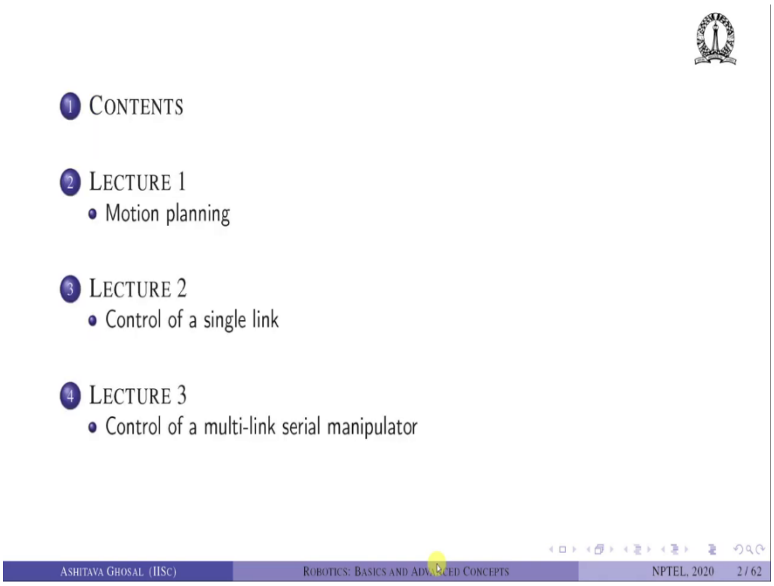
Robotics: Basics and Selected Advanced Concepts
Prof. Ashitava Ghosal
Department of Mechanical Engineering
Indian Institute of Science, Bengaluru

Lecture - 31
Motion Planning

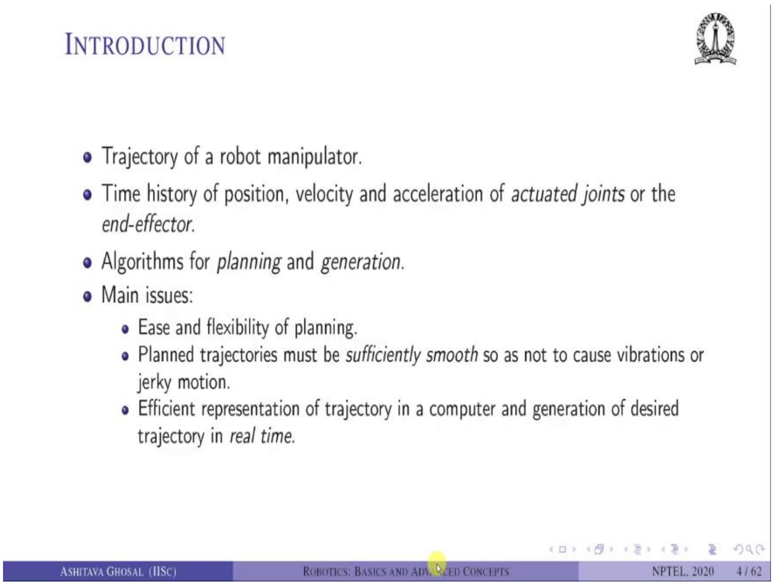
Welcome to this NPTEL lectures on Basics and Advanced Concepts. In this week, we will look at Motion Planning and Linear Control of Robotic Manipulators.

(Refer Slide Time: 00:31)



So, there will be 3 lectures this week, the 1st lecture will be on motion planning, the 2nd lecture will be on control of a single link using linear control. The 3rd lecture will be control of multi-link serial manipulators again using linear control.

(Refer Slide Time: 00:52)

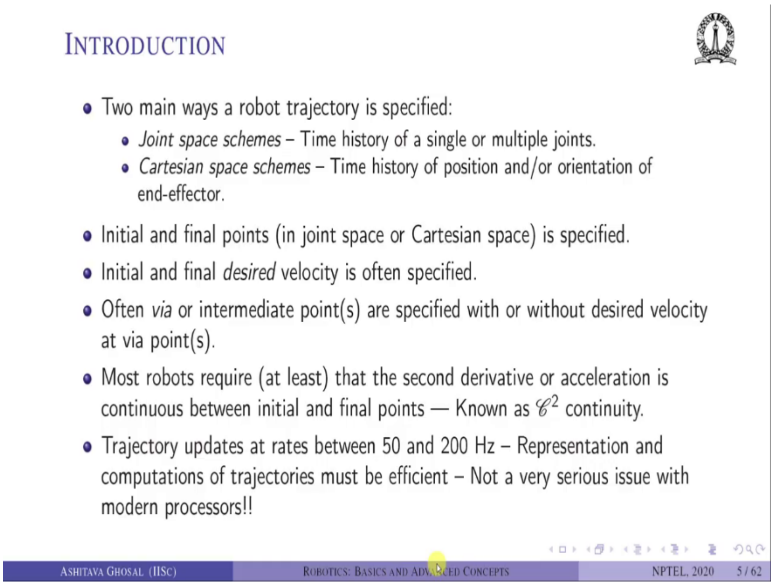


So, 1st lecture is on motion planning, let us start. So, what do we mean by motion planning? We need to specify the trajectory of a robot manipulator. And what do we mean by specify the trajectory of a robot manipulator? Basically, we need to specify the time history of position, velocity, and acceleration of actuated joints or the end-effector, ok.

So, it could be either in terms of the θ which is the joint variable or it could be in terms of the end-effector position and orientation. So, what do we need? We need to develop algorithms for planning and generation of this trajectory. The main issues in motion planning are there should be ease and flexibility in planning, ok.

It should also be that the trajectories which are being planned are sufficiently smooth so as not to cause vibration or jerky motion, ok. Vibration or jerky motion in a robot manipulator will decrease the life of the manipulator. We should also efficiently represent the trajectory in a computer and generate this desired trajectory in real time when required, ok. So, we will look at what is meant by real time little later, ok.

(Refer Slide Time: 02:13)



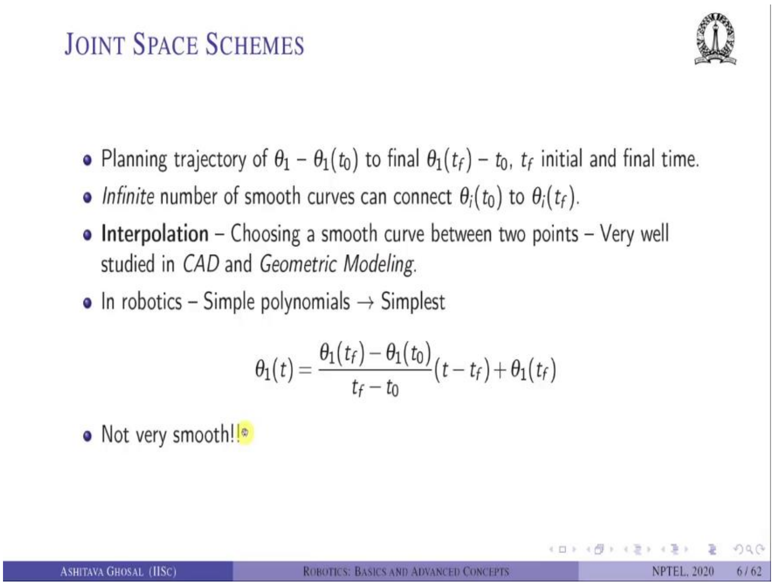
So, there are two main ways a robot trajectory is specified, the first is joint space schemes. So, basically it means the time history of a single or multiple joints ok. In Cartesian space schemes, the time history of position and orientation of the end-effector is specified. And what is typically specified? The initial and final points in joint space or Cartesian space is specified. Often the initial and final desired velocity is also specified.

Additionally, sometimes via or intermediate points are specified. And these come in two varieties, sometimes with the desired velocity at the via points or without the desired velocity at the via point. Most robots require at least that the second derivative or acceleration is continuous between the initial and final points, ok. So, this is known as $\mathcal{C}^2$ continuity.

The trajectory update rates are typically between 50 and 200 Hertz, so between 20 and 5 milliseconds. So, every 20 milliseconds or every 5 milliseconds, we need to generate either the position of the joint θ, $\dot{\theta}$, $\ddot{\theta}$ or the position or an orientation of the end-effector and its derivatives, ok.

So, this representation and computation of trajectories must be efficiently done. So, it is not a very serious issue with modern processors. Modern processors can easily generate the desired trajectories in 5 milliseconds or even faster.

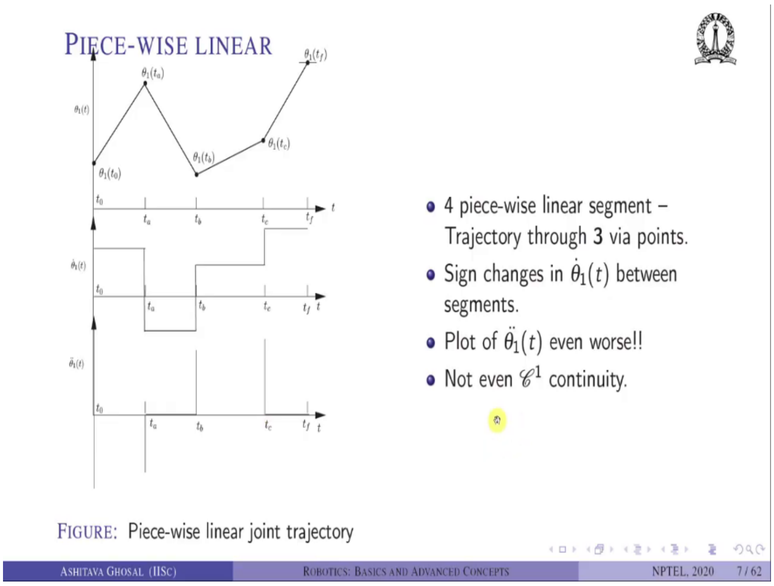
(Refer Slide Time: 04:06)



So, let us continue in joint space schemes, basically they are planned for the trajectory of let us say θ_1. What is given? θ_1 at t_0 and θ_1 at t_f are given, ok. So, where t_0 and t_f are initial and final time. There are infinite number of smooth curves which can connect $\theta_i(t_0)$ to $\theta_i(t_f)$. This topic is called interpolation ok, which is nothing but choosing the smooth curve between two given points.

This is very well studied in CAD and geometric modeling. In robotics, we will look at the simplest possible polynomials. And what is the simplest possible polynomial? It is a linear polynomial between two given points. So, $\theta_1(t) = \frac{\theta_1(t_f) - \theta_1(t_0)}{t_f - t_0}(t - t_f) + \theta_1(t_f)$. So, this is this quantity is like the slope of the straight line connecting $\theta_1(t_f)$ and $\theta_1(t_0)$, ok. This is not very smooth as we can as we will see in the next slide.

(Refer Slide Time: 05:31)



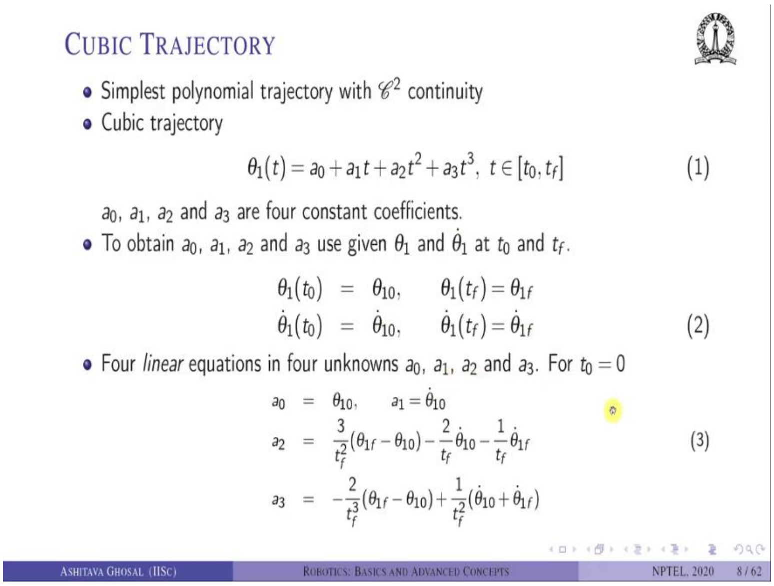
So, in this picture we are showing $\theta_1(t)$ along the y axis and time which is along the x axis. So, let us say the joint initially is at $\theta_1(t_0)$, it will go to $\theta_1(t_a)$, then it comes to $\theta_1(t_b)$, then $\theta_1(t_c)$. So, at 3 time instance t_a, t_b, t_c these are the intermediate θ which are given, ok. And we end up at the final time.

So, if you do linear interpolation or if you connect $\theta_1(t_0)$ to $\theta_1(t_a)$ by a straight line and likewise from t_a to t_b also via a straight line, so each of these are piece wise linear segments ok, through these 3 via points, then the derivative of this straight line is a constant value. So, $\dot{\theta}_1$ between t_0 and t_a will be some value.

Now, when the straight line changes direction the velocity will become from positive to negative. Likewise, again it changes direction again the velocity will go from negative to positive. And if it changes slope again then it will have a different positive value. So, basically there are sign changes in $\dot{\theta}_1$ between segments, ok. And there are distinct changes in the value of $\dot{\theta}_1$ even if there is no sign change.

The plot of $\ddot{\theta}_1$ is even worse. Why? Because the derivative of this so called step function, so when it is constant it is 0. And then when it changes sign there is a change in; so, it is like a delta function which is happening at t_a, t_b, and t_c, ok. So, it is not even $\mathcal{C}^1$ continuity, ok.

(Refer Slide Time: 07:28)

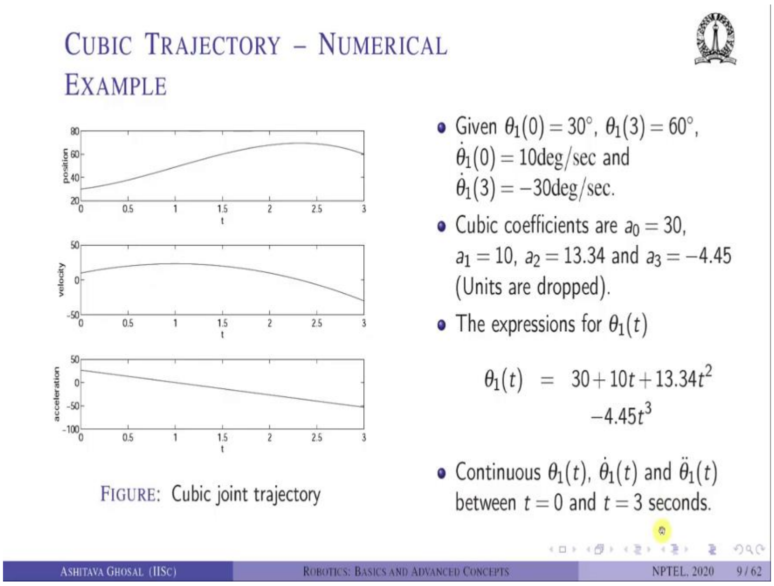


To obtain a simplest polynomial trajectory with $\mathcal{C}^2$ continuity, we can look at a cubic trajectory. So, a cubic trajectory is given by $\theta_1(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$, where t lies between t_0 and t_f. And this a_0, a_1, a_2, and a_3 are four constant coefficients which needs to be determined.

So, to obtain this four constant coefficients we use the fact that θ_1 is given at t_0 and t_f, similarly $\dot{\theta}_1$ is given at t_0 and t_f. So, $\theta_1(t_0) = \theta_{10}$, $\theta_1(t_f) = \theta_{1f}$, $\dot{\theta}_1(t_0) = \dot{\theta}_{10}$, and $\dot{\theta}_1(t_f) = \dot{\theta}_{1f}$. So, if you substitute these 4 given position and velocities at t_0 and t_f back into this cubic equation, we will get four linear equations in four unknowns, ok, a_0, a_1, a_2, and a_3.

So, for the sake of simplicity, let us assume that $t_0 = 0$, so we start at 0 time. Then, if you substitute $t = 0$ in this equation we will see that $\theta_{10} = a_0$. Likewise, $a_1 = \dot{\theta}_{10}$, and a_2 and a_3 will be in terms of t_f, and θ_{1f}, and θ_{10}, and so on, ok. Little complicated expressions for a_2 and a_3, but not very complicated, ok.

(Refer Slide Time: 09:25)

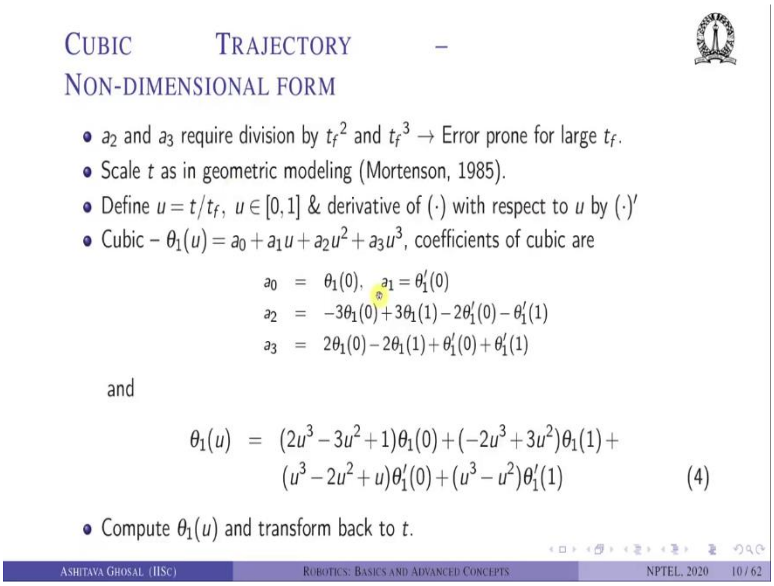


So, as you can see a_2 is divided one term is divided by t_f^2. In a_3, one term is divided by t_f^3, ok. So, as an example let us consider that $\theta_1(t = 0)$ is 30 degrees, $\theta_1(t = 3)$ is 60 degrees, $\dot{\theta}_1(t = 0)$ is 10 degrees per second and $\dot{\theta}_1(t = 3)$ is -30 degrees per second.

So, we substitute this 4 given quantities in the cubic equation and we can show or derive that $a_0 = 30$, $a_1 = 10$, $a_2 = 13.34$, and $a_3 = -4.45$. So, the units are dropped. So, the expression for $\theta_1(t) = 30 + 10t + 13.34t^2 - 4.45t^3$, which we can plot. So, this is a plot of θ_1 as a function of time between 0 and 3, ok. So, this is that cubic curve which we have obtained.

The derivative of the cubic curve is a quadratic, ok. So, this is the quadratic curve which shows $\dot{\theta}_1$ as a function of time. And the second derivative of that cubic curve is linear. So, that is the acceleration. So, what you can see is $\theta_1(t)$, $\dot{\theta}_1(t)$, and $\ddot{\theta}_1(t)$ are all continuous between $t = 0$ and $t = 3$ seconds, ok.

(Refer Slide Time: 10:55)



Let us continue. You can see that this a_2 and a_3 requires division by t_f^2 and t_f^3. So, for example, if t_f is very large let us say it is 5 minutes, and you are going to generate Δt at let us say 10 milliseconds, or 20 milliseconds. So then this t_f can become very very large and t_f^2 and t_f^3 can also become very large.

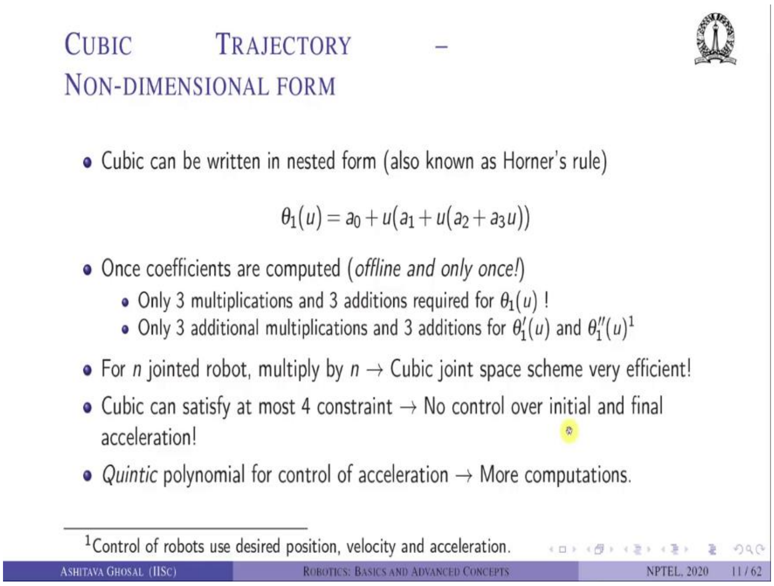
So, we scale t as in geometric modeling that is the solution to this problem and this is given in some geometric molding book by Mortenson. So, we defined a new variable $u = t/t_f$ and u will always lie between 0 and 1, and the derivatives of this cubic is generated by $(\dot{\ })$, but with respect to u we denote it by a prime. So, the cubic now becomes $\theta_1(u) = a_0 + a_1 u + a_2 u^2 + a_3 u^3$.

If you substitute back, all those given quantities which is a t_0 it is θ_{10}, and derivative at $(t = 0)$ is $\theta_1'(0)$, and so on. We can show that $a_0 = \theta_1(0)$, $a_1 = \theta_1'(0)$, $a_2 = -\theta_1(0) + 3\theta_1(1) - 2\theta_1'(0) - \theta_1'(1)$. So, 1 means at $u = 1$. And likewise you can find a_3, ok.

So, basically re-substituting back a_0, a_1, a_2, and a_3 in this cubic we can show that $\theta_1(u) = (2u^3 - 3u^2 + 1)\theta_1(0) + (-2u^3 + 3u^2)\theta_1(1) + (u^3 - 2u^2 + u)\theta_1'(0) + (u^3 - u^2)\theta_1'(1)$.

So, basically as you can see there is no division by a large number, ok. So, we can obtain θ_1 as a function of u which lies between 0 and 1, and $u = t/t_f$, ok. But eventually, we need to supply θ_1 at every time instant. So, we can convert back $\theta_1(u)$ back to $\theta_1(t)$ by using $u = t/t_f$, ok. So, we are not dividing by t_f^2 or t_f^3.

(Refer Slide Time: 13:45)



This cubic which is given here ok, $a_0 + a_1 u + a_2 u^2 + a_3 u^3$, where a's are given by these expressions, can also be written in nested form which is $a_0 + u\big(a_1 + u(a_2 + a_3 u)\big)$. So, this is the nice compact way. What is the advantage of doing this? You can see that it requires only 3 multiplication and 3 additions.

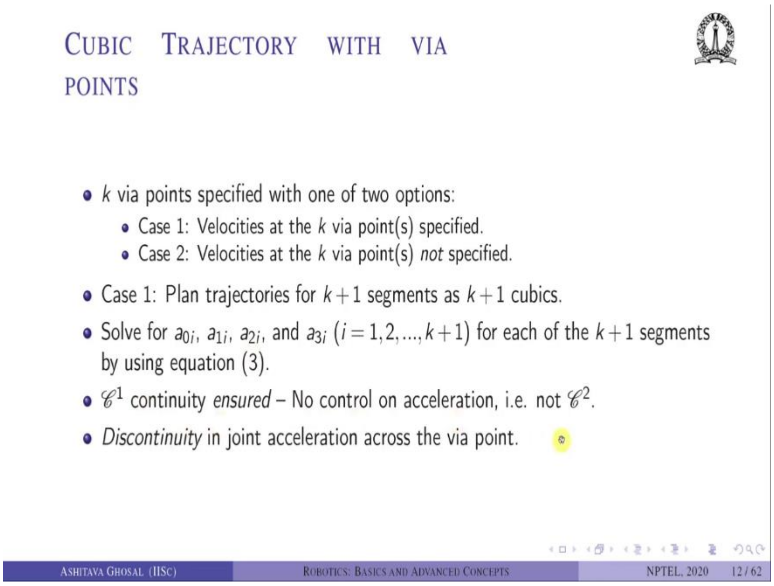
So, when it was in this cubic form it required 1 multiplication here, 3 multiplications here and some u^3 times a_3, so 4 additions and some several multiplications, ok. In this nested form we need only 3 multiplications and 3 additions for to obtain $\theta_1(u)$. We can also find $\theta_1'(u)$ and $\theta_1''(u)$ by just doing multiplication and 3 additions, ok.

So, for a n jointed robot we generate this θ_1, θ_2, θ_3,.., θ_n, separately. So, basically we have n times 3 multiplications and 3 additions to obtain the θ's of each one of the joints. So, in a way this cubic joint space scheme is very efficient, ok. Let us continue. One thing that you can see is the cubic can satisfy at most 4 constraints, ok. So, there are 4 parameters a_0, a_1, a_2, and a_3.

So, there is no control over initial and final acceleration, we can only specify initial and final position, and initial and final velocities. If you also want to specify initial and final positions, we have to use a 5th degree polynomial which is also sometimes called as a quintic polynomial. So, it is not very complicated, but we need a little bit more number of computations, ok.

So, why do we need to specify acceleration? Sometimes in control we will need both position velocity and acceleration. It helps to specify or find out what is the desired velocity and desired acceleration also.

(Refer Slide Time: 16:17)

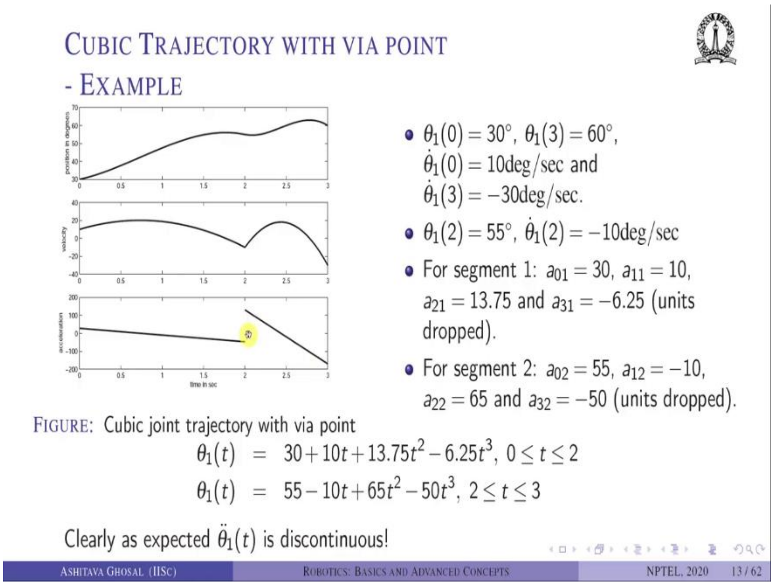


Now, let us continue if you want to generate a cubic trajectory with via points, ok. Let us say there are k via points specified and as I mentioned earlier there are two possible cases, one is the velocity at the k via points are specified and the other case the velocity at the k via points are not specified. Not specified means we do not care, ok.

So, in the case of case 1, then the velocities are specified, so basically we have $k + 1$ segment and we plan $k + 1$ cubic. So, between first initial point and the first via point we plan a cubic, between the first via point and the second via point we plan another cubic and so on. So, for two points and k via points we will have $k + 1$ segments. So, we just plan $k + 1$ cubics.

So, for each one of those we solve for a_{0i}, a_{1i}, a_{2i} and a_{3i} ok, all by using the same cubic equation. So, in this case, we will see later that C^1 continuity is ensured, but since we have no control over the acceleration it is not C^2 continuity. So, there is a discontinuity in joint acceleration across the via point, ok.

(Refer Slide Time: 17:43)



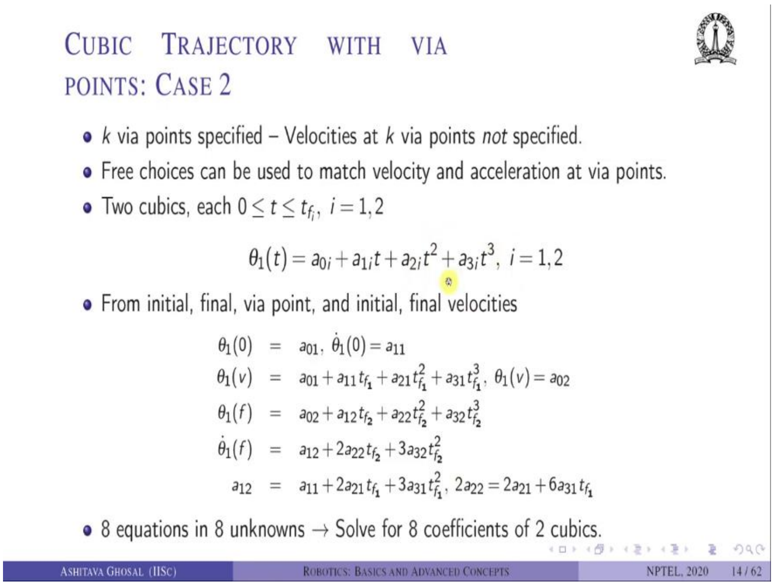
Example. So, we have $\theta_1(0)$ given as 30 degrees, $\theta_1(3)$ given as 60 degrees, $\dot{\theta}_1(0)$ as 10 degrees per second and the $\dot{\theta}_1(3)$ as -30 degrees per second. However, it is also specified that at time $t = 2$, so in between 0 and 3 the angle should be 55 degrees and the velocity should be-10 degrees per second. So, everything is specified.

So, there are two segments for segment 1, $a_{01} = 30$, $a_{11} = 10$, $a_{21} = 13.75$ and $a_{31} = -6.25$. For segment 2, $a_{02} = 55$, $a_{12} = -10$, $a_{22} = 65$ and $a_{32} = -50$, we just solves us to cubic's with these boundary conditions, at $t = 0$ and $t = 2$, and then $t = 2$, and $t = 3$.

So, the equations of the cubics are $\theta_1(t) = 30 + 10t + 13.75t^2 - 6.25t^3$, where t lies between 0 and 2. And likewise, for the 2nd segment $\theta_1(t) = 55 - 10t + 65t^2 - 50t^3$, where t lies between 2 and 3. So, we can plot this and we can see that θ_1 as a function of time consist of one cubic up to 2 and another cubic from 2 to 3.

The velocity when you take the derivatives we will have one quadratic here and another quadratic here. So, you can see that there is a kink sort of thing which is happening. And if you take the second derivative you will have a linear acceleration from 0 to 2, and a different linear acceleration from 2 to 3. So, there is a jump. So, the acceleration is discontinuous.

(Refer Slide Time: 19:51)



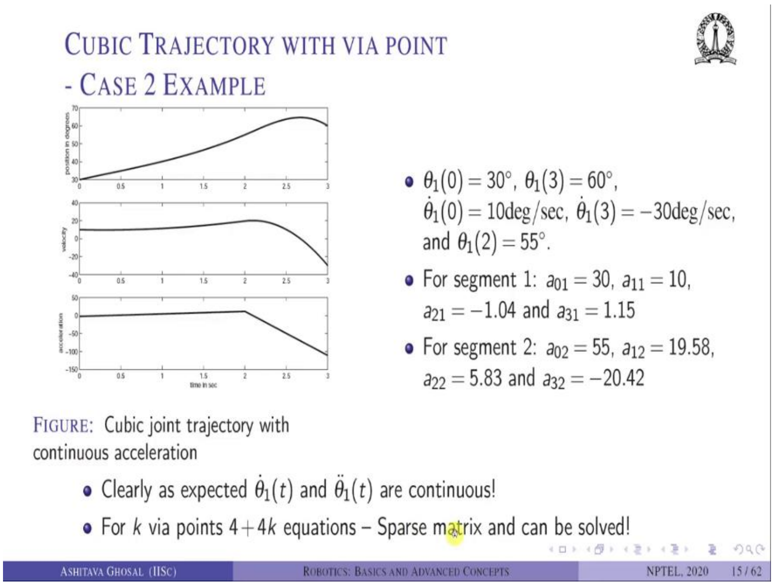
In case 2, if the k via points are specified, but the velocities of the k via points are not specified. Then we are free to choose to match velocities and accelerations at the via points, ok. So, let us start. So, we have two cubics from 0 to t_{f_i}, $i = 1,2$. So, $\theta_1(t) = a_{0i} + a_{1i}t + a_{2i}t^2 + a_{3i}t^3$, $i = 1,2$.

So, from initial final via point and initial final velocity, so there are 5 of these, ok. We can substitute and we can show that $\theta_1(0) = a_{01}$, $\dot{\theta}_1(0) = a_{11}$, $\theta_1(v)$ also we can get, but for the second segment $\theta_1(v) = a_{02}$, ok. So, at the via point the θ_1 at the end of the first segment must be equal to θ_1 at the beginning of the next segment. So, we have actually now 6 constraints. So, we substitute this 6 constraints in these expressions and we can solve, we can get 6 equations.

Now, what we can do is we can match the velocities and acceleration at the via point, ok. So, the velocity is the first derivative of this at $t = t_{f_i}$ will be equal to the start of the next segment. So, we will get one equation which is $a_{12} = a_{11} + 2a_{21}t_{f_1} + 3a_{31}t_{f_1}^2$.

Likewise, we take the second derivative which is $2a_{22} = 2a_{21} + 6a_{31}t_{f_1}$. So, acceleration in the end of the first segment should be equal to the acceleration with the beginning of the second segment. So, now, you can see that there are 8 equation in 8 unknowns which we can solve and obtain all the 8 coefficients of the two parts of the cubic.

(Refer Slide Time: 22:15)

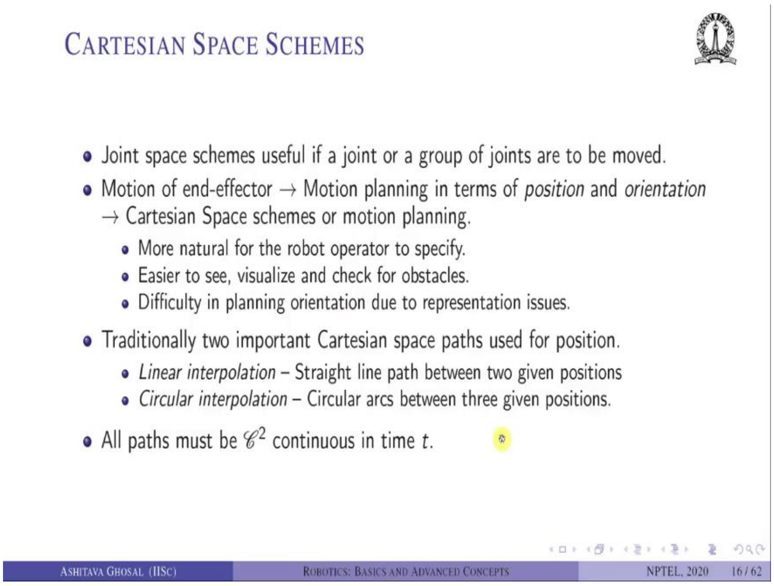


So, if you plot this for the same data, so you can see that for the first segment $a_{01} = 30$, $a_{11} = 10$, $a_{21} = -1.04$, $a_{31} = 1.15$. Likewise, for segment 2, $a_{02} = 55$, $a_{12} = 19.58$, $a_{22} = 5.83$ and $a_{32} = -20.42$. So, this is one cubic till 2, this is another cubic from 2 to 3.

If you take the derivative this is the quadratic up to here and another quadratic up to here. However, there is no kink. Why? Because we have matched the velocities at the via point $t = 2$. Likewise, since there is no kink there is no jump in slope. So, we have acceleration which is continuous which is a straight line from 0 to 2 and another straight line between 2 1 3, but there is no discontinuity at $t = 2$, ok.

So, clearly as expected $\dot{\theta}_1$ and $\ddot{\theta}_1$ are continuous, ok. So, for k via points we can get $4 + 4k$ equations. And it turns out this is the sparse matrix and we can easily solve for all the coefficients if you have k via points, but at the via points we match velocity and accelerations, ok.

(Refer Slide Time: 23:48)



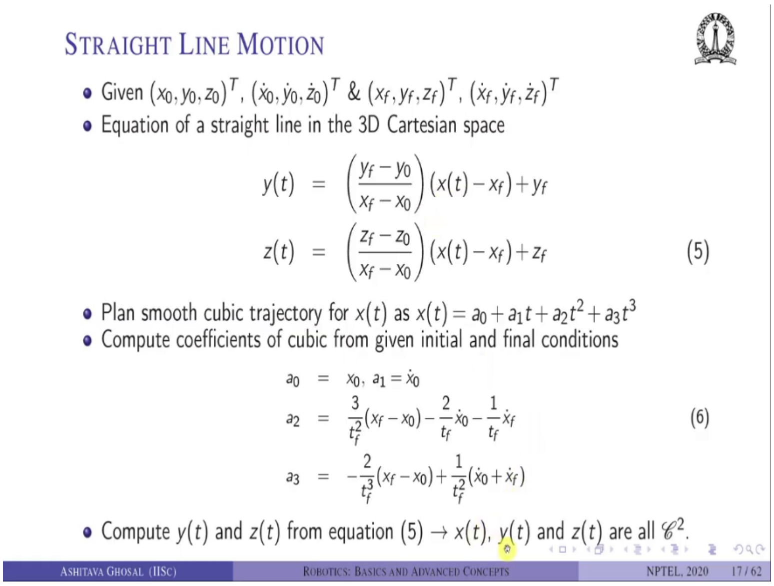
So, the joint space schemes is useful if a joint or a group of joints are to be moved. Typically, we are looking at the motion of the end-effector, ok. So, if you want to do motion planning in terms of position and orientation of the end-effector, then we need to use what are called Cartesian space schemes for motion planning, ok.

So, this is more natural for the robot operator to specify. A robot operator can see the end-effector is moving along the straight line or it is moving along a curve in 3D space ok, which is easier to see and visualize and check for obstacles also. Say, if I say that I want to go from one point in the Cartesian space to another point in the Cartesian space, and if I generate the straight line we can see if this in this straight line it is hitting any object or not, ok.

It is difficult little bit to plan orientation due to the representations of orientation, ok. Remember orientation can be represented using rotation matrices, angle axis ok, Euler angles and so on. So, in all cases there are some small problems. Rotation matrix lots of constraints, Euler angles there are singular configurations. So, traditionally two important Cartesian space parts are used for position planning, ok.

So, one is called linear interpolation which is basically nothing, but straight line path between two given positions. The second thing is called circular interpolation which is we go on a circular path between three given positions, ok. And as in any motion planning scheme all parts must be $\mathcal{C}^2$ continuous in time t.

(Refer Slide Time: 25:53)

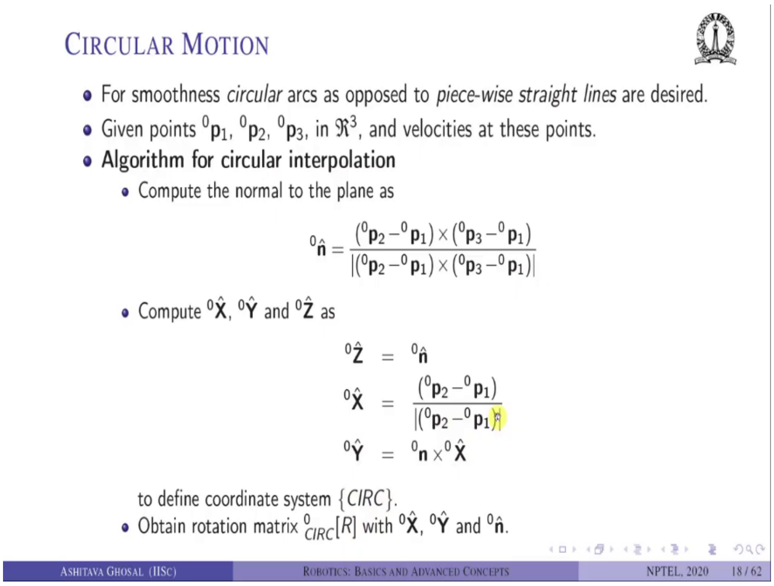


So, straight line motion. How do we plan a straight line motion for the end-effector? So, what are we given? We are given $(x_0, y_0, z_0)^T$ at $t = 0$ initial point initial velocity $(\dot{x}_0, \dot{y}_0, \dot{z}_0)^T$; final point $(x_f, y_f, z_f)^T$ and final velocity $(\dot{x}_f, \dot{y}_f, \dot{z}_f)^T$. So, the equation of a straight line in 3D space can be represented by two equations.

So, it is like $y = mx + c$ and $z = m_2 x + c_2$, ok. In general, we can write as y is some slope times $(x(t) - x_f)$ plus y_f; and $z(t)$, z as a function of time is some slope times $(x(t) - x_f)$ plus z_f. So, we plan a smooth cubic trajectory for $x(t)$ as $x(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$. Compute the coefficients from the given initial and final conditions, $a_0 = x_0$, $a_1 = \dot{x}_0$ and so on.

So, once we have $x(t)$ at the smooth trajectory we can compute $y(t)$ and $z(t)$ from this equation, ok. So, which will ensure that all the generated $x(t), y(t), z(t)$ are all C^2 continuous, right, $x(t)$ is C^2 continuous, so both $z(t)$ and $y(t)$ should be C^2 continuous and more importantly all the $x(t), y(t), z(t)$ are exactly lying on a straight line in 3D space, ok.

(Refer Slide Time: 27:37)



Sometimes for smoothness, we specify circular arcs as opposed to piece-wise straight lines between 3 points. So, if you are given 3 points, I can do two straight lines or I can pass a circular arc. So, circular arc is little preferred, better preferred because it is more smoother, in terms of 3D space, in time we will always be smooth. So, how do we obtain the trajectory?

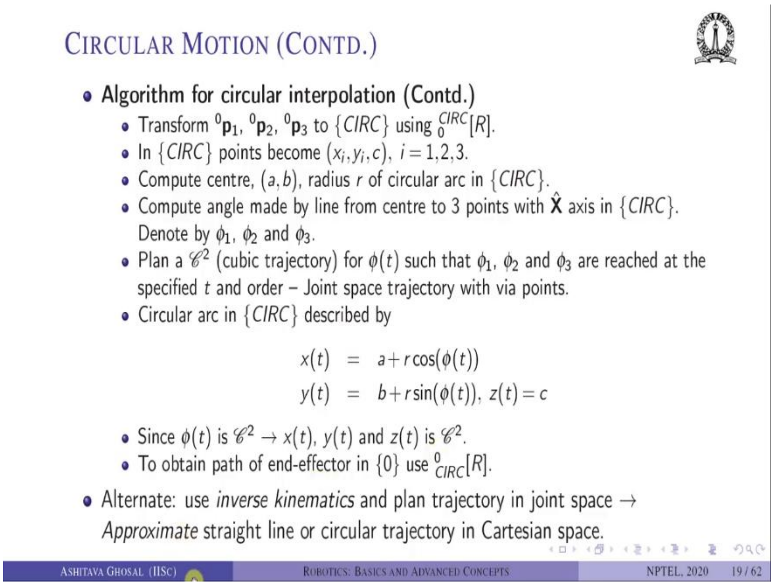
When 3 given points let us say ${}^0\boldsymbol{p}_1$, ${}^0\boldsymbol{p}_2$, ${}^0\boldsymbol{p}_3$, all in some fixed coordinate system are given and the velocities at these points are also given. So, first thing is, we need to define an algorithm for circular interpolation. Meaning what? We find out the equation of the circle, all these 3 points will line in a plane. So, we need to find the equation of the circular arc in this plane.

So, what are the steps? First thing is we compute the normal to this plane. So, there is a vector from ${}^0\boldsymbol{p}_1$ to ${}^0\boldsymbol{p}_2$. Another vector from ${}^0\boldsymbol{p}_1$ to ${}^0\boldsymbol{p}_3$, the normal to this plane is the cross product of these two vectors. We also need to make sure that they are unit vectors. Then, we compute the location of the axis in this plane let us call them ${}^0\widehat{\boldsymbol{X}}$, ${}^0\widehat{\boldsymbol{Y}}$ and ${}^0\widehat{\boldsymbol{Z}}$.

So, the ${}^0\widehat{\boldsymbol{Z}}$ axis is along the normal, the ${}^0\widehat{\boldsymbol{X}}$ axis is between ${}^0\boldsymbol{p}_1$ and ${}^0\boldsymbol{p}_2$, again made into a unit vector and ${}^0\widehat{\boldsymbol{Y}}$ axis is ${}^0\boldsymbol{n} \times {}^0\widehat{\boldsymbol{X}}$. So, right handed coordinate system again. So, these ${}^0\widehat{\boldsymbol{X}}$, ${}^0\widehat{\boldsymbol{Z}}$ and ${}^0\widehat{\boldsymbol{Z}}$ axis will define the coordinate system {CIRC} for denoting circular, ok. So, what is the rotation matrix of this coordinate system {CIRC} with respect to {0}?

The first column is the ${}^{0}\hat{\boldsymbol{X}}$ axis which is this, second column is the ${}^{0}\hat{\boldsymbol{Y}}$ axis which is this, and third column is the ${}^{0}\hat{\boldsymbol{Z}}$ axis which is the normal to the plane.

(Refer Slide Time: 29:50)



So, now we transform this ${}^{0}\boldsymbol{p}_1$, ${}^{0}\boldsymbol{p}_2$, ${}^{0}\boldsymbol{p}_3$ to this coordinate system {CIRC} using a rotation matrix ${}^{CIRC}_{0}[R]$. So, you can see ${}^{CIRC}_{0}[R]\ {}^{0}\boldsymbol{p}_1$ will give me this point p_1 in the {CIRC} coordinate frame, ok. So, in this {CIRC} coordinate frame the points will be (x_i, y_i, c_i). So, the z coordinates will not change ok, because all 3 are in a plane.

So, now, from these points (x_i, y_i, c_i) we can compute (a, b), which is the centre of the circular arc and r which is the radius of the circular arc, ok. How can we do that? There are many ways. So, one is you connect (x_i, y_i): $(x_1, y_1), (x_2, y_2), (x_3, y_3)$, you draw the straight lines find the perpendicular bisector of these two segments wherever they meet, that is the centre of the circular arc. And then radius is from the centre to any one of the points.

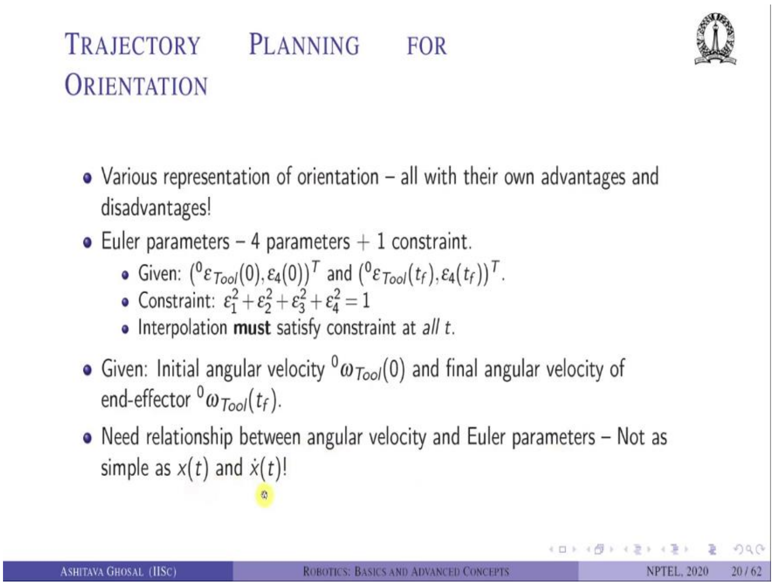
We can also compute the angle made by the line from the centre to the 3 points with the ${}^{0}\hat{\boldsymbol{X}}$ axis in CIRC. So, let us denote them by ϕ_1, ϕ_2, and ϕ_3, ok. So, I have found the centre, I found an ${}^{0}\hat{\boldsymbol{X}}$ axis, and then from 1, 2 and 3 I can find the angle, ok. We can plan as C^2, cubic trajectory for $\phi(t)$ such that ϕ_1, ϕ_2, ϕ_3 are reached in the specified order and in specified time t, ok.

So, maybe at $t = 0$, then $t = t_{f_1}$, and then some $t = t_{f_2}$, ok. So, once we find this circular arc the equation of a points on the circular arc will be given by $x(t) = a + r\cos(\phi(t))$ and $y(t) = b + r\sin(\phi(t))$ and $z(t)$ will be constant, ok. So, since $\phi(t)$ is $\mathcal{C}^2$ continuous which is done using a cubic trajectory $x(t)$, $y(t)$, and $z(t)$ are also $\mathcal{C}^2$.

And then finally, to obtain the end-effector path in the 0th coordinate system, the fixed reference coordinate system we pre multiply $x(t)$, $y(t)$, $z(t)$ by the ${}^{CIRC}_{0}[R]$, ok. So, what have we done? Basically, we have taken these 3 points, find the equation of the centre of the circular arc in some plane, then planed the trajectory in that plane, and then converted it back to the 0th coordinate system.

The alternate would be to use inverse kinematics and plan the trajectory in joint space. So, if I give you $x(t)$, $y(t)$, $z(t)$, we can do inverse kinematics find the joint angles plan, the trajectory in joint space and then convert it back to the $x(t)$, $y(t)$, $z(t)$. However, this will only give you approximate straight line ok or circular trajectory in Cartesian space. Because only at the points where you do the inverse kinematics it will be exact, in between it will not be exact.

(Refer Slide Time: 33:24)



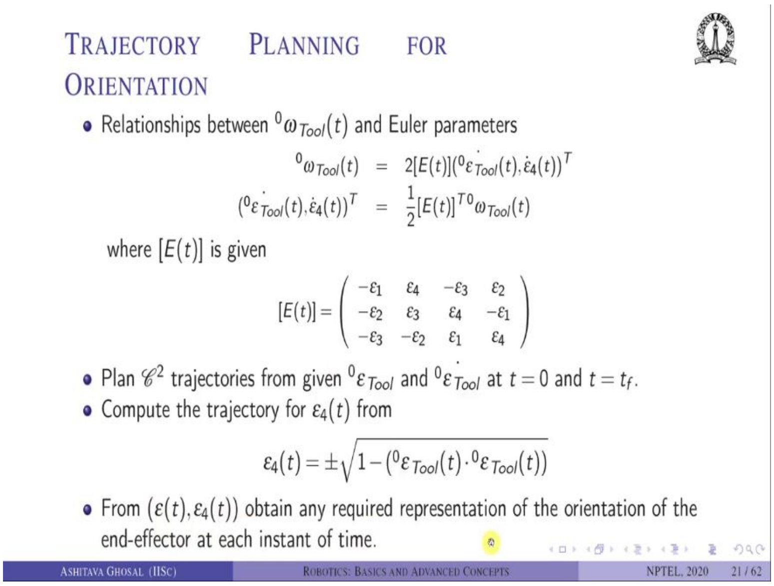
Trajectory planning for orientation. So, as I said there are various representation of orientation, all with their own advantages and disadvantages, ok. When you want to do trajectory planning for orientation it turns out that this representation using Euler parameters is more suitable.

So, Euler parameters are nothing, but 4 parameters plus 1 constraint. So, we have some ε_1, ε_2, ε_3 in the 0^{th} coordinate system and a scalar ε_4. So, what are we given? We are given this vector at $t = 0$, ε_4 also at $t = 0$ and it is likewise this vector at t_f and ε_4 at t_f. So, we have to ensure that the $\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 + \varepsilon_4^2 = 1$.

And this interpolation must satisfy this constraint at all time t. So, what do we do? What is given? We are given some initial and final Euler parameters. We are also given initial and final angular velocity of the end-effector, ok. Remember in position we are given initial and final position, but also the initial and final velocity. In the case of rotation or orientation, we can only specify angular velocity, ok.

And if you go back and remember the angular velocity is not directly related to the rotation matrix, ok. It is simply related to the rotation matrix. It was some $[\dot{R}][R]^T$ and so on. So, we need to find the relationship between angular velocity and Euler parameters, this is not as simple as $x(t)$ and $\dot{x}(t)$.

(Refer Slide Time: 35:22)



But it turns out that there is a relationship which is why Euler parameters are used. So, it turns out that the angular velocity of the tool coordinate system can be written in terms of rate of change of the Euler parameters ${}^{0}\dot{\varepsilon}_{Tool}$ and $\dot{\varepsilon}_4$ pre-multiply by a matrix $2[E(t)]$, ok. $[E(t)]$ is given here below.

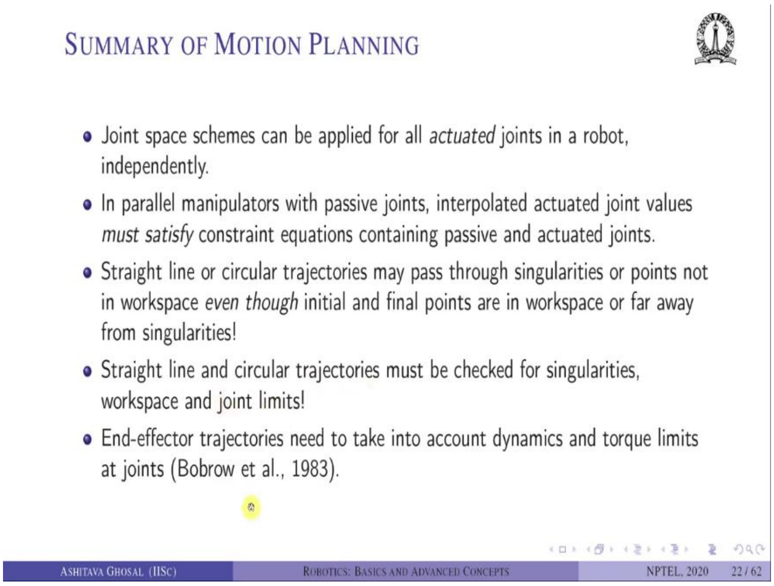
The inverse is always also possible. So, if I gave you the angular velocity of the tool, I can pre multiply $\frac{1}{2}[E(t)]^T$. And get the rate of change of $\dot{\varepsilon}_1$, $\dot{\varepsilon}_2$, $\dot{\varepsilon}_3$ and $\dot{\varepsilon}_4$. And $[E(t)]$ is a non-square matrix. So, this is the interesting part.

So, I am going from ω to ε or from $\dot{\varepsilon}$ to ω both directions, but I am not inverting any matrices, ok. So, it turns out that this is the real nice feature of Euler parameter. So, for a given $E(t)$ which is specified by $\begin{pmatrix} -\varepsilon_1 & \varepsilon_4 & -\varepsilon_3 & \varepsilon_2 \\ -\varepsilon_2 & \varepsilon_3 & \varepsilon_4 & -\varepsilon_1 \\ -\varepsilon_3 & -\varepsilon_2 & \varepsilon_1 & \varepsilon_4 \end{pmatrix}$. This in one case we use $2[E(t)]$ and one case we use $\frac{1}{2}[E(t)]^T$.

So, what can we do to plan trajectory for orientation? Ok. We plan C^2 continuous trajectories for given Euler parameter at $t = 0, t_f$ and rate of change of Euler parameters at $t = 0, t_f$. We compute the trajectory for ε_4. So, ε_4 is not generated, $\varepsilon_4(t) = \pm\sqrt{1 - {}^{0}\dot{\varepsilon}_{Tool} \cdot {}^{0}\dot{\varepsilon}_{Tool}}$. So, this ensures that the constraint is always satisfied at all t, ok.

So, from ${}^{0}\varepsilon_{Tool}(t)$ and $\varepsilon_4(t)$ obtain any required representation of the orientation of the end-effector at each instant of time. So, if I want to show you what are the Euler angles, I can obtain from $\varepsilon_1, \varepsilon_2, \varepsilon_3$ and ε_4 what are the Euler angles. I can also obtain the rotation matrix or even the axis and angle.

(Refer Slide Time: 37:59)



So, in summary, the joint space schemes can be applied for all actuated joints in a robot independently, ok. So, when I am planning the trajectories for θ_1, I do not need to worry about what is happening to θ_2. In parallel manipulators with passive joints, interpolated actuated joint values must satisfy the constraint equation containing passive and actuated joints, ok. This is important to see or to check, ok.

Straight line or circular trajectories may pass through singularities or points not in the workspace, ok. So, I have tried to move from one place to another place in between there is a hole in the workspace. So, even though the initial and final points are in the workspace or far away from singularity, in between points can be outside the work space or close to a singularity. So, these needs to be checked also, ok.

Straight line and circular trajectories must be checked for singularities, weather it lies always in the work space and also for joint limits. This is important, ok. So, not all joints in a robot can rotate fully. Till now we are assuming there are no constraints on the joint limits. But if there are joint limits we need to check for that.

Finally, all these end-effector trajectories or even the joint space trajectories that we have generated, do not take into account the dynamics and torque limits of the joint, ok. So, we can peacefully go and plan a trajectory. However, it turns out that in someplace the desired acceleration is very high, and it cannot be supplied by the motor which is there in the robot.

So, if you want to plan trajectories taking into account the dynamics of the torque limits then we need to do more advanced joint motion planning, ok. This was done in a paper by Bobrow in 1983, longtime back. So, it is not a very new thing.

So, with this we come to end of this lecture on Motion Planning. In the next lecture, we will look at control of a single link.