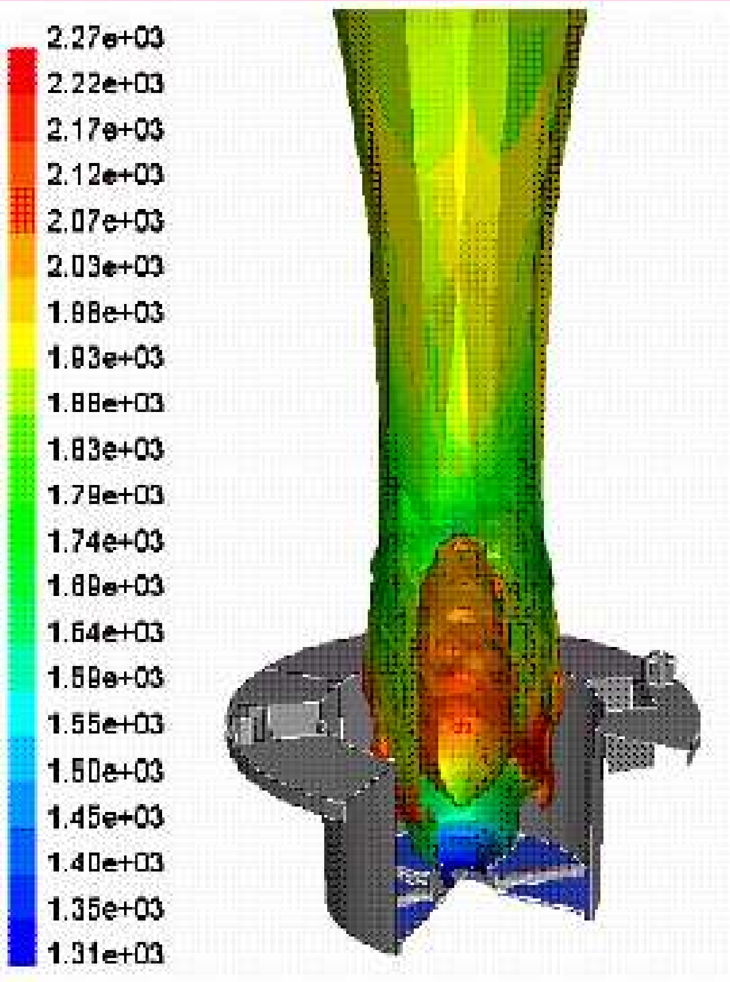
CFD7 # Computer Fluid Dynamics

2181106

E181107

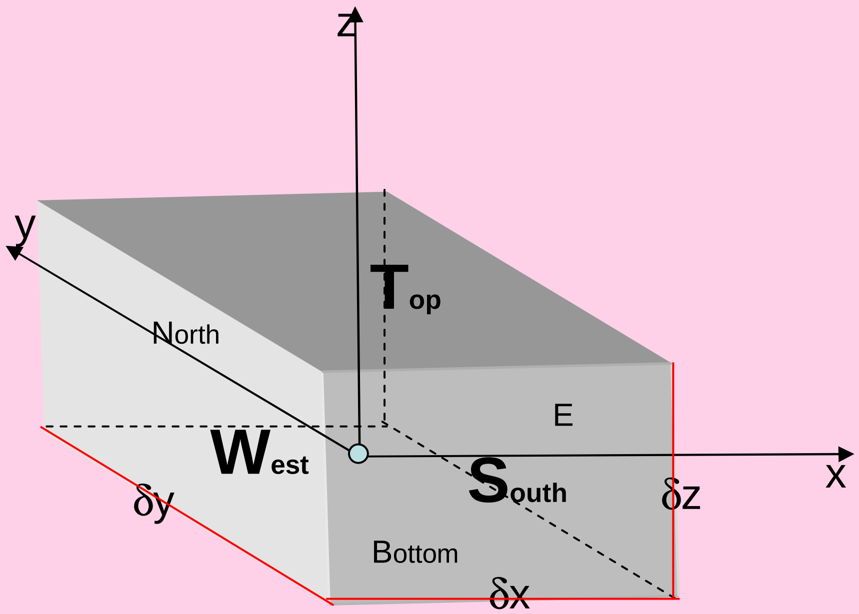


Solvers, schemes SIMPLEx, upwind,…

Remark: foils with „black background“ could be skipped, they are aimed to the more advanced courses

Rudolf Žitný, Ústav procesní a zpracovatelské techniky ČVUT FS 2010

CFD7 FINITE VOLUME METHOD



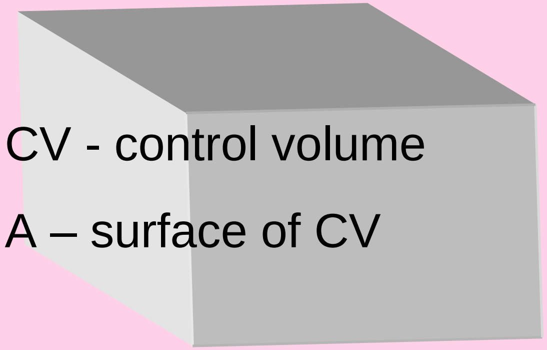
FINITE CONTROL VOLUME δx = Fluid element dx

Finite size

Infinitely small size

CFD7 FVM diffusion problem 1D

$$\int_{CV} [\nabla \bullet (\Gamma \nabla \Phi) + S_\Phi] dV = 0$$



CV - control volume

A – surface of CV

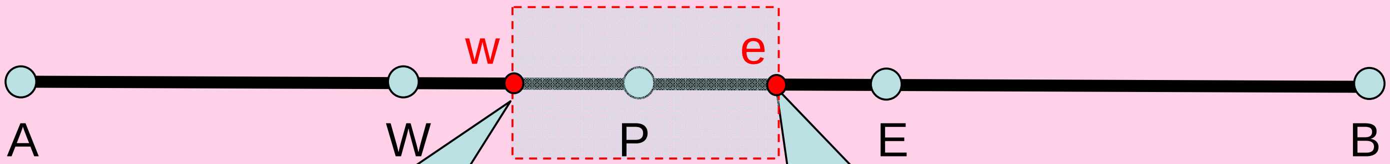
Gauss theorem

$$\int_{A} \vec{n} \bullet (\Gamma \nabla \Phi) dA + \int_{CV} S_\Phi dV = 0$$

Diffusive flux

1D case

$$\frac{d}{dx}\left(\Gamma \frac{d\Phi}{dx}\right) + S = 0 \qquad \Phi(A) = \Phi_A \quad \Phi(B) = \Phi_B$$



$$\Gamma_w = \frac{\Gamma_P + \Gamma_W}{2} \qquad \frac{d\Phi}{dx}\Big|_w = \frac{\Phi_P - \Phi_W}{\delta x_{WP}}$$

$$\Gamma_e = \frac{\Gamma_P + \Gamma_E}{2} \qquad \frac{d\Phi}{dx}\Big|_e = \frac{\Phi_E - \Phi_P}{\delta x_{PE}}$$

CFD7 FVM diffusion problem 1D

$$\int_A \vec{n}\bullet(\Gamma\nabla\Phi)\,dA + \int_{CV} S_\Phi\, dV = 0$$

Overall flux from the west side

$$\left(\Gamma A\frac{d\Phi}{dx}\right)_e - \left(\Gamma A\frac{d\Phi}{dx}\right)_w + \bar{S}\Delta V = 0$$

Linearization of source term

$$\Gamma_e A_e \frac{\Phi_E-\Phi_P}{\delta x_{PE}} - \Gamma_w A_w \frac{\Phi_P-\Phi_W}{\delta x_{WP}} + S_U + S_P\Phi_P = 0$$

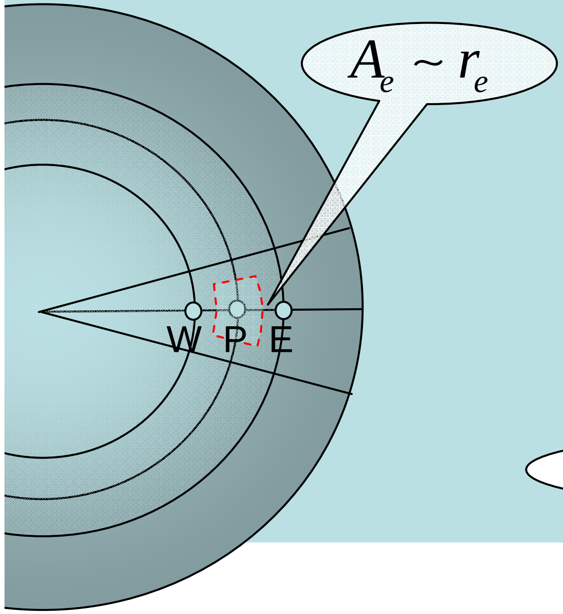
$$\left(\frac{\Gamma_e A_e}{\delta x_{PE}} + \frac{\Gamma_w A_w}{\delta x_{WP}} - S_P\right)\Phi_P = \left(\frac{\Gamma_w A_w}{\delta x_{WP}}\right)\Phi_W + \left(\frac{\Gamma_e A_e}{\delta x_{PE}}\right)\Phi_E + S_U$$

a_P *a_W* *a_E*

Check properties

- $a_P = a_W + a_E$ for S=0 (without sources)
- All coefficients are positive

CFD7 Tutorial – sphere drying



Heat transfer from air at temperature T_o

Enthalpy of evaporation

$$\frac{1}{r^2}\frac{d}{dr}(r^2 k \frac{dT}{dr}) = 0 \qquad \frac{dT}{dr}|_{r=0} = 0 \qquad k\frac{dT}{dr}|_{r=R} = \alpha(T_o - T(R)) + \Delta h D \frac{d\omega_w}{dr}$$

$$\frac{1}{r^2}\frac{d}{dr}(r^2 D \frac{d\omega_w}{dr}) = 0 \qquad \frac{d\omega_w}{dr}|_{r=0} = 0 \qquad k\frac{d\omega_w}{dr}|_{r=R} = \beta(\omega_e - \omega_w(R))$$

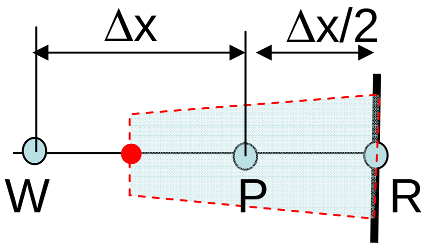
Diffusion coefficient

Mass transfer coefficient

Internal control volumes

$$r_e k_e \frac{T_E - T_P}{\delta x_{PE}} - r_w k_w \frac{T_P - T_W}{\delta x_{WP}} = 0$$

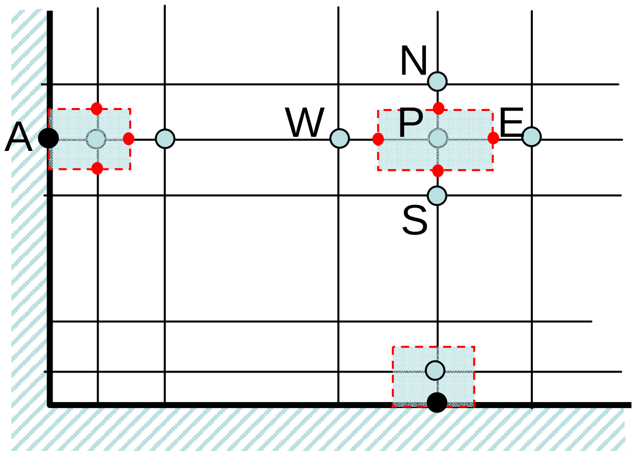
T_R temperature at surface of sphere must be extrapolated (using T_W and T_P)



$$-R\alpha(T_o - T_R) - r_w k_w \frac{T_P - T_W}{\delta x_{WP}} = 0$$

CFD7 FVM diffusion problem 2D

$$\frac{\partial}{\partial x}(\Gamma\frac{\partial \Phi}{\partial x})+\frac{\partial}{\partial y}(\Gamma\frac{\partial \Phi}{\partial y})+S=0$$



$$a_P\Phi_P = a_W\Phi_W + a_E\Phi_E + a_N\Phi_N + a_S\Phi_S + S_U$$

$$a_W = \frac{\Gamma_w A_w}{\delta x_{WP}}, \quad a_E = \frac{\Gamma_e A_e}{\delta x_{PE}}, \ldots.$$

$$a_P = a_W + a_E + a_N + a_S - S_P$$

Boundary conditions

- First kind (Dirichlet BC) $\Phi(A) = \Phi_A$
- Second kind (Neumann BC) $\frac{\partial \Phi}{\partial x}|_A = q_A$
- Third kind (Newton's BC) $k\frac{\partial \Phi}{\partial x}|_A = \alpha(\Phi(A) - \Phi_e)$

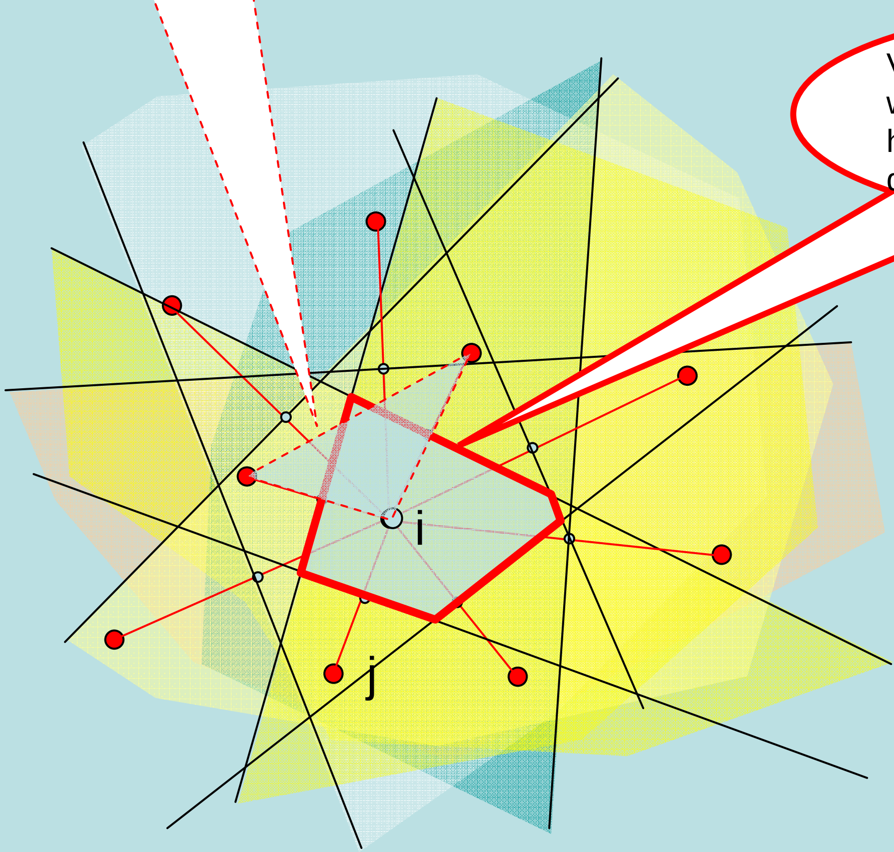
Example: insulation q=0

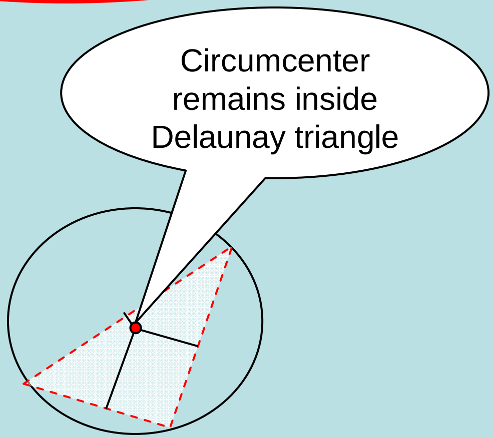
Example: Finite thermal resistance (heat transfer coefficient α)

CFD7 FVM unstructural mesh generation

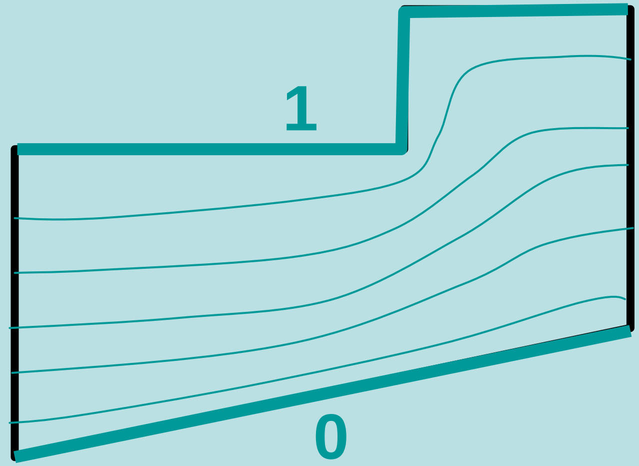
Voronoi polygons → Finite Volume Method

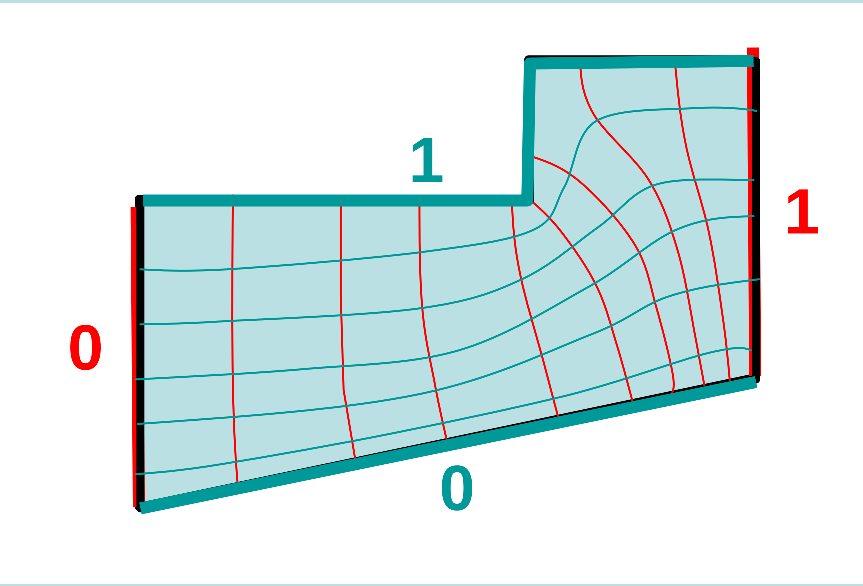
Delaunay triangulation → Finite Element Method

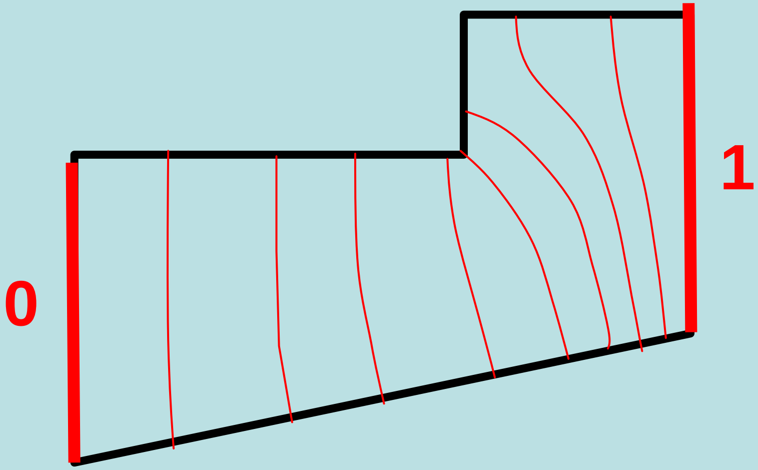




CFD7 FVM structural mesh generation







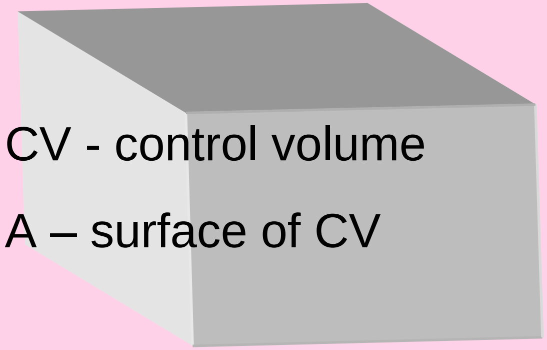
Laplace equation solvers

$$\nabla \bullet \nabla X = 0$$

$$\nabla \bullet \nabla Y = 0$$

CFD7 FVM convection diffusion 1D

$$\int_{CV} \nabla \bullet (\rho \vec{u} \Phi) dV = \int_{CV} [\nabla \bullet (\Gamma \nabla \Phi) + S_{\Phi}] dV$$



CV - control volume

A – surface of CV

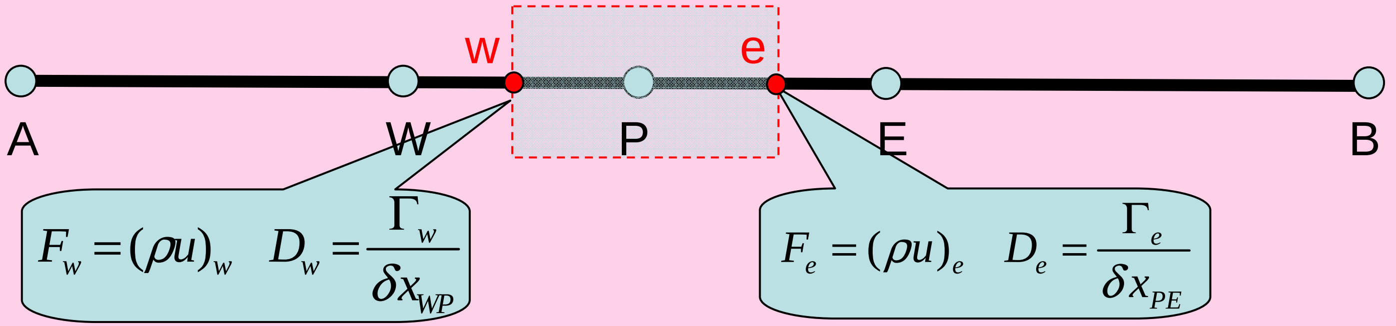
Gauss theorem

$$\int_{A} \vec{n} \bullet (\rho \vec{u} \Phi) dA = \int_{A} \vec{n} \bullet (\Gamma \nabla \Phi) dA + \int_{CV} S_{\Phi} dV$$

1D case

$$\frac{d}{dx}(\rho u \Phi) = \frac{d}{dx}\left(\Gamma \frac{d\Phi}{dx}\right) + S$$

$$F_e \Phi_e - F_w \Phi_w = D_e(\Phi_E - \Phi_P) - D_w(\Phi_P - \Phi_W) + S_U + S_P \Phi_P$$



CFD7 FVM convection diffusion 1D

F-mass flux (through faces A) $F_w = (\rho u)_w \quad F_e = (\rho u)_e$

Continuity equation (mass flowrate conservation) $F_e=F_w$

D-diffusion conductance $D_w = \frac{\Gamma_w}{\delta x_{WP}} \quad D_e = \frac{\Gamma_e}{\delta x_{PE}}$

Relative importance of convective transport is characterised by Peclet number of cell (of control volume, because Pe depends upon size of cell)

$$Pe = \mathrm{Re} = \frac{\rho u \delta x}{\mu} = \frac{F}{D} \qquad \Gamma = \mu \; [Pa.s = \frac{kg}{m \cdot s}]$$

Prandtl

Thermal conductivity

$$Pe = \frac{\rho u \delta x}{k / c_p} = \mathrm{Re}\,\mathrm{Pr} \qquad \Gamma = \frac{k}{c_p} \; [\frac{W}{m \cdot K} \frac{kg \cdot K}{J} = \frac{kg}{m \cdot s}]$$

Schmidt

Binary diffusion coef

$$Pe_m = \frac{\rho u \delta x}{\rho D_{AB}} = \mathrm{Re}\,Sc \qquad \Gamma = \rho D_{AB} \; [\frac{kg}{m^3} \frac{m^2}{s} = \frac{kg}{m \cdot s}]$$

$$Pe = \frac{F}{D}$$

CFD7

FVM convection diffusion 1D

Methods differ in the way how the unknown transported values at the control volume faces (Φ_w , Φ_e) are calculated (different interpolation techniques)

Result can be always expressed in the form

$$a_P \Phi_P = \sum_{neighbours} a_{nb} \Phi_{nb}$$

Properties of resulting schemes should be evaluated

- Conservativeness
- Boundedness (positivity of coefficients)
- Transportivity (schemes should depend upon the Peclet number of cell)

CFD7

FVM central scheme 1D

$$\Phi_e = \frac{1}{2}(\Phi_P + \Phi_E) \qquad \Phi_w = \frac{1}{2}(\Phi_P + \Phi_W)$$

$$a_P \Phi_P = a_W \Phi_W + a_E \Phi_E$$

$$a_W = D_w + \frac{F_w}{2} \qquad a_E = D_e - \frac{F_e}{2}$$

- Conservativeness (yes)
- Boundedness (positivity of coefficients)
- Transportivness (no)

$$Pe_e = \frac{|F_e|}{D_e} < 2$$

Very fine mesh is necessary

CFD7 FVM upwind 1st order 1D

$\Phi_e = \Phi_P$ for flow direction to right $F_e > 0$ else $\Phi_e = \Phi_E$

$\Phi_w = \Phi_P$ for $F_w < 0$ else $\Phi_w = \Phi_W$

$$a_P \Phi_P = a_W \Phi_W + a_E \Phi_E$$

$$a_W = D_w + \max\left(F_w, 0\right) \qquad a_E = D_e + \max\left(-F_e, 0\right)$$

- Conservativeness (yes)
- Boundedness (positivity of coefficients) YES for any Pe
- Transportivness (YES)

But at a prize of decreased order of accuracy (only 1st order!!)

CFD7

FVM hybrid upwind Spalding 1972 1D

- Central scheme for Pe<2
- Upwind with diffusion supressed for Pe>2

$$a_P \Phi_P = a_W \Phi_W + a_E \Phi_E$$

$$a_W = \max\left(F_w, D_w + \frac{F_w}{2}, 0 \right) \qquad a_E = \max\left(-F_e, D_e - \frac{F_e}{2}, 0 \right)$$

- Conservativeness (yes)
- Boundedness (positivity of coefficients) YES for any Pe
- Transportivness (YES)

But at a prize of decreased order of accuracy at high Pe

CFD7 FVM power law Patankar 1980 1D

- Polynomial flow for Pe<10
- Upwind with zero diffusion for Pe>10

$$a_P \Phi_P = a_W \Phi_W + a_E \Phi_E$$

$$a_W = D_w \max\left((1 - \frac{Pe_w}{10})^5, 0 \right) + \max(F_w, 0)$$

$$a_E = D_e \max\left((1 - \frac{Pe_e}{10})^5, 0 \right) + \max(-F_e, 0)$$

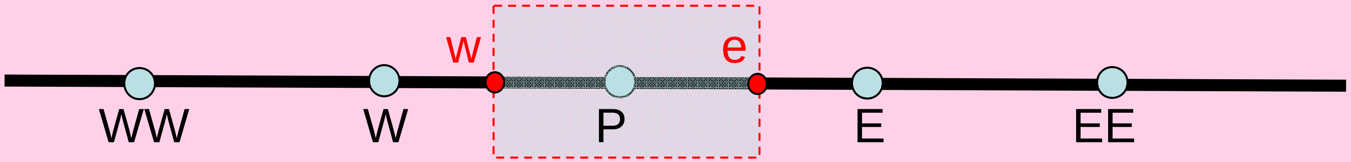
$$(1 - \frac{Pe}{10})^5 \cong 1 - \frac{Pe}{2}$$

- Conservativeness (yes)
- Boundedness (positivity of coefficients) YES for any Pe
- Transportivness (YES)

CFD7

FVM QUICK Leonard 1979 1D

Quadratic Upwind Interpolation – more nodal points necessary



$$a_P\Phi_P = a_W\Phi_W + a_E\Phi_E + a_{WW}\Phi_{WW} + a_{EE}\Phi_{EE}$$

$$a_W = \frac{F_e}{8}\alpha_e + \frac{3F_w}{8}(\alpha_w + 1) + D_w \qquad a_E = -\frac{F_w}{8}(1-\alpha_w) - \frac{3F_w}{8}(2-\alpha_e) + D_e$$

$$a_{WW} = -\frac{F_w}{8}\alpha_w \qquad a_{EE} = \frac{F_e}{8}(1-\alpha_e)$$

- **Third order of accuracy** **very low numerical diffusion comparing with other schemes**
- Scheme is not bounded (stability problems are frequently encountered)

α_e=1 for F_e>0 else α_e=0. α_w=1 for F_w>0 else α_w=0.

CFD7 FVM exponential Patankar 1980 1D

➢Exact solution for constant velocity and diffusion coefficient Γ

$$a_P \Phi_P = a_W \Phi_W + a_E \Phi_E$$

$$Pe = \frac{F}{D}$$

$$a_W = \frac{e^{Pe/2}}{e^{Pe/2} + e^{-Pe/2}} \qquad a_E = \frac{e^{-Pe/2}}{e^{Pe/2} + e^{-Pe/2}}$$

➢Conservativeness (yes)

➢Boundedness (positivity of coefficients) YES for any Pe

➢Transportivness (YES)

CFD7

FVM exponential proof

- Exact solution for constant velocity and diffusion coefficient Γ

Assume constant mass flux F and constant Pe=F/D.

$$\frac{\partial \Phi F}{\partial x} = \frac{\partial}{\partial x}\left(\Gamma \frac{\partial \Phi}{\partial x}\right) \rightarrow \Phi = \frac{\Gamma}{F}\frac{\partial \Phi}{\partial x} + c_1 \rightarrow \Phi = \frac{\delta x}{Pe}\frac{\partial \Phi}{\partial x} + c_1 \rightarrow \Phi = c_2 e^{Pe\frac{x}{\delta x}} + c_1$$

Analytical solution

$$\Phi_E = c_2 e^{Pe} + c_1$$

$$\Phi_W = c_2 e^{-Pe} + c_1$$

Boundary conditions

$$c_1 = \frac{\Phi_E e^{-Pe} - \Phi_W e^{Pe}}{e^{-Pe} - e^{Pe}}$$

$$c_2 = \frac{\Phi_E - \Phi_W}{e^{Pe} - e^{-Pe}}$$

Identified constants

$$\Phi_P = \frac{\Phi_E - \Phi_W}{e^{Pe} - e^{-Pe}} + \frac{\Phi_E e^{-Pe} - \Phi_W e^{Pe}}{e^{-Pe} - e^{Pe}} = \frac{\Phi_E (1 - e^{-Pe}) + \Phi_W (1 - e^{Pe})}{e^{Pe} - e^{-Pe}}$$

Exact solution at central point

$$a_E = \frac{1 - e^{-Pe}}{e^{Pe} - e^{-Pe}} = \frac{e^{-Pe/2}(e^{Pe/2} - e^{-Pe/2})}{(e^{Pe/2} - e^{-Pe/2})(e^{Pe/2} + e^{-Pe/2})} = \frac{e^{-Pe/2}}{e^{Pe/2} + e^{-Pe/2}}$$

$$a_W = \frac{e^{Pe/2}}{e^{Pe/2} + e^{-Pe/2}}$$

CFD7 FVM central scheme with modified viscosity

Almost any scheme (with the exception of QUICK) can be converted to central scheme introducing modified transport coefficient Γ

$$a_P\Phi_P = a_W\Phi_W + a_E\Phi_E$$

$$a_W = D_w^* + \frac{F_w}{2} \qquad a_E = D_e^* - \frac{F_e}{2}$$

Modified diffusion conductances are

$$D_w^* = \frac{\Gamma_w^*}{\delta x_{WP}} \qquad D_e^* = \frac{\Gamma_e^*}{\delta x_{PE}}$$

and

- $$\frac{\Gamma^*}{\Gamma} = 1$$ Central scheme

$$\frac{\Gamma^*}{\Gamma} = 1 + \frac{|Pe|}{2}$$ Upwind scheme

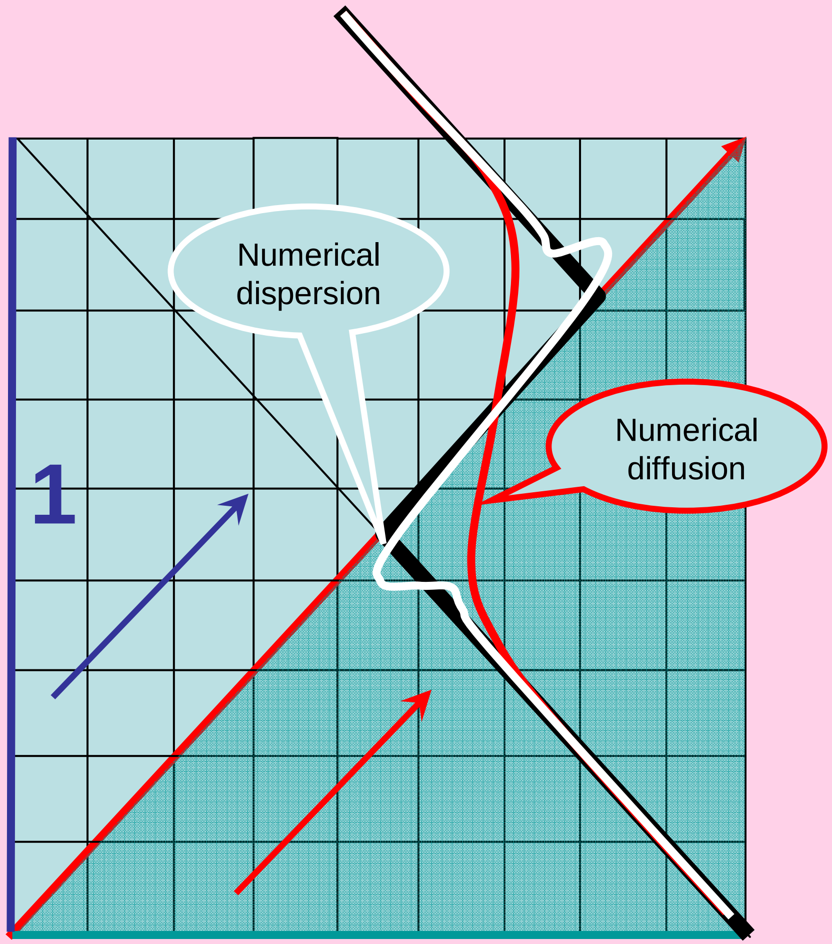
$$\frac{\Gamma^*}{\Gamma} = \frac{Pe/2}{\tanh Pe/2}$$ Exponential scheme

CFD7 FVM dispersion / diffusion

There are two basic errors caused by approximation of convective terms

➢**False diffusion** (or numerical diffusion) – artificial smearing of jumps, discontinuities. Neglected terms in the Taylor series expansion of first derivatives (convective terms) represent in fact the terms with second derivatives (diffusion terms). So when using low order formula for convection terms (for example upwind of the first order), their inaccuracy is manifested by the artificial increase of diffusive terms (e.g. by increase of viscosity). The false diffusion effect is important first of all in the case when the flow direction is not aligned with mesh, see example

➢**Dispersion** (or aliasing) – causing overshoots, artificial oscillations. Dispersion means that different components of Fourier expansion (of numerical solution) move with different velocities, for example shorter wavelengths move slower than the velocity of flow.



0 - boundary condition, all values are zero in the right triangle (without diffusion)

CFD7 FVM Navier Stokes equations

Example: Steady state and 2D (velocities **u,v** and pressure **p** are unknown functions)

This is equation for **u**

$$\frac{\partial \rho u^2}{\partial x}+\frac{\partial \rho uv}{\partial y}=-\frac{\partial p}{\partial x}+\frac{\partial}{\partial x}(\mu\frac{\partial u}{\partial x})+\frac{\partial}{\partial y}(\mu\frac{\partial u}{\partial y})+S_u$$

This is equation for **v**

$$\frac{\partial \rho uv}{\partial x}+\frac{\partial \rho v^2}{\partial y}=-\frac{\partial p}{\partial y}+\frac{\partial}{\partial x}(\mu\frac{\partial v}{\partial x})+\frac{\partial}{\partial y}(\mu\frac{\partial v}{\partial y})+S_v$$

$$\frac{\partial \rho u}{\partial x}+\frac{\partial \rho v}{\partial y}=0$$

This is equation for **p?**

But pressure p is not in the continuity equation!

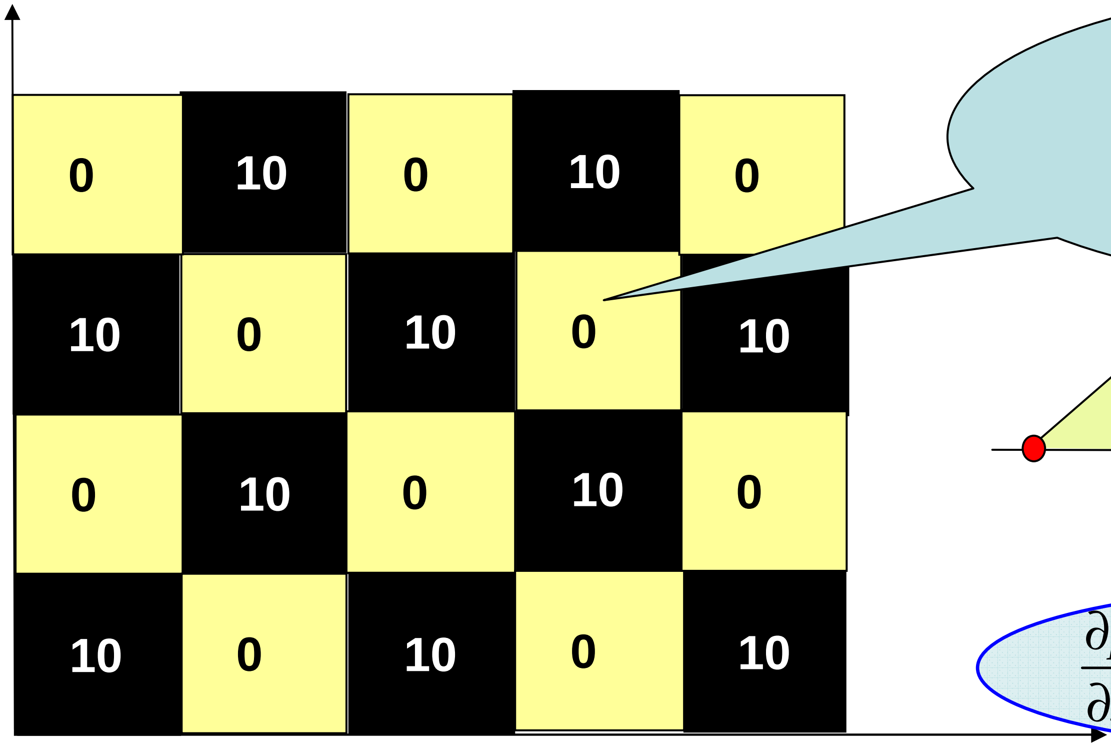
Solution of this problem is in SIMPLE methods, described later

CFD7

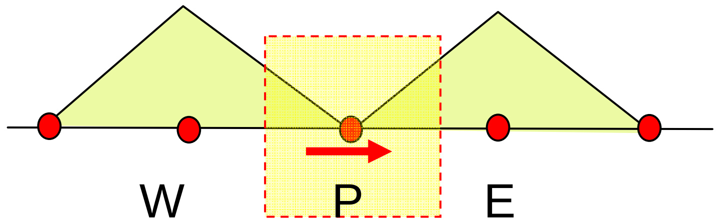
FVM checkerboard pattern

Other problem is called checkerboard pattern

This nonuniform distribution of pressure has no effect upon NS equations



The problem appears as soon as the u,v,p values are concentrated in the same control volume

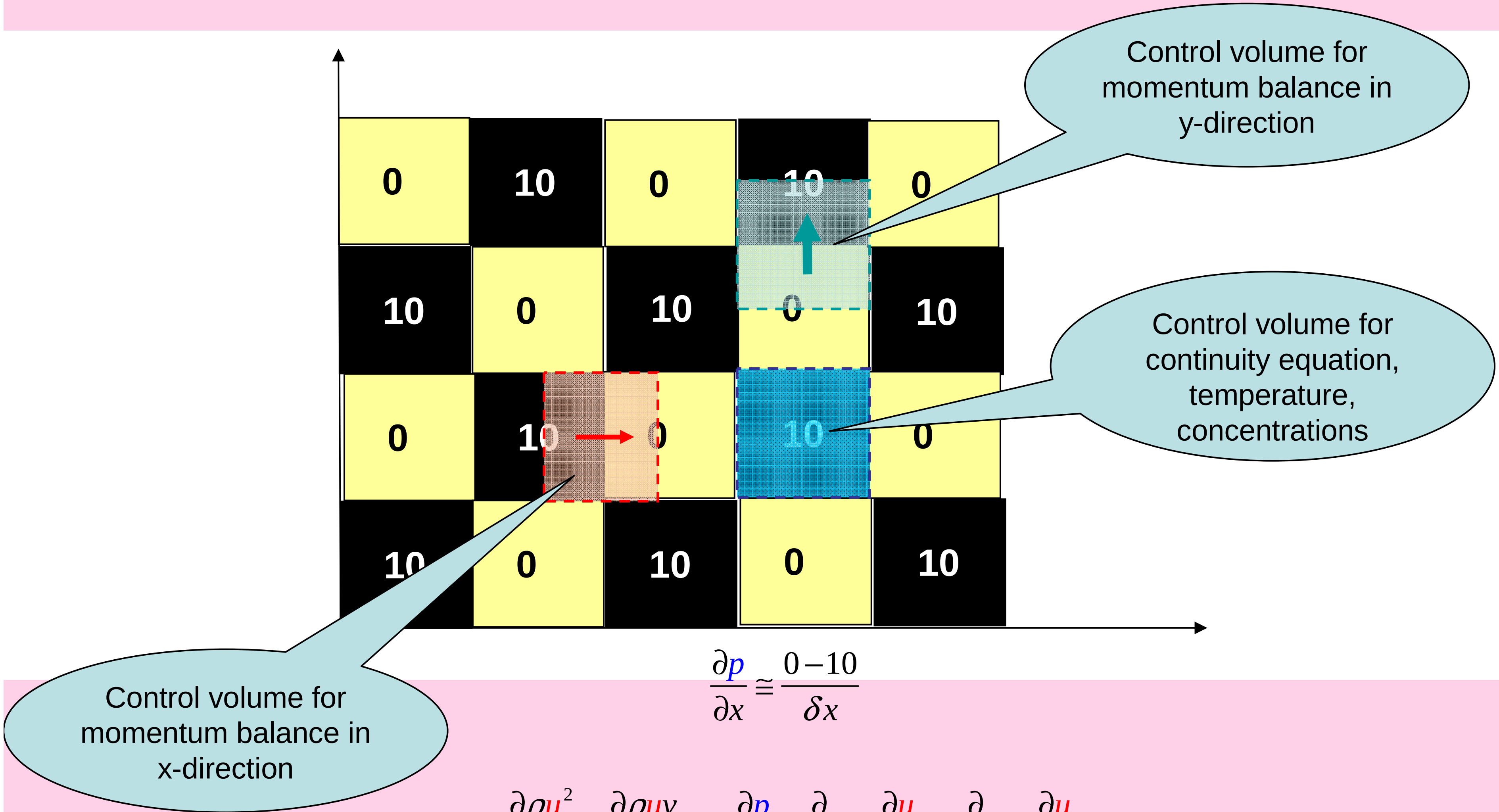


$$\frac{\partial p}{\partial x}\Big|_P \cong \frac{p_E - p_W}{2\delta x} = \frac{10-10}{2\delta x} = 0$$

$$\frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho u v}{\partial y} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\left(\mu \frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu \frac{\partial u}{\partial y}\right) + S_u$$

CFD7 FVM staggered grid

Different control volumes for different equations



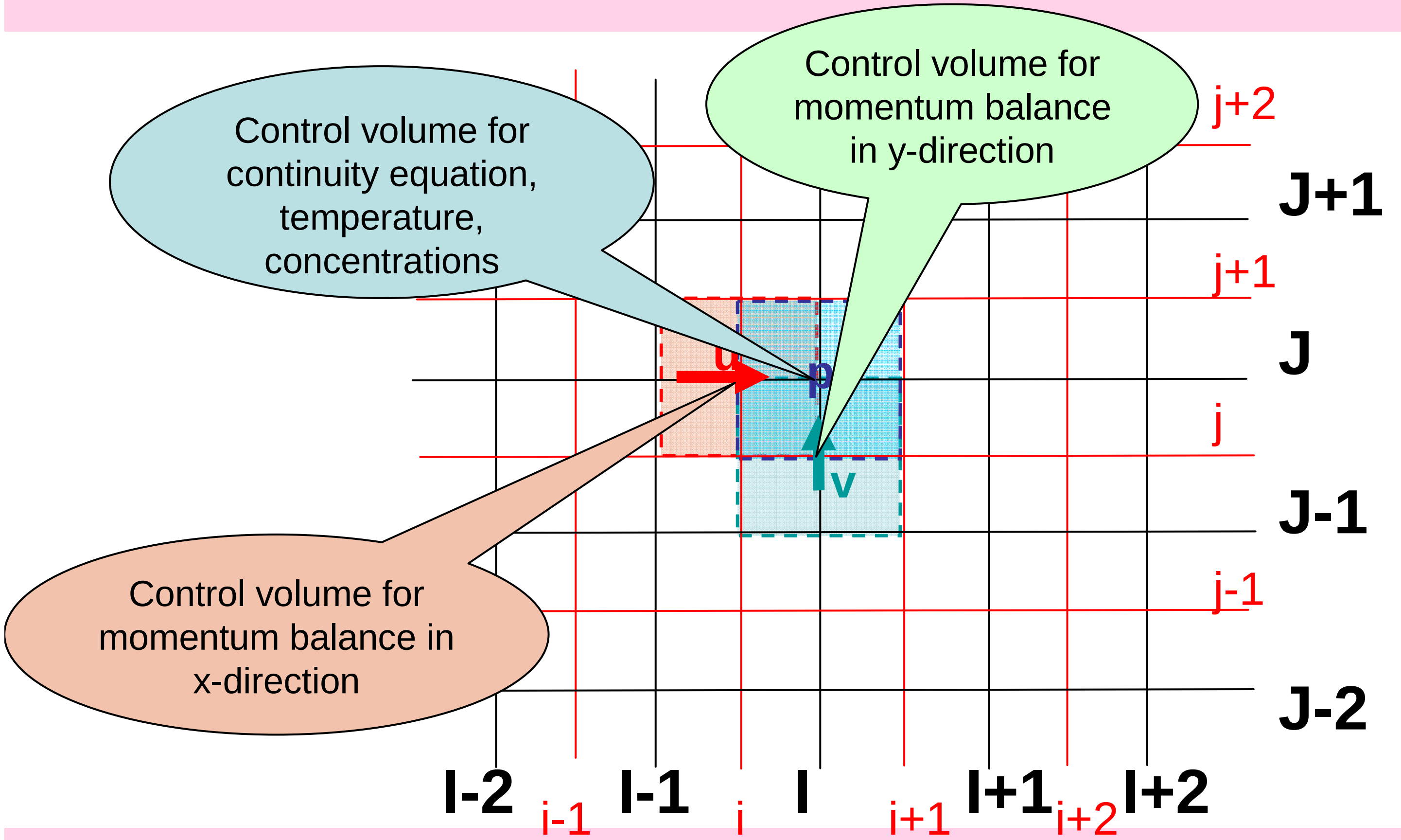
$$\frac{\partial p}{\partial x} \cong \frac{0-10}{\delta x}$$

$$\frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho u v}{\partial y} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}(\mu \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(\mu \frac{\partial u}{\partial y}) + S_u$$

CFD7

FVM staggered grid

Location of velocities and pressure in staggered grid



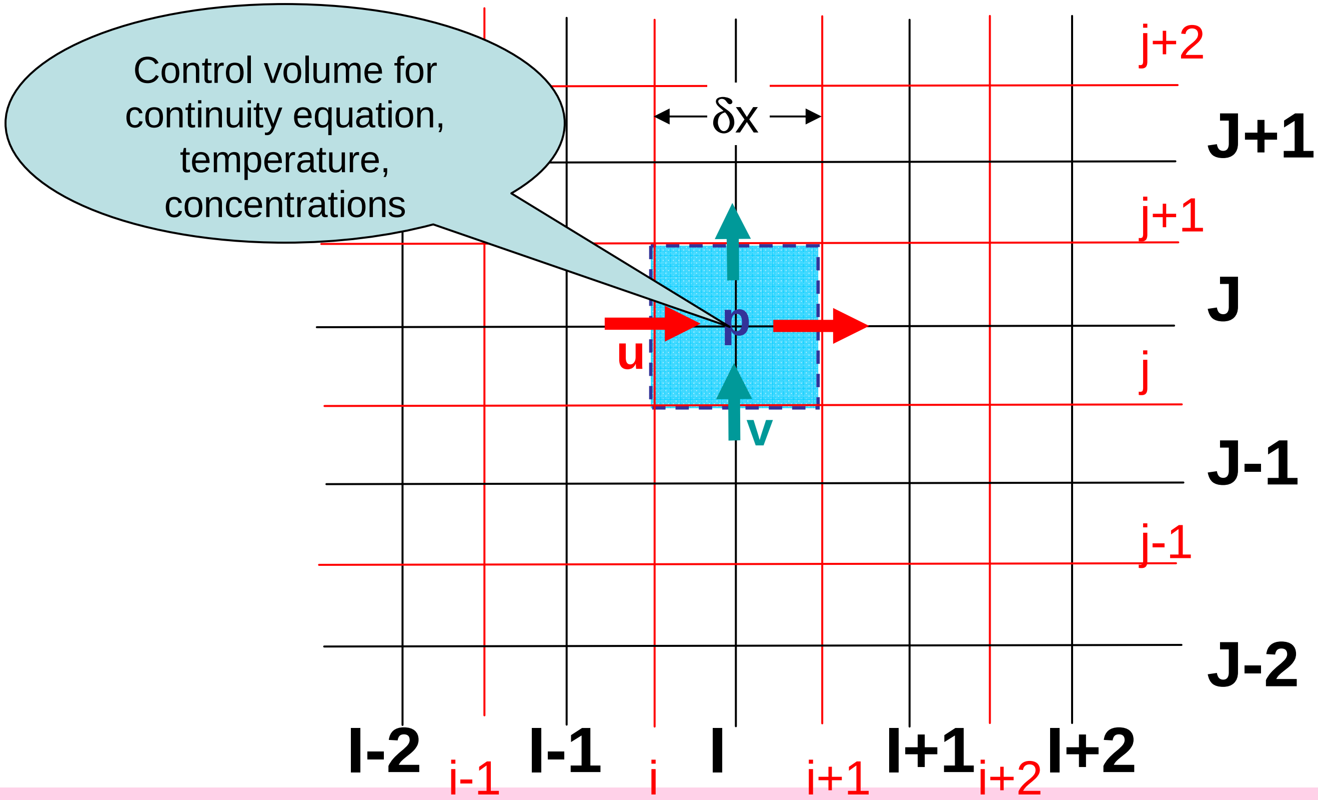
Pressure I,J

U i,J

V I,j

CFD7 FVM continuity equation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$



Pressure I,J

U i,J

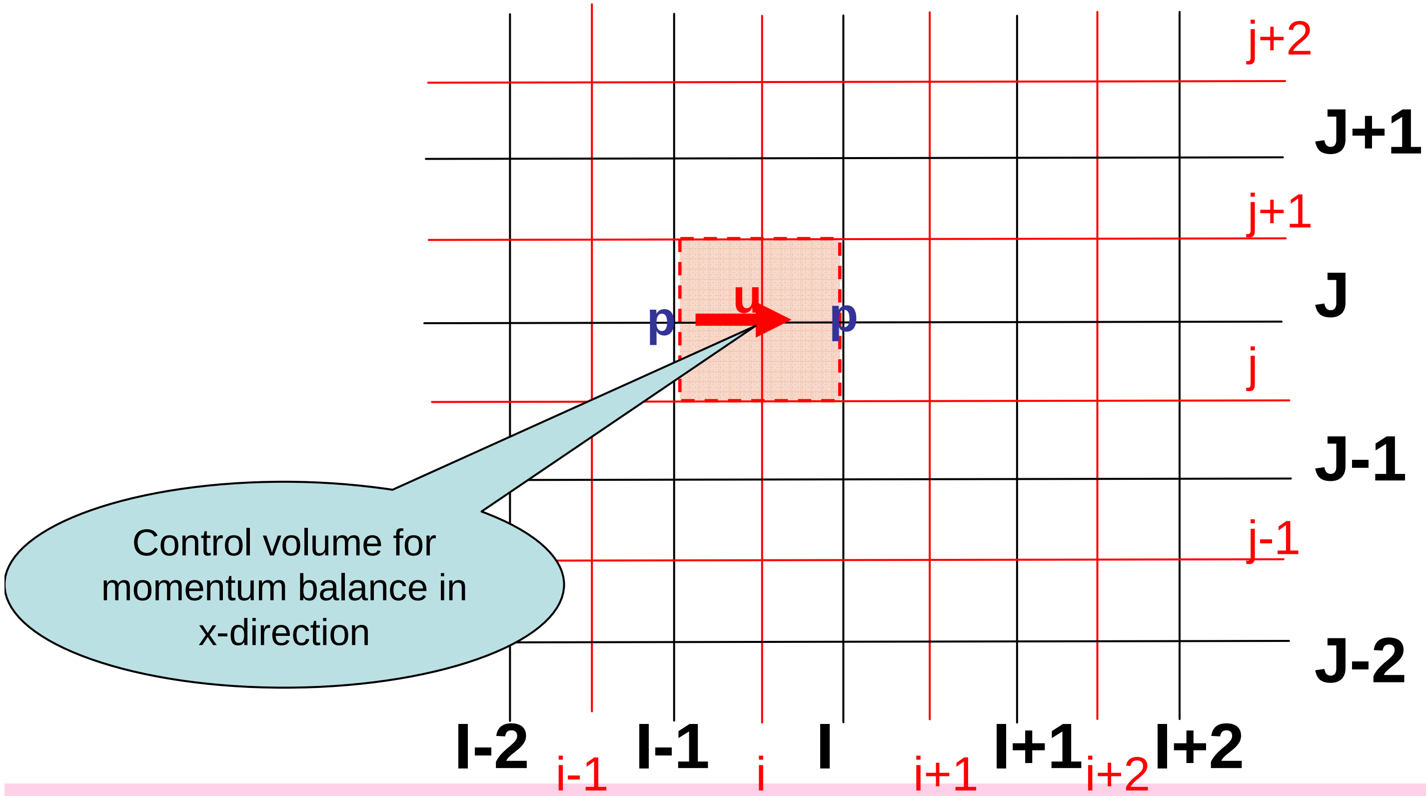
V I,j

$$\frac{u_{i+1,J} - u_{i,J}}{\delta x} + \frac{v_{I,j+1} - v_{I,j}}{\delta y} = 0$$

CFD7

FVM momentum x

$$\frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho u v}{\partial y} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}(\mu \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(\mu \frac{\partial u}{\partial y}) + S_u$$



Pressure I,J

U i,J

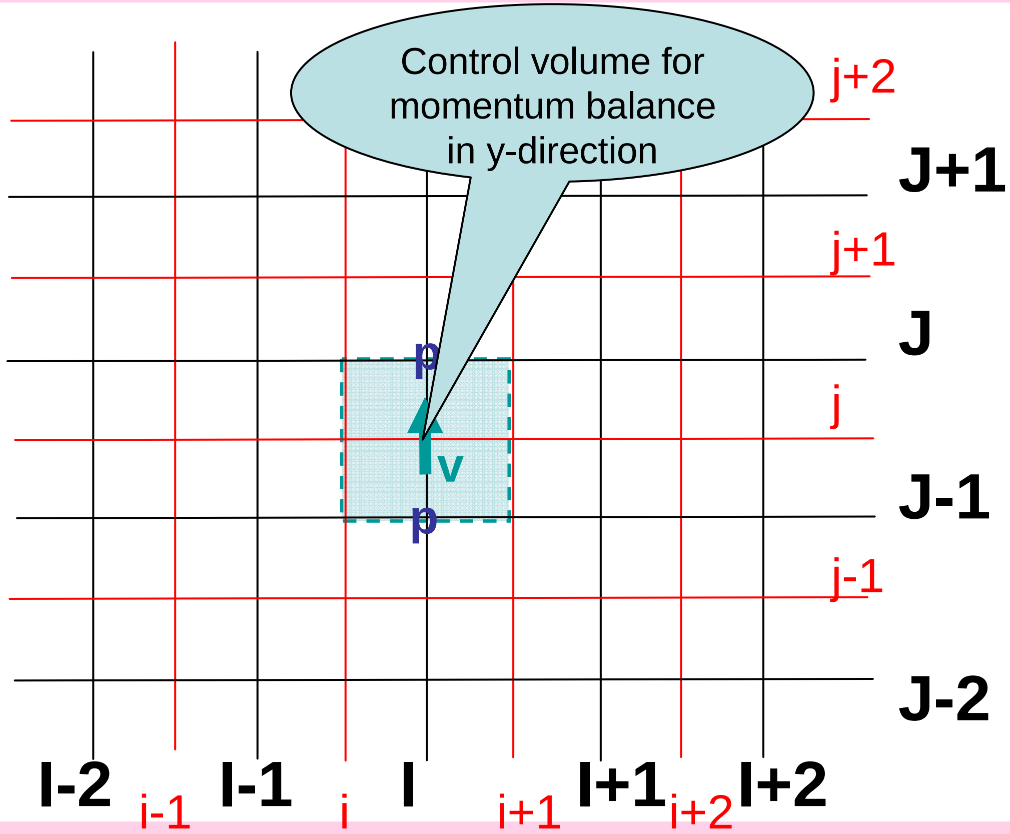
V I,j

$$a_{i,J} u_{i,J} = \sum_{neighbours} a_{nb} u_{nb} + A_{i,J} (p_{I-1,J} - p_{I,J}) + b_{iJ}$$

CFD7

FVM momentum y

$$\frac{\partial \rho u v}{\partial x}+\frac{\partial \rho v v}{\partial y}=-\frac{\partial p}{\partial y}+\frac{\partial}{\partial x}(\mu\frac{\partial v}{\partial x})+\frac{\partial}{\partial y}(\mu\frac{\partial v}{\partial y})+S_v$$



Pressure	I,J
U	i,J
V	I,j

$$a_{I,j}v_{I,j}=\sum_{neighbours} a_{nb}v_{nb}+A_{I,j}(p_{I,J-1}-p_{I,J})+b_{I,j}$$

CFD7 FVM SIMPLE step 1 (velocities)

Input: Approximation of pressure p*

$$a_{i,J}u^*_{i,J} = \sum_{neighbours} a_{nb}u^*_{nb} + A_{i,J}(p^*_{I-1,J} - p^*_{I,J}) + b_{iJ}$$

$$a_{I,j}v^*_{I,j} = \sum_{neighbours} a_{nb}v^*_{nb} + A_{I,j}(p^*_{I,J-1} - p^*_{I,J}) + b_{I,j}$$

CFD7 FVM SIMPLE step 2 (pressure correction)

$$p = p^* + p' \qquad u = u^* + u' \qquad v = v^* + v'$$

$$a_{i,J} u_{i,J} = \sum_{neighbours} a_{nb} u_{nb} + A_{i,J}(p_{I-1,J} - p_{I,J}) + b_{iJ}$$

"true" solution

$$a_{i,J} u^*_{i,J} = \sum_{neighbours} a_{nb} u^*_{nb} + A_{i,J}(p^*_{I-1,J} - p^*_{I,J}) + b_{iJ}$$

subtract

$$a_{i,J} u'_{i,J} = \sum_{neighbours} a_{nb} u'_{nb} + A_{i,J}(p'_{I-1,J} - p'_{I,J})$$

Neglect!!!

$$u'_{i,J} = d_{i,J}(p'_{I-1,J} - p'_{I,J}) \qquad v'_{I,j} = d_{I,j}(p'_{I,J-1} - p'_{I,J})$$

Substitute *u*+u'* and *v*+v'* into continuity equation

$$a_{IJ} p'_{IJ} = \sum_{neighbours} a_{nb} p'_{nb} + b'_{IJ}$$

Solve equations for pressure corrections

CFD7 FVM SIMPLE step 3 (update p,u,v)

Only these new pressures are necessary for next iteration

$$p_{I,J} \leftarrow p^*_{I,J} + p'_{I,J}$$

$$u_{i,J} \leftarrow u^*_{i,J} + d_{i,J}(p'_{I-1,J} - p'_{I,J})$$

$$v_{I,j} \leftarrow v^*_{I,j} + d_{I,j}(p'_{I,J-1} - p'_{I,J})$$

Continue with the improved pressures to the step 1

CFD7 FVM SIMPLEC (SIMPLE corrected)

$$a_{i,J} u'_{i,J} = \sum_{neighbours} a_{nb} u'_{nb} + A_{i,J}(p'_{I-1,J} - p'_{I,J})$$

$$a_{i,J} u'_{i,J} - \sum_{neighbours} a_{nb} u'_{i,J} = \sum_{neighbours} a_{nb} u'_{nb} - \sum_{neighbours} a_{nb} u'_{i,J} + A_{i,J}(p'_{I-1,J} - p'_{I,J})$$

Neglect!!! Good estimate as soon as neighbours are not far from center

$$u'_{i,J} = \frac{A_{i,J}}{a_{i,J} - \sum_{neighbours} a_{nb}}(p'_{I-1,J} - p'_{I,J})$$

Improved d_{iJ}

CFD7

FVM SIMPLER (SIMPLE Revised)

Idea: calculate pressure distribution directly from the Poisson's equation

$$\nabla \bullet (\rho \vec{u}\vec{u}) = -\nabla p + \mu \nabla^2 \vec{u}$$

$$\nabla \bullet \nabla \bullet (\rho \vec{u}\vec{u}) = -\nabla \bullet \nabla p + \mu \nabla^2 \nabla \bullet \vec{u}$$

$$\nabla^2 p = f(u, v)$$

CFD7

FVM Rhie Chow (Fluent)

The way how to avoid staggering (difficult implementation in unstructured meshes) was suggested in the paper

Rhie C.M., Chow W.L.: Numerical study of the turbulent flow past an airfoil with trailing edge separation. AIAA Journal, Vol.21, No.11, 1983

we have at each cell discretised equation in this form,

$$a_p \vec{v}_P = \sum_{neighbours} a_l \vec{v}_l - \frac{\nabla p}{V}$$

For continuity we have

$$\sum_{faces} \left[\frac{1}{a_p} H \right]_{face} = \sum_{faces} \left[\frac{1}{a_p} \frac{\nabla p}{V} \right]_{face}$$

where

$$H = \sum_{neighbours} a_l \vec{v}_l$$

This interpolation of variables H and ∇p based on coefficients a_l for pressure velocity coupling is called **Rhie-Chow interpolation**. See also

CFD7 FVM solvers

Solution of linear algebraic equation system **Ax=b**

- TDMA – Gauss elimination tridiagonal matrix
(obvious choice for 1D problems, but suitable for 2D and 3D problems too, iteratively along the x,y,z grid lines – method of alternating directions ADI)

- PDMA – pentadiagonal matrix (suitable for QUICK), Fortran version

- CGM – Conjugated Gradient Method (iterative method: each iteration calculates increment of vector Φ in the direction of gradient of minimised function – square of residual vector)

- Multigrid method (Fluent)

CFD7

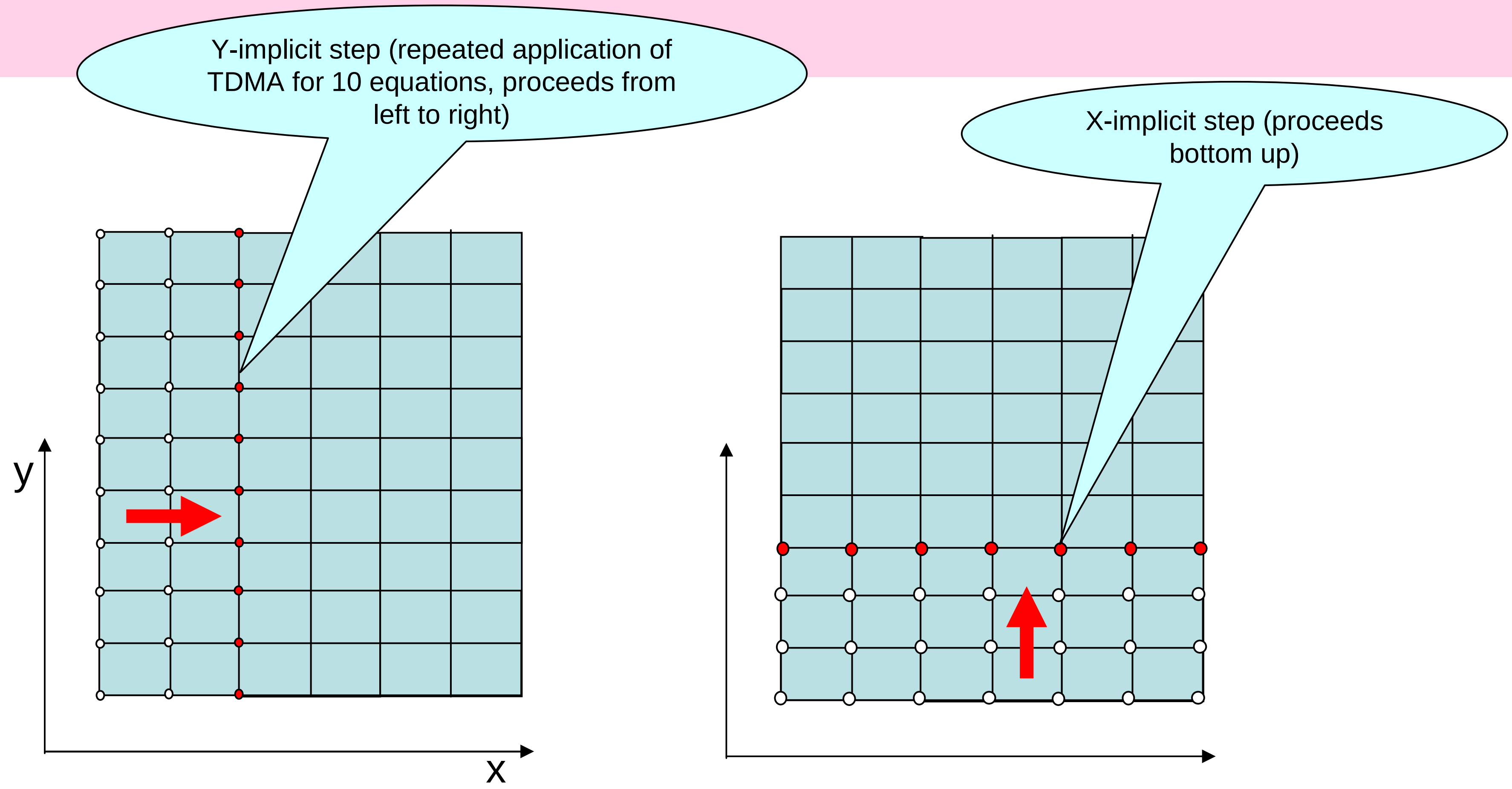
Solver TDMA tridiagonal system MATLAB

$$a_i x_{i-1} + b_i x_i + c_i x_{i+1} = d_i \qquad i = 1, 2, ..., n$$

```
function x = TDMAsolver(a,b,c,d)
%a, b, c, and d are the column vectors for the compressed tridiagonal matrix
n = length(b); % n is the number of rows

% Modify the first-row coefficients
c(1) = c(1) / b(1); % Division by zero risk.
d(1) = d(1) / b(1); % Division by zero would imply a singular matrix.

for i = 2:n
    id = 1 / (b(i) - c(i-1) * a(i));  % Division by zero risk.
    c(i) = c(i)* id;                 % Last value calculated is redundant.
    d(i) = (d(i) - d(i-1) * a(i)) * id;
end

% Now back substitute.
x(n) = d(n);
for i = n-1:-1:1
    x(i) = d(i) - c(i) * x(i + 1);
end
end
```

Forward step (elimination entries bellow diagonal)

CFD7 Solver TDMA applied to 2D problems (ADI)



CFD7 Solver PDMA pentagonal system MATLAB

```
function x=pentsolve(A,b)

% Solve a pentadiagonal system Ax=b where A is a strongly nonsingular
matrix
% Reference: G. Engeln-Muellges, F. Uhlig, "Numerical Algorithms with C"
%            Chapter 4. Springer-Verlag Berlin (1996)%
% Written by Greg von Winckel 3/15/04
% Contact: gregvw@chtm.unm.edu
%
[M,N]=size(A);
x=zeros(N,1);
% Check for symmetry
if A==A'   % Symmetric Matrix Scheme
   % Extract bands
   d=diag(A);
   f=diag(A,1);
   e=diag(A,2);
   alpha=zeros(N,1);
   gamma=zeros(N-1,1);
   delta=zeros(N-2,1);
   c=zeros(N,1);
   z=zeros(N,1);
   % Factor A=LDL'
   alpha(1)=d(1);
   gamma(1)=f(1)/alpha(1);
   delta(1)=e(1)/alpha(1);
   alpha(2)=d(2)-f(1)*gamma(1);
   gamma(2)=(f(2)-e(1)*gamma(1))/alpha(2);
   delta(2)=e(2)/alpha(2);
   for k=3:N-2
      alpha(k)=d(k)-e(k-2)*delta(k-2)-alpha(k-1)*gamma(k-1)^2;
      gamma(k)=(f(k)-e(k-1)*gamma(k-1))/alpha(k);
      delta(k)=e(k)/alpha(k);
   end
   alpha(N-1)=d(N-1)-e(N-3)*delta(N-3)-alpha(N-2)*gamma(N-2)^2;
   gamma(N-1)=(f(N-1)-e(N-2)*gamma(N-2))/alpha(N-1);
   alpha(N)=d(N)-e(N-2)*delta(N-2)-alpha(N-1)*gamma(N-1)^2;
   % Update Lx=b, Dc=z
   z(1)=b(1);
   z(2)=b(2)-gamma(1)*z(1);
   for k=3:N
      z(k)=b(k)-gamma(k-1)*z(k-1)-delta(k-2)*z(k-2);
   end
   c=z./alpha;
% Backsubstitution L'x=c
   x(N)=c(N);
   x(N-1)=c(N-1)-gamma(N-1)*x(N);
   for k=N-2:-1:1
      x(k)=c(k)-gamma(k)*x(k+1)-delta(k)*x(k+2);
   end
else      % Non-symmetric Matrix Scheme
   d=diag(A);
   e=diag(A,1);
   f=diag(A,2);
   h=[0;diag(A,-1)];
   g=[0;0;diag(A,-2)];
   alpha=zeros(N,1);
   gam=zeros(N-1,1);
   delta=zeros(N-2,1);
   bet=zeros(N,1);
   c=zeros(N,1);
   z=zeros(N,1);
   alpha(1)=d(1);
   gam(1)=e(1)/alpha(1);
   delta(1)=f(1)/alpha(1);
   bet(2)=h(2);
   alpha(2)=d(2)-bet(2)*gam(1);
   gam(2)=( e(2)-bet(2)*delta(1) )/alpha(2);
   delta(2)=f(2)/alpha(2);
   for k=3:N-2
      bet(k)=h(k)-g(k)*gam(k-2);
      alpha(k)=d(k)-g(k)*delta(k-2)-bet(k)*gam(k-1);
      gam(k)=( e(k)-bet(k)*delta(k-1) )/alpha(k);
      delta(k)=f(k)/alpha(k);
   end
   bet(N-1)=h(N-1)-g(N-1)*gam(N-3);
   alpha(N-1)=d(N-1)-g(N-1)*delta(N-3)-bet(N-1)*gam(N-2);
   gam(N-1)=( e(N-1)-bet(N-1)*delta(N-2) )/alpha(N-1);
   bet(N)=h(N)-g(N)*gam(N-2);
   alpha(N)=d(N)-g(N)*delta(N-2)-bet(N)*gam(N-1);
   % Update b=Lc
   c(1)=b(1)/alpha(1);
   c(2)=(b(2)-bet(2)*c(1))/alpha(2);

   for k=3:N
      c(k)=( b(k)-g(k)*c(k-2)-bet(k)*c(k-1) )/alpha(k);
   end
% Back substitution Rx=c
   x(N)=c(N);
   x(N-1)=c(N-1)-gam(N-1)*x(N);
   for k=N-2:-1:1
      x(k)=c(k)-gam(k)*x(k+1)-delta(k)*x(k+2);
   end

end
```

CFD7 **Solver CGM** **conjugated gradient GNU**

Solution of linear algebraic equation system **Ax=b**

```
function [x] = conjgrad(A,b,x)
    r=b-A*x;
    p=r;
    rsold=r'*r;

    for i=1:size(A)(1)
        Ap=A*p;
        alpha=rsold/(p'*Ap);
        x=x+alpha*p;
        r=r-alpha*Ap;
        rsnew=r'*r;
        if sqrt(rsnew)<1e-10
              break;
        end
        p=r+rsnew/rsold*p;
        rsold=rsnew;
    end
end
```

Minimisation of quadratic function

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x} - \mathbf{x}^{\mathrm{T}}\mathbf{b}, \quad \mathbf{x} \in \mathbf{R}^n.$$

Residual vector in k-th iteration

$$\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k.$$

Vector of increment conjugated with previous increments

$$\mathbf{p}_{k+1} = \mathbf{r}_k - \sum_{i \le k} \frac{\mathbf{p}_i^{\mathrm{T}}\mathbf{A}\mathbf{r}_k}{\mathbf{p}_i^{\mathrm{T}}\mathbf{A}\mathbf{p}_i}\mathbf{p}_i$$

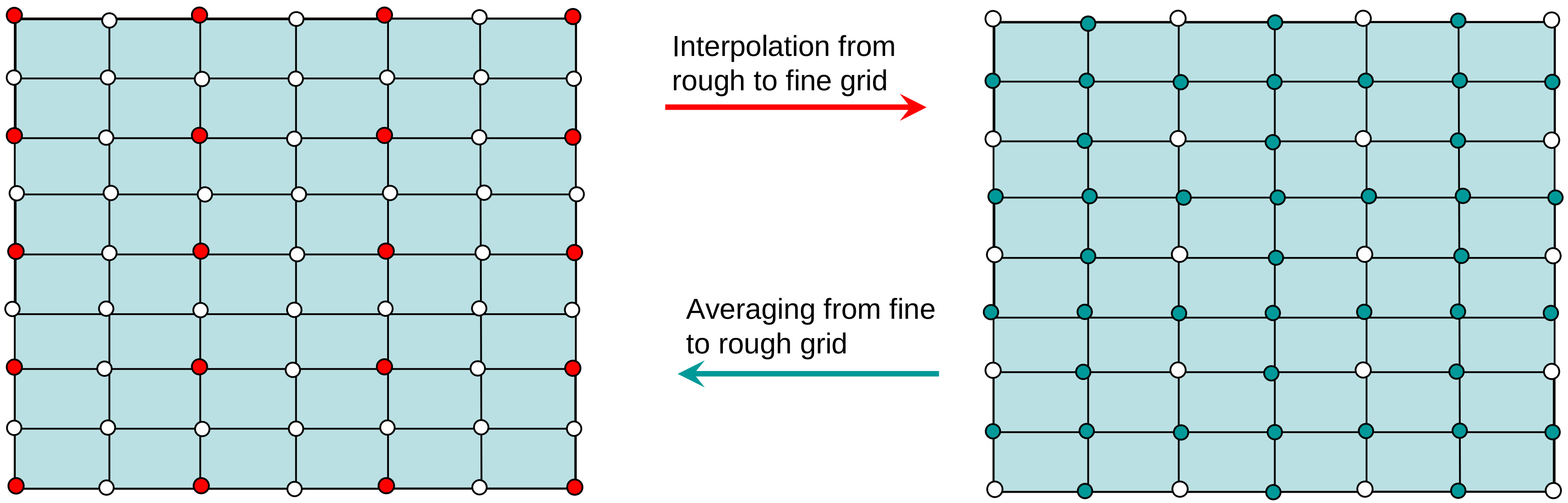
$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_{k+1}\mathbf{p}_{k+1}$$

$$\alpha_{k+1} = \frac{\mathbf{p}_{k+1}^{\mathrm{T}}\mathbf{r}_k}{\mathbf{p}_{k+1}^{\mathrm{T}}\mathbf{A}\mathbf{p}_{k+1}}$$

CFD7 **Solver MULTIGRID**

Solution on a rough grid takes into account very quickly long waves (distant boundaries etc), that is refined on a finer grid.

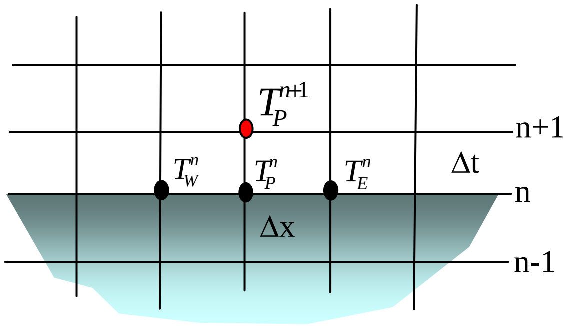
Rough grid (small wave numbers) **Fine grid** (large wave numbers details)

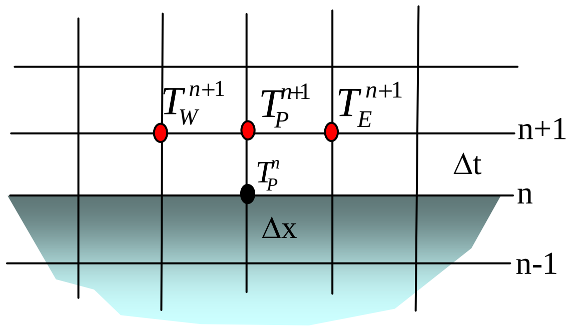


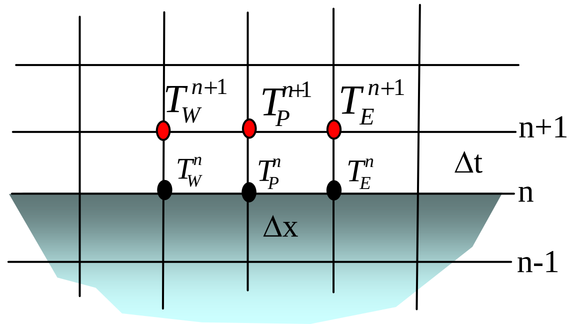
CFD7 Unsteady flows

Discretisation like in the finite difference methods discussed previously

Example: Temperature field $$\frac{\partial T}{\partial t} = a \frac{\partial^2 T}{\partial x^2}$$







EXPLICIT scheme

First order accuracy in time

Stable only if

$$\Delta t < \frac{\Delta x^2}{2a}$$

IMPLICIT scheme

First order accuracy in time

Unconditionally stable and bounded

CRANK NICHOLSON scheme

Second order accuracy in time

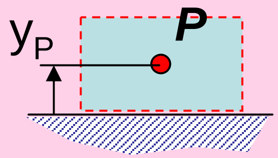
Unconditionally stable, but bounded only if

$$\Delta t < \frac{\Delta x^2}{a}$$

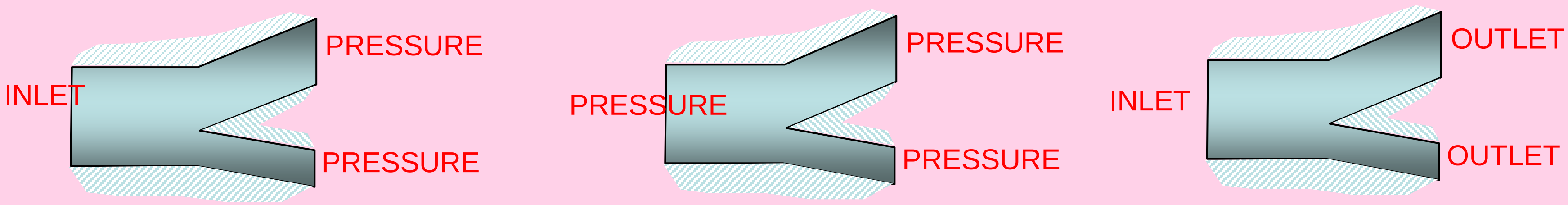
CFD7 FVM Boundary conditions

Boundary conditions are specified at faces of cells (not at grid points)

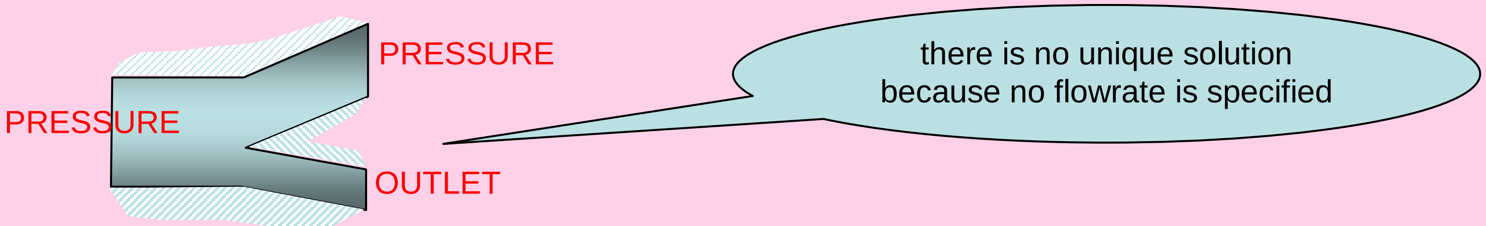
- **INLET** specified u,v,w,T,k,ε (but not pressure!)
- **OUTLET** nothing is specified (p is calculated from continuity eq.)
- **PRESSURE** only pressure is specified (not velocities)
- **WALL** zero velocities, k_P, ε_P calculated from the law of wall



Recommended combinations for several outlets

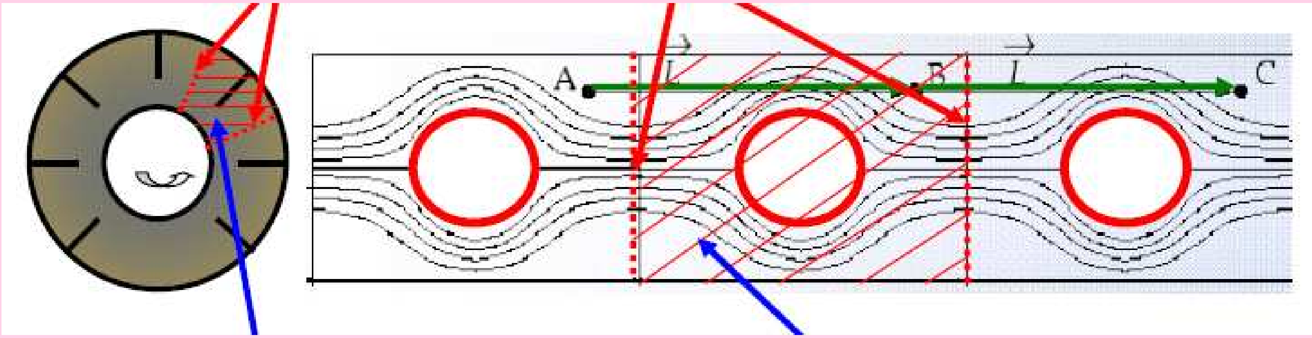


Forbidden combinations for several outlets



CFD7 FVM Boundary conditions

- ➢**SYMMETRY**
- ➢**PERIODICAL**



Copied from Kozubkova: Fluent-textbook