Chapter 9

Numerical Models for Mixing in Rivers and Estuaries
Chapter 9
Numerical Models for Mixing in Rivers and Estuaries

Contents

9.1 Modeling Mixing in Rivers and Estuaries
9.2 Finite Difference Model
9.3 Finite Element Model
9.4 Finite Volume Model
9.5 Finite Particle Model
Objectives

- Think about the definition of model
- Introduce procedure of modeling
- Study various types of numerical schemes
9.1 Modeling Mixing in Rivers and Estuaries

9.1.1 Introduction

- Definition of Model
  - What is a model?
    - a deliberate misrepresentation of reality
    - simplification (approximation) of real system
  - Reason: convenience
  - Purpose:
    - gain understanding
    - predict an outcome
  - Constraints:
    - degree of accuracy depends on degree of simplification
9.1 Modeling Mixing in Rivers and Estuaries

- River model
  Describe the real system of river dynamics using physical or mathematical approach

- Purpose
  Understanding and prediction of river dynamics
9.1 Modeling Mixing in Rivers and Estuaries

- **Uncertainty**
  Most models are intermediate forms between physical-based models and empirical models.

- **Parameter**
  Model have its own parameters to represent their characteristics.

- **Calibration**
  Comparison of model output with observations to tune the model parameters.
  → The calibrated models can be called empirical models.

- **Validation**
  Comparison of output from the calibrated models with observations to evaluate validity of the calibrated models.
## 9.1 Modeling Mixing in Rivers and Estuaries

### 9.1.2 Selection of Model

<table>
<thead>
<tr>
<th>Code</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1A   | One-dimensional tidally averaged | A numerical solution of 1-D tidally averaged dispersion equation [Eq.8.38]  
- steady state model: coefficients are constant in time.  
- unsteady model: flow parameters and dispersion coefficient vary between tidal cycles. |
| 1T   | One-dimensional tidally varying | A numerical solution of Eq. (8.46)  
- Tidal evaluation, velocity and dispersion coefficient vary during tidal cycle. |
### 9.1 Modeling Mixing in Rivers and Estuaries

<table>
<thead>
<tr>
<th>Model</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>1TB</td>
<td>Branching 1-D tidally averaged</td>
<td>A network of 1T models connected at junctions.</td>
</tr>
<tr>
<td>2VA</td>
<td>Two-dimensional tidally averaged</td>
<td>A numerical solution of 2-D tidally averaged dispersion equation.</td>
</tr>
</tbody>
</table>
| 2HA   | Two-dimensional tidally averaged | - 2VA: horizontally averaged model → reservoir, lake  
- 2HA: vertically averaged model → river, estuary |
| 2VT   | Two-dimensional tidally varying | A numerical solution of 2-D tidally varying dispersion equation |
| 2HT   | Two-dimensional tidally varying | |
| 3A    | Three-dimensional tidally averaged | A numerical solution of 3-D tidally averaged dispersion equation |
## 9.1 Modeling Mixing in Rivers and Estuaries

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>3T</strong></td>
<td>Three-dimensional tidally varying</td>
<td></td>
</tr>
<tr>
<td><strong>P</strong></td>
<td>Physical model</td>
<td>A small-scale physical replica of the prototype geometry with provisions of generating tidal and river flow</td>
</tr>
<tr>
<td><strong>NP</strong></td>
<td>Hybrid numerical physical</td>
<td>A combination of a physical and a numerical model, using one model to generate input information for the other</td>
</tr>
</tbody>
</table>
### 9.1 Modeling Mixing in Rivers and Estuaries

<table>
<thead>
<tr>
<th>Mixing mechanism</th>
<th>Appropriate Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trapping</td>
<td>2HT physical model</td>
<td>Well verified for simulation of trapping mechanism</td>
</tr>
<tr>
<td></td>
<td>1TB</td>
<td>Branches represent traps.</td>
</tr>
<tr>
<td>Density-driven</td>
<td>2VA 2VT</td>
<td>In case transverse gravitational circulation is not important.</td>
</tr>
<tr>
<td>circulation</td>
<td>3A 3T</td>
<td>If density-driven currents are important, the equations determining the flow and the salinity distribution are coupled.</td>
</tr>
</tbody>
</table>
### 9.1 Modeling Mixing in Rivers and Estuaries

<table>
<thead>
<tr>
<th>Effect Type</th>
<th>Solution</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tidal pumping</td>
<td>2HT</td>
<td>Accuracy of 2HT may be difficult to establish.</td>
</tr>
<tr>
<td>Physical model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shear flow dispersion</td>
<td>2HT, 2VT</td>
<td></td>
</tr>
<tr>
<td>Wind effects</td>
<td>2HT, 3T</td>
<td></td>
</tr>
<tr>
<td>Rotational effects</td>
<td>2HT</td>
<td>Easily modeled in 2HT models.</td>
</tr>
<tr>
<td>Physical model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Catastrophic/seasonal changes</td>
<td>1A</td>
<td>Long term simulation for a period of a year or more</td>
</tr>
<tr>
<td>Physical model</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
9.1 Modeling Mixing in Rivers and Estuaries

9.1.3 Modeling Procedure

- Model parameters
  - Model parameters exist only in context of model.
  - Model coefficients

- Model calibration
  - Parameter tuning to fit observed data to predicted data
  - To estimate parameters of model from available information
  - Model becomes less mechanistic (more empirical)
  - Parameter identification problem
9.1 Modeling Mixing in Rivers and Estuaries

- Procedure of Calibration and Verification

<table>
<thead>
<tr>
<th></th>
<th>Data Set I</th>
<th>Data Set II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>$I_1$</td>
<td>$I_2$</td>
</tr>
<tr>
<td>Output</td>
<td>$O_1$</td>
<td>$O_2$</td>
</tr>
<tr>
<td>Parameter</td>
<td>$?$</td>
<td>$P$</td>
</tr>
</tbody>
</table>

i) Calibration

$I_1 \rightarrow \text{Model} (?) \rightarrow \tilde{O}_1$

Fit $\tilde{O}_1$ to $O_1$

Find $P$ (set of values of parameters(coefficients))

ii) Verification

$I_2 \rightarrow \text{Model}(P) \rightarrow \tilde{O}_2$

Predict $\tilde{O}_2$ with calibrated parameter $P$

Compare $\tilde{O}_2$ to $O_2$ to see if $\tilde{O}_2 = O_2$
9.1 Modeling Mixing in Rivers and Estuaries

- Best fit
  - techniques for determining the "best", or "optimal" values of the model coefficients
  i.e., values that make the predicted values and the measured ones sufficiently close
to each other
9.1 Modeling Mixing in Rivers and Estuaries

9.1.4 Numerical Solutions

- **Analytical Solution**
  - A closed-form algebraic expression for temporal and spatial distribution of the constituent
  - easier to use than a numerical model

- **Numerical Solution**
  - The complex water body geometry and flow fields and nonlinearities of the source and sink terms make it impossible to obtain analytical solutions to the differential equation
  - solve using numerical techniques
9.1 Modeling Mixing in Rivers and Estuaries

- Numerical techniques
  - simultaneous solution of a series of mass balances on a number of small fluid elements
  - matrix-inversion methods

- Types of numerical techniques
  1) FDM (Finite Difference Method)
  2) FEM (Finite Element Method)
  3) FVM (Finite Volume Method)
  4) FPM (Finite Particle Method)
9.1 Modeling Mixing in Rivers and Estuaries

- Errors in numerical solutions

- Truncation error: discretization error

- Round-off error: error occurred in the arithmetic operations needed to solve FDE
9.2 Finite Difference Model

9.2.1 Explicit Finite-Difference Methods

- Procedure of finite difference scheme

a. Break $x$ into finite segments of $\Delta x$ in length

b. Subscript all variables and constants, $C_i, U_i, A_i, E_i...etc.$ such that $i$ subscript indicates the value of variable or parameter at point $i$

c. Apply Taylor series expansions to approximate the differentials
9.2 Finite Difference Model

- **Taylor Series expansions**

\[
C_{i+1} = C_i + \Delta x \frac{\partial C_i}{\partial x} + \frac{\Delta x^2}{2} \frac{\partial^2 C_i}{\partial x^2} + \frac{\Delta x^3}{3!} \frac{\partial^3 C_i}{\partial x^3} + O\Delta x^4 \quad (a)
\]

\[
C_{i-1} = C_i - \Delta x \frac{\partial C_i}{\partial x} + \frac{\Delta x^2}{2} \frac{\partial^2 C_i}{\partial x^2} - \frac{\Delta x^3}{3!} \frac{\partial^3 C_i}{\partial x^3} + O\Delta x^4 \quad (b)
\]

\[
\frac{\partial C_i}{\partial x} \equiv \frac{\partial C}{\partial x} \bigg|_{x=i}
\]

\[
\Delta x^2 = (\Delta x)^2
\]

\[
O\Delta x^4 = \text{order of } (\Delta x^4) \text{ and smaller}
\]
9.2 Finite Difference Model

1) **Forward-difference** ← (a)

\[
\frac{\partial C_i}{\partial x} \approx \frac{C_{i+1} - C_i}{\Delta x} - \Delta x \frac{\partial^2 C_i}{\partial x^2} - \frac{\Delta x^2}{3!} \frac{\partial^3 C_i}{\partial x^3} - O\Delta x^3
\]

where \( O\Delta x \sim \text{first-order error} \)

2) **Backward-difference (Upwind difference)** ← (b)

\[
\frac{\partial C_i}{\partial x} \approx \frac{C_i - C_{i-1}}{\Delta x} + \Delta x \frac{\partial^2 C_i}{\partial x^2} - \frac{\Delta x^2}{3!} \frac{\partial^3 C_i}{\partial x^3} + O\Delta x^3
\]

(c) and (d) illustrate the approximations graphically.
9.2 Finite Difference Model

3) Central-difference

Subtract (b) from (a)

\[
\frac{\partial C_i}{\partial x} \approx \frac{C_{i+1} - C_{i-1}}{2\Delta x} - \frac{1}{3} \Delta x^2 \frac{\partial^3 C_i}{\partial x^3}
\]

\(O(\Delta x^2) \sim 2\text{nd-order error}\)

4) Central-difference for 2nd derivative

Add (a) and (b)

\[
\frac{\partial^2 C_i}{\partial x^2} \approx \frac{C_{i+1} - 2C_i + C_{i-1}}{\Delta x^2} - O(\Delta x^2)
\]
9.2 Finite Difference Model

9.2.2 Model Assembling for Explicit Scheme

1D advection-dispersion equation for conservative pollutant is

\[
\frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = \frac{1}{A} \left( \frac{\partial}{\partial x} \left( E A \frac{\partial C}{\partial x} \right) \right)
\]

\[A, E, U = f_n(x)\]

1) Formulation  \textit{Ex-a}

- Explicit solution  \textit{Ex-a}:
  
  ① Forward-difference for time derivative \rightarrow \text{explicit method}
  
  ② Forward-difference for 1st derivative in \(x\)
9.2 Finite Difference Model

\[ \frac{\partial C}{\partial t} \approx \frac{C_{i+1}^{n+1} - C_{i}^{n}}{\Delta t} \]

\[ \frac{\partial C}{\partial x} \approx \frac{C_{i+1}^{n} - C_{i}^{n}}{\Delta t} \]

\[ \frac{1}{A} \left( \frac{\partial}{\partial x} \left( \frac{\partial C}{\partial x} \right) \right) \approx \frac{E_{i}A_{i}(C_{i+1}^{n} - C_{i}^{n}) - E_{i-1}A_{i-1}(C_{i}^{n} - C_{i-1}^{n})}{A_{i} \Delta x^2} \] ← Combination of FD and BD

Substituting & rearranging

\[ C_{i}^{n+1} = C_{i}^{n} - \frac{\Delta t}{\Delta x} u_{i}(C_{i+1}^{n} - C_{i}^{n}) + \frac{E_{i} \Delta t}{\Delta x^2} (C_{i+1}^{n} - C_{i}^{n}) - \frac{E_{i-1}A_{i-1}}{A_{i}} \frac{\Delta t}{\Delta x^2} (C_{i}^{n} - C_{i-1}^{n}) \]

Rearranging further

\[ C_{i}^{n+1} = \left( 1 + \frac{u_{i} \Delta t}{\Delta x} - \frac{E_{i} \Delta t}{\Delta x^2} \frac{E_{i-1}A_{i-1}}{A_{i}} \frac{\Delta t}{\Delta x^2} \right) \left( C_{i}^{n} \right) + \left( \frac{E_{i} \Delta t}{\Delta x^2} - \frac{u_{i} \Delta t}{\Delta x} \right) \left( C_{i+1}^{n} \right) + \frac{E_{i-1}A_{i-1}}{A_{i}} \frac{\Delta t}{\Delta x^2} \left( C_{i-1}^{n} \right) \]
9.2 Finite Difference Model

Let

\[ \frac{u_i \Delta t}{\Delta x} = a_i \]  

\( \rightarrow \) Courant No. \( \rightarrow \) stability

\[ \frac{E_i \Delta t}{\Delta x^2} = b_i \]  

\( \rightarrow \) Peclet No. \( \rightarrow \) numerical dispersion

\[ \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} = d_i \]

Then

\[ C_{i+1}^{n+1} = d_i C_{i-1}^n + (1 + a_i - b_i - d_i) C_i^n + (b_i - a_i) C_{i+1}^n \]  

(1)
9.2 Finite Difference Model

\[ \phi_i + (1 + \phi_i - \phi_{i-1} - \phi_{i+1}) + (\phi_{i+1} - \phi_i) = 1 \]

\[ \therefore C_{i+1}^{n+1} \text{ is weighted average of } C_i^n, C_i^t, \text{and } C_{i+1}^n \]

Solution

Boundary conditions:

1. \( C \) known for all \( x @ t = 0 \)
2. \( C \) known for all \( t @ x = 0 \)

Procedure:

1. Use Eq. (1) to get \( C_1^1, C_2^1, C_3^1 \) etc.
2. Then get \( C_1^2, C_2^2, C_3^2 \) on the basis of \( C_1^1, C_2^1 \) etc.
3. Continue as far in time as desired
9.2 Finite Difference Model
2) Formulation *Ex-b*

① **Forward difference** for time derivative → explicit scheme

② Backward difference for spatial derivative

\[
\frac{\partial C}{\partial t} \approx \frac{C_{i}^{n+1} - C_{i}^{n}}{\Delta t}
\]

\[
\frac{\partial C}{\partial x} \approx \frac{C_{i}^{n} - C_{i-1}^{n}}{\Delta x}
\]

\[
\frac{1}{A} \left( \frac{\partial}{\partial x} E A \frac{\partial C}{\partial x} \right) \approx \frac{E_{i+1} A_{i+1} (C_{i+1}^{n} - C_{i}^{n}) - E_{i} A_{i} (C_{i}^{n} - C_{i-1}^{n})}{A_i \Delta x^2}
\]

Let's include the source/sink term in this time

\[
S_{i} = f(C)
\]
9.2 Finite Difference Model

Substituting and rearranging

\[
C_{i}^{n+1} = C_{i}^{n} - \frac{u_{i} \Delta t}{\Delta x} (C_{i}^{n} - C_{i-1}^{n}) + \frac{E_{i} A_{i+1}}{A_{i}} \frac{\Delta t}{\Delta x^2} (C_{i+1}^{n} - C_{i}^{n})
\]

\[
- \frac{E_{i} \Delta t}{\Delta x^2} (C_{i}^{n} - C_{i-1}^{n}) + f\left(C_{i}^{n}\right) \Delta t
\]

\[
= \left(1 - \frac{u_{i} \Delta t}{\Delta x} - \frac{E_{i} A_{i+1}}{A_{i}} \frac{\Delta t}{\Delta x^2} - \frac{E_{i} \Delta t}{\Delta x^2}\right) C_{i}^{n} + \left(\frac{u_{i} \Delta t}{\Delta x} + \frac{E_{i} \Delta t}{\Delta x^2}\right) C_{i-1}^{n}
\]

\[
+ \frac{E_{i} \Delta t}{\Delta x^2} \frac{A_{i+1}}{A_{i}} C_{i+1}^{n} + f\left(C_{i}^{n}\right) \Delta t
\]

Let \[
\frac{E_{i+1} A_{i+1}}{A_{i}} \frac{\Delta t}{\Delta x^2} = d_{i}
\]

\[
C_{i}^{n+1} = (a_{i} + b_{i}) C_{i-1}^{n} + d_{i} C_{i+1}^{n} + \left(1 - a_{i} - b_{i} - d_{i}\right) C_{i}^{n} + f\left(C_{i}^{n}\right) \Delta t
\]
9.2 Finite Difference Model

Assume first-order decay

\[ S = f(C) = -kC \]

\[ \therefore f(C_i^n) \Delta t = -k_i \Delta t C_i^n \]

\[ \therefore C_i^{n+1} = (a_i + b_i) C_{i-1}^n + (1 - a_i - b_i - d_i - k_i \Delta t) C_i^n + d_i C_{i+1}^n \]

Note that now

\[ \sum \text{Coeffs} \neq 1 \]
9.2 Finite Difference Model

3) Formulation *Ex-cd*

Forward difference for time derivative $\rightarrow$ explicit method

Central difference for spatial derivatives, 1\textsuperscript{st} and 2\textsuperscript{nd} derivatives

\[
\frac{\partial C}{\partial t} \approx \frac{C_{i}^{n+1} - C_{i}^{n}}{\Delta t}
\]
\[
\frac{\partial C}{\partial x} \approx \frac{C_{i+1}^{n} - C_{i-1}^{n}}{2\Delta x}
\]
\[
\frac{\partial^{2} C}{\partial x^{2}} \approx \frac{C_{i+1}^{n} - 2C_{i}^{n} + C_{i-1}^{n}}{\Delta x^{2}}
\]
9.2 Finite Difference Model

Substitute these into 1-D transport equation

\[
\frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = E \frac{\partial^2 C}{\partial x^2}
\]

\[
\frac{C_i^{n+1} - C_i^n}{\Delta t} + u \frac{C_{i+1}^n - C_{i-1}^n}{2\Delta x} = E \frac{C_{i+1}^n - 2C_i^n + C_{i-1}^n}{\Delta x^2}
\]

\[
C_i^{n+1} = \left(1 - 2\frac{E\Delta t}{\Delta x^2}\right)C_i^n + \left(\frac{E\Delta t}{\Delta x^2} - \frac{u\Delta t}{2\Delta x}\right)C_{i+1}^n + \left(\frac{E\Delta t}{\Delta x^2} + \frac{u\Delta t}{2\Delta x}\right)C_{i-1}^n
\]

\[
C_i^{n+1} = \left(\frac{a}{2} + b\right)C_{i-1}^n + (1 - 2b)C_i^n + \left(b - \frac{a}{2}\right)C_{i+1}^n
\]

\[
a = \frac{u\Delta t}{\Delta x}
\]

\[
b = \frac{E\Delta t}{\Delta x^2}
\]
9.2 Finite Difference Model

9.2.3 Error Analysis

(1) **Source of Errors**

- **Mathematical Model**
  - PDE $\bar{C}$

- **Numerical Model**
  - FDE $\tilde{C}$

  - **Truncation error**: due to finite approximations of limiting processes

  - **Roundoff error**: stem from a finite number of digits in a computer word or from initial data

- **Solution to FDE**: $C$

• Errors in machine computations
9.2 Finite Difference Model

- Roundoff Error
  
  (i) Decimal-binary conversion error
  ~ computer converts decimal number to its binary equivalent
  ~ conversion error may be introduced because of finite word length of computer particularly if there is not exact binary equivalent

  (ii) Non decimal-binary conversion error
  ~ if calculation requires more digits than available digits through a machine

  (decimal computer)

[Examples]

(i) \[ 0.625 = \left( \frac{1}{2} \right)^1 + \left( \frac{1}{2} \right)^3 \Rightarrow .101 \]

(ii) \[ 0.626 = \left( \frac{1}{2} \right)^1 + \left( \frac{1}{2} \right)^3 + \left( \frac{1}{2} \right)^{10} + \left( \frac{1}{2} \right)^{16} + \left( \frac{1}{2} \right)^{17} + \left( \frac{1}{2} \right)^{21} + \cdots \Rightarrow .101000 \] (infinite series)
9.2 Finite Difference Model

If binary machine has 20 bits available binary-decimal reconversion to a decimal equivalent with 8-digit accuracy

\[ 0.62599945 \text{ without rounding} \]

\[ 0.62600040 \text{ with rounding} \]

(ii) If decimal computer of capacity of 8 significant digits

\[
\begin{align*}
0.33333333 & \quad 0.33333333 \\
+ 0.33333333 & \quad +0.33333333 \\
\text{add 3,000 times} & \quad \text{add 3,000 times} \\
\text{Expected value} = 999.99999 & \quad +0.33333333 \\
\text{Rounded-off value} = 999.99091 & \quad 1.33333332 \\
\text{Roundoff error} = 0.00908 & \quad \text{True value} \quad \text{Truncated to} \quad \text{8digit} \\
\text{Error} = 0.00000002 & 
\end{align*}
\]
9.2 Finite Difference Model

(2) Lax Equivalence Theorem

Consistency + Stability $\rightarrow$ Convergence

• **Convergence**

The numerical scheme is convergent if for any fixed time $T = n\Delta t$ and fixed location $X = i\Delta x$,

$$C(X,T) \rightarrow \overline{C}(X,T) \quad (or \quad \left| C(X,T) - \overline{C}(X,T) \right| = 0) \quad as \quad \Delta x \rightarrow 0 \quad and \quad \Delta t \rightarrow 0$$

in which $C(X,T) =$ computed value at the fixed point $X, T$ of the FDE

$\overline{C}(X,T) =$ exact solution to the PDE
9.2 Finite Difference Model

- **Consistency**
  The FDE is consistent with the PDE if the local truncation error goes to zero as
  \( \Delta x \rightarrow 0 \) and \( \Delta t \rightarrow 0 \).

- **Stability**
  The numerical scheme is stable if \( e_i^n \) remains bounded as \( n \rightarrow \infty \) for fixed \( \Delta t \)
  \( t \rightarrow \infty \) or as computation proceeds

  in which \( e_i^n = \) roundoff errors
  \[ e_i^n = \tilde{C}(x,t) - C(x,t) \]
  \( C(x,t) \) = computed value of FDE by computer
  \( \tilde{C}(x,t) \) = exact solution to the FDE
9.2 Finite Difference Model

(3) **Analysis of Stability**

Von Neumann Method

Matrix Method

- Explicit solutions
  ~ accurate & easy
  ~ may be unstable → need a stability criterion

1) **Formulation** *Ex-a*

Let \( C_i^n = T_i^n + e_i^n \) \hspace{2cm} (1)

\( C_i^n = \) computed value of FDE by computer

\( T_i^n = \) true (exact) value to FDE at the \( x \) and \( t \) associated with \( i \) and \( n \)

\( e_i^n = \) error at that point
9.2 Finite Difference Model

Substitute this into Formulation a)

\[ e_{i}^{n+1} = \frac{(1 + a_{i} - b_{i} - d_{i})e_{i}^{n} + (b_{i} - a_{i})e_{i+1}^{n} + d_{i}e_{i-1}^{n}}{-T_{i}^{n+1} + (1 + a_{i} - b_{i} - a_{i})T_{i}^{n} + (b_{i} - a_{i})T_{i+1}^{n} + d_{i}T_{i-1}^{n}} \]  

\[ \rightarrow \text{error for newly-calculated concentration depends not only on true concentration (exact solution to FDE) (T-terms) but also on other errors (e-terms)} \]

~ Part ② may not be zero because of truncated terms in formulating FDE out of PDE
9.2 Finite Difference Model

But, we assume truncation error is zero, and worry only about the \( e \)-terms or propagation (magnification) of roundoff errors.

To insure stability (to prevent magnification of errors)

\[
|e_i^{n+1}| \leq \max \left[ |e_{i-1}^n|, |e_i^n|, |e_{i+1}^n| \right]
\]  
(3)

For Formulation \( a \)

\[
|e_i^{n+1}| \leq \left\{ |1 + a_i - b_i - d_i| + |b_i - a_i| + |d_i| \right\} e_i^n
\]

\[
|1 + a_i - b_i - d_i| + |b_i - a_i| + |d_i| \leq 1
\]  
(4)

Absolute values of the coefficients should add to less than one.

Now, since \( a_i, b_i, d_i \geq 0 \), there are 4 possibilities.

\[
i) \quad \text{if} \quad 1 + a_i - b_i - d_i > 0 \quad \& \quad b_i - a_i > 0
\]

then

\[
1 + a_i - b_i - d_i + b_i - a_i + d_i = 1 \quad \leq 1
\]
9.2 Finite Difference Model

which is satisfied for all values of $i$ and which meet these conditions.

\[ \text{ii) if } 1 + a_i - b_i - d_i > 0 \& b_i - a_i \leq 0 \]
\[ 1 + a_i - b_i - f_i - b_i + a_i + f_i \leq 1 \]
\[ 2a_i \leq 2b_i \]
\[ a_i \leq b_i \]

\[ \text{iii) if } 1 + a_i - b_i - d_i < 0 \& b_i - a_i > 0 \]
\[ -1 - a_i + b_i + d_i + b_i - a_i + d_i \leq 1 \]
\[ 2d_i + 2b_i - 2a_i \leq 2 \]
\[ d_i + b_i - a_i \leq 1 \]
\[ \text{if } d_i = b_i \]
\[ \text{then } 2b_i \leq 1 + a_i \]
9.2 Finite Difference Model

iv) if \( 1 + a_i - b_i - d_i < 0 \) & \( b_i - a_i \leq 0 \),
    \[-1 - a_i + b_i + d_i - b_i + a_i + d_i \leq 1 \],
    \( 2d_i \leq 2 \),
    \( d_i \leq 1 \) or \( b_i \leq 1 \) if \( b_i = d_i \).

Combine these conditions.
9.2 Finite Difference Model

- Restrictions on $\Delta x$ and $\Delta t$

  ii) $a \leq b$

  \[
  \frac{u\Delta t}{\Delta x} \leq \frac{E\Delta t}{\Delta x^2} \quad \rightarrow \quad \Delta x \leq \frac{E}{u} \quad (1)
  \]

  iii) $2b \leq 1 + a$

  \[
  2\frac{E\Delta t}{\Delta x^2} \leq 1 + \frac{u\Delta t}{\Delta x} \quad \rightarrow \quad \frac{1}{\Delta t} \geq \frac{2E}{\Delta x^2} - \frac{u}{\Delta x}
  \]

  \[
  \rightarrow \quad \Delta t \leq \frac{1}{2\frac{E}{\Delta x^2} - \frac{u}{\Delta x}} \quad (2)
  \]
9.2 Finite Difference Model

Substitute (1) into (2)

\[
\Delta t < \frac{1}{\frac{2E}{u} - \frac{u}{(\frac{E}{u})^2}} = \frac{1}{\frac{2u^2}{E} - \frac{u^2}{E}} = \frac{1}{\frac{u^2}{E}} = \frac{E}{u^2}
\]

\[
\Delta t < \frac{E}{u^2}
\]

(3)
9.2 Finite Difference Model

2) Formulation \textit{Ex-b}

Consider formulation w/o decay term

\[ |1 - a_i - b_i - d_i| + |a_i + b_i| + |d_i| \leq 1 \]

There are only 2 cases to be considered

\begin{enumerate}
  \item \textit{i) if} \quad 1 - a_i - b_i - d_i \geq 0
  \[ 1 - a_i - b_i - d_i - \alpha_i - \beta_i - \gamma_i + \varphi_i - \varphi_i \leq 1 \]
  \[ 1 \leq 1 \]
  \textit{Satisfied for any values of} \ a_i, b_i, \text{and} \ d_i

  \item \textit{ii) if} \quad 1 - a_i - b_i - d_i \leq 0
  \[ -1 + a_i + b_i + d_i + a_i + b_i + d_i \leq 1 \]
  \[ 2(a_i + b_i + d_i) \leq 2 \]
  \[ a_i + b_i + d_i \leq 1 \quad \text{if} \quad b_i = d_i \\
  a_i + 2b_i \leq 1 \quad a_i \leq 1 - 2b_i \]
\end{enumerate}
9.2 Finite Difference Model

\[ a + 2b \leq 1 \]

\[ \frac{u\Delta t}{\Delta x} + 2 \frac{E\Delta t}{\Delta x^2} \leq 1 \]

\[ \Delta t \leq \frac{1}{u \frac{2E}{\Delta x} + \frac{2E}{\Delta x^2}} \]
3) Formulation *Ex-cd*

\[ |1 - 2b| + \left| \frac{a}{2} + b \right| + \left| b - \frac{a}{2} \right| \leq 1 \]

i) If \( 1 - 2b > 0 \) \& \( b - \frac{a}{2} > 0 \)

\[ 1 - 2b + \frac{a}{2} + b + b - \frac{a}{2} \leq 1 \]

\[ 1 \leq 1 \]

ii) If \( 1 - 2b > 0 \) \& \( b - \frac{a}{2} < 0 \)

\[ 1 - 2b + \frac{a}{2} + b - b + \frac{a}{2} \leq 1 \]

\[ a - 2b \leq 0 \]

\[ a \leq 2b \]
iii) if $1 - 2b < 0 \& b - \frac{a}{2} > 0$

$$-1 - 2b + \frac{a}{2} + b + b - \frac{a}{2} \leq 1$$

$$b \leq \frac{1}{2}$$

iv) if $1 - 2b < 0 \& b - \frac{a}{2} < 0$

$$-1 + 2b + \frac{a}{2} + \beta - \beta + \frac{a}{2} \leq 1$$

$$2b + a \leq 2$$

$$b + \frac{a}{2} \leq 1$$
\[ a \leq 2b \]

\[ \frac{u \Delta t}{\Delta x} \leq 2 \frac{E \Delta t}{\Delta x^2} \]

\[ \Delta x \leq 2 \frac{E}{u} \]

\[ b \leq \frac{1}{2} \]

\[ \frac{E \Delta t}{\Delta x^2} \leq \frac{1}{2} \]

\[ \Delta t \leq \frac{\Delta x^2}{2E} \]
9.2 Finite Difference Model

9.2.4 Implicit Finite-Difference Methods

1) Formulation *Im-c*

Backward difference for \( \frac{\partial C}{\partial t} \)

Forward difference for \( \frac{\partial C}{\partial x} \)

\[
\begin{align*}
\frac{\partial C}{\partial t} & \approx \frac{C_{i}^{n} - C_{i}^{n-1}}{\Delta t} \\
\frac{\partial C}{\partial x} & \approx \frac{C_{i+1}^{n} - C_{i}^{n}}{\Delta x} \\
\frac{1}{A} \frac{\partial}{\partial x} \left( E A \frac{\partial C}{\partial x} \right) & \approx \frac{1}{A \Delta x^{2}} \left\{ E_{i} A_{i} \left( C_{i+1}^{n} - C_{i}^{n} \right) - E_{i-1} A_{i-1} \left( C_{i}^{n} - C_{i-1}^{n} \right) \right\}
\end{align*}
\]
9.2 Finite Difference Model

Substituting and rearranging yields

\[ C_i^n - C_i^{n-1} = -\frac{u_i \Delta t}{\Delta x} (C_{i+1}^n - C_i^n) + \frac{E_i \Delta t}{\Delta x^2} (C_{i+1}^n - C_i^n) - \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} (C_i^n - C_{i-1}^n) \]

\[ (1 - \frac{u_i \Delta t}{\Delta x} + \frac{E_i \Delta t}{\Delta x^2} + \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2}) C_i^n + (\frac{u_i \Delta t}{\Delta x} - \frac{E_i \Delta t}{\Delta x^2}) C_{i+1}^n \]

\[ - \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} C_{i-1}^n = C_i^{n-1} \]

let

\[ a_i = \frac{u_i \Delta t}{\Delta x} \quad b_i = \frac{E_i \Delta t}{\Delta x^2} \quad d_i = \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} \]

\[ (1 - a_i + b_i + d_i) C_i^n + (a_i - b_i) C_{i+1}^n - d_i C_{i-1}^n = C_i^{n-1} \]

\[ \rightarrow C_i^{n-1} = \text{weighted average of } C_i^n, C_{i+1}^n, \text{ and } C_{i-1}^n \]
9.2 Finite Difference Model

We need I.C., UBC, and DBC to solve system of algebraic equation

Let \( L_i = -d_i; \quad M_i = 1 - a_i + b_i + d_i, \quad U_i = a_i - b_i \)

Then \( L_i C_{i-1}^n + M_i C_i^n + U_i C_{i+1}^n = C_i^{n-1} \)

(i) If \( C \) is known

\[
\begin{align*}
\begin{cases}
  x = 0 & (i = 0) \\
  x = \infty & (i = m + 1)
\end{cases}
\end{align*}
\]

\( C(0), C(m + 1) \)

→ Dirichlet (The 1st kind) type B.C.

\( i = 1: \quad L_1 C_0^n + M_1 C_1^n + U_1 C_2^n = C_1^{n-1} \)

\( \rightarrow M_1 C_1^n + U_1 C_2^n = C_1^{n-1} - L_1 C_0^n \) \( \text{Known} \)

\( i = 2 \): \( L_2 C_1^n + M_2 C_2^n + U_2 C_3^n = C_2^{n-1} \)

\( i = m \): \( L_m C_{m-1}^n + M_m C_m^n + U_m C_{m+1}^n = C_m^{n-1} \)

\( \rightarrow L_m C_{m-1}^n + M_m C_m^n = C_m^{n-1} - U_m C_{m+1}^n \) \( \text{Known} \)
9.2 Finite Difference Model

\[
\begin{pmatrix}
M_1 & U_1 & 0 & \cdots & 0 \\
L_2 & M_2 & U_2 & 0 & \cdots & 0 \\
0 & L_3 & M_3 & U_3 & 0 & \cdots & 0 \\
0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\
\end{pmatrix}
\begin{bmatrix}
C_1^n \\
C_2^n \\
C_3^n \\
\vdots \\
C_{m-1}^n \\
C_m^n
\end{bmatrix}
= 
\begin{bmatrix}
C_1^{n-1} - L_1 C_0^n \\
C_2^{n-1} \\
C_3^{n-1} \\
\vdots \\
C_{m-1}^{n-1} \\
C_m^{n-1} - U_m C_{m+1}^n
\end{bmatrix}
\]
9.2 Finite Difference Model

→ All concentrations for one value of \( n \) are solved for simultaneously, and the solution marches in time.

→ Implicit solution

- Tridiagonal matrix → Gaussian elimination
  
  Thomas Algorithm (Stone and Brian, 1963)


(ii) If \( C \) known
@ \( x = 0 \) (\( i = 0 \)) → Dirichlet

And \( \frac{\partial C}{\partial x} \) known
@ \( x = \infty \) (\( i = m + 1 \)) → Neumann (2nd kind)
9.2 Finite Difference Model

Suppose no flux @ boundary → \( \frac{\partial C^n}{\partial x}_{m+1} = 0 \) → Reflecting boundary

\[
\left. \frac{\partial C^n}{\partial x} \right|_{m+1} \approx \frac{C_{m+1}^n - C_m^n}{\Delta x} = 0
\]

\[\therefore \ C_{m+1}^n = C_m^n\]

\[i = m : L_mC_{m-1}^n + M_mC_m^n + U_mC_{m+1}^n = C_{m-1}^{n-1}\]

\[\therefore \ L_mC_{m-1}^n + (M_m + U_m)C_m^n = C_{m-1}^{n-1}\]

\[
\begin{bmatrix}
M_1 & U_1 & 0 \\
L_2 & M_2 & U_2 & 0 \\
L_{m-1} & M_{m-1} & U_{m-1} \\
0 & L_m & M_m + U_m
\end{bmatrix}
\begin{bmatrix}
C_1 \\
C_2 \\
\vdots \\
C_m
\end{bmatrix}^n
= \begin{bmatrix}
C_1^{n-1} - L_1C_0^n \\
C_2^{n-1} \\
\vdots \\
C_m^{n-1}
\end{bmatrix}\]
9.2 Finite Difference Model

[Re] Boundary Conditions

- **Dirichlet (1st type):** \( C(x, y) = f_1(x, y) \) on \( T_1 \)
- **Neumann (2nd type):** \( \frac{\partial C}{\partial n} = f_2(x, y) \) on \( T_2 \)
  \[
  \frac{\partial C}{\partial n} = \text{derivative normal to a boundary} = \frac{\partial C}{\partial x} \text{ or } \frac{\partial C}{\partial y}
  \]
- **Mixed (3rd type):** \( a \frac{\partial C}{\partial n} + bC = f_3(x, y) \) on \( T_3 \)
2) Formulation \textit{lm-d}

\textit{~ most commonly used formulation}

Backward difference for \( \frac{\partial C}{\partial t} \)

Backward difference for \( \frac{\partial C}{\partial x} \)

\[
\frac{\partial C}{\partial t} \approx \frac{C^n_i - C^{n-1}_i}{\Delta t}
\]

\[
\frac{\partial C}{\partial x} \approx \frac{C^n_i - C^n_{i-1}}{\Delta x}
\]

\[
\frac{1}{A} \left( \frac{\partial}{\partial x} EA \frac{\partial C}{\partial x} \right) \approx \frac{1}{A_i \Delta x^2} \left[ E_{i+1} A_{i+1} (C^n_{i+1} - C^n_i) - E_i A_i (C^n_i - C^n_{i-1}) \right]
\]
9.2 Finite Difference Model

Substituting and rearranging yields

\[
(1 + a_i + b_i + d_i)C_i^n - (a_i + b_i)C_{i-1}^n - d_iC_{i+1}^n = C_{i-1}^n
\]

where 

\[
d_i = \frac{E_{i+1}A_{i+1}}{A_i} \frac{\Delta t}{\Delta x^2}
\]

let

\[
L_i = -(a_i + b_i)
\]

\[
M_i = (1 + a_i + b_i + d_i)
\]

\[
U_i = -d_i
\]

then

\[
L_iC_{i-1}^n + M_iC_i^n + U_iC_{i+1}^n = C_{i-1}^n
\]
9.2 Finite Difference Model

3) Formulation \textit{lm-cd}

Backward difference for \( \frac{\partial C}{\partial t} \)

Central difference for \( \frac{\partial C}{\partial x} \)

\[
\frac{\partial C}{\partial t} \approx \frac{C^n_i - C^{n-1}_i}{\Delta t}
\]

\[
\frac{\partial C}{\partial x} \approx \frac{C^n_{i+1} - C^n_{i-1}}{2\Delta x}
\]

Final discretized equation is given as

\[
L_i C^n_{i-1} + M_i C^n_i + U_i C^n_{i+1} = C^{n-1}_i
\]

\[
L_i = -\frac{a_i}{2} - b_i \quad M_i = 1 + 2b_i \quad U_i = \frac{a_i}{2} - b_i \quad a_i = \frac{u_i \Delta t}{\Delta x} \quad b_i = \frac{E_i \Delta t}{\Delta x^2}
\]
9.2 Finite Difference Model

9.2.5 Numerical Dispersion for Finite-Difference Methods

- Numerical Dispersion
  - **Truncation error of Taylor’s series** in converting PDE of diffusion equation into FDE
  - artificial viscosity, numerical dissipation
  - smearing of concentration fronts due to excessive damping

<Ref>
9.2 Finite Difference Model

- **Numerical dispersion of Ex-cd**

From Taylor series expansion, we get

\[
\frac{\partial C}{\partial x} = \frac{C_{i+1}^n - C_{i-1}^n}{2\Delta x} + \frac{\Delta x^2}{3!} \frac{\partial^3 C}{\partial x^3} + O(\Delta x^3) \quad (a) \text{ central}
\]

\[
\frac{\partial C}{\partial t} = \frac{C_{i}^{n+1} - C_{i}^{n}}{\Delta t} - \frac{\Delta t}{2} \frac{\partial^2 C}{\partial t^2} - O(\Delta t^2) \quad (b) \text{ forward}
\]

By the way, think about 1-D transport equation w/o dispersion term (pure advection)

\[
\frac{\partial C}{\partial t} = -u \frac{\partial C}{\partial x} \quad (1)
\]

Differentiate w.r.t \(x\)

\[
\frac{\partial^2 C}{\partial x \partial t} = -u \frac{\partial^2 C}{\partial x^2} \quad (2)
\]
9.2 Finite Difference Model

Differentiate (1) w.r.t. \( t \)

\[
\frac{\partial^2 C}{\partial t^2} = -u \frac{\partial^2 C}{\partial t \partial x} \tag{3}
\]

(2):

\[
\frac{\partial^2 C}{\partial x \partial t} = -u \frac{\partial^2 C}{\partial x^2} \]

(3):

\[
\frac{\partial^2 C}{\partial t \partial x} = - \frac{1}{u} \frac{\partial^2 C}{\partial t^2} \]

\[
\therefore -u \frac{\partial^2 C}{\partial x^2} = - \frac{1}{u} \frac{\partial^2 C}{\partial t^2} \]

\[
\therefore \frac{\partial^2 C}{\partial t^2} = u^2 \frac{\partial^2 C}{\partial x^2} \tag{4}
\]
9.2 Finite Difference Model

Formulate (1) with (a) and (b)

\[
\frac{C_{i}^{n+1} - C_{i}^{n}}{\Delta t} + \frac{\Delta t \partial^2 C}{2 \partial t^2} = -u \left( \frac{C_{i+1}^{n} - C_{i-1}^{n}}{2\Delta x} \right) + O(\Delta t^2, \Delta x^2) \quad (5)
\]

Substitute (4) into (5)

\[
\frac{C_{i}^{n+1} - C_{i}^{n}}{\Delta t} = -u \left( \frac{C_{i+1}^{n} - C_{i-1}^{n}}{2\Delta x} \right) + \frac{\Delta t}{2} u^2 \frac{\partial^2 C}{\partial x^2} + O(\Delta t^2, \Delta x^2)
\]

Let \( E_n = \text{numerical dispersion coefficient} \)

\[
E_n = \frac{\Delta t}{2} u^2 = \frac{u\Delta x}{2} a \quad (a = \frac{u\Delta t}{\Delta x} = \text{Courant No})
\]

\[
\frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = E_n \frac{\partial^2 C}{\partial x^2} + O(\Delta t^2, \Delta x^2) \quad (6)
\]
9.2 Finite Difference Model

- **Numerical dispersion of \( lm-c \)**

\[
C_i^{n-1} = C_i^n - \Delta t \frac{\partial C}{\partial t} + \frac{\Delta t^2}{2} \frac{\partial^2 C}{\partial t^2} - O(\Delta t^3)
\]

\[
\frac{C_i^n - C_i^{n-1}}{\Delta t} = \frac{\partial C}{\partial t} - \frac{\Delta t}{2} \frac{\partial^2 C}{\partial t^2} + O(\Delta t^2) \quad (c) \text{ backward}
\]

\[
C_i^{n+1} = C_i^n + \Delta x \frac{\partial C_i}{\partial x} + \frac{\Delta x^2}{2} \frac{\partial^2 C_i}{\partial x^2} + O(\Delta x^3)
\]

\[
\frac{C_{i+1}^n - C_i^n}{\Delta x} = \frac{\partial C_i}{\partial x} + \frac{\Delta x}{2} \frac{\partial^2 C_i}{\partial x^2} + O(\Delta x^2) \quad (d) \text{ forward}
\]
9.2 Finite Difference Model

Substituting (c) and (d) into (1)

\[ \frac{\partial C}{\partial t} + \frac{\Delta t}{2} \frac{\partial^2 C}{\partial t^2} = -u \left\{ \frac{\partial C}{\partial x} - \frac{\Delta x}{2} \frac{\partial^2 C}{\partial x^2} \right\} + O(\Delta t^2 + \Delta x^2) \]  

(7)

Substituting (4) into (7) yields

\[ \frac{\partial C}{\partial t} = -u \frac{\partial C}{\partial x} + \left\{ \frac{u \Delta x}{2} - \frac{\Delta t}{2} u^2 \right\} \frac{\partial^2 C}{\partial x^2} + O(\Delta t^2 + \Delta x^2) \]

Numerical dispersion (TE)

Define numerical dispersion coefficient as

\[ E_n = \frac{u \Delta x}{2} \left( 1 - \frac{u \Delta t}{\Delta x} \right) = \frac{u \Delta x}{2} (1 - a) \]

Then

\[ \frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = E_n \frac{\partial^2 C}{\partial x^2} + O(\Delta t^2 + \Delta x^2) \]

If we include real dispersion term

\[ \frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = \underbrace{(E + E_n)}_{E_c = \text{Computed dispersion}} \frac{\partial^2 C}{\partial x^2} \]  

(8)
9.2 Finite Difference Model

- How to remove $E_n$
  (i) Choose and $\Delta t$ and $\Delta x$ such that $E_n = 0$

$$E_n = \frac{u \Delta t}{2} (1 - a) = 0$$

$$\therefore a = \frac{u \Delta t}{\Delta x} = 1$$  \hspace{1cm} (9)

However, stability criterion for Formulation Ex-b is

$$\frac{u \Delta t}{\Delta x} + \frac{2 E \Delta t}{\Delta x^2} \leq 1$$  \hspace{1cm} (10)

If we make $\frac{u \Delta t}{\Delta x} = 1$ then (10) becomes

$$\frac{E \Delta t}{\Delta x^2} \leq 0$$  \hspace{1cm} (11)

Therefore we have to choose $\Delta t$ and $\Delta x$ satisfying both (9) & (11) $\rightarrow$ impossible
9.2 Finite Difference Model

<Example> For Ex-b

\[ u = 1 \quad \Delta x = 2, \Delta t = 1 \]

\[ a = \frac{1 \cdot 1}{2} = \frac{1}{2} \]

\[ E_n = \frac{1(2)}{2} \left( 1 - \frac{1}{2} \right) = \frac{1}{2} \]

SC for Ex - b: \[ \frac{1}{2} + \frac{2(1)}{(2)^2} E = \frac{1}{2} + \frac{E}{2} \leq 1 \]

\[ \frac{E}{2} \leq \frac{1}{2} \]

\[ E \leq \frac{1}{4} \]
(ii) Dispersion correction technique

\[ E_c = E \]

For Formulation \( Ex-b \), subtract \( E_n \) from \( E_c \)

\[ E'_c = E + E_n - E_n = E \]

(iii) make \( \Delta x \) and \( \Delta t \) small

\[ \rightarrow \text{make } E_n \text{ small} \]
### 9.2 Finite Difference Model

#### Summary

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Numerical dispersion, $E_n$</th>
<th>measure to ND</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Ex - a$</td>
<td>$-\frac{u\Delta x}{2}(1 + a)$</td>
<td>$Add(-E_n)$</td>
</tr>
<tr>
<td>$Ex - b$</td>
<td>$\frac{u\Delta x}{2}(1 - a)$</td>
<td>$Subtract E_n (Be careful when E &lt; E_n)$</td>
</tr>
<tr>
<td>$Ex - cd$</td>
<td>$\frac{u^2}{2}\Delta t$</td>
<td>$Subtract E_n$</td>
</tr>
<tr>
<td>$Im - c$</td>
<td>$\frac{u\Delta x}{2}(1 - a)$</td>
<td>$Make \ a = 1$</td>
</tr>
<tr>
<td>$Im - d$</td>
<td>$\frac{u\Delta x}{2}(1 + a)$</td>
<td>$Subtract E_n$</td>
</tr>
<tr>
<td>$Im - cd$</td>
<td>$\frac{u^2}{2}\Delta t$</td>
<td>$Subtract E_n$</td>
</tr>
</tbody>
</table>
9.2 Finite Difference Model

9.2.6 Lagrangian Formulations

- Lagrangian approach
  ~ Observer is traveling at the same speed as the parcel of water under observation

- Two-step explicit method (Bella & Dobbins, 1968)
  → Two processes of the advection-diffusion equation are assumed to occur sequentially rather than simultaneously as in the prototype

1) 1st step (advection process)
  ~ to translate the pollutant downstream for one-time step in Eulerian frame

\[
C^n_i = C^{n+1}_i \\
C^n_0 = C^{n+1}_0
\]
2) 2nd step (dispersion process)

~ to calculate new values on the \( n+1 \) row using only the dispersion in Lagrangian frame

\[
\frac{C_i^{n+1} - C_i^n}{\Delta t} = \frac{E}{\Delta x^2} (C_{i-1}^n - 2C_i^n + C_{i+1}^n)
\]

\[
C_i^{n+1} = C_i^n + \frac{E \Delta t}{\Delta x^2} (C_{i-1}^n - 2C_i^n + C_{i+1}^n)
\]
9.2 Finite Difference Model

1) Formulation \(La-e\)

\[ \frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = \frac{1}{A} \left( \frac{\partial}{\partial x} \frac{EA}{\partial x} \frac{\partial C}{\partial x} \right) + S \]

Forward difference formula at the \(i+1\) grid point for \(\frac{\partial C}{\partial t}\)

\[ \frac{\partial C}{\partial t} \approx \frac{C_{i+1}^{n+1} - C_{i+1}^{n}}{\Delta t} \]

Forward difference formula for \(\frac{\partial C}{\partial x}\)

\[ \frac{\partial C}{\partial x} \approx \frac{C_{i+1}^{n} - C_{i}^{n}}{\Delta x} \]

Eulerian formulation for second derivative

\[ \frac{1}{A} \left( \frac{\partial}{\partial x} \frac{EA}{\partial x} \frac{\partial C}{\partial x} \right) \approx \frac{1}{A_i \Delta x^2} \left[ E_i A_i \left( C_{i+1}^{n} - C_{i}^{n} \right) - E_{i-1} A_{i-1} \left( C_{i}^{n} - C_{i-1}^{n} \right) \right] \]
9.2 Finite Difference Model

Substituting this into Governing Eq. yields

\[ C_{i+1}^{n+1} = C_i^n - \frac{u_i \Delta t}{\Delta x} (C_{i+1}^n - C_i^n) + \frac{E_i \Delta t}{\Delta x^2} (C_{i+1}^n - C_i^n) - \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} (C_i^n - C_{i-1}^n) + S_i \Delta t \]

Rearranging further gives

\[ C_{i+1}^{n+1} = (1 - \frac{u_i \Delta t}{\Delta x} - \frac{E_i \Delta t}{\Delta x^2}) C_i^n + \left( \frac{u_i \Delta t}{\Delta x} - \frac{E_i \Delta t}{\Delta x^2} - \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} \right) C_i^n + \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} C_{i-1}^n + S_i \Delta t \]

Let \( a_i = \frac{u_i \Delta t}{\Delta x} \), \( b_i = \frac{E_i \Delta t}{\Delta x^2} \), \( d_i = \frac{E_{i-1} A_{i-1} \Delta t}{A_i \Delta x^2} \)

Then

\[ C_{i+1}^{n+1} = (1 - a_i + b_i) C_{i+1}^n + (a_i - b_i - d_i) C_i^n + d_i C_{i-1}^n + S_i \Delta t \]

→ We need 2 UBC and IC, need no DBC
9.2 Finite Difference Model

- Numerical Dispersion of Formulation $La-e$

Pure Advection Problem

$$\frac{C_{i+1}^{n+1} - C_{i+1}^n}{\Delta t} = -u \frac{C_{i+1}^n - C_i^n}{\Delta x} \quad (1)$$

Taylor Series expansion in $t$ direction

$$C_{i+1}^{n+1} = C_{i+1}^n + \Delta t \frac{\partial C}{\partial t} + \frac{\Delta t^2}{2} \frac{\partial^2 C}{\partial t^2} + O(\Delta t^3)$$

$$\Rightarrow \frac{C_{i+1}^{n+1} - C_{i+1}^n}{\Delta t} = \frac{\partial C}{\partial t} + \frac{\Delta t}{2} \frac{\partial^2 C}{\partial t^2} + O(\Delta t^2) \quad (2)$$

Taylor Series expansion in $x$ direction

$$C_{i+1}^n = C_i^n + \Delta x \frac{\partial C}{\partial x} + \frac{\Delta x^2}{2} \frac{\partial^2 C}{\partial x^2} + O(\Delta x^3)$$

$$\Rightarrow \frac{C_{i+1}^n - C_i^n}{\Delta x} = \frac{\partial C}{\partial x} + \frac{\Delta x}{2} \frac{\partial^2 C}{\partial x^2} + O(\Delta x^2) \quad (3)$$
9.2 Finite Difference Model

Substitute ② & ③ into ①

\[
\frac{\partial C}{\partial t} + \frac{\Delta t \partial^2 C}{2 \partial t^2} + O(\Delta t^2) = -u \left\{ \frac{\partial C}{\partial x} + \frac{\Delta x \partial^2 C}{2 \partial x^2} + O(\Delta x^2) \right\}
\]

\[
\frac{\partial C}{\partial t} = -u \frac{\partial C}{\partial x} - \frac{\Delta t}{2} \left( \frac{\partial^2 C}{\partial t^2} - \frac{\Delta x}{2} \frac{\partial^2 C}{\partial x^2} + O(\Delta x^2 + \Delta t^2) \right)
\]

\[
\therefore \frac{\partial C}{\partial t} = -u \frac{\partial C}{\partial x} - \frac{\Delta x (1 + a)}{2} \frac{\partial^2 C}{\partial x^2} + O(\Delta x^2 + \Delta t^2)
\]

\[
\therefore E_n = \frac{-u \Delta x}{2} (1 + a)
\]

• Stability Criteria

\[
1 - a_i + b_i \geq 0 \quad \& \quad a - b_i - d_i \geq 0
\]

\[
\Delta t \leq \frac{\Delta x^2}{u \Delta x - E} \quad \Delta x \geq \frac{2E}{u}
\]
9.2 Finite Difference Model

2) Formulation \( La-f \)

~ implicit method

Backward difference for \( \frac{\partial C}{\partial t} \) at the \( i\)-l grid point

Backward difference for \( \frac{\partial C}{\partial x} \)

\[
\frac{\partial C}{\partial t} \approx \frac{C_{i-1}^n - C_{i-1}^{n-1}}{\Delta t}
\]

\[
\frac{\partial C}{\partial x} \approx \frac{C_i^n - C_{i-1}^n}{\Delta x}
\]

\[
(a_i + d_i - 1)C_{i-1}^n + (-a_i - b_i - d_i)C_i^n + b_iC_{i+1}^n + S_i \Delta t = C_{i-1}^{n-1}
\]

- Numerical Dispersion

\[
E_n = \frac{u \Delta x}{2} (1 + a) \sim Im - d
\]
9.2 Finite Difference Model

9.2.7 Crank-Nicholson Scheme

1) Formulation \( CN-b \)

\[ \frac{C_{i}^{n+1} - C_{i}^{n}}{\Delta t} = -u \frac{\Delta x}{\Delta x} (C_{i}^{e} - C_{i-1}^{e}) + \frac{E}{\Delta x^2} (C_{i+1}^{e} - 2C_{i}^{e} + C_{i-1}^{e}) \]

\[ \varepsilon = n \quad \rightarrow \text{Explicit} \quad \text{\Rightarrow 4 point scheme} \]

\[ \varepsilon = n + 1 \quad \rightarrow \text{Implicit} \]

\[ \varepsilon = n + \frac{1}{2} \quad \rightarrow \text{Crank-Nicholson} \]

\[ C_{i}^{n+\frac{1}{2}} = \frac{1}{2} (C_{i}^{n} + C_{i}^{n+1}) \]

\[ \therefore \quad \frac{C_{i}^{n+1} - C_{i}^{n}}{\Delta t} = -u \left\{ \frac{1}{2} (C_{i}^{n} + C_{i}^{n+1}) - \frac{1}{2} (C_{i-1}^{n} + C_{i-1}^{n+1}) \right\} \]

\[ + \frac{E}{\Delta x^2} \left\{ \frac{1}{2} (C_{i+1}^{n} + C_{i+1}^{n+1}) - \frac{1}{2} \cdot 2(C_{i}^{n} + C_{i}^{n+1}) + \frac{1}{2} (C_{i-1}^{n} + C_{i-1}^{n+1}) \right\} \]
9.2 Finite Difference Model

\[
\left( \frac{E \Delta t}{2 \Delta x^2} - \frac{u \Delta t}{2 \Delta x} \right) C_{i-1}^{n+1} + \left( 1 + \frac{u \Delta t}{2 \Delta x} + \frac{E \Delta t}{\Delta x^2} \right) C_i^{n+1} - \frac{E \Delta t}{2 \Delta x^2} C_{i+1}^{n+1}
\]

\[
= \left( \frac{E \Delta t}{2 \Delta x^2} + \frac{u \Delta t}{2 \Delta x} \right) C_{i-1}^n + \left( 1 - \frac{u \Delta t}{2 \Delta x} - \frac{E \Delta t}{\Delta x^2} \right) C_i^n + \frac{E \Delta t}{2 \Delta x^2} C_{i+1}^n
\]

\[
[A] \{C\}^{n+1} = [B] \{C\}^n + \{b\}
\]

\[
[A],[B] \rightarrow \text{Tridiagonal method}
\]

C-N method \rightarrow O(\Delta x + \Delta t^2)

Fully Implicit \rightarrow O(\Delta x + \Delta t)

\[6 \text{ Point scheme}\]

3 knowns @ time level n
3 unknowns @ time level n+1
9.2 Finite Difference Model

2) Formulation \( CN-cd \)

\[
\frac{\partial C}{\partial x} 
\]

\[
C_{i}^{n+1} - C_{i}^{n} = -\frac{u\Delta t}{2\Delta x} \left( C_{i+1}^{e} - C_{i-1}^{e} \right) + \frac{E\Delta t}{\Delta x^2} \left( C_{i+1}^{e} - 2C_{i}^{e} + C_{i-1}^{e} \right)
\]

\[
\therefore \quad C_{i}^{n+1} - C_{i}^{n} = -\frac{u\Delta t}{2\Delta x} \left\{ \frac{1}{2} \left( C_{i+1}^{n} + C_{i+1}^{n+1} \right) - \frac{1}{2} \left( C_{i-1}^{n} + C_{i-1}^{n+1} \right) \right\} + \frac{E\Delta t}{\Delta x^2} \left\{ \frac{1}{2} \left( C_{i+1}^{n} + C_{i+1}^{n+1} \right) - \frac{1}{2} \cdot 2 \left( C_{i}^{n} + C_{i}^{n+1} \right) + \frac{1}{2} \left( C_{i-1}^{n} + C_{i-1}^{n+1} \right) \right\}
\]

\[
\left( \frac{E\Delta t}{2\Delta x^2} - \frac{u\Delta t}{4\Delta x} \right) C_{i-1}^{n+1} + \left( 1 + \frac{E\Delta t}{\Delta x^2} \right) C_{i}^{n+1} + \left( \frac{u\Delta t}{4\Delta x} - \frac{E\Delta t}{2\Delta x^2} \right) C_{i+1}^{n+1}
\]

\[
= \left( \frac{E\Delta t}{2\Delta x^2} + \frac{u\Delta t}{4\Delta x} \right) C_{i-1}^{n} + \left( 1 - \frac{E\Delta t}{\Delta x^2} \right) C_{i}^{n} + \left( \frac{E\Delta t}{2\Delta x^2} - \frac{u\Delta t}{4\Delta x} \right) C_{i+1}^{n}
\]
9.3 Finite Element Model

9.3.1 Introduction

- Difference between FDM and FEM solutions to PDE

\[
\frac{\partial C}{\partial t} = \frac{\partial}{\partial x}\left( E \frac{\partial C}{\partial x} - uC \right)
\]

\[\text{①}\]

\begin{tikzpicture}
  \node (pde) {PDE};
  \node (fdm) [below left of=pde] {FDM};
  \node (fem) [below right of=pde] {FEM};
  \node (derivatives) [below of=fdm] {Approximating Derivatives};
  \node (system) [above right of=derivatives] {System of Linear Algebraic Eq.};
  \node (integral) [below of=system] {Approximating Solution → Integral Eq.};
  \node (numerical) [below of=integral] {Numerical Solution (Approximate Solution)};
  \node (matrix) [above right of=integral] {Matrix Solver};

  \draw [->] (pde) -- (fdm);
  \draw [->] (pde) -- (fem);
  \draw [->] (fdm) -- (derivatives);
  \draw [->] (fem) -- (derivatives);
  \draw [->] (derivatives) -- (system);
  \draw [->] (system) -- (integral);
  \draw [->] (integral) -- (numerical);
  \draw [->] (matrix) -- (numerical);
\end{tikzpicture}
9.3 Finite Element Model

9.3.1 Introduction

FDM: A domain of interest is replaced by a set of discrete points. Then, the function of $C$ is represented by an approximate function using Taylor series.

FEM: A domain is divided into subdomains (finite elements). Then, the unknown function $C$ is represented by an interpolating polynomials within each element.
9.3 Finite Element Model

- Procedure (Summary) of FEM
  
i) Discretize domain into elements  
ii) Select basis functions  
iii) Derive an integral equation based on Method of Weighted Residuals (MWR)  
iv) Compute element matrix and vectors  
v) Assemble global matrix and vectors  
vi) Incorporate boundary conditions  
vii) Use a finite difference for time discretization  
viii) Solve a system of simultaneous linear algebraic equation
9.3 Finite Element Model

9.3.2 FEM Formulation

(1) Domain Discretization

\[ C(x) = \text{true and unknown solution to PDE} \]

\[ \hat{C}(x) = \text{approximate solution} \]

\[ \text{continuous function of } x \]

\[ \text{piecewise continuous function} \]
9.3 Finite Element Model

We may approximate the true solution by a polynomial

\[
\hat{C}^e(x) = \sum_{j=1}^{m} C_j \phi_j^e(x) \tag{2}
\]

in which

\[ \phi_j = \text{basis function (shape, approximate) function} \]

Now, we are seeking the "best" value of the \( C_j \) to give us the best values for \( \hat{C}^e(x) \)
9.3 Finite Element Model

(2) Selection of Basis Function

1) Lagrangian interpolating polynomials

\[ \phi_j(x) = \prod_{k=1}^{m} \frac{x - x_k}{x_j - x_k} \]

i) Linear function: \( m = 2 \)

\[ \phi_1(x) = \frac{x - x_2}{x_1 - x_2} \]
\[ \phi_2(x) = \frac{x - x_1}{x_2 - x_1} \]

\[ \therefore \hat{C}^e(x) = C_1\phi_1(x) + C_2\phi_2(x) \]
9.3 Finite Element Model

ii) Quadratic function: $m = 3$

$$\phi_1(x) = \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)}$$

$$\phi_2(x) = \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}$$

$$\phi_3(x) = \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}$$

\[ \therefore \hat{C}^e(x) = C_1\phi_1(x) + C_2\phi_2(x) + C_3\phi_3(x) \]

2) Hermitian interpolating polynomials

~ interpolate $C(x_i)$ and $\left. \frac{dC}{dx} \right|_{x_i}$ (function and slope)

$$\hat{C}^e(x) = \sum_{j=1}^{m} \left[ C_j\phi_j^{(0)} + \left. \frac{dC}{dx} \right|_j \phi_j^{(1)} \right]$$
9.3 Finite Element Model

(3) Method of Weighted Residuals

- Formulation of approximating integral equation

\[ \text{Variational Method} \]

\[ \text{Method of Weighted Residuals (MWR)} \]

Select MWR

Substitute (2) into (1)

\[
\frac{\partial \hat{C}}{\partial t} - \frac{\partial}{\partial x} \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) \neq 0 = R(x,t) \rightarrow \text{Residual} \quad (3)
\]

If \( \hat{C} = C \) then \( R(x,t) = 0 \)

But \( \hat{C} \neq C \) \( R(x,t) \neq 0 \)

So, in the MWR, an attempt is made to force this residual to zero through selection of the constant \( C_j \) \( (j = 1, 2, \ldots, M) \).
9.3 Finite Element Model

Let’s set the weighted integrals of the residual to zero → MWR

\[ \int_{\Omega} R(x,t) \omega_i(x) \, d\Omega = 0, \quad i = 1, 2, \ldots, M \rightarrow \text{Integral Eq.} \]  

\[ \int \left\{ \frac{\partial \hat{C}}{\partial t} - \frac{\partial}{\partial x} \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) \right\} \omega_i(x) \, dx = 0 \]

There are several MWRs which is distinguished by the choice of weighting function \( \omega_i \)

1) Galerkin method: \( \omega_i = \phi_i(x) \)

2) Subdomain method
   ~ divide domain \( B \) into \( M \) subdomains \( B_i \)

\[ \omega_j = \begin{cases} 1, & \text{x in } B_i \\ 0, & \text{x not in } B_i \end{cases} \]
3) Collocation method

$M$ point $x_i$ (collocation points) are specified in $B$ and weighting functions are Dirac delta functions

$$\omega_i = \delta(x - x_i)$$

which have the property that

$$\int_B R(x) \omega_i dx = R(x_i) = 0$$

4) Least Squares Method

$$\omega_i = p(x) \frac{\partial R}{\partial a_i}$$

$p(x) =$ arbitrary positive function

minimize the integrated square residual w.r.t $a_i$

$$I = \int p(x) R^2(x) dx$$

$$\therefore \frac{\partial I}{\partial a_i} = 0 \quad (i = 1, 2, \ldots, M)$$
9.3 Finite Element Model

[Re] Basis function vs Weighting function

The unknown function $C$ is represented by an interpolating polynomials within each element

$$u(\cdot) \approx \hat{u}(\cdot) = \sum_{j=1}^{N} u_j \phi_j(\cdot), \quad j = 1, 2, \ldots, N$$

$u_j =$ undetermined coefficient

$\phi_j(\cdot) =$ basis function over both time and space

In MWR, the objective is to select the undetermined coefficients $u_j$ such that this residual is minimized in some sense.

$$\int_I \int_V R(\cdot) \omega_i(\cdot) dvdt = 0, \quad i = 1, 2, \ldots, N$$

$\omega_j(\cdot) =$ weighting function
9.3 Finite Element Model

- Galerkin Method

Weighting function is chosen to be the basis function

$$\int_{t}^{\cdot} \int_{v} R(\cdot) \phi_i(\cdot) dvdt = 0, \quad i = 1, 2, \ldots, N$$

![Diagram of basis and weighting function](image)
9.3 Finite Element Model

- Subdomain Method

\[ \int_{V} R(x) \omega_i \, dv = 0, \quad i = 1, 2, \ldots, N \]

where

\[ \omega_i = \begin{cases} 
1, & (x, y, z) \text{ in } v_i^* \\
0, & (x, y, z) \text{ not in } v_i^* 
\end{cases} \]

\~ integrations are less tedious than those in Galerkin's method
9.3 Finite Element Model

- Collocation Method

Weighting function is chosen to be the Dirac delta

\[ \omega_i = \delta(x - x_i) \]

\[ \int_{t_i}^{t_{i+1}} R(\cdot) \delta_i(\cdot) dvdt = 0, \quad i = 1, 2, 3, \ldots, N \]

~ Calculate the value of residual at the selected points

\[ \int_{t_i}^{t_{i+1}} a(\cdot) \delta_i(x - x_i, y - y_i, z - z_i, t - t_i) dvdt \equiv a \bigg|_{x_i, y_i, z_i, t_i} \]
9.3 Finite Element Model

[Example]

\[ \frac{dT}{dt} + k(T - T_e) = 0 \]

\[ 0 \leq t \leq 1 \]

\[ T(t = 0) = 1 \]

\[ k = 2 \quad ; \quad T_e = \frac{1}{2} \]

\[ \phi_i = \begin{cases} 
\frac{t - t_{i-1}}{t_i - t_{i-1}}, & t_{i-1} \leq t \leq t_i \\
\frac{t_{i+1} - t}{t_{i+1} - t_i}, & t_i \leq t \leq t_{i+1} 
\end{cases} \]

\[ i = \text{nodal points} \]
9.3 Finite Element Model

i) \[ T \approx \hat{T} = \sum_{j=1}^{3} T_j \phi_j (t) \]

ii) \[ \int_{t} R(t) \phi_i (t) dt = 0, \quad i = 1, 2, 3 \]
\[ \int_{t} \left\{ \frac{d \hat{T}}{dt} + k(\hat{T} - T_e) \right\} \phi_i (t) dt = 0 \]
\[ \int_{t} \left\{ \sum_{j=1}^{3} T_j \left( \frac{d \phi_j}{dt} + k\phi_j \right) - kT_e \right\} \phi_i (t) dt = 0, \quad i = 1, 2, 3 \]

Apply Galerkin method

\[ \sum_{j=1}^{3} T_j \int_{0}^{1} \left\{ \frac{d \phi_j}{dt} + k\phi_j \right\} \phi_i dt = \int_{0}^{1} kT_e \phi_i dt, \quad i = 1, 2, 3 \]
\[ i = 1 : \sum_{j=1}^{3} T_j \int_{0}^{1} \left\{ \frac{d \phi_j}{dt} + k\phi_j \right\} \phi_1 dt = \int_{0}^{1} kT_e \phi_1 dt \]
9.3 Finite Element Model

\[ i = 1 \quad T_1 \int_0^1 \left\{ \frac{d\phi_1}{dt} + k\phi_1 \right\} \phi_1 dt + T_2 \int_0^1 \left\{ \frac{d\phi_2}{dt} + k\phi_2 \right\} \phi_2 dt + T_3 \int_0^1 \left\{ \frac{d\phi_3}{dt} + k\phi_3 \right\} \phi_3 dt = \int_0^1 kT_e \phi_1 dt \]

\[ i = 2 \quad T_1 \int_0^1 \left\{ \frac{d\phi_1}{dt} + k\phi_1 \right\} \phi_2 dt + T_2 \int_0^1 \left\{ \frac{d\phi_2}{dt} + k\phi_2 \right\} \phi_2 dt + T_3 \int_0^1 \left\{ \frac{d\phi_3}{dt} + k\phi_3 \right\} \phi_2 dt = \int_0^1 kT_e \phi_2 dt \]

\[ i = 3 \quad T_1 \int_0^1 \left\{ \frac{d\phi_1}{dt} + k\phi_1 \right\} \phi_3 dt + T_2 \int_0^1 \left\{ \frac{d\phi_2}{dt} + k\phi_2 \right\} \phi_3 dt + T_3 \int_0^1 \left\{ \frac{d\phi_3}{dt} + k\phi_3 \right\} \phi_3 dt = \int_0^1 kT_e \phi_3 dt \]

\[ \phi_1 \phi_3 = 0 \]
9.3 Finite Element Model

iv) Expansion yields the following matrix equation

\[
\begin{bmatrix}
\int_0^1 \left( \frac{d\phi_1}{dt} + k\phi_1 \phi_1 \right) dt & \int_0^1 \left( \frac{d\phi_2}{dt} + k\phi_2 \phi_2 \right) dt & 0 \\
\int_0^1 \left( \frac{d\phi_1}{dt} + k\phi_1 \phi_2 \right) dt & \int_0^1 \left( \frac{d\phi_2}{dt} + k\phi_2 \phi_2 \right) dt & \int_0^1 \left( \frac{d\phi_2}{dt} + k\phi_2 \phi_2 \right) dt \\
0 & \int_0^1 \left( \frac{d\phi_2}{dt} + k\phi_2 \phi_3 \right) dt & \int_0^1 \left( \frac{d\phi_3}{dt} + k\phi_3 \phi_3 \right) dt
\end{bmatrix}
\begin{bmatrix}
T_1 \\
T_2 \\
T_3
\end{bmatrix} =
\begin{bmatrix}
\int_0^1 kT_e \phi_1 dt \\
\int_0^1 kT_e \phi_2 dt \\
\int_0^1 kT_e \phi_3 dt
\end{bmatrix}
\]

\[
\begin{bmatrix}
-1 + \frac{k}{3} & 1 + \frac{k}{6} & 0 \\
\frac{1}{2} & -1 + \frac{k}{6} & \frac{2k}{3} \\
0 & -1 + \frac{k}{6} & 1 + \frac{k}{3}
\end{bmatrix}
\begin{bmatrix}
1 \\
T_2 \\
T_3
\end{bmatrix} =
\begin{bmatrix}
\frac{kT_e}{2} \\
kT_e \\
\frac{kT_e}{2}
\end{bmatrix}
\]
9.3 Finite Element Model

- **Basis functions (Interpolation)**

\[ C(x,t) \approx \hat{C}(x,t) = \sum_{j=1}^{m} \hat{C}_j \phi_j^e(x,t) = \sum_{j=1}^{m} \hat{C}_j(t) \phi_j^e(x) \]

\[ \frac{\partial \hat{C}}{\partial x} = \sum_{j=1}^{m} C_j \frac{\partial \phi_j}{\partial x} \quad \frac{\partial \hat{C}}{\partial t} = \sum_{j=1}^{m} \frac{dC}{dt} \phi_j(x) \]

- **Natural coordinate system for element basis function**

(Dimensionless \( \xi \) coordinate system where \(-1 \leq \xi \leq 1\))

i) **Linear**

\[ \phi_1^e(\xi) = \frac{1}{2}(1-\xi) \quad \phi_2^e(\xi) = \frac{1}{2}(1+\xi) \]

\[ \frac{d\phi_1^e}{d\xi} = -\frac{1}{2} \quad \frac{d\phi_2^e}{d\xi} = \frac{1}{2} \]
9.3 Finite Element Model

ii) Quadratic

\[ \phi_{-1}(\xi) = -\frac{1}{2} \xi (1 - \xi) \]

\[ \phi_0(\xi) = 1 - \xi^2 \]

\[ \phi_1(\xi) = \frac{1}{2} \xi (1 + \xi) \]
9.3 Finite Element Model

- Galerkin method

Select the basis functions as the weighting functions

\[ \omega_i = \phi_i \]

Thus the weighted integral equation of the residuals becomes

\[
\int_{\Omega^e} \left\{ \frac{\partial \hat{C}}{\partial t} - \frac{\partial}{\partial x} \left( E \frac{\partial \hat{C}}{\partial t} - u \hat{C} \right) \right\} \phi_i \, dx = 0
\]

\[
\int_{\Omega^e} \frac{\partial \hat{C}}{\partial t} \phi^e_i \, dx - \int_{\Omega^e} \frac{\partial}{\partial x} \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) \phi^e_i \, dx = 0
\] ⑥
9.3 Finite Element Model

Term A: use linear basis function

\[
\int_{\Omega^e} \frac{\partial \hat{C}}{\partial t} \phi_i dx = \int_{\Omega^e} \sum_{j=1}^{m} \frac{dC_j}{dt} \phi_j^e(x) \phi_i^e dx
\]

\[
= \sum_{j=1}^{m} \frac{dC_j(t)}{dt} \int_{\Omega^e} \phi_j(x) \phi_i(x) dx
\]

Term B: Integration by parts

\[
\int udv = uv - \int vdu
\]

\[
-\int_{\Omega^e} \frac{\partial \phi_i^e(x)}{\partial x} \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) + \left[ \phi_i^e(x) \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) \right]_{x_1}^{x_m}
\]

\[
= - \frac{\partial \phi_i^e(x)}{\partial x} \left\{ E \sum_{j=1}^{m} C_j(t) \frac{\partial \phi_j^e(x)}{\partial x} - u \sum_{j=1}^{m} C_j(t) \phi_j^e(x) \right\} dx + \left[ \phi_i^e(x) \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) \right]_{x_1}^{x_m}
\]

\[
= + \sum_{j=1}^{m} C_j(t) \left\{ \int_{\Omega^e} E \frac{\partial \phi_j^e(x)}{\partial x} \frac{\partial \phi_j^e}{\partial x} dx - \int_{\Omega^e} u \phi_j^e(x) \frac{\partial \phi_j^e}{\partial x} dx \right\} - \left[ \phi_i^e(x) \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) \right]_{x_1}^{x_m}
\]
9.3 Finite Element Model

\[ \sum_{j=1}^{M} \frac{d\hat{C}_j(t)}{dt} \int_{\Omega^e} \phi_j(x) \phi_i(x) \, dx + \sum_{j=1}^{m} C_j(t) \left\{ \int_{\Omega^e} E \frac{\partial \phi_j^e(x)}{\partial x} \frac{\partial \phi_i^e(x)}{\partial x} \, dx \right\} - \left[ \phi_i^e(x)(E \frac{\partial \hat{C}}{\partial x} - u\hat{C}) \right]_{x_1}^{x_m} = 0 \]

Let

\[ a_{ij}^e = \int_{\Omega^e} E \frac{\partial \phi_j^e(x)}{\partial x} \frac{\partial \phi_i^e(x)}{\partial x} \, dx - \int_{\Omega^e} u\phi_j^e(x) \frac{\partial \phi_i^e(x)}{\partial x} \, dx, \quad i=1,\ldots,m \]

\[ m_{ij}^e = \int_{\Omega^e} \phi_j(x) \phi_i(x) \, dx, \quad \{B\}^e = \begin{bmatrix} \phi_1^e(x)(E \frac{\partial \hat{C}}{\partial x} - u\hat{C})_{x_1} \\ \vdots \\ \phi_m^e(x)(E \frac{\partial \hat{C}}{\partial x} - u\hat{C})_{x_m} \end{bmatrix} \]
9.3 Finite Element Model

(4) Element matrix equation

The element matrix equation results in

$$[A]^e \{\dot{C}\} + [M]^e \left\{ \frac{d\dot{C}}{dt} \right\} + \{B\}^e = 0$$

Use linear basis function

$$a^e_{ij} = \int_{\Omega}^e E \frac{\partial \phi_i^e(x)}{\partial x} \frac{\partial \phi_j^e(x)}{\partial x} \left( \int_{\Omega}^e u\phi_i^e(x) \frac{\partial \phi_j^e}{\partial x} \, dx \right)$$

$$= \int_{-1}^1 E \frac{\partial \phi_j}{\partial \xi} \frac{\partial \xi}{\partial x} \frac{\partial \phi_j}{\partial \xi} \frac{\partial \xi}{\partial x} \left( \int_{\Omega}^e u\phi_j \frac{\partial \phi_j}{\partial \xi} \frac{\partial \xi}{\partial x} \, dx \right)$$

$$= \int_{-1}^1 E \frac{\partial \phi_j}{\partial \xi} \frac{\partial \phi_i}{\partial \xi} \left( \int_{\Omega}^e u\phi_j \frac{\partial \phi_i}{\partial \xi} \, dx \right)$$
9.3 Finite Element Model

By the way

\[
\frac{dx}{d\xi} = x_j \sum_{j=1}^{2} \frac{d\phi_j^e(\xi)}{d\xi} = x_1 \frac{d\phi_1^e}{d\xi} + x_2 \frac{d\phi_2^e}{d\xi} = x_1 \left(-\frac{1}{2}\right) + x_2 \left(\frac{1}{2}\right) = \frac{1}{2}(x_2 - x_1) = \frac{\Delta x}{2}
\]

\[
\therefore a_{ij}^e = \int_{-1}^{1} E \frac{\partial \phi_j}{\partial \xi} \frac{\partial \phi_i}{\partial \xi} d\xi \left(\frac{2}{\Delta x}\right) - \int_{-1}^{1} u \phi_j \frac{\partial \phi_i}{\partial \xi} d\xi
\]

\[
m_{ij}^e = \int_{\Omega^e} \phi_j(x) \phi_i(x) dx = \int_{-1}^{1} \phi_j(\xi) \phi_i(\xi) d\xi \frac{dx}{d\xi} = \int_{-1}^{1} \phi_j \phi_i d\xi \frac{\Delta x}{2}
\]

\[
B_i^e = -\left[\phi^e_i(x) \left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right) \right]_{x_1}^{x_2}
\]

\[
[ A]^e = \begin{bmatrix} 
\frac{E}{\Delta x} - \frac{u}{2} & -\frac{E}{\Delta x} + \frac{u}{2} \\
-\frac{E}{\Delta x} - \frac{u}{2} & \frac{E}{\Delta x} + \frac{u}{2} 
\end{bmatrix}
\]

\[
[M]^e = \frac{\Delta x}{6} \begin{bmatrix} 2 & 1 \\
1 & 2 
\end{bmatrix}
\]

\[
\{ B \}^e = \begin{bmatrix} 
\left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right)_{x_1} \\
-\left( E \frac{\partial \hat{C}}{\partial x} - u \hat{C} \right)_{x_2}
\end{bmatrix}
\]
9.3 Finite Element Model

(5) Assemble global matrix equations

Combining element equations

For each element, apply

\[
[M]^e \left\{ \frac{d\hat{C}}{dt} \right\} + [A]^e \{\hat{C}\} + \{B\}^e = 0
\]

\[
[M]^e \text{ & } [A]^e \sim 2 \times 2 \text{ matrices}
\]

- Numbering Systems

<table>
<thead>
<tr>
<th>Local</th>
<th>1</th>
<th>2</th>
<th>1</th>
<th>2</th>
<th>1</th>
<th>2</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Element No</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Global</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>...</td>
</tr>
</tbody>
</table>
### 9.3 Finite Element Model

Let $N$ (= number of element) = 30

number of nodes = 31

Then

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_N$</th>
<th>$x_{N+1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$e=1$</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$$
\begin{bmatrix}
    a_{11}^{(1)} & a_{12}^{(1)} \\
    a_{21}^{(1)} & a_{22}^{(1)} & a_{11}^{(2)} & a_{12}^{(2)} \\
    0 & a_{22}^{(2)} & a_{11}^{(3)} \\
    \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
    a_{22}^{(29)} & a_{11}^{(30)} & a_{12}^{(30)} & a_{22}^{(30)}
\end{bmatrix}
$$

$$
[ A ] = \sum_{e=1}^{30} [ A ]^e
$$
9.3 Finite Element Model

\[
[M] = \sum_{e=1}^{30} [M]^e \quad \sim \quad \text{same as} \quad [A]
\]

\[
[B] = \sum_{e=1}^{30} \{B\}^e = \begin{align*}
B_1^{(1)} \\
B_2^{(1)} + B_1^{(2)} \\
B_2^{(2)} + B_1^{(3)} \\
& \quad \quad \quad \vdots \\
B_2^{(29)} + B_1^{(30)} \\
B_2^{(30)}
\end{align*}
\]

Since \( q_u^e + q_L^{e+1} = 0 \)

- Boundary Conditions

\[
\begin{align*}
At \quad & x = x_1 \quad ; \quad u \hat{C} - E \frac{\partial \hat{C}}{\partial x} = u C_o \\
At \quad & x = x_N \quad ; \quad u \hat{C} - E \frac{\partial \hat{C}}{\partial x} = u C_N
\end{align*}
\]
9.3 Finite Element Model

(6) Time discretization

1) Fully Implicit

\[
\left[ M \right] \frac{\{ \hat{C} \}^{k+1} - \{ \hat{C} \}^k}{\Delta t} + [A] \{ \hat{C} \}^{k+1} + \{ B \}^{k+1} = 0 \quad (9)
\]

\[
\left( \frac{[M]}{\Delta t} + [A] \right) \{ \tilde{C} \}^{k+1} = \frac{[M]}{\Delta t} \{ \hat{C} \}^k - \{ B \}^{k+1} \quad (10)
\]

\[
[R] \{ \hat{C} \}^{k+1} = [S] \{ \hat{C} \}^k - \{ B \} \quad (11)
\]

In which

\[
[R] = \frac{[M]}{\Delta t} + [A] \quad [S] = \frac{[M]}{\Delta t}
\]
9.3 Finite Element Model

(2) Crank-Nicholson scheme

\[
[M] \frac{\{\hat{C}\}^{k+1} - \{\hat{C}\}^k}{\Delta t} + [A] \{\hat{C}\}^{k+\frac{1}{2}} + \{B\}^{k+\frac{1}{2}} = 0
\]

\[
[M] \frac{\{\hat{C}\}^{k+1} - \{\hat{C}\}^k}{\Delta t} + [A] \left( \frac{\{\hat{C}\}^{k+1} + \{\hat{C}\}^k}{2} \right) + \{B\}^{k+\frac{1}{2}} = 0
\]

\[
\therefore \left( \frac{[M]}{\Delta t} + \frac{[A]}{2} \right) \{\hat{C}\}^{k+1} = \left( \frac{[M]}{\Delta t} - \frac{[A]}{2} \right) \{\hat{C}\}^k - \left( \frac{\{B\}^{k+1} + \{B\}^k}{2} \right)
\]

\[
\therefore [P] \{\hat{C}\}^{k+1} = [Q] \{\hat{C}\}^k - \{B\}
\]
9.4 Finite Volume Model

9.4.1 FVM for one-dimensional diffusion

- Finite Difference Method:
  A domain of interest is replaced by a set of discrete points. Then, the function of $C$ is represented by an approximate function using Taylor series.

- Finite Volume Method:
  A domain is divided into the discrete control volumes. Then, the governing equation is integrated over a control volume to yield a discretized equation at a nodal point.

Consider the steady state diffusion of a concentration, $C$ in a one-dimensional domain

$$\frac{d}{dx} \left( D \frac{dC}{dx} \right) + S = 0$$
9.4 Finite Volume Model

- Grid generation
  ~ divide the domain into discrete control volumes

\[ P : \text{Nodal point} \]
\[ W \text{ and } E : \text{Neighbor nodes to west and east respectively} \]
\[ w \text{ and } e : \text{west and east side of face of control volume} \]

The faces of control volumes are positioned mid-way between adjacent nodes.

→ each node is surrounded by a control volume (cell)
9.4 Finite Volume Model

• Discretization

The integration of the governing equation over a control volume to yield a discretized equation at its nodal point \( P \)

\[
\int_{\Delta V} \frac{d}{dx} \left( D \frac{dC}{dx} \right) dV + \int_{\Delta V} S dV = \left( DA \frac{dC}{dx} \right)_{e} - \left( DA \frac{dC}{dx} \right)_{w} + \bar{S} \Delta V = 0
\]

where \( A \) : the cross-sectional area of the control volume face.

\( \Delta V \) : the volume of the control volume.

\( \bar{S} \) : the average value of source \( S \) over the control volume.
The diffusion coefficient, $D$ and the gradient, $\frac{dC}{dx}$ at east($e$) and west($w$) faces are required in order to derive the discretized equations.

→ use **Linear approximation** (Central differencing)

$$
\left( DA \frac{dC}{dx} \right)_e = D_e A_e \left( \frac{C_E - C_P}{\Delta x_{PE}} \right)
$$  \hspace{1cm} (2)

$$
\left( DA \frac{dC}{dx} \right)_w = D_w A_w \left( \frac{C_P - C_W}{\Delta x_{WP}} \right)
$$  \hspace{1cm} (3)

If the source term is approximated by means of a linear form,

$$
\overline{S} \Delta V = S_u + S_p C_p
$$  \hspace{1cm} (4)

Substituting equations (2), (3) and (4) into equation (1) gives

$$
D_e A_e \left( \frac{C_E - C_P}{\Delta x_{PE}} \right) - D_w A_w \left( \frac{C_P - C_W}{\Delta x_{WP}} \right) + (S_u + S_p C_p) = 0
$$  \hspace{1cm} (5)
Equation ⑤ can be rearranged as

\[
\left( \frac{D_e}{\Delta x_{PE}} A_e + \frac{D_w}{\Delta x_{WP}} A_w - S_p \right) C_P = \left( \frac{D_w}{\Delta x_{WP}} A_w \right) C_W + \left( \frac{D_e}{\Delta x_{PE}} A_e \right) C_E + S_u
\]

\[\downarrow\]

\[a_p C_p = a_w C_W + a_E C_E + S_u\] ⑥

where

<table>
<thead>
<tr>
<th>(a_w)</th>
<th>(a_E)</th>
<th>(a_p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\frac{D_w}{\Delta x_{WP}} A_w)</td>
<td>(\frac{D_e}{\Delta x_{PE}} A_e)</td>
<td>(a_w + a_E - S_p)</td>
</tr>
</tbody>
</table>

Discretized equation ⑥ must be set up at each of the nodal points. For control volume that are adjacent to the domain boundaries, the general discretized equation ⑥ is modified to incorporate boundary conditions.
9.4 Finite Volume Model

- Treatment of the node adjacent to boundary

Node 1 that is located adjacent to boundary can be treated in the same manner of derivation of discretized equation (6)

Integration of governing equation at Node 1,

\[ D_e A_e \left( \frac{C_E - C_p}{\Delta x} \right) - D_A A_A \left( \frac{C_p - C_A}{\Delta x / 2} \right) + \overline{S} \Delta V = 0 \]

\[ a_p C_p = a_w C_w + a_E C_E + S_u \]
9.4 Finite Volume Model

Where

<table>
<thead>
<tr>
<th>$a_W$</th>
<th>$a_E$</th>
<th>$a_p$</th>
<th>$S_p$</th>
<th>$S_u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$\frac{D_e}{\Delta x} A_e$</td>
<td>$a_W + a_E - S_p$</td>
<td>$\frac{-2D_A}{\Delta x} A_A$</td>
<td>$\overline{s} \Delta V + \frac{2D_A A_A}{\Delta x} C_A$</td>
</tr>
</tbody>
</table>
9.4 Finite Volume Model

9.4.2 FVM for two-dimensional diffusion

Two-dimensional steady state diffusion equation is given by

\[
\frac{d}{dx} \left( D \frac{dC}{dx} \right) + \frac{d}{dy} \left( D \frac{dC}{dy} \right) + S = 0
\]
9.4 Finite Volume Model

- Grid generation

~divide the domain into discrete control volumes

Upper case letter \( E, W, N, W \): Neighbor nodes

Lower case letter \( e, w, n, w \): control volume faces
9.4 Finite Volume Model

○ Discretization
~integrate governing equation over control volume (cell)

\[
\int_{\Delta V} \frac{d}{dx} \left( D \frac{dC}{dx} \right) \, dx \cdot dy + \int_{\Delta V} \frac{d}{dy} \left( D \frac{dC}{dy} \right) \, dx \cdot dy + \int_{\Delta V} S \, dV = 0
\]

Noting that \( A_e = A_w = \Delta y \) and \( A_n = A_s = \Delta x \) we obtain

\[
\left[ D_e A_e \left( \frac{\partial C}{\partial x} \right)_e - D_w A_w \left( \frac{\partial C}{\partial x} \right)_w \right] + \left[ D_n A_n \left( \frac{\partial C}{\partial y} \right)_n - D_s A_s \left( \frac{\partial C}{\partial x} \right)_s \right] + \overline{S} \Delta V = 0
\]

→ The balance of the generation of \( C \) in a control volume and the fluxes through its cell faces.
9.4 Finite Volume Model

Flux across the west face \[= D_w A_w \left( \frac{\partial C}{\partial x} \right)_w = D_w A_w \frac{C_p - C_W}{\Delta x} \]

Flux across the east face \[= D_e A_e \left( \frac{\partial C}{\partial x} \right)_e = D_e A_e \frac{C_E - C_p}{\Delta x} \]

Flux across the south face \[= D_s A_s \left( \frac{\partial C}{\partial x} \right)_s = D_s A_s \frac{C_p - C_S}{\Delta y} \]

Flux across the north face \[= D_n A_n \left( \frac{\partial C}{\partial x} \right)_n = D_n A_n \frac{C_N - C_p}{\Delta y} \]

Substituting the above expressions into equation

\[D_e A_e \frac{C_E - C_p}{\Delta x} - D_w A_w \frac{C_p - C_W}{\Delta x} + D_n A_n \frac{C_N - C_p}{\Delta y} - D_s A_s \frac{C_p - C_S}{\Delta y} + S \Delta V = 0\]
9.4 Finite Volume Model

Assuming the source term in linearized form is \( \bar{S}\Delta V = S_u + S_p C_p \) and rearranging

\[
\left( \frac{D_w A_w}{\Delta x} + \frac{D_e A_e}{\Delta x} + \frac{D_s A_s}{\Delta y} + \frac{D_n A_n}{\Delta y} - S_p \right) C_p
\]

\[
= \left( \frac{D_w A_w}{\Delta x} \right) C_w + \left( \frac{D_e A_e}{\Delta x} \right) C_E + \left( \frac{D_s A_s}{\Delta y} \right) C_S + \left( \frac{D_n A_n}{\Delta y} \right) C_N + S_u
\]

The general discretized equation form is

\[
a_p C_p = a_w C_w + a_E C_E + a_s C_S + a_N C_N + S_u
\]

Where

<table>
<thead>
<tr>
<th>( a_w )</th>
<th>( a_E )</th>
<th>( a_s )</th>
<th>( a_N )</th>
<th>( a_p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{D_w A_w}{\Delta x} )</td>
<td>( \frac{D_e A_e}{\Delta x} )</td>
<td>( \frac{D_s A_s}{\Delta y} )</td>
<td>( \frac{D_n A_n}{\Delta y} )</td>
<td>( a_w + a_E + a_s + a_N - S_p )</td>
</tr>
</tbody>
</table>

At the boundary where the concentration are known, the discretized equations are modified to incorporate boundary conditions in the manner demonstrated in one-dimensional problem.
9.4 Finite Volume Model

9.4.3 FVM for three-dimensional diffusion

Steady state diffusion in a three-dimensional situation is governed by

$$\frac{d}{dx}\left( D \frac{dC}{dx} \right) + \frac{d}{dy}\left( D \frac{dC}{dy} \right) + \frac{d}{dz}\left( D \frac{dC}{dz} \right) + S = 0$$

- Grid generation

A cell contacting node P has six neighboring nodes identified as west, east, south, north, bottom and top ($W, E, S, N, B, T$).

The notation $w, e, s, n, b$ and $t$ is used to refer to the west, east, south, north, bottom and top cell faces.
9.4 Finite Volume Model

Discretization

~integrate governing equation over control volume (cell)

\[
\begin{aligned}
\left[ D_e A_e \left( \frac{\partial C}{\partial x} \right)_e - D_w A_w \left( \frac{\partial C}{\partial x} \right)_w \right] + \\
 \left[ D_n A_n \left( \frac{\partial C}{\partial y} \right)_n - D_s A_s \left( \frac{\partial C}{\partial y} \right)_s \right] + \\
 \left[ D_t A_t \left( \frac{\partial C}{\partial z} \right)_t - D_b A_b \left( \frac{\partial C}{\partial z} \right)_b \right] + \overline{S} \Delta V = 0
\end{aligned}
\]

~ Follow the procedure developed for one- and two-dimensional cases the discretized form of equation is obtained:

\[
\begin{aligned}
\left[ D_e A_e \frac{C_E - C_p}{\Delta x} - D_w A_w \frac{C_p - C_W}{\Delta x} \right] + \\
 \left[ D_n A_n \frac{C_N - C_p}{\Delta y} - D_s A_s \frac{C_p - C_s}{\Delta y} \right] + \\
 \left[ D_t A_t \frac{C_T - C_p}{\Delta z} - D_b A_b \frac{C_p - C_B}{\Delta z} \right] + \left( S_u + S_p C_p \right) = 0
\end{aligned}
\]
The general discretized equation form is

\[ a_p C_p = a_w C_w + a_E C_E + a_s C_s + a_N C_N + a_B C_B + a_T C_T + S_u \]

Where

<table>
<thead>
<tr>
<th>( a_w )</th>
<th>( a_E )</th>
<th>( a_s )</th>
<th>( a_N )</th>
<th>( a_B )</th>
<th>( a_T )</th>
<th>( a_p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \frac{D_w A_w}{\Delta x} )</td>
<td>( \frac{D_E A_E}{\Delta x} )</td>
<td>( \frac{D_s A_s}{\Delta y} )</td>
<td>( \frac{D_n A_n}{\Delta y} )</td>
<td>( \frac{D_b A_b}{\Delta z} )</td>
<td>( \frac{D_t A_t}{\Delta z} )</td>
<td>( a_w + a_E + a_s + a_N + a_B + a_T - S_p )</td>
</tr>
</tbody>
</table>
9.4 Finite Volume Model

9.4.4 FVM for advection-dispersion

Steady advection and dispersion of concentration $C$ in a given one-dimensional flow field $U$ is governed by

$$\frac{d}{dx}(\rho UC) = \frac{d}{dx}\left(K \frac{dC}{dx}\right)$$

The flow must also satisfy continuity equation.

$$\frac{d}{dx}(\rho U) = 0$$
9.4 Finite Volume Model

- Grid generation

- divide the domain into discrete control volumes.

The same notation as in the one-dimensional diffusion problem is used.

$U_w$ and $U_e$ are cross-sectional average velocity on the faces of control volume.
9.4 Finite Volume Model

○ Discretization

Integration of transport equation over the control volume

\[(\rho UAC)_e - (\rho UAC)_w = \left( KA \frac{dC}{dx} \right)_e - \left( KA \frac{dC}{dx} \right)_w \]

Integration of continuity equation over the control volume

\[(\rho UA)_e - (\rho UA)_w = 0\]

For convenience, convective mass flux unit area and dispersion term are replaced to:

\[F = \rho U, \quad D = \frac{K}{\Delta x}\]

The cell face values of the F and D can be written as

\[F_w = (\rho U)_w, \quad F_e = (\rho U)_e\]

\[D_w = \frac{K_w}{\Delta x}, \quad D_e = \frac{K_e}{\Delta x}\]
9.4 Finite Volume Model

Assuming that \( A_w = A_e = A \), the integrated advection-dispersion equation can be written as
\[
F_e C_e - F_w C_w = D_e \left( C_E - C_p \right) - D_w \left( C_p - C_W \right)
\]

And the integrated continuity equation as
\[
F_e - F_w = 0
\]

When we have one-dimensional flow field \( U \), \( F_e \) and \( F_w \) can be calculated.

~ need to know the transported concentration \( C \) at the \( e \) and \( w \) faces

(1) Central differencing scheme for advection

For a uniform grid, the cell face values of concentration \( C \) as
\[
C_e = \left( C_p + C_E \right) / 2
\]
\[
C_w = \left( C_W + C_p \right) / 2
\]

Substitution of the above expressions into the advection term
\[
\frac{F_e}{2} \left( C_p + C_E \right) - \frac{F_w}{2} \left( C_W + C_p \right) = D_e \left( C_E - C_p \right) - D_w \left( C_p - C_W \right)
\]
This can be rearranged to give

\[
\left( D_w + \frac{F_w}{2} \right) + \left( D_e - \frac{F_e}{2} \right) + \left( F_e - F_w \right) \right] C_p = \left( D_w + \frac{F_w}{2} \right) C_W + \left( D_e - \frac{F_e}{2} \right) C_E
\]

Central differencing expressions for the discretized advection-dispersion equation are

\[
a_p C_p = a_W C_W + a_E C_E
\]

where

<table>
<thead>
<tr>
<th>(a_W)</th>
<th>(a_E)</th>
<th>(a_p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(D_w + \frac{F_w}{2})</td>
<td>(D_e - \frac{F_e}{2})</td>
<td>(a_W + a_E + (F_e - F_w))</td>
</tr>
</tbody>
</table>
9.4 Finite Volume Model

(2) Upwind differencing scheme for advection

In a strongly convective flow, the cell face should receive much stronger influencing from upstream node.

~ UDS takes into account the flow direction when determining the value at a cell face.

~ The transported value of C at a cell face is taken to be equal to the value at the upstream node.

○ Flow in the positive direction

\[
u_w > 0 \quad , \quad u_e > 0
\]

\[C_w = C_w \quad \text{and} \quad C_e = C_p\]
The discretized equation becomes

\[ F_e C_p - F_w C_W = D_e \left( C_E - C_p \right) - D_w \left( C_p - C_W \right) \]

This can be rearrange as

\[ \left[ (D_w + F_w) + D_e + (F_e - F_w) \right] C_p = (D_w + F_w) C_W + D_e C_E \]

○ Flow in the negative direction

\[ u_w < 0 \quad \text{and} \quad u_e < 0 \]

\[ C_w = C_p \quad \text{and} \quad C_e = C_E \]

The discretized equation becomes

\[ F_e C_E - F_w C_p = D_e \left( C_E - C_p \right) - D_w \left( C_p - C_W \right) \]

This can be rearrange as

\[ \left[ D_w + \left( D_e - F_e \right) + \left( F_e - F_w \right) \right] C_p = D_w C_W + \left( D_e - F_e \right) C_E \]
9.4 Finite Volume Model

Upwind differencing expressions for the discretized advection-dispersion equation are

\[ a_p C_p = a_w C_w + a_e C_e \]

Where

<table>
<thead>
<tr>
<th></th>
<th>( a_w )</th>
<th>( a_e )</th>
<th>( a_p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F_w &gt; 0, \ F_e &gt; 0 )</td>
<td>( D_w + F_w )</td>
<td>( D_e )</td>
<td>( a_w + a_e + (F_e - F_w) )</td>
</tr>
<tr>
<td>( F_w &lt; 0, \ F_e &lt; 0 )</td>
<td>( D_w )</td>
<td>( D_e - F_w )</td>
<td></td>
</tr>
</tbody>
</table>

(3) Hybrid differencing scheme for advection

~ based on a combination of central and upwind differencing schemes.

~central differencing scheme is employed for small Peclet number (\( Pe < 2 \))

~upwind differencing scheme is employed for large Peclet number (\( Pe \geq 2 \))

~Peclet number is evaluated at the face of the control volume.
9.4 Finite Volume Model

For west face of control volume

\[ Pe_w = \frac{F_w}{D_w} = \frac{(\rho U)_w}{K_w / \Delta x} \]

The hybrid differencing formula for the net flux per unit area though the west face is

\[ q_w = \begin{cases} 
  F_w \left[ \frac{1}{2} \left( 1 + \frac{2}{Pe_w} \right) C_w + \frac{1}{2} \left( 1 - \frac{2}{Pe_w} \right) C_p \right] & \text{For } -2 < Pe_w < 2 \\
  F_w C_w & \text{For } Pe_w \geq 2 \\
  F_w C_p & \text{For } Pe_w \leq -2 
\end{cases} \]

East face of control volume also follow the same manner.
The general form of the discretized equation is

\[ a_p C_p = a_w C_w + a_E C_E \]

Where

<table>
<thead>
<tr>
<th>(a_w)</th>
<th>(a_E)</th>
<th>(a_p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\max\left[ F_w, \left( D_w + \frac{F_w}{2} \right), 0 \right])</td>
<td>(\max\left[ -F_e, \left( D_e - \frac{F_e}{2} \right), 0 \right])</td>
<td>(a_w + a_E + (F_e - F_w))</td>
</tr>
</tbody>
</table>

(4) Quadratic upwind differencing scheme for advection: the QUICK scheme

- uses a three-point upstream-weighted quadratic interpolation for cell face values.
- The face value of \(C\) is obtained from a quadratic function passing through two bracketing nodes and a node on the upstream side.
9.4 Finite Volume Model

For uniform grid the value of $C$ at the cell face between two bracketing nodes $i$ and $i-1$ and upstream node $i-2$ is given by:

$$C_{\text{face}} = \frac{6}{8} C_{i-1} + \frac{3}{8} C_i - \frac{1}{8} C_{i-2}$$

When $u_w > 0$, the bracketing nodes for the west face $w$ are $W$ and $P$, the upstream node is $\text{WW}$

$$C_w = \frac{6}{8} C_W + \frac{3}{8} C_p - \frac{1}{8} C_{\text{WW}}$$

When $u_e > 0$, the bracketing nodes for the west face $e$ are $P$ and $E$, the upstream node is $W$

$$C_e = \frac{6}{8} C_p + \frac{3}{8} C_E - \frac{1}{8} C_W$$

The discretized equation becomes

$$F_e C_e - F_w C_w = D_e \left( C_E - C_p \right) - D_w \left( C_p - C_W \right)$$

$$F_e \left( \frac{6}{8} C_p + \frac{3}{8} C_E - \frac{1}{8} C_W \right) - F_w \left( \frac{6}{8} C_W + \frac{3}{8} C_p - \frac{1}{8} C_{\text{WW}} \right) = D_e \left( C_E - C_p \right) - D_w \left( C_p - C_W \right)$$
This can be rearrange as

\[
\left[ D_w - \frac{3}{8} F_w + D_e + \frac{6}{8} F_e \right] C_p = \left[ D_w + \frac{6}{8} F_w + \frac{1}{8} F_e \right] C_W + \left[ D_e - \frac{3}{8} F_e \right] C_E - \frac{1}{8} F_w C_{ww}
\]

The general form of the discretized equation is

\[ a_p C_p = a_w C_W + a_e C_E + a_{ww} C_{ww} + a_{ee} C_{ee} \]

where

\[
\begin{array}{|c|c|c|c|c|}
\hline
a_w & a_{ww} & a_e & a_{ee} & a_p \\
\hline
D_w + \frac{6}{8} \alpha_w F_w & -\frac{1}{8} \alpha_w F_w & D_e - \frac{3}{8} \alpha_e F_e & -\frac{6}{8} (1 - \alpha_e) F_e & \frac{1}{8} (1 - \alpha_e) F_e \\
\frac{1}{8} \alpha_e F_e & -\frac{6}{8} (1 - \alpha_e) F_e & \frac{1}{8} (1 - \alpha_e) F_e & a_w + a_e + a_{ww} + a_{ee} + (F_e - F_w) \\
\frac{3}{8} (1 - \alpha_w) F_w & \frac{1}{8} (1 - \alpha_w) F_w & & & \\
\hline
\end{array}
\]

Where

\[
\alpha_w = 1 \text{ for } F_w > 0 \quad \text{and} \quad \alpha_e = 1 \text{ for } F_e > 0
\]

\[
\alpha_w = 0 \text{ for } F_w < 0 \quad \text{and} \quad \alpha_e = 0 \text{ for } F_e < 0
\]
9.5 Finite Particle Model

9.5.1 Concepts

• **Finite Particle Model**

  A number of particles, each representing a finite mass of solute, are released at a rate proportional to the strength of each source. The particles are then "tracked" in space and time.

  → Particle Tracking Method (Prickett et al., 1981)

  → Giant Molecule Method

• Distribution of concentration of solute

  ~ represented by the distribution of a finite number of discrete particles

  ~ each particle which is assigned a mass which represents a fraction of the total mass of chemical constituent, is moved by flow and dispersed by the random mixing
9.5 Finite Particle Model

- In the computer code, enough particles are introduced so that their locations and density are adequate to describe the distribution of the dissolved constituent of interest.
- New position = Old position + Advection + Dispersion
- Dispersion is based on the concept of random process.
9.5 Finite Particle Model

The advection-dispersion equation is given as

\[
\frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} = E \frac{\partial^2 C}{\partial x^2} - kC \tag{1}
\]

The analytical solution for a unit slug of solute placed initially at \( x = 0 \) is given

\[
C(x, t) = \frac{1}{\sqrt{4\pi Et}} \exp \left[ -\frac{(x - ut)^2}{4Et} \right] \tag{2}
\]
9.5 Finite Particle Model

- Statistically, a random variable \( x \) is said to be normally distributed if its density function, \( n(x) \) is given by

\[
n(x) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left[-\frac{(x - \mu)^2}{2\sigma^2}\right]
\]  

(3)

\( \sigma \) = standard deviation

\( \mu \) = mean

Now, if we let

\[
\sigma = \sqrt{2Et}
\]  

(4)

\[
\mu = ut
\]  

(5)

\[
n(x) = C(x, t)
\]  

(6)
9.5 Finite Particle Model

Then, Eqs. (2) and (3) are equivalent.

So, the key to solute transport is the realization that dispersion can be considered a random process, tending to the normal distribution.

**→ Random Walk Model**
9.5 Finite Particle Model

Advective distance \( = u \Delta t \) \hspace{1cm} (7)

\( \Delta t = \text{time increment} \)

Dispersive distance \( = \pm 6\sigma \)

\[ = \sqrt{2E\Delta t} \cdot \text{ANORM}(0) \] \hspace{1cm} (8)

in which \( \pm 6\sigma = \) probable locations of particles out to 6 standard deviations either side of the mean ( > 99.9% )

\( \text{ANORM}(0) = \) a random number between -6 and +6, drawn from a normal distribution of numbers having a standard deviation of 1 and a mean of zero.

\[ \therefore \text{New position of the particle} = \text{Old position} + u \Delta t + \sqrt{2E\Delta t} \cdot \text{ANORM}(0) \] \hspace{1cm} (9)
9.5 Finite Particle Model

Repeat for numerous particles, all having the same initial position and advection term.

→ Create a map of the new positions of the particles having the discrete density function.

\[
C(x,t) \rightarrow n(x) \rightarrow \frac{N}{\Delta x} = \frac{N_o}{\sqrt{2\pi} \sqrt{2\Delta x E \Delta t}} \exp \left[ -\frac{(x - u \Delta t)^2}{4\Delta x E \Delta t} \right]
\]

(10)

in which \( \Delta x \) = incremental distance over which \( N \) particles are found

\( N_o \) = total number of particles in the experiment

The distribution of particles around the mean position, \( u \Delta t \), is made to be normally distributed via the function \( \text{ANORM}(0) \).
9.5 Finite Particle Model

- Generation of ANORM(0) in computer code.

1) Summation of random function

\[
ANORM(0) = \sum_{i=1}^{12} RF(0,1) - 6
\]

In EXCEL, \text{RAND}() use @RAND function to generate a uniform random number between 0 and 1 = \text{U}(0,1)

2) Multiply random function

\[
ANORM(0) = RF(0,1) \times 12 - 6
\]

3) Numerical Recipes

ANORM(0)\text{= GASDEV (IDUM)}

RAND() = RAN1(IDUM)
9.5 Finite Particle Model

- Advantages of the Random Walk Model
  i) There is no numerical dispersion, despite the use of an Eulerian framework.
  ii) Computer CPU time is drastically reduced.
  iii) Solutions are additive. If not enough particles are included for adequate definition in one run, subsequent runs may be made and the results of these may be superimposed upon the first run.
  iv) This method is particularly suited to time-sharing systems where velocity fields can be stored.

- Disadvantages
  i) It may require a large number of particles to obtain meaningful results.
  ii) It doesn't easily accommodate non-linear kinetic expressions.
9.5 Finite Particle Model

9.5.2 2D Random Walk Model

Depth-averaged advection-dispersion equation is

\[
\frac{\partial C}{\partial t} + u(x, y) \frac{\partial C}{\partial x} + v(x, y) \frac{\partial C}{\partial y} = \frac{\partial}{\partial x} \left( E_x \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left( E_y \frac{\partial C}{\partial y} \right) + S
\]

\[
E_x = 5.93 du_*
\]

\[
E_y = 0.6 du_*
\]

\[
u_* = \sqrt{gds}
\]
9.5 Finite Particle Model

2D Random Walk Model

Advection: \( \sqrt{(u \Delta t)^2 + (v \Delta t)^2} \)

New position \((x, y)\)

Transverse Dispersion: \( \sqrt{2 \, E \, \Delta t \cdot ANORM(0)} \)

Longitudinal Dispersion: \( \sqrt{2 \, E \cdot \Delta t \cdot ANORM(0)} \)
9.5 Finite Particle Model

Longitudinal and transverse dispersion take place simultaneously

\[ x = x_0 + u \Delta t + \sqrt{2E_x \Delta t} \cdot ANORM(0) \]

\[ y = y_0 + v \Delta t + \sqrt{2E_y \Delta t} \cdot ANORM(0) \]

In natural rivers

\[ E_x = 5.93 \, du_* \]

\[ E_y = 0.6 \, du_* \]