Dynamics and Control of Mechanical Systems

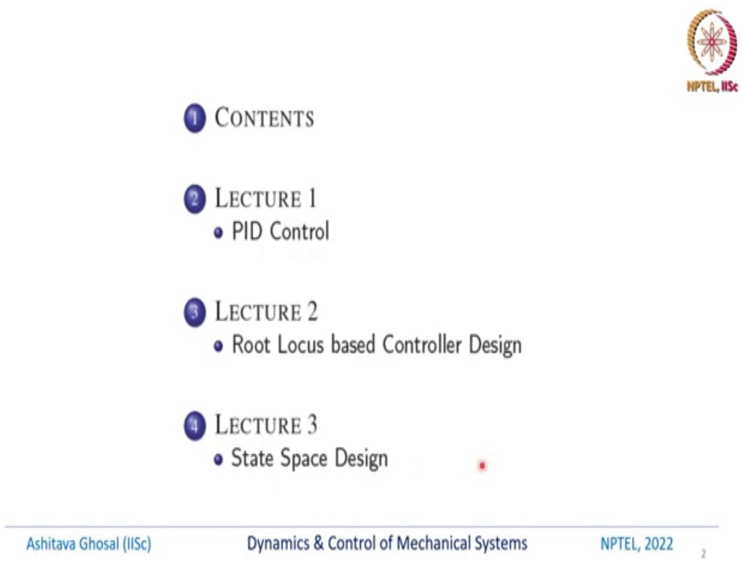
Prof. Ashitava Ghosal
Department of Mechanical engineering
Indian Institute of Science – Bengaluru

Lecture – 29
PID Control

Welcome to these NPTEL lectures on Dynamics and Control of Mechanical Systems. My name is Ashitava Ghosal, I am a professor in the Department of Mechanical Engineering, Centre for Product Design and Manufacturing and also in the Robert Bosch Centre for Cyber Physical Systems at the Indian Institute of Science, Bangalore. Last week we looked at classical approaches for Analysis of Control Systems.

Prior to that we had looked at state space formulation, modelling and analysis of control systems. In this week we will look at Design of Controllers and in design of controllers we look at three commonly used techniques - when you want to design a control system, to meet certain objectives, to meet certain requirements.

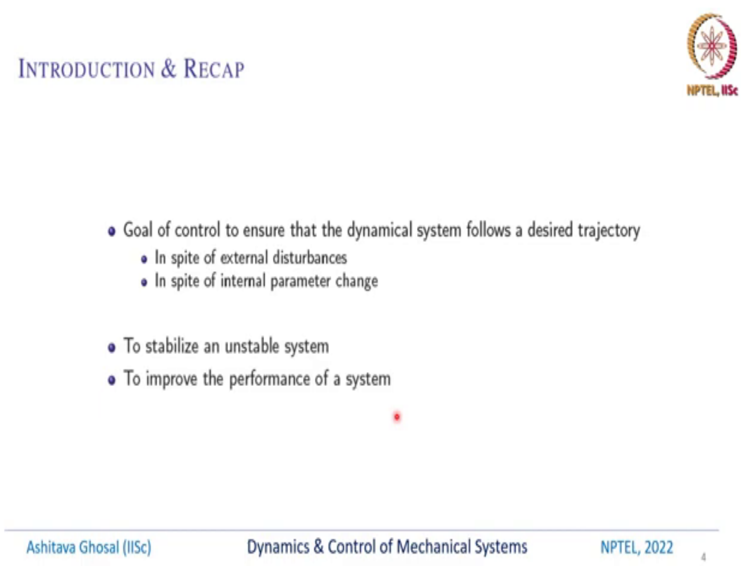
(Refer Slide Time: 00:01:19)



So, in this week there will be three lectures, the first lecture is on PID control, the second lecture would be what is called as root locus based controller design, and the third lecture will be state space based design of control systems. The PID control is very commonly used in industry, and we will first start with that.

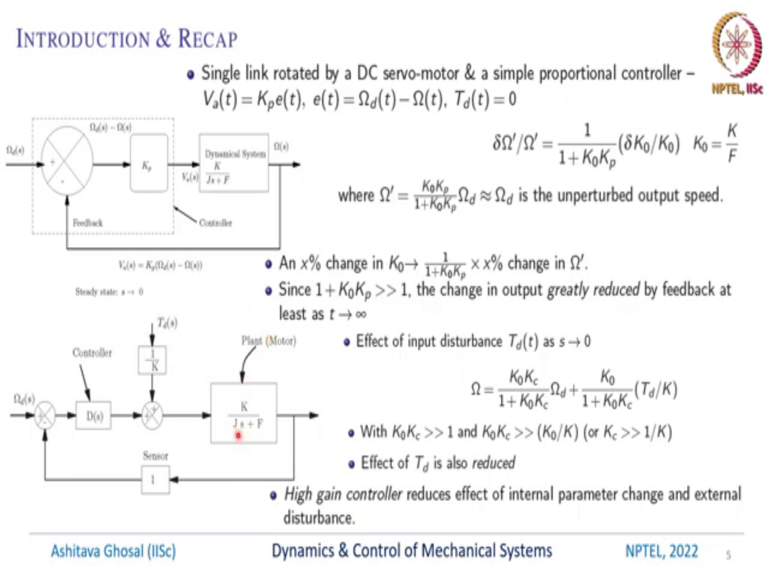
(Refer Slide Time: 00:01:46)

(Refer Slide Time: 00:01:48)



So, a quick introduction and recap. So, the goal of control is to ensure that a dynamical system follows a desired trajectory. And, in particular, in spite of external disturbances and in spite of internal parameter change. Sometimes, the control system is also used to stabilize an unstable system, and also to improve the performance of a system, we have seen how these things can be achieved by a control system for a simple case.

(Refer Slide Time: 00:02:23)



So, let us quickly go back and see that simple case which we had looked earlier. So, we have this DC motor which is rotating a link. The transfer function of the DC motor could be written in as $K/(Js + F)$.

The voltage which is applied to the motor is denoted by $V_a(s)$, the output speed is $\Omega(s)$, and we could achieve a controller or a control system with a simple proportional controller. Which basically takes the error between what we want as the output speed and what is the measured output speed. So, error is $\Omega_d - \Omega$ and then it is multiplied by this constant proportional gain, and we get this voltage. So, this we had looked at and looked at in detail. And then we showed that this kind of control system, a simple proportional controller, can be used to track this desired Ω_d, in spite of changes in the internal parameters which are basically K and F because we are interested in s tending to 0, in the steady state, where this inertia J does not play any role. And I had also showed you what happens when you have $T_d = 0$, start with $T_d = 0$. This we had discussed in great detail earlier and the main result was that the change in Ω which is the output which is given by $\delta\Omega'/\Omega'$, basically this is the percentage change in the output speed. It can be written in terms of a percentage change in the internal parameters where K_0 is the internal parameter which is K/F and Ω' is the unperturbed output speed. So, basically Ω' is $K_0 K_p / (1 + K_0 K_p)$ into Ω_d and if you recall we had chosen $K_0 K_p$ as much greater than 1. So, this is approximately equal to Ω_d.

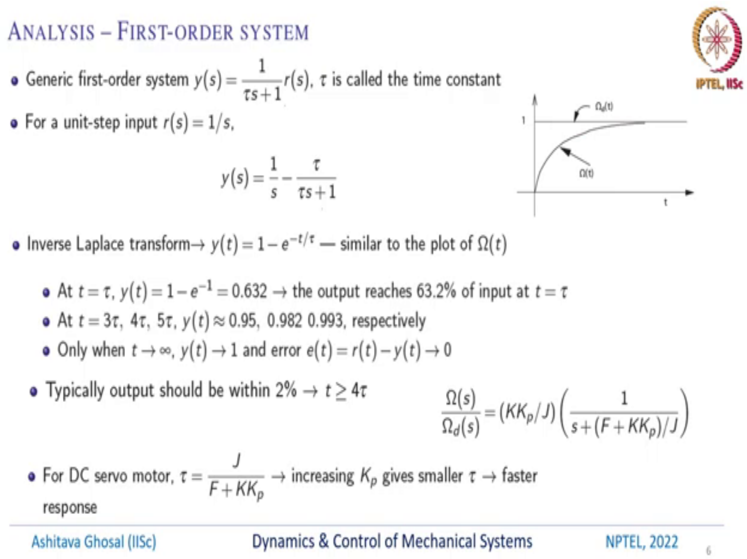
So, the main take away from this expression is that any $x\%$ change in K_0, which is K / F, will result in $(1 / 1 + K_0 K_p)$ into $x\%$ change in the unperturbed output speed, which is roughly the same as what we want, Ω_d. And this is because we had $1 + K_0 K_p$, we choose K_p such that $1 + K_0 K_p$ is much, much greater than 1. And, hence the change in output is greatly reduced by the feedback at least as t tends to ∞ or at least in the steady state.

When we looked at the effect of the disturbance, so, we have this disturbance which is coming in at the input. So, the controller output now, is u or voltage and then the input to the plant is both the disturbance and the voltage. So, again when we looked at the effect of the input disturbance as this disturbance T_d as s tends to 0, I showed you that we can derive an expression for the output speed which is Ω as some $(K_0 K_c / 1 + K_0 K_c) \Omega_d$. K_c is nothing but this controller transfer function - we have intentionally chosen a different symbol. Because now, we want to look at the controller transfer function when both a disturbance and internal parameter changes are happening. So, going back Ω is $(K_0 K_c / 1 + K_0 K_c) \Omega_d + (K_0 / 1 + K_0 K_c) (T_d / K)$.

So, if I choose $K_0 K_c$ much greater than 1 or $K_0 K_c$ much greater than K_0 / K, finally K_c much, much greater than $1/ K$, you can see that the effect of the disturbance T_d is also reduced. So that is the takeaway, that if I choose the controller gain K_c in a particular way, which is K_c much, much greater than $1/ K$, the effect of T_d is reduced and of course, the effect of internal parameter change, which is δK_0 where remember K_0 is K/F, that is also reduced.

So, we had seen that a high gain controller reduces the effect of the internal parameter change and external disturbance. Why do we call high gain? Because the controller gain is much, much greater than 1. So, $1 + K_0 K_p$ is much, much greater than 1. So, a high gain controller basically negates the effect of the external disturbance and the internal parameter change on the output. At least in steady state and at least for the simple first order system.

(Refer Slide Time: 00:07:59)



So, let us look at a generic first order system which is given by $y(s)$ which is the output and the transfer function between the output $y(s)$ and the reference input is $1/ (\tau s + 1)$. So, this is like a very simple generic form of a first order system and τ is called the time constant. Instead of 1 there could be some other constant but that does not matter so, we are really interested in the nature of a generic first order system.

So, for a unit step input which is, $r(s) = 1/ s$, I can obtain $y(s)$ is $1/ (\tau s + 1)$ into $1/s$, and then I can use the method of partial fractions and rewrite it as $1/s - \tau / (\tau s + 1)$. This is a very straightforward trick to factorize the denominator and write it as partial fractions. The inverse Laplace transform of this $y(s)$, will give me $y(t)$ and we can see that this is 1, $1/ s$ gives you 1, and $\tau / (\tau s + 1)$ will give you $e^{-t/\tau}$. And this is very similar to what we had seen earlier, remember when we applied a voltage to the motor and this voltage was proportional to the error -- $K_p (\Omega_d - \Omega)$, and if I had applied the step input, the output will reach this $\Omega_d = 1$ in some time and this is an exponential plot. This expression of $y(t)$ we have seen earlier in the case of a DC motor driving a single link.

We can look at this plot and see what are some of the important points in this plot. So, for example at $t = \tau$, we will get $y(t)$ is $1 - e^{-1}$ - this is 0.632. So, the output $y(t)$ reaches 63.2% of the input at $t = \tau$. This τ is also sometimes called as a time constant. So, it is basically the time required to reach 63.2% of a unit step input. At $t = 3\ \tau$ or $4\ \tau$ or $5\ \tau$, $y(t)$ can be again computed from this expression $1 - e^{-t/\tau}$, and you can see that y will be approximately 0.95, 0.982 and 0.993. So, only when t tends to infinity, $y(t)$ will tend to 1 and the error which is

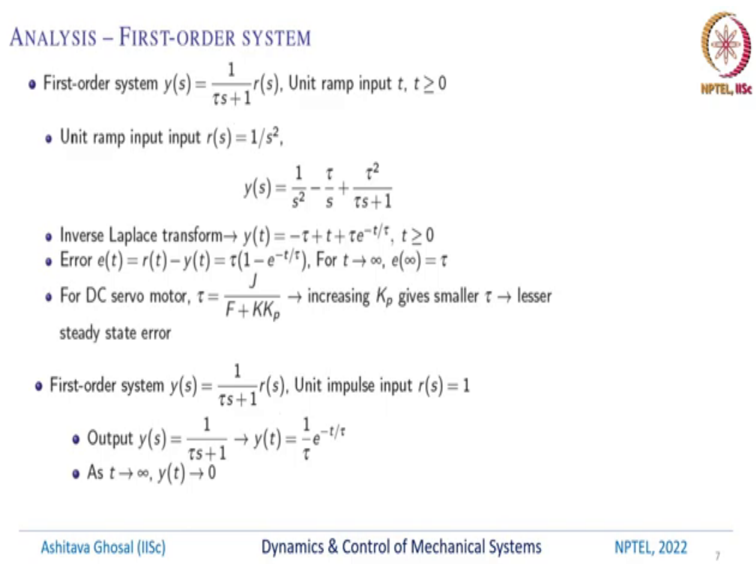
$r(t) - y(t)$ will go to 0. And you can see that in this plot for that case when we looked at the speed of rotation of the link driven by a DC servo motor.

Typically, the output should be within 2% of the $\Omega_d\,(t)$ which is 1. So, if it is within 2% so, basically, we should have t sort of greater than or equal to 4 τ. Because you can see when t is 4τ, we get 0.982 which is within 2% of the desired quantity which is $\Omega_d = 1$. This is an important observation; this is sometimes used in what is called a settling time later which we will see soon.

So, one more useful piece of observation that if you look back at the transfer function between the output Ω and Ω_d,, from which this plot was obtained. We could see that it was given by

$K\,K_p\,/J$ and this is $1/\,(s + F + K\,K_p\,)/\,J$. Please, go back and see your notes J is the inertia K_p is of course the controller -- proportional controller K_p, K was a constant, F was also some friction term which was also a constant. So, what we can see is for the DC servo motor example, this τ is $J\,/\,(F + K\,K_p\,)$. This is $1/\,(\tau\,s + 1)$ so, if you divide take out $(F + K\,K_p)/\,J$, so, you will have τ as $J\,/\,(F + K\,K_p)$. Hence, as K_p increases τ becomes smaller and if τ is smaller it reaches this desired trajectory faster, quicker. So, τ smaller gives a faster response and this is an important observation, and we can see that by changing this proportional gain or this controller gain K_p, we can make the output reach the desired reference trajectory faster.

(Refer Slide Time: 00:13:33)

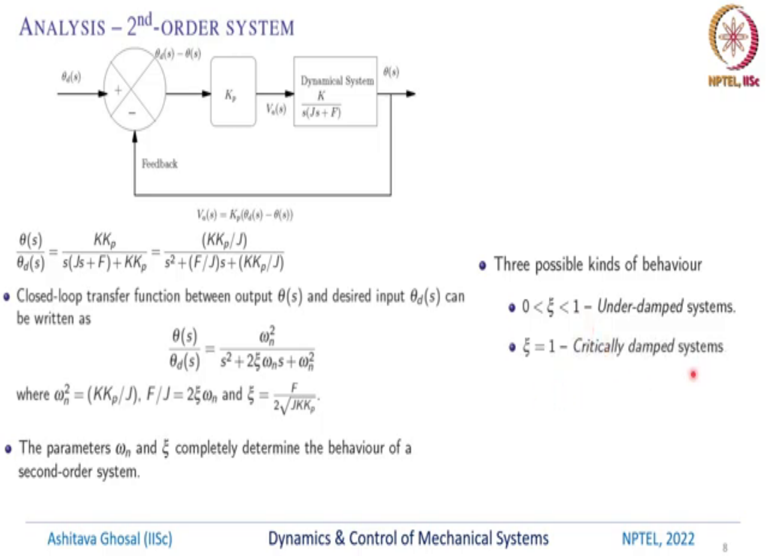


So, let us continue a little bit more with the first order system which is $y(s) = 1/(\tau s + 1)$ into $r(s)$. Now, instead of $r(s)$ being a step input, let us see what happens when you give a ramp input? Ramp input is nothing but $r(t) = t$. So, it is a slope linear curve with 45 degree slope. So, if you plot $r(t)$ versus time it is a line passing through the origin at 45 degrees which is $r(t)$ is t and for t greater than or equal to 0, before that it is 0.

So, for unit ramp input $r(s)$ is $1/s^2$ - the Laplace transform of t is $1/s^2$. And again, we can find what is $y(s)$ by method of partial fractions? We will get one term which is $1/s^2$, there is a term which is $-\tau/s$ plus another term which is $\tau^2/\tau s + 1$. And the inverse Laplace transform of this will give $y(t)$ which is $-\tau + t + \tau e^{-t/\tau}$. So, the error which is $r(t) - y(t)$ is $\tau(1 - e^{-t/\tau})$. What can we see from this expression? What you can see is as t tends to infinity, so, this $e^{-\infty}$ will go to 0 and hence the error as t tends to infinity will become τ. So, again what you can see is, since τ was $J/(F + K K_p)$, if you increase K_p which will give smaller τ and hence a lesser steady state error.

So, $e(\infty) = \tau$ means even at t tends to ∞, even in steady state there will be a small error. And that is that is τ and I can reduce this τ by increasing K_p. So, the last type of input which we can think of for $r(s)$ is what is called as a unit impulse. In that case $r(s)$ is 1. So, if you have a first order system subjected to a unit impulse, so then we can easily find what is $y(s)$ which is $1/(\tau s + 1)$. And from inverse Laplace transform you can find $y(t)$ is $(1/\tau) e^{-t/\tau}$. So, as t tends to infinity, $y(t)$ goes to 0. So, the response to an impulse input for a first order system is steady state error is 0. The response or the steady state error for a unit ramp input is τ and for the step input again as t tends to infinity the error is 0.

(Refer Slide Time: 00:17:00)



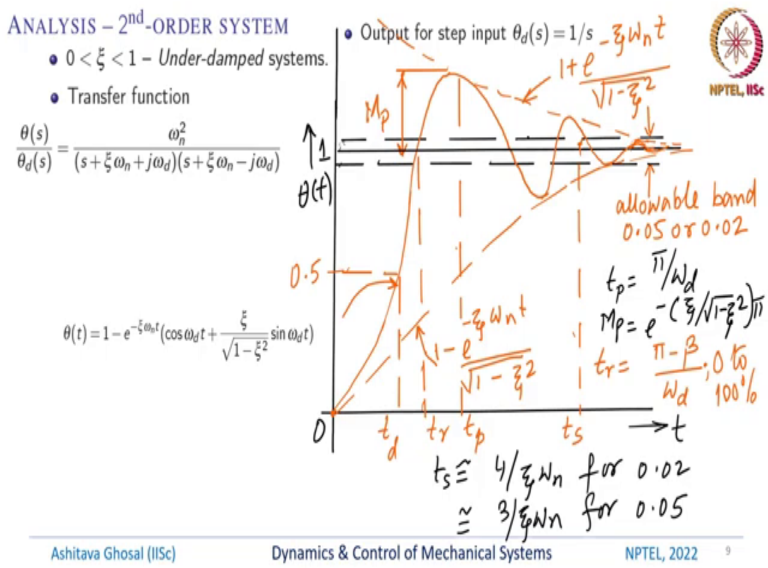
Let us continue and look at the second order system and again we had seen this earlier. In the second order system example, we said that we were interested in the rotation of the link, rotation of the motor shaft. So, in that case we had a second order differential equation so, it was something like $J\ddot{\theta} + F\dot{\theta}$ is some $K V_a$. And the transfer function for the second order system was $K/ s (Js + F)$.

The input is some voltage, and the output is θ and again we have this proportional controller and then we measure $\theta(s)$, and then we subtract it from the desired θ_d and we will get some error which is $\theta_d - \theta$, and the output of the controller is K_p into this error or $K_p(\theta_d - \theta)$. So, the closed loop transfer function between output theta and this reference trajectory θ_d can be written as $K K_p / [s (Js + F) + K K_p]$. And we can do a little bit of simplification and we can write this as $(K K_p/J) / [s^2 + (F/J) s + (K K_p/J)]$. And we had shown that this can be rewritten in terms of natural frequency and damping. So, $\theta(s)/ \theta_d(s)$ is some ω_n^2, ω_n^2 is $(K K_p/J)$, divided by $(s^2 + 2\xi\omega_n s + \omega_n^2)$, where ξ is the damping and ω_n is the natural frequency.

Just by comparing these two, you can see that the damping is nothing but $F / 2\sqrt{(J K K_p)}$. So, as discussed earlier and again you can see it here, the second order system can be completely determined by this ω_n which is natural frequency and the x. So, this is a canonical form of describing a second order system in terms of some natural frequency and damping.

There are three possible kinds of behaviour of a second order system, and we had looked at this earlier, please go back and see your earlier lectures. There is one which is 0 < x < 1. So, the x is between 0 and 1, these are called under-damped system, so, the damping coefficient is between 0 and 1. You can also have x = 1, these are critically damped systems and then you can have x > 1, these are over-damped systems.

(Refer Slide Time: 00:20:01)



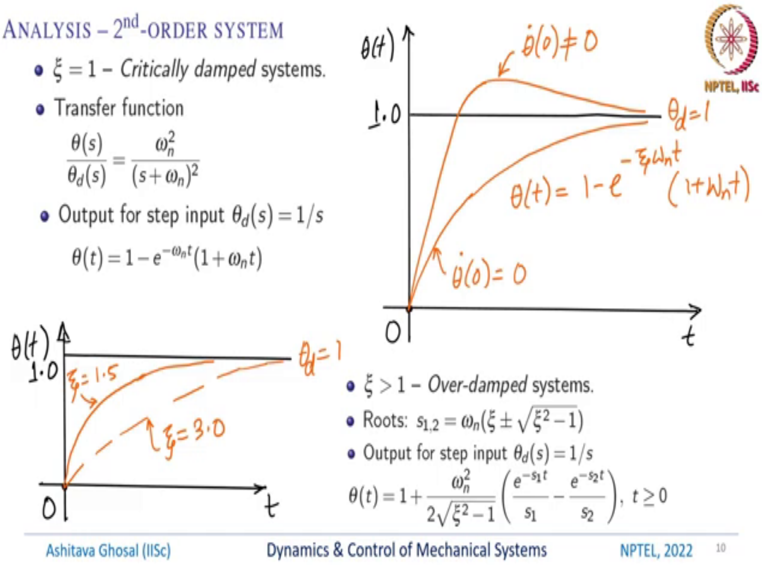
So, let us look at the case, sort of interesting case when x lies between 0 and 1 or the under-damped system. The transfer function closed loop transfer function can be written as θ / θ_d is ω_n^2, now, the denominator polynomial can have two roots and in general it can be written as (s + x ω_n + j ω_d) and (s + x ω_n – j ω_d), and we had discussed this earlier. And I am just repeating the figure once more, ω_d is called as the damped natural frequency. Like ω_n is the natural frequency x is damping and ω_d , if you remember is ω_n Ö(1 – x^2). And for a step input so, if I give θ_d as 1/s. So, this is θ_d, the inverse of this is θ (t) is 1, the output θ (t) can be solved for again by using partial fractions and inverse Laplace transform. And we will get something like 1 – $e^{-x\,\omega_n t}$ (cos ω_d t + x /Ö(1 – x^2) sin ω_d t). So, this is this curve, and you can see that this is going up above θ (t) = 1. And then it will oscillate and then slowly converge to this θ (t) = 1 and as we discussed this curve lies between two exponentials. So, these two exponentials are asymptote to this θ (t). And these exponentials are $1 + e^{-\xi\omega_n t} / \sqrt{(1-\xi^2)}$ and

$1 - e^{-\xi\omega_n t} / \sqrt{(1-\xi^2)}$.

So, this is the general response of a second order system to a unit step input, and as discussed earlier there are several interesting or important parameters in the second order system. One is something called as the peak overshoot, how much the output θ (t) crosses θ (t) = 1? This is given by M_p. Then the other useful thing is once it starts oscillating and dying down because of damping. Once it goes into this band, which is an allowable band, and this is either 0.05 or 0.02 -- this is called as the settling time. So, once it reaches inside this band it never comes out. So, this is in some sense, you need to wait for this time t_s, such that the output is within some allowable limit of the whatever we want, the desired is θ (t) = 1.

And this t_s, is given by $4/\xi\omega_n$ for 0.02 band and $3/\xi\omega_n$ for 0.05 band. Additionally, there are these time at which we hit the peak which is called t_p which is given by π/ω_d. The value of this M_p, which is peak overshoot, can also be obtained in terms of ξ, and also, we can obtain something called as a rise time, the first time it reaches this θ (t) = 1. And there are various definitions of rise time -- in this example I am showing you rise time for 0 to 100% and this is given by $(\pi - \beta)/\omega_d$, β is some angle which we had discussed earlier. So, β is, $\theta = \sin^{-1}\xi$ and $\theta + \beta$ will give you 90 degrees.

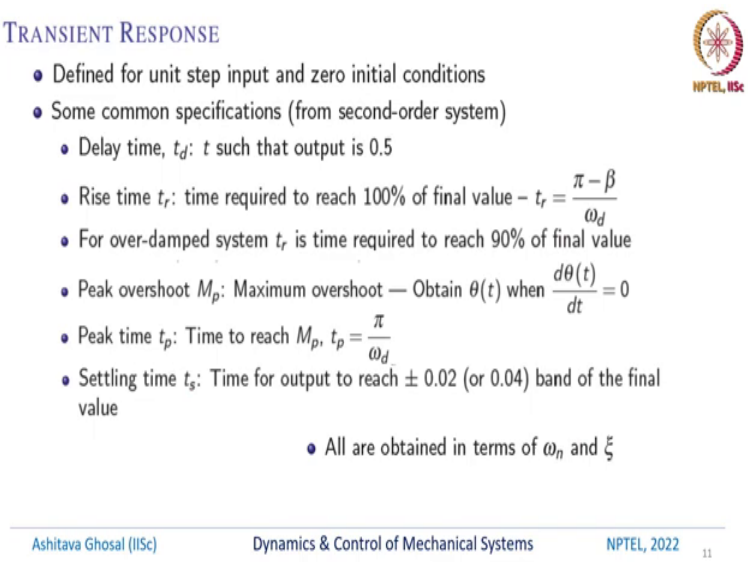
(Refer Slide Time: 00:24:18)



If you have over-damped system, xi greater than 1 or critically damped system which is $x = 1$ then there is no oscillation. So, when you have critically damped it can cross once but nevertheless it will settle down to $\theta_d = 1$. The equation of $\theta(t) = 1 - e^{-x \omega_n t}(1 + \omega_n t)$. So, you can see that there are no cos and sin terms -- there is no oscillations.

Likewise, if you have over damp system which is $x > 1$, then you have two roots which are ω_n (x ±Ö (x^2 –1), and the output $\theta(t)$ is some $e^{-s_1 t}$ and some $e^{-s_2 t}$. So, it is a sum of two exponentials again there are no oscillations, in the case of critically damped or in the case of over-damped. This is more or less what the second order system can do in general. It can oscillate and slowly die down to the desired quantity or it can go up as an exponential and reach that desired quantity in some time.

(Refer Slide Time: 00:25:45)



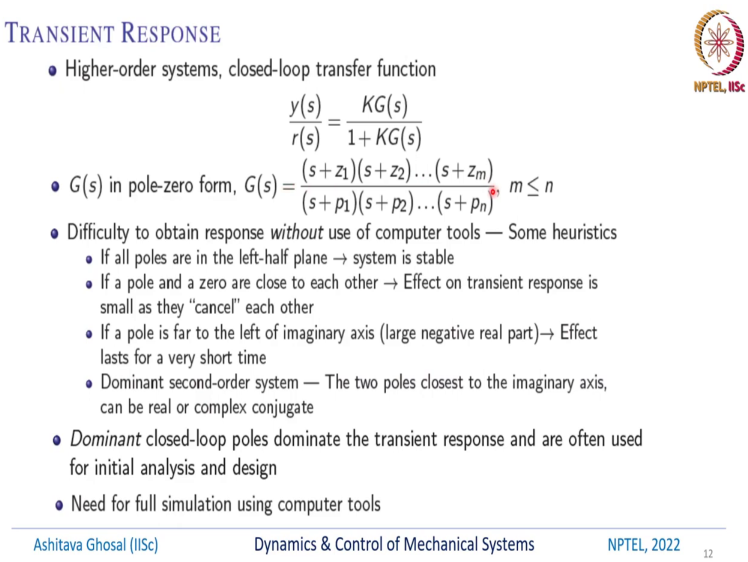
In order to look at second order systems or for that matter any or systems, we need to look at what is called as a transient response. The transient response is most of the time defined for a unit step input and zero initial conditions. There are some common specifications, and these are all derived from a second order system, and one of the reason why we look at second order system is because we can easily derive the solution and also we can see exactly what is happening in this case of second order system. We can physically see this peak overshoot, settling time and we can easily visualize all these common specifications. As I said there could be something called as the delay time and this is the time such that the output is 0.5. Remember, we have a unit step input so, once it reaches half the input, that time is called delay time.

You can also have rise time which is time required to reach 100% of the final value. If it is an overdamped system rise time is sometimes defined as time to reach 90% of the final value. So, for under damped t_r is $(\pi - b)/\ \omega_d$ as I have shown you earlier. The peak overshoot, which is the maximum overshoot, can be obtained by finding $\theta\ (t)$ when $d\ \theta\ (t)/\ dt = 0$. So, remember it reaches the maximum and hence we can obtain the derivative of this θ as a function of time equated to 0, and we can find the time such that this happens and then we can put that time into the equation of $\theta\ (t)$ and find out the peak overshoot. So, the time when $d\ \theta\ (t)/\ dt = 0$ is given by $\pi/\ \omega_d$, and we can also find out what is M_p from the substituting this t_p into the equation of $\theta\ (t)$.

The settling time is the time to reach the output within a certain band. As I have discussed, and I have shown you in the figure earlier. So, if it is ±2%, the value of theta within this ±2% is also called settling time. We can also have ± 0.05. So, in one case it was $4/\ x\ \omega_n$ and in one case it was $3/\ x\ \omega_n$. So, if you have 0.02 you have to wait longer.

The important thing is in all of these second order system, there are only two parameters. There is natural frequency and there is damping and all this delay time, rise time, settling time, peak overshoot, everything can be obtained from the natural frequency and damping.

(Refer Slide Time: 00:29:05)



If you have a higher order system, the closed loop transfer function can be obtained as $y(s)\ /r(s)$ which is $KG\ /\ 1 + KG$. So, you can see that the K is the controller G is the plant or the transfer function of the plant is $G(s)$. The closed loop between output y and reference input r can be written in this form.

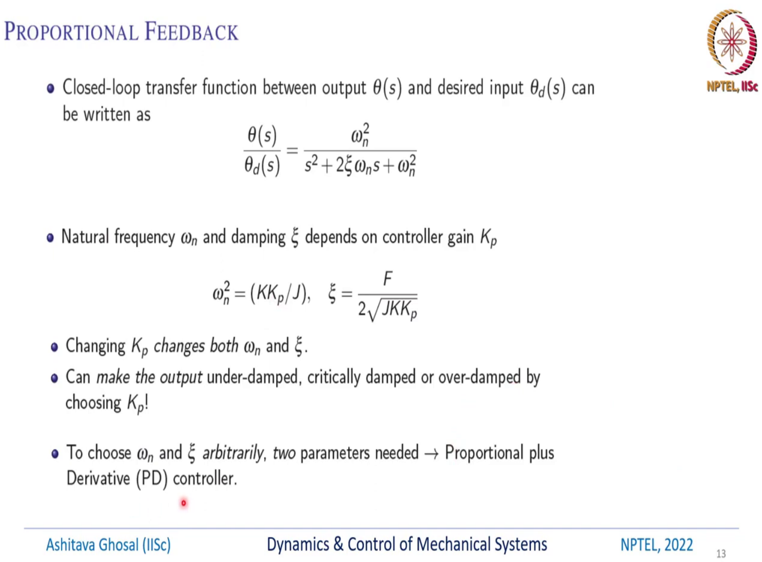
Often this $G(s)$ is given in what is called as a pole-zero form. And we had looked at this pole-zero form, when we looked at root locus. So, $G(s)$ could be written as $(s + z_1)$ $(s + z_2)$ all the way till $(s + z_m)$ and the denominator is $(s + p_1)$ $(s + p_2)$… $(s + p_n)$. Remember, $m <= n$. So, this z_1, z_2, z_m are called the zeros and p_1, p_2, all the way to p_n are called the poles of the transfer function $G(s)$ and this is the pole – zero form.

It is reasonably difficult to find the response of such a complicated system. Especially when you have more than two poles, without using computer tools. So, there are some heuristics or some thumb rules which we can use. So, if all the poles are in the left half plane, the system is stable. So, this we had seen earlier when we looked at root locus. If a pole and a zero are close to each other the effect on the transient response is small because in some sense they cancel each other.

If a pole is far to the left of the imaginary axis, basically a large negative, real part, the effects last for a very short time -- because it is like $e^{-a\,t}$ and if a is large $e^{-a\,t}$ will go to 0 very quickly. In all these, transient response of higher order system, most of the time we look at what is called as a dominant second order system. So, we find the two poles which are closest to the imaginary axis. They can be real or complex conjugate and these two poles most of the time can be used to sort of specify what is the transient response. Because the dominant second order system, we can again find out what is peak, overshoot settling time, rise time all the various things which we had discussed, and if all the other poles are much further to the left, their effect will die down very quickly, and the transient response will be sort of dominated by the two poles which are closest to the imaginary axis. And this dominant closed loop poles, as I discussed or at least I have argued, they dominate the transient response, and these are often used for initial analysis and design. If I know that these are the two poles, I can quickly see what is M_p, t_s, rise time, various things, and then I can analyse the system. Then I can design the system, as I will show you in a short while, and then we can design a controller. And then finally, what we can do is we can look at the entire system with all the poles and zeros and we can use a computer tool to find the response of the system, both the transient and steady state response of the system.

For the full complex higher order system, we need some simulation tools, and these simulation tools are nowadays very readily available. For example, Matlab is a software tool which is available to all the candidates in this NPTEL course and it is easily available, otherwise, also. We can use Matlab or other software tools to find the transient response or the steady state response of a higher order system which is given in this pole - zero form or any other form. If it was given as a polynomial also Matlab can handle it.

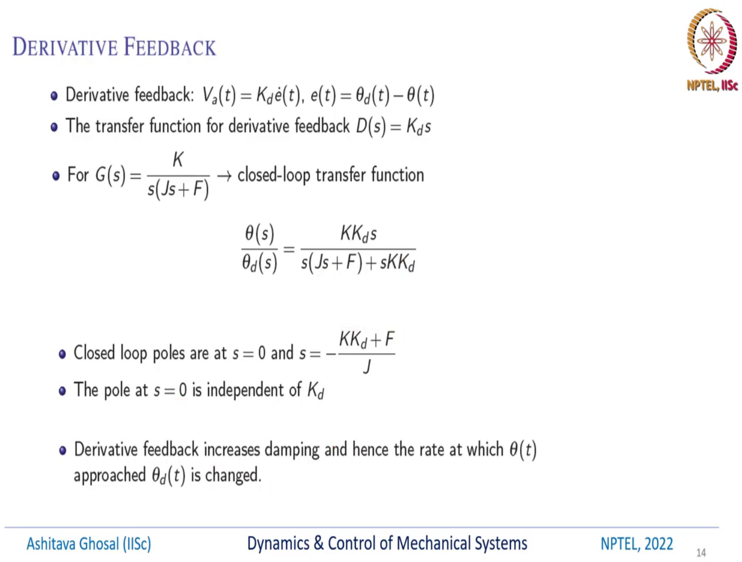
(Refer Slide Time: 00:33:44)



So now, let us come to the control part. So, as I said, we can have what is called as proportional feedback. So, basically, the closed loop transfer function between the output θ (*s*) and the desired input θ_d or the reference trajectory θ_d, can be written as $\omega_n^2 / (s^2 + 2$ x $\omega_n s + \omega_n^2)$. I am repeating this many times, because the second order system is very, very fundamental and is the basis of understanding everything.

The natural frequency ω_n and the damping x depends on the controller gain K_p. I had derived this, and I am repeating it once more. So, ω_n^2 is KK_p / J and x is F / 2 Ö ($J KK_p$). If I were to change K_p which is the proportional gain or the proportional feedback gain, it automatically changes both ω_n and x. So, if I increase K_p, natural frequency will increase and automatically x will sort of decrease - x will also, K_p, increases, so, x will decrease – correct? So, the important point is changing K_p changes both natural frequency and damping. So, we could change K_p, and so, suppose you have an under damped system -- x is between 0 and 1 -- I could easily make it critically damped or over damped by choosing K_p. But I cannot choose or determine ω_n and x arbitrarily. If I want to change both K_p, in both of these expressions, both natural frequency and x will change on it is own. But suppose I want to independently change natural frequency and x, I cannot do only by changing K_p. I need two parameters and this is the basis of proportional plus derivative controller. This is a very commonly used controller which is called as a PD controller.

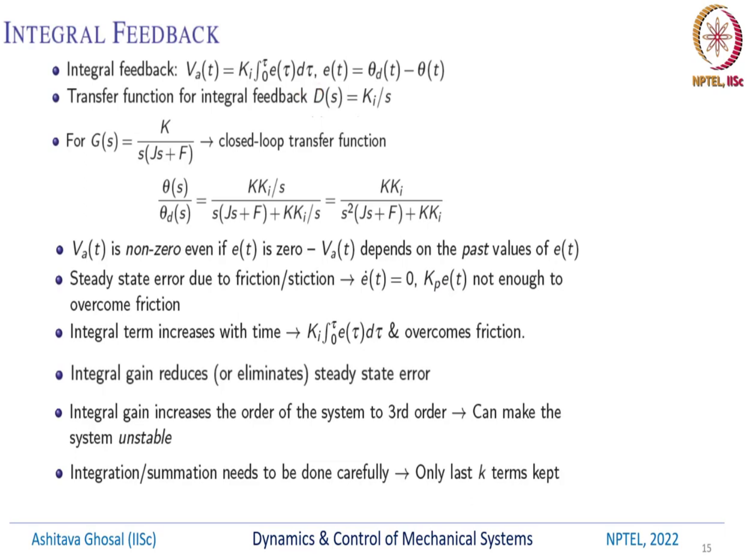
(Refer Slide Time: 00:36:22)



So, let us look at what happens when you have a PD controller? So, when you have a PD controller, the voltage which the plant is seeing is proportional to $K_d \dot{e}(t)$. So, what is e? e is $(\theta_d - \theta)$. If you look at the transfer function of this, of this feedback controller, $D(s)$ is $K_d s$.

So, if you have the transfer function, $G(s)$ which is $K/ s (Js + F)$, and if you have derivative feedback, then $\theta(s) / \theta_d(s)$ is KK_d and the denominator this $s(Js + F) + sKK_d$. So, the closed loop poles are now at $s = 0$ and $s = -(KK_d + F)/ J$. So, the pole at $s = 0$ is independent of K_d, you can see that. So, if I were to increase K_d, it increases damping. So, remember it is e^{-at} and it is like in some sense a pole on the real axis. Hence, if I were to increase, the increases damping and the rate at which $\theta(t)$ approaches $\theta_d(t)$ is changed. So, the derivative controller is different because now, in the numerator we have a $KK_d s$. Remember in the proportional controller, it was KK_p - there was no s. Likewise in the denominator this was $s(Js + F)$ and this is KK_d but now there is an s term also here. So, there is a closed loop pole at $s = 0$. In the case of a proportional controller, the open loop poles were at $s = 0$ and $-F/J$. But if you change K_p, the poles would change. In this case, if you change K_d, the pole at $s = 0$ does not change.

(Refer Slide Time: 00:38:48)



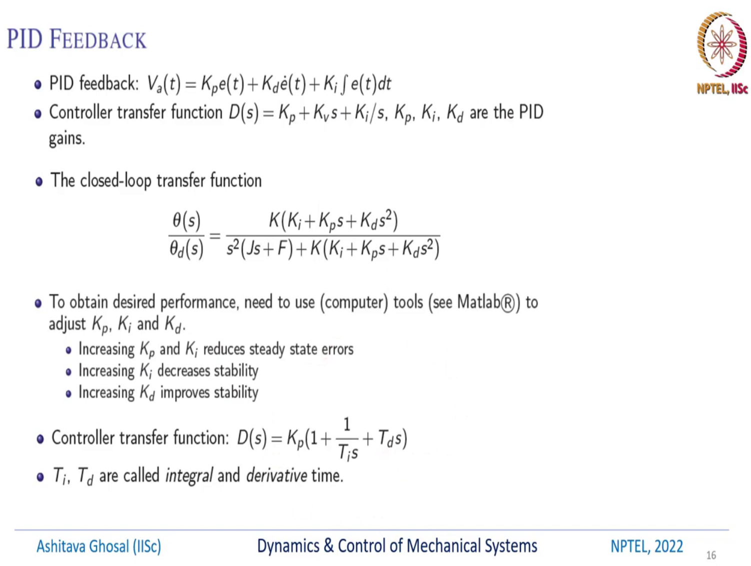
We can also have something called as an integral feedback. An integral feedback is given by the following equation - the voltage applied is equal to $K_i \int_0^{\tau} e(\tau) d\tau$. We have error which is ($\theta_d - \theta$), so, we integrate this error from some 0 to some time period τ. So, the transfer function for integral feedback is K_i/s and if you look at the closed loop transfer function with an integral feedback, again, $\theta(s)/\theta_d(s)$ is now KK_i/s and the denominator is $s(Js + F) + KK_i/s$. If you simplify this, you will get KK_i but the denominator now has $s^2(Js + F) + KK_i$. One of the major difference for an integral controller, as opposed to both a proportion or a derivative controller, is that the denominator now is third order. So, we have some s^3 term -- this is $s^2 Js$ so, you will get $s^3 J$.

And what is the effect of this integral feedback? This output of the controller which is $V_a(t)$, which is the voltage, which is going into the plant, is non-zero, even if $e(t)$ is 0. How is that? Because when you do integral so, basically you are taking the sum of the past few values of this $e(t)$. Integral is nothing but summation over the last few terms, or actually from 0 to τ, all the terms you keep on summing. So, even if at any instant of time, $e(t)$ is 0, if the $e(t)$ in the previous time instance were non-zero, this you will get some term, you will get some voltage. So, integration depends on summation, is an approximation of summation and this V voltage of which you are applying to the system, depends on the past values of $e(t)$.

So, the steady state error due to friction and stiction will lead to some $\dot{e}(t) = 0$ and let us assume that we have some friction at this bearing friction at the joints. Go back to that remember the example of this link being rotated by a DC motor. So, we had some friction, and this friction is reasonably large and this quantity $K_p\, e(t)$, which is like the voltage, which is dependent on the proportional controller which is $K_p\, e(t)$, is not enough to overcome that friction. Hence, at some stage the link will move but then there is some significant friction, and it comes to a stop. So, $\dot{e}(t)$ is 0, so, the derivative part does not contribute anything, and this proportional part is not enough but the integral controller will still work because it keeps on remembering what is the past error. It keeps on summing the past errors and then eventually $K_i \int_0^{\tau} e(\tau) d\tau$, this integral term, will become large enough to overcome the friction. This is the reason why integral term is used very often. The integral gain reduces or eliminates the steady state error, because after a while this voltage which is the integration of this will be enough to overcome the friction.

What is the disadvantage? The integral gain increases the order of the system to third order. And we have seen that in third order you will have three poles and it is entirely possible at least we have seen one example earlier that the branch of the root locus can go to the right half plane for some values of gains, so, it can make a system unstable. A second order system, on the other hand, is never unstable because the poles will stay to the left half plane it will go to infinity along this vertical line. Whereas if you have a third order system then the poles can go to the right half plane and hence can make the system unstable. This integral control needs to be done very carefully. So, this integration gain K_i needs to be chosen small and carefully. Additionally, this integration is approximated by sum, and we cannot keep on summing all the way to the beginning of time at $t = 0$. So, most of the time we sum over the last k terms so that is one way to make sure that it does not become unstable.

(Refer Slide Time: 00:44:15)



The PID feedback is now given in this form. So, the voltage applied to the plant is proportional to the error so, $K_p\ e\ (t) + K_p\ \dot{e}(t) + K_i \int_0^{\tau} e(\tau)d\tau$ -- it is proportional to the derivative of the error and also proportional to the integral of the error, that is where this PID come from. \`P' means proportional, \`I' means integral and \`D' means derivative.

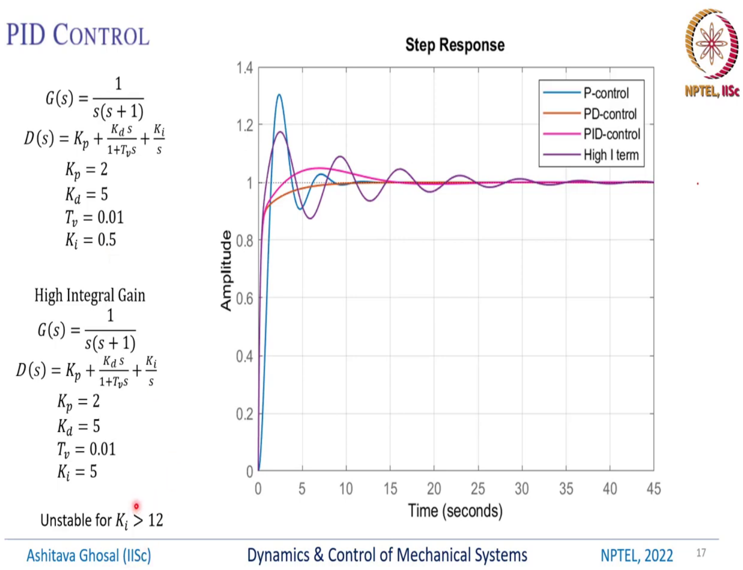
The transfer function of this can be written as $K_p + K_d\ s + K_i\ /s$, and this K_p, K_d and K_i are called the PID gains. The closed loop transfer function can be written as $\theta\ (s)/\ \theta_d\ (s)$ and then in the numerator we have $K\ (K_i + K_p\ s + K_d\ s^2)$, and in the denominator we have $s^2\ (Js + F)$ and then again, $K\ (K_i + K_p\ s + K_d\ s^2)$. So, you can see that the denominator is third order, and the numerator is second order -- here is an s^2, whereas in the denominator it is s^3. And to obtain desired performance we need to use computer tools.

It is not very straightforward to play around with these three gains, K_p, K_d and K_i. We have to search, in some sense in three dimensions of K_p, K_d and K_i, and we can use computer tools -- again Matlab provides computer tools to choose these PID gains, so as to achieve a desired output -- to desired both transient response, as well as steady state response.

A few heuristics if you were to increase K_p and K_i, it reduces steady state errors. As you saw that the steady state error is less when K_p is increased, and it will be 0, if K_i is increased by some amount. I am not going to go into the details, but K_i will ensure that the steady state error is 0 for some lower order systems. However, increasing K_i decrease stability because once you have `I' feedback then it makes the system third order, and it decreases stability. Increasing K_d improves stability because it makes the system more damped and hence there is less oscillations. So, we need to play around with this K_p, K_i and K_d such that we get the desired steady state error, as well as the desired stability.

Sometimes instead of using K_p, K_d and K_i, we can also write the controller transfer function as K_p (1 + 1/ T_i s + T_d s). So, instead of K_p, K_d and K_i, now we have only one K_p but then there are these two time constants T_i and T_d. So, these T_i and T_d are sometimes called as the integral and derivative time. So, both these things are more or less exactly the same, except it is written in a different form. You can think of K_p / T_i as K_i and K_p T_d as K_d. So, this is just another way of describing the controller transfer function - both are exactly the same.

(Refer Slide Time: 00:48:16)

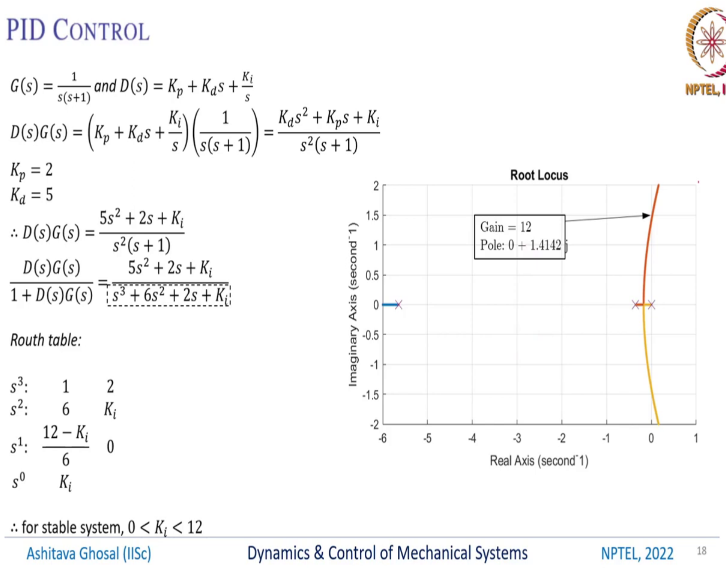


Now, let us look at an example, so, we go back to our usual familiar example which is G (s) is 1/ s (s + 1). This is the example of a single link, or a single rigid body connected to a DC servo motor, and we were trying to rotate that link. We want to achieve a desired θ of the motor shaft, and the transfer function was 1/ s (s + 1). So, this is one because remember, K is chosen as 1, friction term F is chosen as 1 and J is also chosen as 1. So, this is just to make life simpler. We could easily have chosen J as some other number F instead of 1 could be something else and so on and K also could have been something else. We want to try and see what happens when you have a PID control? So, the output now is the controller is given by

$K_p + K_d\, s/(1 + T_v\, s) + K_i\, /s$. So, we will see why this T_v comes. This is because we will see little later that we cannot have a term which is just simply K d into s.

In this example, we have chosen K_p as 2, K_d as 5, T_v as 0.01 and K_i as 0.5. So, if you just picked K_p then that is proportional controller and this blue curve this light blue curve is the effect of a proportional controller on this transfer function, $1/\, s\, (s + 1)$. The PID control, when you have both K_p and K_d is given by this orange line. So that you can see there are no more oscillations and it is damped. The PID controller is this pink line so, you can see that there is the damping is becoming less. So, there is some oscillations, but it is not too much. If you use a high gain in K_i, so if K_i is very large, for example, K_i is 5, then you can see that the oscillations are increasing. So, as I said, the proportional controller is making it damped. I could control this oscillation by using the derivative gain and then if I increase K_i to some larger number so, from 0.5 to 5, again these things are taken sort of arbitrarily, you can see again the oscillations are increasing.

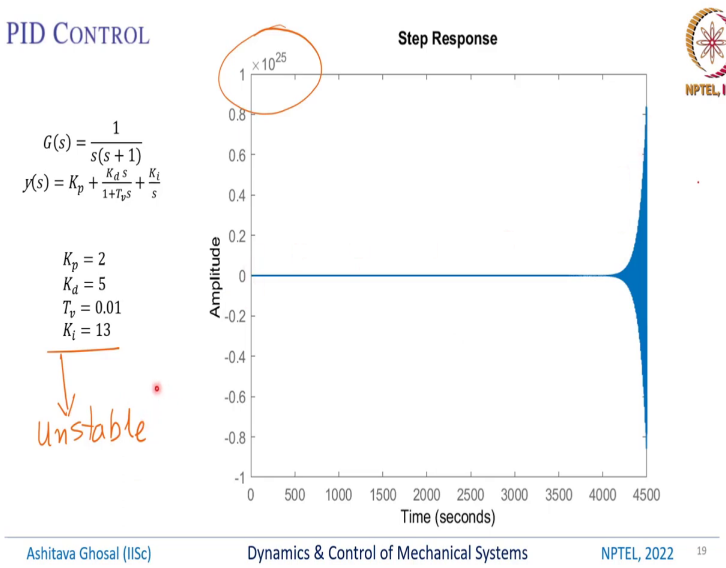
(Refer Slide Time: 00:51:15)



We can also look at the root locus for some values of K_p and K_d. So, the transfer function $D(s)G(s)$ is now given by $(5s^2 + 2s + K_i)$ and divided by $s^2\, (s + 1)$. The closed loop transfer function is $D(s)G(s)/\, 1 + D(s)\, G(s)$ and you can see in the denominator, we have
$s^3 + 6s^2 + 2s + K_i$.

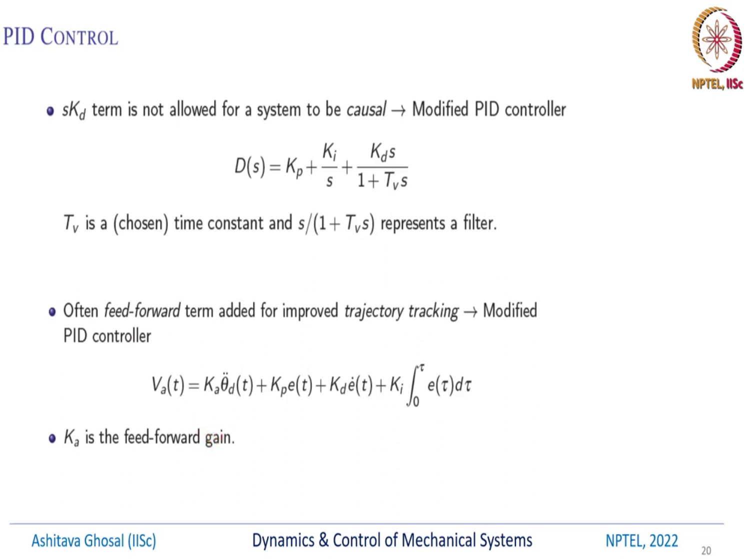
We can create this Routh table so, basically s^3 is 1 and 2, s^2 is 6 and K_i, and then we can find the coefficient here which is $(12 - K_i)/ 6$ and for s^0 it is K_i. So, what you can see is the system is stable from 0 to 12. If K_i is more than 12, then the system becomes unstable. We can plot the root locus for this, and you can see that there are these two poles, and the root locus will go off to the right half plane if the gain crosses 12. And it crosses this j ω line at 1.4142 plus and minus – 2 complex conjugate poles. And it is on the right half plane when the gain crosses 12 and we can also try and plot this and see what is exactly happening.

(Refer Slide Time: 00:52:49)



This is cooked up example, we have this $G(s)$ which is $1/ s(s + 1)$. That controller transfer function is $K_p + K_d\, s/(1 + T_v\, s)$ and this is K_i /s. We have chosen K_p as 2, K_d as 5, T_v as 0.01 and K_i is 13. Remember if K_i is more than 12, it is unstable, and we can plot the step response for such a system. You can see that it is almost very small and then it shoots up. The important thing to realize is that this amplitude is like 1 into 10^{25}. It is very, very large and hence it is unstable. So, the output amplitude of the output θ goes off to basically infinity as you go in time.

(Refer Slide Time: 00:53:46)

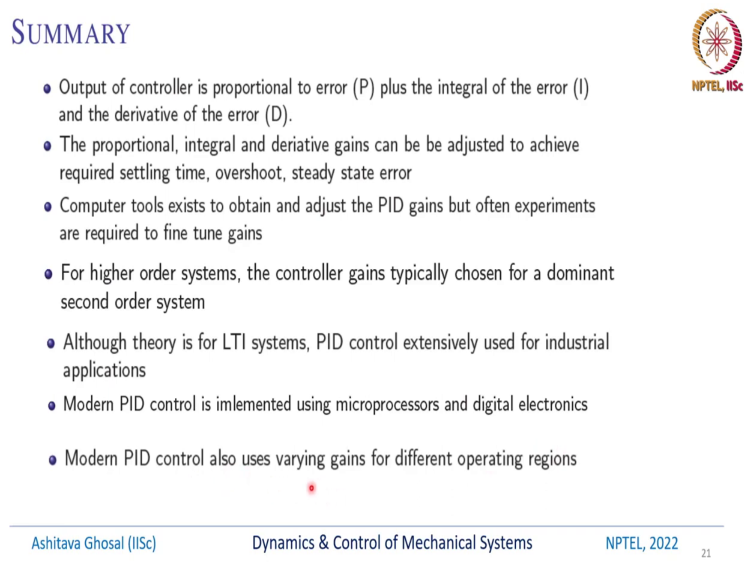


So, as I said, we cannot really use $s\ K_d$. If you have a term or a transfer function which is $s\ K_d$, this is not allowed because the system is not causal. So, remember in root locus we had looked at, if you have the numerator polynomial order greater than the denominated polynomial order, then the system is not physical, it is not causal. So, in order to avoid this problem, we modify the PID controller. Which is basically $K_p + K_i /s$ -- these two terms are fine --now, instead of $K_d\ s$, we have $K_d\ s/(1 + T_v\ s)$. So, this quantity $s/ (1 + T_v\ s)$ is sometimes called as a filter and T_v is a chosen time constant, and this is like a filter. In Matlab you can choose some of the filter coefficients and then you can simulate the system for some different values of T_v -- see in the example which I showed you earlier T_v was chosen as 0.01.

In nowadays, sophisticated controllers there is also a feed forward term which is added. So, the voltage is proportional to the error it is proportional to the derivative of the error, it is proportional to the integral of the error, so, it is $K_p\ e\ (t) + K_p\ \dot{e}(t) + K_i \int_0^\tau e(\tau) d\tau$. But in robots and many other sophisticated, very accurate devices, we also have the desired acceleration.

So, $e\ (t)$ is $(\theta_d - \theta)$, θ_d is the desired rotation of the link, $\dot{e} = \dot{\theta}_d - \dot{\theta}$. So, we need to measure $\dot{\theta}$. In robots and sophisticated instruments, we can plan the trajectory such that we also have the desired acceleration of the output -- so, it is $\ddot{\theta}_d$ is also available. So, in some control systems we can use that information and we have another gain which is called as $K_a\ \ddot{\theta}_d$. So, this K_a is often called as the feed forward gain. So, not only we have a proportional gain, a derivative gain, an integral gain but also a feed forward gain. And in many control systems this is used and it happens, that if you use a feed forward gain, it can track the trajectory better.

(Refer Slide Time: 00:56:41)



In summary, the output of a controller is proportional to the error plus the integral of the error plus the derivative of the error. So, this is a PID controller. The proportional, integral and derivative gains can be adjusted to achieve required, settling time, overshoot or even steady state error. Remember, the integral term can be used to reduce or even eliminate steady state error.

The proportional and derivative gain can be used to reduce or change the peak overshoot and the settling time and other parameters which gives the transient response. Computer tools exist to obtain and adjust the PID gains, but often experiments are required to fine tune the gains. Remember all this design of controllers is for a linear system. Actual systems are never linear.

So, we have a non-linear system with complicated terms, non-linearities and some backlash and friction at the gears which are not constants. So, in those cases we have to do experiments. Nevertheless, these PID gains are a good starting point and then we can tweak the gains to achieve the desired performance of a actual system.

For most higher order systems, the controller gains are typically chosen for the dominant second order system. Again, this is a starting point. We assume that the higher order system is a dump is like a second order system. So, we pick the two dominant poles which are close to the j ω axis, to the imaginary axis in the *s* plane, and we design a controller based on this dominant second order system and then again, we can do extensive simulation and experiments to make sure whatever we have done is correct. So, our initial computation is a good choice to start with.

Although this whole theory is for linear time, invariant system, PID control is extensively used for industrial applications. Because it is very well understood, people know exactly how to change the gains. They have experience in setting the gains and although the system is actually, a non-linear system, it is not a linear time invariant system, even then PID works. So many, many devices in industry still use PID controllers.

Modern PID control is of course, implemented using microprocessors and digital electronics. We have looked at the PID controllers in continuous time. We are using Laplace transforms and we are using continuous time but actually, it is implemented using microprocessors and digital electronics which brings it is own problems. It also brings, it is own advantages. And we do not want to get into this in this course. If you are interested, please look at some textbook on digital control.

Modern PID control, algorithms also use varying gains. So, remember we found K_p, K_d and we are assuming, in this lecture, that K_p, K_d and K_i were fixed. They were constant - so, you must have done some experimentation and then you found some numbers for K_p, K_d and K_i. But in actual practice, in some modern PID control, also uses what are called as varying gains. So, some portion of the operating region, we will use this set of gains in some other operating region, we will use some other gains, and that is shown to work better.