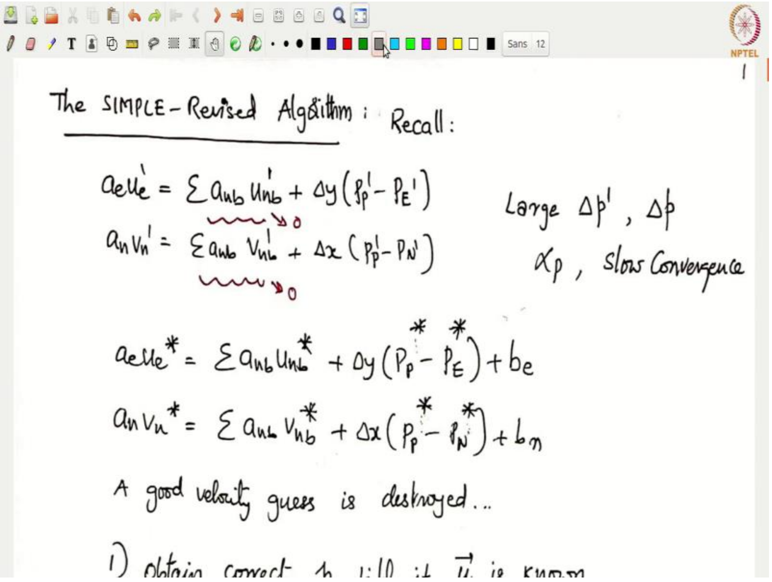
Computational Fluid Dynamics Using Finite Volume Method
Prof. Kameswararao Anupindi
Department of Mechanical Engineering
Indian Institute of Technology, Madras

Lecture – 39
Finite Volume Method for Fluid Flow Calculations: SIMPLE – Revised and SIMPLE – Corrected Algorithm

(Refer Slide Time: 00:14)



Let us get started. So, welcome to another lecture as part of our ME 6151 computational heat and fluid flow course. So, in the last lecture, we solved sample problems a couple of three exercise problems from Patankar's book; we have also looked at the corresponding programs, ran them and understood them. So, in today's lecture, we are going to look at couple of variants of the simple method and then we will kind of see how do we extend the staggered grid approach to curvilinear meshes or to unstructured meshes.

So, or the essentially the difficulties associated by in extending this staggered grid approach to unstructured meshes is what we will look at towards the end of the lecture today, alright. So, we kind of look at a variants of simple algorithm today, that is basically one of them is called simple R that is simple revised algorithm, ok.

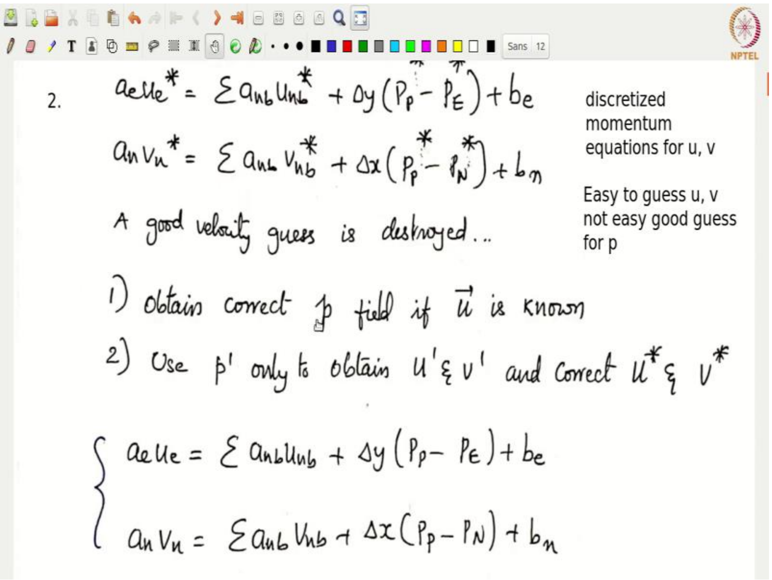
So, essentially this kind of tries to address some of the shortcomings of the original simple algorithm. So, one of the approximations we made in the simple algorithm was; when you write the velocity correction equation that is $a_e u'_e = \sum \mathrm{a_{nb}} \mathrm{u}'_{nb} + \Delta y(P'_P - P'_E)$. We said the contribution of the neighbouring cells for the velocity corrections would be taken as 0.

Essentially this is to make, this is to avoid the global dependence of the of the pressure corrections, right. So, we said these two are 0's; $\sum \mathrm{a_{nb}} \mathrm{u}'_{nb}$ and $\sum \mathrm{a_{nb}} \mathrm{v}'_{nb}$ are 0 and then kind of derived the velocity corrections in terms of only pressure corrections.

And as a result we got a pressure correction equation; kind of we got a pressure correction equation from the continuity equation, right. But this was ok; although the downside is that, the pressure corrections alone are responsible for correcting the velocities.

As a result the pressure corrections were large. So, we ended up with large changes in pressure correction, which would eventually result in large changes in pressure; $P = P^* + P'$ is what we will use to correct the pressures. As a result we have to kind of under relax the pressure right; we had to under relax the pressure and because of the under relaxation, we will end up with slow convergence, right. So, this is one of the downsides of the original simple algorithm.

(Refer Slide Time: 03:01)



Another thing is basically; if you look at the momentum equation, so what we have is we have $a_e u_e^*$. So, so that this is basically the first issue, this is basically the first issue. The second one is basically; if you look at the momentum equation, we have $a_e u_e^*$ equals $\sum \mathrm{a_{nb}} \mathrm{u}_{nb}^*$plus the pressure gradient that is $\Delta y(P_P * - P_E^*)$ plus $\mathrm{b_e}$, right.

Similarly, we have the y momentum equation is $a_n v_n^*$ equals $\sum \mathrm{a_{nb}} \mathrm{v}_{nb}^*$ plus $\Delta x(P_P^* - P_N^*)$ plus $\mathrm{b_n}$, right. So, this is basically your momentum or discretized momentum equations right for x and y or for u and v, right. One thing what we see here is that, even if you have a you basically have to come up with a pressure guess and a guess for the velocities, right.

You have to come up with star values for u, v and pressure. Now, what we see is that, although we have a good velocity guess; this velocity guess will be has to be accompanied by a good guess for the pressure, right.

If you do not have a good guess for pressure; then this pressure guess field who is going to kind of destroy the velocity guess that we have, because eventually you will get a new u^* values at every cell. Although there will be some contribution coming from the velocities, but will be kind of overridden or destroyed with the pressure guess values.

As a result we have to. So, even if you have a good velocity guess; unless you it is accompanied with a good pressure guess, it is not going to survive, right. So, then it is not a good idea; because usually it is easy for any problem for us to kind of easy to guess the velocity field, ok. Essentially the guess fields for velocities can be intuitively guessed; whereas it is not easy to come up with a good guess for pressure field.

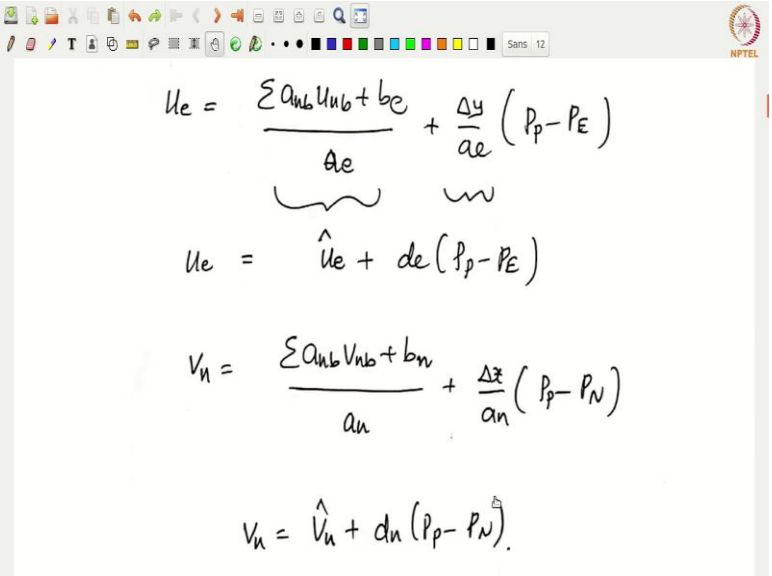
So, as a result this is not an ideal scenario ok; so that means a good velocity guess is destroyed, if even if you do not have a good guess for pressure, ok. So, as a result what we would like is, we would like to essentially have an equation which will kind of recover the pressure field; that means which is difficult to guess from a good velocity guess, ok. That means, we are looking at obtaining a pressure field directly from the velocity field somehow, ok.

So that, that pressure field can be used and then on together with the velocity field and you can obtain the, you can solve the momentum equations with that new pressure field which is kind of good, alright. So, that means, the idea here in simpler or simple are suggests that, we obtain essentially correct velocity field using let us say velocity field ok; obtain the correct pressure field using velocity field, using a known velocity field.

And then limit the use of this P' that we have the pressure corrections only to correct the velocities ok; because you already have an equation for pressure, do not use the pressure corrections to correct the pressure again, ok.

As a result the P' equation after it is converged will be used only to correct the velocities that is u' and v', and use another equation which is for the pressure to solve for the evolution of pressure, ok. So, essentially that is the basic idea, we will look at this in detail.

(Refer Slide Time: 06:39)



So, if we go back to the original momentum equation. So, of course, this is again your x and y momentum equations that is $a_e u_e = \sum \mathrm{a_{nb}u}_{nb} + \Delta y(P_P - P_E) + \mathrm{b_e}$ on the staggered mesh. Similarly, on the north face we have $a_n v_n = \sum \mathrm{a_{nb}v}_{nb} + \Delta x(P_P - P_N) + \mathrm{b_n}$, ok. So, this is your on the east face and the north face for the staggered mesh, the moment discrete momentum equations.

Of course, these have to be kind of come up with you, basically have to write stars for the pressures and stars of the velocities which I have not written here; but it is understood that, given if you have a velocity guess and a pressure guess, you can solve for the momentum equations. That is understood; now we kind of try to come up with an equation for pressure from the velocities, ok.

So, as a result I want to kind of divide this first equation with a_e everywhere. So, basically divide, send this a_e to the right hand side. So, what we get is, u_e equals $\sum \mathrm{a_{nb}u}_{nb}$ plus b_e divided by a_e ok; I am dividing with a e throughout plus we have $\Delta y/a_e$ times $(P_P - P_E)$.

Now, here we call again essentially this term; this is $\sum a_{nb} u_{nb}$ plus b_e divided by b_e as some $\hat{u}_e$, ok. So, this is some $\hat{u}_e$ plus we already know, we were calling this $\Delta y / a_e$ as d_e right, some other coefficient.

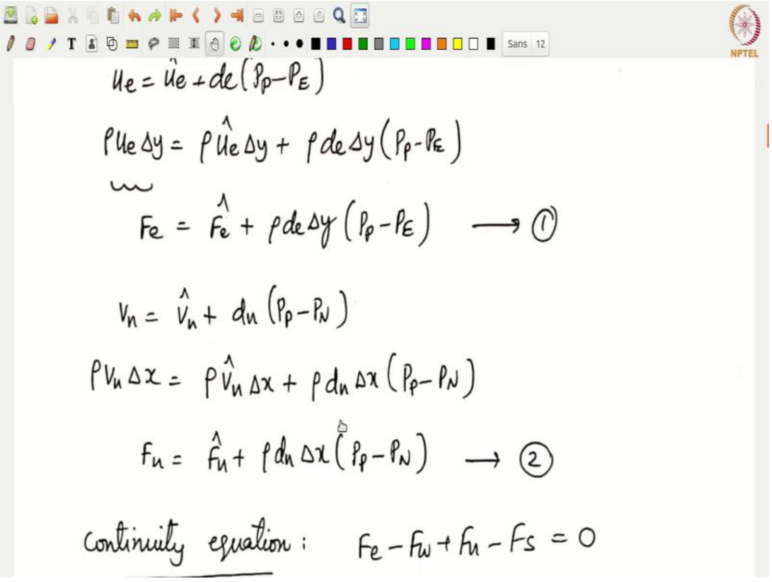
So, we can write essentially your momentum equation, rewrite the momentum equation for velocity on the east face as u_e equals $\hat{u}_e$ plus d_e times $P_P - P_E$, ok. So, this is your now the new momentum equation, right. So, this is your new momentum equation, and we have rewritten as u_e equals $\hat{u}_e$ plus d_e times $P_P - P_E$, alright. Then let us look at the other equation that is the y momentum equation.

This also we can send a n to the right hand side, basically divide the right hand side with a n; then we can write v_n as v_n equals some v_n plus d_n times $P_P - P_N$, right. So, that is basically v_n equals $\sum a_{nb} v_{nb} + b_n$by a_n plus Δx; essentially $\Delta x / a_n$ times $P_P - P_N$ plus b_n, that is your new equation here.

So, this is your new, essentially this is basically the same y momentum equation; but written in a different way right, because we have absorbed all the neighbouring coefficients and the source terms into this hat equation here, alright. And of course, we have divided with the a p coefficient or a east or a north coefficient in this context, alright.

Then what we can do is now that, we have the velocities in terms of the; we have the face velocities in terms of the pressure differences and the hat velocities right, both for u and v. What we can do is, we can multiply these two equations, essentially each of these equations with the density times area to get an expression for flow rates, right.

(Refer Slide Time: 09:55)



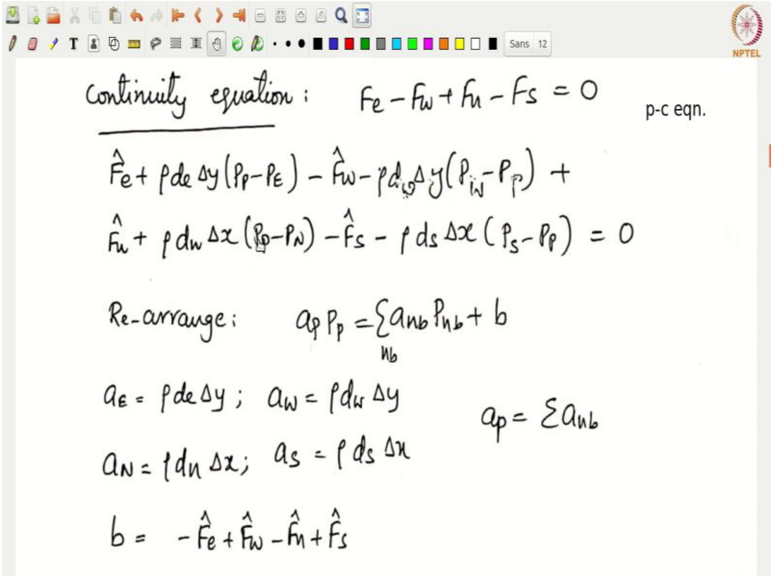
So, I multiply, essentially I start off with u_e equals $\hat{u}_e$ plus d times $P_P - P_E$ that is this equation, right. We start off with this and then we multiply entire equation with $\rho\Delta y$, right.

So, what we have is $\rho u_e \Delta y$ equals $\rho\hat{u}_e\Delta y$plus ρd_e times Δy times $P_P - P_E$. Of course, we know that this quantity rho u_e Δy is nothing, but your mass flow rate across east face equals rho $\hat{u}_e$ Δy; we would like to represent that as, represent it as represent it as some F_e ok, which is basically mass flow rate defined based on $\hat{u}_e$ as the velocity plus we have $\rho d_e \Delta y$ times $P_P - P_E$, ok.

So, let us call this equation 1. Similarly, if you if you start with the y moment, rearranged y momentum equation; then we have v_n equals $\hat{v}_n$ plus d_n times $P_P - P_N$. Again if you multiply on both sides by the density times the corresponding area, the normal area for this velocity; then what we get is v_n equals $\rho\Delta x\hat{v}_n$ plus $\rho d_n \Delta x$ times $P_P - P_N$, ok.

Again we can call this as the mass flow rate through the north face equals mass flow rate through the north face based on hat velocities on the face, that is $\hat{v}_n$; so that means F_n equals F_n hat plus $\rho d_n \Delta x$ times $P_P - P_N$, ok.

(Refer Slide Time: 11:26)



So, similarly we can write equations for the west face and the south face; then essentially we can invoke the continuity equation. The continuity equation is the mass flow rates, all the mass flow rates through the faces should sum to 0. So, that is $F_e - F_w + F_n - F_s = 0$. Then you plug in the definitions for flow rates in terms of hat velocities ok; that means F e equals $\widehat{F_e} + \rho d_e \Delta y(P_P - P_E)$. So, substitute that here.

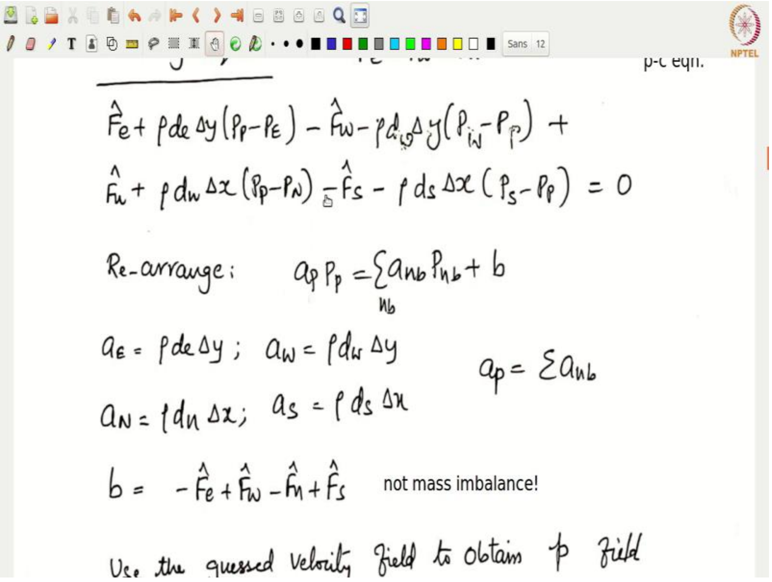
Similarly, for F west we would, we would basically get $\widehat{F_w}$ plus $\rho d_w \Delta y$; but there is a minus here, so both the terms become minus times $P_W - P_P$ plus essentially $\widehat{F_n} + \rho d_n \Delta x(P_P - P_N)$ minus $\widehat{F_s} - \rho d_s \Delta x(P_S - P_P)$ equals 0, ok.

So, this basically looks very similar to what we have done before for the pressure correction equation, right. I mean instead of writing $u = u^* + u'$; we basically came up with some equation that is relating velocities to pressures right, using hat velocities and then we invoke the continuity equation and somehow got an equation for pressure.

Now, this equation for pressure again looks very similar except that there are no primes here; because this is the pressure itself to what we have done before in the simple algorithm, ok. Now, only thing is that now if I rearrange this; I can of course write this as in the standard form as $a_P P_P = \sum a_{\text{nb}} \text{P}_{nb} + b$ ok, where b of course is all these hat flow rates taken to the right hand side.

And $a_E = \rho d_e \Delta y$; $a_W = \rho d_w \Delta y$; $a_N = \rho d_n \Delta x$; $a_S = \rho d_s \Delta x$ and a_P would be of course again sum of all these coefficients, right. Basically all the neighbouring coefficients will go to the right hand side and the a_P will remain on the left hand side. So, this is basically a_P would be equal to $\sum a_{nb}$.

(Refer Slide Time: 13:41)



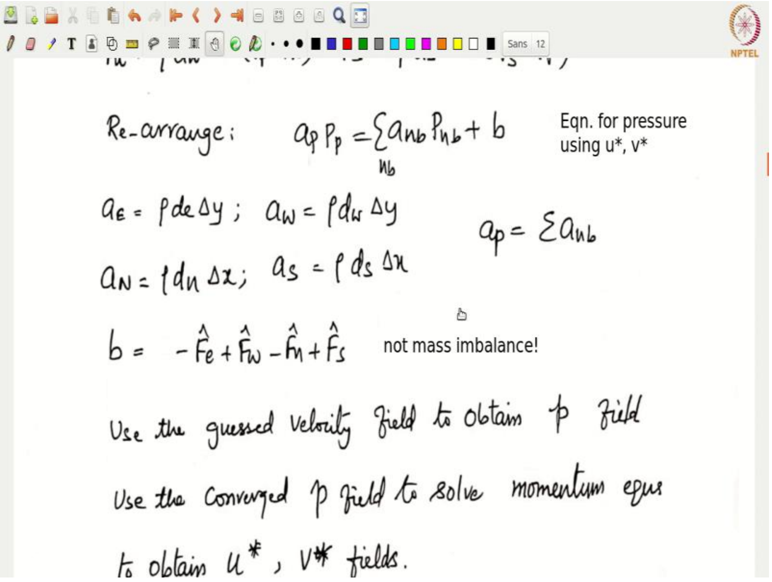
Then your b term of course is now basically minus $\widehat{F_e}$ when it is sent to the right hand side plus $\widehat{F_w}$ minus $\widehat{F_n}$ plus $\widehat{F_s}$, ok. So, this is your b term. Remember in the pressure correction equation, the b term was minus F_e^* plus F_w^* minus F_n^* plus F_s^*, right.

These were all based on the star velocities, where we said that the b term in the pressure correction equation is basically the mass imbalance right or the mass imbalance or the amount of mass, or the amount by which the momentum equation or the velocities obtained by the momentum equations do not satisfy the continuity equation right; that is the amount that b denotes in the pressure correction equation b term.

Whereas here this is although it looks like some mass flow rate; this is basically not the mass imbalance, right. Because, because $\hat{u}_e$ here is basically not the velocity of the face, rather it is only some component of it right; because if you go back, your $\hat{u}_e$ is defined based on $\sum a_{nb} u_{nb}$ plus b_e by a_e, right.

So, this is not exactly the velocity flow rate; because you have this pressure term as well which is not incorporated to this thing. So, that is why this hat velocity is kind of used for working of the algorithm; but it is does not represent a physical velocity through the face right, because the pressure gradient is not involved in this, alright.

(Refer Slide Time: 15:22)



So, essentially we realize that the flow rates based on the hats, hat velocities is not the mass imbalance like or similar to the you know unlike the flow rates based on the star velocities which was representing a mass imbalance, alright. Now, essentially what we did is, in the process we came up with an algorithm.

Basically we obtained a an equation for pressure itself right with a pressure itself, using essentially u and v as the values right; u and v basically meaning u^* and v^* right, because these u^*, v^* will go into u hat v hat and u hat v hat will eventually go into b.

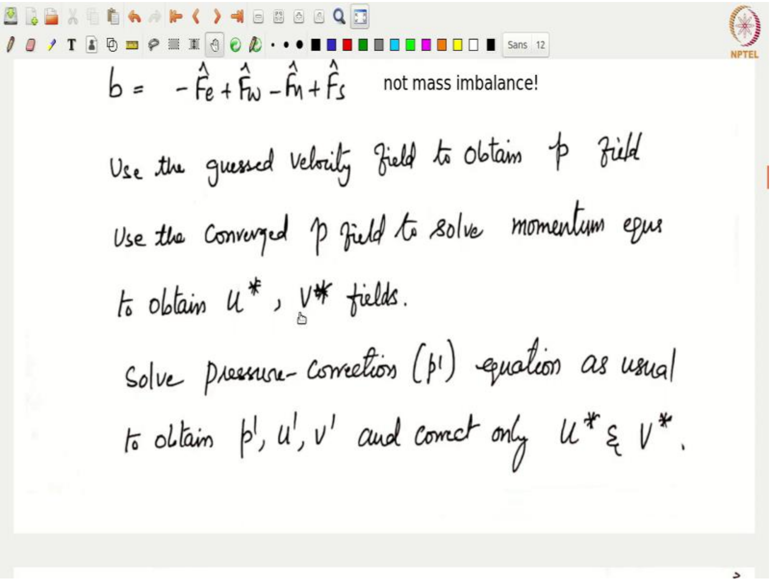
So, depending on those b values, you will get a pressure field and this pressure field if you converge this thing; you are going to get a pressure field that is coming from the guessed velocity fields, right. So, what we did is basically, because it is difficult to come up with a good pressure guess; but it is easy to come up with a good velocity guess, we use the velocity guess and solve for the pressure using the pressure equation here and obtain the pressure guess.

Now, this pressure guess that is coming out of this converge solution and this velocity guess can be used in the momentum equations directly, so that the velocity guess this guess is not going to be destroyed and we can somewhat hope to converge faster, ok. Of course, what we have done is in the process; we introduced another equation which requires solution of Gauss Seidel again, right.

So, already we have to solve for two Gauss Seidel's; two system of linear equations that is for x momentum and y momentum, then we had to solve for one system of linear equations for pressure correction. Then in the simple revised algorithm, we are introducing another system of linear equations that is $a_P P_P = \sum a_{\mathrm{nb}} \mathrm{P}_{nb} + b$ this is one equation for pressure. So, essentially we added one more system to be solved in the, into the algorithm.

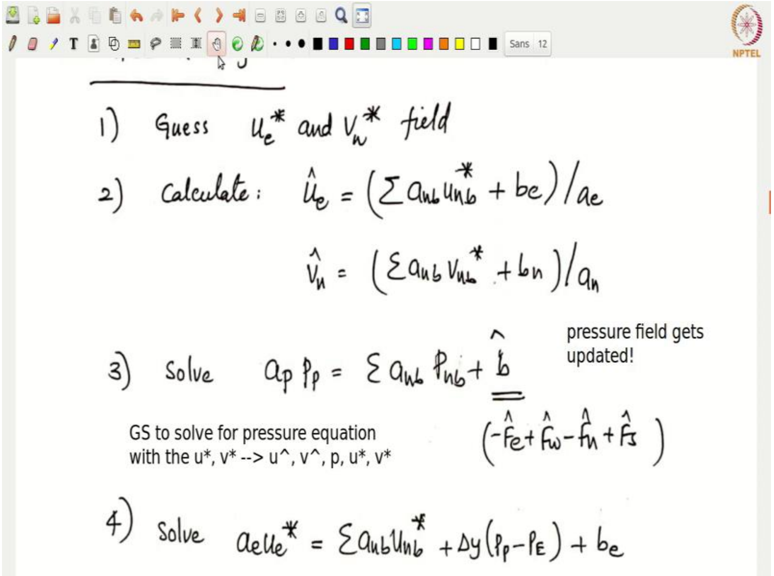
With a hope that this although we are doing work here, essentially you get paid off when you solve for the momentum equations; because the good velocity guesses that you obtain would be eventually help you converge quickly, ok. So, that is the idea; that means now use the guessed velocity field to obtain pressure field ok, that is what we just discussed. Now, use the converged P to solve for the momentum equations to obtain u star v star, ok.

(Refer Slide Time: 17:45)



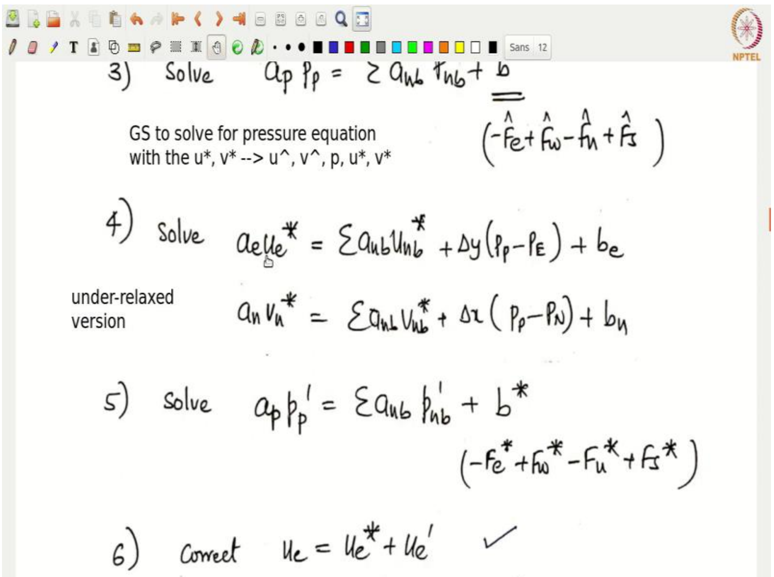
So, essentially now once you solve for the momentum equations; then you obtain $u^* v^*$ fields, then use these to calculate the b term and solve for the pressure correction, right. And use this pressure correction only to obtain or only to correct u^* and v^*, not pressure; because pressure we will use whatever you get from here, alright.

(Refer Slide Time: 18:09)



So, let us see the complete algorithm for simple revised. So, we similar to simple we start off with the guess values for u_e^* and v_n^* fields; then we calculate from once you have this guess values, you calculate the $\widehat{u_e}$ velocities that is $\sum a_{nb} u_{nb}^*$ plus u_e plus b_e upon a_e and $\widehat{v_e}$ equals $\sum a_{nb} v_{nb}^*$ plus b_n upon a_n, ok.

(Refer Slide Time: 18:38)



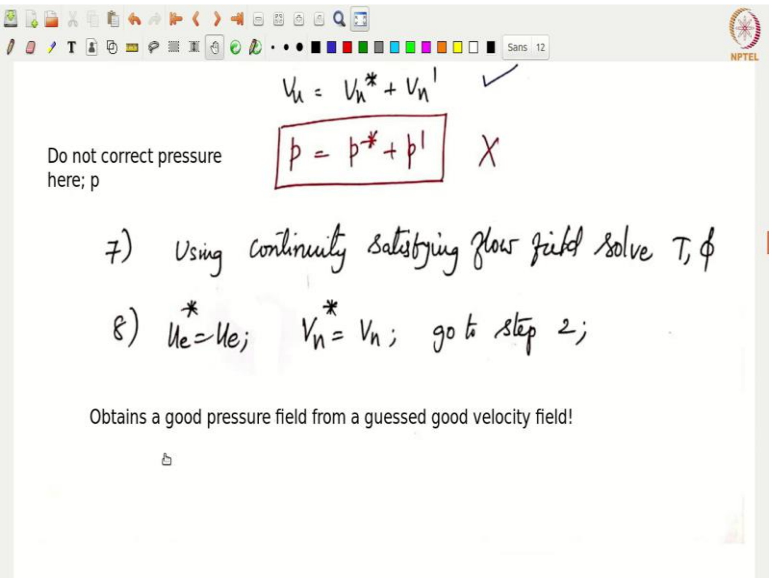
So, we just basically guess the velocities and obtain the hat velocities; then we solve for the pressure equation, ok. So, the pressure equation is $a_P P_P = \sum a_{nb} P_{nb} + \hat{b}$; this $\hat{b}$ contains these hat velocities, which is basically minus $\widehat{F_e}$ plus $\widehat{F_w}$ minus $\widehat{F_n}$ plus $\widehat{F_s}$. So, essentially you need one Gauss Seidel or solution of linear system here to solve for pressure equation with the guess values for $u^* v^*$ which are going in as basically $\hat{u}$ and $\hat{v}$, right ok.

Then we obtained a pressure. So, this pressure that you get here, we can we can call it of course P^*; but we will just leave it as P. So, this P and these two converged velocity guesses right, that is whatever you get out of this equation is the pressure and this converged pressure and the guess to u^* v^* are now kind of in a good form; because this pressure now is consistent with the with the guessed velocity field, ok.

So, that means then we can solve for the momentum equation. So, as usual this is basically; this was the fourth step here is basically was the first step for simple right, because we started off with once you have the guessed velocities and guess pressures you, you went to directly to solving the momentum equations.

But now what you do is, you converge the pressure correction; then use those pressures and the guessed velocities and solve for the momentum equations. Of course, I have written the equations here; but you have to write them in the under relaxed version of these for the momentum equations. That is because you still have to use under relaxation for momentum equations; because you know linearity in the convection term or the nonlinearity that may come up in the source terms or something like that, alright.

(Refer Slide Time: 20:40)



Then once you have the converged u^*v^*, you calculate for pressure correction. Now, why do we need pressure correction? Because we still need pressure correction to collect correct our velocities right; we do not need pressure correction to correct pressure itself, because we already have a pressure field.

So, we still solve for pressure correction, again you have another solve for another Gauss Seidel here; solve for pressure correction till convergence, obtain the converged P prime field, ok. Which is basically $a_P P_P' = \sum \mathrm{a_{nb}} \mathrm{P}_{nb}' + b$, where the b; here I represent using b star to kind of distinguish it from the $\hat{b}$ that we have used in the pressure equation. So, b^* is basically minus F_e^* plus F_w^* minus F_n^* plus F_s^*, ok.

So, once you have the pressure correction, you correct your velocities from the pressure correction. So, u_e' is again expressed in terms of p primes right, same as before right; we had u_e equals u_e^* plus u_e'; v_n equals v_n^* plus v_n', but do not correct the pressure.

So, do not correct pressure here; because we already know, we can already obtain pressure from the velocities, ok. So, once you have this u_e ; v_n, we can directly obtain the pressure.

The idea is basically you do not have to correct for pressure; so as a result the pressure would not come with large changes in the pressure, as a result we can avoid the slow convergence of the pressure equation or the pressure correction equation or in terms of destroying a good velocity guess, ok. So, these are the things that are addressed.

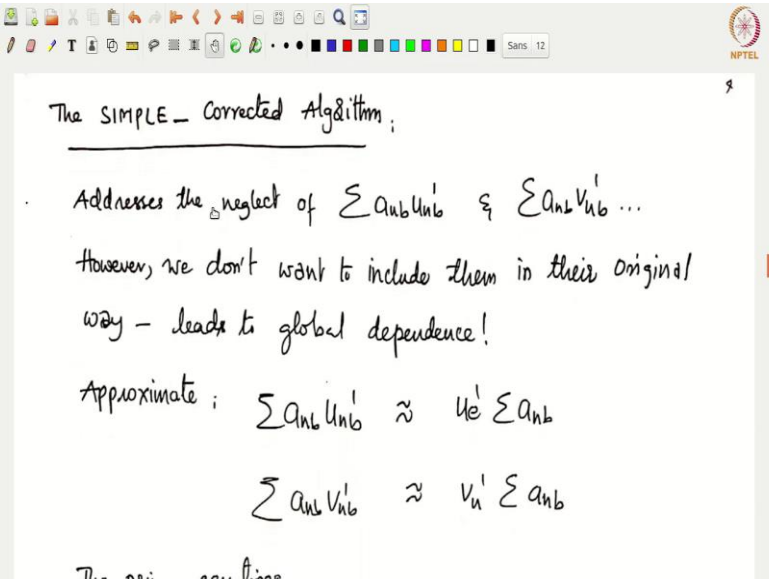
Now, using the this continuity satisfying field, like after you correct the velocities, this satisfies continuity; then of course, you can solve for any other scalar, such as temperature, species transport or any other ϕ in the flow field. Now, then we essentially come and update your star values with the updated values, the corrected values; that means u_e^* equals u_e, v_n^* equals v_n and if it is not converged, go back to step 2.

Now, now again you got a new velocities that are just corrected as the guesses. So, with these guess velocities and the pressure, you go back to step 2 ok. And what you do? You use the updated velocities and calculate the hat velocities; then you solve for pressure, ok. So, this is where basically pressure field gets updated, fine ok. So, that means we do not have to under relax for pressure; but we still have to relax for the momentum equations to account for the non-linearity in the problem, ok.

Of course, we understand that in this process we have introduced one extra equation which requires Gauss Seidel as well, ok. So, simple revised kind of essentially fixes this problem that, of the slow convergence, of the pressure equation and also fixes the problem that there is essentially a it kind of obtains a kind of good pressure field from a guessed or good velocity field. So, as a result we are not trying to find the good pressure field after destroying the velocity field, ok.

So, that is kind of the advantage of simple revised; of course the computational effort goes up because of the Gauss Seidel ok. A Gauss Seidel loop that has come up because of solid in the pressure equation, alright. Let us look at another variant of simple that is known as simple corrected or simple C.

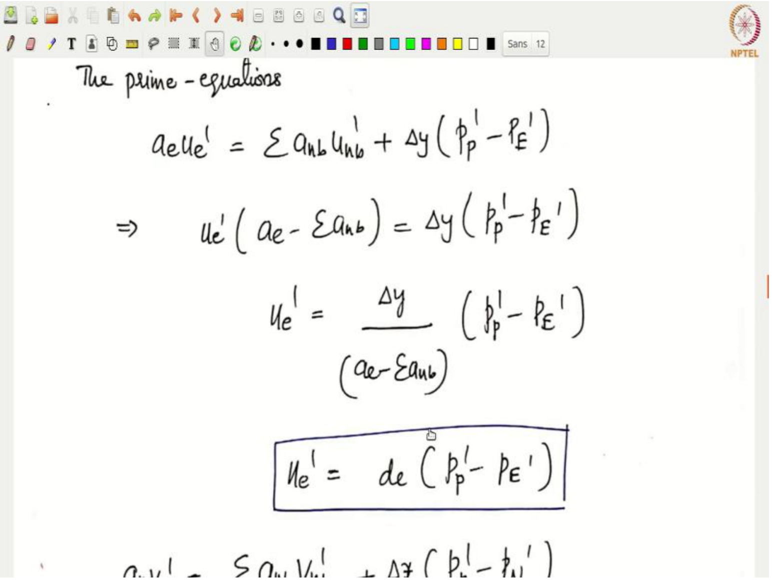
(Refer Slide Time: 24:31)



So, simple corrected kind of addresses the neglect of the neighbouring contributions that is; remember you $\sum a_{nb}u'_{nb}$ and $\sum a_{nb}v'_{nb}$ were neglected in the u'_e and v'_n equations, ok. But of course, we realize that we do not want to include them; because if you include them, you end up with this unmanageably long equation, which leads to global dependence of pressure, right. Again that is not what we want to do; because we want to keep things to the near neighbouring cells.

So, as a result we cannot come, but at the same time we do not want to neglect them completely; because if you neglect them all this completely, then the entire burden of correcting the velocities falls on the pressure.

(Refer Slide Time: 25:25)



So, as a result a simple character proposes that, you approximate $\sum a_{nb} u'_{nb}$ with $\sum a_{nb} u'_e$. So, essentially it says, the contribution of the neighbouring cells times the neighbouring corrections can be taken as the neighbouring, the primary cell correction itself multiplied by the coefficients of the neighbouring cells.

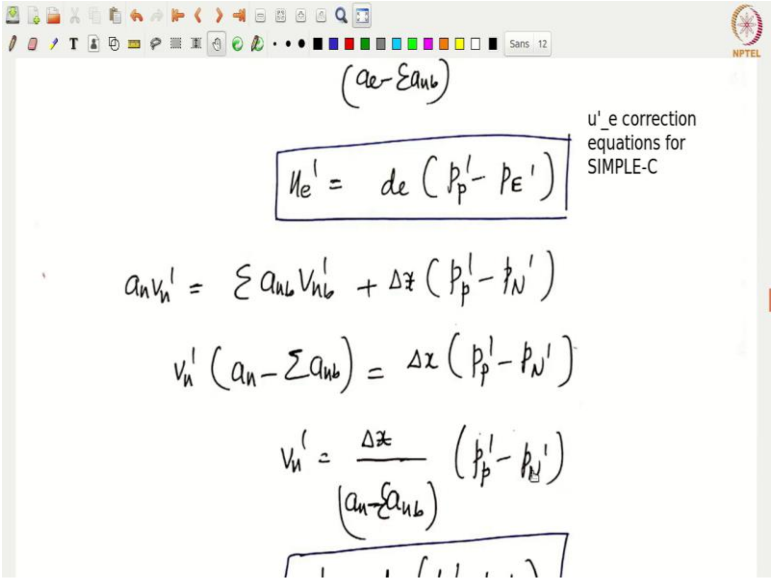
So, that means you see now this will simplify things right; because you do not have to, because the moment you write u'_{nb} as u'_e, you do not have to solve for a system. And you do not have that recursion of including for the neighbours and its neighbours will not come into picture, ok. But the as an approximation, the coefficients are accounted for, ok.

So, that means, $\sum a_{nb} u'_{nb}$ it says approximate as u'_e time $\sum a_{nb}$ and $\sum a_{nb} v'_{nb}$, you approximate it as $\sum a_{nb}$ times v'_n, ok. So, these are the approximations that are proposed by simple corrected algorithm; that means our prime equations for velocities now get modified.

So, instead of this was, this was earlier taken as a 0 right, this was taken as 0; but now we do not take it as 0, rather you write this as $\sum a_{nb}\, u'_e$. That means, if you take it to the left hand side; we can write u'_e times $a_e - \sum a_{nb}$ equals $\Delta y(P'_P - P'_E)$.

So, your u'_e is now Δy by instead of simply a_e, you have a_e minus $\sum a_{nb}$, ok. So, this is basically u'_e equals some $d_e(P'_P - P'_E)$, where d_e equals Δy by $a_e - \sum a_{nb}$, ok

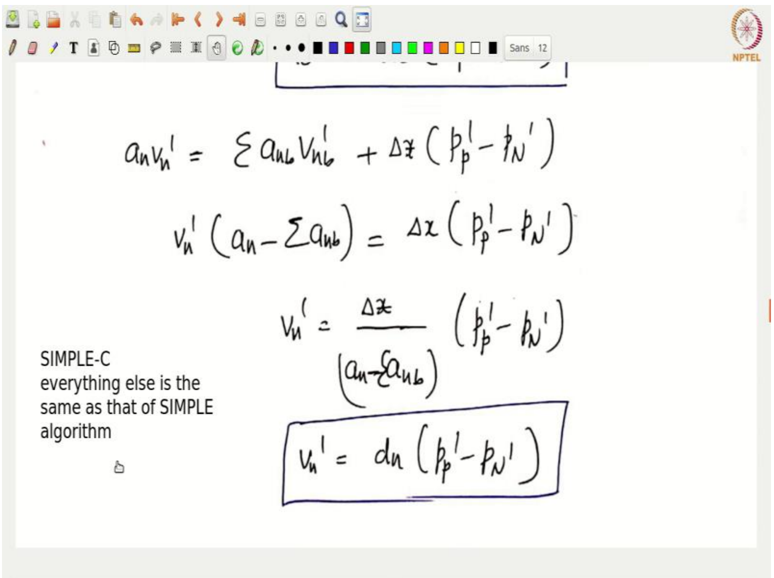
(Refer Slide Time: 27:07)



So, this is the new or the velocity correction right; this is velocity u_e' in terms of the pressure corrections, right. So, this is basically the velocity correction equation for simple corrected ok, where d_e is not the same as before; it has now the neighbouring contributions as well.

Of course, similarly we can write it for the north face $a_n v_n'$ equals $\sum \mathrm{a_{nb}} \mathrm{v}_{nb}'$ plus $\Delta x(P_P' - P_N')$. Then again instead of neglecting this completely, we say $\sum \mathrm{a_{nb}} \mathrm{v}_{nb}'$ equals v_n' times $\sum \mathrm{a_{nb}}$. So, if you take it to the left hand side, you get v_n' times $a_n - \sum \mathrm{a_{nb}}$ equals $\Delta x(P_P' - P_N')$.

(Refer Slide Time: 28:13)



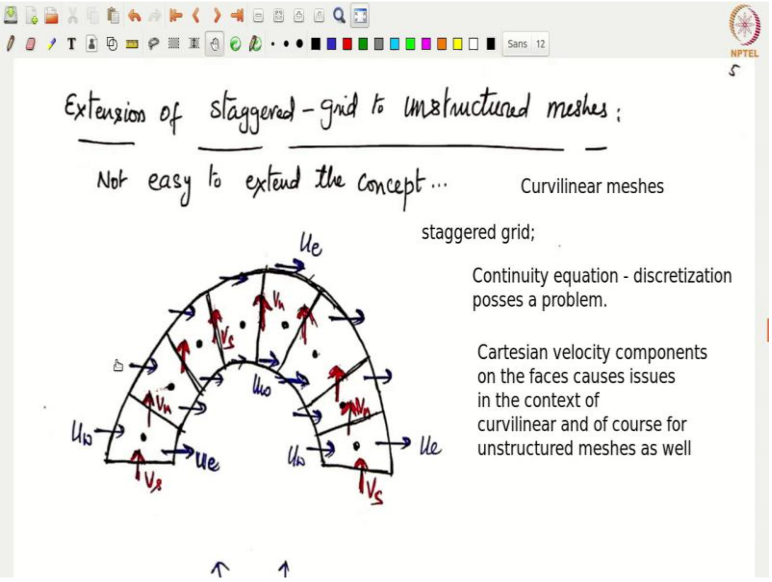
That basically gives you v_n' equals some $d_n(P_P' - P_E')$, where d_n equals $\Delta x/(a_n - \sum \mathrm{a_{nb}})$, ok. So, this is the correction equation for north face.

Now, that is the only change that we do for simplex or simple C; otherwise everything else is the same as that of a simple algorithm. That means, we do not have an equation for pressure; we only have an equation for pressure correction. And we have to still correct the pressure as well as velocities and so on ok. So, simple C is basically a modification on top of simple, where the neighbouring coefficient contribution is not neglected in the velocity correction equation.

So, this still simple C still suffers from the problem that, if you do not give a good pressure guess; then your velocity guess might be destroyed because of a poor pressure guess ok, which was addressed in the simple revised algorithm, ok. So, as a result that needs to be taken care here; of course simple C does not require an additional equation like what we had before in the simple revised, ok. So, as a result this is something quite different from the simple revised algorithm, ok.

Of course, there are many other variants of simple called simple M simple and I think simple best and a lot of other things are also available, which we are not going to detail into all of them and we are basically just looking at simple R and simple C algorithms, alright. Now, then let us look at how do we extend the concept of staggered mesh for, let us say if you have curvilinear meshes or if you have unstructured meshes.

(Refer Slide Time: 29:56)



Now, let us look at extension of this staggered grid approach to unstructured meshes or to start with we will look at curvilinear meshes; of course things are much more complicated to extend them to unstructured measures,. So, with that in the back of the mind, we apparently say that it is not very easy to extend this concept of concept of staggered grid; because if you have a Cartesian mesh, it is much easier to do things in a staggered way.

If you have a curvilinear mesh, we will see that we will run into problems and if we have unstructured mesh, the problems we run into are more so, ok. So, as a result let us kind of see some examples of a curvilinear mesh; here we look at let us say some kind of a curvilinear mesh like this, which is taken and the horizontal velocities here shown in blue kind of denote the u east and u west.

And the vertical velocity vectors here denote the v north and the v south for any of the cells and the cells are have a centroid that is denoted using this filled circle here, ok. So, of course, right away we see one problem; basically if you start storing the horizontal velocities and the vertical velocities on the faces, then we see that as we keep going, we see that all of a sudden we come up with a cell, where for this cell the vertical velocity is instead of being out of the face, it is now parallel to the face, right.

And similarly, the horizontal velocity instead of going out of the out of the face, it is now parallel to the face. Now, this is a problem, this problem will be difficult to address or essentially it shows up in solving the continuity equation; because now we do not have enough data to solve for continuity, right.

Because up till now if you take this example, you had some flow rate leaving and from some flow rate entering and some leaving here to the north face and entering to the south face; whereas here all of a sudden you do not have a representation for flow rate, because now v n is parallel to the parallel to the this face.

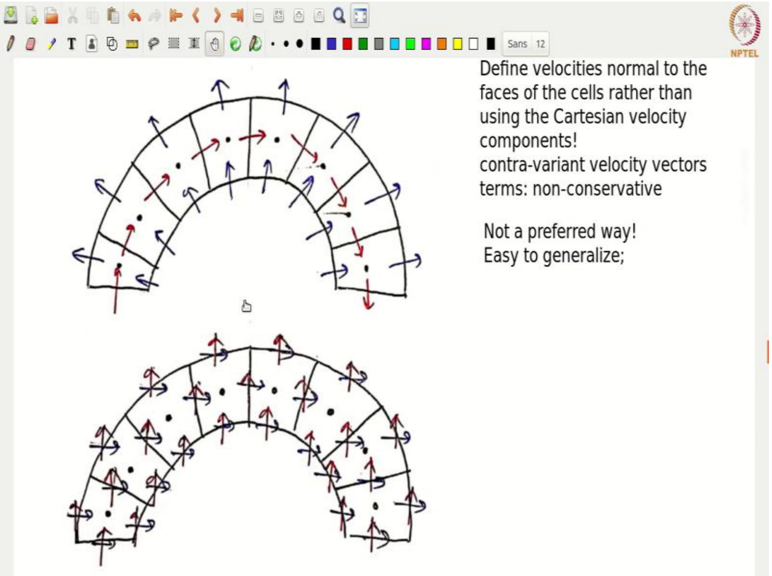
Now, of course, again this face happens to be, supposed to be some kind of an east face; now it happens to be north face right, because it kind of slowly developed into, because the mesh was turning, right. So, as a result you do not have enough data for the flow rates to be reconstructed from calculated from these velocities on the, this u e u east and the v north values.

As a result continuity or discretizing the continuity equation, discretization poses a problem and we cannot simply solve this problem unless some more data is specified which we do not have for the system to be solved, ok.

So, that is one problem that you see directly if you extend the concept of staggered mesh, that is basically storing the, storing the basically the Cartesian velocity components on the faces; just like the way we did for the Cartesian measures to essentially on the faces causes issues in the context of curvilinear and of course, for unstructured meshes as well, alright.

Then we can be clever, we say ok, we do not want to store; why should we store the Cartesian velocity components, I will come up with another way of doing it. Because we are dealing with curvilinear meshes, I will solve or I will store curvilinear velocity components, ok.

(Refer Slide Time: 33:32)



That means, we come up with something here, which is basically a curvilinear mesh. Now, we define the face velocities as not as the Cartesian components like u east and u west; rather as something that is normal to the particular face, ok. This is nice, because now every face has a normal and we define velocity vectors normal to that particular face. So, as a result red colours here denote your something like your east velocities, and the blue colour arrows here denotes something like your north, south velocities, ok.

And as a result you have a description of all the velocities on all the faces for all the cells and you do not have a problem, right. So, essentially this is kind of a work around; of course this can be done, essentially store or define velocities that is basically normal to the faces of the cells right, rather than using the Cartesian velocity components. So, this was tried in the literature.

So, this is tried in the literature and this kind of definition of velocities is known as contra-variant velocity vectors. This, so this was tried in the literature, so this exists. So, essentially curvilinear measures with contra-variant velocity vectors can be solved using staggered approach. But one problem is that, with this the terms that you get in the equation, such as the diffusion and things like that will become; they will not, they will become basically non conservative.

Because of the, because now the velocities are defined in a curvilinear fashion; as a result you end up with non-conservative form of equations or non-conservative discretized equations which are not very really good to work with for all sorts of meshes, ok. So, that is one issue and also it becomes non conservative; because now you end up with the control volumes that kind of overlap with each other because of these definitions and causes issues with non-conservation, ok.

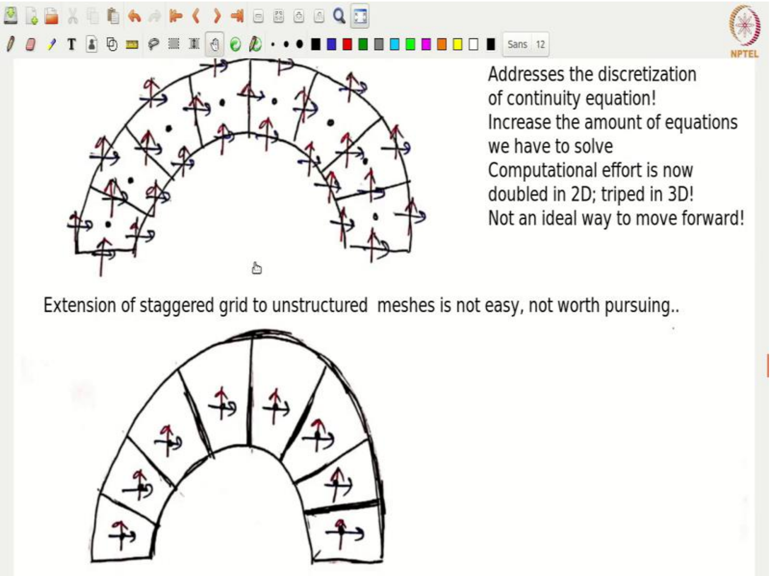
So, as a result this is also not a; so this kind of storing and solving is also not a preferred way, because we end up with losing the conservation for the diffusion terms and things like that. So, as a result this is not very much preferred in the context of unstructured meshes at least, ok. Now, another thing is basically the, it is not very easy to generalize or what you get is basically not very not very easy to implement, ok.

So, not very easy to generalize for different unstructured measures; as a result this is not preferred in the literature. Of course, then we can come up with another solution; another solution is basically, why not store both the components of velocities at all the faces?

So, earlier we said if you only store the Cartesian x velocities at a particular face and y velocities at the particular face, like what we have done in the Cartesian case; then we end up with the problem. Then let us now be clever and say that, we will store both velocities at both the faces, at all the faces; that means both store x and y, x and y and so on.

That means we end up, of course one thing you can clearly see is that, this will avoid the problem of mass conservation discretizing continuity equation; because although this becomes parallel, you have the other velocity which will give you in constructing the velocity vector at this particular face correctly, right. So, it is not a problem, now your face can be inclined at any angle; you have both the velocity vectors, then you can kind of obtain the correct mass flow rate through a face, ok.

(Refer Slide Time: 37:18)

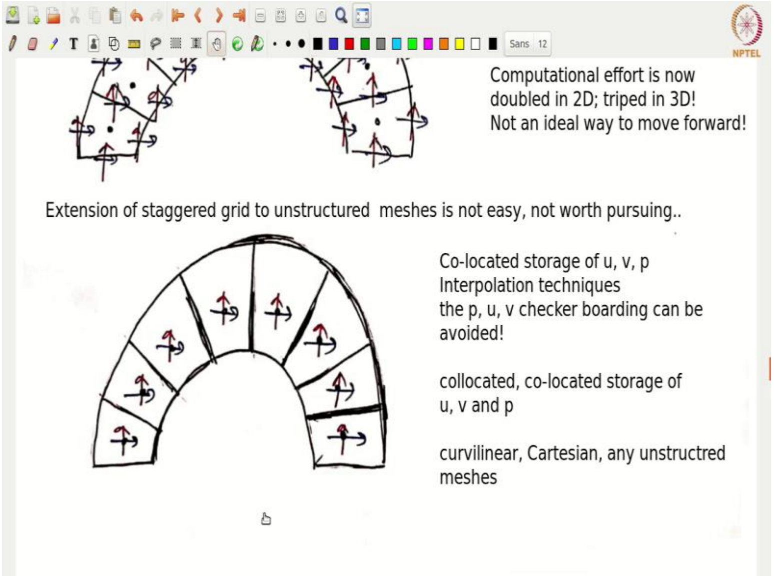


So, this kind of addresses the discretization of continuity equation; however the problem you see clearly is that, this is increases the amount of equations we have to solve. For example, instead of solving for four equations right; essentially one for u east, one for u west, one for v north, one for v south four momentum equations, we are now solving for eight momentum equations. That means, the computational effort has is now doubled in 2 D; of course it will get tripled in 3 dimensions, ok.

This is not very good. As a result, although this method was again tried in the literature, it exists in the literature; this is not an ideal way to move forward ok, because it increases the computational cost by several times. Of course, then what is the preferred way?

That means extending the staggered approach to curvilinear unstructured measures is not very easy. So, extension of staggered grid to unstructured meshes is not easy and it is not worth pursuing, ok. So, as a result, so it is not worth pursuing in its own sense of creating a staggered velocities, ok. So, as a result people finally, came up with reverted back to the co-located approach; that is basically only store both components at the cell centroids, ok.

(Refer Slide Time: 39:07)



So, this is basically the co-located storage of velocities and pressures right, which we said is not good; because it causes pressure velocity checker boarding right, as a result we did not prefer that. But it looks like, if you use some kind of interpolation or interpolation techniques, using some interpolation techniques; the pressure velocity the checker boarding can be avoided.

As a result the collocated or co-located storage of u, v and pressure all at the same location of the cell centroid and the use of the Cartesian components of velocities is preferred both for solving for curvilinear, of course for Cartesian and or for any unstructured meshes.

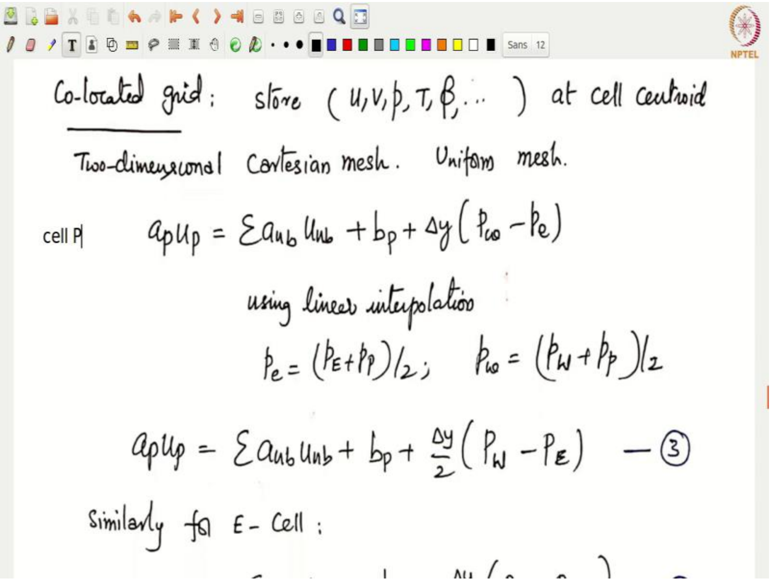
Of course, we have to come up with some kind of an interpolation technique and this interpolation technique that we are going to see is basically does what the staggering has done through a separate grid; this interpolation technique will do through equations, ok.

So, the interpolation technique is basically will do whatever is that is done by the staggering, but through equations, ok. So, we do not really have to store velocities here; but we somehow use interpolation techniques and construct the velocities on the faces from these velocities that are stored at the cell centers, such that we do not run into checker boarding, ok.

So, that is the idea. So, we will revert back to the co-located approach for solving all the Cartesian curvilinear and unstructured meshes; because the extension of the staggered approach to curvilinear unstructured measures is completely not useful at all, ok. So, now we are going to see one particular type of interpolation in this course that is predominantly used in all the software for solving the incompressible fluid flow equations, ok.

Now, if you look at all the packages that are out there, all of them use only this particular interpolation and all of them store co-located all the, all of them store the velocities and pressures in a co-located way, ok.

(Refer Slide Time: 41:41)



So, that is the idea. So, let us get back and look at the equations once again and the concept of the pressure and velocity checker boarding, before we kind of look at the particular interpolation technique, ok. So, let us see what is the problem again to understand it, such that from the equations; let us understand the problem again and then we will kind of devise a method that will fix the problem of the pressure and velocity checker boarding,.

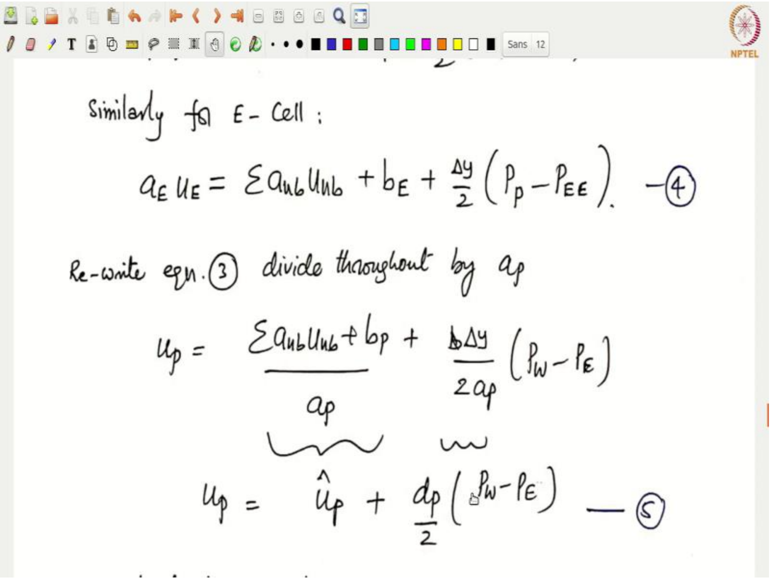
So, we revert back to the co-located grid, where we store the velocities and pressures and temperature and phi everything at the cell centroid. Of course, in the staggered mesh, even the code also does not look good; because now you have so many staggering's that are available, which will make the code very difficult to read and understand as well, ok.

So, in that sense also co-located is much more, much more nicer to work with, ok. So, we store all the velocity components pressure, temperature, any other scalar at the cell centroid, ok. Now, we are also considering a two dimensional Cartesian mesh and we will also for the sake of simplicity use a uniform mesh, alright. Then if you go back to the momentum equation.

Now, let us, now we are back to the cell P ok; because, we do not write it for $a_e u_e$ ok, we are back to the cell P. Then the momentum equation is basically $a_P u_P$ equals $\sum \mathrm{a_{nb}u}_{nb}$ plus $\mathrm{b_P}$ plus $\Delta y(P_w - P_e)$ these are the pressures on the faces. Now, using linear interpolation, if you use a linear interpolation for pressures; then P_w can be written as $(P_W + P_P)/2$

P_e can be written as $(P_E + P_P)/2$, ok. So, it is basically an arithmetic average of the cell values of the pressure, because that is where pressure is stored. That means, our $a_P u_P$ equation will read with these values substituted here as $\sum \mathrm{a_{nb}u}_{nb}$ plus $\mathrm{b_P}$ times $\Delta y(P_w - P_e)$. Note that I think in one of the previous lectures, I on the, I think in the introduction to fluid flow; probably I missed this factor half ok, somebody pointed it out, alright.

(Refer Slide Time: 43:44)



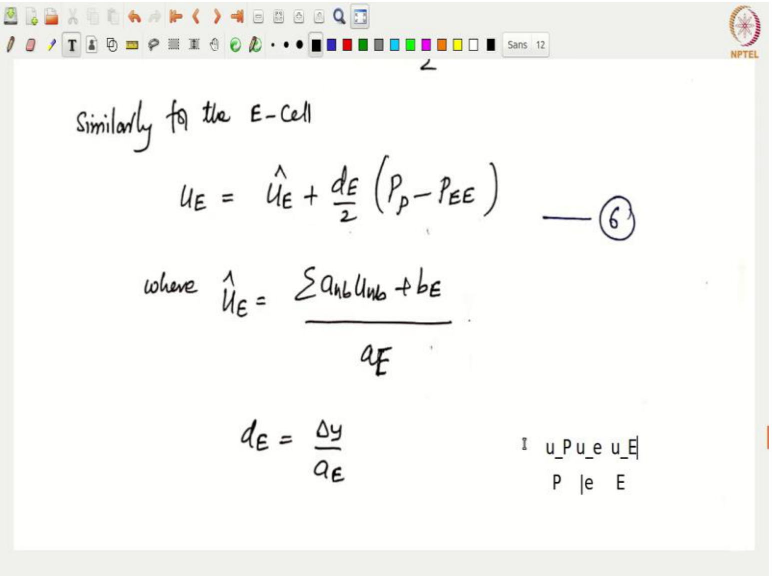
So, then you have this factor half when you use linear interpolation into the momentum equation. Then equation 3 is now your x component of momentum equation for a cell P. Similarly, now if I use the same concept for its east neighbour that is for cell E; I can write a x component of momentum equation, right. That is basically for the cell E, for the cell centroid of the east cell; that will be $a_E u_E$ equals $\sum \mathrm{a_{nb}u}_{nb}$, this will be the neighbours of East cell plus instead of b_P I have b_E plus we have $\Delta y/2$.

And what would be these values? This will be the west cell and the east cell. So, the west cell for the east cell will be west cell for east cell will be P cell, ok. And what will be the east cell for the east cell? East cell of East cell will be East of East; that meanswe end up with the equation 4 which is basically the discrete momentum equation using co-located storage for east cell, ok.

Now, we are doing this basically because, you have momentum equation for cell P and you have momentum equation for cell E from which you can calculate what is the cell centroid values u_P and u_E. But eventually if you write the continuity equation, then you would need the face value of u little e right; that means you need u_P plus u_E by 2 or something like that, which will of course lead to some kind of velocity checker boarding and pressure checker boarding that is what we are kind of going to see, alright.

Now, again I can rewrite equation 3 as by dividing with a_P everywhere; I can rewrite this as u_P equals $\sum \mathrm{a_{nb}} \mathrm{u}_{nb} + b_P$ upon a_P plus you have $\frac{\Delta y}{2a_P}$ times $(P_W - P_E)$. Like we have what we have done in this simple revised algorithm, let us call this coefficient as some hat velocity. So, your u at the cell centroid P, u_P equals $\hat{u}_P$ plus; let us call this as some $\Delta y/a_P$ as some d_P, ok. Some coefficient d, that is $d_P/2$ times $(P_W - P_E)$, ok.

(Refer Slide Time: 46:03)

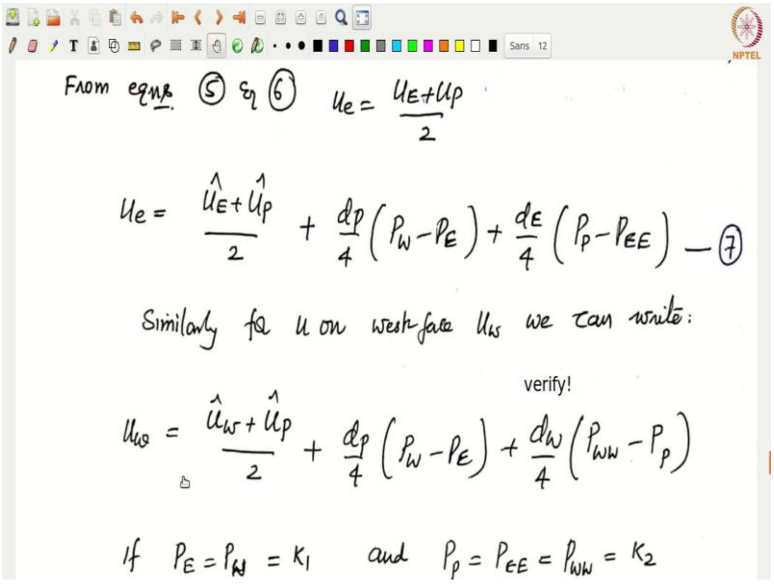


So, we obtained one equation for u_P. Similarly, I can write for u_E as well right; that is basically divide this equation 4 by this center coefficient that is $\sum \mathrm{a_{nb}} \mathrm{u}_{nb} + b_E$ upon u_E plus $\frac{\Delta y}{2a_E}$ times $P_P - P_{EE}$. That means, we got u_E equals $\hat{u}_E$ plus $d_E/2$ times $P_P - P_{EE}$, alright, where $\hat{u}_E$ is basically this quantity and d_E is this $\frac{\Delta y}{a_E}$, ok.

So, we left the factor two here, alright. So, now, we got essentially velocities at the cell centroids of P and the East cell. Now, we want to calculate what is the velocity on the face that is basically we want to calculate on the face little e; because your East cell is here and your P cell is here right, essentially we want to calculate on the particular face that is this one, ok.

How do we do this? Of course, we can it is; because we have assumed it to be uniform mesh, we can take it as a an arithmetic average of the velocities of u_P and u_E to calculate u_e right, essentially we have u_P calculate u_e from u_E and u_P, right ok.

(Refer Slide Time: 47:25)



Let us do that; that means I can calculate what is velocity on the face as the arithmetic average of these cell values. If I do that, what I get is basically u_E equals I have $\hat{u}_P$ plus $\hat{u}_E$ by 2 right; because I am adding these two up by 2.

So, this is these two up by 2, that is $\hat{u}_E$ plus $\hat{u}_P$ by 2 plus we have this plus this by 2; that is basically each of them is going to get a factor of 1 by 2 in the front. So, this will be d_P by 4 times pressure difference plus d_E by 4 times $P_P - P_{EE}$. So, that is the final equation we get, basically the velocity on the face e is equal to $\hat{u}_E$ plus $\hat{u}_P$ by 2 plus d_P times $P_W - P_E$ by 4 plus d_E times $P_P - P_{EE}$ by 4.

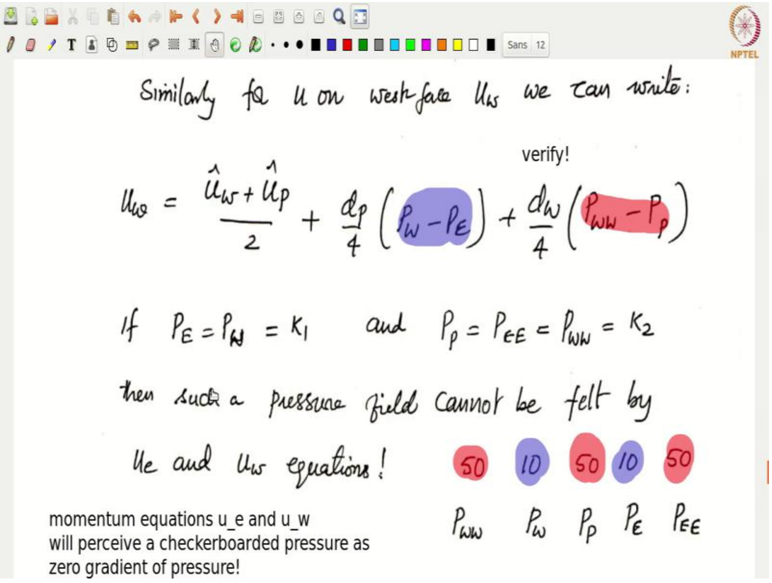
Let us call this equation number 7. Now, this equation is basically an arithmetic average of the cell centroid velocities to obtain the face velocity, correct. We basically have done not much here; basically what we have done is, we took the momentum equations, we have rewritten it such that we calculate the u at the P cell and the East cell and took an average of that, that is all we kind of use this concept of hats and pressures to make it simple to write, right ok.

Now, this is the velocity for the East face; we will also need a velocity for the west face, right. And similarly you can write if you do the entire algebraic; you will get u_w will be $\hat{u}_W$ plus $\hat{u}_P$ by 2 similar to this plus you get d P by 4 and you get $P_W - P_E$ coming from the d P.

Then you get instead of d_E by 4 you will get a d_W by 4; instead of $P_P - P_{EE}$ for the East face, you will get for the west face you will get $P_{WW} - P_P$ ok. So, this is something you have to verify, alright. So, essentially you got an equation for u_w.

So, each of these velocities what they suggest is basically; your the face velocity has pressures which are basically P_P, P_E, P_W, P_{EE} and P_{WW}, right. If you look at East and West, the velocities are there now; that means if P_E equals P_W, that means this term is zero.

(Refer Slide Time: 49:52)



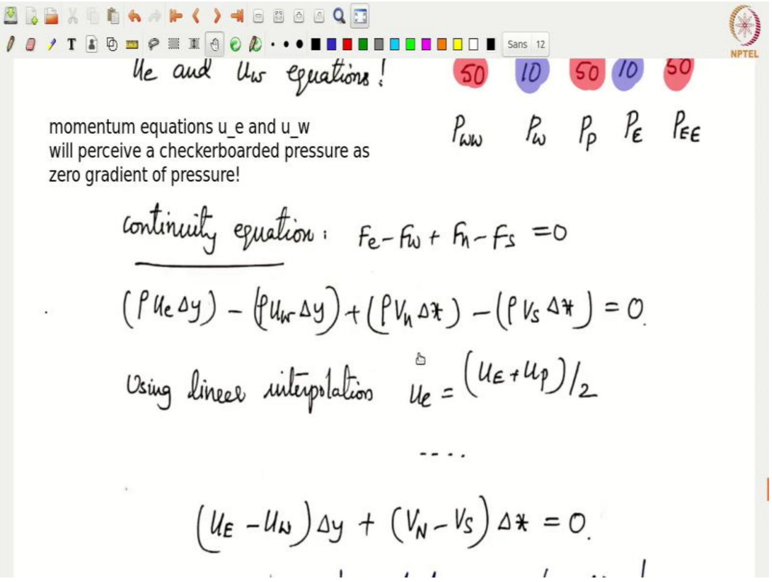
Similarly, P_E equals P_W this term is zero; that means if P_E equals P_W equal to some constant k_1 and P_P equals P_E is equals P_{WW} equals some other constant k_2. Then if you have such a pressure field, then your face velocities will see that as zero pressure gradient, right.

That means, if I have a pattern like this, if I have let us say my P_P, P_E, P_W, P_{EE}, P_{WW} such that I have 50, 10, 50, 10; then essentially the 10 and here we, because both are equal will make this term to be 0 right as well as this term also to be 0, right.

Similarly, the 50 coming from here, here and here will make P_W equals P_{WW} equals P_P. So, this term will be 0 and also this term will be 0; that means the pressure gradient although it looks like a checker boarded pressure which is 50, 10, 50, 10 and 50, this will be not felt by the momentum equations, ok.

That means, the momentum equations for u e and u w will perceive a checker boarded pressure as zero gradient of pressure; that means they support the checker boarding of pressure concept ok, the momentum equations. Now, the idea is how do we fix this? Now, what we do is we will not fix this in this particular case for the momentum equations; we will fix it for the continuity equation.

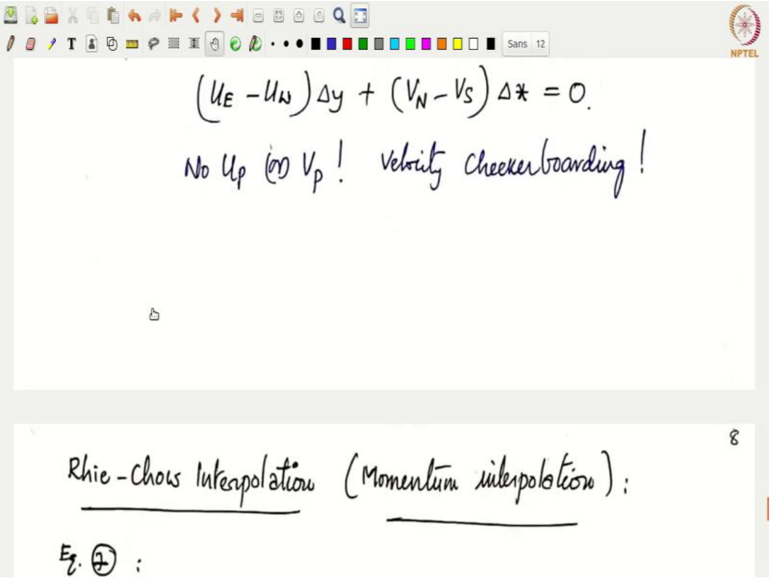
(Refer Slide Time: 51:43)



So, let us look at the continuity equation. The continuity equation is F_e minus F_w plus F_n minus F_s equal to 0. So, that means if you look at the continuity equation; if you substitute for $\rho u_e \Delta y$ in terms of the flow rates, we get $\rho u_e \Delta y$ minus $\rho u_w \Delta y$ plus $\rho v_n \Delta x$ minus $\rho v_s \Delta x$ equals 0.

So, again if you use linear interpolation for the face velocities, what you get is; u_E plus u_P by 2, and so on. And the basically and then you have, if you plug in the face velocities back into this you get what you get is basically u_E, u_W; because u_P gets cancelled v_N and v_S.

(Refer Slide Time: 52:30)



So, basically what you get is $(u_E - u_W)\Delta y$ plus $(v_N - v_S)\Delta x$ equals 0; that means there is no u_P or v_P in this discrete continuity equation. So, as a result this supports velocity checker boarding, ok. So, that means the momentum equations support pressure checker boarding, and the continuity equation supports velocity checker boarding.

Now, if both equations stops support the checker boarding; both the pressure and velocity have to be satisfied by both continuity and momentum equations, ok. So, as long as both are supporting it, then these may still remain in the final solution.

As a result what we try to do is, we will let the momentum equation still support checker boarding; whereas we will fix the continuity equation to not support checker boarding of the velocities or the pressures. As a result the final solution will not have this checker boarding in the in the solution, ok.

So, we will look at the Rhie Chow interpolation and we could not do it; I think we are already out of time. So, we will look at basically Rhie Chow interpolation that is also known as momentum interpolation and see how to formulate the co-located meshes in the next lecture, ok.

So, I am going to stop here. I will, we will pick it up from the co-located approach for solving the incompressible flow equations using the Rhie Chow interpolation in the next lecture, alright. If you have any questions, do let me know through email; I will get back to you, ok.

Thank you.