

Advanced Aircraft Control Systems With MATLAB / Simulink

Prof. Dipak K. Giri

Department of Aerospace Engineering

Indian Institute of Technology Kanpur

Lecture 23

Stability Augmentation System for full longitudinal dynamics

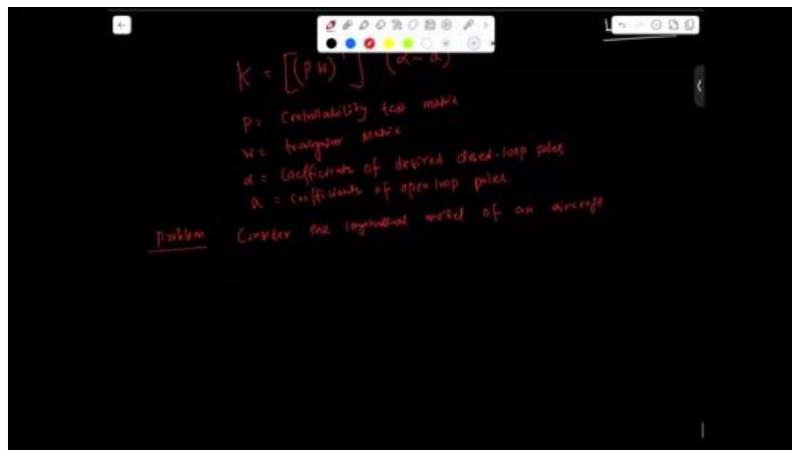
Hello everyone, this lecture will be concerning the full longitudinal motion dynamics and how we can design the stability augmentation system using the Bass-Gura method. In the Bass-Gura method, we have briefed in the last lecture, we can find the control matrix

$$K = [(PW)^T]^{-1}[\alpha - a]$$

So, here P is the, let me write, P is the controllability test matrix W triangular matrix which will be formed using the polynomial of the A matrix system matrix and α equal to coefficients of desired closed-loop poles and A coefficients open-loop poles. The problem is the example we are going to solve is how we can design the stability augmentation system for the longitudinal model of an aircraft problem. Consider the longitudinal model of an aircraft. Here we have the derivative one of the state vector

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -0.01 & 0.1 & 0 & -32.2 \\ -0.4 & -0.8 & 180 & 0 \\ 0 & -0.003 & -0.5 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta \theta \end{bmatrix} + \begin{bmatrix} 0 \\ -10 \\ -2.8 \\ 0 \end{bmatrix} \Delta \delta_e$$

(Refer Slide Time: 02:26)



Now, to design the stability augmentation system using Bass-Gura, this is a numerical method Bass-Gura method such that the augmented aircraft, augmented means the closed-loop system, has the following short and long period. characteristics as for the short period,

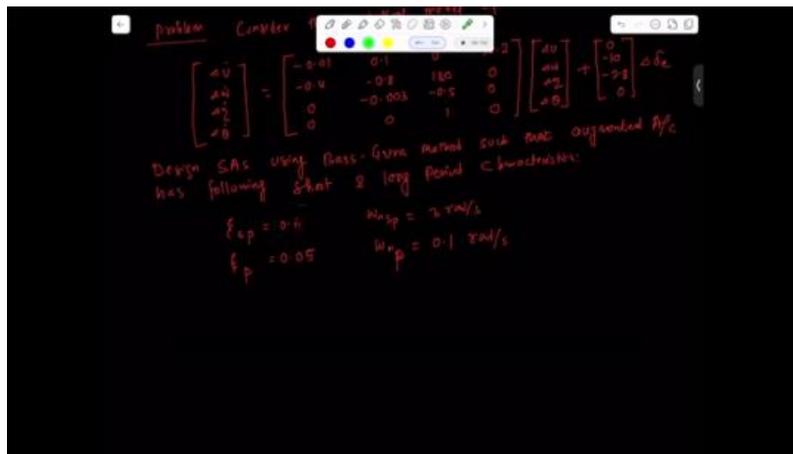
$$\xi_{sp} = 0.6 \quad \omega_{n,sp} = 3 \text{ rad/s}$$

And for a long period, the damping ratio

$$\xi_p = 0.05 \quad \omega_{n,p} = 0.1 \text{ rad/s}$$

P for phugoid, also known as long-period motion or sometimes called phugoid motion, It is clear from the given values that the damping ratio for the long period is quite less compared to the short-period motion. That is why it takes more time to get stabilized. Now, we will be using the Bass-Gura method to solve this problem. This Bass-Gura method explains how to find the control matrix K. Here, we will go step by step.

(Refer Slide Time: 05:45)



We will find the open-loop characteristic equation. So, we can find the system's characteristic equation. The characteristic equation we can find

$$|sI - A| = 0$$

$$\begin{vmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & s \end{vmatrix} - \begin{vmatrix} -0.01 & 0.1 & 0 & -32.2 \\ -0.4 & -0.8 & 180 & 0 \\ 0 & -0.003 & -0.5 & 0 \\ 0 & 0 & 1 & 0 \end{vmatrix} = 0$$

One thing I should mention is that this data, which you are considering, can be used with the poly function, and if you enter this, you can get the equation.

$$p = \text{poly}(p)$$

$$p = [1 \quad 1.31 \quad 0.99 \quad 0.029 \quad 0.038]$$

and if you use these coefficients of the polynomial with the polynomial equation, you get

$$s^4 + 1.31s^3 + 0.99s^2 + 0.029s + 0.038 = 0$$

Now, we could write it in the form of a's, that is We can write

$$s^4 + a_1s^3 + a_2s^2 + a_3s + a_4 = 0$$

and from this, we can form the a vector, which is

$$a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 1.31 \\ 0.99 \\ 0.029 \\ 0.038 \end{bmatrix}$$

and also, we can find the open-loop poles.

(Refer Slide Time: 07:46)

$s^4 + 1.31s^3 + 0.99s^2 + 0.029s + 0.038 = 0$
 Sol: The system CE.
 $|sI - A| = 0$
 $\begin{vmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & s \end{vmatrix} - \begin{vmatrix} -0.01 & 0.1 & 0 & -32.1 \\ -0.4 & -0.8 & 100 & 0 \\ 0 & -0.003 & -0.5 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} = 0$

(Refer Slide Time: 09:32)

$p = [1 \quad \dots]$
 $s^4 + 1.31s^3 + 0.97s^2 + 0.027s + 0.038 = 0$
 i.e., $s^4 + a_1s^3 + a_2s^2 + a_3s + a_4 = 0$
 $\Rightarrow a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} \dots \\ \dots \\ \dots \\ \dots \end{bmatrix}$

$$r = \text{roots}(p)$$

$$s_{1,2} = -0.6 \pm 0.7i$$

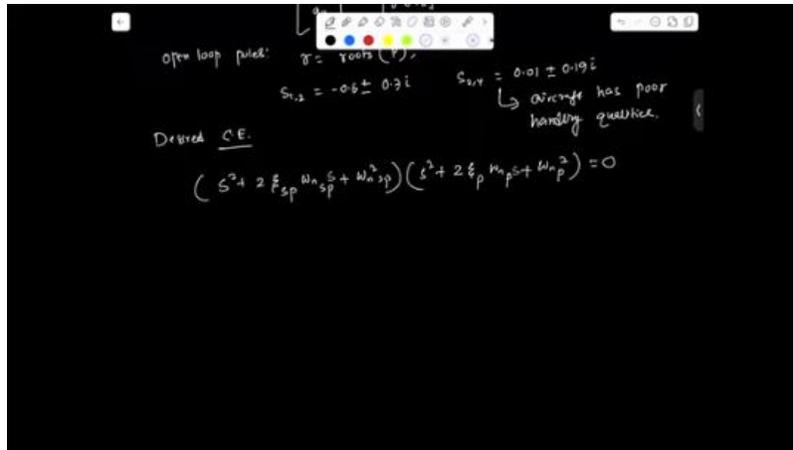
$$s_{3,4} = 0.01 \pm 0.19i$$

Okay, this is quite obvious from these poles. This part is basically the short period, and this is basically the long period motion, right? So, these are the poles of the original system A matrix. Now, we will form the desired characteristic equation, desired characteristic. Okay, one thing if you notice here carefully, if you notice here, the poles of the original system, one is positive. So, this actually, you can see the system handling quality. This pole, we can say for this pole, the aircraft has four handling qualities. So, we have to design a control system which can improve these poor handling qualities. The desired characteristic equation you can form, you can find

$$(s^2 + 2\xi_{sp}\omega_{n,sp}s + \omega_{n,sp}^2)(s^2 + 2\xi_p\omega_{n,p}s + \omega_{n,p}^2) = 0$$

And these values are given to us. If you go back to the problems, these values are given, right. So, if we substitute these values, we can find the desired characteristic equation that the control system should follow. So, substituting the values of zeta and omega n in this equation

(Refer Slide Time: 12:32)



$$(s^2 + 3.6s + 9)(s^2 + 0.01s + 0.01) = 0$$

$$s^4 + 3.61s^3 + 9.05s^2 + 0.126s + 0.09 = 0$$

$$s^4 + \alpha_1s^3 + \alpha_2s^2 + \alpha_3s + \alpha_4 = 0$$

$$\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} = \begin{bmatrix} 3.61 \\ 9.05 \\ 0.126 \\ 0.09 \end{bmatrix}$$

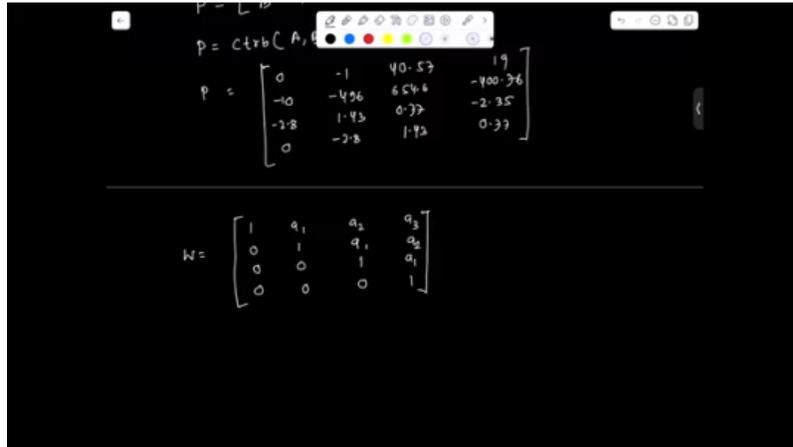
So, now we have to find P and W. Once you find that, then we can use the main equation to find the So, now we have to find the P matrix or controllability test matrix. So, you can write the controllability test matrix as P equal to [B, AB, A²B, A³B]. And if you also use the MATLAB command P = ctrb(A, B) and enter it, we have the P matrix as

$$P = \begin{bmatrix} 0 & -1 & 40.57 & 19 \\ -10 & -496 & 654.6 & -400.76 \\ -2.8 & 1.43 & 0.77 & -2.35 \\ 0 & -2.8 & 1.93 & 0.77 \end{bmatrix}$$

This is the controllability test matrix. So, P has been found. Now we have to find W. W can be formed using the terms A, which you have found before, the polynomial of A matrix:

$$W = \begin{bmatrix} 1 & a_1 & a_2 & a_3 \\ 0 & 1 & a_1 & a_2 \\ 0 & 0 & 1 & a_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1.31 & 0.99 & 0.029 \\ 0 & 1 & 1.31 & 0.99 \\ 0 & 0 & 1 & 1.31 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(Refer Slide Time: 17:18)



So, we have found the W matrix. Now, another important term in the Bass-Gura method, if you notice this term, to find this term is also very tedious. So, let us calculate this. So, now we will find it. If you use MATLAB, you can easily find it, or if you know, in MATLAB, you can easily put the parameter $P * W$ transpose but we are directly writing here the value is found to be. We can request to use the MATLAB code to find this matrix.

$$[(PW)^T]^{-1} = \begin{bmatrix} 0.0008 & -0.001 & 0.0004 & 0.013 \\ 0.001 & -0.001 & 0.002 & -0.001 \\ -0.36 & 0.0068 & -0.008 & 0.005 \\ 0.032 & -0.013 & -0.44 & 0.24 \end{bmatrix}$$

$$K = [(PW)^T]^{-1}[\alpha - a]$$

$$= \begin{bmatrix} -0.005 \\ -0.012 \\ -0.770 \\ -0.065 \end{bmatrix}$$

so this is basically our control gain vector, and the control then we can form as

$$\delta_e = -K^T X = 0.005\Delta u + 0.012\Delta w + 0.770\Delta q + 0.065\Delta\theta$$

(Refer Slide Time: 22:32)

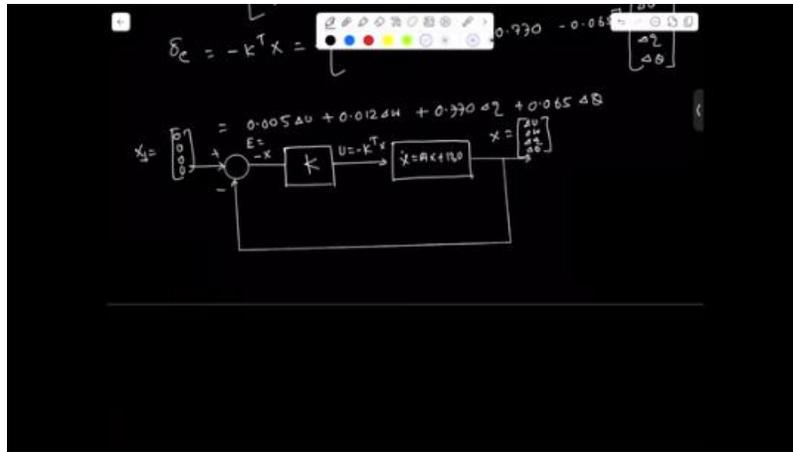
So, here we can draw the closed-loop block diagram to show how the control system can be designed for this particular system. So, here we have the desired value,

$$X_d = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

And from this K , we can form the control, which is $U = -K^T X$. So here, if you notice, we are having $-K$ here, which is E equal to error, and this is going to the plant $\dot{X} = AX$. And we have the output, which is x , is nothing but all these states Δu , Δw , Δq , and $\Delta \theta$.

And we have the feedback here. This is negative, this is positive. This is basically the control system for this particular motion dynamics, and we can control all the states in the system. Now, the question is, if you notice here carefully in the example, all the examples we have taken so far in this course, we are assuming one control input, right. So, this is actually making our problems easier to design the control.

(Refer Slide Time: 25:36)



But what will happen if the system has $\dot{X} = AX + BU$? And for example, U is not a scalar. If it is a vector, what will happen? So, in the next lecture, we will be taking the lateral stability augmentation, and where we will be using two controls. So, as of now, we have discussed in the last two lectures how we can design the longitudinal stability augmentation.

In the next lecture, we will be discussing lateral stability augmentation, where we will be taking a more critical system where two control inputs will be present. And having two control inputs, there are some different terminologies you have to use to design the control. And the MATLAB code for this example, this is the MATLAB result we have. So, we have found this A here, this is the A and the rest of the code, you can easily write how to find the response of the individual state.

So, that is common for all the systems. The way we did the previous example, how to find the response, you can do it. So, you have to find the closed-loop system model. And you can, you have to use the initial command, which will give you the state information. You can plot over time.

(Refer Slide Time: 27:38)

The image shows a MATLAB script in a dark-themed editor. The script performs the following steps: 1. Clears the workspace and closes all figures. 2. Defines matrix A as [-0.01 0.1 0 -32.2; -0.4 -0.8 180 0; 0 -0.003 -0.5 0; 0 0 1 0]. 3. Defines matrix B as [0 -10 -2.8 0]'. 4. Computes the damping ratio and poles of A using the damp function. 5. Plots the root locus for the system using rlocz. 6. Finds the closed-loop poles p using the roots function. 7. Finds the desired poles r using the roots function. 8. Finds the eigenvectors V corresponding to the desired poles. 9. Computes the feedback gain K using the place function. 10. Computes the closed-loop system matrix ACL = A - B*K. 11. Computes the damping ratio and poles of the closed-loop system using the damp function.

```
1  clc;clear;close all;
2  A = [-0.01 0.1 0 -32.2;-0.4 -0.8 180 0;0 -0.003 -0.5 0;0 0 1 0];
3  B = [0 -10 -2.8 0]';[wn,zeta,poles] = damp(A);
4
5  rlocz(A,B);
6
7  p = [1 3.61 9.05 0.126 0.09];
8  r = roots(p);
9  V = r';
10 K = place(A,B,V);
11 ACL = A - B*K;
12 [wn,zeta,poles] = damp(ACL);
```

$\dot{x} = Ax + Bu$
 $U = Kx$

So, let us stop it here. We will come up with the new topic in the next lecture, lateral stability augmentation, how we can implement it for multi-input systems. Thank you.