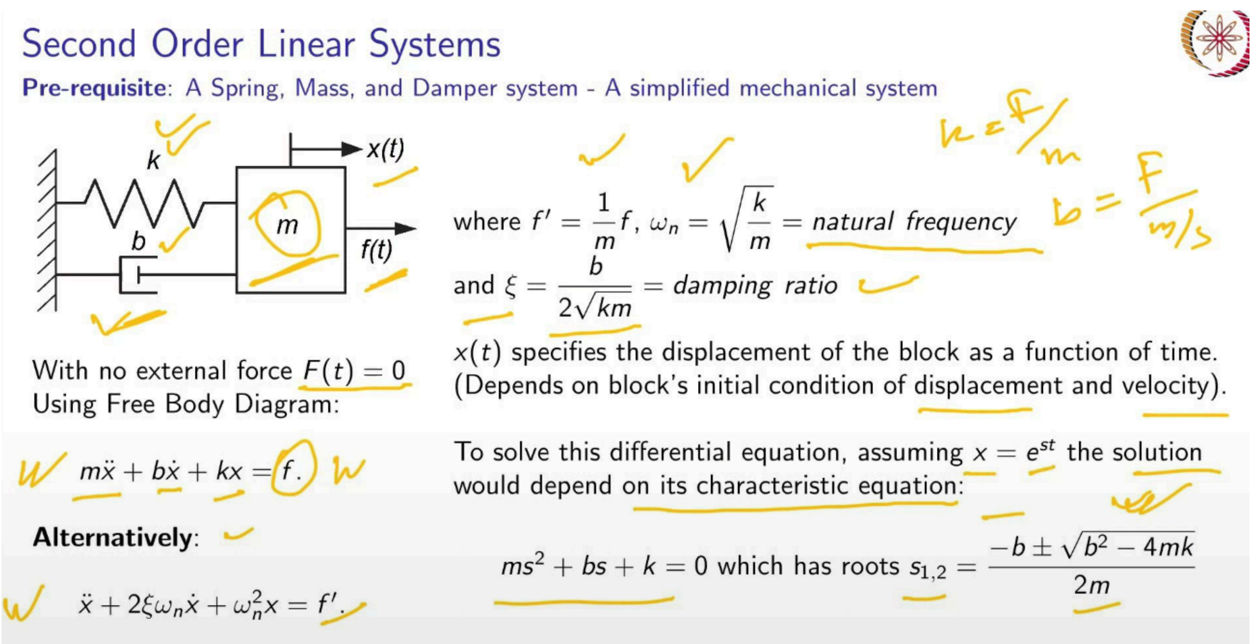
NPTEL Online Certification Courses
Industrial Robotics: Theories for Implementation
Dr Arun Dayal Udai
Department of Mechanical Engineering
Indian Institute of Technology (ISM) Dhanbad
Week: 11
Lecture 44

Response of a Second Order Linear System

Welcome back. So, in the last class, you learnt what a controller is and why a robot needs a controller. You also saw an example of a spring mass damper system. That is a second-order linear system. So, today, we will move further with that, and we will understand the behaviour of such systems with different parameters and their behaviour. So, today, we will begin with the response of a second-order linear system. So, let us continue.



So, this was the slide that you saw even in the last class. So, you saw, this is a typical spring mass damper system with here is your mass, this is your damper-with damping constant as b, damping coefficient as b and spring coefficient, that is k. So, k is nothing but force generated per unit displacement. So, the unit could be Newton per meter. Similarly, you have b, which is given by force generated per unit velocity of this system, So it should be metered per second. So, the units are accordingly. So, you see, these are the two constants that are there. That basically determines the behaviour of this system. Basically, that also determines the characteristics of this system. So, that is also defined by the mass m, which is there. It is applied by an external force, f(t) you see, it is here. Let us say I have a displacement which is given by x(t) at any instant of time. So, with no external force, that is, with f(t) is equal to 0. The system can just be written as so this goes 0. So, it is mx double dot plus bx dot plus kx.

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

So, that becomes your time domain equation, the dynamic equation of this spring mass damper system. This is nothing but a differential equation of second order. Alternatively, the same can be written as you saw. It can also be written as x double dot plus 2 ξ omega n x dot plus omega n. square x is equal to the f dash.

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

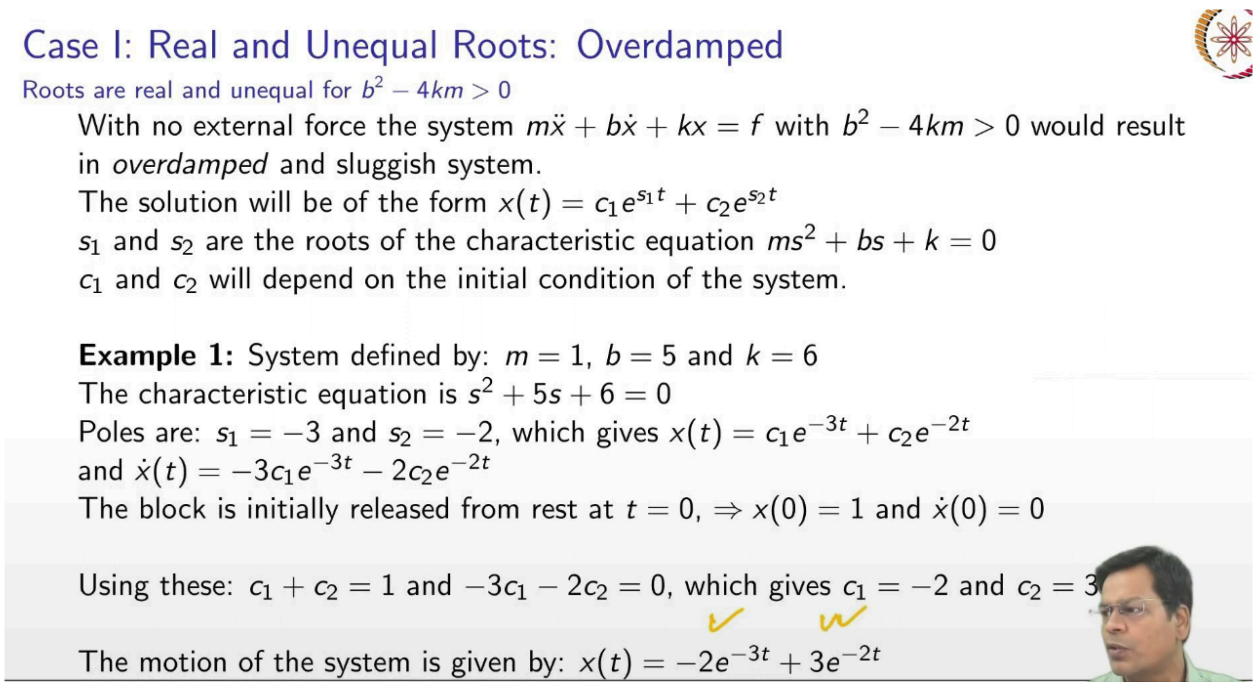
So, I have divided the whole of the equation by m. So, effectively, the f dash is nothing but f divided by m. So, that is here. So, it is proportional to f. So, whatever is the behaviour of f that is fed as an input. So, the f-dash also goes similarly. So, omega n is your square root of k by m., which is known as the natural frequency of the system. ξ is given by b by root over km.

$$\xi = \frac{b}{2\sqrt{km}}$$

So, that is nothing but a damping ratio. x(t) specifies the displacement of the block as a function of time. That depends on the block's initial condition also, that is, displacement and the velocity. To solve this differential equation, we assumed in the last class that x is equal to e to the power st, and the solution you know now depends on its characteristics' equation that is given by ms square plus bs plus k, which has a root of s1 and 2. So, that is s12. So, it is minus b plus minus root over. This is the discriminant, which is here that is b square minus 4mk root over divided by 2m.

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

So, it has something which is under square root that can go positive; it can be a real value. In case it is negative, it will have a complex value. So, in that case, it becomes minus b plus-minus, a complex number out here divided by 2m. So, the whole of this s1,2 is known as the roots of the characteristics equation, and this decides the behaviour of this system.



Let us begin with different kinds of such systems. One such system could be an over-damped system in which b is dominant; that is, damping is dominant over the spring stiffness. So, the roots are real and unequal. So, in that case, we have no external force. So, f becomes equal to 0, and this is the time domain solution of your system and with b square minus 4km going to be greater than 0. That would result in an over-damped system, and it behaves in a sluggish way. The solution would be of the form: x(t) is equal to C1 e to the power s1t plus C2 e to the power s2t, where s1 and s2 are roots of the characteristics equation. C1 and C2 are the constants which can be found by the initial condition, the boundary condition, that is, initial velocity and displacement, which was given while you just pulled it at a distance and just released it so that it can oscillate or it can dampen.

$$x(t) = c_1 e^{s_1 t} + c_2 e^{s_2 t}$$

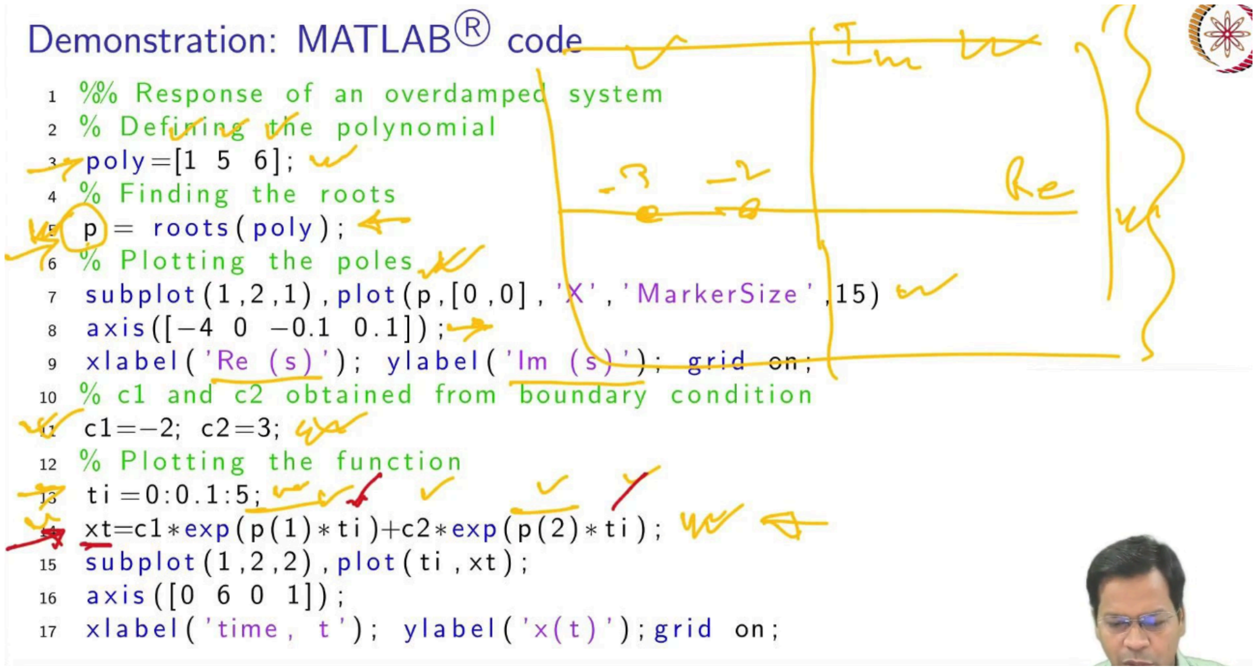
So, s1 and s2 are the roots of these characteristics equation. C12 would depend on the initial condition. So, yes, let us just take one such example and see the behaviour of this system. So, for this system, I have assumed m is equal to 1. Units are accordingly in the SI system. So, if it is 1, it is 1 kg. Similarly, b, which is the damping constant, is 5. Stiffness is 6, so the characteristics' equation can be written as ms square, that is, s square plus bs, that is 5s and plus k, so it is 6, it is equal to 0. So, that goes here. So, in the last class, you saw why this is called a characteristics equation. Basically, this relates to the output by input. So, output and input are related by this and in the denominator, you saw this goes. So, if that is real, that is 0. That is a constant that has some imaginary numbers. So, that will basically govern the behaviour of the system, and that is the reason it is known as the characteristics equation. Let us start analysing this. So, what are the roots here, which will be known as the poles?

Poles are: $s_1 = -3$ and $s_2 = -2$, which gives $x(t) = c_1 e^{-3t} + c_2 e^{-2t}$

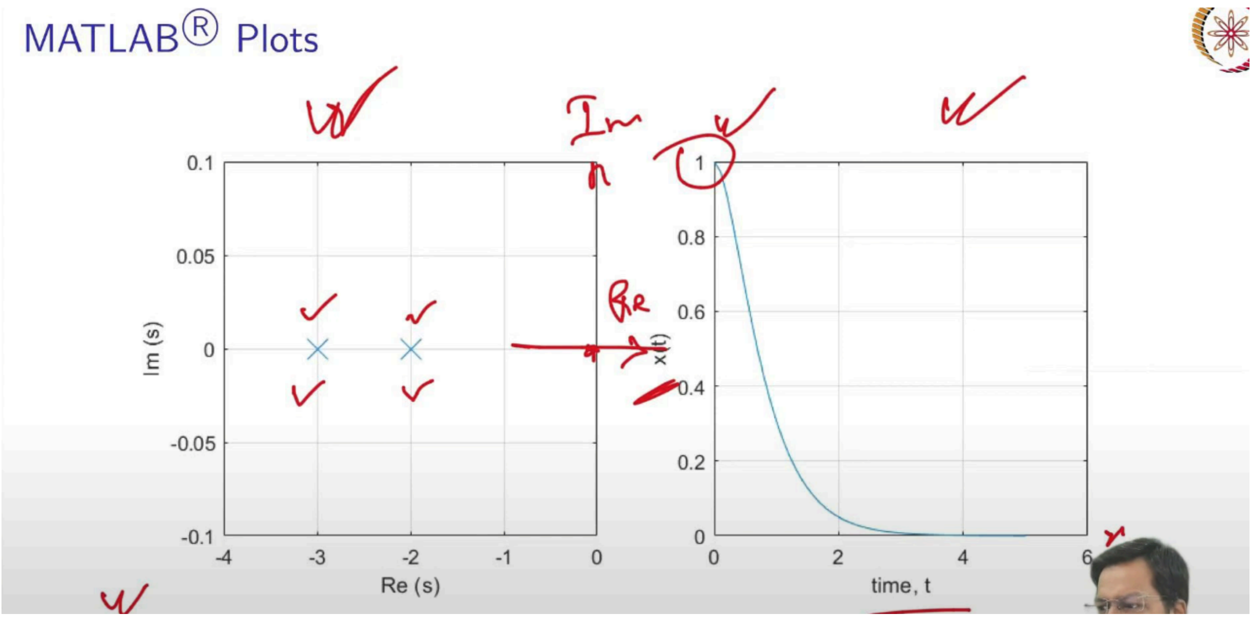
That is, s1 is minus 3, and s2 is minus 2. So now, the equation which I told you should be like this. So, the time domain solution would look like this: x(t) is equal to C1, e to the power minus 3. One of these will go here, and similarly, s2, that is minus 2, that will go here. So, effectively, your system time domain solution should look like this: now, C1 and C2 are to be determined. We know the system has started from rest. The initial displacement was given. So, first, let me just take a derivative of this. Taking a derivative would give me in time. It should give me x dot t, which is equal to minus 3 C1, e to the power minus 3 t and minus 2 C2, e to the power minus 2 t got it.

$$\dot{x}(t) = -3c_1e^{-3t} - 2c_2e^{-2t}$$

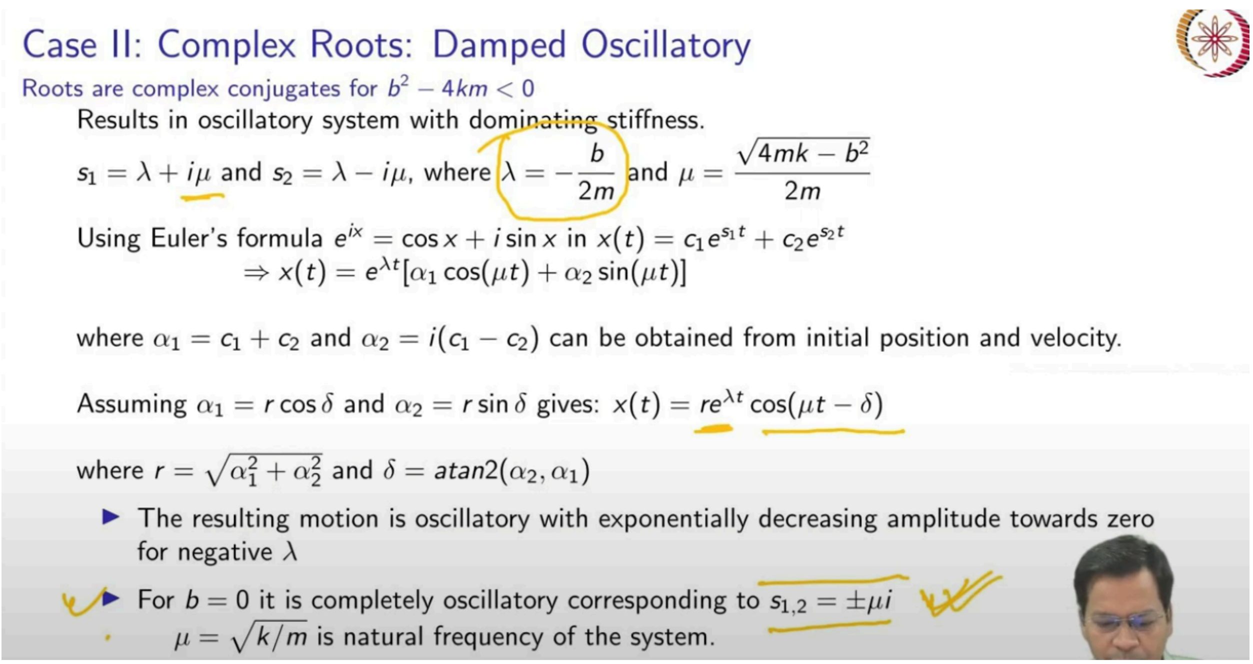
So, I now know time domain solution like this, velocity, which is like this. So, displacement and velocity substitute the initial condition. I know t is equal to 0 at time t is equal to 0. When the block was released. It has a displacement of some unit meter, so 1 meter. And then x dot 0 is equal to 0; that is, the initial velocity is 0. So, this is how these two conditions can be put over here and here. So, at t is equal to 0. I just put t is equal to 0 in this and put at is equal to 1 here. So, I will get one equation. And again, x dot 0 is equal to 0. So, I will put again that t is equal to 0 in this equation, and I will get one equation for x dot t is equal to 0. Put it here. So, these two will give me two equations: this one and this one. Solving these two simultaneous equations. I could extract C1 and C2. So, you see, I got C1 as minus 2 and C2 as 3. These values make this time domain solution of this system complete. So, I will put them here. And I got to the motion of the system, which is given by: at is equal to minus 2, e to the power minus 3t and plus 3 e to the power minus 2t. So, this is the time domain solution of my system. So, now I can plot this with varying times and I can see how it is moving over time. So, it may oscillate, it may dampen so that I will see the behaviour also. I know the poles of this system also. So, those are given by minus 3 and minus 2, so that can also be plotted somewhere.



So, let us see how to go about it. So, I will start with a simple MATLAB code. I assume quite a few of you must be having this MATLAB also with you, even if you don't have it. So, Octave is a free software which can run any MATLAB code of this type. So, you know, displacement is to be plotted against time. So, you have the equation which is given. So, this can even be coded with simple MATLAB code, and you can just plot the results. You just have to plot x with time with the same variables that I will be taking now. So, yes, this is your system. That is the polynomial of the characteristics equation. It was given as ms square plus bs plus k, three things were there. So, that is put here, as I have just created a polynomial with the coefficients which are here. So, the roots of the polynomial will be stored in the variable which is p over here. So, p will have the value. So, what value it will have? It will have minus 3 and minus 2 got it. So, that goes to p, and I have created two plots. So, one of them will plot p, so that is the first one. That is nothing, but p means you have two values over, so one of them will plot p. that is in a complex place. So, you have the real number that goes to one axis, and you also have an imaginary number. So, the imaginary is plotted along here, real number is plotted over here. So, you know you have both the roots with minus 3 and minus 2. So, both the roots will be somewhere over here in the real number line. So, minus 2 and minus 3 should be visible somewhere over here. So, yes, all the poles are drawn in a complex plane so as to analyse your system. This has a special significance you will come to know now. So, why are we plotting the roots of the characteristics equation in the imaginary plane somewhere like this? What does this signify and how can this dictate the behaviour of your system? You will come to know now. So, yes, with the extents, I have just put the extents so that you can see the range. So, the whole of the extent is now set so that you can see the range of your values clearly, and I have labelled it along the x-axis along the y-axis like this. So, these are the two constants. With this, the time domain equation of your displacement varies. So, x(t) is equal to C1, e to the power, S1t is here, similarly, plus C2, e to the power s2t. So, you have all the values of constant C1 and C2, whereas S1 and S2 are nothing but that you have stored your polynomial roots in p. So, p1 and p2 will tell that. So, that goes here. So, effectively, I am writing the time domain solution of my displacement over here, and this is to be plotted, so what have I done? I have simply made Ti vary from 0 to 5 seconds in a step of 0.1 that is stored here. Similar is the way in Python is in octave, so you can just key in like that. So, Ti goes like this. So, the whole of the Ti is inserted here and here, and the x(t) array is created. So, you get, for each value of Ti, the system has calculated x(t), and x(t) is stored over here. This is to be plotted against Ti, so x(t) against Ti. So, I have created another plot.



So, what do I get? There are two plots, so this is the one which is the plot of roots, that is minus 2 and minus 3. Origin is here. So, this is your imaginary axis, and this is your real number line. So, you see, it has some values in a real number line. So, you have some values which are here. So, these two are the poles and both the poles are on the negative side in the complex plane. So, both the poles and they are on the real number line. They are not imaginary, so this is your system, which was plotted against time. So, you have time which goes here and x(t) that comes here. So, you see, you have released it from a displacement, which is 1, and then it gradually comes to 0. So, without any oscillation. You see, it is a system which is given by roots where real and unequal. So, b was dominant, so it was an over-damped system. So, it is an over-damped system. It quickly comes to rest. So, that is the 0 thing. So, this is the behaviour of your system for real roots, which are on the negative side of the complex plane. So, this is the system, this should govern the system and this is your actual system how it is behaving. So, just remember this, and we will compare this with other results that we will be getting in the next system example that I am going to give, so this is first.



Now, the second one. So, this is a system where you have b square minus 4 km, less than 0. That means I should see complex conjugates. Complex number roots always appear in conjugates. You know that already. If it is a quadratic equation like your characteristics' equation, those two numbers, those two poles, those are the roots, may be written as lambda plus iμ and lambda minus iμ, where lambda is given by minus b by 2m, and similarly μ may be written as this:

$$\mu = \frac{\sqrt{4km - b^2}}{2m}$$

So, I have just taken i square is equal to minus 1. so that is a complex number thing that you already are familiar with. So, I will just put it here. And you're substituting that value, S1 and S2, to this equation. So, what do you get? You can quickly write it as x(t) is equal to C1 e to the power lambda plus i, μ plus C2 e to the power lambda minus i μ, into t and again here, also, you have t. So, this can be taken out as e to the power lambda t. That can be taken out, and inside, you will have C1 e to the power i μt plus C2 e to the power minus i μt, so that goes inside.

$$x(t) = c_1 e^{(\lambda + i\mu)t} + c_2 e^{(\lambda - i\mu)t}$$

$$e^{\lambda t}[c_1 e^{i\mu t} + c_2 e^{-i\mu t}]$$

So, now I can substitute the second part of it, this one e to the power i x is equal to cos x plus i sin x. So, if you substitute that once again here. So, you can get e to the power lambda t, which is still outside, and C1. So, here I will write it as cos μ t plus i sin μ t. So, that comes here. Similarly, over here, I can write it as plus C2 cos μ t. This time, it should be with a negative sign here: cos minus μ t, and again, you have plus I sin minus μ t. So, this can be written like this: so that comes here: minus μ t, minus μ t, and finally, whole of the bracket will come here and that is it. So, if you take all of these common, you can write it as e to the power lambda t, C1 plus C2 cos μt, that will come here, and plus i C1 minus C2, and you can write sin μ t, got it. So, this is how you can write your system x(t). So, that goes exactly like this.

$$e^{\lambda t}[c_1(\cos\mu t + i\sin\mu t) + c_2(\cos(-\mu t) + i\sin(-\mu t))]$$

$$x(t) = e^{\lambda t}[(c_1 + c_2)\cos\mu t + i(c_1 - c_2)\sin\mu t]$$

So, if I again take this (c1+c2) as alpha 1 and this i(c1-c2) as alpha 2, you can proceed like this:

$$x(t) = e^{\lambda t}[\alpha_1\cos\mu t + \alpha_2\sin\mu t]$$

So that you can write it as At is equal to, exactly like this, everything comes here. So, whereas alpha 1 and alpha 2 are nothing but C1 plus C2 and i C1 minus C2. they can be obtained from initial position and velocity. So, you know, you already know when you, when you have released your block and at what displacement you have released, what velocity you have imparted. So, those initial conditions can be put the way we did it just now in the earlier example, and I can evaluate alpha 1 and alpha 2. So yes, again, I will assume alpha 1 and alpha 2 as rcos delta and rsin delta. That can lead me to this.

$$x(t) = re^{\lambda t}\cos(\mu t - \delta)$$

So, how, I will tell you once again. So, what was that? So you just got x(t) is equal to e to the power lambda t here, you see you have rcos delta and cosine μt plus rsin delta and sin μt. So, that is here. So, you see, you can bring out your r that goes as e to the power lambda t.

$$x(t) = e^{\lambda t}[r\cos\delta\cos\mu t + r\sin\delta\sin\mu t]$$

Inside what you see, cos delta, cos μt, sin delta, sin μt. So, that can be written as cos μt minus delta.

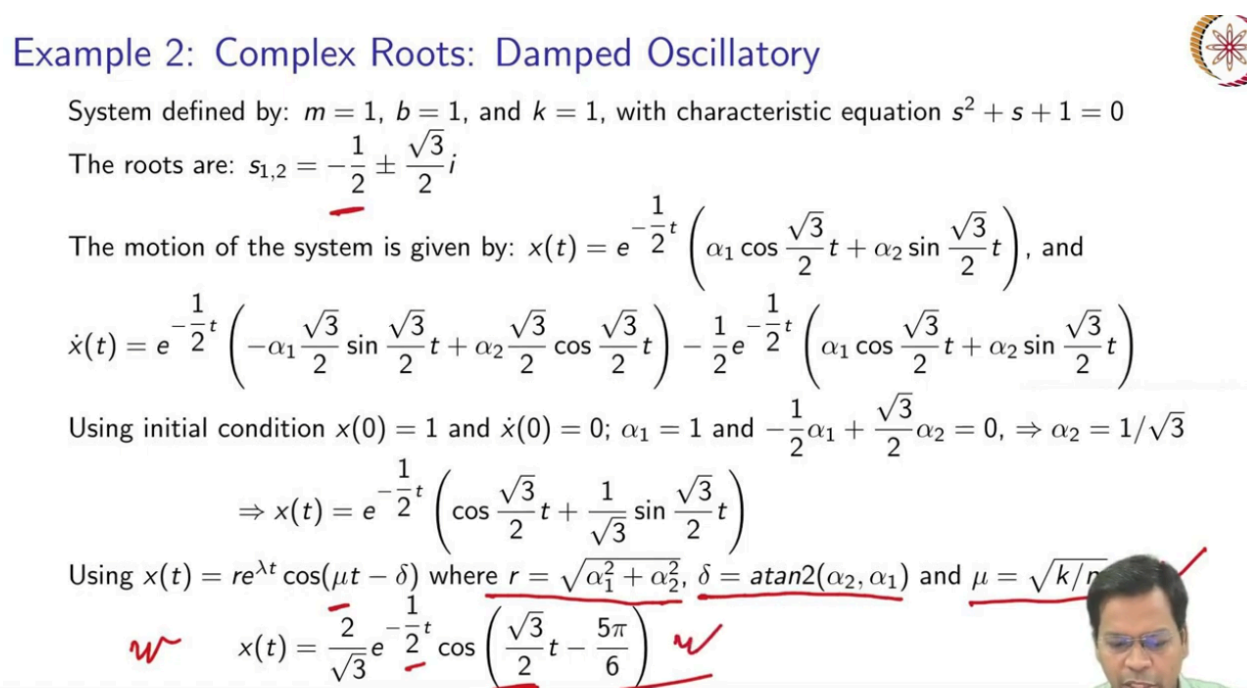
$$x(t) = re^{\lambda t}[\cos(\mu t - \delta)]$$

Those are simple trigonometrical substitutions. So, you see, your system behaviour will now be given by this. So, this is the time-domain solution of this. So, if you just square and add your substitution which you made here. So, squaring and adding will give you alpha 1, square plus alpha 2. Square, root over becomes r, and by dividing alpha 2 by alpha 1 you can get to this. That is, delta is equal to tan, inverse alpha 2 by alpha 1. So, that is there. So, quickly you have obtained your equation for the time dependency of your system. So, you can plot your system now with time, the displacement with time you can obtain from this. That will tell you how your system is behaving, whether it is oscillating or what it is doing. You already know what your roots are. Your roots are nothing but complex conjugates. The motion is oscillatory with exponentially decreasing amplitude. You see, this ($re^{\lambda t}$) is a constant; for any value of lambda, this is a constant, but multiplied over e to the power lambda t, lambda is negative. You see, lambda is negative, so it is exponentially decreasing amplitude. So, this basically creates the amplitude. So, that is, you see, it is exponentially decreasing, and this is creating an oscillatory behaviour. With time, if you plot, your displacement will behave in a cosine manner, with some phase over here, so it won't start with the exact position where you see the peak. But you see, here is you have some delta, which is here, which is given by this.

$$x(t) = re^{\lambda t}[cos(\mu t - \delta)]$$

So, the resulting motion should be oscillatory and exponentially decreasing amplitude towards zero for the negative value of lambda. Lambda is negative, you know that. For b is equal to zero, that is, if at all. You substitute b is equal to zero, so this becomes zero, so lambda is zero. So, in that case, this is purely a constant value, and you are left with just cosine oscillations. So, your system now is completely oscillatory in nature, corresponding to s12, which is just plus -iμ. So, what is this? It is just pure, complex numbers. So, it doesn't have any real value. So, it is imaginary, so both the roots are imaginary, and they are conjugates, and you don't have anything. So, it is exactly on the real, imaginary number line, and you have μ, which is given by the square root of k by m. That is basically the natural frequency of the system. So, you have mu. So, that is the angular frequency of cosine. So, you see, μ was here and what was μ? So you just substitute: b is equal to zero. Here you get to mu. So, μ is equal to the square root of k by m which will show the natural frequency of the system with which it will oscillate.

$$\mu = \sqrt{\frac{k}{m}}$$



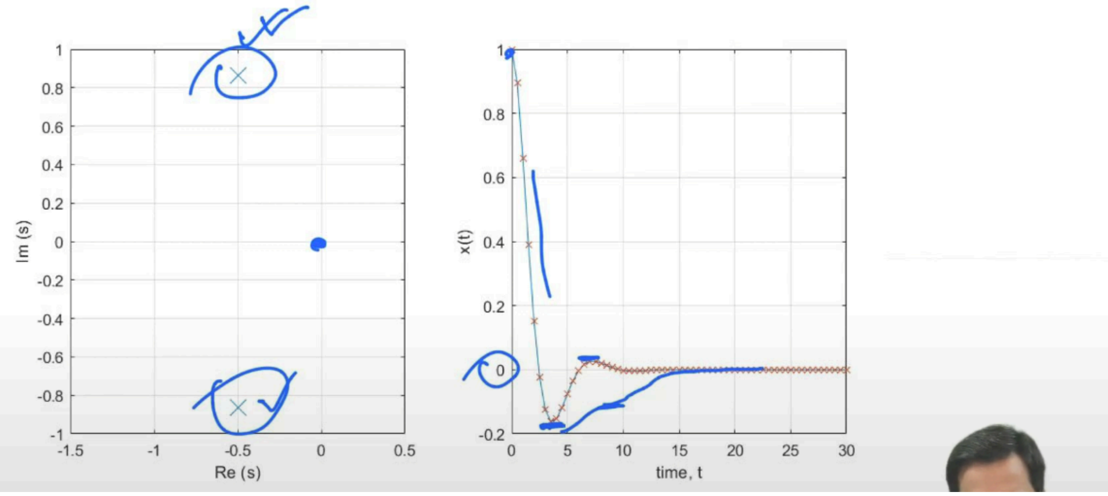
Now, let us just see how to plot this with the system, which is similar to this. So, you have your system. Given by m is equal to 1, b is equal to 1, and k is equal to 1. The characteristic equation is given by s square plus s plus 1, roots are. You see it is complex, conjugate, with real value here and imaginary value here. The motion of the system will be given as x(t). x(t) is e to the power lambda, t, r, e to. The power lambda t. you see, r is 1 here, so e to the power lambda t, so lambda was minus 1 by 2. That comes here, and plus-minus μ. So, μ will go here μt, so alpha 1 cos μt plus alpha 2, sine μt, μ comes here. So, again, moving further, if I take the derivative of this, I should be getting this. You can do it yourself. So, first into a derivative of second and second into a derivative of first will give you this. So, this I have done just to get the position and velocity equation of my system, and when I substitute the initial condition, at time, t is equal to 0, x is equal to 1, and similarly for velocity, x, dot is equal to 0, at time t is equal to 0. Putting these two values in these two equations, I can get alpha 1 is equal to 1 and minus 1 by 2. Alpha 1 plus root 3 by 2, alpha 2 is equal to 0. So, these are two simultaneous equations I am getting out of these two input equations and solving this. I also will get alpha 2 as 1 by root 3. Alpha 1, you got here, and alpha 2 is here. That makes this system time domain solution complete. So, now my system equation will tell: at is equal to e to the power, lambda t, alpha 1, and alpha 2 will come here, and you get the complete solution. So, this is all your system will behave. So, if I can again substitute for those constants, you already know that. So, you get to a cosine variation of your equation. So, you can directly use this substitution. Where r is equal to this, and delta is equal to this, μ is equal to this. So, you know you can write your equation in a new form which looks like this. So, here you see, this is your r-value that is going to come, and e to the power minus lambda t, so you have e to the power, lambda t. lambda was minus 1 by 2. So, r is calculated here and cosine mu, t minus delta, z, delta is 5 pi by 6. So, this is your system dynamic equation. So, this is a time-domain solution. You can quickly plot this and see its behaviour. You already know the roots. Roots are complex conjugates, so let us plot both. You can plot in the complex plane. So, your poles, where does it lie? And you can also plot the behaviour.

Demonstration: MATLAB® code

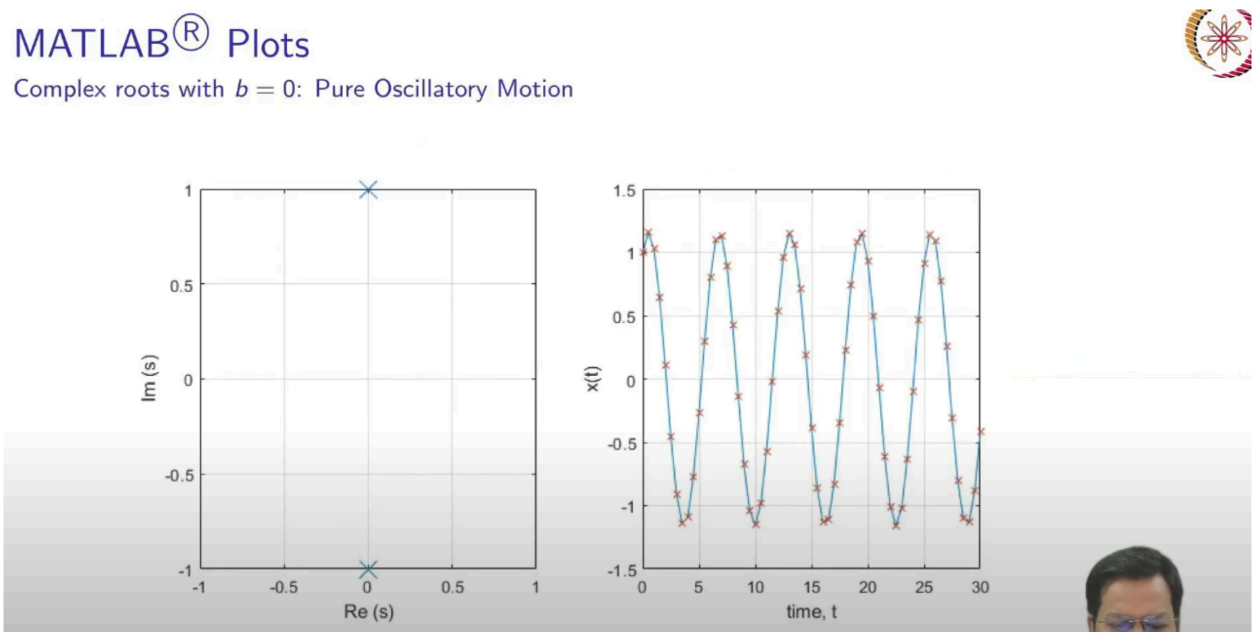
```
%% Response of an underdamped system
% Defining the polynomial
poly=[1 1 1];
% poly=[1 0 1]; % For damping b=0
% Finding the roots
p = roots(poly); pr=real(p);pim=imag(p);
% Plotting the poles
subplot(1,2,1),plot(pr,pim,'X','MarkerSize',15)
xlabel('Re (s)'); ylabel('Im (s)'); grid on;
% Plotting the function
% The symbol *. is to multiply two vertors term by term
al1=1; al2=1/sqrt(3); ti=0:0.5:30;
xt1=exp(pr(1)*ti).*(al1*cos(pim(1)*ti)+al2*sin(pim(1)*ti));
xt2=2/sqrt(3)*exp(pr(1)*ti).*cos(pim(1)*ti-atan2(al2,al1));
subplot(1,2,2),plot(ti,xt1,ti,xt2,'x');
xlabel('time, t'); ylabel('x(t)');grid on;
```

Now, I am doing that. So, I am using my MATLAB code once again. So, I am defining my system is storing the roots here, storing the real and the imaginary values here and showing them in the complex plane. Now, I am labelling them also here. Again, I am defining my system over here for the time variation, how my at will go, and I am putting them, plotting them once again here, and I am labelling it here.

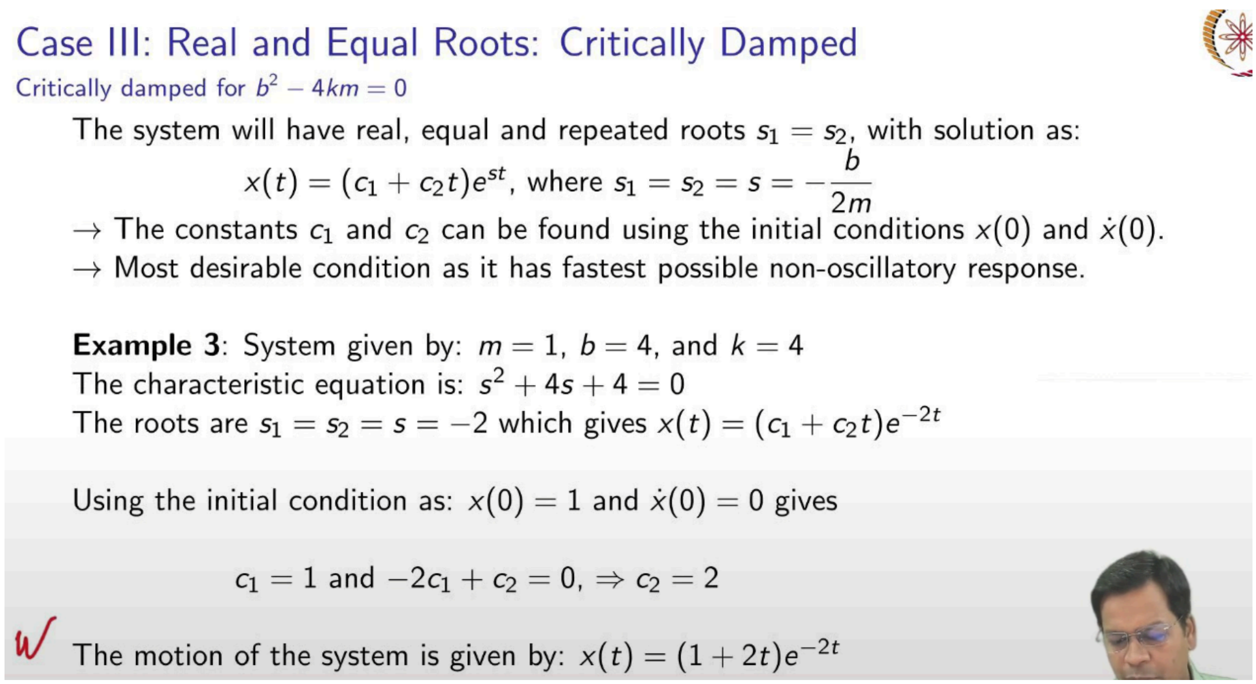
MATLAB® Plots



So, see how my system is behaving. You see, I have 0 and 0. So, that is coming here. That is your origin. So, both the poles are on the negative half of your complex plane, and both are complex conjugates. So, whatever is this? The same value mirrored like this and your system. You see, this time, it started from here. It is overshooting to some value, overshooting on the other side and coming back. It is still oscillating, which is not clearly visible here, but it is oscillating and finally reaches 0. So, what kind of system did you expect here? It is a damped oscillatory nature. You see, your system is damped, oscillatory and amplitude every time decreases exponentially. So, whatever the amplitude here, the next amplitude at every cycle will be reduced exponentially and finally, that comes to this. So, that is how it behaves. So, you now see again, you can compare with your earlier results. So, this time you have, you have poles which are on the complex side of it. So, you have these two value complex numbers, and they are on the negative side of the complex plane. So, the negative and complex conjugates system is damped, oscillatory behaviour.



Now, let us look at the system when it was the unnamed system. So, it is purely oscillatory. You have seen by analysis, so by plot, both the roots are shown here. You only have µ here. That is the plus-minus side in the complex µi and minus µi. So, that comes here exactly on the imaginary line, you see. So, this is your roots. Roots are imaginary this time, and it doesn't have any real value. The system is oscillating. The system is plotted here, and you see it has the same amplitude every time. It does not exponentially decrease amplitude. So, based on the roots that lie in the complex plane, you have the system behaviour. So, roots basically decide the behaviour of your system. So, as soon as you get your system, you quickly find out the roots of its characteristics equation. You can quickly say how my system is going to behave without further looking at the time, domain, solution and plots, so you can directly tell. So, this is the beauty of having such plots.



Now, let us look at the third case when you have real and equal roots. That is a critically damped case. I will tell you what a critically damped thing is. So, this time, you have b square minus 4 km is equal to 0.

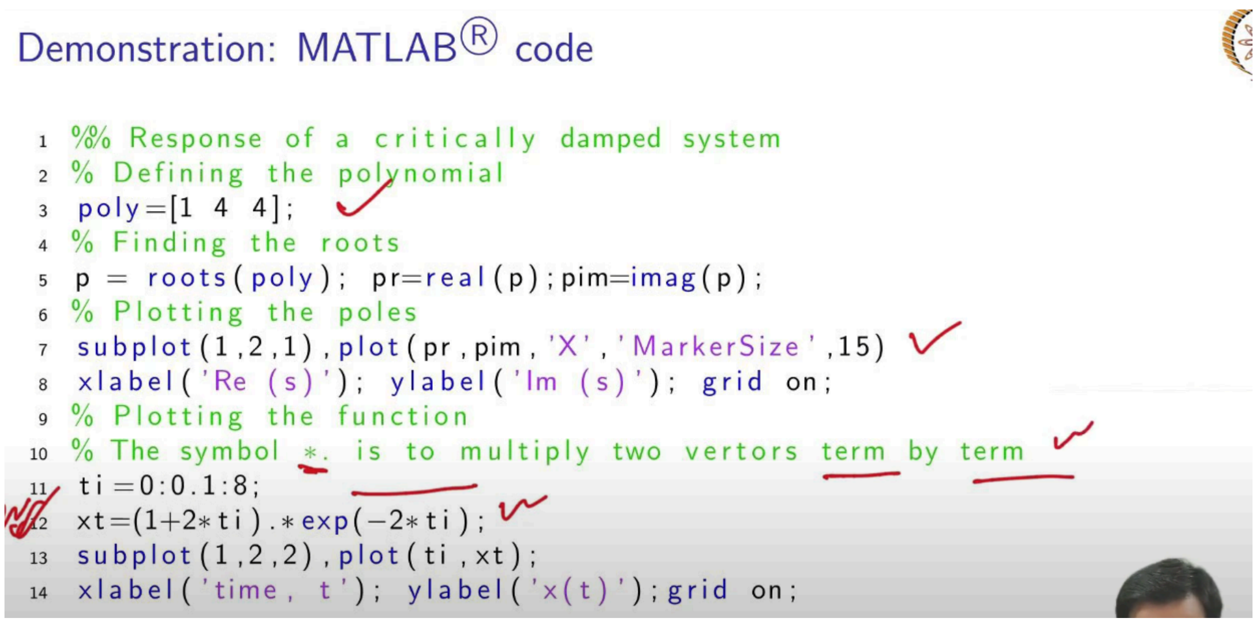
$$b^2 - 4km = 0$$

This time, you have repeated roots: s1 is equal to s2, and the solution by differential equation you already know. So, this x(t) is given by this c1 plus c2t e to the power st, s is nothing but s1 and s2 is s, and that is given by minus b, by 2 m.

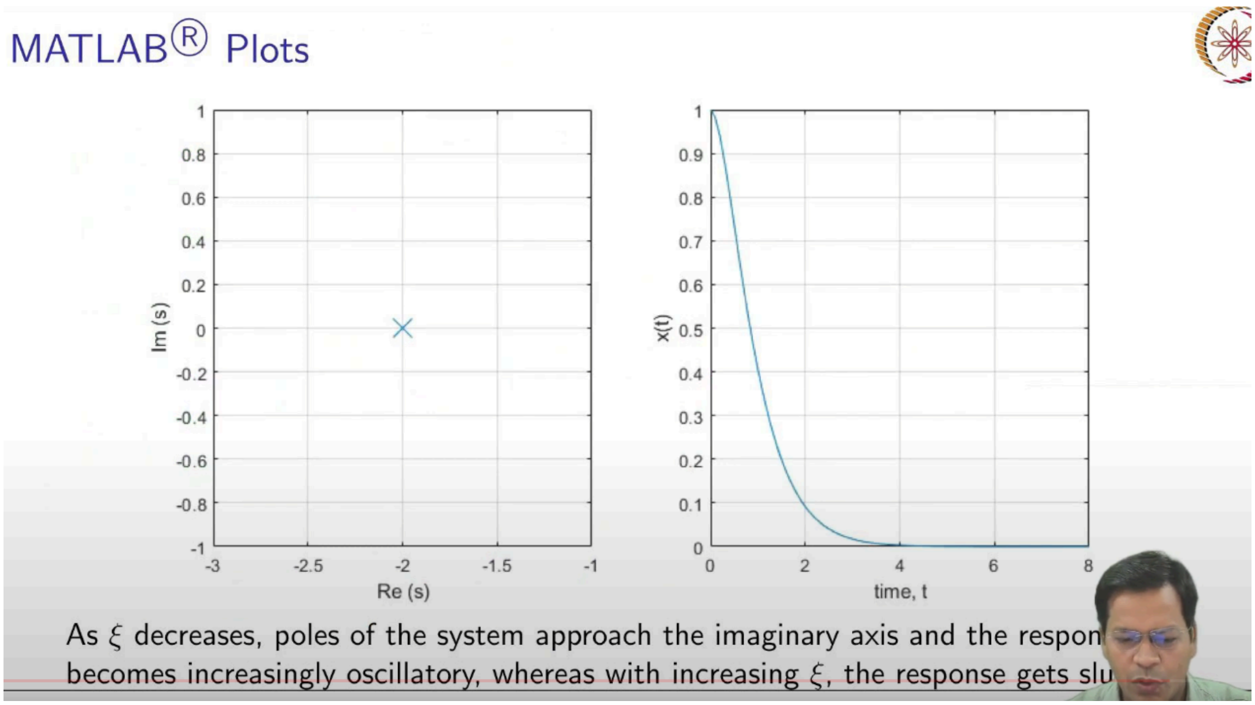
$$x(t) = (c_1 + c_2)e^{st}$$

$$s_1 = s_2 = s = -\frac{b}{2m}$$

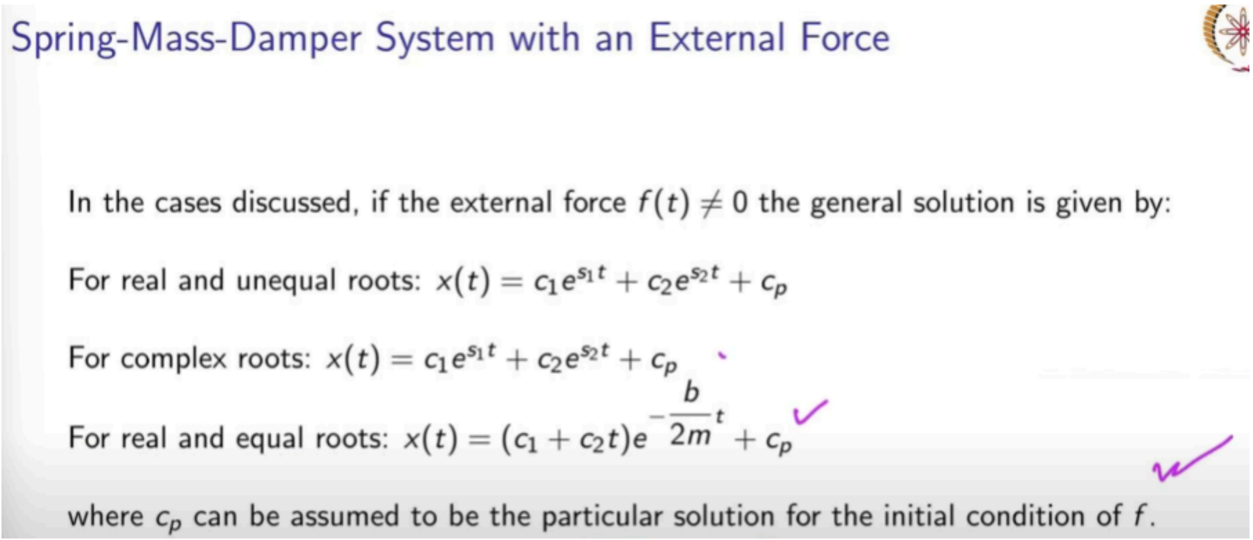
So, that quickly comes here. So, you can directly substitute s in this equation, and you already know c1 and c2 can be obtained by taking the derivative of this, that is, x and x dot t and substituting the boundary condition, you can obtain two equations. Solving those two equations, you can quickly get c1 and c2. So, c1 and c2 are obtained by the boundary condition, the initial conditions, so that is this. At the time, t is equal to 0. so this is the most desirable condition as it is the fastest non-oscillatory response. So, the system is not at all oscillatory in any case, and it quickly comes to 0. So, this is the behaviour you want. You know, you, what you are doing with all this system. You have commanded your robot to go to a place. Your robot may be a second-order system. In this case, you just command your robot to go to a place, and it simply can oscillate and come back to a location, or it can just go to a place and stop there, or it can go to that place very fast and still stop there without oscillating. So, this is the case of a critically damped case, in which it quickly goes to that place and is stopped there. So, this is the behaviour you want. Most of the time, you should be like this. Let us say your system is oscillating. What will happen? You reach a place and oscillate. So, that is very much undesirable because what will you see have been commanded to pick a ball from a table and what will it do? It will simply hit the table because it will overshoot and come back. It will overshoot on the other side again overshoot. So, it can hit the table without actually going to the precise location on top of the ball and holding it. It will hit the table. So, that is what is strictly undesirable in robotics. So, you should either be overdamped, or you can be critically damped. Critically damped is not always possible. So, yes, you can be near to a critically damp. So, this is the most desirable condition. This is what should be obtained through various parameter tuning. We'll see tuning later on also. So, this is a non-oscillatory response, and let us just start with one example once again. So, this is m is equal to 1, b is equal to 4, and k is equal to 4. Your characteristics' equation is like this: now see the roots: roots are equal, that is minus 2, and I have made my time domain solution. Taking the derivative of this as substituting the initial boundary condition, I can get two equations that is, C1 is equal to 1 and minus 2, and C1 plus C2 is equal to 0. So, these two are the simultaneous equations solving this, I got. C2 is equal to 2, so C1 and C2 both are obtained. So, that can now be put here, and you get the complete time domain solution. So, it is your displacement, given over time, and it will vary over time like this. This is how your system will behave so that you can plot this. You can plot your roots also. Roots are nothing but real and equal roots.



So again, I am using a similar MATLAB code. So, this is your system Polynomial. Roots are stored p, so you have taken real I mean imaginary here. You know already that you should not get any imaginary value here and plot your markers. So, that is nothing but your roots that can be plotted in a complex plane. So, this is the first plot and time. This time, I am varying my time from exactly like this, so your time exactly goes like this: it is from 0 to this. So, yes, you see, you have xt, which is defined here. It is 1 plus 2t, and you have exponential variation. That is defined here. So, your xt is defined like this. So, the symbol star dot, especially say, is to multiply the two vector terms by term, so that is the way to program it in MATLAB. So, now I will plot xt versus ti. So, ti will vary along the x-axis, and xt is plotted, so this will show the behaviour of my system with time, and those are labelled here.



So, let me just see the system; what does it look like? So, you know, this time, my roots were exactly lying at the same location, and that is real. So, both the values were in the imaginary plane: it was 0, it was purely real, and both were minus 2. It is shown here. So, again, it is at the left half side of your complex plane, so it is on the left side. Both are equal and real, and your system is critically damped. So, again, you remember this. So, in this case, both are real-left sides in the complex plane- and your system is critically damped. Real and equal gives you this case. So, this is the fastest non-oscillatory response. So, you see, as your damping ratios, ξ decrease, the poles of the system approach the imaginary axis, and the response becomes increasingly oscillatory, whereas, with increasing damping, the response gets sluggish. So, there is a time in between in which you should see a critically damped case. After that, it becomes overdamped. So, this is the set of behaviours that you should see. In all these cases, I have given some examples to make you understand, so you just have to remember how and where you are in the second-order system. You should see those roots. Where does it lie? Okay, so it should lie in the left half side of it, and preferably would prefer to get this, but if not, you should be happy with a critically damped case, or you should be better having an over-damped case, but definitely not an oscillatory nature, and pure oscillatory is definitely very, very bad. So, it will infinitely keep on oscillating. So, those are the undesirable cases.



So, yes, let us move ahead now and see how your system behaves with external forces. This is what your system is, actually. It is not that it is naturally oscillating in some manner with its natural characteristics. So, that is what we have analysed now. So, how does that affect any external force? So, you get to see similar results. So, in the cases as discussed, if an external force is not equal to 0, the general solution in the first case, when real and unequal roots, you should see a similar result. The only thing that will change here is you will see the term which is here.

$$x(t) = c_1 e^{s_1 t} + c_2 e^{s_2 t} + c_p$$

So, apart from the constant which is C1 and C2, this is the additional constant. So, you should know X0, you should also know X dot 0, and at that, t is equal to 0; you should also know how much your f is, what is the force value? So, if you also know the third condition, you can get to the third constant. So, these two will create C1 and C2 variations, whereas this one will directly affect the one, and all three constants will be calculated using three boundary conditions, so that is for real and unequal roots.

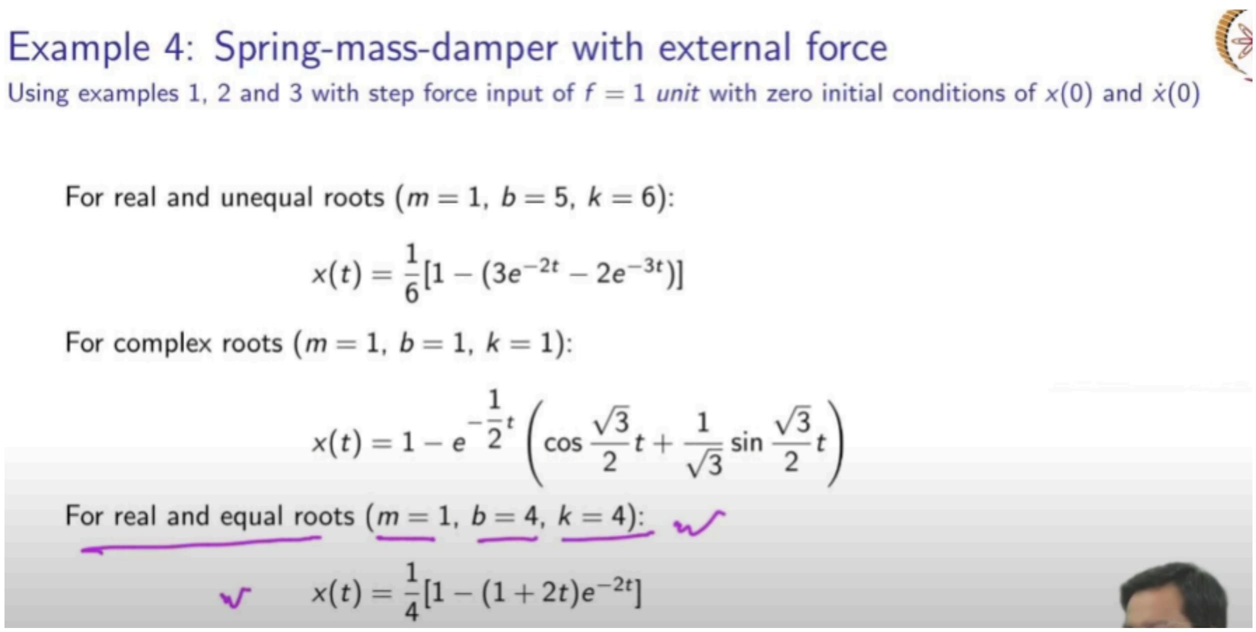
Similarly, for complex roots, you should see one similar equation, with Cp coming here;

$$x(t) = c_1 e^{s_1 t} + c_2 e^{s_2 t} + c_p$$

Again, the same parameters are required to find out these constants, S1 and S2 being the roots of the characteristics' equation. So, your system characteristics equation remains the same so that it is not affected by the external force. So, that is why it is external, and in the case of real and equal roots, you see this equation again once again.

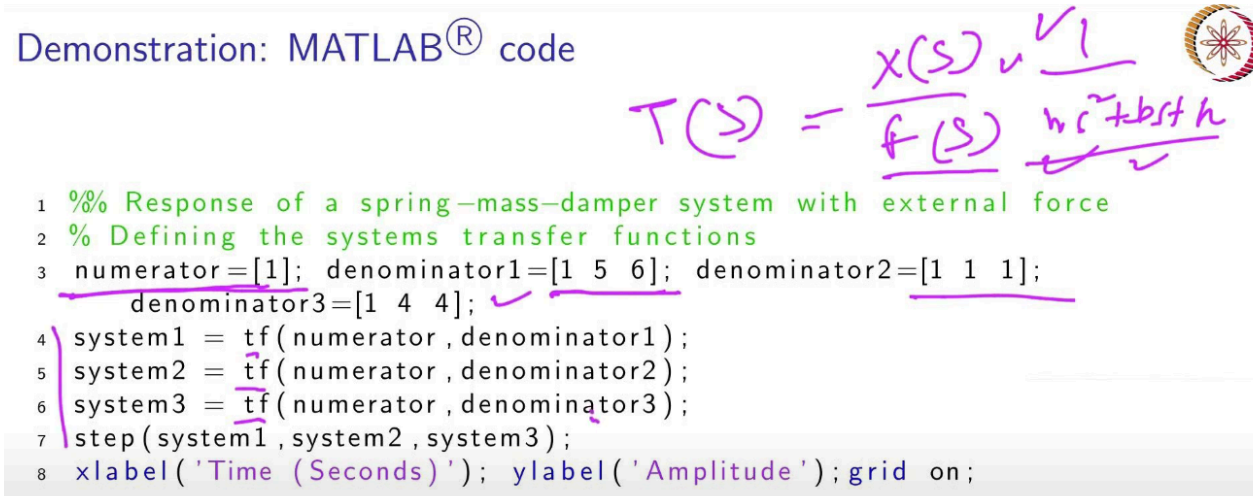
$$x(t) = (c_1 + c_2 t) e^{-\frac{b}{2m} t} + c_p$$

And this is your Cp. So, Cp can be calculated by using all those initial parameters. So, yes, Cp can be assumed to be a particular solution for the initial condition of f, so that is, with the initial condition of f, you can find out the value of Cp in all these.



Let us just see a few examples. So, for real and unequal roots, the equation that we have taken, we started with the same example. I have put once again over here: m is equal to 1, b is equal to 5, and k is equal to 6. The characteristics equation remains the same. So, the roots go here, and S1 and S2 will come here. The only thing that has changed is the Cp value if you relate it to the previous one. So, Cp can directly come to the other side of the equation. I can write it like this: so this is your Cp value. That has come here, so this becomes your time-domain solution. Again. In the case of complex roots, again, you can write your equation like this: you have lambda plus minus I μ so that the μ term will go here, lambda term will go here, and Cp comes here, and the time domain solution will be like this. So, by plotting these, you can see the influence of external force and how your system behaves with that external force.

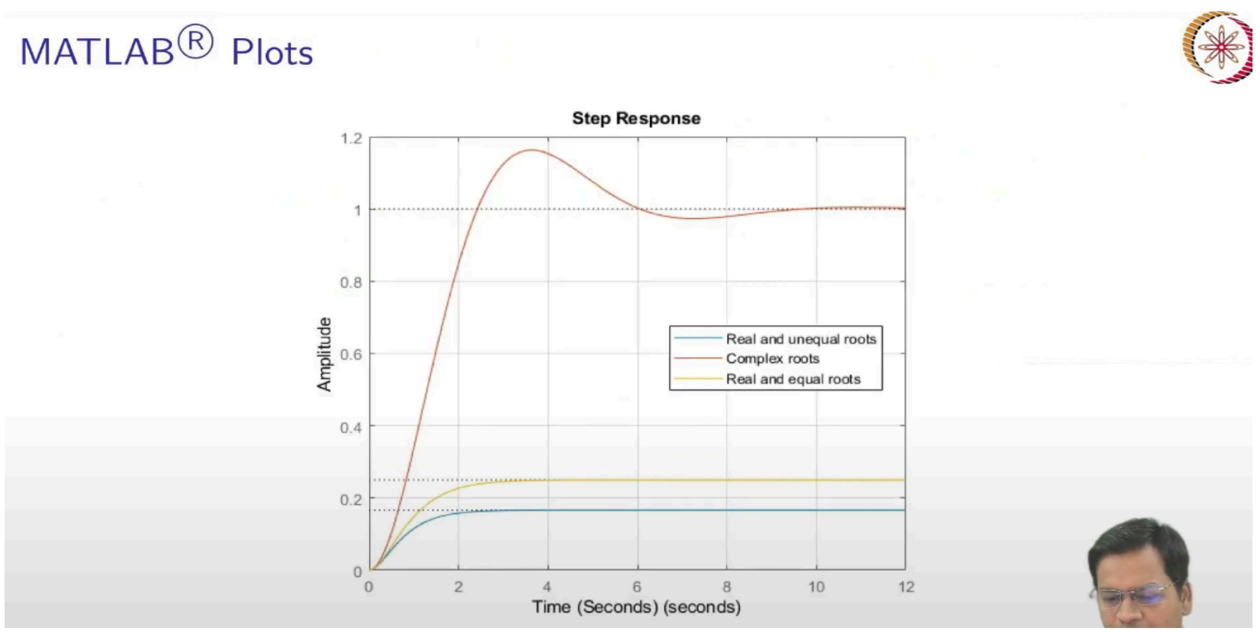
Similarly, for real and equal roots, m is equal to 1, b is equal to 4, and k is equal to 4. Roots are equal, and they are b square minus 4 Ac is equal to 0. In that case, you get to see through this equation. This should be a critically damped case once again, and then this is your Cp value. That comes here, and everything is like this: so now I will plot all of them.



So, this time, I am using a different way to plot it. I am defining it as a transfer function, the way we discussed it in the first class, so this is your first numerator denominator that is basically X(s), which is the output by F(s). So, this becomes your transfer function. So, X(s) is the numerator. So, numerator, you have only one denominator. You see, you have F(s) output by input. So, that is the transfer function. So, you see, it is X(s) by F(s). So, that should tell you, 1 by ms square plus bs plus k. So, that basically gives you the transfer function.

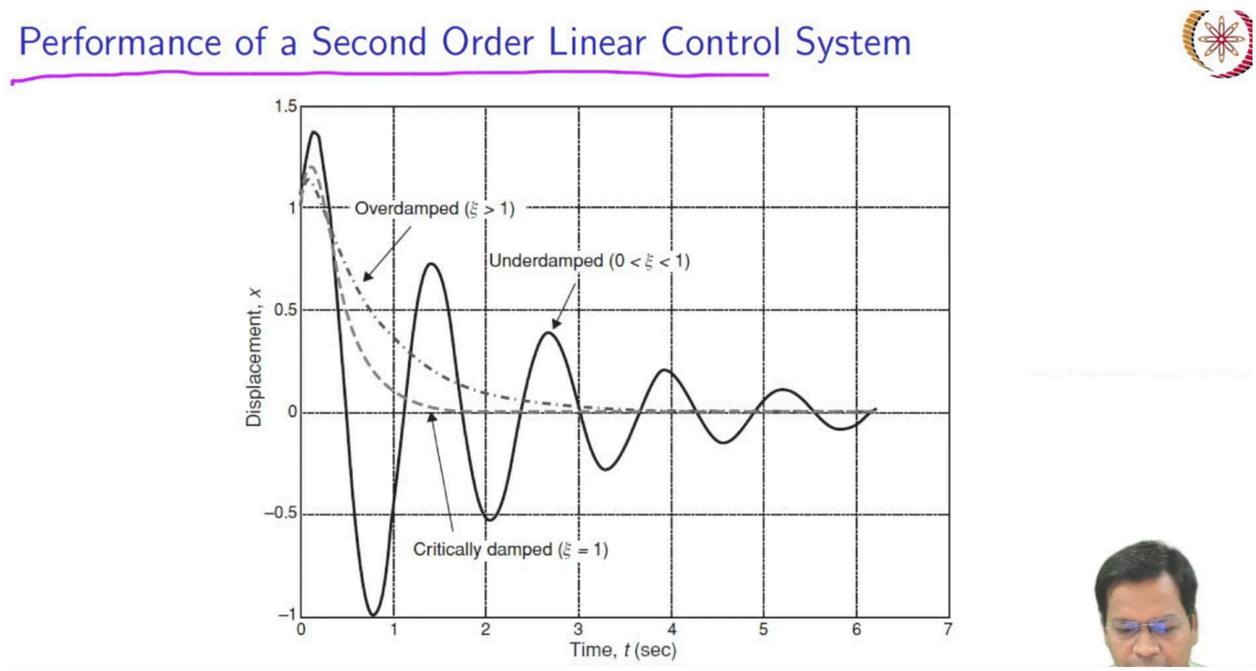
$$T(s) \;=\; \frac{X(s)}{F(s)} = \frac{1}{ms^2+bs+k}$$

So, now this becomes your numerator is your denominator. For the numerator you have the same values for all the cases, whereas this will change. So, that is the denominator for the first case, for the second case and for the third case. So, you have three systems. All of these can be plotted together in MATLAB. I am labelling it like this.



So, I am plotting all the systems together here. So, this was a system with real and unequal roots.

This is the one where you have complex roots. It is oscillatory in nature. It has overshot, it has come back, and it is like this. And the third one is real and equal roots. So, this is the critical condition in which it settles down very fast. So, you see, you have an external force, and with time, it settles to a value. This time, it won't settle for a value which is zero. This time, it gradually came to rest. That is here. This is a step-input response. So, the step was just one unit. So, you quickly reach the one which is here, and you just overshoot. It came back. Ultimately, you settle somewhere over one for different roots. This is the thing which will basically define your system and this is the behaviour of your system. So, these are the MATLAB plots. You can do that yourself also in octave or maybe in Python.



This is what is the Performance of your control system. So, you can quickly write it as displacement plotted here, and your time varies like this: and you see you have over damped case for ξ greater than 1, under damped case for ξ lying between 0 and 1. This is the case which was critically damped case in which it is equal to ξ is equal to 1. So, in that case, it is a critical time, and in this case system settles to the desired value very quickly. So, this is the fastest, non-oscillatory response. So, everything can be seen here. So, this is the plot of the system. So, you already know where your roots lie. So you can, based on the roots and their location in the complex plane, you can decide how your system is going to behave. So, you know how to get the characteristics equation of your system, and you can find out the roots. You can say how my system is.

So that's all for today. In the next class, we'll see the transfer function approach and state space approach to analysing your system. We'll also look at a DC motor model of a robot joint, and we'll discuss it further. So, yes, that's all for today. Thanks a lot.