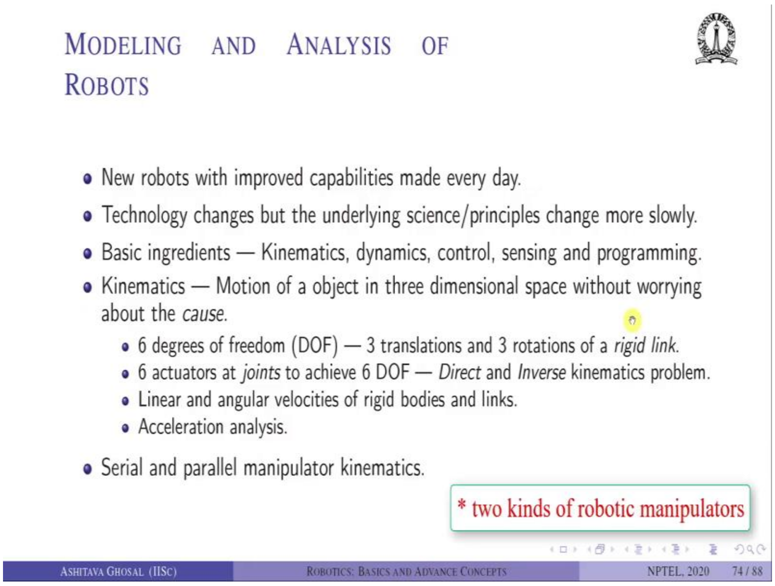
Robotics: Basics and Selected Advanced Concepts
Prof. Ashitava Ghosal
Department of Mechanical Engineering
Centre for Product Design and Manufacturing (CPDM)
Indian Institute of Science, Bengaluru

Lecture - 03
Modeling and Analysis of Robots

Welcome to this NPTEL lectures on Robotics, Basics and Advanced Concepts. In the last lecture, we looked at some of the main elements of a robot namely the links and joints, the actuators, the transmission mechanism and sensors ok. In this lecture, we will look at the Modeling and Analysis of Robots, the tools which are used to model and analyze a robot.

(Refer Slide Time: 00:46)



So, new robots with improved capabilities are made every day. When new motors are coming up, new sensors are coming up, the technology is changing, but the underlying science and principles of analysis change much more slowly ok. So, the basic ingredients in modelling and analysis of robots are kinematics, dynamics, control, sensing and programming.

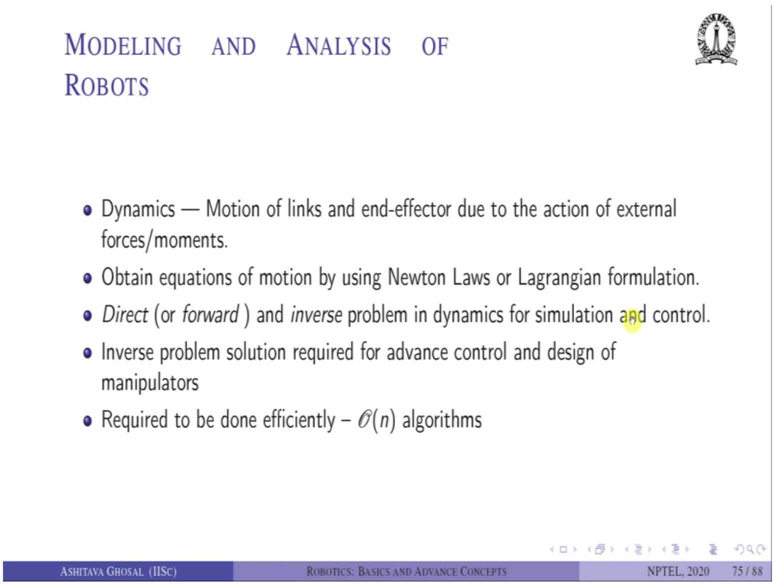
In kinematics, the motion of an object in three-dimensional space is considered without worrying about the cause of the motion ok. So, we do not worry about the forces and moments and external forces and moments which are coming on to the object. In kinematics, we have a rigid body in 3D space that is the most basic element in kinematics. It has 6 degrees of freedom a rigid body requires you to position means x, y, z coordinates. So, it can translate along this x, y and z coordinates, and it can also orient ok. So, there are three other parameters which describe the orientation.

So, the link of a robot which is considered to be a rigid body can have 6 degrees of freedom in general. Six actuators at the joints are required to achieve this 6 degrees of freedom of the end effector ok. So, we are worried about the 6 degrees of freedom of the rigid body which is the end effector. And then we need to somehow have 6 actuators which enables us to achieve all those 6 degrees of freedom ok.

There are two main problems in kinematics one is called the direct problem and one is called the inverse problem. We also look at the linear and angular velocities of rigid bodies and links in kinematics, and finally, we also look at the acceleration analysis. There are two kinds of robotic manipulators, one is serial and one is parallel manipulator as we have seen earlier.

And they have two different kinds of kinematics ok. So, the direct and the inverse problem for a serial manipulator is very different from the direct and inverse problem for a parallel manipulator.

(Refer Slide Time: 03:16)

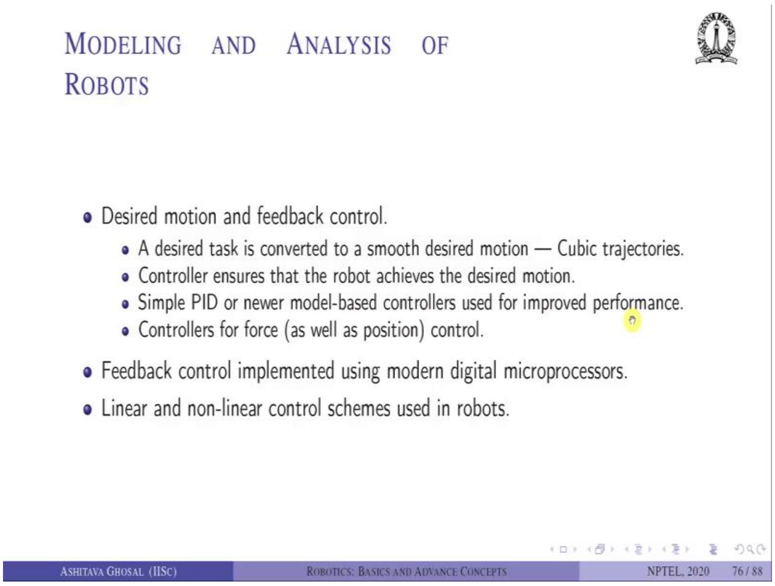


In dynamics, we look at the motion of the links and end effectors due to the action of external forces and moments. So, we obtain the equation of motion using various principles.

So, we can use, for example, the Newton's law or the Lagrangian formulation or there are some other ways of deriving equations of motion (Refer Time: 03:42)called Kane's formulation. And then we look at two problems in dynamics again – one is the direct or sometimes called the forward problem; and one is the inverse problem which is both are used for simulation and control.

The inverse problem requires is required for advance control and design of manipulators; the direct problem requires you to solve the equations which you have derived and then it is used for simulation and control. What we want to do is we want to solve the dynamics both the direct and the inverse problem efficiently ok. So, basic goal is to obtain what is called as algorithms which have $\mathcal{O}(n)$ complexity where n is the number of links.

(Refer Slide Time: 04:31)

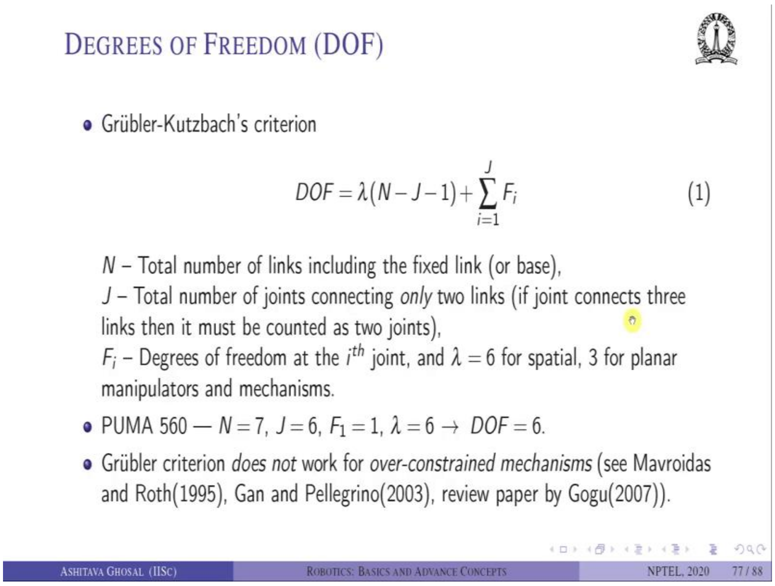


Once we have a desired motion, we need feedback control to achieve that desired motion ok. So, most of the time a desired task is converted to a smooth desired motion through what are called as cubic trajectories which you will see later on in the lecture on controls. The controller ensures that the robot achieves the desired motion ok. Typically a simple PID, nowadays newer model based controllers are used for improved performance.

We can also control the force which the robot applies to the environment. And for this we develop or we or what are called as force controllers are used. So, we sometimes need to control forces, and sometimes we need to control position. So, most of the time, the feedback control which is used to control the and obtain the desired motion are implemented using modern digital microprocessors ok.

So, as the microprocessors are becoming better and better, we can control more number of links with the same processor, or we can use sophisticated model based control which requires large amount of computation ok. Most of the time in a robot a linear control scheme is used which is as I said it is a simple PID control scheme or as I said sometimes newer non-linear control schemes are also being used nowadays.

(Refer Slide Time: 06:12)



Let us go a little bit deeper into kinematics. One of the most important concept in kinematics is this concept of degrees of freedom ok. So, the degree of freedom of a mechanism or a robot can be obtained by this formula which is

$$DOF = \lambda(N - J - 1) + \sum_{i=1}^{J} F_i$$

So, in this equation, N is the total number of links including the fixed base ok, this is due to convention; J is the total number of joints connecting only two links.

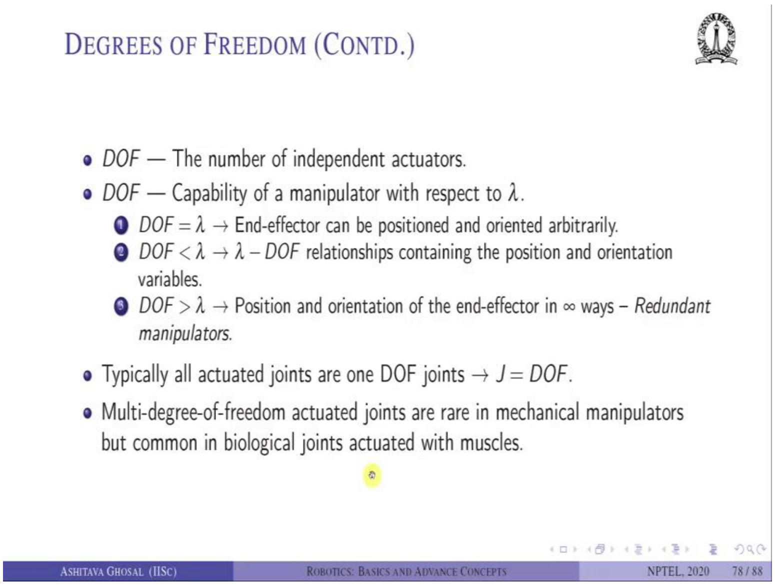
See if a joint connects three links, then it must be counted as two joints. And F_i is the degree of freedom at the i^{th} joint. So, remember if you have a rotary joint, it has one degree of freedom. If it has a prismatic joint, it has one degree of freedom. If you have spherical joints, it has three degrees of freedom ok. And this λ in some sense represents the space in which the motion is taking place. So, λ is equal to 6 if the motion is taking place in 3D space. And it has 3 if the motion of the manipulator or the mechanism is in a plane ok.

So, we can use this formula very quickly. Recall that in the PUMA example. There were N = 7. There were 6 links, but we have to also include the fixed base, so that is why N is 7. The number of joints were 6.

There were 6 rotary joints ok. Each rotary joint had 1 degree of freedom λ is 6, hence the degree of freedom of the PUMA is found to be 6. What does it mean? If you attach an end effector at the PUMA robot at the end of the PUMA robot, I can position and orient the end effector arbitrarily in 3D space.

This Grubler criteria is a very very powerful equation. However, it does not work always ok. So, it does not work for what are called as over constrained mechanisms ok. So, if you want to look at over constrained mechanisms, a large number of researchers have worked on it, and these are some of the papers. So, there is a nice review paper by Gogu which appeared in 2007 which discusses in depth what are over constrained mechanisms and when this Grubler criteria cannot be used to obtain the degrees of freedom.

(Refer Slide Time: 08:43)



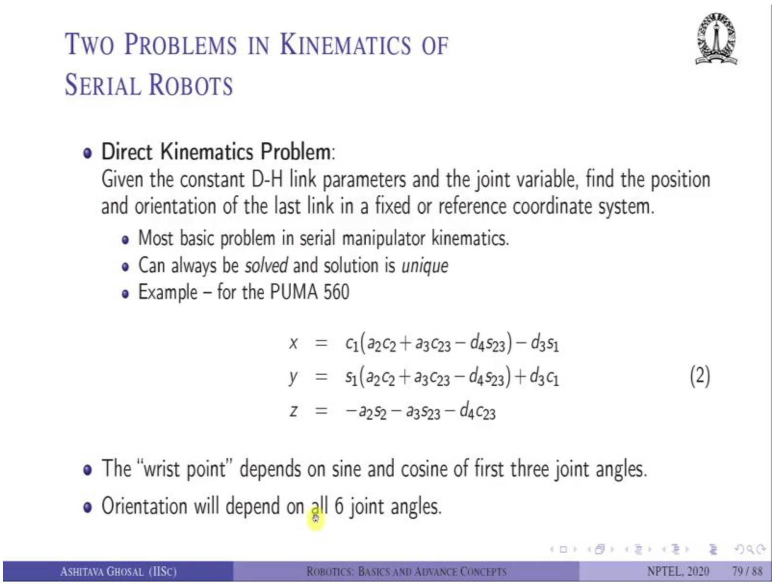
Let us go a little bit deeper that quantity DOF which we calculate is the same as the number of independent actuators ok. So, if the DOF is 6, I cannot put 7 actuators ok. I can put 5, but then that is the special case those are called under actuated mechanisms. The DOF also gives you the capability of the manipulator with respect to λ ok. So, if lambda were 6 or 3, DOF is equal to λ, then the end effector can be positioned and oriented arbitrarily either in 3D space or in the plane.

If DOF is less than λ, then $\lambda -$ DOF relationships containing the position and orientation variables must exist ok. So, if you have a SCARA robot which you will see later, it has 4 degrees of freedom DOF is 4. So, basically the end effector does not have two orientations possible ok. If DOF is greater than λ, the position and orientation of the end effector can be obtained in infinitely many ways, these are called redundant manipulators.

So, these are also called snake manipulators ok. In one of the videos, I had shown you a snake manipulator which was developed in IISc. Typically all joints in a robot are one degree of freedom joints ok. Why? Because we need to actuate these joints. In some robots, specially parallel robots there are joints which are not actuated and hence they can be passive, and we can have multi degree of freedom joints. So, most of the time J is equal to DOF. So, example in the PUMA, the DOF was 6 and the number of joints were 6.

Multi-degree-of-freedom actuated joints are rare in mechanical manipulators, but they are very common in biological joints. So, for example, if you look at your shoulder joint, it has three degrees of freedom, but it are they are actuated using muscles ok, the arrangement is different. If you have a three degree of freedom spherical joint and if you want to actuate all those three degrees of freedom, then you have to put three motors at the same place, and it is almost impossible ok.

(Refer Slide Time: 11:17)

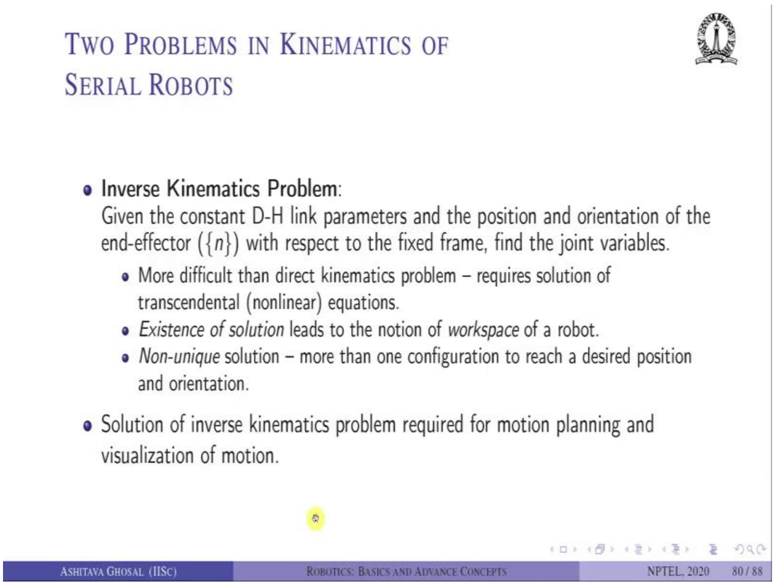


Let us continue as I said there are two problems in kinematics of serial robots one is called the direct kinematics problem. Basically what is given is we are given all the constant D-H parameters, remember I said that there are four D-H parameters out of which three are constant, and one is a joint variable. So, we are given the constant D-H link parameters and the joint variable, find the position and orientation of the last link in a fixed or a reference coordinate system ok.

So, this is the most basic problem in serial manipulator. It can always be solved and the solution is unique ok. We will see later. It basically depends on some matrix multiplication. So, for example, for the PUMA 560, the wrist point which is x, y, and z can be obtained in terms of θ_1, θ_2, θ_3. So, c_1 here means $\cos\theta_1$, s_{23} means $\sin(\theta_2 + \theta_3)$, and this a_2, a_3, d_4 and d_3, they are the constant D-H parameters ok.

What is of interest is? That the wrist point depends on the sine and cosine of the first three joint angles in the case of PUMA ok. In other robots, it need not be that. The orientation of the end effector on the other hand for a PUMA depends on all the 6 joint angles. Say if I were to write a rotation matrix or prescribe the orientation of the wording touch with this PUMA is carrying, it will depend on all the 6 angles.

(Refer Slide Time: 13:00)



The inverse kinematics problem is stated as follows. Given the constant D-H parameters and the position and orientation of the end effector coordinate system this {n} here means its a coordinate system and curly bracket means is the end effector coordinate system or the nth coordinate system with respect to a fixed frame, find the joint variables ok. So, in the case of a

PUMA, you are given the position and orientation of the end effector. You have to find θ_1 to θ_6 ok.

So, why do we need to solve the problem? If the robot is at end effector it is at some x y, z and orientation, and I want to move to another x, y, z, and orientation, I need to compute how much each joint I need to rotate ok. So, if we solve the inverse kinematics problem at these two position and orientations, we find the θ's corresponding to both of them, and we subtract both of them and then we know how much the motor should rotate.

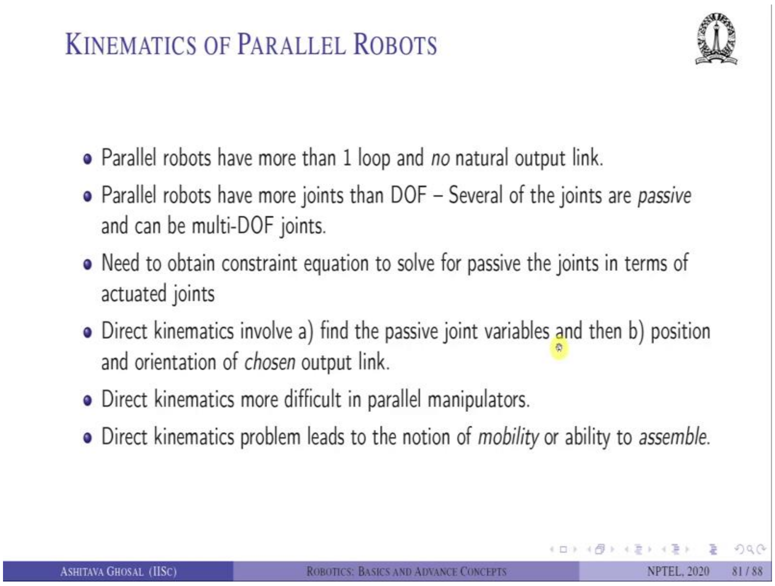
This is the more difficult problem than the direct kinematics problem ok. It requires the solution of transcendental non-linear equations ok. We will come to that later when we look at inverse kinematics of manipulators. It also depends on the existence of the solution ok.

Sometimes for a given position and orientation, the solution may not exist ok. And points for which the solution does not exist, orientation for which the solutions does not exist, they lie outside the workspace of the robot. So, the analysis of existence of the inverse kinematics problem leads to the notion of workspace of a robot.

The inverse kinematics solution is also not non it is not unique ok. The non-uniqueness nature of the solution means that I can achieve one position and orientation in more than one way ok. So, I can have two or more sets of values of joint variables which can achieve the same position and orientation ok. The solution of inverse kinematics problem required is required for motion planning and visualization of the motion.

So, as I said I am at one position and orientation I need to move to the other position and orientation. So, I solve the inverse kinematics problem at some let us say several intermediate steps. And I can show how the robot is moving from one position and orientation to the other position and orientation. So, we can visualize the motion of the robot and all its links.

(Refer Slide Time: 15:52)



In kinematics of parallel robot, there are more complications. The first complication is the parallel robots have more than 1 loop and no natural output link ok. So, it is not as if there is a fixed base 1 joint 1 link, 1 joint 1 link, all the way to the free end effector. In a parallel robot, there are loops ok.

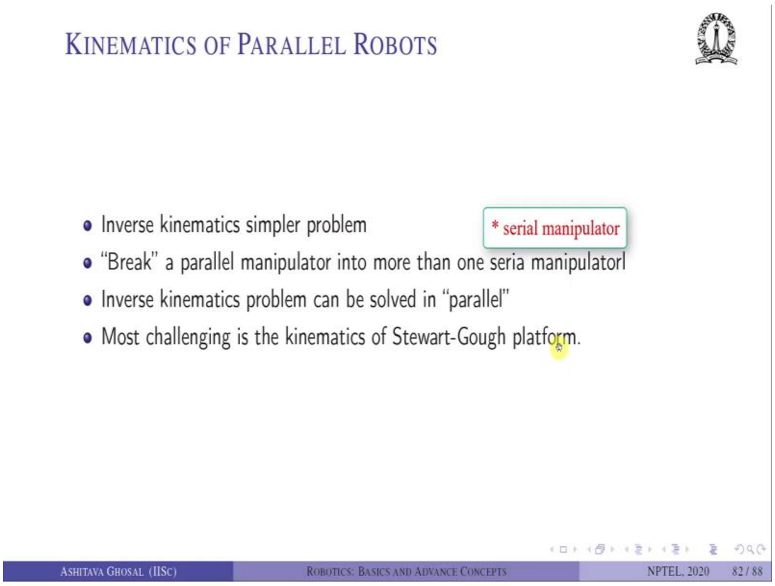
The second complication is parallel robots have more joints than the degree of freedom ok. So, several of the joints are passive. And since they are passive there can be multi-degree-of-freedom joints. So, for example, in the Stewart Gough platform which I showed you, there was a spherical joint which is a three degree of freedom joint.

So, in order to solve the kinematics of a parallel robot, we need to obtain constraint equations to solve for the passive joints in terms of the actuated joints ok. The actuated joints are the ones we have motors, we can measure the rotations of the actuated joints or measure the translations of the actuated joints. But in order to draw this parallel robot are describe the configuration of the parallel robot, we also need to compute the values of the passive joint variables ok, so that is obtained by solving what are called as constraint equations.

The direct kinematics of a parallel robot is two steps. First is you find the passive joint variables and then you find the position and orientation of a chosen output link ok. There is no natural output link; there is no free end. So, we have to say this is the output link, and this is how we obtain the position and orientation. But before that, we need to solve for the passive joint variables.

The direct kinematics is more difficult in parallel manipulators because of these two steps and because of this existence of passive joints. The direct kinematics problem leads to the notion of mobility or the ability to assemble the closed loop mechanism at that configuration ok. We will discuss this in more detail in a future lecture ok. Just like the serial robot existence led to the notion of a workspace, the direct kinematics problem in parallel robots leads to the notion of mobility.

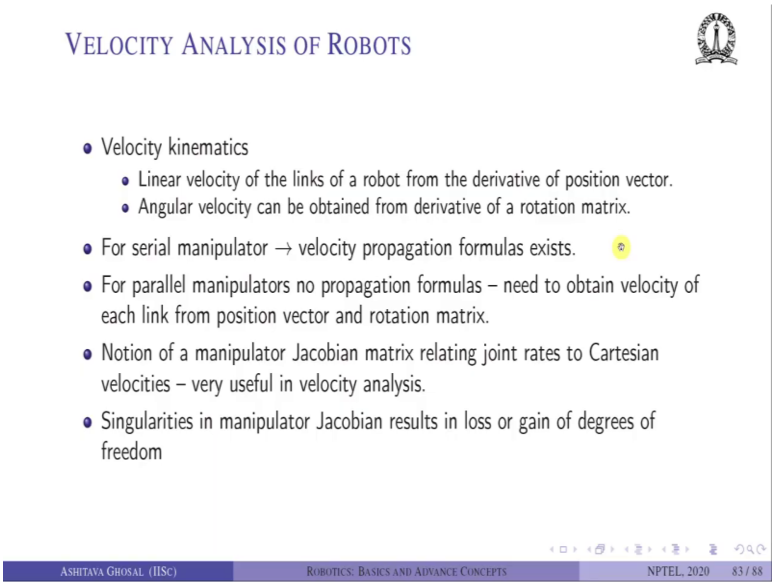
(Refer Slide Time: 18:21)



The inverse kinematics of a parallel robot is a much simpler problem ok. So, the basic idea is you break a parallel robot into more than one serial manipulator ok. We solve the inverse kinematics for each one of these serial manipulators, and we can solve it independent of each other, we can solve them in parallel ok. And hence sometimes what parallel manipulators comes from the fact that we can solve the inverse kinematics in parallel.

One of the most challenging problem in parallel manipulator kinematics is the Stewart Gough platform ok. The direct kinematics of a Stewart Gough platform is very complex, a general Stewart Gough platform is very complex; and only very recently they have been solved.

(Refer Slide Time: 19:14)



In kinematics, we also look at the velocities of the joints and the links ok. So, in velocity kinematics, we look at the linear velocities of the links of a robot from the derivative of the position vector ok. The angular velocity can be obtained from the derivative of a rotation matrix ok. We will come to these things in more detail later.

For serial manipulators, we can derive what are called as velocity propagation formulas ok. So, we start from the fixed base which is zero velocity. Then due to the joint the first link will have some velocity; then due to the velocity of the first link we will add in some nice way the velocity at the joint of the at the end of the first link and we get the velocity of the second link and so on. So, we can propagate the linear and angular velocity from the fixed base all the way to the free end effector.

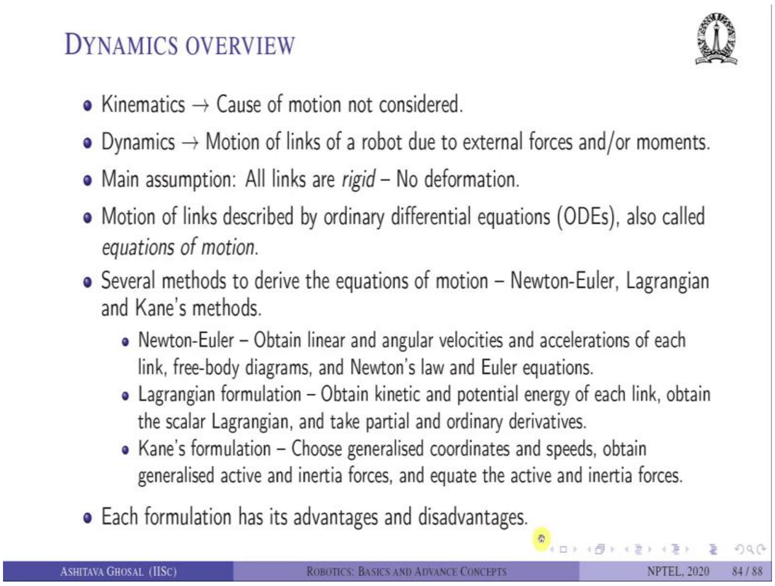
For parallel manipulator, no such propagation formulas exist ok. Why? Because there are loops and we need to obtain the velocity of each link from the position vector and the rotation matrix. We cannot use these nice propagation formulas. The notion of a velocity linear and angular velocity of the links of a robot and the rotations at the joint, the rates of rotation at the joints gives raise to what is called as a Jacobian matrix ok.

So, this velocity Jacobian matrix relates the joint rates to the Cartesian velocities of the links of a robot ok. And its a very very useful tool for velocity analysis ok. So, it tells you that given the joint rates, I can find the linear and angular velocity of the end effector. And under certain conditions given the linear and angular velocity of the end effector I can find the joint rates we can go back and forth.

The Jacobian matrix also can be used to obtain what are called a singularities in the manipulator ok. So, what is the singularity? At the singularity the manipulator can be shown to lose one or more degrees of freedom, this is what happens in a serial robot. In a parallel robot, it can lose one or more degrees of freedom, but it can also gain one or more degrees of freedom ok.

So, these are also very very important notions, conceptual things which we need to know in a robot ok. If you lose a degree of freedom, then you can show that certain joint rates become very high as we approach that loss of degree of freedom. And hence the controller needs to know otherwise a controller can keep on supplying more and more voltage and it will fail ok. It can try to obtain the desired velocity, and it may not be possible ok. So, it can result in some problems with the controller.

(Refer Slide Time: 22:25)



The overview of dynamics; let us look at it very briefly. In kinematic, the cause of motion was not considered. In dynamics, the motion of the links of a robot due to external forces and moments are considered. The main assumption is all links are rigid ok. We do not consider in most of robotics the flexibility of the links. As soon as you include the flexibility of the links in a robot, it will becomes very very complicated ok.

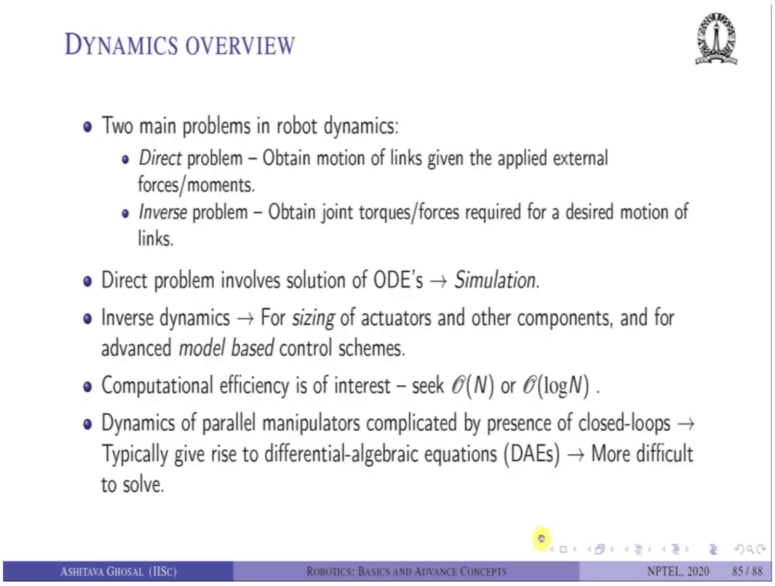
The motion of the links of a robot which are modeled as rigid are as ordinary differential equations ok. We can show that they lead to ordinary differential equations. As soon as you introduce flexibility and deformation, we get what are called as partial differential equations, and they are much more harder to solve. There are several methods to derive the equations of motion.

The three main ones are Newton-Euler, where we use a Newton's law and the Euler's equation of motion we have one which is called as the Lagrangian formulation and one which is called the Kane's method. In Newton-Euler method, we obtain the linear and angular velocity and acceleration of each link in the robot, we draw free body diagrams, and then we use Newton's laws and Euler's equation.

In Lagrangian formulation, it is an energy based method. We obtain the kinetic and potential energy of each link. We obtain the scalar Lagrangian which is nothing but the sum of all the kinetic energy minus sum of the potential energy, and then we do a set of partial and total derivatives and ordinary derivatives to obtain the equations of motion.

In Kane's formulation, we choose a generalised coordinates and speeds ok, obtain the generalised active and inertia forces and equate the active and inertia forces ok. So, this is not very common, but there are claims of advantages of using Kane's formulation ok. Each formulation has its own advantage and disadvantages.

(Refer Slide Time: 24:48)



There are two main problems in robot dynamics. First is this direct problem which is stated as obtain the motion of links given the applied external forces and moments ok. The inverse problem on the other hand is obtain the joint torques and forces for required for a desired motion of the links. So, I want to move the links with certain velocities and acceleration what is the joint torques that are required.

The direct problem involves the solution of the ODEs – Ordinary Differential Equation which is also sometimes called simulation. The inverse dynamics on the other hand is required to size the actuators ok.

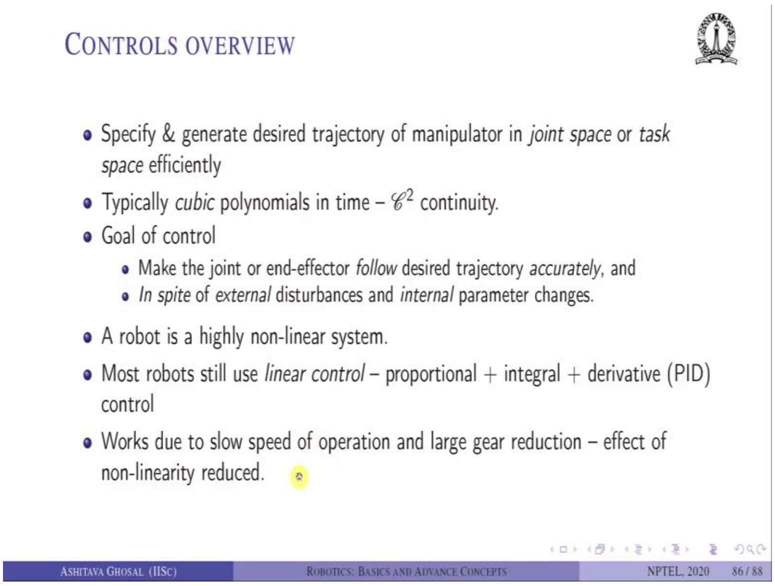
If you want to design a robot capable of doing certain motions ok, then we need to know what is the power of the motor that we want ok, how to design the components to withstand the forces and moments which are occurring when the robot is moving ok; it is also used in what is called as advanced model based control schemes.

In both of these the computational efficiency is of interest ok. We seek a linear complexity algorithm or better a log(N) complexity algorithm ok. And this is because now there are systems with many many links ok. So, we will see later that this dynamics formulation, Newton-Euler formulation have been apply into what is called as protein kinematics ok. So, each element or the amino acid in a protein chain is assumed to be a rigid body, and then this protein will fold to a certain shape ok.

So, in order to find the motion of this chain, we have to solve the dynamics equations of motion ok. We have to derive and solve. So, we need very very efficient algorithms because the number of amino acids in a chain could be as much as 500 ok, not in a robot. Robots is normally a 6, 10 like that; you could have hundreds of elements in a protein.

So, the dynamics of parallel robots are complicated by presence of closed loops as I said earlier. They give rise to what are called as differential algebraic equations ok. The differential equation comes from the derivatives, but you also have this constraint equations which are algebraic equation. So, we need to solve both of them at the same time. These are much more difficult to solve than simple ordinary differential equations.

(Refer Slide Time: 27:31)



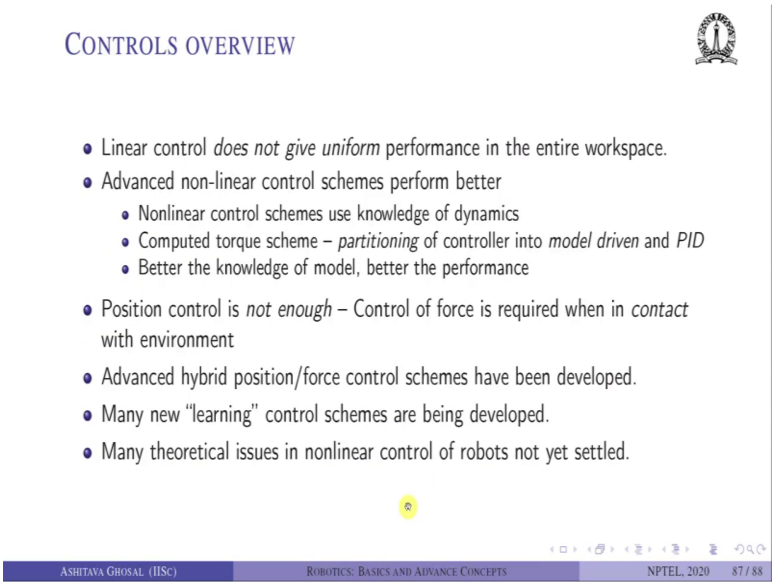
A quick review overview of control. So, basically in control, we specify and generate desired trajectory of a manipulator in joint space or task space ok. Most of the time cubic polynomials in time are used. Cubic polynomials gives you C^2 continuity ok, even the accelerations are continuous. The goal of the control is to make the joint or end effector follow desired trajectory accurately ok; and in spite of external disturbance and internal parameter change ok.

So, what is accurately? depends on what is the task. In some cases, we might want to follow within a millimeter of the desired trajectory; in some cases, it could be larger or smaller. This notion of in spite of external disturbance and internal parameter change is also very important in control. I could have designed a robot and a controller to work at some environment ok.

I take it to a very hot or a very cold climate ok, then some of the parameters of the motors or of the robot can change, but the robots should still work ok. So, in spite of some external disturbance which is happening or internal parameter change, the controller should still work. A robot is a highly non-linear system ok. We will see this later when we derive the equations of motion of a robot that it has many non-linear terms.

Most robot still use linear control ok, and in particular a proportional integral and derivative, PID control is used ok. This works simply because most robots operate at slow speed, and there are large gear reductions ok. If you move at slow speeds and the there are large gear reductions, the effect of nonlinearity is reduced.

(Refer Slide Time: 29:34)



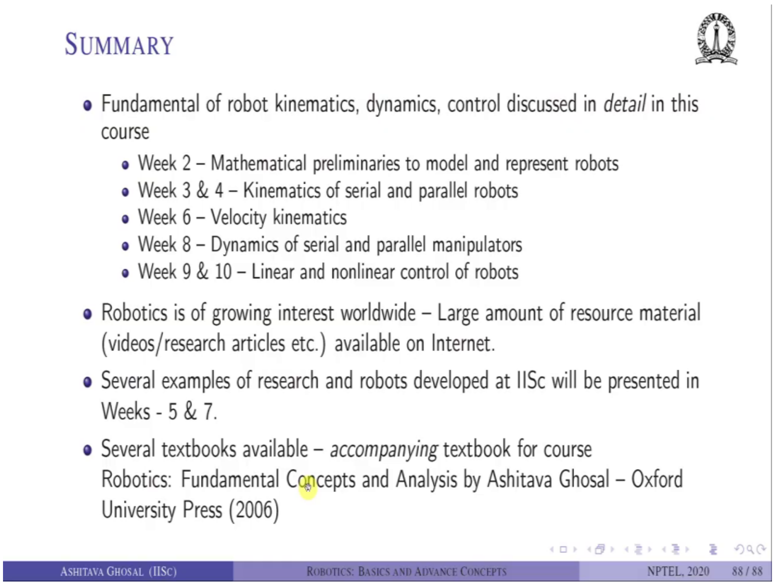
Linear control does not give uniform performance in the entire workspace of the robot ok. As a result several advanced non-linear schemes have been developed, and they can be shown to perform better ok. Nonlinear control schemes use knowledge of the dynamics of the robot ok. One such scheme called computed torque control scheme it partitions the controller into a part which is model driven meaning it uses the dynamic equations of motion and part which is PID ok.

And it can be shown that if you know the model better if you have a better knowledge of the model, the performance is better. A third important point in control is that position control is not enough ok. Control of force is required when in contact with environment ok. The robot needs to apply certain force it needs to grip an object and not break it ok. So, for such applications where you need to apply a desired force, advanced hybrid position force control schemes have been developed ok.

Nowadays, we are also developing what are called as learning control schemes ok. So, the controller can learn the parameters of the robot or it can learn how to efficiently move from one place to another place ok. So, it uses this modern techniques of reinforcement learning and the other kinds learning.

Finally, many theoretical issues in non-linear control is not yet settled. So, for example, if you have stability issues in a robot ok, can we prove that certain gains will give you stability, or can we prove that this non-linear control scheme is stable under such and such conditions ok? None of these results are known. So, non-linear controls is still an open area in robotics.

(Refer Slide Time: 31:51)



So, in summary the fundamentals of robot kinematics, dynamics, control will be discussed in this course ok. This fundamental concepts of kinematics dynamics control do not change much. The technology keeps on changing. The different kinds of motor sensors appear, but the basic notions of kinematics dynamics and control do not change, and we will discuss this in great detail in this course.

So, in Week 2, we will look at mathematical preliminaries to model and represent robots; in Week 3 and 4, we look at the kinematics of serial and parallel robots; in Week 6, we look at velocity kinematics. In Week 8, we look at the dynamics of serial and parallel manipulators; and Week 9 and 10, we look at the linear and non-linear control of robot ok.

Robotics is a growing interest worldwide ok. Large amount of resource material videos or research articles, you know various software tools, they are available in the internet ok. And you can go and see many of these very nice videos and research articles. We will also show you some examples or some research which we have done in IISc ok. These will be contents of Week 5 and 7.

There are several textbooks which are available for this course ok. The accompanying textbook for this course is the book written by me, accompanying means many of the detailed steps ok, some calculations and extra reading material which I will just mentioned in the course are available in the textbook, and also that references in each of the chapters in the textbook.

So, anyone who is interested in going into more detail on any of these topics – kinematics, dynamics, control, sensing, various other things, I recommend that you look at literature available in the internet or in textbooks ok. So, thank you. From next week, we will start with the mathematical preliminaries to model and represent robots.