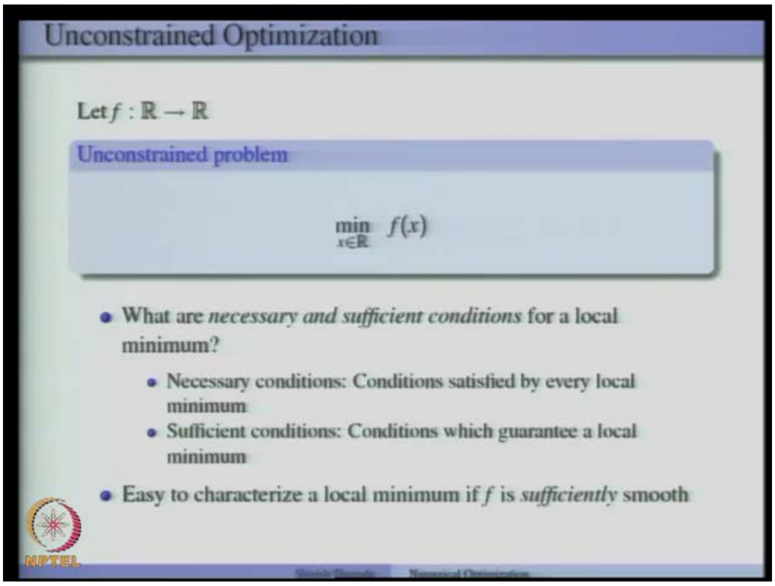
Numerical Optimization
Prof. Shirish K. Shevade
Department of Computer Science and Automation
Indian Institute of Science, Bangalore

Lecture - 5
One Dimensional Optimization (contd)

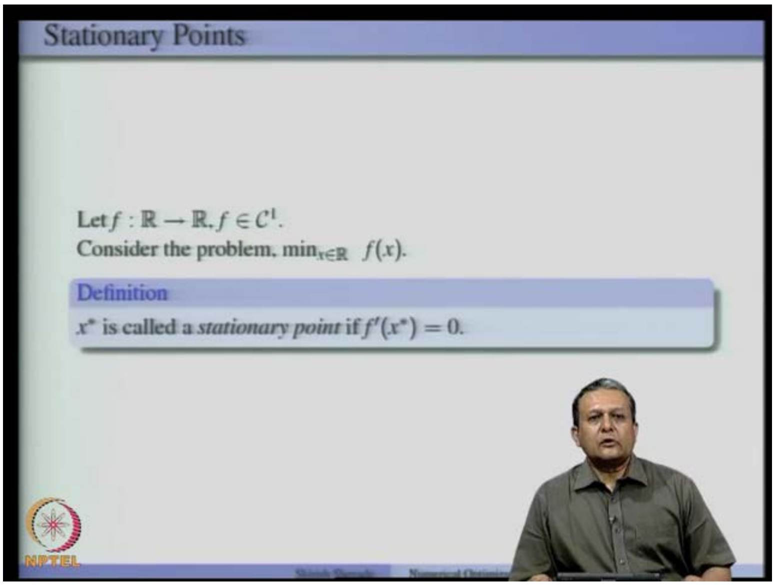
(Refer Slide Time: 00:28)



So, welcome back to this series of lectures on numerical optimization. So, in the last class, we were looking at unconstrained optimization problem. We saw that for solving any general constrained optimization problem, we need to solve unconstrained optimization problem. So, this one-dimensional unconstrained optimization problem plays a very important role in solving multi dimensional constant optimization problems. So, that is why, we have to spend some time studying about how to find the solution of a one dimensional constrained optimization problem.

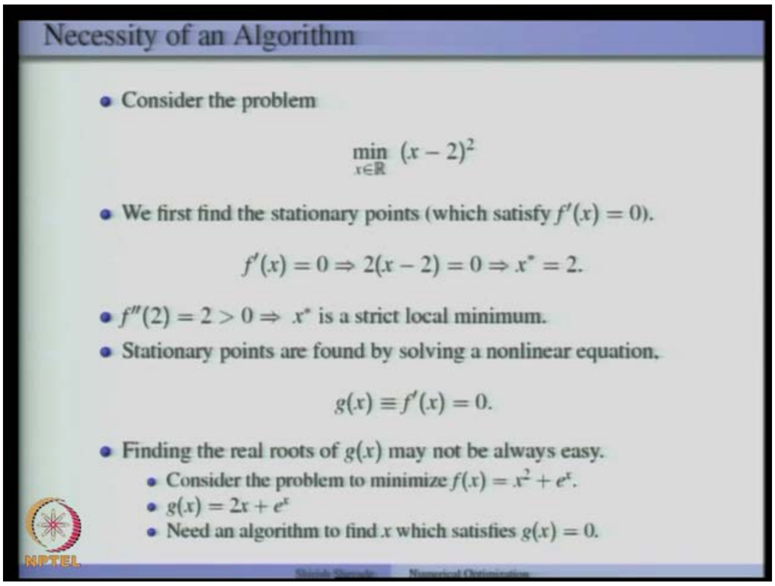
So, here is a problem that we are looking at where f is the function from R to R. We want to minimize f (x), x were reserve the entire set of real numbers. In the last class, we looked at the necessary and sufficient conditions for a local minimum. So, the necessary conditions are the conditions, which are satisfied by every local minimum. The sufficient conditions are the ones which guarantee a local minimum. Now, if the function f is sufficiently smooth, then we saw that it is easy to characterize a local minimum.

(Refer Slide Time: 01:40)



We also saw the definition of a stationary point in the last class, where we defined the stationary point for a continuously differentiable function f to be a point where the derivative of the function vanishes. So, it is this stationary point that will be interested in if a function is a differentiable.

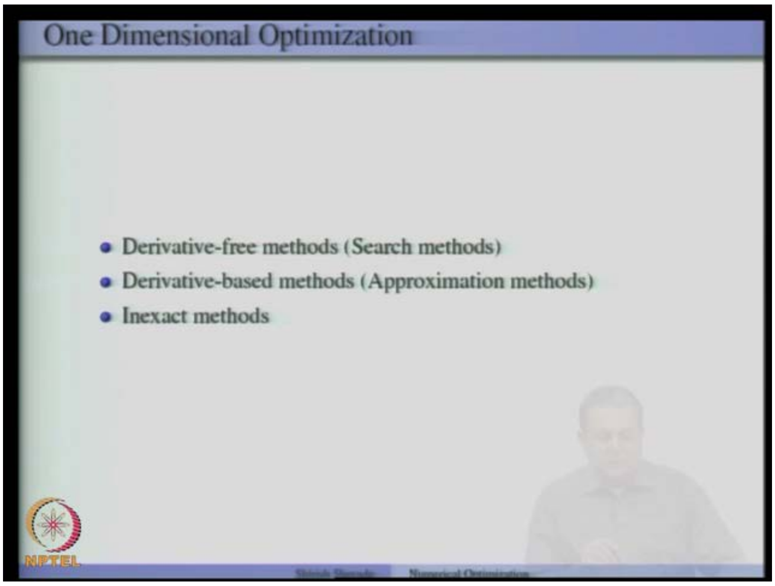
(Refer Slide Time: 02:01)



So, if we want to find out the stationary point, we have to equate the derivative of a function to 0. So, this is the example that we saw last time that if we consider a function x minus 2 square and if you want to minimize this, so we first find out f dash x and equate it to 0. That gives us the minimum. Then we check the second derivative and if it is greater than 0, we can say that x star is a strict local minimum.

Now, in general, for a any non-linear function, it may not be always easy to find out a stationary point like what we have got here. For example, if we consider a function f x is equals to x square plus e to the power x, which is shown here. Then the derivative is 2 x plus e to the power x. If we equate it to 0, we cannot get a closed form solution for x, so we need an algorithm to find out the x, which satisfies g x equal to 0. So, in today's class, we are going to look at some of those algorithms. Now, remember that we are looking at 1 dimensional optimization problem. So, there are different methods which can be used to solve this 1 dimensional optimization problem.

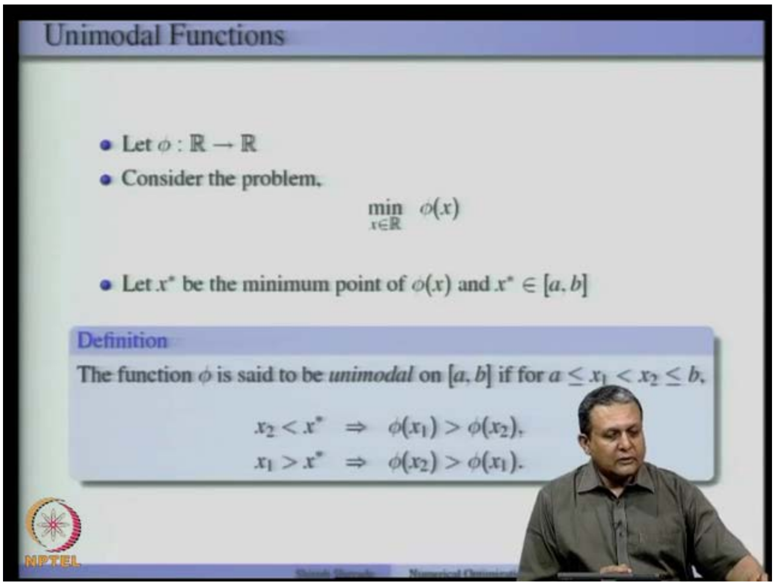
(Refer Slide Time: 03:25)



So, they can be divided into different categories. So, 1 set of methods are called search free methods or search methods. So, these methods are derivative free methods. So, one looks at the function values at different places in the interval and then tries to find out, tries to reduce the interval of uncertainty and so on. The other set of methods are called approximation methods. They are based on derivative information. So, it could be either a first derivative or second derivative. Then another side of method exists, which are called inexact methods. So, in which case, we are not really worried about finding the exact solution of given 1 dimensional optimization problem?

So, this will be important when we have to solve one dimensional optimization problem many times. So, every time going towards the exact solution does not makes sense. So, one has to resort to inexact methods. So, we will study in exact methods some time later in this course, but in this class, we will study the derivative free methods and derivative based methods.

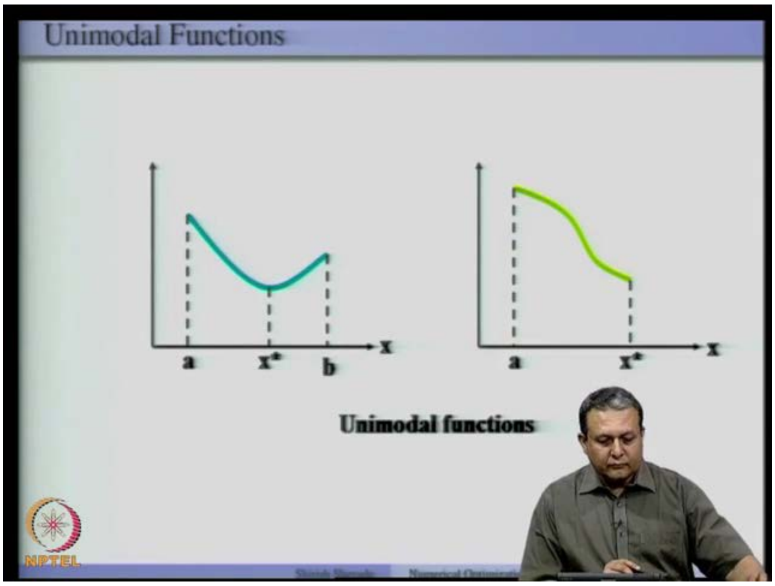
(Refer Slide Time: 04:38)



So, to understand these methods, we need the notion of unimodal functions. So, let phi be function from R to R. Let us consider the problem to minimize phi x over R. Let us also assume that x star be the minimum point of phi x. It belongs to the closed interval a to b. Now, here is a definition of a unimodal function. The function phi is unimodal if if we have 2 points x 1 and x 2 in the close interval a to b, where x 1 is less than x 2. Then x 2 less than x star implies phi x 1 greater than phi x 2.

So, what it means that x 1 and x 2 both lie on the left side of x star. In such a case, the function is strictly decreasing on the left side of x star. If x 1 is greater than x star, so that means that x 1 and x 2 both lie on the right side of x star. So, on the right side of x star, the function is strictly increasing. So, this is the definition of a unimodal function. So, let us now see how this unimodal functions look like.

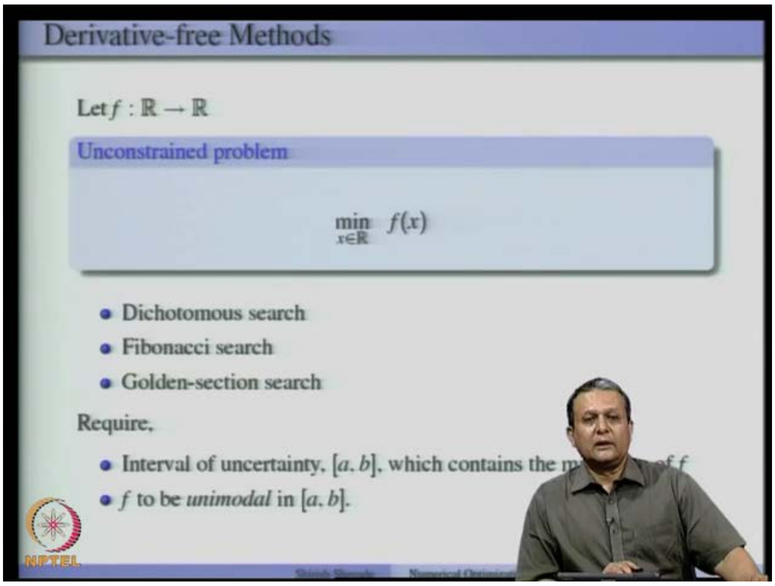
(Refer Slide Time: 05:53)



So, here are some examples of unimodal functions. So, a, b is the interval in which the minimum of the function lie. So, here b is nothing but x star in the right panel. Now, if you look at the left panel, this is the minimum of the function in the interval a b. so you will see that on the left side, the function is strictly decreasing. Then on the right side of x star, the function is strictly increasing.

Similar is the case in the figure on the right side. So, on the left side of x star, the function is strictly decreasing. Then star being a point of that interval, there is nothing on the right side of x star. So, this unimodal functions function assumption will be used in most of these lectures except the last method, which is Newton method.

(Refer Slide Time: 06:52)

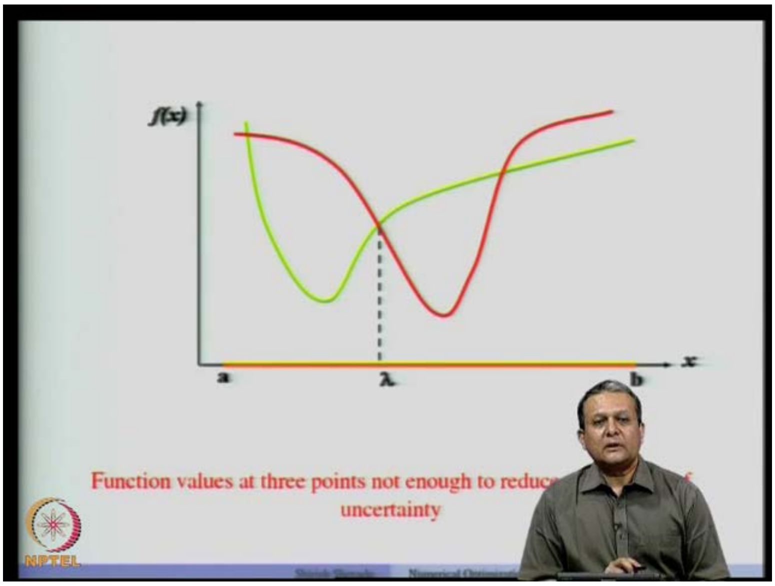


So, with definition of unimodal functions, we will now start looking at the derivative free methods. So, as the name suggests, these methods do not use any derivative information of a function. It just tries to use the function value at different points in a given interval. So, we are again considering a unconstrained optimization problem, trying to minimize f of x, x belongs to r. Here are some methods that we are going to discuss. There exist many others methods as well, but we will restrict ourselves to these 3 methods, which are quite practical methods.

So, one method is the dichotomous search method. The other one is Fibonacci search method. Then the limiting Fibonacci search method is called golden section search method. Now, all these methods require given interval of uncertainty a, b which contains the minimum of f. We also assume that f is unimodal in the closed interval a, b. So, the idea is that we start with some interval of uncertainty a, b and try to reduce the length of that interval has the iterations progress.

Finally, we reach close to the solution. This interval of uncertainty is also called bracket. So, we have to ensure that the minimum of the function f always lies in the bracket and then keeps on reducing the bracket such that the minimum gets always gets trapped in the bracket. Finally, we reach the actual minimum of f. The important assumption that we are using here is that f is unimodal in the close interval a, b.

(Refer Slide Time: 08:38)

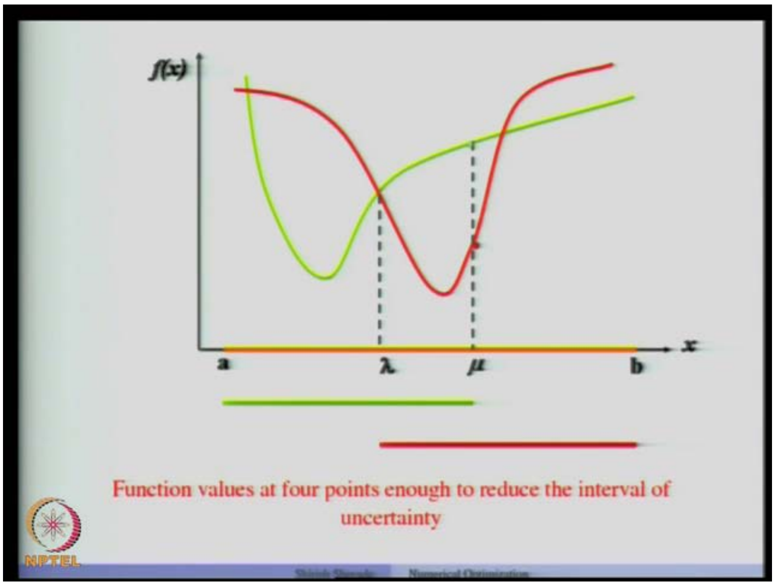


So, now, here are 2 functions. One is shown with the green line and the other one is shown with the red line. Now, a, b is the interval of uncertainty, which is given to us. So, that means that the interval a, b contains the minimum of each of this functions. Now, note also that each of these functions is unimodal functions. Now, suppose I know the function values are at a and b. So, I know f of a and f of b. Then I choose some point lambda somewhere in this interval and find out the function value f of lambda.

Now, I have 3 pieces of information, f of a, f of lambda and f of b for the green line function and the red line function. Now, let us concentrate on the green line function. Now, you you can see that the the function has a minimum at this point and that lies in the interval a to lambda. If you look at the red line function, that has the minimum at this point. That lies in the interval lambda to b.

So, if I evaluate a function at a particular point, I really cannot say anything about which part of the interval does a minimum lie. It could lie either in l lambda or it could lie in the lambda b also. So, function values at 3 points are not enough to reduce the interval of uncertainty. Remember that we always have to reduce the interval of uncertainty in every iteration. So, knowledge of function values at 3 points is not going to be enough.

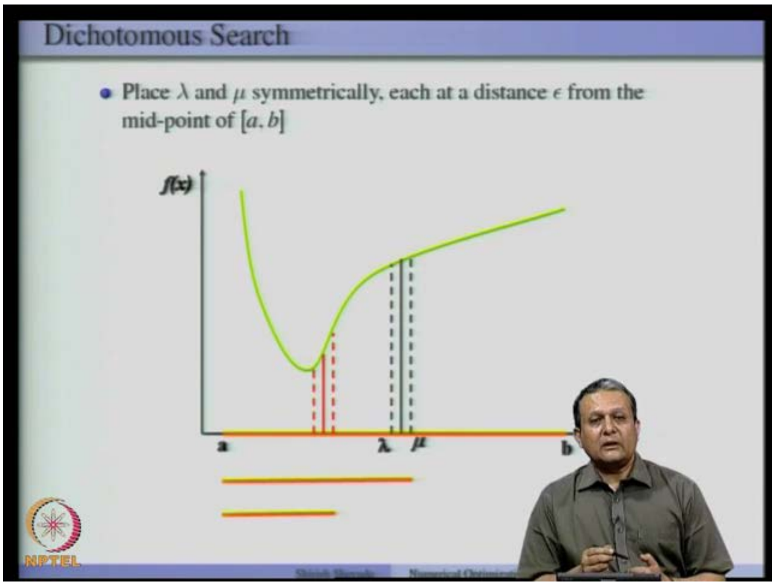
(Refer Slide Time: 10:34)



So, we would require function value at one more point. So, again we consider the same set of functions. Now, suppose I have the knowledge of the function values at lambda and mu lambda and mu both lie in the closed interval a and b. Now, let us look at the green function. Now, f of lambda is less than f of mu. So, on the right, the function is strictly increasing on the right side of lambda. Now, because of the unimodularity, there may not exist any minimum on the right side of mu as far as this green function is concerned. So, what we can conclude is that the minimum of this function would be bracketed in the interval a to mu.

On the other hand, if you look at the red function, red colored function, you will see that the minimum is bracketed in the interval lambda to b. So, for the green color function, the minimum would lie in the interval a to mu. So, that means that we have reduced the original interval a, b to the interval a mu, which is half size lesser than a, b. Similarly, for the red function, we have reduced the interval from a, b to lambda b. So, by having knowledge of function values at 4 points, we are able to reduce the interval of uncertainty. So, this is the very important fact that we will use for studying different derivative free methods.

(Refer Slide Time: 12:26)



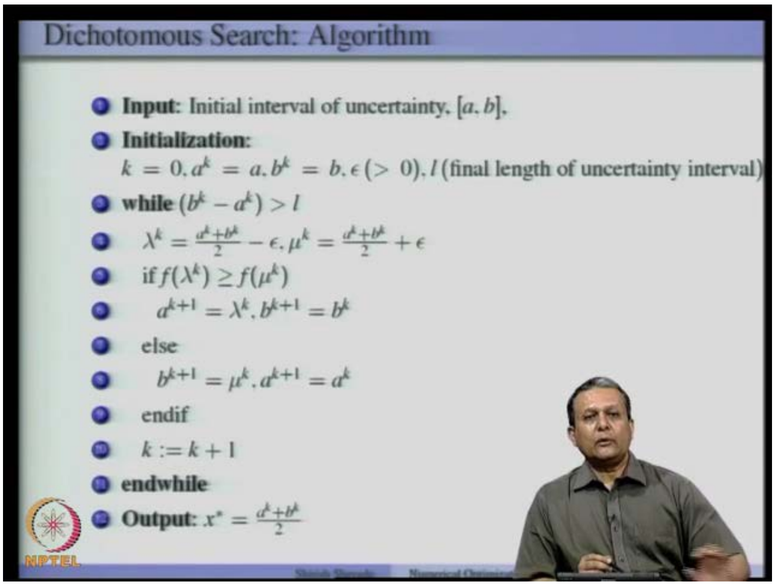
So, one of the simplest methods that is based on the derivative free approach is called the dichotomous search method. So, the idea here is very simple that given a interval a, b plus lambda and mu symmetrically, each at a distance epsilon from the midpoint of a, b. So, we start with the interval a, b. This is the midpoint of that interval. Then lambda and mu are plus symmetrically are at a distance of epsilon around this midpoint. Then we evaluate the function values at lambda and mu. So, this functions value function values are given here.

Now, you will see that f lambda is less than f mu. So, this means that the function is strictly increasing on the right side of lambda. So, that means that the root of this function, the minimum of this function should lie in the interval a to mu. So, the mu interval, which is shown here is the new interval of uncertainty or the new bracket. Now, you will see that this new bracket is about half the length of the original intervals length. So, that is we had this original interval a, b. We reduced it by almost a half quantity to form a new interval.

Now, now if you restrict ourselves to the interval a to mu, then again we have to do the same procedure. We will find out a midpoint of this, which is shown by the middle line. Then lambda and mu are placed around it at the distance of epsilon. Then the functional values are evaluated. You will see that f lambda is less than f mu. So, again we conclude that the interval of uncertainty is on the left side of this midpoint. In fact, this f of mu and the new interval of uncertainty are shown here.

So, you will see that every time we are trying to reduce the interval of uncertainty by half. Using this procedure, one can find the minimum to a reasonable accuracy. Now, one important point to note is that in every iteration, we do 2 function evaluations. So, one is f lambda and the other one is f mu. Similar thing is done here. It is a every iteration of dichotomous search requires 2 function evaluations and sometimes this number of function evaluations met are not to be expensive. So, one has to be careful about using the dichotomous search anyway.

(Refer Slide Time: 15:25)



It is a very simple method and very easy to implement. So, it is a simple algorithm, which implements dichotomous search method. So, we are given initial interval of uncertainty a, b and start with the initial iteration a is equal 0. The first point 0 is said to a and b 0 is said to b epsilon is some quantity, which is given, which is greater than 0. What is given to us is the final length of uncertainty interval.

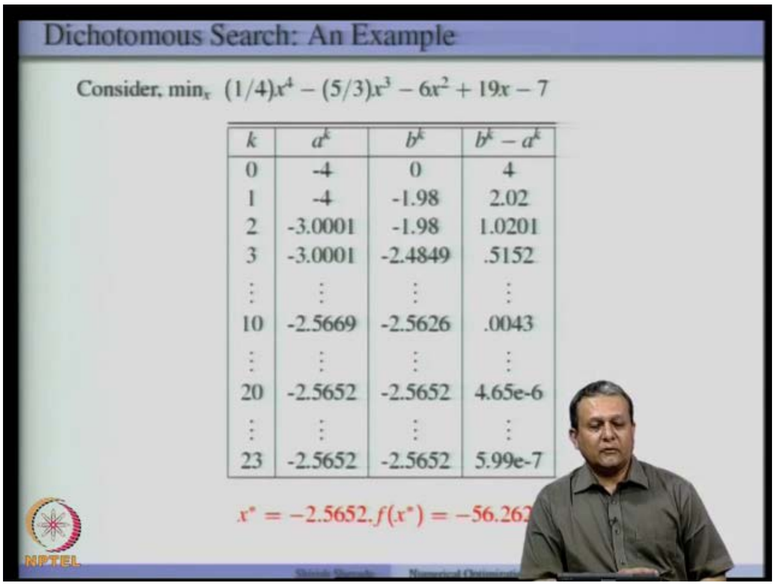
So, we have to iterate till the final interval as this length. So, as long as the the interval length is greater than all greater than l, we compute lambda k and mu k. They are computed using the midpoint of a k and b k minus epsilon and point of a k and b k plus epsilon. Now, if lambda k is greater than f of mu k, then what we have to do is that the function is strictly decreasing on the left side of lambda.

So, lambda k becomes a k plus 1 and b k will remain as b k plus 1. Otherwise, b k plus 1 will be mu k that means that function is the f lambda k is strictly less than f mu k. That means function is strictly increasing on the right side of lambda k. So, the right end of the bracket is mu k. Then the left end remains as a k. Then we increase the iteration counter and go to the next iteration.

So, this procedure is repeated. Finally, what we get is the interval of uncertainty a k b , which of the until less than or equal to r less than or equal to l. Once we get that, the algorithm will come out. Finally, the x star, the minimum of the function will be that midpoint of the final interval of uncertainty. So, it is a very simple algorithm, but requires 2 function evaluations, the f lambda k and f mu k at every iteration. That may turn out to be expensive in some cases.

So, if you look at this algorithm, every time the length of the interval is reduced by half, so after k iterations, the length will be 1 over 2 to the power k into b b 0 minus a 0 or b minus a, which is the original interval of uncertainty. So, by controlling k, one can reach any desired accuracy of the final length of uncertainty interval.

(Refer Slide Time: 18:16)

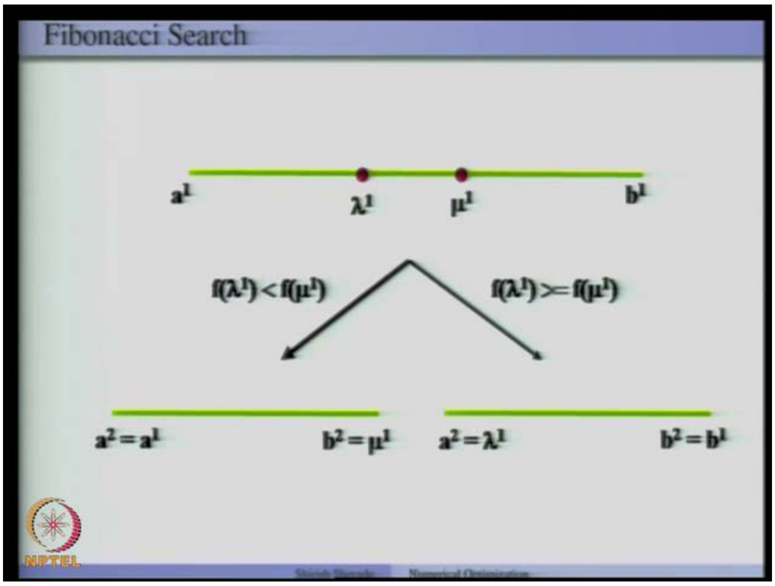


Now, here is a simple example. We have considered the function, to minimize the function, which is to be minimized is given here. Here are the iterations of dichotomous search. So, they start with the interval uncertainty to be minus 4 and 0. So, the difference between b k and a k is 4. As the iterations progress, this difference between b k and a k comes down as you see here.

So, you will see that initially the difference was 4; and thereafter 10 iterations, it is about 0.0043. If you go down further, this difference comes down further. Suppose, if you want the final interval of uncertainty to be less than 1 1 into 10 to the power minus 6, then we would end up in something in a solution like this. So, you will see that the a k and b k are almost close to each other with a very small value.

So, it is a very small difference between them and x star. The minimum of the function turns out to be minus 2.5652 in this case. The corresponding value of f of x star is given here. So, you will see that there is a significant reduction in the interval of uncertainty at every time. So, 1 over 2 to the power k that is the reduction of the ratio of the final length of the k th interval divided by the length of the first interval that will be 1 over 2 to the power k.

(Refer Slide Time: 20:17)

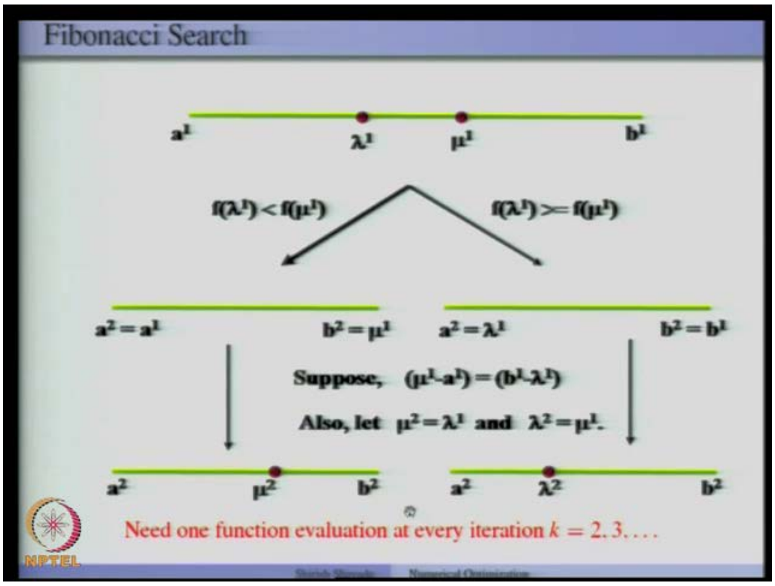


Now, in the previous method that we studied, we saw that we always needed 2 function evaluations per iteration and that 2 turnout to be expensive. So, can we better look at another case? Let us first look at simple case. Now, a 1 b 1 is a interval of uncertainty, which is given. Suppose that we have placed lambda 1 and mu 1 at 2 different places in this interval. Now, we know that the function is unimodal.

So, f of lambda 1 is less than f of mu 1. So, that means that the function is increasing on the right side of lambda 1. So, the the bracket has to be a 1 to mu 1. Similarly, if f of lambda 1 is greater than equal to f of mu 1, then the bracket has to be lambda 1 to b 1. So, this is what is indicated here. Now, you will see that this n point of this interval is same as mu 1. The n point of this interval in the other case is same has lambda 1. Now, we also have, if you consider this case, we also have knowledge of 1 point lambda 1 in this interval. Now, can we use that to some of its some method?

Similarly, if you look at the other interval, so whose left and point is lambda 1. Now, we know the value of f of mu 1 here where mu 1 lies in this interval. Now, 1 function evaluation is already done. So, we just have to add 1 extra point, which could be on the left of this lambda 1 and could be on the right side of this mu 1. So, that would reduce the number of function evaluations. So, let us see how to do that.

(Refer Slide Time: 22:30)



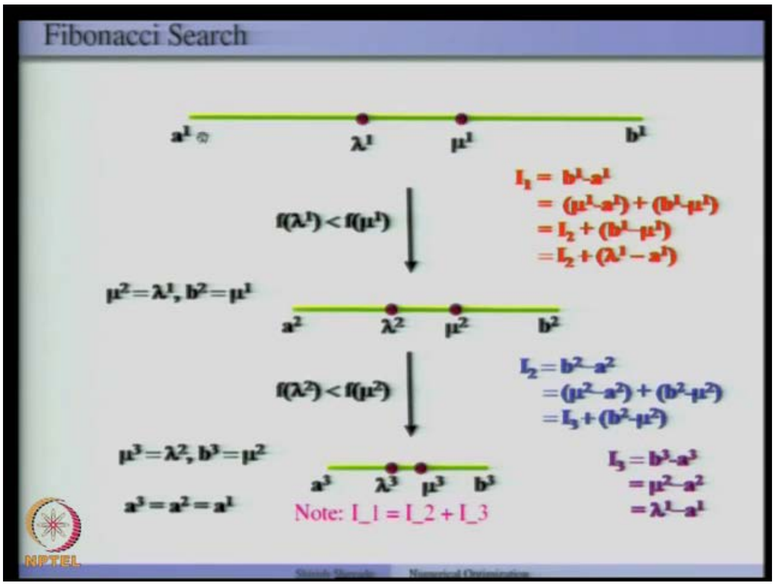
So, see the same case. Now, let us assure 2 things; 1 is that mu 1 minus a 1 is same as b 1 minus lambda 1. So, that means that lambda 1 and mu 1 are always symmetrically placed. So, when we go to k th iteration, we will have lambda k and mu k. They are symmetrically placed so that the assumption that we make, so whenever we get those 2 points, they are always the symmetrically placed in the interval. Now, let us also assume that mu 2 is equal to lambda 1.

So, if we look at this interval, this branch, we are concentrating on the interval a 1 to mu 1. Now, this mu 1 becomes a new end point of the next iteration b 2 and a a 2 is the nothing but a 1, which is the left end point of the bracket of further k th iteration. Now, let us consider this branch. Let us assume that mu 2 is lambda 1. So, this lambda 1 is basically used here. That we will, we are going to call it as mu 2. Then what we have to do is we just have to find out lambda 2 here using some method. So, that means that from the first iteration to the second iteration, we will just need 1 function evaluation as compared to 2 function evaluations used by the dichotomous search.

Similarly, it is the case if you look at the right branch where f lambda 1 is greater than or equal to f of mu 1. Now, the left bracket of this is obtained using lambda 1. So, a 2 is set to lambda 1. The right bracket remains as b 1. So, b 2 is b 1. Now, what we do is that this mu 1, which was earlier available, we will call it has lambda 2. Now, the function value at lambda 2 is known because the function value of mu 1 is known.

Now, our aim is just to get mu 2. So, that means that it will require 1 function evaluation from the second iteration onwards. So, in the first iteration, we will need both lambda 1 and mu 1; the 2 function evaluations, but from the second iteration onwards, we will need only 1 function evaluation. If we cleverly choose our lambda 2 or mu 2 depending upon the case, that is what we will see now how to do that.

(Refer Slide Time: 25:08)



Now, we started with lambda 1 mu 1. Let us assume that f of lambda 1 is less than f of mu 1. Now, f of lambda 1 less than f of mu 1 means that our new bracket will be a 1 to mu 1. So, b 2, which is the right end of the bracket, is mu 1 and a 2 is same as a 1. Then lambda 1 became mu 2. Suppose that we have a way to find out a lambda 2. So, then the function evaluation of f of lambda 2 is done. Now, suppose f of lambda 2 also is less than f of mu 2. So, in that case, this mu 2 will become the right bracket, which is nothing but b 3 a 2 remains as a 3. Then this lambda 2 becomes mu 3. Then we find lambda 3. So, this is how the iterations would progress.

So, you will see that the difference between lambda k and mu k comes down as the iterations progress. Finally, there would be situation where lambda k and mu k would merge. So, our aim is to get this required lambdas and mus appropriately. So, we will see how to do that. Now, this is the little bit of abuse of notations here in the sense that sometimes we will call I 1 as the interval. In this case, I have used I 1 as a length of the interval. So, depending upon the context, we will know what I 1 or I 2 mean, either they could mean interval or a length of the interval.

Now, if you look at the the interval length at the first iteration, so the interval length is b 1 minus a 1. Now, b 1 minus a 1 is same as mu 1 minus a 1 plus b 1 minus mu 1. So, mu 1 minus a 1 assume that f of lambda 1 is length than f of mu 1. So, that means that a 1 to mu 1 is the interval length at the second iteration. So, mu 1 minus a 1 is I 2 and that is added to b 1 minus mu 1.

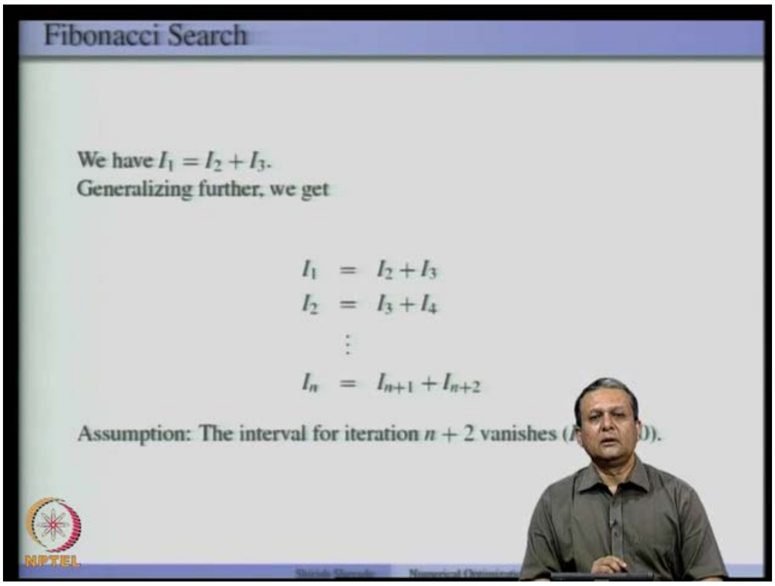
Now, remember that we have placed lambda 1 and mu 1 symmetrically. So, what it means is that b 1 minus mu 1 is same as mu 1 minus a 1. So, I can write b 1 minus a 1 as lambda 1 minus a 1. Now, now, let us go to second step. Suppose we find out lambda 2 because we already have got mu 2 that is equal to lambda 1. Now, suppose that we have found mu 2 and f of lambda 2 is less than f of mu 2. Then we find out that we are interested in the interval a to 2 mu 2. Now, b 3 is nothing but mu 2 and mu 3 is nothing but lambda 2. We find lambda 3. So, you will see that the interval length I 2 is nothing but b 2 minus a 2. One 1 can calculate to be I 3 plus b 2 minus mu 2; so I 3 plus b 2 minus mu 2.

Now, if you look at I 3, how is I 3 derived? I 3 is derived from b 3 a 3 to b 3 interval. So, the length of interval I 3 is b 3 minus I 3 and b 3 minus I 3 a 3 is same as mu 2 minus a 2.

So, b 3 minus a 3 is same as mu 2 minus a 2 because we have got this interval based on this condition and mu 2 minus a 2 because of the symmetry is same as lambda 1 minus a 1. So, what we can see is that I 1 can be written as I 2 plus I 3.

So, this is the very important observation that we have made here that the length of the interval, the first interval is the sum of the lengths of second interval and the interval obtained at the third iteration. So, you will see that this is I 1. Then I 2 is nothing but a 1 2 mu 1. Since, lambda 1 and mu 1 are symmetrically placed, so a 1 to mu 1 is same lambda 1 to b 1. This a 1 to lambda 1 is same has a 3 2to b 3. So, you will see that I 3 is same as I 1 is same as I 3 plus I 2. So, this is the very important observation that we are going to use.

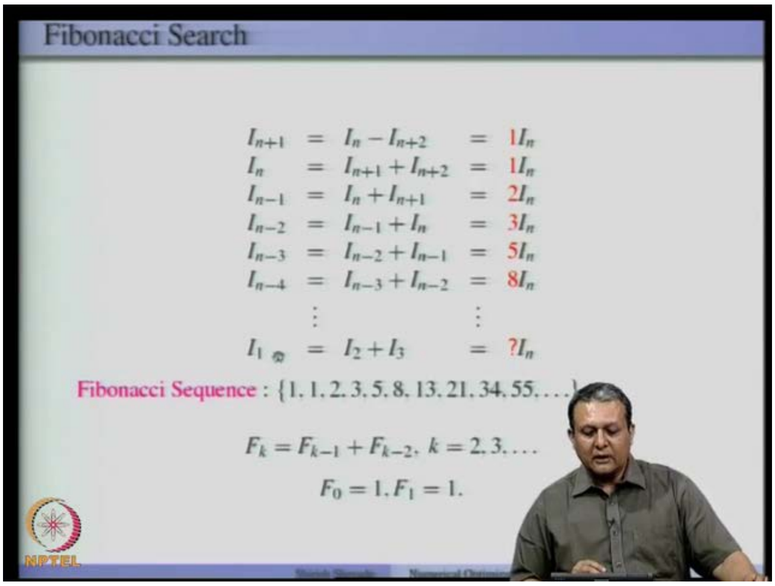
(Refer Slide Time: 30:17)



Now, we have this I 1 is equal to I 2 plus I 3. Now, if you generalize further, so what we get is I 2 is equal to I 3 plus I 4 and so on. Then we get I n is equal to I n plus 1 plus I n plus 2. So, now, we have n equations. Now, out of these n equations, I 1 is a given interval of uncertainty that before we started using our approach. So, I 1 is known. Now, there are n equations and n minus 1 variable. So, there exists infinite number of solutions assuming that they are consistent. Now, suppose we make 1 assumption that I n plus 2 is 0. So, that means that after n plus 2 iterations, the length of the interval is going to vanish.

So, let us make this assumption that after n n plus 2 iterations, I n plus 2 will be equal to 0. So, that means that we are now left with n variables, which are I 2 to I n plus 1 and n unknowns. So, we can find out, we can generate a sequence of intervals I 2 to n plus 1 with I 1 given and I n plus 2 0. So, let us see how to do that.

(Refer Slide Time: 31:42)



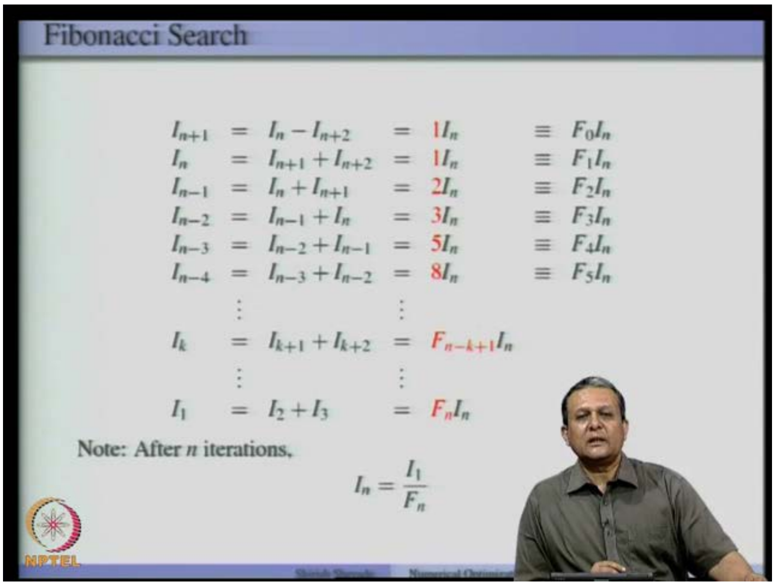
So, let us look I n plus 1. Now, if you look at I n plus 1, I n plus 1 is nothing but I n minus I n plus 2. So, I n minus I n plus 2 is nothing but I n plus 1. We know that I n plus 2 is going to be 0. So, that is nothing but I n. So, I n plus 1 is equal to 1 into I n. Now, if we go further, let us write write down I n. So, I n is nothing but I n plus 1 plus I n plus 2. Now, I n plus 2 is 0 and I n plus 1 that we have already seen. That is I n. So, I n is nothing but I n plus 1 that is nothing but 1 I n.

Now, we go 1 step further. So, let us look at I n minus 1. So, I n minus 1 is I n plus I n plus 1. Now, you will see that I n plus 1 is 1 into I n, I n is 1 into I n. so I n minus 1 will be 1 I n plus 1 I n that is nothing but 2 I n. So, what we have done is that we have started with I n plus 1 and I n. Then we can now find out I n minus 1. So, what it means is that at the n minus 1 th iteration, the interval is the interval length is 2 into I n, if you go further, so you will see that I n minus 2 is nothing but 3 into I n. So, the interval length at I n minus 2 is 3 into I n. How is this 3 derived? This 3 is derived from I n and I n minus 1. So, it means that we have derived it from this 2.

So, we take a sum of the previous 2 entries. Then we get 3. So, if you proceed further, so you will see that the length of every interval is obtained using the previous 2. If you go from I n plus 1 down to I 1, so I 1 is nothing but I 2 plus I 3. Then what should be the length here? So, that is the question that we would like to ask. Now, if you look at this sequence 1, 1, 2, 3, so let us assume that this 2 are given to us 1 and 1. Then we sum the previous 2 to get this quantity. Then similarly, from the previous 2 to get this, then 5 plus 3 will give us 8 and so on. So, that is how the the lengths are obtained.

Now, this sequence that we have given here in the red color is called the Fibonacci sequence. The recursive relation for this Fibonacci sequence is F k is equal to F k minus 1 plus F k minus 2. So, that means from k equal 2 onwards, you obtain this sequence by finding the sum of the previous 2 elements in the sequence. Then of course, we will assume that the first member of this sequence is 1 and the second member is 1. So, these 2 are 1. So, once they are given, we can start calculating F 2 onwards. So, this is called the Fibonacci sequence. Then if we assume that I n plus 2 is 0, then we can get the lengths of all the intervals based on the element of which element of the sequence we are talking about.

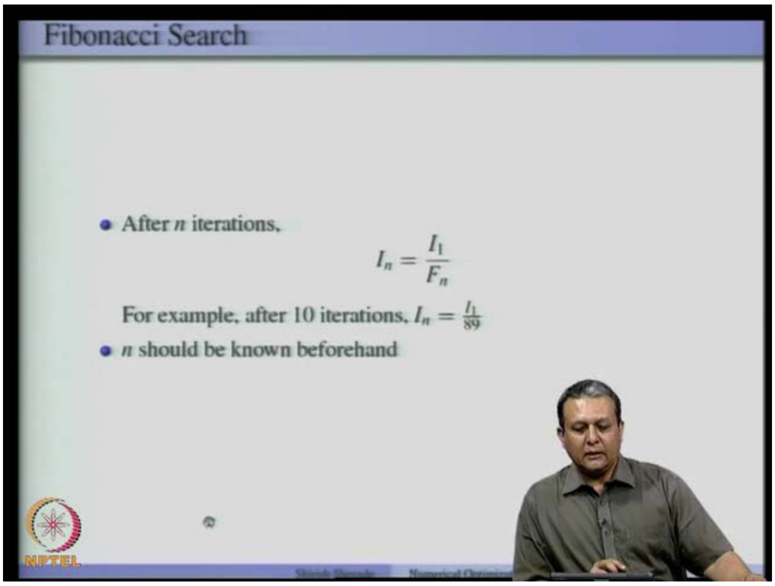
(Refer Slide Time: 35:11)



So, we will see how to do that. So, I n we saw that I n plus 1 is nothing but 1 I n. That is the first element of the Fibonacci sequence. So, F 0 I n I n is also 1 I n that is nothing but F 1 I n and so on. So, we can based on the value of n, we can decide which sequence, which element of the sequence of Fibonacci sequence are talking about. So, if we go further at the at the k th iteration, we are talking about F n minus k plus 1.

Therefore, at I 1, it will be F n I n. Now, this is the very important relation. I 1 is nothing but F n I n, which means that after any iterations, the length of the original interval I 1. So, the ratio of I n, the length of the interval after any iterations and the length of the interval at the original iteration is 1 over 5 n. In other words, I n is nothing but I 1 by F n. So, this is the important relation that we would like to use. So, if we generate the sequence like this, then how how is it going to be useful in generating our lambda and mu? That is what we will see now.

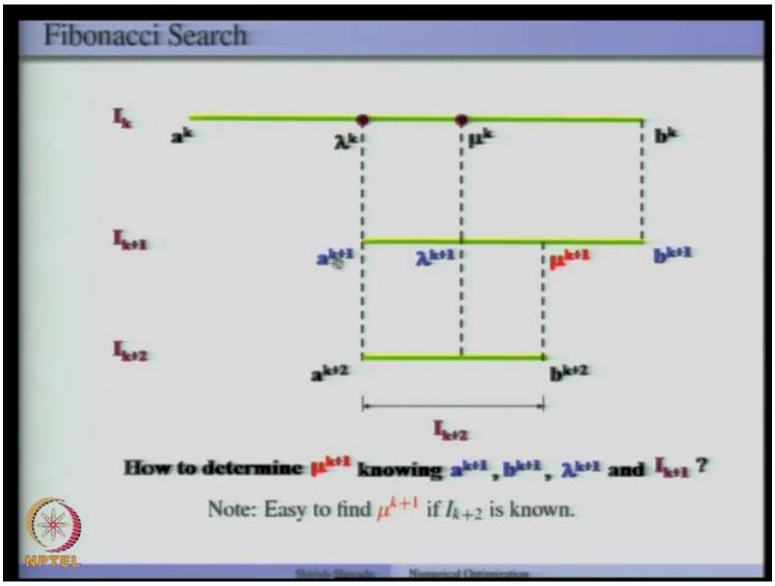
(Refer Slide Time: 36:38)



Now, a quick comment about the number of iterations needed by this Fibonacci search method is that after n iterations, the length of the interval comes to I 1 by F n. Now, F n if we consider 10 iterations, so F n is F 10. F 10 is 89. So, I n becomes I 1 by 89. Now, after 10 iterations, the number of function evaluations will be 2 for the first iteration. This is because initially, we do not have any idea. Suppose if you cleverly choose lambda and mu, then for the remaining 9 iterations, we will require 9 function evaluations.

So, in all, we will require 2 plus 9 that is 11 function evaluations. In 11 function evaluations, we were able to reduce the length of the interval by about 1 percent. So, 1 by 89 is almost close to 1 by 100. So, the length of the interval after n iterations is almost close to 1 percent of the original length of interval. Now, the only disadvantage here is that we should know n beforehand. Only then we can calculate what is what is going to be the final interval length.

(Refer Slide Time: 38:09)

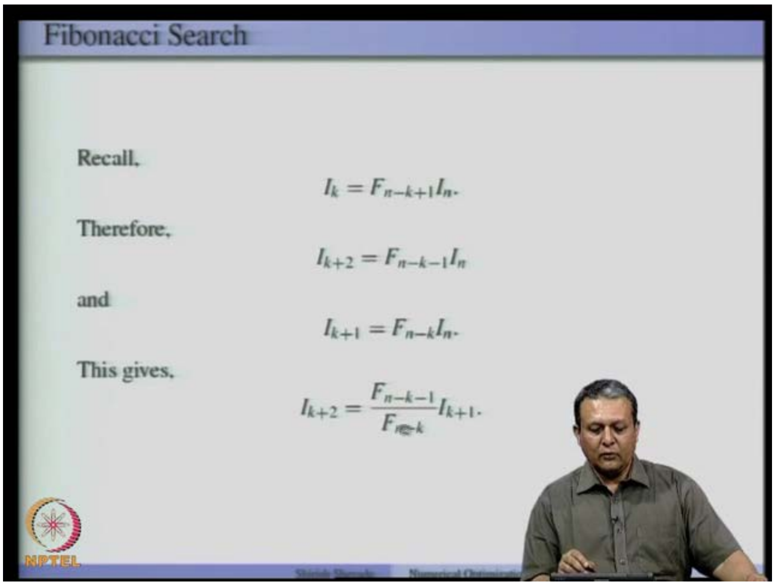


Now, let us see how to get lambda k and mu k from the previous iterations. So, at the k th iteration, we have a k and b k. Then lambda k and mu k are given here which are known. Now, let us assume that in the next interval, in the next iteration we are talking about the interval lambda k to b k. So, that means b k plus 1 is b k a k plus 1 is lambda k. Then lambda k plus 1 is nothing but mu k as we saw earlier.

Our aim is to get mu k plus 1. Now, we know that they are to be symmetrically placed right. Then suppose we know mu k plus 1 right and it turns out that the in the next iteration that k plus 2. The iteration we have to only worry about is the interval a k plus 1 to mu k plus 1. So, by the things that we have studied so far, we know that this mu k plus 1 will be the right end point of the bracket at I k plus 2. So, if I can get the interval length at the k plus 2 iteration, then I can, what I can do is that I can add that interval length to get mu k plus 1. Since, they are symmetrically placed; mu k plus 1 minus a k plus 1 will be same as b k plus 1 minus lambda k plus 1.

So, if we know a k plus 1, if we know b k plus 1, if we know lambda k plus 1, I k plus 2 and I k plus 1, I k plus 2, how do we get mu k plus 1? So, that is the question that we would like to answer. So, it turns out that if I know I k plus 2, mu k plus 1 is easily available. So, what I have to do is that I just have to add this interval length to a k plus 1 to get mu k plus 1.

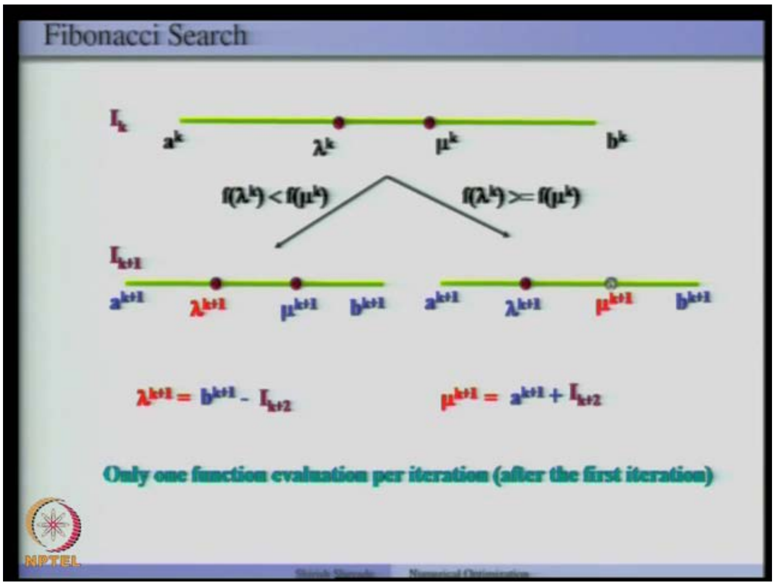
(Refer Slide Time: 40:10)



Now, here is a procedure to do that. So, let us recall that I k was found to be F n n minus k plus 1 into I n. Therefore, I k plus 2 also can be written as F n minus k minus 1 into I n. now, I k plus 1 also is written as F minus k into I n. Therefore, if you take using if you use these 2 equalities, then one can write I k plus 2 as the ratio of F n minus k minus 1 and F n minus k into I k plus 1.

So, if I know I k plus 1, if I know n and k, then I can derive I k plus 2. All this was possible because of this relation. This relation was possible because of the way we generated the sequence. So, we assume that given I n given I 1 and given that I n plus 2 equal to 0. We were able to generate a Fibonacci sequence and from which we derived this expression for I k. This is because of which we are able to get I k plus 2 with the knowledge of the length of the interval at k plus 1 th iteration. This ratio is easy to find.

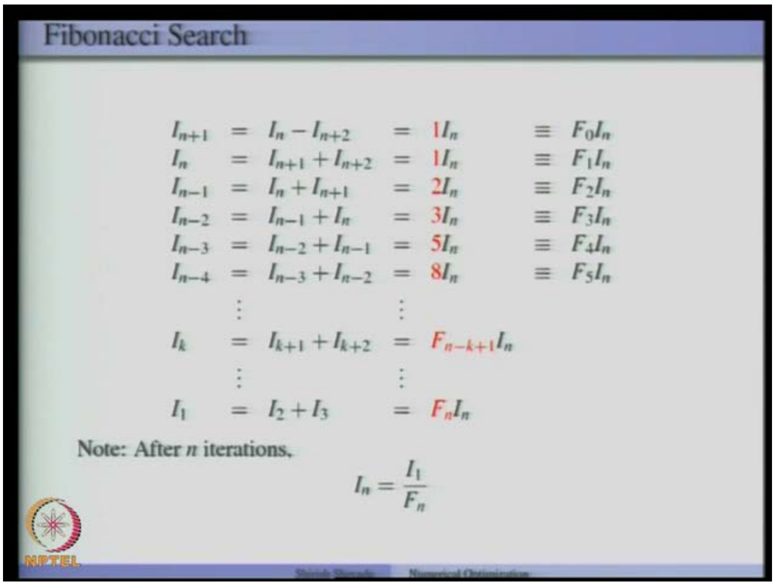
(Refer Slide Time: 41:30)



So, here is a procedure. So, we have the interval I k, which is a k to b k. Lambda k and mu k are given. The function values at these 2 lambda k and mu k are evaluated. f lambda k suppose is less than f of mu k, then we know that b k plus 1 is nothing but mu k. Mu k plus 1 nothing but lambda k a k plus 1 is nothing but a k. Lambda k plus 1 is evaluated using b k plus 1 minus I k 2. Similarly, in the other case, when f of lambda k is greater than or equal to f of mu k, so in this case, these quantities are known. Only this quantity which is shown in a red is unknown and that is evaluated by adding I k plus 2 to a k plus 1. So, that is what is shown here that mu k plus 1 is equal to a k plus 1 plus I k plus 2.

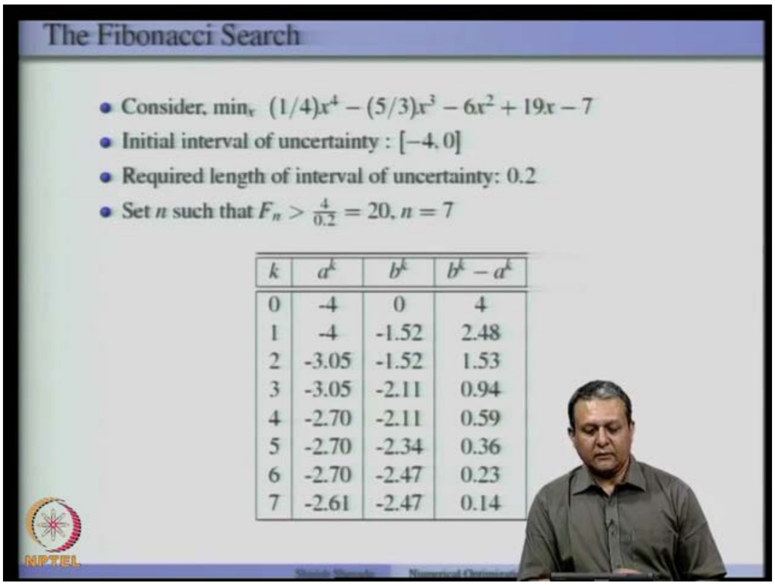
So, that important point to note is that after the first iteration, what we need to evaluate is only either lambda k plus 1 or mu k plus 1, the function and then the function value at that. So, only 1 function evaluation per iteration is needed after the first iteration. Also, it is easy to calculate these values if we know I k plus 2. So, with that, the algorithm becomes very simple. So, the idea is that we start with some initial interval a a 1 b 1 and then keep on reducing the interval of uncertainty. Then at final point, if we look at the previous expressions, we see this.

(Refer Slide Time: 43:13)



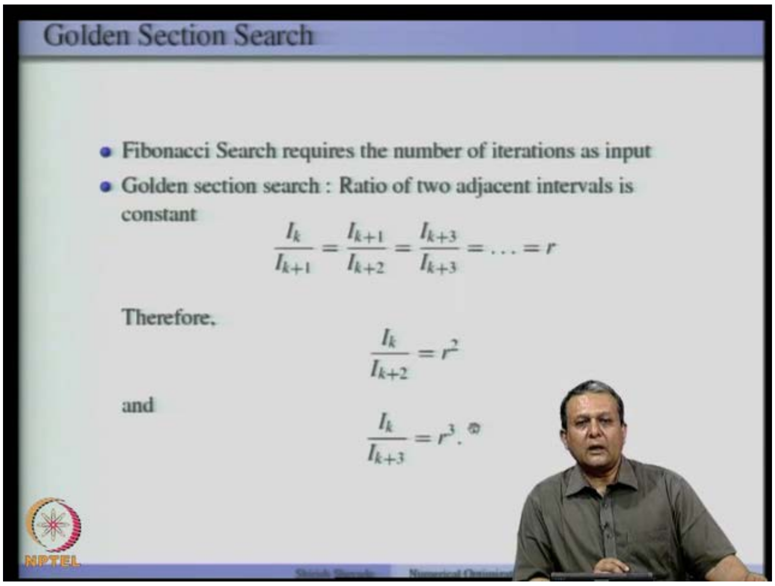
So, you will see that I n minus 1 is nothing but 2 I n. Then after this interval of uncertainty becomes half of the previous interval of uncertainty, so at that point, the lambda and mu merge into each other. So, we after n minus 1 iterations, will have lambda n minus 1 is equal to mu n minus 1. Then one just has to look at either the left interval that is a a n minus 1 2 lambda n minus 1 or lambda n minus 1 to b n minus 1. Then do an iteration of dichotomous search to find out what is the minimum point. So, if use this algorithm, the algorithm is easy to write. So, let us consider the same example that we saw earlier.

(Refer Slide Time: 44:13)



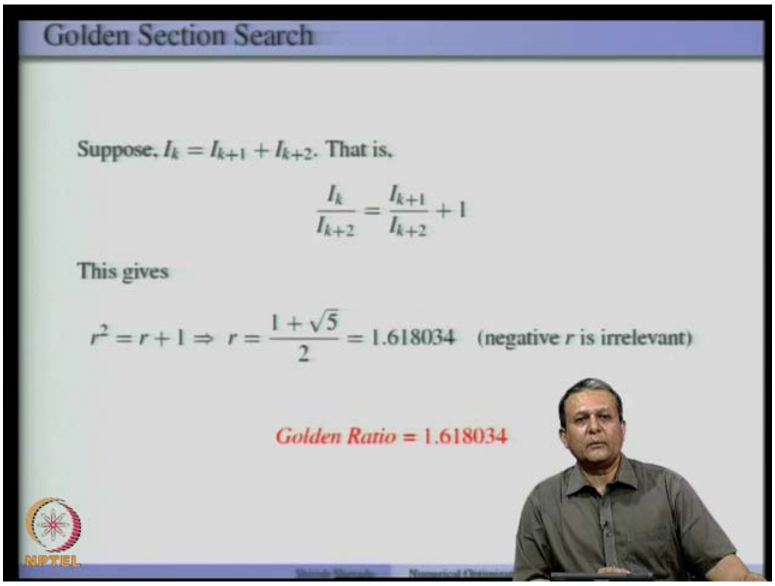
We are trying to minimize this function. The interval of uncertainty is the close interval minus 4 to 0. Suppose that the required length of interval of uncertainty is 0.2, then if you set n to b say one that is based on F n, then this is the set of iterations, which are given here for Fibonacci search method. So, you will see that the length of the uncertainty interval, which was 4 initially after 7 iterations, it came to 0.14. Then one has to use the appropriate method, 1 iteration of dichotomous search after n minus 1 iterations. That will give us the final minimum of the function at 2 certain accuracy.

(Refer Slide Time: 45:12)



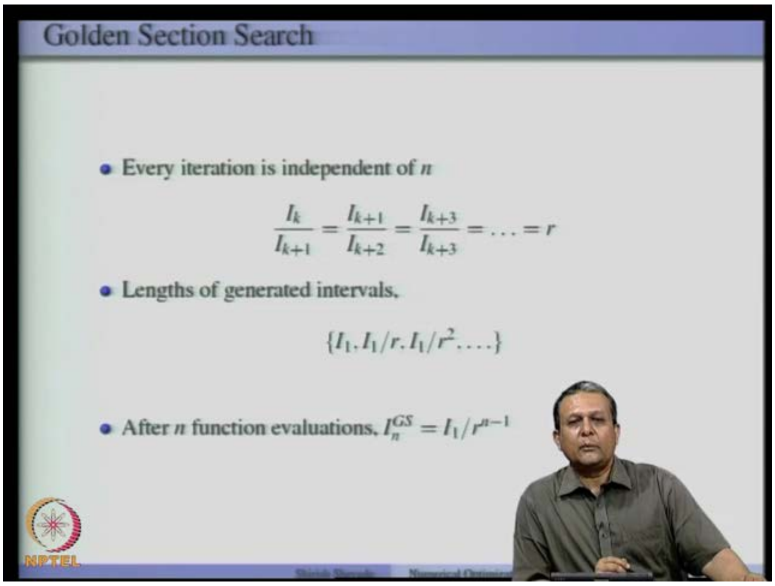
Now, as I said earlier that Fibonacci search requires n as the input. We have to fix n beforehand. Only then we can use Fibonacci search. Now, there is another method, which is called golden section search method, which is a limiting case of a Fibonacci search. So, here we assume that the ratio of 2 adjacent intervals is constant. So, that is I k by I k plus 1 is same has I k plus 1 by I k plus 2 and I k plus 2 by I k plus 3 and so on. This should be I k plus 2. Therefore, if take I k by I k plus 2, so that will be r square. Then I k by I k plus 3 is equal to r to the power 3 and so on.

(Refer Slide Time: 46:01)



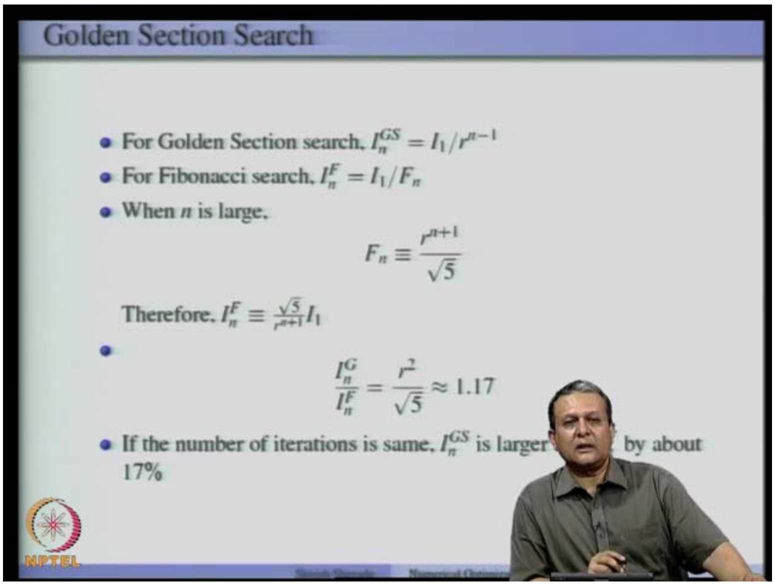
So, if we suppose continue to use the same relationship that we had for the Fibonacci search that is I k is equal to I k plus 1 plus I k plus 2 or in other words, I k by I k plus 2 is same as I k plus 1 by I k plus 2 plus 1. So, now, I k by I k plus 2 is r square. I k plus 1 by I k plus 2 is r. so we can write r square is equal to r plus 1. If we consider the positive root of this polynomial, then we get r equal to 1 plus root 5 by 2. This is nothing but 1.618034. We are neglecting the negative root of r because we do not want the negative ratios. So, this ratio is called the golden ratio. Hence, the name of the method has the golden section search.

(Refer Slide Time: 46:57)



Now, one important point to note for golden section search is that every iteration is independent of n that we saw earlier. Again, this should be I k plus 2 by I k plus 3. Now, if you look at the lengths of the intervals, which are generated after every iteration, so we start with I 1. Then the length of the next interval is I 1 by r and then I 1 by r square and so on. So, after n function evaluations, assuming that we use the same technique as we used earlier, we will see that for the golden section search, the length of the interval after n iterations is I 1 by r to the power n minus 1. Now, how how does it compare with Fibonacci search?

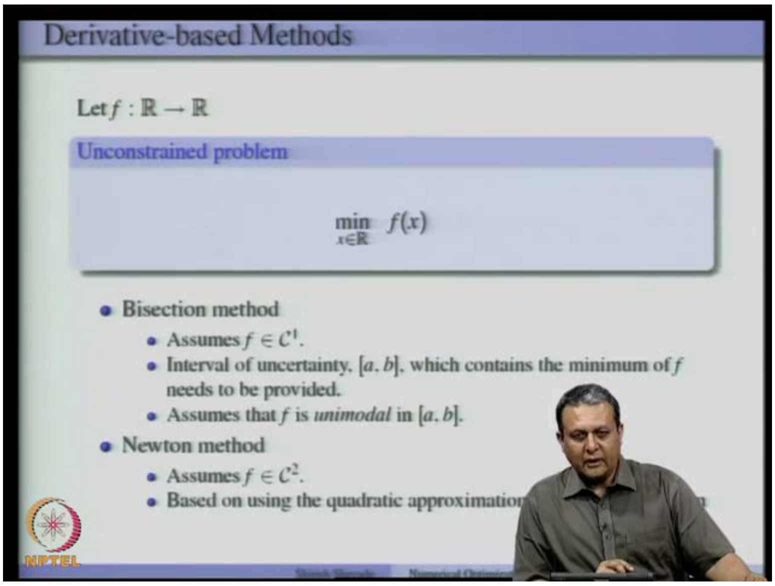
(Refer Slide Time: 47:48)



Now, if you look at the Fibonacci search, the length of the interval after n iterations is I 1 by F n and for the golden sections search, we saw that it is I1 by r to the power n minus 1. So, how do they compare? Now, one uses a relationship, which holds when n is very large. So, the relationship is that F n is approximately equal to r to the power n plus 1 by root 5. Therefore, one can write I F n as this.

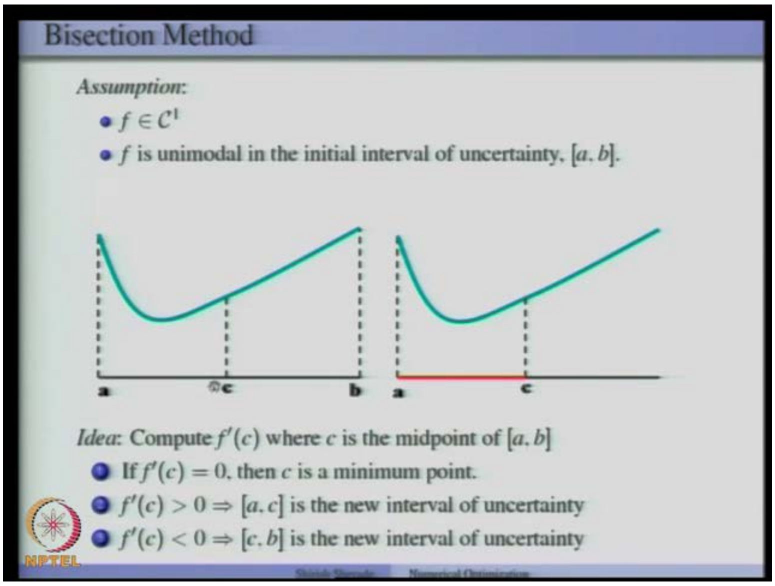
Then, the ratio of this 2 turns out to be r square by root 5, which is 1.17. So, what it tells is that when n is large, when this relationship holds after n evaluations, the length of the interval given by the golden section search is about 17 percent higher than the length given by the Fibonacci search. But, one advantage of golden section search is that it does not require knowledge of n beforehand, while Fibonacci search needed the knowledge of n beforehand. So, golden section search is typically used in practice because one need not know the value of n. One can reach the desired interval, one can reach the desired length of the interval of uncertainty has the iterations progress.

(Refer Slide Time: 49:16)



Now, let us look at some derivative based methods. Remember that we are trying to solve this unconstraint optimization problem. We are going to see a couple of methods. One of them is bisection method. These methods are derivative based. We assume that f is continuously differentiable for this bisection method. We also assume that the function f is unimodal. There is another method called Newton method, for which the function is assumed to be continuously twice differentiable. Then this is based on the quadratic approximation of the function at every iteration.

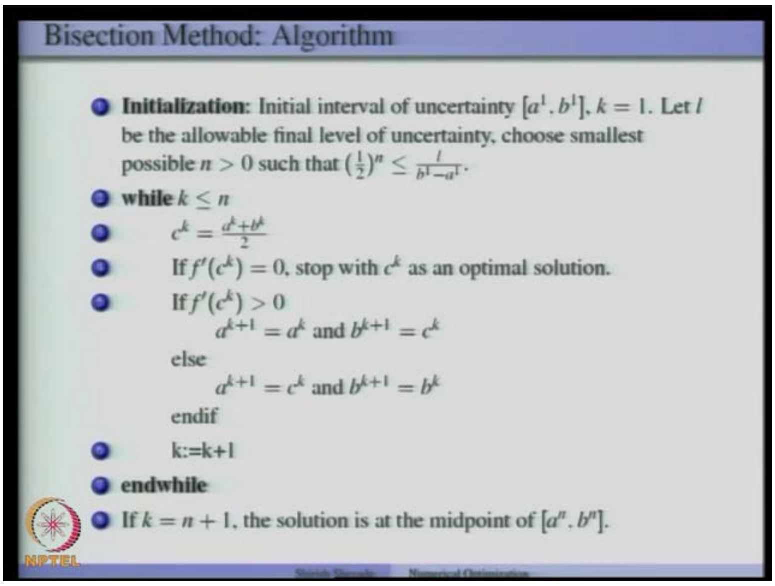
(Refer Slide Time: 50:01)



Now, let us see how the bisection method works. So, this is basically a variant of the dichotomous search that we studied. So, if you can recall that in the dichotomous search, we find the midpoint of the interval and then look at the function values at those 2 points, which are at a distance of epsilon from the midpoint. Now, instead of doing that, we find the midpoint and at this midpoint, we find out how what is the sign of the derivative? Now, if the sign of the derivative is positive as in this case, so that means that the function is going to increase further. Remember that we are using the assumption that the function is unimodal. So, next time we have to worry only about the interval a, c. So, the length of the interval gets reduced by half every time.

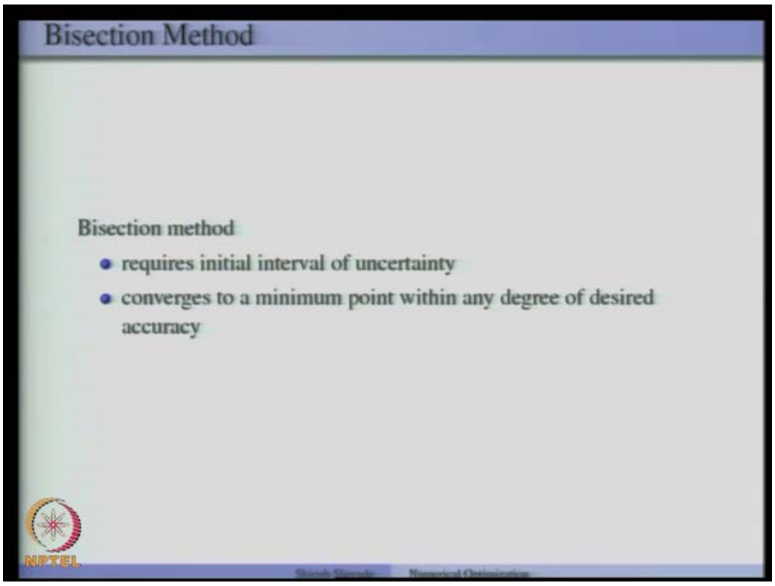
So, we compute f dash c where c is the midpoint of a, b. Then if dash c is 0, then we stop. If it is not 0, if it is greater than 0 that means we only concentrate on the left side interval a, c. f dash c less than 0 means we concentrate on the right side interval c, b. so every time the interval is reduced by half, so this is just a variant of the dichotomous search. So, instead of using 2 function evaluations around the midpoint, we use the derivative of the function at the midpoint.

(Refer Slide Time: 51:24)



So, the algorithm is a very simple. So, it is easy to understand this algorithm. So, I will not spend much time on this.

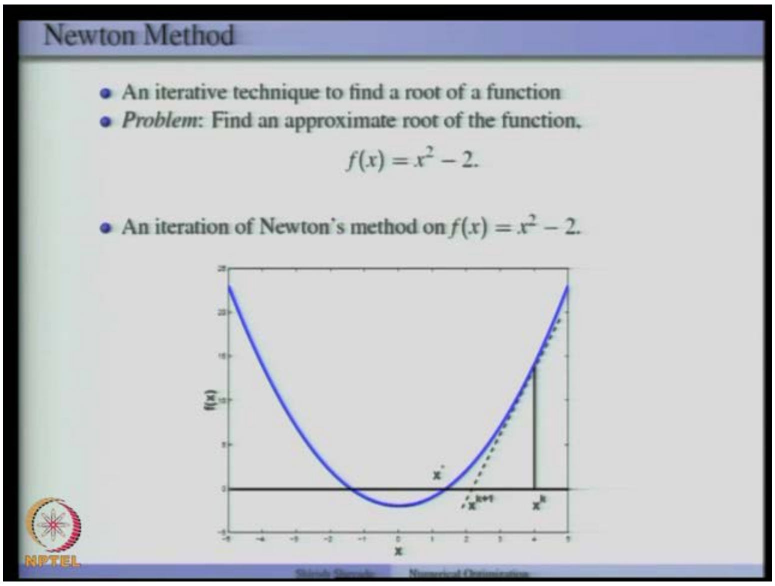
(Refer Slide Time: 50:32)



So, let us go to the some of the properties of the bisection method. So, it requires initial interval of uncertainty. It converges to a minimum point within any degree of desired accuracy. So, like the dichotomous search case, every time the interval length gets reduced by half in the dichotomous search case. It was almost close to half. Where?

Here, it is every time; it is half of the previous length. So, based on the desired degree of accuracy, one can reach the minimum point. Now, let us look at Newton method.

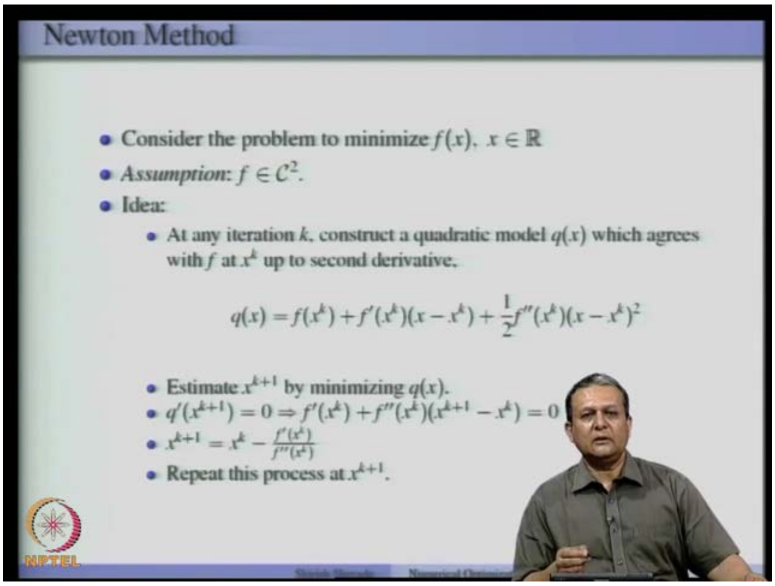
(Refer Slide Time: 52:05)



This is the iterative technique to find a root of a function. You might have studied it in earlier classes. Let us look at some simple problem, where we want to find an approximate root of the function. So, root of a function is a point where the function crosses the x axis. Now, there could be multiple points or multiple roots or a function. But, suppose that we are interested in finding one particular root. So, you might have used this method earlier. So, suppose, we start with a point which is x k, at x k, we draw a tangent to the function and where this tangent hits the x axis, that point will be our new point x k plus 1.

Now, at x k plus 1, we follow the same procedure that we draw a tangent to a function at x at this point and then again, do this. Finally, we will reach the point x star. Now, this function has another root, which is here. So, if we suppose start from a point here, we may end up in this root. So, depending upon the initial point, your root, your functions, the the root obtained using Newton's method would change.

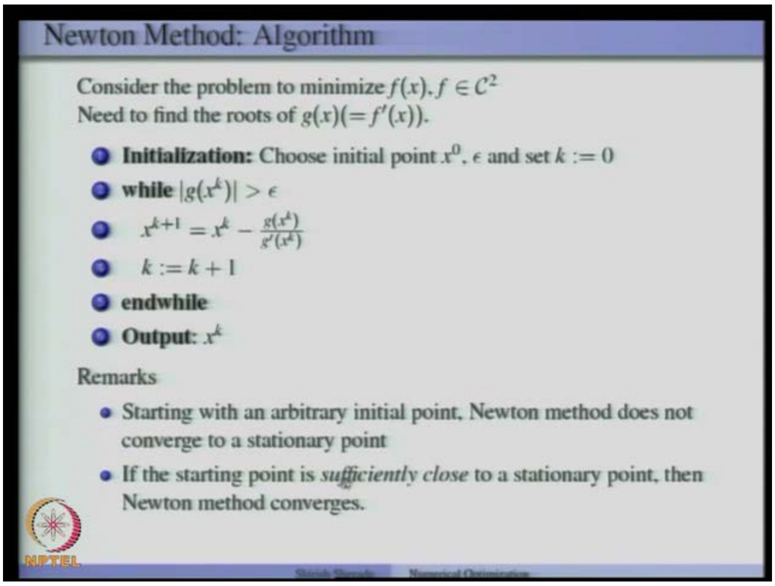
(Refer Slide Time: 53:26)



Now, how do we use this method to find the minimum of a function? So, let us assume that the function is twice continuously differentiable. Then the idea of the Newton method is very simple. So, at every iteration k, you approximate a function b by a quadratic function. So, q x is the quadratic approximation of f at x k. So, that is obtained using the approximate using the Taylor series. But, Taylor series is up to second order. Remember that this is not the truncate at Taylor series. We are using f 2 dash x k, so this is a quadratic approximation of the function.

So, we estimate the new point of the iteration by minimum, finding the minimum of quadratic. Minimum of quadratic is easy to find. We just take the derivative and equate to 0. That will give us the minimum. So, here is a derivative q dash x k plus 1 is nothing but that is equated to 0. That gives us x k plus 1 is equal to x k minus f dash x k by f 2 dash x k. Now, the same procedure is again repeated that you go to a new point, again find the quadratic approximation of the function and go to the minimum of that quadratic approximation and repeat. So, this is the Newton method for finding the minimum of a function.

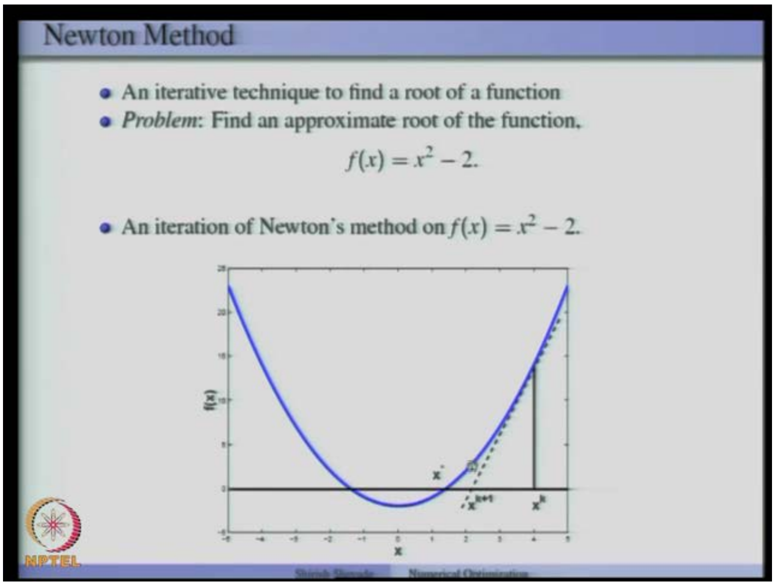
(Refer Slide Time: 54:56)



So, it is very simple that we start with some initial point, as long as the gradient is within is greater than the desired epsilon value. So, we just iterate x k plus 1 is nothing but x k minus g of x k. So, remember that we are trying to find the stationary point. So, we have to look at the problem where we want to find the roots of the g x, the gradient of the function rather than the f x we are trying to minimize. So, to get a stationary point, we want to find the root of g x. This is an algorithm to find the root of g x.

Finally, you will get x k, the n point. Now, this entire procedure works provided the second additive of the function f is the first derivative of function g is positive. Then as you will see now that Newton method depends on the initial point. So, we are sufficiently close to the solution. Then the Newton method will converge. Now, we will see some evaluation Newton method on different functions.

(Refer Slide Time: 56:09)

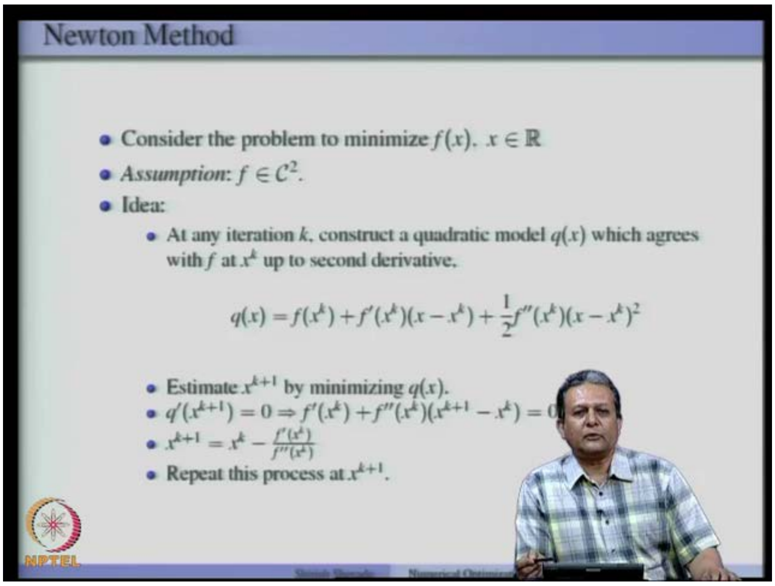


So, Newton method as all of you know, you must have used it to find the roots of a function. It is an iterative technique to find the root of a function. So, suppose if you consider a problem that we want to find an approximate root of the function f x equals to x square minus 2. Then what we do is that we use Newton method to solve this problem. So, we start with some point x k, which is shown here. At x k, you take a tangent to the curve at x k and where the point at which the tangent meets, that becomes your new point x k plus 1. At x k plus 1, again, you follow the same procedure. Again, draw a tangent to this curve. The point at which it intersects the x axis will be your x k plus 2 and so on.

So, finally, you will see that as the iterations progress, you will move to the point, which is x star. It is the one of the roots of this function. Now, if if we start from a point somewhere in this region, then we may end up in getting a root. It is this. So, remember that this function has 2 roots. One is located on the on the positive side. One is located on the negative side.

Depending upon what our initial point is, we will end up going to the closest root. So, this is the Newton method and that can be used to minimize a function. So, when you want to minimize a function, we basically want to find out the stationary points. So, the points where f dash x or g x is 0, we are interested in finding the roots of g x when we want to minimize f of x.

(Refer Slide Time: 58:09)



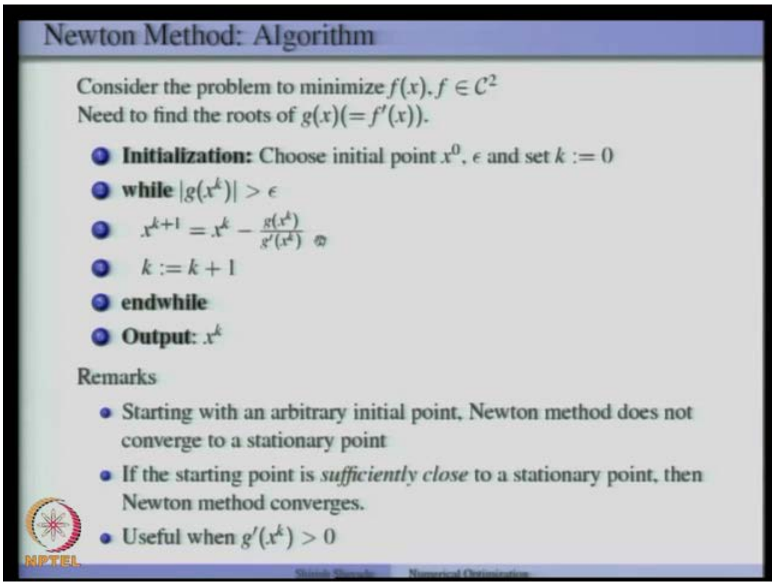
So, let us see how to do that when we use Newton method. So, let us consider the problem to minimize f of x where x is value from the set of real numbers. Now, the important thing that we have to assume is that f belongs to C 2. So, the second derivatives of f are continuous or f is twice continuously differentiable. Now, given that the Newton method is just a simple idea that at every iteration, it tries to construct a quadratic model of a function, which based which agrees with the function at x k up to the second derivative.

So, at x k, if we can find out f dash x k and f 2 dash x k, then we can construct a quadratic model of the given function at x k. Now, given this quadratic model, then the Newton method simply finds out the minimum of this quadratic function. So, to find minimum of this quadratic function, what we have to do is that we have to take the derivative of this with respect derivative of to x with respect to x equated to 0.

We estimate the new point x k plus 1. So, 2 dash x k plus 1 if we calculate, you will see that q dash x k plus 1 is nothing but f dash x k plus f 2 dash x k into x minus x k. Now, at x k plus 1, this derivative of q should vanish. So, that means that f dash has k into f 2 dash x k f dash x a plus f 2 dash x k into x k x k plus 1 minus x k is 0. So, from this, we can get a formula for x k plus 1.

So, given x k, given the first derivative of the function, the second derivative of the function at x k, then x k plus 1 can be calculated as x k plus 1 nothing but x k minus f dash x k divided by f 2 dash x k. Of course, we have to assume that f 2 dash x k is non 0 in this case so that this division makes sense. Now, this procedure is repeated at x k plus 1. Again, we form a quadratic model at x k plus 1 and so on till some stopping criterion is reached.

(Refer Slide Time: 1:00:33)



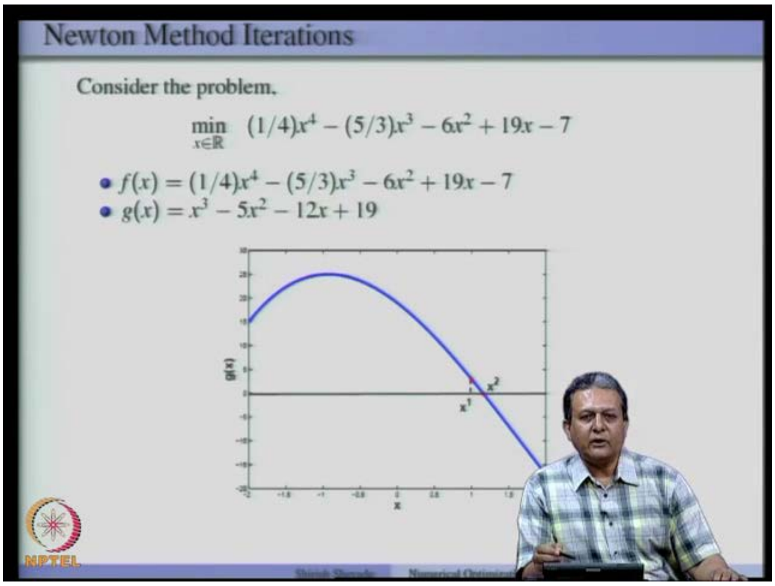
So, here is the simple algorithm called Newton method. Remember that, we are trying to minimize f of x where f is f belongs to C 2.We are interested in finding the roots of g x and g x is nothing but f dash x. Now, the first step in the Newton method is to initialize the point x 0. So, one can randomly choose the initial point x 0. Then choose some epsilon, which is going to be used for deciding the stopping criteria and set the iteration number to 0.

Now, while the gradient of the function at x k, the absolute value of the gradient is greater than epsilon. So, that is why, this epsilon is used. So, one does another iteration of Newton method, where one finds x k plus 1 based on x k and f dash x k. It is nothing but g of x k and f 2 dash x k. Then the iteration counter is increased and the whole process is repeated till the absolute value of the gradient at x k is less than or equal to epsilon and the output of this algorithm. When the algorithm terminates, what we get is x k and f of x k is the optimal of objective function value.

Now, the algorithm looks very simple, but a lot depends on your initial point as we will see some examples now. If the initial points are far away, then this method will Newton method will not converge. The second point that is important is that the g dash x k should always be positive, only then the minimization will make sense. If g dash x k turns out to be 0, then this division is not possible and g dash x k turns out to be negative. Then we may not be able to solve this problem minimize f of x.

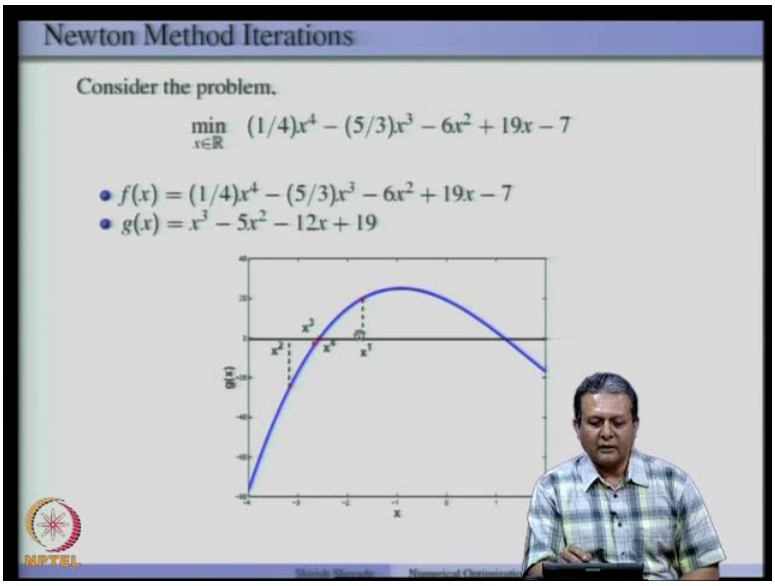
So, one has to be careful about these 2 points. Some remarks that if we start with an arbitrary initial point, the Newton method does not converge to a stationary point. So, this is the important remark. If we start sufficiently close to a stationary point, then the Newton method converges. This method is useful only when g dash x k is greater than 0. That means that the the curve has a positive curvature at every point. Now, let us see how the Newton method works on different problems.

(Refer Slide Time: 1:03:16)



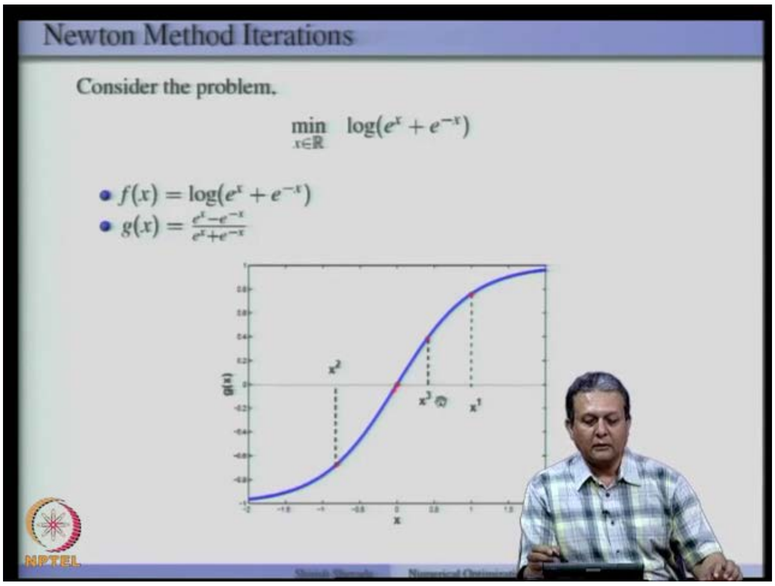
So, let us take the same problem that we considered earlier to minimize a function, which is shown here. Its derivative g x that is f dash x is also shown here. We are interested in finding the roots of this function g x. Now, here is the graph of that function g x. You will see that one of the roots is of that function is somewhere here. Now, if I if we choose x 1 somewhere in the vicinity of that route, you will see that from x 1 b star, we go to x 2 which is very close to the root here and then so on. So, after a few iterations, the algorithm will converge to the actual x star. Now, for the same function, you start with a different initial point.

(Refer Slide Time: 1:04:07)



So, earlier we started with somewhat close to this root. Now, suppose if you start with close to this root. So, suppose our x 1 is here. One can check that from x 1. Then next point x 2 is somewhere here; then x 3, x 4 and so on. Then finally, 1 converges to the x star. So, the important point is that depending upon the initial point x naught, the solution that we would get would differ.

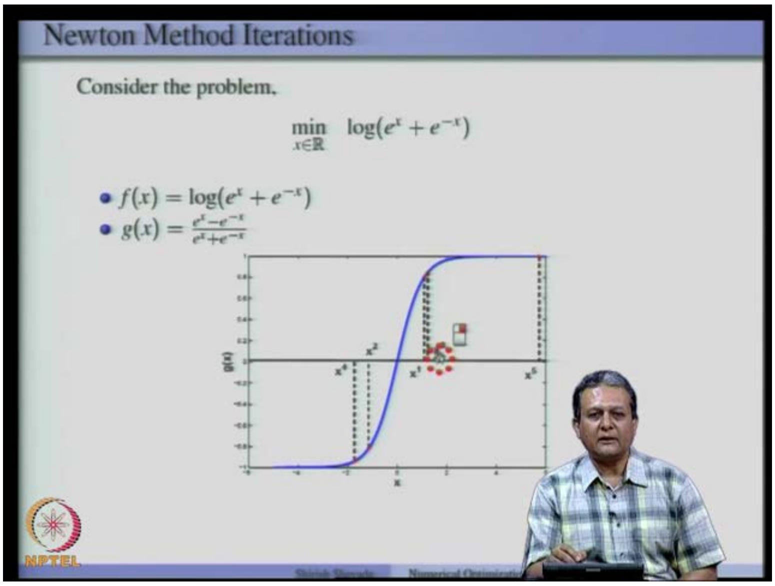
(Refer Slide Time: 1:04:43)



Now, suppose that we consider a function, which is to minimize log of e to the power x plus x to power minus x. If you write the derivative of that, it turns out to be e to the power x minus x to the power minus x divided by e to the power x plus e to the power minus x. So, remember that this function is twice continuously differentiable. So, we can use Newton method. Now, if we plot the graph of g x, so it would look something like this. Suppose, we start from a point x 1, then x 2 point is somewhere here.

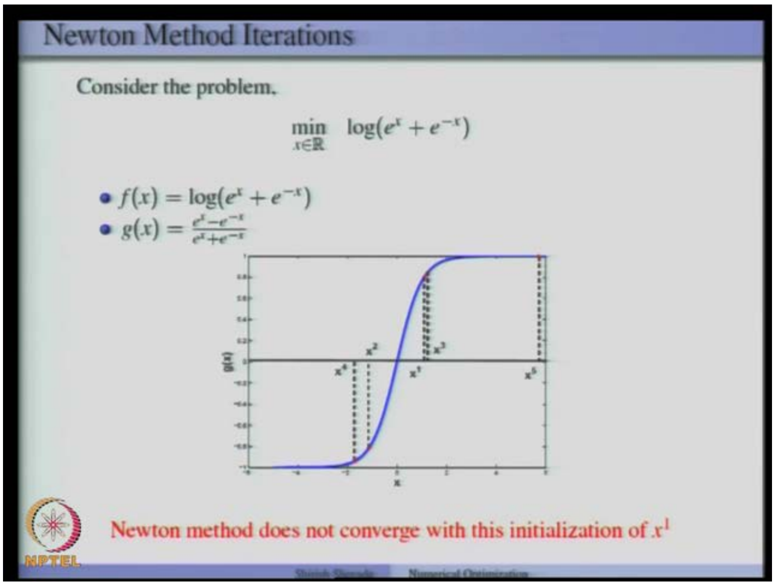
So, if you take a tangent to this curve at x 1, it is the x axis. At x 2, at x 2, we again take the tangent to this curve. It cuts the x axis at x 3 and so on. So, you will see that slowly this method will converge to the point x star where g of x becomes 0. Now, this is the only root of this function g x and a, because as x increases, it goes further up and as x decreases, it goes down. So, it never touches the x axis at any point of time.

(Refer Slide Time: 1:06:06)



Now, let us again consider the same function. But, start with a different initial point. So, our initial point is supposing somewhere here. We are talking about the same function f of x and then the same g x. Now, if our initial point x 1 is here, then take a tangent to the curve g x at x 1. So, it will meet the x axis at x 2 and we repeat the procedure. So, x 2 is here. Then x 3 comes on the right side of x 1.

(Refer Slide Time: 1:06:43)



x 3 becomes on the right side of x 1. Then x 4 is again on the left side of x 2. Now, you will see that x 2, x 5 has gone beyond much beyond x 3. Now, the algorithm does not have any hope to converge to this point. So, you will see that at every iteration, because our initial point was such that it tended to move away from the stationary point. You will see that after x 5, it will be difficult to again come back to this. So, this algorithm will not converge if we start with this particular x 1. So, Newton method is very sensitive to the initial point. If we start sufficiently close to the solution only then there is a chance that it will converge.

So, with this I will end up my discussion on 1 dimensional unconstrained optimization algorithms. Later on, we will come back to multi dimensional unconstrained optimization problems, where these 1 dimensional unconstrained optimization algorithms are useful. But, before we move on to that, let us start with a new topic, which are convex sets.