Numerical Methods

Krzysztof Tesch

Mechanical Engineering Department
Gdańsk University of Technology
<table>
<thead>
<tr>
<th>Description of fluid/solid at different scales</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finite Difference Method</td>
</tr>
<tr>
<td>Finite Volume Method</td>
</tr>
<tr>
<td>Finite Element Method</td>
</tr>
<tr>
<td>Monte Carlo Method</td>
</tr>
<tr>
<td>Smoothed Particle Hydrodynamics</td>
</tr>
<tr>
<td>Lattice Boltzmann Method</td>
</tr>
<tr>
<td>Turbulence modelling</td>
</tr>
<tr>
<td>References</td>
</tr>
</tbody>
</table>

**Contents**

- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
Description of fluid/solid at different scales

K. Tesch; Numerical Methods
Fluid motion may be described by three types of mathematical models according to the observed scales:

- Microscopic description (MD)
- Mesoscopic description
  - kinetic theory
  - LD
  - BD
  - DPD
  - SPH
  - LBM
- Macroscopic description – continuum (FDM, FEM, FVM, LBM)
Molecular mechanics takes advantage of classical mechanics equations to model molecular systems whereas molecular dynamics simulates movements of atoms in the context of N-body simulation. The motion of molecules is determined by solving the Newton's equation of motion

$$m \frac{d^2 \mathbf{r}_i}{dt^2} = \mathbf{G}_i + \sum_{j=1 \neq i}^{N} \mathbf{f}_{ij} \tag{1}$$

The force exerted on a molecule consists of the external force such as gravity $\mathbf{G}_i$ and the intermolecular force $\mathbf{f}_{ij} = -\nabla V$ usually described by means of the Lennard-Jones potential.
Microscopic description – MD

\[ V = 4\epsilon \left( \left( \frac{\sigma}{\|r\|} \right)^{12} - \left( \frac{\sigma}{\|r\|} \right)^6 \right) \]  

(2)

In the above equations \( \|r\| \) is the distance between particles, \( \epsilon \) – the depth of the potential well that characterises the interaction strength and \( \sigma \) – the finite distance describing the interaction range. Further, the ensemble average makes it possible to obtain a macroscopic quantity from the corresponding microscopic variable. The disadvantage of molecular dynamics method is that the total number of molecules even in small volume is too large – proportional to \( 10^{23} \).
Microscopic description – MD

\[ V = 4\varepsilon \left( \left( \frac{\sigma}{\|r\|} \right)^{12} - \left( \frac{\sigma}{\|r\|} \right)^{6} \right) \]  

\( (3) \)
Molecular dynamics pseudocode

\[ t \leftarrow 0; \]
Calculate initial molecule position \( r \);

\begin{align*}
\textbf{while} \; \text{not the end of calculations} \; \textbf{do} \\
& f_{ij} \leftarrow -\nabla V; \\
& a \leftarrow m^{-1} f_{ij}; \\
& r \leftarrow r + v \Delta t + \frac{1}{2} a \Delta t^2; \\
& t \leftarrow t + \Delta t;
\end{align*}
Molecular dynamics - example
The key concept is the probability distribution function \( f^{(N)} \) in the phase space. The phase space is constituted of 3\( N \) spatial coordinates \( q_1, \ldots, q_N \) and 3\( N \) momenta \( p_1, \ldots, p_N \). The probability distribution function \( f^{(N)} \) allows to express the probability to find a particle within the infinitesimal phase space

\[
(q_1, q_1 + dq) \times \ldots \times (q_N, q_N + dq) \times \\
(p_1, p_1 + dp) \times \ldots \times (p_N, p_N + dp) \quad (4)
\]

The total number of molecules within the infinitesimal phase space is then

\[
f^{(N)} (q_1, \ldots, q_N, p_1, \ldots, p_N) \ dq^N \ dp^N \quad (5)
\]
The time evolution of the probability distribution function $f^{(N)}$ follows the Liouville equation

$$\frac{df^{(N)}}{dt} = \frac{\partial f^{(N)}}{\partial t} + \sum_{i=1}^{N} \left( \frac{\partial f^{(N)}}{\partial p_i} \cdot \frac{dp_i}{dt} + \frac{\partial f^{(N)}}{\partial q_i} \cdot \frac{dq_i}{dt} \right) = 0$$

This means that the distribution function is constant along any trajectory in phase space. The reduced probability distribution function is defined as

$$F_s \left( q_1, \ldots, q_s, p_1, \ldots, p_s \right) = \int \int f^{(N)} \left( q_1, \ldots, q_N, p_1, \ldots, p_N \right) dq^{N-s} dp^{N-s}$$
The above function is called the s-particle probability distribution function. A chain of evolution equations for $F_s$ for $1 \leq s \leq N$ is derived and called BBGKY hierarchy. This means that the $s^{th}$ equation for the $s$-particle distributions contains $s + 1$ distribution. That hierarchy may be truncated. Truncating it at the first order results in Boltzmann equation

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f = \Omega(f)$$

for the probability distribution function

$$f(\mathbf{r}, \mathbf{v}, t) = mN F_1(q_1, p_1, t)$$

for binary collisions with uncorrelated velocities before that collision.
Mesoscopic description – DPD

Contents

Description of fluid/solid at different scales
Finite Difference Method
Finite Volume Method
Finite Element Method
Monte Carlo Method
Smoothed Particle Hydrodynamics
Lattice Boltzmann Method
Turbulence modelling
References
The DPD (Dissipative Particle Dynamics) method simulate only a reduced number of degrees of freedom (coarse-grained models). The motion of particles is determined by solving the Newton's equation of motion

\[
m \frac{d^2 \mathbf{r}_i}{dt^2} = \mathbf{G}_i + \sum_{j=1 \neq i}^{N} \left( f_{ij}^C + f_{ij}^D + f_{ij}^R \right)
\]

where the interaction forces are the sum of

- \( f_{ij}^C \): conservative or repulsion forces
- \( f_{ij}^D \): dissipative forces
- \( f_{ij}^R \): random force
Mesoscopic description – DPD

- **Dissipative forces**

  \[ \mathbf{f}_{ij}^D = -\gamma \omega_D (\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij}) \hat{\mathbf{r}}_{ij} \]  
  \[ (8) \]

- **Random force**

  \[ \mathbf{f}_{ij}^R = \sigma \omega_R \hat{\mathbf{r}}_{ij} \frac{\theta_{ij}}{\sqrt{\Delta t}} \]  
  \[ (9) \]

- **Conservative forces**

  \[ \mathbf{f}_{ij}^C = \alpha \omega_R \hat{\mathbf{r}}_{ij} \]  
  \[ (10) \]

\[ \omega_D = \omega_R^2 \text{ and } \sigma^2 = 2\gamma k_B T \]
The Langevin dynamics equation of motion

\[ m \frac{d^2 x_i}{dt^2} = f_i^C - \gamma v_i + f_i^R \]  \hspace{1cm} (11)

where

- Dissipative forces \(-\gamma v_i\)
- Random force \(f_i^R\)
- Conservative forces

\[ f_i^C = -\nabla V \]  \hspace{1cm} (12)
The Brownian dynamics equation of motion

\[ 0 = f_i^C - \gamma v_i + f_i^R \]  \hspace{1cm} (13)

or

\[ \gamma \frac{dx_i}{dt} = f_i^C + f_i^R \]  \hspace{1cm} (14)

where

- Random force \( f_i^R \)
- Conservative forces

\[ f_i^C = -\nabla V \]  \hspace{1cm} (15)
Mesoscopic description – SPH

The Smoothed Particle Hydrodynamics equation of motion

\[ \frac{d\mathbf{u}_i}{dt} = \mathbf{g}_i - \mathbf{f}_{pi} + \mathbf{f}_{\mu i} \]  \hspace{1cm} (16)

where

- Body force \( \mathbf{g}_i \)
- Pressure gradient forces \( \mathbf{f}_{pi} \)
- Viscous forces \( \mathbf{f}_{\mu i} \)
Macroscopic description – conservation equations

Conservation of mass

\[ \frac{d\rho}{dt} + \rho \nabla \cdot \mathbf{u} = 0 \quad (17) \]

Conservation of linear momentum

\[ \rho \frac{d\mathbf{u}}{dt} = \rho \mathbf{f} + \nabla \cdot \mathbf{\sigma} \quad (18) \]

Decomposition of stress tensor

\[ \mathbf{\sigma} = -p \mathbf{\delta} + \mathbf{\tau} \quad (19) \]

Another form of conservation of linear momentum

\[ \rho \frac{d\mathbf{u}}{dt} = \rho \mathbf{f} - \nabla p + \nabla \cdot \mathbf{\tau} \quad (20) \]
## Macroscopic description – energy equations

<table>
<thead>
<tr>
<th>Energy</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kinetic</td>
<td>$\rho \frac{d\text{e}_k}{dt} = \rho \mathbf{f} \cdot \mathbf{u} + \nabla \cdot (\sigma \cdot \mathbf{u}) - \nabla \cdot \mathbf{q}$</td>
</tr>
<tr>
<td>Total</td>
<td>$\rho \frac{d\text{e}_c}{dt} = \frac{\partial p}{\partial t} + \nabla \cdot (\tau \cdot \mathbf{u}) - \nabla \cdot \mathbf{q}$</td>
</tr>
<tr>
<td>Mechanical</td>
<td>$\rho \frac{d\text{e}_m}{dt} = \nabla \cdot (\sigma \cdot \mathbf{u}) - \sigma : \mathbf{D}$</td>
</tr>
<tr>
<td>Internal</td>
<td>$\rho \frac{d\text{e}}{dt} = \sigma : \mathbf{D} - \nabla \cdot \mathbf{q}$</td>
</tr>
<tr>
<td>Enthalpy</td>
<td>$\rho \frac{dh}{dt} = \tau : \mathbf{D} - \nabla \cdot \mathbf{q} + \frac{dp}{dt}$</td>
</tr>
</tbody>
</table>
Macroscopic description – general transport equations

\[ \frac{\partial (\rho f)}{\partial t} + \nabla \cdot (\rho u f) = S_f - \nabla \cdot k \quad (21) \]

Left hand side represents transient and convection effects. It expresses the rate of change
\[ \rho \frac{df}{dt} = \frac{\partial (\rho f)}{\partial t} + \nabla \cdot (\rho u f). \]
Right hand side represents sources (positive and negative) and fluxes (transport due to other mechanism than convection).

- mass conservation equation
  \[ f := 1, \quad S_f := 0, \quad k := 0 \]
- linear momentum conservation equation
  \[ f \leftarrow u, \quad S_f \leftarrow \rho f, \quad k \leftarrow -\sigma \]
- energy conservation equation
  \[ f := e_k, \quad S_f := \rho f \cdot u, \quad k := q - \sigma \cdot u \]
Macroscopic description – laws of thermodynamics

Second law of thermodynamics

\[ \rho \frac{ds}{dt} \geq -\nabla \cdot \frac{q}{T} \quad (22) \]

Entropy balance

\[ \rho \frac{ds}{dt} = \frac{\phi}{T} - \nabla \cdot \frac{q}{T} \quad (23) \]

First law of thermodynamics

\[ \rho \frac{de}{dt} = \tau : D - p \nabla \cdot u - \nabla \cdot q \quad (24) \]
Macroscopic description – constitutive equations

- Mechanical (rheological) constitutive equations
- Equations of state
- Fluxes
Newtonian fluids $\tau = 2\mu D$

Non-Newtonian fluids

- Generalised Newtonian fluids $\tau = 2\mu(\gamma)D$
- Differential type fluid $\tau = f(A_1, A_2, \ldots)$

\[
A_{i+1} = \frac{dA_i}{dt} + A_i \cdot \frac{\partial u}{\partial r} + \nabla u \cdot A_i, \quad i = 1, 2, \ldots
\]

- Integral type fluids

\[
\tau = \int_{-\infty}^{t} f(t - \tau) (\delta - C_t(\tau)) \, d\tau
\]

- Rate type fluids $\dot{\tau} = f(\tau, D, \dot{D})$

\[
\tau + \lambda_1 \dot{\tau} = 2\mu \left( D + \lambda_2 \dot{D} \right)
\]
# Generalised Newtonian fluids

<table>
<thead>
<tr>
<th>Szulman</th>
<th>Generalised Herschel</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau^{\frac{1}{n}} = \tau_0^{\frac{1}{n}} + (k\gamma)^{\frac{1}{m}}$</td>
<td>$\tau = \tau_0 + k\gamma^n + \mu_\infty \gamma$</td>
</tr>
<tr>
<td>$\mu^{\frac{1}{n}} = \left(\frac{\tau_0}{</td>
<td>\gamma</td>
</tr>
</tbody>
</table>

$m := n$  
$n := 1, m := \frac{1}{n}$  
$\mu_\infty := 0$  
$n := \frac{1}{2}$

<table>
<thead>
<tr>
<th>Generalised Casson</th>
<th>Generalised Herschel</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau^{\frac{1}{n}} = \tau_0^{\frac{1}{n}} + (k\gamma)^{\frac{1}{n}}$</td>
<td>$\tau = \tau_0 + k\gamma^n$</td>
</tr>
<tr>
<td>$\mu^{\frac{1}{n}} = \left(\frac{\tau_0}{</td>
<td>\gamma</td>
</tr>
</tbody>
</table>

$n := 2$  
$n := 1$  
$n := 1$  
$\tau_0 := 0$

<table>
<thead>
<tr>
<th>Herschel-Bulkley</th>
<th>Luo-Kuang</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau = \tau_0 + k\gamma^n$</td>
<td>$\tau = \tau_0 + k\sqrt{\gamma} + \mu_\infty \gamma$</td>
</tr>
<tr>
<td>$\mu = \frac{\tau_0}{</td>
<td>\gamma</td>
</tr>
</tbody>
</table>

$n := 2$  
$n := 1$  
$n := 1$  
$\tau_0 := 0$

<table>
<thead>
<tr>
<th>Casson</th>
<th>Bingham</th>
<th>Ostwald-de Waele</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sqrt{\tau} = \sqrt{\tau_0} + \sqrt{k\gamma}$</td>
<td>$\tau = \tau_0 + k\gamma$</td>
<td>$\tau = k\gamma^n$</td>
</tr>
<tr>
<td>$\sqrt{\mu} = \sqrt{\frac{\tau_0}{</td>
<td>\gamma</td>
<td>}} + \sqrt{k}$</td>
</tr>
</tbody>
</table>

$k := 0$,  
$\mu_\infty := k$,  
$\tau_0 := 0$

<table>
<thead>
<tr>
<th>Newton</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau = k\gamma$</td>
</tr>
</tbody>
</table>

$\tau_0 := 0$  
$n := 1$

---

K. Tesch; Numerical Methods

---

# Contents

- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
Macroscopic description – equations of state

- **Fundamental equation of state** \( e = f(s, \rho^{-1}) \)

\[
\frac{de}{dt} = T \frac{ds}{dt} + \frac{p}{\rho^2} \frac{d\rho}{dt}
\]  \( (25) \)

- **Thermal equation of state** \( p = f(T, \rho^{-1}) \)

\[
p = \rho RT
\]  \( (26) \)

- **Caloric equation of state** \( e = f(T, \rho^{-1}) \)

\[
de = c_v \, dT + \left( T \frac{\partial p}{\partial T} - p \right) \, d\rho^{-1}
\]  \( (27) \)
Macroscopic description – fluxes

General form \( \mathbf{w} = -\mathbf{T} \cdot \nabla \varphi \) due to assumption \( \mathbf{w} = f(\nabla \varphi) \). More precisely \( \mathbf{w} \) depends only on \( \varphi \) and \( \nabla \varphi \).

- Fourier’s law
  \[
  q = -\lambda \cdot \nabla T
  \]  \hspace{1em} (28)

- Fick’s law
  \[
  j^i = -\rho D^{ij} \cdot \nabla g^i
  \]  \hspace{1em} (29)

- Darcy’s law
  \[
  \mathbf{u} = -\mu^{-1} \mathbf{K} \cdot \nabla p
  \]  \hspace{1em} (30)

In the case of isotropy \( \mathbf{T} = \alpha \delta \) and

\[
\mathbf{w} = -\alpha \delta \cdot \nabla \varphi = -\alpha \nabla \varphi
\]  \hspace{1em} (31)
Macroscopic description – general transport equations

General transport equations

\[ \frac{\partial (\rho f)}{\partial t} + \nabla \cdot (\rho u f) = S_f - \nabla \cdot k \]  \hspace{1cm} (32)

Fluxes

\[ k = -\Gamma \delta \cdot \nabla f = -\Gamma \nabla f \]  \hspace{1cm} (33)

In the case of isotropy the general transport equations becomes

\[ \frac{\partial (\rho f)}{\partial t} + \nabla \cdot (\rho uf) = S_f + \nabla \cdot (\Gamma \nabla f) \]  \hspace{1cm} (34)
Macroscopic description

General form of the Navier-Stokes equation for Newtonian fluids

\[ \rho \frac{d\mathbf{u}}{dt} = \rho \mathbf{f} - \nabla p + \nabla \cdot (2\mu \mathbf{D}^\varepsilon) \quad (35) \]

- incompressible flow
- creeping flow
- inviscid flow
- Boussinesq approximation
- Oseen approximation
- filtration
- one-dimensional flows
- heat transfer
- surface tension
- incompressible fluid ($\rho = \text{const}$)

\[ \rho \frac{\mathbf{d}\mathbf{u}}{\mathbf{d}t} = \rho \mathbf{f} - \nabla p + \nabla \cdot (2\mu \mathbf{D}) \quad (36) \]

\[ \mu = \text{const} \]

\[ \rho \frac{\mathbf{d}\mathbf{u}}{\mathbf{d}t} = \rho \mathbf{f} - \nabla p + \mu \nabla^2 \mathbf{u} \quad (37) \]

\[ \mu = \rho \nu \]

\[ \frac{\mathbf{d}\mathbf{u}}{\mathbf{d}t} = \mathbf{f} - \nabla \frac{p}{\rho} + \nu \nabla^2 \mathbf{u} \quad (38) \]

\[ p_k = \frac{p}{\rho} + \mathbf{g} \cdot \mathbf{r} \]

\[ \frac{\mathbf{d}\mathbf{u}}{\mathbf{d}t} = -\nabla p_k + \nu \nabla^2 \mathbf{u} \quad (39) \]
Macroscopic description

– creeping flow

\[ \rho \frac{\partial \mathbf{u}}{\partial t} = \rho \mathbf{f} - \nabla p + \nabla \cdot (2\mu \mathbf{D}) \]  \hspace{1cm} (40)

steady state

\[ \nabla p = \nabla \cdot (2\mu \mathbf{D}) \]  \hspace{1cm} (41)

2D creeping flow

\[ \nabla^4 \psi = 0 \]  \hspace{1cm} (42)
Macroscopic description

- inviscid flow ($\mu = 0$)

\[ \rho \frac{d\mathbf{u}}{dt} = \rho \mathbf{f} - \nabla p \] (43)

- potential flows ($\nabla \times \mathbf{u} = 0 \iff \mathbf{u} = \nabla \varphi$)

\[ \nabla^2 \varphi = \nabla^2 \psi = 0 \] (44)
Macroscopic description

- Boussinesq approximation

\[ \rho \mathbf{f} = \rho \mathbf{g} = \rho_0 \mathbf{g} + (\rho - \rho_0) \mathbf{g} \]  \hspace{1cm} (45)

\[ \rho - \rho_0 = -\rho_0 \beta (T - T_0) \]  \hspace{1cm} (46)

\[ \rho \frac{d\mathbf{u}}{dt} = \rho_0 \mathbf{g} \left(1 - \beta (T - T_0)\right) - \nabla p + \nabla \cdot \left(2\mu \mathbf{D}\right) \]  \hspace{1cm} (47)
- Oseen approximation (linearisation)

\[ u \cdot \nabla u \approx u_\infty \cdot \nabla u \quad (48) \]

\[ \rho \frac{\partial u}{\partial t} + \rho u_\infty \cdot \nabla u = \rho f - \nabla p + \nabla \cdot (2\mu D^D) \quad (49) \]
– filtration

$$\rho \frac{du}{dt} = \rho f - \nabla p + \mu \nabla^2 u - R_1 u$$  \hspace{1cm} (50)

– one-dimensional flows

$$\nabla^2 u = a$$  \hspace{1cm} (51)
– heat transfer The ‘fluid’ Fourier equation describes the temperature field in the fluid.

\[
  c_v \left( \frac{\partial (\rho T)}{\partial t} + \nabla \cdot (\rho T \mathbf{u}) \right) = \phi_\mu + \nabla \cdot (\lambda \nabla T) \quad (52)
\]

For solids where \( \mathbf{u} = 0 \) the above equation simplifies to the ‘solid’ Fourier-Kirchhoff equation

\[
  c \frac{\partial (\rho T)}{\partial t} = \nabla \cdot (\lambda \nabla T) + S_E \quad (53)
\]

where internal energy sources are given by \( S_E \).
Generally, the ‘fluid’ equation should be solved together with ‘solid’ equation. This is called conjugate heat transfer. Not having to know the heat transfer coefficient is an advantage of this approach. The disadvantage is the necessity of increasing the total number of mesh elements due to the additional solid volume.

It is not always possible because of storage limitations. Then either the temperature or heat flux must be specified at the wall. Alternatives, through boundary conditions, are discussed further such as specified temperature, specified heat flux, specified temperature and heat flux, adiabatic or specified heat transfer coefficient.
Macroscopic description

– surface tension

\[
\frac{du}{dt} = f - \nabla \frac{p}{\rho} + \nu \nabla^2 u \tag{54}
\]

body forces

\[
f = g + \rho^{-1} f_\sigma \tag{55}
\]

\[
f_\sigma = \sigma \kappa \nabla \alpha \tag{56}
\]

\[
\kappa = - \nabla \cdot \hat{n} = - \nabla \cdot \frac{\nabla \alpha}{\| \nabla \alpha \|} \tag{57}
\]

\(\alpha\) transport equation

\[
\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha u) = 0 \tag{58}
\]
Macroscopic description – dimensionless form of equations

Mass conservation equation

$$\nabla^+ \cdot u^+ = 0 \quad (59)$$

Linear momentum conservation equation

$$\rho^+ \left( Sh \frac{\partial u^+}{\partial t^+} + u^+ \cdot \nabla^+ u^+ \right) =$$

$$= \frac{\rho^+ f^+}{Fr} - Eu \nabla^+ p^+ + \frac{\mu^+}{Re} \nabla^{2+} u^+ \quad (60)$$

Fourier-Kirchhoff (internal energy)

$$\rho^+ c_v^+ \left( Sh \frac{\partial T^+}{\partial t^+} + u^+ \cdot \nabla^+ T^+ \right) =$$

$$= \frac{Ec}{Re} \phi_{\mu}^+ + \frac{\lambda^+}{Pr \ Re} \nabla^{2+} T^+ \quad (61)$$
Macroscopic description – dimensionless numbers

\[ Sh = \frac{L}{Ut_0} = \frac{t_{ch}}{t_0} = \frac{L_f}{U} = \frac{\mu}{t_0^2} \]

\[ Fr = \frac{U^2}{f_0L} = \frac{\rho_0U^2}{\rho_0f_0} \]

\[ Re = \frac{LU\rho_0}{\mu_0} = \frac{LU}{\nu_0} = \frac{\rho_0U^2}{\mu_0U/L^2} \]

\[ Eu = \frac{p_0}{\rho_0U^2} = \frac{p_0}{L} \]

\[ Ec = \frac{U^2}{c_{v0}T_0} \]

\[ Pr = \frac{\nu_0}{\lambda_0} = \frac{\nu_0}{c_{v0}\rho_0} \]

\[ Sc = \frac{\mu_0}{\rho_0D_0} = \frac{\nu_0}{\rho_0D_0} \]

\[ Da = \frac{K_0}{L^2} \]

\[ De = \frac{\lambda_0}{t_0} \]

\[ Wi = \frac{\lambda_0}{\gamma_0} \]
General transport equation

\[
\frac{\partial (\rho f)}{\partial t} + \nabla \cdot (\rho u f) = S_f - \nabla \cdot k \tag{62}
\]

From Reynolds’ transport theorem arises general compatibility condition \( \hat{n} \cdot [\rho u f + k] = 0 \)

- Mass conservation: \( f := 1, S_f := 0, k := 0 \). C.C. takes form \( \hat{n} \cdot [\rho u] = 0 \) or \( \hat{n} \cdot [u] \equiv [u_n] = 0 \)
- Linear momentum: \( f \leftarrow u, S_f \leftarrow \rho f, k \leftarrow -\sigma \) and C.C. \( \hat{n} \cdot [\rho uu - \sigma] = 0 \) or \( \hat{n} \cdot [\sigma] = [\sigma_n] = 0 \)
- Energy conservation: \( f := e_k, S_f := \rho f \cdot u, k := q - \sigma \cdot u \) and C.C. \( \hat{n} \cdot [\rho u e_k - \sigma \cdot u + q] = 0 \) or \( \hat{n} \cdot [q] \) or \( [\lambda \frac{\partial T}{\partial n}] = 0 \)
Compatibility conditions are insufficient! Further conditions are needed:

- **adhesion** \( \hat{l} \cdot \mathbf{u} = u_l = 0 \)
- **thermal equilibrium on surfaces** \([T] = 0\)

Boundary condition related to heat transfer (arise from C.C.)

- **Dirichlet**: \( T = f_1(x, y, z, t) \)
- **Neumann**: \( q_n = \hat{n} \cdot \mathbf{q} \) or \( q_n = f_2(x, y, z, t) \)
- mixed: \( \alpha T - \lambda \frac{\partial T}{\partial n} = f_3(P, t) \)
Other surfaces than walls

- inlet: $n - 1$ conditions where $n$ stands for the number of equations
- outlet: Generally, $\sigma_n = \hat{n} \cdot \sigma$ plus $T$ distribution. Usually $p$ distribution due to $\sigma_n \approx -p\hat{n}$ plus $\frac{\partial T}{\partial n} = 0$
- symmetry: $\frac{\partial \varphi}{\partial n} = 0$ for all scalar variables $\varphi$
- periodicity (translation and rotation): $\varphi(P_1) = \varphi(P_2)$ where $P_1$ and $P_2$ are corresponding points on periodic surfaces
Mathematical classification

General partial differential equation

\[ F \left( \frac{\partial^k f}{\partial x_1^{n_1} \ldots \partial x_n^{n_n}}, \frac{\partial^{k-1} f}{\partial x_1^{m_1} \ldots \partial x_n^{m_n}}, \ldots, \frac{\partial f}{\partial x_i}, f, x \right) = 0 \]  
(63)

- linear
- semi-linear

\[ \sum_i a_i(x) \frac{\partial^k f}{\partial x_1^{n_1} \ldots \partial x_n^{n_n}} + a_0 \left( \frac{\partial^{k-1} f}{\partial x_1^{m_1} \ldots \partial x_n^{m_n}}, \ldots, \frac{\partial f}{\partial x_i}, f, x \right) = 0 \]  
(64)

- quasi-linear
- fully non-linear
The Navier-Stokes equations are second order nonlinear partial differential equations. In general, they cannot be easily classified. However they posses properties of semi-linear and linear second order partial differential equations. Sometimes they can be simplified to those and can be divided into:

- hyperbolic
- parabolic
- elliptic
Semi-linear second order PDEs

For two independent variables \( x, y \):

\[
A(x, y) \frac{\partial^2 f}{\partial x^2} + B(x, y) \frac{\partial^2 f}{\partial x \partial y} + C(x, y) \frac{\partial^2 f}{\partial y^2} + \\
F \left( x, y, f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) = 0 \quad (65)
\]

for all \((x, y)\) over a domain \(\Omega\) the above equation is:

- hyperbolic if \(B^2 - 4AC > 0\)
- parabolic if \(B^2 - 4AC = 0\)
- elliptic if \(B^2 - 4AC < 0\)
Important elliptic second order PDEs

- **Laplace equation**

\[
\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = 0 \quad (66)
\]

- **Poisson equation**

\[
\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f(x, y) \quad (67)
\]

- **Helmholtz equation**

\[
\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + k^2 f = 0 \quad (68)
\]
Important parabolic second order PDEs

■ Heat equation

\[
\frac{\partial f}{\partial t} - \alpha \frac{\partial^2 f}{\partial x^2} = 0 \quad (69)
\]

\[
\frac{\partial f}{\partial t} - \alpha \frac{\partial^2 f}{\partial x^2} = g(x, t) \quad (70)
\]

■ Convection-diffusion equation

\[
\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} = \nu \frac{\partial^2 f}{\partial x^2} \quad (71)
\]
Important hyperbolic second order PDEs

- Wave equation
  \[ \frac{\partial^2 f}{\partial t^2} - a^2 \frac{\partial^2 f}{\partial x^2} = 0 \]  \hspace{1cm} (72)

- Telegraph equations
  \[ \frac{\partial^2 f}{\partial x^2} - a \frac{\partial^2 f}{\partial t^2} - b \frac{\partial f}{\partial t} - c f = 0 \]  \hspace{1cm} (73)

- Convection equations
  \[ \frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} = 0 \]  \hspace{1cm} (74)
Important mixed type second order PDEs

- Euler-Tricomi equation

\[
\frac{\partial^2 f}{\partial x^2} - x \frac{\partial^2 f}{\partial y^2} = 0 \quad (75)
\]

It is of hyperbolic type for \( x > 0 \), parabolic at \( x = 0 \) and elliptic for \( x < 0 \).

- Generalised Euler-Tricomi equation

\[
\frac{\partial^2 f}{\partial x^2} - g(x) \frac{\partial^2 f}{\partial y^2} = 0 \quad (76)
\]

K. Tesch; Numerical Methods
Potential gas flow

\[(1 - \text{Ma}^2_\infty) \frac{\partial^2 \varphi'}{\partial x^2} + \frac{\partial^2 \varphi'}{\partial y^2} = 0 \quad (77)\]

It is of hyperbolic type for \(\text{Ma}^2_\infty > 1\), parabolic at \(\text{Ma}^2_\infty = 1\) and elliptic for \(\text{Ma}^2_\infty < 1\)

The velocity potential for the \(x\) axis dominated flow is

\[\varphi(x, y) = u_\infty x + \varphi'(x, y) \quad (78)\]

Velocity components are then given as

\[u_x(x, y) = \frac{\partial \varphi'}{\partial x} = u_\infty + u_x(x, y), \quad (79)\]

\[u_y(x, y) = \frac{\partial \varphi'}{\partial y} = u_y(x, y) \quad (80)\]
VoF – Volume of Fluid

\[ \rho = \alpha \rho_l + (1 - \alpha) \rho_g \]  
\[ \mu = \alpha \mu_l + (1 - \alpha) \mu_g \]  

(81a) \hspace{1cm} (81b)

\( \alpha \) is a volume fraction

\[ \alpha = \begin{cases} 
1, & \text{liquid;} \\
0, & \text{gas;} \\
0 \leq \alpha \leq 1, & \text{interface} 
\end{cases} \]  

(82)
VoF method

Mass conservation

\[ \nabla \cdot \mathbf{u} = 0 \]  
(83)

the Navier-Stokes equation

\[ \frac{\partial}{\partial t} (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \sigma \kappa \nabla \alpha - \nabla p_{\text{rgh}} - \mathbf{g} \cdot \mathbf{h} \nabla \rho + \nabla \cdot (2\mu \mathbf{D}) \]  
(84)

modified pressure \( p_{\text{rgh}} = p - \rho \mathbf{g} \cdot \mathbf{h} \)

\( \alpha \) transport equation

\[ \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) = 0 \]  
(85)
Complex phenomena

- Free surface flows
- Cavitation
- Melting
Free surface flows

Mixture mass conservation equation

\[ \nabla \cdot \mathbf{u} = 0 \quad (86) \]

Mixture Navier-Stokes equation

\[ \frac{\partial}{\partial t} (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \sigma_k \nabla \alpha \\
\geq 0, \neq 0 \]

\[ - \nabla p_{rgh} - \mathbf{g} \cdot \mathbf{h} \nabla \rho + \nabla \cdot (2\mu\mathbf{D}) \quad (87) \]

Volume fraction transport equation

\[ \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) = 0 \quad (88) \]
Mixture mass conservation equation
\[ \nabla \cdot \mathbf{u} = 0 \] (89)

Mixture Navier-Stokes equation
\[
\frac{\partial}{\partial t} (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \sigma \kappa \nabla \alpha
\]
\[= 0, \neq 0 \]
\[- \nabla p_{rgh} - \mathbf{g} \cdot \mathbf{h} \nabla \rho + \nabla \cdot (2\mu \mathbf{D}) \] (90)

Volume fraction transport equation
\[
\frac{\partial \alpha_l}{\partial t} + \nabla \cdot (\alpha_l \mathbf{u}) = \frac{S_m}{\rho_l} \] (91)

- Merkle $S_m = \ldots$
- Kunz
- Schnerr-Sauer
Melting

Mixture mass conservation equation

\[ \nabla \cdot \mathbf{u} = 0 \quad (92) \]

Mixture Navier-Stokes equation

\[
\frac{\partial}{\partial t} (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \rho_0 \mathbf{g} - \rho_0 \beta (T - T_0) \mathbf{g} \\
- \nabla p + \nabla \cdot (2\mu \mathbf{D}) + \mathbf{S}_u \quad (93)
\]

Volume fraction transport equation

\[
\frac{\partial \alpha_l}{\partial t} + \nabla \cdot (\alpha_l \mathbf{u}) = \frac{S_m}{\rho_l} \quad (94)
\]

Enthalpy equation

\[
\frac{\partial}{\partial t} (\rho c_p T) + \nabla \cdot (\rho c_p T \mathbf{u}) = \nabla \cdot (\lambda \nabla T) + S_h \quad (95)
\]
Finite Difference Method
The finite difference method (introduced by Euler in XVIII century) replaces the region by a finite mesh of points at which the dependent variable is approximated. All partial derivatives at each mesh point are approximated from neighbouring values by means of Taylor’s theorem. This means that derivatives at each point are approximated by difference quotients.
Taylor’s theorem

Assuming that \( f \) has continuous derivatives over certain interval the Taylor expansion is used

\[
f(x_0 + \Delta x) = \sum_{n=0}^{m-1} \frac{d^n f(x_0)}{n!} + \frac{d^m f(c)}{m!}.
\] (96)

where \( x := x_0 + \Delta x \), \( c = x_0 + \theta \Delta x \) and \( \theta \in ]0; 1[ \).

The above equation may also be written as

\[
f(x_0 + \Delta x) = f(x_0) + f'(x_0) \Delta x + \frac{1}{2} f''(x_0) \Delta x^2 \\
+ \frac{1}{6} f'''(x_0) \Delta x^3 + \ldots + \frac{1}{m!} f^{(m)}(c) \Delta x^m.
\] (97)
Instead of \( f^{(m)} \) at unknown point \( c \) it is rewritten in terms of another unknown quantity of order \( \Delta x^m \)

\[
f(x_0 + \Delta x) = f(x_0) + f'(x_0) \Delta x + f''(x_0) \frac{\Delta x^2}{2} + \ldots + f^{(m-1)}(x_0) \frac{\Delta x^{m-1}}{(m-1)!} + \mathcal{O}(\Delta x^m) \tag{98}
\]

Discarding (truncating) \( \mathcal{O}(\Delta x^m) \) one gets an approximation to \( f \). The error in this approximation is \( \mathcal{O}(\Delta x^m) \). Roughly speaking it says that knowing the value of \( f \) and the values of its derivatives at \( x_0 \) it is possible to write down the equation for its value at the point \( x_0 + \Delta x \).
First order finite difference

Taking under consideration the Taylor expansion up to the first derivative

\[ f(x_0 + \Delta x) = f(x_0) + f'(x_0)\Delta x + \mathcal{O}(\Delta x^2) \quad (99) \]

then neglecting \( \mathcal{O}(\Delta x) \) and rearranging gives the first order finite difference approximation to \( f'(x_0) \)

\[ f'(x_0) \approx \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}. \quad (100) \]

This approximation is called a forward approximation. Replacing \( \Delta x \) by \(-\Delta x\) in Taylor expansion one gets backward approximation

\[ f'(x_0) \approx \frac{f(x_0) - f(x_0 - \Delta x)}{\Delta x}. \quad (101) \]
Taking under consideration the Taylor expansion up to the second derivative

\[ f(x_0 + \Delta x) = f(x_0) + f'(x_0) \Delta x + f''(x_0) \frac{\Delta x^2}{2} + O(\Delta x^3) \]  

(102)

then neglecting \( O(\Delta x^2) \). Doing the same for \( -\Delta x \) and combining the two above we have

\[ f'(x_0) \approx \frac{f(x_0 + \Delta x) - f(x_0 - \Delta x)}{2\Delta x} \]  

(103)

after neglecting \( O(\Delta x^2) \). This is so called the second order central difference approximation to \( f'(x_0) \).
Higher order approximation to derivatives is also possible. This can be done by taking more terms in the Taylor expansion. Doing so up to the third we get

\[ f(x_0 + \Delta x) = f(x_0) + f'(x_0)\Delta x + \frac{1}{2}f''(x_0)\Delta x^2 + \frac{1}{6}f'''(x_0)\Delta x^3 + O(\Delta x^4). \] (104)

Replacing \( \Delta x \) for \(-\Delta x\) and combing the results then dropping \( O(\Delta x^4) \) gives the second order symmetric difference approximation to \( f'' \)

\[ f''(x_0) \approx \frac{f(x_0 + \Delta x) - 2f(x_0) + f(x_0 - \Delta x)}{\Delta x^2}. \] (105)
Selected finite differences approximation to first and second derivatives are given in the following table. These can be used to solve ordinary differential equations by replacing derivatives by their approximations.

<table>
<thead>
<tr>
<th>Approximation</th>
<th>Type</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f'(x_0)$</td>
<td>forward</td>
<td>$1^{st}$</td>
</tr>
<tr>
<td>$f'(x_0)$</td>
<td>backward</td>
<td>$1^{st}$</td>
</tr>
<tr>
<td>$f'(x_0)$</td>
<td>central</td>
<td>$2^{nd}$</td>
</tr>
<tr>
<td>$f''(x_0)$</td>
<td>symmetric</td>
<td>$2^{nd}$</td>
</tr>
</tbody>
</table>
Equations for $f'$ approximate the slope of the tangent in $x_0$ by means of chords (backward, forward and central finite difference).
The typical subscript notation is

\[ f(x_0 + m h, y_0 + n h) \equiv f_{i+m,j+n}. \quad (106) \]

Now it is possible to express selected finite differences approximations to derivatives in somewhat simpler manner

<table>
<thead>
<tr>
<th>Approximation</th>
<th>Type</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f'_i )</td>
<td>forward</td>
<td>1(^{st})</td>
</tr>
<tr>
<td>( f'_i )</td>
<td>backward</td>
<td>1(^{st})</td>
</tr>
<tr>
<td>( f''_i )</td>
<td>central</td>
<td>2(^{nd})</td>
</tr>
<tr>
<td>( f''_i )</td>
<td>symmetric</td>
<td>2(^{nd})</td>
</tr>
</tbody>
</table>
\[
\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi \mathbf{u}) = \nabla \cdot (\Gamma \nabla \phi)
\]


\text{(107)}

- **Parabolic.** Additionally, if \( \mathbf{u} = 0 \)

\[
\frac{\partial (\rho \phi)}{\partial t} = \nabla \cdot (\Gamma \nabla \phi)
\]

\text{(108)}

- **Hyperbolic,** if \( \Gamma = 0 \)

\[
\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi \mathbf{u}) = 0
\]

\text{(109)}

- **Elliptic,** if \( \frac{\partial}{\partial t} = 0 \) or at the same time \( \frac{\partial}{\partial t} = 0 \) and \( \mathbf{u} = 0 \)

\[
\nabla \cdot (\Gamma \nabla \phi) = 0
\]

\text{(110)}
If $\Gamma = 0$ and $\rho = 1$ we have

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi u) = 0$$  \hspace{1cm} (111)$$

One dimensional version

$$\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0$$  \hspace{1cm} (112)$$

The analytical solution of the above is

$$\phi(x, t) = f(x - u_x t)$$  \hspace{1cm} (113)$$
Convection equation – FTCS scheme

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0
\]

(114)

FTCS discretisation scheme (Forward Time Centred Space)

\[
\frac{\phi_{i}^{n+1} - \phi_{i}^{n}}{\Delta t} + u_x \frac{\phi_{i+1}^{n} - \phi_{i-1}^{n}}{2\Delta x} = 0
\]

(115)

or explicitly

\[
\phi_{i}^{n+1} = \phi_{i}^{n} - \frac{1}{2} \text{Co} \left( \phi_{i+1}^{n} - \phi_{i-1}^{n} \right)
\]

(116)

where Co is the Courant number

\[
\text{Co} = \frac{u_x \Delta t}{\Delta x}
\]

(117)
Convection equation – FOU scheme

\[\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \]

(118)

FOU discretisation scheme (First Order Upwind)

\[\frac{\phi^{n+1}_i - \phi^n_i}{\Delta t} + u_x \frac{\phi^n_i - \phi^n_{i-1}}{\Delta x} = 0 \]

(119)

or explicitly

\[\phi^{n+1}_i = \phi^n_i - Co (\phi^n_i - \phi^n_{i-1}) \]

(120)
Convection equation – BTCS scheme

\[ \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \]  (121)

BTCS discretisation scheme (Backward Time Centred Space)

\[ \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_{i+1}^{n+1} - \phi_{i-1}^{n+1}}{2\Delta x} = 0 \]  (122)

For comparison, FTCS scheme is

\[ \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_{i+1}^n - \phi_{i-1}^n}{2\Delta x} = 0 \]  (123)
Convection equation – Crank-Nicolson

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \quad (124)
\]

The Crank-Nicolson scheme is a combination of the FTCS and BTCS

\[
\frac{\phi^n_{i+1} - \phi^n_i}{\Delta t} + \frac{1}{2} \left( u_x \frac{\phi^n_{i+1} - \phi^n_{i-1}}{2\Delta x} + u_x \frac{\phi^{n+1}_{i+1} - \phi^{n+1}_{i-1}}{2\Delta x} \right) = 0 \quad (125)
\]
Convection equation – backward

\[ \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \]  \hspace{1cm} (126)

Second-order backward using two previous time-step values \( \phi^n_i, \phi^{n-1}_i \)

\[ \frac{3\phi^{n+1}_i - 4\phi^n_i + \phi^{n-1}_i}{2\Delta t} + u_x \frac{\phi^{n+1}_{i+1} - \phi^{n+1}_{i-1}}{2\Delta x} = 0 \]  \hspace{1cm} (127)
Convection equation – Lax-Wendroff

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \quad (128)
\]

\(\phi\) is expanded up to the third order

\[
\frac{\partial \phi_i}{\partial t} \approx \frac{\phi_{i}^{n+1} - \phi_{i}^{n}}{\Delta t} - \frac{1}{2} \frac{\partial^2 \phi_i}{\partial t^2} \Delta t \quad (129)
\]

Also, the centred space approximation is used

\[
\phi_{i}^{n+1} = \phi_{i}^{n} - \frac{1}{2} \text{Co} \left( \phi_{i+1}^{n} - \phi_{i-1}^{n} \right) + \frac{1}{2} \text{Co}^2 \left( \phi_{i-1}^{n} - 2\phi_{i}^{n} + \phi_{i+1}^{n} \right) \quad (130)
\]

K. Tesch; Numerical Methods
Convection equation – Lax-Friedrichs

\[ \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \]  \hspace{1cm} (131)

Lax-Friedrichs scheme is a modification of the FTCS scheme

\[ \phi_{i}^{n+1} = \frac{\phi_{i-1}^{n} + \phi_{i+1}^{n}}{2} - \frac{1}{2} \text{Co} \left( \phi_{i+1}^{n} - \phi_{i-1}^{n} \right) \]  \hspace{1cm} (132)

For comparison, FTCS scheme is

\[ \phi_{i}^{n+1} = \phi_{i}^{n} - \frac{1}{2} \text{Co} \left( \phi_{i+1}^{n} - \phi_{i-1}^{n} \right) \]  \hspace{1cm} (133)
If \( u = 0 \) and \( \rho = 1 \) we have

\[
\frac{\partial \phi}{\partial t} = \nabla \cdot (\Gamma \nabla \phi)
\]  \hspace{1cm} (134)

One dimensional version for \( \Gamma = \text{const} \)

\[
\frac{\partial \phi}{\partial t} = \Gamma \frac{\partial^2 \phi}{\partial x^2}
\]  \hspace{1cm} (135)
Transient diffusion equation – FTCS

\[ \frac{\partial \phi}{\partial t} = \Gamma \frac{\partial^2 \phi}{\partial x^2} \]  \hspace{1cm} (136)

FTCS discretisation scheme (Forward Time Centred Space)

\[ \frac{\phi_{i}^{n+1} - \phi_{i}^{n}}{\Delta t} = \Gamma \frac{\phi_{i+1}^{n} - 2\phi_{i}^{n} + \phi_{i-1}^{n}}{\Delta x^2} \]  \hspace{1cm} (137)

or explicitly

\[ \phi_{i}^{n+1} = \phi_{i}^{n} + \Gamma \frac{\Delta t}{\Delta x^2} \left( \phi_{i+1}^{n} - 2\phi_{i}^{n} + \phi_{i-1}^{n} \right) \]  \hspace{1cm} (138)
Transient diffusion equation – BTCS

\[ \frac{\partial \phi}{\partial t} = \Gamma \frac{\partial^2 \phi}{\partial x^2} \] \hspace{1cm} (139)

BTCS discretisation scheme (Backward Time Centred Space)

\[ \frac{\phi_{i}^{n+1} - \phi_{i}^{n}}{\Delta t} = \Gamma \frac{\phi_{i+1}^{n+1} - 2\phi_{i}^{n+1} + \phi_{i-1}^{n+1}}{\Delta x^2} \] \hspace{1cm} (140)
The Crank-Nicolson scheme is a combination of the FTCS and BTCS

\[
\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = \frac{1}{2} \left( \frac{\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n}{\Delta x^2} + \frac{\phi_{i+1}^{n+1} - 2\phi_i^{n+1} + \phi_{i-1}^{n+1}}{\Delta x^2} \right)
\]  

(142)
Transient diffusion equation – backward

\[ \frac{\partial \phi}{\partial t} = \Gamma \frac{\partial^2 \phi}{\partial x^2} \]  \hspace{1cm} (143)

Second-order backward using two previous time-step values \( \phi_i^n, \phi_i^{n-1} \)

\[ \frac{3\phi_i^{n+1} - 4\phi_i^n + \phi_i^{n-1}}{2\Delta t} = \Gamma \frac{\phi_{i+1}^{n+1} - 2\phi_i^{n+1} + \phi_{i-1}^{n+1}}{\Delta x^2} \]  \hspace{1cm} (144)
If $\mathbf{u} = 0$, $\rho = 1$ and $\frac{\partial}{\partial t} = 0$ we have

$$\nabla \cdot (\Gamma \nabla \phi) = S_\phi \tag{145}$$

Two types of equations can be distinguished for $\Gamma = \text{const}$

- Poisson equation, $S = \Gamma^{-1} S_\phi$
  $$\nabla^2 \phi = S \tag{146}$$

- Laplace equation, $S = 0$
  $$\nabla^2 \phi = 0 \tag{147}$$
Poisson equation is $\nabla^2 U_z = a$. Two dimensional versions of this equation is written as

$$\frac{\partial^2 U_z}{\partial x^2} + \frac{\partial^2 U_z}{\partial y^2} = a. \quad (148)$$

The next step would be to replace second order derivatives by symmetric finite difference approximation

$$\frac{U_{i+1,j} - 2U_{ij} + U_{i-1,j}}{h^2} + \frac{U_{ij+1} - 2U_{ij} + U_{ij-1}}{h^2} = a. \quad (149)$$

It can be rewritten to give $U_{ij}$ as a function of surrounding variables

$$U_{ij} = \frac{U_{i+1,j} + U_{i-1,j} + U_{ij+1} + U_{ij-1} - ah^2}{4}. \quad (150)$$
The domain is discretised in the $x$ and $y$ directions by means of constant mesh size $h$ (figure on the left). $U_z$ is unknown at black mesh points and known at white points from the boundary condition.

For instance the Dirichlet boundary condition specifies the values of $U_z$ directly. In this case $U_z = 0$ meaning no slip wall. If the boundary values are known then discrete Poisson equation gives a system of linear equations for $U_{ij}$. 

K. Tesch; Numerical Methods
The accuracy of results depends on the size of the mesh represented here by $h$. Mesh size should be decreased until there is no significant influence on numerical results.

The set of linear equations can be solved either directly by means of an appropriate method (Gauss elimination for instance) or indirectly by means of iterative solution methods or the relaxation method (point-Jacobi iteration)

$$U_{ij}^{n+1} = \frac{U_{i+1j}^n + U_{i-1j}^n + U_{ij+1}^n + U_{ij-1}^n - ah^2}{4}$$ (151)

or point-Gauss-Seidel (faster than point-Jacobi)

$$U_{ij}^{n+1} = \frac{U_{i+1j}^n + U_{i-1j}^{n+1} + U_{ij+1}^n + U_{ij-1}^n - ah^2}{4}.$$ (152)
Another indirect method is so called Successive Over-Relaxation method

\[
U_{ij}^{n+1} = (1 - w)U_{ij}^n + \frac{w}{4}(U_{i+1j}^n + U_{i-1j}^n + U_{ij+1}^n + U_{ij-1}^n - a h^2)
\]  

(153)

where \( w \) is a relaxation parameter. For \( w \in ]1, 2[ \) we have over-relaxation and for \( w := 1 \) this method corresponds to the point-Gauss-Seidel method. One can also consider under-relaxation method for \( w \in ]0, 1[ \).

The best choice of \( w \) value needs numerical experiments. It also depends on specific problems.
Poisson FDM pseudocode

**Data:** Read input variables and BCs

\[ w \leftarrow 1; \ n \leftarrow 1; \]

repeat

\[ R \leftarrow 0; \]

for \( i \leftarrow 1 \) to \( i_{\text{max}} \) do

for \( j \leftarrow 1 \) to \( j_{\text{max}} \) do

if not boundary \((U_{i,j}^n)\) then

\[
U_{i,j}^{n+1} \leftarrow \frac{U_{i+1,j}^n + U_{i-1,j}^n + U_{i,j+1}^n + U_{i,j-1}^n - ah^2}{4};
\]

\[ R \leftarrow \max \left( |U_{i,j}^{n+1} - U_{i,j}^n|, R \right); \]

\[ U_{i,j}^{n+1} \leftarrow (1 - w)U_{i,j}^n + w U_{i,j}^{n+1}; \]

\[ n \leftarrow n + 1; \]

until \( n \leq n_{\text{max}} \) and \( R > R_{\text{min}} \);
Results - Poisson equation

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
Laplace equation is $\nabla^2 \varphi = 0$. Two dimensional versions of this equation is written as

\[
\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = 0. \tag{154}
\]

Replacing second order derivatives by symmetric finite difference approximation

\[
\frac{\varphi_{i+1,j} - 2\varphi_{i,j} + \varphi_{i-1,j}}{h^2} + \frac{\varphi_{i,j+1} - 2\varphi_{i,j} + \varphi_{i,j-1}}{h^2} = 0.
\tag{155}
\]

It can be rewritten to give $\varphi_{ij}$ as a function of surrounding variables

\[
\varphi_{ij} = \frac{\varphi_{i+1,j} + \varphi_{i-1,j} + \varphi_{i,j+1} + \varphi_{i,j-1}}{4}. \tag{156}
\]
Laplace eq. – Neumann boundary condition

Neumann boundary condition specifies values of the derivative \( \frac{\partial}{\partial n} \) of a solution \( \varphi \) on boundary \( \partial \Omega \) to fulfil

\[
\frac{\partial \varphi}{\partial n} = N(x, y)
\]  

(157)

where the normal derivatives is defined as

\[
\frac{\partial \varphi}{\partial n} = \hat{n} \cdot \nabla \varphi = n_x \frac{\partial \varphi}{\partial x} + n_y \frac{\partial \varphi}{\partial y}
\]  

(158)

and \((x, y) \in \partial \Omega\). If \( \mathbf{u} = \nabla \varphi \) we get

\[
\frac{\partial \varphi}{\partial n} = \hat{n} \cdot \mathbf{u} = n_x u_x + n_y u_y.
\]  

(159)
We have two equations for a point located on boundary ‘2’

\[
\frac{\partial^2 f}{\partial x^2} = \frac{f_{i+1j} - 2f_{ij} + f_{i-1j}}{h^2},
\]

\[
\frac{\partial f}{\partial x} = \frac{f_{i+1j} - f_{i-1j}}{2h} = N_{ij}.
\]

Point \( f_{i-1j} \) is located outside the \( \Omega \) area. Eliminating it we get

\[
\frac{\partial^2 f}{\partial x^2} = \frac{2f_{i+1j} - 2N_{ij}h - 2f_{ij}}{h^2}
\]

and

\[
f_{ij} = \frac{2f_{i+1j} + f_{i+1j} + f_{ij-1} - 2N_{ij}h}{4}.
\]
Laplace FDM pseudocode

Data: Read input variables and BCs

\[ w \leftarrow 1; \ n \leftarrow 1; \]

repeat

\[ R \leftarrow 0; \]

for \( i \leftarrow 1 \) to \( i_{max} \) do

for \( j \leftarrow 1 \) to \( j_{max} \) do

if boundary(\( \varphi_{ij}^n \)) \( \neq 0 \) then

switch \( \varphi(U_{ij}^n) \) do

\[ \text{case 1: do} \]

\[ \varphi_{ij}^{n+1} \leftarrow \frac{\varphi_{i+1j}^n + \varphi_{i-1j}^n + \varphi_{ij+1}^n + \varphi_{ij-1}^n}{4}; \]

\[ \varphi_{ij}^{n+1} \leftarrow \frac{2\varphi_{i+1j}^n + \varphi_{ij+1}^n + \varphi_{ij-1}^n - 2hN_{ij}}{4}; \]

\[ \text{...} \]

\[ R \leftarrow \max \left( |\varphi_{ij}^{n+1} - \varphi_{ij}^n|, R \right); \]

\[ \varphi_{ij}^{n+1} \leftarrow (1 - w)\varphi_{ij}^n + w\varphi_{ij}^{n+1}; \]

\[ n \leftarrow n + 1; \]

until \( n \leq n_{max} \) and \( R > R_{min} \);
Results - Laplace equation

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
The biharmonic equation is $\nabla^4 \psi \equiv \nabla^2 \cdot \nabla^2 \psi = 0$. Two dimensional versions of this equation is written as

$$\frac{\partial^4 \psi}{\partial x^4} + 2 \frac{\partial^4 \psi}{\partial x^2 \partial y^2} + \frac{\partial^4 \psi}{\partial y^4} = 0. \quad (161)$$

It is a fourth-order elliptic partial differential equation that describes creeping flows in terms of a stream function $\psi$ where the velocity components are $u_x = \frac{\partial \psi}{\partial y}$ and $u_y = -\frac{\partial \psi}{\partial x}$. The Dirichlet boundary condition specifies both: a stream function $\psi$ and its normal derivative $\frac{\partial \psi}{\partial n}$. Two conditions are needed due to the fourth order of the biharmonic equation.
Biharmonic equation - approximation to derivatives

The finite difference approximations to $\frac{\partial^4 \psi}{\partial x^4}$, $\frac{\partial^4 \psi}{\partial y^4}$ are

\[
\frac{\partial^4 \psi}{\partial x^4} = \frac{\psi_{i+2j} + \psi_{i-2j} - 4\psi_{i+1j} - 4\psi_{i-1j} + 6\psi_{ij}}{h^4},
\]

(162a)

\[
\frac{\partial^4 \psi}{\partial y^4} = \frac{\psi_{ij+2} + \psi_{ij-2} - 4\psi_{ij+1} - 4\psi_{ij-1} + 6\psi_{ij}}{h^4}.
\]

(162b)

The fourth order mixed derivative is approximated as

\[
\frac{\partial^4 \psi}{\partial x^2 \partial y^2} = \frac{\psi_{i+1j+1} + \psi_{i-1j-1} + \psi_{i-1j+1} + \psi_{i+1j-1}}{h^4} + \frac{4\psi_{ij} - 2\psi_{i+1j} - 2\psi_{i-1j} - 2\psi_{ij+1} - 2\psi_{ij-1}}{h^4}.
\]

(163)
From the discrete biharmonic equations $\psi_{ij}$ can be expressed as a function of surrounding variables

$$
\psi_{ij} = \frac{-\psi_{i+2j} - \psi_{i-2j} - \psi_{ij+2} - \psi_{ij-2} + 4\psi_{ij}}{20} \\
+ 8 \frac{\psi_{i-1j} + \psi_{ij+1} + \psi_{ij-1} + \psi_{i+1j}}{20} \\
- 2 \frac{\psi_{i+1j+1} + \psi_{i-1j-1} + \psi_{i-1j+1} + \psi_{i+1j-1}}{20}.
$$

(164)
From the below figure two purely geometric relationships arise \( \frac{\partial \psi}{\partial n} = \hat{n} \cdot \nabla \psi = -u_l, \)
\( \frac{\partial \psi}{\partial l} = \hat{l} \cdot \nabla \psi = u_n. \) For an impermeable boundary one gets \( u_n = 0 \Rightarrow \frac{\partial \psi}{\partial l} = 0. \) The general relationship between volumetric flow rate and the stream functions is
\[
\dot{V} = \int_{L} \mathbf{u} \cdot \hat{n} \, dL = \int_{L} \frac{\partial \psi}{\partial l} \, dL = \int_{L} d\psi = \psi_A - \psi_B.
\]
(165)
Biharmonic FDM pseudocode

Data: Read input variables and BCs

\( w \leftarrow 1; \quad n \leftarrow 1; \)

repeat

\( R \leftarrow 0; \)

for \( i \leftarrow 1 \) to \( i_{max} \) do

for \( j \leftarrow 1 \) to \( j_{max} \) do

if not boundary(\( U_{ij}^n \)) then

\[
U_{ij}^{n+1} \leftarrow \frac{U_{i+2j}^n - U_{i-2j}^n - U_{ij+2}^n - U_{ij-2}^n + 4U_{ij}^n}{20} + \frac{U_{i-1,j}^n + U_{ij+1}^n + U_{ij-1}^n + U_{i+1,j}^n}{8} - \frac{U_{i+1,j+1}^n + U_{i-1,j-1}^n + U_{i-1,j+1}^n + U_{i+1,j-1}^n}{2},
\]

\( R \leftarrow \max (\{|U_{ij}^{n+1} - U_{ij}^n|, R\}); \)

\( U_{ij}^{n+1} \leftarrow (1 - w)U_{ij}^n + wU_{ij}^{n+1}; \)

\( n \leftarrow n + 1; \)

until \( n \leq n_{max} \) and \( R > R_{min}; \)
Results - biharmonic equation
Convection-diffusion equation

If $\Gamma = \nu = \text{const}$ and $\rho = 1$ we have

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi u) = \nu \nabla^2 \phi \quad (166)$$

One dimensional version

$$\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = \nu \frac{\partial^2 \phi}{\partial x^2} \quad (167)$$
Convection-diffusion equation – FTCS

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = \nu \frac{\partial^2 \phi}{\partial x^2} \tag{168}
\]

Forward time centred space discretisation scheme

\[
\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_{i+1}^n - \phi_{i-1}^n}{2\Delta x} = \nu \frac{\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n}{\Delta x^2} \tag{169}
\]

or explicitly

\[
\phi_i^{n+1} = \phi_i^n - \frac{C}{2} \left( \phi_{i+1}^n - \phi_{i-1}^n \right) + \frac{1}{\text{Re}} \left( \phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n \right) \tag{170}
\]

where

\[
\text{Re} = \frac{\Delta x^2}{\nu \Delta t} \tag{171}
\]
Convection-diffusion equation – BTCS

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = \nu \frac{\partial^2 \phi}{\partial x^2} \quad (172)
\]

Backward time centred space discretisation scheme

\[
\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_{i+1}^{n+1} - \phi_{i-1}^{n+1}}{2\Delta x} = \nu \frac{\phi_{i+1}^{n+1} - 2\phi_i^{n+1} + \phi_{i-1}^{n+1}}{\Delta x^2} \quad (173)
\]

or

\[
\phi_i^{n+1} - \phi_i^n + \frac{C_o}{2} \left( \phi_{i+1}^{n+1} - \phi_{i-1}^{n+1} \right) = \frac{1}{Re} \left( \phi_{i+1}^{n+1} - 2\phi_i^{n+1} + \phi_{i-1}^{n+1} \right) \quad (174)
\]
Point 1 is located inside the $\Omega$ area

$$f_1 = \frac{hf_0 + df_2}{h + d} \quad (175)$$

Point 1 is located outside the $\Omega$ area

$$f_1 = \frac{hf_0 - df_2}{h - d} \quad (176)$$
The decomposition of $\phi_{nj}^n$ into a Fourier series is

$$\phi_{nj}^n = \sum_{m=-N}^{N} A_{nm}^n e^{i k_m j \Delta x}$$

(177)

where the wave number $k_m$ is

$$k_m = \frac{m \pi}{N \Delta x} = \frac{\theta}{\Delta x}$$

(178)

A single mode determines the time evolution of $\phi_{nj}^n$

$$\phi_{nj}^n = A_{nm}^n e^{ij \theta}$$

(179)

A numerical scheme is stable if and only if

$$\left| \frac{A_{nm}^{n+1}}{A_{nm}^n} \right| = |G| \leq 1$$

(180)
Stability analysis – FTCS

- **Convection equation**

\[ |G| = \sqrt{1 + Co^2 \sin^2 \theta} \geq 1 \quad (181) \]

- **Diffusion equation**

\[ 2 \operatorname{Re}^{-1} \leq 1 \quad (182) \]

- **Convection-diffusion equation**

\[ Co^2 \leq 2 \operatorname{Re}^{-1} \leq 1 \quad (183) \]
Stability analysis – BTCS

- Convection equation

\[ |G'|^2 = \frac{1}{1 + Co^2 \sin^2 \theta} \leq 1 \]  \hspace{1cm} (184)

- Diffusion equation

\[ |G| \leq 1 \]  \hspace{1cm} (185)

- Convection-diffusion equation

\[ |G'|^2 = \frac{1}{(1 + 4 \text{Re}^{-1} \sin^2 \frac{\theta}{2})^2 + Co^2 \sin^2 \theta} \leq 1 \]  \hspace{1cm} (186)
Stability analysis – Crank-Nicolson

- convection equation

\[ |G| = 1 \]  \hspace{1cm} \text{(187)}

- diffusion equation

\[ |G| \leq 1 \]  \hspace{1cm} \text{(188)}

- convection-diffusion equation

\[ |G| \leq 1 \]  \hspace{1cm} \text{(189)}
Using a specific discretisation scheme one obtains a discrete counterpart of the original PDE;

Taylor’s expansion is applied around $\phi_i^{n+1}$ and substituted into the specific scheme;

Resulting equation is rearranged in order to recover the original PDE;

The remaining terms are the truncation errors associated with the specific discretisation scheme;

The modified equation is actually solved rather than the original PDE;

Numerical diffusion and dispersion can be analysed now;
Model equations

Convection-diffusion equation

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = \nu \frac{\partial^2 \phi}{\partial x^2}
\]  
(190)

Korteweg-de Vries equation

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = -\varepsilon \frac{\partial^3 \phi}{\partial x^3}
\]
(191)
Convection equation – FTCS scheme

\[ \partial \phi \partial t + u_x \partial \phi \partial x = 0 \quad (192) \]

FTCS discretisation scheme (Forward Time Centred Space)

\[ \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_i^{n+1} - \phi_i^{n-1}}{2\Delta x} = 0 \quad (193) \]

The modified equation

\[ \frac{\partial \phi_i^{n+1}}{\partial t} + u_x \frac{\partial \phi_i^{n+1}}{\partial x} = -\frac{1}{2} u_x \Delta x \text{Co} \frac{\partial^2 \phi_i^{n+1}}{\partial x^2} \]

\[ -\frac{1}{6} u_x \Delta x^2 (2 \text{Co}^2 + 1) \frac{\partial^3 \phi_i^{n+1}}{\partial x^3} + \ldots \quad (194) \]
Convection equation – FTCS scheme

\( \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \) \hspace{1cm} (195)

FTCS discretisation scheme (Forward Time Centred Space)

\[ \frac{\phi_{i}^{n+1} - \phi_{i}^{n}}{\Delta t} + u_x \frac{\phi_{i+1}^{n} - \phi_{i-1}^{n}}{2\Delta x} = 0 \] \hspace{1cm} (196)

The modified equation

\[ \frac{\partial \phi_{i}^{n+1}}{\partial t} + u_x \frac{\partial \phi_{i}^{n+1}}{\partial x} = \nu_N \frac{\partial^2 \phi_{i}^{n+1}}{\partial x^2} - \varepsilon_N \frac{\partial^3 \phi_{i}^{n+1}}{\partial x^3} + \ldots \] \hspace{1cm} (197)

or

\[ \frac{\partial \phi_{i}^{n+1}}{\partial t} + u_x \frac{\partial \phi_{i}^{n+1}}{\partial x} = \nu_N \frac{\partial^2 \phi_{i}^{n+1}}{\partial x^2} \] \hspace{1cm} (198)
Convection equation – FOU scheme

\[ \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \quad (199) \]

FOU discretisation scheme (First Order Upwind)

\[ \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_i^n - \phi_{i-1}^n}{\Delta x} = 0 \quad (200) \]

The modified equation

\[ \frac{\partial \phi_i^{n+1}}{\partial t} + u_x \frac{\partial \phi_i^{n+1}}{\partial x} = \frac{1}{2} u_x \Delta x (1 - Co) \frac{\partial^2 \phi_i^{n+1}}{\partial x^2} + \]
\[ \frac{1}{6} u_x \Delta x^2 (3 \text{Co} - 2 \text{Co}^2 - 1) \frac{\partial^3 \phi_i^{n+1}}{\partial x^3} + \ldots \quad (201) \]
Convection equation – FOU scheme

\[ \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \] (202)

FOU discretisation scheme (First Order Upwind)

\[ \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_i^n - \phi_{i-1}^n}{\Delta x} = 0 \] (203)

The modified equation

\[ \frac{\partial \phi_i^{n+1}}{\partial t} + u_x \frac{\partial \phi_i^{n+1}}{\partial x} = \nu_N \frac{\partial^2 \phi_i^{n+1}}{\partial x^2} - \varepsilon_N \frac{\partial^3 \phi_i^{n+1}}{\partial x^3} + \ldots \] (204)

What if Co = 1?
Convection equation – BTCS scheme

\[ \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \]  \hspace{0.5cm} (205)

BTCS discretisation scheme (Backward Time Centred Space)

\[ \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + u_x \frac{\phi_{i+1}^{n+1} - \phi_{i-1}^{n+1}}{2\Delta x} = 0 \]  \hspace{0.5cm} (206)

The modified equation

\[ \frac{\partial \phi_i^{n+1}}{\partial t} + u_x \frac{\partial \phi_i^{n+1}}{\partial x} = \frac{1}{2} u_x \Delta x \text{Co} \frac{\partial^2 \phi_i^{n+1}}{\partial x^2} + \frac{1}{6} u_x \Delta x^2 \left( \text{Co}^2 - 1 \right) \frac{\partial^3 \phi_i^{n+1}}{\partial x^3} + \ldots \]  \hspace{0.5cm} (207)
Convection equation – BTCS scheme

\[
\frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \tag{208}
\]

BTCS discretisation scheme (Backward Time Centred Space)

\[
\frac{\phi_i^{n+1} - \phi_i^{n}}{\Delta t} + u_x \frac{\phi_{i+1}^{n+1} - \phi_{i-1}^{n+1}}{2\Delta x} = 0 \tag{209}
\]

The modified equation

\[
\frac{\partial \phi_i^{n+1}}{\partial t} + u_x \frac{\partial \phi_i^{n+1}}{\partial x} = \nu_N \frac{\partial^2 \phi_i^{n+1}}{\partial x^2} - \varepsilon_N \frac{\partial^3 \phi_i^{n+1}}{\partial x^3} + \ldots \tag{210}
\]

What if Co = 1?
Convection equation – Crank-Nicolson

\[ \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \]  (211)

The Crank-Nicolson scheme is a combination of the FTCS and BTCS

\[
\frac{\phi_{i}^{n+1} - \phi_{i}^{n}}{\Delta t} + \frac{1}{2} \left( u_x \frac{\phi_{i+1}^{n} - \phi_{i-1}^{n}}{2\Delta x} + u_x \frac{\phi_{i+1}^{n+1} - \phi_{i-1}^{n+1}}{2\Delta x} \right) = 0 \]  (212)

The modified equation

\[
\frac{\partial \phi_{i}^{n+1}}{\partial t} + u_x \frac{\partial \phi_{i}^{n+1}}{\partial x} = -\frac{1}{6} u_x \Delta x^2 \left( \frac{1}{2} \text{Co}^2 + 1 \right) \frac{\partial^3 \phi_{i}^{n+1}}{\partial x^3} + \ldots 
\]  (213)
\( \frac{\partial \phi}{\partial t} + u_x \frac{\partial \phi}{\partial x} = 0 \) \hspace{1cm} (214)

The Crank-Nicolson scheme is a combination of the FTCS and BTCS

\[
\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} + \frac{1}{2} \left( u_x \frac{\phi_{i+1}^n - \phi_{i-1}^n}{2\Delta x} + u_x \frac{\phi_{i+1}^{n+1} - \phi_{i-1}^{n+1}}{2\Delta x} \right) = 0 \hspace{1cm} (215)

The modified equation

\[
\frac{\partial \phi_i^{n+1}}{\partial t} + u_x \frac{\partial \phi_i^{n+1}}{\partial x} = -\varepsilon_N \frac{\partial^3 \phi_i^{n+1}}{\partial x^3} + \ldots \hspace{1cm} (216)
\]
Typical numerical approaches for the incompressible Navier-Stokes equations:

- $\Omega_z - \psi$ (vorticity-stream function) formulation method
- Artificial compressibility method
- Pressure/velocity correction (operator splitting methods)
  - Projection methods
  - Explicit and implicit operator splitting methods
  - Fractional step method
  - PISO, SIMPLE, PIMPLE (PIso + siMPLE)
The incompressible Navier-Stokes equations

\[
\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} \tag{217}
\]

Explicit forward difference in time

\[
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n = -\frac{1}{\rho} \nabla p^n + \nu \nabla^2 \mathbf{u}^n \tag{218}
\]

Problems:

\[
\nabla \cdot \mathbf{u}^{n+1} \neq 0, \\
p^{n+1} = ?
\]
Better idea

The incompressible Navier-Stokes equations

\[ \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u}, \quad (219a) \]
\[ \nabla \cdot \mathbf{u} = 0 \quad (219b) \]

\[ \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n = -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^n, \quad (220a) \]
\[ \nabla \cdot \mathbf{u}^{n+1} = 0 \quad (220b) \]

Problems:

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

BCs?
The incompressible Navier-Stokes equations

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

\[
\nabla \cdot \mathbf{u} = 0
\]

Semi implicit approach

\[
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n = -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^{n+1},
\]

\[
\nabla \cdot \mathbf{u}^{n+1} = 0
\]
The incompressible Navier-Stokes equations

\[
\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u}, \quad (223a)
\]

\[
\nabla \cdot \mathbf{u} = 0 \quad (223b)
\]

Semi implicit approach

\[
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \underbrace{\mathbf{u}^n \cdot \nabla \mathbf{u}^{n+1}}_{\text{linearised}} = -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^{n+1}, \quad (224a)
\]

\[
\nabla \cdot \mathbf{u}^{n+1} = 0 \quad (224b)
\]
The incompressible Navier-Stokes equations

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

(225a)

\[
\nabla \cdot \mathbf{u} = 0.
\]

(225b)

Fully implicit approach

\[
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^{n+1} \cdot \nabla \mathbf{u}^{n+1} = -\frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 \mathbf{u}^{n+1},
\]

(226a)

\[
\nabla \cdot \mathbf{u}^{n+1} = 0.
\]

(226b)
Artificial compressibility method

The incompressible Navier-Stokes equations

\[
\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u}, \quad (227a)
\]

\[
\nabla \cdot \mathbf{u} = 0. \quad (227b)
\]

Explicit forward difference in time

\[
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n = -\frac{1}{\rho_0} \nabla p^n + \nu \nabla^2 \mathbf{u}^n, \quad (228a)
\]

\[
\beta \frac{p^{n+1} - p^n}{\Delta t} + \nabla \cdot \mathbf{u}^n = 0. \quad (228b)
\]
Projection method

\[
\frac{u^* - u^n}{\Delta t} = -u^n \cdot \nabla u^n + \nu \nabla^2 u^n \tag{229}
\]

\[\nabla \cdot u^* \neq 0, \text{ BC}\]

\[
\frac{u^{n+1} - u^*}{\Delta t} = -\frac{1}{\rho} \nabla p^{n+1} \tag{230}
\]

\[\nabla \cdot u^{n+1} = 0, \neg \text{ BC}\]

\[
\nabla^2 p^{n+1} = \frac{\rho}{\Delta t} \nabla \cdot u^* \tag{231}
\]
Square cylinder

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
Square cylinder

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
Decomposition method

\[
\frac{u^* - u^n}{\Delta t} = -u^n \cdot \nabla u^n - \frac{1}{\rho} \nabla p^n + \nu \nabla^2 u^n \quad (232)
\]

\[
\nabla \cdot u^* \neq 0
\]

\[
\frac{u^{n+1} - u^n}{\Delta t} = -u^n \cdot \nabla u^n - \frac{1}{\rho} \nabla p^{n+1} + \nu \nabla^2 u^n \quad (233)
\]

\[
\nabla \cdot u^{n+1} = 0
\]

\[
u^c = u^{n+1} - u^* = -\frac{1}{\rho} \Delta t \nabla (p^{n+1} - p^n) \quad (234)
\]

\[
\nabla \cdot u^c = -\nabla \cdot u^* = -\frac{1}{\rho} \Delta t \nabla^2 (p^{n+1} - p^n) \quad (235)
\]

\[
u^{n+1} = u^* + u^c
\]
The incompressible 2D Navier-Stokes equations

\[
\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u}, \tag{236a}
\]

\[
\Omega_z = \frac{\partial u_y}{\partial x} - \frac{\partial u_x}{\partial y}. \tag{236b}
\]

2D Helmholtz equation \((u_x = \frac{\partial \psi}{\partial y}, u_y = -\frac{\partial \psi}{\partial x})\)

\[
\frac{\partial \Omega_z}{\partial t} + \mathbf{u} \cdot \nabla \Omega_z = \nu \nabla^2 \Omega_z, \tag{237a}
\]

\[
\nabla^2 \psi = -\Omega_z. \tag{237b}
\]

\[
\frac{\Omega_{z}^{n+1} - \Omega_{z}^{n}}{\Delta t} + \mathbf{u}^n \cdot \nabla \Omega_z^n = \nu \nabla^2 \Omega_z^n, \tag{238a}
\]

\[
\nabla^2 \psi^{n+1} = -\Omega_z^{n+1}. \tag{238b}
\]
Finite Volume Method
Since all the transport equation have common terms, the general transport equation for a quantity $\phi$ has the form of

$$\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi \mathbf{u}) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \quad (239)$$

Four transport effects can be summarised at least, namely

- unsteadiness $\frac{\partial (\rho \phi)}{\partial t}$,
- convection $\nabla \cdot (\rho \phi \mathbf{u})$,
- diffusion $\nabla \cdot (\Gamma \nabla \phi)$.
- overall source term $S_\phi$. In the above $\Gamma$ is the diffusivity for $\phi$. 

K. Tesch; Numerical Methods
The integral form of the transport equation over a control volume $V_P$ is now expressed as

$$\frac{d}{dt} \iiint_{V_P} \rho \phi \, dV + \iiint_{V_P} \nabla \cdot (\rho \phi \mathbf{u}) \, dV =$$

$$\iiint_{V_P} \nabla \cdot (\Gamma \nabla \phi) \, dV + \iiint_{V_P} S_\phi \, dV \quad (240)$$

where a finite volume $V_P$ and its measure is $|V_P|$. 
- the finite volume centroid $P$ located at $x_P$,
- the vector $d$ connects the centroid $P$ with its neighbour centroid $N$,
- the surface $S_f$ is oriented by means of a surface normal vector $S_f$ pointing outward and the face $S_f$ centroid is located at $x_f$. 
The order of discretisation is usually equal or higher in comparison with the order of the discretised equation. Exceptions to this rule are sometimes permitted. In order to keep the second order accuracy of spatial discretisation the following variation of $\phi$ around $P$ is assumed

$$\phi(x) = \phi_P + (x - x_P) \cdot (∇\phi)_P$$ \hspace{1cm} (241)

This can be proved by means of Taylor series expansion. Also, the unknown variable $\phi_P$, located at the centroid $x_P$ of a control volume $V_P$, is calculated as $\phi_P = \phi(x_P)$. 
In order to transform volume integrals into surface integrals the Gauss’s (divergence) theorem is applied

\[
\iiint_{V_P} \nabla \cdot \mathbf{w} \, dV = \iint_{\partial V_P} \mathbf{w} \cdot d\mathbf{S} \quad (242)
\]

where \(d\mathbf{S}\) stands for the differential of the surface area vector pointing outward. Now the general transport equation can be rewritten as

\[
\frac{d}{dt} \iiint_{V_P} \rho \phi \, dV + \iint_{\partial V_P} \rho \phi \mathbf{u} \cdot d\mathbf{S} =
\iint_{\partial V_P} \Gamma \nabla \phi \cdot d\mathbf{S} + \iiint_{V_P} S_\phi \, dV \quad (243)
\]
Volume integrals

\[ \frac{d}{dt} \iiint_{V_P} \rho \phi \, dV + \iiint_{\partial V_P} \rho \phi \mathbf{u} \cdot d\mathbf{S} = \]

\[ \iiint_{\partial V_P} \Gamma \nabla \phi \cdot d\mathbf{S} + \iiint_{V_P} S_\phi \, dV \quad (244) \]

The following definition of an average value \( \phi_P \) of the function \( \phi \) located at the centroid of \( V_P \) is assumed

\[ \phi_P = \frac{1}{|V_P|} \iiint_{V_P} \phi \, dV \quad (245) \]

The volume integral is expressed by means of the averaged value \( \bar{\phi} \) of unknown function \( \phi \) and the control volume measure \( |V_P| \). Next, the averaged value is replaced by the value at \( V_P \) centroid \( \phi_P \).
If the source term $S_\phi$, i.e. the fourth integral in the general transport equation, depends on the unknown function $\phi$ it should be linearised first

$$S_\phi(\phi) = S_C + S_P\phi$$  \hspace{1cm} (246)

Subsequently, it can be integrated similarly and the discretised source terms is now

$$\iiint_{V_P} S_\phi \, dV = S_C|V_P| + S_P|V_P|\phi_P$$  \hspace{1cm} (247)

If, however, the source term $S_\phi$ does not depend on $\phi$ the discretised form is simpler since $S_P = 0$. 

K. Tesch; Numerical Methods
For the sake of simplicity let us assume further incompressibility $\rho = \text{const}$. This assumption is also valid for gases provided that $\text{Ma} < 0.3$. Furthermore, if the control volume $V_P$ is constant in time, i.e. is not deforming, then it is now possible to express the general transport equation as

$$\frac{\rho}{dV_P} \frac{d\phi_P}{dt} |V_P| + \rho \iint_{\partial V_P} \phi \mathbf{u} \cdot d\mathbf{S} =$$

$$\iint_{\partial V_P} \Gamma \nabla \phi \cdot d\mathbf{S} + S_C |V_P| + S_P |V_P| \phi_P \quad (248)$$
Convection terms

Convection terms involving $\nabla \cdot (\rho \phi \mathbf{u})$ are already transformed by means of the Gauss’s theorem and expressed as surface integrals. The surface integral over the individual surface $S_f$ is now expressed by means of the vector $\mathbf{w}_f$ value located the face $S_f$ centroid and the surface area vector $\mathbf{S}_f$ pointing outward, namely

$$\iint_{S_f} \mathbf{w} \cdot d\mathbf{S} = \mathbf{w}_f \cdot \mathbf{S}_f \quad (249)$$

This also means that the vector $\mathbf{w}$ distribution over the surface $S_f$ is now expressed by means of a single value $\mathbf{w}_f$. 
Since the boundary \( \partial V_P \) of a control volume \( V_P \) consists of \( f \) planar surfaces \( S_f \), i.e. \( \bigcup_f S_f = \partial V_P \), the convection terms is now given by the following approximation being second order accurate

\[
\iint_{\partial V_P} \phi \mathbf{u} \cdot d\mathbf{S} = \sum_f \phi_f \mathbf{u}_f \cdot \mathbf{S}_f \quad (250)
\]

The term \( \phi_f \mathbf{u}_f \cdot \mathbf{S}_f \) is also referred to as a face flux. The general transport equations is now given by

\[
\rho \frac{d\phi_P}{dt} |V_P| + \rho \sum_f \phi_f \mathbf{u}_f \cdot \mathbf{S}_f = \iint_{\partial V_P} \Gamma \nabla \phi \cdot d\mathbf{S} + S_C |V_P| + S_P |V_P| \phi_P \quad (251)
\]
What is important, is the fact that the discretised convection term needs to be interpolated further by means of cell centred values because the values $\phi_f$ are located at the face centroids. Several methods are in common use. These include, among others:

- linear interpolation or central differencing (CD),
- upwind differencing (UD),
- blending differencing,
- second order upwind differencing (SOU) or linear upwind differencing (LUD).
Convection terms – linear interpolation

A linear distribution of $\phi$ between two points $P$ and $N$ is assumed. This leads to the following face value $\phi_f$

$$\phi_f = f_x \phi_P + (1 - f_x) \phi_N \quad (252)$$

where the weighting factor $f_x$ is a ratio of respective distances

$$f_x = \frac{\|x_d - x_N\|}{\|d\|} \quad (253)$$

This method is known to be second order accurate. Nonetheless, it may lead to non-physical oscillations for some convection dominated flows or in the presence of strong gradients.
Convection terms – upwind differencing

This interpolation depends on the flow direction $\mathbf{u}_f \cdot \mathbf{S}_f$. The face value $\phi_f$ is interpolated by means of the upstream node $P$ or $N$, depending on the flow direction, namely

$$
\phi_f = \begin{cases} 
\phi_P; & \mathbf{u}_f \cdot \mathbf{S}_f \geq 0, \\
\phi_N; & \mathbf{u}_f \cdot \mathbf{S}_f < 0
\end{cases}
$$

Boundedness of the solution is guaranteed, however, it comes at a price of having to sacrifice accuracy. This is because the numerical diffusion term is implicitly introduced. Additionally, upwind differencing is only first order accurate.
Convection terms – blending differencing

This type of interpolation tries to maintain reasonable accuracy and boundedness of the solution at the same time. Typically, it combines upwind differencing and central differencing in the following manner

\[ \phi_f = \gamma \phi_{fCD} + (1 - \gamma) \phi_{fUD} \]  

(255)

Other methods also exist. In the above equation \( \gamma \) is the so called flux limiter also referred to as a blending coefficient. The idea of flux limiter makes it possible, among others, to limit towards first order upwind in regions of rapidly changing gradients.
Convection terms – SOU or LUD

Second order upwind differencing (SOU) or linear upwind differencing (LUD). More information is required than the nearest neighbours of the control volume. This leads to higher order accuracy. The face value $\phi_f$ depends on the flow direction and is interpolated by means of two the upstream nodes $P$, $PP$ or $N$, $NN$

$$
\phi_f = \begin{cases} 
\phi_P + \frac{1}{2} (\phi_P - \phi_{PP}) ; & \mathbf{u}_f \cdot \mathbf{S}_f > 0, \\
\phi_N + \frac{1}{2} (\phi_N - \phi_{NN}) ; & \mathbf{u}_f \cdot \mathbf{S}_f < 0
\end{cases} \tag{256}
$$

LUD interpolation is second order accurate and unbounded.
Convection terms – limiters

Flux limiter formulation \((\mathbf{u}_f \cdot \mathbf{S}_f > 0)\)

\[
\phi_f = \phi_P + \frac{1}{2} \psi(r) (\phi_P - \phi_{PP})
\]  \hspace{1cm} (257)

where \(\psi\) is a limiter function. The limiter \(\psi\) is a function of gradients ratio \(r\) (1D version)

\[
r = \frac{\phi_N - \phi_P}{\phi_P - \phi_{PP}}
\]  \hspace{1cm} (258)

General 3D version

\[
r = 2 \frac{\mathbf{d} \cdot (\nabla \phi)_P}{\mathbf{d} \cdot (\nabla \phi)_f} - 1
\]  \hspace{1cm} (259)
A limiter is symmetric if

\[ \psi \left( \frac{1}{r} \right) = \frac{\psi(r)}{r} \]  

(260)

TVD conditions:
- \( 0 \leq \psi(r) \leq 2r \),
- \( 0 \leq \psi(r) \leq 2 \)

Second order TVD conditions:
- \( 0 \leq \psi(r) \leq 2r \), \( r \in [0; 1] \),
- \( 1 \leq \psi(r) \leq r \), \( r \in [1; 2] \),
- \( 0 \leq \psi(r) \leq 2 \), \( r > 2 \),
- \( \psi(1) = 1 \)
Convection terms – NVF

Normalised Variables Formulation

\[ \tilde{\phi}_C = \frac{\phi_C - \phi_U}{\phi_D - \phi_U} \quad (261) \]

\[ \tilde{\phi}_f = \frac{\phi_f - \phi_U}{\phi_D - \phi_U} \quad (262) \]
NVD conditions:

- $\tilde{\phi}_f$ is continuous,
- $\tilde{\phi}_C \leq \tilde{\phi}_f(\tilde{\phi}_C) \leq 1$, $\tilde{\phi}_C \in [0; 1]$,
- $\tilde{\phi}_f(\tilde{\phi}_C) = \tilde{\phi}_C$, $\tilde{\phi}_C \notin [0; 1]$

Second (or above) order NVD conditions:

- all above,
- $\tilde{\phi}_f(0.5) = 0.75$
Convection terms – example

\[ \frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi \mathbf{u}) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \] (263)

\[ \frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathbf{u}) = 0 \] (264)
Upwind Differencing (First Order Upwind)

\[ \psi(r) = 0 \]

- TVD, NVD,
- 1\textsuperscript{st} order

\[ \tilde{\phi}_f(\tilde{\phi}_C) = \tilde{\phi}_C \]
Convection terms – UD
Convection terms – SOU

Second Order Upwind (Linear Upwind Differencing)

\[ \psi(r) = 1 \]

- Not TVD, not NVD,
- 2nd order

\[ \tilde{\phi}_f(\tilde{\phi}_C) = \frac{3}{2} \tilde{\phi}_C \]
Convection terms – SOU

Contents

Description of fluid/solid at different scales

Finite Difference Method

Finite Volume Method

Finite Element Method

Monte Carlo Method

Smoothed Particle Hydrodynamics

Lattice Boltzmann Method

Turbulence modelling

References
Quadratic Upwind Interpolation for Convective Kinematics (Quadratic Upwind Differencing)

For a uniform mesh

\[
\phi_f = \frac{3}{8} \phi_D + \frac{3}{4} \phi_C - \frac{1}{8} \phi_U
\]

- not NVD,
- 3\textsuperscript{rd} order

\[
\tilde{\phi}_f(\tilde{\phi}_C) = \frac{3}{8} \left( 1 + 2\tilde{\phi}_C \right)
\]
Convection terms – QUICK

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References

K. Tesch; Numerical Methods
Convection terms – LD

Linear Differencing (Central Differencing)

\[ \psi(r) = r \]

- Not TVD, not NVD,
- 2nd order

\[ \tilde{\phi}_f(\tilde{\phi}_C) = \frac{1}{2} \left( 1 + \tilde{\phi}_C \right) \]
Convection terms – LD

Contents
Description of fluid/solid at different scales
Finite Difference Method
Finite Volume Method
Finite Element Method
Monte Carlo Method
Smoothed Particle Hydrodynamics
Lattice Boltzmann Method
Turbulence modelling
References

K. Tesch; Numerical Methods

163
Convection terms – minmod

\[ \psi(r) = \max(0, \min(1, r)) \]

- TVD, NVD,
- 2nd order,
- piecewise linear

\[
\tilde{\phi}_f(\tilde{\phi}_C) = \begin{cases} 
\frac{3}{2} \tilde{\phi}_C; & 0 \leq \tilde{\phi}_C < \frac{1}{2}, \\
\frac{1}{2} \left(1 + \tilde{\phi}_C\right); & \frac{1}{2} \leq \tilde{\phi}_C \leq 1, \\
\tilde{\phi}_C; & \text{otherwise}
\end{cases}
\]
Convection terms – minmod
Convection terms – superbee

\[ \psi(r) = \max(0, \min(2r, 1), \min(r, 2)) \]

- TVD,
- 2\textsuperscript{nd} order,
- symmetric.
- piecewise linear
Convection terms – superbee
Upstream Monotonic Interpolation Scalar Transport

\[ \psi(r) = \max(0, \min(2r, \frac{1}{4} + \frac{3}{4} r, \frac{3}{4} + \frac{1}{4} r, 2)) \]

- TVD,
- 2\textsuperscript{nd} order,
- \( \lim_{r \to \infty} \psi(r) = 2 \),
- piecewise linear
Contents
Description of fluid/solid at different scales
Finite Difference Method
Finite Volume Method
Finite Element Method
Monte Carlo Method
Smoothed Particle Hydrodynamics
Lattice Boltzmann Method
Turbulence modelling
References
Monotonic Upwind Scheme for Conservation Laws

\[ \psi(r) = \max(0, \min(2r, \frac{1}{2} + \frac{1}{2}r, 2)) \]

- TVD, NVD,
- 2\textsuperscript{nd} order,
- \( \lim_{r \to \infty} \psi(r) = 2 \),
- piecewise linear

\[ \tilde{\phi}_f(\tilde{\phi}_C) = \begin{cases} 
2\tilde{\phi}_C; & 0 \leq \tilde{\phi}_C < \frac{1}{4}, \\
\frac{1}{4} + \tilde{\phi}_C; & \frac{1}{4} \leq \tilde{\phi}_C < \frac{3}{4}, \\
1; & \frac{3}{4} \leq \tilde{\phi}_C \leq 1, \\
\tilde{\phi}_C; & \text{otherwise}
\end{cases} \]
Convection terms – MUSCL

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References

K. Tesch; Numerical Methods
Convection terms – van Leer

\[ \psi(r) = \frac{r + |r|}{1 + |r|} \]

- TVD, NVD,
- 2\textsuperscript{nd} order,
- symmetric,
- \( \lim_{r \to \infty} \psi(r) = 2 \)

\[ \tilde{\phi}_f(\tilde{\phi}_C) = \begin{cases} \tilde{\phi}_C \left( 2 - \tilde{\phi}_C \right); & 0 \leq \tilde{\phi}_C \leq 1, \\ \tilde{\phi}_C; & \text{otherwise} \end{cases} \]
Convection terms – van Leer
Diffusion terms involving $\nabla \cdot (\Gamma \nabla \phi)$ are treated in the same way as convection terms

$$\iint_{\partial V_P} \Gamma \nabla \phi \cdot dS = \sum_f \Gamma_f (\nabla \phi)_f \cdot S_f \quad (265)$$

Most importantly, the discretised diffusion term needs to be interpolated further by means of cell centred values. The spatially discretised general transport equations is now given by

$$\rho \frac{d\phi_P}{dt} |V_P| + \rho \sum_f \phi_f \mathbf{u}_f \cdot \mathbf{S}_f = \sum_f \Gamma_f (\nabla \phi)_f \cdot \mathbf{S}_f + S_C |V_P| + S_P |V_P| \phi_P \quad (266)$$
If the considered mesh is orthogonal the dot product of the face centred gradient \((\nabla \phi)_f\) and surface normal vector \(S_f\), being in fact the surface normal gradient, is calculated according to the following equation

\[
(\nabla \phi)_f \cdot S_f = \frac{\phi_N - \phi_P}{\|d\|} \|S_f\| \quad (267)
\]

which takes under considerations two two centroids \(P\) and \(N\) values around the surface \(S_f\). This approach is the central difference approximation of the first order derivative and is know to be second order accurate.
For generic non-orthogonal meshes a correction is introduced. The two contributing parts are considered based upon the following decomposition \( S_f = S_\perp + k \). Here, \( S_\perp \) is parallel with \( d \). Finally, the corrected equation is

\[
(\nabla \phi)_f \cdot S_f = \frac{\phi_N - \phi_P}{\|d\|} \|S_\perp\| + (\nabla \phi)_f \cdot k
\]  

(268)

The right hand side of the above formula represents the orthogonal and non-orthogonal contributions. The latter requires the face centred gradient interpolation. This is usually achieved by the linear interpolation of cell centred gradients \((\nabla \phi)_P\) and \((\nabla \phi)_N\), i.e.

\[
(\nabla \phi)_f = f_x (\nabla \phi)_P + (1 - f_x) (\nabla \phi)_N
\]

(269)
\[(\nabla \phi)_f = f_x (\nabla \phi)_P + (1 - f_x) (\nabla \phi)_N \quad (270)\]

Two most commonly met methods of cell centred gradients evaluations are Gaussian integration and least squares method:

- Gaussian integration,
- least squares method.
Diffusion terms – Gaussian integration

Gaussian integration. Cell centred gradients are evaluated by means of the Gauss’s theorem for \( \mathbf{w} = \phi \mathbf{c} \) where \( \mathbf{c} \) stands for a constant vector

\[
\int \int \int_{V_P} \nabla \phi \, dV = \int \int_{\partial V_P} \phi \, d\mathbf{S}
\]

These terms are converted by means of Gaussian integration. Secondly, the average value of \( \nabla \phi \) is replaced by the cell centred value \((\nabla \phi)_P\)

\[
(\nabla \phi)_P = \frac{1}{|V_P|} \int \int \int_{V_P} \nabla \phi \, dV =
\]

\[
\frac{1}{|V_P|} \int \int_{\partial V_P} \phi \, d\mathbf{S} = \frac{1}{|V_P|} \sum_f \phi_f S_f \quad (272)
\]
Diffusion terms – least squares method

Equation

\[ \phi_N = \phi_P + (\mathbf{x}_N - \mathbf{x}_P) \cdot (\nabla \phi)_P \]  

(273)

allows for extrapolation of the values \( \phi_P \) by means of their gradients \( (\nabla \phi)_P \) to the neighbouring points \( \phi_N \). Introducing the following vector connecting point \( P \) with its neighbours \( N \), namely \( \mathbf{d}_N = \mathbf{x}_N - \mathbf{x}_P \), it is now possible to provide \( N \) equations

\[ \mathbf{d}_N \cdot (\nabla \phi)_P = \phi_N - \phi_P \]  

(274)

where \( N \) stands for the total number of neighbours of \( P \) and is always larger or equal four. This is because the simplest polyhedral volume consists of four faces.
Since $N$ is larger than the three components of the gradient $(\nabla \phi)_P$ to be computed, it is necessary to minimise the sum of the square of weighted errors for all $N$ neighbours. In order to find the gradient $(\nabla \phi)_P$ a linear system of equations is formulated

$$A \cdot (\nabla \phi)_P = y \quad (275)$$

where known $N \times 3$ matrix is $A = (d_N, \ldots)^T$, the unknown $3 \times 1$ gradient is $(\nabla \phi)_P$ and finally the known $N \times 1$ vector is $y = (\phi_N - \phi_P, \ldots)^T$. 
Diffusion terms – least squares method

The sum of squared residuals or the norm to be minimised is defined by

\[ \| A \cdot (\nabla \phi)_P - y \|^2 = \sum_N (d_N \cdot (\nabla \phi)_P - (\phi_N - \phi_P))^2 \quad (276) \]

Finally, the unknown gradient \((\nabla \phi)_P\) is the solution of the following linear equations system

\[ (A^T \cdot A) \cdot (\nabla \phi)_P = A^T \cdot y \quad (277) \]
Temporal discretisation

Time dependent (transient) problems require temporal discretisation of the general transport equation. Integration it with respect to time from $t$ to $t + \Delta t$ results in

\[
\rho |V_P| \int_t^{t+\Delta t} \frac{d\phi_P}{dt} \, dt = \int_t^{t+\Delta t} \left( -\rho \sum_f \phi_f \mathbf{u}_f \cdot \mathbf{S}_f \right) \, dt + \\
\int_t^{t+\Delta t} \sum_f \Gamma_f (\nabla \phi)_f \cdot \mathbf{S}_f \, dt + \int_t^{t+\Delta t} (S_C |V_P| + S_P |V_P| \phi_P) \, dt
\]

(278)

The right hand side of the above represents the time integral of all the spatial values.
The following abbreviations are assumed:

- the new value \( \phi_P^{n+1} = \phi_P(t + \Delta t) \), i.e. value the solver is calculating for,
- old value \( \phi_P^n = \phi_P(t) \), i.e. known from the previous time step,
- old old value \( \phi_P^{n-1} = \phi_P(t - \Delta t) \) known from the time step prior to the previous.
Temporal discretisation

The left hand side of the discretised equation can be evaluated directly and the right hand side integrand is denoted as \( f(\phi_f, \phi_P) \)

\[
\rho \left( \phi^P_{n+1} - \phi^P_n \right) |V_P| = \int_t^{t+\Delta t} f(\phi_f(t), \phi_P(t)) \, dt \quad (279)
\]

This time, however, the right hand side cannot be integrated directly. This means that it has to be approximated by \( F(\phi_f, \phi_P) \Delta t \). Above equation now reads

\[
\rho \frac{\phi^P_{n+1} - \phi^P_n}{\Delta t} |V_P| = F(\phi_f, \phi_P) \quad (280)
\]
Several treatments of the spatial derivatives in a transient problem are possible. The most popular are:

- explicit Euler,
- implicit Euler,
- Crank-Nicolson (linear interpolation),
- backward differencing.

Euler methods as well as Crank-Nicolson method require only values of the unknown function at two different times $\phi_{P}^{n+1}$ and $\phi_{P}^{n}$ and are referred to as two-level methods.

Backward differencing is the so-called three-level method because it requires the values of the unknown function $\phi_{P}$ at three different times, namely $\phi_{P}^{n+1}$, $\phi_{P}^{n}$, and $\phi_{P}^{n-1}$.
Temporal discretisation – explicit Euler

The right hand side of previous equation is approximated explicitly by means of old values \( \phi^n \) which is denoted as \( F(\phi^n_f, \phi^n_P) \). The discrete version of the general transport equation is

\[
\rho \frac{\phi_P^{n+1} - \phi_P^n}{\Delta t} |V_P| + \rho \sum_f \phi^n_f u_f \cdot S_f = \\
\sum_f \Gamma_f (\nabla \phi)_f^n \cdot S_f + S_C |V_P| + S_P |V_P| \phi^n_P
\] (281)

The method is first order accurate in time. However, it is also unstable if Co > 1. Despite this, explicit Euler discretisation is very easy to implement and does not require substantial computer resources.
Temporal discretisation – implicit Euler

This time the right hand side of previous equation is approximated implicitly by means of current values $\phi^{n+1}$ which is denoted as $F(\phi^+_{f}, \phi^+_{P})$. The discrete version of the general transport equation is

$$\rho \frac{\phi^{n+1}_P - \phi^n_P}{\Delta t} |V_P| + \rho \sum_{f} \phi^{n+1}_f u_f \cdot S_f = $$

$$\sum_{f} \Gamma_f (\nabla \phi)_f^{n+1} \cdot S_f + S_C |V_P| + S_P |V_P| \phi^{n+1}_P$$ (282)

The method is first order accurate in time and is unconditionally stable in contrast to explicit method. Implicit Euler discretisation is more complicated to implement in comparison with its explicit formulation and requires iterative approach.
Temporal discretisation – Crank-Nicolson

Formally, the method utilises the trapezoid rule. This is equivalent to an arithmetical average of current and old values

\[ \rho \frac{\phi_P^{n+1} - \phi_P^n}{\Delta t} |V_P| = \frac{F (\phi_f^n, \phi_P^n) + F (\phi_f^{n+1}, \phi_P^{n+1})}{2} \]  

(283)

The method is known to be second order accurate in time and unconditionally stable.
Temporal discretisation – backward differencing

The integrand $\frac{d\phi_P}{dt}$ of the left hand side of the previous equation is discretised first rather than being evaluated directly

$$\frac{d\phi_P}{dt} = \frac{3\phi_P^{n+1} - 4\phi_P^n + \phi_P^{n-1}}{2\Delta t}$$  \hspace{1cm} (284)

Now the left hand side of the previous equation can be integrated. The implicit discrete version of the general transport equation is

$$\rho \frac{3\phi_P^{n+1} - 4\phi_P^n + \phi_P^{n-1}}{2\Delta t} |V_P| + \rho \sum_f \phi_f^{n+1} u_f \cdot S_f = \sum_f \Gamma_f (\nabla \phi)_f^{n+1} \cdot S_f + S_C |V_P| + S_P |V_P| \phi_P^{n+1}$$  \hspace{1cm} (285)
Let us consider differential form of the general transport equation

\[
\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \phi \mathbf{u}) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \quad (286)
\]

To obtain the integral form of this equation one needs Gauss’s (divergence) theorem. Two dimensional version has the following form

\[
\iint_{\Omega_i} \nabla \cdot \mathbf{w} \, d\Omega = \oint_{\partial \Omega_i} \mathbf{w} \cdot d\mathbf{L} \quad (287)
\]

where \( d\Omega \equiv dx \, dy \) and \( d\mathbf{L} \equiv \hat{n} \, dL \equiv \hat{i} \, dy - \hat{j} \, dx \)
Integrating over the two dimensional domain (finite ‘volume’) $\Omega_i$ and utilising Gauss’s theorem results in

$$\frac{d}{dt} \iiint_{\Omega_i} \rho \phi \, d\Omega + \int_{\partial \Omega_i^+} \rho \phi \mathbf{u} \cdot \, d\mathbf{L} = \int_{\partial \Omega_i^+} \Gamma \nabla \phi \cdot \, d\mathbf{L} + \iint_{\Omega_i} S \phi \, d\Omega \quad (288)$$

First and last integral in the above equation suggest the following definition of an average $\bar{\phi}_i$ value of $f$ over $\Omega_i$

$$\bar{\phi}_i = \frac{1}{|\Omega_i|} \iint_{\Omega_i} \phi \, d\Omega \quad (289)$$

The average value $\bar{\phi}_i$ is typically located at the centre of the volume $\Omega_i$. 

K. Tesch; Numerical Methods
The next step would be the spatial discretisation over the volume $\Omega_i$ boundary $\partial \Omega_i$. The line integral represents the total flux out of volume $\Omega_i$ and is replaced by a sum

$$\int_{\partial \Omega_i^+} \mathbf{w} \cdot d\mathbf{L} \approx \sum_k \mathbf{w}_k \cdot \Delta \mathbf{L}_k \quad (290)$$

Boundary $\partial \Omega_i$ consists of lines indexed by subscript $k$. There are at least three lines (triangle). The vector $\mathbf{w}$ is either $\phi \mathbf{u}$ or $\Gamma \nabla \phi$. Because that vector $\mathbf{w}$ is typically not constant along each line it has to be approximated by a single value $\mathbf{w}_k$ at the centre of each line.
The last step would be time discretisation. Among many possibilities the simplest is the first order forward finite difference approximation

\[
\frac{d\bar{\phi}_i}{dt} \approx \frac{\bar{\phi}_i^{n+1} - \bar{\phi}_i^n}{\Delta t} \tag{291}
\]

Time step of this approximation is denoted here as \(\Delta t\). Finally, one gets the following discretised version of transport equation (i.e. Finite Volume Scheme)

\[
\rho \frac{\bar{\phi}_i^{n+1} - \bar{\phi}_i^n}{\Delta t} |\Omega_i| + \rho \sum_k (\phi u)_k \cdot \Delta L_k = \sum_k (\Gamma \nabla \phi)_k \cdot \Delta L_k + \bar{S}_{\phi_i} |\Omega_i| \tag{292}
\]

\( |\Omega_i| \) stands for the area of control volume \(\Omega_i\)
Two dimensional and steady state diffusion equation of a $f$ quantity arises, as previously, from the general transport equation (convection-diffusion equation)

$$\nabla \cdot (\Gamma \nabla \phi) + S_\phi = 0$$  \hspace{1cm} (293)

If the diffusion coefficient is constant $\Gamma = 1$ and the source term $S_\phi = a$ then the above equation simplifies to

$$\nabla \cdot \nabla \phi + a = 0$$  \hspace{1cm} (294)

or $\nabla^2 \phi = -a$ which is Poisson equation.
2D FVM diffusion problem

The discrete version of the diffusion equation (simplified version of general transport equation) is

\[
\sum_k (\Gamma \nabla \phi)_k \cdot \Delta L_k + \bar{S}_{\phi ij} |\Omega_{ij}| = 0
\]  

(295)

For a structural and Cartesian mesh (next slide) the normal \( \Delta L_k \) vectors are

\[
\Delta L_{AB} = |AB| \hat{i} = \Delta y_i \hat{i} \quad (296a)
\]

\[
\Delta L_{BC} = |BC| \hat{j} = \Delta x_i \hat{j} \quad (296b)
\]

\[
\Delta L_{CD} = |CD| (-\hat{i}) = -\Delta y_i \hat{i} \quad (296c)
\]

\[
\Delta L_{DA} = |DA| (-\hat{j}) = -\Delta x_i \hat{j} \quad (296d)
\]
2D FVM diffusion problem

\[ \phi_{i-1,j+1} \quad \phi_{i,j+1} \quad \phi_{i+1,j+1} \]

\[ \phi_{i-1,j} \quad \phi_{ij} \quad \phi_{i+1,j} \]

\[ \phi_{i-1,j-1} \quad \phi_{ij-1} \quad \phi_{i+1,j-1} \]

\[ \Delta L_{BC} \quad \Delta L_{CD} \quad \Delta L_{DA} \quad \Delta L_{AB} \]
The discrete version of two dimensional diffusion equation is now

$$
\Gamma \frac{\partial \phi_{i+\frac{1}{2},j}}{\partial x} \Delta y_i + \Gamma \frac{\partial \phi_{ij+\frac{1}{2}}}{\partial y} \Delta x_i - \Gamma \frac{\partial \phi_{i-\frac{1}{2},j}}{\partial x} \Delta y_i - \Gamma \frac{\partial \phi_{ij-\frac{1}{2}}}{\partial y} \Delta x_i + \bar{S}_{\phi_{ij}} \Omega_{ij} = 0 \quad (297)
$$

where the area of volume $\Omega_{ij}$ is $|\Omega_{ij}| = \Delta x_i \Delta y_i$ and the diffusion coefficient is constant $\Gamma = 1$. If so, then

$$
\frac{\partial \phi_{i+\frac{1}{2},j}}{\partial x} \Delta y_i + \frac{\partial \phi_{ij+\frac{1}{2}}}{\partial y} \Delta x_i - \frac{\partial \phi_{i-\frac{1}{2},j}}{\partial x} \Delta y_i - \frac{\partial \phi_{ij-\frac{1}{2}}}{\partial y} \Delta x_i + \bar{S}_{\phi_{ij}} \Omega_{ij} = 0 \quad (298)
$$
The average value of $S_\phi$ at the centre of finite volume is approximated by means of known values of $S_\phi$ at the boundary of finite volume

$$\bar{S}_{\phi ij} = \frac{1}{4} \left( S_{\phi i \frac{1}{2} j \frac{1}{2}} + S_{\phi i \frac{1}{2} j \frac{1}{2}} + S_{\phi i \frac{1}{2} j \frac{1}{2}} + S_{\phi i \frac{1}{2} j \frac{1}{2}} \right)$$  \hspace{1cm} (299)$$

Derivatives at the boundary of finite volume are approximated by means of the second order scheme as

$$\frac{\partial \phi_{i+\frac{1}{2}j}}{\partial x} \approx \frac{\phi_{i+1j} - \phi_{ij}}{x_{i+1j} - x_{ij}} = \frac{\phi_{i+1j} - \phi_{ij}}{\Delta x_{i+1}} \hspace{1cm} (300a)$$

$$\frac{\partial \phi_{ij+\frac{1}{2}}}{\partial y} \approx \frac{\phi_{ij+1} - \phi_{ij}}{y_{ij+1} - y_{ij}} = \frac{\phi_{ij+1} - \phi_{ij}}{\Delta y_{j+1}} \hspace{1cm} (300b)$$
2D FVM diffusion problem

\[
\frac{\partial \phi_{i-\frac{1}{2}j}}{\partial x} \approx \frac{\phi_{ij} - \phi_{i-1j}}{x_{ij} - x_{i-1j}} = \frac{\phi_{ij} - \phi_{i-1j}}{\Delta x_{i-1}}
\]  \hspace{1cm} (301a)

\[
\frac{\partial \phi_{ij-\frac{1}{2}}}{\partial y} \approx \frac{\phi_{ij} - \phi_{ij-1}}{y_{ij} - y_{ij-1}} = \frac{\phi_{ij} - \phi_{ij-1}}{\Delta y_{j-1}}
\]  \hspace{1cm} (301b)

The specific form of a finite volume scheme is now

\[
\bar{S}_{\phi ij} \Delta x_i \Delta y_i + \frac{\phi_{i-1j} \Delta y_i}{\Delta x_{i-1}} + \frac{\phi_{i+1j} \Delta y_i}{\Delta x_{i+1}} + \frac{\phi_{ij-1} \Delta x_i}{\Delta y_{j-1}} + \frac{\phi_{ij+1} \Delta x_i}{\Delta y_{j+1}} - \\
\phi_{ij} \left( \frac{\Delta y_i}{\Delta x_{i-1}} + \frac{\Delta y_i}{\Delta x_{i+1}} + \frac{\Delta x_i}{\Delta y_{j-1}} + \frac{\Delta x_i}{\Delta y_{j+1}} \right) = 0
\]  \hspace{1cm} (302)
2D FVM diffusion problem

It can also be rewritten to give $\phi_{ij}$ as a function of surrounding variables

$$
\phi_{ij} = \frac{\phi_{i-1j} \Delta y_i}{\Delta x_{i-1}} + \frac{\phi_{i+1j} \Delta y_i}{\Delta x_{i+1}} + \frac{\phi_{ij-1} \Delta x_i}{\Delta y_{j-1}} + \frac{\phi_{ij+1} \Delta x_i}{\Delta y_{j+1}} + \bar{S}_{\phi_{ij}} \Delta x_i \Delta y_i
$$

(303)

For

$$
\Delta x_i = \Delta y_i = \Delta x_{i-1} = \Delta x_{i+1} = \Delta y_{i-1} = \Delta y_{i+1} = h
$$

(i.e. uniform mesh) the finite volume scheme reduced to a finite difference scheme for Poisson equation

$$
\phi_{ij} = \frac{\phi_{i-1j} + \phi_{i+1j} + \phi_{ij-1} + \phi_{ij+1} + \bar{S}_{\phi_{ij}} h^2}{4}
$$

(304)
Data: Read volumes data and BCs
Create nodes and ghost nodes;
\( n := 1; \)
repeat
\( R := 0; \)
for \( i := 2 \) to \( i_{\text{max}} - 1 \) do
  for \( j := 2 \) to \( j_{\text{max}} - 1 \) do
    \( \phi_{ij}^{n+1} := \)
    \[ \frac{\phi_{i-1,j}^n \Delta y_i}{\Delta x_{i-1}} + \frac{\phi_{i+1,j}^n \Delta y_i}{\Delta x_{i+1}} + \frac{\phi_{i,j-1}^n \Delta x_i}{\Delta y_{j-1}} + \frac{\phi_{i,j+1}^n \Delta x_i}{\Delta y_{j+1}} + S_{\phi_{ij}} \Delta x_i \Delta y_i + \]
    \[ \frac{\Delta y_i}{\Delta x_{i-1}} + \frac{\Delta y_i}{\Delta x_{i+1}} + \frac{\Delta x_i}{\Delta y_{j-1}} + \frac{\Delta x_i}{\Delta y_{j+1}} \];
  \( R := \max \left( |\phi_{ij}^{n+1} - \phi_{ij}^n|, R \right); \)
Update ghost nodes;
\( n := n + 1; \)
until \( n \leq n_{\text{max}} \) and \( R > R_{\text{min}} \);
Nonuniform and uniform volumes and nodes
Results for nonuniform and uniform mesh
Results for nonuniform and uniform mesh
1D FVM diffusion problem

\[ \Delta x_{i-1} \quad \Delta x_i \quad \Delta x_{i+1} \]

\[ x_i - \frac{1}{2} \quad x_i + \frac{1}{2} \]

\[ x_{i-1} \quad x_i \quad x_{i+1} \]

\[ \Delta x_i \]

\[ \Delta x_i \]

\[ \Delta x_{1/2} \]

\[ \Delta x_{N/2} \]

\[ x_0 \quad x_1 \quad x_2 \quad x_3 = x_N \]

\[ \Delta x_1 \quad \Delta x_N \]
One dimensional and steady state diffusion equation of a \( f \) quantity arises from the general transport equation (convection-diffusion equation)

\[
\frac{\partial (\rho \phi)}{\partial t} + \nabla \cdot (\rho \mathbf{u} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \quad (305)
\]

If the diffusion coefficient \( \Gamma \) is constant then the above equation simplifies to

\[
\nabla \cdot (\Gamma \nabla \phi) + S_\phi = 0 \quad (306)
\]

or more precisely

\[
\frac{d}{dx} \left( \Gamma \frac{d\phi}{dx} \right) + S_\phi = 0 \quad (307)
\]
The integral form of one dimensional diffusion equation takes the following form

\[
\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \frac{d}{dx} \left( \Gamma \frac{d\phi}{dx} \right) \, dx + \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} S_\phi \, dx = 0 \quad (308)
\]

There is no need to take advantage of Gauss’s theorem. This is because the first term can be integrated directly

\[
\left( \Gamma \frac{d\phi}{dx} \right)_{i+\frac{1}{2}} - \left( \Gamma \frac{d\phi}{dx} \right)_{i-\frac{1}{2}} + \bar{S}_\phi \Delta x_i = 0 \quad (309)
\]

where \( \Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \).
The average value of $S_f$ at the centre of finite volume can be approximated by means of trapezoidal rule by means of known values of $S_f$ at the boundary of finite volume

$$
\bar{S}_{fi} = \frac{S_{fi-\frac{1}{2}} + S_{fi+\frac{1}{2}}}{2} \quad (310)
$$

Let us consider ODE

$$
y''(x) + 20x = 0 \quad (311)
$$

subjected to the Dirichlet boundary conditions

$y(0) = y(1) = 0$. The specific solution is

$$
y(x) = -\frac{10}{3}(x^3 - x) \quad (312)
$$
In other word, the diffusion coefficient $\Gamma = 1$ and source term $S_f = 20x$. If so, then discrete version of one dimensional diffusion equation is now

$$
\frac{d\phi_{i+1/2}}{dx} - \frac{d\phi_{i-1/2}}{dx} + \overline{S}_{\phi i} \Delta x_i = 0 \quad (313)
$$

Derivatives or diffusive fluxes at the boundary of finite volume are approximated by means of the second order scheme as

$$
\frac{d\phi_{i+1/2}}{dx} \approx \frac{\phi_{i+1} - \phi_i}{x_{i+1} - x_i} \quad (314a)
$$

$$
\frac{d\phi_{i-1/2}}{dx} \approx \frac{\phi_i - \phi_{i-1}}{x_i - x_{i-1}} \quad (314b)
$$
The specific form of a finite volume scheme is now

\[
\frac{\phi_{i+1} - \phi_i}{x_{i+1} - x_i} - \frac{\phi_i - \phi_{i-1}}{x_i - x_{i-1}} + \bar{S}_{\phi_i} \Delta x_i = 0
\]

(315)

It can also be rewritten to give \( \phi_i \) as a function of surrounding variables

\[
\phi_i = \frac{\Delta x_{i+1} \phi_{i-1} + \Delta x_{i-1} \phi_{i+1} + \bar{S}_{\phi_i} \Delta x_{i-1} \Delta x_{i+1} \Delta x_i}{\Delta x_{i-1} + \Delta x_{i+1}}
\]

(316)

where \( \Delta x_{i-1} = x_i - x_{i-1} \) and \( \Delta x_{i+1} = x_{i+1} - x_i \).

For \( \Delta x_{i-1} = \Delta x_{i+1} = \Delta x_i = h \) (i.e. uniform mesh)
the finite volume scheme is reduced to a finite difference scheme

\[
\phi_i = \frac{\phi_{i-1} + \phi_{i+1} + \bar{S}_{\phi_i} h^2}{2}
\]

(317)
1D FVM pseudocode

**Data:** Read volumes data and BCs
Create nodes and ghost nodes;

\[ n := 1; \]

repeat

\[ R := 0; \]

for \( i := 2 \) to \( i_{\text{max}} - 1 \) do

\[
\phi^{n+1}_i := \frac{\Delta x_{i+1} \phi^n_{i-1} + \Delta x_{i-1} \phi^n_{i+1} + \bar{S}_i \Delta x_{i-1} \Delta x_{i+1} \Delta x_i}{\Delta x_{i-1} + \Delta x_{i+1}}; \\
R := \max \left( |\phi^{n+1}_i - \phi^n_i|, R \right); \\
\]

Update ghost nodes;

\[ n := n + 1; \]

until \( n \leq n_{\text{max}} \) and \( R > R_{\text{min}} \);
Results for nonuniform and uniform grids
Let us consider the same ordinary differential equation

\[ y''(x) + 20x = 0 \]  \hspace{1cm} (318)

subjected to both the Dirichlet \( y(0) = 0 \) and Neumann \( y'(1) = 0 \) boundary conditions. The specific solution is now

\[ y(x) = -10x \left( \frac{x^2}{3} - 1 \right) \]  \hspace{1cm} (319)
Results for nonuniform and uniform grids
Finite Element Method
Method of weighted residuals

The mathematical foundation of the finite element method is in the method of weighted residuals. Imagine ordinary differential equation

\[ y''(x) + 20x = 0 \]  \hspace{1cm} (320)

subjected to boundary conditions \( y(0) = y(1) = 0 \). The exact general solution of this equation is

\[ y(x) = -\frac{10}{3} x^3 + C_1 x + C_2 \]  \hspace{1cm} (321)

and a specific solution subjected to boundary conditions

\[ y(x) = -\frac{10}{3} (x^3 - x) \]  \hspace{1cm} (322)
Method of weighted residuals

The method seeks an approximate solution $\hat{y}$ in the general form

$$\hat{y}(x) = \sum_{i=1}^{N} C_i N_i(x)$$  \hspace{1cm} (323)

where $N_i$ are known trial functions which should be continuous and fulfilled boundary conditions. The constants $C_i$ are unknown and they will be determined. A residual $R$ appears when substituting approximate solution $\hat{y}$ into the differential equations

$$R(x) = \hat{y}''(x) + 20x \neq 0$$  \hspace{1cm} (324)

The unknown $C_i$ constant are determined for $i = 1, \ldots N$ from

$$\int_{0}^{1} W_i(x) R(x) \, dx = 0$$  \hspace{1cm} (325)
Method of weighted residuals

\[ \int_0^1 W_i(x) R(x) \, dx = 0 \quad i = 1, \ldots, N \quad (326) \]

Choices for the weighting functions \( W_i \)

- Collocation method \( W_i(x) = \delta(x - x_i) \)
- Subdomain method
  \[ W_i(x) = H(x - x_{i-1}) - H(x - x_i) \]
- Galerkin’s method \( W_i(x) = N_i(x) \)

\[ \int_0^1 N_i(x) R(x) \, dx = 0 \quad i = 1, \ldots, N \quad (327) \]

- Least Squares Method \( W_i(x) = \frac{\partial R}{\partial C_i} \)

\[ \int_0^1 \frac{\partial R}{\partial C_i} R(x) \, dx = 0 \quad i = 1, \ldots, N \quad (328) \]
Method of weighted residuals - example

A polynomial trial functions can be assumed

\[ N(x) = x^r(x - 1)^s \] (329)

It is continuous and fulfils boundary conditions. Just one trial function for \( r = s = 1 \) is the simplest case

\[ N_1(x) = x(x - 1) \] (330)

The approximate solution \( \hat{y}(x) = \sum_{i=1}^{N} C_i N_i(x) \) where \( N = 1 \) takes the following form

\[ \hat{y}(x) = C_1 N_1(x) = C_1(x^2 - x) \] (331)

Residual may now be expressed as

\[ R(x) = 2C_1 + 20x \neq 0 \] (332)
The unknown constant $C_1$ may be determined upon integrating (Galerkin’s method of weighted residuals)

\[
\int_0^1 x(x - 1)(2C_1 + 20x) \, dx = 0 \quad (333)
\]

This gives $-\frac{1}{3}(5 + C_1) = 0$ and allows to determine $C_1 = -5$. The approximate solution is now

\[
\hat{y}(x) = -5x(x - 1) \quad (334)
\]

and can be compared with the exact solution

\[
y(x) = -\frac{10}{3}(x^3 - x) \quad (335)
\]
The simplest case with just one trial function approximates the exact solution more or less acceptably. Better agreement is possible with more than one trial functions.
The two polynomial trial functions can be assumed

\[ N_1(x) = x(x - 1), \quad N_2(x) = x^2(x - 1) \quad (336) \]

Both are continuous and fulfil boundary conditions.

The approximate solution \( \hat{y}(x) = \sum_{i=1}^{N} C_i N_i(x) \)

where \( N = 2 \) takes the following form

\[ \hat{y}(x) = C_1 N_1(x) + C_2 N_2(x) = C_1(x^2 - x) + C_2(x^3 - x^2) \quad (337) \]

Residual may now be expressed as

\[ R(x) = 2C_1 + 2C_2(3x - 1) + 20x \neq 0 \quad (338) \]
The unknown constants $C_1, C_2$ may be determined upon integrating

\[
\int_0^1 x(x - 1)(2C_1 + 2C_2(3x - 1) + 20x) \, dx = 0 \\
\int_0^1 x^2(x - 1)(2C_1 + 2C_2(3x - 1) + 20x) \, dx = 0
\]

This gives $10 + 2C_1 + C_2 = 0$ and

$1 + \frac{1}{6}C_1 + \frac{2}{15}C_2 = 0$ and allows to determine $C_1 = C_2 = -\frac{10}{3}$. The approximate solution is now

\[
\hat{y}(x) = -\frac{10}{3} x(x - 1)(x + 1) \tag{339}
\]
The case with two trial function approximates the exact solution very well. It is far better that the previous case. There is no visible difference. In fact, it is even the exact solution

\[-\frac{10}{3} x(x - 1)(x + 1) = -\frac{10}{3} (x^3 - x) \] (340)
The method of weighted residuals constitutes foundation of the final element method. The method exploits an integral formulation to minimise residual errors. Trail functions of this method are global. It is usually difficult task to find a proper one that satisfies boundary conditions. The more dimensions the worse.
‘Element’ formulation

The approximate solution is expressed as

\[ \hat{y}_e = y_j N_1(x) + y_{j+1} N_2(x) = \mathbf{N} \cdot \mathbf{y}_e \]  

(341)

where the known local trial functions \( \mathbf{N} \) and the unknown nodal values \( \mathbf{y}_e \) are collected as vectors

\[ \mathbf{N} = (N_1, N_2) \]  

(342a)

\[ \mathbf{y}_e = (y_j, y_{j+1}) \]  

(342b)

The local trial functions are simply a linear interpolation

\[ N_1 = \frac{x_{j+1} - x}{x_{j+1} - x_j} \quad x_j \leq x \leq x_{j+1} \]  

(343a)

\[ N_2 = \frac{x - x_j}{x_{j+1} - x_j} \quad x_j \leq x \leq x_{j+1} \]  

(343b)
‘Element’ formulation

For each element we have the Galerkin residual condition

\[ \int_{x_j}^{x_{j+1}} N R \, dx = 0 \quad j = 1, \ldots, N \] (344)

Taking under consideration our differential equation
\[ y'' + 20x = 0 \]
and the approximate solution \( \hat{y}_e \) it is now possible to express the residual as

\[ \int_{x_j}^{x_{j+1}} N \left( \frac{d^2 \hat{y}_e}{dx^2} + 20x \right) \, dx = 0 \] (345)

The second derivative has to be replaced. This is due to linear nature of the trial functions.
Integration by parts makes it possible to replace the second derivative

\[
N \frac{d\hat{y}_e}{dx} \bigg|_{x_j}^{x_{j+1}} - \int_{x_j}^{x_{j+1}} \frac{dN}{dx} \frac{d\hat{y}_e}{dx} \, dx + \int_{x_j}^{x_{j+1}} N20x \, dx = 0
\]  

(346)

Finally, matrix form of the Galerkin residual condition for each element is now

\[
\int_{x_j}^{x_{j+1}} \frac{dN}{dx} \frac{dN}{dx} \, dx \cdot \vec{y}_e = \int_{x_j}^{x_{j+1}} N20x \, dx + N \frac{d\hat{y}_e}{dx} \bigg|_{x_j}^{x_{j+1}}
\]

\[
j = 1, \ldots, N \quad (347)
\]
The so called ‘stiffness’ matrix for each element $e$ may be introduced

$$K_e = \int_{x_j}^{x_{j+1}} \frac{dN}{dx} \frac{dN}{dx} \, dx$$  \hspace{1cm} (348)

The above matrix is symmetric. The so called ‘displacement’ vector is also introduced

$$F_e = \int_{x_j}^{x_{j+1}} N20x \, dx + N \frac{d\hat{y}_e}{dx} \Big|_{x_j}^{x_{j+1}}$$  \hspace{1cm} (349)

The Galerkin residual condition for each element may now be written as

$$K_e \cdot y_e = F_e$$  \hspace{1cm} (350)
Element equations

It is possible to simplify the matrices even further. For the linear trial functions one gets the ‘stiffness’ matrix

\[
K_e = \int_{x_j}^{x_{j+1}} \left( \frac{dN_1}{dx} \frac{dN_2}{dx} \frac{dN_1}{dx} \frac{dN_2}{dx} \right) dx = \frac{1}{x_{j+1} - x_j} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}
\]

(351)

and the ‘displacement vector’

\[
F_e = \int_{x_j}^{x_{j+1}} \left( N_1 20x \right) dx + \left( N_1 \frac{dy_e}{dx} \right|_{x_{j+1}} \left( x_{j+1} \right) - \left( N_2 \frac{dy_e}{dx} \right|_{x_{j}} \left( x_j \right) \right)
\]

(352)

If the gradients are dropped, as discussed further, we have

\[
F_e = -\frac{10}{3} (x_j - x_{j+1}) \left( 2x_j + x_{j+1} \right) \left( x_j + 2x_{j+1} \right)
\]

(353)
Element equations - example

For the interval $[0; 1]$ divided equally into 3 elements we have the element matrices

$$K_1 = K_2 = K_3 = \begin{pmatrix} 3 & -3 \\ -3 & 3 \end{pmatrix} \quad (354)$$

The global assembly process (coupling):

$$K = \begin{pmatrix} K_{11}^{11} & K_{12}^{11} & 0 & 0 \\ K_{12}^{11} & K_{12}^{22} + K_{21}^{11} & K_{12}^{12} & 0 \\ 0 & K_{22}^{12} & K_{22}^{22} + K_{31}^{11} & K_{22}^{12} \\ 0 & 0 & K_{32}^{12} & K_{32}^{22} \end{pmatrix} \quad (355)$$

results in

$$K = \begin{pmatrix} 3 & -3 & 0 & 0 & -3 & 6 & -3 & 0 \\ 0 & -3 & 6 & -3 & 0 & -3 & 6 & -3 \\ 0 & 0 & -3 & 3 \end{pmatrix} \quad (356)$$
The ‘displacement’ vector for equally divided interval [0; 1] take form

\[
F_1 = \left( \frac{10}{27} - \frac{dy(0)}{dx} \right), \quad F_2 = \left( \frac{40}{27} - \frac{dy(\frac{1}{3})}{dx} \right), \quad F_3 = \left( \frac{70}{27} - \frac{dy(\frac{2}{3})}{dx} \right)
\]

After the global assembly process one finally gets

\[
F = \begin{pmatrix} F^1_1 \\ F^2_1 + F^1_2 \\ F^2_2 + F^1_3 \\ F^2_3 \end{pmatrix} = \begin{pmatrix} \frac{10}{27} - \frac{dy(0)}{dx} \\ \frac{60}{27} \\ \frac{120}{27} + \frac{dy(1)}{dx} \\ \frac{80}{27} \end{pmatrix}
\]
The global (assembled) system of linear equations is

\[
\begin{pmatrix}
3 & -3 & 0 & 0 \\
-3 & 6 & -3 & 0 \\
0 & -3 & 6 & -3 \\
0 & 0 & -3 & 3 \\
\end{pmatrix}
\cdot
\begin{pmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
\end{pmatrix}
=
\begin{pmatrix}
\frac{10}{27} - \frac{dy(0)}{dx} \\
\frac{60}{27} \\
\frac{120}{27} \\
\frac{80}{27} + \frac{dy(1)}{dx} \\
\end{pmatrix}
\]  

(359)

It cannot, however, be solved yet. This is due to necessity of applying the global boundary conditions. These are \( y_1 = y_4 = 0 \). Two typical methods of applying them are discussed further.
Boundary conditions

Extracting only these equations that are related to unknown functions $y_2$ and $y_3$ for $y_1 = y_4 = 0$ results in

\[
\begin{pmatrix}
\cdot & \cdot & \cdot & \cdot \\
\cdot & 6 & -3 & \cdot \\
\cdot & -3 & 6 & \cdot \\
\cdot & \cdot & \cdot & \cdot
\end{pmatrix}
\begin{pmatrix}
y_2 \\
y_3
\end{pmatrix}
= 
\begin{pmatrix}
60 \\
\frac{27}{120} \\
\frac{27}{27}
\end{pmatrix}
\] (360)

or simpler in

\[
\begin{pmatrix}
6 & -3 \\
-3 & 6
\end{pmatrix}
\begin{pmatrix}
y_2 \\
y_3
\end{pmatrix}
= 
\begin{pmatrix}
60 \\
\frac{27}{120}
\end{pmatrix}
\] (361)

The above system may now be solved to obtain the unknown values $y_2, y_3$. 
The second method does not change the layout of the matrices. However, it involves modification of specific elements by multiplying them by a ‘large’ number. These elements are located on the diagonal of the ‘stiffness’ matrix and corresponding positions of the ‘displacement’ vector (if non-zero)

\[
\begin{pmatrix}
3 \cdot 10^7 & -3 & 0 & 0 \\
-3 & 6 & -3 & 0 \\
0 & -3 & 6 & -3 \\
0 & 0 & -3 & 3 \cdot 10^7 \\
\end{pmatrix}
\begin{pmatrix}
y_1 \\
y_2 \\
y_3 \\
y_3 \\
\end{pmatrix} =
\begin{pmatrix}
0 \\
60 \\
27 \\
120 \\
27 \\
0 \\
\end{pmatrix}
\]

(362)
ODE FEM pseudocode

**Data:** Read $N$ elements, nodes and BCs
Create global matrix $K$ and vectors $F$, $y$;

for $e := 1$ to $N$ do

$K_e := \int_e \frac{dN}{dx} \frac{dN}{dx} \, dx$;
$F_e := \int_e N20x \, dx$;
Add $K_e$ to $K$;
Add $F_e$ to $F$;

Apply BCs;
Solve linear system $K \cdot y = F$;
3 elements, 4 nodes
Results - ODE

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References

6 elements, 7 nodes
9 elements, 10 nodes
Results - ODE

15 elements, 16 nodes

$y$ vs $x$
Linear and quadratic interpolation

Considering line equation $\hat{y}_e = a + bx$ and utilising it for two different points $(x_j, y_j)$ and $(x_{j+1}, y_{j+1})$ we can get the following system of equations

$$
\begin{pmatrix}
  y_j \\
  y_{j+1}
\end{pmatrix} =
\begin{pmatrix}
  1 & x_j \\
  1 & x_{j+1}
\end{pmatrix} \cdot
\begin{pmatrix}
  a \\
  b
\end{pmatrix}
$$

(363)

It can be easily solved for $a$ and $b$

$$
\begin{pmatrix}
  a \\
  b
\end{pmatrix} =
\begin{pmatrix}
  1 & x_j \\
  1 & x_{j+1}
\end{pmatrix}^{-1} \cdot
\begin{pmatrix}
  y_j \\
  y_{j+1}
\end{pmatrix}
$$

(364)

Keeping in mind that $\hat{y}_e = N \cdot y_e$ where $N = (N_1, N_2)$ and $y_e = (y_j, y_{j+1})$ we can utilise the solution for $a$ and $b$ to get

$$
\hat{y}_e = a + bx = \frac{x_{j+1} - x}{x_{j+1} - x_j} y_j + \frac{x - x_j}{x_{j+1} - x_j} y_{j+1} = N_1 y_j + N_2 y_{j+1}
$$

(365)
Linear and quadratic interpolation

Introducing $L_1$ and $L_2$ for a one-dimensional element

\[
L_1 = N_1 = \frac{x_{j+1} - x}{x_{j+1} - x_j} \quad (366a)
\]
\[
L_2 = N_2 = \frac{x - x_j}{x_{j+1} - x_j} \quad (366b)
\]

one can formulate similar system of equation for quadratic interpolation $\hat{y}_e = a + bx + cx^2$ through the points $(x_j, y_j)$, $(x_j + \frac{1}{2}, y_j + \frac{1}{2})$ and $(x_{j+1}, y_{j+1})$

\[
\begin{pmatrix}
  y_j \\
  y_{j+\frac{1}{2}} \\
  y_{j+1}
\end{pmatrix} =
\begin{pmatrix}
  1 & x_j & x_j^2 \\
  1 & x_{j+\frac{1}{2}} & x_{j+\frac{1}{2}}^2 \\
  1 & x_{j+1} & x_{j+1}^2
\end{pmatrix}
\cdot
\begin{pmatrix}
  a \\
  b \\
  c
\end{pmatrix} \quad (367)
\]
The quadratic trial functions can be also expressed in terms of linear trial functions

\[ N_1 = L_1 (2L_1 - 1), \]
\[ N_2 = 4L_1 L_2, \]
\[ N_3 = L_2 (2L_2 - 1) \]
The ‘stiffness’ matrix for each element \( e \) may now be calculated as

\[
K_e = \int_{x_j}^{x_{j+1}} \frac{dN}{dx} \frac{dN}{dx} \, dx = \frac{1}{3(x_{j+1} - x_j)} \begin{pmatrix} 7 & -8 & 1 \\ -8 & 16 & -8 \\ 1 & -8 & 7 \end{pmatrix}
\]

(369)

The ‘displacement’ vector is now

\[
F_e = \int_{x_j}^{x_{j+1}} N20x \, dx = \frac{10}{3}(x_{j+1} - x_j) \begin{pmatrix} x_j \\ 2(x_j + x_{j+1}) \\ x_{j+1} \end{pmatrix}
\]

(370)

The Galerkin residual condition for each element is the same as previously

\[
K_e \cdot y_e = F_e.
\]

(371)
ODE quadratic FEM pseudocode

**Data:** Read $N$ linear elements, $n$ nodes and BCs

Insert midpoints $x_{j+\frac{1}{2}} := \frac{x_j + x_{j+1}}{2}$;

$n := 2n - 1$;

Create global matrix $K$ and vectors $F, y$;

**for** $e := 1$ to $N$ **do**

$K_e := \int_e \frac{dN}{dx} \frac{dN}{dx} \, dx$;

$F_e := \int_e N20x \, dx$;

Add $K_e$ to $K$;

Add $F_e$ to $F$;

Apply BCs;

Solve linear system $K \cdot y = F$;
ODE - linear vs quadratic interpolation

6 elements, 7 nodes. 3 elements, 7 nodes.
The generalised $p$-norm is given by

$$\| f \|_p = \left( \int_L |f(x)|^p \, dx \right)^{\frac{1}{p}} \quad (372)$$

where for $p = 2$ we have a special case

$$\| f \|_2 = \sqrt{\int_L f^2(x) \, dx} \quad (373)$$

The error $E = y - \hat{y}$ of a finite element solution $\hat{y}$ may now be defined by means of 2-norm. It may take the following form

$$\| E' \|_2^2 \leq C \sum_{e=1}^{N} r_e^2 \quad (374)$$
The element residue $r_e$ is defined as

$$r_e = |L_e| \| f + \hat{y}_e'' \|_2$$

but due to the linear form of trial functions $N_i'' = 0$ it is true that $\hat{y}_e'' = 0$. This means that the element residue is $r_e = |L_e| \| f \|_2$ and solution error

$$\| E' \|_2^2 \leq C \sum_{e=1}^{N} |L_e|^2 \| f \|_2^2$$

Element’s length is $|L_e| = x_{j+1} - x_j$ and utilising the trapezoidal rule we can express the element residue as

$$r_e = |L_e| \sqrt{\int_{L_e} f(x)^2 \, dx} \approx (x_{j+1} - x_j)^{3/2} \sqrt{\frac{f_j^2 + f_{j+1}^2}{2}}$$

The above approximation is used for mesh refinement.
Mesh refinement - results

![Mesh refinement diagram](image)

### Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References
Mesh refinement - results

![Graph showing mesh refinement results with x and y axes labeled.](image)

---

Mesh refinement - results

Contents

- Description of fluid/solid at different scales
  - Finite Difference Method
  - Finite Volume Method
  - Finite Element Method
  - Monte Carlo Method
  - Smoothed Particle Hydrodynamics
  - Lattice Boltzmann Method
  - Turbulence modelling

References
Mesh refinement - results

![Graph of mesh refinement results]

K. Tesch; Numerical Methods
Mesh refinement - results

Contents

Description of fluid/solid at different scales

Finite Difference Method

Finite Volume Method

Finite Element Method

Monte Carlo Method

Smoothed Particle Hydrodynamics

Lattice Boltzmann Method

Turbulence modelling

References

K. Tesch; Numerical Methods
Mesh refinement - results

Contents
Description of fluid/solid at different scales
Finite Difference Method
Finite Volume Method
Finite Element Method
Monte Carlo Method
Smoothed Particle Hydrodynamics
Lattice Boltzmann Method
Turbulence modelling
References

K. Tesch; Numerical Methods
FEM for Poisson equation

Let the two dimensional form of the Poisson equation on $\Omega$

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = -a$$  \hspace{1cm} (378)

be subjected to the Dirichlet boundary condition $U(x, y) = 0$ for every $(x, y) \in \partial\Omega$. It is true that

$$\iint_{\Omega} f(x, y) \, dx \, dy = \sum_e \iint_{\Omega_e} f(x, y) \, dx \, dy$$  \hspace{1cm} (379)
The approximate solution is expressed as

\[ \hat{U}_e = N \cdot U_e \]  

(380)

where the known local trial functions \( N \) and the unknown nodal values \( U_e \) are

\[ N = (N_1, N_2, N_3) \]  

(381a)

\[ U_e = (U_1, U_2, U_3) \]  

(381b)

For each element we have the Galerkin residual condition

\[ \int \int_{\Omega_e} N R \, dx = 0 \]  

(382)
Taking under consideration Poisson equation and the approximate solution \( \hat{U}_e \) it is now possible to express the residual as

\[
\iint_{\Omega_e} \mathbf{N} \left( \frac{\partial^2 \hat{U}_e}{\partial x^2} + \frac{\partial^2 \hat{U}_e}{\partial y^2} + a \right) \, dx \, dy = 0 \quad (383)
\]

The second derivative has to be replaced (due to linear nature of the trial functions). This can be done by means of Green’s first identity

\[
\iint_{S} \left( \psi \frac{\partial^2 \varphi}{\partial x^2} + \psi \frac{\partial^2 \varphi}{\partial y^2} \right) \, dx \, dy = \int_{\partial S} \psi \frac{\partial \varphi}{\partial n} \, dL
\]

\[
- \iint_{S} \left( \frac{\partial \psi}{\partial x} \frac{\partial \varphi}{\partial x} + \frac{\partial \psi}{\partial y} \frac{\partial \varphi}{\partial y} \right) \, dx \, dy \quad (384)
\]
‘Element’ formulation

Integration by means of Green’s identity makes it possible to replace the second derivative

\[ \iint_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial \hat{U}_e}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial \hat{U}_e}{\partial y} \right) \, dx \, dy \]

\[ - \int_{\partial \Omega_e} N \frac{\partial \hat{U}_e}{\partial n} \, dL - \iint_{\Omega_e} N a \, dx \, dy = 0 \quad (385) \]

The matrix form of the Galerkin residual condition for each element can now be expressed

\[ \iint_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial N}{\partial y} \right) \, dx \, dy \cdot \mathbf{U}_e = \iint_{\Omega_e} N a \, dx \, dy \]

\[ (386) \]
Introducing the ‘stiffness’ matrix for each element $e$

$$K_e = \int \int_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial N}{\partial y} \right) \, dx \, dy$$  \hspace{1cm} (387)

and the ‘displacement’ vector

$$F_e = \int \int_{\Omega_e} N a \, dx \, dy$$  \hspace{1cm} (388)

one may obtain the Galerkin residual condition for each element in the form

$$K_e \cdot U_e = F_e$$  \hspace{1cm} (389)
The expanded version of the ‘stiffness’ matrix is

\[
\mathbf{K}_e = \iiint_{\Omega_e} \left( \begin{array}{ccc} \frac{\partial N_1}{\partial x} \frac{\partial N_1}{\partial x} + \frac{\partial N_1}{\partial y} \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} \frac{\partial N_2}{\partial x} + \frac{\partial N_1}{\partial y} \frac{\partial N_2}{\partial y} & \frac{\partial N_1}{\partial x} \frac{\partial N_3}{\partial x} + \frac{\partial N_1}{\partial y} \frac{\partial N_3}{\partial y} \\ \frac{\partial N_1}{\partial x} \frac{\partial N_2}{\partial x} + \frac{\partial N_2}{\partial y} \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial x} \frac{\partial N_2}{\partial x} + \frac{\partial N_2}{\partial y} \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} \frac{\partial N_3}{\partial x} + \frac{\partial N_2}{\partial y} \frac{\partial N_3}{\partial y} \\ \frac{\partial N_1}{\partial x} \frac{\partial N_3}{\partial x} + \frac{\partial N_3}{\partial y} \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial x} \frac{\partial N_3}{\partial x} + \frac{\partial N_3}{\partial y} \frac{\partial N_2}{\partial y} & \frac{\partial N_3}{\partial x} \frac{\partial N_3}{\partial x} + \frac{\partial N_3}{\partial y} \frac{\partial N_3}{\partial y} \end{array} \right) \, dx \, dy
\]

(390)

Similarly, the same for the ‘displacement’ vector

\[
\mathbf{F}_e = a \iiint_{\Omega_e} \begin{pmatrix} N_1 \\ N_2 \\ N_3 \end{pmatrix} \, dx \, dy
\]

(391)

The actual form of matrices depends on the trial functions. Linear form of these are discussed further.
Considering plane equation

\[ \hat{U}_e = a + bx + cy \]  \hspace{1cm} (392)

one can formulate the following system of equations

\[ U_1 = a + bx_i + cy_i \]  \hspace{1cm} (393a)
\[ U_2 = a + bx_j + cy_j \]  \hspace{1cm} (393b)
\[ U_3 = a + bx_k + cy_k \]  \hspace{1cm} (393c)

for three different points \((x_i, y_i), (x_j, y_j), (x_k, y_k)\).

Solving these for \(a, b\) and \(c\) results in

\[ \hat{U}_e = \frac{U_1}{2S_e}(a_i+b_ix+c_iy) + \frac{U_2}{2S_e}(a_j+b_jx+c_jy) + \frac{U_3}{2S_e}(a_k+b_kx+c_ky) \]

\hspace{1cm} (394)
The linear trial functions then are

\[ N_1 = \frac{1}{2S_e} \left( a_i + b_i x + c_i y \right) \]  \hspace{1cm} (395a)

\[ N_2 = \frac{1}{2S_e} \left( a_j + b_j x + c_j y \right) \]  \hspace{1cm} (395b)

\[ N_3 = \frac{1}{2S_e} \left( a_k + b_k x + c_k y \right) \]  \hspace{1cm} (395c)

where

\[ a_i = x_j y_k - x_k y_j; \quad a_j = x_k y_i - x_i y_k; \quad a_k = x_i y_j - x_j y_i; \]

\[ b_i = y_j - y_k; \quad b_j = y_k - y_i; \quad b_k = y_i - y_j; \]

\[ c_i = x_k - x_j; \quad c_j = x_i - x_k; \quad c_k = x_j - x_i; \]

\[ S_e = \frac{1}{2} |c_k b_j - c_j b_k| \]
Linear interpolation
Now it is possible to calculate necessary derivatives appearing in the ‘stiffness’ matrix

$$\frac{\partial N_1}{\partial x} = \frac{b_i}{2S_e}, \quad \frac{\partial N_2}{\partial x} = \frac{b_j}{2S_e}, \quad \frac{\partial N_3}{\partial x} = \frac{b_k}{2S_e}, \quad (397a)$$

$$\frac{\partial N_1}{\partial y} = \frac{c_i}{2S_e}, \quad \frac{\partial N_2}{\partial y} = \frac{c_j}{2S_e}, \quad \frac{\partial N_3}{\partial y} = \frac{c_k}{2S_e} \quad (397b)$$

The same concerns integrals appearing in the ‘displacement’ vector

$$\iint_{S_e} N_1^\alpha N_2^\beta N_3^\gamma \, dx \, dy = 2S_e \frac{\alpha! \beta! \gamma!}{(\alpha + \beta + \gamma + 2)!} \quad (397c)$$
Now it is possible to simplify the matrices even further. For the linear trial functions one gets the 'stiffness' matrix

$$K_e = \frac{1}{4S_e} \begin{pmatrix}
    b_i^2 + c_i^2 & b_i b_j + c_i c_j & b_i b_k + c_i c_k \\
    b_i b_j + c_i c_j & b_j^2 + c_j^2 & b_j b_k + c_j c_k \\
    b_i b_k + c_i c_k & b_j b_k + c_j c_k & b_k^2 + c_k^2
\end{pmatrix} \quad (398)$$

and the 'displacement vector'

$$F_e = \frac{S_e}{3} \begin{pmatrix}
    a_i \\
    a_j \\
    a_k
\end{pmatrix} \quad (399)$$
Four element example

\[ \Omega \]

\[ \partial \Omega \]

\[ U = 0 \]

\[ U = 0 \]

\[ U = 0 \]

\[ U = 0 \]

\[ U = 0 \]

\[ U = 0 \]

\[ \Omega \]

\[ \partial \Omega \]

\[ 1(0, 0) \]

\[ 2(10, 0) \]

\[ 3(10, 10) \]

\[ 4(0, 10) \]

\[ 3(10, 10) \]

\[ 4(0, 10) \]

\[ 5(5, 5) \]

\[ 2(10, 0) \]

\[ 1(0, 0) \]

\[ 4(0, 10) \]

\[ 3(10, 10) \]
The global assembly process (coupling) for the considered four element case:

\[
K = \begin{pmatrix}
K_{11}^{11} + K_{41}^{11} & K_{12}^{12} & 0 & K_{42}^{12} & K_{13}^{13} + K_{43}^{13} \\
K_{12}^{12} & K_{22}^{12} + K_{11}^{11} & 0 & K_{42}^{12} & K_{23}^{13} + K_{12}^{13} \\
0 & K_{22}^{12} & K_{22}^{12} + K_{33}^{11} & K_{33}^{12} & K_{23}^{13} + K_{33}^{13} \\
K_{41}^{13} + K_{44}^{13} & K_{23}^{13} + K_{43}^{13} & K_{23}^{13} + K_{33}^{13} & K_{33}^{13} + K_{43}^{13} + K_{44}^{33} & K_{33}^{13} + K_{43}^{13} + K_{44}^{33}
\end{pmatrix}
\]

The element matrices are identical

\[
K_1 = K_2 = K_3 = K_4 = \frac{1}{2} \begin{pmatrix}
1 & 0 & -1 \\
0 & 1 & -1 \\
-1 & -1 & 1
\end{pmatrix}
\]

Finally, the global ‘stiffness’ matrix is

\[
K = \begin{pmatrix}
1 & 0 & 0 & 0 & -1 \\
0 & 1 & 0 & 0 & -1 \\
0 & 0 & 1 & 0 & -1 \\
0 & 0 & 0 & 1 & -1 \\
-1 & -1 & -1 & -1 & 4
\end{pmatrix}
\]
The ‘displacement’ element vector for the considered four elements

\[
\mathbf{F}_1 = \mathbf{F}_2 = \mathbf{F}_3 = \mathbf{F}_4 = \frac{25}{3} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad (403)
\]

After the global assembly process one finally gets

\[
\mathbf{F} = \begin{pmatrix} 
\mathbf{F}_1^1 + \mathbf{F}_4^1 \\
\mathbf{F}_1^2 + \mathbf{F}_2^1 \\
\mathbf{F}_2^2 + \mathbf{F}_3^1 \\
\mathbf{F}_3^2 + \mathbf{F}_4^2 \\
\mathbf{F}_1^3 + \mathbf{F}_2^3 + \mathbf{F}_3^3 + \mathbf{F}_4^3 
\end{pmatrix} = \frac{50}{3} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 2 \end{pmatrix} \quad (404)
\]
Poisson FEM pseudocode

**Data:** Read $N$ elements, nodes and BCs
Create global matrix $K$ and vectors $F$, $y$;

for $e := 1$ to $N$ do

\[
K_e := \iint_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial N}{\partial y} \right) \, dx \, dy;
\]

\[
F_e := \iint_{\Omega_e} Na \, dx \, dy;
\]

Add $K_e$ to $K$;
Add $F_e$ to $F$;

Apply BCs;
Solve linear system $K \cdot y = F$;
Contents

- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling

References
8 elements, 9 nodes
Results - PDE

Contents
- Description of fluid/solid at different scales
- Finite Difference Method
- Finite Volume Method
- Finite Element Method
- Monte Carlo Method
- Smoothed Particle Hydrodynamics
- Lattice Boltzmann Method
- Turbulence modelling
- References

50 elements, 36 nodes
200 elements, 121 nodes
Results - PDE

800 elements, 441 nodes
FEM for Laplace equation

Let the two dimensional form of Laplace equation

\[
\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = 0
\]  

(405)

or \( \nabla^2 \varphi = 0 \) on \( \Omega \) be subjected to both boundary conditions on \( \partial \Omega \):

- **Neumann**
  \[
  \frac{\partial \varphi}{\partial n} = \hat{n} \cdot \nabla \varphi = \hat{n} \cdot (U_x, U_y) = n_x U_x + n_y U_y = -f_N,
  \]

- **Dirichlet** (as previously)
  \[
  \varphi = \text{const} = f_D.
  \]  

(406)
As previously, FEM formulation for Poisson equation subjected to Dirichlet and Neumann BCs is

\[
\iint_{\Omega_e} \left( \frac{\partial \mathbf{N}}{\partial x} \frac{\partial \hat{\mathbf{U}}_e}{\partial x} + \frac{\partial \mathbf{N}}{\partial y} \frac{\partial \hat{\mathbf{U}}_e}{\partial y} \right) \, dx \, dy \\
- \int_{\partial \Omega_e} \mathbf{N} \frac{\partial \hat{\mathbf{U}}_e}{\partial n} \, dL - \iint_{\Omega_e} \mathbf{N} a \, dx \, dy = 0 \quad (407)
\]

- Poisson \( \nabla^2 \varphi = -a \) with Dirichlet BC

\[
\iint_{\Omega_e} \left( \frac{\partial \mathbf{N}}{\partial x} \frac{\partial \mathbf{N}}{\partial x} + \frac{\partial \mathbf{N}}{\partial y} \frac{\partial \mathbf{N}}{\partial y} \right) \, dx \, dy \cdot \mathbf{U}_e = \iint_{\Omega_e} \mathbf{N} a \, dx \, dy
\]

- Laplace \( \nabla^2 \varphi = 0 \) with Dirichlet + Neumann BC

\[
\iint_{\Omega_e} \left( \frac{\partial \mathbf{N}}{\partial x} \frac{\partial \mathbf{N}}{\partial x} + \frac{\partial \mathbf{N}}{\partial y} \frac{\partial \mathbf{N}}{\partial y} \right) \, dx \, dy \cdot \varphi_e = - \int_{\partial \Omega_e} \mathbf{N} f_N \, dL \quad (408)
\]
Introducing the ‘stiffness’ matrix as previously for each element \( e \)

\[
K_e = \iint_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial N}{\partial y} \right) \, dx \, dy \tag{410}
\]

and the ‘displacement’ vector

\[
F_e = - \int_{\partial \Omega_e} N f_N \, dL \tag{411}
\]

one may obtain the Galerkin residual condition for each element in the form

\[
K_e \cdot \varphi_e = F_e. \tag{412}
\]
The expanded version of the ‘stiffness’ matrix look the same as previously but the ‘displacement’ vector is now

\[ \mathbf{F}_e = f_N \int_{\partial \Omega_e} \mathbf{N} \, d\mathbf{L} = -\frac{1}{2} |L| \mathbf{f}_N \mathbf{1} \]  

(413)

The vector \( \mathbf{1} \) may take of the three following forms

\[
\begin{pmatrix}
1 \\
1 \\
0
\end{pmatrix},
\begin{pmatrix}
1 \\
0 \\
1
\end{pmatrix},
\begin{pmatrix}
0 \\
1 \\
1
\end{pmatrix}
\]  

(414)

\( |L| \) stands for element side length.
Laplace FEM pseudocode

**Data:** Read $N$ elements, nodes and BCs

Create global matrix $K$ and vectors $F$, $y$;

for $e := 1$ to $N$ do

$$K_e := \iint_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial N}{\partial y} \right) \, dx \, dy;$$

$$F_e := -\int_{\partial\Omega_e} N f_N \, dL;$$

Add $K_e$ to $K$;

Add $F_e$ to $F$;

Apply BCs;

Solve linear system $K \cdot y = F;$
Geometry and mesh - Laplace equation

\[ \frac{\partial \varphi}{\partial y} = 0 \]

\[ \frac{\partial \varphi}{\partial n} = 0 \]

\[ \varphi = \text{const} \]

456 nodes and 818 elements
Geometry and mesh - Laplace equation

\[ \frac{\partial \varphi}{\partial y} = 0 \]

351 nodes and 607 elements
Results - Laplace equation

K. Tesch; Numerical Methods
Creeping flow – Stokes equations

\[ \text{Re} \ll 1 \quad (415) \]

\[ 0 = \rho g - \nabla p + \mu \nabla^2 \mathbf{u} \quad (416a) \]

\[ \nabla \cdot \mathbf{u} = 0 \quad (416b) \]

\[ 0 = \rho g_x - \frac{\partial p}{\partial x} + \mu \left( \frac{\partial^2 u_x}{\partial x^2} + \frac{\partial^2 u_x}{\partial y^2} \right) \quad (417a) \]

\[ 0 = \rho g_y - \frac{\partial p}{\partial y} + \mu \left( \frac{\partial^2 u_y}{\partial x^2} + \frac{\partial^2 u_y}{\partial y^2} \right) \quad (417b) \]

\[ \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} = 0 \quad (417c) \]
The approximate solution is expressed as

$$\hat{u}_e = \mathbf{N} \cdot \mathbf{u}_e$$  \hspace{1cm} (418)

where the quadratic trial functions $\mathbf{N}$ and the unknown nodal values $\mathbf{u}_e$ are

$$\mathbf{N} = (N_1, N_2, N_3, N_4, N_5, N_6) \hspace{1cm} (419a)$$
$$\mathbf{u}_e = (U_1, U_2, U_3, U_4, U_5, U_6) \hspace{1cm} (419b)$$

For each element we have the Galerkin residual condition

$$\int \int_{\Omega_e} \mathbf{N} \cdot \mathbf{R} \, dx = 0 \hspace{1cm} (420)$$
The quadratic trial functions can be expressed in terms of linear trial functions

\[ N_1 = L_1 (2L_1 - 1), \]
\[ N_2 = L_2 (2L_2 - 1), \]
\[ N_3 = L_3 (2L_3 - 1), \]
\[ N_4 = 4L_1 L_2, \]
\[ N_5 = 4L_2 L_3, \]
\[ N_6 = 4L_1 L_3. \]
The Galerkin residual condition

\[
\iint_{\Omega_e} N \left( \rho g_x - \frac{\partial \hat{p}_e}{\partial x} + \mu \left( \frac{\partial^2 \hat{u}_{xe}}{\partial x^2} + \frac{\partial^2 \hat{u}_{xe}}{\partial y^2} \right) \right) \, dx \, dy = 0
\]

(422)

by means of Green’s first identity

\[
\iint_S \left( \psi \frac{\partial^2 \varphi}{\partial x^2} + \psi \frac{\partial^2 \varphi}{\partial y^2} \right) \, dx \, dy = \int_{\partial S} \psi \frac{\partial \varphi}{\partial n} \, dL
\]

\[
- \iint_S \left( \frac{\partial \psi}{\partial x} \frac{\partial \varphi}{\partial x} + \frac{\partial \psi}{\partial y} \frac{\partial \varphi}{\partial y} \right) \, dx \, dy
\]

(423)
Momentum conservation equation

\[
\rho g_x \iint_{\Omega_e} N \, dx \, dy - \iint_{\Omega_e} N \frac{\partial N}{\partial x} \, dx \, dy \cdot p_e \\
- \mu \iint_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial N}{\partial y} \right) \, dx \, dy \cdot u_{xe} = 0
\]

(424)

or

\[
K_{pxe} \cdot p_e + K_{xye} \cdot u_{xe} = g_x F_e
\]

(425)

Similarly

\[
K_{pye} \cdot p_e + K_{xye} \cdot u_{ye} = g_y F_e
\]

(426)
Momentum conservation equation

\[ K_{pxe} = \int \int_{\Omega_e} N \frac{\partial N}{\partial x} \, dx \, dy \]  \hspace{1cm} (427)

\[ K_{pye} = \int \int_{\Omega_e} N \frac{\partial N}{\partial y} \, dx \, dy \]  \hspace{1cm} (428)

\[ K_{xye} = \mu \int \int_{\Omega_e} \left( \frac{\partial N}{\partial x} \frac{\partial N}{\partial x} + \frac{\partial N}{\partial y} \frac{\partial N}{\partial y} \right) \, dx \, dy \]  \hspace{1cm} (429)

\[ F_e = \rho \int \int_{\Omega_e} N \, dx \, dy \]  \hspace{1cm} (430)
The Galerkin residual condition

\[ \int \int_{\Omega_e} N \left( \frac{\partial \hat{u}_{xe}}{\partial x} + \frac{\partial \hat{u}_{ye}}{\partial y} \right) \, dx \, dy = 0 \quad (431) \]

The matrix form of the Galerkin residual condition for each element can now be expressed

\[ \int \int_{\Omega_e} N \frac{\partial N}{\partial x} \, dx \, dy \cdot u_{xe} + \int \int_{\Omega_e} N \frac{\partial N}{\partial y} \, dx \, dy \cdot u_{ye} = 0 \quad (432) \]

or

\[ K_{u_{xe}} \cdot u_{xe} + K_{u_{ye}} \cdot u_{ye} = 0 \quad (433) \]
Mass conservation equation

\[ K_{uxe} = \iint_{\Omega_e} N \frac{\partial N}{\partial x} \, dx \, dy \quad (434) \]

\[ K_{uye} = \iint_{\Omega_e} N \frac{\partial N}{\partial y} \, dx \, dy \quad (435) \]
Element equations

\[ K_{px e} \cdot p_e + K_{xy e} \cdot u_{xe} = g_x F_e, \quad (436a) \]
\[ K_{py e} \cdot p_e + K_{xy e} \cdot u_{ye} = g_y F_e, \quad (436b) \]
\[ K_{ux e} \cdot u_{xe} + K_{uy e} \cdot u_{ye} = 0 \quad (436c) \]

\[
\begin{pmatrix}
K_{xy e}^{6\times6} & 0^{6\times6} & K_{px e}^{6\times3} \\
0^{6\times6} & K_{xy e}^{6\times6} & K_{py e}^{6\times3} \\
K_{ux e}^{3\times6} & K_{uy e}^{3\times6} & 0^{3\times3}
\end{pmatrix}
\cdot
\begin{pmatrix}
u_{xe}^{6\times1} \\
u_{ye}^{6\times1} \\
p_e^{3\times1}
\end{pmatrix}
=
\begin{pmatrix}g_x F_e^{6\times1} \\
g_y F_e^{6\times1} \\
0^{3\times1}
\end{pmatrix} \quad (437)
\]

\[ K_e^{15\times15} \cdot u_e^{15\times1} = f_e^{15\times1} \quad (438) \]
Monte Carlo Method
Smoothed Particle Hydrodynamics
Lattice Boltzmann Method
Turbulence modelling
References


