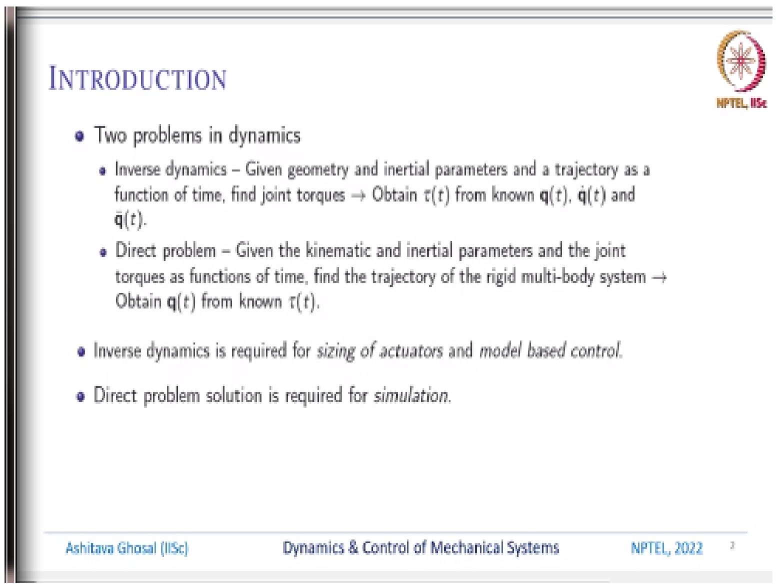
Dynamics and Control of Mechanical Systems
Prof. Ashitava Ghosal
Department of Mechanical Engineering
Indian Institute of Technology-Bengaluru

Lecture-18
Inverse Dynamics and Simulations of Equations of Motion

In this lecture we look at the 2 main problems in dynamics of multi-body systems. The first is the inverse dynamics and the second is the simulation of equations of motion.

(Refer Slide Time: 00:36)



So, there are 2 main problems in dynamics of multi-body systems, the first is called as the inverse dynamics. In the inverse dynamics problem we are given the geometry and inertial parameters of all the rigid bodies in a multi-body system and we are also given a trajectory as a function of time. So, what I mean by trajectory is? We are given the generalized coordinates $q(t)$, $\dot{q}(t)$, $\ddot{q}(t)$.

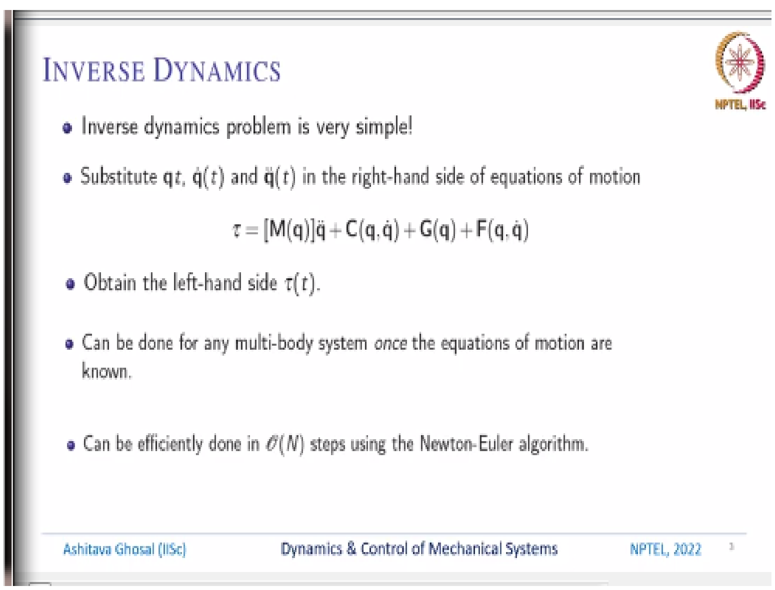
And the goal in the inverse dynamics problem is to obtain the torque which must be applied at the joints, so as to get these $q(t)$, $\dot{q}(t)$, $\ddot{q}(t)$. In the direct problem on the other hand we are given the kinematic and inertial parameters and the joint torques as a function of time. So, we are basically given $\tau(t)$ as a function of time and the goal is to find the trajectory of the rigid multi-body system.

So, the goal is to find $q(t),\ \dot{q}(t), \ddot{q}(t)$, so as you can see which is in some sense opposite of the inverse problem. The inverse problem or the inverse dynamics problem is required for sizing of actuators and model based control as I have mentioned earlier. So, once I know that the multi-body system should be able to have a desired trajectory given by $q(t),\ \dot{q}(t), \ddot{q}(t)$ to we need to choose the motors and the actuators in the multi-body system.

So, we need to solve this inverse dynamics problem to get an estimate of the torque required to achieve this $q(t),\ \dot{q}(t), \ddot{q}(t)$. So, hence the solution of the inverse dynamics can be used to estimate the size of the actuators, it is also used in model based control. The direct problem on the other hand is required for simulation, so if I make a model of my multi-body system.

And if I tell that the torque which you are applying at the actuator is given by $\tau(t)$, I need to show you how the rigid bodies of this multi-body system are going to move. So, hence we need to solve the equations of motion and that is called simulation.

(Refer Slide Time: 03:42)



The inverse dynamics problem in multi-body system is relatively simple, it is actually quite simple. All we require to do to solve the inverse dynamics problem is to take $q(t),\ \dot{q}(t), \ddot{q}(t)$ in the right hand side of the equations of motion. So, we have an equation of motion which is given in this form

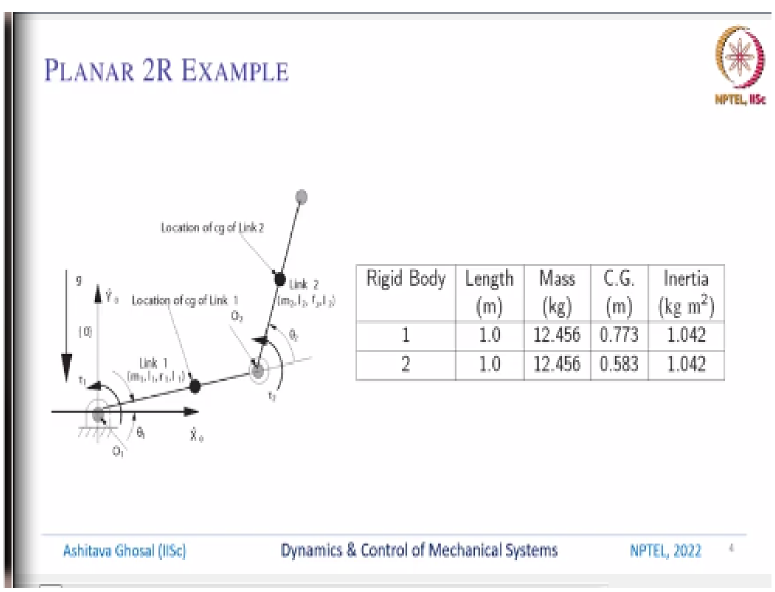
$$\tau = [M(q)]\ \ddot{q} + C\left(q, \dot{q}\right) + G(q) + F(q, \dot{q}),$$ there might be also be a friction term.

All we need to do is substitute $q(t),\ \dot{q}(t), \ddot{q}(t)$ on the right hand side and evaluate torque as a function of time. So, the left hand side is nothing but after substituting this $q(t),\ \dot{q}(t), \ddot{q}(t)$we get torque as a function of time. So, as you can see it is relatively simple problem, it can be done for any multi-body system once the equations of motion are known.

So, once the right hand sides are the full equations of motion are known the inverse dynamics problem is nothing but simple substitution. That is why I said that the inverse dynamics problems for multi-body systems are very simple. It can be done very, very efficiently, so it can be done for example in O N steps using the Newton-Euler algorithm which we have looked earlier. So, we can start with the base we can substitute $q(t),\ \dot{q}(t), \ddot{q}(t)$.

And find all the terms on the right hand side as we go up to the final node. And then we can come down and find the torque as a function of time, so it is the linear complexity algorithm.

(Refer Slide Time: 05:47)



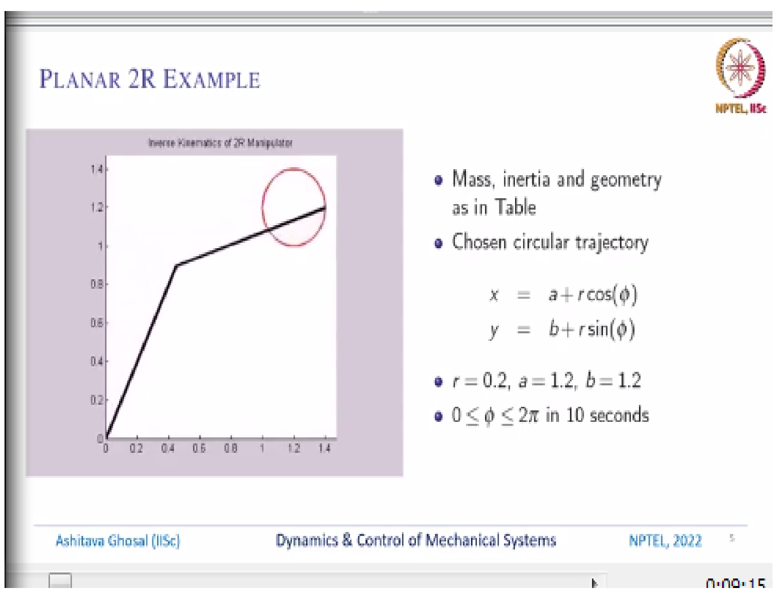
Rigid Body	Length (m)	Mass (kg)	C.G. (m)	Inertia (kg m^2)
1	1.0	12.456	0.773	1.042
2	1.0	12.456	0.583	1.042

So, let us take an example and again we will use our simple example which is the planar 2 degree of freedom of 2 rigid bodies connected by 2 rotary joints. So, there is a link 1 and a link 2 and again the link 1 is specified by mass, length, location of CG and inertia, likewise link 2 is specified by mass m_2, l_2, r_2 and I_2 , again I_2 is the z component of the inertia which is of relevance here because this is a planar motion.

So, to solve the inverse problem we need the numbers for the geometry and the mass properties. So, in this example I have assumed that the length of this link is 1 meter, the length of the second link is also 1 meter, the mass is some 12.456 kg, this is from some actual example which we considered long time back. The location of the CG of the first link is at 0.773 meters from this origin.

And for the second link it is at 0.583 meters again from this origin and the inertia is 1.042 kg meter square, each link has the same geometry it is made of the same material and hence the inertia about the z axis about the CG are same.

(Refer Slide Time: 07:24)

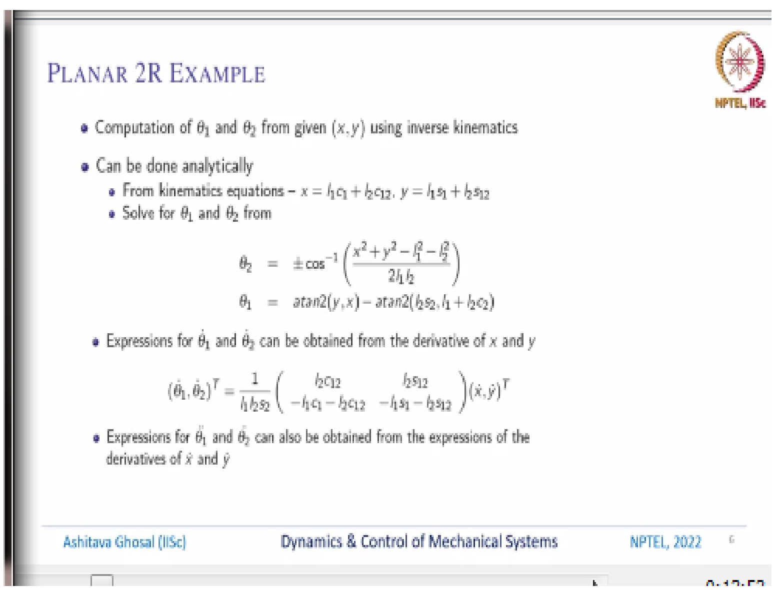


So, for the mass inertia and geometry parameters which were shown in the table, we can now solve the inverse dynamics problem. So, the next thing that we need to solve the inverse dynamics problem we need to choose a trajectory. And in this example we have chosen a circular trajectory, meaning that the tip of this 2R manipulator or rigid body chain with 2 degrees of freedom will trace a circle which is given by $x = a + r \cos cos\, \phi$, $y = b + r \sin sin\, \phi$.

Where a and b are the center of the circle and r is the radius of the circle, so it will be some circle which is located somewhere in this region. And to find out actually what the circle is and where the circle is we need to choose some $r,\ a\ and\ b$. And again in this example I have chosen r as 0.2 meters, a as 1.2 and b as 1.2. So, the center of this circle is somewhere here 1.2 here and 1.2 here and it is a circle of radius 0.2.

To bring time into the picture we need to now specify how does this tip of this robot traces the circle in how much time? So, that can be specified by saying that this angle phi in the parametric equation of the circles goes from 0 to 2 π in 10 seconds. So, here is a picture of the circle which it is actually going to trace, so this is very straightforward computation. You substitute phi going from 0 to 2 π, we substitute $r,\ a\ and\ b$ in these equations and we trace this circle.

(Refer Slide Time: 09:31)



So, in the previous slide I showed you that the tip of this robot will trace the circle of radius 0.2 and located at 1.2, 1.2. Now to solve the inverse dynamics problem we need to find θ_1 and θ_2 as a function of time. We also need to find out $\dot{\theta}_1$ and $\dot{\theta}_2$ and also the second derivatives of θ_1 and

θ_2 as a function of time. Because remember, the equations of motion are tau equals sum m into $\ddot{\theta}$ plus some coriolis term plus some gravity term.

We have specified the trajectory using x and y but I mean the trajectory is in terms of θ_1 as a function of time and θ_2 as a function of time. How do I obtain this? We can obtain θ_1 and θ_2 as a function of time by what is called as inverse kinematics. So, basically this can be done analytically and all we need to do is we need to take the kinematics equation which is x = $l_1 c_1$ + $l_2 c_1 2$ and y = $l_1 s_1 + l_2 s_{12}$.

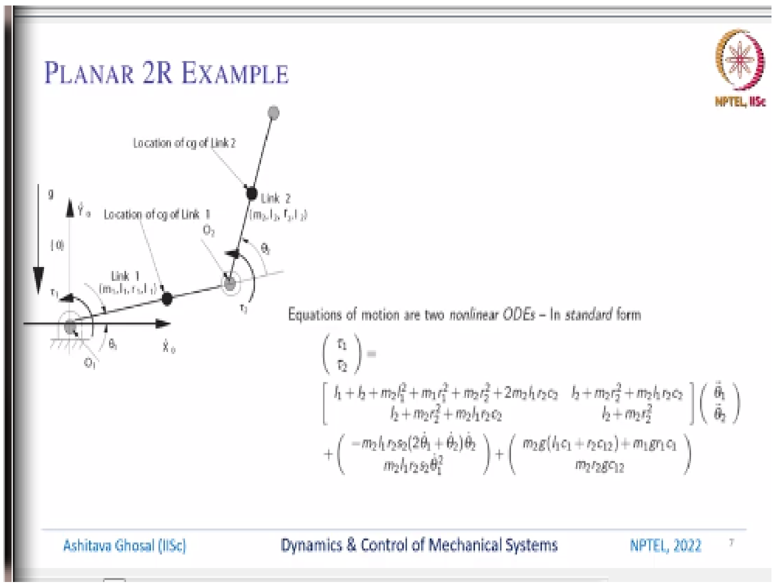
So, we have 2 equations in 2 unknowns we are given x and y, we have to solve for θ_1 and θ_2. So, remember c_1 means cos θ_1, $c_1 2$ means cos $\theta_1 + \theta_2$, s_1 means sine θ_1, s_{12} is sine $\theta_1 + \theta_2$. So, we can solve these 2 equations in 2 unknowns by doing some tricks. So, basically if you do x square + y square - l_1 square - l_2 square then you will get θ_2, so θ_2 is cos inverse of this quantity divided by 2 l_1 l_2.

θ_1 can also be found once you find θ_2 and that is given by 8 and 2 of y, x - a tan 2 of l_2 s 2, l_1 + l_2 c_2. So, what is a tan 2? This is a function which is like tan inverse of y by x, however a tan 2 is giving an angle in the right quadrant. We had seen this a tan 2 when we were doing Euler angles and that is exactly the same thing that we have using here. We want to obtain the correct angle in the correct quadrant.

So, tan inverse of -1, -1 will give you 45 degrees but a tan 2 of -1, -1 will give you in the third quadrant. The expression for $\dot{\theta}_1$ and $\dot{\theta}_2$ can also be obtained analytically. So, basically we take the derivatives of these 2 equations we have x dot, then we will see this will be - $l_1 s_1 \dot{\theta}_1$ - $l_2 s_{12}$ $\dot{\theta}_1 + \dot{\theta}_2$ and so on. And then we can solve $\dot{\theta}_1$ and $\dot{\theta}_2$ in terms of x dot and y dot and that is given by this expression, you can verify it yourself.

That $\dot{\theta}_1$, $\dot{\theta}_2$ is 1 by $l_1\, l_2$ s 2 into sum matrix which contains $l_2\, c_1 2\, l_2\, s_{12}$ - $l_1\, c_1$ - $l_2\, c_1 2$, so this is the first column. And the second column is $l_2\, s_{12}$ - $l_1\, s_1 - l_1\, s_{12}$ **okay** and then we multiplied by x dot, y dot. So, we can obtain this expression for $\dot{\theta}_1$ and $\dot{\theta}_2$, so this is actually l_2. The expression for $\ddot{\theta}_1$ and $\ddot{\theta}_2$ can also be obtained from the expressions of the derivative of x dot and y dot.

(Refer Slide Time: 13:54)

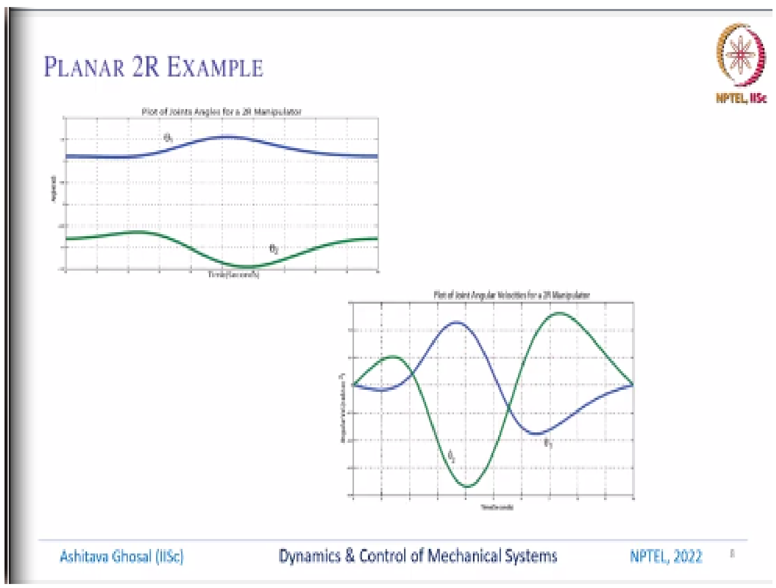


So, now that we have expressions for θ_1 and θ_2 as a function of time, likewise $\dot{\theta}_1$ and $\dot{\theta}_2$ as a function of time and $\ddot{\theta}_1$, $\ddot{\theta}_2$ as a function of time. We can go back and look at the equations of motion which are given here, so we have τ_1, τ_2 some matrix into $\ddot{\theta}_1$, $\ddot{\theta}_2$ and then this coriolis and centripetal term which have $\dot{\theta}_1$ square and $\dot{\theta}_1\, \dot{\theta}_2$ and so on.

And then the gravity term which contains m 2 g $l_1\, c_1 + r_2 c_{12}$ and so on. So, functions of angles θ_1 and θ_2 as a function of time and gravity. So, we have now obtained θ_1 as a function of time, θ_2 as a function of time, all we need to do is substitute in the right hand side of these equations of motion. So, if you substitute θ_1 as a function of time and θ_2 as a function of time everything inside this matrix is known.

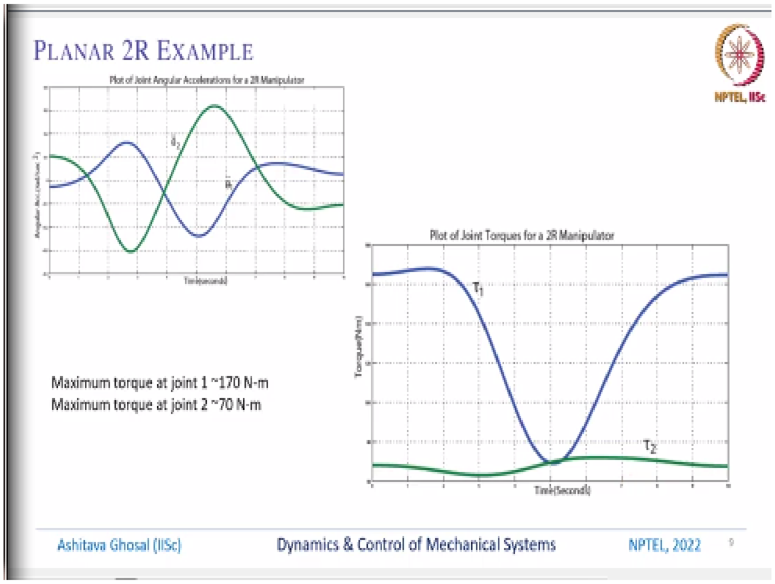
Then we multiply $\ddot{\theta}_1$ as a function of time and $\ddot{\theta}_2$ as a function of time with the relevant elements of the mass matrix and we can get evaluate this term. So, we can write one code in MATLAB where you give the inputs which are θ_1, $\dot{\theta}_1$, $\ddot{\theta}_1$, θ_2 $\ddot{\theta}_2$, $\dot{\theta}_2$ as a function of time and then you substitute here and we get τ_1 as a function of time and τ_2 as a function of time which is the inverse dynamics problem.

(Refer Slide Time: 15:48)



So, here are some plots for that circle which I showed you θ_1 varies in this form, θ_2 is varies in this form, $\dot{\theta}_1$ and θ_2 looks like this. So, the blue one is $\dot{\theta}_1$ and the green one is $\dot{\theta}_2$ and so on.

(Refer Slide Time: 16:12)

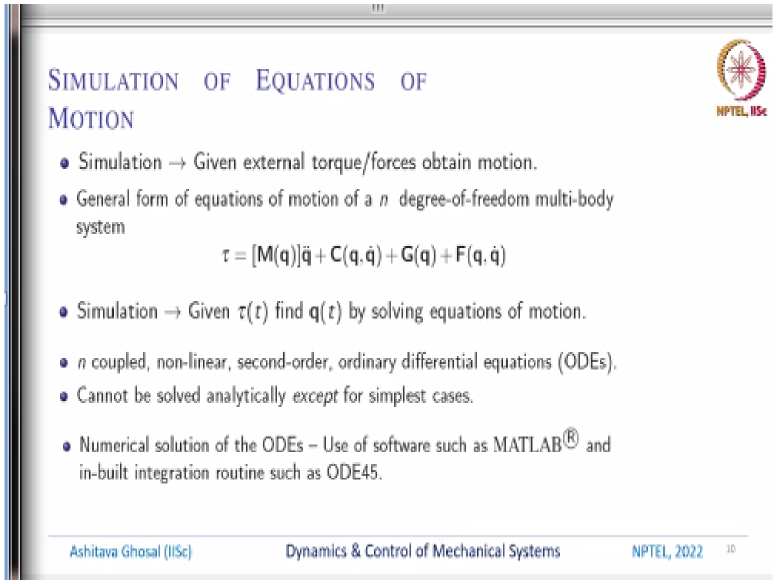


And then we can also obtain $\ddot{\theta}_2$ and $\ddot{\theta}_1$, so it is not very important to see what are these numbers but this is over a period of 10 seconds, the x axis is 10 seconds and the y axis **is** in this case it is $\ddot{\theta}_1$ and $\ddot{\theta}_2$. So, as you substitute back in the right-hand side of the equations of motion we can plot τ_1 and τ_2 as a function of time. So, you can see that the τ_1 is maximum, τ_1 is of the order of 170 Newton meter, the maximum τ_2 is of the maximum is of the order of 17 Newton meter.

These numbers are not important but basically what I am trying to show you is for a given θ_1, θ_2, $\dot{\theta}_1$, $\dot{\theta}_2$, $\ddot{\theta}_1$ and $\ddot{\theta}_2$ basically the trajectory of this multi-body chain. In this case 2R manipulator, I can find out the torque required to achieve that trajectory. You can have many different trajectories and for every single trajectory you can obtain τ_1 and τ_2.

And what is the estimate or which motors will you choose you find out which is the worst case? So, you find the maximum τ_1 and maximum τ_2 which you get for all the possible trajectories that you can think of or all the trajectories which this 2R robot is planned to be used for and then you can get an idea, okay I need so many Newton meter torque for the first motor and so many Newton meter torque for the second motor.

(Refer Slide Time: 17:59)



So, next we look at the simulation of the equations of motion. So, simulation is exactly the opposite problem, this is also the direct problem. In simulation we are given the external torques and forces and we obtain the motion of the multi-body system. So, in this case τ_1 and τ_2 as a function of time will be given to us for this 2R chain and then we have to find out what is θ_1 as a function of time θ_2 as a function of time and so on.

So, how do we do this? We go back again to the general equation of motion of n degree of freedom multi-body system. So, we have this equation of motion which is torque equals $\tau = [M(q)]\ddot{q} + C(q, \dot{q}) + G(q) + F(q, \dot{q})$, which we have added in adhoc manner. So, in simulation we are given the left hand side as a function of time, we are given tau as a function of time and we have to find out $q\ (t)$ by solving these differential equations.

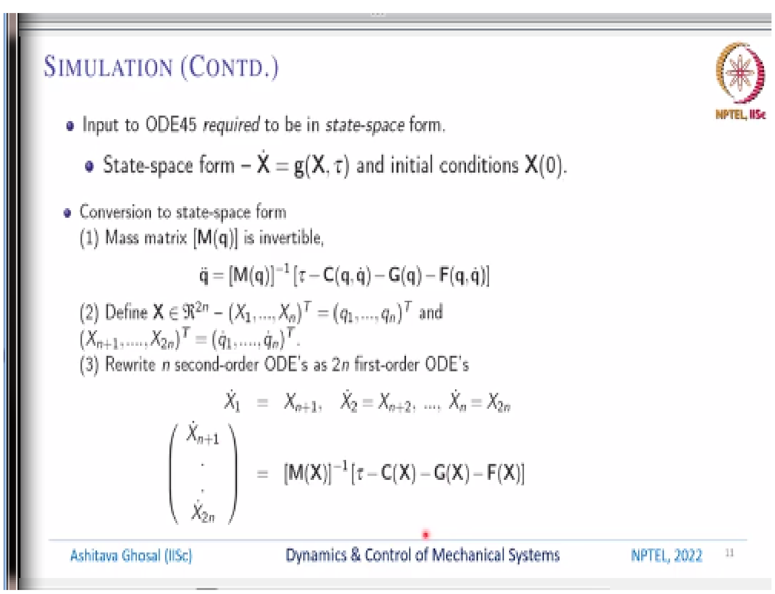
So, these are n coupled non-linear second order ordinary differential equations. Coupled, because remember the mass matrix contains even for the 2R case contains θ_2, these have $\dot{\theta}_1^2$ and $\dot{\theta}_2^2$ and all possible combinations. So, for the 2R manipulator example both the equations were coupled and these are also non-linear because you have sine θ_2, cos θ_2 and so on.

For the 2R case cos($\theta_1 + \theta_2$) and then something like sine θ_2 into $\dot{\theta}_1^2$, so all kinds of non-linearities is there in this equations of motion. So, for most of these cases we cannot solve it analytically, differential equations are very hard to solve if it is non-linear and coupled. So, only for maybe some very, very simple case we can solve these equations of motion, basically we can integrate those equations of motion given the left hand side as a function of time, we can find q as a function of time.

So, that is most of the time are in fact actually almost everywhere it is not possible to do analytically. So, we need to do numerically, so the numerical solutions of ODE are very well known it is a very, very again a very mature topic. There are many software tools which are available and we are going to use MATLAB. So, in this NPTEL course you have access to MATLAB and you can try getting some numerical solutions of the differential equations.

So, we will give at least 1 homework where you can solve numerically this differential equation to see what is the solution or what is the simulation of a multi-body system. So, MATLAB gives you in-built integration routines such as ODE45. So, if you see the MATLAB tutorial which is written by one of the TAs for this course. You can see the he has used ODE45 for some solutions for some solution of certain differential equations, so please get familiar with ODE45.

(Refer Slide Time: 21:43)



So, the input to ODE45 is required to be in the state space form. So, what exactly is the state space form? We want the differential equations as a first order equation, so basically we want it in the form $\dot{X} = g(X, \tau)$. So, this right hand side could be nonlinear functions of X and also torque. But we the equations of motion are second order because they contain $\ddot{\theta}$. So, what we want everything in the form of a first order equation, this is what ODE45 requires you to give.

So, basically ODE45 requires you to give what is the right hand side of this first order differential equation and it also requires you to give some initial condition. So, we need to say what is the solution at t = 0, all integration routines require an initial condition, even analytically if you want to solve or integrate then you need some initial conditions. So, how do we convert the second order ODEs into the state space form, so how do we do it?

We start with the equation of motion which is M $\ddot{q} = \left(\tau - [C]\dot{q} - G - F\right)$. So, we know that the mass matrix is invertible, so hence we can always write$\ddot{q} = M^{-1}\left(\tau - [C]\dot{q} - G - F\right)$, so F is the friction term, this is the gravity term, this is the centripetal coriolis term and this is the external torque. Next we define X which is it is an element twice the number of q's, so we have X_1 through X_n as q_1 through q_n.

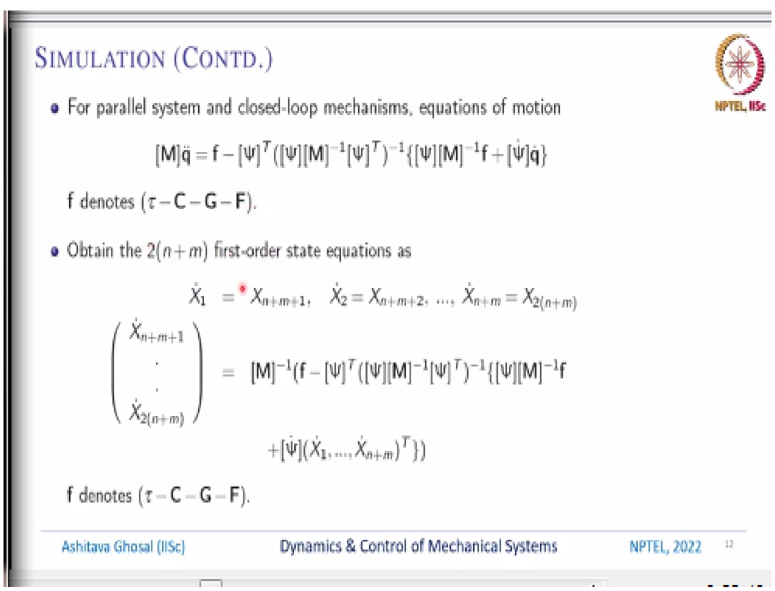
So, remember we had θ_1, θ_2, $\dot{\theta}_1$, $\dot{\theta}_2$, so we need to say X_1 is q_1 , $X_2$is q_2 or X_1 is θ_1, $X_2$is θ_2 and then we write X_3 and X_4 as $\dot{\theta}_1$ and $\dot{\theta}_2$. So, in general what we have is X_1 through X_n are the generalized coordinates q_1 through q_n and X_{n+1} to X_{2n} are the derivatives of the generalized coordinates. So, we had n q's and n $\dot{q}$s, so hence we have X which are 2n of them.

And then we write this n second order ODE's as 2 and first order ODE's and how do we do that? You can see that the $\dot{X}_1$ which is nothing but dot is same as X_{n+1}, similarly $\dot{X}_2$ which is $\dot{q}_2$ will be X_{n+2} and so on. So, $\dot{X}_n$ will be X_{2n} and what about $\dot{X}_{n+1}$ all the way to $\dot{X}_{2n}$? That we can obtain from these equations of motion which is $M^{-1}\left(\tau - [C]\dot{q} - G - F\right)$,.

So, instead of writing it as q and $\dot{q}$ we can write that C (X), this τ is a function of time, so it is not a function of X. But gravity could be a function of q but it is not a function of $\dot{q}$ but nevertheless we write it as G as a function of X, the friction will be a function of both q and $\dot{q}$ which is a function of X. Remember, there are 2n of these X's, so we have n first order equations of the form $\dot{X}_1 = X_{n+1}$, $\dot{X}_2 = X_{n+2}$ and so on.

And then $\dot{X}_{n+1}$ all the way till $\dot{X}_{2n}$ as equations of motion which is $M^{-1}\left(\tau - [C]\dot{q} - G - F\right)$,. So, we have converted n second order equation into 2n first order equations, this is a very standard technique. Those of you who have used it should be familiar but otherwise it is very easy to convert a second order equation into 2 first order equations. So, hence if you have n second order equations we will get 2n first order equations.

(Refer Slide Time: 26:34)

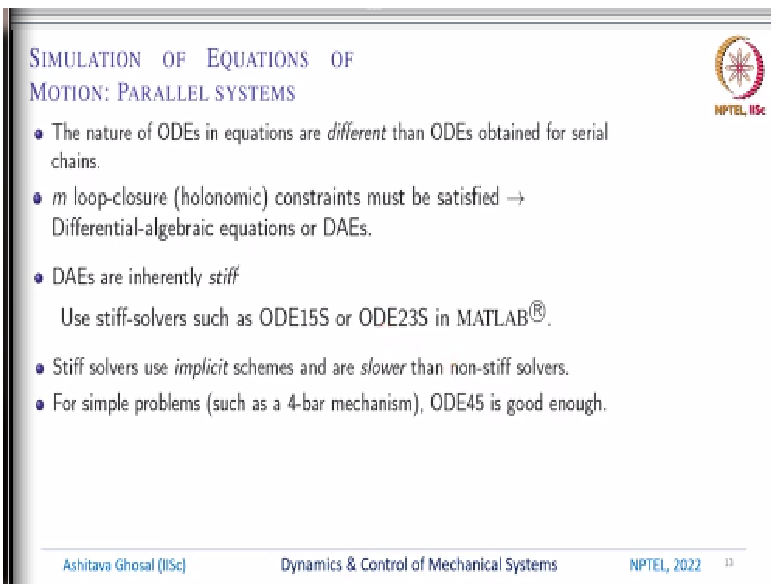


So, for parallel systems and closed-loop mechanisms the equations of motion are little bit more complicated. So, we have $M\ddot{q} = f - \psi]^T(\;[\psi]M^{-1}[\,\psi]^T)^{-1}\{\;[\psi]M^{-1}f + \;[\dot{\psi}]\,\dot{q}\,\}$and all those things, where f now denotes $\left(\tau - [C]\dot{q} - G - F\right)$. So, again we can obtain 2 $(n + m)$ first order equations because here the number of equations after elimination of lambda which is this is n + m.

So, when you convert it into first order equation, we will get $2(n + m)$ first order equations and again it is very easy. $\dot{X}_1 = X_{n+m+1}$, $\dot{X}_2 = X_{n+m+2}$ and so on. And $\dot{X}_{n+m+1}$ all the way till $\dot{X}_{2(n+m)}$ are obtained from these equations of motion. **okay** And again this f denotes $\left(\tau - [C]\dot{q} - G - F\right)$, so it is a very reasonably straightforward approach of converting n second order ordinary differential equations into 2 n first order ordinary differential equation.

In the case of parallel systems and closed loop mechanisms we have 2 $(n + m)$ first order equations.

(Refer Slide Time: 28:11)



The nature of the ODE is however as quite different from the ODEs obtained from serial chain, let us elaborate a little bit on that. So, we started with m loop-closure holonomic constraints which needed to be satisfied. Remember, we had broken this 4-bar linkage at some point and then we went from one side to that point and we went from the other side to the point and we have equated the X and the Y coordinates. So, they were loop-closure constraint equations they were holonomic.

So, what we have inherently is a set of differential equations and some holonomic constraints which involve only the position you know x, y θ_1, θ_2, they do not contain $\dot{\theta}_1$ and $\dot{\theta}_2$ or $\ddot{\theta}_1$ and so on. So, these are called as differential algebraic equations in numerical analysis. And DAEs are not the same as ODE’s and I do not want to go into full details but DAEs are what are called as inherently stiff equations.

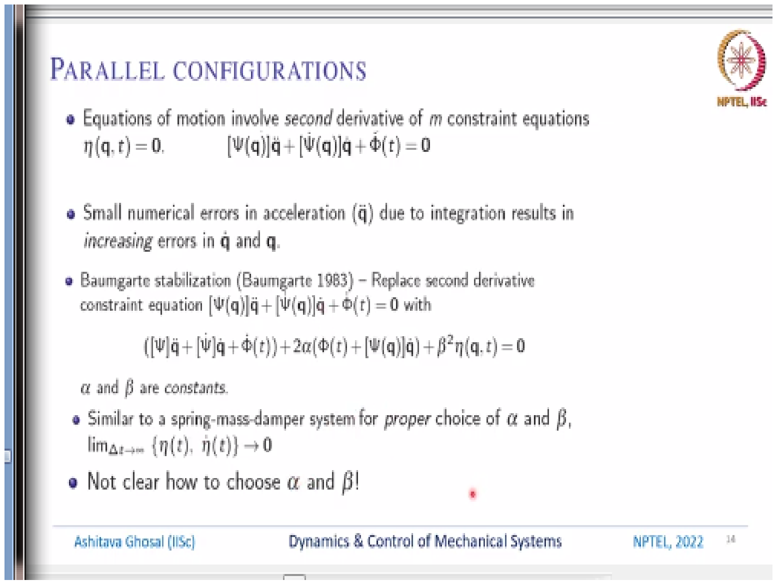
So, basically what you need to do is most of the time you can get away with ODE45 but for some set of differential equations or differential algebraic equations you need to use what are called stiff-solvers and MATLAB gives you these stiff-solvers. So, this ODE15S is the way to solve stiff differential equations, so ODE23S is another such product or another such package or another such routine which are given by MATLAB to solve stiff equations.

So, please take a look at MATLAB reference material and see how ODE15S or ODE23S is really different from let us say ODE23. There is a ODE23 also and there is an ODE23S, so what is the difference? You can see yourself. So, stiff-solvers in general broadly speaking they use what are called as implicit schemes and they are slower than non-stiff solvers. Anybody who knows what is implicit should know that the opposite of implicit is explicit.

In an explicit scheme you can march forward in time, in an implicit scheme at every point you have to solve a set of linear equations, so hence they are slower than non-stiff solvers. For simple problems such as the 4-bar mechanism ODE45 works fine, so we do not have to use any of the stiff-solvers but you can try, it is very easy to change in a MATLAB code from ODE45 to say let us say ODE23S.

All you need to do is wherever you are calling the solver, ODE solver you can change from ODE45 to ODE23S nothing much needs to be changed. And then you can see what happens when you solve a differential equation with ODE45 as opposed to a stiff-solver.

(Refer Slide Time: 31:52)



So, let us continue in parallel configurations. There is also one another big difference which is the following. So, actually what we want to solve is this $\eta\,(q,\ t) = 0$, so this is the actual loop-closure constraint equations. However, what we have actually done is? We have taken the derivative of this and then the second derivative of this and we are using the equation after taking the second derivatives to eliminate the Lagrange multipliers.

So, we are actually solving the constraint in this form, not in this form and what is the difference between these 2? What you can see is instead of $\eta\,(q,\ t)$ suppose you have $\eta\,(q,\ t) + c_1 + c_2$ t = 0, where c_1 and c_2 are some small numbers. So, when you take the $\ddot{\eta} + c_1 + c_2$ t = 0 what you will get is this. So, any small changes numerical errors in $\ddot{q}$ due to integration will cause increasing $\dot{q}$ and q because you are not solving this, you are solving actually this.

So, what you will get is not $\eta\,(q,\ t) = 0$ but it is possible that you are using $\eta(q) + c_1 + c_2$ and this c_1, c_2 t if c_1 and c_2 are increasing with time, so the actual $\eta(q)= 0$ will not be satisfied. And this c_1 and c_2 can arise always due to the numerical solvers, so we are using some numerical techniques to solve the differential equation, so it will never be exactly satisfying the constraint equation, so it will be some small changes.

And because we are using the second derivative we will slowly grow with time. So, Baumgarte in 1983 proposed a way to actually try and solve for $\eta\,(q,\ t) = 0$. So, what he said is we replace the second derivative constraint equation which is this with something like this. So, instead of $\psi\ddot{q} + \dot{\psi}\,\dot{q} + \dot{\phi}(t) = 0$, we will replace it as $\psi\ddot{q} + \dot{\psi}\,\dot{q} + \dot{\phi}(t) + 2\alpha(\phi(t) + [\psi(q)]\dot{q}) + \beta^2\eta\,(q,\ t) = 0$.

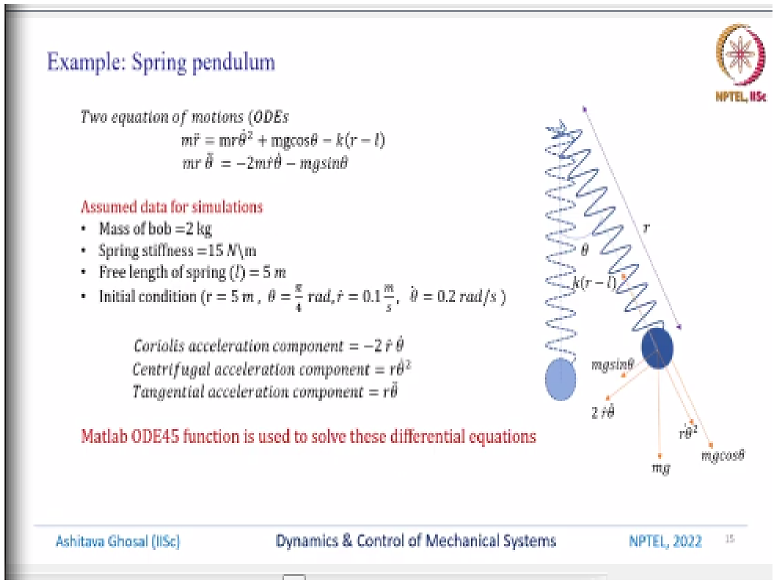
So, if you look carefully this sounds or looks very much like a spring mass damper system, where $\alpha\ and\ \beta$ are constants. So, this is like $\ddot{X} + 2\,C\,\dot{X} + \omega^2 X$. So, if you choose $\alpha\ and\ \beta$ just like in a spring mass damper system properly the oscillations die down and what is oscillation here which is $\ddot{X},\ \dot{X}\ and\ X$. So, if you choose $\alpha\ and\ \beta$ a properly then X will go to 0, $\dot{X}$ will go to 0 and $\ddot{X}$ will also go to 0.

In a spring mass damper system, if it is damped properly it will us $\ddot{X},\ \dot{X}\ and\ X$ all of them go to 0, so that is what he suggested. Then instead of trying to use this equation to find the Lagrange multipliers and finally find the equations of motion we will use this equation. Then the question comes, how do we choose $\alpha\ and\ \beta$? You might say that it looks like a spring mass damper system but it is not really a spring mass damper system.

This is nowhere like $[M(X)]\ \ddot{X}$, this is nowhere like $C\,\dot{X}$ or this is nowhere like $K\,X$ because it is a very non-linear function. But nevertheless it is a reasonably good idea if I play around with this $\alpha\ and\ \beta$ I can ensure that not only $\eta\,(q,\ t)$= 0 but $\dot{\eta}$ of time will also go to 0, so all of them will go to 0. And hence the loop-closure constrained equations are exactly satisfied at 0, so this is called Baumgarte stabilization.

So, the errors in the loop-closure constraint equations will not grow with time because they as t tends to infinity they will go to 0. But it is not very clear how to choose $\alpha\ and\ \beta$; you have to play around with $\alpha\ and\ \beta$.

(Refer Slide Time: 37:29)

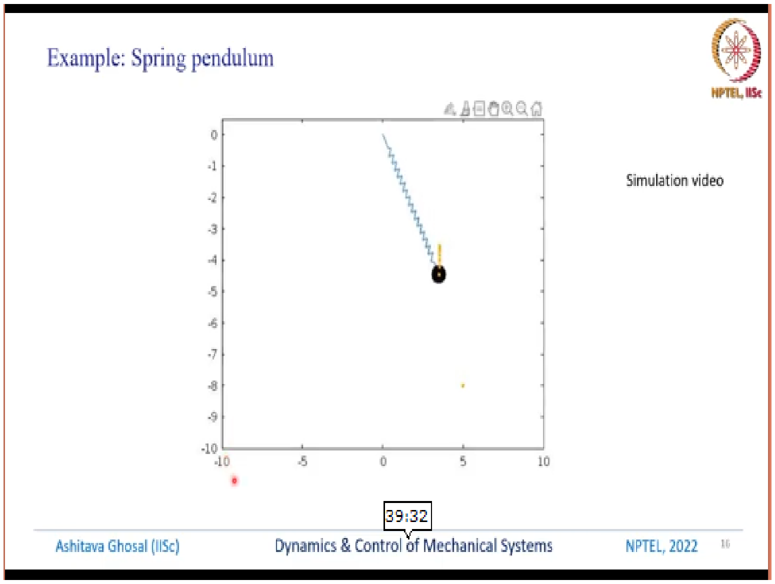


Let us take some examples now. So, another example is that we have worked on earlier which is called the spring pendulum. So, basically we had a mass here which is mounted on a spring, the free length is l and then once the spring is compressed we have a force which is $k(r - l)$. And there is gravity acting and then there is these components $mg \sin \theta$, $mg \cos \theta$ and also we have this centripetal term and we have this Coriolis term.

So, in order to solve this equations we have 2 equations basically 2 ODE's and then we can write down this ODE's and we need to now find out some numbers for the mass of this bob, spring stiffness, free length and some initial conditions. So, for the example we are assuming that the mass is 2 kg, the spring system spring stiffness is 15 Newton per meter, the free length is 5 meters and initial conditions are r is 5 meters, $theta\ is \frac{\pi}{4}$, $\dot{r} = 0.1\ and\ \dot{\theta} = 0.2$ radians per second.

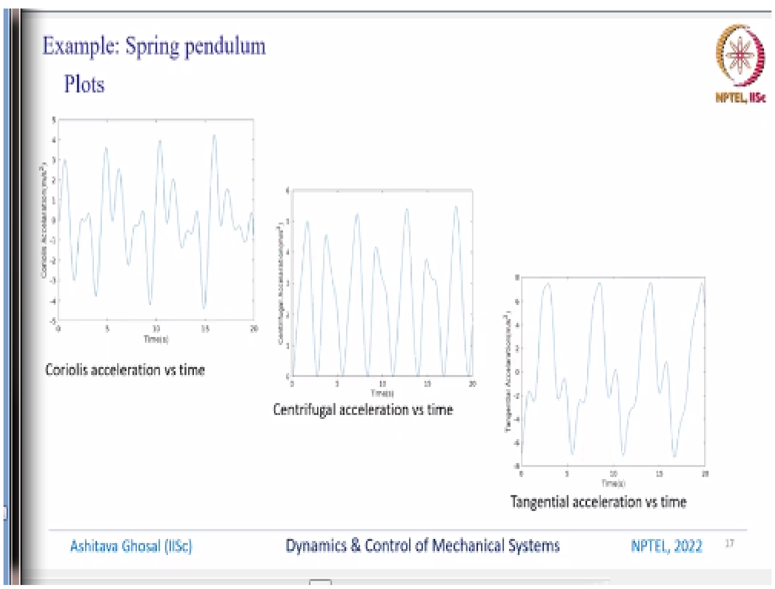
And we will also compute the various components of the acceleration just to show how they are changing with time. So, both of these 2 equations can be converted to first order ODE's and then we will use MATLAB ODE function to solve these differential equations.

(Refer Slide Time: 39:17)



(Video Starts: 39:34) So, here is a video of this spring pendulum and you can see this animation of this mass which is the bob as it is going around. So, what you can see is the spring is lengthening and shortening, so unlike a pendulum there is a spring which is here and the spring can go in and out and this mass can also go in and out. So, this is the simulation which is obtained after solving those 2 differential equations. **(Video Ends: 40:07)**

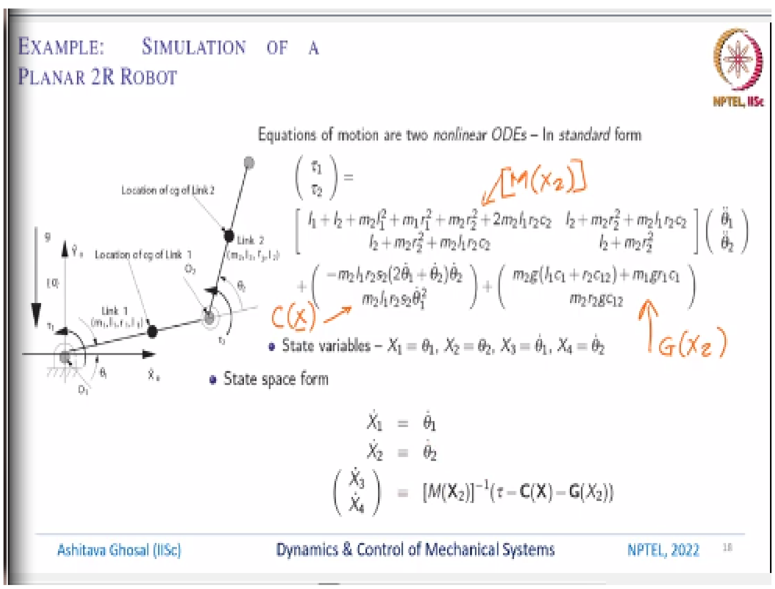
(Refer Slide Time: 40:07)



And as I said I want to plot the various acceleration, just to get a feel for what kind of numbers we are talking about. So, the coriolis acceleration from 0 to 20 which is the motion of this bob looks like this. The centrifugal acceleration which is $\dot{\theta}^2$ and this is $2\,r\,\dot{\theta}$ and this is $r\,\dot{\theta}^2$ looks like this and the tangential acceleration which is $\ddot{\theta}$ looks like this.

So, what you can see is this coriolis acceleration is maybe slightly smaller; the tangential acceleration is probably the largest. Of course this is because we have chosen some $k,\ m,\ r$ and so on, if it is some other values maybe these plots could look differently. But the point is we can take these 2 differential equations, we can use ODE45 and we can find out the motion of the bob which is what I showed you in the video, not only that we can find out the velocities and accelerations and I am showing you 3 of the components of the acceleration.

(Refer Slide Time: 41:31)

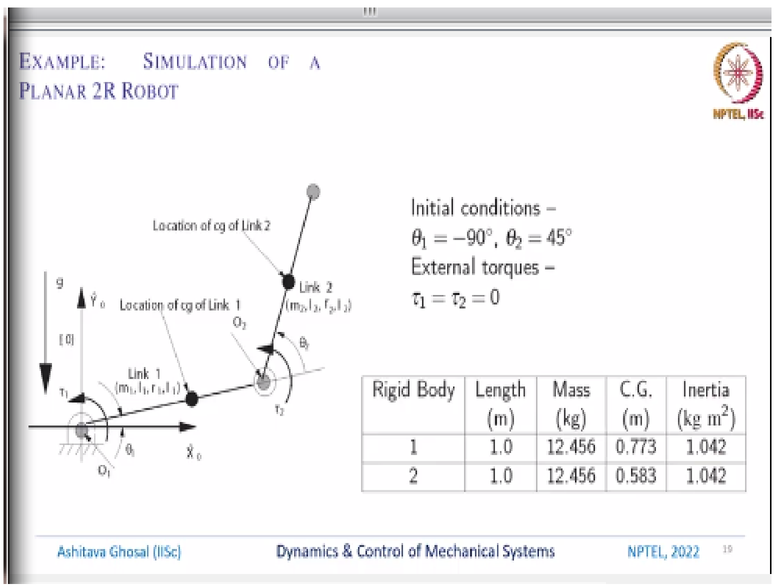


Let us continue, now we will see how we can simulate the motion of a planar 2R robot. So, we had this 2R robot the first link was m_1, l_1, r_1, I_1 this is the location of the CG, there is a torque which is acting, gravity is acting this way, this is a τ_2 which is acting on the second link, **okay** it is exactly the same, nothing new. And we can have these equations of motion which is τ_1, τ_2 is given by some $M(X_2)$ and $C(X)$ *and* $G(X_2)$. So, why X_2?

Because it is only θ_2 which is important whereas the coriolis term can contains θ_1, θ_2, $\dot{\theta}_1$, $\dot{\theta}_2$ which is what this vector X is. The mass matrix is only a function of θ_2 which is why I have written it as $M X_2$. So, this is a 2 by 2 matrix which is a function of only θ_2, remember θ_1 was X_1, θ_2 was X_2, X_3 was $\dot{\theta}_1$, X_4 was $\dot{\theta}_2$. So, the state variables as $X_1 = \theta_1$, $X_2 = \theta_2$, $X_3 = \dot{\theta}_1$, $X_4 = \dot{\theta}_2$.

And we can write it in the state space form which is $\dot{X}_1 = \dot{\theta}_1$, $\dot{X}_2 = \dot{\theta}_2$, $\dot{X}_3$,$\dot{X}_4$ is this. So, we have written it in the state space form, so actually $\dot{\theta}_1$, we have to write it as X_3, $\dot{\theta}_2$ we have to write it as X_4. So, because we want the state equations as $G(X\ ,\ \tau)$ to be equal to 0, so we have written it in the state space form.

(Refer Slide Time: 43:30)



So, once we have those 4 first order equations in the required state space form we can give some initial conditions. So, for this simulation we are choosing θ_1 as -90 and θ_2 is 45 degrees, so what does it mean? This link is hanging down because it is -90, so this angle is -90 and then θ_2 is 45 degrees, so we have a link which is hanging down and the second link is like this at an angle of 45 degrees and there is no torque.

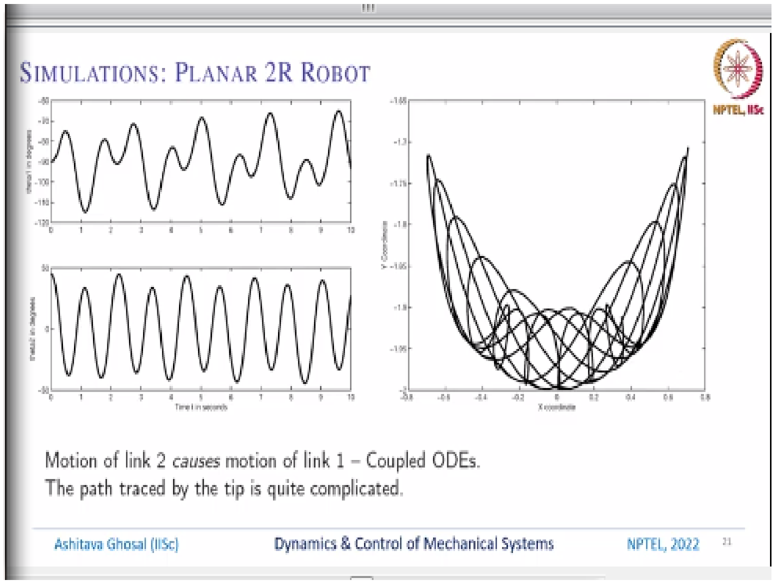
Again just to illustrate that we can solve all kinds of differential equations, we have chosen this is a nice interesting case that there are no external torques, we can put τ_1 and τ_2 are 0, so it is like a double pendulum. So, we have this pendulum one link and then the second link, so the first link is hanging vertically down and then we displace the second link by 45 degrees and then we let it go. So, I want to see what the motion of these 2 links is.

So, in order to solve these differential equations we need numbers. So, again we have assumed length is 1 meter, mass is 12.456 equal, CG is this and inertia is 1.042, so these are the same numbers which were used for the inverse dynamics. **(Video Starts: 45:07)** So, here is a video of the 2 degree of freedom robot. So, as I said our double pendulum, as I said we will put the first link down and the second link is rotated by 45 degrees and then we leave it.

So, as you can see the motion of the tip is quite complicated, it is not like a simple arc of a circle, it is going forward and doing all kinds of loops and then it is coming down. So, I am not sure how many of you expected that it will do all kinds of crazy motions like this but a double pendulum actually does motions like this. And I can clearly see that for this special case of this link length which is 1 meter and 1 meter and no external torques and the CG located at some places, this is the motion of the tip.

I could have also plotted the motion of any other point on this link or any other point on this link, MATLAB once you find θ_1 as a function of time, I can find out what the CG is doing. What is the CG? This is the distance r_1 along the link, so we can plot what is CG is doing. **(Video Ends: 46:42)**

(Refer Slide Time: 46:42)

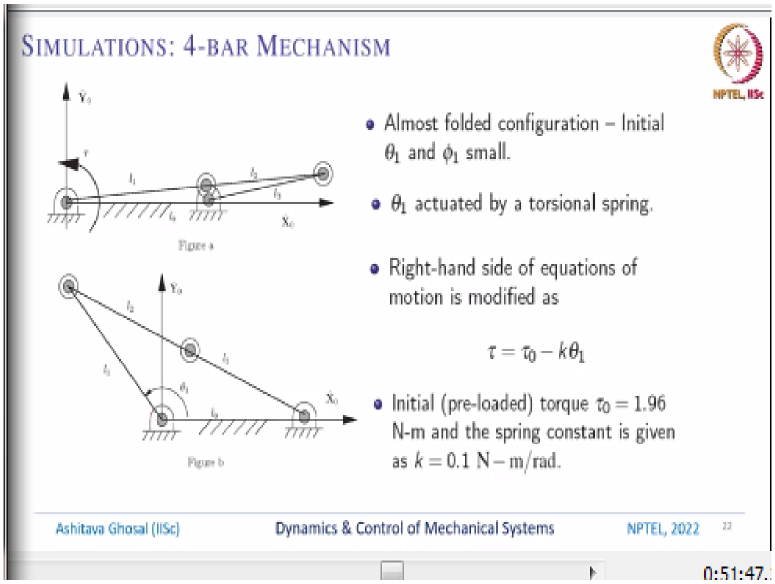


We can also plot actually θ_1 and θ_2 as a function of time. So, what you can see is θ_1 is doing all kinds of things like this, so it is not a simple harmonic motion although we do not should not expect a simple harmonic motion like a sine wave. The second link is little bit more periodic because again it is not because what you can see is the amplitude is changing as we go along. The first link is definitely doing interesting things. So, why does this happen?

Basically what is happening is that the 2 differential equations they are coupled. So, once you leave the second link the motion of the second link causes a motion of the first link. Remember, we started with θ_2 as 45 degrees and θ_1 is -90 degrees, so it is vertically hanging down with second link displaced by 45 degrees. So, when you leave the second link it will start oscillating but then due to the motion of the second link the first link will also start moving and that is the nature of this coupled ODE's.

And the path traced by this tip of this second link is as I showed you it is quite complicated. And we can solve the equations of motion and exactly show you what the tip motion is. So, we started from somewhere here and it went down and it went this and did all kinds of strange motions.

(Refer Slide Time: 48:27)



We can also simulate the 4-bar mechanism. So, we have derived the equations of the 4-bar mechanism. And what I am going to do is we are going to place the 4-bar mechanism in this almost folded configuration. So, the first fixed link is this, the second link is this way, the third link is this way and the fourth link is this, so this is exaggerated a little bit. But basically all the 4 links are lying along the X axis this is called as the folded configuration.

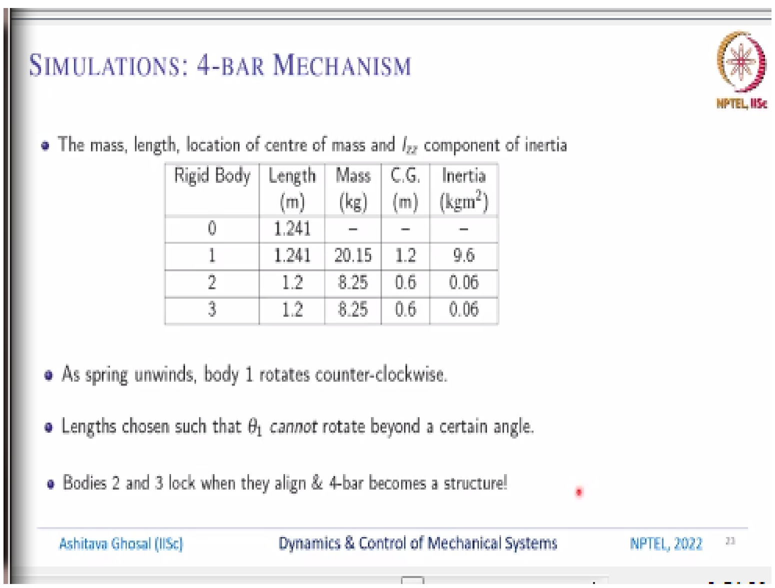
The initial θ_1 and ϕ_1 are small, so it is little bit exaggerated here in this drawing. And what you can see is you can do some simple calculations that if you choose l_1, l_2, l_3 and l_0 in a particular way eventually this mechanism and if you start to rotate l_1 by after some angle the rotation θ_1 it will stop and it will form a triangle. That is the reason why because l_1 and l_2 and l_3 and l_0 are chosen in this way.

So, that it cannot rotate beyond this some angle θ_1 because it will form a triangle. So, in the actual simulation we will assume that this θ_1 is rotated by means of a torsion spring. So, θ_1 is actuated using a torsion spring, the right hand side of the equations of motion is now modified because now we have tau which is acting here and the first joint is some $\tau_0 - k\theta_1$.

So, the idea is that there is a constant initial torque but as θ_1 increases the torque which the spring is supplying reduces. So, there is a initial preloaded torque τ_0, we have chosen this number 1.96 Newton meter and the spring constant is given as 0.1 Newton meter per radian. And what is the basic idea that I do not want to keep on applying this constant torque because if you go back to your basic mechanics this torque will be something like $I\,\alpha$.

So, if the torque is constant α will be constant but then integral of α which is omega will increase with time and integral of omega is rotation which is θ_1 that will be like square, so it will keep on increasing in a parabolic path approximately, so we do not want that. We want eventually that θ_1 should stop.

(Refer Slide Time: 51:48)



So, for this simulation we assume that these are the masses and lengths. So, for this rigid body 1, it is 1.241 meters, for the fixed length it is 1.241 m , so both the crank and the base is 1.241 meters. The second link is 1.2 m; the third link is 1.2 m , so if you choose these kinds of link lengths then θ_1 cannot rotate fully. As I showed you in the picture before after a while it will form a triangle and the motion must stop.

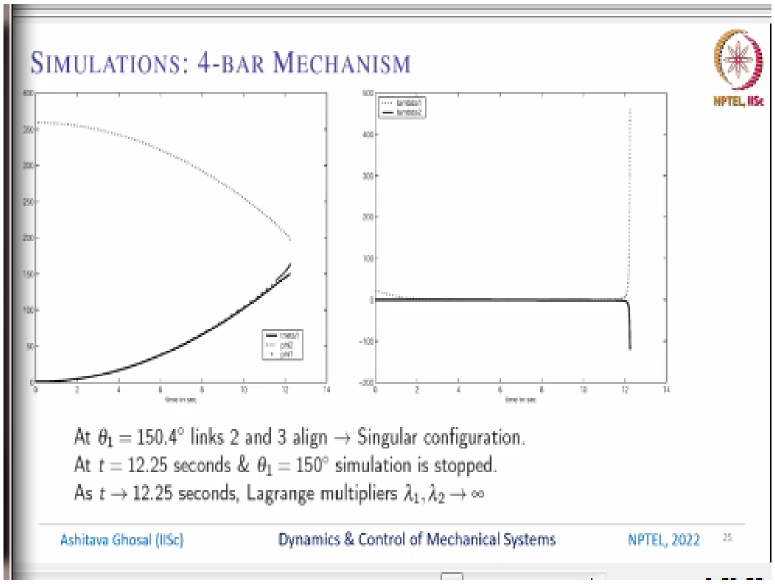
So, the masses are 20 kgs initial crank is the heaviest and all other links are smallest. So, this is actually some numbers which were obtained from a actual physical device, so these are what are called as deployable systems. So, you keep these links in a horizontal position and then you go to space or somewhere else and then we actuate the actuator, or you start giving torque due to a spring or a motor and then it will unfold and it will form the triangle.

So, there are many such applications where you start from a folded configuration and then you go to an unfolded configuration. So, we want to know how does these 4-bar mechanism moves as it unfolds. So, as the spring unwinds body 1 rotates counter-clockwise the lengths are chosen such that θ_1 cannot rotate beyond a certain angle. So, the bodies 2 and 3 will lock when they align and the 4-bar will become a triangle.

And if you go back to some statics a triangle is basically like a structure, the degree of freedom of a triangle is 0, the number of links, the number of joints and so on you can see that the degree of freedom is 0 and it will become a structure. **(Video Starts: 54:07)** So, here is a video of this 4-bar. So, as you can see this links are almost flying in the horizontal direction. So, you saw this motion, so the basic idea is it starts up very fast but then the torque is some $\tau_0 - k\theta_1$ so the torque is reducing.

So, towards the end it slowed down and then we form this triangle. So, what else can we find out from once we have such a system and once we have solved the differential equations using MATLAB what else can we find out? **(Video Ends: 54:55)**

(Refer Slide Time: 54:56)



So, what you can see is we can plot the various angles. So, for example θ_1 is this dark line, ϕ_2 which was the second angle is this dotted line and then ϕ_1 which is the output angle is like this. We can also plot the 2 Lagrange multipliers, remember this is a closed-loop system; this has 2 loop closure constraints. And associated with those 2 loop closure constraints we have two λ_1 *and* λ_2 , so I can plot λ_1 *and* λ_2 as a function of time.

And what you can see is after about some 12 point something seconds this lambda was an and λ_2 go off to infinity. So, what does it means? That we do not have 2 Lagrange multipliers after some time, at that time 12 point something seconds 2.5 seconds exactly θ_1 is 150.4 degrees and this link 2 and link 3 will align, so they will become like 180 degrees and θ_1 is approximately 150.4, at 12.25 the Lagrange multipliers go off to infinity.

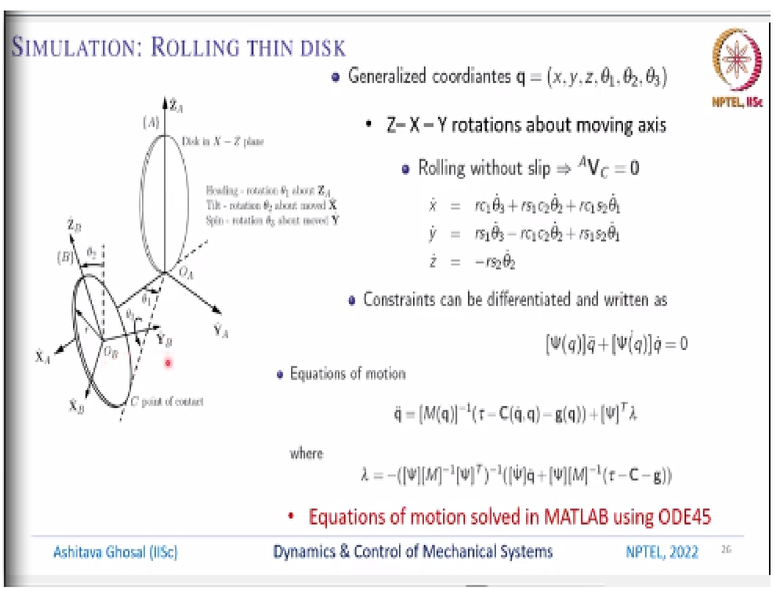
So, if you go back and recall the Lagrange multipliers are basically denoting some constraint forces. So, what do we have? As this mechanism is unfolding it is going from a 4-bar to a triangle at some point of time this Lagrange multipliers go up to infinity, so the constrained forces are going off to infinity. And physically what it means is the mechanism can no longer rotate, so there is lot of internal forces in the second and the third link and that is what this Lagrange multiplier is going off to infinity is showing.

So, the useful things in this simulation are the following. That I can find out exactly when this 4-bar will become like a triangle, when these 2 links will align and it will lock and then we have a triangle and that is what approximately 12.25 seconds. And then we can also find out what is the forces which are your joint should see at around 12.25 seconds. The simulations cannot proceed beyond 150.4 degrees, why?

Because it is no longer moving, it is a structure; it is a triangle at that point of time. So, the second order differential equation is no longer valid then, it is not moving anymore. So, the simulations are telling me what should be the time for deployment and also at what angle it will stop. So, for different link lengths this time and this angle will be different, so we can have various designs of this 4-bar mechanism, if you are planning to use this 4 bar mechanism for some application where we want to deploy it.

So, we can try out these simulations in the lab, in front of computer and then find out what is the time, what is the forces and so on. And this is one of the goals or usefulness of simulations, I can give you some very good idea about what exactly this 4-bar mechanism or for that matter any other multi rigid body systems is doing is going to do.

(Refer Slide Time: 58:56)



Let us continue, let us look at this simulation of the rolling disk. So, the first simulation was 2R robot with no constraints, the second simulation of the 4-bar mechanism had these loop closure constraints and then we had these Lagrange multipliers. In this third simulation we impose what are called as non-holonomic constraints, so constraints involving velocities. So, quickly the generalized coordinates for this rolling disk are x, y, z which is the center of this disk.

There is a rotation matrix between this disk and this reference coordinate system which is the $Z - X - Y$. Z because the first rotation is θ_1 about the Z axis, the second rotation is θ_2 which is the tilting of the disk about the X axis and the third rotation is about Y axis which is the spinning of the disk. So, θ_1 is called heading, θ_2 is called tilt, θ_3 is called spin. If you have rolling without slip the constraint is that the velocity of the point of contact is 0 and velocity involves derivatives of this generalized coordinates.

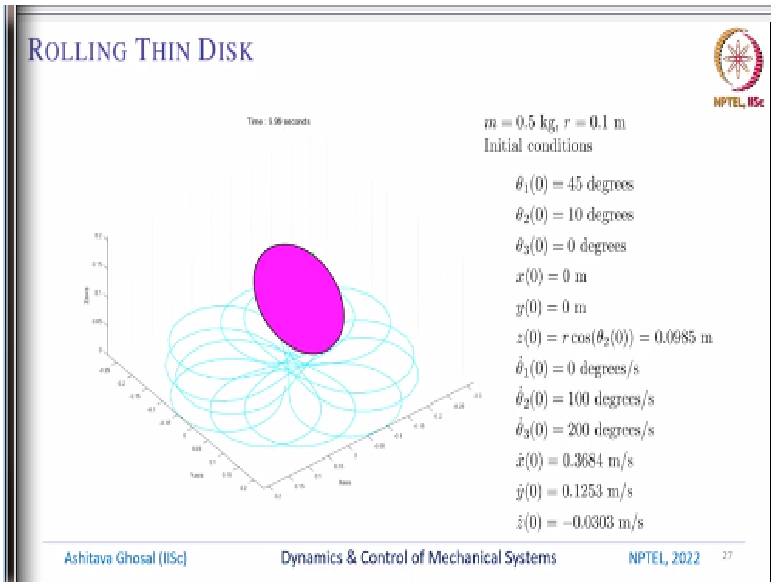
In the case of loop closure constraints there was no derivatives involved in the constraint. And these constraints of that point of contact velocity being 0 can be written in this form, this was done earlier also. So, we had $\dot{x}$ as a function of $\dot{\theta}_3$, $\dot{\theta}_2$ and $\dot{\theta}_1$ in this form $rc_1\dot{\theta}_3 + rs_1c_2\dot{\theta}_2 + rc_1s_2\dot{\theta}_1$ and so on. And the $\dot{z} = -rs_2\dot{\theta}_2$, again c_1 means cos θ_1, s_2 means sin θ_2 and so on.

So, these constraints can be differentiated and written in this form, again it is repeating, we have done this earlier also. And then the equations of motion are given by $\ddot{q} = M^{-1}\left(\tau - [C]\dot{q} - G\right) + [\psi]^T\lambda$, the λ are the Lagrange multipliers introduced for these 3 constraints. We can solve for that lambda with this long complicated expression involving $\left([\psi]M^{-1}[\psi]^T\right)^{-1}$.

And then we can obtain the equations of motion and they can be solved in MATLAB using ODE45. So, we will substitute lambda back into this equation, so we have second order equations, how many second order equations do we have? There are 6 second order equations because q is x, y, z, θ_1, θ_2, θ_3, so $\ddot{q}$ will be 6 of them. And then when you convert it into first order form we will have 12 first order equations.

In this case also there are no external torqueses, this is simply rolling, gravity is there but there are no external torques. So, if this was like a wheel with a motor here then we would have to take into account the torque given by the motor, so that is a separate story altogether.

(Refer Slide Time: 1:02:48)



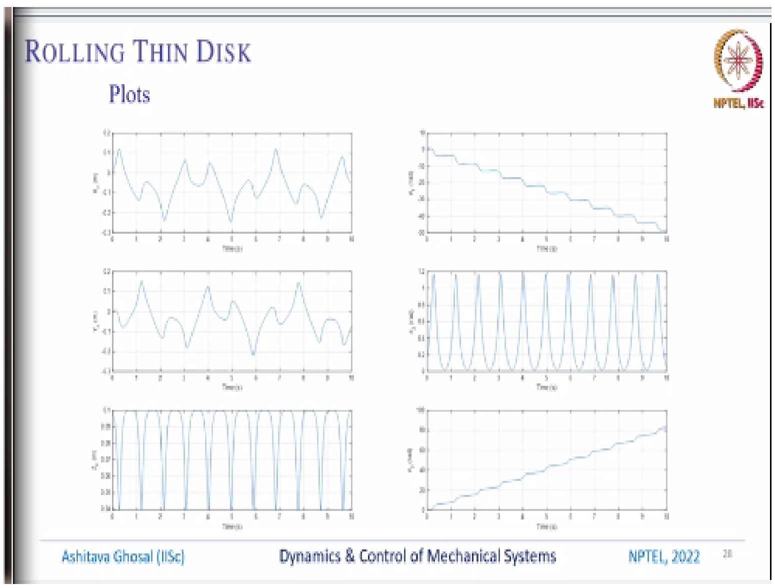
So, let us look at some simulations. Before we do any simulation we have to assume some numbers because any simulation software we have to give some numbers. So, in this case we are going to assume that the mass is 0.5 kg, the radius of this disk is 0.1 meters and then we need initial conditions for all the generalized coordinates x, y, z, θ_1, θ_2, θ_3 at $t = 0$. So, we have randomly chosen $\theta_1(0)$ as 45 degrees, $\theta_2(0)$ as 10 degrees, $\theta_3(0)$ as 0 degrees, x and y is 0 basically at the origin, so we are going to start at the origin.

The z is tilted a little bit and the z coordinate is 0.0985, so if $\theta_2(0)$ is 10 degrees, then into $z(0) = \cos cos\, \theta_2(0)\ \ r$, so this is very close to 1 but not exactly 1. And we also need initial conditions for $\dot{\theta}_1$, $\dot{\theta}_2$, $\dot{\theta}_3$, $\dot{x}$, $\dot{y}$, $\dot{z}$ and these are chosen here. So, $\dot{\theta}_1$ is 0, $\dot{\theta}_2$ is 100 degrees per second, 200 degrees per second, this is 0.3684 meters per second, $\dot{y}(0)$ is 0.1253 meters per second, $\dot{z}$ = -0303.

So, these have been chosen sort of randomly by trial and error just to make the simulation interesting. So, what do you think the simulation of this disk will look like and how do we find out? So, we basically need to solve the equations of motion. **(Video Starts: 1:04:41)**

So, this video is showing what the disk is doing. So, as you can see it is doing something reasonably interesting. And at every place at the point of contact there is no slip, so as you can see that the trace of this point of contact is very, very interesting, it will make this curves on the ground. **(Video Ends: 1:05:15)**

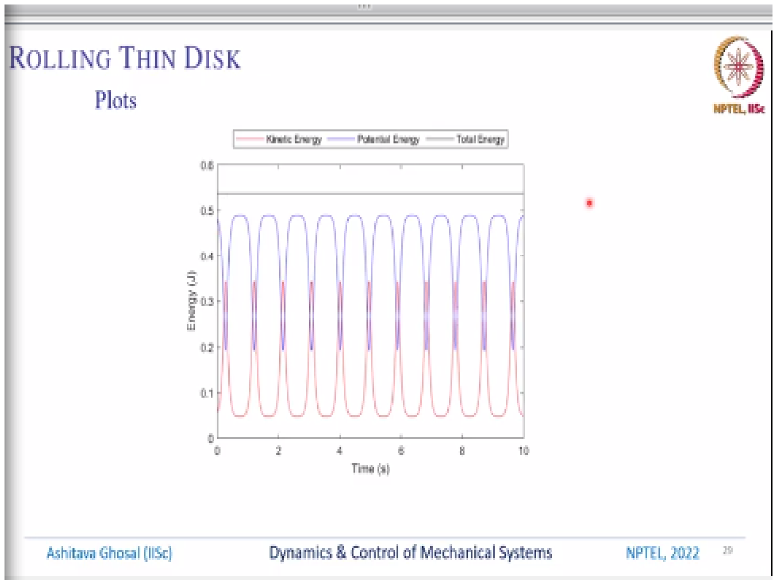
(Refer Slide Time: 1:05:16)



We can also now plot the center of the disk, how it is moving? The Y coordinate, the X coordinate, so this is the X coordinate, this is the Y coordinate and this is the Z coordinate. And this is the way the Z coordinate is changing with time, the simulation is for 10 seconds, so the Z is sort of approximately understandable it has come to 0 go back up and down, it keeps on doing this.

The X and Y coordinates are much more interesting. We can also find what θ_1, θ_2 and θ_3 are doing. So, this is the plot of θ_1, θ_2 and θ_3 in radians, so you can see there is some interesting plot as how the heading, the tilt and the spin of the disk is changing with time.

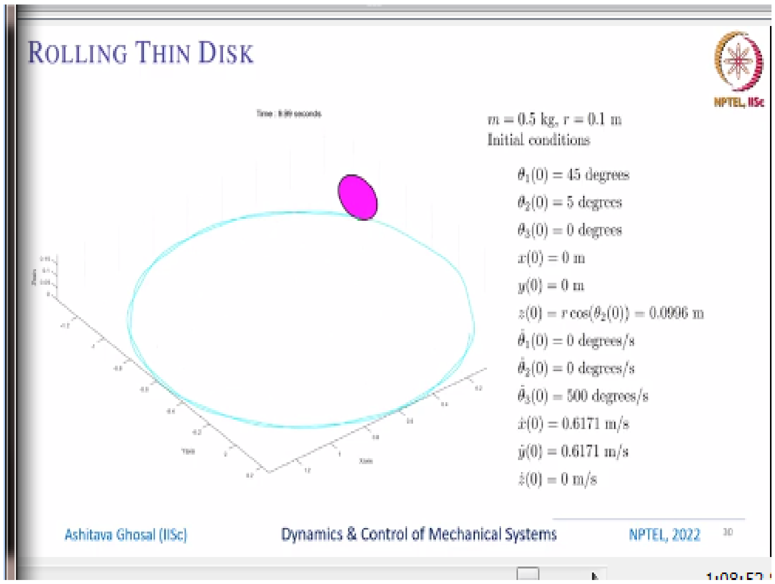
(Refer Slide Time: 1:06:09)



How do we know all these simulations are correct? One test is that we need to ensure that the total energy of the system is constant. Remember, there is no external forces, so whatever is the initial starting energy which is the kinetic energy and the potential energy, we know what is the potential energy it is like $mg\ r\ cos\ \theta_2$, we know what is the kinetic energy, there is an expression for the kinetic energy which I have shown earlier.

So, the sum of kinetic plus potential energy should be constant and we can plot that. So, in this example the red curve is that of the kinetic energy of this disk, some $\frac{1}{2}\ I\,\omega^2\ + \frac{1}{2}\, mV_c^2$ and potential energy is some $mg\ r\ cos\ \theta_2$, . So, the potential energy is the blue one and the sum of both of them is this dark one which is more or less constant. So, this gives us some confidence that whatever the code you have written, whatever the simulations you have done makes sense.

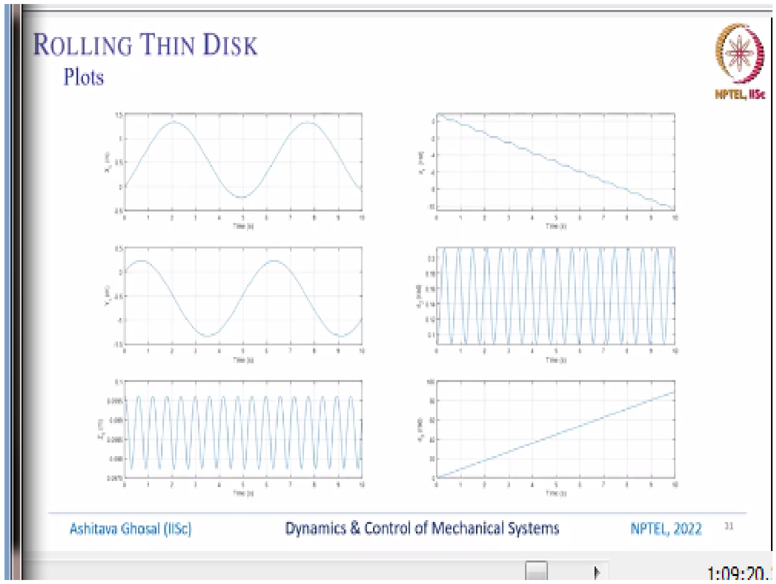
(Refer Slide Time: 1:07:27)



Let us look at another interesting example of this rolling disk. So, in this case the initial conditions are different. So, now we start with θ_1 (0) as 45 degrees, θ_2 (0) as 5 degrees θ_3 as 0, again the x *and* y coordinates are starting from origin, the tilt is around the same. But then we give some initial $\dot{\theta}_3$, initial $\dot{x}$, initial $\dot{y}$ and initial $\dot{z}$, so we are going to give some motion to the center of mass of this disk.

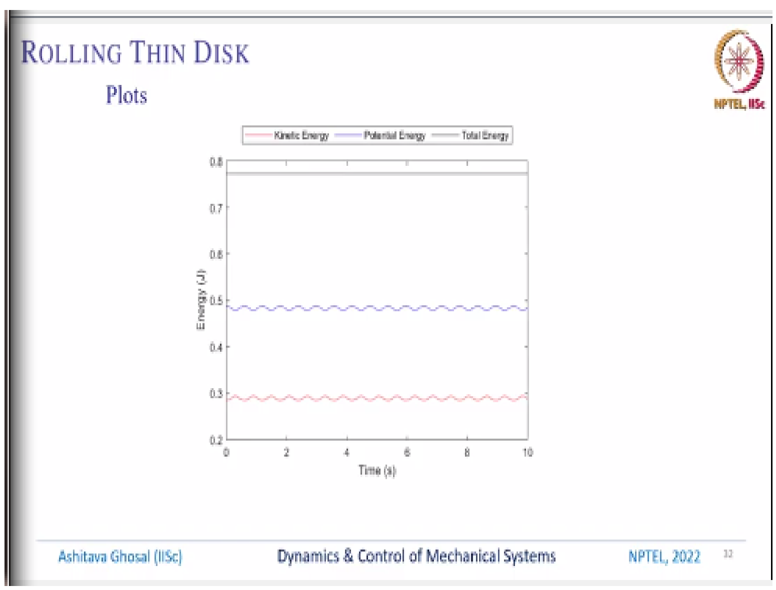
So, let us see what this disk will do now, so what is the motion of this disk? **(Video Starts: 1:08:19)** So, this is a plot which is obtained in MATLAB, MATLAB allows you to do all this simulation also do this animation. So, as you can see it looks pretty good that we have this disk which is going around in this circle, it is also tilting a little bit because θ_2 is not 0, there is some 5 degrees initially. So, it is roughly going around the circle but it is also tilting in and out of this from the vertical. **(Video Ends: 1:08:52)**

(Refer Slide Time: 1:08:53)



Again we can plot X_c, Y_c, Z_c , so these are some plots which you have obtained from MATLAB. We can also plot θ_1, θ_2 and θ_3, so θ_3 is pretty much constant it is going up, θ_2 is like some oscillation and θ_1 is also going in the opposite direction.

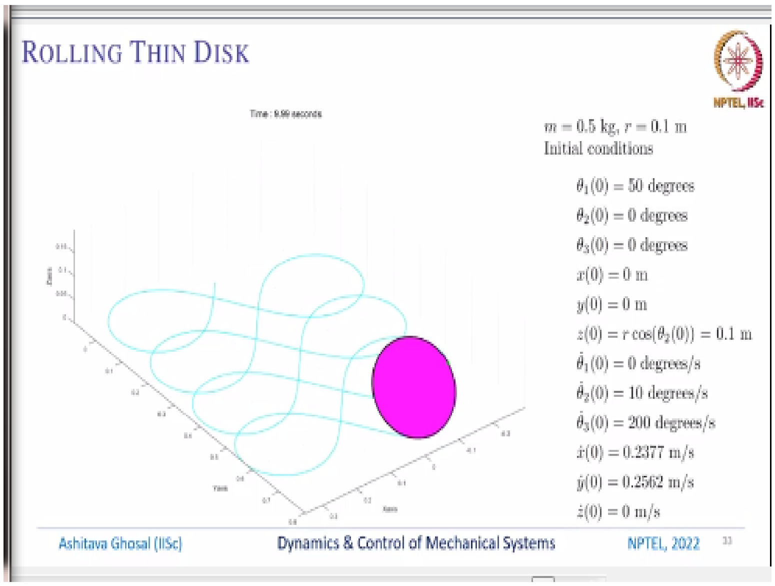
(Refer Slide Time: 1:09:21)



And again to check whether this simulation is correct or not, we can plot the kinetic energy which is this red which is oscillating like this, the potential energy which is oscillating like this. So, remember the disk is not tilting too much; it was tilting a little bit, so hence the potential energy itself is very constant. Similarly the kinetic energy is not going to change much, it is going more or less in a circle sort of like a constant speed with some little bit of perturbation.

And the sum of both of them again is whatever is the initial kinetic and potential energy and that is staying constant.

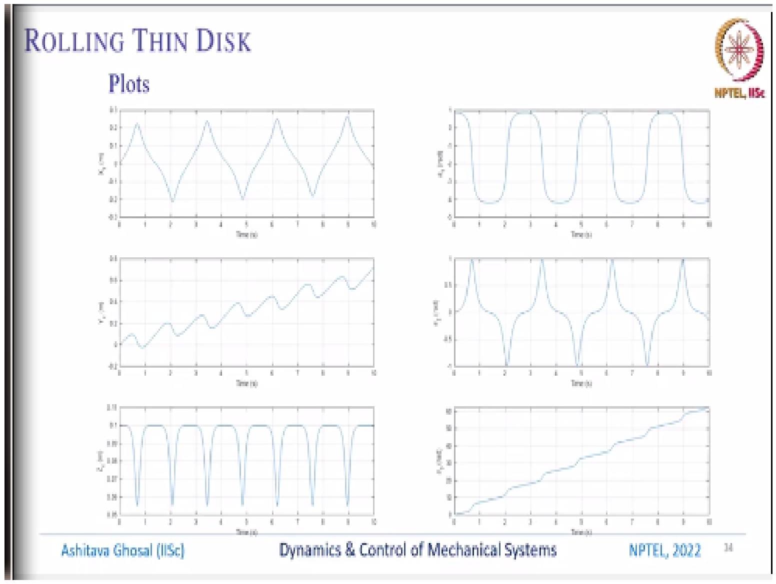
(Refer Slide Time: 1:10:04)



Let us look at third simulation, this is a nice interesting problem now you can see all these nice motions of this disk. So, again here we have some other initial conditions, so basically z now is some initially it is 0.1 meter and then we have this some different initial $\dot{\theta}_1$, $\dot{\theta}_2$ and $\dot{\theta}_3$ and also different $\dot{x}$, $\dot{y}$, $\dot{z}$. So, it is not very easy to guess what this rolling disk will do with this initial condition, so let us find out. **(Video Starts: 1:10:48)**

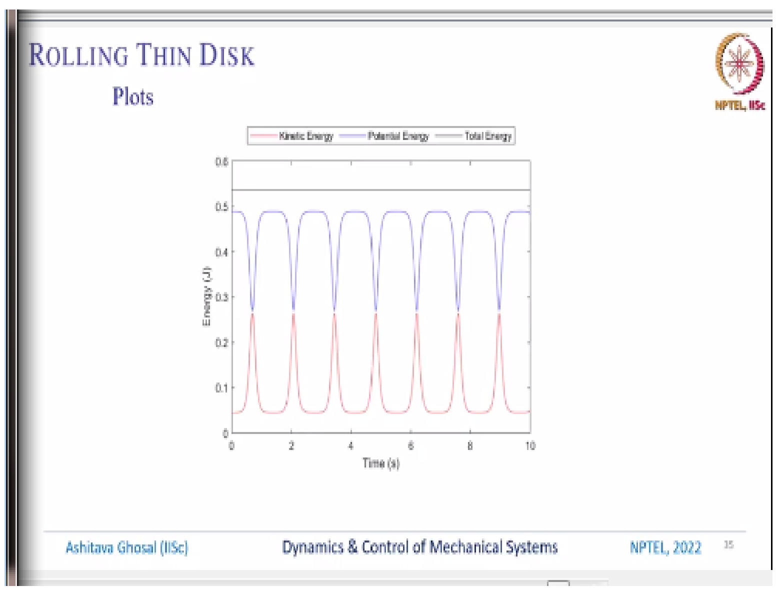
So, it is sort of similar to the first case but as you can see it is doing this but not going at the same place, it is also moving along the y axis. So, it is moving in the x and z axis but it is also moving forward and we can do all this very, very nice simulation and animations in MATLAB. **(Video Ends: 1:11:19)**

(Refer Slide Time: 1:11:20)



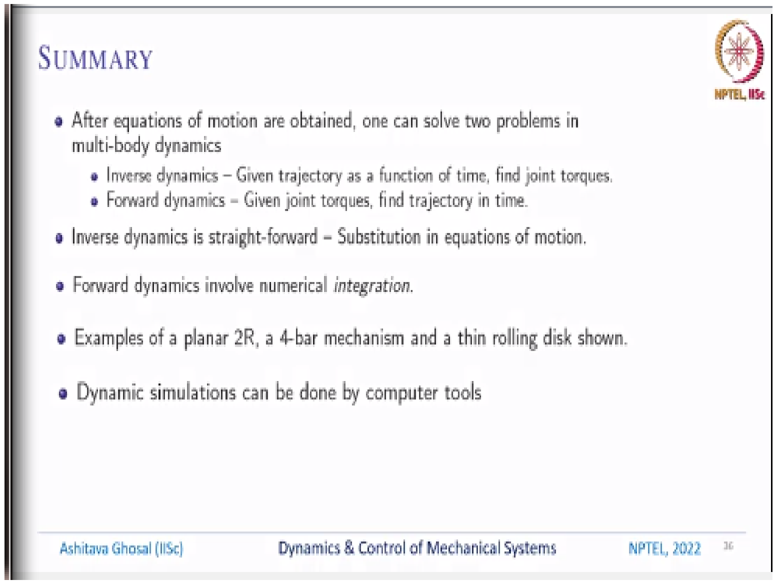
And again we can plot this x, y and z, we can also plot θ_1, θ_2 and θ_3, so this is θ_1, θ_2 and θ_3 this is x, y and z you can make this plots.

(Refer Slide Time: 1:11:40)



And we can again check that the sum of the kinetic energy and the potential energy is constant. So, again the kinetic energy is shown here, this is the potential energy and the sum is this and you can play around with these codes. So, the TA will put up this code in some available place, you can try it with some different initial conditions and you can see what the disk is doing.

(Refer Slide Time: 1:12:13)



So, in summary after the equations of motion of a multi-body system are obtained one can solve 2 problems in multi-body dynamics. One is this is called as the inverse dynamics problem which is that given trajectory as a function of time, find joint torques. This is just simple substitution on the right hand side of the equations of motion. And the next one is forward dynamics which is given the joint torques, find the trajectory in time.

And these are I have shown you for the thin disk and even the 4-bar mechanism and even the planar 2R. So, if I give some torqueses which are acting at the joint what does the mechanism do? How does it move? The inverse dynamics as I said is straightforward; it is just substitution in the equations of motion. Forward dynamics involve numerical integration, so you have to go to MATLAB with some initial conditions and in the state space form and then you can integrate.

And I have showed you examples of the planar 2R, a 4-bar mechanism and a thin rolling disk. These dynamic simulations can be done by computer tools, so right now we have derived the equation of motion and we have solved it in MATLAB. But we can also without deriving the equations of motion we can solve it in computer tools and that is what we will discuss later.