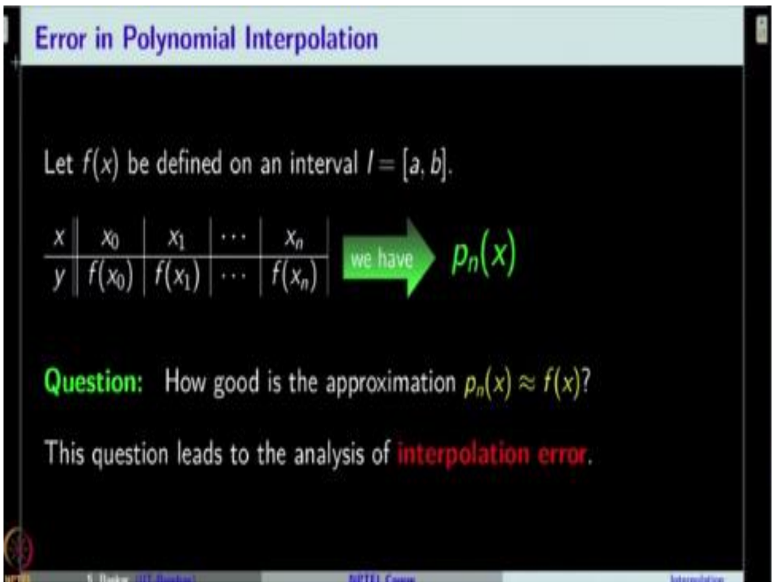
Numerical Analysis
Prof. S. Baskar
Department of Mathematics
Indian Institute of Technology, Bombay

Lecture - 41
Polynomial Interpolation: Mathematical Error in Interpolating Polynomial

Hi, we are learning polynomial interpolation for univariate functions. In this we have learned the existence of uniqueness results and also, we learn two ways to construct polynomial interpolation. In this lecture, we will start our study on errors involved in polynomial interpolations. We will study in detail the mathematical error involved in the polynomial interpolation in this class. There are two levels of errors involved in the polynomial interpolation.

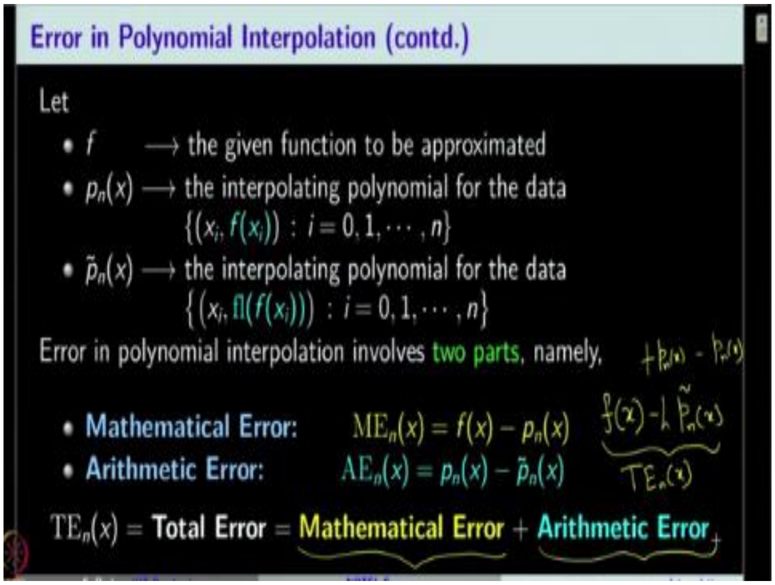
(Refer Slide Time: 01:01)



We will quickly see what they are. You are given a function f defined on the interval $[a, b]$. And you are also given n+1 nodes in the interval $[a, b]$. Generally, we take x_0 as a and x_n as b. And now we have generated this data set by taking the function values at the node points $x_0, x_1, \cdots, x_n$. And we know that we can find a unique interpolating polynomial for this data set, let us call it $p_n(x)$.

Now the question is, if we want to use the polynomial $p_n(x)$ as an approximation to $f(x)$. Then how good is the approximation? Well, this leads to the analysis of interpolation errors.

(Refer Slide Time: 01:57)



Let us go one step closer and see, what are all the levels of errors that are involved in the polynomial interpolation. You are given the function f which we want to approximate, from there we have generated a data set, like this. And then we have constructed an interpolating polynomial $p_n(x)$. But generally, we do this construction on a computer. Therefore, the first step of this is to feed in our data set.

When we feed in the data set in a computer, the computer mostly will not take the function values exactly because it will do the floating-point approximation when evaluating the value of the function at the node points. In that way what we get is an approximate value of the function at the node points. The computer will also take the node points in the floating-point approximation.

But to some extent this error can be minimized because the node points are chosen by the user. You may choose the node points conveniently so that the floating-point approximation, that is the rounding error involved in the node points, is minimized. In that way we will always assume that node points are exactly represented whereas the rounding error or the arithmetic error is involved only in the function values.

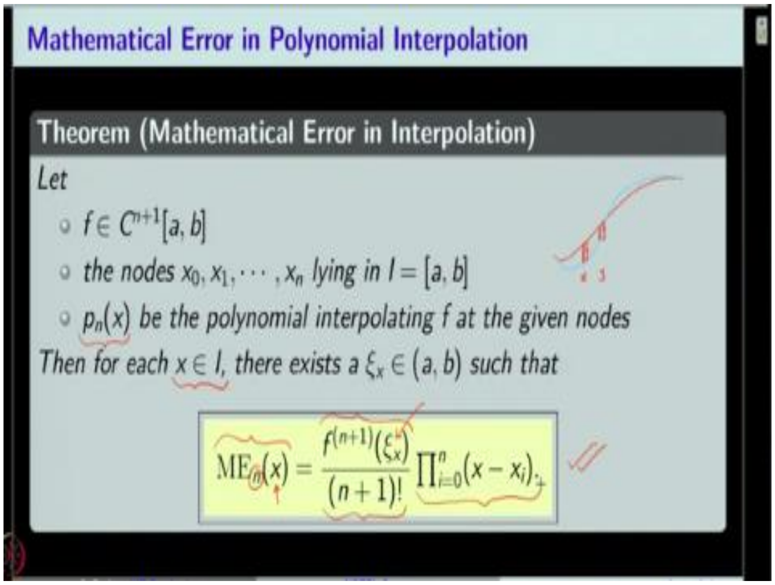
And thereby we are having a data set like this which is clearly different from the data set that we want to work with. Therefore, the computer will generate a polynomial interpolation which is different from what we want to consider. Let us denote this polynomial interpolation generated from the data set with floating point approximation as $\tilde{p}_n$. Therefore, there are two levels of errors involved in the output of the polynomial interpolation.

What is the output that we get? We get this polynomial as the output for the approximation of the function f. But we want this one which we will never get. If we go to compute the polynomial interpolation on a computer. The first level of the error is the mathematical error which is nothing but the difference between the exact value and the polynomial that we want to construct. So, this is typically the error involved due to the mathematical derivation of this approximation method.

So, we will call this error as mathematical error. The next level of error is purely because of the computational factor. That is nothing but the difference between the polynomial that we want and the polynomial that the computer has given to us. And therefore, we will call this error an arithmetic error. What is the total error? Total error is nothing but the sum of the mathematical error and the arithmetic error.

In other words, the total error is actually, what we are interested in minus what we finally got. So, this is the total error involved in this computation and that can be obtained by adding $p_n(x)$ and subtracting $p_n(x)$. And that can lead to the mathematical error plus the arithmetic error. And that is the final expression for the total error. Therefore, in order to understand total error, we have to understand mathematical error as well as arithmetic error.

(Refer Slide Time: 06:41)



In this class, let us understand the mathematical error. We have given the definition of mathematical error here as $f(x) - p_n(x)$. Now we will try to derive an expression for the mathematical error. Let us state this expression in the form of a theorem. Here we need to assume that f is a C^{n+1} function defined on the interval $[a, b]$. What does it mean? It means f is continuously differentiable up to the order $n + 1$.

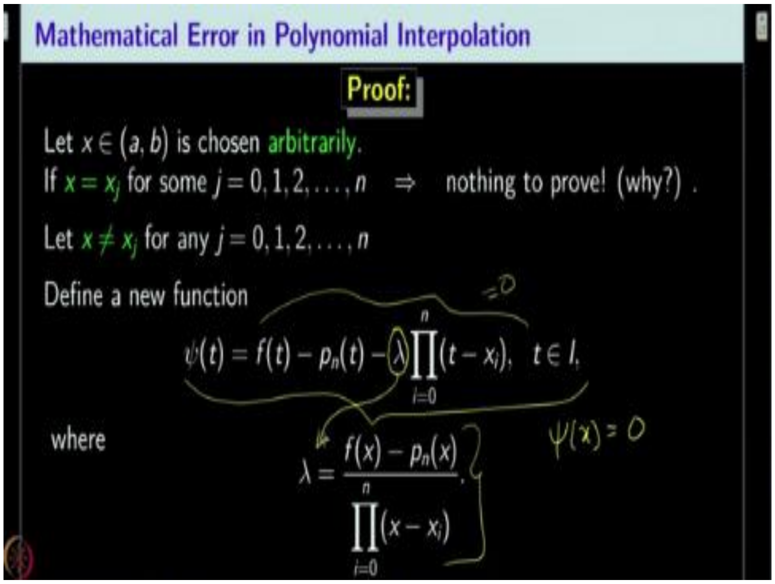
It means f is continuous and f' exists, f' is also a continuous function. Similarly, f'' also exists f'' is also a continuous function. And so on till the f^{n+1} derivative also exists and it is also continuous. That is what is meant by saying that f is a C^{n+1} function and notationally, it is written like this. We are given $n + 1$ distinct nodes $x_0, x_1, \cdots, x_n$ all are in the interval $[a, b]$ with understanding that $x_0 = a$ and $x_n = b$.

Although it is not strictly necessary to assume this. But in most of our analysis we will assume this at least when we are working with equally spaced nodes. Once we are given the nodes and the function you can generate the data set and then you can construct the interpolating polynomial for the data set which we will call as the interpolating polynomial or polynomial interpolating the function f.

Then the mathematical error involved in $p_n(x)$ when compared to the exact value $f(x)$ is given by this expression. You can see that the mathematical error will be a function of x because you have the function f and suppose you are trying to get a polynomial like this. Then at every point you can see that the mathematical error will be different, it depends on the point x. So, here the mathematical error at this point say x is different from the mathematical error at this point say y. Therefore, the mathematical error is a function of x.

That is why we write this notation ME stands for mathematical error and it depends on x and also it is the mathematical error in the interpolating polynomial of degree n. So, this suffix n is also used in the notation and that is given by the n + first derivative of f evaluated at an unknown point ξ And this ξ depends on the point x at which we are interested in seeing the mathematical error divided by $(n+1)!$ into this product. Let us try to derive this formula or expression for the mathematical error for this.

(Refer Slide Time: 10:27)



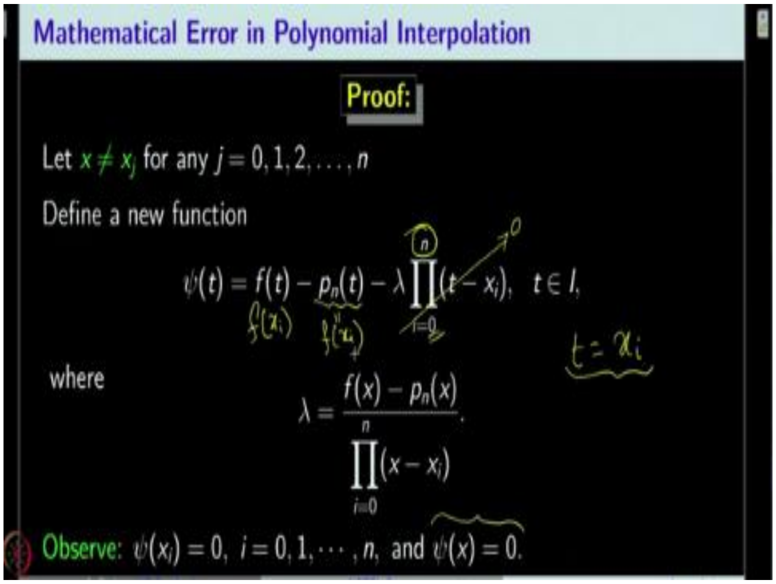
Remember we have to derive this formula for every x in the interval (a, b). For that we will choose a x arbitrarily in the interval (a, b). There are two possibilities now, x may coincide with one of the node points. Because we are choosing x arbitrarily it may happen that the choice of x is in such a way that it coincides with one of the nodes. If that is so then there is nothing to prove. Why, because if x is a node, then the mathematical error is equal to 0 because of the interpolating condition.

Therefore, there is nothing interesting for us to prove, in this case mathematical error should become 0 and it can also be seen from this expression if x is equal to one of this x_i's, then that term will become 0. And therefore, this entire product will become 0. Hence the mathematical error will be equal to 0 for any choice of ξ_x, $f^{(n+1)}$ is a continuous function defined on the closed and bounded interval therefore it is a bounded function.

Therefore, a finite number into 0 will come and that is obviously 0. So, there is nothing really interesting for us to prove because the expression is automatically giving us that the mathematical error should be 0, if x is one of the nodes. We will assume that x is not equal to any of the nodes that we have chosen to construct $p_n(x)$. In this case we have something non trivial to prove therefore let us see how to derive the mathematical error formula in this case.

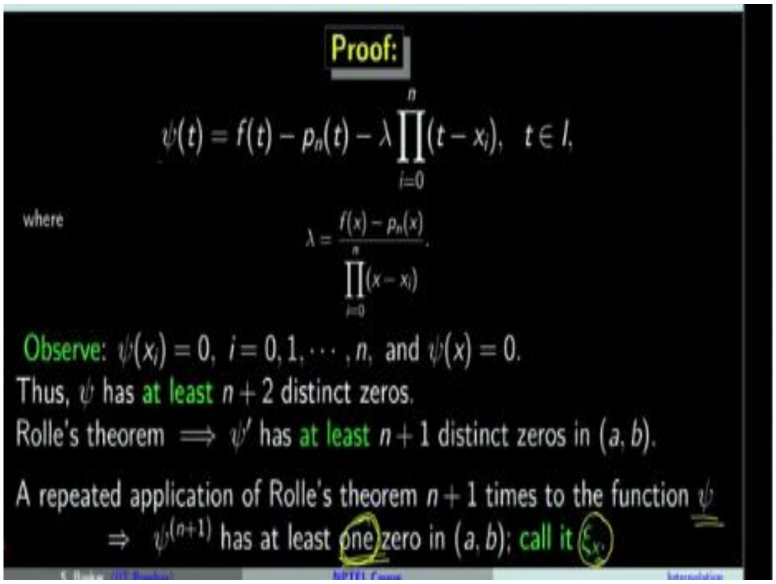
The first step is to consider a function like this, where this λ is something which we will now choose in such a way that $\psi(x) = 0$. You can see that by taking $t = x$. In this expression you will see that the right hand side becomes 0. If we choose your λ like this, that is why we have inserted a parameter λ here and then we have chosen it conveniently like this.

(Refer Slide Time: 13:13)



We have seen that x is a zero for the function ψ. You can also see that all the nodes $x_0, x_1, \cdots, x_n$ are also zeros of the function ψ, why it is so? When you take $t = x_i$ then this term will go to 0 for any i = 0, 1, 2 up to n. Any such choice will make this term to be 0 and this will be $f(x_i)$. This is our interpolating polynomial at the node points therefore this will also be equal to $f(x_i)$ thanks to the interpolation condition. Therefore, ψ will vanish at all the node points.

(Refer Slide Time: 14:09)



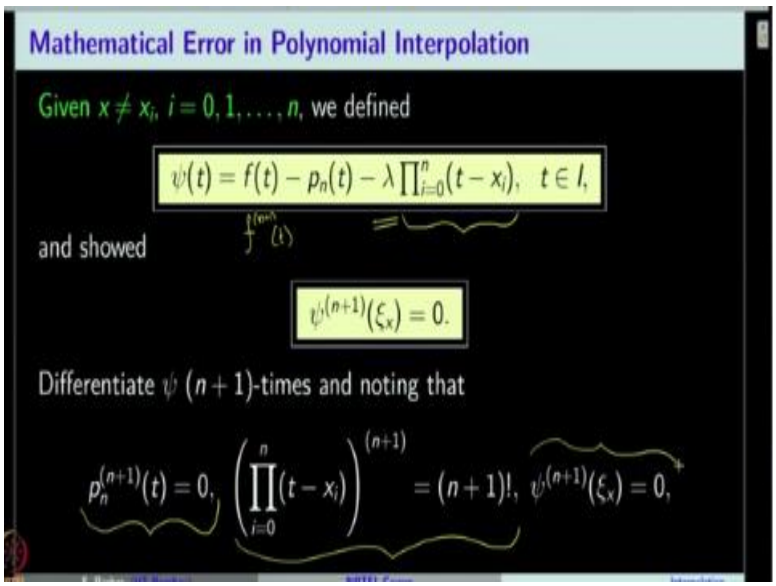
Therefore, how many zeros are there for the function ψ, you can see that there are n + 1 zeros coming from the nodes, plus one zero that is coming from the point x that we have chosen arbitrarily to derive the mathematical error. So, there are totally n + 2 distinct nodes. Now you have to go back to our mathematical preliminaries chapters, exercise problems. There is an exercise problem which says that a function ψ has n + 1 zeros, then psi dash will have at least n zeros, at least that is very important.

Why is it so? It can come from Rolle's theorem. So, if you have two zeros say x_1 and x_2. Then Rolle's theorem says that there is at least one point in between these two zeros x_1 and x_2 let us call this ξ_1 where $\psi'(\xi_1)$ = 0. Therefore, if you have two point zeros then you have at least one 0 in between that.

Suppose you have 3 zeros then you have at least 2 points at which ψ' will vanish. So, in general if you have n + 2 distinct zeros then you have at least n + 1 distinct zeros in the interval (a, b). Now you apply the same idea to ψ' and try to see how many zeros are there in ψ'', ψ'' will have at least n distinct zeros. Similarly, ψ''' will have at least n -1 distinct zeros and so on.

At the end, you can see by applying the Rolle's theorem repeatedly $n + 1$ times on the function ψ, we can conclude that the $n + 1$ derivative of the function ψ has at least 1 root. So, that is what you can conclude from Rolle's theorem, let us call this root as ξ_x. Why this suffix *x*, because ψ depends on *x* as a parameter. Therefore, by changing this *x* your definition of ψ will change and therefore the zero of the function $\psi^{(n+1)}$ will also change. To denote that this ξ_x is going to depend on *x*, we have used this notation.

(Refer Slide Time: 17:53)



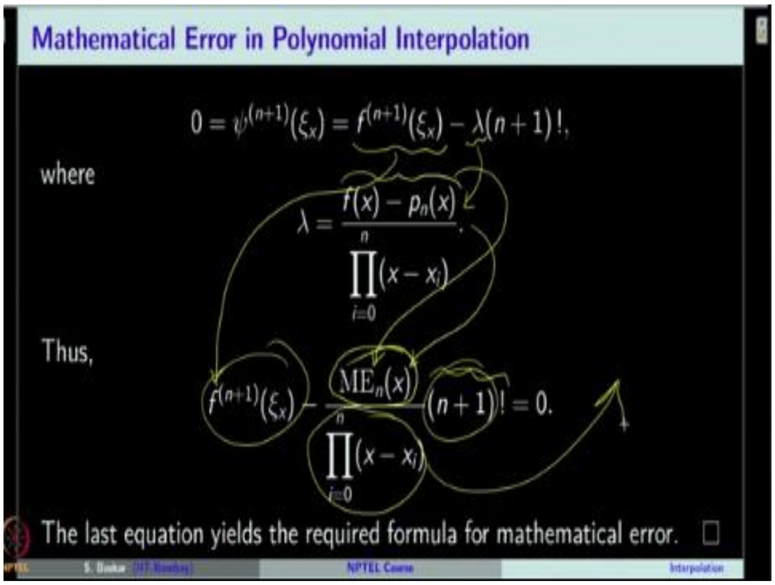
Let us summarize, we have chosen a point *x* arbitrarily in the interval (a, b) in such a way that it does not coincide with any of the node points. And then we defined this magic function $\psi(t)$ and we have chosen this λ in such a way that ψ has at least $n + 2$ roots. So, once we have this result by applying Rolle's theorem repeatedly, we see that the $n + 1$ derivative of ψ has at least one zero in the interval (a, b) and we have denoted that zero by ξ_x.

This is what we have done so far. Now what is the next step? We have this expression, we will differentiate this function $n + 1$ times we will get an expression out of that. Substitute $t = \xi_x$ in that expression and equate it to 0 and let us see what is happening. So, we will go to differentiate this function $n + 1$ times with respect to *t*, note that in this differentiation, *x* is considered to be a constant because it is used only as a parameter in ψ.

So, it will not be a variable, it is treated as a constant and we are differentiating ψ only with respect to t. You can see that in the first term we will get $f^{(n+1)}(t)$ and in the second term when you go to differentiate the polynomial $p_n(t)$. Remember p_n is a polynomial of degree less than or equal to n. Therefore, its $n+1$ derivative is equal to 0. Then we are not going to differentiate λ because it is independent of t.

Then we will have to differentiate this product and that will give us $(n+1)!$. Now after all this we have to keep in mind that $\psi^{(n+1)}(\xi_x) = 0$.

(Refer Slide Time: 20:27)



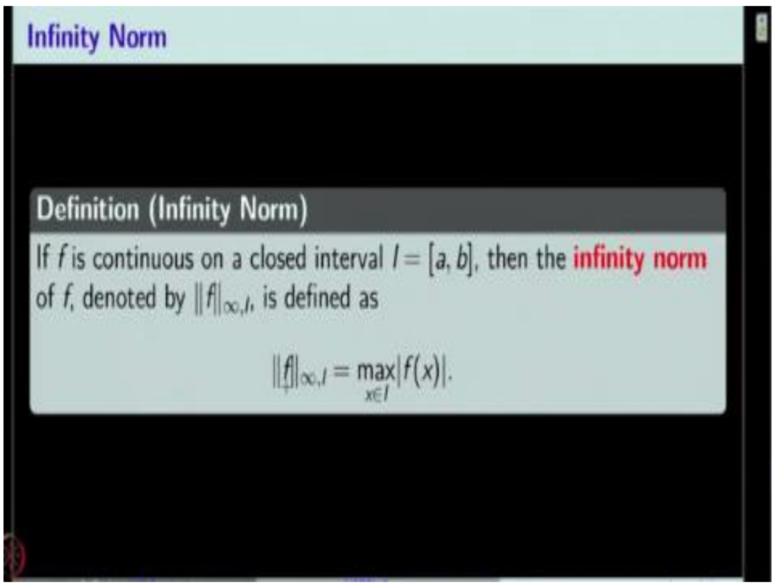
Now let us put all this into this expression and see what happens, $0 = \psi^{(n+1)}(\xi_x)$, on the right-hand side we have differentiated this expression and that is given by this and we have substituted $t = \xi_x$ in that. What is λ? λ is given by this expression; you can see that this is precisely the definition of the mathematical error that we want to understand. So, that is what is sitting here and this is nicely sitting in this expression.

Let us plug in that and see we have $f^{(n+1)}(\xi_x)$. That is kept as it is. Instead of λ, I am putting this expression and then the $(n+1)!$ is kept as it is. Now you see instead of this, I am just putting the notation of the mathematical error that is all. Now you have this expression where you can now keep $ME_n(x)$ on the left hand side take all the other terms on the right hand side. And you can go back to the statement of the theorem.

And see that you will get the precise expression for the mathematical error by doing this. And that completes the proof of our theorem. We got a nice expression for the mathematical error however it cannot be used to quantify the mathematical error exactly. Why? Because this part is still unknown to us. However, one can use this expression to obtain some estimates on mathematical error. Such an estimate will depend on the problem that we are working on here.

Here we will now try to consider a few simple examples through which we will try to get an idea of how to estimate the mathematical error using the expression that we have derived.

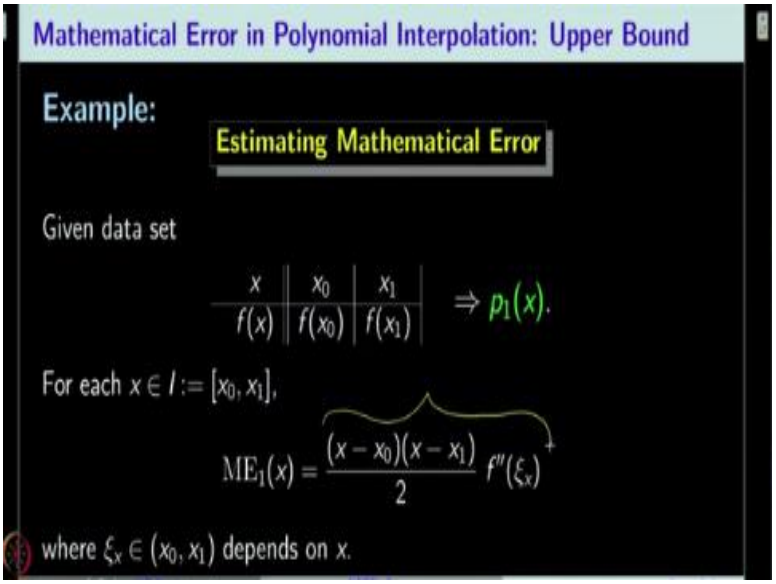
(Refer Slide Time: 22:42)



For this we need a notation called infinite norm, this is not something new to us. If you recall in the matrix norms, we have come across such a concept l_∞ norm. The same idea is here also. But now we are applying this idea to define norm for a function. So, let us consider a continuous function on a closed interval $[a, b]$. Why am I considering a continuous function? Because we know that a continuous function on a closed bounded interval is bounded and therefore it attains its maximum in the interval.

And now the infinite norm of f, which is denoted by this notation, is nothing but you take the absolute value of f evaluated at each point in the interval $[a, b]$ and then take the maximum. That maximum is what we will call an infinite norm, this is the notation. Therefore, wherever I put this notation you have to understand that I am taking the maximum over the modulus of $f(x)$ in the interval $[a, b]$.

(Refer Slide Time: 23:57)



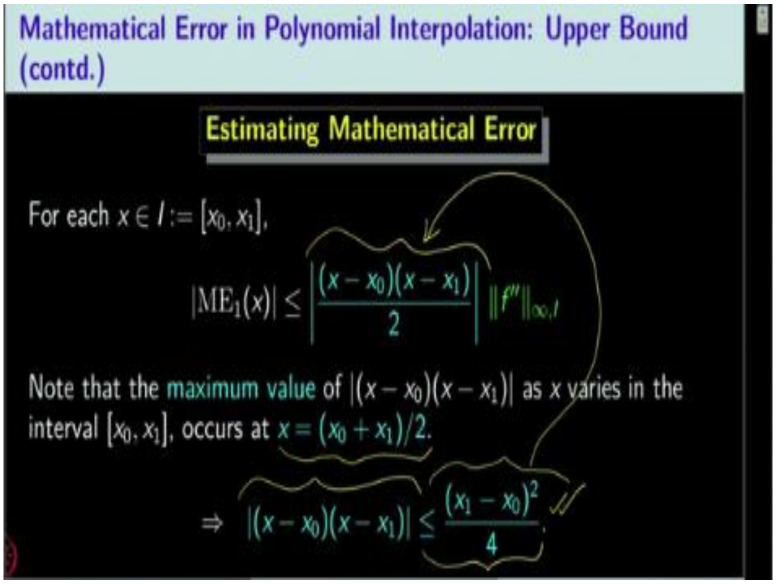
With that in mind, let us see two examples where we estimate the mathematical error using the expression that we derived in the previous theorem. As a first example let us consider the most simplest polynomial that is the linear interpolating polynomial of a function *f*. We are taking two node points x_0 and x_1. From there you can generate the linear interpolating polynomial $p_1(x)$. Now the question is what is the mathematical error involved in it?

That is nothing but $f(x) - p_1(x)$. And we now have an expression for which you go back to the previous theorem and see how to write the mathematical error expression involved in $p_1(x)$, that is given by this expression where ξ_x is an unknown lying between the nodes x_0 and x_1. Now if you want to find an estimate for this error, the first step is to take modulus on both sides and then you just split this modulus.

Now you see you have two problems involved in it. One is you have to eliminate or dominate this variable *x*. Remember now *x* is the variable and also this ξ_x which is the unknown depending on the variable *x*. All this has to be dominated and we should get an upper bound where the upper bound is a fixed number which can be computed with our known information. We will assume that f'' is known to us.

With that in mind, we will do or at least we will know the upper bound of f'', that is what we will assume to derive this estimate. Now the first step is to take this function f'' by our assumption f'' is a C^2 function and therefore f'' is a bounded function in the interval $[x_0, x_1]$. And that shows that its maximum is achieved in the interval $[x_0, x_1]$. So, you can replace this term by its maximum.

(Refer Slide Time: 26:47)

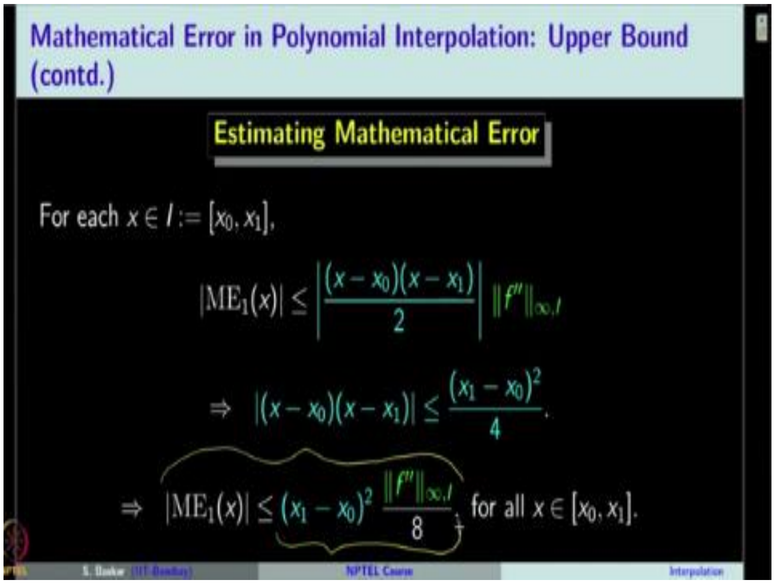


Remember, we have given a notation for that maximum therefore I will use that notation to denote, that I am replacing $f''(\xi_x)$ by its maximum and thereby we are just freezing the unknown ξ_x by putting this known quantity. And for that we have to also replace = symbol by ≤ symbol because we are replacing $|f''|$ by the maximum. That is why this happens. Now we have to only deal with this term.

Now how to deal with this? You can see that it is nothing but the modulus of a quadratic polynomial and you can easily see where the maximum is achieved in the interval $[x_0, x_1]$. You can see that the maximum of this function is achieved at the point $\frac{x_0+x_1}{2}$. That is at the midpoint of the interval $[x_0, x_1]$. So, from here you can see that this function which is sitting here is less than or equal to this quantity.

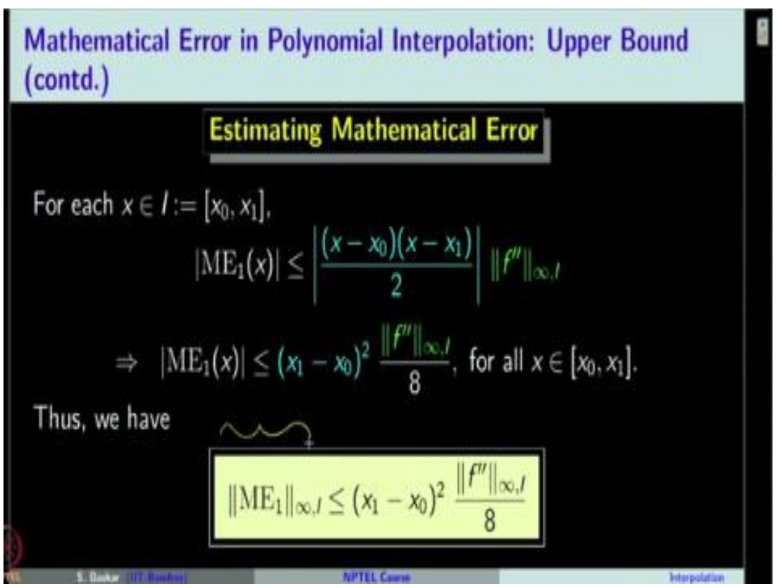
That is, we are evaluating this function at the point where it achieves its maximum. Therefore, this function is less than or equal to this maximum value. So, you can now substitute this value into this expression and thereby you can also eliminate x or dominate this term by its maximum.

(Refer Slide Time: 28:30)



So, that is precisely what we have done here. By this we got an upper bound for the mathematical error at the point x. And this upper bound is independent of x. What does it mean? It means this inequality holds for all x in the interval $[x_0, x_1]$.

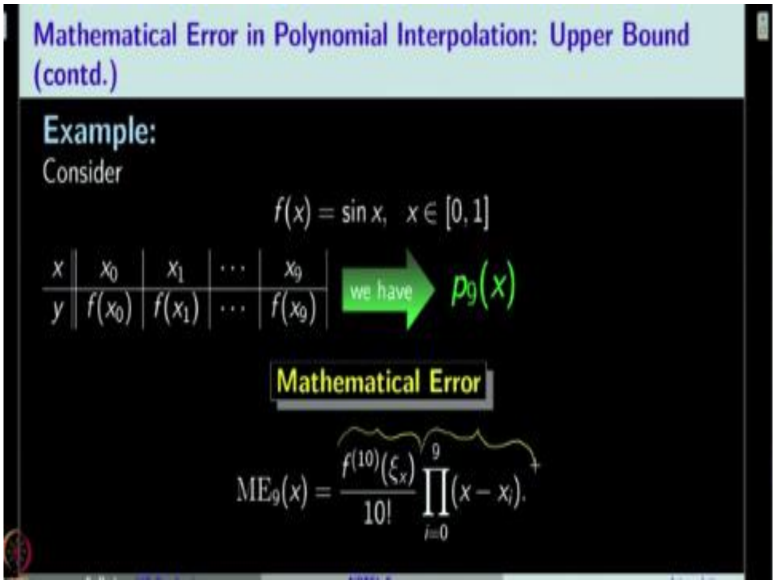
(Refer Slide Time: 28:57)



And that implies that even if you take the maximum norm of this mathematical error that will also satisfy this inequality. Therefore, finally we will consider this as the estimate for the mathematical error. What does it say? It says that whatever may be the mathematical error we do not know that but the worst case of the mathematical error because we have taken the maximum of it.

Therefore, that is the worst mathematical error that we will come across in the linear polynomial interpolation. And that will surely be less than or equal to this number. You can see that $if\ \ f''$ is something known to us or at least its maximum is known to us then the right hand side is fully known to us. So, in that way we obtained an estimate for the mathematical error.

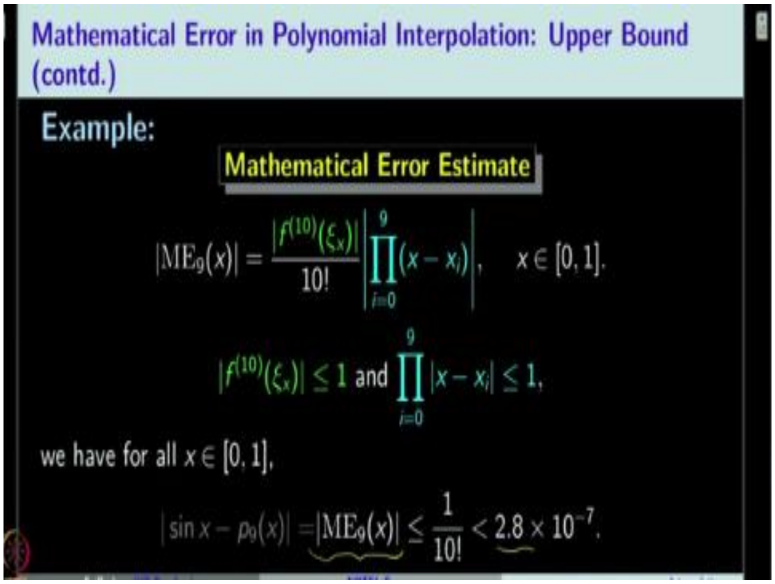
(Refer Slide Time: 29:54)



Let us give a more precise example. In this case we will take the function $f(x) = sin(x)$. And let us restrict ourselves to the interval [0,1]. Now we will consider 10 node points in the interval [0,1]. And the corresponding values of the sin function at these nodes, and thereby we can construct the 9^{th} degree interpolating polynomial for this data set. Now our question is what is the estimate of this mathematical error?

For that we have to first find the upper bound for this function and also, we have to find the upper bound for this part.

(Refer Slide Time: 30:42)

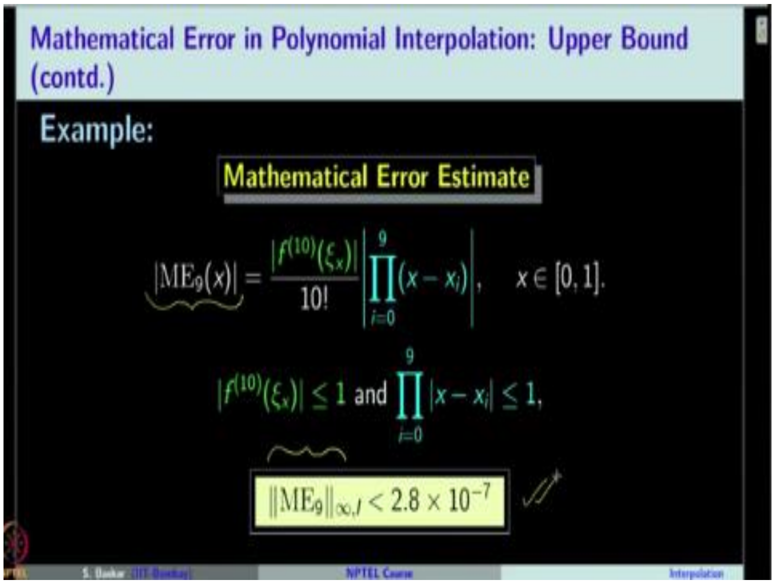


Let us go and take the modulus on both sides, that is the easiest first step that we can do now we have to estimate this term and we have to estimate this term. This term in this particular example is not something difficult. Because it is just $cos\ x$ and we can just put this over estimate for this. Because we know whatever may be x cos actually going to lie between -1 and 1. Therefore we can in fact put this overestimate we can also make a better estimate by taking the fact that x lies between 0 and 1.

Remember x should be taken in radians and you have to compute $cos\ x$ and see where it attains its maximum in the interval $[0,1]$. It happens to be 1 only therefore this is actually not a bad estimate in this particular example. But how to get the upper bound for this function? This is the absolute value of the 10th degree polynomial. If you go to find the maximum of this function that may sound a bit difficult. But what you can do is, a clever observation, that whatever may be x and x_i they both are going to lie in this interval $[0,1]$.

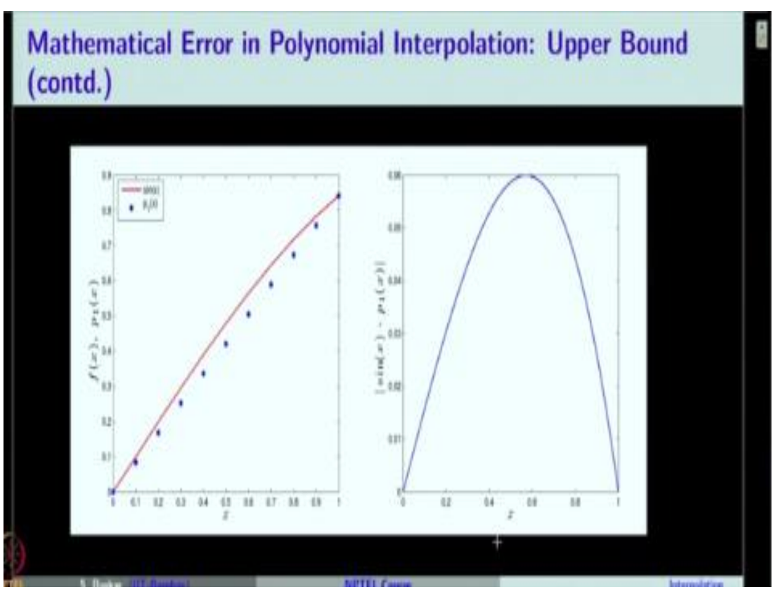
Therefore, the value, at least the absolute value, is going to be less than or equal to 1. Therefore, whatever may be the maximum of this term its maximum is surely not going to exceed 1. Therefore, I can use this over estimate for the second term and thereby I can say that the mathematical error involved in the 9^{th} degree interpolating polynomial of the sin function is surely going to be less than or equal to $\frac{1}{10!}$ which is approximately less than or equal to 2.8×10^{-7}.

(Refer Slide Time: 32:56)



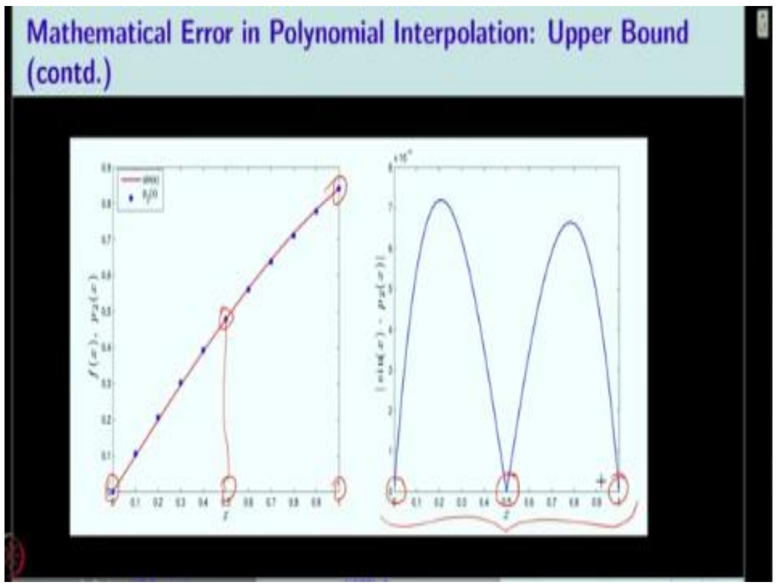
And remember this holds for all x in the interval [0, 1]. Therefore, you can replace $|ME_9(x)|$ by its maximum norm also because it holds for all x and the upper bound is independent of x. Therefore, this will even hold for that x at which ME_9 attains its maximum. Therefore, this is the final estimate that we want to get. Again, you can see that whatever may be the worst-case mathematical error for the 9th degree interpolating polynomial, it is surely going to be less than 2.8×10^{-7}, that is what our analysis says. Now if you think that you can tolerate this much error then you can happily go for the 9th degree polynomial interpolation for your function. So, this is one illustration where you can use the expression for mathematical error to estimate your error.

(Refer Slide Time: 33:58)



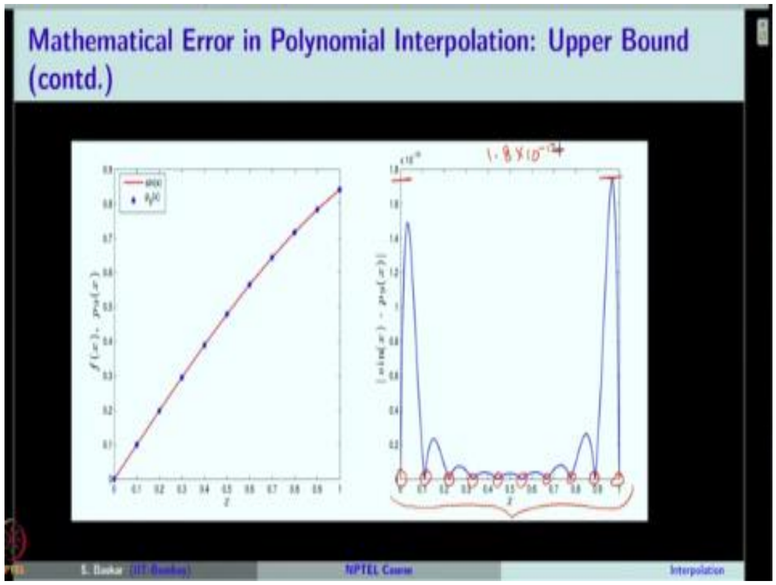
Let us have a visual feeling for the example that we have considered. In this left side graph we have plotted the sin function with a red solid line in the interval [0, 1] and the blue dotted line indicates the linear interpolation polynomial for the function sin x at some points where the nodes are taken as 0 and 1. You can see the error is shown on the right side plot in the blue solid line. Although I have written here, as a mathematical error strictly speaking it is not the mathematical error because we have computed it on a computer but still at the first-degree polynomial level the arithmetic error is negligible. Therefore, we may still see this graph as the graph of mathematical error.

(Refer Slide Time: 35:04)



Let me increase the degree of the interpolating polynomial. Now I am considering the quadratic interpolating polynomial with nodes as 0, 0.5 that is this point and 1 that is this point. You can see that the error is now reduced a little bit. And also, you can see the interesting part is that the error is touching the line $y = 0$ at the node points, why? Because of the interpolation condition.

(Refer Slide Time: 35:40)



Now I am just going to show you the plot of the 9th degree interpolating polynomial for which we have obtained an estimate using the theoretical result. Here I have taken 10 nodes $x_0, x_1, \cdots, x_{10}$. What are all the values of these nodes? You can see the error plot and try to guess what are all the node points that we have taken. Precisely, these are the points where the graph of the error hits the x axis.

And the interesting part is what is the maximum value of this error in the interval $[0, 1]$. It is something very near to 1.8×10^{-12} that is the maximum that we have obtained. In this particular example we can see explicitly what that error is. If you go back to what the theoretical estimate gave us. The theoretical estimate gave us 2.8×10^{-7}.

Therefore, computationally the error that we have attained is much less than the theoretical estimate. And hence our theoretical estimate is working well in this particular example. With this, our discussion on mathematical error is done. Thank you for your attention.