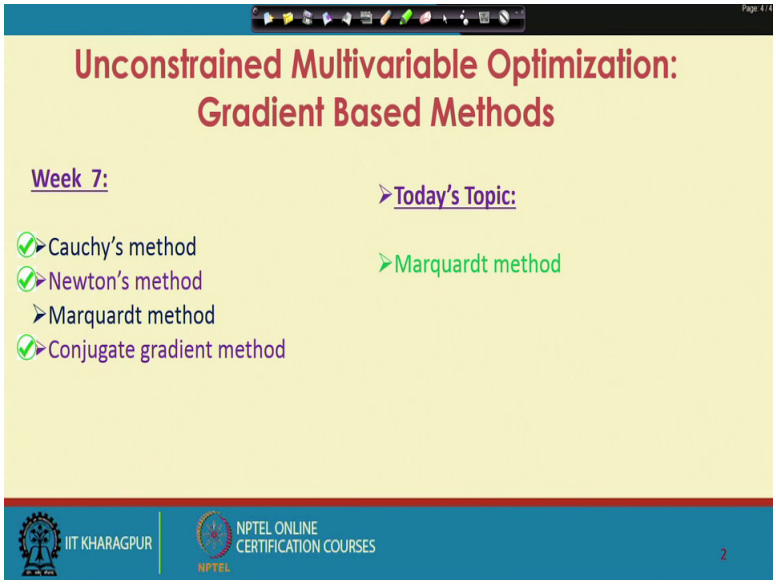
**Optimization in Chemical Engineering**
**Prof. Debasis Sarkar**
**Department of Chemical Engineering**
**Indian Institute of Technology, Kharagpur**

**Lecture - 34**
**Unconstrained Multivariable Optimization: Gradient Based Methods (Contd.)**

Welcome to lecture 34. This is week 7 and we are talking about gradient based methods for Unconstrained Multivariable Optimization. As of now we have talked about Cauchy's steepest descent method conjugate gradient method and Newton's method. Today we will learn about Marquardt method.

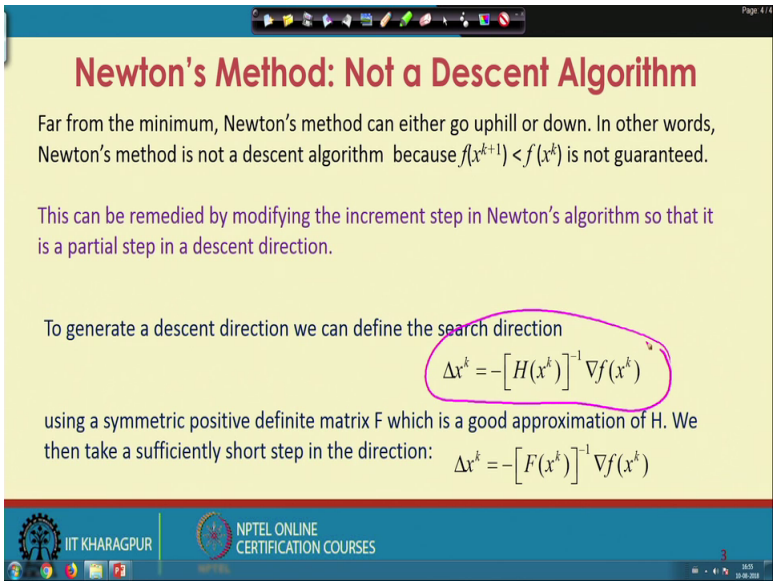
You have seen that steepest method is very useful when you start from a point which is very far from the two optimal point whereas, Newton's method when we start from very close to the optimum point the convergence is very rapid. But, far from minimum point or the optimal point the Newton's method may either converge or diverge. So, if I can combine the Cauchy's steepest descent method and Newton's method I will have an improved algorithm compared to classical Newton's method. So, this is what we will see today in Marquardt method.

(Refer Slide Time: 01:34)




So, before that let us first talk about few disadvantages of Newton's method or classical Newton's method.

(Refer Slide Time: 01:44)

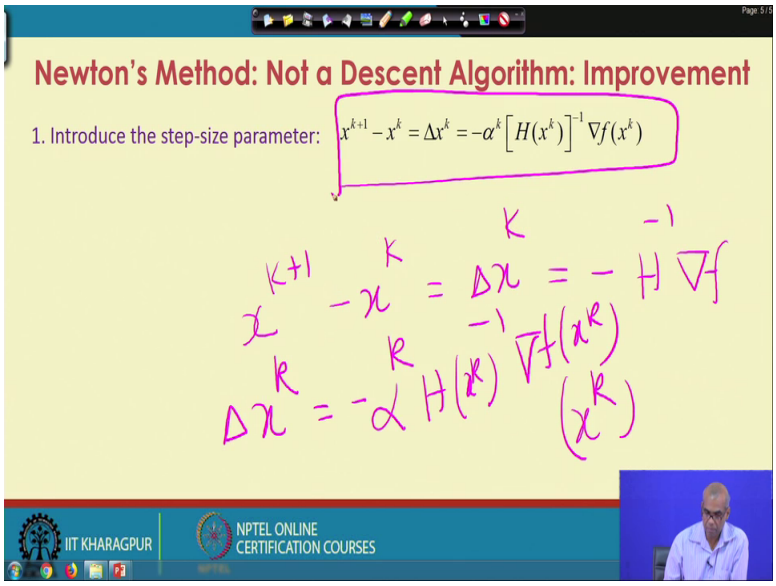



Newton's method is not a descent algorithm, what I mean by that is far from the minimum Newton's method can either go uphill or down. In other words Newton's method is not a descent algorithm because the function value at k plus 1 iteration will not be necessarily less than the function value at kth iteration. So, descent is not guaranteed. However, this can be remedied by modifying the increment step in Newton's algorithm so that it is a partial step in a descent direction.

To generate a descent direction we can redefine the search direction in classical Newton's method using a symmetric positive definite matrix F which is a good approximation of the Hessian matrix H, then we can take a sufficiently short step in that direction. So, what is meant is, if you look at this, this is the expression for increment state when you use classical Newton's method.

Now, if I replace this Hessian matrix by a symmetric positive definite matrix which is a good approximation of the Hessian matrix, and then take a short step in the direction f inverse into gradient of f both evaluated at the current estimate we can generate a descent direction.

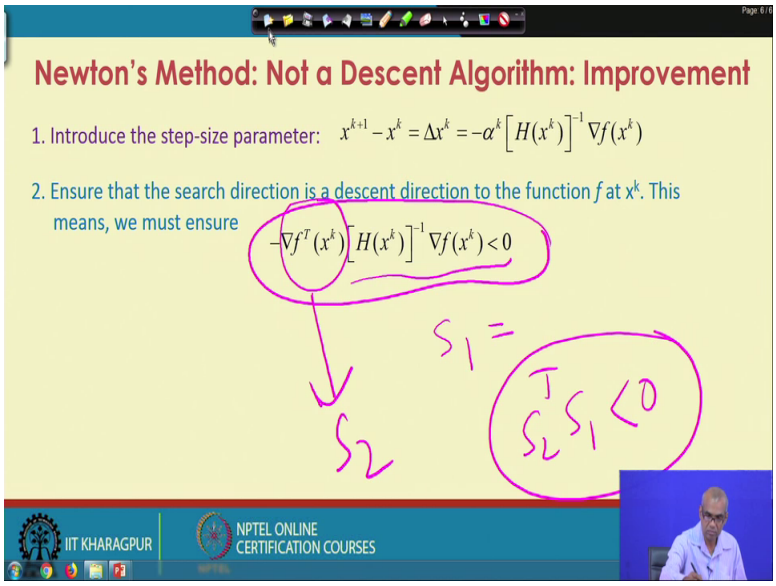
(Refer Slide Time: 03:59)




So, we can modify the Newton's method by introducing the two following features, one is we introduce the step size parameter alpha. So, the classical Newton's method has the increment state as x k plus 1 minus x k which you say as delta x k is minus Hessian inverse into gradient f not both Hessian and gradient of the f is evaluated at x of k.

Now, if I introduce a step length parameter my delta x k can be written as minus alpha k Hessian at x k inverse gradient f x k so this. So, we can introduce a step length parameter in the classical Newton's increment step to help in obtaining descent that means the function value decrease as we go from one iteration to another.

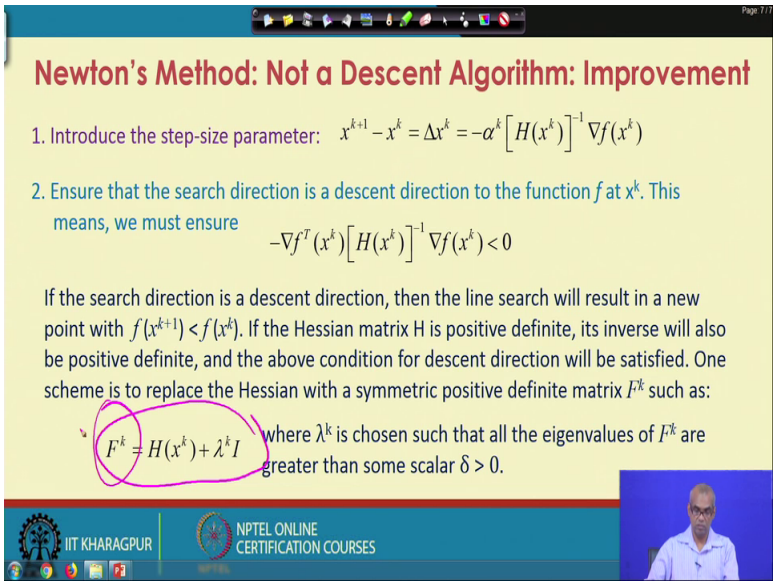
(Refer Slide Time: 05:42)




To ensure that the search direction is a descent direction to the function f at current point x k we have to ensure that this relationship is valid. What is this relationship? If we know if we remember that two search relations let us say S 1 and S 2. So, this let us consider as S 1 and this if we call as S 2 the minus side is included let us say in S 1 which is the search direction for the classical Newton's increment step, then if I call this S 2 and the other one is S 1.

So, S 2 transpose S 1 will be less than 0. So, that we have descent. So, here the gradient f transpose of that into minus of H inverse and gradient f should be less than 0. So, that the descent is ensure.

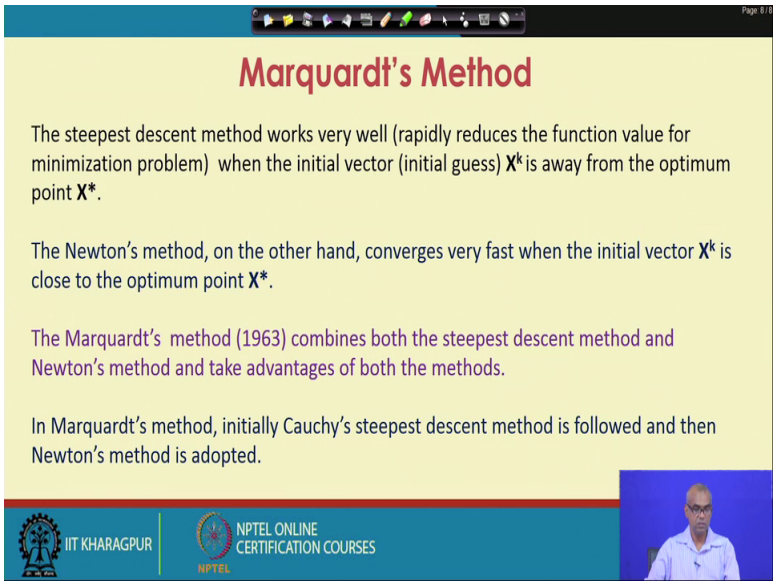
(Refer Slide Time: 07:35)




If the search direction is a descent direction then the line search will result in a new point where the function value at k plus  iteration will be less than function value at kth iteration. If the Hessian matrix H is positive definite its inverse will also be positive definite and the above condition for the descent direction will be satisfied.

One scheme is to replace the Hessian with a symmetric positive definite matrix F k such that F k is Hessian plus some constant value lambda into identity matrix of the same size as Hessian matrix. Here we choose lambda such that all the eigenvalues of the matrix F are greater than some scalar delta which is greater than 0 that means, all eigenvalues of the matrix F will be positive in that case the matrix F will be a positive definite matrix. So, if we have a the positive, if you have a symmetric positive definite matrix F the descent is ensure.

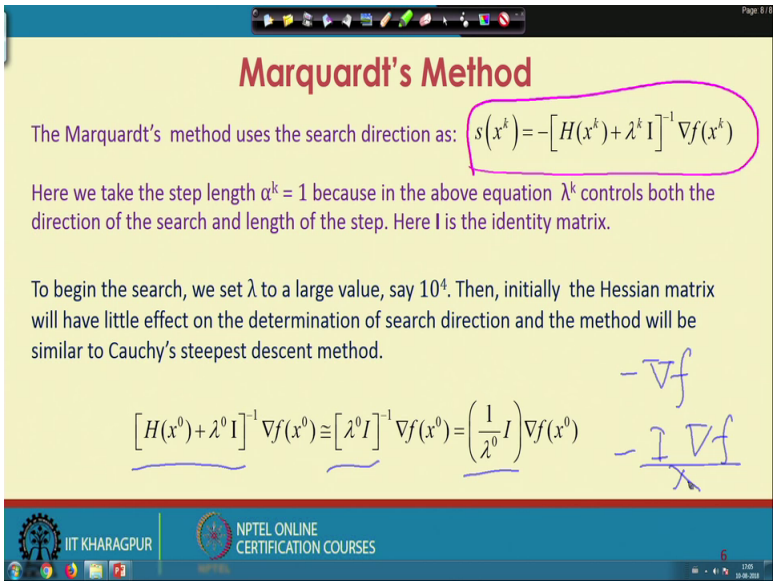
(Refer Slide Time: 09:08)




Now, let us see how Marquardt's method improves the classical Newton's method. The steepest descent method works very well that means it rapidly reduces the function value for minimization problem when the initial vector is away from the optimum point. The Newton's method on the other hand converges very fast when the initial vector is close to the optimum point.

The Marquardt method combines both the steepest descent method and Newton's method and take advantages of both the methods, it was introduced in year 1963. In Marquardt's method initially Cauchy's steepest descent method is followed and then Newton's method is adopted. So, the idea is when we are far from the optimal point we use Cauchy's steepest descent method so that we can rapidly go somewhere near the optimum point and after a few iterations when we can expect that you are close to the true optimum point we switch over to Newton's method.

(Refer Slide Time: 10:40)



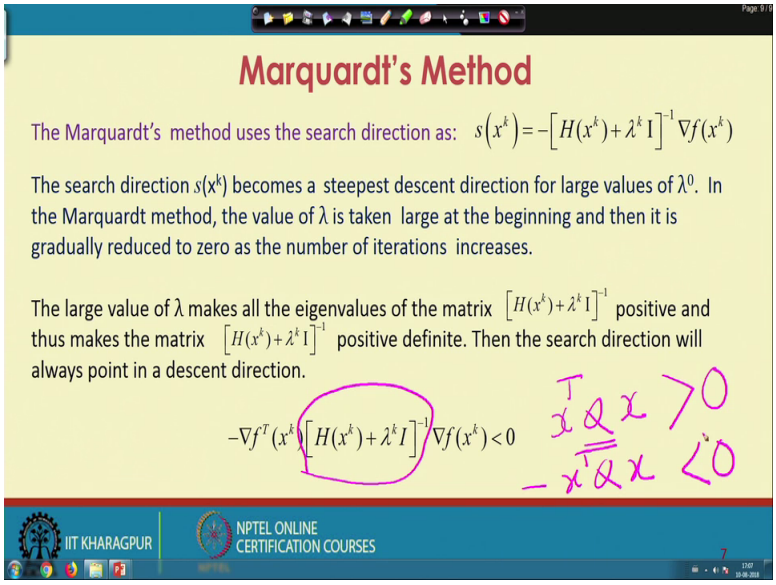

So, in case of Newton's method the search direction is modified when you compare it with the classical Newton's method. Note that in case of classical Newton method the search direction was minus Hessian inverse into gradient of f evaluated at the current estimate x k. In case of Marquardt's method the search direction is minus of Hessian plus lambda into I, where I is the identity matrix you take inverse of that and multiply with the gradient of f which is objective function. So, basically instead of Hessian we take Hessian plus lambda into I, where I is the identity matrix and lambda initial is chosen as a large value and then with iteration with slowly decrease it to 0 value.

In case of Marquardt method we take the step length alpha equal to 1 because if you look at the expression for the search direction, lambda controls both the direction of the search and the length of the step. Note that the identity matrix I will be of same size as the Hessian matrix. So, that H plus lambda I is defined.

To begin the search we said the parameter lambda to a very large value say 10 to the power 4, then initially the Hessian matrix will have little effect on the determination of search direction and the method will be similar to Cauchy's steepest descent method. Why? Because if I take lambda very large then H plus lambda I inverse will be approximately equal to lambda I inverse, which is equal to I by lambda 0 I by lambda.

So, in case of steepest descent method the search direction was minus of gradient, but here what is happening is minus of I gradient of f by lambda. So, basically in the beginning the search direction is similar to Cauchy's steepest descent method.

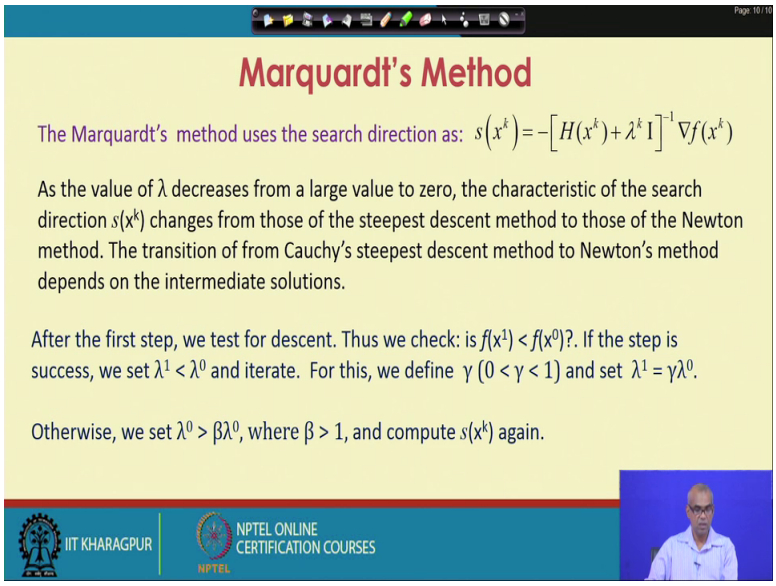
(Refer Slide Time: 14:01)




So, the search direction becomes a steepest descent direction for large value of lambda. In the Marquardt method the value of lambda is taken large at the beginning and then it is gradually reduced to 0 as the number of iterations increases. The large value of lambda makes all the eigenvalues of the matrix H plus lambda I inverse positive and thus makes the matrix positive definite, then the search direction will always point in a descent direction.

Note that, if this is positive definite it is something similar to x transpose Q x where Q is positive definite. So, x transpose Q x will always be greater than 0, but I have minus of x transpose Q x. So, it will always be less than 0 that means, the direction is always point in a descent direction that means, in the direction where function value is decreasing.

(Refer Slide Time: 15:48)



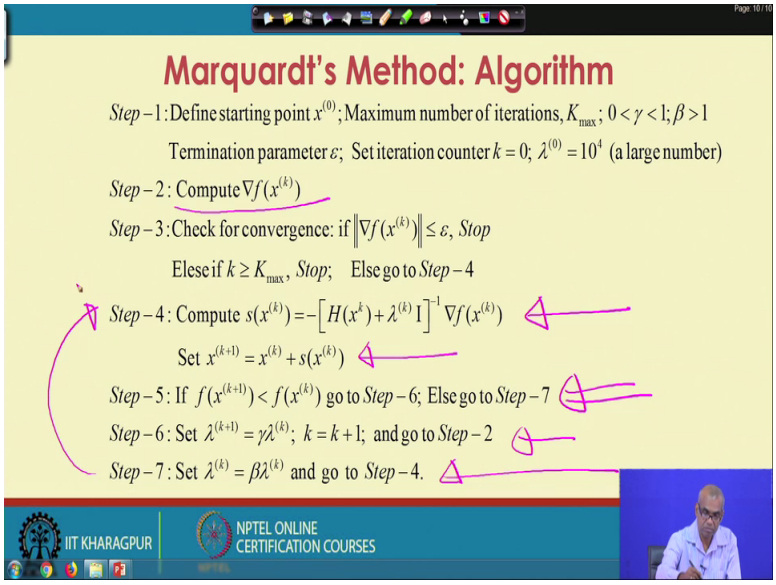

As the value of lambda decreases from a large value to 0, the characteristic of the search direction changes from those of the steepest descent method to those of the Newton's method. This is easy to understand because of the search direction in case of Marquardt's method is minus H plus lambda inverse into gradient of f. So, if lambda equal to 0 this is same as minus H inverse gradient of f. So, in the lambda becomes 0 or very small value the characteristics of the search direction changes from steepest descent method to the Newton's method. The transition from Cauchy's steepest descent method to Newton's method depends on the values of the intermediate solutions.

After the first step we test for descent that means, after the first step we check whether the function value is decreasing. So, if I start from the initial value x 0 and after first step I get the point x 1, I check is the function value f of x 1 is less than f of x 0. If the step is success that means, the function value at x 1 is less than the function value at x 0 we set lambda 1 less than lambda 0, where lambda 0 is the initial starting value of the parameter lambda. So, what it means is if you are moving in a descent direction I decrease the value of the parameter lambda and then iterate further. So, to do that we can define a parameter gamma let us say gamma which is in between 0 to 1, let us say 0.5 and set lambda 1 as gamma into lambda 0.

If the function value at x 1 is not less not less than function value at x 0 that means, we are not moving in a descent direction, we will increase the value of the parameter

lambda. So, for that we define another parameter beta greater than 1 and set lambda 0 greater than beta into lambda 0 and then we compute the search direction H minus H plus lambda I inverse gradient of f again.

(Refer Slide Time: 18:45)




So, this can be understood clearly now from this algorithm. In the first step we define the starting point x 0 maximum number of iterations k max the parameter for reducing the value of lambda gamma which is between 0 and 1. The parameter for increasing the value of parameter lambda which is beta, beta greater than 1, a termination parameter epsilon we set the iteration counter k equal to 0 and set lambda 0 that means, lambda initially a larger number let us say 10 to the power 4.

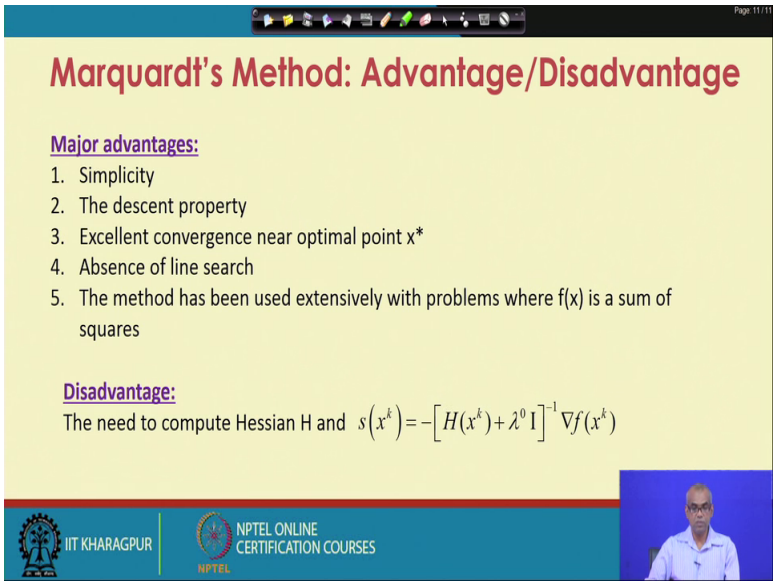
So, the first step is basically initialization where we set values of all required parameters. In the step 2 we compute the value of the gradient at the current estimate. So, initial it will be gradient of f evaluate at x 0. In general for kth iteration we are writing compute gradient of f at x k. At this table it has check for convergence by checking the norm of the gradient, if the norm of the gradient is less or equal to the termination parameter what is a very small value may be 10 to the power minus 4, we stop. We also stop if k that means, number of iterations has reached the maximum allowable iterations otherwise we will go to step 4, in the step 4 we compute the search direction.

So, in the step 4 we compute the search direction which is minus of Hessian plus lambda I inverse into gradient of f. Once we have the search direction s for the current estimate x

k we will follow the usual update rule as x k plus 1 equal to x k plus s of x to the power x k. Now, check for descent. So, if the function value at k plus x k plus 1 is less than f of x k we will go to step 6, where we decrease the value of the lambda as lambda k plus 1 equal to lambda into gamma into lambda k remember gamma is less than 1. So, the lambda at at k plus 1th iteration is lower than the lambda at kth iteration, we set k equal to k plus 1 and go to step two where again we compute the gradient.

If descent is not obtained; that means, the function value at x k plus 1 is not less than the function value at x k, we increase the parameter lambda as lambda k equal to beta into lambda k beta greater than 1 and then go to step 4 where we again compute the search direction. After computing the search direction we will again check whether the descent is obtained or not, until descent is obtained we will increase the lambda value. So, this way we iterate until convergences achieved that means, the norm of the gradient will be very small that means, the magnitude of the gradient will be very small or the number of iterations will exceed the allowable maximum number of iterations.

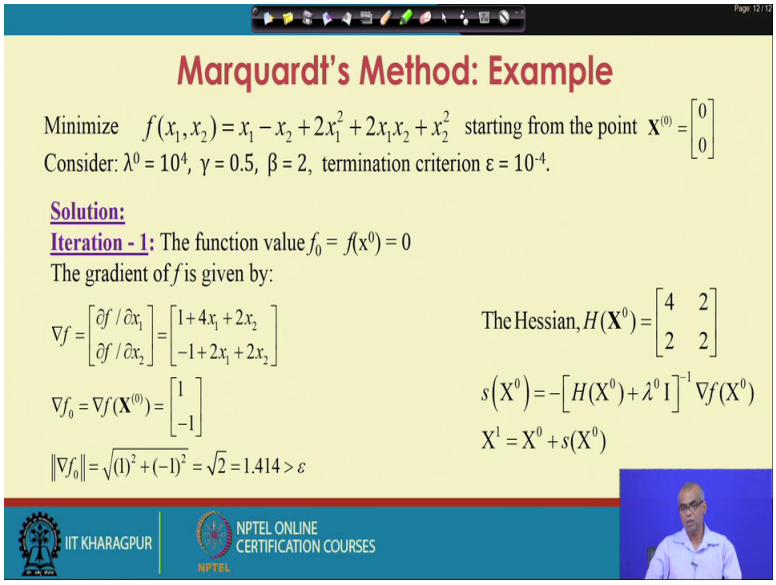
(Refer Slide Time: 23:43)




Here are some advantages of Marquardt's method. The major advantages are its simplicity. It is a very simple method the descent property. So, decreasing function value is ensured excellent convergence near optimal point because near optimal point it works or behaves like a classical Newton's method. Absence of line search, we are not doing any line search here. This method has been used extensively with problems where the

objective function is a sum of squares. However, this disadvantage is this that we have to compute the Hessian; computation of Hessian particularly inverse of Hessian may be computationally expensive.

(Refer Slide Time: 25:04)

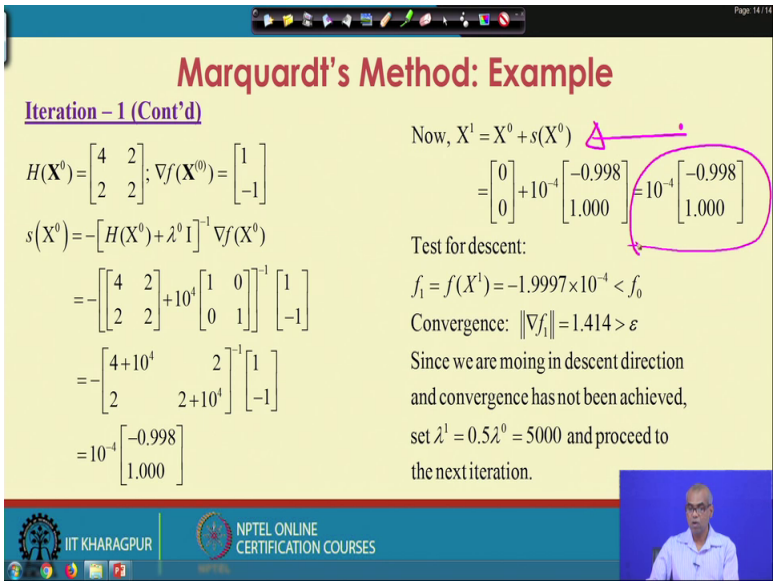



Let us now look at an example. Let us minimize a quadratic function in two variables f of x 1, x 2 equal to x 1 minus x 2 plus 2 x 1 square plus 2 x 1 x 2 plus x 2 square starting from the point 0 0. Let us consider the initial value of the lambda is 10 to the power 2 lambda reduction parameter gamma 0.5 lambda increase parameter beta equal to 2. Termination criteria we set as 10 to the power minus 4 let us say.

So, a given initial guess or initial starting point x 0 as 0 0. So, if I put x 1 equal to 0, x 2 equal to 0 in the given function the function value is 0. The gradient of the f will have two components del f del x 1 del f del x 2 which can be computed as 1 plus 4 x 1 plus 2 x 2 which is del f del x 1 and minus 1 plus 2 x 1 plus 2 x 2 which is del f del x 2. So, the gradient evaluated at x 0 initial starting point is 1 minus 1. The norm of the gradient is square root of 2 which is 0.414 and this is greater than epsilon the termination parameter. So, we have to go for next step.

Let us compute the Hessian is the quadratic function. So, Hessian will be a constant matrix which is obtained as 4 2 2 2. So, the search direction is minus H plus lambda inverse into gradient of f and then we will increment or we will get the next estimate as X 1 equal to X 0 plus search direction at X 0.

(Refer Slide Time: 27:54)

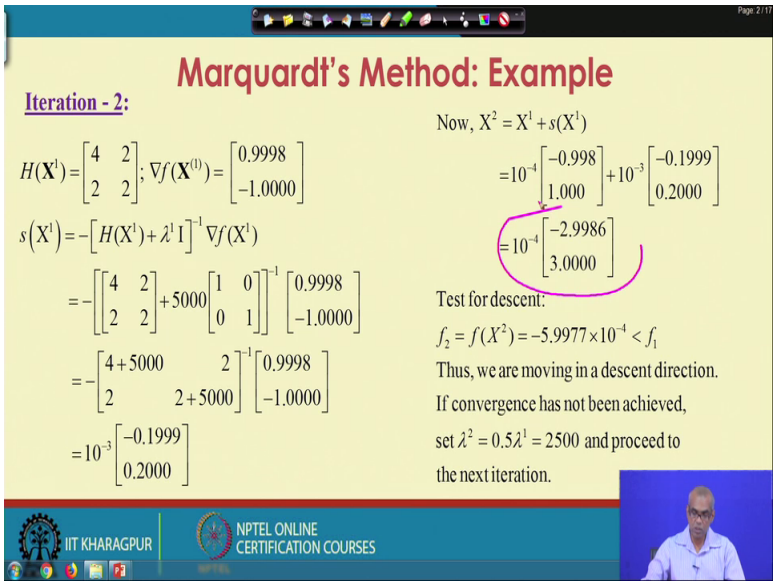



So, it is already obtained as a constant matrix the gradient at x 0 is obtained as 1 minus 1. So, let us compute the search direction which is minus Hessian plus lambda 0 I inverse into gradient of f at x 0. So, if I compute these I get 10 to the power minus 4 into 0.98 and 1. So, I get the search direction vector at X 0.

So, now, I can obtain the next estimate of the optimal point X 1 as X 0 plus s which is search direction at X 0. So, this is obtained as 10 to the power minus 4 into minus 0.998 and 1. At this stage let us test for descent, so function value at X 1 I obtained as minus 1.9997 into 10 to the power minus 4. The function value at X 0 was 0. So, the function value has decreased, so we are moving in a descent direction. If we find out the norm of the gradient at X 1 we obtain it as 1.414 which is greater than the termination parameter.

So, we are moving in a descent direction and the convergence has not been achieved. So, we will decrease the parameter lambda. So, let us set lambda for the next step lambda 1 equal to 0.5 into lambda 0. So, that it becomes 5000 it was 10,000 initially, now it becomes 5000 and proceed to the next iteration.

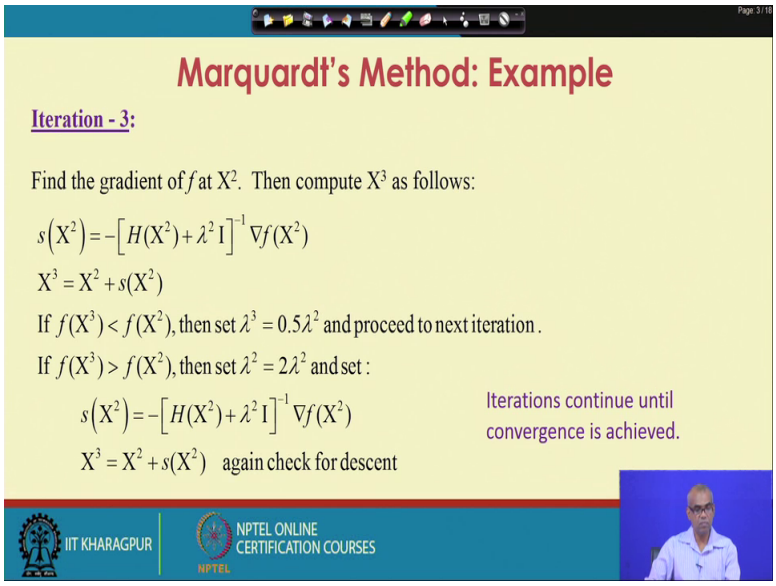
(Refer Slide Time: 30:08)




In the next iteration H or Hessian matrix is already I consider matrix 4 2 2 2. So, compute the gradient at X 1 which is obtained as 0.9998 minus 1. So, again I compute the search direction at X 1 and search direction can be computed as minus Hessian evaluated at X 1 plus lambda 1 I inverse into gradient of f evaluated at X 1. So, if we compute the search direction I get 10 to the power minus 3 into minus 0.1999, 0.2000.

So, once I have the search direction at X 1 I can obtain the next estimate of X which is X 2 and X 2 can be obtained as X 1 plus s of X 1. So, this is obtained as 10 to the power minus 4 into minus 2.9986 and 3. Again you test for descent we see that the function value at X 2 is less than the function value at f 1. So, we are still moving in a descent direction and if the convergence has not been achieved we set lambda 2 is 0.5 lambda 1, so it was 2000, it was 5000 in the previous step now it will be 2500. So, we proceed to the next iteration.

(Refer Slide Time: 31:56)




In the iteration 3, again we find the gradient of the objective function f at X 2 then compute X 3 as X 3 equal to X 2 plus s at X 2. If the function value at X 3 is less than the function value at X 2 will set lambda 3 as 0.5 into lambda 2 and proceed to next iteration. If the function value at X 3 is greater than the function value at X 2 then we will set lambda 2 as 2 into lambda 2 that means we will increase the lambda value. Remember that we have set beta equal to 2 and then we can find the search direction as minus H at X 2 plus lambda 2 into identity matrix I inverse of that into gradient of f at X 2.

So, then X 3 will be obtained as X 2 plus s of X 2, again we check for the descent. So, this way iterations will continue until convergence is achieved, and the convergence will be achieved when the norm of the gradient will be very small or the maximum number of iterations will be achieved.

(Refer Slide Time: 33:07)


So, we stop our discussion on Marquardt's method here.