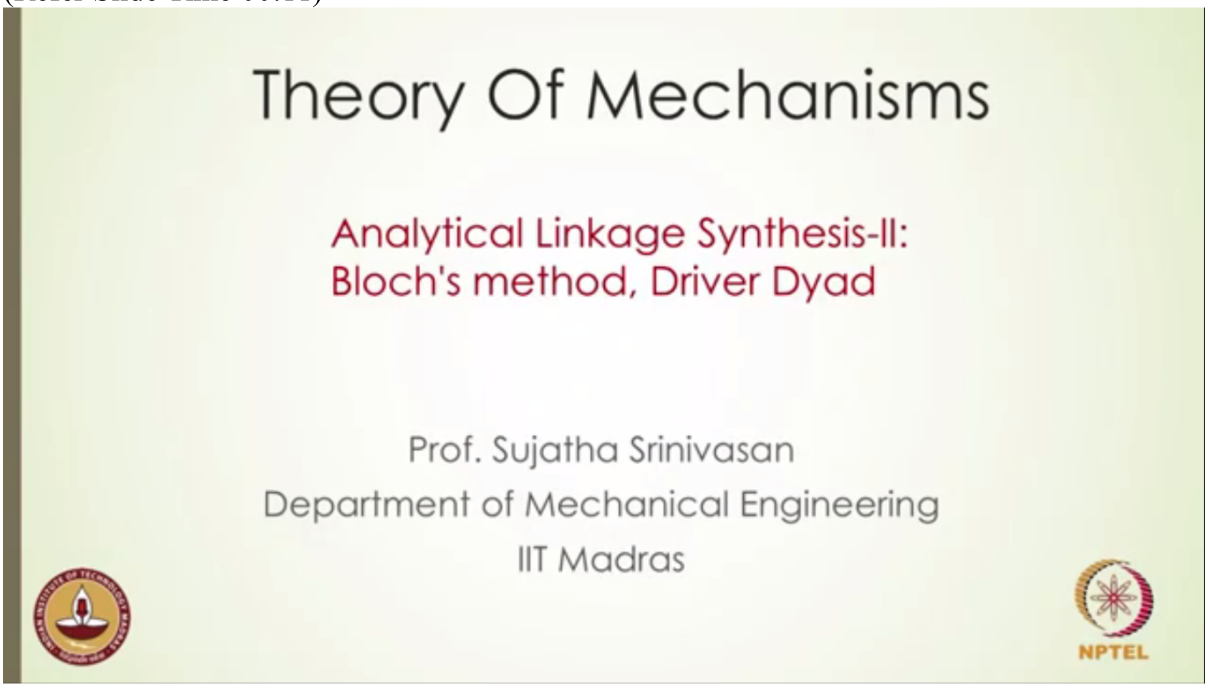
(Refer Slide Time 00:08)

NPTEL
NPTEL ONLINE COURSE

(Refer Slide Time 00:11)

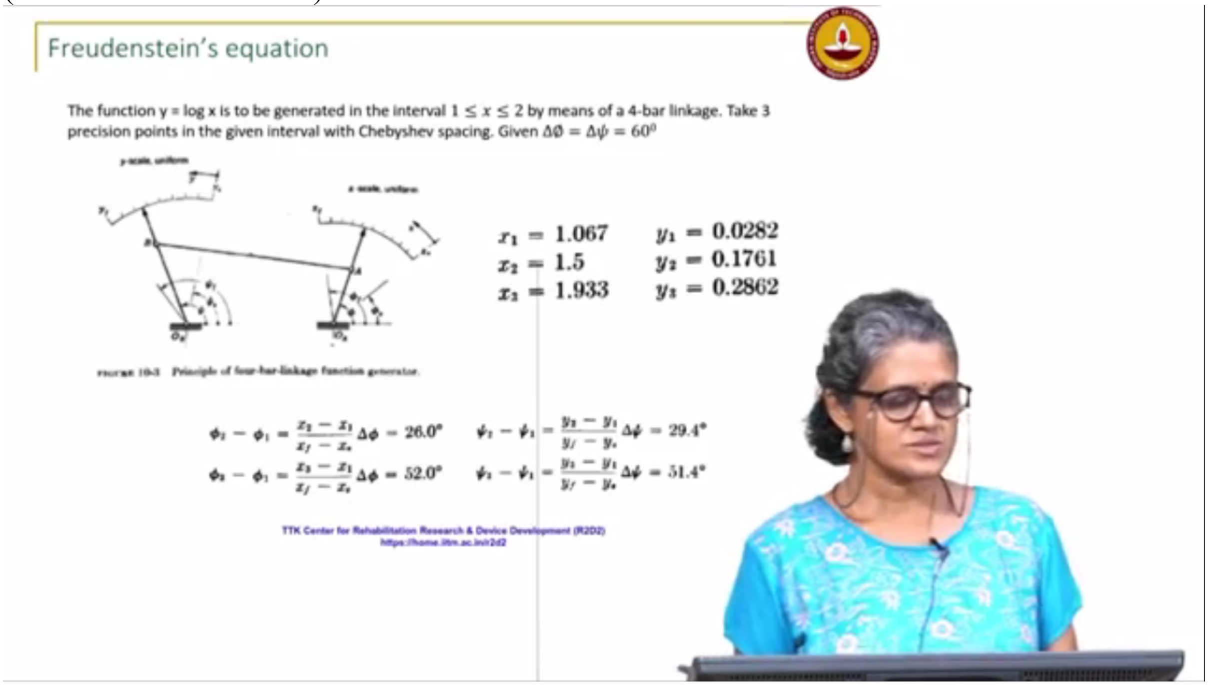


Theory of Mechanisms

Analytical Linkage Synthesis-II:
Bloch's method, Driver Dyad

Prof. Sujatha Srinivasan
Department Of Mechanical Engineering
IIT Madras

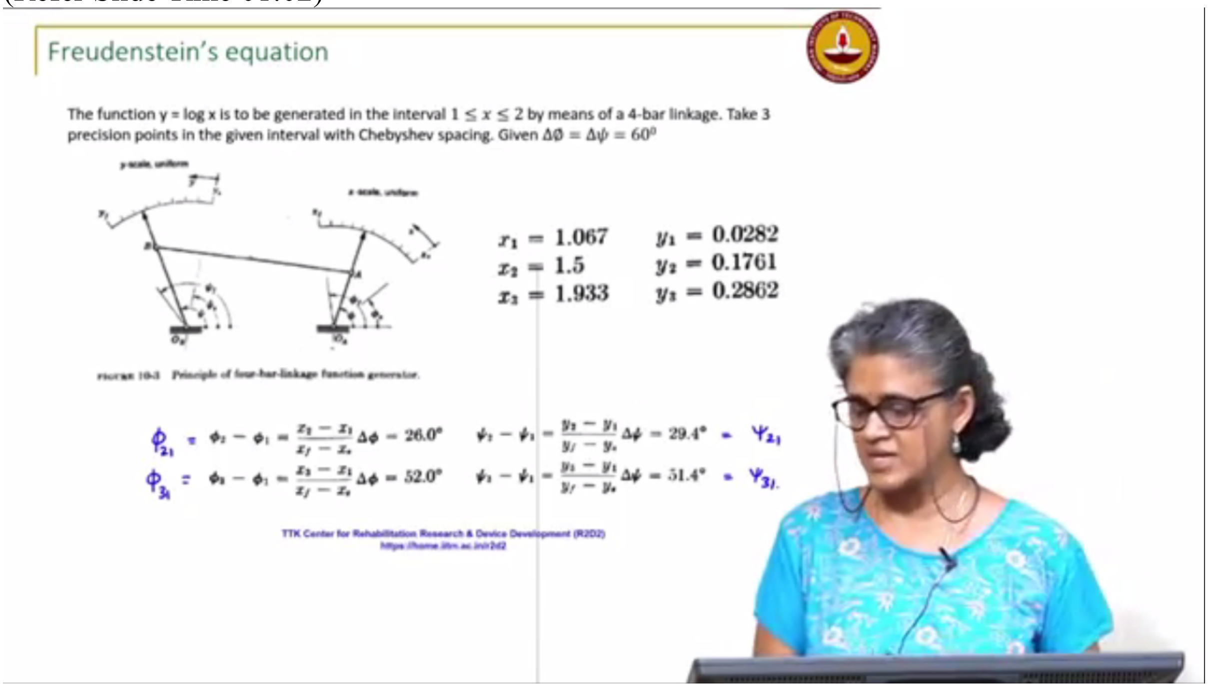
(Refer Slide Time 00:13)



So we look at an analytical method for synthesis of a function generator. So, typically, something like this. So you want to generate a 4-bar with a uniform X scale and a Y scale, and you take the input angles to be proportional to those. So say this is with the three precision points.

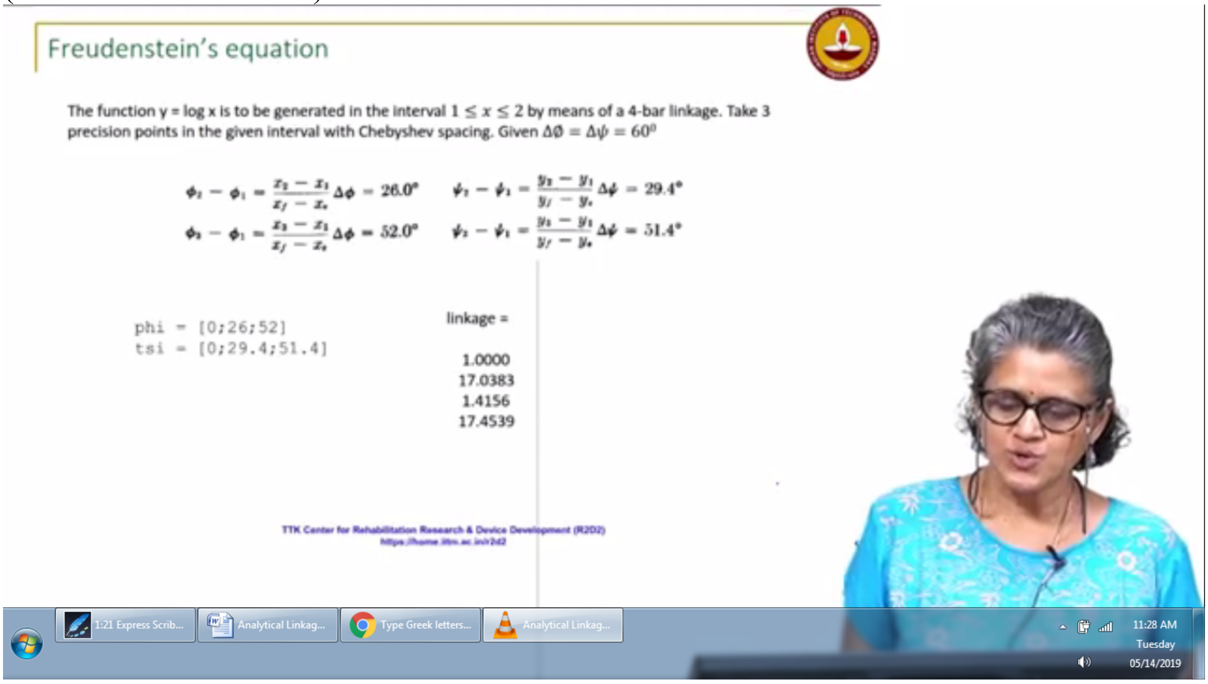
So this would, so from the equations for the Chebyshev's spacing, these are the design criteria you get. Okay. So you get φ_{21}, what we normally refer to as φ_{21}, φ_{31}. This is ψ_{21}, ψ_{31}. Okay.

(Refer Slide Time 01:02)



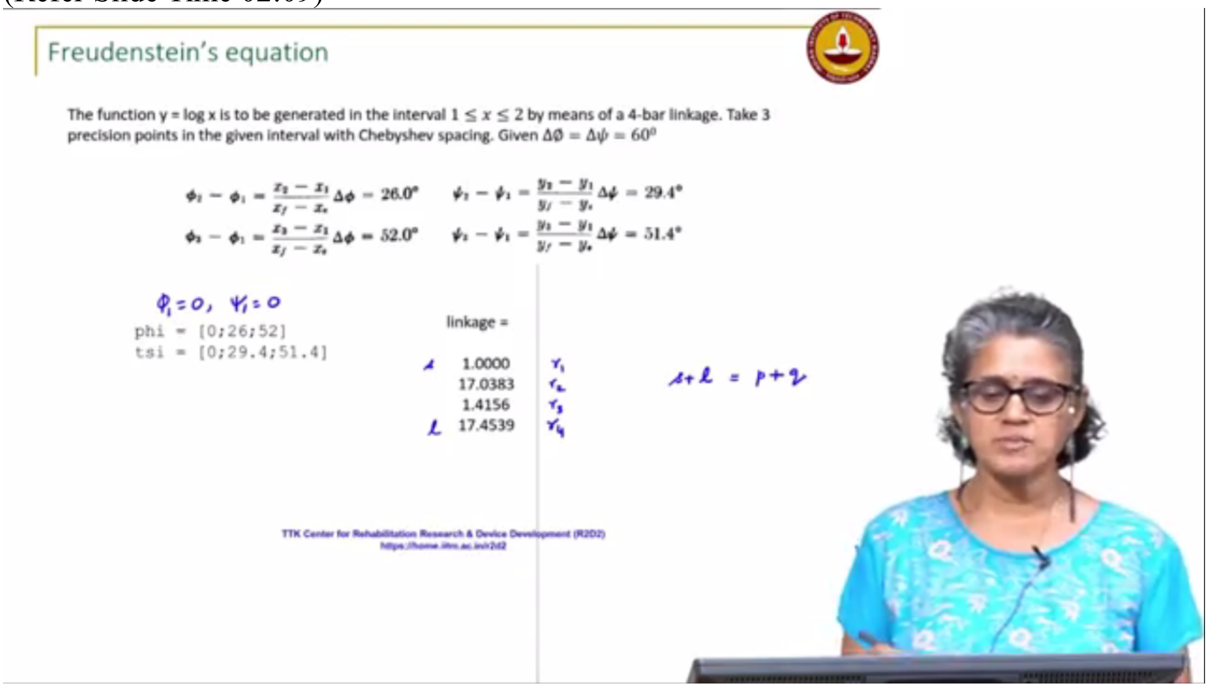
So we want to design a function generator for this. I just wanted to show you, I will have to figure out how to do MATLAB online here, but so I just ran it offline to show you some options.

(Refer Slide Time 01:18)



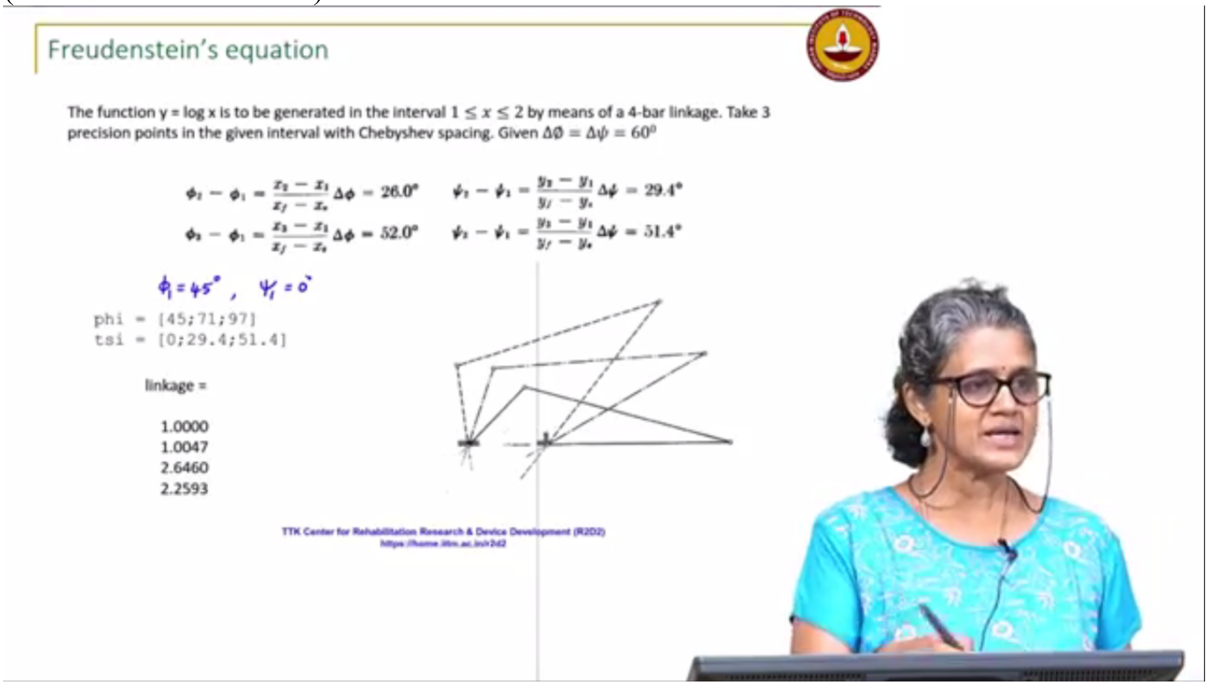
So here in the first case, so I have taken $\varphi_1 = 0$, $\psi_1 = 0$. Okay. And I end up with the link lengths like this. So this is R_1, R_2, R_3, R_4. You can see they are very disproportionate, 17 times. And you can also see if I do, this is the smallest link s, l, s + l = p + q. So you get the change point. You get a Grashof neutral linkage because you chose $\varphi_1 = 0$, $\psi_1 = 0$, the first precision point, right? You get a change point there.

(Refer Slide Time 02:09)



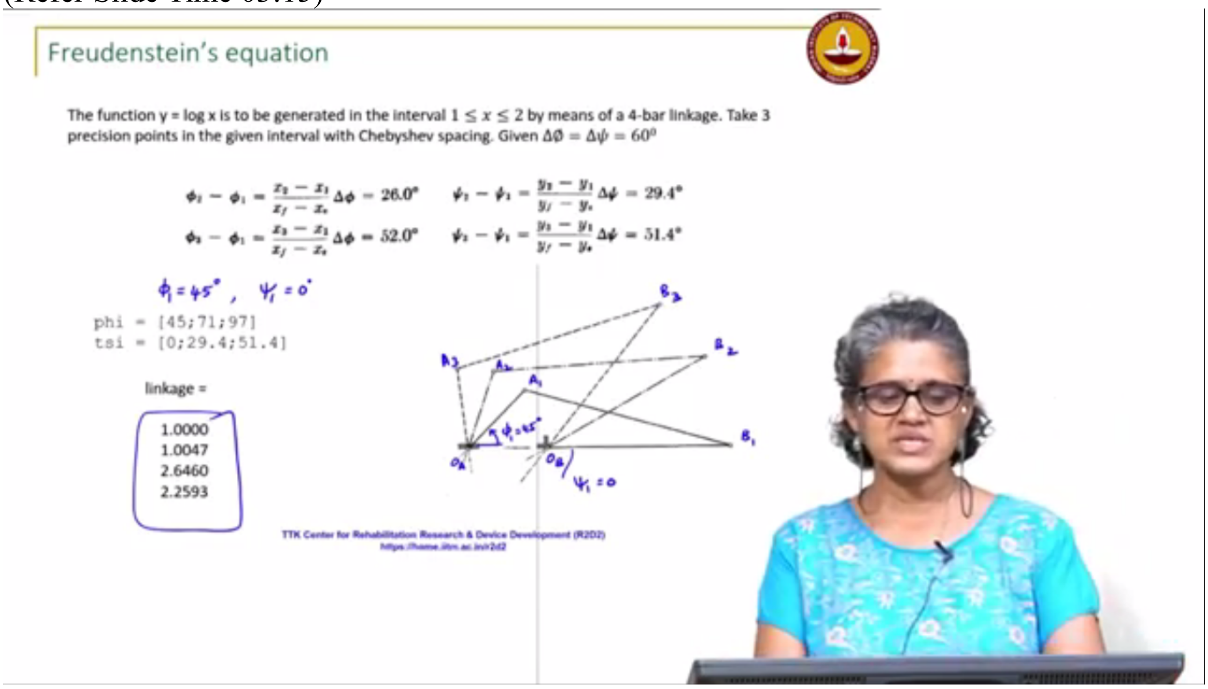
So this is probably not a desirable solution. So I change my conditions a bit and accordingly the scale will change. So I would use this equation. So here I keep φ_1. I change φ_1 to 45 degrees. I retain ψ_1 as 0. Okay.

(Refer Slide Time 02:36)



Then I ran my code again. Okay. And I get a much better linkage for it. Okay. So this is my, this would be the first position A1, B1. You can see this is $\varphi_1 = 45$. This is $\psi_1 = 0$. This is O_A, O_B. This would be A2, B2, A3, B3.

(Refer Slide Time 03:15)



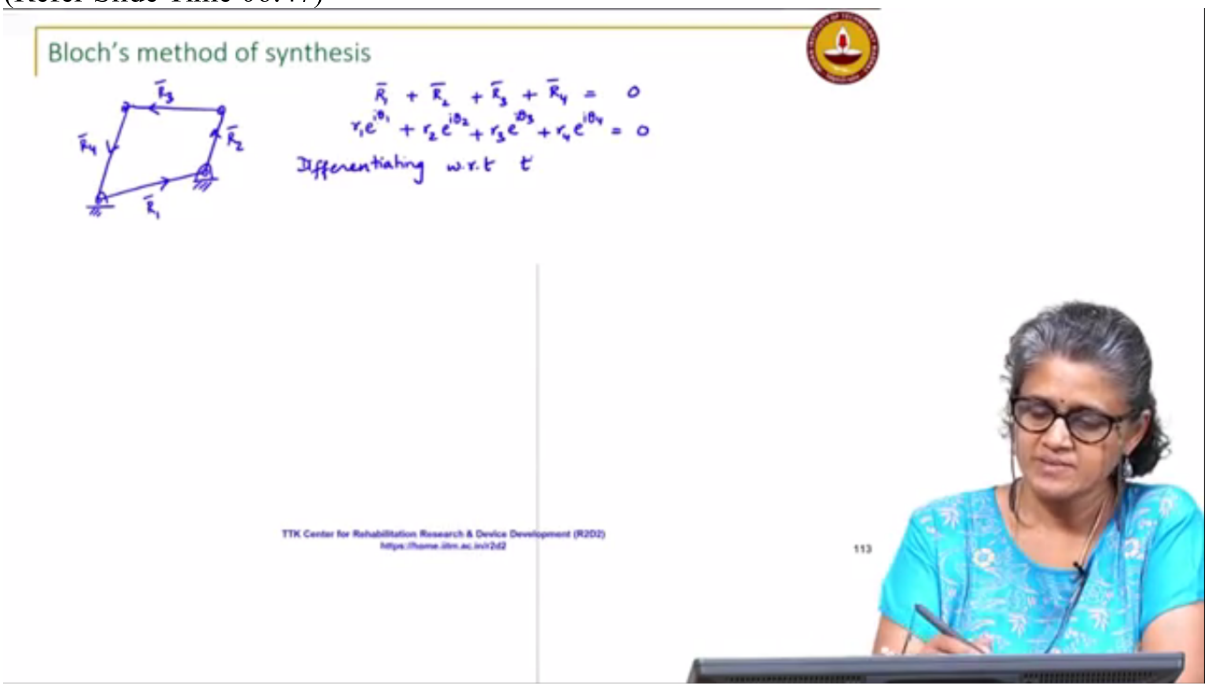
So this would be the example of using Freudenstein's equation using a programming language like MATLAB to do your synthesis. I could have done this graphically also, but it's harder to do to cycle through multiple solutions.

Also in the graphical case if you noticed once we specified one link length, especially for three position synthesis, I didn't really have control over the initial orientation of the output link because of the method that I used. It has more to do with, you know, when I used inversion, I found that I couldn't really specify. I had to find the intersection of the two perpendicular bisectors which is a point which is not necessarily on what I want ψ_1 to be. Okay.

So that's a solution I get, but here I am able to specify. So the geometric methods may have limitations in terms of because of the parameters I have to specify for the construction. I may be limited in some way. In this case I am able to specify both the precision points. There in most cases I only worked with the two displacements, angular displacements and I specified only one of the angles, initial angles. Okay. So that is something to keep in mind.

So we will now move on to Bloch's synthesis, which we started yesterday. So I have, you take a general, so if I take these vectors say R_1, R_2, R_3 And R_4, okay, and say this is my, R1 is my fixed link, so the first equation that I had was $R_1 + R_2 + R_3$ (my loop closure) $+ R_4 = 0$ and I wrote this in complex form to get $r_1e^{i\theta 1} + r_2e^{i\theta 2} + r_3e^{i\theta 3} + r_4e^{i\theta 4} = 0$. I differentiated this with respect to t because now I want to perhaps specify conditions on the derivatives of the linkage. At a particular position, I wanted to have a certain input and output angular velocity or a certain input and angular acceleration.

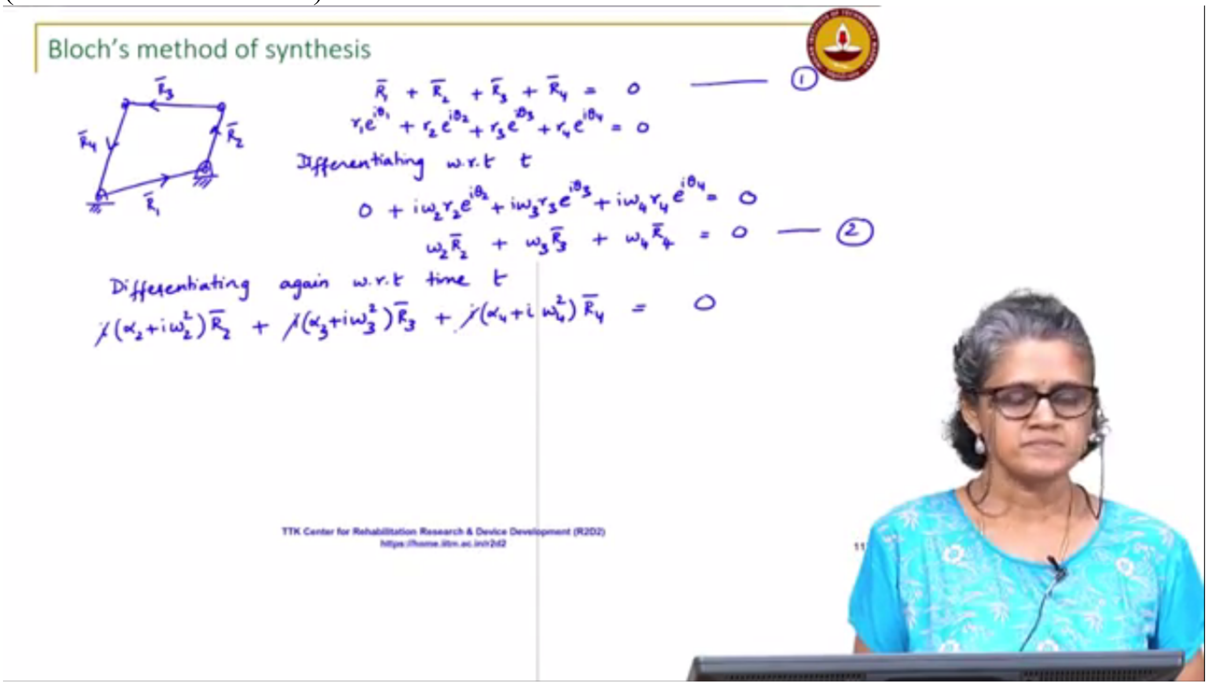
(Refer Slide Time 06:47)



So I get differentiating. So we saw that I will get $i\omega_1$, obviously, the first term will be 0. r1 does not change, so I get $i\omega_2r_2e^{i\theta 2} + i\omega_3r_3e^{i\theta 3} + i\omega_4r_4e^{i\theta 4} = 0$. Okay. i, of course, goes away, so I can write this as $\omega_2r_2 + \omega_3r_3 + \omega_4r_4 = 0$, right, in vector form.

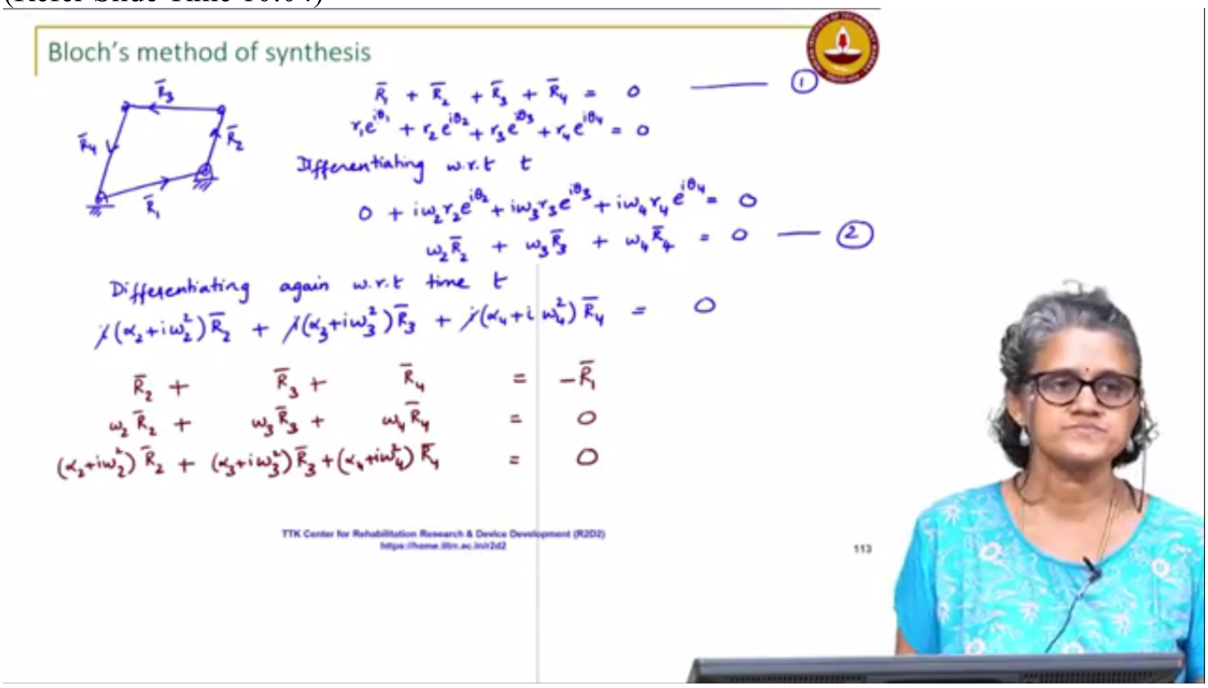
So this is one equation. This is the other equation, and then I differentiating, I differentiate it again with respect to time. I get $i(\alpha_2+i\omega_2^2)R_2 + i(\alpha_3+i\omega_3^2)R_3 + i(\alpha_4+i\omega_4^2)R_4 = 0$. Again, I can knock of the i's.

(Refer Slide Time 09:01)



Okay. So now I have a set of three equations like this. I have R_2 in these three unknowns. I want to synthesize the linkage. So R_2, R_3, the links are my unknowns. This is equal to 0. $\omega_2 R_2 + \omega_3 R_3 + \omega_4 R_4 = 0$ and then $(\alpha_2+i\omega_2^2)R_2$, sorry, this is not equal to 0; this is $-R_1$; $i\omega_3^2)R_3 + (\alpha_4+i\omega_4^2)R_4 = 0$.

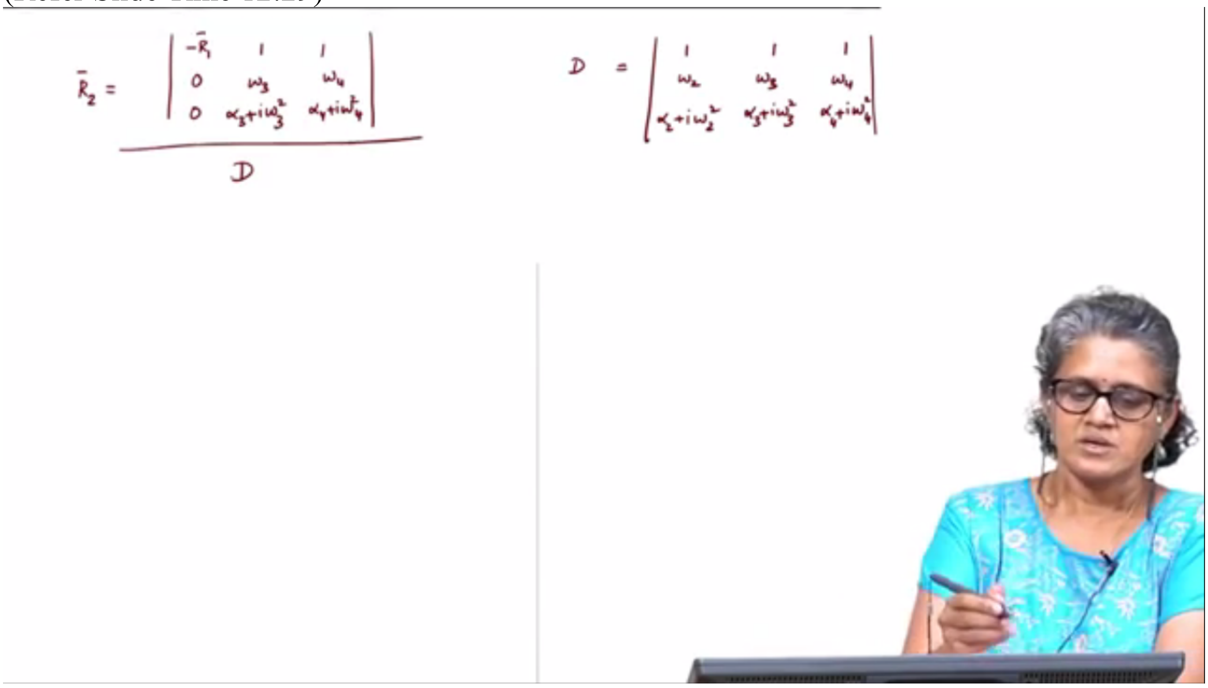
(Refer Slide Time 10:04)



So if I specify, if I choose R_1, okay, I get a set of linear equations in R_2, R_3 and R_4. Okay. They have complex coefficients, but nevertheless it's a set of linear equations in R_2, R_3 and R_4. So I can choose some R_1 again and in case of angular quantities, I can scale, shouldn't be a problem. I should be able to get a linkage to satisfy my design conditions. Okay.

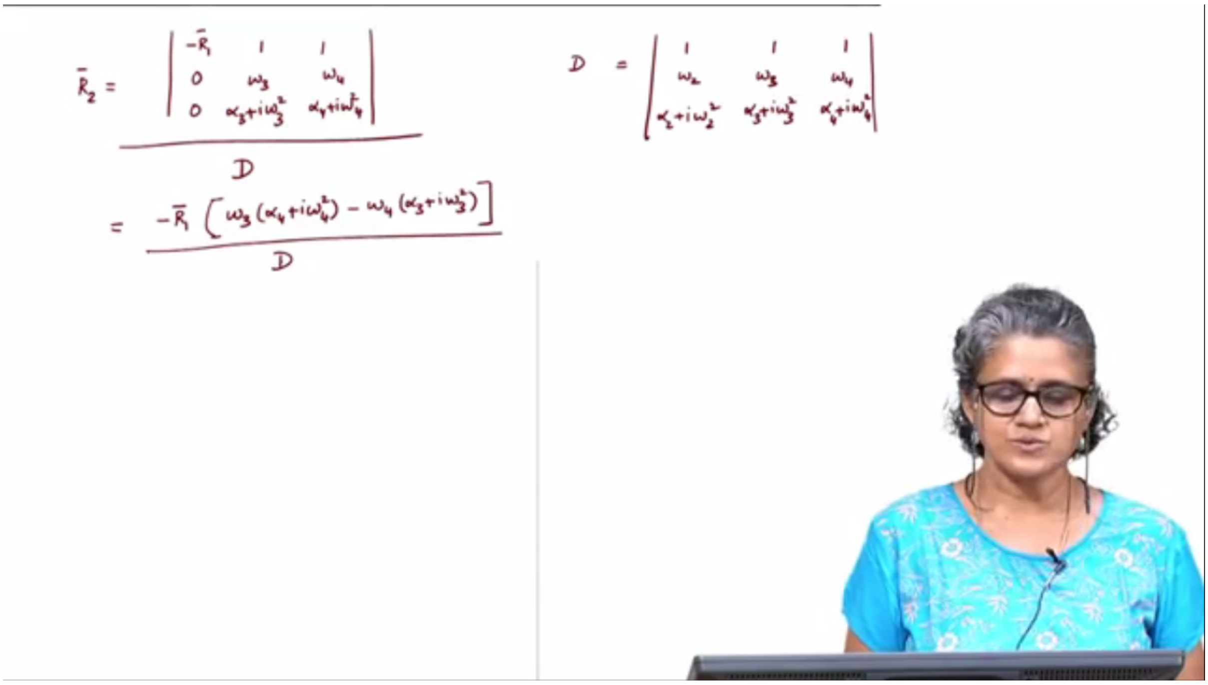
So now if I, I can write this, you know, if I write the solution in determinant form, R_2 will be equal to $-R_1$ 1 1, 0, 0, 1, ω_3, $\alpha_3+i\omega_3^2$, ω_4, $\alpha_4+i\omega_4^2$. That will be the numerator and the denominator will be common for all of them. It will just be, so if I take denominator determinant D will be the determinant of 1, ω_2, $\alpha_2+i\omega_2^2$, 1, ω_3, $\alpha_3+i\omega_3^2$, 1, ω_4, $\alpha_4+i\omega_4^2$. This will be my determinant. Okay. This will be D and D is also going to be a complex number because there are complex coefficients in the determinant.

(Refer Slide Time 12:29)



If I expand this, I get $-R_1(\omega_3(\alpha_4+i\omega_4^2) - \omega_4(\alpha_3+i\omega_3^2))/D$. Okay. And similarly, I will get for the other two, R_2.

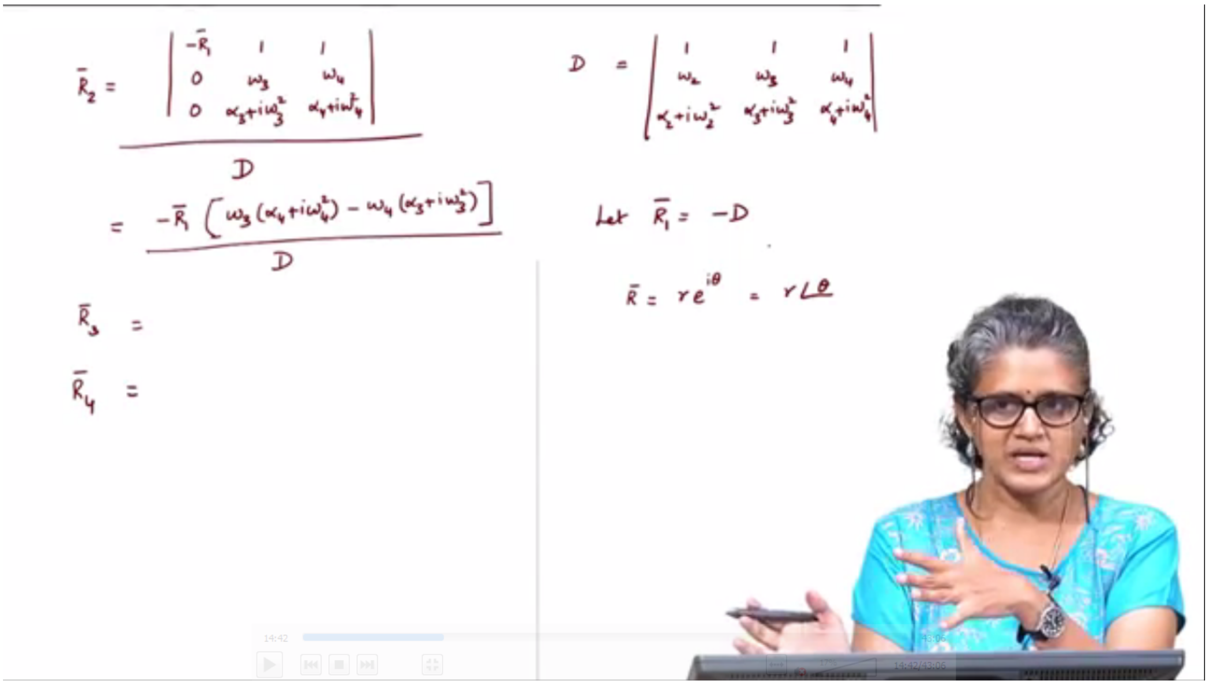
(Refer Slide Time 13:06)



Now my R_1 is a free choice. So one way I can simplify my calculations is to say that R_1 is proportional to D. So say I say let R_1 = -D. D is a complex number, right? So that way I eliminate this and I have only this to evaluate, becomes a lot easier to evaluate that.

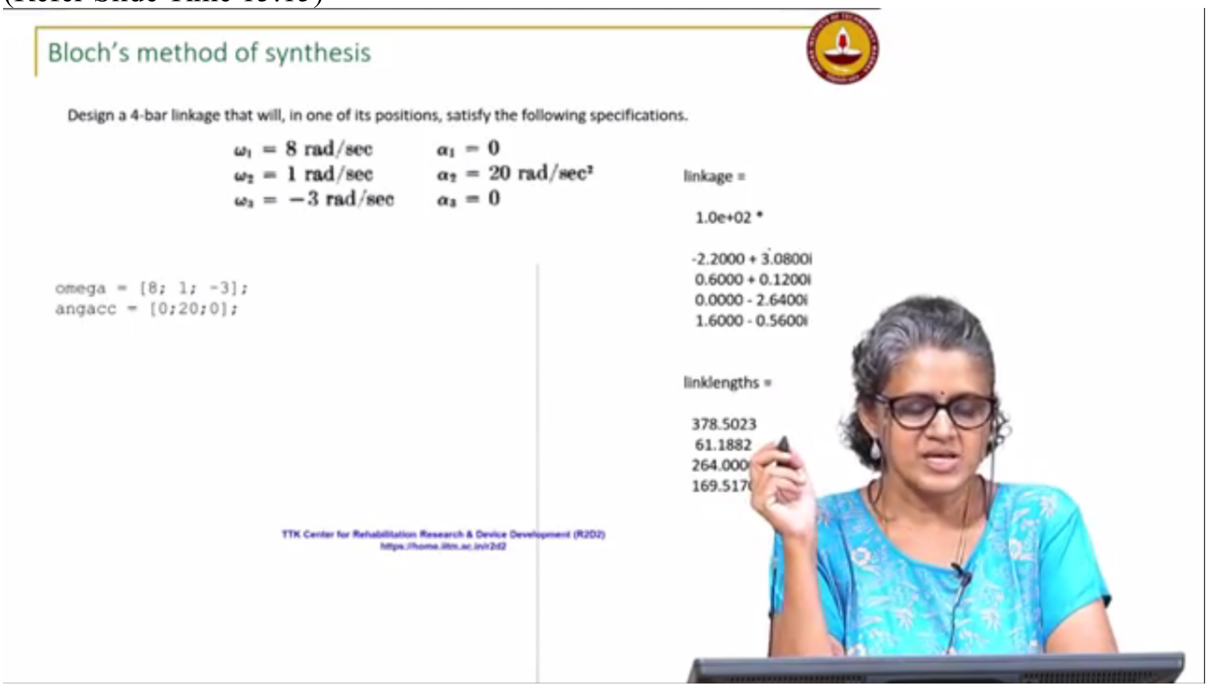
Similarly, I'll have for R_3 and R_4. I'll let you write that out because I'm going to have you do an assignment on programming this. Okay. And then you can verify with the example that I'll show you. So if I choose, so then it just becomes a matter of if I want my R_1 to be horizontal, okay, which is what I started off with. I would basically have to rotate it because any complex number I can write in terms of, so if I have a complex number R, it is some $re^{i\theta}$ where θ is the angle it makes with the positive x-axis, right? The magnitude and the angle in polar coordinates, right? So this would be just $r\angle\theta$ in polar coordinates.

(Refer Slide Time 14:40)



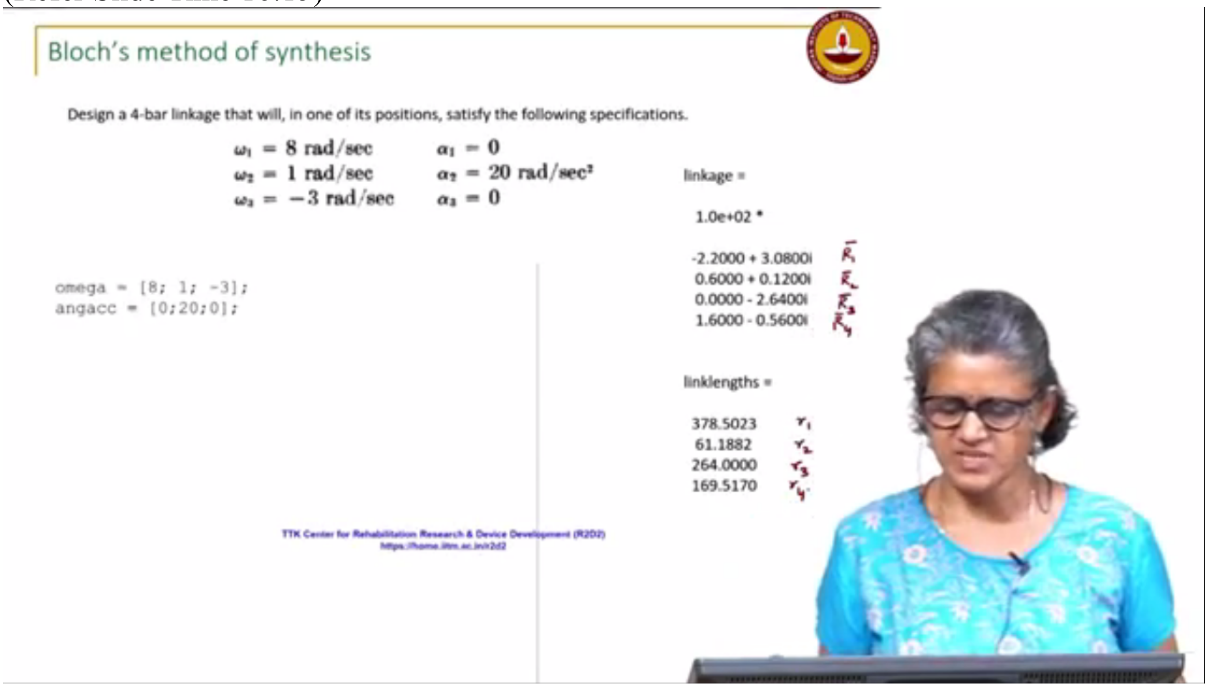
I choose some R_1 to match, so if I choose it to match the denominator, then my calculations become simple. So I can find an R_2, R_3, R_4 in this manner and then, of course, once I find the linkage, we will do the example, which will show you what it looks like what you'll get when you solve it this way, and then how you would get your the configuration that you desire.

(Refer Slide Time 15:15)



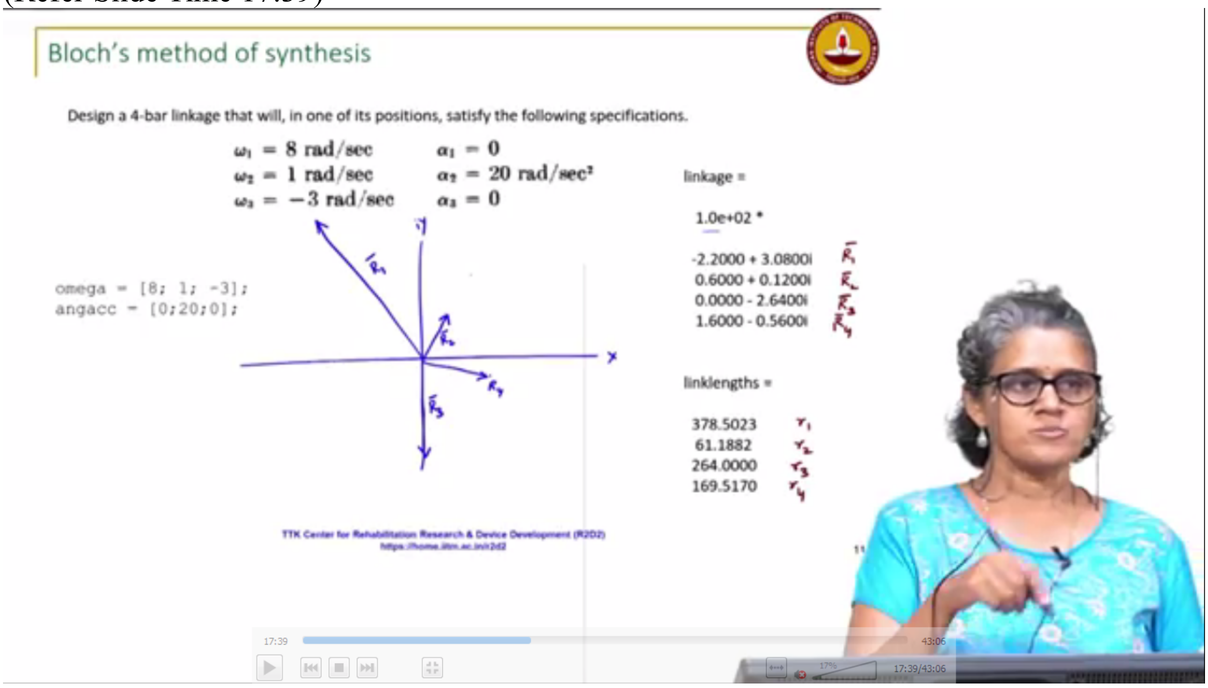
So this is a problem. You design a 4-bar linkage. In one of its positions, you want it to satisfy the following specifications for ω_1, ω_2, ω_3, α_1, α_2, α_3. Okay. So I would program. If I use Bloch's method, then these are my initial conditions and I end up with a linkage. This is my solution from MATLAB and I get these vectors. So this is R_1, R_2, R_3, R_4. These are my link lengths, which are basically the modulus of that, those vectors. Okay. So I get these link lengths.

(Refer Slide Time 16:15)



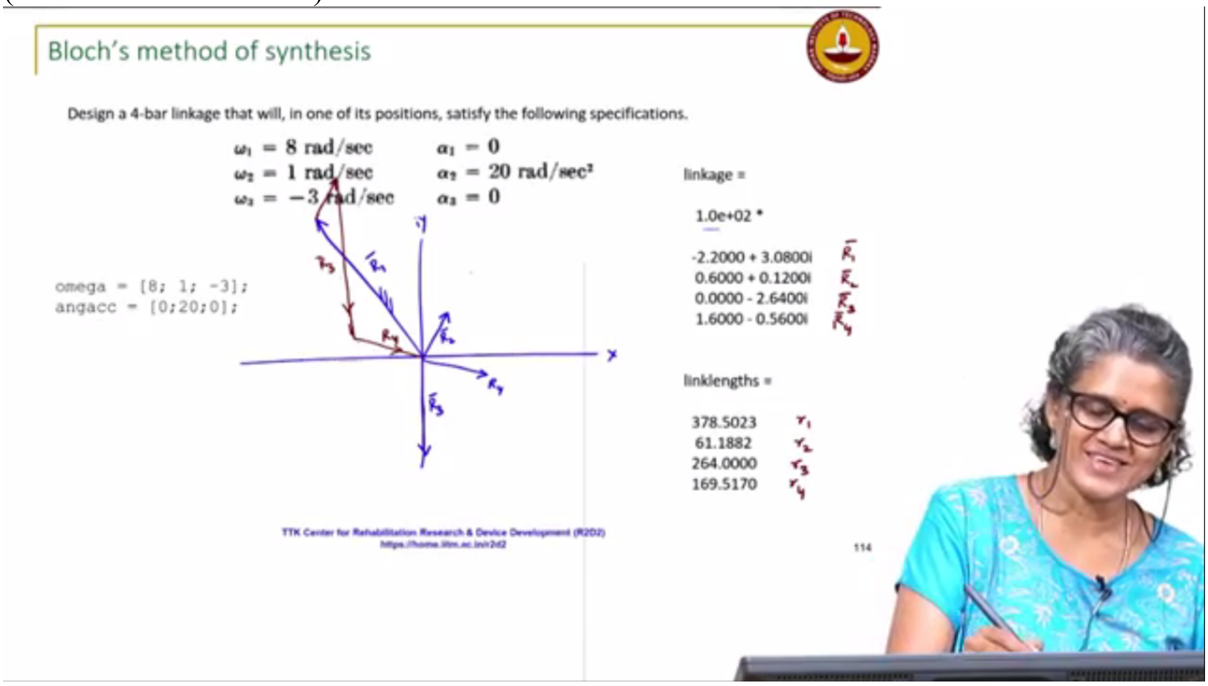
Now let me see what my linkage looks like. So if this is my complex plane R_1 is -2 point, let's say I take the scale to be 1 equals 100 here because it's that times. So I have -2.2 + 3.08. That's the largest link. So it would be somewhere here. That would be my R_1. R_2, 0.6 + 0.12, so I have it will be in the first quadrant, much smaller. So let's say this is R_2. R_3 is along the negative Y-axis and my R_4 is in the third quadrant again. So let's say this is R_4. R_4 is smaller than R_3.

(Refer Slide Time 17:39)



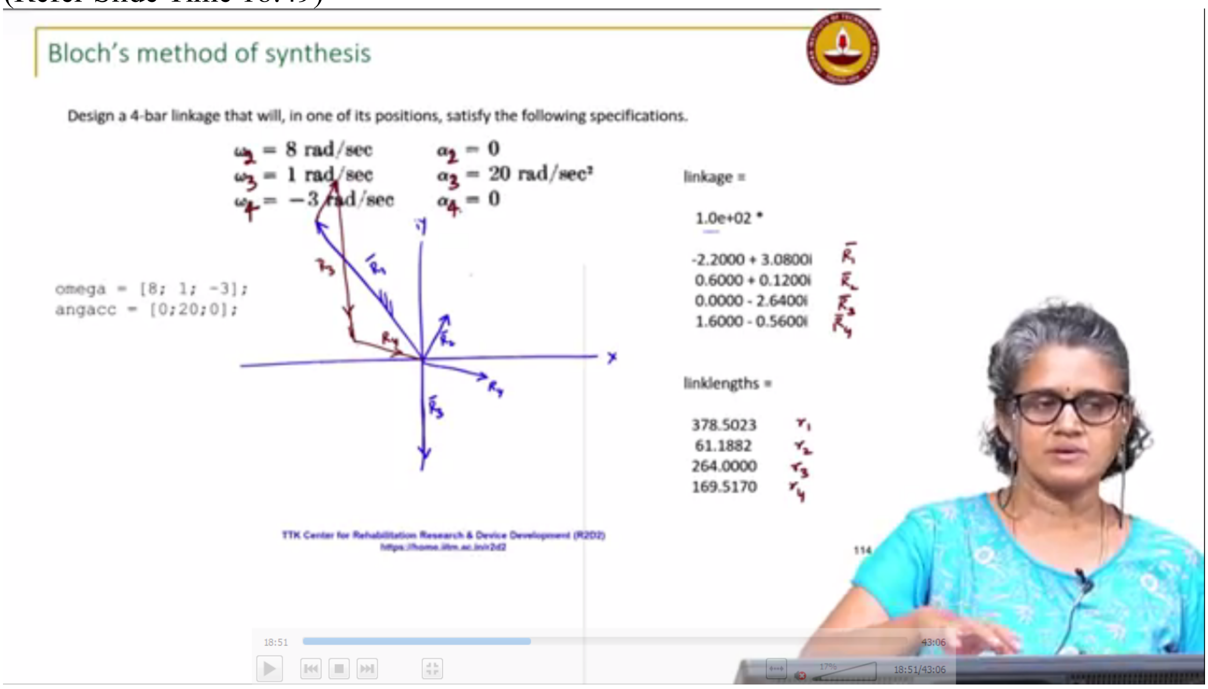
So now I have to form the loop with this. Okay. So this is my R_1, which is the fixed link. Okay. So my loop becomes this is R_2. So I move it. That would be my R_2. My R_3 comes straight down. Okay. So this is my R_3 because I have to satisfy my loop closure equation. $R_1+R_2+R_3$, then, okay, this should be R_4. This should be parallel to this, but yeah, so this is R_4.

(Refer Slide Time 18:32)



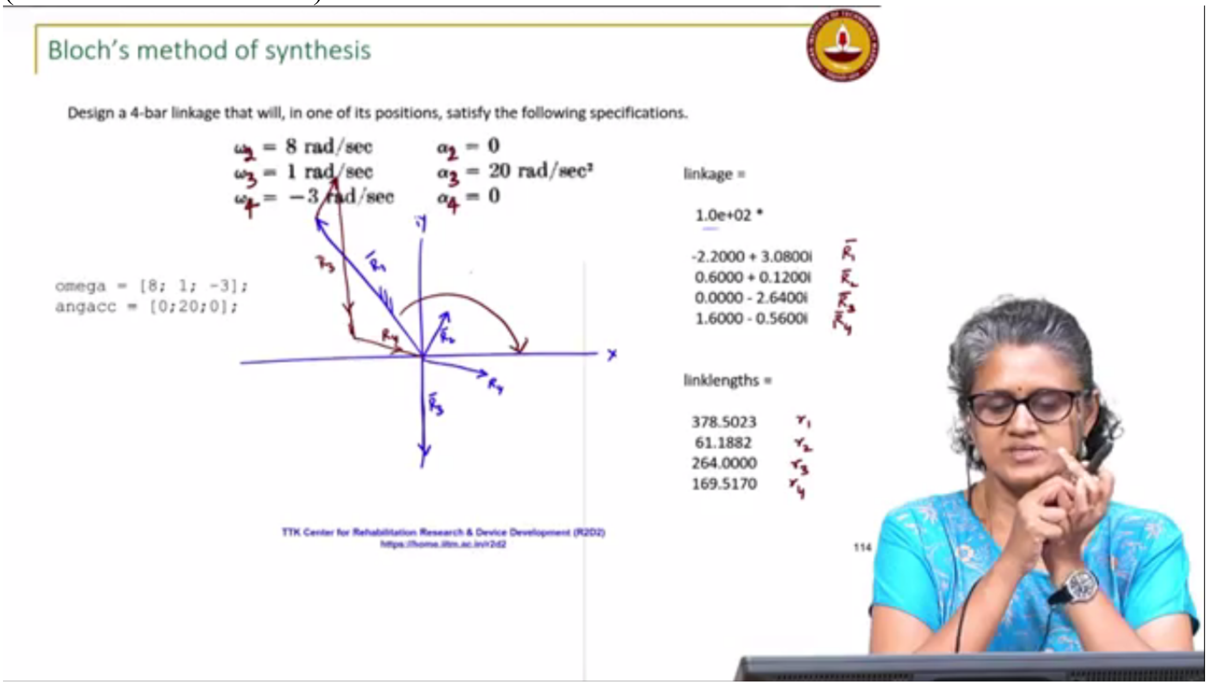
So this is my linkage. The book uses a different notation. Sorry. α_2, α_3, α_4. The book uses 1 for the input link and uses 4 for the fixed link. So this becomes your linkage.

(Refer Slide Time 18:49)



Now if I want to have this with R_1 along the real axis, then, essentially, I have to rotate this whole linkage by clockwise by this angle. All the vectors have to be rotated by the same angle. Otherwise, you can pretty much use it in any orientation in the plane and it will still have the same behaviour. Okay.

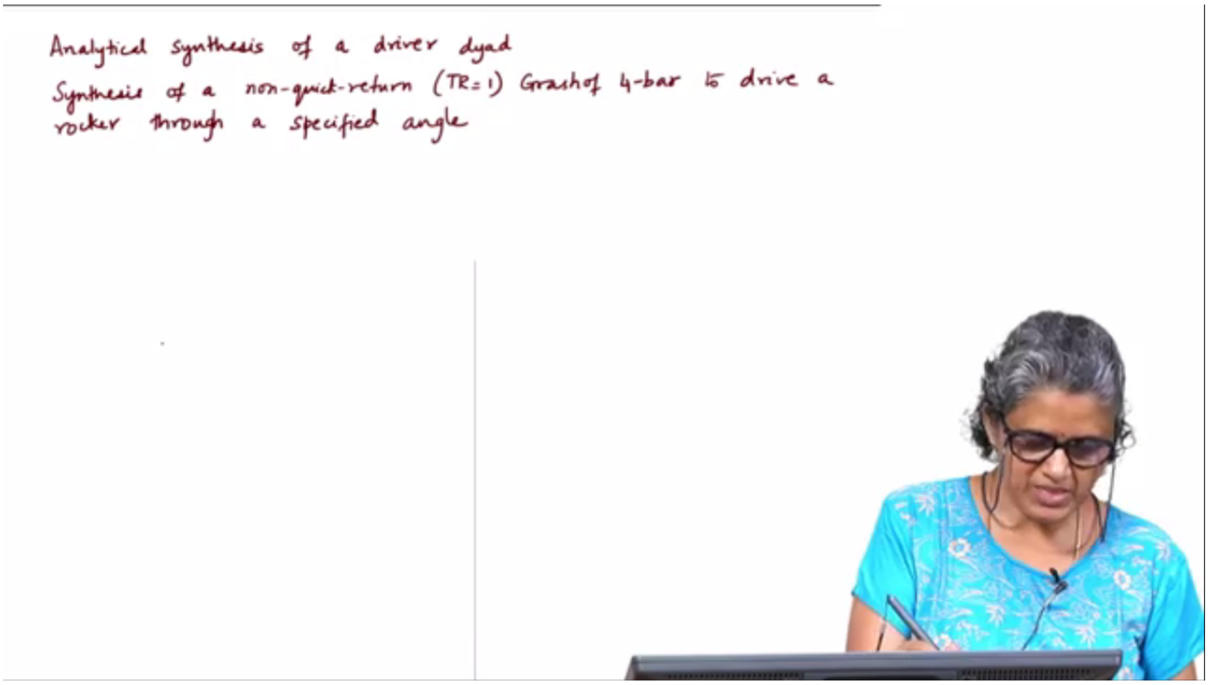
(Refer Slide Time 19:26)



So this is Bloch's method of synthesis. So both Freudenstein's equation and Bloch's method I will have you code in MATLAB so that given a set of conditions, you will be able to synthesize a linkage for that. Otherwise, we will move on to the analytical synthesis of a driver dyad.

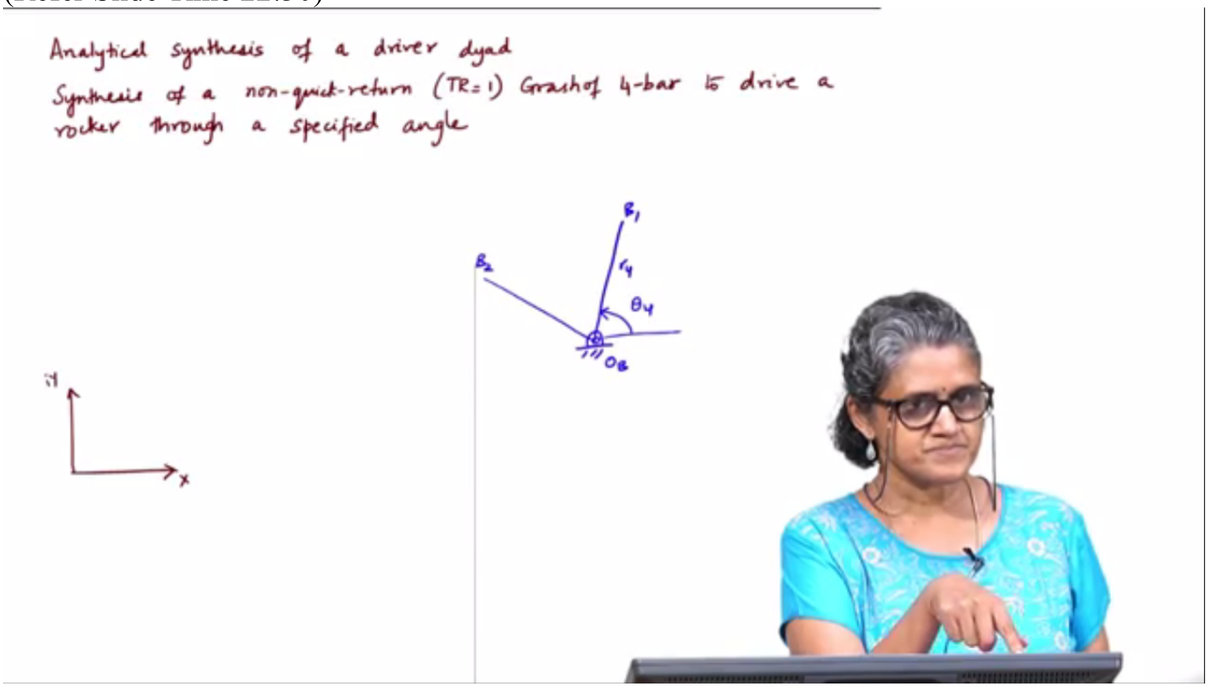
I will do the synthesis of a non-quick return meaning time ratio equal to 1, Grashof 4-bar to drive a rocker through a specified angle. So, practically, usually, your rocker movement should not exceed about 120 degrees. Okay. If you want more than that, you typically go to like a 6-bar. It's difficult to because your transmission angles become very low. So your force transmission is poor. So, practically, you want to keep the rocker excursion, the rocker range to be less than 120 degrees or so. Okay.

(Refer Slide Time 21:21)



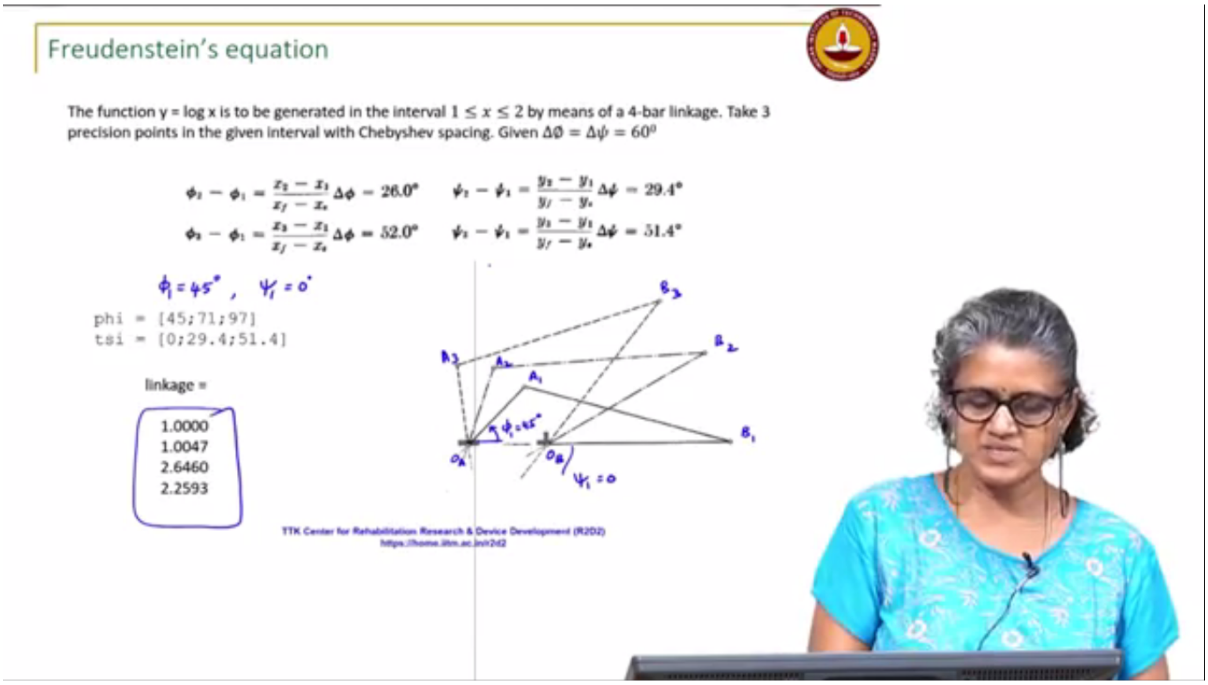
So let's start off with what we are given. I have (inaudible) B2 and say I have some initial angle not, although it looks 90, I don't want it to look 90. So let's make this B_1. This is some initial angle θ_4. Okay. So I'm given R_4. So I know B_1 and B_2. This B_1, this link may actually be the input to another 4-bar like the double-rocker that you did for the function generation in the last example.

(Refer Slide Time 22:30)



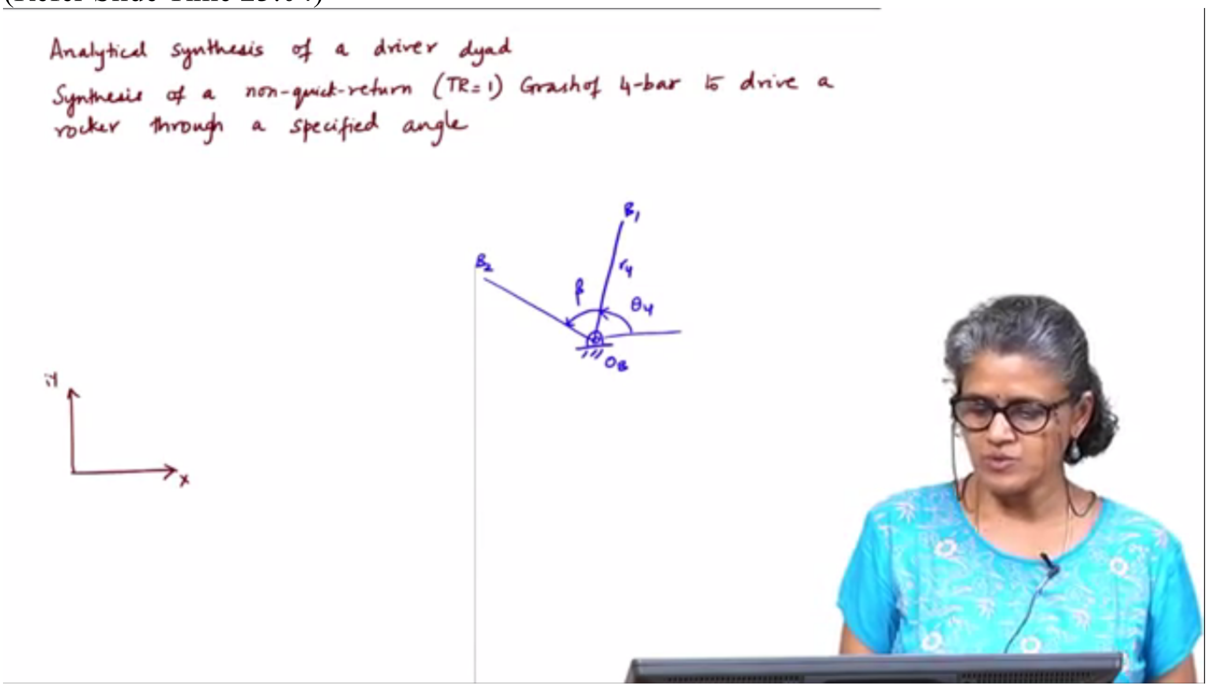
Okay. The one with the Freudenstein example, this one this is a double-rocker.

(Refer Slide Time 22:40)



It's a Grashof double-rocker, but it's not a crank-rocker. So you may want to attach a driver dyad to it in which case the output of this driver dyad is basically the input to that 4-bar as we've seen several examples before and say this angle is β. Okay.

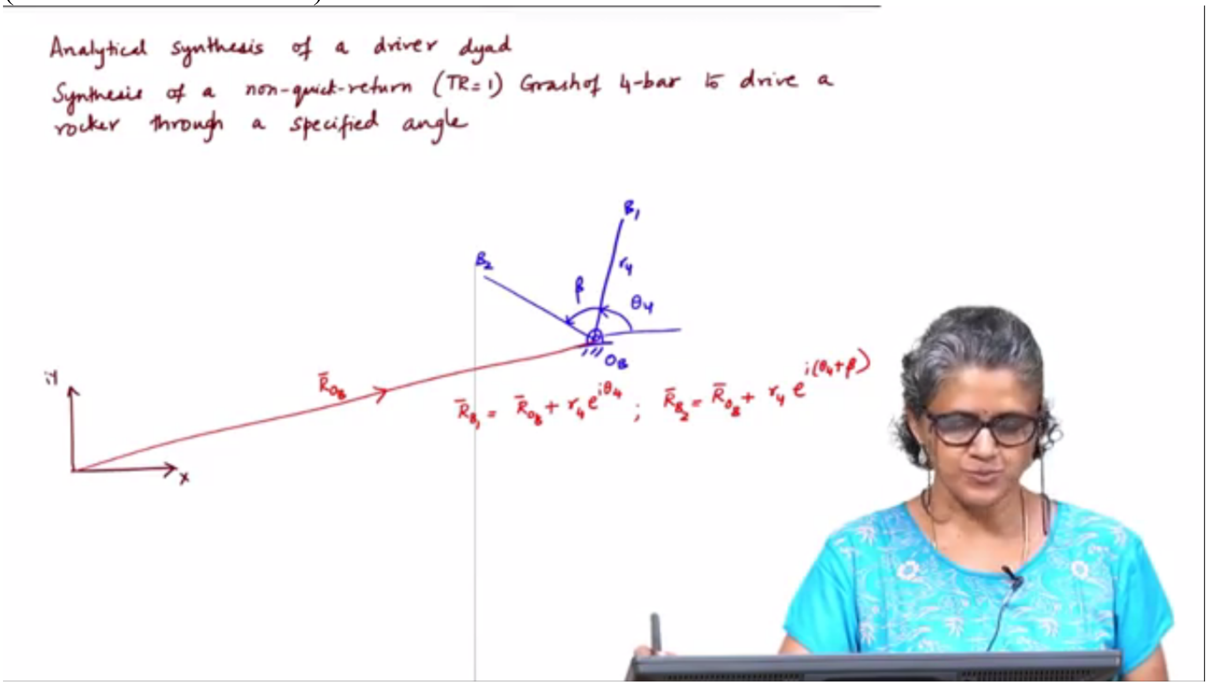
(Refer Slide Time 23:04)



So now my task is to find the other three links R_1, R_2 and R_3 such that this linkage gives a time ratio of 1 and I want to synthesize it analytically. Geometrically, we know, and we'll kind of use that in the synthesis in developing the equations, but if you had to do this geometrically, you would join B_1, B_2, extend it, pick some point on that line, then take half of B_1, B_2 to be your crank, and then you have your synthesis done and then you just check for Grashof. Okay.

So here now we will do. So here this is the vector R_{OB}. So I can find, this is given. I know the location of OB in the plane. So I can find R_{B1} is $R_{OB} + r_4e^{i\theta4}$. $R_{B2} = R_{OB} + r_4e^{i(\theta4+\beta)}$. Okay. So I can find the location of B_1. I find the location of B_2.

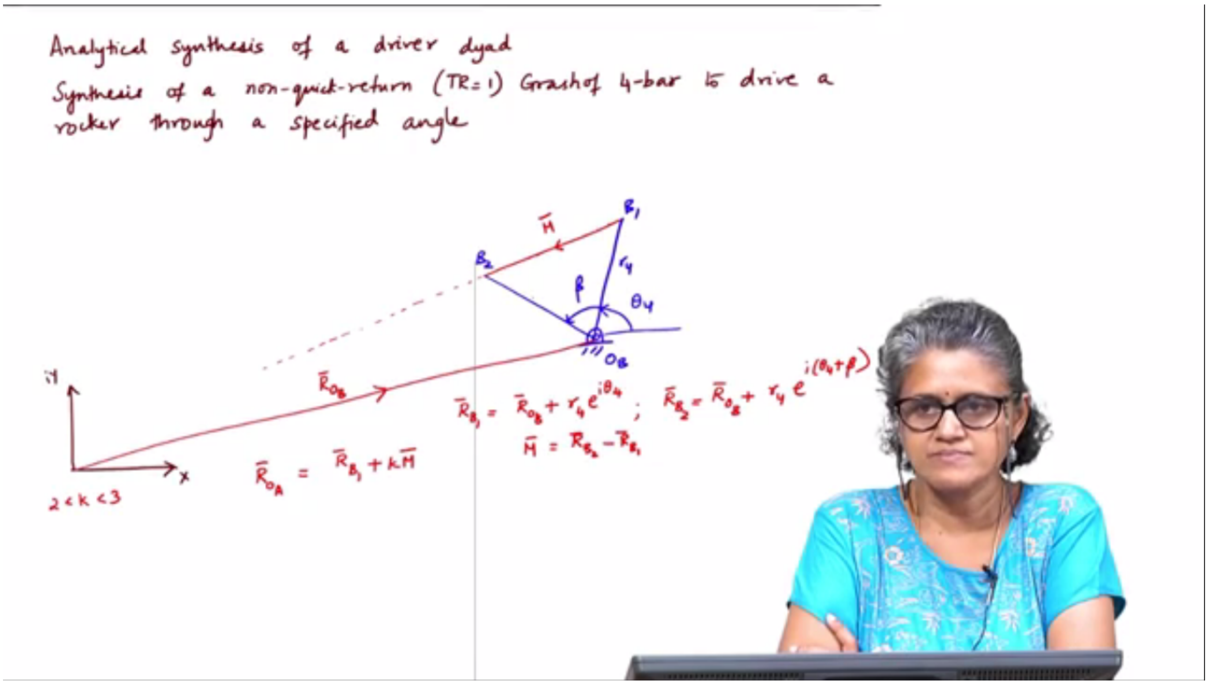
(Refer Slide Time 24:56)



I need to locate my, for a non-quick return linkage, I need to locate my O_2 somewhere, sorry, O_A somewhere on that line. So if I designate this vector, see I know B_2; I know B_1. R_{B2}, R_{B1}. So I know this vector. Let me call it M. Okay. M is nothing but this is R_{B2}. This is R_{B1}. So R_{B1} + M is R_{B2}. So R_{B2} - R_{B1} is M. Okay. So M is R_{B2} - R_{B1}. Just vector addition, right?

So now I can say my R_{OA}, O_A will be located somewhere here on this line, which would be at a location which is, so I can write this as R_{OA} will be equal to R_{B1} or R_{B2} plus some constant times M somewhere along that line and I can choose that constant typically around 2 to 3 times. Okay.

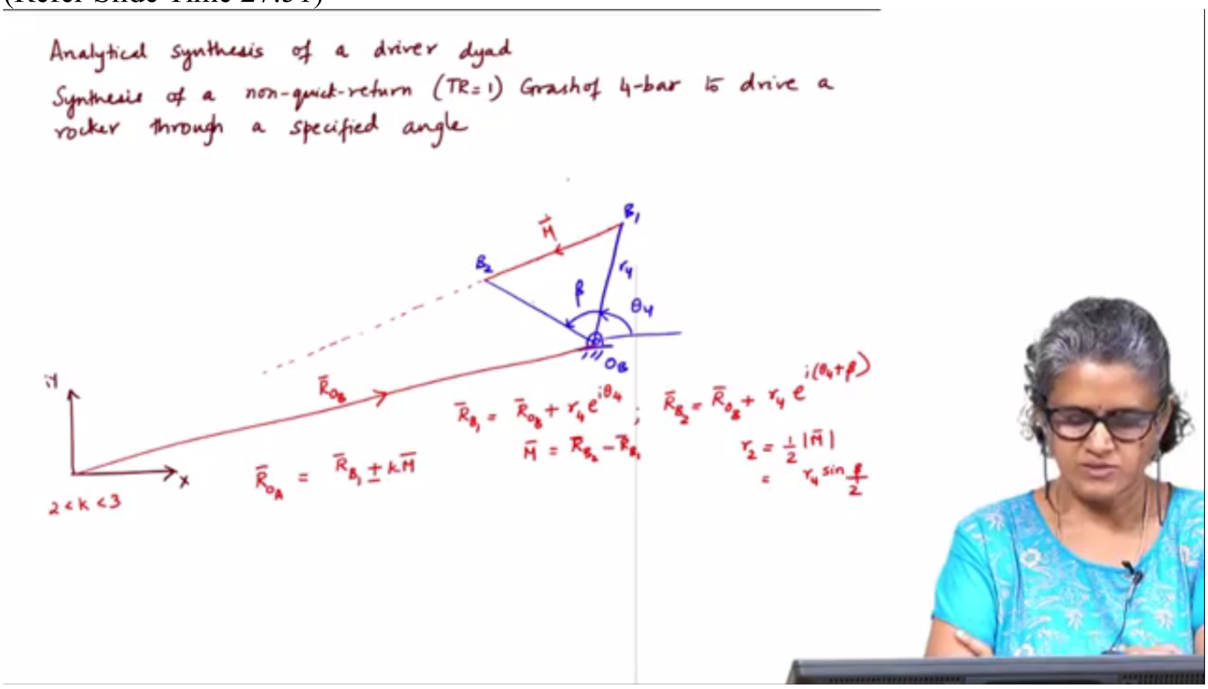
(Refer Slide Time 26:31)



So K, typical values of K if you want to get a crank-rocker somewhere between 2 and 3 is a reasonable choice. Again, it will depend on the problem, but this is just a guideline. It could also be on the other side. So it could be plus or minus K. Okay. Good. I could very well locate.

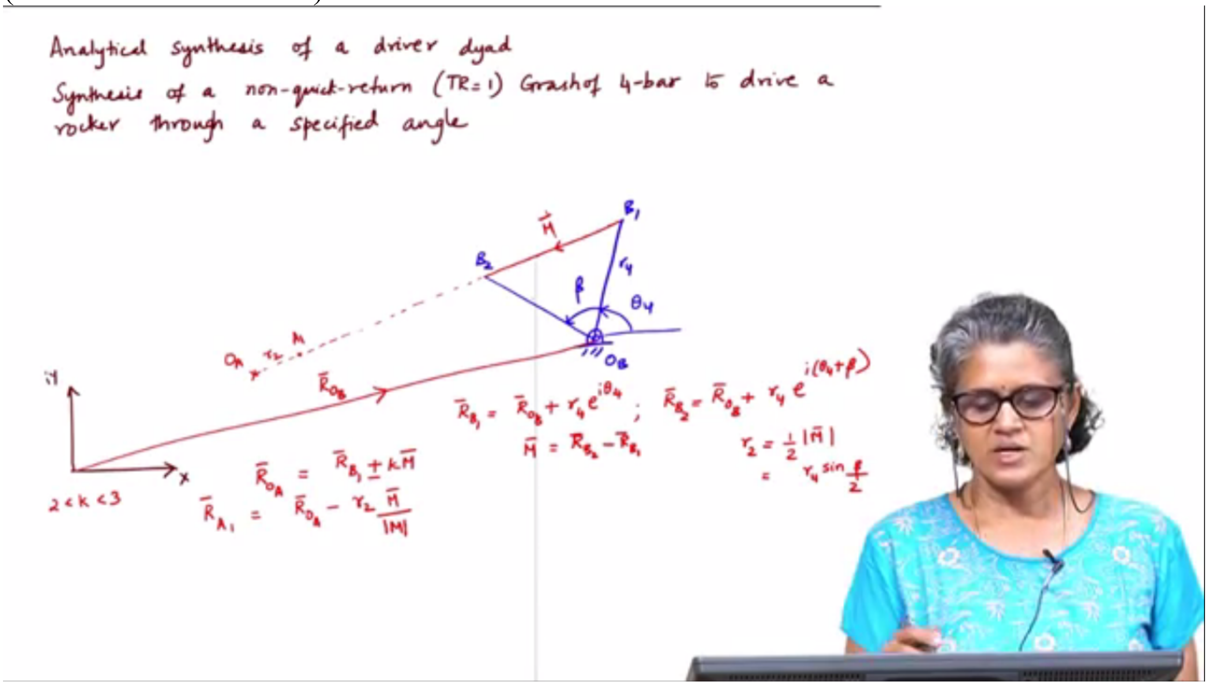
Now I know that R_2, what would be the magnitude of that? It would be half the magnitude of same thing, same, you know, it's exactly as we did the geometrical synthesis. So R_2 is half of this thing. So this is half of in terms of β, I can write it as $(½)r_4 \sin \beta/2$. Perpendicular bisector if I draw $\beta/2$, so this is r_4. No, wait. Yeah. Is it just $r_4 \sin \beta/2$? This is $\beta/2$. Yeah. It's just $r_4 \sin \beta/2$. Not half of that. Half of this whole thing, $r_4 \sin \beta/2$.

(Refer Slide Time 27:51)



So now do I have everything I need? I found O_A. Okay. I know my crank length. The only thing left is R_A. Okay. So if my O_A is somewhere here, this is r_2, okay, and that would be A_1. So R_{A1} will be, again, if I find the unit vector along that line, it will just be, yeah, R of this minus half along that line, minus r_2 along, so if I have $M/|M|$. That would be the unit vector along.

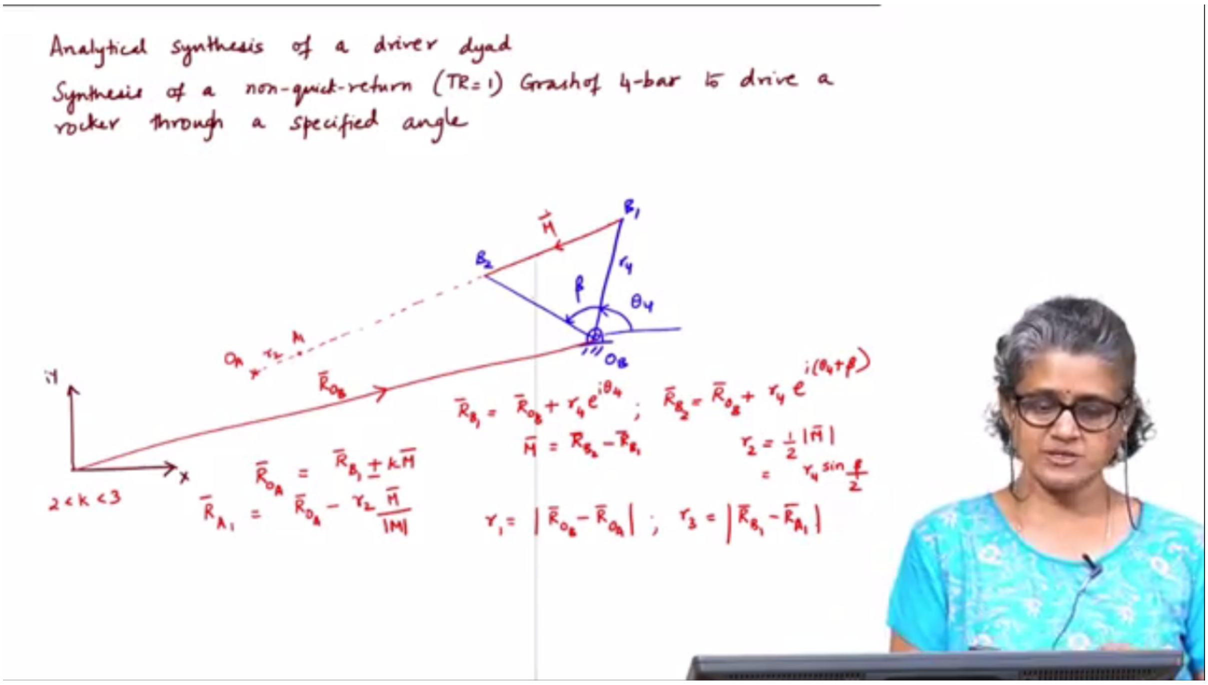
(Refer Slide Time 29:02)



So now I have all the parameters I need for the design. Again, you would program this and you are also going to develop the equations for time ratio not equal to 1 following a similar procedure because you already know how to do it graphically. It's a matter of translating that into equations for doing this. Okay.

So this is the analytical synthesis for a driver dyad. So it could be for a crank-rocker, stand-alone crank-rocker or for a driver dyad for another 4-bar. So output of this becomes the input for the other 4-bar for continuous input and your r_1 will essentially be $|R_{OB}-R_{OA}|$ and r_3 will be $|R_{B1}-R_{A1}|$.

(Refer Slide Time 30:31)



Then you check for, check Grashof. Very important. If you are designing a crank-rocker and you don't check for Grashof, so this method is given in Norton’s book, this analytical synthesis for the driver dyad. Okay.

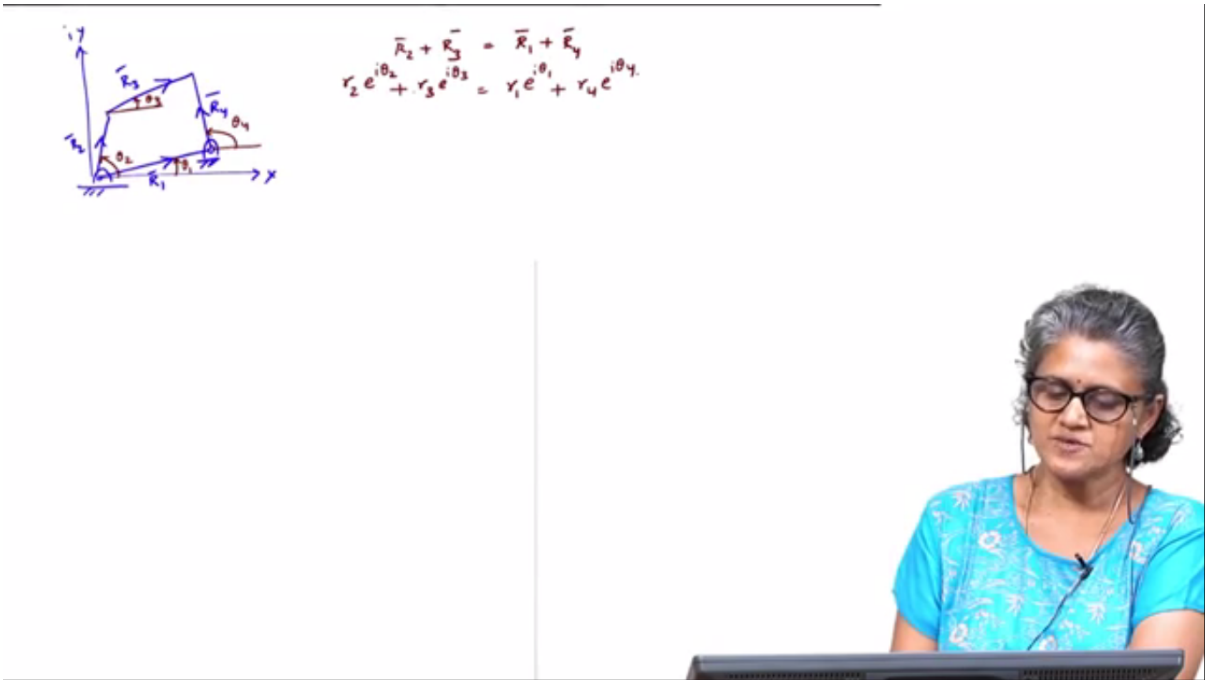
Most of the other material I have been taking from the Denavit and Hartenberg Hartenberg Denavit book, and most of the figures are also that I start putting in every slide because it is freely available and that's, that’s sort of like the textbook that I'm following for this course. For most of this course, we will be using that book. Okay. All right.

So, again, with these methods that we've seen so far, we are only looking at it being at certain positions; assembling the linkage at certain positions. Okay. So it is quite important that you are able to simulate the linkage, okay, over its entire range in order to be able to determine whether it is reaching those positions without encountering a branch change or toggle positions. Okay.

So because of that, let me just do the position analysis. For those of you who have not done the position analysis using the complex number techniques since I have some time today, I'm just going to do that very quickly. It's similar to what we did with Freudenstein's equation, but let me just go through it so that you can program that as well. When you do the design, you do the analysis of the, you do the position analysis as well. Okay. Just for the sake of completeness, I will cover that.

Okay. So we measure the angles with respect to the x-axis. So I have this would be θ_1. You would measure it at the root of the vector θ_4, θ_3 and θ_2. So my loop closure equation for the way I have taken these is $R_2+R_3 = R_1+R_4$. In complex form, $r_1e^{i\theta 1} + r_4e^{i\theta 4}$.

(Refer Slide Time 33:59)

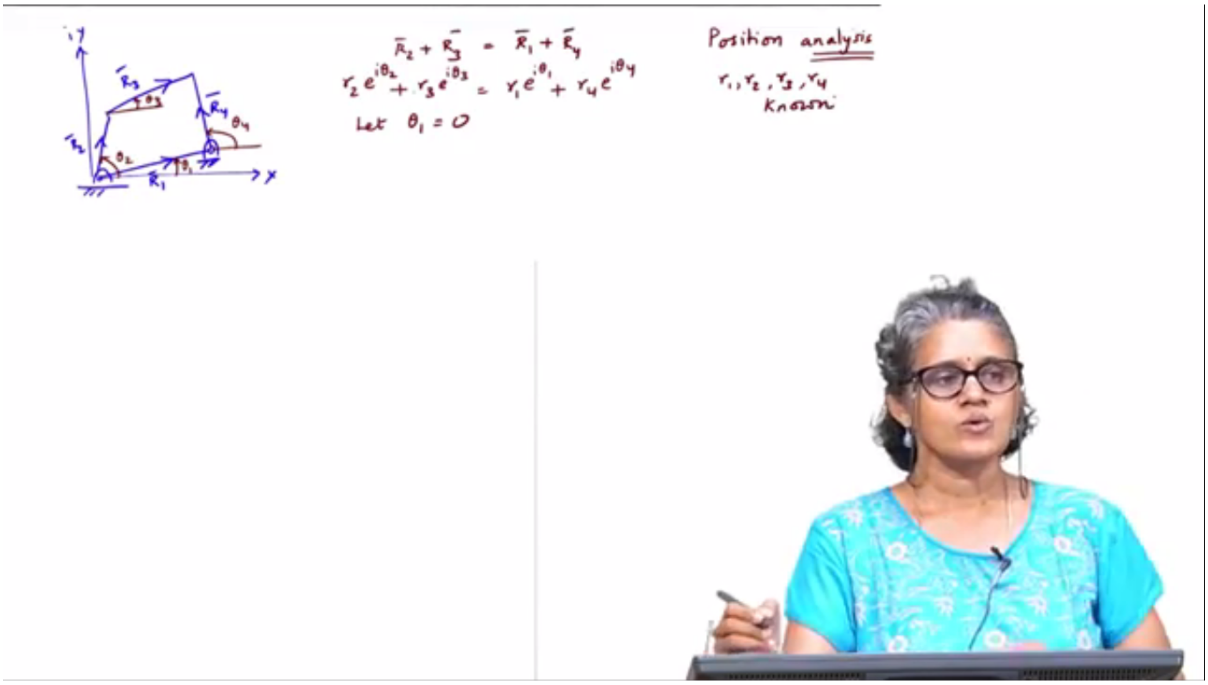


For the sake of simplicity, I can always choose my X, Y in a certain way. So let me choose it such that $\theta_1 = 0$. Okay.

Now the 4-bar linkage, what I am trying to do here is position analysis. What does that mean? I want to determine for a given input, how many inputs do I need to give a 4-bar in order to get a predetermined output? So, okay, when you are doing analysis, okay, this is the difference between what we have done a lot of is synthesis. Synthesis is when you try to determine the link lengths.

Now what you are doing is you know the link lengths. You are given the linkage. You want to see whether it behaves the way you want it to behave. That is analysis. So, yes, in the case of analysis, link lengths are known.

(Refer Slide Time 35:21)

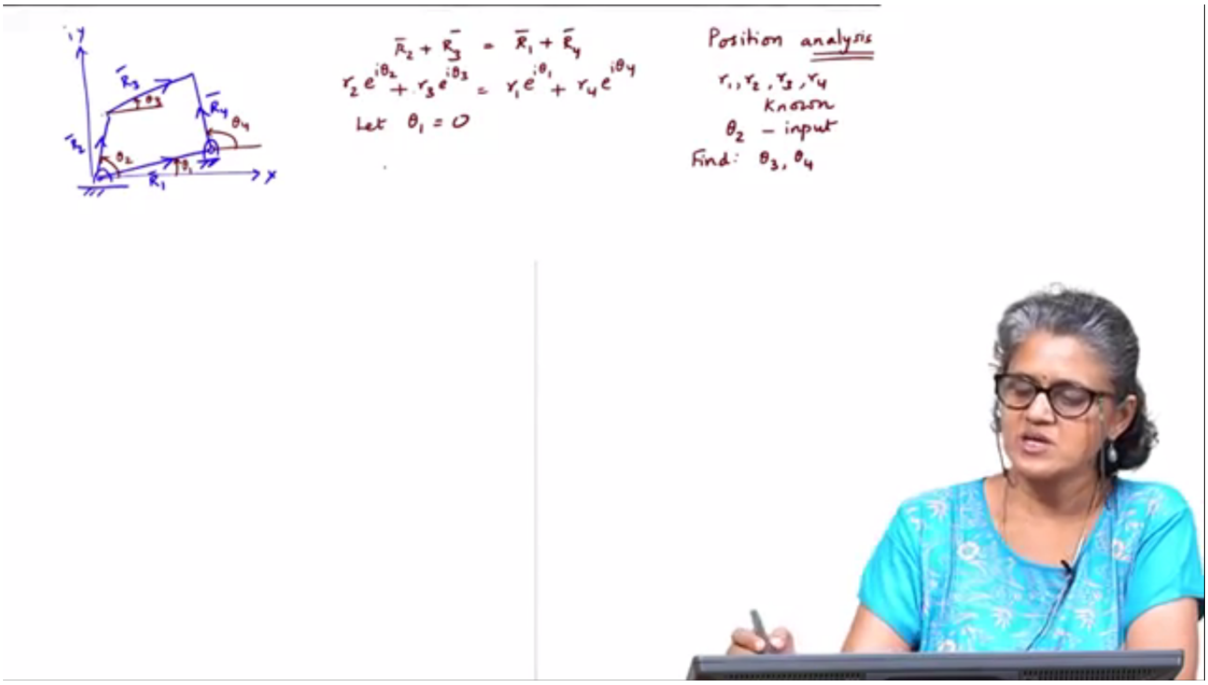


What else would you know if you want to? So the 4-bar linkage has mobility equal to 1, which means if I give it one input, then everything else becomes determined. Okay. Typically, θ_2 will be my input.

So what I have to now find is for every θ_2, I want to find out what are θ_3 and θ_4 because if I know that, the link lengths are all constant; they don't change for the 4-bar, so I can basically construct the 4-bar in every position.

Now it's very easy to do this graphically. You can construct the 4-bar very easily when you do it graphically because you just need the link lengths. Then from this point, you set it at this angle θ_2 an arc with length R_3, another arc with length R_4 and you are done if you know the configuration and in fact in MATLAB, you can construct it that way. You don't have to actually go through all this. You don't have to program these equations. MATLAB I think has this circ circ command. Anybody aware of MATLAB has this circ circ command I think? Intersection of two circles. So you can use that intersection of two arcs to actually find that easily. So you can construct it in MATLAB by doing that as well. Okay. But here we will just go through the analytical set of equations.

(Refer Slide Time 37:18)

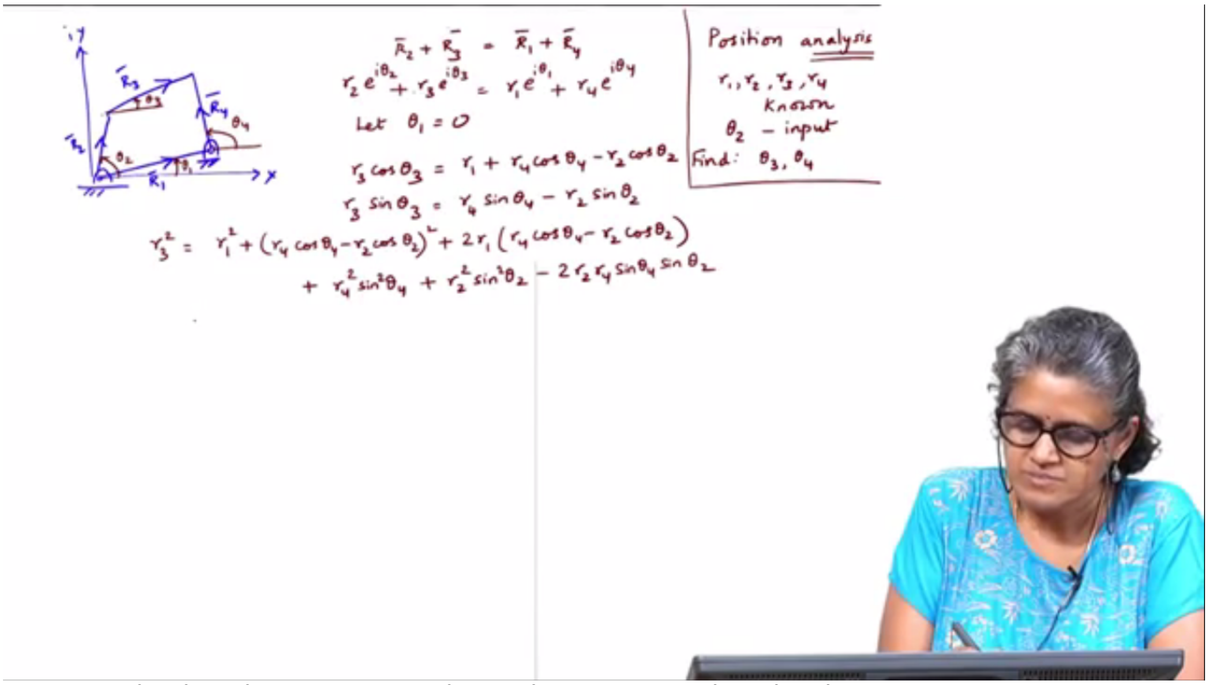


So we want to find θ_3 and θ_4 if θ_2 is the input given. This, these are, this is a vector equation. So I can only solve for two unknowns, which will be θ_3 and θ_4. But again, θ_3 and θ_4 are in transcendental form in this equation. So it's a little bit more tedious to solve it.

So the first thing I'll try to do is to eliminate. So I write this into real and imaginary parts and say I want to eliminate θ_3, like we did when we derived it for Freudenstein's equation. There our purpose was different. We knew the θ_2, θ_4 combinations for which we were finding the link lengths. Now we know the link lengths. We want to find if it behaves the way we want it to.

So I'll just take $r_3 \cos \theta_3$ separating into real and imaginary parts. I have taken θ_1 as 0. So I have $r_1 + r_4 \cos \theta_4 - r_2 \cos \theta_2$. Then I have $r_3 \sin \theta_3 = r_4 \sin \theta_4 - r_2 \sin \theta_2$. Okay. Now if I square and add, my only unknown will be θ_4 in these two equations, right? I square and add these equations and I get $r_3^2 = r_1^2 + (r_4 \cos \theta_4 - r_2 \cos \theta_2)^2 + 2 r_1 (r_4 \cos \theta_4 - r_2 \cos \theta_2) + r_4^2 \sin^2 \theta_4 + r_2 \sin^2 \theta_2 - 2 r_2 r_4 \sin \theta_4 \sin \theta_2$. Okay.

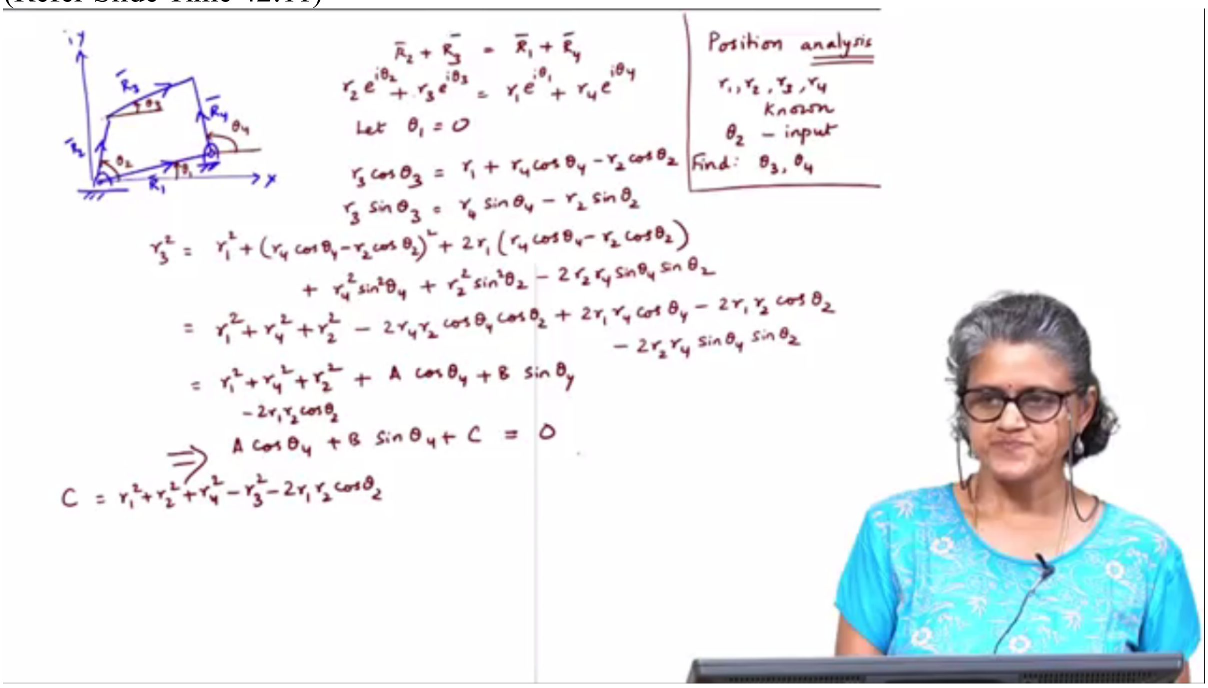
(Refer Slide Time 39:49)



So I get r_1^2, $r_4^2 \cos^2$ I will get $+\sin^2$, so r_4^2. Similarly, $r_2^2 \cos^2 + r_2^2$ - $2 r_4 r_2 \cos \theta_4 \cos \theta_2$, $\cos \theta_2$. These two are gone. $-2r_2r_4 \sin \theta_4 \sin \theta_2$. Here I won't try to simplify this because θ_2 is known. Okay. So I want everything in terms of θ_4. So I'll group them as $r_1^2+r_4^2+r_2^2$, okay, plus some K_1, $A_1 \cos \theta_4 + B_1 \sin \theta_4$. This would also go with this one, $-2 r_1+r_2 \cos \theta_2$.

This would be, so this is a constant. So this, so I can say this is equal to $A_1 \cos \theta_4 + B_1 \sin \theta_4 + C$. I don't even have to call it, $A \cos \theta_4 + B \sin \theta_4$ plus I should take this also to the other side. Okay. Let me just put this as equal to 0 where my C would be $r_1^2+r_2^2+r_4^2-r_3^2-2r_1r_2 \cos \theta_2$. All this is known. θ_2 is my input. All the other angles. Sorry, I'm late.

(Refer Slide Time 42:11)



I want you to, what's the, when you want to solve an equation like this, what's the most common substitution you do? Yeah. Use the tan θ/2 formula. Express both cos θ and sin θ in terms of tan θ/2 and solve. You will get a quadratic equation. Okay.

[Music]

(Refer Slide Time 43:04)