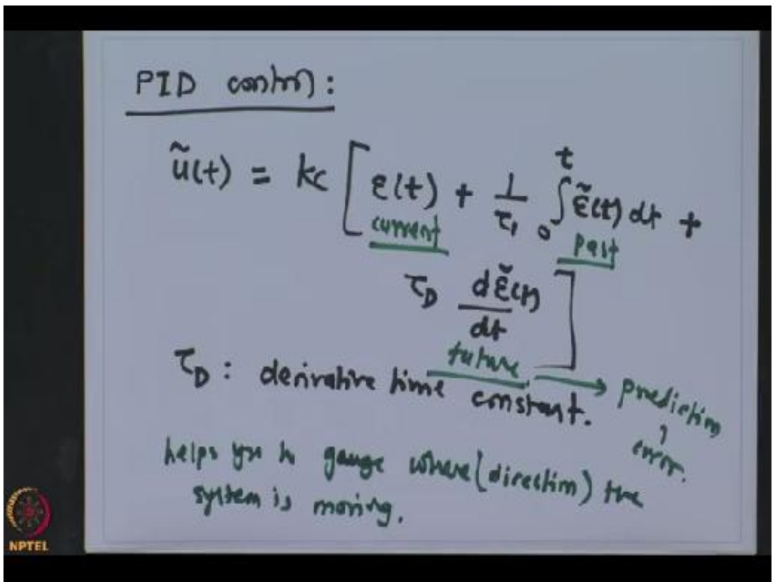
Chemical Process Control
Prof. Sujit S. Jogwar
Department of Chemical Engineering
Indian Institute of Technology - Bombay

Lecture – 25
PID Control

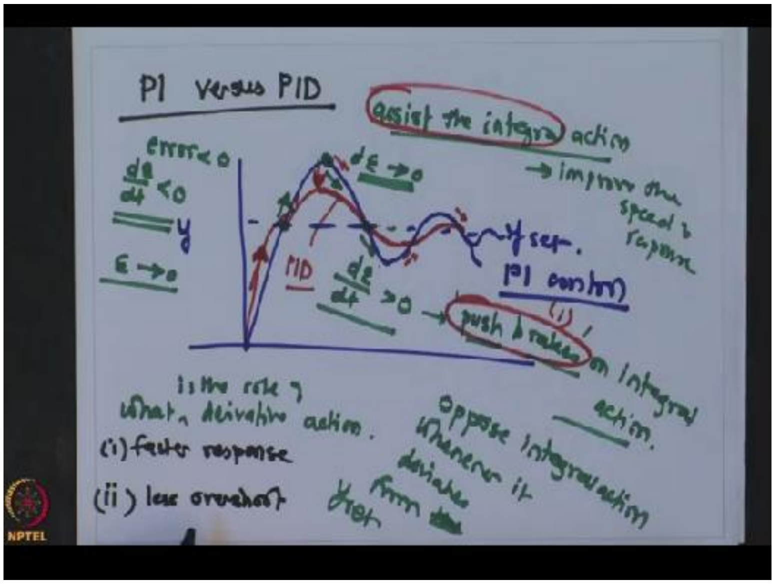
Welcome back. So before break we were looking at why we typically added derivative action in addition to PI control and the reason I said was it improves the speed of response by having some sort of prediction about where the system is moving.

(Refer Slide Time: 00:43)



So with the help of that what we get is known as a PID controller. So in this case the output depends on the present value of the error. It also accommodates any error which was there in the past. And in addition to that we have also look at the derivative of the error. And the multiplying factor is τ_D. τ_D is known as the derivative time constant. But if we look at the overall formula in the PID controller, this accounts for the current error or where currently the system is at, this looks at the history of the system. And in order to improve the performance, it also looks at the future. Because derivative is something which is going to give you some indication of where the error is moving. So it gives you prediction of error and because of the prediction of the error, it is able to fine tune the response. So with the help of this prediction of error, it helps you to gauge where the system is moving. Where in the sense in which direction and helps you make a correct move so that the system reaches the final value at a faster rate. So let me elaborate on that point.

(Refer Slide Time: 02:48)



So let us say we have a response of a PI controller and this is the set point, and the response of the PI controller is something like this. So we have requested an under damped response from a PI controller. So this is a response of a PI controller. And these oscillations occur when we have a larger contribution of integral action. So it always tries to compensate for the past errors more than the current error and in that way you end up having oscillations around a set point.

So now let us look at what the derivative action will do? So let us consider this particular point at which we have the instantaneous value of the output y, has reached the set point. But the system has not stabilized yet. It is still going to oscillate. So now the moment you move away from this point, let us say this next point, what we are going to see is the error is going to increase. The error is going to become negative.

So if I try to plot what is the error, so the moment I move away from this point, the error is negative because I am above the set point and so the rate of change of error is also negative because earlier the error was 0. So what we are going to see is the system based on the derivative action, it is going to detect that the system is moving from this final set point value. So the final value is, ε has to be 0.

So what it is saying is that the system is moving in the wrong direction and because of that what it is going to tell me is that cut down on the controller action. Same way when we are moving in the opposite direction. Same way it is going to tell me that the system is moving from the set point. So in this case, earlier the error was going towards 0. It reached 0 and then the moment it goes here, $d\varepsilon/dt > 0$.

So again the derivative action is going to predict that the system is moving away from the set point. And so it will try to push breaks on the integral action. So the whole idea of having a derivative action is that it is going to apply some breaks on how much weightage we are going on to the past or the integral action. So because of that as whenever the response tries to leave this set point, the derivative action is going to kick in, and it is going to assist, or it is going to oppose.

So it is going to oppose integral action whenever it deviates from the set point value. If we look at a point something like this, what we are seeing is that when the system reaches this particular point, and it tries to move towards the set point, then the derivative action is going to tell me that the system is moving in the correct direction. So it is going to assist the integral action. That way it will improve the speed of response.

So based on the direction in which the error is moving, based on looking at the derivative, what the derivative action is going to do? There are two things it is going to do. One is, it is either going to push breaks on the integral action. And as it pushes the breaks on the integral action, this overshoot keeps on reducing. It is not going to reach the same highest value as in the case of PI control and what we are going to see is the response, the overshoot will keep on reducing as we add the derivative action.

So this is for the PID control. That is because the moment it starts to move away from the set point, the derivative action is going to push breaks on the integral action. So the net integral action is always going to be less, and because of that, the response will have less overshoot compared to a simple PI controller with the same k_c and τ_I values. And whenever, this is the first thing it is going to do.

And the second thing is, whenever the system is moving in the correct direction like here or here or in this case or even in this form. Whenever the system is moving in the correct direction, in that it is going to assist the integral action. So this also causes the speed of response to increase. So what is going to happen is this controller will reach this value even faster because the reason derivative action is assisting the integral action.

So both these actions what it causes the net product of these two is that you have faster response and the second thing is you also get less overshoot. Obviously, in order to get these benefits, the derivative action has to be tuned very properly, and that will be the topic whenever we talk about control system design, we will look at what is the best way to choose these values. But in principal, the whole idea of adding the derivative action is that it is going to improve the speed of response.

It is going to reduce the overshoot, but again it also has its limitations. So we will take a short break here, and when we come back, we will look at the limitations of the PID controller. Thank you.