Numerical Analysis
Prof. S. Baskar
Department of Mathematics
Indian institute of Technology – Bombay

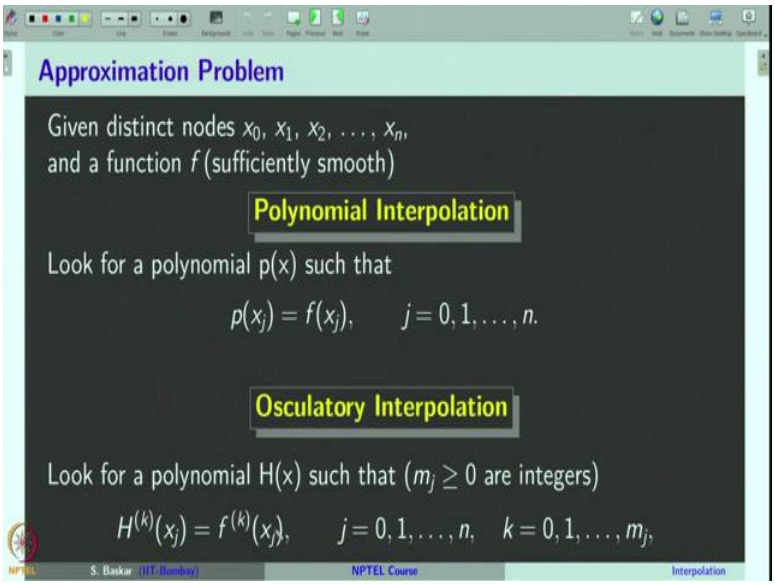
Lecture – 45
Polynomial Interpolation: Hermite Interpolation

Hi, in the previous lecture, we have seen that the Interpolating Polynomials may give poor approximation. If we have too many node points, especially if the node points are equally spaced, we may observe a visible oscillation in the graph of the interpolating polynomial near the boundaries of the interval of interest. One way to improve the approximation is to go for piecewise polynomial interpolations.

We have learned piecewise polynomial interpolations in our previous lecture and we have seen that there is still a drawback in this approach. What is the drawback? Well, the disadvantage is that piecewise polynomial interpolations using Lagrange’s or Newton's formula, can be non-differentiable at the node points. We can overcome this difficulty of loss of smoothness by imposing more smoothness conditions on the interpolant at the node points.

There are at least two ways exist to construct piecewise polynomial interpolations with more smoothness at the node points. One approach is to use piecewise Hermite interpolation and another one is spline interpolation. In this lecture we will learn the Hermite interpolation and postpone the discussion on spline interpolation to the following lecture.

(Refer Slide Time: 02:01)



Let us first define the problem of our interest. We are given n distinct nodes $x_0, x_1, \cdots, x_n$ and we also assume that the given function is sufficiently smooth. Well, before defining our problem, let us recall what we did so far in polynomial interpolations, we seek a polynomial of degree less than or equal to n such that the polynomial value at the node points coincides with the value of the functions at the corresponding node points.

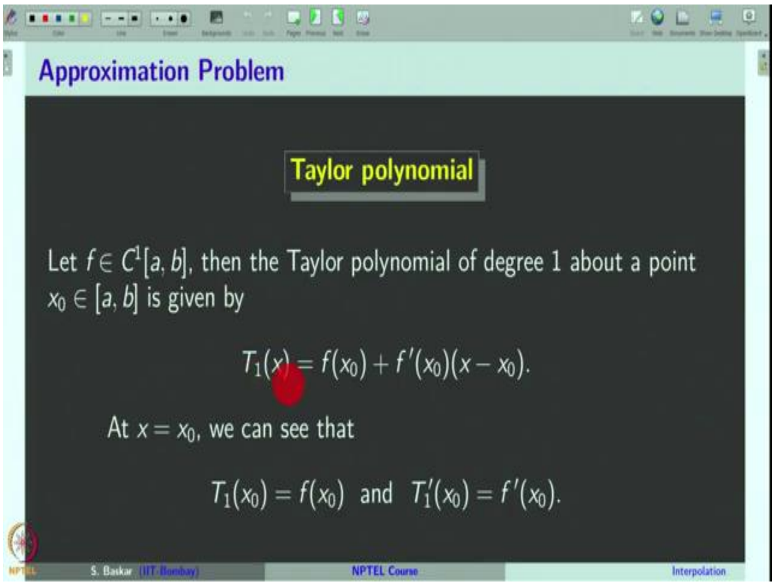
This is the interpolation condition that we have demanded on our interpolating polynomial and we got a unique polynomial of degree less than or equal to n when we are given n + 1 distinct nodes. Now, in order to get more smoothness at the node points, we will have to demand more smoothness at the node points. For this reason, we will assume that f is sufficiently smooth.

And we will look for a polynomial $H(x)$ such that the polynomial value at each node point x_j coincides with the function value at x_j. That is in this expression, I am talking about when you take k = 0 that will look something like this and in addition to this interpolation condition, we now also demand that our polynomials derivative of certain order that is, let us denote it by m_j for each j.

And we will demand the value of the derivative of the polynomial should coincide with the value of the corresponding derivative of the given function at the node point x_j. So, this is what we will demand and at every node point we may have different order of smoothness included in this condition. This is a general problem that we pose and we will ask the question whether we can find such a polynomial for a given set of n + 1 nodes.

Of course, the polynomial has to be of certain degree. We will come to that point little later and such a polynomial is generally called the osculatory interpolation or it is also referred to as the general Hermite interpolation.

(Refer Slide Time: 05:09)

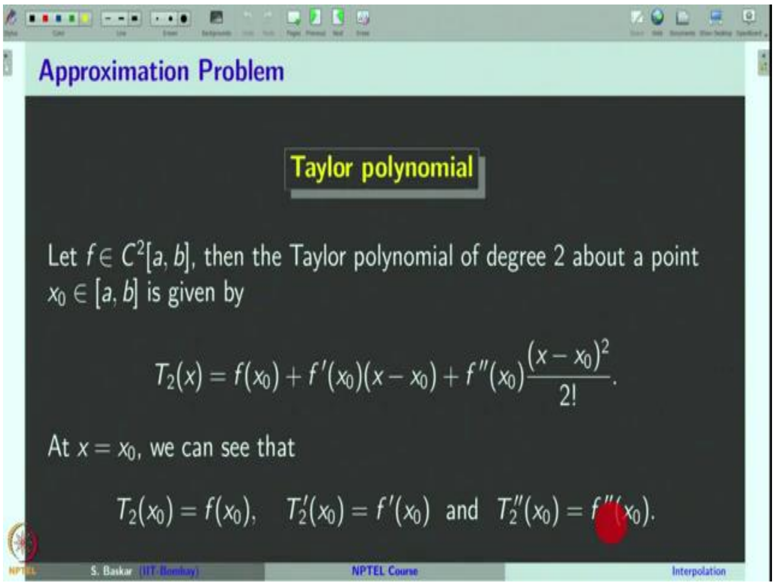


Let us see an example which is familiar to us, the Taylor polynomial. Let $f \in C^1[a, b]$. Then we know that the Taylor's polynomial of degree one about some point $x_0 \in [a, b]$ is given by $T_1(x) = f(x_0) + f'(x_0)(x - x_0)$. Now, let us take $x = x_0$ and see what happens. You can see that if we take $x = x_0$ then $T_1(x_0) = f(x_0)$.

And in fact, we can also see that $T_1'(x_0) = f'(x_0)$. Now, let us compare this property with the definition of osculatory interpolation that we have defined in the previous slide. In the case of Taylor's polynomial of degree one, we have only 1 node that is x_0. And in this polynomial that is in the Taylor polynomial of degree 1 we have taken $m_0 = 1$.

And therefore, our condition now has to be $H(x_0) = f(x_0)$ that corresponds to $k = 0$ and $H'(x_0) = f'(x_0)$. So, this is what we have obtained from the Taylor's polynomial of degree 1. Therefore, Taylor's polynomial of degree 1 is an example for the osculatory interpolation at one single node x_0 with order one at that node.

(Refer Slide Time: 07:18)

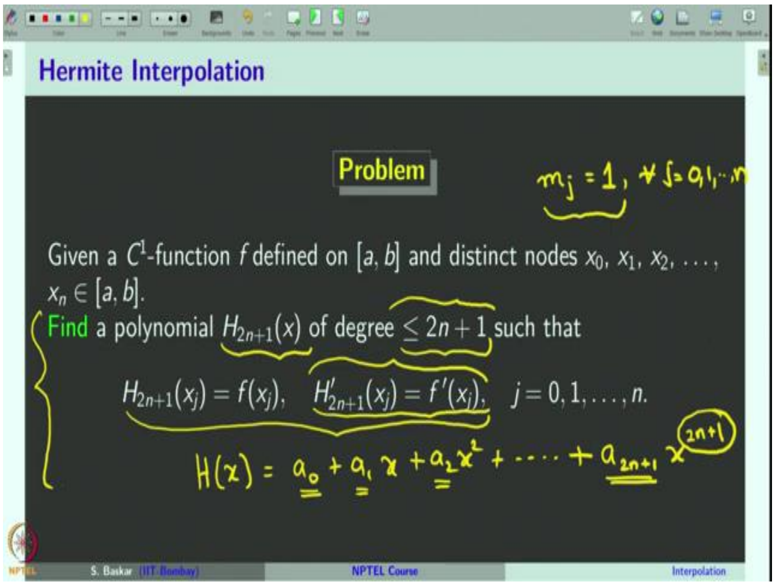


We can in fact, increase the smoothness condition and the node x_0 by 1 more. For this, we need to assume that the function f is a C^2 function on the interval $[a, b]$. Then we can see that the Taylor polynomial of degree 2 about the point x_0 in the interval $[a, b]$ is given like this. Now, the question is this an osculatory interpolation at the point x_0, if so, what is the order?

Let us see, it is not very difficult for us to see that if we take $x = x_0$ in this expression then $T_2(x_0) = f(x_0)$. Then you differentiate T_2 once with respect to x and then put $x = x_0$, you can see that $T_2'(x_0) = f'(x_0)$. Similarly, you differentiate T_2 twice with respect to x and then substitute $x = x_0$, you can see that $T_2''(x_0) = f''(x_0)$.

Therefore, you can again go back to the definition of osculatory interpolations and see that T_2 is an osculatory interpolation for the function f with single node x_0 with order 2 at x_0. That is m_0 = 2 as per the notations introduced in our definition of osculatory interpolations.

(Refer Slide Time: 09:09)



Well in our course, we will restrict ourselves to a particular case of the osculatory interpolation and we refer this particular case as the Hermite interpolation. What is this particular case? Well given as C^1 function defined on an interval $[a, b]$, let us consider n + 1 node points in an interval $[a, b]$. Now, the problem is to find a polynomial $H_{2n+1(x)}$. Well, I will not always say this suffix 2n + 1, I will just say $H(x)$.

This polynomial $H(x)$ is of degree less than or equal to 2n + 1. Such that $H(x_j) = f(x_j)$. That is the interpolation condition of order 0 and then we will also impose the interpolation condition of order one, that is $H'(x_j) = f'(x_j)$. And this should happen at all the given node points. Can you see why we demand the degree of the polynomial H to be less than or equal to 2n + 1?

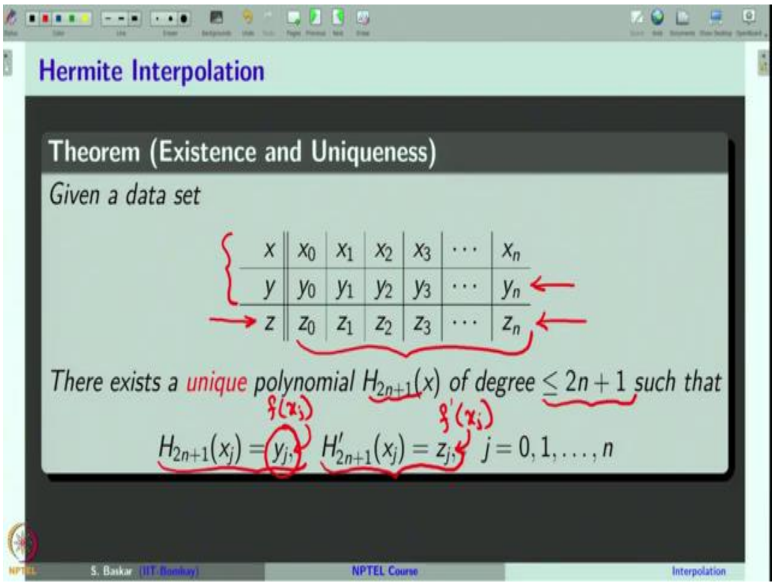
You can see that there are n + 1 conditions from the function value and another n + 1 conditions from the derivative of f. Therefore, totally we have 2n + 2 conditions. Therefore, you have to have the degree of the polynomial H as 2n + 1. Because in order to achieve these 2n + 2 conditions, we have to have the degree of the polynomial as something like $a_0 + a_1 x + a_2 x^2 +$ up to that many terms that results in 2n + 2 unknowns.

For that you need the degree of the polynomial to be 2n + 1 so that you have 2n + 2 unknowns $a_0, a_1, a_2, \cdots, a_{2n+1}$. That is why we need the degree of the polynomial to be something less than or equal to 2n + 1. Recall that this kind of condition is not something new to us. The same idea was also adopted when we were constructing the polynomial interpolation in our previous lectures.

The same idea now, but we have some extra conditions that forced us to increase the degree of the polynomial. There is nothing new in this idea. Note that we are only demanding order 1 here at each node. Therefore, if you compare the definition of oscillating interpolation in our previous slide, what we are doing here is, we are taking m_j = 1 for all j = 0, 1, 2 up to n. At every node we are only demanding the smoothness of order 1.

In that way, this problem is a particular problem of finding osculatory interpolation in general. But in our course, we will call this particular problem as Hermite interpolation. Now, the question is whether Hermite interpolating polynomial exists for a given set of data.

(Refer Slide Time: 13:27)

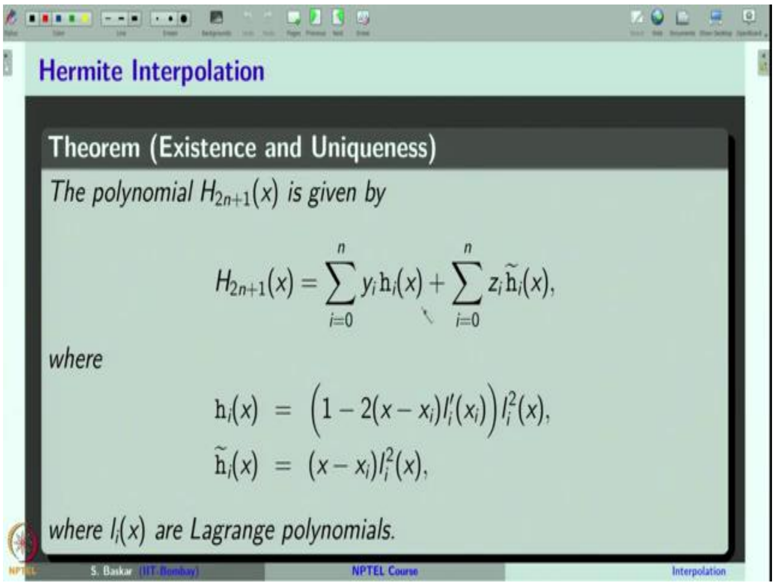


Well first, let us see how the data set should look like recall that when we were constructing polynomial interpolations using Lagrange and Newton's form, we had only two coordinates x and y. But we need one more extra coordinate now to construct the Hermite interpolation that corresponds to the value of the derivative of the given function. Well, here I have only given the value of the function as $y_0, y_1, \cdots, y_n$.

And the values of the derivative of the function as $z_0, z_1, \cdots, z_n$. Because these values may not come from some function. In general, they may come from any other source something like they may come from some experiments or so on. For that reason, I have just posed the data set in a general notation. Well once we provide this data set then our theorem says that we can construct a unique Hermite interpolating polynomial H of degree less than or equal to $2n$ + 1 with the required interpolation conditions as given here which we have shown in our previous slide itself.

Now, I am just using a different notation here. Instead of $f(x_j)$, I am just using y_j and instead of $f'(x_j)$, I am using the notation z_j.

(Refer Slide Time: 15:13)



In fact, the theorem also gives us the explicit form of the interpolating polynomial and this explicit form is given like this. The first term is the linear combination of h_i's where h_i's are given like this. And you can see that h_i's involves square of the Lagrange polynomials. Recall that $l_i(x)$ is the ith Lagrange polynomial. You should go back to our previous lectures and recall how this Lagrange polynomials are defined.

Now, h_i's are defined in terms of the square of Lagrange polynomials and also it involves the first order derivative of the Lagrange polynomial. And you can see that the first term is written as the linear combination of x_i's involving the function value y_i's. The second term is written as the linear combination of $\tilde{h}_i$ and also it involves the value of f' at the nodes denoted by z_i's, here $\tilde{h}_i(x)$ is given by this formula.

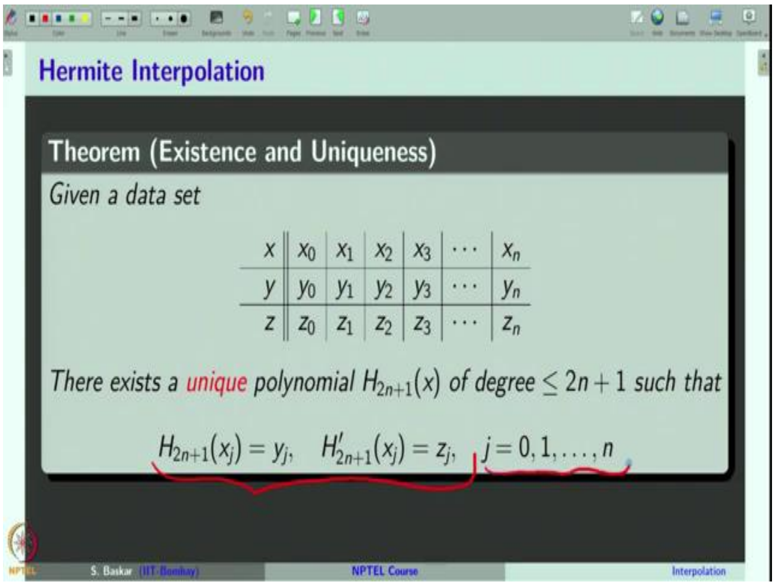
Again, you can observe that the square of Lagrange polynomial is involved in the definition of $\tilde{h}_i$ also. Well, in this way the Hermite interpolation is written in terms of Lagrange polynomials. We can also write Hermite interpolation using Newton's divided difference. We will not cover this in our course but interested students can learn how to write Hermite interpolation in terms of Newton's divided differences, from many books for instance, you can see Burden and Faires for more details.

Well, let us prove this theorem. It is not very difficult to observe that H is a polynomial of degree less than or equal to $2n + 1$. Why is it so? Well, you can see that each h_i's and also $\tilde{h}_i$'s or polynomials of degree $2n + 1$. Why? Because l_i's or polynomials of degree n. And now you are squaring them therefore l_i^2 is a polynomial of degree $2n$.

And you have one more degree coming from here. Therefore, h_i is a polynomial of degree, something $2n + 1$. Similarly, here also, you can see that l_i's of polynomials of degree n and since you are squaring this will be a polynomial of degree $2n$. And then you have one more degree coming from here that will clearly tell that H is a polynomial of degree less than or equal to $2n + 1$.

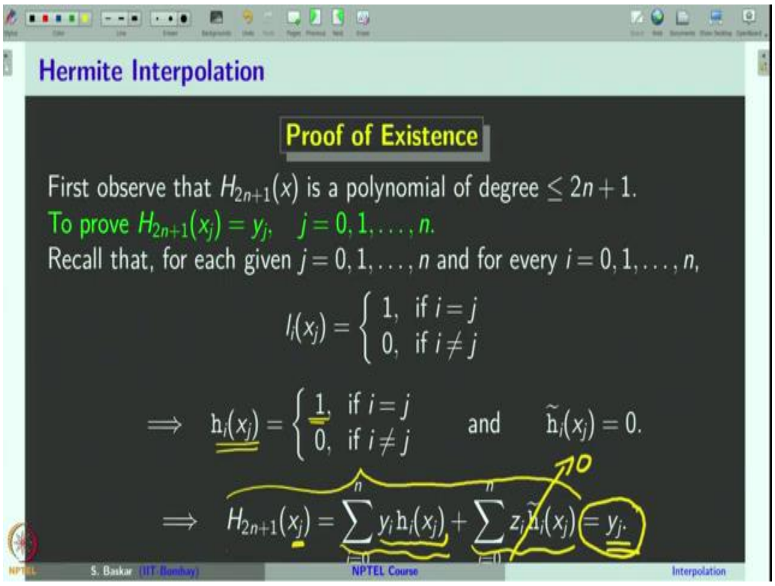
Now, our aim is to further show that the expression given like this is indeed the Hermite polynomial.

(Refer Slide Time: 18:57)



For that we have to show that the polynomial defined in our slide satisfies these two conditions at each node point.

(Refer Slide Time: 19:10)

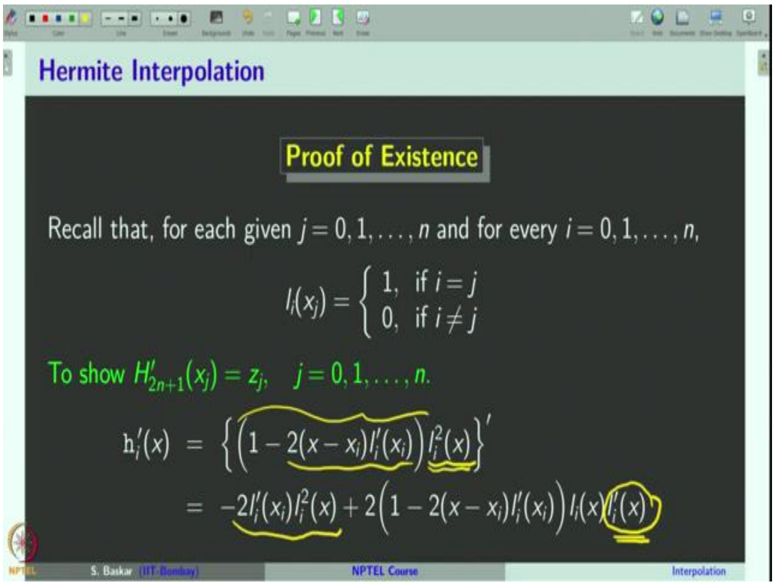


Let us see how to prove this. Let us take the first interpolation condition of zeroth order that is $H(x_j) = y_j$, for j = 0, 1 up to n. Let us see how to prove this recall in one of our previous lectures, we have proved this important property of the Lagrange polynomial. Let us use this property to prove this interpolation condition, how to do that. Well, you can use this property of the Lagrange polynomial directly into the definition of h_i and $\tilde{h}_i$

And you can see that $h_i(x_j)$ also satisfies the same property as the Lagrange polynomial. And also, you can see that $\tilde{h}_i(x_j)$ = 0 for each j. Now, going back to the expression of H that we have proposed in our statement, you can see that $H(x_j) = \sum_{i=0}^{n} y_i h_i(x_j)$. And you can see that $h_i(x_j)$ = 1, only when $i = j$. All other terms will vanish in this sum leaving only y_j and what happens to the second term.

Well, the second term will vanish fully. Therefore, our first interpolation condition is satisfied by the H that we have defined in the statement of our theorem. Therefore, the interpolation conditions of order zero is proved.

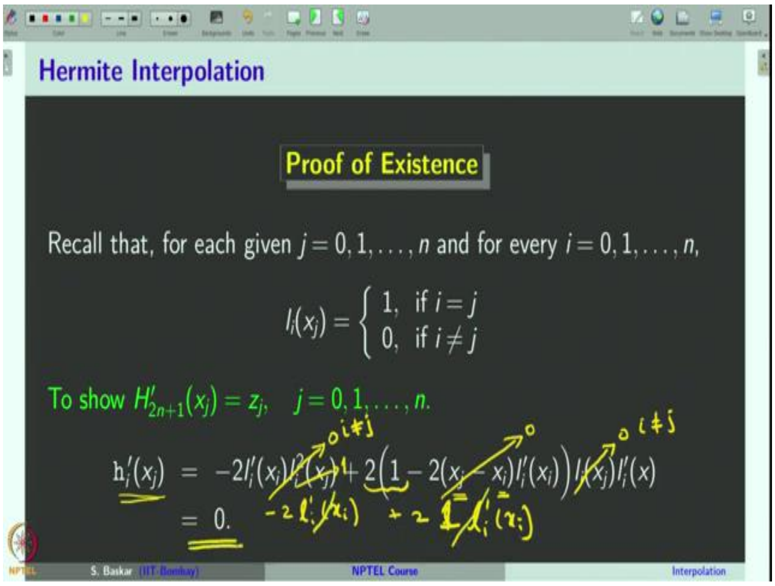
(Refer Slide Time: 21:13)



Now. Let us move on to prove the interpolation condition of order 1. For this, let us first differentiate the given expression of H with respect to x. To get this expression let us first understand how h_i' and $\widetilde{h}_i'$ are obtained, will just differentiate h_i with respect to x. We get this expression, you can see that keeping this and differentiating this gives us the first term.

And similarly, keeping this term and differentiating the second one gives us this term. Well, let us not disturb this path because this is not required in our proof. Therefore, we will not try to compute thi,s we will keep it as it is. And see what happens to this expression when we put $x = x_j$.

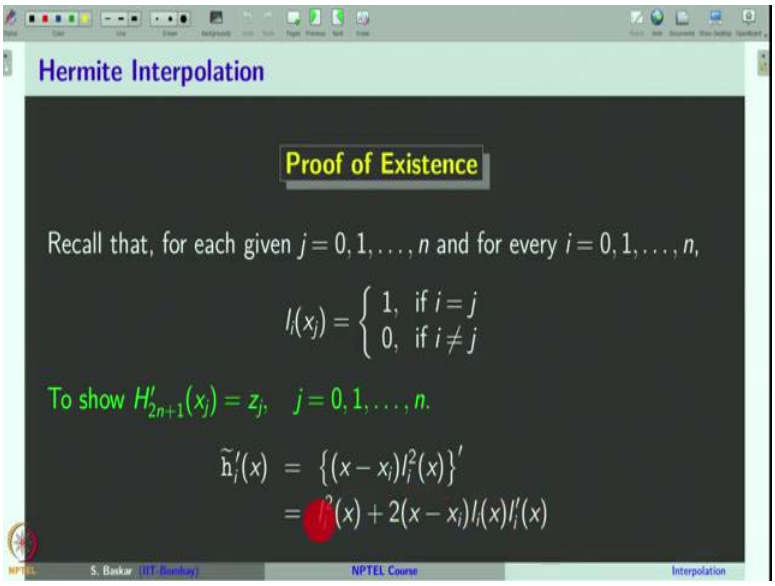
(Refer Slide Time: 22:25)



You can see that when you put $x = x_j$ then this term vanishes for all $i \neq j$. For $i = j$, you will have $-2l_i'(x_i)$. Because this term will become 1 in that case and let us see what happens to the second term. Again, the second term will vanish for all $i \neq j$. Here also it vanishes for all $i \neq j$. And what happens when we put $i = j$. Then again, this part of the term will vanish because you have *j* here and *i* here, when $j = i$ this term vanishes.

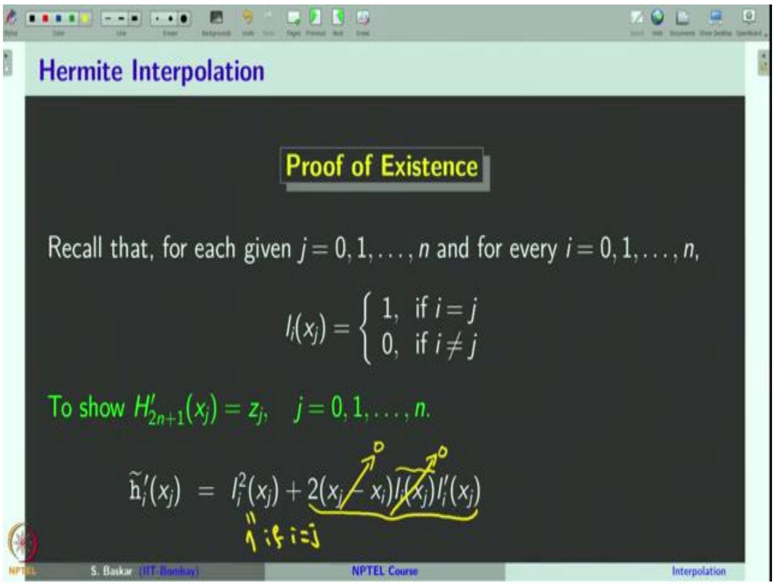
And you will have only the contribution coming from the first term which will be + $2l_i(x_i)$ which will be $y_1 l_i'(x_i)$. So, your first term is $-2l_i'(x_i)$, the second term is $+2l_i'(x_i)$, they will get cancelled and you will have 0.

(Refer Slide Time: 23:54)



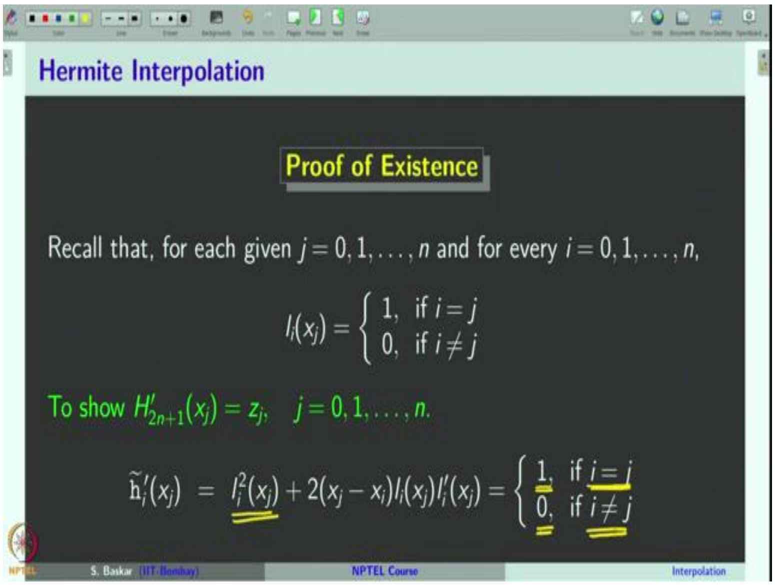
Therefore, you can see that the first term of H' now becomes 0 and we are left out only with the second term. Now, let us see what happens to $\tilde{h}_i'(x_j)$. For that first, we have to differentiate $\tilde{h}_i$ with respect to x whose expression is given like this, when you differentiate it, you get this expression. Again, in this we have to put $x = x_j$ and let us see what happens.

(Refer Slide Time: 24:31)



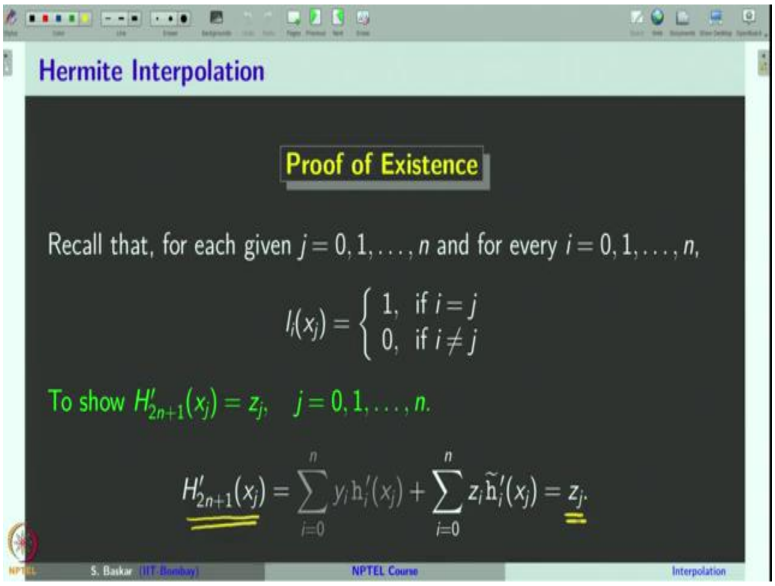
When we put $x = x_j$ we get this expression. You can see that this is equal to 1, if $i = j$ and it vanishes for all $i \neq j$. And then what happens to the second term, well, for $i \neq j$ this part will vanish when $i = j$. This will not vanish but this will make this second term to vanish. Therefore, as a whole the second term will vanish for all j = 0, 1 up to n.

(Refer Slide Time: 25:11)



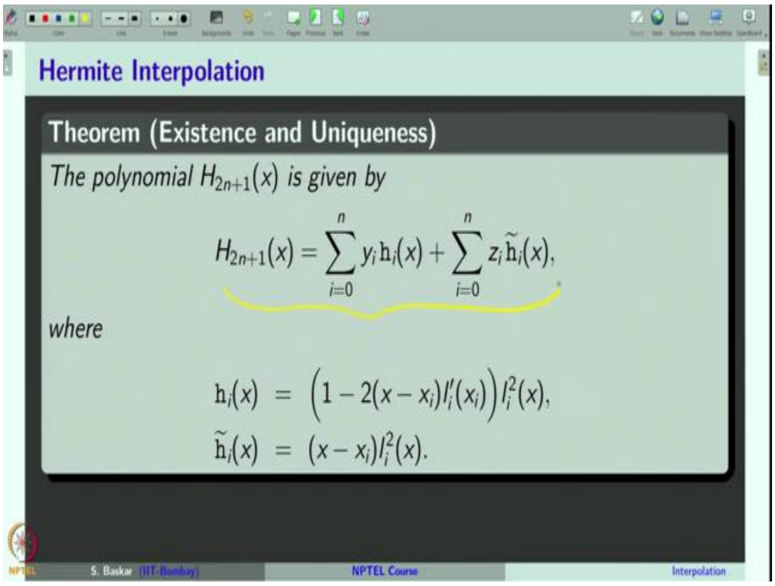
Whereas from the first term you will have $\tilde{h}'_i(x_j) = 1$, is $i = j$ and 0 if $i \neq j$.

(Refer Slide Time: 25:24)



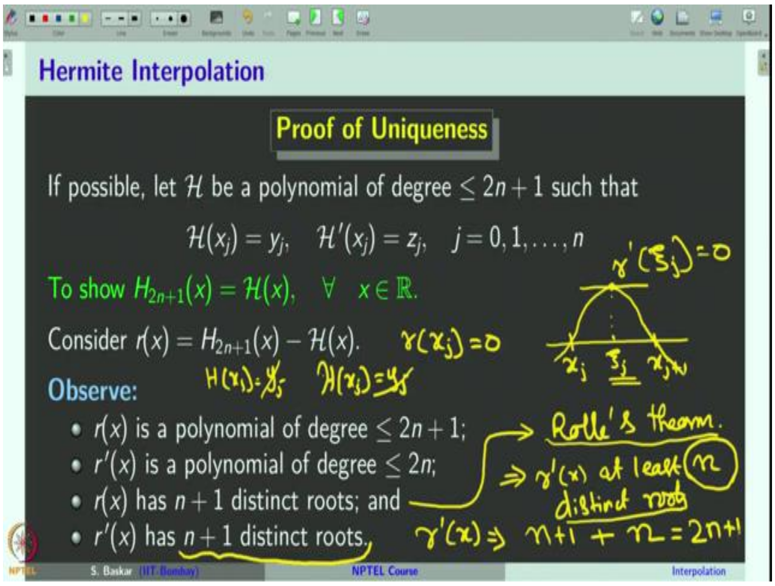
From here, you can see that $H'(x_j) = z_j$ and that proves the second level of interpolation conditions also.

(Refer Slide Time: 25:38)



Thus, we have proved the existence of the Hermite interpolating polynomial for a given data set and the formula is also given to us explicitly.

(Refer Slide Time: 25:50)



Now, let us prove the uniqueness of the Hermite interpolating polynomial for a given data set. If possible, we will assume that there exists another polynomial of degree less than or equal to $2n + 1$ with the same condition. And let us see what happens, remember, we already constructed a polynomial in this form and now what we are doing is we are assuming that there is another polynomial with the same property.

Now what we have to show? We have to show that the polynomial that we constructed should be the same as the polynomial that comes from somewhere. So that is what we want to show. If you show that then it means that this is the only form that you can have for the Hermite interpolation. Let us define $r(x)$ as the difference between these two polynomials.

Therefore, in order to prove that these polynomials are equal for all x, we have to prove that $r(x) = 0 \; \forall x$. To do this let us first observe the following properties of $r(x)$. First, is that since both H and $\mathcal{H}$ are polynomials of degree less than or equal to $2n + 1$ you can see that $r(x)$ is also a polynomial of degree less than or equal to $2n + 1$. Second thing is, since r is a polynomial of degree less than equal to $2n + 1$, $r'(x)$ will be a polynomial of degree less than or equal to $2n$.

Also, from the first set of interpolation conditions, we can see that $r(x)$ has $n + 1$ distinct roots which are precisely the distinct node points from our given data set. This is because at any node point x_j, $r(x_j) = H(x_j) - \mathcal{H}(x_j)$. But we know that $H(x_j) = y_j$ and $\mathcal{H}(x_j)$ is also equal to

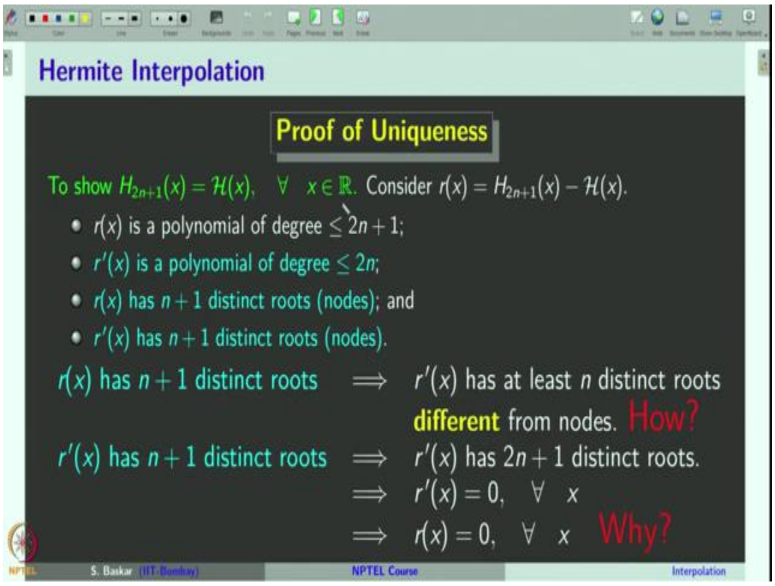
y_j. Therefore, they get cancel and you will have $r(x_j) = 0$ and this happens for each j = 0, 1, 2 up to n.

Therefore, r has n + 1 distinct roots. Also, you can see that r' has n + 1 distinct roots. Why? Again, you differentiate r with respect to x that will be $H'(x) - \mathcal{H}'(x)$. Again, when you put $x = x_j$, you have the second set of interpolation conditions that will make $r'(x_j)$ also equal to 0. Therefore, the polynomial $r'(x)$ will also have n + 1 distinct roots.

Now, let us take this condition, that is $r(x)$ has n + 1 distinct roots and we will use the Roll's theorem. That implies that $r'(x)$ will have at least n distinct roots. That is in between two node points, say x_j and x_{j+1} say, r is something like this. Then you can always find a point in between x_j and x_{j+1}, let us call this as ξ_j at which $r'(\xi_j)$ will be equal to 0.

It means all this n roots which you found from the Rolle's theorem, are different from the n + 1 distinct roots which we have already from our data set. This ξ_j's may not coincide with the node points that we have. In that way $r'(x)$ already has n + 1 distinct roots + now you have n distinct roots. Therefore, $r'(x)$ will have 2n + 1 distinct roots.

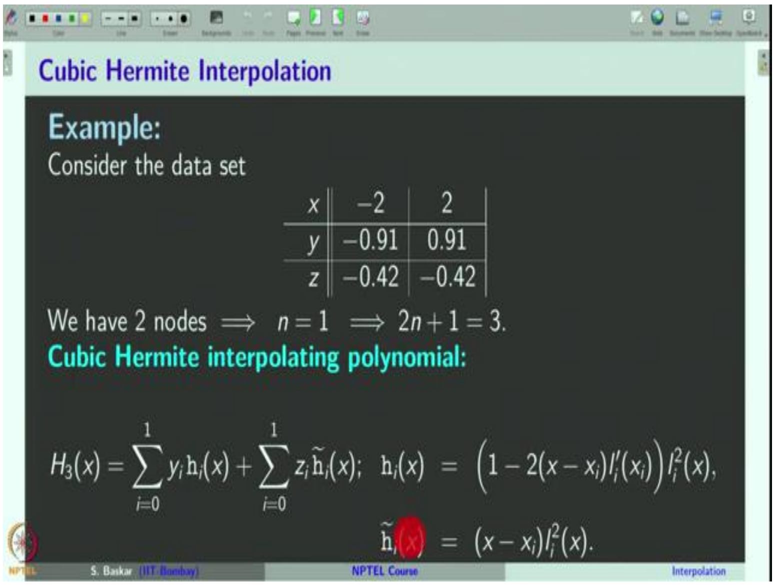
(Refer Slide Time: 31:02)



That is what I am saying here, $r(x)$ has n + 1 distinct roots implies r' has at least n distinct roots different from the node points. How? well, using the Roll's theorem and once you have this, you can see that $r'(x)$ has 2n + 1 distinct roots. What is the problem with that? Well, $r'(x)$ is a polynomial of degree less than or equal to 2n. But now it has 2n + 1 distinct roots.

That implies that $r'(x)$ is a zero polynomial. That implies that $r(x)$ is a constant polynomial but we know that $r(x)$ has $n + 1$ distinct roots. In fact, if you know that $r(x)$ has one root, that is enough to say that this constant polynomial is equal to 0. Thus, we have proved that the polynomial $r(x)$ defined as $H(x) - \mathcal{H}(x)$ is indeed a zero polynomial. That implies that $H(x)$ is equal to $\mathcal{H}(x)$ for all $x \in \mathbb{R}$ and this proves the uniqueness of the Hermite interpolation.

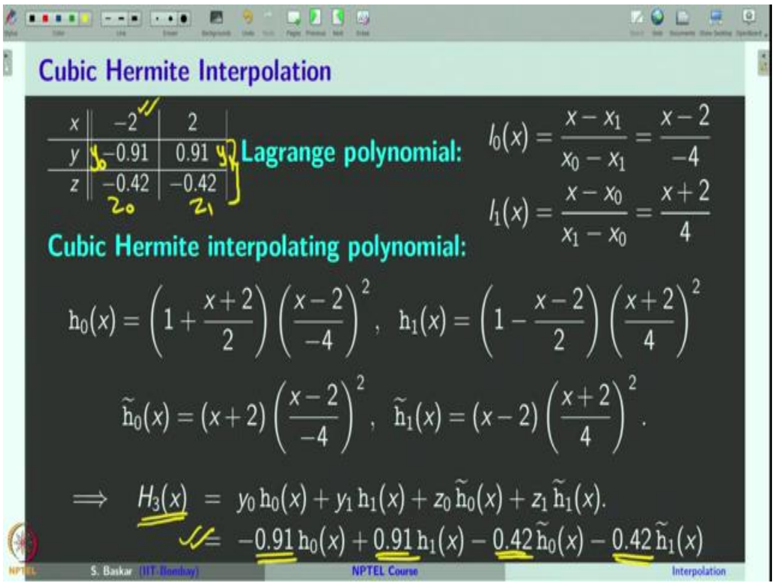
(Refer Slide Time: 32:32)



Let us take an example and construct the Hermite polynomial for this given data set. Remember, we have two distinct nodes x_0 and x_1. Which implies that we have $n = 1$ and therefore, the degree of the Hermite polynomial is $2n + 1$ which is equal to 3. That is, we have to construct the cubic Hermite interpolating polynomial from the given data set. Let us recall the formula for H_3 from our theorem.

H_3 is given like this where now we have $n = 1$ and h_i's and $\tilde{h}_i$ are given, as they are in the theorem. To compute the cubic Hermite interpolating polynomial, we first have to find h_i and $\tilde{h}_i$ and then we can write them in this form.

(Refer Slide Time: 33:49)



Remember, in order to find h_i and $\tilde{h}_i$, first we have to find the Lagrange polynomials. Therefore, we will start with computing the Lagrange polynomials. As a first step, we will compute $l_0(x)$. Remember this is your x_0 and this is your x_1. Therefore, $l_0(x)$ which is equal to $\frac{x-x_1}{x_0-x_1}$ is given like this. Similarly, $l_1(x)$ which is equal to $\frac{x-x_0}{x_1-x_0}$ is given like this.

Now, we have to construct the cubic Hermite polynomial from this data set. Let us recall the formula for $h_i(x)$ which is given like this, where i = 0 and 1. For i = 0, $h_0(x)$ is written in terms of l_0 which is given like this and for i = 1, $h_1(x)$ is written in terms of $l_1(x)$ which is given like this. Remember we have to also differentiate the Lagrange polynomials in order to substitute here.

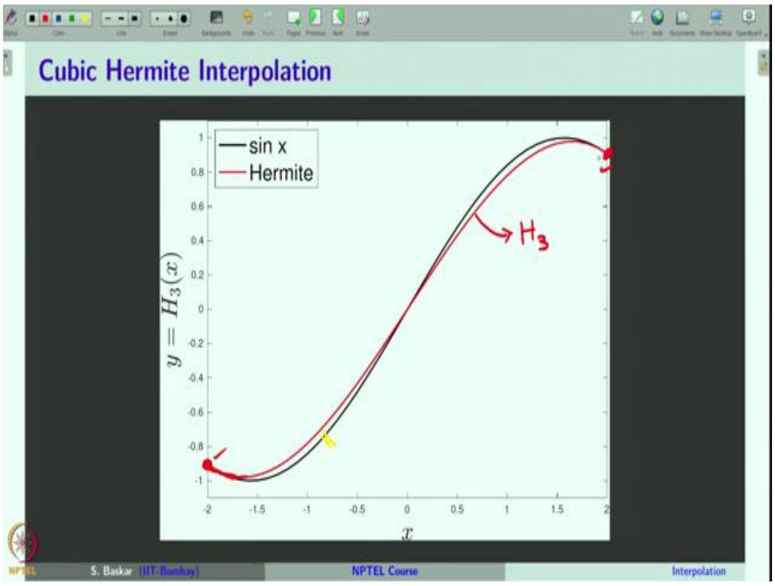
Let us see how to get $h_0(x)$, $h_0(x) = \big(1 - 2(x - x_0)\big)$. Therefore, you have x + 2 and then you have $l_0'(-2)$. You can obtain that from here and you can write it here. And finally, this term will become like this into $l_0^2(x)$. Similarly, you can get the expression for $h_1(x)$ also and that is given by this. Once we have this let us now go to find $\tilde{h}_i$. How $\tilde{h}_i$ is given?

Recall the formula for $\tilde{h}_i$ is this and again for $\tilde{h}_0$, we have to use $l_0(x)$ and similarly, for $\tilde{h}_1$ you have to use $l_1(x)$. And for $\tilde{h}_0(x)$, we obtain the formula like this and $\tilde{h}_1(x)$ is obtained like this. Now, we have $h_0, h_1, \tilde{h}_0$ and $\tilde{h}_1$. Therefore, we can now go to write the cubic Hermite interpolation polynomial.

Remember the formula is given like this, you have $y_0 h_0 + y_1 h_1 + z_0 \tilde{h}_0 + z_1 \tilde{h}_1$. What is y_0? This is y_0 this is y_1. This is z_0 and this is z_1. So, you have to substitute these values and you can leave it in this form or if you wish, you can also simplify it to see that it is indeed a polynomial of degree less than or equal to 3.

Let us visualize H_3 and in fact this data are taken from the sin function with the x values are taken in radians.

(Refer Slide Time: 37:32)

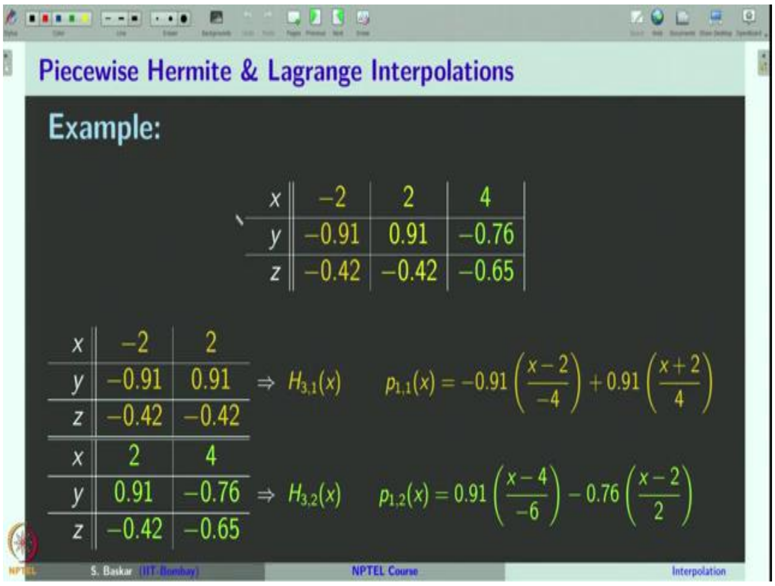


Here, the black solid line represents the graph of the sin function, this one and the red solid line, this one is the graph of the Hermite polynomial H_3 that we have constructed just now. Note that from the given nodes x_0 and x_1 with only the function values, we will get the interpolating polynomial of degree less than or equal to 1 from the Lagrange form or Newton's form.

But here with two node points, we got a cubic interpolating polynomial. Of course, we have to also provide the information of the value of the derivative of the function. That is the cost we are paying here but we are getting a higher degree polynomial with just two nodes. This is what we observe here. You can see that at the node points, the function, value and the polynomial value or coinciding.

This is the first set of interpolation conditions and also you can see that at the node points, the slope of the polynomial and also the slope of the function are coinciding. This is the second level of the interpolation condition which is clearly visible in this graph.

(Refer Slide Time: 37:07)



Well, as a next example, let us consider three nodes and the corresponding function values and the values of the derivative. Again, I have taken these values from the sin function with 2 digit rounding. Note that, with this data set, we can obtain the fifth degree Hermite interpolating polynomial but my interest is to obtain the piecewise cubic Hermite interpolation. For this let us first give the data set in two pieces.

The first piece has two nodes x_0 and x_1 and the corresponding function values and the derivative values. This will give us the corresponding cubic Hermite polynomial interpolation. Remember we have constructed this polynomial in our previous example. But now we will denote it by $H_{3,1}(x)$ because this is the polynomial that is coming from the first piece of the given data.

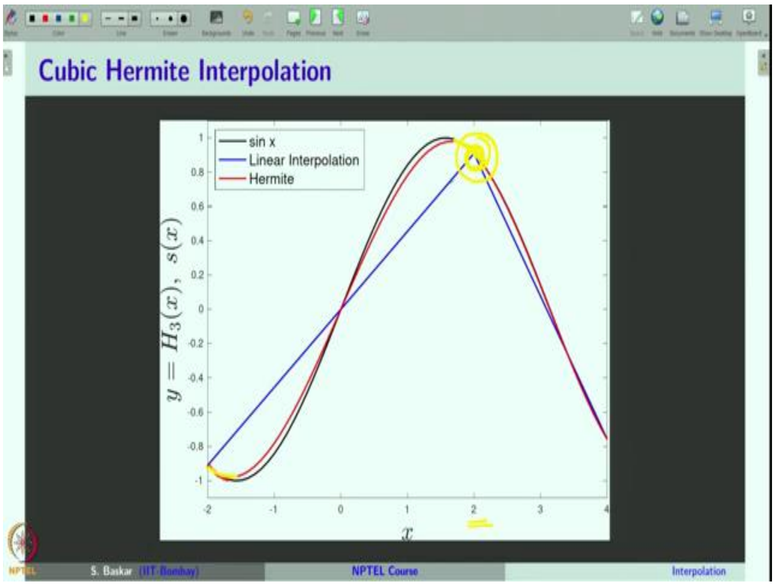
Now, let us take this second part of the data set. That is, the nodes 2 and 4 and their corresponding values of the function and the derivatives. Again, we can construct the cubic Hermite interpolation in a similar way, as we did in the last example and let us denote this Hermite interpolating polynomial by $H_{3,2}(x)$. Because this polynomial is coming from the second piece of our data set.

Now, we will join these two to get the piecewise cubic Hermite interpolation. Remember with the same node points with only the function values we can also obtain quadratic interpolating polynomial in the Lagrange or Newton form or we can also find piecewise linear polynomial interpolation. You can see that the piecewise linear interpolating polynomials with the same data set, can be given like this.

Here $p_{1,1}(x)$ is the linear interpolating polynomial written in the Lagrange form coming from the first piece of the data set. And $p_{1,2}(x)$ is the linear interpolating polynomial, again written in the Lagrange form coming from the second piece of the given data set. Remember here we are only using the function values but not the derivative values to construct the Lagrange polynomials.

The derivative values are used only for the Hermite polynomials. Again combining these two piece of linear polynomials, we can obtain the piecewise linear polynomial for the given data set and we also can obtain piecewise cubic Hermite interpolating polynomial for the given data set.

(Refer Slide Time: 42:34)

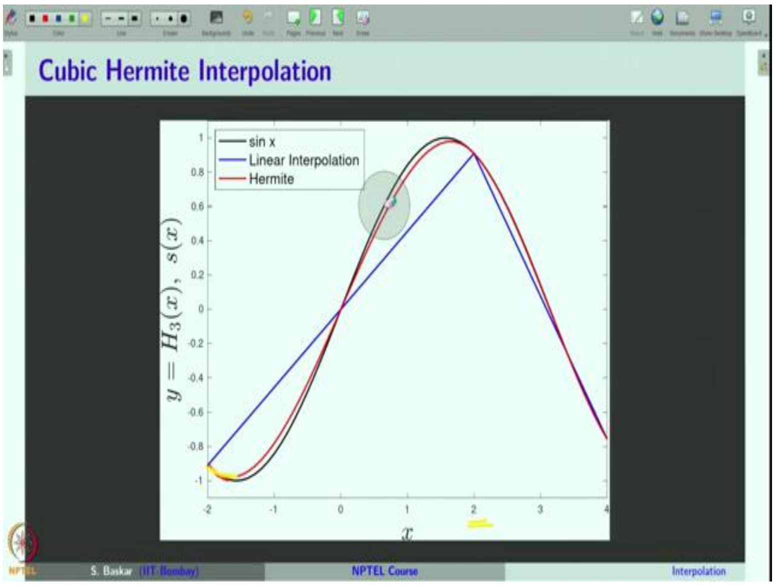


Let us try to see graphically how they look like? The blue solid line represents the piecewise linear polynomial interpolation. As expected, we can see that there is a sharp edge at the interior node of the linear piecewise polynomial interpolation. As remarked at the beginning of this lecture and also in our previous lecture, the piecewise linear polynomial interpolation is not differentiable at the interior node point.

Whereas you can see that the cubic Hermite interpolation is coinciding with the slope of the function at the node points. That is at – 2 here, as well as at the node point 2 here. And similarly, the second piece that is $H_{3,2}$ is again coinciding with the slope of the sin function at the point

x = 2. And that makes the piecewise cubic Hermite interpolation to be at least C^1 at the point , = 2 that is the interior node point. Whereas, as I told the piecewise linear interpolation has a sharp edge here.

(Refer Slide Time: 44:06)



So that is the main advantage of the Hermite interpolation. When we go to construct piecewise Hermite interpolation, we will gain one order of smoothness at the interior nodes, whereas this is not the case with the piecewise interpolation coming from the Lagrange's or Newton's form of interpolation.

As we remarked at the beginning of this lecture, there is another way to construct piecewise polynomial interpolation, with more smoothness condition at the node point called this spline interpolation. We will discuss spline interpolation in our next lecture. Thank you for your attention.