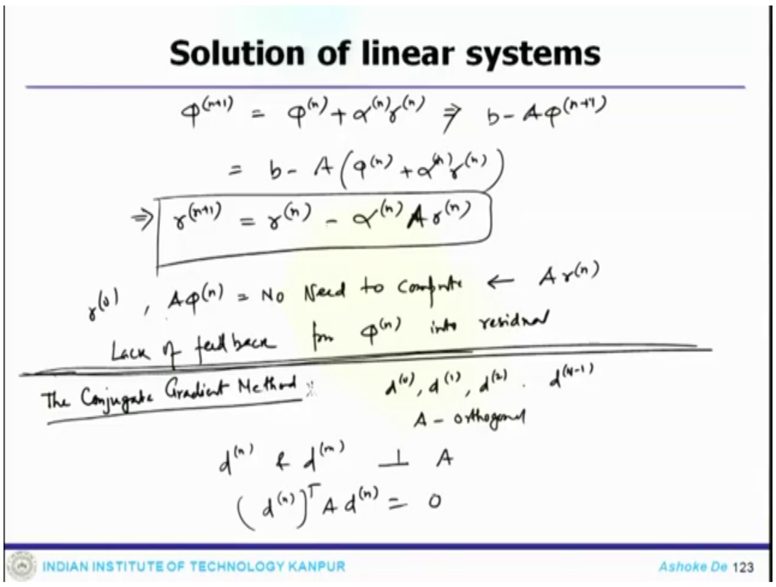
Introduction to Finite Volume Methods – II

Prof. Ashoke De
Department of Aerospace Engineering
Indian Institute of Technology, Kanpur

Lecture – 06
Linear Solvers – VI

So, welcome to the lecture of this Finite Volume Method.

(Refer Slide Time: 00:19)



Solution of linear systems

$$\varphi^{(n+1)} = \varphi^{(n)} + \alpha^{(n)} r^{(n)} \Rightarrow b - A\varphi^{(n+1)}$$

$$= b - A\left(\varphi^{(n)} + \alpha^{(n)} r^{(n)}\right)$$

$$\Rightarrow \boxed{r^{(n+1)} = r^{(n)} - \alpha^{(n)} A r^{(n)}}$$

$r^{(0)}$, $A\varphi^{(n)}$ = No Need to compute $\leftarrow A r^{(n)}$

Lack of feedback from $\varphi^{(n)}$ into residual

The Conjugate Gradient Method: $d^{(0)}, d^{(1)}, d^{(2)} \, . \, d^{(N-1)}$

A – orthogonal

$d^{(n)}$ & $d^{(m)}$ $\perp$ A

$$\left(d^{(n)}\right)^T A d^{(n)} = 0$$

INDIAN INSTITUTE OF TECHNOLOGY KANPUR   Ashoke De 123

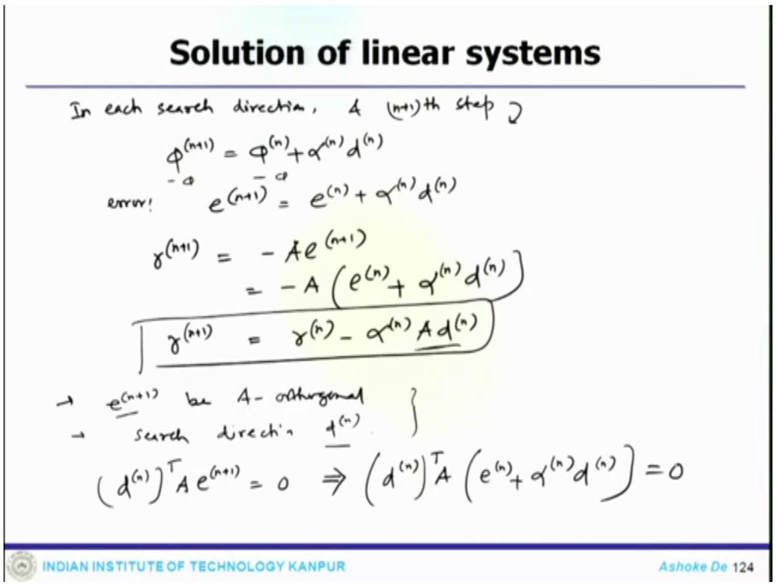
Now, the equation for this r n in step one is needed only to calculate essentially r 0. With this formulation there will be no need to compute A phi n. So, this is no need to compute in this formulation as it is some sort of replaced by A r n. So, that is why, no one, I mean you do not have to or you do not need to calculate this particular term directly as it is replaced.

However, some short comings are there in this particular approach is the lack of feedback from the so, the lack of feedback from phi n from the previous iteration value the feedback is not that into residual. So, which may cause some the solution to converge to a very different from the exact one due to accumulation of some sort of an rounding error. So, this is one of the deficiency of this particular approach can be I mean obtained with a modified or better version.

Now, the other class of system is the conjugate gradient method. So, that is the other class of system while the previous one what we have discussed is the steepest descent method and the present one that we are going to talk about the conjugate gradient method. The steepest gradient method one property was there that it will always guarantee the convergence, but rate could be low. And that slow convergence is caused due to some sort of an oscillations around that local minima forcing the method to search for the same direction only.

So, that because of that algorithm so, to avoid that, some undesirable behaviour can be seen and while doing that search in the direction for different directions for the previous one. So, this can be accomplished by selecting a set of search direction like d 0, d 1 d 2 and so on d n minus 1, that A becomes orthogonal. So, two vectors that d n and d m are said to be orthogonal to A if they satisfy some conditions like d transpose A d n is 0.

(Refer Slide Time: 04:01)



Solution of linear systems

In each search direction, & (n+1)th step ?

$$\phi^{(n+1)} = \phi^{(n)} + \alpha^{(n)} d^{(n)}$$

error! $$e^{(n+1)} = e^{(n)} + \alpha^{(n)} d^{(n)}$$

$$r^{(n+1)} = -Ae^{(n+1)}$$
$$= -A\left(e^{(n)} + \alpha^{(n)} d^{(n)}\right)$$

$$\boxed{r^{(n+1)} = r^{(n)} - \alpha^{(n)} \underline{A d^{(n)}}}$$

→ $\underline{e^{(n+1)}}$ be A- orthogonal
→ search direction $\underline{d^{(n)}}$

$$\left(d^{(n)}\right)^T A e^{(n+1)} = 0 \Rightarrow \left(d^{(n)}\right)^T A \left(e^{(n)} + \alpha^{(n)} d^{(n)}\right) = 0$$

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

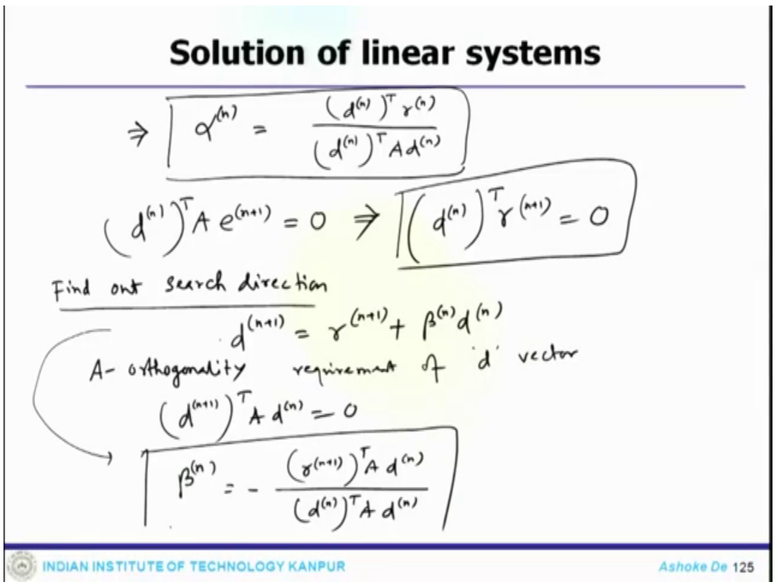
Ashoke De 124

So, now in each search direction in each search direction the right step size is taken and the solution would be found after n steps and n plus 1 step can be written as phi n plus 1 equals to phi n plus alpha n d n. Now, you subtract this one from this one phi from both the sides you get the error and the error at n plus 1 level is so, you subtract the phi from both the sides it gets e n plus alpha n d n.

Now, you put these things together for the residual vector. So, that will get you minus A e n plus 1 which is nothing, but A e n plus alpha n d n that will let you r n minus alpha n A d n. So, that is what you get and in this case the it shows that the new residual at the n plus 1 iteration level is just an linear combination of the previous residual and this guy A d to the power n.

So, now, one has to establish another condition is that this error function e n plus 1 this has to be all also A orthogonal. This also needs to be required. Now, this new condition is also some sort of an equivalent condition to find out the minimum point along the search direction of d n. So, now, using this A, orthogonality condition between e and d n, one can express this condition as d n transpose A e n plus 1 equals to 0 which will be d n transpose A e n plus alpha n d n.

(Refer Slide Time: 07:35)



Solution of linear systems

$$\Rightarrow \boxed{\alpha^{(n)} = \frac{(d^{(n)})^T r^{(n)}}{(d^{(n)})^T A d^{(n)}}}$$

$$(d^{(n)})^T A e^{(n+1)} = 0 \Rightarrow \boxed{(d^{(n)})^T r^{(n+1)} = 0}$$

Find out search direction

$$d^{(n+1)} = r^{(n+1)} + \beta^{(n)} d^{(n)}$$

A- orthogonality requirement of 'd' vector

$$(d^{(n+1)})^T A d^{(n)} = 0$$

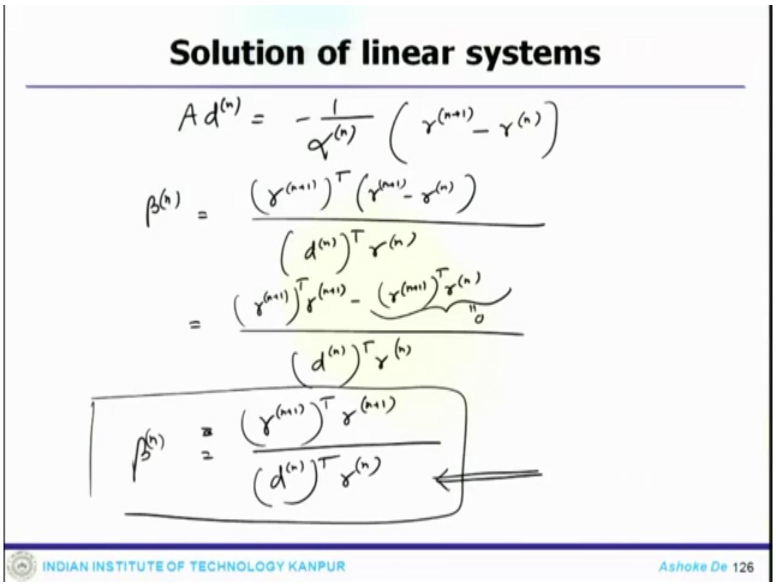
$$\boxed{\beta^{(n)} = -\frac{(r^{(n+1)})^T A d^{(n)}}{(d^{(n)})^T A d^{(n)}}}$$

INDIAN INSTITUTE OF TECHNOLOGY KANPUR Ashoke De 125

Now, which is essentially going to get us alpha n equals to d n transpose r n and d n transpose A d n. So, that is what you get for alpha n. So, this requirement also implies that it implies that d n of transpose A error function at n plus 1 is 0 which is nothing, but d n transpose r n plus 1 equals to 0. So, if the search directions are known; that means, d 0, d 1, d 2, d n then alpha n can be straight away calculated.

Now, the task is to find out find out search direction. So, once we know the search direction then alpha n can be calculated. Now, to find out the search direction one can write this expression for d n plus 1 equals to r n plus 1 plus beta n and d n. Now, A orthogonality this is a condition and this requirement of d vector. So, A orthogonality requirement of d vector implies that d n plus 1 transpose A d n must be 0. Now, you substitute this one now you put this one here and combine these two one can get the beta n equals to minus r n plus 1 transpose A d n divided by d n transpose A d n. So, that is what you get the beta n.

(Refer Slide Time: 10:27)



Solution of linear systems

$$A d^{(n)} = -\frac{1}{\alpha^{(n)}}\left(r^{(n+1)} - r^{(n)}\right)$$

$$\beta^{(n)} = \frac{\left(r^{(n+1)}\right)^T \left(r^{(n+1)} - r^{(n)}\right)}{\left(d^{(n)}\right)^T r^{(n)}}$$

$$= \frac{\left(r^{(n+1)}\right)^T r^{(n+1)} - \underbrace{\left(r^{(n+1)}\right)^T r^{(n)}}_{\overset{\shortparallel}{0}}}{\left(d^{(n)}\right)^T r^{(n)}}$$

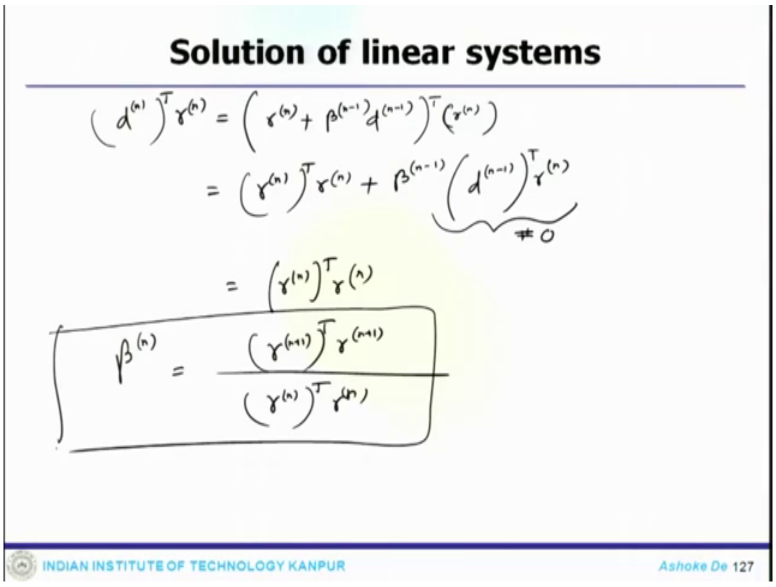
$$\boxed{\beta^{(n)} = \frac{\left(r^{(n+1)}\right)^T r^{(n+1)}}{\left(d^{(n)}\right)^T r^{(n)}}} \Longleftarrow$$

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

Ashoke De 126

Now, you can express your A d n in using this information you can express A d n 1 by alpha n and minus r n. So, you get this. Now, you combined all these expression for alpha n d n, A d n and all these things together you get the expression for beta n which would be now r n plus 1 transpose r n plus 1 minus r n divided by d n transpose r n which is writing like r n plus 1 transpose r n plus 1 minus r n plus 1 transpose r n and d n transpose r n and this guy is essentially 0. So, this boils down to r n plus 1 transpose r n plus 1 divided by d n transpose r n. That is what one get for beta.

(Refer Slide Time: 12:31)



Solution of linear systems

$$(d^{(n)})^T r^{(n)} = \left(r^{(n)} + \beta^{(n-1)} d^{(n-1)}\right)^T (r^{(n)})$$

$$= (r^{(n)})^T r^{(n)} + \beta^{(n-1)} \underbrace{(d^{(n-1)})^T r^{(n)}}_{\neq 0}$$

$$= (r^{(n)})^T r^{(n)}$$

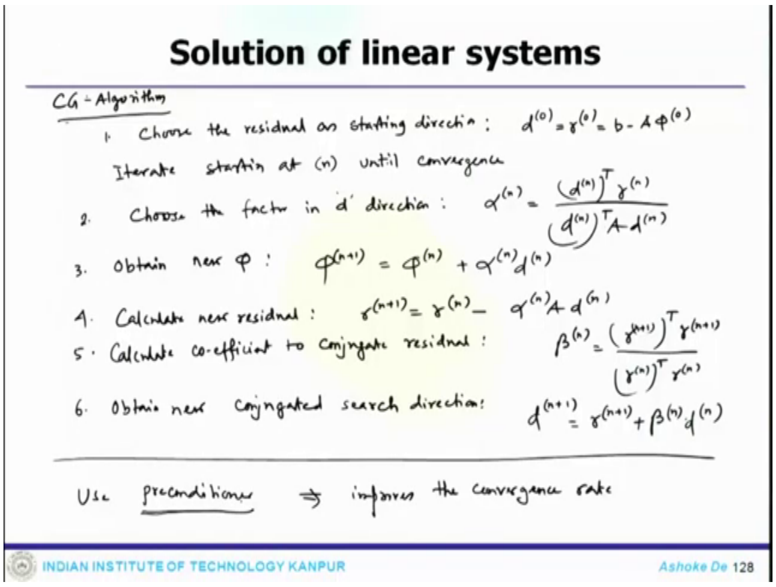
$$\boxed{\beta^{(n)} = \frac{(r^{(n+1)})^T r^{(n+1)}}{(r^{(n)})^T r^{(n)}}}$$

INDIAN INSTITUTE OF TECHNOLOGY KANPUR Ashoke De 127

Now, the denominator which is sitting here in the expression of beta it can be further extended or it can be further expressed like d n transpose r n which is equivalent to r n plus beta n minus 1 d n minus 1 transpose r n. So, that is r n transpose r n plus beta n minus 1 d n minus 1 transpose r n. This component is or equals to 0. So, this by becomes r n transpose r n. So, one can rewrite beta n equals to you can see r n plus 1 transpose r n plus 1 by r n transpose r n.

So, that is a nice expression based on the residual vector one can find out the beta.

(Refer Slide Time: 14:05)



Solution of linear systems

CG - Algorithm

1. Choose the residual as starting direction : $d^{(0)} = r^{(0)} = b - A\phi^{(0)}$

Iterate startin at (n) until convergence

2. Choose the factor in 'd' direction : $\alpha^{(n)} = \frac{(d^{(n)})^T r^{(n)}}{(d^{(n)})^T A d^{(n)}}$

3. Obtain new ϕ : $\phi^{(n+1)} = \phi^{(n)} + \alpha^{(n)} d^{(n)}$

4. Calculate new residual : $r^{(n+1)} = r^{(n)} - \alpha^{(n)} A d^{(n)}$

5. Calculate co-efficient to conjugate residual : $\beta^{(n)} = \frac{(r^{(n+1)})^T r^{(n+1)}}{(r^{(n)})^T r^{(n)}}$

6. Obtain new conjugated search direction: $d^{(n+1)} = r^{(n+1)} + \beta^{(n)} d^{(n)}$

Use <u>preconditioner</u> $\Rightarrow$ improves the convergence rate

INDIAN INSTITUTE OF TECHNOLOGY KANPUR Ashoke De 128

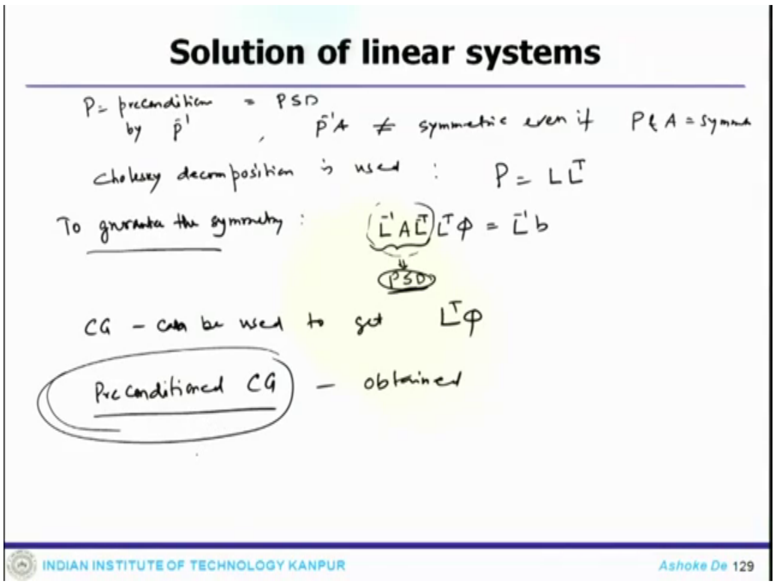
So, if you find out the beta in that way then you can put now, if someone has to look at the algorithm for CG conjugate gradient algo so, that would become now you first choose the residual as starting direction like d naught equals to r naught equals to b minus A phi naught and you iterate start in at n unless or until convergence.

So, you keep on doing the iteration till you obtain the convergence. So, the next level you choose the factor in d direction like alpha n equals to d n transpose r n divided by d n transpose A d n. So, this is the way you choose the factor in d direction. Then you obtain new phi such that phi n plus 1 equals to phi n plus alpha n d n. Now, after obtaining the phi you need to cross check for the residual. So, you can calculate new residual like r n plus 1 equals to r n minus alpha n A d n, that is how you calculate the new residual used in the new value of the variable phi.

And, then one has to calculate the coefficient to conjugate residual. How? Now here you get beta n equals to r n plus 1 transpose r n plus 1 divided by r n transpose r n. So, that is how you calculate the coefficient to conjugate residual and then you finally, obtain the new conjugated search direction like d n plus 1 equals to r n plus 1 plus beta n d n. So, this is how you obtain the calculations.

So, the one important thing is that also in this particular case or rather this kind of this class of methods if you use preconditioner, then it can actually improves the convergence rate improve the convergence rate. So, that is where one can see the utility or applicability of the preconditioner is also very handy in the c g algorithm.

(Refer Slide Time: 19:27)



Solution of linear systems

P = preconditioner = PSD

by P^{-1}, $P^{-1}A \neq$ symmetric even if P & A = symmetric

Cholesky decomposition is used : $P = LL^T$

To guarantee the symmetry : $\underbrace{(L^{-1}AL^{-T})}_{\boxed{PSD}} L^T\phi = L^{-1}b$

CG - can be used to get $L^T\phi$

Preconditioned CG - obtained

INDIAN INSTITUTE OF TECHNOLOGY KANPUR Ashoke De 129

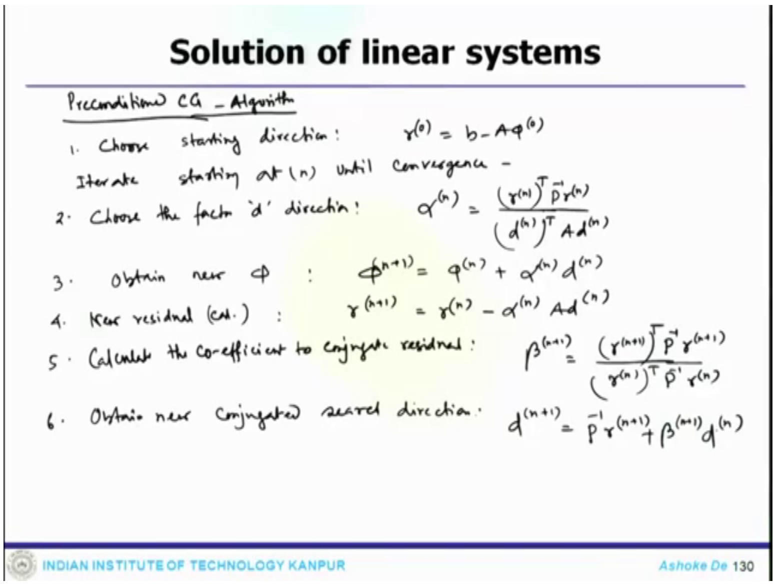
Now, if you have a preconditioner and so, you have a P is a preconditioner and this is also positive symmetric definite matrix and you multiply the original equations by P prime, then the problem is that P inverse a is not necessarily this is not necessarily symmetric even if even if P and A both are symmetric. So, that is the problem.

So, to overcome this problem the Cholesky decomposition is used to write P in the particular form. So, the Cholesky decomposition is used and that is written like P could be LL transpose; that means, and that to guarantee the symmetry to guarantee the symmetry the system equation can be written as L inverse AL transpose L minus transpose L transpose phi equals to L inverse b where your L inverse A L inverse transpose this component particularly this component is positive symmetric definite. So, this is not only symmetric, it is also positive definite.

So, now the CG method can be. So, once you use this kind of Cholesky decomposition to get the preconditioner. Now, this is positive symmetric definite. So, and now here CG method can be used to solve for the now the CG can be used to get L transpose phi for which the phi can be found. Now, for variable substitutions L can be eliminated from the equations without disturbing the symmetry of or affecting the validity of the method. So, performing this kind of step and adopting the terminology used with the CG method the various steps in the preconditioner pre conditioned conjugate gradient method that can be obtained.

So, essentially in top of the CG you use some sort of a preconditioner to get a preconditioner or precondition CG. Now, what happens to the original algorithm now? Now, previously we have seen the algorithm for the conjugate gradient. Now, what happens to the original algorithm?

(Refer Slide Time: 23:07)



Solution of linear systems

Preconditioned CG – Algorithm

1. Choose starting direction: $r^{(0)} = b - A\phi^{(0)}$

Iterate starting at (n) until convergence –

2. Choose the factor 'd' direction: $\alpha^{(n)} = \frac{(r^{(n)})^T P^{-1} r^{(n)}}{(d^{(n)})^T A d^{(n)}}$

3. Obtain new ϕ : $\phi^{(n+1)} = \phi^{(n)} + \alpha^{(n)} d^{(n)}$

4. New residual (cal.) : $r^{(n+1)} = r^{(n)} - \alpha^{(n)} A d^{(n)}$

5. Calculate the co-efficient to conjugate residual: $\beta^{(n+1)} = \frac{(r^{(n+1)})^T P^{-1} r^{(n+1)}}{(r^{(n)})^T P^{-1} r^{(n)}}$

6. Obtain new conjugated search direction: $d^{(n+1)} = P^{-1} r^{(n+1)} + \beta^{(n+1)} d^{(n)}$

INDIAN INSTITUTE OF TECHNOLOGY KANPUR Ashoke De 130

For pre conditioned CG algo. So, first is that again you have to choose starting direction. So, that is important to get that you calculate r naught b minus a phi naught now you iterate starting at n until convergence. So, that is the criteria. So, the next step you do that you choose the factor d direction like alpha n equals to r n transpose P inverse r n divided by d transpose A d n.

Now, the third step is that you obtain new phi, such that phi n plus 1 equals to phi n plus alpha n d n. Fourth step you calculate the new residual. So, calculate that like r n plus 1 equals to r n minus alpha n A d n. Then you calculate essentially the coefficient; coefficient to conjugate residual like beta n plus 1 equals to r n plus 1 transpose r n plus 1 divided by r n transpose r n you get P inverse here. So, that is where your P conditioning system would come.

So, here you get the P inverse r n plus 1 and here you get P inverse r n and then you finally, obtain new conjugated search direction like. So, this is n plus 1. So, conjugated search direction like d n plus 1 is P inverse r n plus 1 plus beta n plus 1 d n.

Now, if you look at from the previous step of conjugate gradient, here you get the new conjugated search directions using beta n and d n and there is no preconditioner. But, when you look at this preconditioning system this preconditioners are coming to the picture. So, these are actually developed with a wide spectrum of sophistication with varying diagonal matrix whose elements are also diagonal and for like Jacobi preconditioner and other things.

So, essentially these are using the different kind of methods or different kind of preconditioning CG methods and the different factorizations would lead to different class of problems. So, we will stop here today and discuss the other things in the subsequent class.

Thank you.