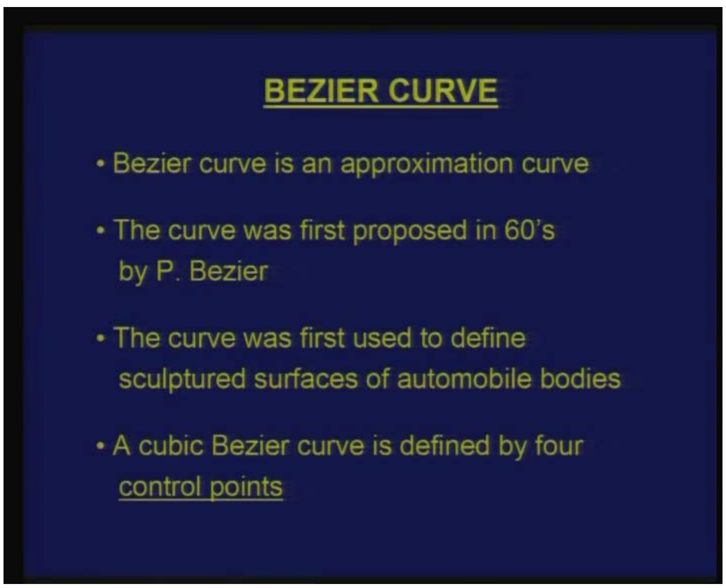
Computer Aided Design
Prof. Dr. P. V. Madhusudhan Rao
Department of Mechanical Engineering
Indian Institute of Technology, Delhi
Lecture No. # 6
Parametric Bezier Curve

So we will start today's lecture. Last time we started the subject of studying free form curves and what we did is to look at one of the curves that is namely Hermite curves which is basically an interpolation curve. So from today onwards we will look at a few representations for free form curves which are mainly an approximation curves. One of the curves which we are going to study today is Bezier curve which is in fact one of the earliest free form curves which has been proposed, when the geometric modelling subject was developed. So the title of today's lecture is Bezier curves.

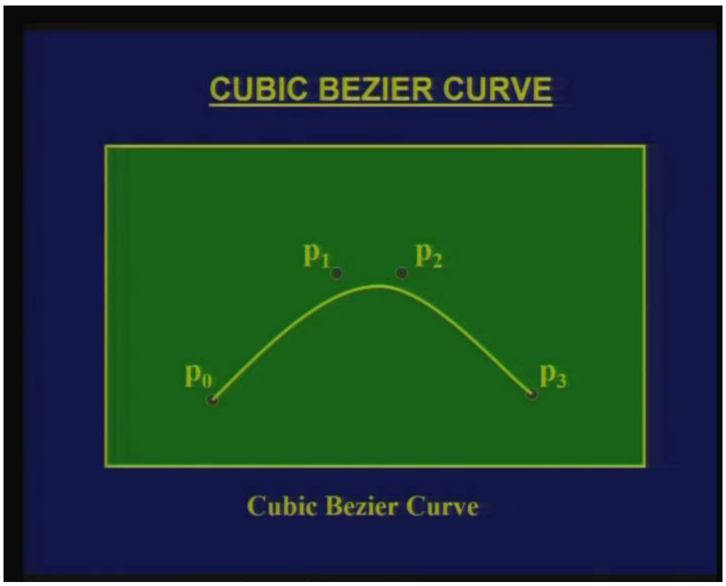
(Refer Slide Time: 00:02:23 min)



Now as I said Bezier curve is an approximation curve. Now when I say approximation curve whatever inputs which you give in order to define the curve, all the inputs are not satisfied. These inputs will be used to define the curve like for example if I take an example of a Hermite curve which we have already studied. When we say there is a four point form if I specify four points, the curve necessarily passes through all the four points but that may not be true when I give four control points here where these points are used to define the curve but the curve doesn't pass through all the points. So that's why this is classified as an approximation curve. Now this was first proposed in 60's by French by name P Bezier. This was a part of a CAD system called as a UNISURF.

So first time the Bezier curves find, they were proposed as a part of this particular package called UNISURF. And the purpose of UNISURF was to basically develop some curve which can used to define free form or sculptured surfaces of automobile bodies. So in order to, like in the process of coming up with a new curve representation for sculptured surfaces, so this particular curve has been proposed first time by the Bezier. And a cubic Bezier curve is defined by four control points, just like a cubic parametric cubic curve or a cubic Hermite curve is defined by four points in a four point form. Similarly a cubic Bezier curve is defined by a four control points and a Bezier curve need not be cubic, cubic is one of the representation which is popular but one can also have higher order curves as far as Bezier is concerned. And the points are called here as a control points because these control points will define the shape or in a way they dictate the shape, what will be shape of the curve. If I change control points, I am also changing the shape of a curve in a Bezier curve.

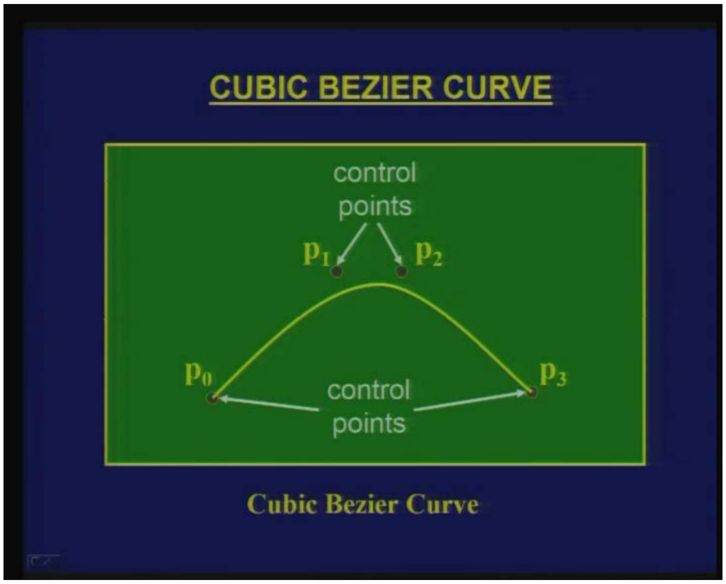
(Refer Slide Time: 00:04:51 min)



Now here is a pictorial representation of the curve. So I have four points that is P_0 then there is P_1, I have P_2 and P_3. You see that they are numbered from 0 to 1. So cubic means you will have, you will go up to P_3. If it is one higher order that is quartic forth then I will go from P_0 to P_4. So it's this kind of representation is easier. So it starts with a point which is P_0 and P_1 P_2 P_3 are the 4 points which are used to defined. Now the curve can be either a planar curve, it can lie in a plane. If all the four points which are defined in a plane so then it's a plane planar curve or if I give four points in a space, I can also define a Bezier curve which is a spatial curve or it's usually called as a space curve. So these P_0 P_1 P_2 P_3 can be anything like whatever it can be lie in a plane or it may be out of plane, whatever it may be.

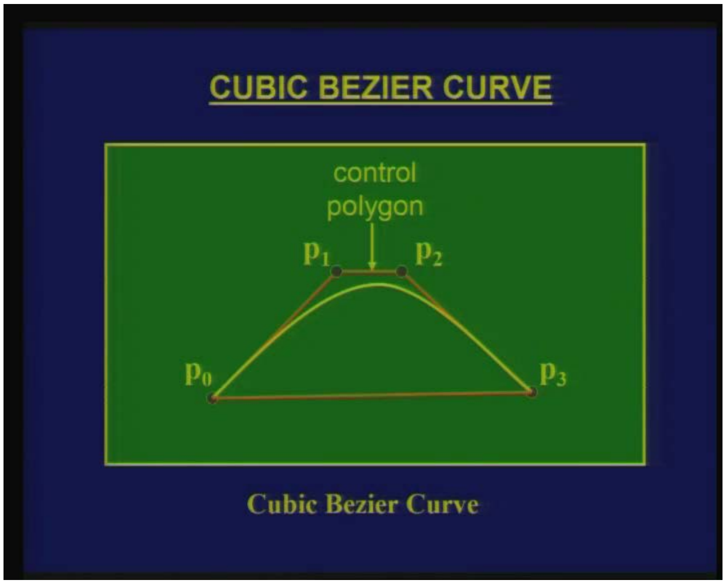
Now one of the things which you see from this particular this thing is the curve passes through the first and last points which are defined, like P_0 is a starting point, it passes through that. P_3 is the last point so it also passes through that but it uses the coordinates of the other points in order to define the curve but it doesn't pass through in this particular case. So that's why we call it as an approximation curve which is shown here.

(Refer Slide Time: 00:06:34 min)



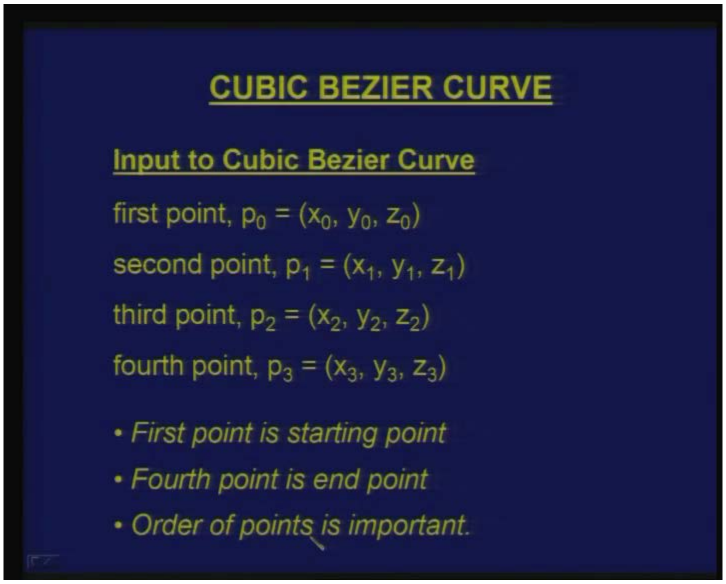
As I said these four points which are used to define the curve are called as a control points. And depending on the like position and position of the control points, the curve will be, the curve will be different. For example if I take let say one of the control points and move it to a new place as a part of the curve design, the curve shape automatically changes. So somebody who is trying to define, let's say curve not necessarily mathematically let's say interactively, interacting with a computer also can use this kind of a representation. First I will display all the control points and the corresponding Bezier curve then keep manipulating these points by moving from one place to another to see how the curve changes its shape and try to arrive at a shape which is usually pleasing for certain applications like particularly in a graphics applications. But it also has a mathematical definition which will be used to manipulate the curve or to define a curve which has a desirable starting and end points and desirable properties.

(Refer Slide Time: 00:07:49 min)



The four points which are used here in order to define a cubic Bezier curve, when I join them by straight lines, the polygon which is formed by joining these vertices is usually called as a control polygon. Basically control polygon is nothing but a polygon formed by joining the control points. Another name which is give for control polygon is characteristic polygon. So that is in some of the text, some people use the word characteristic polygon more often for defining a polygon which is formed by joining the control points. So what you see here is the characteristic polygon which is shown in a different color in this particular aspect.

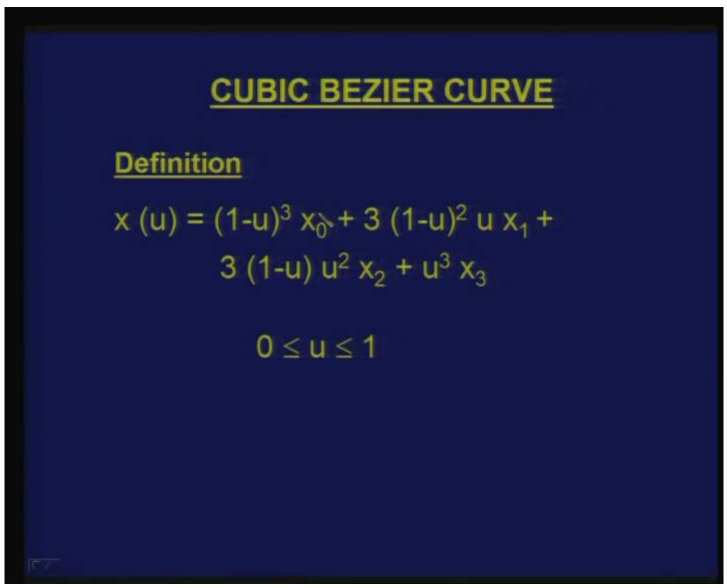
(Refer Slide Time: 00:08:39 min)



So the input if we see to a Bezier curve here is if I am taking an example of a cubic Bezier curve, so I have a first point which is defined by 3 coordinates x_0 y_0 z_0 then I have second and third points. Second point here is x_1 y_1 z_1, third point is x_2 y_2 z_2 and then there is the forth point which is x_3 y_3 z_3. So the first point is also the starting point, the curve passes through that and forth point is an end point curve also passes through that and these are the other control points through which the curve does not pass through.

Now one important thing is order of these points is important. If I change the order of these points, it is not that given four points and you have a unique curve whatever may be the order that is true with even when we do, suppose if you are given a set of points and you are asked to fit a curve passing through that, the order is always important. So similarly here also, what order which I specify is important because first and last are the interpolation points that means the curves passes through these points which is one of the properties of Bezier curve.

(Refer Slide Time: 00:10:08 min)

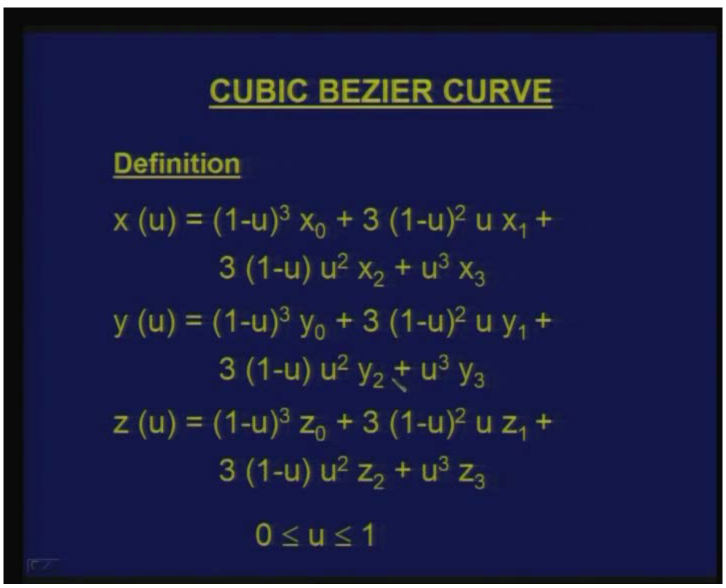


Now mathematically a cubic Bezier curve, this is a definition for a cubic Bezier curve which is shown here. And a curve can be defined as we have seen earlier; any curve can be defined as a function of a single parameter, the parameter which is shown here is a u. And what you see here is only equation which is given for the x coordinate like x which is a function of u is defined by a mathematical expression which is shown here. And what are the inputs which are used to define the x? x_0 that is coordinates, x_1 x_2 and x_3 that is only the x coordinates of the four control points which are given. And if you see the equation, it has two terms like one is 1 minus u cube, like one is basically a power of 1 minus u, another is a power of u. So you have a various combinations of this. For example you start with 1 minus you cube and it is u to the power of 0 that is nothing but 1. Then 1 minus u square, you reduce the order by 1 and it becomes u to the power of 1 and then you have 1 minus u and u square and then it is u cube.

So, slowly you are decreasing the power of 1 minus u and increasing the power of the term u, so it's very easy to remember. That is no difficulty in remembering if I look at this. Another thing which you also see that there is also a coefficient for all these things. You start with 1 then you have a 3 then you have set, third expression also has 3 and forth again reduces to 1, so it is like 1 3 3 1 are the coefficient which are used to do that. And you also have a range for this particular curve because you are referring to a curve segment, so it is the parameter varies between 0 and 1.

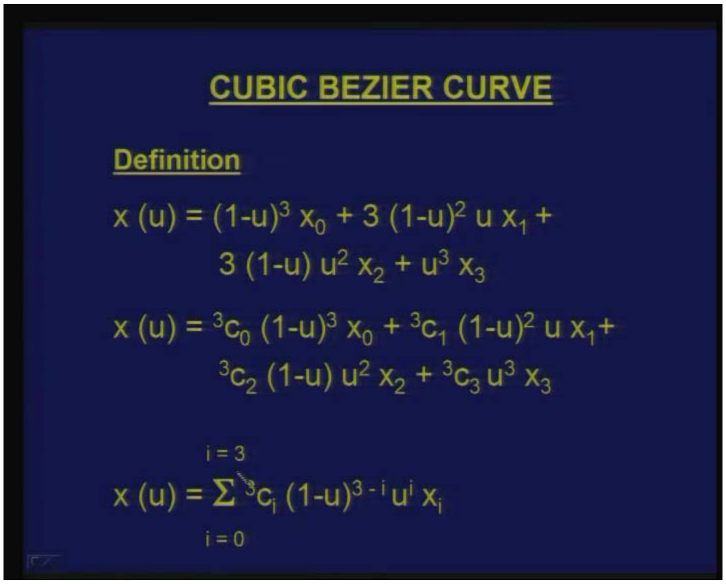
Now, suppose if I substitute u is equal to 0 in this expression, this becomes 0, second expression third and forth terms also become 0. So I am only left with x_0 that means the curve passes through that. Similarly if I put u is equal to 1, the first second and third terms becomes 0 because they all carry 1 minus u and I put u is equal to 1, so it reduces to x_3, so the curve passes through that. So this kind of a representation is easy to remember and this is a definition for a cubic Bezier curve, a mathematical definition which is given.

(Refer Slide Time: 00:12:59 min)



I can extend the same definition to y and z. All that which we have done is that you have the same expression; these x_0 x_1 x_2 and x_3 have been replaced with y_0 y_1 y_2 and y_3. As we have seen earlier that x y and z can be independently defined with like no relation among the coefficient of x y and z which is true with Bezier curve also. That is this, that was true with parametric cubic curve also, Hermite curve which we discussed this thing. So this is the complete definition of a Bezier curve where x y z are defined.

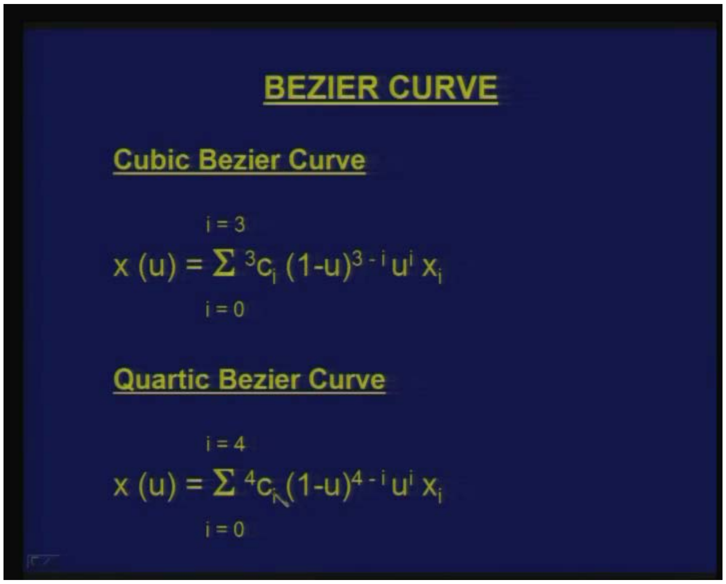
(Refer Slide Time: 00:13:43 min)



This is the expression which is x expression which I have produced. Again, now I can always rewrite this particular equation slightly differently. What is done is you have the same expression, the coefficients which were 1 3 3 1 basically correspond to the binomial coefficients like one is nothing but $3c_0$, 3 is $3c_1$ $3c_2$ is same as $3c_1$ because this is $3c_3$ minus 1 and then $3c_3$ is again 1. So the binomial quotients are like 1 3 3 1 which have been written in this. In fact the way of Bezier curve is defined is that these coefficients are always the binomial quotients, those have been used to define the Bezier curve or I can simplify the expression further by saying that.

So you have a four terms which are these thing like you have 4 terms goes from i is equal to 0 to 3 and if I substitute i is equal to 0 then this is 3 c_0. Then this is i is 0, so this is 1 minus u cube and u to the power of 0 is 1 and this is x_0. So that is nothing but exactly the first term which I am trying to use. Similarly by substituting let's say i is equal to 1, I can get back this term and substituting i is equal to 2, I get the third term and by substituting i is equal to 3 this is $3c_3$ and this becomes 0 you have u cube into x_3. So you can compress the entire definition of let's say a Bezier curve, a cubic Bezier curve which has been taken here as a very simple expression which is shown here. So we can say that this is more generic definition. This is basically an expanded form of the definition. And what is shown here is only for the x, you have similar terms for y and z which are not shown here.

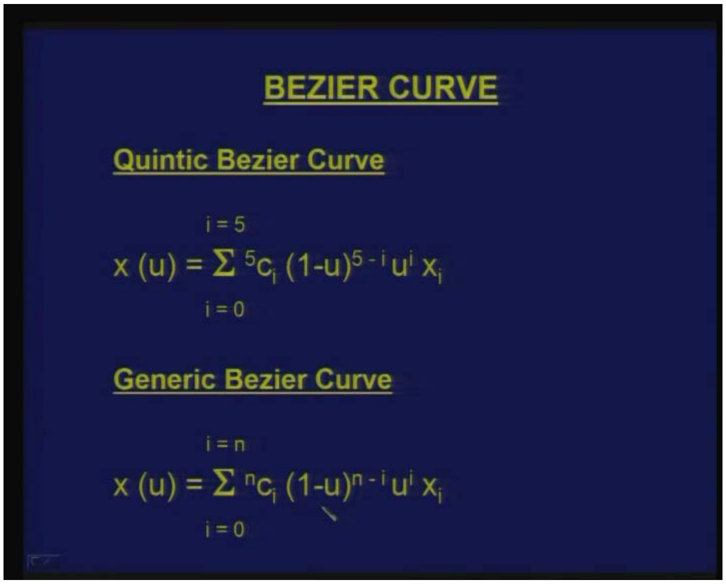
(Refer Slide Time: 00:15:52 min)



So this was the same curve which we saw in the last slide which has been reproduced here. Now I can extend the definition to higher order, so I can have a quartic Bezier curve where I go to terms like 1 minus u to the power of 4 and u to the power of 4 and it is defined by 5 control points. So you go from 0 to 4 so corresponding to 0, 0 to 3 we had 4 terms here so corresponding to 0 to 4, I have 5 terms here. And now the binomial coefficients are different, they start from $4c_0$, $4c_0$ is 1, $4c_1$ is 4 then $4c_2$ is 6 then $4c_3$ is again 4 and $4c_4$ is 1. So I have binomial coefficients in this case is 1 4 6 4 1 which are this thing.

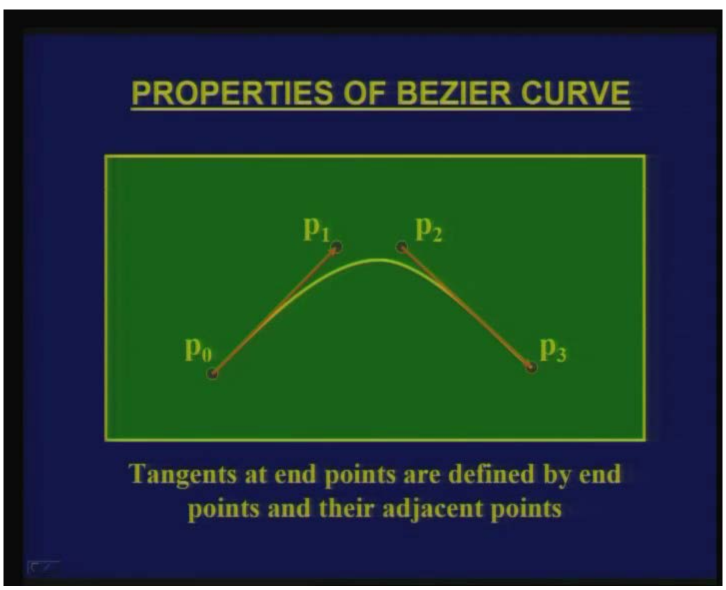
And similarly you have the, slowly you reduce the power of 1 minus u to the u and then increase the power from 0 to 4 and you have 5x coordinates which are given which correspond to 5 control points. Now in this case, the curve again passes through the first and last. It need not pass through any of the 3 control points which are given, the same definition which is extended. So a representation like this becomes a very general or a generic representation.

(Refer Slide Time: 00:17:28 min)



I can go to one order higher, this is called as a Quintic Bezier curve where you go to a terms like u to the power of 5 etc, binomial coefficients are different. I have now total how many terms? 6 terms which are used to define a Quantic Bezier curve? And the definition can be extended to a curve where you have n plus 1 control points. So if it is a forth like if I am having a powers which are maximum power is 4 then you have 5 points. So if I have like maximum power which goes is u to the power of n or 1 minus u to the power of n then it goes from 0 to n it can be defined by n plus 1 control points. We will define an nth order Bezier curve.

(Refer Slide Time: 00:18:43 min)



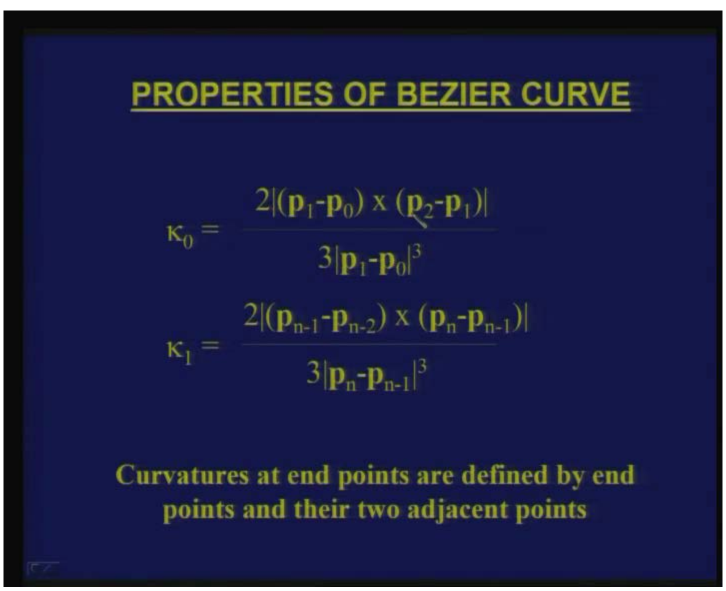
So the definition is very simple, I think given this definition you can have like you can write it for any order whatever may be the n, whether it is 3 4 or 5. So remembering this is much easier. So it's a simple expression to define also.

Now let's look at some of the properties of Bezier curve like what is really a specialty of this curve. Why this curve is given so much importance is one of the like subjects of study now. This is what I have shown here is a cubic Bezier curve where it is defined by 4 control point that is P_0 P_1 P_2 and P_3. One should remember here is that each P_0 P_1 which are shown here is a vector consisting of 3 components x_0 y_0 z_0 or x_1 y_1 z_1. Now the tangents at the end points are defined by end points and their adjacent points. So these are the end points if I look at what are the tangents at these end points, they are defined by the last two points. For example even if I take let's say Bezier curve which has 6 control points, and then I am defining a Quintic Bezier curve.

If I take let's say what the tangent is, tangent always follow the direction where you are going from point if I join them by a straight line, so I have a vector which is going from P_0 to P_1 which will always represent the direction of the curve. So this wills, this straight line will always be a tangent to the curve is one of them. So it interpolates the last points then the slopes which are first derivative are defined by end points and one point which is next to that. So same is true with here like the slope in the case of end point is defined by p_2 and p_3 which are in this case.

Now this property is not restricted to like what is called as end points or the slopes, it can be extended to higher also. That means if I want to know the second derivative of the curve at the end points, they are completely defined by the last three end points like if I take p_0 p_1 p_2 will completely define what is the second derivative or more commonly what you call as a curvature, so curvature is defined by that.

(Refer Slide Time: 00:20:55 min)

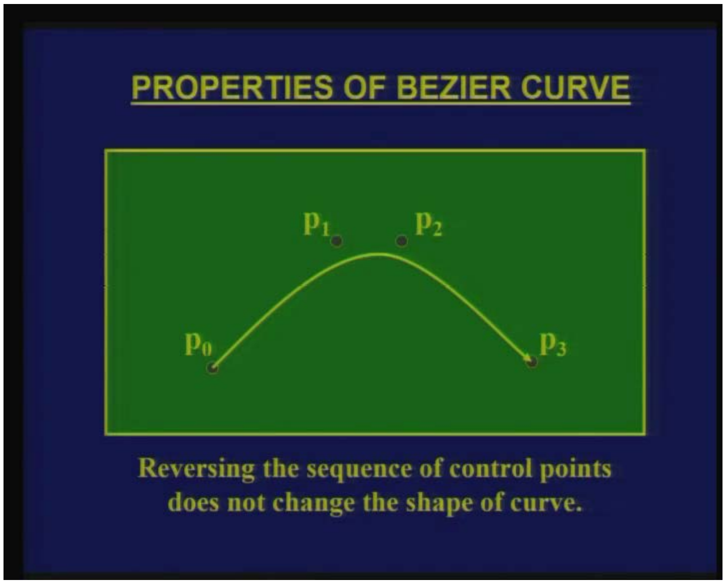


So if I look at expressions for curvature, these are the kappa which is shown here $kappa_0$ and $kappa_1$ are the curvature of the curve at the end points. So what do you mean by a curvature? Curvature is nothing but 1 by the radius of curvature. So the local radius of curvature when I take a reciprocal, I get a curvature. So the curvature at 0 and 1 that means this is not at point 1, this is the end point that is when u is equal to 1, this is when u is equal to 0. I have an expression which is given for the Bezier curve which has been derived. If you really look at this, what are the points on which K_0 depends is it depends on p_0, it depends on p_1 and it depends on p_2.

Now this expression is not necessarily for cubic curve. This expression for curvature is a very generic whether I am using a quartic or a quantic or let's say I am using 10 points to define a Bezier curve, still this is the expression which is used for curvature. And similarly the curvature at the other end point is defined by completely defined by three points. What are those points? It's p_n which is the last point then you have p_{n-1} one prior to that and p_{n-2} that is the second one which is prior to that. This means suppose if I take let's say n is equal to 10 that means I have 11 points which goes from 0 1 2 up to let's say 10.

Whatever may be the position of let's say the points, for example p_0 p_1 and p_2 will affect whatever may be the let's say the coordinates of p_3 p_4 p_5 etc intermediate that doesn't affect the curvature value of the curve. So it is only defined by the three points. So we can say that a second derivative will be defined by the three adjacent points, slope will be defined by two adjacent point. If I go one level lower that is curve passes through the last points and same thing can be extended to higher like if I take the third derivative, this is the second derivative, if I take the third derivative that will be defined by the four points which are starting from end points and the neighboring points. So this property is one of the unique properties of Bezier curve which we have seen. So we have seen basically for the slopes and the curvature but the same thing can be extended to higher orders when it comes to the definition of the curve.

(Refer Slide Time: 00:23:53 min)

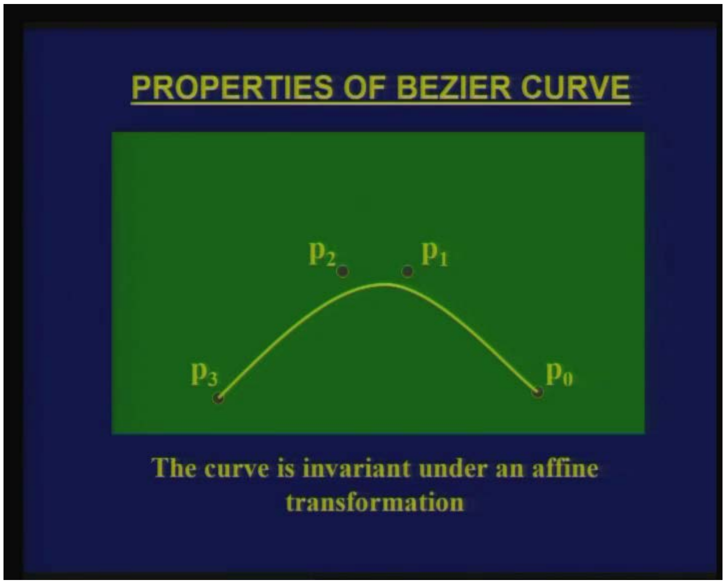


Here is another very interesting property of Bezier curve is that as we said sequence of points is important in order to define the Bezier curve but if I reverse the sequence, I get back the same curve like suppose instead of giving p_0 p_1 p_2 p_3 as input, I give let's say input as p_3 p_2 p_1 and p_0 that means you are keeping the order same but only reversing it. And if I tried to plot the curve or try to come up with let's say mathematical definition for the curve, you have the same curve that means curve doesn't change its shape or size by reversing the order of points. But if I change it other way around then I may have a difference that means one can say that the curve is symmetric about the two functions that is 1 minus u and u.

Only thing which you are trying to do is reversing, you are starting with let's say u cube term and going to 1 minus u cube whereas in the other case it is different. So if I just reverse the points, this is the case where I am starting from p_0 p_1 p_2 p_3 it is the same curve but only thing we change is the direction of parameterization like whenever you define a curve, you also specify a parameter range that is it goes from 0 to 1, u is a range. So now u is equal to 0 corresponds to a different point and when I say u is equal to point three, it will be different for the two different curves but the curves are the same only the direction of parameterization has changed in this particular case. So reversibility of the curve is important.

Now why is this important is suppose you are given a set of points and you have to define a curve like instead of let's say reading from one direction, if I have to read from the other direction or the way I collected the data points may have reversed. So it should not change the shape of a curve, I should still get the unique curve in the case of this. So this is another very interesting property of the Bezier curve which is shown here.

(Refer Slide Time: 00:26:08 min)

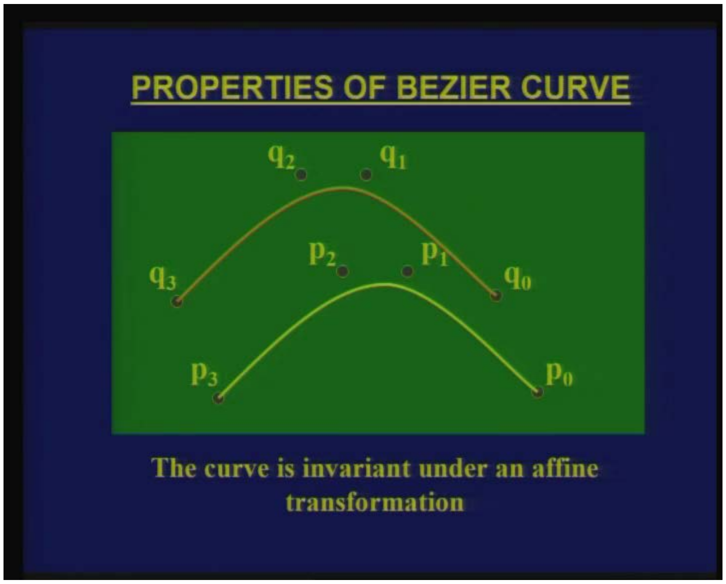


The other important property is that the curve is invariant under affine transformation. Now I think this needs some explanation like usually what happens in a geometric modeling or that is CAD in computer graphics work is that suppose if I have to define a curve let's say which is located somewhere in the space, you first try to define the curve at some convenient location. Let's say either in a xy plane or let's say or curve starting at origin and then you place this particular curve wherever it is intended to be. So in order to do that you do a transformation, these are called as a geometric transformation. So the common transformations which are used, geometric transformation is one is translation. That is you move a curve from one like one place in a space to another place in a space. So a transformation can be defined by defining a vector which is a translation vector which is a translation in x direction, y direction and z direction.

Similarly, I can have a rotation transformation. Rotation transformation could be either rotation about x axis, y axis, z axis or any axis, any arbitrary axis which is defined in a space and or I can have a transformation like scaling. When you are scaling an object in x y and z direction either the scaling factors in x y and z directions can be same or they can be different. Then there is also operations like transformation operations like shearing which is also one of the very common operations. So these are four common transformations which are used that is translation, rotation, the scaling and shearing. The only difference is the translation and rotation transformations are called as rigid transformations. It will not change the shape of the curve, it will only change the location that is where it is located and position whereas scaling and sharing transformations changes the shape of the curve like when I scale an object for example some of its properties changes. What is the length of the curve; it will be a different when I scale an object with certain scaling factor.

Now all these four transformations which are applied to Bezier curve are invariant. What do we mean by invariant is like if apply the transformation to the control points, I'll get back the same curve which is nothing but as if you are applying the transformation to the curve itself. This can be demonstrated by a simple example.

(Refer Slide Time: 00:29:05 min)

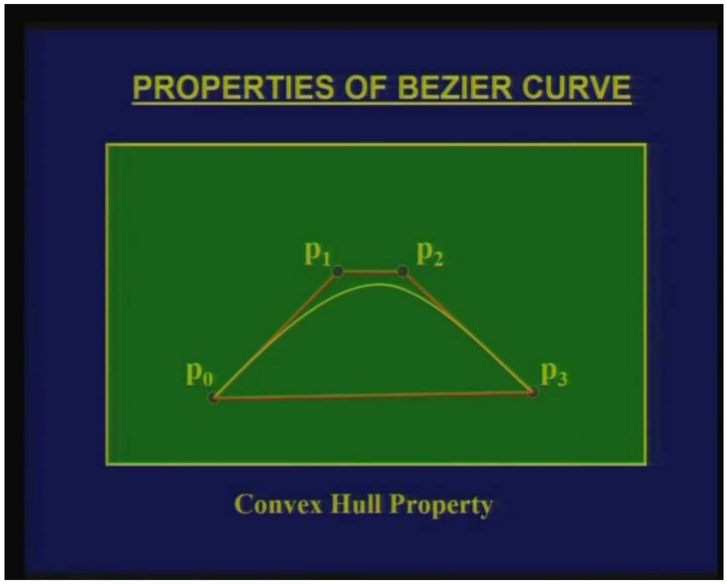


Suppose I have let's say a curve here which is defined by four a cubic Bezier curve which is defined by four control points. Now I move all the control points let's say along certain directions. So you have a vector where p three has been moved to a new position which is called as q_3, p_2 is moved by q_2, so p_1 is moved to q_1 and p_0 is moved to q_0. So if I draw a vector these are all the same vectors, so there is a rigid translation which has been done.

Now the question is whether I should first like if I want to let's say translate the curve, should I first define the curve and translate or translate the control points and define the curve again. These are the two situations. One situation is like first I have defined the curve with p_0 p_1 p_2 p_3 as control points and apply transformation on the definition of the, that is whatever is a mathematical definition which is given to Bezier curve, so that I get back this. Another this thing is if I, since these control points are used, first you transform the control points and then use the new control points to define the again Bezier curve. Both are one on the same that is why you say that the curve is invariant under transformation. Same thing is true with rotation and other transformations also. This is actually a very important property, otherwise if this property is not there then there are definitions for curves which are not invariant.

In those situations you have to also define whether your transformation is applied to a curve or to a control points or to the input which is defined which is like a very difficult thing to handle. So, very convenient thing to do is to use like Bezier curve which are invariant this thing. Same thing can be demonstrated for rotation, scaling and also for operations like shearing operations which are used about that. There is another property of Bezier curve which makes it really appealing is what is called as a convex hull property.

(Refer Slide Time: 00:31:28 min)

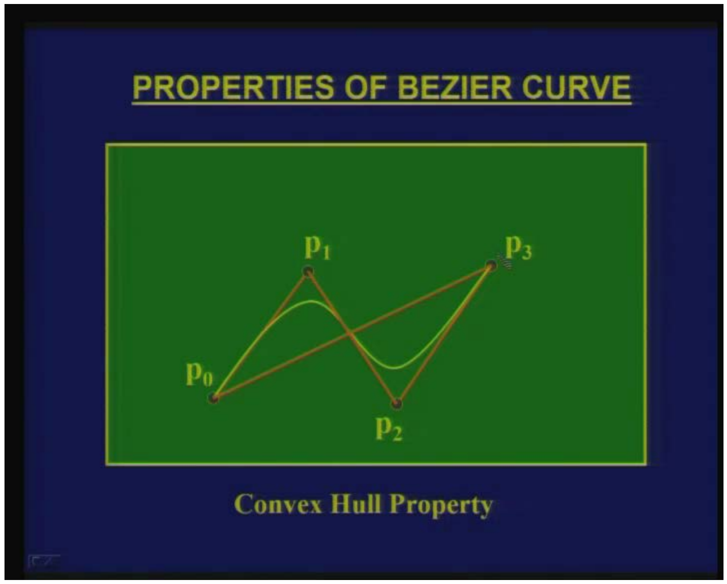


You may have come across the term convex hull. Now what is the convex hull is like convex hull is usually used to define for example often used with polygons. Suppose if I have a let's say a set of points or let's say set of vertices, if I can find out let's say a minimum size polygon which can enclose all these points then you call it as a convex hull and it should be a convex polygon. We are also familiar with what are a convex polygon and a non-convex polygon. A convex polygon is one in which if join any two vertices of the polygon, the line joining these two vertices lies completely inside the polygon. This line completely lies inside a polygon; if it doesn't then you have a non-convex polygon.

So a convex hull is suppose if I have a non-convex polygon then you are trying to define a polygon which is like a convex hull of that. That means the minimum size polygon which encloses all the points and also convex in nature. Now this I think convex hull property here is slightly different in the sense what you are trying to do is I have four control points and I can join them in a sequence 0 to 1, 1 to 2, 2 to 3 and three again back to one. So there will be a polygon which is formed which we know is called as a characteristic polygon or a control polygon. The curve always lies entirely inside the polygon. So it never goes out of the polygon, whatever may be the polygon which I am trying to use, so I will always have this property which is satisfied by the Bezier curve.

Now I think this may not be a clearly evident. Suppose if I take an example, suppose if I change the let's say order of points slightly a different. Suppose this is my first point which I am trying to use then I take this is my second point and come back to this as third and fourth. Then you have totally a different curve which is coming which is still lies inside a polygon. I think I can show you an example.

(Refer Slide Time: 00:33:55 min)



Here is an example which demonstrates the convex hull property. So what is done in this Bezier curve? Bezier curve is shown in yellow here, this is the first point, this is the second point, this is the third point and you have a fourth point which is shown here. Now, this curve has a tangent which is from 0 to 1, so the curve this is the tangent which is shown. Similarly it has tangential to last and last but one. So this is the final curve. What is the polygon here? polygon here is an intersecting, has a intersecting vertices, so it goes from p_0 to p_1, p_1 to p_2, p_2 to p_3 and if you see the curve, this polygon is in two portion, curve is also in two portion and one portion lies inside, entirely inside this triangle other portion lies entirely inside this triangle.

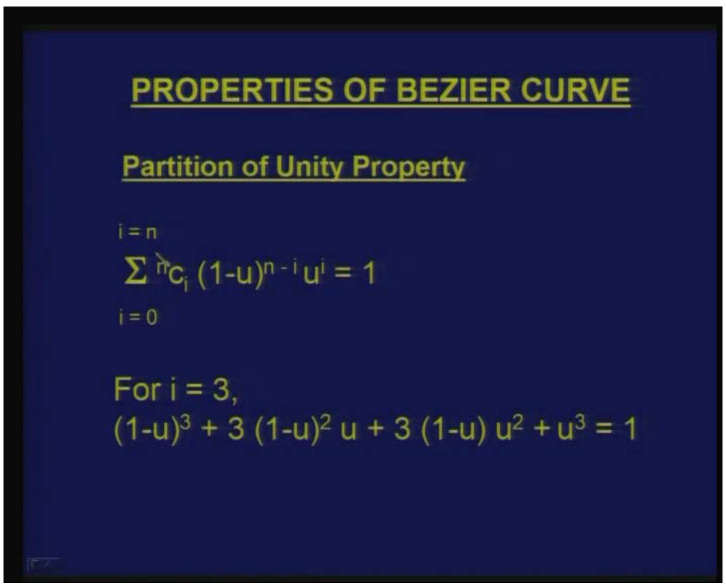
So the convex hull property is still met even when you have a control polygon or a characteristic polygon where you have intersecting edges. This is true for not necessarily for a cubic Bezier curve. This is true for higher order curves irrespective of any number of control points which I take and irrespective of any number of intersections which are happening like I may have a series of points like which are intersecting multiple times, still you will see that the curve always lies inside this particular thing.

Now this is also, this property can be used in number of ways like how this property is helpful is first thing is if there is a curve you can always set the bounds. That means if I join the control points and come up with a control polygon, you know that the curve always lies and it doesn't go outside this. So you are in a way of finding a bound for the curve is one thing. Second thing which we are also trying to use is since like it lies inside this particular a bound characteristic polygon, suppose if I want to find out let's say intersection of this curve with let's say another curve or a polygon or something or I want to find out let's say whether two Bezier curves are intersecting or not, let's say two planar Bezier curve.

What do you do, like how do I evaluate this particular aspect is since both of them are cubic in equation. If I write down the mathematical equation for curve one which is cubic, I have a second one which is also cubic and if I try to solve this is mathematically it possess certain difficulties. You have to, this is usually done numerically and you have also problems. But I can have a quick check, first find out whether the characteristic polygons are intersecting or not. Suppose if the polygons are not intersecting, you really don't have to go and find out whether the curves are intersecting.

So a lot of competition can be saved in many situations like finding the intersections etc which is usually done as a part of this. So convex hull property is used in number of ways in situations like this to find out or basically to find out the intersections or to define the curve or to find a bounds for the curve is basically one of the important properties which Bezier curve has.

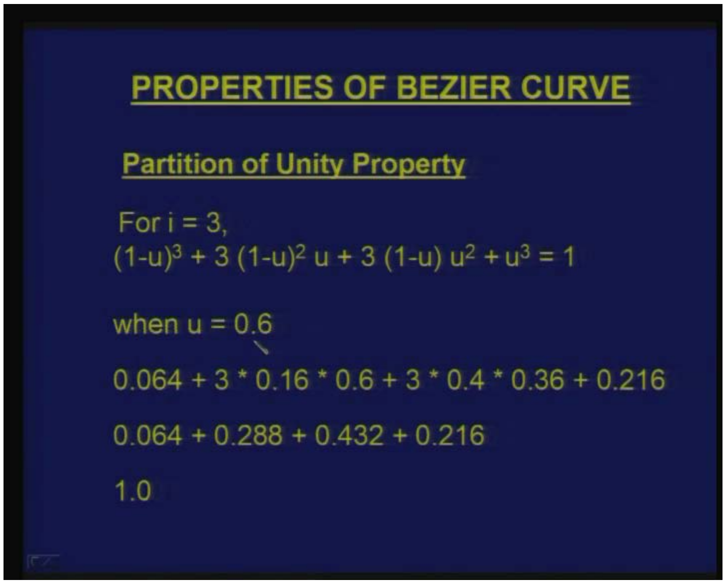
(Refer Slide Time: 00:37:43 min)



Another very interesting property of the curve is what is called as a partition of unity property like we have seen that in the case of cubic curve. Cubic curve is defined by four points like there is x_0, x_1, x_2 and x_3. There is some weightage which is given to x_0 x_1 x_2 x_3 in terms of an expression. for example if I take 3, n is equal to 3 I have this is 3 c_0 which is 1, 1 minus u to the power of cube and u to the power of 0 that is my first term, so I have this. If I remove the x zeroes, x_0 x_1 x_2 and x_3 and just add the remaining portion that means the value which is attached or you can say the value with which x_0 x_1 x_2 x_3 are multiplied and add them it is always unity.

This always unity that means whatever may be the u value, whether u is equal to 0 or u is equal to 0.2 or 0.5 or 0.8 or 1 whatever may be the u value, if I just add the magnitude with which the x_0 x_1 x_2 x_3 are multiplied it always turns out to be unity. I think this is shown here by a simple example. For example if I take cubic curve n is equal to 3, so this expression when you expand it becomes 1 minus u cube which we have seen, 3 is the binomial coefficient, the next one. Then you have 1 minus u square into u then 3 1 minus u into u square plus u cube. So now whatever is the u value which I substitute, you always get back the value which is one. Let's just check, for example if I put u is equal to 0. What happens? So you have, if u is equal to 0 all of them will be 0 and this will be 1. If I put u is equal to 1, all these things will be 0 and I have still as a 1.

(Refer Slide Time: 00:39:49 min)

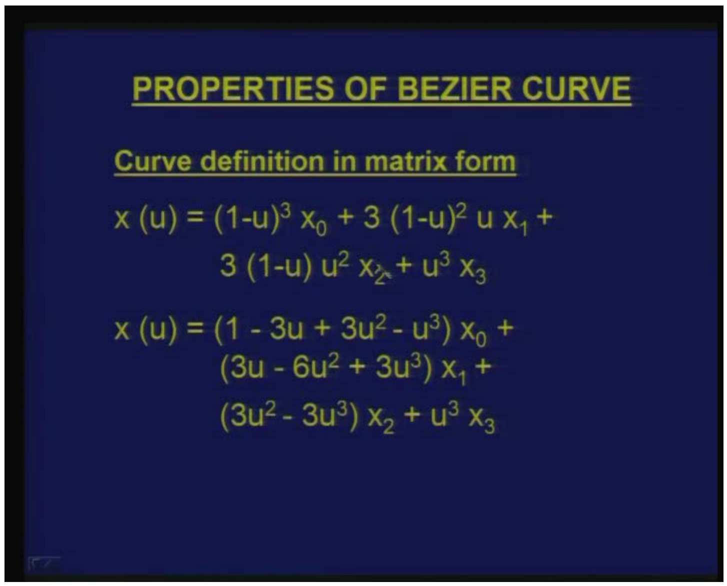


Now let me choose, let's say u is equal to 0.6 just to show this. so I have 1 minus u cube, so 1 minus u cube would be 1 minus 0.6 which is 0.4 into cube which is shown here plus you have a 3 this is 1 minus u square and into u which is shown here plus 3 which is a binomial coefficient multiplied by 1 minus u into u square which is shown here plus u cube which is nothing but cube of 0.6 which is u is equal to 0.6.

Now if I add all these terms that is the four terms which are coming that will be a value which is equal to 1. So this property is usually called as partition of unity property. This is true for other points like other Bezier curves too. For example I may go for higher order than cubic that is a quartic curve; still the partition of unity property is valid. If I go for let's say higher order still it is valid. Now, this also gives some kind of an intuition like an intuitive aspect to the curve in the sense a Bezier curve is nothing but like you are actually taking a different weightages for the curves, weightages for these four points which are trying to define. For example x coordinate is defined by only four points, four values. What is the x coordinate value at the point zero x_0 x_1 x_2 and x_3. You are actually assigning different weightages to these points as you move from u is equal to 0 to u is equal to 1. At the starting point, you give a 100% weightage to the first point and zero weightages to the other point whereas when I go to let's say the other end point, I am giving 100% weightage to the fourth point and not giving a zero weightage. By varying these weightages in certain fashion which is mathematically defined, you are actually getting the x coordinate which is different at different places.

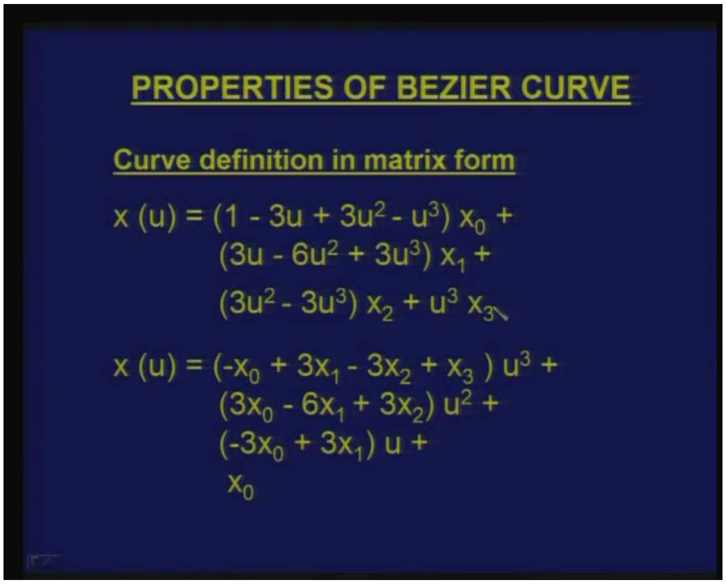
So it gives some kind of like how the curve is defined or the intuitive aspect of this particular curve. Now most of the time like we have seen that when we were discussing about the advantages of parametric representation that most of the parametric curves can be represented in a matrix form. That is very convenient when one goes for let's say programming aspect. So similarly a Bezier curve can also be represented in a matrix form. We have seen I think this for a parametric cubic curve earlier, we will see the same thing for Bezier curve.

(Refer Slide Time: 00:42:45 min)



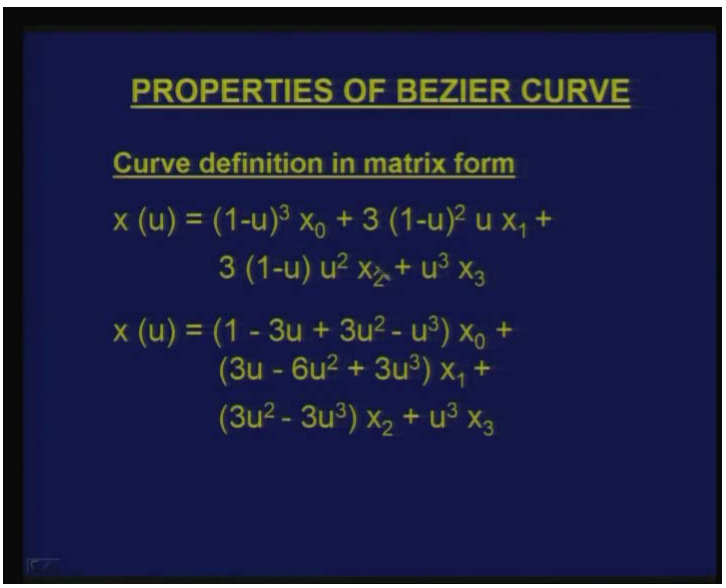
So what is shown here is this is a cubic curve which is shown here? This is the expression which I have just reproduced which we have seen again and again. Now I can expand this particular this thing. So I can expand it for 1 minus u cube and write down this like as a multiplication of x_0 x_1 and others. So these expressions are written in a expanded form.

(Refer Slide Time: 00:43:09 min)



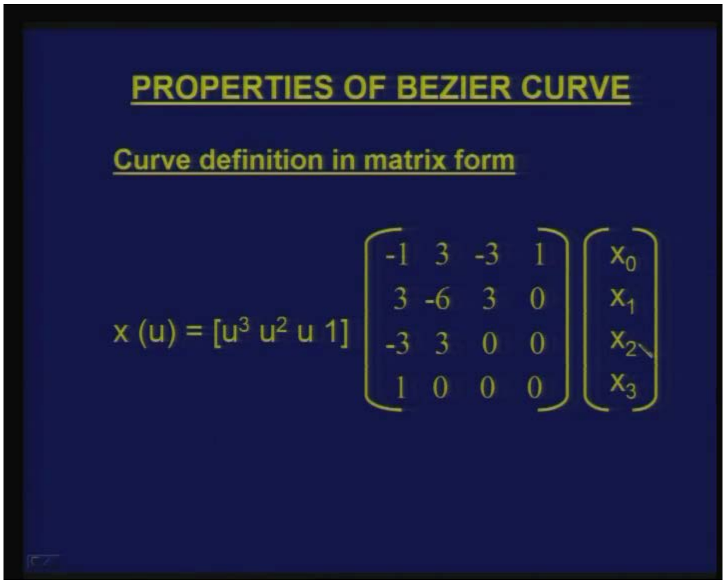
Now I can always rearrange these terms. for example instead of having the terms which are multiplications of x_0 x_1 x_2 and x_3, I would like to rearrange the times like all the terms which are u cube u square u and the constant coefficient.

(Refer Slide Time: 00:43:29 min)



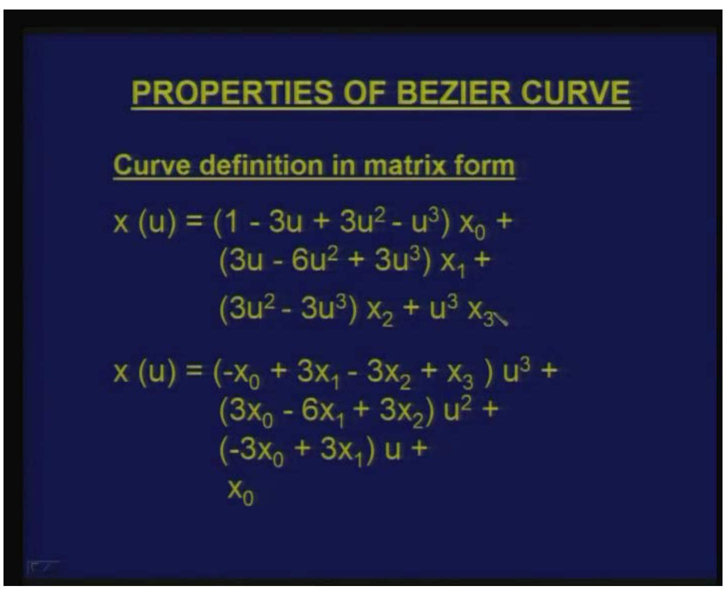
So if I write down then the same expression can be written in this manner. This is the expression where you have x_0 x_1 x_2 and x_3 terms. Here you have u cube u square u_1 constant terms which are rewritten the same expression which is there.

(Refer Slide Time: 00:43:47 min)



This expression I can also write it in a matrix form. What you see here is like if I want to know what is the x of u, so one thing is this is nothing but basically a u cube multiplied by minus x_0 plus 3 x_1 minus 3 x_2 plus x_3 which has been shown here which is in this particular case.

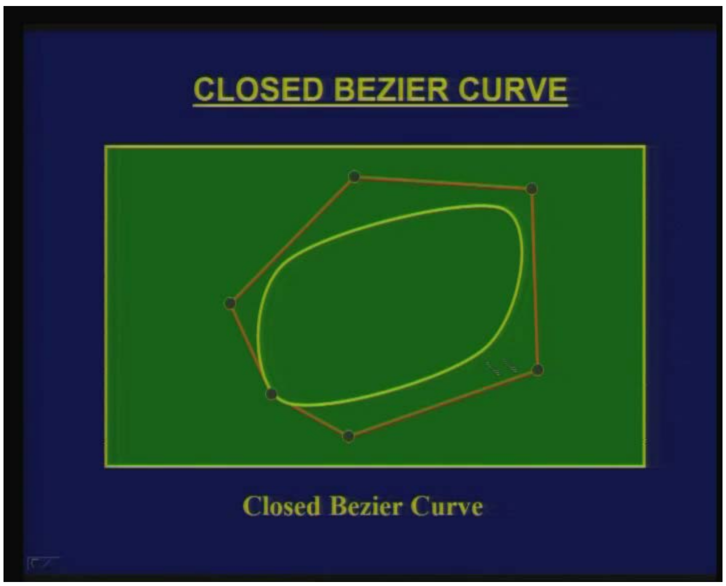
(Refer Slide Time: 00:44:13 min)



So the same thing can be written in a matrix form. Now this matrix form allows you like particularly to carry out transformations etc. Now when we said it is invariant under transformation, suppose if this curve I define a Bezier curve which may be let's say a cubic or any order and I rotate it and I translate it etc.

So what do you do is you don't disturb these two aspects of the matrix. This remains the same, you just transform these points. Applying a transformation on points is much easier than applying the transformations on a curve. So just transform these points which are shown here and again multiply with the same expression which is shown here in order to get back. So transformations are much easier to apply when you define it in a matrix form. Now what is shown here is basically for an x, similarly I can write down for an expression which is y and z which are also functions of u. And in order to make it more specific, you also say that u varies between 0 and 1 that is the parameter range which is to make it a complete expression which is there. So this is another you can say a very, it offers a simple mathematical definition which can also be written in a matrix form in a very convenient manner. Then one can also define the Bezier curve representation to define the closed curves.

(Refer Slide Time: 00:45:58 min)



Now when can you say that a Bezier curve is a closed? Suppose if I have a starting and end points which are same that means, your end point which you are defining has a same coordinates as the starting point and we know that curve passes through the first and last the points. So it will come and close this particular curve that's what is shown here. So how many points you have here? This Bezier curve is defined by 1 2 3 4 5 6 and 7 points. Starting point and end points are the same, so it passes through the first and seventh point. It doesn't pass through all the other points but still you get a very smooth curve. And you can also see that the convex hull property is still there, the curve will lie entirely in this.

And the tangent also for example the first point and the next point will define the direction of the tangent at the starting and at the end it is defined at this. So this means like in this particular situation this curve is smooth except at the starting point where there is a discontinuity in terms of slope like if I take this particular point there are two slopes at the start point and end point. One which is coming from start point which is this direction and another. So that means there is the curve is continuous in terms of points that means a disclosed but it is not continuous in terms of its slope. So what do I do like suppose if I want to make the curve which is continuous in terms of slope. I can always do it by manipulating these control points. Suppose if I make these three control points collinear, if I make them collinear then what you trying to do is that you are actually trying to have the same direction that is as far as slope is concerned at the start which is defined by start point end point is the same. Now they are coinciding, so I get a smoother curve which has continuous slope continuity over the entire period of this. And similarly if I want curvature continuity also for a cubic curve, I need to define these control points in certain manner in order to achieve this.

So cube, the Bezier curves are not only used for open curves they are also used define the smooth closed curves which can have either a continuity. now whenever a curve is defined, there is something which is I think one should also know about the continuity like if the curve is defined in such a manner that it has, there is no continuity in terms of points or I may have curve which is continuous in terms of points like all the curves which we have seen including this has continuity in terms of points. This is usually called as a c_0 continuity that means it is continuous in terms of points. If I take let's say there is a unique first derivative or unique slope at all the points on the curve, then you say that it is c_1 continuity that means one order higher.

If the second derivative is also continuous that means there is a unique value for the entire curve then it is usually called as a c_2 continues or c_2 continuity. And one of the reasons for choosing let's say a cubic curve both for Hermite and the Bezier is defined, if I am going up to cubic this c_2 continuity is met, up to second derivative I have a continuity requirements which are met which is basically what is the requirement in many of the situations which we are trying to use. So these are like some of the properties of Bezier curve and what I have discussed is like mainly the advantages but Bezier curve also has limitations.

What is the limitation? Like, suppose if I want to define a Bezier curve which is defined by 10 points. So what would be the order of curve? so you have to go for a terms which are like u to the power of 9 or 1 minus u to the power of 9 which is not like, it's not easier to handle the powers like u to the power of 9 and 1 minus u to the power of 9 in terms of like mathematically because suppose if I want to know like if I want to solve this particular equation, so i have to find all the 9 roots of equation etc which is not a very convenient in some situations. So it's better to go up to cubic. So that is one problem that means number of points control points will dictate what is the order of the curve which we are going to use. So but ideally I would like to have a cubic which is always fixed that you can only define by 4 points.

So how to overcome this is one way to do is what is called as a piecewise Bezier curve like I can take all the 9 or 10 points control points but don't define let's say curve which has a terms like u to the power of 9 etc. I define the curve which is interpolating these points by in terms of a cubic curves which are in terms of pieces just like if I have set up points, I join them as a piecewise linear segments instead of a curve. The same thing can be done for what is called as a cubic, piecewise cubic Bezier curve which is a more common representation. Another disadvantage which we will find here in terms of cubic curve is usually called as a global propagation of a curve. This disadvantage is usually called as a global, it's also a property but not let's say a property which is used for advantage which is called as a global propagation. What do you mean by that?

If suppose I have a curve which is defined by a set of control points. Let's say I use 6 control points to define let's say quantic Bezier curve or take for example this closed this thing. And I have a curve which is defined by all these control points. Now I want to change the shape of curve locally. Now if I move one of the control points, it will change the entire shape of the curve. You cannot restrict the changing the shape of curve locally because all the control points participate in terms of defining this thing. since once the control point changes, except for the starting and end points where it passes through the curve definition changes entirely or you have a small change which is made locally propagates throughout which is called as a global propagation of a Bezier curve.

So this again can be overcome if I am going for let's say a piecewise cubic Bezier curve. If I am doing it in terms of pieces, if I change the control point it will only affect one or two or may be more Bezier curves which are, in which it is participating and the propagation is not carried out. But a better way to do, a better way to take care of a global propagation would be to define another curve which is more commonly known as a B-spline curve. B-spline curve has an advantage that there is no global propagation. It is still, B-spline curve is again used to define by a control points which can be any number 10, 15 or this thing. If I am using let's say cubic B-spline line curve then changing the control point like if I change the control point, shape of the curve changes locally and it doesn't change globally.

So in order to overcome this disadvantage, the B-spline curves are used and we will study about B-spline curve in our next lecture. And then the B-spline curves can be further extended to define what we call as nurbs or non-uniform rational B-splines. If I use the rational representation of the B-spline then it becomes what is called as nurbs which are you can say a most common or a standard representation which is used by, which is used in CAD CAM industry. So in next lecture we will study about B-spline curves and we will extend the definition to nurbs also because there are lot of commonality in terms of defining a B-spline curve and a nurbs curve. And once we have defined a nurbs curve, we can extend all these definitions to surfaces like so far we have been restricting our study only to the curves but I can also extend a Hermite curve or a Bezier curve or a B-spline curve or a nurbs curve to define like a bicubic patch or let's say a Bezier patch or a B-spline patch or a nurbs patch which is nothing but a surface. We will see that as we go along this particular thing.

So I will just stop it here as far as Bezier curve discussion is concerned. If you have any questions please feel free to ask. Control in the sense, you are actually those points are used to control the shape of a curve. If I move the control point, the curve shape changes so a designer still can make use of all those control points to change the shape of the curve. So you are actually since you are controlling the shape of a curve, the word control comes. It doesn't pass through but they affect their coordinates. Yeah, you had a question. Yeah can I repeat? Yeah. See I think what, the question is what c0 c1 and c2 continuity requirement are. Now like take for example an example of a polygon.

Let's say I have a closed polygon, if I take an example of the closed polygon I see that I have let's say if I try to trace this particular polygon vertices, I can start from a point and come back to an end point without any break. But the problem with a closed polygon is when I come to a vertex I have two vertices that is basically two edges which are intersecting. So at the vertex point I do not have a unique direction or a slope, so you have a discontinuity as far as the slope is concerned.

Since, it is continuous in terms of points but not in terms of slope so, I can call it as a c_0 continuous curve. Take for example a circle. Circle is an example where if I start from a point and come back to my starting after tracing one complete looping, during my entire path there is no discontinuity in terms of slope. There is only a unique slope which is defined at all the points. So you say that circle is an example of a curve which is not only c_0 continuous because there is a point continuity but there is also a c_1 continuity there is a slope continuity.

Similarly I can extend the definition to higher order as if the curvature is continuous, i can call it as a c_2 continuous etc. Yeah any other? Yeah, affine your question is what affine transformation is. See affine transformations are as I said geometric transformation where you have translation rotation scaling. In fact the word transformation is used in many ways in graphics and CAD CAM like you also have what is called as a viewing transformation. In viewing transformation what you do is that you have an object which is sitting in a space but you are viewing from different directions. So in this case you are actually not changing the like shape or location of this particular point.

So it is like it's purely from the viewing point of view you are looking at, but geometric transformation is usually like you define a curve and then rotate or move it in a space or scale it in order to get an object. For example I want to define let's say a solid cylinder in a space which is inclined like whose access is arbitrarily inclined to three coordinate axis. So how do I do that is first I define let's say cylinder which whose axis coincidence with x axis and then apply the transformation and do let's say get back cylinder where it has to be there. That means you are placing the cylinder, orienting the cylinder wherever it is intended to be because defining directly a cylinder along an arbitrary axis may be much more difficult. So transformation helps you to do this particular thing. So I think, I will stop it here and we will take up B-spline curves in our next lecture.