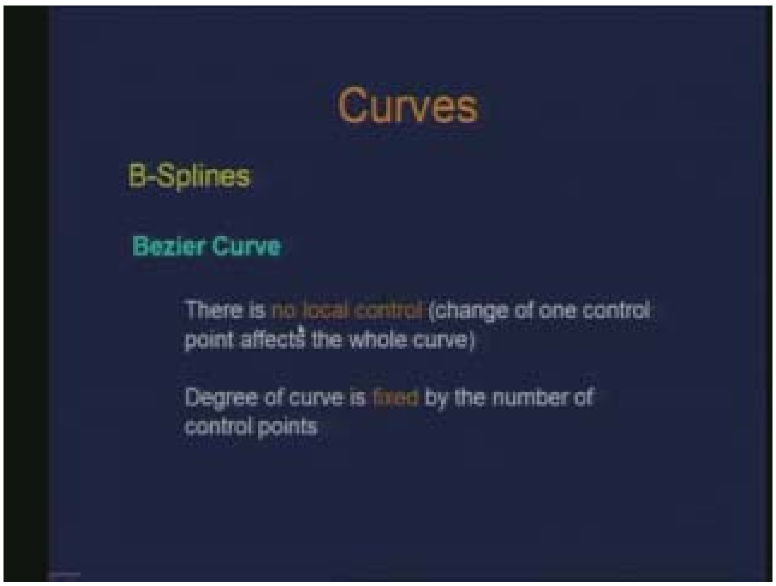
Introduction to Computer Graphics
Dr. Prem Kalra
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi
Lecture # 14
Curves (Contd….)

We have been talking about parametric curves. We observed certain limitations in Bezier curves. What we basically observed is that these Bezier curves have somewhat limited local control.

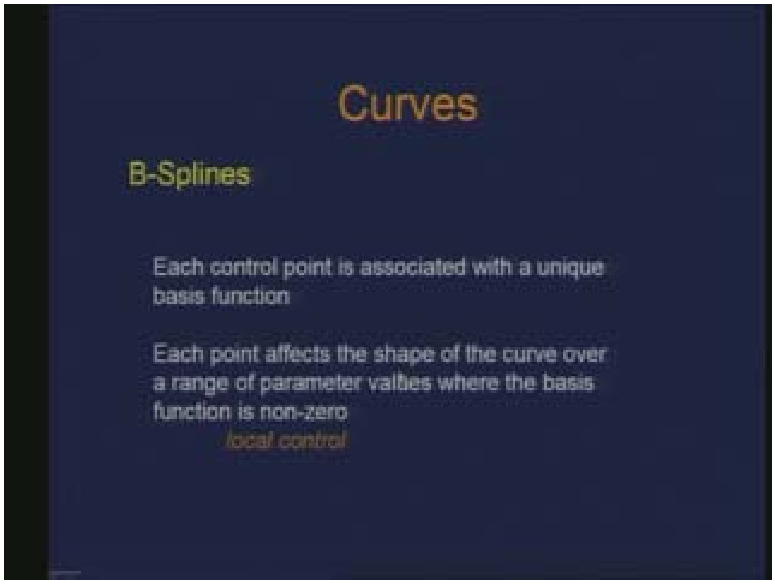
(Refer Slide Time: 01:21)



And with local control we meant that if I change one of the control points of the control polygon then the entire curve changes. So I have to compute the entire curve though there is a notion of pseudo local control where you observe a degree of influence of a particular control point on the curve. However, as far as the computation is concerned the entire curve needs to be recomputed because there are even small changes beyond the influence of that point to the curve. So there is a limited or no local control in Bezier curves. So that in fact many a times poses a problem to the user. You may design a gross shape of the curve and then you suddenly want a dip in the curve. So you intend to change the position of the control point and what happens is the entire curve changes. So a local dent to the curve is not so much possible in Bezier construction.

The other thing which we observed is that the degree of the curve is actually determined by the number of control points we have in the Bezier polygon. So there is a direct relationship between them. In order to compensate some of the issues which are for the local control you may want to add control points but that directly increases the degree of the curve so which may not be always desirable. So in order to overcome some of these limitations we would like to see what B Splines offer. Some of these limitations are actually overcome in B Splines.

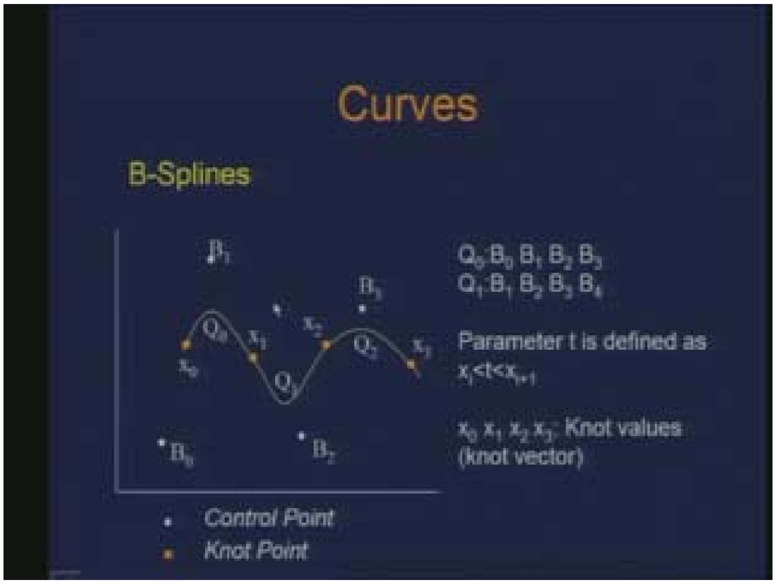
(Refer Slide Time: 03:56)



What makes the B Splines different from the Bezier curves is that first of all each control point is associated with a unique basis function. Then as a consequence each point affects the shape of the curve over a range of a parameter values. We have the basis function as non zero. So it actually well defines the influence of a control point and that offers the flexibility of having local control in the curve. So this is the broad way of defining these B Splines.

If we define a curve using control points like B_0 B_1 B_2 B_3 B_4 and so on then what happens in the case of B Splines is if we choose a particular degree of the curve we want to have then we observe that there are segments of the B Spline curve which are influenced by some number or certain number of control points.

(Refer Slide Time: 05:34)



So for instance there is this segment Q_0 which is basically influenced by these B_0 B_1 B_2 B_3. So in a way the construction of this is defined by these four control points. Similarly, Q_1 is through B_1 B_2 B_3 B_4 and so on. So it is basically some sort of a collection of these pieces of the curve which are the joints at some junction points. These are the junction points of these segments and the corresponding parametric value to these junction points which are x_0 x_1 x_2 and so on. This is what we call as the knot values and the main part of the parameter value using these x_0 x_1 x_2 x_3 is the knot vector. Therefore if you move the point B_1 it is going to affect only certain segments of the entire B Spline curve which in this case namely be Q_0 and Q_1.

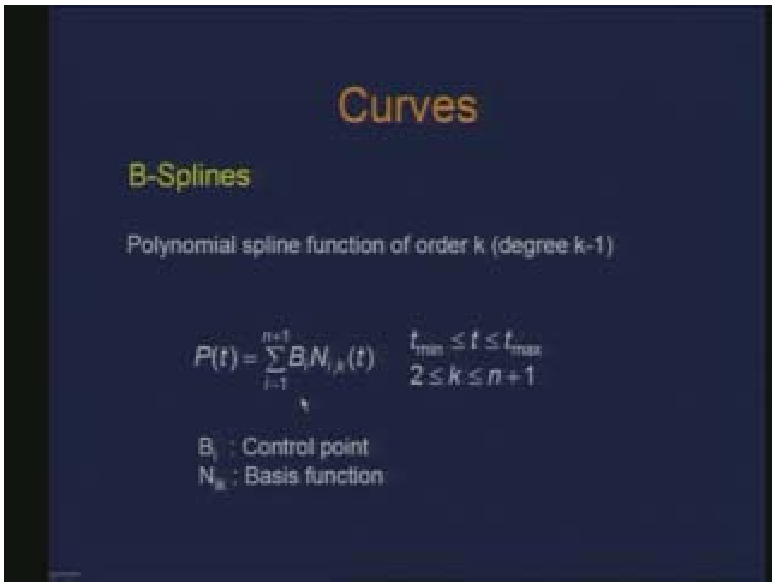
Immediately we see that there is some sort of a locality of the influence of the control point just the way we define these. If I just want to move the point B_1 because the subsequent segments are not going to have B_1 there. So this is just a broad sketch of the idea behind having these B Splines. So it is something what we also observed in the subdivision. We sub divided the Bezier curve so what was in turn happening is that segments were being formed and the junction point was considered as the point on the curve were the subdivision was taking place. Then the left part of the junction point could be influenced by the left side of the control points. So the spirit is similar to what we observed in subdivision.

Let us look at the definition of the basis functions which would incorporate such features or properties. So there are things like control points and there are also things like knot vector which play a role in the definition of these B Splines. So the way these B Splines are defined again the definition incorporates some sort of a blending of geometric properties and blending is done through basis functions or the blending functions.

Again the formulation or the way we define is very similar to what we have seen in the case of Bezier curve or even cubic splines. Here I am using the geometric information coming as the control points represented using B and these Nik's are the basis function. So the parameter t could have the range defined between some minimum value of t to maximum value of t.

In fact that is what accounts for the knot values or the knot vectors which are coming because those x values are nothing but the domain parameter and that is what gets incorporated through t. And this k actually captures the order or the degree of the curve. So we also wanted this flexibility to be able to decouple the degree of the curve or the order of the curve from the number of control points.

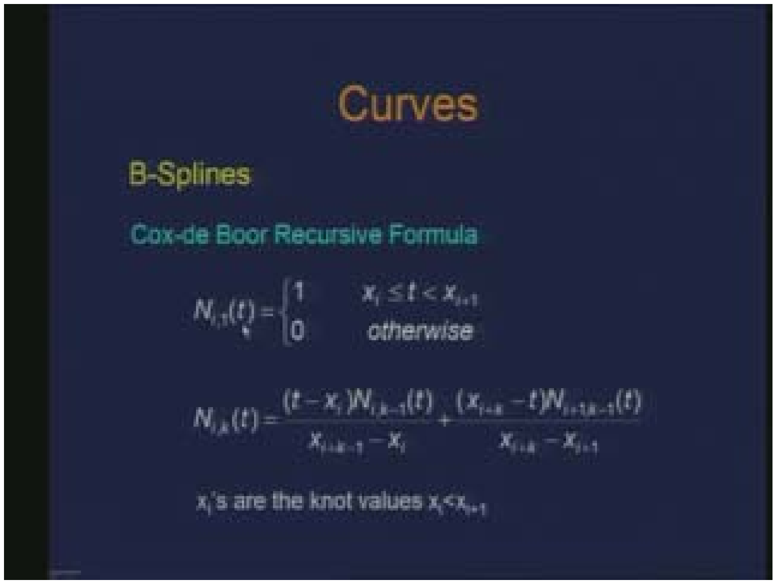
(Refer Slide Time: 09:08)



So this k is sort of an independent parameter to us which could facilitate that. So k is the order of the curve and ranges from 2 to n plus 1. So the entire range of the degrees of the curve which we can obtain is degree of 1 and degree of n which is coming from the number of control points we have. There is a small difference in terms of the way we give the index to them is from 1 to n plus 1. In the case of Bezier curve we used 0 to 1.This is just for some simplicity. This is how we basically define the B Splines.

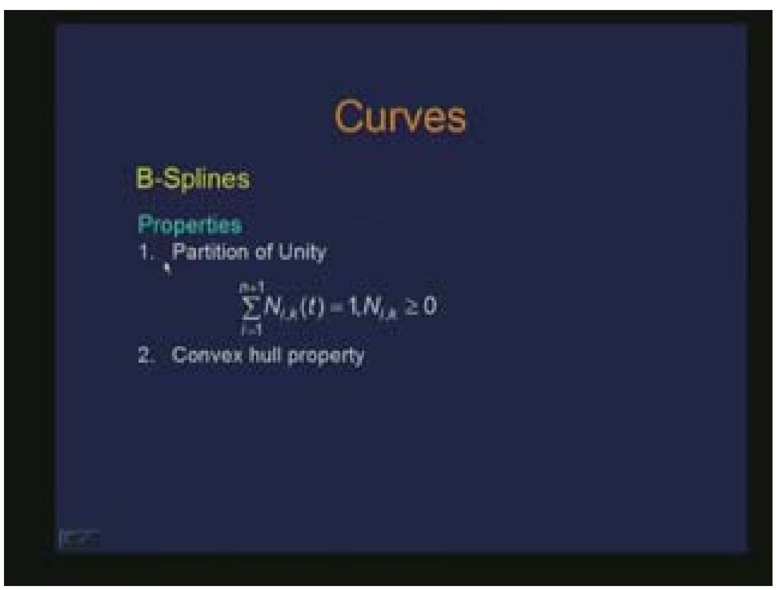
We will look at the anatomy of these $N_{i,k}$'s as they play a major role. Now these Nik's are basically defined through a recursive formula which was given by Cox de Boor. It is defined as a recursive formulation. These $N_{i,k}$'s are basically combinations of Nik minus 1 here Ni plus 1k minus 1. Therefore it is a combination of one order less Nik's which we combine here to get these Nik's and the base case is through when k is equal to 1. This is sort of a step function equal to 1 when the parameter t is within the knot values xi and xi plus 1 otherwise it is 0.

(Refer Slide Time: 12:18)



And as far these x_i's are concerned we observe that there is a monotonicity in the definition of these x_i's. So the way we define the knot values xi is less than or equal to x_i plus 1.

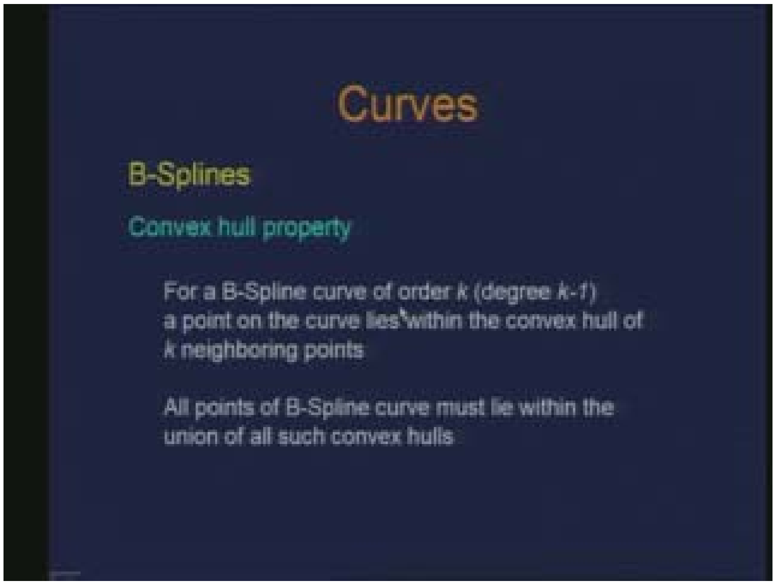
(Refer Slide Time: 13:54)



Now just to see some of the properties which are exhibited and which we desire for these B Splines to have is, one fact is that we assert this partition of unity so for the non zero values N_{ik}'s it sums to 1 and all these N_{ik}'s are non negative. And this can also then be used for having some convex hull property for the curve. In fact this convex hull property is even stronger in the case of B splines.

This basically says that for a B spline curve if you look at the curve of order k degree k minus 1 then a point on the curve lies within the convex hull of k neighboring points. Therefore if you are looking at for all points of the B spline curve then they must lie within the union of all such convex hulls found by these k neighboring points.

(Refer Slide Time: 14:49)



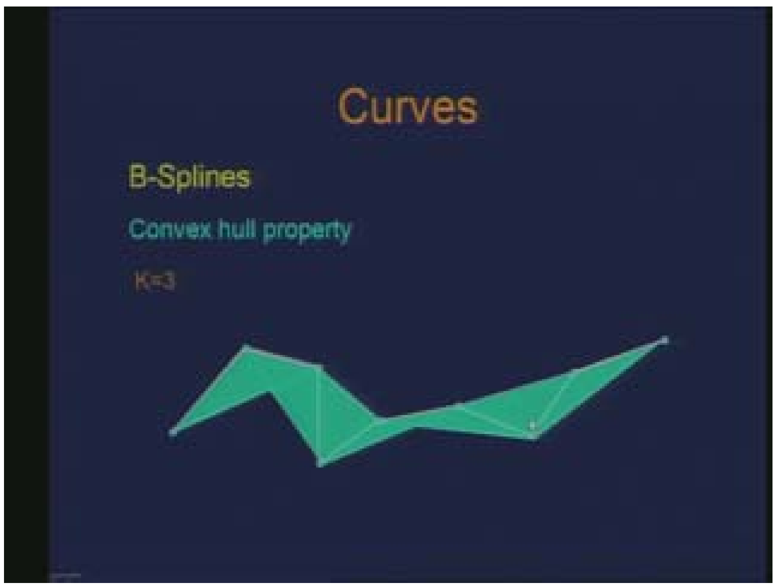
So I consider the simplest example where I say k is equal to 2.

(Refer Slide Time: 15:43)

And k is equal to 2 means the degree of the curve I am looking at is 1. And if these are the control points then the simplest B Spline curve which I am going to have is the line joining them and that is the curve of degree 1 or order 2 so k is equal to 2. Therefore if you want to see the convex hull property here is basically the curve is lying on the B Spline polygon itself.

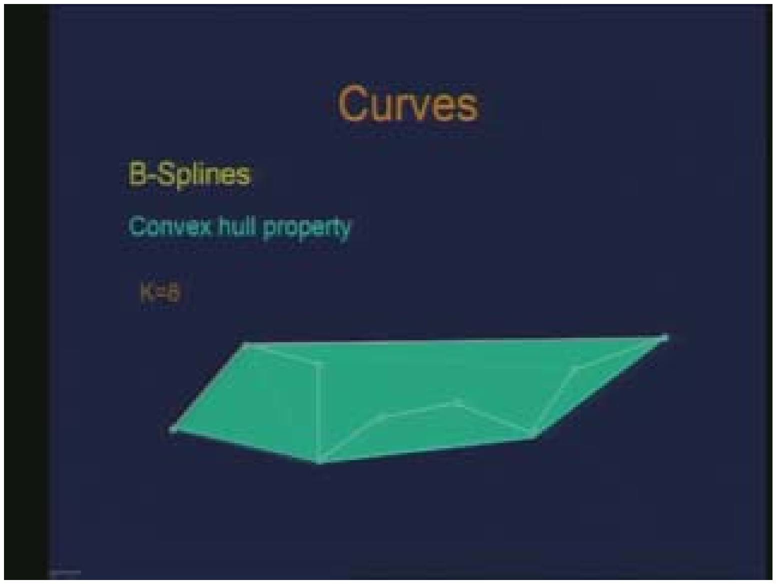
Now I increase the order consider k is equal to 3. So here I am looking at k neighboring three neighboring points so this is one set this is another set and so on. This is one set the first three points, this is another set and so on and then the entire convex hull is the union of them. Now what you observe is that the convex hull property is stronger compared to what we observed in the Bezier curves where it was considered as the convex hull of the entire Bezier polygon.

(Refer Slide Time: 17:46)



So here if I am having the curve of order 3 then I have a stricter definition of the convex hull which says that the curve is going to lie within this. Any curve defined of the order k is equal to 3 using these B Splines polygon will lie in this. Now, if I further increase k is equal to 8 I am talking about eight neighboring points so from here to here this defines 1 series of points and from here to here this defines the other. So now I have the entire convex hull which is the union of the two which happens to be the same as the convex hull of all the points.

(Refer Slide Time: 18:27)

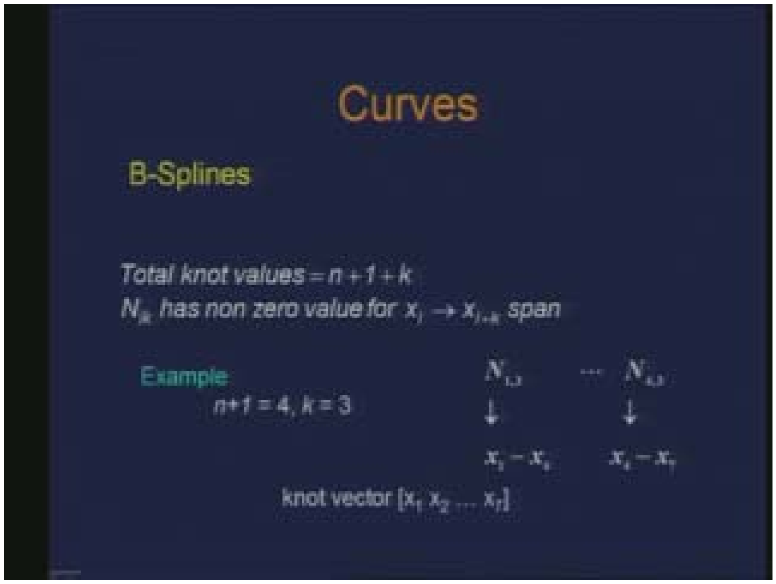


Only when the order goes high then I have to consider the convex hull of the entire polygon.

Now let us try to see more of the evaluation of these N_{ik}’s. We looked at the definition of these basis functions where this was the base case and this is the recursive formulation we have. What we observe through the definition of these basis functions is looking at the support of the knot values to the individual N_{ik}’s, If I look at the support of N_{ik}, by support I mean the range where it has non zero values then the support is from x_i to x_i plus k and that in fact will in turn determine the total number of knot values which is going to be n plus 1 plus k.

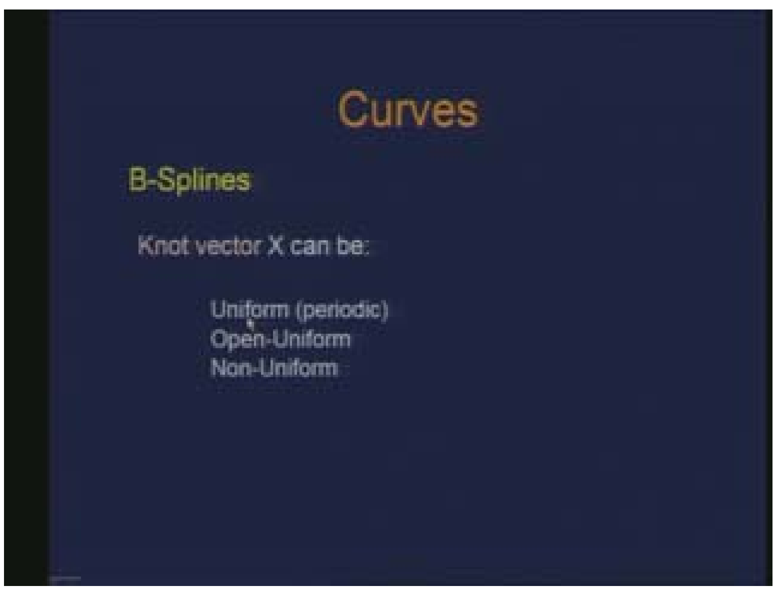
If I take an example here where n plus 1 is equal to 4 so I have four control points and k is equal to 3 the order is 3 so the basis functions which are going to be considered for the definition of the curve would be N_{i3}’s where the i is changing from 1 to 4 so I have $N_{1,3}$ to $N_{4,3}$. If I see the support of $N_{1,3}$ it is from x_1 to x_4 and if I see the support of $N_{4,3}$ it is from x_4 to x_7. So the knot values or the knot vector which I am considering will go from x_1 to x_7. So the total number of knots I have is 4 plus 3 which is 7.

(Refer Slide Time: 19:36)



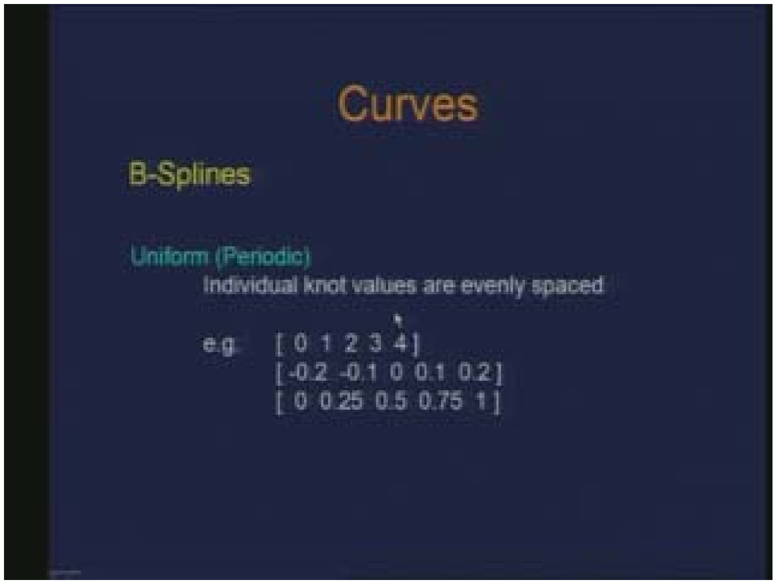
Now we see some sort of a role of these knot values or knot vector. So there are various ways in which we can consider these knot vector X.

(Refer Slide Time: 21:53)



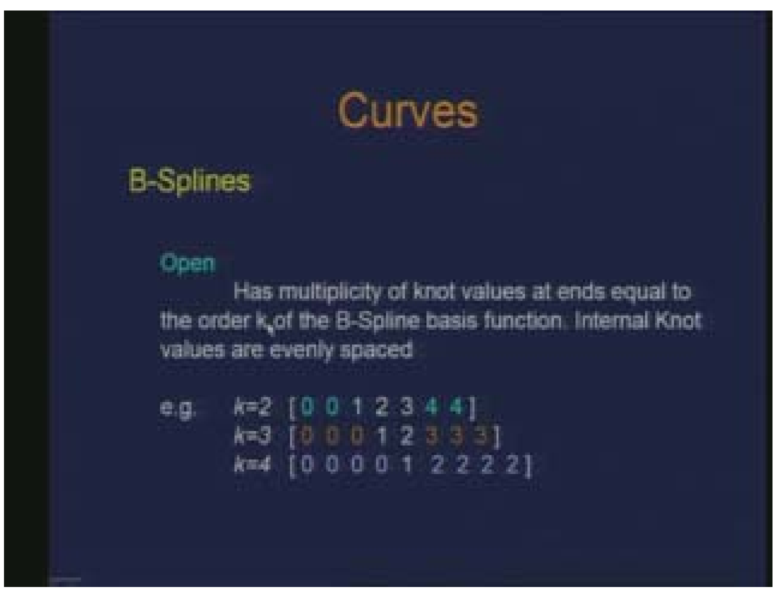
So they could be uniform or periodic knot values, they could be open or sometimes also called as open uniform or they could be non uniform. So let us see examples of each. By uniform or periodic knot values or knot vector the individual knot values are evenly spaced. For example, you have knot values defined as 0 1 2 3 4 or it could be minus 0.2 to minus 0.1, 0 and so on. So the difference between each pair of knot values is the same, they are evenly spaced.

(Refer Slide Time: 22:20)



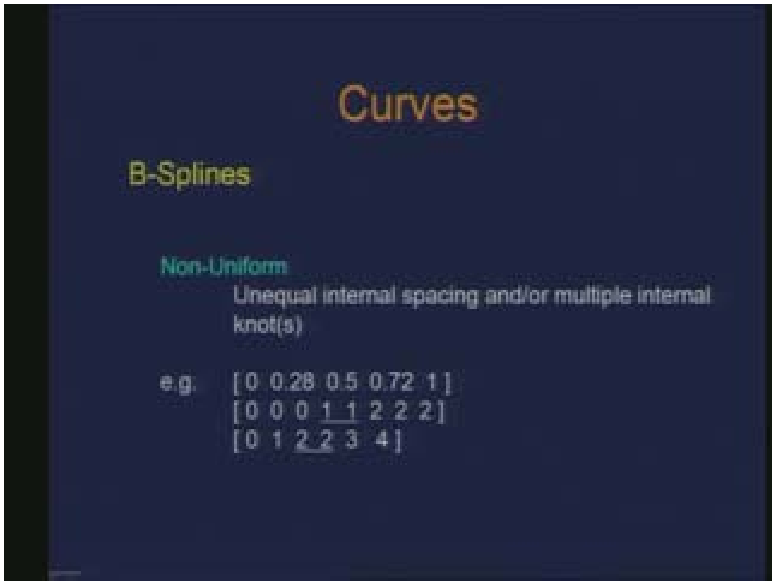
In the case of open knot values they have multiplicity of knots towards the two ends and this multiplicity is equal to the order k of the curve and the internal knot values are evenly spaced. For instance, if I have k is equal to 2 there is a repetition of the knots at this end and at this end so there is a multiplicity. Similarly, for k is equal to 3 I have this three times repeated and the last value three times repeated and the internal knots are equally spaced.

(Refer Slide Time: 23:01)



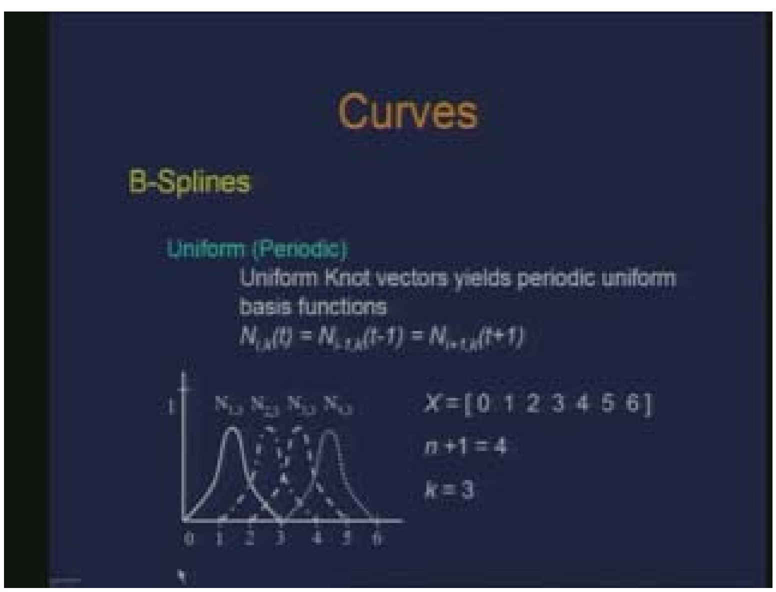
Non uniform as the name suggests is going to violate many of those conditions which we considered in the case of uniform and open. They are unequal internal spacings and or multiple internal knots. The difference between the knots is not the same and you may have a multiplicity even for the internal knots.

(Refer Slide Time: 24:23)



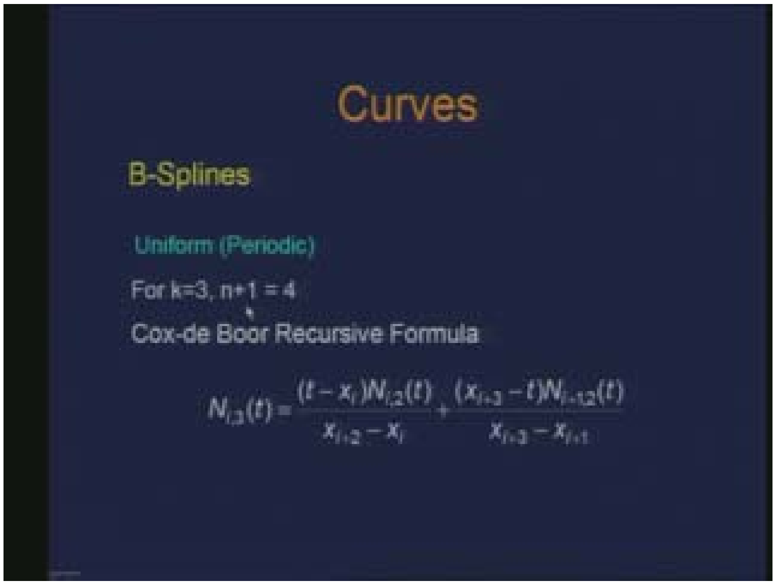
But all these types of knot vectors which we are seeing have the basic property of the monotonicity increase, they are all monotonically increasing. Let us learn about knot vectors and how they influence the construction. In the case of uniform knot vectors we observe that they yield periodic uniform basis functions with a translation and that is why they are also called periodic. Uniform from the point of view that they are equally spaced and periodic because they yield periodic Nik's with a shift.

(Refer Slide Time: 25:18)



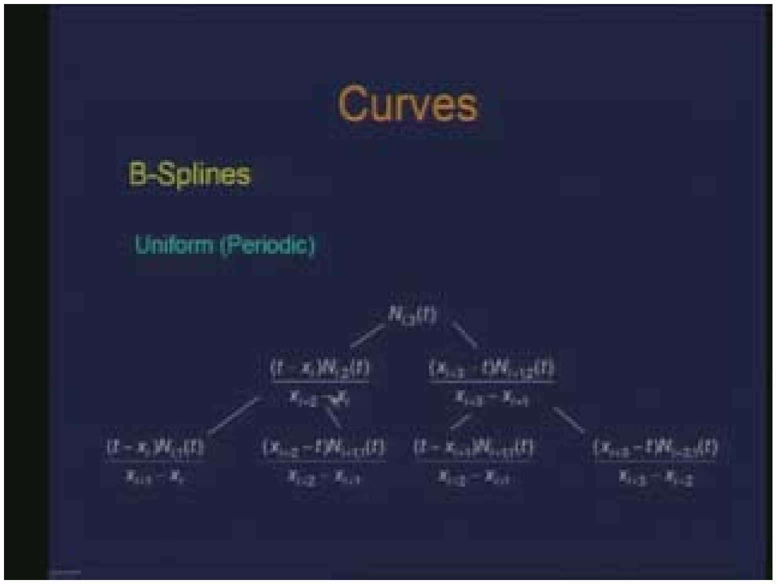
So, if I take an example where N plus 1 is 4 and the order equals to 3 and I assume that knot values I define using considering the positive integer values starting from 0 then I have these knot values as 0, 1, 2, till 6. Then the corresponding N_{ik}'s which I obtain using this knot vector they look like this. So what we observe is that this is $N_{1,3}$ $N_{2,3}$ $N_{3,3}$ and $N_{4,3}$ so they are in some sense periodic with some translation vector there. We have considered the example for k is equal to 3 and n plus 1 is equal to 4 four control points and this is what our recursive formula is.

(Refer Slide Time: 27:48)



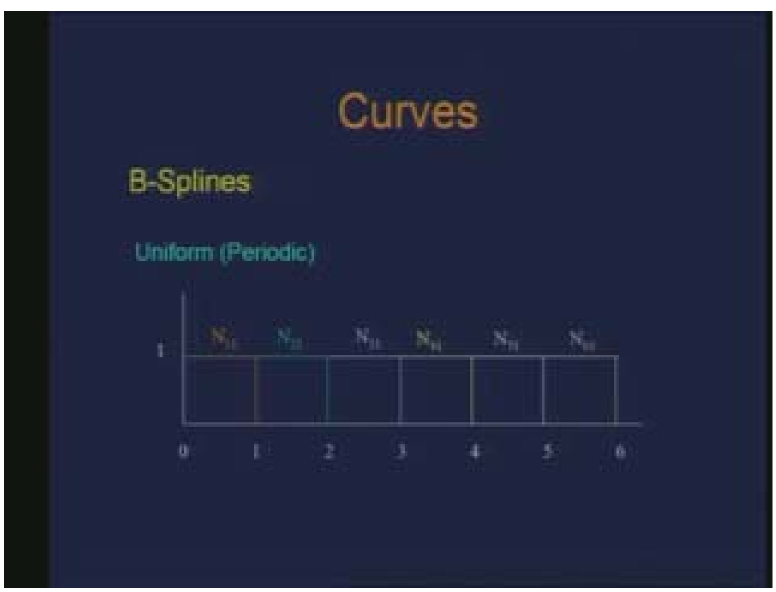
Now computationally when you see this is what is happening. It is a recursive formulation so you have the evaluation of $N_{i,3}$ coming as combination of $N_{i,2}$ here and N_i plus 1, 2 here, I am just expanding that formula nothing more.

(Refer Slide Time: 28:02)



And each one of these in turn will expand to this. So, at the base I have all the $N_{i,1}$ so this is $N_{i,1}$ this is N_i plus 1, 1 this is again N_i plus 1, 1 and this is N_i plus 2, 1 and that is what I can determine through the base case. And then they are going to percolate up for the evaluation of higher N_{ik}'s. So it is just a free formulation. Now if you want to see diagrammatically if you look at the base level of the recursive form you have all the $N_{i,1}$s, you have N_1, 1 N_2, 1 and all these are with respect to the examples we have considered. Therefore all these Ni1s are nothing but a step function of 1. So you have this as 1 and for the rest of the values of the parameter it is all 0.

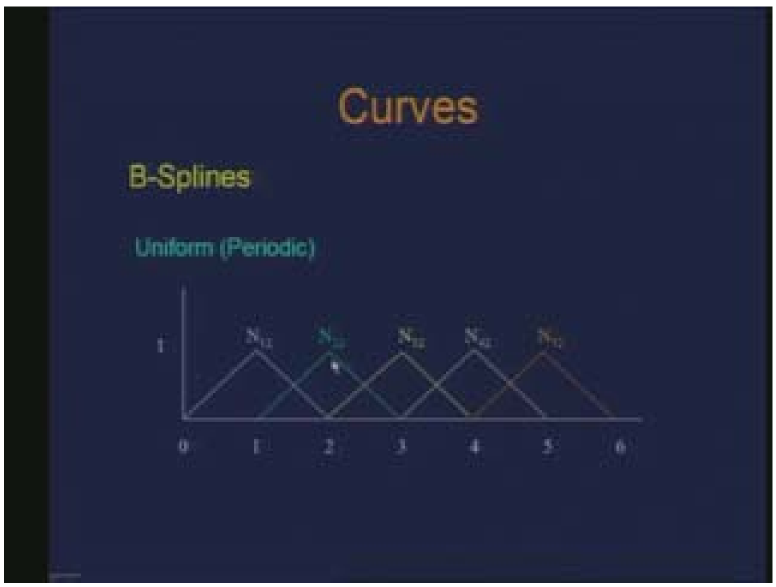
(Refer Slide Time: 29:10)



Similarly N_2, 1 has one value here and for the rest it is 0. Just by the basic definition of N_{ik}'s if you remember ……. (Refer Slide Time: 30:13). So $N_{i,1}$ is nothing but 1, 0 when t is between x_i and x_i plus 1 otherwise it is 0 so all you get is a step function. Therefore this is what you will get for the entire $N_{i,1}$s.

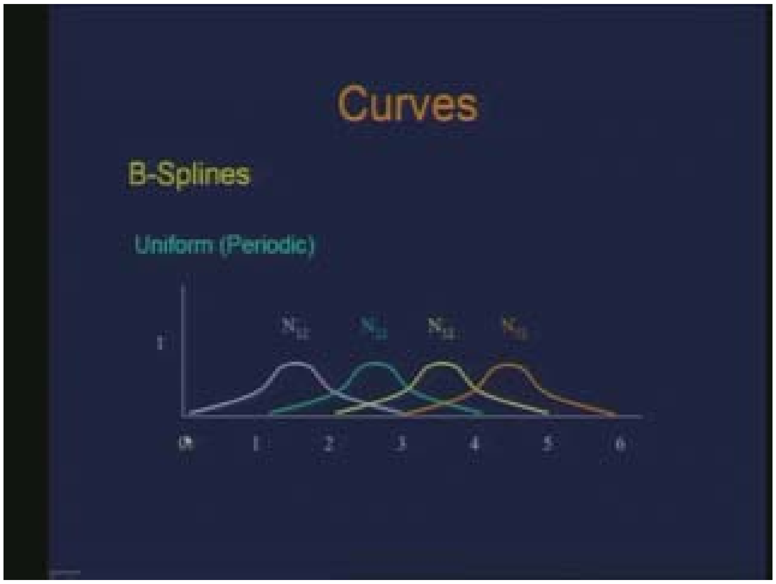
And now when we go a level up we are going to basically use the combinations of them and the span support of these N_{ik}'s is going to increase. So this is what will happen, for one level up you are going to have $N_{i,2}$ which now computes from x_1 to x_3 which is 0 and 2. Similarly it is N_2, 2. So these are triangular functions now which are nothing but linear combinations of the step functions which we had in the base level. And if you go one level up they are going to use the combinations of these which will be like a Gaussian. So it is a quadratic kind of a function like a Gaussian. So that is what we will have.

(Refer Slide Time: 31:13)



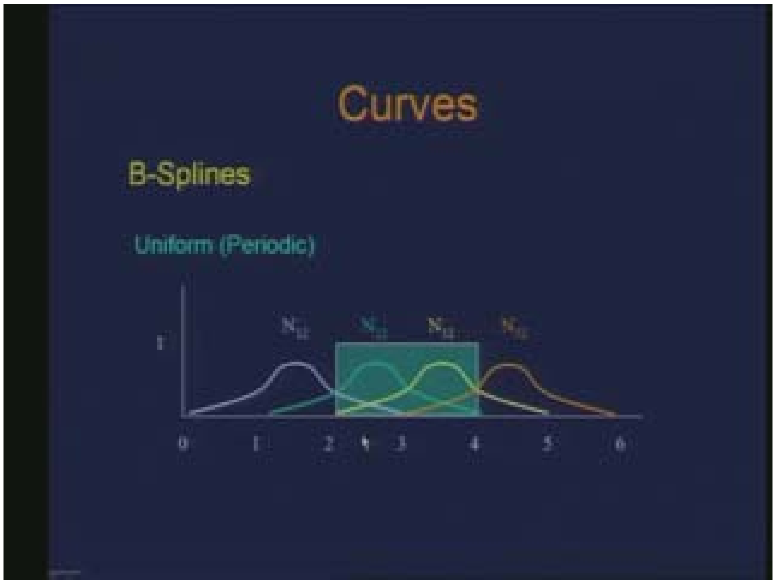
And again this span would increase. So, from here to here this is what you get evaluated as N_1, 2. And the final evaluation of the curve when we use the summation of N_{ik}'s the support of the N_{ik}'s is from x_i to x_i plus k and that is what we see here also.

(Refer Slide Time: 32:22)



Now there is another interesting thing. These are the knot values which we had considered of the type uniform or periodic. And remember that we have talked about the property that these N_{ik}'s for the parameter of interest which we want to design with the B Splines should sum this partition of unity to 1. Are they summing to 1 for the entire range? No, they are not summing to 1. So there is a range of the parameter in this span where it is summing to 1 and that is the range.

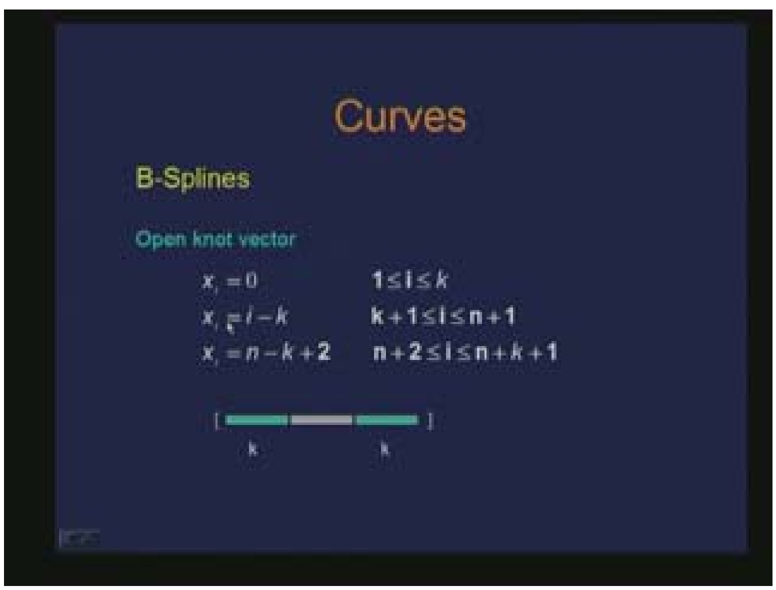
(Refer Slide Time: 34:25)



In this range they sum to 1, so that is the support of the parameter which we would like to use for constructing the B Splines because we want to assert certain properties like convex hull. So this is sort of a usable parameter range for the B Splines. So here the summation of N_{ik}'s is equal to 1. And in fact not been able to use all the definitions of N_{ik}'s we can overcome that through the open knot vector. Therefore the open knot vectors actually would enable you to have the use of entire range of the knots for the parameter. So, if you look at open knot vector, we say that there is a repetition or multiplicity of the knots at the ends. So this is repeated for k times, this is also repeated for k times where there is a k multiplicity of the value of the knot at the two ends which in fact can make us define the values of all the x_i's.

Let us say if I consider the first value as 0 so all these are defined as 0s so I am basically using the positive integer values for defining the knots starting at 0. So, if I have xi is equal to 0 then I can easily deduce that so this is going to happen from 1 to k where all these greens are 0s, from k plus 1 to n plus 1 I am going to have as i minus k and the last k values are nothing but n minus k plus 2. That is easy to figure out by definition that I am using k multiplicity of the knot. Now computing the N_{ik}'s using this knot vector would give you different types of a construction and then it actually enables you to have the use of the entire span of the knots.

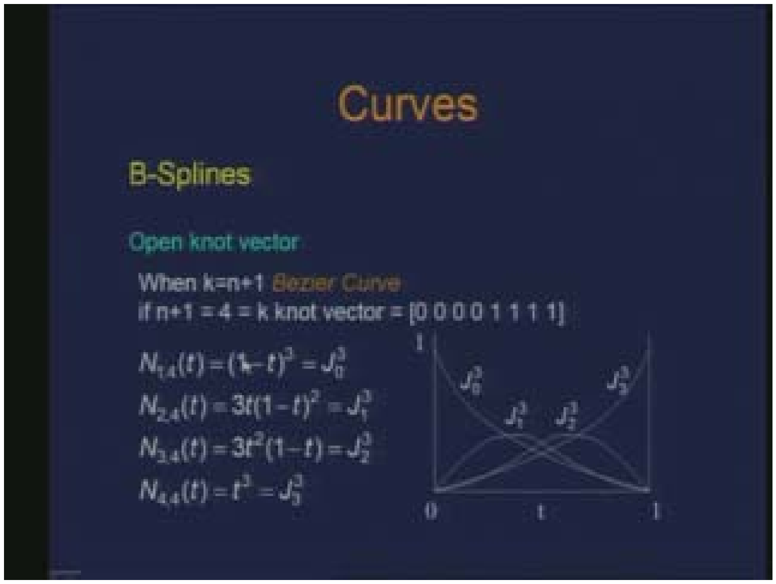
(Refer Slide Time: 38:13)



So if I have 0 1 2 3 and 0 is repeated on the left hand and 3 is repeated on the right hand then I am going to use the entire span from 0 to 3 unlike the case of the periodic knot values. And there is an interesting observation on the open knot vector that if I have k is equal to n plus 1 the order of the curve is the same as the number of control points I have then I obtain Bezier curve and that is sort of easy to verify.

If I again take a similar definition where I consider the knot values starting at 0 with integer increments of 1 then the knot vector I would have for n plus 1 is equal to 4 where I am talking about cubic B Spline will be this. All these 4 values here are 0s and all these 4 values are 1s which means that if I evaluate the basis function using this knot vector which will have the values for all the N_{ik}'s for the last basis the root of the recursion then it would be equal to the same as the Bernstein polynomials. So $N_{1,4}$ is 1 minus t power 3 which is nothing but J_3, 0. And similarly $N_{2,4}$ is 3t(1 minus t power 2) which is nothing but J_3, 1 and so on.

(Refer Slide Time: 40:10)



So, if you if you just do the evaluation of N_{ik}'s for k is equal to 4 then you observe that these polynomials are nothing but the Bernstein polynomials. Therefore the resulting curve what you get is the Bezier curve. If you look at the plot of these curves they are going to be like this. So here if you observe they all sum to 1 and the effective parameter range which I use is from 0 to 1 which is the entire range of the x values. Thus, Bezier curve is nothing but a special case of B Splines.

Basically when we look at the construction of B Splines there are certain control handles. What are these? These are the control points themselves, the degree of the curve which I want to construct k and the knots. Therefore each one of them is actually playing a role towards the construction of these curves. Therefore they also become control handles for changing or constructing a particular shape.