Robotics and Control: Theory and Practice

Prof. Felix Orlando
Department of Electrical Engineering
Indian Institute of Technology, Roorkee

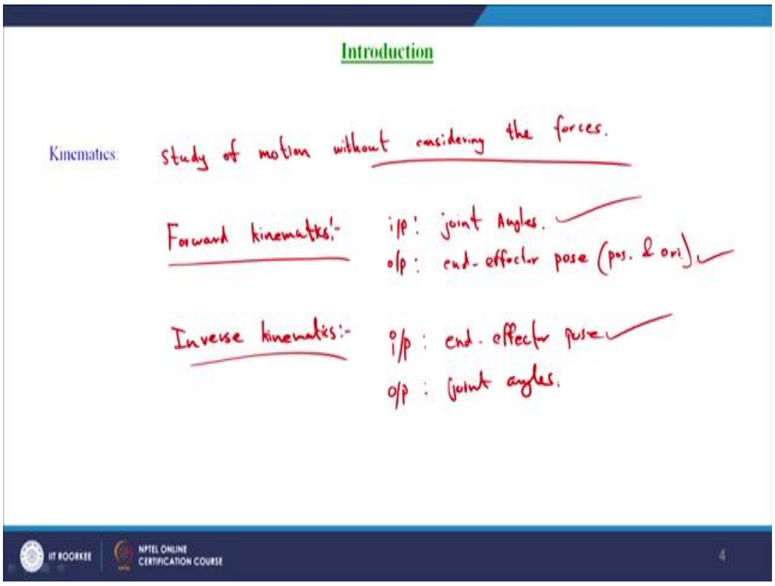
Lecture – 39
Model Based Control of Robot Manipulators

Good morning. Today we are going to see about Model Based Control of Robot Manipulators in this lecture.

(Refer Slide Time: 00:30)

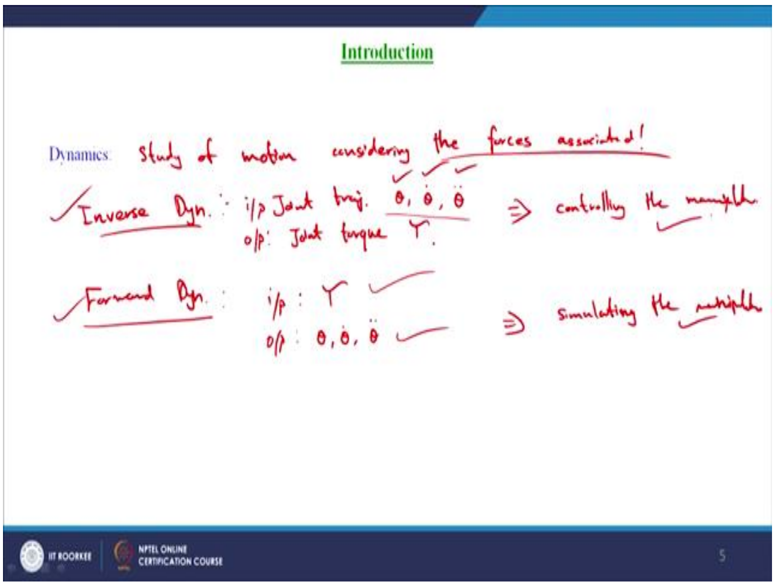
The outline of this lecture will be as follows; first we have the introduction where we will be discussing quickly about kinematics and dynamics; kinematics definition, dynamics definition what is forward and what is inverse kinematics and what is forward dynamics and what is inverse dynamics and then we come into the major portion, which is the model based control of robotic systems. Model based implies both kinematic model based scheme and dynamic model based control scheme and finally, we conclude.

(Refer Slide Time: 01:08)



Now, coming to the introduction, first we talk about kinematics what is kinematics? Kinematics is the study of motion without considering the forces associated with. Kinematics can be classified the robot kinematics can be classified into forward kinematics, which is the input is joint angles whereas, the output is end effector pose which is nothing, but position and orientation. Whereas, inverse kinematics that is nothing, but the input is end effector pose and the output is joint angle.

(Refer Slide Time: 02:49)



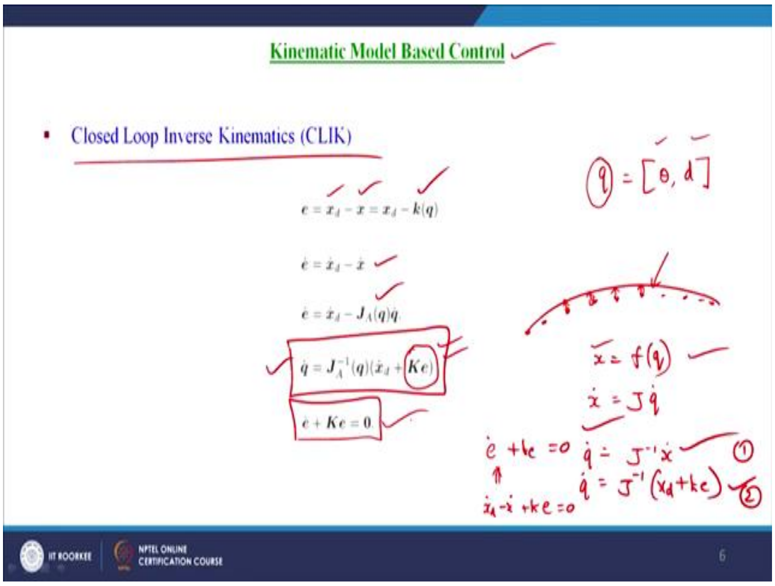
Now, coming to the dynamics portion what is dynamics? Dynamics is the study of motion considering the forces associated with that motion. Dynamics can be further classified into inverse dynamics and forward dynamics. What is inverse dynamics?. Inverse dynamics the input is joint trajectories in terms of $\theta,\ \dot{\theta},$ and $\ddot{\theta}$ position, velocity and acceleration of the joint angles and the output is this is the input is joint trajectory, the output is joint torque.

Whereas, for the forward dynamics the input is joint torque, the output is joint trajectories $\theta,\ \dot{\theta},$ and $\ddot{\theta}.$ The inverse dynamics is useful in controlling the manipulator whereas, the power dynamics is useful in simulating the manipulators. Thus inverse dynamics is meant for control of manipulators and forward dynamics is meant for simulation of manipulators.

Thus, we summarize the introduction part in such a way that kinematics is a study of motion without considering the forces associated with that motion and it has been classified further into forward kinematics and inverse kinematics where forward kinematics the input is joint angles and output is the end effector position and orientation. And the inverse kinematics the input is end effector position and orientation and the output is joint angles.

Similarly, dynamics is the study of motion considering the forces associated with that motion. It has been further classified into inverse dynamics and forward dynamics where the inverse dynamics the input is joint trajectories theta, theta dot and theta double dot and the output is the joint torque tau. And the forward dynamics it is the input is joint torque and output is joint trajectories. Forward dynamics is used for simulation of the manipulator and inverse dynamic system meant are used for controlling the manipulator.

(Refer Slide Time: 05:59)



Now, coming to the control based on the model of the manipulator, thus model based control first we discuss about kinematic model based control. It is nothing, but closed loop inverse kinematic scheme. Because we ensure that the error goes to 0 as the time tends to infinity by a first order dynamic equation.

So, see a like a closed loop inverse kinematics algorithm is derived in such a way that, the control law is given by $\dot{q}$ which is the generalized coordinate of the joint is given by q, which can take either θ for prismatic joint or d for θ for revolute joint and d for prismatic joint the joint variables.

So, generalized to joint coordinate can be represented by the letter q. So, the joint velocity is represented by $\dot{q}$ and the control law under this CLIK is given by $\dot{q} = J^{-1}$ which is a function of q multiplied by $\dot{x}_d + K_e$. So, this takes care of the error that is associated with the Cartesian trajectory desired trajectory actual trajectory.

Well in the inverse kinematics desire trajectory is given in the Cartesian space for the tip of the manipulator and the actual tip should follow that trajectory by the computation of the joint angles. That computation of joint angles is obtained by this control law which takes care of this error getting minimized, this discrepancy getting minimized is ensured by this control law.

So, the control law how we can derive this, which states that first order dynamic aerodynamic equation. So, the error is given by the discrepancy between the desired trajectory and the actual trajectory. Where the actual trajectory is given by the forward kinematics function of q that is what is represented as $k(q)$ here which is nothing, but the end effector position.

So, error is the difference between the desired trajectory and the actual trajectory and the error dot is the time derivative of the error is $\dot{e} = \dot{x}_d - \dot{x}$. Now, $\dot{x}$ is represented by $\dot{x} = J\dot{q}$ we know that from this is forward kinematics the differential kinematics is $\dot{x} = J\dot{q}$.

That is what this may push put here and from this we have this $\dot{q}$ given by this control law also $\dot{q}$ is given by this control law. So, this, when we equate $\dot{q}$ is given by

$$\dot{q} = J^{-1}(\dot{x}_d - \dot{x})$$

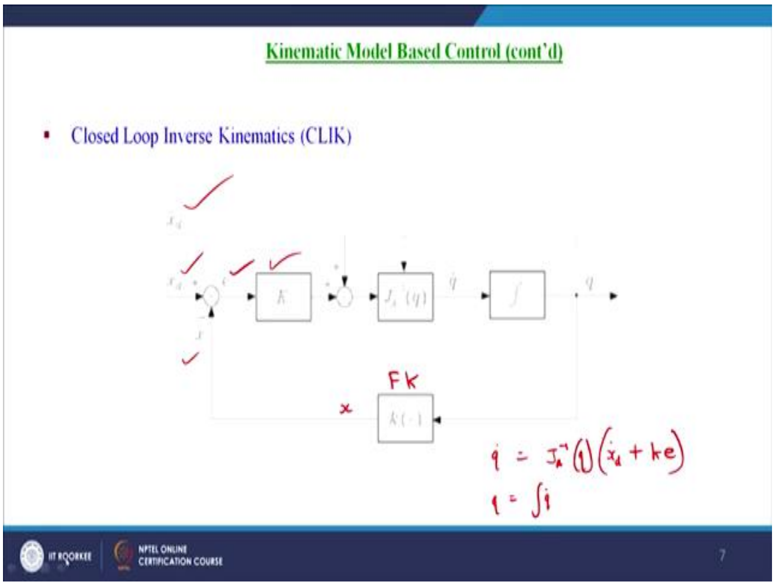
So, these two equations equation 1 and see equation 2 both corresponds to both correspond to $\dot{q}$.

So, when we equate this we will have $\dot{e}$ which is we will have the

$$\dot{x}_d - \dot{x} + k_e = 0$$

which is given by $\dot{e} + k_e = 0$ that is what this first order dynamic equation.

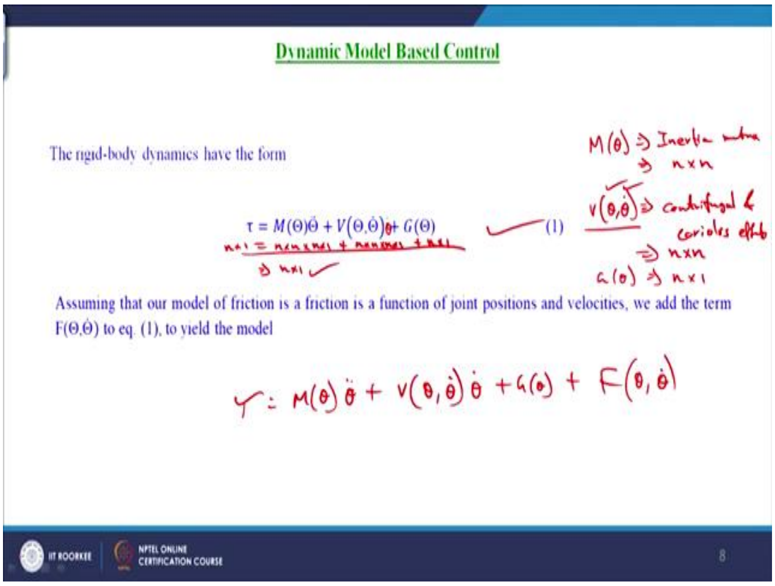
(Refer Slide Time: 10:30)



Now, coming to the block diagram for that CLIK Closed Loop Inverse Kinematics. The closed loop inverse kinematics block diagram is given in such a way that given the desired trajectory of the Cartesian space for the end effector tip, we have $\dot{x}_d$ x_d that gets into the control gain, the error between the desired trajectory and the actual trajectory and that is summarized here so, that it is finally, $\dot{q} = J_A^{+}$.

So, g^{-1} this one into $\dot{x}_d + k_e$ And then q is obtained by the numerical integration of the joint velocity and this portion is nothing, but the forward kinematics given the joint angle obtain the actual tip trajectory, that is compared to get the error and this iteration process continues till we reach the final trajectory.

(Refer Slide Time: 11:56)



Now, coming to the dynamic model based control scheme. The rigid body dynamics is given by

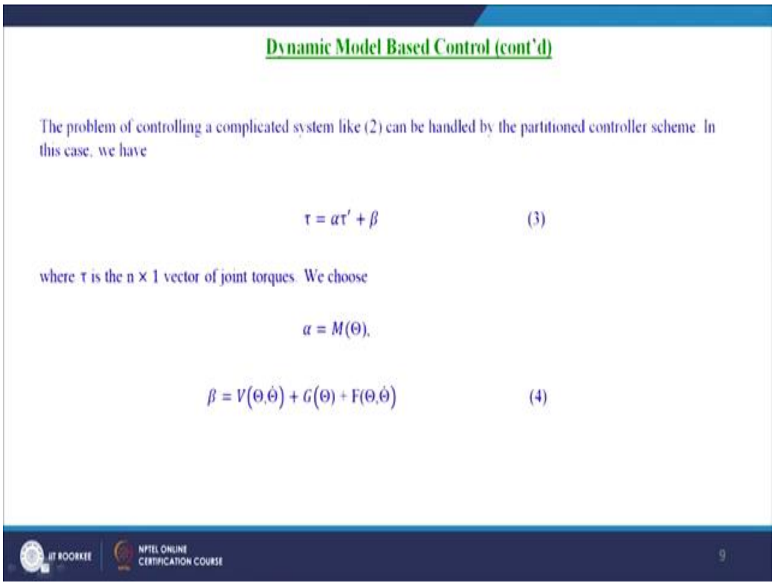
$$\tau = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta})\dot{\theta} + G(\theta)$$

where $M(\theta)$ is called inertia matrix, which is of size $n \times n$.

Where n is the degrees of freedom of the manipulator and $V(\theta, \dot{\theta})$ is given by it is called as the matrix which involves the centrifugal and Coriolis effects. Centrifugal and Coriolis effects this is also of size $n \times n$.

The initial matrix depends on only the joint angle not the joint velocity whereas, this matrix which covers the centrifugal and Coriolis effects the one which depends on both joint angle as well as the joint velocity and finally, $G(\theta)$ is the gravity vector of size $n \times 1$. Thus the left hand side will be $n \times 1$ which is equal to $n \times n$ multiplied by $(n \times 1) + n \times n$ plus $n \times n$ multiplied by $n \times 1$ plus $n \times 1$ eventually gives $n \times 1$ which is the joint torque vector of size $n \times 1$.

(Refer Slide Time: 14:17)



Assuming that our model of friction is a function of joint positions and velocity, we add that term which is the $F(\theta, \dot{\theta})$ two equation 1 so, that the model we obtain is will be

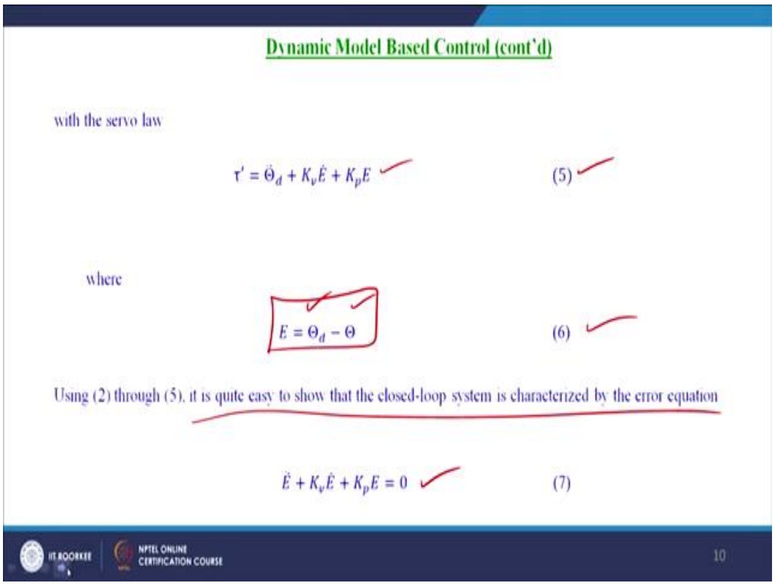
$$\tau = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta})\dot{\theta} + G(\theta) + F(\theta, \dot{\theta})$$

So, the problem of controlling such a complicated system given by say equation 2 can be handled by partition control scheme. So, the partition control scheme can be modeled in such a way that

$$\tau = \alpha\tau' + \beta \qquad (3)$$

Where τ is the $n \times 1$or $1 \times n \times 1$ vector of joint torques and here we choose from this model we choose, α equal to inertia matrix $M(\theta)$ and β is the rest of that dynamic equation which is $V(\theta, \dot{\theta})\dot{\theta} + G(\theta) + F(\theta, \dot{\theta})$.

(Refer Slide Time: 15:33)



With the servo law, we have the servo law considered so, that the feedback from both velocity of the joint trajectory also the position of the joint trajectory error is considered in the servo law. So, that τ' is servo law given by

$$\tau' = \ddot{\theta}_d + K_v\dot{E} + K_pE.$$

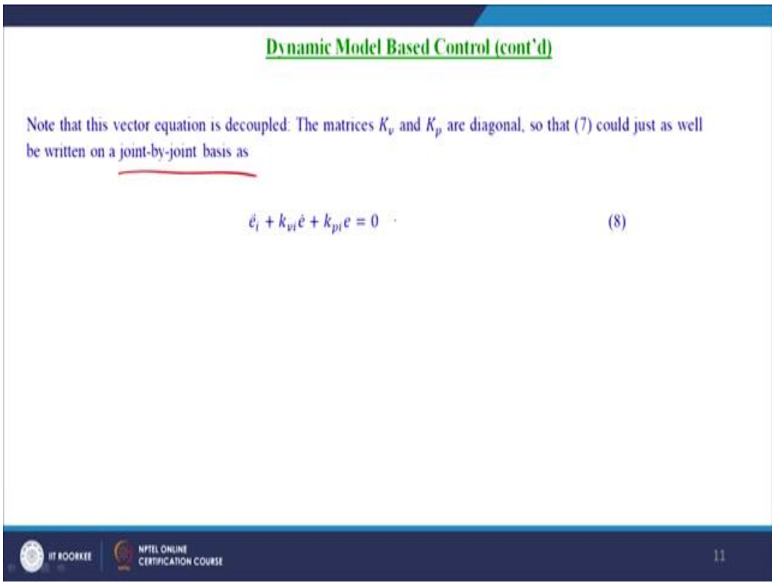
Where the error E is given by the discrepancy between the decide joint angle and the actual joint angle.

So, using the equations from 2 till 6 or till 5, it is quite easy to show that the closed loop system is characterized by the error equation. Second order error equation which is given by

$$\ddot{E} + K_v\dot{E} + K_pE = 0 \qquad (7)$$

So, this is the second order error dynamic equation which shows that as a time tends to infinity the error tends to 0.

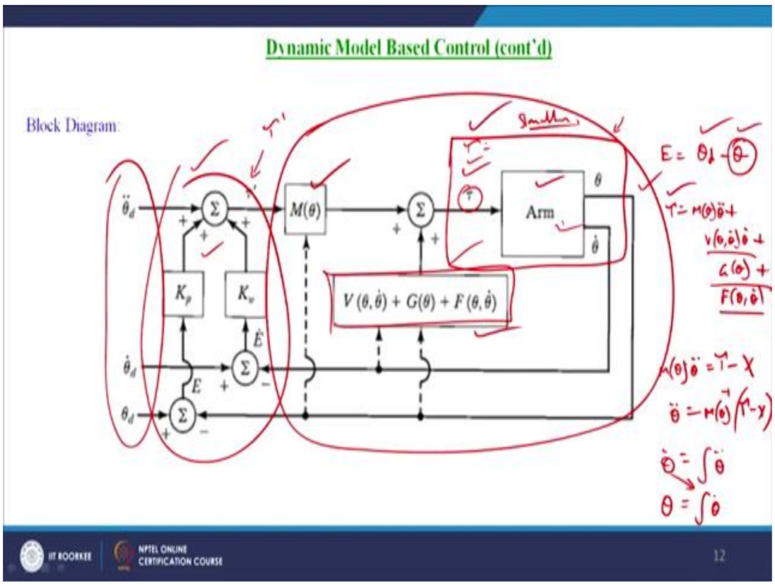
(Refer Slide Time: 16:48)



Note this the matrices K_v and K_p are diagonal. So, for a joint by joint basis for the joint i that aerodynamic equation can be written as

$$\ddot{e}_i + K_{v1}\dot{e} + K_{p1}e = 0 \qquad (8)$$

(Refer Slide Time: 17:12)



Now, coming to the block diagram, explaining or schematically representing the dynamic model based control scheme is shown here, where we give the joint trajectories as the input and we are computing the joint torque, based on the servo part here based on the servo law that is given by tau dash ok.

So, that the servo law is here which takes care of the actual position of the robotic system and this is the portion which is the model based portion. So, we have the dynamic model here when we add this initial matrix with this rest of this portion, we get the tau which is basically the dynamic model of the system.

Thus this portion of the block diagram is called model portion and this portion is called servo portion. Thus it has been partitioned into servo law on the model based law because the dynamic model is getting involved here. So, now the input here is the joint trajectories and the output here is the joint torque.

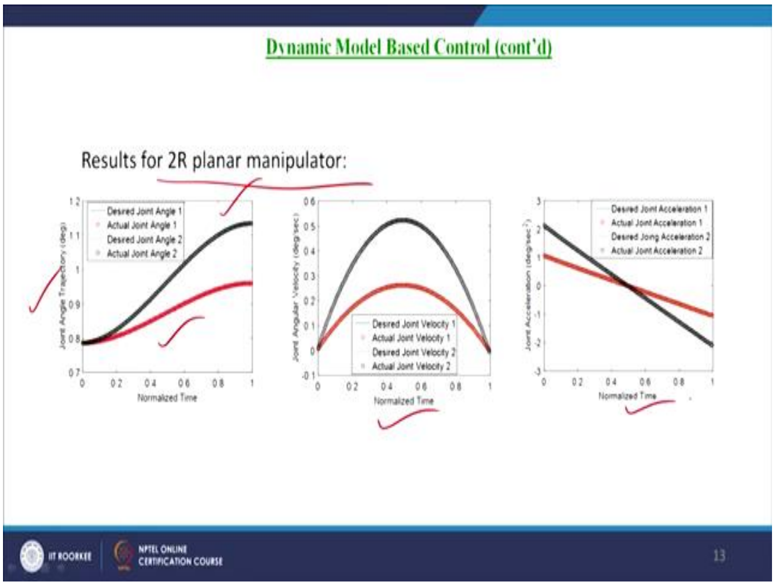
This is the control part and giving the joint torque to the robotic arm, how we get the joint angle, joint velocity. Because we need to have the error coming out from the servo law which will be happening because of the discrepancy between the desired trajectory and the actual trajectory.

So, we need to know the actual trajectory that is done by this portion, which is here the simulation is happening which is giving the joint torque to the our robot arm, we are getting the joint velocity that is

$$\tau = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta})\dot{\theta} + G(\theta) + F(\theta, \dot{\theta})$$

So, τ we know because we are giving that as the input. So, theta $M(\theta)\ddot{\theta}$ is equal to τ minus see the rest of the portion is say x and $\ddot{\theta}$ equal to $M^{-1}(\theta)[\tau - x]$. So, $\dot{\theta}$ will be obtained now by integration of this acceleration further θ, which is joint angle will be obtained by the integration of joint velocity. So, this is how we obtain the joint actual trajectories from the manipulator given the joint torque.

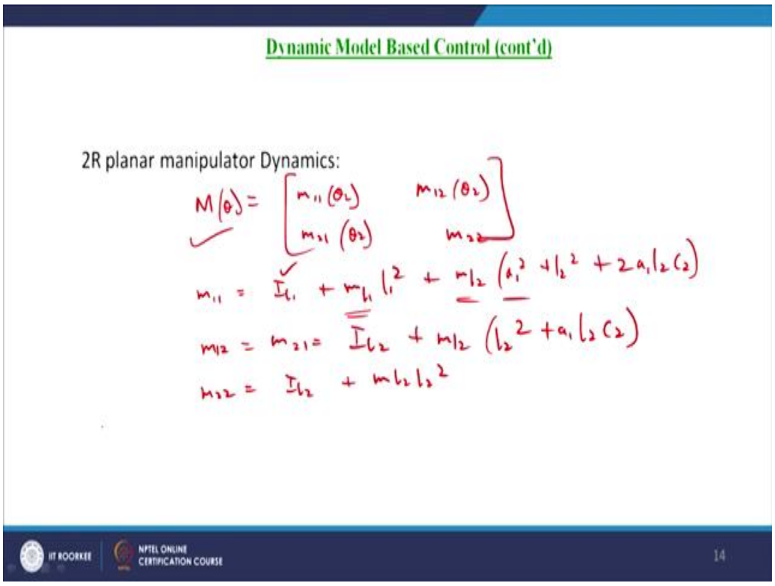
(Refer Slide Time: 20:49)



And we have simulated this for a two link revolute planar manipulator with this model based control scheme, for the given decide joint trajectories of the two joints. So, we have from the controller, we have obtained that the joint trajectories decide joint trajectories have been exactly matched by the control scheme involving the dynamic model.

So, the figure one shows the joint trajectory matching joint angular trajectory matching and the figure two shows joint velocity trajectory matching and finally, the figure three shows the joint acceleration trajectory matching and let me explain you what are all the parameters we have used for the to our planar manipulator dynamic.

(Refer Slide Time: 21:31)



So, that you can also simulate that where we have taken the inertia matrix

$$M(\theta) = \begin{bmatrix} M_{11}(\theta_2) & M_{12}(\theta_2) \\ M_{21}(\theta_2) & M_{22} \end{bmatrix}$$

Where M_{11} is given by which is a function of θ_2 only not θ_1 is given by

$$M_{11} = I_{l1} + M_{l1}l_1^2 + M_{l2}(e_1^2 + l_2^2 + 2a_1l_2cos\theta_2)$$

Where I_{l1} and I_{l2} are nothing, but the moment of inertia relative to the center of mass of links l_1 and l_2 . I repeat I_{l1} and I_{l2} are the moment of inertia related to the center of mass of the links l_1 and l_2 . The M_{l1} is the mass of link l_1 and M_{l2} is the mass of link l_2 ; a_1 is the length of linking l_1 and a_2 is the length of link l_2 and l_1 l_2 nothing, but the distance from the joints respective joints joint 1 and 2 to the center of masses of those two links.

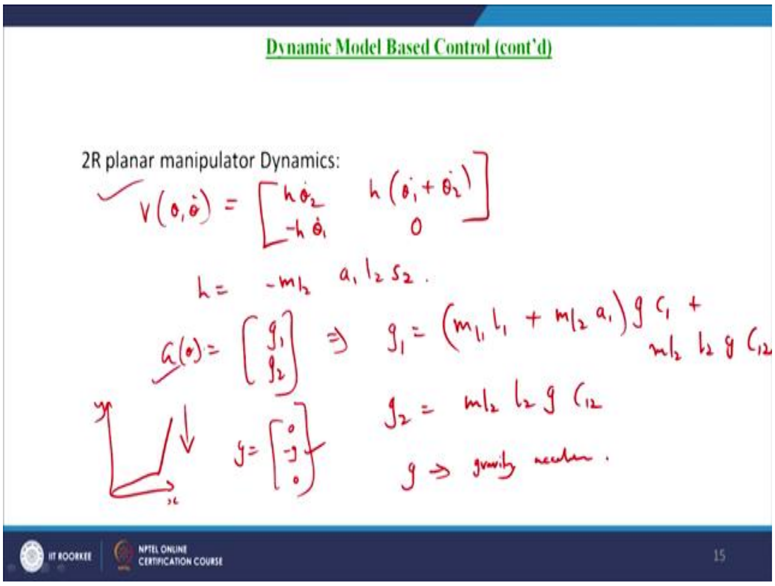
Now,

$$M_{12} = M_{12} = I_{l2} + M_{l2}(l_2^2 + a_1l_2cos\theta_2),$$

c_2 is nothing, but cos of θ_2 and

$$M_{22} = I_{l2} + M_{l2}l_2^2.$$

(Refer Slide Time: 23:53)



So, the $M(\theta)$ part that the inertia matrix part is over, now coming to the matrix $v(\theta, \dot{\theta})$ which is nothing, but the matrix which involves the Coriolis and centrifugal efforts which is given by

$$v(\theta, \dot{\theta}) = \begin{bmatrix} h\dot{\theta}_2 & h(\dot{\theta}_1 + \dot{\theta}_2 \\ -h\dot{\theta}_1 & 0 \end{bmatrix}$$

where $h = -ml_2 a_1 sin\theta_2$.

Finally, the gravity term $G(\theta) = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}$

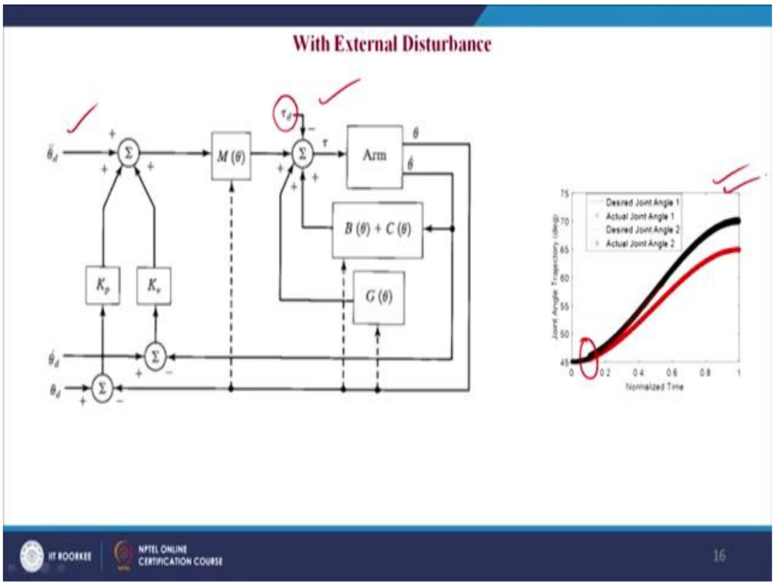
because of size n $\times$ 1 2 $\times$ 1 for the two link manipulator where $g_1 = (m_{l1}l_1 + m_{l2}l_2)gc_1 + m_{l1}l_2gc_{12}$. And $g_2 = m_{l1}l_2gc_{12}$ where g is nothing, but gravity acceleration gravity acceleration and in the planar manipulator which is x and y the vector

$$g = \begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix}$$

Why because now this is the gravity direction which is the $-y$ that is why we put 0 for x 0 for z and for y it $-g$ term that is all.

So, with these values of the matrix and the gravity term, I could able to simulate that to get this trajectory so, that you can also try with this block diagram so, that given the desired trajectory, you can obtain the control law which is given by τ' which is basically the combination of the servo law and the model based law so, that we can able to obtain the joint torque τ, then we simulate the system by this methodology to get the actual joint trajectories.

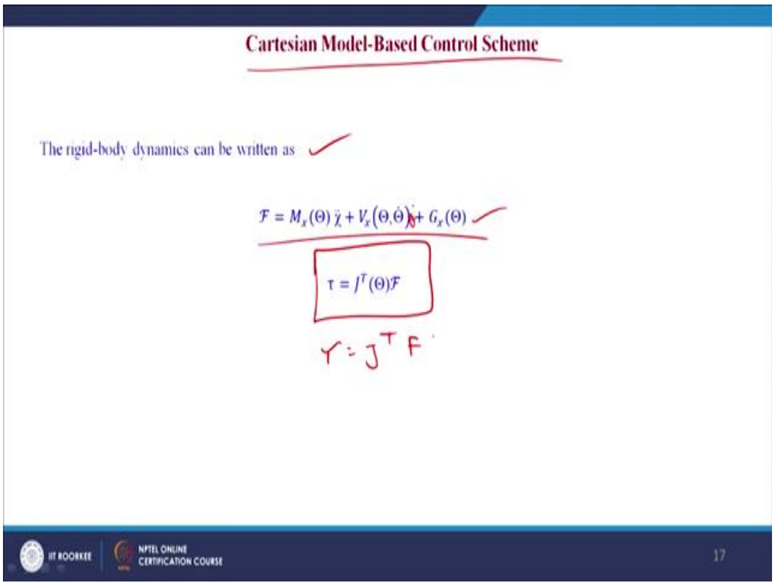
(Refer Slide Time: 26:17)



Now, coming to the concept with the disturbance, the same concept with the disturbance what happens with the disturbance. When we include disturbance τ_d as one of the inputs here along with the system input joint trajectories, we will have with the same scheme, we will have the disturbance getting compensated in such a way that the disturbance can in our case we have used that as a constant almost in the beginning of the trajectories and that is getting compressed so, that the trajectory is maintained in order to track the given decided trajectory.

This is the scheme we have used by including the constant disturbance a constant value disturbance for a while and then we have obtained the desired trajectory matching with the actual trajectory perfectly as can be observed in the plot.

(Refer Slide Time: 27:22)



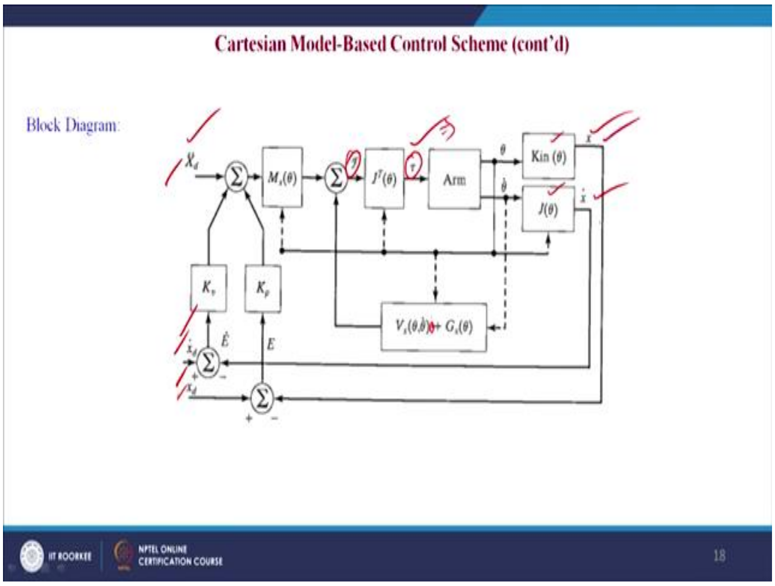
Then we come to Cartesian model based control scheme. So, far we have seen joint space model based control scheme and this is what Cartesian model based control scheme. So, the rigid body dynamics can be written as

$$F = M_x(\theta)\ddot{x} + V_x\big(\theta,\dot{\theta}\big)\dot{\theta} + G_x(\theta)$$

F equal to M x of theta into x double dot plus V x of theta comma theta dot into theta dot plus G x of theta. With the joint torque computation or the control law here is

$$\tau = J^T F$$

(Refer Slide Time: 28:13)



So, the block diagram associated with that Cartesian model based control scheme is given here where we give the end effector trajectories the tip trajectories in terms of position velocity and acceleration. We could be able to obtain the joint torque or the control law which is,

$$\tau = J^T F$$

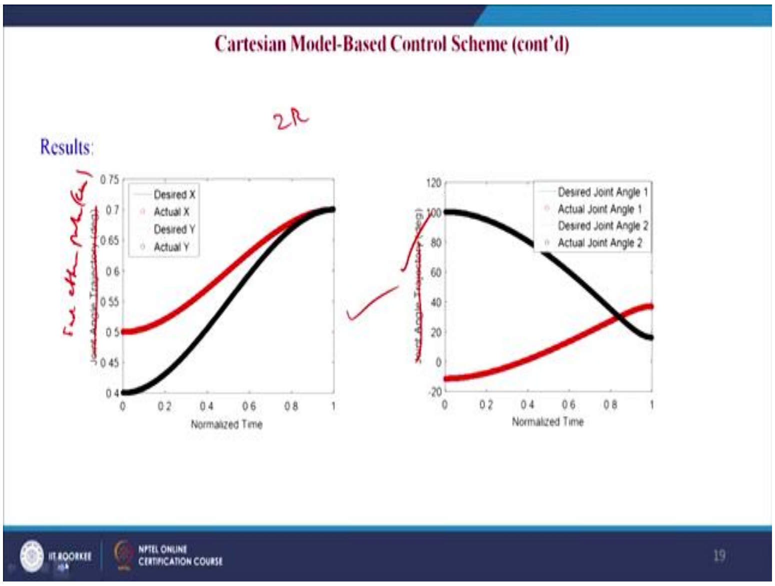
where

$$F = M_x(\theta)\ddot{x} + V_x(\theta,\dot{\theta})\dot{\theta} + G_x(\theta)$$

In the same way simulation is done with the dynamic model and from the joint angle obtained from the simulation, we could be able to do the forward kinematics and the differential kinematics to obtain the actual position and the actual velocity of the end effector tip, that has been fed back through the servo law in order to obtain the actual in order to obtain the error.

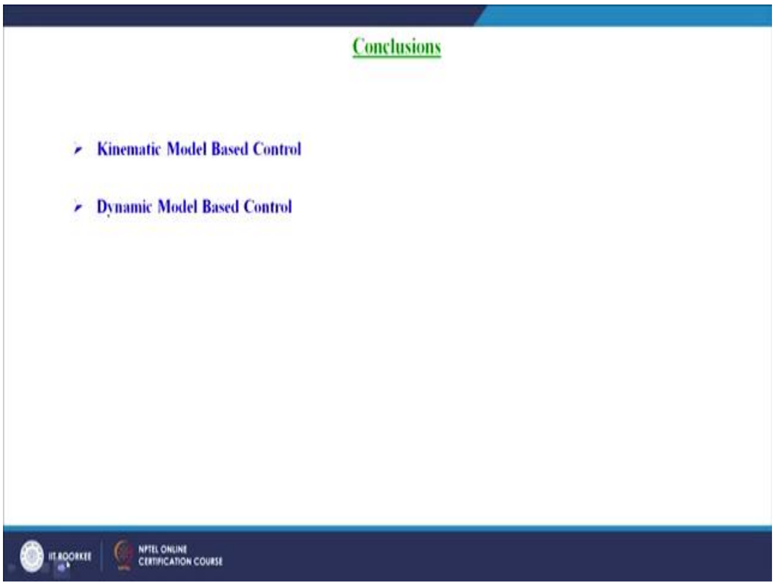
So, that we could able to come to the joint torque, $\tau = J^T F$ we could be able to get the actual joint torque and from that we could be to obtain the joint angle through forward kinematics and inverse kinematics, we could be able to forward kinematics and differential kinematics we could be able to get the position and velocity of the manipulator.

(Refer Slide Time: 29:48)



And similarly for the two planar manipulator case, we could able to obtain the decide trajectory. So, this is basically end effector position which is in centimeter and this is also the joint angle trajectories this is correct this joint angle trajectories.

(Refer Slide Time: 30:22)



Now, coming to the final conclusion part. So, we just finished with this Cartesian based control and now we conclude this session, in such a way that in this lecture we have simulated both the kinematic model based control scheme and the dynamic based control scheme.

We have in this lecture we have seen the basic definitions of kinematics, dynamics and the types of kinematics, the classification of kinematics and the classification of dynamics in this lecture and then we entered into the model based control scheme with which is further classified into kinematics model based control scheme and dynamics model based control scheme.

In the kinematics model based control scheme we have seen the CLIK that is Closed Loop Inverse Kinematics showing the aerodynamic equation which is the first order aerodynamic equation whereas, for the dynamics based control scheme we have seen the inverse dynamics based control scheme which is the model dynamic model based control scheme which is involving partition control scheme and also we have done that through joint space control and Cartesian space control.

Thus, in this lecture we have seen the control scheme of manipulate is based on the model based control, the model based control has been classified into kinematics model based control scheme and dynamics model based control scheme. The kinematics model based control scheme what we have seen in our lecture is closed loop inverse kinematics that is called CLIK closed control scheme, where we have proved that through the first order aerodynamic equation which issues that the error converges to 0 as a time tends to infinity.

Then in the dynamics model based control scheme we have seen partition control scheme involving both joint space control scheme as well as the Cartesian space control scheme. And in the beginning of the lecture we have briefed the definition of kinematics and dynamics and also the classification, that is the further broad classification of kinematics which is inverse dynamics and inverse kinematics and forward kinematics and the classification of dynamics which is inverse dynamics and forward dynamics.

Thank you so much.