**Course Name: INTELLIGENT FEEDBACK AND CONTROL**

**Professor Name: Leena Vachhani**

**Department Name: Multi-disciplinary primarily for Mechanical, Electrical, Aerospace and Chemical engineering streams**

**Institute Name: Indian Institute of Technology Bombay**

**Week - 03**

**Lecture - 17**

Hello, in this video we will be looking into feedback decoupler and centralized controller designs. In the previous video we had an idea about what decoupler means, and at the same time we had seen the feedforward way of decoupling. Again, the decoupler is of the form, which is helping us in getting paired input and output for the multivariable SISO problem to be solved. Let's start understanding feedback decoupler design. To begin with, we'll consider the system as a linear system, and the system has M inputs and M outputs.

We'll consider the square system matrix dimensions. The dimensions is such that the system matrix is square. And at the same time, we'll consider that the entire state is measured. We'll understand why these assumptions are important. We already know that this linear system state-space representation is given by x dot equals ax plus bu and y is equal to cx plus du.

Now, in this case, we will consider that there is no direct input-output relationship. But if it is there, then we will consider that even if it is there, then we can say that the ith row of the D is equal to 0. If we will consider this way, then we will consider applying chain rule. If I'm considering, it is easy to consider that this is zero by considering that okay the entire input-output relationship I'm not considering for the control design that can be considered later on. When I apply this particular chain rule, then I take the derivative of i-th row of y.

This is going to be the i-th row of c times x dot. Now, x dot is now replaced by ax plus bu that I already have. Given this, what are we going to achieve with this is that we will keep taking this derivative. Now, this derivative is to be taken till I have some non-zero term for B appearing. I am looking forward for designing a decoupler.

Now, this decoupler is going to give me the direct input and output relationship. Now, since this input is not appearing in this particular Y output because I am not considering that particular D to be zero. If this D is not there, then I would like to bring that in terms of considering the derivative of it. Now, if I am taking the derivative, let us take first derivative of it. So, I will get the term in Yi dot, I get term of u as iCBu.

Now, let us say this is equal to 0. So, then I will take the second derivative of yi, the second derivative for the ith. So, this way I will keep doing it. So, for the second derivative, what I will give, get iC A to the power 2 B u. So, this is what is the u term.

The rest of the terms are also there. But this is what is my u term. So, this is i-th row of CA square and B. Now, if this is 0 then I will keep doing this particular exercise of getting this yi third derivative. I will keep doing it till I get this particular term which is corresponding to u as non-zero.

So, this particular R i times that derivative is taken till the u term is appearing is called output relative degree. Now, this output relative degree is going to give me this particular d R i d t R i. So, this I have taken the derivative R i times of the ith row of y. This gives me this particular relationship and we know that we are stopping it which means this is not equal to 0 and rest of the other iCA are i minus 2, B are all 0s. So, this way what we are achieving here is that the output is being related to the input through the system matrix and this is going to help us in designing the decoupler as we see in the next slide.

So now we will start by compiling all these I measurements. And I say this is a Y tilde, for example. This is having certain derivatives of Y1, Y2 up to YM because there are M measurements available. So, certain cases this will be say R1 is just 1, R3 is just 3 and so R3 is 2 and so on. So, then I have say dy1 by dt, but the other one was d to the power of 5, 4, y2 by dt 4.

This can happen. So, I will keep taking the derivatives till I am getting the u term, that is what we will consider. So, this R1, R2, Rm could be different for individual measurements, that is the idea. And now, when I am clubbing all these equations relative to Ith rows, I get this vector, this particular matrix and this particular matrix which I am turning down as H tilde X plus Q tilde U, All right. Now, looking at this particular equations, we have Y tilde equals H tilde X plus Q tilde U.

This we have derived. Now, in this case, my control input can be returned as Q tilde inverse Y tilde minus H tilde X. What is Q tilde in this case? This Q tilde is nothing but this particular matrix. H tilde is this matrix.

All right. So we have we can now set a form of state feedback such that U equals minus KD times X plus F times Y tilde. This will get clear a little later when I show you the block diagram. As of now, you can understand that now I have a transformed y tilde means, instead of y, I have y tilde, a transformed variable y dot, y tilde. So what is this particular y tilde?

Let's understand this. This y tilde is nothing but the different derivatives, ith derivatives that I have taken. So, when I see the relationship between Ys and Y tilde, this is nothing but a diagonal relationship in terms of the transfer function forms. This is integrator, secondary integrator, multiple integrations of the power of R1. Those many integrators equal to the R1 times of integrators.

But interesting thing to observe is that this particular matrix is now diagonal. So, I need to design now Y tilde the controller between Y tilde and Y of S. Correct. So this is my final. More or less, we are saying that the feedback decoupler is helping us in getting the transform variable Y tilde of S, which is giving me the relationship, which is of a diagonal form.

So, this is my output of the system. This becomes my Y tilde as a transformed U star in case of the ideal decoupler. And this becomes a diagonal form of the system. Now, in order to understand this better, what we have is now the equations saying that Y tilde is equal to H tilde X plus Q tilde U. So, we have Y tilde X given by H tilde X, H tilde, sorry, let me, let me go back, H tilde X plus Q tilde U, Okay, so what we have here is Y tilde given by H tilde X plus Q tilde U. And this is something I am representing in the block diagram form. If I have Y tilde given by Q inverse, so this term is your Q inverse, Q tilde inverse Y tilde. So your U becomes Q tilde inverse Y tilde plus H tilde X. Correct.

So this is exactly what is happening. And and of course, there is a small correction here. This will take care of this Q and either either I should have Q tilde inverse over here or instead of that, this block should be over here. Q tilde inverse. All right.

Having said that, the idea here is to tell you, show you that this particular form is now my diagonal form. And because there is a relationship between Y and Y tilde, which is giving you a diagonal form. And we solve this by the way of a feedback methodology, which is involving Q tilde inverse H tilde. All right. Similar to the previous case, you have Q tilde and its inverse appearing here.

But at the same time, there is a high chance that you don't. Again, this Q tilde inverse is dependent on the A matrix, C matrix and so on and so forth. C vector and so on and so forth. Fair enough. Let's see, would you be able to find a feedback decoupler if you have been given this particular A, B, and C matrices?

This is a homework problem. Use the same method to convert the system in the diagonal form using feedback. As a summary, the diagonal behavior using coordination of the sensors and actuators when this is needed, this becomes a very powerful technique. Mainly because it uses a feedback methodology for conversion to the diagonalization. It cannot be achieved using standard industrial regulators, but at the same time, one can design since now we already have digital methods available, we should be able to design for newer applications this Q tilde and H tilde matrices to get the feedback methodology.

Feedback is always better in terms of the model robustness and so on and so forth. Disturbance rejection is only if your disturbance enters independently at each output. In the previous case, this was very prevalent, feedforward way of decoupler design. Here your disturbance rejection is going to be better because this is getting solved with the help of a feedback method in any case. It is also sensitive to the model errors because you have designed new blocks which are Q tilde inverse and H tilde which are dependent on A matrices, the system matrices and the input and output matrices.

It needs, this particular methodology specifically needs full state measurements. Having said that, this particular method has been used in many practical cases where you have the full state measurement available and you have the nice methods to get the derivatives of the measurements available and so on. All right. Now, coming to a centralized control method. You have looked into MIMO system, which can be simplified as a MVSISO method.

But in certain cases, that is also not possible. So then I will have to design a centralized controller. I am not looking at designing independent SISO systems, but a centralized control system, which will be able to control all the states. So in this case, I'm again going to use a very simple methodology pole placement, which we are using in the SISO methods. Certain times, you will get the solution.

Since it is too many variables, to certain extent, you will get the solution. Number of independent sensors is equal to the order of the system is what our assumption is. In other way, we can say all the states are measured. And so in this case, to understand a little more carefully, I am considering that the C matrix is nothing but an identity matrix, means that the output is directly related to the state of the system. So for the LTI state feedback control methodology, what we consider here is that the control input U is defined by minus KX plus R, where K is my gain matrix, and R is the reference point where I want to settle down.

Because finally, one has to make this particular bias correction every time, because not all the time you would like the state to converge to the origin. You would like the state to converge to a particular reference value R, and that's why this kind of a correction is needed. So for example, now I have this particular gain matrix K11, K12 is to be considered. At the same time, I will talk about this particular variable V, which is driven to the reference vector, set point vector or a reference vector R to be considered. So this particular value V, the bias value that we consider here, where the system has to rest is to be adjusted at the control input side.

And this bias is calculated by the transfer function gain value at S is equal to 0 or the steady state gain value. And this is given by G0 inverse times R. Because this is a vectored form and that's the reason I brought this here. Otherwise, in the SISO form, it is very simple way of adjusting the gain based on the gain of the system at steady state. All right, now what we get here is X dot equals A minus BKX plus B times V.

Because now what I did in this case is nothing but I had a system state space representation as AX plus BU. U is replaced by KX plus V. So, sorry, this is minus KX. So minus KX plus V. Now, if the system is fully reachable, the closed-loop eigenvalues can be assigned to any arbitrary desired position by appropriately selecting the K.

So my job for the controller design now becomes designing the gain matrix K here. And that's interestingly can be solved using the pole placement direct method. Simple way. So we have a polynomial for which I have polynomial, the characteristic polynomial can be obtained by SI minus Acl. I is my diagonal matrix and Acl is the is the closed loop system matrix.

So this is what is my desired closed loop characterization. Now this particular one is to be matched with my A minus BK value because that's how I am adjusting my gain K. Now, zeros of this polynomial are the closed-loop eigenvalues. That is what we understand it. And we need to calculate K so that polynomial is exactly the desired one.

Again, there's a matching purpose needed, but at the same time, it is fairly okay when we are doing the pole placement method. Now, for example, let's take this as an example here to see what comes as a centralized controller design. Let's say this is a three cross three system matrix, which means I have three states. Fair enough. B is two cross three, which means there are two inputs into the system.

Let's say I want to consider desired settling time is one second. So if the desired settling time is one second, I need to place poles at minus three and minus four. If that is the case, then my characteristic equation should turn out to be S plus 3, S plus 4 whole square. Let's say we want to place two poles at minus 4 because I have a third order system sitting there. I need to have some desired poles to be considered as third, three number of poles to be considered.

So this gives me the desired characteristic polynomial as s cubed plus 11s squared plus 40s plus 48 At the same time, I have to design u, which is k11, k12, k13, and x1, x2, x3. So there are six variables, and when I do the coefficient matching, I will get three equations over here. S cube is definitely normalized one. So I will consider coefficient matching.

So then I will get three equations in this case. In the SISO case, we had three equations and three unknowns. So we'll get unique solution. Whereas here we have three equations and six unknowns. So we will have many choices available for taking six such gain values.

So which one to consider? One can apply some kind of an optimal way of finding out what should be the different constraints to be considered for gain values. I can add some extra performance metric and then look forward for getting some optimal solutions for the gain. Typically, we get some of the other gain values which are matching with our performance criteria. This is again, tuning is not at all intuitive.

One has to come up with some trial and error method here. Since this is centralized, this is a non-fault tolerant method. Anything gets spoiled in this controller or anything, any other part is missing, any other failures happen, the entire system shuts down as compared to the MVSISO method where even if one control loop is not working, others will keep working on it. So some performance output on the certain output variables you will get. Further, these centralized control methods are implemented via non-standard equipment such as data acquisition cards or communication or the industrial computers because we need to estimate the state and then this particular state estimation to certain extent is helping us in getting the gain values and so on and so forth because that's where the optimal performance that can be considered over here.

Gain values are again non-tuneable. So one has to enter into it and come up with some method to look at what the performance with respect to the gain. It's not that one, I'm changing this particular gain of K11. What is going to be the effects on K1 to K2 and so on, on other output performance? It's very unpredictable and we will not be able to match it.

And that's the reason tuning is non-intuitive as compared to the PID tunings where we had at least some idea that, OK, when I'm changing the integral gain, what is expected, right? In this case, one has to come up with some optimal way of finding the gain values. That's all. And again, the same two references can be considered for more understanding of decentralized control as well as feedback decoupler method from these two references. Thank you.