

Numerical Approximation Methods - Groundwater 2D - Finite Difference Method

1. Mathematical Description

Differential Equation

The confined groundwater problem can be described by the set of differential equations for flow balance in the model domain and at the boundary of the model domain.

$$(1-1) \quad \frac{\partial}{\partial x_1} \left[K_{11} \frac{\partial h}{\partial x_1} \right] + \frac{\partial}{\partial x_2} \left[K_{22} \frac{\partial h}{\partial x_2} \right] + w = S_s \frac{\partial h}{\partial t}$$

K_{ii} hydraulic conductivity along x_i coordinate $i = 1, 2$

h hydraulic head

w volumetric flux (source/sink term)

S_s specific storage of the soil material (porous material)

x_1, x_2 Cartesian coordinates

t time coordinate

For 2D problems with constant hydraulic conductivity in each coordinate direction the equation can be simplified to:

$$(1-2) \quad K_{11} \frac{\partial^2 h}{\partial x_1^2} + K_{22} \frac{\partial^2 h}{\partial x_2^2} + w = S_s \frac{\partial h}{\partial t}$$

For steady groundwater flow the equation can be simplified to:

$$(1-3) \quad K_{11} \frac{\partial^2 h}{\partial x_1^2} + K_{22} \frac{\partial^2 h}{\partial x_2^2} + w = 0$$

The groundwater flux in the two coordinate directions is defined by:

$$(1-4) \quad \mathbf{q} = \mathbf{n}^T \mathbf{q} = \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}^T \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

\mathbf{n} normal vector at boundary

$$(1-5) \quad q_1 = - K_{11} \frac{\partial h}{\partial x_1} \quad \text{flux vector component}$$

$$(1-6) \quad q_2 = - K_{22} \frac{\partial h}{\partial x_2} \quad \text{flux vector component}$$

Boundary conditions:

$$(1-7) \quad h = h_0 \quad \text{Dirichlet boundary condition, given head}$$

$$(1-8) \quad q = q_B \quad \text{Neumann boundary condition, given flux}$$

$$(1-9) \quad q = f(h) \quad \text{Cauchy boundary condition, flux depends on head}$$

2. Finite Difference Method FDM

2.1 Spatial Approximation of the domain

The model domain is subdivided into equidistant sections and related nodes. A simple solution is a regular rectangle grid parallel to the coordinate axis.

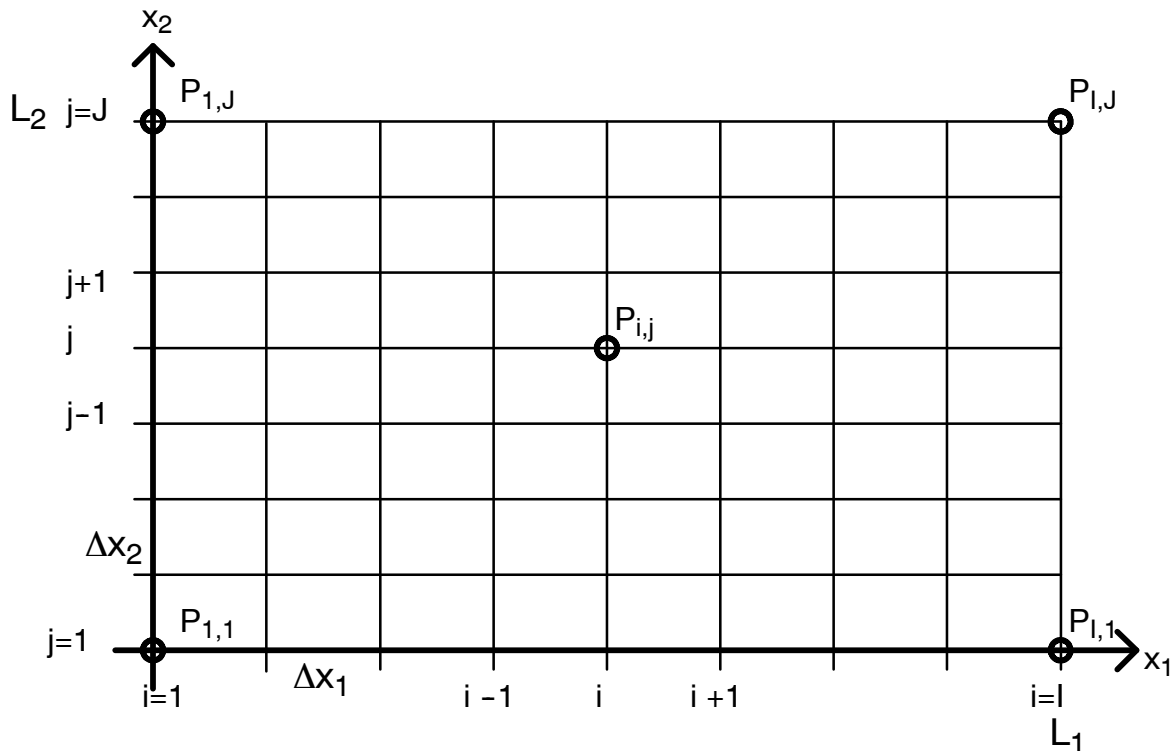


Figure 1: Spatial approximation by regular rectangle grid

I	number of nodes in direction 1
J	number of nodes in direction 2
x_{1i}	x_1 coordinate for nodes with first index i
x_{2j}	x_2 coordinate for nodes with second index j
$P_{i,j}$	node i,j with coordinates (x_{1i}, x_{2j})
L_1	length of the model domain in direction 1 $L_1 = I \Delta x_1$
L_2	length of the model domain in direction 2 $L_2 = J \Delta x_2$
N	number of nodes $N = I * J$

The physical state variable $h(x_1, x_2)$ is described by discrete functions. At each node the related function value $h_{i,j}$ is specified.

2.2 Approximation of the Differential Equation

The groundwater 2D differential equation is a Laplace's equation. To solve this equation the order of an approximation polynomial for $h(x_1, x_2)$ has to be two as minimum in each coordinate direction.

$$(2-1) \quad \frac{\partial^2 h}{\partial x_1^2} = \frac{1}{\Delta x_1^2} \mathbf{s}_{zze}^T \mathbf{h}_{e1}$$

$$(2-2) \quad \frac{\partial^2 h}{\partial x_2^2} = \frac{1}{\Delta x_2^2} \mathbf{s}_{zze}^T \mathbf{h}_{e2}$$

$$(2-3) \quad \mathbf{s}_{zze} = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \quad \text{shape function 2nd derivation vector}$$

$$(2-4) \quad \mathbf{h}_{e1} = \begin{bmatrix} h_{i-1,j} \\ h_{i,j} \\ h_{i+1,j} \end{bmatrix} \quad \text{head value vector at node i,j in direction 1}$$

$$(2-5) \quad \mathbf{h}_{e2} = \begin{bmatrix} h_{i,j-1} \\ h_{i,j} \\ h_{i,j+1} \end{bmatrix} \quad \text{head value vector at node i,j in direction 2}$$

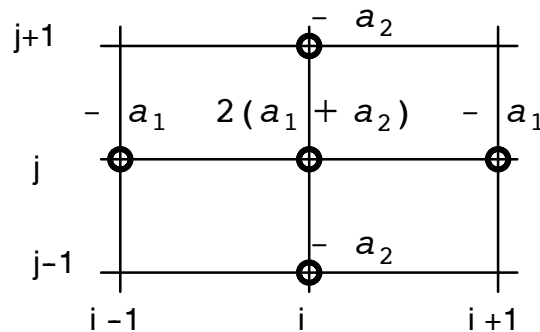
$$(2-6) \quad -\frac{K_{11}}{\Delta x_1^2} \mathbf{s}_{zze}^T \mathbf{h}_{e1} - \frac{K_{22}}{\Delta x_2^2} \mathbf{s}_{zze}^T \mathbf{h}_{e2} = w_{i,j}$$

approximated diff. equation at node n

$$(2-7) \quad w_{i,j} = w(x_{1i}, x_{2j}) \quad \text{flux load at node n,m}$$

The final equation for the differential equation at node n,m is:

$$(2-8) \quad \frac{K_{11}}{\Delta x_1^2} (-h_{i-1,j} + 2h_{i,j} - h_{i+1,j}) + \frac{K_{22}}{\Delta x_2^2} (-h_{i,j-1} + 2h_{i,j} - h_{i,j+1}) = w_{i,j}$$



$$a_1 = \frac{K_{11}}{\Delta x_1^2}$$

$$a_2 = \frac{K_{22}}{\Delta x_2^2}$$

Figure 2: 2D Finite Difference Scheme

2.3 Approximation of the Boundary Conditions

Two boundary conditions types are considered for the 2D groundwater flow approximation: given head values and given flux values.

Given head values can be directly set to the related node values

$$(2-9) \quad h(x_1 = x_{1i}, x_2 = x_{2j}) = h_{i,j}$$

The groundwater flux is specified by a differential equation, which can be approximated by a linear approach (see lecture noted geometrical modelling – shape functions). The model domain has a rectangle shape with four kind of normal vectors:

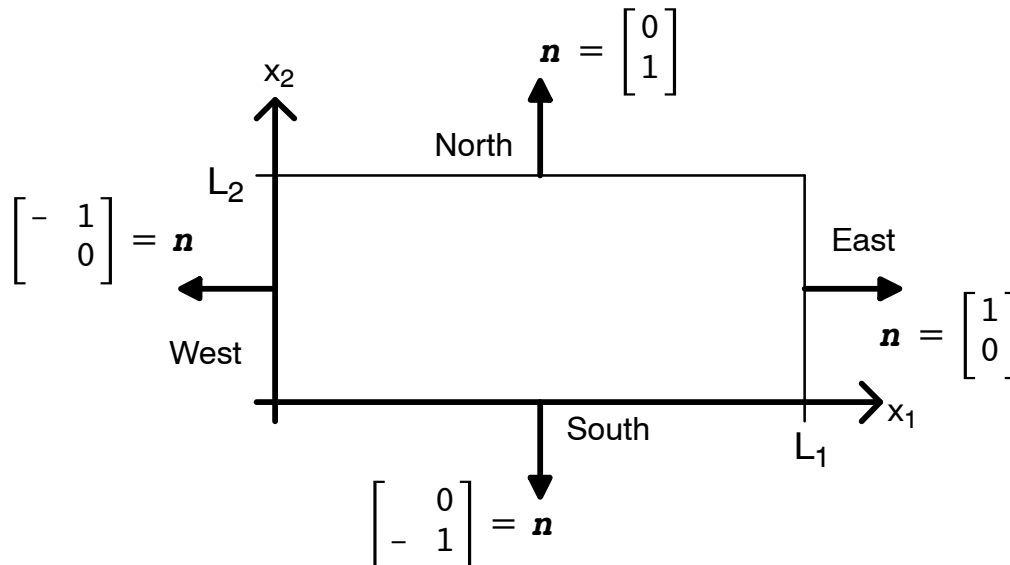


Figure 3: Model Domain Boundary Normal Vector

Depending on the boundary normal / model domain side four approximation equations are used:

$$(2-10) \quad q_B = \frac{-K_{11}}{\Delta x_1} (h_{I,j} - h_{I-1,j}) \quad \text{East}$$

$$(2-11) \quad q_B = \frac{-K_{22}}{\Delta x_2} (h_{i,J} - h_{i,J-1}) \quad \text{North}$$

$$(2-12) \quad q_B = \frac{-K_{11}}{\Delta x_1} (h_{2,j} - h_{1,j}) \quad \text{West}$$

$$(2-13) \quad q_B = \frac{-K_{22}}{\Delta x_2} (h_{i,2} - h_{i,1}) \quad \text{South}$$

2.4 Equation system

The groundwater flow system in Figure 2 is approximated at each inner node by equation (2-8). The given boundary conditions at the boundary nodes by the related equations (2-10) to (2-13). This leads to $N = I \cdot J$ equations with N unknown values ($h_{i,j}$).

The N equations can be written in a matrix-vector notation. The nodes and related physical state variables will be ordered in a sequence using the rule:

$$(2-14) \quad n = j * I + i \quad \text{index of node } i,j \text{ in the matrix-vector system}$$

The unknown head values are summarized in the vector \mathbf{h} ordered by the node index n . The known external flux values $w(x_1, x_2)$ are set in the load vector \mathbf{w} at the node related position. The coefficients related to the unknown head values at each node are considered in a system matrix.

$$(2-15) \quad \mathbf{K}\mathbf{h} = \mathbf{w}$$

$$(2-16) \quad \mathbf{h} = \begin{bmatrix} h_1 \\ \dots \\ h_n \\ \dots \\ h_N \end{bmatrix} \quad \text{(unknown) head vector}$$

$$(2-17) \quad \mathbf{w} = \begin{bmatrix} \dots \\ h_n \\ \dots \\ w_n \\ \dots \\ Q_{Bn} \\ \dots \end{bmatrix} \quad \text{(known) external load vector}$$

Figure 4 shows as example the node index of a system with 4 nodes in each direction.

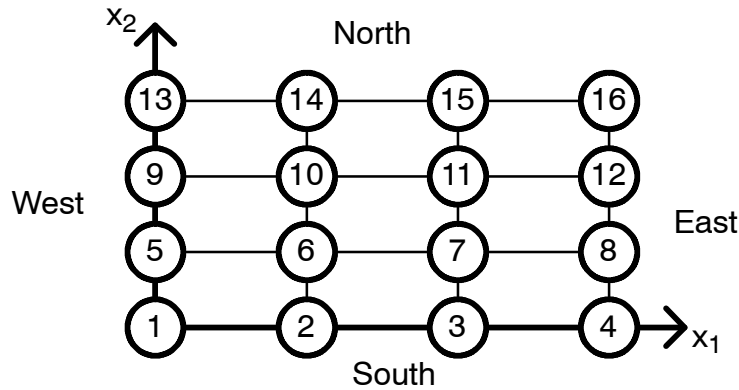


Figure 4: Node index n for regular grid 4×4

Node 6,7,10 and 11 are inner nodes.

Nodes 1-4 are the nodes at the South boundary with given head values (2-9).

Nodes 4,8,12,16 are the nodes at the East boundary with given flux values (2-10).

Nodes 13-16 are the nodes at the North boundary with given flux values (2-11).

Nodes 1,5,9,13 are the nodes at the West boundary with given flux values (2-12).

This leads to a band matrix with a bandwidth of 1 (node number in direction1).

1	1																h_1	h_1
2		1															h_2	h_2
3			1														h_3	h_3
4				1													h_4	h_4
5					h	$-h$											h_5	q_5
6		$-c$			$-b$	a	$-b$			$-c$							h_6	w_6
7			$-c$			$-b$	a	$-b$			$-c$						h_7	w_7
8							h	$-h$									h_8	q_8
9									h	$-h$							h_9	q_9
10						$-c$			$-b$	a	$-b$			$-c$			h_{10}	w_{10}
11							$-c$			$-b$	a	$-b$			$-c$		h_{11}	w_{11}
12										h	$-h$						h_{12}	q_{12}
13									v				$-v$				h_{13}	q_{13}
14										v				$-v$			h_{14}	q_{14}
15											v				$-v$		h_{15}	q_{15}
16												v				$-v$	h_{16}	q_{16}

Figure 5: System Matrix 4x4 Node Approximation

$$a = 2 \frac{K_{11}}{\Delta x_1^2} + 2 \frac{K_{22}}{\Delta x_2^2} \quad \text{see (2-8)}$$

$$b = \frac{K_{11}}{\Delta x_1^2} \quad \text{see (2-8)}$$

$$c = \frac{K_{22}}{\Delta x_2^2} \quad \text{see (2-8)}$$

$$h = \frac{K_{11}}{\Delta x_1} \quad \text{see (2-10),(2-12)}$$

$$v = \frac{K_{22}}{\Delta x_2} \quad \text{see (2-11),(2-13)}$$

3. Unsteady Flow

The basic equation of unsteady groundwater flow in 2D is (see (1-2)):

$$(3-1) \quad K_{11} \frac{\partial^2 h}{\partial x_1^2} + K_{22} \frac{\partial^2 h}{\partial x_2^2} = S_s \frac{\partial h}{\partial t} - w(t)$$

The left hand side of the equation has been spatial approximated using the finite difference method, which leads to the linear equation system (2-15). The right hand side of the equation contains the time depending part to define unsteady flow.

3.1 Time Approximation

The time approximation is usually done by introducing time steps with equidistant time increment within the desired time window.

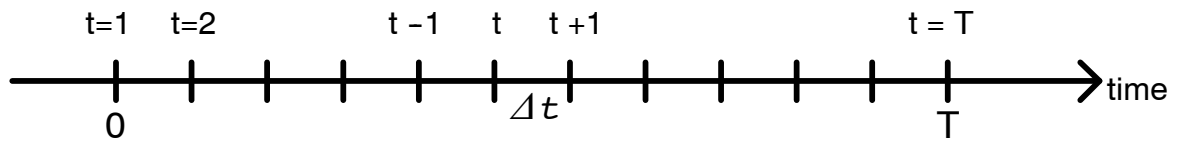


Figure 6: 1D time approximation

3.2 Time Depending Physical State Variables

The physical behaviour is described by relationships of discrete physical state variables. Time depending physical behaviour such as unsteady flow leads to time depending physical state variables. They are modelled in a discrete way by related values for each time step. In groundwater 2D the head variable is time depending. The spatial approximation leads towards the head (space) vector (2-16). This head vector is specified for each time step for unsteady flow:

$$(3-2) \quad \mathbf{h}^t = \begin{bmatrix} h_1 \\ \dots \\ h_n \\ \dots \\ h_N \end{bmatrix}^t = \begin{bmatrix} h_1^t \\ \dots \\ h_n^t \\ \dots \\ h_N^t \end{bmatrix} = \begin{bmatrix} h_{1,1}^t \\ \dots \\ h_{i,j}^t \\ \dots \\ h_{I,J}^t \end{bmatrix} \quad \text{head vector time step } t$$

$$(3-3) \quad h_n^t \quad \text{head value at node } n \text{ for time step } t$$

All other time depending physical state variables and boundary condition values are modelled in the same way by time step related values marked with a sup index t for time step t .

3.3 Numerical Approximation

Equation (3-1) as two-dimensional diffusion equation contains separated derivation on space and time. The derivation on space (2nd order) can be expressed by equation (2-1). Example in one direction:

$$(3-4) \quad \frac{\partial^2 h}{\partial x^2} = \frac{1}{\Delta x^2} \mathbf{S}_{zze}^T \mathbf{h}_e = \frac{h_{i-1} - 2h_i + h_{i+1}}{\Delta x^2}$$

For a time depending modelling this equation can be used on each time step:

$$(3-5) \quad \frac{\partial^2 h}{\partial x^2} \Big|_t = \frac{h_{n-1}^t - 2h_n^t + h_{n+1}^t}{\Delta x^2}$$

The 1st order derivation of the head variable on time in equation (3-1) can be approximated by

$$(3-6) \quad \frac{\partial h_n}{\partial t} = \frac{h_n^{t+1} - h_n^t}{\Delta t} \quad (\text{forward difference})$$

$$(3-7) \quad \frac{\partial h_n}{\partial t} = \frac{h_n^{t+1} - h_n^{t-1}}{2\Delta t} \quad (\text{central difference})$$

$$(3-8) \quad \frac{\partial h_n}{\partial t} = \frac{h_n^t - h_n^{t-1}}{\Delta t} \quad (\text{backward difference})$$

These basic derivation equations can be used for the numerical approximation of the general 2D diffusion equation.

$$(3-9) \quad \frac{\partial h}{\partial t} - a_1 \frac{\partial^2 h}{\partial x_1^2} - a_2 \frac{\partial^2 h}{\partial x_2^2} = b(t)$$
$$a_1 = \frac{K_{11}}{S_s}, \quad a_2 = \frac{K_{22}}{S_s}, \quad b(t) = \frac{w(t)}{S_s}$$

Combinations of different time step levels for spatial derivatives and different 1st order time derivatives lead to several different numerical schemes.

3.3.1 Explicit Schemes

Explicit schemes are using a time and space approximation which leads to equations, in which only one unknown state variable appears. These equations can be used to calculate the unknown values without solving an equation system.

FTCS Scheme (forward time, central space)

The FTCS scheme is an explicit scheme using equation (3-6) for time derivation and equation (3-5) for space derivation. This leads to:

$$(3-10) \quad \frac{h_{i,j}^{t+1} - h_{i,j}^t}{\Delta t} + \frac{a_1}{\Delta x_1^2} \left(-h_{i-1,j}^t + 2h_{i,j}^t - h_{i+1,j}^t \right) + \frac{a_2}{\Delta x_2^2} \left(-h_{i,j-1}^t + 2h_{i,j}^t - h_{i,j+1}^t \right) = b^t$$

When all head values of time level t are known, the head value at node i,j for time step $t+1$ can be explicitly calculated:

$$(3-11) \quad h_{i,j}^{t+1} = (1 - 2s_1 - 2s_2)h_{i,j}^t + s_1h_{i-1,j}^t + s_1h_{i+1,j}^t + s_2h_{i,j-1}^t + s_2h_{i,j+1}^t + \Delta t b^t$$

$$s_1 = a_1 \frac{\Delta t}{\Delta x_1^2}$$

$$s_2 = a_2 \frac{\Delta t}{\Delta x_2^2}$$

This scheme has a truncation error of $O(\Delta t, \Delta x_1^2, \Delta x_2^2)$ and a stability condition:

$$(3-12) \quad s_1 + s_2 \leq 0.5$$

3.3.2 Implicit Scheme

Implicit schemes are using a time and space approximations which links several unknown state variables in a linear equation. A set of such equations leads to a linear equation system, which has to be solved.

Similar to the FTCS scheme the discretisation of the spatial derivations can be done on the time level $t+1$. This leads to an implicit scheme with the equation:

$$(3-13) \quad -h_{i,j}^t = (-1 - 2s_1 - 2s_2)h_{i,j}^{t+1} + s_1h_{i-1,j}^{t+1} + s_1h_{i+1,j}^{t+1} + s_2h_{i,j-1}^{t+1} + s_2h_{i,j+1}^{t+1} + \Delta t b^{t+1}$$

This scheme has a truncation error of $O(\Delta t, \Delta x_1^2, \Delta x_2^2)$ and is unconditionally stable. However, for each time step a linear equation system has to be solved.

Several other implicit schemes are described in the literature. Example is the Crank-Nicolson Scheme using an average of the spatial derivations in time level t and time level $t+1$.

$$(3-14) \quad b^t = \frac{h_{i,j}^{t+1} - h_{i,j}^t}{\Delta t} + \frac{1}{2}a_1 \left[\frac{h_{i-1,j}^t + 2h_{i,j}^t - h_{i+1,j}^t}{\Delta x_1^2} + \frac{h_{i-1,j}^{t+1} + 2h_{i,j}^{t+1} - h_{i+1,j}^{t+1}}{\Delta x_1^2} \right] + \frac{1}{2}a_2 \left[\frac{h_{i,j-1}^t + 2h_{i,j}^t - h_{i,j+1}^t}{\Delta x_2^2} + \frac{h_{i,j-1}^{t+1} + 2h_{i,j}^{t+1} - h_{i,j+1}^{t+1}}{\Delta x_2^2} \right]$$

Advantage of this scheme is the truncation error of $O(\Delta t^2, \Delta x_1^2, \Delta x_2^2)$

3.4 Boundary Conditions

The numerical approximation of the boundary conditions for unsteady flow is similar for steady flow. The given external head and flux values could be steady or time depending.

Given head values can be directly set to the related node values.

$$(3-15) \quad h(x_1 = x_{1i}, x_2 = x_{2j}, t = \Delta t \quad t) = h_{i,j}^t$$

As described in 2.3 four types of groundwater flux boundary equations can be introduced in respect to the four borders of the rectangular model domain and the related normal vectors:

$$(3-16) \quad q_B^t = \frac{-K_{11}}{\Delta x_1} (h_{i,j}^t - h_{i-1,j}^t) \quad \text{East}$$

$$(3-17) \quad q_B^t = \frac{-K_{22}}{\Delta x_2} (h_{i,j}^t - h_{i,j-1}^t) \quad \text{North}$$

$$(3-18) \quad q_B^t = \frac{-K_{11}}{\Delta x_1} (h_{i,j}^t - h_{i+1,j}^t) \quad \text{West}$$

$$(3-19) \quad q_B^t = \frac{-K_{22}}{\Delta x_2} (h_{i,j}^t - h_{i,j+1}^t) \quad \text{South}$$

These equations can be used for the implicit scheme.

Flux Boundary Conditions for Explicit Schemes

The described equations for flux boundary conditions are not suitable for explicit schemes. The equations contain two state variables on the same time level. Explicit schemes require one unknown state variable on the new time level and several known state variables on other “older” time levels. Another problem is the truncation error of the space approximation $O(\Delta x_1, \Delta x_2)$, which is lower as the truncation error of the equation for the interior nodes (such as (3-11)).

This problem can be avoided by using a central difference with a virtual node outside the model domain. Equation (3-20) shows this for the western boundary:

$$(3-20) \quad q_B^t = \frac{-K_{11}}{2\Delta x_1} (h_{2,j}^t - h_{0,j}^t) \quad \text{West}$$

This can be rearranged for the head value outside the model domain:

$$(3-21) \quad \frac{2\Delta x_1}{K_{11}} q_B^t + h_{2,j}^t = h_{0,j}^t \quad \text{West}$$

The state value of the node outside the model domain can be eliminated using an interior equation such as (3-11) for node 1,j:

$$(3-22) \quad h_{1,j}^{t+1} = (1 - 2s_1 - 2s_2)h_{1,j}^t + s_1h_{0,j}^t + s_1h_{2,j}^t + s_2h_{1,j-1}^t + s_2h_{1,j+1}^t \quad \text{West}$$

The combination of equation (3-21) and (3-22) leads to an equation for the unknown head value at node 1,j for time step t+1 depending on known head values of time step t. This equation has the same truncation error as the equation for the interior nodes.

$$(3-23) \quad h_{1,j}^{t+1} = (1 - 2s_1 - 2s_2)h_{1,j}^t + s_1\left(\frac{2\Delta x_1}{K_{11}}q_B^t + h_{2,j}^t\right) + s_1h_{2,j}^t + s_2h_{1,j-1}^t + s_2h_{1,j+1}^t \quad \text{West}$$

Similar equations can be set up for all four boundaries of the rectangular model domain.

Central difference:

$$(3-24) \quad q_B^t = \frac{-K_{11}}{2\Delta x_1} (h_{I+1,j}^t - h_{I-1,j}^t) \quad \text{East}$$

$$(3-25) \quad q_B^t = \frac{-K_{22}}{2\Delta x_2} (h_{i,J+1}^t - h_{i,J-1}^t) \quad \text{North}$$

$$(3-26) \quad q_B^t = \frac{-K_{11}}{2\Delta x_1} (h_{2,j}^t - h_{0,j}^t) \quad \text{West}$$

$$(3-27) \quad q_B^t = \frac{-K_{22}}{2\Delta x_2} (h_{i,2}^t - h_{i,0}^t) \quad \text{South}$$

Head value outside domain:

$$(3-28) \quad -\frac{2\Delta x_1}{K_{11}} q_B^t + h_{I-1,j}^t = h_{I+1,j}^t \quad \text{East}$$

$$(3-29) \quad -\frac{2\Delta x_2}{K_{22}} q_B^t + h_{i,J-1}^t = h_{i,J+1}^t \quad \text{North}$$

$$(3-30) \quad \frac{2\Delta x_1}{K_{11}} q_B^t + h_{2,j}^t = h_{0,j}^t \quad \text{West}$$

$$(3-31) \quad \frac{2\Delta x_2}{K_{22}} q_B^t + h_{i,2}^t = h_{i,0}^t \quad \text{South}$$

Differential Equation

$$(3-32) \quad h_{I,j}^{t+1} = (1 - 2s_1 - 2s_2)h_{I,j}^t + s_1h_{I-1,j}^t + s_1h_{I+1,j}^t + s_2h_{I,j-1}^t + s_2h_{I,j+1}^t \quad \text{East}$$

$$(3-33) \quad h_{i,J}^{t+1} = (1 - 2s_1 - 2s_2)h_{i,J}^t + s_1h_{i-1,J}^t + s_1h_{i+1,J}^t + s_2h_{i,J-1}^t + s_2h_{i,J+1}^t \quad \text{North}$$

$$(3-34) \quad h_{1,j}^{t+1} = (1 - 2s_1 - 2s_2)h_{1,j}^t + s_1h_{0,j}^t + s_1h_{2,j}^t + s_2h_{1,j-1}^t + s_2h_{1,j+1}^t \quad \text{West}$$

$$(3-35) \quad h_{i,1}^{t+1} = (1 - 2s_1 - 2s_2)h_{i,1}^t + s_1h_{i-1,1}^t + s_1h_{i+1,1}^t + s_2h_{i,0}^t + s_2h_{i,2}^t \quad \text{South}$$

Final equations:

$$(3-36) \quad h_{I,j}^{t+1} = (1 - 2s_1 - 2s_2)h_{I,j}^t + \text{East}$$

$$s_1 h_{I-1,j}^t + s_1 \left(-\frac{2\Delta x_1}{K_{11}} q_B^t + h_{I-1,j}^t \right) + s_2 h_{I,j-1}^t + s_2 h_{I,j+1}^t$$

$$(3-37) \quad h_{i,J}^{t+1} = (1 - 2s_1 - 2s_2)h_{i,J}^t + \text{North}$$

$$s_1 h_{i-1,J}^t + s_1 h_{i+1,J}^t + s_2 h_{i,J-1}^t + s_2 \left(-\frac{2\Delta x_2}{K_{22}} q_B^t + h_{i,J-1}^t \right)$$

$$(3-38) \quad h_{i,1}^{t+1} = (1 - 2s_1 - 2s_2)h_{i,1}^t + \text{West}$$

$$s_1 h_{i-1,1}^t + s_1 h_{i+1,1}^t + s_2 \left(\frac{2\Delta x_2}{K_{22}} q_B^t + h_{i,2}^t \right) + s_2 h_{i,2}^t$$

$$(3-39) \quad h_{1,j}^{t+1} = (1 - 2s_1 - 2s_2)h_{1,j}^t + \text{South}$$

$$s_1 \left(\frac{2\Delta x_1}{K_{11}} q_B^t + h_{2,j}^t \right) + s_1 h_{2,j}^t + s_2 h_{1,j-1}^t + s_2 h_{1,j+1}^t$$

3.5 Initial Conditions

The initial condition problem of unsteady groundwater flow requires a complete set of known values for all physical state variables, solving the specified differential equations.

$$(3-40) \quad h(x_1 = x_{1i}, x_2 = x_{2j}, t = 1) = h_{i,j}^1$$

Usually these values can be specified by three types of methods:

- Setting up a simple physical state by physical insight
- Steady Flow Simulation
- Hotstart

The first version to specify initial conditions is using a simple physical state in the model domain which can be specified by physical insight without any simulations. Example for groundwater flow would be a system with constant head values at all boundaries and within the model domain. Disadvantage of this method are often constant physical states at the start of the simulation, which did not represent the initial conditions in nature. The model might require longer simulation over time until the impact of the initial conditions has been reduced efficiently.

Second option is the use of the steady flow simulation to generate a complete set of values for a valid physical state in the model domain. This allows specifying suitable but flexible boundary conditions and sink/source terms for the initial state and might reduce the non-physical impact of the initial conditions to unsteady flow situations during the simulation.

Third option is to use the result of a former unsteady simulation run as initial condition for a new time depending simulation. This can be used to (re)start/change a simulation on different time steps of a behaviour history.

4. Adaptive Grids for Unsteady Flow

Chapter 3. introduced the theoretical background for the numerical simulation of unsteady 2D groundwater flow (2D diffusion equation) using the finite difference method. The spatial approximation described in 2.1 is done by a regular uniform rectangular grid. The grid size is defined by the required minimum grid size to discretise the physical state variable in all part of the model domain. A local effect (e.g. large gradient of a shock wave), which requires a small grid size will lead to a small grid size in the whole domain. This will increase the number of nodes and related unknown values as well as computational resources.

Adaptive grids can be used to avoid the impact of local effects to the grid structure in the whole model domain. Key idea is to use in all model domains always a grid size suitable for the local requirements. Parts of the model domain with smooth changes of the physical state variables can be determined by a larger grid size. Sub domains with stronger changes of the physical state variables will be discretised by small grid size.

Example for such adaptive grids is the quad-tree based quadrilateral grid. A simple example for such grid is shown in Figure 7.

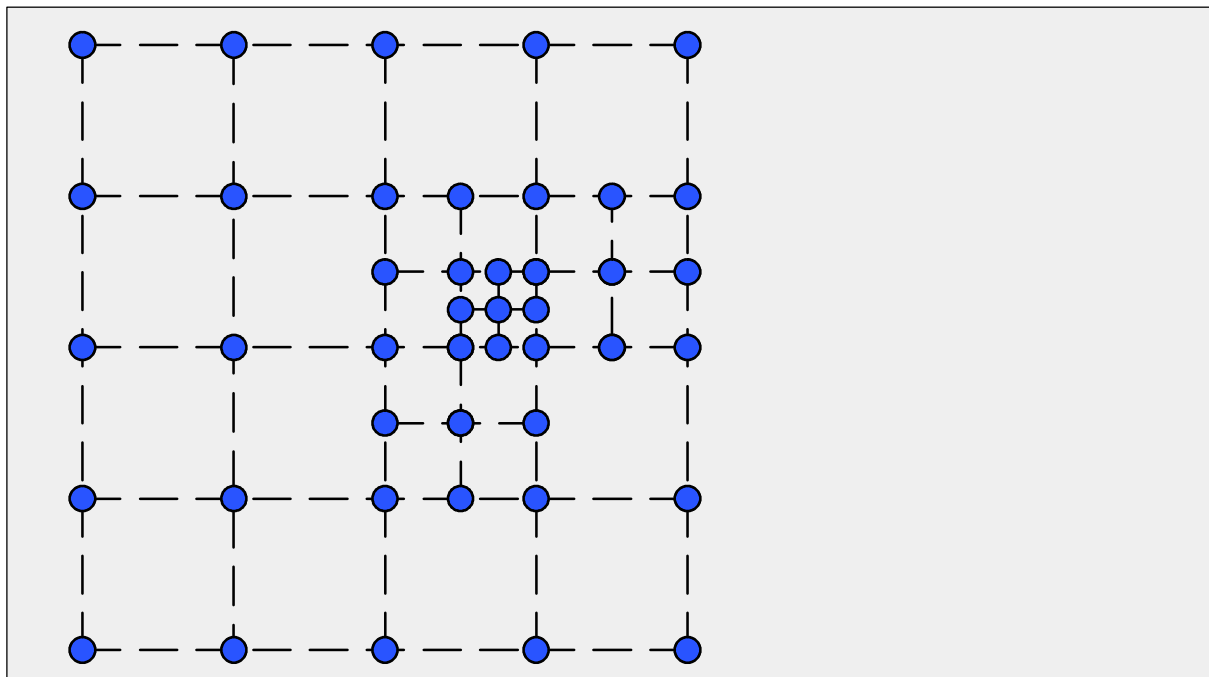


Figure 7: Example Quadrilateral Grid with local Refinement

Details on grid modelling for such adaptive grids are described in the related lecture notes. The application of such adaptive grid leads to three extension of the described methods:

- criteria for the level of local refinement
- scaling of physical state variables
- adaptation of the numerical scheme for “hanging” nodes

These three topics are touched briefly.

4.1 Refinement Criteria

Adaptive grids allow a local refinement. The level of refinement has to be defined by a combination of refinement criteria. Typical criteria are:

- numerical criteria
- physical state variable approximation

Numerical criteria are derived from the applied numerical scheme. Physical state variable approximation criteria are based on the “quality” of the applied approximation. Two simple examples for refinement criteria in unsteady 2D groundwater flow FDM are given below.

4.1.1 Numerical Criteria

Examples for numerical criteria are stability conditions. The FTCS scheme has a stability criteria defined in equation (3-12):

$$(4-1) \quad a_1 \frac{\Delta t}{\Delta x_1^2} + a_2 \frac{\Delta t}{\Delta x_2^2} \leq 0.5$$

For a predefined time step and equal spatial increments in both directions, the stability criteria can be expressed by:

$$(4-2) \quad 2(a_1 + a_2)\Delta t \leq \Delta x^2$$

This equation defines a maximum limit for the refinement to guarantee stability.

4.1.2 Criteria for physical state variable approximation

The grid size has an impact on the quality of the approximation of the physical state variables. For groundwater 2D the physical state variable $h(x_1, x_2)$ is expressed by a discrete approximation by element with quadratic shape functions. A simple check of the quality of the spatial approximation is the comparison of the gradient left and right (up and down) of a node based on a linear interpolation approach. Physically interpretation is the comparison of flux related values. The left / right or up /down gradients for the nodes i, j can be calculated by a forward and a backward difference;

$$(4-3) \quad g_R = \frac{\partial h_{i,j}}{\partial x} \Big|_R = \frac{h_{i+1,j} - h_{i,j}}{\Delta x} \quad (\text{forward difference})$$

$$(4-4) \quad g_L = \frac{\partial h_{i,j}}{\partial x} \Big|_L = \frac{h_{i,j} - h_{i-1,j}}{\Delta x} \quad (\text{backward difference})$$

The two values can be compared against a relative or an absolute limits.

$$(4-5) \quad |g_L - g_R| > \epsilon_1 \quad \text{absolute limit}$$

$$(4-6) \quad |g_L - g_R| > \epsilon_2 \max(|g_L|, |g_R|) \quad \text{relative limit}$$

If the change of gradient is not “smooth” enough, the grid will be adapted by refinement. Another option to compare gradient values is the central difference.

$$(4-7) \quad g_C = \frac{\partial h_{i,j}}{\partial x} \Big|_C = \frac{h_{i+1,j} - h_{i-1,j}}{2\Delta x} \quad (\text{central difference})$$

4.2 Scaling

Unsteady flow simulation requires known values for the “old” time step t to determine the “new” values for time step $t+1$. In case of an adaptive grid application, the number and location of nodes change between the two time steps. In case of refinement, new nodes are added, in case of coarseness some nodes are not any more considered.

The physical state variables are linked to nodes. Due to the grid adaptation, the related physical state variables have to be adapted to the new grid approximation level. In case of grid refinement this is called down scaling, in case of grid coarseness, this is called up scaling. Some simple methods for the unsteady FDM 2D groundwater flow example are described below.

4.2.1 Downscaling

Grid refinement leads to new nodes. For the new nodes related physical state variables values has to be determined. One simple method is an interpolation of the physical state variables at the new nodes of the new, lower quad-tree level based on the physical state variables at the nodes of the upper quad-tree level. This strategy can be applied on each level of adaptive grids.

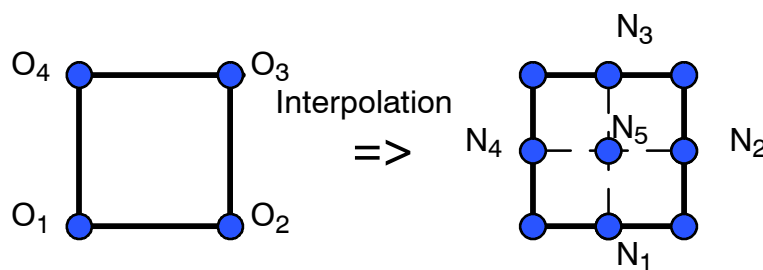


Figure 8: Interpolation for Refinement

O_i nodes on the upper quad-tree level with known values

N_i nodes on the lower quad-tree level with unknown values

The interpolation can be performed using the shape function approach described in the related lecture notes. For the inner Node N_5 a 2D linear approach can be used using all four nodes O_i . The four centre nodes on the edges of the 2D element are handled with a 1D linear approach using the two edge related O_i nodes. This guaranties the continuity of the interpolation result with an interpolation in the related neighbourhood element on the same grid element.

$$(4-8) \quad f(N_1) = 0.5 (f(O_1) + f(O_2))$$

$$(4-9) \quad f(N_2) = 0.5 (f(O_2) + f(O_3))$$

$$(4-10) \quad f(N_3) = 0.5 (f(O_3) + f(O_4))$$

$$(4-11) \quad f(N_4) = 0.5 (f(O_1) + f(O_4))$$

$$(4-12) \quad f(N_5) = 0.25 (f(O_1) + f(O_2) + f(O_3) + f(O_4))$$

4.2.2 Upscaling

Grid coarsening leads to a losing of nodes and related physical state variables values. The remaining nodes do have physical state variables. A simple solution for grid coarseness is to use the physical state variables of the lower quad-tree level approximation directly as physical state variables of the upper quad-tree level. In this case the physical state variables of the lost nodes in the lower quad-tree level are ignored and do not have an impact for the upscaling solution.

This can be improved by integration methods. Key idea is to define integral values as conservation criteria, which have to be the same value on the upper and lower quad-tree grid level.

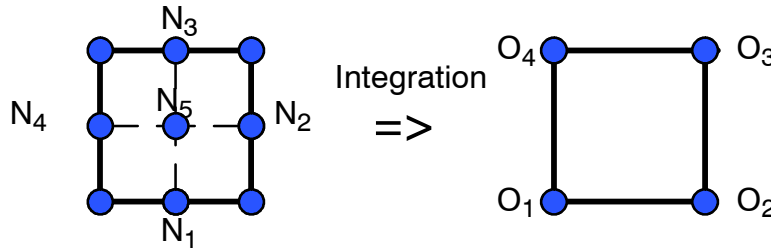


Figure 9: Integration for Coarsening

For each lost node of the lower quad-tree level a correction term for the remaining nodes on the upper level can be derived. Example is the correction term for the inner node N_5 . Basic equation could be the integral over the 2D element of the physical state variable, which should be the same for the lower level and upper level approximation:

$$(4-13) \quad I = \int_{A_e} a(\mathbf{O} \cup \mathbf{N}) dA = \int_{A_e} a(\mathbf{O}) dA$$

As an example this approach will be demonstrated for a 1D element with three nodes:

$$(4-14) \quad I = \int_{L_e} h(\mathbf{O} \cup \mathbf{N}) dx = \int_{L_e} h(\mathbf{O}) dx$$

$$(4-15) \quad \frac{1}{2} (\bar{h}_{o_1} + \bar{h}_{o_2}) \Delta x - \frac{1}{6} (h_{o_1} + 4h_{N_1} + h_{o_2}) \Delta x = 0$$

The length of the 1D element can be eliminated. Both nodes o_1 and o_2 get the same weighting for the correction term due to the elimination node n_1 . This leads to:

$$(4-16) \quad \frac{1}{2} (h_{o_1} + h_{o_2} + 2\Delta h) - \frac{1}{6} (h_{o_1} + 4h_{N_1} + h_{o_2}) = 0$$

$$\bar{h}_{o_1} = h_{o_1} + \Delta h \quad \bar{h}_{o_2} = h_{o_2} + \Delta h$$

The correction term can be determined by:

$$(4-17) \quad \Delta h = -\frac{1}{4} (h_{o_1} + h_{o_2}) + \frac{1}{12} (h_{o_1} + 4h_{N_1} + h_{o_2})$$

$$(4-18) \quad \Delta h = \frac{1}{3} h_{N_1} - \frac{1}{6} (h_{o_1} + h_{o_2})$$

The principle of using integral equations to develop upscaling methods can be applied for any kind of integral to consider conservative variables. Typical examples are mass balance or energy balance integral equations. For groundwater flow the mass balance would be expressed by flux terms using 1st order derivations of the head variable. Energy terms would be proportional to h^2 terms.

Using a simple linear approach the correction terms can be expressed by:

$$(4-19) \quad \Delta f(O_{1-4}) = f(N_5) - 0.25 \sum_{i=1,4} f(O_i)$$

The information of nodes, which are eliminated due to coarseness, is lost for the next time step. Depending on the reason / criteria for the grid adaptation, this can be partial compensated by integration methods.

4.3 Hanging Node

The developed numerical schemes are using a regular space approximation for the discretisation of the 1st and 2nd order derivations. In case of an adaptive grid the grid size in the neighbourhood might be finer or coarser. The numerical scheme has to be adapted to these grid size changes.

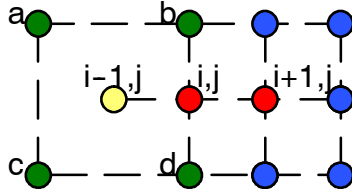


Figure 10: Hanging Node

Figure 10 shows a grid cell with the nodes a, b, c, d on the quad-tree level q. The left hand side cell is one quad-tree level lower q+1. The node i,j is on the border edge of these both cells and member of the node set of quad-tree level q+1 but not of level q. The spatial 2nd derivation in direction 1 in a regular grid is defined at node i,j in general by:

$$(4-20) \quad \frac{\partial^2 h}{\partial x_1^2} = \frac{h_{i-1,j} - 2h_{i,j} + h_{i+1,j}}{\Delta x_1^2}$$

This equation has to be adapted for adaptive grids. Quad-tree based adaptive grids leads to so-called hanging nodes, when the neighbourhood has a higher quad-tree level and related coarser spatial approximation. In the example of Figure 10 node i,j is a hanging node on the level q+1, The right hand side node i-1,j is missing, as the related grid cell is on the level q and the related centre node is not existing. The missing head value $h_{i-1,j}$ at this node has to be expressed by the interpolation values in the related coarser grid element. This leads to an equation with head values on existing nodes.

$$(4-21) \quad \frac{\partial^2 h}{\partial x_1^2} = \frac{\bar{h}_{i-1,j} - 2h_{i,j} + h_{i+1,j}}{\Delta x^2}$$

$$\bar{h}_{i-1,j} = \frac{1}{4} (h_a + h_b + h_c + h_d)$$

In case of a finer grid in one of the both neighbourhood grid elements the related head value at the equidistant node can be used. However, information on a finer grid level are ignored in this way. This information could be added by correction terms if suitable.

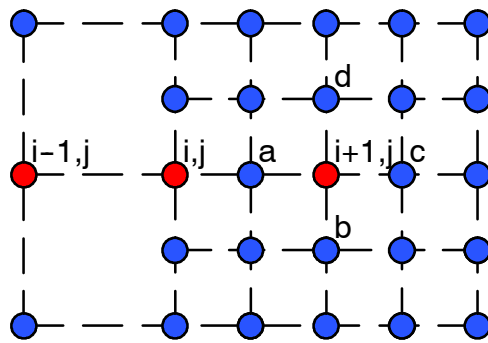


Figure 11: Finer Grid Element

$$(4-22) \quad \frac{\partial^2 h}{\partial x_1^2} = \frac{h_{i-1,j} - 2h_{i,j} + \bar{h}_{i+1,j}}{\Delta x^2}$$

Example for a correction term is the weighted average value of the four nodes on the lower level at the neighbourhood edges. Using the weighting value $g = 0$ is the equation without correction term.

$$(4-23) \quad \bar{h}_{i+1,j} = (1 - g)h_{i+1,j} + g\frac{1}{4}(h_a + h_b + h_c + h_d)$$