**Lecture Packet #9: Numerical Modeling of Groundwater Flow**

<u>Simulation:</u> The prediction of quantities of interest (dependent variables) based upon an equation or series of equations that describe system behavior under a set of assumed simplifications.

### Groundwater Flow Simulation

- Predict hydraulic heads (1D, 2D, 3D)
- For particular conditions – confined, unconfined, isotropic, anisotropic, homogeneous, heterogeneous, infinite, finite, steady, transient.
- Varying levels of complexity:
    - Analystic solutions – Theim, Theis, etc.
        - Advantages: exact, simple, cheap, can provide sufficient insight
    - Analog simulation
    - Physical models – scale models of aquifers
    - Numerical Simulation

### Numerical Simulation

- Given a PDE and appropriate ICs and BCs
- Discretize the system
- Approximate the PDE corresponding to the discretization
- Solve the approximated PDE on a computer
- Commonly finite differences or finite elements for GW flow
- Advantages: can handle complex geometries, ICs, and C conditions; can be used for nonlinear systems.

### Beware of the term "Model" and how it is used

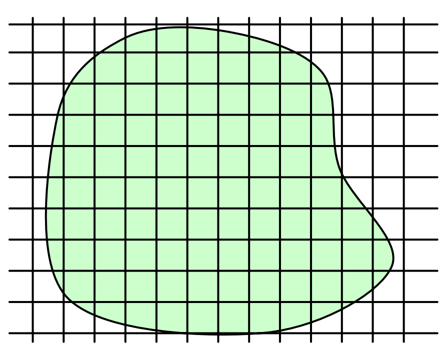
- "A model should be used as simple as possible, but not simpler"
- Mathematical "model" – a PDE
- Numerical "model" – a particular technique is applied
- Computer or simulation "model" – a code

### Finite Differences

A numerical method that approximates the governing PDE by replacing the derivatives in the equation with their respective difference representations.

Procedure involves: Grid (or Mesh) and Equation

Grid – a representation of the physical domain that enables one to account for the boundaries and internal features



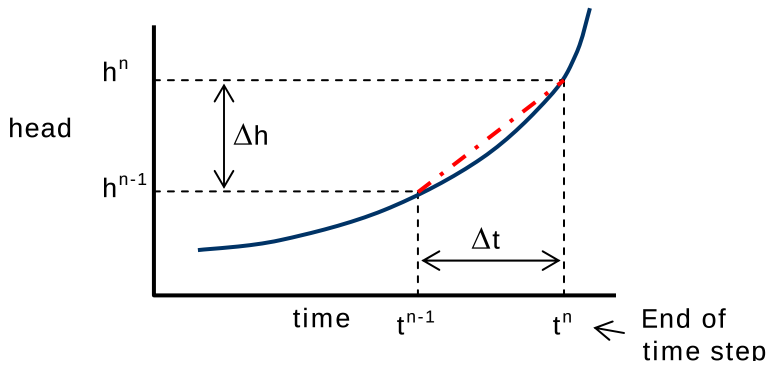
**Equation – Difference Approximation of Derivatives**

$$\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2} = \frac{S}{T}\frac{\partial h}{\partial t}$$ 2D flow equation

Approximating the Time Derivative:

*Backward Difference:*

$$\left(\frac{\partial h_{i,j}}{\partial t}\right)_{n\Delta t} \approx \frac{\Delta h}{\Delta t} \approx \frac{h_{i,j,n} - h_{i,j,n-1}}{\Delta t}$$

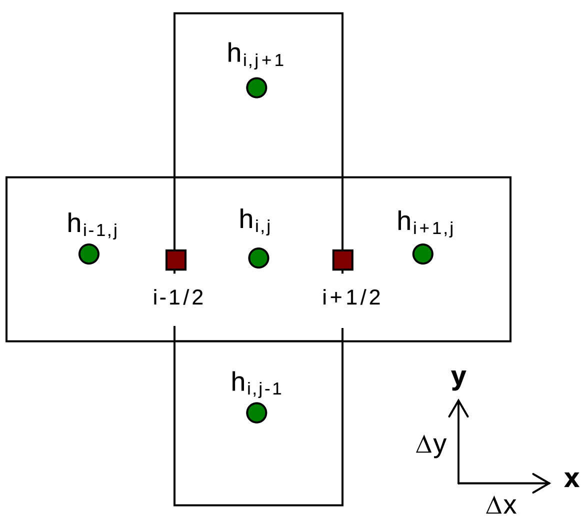



where,

- n = the current time step index
- $\Delta t$ = time step
- i,j = x and y coordinate indices

Approximating the Space Derivatives:

Consider a 2D discretization, if we assume that the grid spacing in the x-direction and y-direction are the same, our discretized grid for an internal node will be:



*In the x-direction*:

Approximate derivative at location $h_{i,j}$:

$$\left(\frac{\partial^2 h}{\partial x^2}\right)$$

Approximate the **second** spatial derivative at **i,j** as follows:

$$\left(\frac{\partial^2 h}{\partial x^2}\right) = \frac{\partial}{\partial x}\left(\frac{\partial h}{\partial x}\right) \approx \frac{\left(\frac{\partial h_{i+1/2,j}}{\partial x} - \frac{\partial h_{i-1/2,j}}{\partial x}\right)}{\Delta x}$$

We can approximate the **first** spatial derivative at i-1/2,j and i+1/2,j as follows:

$$\left(\frac{\partial h_{i-1/2,j}}{\partial x}\right) \approx \left(\frac{h_{i,j} - h_{i-1,j}}{\Delta x}\right) \text{ and } \left(\frac{\partial h_{i+1/2,j}}{\partial x}\right) \approx \left(\frac{h_{i+1,j} - h_{i,j}}{\Delta x}\right)$$

Substitute the above equations to obtain:

$$\left(\frac{\partial^2 h_{i,j}}{\partial x^2}\right) \approx \frac{\left(\frac{h_{i+1,j} - h_{i,j}}{\Delta x} - \frac{h_{i,j} - h_{i-1,j}}{\Delta x}\right)}{\Delta x}$$

or

$$\left(\frac{\partial^2 h_{i,j}}{\partial x^2}\right) \approx \frac{h_{i-1,j} - 2h_{i,j} + h_{i+1,j}}{\Delta x^2}$$

In the y-direction:

$$\left(\frac{\partial^2 h_{i,j}}{\partial y^2}\right) \approx \frac{h_{i,j-1} - 2h_{i,j} + h_{i,j+1}}{\Delta y^2}$$

Combining Flow Equation Terms:

$$\frac{\partial^2 h_{i,j}}{\partial x^2} + \frac{\partial^2 h_{i,j}}{\partial y^2} = \frac{S}{T}\left(\frac{\partial h_{i,j}}{\partial t}\right)_{\eta \Delta t}$$

For the backwards difference time derivative (note: all left hand side values of h are for time = n-1)

$$\frac{h_{i-1,j} - 2h_{i,j} + h_{i+1,j}}{\Delta x^2} + \frac{h_{i,j-1} - 2h_{i,j} + h_{i,j+1}}{\Delta y^2} = \frac{S}{T}\frac{h_{i,j,n} - h_{i,j,n-1}}{\Delta t}$$

⟵ Linear diff. eq'n. This eq'n is not explicit for h at any particular time

If $\Delta$x and $\Delta$y are equal – this is the finite difference groundwater flow equation

$$\frac{h_{i-1,j} + h_{i+1,j} + h_{i,j-1} + h_{i,j+1} - 4h_{i,j}}{\Delta x^2} = \frac{S}{T}\frac{h_{i,j,n} - h_{i,j,n-1}}{\Delta t}$$

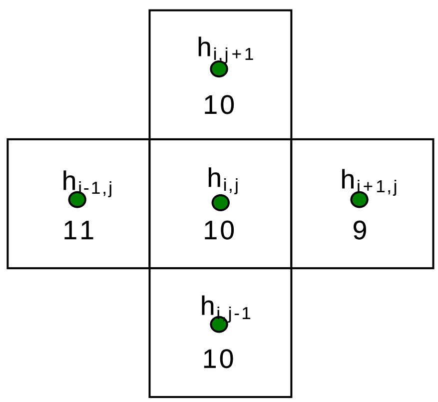
What does the finite-difference equation indicate for steady state conditions?

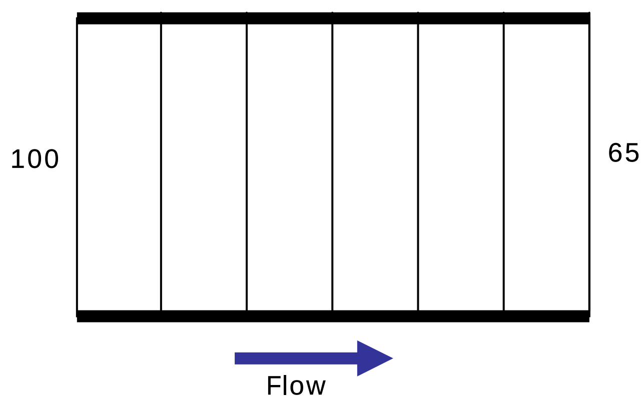
$$S\frac{\partial h}{\partial t} \approx S\left(\frac{h_{i,j,n} - h_{i,j,n-1}}{\Delta t}\right) = 0$$

$$\frac{h_{i-1,j} + h_{i+1,j} + h_{i,j-1} + h_{i,j+1} - 4h_{i,j}}{\Delta x^2} = 0$$

$$h_{i-1,j} + h_{i+1,j} + h_{i,j-1} + h_{i,j+1} - 4h_{i,j} = 0$$

$$\frac{h_{i-1,j} + h_{i+1,j} + h_{i,j-1} + h_{i,j+1}}{4} = h_{i,j}$$



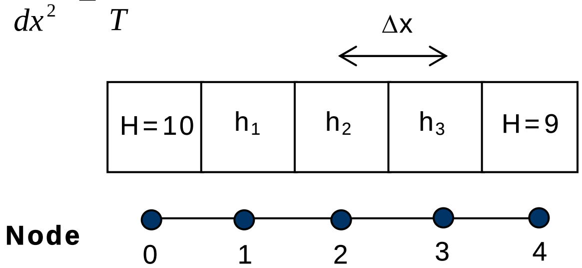




Value of head at node is the average of the surrounding nodes (for SS isotropic homogeneous case)

Consider the 1D steady-state flow equation with a sink is:

$$T\frac{d^2H}{dx^2} - w' = 0 \qquad \text{or}$$

$$\frac{d^2H}{dx^2} = \frac{w'}{T}$$



The nodal finite difference equation is:

$$\frac{h_{i,j-1} - 2h_{i,j} + h_{i,j+1}}{\Delta x^2} = \frac{w'}{T}$$

Or

$$h_{i,j-1} - 2h_{i,j} + h_{i,j+1} = \frac{(\Delta x)^2 w'}{T}$$

<u>Node 1:</u>

$1H_0 + -2h_1 + 1h_2 = 0$

$1(10) + -2h_1 + 1h_2 = 0$

$-2h_1 + 1h_2 = -10$

<u>Node 2:</u>

$1h_1 + -2h_2 + 1h_3 = 0$

<u>Node 3:</u>

Node 3 has a forcing term due to the pumping of the well. This translates into an initial righthand side of the equation:

Given $\Delta x = 0.1$, $\frac{(\Delta x)^2 w'}{T} = \frac{(0.1)^2 1}{.01} = 1$

$1h_2 + -2h_3 + 1H_4 = 1$

$1h_2 + -2h_3 + 1(9) = 1$

$1h_2 + -2h_3 = 1-9 = -8$

The <u>system</u> of finite-difference equations consists of 3 equations and 3 unknowns

$$\begin{array}{cccccc} -2h_1 & 1h_2 & & = & -10 \\ h_1 & -2h_2 & 1h_3 & = & 0 \\ & 1h_2 & -2h_3 & = & -8 \end{array}$$

**Unknowns** **Knowns**

Or in <u>coeffiecient matrix form</u>

$$\begin{pmatrix} -2 & 1 & \\ 1 & -2 & 1 \\ & 1 & -2 \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} -10 \\ 0 \\ -8 \end{pmatrix}$$

FD Coeff. Matrix — Unknown Heads — RHS containing boundary conditions and known pumping

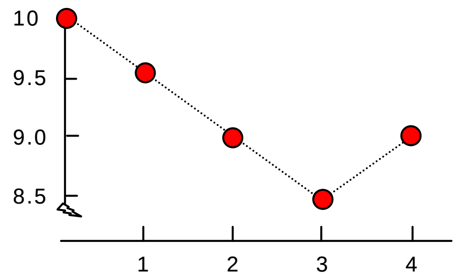
Or in matrix notation it can be written as

Ah = b'

A   is the matrix of difference coefficients
H   is the vector of unknown heads
b'  is the RHS vector of known quantities

A computational linear solve yields the vector **h**:

$$\begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} 9.5 \\ 9 \\ 8.5 \end{pmatrix}$$




**Transient Simulation Using Finite Differences**

Procedure: March Through Time

- Start with initial conditions (these are known)
- Solve for heads at end of first time step $\Delta t$; this give the spatial distribution of head (a map) after a small time increment.
- Given known heads at end of first time step solve for heads at the end of the second time step.
- With known value at the end of time step solve for next time step – this is called marching through time

$Ah_n = b^*$     where     $b^* = b' + h_{n-1}$

The right-hand side always contains knowns.  The matrix A is the matrix of finite difference coefficients reflecting the system parameters and discretization.

So if you can solve spatial equations for one time step. Then you can solve it for as many time steps as you like.

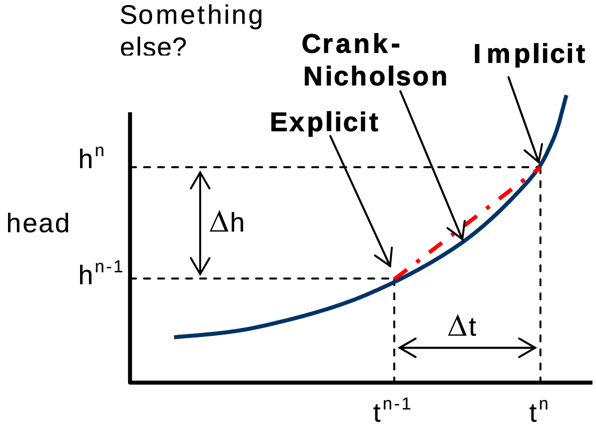
The time step must be small when changes in heads are rapid – such as, when you start to pump a well, $\Delta t$ must be seconds or minutes.  It can be increased as changes in head become smaller.

**FD Simulation Models**:  codes that solve the above system of linear algebraic equations – fairly robust.

**Time Stepping**

$$\left\{\frac{h_{i-1}-2h_i+h_{i+1}}{\Delta x^2}\right\} \quad = \frac{S}{T}\frac{h_{i,n}-h_{i,n-1}}{\Delta t}$$

n? n-1? Something else?




$$\frac{T\Delta t}{s\Delta x^2}\left(h_{i-1}-2h_i+h_{i+1}\right)=h_{i,n}-h_{i,n-1}$$

**Explicit:** $h_{i,n} = ch_{i-1,n-1} + (1-2C)h_{i,n-1} + ch_{i+1,n-1}$

Where: $c=\frac{T\Delta t}{s\Delta x^2}$

Easy to calculate – no linear algebra, but unstable if time-steps are too large.

**Implicit:** $ch_{i-1,n} - (1+2C)h_{i,n} + ch_{i+1,n} = -h_{i,n-1}$

Results in system of linear equations that must be solved simultaneously. Stable, but issues of accuracy.

**Crank Nicholson:** $\frac{c}{2}h_{i-1,n}-(1+c)h_{i,n}+\frac{c}{2}h_{i+1,n}=(c-1)h_{i-1,n}-\frac{c}{2}\left(h_{i-1,n-1}+h_{i-1,n-1}\right)$

Not much more numerical expense than fully implicit, but more accurate

**Boundary Conditions:**

Constant Head – Don't write equation for constant head node. Use value of constant head in equations for neighboring nodes.

Flux Boundary – replace first derivative term with constant

$$\left(\frac{\partial^2 h_{i,j}}{\partial x^2}\right)\approx\frac{\left(\frac{h_{i-1,j}-h_{i,j}}{\Delta x}-\frac{h_{i,j}-h_{i+1,j}}{\Delta x}\right)}{\Delta x}$$

$$\frac{h_{i,j}-h_{i+1,j}}{\Delta x}=\frac{Q}{T}$$