Advanced Aircraft Control Systems With MATLAB / Simulink

Prof. Dipak K. Giri

Department of Aerospace Engineering

Indian Institute of Technology Kanpur

Lecture 24

Stability Augmentation for lateral dynamics

In today's lecture, we will be discussing how we can design a multi-input control law algorithm. This is very important in modern control techniques. As of now, you have built with the single input, but if the system has multiple inputs to control the state of the system, how can we build this kind of problem? And also, we will have the lateral stability augmentation. Concept and how we can apply that concept to design the stability augmentation system for the lateral motion of the aircraft. So, here let us assume we have a second-order system

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

And if you notice here carefully, we are having two control inputs, but in the previous example, whatever the system we have taken, we had only one control input. So, how to deal with this kind of problem? So, here if you write In state space form, the system, we can write

$$\dot{X} = AX + BU \ldots Eq(1)$$

So, here u is actually u_1 and u_2. So, there is a scientist, Sriddharth Oehman. The proposed method allows us to design multiple input control algorithms for the system. So, the control algorithm, the control algorithm can be defined as for this multi-input system

$$U = -GK^T X + U_p \ldots Eq(2)$$

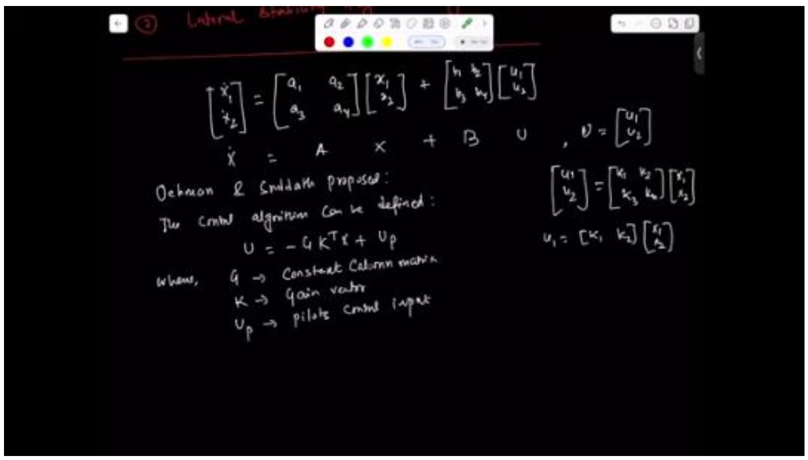
So, here you can write where G is a very important parameter, a constant matrix, and K is the gain vector, and U_p is the pilot control inputs. So, before you push it, if you would like to design this

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} k_1 & k_2 \\ k_3 & k_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$u_1 = [k_1 \quad k_2] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

This is how we have designed it. But in this case, we are going to design two control inputs separately. So, our control gain matrix should be in the structure, but the system is a second-order system. We need to find four different control parameters, so it is quite challenging for unknowns. If we have a system 2 by 2, this is to avoid this problem. Our control algorithm has been modified in the structure where G is assumed to be a row vector and K is the control matrix. Substituting equation 2 in equation 1

(Refer Slide Time: 05:24)



$$\dot{X} = (A - BGK^T)X + BU_p$$

$$= A_{CL}X + BU_p$$

Where $A_{CL} = (A - BGK^T)$, So, now the question is how we will choose this G. This is very important. So, G generally, as a design and control design engineer, we choose suppose G is the row vector for the second order system. We can form G as

$$G = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}$$

so for this, we have to choose g_1 or g_2 Is equal to 1, and what we are going to do is we'll use this condition, and from this, we can find if, for example, g_1 is 1, so we can form some ratio

$$\frac{g_1}{g_2} = \frac{u_1}{u_2}$$

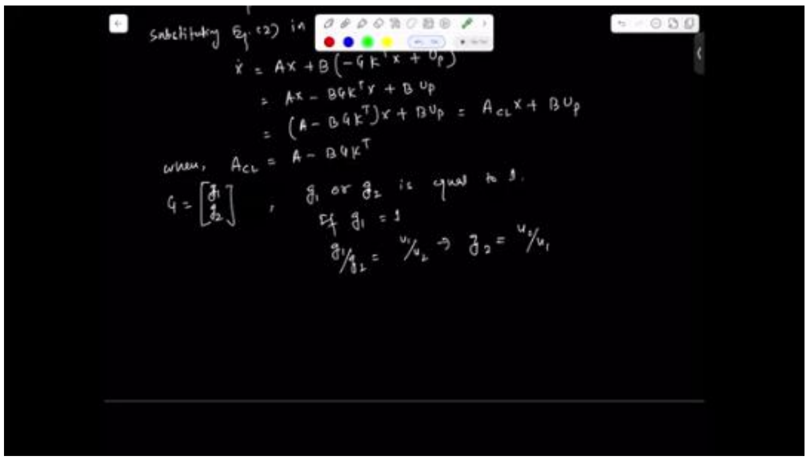
$$g_2 = \frac{u_2}{u_1}$$

This is how we can define, and we'll have an example of how to build this problem just now. So, this is how we will be choosing basically in our problem g_1 g_2 is known to us. Values and based on that, we can in that case,

$$GK^T = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} [k_1 \quad k_2]$$

$$= \begin{bmatrix} g_1 k_1 & g_1 k_2 \\ g_2 k_1 & g_2 k_2 \end{bmatrix}$$

(Refer Slide Time: 08:52)



So only two parameters are known. The same way, I mean, if we apply this concept, we can convert the system into the simple form that we have done before. So let's take an example. This part will be clear. So, example. Let's take an example. Design Lateral stability augmentation system to improve the Dutch roll. So, Dutch roll is one of the approximations in lateral stability. This part is already Roll motion, Dutch roll characteristic of the aircraft. So, here we are having two parameters which get variable

$$\begin{bmatrix} \Delta\dot{\beta} \\ \Delta\dot{r} \end{bmatrix} = \begin{bmatrix} -0.04 & -0.99 \\ 1.5 & -0.21 \end{bmatrix} \begin{bmatrix} \Delta\beta \\ \Delta r \end{bmatrix} + \begin{bmatrix} 0 & 0.01 \\ -0.008 & -0.008 \end{bmatrix} \begin{bmatrix} \Delta\delta_a \\ \Delta\delta_r \end{bmatrix}$$

Okay, so if you notice in this equation, we are having two control inputs One is then delay and this is and the desired handling characteristics are given by

$$\xi = 0.3 \qquad \omega_n = 1\, rad/s$$

So, using these parameters, we can form the desired Cartesian equation, and we can tally the augmented Cartesian equation and the desired Cartesian equation to find the unknown parameters k_1 k1 and k_2. So, let's start the problem. So, we'll be using this concept, which you have done on this page. We'll use this concept to find the multi-input controller algorithm solution.

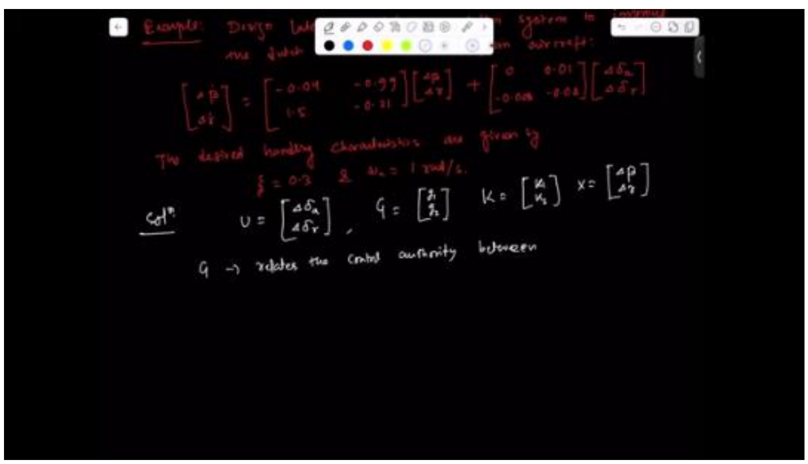
$$u = \begin{bmatrix} \Delta\delta_a \\ \Delta\delta_r \end{bmatrix}, \quad G = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}, \quad K = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}, \quad X = \begin{bmatrix} \Delta\beta \\ \Delta r \end{bmatrix}$$

Here, we can see that G actually relates the control authority between. Between aileron and rudder, and you can say for augmentation, so we are assuming, similar to the explanation we had here, so we are assuming. g_1 equal to 1, and g_2 we can find.

$$g_2 = 0.25 = \frac{\Delta\delta_r}{\Delta\delta_a}$$

So, this is the value we are assuming, and this is our assumption. So, we can write with this assumption. Write here with so these are the things you have to assume as a control engineer. You have to choose these values and see the result, whether you are able to track the desired values or not, and how you can infer the system performance. So, this is basically trial and error in some sort. So, with this assumption.

(Refer Slide Time: 14:51)



The maximum authority of rudder and aileron would be achieved. So, now we will find the augmented matrix

$$A_{CL} = (A - BGK^T) = A - B^*K^T \qquad (B^* = BG)$$

So here, if you notice carefully, this system has been converted to a simple augmented system.

So here We are going to find this value, so first we'll find okay, one more thing here. We are going to use the Bass-Gura method. The Bass-Gura method is what you have done in the previous lecture to design the control. So, now first let us find B^*.

$$B^* = BG = \begin{bmatrix} 0 & 0.012 \\ -0.008 & -0.08 \end{bmatrix} \begin{bmatrix} 1 \\ 0.25 \end{bmatrix}$$

$$= \begin{bmatrix} 0.003 \\ -0.028 \end{bmatrix}$$

And also we are given the desired characteristic of the desired dynamics. So from the given values, we can find the desired characteristic equation. So the desired characteristic equation will be formed as

$$s^2 + 2\xi\omega_n s + \omega_n^2 = 0$$

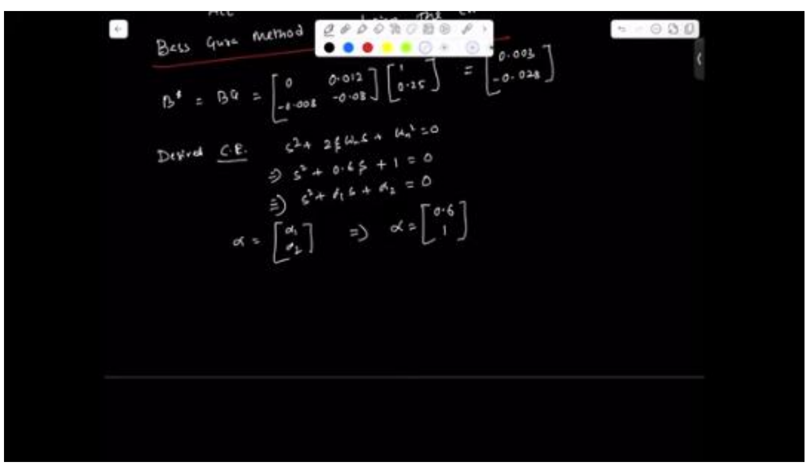
$$s^2 + 0.6s + 1 = 0$$

$$s^2 + \alpha_1 s + \alpha_2 = 0$$

$$\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} 0.6 \\ 1 \end{bmatrix}$$

Now, we have to find the A vector also of the system polynomial. So, that we can find now A vector because in the Bass-Gura parameter, if you remember, we had α, a and also P and W. So, these are the parameters we had.

(Refer Slide Time: 20:27)



This is the expression we had for the Bass-Gura method, right? The K, how to find the K matrix of the basic control gain matrix. So, now, a vector can be obtained from the plant matrix. Eight, right? So, for that, we have to use, we have to find the characteristic equation:

$$\left|\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} -0.049 & -0.99 \\ 1.5 & -0.21 \end{bmatrix}\right| = 0$$

$$s^2 + 0.25s + 1.5 = 0$$

$$s^2 + a_1 s + a_2 = 0$$

$$a = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 0.25 \\ 1.5 \end{bmatrix}$$

Now, we have to find the P matrix, the controlled matrix, so we can use, we have to use this P matrix here. So, here the controllability test matrix is

$$P = [B^* \quad AB^*]$$

$$AB^* = \begin{bmatrix} 0.02 \\ 0.01 \end{bmatrix}$$

$$P = \begin{bmatrix} 0.003 & 0.02 \\ -0.028 & 0.01 \end{bmatrix}$$

$$W = \begin{bmatrix} 1 & a_1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0.25 \\ 0 & 1 \end{bmatrix}$$

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

And solving this, the k found to be

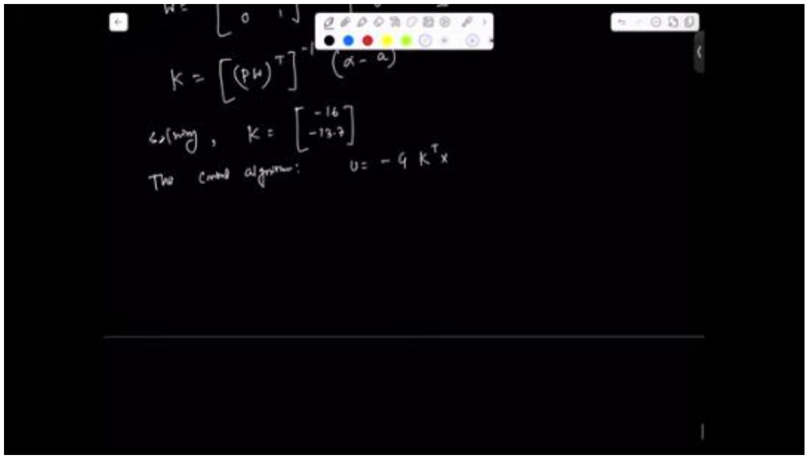
$$K = \begin{bmatrix} -16 \\ -13.7 \end{bmatrix}$$

and we can find the control, the control algorithm is found to be

$$U = -GK^T X = \begin{bmatrix} \Delta\delta_a \\ \Delta\delta_r \end{bmatrix}$$

$$-\begin{bmatrix} g_1 \\ g_2 \end{bmatrix} [k_1 \quad k_2] \begin{bmatrix} \Delta\beta \\ \Delta r \end{bmatrix}$$

(Refer Slide Time: 26:19)

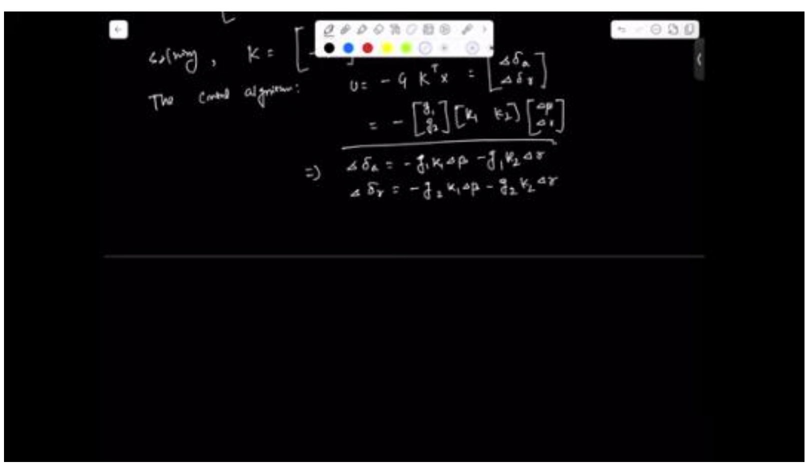


$$\Delta\delta_a = -g_1k_1\Delta\beta = -g_1k_2\Delta r$$

$$\Delta\delta_r = -g_2k_1\Delta\beta = -g_2k_2\Delta r$$

So, these are the control algorithms for this system. If you notice, this control has two controls, right? So, now this is how we can find our variable parameter. Now, if you want to use a different formula, you can easily use it. Instead of vast parameter, you can use a different concept of this problem, you can use the place command and all this stuff. You can find the system because linear dynamics is stable. You can find it. OK, if you want to use another method, let me give you the hints: another method to find control parameters.

(Refer Slide Time: 27:43)



So here, you want to use basically

$$A_{CL} = A - BGK^T$$

$$|SI - A_{CL}| = 0$$

$$|SI - A + BGK^T| = 0$$

So I is basically a two-cross-two identity matrix. So we can have,

$$s^2 + (0.25 + 0.003k_1 - 0.02k_2)s + 1.5 + 0.02k_1 + 0.003k_2 = 0$$

And also, we have the desired dynamics. Desired characteristic equation we have,

$$s^2 + 0.6s + 1 = 0$$

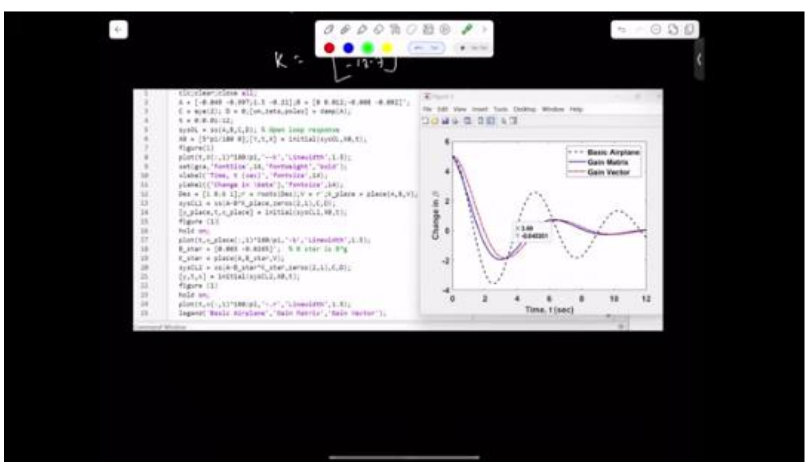
So, if you compare both these equations, K is found to be

$$K = \begin{bmatrix} -16 \\ -13.7 \end{bmatrix}$$

You can control the matrix. So if you notice here, the k found by this method and K found by this method are the same. So now we can go to the MATLAB code to see how we can solve this problem in MATLAB. If you notice, we have this here. Actually, we have different elements. So, we have done this for the beta, and the dotted line represents the basic airplane, while the solid line represents the gain matrix and gain vector. So if you notice here, you can see that the system goes to the zero equilibrium point, and this is what we have done here.

The same thing we have done in MATLAB. So, you can go through the details. You can try it in your MATLAB command window, and we are supposed to get this result. I am not going through the details. This is the same.

(Refer Slide Time: 31:45)



The A matrix is given, and the C matrix is given. So, for the C matrix, we are assuming all states are available in the output, and D is 0. So, this is how we can, I mean, this is the closed-loop system, and the initial condition is assumed to be 5. A degree that is 5, and this is 5 radians. This is the closed-loop system, and you have the initial command to find the solution of the closed-loop system equation, and this is basically how to plot the states. So, you can try it in MATLAB, and you can find the result.

So, let us stop here. We will continue in the next lecture with how we can design the stability augmentation system for the full lateral motion dynamics of the aircraft, and we will also be assuming multi-input conditions. Thank you.