

# Solving the Navier-Stokes equations

## Numerical Flow Simulation

Edouard Boujo

Fall 2025

# Reminder: the NS equations

- General conservation equation:

$$\frac{\partial(\rho\phi)}{\partial t} + \text{div}(\rho\phi\mathbf{u}) = \text{div}(\Gamma \text{grad}(\phi)) + S$$

- So far, solved for  $\phi$  assuming known density and velocity.
- In general, velocity field not known and must be computed too.
- The Navier-Stokes eq. themselves have the same general form (week 2):

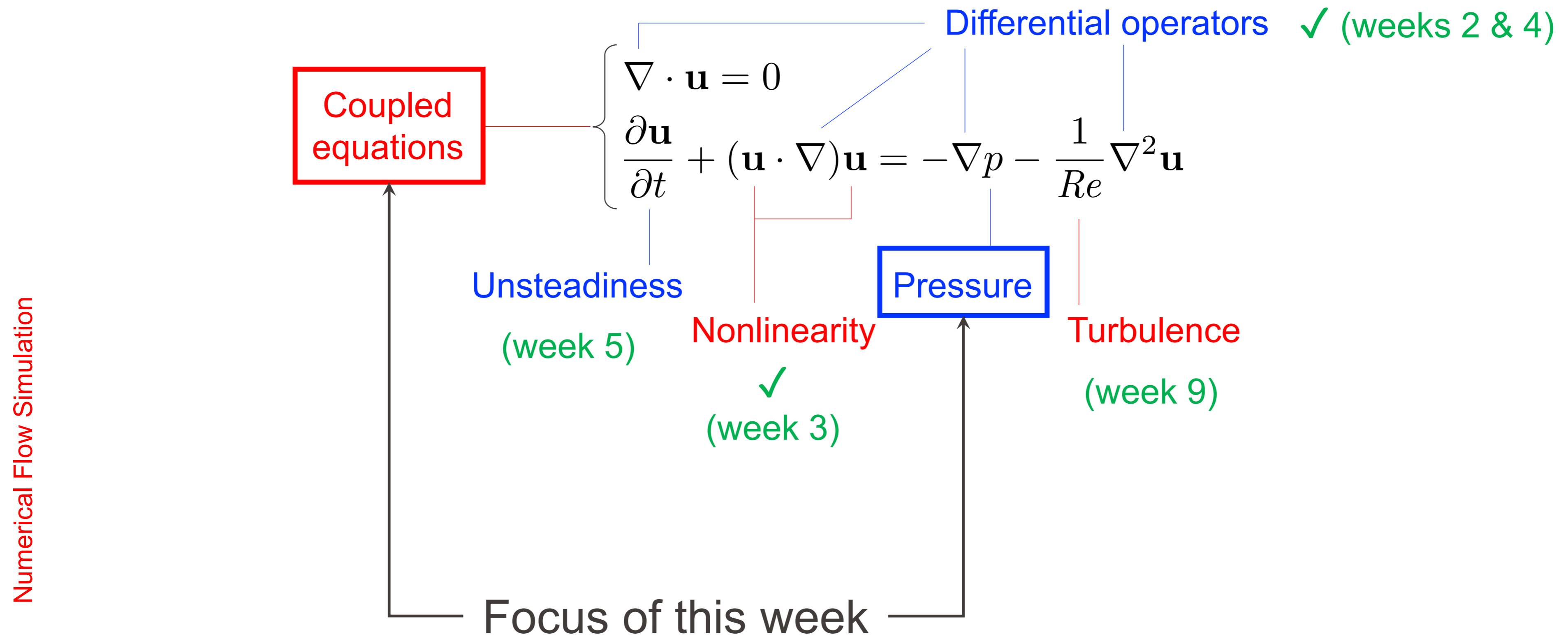
$$\frac{\partial\rho}{\partial t} + \text{div}(\rho\mathbf{u}) = 0$$

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \text{div}(\rho\mathbf{u}\mathbf{u}) = \text{div} \left[ \left( -p - \frac{2}{3}\mu \text{div}(\mathbf{u}) \right) \mathbf{I} + 2\mu\mathbf{d} \right] + \rho\mathbf{f}$$

- All the methods seen so far should apply?...

# Reminder: notoriously difficult equations

- Some elements that make the Navier-Stokes equations difficult to solve:



# Issue 1: coupled equations

- The variable solved for in each eq. appears in the other equations.

For ex. steady incompressible 2D:

$$\left\{ \begin{array}{l} \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = 0 \\ \frac{\partial(\rho u u)}{\partial x} + \frac{\partial(\rho u v)}{\partial y} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial u}{\partial y} \right) \\ \frac{\partial(\rho u v)}{\partial x} + \frac{\partial(\rho v v)}{\partial y} = -\frac{\partial p}{\partial y} + \frac{\partial}{\partial x} \left( \mu \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial v}{\partial y} \right) \end{array} \right.$$

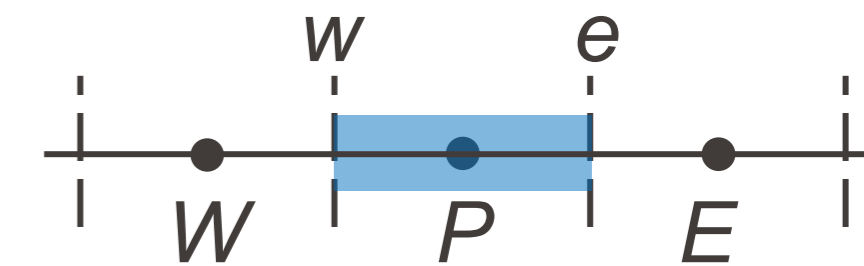
- **Coupled method:**

- In principle, can solve all equations simultaneously.  
Define  $\phi = (u, v, w, \rho)$  or  $(u, v, w, p)$  and solve a single system.
- Works best for linear, strongly coupled equations. Better convergence, but more memory requirement. Sometimes not possible with large meshes.

- **Segregated method:**

- Solve each equation one by one, iteratively.
- Works best for nonlinear equations. Slower convergence, but less memory requirement.

# Issue 2: pressure checkerboard



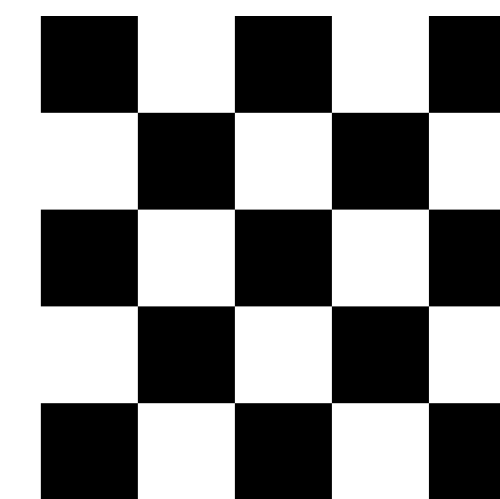
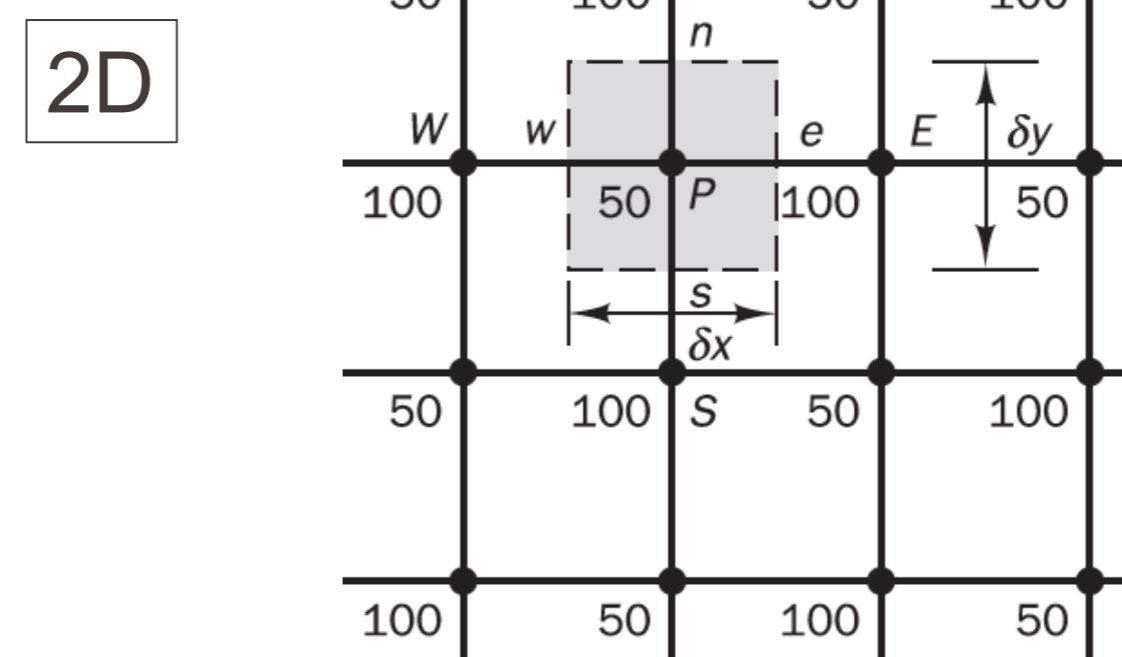
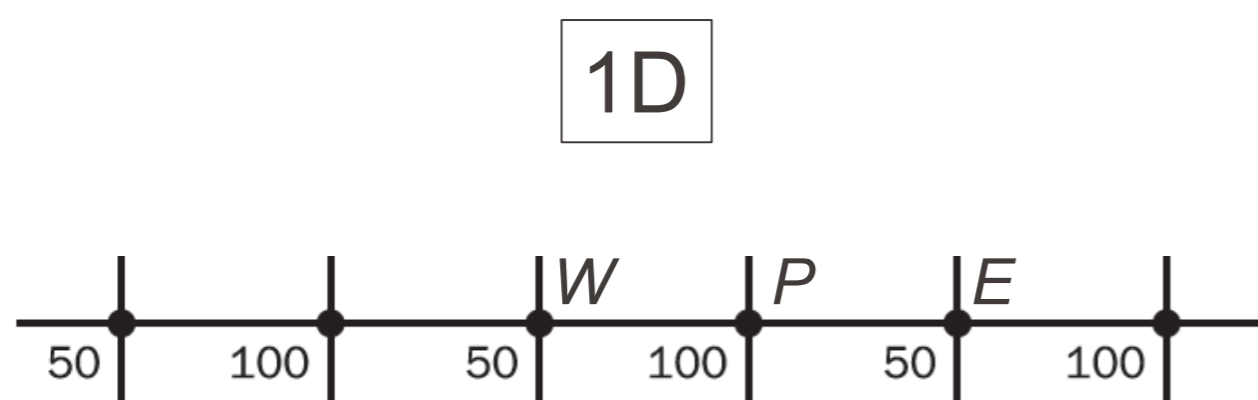
- Example: steady 1D momentum equation 
$$\frac{\partial(\rho u u)}{\partial x} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right)$$

- Discretization of the pressure gradient with central differencing:

$$-\int_w^e \frac{\partial p}{\partial x} dx = p_w - p_e = \frac{p_P + p_W}{2} - \frac{p_E + p_P}{2} = \boxed{\frac{p_W - p_E}{2}}$$

- Involves nodes  $W$  and  $E$  that are 2 CVs apart. The central node  $P$  does not appear. Risk of “**checkerboard pressure mode**”: not physical, but numerically possible because does not contribute to the discretized momentum eq. (zero pressure gradient, just like a uniform pressure field).

Numerical Flow Simulation



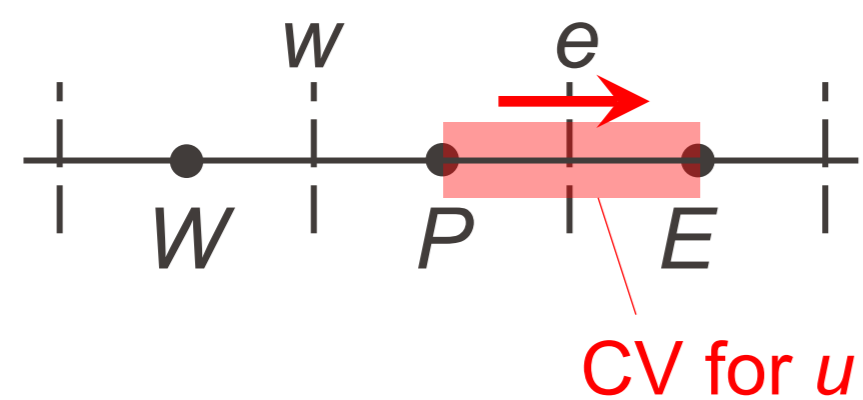
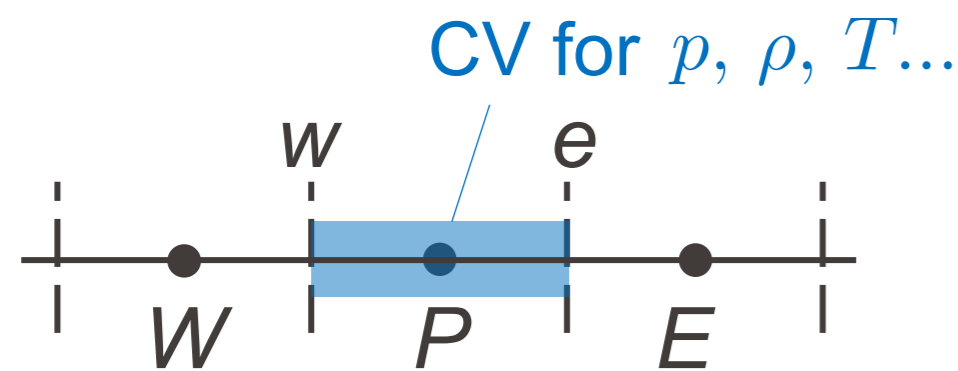
# Staggered grid

- Remedy to checkerboard pressure: staggered grid
  - Evaluate **pressure** (and other scalars  $\phi$ ) at the nodes of the **original mesh**
  - Evaluate **velocities** at the nodes of a **staggered mesh**
  - CV centers of the staggered mesh correspond to face centers of the original mesh

Numerical Flow Simulation

1D

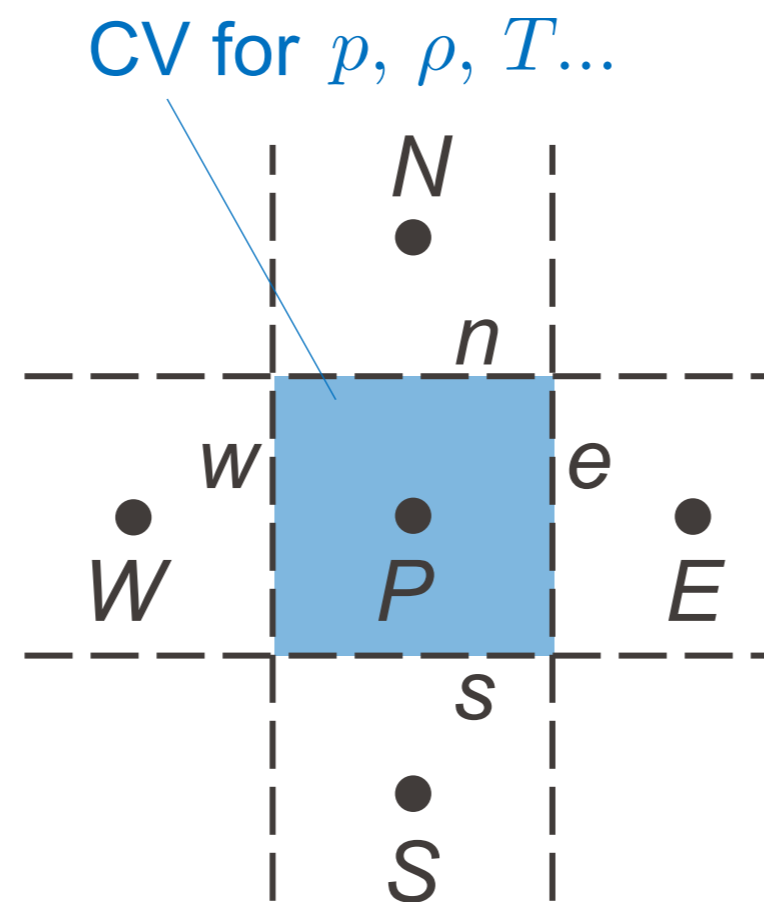
Original mesh



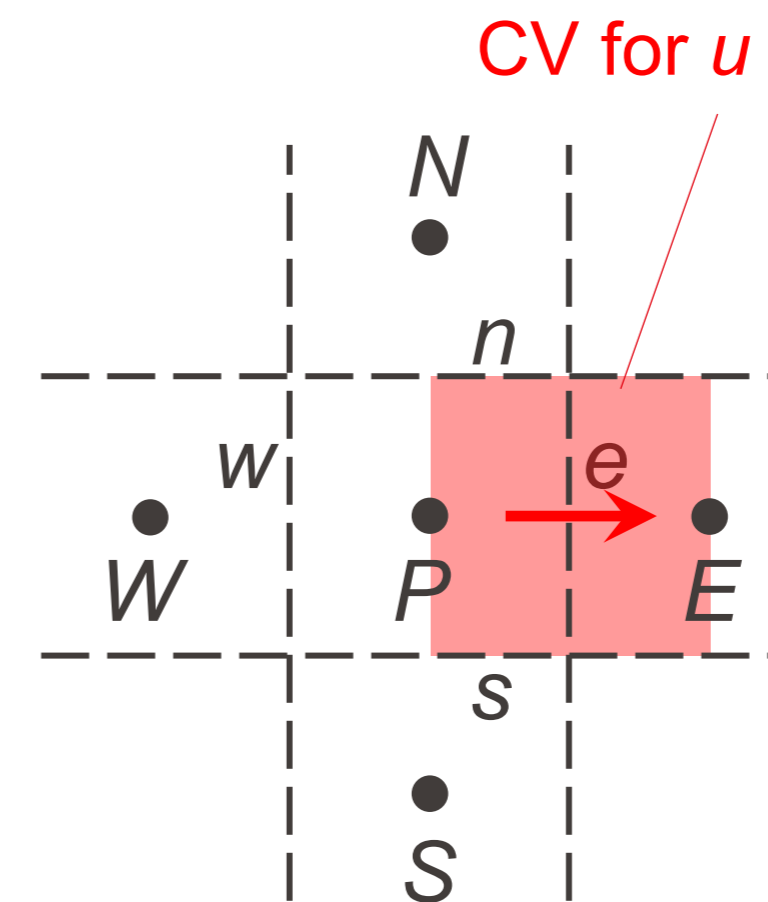
Staggered mesh

2D

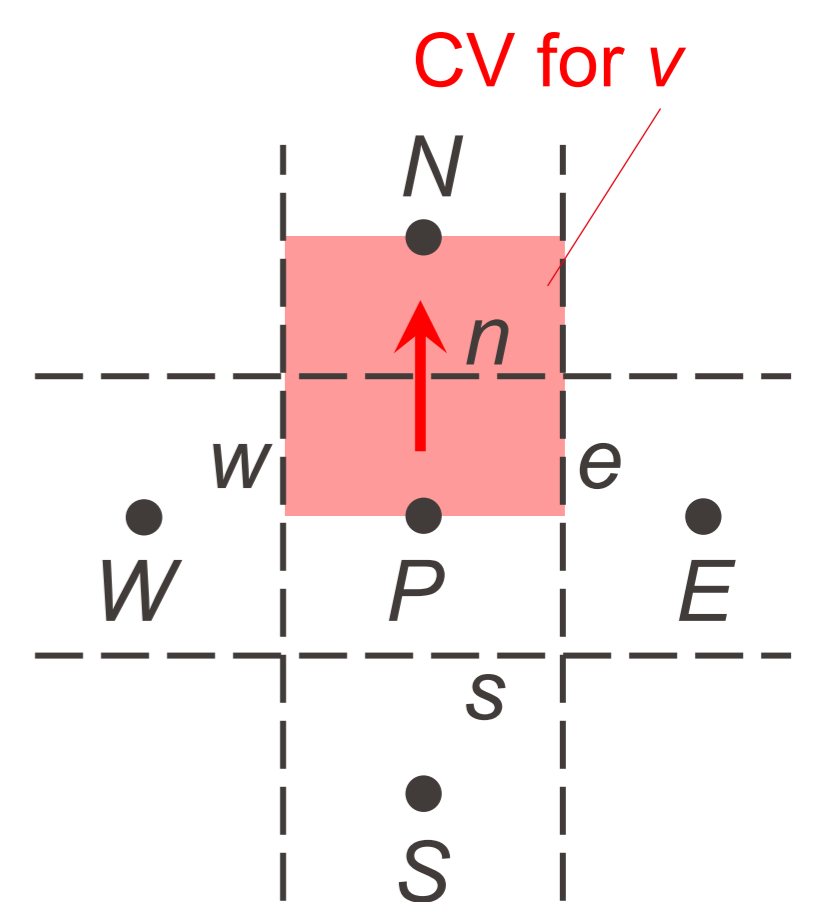
Original mesh



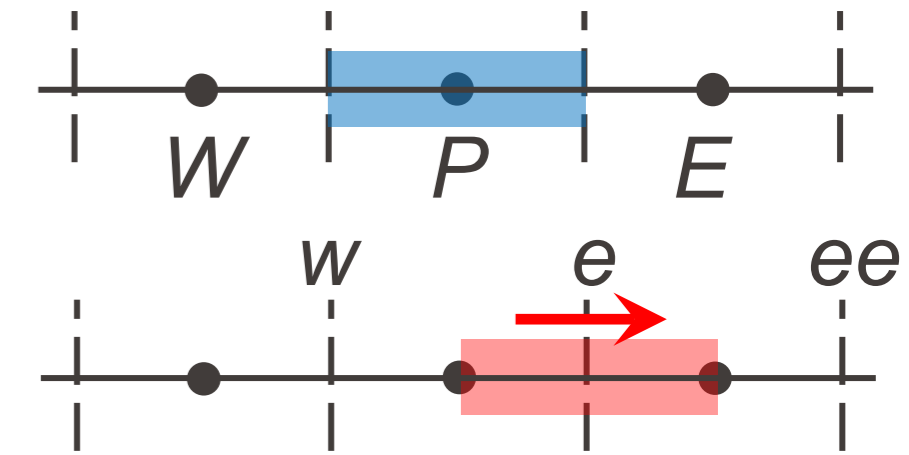
Mesh staggered in x



Mesh staggered in y



# Staggered grid



- 1D discretization for  $u$  on the **staggered mesh**:

$$\frac{\partial(\rho u)}{\partial x} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right)$$

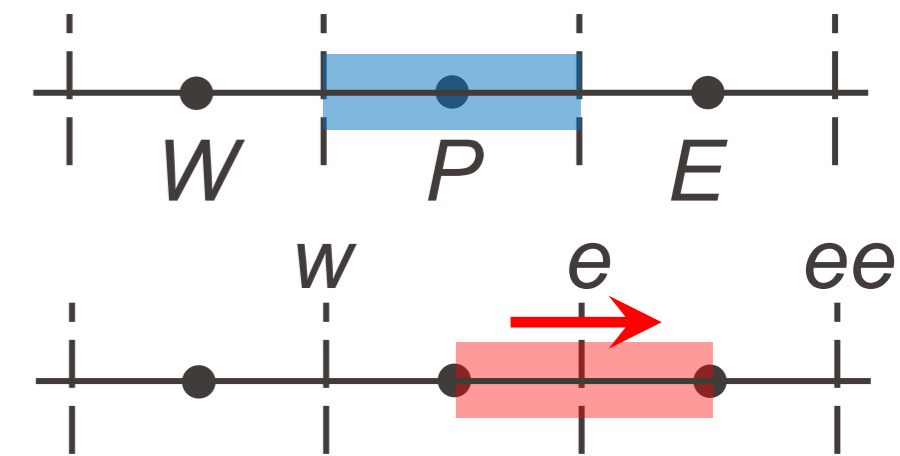
$$\int_P^E \frac{\partial(\rho u)}{\partial x} dx = - \int_P^E \frac{\partial p}{\partial x} dx + \int_P^E \frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right)$$

(Assume constant cross-section area  $A$ )

$$(\rho u)_E - (\rho u)_P = \boxed{p_P - p_E} + \left( \mu \frac{\partial u}{\partial x} \right)_E - \left( \mu \frac{\partial u}{\partial x} \right)_P$$

- No risk of checkerboard pressure any more, because would give a non-zero contribution.
- Physically consistent: pressure difference between neighboring CVs drives the flow across the face in between.

# Staggered grid



- 1D discretization for  $u$  on the **staggered mesh**:

$$(\rho u u)_E - (\rho u u)_P = p_P - p_E + \left( \mu \frac{\partial u}{\partial x} \right)_E - \left( \mu \frac{\partial u}{\partial x} \right)_P$$

For instance, using CD for the diffusion term:

$$F_E u_E - F_P u_P = p_P - p_E + D_E (u_{ee} - u_e) - D_P (u_e - u_w)$$

Recall the notations  
(week 4):  $F = \rho u^*$   
 $D = \frac{\mu}{\Delta x}$

Use any scheme for the convective term to express velocities at faces (week 4).

Obtain an algebraic equation:

$$\underline{a_e u_e} = a_w u_w + \underline{a_{ee} u_{ee}} + \underline{(p_P - p_E)}$$

Coefficients of face values depend on  $D$  and  $F$  at the nodes (CV centers):

$$a_f = a_f(D_C, F_C)$$

Need for  $C \rightarrow f$  and  $f \rightarrow C$  interpolation:

$$F_C = (\rho u^*)_C = \frac{1}{2} ((\rho u^*)_{f_1} + (\rho u^*)_{f_2}) \quad \rho_f = \frac{1}{2} (\rho_{C_1} + \rho_{C_2})$$

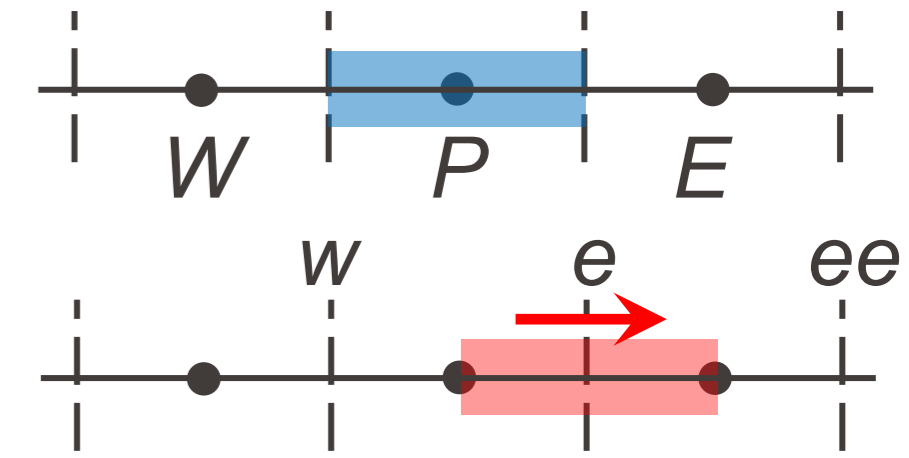
Compare with discretization on **original mesh** (previous weeks)... and don't get lost!

$$a_P \phi_P = a_W \phi_W + a_E \phi_E + b$$

Coefficients of nodal values depend on  $D$  and  $F$  at the faces:

$$a_C = a_C(D_f, F_f)$$

# Staggered grid



- 1D discretization of the continuity eq. on the **original mesh**:

$$\frac{d(\rho u)}{dx} = 0 \quad \rightarrow \quad \frac{(\rho u)_e - (\rho u)_w}{F_e - F_w} = 0$$

Velocities at faces available (no interpolation required).

# Staggered grid

- Momentum equations in 2D (staggered mesh):

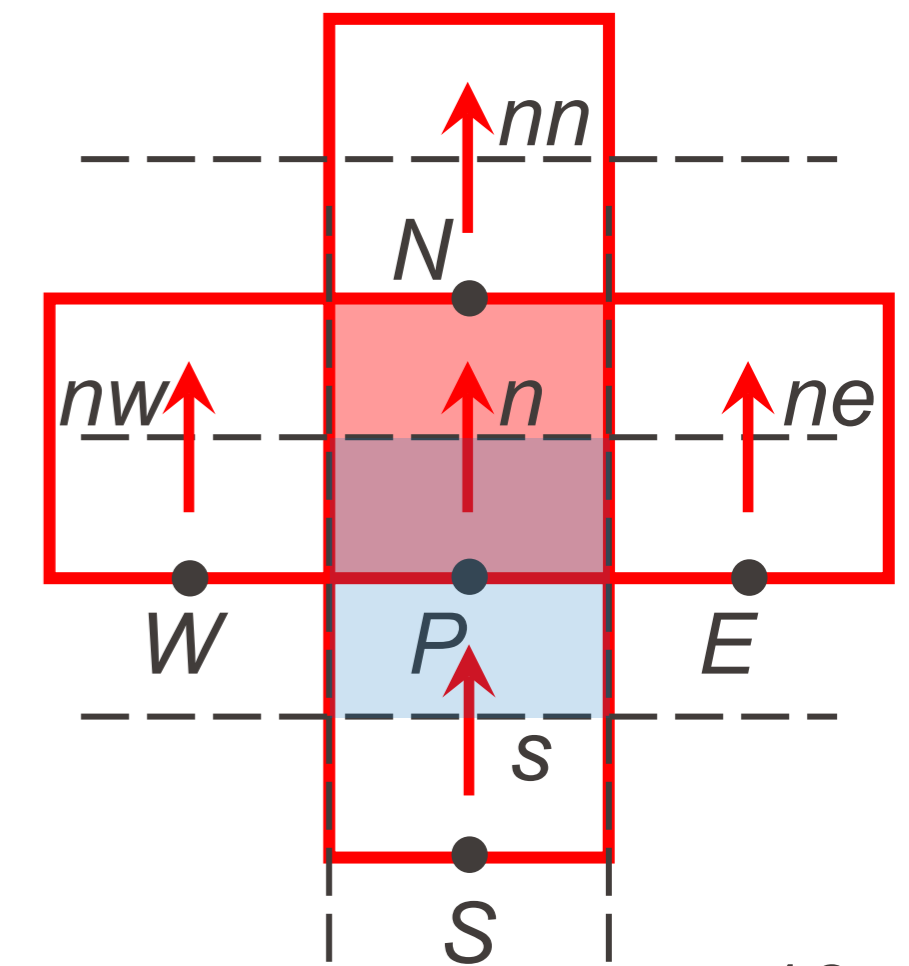
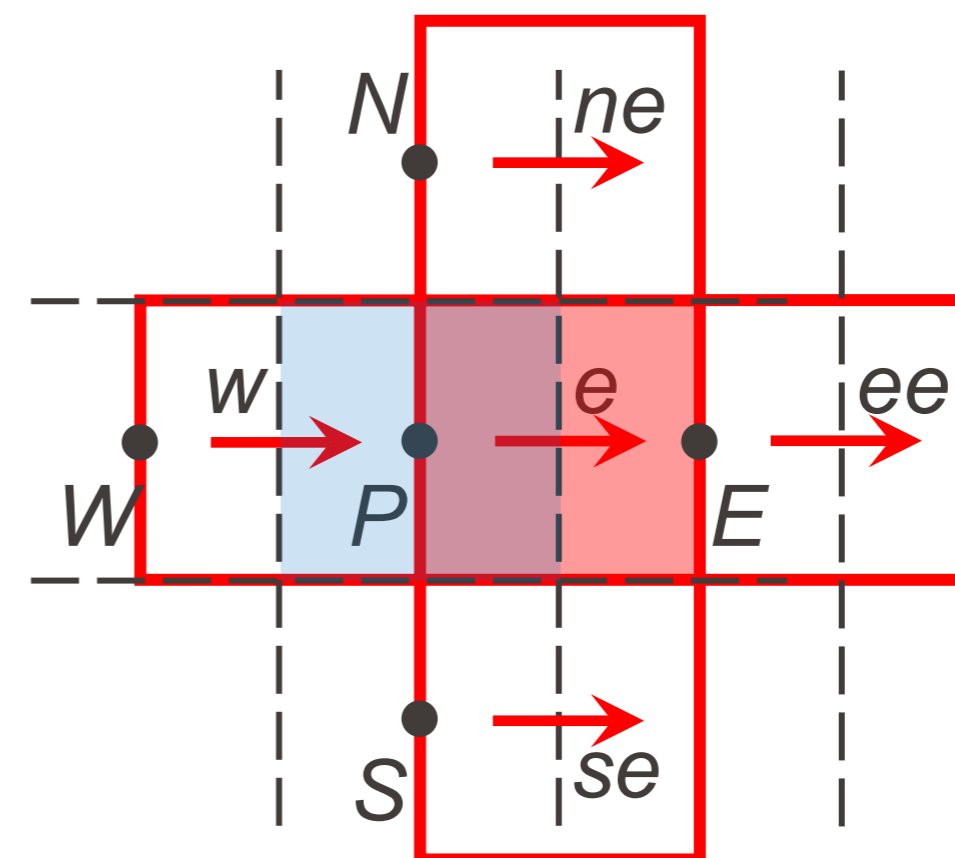
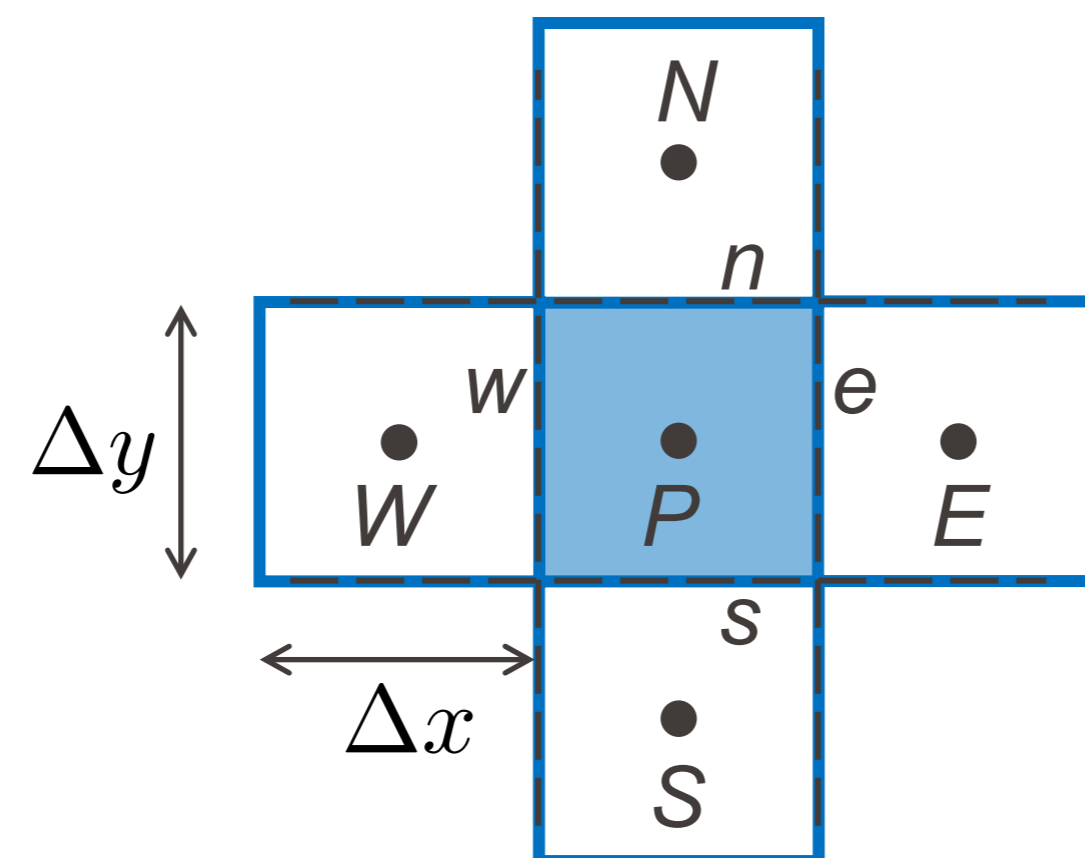
$$a_f = a_f \left( D_C = \left( \frac{\mu}{\Delta} \right)_C, F_C = \frac{F_{f1} + F_{f2}}{2} \right)$$

$$\begin{cases} a_e u_e = \sum_{f=w, ee, ne, se} a_f u_f + (p_P - p_E) A_e \\ a_n v_n = \sum_{f=nw, ne, nn, s} a_f v_f + (p_P - p_N) A_n \end{cases}$$

- Continuity eq. in 2D (original mesh):  $F_e - F_w + F_n - F_s = 0$

- Other scalar unknowns (original mesh):  $a_P \phi_P = \sum_C a_C \phi_C + b$

$$a_C = a_C \left( D_f = \left( \frac{\mu}{\Delta} \right)_f, F_f \right)$$



# Issue 3: no equation for pressure?

- **Compressible** flows: continuity eq. = eq. for density. Can deduce pressure with an equation of state such as  $p = p(\rho)$ . (Pressure = “thermodynamic” variable.)

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u}) = 0 \quad \frac{\partial(\rho \mathbf{u})}{\partial t} + \operatorname{div}(\rho \mathbf{u} \mathbf{u}) = \operatorname{div} \left[ \left( -p - \frac{2}{3} \mu \operatorname{div}(\mathbf{u}) \right) \mathbf{I} + 2\mu \mathbf{d} \right]$$

- **Incompressible** flows: constant density, no eq. for pressure. The role of pressure is to enforce a constraint: continuity. (Pressure = “mathematical” variable.)

$$\operatorname{div}(\mathbf{u}) = 0 \quad \frac{\partial(\rho \mathbf{u})}{\partial t} + \operatorname{div}(\rho \mathbf{u} \mathbf{u}) = -\operatorname{grad}(p) + \operatorname{div}(\mu \operatorname{grad}(\mathbf{u}))$$

No problem when solving steady problems with a coupled method. However, when solving unsteady problems and/or when using a segregated method, need to find a way to compute pressure.

# Poisson eq. for pressure

$$\nabla \cdot \mathbf{u} = 0$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u}$$

- Key idea: obtain a scalar eq. for pressure (Poisson eq.) by taking the divergence of the momentum eq.

$$\nabla \cdot \left[ \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} \right]$$

$$\frac{\partial(\nabla \cdot \mathbf{u})}{\partial t} + \nabla \cdot [(\mathbf{u} \cdot \nabla) \mathbf{u}] = -\frac{1}{\rho} \nabla^2 p + \nu \nabla^2 (\nabla \cdot \mathbf{u})$$

- If the velocity field satisfies continuity (divergence free):

$$\nabla \cdot [(\mathbf{u} \cdot \nabla) \mathbf{u}] = -\frac{1}{\rho} \nabla^2 p \quad \longrightarrow \quad \text{Can solve for } p \text{ if } \mathbf{u} \text{ is known.}$$

- Need a boundary condition: project the momentum eq. on the boundary normal

$$\left[ \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} \right] \cdot \mathbf{n} \quad \rightarrow \quad \nabla p \cdot \mathbf{n} = \rho \left[ -\frac{\partial \mathbf{u}}{\partial t} - (\mathbf{u} \cdot \nabla) \mathbf{u} + \nu \nabla^2 \mathbf{u} \right] \cdot \mathbf{n}$$

# Unsteady incompressible NS

$$\nabla \cdot \mathbf{u} = 0$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u}$$

- Assume an **explicit** temporal scheme (week 5) for velocity:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^n \cdot \nabla) \mathbf{u}^n = -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^n$$

Take the divergence:  $\frac{\cancel{\nabla \cdot \mathbf{u}^{n+1}} - \cancel{\nabla \cdot \mathbf{u}^n}}{\Delta t} + \nabla \cdot [(\mathbf{u}^n \cdot \nabla) \mathbf{u}^n] = -\frac{1}{\rho} \nabla^2 p^{n+1} + \nu \nabla^2 (\cancel{\nabla \cdot \mathbf{u}^n})$

Require divergence-free velocity at new time step

$$\nabla \cdot [(\mathbf{u}^n \cdot \nabla) \mathbf{u}^n] = -\frac{1}{\rho} \nabla^2 p^{n+1}$$

Divergence-free velocity from previous time step

- Solve Poisson eq. for new  $p^{n+1}$  (linear system).

- Compute new velocity: 
$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \left[ -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^n - (\mathbf{u}^n \cdot \nabla) \mathbf{u}^n \right]$$

Not applicable to **implicit** temporal schemes (because  $\mathbf{u}^{n+1}$  would appear in eq. for  $p^{n+1}$ ).

# Helmholtz-Hodge decomposition (pressure projection)

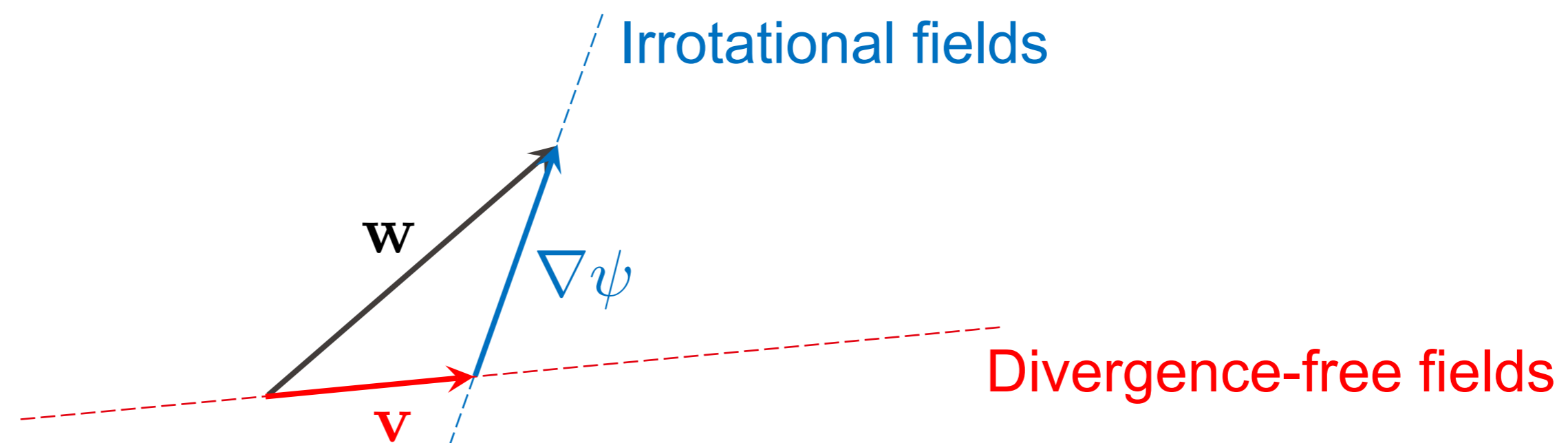
- Helmholtz-Hodge decomposition: any vector field is the sum of a solenoidal (divergence-free) part and an irrotational part:

$$\mathbf{w} = \underbrace{\mathbf{v}}_{\substack{\text{solenoidal} \\ \nabla \cdot \mathbf{v} = 0}} + \underbrace{\nabla \psi}_{\substack{\text{irrotational} \\ \nabla \times (\nabla \psi) = 0}}$$

$\psi$  is solution of a Poisson eq.:  $\nabla \cdot \mathbf{w} = \cancel{\nabla \cdot \mathbf{v}} + \nabla^2 \psi = \nabla^2 \psi$

→ can extract the divergence-free part of any field:  $\mathbf{v} = \mathbf{w} - \nabla \psi$

- Geometric interpretation: projection onto the space of divergence-free fields.



- The incompressible NS eq. can be seen as such a projection onto divergence-free velocity fields. This projection is done via the pressure  $p$ .

# Unsteady incompressible NS: fractional step method

- Idea:

1. Compute tentative velocity from momentum eq., ignoring pressure,
2. Compute the divergence-free component part of  $\mathbf{u}^*$ .

- Assuming again an **explicit** temporal scheme:

Split  $\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^n \cdot \nabla)\mathbf{u}^n = -\frac{1}{\rho}\nabla p^{n+1} + \nu\nabla^2\mathbf{u}^n$  into

$$\left\{ \begin{array}{l} \frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^n \cdot \nabla)\mathbf{u}^n = -\nu\nabla^2\mathbf{u}^n \\ \frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\frac{1}{\rho}\nabla p^{n+1} \end{array} \right.$$

1. Compute tentative velocity  $\mathbf{u}^*$  with 1<sup>st</sup> eq.

Take divergence of 2<sup>nd</sup> eq.:  $\frac{\nabla \cdot \mathbf{u}^{n+1} - \nabla \cdot \mathbf{u}^*}{\Delta t} = -\frac{1}{\rho}\nabla^2 p^{n+1} \rightarrow \frac{\rho}{\Delta t}\nabla \cdot \mathbf{u}^* = \nabla^2 p^{n+1}$   
 (require div.-free  $\mathbf{u}^{n+1}$ )

- 2.a. Solve Poisson eq. for new  $p^{n+1}$  (linear system)

- 2.b. Compute new velocity:  $\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\Delta t}{\rho}\nabla p^{n+1}$  It's indeed a HH decomposition:  $\mathbf{v} = \mathbf{w} - \nabla\psi$

Also applicable to **implicit** temporal schemes ( $\mathbf{u}^{n+1}$  doesn't appear in eq. for  $p^{n+1}$ ). 15

# Unsteady incompressible NS: fractional step method

- Can also keep pressure term, solve with  $p^n$  as a guess, and solve for a pressure correction (instead of pressure). Example with **implicit** scheme:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^{n+1} \cdot \nabla) \mathbf{u}^{n+1} = -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^{n+1} \quad \left\{ \begin{array}{l} \frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^* \cdot \nabla) \mathbf{u}^* = -\frac{1}{\rho} \nabla p^n + \nu \nabla^2 \mathbf{u}^* \\ \frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\frac{1}{\rho} \nabla (p^{n+1} - p^n) = -\frac{1}{\rho} \nabla p' \end{array} \right.$$

1. Compute tentative velocity  $\mathbf{u}^*$  with 1<sup>st</sup> eq.

Take divergence of 2<sup>nd</sup> eq. (require divergence-free  $\mathbf{u}^{n+1}$ ):  $\frac{\rho}{\Delta t} \nabla \cdot \mathbf{u}^* = \nabla^2 p'$

- 2.a. Solve Poisson eq. for correction  $p'$  (linear system)

- 2.b. Compute new velocity:  $\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\Delta t}{\rho} \nabla p'$

Actually, the sum of the 2 eq. is not exactly the intended scheme:  $\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^* \cdot \nabla) \mathbf{u}^* = -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^*$

But the error is of 2<sup>nd</sup> order in time, consistent with other errors of 2<sup>nd</sup>-order temporal schemes.

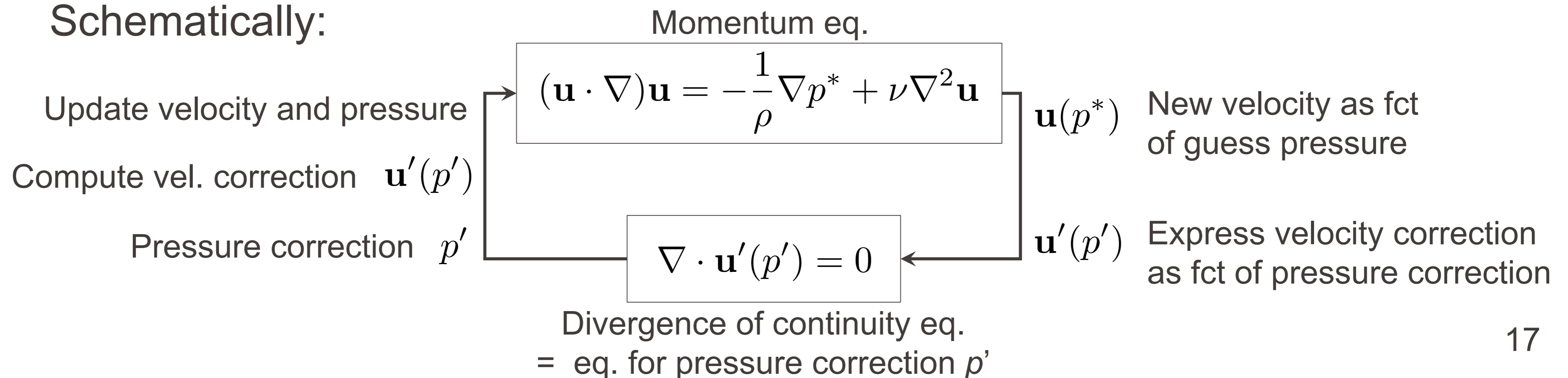
# Steady incompressible NS

$$\nabla \cdot \mathbf{u} = 0$$

$$(\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\mathbf{u}$$

- For steady equations, can't split the time derivative into 2 parts. Use instead other types of “predictor-corrector” approaches.
- Idea:
  1. Solve the momentum eq., using a guess velocity to linearize the nonlinear convective term (Picard) + a guess pressure,
  2. Use the continuity eq. to deduce an eq. for pressure or pressure correction,
  3. Update velocity and pressure; iterate.

Schematically:



# Steady incompressible NS: the SIMPLE algorithm

“Semi-Implicit Method for Pressure-Linked Equations” (originally developed for unsteady flows).

1. Solve momentum eq. with guess velocity (for nonlinear term) and guess pressure  $p^*$   $\rightarrow$  new (tentative) velocity  $\mathbf{u}^*$ . For ex. (2D, staggered grid):

$$a_e u_e^* = \sum_f a_f u_f^* + (p_P^* - p_E^*) A_e \quad \rightarrow u_e^*$$
$$a_n v_n^* = \sum_f a_f v_f^* + (p_P^* - p_N^*) A_n \quad \rightarrow v_n^*$$

At this stage,  $\mathbf{u}^*$  has no reason to satisfy continuity  $\rightarrow$  introduce velocity correction  $\mathbf{u}'$  and pressure correction  $p'$ , aimed at enforcing continuity:

$$\mathbf{u} = \mathbf{u}^* + \mathbf{u}', \quad p = p^* + p'$$

# Steady incompressible NS: the SIMPLE algorithm

2. Derive an equation for the pressure correction as follows:

- Subtract momentum eq. for  $\mathbf{u}^*$  from those for  $\mathbf{u}$

$$a_e(u_e - u_e^*) = \sum a_f(u_f - u_f^*) + [(p_P - p_E) - (p_P^* - p_E^*)] A_e \quad \rightarrow \quad a_e u'_e = \cancel{\sum a_f u'_f} + (p'_P - p'_E) A_e$$

$$a_n(v_n - v_n^*) = \sum a_f(v_f - v_f^*) + [(p_P - p_N) - (p_P^* - p_N^*)] A_n \quad \rightarrow \quad a_n v'_n = \cancel{\sum a_f v'_f} + (p'_P - p'_N) A_n$$

- Main approximation of the SIMPLE algorithm: neglect neighbor contributions in the eq. for velocity corrections

$$\begin{aligned} u'_e &= (p'_P - p'_E) d_e & \rightarrow & \quad u_e = u_e^* + (p'_P - p'_E) d_e \\ v'_n &= (p'_P - p'_N) d_n & \rightarrow & \quad v_n = v_n^* + (p'_P - p'_N) d_n \end{aligned} \quad \text{where} \quad d_e = \frac{A_e}{a_e} \quad d_n = \frac{A_n}{a_n}$$

- Substitute  $\mathbf{u}$  in continuity equation:  $(\rho u A)_e - (\rho u A)_w + (\rho v A)_n - (\rho v A)_s = 0$

$$\begin{aligned} &\rho_e [u_e^* + (p'_P - p'_E) d_e] A_e - \rho_w [u_w^* + (p'_W - p'_P) d_w] A_w \dots \\ &+ \rho_n [v_n^* + (p'_P - p'_N) d_n] A_n - \rho_s [v_s^* + (p'_S - p'_P) d_s] A_s = 0 \end{aligned}$$

$$\text{Equation for } p': \quad a_P p'_P = a_E p'_E + a_W p'_W + a_N p'_N + a_S p'_S + b$$

# Steady incompressible NS: the SIMPLE algorithm

2. Pressure correction equation:  $a_P p'_P = a_E p'_E + a_W p'_W + a_N p'_N + a_S p'_S + b$

■ Coefficients:  $a_E = \rho_e d_e A_e, \quad a_W = \rho_w d_w A_w, \quad a_N = \rho_n d_n A_n, \quad a_S = \rho_s d_s A_s$   
 $a_P = a_E + a_W + a_N + a_S$

■ Source term:  $b = (\rho u^* A)_w - (\rho u^* A)_e + (\rho v^* A)_s - (\rho v^* A)_n$

- Balance of convective fluxes, based on tentative velocities. (Discrete version of  $div(\mathbf{u}^*)$ .)
- Before convergence, continuity not satisfied, therefore  $b \neq 0 \Rightarrow p' \neq 0 \Rightarrow \mathbf{u}' \neq 0$ .
- Convergence when continuity satisfied:  $b = 0 \Rightarrow p' = 0, \mathbf{u}' = 0 \Rightarrow p = p^*, \mathbf{u} = \mathbf{u}^*$ .
- **$b$  is an indicator of convergence** (in addition to the residuals).
- The omission of neighbor contributions doesn't affect the final converged solution.

# Steady incompressible NS: the SIMPLE algorithm

## 3. Update and iterate:

- New value = old guess + correction:  $\mathbf{u} = \mathbf{u}^* + \mathbf{u}'$ ,  $p = p^* + p'$

- Generally, need under-relaxation to stabilize the procedure:

$$\mathbf{u}_{new} = \alpha_{\mathbf{u}}\mathbf{u} + (1 - \alpha_{\mathbf{u}})\mathbf{u}_{old}$$

$$p_{new} = \alpha_p p + (1 - \alpha_p)p^* = p^* + \alpha_p p'$$

- With this type of velocity under-relaxation, the discretized momentum eq. become

$$\left(\frac{a_e}{\alpha_u}\right) u_e^{(k+1)} = \sum a_f u_f^{(k+1)} + (p_P - p_E)A_e + \left(\frac{1 - \alpha_u}{\alpha_u}\right) a_e u_e^{(k)}$$

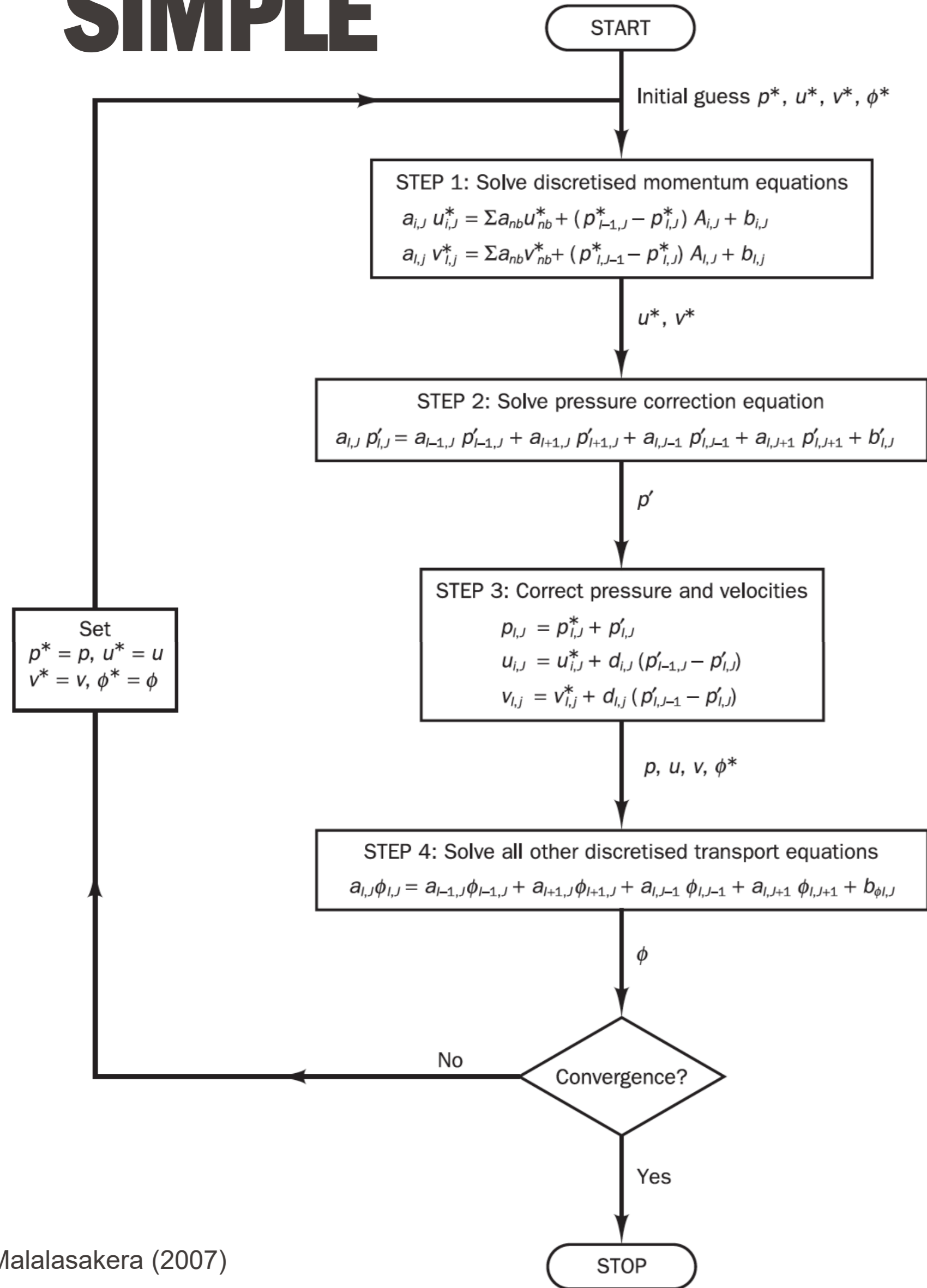
$$\left(\frac{a_n}{\alpha_v}\right) v_n^{(k+1)} = \sum a_f v_f^{(k+1)} + (p_P - p_N)A_n + \left(\frac{1 - \alpha_v}{\alpha_v}\right) a_n v_n^{(k)}$$

Analogous to pseudo-transient simulation, i.e. unsteady simulation with space-dependent time step (week 5).

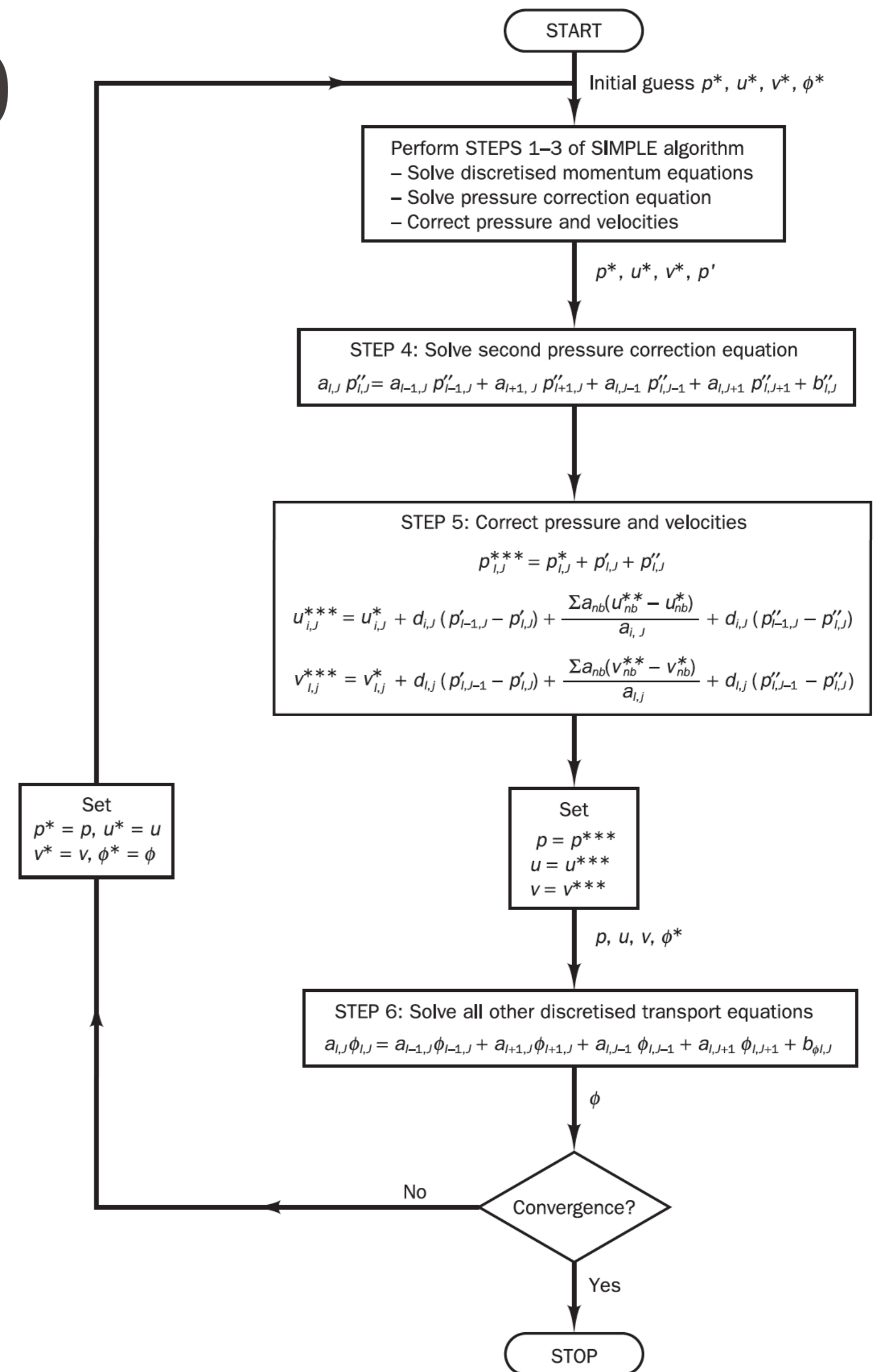
# The SIMPLE algorithm, and improved variants

- SIMPLE
  - Routinely implemented in CFD codes.
  - The pressure correction is satisfactory for correcting velocities, but not so good for correcting pressure itself. A correct velocity field does not yield a correct pressure.
  - Rather slow convergence due to crude approximation.
- SIMPLER
  - Velocity: same method as SIMPLE (velocity correction from pressure correction).
  - Pressure: not corrected, directly calculated from discretized continuity equation.
- SIMPLEC
  - Less crude approximation in the velocity correction eq. (omit less terms).
- PISO (“Pressure Implicit with Splitting of Operators”)
  - One additional correction step.
  - More expensive per iteration, but converges faster.

# SIMPLE



# PISO

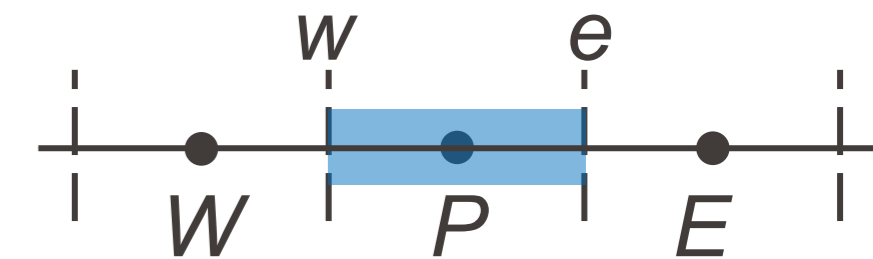


# Staggered grid vs. collocated grid

- Staggered grid: popular until a few decades ago. But several drawbacks: must store more geometric information; difficulties at boundaries.
- Alternative: stick to collocated grid (same grid for velocity and pressure); add higher-order pressure term to avoid checkerboard mode.

Discretize momentum eq. and continuity eq. on same grid:

$$a_P u_P = \sum a_C u_C + \frac{p_W - p_E}{2} A_P \quad a_E u_E = \sum a_C u_C + \frac{p_P - p_{EE}}{2} A_E \quad \sum (\rho \mathbf{u} \cdot \mathbf{n} A)_f = 0$$



If interpolate face velocities as usual,  $u_e = (u_P + u_E)/2$  doesn't contain the physically expected pressure difference  $p_E - p_P$  across face  $e \rightarrow$  pressure correction can be oscillatory (checkerboard).

- Example of alternative discretization: Rhie-Chow interpolation (expression on 1D uniform grid)

$$u_e = \underbrace{\frac{u_P + u_E}{2}}_{\text{Usual term}} + \underbrace{\frac{1}{2}(d_P + d_E)(p_P - p_E) - \frac{1}{4}d_P(p_W - p_E) - \frac{1}{4}d_E(p_P - p_{EE})}_{\text{Higher-order pressure term, with desired pressure difference}} \approx \frac{u_P + u_E}{2} + \frac{d}{4} \frac{\partial^3 p}{\partial x^3} \Big|_e (\Delta x)^3$$

(if all  $d$  values equal)

Additional term is small  $\rightarrow$  doesn't compromise solution accuracy.

Damping of spurious pressure oscillations  $\rightarrow$  also called "pressure dissipation".

# Summary and guidelines

- Momentum and continuity eq. are coupled. Usually solved with segregated (sequential) method. Coupled (simultaneous) method can be used too; better convergence rate but requires more memory.
- Risk of checkerboard pressure mode on collocated grid. Remedies: staggered grids for velocities, or higher-order pressure term (Rhie-Chow).
- Incompressible flows: no eq. for pressure. Most methods derive a Poisson eq. for pressure (taking the divergence of the momentum eq., and enforcing continuity).
  - Fractional step method: non-iterative, 2-step method (unsteady)
  - SIMPLE algorithm: iterative, 2-step method (steady or unsteady)
- Fluent: 1 coupled solver + 4 segregated solvers: SIMPLE, SIMPLEC, PISO, Fractional Step Method. Collocated grid.

# Appendix: pressure as a Lagrange multiplier

- Rigorous demonstration for the Stokes equations ( $Re=0$ ):  $\nabla \cdot \mathbf{u} = 0$ ,  $-\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{f} = \mathbf{0}$
- The solution of the Stokes equations satisfies a minimum dissipation problem MP:

$$\min_{\mathbf{u}} \int_V \left( \frac{1}{2} \mu (\nabla \mathbf{u})^2 - \mathbf{f} \cdot \mathbf{u} \right) dV \quad \text{such that} \quad \nabla \cdot \mathbf{u} = 0$$

- Transform this constrained optimization problem into an unconstrained one by introducing the Lagrangian

$$\mathcal{L}(\mathbf{u}, \lambda) = \int_V \left( \frac{1}{2} \mu (\nabla \mathbf{u})^2 - \mathbf{f} \cdot \mathbf{u} - \lambda (\nabla \cdot \mathbf{u}) \right) dV$$

where  $\lambda$  is a (yet unknown) Lagrange multiplier enforcing the incompressibility constraint.

- A solution of the optimization problem corresponds to a stationary point of the Lagrangian, i.e.  $\delta \mathcal{L} = 0$ , or:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{u}} = 0 \quad \text{and} \quad \frac{\partial \mathcal{L}}{\partial \lambda} = 0$$

# Appendix: pressure as a Lagrange multiplier

- Variation with respect to  $\lambda$ :  $\int_V -\delta\lambda(\nabla \cdot \mathbf{u}) dV = 0 \quad \forall \delta\lambda \quad \Rightarrow \quad \boxed{\nabla \cdot \mathbf{u} = 0}$

By construction, obtain the constraint.

- Variation with respect to  $\mathbf{u}$ :  
$$\int_V (\mu \nabla \mathbf{u} : \nabla \delta \mathbf{u} - \mathbf{f} \cdot \delta \mathbf{u} - \lambda (\nabla \cdot \delta \mathbf{u})) dV = 0 \quad \forall \delta \mathbf{u}$$
$$\Rightarrow \int_V (-\mu \nabla^2 \mathbf{u} \cdot \delta \mathbf{u} - \mathbf{f} \cdot \delta \mathbf{u} + \nabla \lambda \cdot \delta \mathbf{u}) dV = 0 \quad \forall \delta \mathbf{u}$$
$$\Rightarrow \int_V (-\mu \nabla^2 \mathbf{u} - \mathbf{f} + \nabla \lambda) \cdot \delta \mathbf{u} dV = 0 \quad \forall \delta \mathbf{u}$$
$$\Rightarrow \boxed{\mu \nabla^2 \mathbf{u} + \mathbf{f} - \nabla \lambda = \mathbf{0}}$$

Therefore, the velocity field and the Lagrange multiplier that are solution of the minimization problem MP are also solution of the Stokes equations.

Additionally, one recognizes that the pressure  $p$  is the Lagrange multiplier  $\lambda$  enforcing the incompressibility constraint.