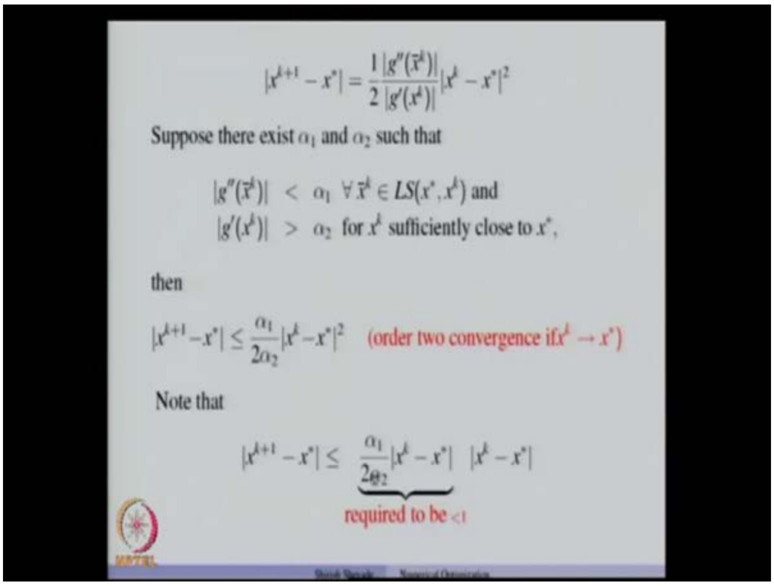
Numerical Optimization

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

Lecture - 15
Trust Region and Quasi-Newton Methods

Hello, welcome back to this series of lectures on numerical optimization. In the last class we started looking at convergence of Newton method, the order two convergence of Newton method. So, let us continue with that proof.

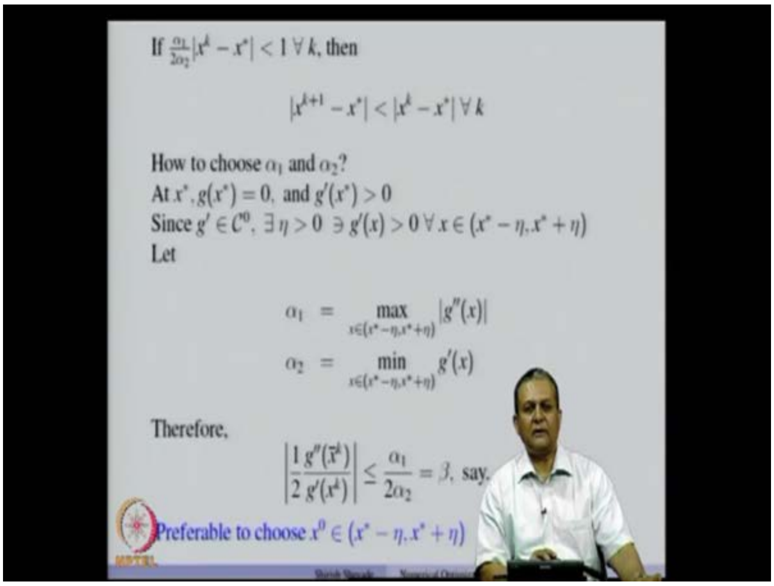
(Refer Slide Time: 00:39)



So, in the last class we showed that if the distance between x k plus 1 and x star is equal to half of mod of g 2 dash x bar k by g dash x k into x k minus x star square. And suppose, if you are able to get some alpha 1 and alpha 2 which both are positive such that the numerator here is less than alpha 1 for all x bar k and then denominator here is greater than alpha 2 for all x k, sufficiently close to x star. Then we can write that mod of x k plus 1 minus x star is less than or equal to alpha 1 by 2 alpha 2 x k minus x star square. Now, if the sequence x k converges to x star, then we can clearly see that this is order two convergence.

Now, let us see how to show that x k does converge to x star. Now, if you look at x k plus 1 minus x star mod of x k plus 1 minus x star and that is less than or equal to this quantity. This quantity can be split into two parts: the one part is alpha 1 by 2 alpha 2 into mod of x k minus x star and the other quantities x k minus x star. Now, in order that the distance of x k plus 1 from x star is less than the distance of x k from x star what we want is this quantity to be less than 1.

(Refer Slide Time: 02:14)



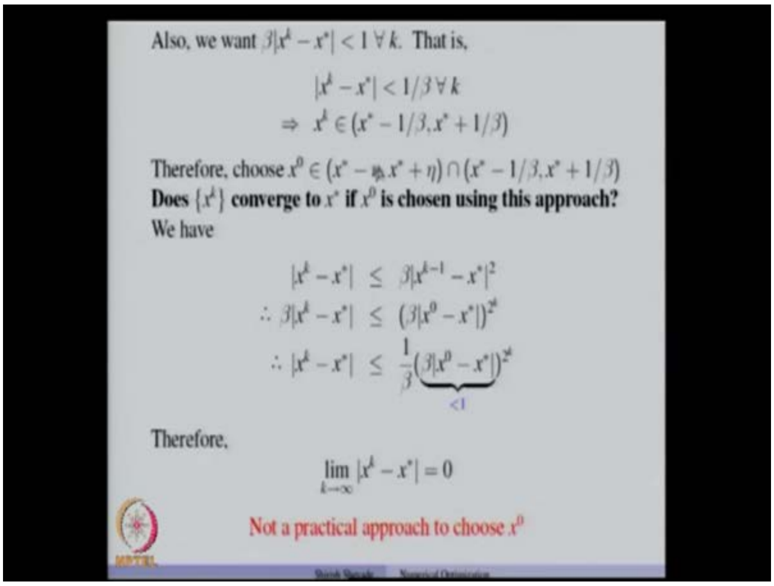
So, in other words if this quantity is less than 1 then we can say that mod of x k plus minus x star is less than mod of x k minus x star. So, that means that the new point is more close to x star than the current point x k and if this happens for every k, then certainly we would converge to x star. Now, the question is that how dowe choose that alpha 1 and alpha 2 which we mentioned earlier. Now, at x star we know that the gradient of the derivative of the function vanishes and then the second order derivative of the function is greater than 0.

So, since we know that f in fact we have chosen f to be thrice differentiable. So, the second derivative of the function f is certainly continues and therefore, there exists some eta which is greater than 0 such that g dash x is greater than 0 for all x in the open interval x star minus eta to x star plus eta that is because of the continuity of g dash and we know that g dash x star is greater than 0 so at least in the neighborhood of x star the g dash has to be greater than 0.

So therefore, if you choose alpha one to be max of mod of g two dash x where x lies in the interval x star minus eta to x star plus eta and alpha two to be min of g dash x where x is again in the same interval. Then what we have is, we have mod of half into g two dash x bar k by g dash x k less than or equal to alpha 1 by 2 alpha 2 for all x bar k in that interval and x k in the same interval.

So, let us call this quantity as beta. Therefore, it important for us to choose x naught to be in the interval x star minus eta to x star plus eta, then we can show that this holds and since, this inequality holds then if x naught is chosen properly then we have this quantity which is less than 1 and then in which the sequence in x k plus one minus x star or the mod of that quantity is less than mod of x k minus x star for all k.

(Refer Slide Time: 04:57)

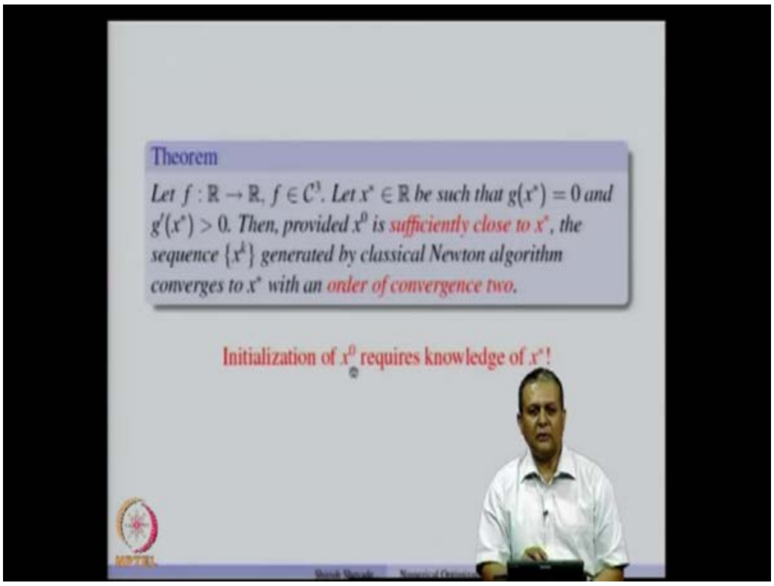


Now, we also want the alpha 1 by 2 alpha 2 into x k minus x star to be less than 1. So, which means that the x k minus x star should be less than 1 over beta for all k or in other words x k should belong to the interval x star minus 1 by beta to the x star plus 1 by beta. Therefore, we have to choose x naught such that, the x naught is in the intersection of x star minus eta to x star plus eta and x star minus 1 by beta to x star plus 1 by beta. And now, if we choose x naught like this, the question is that does x k converts to x star or not? Now, let us look at this expression that we had mod of x k minus x star is less than or equal to beta into x k x k minus 1 minus x star square.

Now, the same quantity can be written. So, x k minus 1 x star can be written in terms of x k 2 minus x star and so on and so for. So, finally x 1 minus x star can be written as less than or equal to beta into x 0 minus x star square. So, if we combine all those expressions and write x k minus x star in terms of x 0 minus x star, then this is what we will get. So, we are multiplied throughout by beta and we get this quantity.

Now, this x k minus x star square is less than or equal to 1 over beta into this quantity and this quantity since, we have seen that beta x k minus x star is less than 1 for all k. So, we also know that beta x 0 into mod of x 0 minus x star is also less than 0. So, this quantity is less than 1. Therefore, mod x k minus x star is less than or equal to 1 over beta into very small quantity as k increases and that quantity will go to 0 as k increases. Therefore, limiters k tends to infinity x k minus x star mod of x k minus x star becomes 0. But, you will notice that this is not a practical approach to choose x naught. For example, to choose x naught we need the knowledge of x star. But, nevertheless this proof shows that if we start Newton method from some point x naught which is sufficiently close to x star, then it converges to x star with order of convergence 2.

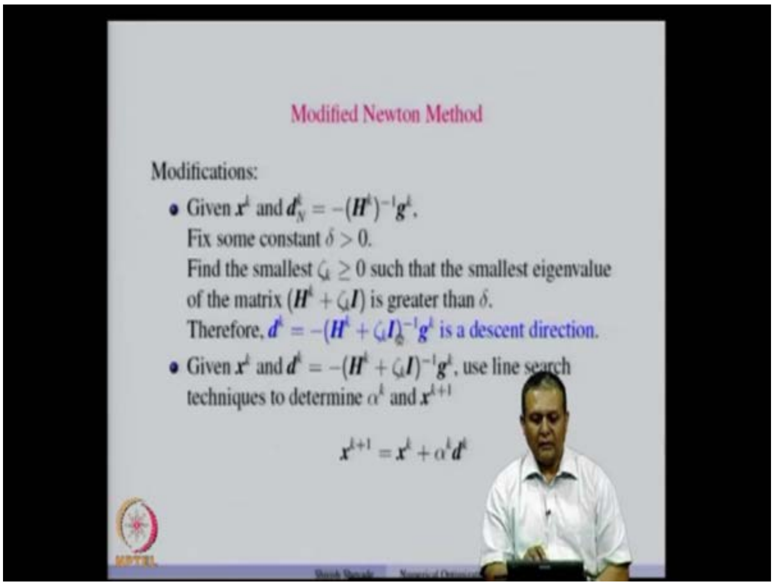
(Refer Slide Time: 07:54)



So, we have a theorem let f be a real valued function and in thrice continuously differentiable. The thrice continues differentiability in needed because we wanted the derivative to be lipschitz continuous. So, that is why we need this and let x star belong to R be such that g of x star is 0, the gradient vanishes and then the second derivative is greater than 0 then provided x 0 is sufficiently close to x star, a sequence generated by classical Newton algorithm that converges to x star with order of convergence 2.

So, this result can be extended to a general case where you we have a function f from R into R. But, the important part is that the Newton method when it starts with the point which is sufficiently close to x star, it does converges to x star with order of convergence two. But, the problem is that the initialization of x naught requires the knowledge of x star and our aim is to minimize f x to get x star. So, this knowledge of x star is not there. And therefore, we cannot initialize x 0 properly so that, we can get global convergence of Newton method. In other words, the Newton method does depend a lot on x 0 a classical Newton method does depend a lot on x 0 and as this theorem says the x 0 requires knowledge of x star which is not available. But, however the important fact is that if you start sufficiently close to x star we do get order two convergence. Now, let us look at some modifications of Newton method which are globally convergent. So, in other words they do not require the knowledge of x star to get the initial point x 0. One can start from any point and converge to the local minimum.

(Refer Slide Time: 10:27)



So, those methods are called modified Newton methods. So, we will see couple of those methods in this course and let us look at the places where modifications for the Newton method are required. Now, given x k and direction Newton direction at the point x k which is nothing but negative of the hessian inverse into g k. Now, there are two possibilities, one is that the hessian matrix is not invertible or is close to singular. So in that case this direction really does not make much sense. Secondly, the hessian matrix is suppose negative definite so in that case this direction is not a descent direction. So, if either hessian matrix is negative definite or hessian matrix is close to singular. Now, how do we modify Newton method so that this expression is positive? The quantity in the parentheses is a positive definite matrix. Now, suppose if you fix some constant delta which is greater than 0 and find the smallest zeta k which is greater than equal to 0 such that the smallest Eigen value of the matrix H k plus zeta k i is greater than delta.

Now, by ensuring that the smallest Eigen value of this matrix is greater than delta which is a positive constant, we are ensuring that H k plus zeta k i is a certainly a positive definite matrix. So, once we find the zeta k for a given H k then we can choose a direction to be minus of the new direction can be chosen as minus of H k plus zeta k i inverse g k and that is going to be a descent direction because H k plus zeta k i is a positive definite matrix. So, this is one modification that one can use.

Now, secondly if you recall that the classical Newton method does not use line search or in other words, it uses alpha k to be 1. Now, instead of that we can if you are given x k and d k which is nothing but minus H k plus zeta k i inverse g k what we can do is that we can use line search techniques to determine alpha k and then x k plus 1 is determine as x k plus 1 is equal to x k plus alpha k d k.

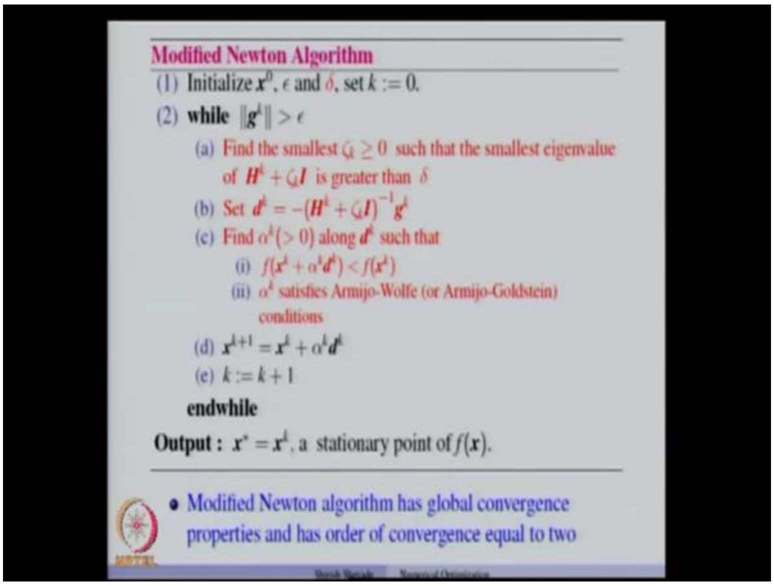
So, the two problems which are associated with classical Newton algorithm one was that the hessian is the hessian matrix at x k is not necessarily positive definite so that, problem is taken care of by ensuring that we add some zeta k i matrix or a constant times identity matrix to H k. So, that the smallest Eigen value of this matrix is greater than delta where delta is positive constant. So, once you ensure that and chose this direction as our new direction for search then we certainly get a descent direction. Now, after having obtain a descent direction the next step is to get alpha k using line search and update x k to get x k plus 1.

Now, if we ensure that this line search techniques satisfy Armijo-Goldstein or in Armijo-Wolfe condition. Then we have all the necessary criteria for global convergence of an optimization algorithm that every time the direction chosen is a descent direction. The function value decreases and there is sufficient decrease in the objective function then certainly the method is going to converge. Now, the question is that how do we get this zeta k? Sometimes, very simple techniques are used. Suppose we start with some reasonable value of zeta k and do the Cholesky factorization of H k plus zeta k i.

Now, if that Cholesky factorization fells because H k plus zeta k i is not positive definite, what one can do is that increase zeta k or may be double the value of zeta k and try the Cholesky factorization again. So, while the Cholesky factorization of H k plus zeta k i is not successful, keep increasing the value of zeta k till that becomes successful. Now, remember that this zeta k has a very important role to play here. Now, if zeta k is very large quantity. Then that dominates this expression H k plus zeta k i and the method behaves like a steepest descent method. So, the order two convergence of Newton method is lost. But, if zeta k is small then H k will be a dominant thing.

So, typically when the quadratic approximation of the function is good, the zeta k value will be typically small and the most dominant term here between the two is H k and therefore, this direction will be close to the Newton direction and if zeta k is very large then the direction becomes close to steepest descent direction. So, whenever the approximation of the quadratic approximation of a given function is good, you will see that zeta k becomes a small quantity and therefore, we have the direction which is close to Newton direction and the convergence will be very fast and that typically happens near the solution. So, initially wherever the initial point x 0 this method because of this the extra addition of this extra term to the hessian matrix. Make sure that the steps are taken in appropriate way so that, the convergence is achieved. So, one can show that this method is globally convergent.

(Refer Slide Time: 17:29)

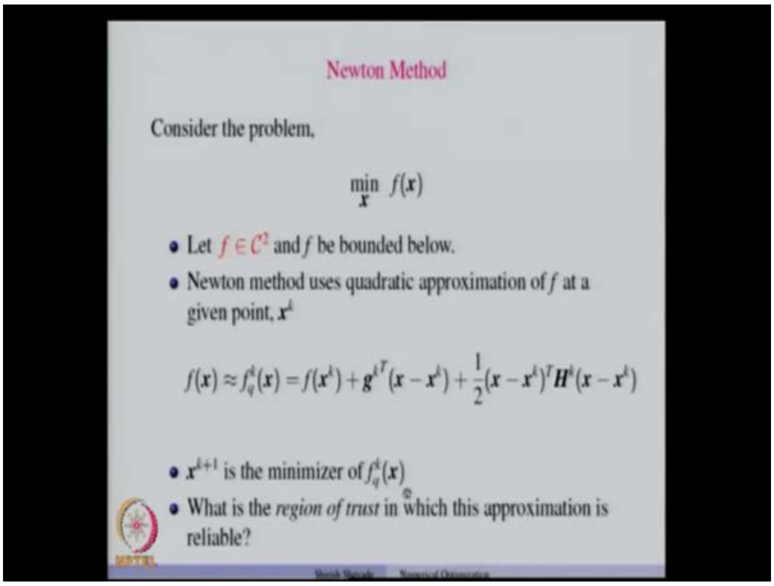


Now, let us look at the algorithm. So, as I mentioned that apart from the initialization of x naught and epsilon what we need is parameter delta and that parameter delta is typically given by the domain expert or it requires the domain knowledge to set that parameter, the iteration counter is set to 0. So, while the norm of the gradient is at the current point x k is greater than epsilon, you find the smallest zeta k such that the smallest Eigen value of x k plus zeta k i is greater than delta. Now, once we do that then the direction that it chosen is minus x k plus zeta k i inverse g k and this direction is certainly in descent direction because H k plus zeta k i is a positive definite matrix. Then we use the line search so that, the function value decreases along the direction d k and alpha k satisfies the Armijo-Wolfe or Armijo-Goldstein conditions.

So, compare now this with the classical Newton algorithm that we have seen. So, this step was not there in the classical Newton algorithm. Here, in the classical Newton algorithm use d k is equal to minus H k inverse g k, then this the step length determination procedure was not there in the classical Newton algorithm because we chose alpha k to be 1 in the classical Newton algorithm and the rest of the steps are similar to the ones that we had earlier that x k plus 1 is nothing but x k plus alpha k d k and then k is equal to k a plus 1 that is the iteration counter is increased and the whole procedure is repeated till norm of the gradient current point x k is less than or equal to epsilon and then we stop and we get a stationery point x k. So, we will not prove result related to this modified Newton algorithm. But, this modified Newton algorithm it does have global convergence properties and has order of convergence equal to 2.

So, the important thing that one should keep in mind is that the initial point x naught, there are no restrictions on initial point x naught. So, unlike classical Newton algorithm where, which is very sensitive to x naught. This modified Newton algorithm is not sensitive to x naught. Now, we will look at some other modification of Newton method also.

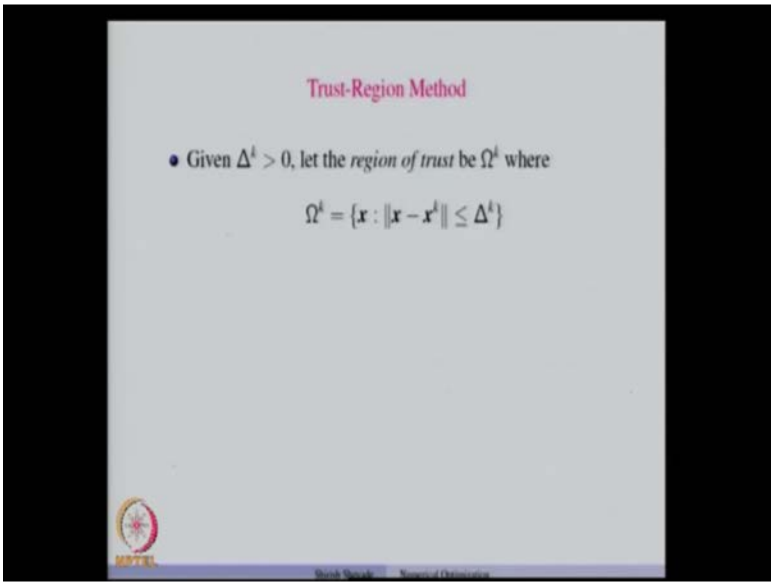
(Refer Slide Time: 20:25)



Let us recall that we are trying to minimize this problem. Minimize f of x and in Newton method, we consider f to be twice continuously differentiable and bonded blow. So, we continue to use this assumption that f is bounded below. Now, the Newton method you just quadratic approximation of a given function f at a given point and x k plus 1 is the minimizer of this quadratic approximation. So, the quadratic approximation uses the first order derivative and second order derivative information and so, this is quadratic approximation is used at a current point x k and x k plus 1 is found as the minimizer of this quantity. Now, the question that one would like to ask is that how good is this quadratic approximation at a given point?

Here, in other words what is the region of trust in which this approximation is reliable. Now, if the quadratic approximation is reliable, then we certainly would like to minimize this quantity and go to the next step x k plus 1. But, if the region of trust is not reliable or the quadratic approximation is not reliable in the region then we would like to reduce the region of trust or reduce a region in which we can approximate f by a quadratic function. So, such methods which use the region of trust of the quadratic approximation are called trust region methods.

(Refer Slide Time: 22:31)



Now, suppose delta k is a positive quantity which is given to us then one can define the region of trust to be omega k and omega k is defined as set of all points around x k which are at a distance at the most delta k from x k. So, this norm is typically a Euclidean norm.

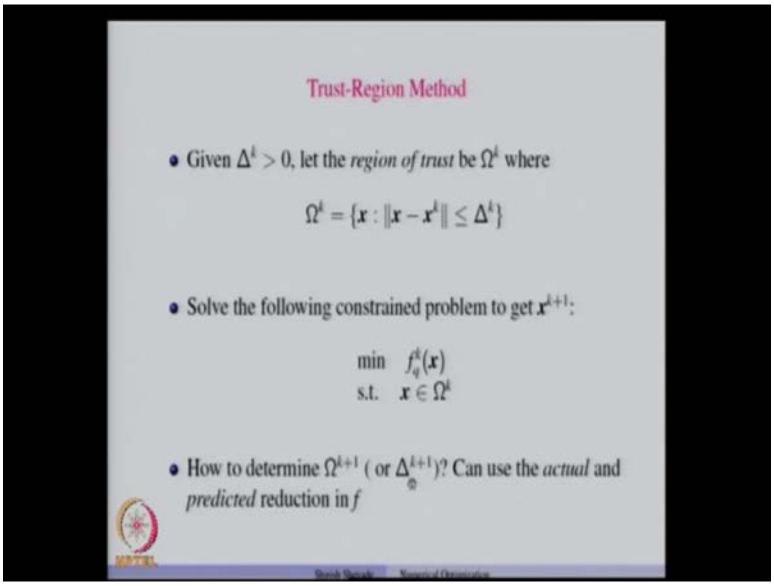
(Refer Slide Time: 23:16)

Now, let us look at an example suppose that we have the function contours like this. And suppose this is our current point and suppose the quadratic approximation of the function like this. Now, if one wants to go to the minimum we simply take a step like this and that may not reduce the objective function or you (( )) it reduces, if we do the line search the reduction in the objective function will be typically small. So, instead what one can do is that in such cases remember that the minimum like somewhere in this direction. So, in such cases what one can do is that, one can construct a region of trust around x k. So, this is x k we can construct a region of trust around x k and then minimize the quadratic approximation that we have got in this region of trust and then go to the new point.

Now, if it so happens that the region of trust is good then if we go to the new point and if we find that the approximation is really good, then we can expand this region of trust. On the other hand, if we realize that when we move from x k to x k plus 1 that function value has actually increased which we do not want, then one can string the region of trust.

So, depending upon how good is the above quadratic approximation of the function at a given point, we can decide whether to expand this region of trust or reduce this region of trust. So, by adopting the size of this region of trust one can move to a new point and that will ensure that we are not going in a direction which is away from the local minimum. So, let us see how to do that.

(Refer Slide Time: 25:58)



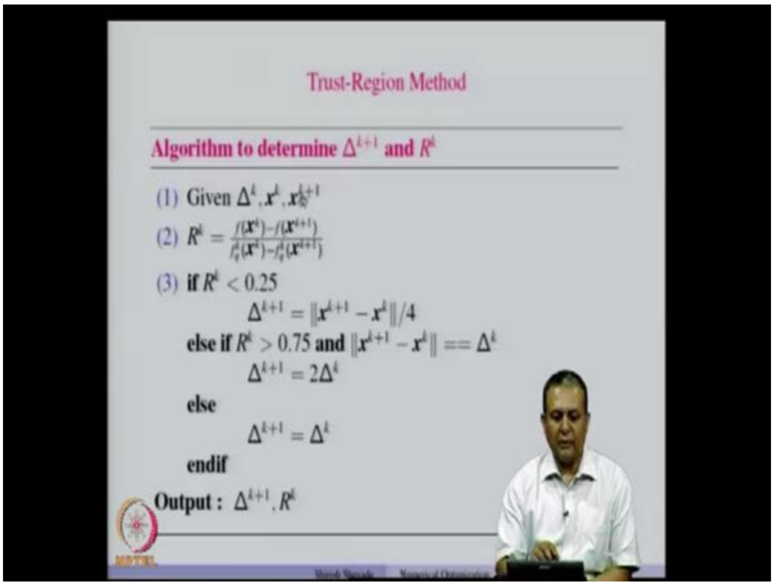
So, omega k is the region of trust that we have define based on the value of delta k. Now, instead of directly minimizing the quadratic approximation of the function which is f q x, what we do is that we minimize the quadratic approximation subject to the constrained that x belongs to omega k.

Now, remember that this f q x is a quadratic approximation. So, the subscript q stands for the quadratic approximation and k star stands for the k-th iteration so that means that the quadratic approximation is defined at the point x k and or it is defined around point x k.

So, this is a slight change in notation here that, I have now used this quadratic. This constraint approximation problem is easy to solve. But, now the question is that once we find x plus x k plus 1 which is solution of this problem, then how do we determined the new region of trust which is omega k plus 1 or how do you determine delta k plus 1 because once you determine delta k plus 1 that can easily determined what is omega k plus 1.

So, in other words what we are interested in is finding out that how this delta k plus 1 is obtained? If we have knowledge of x k plus 1 which is a constrained minimizer of this optimization problem now, one can use the ratio of the actual reduction in f to the predicted reduction in f to find out delta k plus 1. And we will see how to do that.

(Refer Slide Time: 27:50)



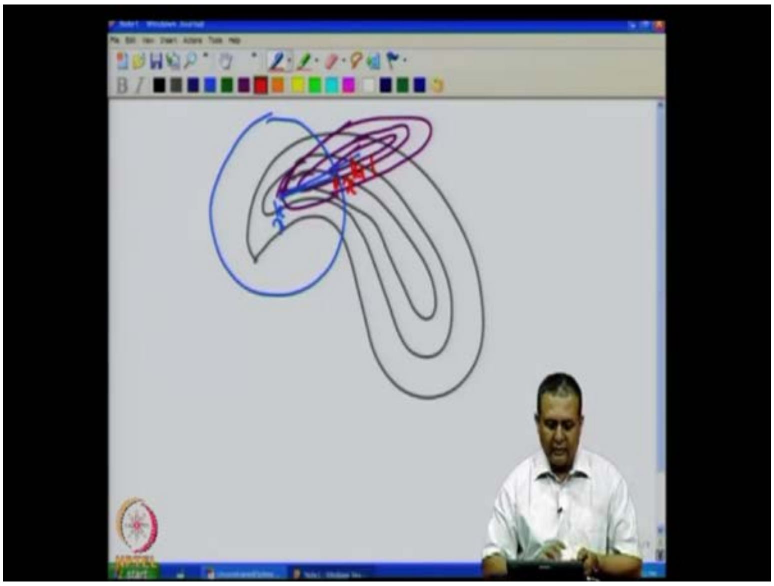
Now, here is simple algorithm to determine delta k plus 1 and this R k we will define it now, so given that delta k x k and x k plus 1. So, we initially start with x k and then solve this optimization problem to get x k plus 1 and that optimization problem used delta k. So, given the knowledge of x k delta k and x k plus 1. How do we determine delta k plus 1 that is the question that we would like to ask.

So, let us define the ratio R k to be the function value at x k minus function value at x k plus 1 divided by f q k x k minus x q k plus 1. So, f q k is the quadratic approximation of the function at a given x k and f q k is the function that quadratic approximation value at the point x k plus 1. So, this the numerator denotes the actual decrease in the objective function and denominator denotes the predicted decrease in the objective function. Now, if the quadratic approximation is good, then this ratio is close to 1.

And if, the quadratic approximation at a current point x k is bad, this ratio will be close to 0 and there could be situations where there could be a increase in the function value at x k plus 1, so that means that this ratio could be negative. So, there are different possibilities for R k. Now, if R k is less than 0.25 such these numbers which are mentioned here. The algorithm is not so much sensitive to this numbers so, one can change appropriately.

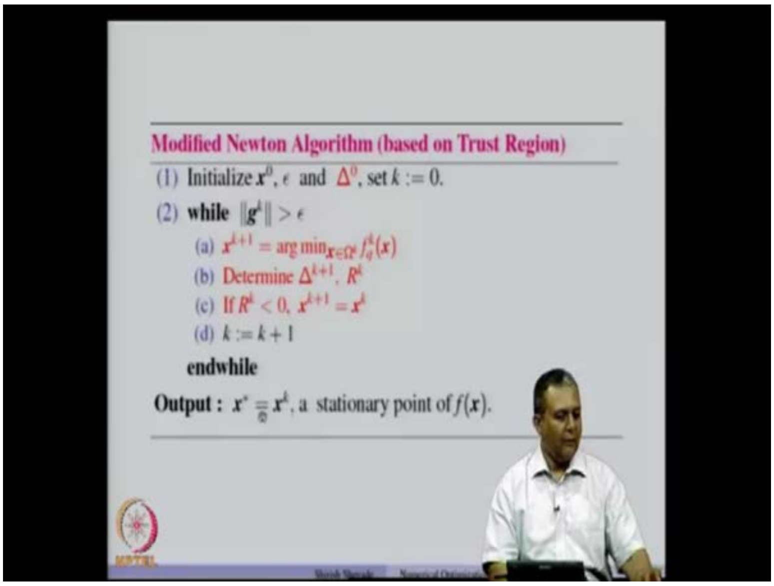
So, if the ratio is small which means that the actual decrease, by the predicted decreases is very close to or less than 0.25, then which means that the quadratic approximation is not good enough. Now, if the quadratic approximation is not good enough that means that we have to shrink the region of trust and that is done by typically setting delta k plus 1 to be the distance between the x k plus 1 minus x k by 4. So, all the numbers mentioned here are typical numbers and if one wants, one can change it based on the application. So, the delta k plus 1 reduces and hence the region of trust also reduces for the next iteration. That is, because the current approximation is not good enough. On the other hand, if the current approximation is good and the step taken is equal to delta k. So, if you look at this figure.

(Refer Slide Time: 31:24)



So, if starting from x k if we go to a point x k plus 1 and x k plus 1 lies on the boundary so, that means that the distance between x k plus 1 and x k is nothing but delta k. So, in such a case what one can do is increase the region of trust to 2 delta k. So, whatever was the previous region of trust or previous delta k that is a multiplied by 2 and if none of these two things happen then you return delta k plus 1 to be delta k and the algorithm returns delta k plus 1 and r k r k is this ratio and delta k plus 1 is the new delta value for the new region of trust.

(Refer Slide Time: 32:25)

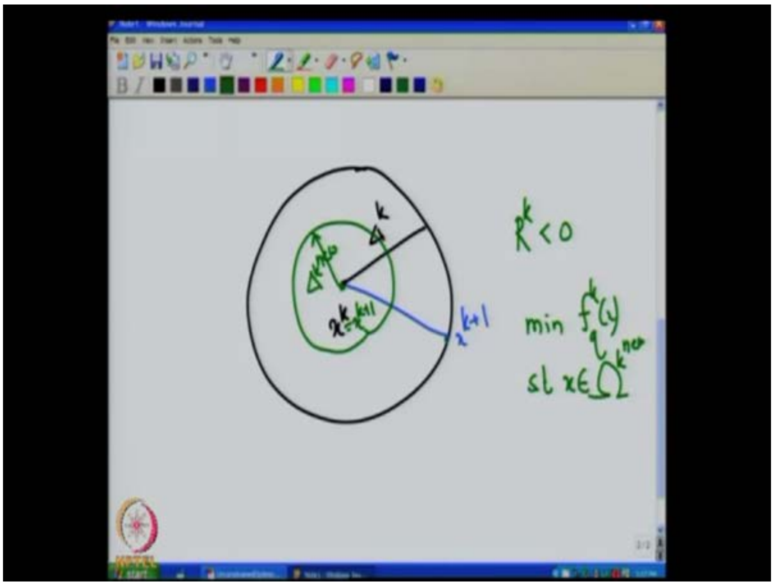


Now, how does the modified Newton algorithm which is based on trust region method look like. So, as usual we initialize x 0 and epsilon and also initialize delta 0. So, this is a parameter of that, we will need and the iteration count is set to 0. Now, while the norm of the gradient is greater than epsilon you find x k plus 1 to be arg min x belongs to omega k f q k x.

So, we minimize this quadratic approximation of f at x k subject to the constraint that x belongs to the set omega k. Now, we have x k x k plus 1 and delta k. Now, we are in a position to use our previous algorithm to determine delta k plus 1 so that delta k plus 1 and R k are determine using the previous algorithm. Now, if we realize that R k is less than 0 that means actually there is a increase in the function value from x k to x k plus 1 then what we do is that x k plus 1 is set equal to x k. So, that means that the new point is same as the current point. So, we do not move to the new point. So, the change which was made here that will be undone by this statement x k plus 1 is equal to x k. And remember that if R k is less than point twenty five, we had shrunk the region of trust.

So, that anyway will hold here and therefore, will be sees the initial region of trust was not good. We again use thus, reduce the region of trust if R k is less than 0 and see how to move from x k to the new point and then the rest of the procedure is at k equal to k plus 1 the iteration counter is increased and the process is repeated till norm of g k is less than or equal to epsilon and as output what we get is stationery point.

(Refer Slide Time: 34:57)

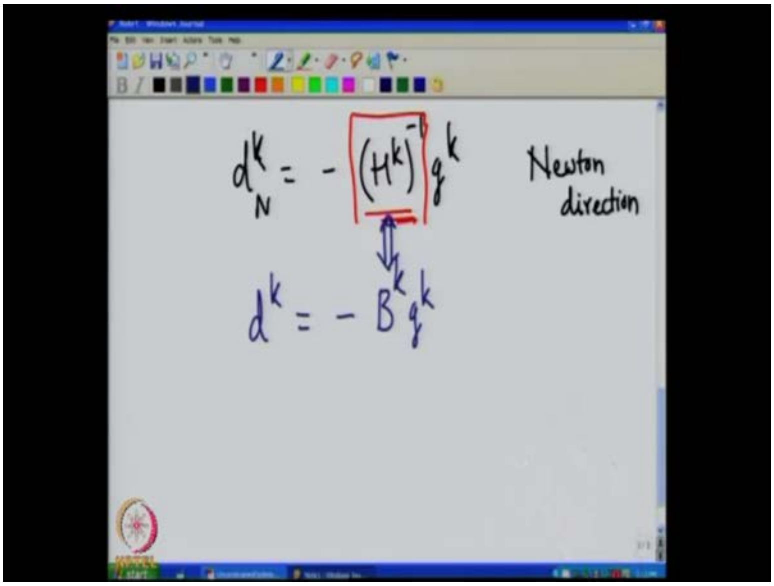


So, in other words if this is our x k, and this is the region of trust that we have and the corresponding radius is delta k. Now, at this suppose the algorithm takes a step to be new point x k plus 1 and if at this point, we realize that the function value has increased so, that means that this region of trust was really not good. So, we reduce the region of trust to a new boundary. So, this is the case where R k is less than 0.

So, the new region of trust is defined and then we against so, x k plus 1 is same as x k plus 1 x k. So, instead of taking this step where the function value actually increased what we do is we come back to x k reduce the region of trust and then this is our new point x k plus 1 and then we try to solve the problem minimize f k q x subject to x belongs to now. So, let me call this as delta k new and so the corresponding omega k new. So, that way we make sure that the function value does decrease at the end of every iteration. So, this method is called trust region method.

So, both these methods, the earlier method that we saw and this method they still use the Newton method ideas. But, the way the steps are taken is not like classical Newton method. So, in the previous method that we saw it was make sure that the hessian matrix are the direction, that the direction d k that we get is a descent direction which is unlike Newton method. Where the direction is always is in the classical Newton method the direction always minus H k inverse g k and which need not be a decent direction. The second approach that we saw here, the trust region based approach that makes sure that the maximum step is taken the approximation is good otherwise or in the region of trust is large, otherwise the region of trust is shrunk so that, we do not reverse it. Now, let us look at some modifications of Newton methods.

(Refer Slide Time: 38:55)

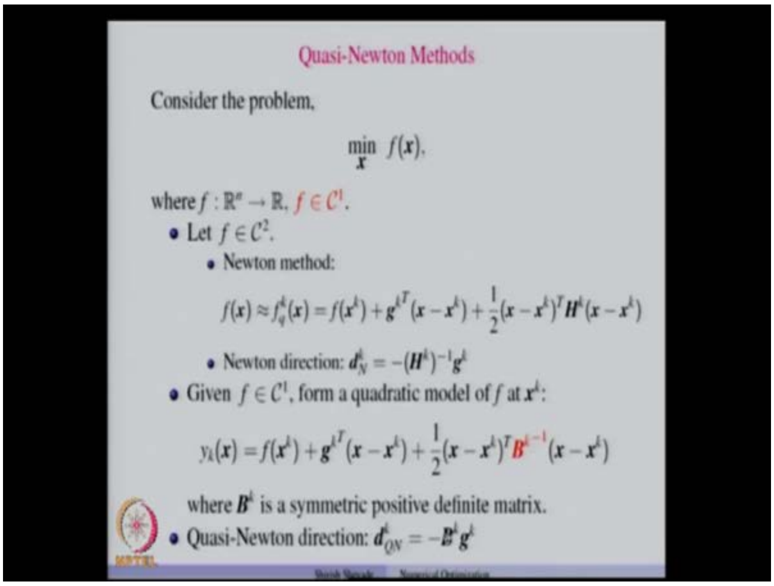


Now, remember that in the Newton method we have the direction d k to be minus h k inverse g k and this requires. So, this is Newton direction now, this requires the hessian matrix and its inverse now that is many a times computationally expensive operation, especially when the size of H k is very large. So, a set of methods were suggested and what they do is that they approximate this hessian inverse by a matrix and then choose a direction d k to be minus B k g k.

Now, if B k is a positive definite matrix then we know that this is a descent direction. The direction d k is the descent direction. Now, how good is this approximation that is the question that we would like to answer. So, in some cases it turns out that especially for quadratic functions. If this B k matrix is obtained in such a careful way, then this B k after certain number of iterations approximates the inverse of the hessian. That is, the typical of a quadratic function but may not least hold. But, nevertheless this approximation of Newton method or modification of Newton method is very popular because if you look at this operation. This does not require inversion of a matrix.

So, computational effort required by this method to find the direction is much less or is much less compared to the computational effort required by the Newton direction. Typically, if we want to invert n by n matrix the complexity of the operations order n q and that is avoided in this case, because to get the new direction d k. We just need matrix vector multiplication and that needs order n square effort so the computational effort is saved in this method. So, such methods are called quasi- Newton methods and let us now study some of those methods.

(Refer Slide Time: 42:02)

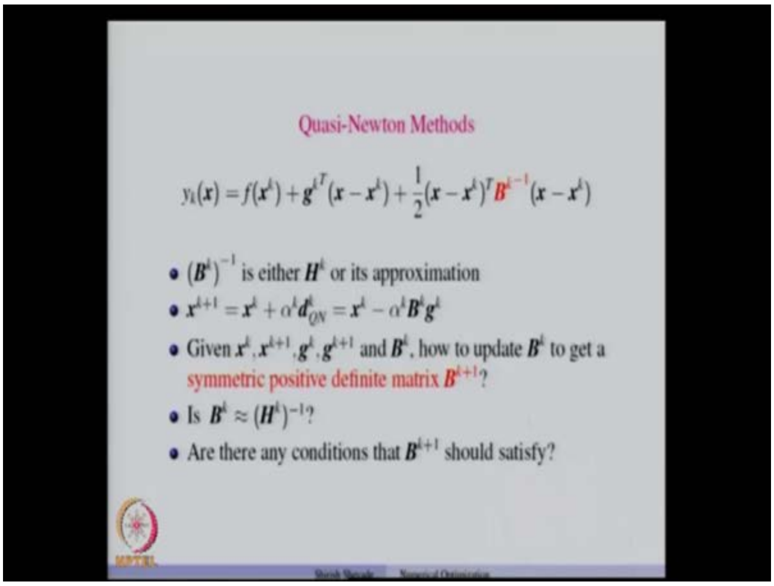


Now, consider the problem to minimize f of x, where f is function from r n to r and more importantly f is a continuously differentiable function. Now, if we use Newton method we have to assume that f belongs to c 2. And then as we saw earlier, that Newton method depends on quadratic approximation of the function at a given point x k. So, this is the quadratic approximation which requires the first derivative and the second derivative information and the Newton direction chosen is minus h k inverse g k.

But, now suppose we want to work in this setup where you want to minimize f x, where f is a continuously differentiable function that means we do not want to use the second order information for this purpose now, given f which is a continuously differentiable. Suppose we form quadratic modular f at x and let us denoted by y. So, y k x. So, this k stands for the quadratic approximation of f at x k. So, using Taylor series, if you write f of x k plus g k transpose x minus x k plus half x minus x k transpose now, instead of the hessian matrix because the functions are only continuously differentiable. So, the second order information is not available. So, we write some matrix b k inverse into x minus x k, where b k is a symmetric and positive definite matrix.

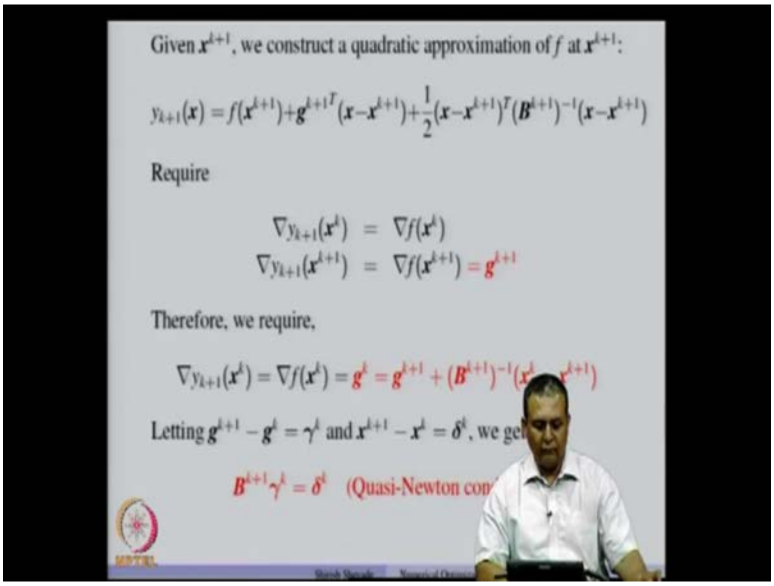
So, make sure that this matrix b k whatever we have is a symmetric matrix. This is like hessian matrix which also symmetric and also positive definite. Now, the hessian matrix is not always positive definite that is what you saw earlier, when we studied Newton method and in fact, that is one of the drawbacks of Newton method that every time the hessian matrix is not positive definite. So, if we make sure this matrix B k is positive definite then one can chose a direction d k to be minus B k g k and that direction is called a quasi-Newton direction. Now, let us see how to use this quasi-Newton direction in our approach.

(Refer Slide Time: 44:38)



So, the quasi-Newton methods use the quadratic model of f around x k using the matrix B k inverse rather than the matrix h k. Now, as I said earlier that this B k inverse is either h k ideally, you would like it to be h k or its approximation. So, the question is that if we use the direction d k to be minus B k g k which is quasi-Newton direction and get x k plus 1 to be x k plus alpha k d k which is nothing but x k minus alpha k B k g k. So, how good is the approximation of B k inverse with respect to h k? And the second question is that once we move to the point x k plus 1, how do we get B k plus 1? How to update B k using the information x k x k plus 1 g k g k plus 1 and B k to get a symmetric positive definite matrix B k plus 1? So, this is an important question that we would like to answer and whenever, we update B k to be B k plus 1. How good is that approximation to corresponding h k plus 1 inverse and when we do this update or there any conditions that B k plus 1 should satisfy.

(Refer Slide Time: 46:32)



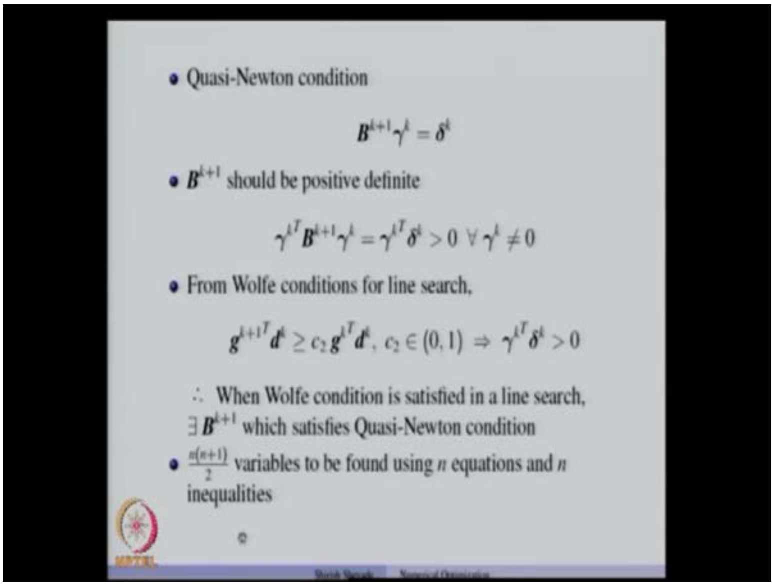
So, these are some of the questions that we would like to answer. So, let us take a given point x k plus 1 and construct the quadratic approximation of f at x k plus 1 and that quadratic approximation will be denoted by y k plus 1. So, this k plus 1 stands for file that quadratic approximation is obtained around x k plus 1. So, this function would look like f of x k plus 1 plus g k plus 1, transpose x minus x k plus 1 plus 1/2 x minus x k plus transpose B k plus 1 inverse x minus x k plus 1.

Now, how good is this quadratic approximation? Now, for this quadratic approximation to be good, what we want is that the gradient of this information at x k plus 1 should be the gradient of f at x k plus 1. And gradient of this function at x k should be equal to the gradient of the function at x k. So, at the 2 consecutive points x k and x k plus 1 the gradients of the actual function f and the gradient of the function y, they should match with each other. So, in other words what we want is that gradient of y k plus 1, gradient of y k plus 1 is evaluated at x, k should be equal to gradient of x k and gradient y k plus 1 evaluated at x k plus 1 should be equal to gradient of f of x k plus 1.

Now, you will see that the gradient of y k plus 1 is nothing but g k plus 1 which is the gradient of y k plus 1 evaluated at x k plus 1 is g k plus which is same as that of f at x k plus 1. So, this equation is all satisfied. Now what about the first equation? We want that equation also to hold or in other words, gradient of function f at x k should be same as the gradient of y k plus 1 at x k and so, we know that gradient of f at x k is nothing but g k and the this g k should be equal to the gradient of y k plus 1. Evaluate at x k. Now, if we evaluate gradient of this function at x k, what we get is g k plus 1 transpose into plus B k plus inverse into x k minus x k plus 1.

So, we want B k plus 1 or we want this quadratic approximation to be such that, this condition is satisfied or in other words, B k plus1 should be such that B k plus 1 equal to B k plus one gamma k is equal to gamma k, the way this is obtain is using this expression. So since, B k plus1 is, we want is a invertible matrix. So, we can write g k plus1 minus g k to be gamma k and x k plus 1 minus x k to be delta k and write this quantity as B k plus 1 gamma k equal to delta k. So, this we want B k plus 1 to satisfy this condition and this condition is called quasi-Newton condition.

(Refer Slide Time: 50:21)

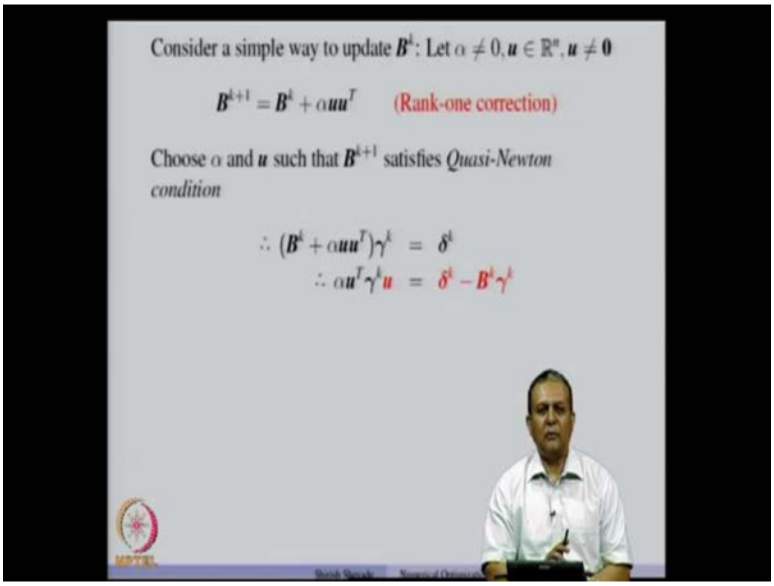


Now, let us look at this quasi-Newton condition B k plus 1 gamma k equal to delta k. Now, certainly B k plus 1, we want it to be positive definite. Now, gamma k transpose b k plus 1 gamma k is nothing. But, gamma k transpose delta k and that is greater than 0 for all gamma k not equal to 0. If we use Wolfe conditions for line search or in other words, suppose if we take Wolfe conditions for line search. So, what we get is g k plus 1 transpose d k is greater than or equal to c 2 g k transpose d k lets c 2 is a positive fraction, in such c 2 should be in the intervals c 1 to1 but here i have intentionally mentioned to be 0 to 1. Just to indicate that it is a positive fraction and so, if you take the right hand side to the left side so that we get is g k plus 1 minus c 2 g k into whole transpose d k is greater than or equal to 0 or in other words, so what we get from this is that gamma k transpose delta k which is greater than 0.

So, if we make sure that Wolfe conditions for line search are satisfied. Then we ensure that gamma k transpose delta k is greater than 0 and gamma k transpose delta k greater than 0 essentially means that the B k plus 1 matrix should be positive-definite matrix. Therefore, when Wolfe condition is satisfied in a line search then exists of B k plus 1 satisfies quasi-Newton condition.

Now, how to get this B k B k plus 1? Now, B k plus 1 is a symmetric matrix. So, a symmetric matrix has of size n as n into m plus 1 by 2 are independent variables. So, the number of variables here is n into m plus 1 by 2 number of equality constraints here or number of equalities here are n and in addition to symmetricity, we also need positive definiteness of B k plus 1 that is, all the principle minus of B k plus 1 should be positive so that will result in n inequalities. So, in all we have n into n plus 1 by 2 variables n equalities and because of the positive definiteness, we want n inequalities to be satisfied. So, in other words this is a problem in m n plus 1 by 2 variable which are to be found using n equations and n inequalities. So, you will see that the number of variables is much larger than the number of equations are inequalities. So, there exist many solutions which satisfy this quasi-Newton condition. Now, let us look at some simple ways of finding B k plus 1.

(Refer Slide Time: 53:56)



So, let us consider simple way to update B k and let us assume that alpha is scalar which is non-0 and u is n-dimensional vector which is again non-0 and let us update B k to BH k plus 1 as B k B k plus 1 is equals to B k plus alpha u transpose. Now, alpha is a scalar u transpose is a symmetric matrix of rank 1. So, that is why this is called the rank 1 correction. So, if B k is a symmetric matrix, then certainly B k plus alpha u, u transpose is also a symmetric matrix and therefore, B k plus 1 will be a symmetric matrix.

Now, if B k is positive definite, is there a guarantee that B k plus 1 is also positive definite that is the question, that we would like to answer. Now, before we do that we want to find out what are possible values of alpha and u that could be used. So, that quasi-Newton condition is satisfied by the matrix B k plus 1 or in other words, what is that alpha and u such that B k plus 1, gamma k is equal to delta k. So, we will chose alpha and u such that B k plus 1 satisfies the quasi-Newton condition or in other words, B k plus 1 gamma k equal to delta k implies B k plus alpha u, u transpose into gamma k equal to delta k.

Now, how do we make sure that or how do you get this alpha and u from this equation. So, as you will see that there are lots of possibilities to do this and we look at a simplest possibilities, so if you expand this and then equate u to be delta k minus b k gamma k then we can chose alpha equal to 1 over u transpose gamma k or in other words, alpha u transpose gamma k to be 1 and then that will satisfy this equality and B k in B k plus 1. In this case will satisfied quasi-Newton condition, now will that B k also B k plus 1 also be positive definite and how good is this rank 1 correction. We will discuss about those things in the next class.

Thank you.