Confined aquifer flow

Thomas Mitchell (71506388)

Introduction

The purpose of this report was to study the flow in a confined aquifer of length L (80 m) and width W (40 m) which connected two lakes. The governing equation for 2-D groundwater flow was written in 2D form, Equation 1.

$$\frac{\partial}{\partial x}\left(K\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial y}\left(K\frac{\partial h}{\partial y}\right)=0 \qquad (1)$$

The groundwater flow equation was a function of the hydraulic head $h(x,y)$ (units: m) and the hydraulic conductivity $K(x,y)$ (units: ms^{-1}). The flow velocity was given by Darcy's law, Equations 2a and 2b.

$$q_x=-K\frac{\partial h}{\partial x} \qquad (2a)$$

$$q_y=-K\frac{\partial h}{\partial y} \qquad (2a)$$

The Dirichlet boundary condition was applied to the hydraulic head at the lakes. The other boundaries were impermeable, Equations 3a, 3b, 3c and 3d.

$$h(x=0,y)=18\,m \qquad (3a)$$

$$h(x=L,y)=12\,m \qquad (3b)$$

$$\left.\frac{\partial h}{\partial y}\right|_{y=0}=0 \qquad (3c)$$

$$\left.\frac{\partial h}{\partial y}\right|_{w=W}=0 \qquad (3d)$$

The finite-difference method was used to solve the governing partial differential equation (PDE). The problem was discretised and solved in MATLAB using the Leibmann method. Uniform and non-uniform grid spacings were considered. The efficiency and efficacy of the MATLAB solution was evaluated by performing a mesh convergence study, and by comparing the finite-difference solution to a finite-element solution developed in COMSOL.

Uniform *K* discretisation

The equation was uniformly discretised using finite difference schemes, Equation 4.

$$h(x,y)=-\frac{6}{80}x+18 \qquad (4)$$

The discretised flow equation was solved using the Liebmann method in MATLAB. The algorithm solved the system of linear equations by iteratively improving upon an initial guess. The system had converged when the norm of the relative error fell below a stopping criterion.

Homogeneous aquifer

The iterative convergence of the hydraulic head at the centre of the aquifer is shown in Figure 1. The finite-difference solution calculated in MATLAB converges to the analytical value, thereby verifying the integrity of the numerical solution.

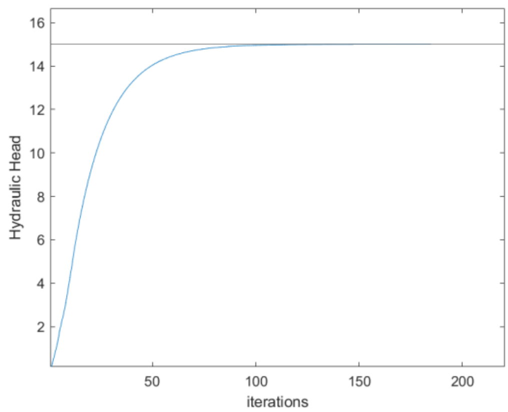

Figure 1: Iterative convergence of the hydraulic head at the centre of the aquifer.

The optimal over-relaxation factor was determined to be 1.76; this value required the least number of iterations over a range of over-relaxation factors.

The flow rates along the top and bottom were zero as there they were physically constrained. The flow rates along the sides were 0.006 m/s; this was consistent with the physical system as there was more water head on one side of the aquifer than the other, which forced the water into the aquifer.

Non-uniform *K* discretisation

The flow equation was discretised with non-uniform spacing. The discretised equation can be found in the Appendix. The non-uniform discretisation system was solved in a similar way to the uniformly discretised system; a set of initial guesses were iteratively improved to compute the hydraulic head.

To evaluate the circular inclusion within the aquifer, each point within the grid was checked to determine whether it was within the circle radius of 5 m; that being the case, the bulk hydraulic conductivity was reduced according to $K(x,y) = \frac{K_b}{10}$.

Aquifer with inclusion

Figure 2 shows the mesh convergence of the MATLAB algorithm using a sinh mesh.

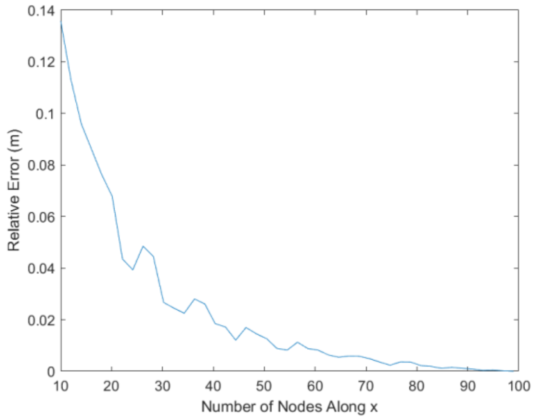

Figure 2: Mesh convergence of the MATLAB algorithm using a sinh mesh.

A tanh function was initially used for meshing: this mesh clustered nodes towards the outside of the domain. It was decided that a mesh that clustered nodes towards the inside of the domain was preferable as this would provide a higher degree of accuracy for the inclusion zone. Therefore, the tanh function was replaced with a sinh function.

The sinh meshed model was compared to the COMSOL model, as shown in Figure 3.

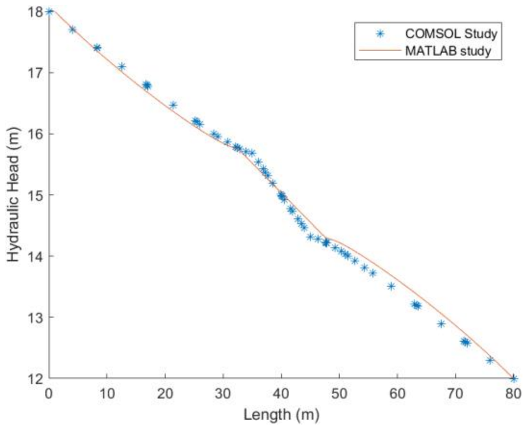

Figure 3: COMSOL study vs MATLAB study along centre plane.

The MATLAB model closely matched with the COMSOL model, however there was a non-negligible margin of error. This was likely because the two models used different solution methods; the MATLAB model employed a finite-difference scheme, whereas the COMSOL model employed a finite-element scheme. The finite-element model was able to capture the solution to a higher degree of accuracy, and therefore was more accurate than the finite-difference model.

The velocity vector field of the system was plotted in MATAB, Figure 4.

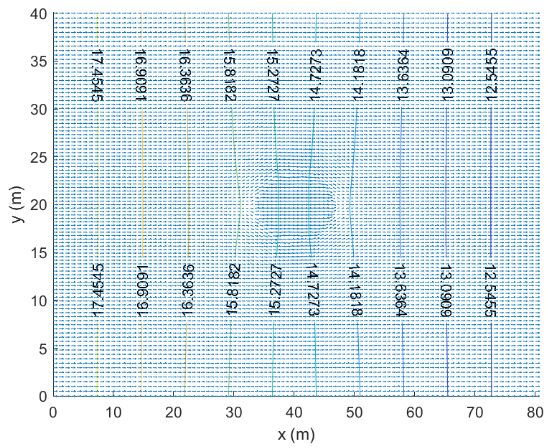

Figure 4: *Velocity vector field of system.*

A high pressure is induced at the top of the inclusion zone, and a low pressure at the bottom. In the middle of the domain the arrows are smaller and closer together. Additionally, the arrows are small on the left and right-hand sides of the inclusion zone; this represents the water particles slowing down to avoid the inclusion.

Conclusion

In summary, the finite-difference method can be used to solve PDEs. The hydraulic head was found using the Leibmann method; the efficiency of this computation could be improved by choosing an appropriate over-relaxation factor. Non-uniform grid spacings can be used to find solutions involving complex geometries, however care should be taken to choose an appropriate meshing strategy to ensure the desired level of accuracy is achieved. Comparing the finite-difference solution with the finite-element solution revealed that the finite-element solution solved the system to a higher degree of accuracy.

Appendix

Discretised flow equation with non-uniform spacing:

$$h_{i,j} = \frac{\frac{2}{\Delta x_{cd}}\left[\frac{k_{i+\frac{1}{2},j}}{\Delta x_a}h_{i+1,j} + \frac{k_{i-\frac{1}{2},j}}{\Delta x_b}h_{i-1,j}\right] + \frac{2}{\Delta y_{cd}}\left[\frac{k_{i,j+\frac{1}{2}}}{\Delta y_a}h_{i,j+1} + \frac{k_{i,j-\frac{1}{2}}}{\Delta y_b}h_{i,j-1}\right]}{\frac{2}{\Delta x_{cd}}\left[\frac{k_{i+\frac{1}{2},j}}{\Delta x_a} + \frac{k_{i-\frac{1}{2},j}}{\Delta x_b}\right] + \frac{2}{\Delta y_{cd}}\left[\frac{k_{i,j+\frac{1}{2}}}{\Delta y_a} + \frac{k_{i,j-\frac{1}{2}}}{\Delta y_b}\right]}$$

$$k_{i+\frac{1}{2},j} = \frac{k_{i+1,j} + k_{i,j}}{2}$$

$$k_{i-\frac{1}{2},j} = \frac{k_{i,j} + k_{i-1,j}}{2}$$

$$k_{i,j+\frac{1}{2}} = \frac{k_{i,j+1} + k_{i,j}}{2}$$

$$k_{i,j-\frac{1}{2}} = \frac{k_{i,j} + k_{i,j-1}}{2}$$

$$\Delta x_{cd} = x_{i+i,j} - x_{i-1,j}$$

$$\Delta y_{cd} = y_{i+i,j} - y_{i-1,j}$$

$$\Delta x_a = x_{i+i,j} - x_{i,j}$$

$$\Delta x_b = x_{i,j} - x_{i-1,j}$$

$$\Delta y_a = x_{i+i,j} - x_{i,j}$$

$$\Delta y_b = x_{i,j} - x_{i-1,j}$$