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 - 02

Lecture - 12

Hi, this video is about internal model control. This particular method of internal model control is being also used with nonlinear systems or complex systems methods, complex systems and the controller designs for that. The objective here is to introduce you to internal model control and to simplify it and to understand that how is it related to PI control or PID controls. This is also similar to the pole placement method. But at the same time, if we look at from the perspective of designing this IMD control, which we will see its objective turns out to be output disturbance rejection.

The name internal model controller derives from the fact that the controller contains the model of the process internally. Once you see the method for designing this controller, you will understand that if the model of the system is known, then one can design it appropriately for a particular objective, which is output disturbance rejection here. And since it uses the model completely, that's the name turns out to be internal model control. All right, so let's see the method here. It turns out that if I have this particular process P, which is the model of the process is known.

For example, its transfer function is P, and it's fairly accurately known, and its estimate is given by P hat. Then if we structure the controller in this way, which involves a particular filter GF, we will come to know about the role of this GF here. And this is what is my P hat in pseudo inverse, because the process can be having multiple inputs and multiple outputs. And so the transfer functional matrix can have its pseudo inverse applied here. So it is it is more or less a general methodology for the output disturbance rejection.

So one can see that if I'm placing it this way, one can answer by saying, okay, if my GF is equal to one, for example, it's just a simple way of trying to understand that let's say GF is not at all there and P hat is exactly equal to P. Then is it having the total disturbance rejection? One can find the transfer function between Y and D and see to it that it is almost equal to zero. All right. So but at the same time, what turns out that this controller, when we start this particular controller, when we start writing its transfer function, which is nothing but equal to U of S by YSP of S.

So, which is output because the controller output is given by U here and the input is YSP. And when we write it in the transfer function way, it turns out the controller transfer function is given by GF P pseudo inverse by 1 minus GF P pseudo inverse P. If my estimate of the process P hat is equal to P, then what happens? P hat is equal to P and GF is equal to identity. Then this blows up.

And that's why GF is playing an important role in designing the controller here. We are not saying that the process model is accurately known. Even if there is an estimate available, one should be able to design a controller for the output disturbance rejection here. So this is a very generalized methodology which is based on the model, but it does not expect that the transfer function should be or the model of the system should be completely known. If it is completely known, then it's a perfect disturbance rejection.

But at the same time, with the help of a proper design of GF, one can still achieve the very, very good output disturbance rejection here. And GF is also important if my p hat is exactly equal to p. so that the controller is just not behaving like an infinite controller, infinite system here. All right, so my GF is introduced to open a system that is less sensitive to modeling errors. So we have some discrepancy between p-hat and p.

We saw that c is having some kind of a matching being done over here with p-hat pseudo-inverse times p-hat. So this is my estimate. P hat is an estimate. And this estimate, its inverse and its multiplication is being cancelled out over here. Fair enough.

But then this particular pseudo inverse should match with P in order to have the complete output disturbance rejection. If my P hat inverse and P, so this C block multiplied by P is my control system block, right? So this turns out to be the requirement that P hat pseudo inverse P should be equal to identity. If there is a complete match, there's a complete match, complete identity matrix turns out here. But if there's not, then GF can be designed in order to reduce the disturbance thing coming out of disturbance input reaching to the output.

So therefore, this GF is playing a role in making sure that even if this is not identity, there is a very less disturbance that is getting affecting the output Y. All right. It typically gives higher order controllers. I mean, even for the complex systems, one can design the IMC way, but one can simplify by making some special assumptions so that it gives you a PI or PID controller, right? Let's take certain example in order to motivate you what we are saying with the simplifications that we say. For example, my transfer function or the process is simple FOTD system.

Of course, we have already talked about it, that even the system with higher order system can further be simplified as an FOTD system, if my dominant pole is of that sort. Now, approximate inverse, we are not considering any non-causal systems. So, the approximate inverse of this particular process is given by 1 plus ST by KP. We are not considering E power minus SL as of now. All right.

So this particular filter design, this GF of S is a first order system or a low pass filter design way 1 by 1 plus STF here. There the transfer of this particular time period or time constant TF can be selected appropriately as compared to the time constant of the system P. And this particular time delay, if I consider e power minus SL as one by one minus SL, then we can consider adding this into the approximation here. And now C of S becomes one plus ST, one plus ST by KPS L plus TF. So my transfer function, so this time delay L turns out to be adding to L plus TF.

Now, what is this form? This form of the controller is nothing but my PI controller. Similarly, if I make a Pade's approximation to e power minus SL. So I'll get e power minus SL equals 1 minus SL by 2 by 1 plus SL by 2. And then one can find C of S given by adding this GF as well as C of S given by one plus SL by two, one plus ST, and just appropriately making sure that I'm incorporating this instead of one minus SL form, which can be approximated in this way, where my, what is the approximation we considered here is that KPS, this particular term, STFL by two is very small.

TF is anyway in my hands, so I can always make sure that it is matching with the delay L here. All right, so this particular way of understanding is nothing but this C of S turns out to be my PID controller. Try understanding what we are trying to achieve by representing this IMC concept as a PID control. If I had this C of S design, which was motivated from the model description over here. PI controller design, if I go with a pole placement way, we started by saying that, okay, this is a pole placement.

Pole placement will also give me something similar results with the approximation of the delay system as 1-SF. Same way, if I see the IMC way, my transfer function turns out to be again a PI controller. So all these methods to a certain extent are helping us in understanding that if we simplify the process, even the IMC method turns out to be giving us a PI or PID controller. If I interpret this kind of a method in terms of complex systems or the nonlinear systems, we would be able to understand that, okay, finally, the controllers are of the form of proportional integral or a derivative controller. Even if the system is nonlinear, even if my controller is nonlinear, its behavior is something, overall behavior is something that can be interpreted as proportional, integral, or derivative form.

That's all. With the IMC way of understanding, we can say that robustness is considered explicitly in the design because of the GF transfer function block that we have considered here. It can be adjusted by selecting the filter GF properly. There is always a tradeoff between performance and robustness that can be made by using the filter constant as a design parameter. And this particular design parameter is nothing but the filter time constant tf.

It can be designed to give excellent response to the set point changes because to a certain extent, one can see that this is a PID block. All right. One can design it for the feedforward way in order to get the setpoint changes. So mixed up methodologies can be used to achieve set point changes, output disturbance rejection, robustness. Everywhere one has to look into getting the trade-off between these two or three things.

Especially with the IMC way, there's a trade-off between performance and the robustness and performance in terms of the output disturbance rejection. That's all for this video. Thank you.