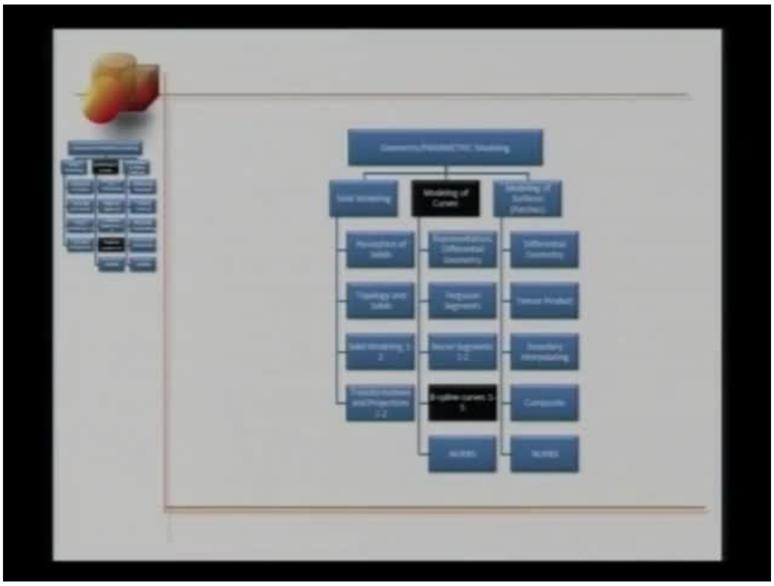
Computer Aided Engineering Design
Prof. Anupam Saxena
Department of Mechanical Engineering
Indian Institute of Technology, Kanpur

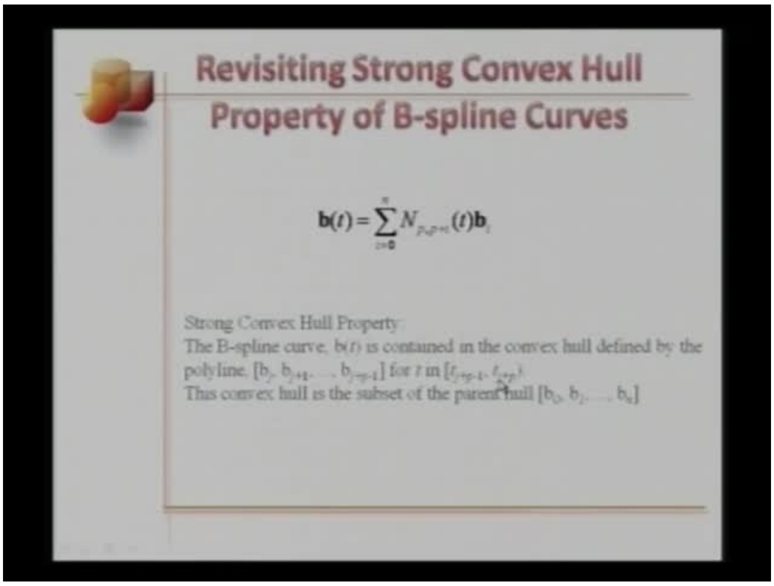
Lecture - 30

Good morning, we continue with our discussion on B spline segment and curves.

(Refer Slide Time: 00:22)

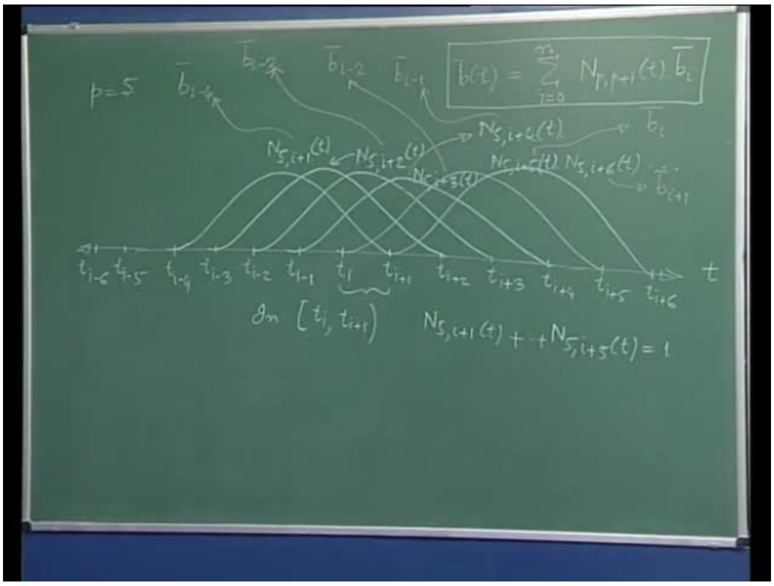


Refer Slide Time: 00:25)



Previously, we had discussed the strong convex hull property B spline curves to quite some extend. Let us revisit the property. So, this is our B spline segment b of t, which is defined that summation i going from 0 to n n p p plus i of t times b i. These are our B spline basis functions, and these are the designed points that q specified. It is strong convex hull property with straight, that the B spline curve b of t is contained in the convex hull defined by the polyline b j b j plus 1 up to b j plus p minus 1 for values of t in the intervals t j plus p minus 1 t j plus p. This convex hull is actually a sub set of the parent hull defined by all the sign points that use specify from b 0 to b n. Just in case, if you think you are getting last in these and this is, let me try to explain how this property works on board?

(Refer Slide Time: 01:50)



Let us try to work with order phi B spline basis functions. We have the definition of this curve as b of t, so vector equals summation, i going from 0 to n. These are n plus 1 b spline points n p p plus i of t times b of i. These are the resent point that you have specified. As I said you working with p equals 5. This having intermediate knot t i to the left t i minus 1 t i minus 2 t i minus 3 t i minus 4 t i minus 5 may be t i minus 6. To the right we have t i plus 1 t i plus 2 t i plus 3 t i plus 4 t i plus 5 t i plus 6, this is r parametric axis t.

Let us concentrate on this interval t i t i plus 1 the first B spline basis function of order 5 will be standing on 1, 2, 3, 4, 5 knot spans will be starting from t i minus 4. I have to be careful with the continuity conditions here and it will be ending at t i plus 1. This is n 5 equals 5 here, remember? i plus 1. The second one similarly, will be n 5 i plus 2. The third one, n 5 i plus 3. The fourth one, this is n 5 i plus 4, the fifth one, n 5 i plus 5. All these are functions of t. sixth one, n 5 i plus 6 of t and so on.

Now, remember over this interval t i t i plus 1 n 5 i plus 1 n 5 i plus 2, which this point here. n 5 i plus 3, which is this one, n 5 i plus 4 and n 5 i plus 5, all these five basis functions will be non zero here. Further n 5 i plus 1 of t, n 5 i plus 2 of t, n 5 i plus 3 of t, n 5 i plus 4 of t and n 5 i plus 5 of t, they will sum to one. So, in the sense in the interval t i t i plus 1 n 5 i plus 1 of t 2 n 5 i plus 5 of t, there will sum to one all the other B spline basis functions over this intervals will be 0.

Let us look at this definition of a curve, what is the design point associated with n 5 i plus 1 with n 5 i plus 5 b i is associated? So, let us work backwards, so the this 1 p i is associated this 1 here b i minus 1 is associated with in 5 i plus 3 b i minus 2 is affiliated with in 5 i plus 2 b i minus 3. With n 5 i plus 1 b i minus 4 are associated. Remember all these are vector design points for values of t within this intervals t i t i plus 1. These 5 B spline basis functions, they are locally valid centric you would know. In fact you know what I mean? They all are non negative and this sum to 1.

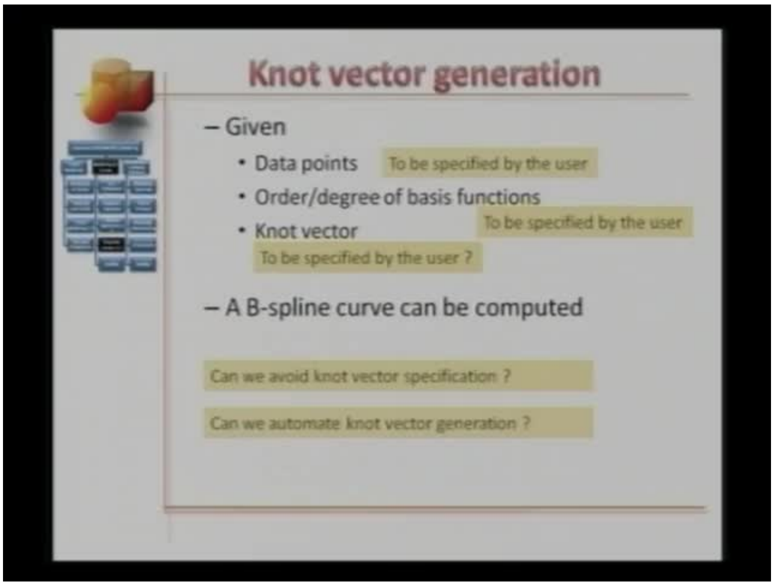
Because of weight of these 5 design points b i minus 4 b i minus 3 b i minus 2 b i minus 1 and b i from local are mix up. In a sense values of t in between t i and t i plus 1, a part of segment of b i will be lined, within the convex hull defined by this five points. This continuing further with n 5 i plus 6, we have the point b i plus 1 associated with it. This story for the interval t i plus 1 t i plus 2 is identical. Here will have n 5 i plus 2, n 5 i plus 3, n 5 i plus 4, n 5 i plus 5 and n 5 i plus 6, non zero and they would sum to 1. Because of which the subsequence of five points equals 5, the subsequence of five points which are b i minus 3, b i minus 2, b i minus 1, b i and b i plus 1, they will from a local convex hull.

In other words the values of t in between t i plus 1 and t i plus 2, these five points will form local convex hull. The segment b of t will have a part a weight within the convex hull defined by this five points for values of t in this interval. So, in case you get confused within this is always a good idea for you to sketch B spline basis function on board or on piece of paper and try to understand yourself, how local by simplicity works? In the previous lecture, we have seen something very similar.

Then there will be working with multiple knots, with such an exercise it become lot of easier for us to understand, how to design B spline curves with multiple knots and with multiple control points? For example, these five points co lease to become a single point, the convex hull would degenerate to a single point itself and force the curve to pass through that point. More specifically values of t equals t i and 5 at 5 i plus 5 is 0. As you have notice and n 5 i plus 1 n 5 i plus 2 n five i plus 3 n five i plus 4 and non zero, and these sum to 1.

Now, if i co lease for these points together. That means b i minus 4 equals b i minus 3 equal b i minus 2 which is equal to b i minus 1 for t equals t i the curve is bound to pass through these points. You can book out other properties in very similar manner. This is lecture number 30. Today we will concentrate specifically on knot vectors, knot vector generation.

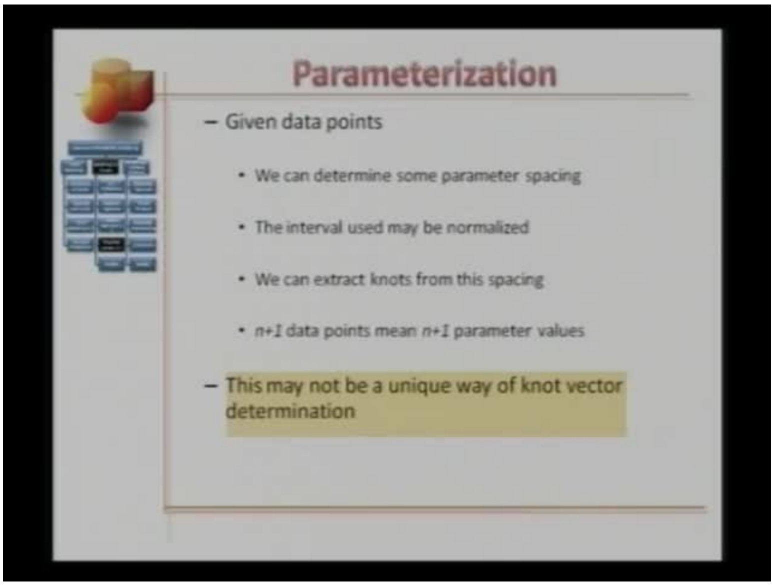
(Refer Slide Time: 12:31)



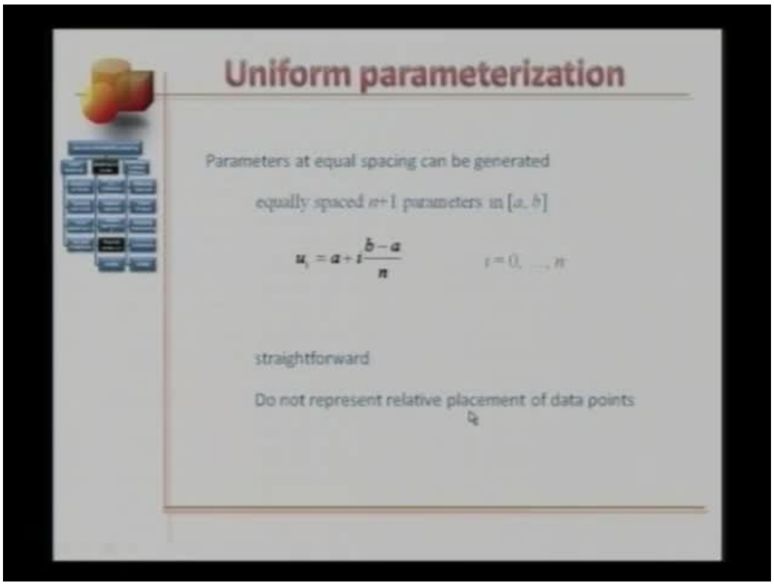
So, if you have given data points, degree basis functions and knot vector, a B spline curve can be computed. Now, data points there are usually specified by the user. So, is the case will be order the degree of basic functions and therefore, a B spline curve. How about the knot vector? Is to be specified by the user? May be or maybe not, we will talk about that in this lecture. One would notice here that we have too much of freedom, when designing B spline curve.

As a user I can freely specify and not practically, but it is possible us to avoid knot vector specification? We are specifying any ways too much of information all ready. If it is possible that we do not specify a knot vector, is it further possible for us to alternate knot vector generation? These are the questions that will be keeping in mind, when discussing concepts in to this lecture.

(Refer Slide Time: 14:04)



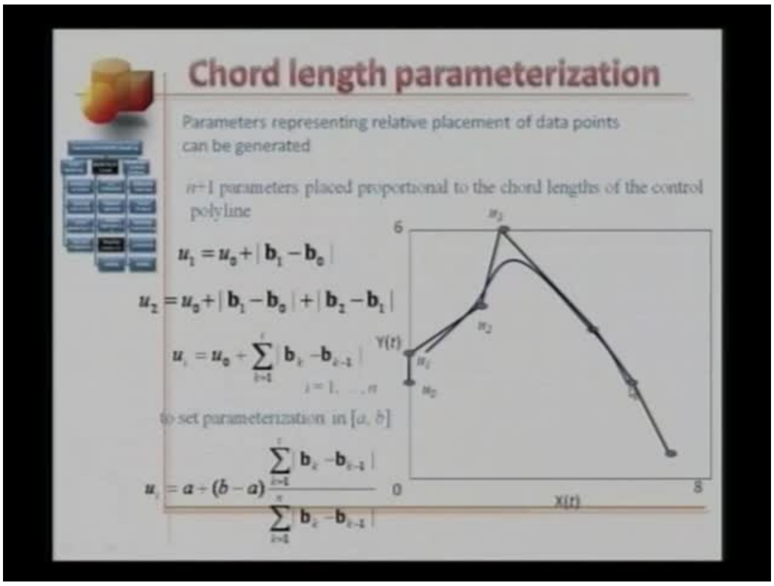
(Refer Slide Time: 15:03)



Let us talk about some parameterization schemes; given a few data points we can determine some parameter spacing. The interval used may be normalized. Maybe we can extract knots from this spacing, will be specified n plus 1 data points. These could be imply that might be able to extract n plus 1 parameter values. Of course, this may not be a unique way to determine a knot vector, there could be many other ways are doing that.

Let us focus on uniform parameterization now. By uniform it is essentially implied, that parameter at equal spacing can be generated. Let us say we have an interval a b and we would like to equally space n plus 1 parameter in this intervals. How we do this? Well say a parameter values are given by u sub i and they are equal to a plus i times b minus a over n. Here i is an index going from 0 to n. For i is equals 0, I have u of 0 equal a and for i equals n these two n would cancel out. This a would cancel out with this minus a and therefore, u of n will be equal to b. The other parameters are spaced by some length b minus a over n. This is straight forward, but this uniform parameterization scheme does not represent relative placement of data points. For that we have chord length parameterization.

(Refer Slide Time: 16:40)

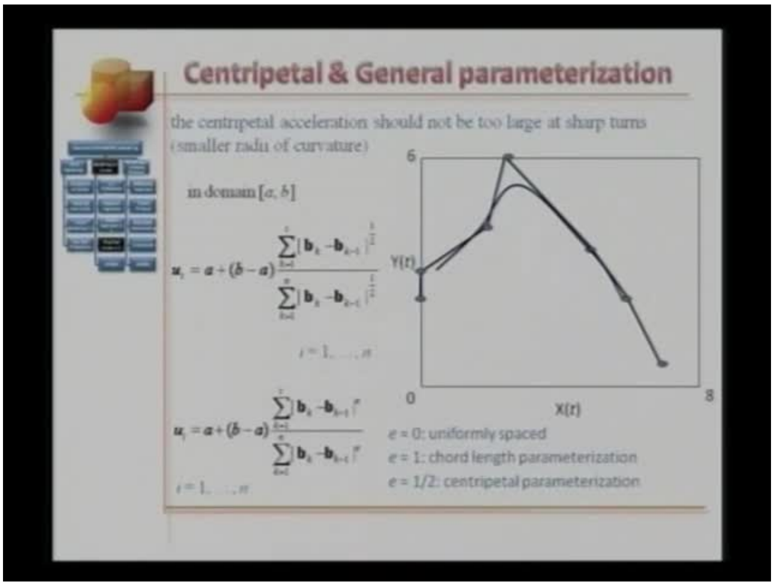


Here parameter representing, relative placement of data points can be generated. n plus 1 parameter that could be looking for can be placed proportional to the chord length of the control polyline. Let us work with the example, at we have been working so far. Let us we have a few data points here b 0, b 1, b 2, b 3, b 4, b 5, b 6, 7 of them. Corresponding to the first data point, we have the parameter value of 0. Corresponding to the second point b 1, we have the parameter value b 1 and so on. I can express u 1 as u of 0 plus the absolute value or the chord length between these two data points. Absolute value of b 1 minus b 0, remember these data points were vectors specifically position vectors. How about computing u 2?

I can compute u 2 as u 0 plus the absolute value b 1 minus b 0 plus the absolute value of b 2 minus b 1. In a sense u 2 is t 0 plus this chord length here, plus this chord length here. Now, for u 3 we will be performing a computation in a very similar manner. u 3 will be given by u 0 plus plus this chord length, plus this chord length, plus this chord length. We can generalized these process and we can say u of i equal u of 0 plus summation going from k to i of the absolute value of b k minus b k minus 1. Here the index i would go from 1 to n.

So, notice or realize there are two in this working here, one is index k the other one the index i. Now, to set the parameterization in a b, we have to perform some scaling. We can do this scaling and this manner. u of i equals a plus b minus a times. In the numerical we have summation k equals 1 to i of the absolute value b k minus b k minus 1. In the denominater we have index k going from n to 1 in this summation of the absolute value of b k minus b k minus 1. So, we notice the denominated here represents the all sum chord length,

(Refer Slide Time: 21:37)



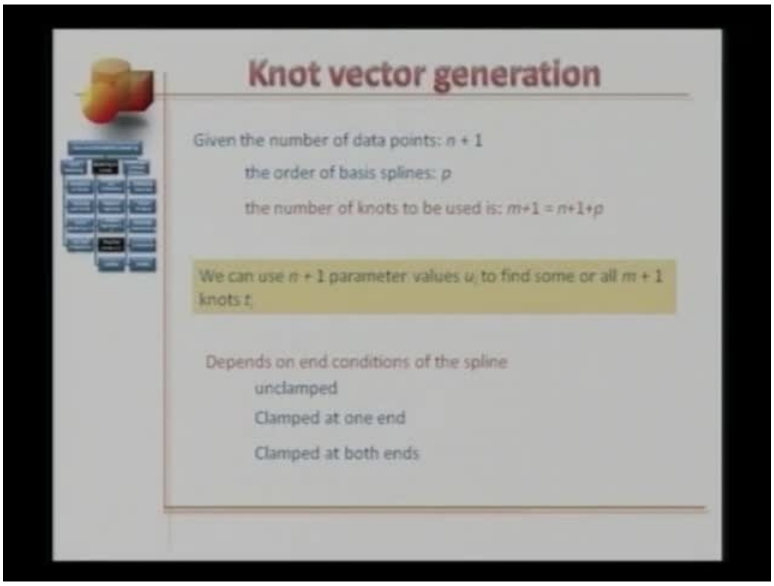
Now, notice how the scaling works? Let us say, for i equals 0 you looking of the first parameter here, i equals 0, this is equal 0 of which u of 0 will be equal 2 a, for i equals n the numerator and the denominator or identical. So, this fraction will be 1 or which u of n is equal to a plus b minus a, which is b. So, all these chore length parameters are now the intervals of a and b, but they are not uniformed. In fact they are placed continuously as per the chord lengths.

Centripetal parameterization technique. You are working with the same example, the idea the centripetal acceleration should not be too large at sharp turns of the smaller radii of curvature. Now, in the domain given by two value a and b between the chore length parameterization and centripetal parameterization. This exponent in both numerical and the denominator is the only differences. So, u of i is given by u of i equals to a plus b minus a times. In the numerical we have k plus 1 to i summation of the absolute value of b k minus b k minus 1 raise to half.

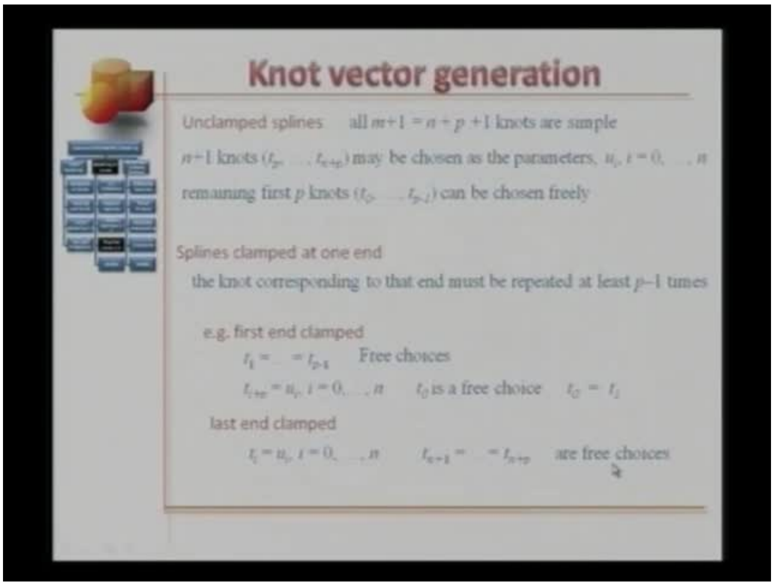
So, if we notice we are considering this square root of the co ordinance. In the denominator we have k equal 1 to n in this summation of the absolute value of b k minus b k minus 1 raise to half again. There we index i is going from 1 to n now. What I can do is, I can combined the uniform parameterization technique, the chord length as well as centripetal techniques in to a single expression. Given by u equals a plus b minus a times summation k going from 1 to i of the absolute value b k minus b k minus 1 raise to e some exponent. I talked about that while over the summation k going from 1 to n of the absolute value of b k minus b k minus 1 raise to e again.

The index i is goes to as usual 1 to n. Now, for the exponent e equal 0, we get uniformly space parameters for e equals to 1. We achieve the chord length parameterization and e equals half we get centripetal parameterization. Well I can use e as a p choice. I can specify any value I want for it. It would not make much of a difference in terms of this expression, but yes, it space how these parameter value in different ways. Now, so far you would probably not realize the relation between these parameter values of u i and not the vector of knots. You might little be a patience as of now, we would be able to specify that relation very soon.

(Refer Slide Time: 25:22)



(Refer Slide Time: 26:59)

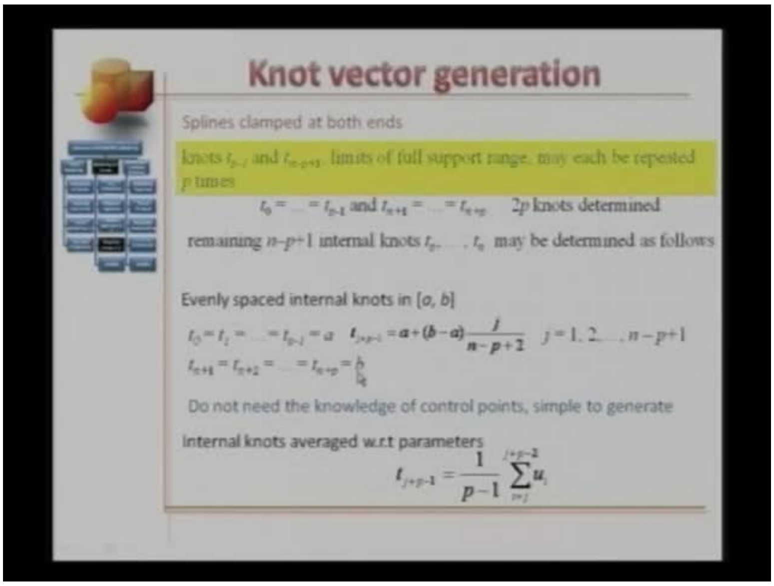


Coming to knot vector generation the given the number of data points n plus 1, the order basis splines P. These are the choices that be used in the exercise. You would know very well that number of knots could be used is given by n plus 1, it has to be equal to n plus 1 plus p the numbers of the design points plus the order the B spline curve. We can use n plus 1 parameter values u I, that we have seen before to find some or all m plus 1 knots t i. This is how these parameters getting related to the knots. Now, this would depends on the end conditions that you would prefer to choose for any spline enhance. Well we can talk about unclamped B splines, may be those which are clamped at one end or some which are clamped at both ends. Let us take these cases one by one.

Let us considered unclamped splines. Here we are know that n plus 1 equals n plus p plus 1 knots are all simple answers. Of these n plus 1 knots namely t p, t p plus 1, t p plus two, up to t n plus p, may be chosen as the parameter values themselves. u of i well, i is go from 0 to n. This u of i can either be uniform or they can be computed using chord length or centripetal parameterization. Well how about the remaining first p knots? We can choose these knots t 0 t 1 up to t p minus 1, freely.

In the next case, is of splines clamped at 1 end. Well you have seen before that the knot corresponding to that end must be repeated at least p minus 1 times. Why this is so? Once again we have to use the local by it is centre city property curves. For example, the first end for the clamped, then we have the knots t 1 equals t 2 t 3 up to t p minus 1. These would be a free choices. Now, the knots t i plus p can be set to be equal to u of I, that we have would be computed using different parameterization, which we discussed. This now, index i is going from 0 to n.

(Refer Slide Time: 31:29)

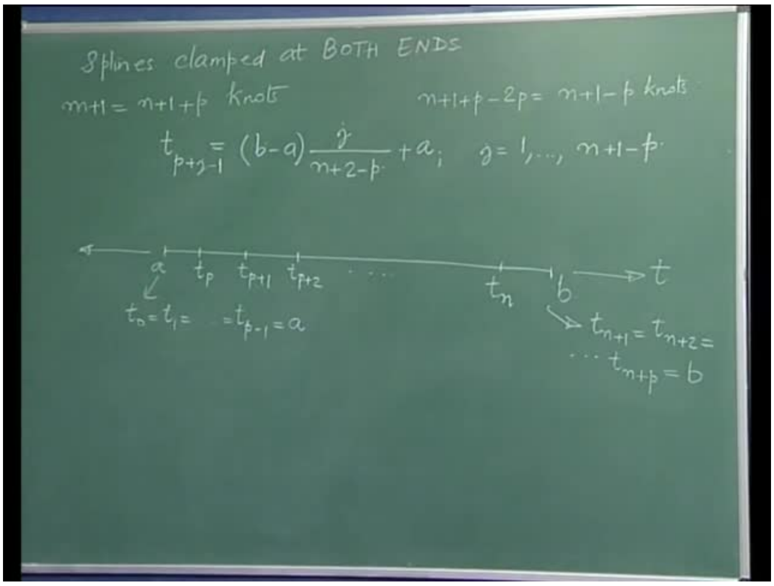


How about t sub zero? As you would know this is free choices, so we can said t 0 equals t 1, implied that the first p knots now are equals. It does not change things much, but still getting splines clamped at the first time. Now the other case, the last end clamped. Well here we can have the n plus 1 knot as t i equals u i. We can set t n plus 1 equal t n 2 equal t n plus 2 equal t n plus 3 up to t n plus p. How many of these? There are p of them and there are free choices for us.

Now, this 1 could be a trigged one, splines clamped at both ends. Now, notice at the knots t p minus 1 and t m minus p plus 1, which are the limits of the full support range may each repeated p times. This could mean that t 0 equals to t 1 equals to t 2 up to t p minus 1. So, these are the first p knots, which are equals. t n plus 1 equal t n plus 2 equals t n plus 3 up to t n plus p, these are the last p knots, which are equals. Note that n will be equal to n minus p. So, we have determined p plus p 2 p knots enter it.

So, this would correspond to of free choice for me and this would correspond to another free choice. How about the rest of the knots? They will remaining n plus p minus 1 internal knots given by t p t p plus 1 t p plus 2 up to t sub n. These can be determined as follows. We can be either evenly spaced in the interval a b, which would mean there the first p knots t 0, t 1, t 2 up to t p minus 1 equals a. The intermediate knots given by t j plus p minus 1 can be computed as a plus b minus a times j over n minus p plus 2 here. The index j is going from 1 2 3 up to n minus p plus 1. Last p knots, which are t n plus 1 t n plus 2 t n plus 3 up to t n plus p, they can all set to be b.

(Refer Slide Time: 34:50)



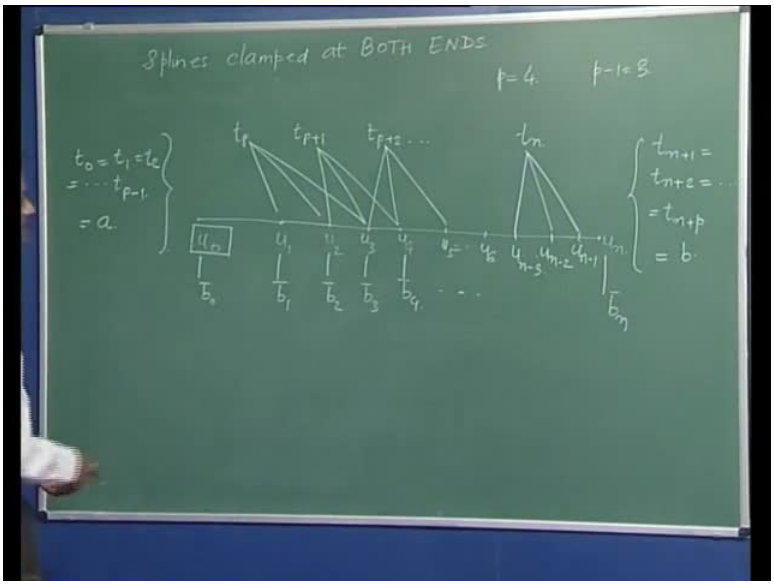
Let us try to visualize what happening here. So, we have the parameter axis here and we are trying to fit m plus 1 equals n plus 1 plus p knots within the interval given by a b we are working with splines clamped at both ends. Now, what we have done is, we have repeated the knot here a times. We have repeated here b times as well. So, we have set t 0 equals t 1 equals up to t p minus 1 here all equal to a. t n plus 1 equals t n plus 2 equals t n plus p. Here all equal to value b, what remains is n plus 1 plus p minus 2 p, which is n plus 1 minus p knots, that we need to determined within this intervals.

So, with the evenly space with the scheme here, saying that this inter mediate knots they are placed as b minus a times j over n plus 2 minus p. We have something here, do not worry about that, here j is going from 1 up to n plus 1 minus p. Now, we have to determines the knots starting from t p and ending at t n. What could be the index corresponding to this t? How would this index related to j? Clearly, this is p plus j minus 1 when j equals 1, we get to specify in the next knot t sup p.

When j equals n plus 1 minus p, the last knot becomes t sub p plus n plus 1 minus p minus 1, which is t n write here, what derive miss here? I missed this term, so this could be by knot t p corresponding to j equals 1. This is t p plus 1 for j equals to 2 t p plus 3 for j equals 3 after t n for j equal n plus 1 minus t. Now, notice that these not splines will be equal in length. Now, realize that this scheme here did not use the knowledge of the positions control points. Relatively simple for us to generate to this intermediate knots between the interval of within the interval a b.

Now, if we need to use parameterization schemes likely chord length parameterization or the centripetal parameterization, we will have to multiply this scheme a little bit. We can be thinking computing the internal knots by averaging a few parameters. This could be one of the candidate schemes. So, the internal knots t j plus p minus 1 can be equal to 1 over p minus 1, summation i going from j to j plus p minus 2 of u sub i. These are the parameters we have extracted by the chord length or centripetal parameterization here. Again t 0 t o1ne up to t p minus 1, there all equal to a and t n plus 1 t n plus 2 t n plus 3 up to t n plus p are all equal to b.

(Refer Slide Time: 41:50)



Now, what is happening here? So, we have parameter u 0, u 1, u 2, u 3, u 4, and so on up to u n. I may have generated this parameters using either chord length parameterization or centripetal parameterization. We are using the general scheme, which we have discussed before for any value of p. Let me joint on a few more parameters and so on. This value corresponding to the design point to b 0, this one corresponding to b 1 this 1 b, 2 b, 3 b, 4 and so on up to u n between correspond to the last point b n.

They are all vectors, but note of these values u 0, u 1, u 2, up to u n, they are all scalars. Now, but this times I have determined my first p knots. Also I have determinates my last p knots, say they are equal to a. Here these all equal to b, I have to compute my internal knots t p t p plus 1 t p plus 2 and so on up to t n, in such a way I can use these parameters values, that I have generated from 4. We have to compute t p, I can think of choosing the p minus 1 parameter value starting from u sub 1. So, I will have to choose u 1, u 2, u 3 up to u p minus 1, so p is 4.

The last parameter value that I have choose, u 3. Anyhow the idea is to choose p minus 1 of these parameter values. Average the mark, let us say the p equals 4 in our case. Then we have to choosing p minus 1 equals 3 parameter value at a time, averaging the mark, so for instances t p would be the average of u 1, u 2, and u 3. t p plus 1 will be the average of u 2, u 3, and u 4. t p plus 2 will be the average of u 3, u 4, u 5 and so on.

Notice that, I have not used the first parameter value u 0. I have started from the second parameter value.

How about the t n for p equals 4? Let say I do not use u of n instead I use u of n minus 1, u of n minus 2 and u n minus 3 so to compute t n, I simply be performing the average of these three parameter values. The first p knots of a and say u is equal to u sub zero with last p knots as b. I will be set u n equals b my internal p knots t p t p plus 1 t p plus 2 up to t n, can be computed as averages of these parameter values. Let us verify, if this is really the case.

Let us say for the first value of j, j equals 1, we have in the summation i going from 1 up to t minus 1. u of I, this index i will be going from 1 2 t minus 1. For j equals 1 we computing t sub t. So, in a sense t sub t is the average of the first t minus 1 none's. Of course, we are ignoring the value u sub 0 here. How about j is equals 2? i goes from 2 to p again p minus 1 parameters. These parameters will be u 2 to u p again t minus 1 above and so on. Then summary t p is the average of u 1 u 2 u t minus 1 t p plus 1 is the average u 2 u 3 up to u p.

So, on let me summarize, now you will have to specify say n plus 1 design points will have to specify the order of the B spline curve. Then once you know the number of the knots, you can use the techniques that we have discuss today to generate those knots, depending on different cases on B spline curves. You are thinking of they could simple B spline curves that mean unclamped or you can clamp a spline at one of the ends or may be at both ends. Accordingly you can think of generating the knots using different schemes that we have discussed today.