the Figure 2.

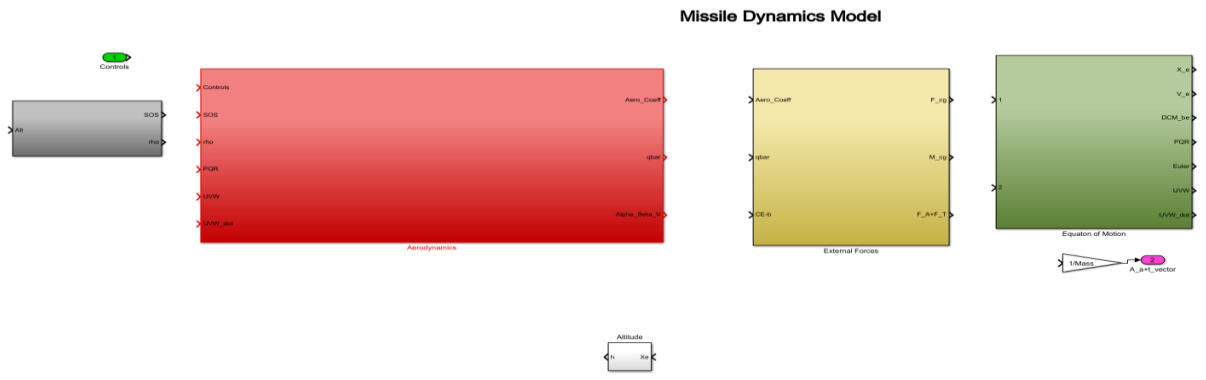

Figure 2 - Missile Model

6 degrees of freedom (6DOF) with Euler angles block is inside the equation of motion model and is used to estimate missile states. To estimate them, aerodynamic and external forces are used by equation of motion model. Using 6DOF with Euler angle has a benefit for model. This benefit is Euler angles straight associated with the axis of missile and it makes Euler angles easier to use than quaternions. Navigation and kinematic calculation are made by 6DOF block. Then, it feeds back aerodynamic model with body rotation rates (PQR), velocity (UVW) and acceleration (UVW_dot) to incidence (Alpha) and sideslip (Beta) angles, airspeed (V) and rotation rates in wind axes (PQR_s). These were used with speed of sound, density and control inputs to calculate aerodynamic coefficients in aerodynamic model.

Flight Dynamics

It examines and measures the specifications of systems that must take into consideration all six degrees of freedom. These are complex nonlinear equations that occur in both rigid and aerodynamic models. There are also many complex interactions between aerodynamics and rigid body. This complexity arises because the motion of an object in air can involve complex physics flight, and hence properly derived equations of motion must reflect this complexity. However, complex physics flight is absent in most practical flight modes. For instance, level flight at constant speed or simple manoeuvres must accept simpler equations because these types of flights do not use the dynamics of the vehicle. These are the special cases of the general 6dof equations of motion, and the vast majority of flights are limited to these special cases. Newton's second law obtains these equations of motion. They occur by equalizing the sum of external forces on the body to rate of change of momentum and external moments on the body to rate of change of angular momentum. Aerodynamic forces, thrust, weight are the external forces and aerodynamic moments and moment occurred by thrust are external moments. 6DOF equations of motion can be obtained as:

Forces are:

$$m(\dot{u} - vr + wq) = -mg\sin\theta + F_{AX} + F_{TX} \quad \text{Equation 1}$$

$$m(\dot{u} - ur + wp) = mg\sin\phi\cos\theta + F_{AY} + F_{TY} \quad \text{Equation 2}$$

$$m(\dot{w} - uq + vp) = mg\cos\phi\cos\theta + F_{AZ} + F_{TZ} \quad \text{Equation 3}$$

Moments are:

$$L_A + L_T = I_{xx}\dot{p} - I_{xz}\dot{r} - I_{xz}p + (I_{zz} - I_{yy})qr \quad \text{Equation 4}$$

$$M_A + M_T = I_{yy}\dot{p} + (I_{xx} - I_{zz})pr + I_{xz}(p^2 - r^2) \quad \text{Equation 5}$$

$$N_A + N_T = I_{xx}\dot{r} - I_{xz}\dot{p} + (I_{yy} - I_{xx})pq + I_{xz}qr \quad \text{Equation 6}$$

After simplification

$$L_A + L_T = I_{xx}\dot{p} \quad \text{Equation 7}$$

$$M_A + M_T = I_{yy}\dot{q} + (I_{xx} - I_{zz})pr \quad \text{Equation 8}$$

$$N_A + N_T = I_{zz}\dot{r} + (I_{yy} - I_{xx})pq \quad \text{Equation 9}$$

Modelling

Model of the missile without the autopilot is in figure 3 given below. The blocks are connected to each other in a proper way. For this part change in roll, pitch and yaw are taken as constants that equal to zero to calculate control parameters. Missile Model block takes these control parameters to calculate Missile States. The main goal of this part is to simulate a realistic model for a missile by using all 6DOF. Outputs are calculated by this model. Outputs are Euler angles aerodynamic rates, velocity, alpha, beta, propulsion forces, acceleration, position, V and aerodynamic forces.

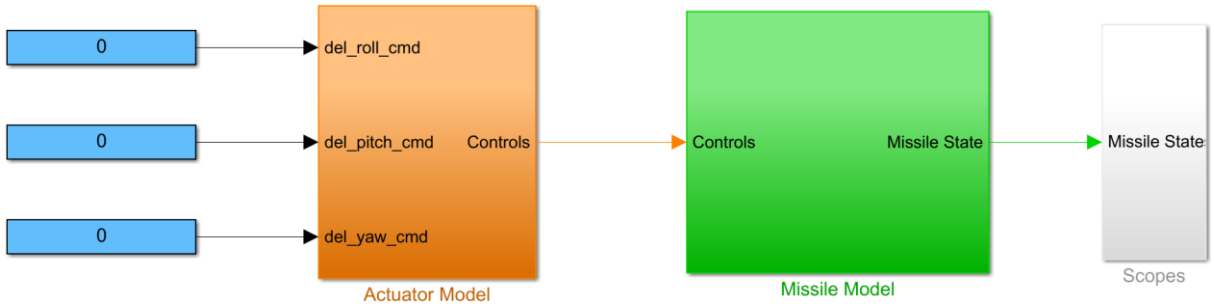

Figure 3 - Basic Block Diagram of the Missile

Effective control input is computed by plugging roll, pitch and yaw angles into provided equation on the paper after taking control parameters. Multiple distinct block outputs are connected to the aerodynamic block's input.

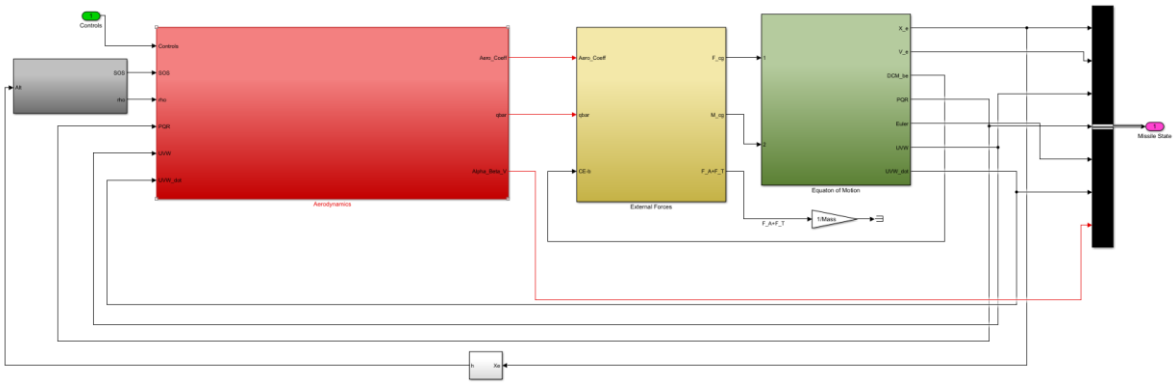

Figure 4 - Missile Model Block

Aerodynamic Model

Aerodynamic model has two subsystems, wind-axes velocities and aerodynamics coefficient, shown in the Figure 5. Wind-axes velocities block calculates angle of attack, side slip angle (alpha and beta), Mach number and dynamic pressure ($\bar{q}$). The forces and moment coefficients of the actuator model are afterwards derived using the estimated parameters as well as the control parameters. Although, the missile's coefficients are supplied in aeroballistics coordinates. Thus, axis conversion was used to convert body axis to aeroballistics. Additionally, feedback has come from 6dof to feed the aerodynamic model to calculate aerodynamic coefficients. These coefficients were calculated by outputs of wind-axes velocities block, controls input and feedback loop of 6dof.

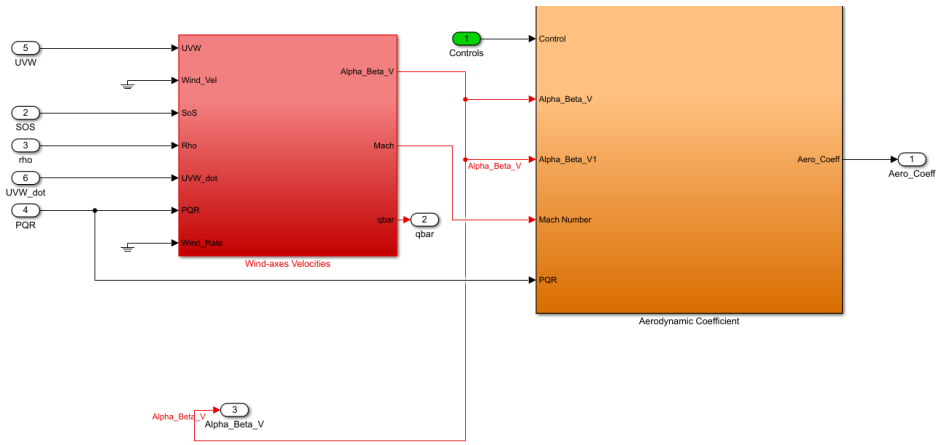

Figure 5 - Inputs Connection of Aerodynamic Block

Conversion from body axes to aeroballistics axes has been applied by using the equations:

$$\begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} = [C_b^a]^T \begin{bmatrix} C_x^a \\ C_y^a \\ C_z^a \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_T & \sin\phi_T \\ 0 & -\sin\phi_T & \cos\phi_T \end{bmatrix} \begin{bmatrix} C_x^a \\ C_y^a \\ C_z^a \end{bmatrix} \quad \text{Equation 10}$$

$$\begin{bmatrix} C_L \\ C_M \\ C_N \end{bmatrix} = [C_b^a]^T \begin{bmatrix} C_L^a \\ C_M^a \\ C_N^a \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_T & \sin\phi_T \\ 0 & -\sin\phi_T & \cos\phi_T \end{bmatrix} \begin{bmatrix} C_L^a \\ C_M^a \\ C_N^a \end{bmatrix} \quad \text{Equation 11}$$

Mux gathered together the aerodynamic coefficients. Aerodynamic forces and moments were calculated by using aerodynamic coefficients by feeding the external forces subsystem.

External Forces

External forces are force caused by aerodynamic coefficients, dynamic pressure and thrust, and moment caused by aerodynamic coefficients, dynamic pressure and thrust. Thus, external forces are product of aerodynamic coefficients, dynamic pressure and thrust. It also uses feedback from equation of motion subsystem which transforms earth-fixed axes to body axes (DCM_{be}). These connections are shown in the Figure 6.

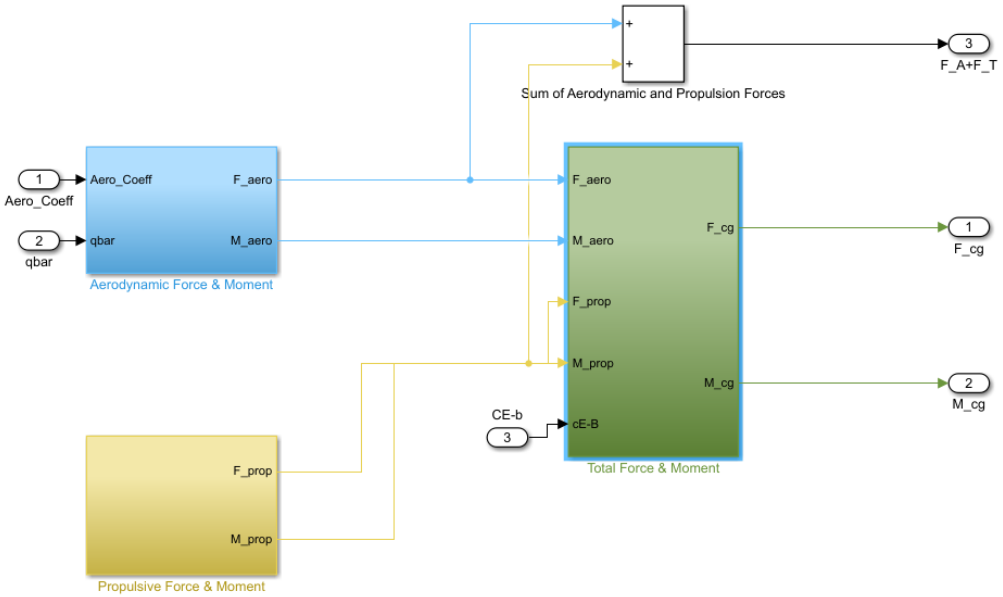

Figure 6 - External Forces Block

Aerodynamic forces and moments were calculated as follows

$$F_{Ax} = QSC_x \quad \text{Equation 12}$$
$$F_{Ay} = QSC_y \quad \text{Equation 13}$$
$$F_{Az} = QSC_z \quad \text{Equation 14}$$
$$L_A = QSDC_L \quad \text{Equation 15}$$

$$M_A = QSDC_M \quad \text{Equation 16}$$
$$N_A = QSDC_N \quad \text{Equation 17}$$

Although, moment and forces occurred by thrust is zero due to the absence of the thrust. Furthermore, there is not any gravity moment on the missile because of the rigid-body assumption.

Equations of Motion

As it was mentioned before 6DOF with Euler angles was used since it straight associated with the axis of missile and it makes Euler angles easier to use than quaternions. Initial conditions was set before to let the 6DOF block make the calculations.

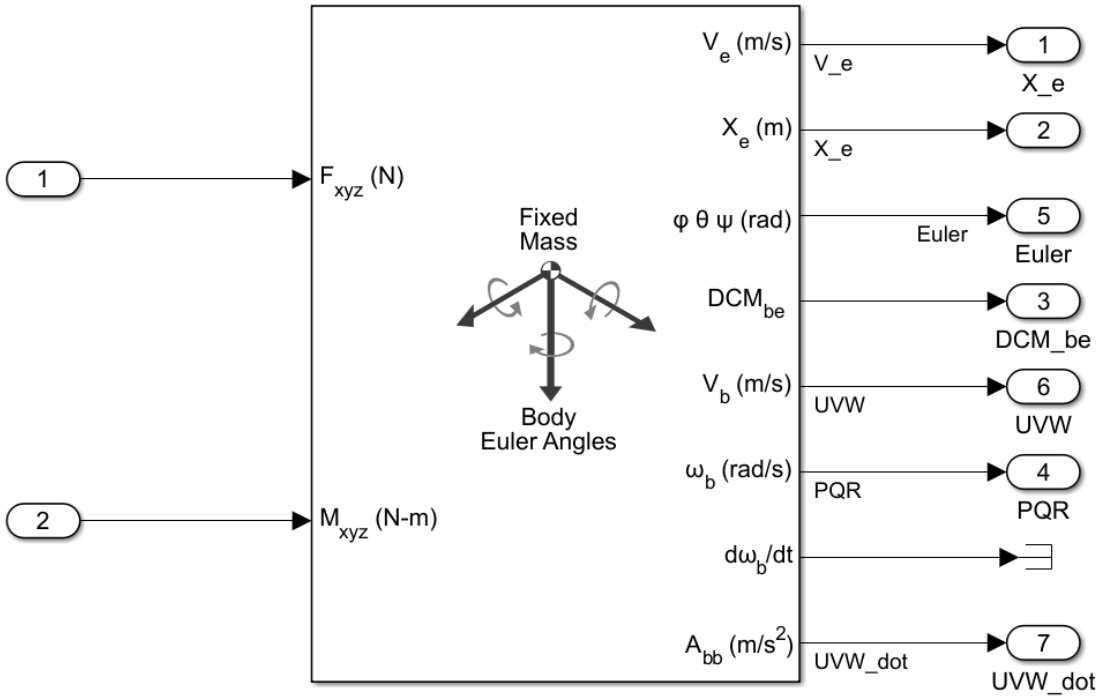

Figure 7 – 6DOF Equations of Motion

X_e was used as feedback to calculate altitude of the missile. This altitude information is helpful to calculate speed of sound and the density of the air (ρ) which help to calculate speed of sound and Mach number.

$$a = \sqrt{\gamma RT} \quad \text{Equation 18}$$
$$M = \frac{V}{a} \quad \text{Equation 19}$$

$where\ a\ is\ speed\ of\ sound$

$V\ vehicle\ speed$

$and\ M\ is\ Mach\ number$

T is the temperature at a given altitude in Kelvin, R and γ are constants.

Figure 8 – Calculation of SOS and ρ

Simulation results were obtained by using fixed step, 4th order differential equation (Runge-kutta method) and fixed step size which is 5.

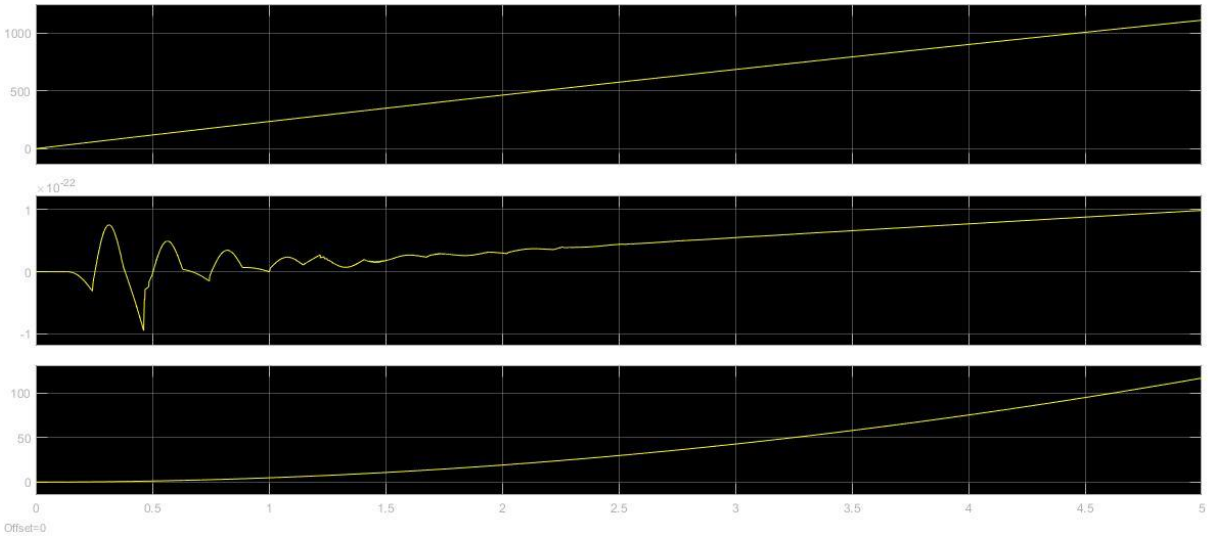

Figure 9 – Xe

Position change in Y axis is small enough to neglect. However, position in X axis increases demonstrating a correct route which is more than 1000 metres. Z was set as zero but increases slightly above 100 metres because there is a lift occurs due to the velocity of the missile. There is nothing to neutralize the lift since rigid-body assumption was made.

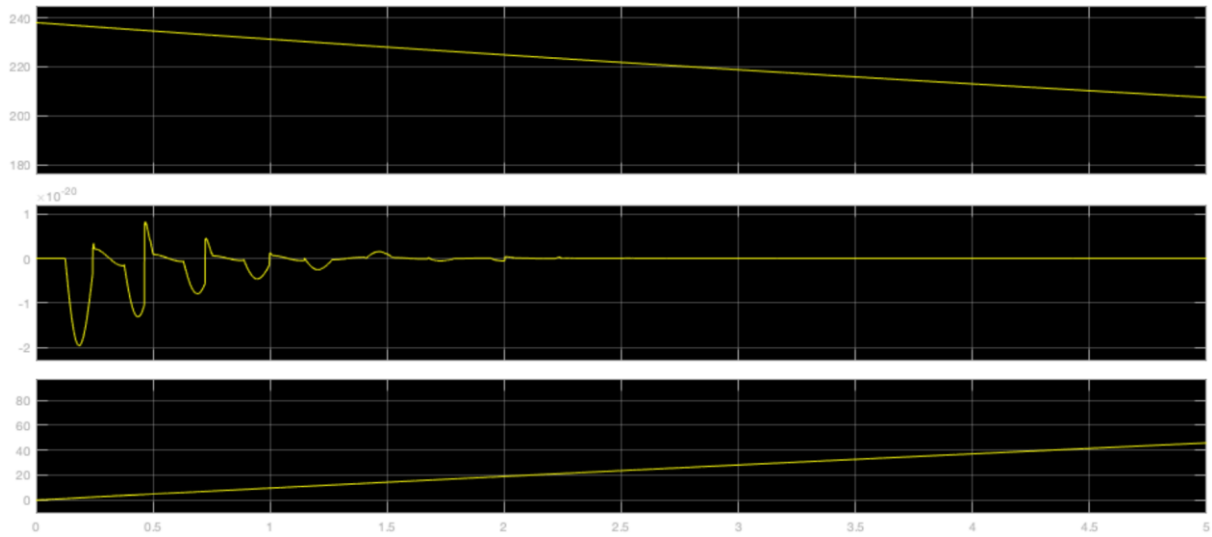

Figure 10 – Ve

The missile is getting slower in X axis as expected. Y component of the velocity oscillates at the beginning for 2 seconds but settles to 0. Additionally, Z component increases as expected, there is a velocity value in at least one of the other axes which creates lift force on the missile.

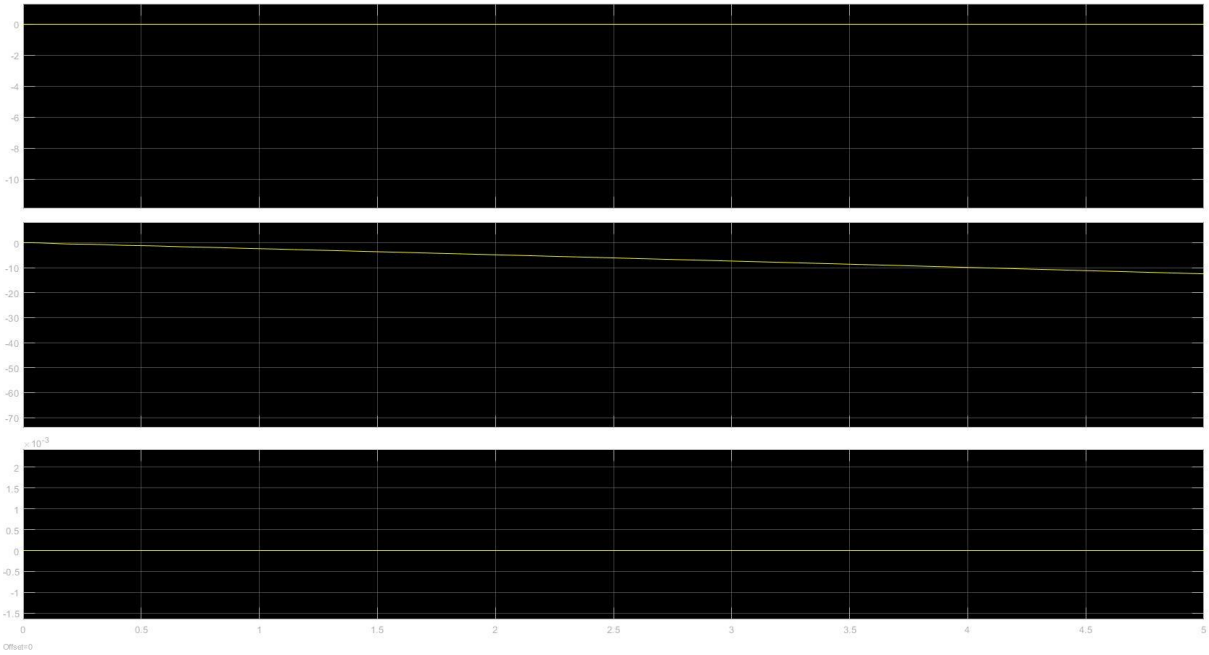

Figure 11 - Euler Angles

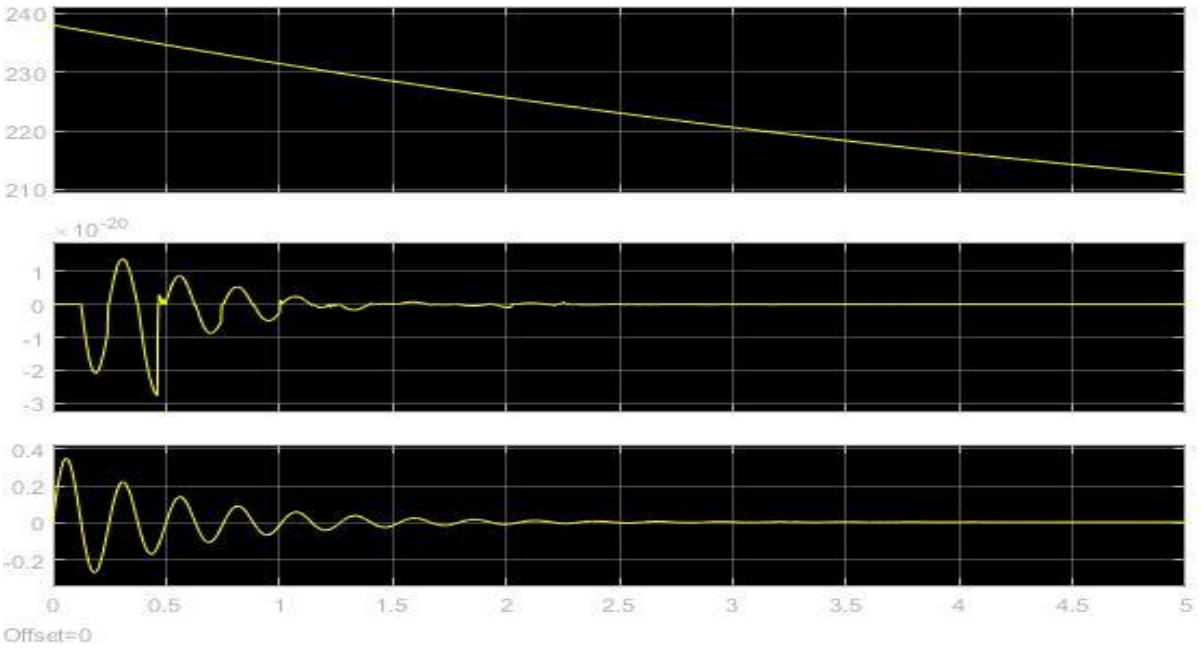

Figure 12 – UVW

V and W components of the angular velocity oscillate for 2 and 2.5 seconds respectively and then settle at zero. However, u component decreases logarithmically.

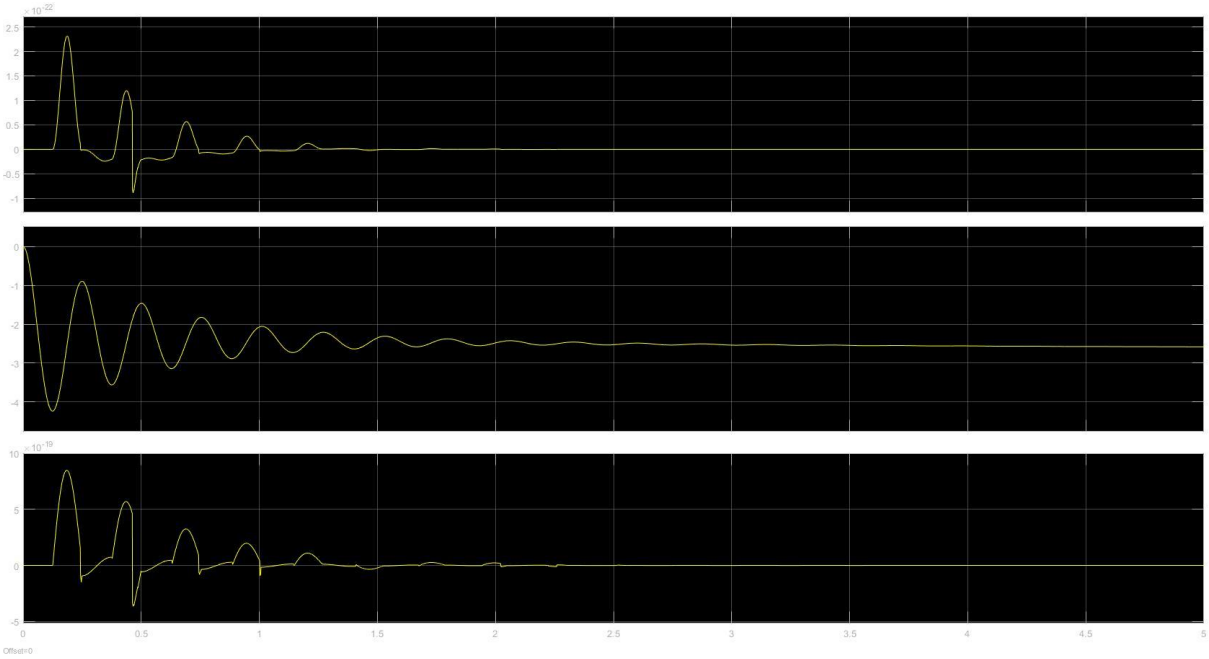

Figure 13 – PQR

Angular rates are oscillating for a period of time but eventually, they are settled.

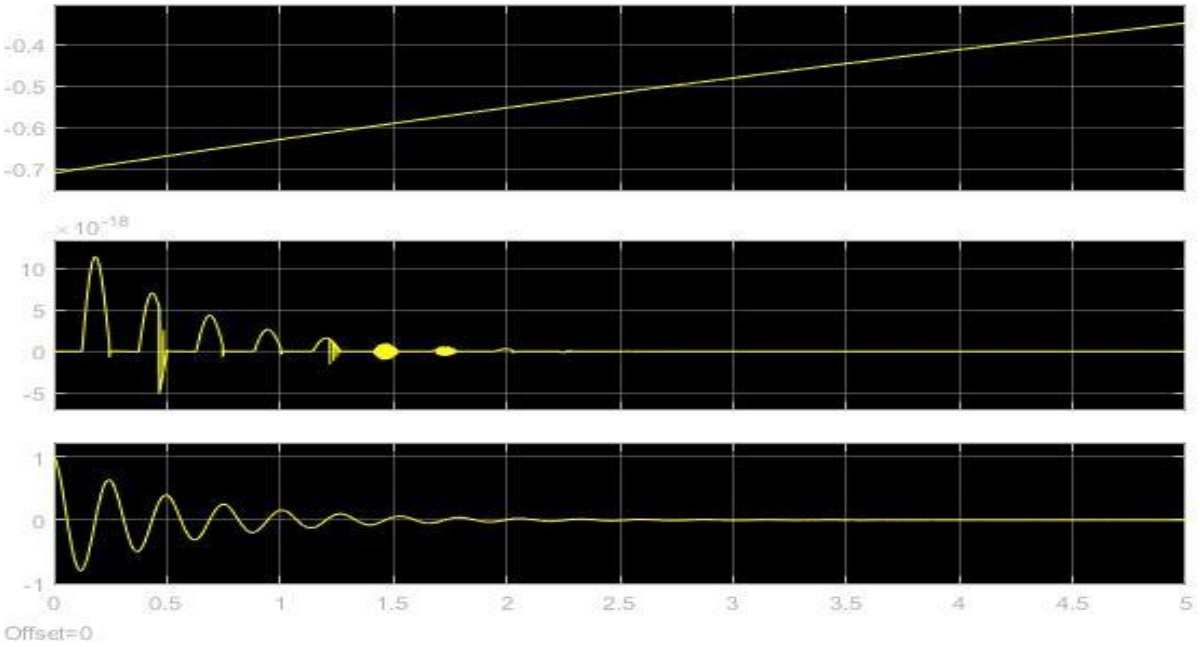

Figure 14 – Ab

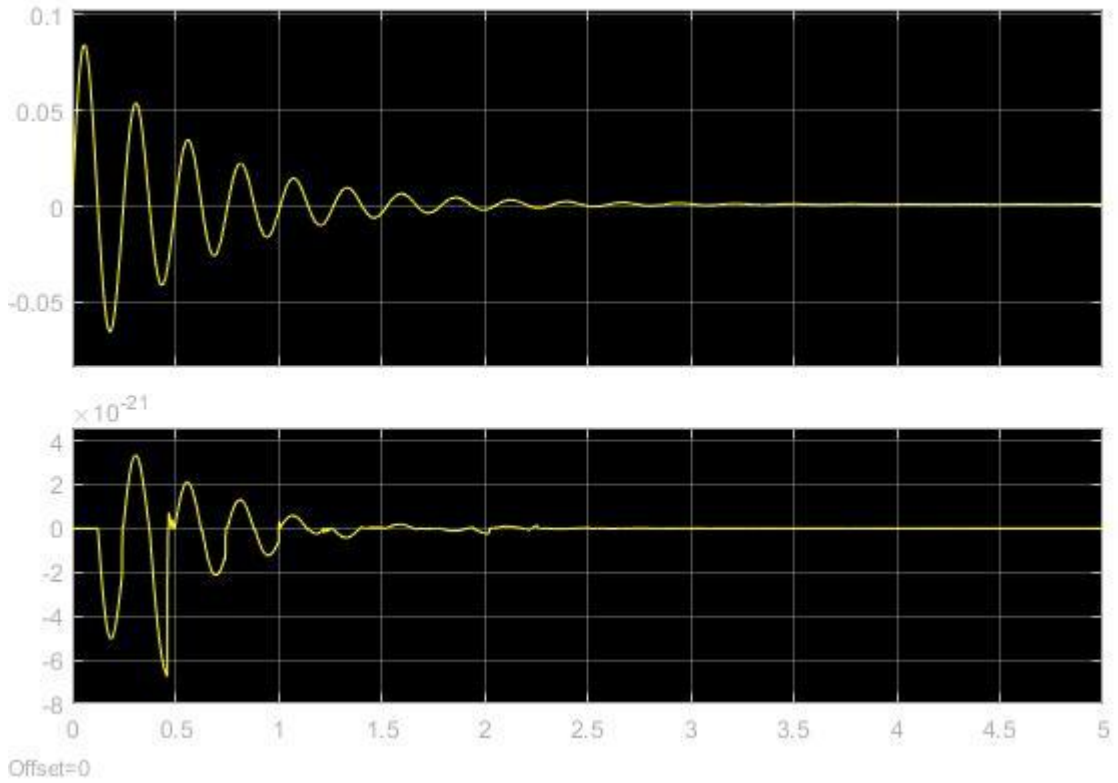

Figure 15 - Alpha & Beta

Both angle of attack and side slip angle (alpha and beta) oscillate due to aerodynamic forces occurred by the velocity components. However, they settle in a short period of time at zero. Alpha oscillates between higher values compare to beta. It is because of the lift again.

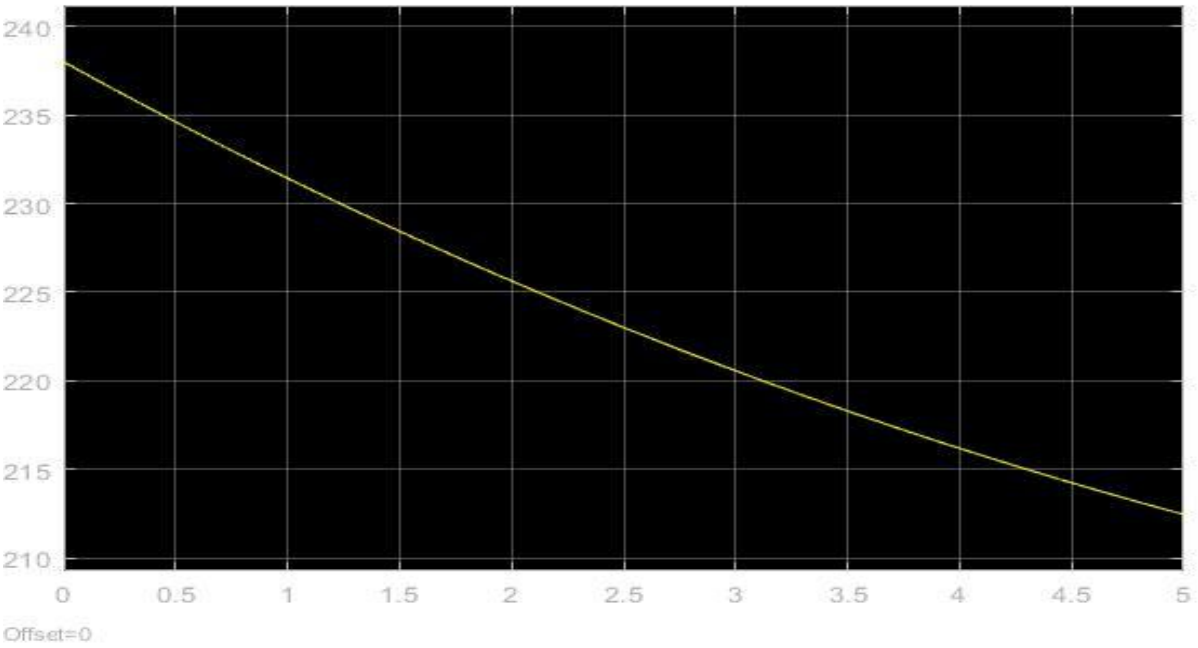

Figure 16 – VT

Figure 16 shows the total velocity graph over time and it decreases as time passes.

To sum up, the graphs showed that the missile seems to have the trend to increase in X axis and oscillations in roll angle. Increase in Z direction shows that there is a lift force on the missile causes this increase in velocity and change in the position. Although, Y axis seemed very passive it had some oscillations but, they are very scaled numbers.

Model Verification

MATLAB codes were implemented to verify output of the Simulink model. Aerodynamic coefficients obtained from codes and Simulink model were compared. Parameters were used for MATLAB codes are:

Table 1 Simulation Parameters

Parameters	Values
XCG	0.809
XREF	0.809
D	0.15
ρ	1.225
Speed of Sound a	340.2941
Angular Rates (P, Q, R)	(0, 0, 0)
Euler Angles (θ, ψ, ϕ)	(0, 0, 0)
Velocities (U, V, W)	(238.1876, 0, 0)
Control Surfaces $(\delta_R,\ \delta_P,\ \delta_Y)$	(0.0175, 0.0175, 0.0175)

Other needed parameters were obtained by using the following equations

$$V_b = \sqrt{U^2 + V^2 + W^2} \quad \text{Equation 20}$$

$$\alpha = \text{atan}\frac{W}{U} \quad \text{Equation 21}$$

$$\beta = \text{asin}\frac{V}{V_b} \quad \text{Equation 22}$$

$$Q = \frac{1}{2}\rho V^2 \quad \text{Equation 23}$$

$$M = \frac{V}{a} \quad \text{Equation 24}$$

$$\alpha_T = \text{acos}\, cos\alpha \cos\beta \quad \text{Equation 25}$$

$$\phi_T = \text{atan}\frac{\tan\beta}{\sin\alpha} \quad \text{Equation 26}$$

Aerodynamic coefficients are obtained from Simulink model and the MATLAB codes are shown in Figure 17 and 18. They have the same results which means the model is created correctly.

```
aero_coefficients =

   -0.3410    0.1325   -0.1325    0.1472   -0.5173   -0.0090
```

Figure 17 - Aerodynamic Coefficient from Mathematical Model

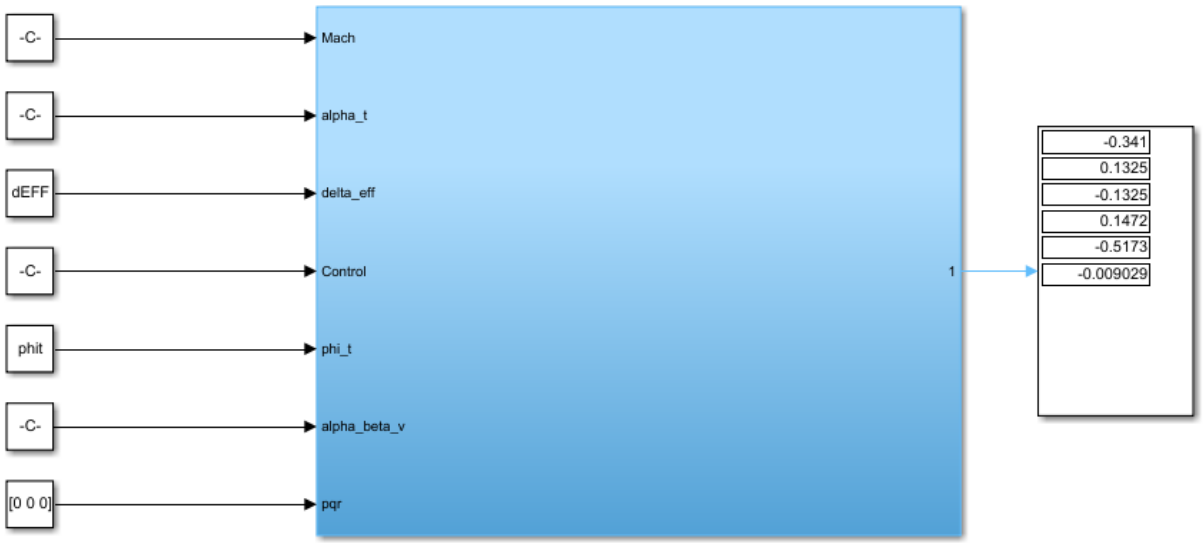

Figure 18 - Aerodynamic Coefficient from Simulation Model

Part 2 – Trimming/Linearization/Stability Analysis

Trimming, Linearization and Stability

Trimming is being used to obtain steady state flight conditions to let the contrast of the flight performance by discovery of the trim points since the missile model was a nonlinear. It looks for input, state and output values of the model that fulfil the required constraints. Later, the model's linearization is aided by the points discovered in the trimming. Additionally, the linearization of the model enables that to visualise diagrams, including the Pole-Zero map, Bode and Root-locus diagrams to assess the stability of the system. Finding a function's linear approximation at a certain point is known as linearization from a mathematical perspective. The First Order Taylor expression is used to approximate a nonlinear system around the target point. Higher ordered terms are small enough to neglect in Taylor expression so, first order is used. The trim points are taken into account as a substituted solution to states and control inputs in the nonlinear equations of 6dof to achieve the desired steady state flying.

Trim points in the trimming states are used as the solution of the 6dof nonlinear equations with states x(t) and control inputs u(t) that calculates a desired steady state flight path. Because nonlinear equations cannot be solved numerically, numerical optimization is used to solve them. Trimming is calculated by inserting restrictions into some of the state variables, causing the values to change beneficial to decrease the value to zero. This helps to provide a more precise answer for a desired flying state. This optimization function is to minimize the Jacobian matrix obtained from trimming points.

$$\min_{\mathrm{x_{tim}},u_{trim}} J = \dot{x}_{trim}^{T} W \dot{x}_{trim} + u_{trim}^{T} R u_{trim} \qquad \text{Equation 27}$$

Laplace transformation of the occurred state space and plots of the pole-zero map of the system is used to solve numerical optimization of the equations of motion after the conditions are found by trimming.

Trim Solution and Analysis

The trimmed model of the missile is shown in Figure 19.

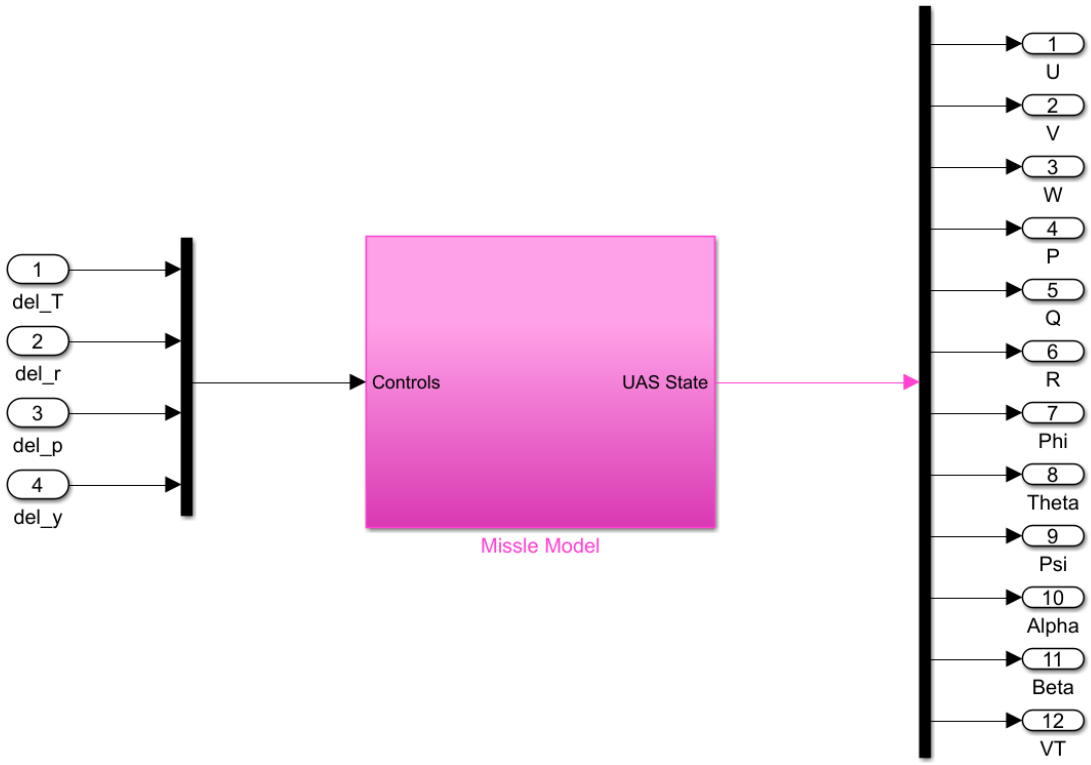

Figure 19 - Missile Model for Trimming

Trim model has only missile model and inputs, outputs and desired states of the system. Constraints were added into the 6dof, Mach number, altitude with additional constraints G_{turn} γ_{Trim}, $\mathrm{Speed_{Trim}}$.

The position states are excluded because the navigation variable has no effect on stability, therefore the integrator of X_e is eliminated from 6dof block to achieve the obtained trim. Also, the $\dot{\theta}_{trim}$ was added into the $\dot{\phi}, \dot{\psi}, \dot{\theta}$ of the Euler angles of the subsystem. Although, the wind velocity model was eliminated since it is assumed that there is no wind.

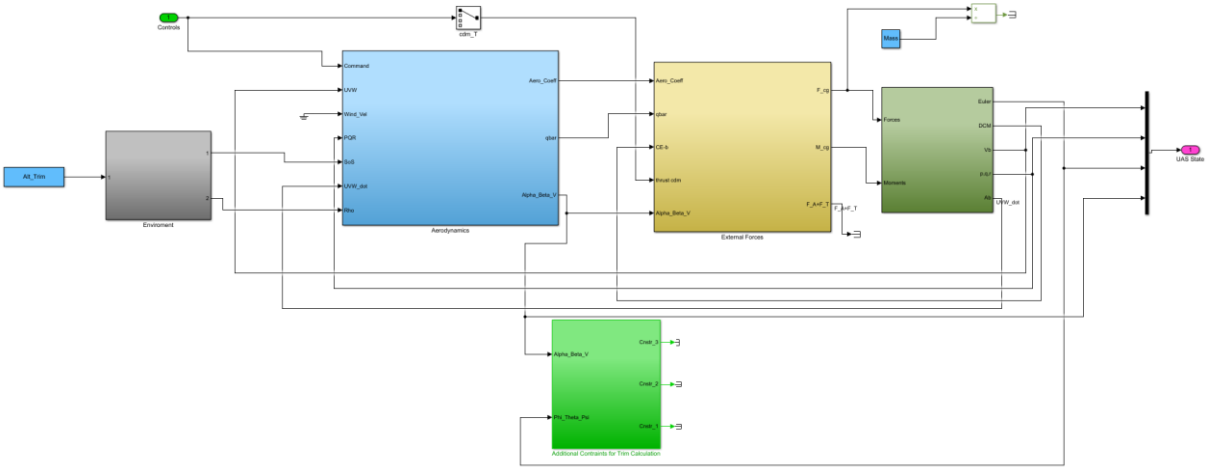

Figure 20 - Trim of Aerodynamics

Trimming constraints were calculated as follows

$$\dot{x}_{c1} = CTC \ (CTC = Coordinate\ Turn\ Constraint$$
$$CTC = \sin\phi - G_{turn}\cos\beta(\sin\alpha\tan\theta + \cos\alpha\cos\phi) = 0 \qquad \text{Equation 28}$$
$$where\ G_{turn} = 0$$

$$ROC = \ \dot{x}_{c2} = \sin\gamma_{trim} - a\sin\theta + b\cos\theta = 0 \qquad \text{Equation 29}$$
$$ROC = Rate\ of\ Climb$$
$$where\ \gamma_{trim} = 0$$
$$a = \cos\alpha\cos\beta$$
$$b = \sin\phi\sin\beta + \cos\phi\sin\alpha\cos\beta$$

$$\dot{x}_{c3} = SC\ (Speed\ Constraint) \qquad \text{Equation 30}$$
$$SC = V_{trim} - V = 0$$

Mach = 0.7 and Altitude = 0

Trimming points were selected by adjusting the model's outputs and states. The model's goal is to determine the trim states and inputs when the Mach number and altitude are set to 0.7 and 0m respectively. The trim function in MATLAB and Simulink is used to solve the problem. Trim inputs, outputs and states are as follows

$$x_{trim} = [u\ \ v\ \ w\ \ p\ \ q\ \ r\ \ \phi\ \ \theta\ \ \psi\ \ \dot{x}_{c1}\ \ \dot{x}_{c2}\ \ \dot{x}_{c3}]^T$$
$$y_{trim} = [\alpha\ \ \beta\ \ V]^T$$
$$u_{trim} = [\delta T\ \ \delta R\ \ \delta P\ \ \delta Y]$$

Results are as follows when Mach number and altitude are 0.7 and 0m respectively.

Table 2 - x_{trim} when Mach = 0.7 at altitude 0m

x_{trim}								
U (m/s)	V (m/s)	W (m/s)	P (deg/s)	Q (deg/s)	R (deg/s)	ϕ (deg/s)	θ (deg/s)	ψ (deg/s)
238.1547	0	3.9578	0	0	0	0	0.9521	0

Table 3 - y_{trim} when Mach = 0.7 at altitude 0m

y_{trim}		
α (deg)	β (deg)	V_T (m/s)
0.9521	0	238.1876

Table 4 - u_{trim} when Mach = 0.7 at altitude 0m

u_{trim}			
δ_T (N)	δ_R (deg)	δ_P (deg)	δ_Y (deg)
216.0976	0	-1.4886	0

Simulink model of verification has verified the trim points.

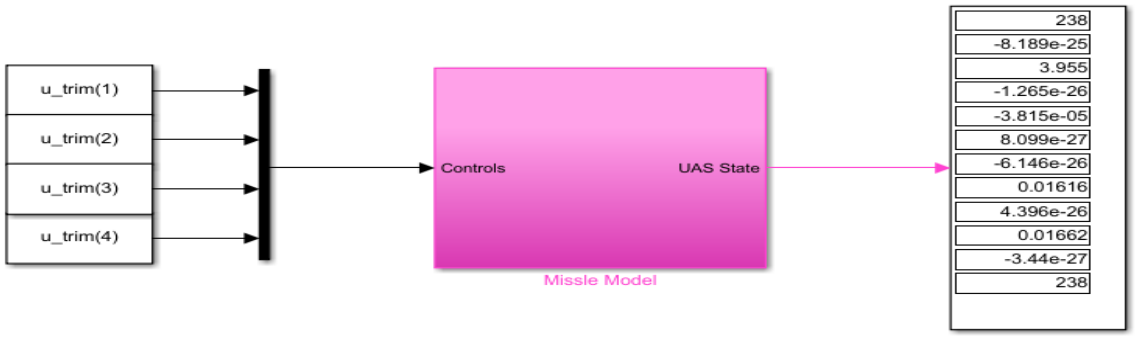

Figure 21 – Verification of Trim Points

Mach = [0.3 0.4 0.6 0.7] and Altitude = 0

Altitude fixed at 0m and Mach values are [0.3 0.4 0.6 0.7] for the trim model to determine trim solution.

Table 5 - x_{trim} with changing Mach at 0m of Altitude

$\boldsymbol{x_{trim}}$									
Mach Values	U (m/s)	V (m/s)	W (m/s)	P (deg/s)	Q (deg/s)	R (deg/s)	ϕ (deg/s)	θ (deg/s)	ψ (deg/s)
0.3	101.4478	0	11.3467	0	0	0	0	6.3819	0
0.4	135.8431	0	8.4751	0	0	0	0	3.57	0
0.6	204.0948	0	5.192	0	0	0	0	1.4572	0
0.7	238.1517	0	3.9578	0	0	0	0	0.9521	0

Table 6 - y_{trim} with changing Mach at 0m of Altitude

$\boldsymbol{y_{trim}}$			
Mach Values	α (deg)	β (deg)	V_T (m/s)
0.3	6.3819	0	102.0804
0.4	3.57	0	136.1072
0.6	1.4572	0	204.1608
0.7	0.9521	0	238.1876

Table 7 - u_{trim} with changing Mach at 0m of Altitude

$\boldsymbol{u_{trim}}$				
Mach Values	δ_T (N)	δ_R (deg)	δ_P (deg)	δ_Y (deg)
0.3	87.9640	0	-9.8236	0
0.4	97.5514	0	-5.55	0
0.6	163.4163	0	-2.2769	0
0.7	216.0976	0	-1.4886	0

Missile velocity increases as the Mach number increases when altitude does not change. Speed of sound does not change since the altitude is the same and Mach number is a function of speed of sound and velocity of the vehicle. This explains the increase in both Mach number and the missile velocity. Additionally, lift force on the missile increases as velocity increases so, change in elevation angle is positive. Thus, negative pitch moment occurs.

Mach = 0.7 and Altitude = [0 1000 2000 3000 4000]

In this part, Mach number was set at 0.7 and altitude varies. Altitude values are [0 1000 2000 3000 4000].

Table 8 - x_{trim} with constant Mach and Changing Altitude

x_trim									
Altitude (m)	U (m/s)	V (m/s)	W (m/s)	P (deg/s)	Q (deg/s)	R (deg/s)	ϕ (deg/s)	θ (deg/s)	ψ (deg/s)
0	238.1547	0	3.9578	0	0	0	0	0.9521	0
1000	235.443	0	4.4112	0	0	0	0	1.0734	0
2000	232.7006	0	4.9290	0	0	0	0	1.2134	0
3000	229.9208	0	5.522	0	0	0	0	1.3758	0
4000	227.1031	0	6.2035	0	0	0	0	1.5647	0

Table 9 - y_{trim} with constant Mach and Changing Altitude

y_trim			
Altitude (m)	α (deg)	β (deg)	VT (m/s)
0	0.9521	0	238.1876
1000	1.0734	0	235.4856
2000	1.2134	0	232.7528
3000	1.3758	0	229.9871
4000	1.5647	0	227.1878

Table 10 - u_{trim} with constant Mach and Changing Altitude

u_{trim}				
Altitude (m)	δ_T (N)	δ_R (deg)	δ_P (deg)	δ_Y (deg)
0	216.0976	0	-1.4886	0
1000	193.3617	0	-1.6782	0
2000	172.9839	0	-1.8972	0
3000	154.8275	0	-2.1510	0
4000	138.7715	0	-2.4463	0

The missile velocity decreases as altitude increases because Mach number is constant for every altitude and speed of sound increases as altitude increases. Additionally, there is a negative elevation force on the missile because velocity decreases and lift force acting on the missile decreases according to this decrease. Increase in elevation causes a positive pitch moment on the missile.

Numerical Linearization

Model has only the missile model alike trimming part because unnecessary subsystems and sources of disturbance were eliminated for numerical linearization. Numerical linearization was performed around the first trim condition according to question. Finally, a full linear model occurred to decoupled into lateral and longitudinal motions. Also, X_e, PQR, Euler angles and UVW are selected states.

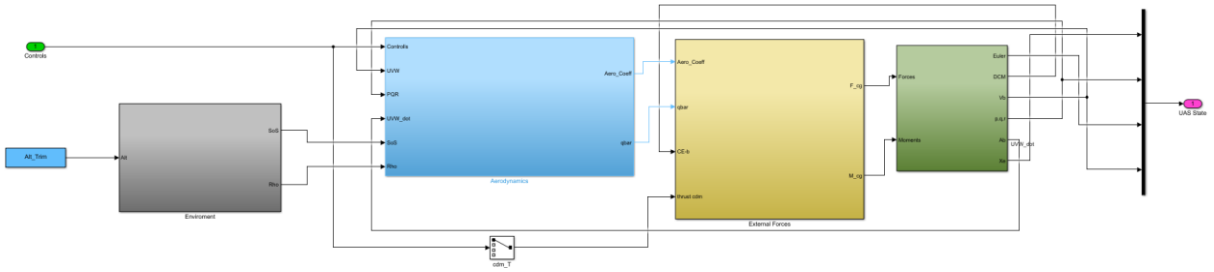

Figure 22 - Missile Model for Linearization

This model has two different modes of motion:

Longitudinal Motion

Longitudinal motion has been chosen to obtain the inputs and the states of the following:

$$x_{long} = [u \;\; w \;\; q \;\; \theta]^T$$

$$u_{long} = [\delta P]^T$$

$$\dot{x}_{long} = A_{long}x_{long} + B_{long}u_{long}$$

$$\begin{bmatrix}\dot{u}\\ \dot{w}\\ \dot{q}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}-0.0607 & 0.0039 & -3.9578 & -9.8086\\ -0.0561 & -3.5066 & 238.1547 & -0.1630\\ 0.0450 & -2.7205 & -0.2472 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}u\\ w\\ q\\ \theta\end{bmatrix} + \begin{bmatrix}6.1451\\ -155.0421\\ -413.4210\\ 0\end{bmatrix}[\delta_{elevator}]$$

Lateral Motion

Lateral motion has been chosen to obtain the inputs and the states of the following:

$$x_{lat} = [v\ \ p\ \ r\ \ \phi]^T$$

$$u_{lat} = [\delta P\ \ \delta Y]^T$$

$$\dot{x}_{lat} = A_{lat}x_{lat} + B_{lat}u_{lat}$$

$$\begin{bmatrix}\dot{v}\\ \dot{p}\\ \dot{r}\\ \dot{\phi}\end{bmatrix} = \begin{bmatrix}-3.5310 & 3.9578 & -238.1547 & 9.8086\\ -0.0064 & -9.8464 & 0 & 0\\ 0.0161 & 0 & -0.2472 & 0\\ 0 & 1 & 0.0166 & 0\end{bmatrix}\begin{bmatrix}v\\ p\\ r\\ \phi\end{bmatrix} + \begin{bmatrix}0 & 155.0421\\ 0 & 0\\ 0 & -7.2156\\ 0 & 0\end{bmatrix}\begin{bmatrix}\delta_{elevator}\\ \delta_{rudder}\end{bmatrix}$$

Stability Analysis

Linearization calculated the state-space equation of the system. It was transformed to the transfer function to govern the stability of the system. It can be understood from the pole zero map the stability of the system.

Longitudinal Motion

The poles of the longitudinal motion are as follows

Table 11 - Poles of Longitude Motion

Modes	Poles	Damping Ratio	Natural Frequency (rad/s)	Time (s)
Phugoid Mode	$-3.12x10^{-2} + 6.07x10^{-2i}$	$4.58x10^{-1}$	$6.82x10^{-2}$	$3.2x10^{1}$
	$-3.12x10^{-2} - 6.07x10^{-2i}$			
Short Period	$-1.88 + 2.54x10^{i}$	$7.36x10^{-2}$	$2.55x10^{1}$	$5.33x10^{-1}$
	$-1.88 - 2.54x10^{i}$			

There are no positive poles which means the system is stable. Also, phugoid mode's poles are closer to the zero so, it is more stable than short period. However, phugoid mode have a tendency to to overshoot over the time more than short period because it has higher damping ratio. Pole zero map shown in Figure 23 supports the stability of the mode.

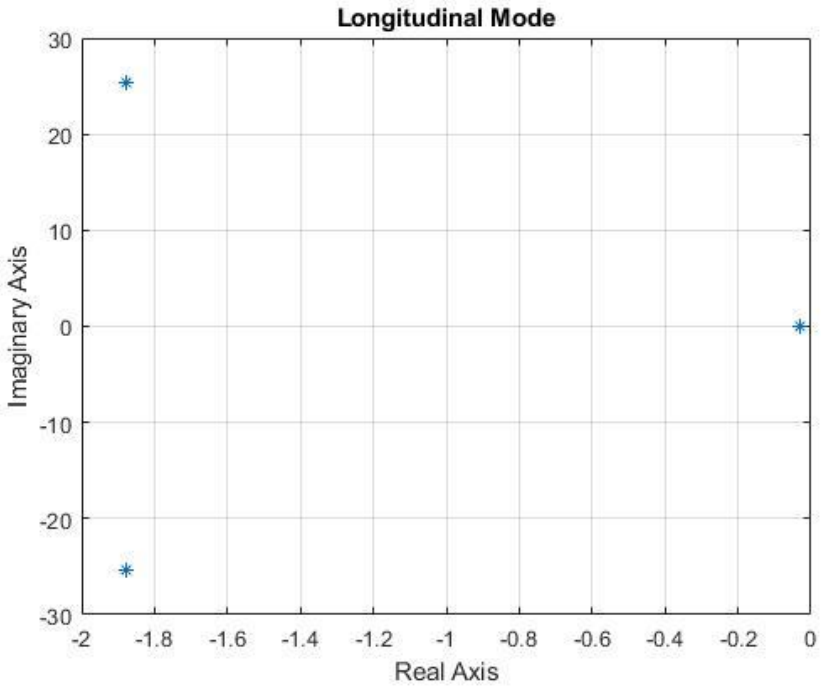

Figure 23 - Longitude Motion

Lateral Motion

The poles of the lateral motion are as follows:

Table 12 - Poles of Lateral Motion

Modes	Poles	Damping Ratio	Natural Frequency (rad/s)	Time (s)
Spiral	$2.22x10^{-4}$	-1	$2.22x10^{-4}$	$-4.50x10^{3}$
Dutch	$-1.89 + 1.07x10^{0i}$	$8.71x10^{-1}$	2.17	
Roll	$-1.89 + 1.07x10^{0i}$			
Roll	-9.84	1	9.84	$1.02x10^{-1}$

Poles with complex numbers are related to the Dutch roll, pole with positive real number associates with the Spiral mode and pole with negative real number is Roll mode. Dutch roll is stable because poles of the system are on the left half plane of the map. Although, spiral mode is not stable because its pole is positive.

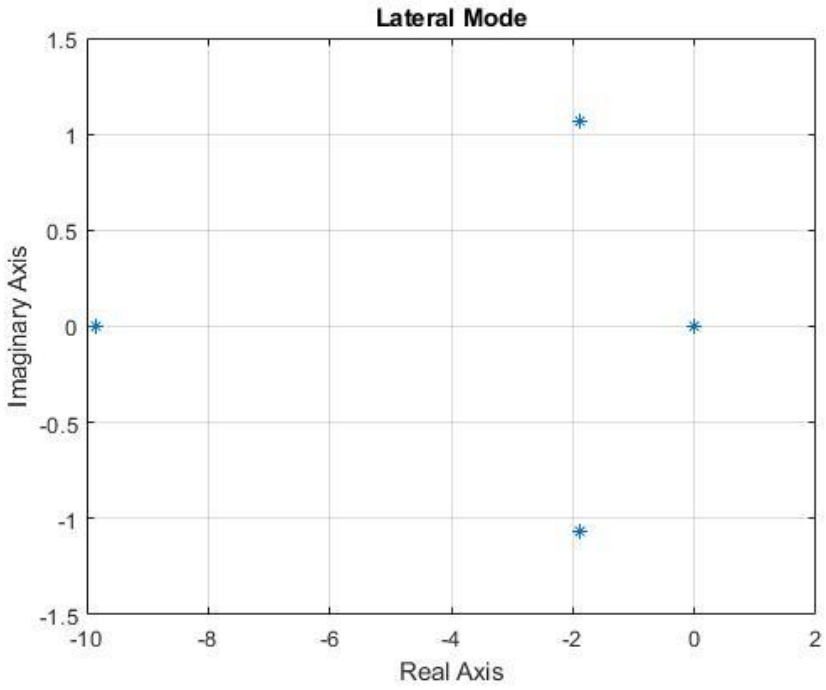

Figure 24 - Lateral Motion

Changing Centre of Gravity

The value of the CG was changed to be 10% backwards and forwards of the nominal value and this was done to compare the stability of the system.

When it moves forward by 10%, system stays stable. It is shown in the Figures 25 and 26.

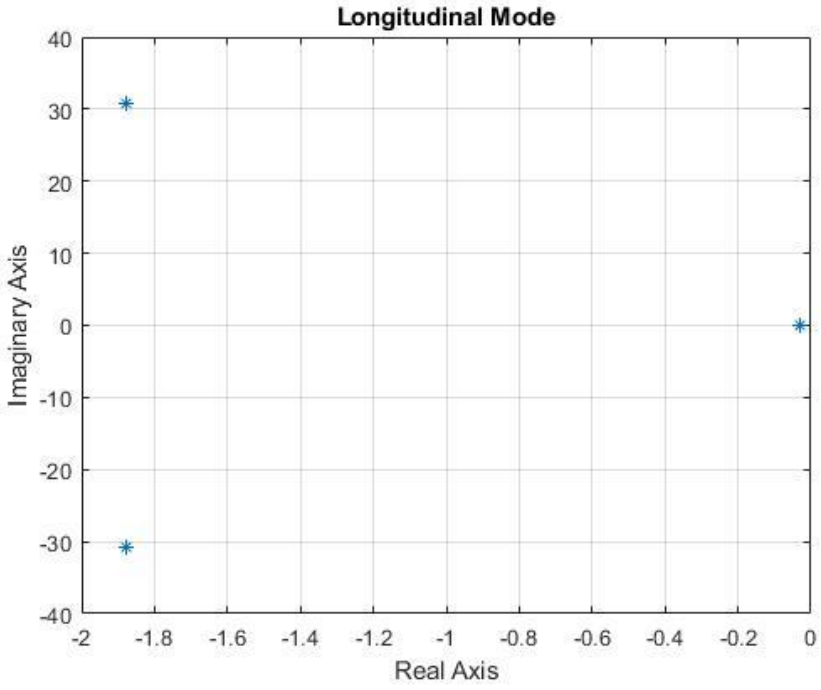

Figure 25 - Longitude Motion with 10% Forward CG

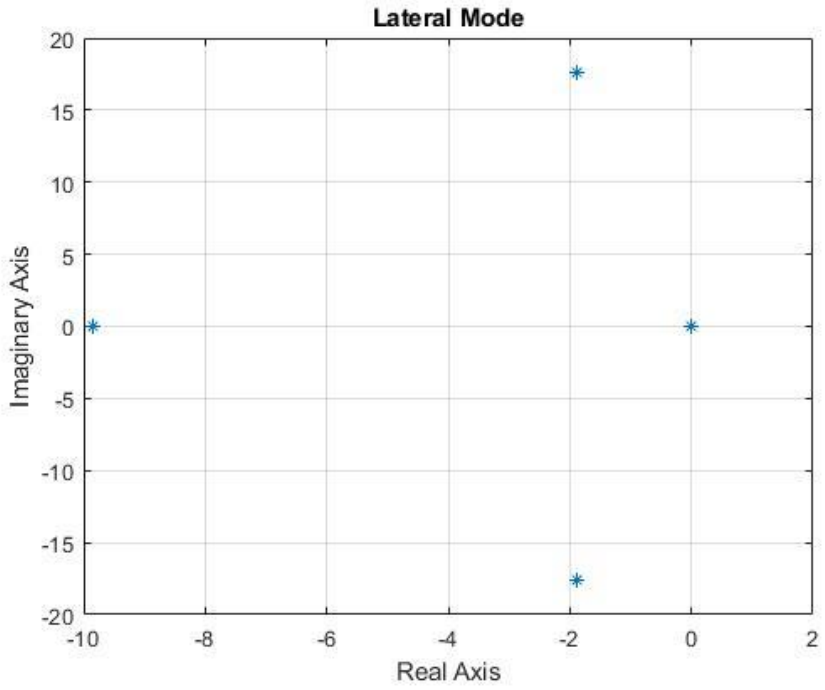

Figure 26 - Lateral Motion with 10% Forward CG

System does not remain stable when CG moves backward by 10% because some poles of lateral motion moved the right half plane of the pole zero map. Figures 27 and 28 shows the poles of longitude and lateral motions' poles respectively.

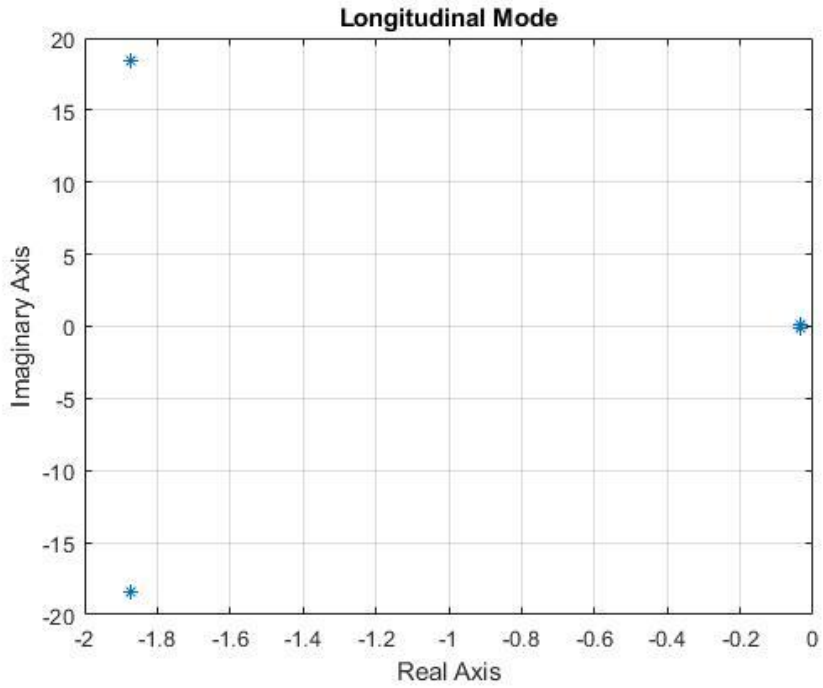

Figure 27 - Longitude Motion with 10% Backward CG

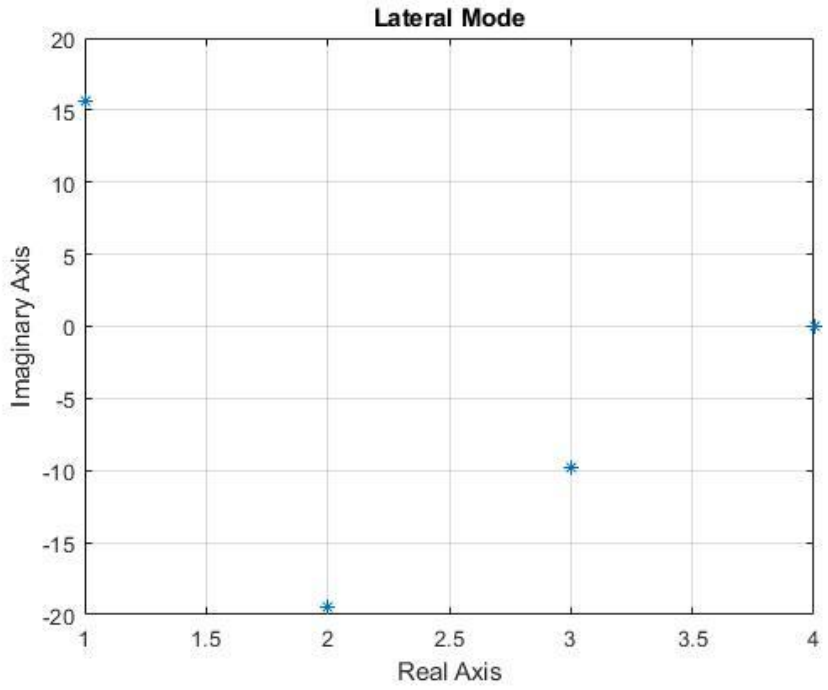

Figure 28 - Lateral Motion with 10% Backward CG

Part 3 – Implementation of Control Laws

Implementation of control laws has been done by adding the autopilot model. Autopilot model is the controller of the system which uses the gains:

Table 13 - Gains of Autopilot

Gain	K_ϕ	K_p	K_{dc}	K_a	W_i	K_r
Value	20	0.01	0.8443	0.0270	-6.5208	0.0777

u_{trim} values when Mach number is 0.7 at altitude of 0m were used to run this simulation. $a_{y,cmd}$, $a_{z,cmd}$ and roll command were set as 10 m/s, 10 m/s and 0 respectively. Graphs of $a_{z,cmd}$ and $a_{y,cmd}$ are shown in Figures 29 and 30 respectively.

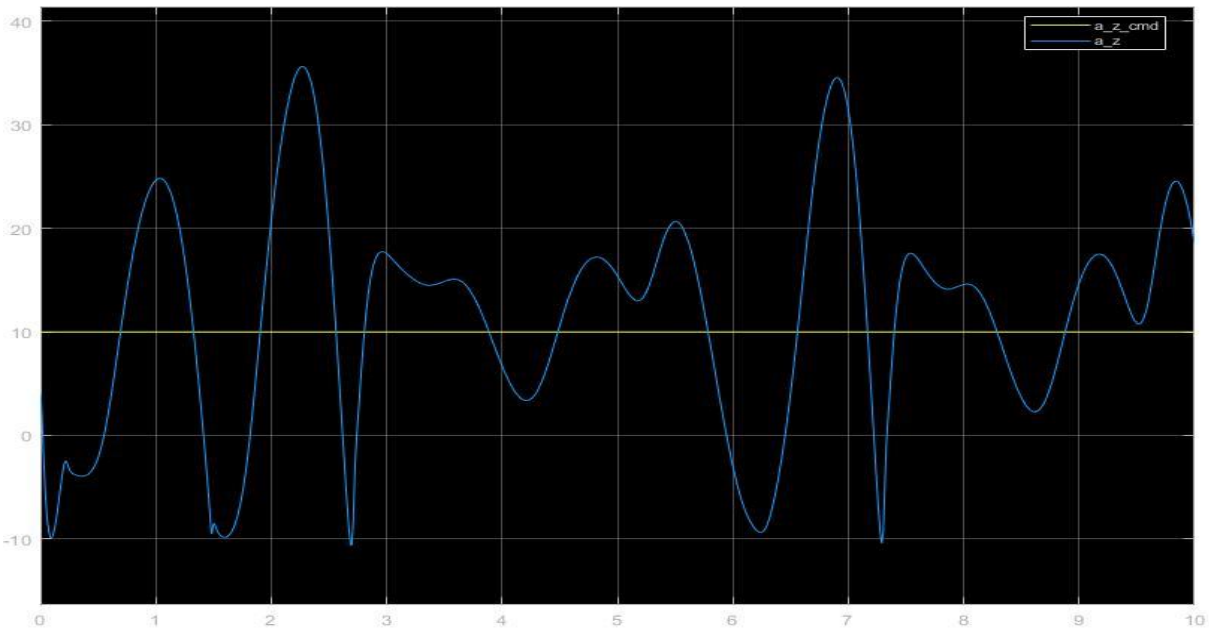

Figure 29 - a_z Overshoot

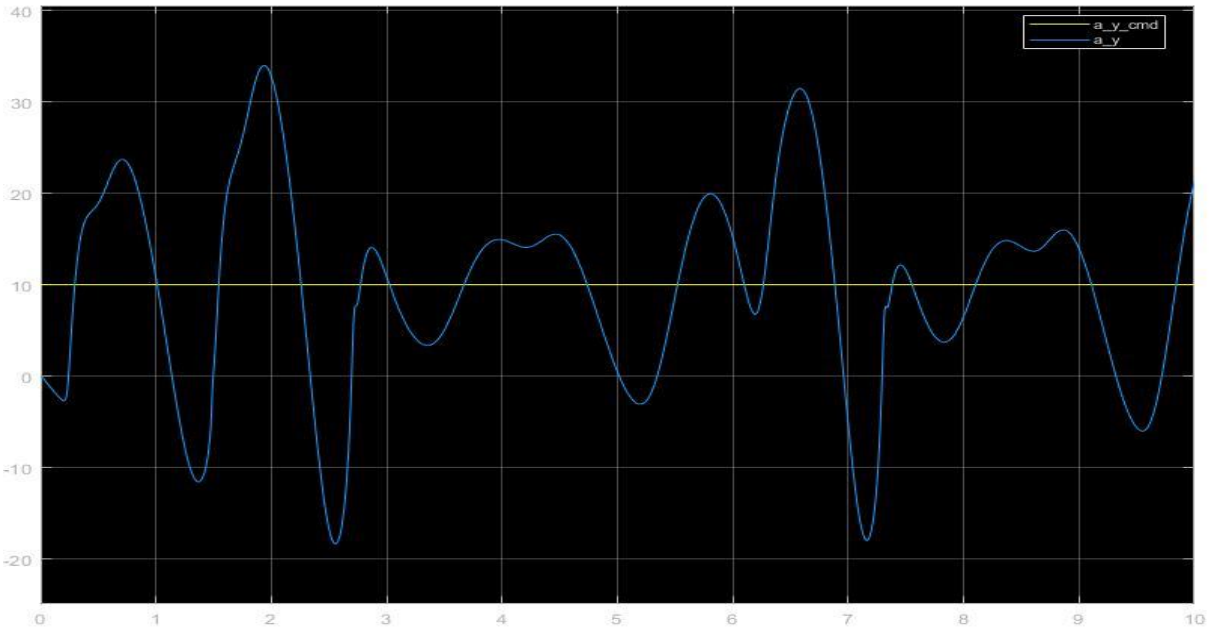

Figure 30 - a_y Overshoot

Figures 29 and 30 showed that oscillation and overshoot were generated by acceleration commands on z and y axes. It is easy to demonstrate that the system is not stable. Thus, system had to be optimized by changing the gain values. System was tuned by using tunning tools of Simulink. The new gain values are:

Table 14 - Tuned Gians of Autopilot

Gain	K_A	K_{DC}	K_p	K_R	K_ϕ	W_i
Values	0.87	0.96	0.01	0.0254	20	-0.11

After tunning the gain values, acceleration commands changed. New acceleration commands are shown in Figure 31 and 32. As it can be seen in the figures, system does not oscillate or overshoot so, it is stable now.

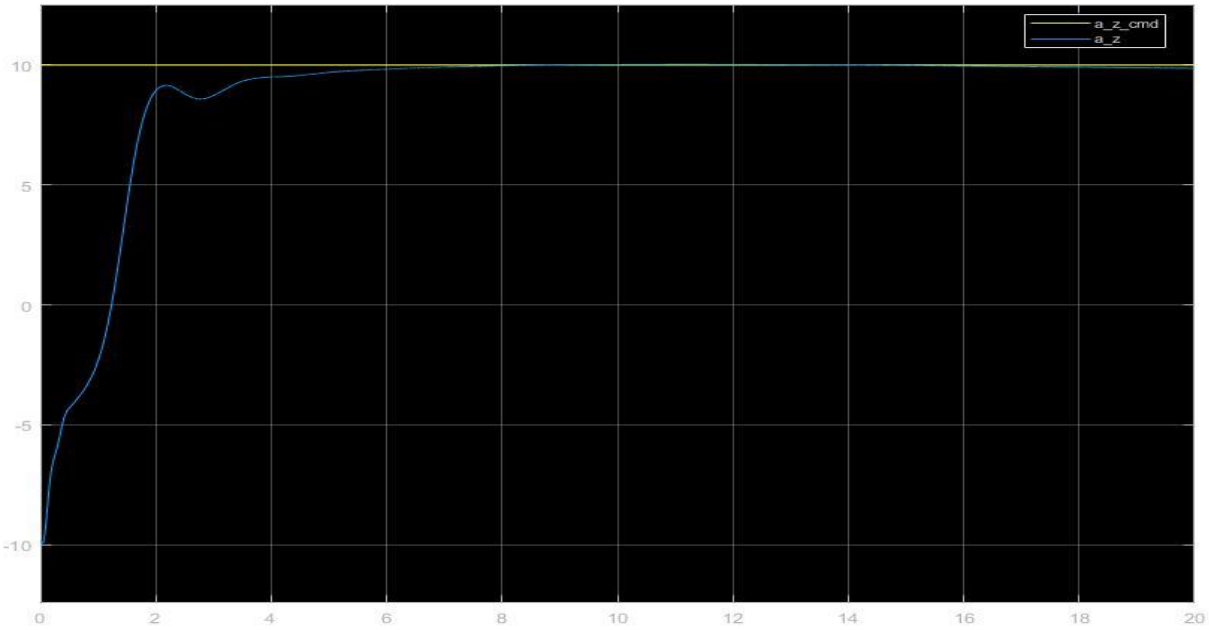

Figure 31 – Optimized a_z

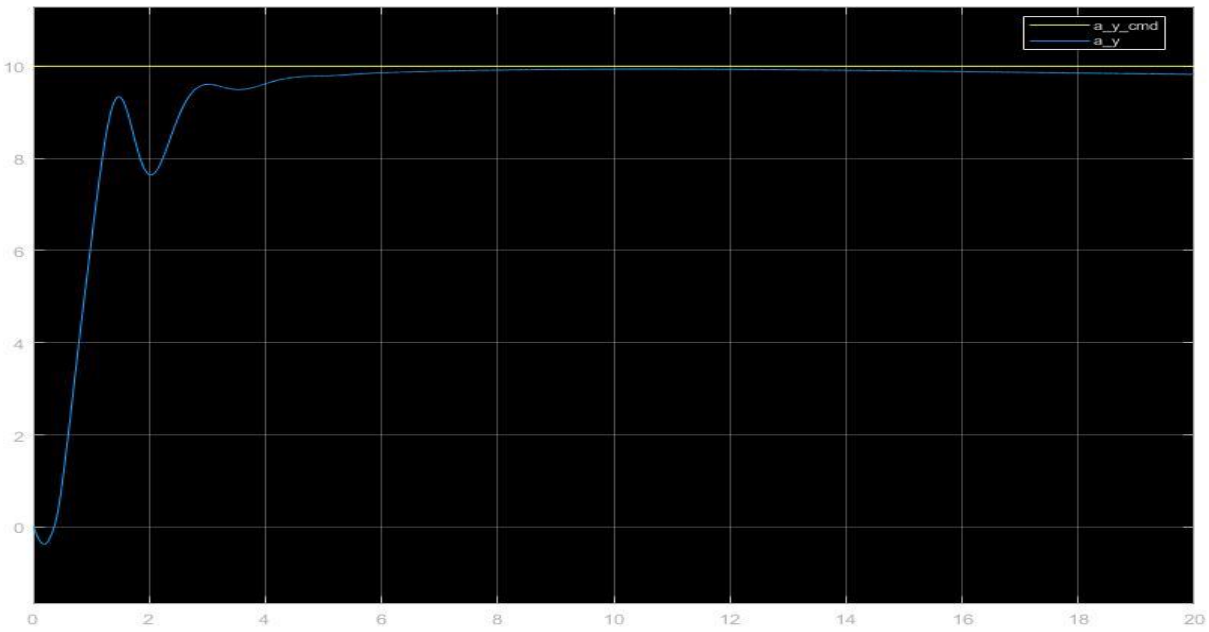

Figure 32 - Optimized a_y