NPTEL Online Certification Courses

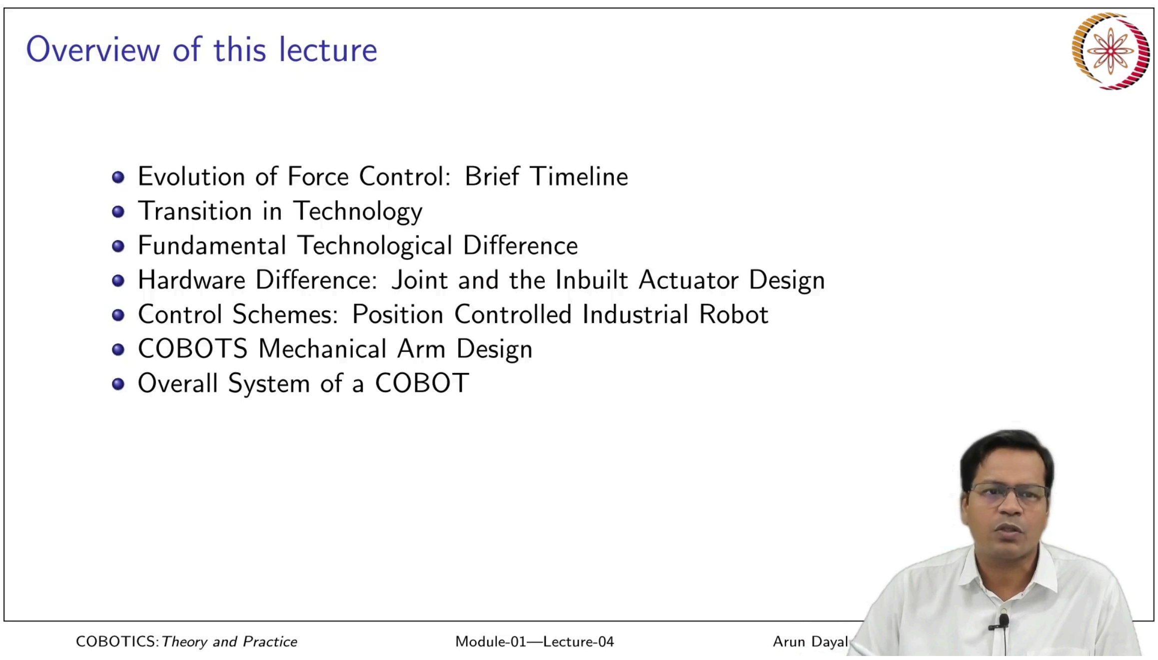
COLLABORATIVE ROBOTS (COBOTS): THEORY AND PRACTICE

Dr Arun Dayal Udai
Department of Mechanical Engineering
Indian Institute of Technology (ISM) Dhanbad

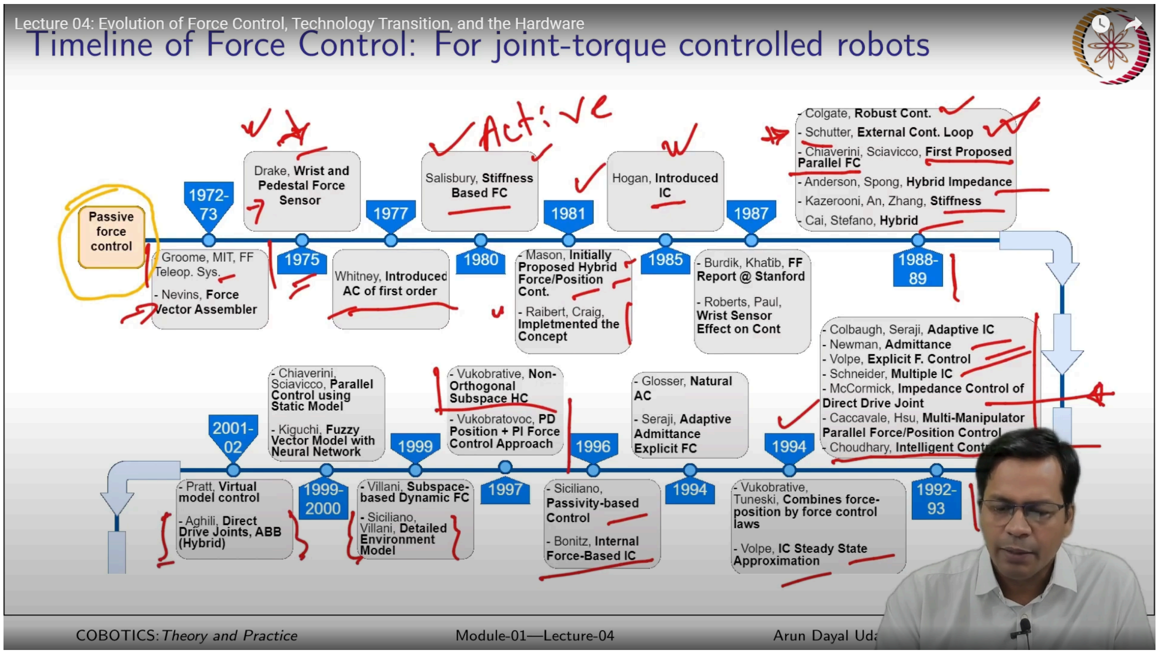
Week: 01
Lecture: 04

Evolution of Force Control, Technology Transition, and the Hardware

Welcome back. So far, I have introduced you to Cobots, its requirements and applications in the modern industry, ISO standards, its mode of operation, risk levels, risk assessment, and its reduction measures.

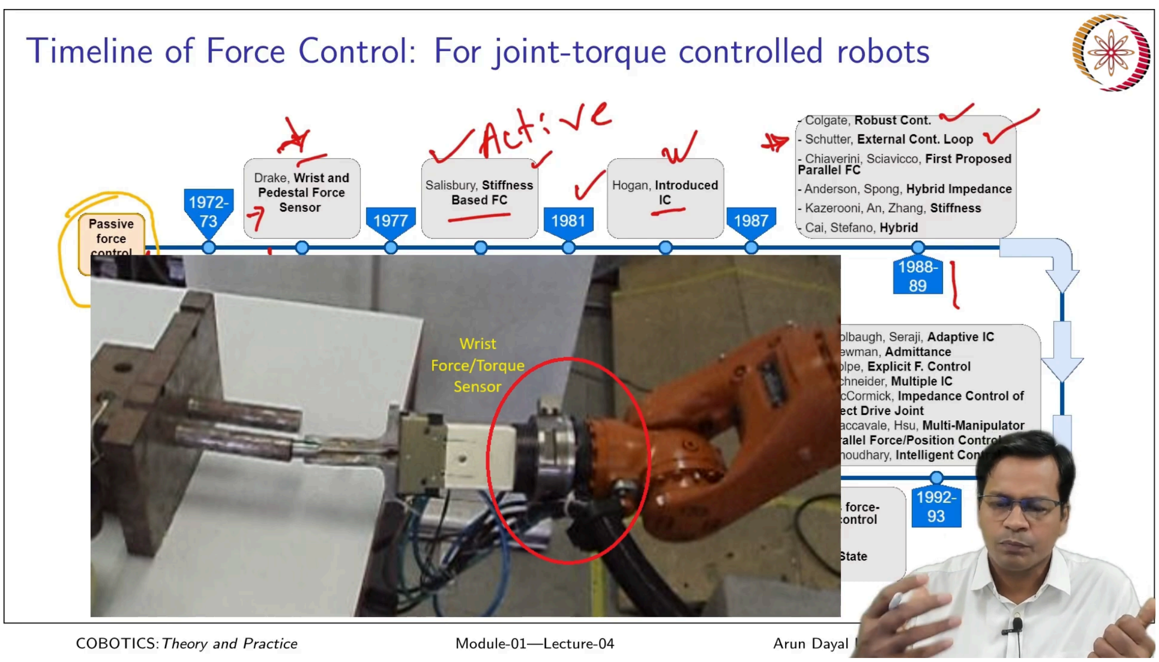


So, the overview of this lecture is as follows. So, I will be discussing on Evolution of Force Control and its Brief Timeline. Transition in Technology, Fundamental Technological Difference. Hardware Difference- joint and the inbuilt actuator design. Control Schemes- Position Controlled Robot as well as for the COBOTS, COBOTS Mechanical Arm Design and the overall system of a COBOTS.



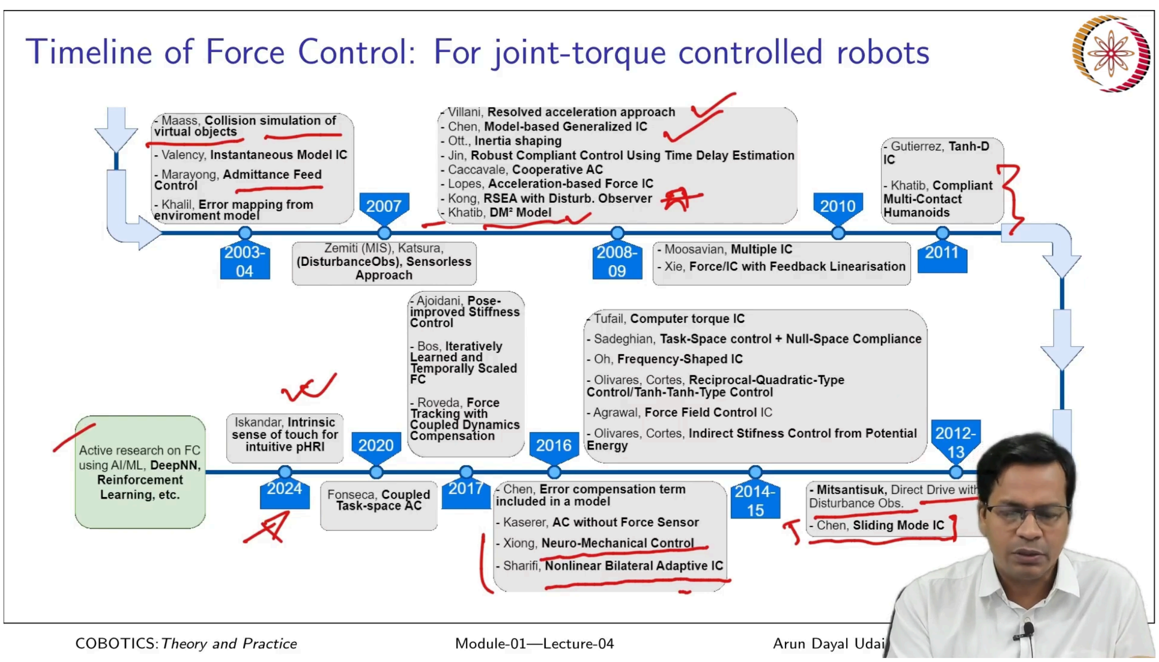
So, let us begin with the timeline for Force Control. So, what exactly is passive force control here? So, passive control was a device that was fitted in between the the gripper or any kind of end effector tool and the robot plane. So, that came in between. So, that was compliance. So, what happens if the forces come onto the gripper or the tool that is isolating the body of the robot from the gripper? So, that intermediate tool was giving it passive compliance. It was a spring-like structure which is fitted at the end to take care of any collision or anything that was happening beyond the gripper. The whole of the robot was still stiff. So, this is one of the earliest methods of doing that. That was done by Groome MIT Feed Forward (FF) Teleoperation System and by Nevins, and then later on, in 1975, wrist and pedestal force sensing came in. That is where we started using end effectors for stock sensors instead of this passive compliance, and the whole of the system became active. That means the life forces we are taking into account we are creating it in a closed loop, and we could Desire for a particular force that was not possible in the case of a passive compliance device.

Now, in case of active compliance, we can maintain a particular force at the end effector or the torque at the end effector. So, this is where we started with active force control in 1975. Then Whitney introduced active control of first order, which was known as Admittance control here. Salisbury introduced the Stiffness controller, which is again trying to make the robot behave like a spring after the end effector after the flange actually handles any external things. So, that was again an active force control only, but it is in a closed loop. And then, in 1981 actually, Mason he provided with hybrid position force control. That is a hybrid of position as well as force control. Robots can go to a particular place and handle the forces as well. So, again similar one was given by Raibert and Craig that is there in the test book also. He implemented a hybrid force control technique with a standard industrial robot, where he partitioned the workspace into orthogonal spaces of force and position control. That was his contribution in 1981, and again, Hogan introduced the impedance controller. This is one of the benchmark introductions in force control where impedance control was introduced, and Even in current days all the Cobots use impedance controllers of advanced implementation.



And then, a series of things came in in 1988-89 when a Robust controller external control loop was given by Schutter. He again utilised a force control that is sensed through the wrist force-torque sensor and created a closed loop, which is reported even by this. So, it is an advanced version of this that was reported by Schutter that uses the standard industrial robot and does the force control job. Then he first proposed the parallel force control that was given by Chiavweini Sciavicco, and then hybrid impedance came in stiffness is here hybrid again. So, you see, a set of reporting started coming in in 1988-89 and again in 1992-93 another huge set of controllers that came in Adaptive Control, Admittance Control, Explicit Force Control, Multiple Impedance Control.

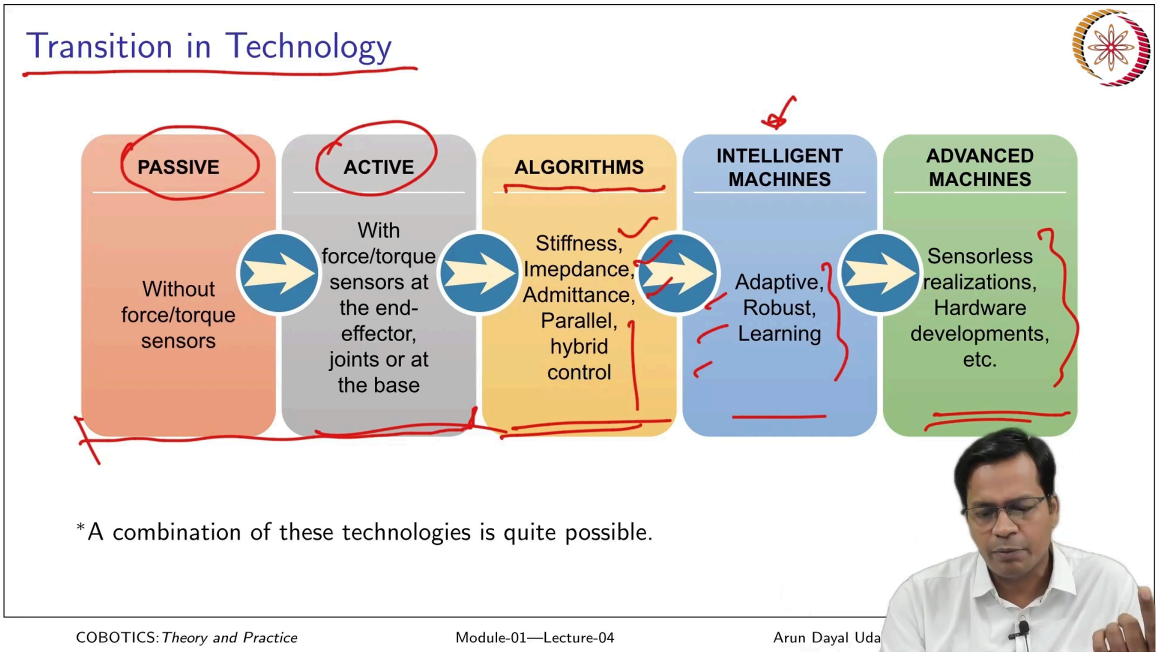
Here, this is again an important on,s the Direct Drive Joint, which was later on adapted by ABB also. So, impedance control of the direct drive joint, direct drive means there is no gearbox in between; the link was directly coupled to the high-powered motor which is there. Then it became very much compliant enough because there was no gearbox, it was back drivable and the torque reflectance ratio was too good. So, this is one of the major benchmarks. So, that is there, and then Caccavale came up with multi-manipulator parallel force position control intelligent control, also reported here. From here, you see quite a lot of papers started coming in intelligent control. And then 1994, some steady state approximation came in. Passivity-based controller, force based impedance controller and multiple control approaches PD and PI. PD for position PI for force control was used. Non-orthogonal subspaces, the hybrid controller was given. A detailed environmental model is developed here. Then, Aghili finally came up with the Direct Drive Joints for the ABB robot, which was a hybrid controller that was implemented.



Moving further, the collision simulation of virtual objects was taken into account. Admittance feed controller, Resolved acceleration approach, and model-based generalised impedance controller came in. The rotary series elastic actuator and disturbance observer came in here. This is again a very good model which was there. And DM square model (DM^2), when parallel motors were used at a joint and with high and low frequencies, was given by Khatib and a team from Stanford that was given. That is again a very important thing which was given. Khatib came up with a compliant multi-contact humanoid also, when multiple contacts are taken into account. And then sliding mode impedance controller. Apart from various other impedance controllers, sliding mode control, which was much better, was reported by Chen here. Direct drive with disturbance observer, again it came in.

Neuro-mechanical control, nonlinear bilateral adaptive control. So, you see, these are the controllers with nonlinear and adaptive. Neural ANN kind of algorithms were introduced here. So, you see now things are moving towards intelligent approaches and sensorless realisations started in the direct drive with disturbance observer technique is one of them. Intrinsic sense of touch for intuitive pHRI. This is the recent report that uses a standard in this talk sensor, which is already there at the joint. They used it to detect the touch without having a tactile sensor. So, it is a kind of sensor-less realisation of tactile sensors in the robot. I will show you the video of this also.

So, and then you see active research is going on in force control using AIML-based approaches Deep Neural Networks, Reinforcement Learning, and those approaches are now going on.

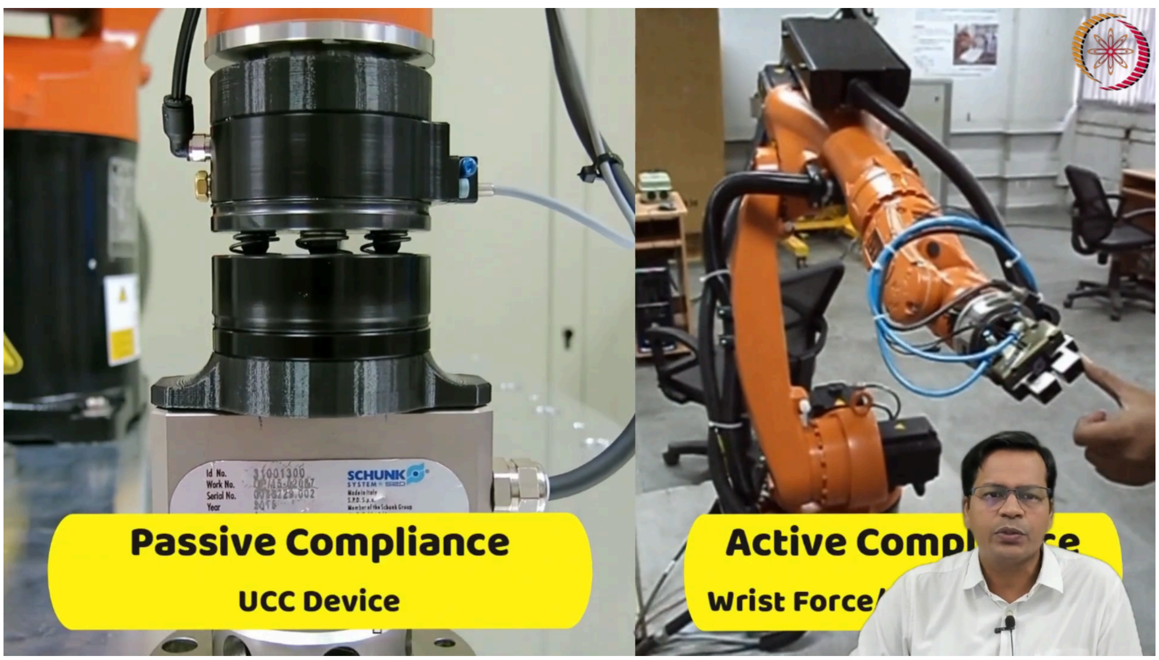


So, what we see is the transition in technology. We started with passive and went on to active with different sensors at the joint at the pedestal of the robot. Even at the wrist of the robot. So, different sensors are used and use that in a closed loop to create the forces at the end, at the joint compliance is established. Various algorithms came in with stiffness impedance admittance, and parallel hybrid force control was reported and finally, we started working with intelligent machines that used adaptive and robust learning-based approaches here, and now it is on sensor-less realisation, and hardware developments are becoming very, very popular. Now, they are trying to make things very compact and very much implementable for all these algorithms that were there. So, a combination of technologies is also there. Let's say it can be a combination of active and passive controllers. Like, let's say a robot made an active controller to control the joint forces and joint torque, whereas the end effector you can have still has a passive compliance spring-like structure. And you can have multiple algorithms with active sensing. Robust adaptive can be made along with an impedance controller. Even sensorless control, like tactile sense can be done with admittance thing. So, these technologies are also used in combinations.

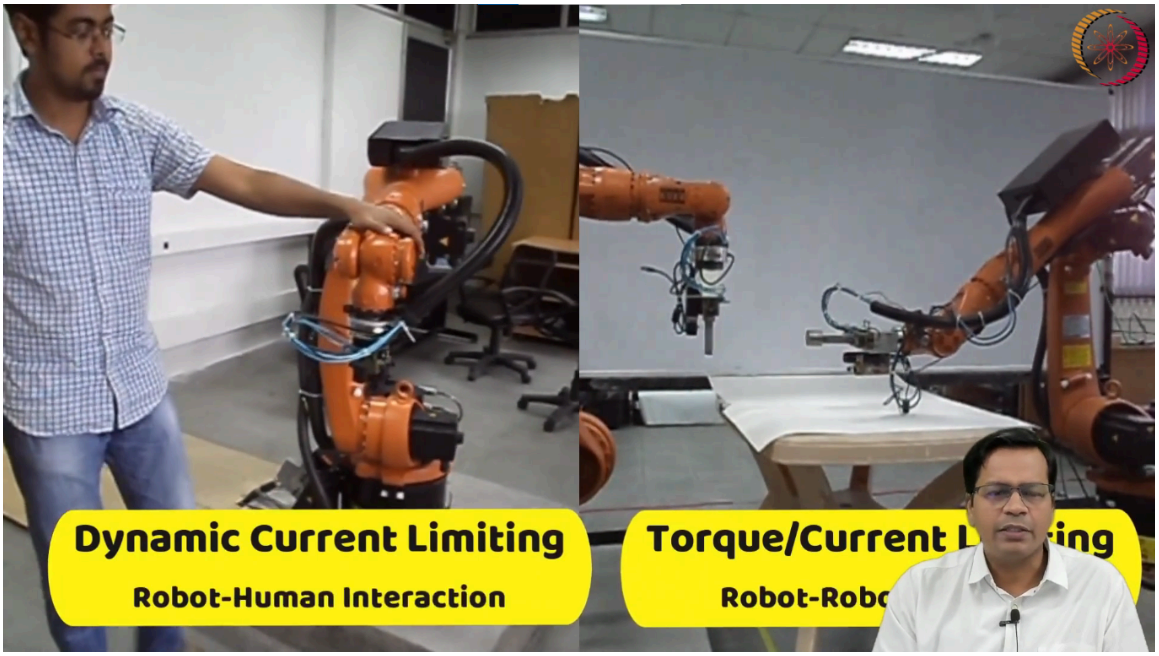
Videos: Technology Transition

Videos: Technology Transition

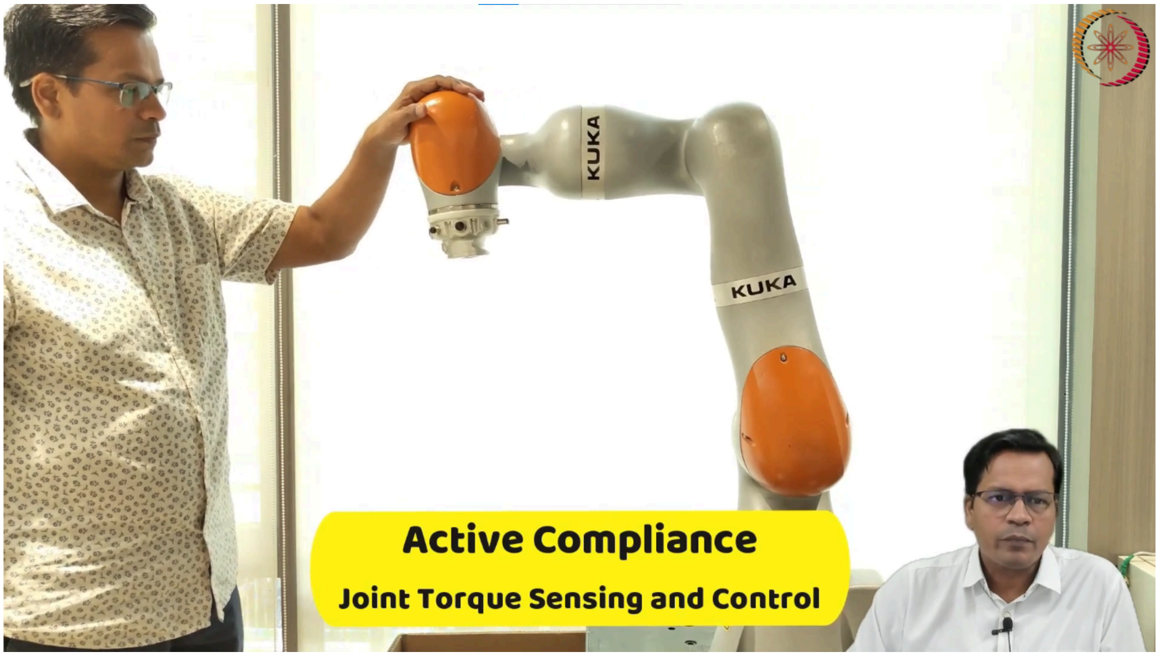
- Passive Compliance: Universal Compliance Compensator
- Force of Position Controlled Industrial Robots: with Wrist Force/Torque Sensor (External Loop)
- Dynamic Current Limiting and Joint Compliance: Human-Robot Interaction
- Current/Torque Limiting: Robot-Robot Collision
- COBOT: Joint Torque Sensing and Active Cartesian Stiffness/Impedance Control
- COBOT: Human-Robot Interaction, Multiple Contact sensing, Touch Sensing and Interaction, Touch Recognition



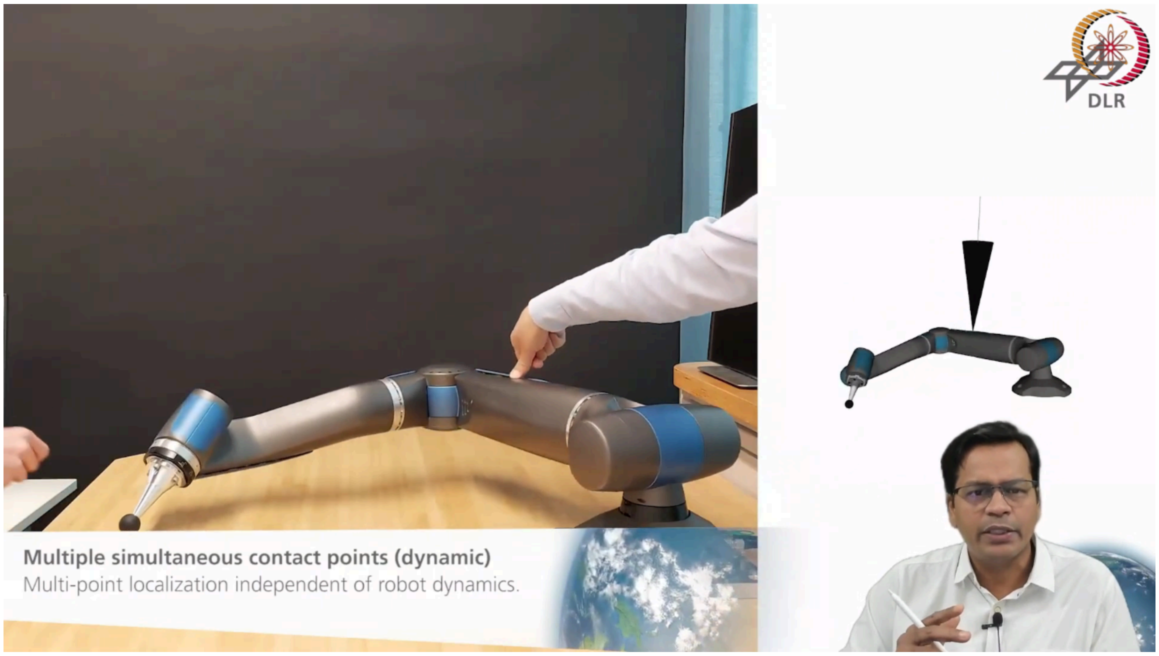
So, let us just watch a small video here, which will elaborate upon the transition that has happened. So, in the left side video, you see there is a compliant structure, passive compliance. That is an ATI device, and on the right side of it, you see you have a video that shows a wrist sensor with a gripper that comes at the end. So, in between the robot flange and the end effector, you have four sensors sandwiched here. So, that you create a closed loop and make things compliant. I am able to handle the robot using my finger touch.



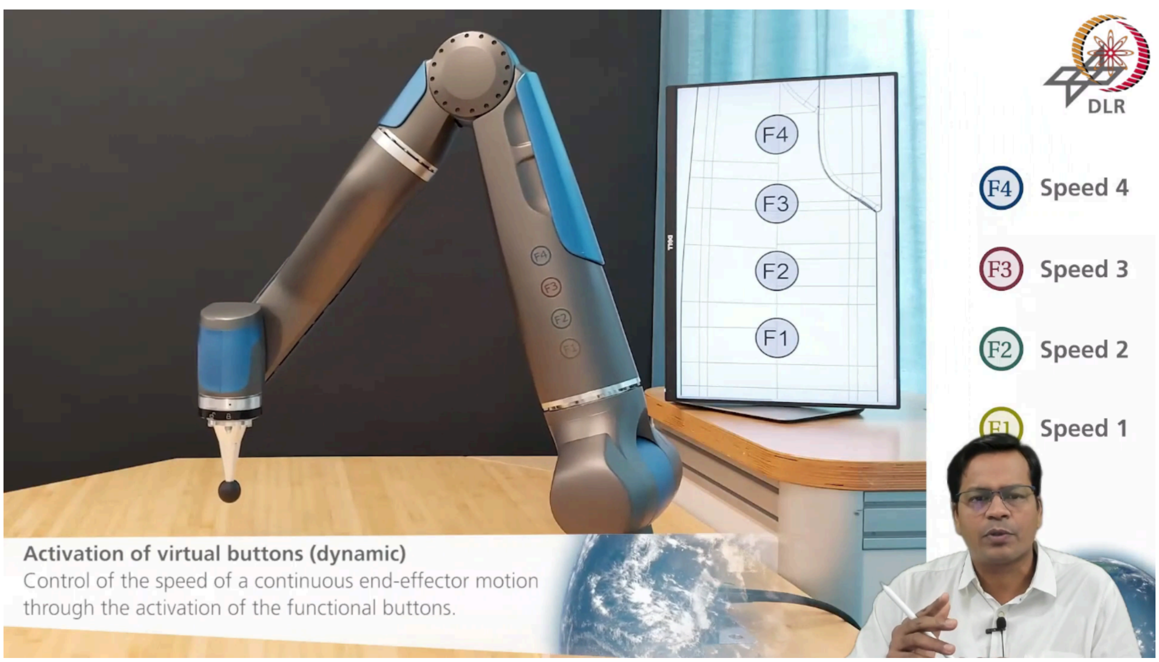
This is again a dynamic current limiting when human-robot interaction is shown, and the right side you see it is having a collision, and still the joints are compliant. In a way, I am trying to limit the joint torques that can be delivered out of some current. So, essentially, I am limiting the current at the joint, and I will able to make the robot compliant. So, even in industrial robots, joints can be compliant, but this is only for safety purposes, not for precise force control. See, I am able to move the Joint one by one. So, each and every joint has limited current that can flow through it. Even the base joint, which is one of the heaviest joints which is there in the robot, I am able to move it. So, this is reported in one of the publications that I have shown below.



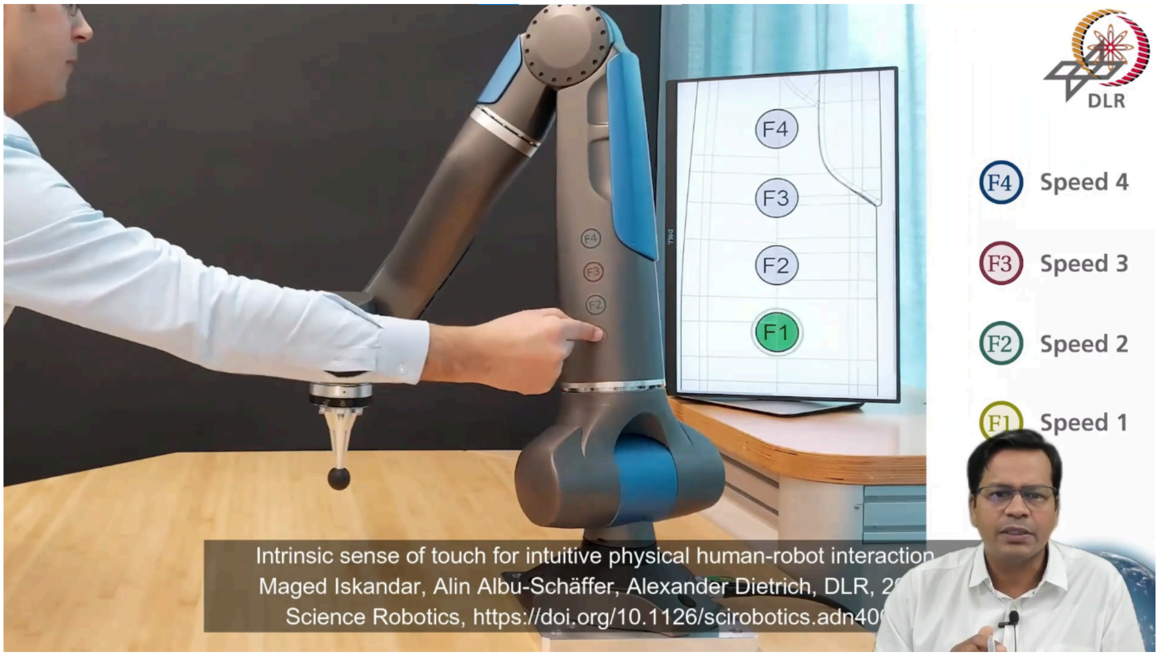
Active compliance- this is the most modern Cobot that you see here. It uses active compliance. It has a joint torque sensor at each of its joints. All the joints it has it. So, with this impedance controller and end effector force control, you can do things like this. So, the whole of the robot is compliant, not just the part of the robot that comes after the Wrist is compliant, even before that whole of the link is compliant.



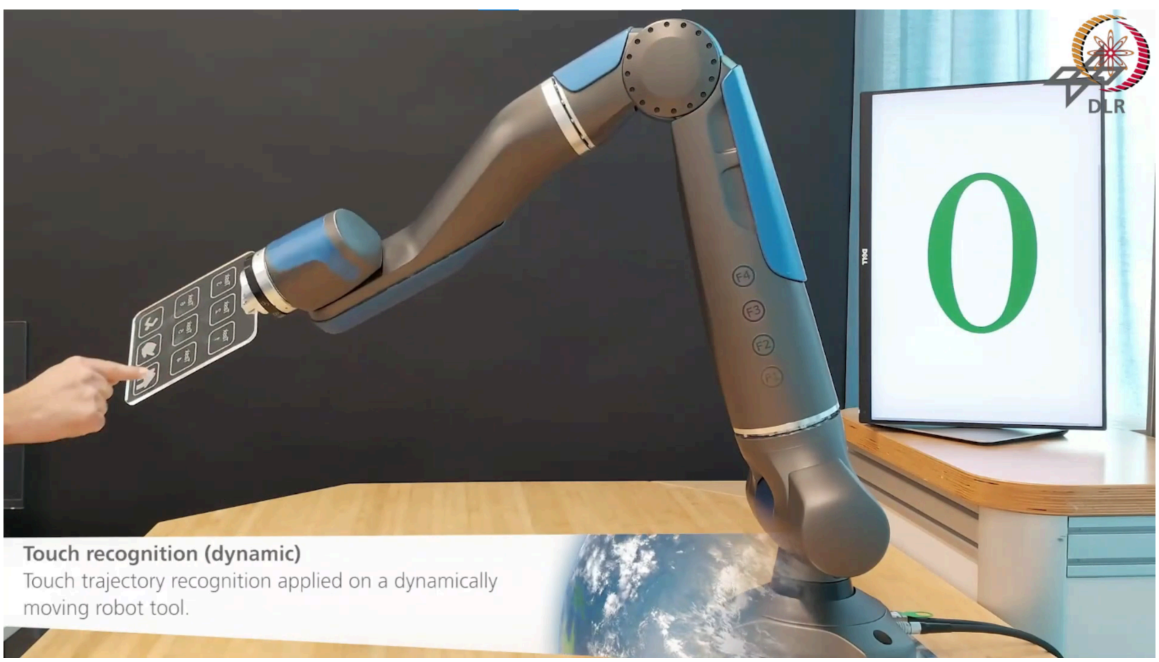
This is multiple simultaneous contact points that are reported. This is the latest one in 2024 it has come.



It is an example of learning sensorless-based approaches.

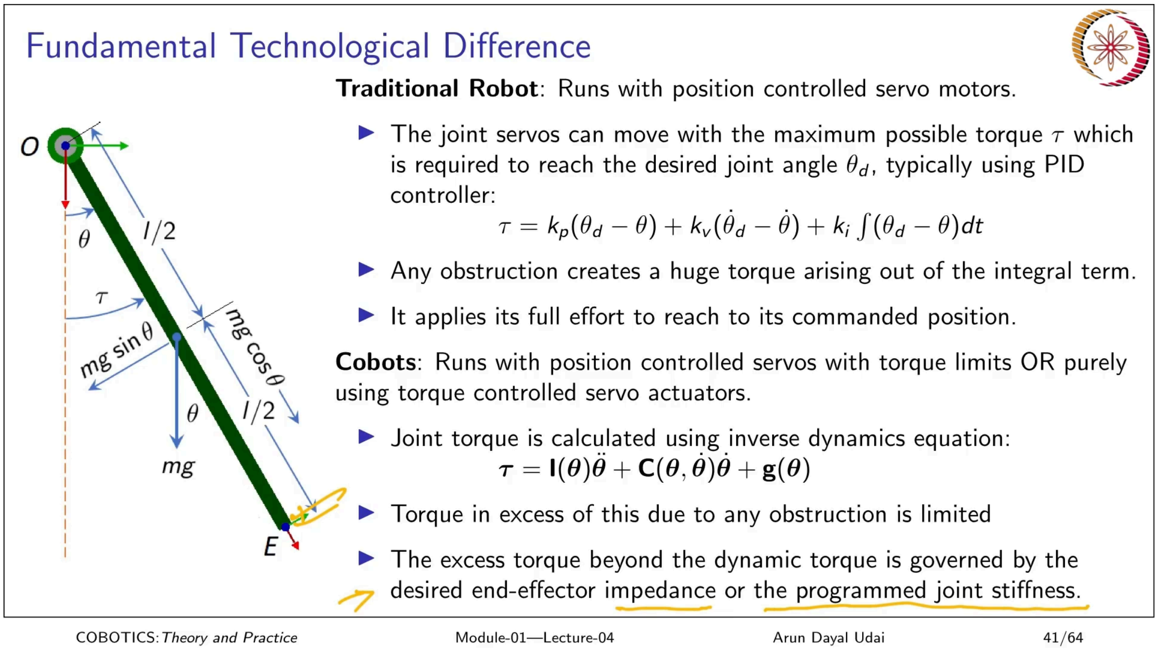


When a person is able to detect the touch even on the surface of the robot's body. So, precisely it is trained using the sensors which is already there in the robot.



And it can detect the sense of touch. So, touch sensor capability is reported, and even the end effectors there that are compliant joints are compliant. You can even have something at the end or anywhere on the robot you can write something, and it recognises it as a character if you are writing it. So, it is a completely intelligent thing that is reported now.

So, this is where we are now. This is one of the latest things which is there in Cobot. So, apart from having the whole of the robot compliant, now it also has tactile senses. It is becoming very near to humans nowadays. So, these are some of the videos that I have shown.



So, what fundamental technological difference is there? So, let us start with a small link like a pendulum with mass m link length l. So, the centre of gravity is situated here at l by two distances from the top. This is the axis of rotation, theta ($\boldsymbol{\theta}$) is the joint angle, and tau ($\boldsymbol{\tau}$) is the torque at any instant of time. So, you see, mg is the force that will come here, and g is the acceleration due to gravity. So, this is the model I will use to explain the fundamental difference. So, normally, traditional robot, traditional industrial robots, they run with position control servos. That means at the joint, they have an encoder, or the joint sensors joint angle sensors that continuously monitor the joint angle. Keeps it in a closed loop and creates a PID on top of it. So, that works with joint angular error ($\boldsymbol{\theta}_d$ - $\boldsymbol{\theta}$). So, this is the joint angular error multiplied by k_p, that is the position gain. This (k_v) is velocity gain. So, this ($\dot{\boldsymbol{\theta}}_d$ - $\dot{\boldsymbol{\theta}}$) is a joint angular velocity error desired versus actual. So, the left one ($\boldsymbol{\theta}_d$) is the desired, right one ($\boldsymbol{\theta}$) is the actual. Same here, this ($\dot{\boldsymbol{\theta}}_d$ - $\dot{\boldsymbol{\theta}}$) is a joint velocity error multiplied by k_v . ($\boldsymbol{\theta}_d$ - $\boldsymbol{\theta}$) joint angular error integrated over a period of time multiplied with K_i that is an integral gain. So, this essentially is a PID controller that provides the torque. So, torque is an outcome of joint angular error. As long as the error is there, it is going to provide continuous torque to the robot joints. So, what happens if there is any obstruction anywhere over here? This continuous torque will provide continuous current to the joints, and the motor may get damaged or too much heated up. So, any obstruction creates a huge torque arising out of the integral term. So, if the error remains there for a longer amount of time, So, an error integrated over a period of time creates a huge amount of torque and current at the joint. So, this is bad for the robot joint that is for the actuator or the motor if it is there. It applies its full effort to reach to the commanded position, that is the desired position. So, if there is any obstruction on the way so that, the robot may get damaged, or the tool may will get damaged, or the environment will get damaged, motors may get damaged. So, something wrong will happen. So, it applies its full effort to reach to a place because it only works with joint position error irrespective of the torque which is going to come at the joint. It doesn't bother about the torque.

$$\boldsymbol{\tau} = k_p(\Theta_d - \boldsymbol{\Theta}) + k_v(\dot{\boldsymbol{\Theta}}_d - \dot{\boldsymbol{\Theta}}) + k_i\int(\Theta_d - \boldsymbol{\Theta})\,dt$$

So, whereas in the case of Cobots, it runs with a position-controlled servo with torque limits. So, it also cares about the torque, which is here ($\boldsymbol{\tau}$). It continuously keeps track of torque by way of measuring the current, or sometimes it physically has a torque sensor, strain gauge torque sensor or maybe encoder difference kind of torque sensor; it is there at the joint. With that, it comes to know the actual torque at the joint, and it precisely controls the torque, or sometimes it limits the torque. Both are equally okay. Purely using the torque-controlled servo actuator. So, you can continuously keep track of the torque which is at the joint and create a closed loop.

So, joint torque is calculated using the inverse dynamic equation. Mind it, this equation will be derived while we will be doing Cobot dynamics and even this PID controller and all will be discussed very much in detail towards the end of this course in control force control thing.

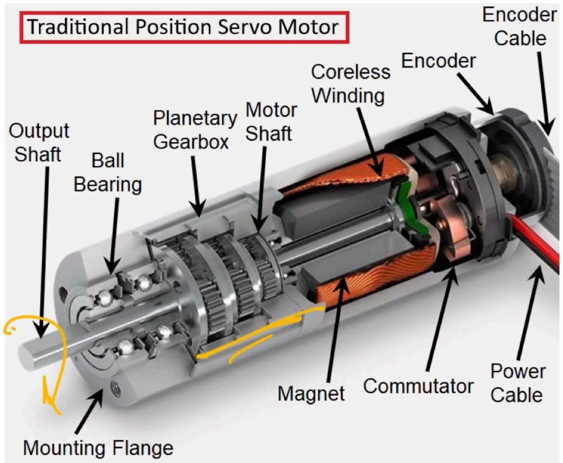
$$\tau = I(\Theta)\,\ddot{\Theta} + C(\Theta,\dot{\Theta})\,\dot{\Theta} + g(\Theta)$$

So, yes, this torque goes to the robot joint corresponding to the joint acceleration, joint velocity and joint angle that you want. So, this is the torque that should go to the joint. Even when the robot is stationary, these first two terms goes to zero; only the last term, that is due to the gravity torque continuously has to go to the joint in order to stay there. Because brakes are not applied, and the robot is free to swing like a pendulum. Okay. Like a serial chain pendulum. So, in that case, this much of torque continuously goes to the robot. So, now, with this torque, if it is in a closed loop and the robot is made to run only with the torque, what happens, any torque in the axis due to any obstruction is limited. In order to go there, how much torque you need is mg sine theta into l by 2. That much is torque is required to be here at this angle.

So, let us say the robot started from here; it started going like this, and in between, it had a collision. So, what happens? It demands more torque than it is required to be there. So, that excess torque this robot cannot provide or a little bit of higher side it can provide. So, in that case, the torque error is very, very less. The desired torque and the actual torque are different, and then in that case, because it cannot provide that much of torque, the robot stops there, and while stopping, there is a very small amount of current that goes to the robot joint to maintain that position okay and that makes the robot safe the motor safe and the collision because it is not surveying for the position, it is not applying its full effort to reach to the commanded position. Okay. It only applies the amount of torque that is required to go to a particular place, not to the full effort.

So, The excess torque beyond the dynamic torque is governed by the desired end effector impedance or the programmed joint stiffness. So, how much the stiff I want my joint to be how much end effector force that should come, that determines how much torque I am going to feed it with. So, this is why these Cobots are very, very safe, and this is the fundamental driving difference of a Cobot and a traditional robot.

Hardware Difference: Joint and the Inbuilt Actuator Design



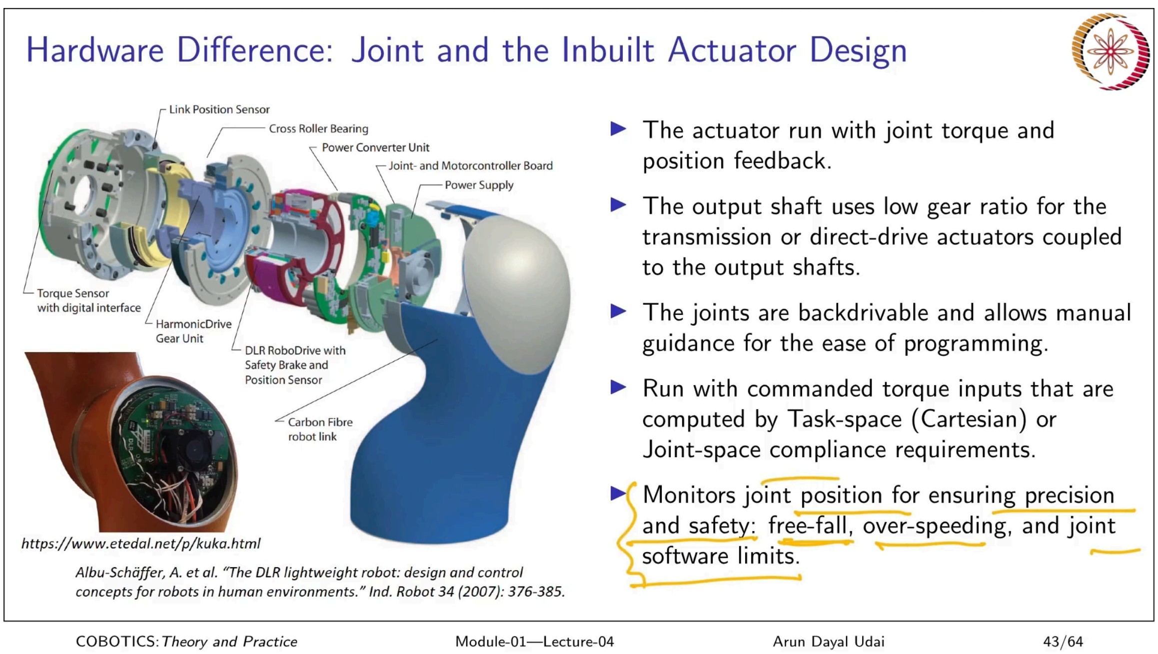
MAXON® DC Servo Motor

- The actuators run with joint angular position feedback.
- The output shaft uses high gear reduction ratio for transmission to increase the torque output and reduce the angular speed so as to make it more controllable.
- It has very poor load side to motor side torque reflectance and hence cannot detect link (load side) disturbance.
- Uses current based torque limits for safety.
- High gear ratio makes the joint non-backdrivable (stiff) and are not compliant to any external forces/moments.

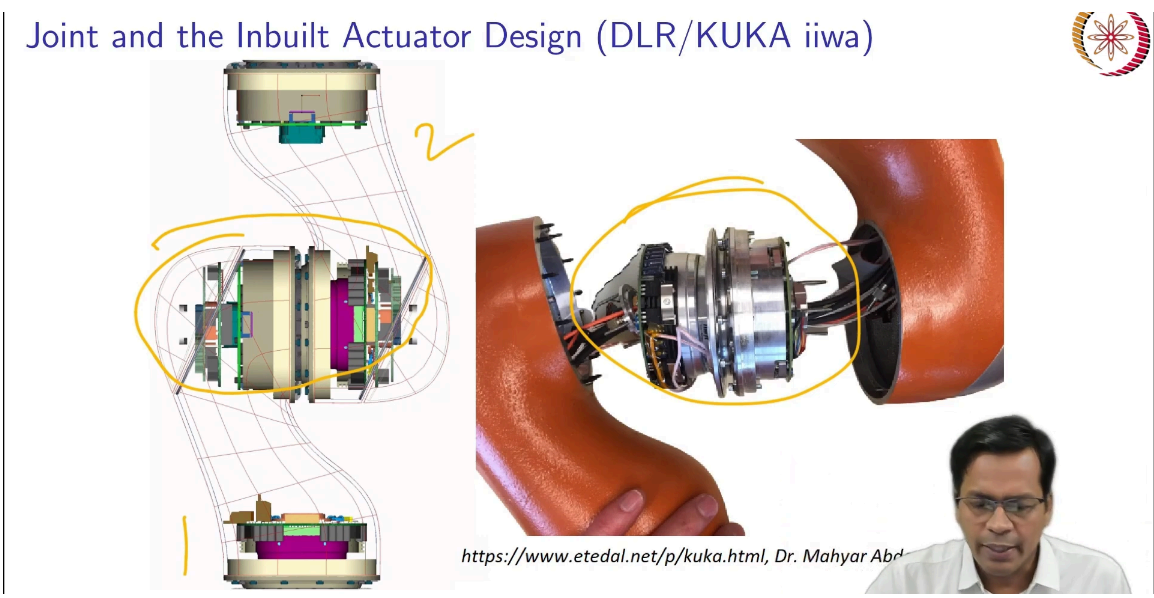
So, this is how the joint is built for a standard traditional positional servo motor. The motor which is here is shown here. This is a DC it shows a commutator also. So, it is a winding motor with permanent magnets here. So, this is a kind of DC motor permanent magnet DC motor with a commutator. So, this is the motor part of it. This is the gearbox. These are the bearings and the output shaft. At the rear part of it, it has an encoder here, a power cable and the encoder. So, essentially it gives the feedback of joint angles through the encoder. So, only position is taken into account. What else you can read is the current that flows through this actuator. So, you can take care of the current also, but yes, You cannot precisely control it because there is a gearbox box here that has useda lot of friction and noise. So, we will talk about this. So, the actuator they run with joint angular position feedback given by the encoder.

The output shaft uses a high gear ratio for transmission to increase the torque output and reduce the angular speed. Reducing the angular speed makes it more controllable. So, this is the reason for putting a gearbox over here. It enhances the torque and reduces the speed. It has a very poor load side, that is, this side where the link is fitted to the motor side torque reflectance. So, if there is any disturbance that comes here because of the gearbox that is here, you see very little change in the torque that is felt to the motor shaft, and that is poorly reflected by the current that flows through the power cable of this motor. So, you can detect a huge change, but definitely not a small change. That may be because of any human collision or maybe a small obstruction if it is there. So, you can take care of the current only to ensure the safety. So, for any huge current surge, you can apply the brake or you can switch off the motor. So, this can be taken care of. But definitely, you cannot precisely control the current to control the torque that comes here. So, this motor itself does not support this to be in a Cobot. It can only do position control.

Next, it uses current based on torque limits only to ensure safety. A high gear ratio makes the joint non-back drivable. So, these joints you cannot apply torque here to drive back the motor. It becomes back drivable due to a huge amount of friction, which is here and they are not compliant to any external torques or moments that act on the robot joints or at the robot tip or somewhere on the robot. So, essentially, the hardware is different.

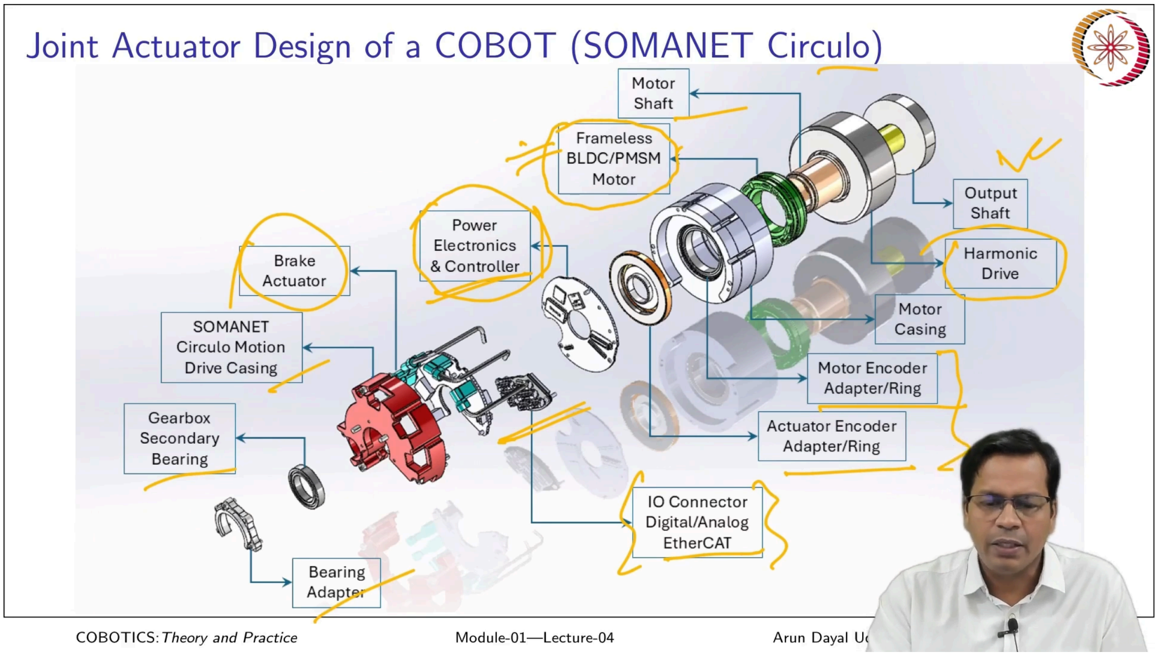


Now, coming to the hardware of a Cobot. So, how does it look like? It has a position sensor. It also has a torque sensor. So, it has both. It can take care of torque in a closed loop as well as it can form a closed loop around a position, so that it ensures precise positioning taking care of the torque also. It has a low gear ratio HarmonicDrive, which is her,e or sometimes it is direct drive also, power converter unit, microcontroller board which is here, that creates a local closed loop also power controller board is here, okay robot drive with position sensor is here. So, the whole of this set also you see it is a center hollow, so that you can pass through the wires from within. All the wires pass through from within the robot arm. So, this is the beauty of having such a nice actuator. So, this actuator can run with joint torque and position feedback. Both the output shaft uses a low gear ratio for the transmission. The gear ratio of this is very, very low, or sometimes it uses a direct drive coupled to the output shaft. There is no gearbox at all. So, it makes it completely back drivable. The joints are back drivable this allows manual guidance. For the ease of programming, lead through programming and those. Okay. It runs with commanded torque inputs that are computed by the cartesian or the task space compliance that you want or the joint compliance requirements. How much stiff should your robot be, depending the on that, you calculate the torque you put in only that much of torque to the joints. So, finally, you also can monitor the position to ensure precision and safety. Safety against free fall because you are continuously surveying the motor brakes are released if at all anything. Let's say you have provided less torque to be there it can freely fall like a serial chain like a pendulum. So, it can detect free fall if you can continuously monitor the joint velocity, any over speeding or joint softer limits. So, if you do not want a joint to rotate beyond certain limits, plus or minus, you can stop that. So, these are the reasons why you are also using a position sensor. So, it has a dual loop of torque as well as position, velocity, and everything. We will talk about that also.



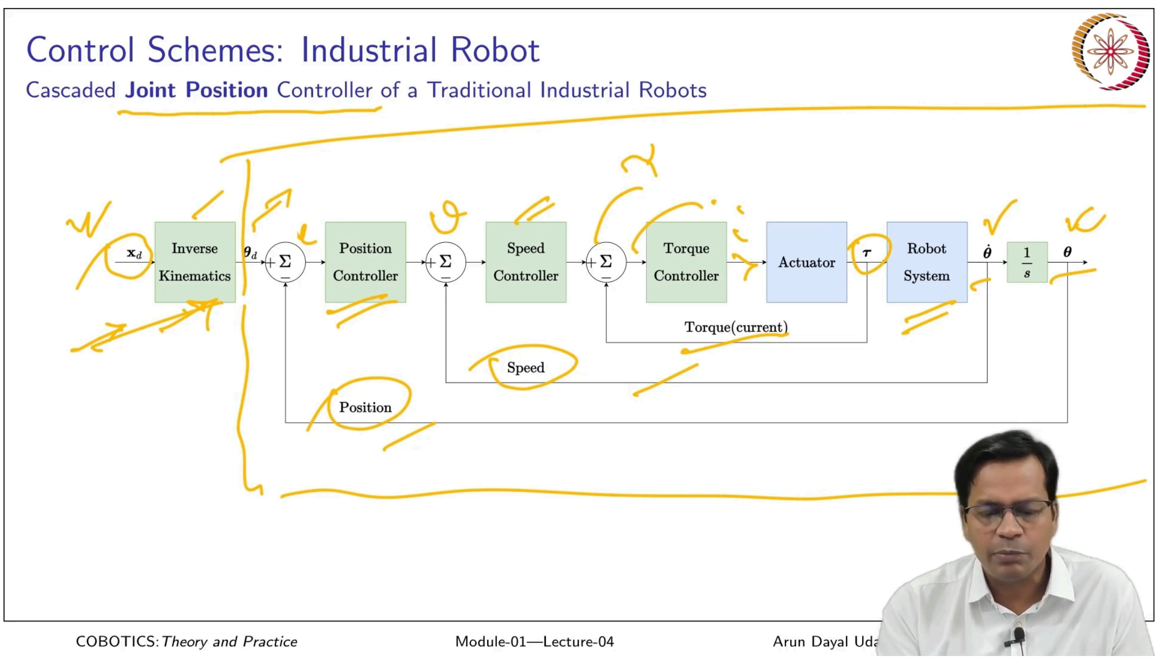
So, you see, the whole of this package goes here. So, you have the first link that comes here, and the next link goes here,e and in between, you have the whole of that package.

This is so nice here when both the links are disassembled, and this is there in the block that I have put here. You can go through that.

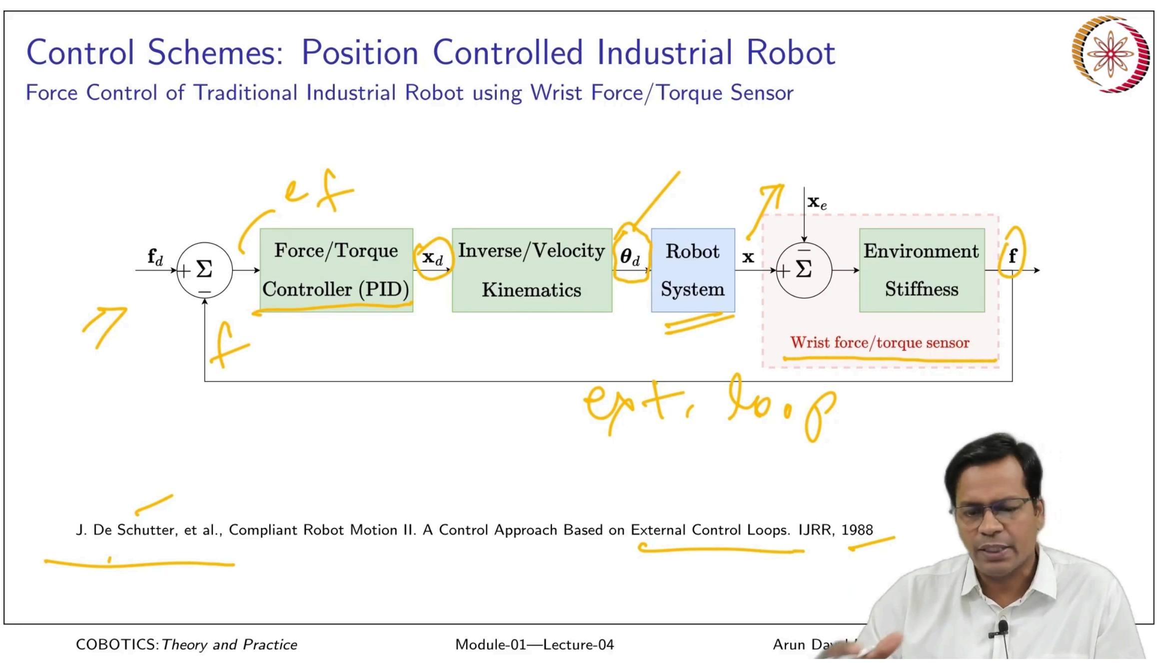


So if you disassemble the whole of this Cobot joint, over here it is a somonet circular which I have shown. So, you see, it has a Frameless BLDC or BLDC that is a brushless DC motor or permanent magnet synchronous motor if it is an AC motor. AC or DC kind of Cobot. So, this is a frameless motor; you cannot have a packaged actuator like industrial robots. You cannot put it like that because the whole of this system has to be very, very light in order to reduce the inertia so as to control any collision damage. So, these motors are inbuilt inside the robot casing. The link itself you saw here link itself provides the housing for this assembly. So, that is why frameless motors are used. The stator and the armature set are there that goes here that is here, and then the motor shaft is designed on top of this harmonic drive that also goes to the casing and finally the output shaft is here. You have encoder here, you have encoder here. agai,n two encoder differences can be used to find out the torque which is there at the joints. So, this can also be used also input and output side of the gearbox if you put. If you know the stiffness of the gearbox, you can calculate the torque based on the encoder readings. This is a primary way of doing it, but it also has a live torque sensor, which is based on the strain gauge or kind of different technology maybe okay. So, that is always there.

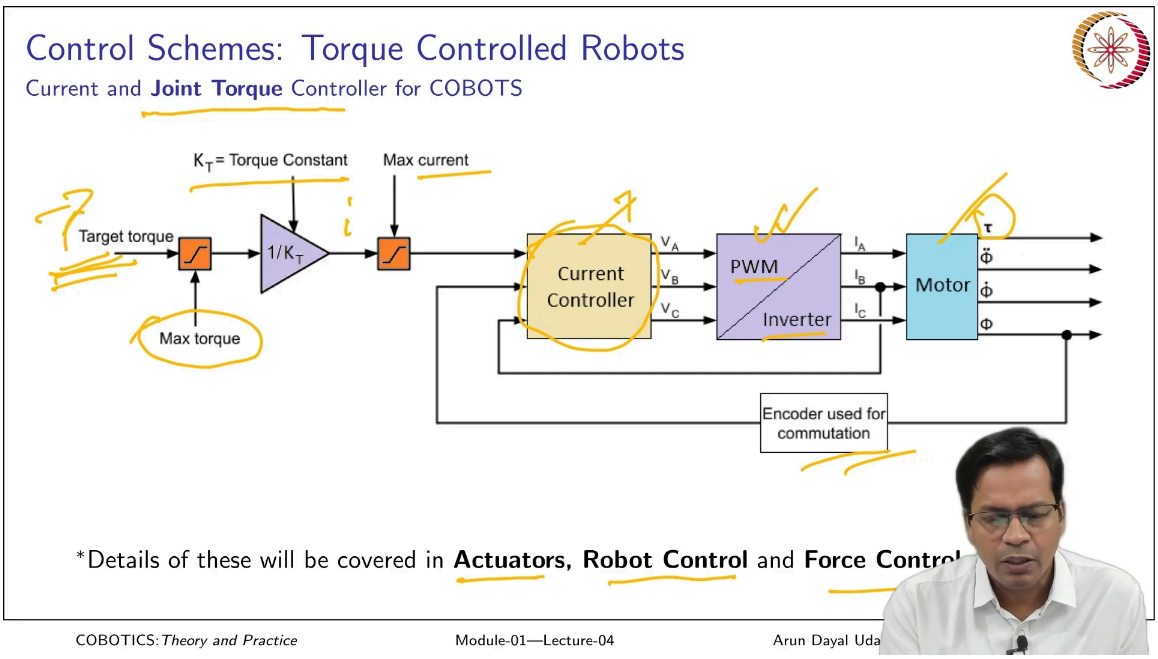
So, this is the IO connector digital to the analog connector and EtherCAT, which is the kind of industrial communication that is there to communicate the data to and fro from the robot joints. So, this joints, all the commanding is done to this okay. Then you have a brake actuator which is here. Power electronics and the local controller which is here. This controller ensures the command to be executed, the amount of torque or amount of current that you want to go to the robot or to the position closed loop. All the loops are closed here. Then, the external PC-based controller takes care of the algorithms and the kinematics. This motor itself has a local closed loop that ensures the position and the torque. So, you have accessories like bearings and all drive casing is there. So, the whole of this assembly is there in the Cobot single joint.



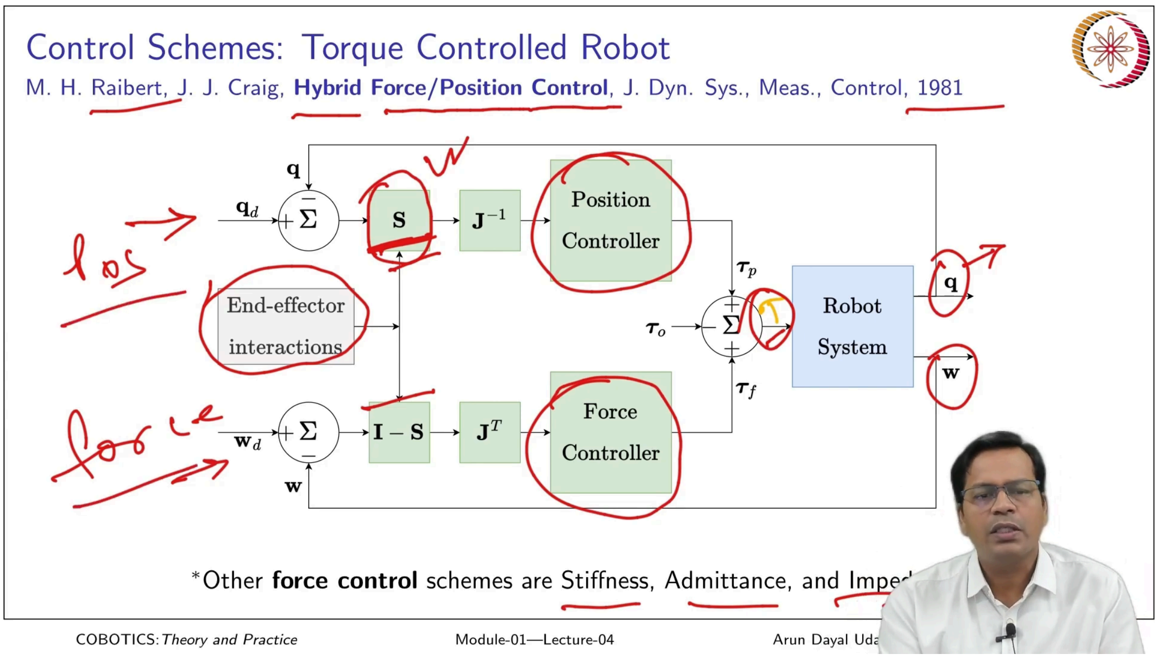
So now, the traditional industrial robot they have a Joint Position Controller. It used to run like this, you see, for the traditional robot, what you want is the end effector position (X_d). We use inverse kinematics here. We will talk about each independent block later on in our course also, like inverse kinematics, and forward kinematics. We will be dealing with this. For now, you just get to the philosophy of this control. So, for the given end effector position, you calculated the joint angles (Θ_d). The actual angle is as feedback from the joint, and you calculate the error in angle, go to the position control, and finally you, what you get is a reference velocity here (v). That velocity is compared with the actual velocity. This is the speed controller. This generates a reference torque here (**τ**). This torque and the actual torque are compared, and an error in torque is calculated that finally sends the current to the actuator. Okay, and finally, the actuator provides the torque, and that goes to the root system. For the output of the robot, you have a velocity sensor, joint velocity sensor and joint angle sensor. So, the velocity may be derived out of the joint angle only. So, these two are the feedback lines. So, this is how this whole closed loop is working. So, it is a cascaded controller where input is the joint angle essentially. So, this is my actuator. This is the system which calculates based on the kinematics if it is there. So, based on the kinematics, it does inverse kinematics, calculates the joint angle and finally whole of the closed loop works on the joint angle feedback. We don't have much control over the torque. Only we can limit this torque, if at all there is any torque limiting feature which is there in the actuator. So, this is the structure of the controller.



Now, if at all you want to do force control with such kind of actuator, a robot that is built with this kind of actuator, so this robot demands what are the joint angles. So, this is ensured by the joint angles are ensured by the joint actuator. So, the joint angle is commanded, and the robot goes to any XY position, and you have what is a wrist force torque sensor. You get the feedback of force here. So, this is an external loop of force that comes here, calculates the error in force goes to the force control PID loop that calculates the desired position depending on the force error. So, this requires an end effector or the wrist force torque sensor. So, the whole of the robot is stiff. At the end you have a force sensor you can control forces after the wrist. That is at the tip of the tool may be. So, that is where you can control the force. So, this is how it works. This work is reported here using External Control Loops in IJRR 1988 reported by J. De Schutter. So, this is how an external force loop can be used to control the forces at the end effector using a standard industrial robot that uses a joint position control servo.



Now next. So, if at all, it is a joint torque-controlled actuator. So, what it has basically it can take in the commanded torque also. So, torque is commanded here. Checked for the any maximum limit. If it should not accept that, multiplied with 1 by K_T. K_T is torque constant. It calculates the current reference. Current again, it is checked for the maximum current that can go to the So, it is limited here, and then this current is finally controlled here, and accordingly, the PWM inverter calculates the phase input that finally goes to the motor. It can be a BLDC motor, or it can be a PMSM motor depending on that. So, essentially it is controlling the current based on the torque input. So, torque input that becomes the torque desired and finally,y that torque is achieved at the output of the motor. It also keeps track of the joint angles for internal purposes to ensure the actual position if it has achieved or not for other purposes, maybe for safety and free-fall detection. So, the entire details of this again will be covered while I will be discussing actuators in the next module. Robot control and force control those things when I will be discussing in those sections; I will explain very much in detail. So, the philosophy here is you can command for a torque, and you can achieve a particular torque. In between, you can control the current. This drive can be of any type. It can be a DC motor also. So, this is the control scheme for the torque control actuator that goes to the Cobot.



So, the control scheme for using such kind of robot. So, this robot accepts torque at the joint. So, that means this robot should be built with an actuator that accepts the torque command, not like a position command, as we saw here (Refer to slide Control Schemes: Position Controlled Industrial Robot). This robot was accepting the position command here. So, now this torque goes here and you see this is given by Raibert and Craig in 1981 that shows Hybrid Force and Position Control. Basically, in this controller the task space of the robot was split in two, that is, position and force control or orthogonal spaces. Let us say if it is, if it needs it can put force control along this direction and the remaining two directions, it can have position control. So, this is a compliance selection matrix that detects the end effector interaction accordingly. This is the position command, which goes here, and this is the force command that comes here and compliance selection actually if it is selected for one of the axes, if it is force or position, the other one should be the position. If it is force over here (I-S) it is position over here (S). So, you have a position controller here, and this is the force controller, and together, it sends the torque to the robot, and the robot runs. So, there is feedback on the position feedback of the forces. Forces here, I mean force and the moment, and for the position, I mean here, the position as well as orientation. So, it has two orthogonal spaces, one for the force and the other one for the position. There are many such schemes like stiffness controllers, admittance controllers or impedance controllers which are there that can do such things and very much in detail again I will be covering in the force control module.

For now, the philosophy is you can control the end effector, that is, the task space forces, as well as the position for two different inputs along those two orthogonal directions. So, this is a kind of scheme that is used to do that okay, and this uses torque-controlled actuators. So, the robot is made up of those actuators.

COBOTS Mechanical Arm Design

- Lightweight: Generally cast aluminum/composites/plastic structure (for safety against collision with environment/humans around)
- Concealed wires: No external wiring (electrical or pneumatic)
- Soft and rounded edges. Better aesthetics in anthropomorphic robots.
- Inbuilt self calibration system and mechanical stops.
- Safe design to avoid pinching and trapping hazards, in compliance to ISO 13849 and ISO 10218.
- Electromagnetic compatibility (IEC 61000 - Immunity and Emission).
- Low Vibration & Noise ($< 65dB$ normally and $< 40dB$ for Medical).
- IP Rating (IEC 60529) depending on application.
- Provisions for **heat dissipation** in the design.

* This definitely makes the system more complex and difficult maintaining.

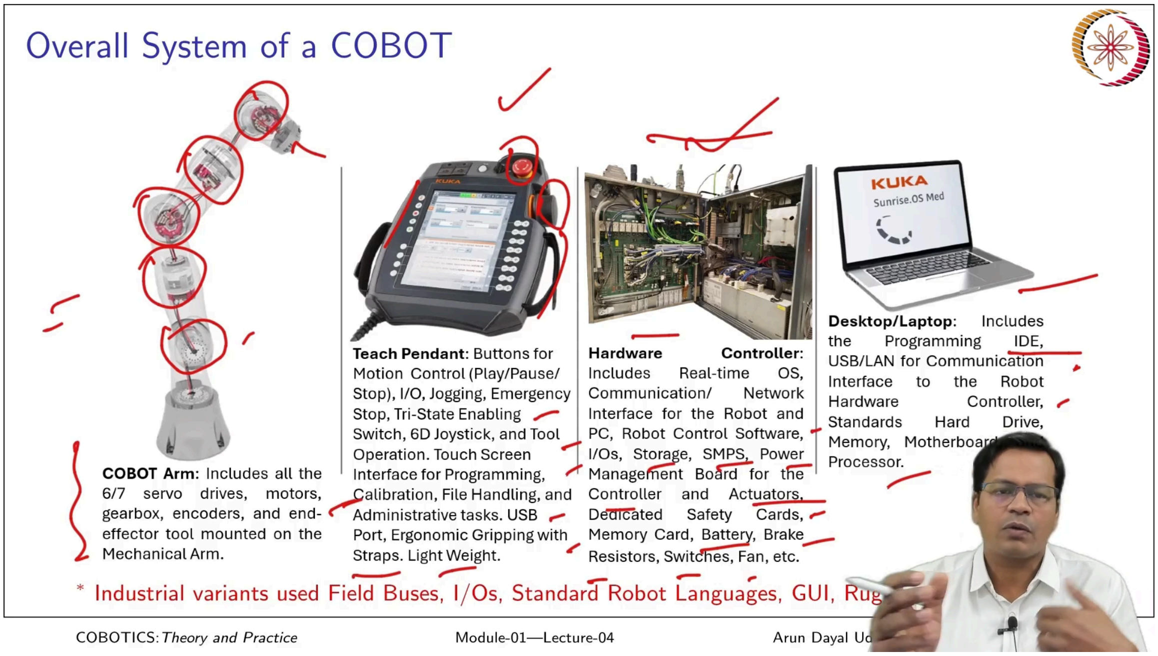
COBOTICS: *Theory and Practice* Module-01—Lecture-04 Arun Dayal Udai 50/64

So, now, Cobot mechanical arm design- so what it goes to design such arm It should be a lightweight arm. Generally, it is cast with aluminium, composites, and plastic structures for safety against a collision with the environment or humans around it. So, the surfaces are very, very soft because of these materials. It also has concealed wires because nothing should get entangled with those wires because humans are around. It can catch up a human organ or maybe a cloth which the person is wearing. So, that should not happen. So, no external wiring is allowed, electrical or pneumatic wiring. Soft and rounded edges- so because of the material, it is soft. It also should have a design that is very much aesthetics, and it should be rounded edges. So, nothing can get trapped into the edges of the robot as well. So, it looks much better aesthetics. In the case of anthropomorphic robots, it also should look like humans. So, they are even more aesthetically better. Inbuilt self-calibration system- no external mechanical stops are there. They are inbuilt into the robot joint. So, again, those can have edges and places where something can get stuck. Safe design is done so as to avoid any pinching trapping hazard in compliance with ISO 13849, that is the machine safety standard, and ISO 10218, that is the industrial robot or robot safety standard.

Electromagnetic compatibility should also be there as per IEC 61000 against Immunity and Emission. Okay, because this motor should work regularly even in case when you have external emissions if at all it is there due to any transformers around. So, this structure is such a safe structure. Okay, it should not fail because it is very near to the environment where humans are around. Low vibration and noise: we prefer even in the case of industrial robots; they remain within 65 decibels, but if it is in for a medical purpose, it should be less than 40 decibels.

IP rating IEC 60529 depending on the application to application. Ingress protection rating is adapted. How much water and dust it should protect against. So, those are taken care of and provisions for heat dissipation because humans can touch the surfaces of the robots. So, that should not become very very hot so as to make it dangerous. That is quite possible in the case of the industrial robot, but in the case of Cobots, the surfaces are not very hot, and a person can touch it. So, provisions for good heat dissipation should be there.

So, overall, while taking care of all these, this definitely makes this system quite complicated, and it becomes very difficult maintaining the robot. But still, care is taken so as to make the robot very much accessible, all the joints are accessible so that you can do maintenance and all.



So, now, overall, the Cobot system is made up of these. So, basically the arm where you see all the joint assemblies are there. So, the Cobot arm includes all the six or seven servo dies depending on the degree of freedom of the robot, motors, gearbox, encoders and end effector tool mounted at the end.

So, this is the arm. It has a teach pendant which has buttons for motion control, play pause, stop, input and output control jogging emergency stop tri-state enabling switch for safety that switch has got three states If you press it hard, it is off. If you release it, it is off. In between, it is on. That is a kind of tri-state enabling switch which is there. 6D joystick, which is here. Emergency stop you can see here. The play and pause button is here. Touch screen interface may be there for easy programming, calibration, file handling, number of programs that are stored in the robot controller you can see here. You can control that. For an administrative task, a USB port may be provided to copy any programs to this system. Ergonomic Gripping with straps are there, and it is lightweight.

Again, the hardware controller, which actually has all the controllers for the robot, that is the kinematics, inverse kinematics, everything is here. A kind of impedance controller is here. It is not at the joint. The Joint has a local loop for current and torque positioning thing only. The rest of the thing comes here. And it is very much like a PC which is there. So, it has a real-time operating system, communication, network interface for the robot and the PC, robot control software, input-output, storage, SMPS, power management board for the controller, dedicated safety card, memory cards, battery, brake registers, switches, fans, etc. So, all those things goes to this hardware controller and at the external to it, there is a desktop or a laptop PC that is used to program this controller. So, it has a programming integrated development environment. The programming environment can be RHOS, it can be C, it can be Python, it can be anything. It can have a dedicated API given by the robot manufacturer, which you can use it with any programming language. So, it has a USB/LAN for communication interface to the robot hardware controller that is this one. A standard hard drive memory motherboard and the processor. So, this all makes the complete Cobot system. So, industrial variants they use Field Buses for safe and faster communication input and output, Standard Robot Languages are there, and GUI and the platform are very much rugged.

So, this is a quick reference to what all I have discussed. So, it has links for the torque sensor. That is the sensor drive link that I have explained here, I have mentioned here. This is not an exhaustive list. You should not just go by this. You can work on the internet and find many such drives; torques are there. So, servo drives, it is Synapticon I have mentioned here, Somanet-Circulo. A frameless motor it can be a Kollmorgen or Beckhoff kind of BLDC motor also you can put, but I have put a tq group that is a robot drive, which is also used by Kuka iiwa earlier. So, that I have mentioned. Harmonic Drive that I have mentioned here. You can use Harmonic Drive AG, that is a German company that I have mentioned here, but there are many, like Simpu, a company of Japanese company is also making the harmonic drive. So, this is not an exhaustive list again.

So, thanks a lot for this lecture. In the next lecture I will be Covering Constrained Motion Tasks, Distinguishing Features of a Cobot and those aspects.