

GROUNDWATER FLOW MODELING



- Groundwater flow modeling implies performing numerical experiments on a groundwater flow model.
- The objective of such an experimentation
 - for practicing engineers - to check the feasibility of any human intervention into the groundwater system, e.g., pumpage, recharge etc.
 - For groundwater academics - to understand various processes involved in the groundwater system.

GROUNDWATER FLOW MODEL



- a tool to project the State variables of the groundwater system for an assigned pattern of forcing function, and known initial and boundary conditions and parameters.
- **State Variables:** the variables that describe the “state” of a system.
 - **Mandatory:** Piezometric head or Water table elevation.
 - **Problem-specific:** variables that are essentially derived from the mandatory variables or other combination. for example, the head distribution in space and time. These could include, depending upon the problem at hand, depth to water table, static storage, influent/ effluent seepage, outflow to sea, sea water intrusion etc.



- **Forcing Function: function comprises of different activities such as:**
 - Withdrawals (i.e., pumpage)
 - Recharge (derived from- rainfall, applied irrigation, seepage from surface water bodies etc.)
 - Evapo-transpiration from the saturated zone
- **Initial conditions:**
 - Condition which comprised initial distribution of state variables
- **Boundary conditions:**
 - Conditions describing the extent to which the hydraulic condition influences across the boundary.

COMOPONENTS OF A GW FLOW MODEL



1. An equation (algebraic or differential) governing the flow
2. An algorithm to solve the chosen equation numerically to compute the time and space distribution of the head
3. A set of algorithms to compute the problem-specific state variables from the pre-computed head distributions
4. Computer codes to implement the selected algorithms



- **Governing Equations:** an expression of the continuity equation in the context of groundwater flow, and stated as:
 - Across any selected domain of saturated flow, the difference between the inflow and the outflow rates equals the rate of change of the storage of water in that domain.
 - The selected domain is usually an infinitesimally small element.
 - ✦ small volume (for 3-D flow)
 - ✦ small area (for 2-D flow)
 - ✦ small length (for 1-D flow)



○ Two types of inflows/ outflows

- ✦ **Internal inflow/outflow:** *ones occurring on account of the prevalent hydraulic gradients.*
 - expressed in terms of the space derivatives of the head, (by invoking Darcy's law.)
- ✦ **external inflow or outflow:** *one driven by the forcing function.*
 - expressed in terms of the time derivative of the head, (e.g., rate of change of the storage)

- Gradient driven flow rates in x- and Y-direction:

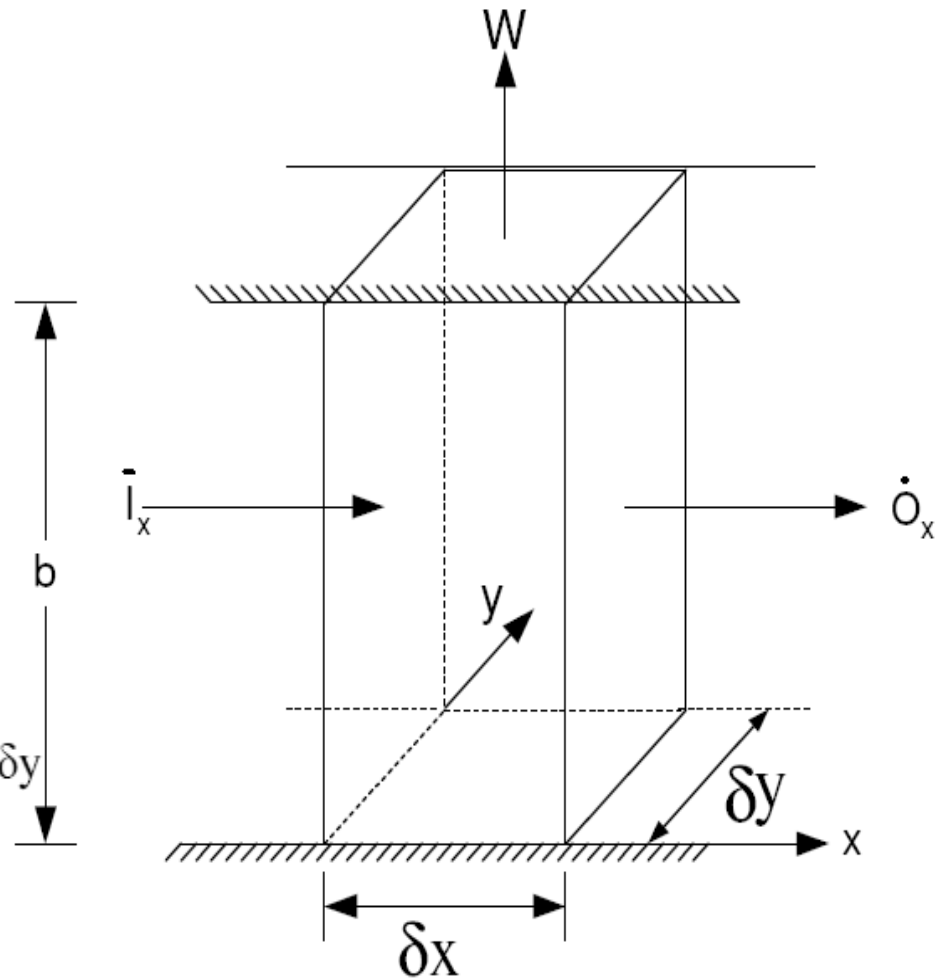
$$\dot{I}_x - \dot{O}_x = \frac{\partial}{\partial x} \left(T_{xx} \frac{\partial h}{\partial x} \right) \delta x \delta y$$

$$\dot{I}_y - \dot{O}_y = \frac{\partial}{\partial y} \left(T_{yy} \frac{\partial h}{\partial y} \right) \delta x \delta y$$

Total gradient driven (Inflow - Outflow) rate

$$= \frac{\partial}{\partial x} \left(T_{xx} \frac{\partial h}{\partial x} \right) \delta x \delta y + \frac{\partial}{\partial y} \left(T_{yy} \frac{\partial h}{\partial y} \right) \delta x \delta y$$

Forcing function driven out flow rate = $W \delta x \delta y$





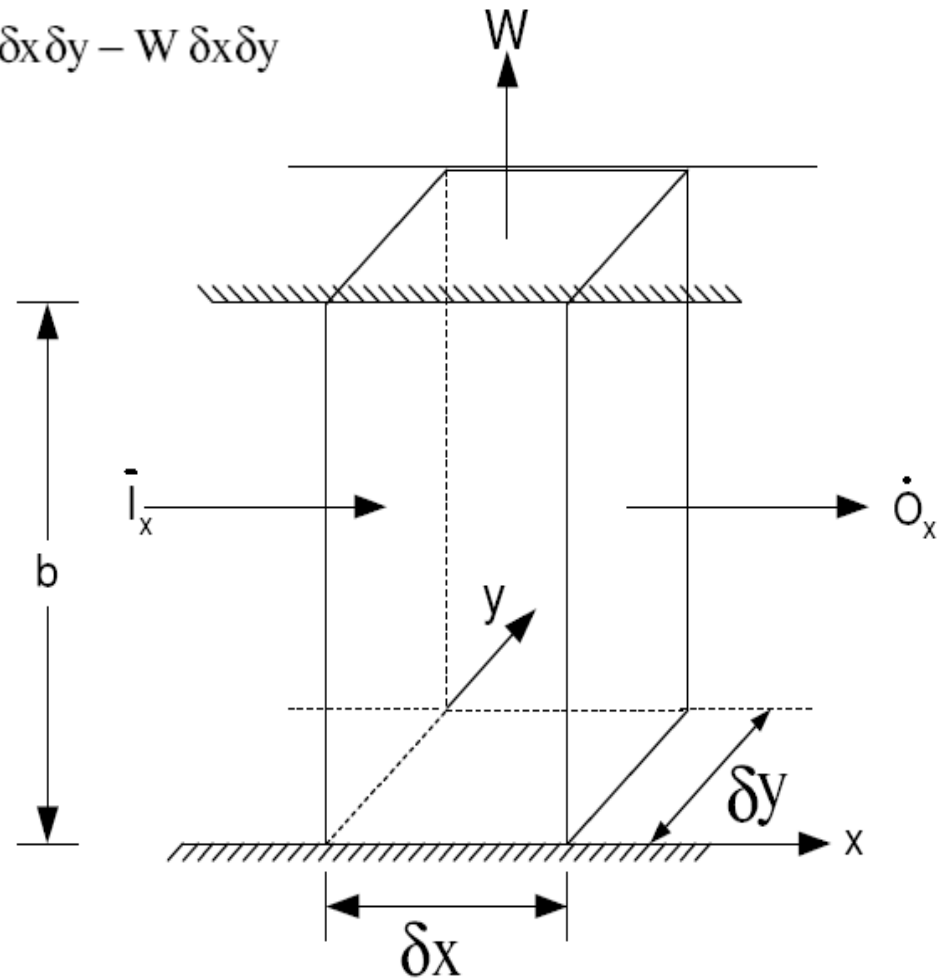
$$\text{Total } (\dot{I} - \dot{O}) = \frac{\partial}{\partial x} \left(T_{xx} \frac{\partial h}{\partial x} \right) \delta x \delta y + \frac{\partial}{\partial y} \left(T_{yy} \frac{\partial h}{\partial y} \right) \delta x \delta y - W \delta x \delta y$$

$$\text{Rate of change of storage} = \frac{\partial h}{\partial t} S \delta x \delta y$$

Resulting governing differential equation:

$$\frac{\partial}{\partial x} \left(T_{xx} \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y} \left(T_{yy} \frac{\partial h}{\partial y} \right) - W = S \frac{\partial h}{\partial t}$$

Where W is the forcing function (net external abstraction rate LT^{-1}) and S is storage coefficient.



Solution Algorithms



- **Finite difference method**
- Finite element method

Finite difference method



- This method essentially involves the following steps:
 - **Discretization of space and time**
 - ✦ the area over which the system response is to be simulated, is discretized by a finite number of points- usually known as *nodes*. Similarly the time domain, i.e., the period over which the response is to be simulated, is discretized by a finite number of *discrete times*.
 - ✦ Thus, a spatial distribution of any variable (say Storage coefficient) implies data comprising the values of the variable at each node. Similarly a spatial and temporal distribution of any variable (say, piezometric head) implies data comprising nodal values of the variable at the selected discrete times.



○ **Marching in time domain**

- ✦ A strategy adopted for computing the nodal values of the head at successively advancing discrete times.
- ✦ This essentially involves: knowing the nodal heads at the beginning of a time step and computing the heads at the end of the time step.
- ✦ the solution commences from the Initial condition and “marches” in the time domain.



- The nodal computation by formulation of linear algebraic equation as:
 - **Interior Nodes:** At each interior node, the space and time derivatives of the head appearing in the governing differential equations are approximated by corresponding finite differences.
 - **Boundary nodes:** An additional linear equation is obtained for each boundary node by invoking the respective known boundary condition.

Finite difference approximation



- Finite difference approximations involve applying Taylor's expansions to the equations (flow and transport) and approximating the derivatives in the equation
- Consider the governing equation and applying Taylor expansion

For two-dimensional transient flow conditions

$$\frac{\partial}{\partial x} \left(T_x \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y} \left(T_y \frac{\partial h}{\partial y} \right) = S \frac{\partial h}{\partial t} + Q$$

S = storativity [unitless],

Q = recharge or withdrawal per unit area [L/T]

T = transmissivity [L²/T]

Consider the governing equation and applying Taylor expansion



neglecting all the higher terms, and rearranging terms gives a useful approximation for $h''(x)$:

$$h''(x) = \left(\frac{d^2 h}{dx^2} \right)_x \approx \frac{1}{\Delta x^2} \{h(x + \Delta x) - 2h(x) + h(x - \Delta x)\}$$

$$h''(y) = \left(\frac{d^2 h}{dy^2} \right)_y \approx \frac{1}{\Delta y^2} \{h(y + \Delta y) - 2h(y) + h(y - \Delta y)\}$$

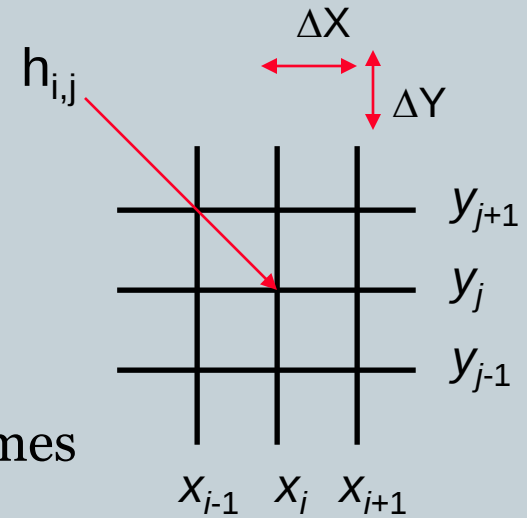
$$h'(t) = \left(\frac{dh}{dt} \right)_t \approx \frac{1}{2\Delta t} \{h(t + \Delta t) - h(t - \Delta t)\}$$

Numerical Soln to Laplace's Equation

$$\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2} = 0$$

For 2-D homogenous steady state flow

- Assume $\Delta x = \Delta y$ [regular square grid]
- Represent $h(x_i, y_j) = h_{i,j}$
- $h(x_i + \Delta x, y_j + \Delta y) = h_{i+1, j+1}$,



Thus replacing terms in Laplace, the F.D.E. becomes

$$h''(x_i, y_j)_x = h''_{i,j} \approx \frac{1}{\Delta x^2} \{h_{i-1,j} - 2h_{i,j} + h_{i+1,j}\}$$

- Do the same for the 'j' direction, and you have the following:

Approximation to Laplace's Eqn.



$$h''(x_i, y_j)_x + h''(x_i, y_j)_y \approx$$
$$\frac{1}{\Delta x^2} \left[\{h_{i-1,j} - 2h_{i,j} + h_{i+1,j}\} + \{h_{i,j-1} - 2h_{i,j} + h_{i,j+1}\} \right] =$$
$$\frac{1}{\Delta x^2} \left[h_{i-1,j} + h_{i,j-1} - 4h_{i,j} + h_{i+1,j} + h_{i,j+1} \right] = 0$$

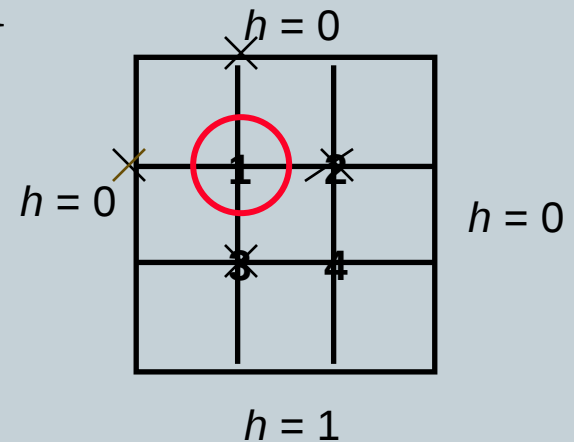
Summing terms and solving for $h_{i,j}$ gives:

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

Application to a Simple 3x3 Grid



- Start with boundary conditions assume $h_1 = h_2 = h_3 = h_4 = 0$ and initial estimate for h ,
- Get a new estimate for h_1 , call it h_1^{m+1}
- $h_1^{m+1} = (\text{top} + \text{right} + \text{bottom} + \text{left})/4$
- $h_1^{m+1} = \{0 + h_2 + h_3 + 0\}/4$
- Repeat four internal points - 2, 3, and 4 using same four-star average calculation



Application to a grid



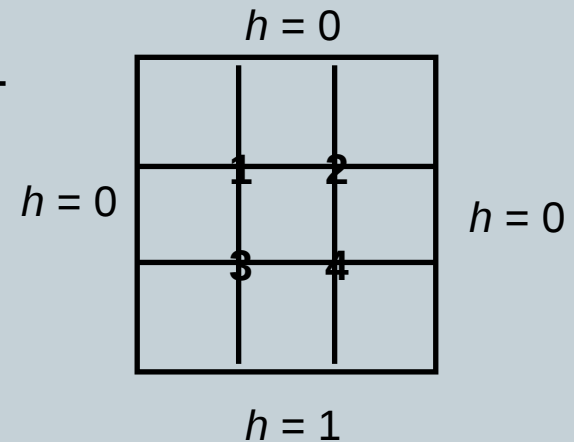
$$h_1^{m+1} = \{0 + h_2^m + h_3^m + 0\} / 4$$

$$h_2^{m+1} = \{0 + 0 + h_4^m + h_1^m\} / 4$$

$$h_3^{m+1} = \{h_1^m + h_4^m + 1 + 0\} / 4$$

$$h_4^{m+1} = \{h_2^m + 0 + 1 + h_3^m\} / 4$$

- Use the initial values and boundary conditions for the first approximation
- Use the updated values ($m+1$) for further approximations ($m+2$)
- Continue until the numbers don't change much (convergence!)



Calibration , Validation, and Sensitivity Analysis



- **Calibration** is the process of making the model match real-world data. Involves making several model runs, varying parameters until the 'best fit' is achieved.
- **Validation** is the process of confirming the validity of your calibration by using the model to fit an independent set of data.
- **Sensitivity Analysis** is the process of changing parameters to see the effects on the model results. The most sensitive parameters need to be checked for accuracy to ensure the best model.