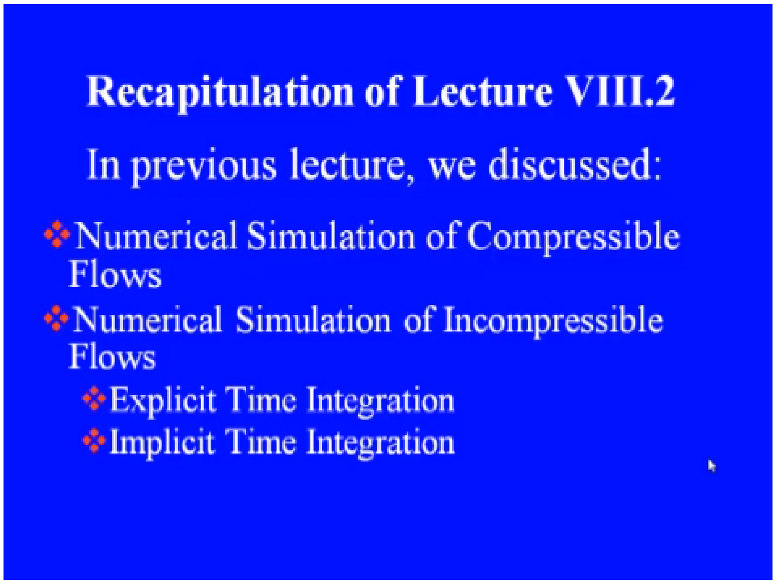
Computational Fluid Dynamics
Dr. Krishna M. Singh
Department of Mechanical and Industrial Engineering
Indian Institute of Technology – Roorkee

Lecture - 37
Implicit Pressure Correction Methods

Welcome to the third lecture in module 8 on Numerical Solution of Navier-Stokes Equations. In this module we have already discussed the basic features of Navier-Stokes Equations. We discussed explicit and implicit time integration algorithms. And today we would focus on implicit pressure correction methods. And in the next lecture we will take up fractional step methods. Before we proceed further let us have a recapitulation of what we discussed in the previous lecture.

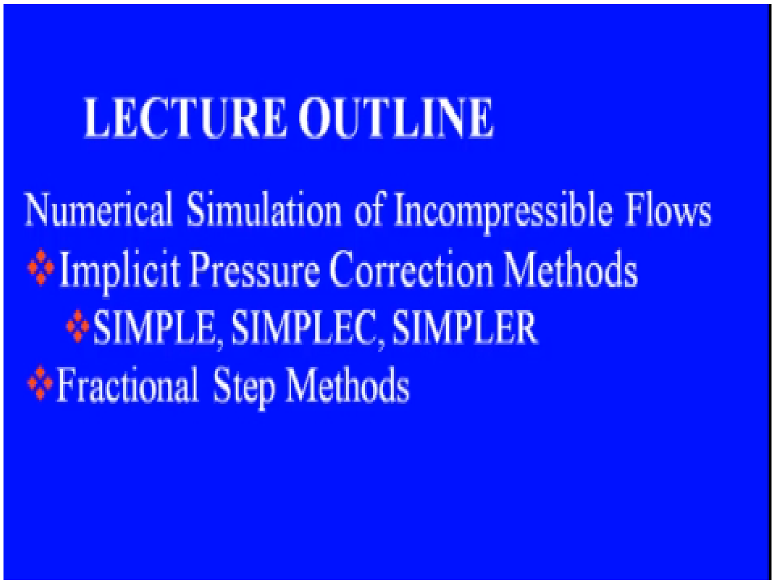
(Refer Slide Time: 00:55)



We briefly discussed the numerical simulation of compressible flows. And then we took up the algorithms of numerical simulation of incompressible flows. We had a look at explicit time integration methodology and implicit time integration methodology wherein we saw that we have to solve a very large system of coupled nonlinear equations in case we use an implicit time integration technique.

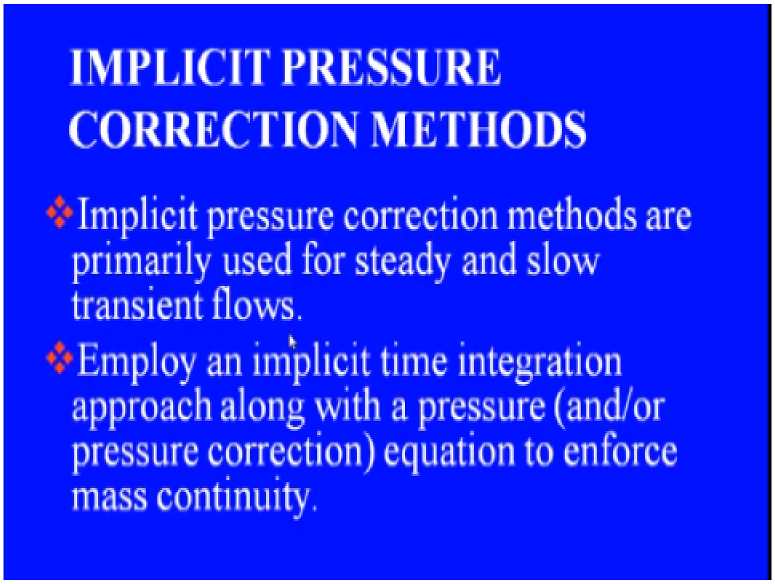
Implicit time integration techniques are required if you want to solve slow transient or steady state problems. So is there a way out in which we can avoid solving a very large system of coupled nonlinear equations accurately. So implicit pressure correction methods which we discussed in this lecture provide an answer wherein we would develop an iterative technique to solve the coupled nonlinear system of equations.

(Refer Slide Time: 01:59)



So we will first discuss very briefly the basic feature of what we call implicit pressure correction methods. And then we will take up few popular algorithms of this category in particular in SIMPLE, SIMPLEC and SIMPLER. If time permit, we will also take up Fractional Step Methods or else we will take this topic up in the next lecture.

(Refer Slide Time: 02:27)

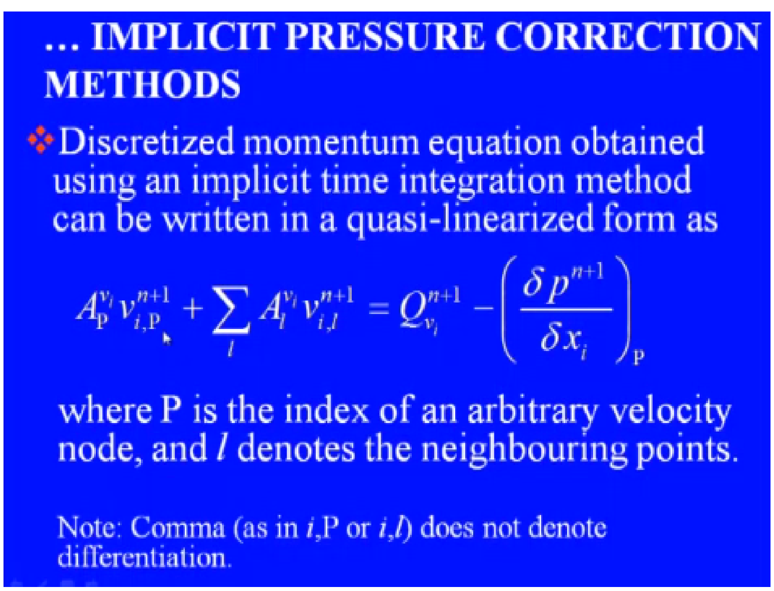


Now this Implicit Pressure Correction methods they are primarily used for steady and flow transient flows. And once again the basic process is very similar to what we discussed earlier in the case of explicit time integration techniques, that to say we would first find out a velocity field which does not satisfy the continuity equation then we will solve a Poisson equation for pressure and incorporate a correction term which will enforce a continuity thereby obtaining a velocity field which is divergence free.

So implicit pressure correction method follow same procedure and they belong to the general clause of method which we call projection methods wherein the first project of velocity to get an intermediate velocity field which does not satisfy continuity and corrected using pressure like tau. And we will employ an implicit time integration like explicit time integration techniques along with a pressure and or a pressure correction equation to enforce mass continuity.

We will first discuss a scheme which basis the corrections on the pressure equation and then we will take up simple algorithm which is based on the pressure correction. Use a pressure correction to enforce the continuity equation and simpler algorithm which would involve both pressure and pressure correction solution. Now in the last lecture we had written or we had obtained the discrete momentum equation which we would obtain from spatial discretization, in fact it was a semi-discrete equation in time.

(Refer Slide Time: 04:12)

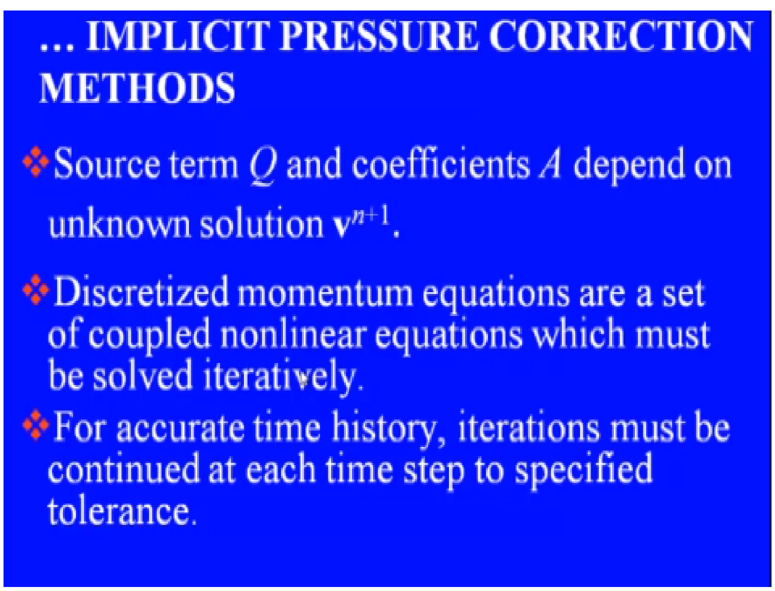


And when we applied an implicit time integration such as Backward Euler we got a system of nonlinear equation. Now that nonlinear momentum equation that can be written in a quasi-linear form as Ap Vi Vi,p n+1+Sigma l, Al vi Vi subscript, l n+1=Qvi n+1-delta pn+1/ delta xi p, but this delta p/del xi this represents the discretization for the pressure term.

Now here the p is the index of an arbitrary velocity node at a given computational node, l or the indexes of neighboring nodes which contribute to that momentum equation and I could take values from 1 to 3 corresponding to 3 momentum equations. Now remember here this Q terms and as they will depend on the unknown velocity field. One more thing we have been using is comma notation we have used earlier to denote differentiation but at not in this equation.

Here i,p that only say that we are dealing with the ith component of velocity or ith momentum equation and capital P subscript that denotes a particular computational node. Similarly, i,l this comma again does not denote differentiation simply says that we are looking at the ith momentum equation for the neighboring node l , so here comma does not denote differentiation. Now the source term Q and coefficients A depend on unknown solution Vn+1; so we can diverge an iterative scheme we have to work or we have to use the values which are known till now.

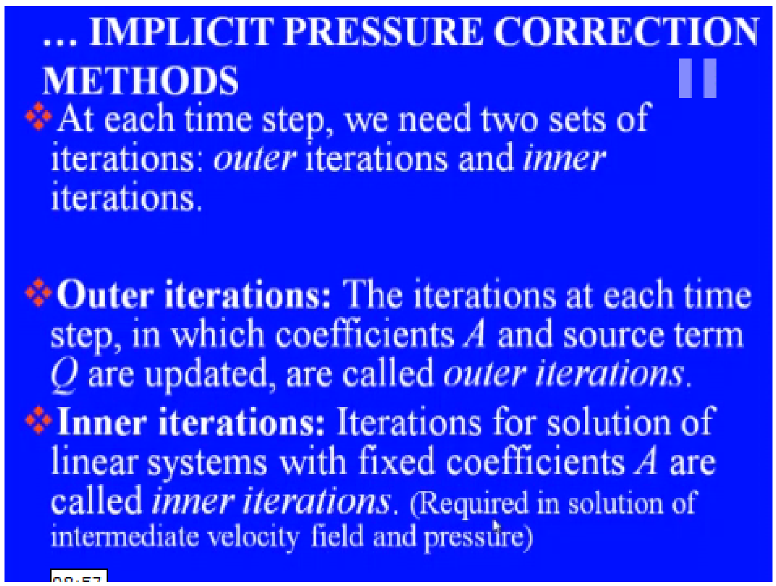
(Refer Slide Time: 06:16)



Now these discretized momentum equations they represent a set of coupled nonlinear equation because we have three such equations for each value of i and we have to solve these equations iteratively. And in case if you are looking for accurate time history what we will have to do is we have to continue the iteration at each time step for this nonlinear system to specified tolerance.

Now in this context what do we do, at each time step we have to solve our nonlinear system and usually we are dealing with a very large system of linear equations or what we call linearized equations, so we would use two sets of iterations one is what we call outer iterations and other set of iteration where refer to as inner iterations. So let us clarify these terms.

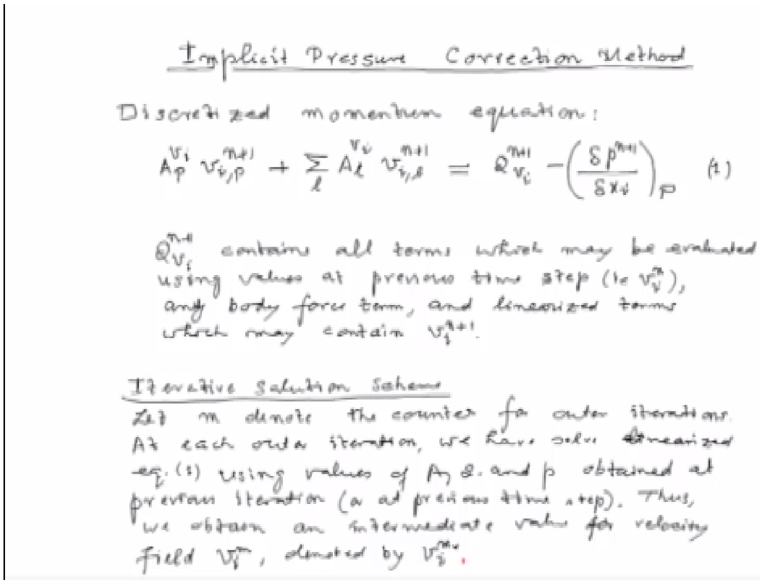
(Refer Slide Time: 07:08)



The outer iterations are the ones the iterations at each time step in which our coefficients A and the source term Q are updated based on the velocity field and the pressure field computed at the previous iterations. So these iterations wherein these coefficients A and Q are updated they are called outer iterations. Now what are Inner iterations? Iterations for solution of linear systems with fixed coefficients A are called inner iterations.

Now these inner iterations are required in this solution of a velocity field from linearized equations or we would also use very often an iterative solution procedure to solve the pressure Poisson equation. So these equations are required which are involved in the solution of the linear algebraic equations for velocity field and the pressure field these would be referred to as our inner iterations. Now let us derive the algorithm before we look at the summary of this particular scheme.

(Refer Slide Time: 08:24)



Okay so let us rewrite our discretized momentum equation. Discretized and also quasi-linear for so we have this ApVi Vi,p n+1+sigma l Al Vi Vi,l n+1=QVi n+1-delta pn+1 over delta xi computational node p, let us call this equation as 1. So we will have a set of three such coupled equations and in this quasi-linear form we have taken one precaution that is to say we will put all the terms on the left hand side for a particular velocity component (()) (10:20) = 1 rest of them they would be put together in this term QVi.

So this QVi n+1 contains all the terms which are evaluated using previous values—which may be evaluated using values at previous time step that is our known values that is Vi at m any, body force term and linearized velocity terms or linearized terms which me may contain in fact they would (()) (11:43) contain that it unknown velocity term that is the reason here we have used the superscript n+1 to denote that even this Q term which is sort of our load term for a linearized system.

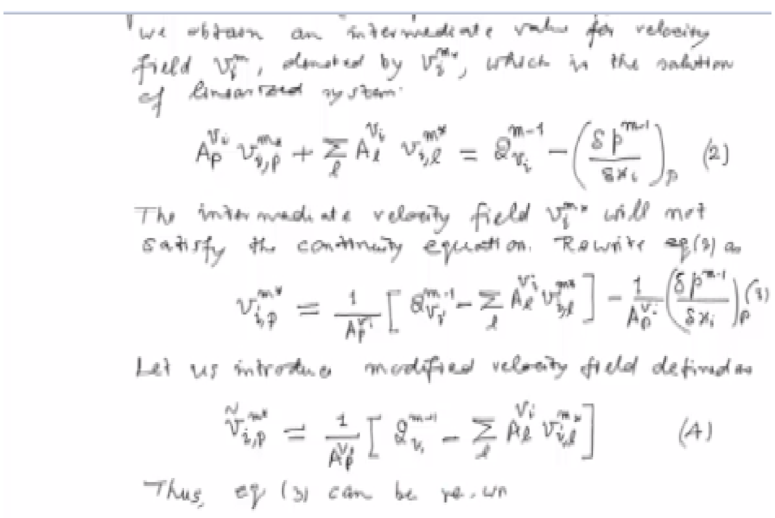
That also depends on 8 unknown values. Now to start off with iteration process we want to diverge an iterative solution scheme and let us symbol m to denote our outer iteration, so let m denote the counter for outer iterations. So what we will try to do is that at each iteration we will have to solve this linearized system using values which are already available that is which have been evaluated at the current iteration which are available to us.

So at each outer iteration we have to solve linearized equation 1 using values of A, Q and P okay obtained at previous iteration or if we at the very first iteration we will say at previous time step. So the start of the outer equations our known values are the values at time t=tn, so those would be used as our initial guess for the field values at the new time step. And we would solve for an intermediate velocity field.

Since on the right hand side we are using the terms which are based on the previous iteration and if we substitute those (()) (14:29) equation they will not satisfy the momentum equation 1 and the velocity field which we obtained thereby that will not satisfy the continuity equation either. So this, what do we do, that would solve for or thus, we do not try to immediately obtain an estimate of the velocity field at the current iteration which will satisfy continuity.

We would rather obtain a velocity field estimate which does not satisfy continuity and we would enforce the continuity later on.

(Refer Slide Time: 15:51)



So thus we obtained an intermediate value for velocity field V i m let us denoted by Vi m* which is the solution of ApVi Vi,p m*+sigma l AlVi Vi,l m* in equation 1 we have just substituted m* in place of superscript n+1 QVi m-1-delta p m-1 over delta xi at p. So let us call this equation as

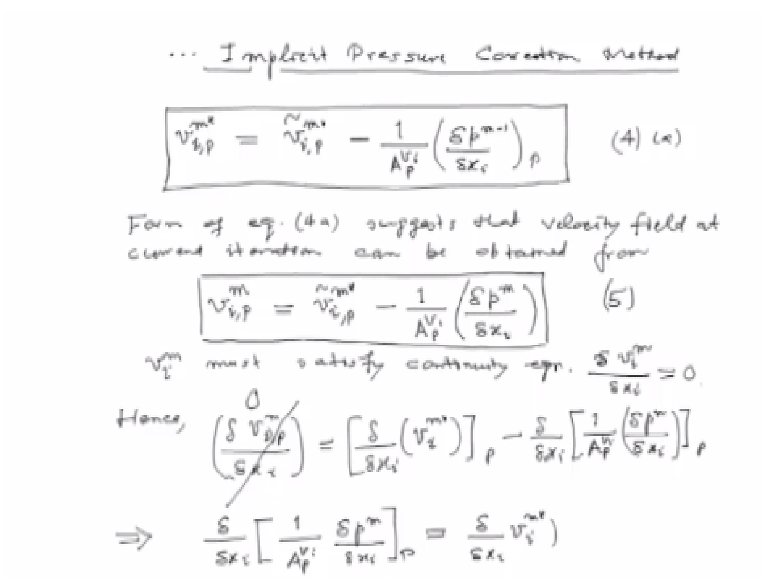
2. In fact, this represents not a single equation it represents a system of equations which have to solved and if you can solve the previous system we can get our intermediate velocity field Vi m*.

And let us implicit this point that intermediate velocity field Vi m* will not satisfy the continuity. Okay, now what do we do to enforce the continuity lets rewrite this equation slightly and that would give us an equation which we can use as your starting point to obtain a corrected velocity value which would satisfy the continuity equation. So let us rewrite the equation 2, let us keep only Vi, p m* in one side so Vi,p m*=1 over ApVi (Qvi m-1-sigma l Al Vi Vi, l m*-1 over Apvi delta p m-1 over delta xi at p. Let us call this equation as equation 3.

This equation will not actually be used to compute anything this is just an intermediate step in our derivation process. And let us introduce (()) (19:31) the modified velocity field. So let us introduce velocity field defined as, we will call it Vip m* delta and in fact this is shorthand notation for the first term on the right hand side of the equation 3, this is 1 over Apvi Qvi m-1-sigma l AlVi Vi,l m*.

Let us call this definition as our equation 4. So in terms of this modified field so thus our equation 3 can be rewritten as, let us go to the next.

(Refer Slide Time: 21:09)



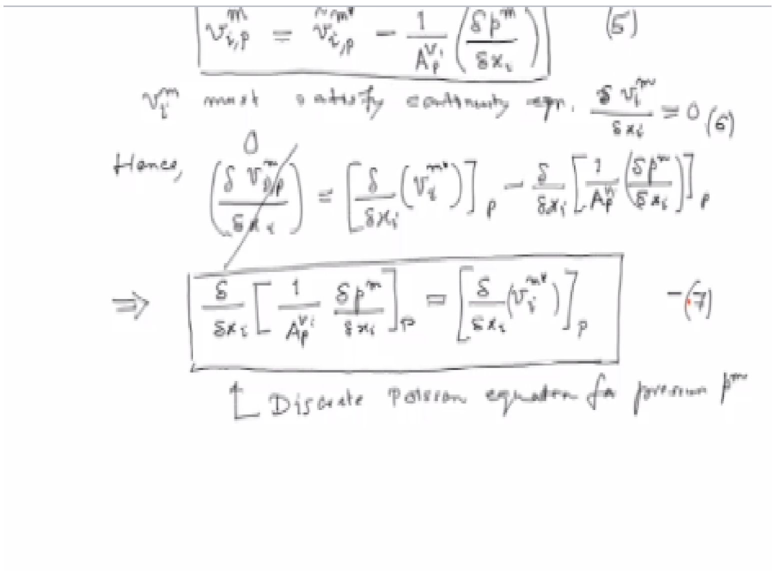
So Vi,p m* is given by Vi,p m* ~ -1 over ApVi delta p over delta xi m-1 at point p. Now it is not this equation per se very important. What is important in the development of algorithm is the form of this equation; let us call this equation as 4a. Now what this equation suggests is that if we had the pressure value at the current instant that could be used in this equation and if you could have used that we should be able to obtain a corrected value for the velocity field that is the velocity field at the current iteration.

So this form of equation 4a suggest that velocity field at current iteration can be obtained from Vi,p m, now we dropped—there is no more the intermediate velocity, what we are claiming that this would be our continuity satisfying velocity field at the current outer iteration m, so Vi,p m*~ -1 over ApVi delta p over delta xi m. Now here the pressure is no more the known pressure.

In fact, this is an unknown pressure value at the current iteration which we have got to find out. Okay, the first term is known because we can obtained Vi m* from the solution of our linearized system. So first term is known. Now let us enforce the continuity. So this Vi m must satisfy our continuity equation that is delta Vi m over delta xi=0, remember we are dealing with incompressible flows, hence let us take divergences of our equation 5, so what do we get?

The first term, delta Vi,p m over delta xi delta over delta xi of Vi m* at point p-delta over delta xi of 1 over ApVi delta P m over delta xi first term should be zero because Vi m satisfy the continuity equation. So this leads to our discrete pressure Poisson equation delta over delta xi of 1/ApVi delta P m over delta xi computational node so p=delta over delta xi Vi m*.

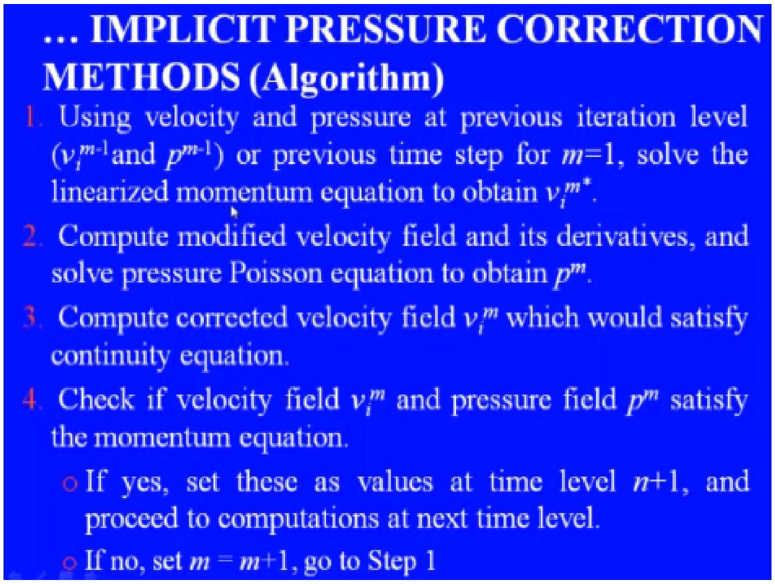
(Refer Slide Time: 26:27)



So this is our pressure Poisson equation; this is Discrete Poisson Equation for pressure pm. Let us call this equation 7. So now let us have look at the summary of the equation which we have got. We had linearized momentum equation, equation 2 which was for the intermediate velocity field Vi m* so we can solve that equation using values of the velocity at the previous iteration and the pressure at previous iteration.

So Vi m* is known. Next we could not solve for our discrete Poisson equation for pm and now this pm can be used in conjunction with Vi m* to obtain the velocity field at the current iteration which would satisfy our continuity equation. Okay. So now let us summarize these steps and put this algorithm formal terms.

(Refer Slide Time: 28:01)

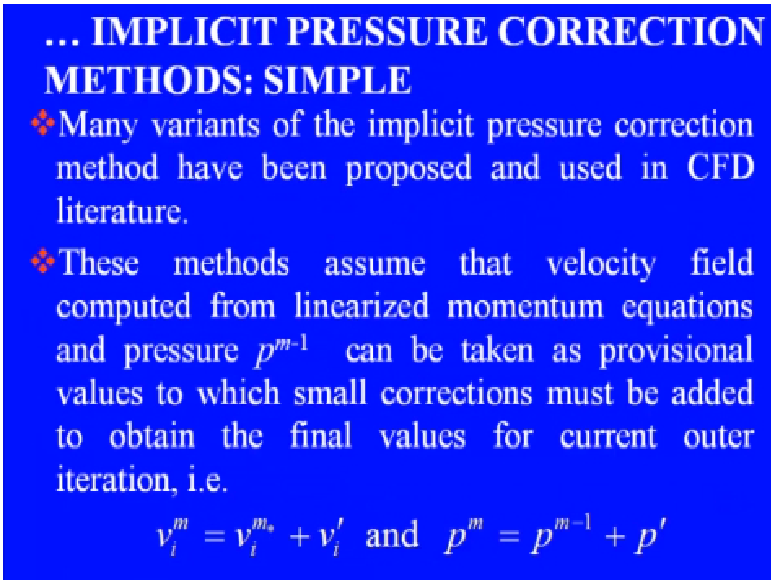


So let us-- using velocity and pressure at previous iteration level that is Vi m-1 and p m-1 or previous time step m=1 solve the linearized momentum equations to obtain Vi m*. And once this Vi m* is known, okay. Now we will compute the modified velocity field Vi m*~ find out its derivatives because that is what could be used in our pressure Poisson equation. Okay. And once that is known then we can compute the corrected velocity field Vi m which would satisfy the continuity equation.

Next step what would be to check if our velocity field Vi m and pressure field pm they satisfy momentum equation to the specified tolerance. If the answer is Yes, then we will say this Vi m and pm now represent the values of the current time step and we will proceed for the computational at next time step, if not we will reset this iteration counter as a m=m+1 and then go back to step 1.

So this is a nutshell our implicit pressure correction method based on an implicit time integration scheme and a linearization procedure.

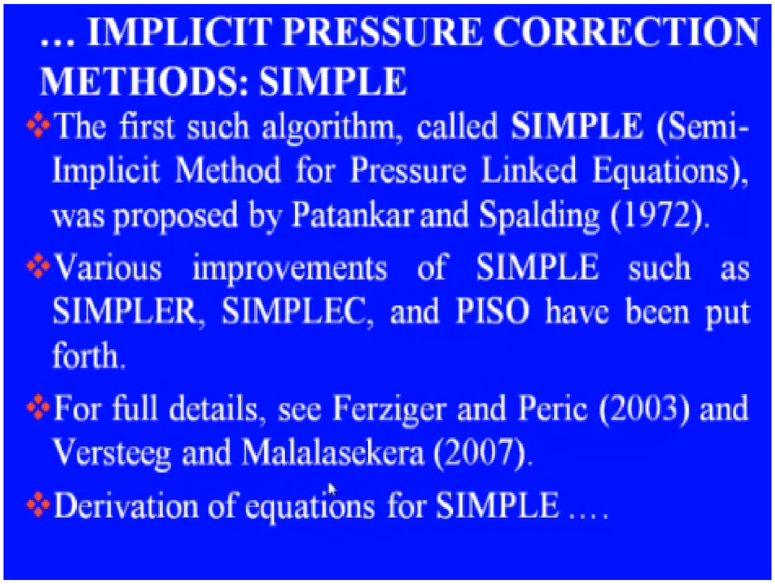
(Refer Slide Time: 29:28)



Now there are many variants of this implicit pressure correction method which have been proposed in the CFD literature. The some variants which suggests that we can use the values of the previous iteration level and we just need to add small correction to get the corrected velocity values that is to say that velocity field computed from linearized momentum equation and pressure pm-1 can be taken a provisional values to which small corrections must be added to obtain the final values for the current outer iteration.

That is Vi m could be obtained as Vi m* + a small correction Vi prime and similarly, pm=m-1 plus a small correction p prime. We will have a look at one such method which is called SIMPLE.

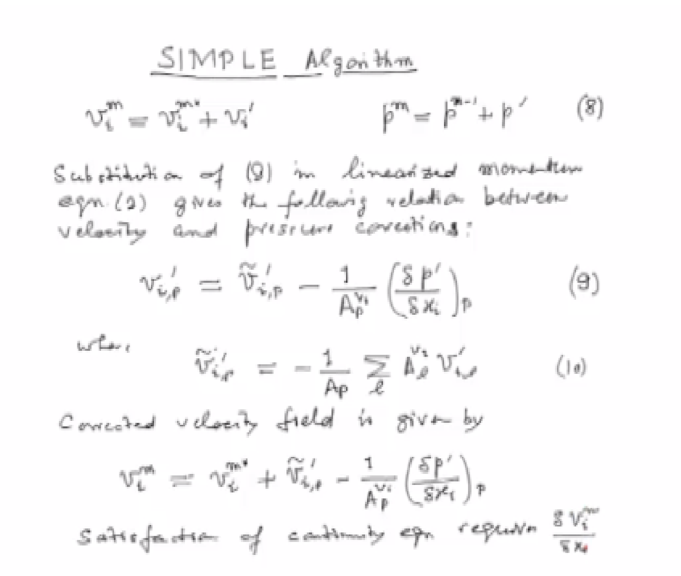
(Refer Slide Time: 30:22)



This was the first algorithm, its names was given as SIMPLE by Patankar and Spalding in 1972 and SIMPLE is essentially is an acronym that is Semi-Implicit Method For Pressure Linked Equations. So each first letter of each of these words that is involved in this different word SIMPLE. Okay. And there are various improvements of SIMPLE such as SIMPLER, SIMPLEC, and PISO; they have been put forth in literature.

For the details, please see the books by Ferziger and Peric and Versteeg and Malalasekera. In this lecture we have got time only to discuss simple and in next lecture we will take up SIMPLEC and SIMPLER schemes. So now let us have a look at equations which are involved in the SIMPLE algorithm. So let us get back to a board.

(Refer Slide Time: 31:21)

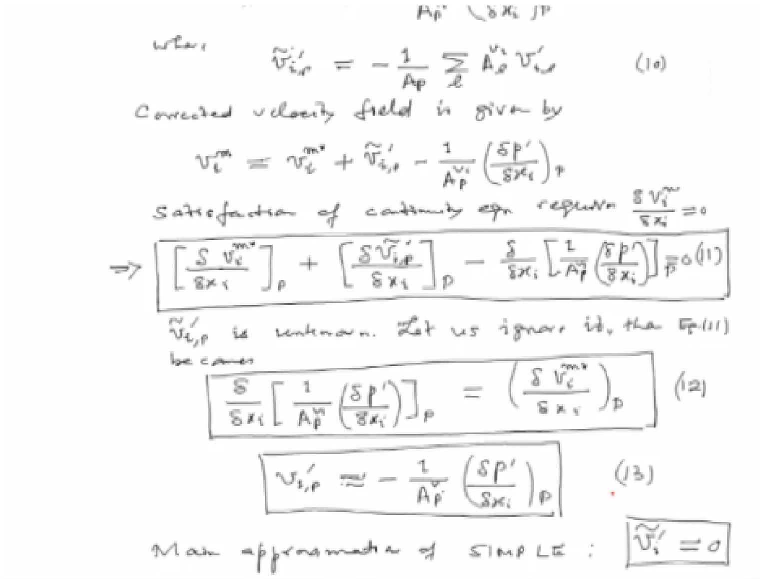


This algorithm is based on what we have discussed earlier for our implicit pressure correction method. So many of the pressure corrections which are derived they would be used as such, the kirks is that we want to represent our corrected velocity Vi m as Vi m* plus small correction added Vi prime and similarly for pressure at pm is obtained by pm-1+ small correction p prime.

So let us continue the same numbering scheme let us call it as equation 8. Now if we substitute these equations, so substitution of 8 in linearized momentum equation which have obtained for Vi m* gives the following a relation between velocity and pressure corrections. So this relation is Vi p prime this is equal to Vi,p ~ prime-1 over ApVi delta p prime over delta xi.p this is got a form very similar to the relation which we have written for Vi m* so here again this Vi p prime ~=-1/Ap sigma l AlVi Vi,l prime. Okay, now let us call this as 10.

Now, what would be our-- if we substitute 9 and 8 our corrected velocity field is given by Vi m=Vi m*+Vi p ~ -1 over ApVi delta p prime over delta xi at p. Now this must satisfy your continuity equation. So satisfaction of continuity equation requires delta Vi m over delta xi=0.

(Refer Slide Time: 35:53)



And this leads to delta over delta xi of Vi m*+delta Vi, p prime ~ over delta xi p-delta over delta xi 1 over ApVi delta p prime over delta xi at p.

So if you look carefully, this equation is a discrete Poisson equation for the pressure correction p prime. The first term on the left hand side that is known as because Vi m* that would have been obtained from the solution of linearized momentum equation. But this Vi p ~ prime that is unknown, we still do not know what our velocity corrections are. So what do we do then? One of the options which Patankar says that this Vi p ~ prime is unknown.

So as your first approximation, so let us ignore it. So if ignore it then what happens? Then, equation 11 becomes delta over delta xi of 1 over ApVi, delta p prime over delta xi at p=delta Vi m* delta xi at p. So this is Poisson equation for pressure correction. And similarly, if we implicit this Vi ~ to 0 so equation 9 becomes our velocity correction equation becomes at Vi p prime-1 over ApVi delta p prime over delta xi p. So let us call this equation is 13.

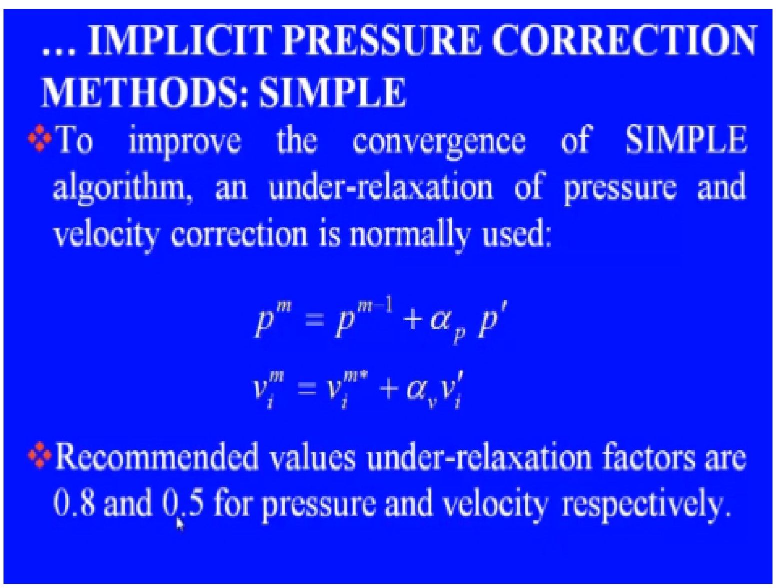
So now we have got an algorithm wherein what we need to do is first let us obtain the intermediate velocity field Vi m prime, use that to obtain a pressure Poisson equation-- solve this pressure Poisson equation to get p prime; once we have this pressure correction field we get this velocity corrections, add it to the velocity value we get the velocity field; add it to the previous pressure value we will get updated pressure value.

So now here we have got one problem here let us note down that the central approximation which we have made or the main approximation, approximation of SIMPLE algorithm that is we have set this Vi prime ~ to 0. Okay. Now let us have a look, let us formularize the steps in an algorithm form. So the first step is very similar to what we have seen earlier that use velocity in pressure at previous iteration level or previous time step to solve for Vi m prime.

Compute its derivatives and solve the Poisson equation of pressure correction p prime. Then compute the velocity correction Vi prime. Updated velocity field Vi m and pressure pm then check for the convergences whether the fields Vi m and pressure field pm satisfy momentum equation, if Yes terminate the iterations for the current time level to go next time level or else increment the iteration current counter and go to step 1 and repeat the SIMPLE algorithm iterations.

Now we have ignored one term which was not known and that leads to a slow convergences of SIMPLE iterations.

(Refer Slide Time: 41:12)



So what Patankar noted is to improve the convergences we must use under-relaxation for pressure and velocity correction. We will not add the total value of p prime and V prime which we calculate we would instead weight them by small factor and that factor would be < 1, so pm would be given as pm-1+alpha p times p prime where alpha p is < 1, similarly, Vi m is Vi m is m*+alpha Vi prime.

So in theory we can have different under-relaxation factors alpha V for each of the velocity components, but normally we take the same values. The values which are recommended by Patankar they are 0.8 and 0.5, 0.8 for the pressure correction and 0.5 for the velocity correction. (()) (42:03) the recommendation is we let choose our velocity correction or velocity under-relaxation factor alpha p. And alpha p is chosen to be 1-alpha v.

So these are two recommended set of values which help in improving the convergences of SIMPLE algorithm. For further details, you can look at these books by Anderson, Chung. Ferziger and Versteeg and Malalasekar. So in this lecture we will stop here. In the next lecture, we will have a look at the improved version of SIMPLE algorithm. And we will also take up fractional step methods.