HWR 516

HYDROLOGIC TRANSPORT PROCESSES

(Yeh, 1992 Fall)

Part 5

Numerical Methods for Advection-Diffusion Equations

File: 516-5.wp

Numerical Solutions of Transport Equations

$$D\frac{\partial^2 C}{\partial x^2} - u\frac{\partial C}{\partial x} = \frac{\partial C}{\partial t}$$

1. Finite Difference Approach
2. Finite Element Approach
3. Methods of characteristics (Eulerian-Langrangian)

§ Finite Difference Approximation

Spatial Discretization

1. Forward Difference

$$c(x+\Delta x) = c(x) + \Delta x(\left.\frac{dc}{dx}\right|_x) + \frac{(\Delta x)^2}{2!}\left.\frac{d^2c}{dx^2}\right|_x + \frac{(\Delta x)^2}{3!}\left.\frac{d^3c}{dx^3}\right|_x .... \qquad (1)$$

Via the Taylor's series expansion

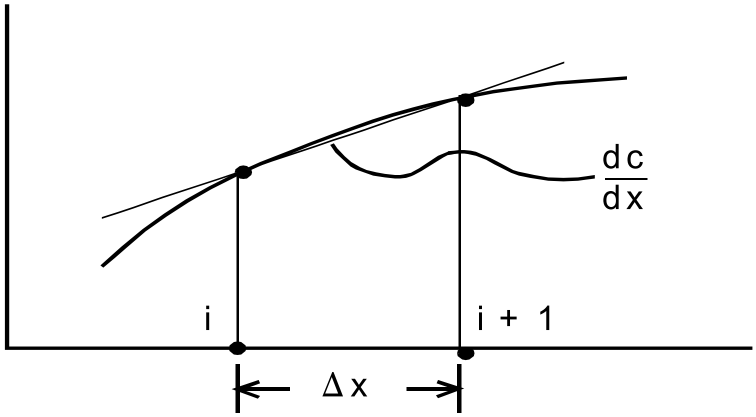
$$\frac{dc}{dx} = \frac{c(x+\Delta x) - c(x)}{\Delta x} - \frac{(\Delta x)}{2!}\frac{d^2c}{dx^2} - \frac{(\Delta x)^2}{3!}\frac{d^3c}{dx^3} + ...$$

First-order approximation

$$\frac{dc}{dx} = \frac{c(x+\Delta x) - c(x)}{\Delta x} + O(\Delta x)$$

The approximation is first order in Δx

$$\frac{dc}{dx} \approx \frac{c_{i+1} - c_i}{\Delta x}$$



2. Backward Finite Difference

$$c(x-\Delta x) = c(x) - \Delta x \frac{dc}{dx} + \frac{(\Delta x)^2}{2!}\frac{d^2c}{dx^2} - \frac{(\Delta x)^3}{3!}\frac{d^3c}{dx^3} + \ldots \quad (2)$$

$$\frac{dc}{dx} = \frac{c_i - c_{i-1}}{\Delta x} + \frac{(\Delta x)}{2!}\frac{d^2c}{dx^2} - \frac{(\Delta x)^2}{3!}\frac{d^3c}{dx^3} + \ldots$$

$$\frac{dc}{dx} \approx \frac{c_i - c_{i-1}}{\Delta x} \qquad 0(\Delta x) \qquad \textit{Truncation} \qquad \textit{error}$$

3. Central Finite Difference

Subtracting (2) from (1)

$$c_{i+1} - c_{i-1} = 2\Delta x \frac{dc}{dx} + \frac{(\Delta x)^3}{3!}\frac{d^3c}{dx^3} + \ldots$$

$$\left.\frac{dc}{dx}\right|_i \approx \frac{c_{i+1} - c_{i-1}}{2\Delta x} \qquad 0\left[(\Delta x)^2\right]$$

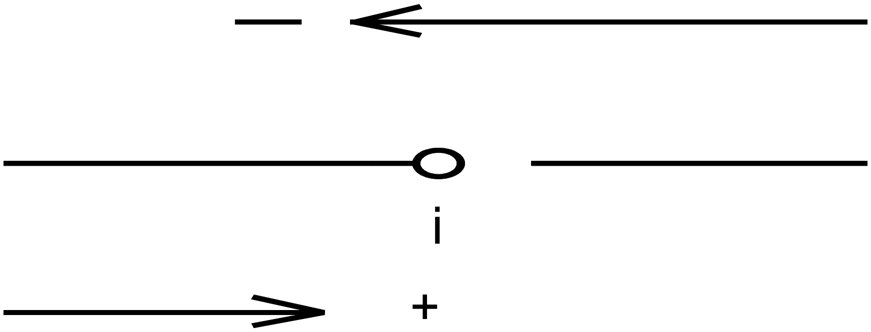
4. Upwind Difference

$$\frac{\partial c}{\partial x} \approx (1-\alpha)\frac{(c_{i+1}-c_{i-1})}{2\Delta x} + \alpha\frac{(c_{i+1}-c_i)}{\Delta x}$$

$\alpha = \pm 1$ is a weighting factor. Its sign depends on the direction of the velocity

$$\alpha = +1 \qquad \frac{\partial c}{\partial x} \approx \frac{c_i - c_{i-1}}{\Delta x}$$

$$\alpha = -1 \qquad \frac{\partial c}{\partial x} \approx \frac{c_{i+1} - c_i}{\Delta x}$$



Finite Difference Analog for the Second Spatial Derivative

Eq. (1) + (2)

$$c_{i+1} + c_{i-1} = 2c_i + \frac{d^2c}{dx^2} + 2\frac{d^4c}{dx^4}\frac{(\Delta x)^4}{4!} + \ldots$$

$$\left.\frac{d^2c}{dx^2}\right|_i \approx \frac{c_{i+1} - 2c_i + c_{i-1}}{(\Delta x)^2} \qquad 0((\Delta x)^2)$$

Temporal Discretization

$$D\frac{\partial^2 c}{\partial x^2} - u\frac{\partial c}{\partial x} = \frac{\partial c}{\partial t}$$

1. Explicit Scheme

$$\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)^t=\left(\frac{\partial c}{\partial t}\right)^t$$

$$\left(D\frac{\partial^2 c}{\partial x^2}\right)^t-\left(u\frac{\partial c}{\partial x}\right)^t=\frac{c^{t+1}-c^t}{\Delta t}$$

$$c^{t+1}=\Delta t\left[D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right]^t+c^t$$

Conditionally stable: instability and oscillation if conditions are not met.

2. Implicit Scheme

$$\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t+1}=\left(\frac{\partial c}{\partial t}\right)_i^{t+1}$$

$$\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t+1}=\frac{c_i^{t+1}-c_i^t}{\Delta t}$$

$$\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t+1}-\frac{c_i^{t+1}}{\Delta t}=\frac{-c_i^t}{\Delta t}$$

Unconditionally stable, First-order accurate in time.

3. Weighted Average Scheme

$$\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t+\theta}=\left(\frac{\partial c}{\partial t}\right)_i^{t+\theta}$$

$$\theta\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t+1}+(1-\theta)\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t}=\frac{c_i^{t+1}-c_i^t}{\Delta t}$$

$$\theta\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t+1}+\frac{c_i^{t+1}}{\Delta t}=-(1-\theta)\left(D\frac{\partial^2 c}{\partial x^2}-u\frac{\partial c}{\partial x}\right)_i^{t}-\frac{c_i^t}{\Delta t}$$

where $0\le\theta\le 1$

if $\theta=0$, the scheme is explicit;

$\theta=\frac{1}{2}$ $\quad$ *Crank – Nicolson*

$\theta=1$ $\quad$ *implicit*

Unconditionally stable, Higher-order of accuracy

Finite Difference Analog for the Advection-Diffusion Equation

Now, we use the following approximations for spatial derivatives:

$$\frac{\partial^2 c^t}{\partial x^2{}_i} = \frac{c_{i+1}^t - 2c_i^{\,t} + c_{i-1}^t}{\Delta x^2}$$

$$\frac{\partial^2 c^{t+1}}{\partial x^2{}_i} = \frac{c_{i+1}^{t+1} - 2c_i^{\,t+1} + c_{i-1}^{t+1}}{\Delta x^2}$$

$$\frac{\partial c^t}{\partial x_i} = \frac{(1-\alpha)(c_{i+1}^t - c_{i-1}^t)}{2\Delta x} + \alpha\frac{(c_i^t - c_{i-1}^t)}{\Delta x}$$

$$\frac{\partial c^{t+1}}{\partial x_i} = \frac{(1-\alpha)(c_{i+1}^{t+1} - c_{i-1}^{t+1})}{2\Delta x} + \alpha\frac{(c_i^{t+1} - c_{i-1}^{t+1})}{\Delta x}$$

and then use the weighted scheme for the temporal derivative:

$$\theta\left(D\frac{\partial^2 c}{\partial x^2} - u\frac{\partial c}{\partial x}\right)_i^{t+1} + \frac{c_i^{t+1}}{\Delta t} = -(1-\theta)\left(D\frac{\partial^2 c}{\partial x^2} - u\frac{\partial c}{\partial x}\right)_i^t - \frac{c_i^t}{\Delta t}$$

Therefore, the advection diffusion equation can be approximated by the following finite difference analog

$$\left[\frac{D\theta}{\Delta x^2} + \frac{u\theta(1-\alpha)}{2\Delta x} - \frac{u\theta\alpha}{\Delta x}\right]c_{i-1}^{t+1} + \left[\frac{-2D\theta}{\Delta x^2} - \frac{u\theta a}{\Delta x} - \frac{1}{\Delta t}\right]c_i^{t+1} + \left[\frac{D\theta}{\Delta x^2} - \frac{u\theta(1-\alpha)}{2\Delta x}\right]c_{i+1}^{t+1} =$$

$$\left[\frac{-(1-\theta)D}{(\Delta x)^2} - \frac{(1-\theta)u(1-\alpha)}{2\Delta x} - \frac{(1-\theta)u\alpha}{\Delta x}\right]c_{i-1}^t + \left[\frac{-1}{\Delta t} + \frac{2(1-\theta)D}{\Delta x^2} - \frac{(1-\theta)u\alpha}{\Delta x}\right]c_i^t +$$

$$\left[\frac{-(1-\theta)D_L}{\Delta x^2} + \frac{(1-\theta)u(1-\alpha)}{2\Delta x}\right]c_{i+1}^t$$

The above finite difference analog can be expressed in the form:

$$Ac_{i-1}^{t+1} + Bc_i^{t+1} + Dc_{i+1}^{t+1} = H_i^t$$

where

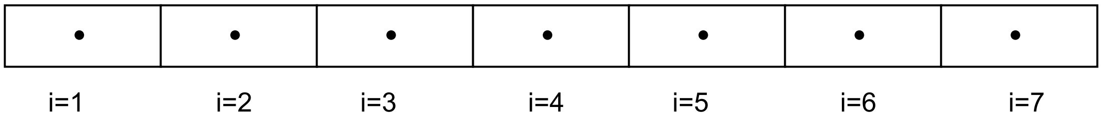
$$A = \left[\frac{D\theta}{\Delta x^2} + \frac{u\theta(1-\alpha)}{2\Delta x} - \frac{u\theta\alpha}{\Delta x}\right],$$

$$B = \left[\frac{-2D\theta}{\Delta x^2} - \frac{u\theta a}{\Delta x} - \frac{1}{\Delta t}\right],$$

$$C = \left[\frac{D\theta}{\Delta x^2} - \frac{u\theta(1-\alpha)}{2\Delta x} \right], \text{ and}$$

$$H_i = \left[\frac{-(1-\theta)D}{(\Delta x)^2} - \frac{(1-\theta)u(1-\alpha)}{2\Delta x} - \frac{(1-\theta)u\alpha}{\Delta x} \right] c_{i-1}^t + \left[\frac{-1}{\Delta t} + \frac{2(1-\theta)D}{\Delta x^2} - \frac{(1-\theta)u\alpha}{\Delta x} \right] c_i^t +$$
$$\left[\frac{-(1-\theta)D_L}{\Delta x^2} + \frac{(1-\theta)u(1-\alpha)}{2\Delta x} \right] c_{i+1}^t$$

Now, we apply the finite different analog to a one-dimensional domain, which has seven blocks as shown below. Each block has a length of Δx and an index, i.



If we expand the equation for different spatial index, i=2, 3, …7, we have

$$\begin{array}{lllllll}
Ac_1^{t+1} & +Bc_2^{t+1} & +Dc_3^{t+1} & & & & = H_2^t \\
 & Ac_2^{t+1} & +Bc_3^{t+1} & +Dc_4^{t+1} & & & = H_2^t \\
 & & Ac_3^{t+1} & +Bc_4^{t+1} & +Dc_5^{t+1} & & = H_2^t \\
 & & & Ac_4^{t+1} & +Bc_5^{t+1} & +Dc_6^{t+1} & = H_2^t \\
 & & & & Ac_5^{t+1} & +Bc_6^{t+1} & +Dc_7^{t+1} = H_2^t
\end{array}$$

Initial condition, $c(x,0) = 0$ or $c_i^0 = 0,$ where i=1, 2, 3, 4, 5, 6, and 7, is implemented by substituting it to H_i^0. And the following boundary conditions

$$q_1(t) = uc - D\frac{\partial c}{\partial x}$$
$$c(L,t) = c_7^t = 0$$

Prescribed boundary conditions are incorporated as follow:

The flux boundary condition can be discretized as

$$q_1 = u_1 c_1 - D\frac{c_2 - c_1}{\Delta x}$$

or

$$c_1 = -\frac{q_1 \Delta x}{(u_1 \Delta x + D)} - D\frac{c_2}{(u_1 \Delta x + D)}$$

After implementation of the two boundary conditions, we have

$$\begin{array}{llllllll}
A\left(\dfrac{-q_1\Delta x}{\left(u_1\Delta x+D\right)}\right) & +\left(B-D\dfrac{c_2}{\left(u_1\Delta x+D\right)}\right)c_2^{t+1} & +Dc_3^{t+1} & & & & = & H_2^t \\
 & Ac_2^{t+1} & +Bc_3^{t+1} & +Dc_4^{t+1} & & & = & H_2^t \\
 & & Ac_3^{t+1} & +Bc_4^{t+1} & +Dc_5^{t+1} & & = & H_2^t \\
 & & & Ac_4^{t+1} & +Bc_5^{t+1} & +Dc_6^{t+1} & = & H_2^t \\
 & & & & Ac_5^{t+1} & +Bc_6^{t+1} & +Dc_7^t & = H_2^t
\end{array}$$

The first term of the first equation is known because the flux $q_1, u_1, \Delta x$ and D are specified a priori. Similarly, the last term of the last equation is known because of the prescribed concentration boundary condition. These terms can be moved to the right-hand side of the equations. Now, we have a system of equations as follows:

$$\begin{array}{lllllllcl}
0 & \left(B-D\dfrac{c_2}{\left(u_1\Delta x+D\right)}\right)c_2^{t+1} & +Dc_3^{t+1} & & & & & = & H_2^t - A\left(\dfrac{-q_1\Delta x}{\left(u_1\Delta x+D\right)}\right) \\
 & Ac_2^{t+1} & +Bc_3^{t+1} & +Dc_4^{t+1} & & & & = & H_2^t \\
 & & Ac_3^{t+1} & +Bc_4^{t+1} & +Dc_5^{t+1} & & & = & H_2^t \\
 & & & Ac_4^{t+1} & +Bc_5^{t+1} & +Dc_6^{t+1} & & = & H_2^t \\
 & & & & Ac_5^{t+1} & +Bc_6^{t+1} & 0 & = & H_2^t - Dc_7^t
\end{array}$$

Given values of c at all blocks at initial time or previous time and the boundary conditions, the right-hand side of the equations can be determined. The coefficients on the left-hand side of the equations are also known. We thus can solve the system of equations to obtain c at all blocks at time=t+1. One possible way to solve this system of equation is the Gaussian elimination. However, the system of equations has a tri-diagonal characteristic. An efficient method, Thomas Algorithm, can be used.

Thomas Algorithm

Example:

$$\begin{array}{llllllll}
b_1c_1 & +d_1c_2 & & & & = & h_1 & (1) \\
a_2c_1 & +b_2c_2 & +d_2c_3 & & & = & h_2 & (2) \\
 & a_3c_2 & +b_3c_3 & +d_3c_4 & & = & h_3 & (3) \\
 & & a_4c_3 & +b_4c_4 & +d_4c_5 & = & h_4 & (4) \\
 & & & a_4c_4 & +b_5c_5 & = & h_5 & (5)
\end{array}$$

Procedure:

Step (1) we divide (1) by b_1 and we have

$$c_1 + \frac{d_1}{b_1} c_2 = \frac{h_1}{b_1}$$

which can be expressed as

$$c_1 + \beta_1 c_2 = y_1 \qquad (6)$$

Step (2), Then multiplying (6) with $-a_2$ yields

$$-a_2 c_1 - a_2 \beta_1 c_2 = -a_2 y_1 \qquad (7)$$

Adding (7) to (2) results in

$$(b_2 - a_2 \beta_2) c_2 + d_2 c_3 = h_2 - a_2 y_1$$

which can be written as

$$\alpha_2 c_2 + d_2 c_3 = h_2 - a_2 y_1 \qquad (8)$$

Dividing (8) by α_2 yields

$$c_2 + \frac{d_2}{a_2} c_3 = \frac{h_2 - a_2 y_1}{a_2} \quad \Rightarrow \quad c_2 + \beta_2 c_3 = y_2 \qquad (9)$$

Step (3). Next, we multiply (10) with $-a_3$ to yield

$$-a_3 c_2 - a_3 \beta_2 c_3 = -a_3 y_2$$

Adding the result to (3), we have

$$(b_3 - a_3 \beta_2) c_3 + d_3 c_4 = h_3 - a_3 y_2$$

which can also be expressed as

$$c_3 + \left(\frac{d_3}{\alpha_3} \right) c_4 = \left(\frac{h_3 - \alpha_3 y_2}{\alpha_3} \right) \quad \Rightarrow \quad c_3 + \beta_3 c_4 = y_3$$

The above procedure can be repeated for the rest equations. You should see that the procedure has a particular pattern, which is the general rule:

$$\alpha_i = b_i - \alpha_i \beta_{i-1}$$
$$\beta_i = d_i / \alpha_i$$
$$y_i = (h_i - \alpha_i y_{i-1}) / \alpha_i$$

where i=1, n, the total number of equations. This is called forward substitution. After the forward substitution, we have

$$\begin{array}{cccccccc} c_1 & +\beta_1 c_2 & & & & & = & y_1 \\ & c_2 & +\beta_2 c_3 & & & & = & y_2 \\ & & c_3 & +\beta_3 c_4 & & & = & y_3 \\ & & & c_4 & +\beta_4 c_5 & & = & y_4 \\ & & & & c_5 & & = & y_5 \end{array}$$

Now, c_5 is known explicitly from the last equation, which can be substitute back to the next equation to solve for c_4. Thus, one can solve for cs' by this backward substitution, which can be expressed in a general rule.

$$c_i = y_i - \beta_1 c_{i+1}$$

Because the regular pattern of forward and backward substitution, a computer program can be easily implemented to solve the tri-diagonal matrix resulting from the finite difference discretization of the advection-diffusion equation.

Numerical Dispersion

While numerical formation of the advection-diffusion equation is straight forward, obtaining a stable and accurate solution, however, confronts many difficulties. We will example some of the problems, next.

1-D Convection-Dispersion Equation

$$\frac{\partial c}{\partial t} + u \frac{\partial c}{\partial x} = D \frac{\partial^2 c}{\partial x^2}$$

Supposed we use the upwind finite difference approach for discretizing the spatial derivatives of the governing equation.

$$\frac{c_i^{t+1} - c_i^t}{\Delta t} + u \frac{c_i^t - c_{i-1}^t}{\Delta x} = D \frac{c_{i-1}^t - 2c_i^t + c_{i+1}^t}{(\Delta x)^2}$$

Stability criterion for the solution of the finite difference equation is

$$\Delta t \leq \left[\frac{u}{\Delta x} + \frac{2D}{\Delta x^2} \right]^{-1}$$

see (P.J.Roache, 1976,Comput.Fluid Dynamics, Hermosa Publisher).
Expand c_i^{t+1} in the above finite difference equation in a Taylor series gives

$$c_i^{t+1} = c_i^t + \Delta t(\frac{\partial c}{\partial t}) + \frac{\Delta t^2}{2}\frac{\partial^2 c}{\partial t^2} + 0(\Delta t^3)$$

Since

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - u\frac{\partial c}{\partial x}$$

we can expand the third term on the right-hand side of the equation,

$$\therefore \frac{\partial}{\partial t}\left(\frac{\partial c}{\partial t}\right) = \frac{\partial^2 c}{\partial t^2} = u^2\frac{\partial^2 c}{\partial x^2} - 2D\left[u\frac{\partial}{\partial x}\left(\frac{\partial^2 c}{\partial x^2}\right)\right] + D^2\left[\frac{\partial^2}{\partial x^2}\left(\frac{\partial^2 c}{\partial x^2}\right)\right] \qquad （4）$$

$$= D\left[D\frac{\partial^2}{\partial x^2}\left(\frac{\partial^2 c}{\partial x^2}\right) - u\frac{\partial}{\partial x}\left(\frac{\partial^2 c}{\partial x^2}\right)\right] +$$

$$-u\left[D\frac{\partial^2}{\partial x^2}\left(\frac{\partial c}{\partial x}\right) - u\frac{\partial}{\partial x}\left(\frac{\partial c}{\partial x}\right)\right]$$

$$= D^2\frac{\partial^2}{\partial x^2}\left(\frac{\partial^2 c}{\partial x^2}\right) - 2D\left[u\frac{\partial}{\partial x}\left(\frac{\partial^2 c}{\partial x^2}\right)\right] + u^2\frac{\partial^2 c}{\partial x^2}$$

Substituting (4) in (3), we have

$$c_i^{t+1} = c_i^t + \Delta t\left(\frac{\partial c}{\partial t}\right) + \frac{\Delta t^2}{2}\left[u^2\frac{\partial^2 c}{\partial x^2}\right] + HOT + HOD$$

where HOT and HOD stand for higher order terms and derivatives.

Likewise the spatial terms in the finite difference analogs can be expressed in a Taylor series:

$$c_{i-1}^t = c_i^t - \Delta x\frac{\partial c}{\partial x} + \Delta x^2\frac{\partial^2 c}{\partial x^2} + ... \qquad \text{-------(6)}$$

$$c_{i+1}^t = c_i^t + \Delta x\frac{\partial c}{\partial x} + \Delta x^2\frac{\partial^2 c}{\partial x^2} + ... \qquad \text{-------(7)}$$

Substituting (5), (6), (7) into (2), we get:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} = D\frac{\partial^2 c}{\partial x^2} + \frac{1}{2}\left[(u\Delta x - u^2\Delta t)\frac{\partial^2 c}{\partial x^2}\right] + ... \qquad \text{-------(8)}$$

This equation is the continuous partial differential equation of our finite difference analog of the advection-diffusion equation. According this equation, it is clear that our finite difference equation does not truly equal to the advection-diffusion equation that we intend to solve but an advection-diffusion equation with extra terms (i.e., an approximate advection-diffusion equation). Since the second term of the right-hand side of the equation is a product of the second derivative of concentration, the equation can also be rewritten as

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} = \left[D + \frac{1}{2}(u\Delta x - u^2\Delta t)\right]\frac{\partial^2 c}{\partial x^2} + ...$$

Therefore, the coefficient of the second partial derivative of concentration, which represents the effect of diffusion, contains not only the physical diffusion coefficient, D, but also a numerical error:

$$D_{numerical} = \frac{1}{2}(u\Delta x - u^2\Delta t)$$

that can act like diffusion. This is the so-called numerical diffusion. The numerical dispersion term is a result of spatial and time (temporal) approximation of advection transport (u term).

Now, let's examine the possibility to avoid the numerical diffusion. Recall the stability criterion discussed previously:

$$\Delta t \le \left[\frac{u}{\Delta x} + \frac{2D}{\Delta x^2}\right]^{-1}$$

which leads to the fact that

$$(u\Delta x - u^2\Delta t) \ge 0$$

We then can control the numerical dispersion if we let

$$u\Delta x = u^2\Delta t$$

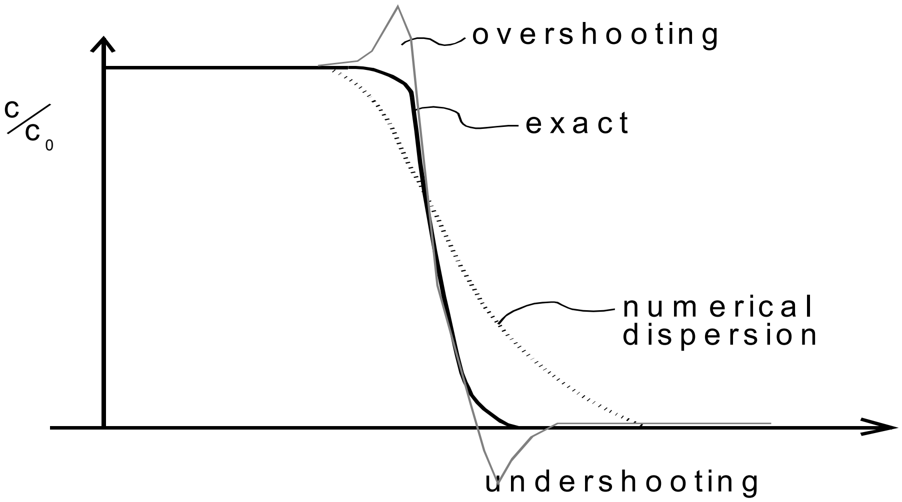
That is, we restrict the ratio (Courant number) equal to unity:

$$\frac{u\Delta t}{\Delta x} = 1$$

Hence, there should be no numerical dispersion. While we can control the numerical diffusion, we encounter another problem:

Problems:

1. While we minimize the numerical dispersion, overshoot is encountered;
2. When overshoot is controlled, numerical dispersion occurs



While these problems are active research topics, the general rule of thumb is that the time step and grid size of the discretization should satisfy the following conditions:

$$\text{Mesh Peclet Number} = \frac{v\Delta t}{D} < 2$$

$$\text{Courant Number} = \frac{v\Delta t}{\Delta x} < 1$$

Please see the following references for details.

1. Pinder & Gray (1977) Finite Element simulation in surface and subsurface hydrology. (AP)
2. Fletcher, C. Computational technique for fluid dynamics, vol. 1, Springer-Verlag

Method of Characteristics

One of the problems associated with the finite difference approach for the advection-diffusion equation is the overshooting and under shooting when the diffusion coefficient is small or equal to zero (i.e., a violation of the mesh peclet numer criterion). To alleviate this problem, one can use the method of characteristics approach, which will be discussed as follows.

Consider a 2-D solute transport equation:

$$\frac{\partial c}{\partial t} = D_L \frac{\partial^2 c}{\partial x^2} + D_T \frac{\partial^2 c}{\partial y^2} - u_x \frac{\partial c}{\partial y} - u_y \frac{\partial c}{\partial y} \qquad \text{-------(1)}$$

Again, we will use the total (or substantial, material) derivative of *c(x, y, t),*

$$\frac{dc}{dt} = \frac{\partial c}{\partial t} + \frac{\partial c}{\partial x}\frac{dx}{dt} + \frac{\partial c}{\partial y}\frac{dy}{dt} \qquad \text{-------(2)}$$

where

$$\frac{dx}{dt} = u_x \qquad \frac{dy}{dt} = u_y \qquad \text{-------(3)}$$

Substitute (2) into (1), we have

$$\frac{dc}{dt} = D_L \frac{\partial^2 c}{\partial x^2} + D_T \frac{\partial^2 c}{\partial y^2}$$

Again, this change in concentration is the one perceived by an observer moving at the same velocity as the fluid (or at a moving frame work, or Lagrangian coordinates). Equations (3) and (4) are the characteristics of Equation (1). According to our intuition, we can solve the advection-diffusion equation in two steps.

1) Solving for the convective terms (3)
2) Solving for the Diffusive-Dispersion terms (4)

In other words, we are not solving the C-D eq. directly, and we are solving the equivalent equations of the C-D eq. The equivalent equations are the characteristics of C-D, i.e.,

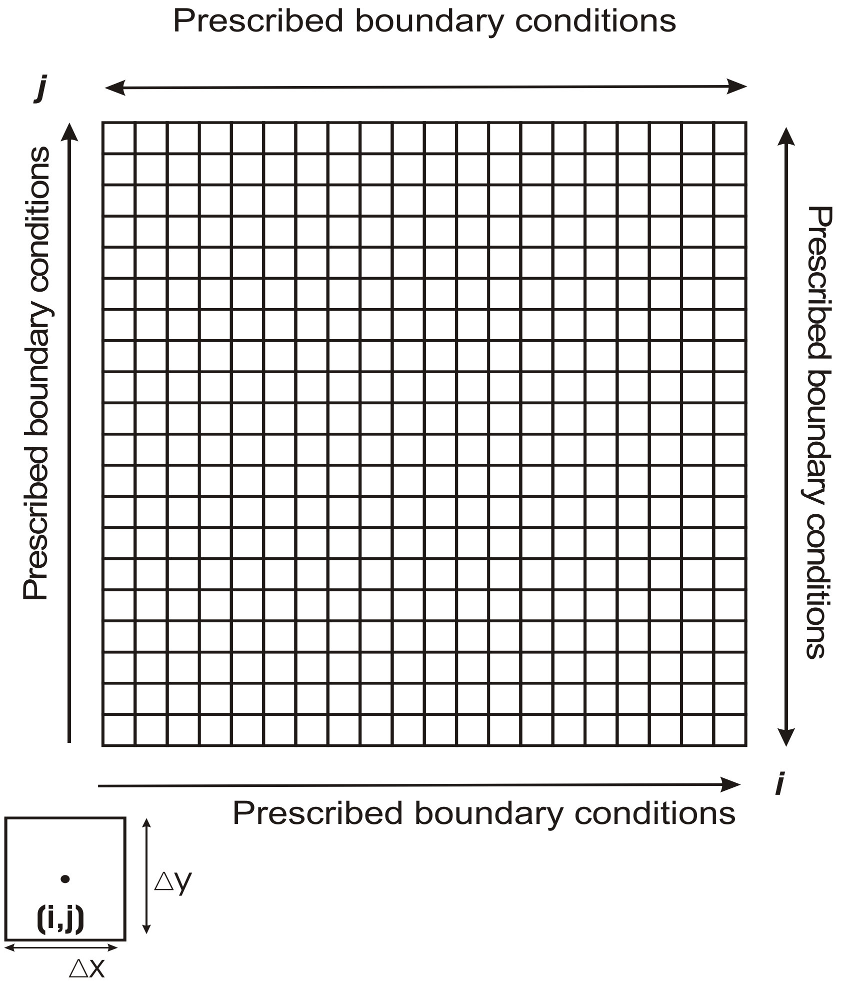
$$\frac{dx}{dt} = u_x$$

$$\frac{dy}{dt} = u_y$$

$$\frac{dc}{dt} = D_L \frac{\partial^2 c}{\partial x^2} + D_T \frac{\partial^2 c}{\partial y^2}$$

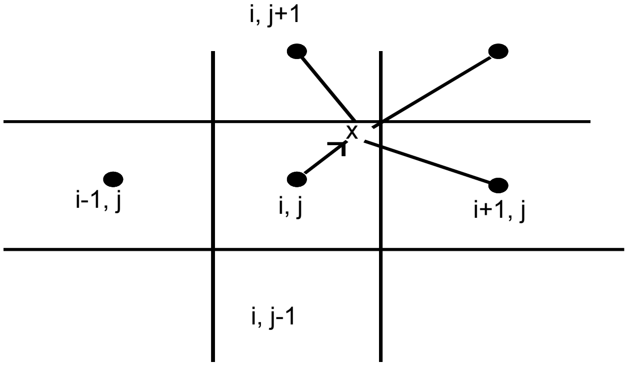
Several steps are used.

Step 1. Discretization of the solution domain using a block-centered grid system (i.e., all the variables and parameters are specified at the center of a finite difference block of a dimension of Δx and Δy . The finite difference block is identified by the indices, i, and j. Next, assign boundary and initial conditions to boundary and all blocks, respectively.



Step. 2. Particle Tracking. Introduce a set of moving points distributed uniformly in the solution domain with initial coordinates, $x_{p,o}$ $y_{p,o}$. and initial concentration $c_{p,o}$ where p is an index identifying the moving points, and o denotes the initial concentration.

Step. 3. Assume that from the solution of a two-dimensional flow equation, we know the heads at all stationary points within the grid system and we have calculated the velocities, $u_x(i,j)$ and $u_y(i,j)$.



Use the velocities, which apply to the center of blocks, u_x and u_y, to determine the velocities for the moving points at any location (*x, y*) within the block by a linear interpolation scheme that uses the velocity at adjacent blocks.

Step. 4. Determine the new positions of the moving points according to the moving point velocities. That is,

$$\left(\frac{dx}{dt}\right)_p^{n+1} = v_x(x_p^n, y_p^n), \qquad \left(\frac{dy}{dt}\right)_p^{n+1} = v_y(x_p^n, y_p^n)$$

$$\frac{x_p^{n+1} - x_p^n}{\Delta t} = v_x \qquad \Delta t = t^{n+1} - t^n$$

where *n* + *1* = a new time level, and *n* = the current time level. The new coordinates of a moving point is evaluated by

$$x_p^{n+1} = x_p^n + \Delta t \cdot v_x(x_p^n, y_p^n) \text{ and } y_p^{n+1} = y_p^n + \Delta t \cdot v_y(x_p^n, y_p^n)$$

This is an Euler formula, which has a first order accuracy in time, $O(\Delta t)$. Higher-order approximation (such as Runge-Kutta type formula) can also be used.

Step 5. After moving the points, coordinates of the new position of the moving point $\left(x_p^{n+1}, y_p^{n+1}\right)$ are tested to see which block the moving point lies in. Now a temporary concentration in each block c_{ij}^{*n} is determined by taking the average of the concentration of all moving points, c_p^n, within the block.

Step. 6. We next calculate the change in concentration due to diffusion by solving finite difference analog form of eq. (4). That is,

$$\frac{dc}{dt} = \frac{\Delta c_{ij}^n}{\Delta t} = \left[D_L \frac{\partial^2 c^*}{\partial x^2} + D_T \frac{\partial^2 c^*}{\partial y^2} \right]$$

where

$$\frac{\partial^2 c*}{\partial x^2} = (c^{*n}_{i-1,j} - 2c^{*n}_{ij} + c^{*n}_{i+1,j}) / \Delta x^2$$

$$\frac{\partial^2 c*}{\partial y^2} = (c^{*n}_{i,j-1} - 2c^{*n}_{ij} + c^{*n}_{i,j+1}) / \Delta y^2$$

Step. 7. Update the concentration of moving points by

$$c_p^{n+1} = c_p^n + \Delta c_{ij}^n$$

It is assumes that the moving points lying within a block undergo the same change in concentration due to diffusion.

Step. 8. The concentrations at the stationary grid points are corrected for the new time step by:

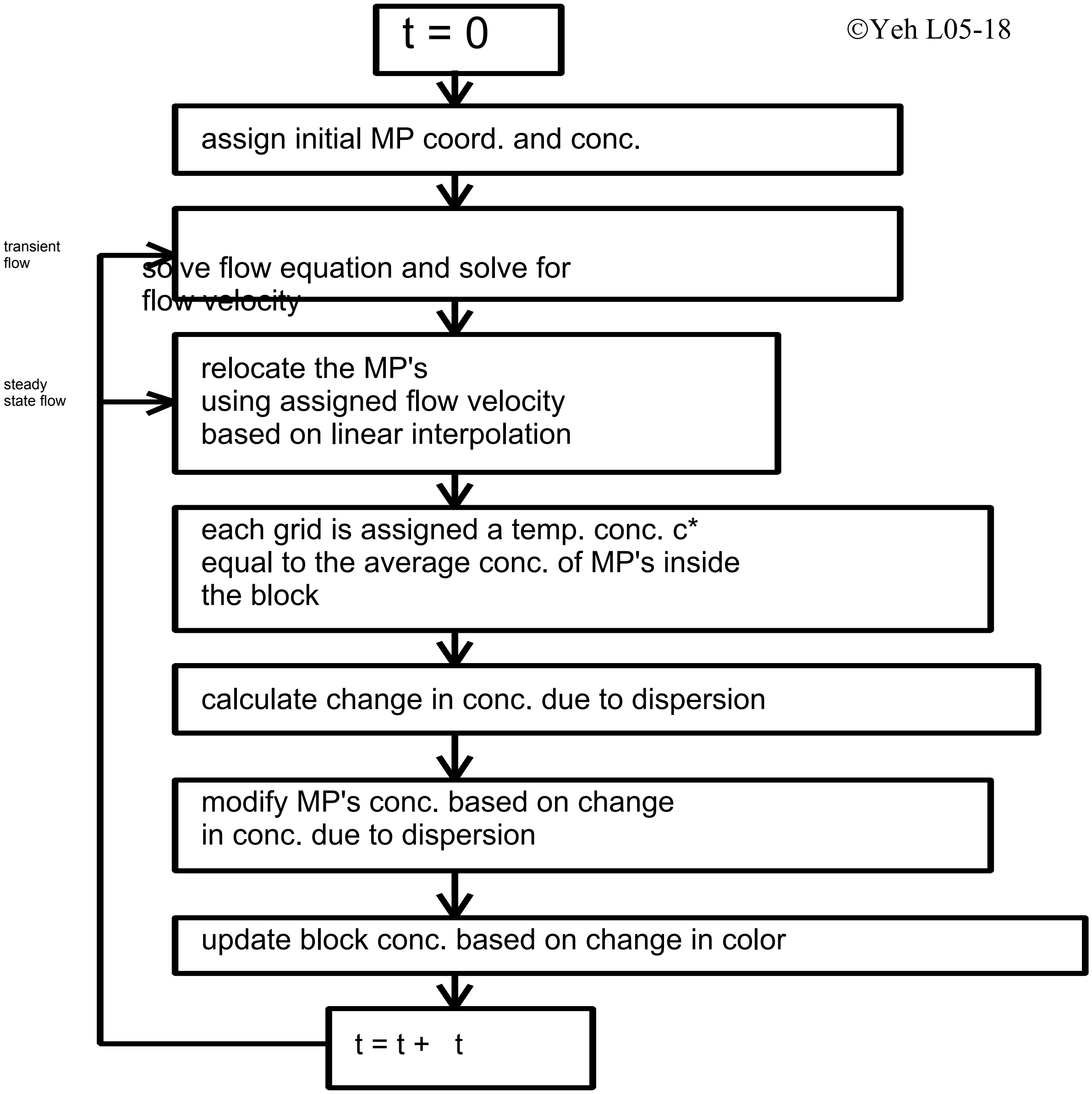
$$c_{ij}^{n+1} = c^{*n}_{\ ij} + \Delta c_{i,j}^n$$

This completes the step from t^n to t^{n+1}. The procedure is repeated for each subsequent time step. A flow chart of these steps is illustrated in the figure.

The advantage of the methods of characteristics is that it can simulate solute transport under purely convection conditions (i.e., diffusion coefficient is zero). However, it suffers from numerical diffusion because of the average of moving point concentrations. It also is constrained by the Courant number, discussed earlier.

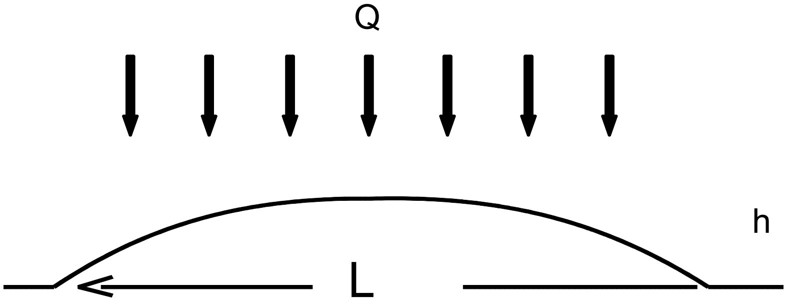
FLOW CHART FOR MOC

Particle Velocity Interpolation in Block-Centered Finite Difference Groundwater Flow Models. by D.J. Geode, *WRR*, 26(5), 925-940, 1990.


t = 0
assign initial MP coord. and conc.
transient flow
solve flow equation and solve for flow velocity
steady state flow
relocate the MP's using assigned flow velocity based on linear interpolation
each grid is assigned a temp. conc. c* equal to the average conc. of MP's inside the block
calculate change in conc. due to dispersion
modify MP's conc. based on change in conc. due to dispersion
update block conc. based on change in color
t = t + t

FINITE ELEMENT APPROACH

Finite Element Method also uses an approximate solution to obtain the solution of a partial deferential equation. For example, consider steady flow in a one-dimensional confined aquifer subject to a constant recharge, Q.



The governing partial differential equation for the flow is given by

$$K\frac{d^2h}{dx^2}+Q=0$$

Suppose $\hat{h}(x)$ is an approximate solution to the partial differential equation.

$$K\frac{d^2\hat{h}(x)}{dx^2}+Q=R(x)\neq 0$$

Since $\hat{h}(x)$ is an approximate solution, it will not satisfy the partial differential equation. To account for the approximate nature of the solution, one of the approaches used in the finite element is the weighted residual method, i.e.,

$$\int_0^L w_i(x)R(x)dx=0$$

where i is the index for weighting function, w; the number of w's depends on the number of unknown to be sought in the approximate solution. There are several choices for the weighting functions.

1. Collocation Method
2. Subdomain Method
3. Galerkin's Method
4. Least Squares Method

Here we will focus on the last two methods.

1. Least Squares Method (Regression type)

This method uses the error (residual) as the weighting function.

$$Error = E = \int_0^L [R(x)]^2\, dx$$

The aim of the approach is to find a solution that yields minimum E. This aim can be accomplished by optimization. That is, the error is minimized with respect to the unknown coefficients in the approximate solution.

Example:

Say, $\hat{h}(x) = A\sin\frac{\pi x}{L}$

where A is unknown (i.e., a regression model) and $\sin\frac{\pi x}{L}$ is the shape function.

$$E = \int_0^L \left[-\frac{K\pi^2}{L^2} A\sin\frac{\pi x}{L} + Q \right]^2 dx = \frac{A^2 L}{2}\left[\frac{K\pi^2}{L^2}\right]^2 - \frac{4QK\pi}{L}A + Q^2 L$$

Minimizing E by taking its derivative with respect to the unknown A and setting the resultant to zero yields

$$\frac{\partial E}{\partial A} = AL\left[\frac{K\pi^2}{L^2}\right]^2 - \frac{4QK\pi}{L} = 0$$

Solve for

$$A = \frac{4QL^2}{\pi^3 K}$$

Then the approximate solution for the problem is

$$\hat{h}(x) = \frac{4QL^2}{\pi^3 K}\sin\frac{\pi x}{L}$$

2. Galerkin's Method.

Instead of using the least-squares criterion, this method uses a weighted residual criterion:

$$\int_0^L w_i(x) R(x) dx = 0$$

in which the shape function of the approximate solution is used as $w_i(x)$. Again, we use

$$\hat{h}(x) = A\sin\frac{\pi x}{L}$$

as our approximate solution. The weighted residual criterion becomes

$$\int_0^L \sin\frac{\pi x}{L}\left[-K\frac{A\pi^2}{L^2}\sin\frac{\pi x}{L} + Q \right] dx = 0$$

Integrating the criterion yields

$$\frac{K\pi^2 A}{2L} - \frac{2QL}{\pi} = 0$$

Then, the unknown is derived:

$$A = \frac{4QL^2}{\pi^3 K}$$

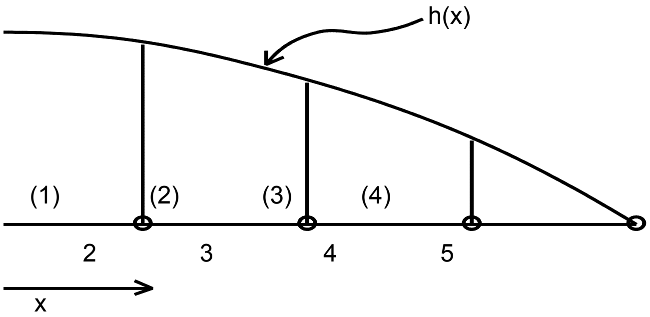
Subsequently, the approximate solution is given as

$$\hat{h}(x) = \frac{4QL^2}{\pi^3 K}\sin\frac{\pi x}{L}$$

(2) Select Interpolation Functions

The interpolation functions sometimes are called shape functions, basis functions, or interpolating polynomial. The finite element method assigns values of a field variable at finite element nodes, and then uses a shape function to represent the variation of field variable over the element

Example: Consider the hydraulic head h(x) as a continuous field variable.



If we choose a shape function, N_i^e , for each element (where $i = 1.2$ nodes) such that

$$N_1^e = \frac{1}{L}(x_2 - x)$$

$$N_2^e = \frac{1}{L}(x - x_1)$$

where

L = the length of the element #1

x_1 & x_2 = the coordinates of nodes 1 & 2

x = distance from x_1

The head variation between the two nodes thus is fully described by

$$h(x) = h_1 N_1^e + h_2 N_2^e$$

For example, Examine element #1 and assume we know head values at x_1 and x_2, what's the head value at x = L/3?

$$h(\frac{L}{3}) = \frac{1}{L}(x_2 - x)h_1 + \frac{1}{L}(x - x_1)h_2 = \frac{1}{L}(L - \frac{L}{3})h_1 + \frac{1}{L}(\frac{L}{3} - 0)h_2 = \frac{2}{3}h_1 + \frac{1}{3}h_2$$

Therefore, the head field within an element can be expressed in general as

$$h(x) = \sum_{i=1}^{n^e} h_i N_i^e(x)$$

where n_e is the total number of nodes in the element; h_i is the value of head at node i; N_i^e is the shape function associated with node i.

In general, polynomials are used as shape functions. The degree (order) of the polynomial depends on:

(1) Number of nodes assigned to the element.
(2) The nature of the unknown at each node.
(3) Certain continuity requirements imposed at the nodes.

A linear shape function is the most simple function that commonly used. A linear shape function for a one-dimensional element consists of

$$N_i(x) = \frac{X_j - x}{L} \qquad N_j(x) = \frac{x - X_i}{L}$$

Two important properties of 1-D element

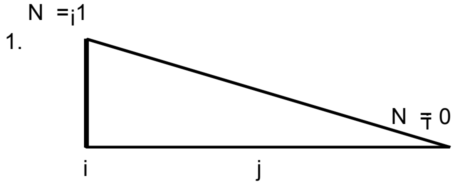
First property.

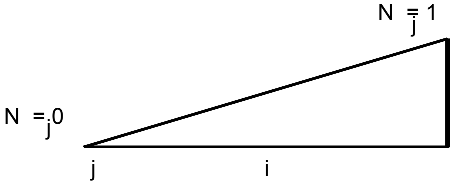
$$at \quad x = X_i \quad N_i = \frac{X_j - X_i}{L} = 1$$

$$at \quad x = X_j \quad N_i = \frac{X_j - X_j}{L} = 0$$

at $x = X_j \quad N_i = 1$

At $x = X_j \quad N_j = 0$





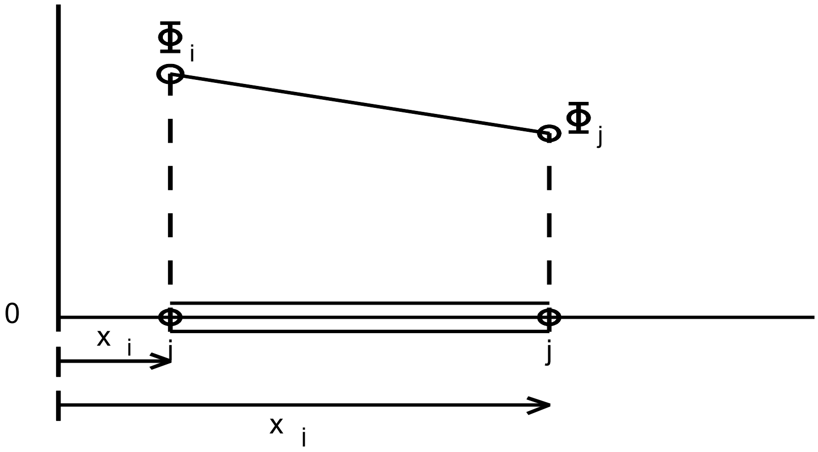
The second property

$N_j^{(x)} + N_i^{(x)} = 1$ at any x

Development of Linear Shape functions (Linear Interpolation Polynomials)

In general, most shape functions are polynomials. We will start with the simplest polynomials (linear ones).

1-D element (1 element-two nodes).



Consider 1-D flow through a soil block. If we know head values, Φ_i and Φ_j , at nodes i and j, respectively, how can we determine a shape function to represent the variation of φ between Φ_i and Φj?

According to the graph,

$\phi = \alpha_1 + \alpha_2 x$ (Linear polynomial)

and we know

$\phi = \Phi_i$ at $x = X_i$

$\phi = \Phi_j$ at $x = X_j$

The question is that what are the coefficients α_1 and α_2 that will satisfy the above two facts:

$$\Phi_i = \alpha_1 + \alpha_2 X_i \qquad \text{-----(1)}$$

$$\Phi_j = \alpha_1 + \alpha_2 X_j \qquad \text{----(2)}$$

Solve for α_1 and α_2

$$\alpha_1 = \frac{\Phi_i X_i - \Phi_j X_j}{L}$$

$$\alpha_1 = \frac{\Phi_j - \Phi_i}{L}$$

where $L = X_j - X_i$

Substituting α1 and α2 into the linear polynomial produces

$$\phi = \left(\frac{\Phi_i X_j - \Phi_j X_i}{L}\right) + \left(\frac{\Phi_j - \Phi_i}{L}\right)x$$

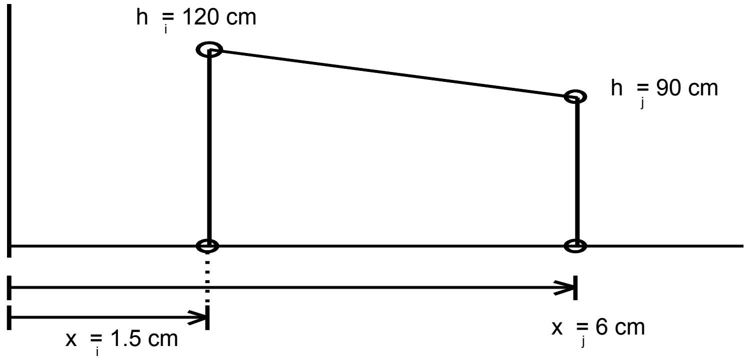
$$= \left(\frac{X_j - x}{L}\right)\Phi_i + \left(\frac{x - X_i}{L}\right)\Phi_j$$

$$= N_i \Phi_i + N_j \Phi_j$$

$$\phi = [N]\{\Phi\}$$

In other words, using the linear shape function, we say that the head variation between the two known values is described by a linearly weighted sum of Φ_i and Φ_j.

Example: 1-D flow



The head value, h(x), within the element is given by:

$$h(x) = \left(\frac{X_j - x}{L}\right)h_i + \left(\frac{x - X_i}{L}\right)h_j$$

$$L = x_j - x_i = 4.5cm$$

What is h(x = 4.0 cm) = ?

$$h(4.0) = \left(\frac{6.0 - 4.0}{4.5}\right)120 + \left(\frac{4 - 1.5}{4.5}\right)90 = 103.33cm$$

Gradient:

$$\frac{dh}{dl} = -\frac{1}{L}h_i + \frac{1}{L}h_j = \frac{1}{L}(h_j - h_i) = -6.67cm/cm$$

Formulation of One-Dimensional Convection-Dispersion Equation by Galerkin's Method.

Consider solute transport in a semi-infinite sand column and assume that the process can be described by the convection-dispersion equation

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} - D\frac{\partial^2 c}{\partial x^2} = 0$$

with boundary conditions

$$c(0,t) = c_0 \qquad t \ge 0$$

$$c(\infty,t) = 0 \qquad t \ge 0$$

and initial conditions

$$c(x,0) = 0 \qquad x > 0$$

Formulation of Element Matrices

Use the Galerkin Criterion at the element level,

$$\int_0^L \ell(\hat{c})N_i dx = 0$$

where *i=1* and *2*, N_i is a linear element shape function, and let

$$\ell(c) = \frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} - D\frac{\partial^2 c}{\partial x^2} = 0$$

The Galerkin criterion becomes

$$\int_0^L \left(\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} - D\frac{\partial^2 c}{\partial x^2} \right) N_i dx = 0$$

Integration of the third term in the bracket by parts yields

$$\int_0^L D\frac{\partial^2 c}{\partial x^2} N_i dx = N_i D\frac{\partial c}{\partial x}\bigg|_{x=0}^{x=L} - \int_0^L D\frac{\partial c}{\partial x}\frac{\partial N_i}{dx} dx$$

$$= q_L - q_0 - \int_0^L D\frac{\partial c}{\partial x}\frac{\partial N_i}{dx} dx$$

The Galerkin criterion is then written as

$$\int_0^L (\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x}) N_i dx + \int_0^L D\frac{\partial c}{\partial x}\frac{\partial N_i}{dx} dx = -q_L + q_0$$

Now, we use the following as the approximation solution

$$c(x,t) \simeq \hat{c}(x,t) = \sum_{i=1}^{2} N_j(x)\hat{c}_j(t)$$

where N_j is the shape function for the element and $\hat{c}_j(t)$ is the approximated solution of concentration at node j of the element. Substitution of this approximated solution to the equation results in

$$\sum_{i=1}^{2}\int_0^L \left[N_i N_j \frac{d\hat{c}_j}{dt} + uN_i \frac{\partial N_j}{\partial x}\hat{c}_j + D\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x}\hat{c}_j \right] dx = 0$$

This formula is called the consistent formulation of the convection-dispersion equation. Another approach is called the lumped approach, which is given as follows

$$\sum_{i=1}^{2}\int_0^L \left[N_i \frac{d\hat{c}_j}{dt} + uN_i \frac{\partial N_j}{\partial x}\frac{\partial N_i}{\partial x}\hat{c}_j + D\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x}\hat{c}_j \right] dx = 0$$

In matrix forms

$$\begin{array}{cccc} [A] & \{\hat{c}\} & +[B] & \left\{\dfrac{d\hat{c}}{dt}\right\} = 0 \\ 2\times 2 & 2\times 1 & 2\times 2 & 2\times 1 \end{array}$$

Elements in matrices [A] and [B] are

$$a_{ij} = \int_0^L \left[D\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + uN_i \frac{\partial N_j}{\partial x} \right] dx$$

$$b_{ij} = \int_0^L N_i N_j dx \qquad \text{for the consistent formulation}$$

$$b_{ij} = \int_0^L N_i dx \qquad \text{for the lumped formulation}$$

Rewrite these elements in matrix form

$$[A] = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = D\int_0^L \begin{bmatrix} \dfrac{\partial N_1}{\partial x}\dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_1}{\partial x}\dfrac{\partial N_2}{\partial x} \\ \dfrac{\partial N_2}{\partial x}\dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_2}{\partial x}\dfrac{\partial N_2}{\partial x} \end{bmatrix} dx + u\int_0^L \begin{bmatrix} N_1\dfrac{\partial N_1}{\partial x} & N_1\dfrac{\partial N_2}{\partial x} \\ N_2\dfrac{\partial N_1}{\partial x} & N_2\dfrac{\partial N_2}{\partial x} \end{bmatrix} dx$$

Used the linear shape function

$$N_i = 1 - \frac{x}{L}, N_j = \frac{x}{L}$$

Integrals within the A matrix can be expressed as

$$\int_0^L \frac{\partial N_1}{\partial x}\frac{\partial N_1}{\partial x}dx = \int_0^L \left(\frac{-1}{L}\right)\left(\frac{-1}{L}\right)dx = \frac{1}{L}$$

$$\int_0^L \frac{\partial N_1}{\partial x}dx = \int_0^L \left(1-\frac{x}{L}\right)\left(\frac{-1}{L}\right)dx = -\frac{1}{2}$$

Therefore,

$$[A] = \frac{D}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \frac{u}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}$$

Similarly,

$$[B] = \int_0^L \begin{bmatrix} N_1N_1 & N_1N_2 \\ N_2N_1 & N_2N_2 \end{bmatrix} dx = L\begin{bmatrix} 1/3 & 1/6 \\ 1/6 & 1/3 \end{bmatrix}$$ for the consistent formulation and

$$[B] = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix}$$ for the lumped formulation

Now, we will employ a finite difference approximation of time derivative term:

$$[A]\left\{\varepsilon\{\hat{c}\}^{t+\Delta t} + (1-\varepsilon)\{\hat{c}\}^t\right\} + \frac{1}{\Delta t}[B]\left(\{\hat{c}\}^{t+\Delta t} - \{\hat{c}\}^t\right) = \{Q\}$$

$\varepsilon = 1$ Implicit scheme
$\varepsilon = 1/2$ Crankle-Nicolson
$\varepsilon = 0$ the time approximation is explicit

The finite element formulation for each element thus becomes

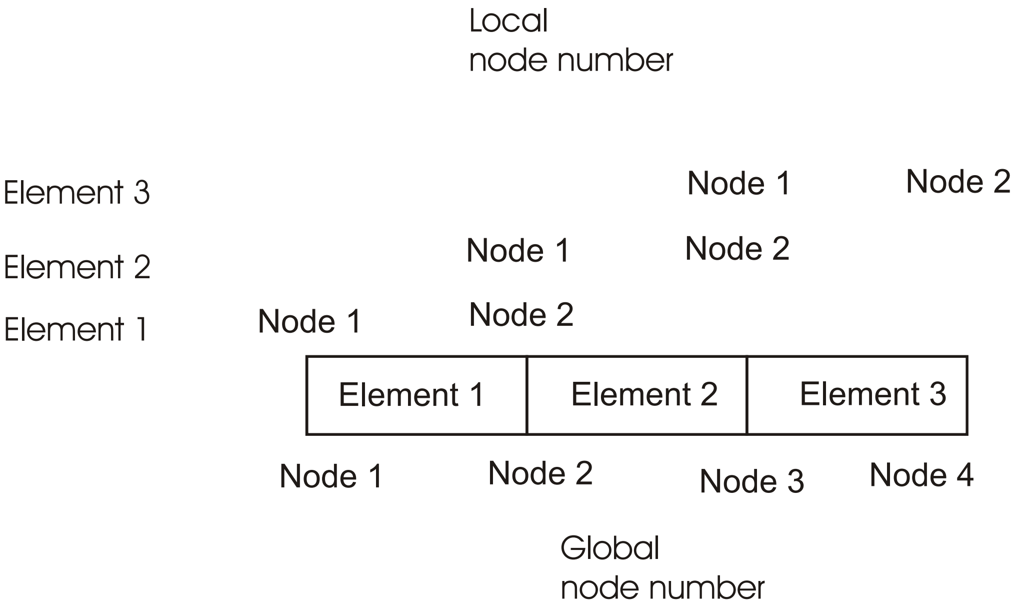
$$\left[\varepsilon\frac{D}{L^e}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \varepsilon\frac{u}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} + \frac{L^e}{\Delta t}\begin{bmatrix} 1/3 & 1/6 \\ 1/6 & 1/3 \end{bmatrix}\right]\begin{Bmatrix} \hat{c}_1 \\ \hat{c}_2 \end{Bmatrix}^{t+\Delta t} =$$

$$\left[(\varepsilon-1)\frac{D}{L^e}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + (\varepsilon-1)\frac{u}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} + \frac{L^e}{\Delta t}\begin{bmatrix} 1/3 & 1/6 \\ 1/6 & 1/3 \end{bmatrix}\right]\begin{Bmatrix} \hat{c}_1 \\ \hat{c}_2 \end{Bmatrix}^{t}$$

Next, we will assemble the element matrices for each element to form a global matrix and then incorporate the given boundary and initial conditions. We afterwards solve the global matrix equation for the concentration at each node at the next time step.

Formulation of Global Matrices

Example: The solution domain consists of three elements and four global nodes. Each element has two nodes.



Adding the matrices of element 1 to the global matrices

$$\left[\frac{\varepsilon D}{L^e}\begin{bmatrix}1 & -1 & 0 & 0\\ -1 & 1 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{L^e}{\Delta t}\begin{bmatrix}1/3 & 1/6 & 0 & 0\\ 1/6 & 1/3 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{u\varepsilon}{2}\begin{bmatrix}-1 & 1 & 0 & 0\\ -1 & 1 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\right]\begin{Bmatrix}\hat{c}_1\\ \hat{c}_2\\ \hat{c}_3\\ \hat{c}_4\end{Bmatrix}^{t+\Delta t}=$$

$$\left[\frac{(\varepsilon-1)D}{L^e}\begin{bmatrix}1 & -1 & 0 & 0\\ -1 & 1 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{L^e}{\Delta t}\begin{bmatrix}1/3 & 1/6 & 0 & 0\\ 1/6 & 1/3 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{u(\varepsilon-1)}{2}\begin{bmatrix}-1 & 1 & 0 & 0\\ -1 & 1 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\right]\begin{Bmatrix}\hat{c}_1\\ \hat{c}_2\\ \hat{c}_3\\ \hat{c}_4\end{Bmatrix}^{t}$$

Adding the matrices of element 2 to the global matrices

$$\left[\frac{\varepsilon D}{L^e}\begin{bmatrix}1 & -1 & 0 & 0\\ -1 & 2 & -1 & 0\\ 0 & -1 & 1 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{L^e}{\Delta t}\begin{bmatrix}1/3 & 1/6 & 0 & 0\\ 1/6 & 2/3 & 1/6 & 0\\ 0 & 1/6 & 1/3 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{u\varepsilon}{2}\begin{bmatrix}-1 & 1 & 0 & 0\\ -1 & 0 & 1 & 0\\ 0 & -1 & 1 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\right]\begin{Bmatrix}\hat{c}_1\\ \hat{c}_2\\ \hat{c}_3\\ \hat{c}_4\end{Bmatrix}^{t+\Delta t}=$$

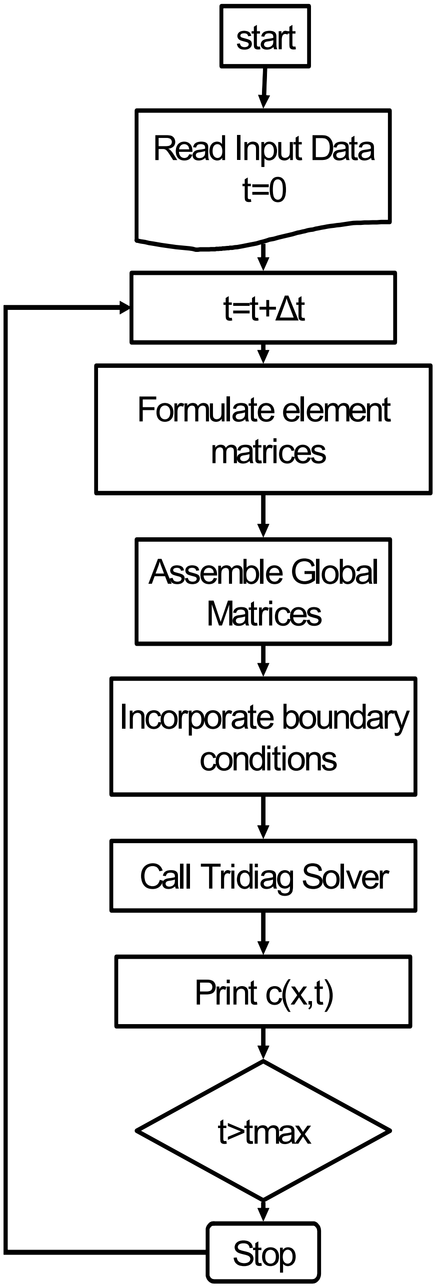
$$\left[\frac{(\varepsilon-1)D}{L^e}\begin{bmatrix}1 & -1 & 0 & 0\\ -1 & 2 & -1 & 0\\ 0 & -1 & 1 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{L^e}{\Delta t}\begin{bmatrix}1/3 & 1/6 & 0 & 0\\ 1/6 & 2/3 & 1/6 & 0\\ 0 & 1/6 & 1/3 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}+\frac{u(\varepsilon-1)}{2}\begin{bmatrix}-1 & 1 & 0 & 0\\ -1 & 0 & 1 & 0\\ 0 & -1 & 1 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\right]\begin{Bmatrix}\hat{c}_1\\ \hat{c}_2\\ \hat{c}_3\\ \hat{c}_4\end{Bmatrix}^{t}$$

Adding the matrices of element 3 to the global matrices

$$\left[\frac{\varepsilon D}{L^e}\begin{bmatrix}1 & -1 & 0 & 0\\ -1 & 2 & -1 & 0\\ 0 & -1 & 2 & -1\\ 0 & 0 & -1 & 1\end{bmatrix}+\frac{L^e}{\Delta t}\begin{bmatrix}1/3 & 1/6 & 0 & 0\\ 1/6 & 2/3 & 1/6 & 0\\ 0 & 1/6 & 2/3 & 1/6\\ 0 & 0 & 1/6 & 1/3\end{bmatrix}+\frac{u\varepsilon}{2}\begin{bmatrix}-1 & 1 & 0 & 0\\ -1 & 0 & 1 & 0\\ 0 & -1 & 0 & 1\\ 0 & 0 & -1 & 1\end{bmatrix}\right]\begin{Bmatrix}\hat{c}_1\\ \hat{c}_2\\ \hat{c}_3\\ \hat{c}_4\end{Bmatrix}^{t+\Delta t}=$$

$$\left[\frac{(\varepsilon-1)D}{L^e}\begin{bmatrix}1 & -1 & 0 & 0\\ -1 & 2 & -1 & 0\\ 0 & -1 & 2 & -1\\ 0 & 0 & -1 & 1\end{bmatrix}+\frac{L^e}{\Delta t}\begin{bmatrix}1/3 & 1/6 & 0 & 0\\ 1/6 & 2/3 & 1/6 & 0\\ 0 & 1/6 & 2/3 & 1/6\\ 0 & 0 & 1/6 & 1/3\end{bmatrix}+\frac{u(\varepsilon-1)}{2}\begin{bmatrix}-1 & 1 & 0 & 0\\ -1 & 0 & 1 & 0\\ 0 & -1 & 0 & 1\\ 0 & 0 & -1 & 1\end{bmatrix}\right]\begin{Bmatrix}\hat{c}_1\\ \hat{c}_2\\ \hat{c}_3\\ \hat{c}_4\end{Bmatrix}^{t}$$

Solve the global matrices for cs.


start
Read Input Data
t=0
t=t+Δt
Formulate element matrices
Assemble Global Matrices
Incorporate boundary conditions
Call Tridiag Solver
Print c(x,t)
t>tmax
Stop