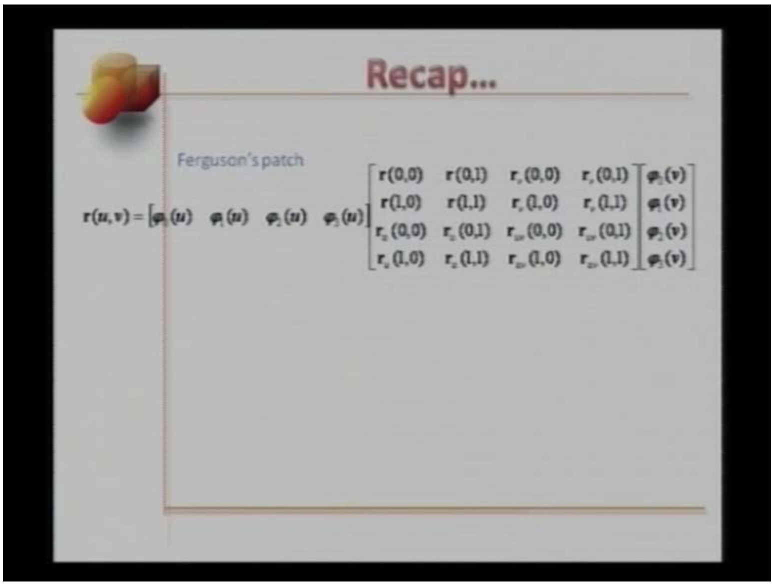
Computer Aided Engineering Design

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

Lecture - 37

Good morning. We have been discussing different models of surface patches. Just a little recap on what we did last time.

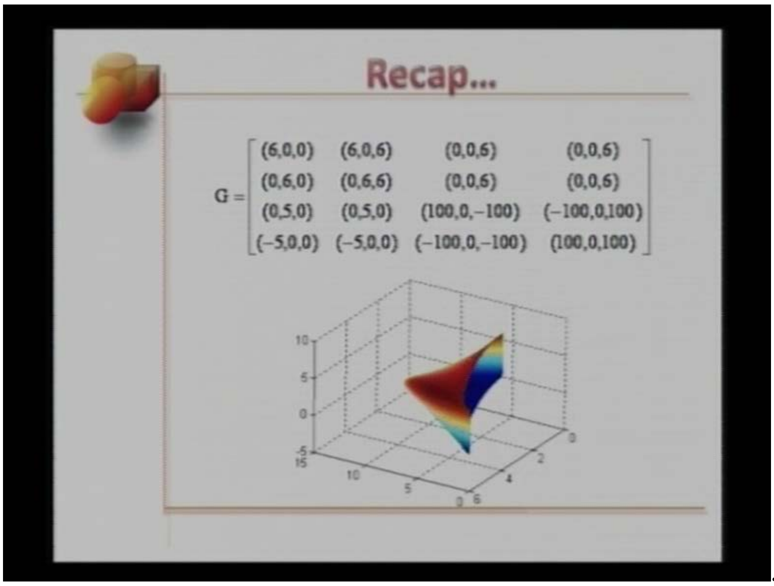
(Refer Slide Time: 00:30)



We studied Ferguson's patch in quite some detail. So, the mathematical expression for the patch is r of u and v is given as the row matrix phi 0 phi 1 phi 2 and phi 3; all functions of parameter u times the geometric matrix, I will come to that little later, times the column vector phi 0 phi 1 phi 2 and phi 3; all functions of parameter v. These phi's happen to be cubic Hermite polynomials as you would know from Ferguson curves, this geometric matrix here, we can think of sub dividing this into four parts, draw a vertical line here and draw a horizontal line here. This part here contains the point information evaluated at u equals 0 v equals 0, u equals 0 v equals 1, u equals 1 v equals 0, and u and v both equal to 1. This region here comprises information pertaining to this slopes along the v direction are sub v which is also partial r over partial v, again evaluated in the same order as these four numbers.

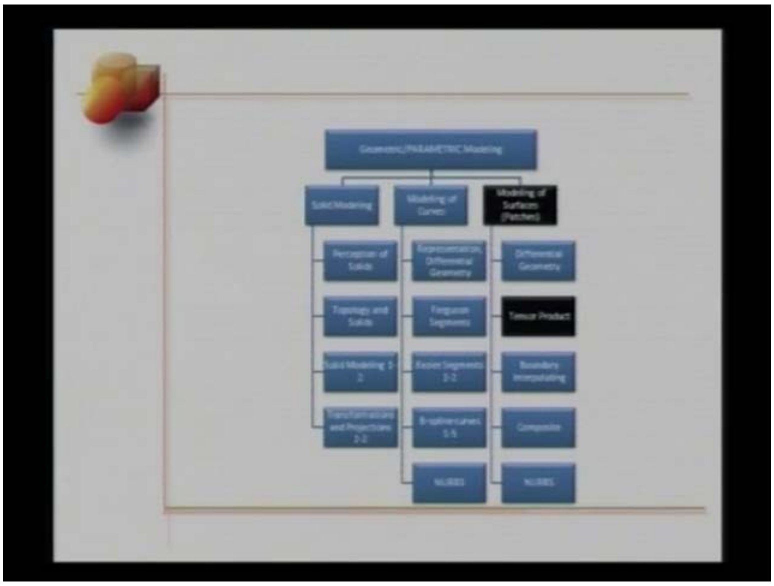
The bottom left region comprises of the slope information along the u direction r sub u or partial r over partial u evaluated as 00, 01, 10, and 11. Finally, the bottom right portion here comprises of twist vectors given by the mixed derivative, the mixed second derivative of r with respect to parameters u and v. We had mentioned last time that as a designer it is not very straight forward for us to provide higher order information. For example, slopes along the respective parameter directions and with respects. That is one prime motivation as to why we would want to consider surface patch models that involve specifying only get a points.

(Refer Slide Time: 03:35)



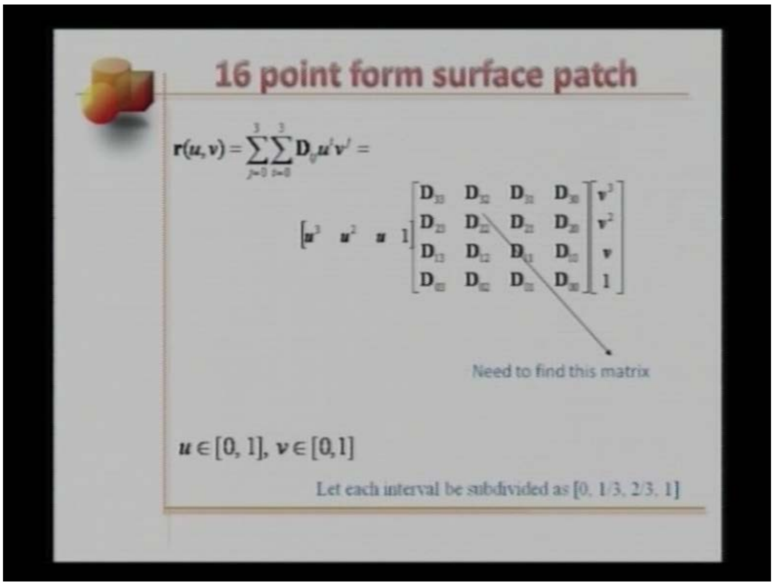
Nevertheless, we looked at 2 examples last time, one corresponding to this geometry matrix where the twist vectors were specified 0 and the slopes along the v parameter direction and the slopes along the u parameter direction were specified arbitraries. This is the result that we got for this geometric matrix and if we specified the twist vectors to be non-zero and arbitrary, for example, through this geometric matrix we got a significantly different result. The point that I am making here is that it is not in curative for us to predict shape changes if we specify the slope and the twist vector information in different ways.

(Refer Slide Time: 04:46)



Today we are going to be looking at tensor product surfaces that use data point specification. This is lecture number 37 on Tensor product surface patches.

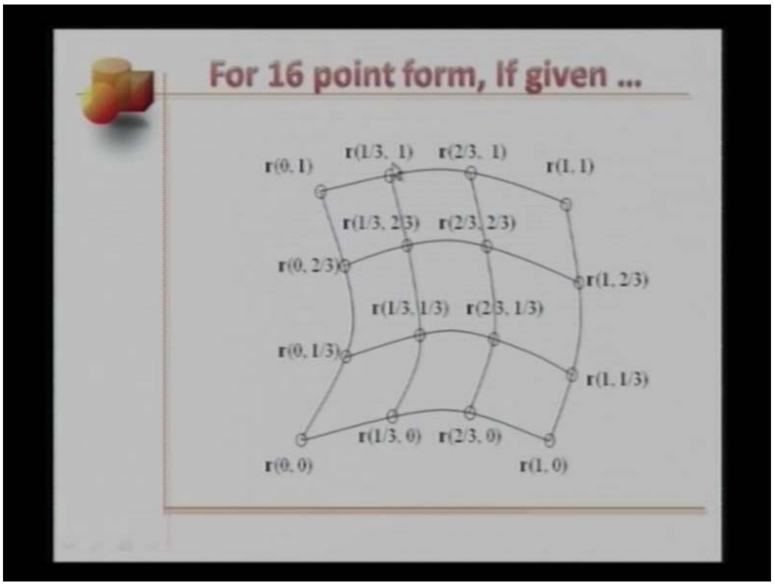
(Refer Slide Time: 05:08)



The first model is 16 point form surface patch. Mathematically the model is expressed as r as a function of parameter u and v is equal to summation index j going from 0 to 3, again summation index i now going from 0 to 3 times d sub i j times u raised to i times v raised to j. In matrix form, this can be written as a row matrix comprising u cube, u square, u, and 1, a geometric matrix where co-efficients D 33 D 32 D 31 D 30 as the first row; D 23 D 22 D 21 D 20 the second row; D 13 D 12 D 11 and D 10 for the third row, and finally, in the last row D 03 D 02 D 01 and D 00. This is to be post multiplied by the column vectors comprising different degrees in v; v cube, v square, v 1, and 1. It is this geometric matrix that we need to determine. We will see in a while, how.

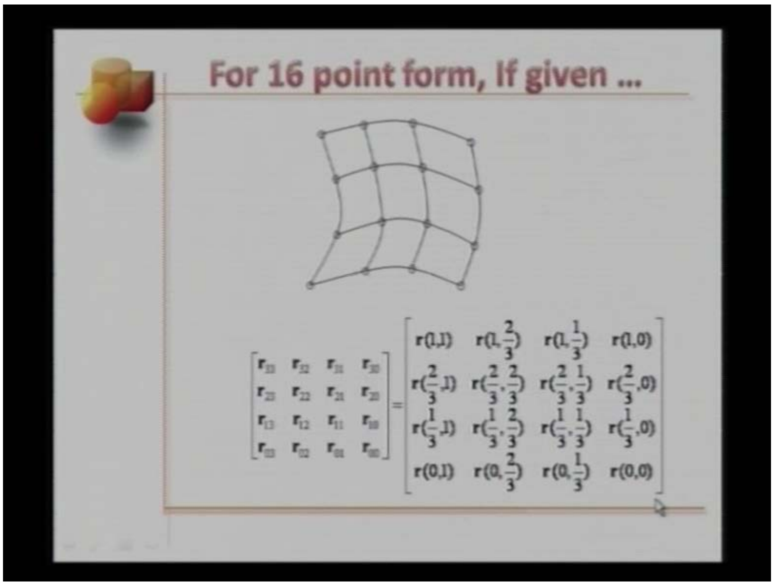
Let me emphasize the parameter values u and v both range between 0 and 1. To be able to determine what the D ij can be, let us sub divide each interval 0,1 as 0, 1 over 3, 2 over 3, and 1 and try to sample data points at these parameter values. So, the idea is that we have four values for parameter u: 0, 1 by 3, 2 by 3, and 1, and likewise we have four identical values for parameter v.

(Refer Slide Time: 08:37)



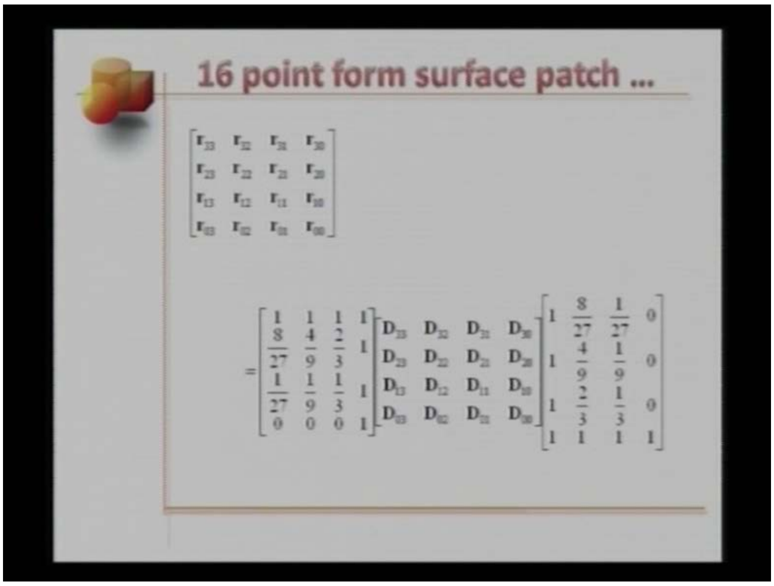
Correspondingly, we will be having 16 sample points that we would need to use to determine the unknown coefficient matrix. Geometrically, this is how the sampling data would look like. We will require point information pertaining to r 00, r 1 over 30, r 2 over 30, r 10 for values of v as 0 and for different values of u along this curve and then r 0 1 over 3, r 1 over 3 1 over 3, r 2 over 3 1 over 3 and r 1 1 over 3. The 9th data point or sample point will be r 0 2 over 3, r 1 over 3, 2 over 3, r 2 over 3, 2 over 3, and r 1 2 over 3. The final four points will be sampled at values of u as 0, 1 over 3, 2 over 3, and 1, and the values of v at 1 1 1 and 1. So, you can notice these are 16 sample points on the surface patch that we will be using to determine the unknown coefficient matrix D ij.

(Refer Slide Time: 10:17)



So, this the same patch with the notations removed. So, if you notice here, this matrix is the sample point matrix we will be using determine D ij. r 33 is r 11, r 32 is r 1, 2 over 3; r 31 is r 1, 1 over 3; r 30 is r 1, 0. Correspondingly, r 23, r 22, r 21, and r 20 are sample points for values of u as 2 over 3 and values of v as 1, 2 over 3, 1 over 3, and 0. r 13 r 12 r 11 and r 10, again are values for u equals 1 over 3 and for v going from 1 to 0 uniformly in four steps. Let us assume that we know the geometric information encapsulated within this matrix.

(Refer Slide Time: 12:01)

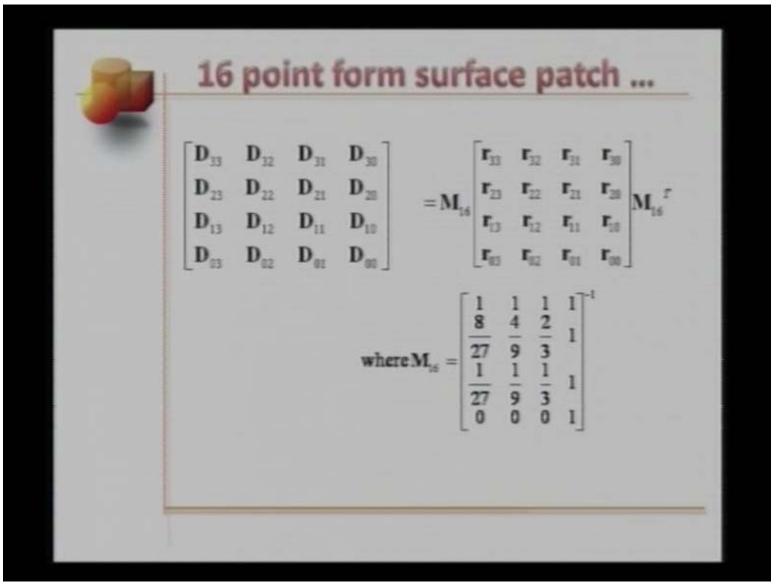


So, essentially, we know the 16 sample points. On this 16 points form surface patch, this would be equal to this 4 by 4 matrix here with elements 1, 1, 1, 1, 8 over 27, 4 over 9, 2 over 3, 1, 1 over 27, 1 over 9, 1 over 3, and 1 and 0, 0, 0, 1. Now, if you notice, each row from left to right would correspond to powers of u cube, u square, u, and 1. The first row corresponds to u equal to 1; the second row corresponds to u equals 2 over 3; the third row is for u equals 1 over 3 and fourth row is for u equals 0, and then we will have this 4 by 4 unknown coefficient matrix.

And again, we will have a 4 by 4 matrix corresponding to different powers of the second parameter v arranged column wise. The first column here corresponds to 1, second column here corresponds to v equals 2 over 3, the third column corresponds to v equals 1 over 3, and the fourth column is for v equals 0. The first row relates to v cube, the second row corresponds to v squared, the third row is for v raised to 1, and the fourth row is simply 1.

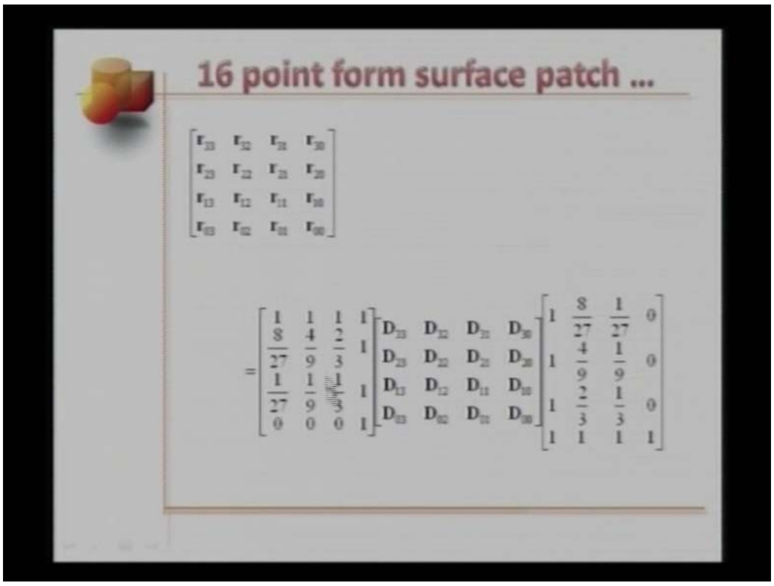
So, what is this situation now? We have this geometric sampling information on the left hand side. We know what this matrix is; we know what this matrix is. All we need to do is we need to pre multiply this matrix on the left hand side by the inverse of this matrix here and we need to post multiply the result here by the inverse of this matrix to get the unknown D ijs.

(Refer Slide Time: 14:57)

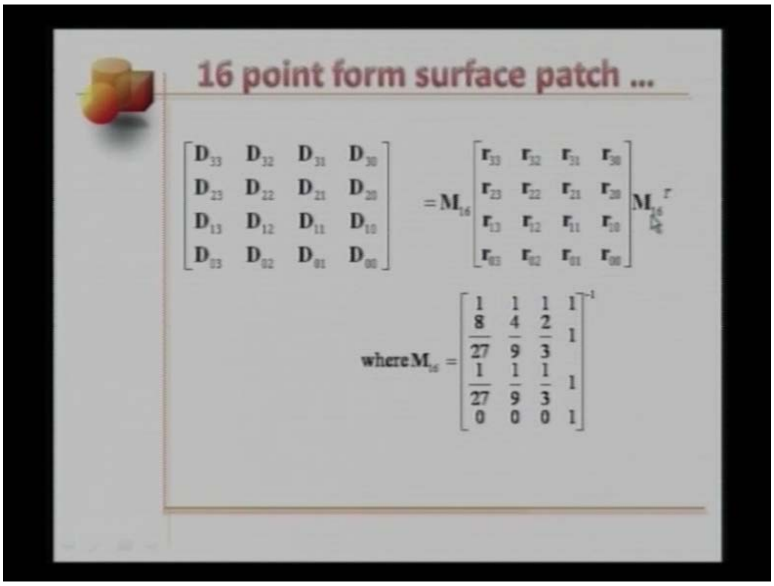


Let us continue. So, the unknown coefficient matrix D ij will be given as M 16, I will tell you what this is in a while, times the sampling geometric matrix times M 16 transpose. What is M 16? M 16 is given by the inverse of the 4 by 4 matrix 1, 1, 1, 1, 8 over 27, 4 over 9, 2 over 3 1, 1 over 27, 1 over 9, 1 over 3, 1, and 0, 0, 0, 1. One thing that you would want to realize and this is interesting is that I am using transpose of M 16 here. Why is that?

(Refer Slide Time: 12:01)

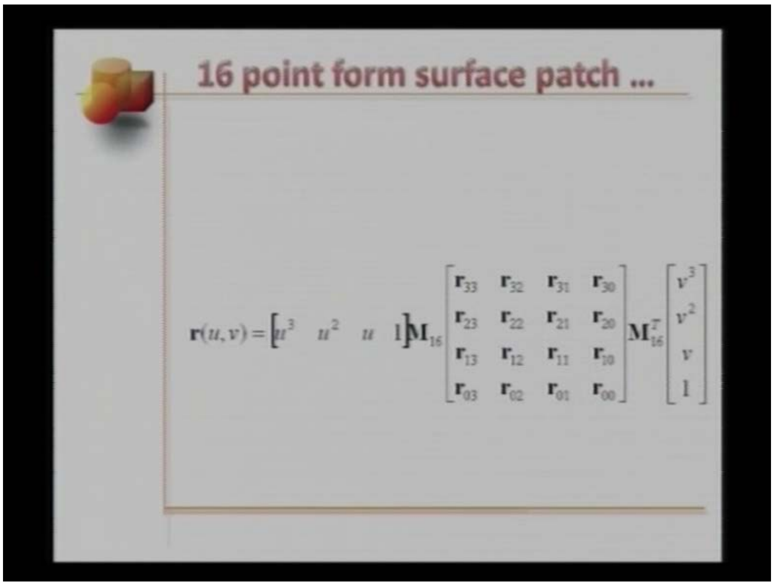


(Refer Slide Time: 14:57)

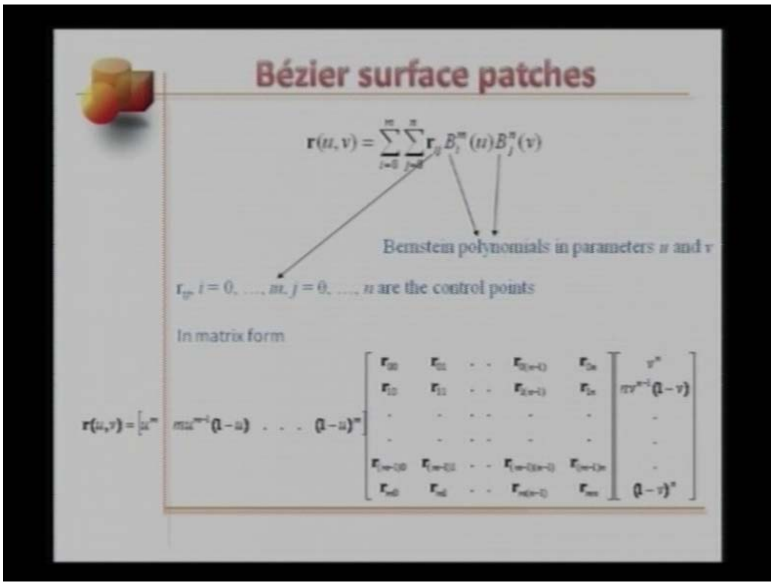


Let us go back and try to investigate these 2 matrixes M 16 is the inverse of 1 these matrixes and if you look at this matrix here, this is nothing but the transpose of this matrices. One reason why we use the transpose of M 16 nevertheless with this sampling information available to us, we can identify the unknown coefficient matrix D ij and once we have that, we have the mathematical expression for the 16 point form surface patch. The discussion about this type of tensor product surface is kind of academic. I am not quite for sure if the CAD industry uses this model often; rather, industry uses Bezier surface patches and Bernstein surface patches more commonly.

(Refer Slide Time: 17:39)



(Refer Slide Time: 18:20)



So, finally, the 16 point form patch looks like r as function of u and v equals u cube, u squared, u 1 row matrix times M 16 times the sampling information that we provide to determine the unknown D ij’s times M 16 transpose times the column vector v cube, v squared, v, and 1.

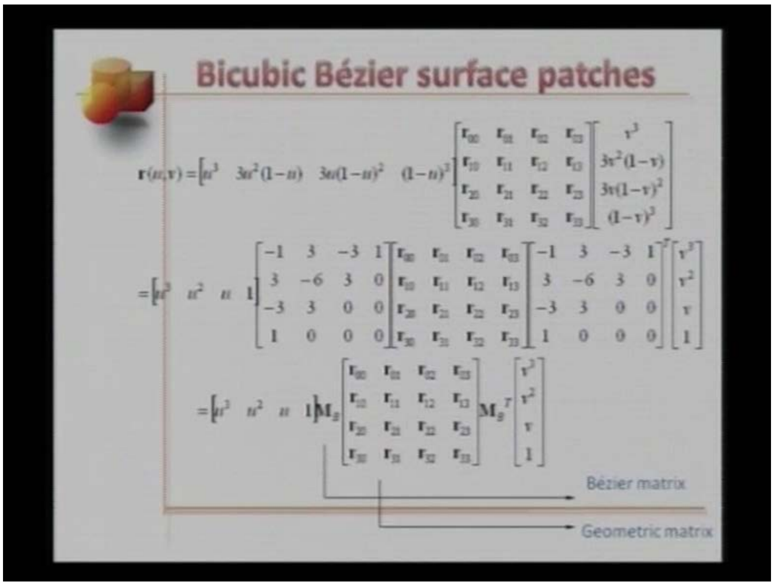
Let us move on now to more commonly used patch models; the first one of those is the Bezier surface patch; ten marks for guessing this; how would the mathematical form of a Bezier patch look like? So, think about it. It would be using Bernstein polynomials, Bernstein polynomials, as function of u and v independently. So, the patch as a function of u and v will be given by summation index i going from 0 to m, index j going from 0 to n, is at 2 different degrees along u and v respectively times r sub ij.

This is where the designer comes in picture. He would specify the x, y, and z coordinates of each designed points r sub i j times the Bernstein polynomial with index j and degree n as a function of v, and another set of Bernstein polynomials B with index i and degree m in parameter u. So, if the designer specifies a bunch of design points or control points, the patch is readily available. We do not need to go through computations that we saw in the previous patch model, the 16 points form surface patch.

As I said r sub ij where index i goes from 0 to m and j goes from 0 to n are the control points are the designed points v im and b jm are Bernstein polynomials in u and v. In compact matrix form r of u and v is given as the row vector with u raised to m, m times u raised to m minus 1 times 1 minus u and so on with the last term as 1 minus u raised to m times the geometric matrix the first row being r 00, r 01 until r 0 n minus 1 r 0 n, r 1 0, r 1 1. The last but one term in the second row will be r 1 n minus 1 and the last term will be r 1 end and so on until we consider the last but one row with entries r m minus 1 0, r m minus 1 1 r m minus 1 n minus 1 r m minus 1 n.

The entries in the last row will be r m 0 r m 1 the last but one entry in this row is r m n minus 1 and the eventual entry is r m n, and of course we will have the corresponding column vector where we place Bernstein polynomial in v. Now, could you guess what the properties of the Bezier surface patch would be like? While discussing Bezier segments, we had extensively studied the properties of both Bernstein polynomials as well as Bezier curves. If you think about it, all those properties will be inherited by the Bezier surface patch models.

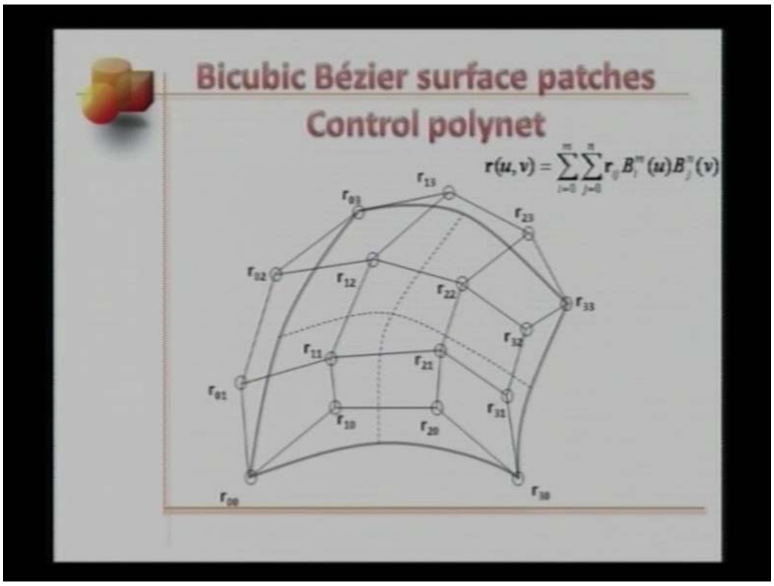
(Refer Slide Time: 24:28)



Let us take an example of a Bicubic Bezier surface patch. Here, you will have degrees 3 in both u and v. So, the mathematical expression is given by r of u and v equals the row vector u cube 3 u square times 1 minus u, 3 u times 1 minus u square 1 minus u cube. The geometric matrix that will be now given by 16 designed points arranged nicely along the parameter directions u and v and we will have Bernstein polynomials of degree 3 as a function of v arranged in a column form. Here, you can do a bit of algebra.

You can replace this row vector by u cube, u square, u and 1, times the Bezier coefficient matrix if you remember with entries minus 1, 3, minus 3, 1, 3, minus six, 3, 0, minus 3, 3, 0, 0 and in the last row 1, 0, 0, 0, times the geometric matrix times the transpose of this Bezier coefficient matrix times the column vector v cube v square v and 1. This is equal to u cube u square u 1 arranged row wise arranged m sub b times the geometric matrix m times sub b transpose times v cube v square v and 1 in column matrix. M b as I mentioned earlier is this 4 by 4 matrix here.

(Refer Slide Time: 27:49)



Bicubic Bezier surface patches control polynet or control net: This is how your polynet on net will look like. So, the user will be specifying 16 data points as the corresponding bicubic Bezier patch will be observed like this with these four as bounding curves. This could be boring, but let me go through the nomenclature of these design points again. Because as a designer, the order in which these design points are specified can be important. So, the first point is r 00, r 01, r 02, r 03. So, this curve here corresponds to the u value equals 0 and the v value progressing from 0 to 1. This point here is r 10, r 11, r 12, and r 13. This is r 20, r 21, r 22, r 23 and finally, r 30, r 31, r 32 and r 33.

Let us now pop up the mathematical expression for the Bezier surface patch model. Here, m equals 3 and n equals 3, and whatever we have learnt from our previous discussion on Bezier segments let us try to use them to understand how we can predict the shape of a Bezier surface patch via the shape of the corresponding control polynet.

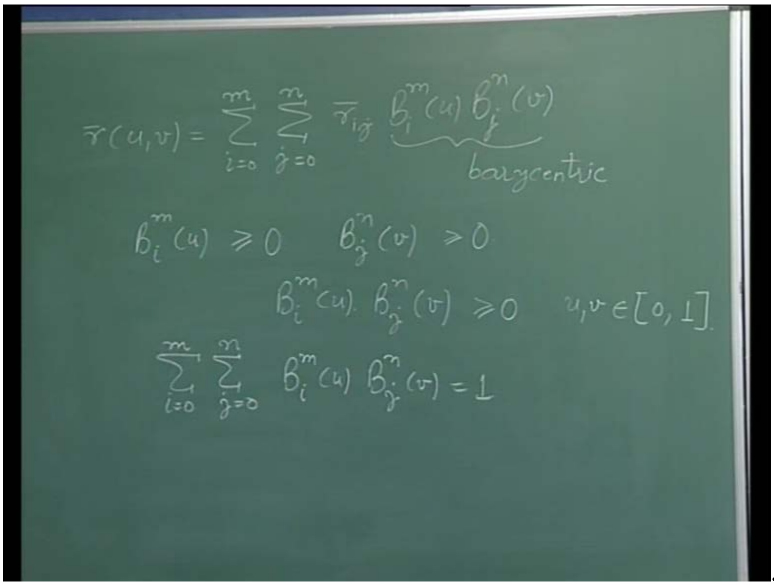
Take a look at this control polyline r 00, r 10, r 20, r 30. This will correspond to the parameter value v equals 0. So, these are the four bounding curves. This curve here is for v equals 0. Now what do you have to say about the shape of this curve? Well, think about the partition of unity property of Bernstein polynomial that gives rise to the convicts called attribute. What do I mean? Well, this curve is going to be lined within the convicts all given by these four design points. Also, the shape of this curve will be loosely predicted by the shape of this control polynet.

Let us take look at another bounding curve; this one for example, this curve corresponds to the value of v equals 1. We have the control polyline given by r 03, r 13, r 23, and r 33. Once again this curve will lie within the convex hall given by these four design points and the shape of this bounding curve will be predicted not strongly, but loosely by the shape of this poly line.

What do you have to say about the nature? Let me be a little precise about the mathematical nature of these 2 curves and this corresponds to v equals 0; this corresponds to v equals 1. Would you agree with me that these are 2 cubic Bezier segments? Think about it. Likewise, how about these 2 bounding curves? This one here corresponds to u equals 0 and for different values of v in between 0 and 1. Correspondingly, we have this control polyline.

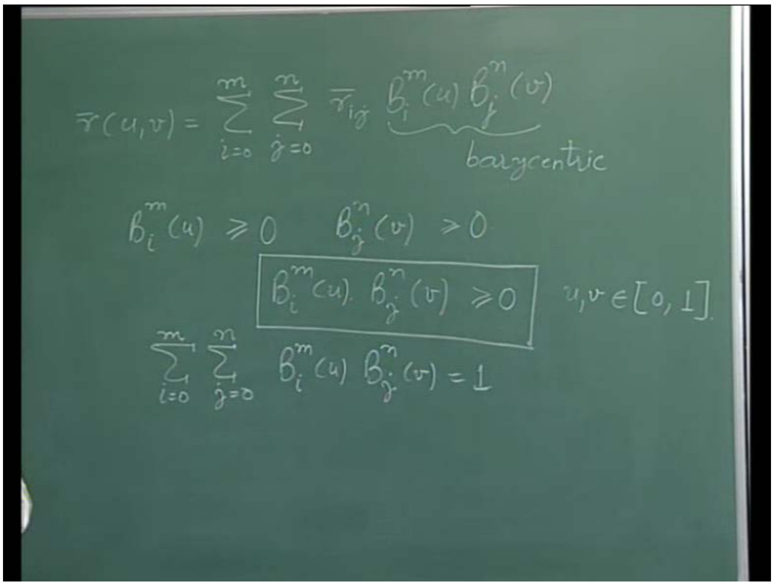
Because of the partition of unity property or the convexual property, this curve will lie within convex hall of these four points and the shape will we predicted by the shape of this poly line and the same is true for this bounding curve here, for u equals 1 and v ranging from 0 to 1; in fact, all of these four bounding curves of Bezier segments each of degree 3. How about the shape of the surface patch overall? Let us look at this expression here. So, the product b sub i m of u and b sub j n of v acts as the weight corresponding to the design point r i j. Would you think these weights will also be barycentric in nature?

(Refer Slide Time: 35:50)



So, this is the expression for the tensor product surface involving Bernstein polynomials r of u v is equal to summation i going to from 0 to m summation j going from 0 to n; the design points r i j times the Bernstein polymonial with index i and degree m in u and the Bernstein polymonial with index j and degree n in b. we need to show whether this product is barycentric on another. So, first realise that B i m u is equal or greater than 0 and also B j n v is equal or greater than 0, for values of u in between 0 and 1 and for value of v in between 0 and 1. Clearly, their product B i m u times B j n v will be greater than equal to 0. This is for u and v both belonging to the interval 0 1. Next, to show that this product is barycentric, we need to show that summation i going from 0 to m summation j going from 0 to n B i m u B j n v equals 1.

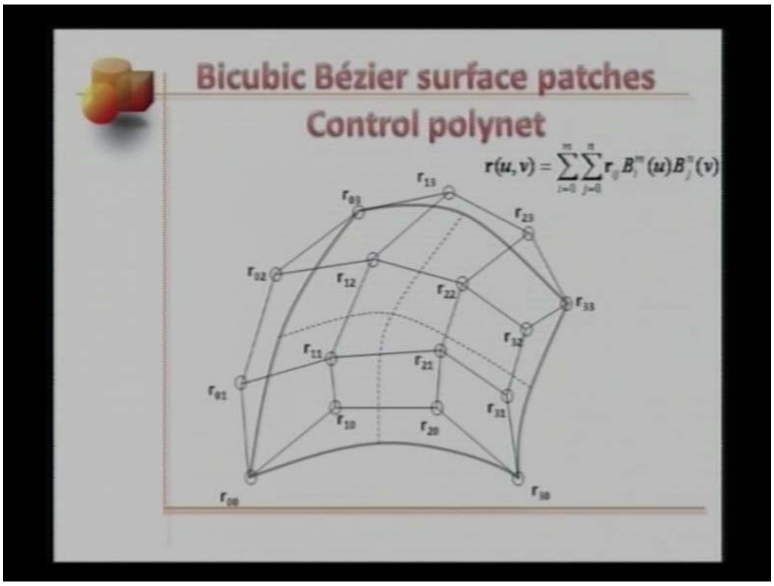
(Refer Slide Time: 38:13)



We show this now. So, we have 1 minus u plus u raised to m equals 1. Also, we have 1 minus v plus v raised to n equals 1. We multiply both these equations together to get 1 minus u plus u raised to m times 1 minus v plus v raised to n equals 1. From our discussion on Bezier segments, we know what this binomial expansion is. So, for the first case we have B 0 m u plus B 1 m u up to B m m u times B 0 n v plus B 1 n v up to B n n v equals 1. You would have realised that the Bernstein polynomials form individual terms in this binomial expansion and B j n of v all of this terms over here are individual terms appearing from this binomial expansion.

We can write these equations in short form: Summation i going from 0 to m B i m u times summations j going from 0 to n B j n v equals 1. I can bring this summation here and I can get summation i going from 0 to m summation j going from 0 to n B i m u B j n v equals 1. Let us block this is result and let us block this result here. These results when combined together make the product of 2 Bernstein polynomials barycentric since the bicubic Bezier patch or any Bezier patch for that matter is derived from the corresponding Bezier curve models by use of Bernstein polynomials.

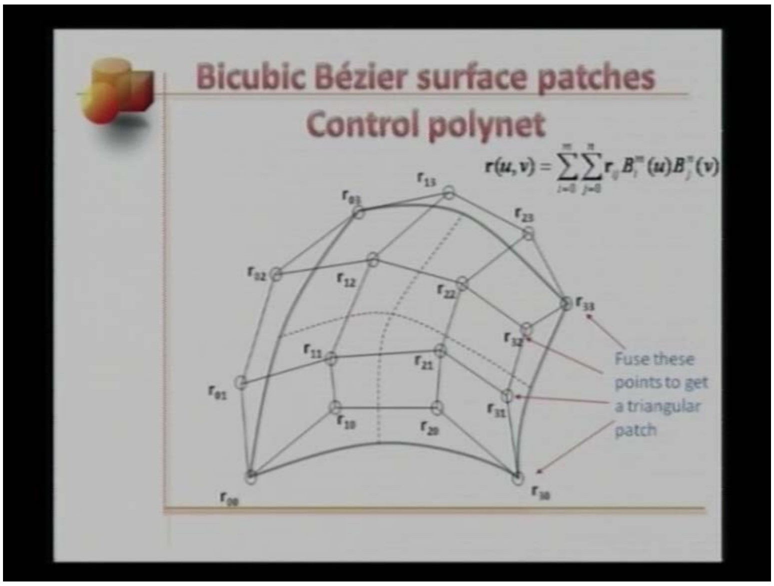
.(Refer Slide Time: 27:49)



All the associated properties get inherited. From the design view point, that means that this patch will be lying within the convex hall defined by this control net; also the shape, the overall shape of this patch will be controlled loosely by the shape of the net; in a sense, if I move for example, r 12 to a different location, the corresponding portion on the patch will get changed in shape accordingly, however the shape change will be global.

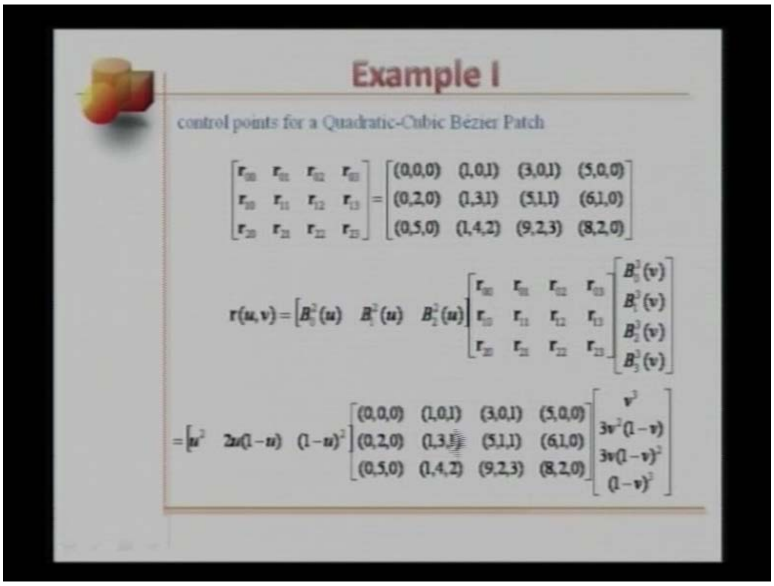
That is the change which will possess in the entire patch as opposed to global. As you know Bernstein polynomials are barycentric; that is not local property as you had seen in place of Bezier function; rather, if we use these lines in Bezier function over here, in place of Bernstein polynomials, and will be studying these models later. The corresponding shape changes will be global.

(Refer Slide Time: 44:16)



Another example we can say fuse these points together to get a triangular patch. Why because since this bounding curve lies within the convex hall of this polyline, using these points together, we will make this curve become a point. And as a consequence, we will not have four bounding segments, but rather 3 of them.

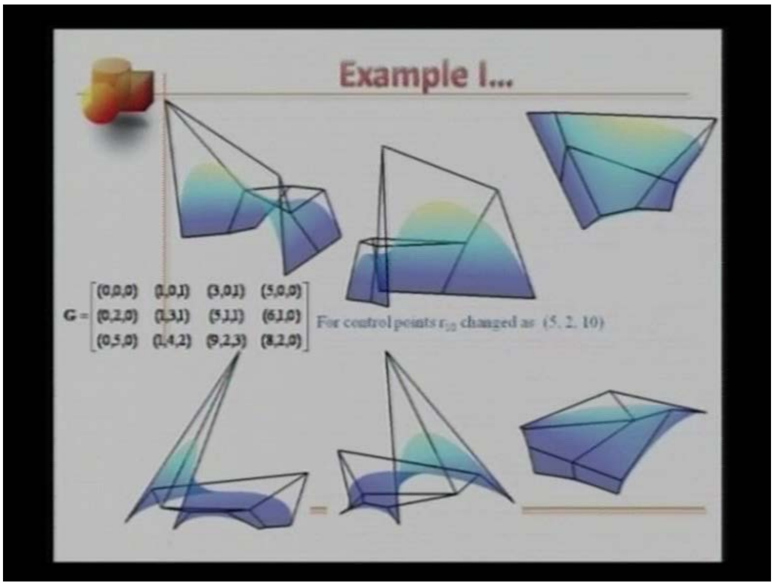
(Refer Slide Time: 45:03)



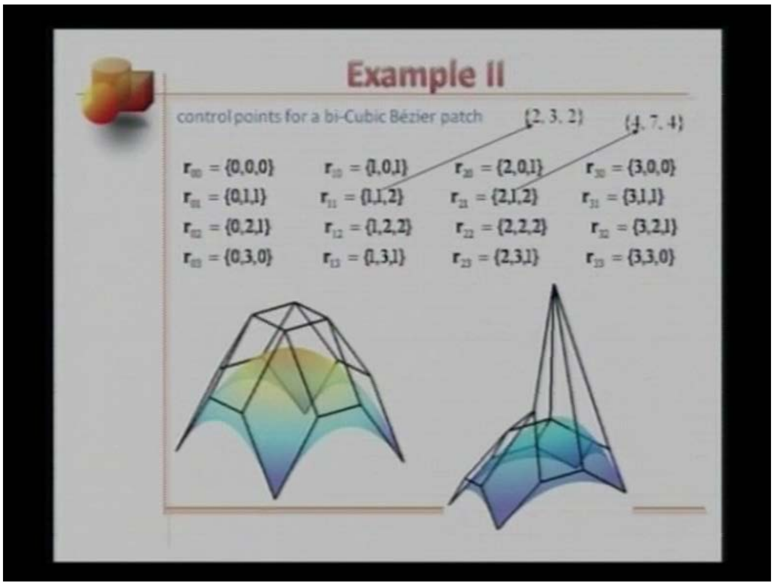
Let us take a look at a few examples: First one, the control points for a quadratic cubic Bezier patch are given by 0 0 0, 1 0 1, 3 0 1, 5 0 0, 0 2 0, 1 3 1, 5 1 1, 6 1 0, 0 5 0, 1 4 2, 9 2 3, 8 2 0.

Here, m equals 2 and n equals 3. Corresponding to u, we have Bernstein polynomial of degree 2 and corresponding to parameter v, will have the same in degree 3. We can do the map and get the x y and z coordinates in terms of u and v. To get the x coordinates we use the first set of values; for the y coordinates we use the second set of values and to get zee scale (( )), we use third set of these values.

(Refer Slide Time: 46:21)



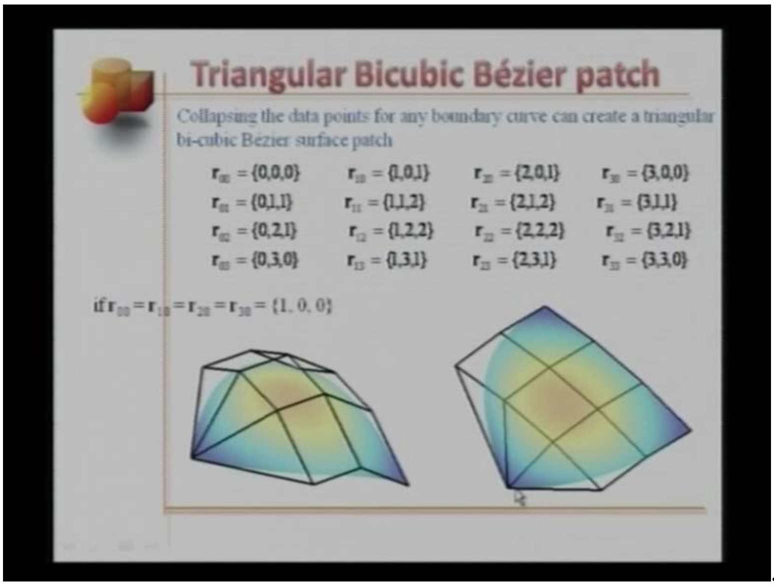
(Refer Slide Time: 47:36)



Let us try to figure how the surface patch looks like. For this geometric matrix, this is how the patch appears and this is the control polynet block. The patch is shown with 3 different view directions. This is the top view. If you recall our discussions on transformations from here to here to here, we are essentially performing a few of them. If you change the control point r 10, 5 2 10 a different surface patch will result; that is as shown here again via 3 different views.

The second example: Now, this is for a bicubic Bezier patch. The control points are given as 0 0 0, 1 0 1 - this is r 1 0; 2 0 1 r 2 0, 3 0 0, 0 1 1, 1 1 2, 2 1 2, 3 1 1 this is for r 3, 1 0 2, 1 1 2, , 2 2 2 and 3 2 1, and finally, r 0 3 is 0 3 0, r 1 3 is 1 3 1, r 2 3 is 2 3 1, r 3 3 is 3 3 0. With this geometric information, the patch looks something like this.

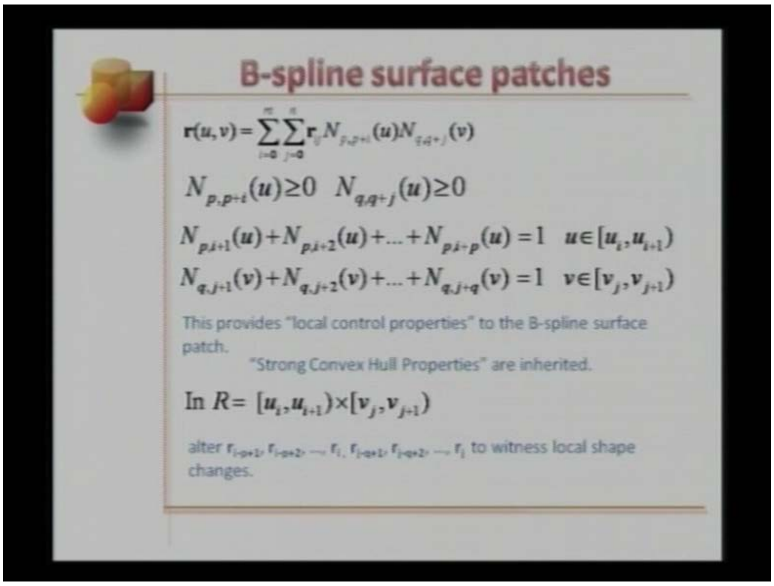
(Refer Slide Time: 49:17)



.

The contol ploynet or control net is shown using lines in black. If we change r 1 1 to let us say 2 3 2 and if we also change r 21 to let us say 4 7 4, we will witness a shape change in the patch and that change, as I mentioned earlier, will be global. Or to get a triangular bicubic Bezier patch, we will have to collapse the data points or the design points for any bounding or boundary curve to create a triangular patch. So, this is the information that we had from the previous example, that if r 0 0, r 1 0, r 2 0 and r 3 0 are all collapsed to a point 1 0 0, this is the triangular patch that we will get. These are 2 different views and you can clearly observe that we have now 3 as opposed to four bounding curves.

(Refer Slide Time: 50:16)



Finally, we come to B-spline surface patches. The mathematical expression is given by r of u and v is equal to summation i going from 0 to m, j going from 0 to n of r i j times N p , p plus i as a function of u times n q q plus j as function of v. What have we done here? We have simply replaced the Bernstein polynomials in Bezier surface patch with the corresponding B-spline basis functions. Corresponding to the indeces i N p N p plus i are basis functions of order p and corresponding to indeces j n q q plus j are basis functions of order q. In general, p may not be equal to q. The 2 orders along the parametric directions u and v can be different. Like in case of Bezier surface patches, we can also argue that this product is now locally barycentric; I emphasize, locally Barycentric.

What do I mean? Each of these individual terms N p p plus i of u is equal to or greater than 0; likewise n q q plus j of u is equal to or greater than 0; correspondingly this product will either be 0 or past 0. Further from our discussion on B-spline basis functions and segments, we know that p of these order p basis splines sum to 1 that is N p i plus 1 of u plus N p i plus 2 of u upto N p i plus p of u is equal to 1 for values of u belonging to the interval u i u i plus 1. These are the knots in the parametric value u. Likewise, few of these order q basis spile functions will also sum to 1; that is N q j plus 1 of v plus N q j plus 2 of v upto N q j plus q of v is equal to 1 for values of v in between v j and v j plus 1. Again these are the knots in the parametric value v.

Like we showed in case of Bernstein polynomials, we can also observe that the product of these B- spline basis functions will sum to 1 in some region and this is the property that provides local shape control to B-spline surface patch. If you understand our discussion on B-spline basis functions, you will not have much difficulty comprehending this property. You would also observe that strong convex hall properties are inherited by B-spline surface patches.

For some region r which is a product of these 2 intervals u i u i plus 1 v j v j plus 1 all we need to do is we need to alter or relocate the control points r i minus p plus 1 r i minus p plus 2 up to r i or r j minus q plus 1 r j minus q plus 2 up to r j to observe or witness local shape changes. Let this be an exercise for you to figure how many knots will be needed corresponding to parameter u and those corresponding to parameter v. Although I will not discuss B-spline surface patches further, you are free to explore these models in detail all by yourself.

.