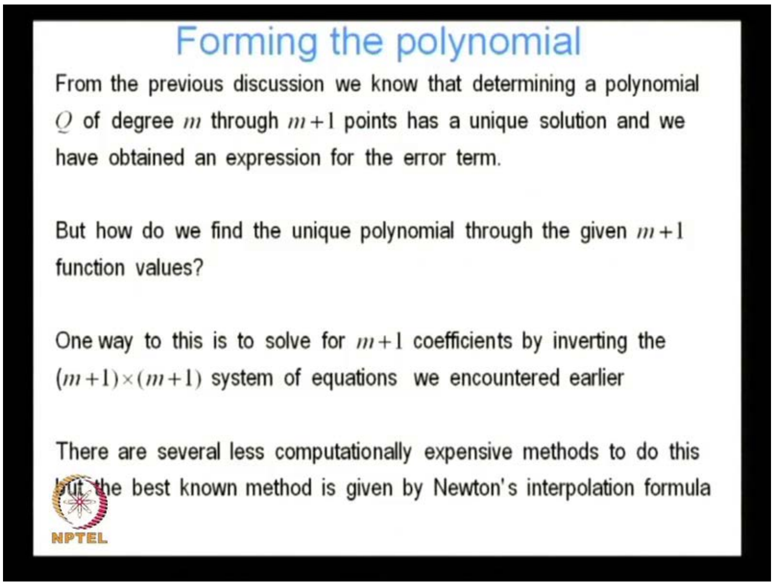
Numerical Methods in Civil Engineering
Prof. Arghya Deb
Department of Civil Engineering
Indian Institute of Technology, Kharagpur

Lecture - 31
Polynomial Fitting

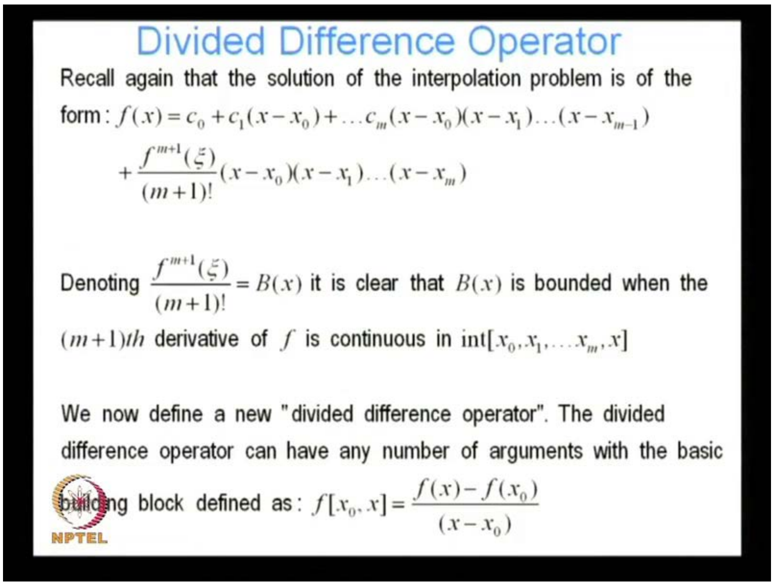
(Refer Slide Time: 00:29)



In lecture 31 of our series on numerical methods in civil engineering, we are going to talk about fitting polynomial through a number of grid points. In our previous discussion, we know that determining a polynomial Q of degree m through m plus 1 point has a unique solution and we have obtained an expression. We also obtained an expression for the error term. We found how to find out the coefficients. We said that you can find if you know m plus 1 point. You can always fit an m th order polynomial through this point by solving a system of equations of size m plus 1, right.

If you see solve that system of equations, you can find out the coefficients, m plus 1 coefficients and if you can find out the m plus 1 coefficients, you have your polynomial, but solving those m by m plus 1 system, every time is of course enormously expensive. One needs to fit a polynomial in any reasonable computation many times, right, many thousands of times, many millions of times. So, if we have to solve that m by m plus 1 by m plus 1 system every time it is not going to be feasible right. So, it turns out that there are simple ways of calculating the coefficients of a polynomial of an m th order polynomial. If you know the function values at m plus 1 point without having to solve that system of m plus 1 by m plus 1 equation, and probably the best known method is given by Newton's interpolation formula, right.

(Refer Slide Time: 02:08)

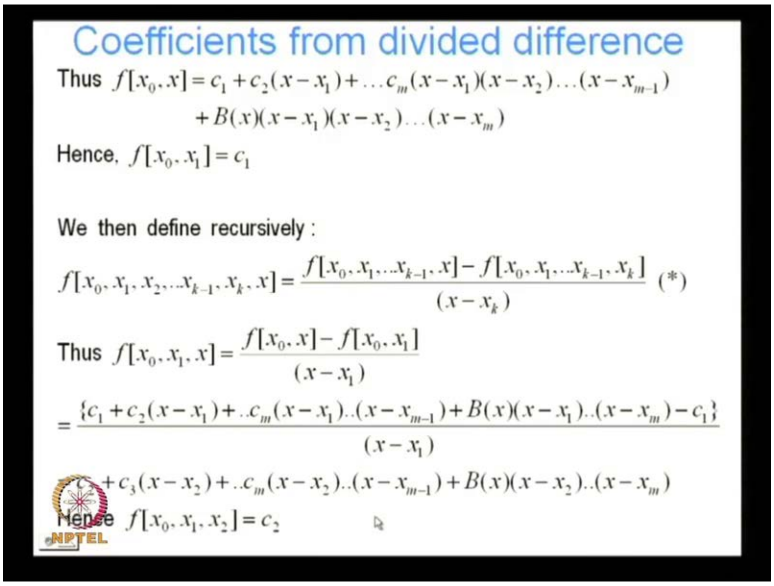


So, we are going to talk about Newton's interpolation formula in somewhat greater detail, but before we do that, we want to talk about something which is known as the divided difference operator, because the Newton's interpolation formula is the best. The coefficients are most easily calculated using something known as the divided difference operator, right. So, before doing that, let us recall that the solution of the interpolation problem is of the following form. So, this is my m-th order expansion, this is my m-th order polynomial and that is my error term, right. That is my error term which I showed last time in considerable detail is of this form, right.

So, if we denote this term by B of x, it is clear that B of x is bounded when m plus one-th derivative of f is continuous in this interval, right. This involves the m plus one-th derivative and if m plus one-th derivative is continuous in that interval, so it has to be bounded, right. Continuous means that the function values cannot become unbounded. It has to grow smoothly, right. It is continuous, right. So, it cannot become unbounded. So, B of x is bounded in that interval. That means that the error term is also bounded, right.

We now define a new divided difference operator. Now, the divided difference operator can have any number of arguments, but the most simple form of the divided difference operator is denoted like this. Divided difference operator of x 0 x, right where it involves the function f is equal to the function evaluated at x minus the function evaluated at x 0 divided by x minus x 0, right. So, this is with just two arguments, but this operator can have any number of arguments mainly depending on the number of grid point, right.

(Refer Slide Time: 04:10)



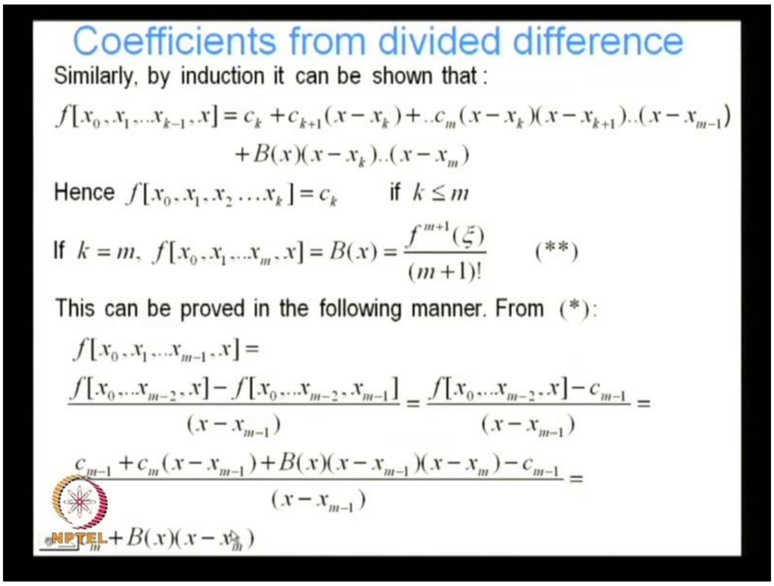
So, that is f of x 0 x is given by this. Why? Well, let us go back and take a look. F of x is given by this, right. So, f of x 0 is equal to c 0, right, f of x 0 is equal to c 0 f of x minus f of x minus f of x 0. Therefore, c 1 x minus x 0 c up to c m x minus x 0, so on plus, this c 0 terms vanishes because f of x 0 is equal to c 0, right. So, that is then I am dividing by x minus x 0. So, I am left with c 1 plus c m x minus x 1 up to x minus x m minus 1 f m plus 1 xi m plus 1 x minus x 1 x minus x m. So, this x minus x 0 term vanishes because I am dividing it by x minus x 0, right. So, I am left with f of x 0 is equal to that, right.

Now, if instead of x as we place x by x 1, what do I get? All the terms vanish except the first term, right. So, that tells me that f of x 0 x 1 is equal to c 1. So, what does this allow me to do? This allows me to express the coefficient c 1 of my polynomial of my m-th order polynomial in terms of the divided difference operator with arguments x 0 and x 1. So, given f if I know f of x 0 x 1, I can find out c 1, right. So, this allows me to express this coefficient in terms of this divided difference operator.

Next, we define in general. In general, we define f, the divided difference operator with k plus 1 with k plus 1 argument from x 0 up to x a k plus 2 arguments actually in this case, right. So, k plus 2 arguments as this, so f of x 0 x 1 up to x i x k minus 1 by x x minus f of x 0 x 1 x k minus 1 x k divided by x minus x k. So, we take this as a matter of definition, right. This is how I divide I define the divided difference operator with that many arguments, right.

So, instead if I have f of x 0 x 1 x, what will I have is f of x 0 x 1 x will be equal to f of x 0 x minus f of x 0 x 1 divided by x minus x 1, right. So, it will be like that, right. Similarly, f of x 0 x 1 x 2 x would be equal to f of x 0 x 1 x minus f of x 0 x 1 x 2 divided by x minus x 2, right. So, this is how I am going to define the generic divided difference operator, right. So, f of x 0 x 1 x is going to be this. I just talked about that, right and f of x 0 x we know is this. We have already calculated that to be this and f of x 0 x 1 is given by that. So, again c 1 c 1 cancels out, right. C 1 cancels out and that c 1 I am dividing by x minus x 1. So, I have c 2 plus c 3 x minus x 2 plus c m x minus x 2 up to this, right. Again if I evaluate this at x 2 f of instead of x, if I evaluate it as x 2, what do I get? Every term vanishes except the first term. So, that allows me to write f of x 0 x 1 x 2 is equal to c 2.

(Refer Slide Time: 08:21)



So, I can write the second coefficient of my m-th order polynomial in terms of the divided difference operator with arguments x 0, x 1 and x 2. Similarly, we can show by induction that f of x 0, x 1, x k minus 1 k x is given by this, right. You can see that this is going to have exactly the same form as this, right. Because of this we can write f of x 0 x 1 x 2 x k.

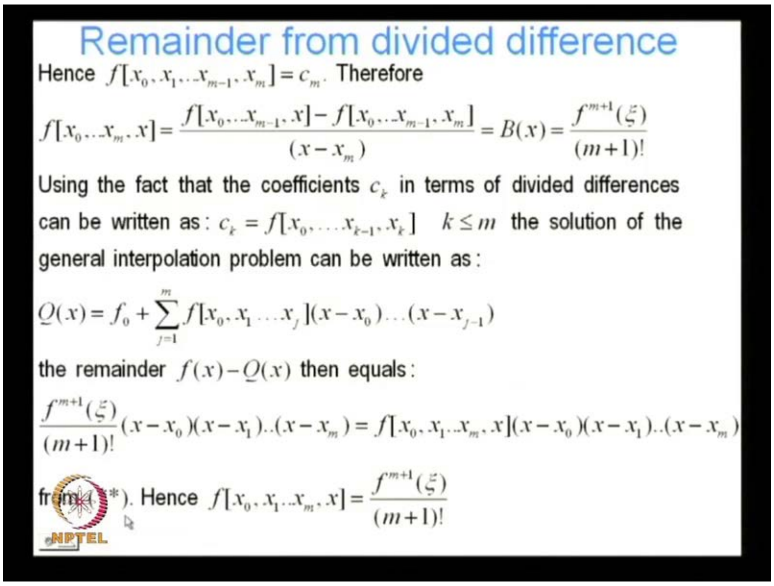
So, again all the terms will vanish except the first term, and that is going to give me c k, right. So, f of x 0, x 1, x 2, x k is equal to c k if k is lesser than or equal to m which is of course understood, right because we are interested in m-th order polynomial, right. So, if k is equal to m, f of x 0, x 1, x m, x is equal to B of x. So, we know that f of x 0, x 1, x 2, x m is equal to c m. That we know from this, right, but if k is equal to m, then x 0, x 1, x m, x is equal to B of x is equal to f m plus 1 xi m plus 1 factorial. That is not so obvious, right.

This is obvious because we have shown it by induction. In every stage we could write the coefficients in terms of the divided difference operator, in terms with the highest term in the divided difference operator has the same index as the coefficient, right. We could find that out, but if I have a divided difference operator which is f x 0, x 1, x m, x, we are going to show, but that is actually equal to my coefficient of my remainder terms, right. Let us go back and take a look at the remainder. Our remainder term was like this, right.

So, all these coefficients have been able to express in terms of the divided difference operator, and I am going to show that the coefficients of the remainder is also given by the divided difference operator with x 0, x 1 up to x m, x, right. So, that is my coefficient of the remainder term. This can be shown like this in the following manner from this expression. From this expression I can write that f of x 0, x 1, x m minus 1 x, right. So, if instead of k I have x m k equal to m minus 1, then I will have f x 0, x 1, x m minus 2 x minus f x 0, x 1, x m minus 2 x m minus 1 divided by x minus x m minus 1, right.

So, that is what I have here, right and this I know is equal to c m minus 1 because it is a x 0 x m minus 2 x m minus 1 that is equal to c m minus 1 that is equal to f x 0, x m minus 2 x and I am dividing by this, right. So, this I know is given by that, right. So, if we look at this expression and I replace k by m k minus 1 by m minus 2, if I replace k minus 1 by m minus 2, then I get this expression, right, c m minus 1 plus c m this minus c m minus 1. So, c m minus 1, c m minus 1 will cancel. I am dividing by x minus x minus 1. So, I am left with c m plus b x x minus x m x x m, right, x m minus 1 that cancel out. So, I am left with this.

(Refer Slide Time: 12:29)



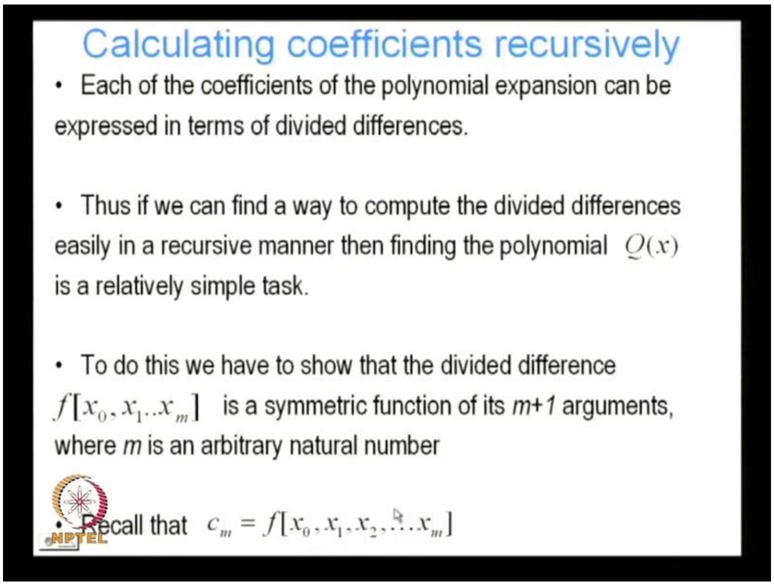
So, hence f of x 0, x 1, x m minus 1 x m is equal to c m. So, here if I replace x by x m, this term vanishes and I am left with c m, right. So, f x 0, x m, x x this is what we wanted to evaluate. If you go back and take a look, this is what we wanted to evaluate and that is equal to f x 0 x m minus 1 x by definition, right. This is how we have defined our divided difference operator x 0 x m minus 1 x minus f x 0 x m minus 1 x m by x minus x m, right.

Now, we know that this thing x 0 x m minus 1 x m is equal to c m, right. We know that that thing is equal to c m plus B x x minus x m, right, this thing is equal to that. So, c m plus B x x m minus x m, this term minus c m is equal to given B x x minus x m x minus x m cancels out. I am left with B x which is my coefficient of the remainder term, right. So, I have been able to show that the coefficient of my remainder term I can write in terms of the divided difference operator like this.

Now, using the fact that coefficient c k in terms of divided difference can be written like this. The solution of the general interpolation problem can therefore be written like this. So, everywhere we will go back to, let us go back to our solution of general interpolation problem which was like this, right. So, every time I am going to replace my coefficients by the appropriate divided difference operator, right. If I do that I can write it like this Q of x is equal to f 0 plus sigma j equal to 1 to m f x 0 x 1 x j, right. So, if j equal to 1, I will have f x 0 x 1, right and this will be from x minus x 0. The second term will be j equal to 2 and that will be f x 0, x 1, x 2 times x minus x 0 into x minus x 1 and so on and so forth, right.

So, that is the first part of my coefficient of my polynomial expansion and the remainder then equals f x minus Q x which is equal to this which we already know which is this. This is the remainder and that is equal to this term and this term from here we can write as f of x 0, x m, x times x minus x 0 x minus x 1 up to x minus x m, right. So, we get this thing is equal to that thing, right which we actually already shown, right. Is that clear?

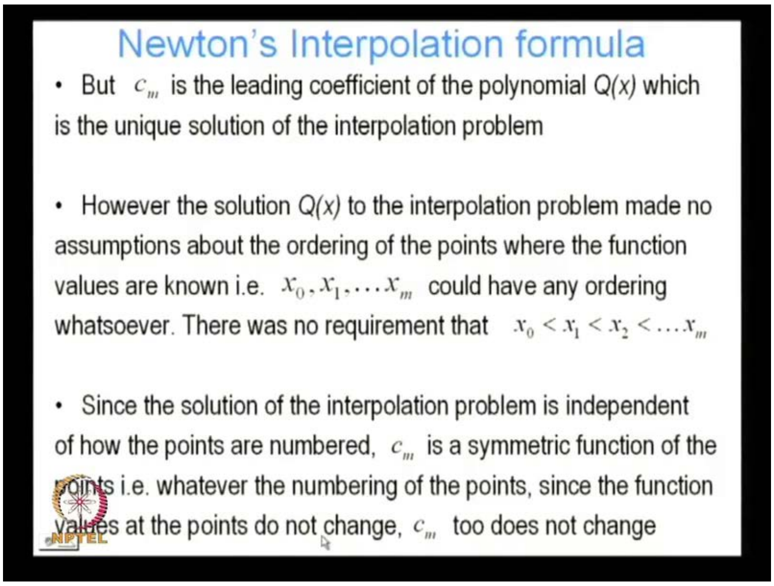
(Refer Slide Time: 15:45)



So, later on we will show that these coefficients, they can be calculated recursively. It is very easy to calculate these coefficients, right, but before we do that let us talk little bit more about this. So, each of the coefficients of the polynomial expansion can be expressed in terms of divided differences. Thus, if we can find a way to compute the divided differences easily in a recursive manner, then finding the polynomial Q x is a relatively easy task, right.

So, given these m m plus 1 point, right and given these m plus 1 function values, it will turn out that I can find out all these coefficients relatively easily using a recursive relationship without having to solve that m plus 1 by m plus 1 equation, right. We will do that using the divided difference operator, but in order to do this, we have to show something before we do that and that is we have to show that the divided difference operator f x 0, x 1 up to x m is the symmetric function of its m plus 1 arguments where m is an arbitrary natural number. What do I mean by a symmetric function? That means, the order in which these x 1 x m appear in this divided difference operator is not important. Why? Recall that this is given by this, c m is given by this, right.

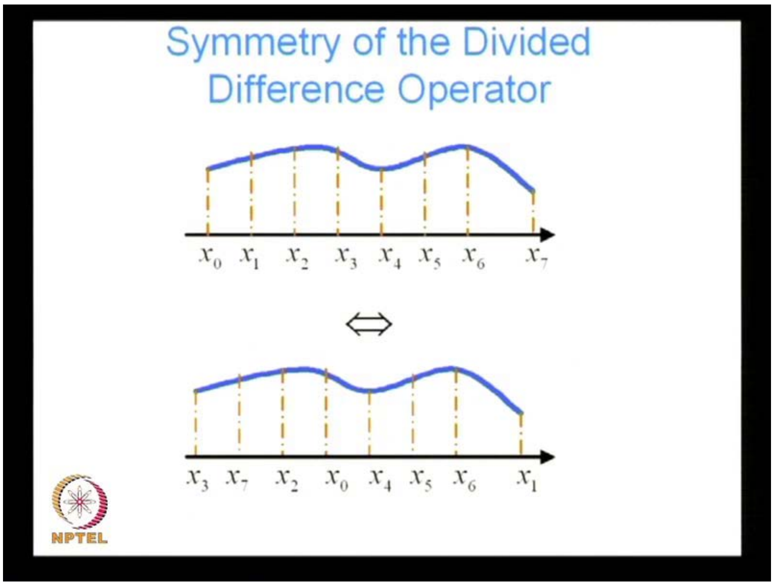
(Refer Slide Time: 17:26)



But c m is the leading coefficient of the polynomial Q x which is the unique solution of the interpolation problem. It is the leading coefficient because it is a coefficient of the largest term of the highest order, right. It is the leading coefficient of the polynomial. However, the solution Q x to the interpolation problem made no assumptions about the ordering of the points where the function values are known. We just said that the function values are known at x 0, x 1, x m plus 1, x m, x 0, x 1, x 2 up to x m, right. We did not say anything about whether how those points are ordered, whether x 1 is greater than x 0 or x 0 is less than x 1. We said nothing about that, right.

So, these points can be ordered arbitrarily, right. So, as long as the function values at those points are known, it does not matter. Their order does not matter. So, similarly their order in the divided difference operator should not matter since the solution of the interpolation problem is independent of how the points are numbered, right. I can call a point x 1, I can call it x m, I can do anything, right it does not matter. So, long at that point, the function value is unique, right. So, I can call a point x 1 as the function value remains the same when I call it x m, right. That does not matter. The polynomial is not going to change because I change the name of the grid point, right.

(Refer Slide Time: 19:04)

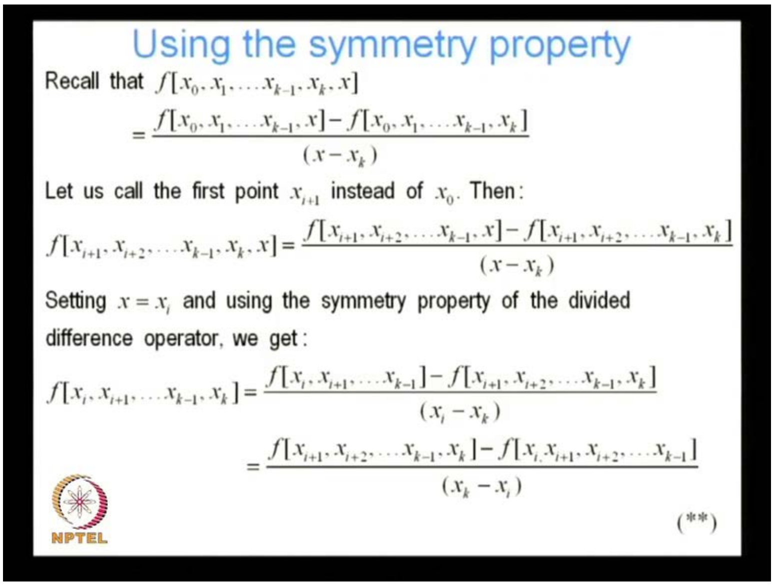


So, I just showed a little picture here. Suppose, I fit a polynomial like this where I initially had the points ordered like this. Now, if instead of calling this point x 0, I call this x 3 and instead of calling this x 1, I call this x 7, nothing changes. I still get the same polynomial because the function value at this point is the same. This function value is the same here, this is the same here, this is the same here.

So, ordering the index by which I refer to those points is not important. Since, the solution of the interpolation problem is therefore independent of how the points are numbered, right. Same is the symmetric functions of the points, that is whatever the ordering of the points, whatever the numbering of the points, since the function values at the points do not change, c m does not change, right. So, what does this mean? So, c m is equal to f x 0, x 1, x 2, x m, right, but c m is also equal to f of x 2, x 0, x 1, x m, right.

So, however, I order those points, the divided difference operator. My coefficients are not going to change because what is c m? C m is the coefficient of the term with the highest power in my polynomial expansion. My polynomial expansion is unique. That means the coefficient of the term with the highest power is also unique, right. It cannot change, right. It should not change with how I call the points, right. So, is that clear? So, that is very important, right c m is the symmetric function of the divided difference operators, right. C m is its coefficients are a symmetric function of divided difference operators.

(Refer Slide Time: 21:06)

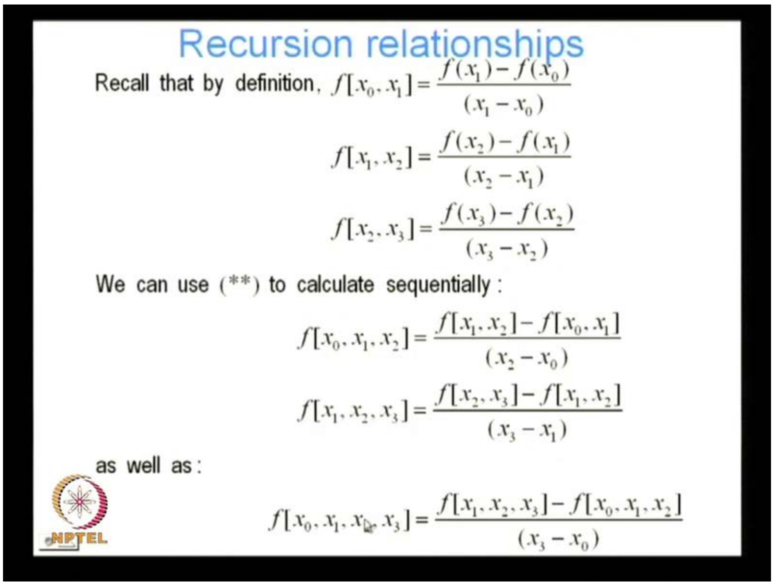


So, let us recall again that f of x 0, x 1, x k minus 1 x k, x is given by this definition. Now, let us call the first point x i plus 1 instead of x 0, right. I can call it anything. So, as long as the function value at that point does not change, my polynomial is not going to change. Since, my polynomial is unique; my coefficients are unique, right. So, instead of calling it x 0, I can call it x i plus 1. In that case what do I have here? I have x i plus 1, x i plus 2 and so on up to x k minus 1 x k x and that is equal to again instead of 0, I have i plus 1, i plus 2, x k minus 1 x.

Here, I have again x i plus 1, x i plus 2, x k minus 1, x k and I am dividing by x minus x k, then I said x is equal to x I, right and then I use the symmetric property of the divided difference operator. Because of the symmetric property f x i plus 1, x i plus 2, x k minus 1, x k x, I have replaced x by x i. I can move the x i in front, right. I can move this x i right here and I get f of x i, x i plus 1, x k minus 1, x k, right. So, this if I replace x by x I, I have i plus 1, i plus 2, k minus 1, x k x i. Then I use the symmetric properties, so that I can interchange the arguments. Any way I like interchange. I put x i plus 1, right in front, right. So, x i right in front.

So, I have x i, x i plus 1, x k minus 1, x k. Similarly, here I replace x by x i and move the x i right in front, right. So, I have x i, x i plus 1, x k minus 1, right and this I leave and change x i plus 1. Here there is no x. So, I did not replace it with x i. So, I have f x i plus 1, x i plus 2, x k minus 1, x k, right and then I just changed, interchange this. So, I move this here and I move this there. That gives me a negative sign, right. So, I get x k minus x i to cancel the negative sign, interchange the order of these arguments, right. So, I move this here, move this there and I get that, right. So, now what do I have? I have x i, x i plus 1, x k minus 1, x k is equal to f x i plus 1, x i plus 2, x k minus 1, x k minus this divided by x k minus 1, x k minus x i. It turns out that this is what we need. This yields the recursion formula. Why? Let us take a look.

(Refer Slide Time: 24:12)



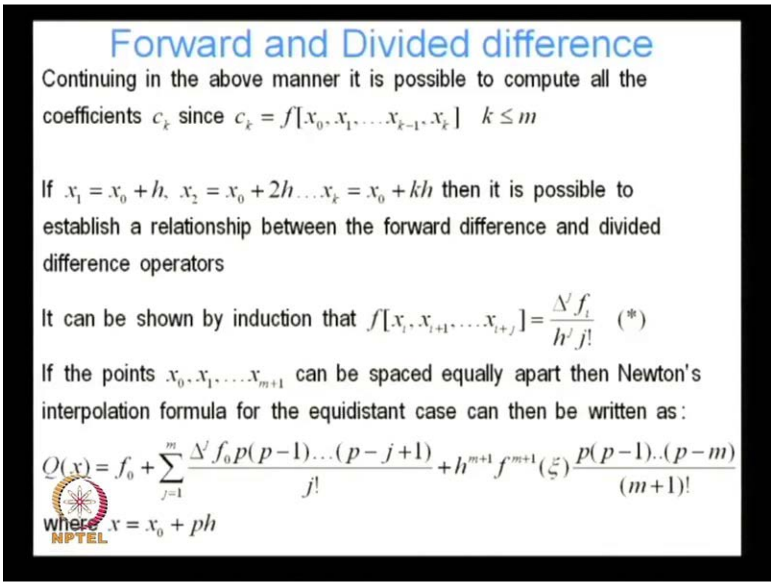
So, we know by definition f of x 0, x 1 is equal to f of x 1 minus f of x 0, x 1 minus x 0. Similarly, f of x 1, x 2, I know f of x 2 minus f of x1. So, all these I can evaluate by definition, right. This I know I can. So, if I know the function values at x 0, x 1 and x 2, I can evaluate these divided difference formulas. The second order divided differences, right. Now, I am going to use this relationship to evaluate the higher order divided differences. Suppose, I want to find out f of x 0, x 1, x 2, right. Let me go back and take a look at my recursion formula. So, I have f of x 0, x 1, x 2. So, how am I going to do that? I will have f of x 0 up to x 1, right, x 2 right. So, that is k minus 1.

So, if f of x 0, x 1 minus f of x 1, x 0, x 1, x 2, the next term x 1 up to x 2, right. So, I have f of x 0, x 1 minus f of x 1, x 2 divided by this thing, divided by that, right x 2 minus x x 0 or is that right. Let us see, x 0 minus x 2 that is x 0 minus x 2. So, what do I get? So, I can evaluate f of x 0, x 1, x 2. If I know these terms and these I already know from definition, right and then I use my recursion formula to find out the third order divided difference formula, right.

Similarly, I can find out f of x 1, x 2, x 3. How am I going to get that? So, I have x 1, x 2, x 3. So, that will be f of x 1, x 2 minus f of x 2, x 3 divided by x 1 minus x 3, right. So, that is x 3 minus x 1, right. So, that is exactly what I am going to get, right. So, now, if I know these, f of x 0, x 1, x 2, f of x 1, x 2, x 3, then I can find out f of x 0, x 1, x 2, x 3. So, remember my coefficients are c 1 is equal to f of x 0, x 1, c 2 is equal to f of x 0, x 1, x 2, c 3 is equal to f of x 0, x 1, x 2, x 3.

So, I can find these coefficients really simply by using these divided difference formulas, right. So, I do not need to solve my m by m plus 1 by m plus 1 system. I can find out given any number of points, right. I can find out the highest degree of polynomial to pass through those points. I can find out the coefficients using this little table, right and if you code it in a computer, this can be done in fraction of a second, right.

(Refer Slide Time: 27:36)



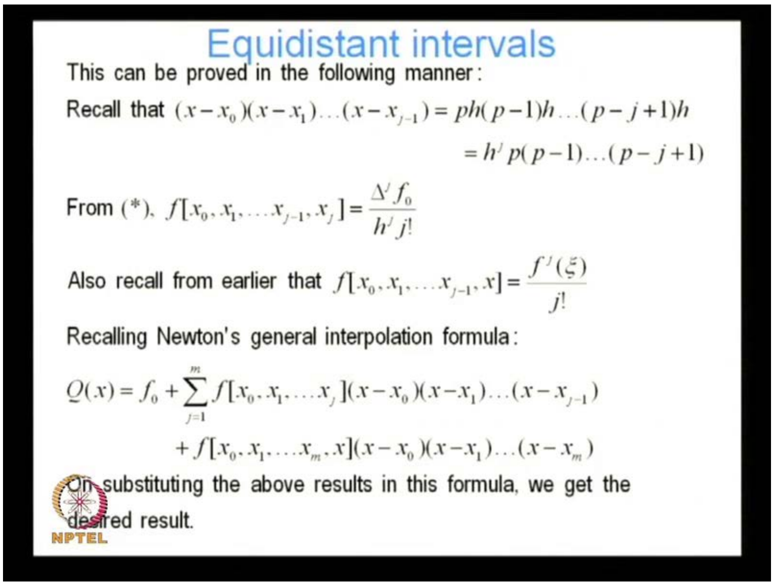
So, continuing in the above manner, it is possible to compute all the coefficients c k. Since, we know c k is equal to this, right c k is given by that for k lesser than or equal to m, right. So, up till this point, we have not said anything about how my grid points are located in space, right. It is not necessary up to from whatever we have done up till now. Assume that those grid points are equally spaced, right. It is not necessary, right, but now if you make further simplification, if you assume that my grid points are equally spaced, right that is if x 0 is my initial point, x 1 is equal to x 0 plus h, x 2 is equal to x 0 plus 2 h and x 3 is equal to x 0 plus 3 h, x k is equal to x 0 plus k h, then it is possible to establish a relation between the divided difference operator and the forward difference operator that we encountered earlier, right. It is possible to do that, right. I have not shown that, but it can be shown that f of x i plus 1, x i plus j is nothing but divided by the forward difference operator operating on f a f i j times, right.

So, if I start with the forward function value at i, right and if I operate with that on that with the forward difference operator j times, I divide it by h to the power j by factorial j, then I get my divided difference operator, right. It sort of makes sense, right because here you can think of it that first if I have f i, right I operate on that with the forward difference operator. What do I get? I get f i plus 1 minus f i. I operate it again. I do it j times. I get this thing, right. It can be shown, right. It is best shown by induction, right. So, I would show that it can easily be shown for j equal to 1, right and then I can show it for j equal to 2 and then I assume that it is true for j equal to k and then I can show that if it is true for j equal to k, it is always going to be true for k plus 1, right.

So, then that can be proved. So, if the points x 0, x 1, x m plus 1 can be spaced equally apart, then in that case Newton’s interpolation formula for the equidistant case can be written like this in terms of my forward difference operator taking advantage of this equivalence between the forward difference operator, and between the forward difference operator and the divided difference operator, I can write my Newton’s interpolation formula in this form where it involves the forward difference operator instead of the divided difference operator.

Earlier, we showed it for the divided difference operator for the most general case, right where there was no restriction on how the points were on the x axis, right. There was no restriction. The intervals could be anything, large, small, equal, unequal anything, but then we said that if those intervals on the x axis are equal, then instead of having to use the divided difference operator, you can write Newton’s interpolation formula using the forward difference operator like that, right and I am going to show that in a relatively simple way, right.

(Refer Slide Time: 31:34)

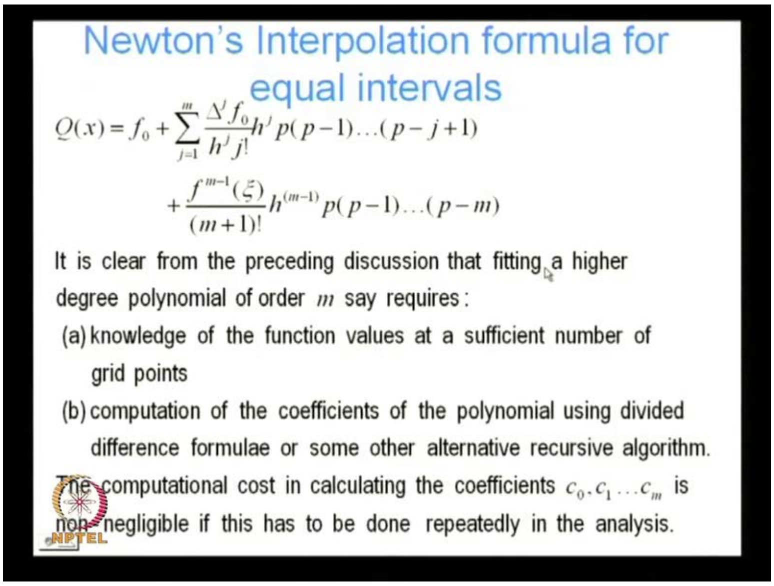


So, we recall that x minus x 0, x minus x 1, x minus x j minus 1 can be written like that. Why can it be written like that? Because x is equal to x 0 plus p h, right. I just said that x is equal to x 0 plus p h, right. Since, x is equal to x 0 plus c p h, x minus x 0 is equal to p h, right, p h x minus x 1 is equal to x x 0 plus p h minus x 0 plus p h plus h, right. So, that is going to give me p minus 1 h and so on and so forth. So, this I can write in terms of powers of h. I can bring out all h's and together I can write it in powers of h to the power j p times p minus 1 up to p minus j plus 1 and we know that f of x, x 0. Let us see f of x 0, x 1, x j minus 1 x j, right and that can be written like this.

So, since i plus 1, i plus j is that, so that is nothing but delta j f 0 h to the power j by factorial j, right, delta j f 0 h to the power j by factorial j, right. I have just used that formula where instead of using i, I have replaced i by 0 and then I can write f of x 0, x 1, x j minus 1, x j is equal to delta j f 0 h j by factorial j, right and we also recall from earlier that this expression f of x 0, x 1, x j minus 1, x is equal to f j x i by factorial j. Where did we see? Well, let us go and see. I do not remember, but I think let us go back right here, right.

This is where we showed that, right. So, we showed that to be true. Now, let us go back to Newton's general interpolation formula that is written terms of the divided difference operator, that is Q x is equal to f 0 plus sigma j equal to 1. This we have already seen in terms of the divided difference operator and this on substituting this here, this term with this, right and this term with that, we get the desired result which is equal to that which shows that in case you have equidistant grid points, you can write Newton's interpolation formula in terms of the forward differences, right.

(Refer Slide Time: 34:52)

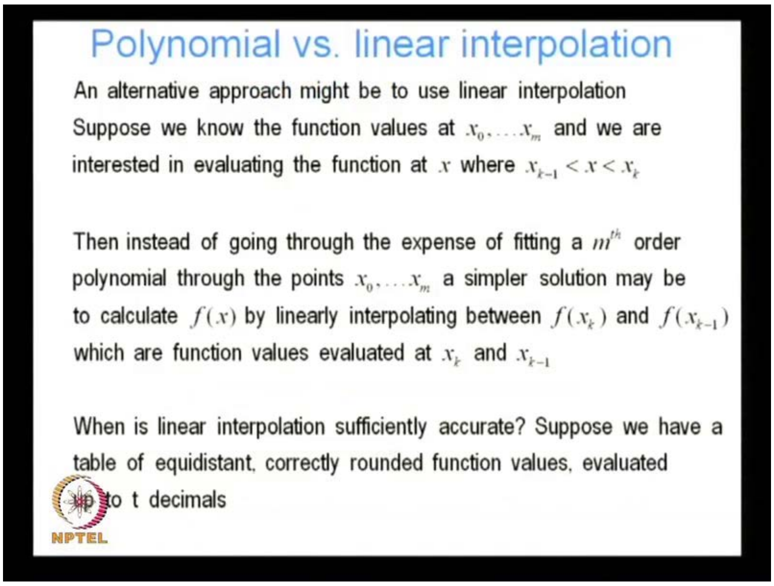


So, it is clear that fitting a higher degree polynomial of order, m requires a knowledge of the function values at a sufficient number of grid points, b, computation of the coefficients of the polynomial using divided difference recursive form of the divided difference algorithm or some other recursive algorithm. There are other recursive algorithms as well, but I am just showing one particular one, right. However, even if we use the divided difference operator to compute the coefficients, if we do it several times, if we have to do this over and over again, it is still non-negligible expense because every time we have these many points, you know the function values to compute it is much cheaper than inverting the matrix. That is for sure, but still the cost is non-negligible, right.

So, somebody might say that well, let us step back and take a look. So, I have m plus 1 points and I have m plus 1 function values, right. Why am I fitting an m-th order polynomial at that point? I am fitting an m-th order polynomial at that point because I want to find the function value at any arbitrary point in the grid, right. So, I have my function values at the grid points, but if I can fit my m-th order polynomial through those function values, I can find the value of the function at any point, not necessarily at a grid point which is not necessarily a grid point. That is what I want to do, right and in order to do that, in order to fit that polynomial, I have to calculate all these coefficients and I found out relatively cheap way to calculate these coefficients, but why somebody might say why do I need to go. Why do at all need to fit a polynomial through these points, right.

If I know the point, if I have m plus 1 grid points and I want to find the function value at a particular point x which does not lie on the grid, if I can figure out between which two grid points that x lies, why do not I just do a linear interpolation. I know the function values at those two grid points, why do not I fit a line through grid points and why do not I get the function value at that point, right. Well, there it turns out that there are advantages as well as disadvantages. It is a lot cheaper fitting line through two points is much cheaper than fitting a polynomial through m plus m-th order polynomial through m plus 1 points, right, but there are problems as well.

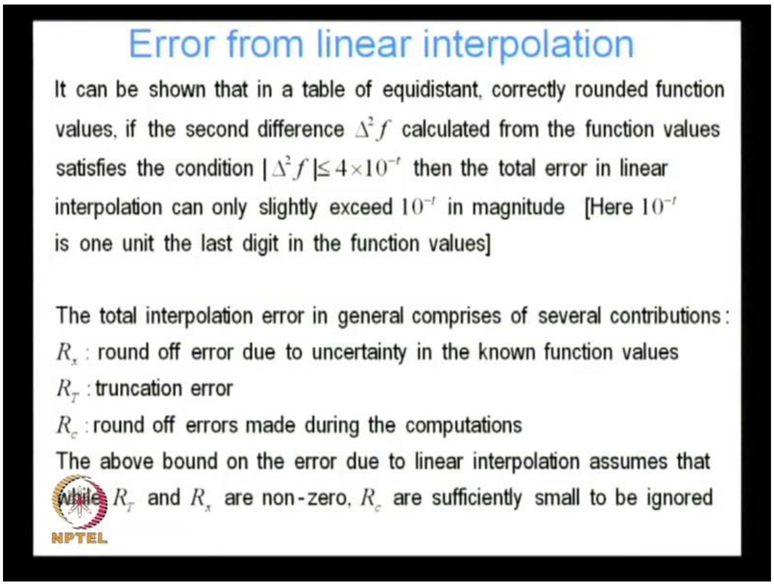
(Refer Slide Time: 38:09)



An alternative approach might be to use linear interpolation. Suppose we know the function values at x 0 through x m and we are interested in evaluating the function at x, where we know that x lies between x k minus 1 and x k, then instead of going through the expensive fitting and m-th order polynomial through the points x 0 through x m, a simpler solution might be to calculate f of x by linearly interpolating between f of x k, and f of x k minus 1 which are function values evaluated at x k and x k minus 1. When is linear interpolation sufficiently accurate? That is a question, right. When is linear interpolation going to give me a good estimate? I know that linear interpolation is going to be cheap, but is there any condition which has to be fulfilled for linear interpolation to give me good results? Well, let us see.

Suppose we have a table of equidistant correctly rounded function values evaluated up to decimals. Suppose I have a grid, equally spaced grid and suppose I know the function values that all those grids points up to t decimal places, right and it is rounded off accurately up to t decimal places.

(Refer Slide Time: 39:36)



It can be shown that in a table of equidistant correctly rounded function values like of table of values m plus 1 values, right and the m plus 1 grid points, m plus 1 function values. If the second derivative, if the second difference, second forward difference del square f calculated from the function values satisfies this criteria, how am I going to calculate the second difference?

Well, I know the x 0, x 1, x 2 up to x m plus x m, right. So, I know the function value at f 0, f 1, f 2, f m, right. I know the function values. I will calculate the first difference f 1 minus f 0, f 2 minus f 1, f 3 minus f 2 and so on and so forth. That is how I am going to calculate the first difference. Then I am going to calculate, I am going to take the difference on that again. So, I can generate the second difference and if for a particular table of function values, if it turns out that all the second differences are less than 4 into

10 to the power minus t, then the total error in linear interpolation is going to be almost equal to 10 to the power minus t.

So, what are the criteria? The criteria is that I have a table of function values and then I calculate the second difference and if the absolute magnitude of the largest second difference is less than the precision level of my calculation times 4, then my total error in linear interpolation is almost going to be equal to my precision, right. It is going to be equal to my precision. Even here we can realise that this condition is going to be relatively very hard to fulfil when in the second I have some function values, right. I have equally spaced grid. I know the values of the functions at all those grid points. I take the first difference, I take the second difference.

When is the second difference going to be really small? The second difference is going to be small when my grid is small, when the variation is relatively small between, when my function does not change much, right. So, if I have very wide wild variations, then there is no guarantee that this thing is going to be small, right. So, either my interval is very small or my function is very well behaved, right. Either of those conditions has to be fulfilled, right. Very well behaved means it is almost very close to a straight line, right. So, both these conditions if I have, interval size is very small. Automatically my function will become closer to a straight line, right or if my function is very well behaved such that these differences becomes smaller and smaller, right.

So, ultimately when is the difference going to be 0? The difference is going to be 0 when I get a straight line or when I get a constant, right. If I get a constant, the difference is going to be 0. So, if I start with a function which is quadratic, if I take the first difference, I am going to get a straight line. If I take the second difference, I am going to get a constant, right. So, if I start with a straight line, I get the first difference. That is going to be a constant. The second difference is probably going to be 0, right.

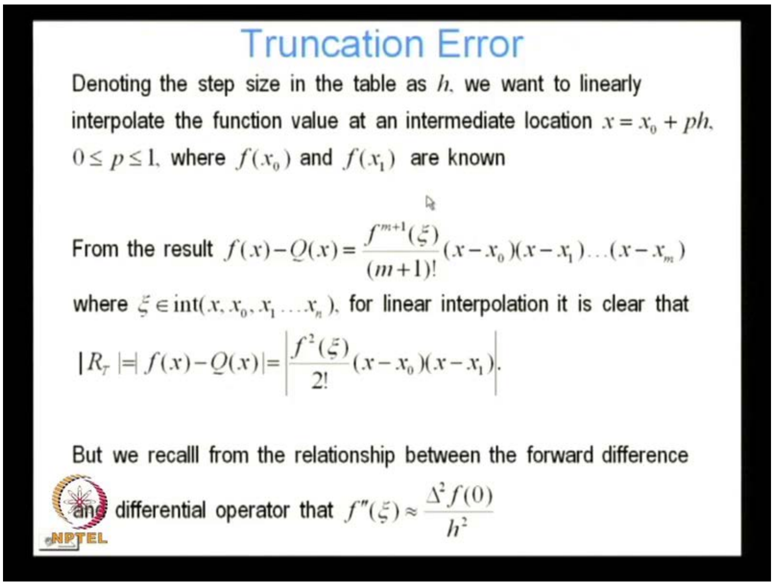
So, this you can see that if the function is very different from a straight line or by interval is sufficiently small, so even if the function is nowhere near linear in that little region, it behaves like a straight line only, then this condition is going to be fulfilled. Only then that condition is going to be fulfilled, but let us try to show this.

How are we going to show that? Well, I am going to show that by dividing my total interpolation error into three parts, right. The first part is the interpolation error due to round of error, due to uncertainty in the known function values. So, I know the function values at those m plus 1 point, but I do not know them exactly, right. I know them only up to the precision of my calculations, right.

So, I know them up to the 10 up to the t-th digit, up to the t-th decimal place, right. So, my function value is there. There is an error due to round off. That is true. Then there is the truncation error. Why is there a truncation error? Because approximating my function by a m-th order polynomial if the function is a higher order polynomial, if it is an exponential function, if it is a hyperbolic function, then my m-th whatever be my m, I am not going to approximate that function exactly, right. So, there is always going to be a truncation error, right and then there are other rounds of errors which I make during the computations, right.

So, these are the three errors and when we say that if this condition is satisfied, then the error due to linear interpolation will be almost equal to 10 to the power minus t. We are looking at these two errors only. We assume that error is sufficiently small to be ignored and what I want to show is that if this condition is satisfied, the sum of this error and this error will be approximately equal to 10 to the power minus t, that is if I use a linear polynomial, this will be about 10 to the power minus sum of this and this error will be about 10 to the power minus t, right.

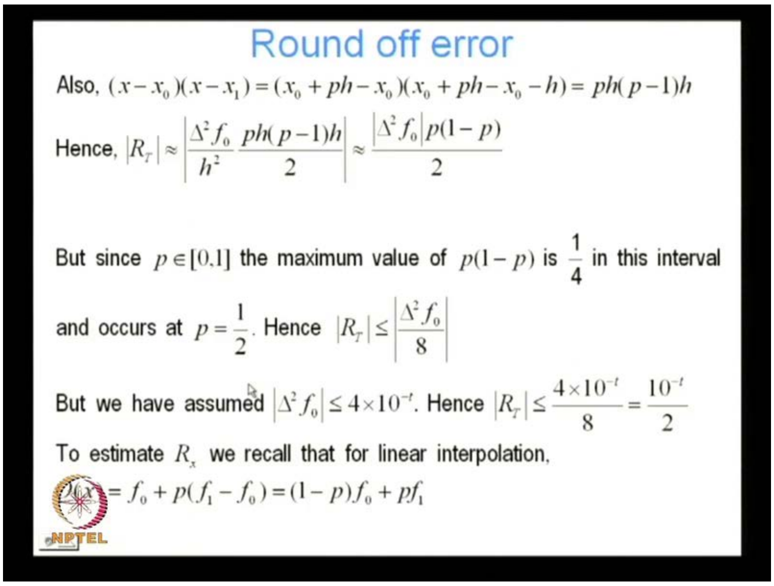
(Refer Slide Time: 45:59)



So, denoting the step size, we are looking at equidistant. Remember, we are now looking at equidistant grid and if we denote this step size in the table as h, we want to linearly interpolate the function value at an intermediate location say x is equal to x 0 plus p h p goes between 0 and 1, where f of x 0 and f of x 1 are known. I know that those simplicity I am looking at a point between x 0 and x 1, it can be anywhere in the grid, right, but I want to find the function value at a point which lies between x 0 and x 1 and it is at a distance p h from x 0.

From the result f of x minus Q x is equal to that, this we know is true, right. This is the error, f of x minus the polynomial approximation to f of x. That is my error x i belongs to that interval for linear interpolation. It is clear that this truncation error. This is the truncation error f of x minus Q of x mod of that is going to be given by this m plus 1. M is equal to 1 because I am using a linear polynomial. So, m plus 1 that is 2 f 2 xi by factorial 2 x minus x 0, x minus x 1, right. So, that is my truncation error, but if we go back several lectures, when we first started talking about the forward difference operator, we were able to show that for the forward difference operator, this relationship is true, right.

(Refer Slide Time: 47:48)

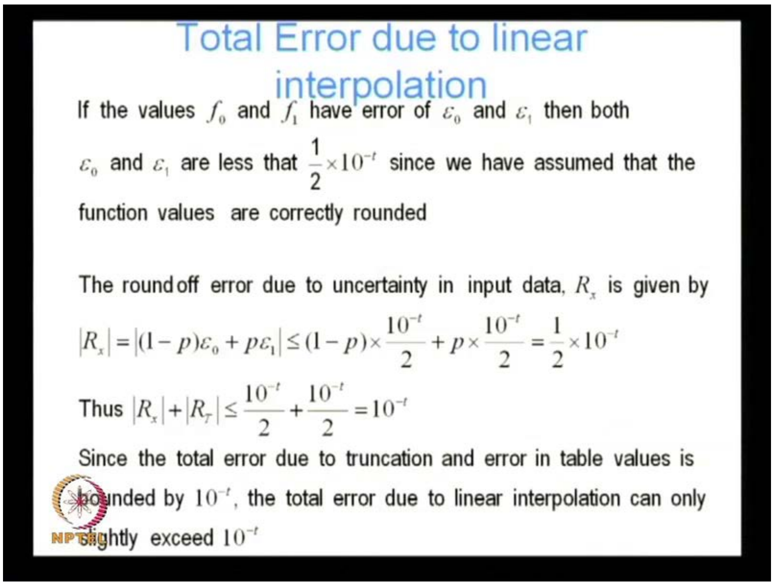


So, I can write that, but before that I want to get an expression for this term in terms of p n h. So, x minus x 0, x minus x 1 is equal to x 0 plus p h minus x 0 because x is equal to x 0 plus p h minus x 0. That is equal to x 0 plus a h minus x 0 minus h, right. So, that is equal to p h into p minus 1 h. Therefore, the truncation error after replacing this thing by that thing and replacing that in terms of this, I get that my truncation error is approximately equal to that which is equal to this times p into 1 minus p by 2, but since p belongs to 0 and 1, why does p belong to 0 and 1. Because it lies between x 0 and x 1, right. That is why p belongs to 0 and 1. We can show that this is a quadratic, right and the maximum value of this quadratic is one-fourth, right. We are interested in finding a bound for this.

So, we are interested in the maximum value. The right hand side can take the maximum value of p times 1 minus p is one-fourth and it occurs when p is equal to half. Therefore, we know that this must always be less than delta square f 0 times half into 1 minus half by 2. So, it is equal to one-fourth by 2. That is equal to one-eighth. So, this is always less than or equal to delta square f 0 by 8, but we have said that this is lesser than or equal to that, right. Therefore, mod of R T, the truncation error must be lesser than or equal to 4 into 10 to the power minus 8 minus t by 8. That is equal to 10 to the power minus t by 2, right.

So, this is how we get a bound on the truncation error. Now that we know the truncation error, we want to calculate the error due to round off, right. So, we know that for linear interpolation Q of x is equal to f 0 plus p f 1 minus f 0, right. So, at x 0 plus p h, the function value is going to be f 0 plus p f 1 minus f 0, right. Just by linear interpolation I have f 0 here, I have f 1 here. So, at a point p between 0 and h, that has to be equal to f 0 plus p times f 1 minus f h f 0. So, that is equal to 1 minus p times f 0 plus p times f 1, but we know that each of these function values have been evaluated correctly up to the t-th decimal place, right. That is what we assumed that each of these function values are evaluated correctly up to the t-th decimal place.

(Refer Slide Time: 50:56)



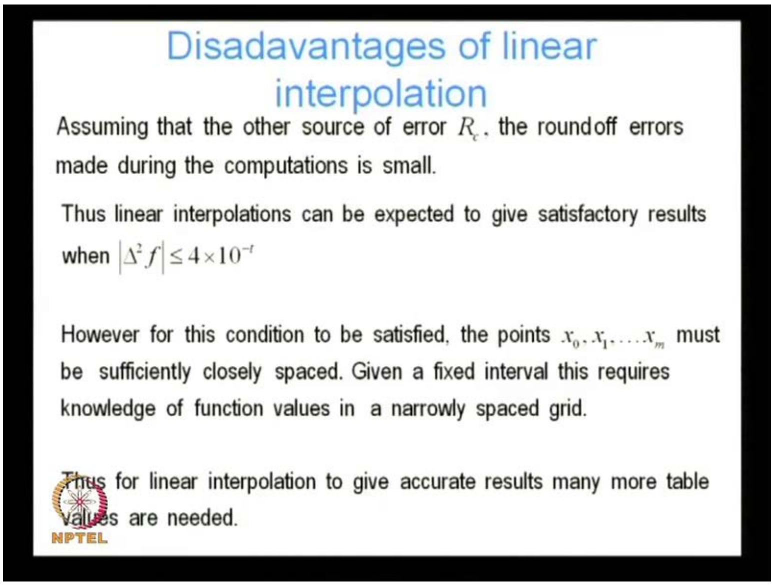
Therefore, if the error in these function values are denoted as epsilon 0 and epsilon 1, then both epsilon 0 and epsilon 1 must be less than half into 10 to the power minus t because it is been rounded off after the t-th decimal place. So, whatever the error is due to round off must be less than half into 10 to the power minus t, right since we assume that the function values have been correctly rounded off after the t-th decimal place.

The round off error due to uncertainty input data R x is therefore given by mod of R x is equal to mod of this, right. So, 1 minus p time epsilon 0 is the error in f 0 plus p of epsilon 1, right. We know that because that is my function value. The error in the function value will be the error arising out of this and the error arising out of that, right. So, 1 minus p epsilon 0 plus p epsilon 1 which is lesser than or equal to 1 minus p times 10 because that is bound on epsilon 0, that is the bound on epsilon 1 must be less than or equal to 1 minus p times 10 to the power minus t by 2 plus p times 10 to the power minus 2 t by 2 which I add together, I get half into 10 to the power minus t.

Now, we saw that the bound on the truncation error was 10 to the power minus t by 2. We have found that the bound on the round-off error is also 10 to the power t 10 to the power minus t by 2. Therefore, if I add these two bounds, so this must be less than or equal to 10 to the power minus t by 2 plus 10 to the power minus t by 2 which is equal to 10 to the power minus t, right.

So, that tells me that if this condition is satisfied and I have calculated my function values correctly up to t decimal places, then if I do a linear interpolation, then I can be assured that the error due to the linear interpolation will be of the same order as my round-off right error, but the critical thing is not going to be satisfied easily unless the interval is really small for an arbitrary function. What does that mean is that linear interpolation I can do with very good accuracy if the interval over which I am doing, the interpolation is very small. If my interpolation is large, I cannot use linear interpolation which is something probably you long knew by intuition, but this is a mathematical proof for that, right.

(Refer Slide Time: 54:26)

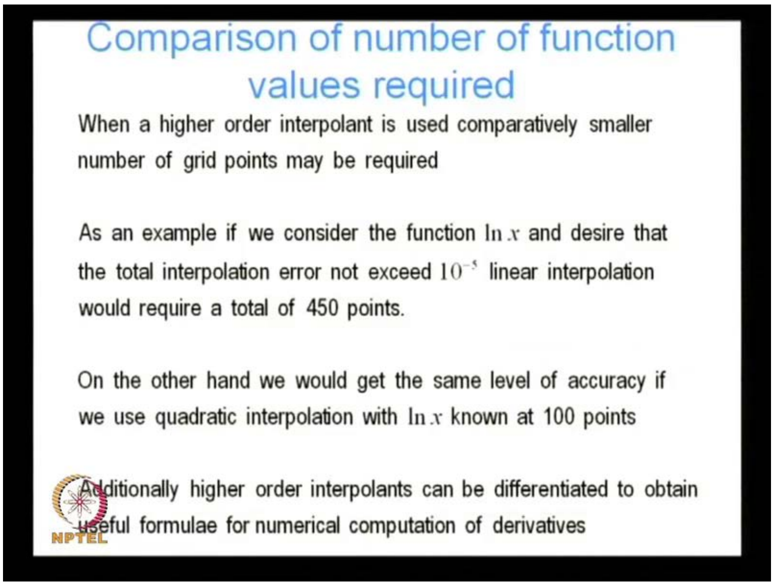


So, this assumes that the other source of error due to the computations with f 0 f 1 is sufficiently small, that is linear interpolation can be expected to give satisfactory results when this condition is satisfied. However, for this condition to be satisfied, the points x 0 x 1 through x m must be sufficiently closely spaced, right. They must be really small interval given, a fixed interval. This requires knowledge of function values in a narrowly spaced grid. That means that if my grid has to be really small, right.

If I am going to interpolate, my grid has to be very narrow. So, again those of you have done finite elements, if you think of p refinement versus h refinement where you make the size of the element smaller and you get good accuracy, that is h refinement and then the other way of getting at it is p refinement. You increase the order of the polynomials.

So, exactly that same thing is based on this idea, right. It is exactly that idea. If I have to get good result with linear interpolation, then I have to use a really small grid, right. If I do not want to use a really small grid, I have to use a higher order polynomial fit, right. That is for linear interpolation to give accurate results; many more table values are needed. I have just given an example here.

(Refer Slide Time: 56:03)



I will stop with that. For instance, if you are considering a function l n x and desire that the total interpolation error not exceed 10 to the power minus 5, then linear interpolation would require a total of 450 points, right. On the other hand, if you would like to get the same level of accuracy, if we use quadratic interpolation, then we need only 100 points, right. So, that is very important. So, we will continue with this discussion in the next lecture.

Thank you.