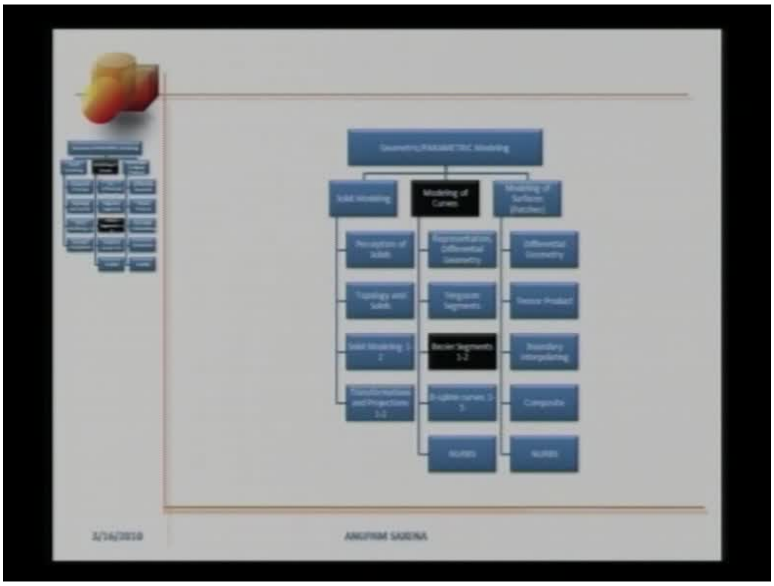
Computer Aided Engineering Design
Prof. Anupam Saxena
Department of Mechanical Engineering
Indian Institute of Technology, Kanpur

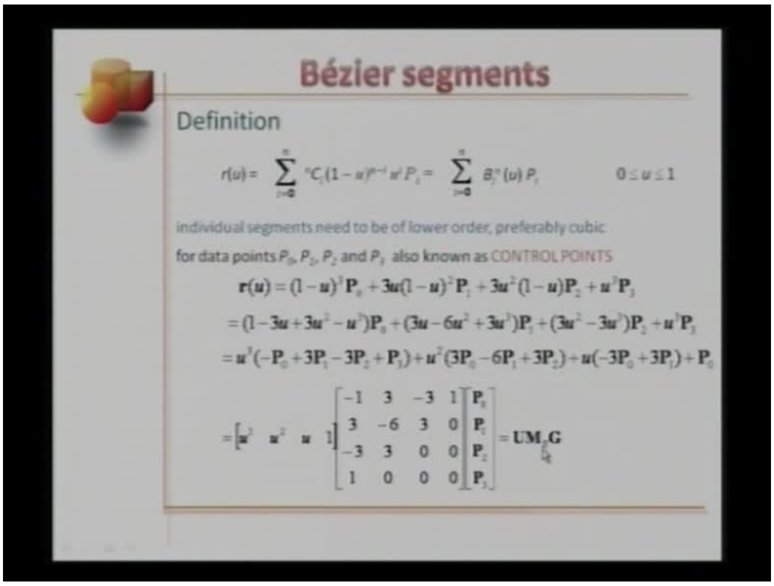
Lecture - 16
Design of Bezier Curves

Hello and welcome to lecture 16 of computer aided engineering design. We will continue with our discussion on the design of Bezier segments and curves.

(Refer Slide Time: 00:38)



(Refer Slide Time: 00:42)

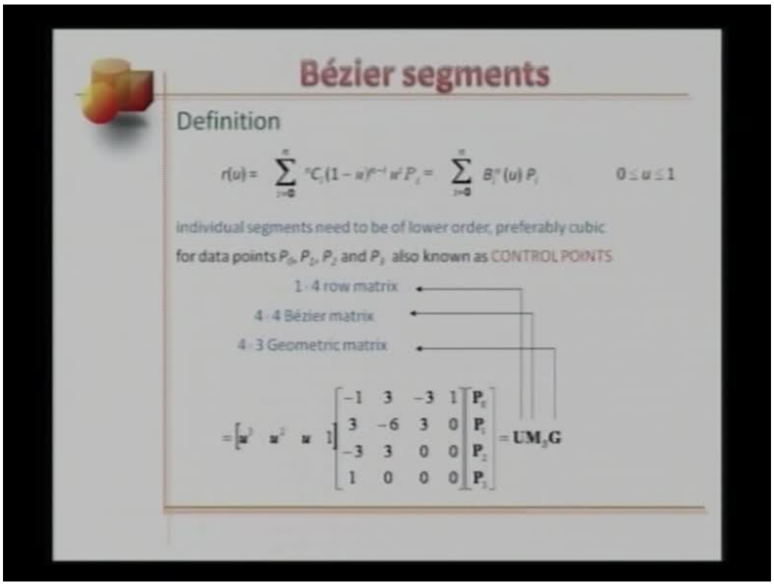


Let me warn you beforehand that mathematics is going to get quite involve from now on. We start the Bezier segments, first we have in informally introduced what Bezier segments are? But formally, this is what the definition is.

The position vector of a point on the Bezier curve is given by summation, i going from 0 to n n combination i times 1 minus u raise to n minus psi times u raise to y times p i. This expression here is our Bernstein polynomial of degree n, u is a parameter and P sub i are the design points, specified by u as a design. This is equal to summation i going from 0 to n B sub i n of u times P. The Bernstein polynomial, parameter u varies between 0 and 1. We have seen this before when we discussing Ferguson curves, and also Bezier curves that individual segments that they need to be of lower order preferably cubic.

Once again to remind us, we need this to minimize oscillations in a segment. If a Bezier segment is cubic, it will be leading four data points P 0, P 1, P 2 and P 3, we can call them as design points or control points. We have seen before that our individual terms emanating from the binomial expansion of 1 minus u plus u raise to n. You would know the expressions for the, I am going to be using those expressions directly.

(Refer Slide Time: 05:22)

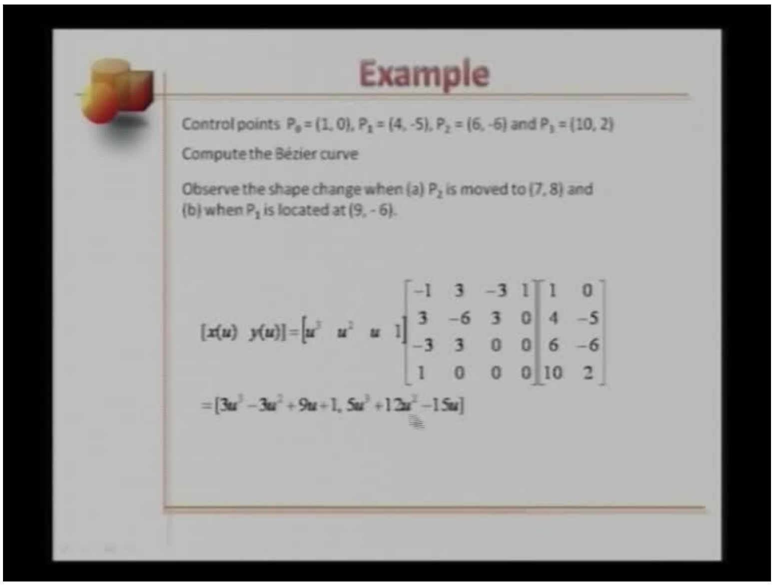


The position vector r u is given as 1 minus u raise to 3 times P 0 plus 3 times u times 1 minus u square times P 1 plus 3 u square times 1 minus u times P 2 plus u cube times P 3. I can expand these expressions and I can rearrange this equation, so that I can get the coefficient of u cube u square u and a constant 1. This coefficients are minus P 0 plus 3 P 1 minus 3 P 2 plus P 3 3 P 0 minus 6 P 1 plus 3 P 2 minus 3 P 0 plus 3 P 1 and P 0.

Like we did this in case of a cubic consequent curve, we are going to be representing this expression in the matrix form or in the compact form. This is a 1 by 4 matrix, containing the terms of u, u cube, u square, u and 1. This is a 4 by 4 constant matrix where the elements in the first row as minus 1 3 minus 3 and 1 in the second row 3 minus 6 3 and 0. For third row we have minus 3 3 0 0 and 1 0 0 0. This here is a 4 by 3 geometric matrix P 0, P 1, P 2 and P 3 constituent of the data points that for designer has specified.

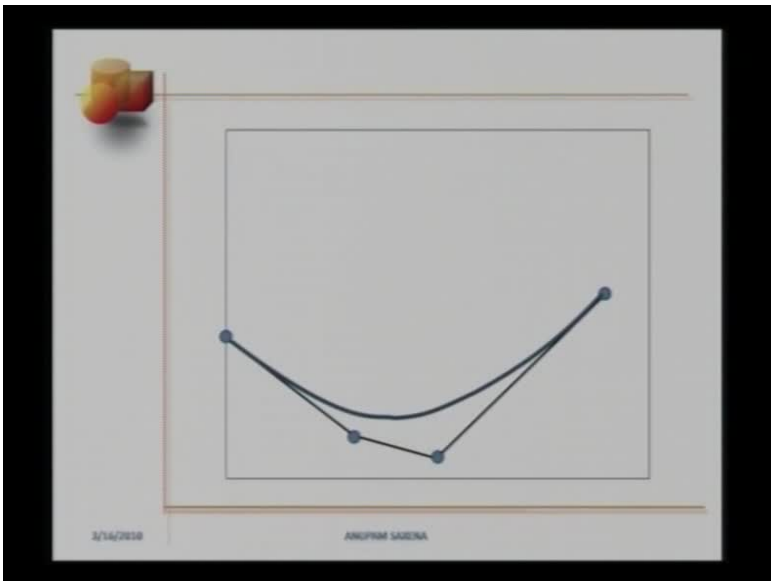
This is a very similar expression that we see in case of cubic. Ferguson curves in chart this 1 by 4 matrix is represented by capital U in bold 4 by four matrix is represented by M sub B, B refers to the B 0 segment and G is this geometric matrix. To re iterate U is 1 by 4 row matrix, M sub B is a 4 by 4 Bezier matrix and G is a 4 by 3 geometric matrix. Recall that in case of cubic Ferguson curves this geometric matrix had information pertaining to the two end points and the two end slopes.

(Refer Slide Time: 05:57)

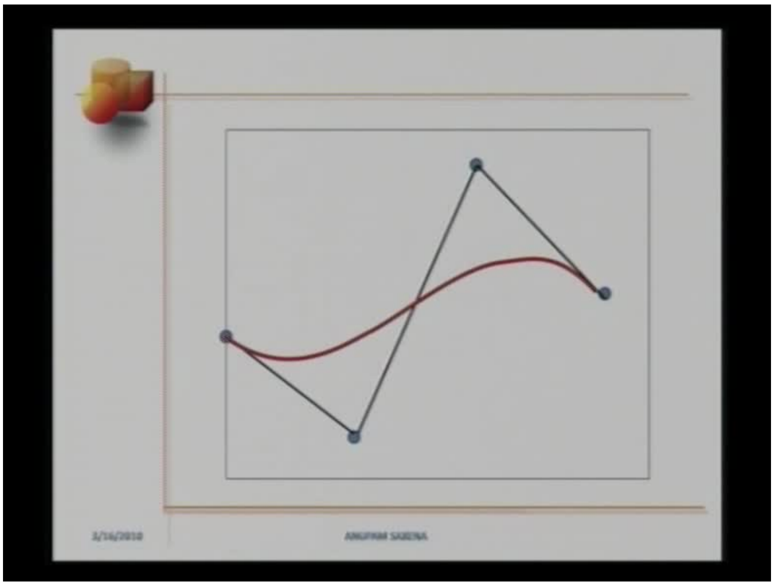


Let us take a look at an example. You have seen we are given control points P 0 by co-ordinates 1 0 P 1 by 4 and minus 5 P 2 by 6 and minus 6 and P 3 by 10 and 2. Let us compute the Bezier curve and further let us observe the shape change when point P 2 is moved from its previous location 6 minus 6 to a new location 7 and 8, following that when point P 1 is moved from its location 4 minus 5 to a new location 9 minus 6. The position vector is given by x u y u, this is a 1 by 4 4 matrix in terms of u. This is a 4 by 4 Bezier matrix and this is where we specify the geometric 1 0 for point P 1 4 minus 5 for P 2, 6 minus 6 for P 3 and 10 2 for P 4. Working out the algebra is not very difficult here. So, I will just give you the results the x component is given by 3 u cube minus 3 u square plus 9 u plus 1 and the y component is given by 5 u cube plus 12 u square minus 15 u.

(Refer Slide Time: 07:48)



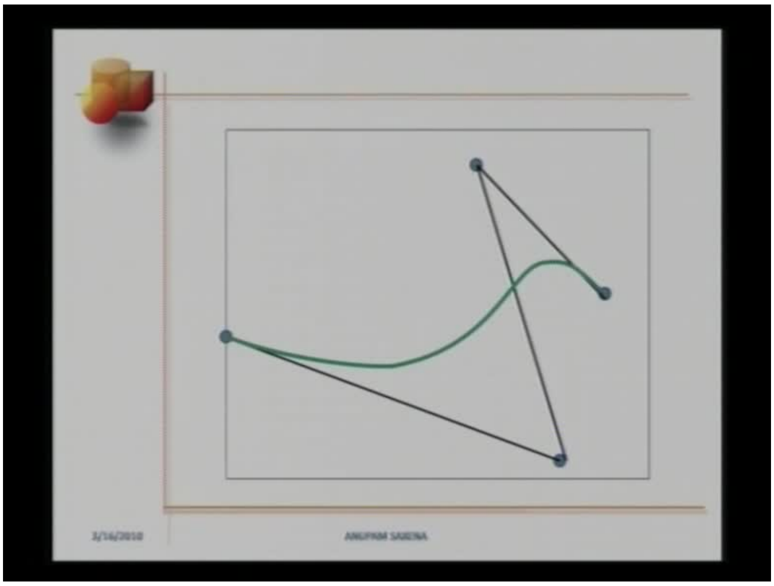
(Refer Slide Time: 08:16)



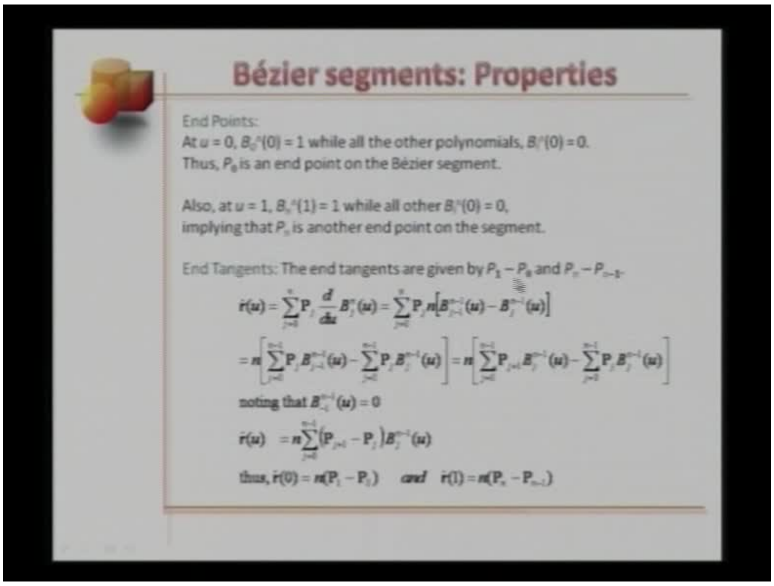
Let us look at the Bezier curve (( )). This is the original set of control points, let us joined them together to form a control parallel line. This is how the first Bezier curve looks.

Now, let us move this point to its new location. This is the resulting polyline, for which the Bezier curve is given by that in red. Let us now move this point to new respective location.

(Refer Slide Time: 08:25)



(Refer Slide Time: 10:19)



The resulting for line and resulting Bezier curve or maybe I should say these are all Bezier segments at degree 3. What to be observed? Let me go back, notice the shape of the blue segment and then towards it. Then between (( )) is the change in shape local or global. You have it right. It is in fact global; a relocation of any data point changes the shape of the entire segment. We will vary about that a little later.

Some properties of Bezier segments, we have discussed properties of Bernstein polynomial before. A lot of the properties Bezier segments and we directly derived from those of Bernstein polynomials. Let us look at some of this. First the end points of a Bezier segment at parameter value u equal 0. The first Bernstein polynomial is 1, while all the other Bernstein polynomials are 0, which would mean that P sub 0, the first design point would be one of the end point of a Bezier segment and this is true for any n degree Bezier curve.

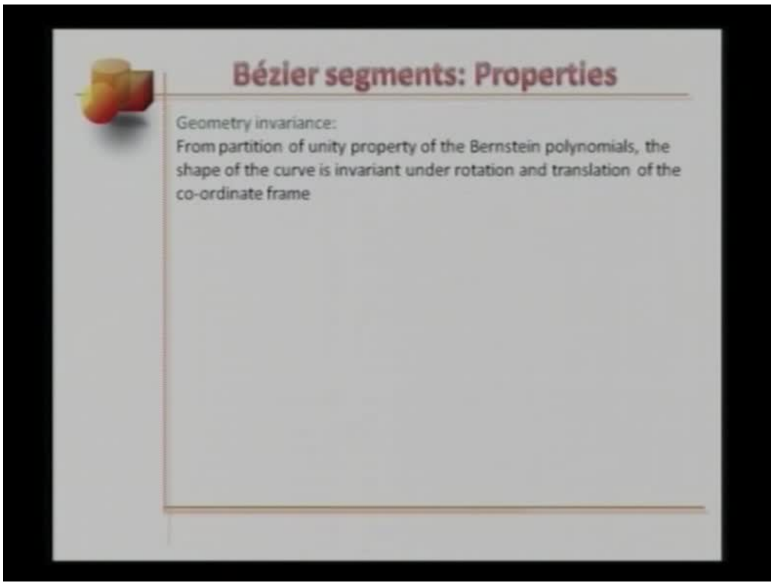
For parameter value u equals 1, the last Bernstein polynomial is 1, while all other polynomials are 0. This would mean that the last design point P sub n is another end point on the segment. End slopes or end tangents, these are given by the vectors P 1 minus P 0 and P n minus P n minus 1. These vectors would correspond to the first segment of the control polyline and the last segment of the control polyline. If you recall we have computed the first derivative of Bernstein polynomials available. If we compute this first derivative of a Bezier curve, you can find that this is given by summation j equals 0 to n P sub j times the first derivative of the Bernstein polynomial.

The j Bernstein polynomial of degree n, that is equal to j going from 0 to n P sub j n times the j minus 1 Bernstein polynomial of degree n minus 1 minus the j polynomial of degree n minus 1, we can bring in this summation sign inside. Once we do that, this expression here is equal to n times summation j going from 0 to n minus 1 B j minus 1 n minus 1. A function of u minus summation j equals 0 to n minus 1 P j B sub j n minus 1. Again a function of u and that is equal to n times summation j going from 0 to n minus 1 P j plus 1 B j n minus 1 minus j going from to n minus 1 P j B sub j super n minus 1.

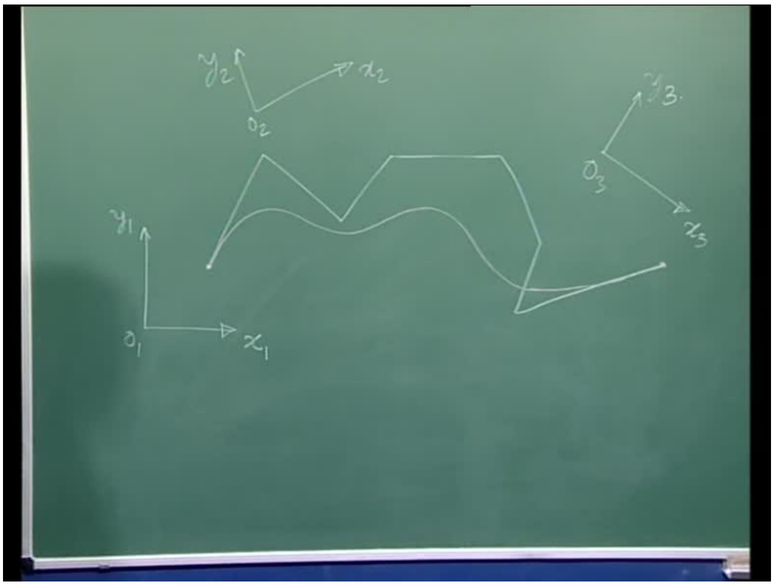
What have we done? We have simply raise the index of this term here and accordingly we have changed the limit here as well. We have the convention that B sub minus 1 super n minus 1 of view is 0. this is an exercise for you to determine, why this is so? Opnce we use this information here, the first derivative of a Bezier curve would become n times summation j going from 0 to n minus 1 P j plus 1 minus within (( )) times the j Bernstein polynomial at degree n minus 1.

Now, for u equals 0 all the other Bernstein polynomials except for the first one will be 0 and for u equals 1 all, but the last one will be 0. We use this observation to find that the first derivative of a Bezier curve at u equals 0 is given by n times P 1 minus P 0. The slope of the curve of the segment at u equals 1 is given by n times P n minus P n minus 1. We have just shown that this stat is true.

(Refer Slide Time: 15:48)



(Refer Slide Time: 16:44)



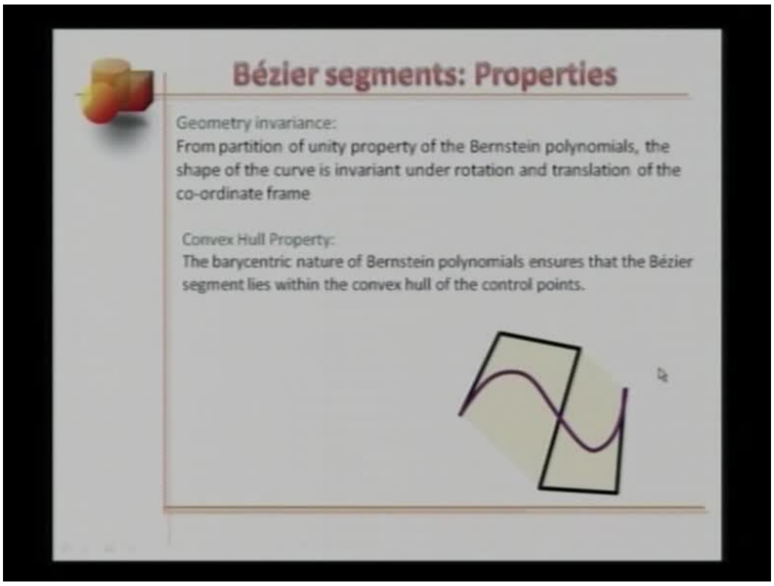
Next, geometric invariance from partition of unity property of the Bernstein polynomials the shape of the curve is invariant under rotation and translation of the co-ordinate frame.

I recommend that you represent the discussion on bary centric co-ordinates and a fine combination. There we have scared that, when the waits in the combination sum to 1, the shape of the curve does not change, when the co-ordinate system is rotated or translated.

Let me explain this concept on board. Let me draw the control parallel line of any generic Bezier curve and let me also sketch roughly the Bezier curve. You have seen that the segment will start from this point and it will end at this point. We have also seen that the first segment of the control parallel line and the last segment of the control parallel line will be tangent to the Bezier segment. This is how a typical Bezier segment will look, we roughly discus what the intermediate shape of a Bezier segment will look like? But that is for later.

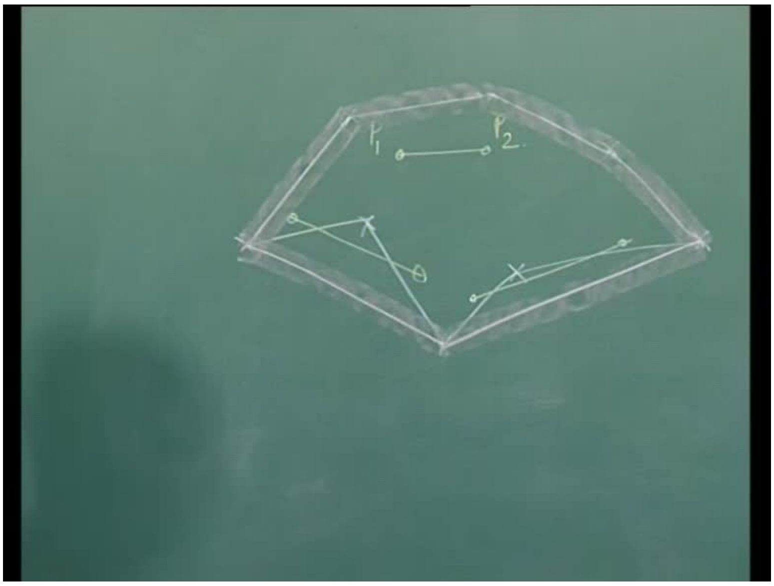
For now the geometric invariance property takes that no matter what co-ordinate system are used to express this segment whether this or whether this or whether this one. No matter what co-ordinate system are use to express the shape of a Bezier segment, this shape will remain and changed. This is due to the part that the constant polynomials, they all sum to one, which would preserve the fine combination.

(Refer Slide Time: 18:54)



This is an important property, the convex hull property. The barycentric nature of Bernstein polynomials ensures that the Bezier segment lies within the convex hull of the control points. For example, let this be a generic control polyline and correspondingly let this be the Bezier segment. The question how would you determine the convex hull or set of points. Now, I will explain that little later. But for now that the convex hull will given by this shaded area. For any value of parameter u, the entire curve will always lie within the convex hull of the given design points or data points. Let me explain, how to draw the convex hull for set of given data points?

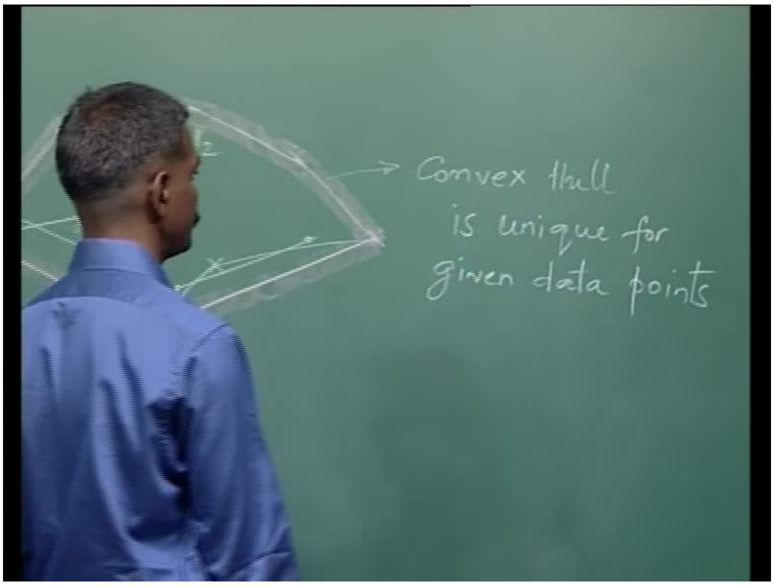
(Refer Slide Time: 20:20)



I start with first specifying the design points. Let us say this is my control polyline, let me close a lope here and let me pose this question to you is this, the convex hull for these data points specifying? The answer to my question is no, why is that so? By definition of a convex hull, if I choose any two points P 1 and P 2 within the hull. If I draw a straight line, then all the points in the straight line or in other words the entire straight line should lie within the convex hull n.

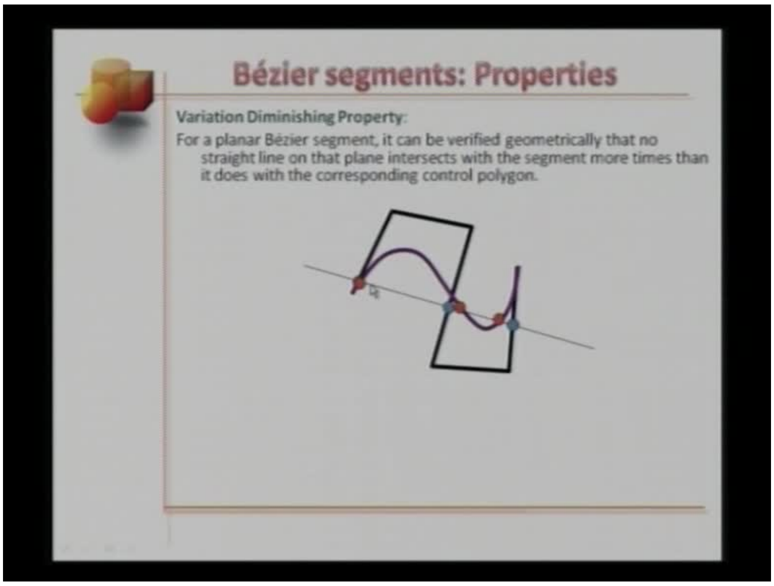
Now, let me choose two points here and here and let me join them using a line segment. You will notice that a part of this line is outside the hull. Let me choose another set of intermediate points and let me join them again using a line segment once again. This portion of the segment is outside the hull, the actual convex hull, which is unique for a given some data points and we sketched like… So, you can use this definition and verify that for any two points within this hull you corresponding line segment joining the two points will always lie with the convex hull. Note that this convex hull is unique for given data points.

(Refer Slide Time: 23:00)



We will further into investigating the properties of Bezier segments variation diminishing property here. We see or we observe, how we can roughly defect the shape of a Bezier segment for a planar Bezier segment.

(Refer Slide Time: 23:32)

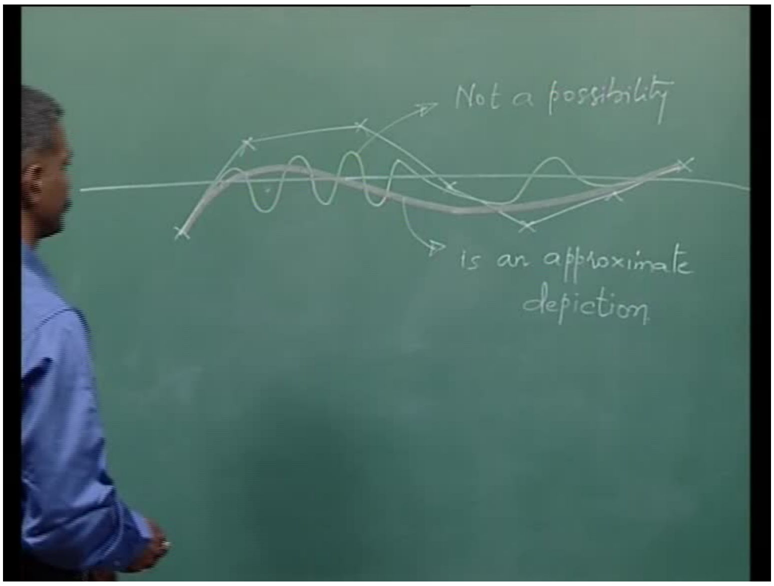


It can be verified geometrically that no straight line on that plane intersects with the segment more times than it does with the corresponding control polygon. Do not worry about the grammar of this long statement, I will explain this few with an example. Let me start with a generic control polygon again. Let me draw the corresponding Bezier segment, this is a two-dimensional example and extension of this example in 3 D would always also work. Now, let me first draw a line segment that intersects with the control parallel line and the Bezier segment.

In other words, this line segment is going to be cutting both the control parallel line and the Bezier segment that is first part of the intersect points between this line and the control parallel line. These are shown using blue circles, let us now observe the section points between the Bezier segment and this intersecting line those are shown using red circles. Once again, the blue circles are the intersection points between the control parallel line and the intersecting line. The red circles are the intersection points between the Bezier segment and this line segment. What do we notice?

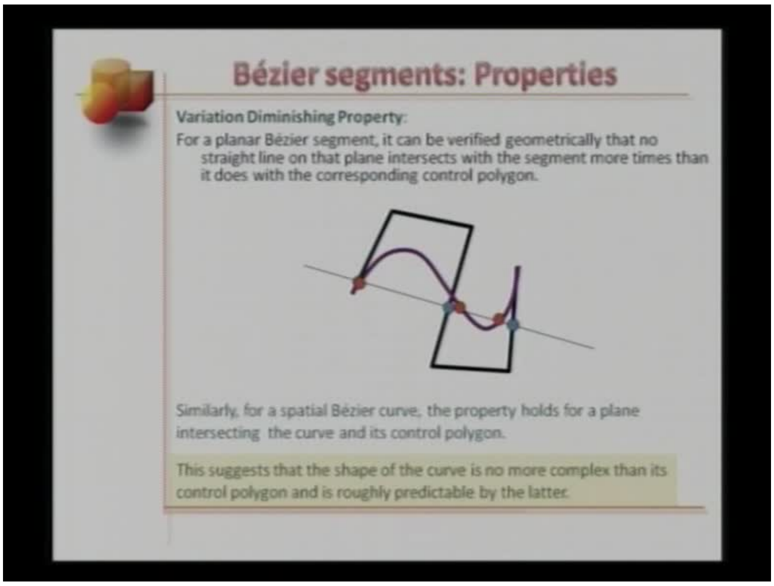
We notice in this example that the intersection points in both cases in number are the same. 3 blue circles and the 3 red circles. What this property says is the following; it will never ever happening, that the number of red circles will be more than the number of blue circles. In other words the number of intersection points between the segment and this line will never be more than the number of intersection points between the control parallel line and intersecting line. This would basically, mean that the shape this Bezier segment will be no more complex than the shape of the control parallel line itself. In other words this control parallel line roughly defects or approximates the shape of the Bezier curve.

(Refer Slide Time: 27:18)



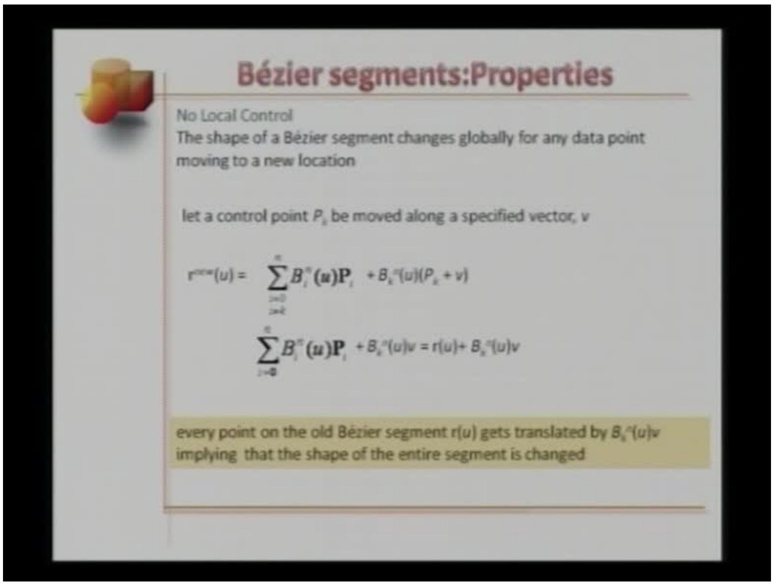
I specified the data points again and I sketch the control parallel line. Let me sketch an arbiter Bezier segment starting from the first point ending at the last point. Once again the first and the last segments of the control parallel line attaches to the Bezier curve. This assumes for now that the shape of the Bezier curve look like this. Let me draw an intersecting line, you would notice that the number of intersection points between the Bezier segment and this line would be a lot more than the number of intersection points between this control parallel line and this intersecting segment. What the property says is that such or scenario is never possible. In fact the actual Bezier segment and we roughly sketched like, so to re empathies this is not a possibility. This is an approximate depiction.

(Refer Slide Time: 29:50)



In a 3 dimensional case for a spatial Bezier curve, the property holds for a plane intersecting a 3 dimensional Bezier segment and its control polygon. This is what I have said before that the shape of a Bezier segment is no more complex than its control polygon and it is roughly predictable by the line.

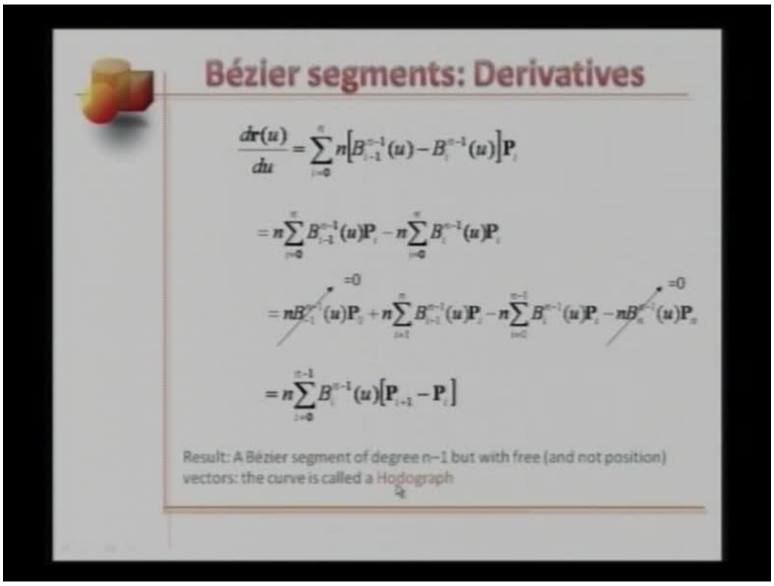
(Refer Slide Time: 30:23)



Continue with the properties, no local control the shape of a Bezier segment changes globally for any data point moving to a new location. Let us try to analyze this property with algebra. Let a control point P sub k be relocated along a specified vector v. So, the new Bezier segment r super mu will be given by summation i going from 0 to n and i naught equal to k. The terms in the summation are the i Bernstein polynomial of n times the control points, P i plus the take Bernstein polynomial of this union times the new relocated control point P sub k plus v.

What we can do is we can bring this term back into this summation to get i going from 0 to n and summation terms B sub I super n times. That is end point plus B sub k super n times the vector v, this is equal to the original Bezier segment plus an additional term B sub k super n a function of u times the vector v. We see that every point on the old Bezier segment r u gets translated by term B sub k super n times v. So, note that this is actually a vector, so individual components x, y and z of the Bezier segment will get translate. Once again every point on the old Bezier segment is translated by term B sub k super n times v implying that the shape of the entire segment gets changed.

(Refer Slide Time: 33:16)

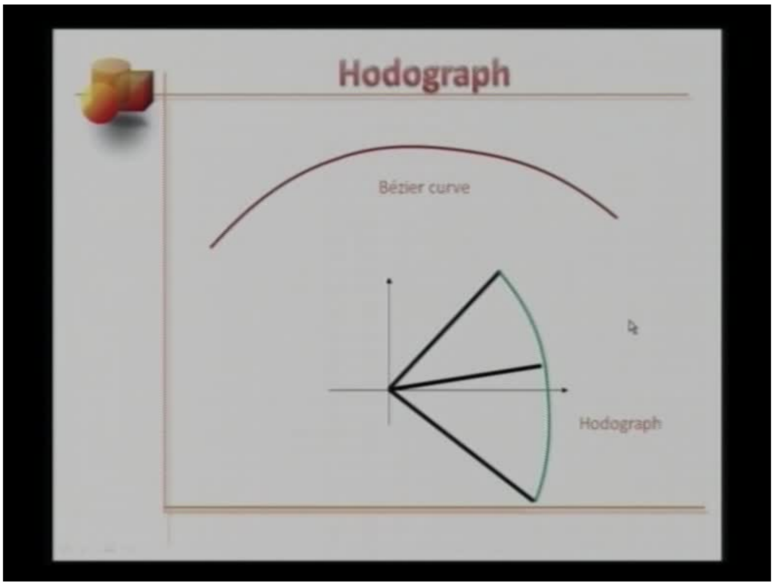


This look at some derivatives of Bezier segments, the first derivative total r u over total u is given by summation i going from 0 to n n times the i minus 1 Bernstein polynomial of degree n minus 1 minus the i Bernstein polynomial of degree n minus 1 times the design point P i. This term here is nothing but the first derivative of Bernstein polynomial B i super n. We can take this term in out and expand this expression to get n summation i going from 0 to n B i minus 1 super n minus 1 times P i minus n times summation i going from 0 to n B sub i super n minus 1 times P i.

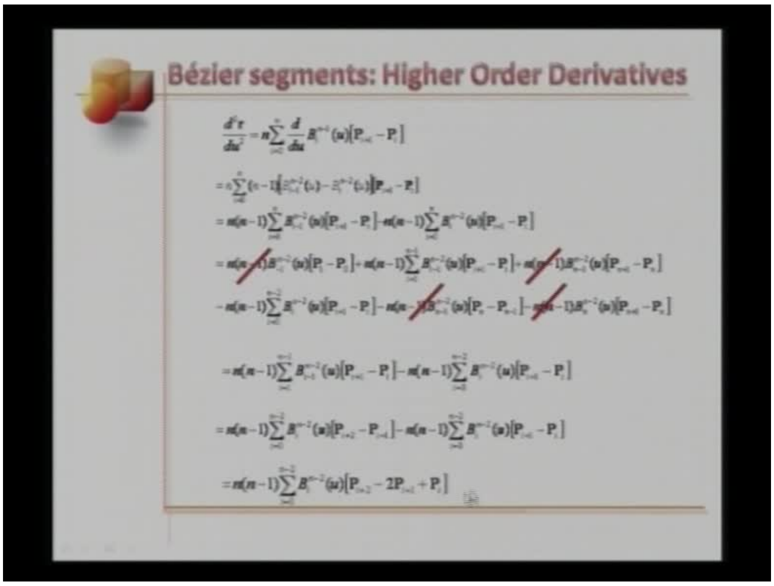
What we do is we write the first term of this expression separately and we write the last of this expression again separately. So, this is the first term in this expression or in this summation and this is the last term in this summation and these are the rest of the term in this expression. This is n times B minus 1 raise to n minus 1 times P 0 plus n times summation i going from 1 to n B i minus 1 n minus 1 times P i minus n times i going from 0 to n minus 1 B sub i super n minus 1 P i minus n times B sub n super n minus 1 times P n by convention B minus 1 n minus 1 is 0.

Likewise, B sub n super n minus 1 is 0. This is an exercise for you to find, why is this? So, we let with these two terms and we combined them together to get n times summation i going from 0 to n minus 1, the i Bernstein polynomial or degree n minus 1 times within parenthesis the i plus 1 design point minus P i design point. What do we see? If you try to compare this expression within end degree Bezier segment, we observe that this is very similar to that the only difference there we had the position vectors of the design points, but here we have free vectors. The degree of this curve will be n minus 1. So, a Bezier segment of degree n minus 1, but with free vectors is attend and this segment is called a Hodograph.

(Refer Slide Time: 37:30)



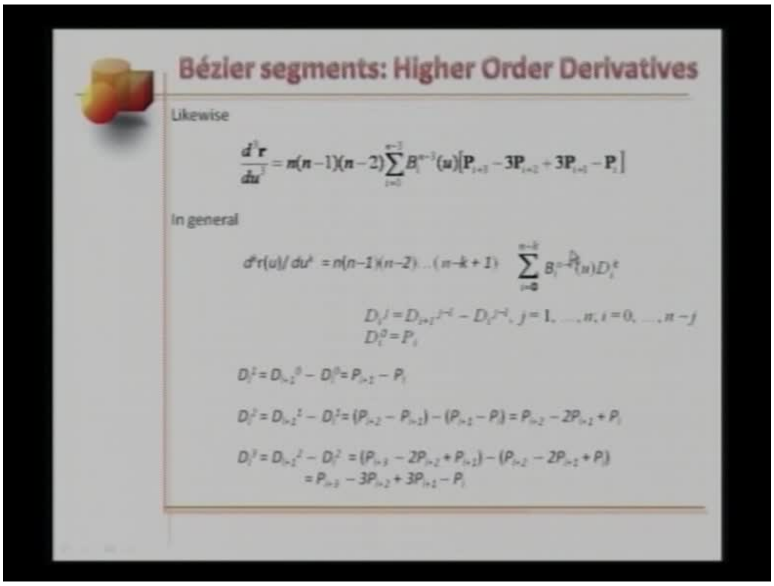
(Refer Slide Time: 38:58)



How would a hodograph look? Let us consider cubic Bezier segment. Notice that these vectors are free vectors. this is P 1 minus P 0 P 2 minus P 1 and P 3 minus P 2. This would mean that, I can move around these vectors any way I want. This curve is a Bezier curve. This move these free vectors at well. I have shown here only one of the cases rather one of the remaining cases. i coincide the tales of all this free vectors at the origin of the coordinate system. This here will lie on this green curve, which is the hodograph. In any other case, I could move these three vectors anywhere else I would want. What am I suggesting? That that this green curve the hodograph is not unique.

Let look at some higher order derivatives of Bezier segments d 2 r over d u square is given by n times summation i going from 0 to n d over d u of B sub i super n minus 1 times P i plus 1 minus P i. This expression here relates to the first derivative of the Bezier segment. I can work the algebra in detail, let me not vary about the intermediate steps, but I will have them for you on the slide just in case you want to take a look at least at later on. Note here, this term here will become 0 because there was a negative one index here for the Bernstein polynomial. Likewise, this term here again will become 0 because the index over here is larger than the degree of the Bernstein polynomial. By convention we make these Bernstein polynomial 0.

(Refer Slide Time: 42:18)



Of course, there is a reason and I have asked you to find out this reason yourself. Likewise, here the index is larger than the degree of the Bernstein polynomial and source the case here. Any how this is, what we find results n times n minus 1 summation i going from 1 to n minus 1, the i minus 1 Bernstein polynomial of degree n minus 2 times i plus 1 design point minus the i design points minus n times n minus 1 summation i going from 0 to n minus 2 the i eth Bernstein polynomial of degree n minus 2 times P i plus 1 minus P i. We rearrange this expression to get the final result as n times n minus 1 summation i going from 0 to n minus 2. The i th Bernstein polynomial of degree n minus 2 now times P i plus 1 minus 2 P i plus 1 plus p i. If you carefully observe, this is actually P i plus 2 minus P i plus 1 minus of P i plus 1 minus P i in parenthesis.

Likewise, we can compute the third derivative of a Bezier segments. That is given by n times n minus 1 times n minus 2 summation i going from 0 to n minus 3 B sub i super n minus 3 times P i plus 3 minus 3 P i plus 2 plus 3 P i plus 1 minus P i. The algebra should not be very difficult, it should be straight forward. You should be able to compute and verify this result. We can generalize these results, we can say the k eth derivative of a Bezier segment is given by n times n minus 1 times n minus 2 up to n minus k plus 1 summation i going from 0 to n minus k.

What do you expect now? You are right, you are expecting the i eth Bernstein polynomial of degree n minus k. Observe like here, i eth Bernstein polynomial of degree n minus 3 for the third derivative. What have done is I have generalized these expressions by D sub i super k. If i start with D sub i super 0 as my original control points P sub i, then I can have a recursive relation given by d sub i super j is equal to D i plus 1 j minus 1 minus D i j minus 1 j would represent the derivative. For example, j is equal to 1 is give the first derivative j is equal to 2, when j gives the second derivative and so on and so forth. j is an index at goes from 1 to n and i is index at goes from 0 to n minus j.

For example, D i 1 is equal to D i plus 1 0 minus D i 0. This is P i plus 1, this is P i. D i 2 is equal to D i plus 1 1 minus D i 1. What is D i plus 1 1, we use a similar expression. This is P i plus 2 minus P i plus 1 minus D i 1. Get this expression directly P i plus 1 minus p i. A little rearrangement, I will give you P i plus 2 minus 2 times P i plus 1 plus P i. Likewise, what is D sub i super 3? To get D sub i plus 1 super 2 all we need to do is raise the index of this by 1. That is correct, we get P i plus 3 minus 2 P i plus 2 plus P i plus 1. We already have the expression for this P i 2 like here. This is P i plus 2 minus 2 P i plus 1 plus P i rearrange to get P i plus 3 minus 3 times P i plus 2 plus 3 times P i plus 1 minus P I, which is the expression like here.