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 - 01

Lecture - 01

Hi, in this video we will look into understanding PID control and those who already know the PID control will still be able to grasp something out of this. So let's start. First of all, we have to understand why are we wanting to control any system? So we already know that we like to control the system because we want it to be a stable system. As in, we want this system to be controlled at a particular equilibrium point.

If there are perturbations, if there are practical impracticalities into the system, certain disturbances into the system, still one has to ensure that it comes back and converges to the equilibrium point. Or a set point in terms of classical control terminology. So now, we look forward for making the system, ensuring that the system is stable under the errors, disturbances, or practical circumstances. The second point that we look at is accuracy. So if I want to consider that I want the system to reach to a particular set point or tracking a particular signal, then what is the accuracy by which the system is able to reach to that particular set point or able to track the signal?

The third point that we look forward for control is responsiveness of the system. If the system is able to respond but it is a very slow responding system, then it may or may not be a good idea to continue with that system. There we would like the control system or controller to take part in improvising the time response of the system. We also look forward for robustness purposes for controlling the system. Today it is working, after 10 days or one year the model parameters are anyways going to change.

Because of the aging issues, because of the environmental changes, the model parameters or the system characteristics are going to change. And based on this, making sure that the controller is still performing some good job is what I aim to achieve for a robust control system. Having known these terminologies, what we look into is now designing certain matrix based on which I would be able to say that the system is stable, accurate or responsive enough or it is robust. So for stability, we already know about the stability criterias, especially for the linear system's transfer function where we look into bounded input, bounded output stability. For more enhanced versions of the stability or more strict versions of the stability are asymptotic stability, exponential stability, and whatnot.

For the system to be accurate, we typically look into the steady state error. What is the steady state error, and are we okay with that particular error or not? We describe responsiveness using rise time, peak time, or settling time. Robustness - The controller is able to detect the disturbance. It is also robust to the model parameter variations or not.

Okay, so now we know that these are the typical requirements coming up for designing a controller. We set out the control objective in terms of stability, accuracy, responsiveness or the robustness. Since the controller is also working with the practical system, there is a possibility that we will not be able to take care of all four kinds of the requirements that the controller is supposed to know. So we have to make sure that we are taking care of the trade-offs between these. For example, if I'm taking care of stability, I may be losing out on the responsiveness criteria.

So one has to describe very particularly of what is the priority requirement. Fine? So once the priority is set out, then one can look forward for designing the controller with the understanding that this is why I'm designing the controller. Moving forward, if I have to design the controller, this is the typical way I look into designing a controller with the help of closed loop control, negative feedback methodology and so on. So this particular diagram basically is introduced to describe the terminology.

So we have a plant which is the system, and then I am controlling it with the help of controller. The system has an output Y and control input U. The controller is receiving the input as error input, error variable, and this error with respect to what? The set point that I want to select, that we would like the output to follow. So, the set point is now set as YSP.

And this YSP minus Y gives you the error. It means how much the output is deviated from the set point. Okay. We already know what are the advantages with the negative feedback system. The negative feedback system is able to take care of the disturbances, process variations and because it also has the relationship between the variables in a system.

Relations as in what is the output coming out, and that output is fed back in order to take care of what command to be given to the plant. Let's see what are the simple forms of feedback now. Before moving to the PID, in order to take the benefits or the advantage of the PID, one has to understand why are we implementing PID or why are we not able to solve the problem in a very simplistic manner. The PID itself is a simple controller, but at the same time, can I do much simpler way? Can I design the controller in a much simpler way?

All right. So there is there are methods available that let's look into that. The first one is the on-off control. The on-off control says that, OK, if my error is positive, then I'm giving some U max. When error is negative, then I'm giving some negative command.

Fixed positive command and fixed negative command based on the sign of the error. This is fairly okay, and this works, but at the same time, what happens is that the process variable or the output oscillates as the system overreacts to a small changes in the error. So very small change in the error, you'll be just simply toggling between the plus command and the minus command. And if you're toggling so much, it means you're changing the control input very much then you are drawing the energy of the system. So one has to also make sure that do we really want to toggle that frequently or do we really want the errors to be almost zero or what not.

So you can see that just near the zero you are just simply toggling between plus and minus which may not be desirable. So then we came up with this dead zone criteria. All right. So in this dead zone, we say that, OK, error is zero. If error is zero, command is zero.

If error crosses by some plus delta value, then I will give some plus command. Or if it is negative than the negative threshold value, then I will give the other command input. But then the issue is that error is 0, command input is 0. If command input is 0, the output is going to be 0 after some time. So, what is not desirable is that if error means output is equal to the set point.

So, that time error is definitely 0 which is desirable with the command input need not be 0. If command input goes to 0, output will slowly become 0 and it will deviate from YSP. So, this is what is not desirable. Now, let us see what next we can do is the hysteresis kind of thing. So in this case, what this hysteresis says is that if the output error is positive, then the output is positive value.

But even if the error becomes little bit of negative delta, still I will keep giving positive value. And then if it crosses that negative value, then I will go to the negative command values. Similarly, if I'm coming from the negative side, I will be okay with some positive error, and then I will switch to a positive command value if error is more than a particular positive value. So the advantage is clear now that it is not when error is zero, the command input is not zero. And at the same time, it is taking care of the issue with the on-off control that it is not toggling that frequently.

All right. Of course, this is methodology is very nice. And this is typically used in your air conditioners too. You will see that if you set out temperature values to say 27 degrees centigrade, your body is not going to understand that this is exactly 27 degree temperature in the room. You will be OK with 26 to 28 degrees centigrade.

So that's the reason there's no need to toggle at exactly 27 degrees centigrade. So one can look forward for designing this hysteresis kind. Hysteresis is again on-off kind such that you are only giving plus command value or a negative command value. So you are either cooling or you are increasing the temperature. So you are shutting down your air conditioning.

All right. Fine, so now let's see what next for us. So here in the on-off case, of course, we were unable to reach to exact set point, but we were okay with a band of values around the set point. But if the requirement comes that the control objective is such that I want to reach to that particular set point and stay there. Stay there.

All right. So then what we will look into is designing a proportional controller. Now, this proportional controller, we already know its structure. It is the command input is equal to the proportional gain times the error YSP minus Y. All right. But what happens again here?

that when error is zero, your command input is zero. So, your process variable will often deviate from the set point as we saw in the on-off control case. So, then fine. So, then let us look into integral control. Now, what integral control does is it integrates the error from time t equal to 0 to a particular time instance of current time instance.

And of course, it is multiplied by the integral control constant. So in this case, what happens is when even when the error is zero, since there is a residual accumulation of error in terms of the integral term, what we get is there's a possibility that the system is, there is some part that comes up here. But if it is a pure integral term, then what happens is u0 is equal to ki e0 times t. If you have residual error remains, means there is some constant error is there, then what happens is this particular term u0 is going to keep increasing because there is a term t turning out over here. So it is the steady state output or steady state control input is equal to the steady state error e0 times t.

So this is not a steady state because this particular steady state is a function of time. Therefore, this is contradicting that this is a steady state. Because if the control input is changing, definitely your output is changing. So this cannot be the condition. It contradicts that this is a steady state.

And therefore, we say that steady state error has to be zero if it's an integral control. So something similar, we can have the proportional integral control term. So if you have proportional plus integral term, that takes care of making the steady state error to zero. The integral term makes sure that the steady state error term goes to zero, whereas the proportional term will make sure that you are approaching the steady state fast. All right.

Similarly, this particular PID term, which has the gain term multiplied with the error, which is your K times e(t) is your proportional gain. This is my this this is the form in terms of the integral time constant or K/Ti is nothing but your integral gain term. Next comes your derivative term, which is K times. derivative time constant times the derivative of error with respect to time. Now, if we look at the properties from the integral term, we are having the past errors. It is taking the accumulation of all the errors in terms of your integral term.

Whereas your P term is taking care of the present term, present error. And this d(e(t))/dt takes care of the future. If I consider the derivative as the, or consider as a linear extrapolation of the error as e plus de/dt. So that is what is more or less taking care of your future errors. So the combination of this past error, present error, and the future error.

Future error is nothing but your prediction that you are looking into. And this prediction, of course, is going to be good if your error is not changing very fast. All right. So that's what... That's what the interpretations of proportional, integral and derivative terms are.

And now let's look into what is the history of this PID control. We are still studying this PID control, which is approximately 250 years old. And the claim here is that 90% of industrial control is still PID. You will soon understand that why this PID control methodology is so powerful when we will simplify the methods and we will try to see that certain control objectives can be satisfied just by the implementation of PID control. But the implementation of PID, their structures, their different ways of implementing the PID is very important to learn here.

And which structure will fit into a particular process model or particular system characteristics is very important to understand. And you would be able to design PID control to satisfy certain control objectives by learning through this course. We see the history as 250 years old, but at the same time, these first controllers were all mechanical. They were designed for windmills or steam engines. The next came the era of pneumatic signal transmissions, and this is the era that brought the industrial process control.

Sensing, control, and actuations were still separate. And then controllers were sitting at us into a separate control room. So all the sensing data is coming to the controller, and control is being done at a control station, and then the actuation was done at a different place altogether. There came the era of, in 1915, electronic controllers. Mainly using OPAMPs, they were designed.

1970s, the introduction of microcontrollers changed the controller to design of the digital controllers. And this also added functionalities such as auto-tuning, adaptations, and diagnostics. And of course, it opened the use cases of controllers and control systems like anything. Then in 2000s, these digital implementations were done with the help of FPGAs, mainly to customize the controller designs and at the same time, very fast to take care of very fast sampling rates. because by FPGAs I can design my own digital logic and I would be able to do it without necessarily using all the functionalities of microcontrollers.

I can customize my digital design as well in FPGAs. So that was major advantage that we got from designing controllers for FPGAs. Now, newer trends is back to the analog implementation. With the quantum computing, with the analog computing being explored, and with these MEMS devices, micro-mechanical electrical systems, where we are again going back to this analog era where the controllers were designed using OPAMPs, where we need not to worry about the transformation of continuous signals to discrete signals, which is involving further approximations and so on. One can still work with continuous signals and design the controllers in analog domain.

That's all for this lecture. This particular video, we'll move to the next one. Thank you.