CFD - Computational fluid dynamics

A.S. differential analysis of fluid motion

5.16 continuity \( \nabla \cdot \mathbf{\bar{v}} + \frac{\partial \rho}{\partial t} = 0 \)

5.27 momentum \( \rho \frac{D\mathbf{\bar{v}}}{Dt} = \rho \mathbf{\bar{g}} - \nabla \rho + \mu \nabla^2 (\mathbf{\bar{v}}) \)

incompressible continuity \( \nabla \cdot \mathbf{\bar{v}} = 0 \)

\[ \begin{align*}
2D \ (x,y) \quad \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} &= 0 \\
\end{align*} \]

momentum - x

\[ \rho \left( \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) = \rho g_x - \frac{\partial \rho}{\partial x} + \mu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \]

steady, 2-D, no body forces, neglect viscous stresses

\[ \rho \left[ u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right] = -\rho \frac{\partial \rho}{\partial x} \]

\[ \rho \left[ u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right] = -\rho \frac{\partial \rho}{\partial y} \]

1, 2, 3 are 3 coupled differential equations, we want to solve for \( u = u(x,y) \quad v = v(x,y) \quad p = p(x,y) \)

For all but the most simple geometries, there is no analytical solution, so we may resort to CFD

we represent the continuously differentiable variables (\( u, v, p \)) with discrete variables at fixed locations (Euclidean description) and utilize our differential analysis / fluid motion at these locations.

Step 1 - Geometry, divide into a grid (a mesh)

Vin \[ \text{(inlet)} \]

Vout \[ \text{(outlet)} \]

\[ L_x = (n-1) \Delta x \quad \Delta x = \frac{L_x}{n-1} \]
Step 2 - apply our differential analysis at the nodes, convert our differential equation into an algebraic difference equation:

\[ \frac{\partial u}{\partial x} \bigg|_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} \]

\[ f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots \]

From first moment, we need \( \frac{\partial u}{\partial x} \) at location \( ij \):

\[ \frac{\partial u}{\partial x} \bigg|_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} \]

From Taylor series about point \( ij \):

\[ u(x_{ij} + \Delta x, y_{ij}) = u(x_{ij}, y_{ij}) + \frac{\partial u}{\partial x} \bigg|_{ij} \Delta x + \frac{\partial^2 u}{\partial x^2} \bigg|_{ij} \frac{(\Delta x)^2}{2!} + \frac{\partial^3 u}{\partial x^3} \bigg|_{ij} \frac{(\Delta x)^3}{3!} + \cdots \]

\[ u_{i+1,j} = u_{i,j} + \frac{\partial u}{\partial x} \bigg|_{ij} \Delta x + \frac{\partial^2 u}{\partial x^2} \bigg|_{ij} \frac{(\Delta x)^2}{2!} + \partial O(\Delta x) \]

The truncation error is of order \( \Delta x \) which we call this a first order method.
When you see a method labeled as \( 2^{nd} \) order \((3, 4, 5, 6)\)

It is referring to a \( T.E. \) of \( \Theta(\Delta x^2) \)

In small \( \Delta x \), \( \Delta x^2 < \Delta x \implies \) For a finer grid

a \( 2^{nd} \) order method is more accurate than a

first order method because it has a lower

truncation error.

We have \( \frac{\partial u}{\partial x} \) use similar methodology

for \( \frac{\partial u}{\partial y} \), \( \frac{\partial u}{\partial x} \) and substitute into

differentiated eq: for example, \( x \)-mon might look

like:

\[
P \left[ \frac{\partial u}{\partial x} \left( \frac{u(i+1,j)-u(i,j)}{\Delta x} \right) + \frac{\partial u}{\partial y} \left( \frac{u(i,j+1)-u(i,j)}{\Delta y} \right) \right] = -\frac{P(i,j) - P(i)}{\Delta x}
\]

Comment on this 'staggered'' scheme' (the algebraic

representation of the differential equation)

1) Must be consistent \( \frac{\partial u}{\partial x} \) goes to \( \frac{\partial u}{\partial x} \) @ \( \Delta x \to 0 \)

2) must be stable

3) must converge

\( \Delta x \to 0 \)

\( u(i,j) \to u(x, y) \)
1) we have geometry + mesh
2) we have numerical schemes
3) initial conditions / boundary conditions

- For steady flows (U-land handout) the initial conditions are "guess" conditions.
- For unsteady flows, they are the real initial condition at time zero, e.g., stationary flow with a piston, open valve at 0, examine flow development.

How do we solve the numerical scheme?

- Explicit/implicit:
  - Elementary, marching, explicit
  - Unsteady, use values at current time steps to get to next step, e.g.,
    \[ \frac{u^n_j - u^n_{j-1}}{\Delta t} + C \frac{u^n_{j+1} - 2u^n_j + u^n_{j-1}}{\Delta x} = 0 \quad C > 0 \]

\[ u^{n+1}_j = u^n_j - C \frac{\Delta t}{\Delta x} (u^n_j - u^n_{j+1}) \quad C > 0 \]

We know these at current time step from initial conditions at time 0.
- implicit

- solve equations simultaneously

\[
\frac{U_{j}^{n+1} - U_{j}^{n}}{\Delta t} + \frac{\partial}{\partial x} \left( C \frac{(U_{j+1}^{n} - U_{j-1}^{n})}{2 \Delta x} \right) = 0
\]

\[
\frac{1}{2} U_{j+1}^{n} + \frac{1}{2} U_{j-1}^{n} - \frac{1}{2} U_{j}^{n+1} = U_{j}^{n}
\]

when \( U = \frac{c \Delta t}{\Delta x} \)

\[
A \cdot U_{j+1}^{n} + d_{j} U_{j}^{n+1} + b_{j} U_{j-1}^{n} = C_{j}
\]

\[
A_{j} = -\frac{1}{2} \quad \quad \quad d_{j} = 1 \quad \quad \quad b_{j} = -\frac{1}{2} \quad \quad \quad C_{j} = U_{j}
\]

- boundary conditions

\[
\begin{bmatrix}
A & U_{-1}^{n} & 0 \\
0 & U_{0}^{n} & 0 \\
\vdots & \vdots & \vdots \\
0 & \ldots & U_{n-1}^{n}
\end{bmatrix}
\begin{bmatrix}
U_{-1}^{n+1} \\
U_{0}^{n+1} \\
\vdots \\
U_{n-1}^{n+1}
\end{bmatrix}
= \begin{bmatrix}
C_{-1} \\
C_{0} \\
\vdots \\
C_{n-1}
\end{bmatrix}
\]

\[
A U = C
\]

\[
A^{-1} U = C
\]

- can solve thus

- tridiagonal system of linear algebraic equations
- sequentially
- Thomas Algorithm
- ordinary

\[
U = A^{-1} C
\]
- Solving equations, how do we know when we are finished?
- Convergence - due to Fluent, convergence means the solution is changing very little from one iteration to the next.
- A residual of 0.001

\[ R_{\text{Iteration}} \leq 0.001 \]

Example: when for continuity

\[ R = \sum_{\text{cells}} \left| \text{rate of mass creation in cell} \right| \]

\[ \text{mass} \]

\[ \frac{\partial u}{\partial N} = n \cdot \nabla u = f(x, t) \]

- Boundary conditions
  - Dirichlet - specify value of function on the boundary \( U = U(x, t) \)
  - Neumann - specify normal derivative of function on the boundary
    \[ \frac{\partial U}{\partial N} = n \cdot \nabla U = f(x, t) \]
- Cauchy - weighted average of \( \frac{\partial U}{\partial N} \)