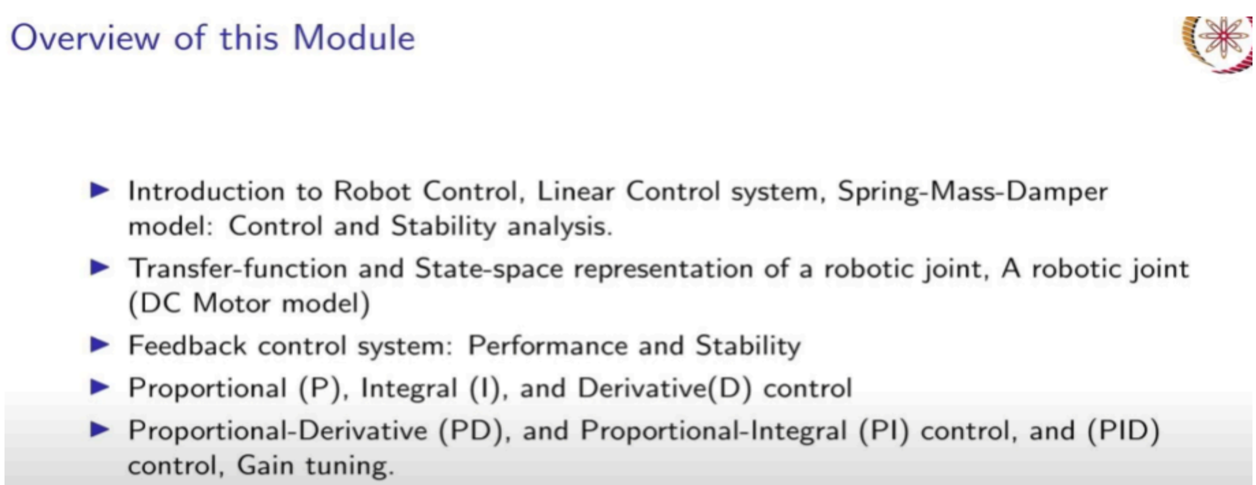
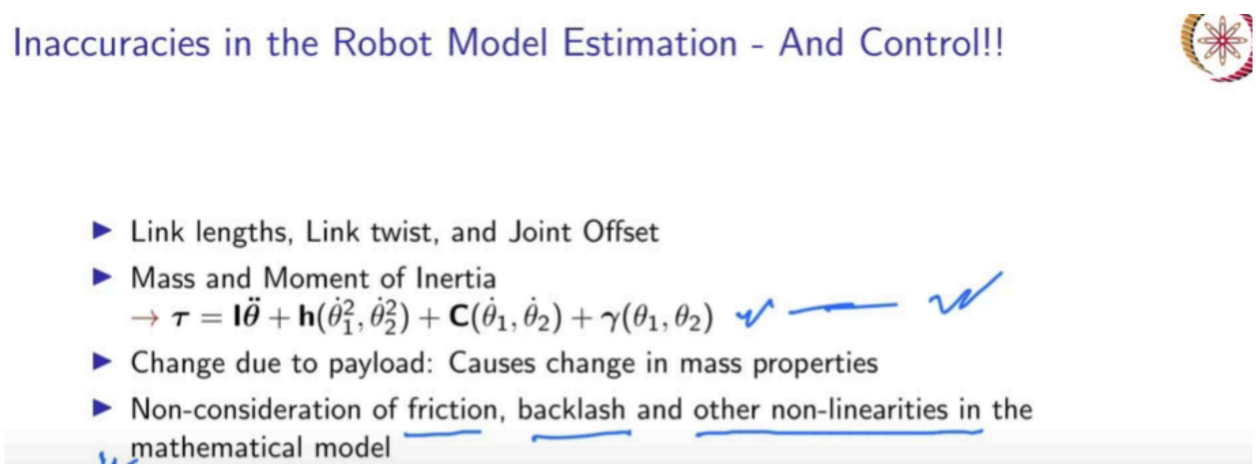
NPTEL Online Certification Courses
Industrial Robotics: Theories for Implementation
Dr Arun Dayal Udai
Department of Mechanical Engineering
Indian Institute of Technology (ISM) Dhanbad
Week: 11
Lecture 43

Introduction to Control: Linear Control, Second Order System

Welcome to the module Robot Control. Now, you are already familiar with the mechanical and electrical hardware of the robot, we have already learnt kinematics. We have learnt dynamics that are necessary prerequisites to make a robot capable of doing some variable motion to do any task. Now, we should be able to run this robot as desired by the robot input commands and not deviate too much from the desired commands. Got it? So, that is the reason the control is there. A robot is to be controlled. So, in this module, we will do some preliminary learning in robot control, because robot control as such is a vast subject. So, over here, because we are targeting to handle industrial robots, we should be familiar with a few terminologies mostly and with the basics of robot control. So, this module has the objective to do that kind of learning over here. So, in this module we will learn how a robot is controlled and that will enable us to understand various concepts. That goes in the background in a controller, which keeps track of any error in the commanded and the actual positions or even in trajectory. So, it keeps track of the error and does necessary corrections in real time while the robot is moving or when it is standing still in any pose. So, let us begin this module and understand what I am trying to say now. So, let us begin.



So, yes, in this module, basically, we will be doing this, particularly in today's class. So, I will be introducing you to robot control, linear control systems, and the Spring-Mass-Damper model. I will discuss and control and stability analysis we will do later on in other lectures and transfer function state space representation of a robotic joint. A robotic joint, basically with a DC motor model, will try to apply to any standard robot. Even if it is a synchronous control robot, a DC motor model is very well applicable because, you see, even a synchronous motor works in a similar way. So, there are a few fundamental equations which does not change much, even in the case of those motors particularly, tau is equal to km into ia. If you have seen it earlier in the actuator model, you must have noticed. Torque is proportional to the current Motor constant was given by some constant, which is known as km. So, you see that fundamental equation that remains the same, and that is the most specific one which actually works over here. So, that is the reason the control we are handling in this class is applicable to most different kinds of robots, which have various other kinds of actuators. Also, the feedback control system will try to understand what it is, what it means and, how it works, how they are implemented in the robot. Performance and stability thing we will discuss Proportional, integral and derivative control. That is the most commonly used kind of control in industrial robots. So, we will familiarise ourselves with this and a combination of PD control that is proportional derivative control, proportional integral control and a combination of all these three, that is, PID control I will discuss- and gain tuning in particular, we will discuss towards the end of this module.

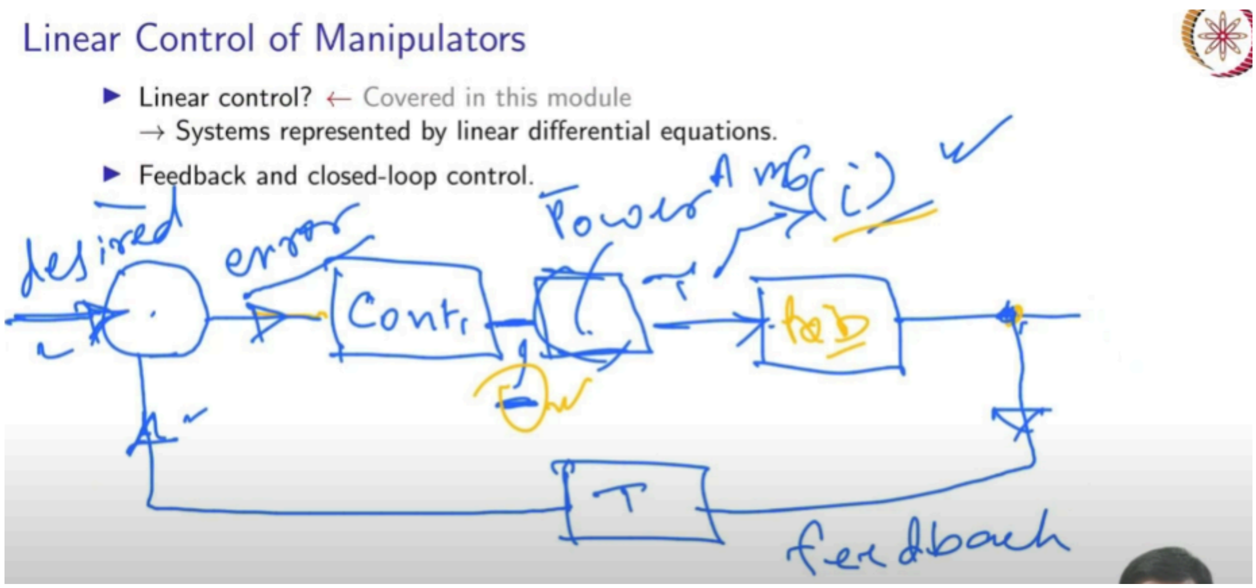


Why are we doing control at all? Because we know if it is an ideal robot if we have designed something in CAD and the manufacturer is able to make it as it is, that is a remote possibility, you see, why? Because there are some manufacturing uncertainties also. And if we think robot, let us say I have given a dimension of 600 mm, so it may be possible that the distance between two joints is not 600 mm. It is 600.5 mm. And similar is the case if we see we have designed two axes to be 100% parallel. Those are the two joints. So, that also may not be possible. There exists a kind of twist between two particular joints next to the next. So, even if we have designed however accurately, manufacturing uncertainties are always there, and that is what is creating this error. So, this link twist is there, even if it is not there in the DH parameter in ideal conditions. So, after manufacturing, there might be some possibility that this is there and even a joint offset. The same is true even for joint offset; there does exist joint offset, even if there is no joint offset in the design itself. So, you got it. So, manufacturing errors are always there, and sometimes, they are not even measurable once it is assembled. So, in that case we have to take all those in control. All those uncertainties are to be dealt with by the controller itself. It cannot be calibrated also. So, by some sort of calibration, we can go again very near to the model or very near to the actual assembly of the robot. But yes, you cannot go accurately. In that case controller takes care of the rest of the uncertainties in the manufacturing of the robot, which has happened. So, that is there and mass, moment inertia you see. The mass and moment of inertia. Those are, again, some parameters which are not known to a very close extent to the design thing or very accurate mass values. We don't know about a link. Also, the same is for moments of inertia because you know there are a few components that go into the robot assembly which were not planned. Quite a lot of time they are not very accurate, also like cables. Cables, however accurate you are when it is manufactured, when it is assembled when it is moving, can be a little saggy, it can be rooted somewhere else. So, accurate design, taking care of that and getting perfect knowledge of cables is very, very difficult. So, that makes, again, it an incorrect model, and that cannot be taken care of by the dynamic equation of motion, which is given here.

$$\boldsymbol{\tau} = \mathbf{I}\ddot{\boldsymbol{\theta}} + \mathbf{h}(\dot{\theta}_1^2, \dot{\theta}_2^2) + \mathbf{C}(\dot{\theta}_1, \dot{\theta}_2) + \gamma(\theta_1, \theta_2)$$

So, it should have a moment of inertia accurately. It should have h and C values, which are calculated based on that. So, they all contain the masses over here, and these components of the torque equation. Over here, it is also there. So, you see, the moment of inertia and masses should be known very accurately. That is again because it is not possible, so it cannot be taken care of alone by this equation. So, this equation can no longer run the robot accurately. So, you need to have some sort of controller that takes care of any error which happens due to unmodelled parameters which are not taken care of over here.

So, yes, then there are changes in the payload. Yes, today, this robot is handling, the robot is sold and used by the payload capacity that it can handle. Let us say it is a five or 20-kg payload capacity robot, and it is not always true that it carries 20 kg. Sometimes it is carrying, let's say, an empty carton which is not 20 kg, and it is just 5 kg. So, that gives you the complete range of the weight that it can handle. And even while picking and placing, sometimes it is picking just 5 kg, sometimes it is taking 20 kg, and you are not adjusting the controller gains. You are not adjusting anything while the robot is running. So, during run time, only the controller has to take care of all these variations that can happen to the payload. You very well know that these cause the mass properties and inertia properties of the robot dynamic model. So, whatever goes into this dynamic model, these masses do affect all the parameters which are here in this equation. And again, this equation never took care of any friction. If it is there in the joint, any backless, if it is there in the gears- transmission gears And any other non-linearities in the mathematical model which is given by this equation, Got it. So, using this is not sufficient. You have to either identify those non-linearities, if it is there, or you have to take care of the controller itself. How does the controller take care of all those uncertainties? So, we will discuss this in today's class a bit and then in this module later on. So, these are some requirements why a robot controller is required.



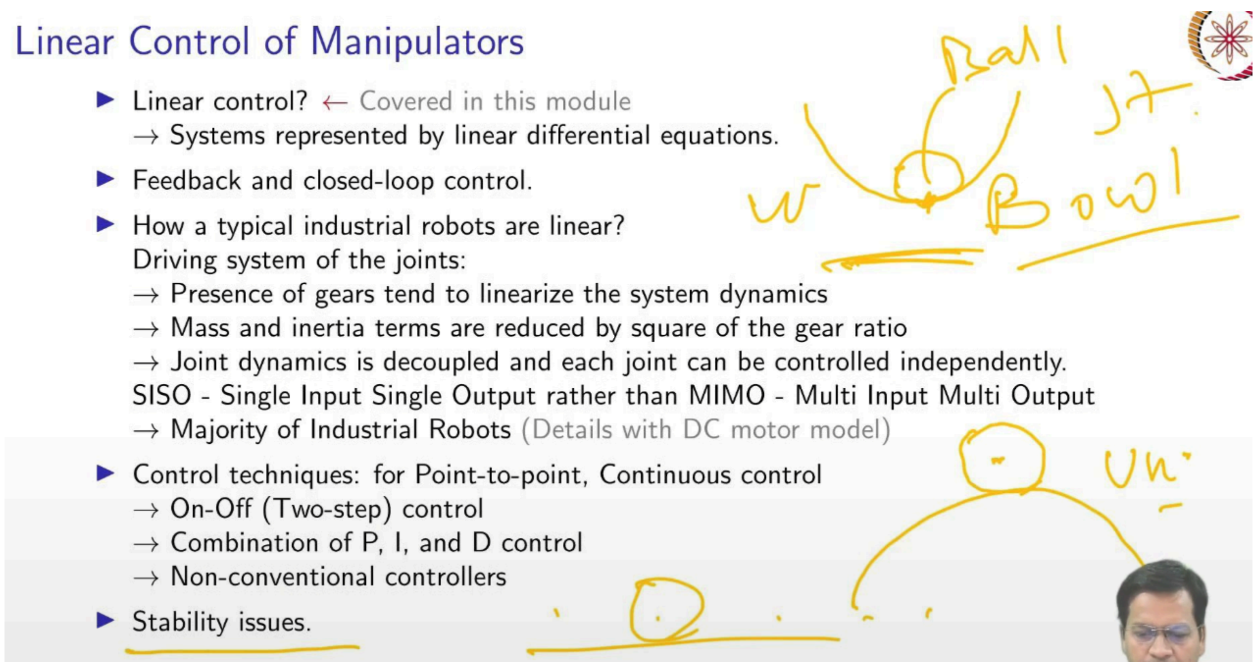
Now, basically, in this module, we will be doing mostly linear control here, and systems are represented mostly by linear differential equations. To be more specific, we will handle mostly second-order linear differential equations because most of the systems that we are handling now can be taken care of well by second-order differential equations, and we should be happy with that. So, I will show you how to use just this work with this limitation, and still, you can do quite a good amount of control of your industrial robot.

Yes, so what is linear control? So, what do you mean by a linear differential equation first of all? So, a linear differential equation may be given, as, let's say, if it is a0(x)y plus a1(x)y dash a2(x)y double dash, and so on, so forth, till an(x)y n is equal to b(x).

$$a_0(x)y + a_1(x)y' + a_2(x)y'' + ... + a_n(x)y^{(n)} = b(x)$$

So, let us say this is the equation where you have y dash, y double dash, y triple dash, and so on, and so forth, till y, n dash. They are all derivatives of unknown function, y, unknown function, y, which is a function of x, y is a function of x. These are all their derivatives. And what are a0, a1, a2, an? They are arbitrary differential functions and need not be linear. They can be, let's say, sin x. They can be log(x), so anything can be here. So, all these kinds. This basically defines your linear differential equation. So, our system, which we are going to handle with this kind of controller, which we are going to discuss in this module system, should be capable of representing this kind of linear differential equation.

We will also be doing feedback or a closed-loop control system. So, what does it look like again? So, let us say you have a robot, and that is commanded by some kind of torque over here. Torque: basically, we don't command directly by torque. Let us say we command by the current that goes to the motors of the robot. So, it gives you some kind of output and this current is fed by a type of power amplifier. Let us say, Here you have a power amplifier that actually converts the small input signal that is the output of our type system which is known as a controller. The controller is here that gives you a small signal which will tell this power amplifier. Now, you feed this current. So, this current is basically decided by the controller output, and this is nothing but a power amplifier because you know this signal is not good enough to directly drive the motors which are there in your robot, which is here. So, you have your robot, which goes here. This is driven by the huge amount of current which is here. But this is nothing but the signal that comes out of the controller. So, now, what goes to the controller? So, basically, you have a feedback line that actually is fed back, like this, through some kind of transducer or some transverse signal conditioning system. So, finally, it comes to a comparator, and this is nothing but an error. And this is the signal. That is the desired signal, The desired signal, whatever position you want to go, whatever set of points you want to traverse. So, all the sets are given over here. Actual values are fed back from here. So, this line is known as the feedback line. This is your transducer. Finally, it goes here. So, both the signals are of the same kind now. And this comparator basically compares and finds the error. The error goes to the controller. A controller can be a PID controller. It can be of various types. We will discuss this later. So, this controller gives you a signal. The whole of this system, which is here is known as a controller, and this is a closed-loop controller. So, we will be dealing with this in this module also.



So, how a typical industrial robots linear? Why can we be assured that our robot can be controlled by simplifying our system to a linear system? It can be handled by a controller which is a linear controller. So, that is assured by a few things which are noted here one by one. So, the driving system of these joints. They have a huge amount of gear. So, first link, let's say you have, and then you have a gearbox. Then comes your second link. So, this is a gearbox which is here, and it can be 300 times gear reduction over here. In order to multiply the torque, the torque at the input is much less than compared to torque at the output. So, what happens to it? It is basically a reducer. It reduces the speed and enhances the torque. Basically, the fundamental equation for this is tau1- omega1 is equal to tau2 omega2. So, if you reduce the speed, you get the higher torque. So, that is what is the reason why it should be here. So, you have a gear reduction which is here. So, doing it like this linearises your system dynamics. So, we will again see when we will do a DC motor modelling for our joint. So, we will see there are electromechanical features which go into that model. So, electrical and mechanical things. Finally, your system gets very much like a linear system if you have a huge gear reduction. So, this basically isolates this link from this link. So, handling this second link, which comes next to the first link. So, it can be handled without worrying about what is going to come over here, and even the joint which is over here which is driving this link. So, this joint can also be handled independently. So, you need not worry about what torque the S link is going to transfer. So, basically, both are isolated. They can be treated like an isolated link. You need not worry. What kind of effect does this link is going to do on this link? So, they are decoupled that way. So, the presence of gear makes things very simple. It helps us to enhance the torque and reduce the speed to make it more controllable. It also isolates the joints, from one joint to the other, from one link to another. So, you need not worry about all that. So, mass and inertia terms are reduced by the square of gear ratio. This is what is proven later in one of the lectures in this module only. So, you see the effect of mass, the effect of inertia that basically the next link is going to put on the previous one is very, very less. It is reduced by the square of the gear ratio. So, you can imagine if it is 100 times you are squaring. So, the effective mass that the previous link sees, the effect of mass that comes here, M2 comes to here, is reduced by 100 square if the gear ratio is 100, you see. So, that is what we are going to prove later on also. So, in doing so, joint dynamics are decoupled. You can treat them independently, and each joint can be controlled independently, So one of the links is not going to have an effect of its motion on to the other. So, that is what isolation is here, I mean. So, single input, single output control is easily possible rather than multiple input, multiple outputs. So, this is what is industrial robot is. So, that is what is simplifying your system to a greater extent. And the majority of industrial robots work like this. And you know, again, we will be using the DC motor model to make you understand all these theories. But yes, they are quite nearly applicable even with any other kind of motors that you see, Even for AC synchronous motors, which are mostly present in industrial robots. So, this theory is also applicable to that. So, what are the control techniques that are there? So, just now, you saw what a closed-loop system looks like. So, yes, it can be controlled by simply switching it on and off. Let us say you have a ceiling fan in your home. So, yes, there are multiple ways to make it go at a particular speed. So, one way of doing so you just to switch it on. Keep it on for a moment. By the time it reaches some speed, switch it off. If it reaches a value, switch it off. It gradually will start reducing its speed due to friction, air drag and all. Finally, it will reduce. So, if it comes down significantly to some value below a certain value, again switch it on. So, you can keep switching on and off, and you can make that fan rotate within a certain range. So, that is one way of controlling it. That is known as a two-step control. Your system behaves like this: It picks up its speed, reduces again, picks up, and reduces. So, every time you switch on, you go like this: switch off, you come down, switch on. So, that way you can definitely control speed in a similar manner; you can even control the position, and you can handle different other parameters also. So, that is one controller. Others could be proportional, integral and derivative control. This is quite frequently used in industry for various, not just robots, but for many other types of equipment also which are there in the industry. So, we will go into very much detail about this type of controller. And non-conventional controllers are also there. Some intelligent controllers are there. AI-based control systems are there. Adaptive controllers are there. So, different other types of controllers are also there, which are normally not there in the industry. Nowadays with robots, yes, you do see even this kind of controller. But we will focus mostly on this one in this module. So, yes, even with the control system over here, even placing all the controls in place, you still have the possibility that your robot can become unstable. What is stable? What is unstable? If you remember a ball problem in which you have a ball placed in a ball? So, this is a ball. This is the ball. So, if you leave this ball anywhere on this ball, it will automatically come to the lowermost position. Got it. But if you have a situation something like this, if you move this ball, even by a small displacement, it will quickly go unstable, and it will fall, Got it? So, this is unstable, This is stable, Got it? So, two different kinds of equilibrium. You must know this. And there is one more kind, which is known as a neutral one. So, this is stable anywhere on this plane. If it is a plane, a flat plane, you can leave this ball anywhere on this plane, and it is okay. It is neither unstable nor stable. It is known as a neutral equilibrium. So, you know, this kind of situation does exist. So, we will try to work around this so that we can be in this. So, that is what some issues are there with that. The reason why we are using the controller is very much due to this also. So, this is all about linear control and the kind of controllers we are going to do.

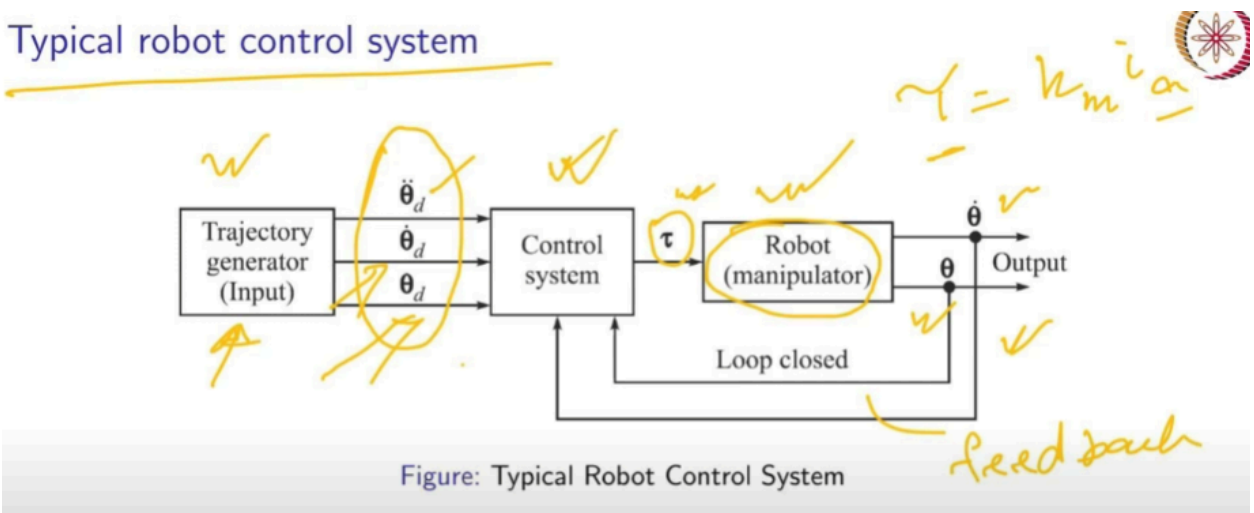


Figure: Typical Robot Control System

So, how is a typical industrial robot controller? So, just now, I have drawn a detailed layout of one of the joints. So, this is a set of joints. How does it work in a robot? So, this is your robot manipulator. You have a robot manipulator that basically is driven by the torque. As I have told you just now, you cannot command directly by the torque. So, you do command by the current that goes into the actuator. I am talking about electrical actuators over here. So, torque is not directly fed. So, if I tell torque, that means it is current that goes to the actuator. So, torque, yes, it is Km into Ia.

$$\tau = k_m i_a$$

So, if it is a current, so current is proportional to the torque that is desired. So, there is something which is known as a power amplifier that is there in between. So, that is implicit everywhere if I say that. So, the output of the robot is the velocity of the joint position of the joint. So, two sensors are there. These are feedback lines, feedbacks that go to the control system, which is here. So, this, basically, is the brain of the whole of your robot. So, this takes input from the trajectory planner. If I want to do welding, I want to move in the air and do some painting operations. So, those trajectories are fed from here. What is this? This (dot Θ_d) is a set of positions that are fed velocity that is required at each moment when you go along, and this (double dot Θ_d) is the acceleration that you want at every instant of time. All these three are fed to the control system, and the controller basically tries to maintain this desired trajectory. That is the input, that is the commanded input. So. it compares with the actual one to the desired one, and it has its brain. Based on that, it does everything. So, you see, this is a typical robot control system. We will talk about what actually goes here and how it is possible to do this basically.

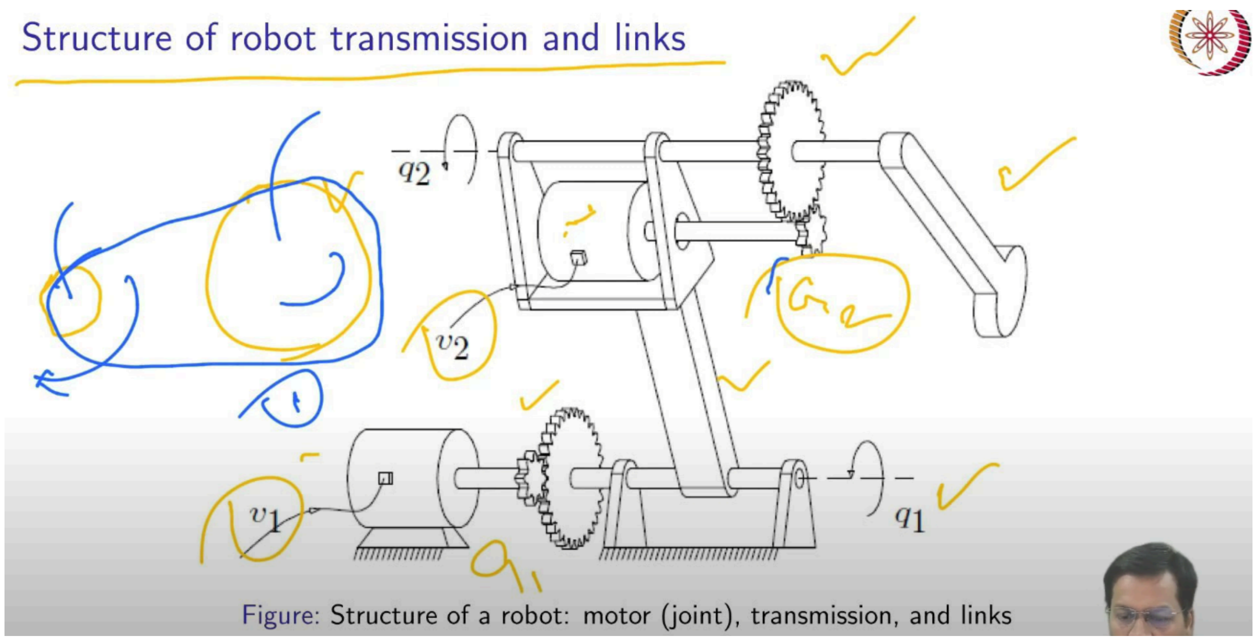
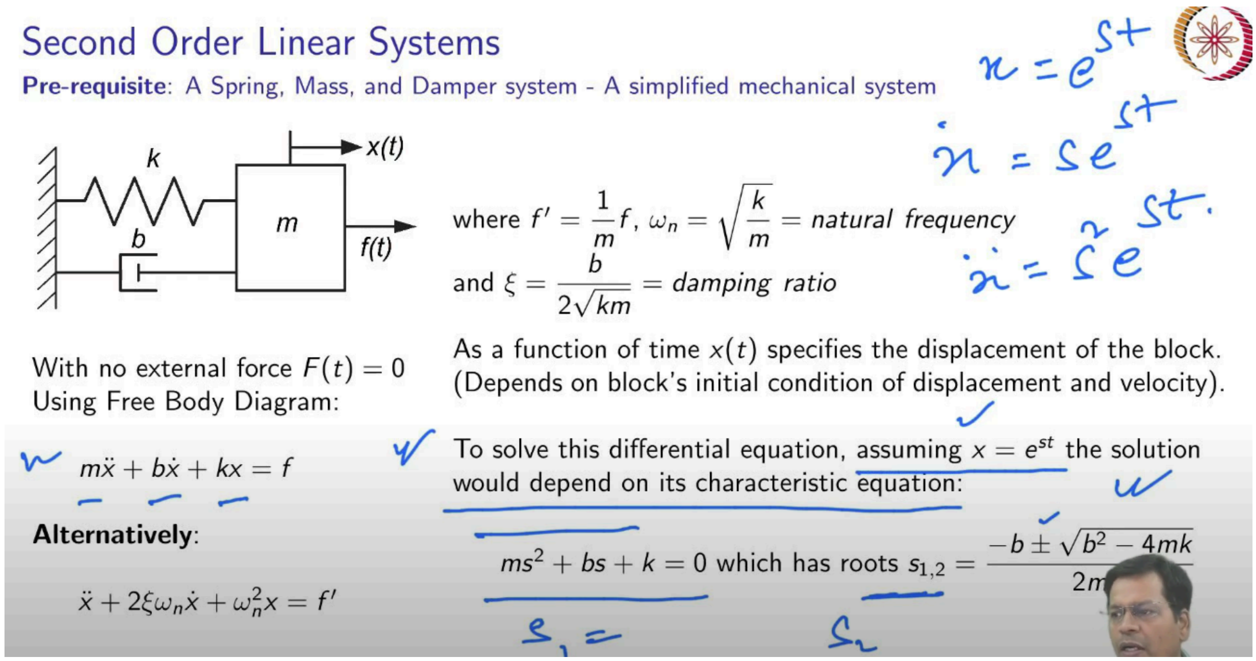


Figure: Structure of a robot: motor (joint), transmission, and links

So, a typical structure of an industrial robot transmission system with its links looks like this. So, you have a motor, you have a gearbox, and sometimes they are belts, and then you have another link, and You have another motor that goes on top of this. This motor again has got a gear reduction here. This was gear reduction 1. And then again you have a link which comes here. So, this is how it is made. So, you have the current and voltage that comes here to the motor, and this is the output joint variable. So, this is how I assume all the robots are made. Yes, there are a few differences at times. You may see belts over here. You can have direct transmission also. So, there are possibilities with some variations. But effectively, if I say it is gear, basically, it is some reduction which is happening. It may be due to the size of two driving and driven pulleys, and you have a belt that goes here. Even this is a reduction. This is the driving pulley, this is the driven one. So, if this rotates by two revolutions, this will rotate. Let us say, just by one revolution. Got it. So, this is also a reduction. So, if I say it is a gearbox, it can be due to various reasons. It can be a combination of belts and gears also. So, that is what is here.



So now, let us try to understand what a second-order linear system looks like. We are familiar with spring mass and a damper system. A typical mechanical system can be represented simplified model can be this one. We will see how a robot equation can also be represented like this later in this module. So, for now, let us say you have a mass which is connected with a spring. Spring is attached to the wall which is here. This is nothing but a damper with a damping constant, which is given by b over here. So, you know this equation very well. So, if you draw a free body diagram of this, you have, if you displace this by a small displacement, delta x. So, spring generates a force called k delta x. Is it not? If you move this way, spring is going to pull it this way. Is it not? And again, if you are moving with a certain velocity x dot. So again, your damper will pull it this way. That is given by bx dot. So, overall, the system is in equilibrium. If you are moving with some acceleration, so you have a pseudo force that tries to act like this, and that is given by mx double dot. Is it not? So, if you move like this, you see inertial force that comes like this. You have damping force, and you have stiffness force. So, combined all together, you can write your equation like this:

$$m\ddot{x} + b\dot{x} + kx = f$$

This is the cause which is actually creating this motion. So, if you are pulling it by force f, so these are the components that are going to come. So, this is the equation that you are going to assume. So, what is this equation? How does this system behave? So, this can be represented. You know already using this.

$$\ddot{x} + 2\xi\omega_n\dot{x} + \omega_n^2 x = f'$$

What is this? If you remember, this is nothing but a spring mass damper system. So, you have a damping constant, which is given by this. This is known as a damping ratio. If you are familiar with simple harmonic motion in your high school physics, so you must have seen this, probably. Even if you are not we will come back to this, and we will do it once again And again. Your omega n is your natural frequency, which is given by the square root of k by m. So, if you leave this system to oscillate on its own without a damper, it will oscillate with this frequency k by m root over. So, that is the natural frequency. So, as a function of time, x(t). x(t) is the displacement, which is a function of time. It specifies the displacement of the block. So, this depends on the block's initial condition of the displacement and the velocity as well. You may start from the mean position, or your system is already in a location, and you have given it a velocity so that the boundary condition may change, but I assume that it has started from rest. So, to solve this differential equation, I have assumed something like this has to substitute x is equal to e to the power st. s is a constant. The solution would depend on the characteristic equation which is given by this.

$$ms^2 + bs + k = 0$$

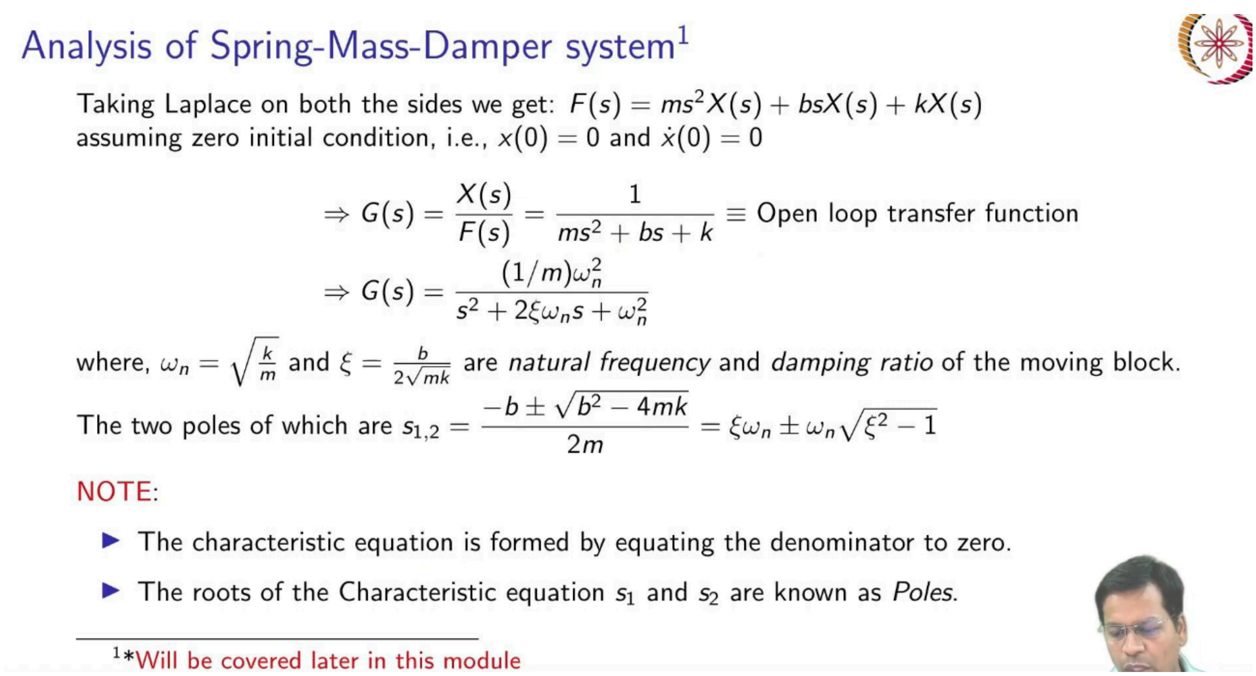
Try substituting this. What you do get is if x is e to the power st. So, what you get- x dot is se to the power st.

$$\dot{x} = se^{st}$$

Similarly, x double dot will give you s square e to the power st. Is it not?

$$\ddot{x} = s^2e^{st}$$

So, if you substitute all these to this x double dot, x dot and x directly, you can take e to the power St common out here, and you are remaining with ms square bs plus k. What is this? This is basically known as the characteristic equation of this, which has roots given by $s_{1,2}$. There are two roots because this is a quadratic equation. So, S1 is the positive of this, and S2 is given by the negative of this, so there are two roots which are there. Based on the characteristics of these roots, your system has some behaviour. That is what we will discuss now.



So, this system can also be treated like this. What makes this system to be treated like this? Basically, this was your fundamental equation. Force is equal to mx double dot plus bx dot plus kx. So, if you take Laplace on both sides. What do you get? So, it is F(s) equal to ms square X(s) plus bs X(s) plus K X(s). So, if you take all these common, what do you get? X(s) into ms square plus bs plus K is equal to F(s). So, you got it.

$$f = m\ddot{x} + b\dot{x} + kx$$

$$F(s) = ms^2X(s) + bsX(s) + kX(s)$$

$$F(s) = X(s)(ms^2 + bs + k)$$

So, now, X(s) is your output, and F(s) is your cause, which is the input. So, output by input is also known as an open-loop transfer function. So, it is given by G(s).

$$G(s) = \frac{X(s)}{F(s)} = \frac{1}{ms^2 + bs + k}$$

So, what do you see here? Output by input can be written as 1 by ms square bs plus K. What is it? This is nothing but the transfer function. So, the part which is here can have multiple roots and that is the reason why this is known as characteristic equation. If the denominator goes zero, so it will have certain values. So, if the denominator is given by the characteristic equation that governs the type of gain that you are going to get, that is actually governing the system output and input relationship. So yes, now I am again expressing it in the way omega n. that was the natural frequency of the system. You know that. So, omega n this is the damping ratio of the moving block. So, if I use these two substitutions to this, I can write the same equation as this,

$$G(s) = \frac{(1/m)\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

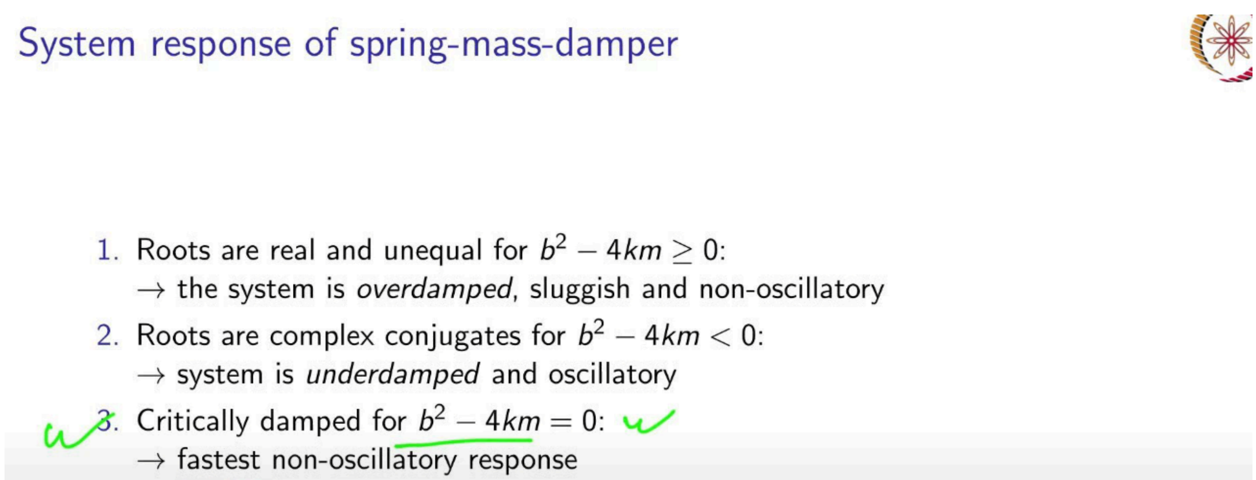
And again, the two poles which are there are this:

$$s_{1,2} = \frac{-b \pm \sqrt{b^2 - 4mk}}{2m}$$

This is nothing but the roots of the equation. This is given by the discriminant, which is here b square minus 4mK. So, this equation can now be written like this:

$$s_{1,2} = \frac{-b \pm \sqrt{b^2 - 4mk}}{2m} = \xi\omega_n \pm \omega_n\sqrt{\xi^2 - 1}$$

So, it is damping is ratio square minus 1, that is, within the square root? This can be less than 1 or greater than 1, so the whole of the behaviour of the system will be governed by this. We will discuss very much in detail this, so, don't worry, we will come back to this once again. So, the characteristic equation is formed by equating the denominator to zero. So, if we equate it to zero, basically, we want to find out the roots of this equation. So, these roots are also known as the poles of this equation, and that is going to control the whole of the system type and the behaviour of the system.



So, what could be the behaviour of the system? Roots are real and unequal that is the case when b square minus 4km is greater than zero. In this case, the system is overdamped, sluggish and non-oscillatory in nature, which means if you pull it or leave it, it gradually goes to the original position, it won't oscillate. So, that is one behaviour that you can see. The next one is roots are complex conjugates. That can happen only when b square minus 4km is less than zero. In this case, the system is underdamped and oscillatory. Oscillatory: the system will oscillate because it has some damping. So, that oscillation will die off a little bit in every next oscillation that it makes. And finally, you see, you can have a behaviour which is something like this. So, it will oscillate. Something like this: This becomes your displacement, and this is your time. So, over the period, it will oscillate about the mean position, and this will gradually decrease. So, we will discuss very much in detail this one. So, this is an underdamped and oscillatory. The third one is critically damped. This is a condition when b square minus 4km is exactly equal to zero. So, this is the fastest non-oscillatory response. Your system is not going to oscillate. It will gradually go to the mean position, but this is the fastest way it can go to the mean position. Faster than this also. So, these are the three behaviours that you can see.

So, that's all for today, and in the next class, we will deal with this problem further, very much in detail. We will discuss the Response of the Second-Order Linear System in the next lecture. So, that's all for today. Thanks a lot.