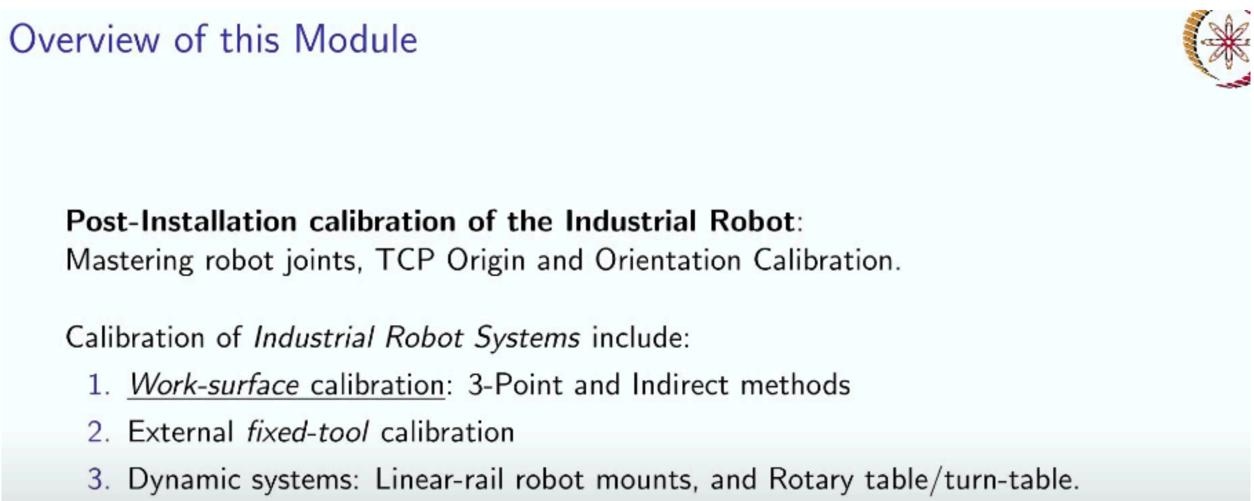
NPTEL Online Certification Courses
Industrial Robotics: Theories for Implementation
Dr Arun Dayal Udai
Department of Mechanical Engineering
Indian Institute of Technology (ISM) Dhanbad
Week: 07
Lecture 30

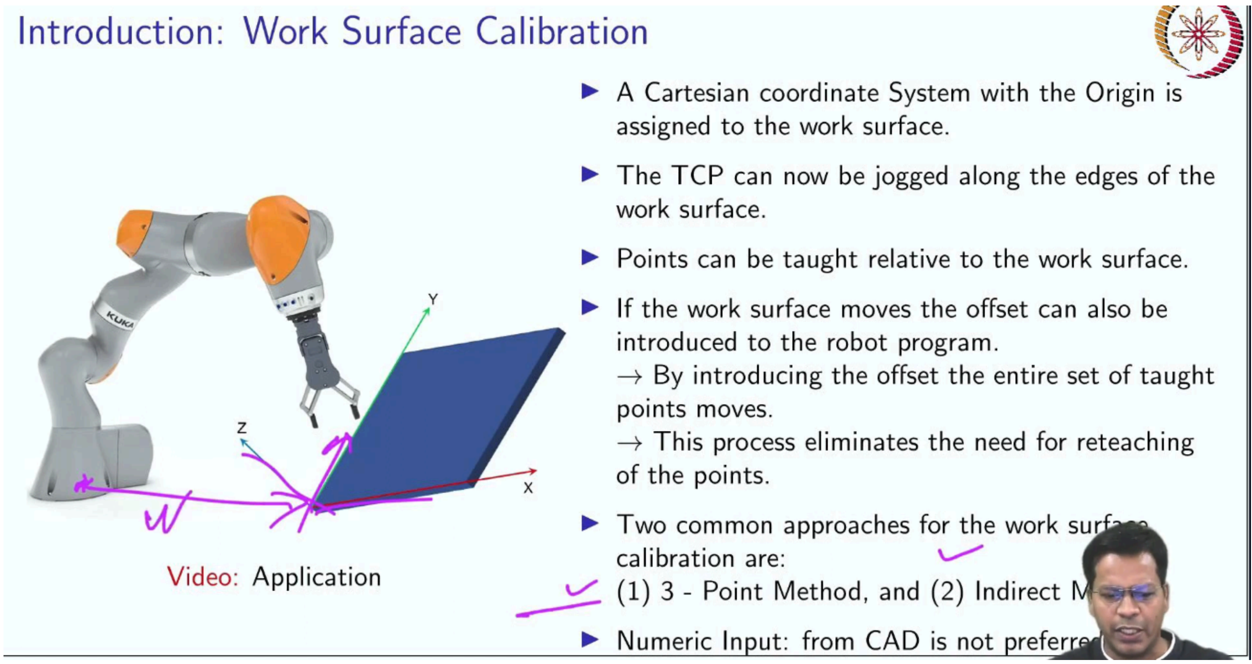
Work surface Calibration

Hello everyone. So, in the last class, we mostly proposed installation calibration of industrial robots' mass-spring of robot joints we did. We also did TCP calibration, TCP tool centre point, origin and orientation calibration that we have done. So, let us quickly look at this module, and what we will be doing in this module. This module is about the calibration of industrial robot systems. Basically, that is, the systems which are associated with robots and their functioning in the workspace when it is implemented in the industry south floor. So, it is surrounded by many other automatic devices and systems which make robots do their functions well. So, let us quickly look at what we are going to do in this class.

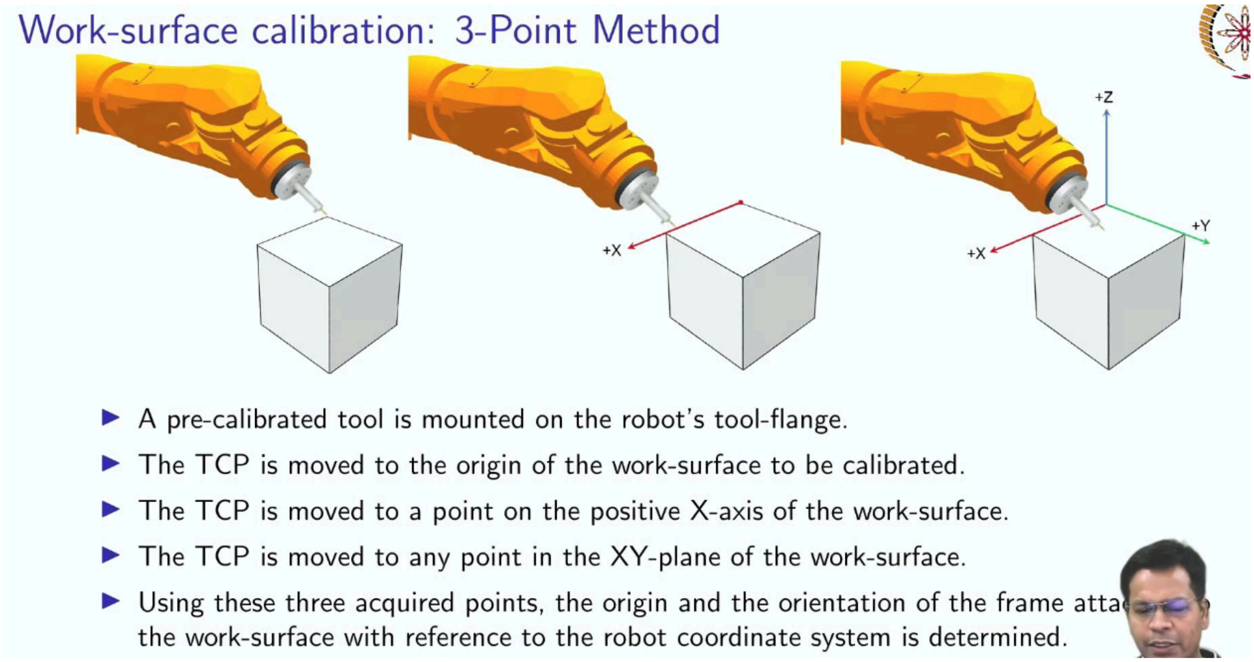


So, this was our post-installation calibration task. We did this with industrial robots, that is, mass-spring of robot joints, TCP origin calibration and TCP orientation calibration. That we did in the previous class, so, in this module we will be doing calibration of industrial robot systems that include work surface calibration. We will be doing this in this lecture. That is today's lecture. That is the point calibration and indirect method. We will closely look at those techniques, And we will be doing external fixed tool calibration. That is, if any tool is mounted in the workspace and the robot is holding the object, maybe it needs to be machined. So, how to handle that kind of environment where a tool is outside, and the tool object is with the robot? So, that kind of calibration we will be looking at in the second lecture of this module. And dynamic systems.

That includes robot mounts, like a linear mount on which a robot can be mounted, and the robot from the base up itself will be mounted on top of it, and it will move on the linear rail. So, there are calibration issues which are involved with that also that we will be looking at in lecture three of this module. We will also be looking at rotary timetables, which are there in the workspace and can run in sync with the other joints of the robot. Normally they are controlled with the industrial robot controller itself. So, that becomes the seventh axis. So, all those issues we will be looking at in lecture three. So, let us move ahead with work surface calibration. That is today's lecture.

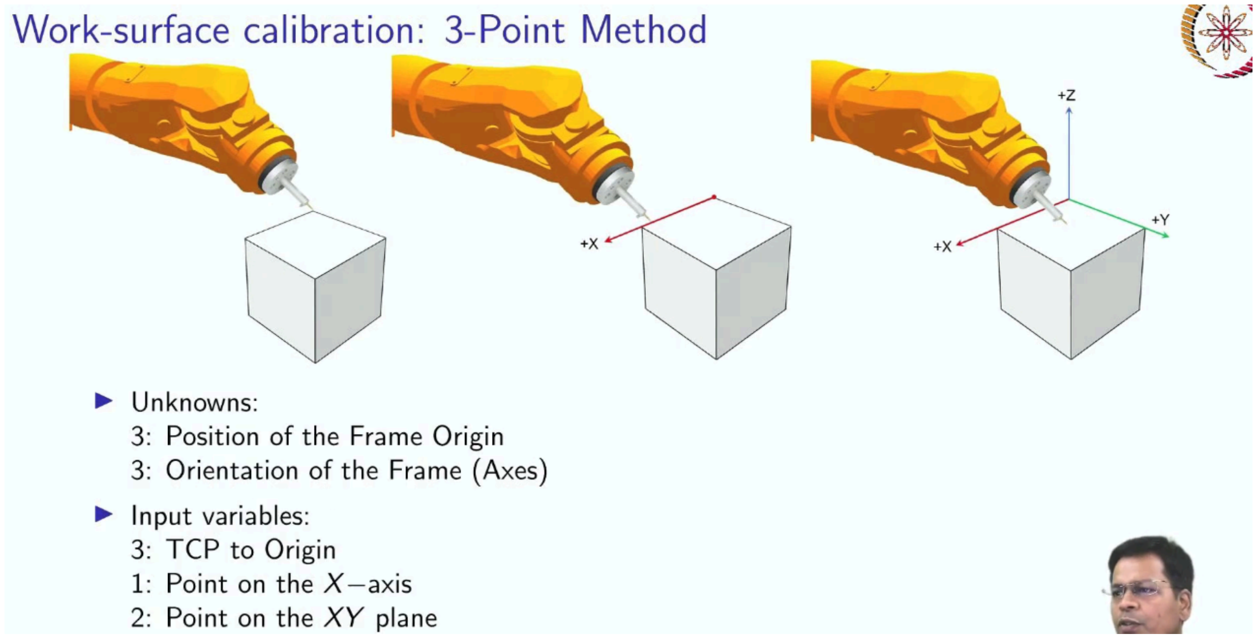


So, what is work surface calibration? A Cartesian coordinate system with the origin is assigned to the work surface. Let us say your robot has to work on a surface which is nearby Instead of having a horizontal flat table. Even a horizontal flat table is a work surface. Different kinds of objects may be placed on the table itself, and the robot is supposed to do pick and place tasks, or it may have to do some array type of job. So, all those kinds of jobs require work surface calibration. It is easy to do if it is calibrated well. You know the frame which is associated with the work surface So that you can put the object with reference to the coordinate system which is there, on the table itself or on the work surface itself. It should not be referred to with respect to the robot base. It will be very difficult to assign those coordinates, and also it will be difficult to understand the coordinates because it is a big robot; most of the time, It becomes very difficult to estimate where my object is, Whereas it is very easy if it is kept on the table with respect to the table coordinate system or the work surface coordinate system. So, that is what is work surface calibration. We will try to assign a coordinate system. If it is a coordinate system, let us say it is attached at the corner of this table. You can have an array of objects which are kept on this work surface, And each one of them is referred to with respect to the origin which is placed here. So, you need to place an origin with respect to the robot base. So, that is our job in today's class, and there are different techniques to do it that will be followed, and TCP can now be jogged. So, you can jog the TCP along the axis of this work surface. If I say move along X of the worktable. So, you can. Your robot goes like this: You can make it move like this, Got it So in different axes which are along the axis of the work surface. The robot can be jogged, or it can be programmed. You can move along the edges of this work surface. So, it is a capability which is very much required in order to program it conveniently. So, if you move with respect to the robot base or with respect to the tool, it becomes very difficult to precisely locate just by jogging, Which is very much required while programming. So, points can now be taught related to the work surface. And if the work surface moves, the whole of this panel, this work surface moves. So, there is a provision. Most of the robots have it. You just need to say how much it has displaced and how much the orientation changes. You have done So, that delta changes. You can just put it into the robot system, robot controller, or robot teach pendant environment, so it quickly understands where is my new origin and how it is now aligned. So, by introducing the offset entire set of taught points which are related to the work surface frame. So, that gets moved, it eliminates any need for reteaching of those points. So, it doesn't need to reprogram if at all, objects are placed on the workspace. There are two common approaches to doing this job. One is the three-point method, the other one is the indirect method. We will start with a three point method now. Other methods could be. Just like the tool calibration we did, you can get this information from the CAD if it is there. So, yes, if you have a CAD which tells this distance, this position vector and the orientation vector of this frame with respect to the robot base, you can quickly do it, and you don't need any sort of calibration. But it is not very advisable to do this because it is a good distance. Even a small amount of error will cause your robot tool tip to hit that surface. So, that is quite probable because this is long, and your error is relatively small with respect to this distance, but it may be quite big as far as your robot's accuracy is concerned. So, it can hit that. So, it is not advisable in case of this.

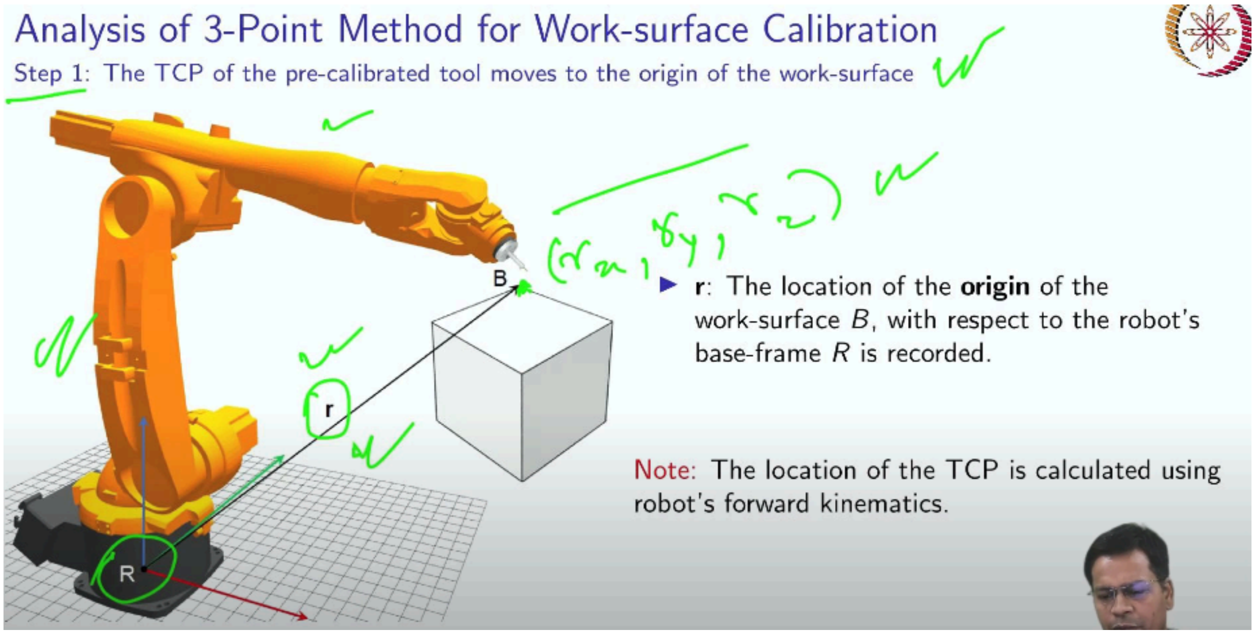


Let us start quickly on the three-point method of how it actually does that calibration. How it actually finds out the orientation and position of your work surface with respect to the robot's base is work surface calibration. So, basically, a pre-calibrated tool is mounted at the tool flange of your robot. It is mostly a sharp tip which can be pointed at any location where you wish. So, that is the first job. So, after doing this, this TCP is calibrated already and your robot forward kinematics can quickly find out the tip of your tool location with respect to the robot base. So, this is the first thing that you have to do.

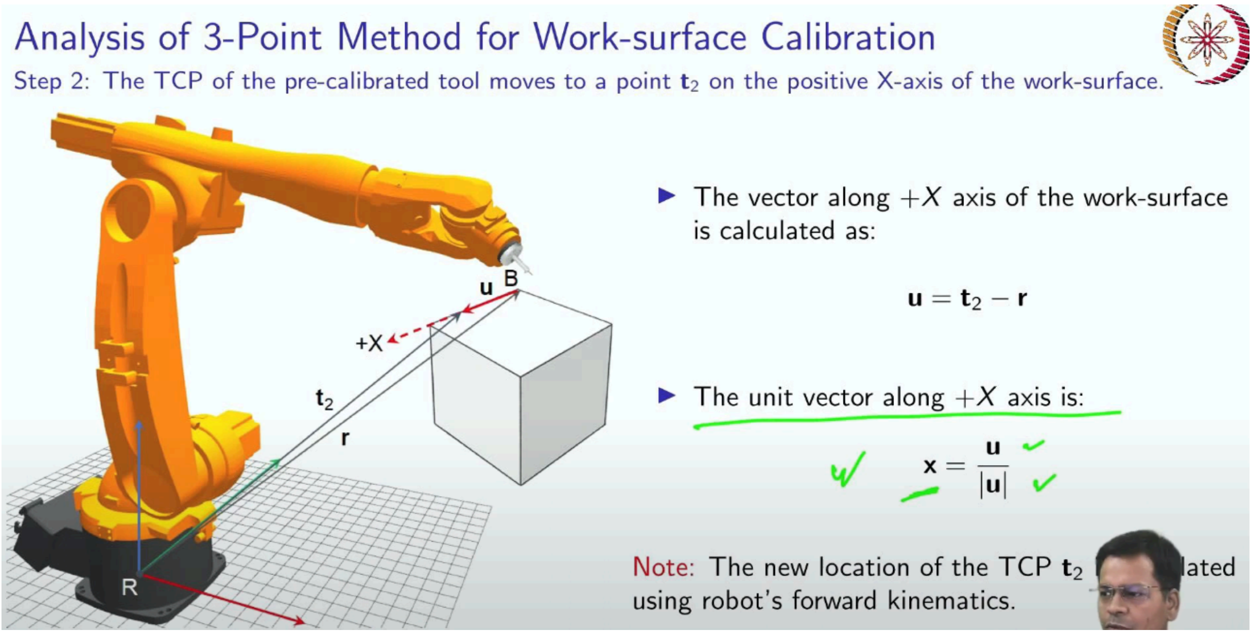
Next, the TCP is now moved to the origin of the work surface. The first thing that you are going to do is you are going to place it at the origin of the work surface. So, if this is your work surface, and you want to make this point as an origin, just take your TCP of the pre-calibrated tool to that particular origin. The next task is now the TCP is moved to a point on the positive x-axis of the work surface. Again, this is the intended positive axis, where you want your x-axis positive to be. So, this was your origin, so this is your direction. So, anywhere you can take your TCP on this axis, this is one of the points that you can put it. Next, the TCP is moved to any point on this plane. This is the XY plane. So, your tip can touch anywhere on that surface. So, one of the points you have to take your TCP to. So, these three steps are involved. First step: you have to put the pre-calibrated tool, or if you have just put a tool, you can calibrate it. That is the first task. Next, you have to touch this origin. The next point is you have to touch one of the points on the x-axis, positive direction, and your final step will be to touch your TCP to one of the points on the XY plane of the work surface. So, using these acquired points, the origin and orientation of the frame, which is attached to the work surface, is calculated by the robot controller, and that is what we are going to learn today: how it actually does it.



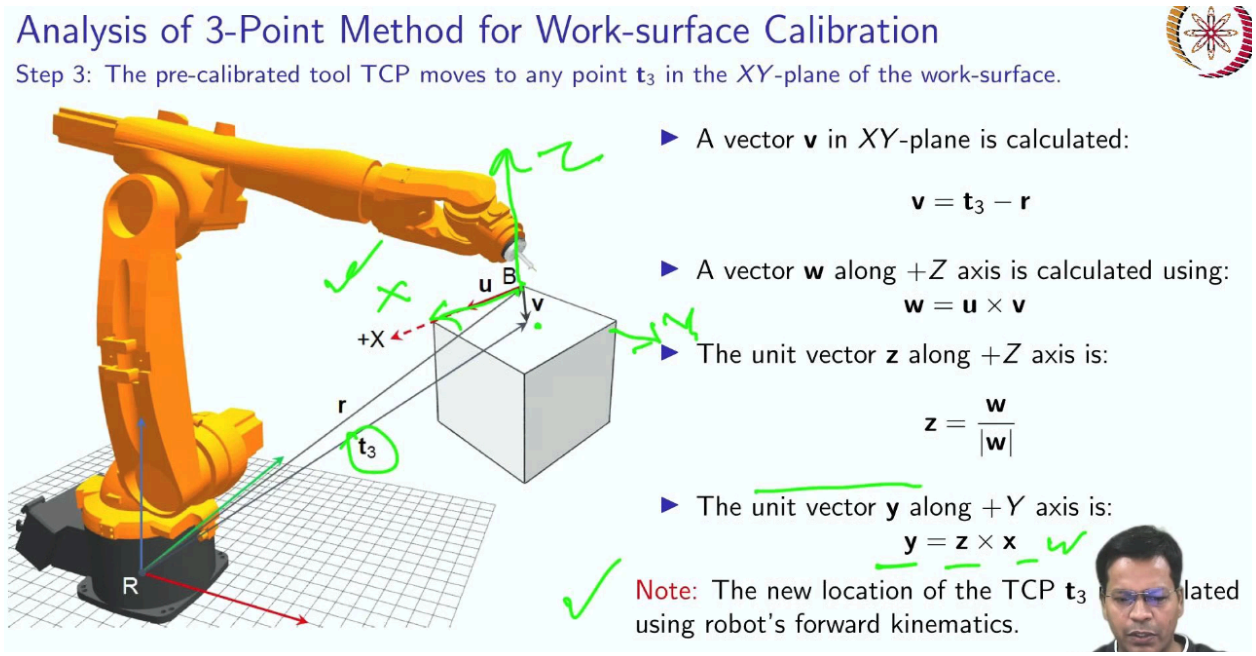
So, let us begin with that. So, how many unknowns are there? The position of the frame is unknown. So, that is three unknowns are there: x, y and z. Similarly, three orientation unknowns are there; that is, rotation can be there, So that is also to be defined. So, three plus three, that is, six unknowns are there. So, how many inputs does it require? Input variables do it require? So, the first thing you are doing you are taking it to a point, so you are ultimately taking it to the origin of the work surface, the work surface frame. So, it is three inputs that you are giving, that is, x, y and z of origin. Now, you are taking one point on the x-axis, so one variable is there, okay, along the constrained x-axis. So, it is just one variable you are taking it to that is there, got it? Second, when you are taking it to the XY plane, it is basically two remaining variables, which are defined here. So, the total variables are six here. So, it becomes a solvable system, and your robot controller can solve it.



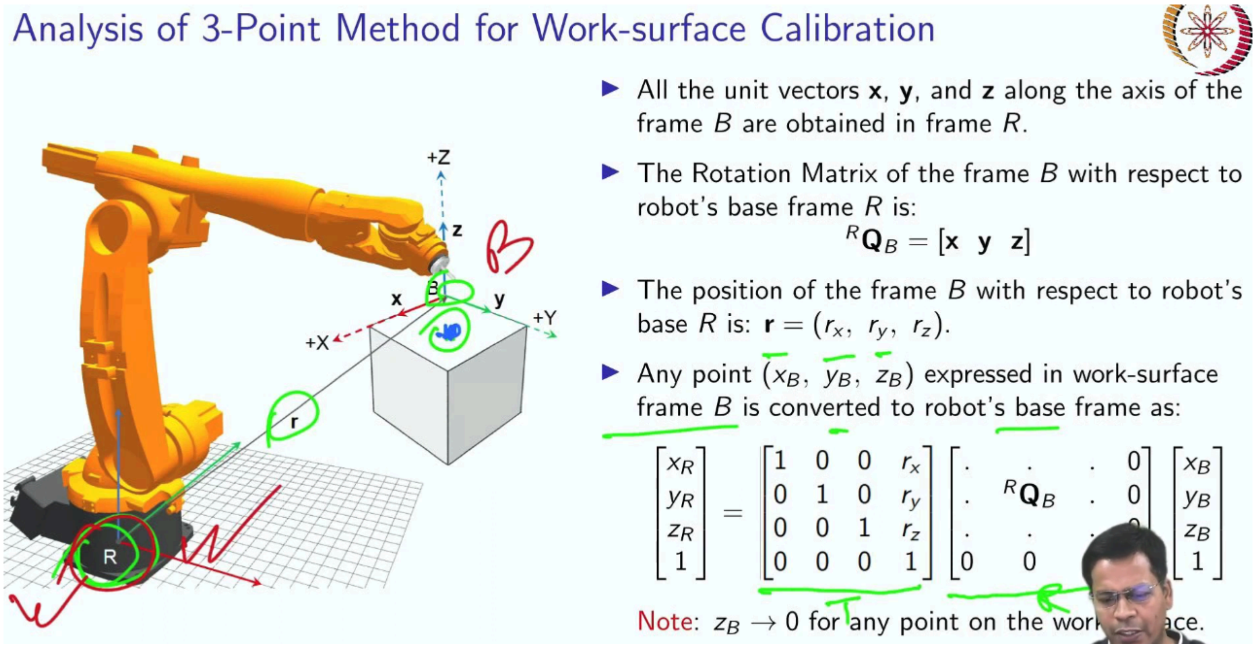
So, let us do it ourselves also. So, the TCP of the pre-calibrated tool moves to the origin of the work surface. So, this is the first thing that you did. So, what it actually did? It calculated this position vector. What is that position vector? It is a vector which connects the robot frame, and the frame that is attached to the work surface is the origin of that. So, that location is. So, it is rx, ry and rz, so that point is quickly assigned to this origin. How is that point calculated? It is calculated using the forward kinematics because once you take your pre-calibrated tool to this location, that means using the joint angles, the dimensions, and the do parameter of all these will let you calculate this quickly. So, that is how r is now known. So, after the first step, what you have in hand is r, r vector. So, that is basically the position vector. of the origin of the work surface frame.



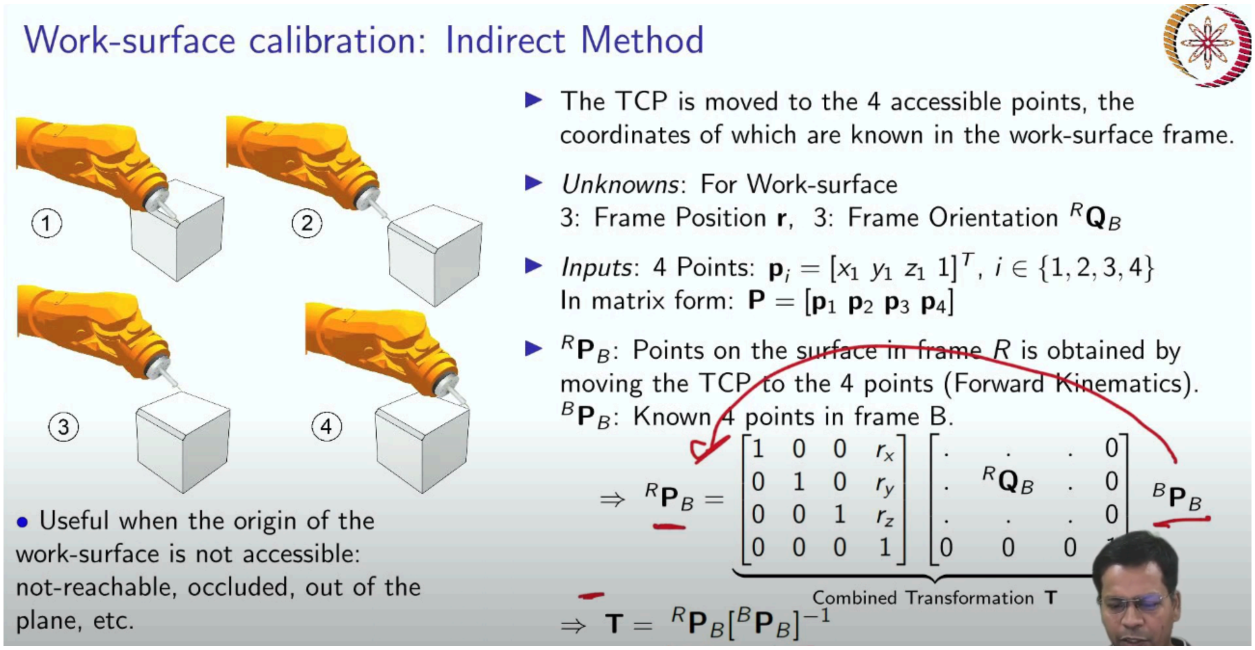
Now, the second step. So, the TCP of the pre-calibrated tool moves to a new point, that is, t2, which is on the x-axis of the positive x-axis of the work surface. It is somewhere over here. So, you took it here, Got it. Now, again, what you did using forward kinematics, you can calculate how much t2 is. So, t2 is known because you made your TCP touch to that location. So, using your forward kinematics, you can calculate t2. You already had r in hand. That is the origin B on the work surface was known, so r is known, t2 is known. So, you can calculate the u vector. What is u? u is a vector which is aligned along the positive x-axis, So that is u. It can be calculated using a simple vector triangle, which is here: u, r and t2. So, that is quickly done. So, a unit vector that is along the x-axis can be calculated as u by the magnitude of u. So, that gives you a unit vector along the positive x-axis. From the origin, this is your x-axis. So, anytime we are taking our TCP to any location, we, use forward kinematics to calculate that tip location. So, that is very easy.



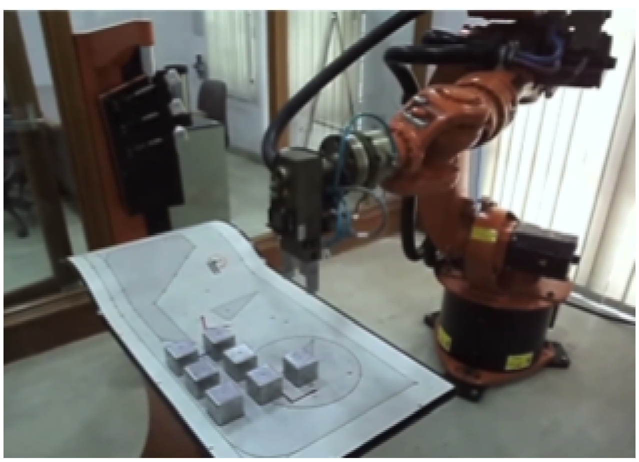
The third step, the pre-calibrated tool TCP now moves to a point t3, which is in the x- y plane of the work surface. So, if this is assuming this is positive y, it is going to be. So, you have to remain in this plane somewhere. So, the robot tooltip is now taken to this location. Now, what do you get? You can immediately obtain t3 using robot forward kinematics. r was already known, so now you can calculate a vector v. What is v? v is a vector in the x- y plane, which is given by t3 minus r, the way we calculated u. exactly in the same way we can calculate v. Okay, but v is not aligned with the y-axis or x-axis; it is just a vector which starts somewhere over here, and it ends anywhere on the x y plane. So, this is your v. So, altogether, u and v lie on a plane, which is an x-y plane. So, now, quickly, you can calculate a vector w, which is along the positive z-axis, by taking the cross product of u and v. If you take the cross product of u and v, which is lying. Both of them are lying in the x- y plane. So, you get a vector which is perpendicular to that plane. So, that is w. So, any vector which is perpendicular to the x- y plane will be aligned with the z-axis, positive z axis. So, if you take it, u cross v, it is a positive z-axis. So, you got what You got w. Now, I am ready to find out my unit vector along the z-axis. So, that is nothing but w by the magnitude of w. So, you got what You got, z. So, you have now obtained the z vector, a unit vector which is aligned along the positive z-axis. So, that is found out. So, now you know the unit vector along the x-axis unit vector along the z-axis. So, you can quickly find out the unit vector along the y-axis by taking the cross product of z and x. So, this was your unit vector z, and this is your unit vector x. So, taking a cross-product like this will give you a unit vector along the y-axis. So, that is your bold y small. So, again I am using TCP location t3 found out by forward kinematics. That is now quite trivial. So, you got all the axes, you got x, that is through the first step. You got y now, as soon as you contacted here, you found out z and finally y. So, all three unit vectors which are aligned along the x, y and z axes of your work surface frame. So, you now know that.

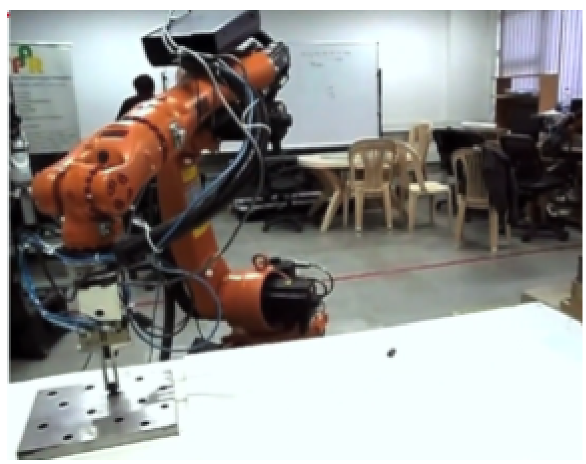


So, all x, y, and z are now known, and you also know r. this r you got to know at the first instant when you brushed it here. The TCP was brushed over here. So, r is known. x, y and z vectors are known. So, now the rotation matrix of frame B with respect to the robot base may be given as this, that is, projection of x vector along your robot frame, because you know these are the vectors which are expressed in frame R. So, x will have components along i, j and k. So, you will have the x component along i, the x component along j, x component along k. So, you will get this, this and this column, and similarly, y and projections of y. All the projections of y will go here, and the projections of z will go here. Altogether, it makes it a rotation matrix. What type of matrix is that? It is the rotation of frame B with respect to the robot's frame. That is, this frame got it. So, you quickly got the orientation matrix of frame B. You already had what r? So, this r was already known, given by r, x, r, y and r z, that is the projection of r along x, y and z. So, everything is now known. So, any point if it is there with respect to this frame, so that is x, b, y, b and z. b, which is expressed in the workspace frame B, can now be quickly converted to the robot frame by multiplying the orientation matrix. That is the first transformation that you do, that is the rotation transformation, and then you do the translation that is, first, you have to rotate, and then you move it to the new location. So, any point, if it is in this frame. So, you have to do these two transformations and calculate that point with respect to the robot's frame. So, if it is in this frame, you can calculate the same point in this frame. So, if it is in a robot's frame, now you can do inverse kinematics quickly, and you can make your robot go to that particular location. So, any point you get in hand, which is expressed in frame B, can now be converted to the point in the frame robot's frame, and you can do inverse kinematics, and you can take your robot there. Not just inverse kinematics can be any other control approaches. You can take your robot there. So, that is how it works. So, this is the first way of calibrating your work surface.

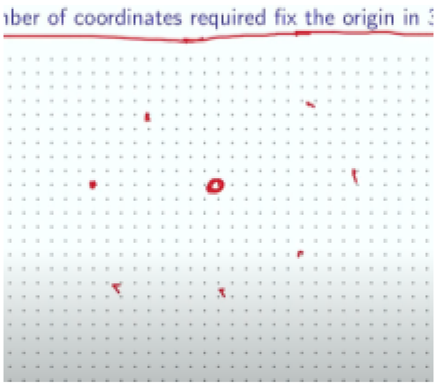


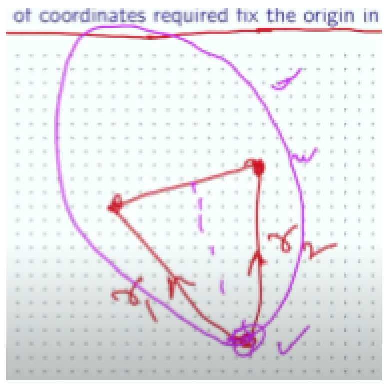
Now, we will look at another method, which is an indirect method. So, in this case, why this is useful? It is useful when the origin of the work surface is not accessible. So, sometimes origin may not be reachable by your robot itself. It is quite far off. So, that is not reachable and sometimes it may be occluded by some fixture in the workspace, or it is out of the plane. Your edges are not straight and towards that end, so forget it. So, you don't need the origin now to be brushed with your TCP. This method is very, very simple; what it says is that your TCP can now be moved to 4 accessible points. Any 4 points in the workspace which is accessible to your robot. The coordinates of which are known with respect to the work surface frame. This is the prerequisite, that is the condition which is here. So, all the points that your robot is now going to brush, those coordinates should be known with respect to your work surface coordinate system. So, you have to brush it to the 4 locations. So, that is, there are unknowns for the work surfaces, 3 for the frame position and 3 for the frame orientation. The same way we had it earlier. So, how many inputs you are feeding is 4 points, that is, Pi, x1, y1, z1, x2, y2, z2, x3, y3, z3, x4, y4, z4. i vary from 1 to 4, and I can arrange all the points in this manner. So, it's a capital P arranged in a matrix way, like x1, y1, z1, 1, x2, y2, z2, 1, and similarly x3, y3, z3, 1, x4, y4, z4, 1. So, this becomes your P matrix. So, what is this matrix? It is nothing but all the points collection. So, you can handle all the points simultaneously through any transformation matrix. So, this is what I am just creating over here. So, now this is what this is: position vector positions of points. All the points on the surface in the frame are obtained by moving the TCP to all 4 locations. So, now, what did you do? You simply took your TCP to those known locations. In 4 steps, you have taken your robot to all those locations because the robot already had a pre-calibrated tool. So, you quickly obtain those locations in the robot's frame. So, now, when you brushed your TCP to all 4 locations. So, points on the surface in frame R are obtained. That is the robot's frame is obtained using forward kinematics. So, two things are now known. That is, the same points in the robot frame and the same point in your base frame. That is your work surface frame. So, both the things you have in hand. What you need is a transformation between them. That is the unknown. Now, this is already known. So, now, this was known already. That is the point in its frame. That is the work surface frame. This is what. This is the point in the robot frame so that includes two transformations. You have to transfer this point from the base frame, that is, the work surface frame, to the robot frame; how is that done? This is your unknown transformation matrix that can do this that you want to identify here. So, it consists of a rotation matrix first and then this is the translation matrix. So, these two were unknown earlier in your previous method also. That is what we want to find out. So, this becomes unknown, so you can quickly take the product and find out t together, in terms of variables, and you can find out that t using those variables can be quickly found out by finding out by taking the inverse. You have to bring this here, taking inverse, and this was already there, this was already there. You brought it here, taking inverse, and you can find out the unknown transformation that can take you from your base, your work surface frame, and any points on the work surface frame. It can bring it to your robot's frame. So, that is what you wanted to know. This is the transformation matrix. A combined transformation matrix can now be obtained. So, that is how any point now, if you get in hand, you can quickly use this transformation matrix and convert it to the robot frame, and now the robot can be programmed to go there very easily.





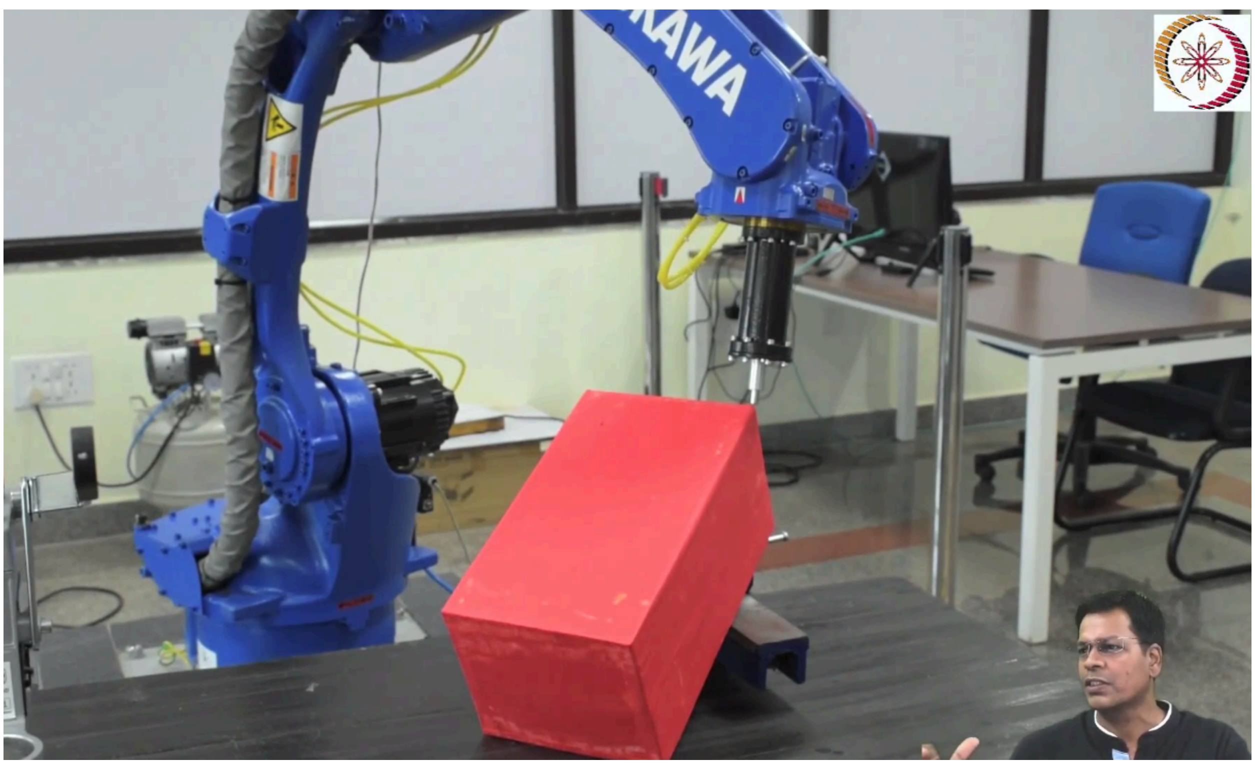
So, that is how, and before I move ahead, I will show you some videos which are applications of this. So, that was you from the intro part only, so I will just quickly show. So, this is an application when a robot is trying to create an array of cubes, but the cubes are not to be put in a coordinate system which is there on the table. You see, the cubes are aligned with respect to the frame which is there on the table on which it is skipping. So, that is an array of cubes. It is quickly taking all the pellets from the location which is a taut location. So, those are the locations which are there with respect to the robot's frame, and it is putting all the pellets into the frame, which is there on the work surface. So, the final cube will now go on top of that and be done. This is one tentative application. There are many such palletising tasks which are there in the industry. So, this robot is using a vision technique to find out the pellet location on the table. It picks it up, corrects it, then it is inserted in a hole which is now again in a frame which is attached to this work surface. It is inserting it, it will pick up another one and again, put it. So, this is yet another application. When the workspace is vertically placed, again it is using vision to find out those pellets and correcting it will correct it. Pellets are lying horizontally, so it is picking up like this and then correcting it properly so that it can be inserted into the hole. So, this is another very good example when you require such work surface calibration.



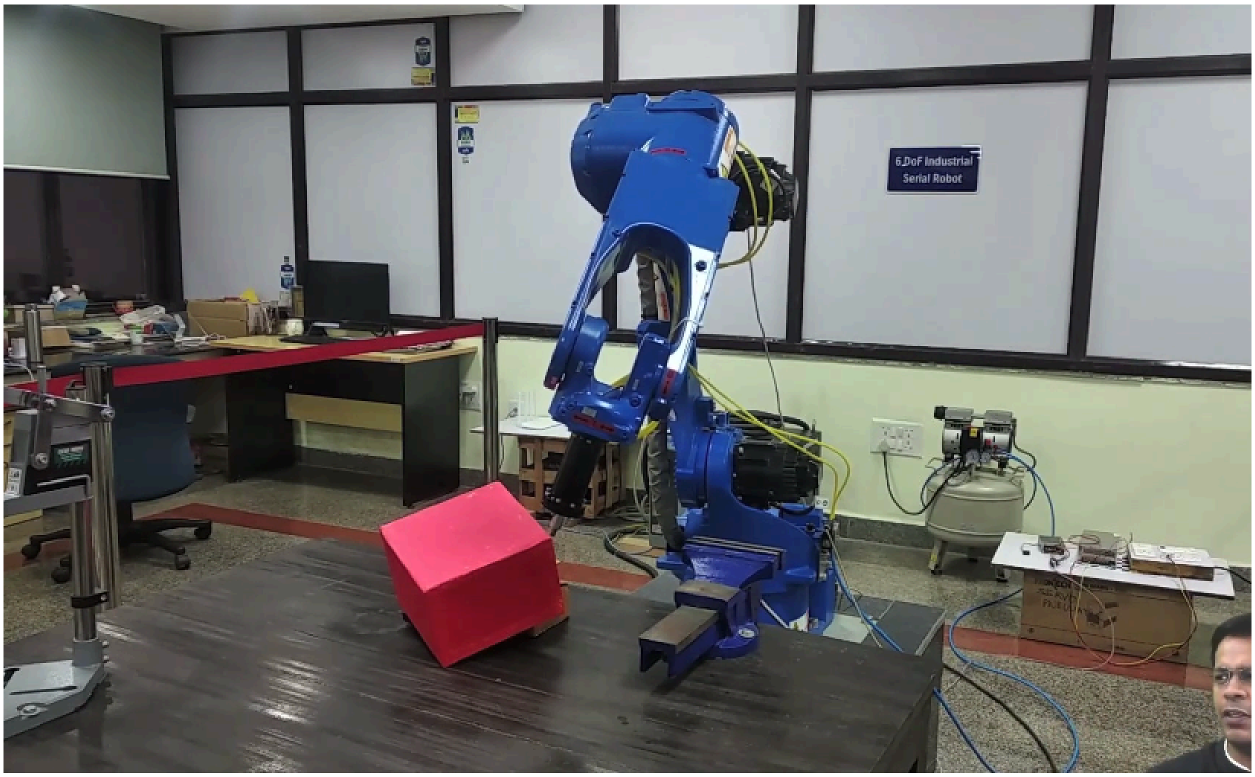


So, yes, this is there, and let us just quickly discuss how many minimum numbers of coordinates which are required to fix the origin in 3D space. This is a small discussion that I want to have because we will be dealing with similar techniques later on as well in this module. So, let us first clarify that. These are some fundamentals that you need to know. So, let us say, if I have just one point, so where your origin can lie, your origin can lie anywhere in the workspace so that you can have an infinite location for your origin. So, one point is not enough to constrain your origin. So, you can have anywhere in the 3D workspace, okay, so it is not enough. You see, now I will do another thing. I'll just add one more point. Now, I have a line segment. So, if these two points are defined in space, where your origin can go any idea, so, yes, your origin can lie if at all this point is at a distance r1, and it is a distance r2 from the origin. So, the locus of your origin will be somewhere on the circle, which is like this: okay, so there is an equivalent radius which will go there and that about which your whole origin can be located anywhere on this. So, you can have an infinite number of vectors which can be there. That bears the origin, and it can tell you the same coordinates. So, your origin is still not located properly. So, two points are not good enough. So, now again, I'll try to put yet another point in the workspace. So, again, I have put yet another point in the workspace. This time, you have three points. Is it enough to state your origin and where your origin can go? So yes, let me draw that triangle, which is formed using this. So, a set of three points is very good enough to define your surface. But your origin, is it constrained, is still so. It can go on any side of your plane, which is defined using these three points. So, your origin can be here, on the upper part of the surface, or it can be somewhere behind the surface. Okay, so this can be r1, r2 and r3. From the other side also, you can draw similar three vectors, which are r1, you have, r2 and r3. So, still, your origin can have at least two positions. So, even three points are not good enough to define your origin, if it is freely defined in the work surface or on the workspace of the robot, your origin is still not constrained. So, finally, as soon as you put your fourth point, as soon as you put your fourth point, one, two, three and four, your origin is fully constrained, mind it. It has a singular solution. Your, this one of the points should be off the plane, of the plane, okay? Otherwise you will get a singular matrix for this, and it becomes not solvable. So, if these three are in the plane, it will define your surface. So, the distance of this origin, the way you constructed the origin, can go here, origin can also go there. But if it is off the plane, you cannot mirror your origin about that plane, so immediately you will see a difference, difference in distance if it is taken from both the new origin location. So, only one solution is now possible. So, that finally fixes your origin location. So, that is why a four-point minimum is required for the method that we followed at the end; that is when the origin is not accessible. Minimum four points to define your origin, and using the orientation that is normal to the surface, you can find out the direction of your frame also. So, yes, the location of one of them can become the origin and the whole of the origin system can be defined. So, this is very, very fundamental. You should know this.

So yes, that's all for today. In the next class, we will be dealing with fixed tool calibration. If it is a tool, it is not an object that the robot is handling. Your object is now held by the robot, and your tool is mounted somewhere in the workspace, and you are trying to machine your object using that tool, which is now fixed. So, how to calibrate that? So, this type of application I will show.



So, let us quickly see some calibration demo demonstrations. What is the result of today's calibration technique? So this is one of the ways where this base, which is a red cuboid that you can see on the top surface, is now calibrated. My robot can now move along the edges of the cube which is there in the workspace of the robot. It is moving along one of the axes, that is the edges. Along the edge of these cuboids, it goes to the origin now. This was so wide that it was covered immediately. It was wide. That it did now, and then it is along X. It is moving. Now, it is along Z. it is X, Y and Z. So, you see, your robot can now move in a frame which is attached to the work surface. The top surface of this cuboid that you can see is your work surface. So, that is the advantage of having that.



Now, let me show you yet another video. In this work surface, your robot can now orient about, the axis is also-it is a combined orientation that I am trying to make, so about any point. Nor can your robot make tilt and roll and of thing. So, once the work surface is calibrated, you can position, or you can orient with respect to the frame which is attached to the work surface. So, that is what was the message of today's lecture.

So, yes, that's all for today. In the next class, we will be dealing with fixed tool calibration. What is that? Your tool is mounted in the workspace. It may be a simple grinding tool that is rotating, and you are holding the object using your robot's gripper, and you can take it there, you can machine it. So, now, the tool is not mounted with the robot, and the object is stationary somewhere in the workspace. Otherwise it is another way which is there. So, you have a tool which is there, and your object is held by the robot. There are a few applications that you have seen in the application video in our very first lectures, that is, that was the introduction to the robots. So, you saw it was an adhesive operation, gluing operation, when an adhesive dispenser could be stationary, and your object was held like this, and you can do that kind of motion. Hope you remember it. So, yes, that's all. Thanks for today. That's all.