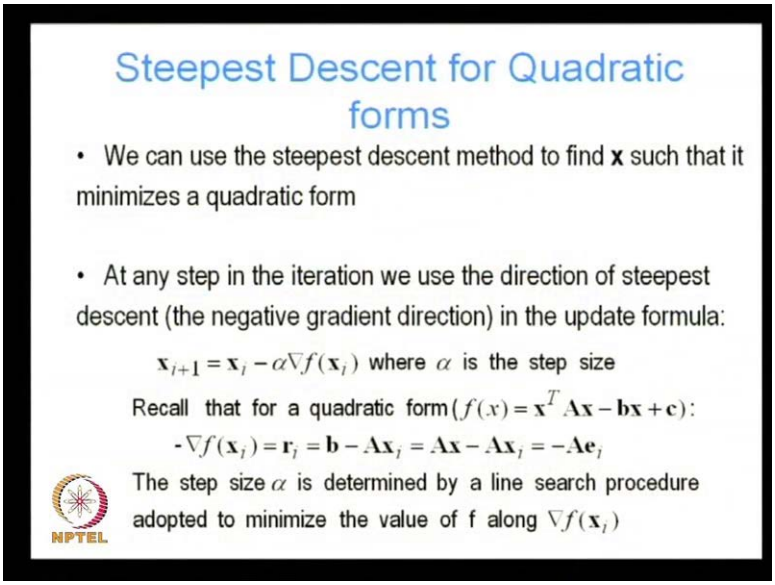
Numerical Method in Civil Engineering
Prof. Arghya Deb
Department of Civil Engineering
Indian Institute of Technology, Kharagpur

Lecture - 17
Gradient Based Method

In lecture 17 of our series on numerical method in civil engineering, we will continue with our discussion on gradient based methods. Last time, we briefly introduce the simplest possible the simplest gradient based method the method of steepest descent where, basically we say that we want to minimize a function f of x.

(Refer Slide Time: 00:30)



We say that at any point in our solution space, if we look at the negative of the gradient direction, that gives me the direction of steepest descend. That gives direction in which if I move in my function space in my variable space. I am going to get the largest reduction in the function value. We said that we can use the steepest descent method to find the minimum of a function, that is to find x such that it minimize a quadratic form. So, at any step in the iteration we use the direction of steepest descent to find the next value next iterate.

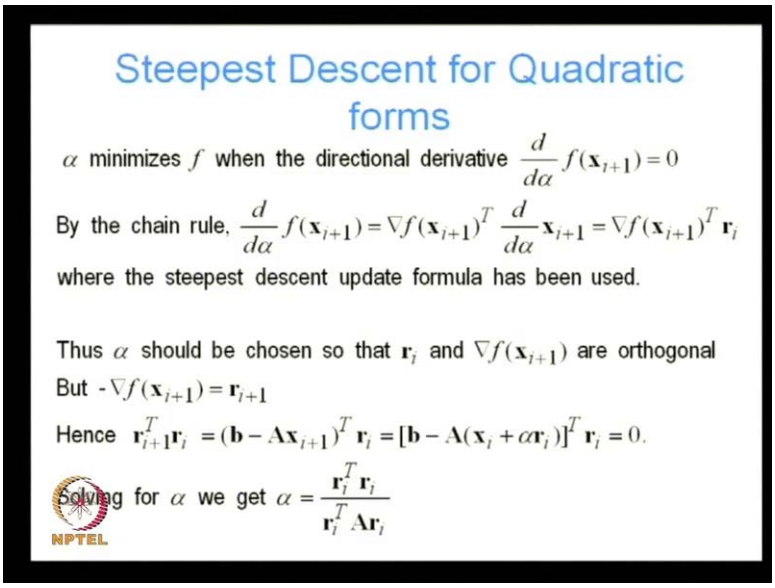
So, if know x the value of the I know x i that is the current value of the iterate to find x i plus 1. I take a step of size alpha in the direction of steepest decent, which is given by the

negative of the gradient of f. So, x i plus 1 is equal to x i minus alpha gradient of f of x i were alpha is the step size. Recall how we defined a quadratic form, we give the expression for the quadratic form. We said it is nothing but a quadratic in multiple dimension. So, we said f of x is equal to x transpose A x minus b x plus c. That is a general expression for the quadratic form. If we evaluate the gradient of this quadratic we calculate grad of f x and take the negative sign. We see that that is equal to b minus A x. So, take the gradient of this right with respect to x.

So, we are going to get b minus A x. I mean negative of the gradient that would be b minus x. So, at when x is equal to x i we will get b minus A x i that will be my negative of the gradient. That is actually nothing but the residual r i is equal to b minus x I, because b is actually a x right b is equal to A x. So, A x minus x i if we define the error to be x i minus x that is equal to minus A e i. So, the residual is equal to the negative of minus A e I, how do we determine the step size? Well, the step size alpha is determined by line search procedure adopted to minimize the value of f along grad of f x. So, at any point in I in my solution space.

I know that the direction I which I need travel is given by minus grad of f x. So, that is the direction, which if I move I will get the steepest reduction in the function, but how far should I move that is the critical problem. We looked at it last time, if I take two largest step I can just cross the valley and go to the next rig. So, I need to calculate the appropriate step size.

(Refer Slide Time: 03:57)



So, alpha minimize f then the direction derivative of alpha at that location is equal to 0. So, what is directional derivative it is nothing but d d alpha f of x i plus 1. So, direction I have a direction, I have the function. I am looking at how the function is varying along that direction. The directional derivative tells me how that function varies with distance along that direction right. So, I keep the direction the same and I only vary the step size.

So, how f varies with the step size? So, how f varies with alpha? So, d d alpha of f at x i plus 1 is equal to 0. When alpha minimize, when it reaches a minimum along that direction. Since, it is a function of alpha along that direction I have to minimize with respect I have to take the derivative with respect to alpha and set that equal to 0. Plus, along that direction, it is only a function of alpha. So, along a direction in that solution space I have fixed the direction. I can only vary the magnitude of the step size and I want to minimize that with respect to alpha, in order to reach a minimum. Then we use the chain rule we evaluate d d alpha f x plus i plus 1, we calculate gradient of f.

So, we take first take the derivative with respect to x. Then we take the derivative of x with respect to alpha nothing but the chain rule. So, this is the gradient of f evaluate x i plus 1 transpose since I have to dot product. So, directional derivative scalar, so its going to be a scalar as well.

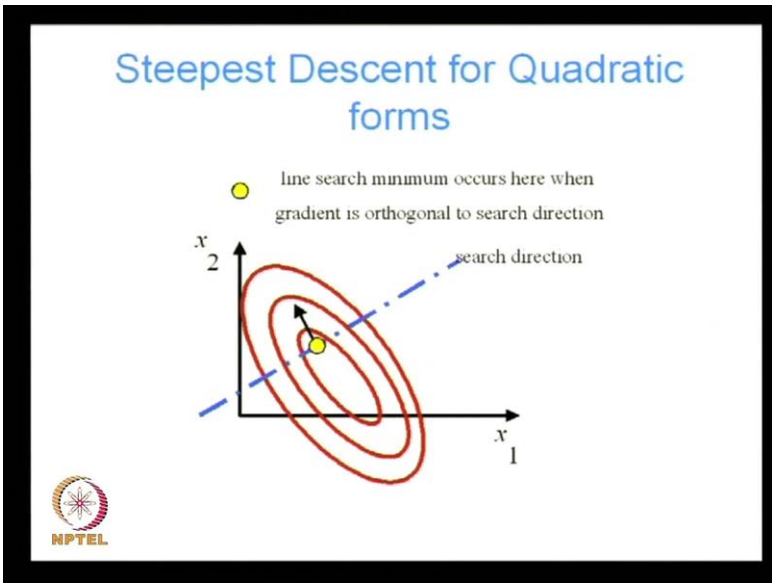
So, I have this thing dotted with that and x i plus 1, if I take the derivative with respect to alpha. Well, you can see that from here, if I take the derivative of x i plus 1 with respect

to alpha, what do I get? I get minus gradient of f x I, which is equal to r i. So, I have r i gradient of f x i plus 1 transpose d d alpha of, this is nothing but the gradient of f x i plus 1 transpose times r i. Where, the steepest descent formula update has been used, thus alpha should be chosen. So, that r i and grad of f x i plus 1 are orthogonal. So, what does it mean? That means, I started with x i right and I am trying to find out x i plus 1. I am moving along the certain direction, which is the gradient, which is the negative of gradient direction at x i.

So, I am saying that, I can if I move along that direction in such a way that my starting residual become orthogonal to the gradient along the line at some point along the line. That is the point, where I have reached my minimum. So, as I travel along the line, where grad of f x i plus 1 and r i become orthogonal. That is the my minimum solution, but what is grad of x i plus 1? It is by definition minus r i plus 1. So, what does that mean that mean at the minimum r i plus 1 transpose r i must b equal to 0, since this is orthogonal to r i. So, r i plus 1 must be orthogonal to r i and r i plus 1 by definition is b minus a x i plus 1, that is my definition.

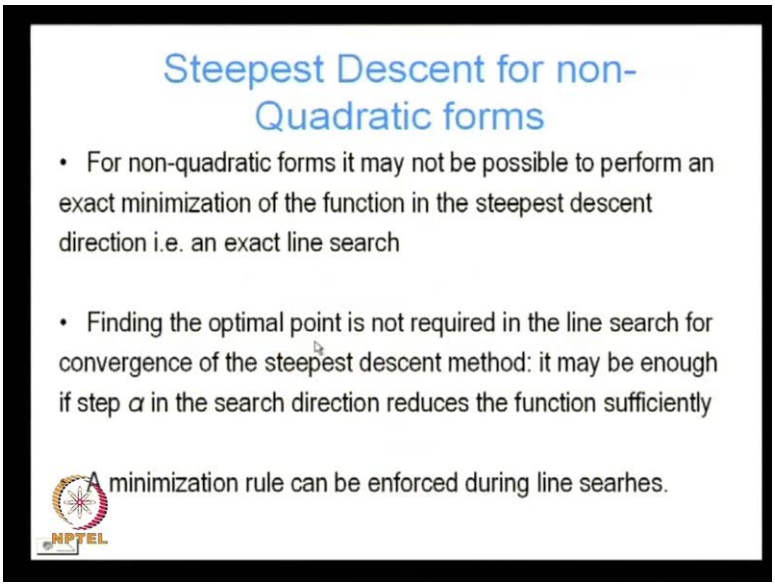
This transpose r i must be equal to 0, this again I replace x i plus 1 in terms of alpha. I write x i plus 1 is equal to x i plus alpha r i in this expression, this gives me an expression for alpha I can evaluate alpha. I get this to be my step size alpha is equal to r i transpose r i divided by r i transpose A r i. So, the step size is determine entirely by the value of the residual at my starting location at my i form my, where I start taking the step. So, the alpha is totally known.

(Refer Slide Time: 08:34)



So, this is the little picture, which shows this, suppose I am traveling in this direction. This is at certain location at my x i at my value of my current iterate the gradient direction is given by this. So, I will continue traveling in that direction. I will continue travelling in that direction until the gradient is orthogonal to the such direction. Here, is my gradient at yellow point a my gradient my negative gradient is pointing like this, when this become orthogonal to this search direction. That is where I am going to get my minimum, that is where I am going to get my minimum because, is that clear? That is where I have to stop. That is where I have my minimum along the line.

(Refer Slide Time: 09:45)



So, this is all very well and good for quadratic form were you can get close from expression for the step size. You can get close from the expression for the size and using that steps size. You can use the steepest descend method in optical manner to find the minimum for the quadratic form. However, what happen if we have a general non-linear function. For non quadratic form, it may not be possible to find that optimal value of alpha. It may not be possible to perform an exact minimization of a function in the steepest decent direction, that is perform an exact line search.

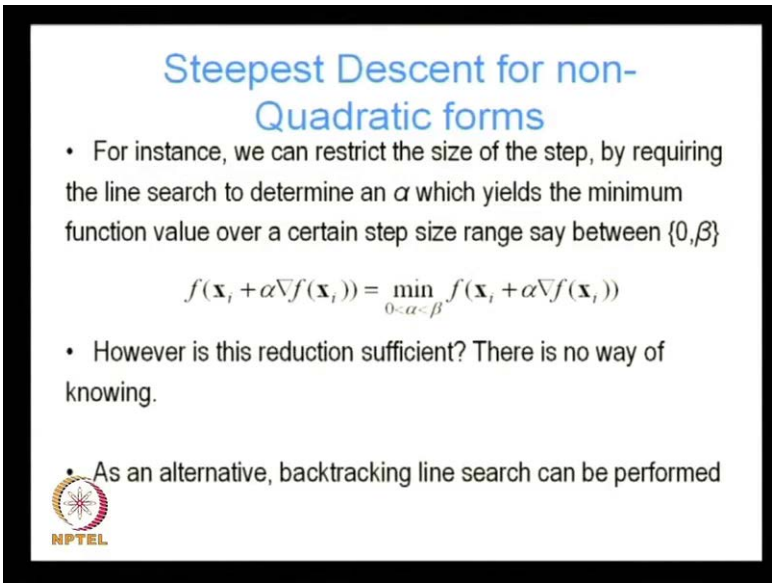
Why is that? Well, because the function is just too complicated. It will not give a close form expression, will give me a root for alpha , but how ever finding the optimal point is not required in the line search, for the convergence of the steepest descend method. It is not really essential that along a particular gradient along a particular directio. Along a search direction, I actually find the minimum. It is very good, if I find the minimum. Because, I have actually travel, I have taken a step along the direction, which is optimal I have done all. I could in that direction to minimize the function, but suppose I am not able to find the close form expression for alpha.

I am not able to find the value this of the of the optimal step size. It does that mean my steepest decent method is not going to work well, known it still going to work, but provided I make sure that I have sufficiently reduce that function in that gradient direction, if I take an arbitrary step.

So, the function actually increases, that the function actually increases along that direction, if I take an arbitrary step along that direction. So, that the function actually increases or it reduces by negligible amount. Then very quickly my steepest decent method is going to collapse. It is not never going to reach a minimum, it is going to diverge. However, provided I take I reduce the function sufficiently in the direction of steepest descent.

I am still guaranteed that my steepest decent method is going to converge, even though I do not find the minimum along that direction, if I reduce the function sufficient. Then it will be convergent, how can we do that? Well, we can we can enforced the minimization rule, during the searches what do I do? For instance we can restrict the size of the step.

(Refer Slide Time: 12:20)

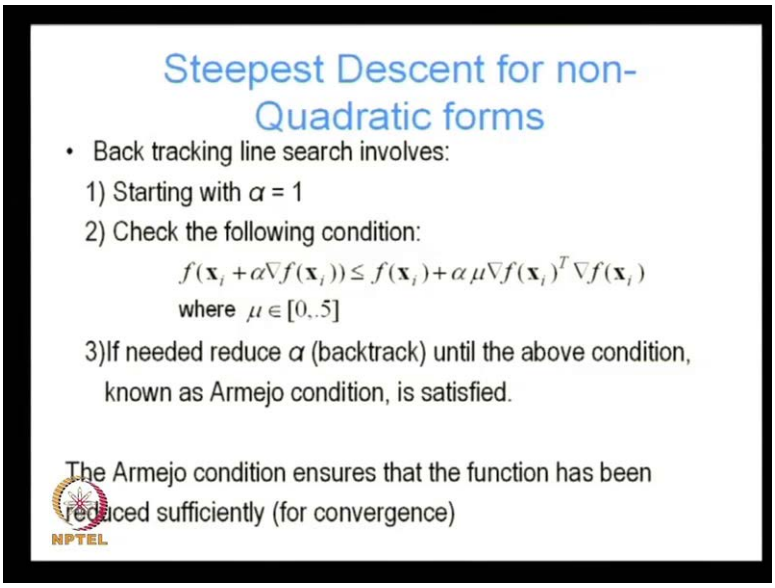


So, is not possible, since we do not have a close form solution. We do not know what is the size of the step to take. So, we say that, well I am going to limit my size of the step. Suppose, I put my size limit to beta I say that my step has to be less than some arbitrary value beta, I said that. Then I say that between 0 and beta, I am going to find the exact minimum. I am going to keep on taking a small step, and I am going to go, until I find the minimum in that interval. Then once I find minimum that internal I move on. Does not matter, if I find the total global minimum along that line does not matter, so I move on.

So, we can restrict the size of the step by requiring the line such determine alpha, which yields the minimum function value over a certain step range say between 0 and beta. So, we say that we going to choose alpha, such that this is the minimum over this range between 0 and beta. So, it is going to minimize this function within the range 0 and beta.

However, is the reduction sufficient, we said that I have orbital set a interval. I said that I am going to minimize the function within that interval, but is the reduction in the function sufficient, we do not know. In that reduction is not sufficient my algorithm may break down. So, as a alternative there is something, which is known as back tacking line search and that has got a mode sound theoretical foundation and that is generally performed. So, what does that mean?

(Refer Slide Time: 14:10)



Well, back tracking line search involves starting with alpha equal to 1. We start with the value of alpha equal to 1. Then we check the following condition. We check whether the value of the function I evaluated with that value of alpha is less than the value of the original function at x i plus some quantity. Plus, some quantity were that some quantity is nothing but the norm of the gradient at x i right norm of the gradient at x i scaled by alpha and another scalar mu, when this condition is satisfied when at the location alpha. At the location of alpha the function is less than this upper bound.

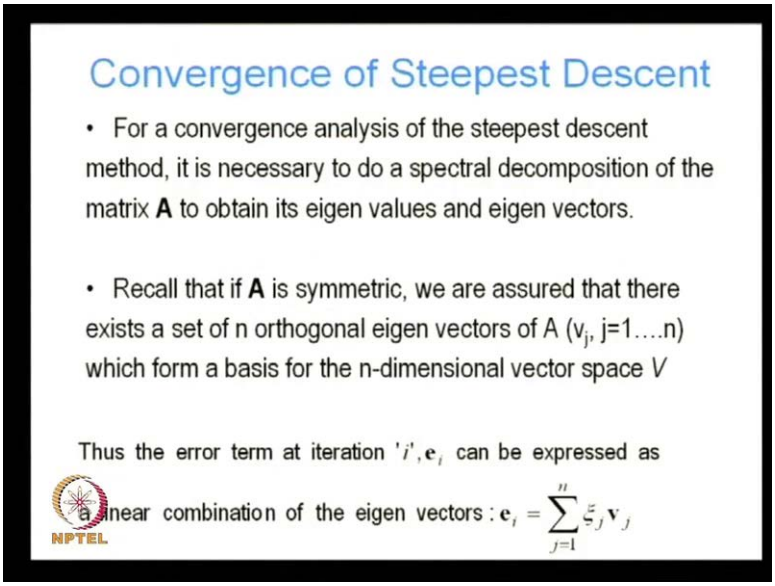
Then we can be sure that the function as been reduced sufficiently. This condition is known as the Armejo condition. If it satisfies the Armejo condition we can be assured that people have, people actually derive those equation. I am not going to do that here, the people have obtain this result, that if we reduce take alpha, such that the function is sufficiently reduced. What is the meaning of the sufficient reduction? It means that f of x i plus alpha at the new location the function must be lesser than or equal to the original value, plus some scalar multiple of the norm of the gradient at the original value. So, if the function has been reduce by this much. Then it has been sufficiently reduced. That means, that it is going to satisfy it is going to converge, it is not going to diverge.

If necessary we need to back track, if we started with the value alpha equal to 1. If we find that the Armejo condition is not being satisfied. We have to reduce alpha, until we satisfy the Armejo condition. If we do that we ensure that the function has been reduce

sufficiently, what the sufficient mean? Well, it is has been reduced sufficiently. So, that eventually it is going to converge, is that clear? So, the steepest decent method, well it works beautifully, if the function is quadratic, if the function is non quadratic, it is not so simple, because why it is not so simple?

Well, the main reason why it is no so simple is because we do not have closed form expression for my step size. Because, I do not have the close form expression for the step size we have to determine the step size iteratively. It is never an optimum, it is very hard to find a optimum step size, but eventually we if we stratify the Armejo condition. We are going to get at the minimum, but it might be very expensive. It may be computationally expensive.

(Refer Slide Time: 17:31)



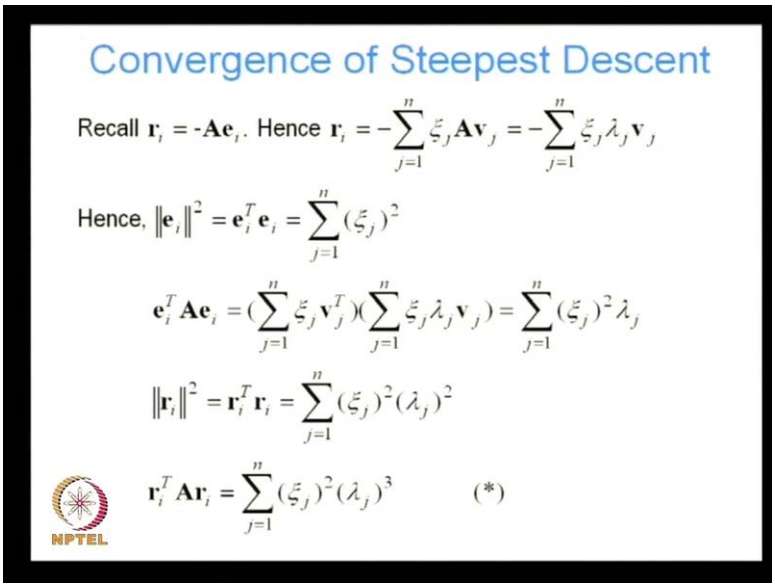
For a convergence analyses of the steepest descent method, like we look at all the previous method that will look all the previous iterative method. We saw that if we have to show convergence. It is going to depend on the some sort of we have to do a Eigen value analyses of the coefficient tenser. Since, for gauss signal for jalopy we did Eigen value. We did spectrally composition, we found the largest Eigen value the spectral radius. We saw over there, if that radius is bounded, then that iterative method converges.

So, similarly for the steepest descent method in order to obtain convergence in order to convergence analyses, it will be necessary to do a spectral decomposition of my matrix a.

So, now, again we are moving back to quadratic form. As soon as I start talking about the matrix a I am going to, I am moving back to quadratic form. Because, all this results for convergences things. Like that they have been obtained for quadratic forms. Because, it is possible, it is amine able. You can get a analytical expression for arbitrary non-linear functions not possible. So, recall that if a is symmetric we are assure that there is the exits a set of n orthogonal Eigen vector of a.

We know that the if a matrix is a symmetric its characteristic spaces, it Eigen vectors are orthogonal to each other and these. Since, they will form a basis for my n dimensional vector space v. Thus, the term at the arrow term iteration, I can be expressed as a linear combination of the Eigen vectors of a. Since, the Eigen vector of a form a basis any vector any vector can be expressed as a linear combination of those Eigen vector. So, any vector error vector also the error vector also belong to that space. The error vector can be expressed as the linear combination of those Eigen vector v j, e i I can write as sigma j is equal to 1 psi j v j.

(Refer Slide Time: 19:55)



Let us recall, we saw that the residual r i is equal to minus A e i. We saw that right in the first slide today, we said that r i is equal to minus A e i. Hence, we can say that r is equal to minus sigma j equal to 1 to n psi j A v j. I am just multiplying the right hand side with a right my matrix vector multiplication and again. Since, v j is an Eigen value of a recall

v j these v j are Eigen value of sorry Eigen vectors of a, since v j is Eigen vector of a. So, A v j is equal to lambda j v j, Eigen value time the Eigen vector.

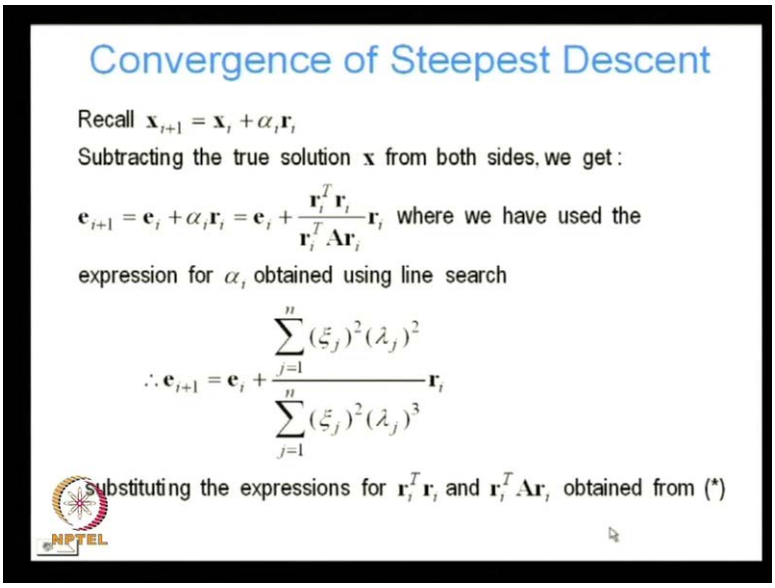
So, we have this, then if we take the norm norm square the l two norm square. I will just take the dot product of e with itself. I get sigma j equal to one psi j square y is that well. Because, my Eigen vector are orthogonal. In this case we actual assume that they are not only orthogonal, but also ortho normal. They have ortho normal, so I get this sigma j equal to 1 is equal to x i j square. So, I have e i transpose A e i e i is given by this quantity sigma j is equal to 1 to n x i j v j, e i is given by this quantity psi j lambda j v j A e i right e i is equal to psi j v j. So, A e i is equal to si j lambda j v j, then I impose the condition that the Eigen vector are orthonormal.

So, only those terms in the product are going to survey, which have this same v j. So, this and this, so that is going to give me x i j square lambda j. So, e i transpose A e i is going to be given by sigma j equal to 1 to n x i j square lambda j norm of the r i square norm of the residual square is nothing but r i transpose r i and r I, we saw is given by this. So, again using the same, idea the Eigen vectors are orthonormal. So, I get psi j square lambda j square. Only, the terms, which have the same index are going to survey. Because, other terms are going to give me 0. So, I get that and r i transpose A r i is we can see form here, r i is given by this. If I have A r I, I am going to get psi j lambda j another lambda j lambda a v j.

So, a v j is going to give me another lambda. That is going to give me x i square lambda j q is that clear? Why? Well, I have psi j lambda j v j a, I am going to take a dot product with I am going to take a dot product with A r i. What is going to be A r i? A r i is going to be psi j lambda j square v j, because a operating on v j is going to give me lambda j v j. So, I am going to have psi j lambda j square v j. If I take this dotted with that this psi j psi j, is this is going to give me psi j square? Now, I am going to get lambda j cube. Because, lambda j square times lambda is going to give me lambda j cube. So, that is going to give my r i transpose A r i.

So, I have written all this product, all this inner product all this error norms residual norm in terms of the Eigen values and the weights. What are the weight? The weights are the coefficient of the Eigen vector in the expression for the error, coefficients of Eigen vector in the expression for the error.

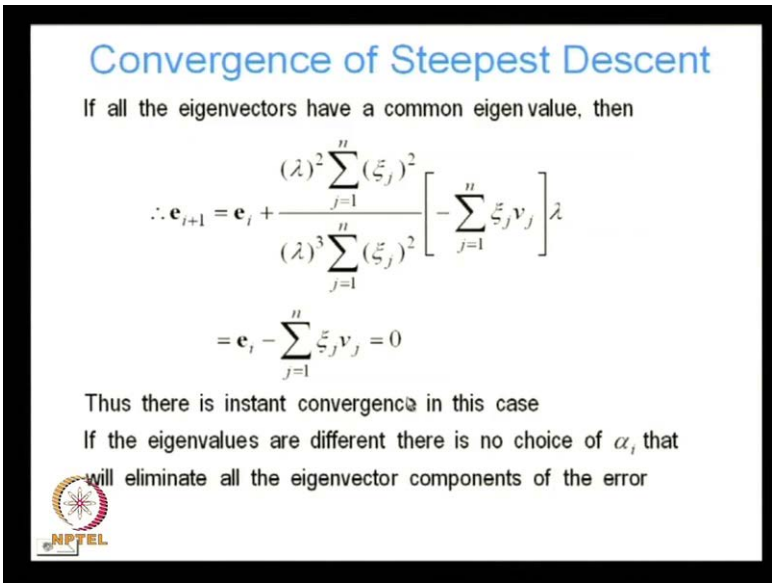
(Refer Slide Time: 24:39)



Now, let us recall that x i plus 1 is equal to x i plus alpha i r i. We have solved, we have seen that before x i plus 1 is equal to x i plus alpha i minus alpha i times gradient of f x. That is equal to r I, minus of that equal to r i. So, I have this if I subtract the true solution x from both sides I have x i plus 1 minus x, which is going to give me e i plus 1 is going to give me error at e i plus 1 x i minus x is going to give me e i. Then I am going to have alpha i r i.

So, I can write this as e i plus r i transpose r i by r i transpose A r i times r I, why is that? Well, that is because well obtain my value of alpha, like that from my minimization, where was that here. So, this is about I obtained alpha to be, so I am going to use that expression for alpha. There and I can write a i plus 1 is equal to a I, plus this times r i. Therefore, e i plus 1 is equal to e i plus, I am going to use this expression. Now, r i transpose r I, which is this to this sigma j equal to 1 time i j square lambda j square, which had put on top. Then I have r i transpose A r I, which I obtain to be sigma j equal to 1 to n psi j square lambda j cube. So, I just substitute that there, then I have the vector r i. So, we have substituted these expressions to get this.

(Refer Slide Time: 26:36)



Now, let us suppose if it is, so happen that a has multiplicity the Eigen value. It has got one Eigen value with multiplicity n. That means, that it has got, it has got just one Eigen value and n Eigen value with the same value. So, it has got my, so all the Eigen value are basically equal. So, I can, so lambda j square I can replace by that common value and pull it out to my sum. So, lambda j square, I can pull out lambda j cube I can pull out. In that case, what do I get? I get e i plus 1 is equal to e i plus lambda square sigma j equal to 1 to n x i j square by lambda cube sigma j equal to 1 x i j square times type. Then I have r I r I, I know is x i j v j right x i j v j, no r i is actually, r i is equal to x i j lambda j v j x i j lambda j v j. Again, I can pull out a lambda from here.

So, my lambda square lambda and this lambda cube cancels I have sigma j equal to 1, 20 n x i j square on top I have sigma j equal to 1 times x i j square. So, that cancels out right and I have e i minus j equal 1 to n psi j v j, which is identically equal to a i and my error becomes 0. So, what does that mean? That mean in that case I have instantaneous convergence. So, whatever might have been my error at step i at step i plus 1 my error is going to go 0. It is going to go 0, well there are there are very fine geometrical argument, which explain this. I did not want to go into too much detail here, but what happens here is that, when all this Eigen values become equal.

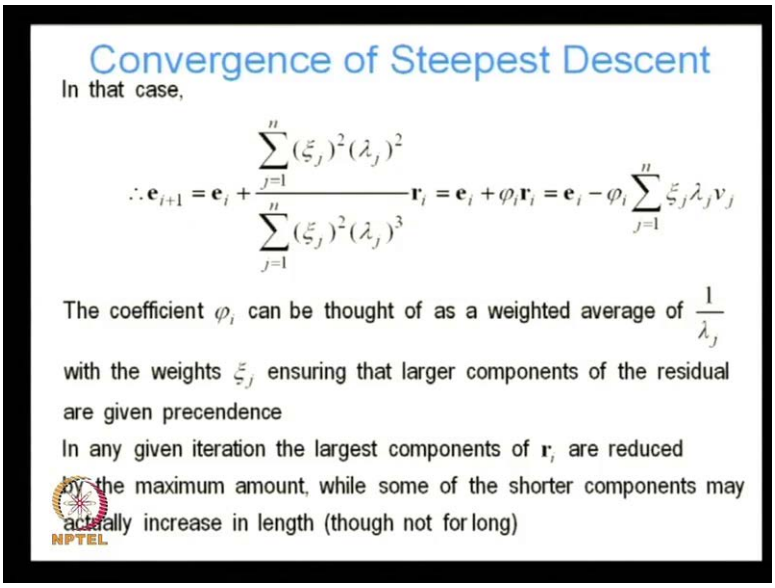
Then this quadratic form this quadratic form it becomes a series of circles instead of ellipse or ellipse solids in multidimensional space or ellipse in two dimensional space it

become circle. So, in a circle where ever you are in a circle, where ever you are in a circle to reach its minimum. You can always take one step to get there, because all points are pointing towards the centre. Suppose, I have this series of circles, in two dimensional space, then if I am at any point at one of those red con tools, if I want to get from here from here to my minimum I just travel along the radial direction.

So, wherever I am there where ever I am I can get to my minimum in one step. Because, I know the direction to travel, which is form here to the centre. I know by how much I must travel, which is the radius. So, I can wherever I am I can get to my minimum in one step. So, people have done a lot of work on this and there very there are papers, which explain these things very beautifully, but I do not want to go too much into that, but this is just an idea right. So, basically the reason why reason why you converge in one increment, one increment, when all the Eigen vector, Eigen values are the same is, because then in n dimensional space the quadratic form instead of being a series of ellipse solids. It become a series of spheres, it become the series of spheres at any point in this sphere, you can go to the minimum just in one step. Because, the direction is known and the amount by which he must travel is also known.

Thus, there is instant convergence. In this case, when all the Eigen values are equal. If the Eigen values are different, there is no choice of alpha i. That will eliminate all the Eigen vector components of the error. There is no choice of alpha i, that will eliminate all the Eigen vector components of the error in one step.

(Refer Slide Time: 31:32)



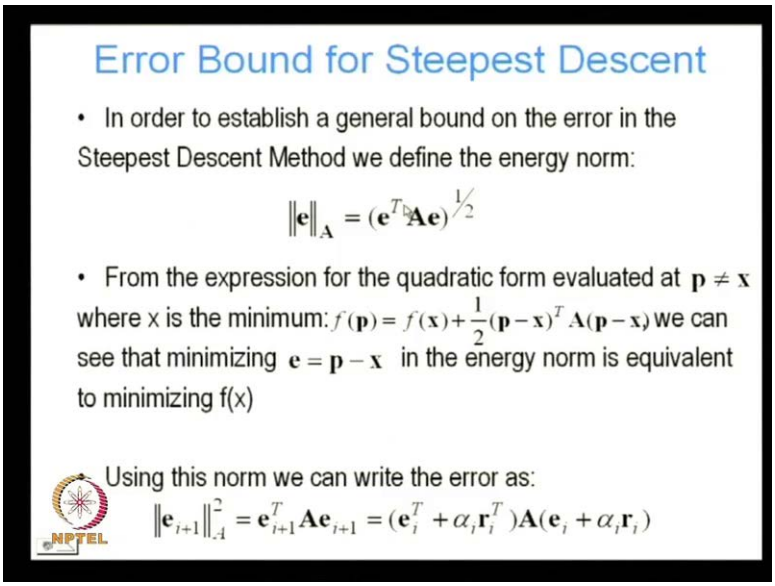
In that case we are stuck with this, we cannot do anything more. So, we are stuck with our previous expression, which was this. So, in that case we have this e i plus 1 equal to e i plus sigma j equal to 1 to n si j square lambda j square divide by sigma j equal 1 to n psi j square lambda j cube r i. This, I am going to write as e i plus some co efficient phi i r I, where pi I can be thought of as a weighted average of one by lambda j, why is that? Well, you have lambda j square on the top you have lambda j cube on bottom, if all these x i j scaling it.

So, this can be thought of as a weighted average of 1 by lambda j of my 1 by lambda j with the weight ensuring that larger component of the residual are given precedence. Because, remember the error is given by psi j v j. So, depending on my psi j, if a particular co efficient x i pi for instance is large, what does that mean? That means, that the error in the directions of the fifth Eigen vector is much larger compare to the error in the direction. In the other Eigen vector error is large then the residual is large, residual is large in that direction. So, what it does is that it scales it, scales it.

So, that the component of the residual large larger components of the residual are given precedence, why? Because, this is scaling the residual, so if pi i is large pi is going to determine, which residual is going to be given precedence. So, e i minus pi r i I am going again as x i j lambda j v j. So, I have that, so in any given iteration the largest component of r i are reduce by the maximum amount. While, some of the shorter component might

actually increase in length for that iteration only, but then eventually it is going to get reduce. So, this pi i this coefficient, it actually scales the residual. So, that i i plus 1 e i plus 1 is found, in such a way that the largest components of the residual get reduce by the maximum amount. It gets scale down by the maximum amount.

(Refer Slide Time: 34:21)



So, that was some discussion on errors, but in order to for this steepest descent method, but in order to obtain a general bound on you. Recall, the Newton Raphson secant for everything we found the bond on the error. We found the order of the convergence, we expressed the error at i plus 1 in terms of the error at i. Whatever, that gave us some idea about the rate of convergence the rate at which the error is going to reduce. So, we want to do the same thing for steepest decent method, but in order to do that, we have to define what is known as the energy norm.

We have to the norm of the error up till now the norms that, we have look at the norms. That we have looked at are nothing but my l two norm. It is the norm, which arises from the inner product, but know we are going to look at norm, which is slightly different, which has a matrix. If you are familiar with the notion of the matrix, it has a matrix. That matrix is the is the matrix a, it is the norm of e with respect to the matrix a. That is given that is define as a energy norm.

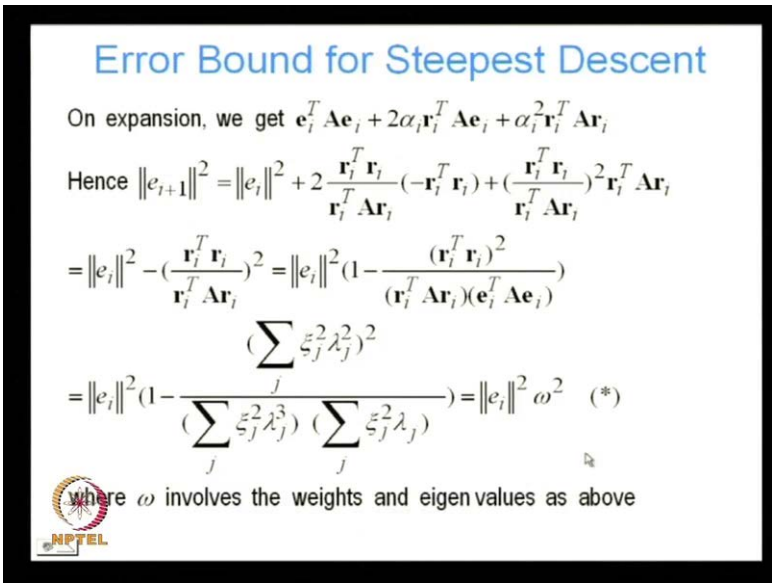
So, this is we define a norm like this e trans it is no longer e transpose A e to the power half. It is given by e transpose A e to the power half, it is a norm with respect to the

matrix a. If we one did not understand the things like matrix, but let us assume that we define a norm like this, e transpose A e to the power half form the co expression for the quadratic form evaluated at p not equal to x. Where, x is the minimum, which we saw last time. If we want to write the expression for the quadratic form at the location p, where p is not equal to x the minimum, we get this expression. We can see that minimizing e is equal to p minus x in the energy norm is equivalent to minimizing f of x.

So, basically if you look at this expression and say that i want to minimize p minus x, what is p minus x? It is the error, this p is some value and x is, where the minimum is. So, when p becomes equal to x i reach a minimum. So, minimizing this is equivalent to minimizing e minimizing the error e, which is equal to p minus x. So, minimizing e equal to p minus x in the energy norm, why is it in the energy norm? Because I am minimizing with to this thing, when I minimize f i minimize this thing is nothing but the norm of e in the energy norm, because e is equal to p minus x. So, minimizing the function is equivalent to minimizing the error in the energy norm minimizing f of x, is the same thing as minimizing the error in the energy norm.

Using this norm, we can write this error as norm of e i plus 1 a square. That is equal to e i plus 1 transpose A e i and e i plus 1 is nothing but e i plus alpha i r I, which we have seen before. We saw that before, we saw it right here, e i plus 1 is equal to e i plus alpha alpha i r i. So, that is this transpose a times this right and then we expand it out.

(Refer Slide Time: 38:44)



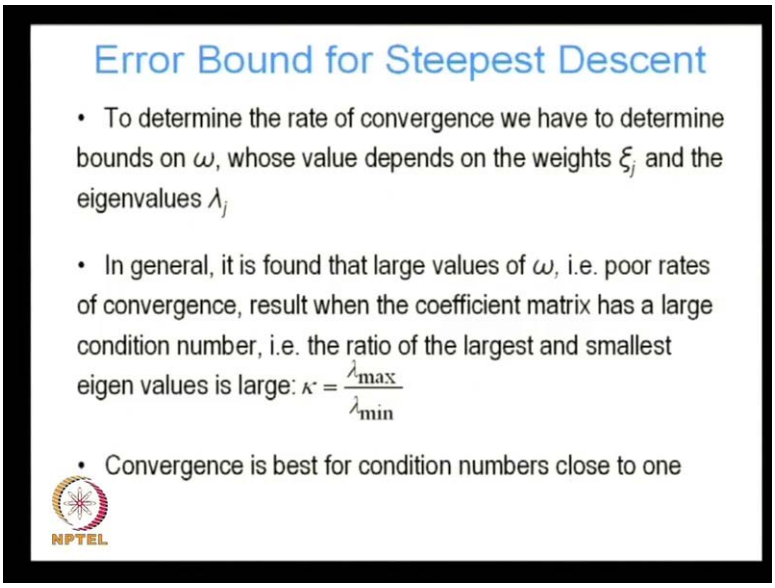
So, when we expand it out, we get e i transpose A e i. Then pull these term together I get the 2 alpha i r i transpose a i plus alpha i square r i transpose A r I, which is expanding this, which is by expanding this expression. This multiplying to going through the dot product and doing that, so we get that. So, we have norm of a i plus 1 square is equal to what is this is nothing but norm of a i square in term, in the in the n with respect to a norm of a i square plus 2 times alpha i. So, for alpha i we substitute the value, which we obtained earlier form my minimization, from my minimization along the line. I got r i transpose r i r i by r i transpose A r i. So, this is my alpha i, then I have r i transpose A e i what is A e i A e i is minus r I, which I saw earlier.

So, I have minus r i transpose r i plus I have alpha i square. So, that is this whole thing square. So, r i transpose r i A r i square r i transpose A r i. So, basically these two term are identical except that there is a negative sign here and two here. So, I get e norm of e i square minus r i transpose r i divided by r i transpose A r i square. That is going to give me norm of i take that out norm of a i square inside the bracket. I have 1 minus r i transpose r i whole square divided by r i transpose A r i. Then I have e i transpose A e i why do I have A e i transpose A e i. Because, I pulled out this and A e i transpose a i is nothing but norm of e i square e i transpose A e i is norm of e i square.

So, I pull that out I get that, then I have I might be missing a square here. So, that is probably true. So, this should be r i transpose A r i square. So, there that miss this for type i. So, I have norm of e i square times 1 minus again these terms. These terms I can replace by those expressions in terms of this i j and the lambda j lambda j is being the Eigen value and i j is being the coefficient of the Eigen vectors, in the expression for the error. So, if I replace those terms with these r i transpose r i with this r i transpose A r i with this e i transpose A e i by this I get norm of a i square times some quantity omega square, where omega square is given by this and omega involves the weights, as well as the Eigen value.

So, now finally, I have been able to right my expression for the error at i plus 1, in terms of the error at i and certain known values. I mean known value, because lambda j lambda j are known. If I know e i I can find out if I know the e I, previous iteration I can find out this i j.

(Refer Slide Time: 42:56)



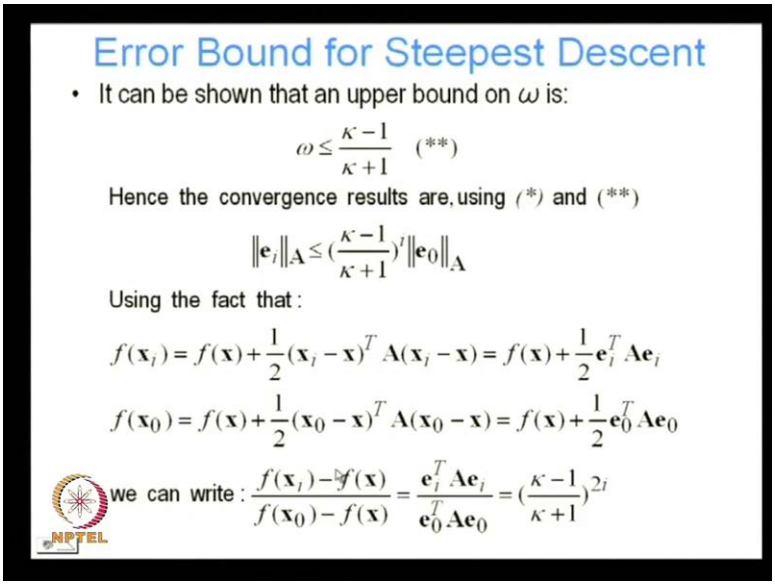
So, to determine the rate of convergence, we have to determine bounce on omega whose value depends on the weight psi j and Eigen value lambda lambda j. So, depending on these value depending on what value omega, take I will have faster slow convergence. If omega is small, what does that mean? That means, the error at e 1 e at i plus 1 is going to be small compare to the error at i. If omega is large what does that mean?

That means, the error is not being reduced as much, so smaller the value of omega the faster my convergence. In general it is found that large values of omega, that is poor rates of convergence result, when the coefficient matrix has a large condition number. That is the ratio of the largest and smallest Eigen value is large you have the Eigen value lambda j. It turns out when the condition number is high, what is the condition number, we define that earlier. It is the ratio of the largest Eigen value of a to the smallest Eigen value of a. When a is in condition, when that condition number is large. In that case omega is also going to be large omega meaning that the rate of convergence is going to be slow.

So, convergence is best for condition number close to one, which we have already seen. If the condition number is equal to 1, what do we get? We get convergence in one iteration largest Eigen value smallest Eigen value both are equal to the same value. Then the condition number is one I converge in one iteration. So, convergence is best for condition numbers close to one close to that, what it brings it together. We saw that,

when you have all the Eigen values equal to 1 you get convergence in one iteration. So, when it close to one you get better convergence, it can be shown that an upper bond omega is we can actually calculate the upper bond in omega.

(Refer Slide Time: 45:24)



So, omega is lesser than or equal to kappa minus 1 by kappa Plus 1, where kappa is my condition number. So, hence the convergence result are using this and this. Using this we can show that norm of e i at using in the a, using the a matrix can be written as kappa minus 1 by kappa plus 1 to the power i times norm of e 0 time a. Well, you can see that is true, because this is equal to e i plus 1 square is equal to norm of e i omega square.

If I take the square root of both sides norm of e i plus 1 is equal to norm of e i omega. Then I do my recursion, I write my norm of e in terms of norm of e i minus 1 e i minus 1 in terms of e minus 2 and so on. So, forth until I get e 0, I know that omega has a certain bound. What is the bounded omega? It is given by this quantity kappa minus 1 by kappa plus 1.

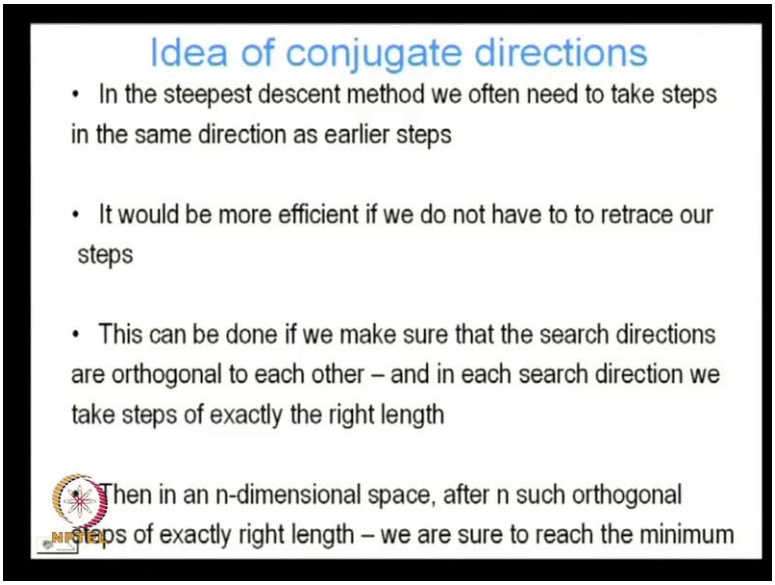
So, the bound on e i is going to be kappa minus one by kappa plus 1 to the power i. Because, I have I recursions every time I get it, scale it by at this bound every time it get scaled by kappa minus 1 by kappa plus 1 kappa. So, norm of at any iteration i it is going to be less than or equal to the original error original error times this. So, this is the absolute upper bound. We can see that it is an upper bounds on the error at any iteration at any iteration i, this steepest decent my error cannot be less than the initial error scaled

by this quantity scaled by this quantity, is that clear? So, using, so then this just this is the simple way to write that, this is what going through.

So, I know that f x i I can write f x i is equal to f of x plus half x i minus x transpose a x i minus x. Definition of quadratic norm at x I, I can write this like this f x plus x half x minus x transpose a x I, which is equal to half is equal to f x. Plus, this is nothing but e i this is the error e i x i minus x. So, half e i transpose A e i. Similarly f x 0 I can write as half f x plus half x 0 minus x transpose a x 0 minus x x 0 minus x is nothing but e 0. So, I get f x is f x plus half is 0 transpose A e 0. Then bring this f x i minus f x i minus f x is equal to e i transpose A e i f x 0 minus f x. We divide this by that after bringing this to the left hand side. So, we have this e i transpose A e i by a 0 transpose A e 0, but this is nothing but the norm of e i norm of e i with respect to the matrix a.

So, that is equal to kappa minus one by kappa plus 1 to the power 2 i. Because, this is the norm square, look we can see from here, norm of e i a by norm of e 0 a is going to give me kappa minus 1 plus kappa plus 1 to the power i. So, this is nothing but the norm of e i square by norm of e 0 square, which is going to give me kappa minus 1 by kappa plus 1 to the power 2 i.

(Refer Slide Time: 49:45)



Next, we want to talk we want to motivate the idea of conjugate direction in this steepest decent method. We often need to take step in the same direction as earlier step. So, look at how we go about this steepest decent method. At any point we move along the

direction of the steepest decent, which is the negative gradient in that direction. We take a step and again, we go if we are thinking of the quadratic. We take a step such that we minimize the function along that direction. When that is minimize, when the residual become when the original residual become orthogonal to the gradient along that direction. We stop there, then again we go on. Then again we calculate the gradient at the new location we move in that direction.

So, we are moving in the solution space right, but there is no guarantee that when we are moving in the solution space we are not retracing our steps. So, If I move in one direction, it may turn out that every after several iteration. I might again be moving in the same direction, which is not efficient. Preferably, what we would like to do in an n dimensional space is that, we would like to move in any one direction. Once minimize the function along that direction. Then we done with it and no more during my iteration process I want to move along that same direction. I want to move along one direction once, only that is probably the most efficient solution, but in the steepest decent method.

There is no guarantee that is going to happen. I might be have to re trace my step, we need to take steps in the same direction. As earlier steps it would be more efficient, if we do not have to retrace our steps. This can be done if we make sure that the search direction are orthogonal to each other. In each search direction, we take the steps of exactly the right length. Now, my search direction are not orthogonal to each other, I am taking. I am moving along the gradient negative gradient here. I am moving along the negative gradient. Here, nobody guarantees that those directions are orthogonal.

So, if I take make sure that I my directions are orthogonal to each other. I need search direction we appt we take we minimize the function. We take the steps of exactly the right length. That is probably going to give me that most optimum gradient based method. Then in n dimensional space after n such orthogonal steps of exactly the right length. We are sure to reach the minimum, we are going to we are obtusely certain that we are going to reach the minimum. So, that would be the most efficient algorithm and that brings us to the method of conjugate direction. So, you want to talk about conjugate direction first and motivate that idea.

We will see that at the end we are going to get to conjugate gradient, but conjugate gradient is nothing but conjugate direction plus 1 minor restriction. So, if we understand

the method of conjugate gradient. If we understand the method of conjugate direction, we understand the method of conjugate gradient. So, I will spent lot of time next class talking about the method of conjugate directions. Hopefully, once we are clear about that understanding the method of conjugate gradient, would be very simple.

Thank you.