Step 2 of the Project: Pitch-Axis Stability Augmentation System Design

A. Stability Axes Model

To design a controller, we first need to linearize aircraft model. Recall from lectures that we prefer to linearize the stability-axes model of aircraft. Therefore, follow the following steps:

- Create three new MATLAB files. Name these files as
  - **`RunAircraftStabilityAxesModel.m`** (similar to **`RunAircraftModel.m`**)
  - **`StabilityAxesModel.m`** (similar to **`FlatEarthEquations.m`**)
  - **`AircraftStabilityAxesModel.slx`** (similar to **`AircraftModel.slx`**)

Plot the trajectory of state-variables. Also, the original flat-Earth equations with the same initial conditions and same simulation time. By comparing the two sets of plots, you can confirm if your stability-axes model is correct or not. Include both sets of plots in your project report.

ANS) The two plots of flat earth equations and stability axix models. Both are ideantical indiacting that our stability axix models is correct as shown in **Figure 1&2.**

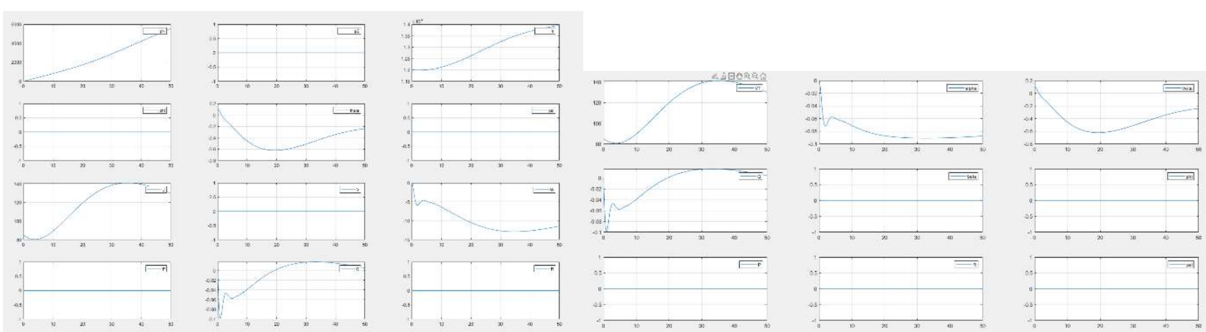

Figure 1&2: The first plot represnets the statevairables for the flat earth and the second plot represnts the state variables for stability axix models

Obtain the longitudinal and lateral A and B matrices and include them in your report. Then, report the eigen values of the A matrix for both longitudinal and lateral dynamics and determine the which eigenvalues correspond to which aircraft dynamical mode. The aircraft is expected to be stable in all its modes (as it can be confirmed from the simulation of flat-Earth equations). If you have some signs of instability, then it is likely that there is something wrong with your calculations! Remember that the beauty of eigenvalues is that they predict the nonlinear simulation behavior at a given operating point. It is worth mentioning that the above initial conditions are very close to aircraft trimmed conditions.

ANS) Figure 3 shows the entire A matrix, the method I used for decopling is by spliting the A matrix and no accounting for the terms in the other dynamic model (logitudianl or latral). I split the A matrix equally and for most of the part its correct yet some inaccureties are resulted which shows why my eigen values resultied one unsatable value.

```
A_implicit =

   -0.0361    4.9326   -9.7610         0   -0.1567    0.9794         0         0         0
   -0.0028   -0.7070         0    0.9672   -0.0005         0   -0.1003         0         0
         0         0         0    1.0000         0         0         0         0         0
   -0.0127   -2.8617         0   -1.1068    0.0300         0         0         0         0
    0.0002   -0.0115    0.0115         0    0.1621    0.1148         0   -1.0000         0
         0         0         0         0         0         0    1.0000         0         0
   -0.0024   -0.0204         0         0   -1.9615         0   -1.3460    0.5842         0
    0.0012   -0.2388         0         0    0.4935         0    0.0554   -0.5533         0
         0         0         0         0         0         0         0    1.0000         0
```

Figure 3: The implicit A matrix of the entire model

```
B_implicit =

    9.7610    9.7610         0         0   -0.0000
         0         0         0   -0.0865         0
         0         0         0         0         0
    0.3924    0.3924         0   -2.9181   -0.0024
   -0.0115   -0.0115         0         0   -0.0271
         0         0         0         0         0
   -0.0407    0.0407   -0.9486         0    0.2901
   -0.7804    0.7804   -0.0199         0   -0.3763
         0         0         0         0         0
```

Figure 4: The implicit B matrix of the entire model

```
>> A_Long = A_implicit(1:4,1:4)

A_Long =

   -0.0361    4.9326   -9.7610         0
   -0.0028   -0.7070         0    0.9672
         0         0         0    1.0000
   -0.0127   -2.8617         0   -1.1068
```

```
>> A_Latral = A_implicit(5:8,5:8)

A_Latral =

    0.1621    0.1148         0   -1.0000
         0         0    1.0000         0
   -1.9615         0   -1.3460    0.5842
    0.4935         0    0.0554   -0.5533
```

Figure 5: The implicit A matrix of the logitudinal and latral

```
>> B_long = [B_implicit(1:4,1) B_implicit(1:4,2) B_implicit(1:4,4)]

B_long =

    9.7610    9.7610         0
         0         0   -0.0865
         0         0         0
    0.3924    0.3924   -2.9181
```

Figure 6: The implicit B matrix of the logitudinal

```
>> B_latral = [B_implicit(5:8,3) B_implicit(5:8,5)]

B_latral =

         0   -0.0271
         0         0
   -0.9486    0.2901
   -0.0199   -0.3763
```

Figure 7: The implicit B matrix of the latral

```
>> eig(A_Long)

ans =

  -0.9131 + 1.6599i
  -0.9131 - 1.6599i
  -0.0654 + 0.0000i
   0.0418 + 0.0000i
```

```
>> eig(A_Latral)

ans =

  -1.4020 + 0.0000i
  -0.0860 + 0.6266i
  -0.0860 - 0.6266i
  -0.1632 + 0.0000i
```

Longitudinal			Lateral		
$\dot{V}_T$	-0.9131 + 1.6599i	short-period mode	$\dot{\beta}$	-1.4020 + 0.0000i	roll subsidence mode
$\dot{\alpha}$	-0.9131 - 1.6599i	short-period mode	$\dot{\phi}$	-0.0860 + 0.6266i	dutch roll mode
$\dot{\theta}$	-0.0654 + 0.0000i	phugoid mode	$\dot{P}$	-0.0860 - 0.6266i	dutch roll mode
$\dot{Q}$	0.0418 + 0.0000i	phugoid mode	$\dot{R}$	-0.1632 + 0.0000i	spiral mode

Design a pitch-axis SAS. The steps need to be taken are similar to the steps given in Example 4.4-1 of the textbook or the L5 lecture notes (November 16, 2021).

- Consider the following actuator dynamics for the elevator.

$$G_A(s) = \frac{1}{15s+1}$$

Note that the actuator dynamics is sluggish. This choice was deliberate to make the control problem interesting. However, we assume that all states are measured appropriately and there is no need for noise filter. With the above information, create an augmented system representation. The dimension of the augmented system must be 5, because there is no filer dynamics included.

Now, simulate the stability-axes equations model, and the one with the controller, both for equal simulation time, using the above initial conditions. Plot their pitch trajectories on the same figure, and comment how the pitch response has improved. Include the results in your report. Note that you will need to add the actuator dynamics $G_A(s)$ to both of your Simulink models.

For the controller design, I have used two methods to design the controller. The first one uses the implicit A matrix of the longitudinal matrix and is save in *Project 2 step 2> Part2>PitchAxisControlDesign. The second method uses the entire implicit matrix of the model and is saved in Project 2 step 2> Part2> controllerdesingbig.*

I have used the root locus of the second method to come up with the following gain values. Figure 8 shows the phugoid poles in a alpha/Ue root locus. K_alpha is assumed to be 25. K_q = 2.62.

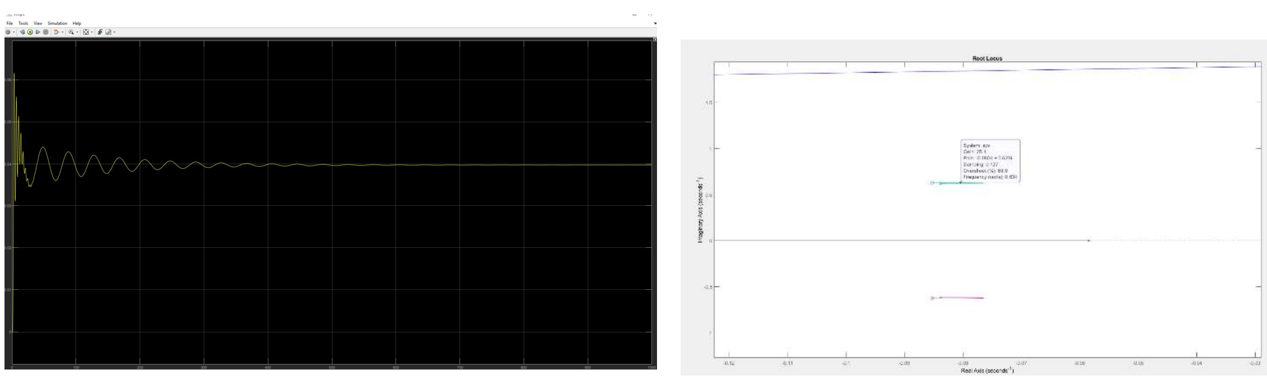

Figure 8: root locus of alpha/Ue

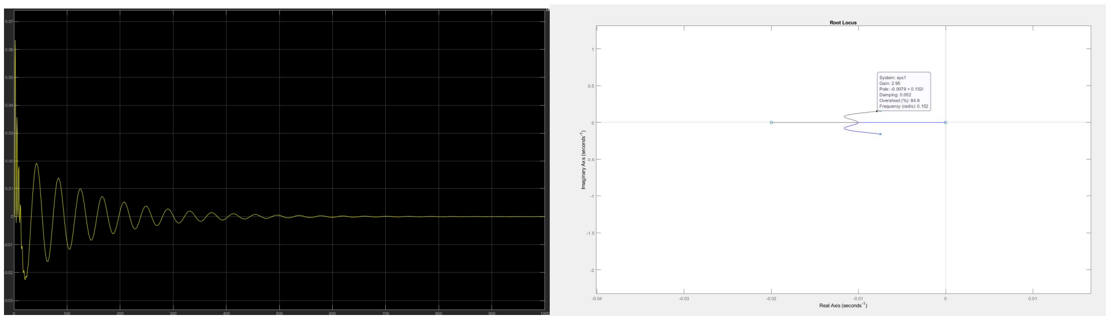

Figure 9: root locus of q/Ue

I have created two files for the closed and open loop system and outputted the results. I couldn't find the appropriate gain value to stabilize the entire model since the K_alpha is not tacking the phugoid poles in my case. So in the model I had the K_alpha gain negative to stabilize the model and the results were promising, if K_alpha is positive then it would be over damped.

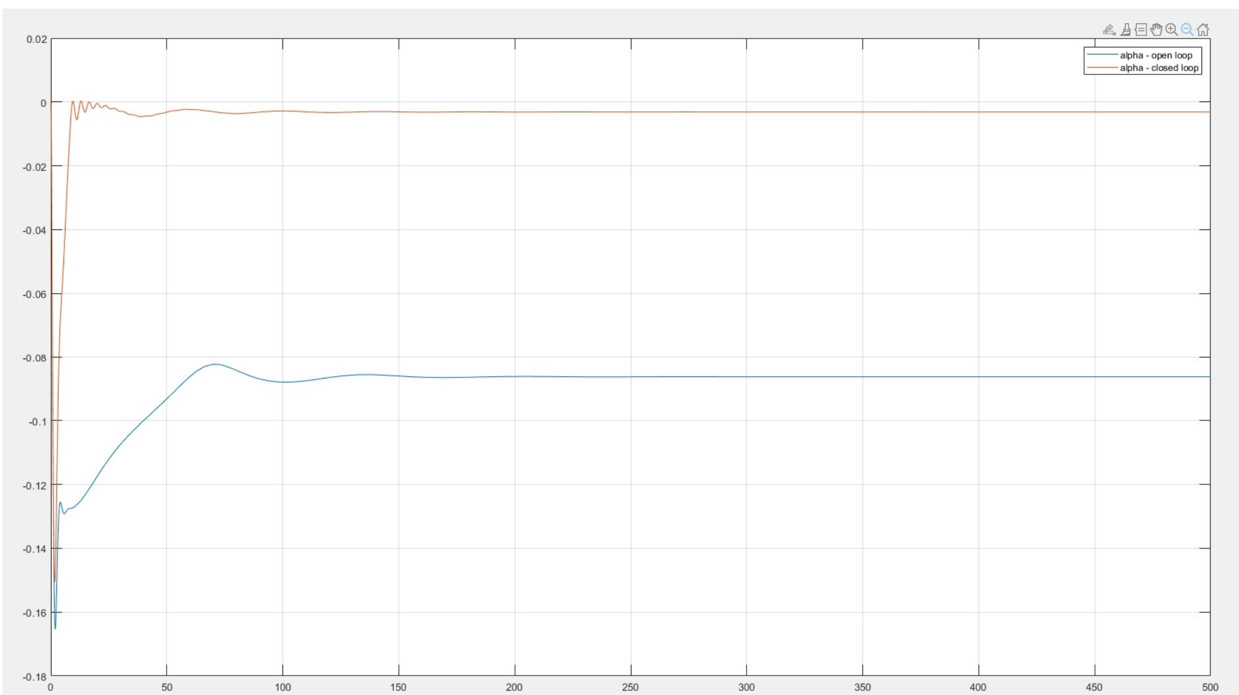

Figure 10: alpha open loop vs closed loop.