GFDM for Simulation of Ground Water Flow

AM Jamaat, AM Behroozi

December 12, 2024

Ground Water Flow: Background & Current Methods

- **Ground Water Flow Equations**
  - Confined Aquifer (Unsteady):

$$S\frac{\partial h}{\partial t} = T(\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2}) + W$$

  - Unconfined Aquifer (Unsteady):

$$S_y\frac{\partial h}{\partial t} = \frac{\partial}{\partial x}(Kh\frac{\partial h}{\partial x}) + \frac{\partial}{\partial y}(Kh\frac{\partial h}{\partial y}) + W$$

- **Traditional Solution Methods**
  - Finite Difference Method (FDM)
  - Finite Element Method (FEM)
- **Limitations of Current FDM**
  - Restricted to regular rectangular domains
  - Poor handling of irregular boundaries
  - Limited flexibility in mesh refinement

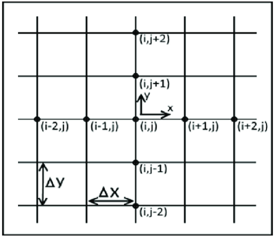
Generalized Finite Difference Method (GFDM)

- **Classical FDM Taylor Expansion (Regular Grid)**
  - For grid points $(i \pm 1, j), (i, j \pm 1)$:

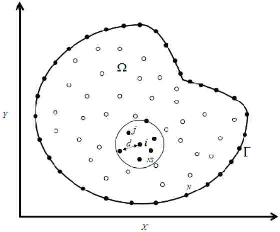
$$f_{i\pm1,j} = f_{i,j} \pm \Delta x f_x + \frac{\Delta x^2}{2!} f_{xx} + O(\Delta x^3)$$

- **GFDM Taylor Expansion**
  - For any point i in support domain:

$$u(x_j, y_j) = u(x_i, y_i) + \frac{\partial u}{\partial x}\Delta x_j + \frac{\partial u}{\partial y}\Delta y_j + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}\Delta x_j^2 + \frac{\partial^2 u}{\partial x \partial y}\Delta x_j \Delta y_j + \frac{1}{2}\frac{\partial^2 u}{\partial y^2}\Delta y_j^2 + O(h^3)$$



Regular Domain



Irregular Domain

GFDM: Step-by-Step Derivative Computation

1. Build M matrix for point i and its n neighbors:

$$\mathbf{M} = \begin{bmatrix} x_1 - x_i & y_1 - y_i & \frac{1}{2}(x_1 - x_i)^2 & (x_1 - x_i)(y_1 - y_i) & \frac{1}{2}(y_1 - y_i)^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n - x_i & y_n - y_i & \frac{1}{2}(x_n - x_i)^2 & (x_n - x_i)(y_n - y_i) & \frac{1}{2}(y_n - y_i)^2 \end{bmatrix}_{n \times 5}$$

2. Choose L vector for desired derivative:

For $\dfrac{\partial}{\partial x} : \mathbf{L} = [1, 0, 0, 0, 0]^T, \quad \dfrac{\partial^2}{\partial x^2} : \mathbf{L} = [0, 0, 2, 0, 0]^T, \quad \nabla^2 : \mathbf{L} = [0, 0, 2, 0, 2]^T$

3. Compute weights using pseudoinverse:

$$\gamma_{1:n} = \mathbf{M}^+ \mathbf{L} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \mathbf{L}$$

4. Add central point weight:

$$\gamma_i = -\sum_{j=1}^{n} \gamma_j$$

5. Approximate derivative:

$$Du_i = \gamma_i u_i + \sum^{n} \gamma_j u_j = \sum^{n} \gamma_j (u_j - u_i)$$

GFDM: Advantages & Applications

- **Properties of GFDM Approximation:**
  - **Consistency:** $\sum_{j=1}^{n} \gamma_j + \gamma_i = 0$
    - Ensures exact derivative for constant functions
    - γ_i is central weight, γ_j are neighbor weights
    - Sum of all weights must be zero
  - **Stability:** $|\gamma_i| > \sum_{j=1}^{n} |\gamma_j|$
    - Central weight must dominate neighbor weights
    - Ensures numerical stability of the method
    - Prevents unbounded growth of numerical errors
  - **Truncation Error:** $\epsilon = O(h^3)$
    - h is characteristic length of stencil
    - Third-order accuracy in spatial discretization
    - Error decreases cubically with stencil size
- **Numerical Benefits**
  - Flexible point distribution
  - Adaptive resolution capability
  - Natural boundary treatment

Implementation for Transient Groundwater Flow PDEs

- **Time integration using Crank-Nicolson ($\theta = \frac{1}{2}$):**
  - Semi-discrete form:

$$\frac{u^{n+1} - u^n}{\Delta t} = \frac{1}{2}[f(u^{n+1}) + f(u^n)]$$

  - With GFDM spatial operators:

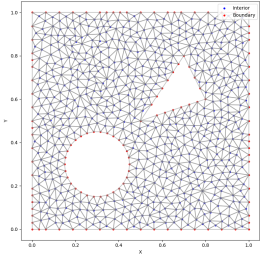
$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2}\left[\sum_{j=1}^{n} \gamma_j (u_j^{n+1} - u_i^{n+1}) + \sum_{j=1}^{n} \gamma_j (u_j^n - u_i^n)\right]$$

  - Rearranged for solution:

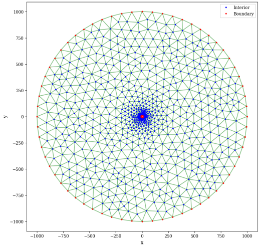
$$(I - \frac{\Delta t}{2}A)u^{n+1} = (I + \frac{\Delta t}{2}A)u^n$$

    where A is the GFDM operator matrix
  - Properties:
    - Second-order accurate in time: $O(\Delta t^2)$
    - Unconditionally stable
    - Conservative

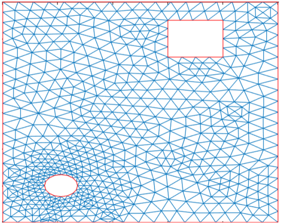
- **Mesh Generation (Discretizing Spatial Domain)**
  - **PyMesh** python package
  - **GMSH** '*.ply2' (An open-source meshing software)
  - Matlab **PDETOOL**
  - It can be extended to any other mesh format with triangulation capability (ArcGIS, QGIS, ...)



PyMesh sample mesh



GMSH sample mesh



PDETOOL sample mesh

Aquifer Parameters:

$T = 100.0 \ \mathrm{m}^2/\mathrm{day}$

$S = 10^{-5}$

$Q = -1000.0 \ \mathrm{m}^3/\mathrm{day}$

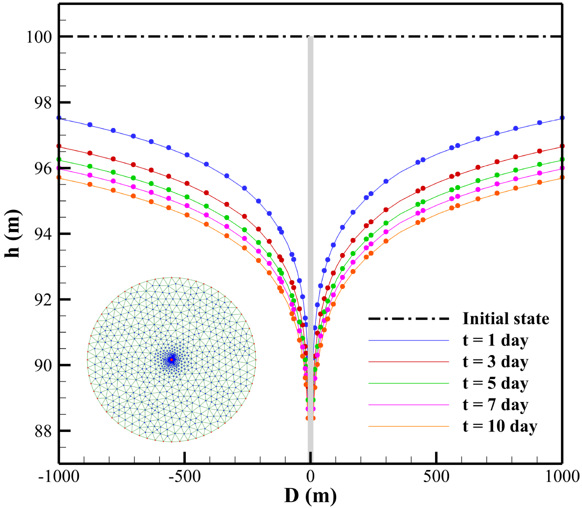
Domain:

$$r_w = 10 \ \mathrm{m}$$
$$R = 1000 \ \mathrm{m}$$

Time Steps:

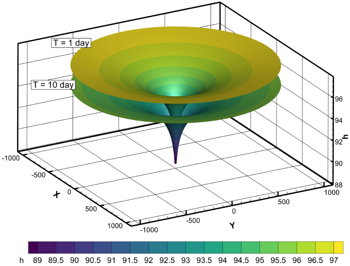
$$\Delta t = 0.1 \ \mathrm{day}$$
$$t_{total} = 10.0 \ \mathrm{days}$$

Validation:
Theis solution

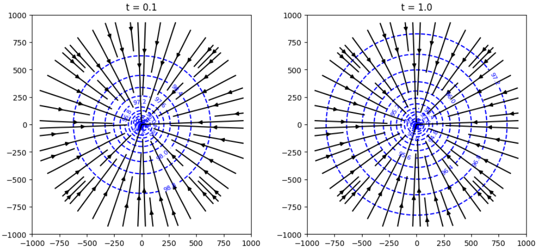


Comparison of numerical and analytical solutions

- Drawdown Evolution and Flow Pattern



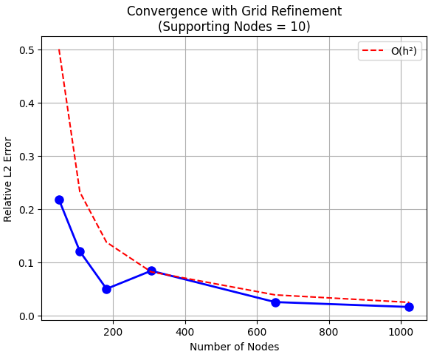
Drawdown surface at
t = 1, 10 days



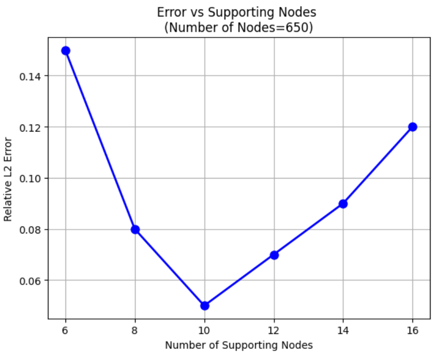
Streamlines at selected times

Solver validation for groundwater flow in confined aquifers

- Mesh Convergence Study



Spatial convergence



Effect of supporting nodes

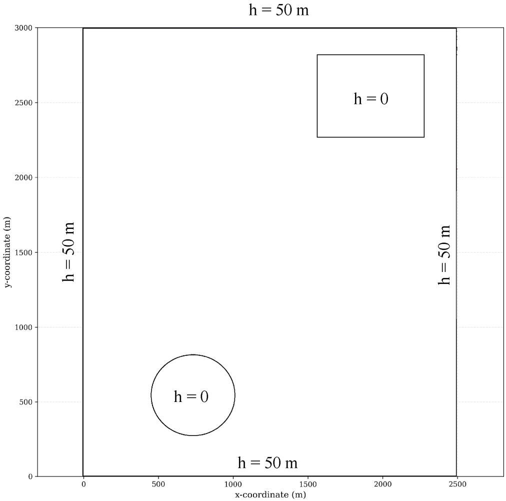
Steady-State Local Drying in Unconfined Aquifer

Problem Statement:

- Unconfined aquifer flow equation:

$$\nabla \cdot (Kh\nabla h) = 0$$

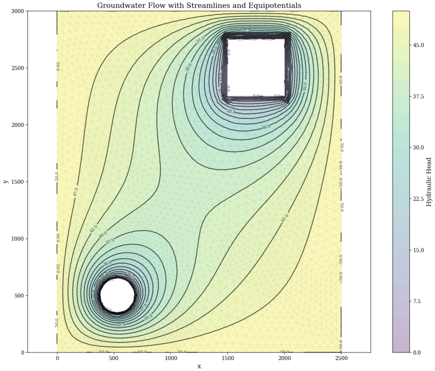
- Boundary conditions:
  - $h = 0$ at interior boundaries
  - $h = 50$ m at outer boundaries
- Study steady-state solution
- Local drying occurs when $h = 0$
- Non-linear due to variable thickness



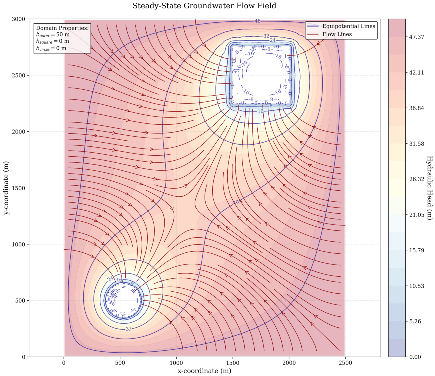
Domain with boundary conditions:
Circle and rectangle: $h = 0$ (dry)
Outer edges: $h = 50$ m

Steady-State Local Drying in Unconfined Aquifer

- Solution: Head Distribution and Flow Pattern



Head distribution and contours (m)



Streamlines

Summary and Conclusions

- **GFDM Implementation:**
  - Successfully validated with Theis solution
  - Optimal performance with 8-10 supporting nodes
- **Method Capabilities:**
  - Handles both confined and unconfined conditions
  - Accurately captures local drying effects
  - Shows good convergence properties
- **Future Work:**
  - Extension to heterogeneous aquifers
  - Implementation of adaptive time-stepping

Thank You for Your Attention

Questions?