

Numerics Exercise 3

Anthony Tran

December 2025

1 Question 1

We consider the following Poisson system:

$$-\nabla^2 u = f \quad (1)$$

$$f(x, y) = 2\pi^2 \sin(\pi x) \cos(\pi y) \quad (2)$$

$$u(0, y) = u(1, y) = 0 \quad (3)$$

$$\partial_y u|_{y=0} = \partial_y u|_{y=1} = 0 \quad (4)$$

1.1 i)

Firstly, we will show that the u is indeed a solution to the Poisson system above:

Given

$$u(x, y) = \sin(\pi x) \cos(\pi y) \quad (5)$$

$$-\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = -\left(-\pi^2 \sin(\pi x) \cos(\pi y) - \pi^2 \sin(\pi x) \cos(\pi y)\right) \quad (6)$$

$$-\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = 2\pi^2 \sin(\pi x) \cos(\pi y) \quad (7)$$

Dirichlet BCs:

$$u(0, y) = \sin(0) \cos(\pi y) = 0$$

$$u(1, y) = \sin(\pi) \cos(\pi y) = 0$$

Neumann BCs:

$$\frac{\partial u}{\partial y} = -\pi \sin(\pi x) \sin(\pi y)$$

Hence,

$$-\pi \sin(\pi x) \sin(0) = -\pi \sin(\pi x) \sin(1) = 0$$

Finding the solution to the above Poisson system is equivalent to minimizing:

$$E(u) = \int_{\Omega} \frac{1}{2} |\nabla u|^2 - f u d\Omega \quad (8)$$

Using $u = u + \delta u$, we get:

$$E(u + \lambda \delta u) = \int_{\Omega} \frac{1}{2} |\nabla(u + \lambda \delta u)|^2 - f(u + \lambda \delta u) d\Omega \quad (9)$$

We now expand our equation to simplify the expression:

$$E(u + \lambda \delta u) = \int_{\Omega} \frac{1}{2} \nabla(u + \lambda \delta u) \cdot \nabla(u + \lambda \delta u) - f(u + \lambda \delta u) d\Omega \quad (10)$$

$$E(u + \lambda \delta u) = \int_{\Omega} \frac{1}{2} |\nabla u|^2 + \lambda \nabla u \cdot \nabla \delta u + \frac{\lambda^2}{2} |\nabla \delta u|^2 - f(u + \lambda \delta u) d\Omega \quad (11)$$

Now, we need to minimize our expression with respect to our parameter λ . hence, we need to find:

$$\frac{dE}{d\lambda} = 0 \quad (12)$$

After minimizing the expression and setting $\lambda = 0$, we get:

$$0 = \int_{\Omega} \nabla u \cdot \nabla \delta u - f \delta u d\Omega \quad (13)$$

We can simply rearrange our equation to get:

$$\int_{\Omega} \nabla u \cdot \nabla \delta u = \int_{\Omega} f \delta u d\Omega \quad (14)$$

The condition is that δu is

$$\delta u(0, y) = \delta u(1, y) = 0 \quad (15)$$

Also, it must be integrable on that space. If we take $w = \delta u$, we find the weak form:

$$\int_{\Omega} \nabla u \cdot \nabla w = \int_{\Omega} f w \quad (16)$$

1.1.1 Weak Form

Alternatively, we start of with the strong formulation. Then, we start by multiply our equation by a test function w , to obtain:

$$(-\nabla^2 u)w = fw \quad (17)$$

We now integrate this over the domain Ω :

$$\int_{\Omega} (-\nabla^2 u)w = \int_{\Omega} fw \quad (18)$$

We apply the following vector identity:

$$\nabla \cdot (w \nabla u) = (\nabla^2 u)w + \nabla u \cdot \nabla w \quad (19)$$

$$\int_{\Omega} (\nabla^2 u)w = \int_{\Omega} \nabla \cdot (w \nabla u) - \int_{\Omega} \nabla u \cdot \nabla w \quad (20)$$

Applying the divergence theorem yields:

$$\int_{\Omega} (\nabla^2 u)w = \int_{\partial\Omega} (w \nabla u) \cdot n - \int_{\Omega} \nabla u \cdot \nabla w \quad (21)$$

$$- \int_{\Omega} (\nabla^2 u)w = \int_{\Omega} \nabla u \cdot \nabla w - \int_{\partial\Omega} (w \nabla u) \cdot n \quad (22)$$

We substitute the expression to get:

$$\int_{\Omega} \nabla u \cdot \nabla w - \int_{\partial\Omega} (w \nabla u) \cdot n = \int_{\Omega} fw \quad (23)$$

The Dirichlet and Neumann boundary conditions tell us that the unit normal term must vanish, i.e for Neumann

$$\int_{\partial\Omega} (w\nabla u) \cdot n = \int_{\partial\Omega_1} (w\nabla u) \cdot n + \int_{\partial\Omega_2} (w\nabla u) \cdot n \quad (24)$$

Note that on $y = 1$, which we call the top, $n=(0,1)$. At the bottom, where $y=0$, $n=(0,-1)$. Given the BCs,

$$\int_{\partial\Omega_1} (w\nabla u) \cdot n + \int_{\partial\Omega_2} (w\nabla u) \cdot n = \int_{\partial\Omega_1} w \frac{\partial u}{\partial y} + \int_{\partial\Omega_2} -w \frac{\partial u}{\partial y} = 0 \quad (25)$$

Hence, we find the weak form:

$$\int_{\Omega} \nabla u \cdot \nabla w = \int_{\Omega} f w \quad (26)$$

1.2 ii)

given that:

$$u \sim u_h$$

$$u_h = \sum_{j=1}^{N_h} u_j \phi_j = u_j \phi_j \quad (27)$$

We choose the global basis function such that $w = \phi_i$

Substituting this expression into the weak form we derived earlier, we get the following algebraic system:

$$\int_{\Omega} \nabla \sum_{j=1}^{N_h} u_j \phi_j \cdot \nabla \phi_i = \int_{\Omega} f \phi_i \quad (28)$$

Equivalently, we may rewrite it in the form:

$$\sum_{j=1}^{N_h} u_j \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i = \int_{\Omega} f \phi_i \quad (29)$$

we can rewrite this in the form:

$$Au = b \quad (30)$$

where:

$$A_{ij} = \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i$$

and

$$b_i = \int_{\Omega} f \phi_i$$

We can also do a similar procedure using the variational Ritz method. Again, we start with the following equation:

$$E(u) = \int_{\Omega} \frac{1}{2} |\nabla u|^2 - f u d\Omega \quad (31)$$

We use:

$$u_h = \sum_{j=1}^{N_h} b_j \Gamma_j = b_j \Gamma_j \quad (32)$$

Hence,

$$E(u) = \int_{\Omega} \frac{1}{2} |\nabla \sum_{j=1}^{N_h} b_j \Gamma_j|^2 - f \sum_{j=1}^{N_h} b_j \Gamma_j = b_j \Gamma_j d\Omega \quad (33)$$

We rewrite the expression as:

$$E(u_h) = \frac{1}{2} \sum_{j=1}^{N_h} \sum_{k=1}^{N_h} b_j b_k \int_{\Omega} \nabla \Gamma_j \cdot \nabla \Gamma_k d\Omega - \sum_{j=1}^{N_h} b_j \int_{\Omega} f \Gamma_j d\Omega. \quad (34)$$

As before, we minimize our expression by finding

$$\frac{dE(u_h)}{db_i} = 0 \quad (35)$$

After rearranging, we find:

$$\sum_{j=1}^{N_h} b_j \int_{\Omega} \nabla \Gamma_j \cdot \nabla \Gamma_i d\Omega = \int_{\Omega} f \Gamma_i d\Omega \quad (36)$$

Clearly, using the variational form yields the same result as the weak discrete Galerkin form.

1.3 Question 3

We introduce our reference coordinate system where we map x, y to $\bar{\eta}$:

$$\bar{\eta} = (\eta_1, \eta_2)$$

To map between the reference coordinates and the real coordinates, we define

$$x(\eta_1, \eta_2) = \sum_{\alpha=0}^{N_k-1} x_{\alpha} N_{\alpha} \quad (37)$$

$$y(\eta_1, \eta_2) = \sum_{\alpha=0}^{N_k-1} y_{\alpha} N_{\alpha}(\eta_1, \eta_2) \quad (38)$$

The shape function for our quadrilaterals is given by:

$$N_1 = \frac{1}{4}(1 - \eta_1)(1 - \eta_2)$$

$$N_2 = \frac{1}{4}(1 + \eta_1)(1 - \eta_2)$$

$$N_3 = \frac{1}{4}(1 + \eta_1)(1 + \eta_2)$$

$$N_4 = \frac{1}{4}(1 - \eta_1)(1 + \eta_2)$$

We also need to relate the derivatives in the x, y space to the derivatives in the local coordinate system. We do this through the Jacobian.

$$J = \begin{bmatrix} \frac{\partial x}{\partial \eta_1} & \frac{\partial x}{\partial \eta_2} \\ \frac{\partial y}{\partial \eta_1} & \frac{\partial y}{\partial \eta_2} \end{bmatrix}$$

We then do the gradient transform:

$$\begin{bmatrix} \frac{\partial V}{\partial x} \\ \frac{\partial V}{\partial y} \end{bmatrix} = (J^T)^{-1} \begin{bmatrix} \frac{\partial V}{\partial \eta_1} \\ \frac{\partial V}{\partial \eta_2} \end{bmatrix}. \quad (39)$$

The integral for a quadrilateral element is then computed as:

$$A_{\alpha\beta}^K = \int_K (\nabla N_{\alpha} \cdot \nabla N_{\beta}) \det(J) d\bar{\eta} \quad (40)$$

This may also be written as:

$$A_{\alpha\beta}^K = \int_K \left(\left((J^T)^{-1} \begin{bmatrix} \frac{\partial N_\alpha}{\partial \eta_1} \\ \frac{\partial N_\alpha}{\partial \eta_2} \end{bmatrix} \right) \cdot \left((J^T)^{-1} \begin{bmatrix} \frac{\partial N_\beta}{\partial \eta_1} \\ \frac{\partial N_\beta}{\partial \eta_2} \end{bmatrix} \right) \right) \det(J) d\bar{\eta} \quad (41)$$

These integrals are then computed using Gaussian quadrature rules.
We also have :

$$b_\alpha = \int_K f(x(\eta_1, \eta_2), y(\eta_1, \eta_2)) N_\alpha(\eta_1, \eta_2) |\det(J(e\bar{t}a))| d\bar{\eta} \quad (42)$$

2 Question 4

2.1 p=1 and various grid sizes.

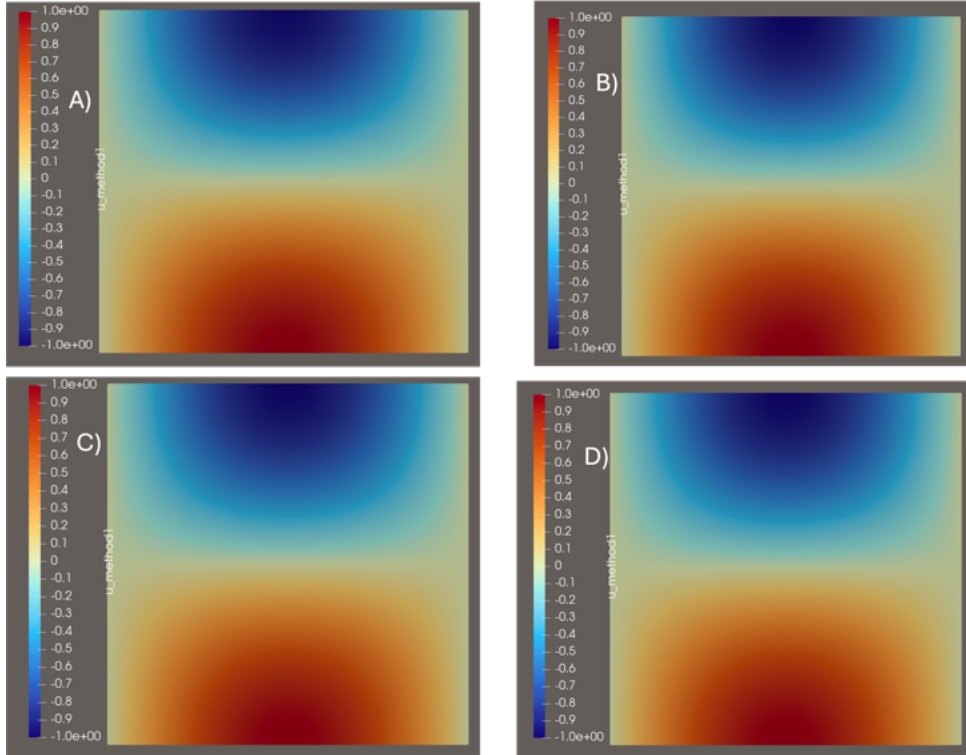


Figure 1: Method 1: Contour plots of velocity for different grid sizes and p=1. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

We start off by considering p=1 and method 1, which is the weak formulation. Figure 2 clearly shows that the error goes down as the number of grid points increases, indicating convergence. Figure 3 shows the L2 norm as a function of the grid size. The slope of -1.99 indicates second order accuracy.

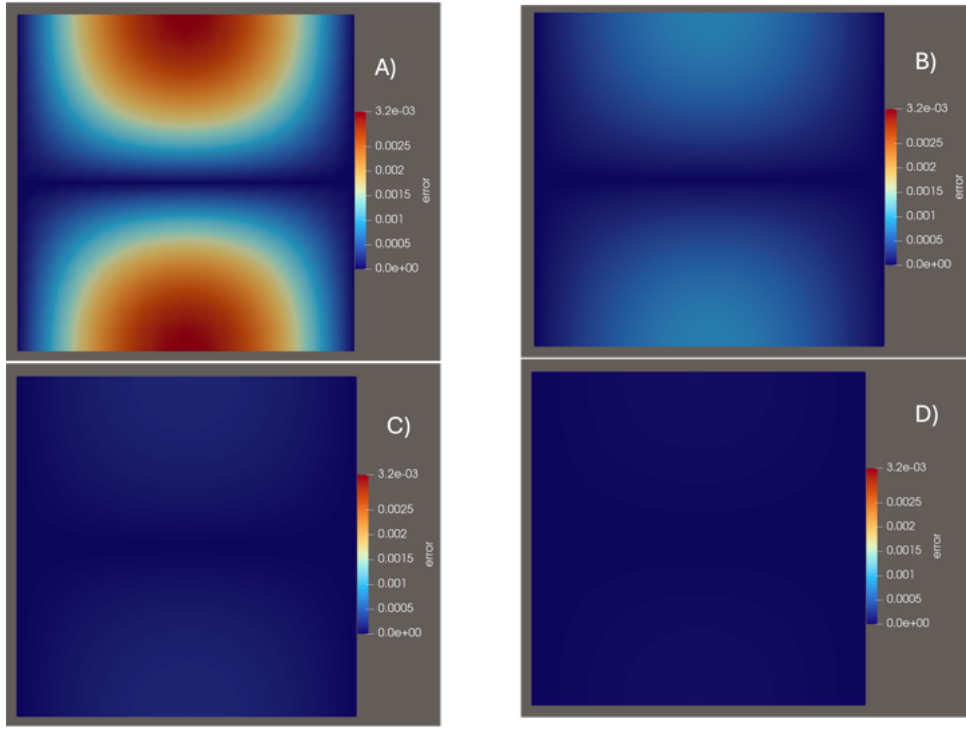


Figure 2: Method 1: Difference between numerical solution and the exact solution for $p=1$. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

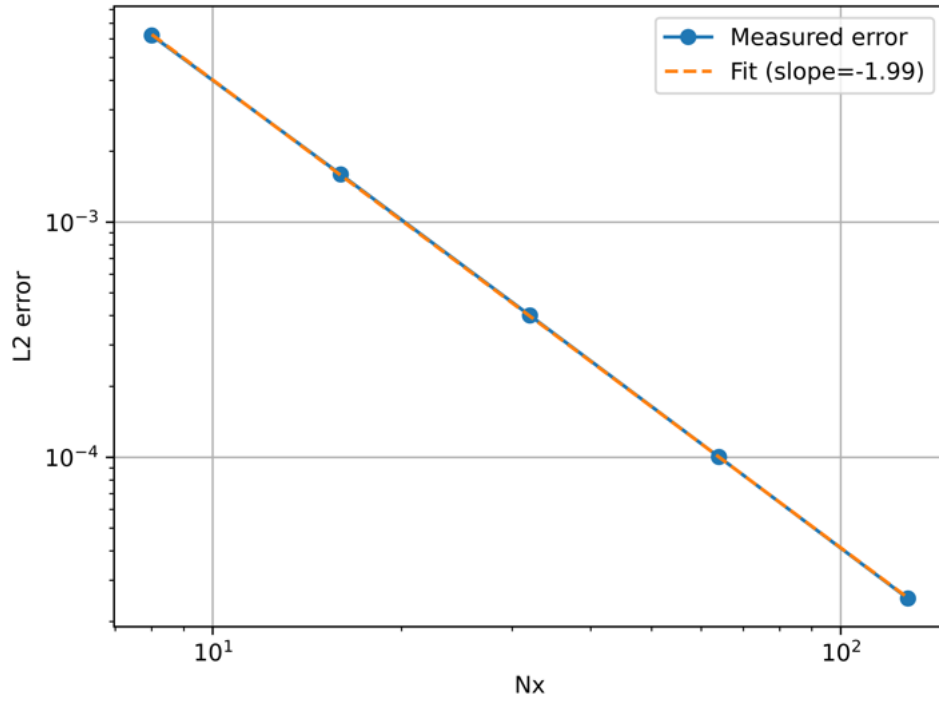


Figure 3: Method 1: L2 norm as a function of grid size for $p=1$.

2.2 p=2 and various grid sizes

Figure 4 below shows the difference between the numerical and exact solutions at different grid sizes. The maximum error decreases from 2.5×10^{-6} for 16x16 to 6.4×10^{-10} for the 128x128 grid. Figure 5 shows the L2 norm as a function of the grid size, along with the line of best fit. The gradient is -4, indicating fourth order convergence in space.

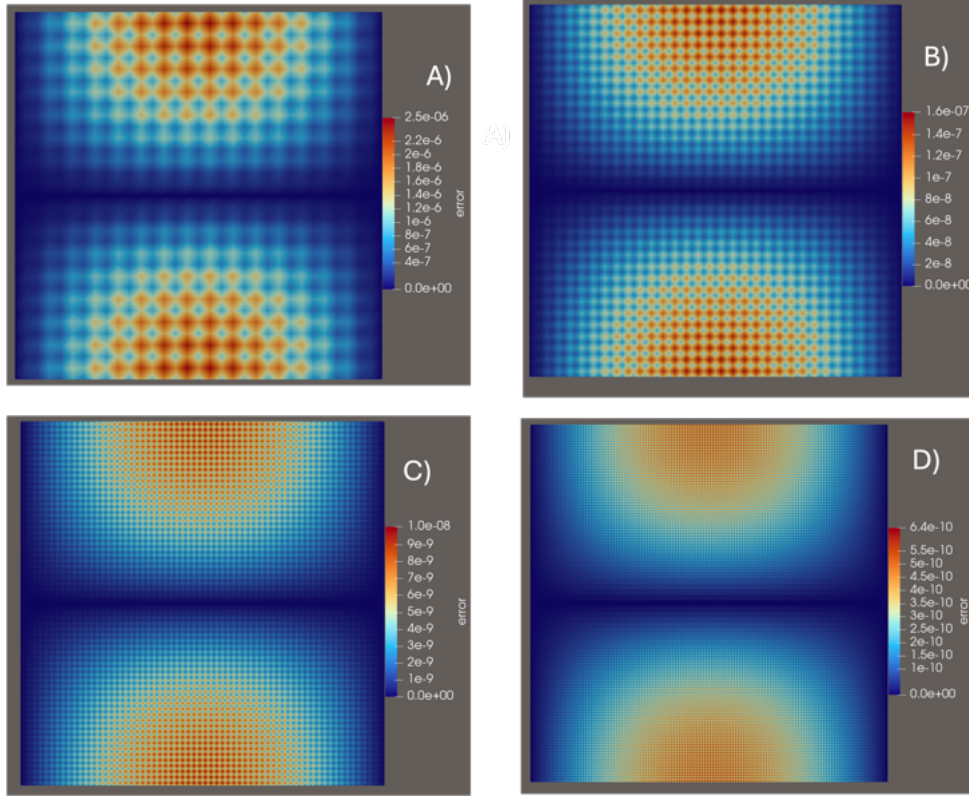


Figure 4: Method 1: Difference between numerical solution and the exact solution for $p=2$. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

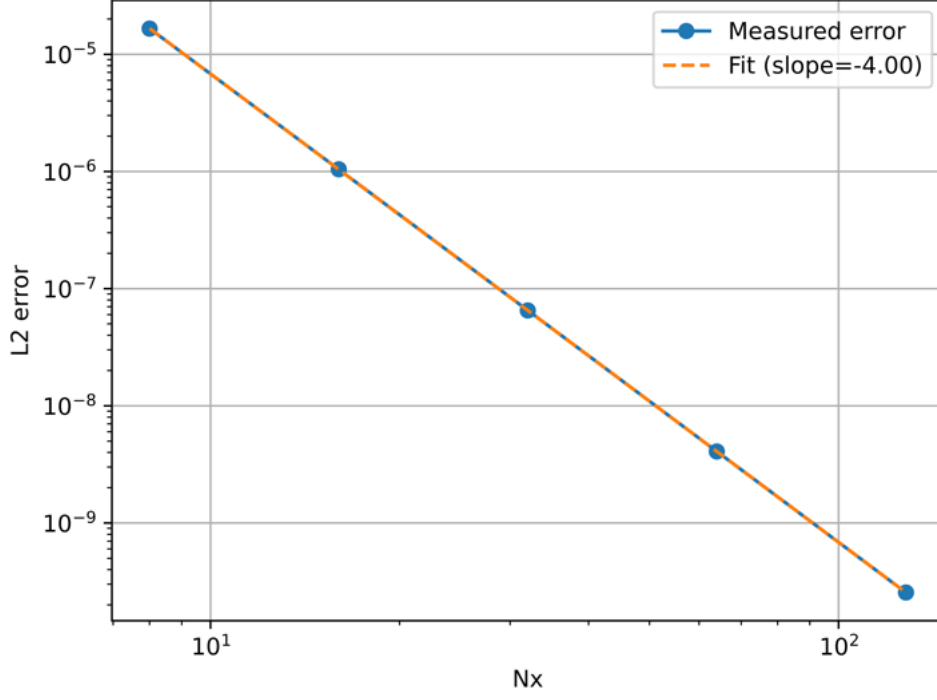


Figure 5: Method 1: L2 norm as a function of grid size for $p=2$.

2.3 $p=3$ and various grid sizes

Again Figure, 6 shows the difference between the numerical and analytical solution. This time, the maximum error starts off at 6.9×10^{-9} for the 16×16 grid and decreases to 1.1×10^{-11} for the 128×128 grid. Strangely, the error does not decrease as predictably as in the other cases, as shown in Figure 7. The gradient of -4.05 suggests fourth order convergence. A possible reason why we do not see higher order could be floating point arithmetic.

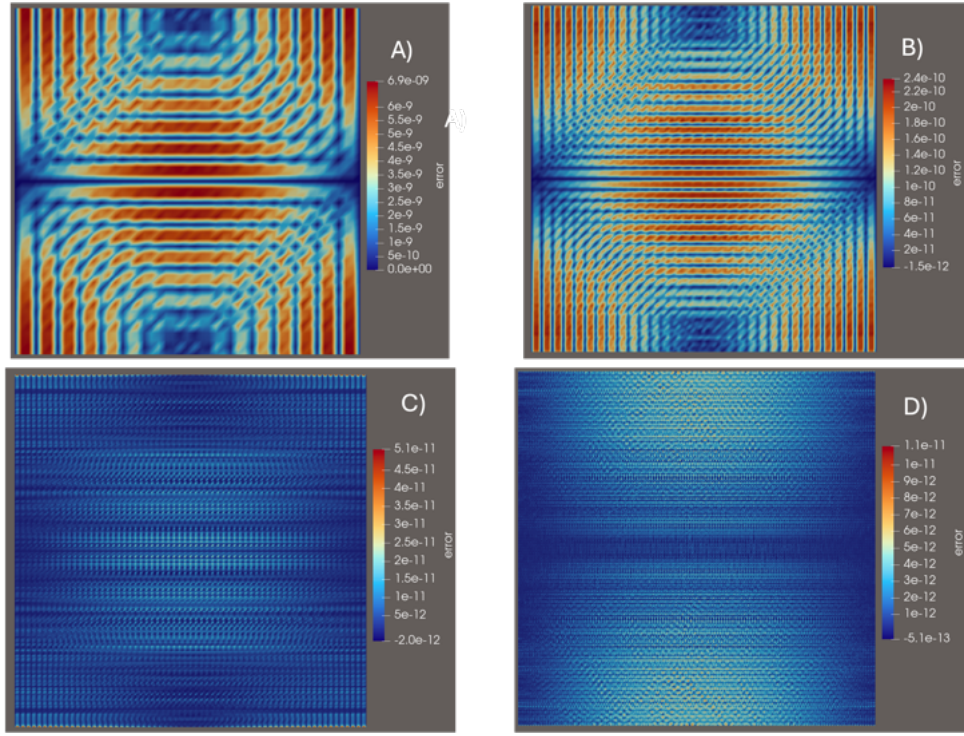


Figure 6: Method 1: Difference between numerical solution and the exact solution for $p=3$. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

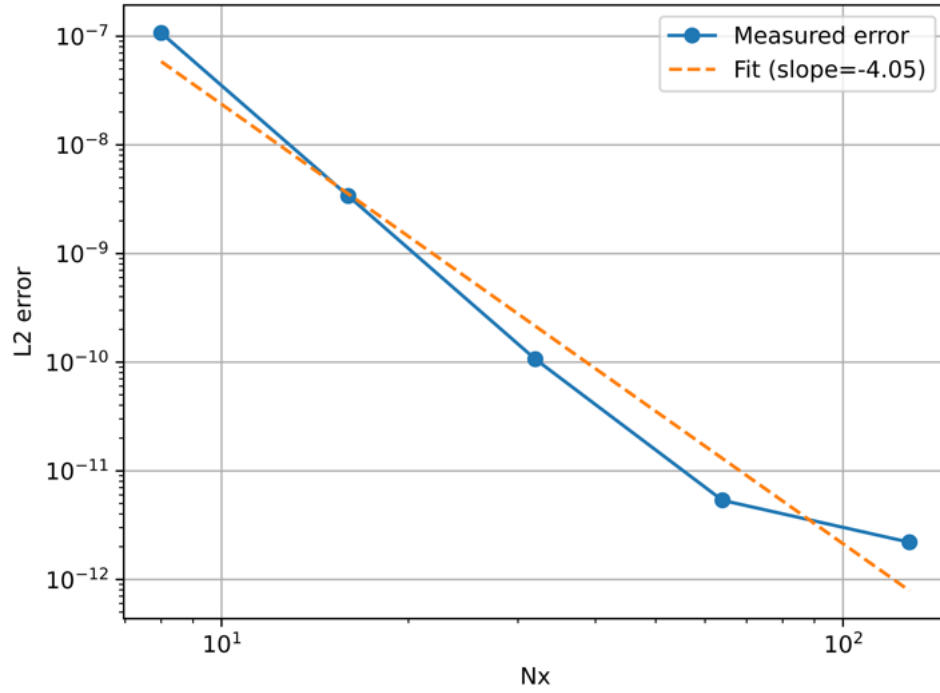


Figure 7: Method 1: L2 norm as a function of grid size for $p=3$.

3 Question 5

:

```
1 from firedrake import *
2 import math
3
4 def solve_poisson_firedrake(nx=16, ny=16, degree=1, CG="CG", save_data=False, quad
   =True, solver_params=None):
5
6     if solver_params is None:
7         solver_params = {'ksp_type': 'cg', 'pc_type': 'none'}
8
9     # Mesh and function space
10    # --- Step 1: Discretize the domain Omega ---
11    # Creates the grid. Quadrilateral=True uses squares instead of triangles.
12
13    mesh = UnitSquareMesh(nx, ny, quadrilateral=quad)
14
15    # This represents  $u_h = \sum(u_j * \phi_j)$ .
16    # 'CG' 1 means Continuous Galerkin (linear polynomials)
17    V = FunctionSpace(mesh, CG, degree)
18
19    x, y = SpatialCoordinate(mesh)
20
21    # Exact solution and forcing
22    u_exact = sin(pi * x) * cos(pi * y)
23    f = Function(V).interpolate(2 * pi**2 * u_exact)
24
25    # Boundary conditions (x = 0 and x = 1)
26    bc_x0 = DirichletBC(V, Constant(0.0), 1)
27    bc_x1 = DirichletBC(V, Constant(0.0), 2)
28    bcs = [bc_x0, bc_x1]
29
30    # -----
31    # Method 1: Manual weak form
32    # -----
33    # u is the TrialFunction (the unknown solution  $u_h$ )
34    # v is the TestFunction (the weight function  $w$  or  $\phi_i$ )
35    u = TrialFunction(V)
36    v = TestFunction(V)
37    #  $a(u, v) = \text{integral}(\text{grad}(u) \cdot \text{grad}(v) \, dx)$  -> Corresponds to Matrix  $A_{ij}$ 
38    a = inner(grad(u), grad(v)) * dx
39    #  $L(v) = \text{integral}(f * v \, dx)$  -> Corresponds to Vector  $b_i$ 
40
41    L = f * v * dx
42
43    u_1 = Function(V, name="u_method1")
44    # Solves the linear system  $Au = b$ 
45    solve(a == L, u_1, bcs=bcs, solver_parameters=solver_params)
46
47    # -----
48    # Method 2: Variational (Ritz Galerkin) formulation
49    # -----
50    u_2 = Function(V, name="u_method2")
51
52    # The Energy Functional  $E(u) = \text{integral}(0.5 * |\text{grad}(u)|^2 - f * u)$ 
53    J = (0.5 * inner(grad(u_2), grad(u_2)) - u_2 * f) * dx
54    # F is the derivative (variation) of the energy.
55    # Solving  $F == 0$  is equivalent to finding the minimum  $dE/du = 0$ .
56    F = derivative(J, u_2, v)
57
58    # Nonlinear solver (Newton) used here to find the stationary point of the
    functional
```

```

59 solve(F == 0, u_2, bcs=bcs)
60
61 # -----
62 # Error analysis
63 # -----
64 u_exact_fun = Function(V).interpolate(u_exact)
65
66 # Calculate L2 Norm of the error: sqrt( integral( (u_approx - u_exact)^2 ) )
67 L2_1 = sqrt(assemble((u_1 - u_exact_fun)**2 * dx))
68 L2_2 = sqrt(assemble((u_2 - u_exact_fun)**2 * dx))
69 L2_diff = sqrt(assemble((u_2 - u_1)**2 * dx))
70
71 if save_data:
72     outfile = VTKFile(f'output_{nx}_{degree}.pvd')
73     outfile.write(u_1, u_2)
74
75 return {
76     "mesh_size": 1.0 / nx,
77     "u_method1": u_1,
78     "u_method2": u_2,
79     "L2_error_method1": L2_1,
80     "L2_error_method2": L2_2,
81     "L2_difference": L2_diff,
82 }

```

Listing 1: Firedrake Poisson Solver

4 Question 6

We use the following u for our manufactured solution:

$$u = \exp(-100(x - 0.5)^2 + (y - 0.5)^2) \quad (43)$$

with

$$u = u_{exact} \quad \text{on} \quad \partial\Omega$$

Figure 8 shows the difference between the exact solution and the numerical solution, while Figure 9 shows 4th order convergence with $p=2$.

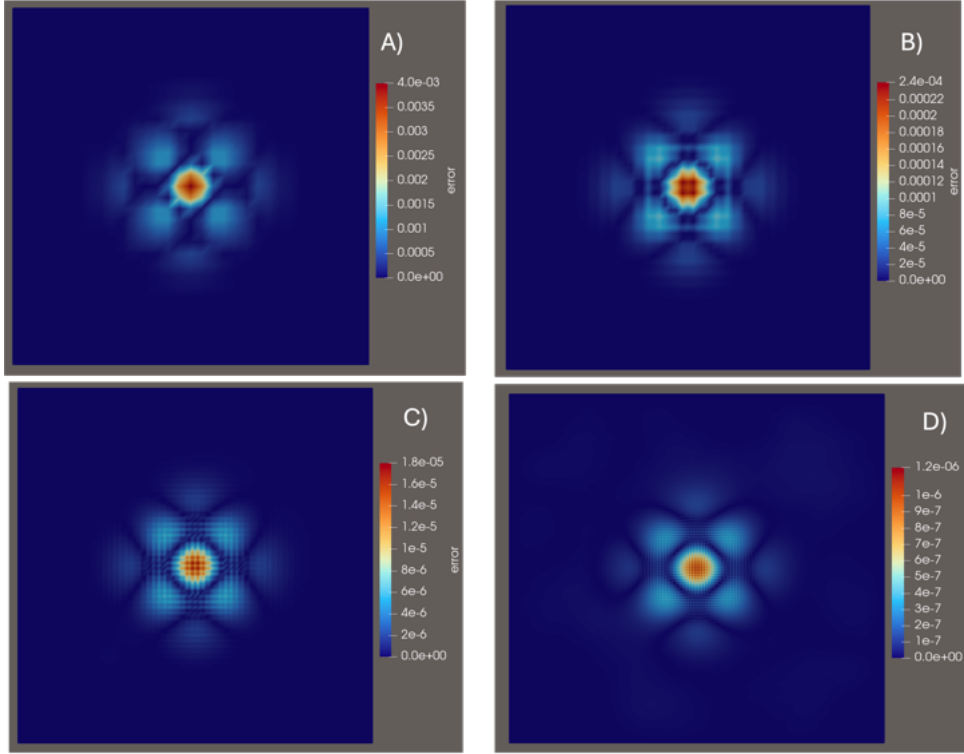


Figure 8: Method 1: Difference between numerical solution and the new exact solution for $p=2$. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

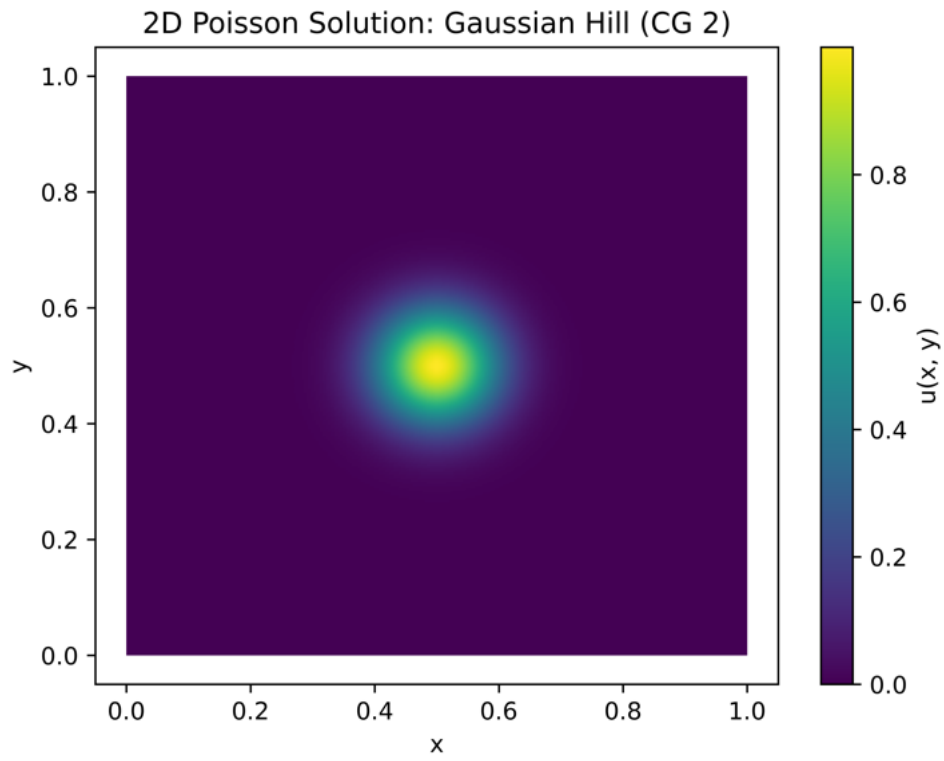


Figure 9: Constructed Analytical solution

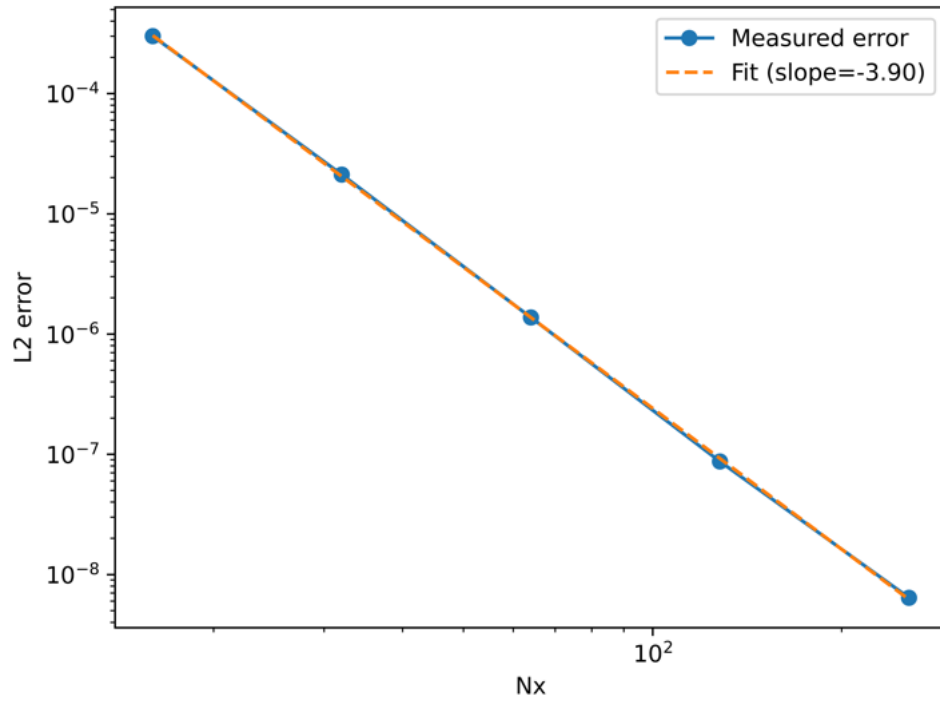


Figure 10: L2 error as a function of the grid size for the new solution.

5 Groundwater

We are given the equations:

$$\partial_t(w_v h_m) - \alpha g \partial_y(w_v h_m \partial_y h_m) = \frac{w_v R}{m_{por} \sigma_e} \quad (11)$$

$$\partial_y h_m = 0 \quad \text{on } y = L_y \quad (12)$$

$$h_m(0, t) = h_{cm}(t) \quad \text{at } y = 0 \quad (13)$$

$$L_c w_v \frac{dh_{cm}}{dt} = w_v m_{por} \frac{\sigma_e}{2} \alpha g \partial_y(h_m^2)|_{y=0} - w_v \sqrt{g} \max(\frac{2}{3} h_{cm}(t), 0)^{3/2} \quad (14)$$

5.1 Question 1)

We multiply equation 11 by a test function, which we call ϕ and integrate over the domain

$$\int_0^{L_y} \partial_t(w_v h_m) \phi - \alpha g \partial_y(w_v h_m \partial_y h_m) \phi \, dy = \int_0^{L_y} \frac{w_v R \phi}{m_{por} \sigma_e} \, dy \quad (44)$$

We integrate by parts:

$$\int_0^{L_y} \alpha g \partial_y(w_v h_m \partial_y h_m) \phi \, dy = \alpha g w_v h_m \partial_y h_m \phi|_{y=L_y} - \int_0^{L_y} \alpha g w_v h_m \partial_y h_m \partial_y \phi \, dy \quad (45)$$

Given that:

$$\alpha g w_v h_m \partial_y h_m \phi|_{y=L_y} = 0$$

$$\int_0^{L_y} \alpha g \partial_y(w_v h_m \partial_y h_m) \phi \, dy = -\alpha g w_v h_m \partial_y h_m \phi|_{y=0} - \int_0^{L_y} \alpha g w_v h_m \partial_y h_m \partial_y \phi \, dy \quad (46)$$

We also notice that:

$$\alpha g w_v h_m \partial_y h_m \phi|_{y=0} = \alpha g w_v \frac{1}{2} \partial_y h_m^2 \phi|_{y=0}$$

Rearranging equation (14) gives:

$$\frac{1}{\sigma_e m_{por}} L_c w_v \frac{dh_{cm}}{dt} + \frac{1}{\sigma_e m_{por}} w_v \sqrt{g} \max(\frac{2}{3} h_{cm}(t), 0)^{3/2} = w_v \frac{1}{2} \alpha g \partial_y(h_m^2)|_{y=0} \quad (47)$$

substituting this back into our IBP equation gives:

$$\int_0^{L_y} \alpha g \partial_y(w_v h_m \partial_y h_m) \phi \, dy = -\frac{\phi}{\sigma_e m_{por}} L_c w_v \frac{dh_{cm}}{dt} - \frac{\phi}{\sigma_e m_{por}} w_v \sqrt{g} \max(\frac{2}{3} h_{cm}(t), 0)^{3/2} - \int_0^{L_y} \alpha g w_v h_m \partial_y h_m \partial_y \phi \, dy \quad (48)$$

Eliminating w_v and substituting the expression back into the original equation gives:

$$\int_0^{L_y} \partial_t(w_v h_m) \phi \, dy + \frac{\phi}{\sigma_e m_{por}} L_c w_v \frac{dh_{cm}}{dt} + \frac{\phi}{\sigma_e m_{por}} w_v \sqrt{g} \max(\frac{2}{3} h_{cm}(t), 0)^{3/2} + \int_0^{L_y} \alpha g w_v h_m \partial_y h_m \partial_y \phi \, dy = \int_0^{L_y} \frac{w_v R \phi}{m_{por} \sigma_e} \, dy \quad (49)$$

Rearranging the equation gives:

$$\int_0^{L_y} \partial_t(h_m) \phi \, dy + \frac{\phi(0)}{\sigma_e m_{por}} L_c \frac{dh_{cm}}{dt} = \int_0^{L_y} \frac{R \phi}{m_{por} \sigma_e} \, dy - \int_0^{L_y} \alpha g h_m \partial_y h_m \partial_y \phi \, dy - \frac{\phi(0)}{\sigma_e m_{por}} \sqrt{g} \max(\frac{2}{3} h_{cm}(t), 0)^{3/2} \quad (50)$$

Yes, we should take $h_m(0, t) = h_{cm}(t)$ to keep the adjoint structure in tact.

We can now discretize our equation. First, we make the following approximation:

$$h_m = \sum_{j=1}^N h_j \phi_j \quad (51)$$

Hence, the left hand side becomes:

$$\int_0^{L_y} \partial_t \left(\sum_{j=1}^N h_j \phi_j \right) \phi_i dy + \frac{\phi(0)}{\sigma_e m_{por}} L_c \frac{dh_{cm}}{dt} \quad (52)$$

We can rewrite this as:

$$\sum_{j=1}^N \partial_t h_j \int_0^{L_y} \phi_j \phi_i dy + \frac{\phi(0)}{\sigma_e m_{por}} L_c \frac{dh_{cm}}{dt} \quad (53)$$

Remembering that $\int_0^{L_y} \phi_j \phi_i dy = M_{ij}$ and using forward Euler, we find:

$$\frac{M_{ij} h_j^{n+1} - M_{ij} h_j^n}{\Delta t} + \frac{L_c}{\sigma_e m_{por}} \frac{h_1^{n+1} - h_1^n}{\Delta t} \delta_{i1} \quad (54)$$

The right hand side of the equation becomes:

$$b_i^n = \int_0^{L_y} \frac{R^n \phi_i}{m_{por} \sigma_e} dy - \int_0^{L_y} \alpha g h_m^n \partial_y h_m^n \partial_y \phi_i dy \quad (55)$$

We can also substitute our expression for h_m to get:

$$b_i^n = \int_0^{L_y} \frac{R^n \phi_i}{m_{por} \sigma_e} dy - \int_0^{L_y} \alpha g \left(\sum_{j=1}^N h_j^n \phi_j \right) \left(\sum_{k=1}^N h_k^n \partial_y \phi_k \right) \partial_y \phi_i dy \quad (56)$$

The update equation is given by:

$$\frac{M_{ij} h_j^{n+1} - M_{ij} h_j^n}{\Delta t} + \frac{L_c}{\sigma_e m_{por}} \frac{h_1^{n+1} - h_1^n}{\Delta t} \delta_{i1} = b_i^n - \frac{1}{\sigma_e m_{por}} \sqrt{g} \max\left(\frac{2}{3} h_1^n(t), 0\right)^{3/2} \delta_{i1} \quad (57)$$

$$M_{ij} h_j^{n+1} + \frac{L_c}{\sigma_e m_{por}} h_1^{n+1} \delta_{i1} = M_{ij} h_j^n + \frac{L_c}{\sigma_e m_{por}} h_1^n \delta_{i1} + \Delta t b_i^n - \frac{\Delta t}{\sigma_e m_{por}} \sqrt{g} \max\left(\frac{2}{3} h_1^n(t), 0\right)^{3/2} \delta_{i1} \quad (58)$$

Since this is a diffusion like problem, our time step restriction is

$$\Delta t < \frac{\Delta y^2}{2\alpha g h_{max}} \quad (59)$$

An advective time step would give:

$$\Delta t < \frac{\Delta y}{\alpha g h_{max}} \quad (60)$$

5.2 Question 2)

5.2.1 Constant $h_{cm}=0.07$

When h_{cm} is constant, we see that the water head first drops to lower levels (e.g 10 s) and then steadily increases until it reaches a steady state (see Figure 11.), which is reached by 100 s. To confirm that a steady state is reached, the simulation was run for 200 s, and the profile at 100 s was plotted together with the profile at 200 s (Figure 12), which confirms that a steady state was reached by 100s. The simulation was also run for $N_x=40$, and the results indicate convergence, as shown in Figure 13.

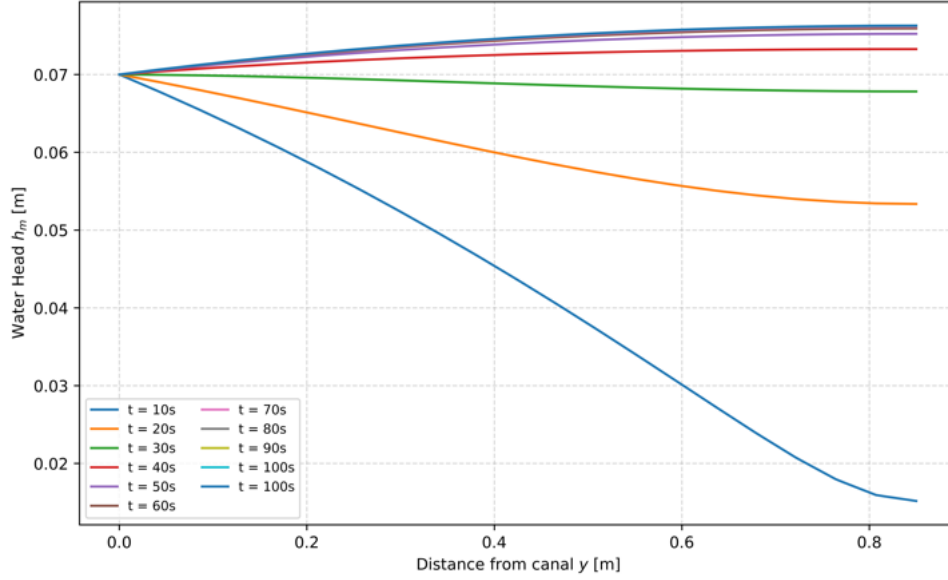


Figure 11: Water head at different times where $h_{cm}=0.07$ is fixed.

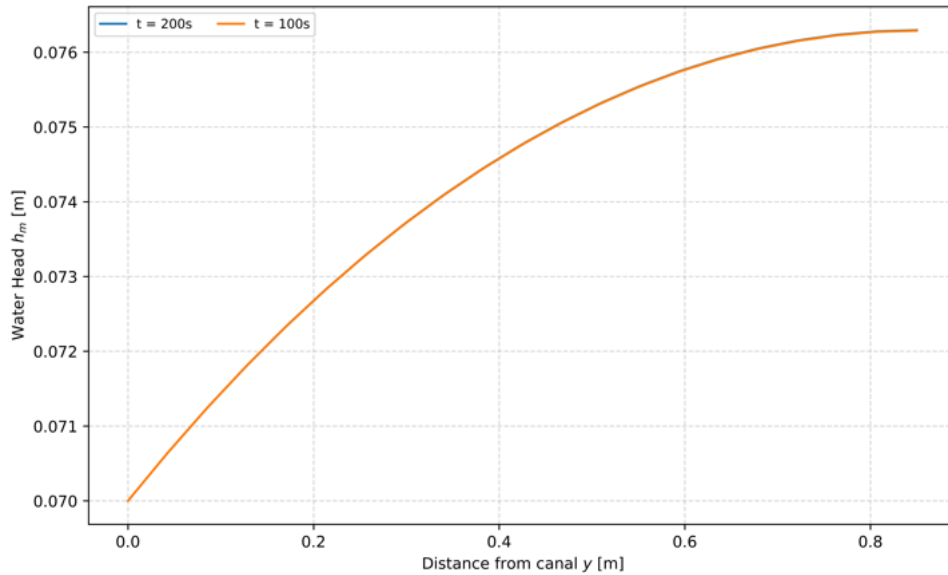


Figure 12: Water head plotted at $t=100$ and $t=200$ for $h_{cm} = 0$. Steady state is reached by 100 seconds.

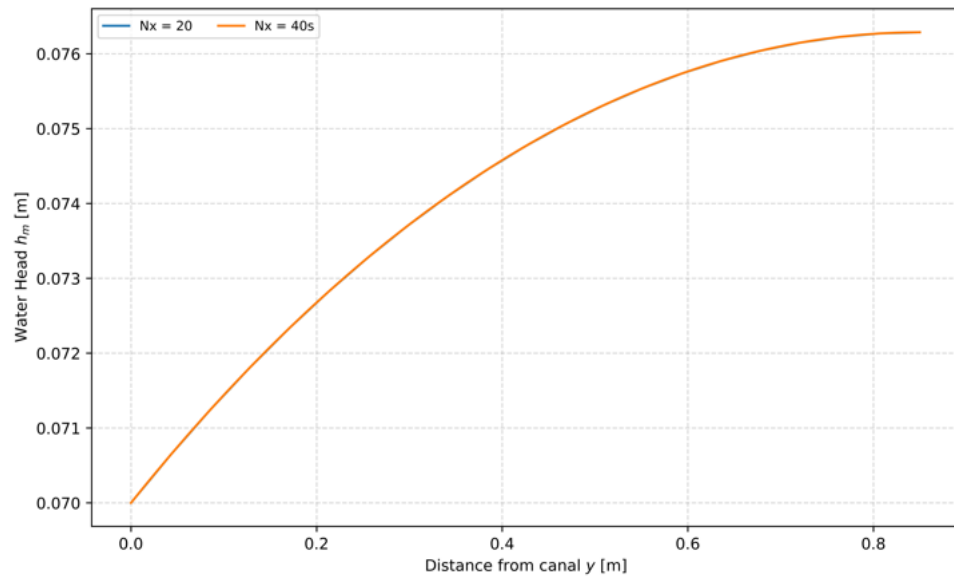


Figure 13: Convergence of the solution. The water head profile at $t=100$ for $Nx=20$ and $Nx=40$.

5.2.2 Case where $h_{cm}=0$

Simulations where h_{cm} is not fixed were also run. Figure 14 shows that the water head profile still develops after 200 s. In Figure 15, which shows the water head profile for 100 s, 190 s and 200 s, it can be seen that a steady state is reached by 200 s. The water head profile at $t=200$ s was plotted for $N_x=20$, $N_x=40$ and $N_x=60$, which can be seen in Figure 16. The profile far away from the canal does not change, but it changes very close to the canal. It appears convergence is reached in this area when $N_x=40$. The h_{cm} profile can be seen in Figure 17, and the steady state value appears to be around 0.0016 m. Note that the rainfall pattern is not shown since it is constant at 0.00125.

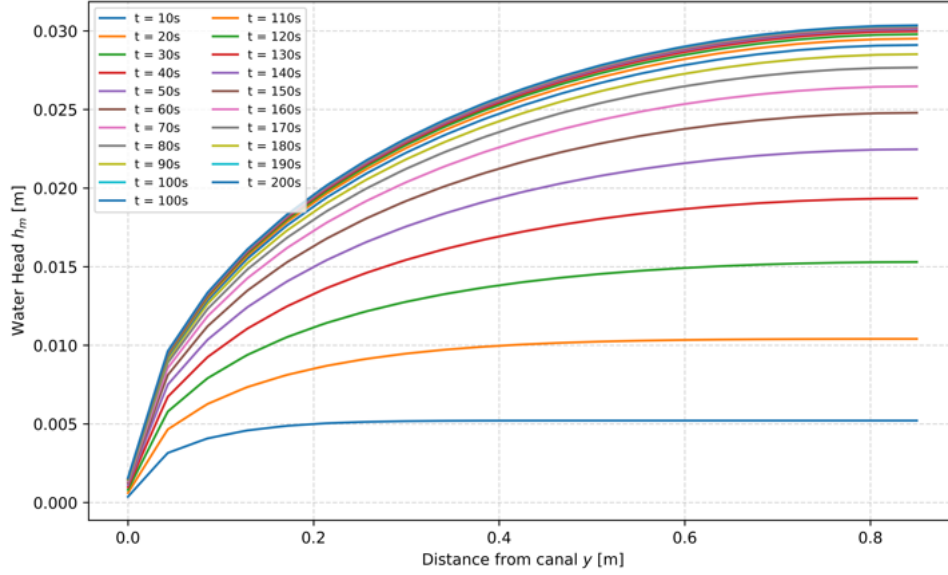


Figure 14: Water head at different times for $h_{cm}=0.0$ and where h_{cm} is not fixed

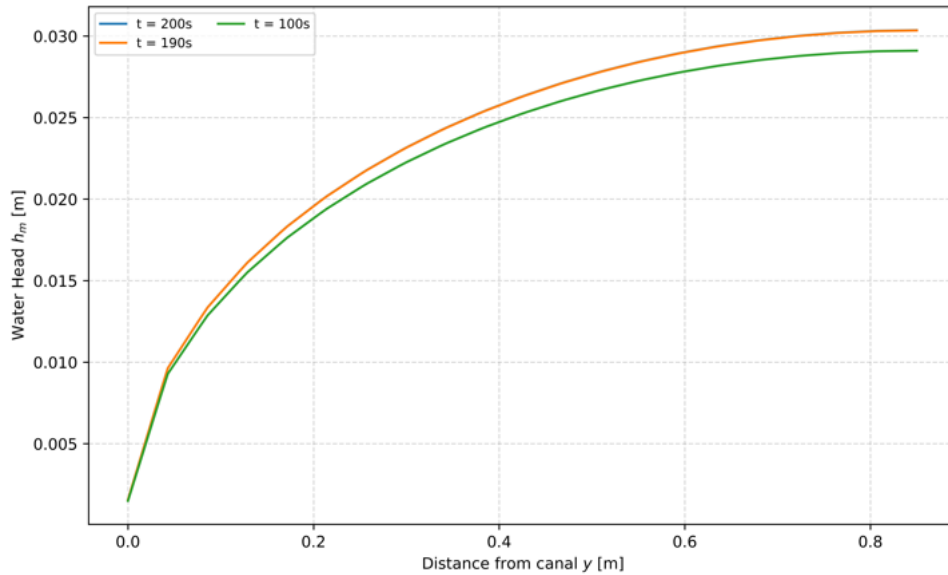


Figure 15: Water head plotted at $t=100$, $t=200$ and $t=190$ s. Steady state is reached by 200s.

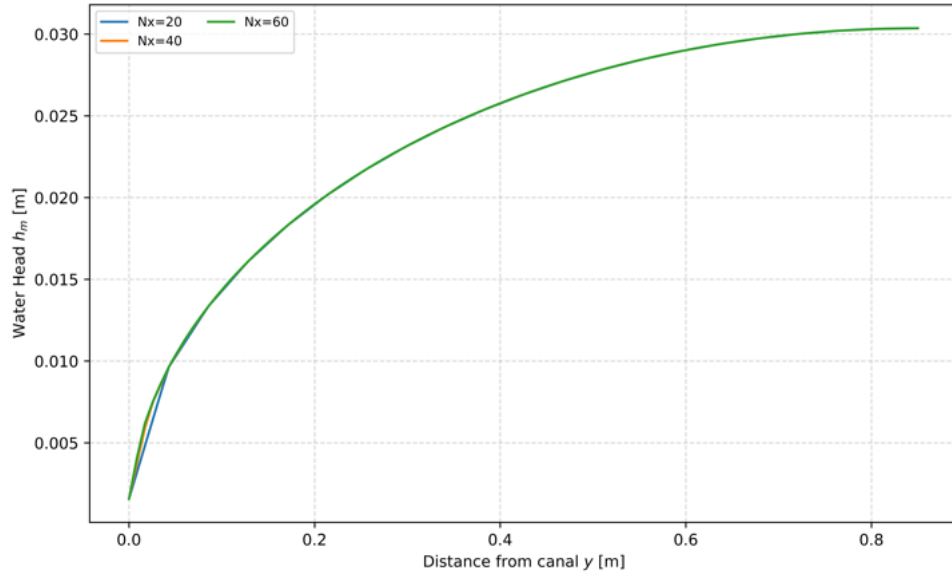


Figure 16: Convergence of the solution. The water head profile at $t=200$ for $N_x=20$, $N_x=40$ and $N_x=60$.

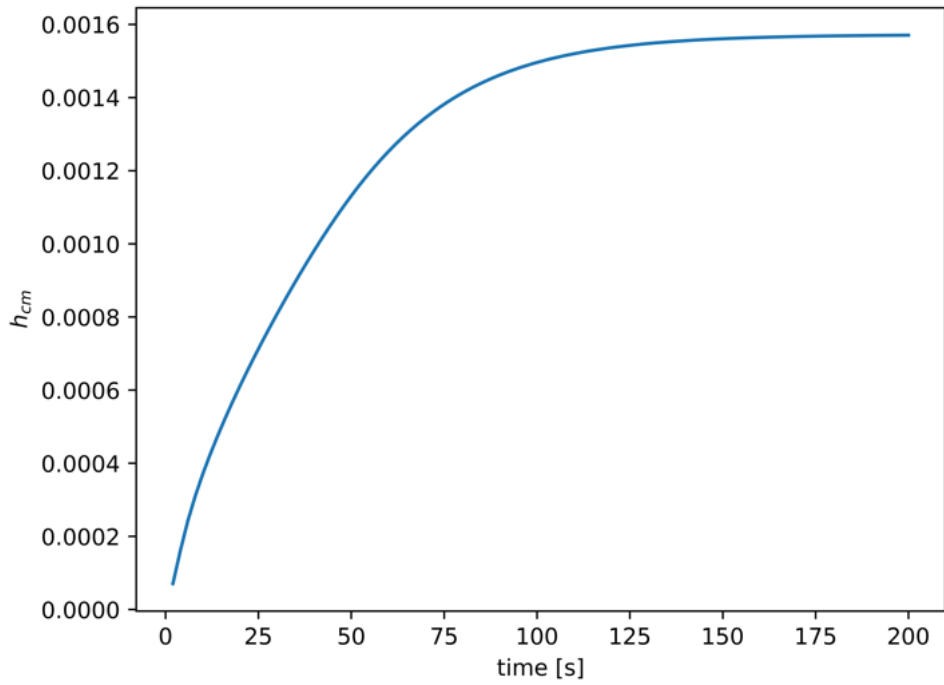


Figure 17: h_{cm} plotted over time. Steady state value is roughly 0.0016.

Variable rainfall was also tested. Every 10-second window, the rain fell only for a fraction of that time, i.e 1s rain and 9s off, 2s rain and 8s off etc.. 1,2,4 and 9 seconds were tested. The water head profiles at $t=200$ s for the different rainfall durations are shown in Figure 18. Clearly, as the rainfall duration increases, the water head profile also increases. The rainfall pattern is also visualized in Figure 19. The h_{cm} profile is shown in Figure 20. As the duration of the rainfall increases, the shape of the graph approaches the one where rainfall is constant (Figure 15). For very short rainfall durations (i.e 1 s or 2 s), there are big spikes due to rainfall and subsequent drainage. In these cases, the system reaches a dynamic steady state rather than a static one.

5.2.3 Variable Rainfall

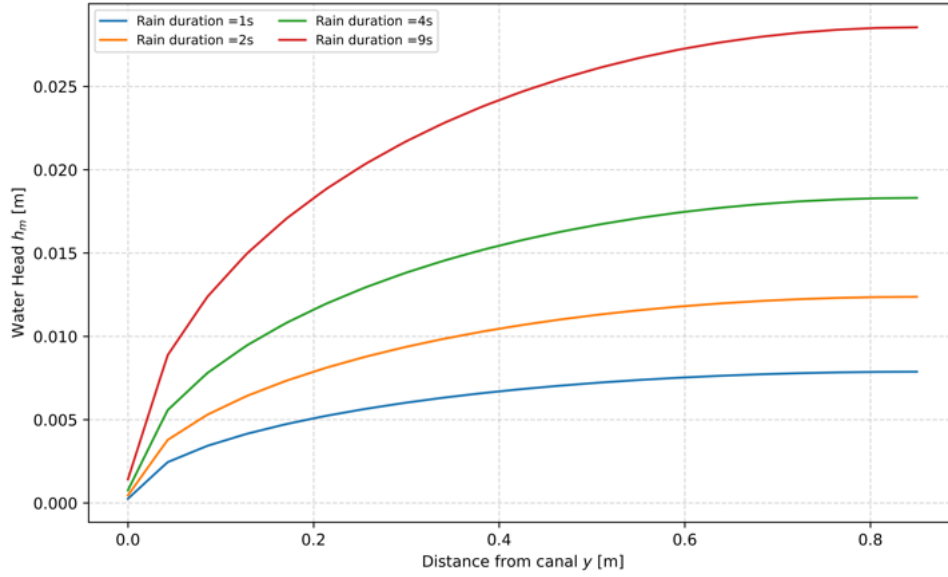


Figure 18: Water head plotted at $t=200$ for rain durations 1s,2s,4s and 9s.

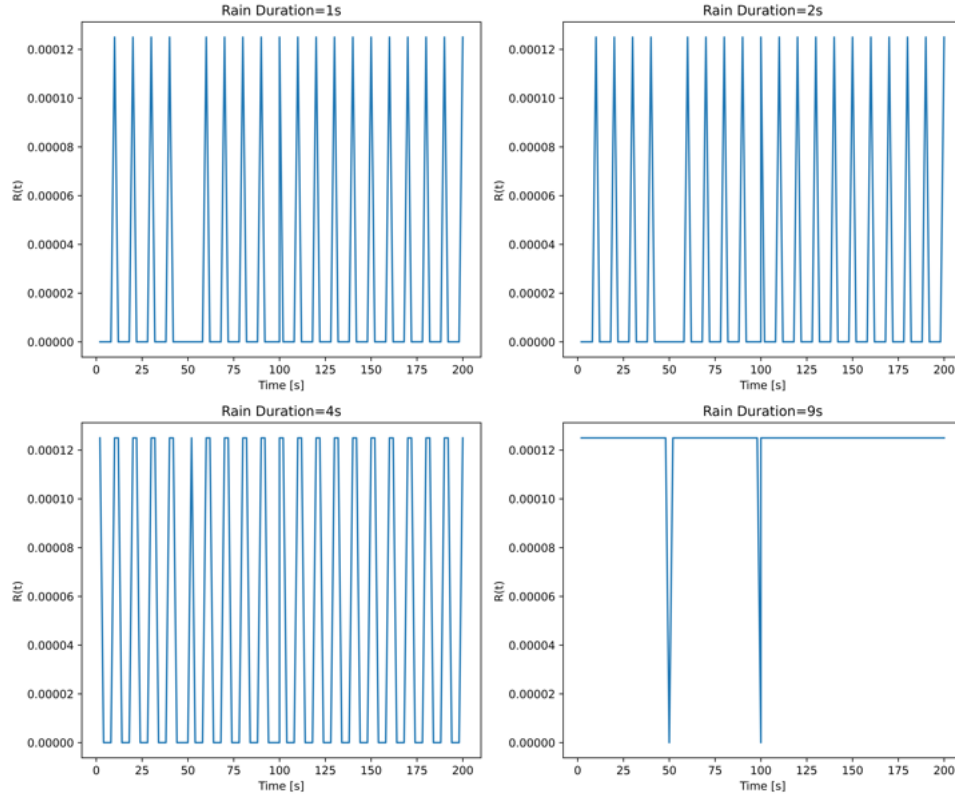


Figure 19: Rainfall plotted over time for rain durations 1s,2s,4s and 9s.

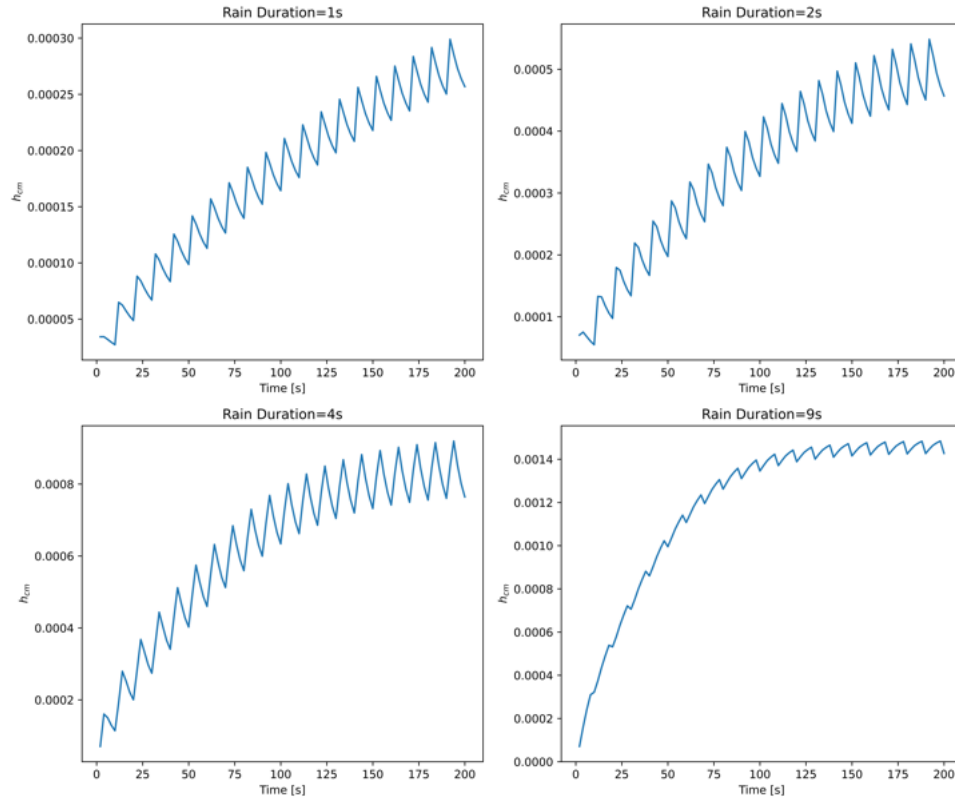


Figure 20: h_{cm} plotted over time for rain durations 1s,2s,4s and 9s.

The other option is to have the rainfall duration based on probabilities. To investigate this, the 1 s duration was assigned a probability of 1/16, the 2 s duration a probability of 7/16, the 4 s duration a probability of 5/16 and the 9 s duration was assigned a probability of 1/16. Figure 21 below shows the water head at $x=0.85$ over time. Clearly, we see that the value increases over time, but it never reaches a value of 0.030 for constant rainfall. This is clearly a result of the fact that extreme events such as 9s are very rare. Similarly, the maximum value h_{cm} over time is only half the maximum value of the constant rainfall case (see Figure 23).

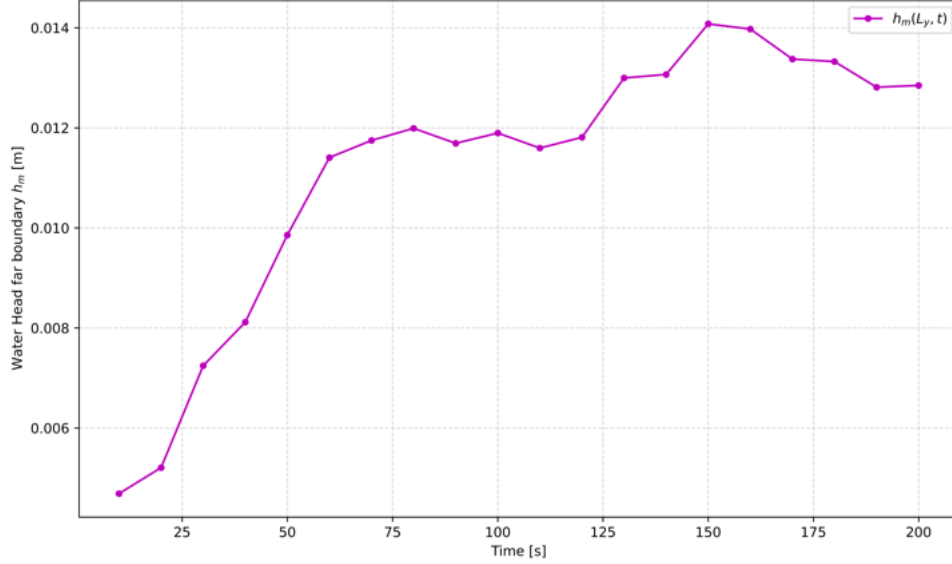


Figure 21: Water head at $x=0.85$ over time for rainfall duration based on probabilities.

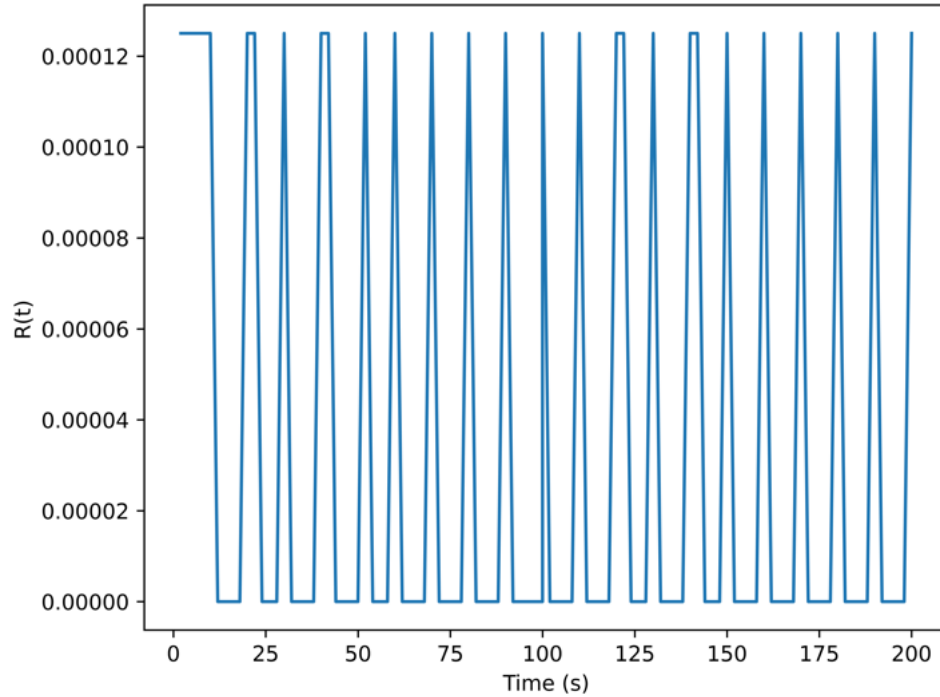


Figure 22: Rain fall over time for rainfall duration based on probabilities.

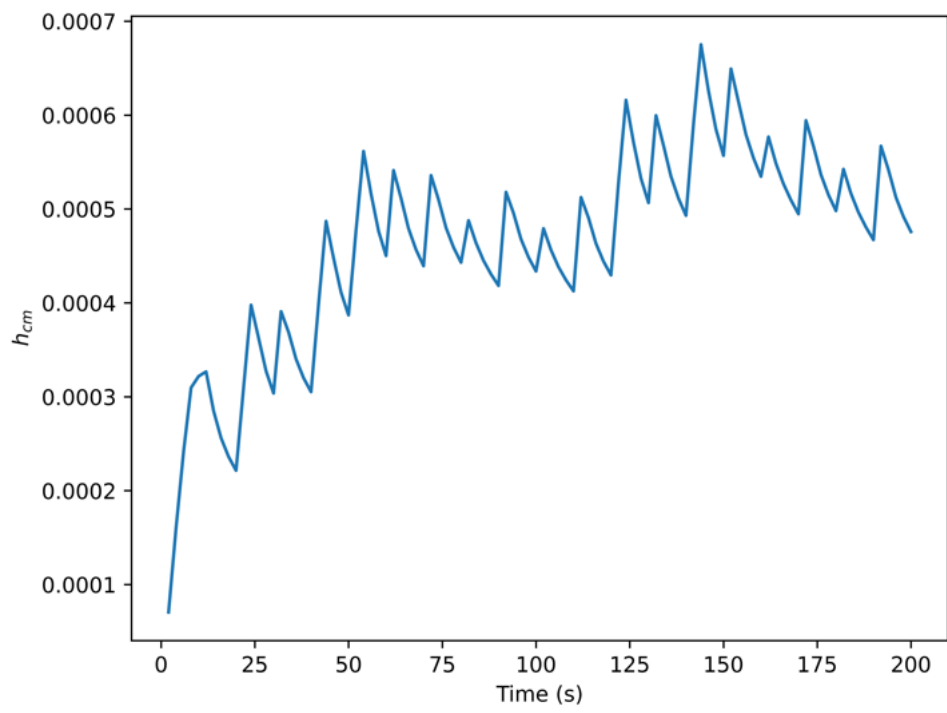


Figure 23: h_{cm} when the rainfall durations are based on probabilities.

5.3 Question 3

The general Crank-Nicolson scheme for our equation is given by:

$$\frac{M_{ij}h_j^{n+1} - M_{ij}h_j^n}{\Delta t} + \frac{L_c}{\sigma_e m_{por}} \frac{h_1^{n+1} - h_1^n}{\Delta t} \delta_{i1} = \frac{1}{2} (F^{n+1} + F^n) \quad (61)$$

where

$$F^n = b_i^n - \frac{1}{\sigma_e m_{por}} \sqrt{g} \max\left(\frac{2}{3} h_1^n(t), 0\right)^{3/2} \delta_{i1}$$

Rearranging the equation gives:

$$M_{ij}h_j^{n+1} + \frac{L_c}{\sigma_e m_{por}} h_1^{n+1} \delta_{i1} - \frac{1}{2} \Delta t F^{n+1} = M_{ij}h_j^n + \frac{L_c}{\sigma_e m_{por}} h_1^n \delta_{i1} + \frac{1}{2} \Delta t F^n \quad (62)$$

$$\begin{aligned} M_{ij}h_j^{n+1} + \frac{L_c}{\sigma_e m_{por}} h_1^{n+1} \delta_{i1} - \frac{1}{2} \Delta t b_i^{n+1} + \frac{1}{2} \Delta t \frac{1}{\sigma_e m_{por}} \sqrt{g} \max\left(\frac{2}{3} h_1^{n+1}(t), 0\right)^{3/2} \delta_{i1} \\ = M_{ij}h_j^n + \frac{L_c}{\sigma_e m_{por}} h_1^n \delta_{i1} + \frac{1}{2} \Delta t b_i^n - \frac{1}{2} \Delta t \frac{1}{\sigma_e m_{por}} \sqrt{g} \max\left(\frac{2}{3} h_1^n(t), 0\right)^{3/2} \delta_{i1}. \end{aligned} \quad (63)$$

To solve the non-linear algebraic system, the NonlinearVariationalSolver in Firedrake is used, which uses Newton iterations to find the solution at each time step. Figure 24 below shows the residual of the last iteration for $t=195$ to $t=200$. Clearly, the residuals are on the order of 1×10^{-16} . Figure 25 shows the residuals at different iterations. Again, the figure clearly shows that the CN scheme converged after 2 iterations. Figure 26 shows the water head profile for the CN scheme, which looks identical to the one where explicit Euler was used.

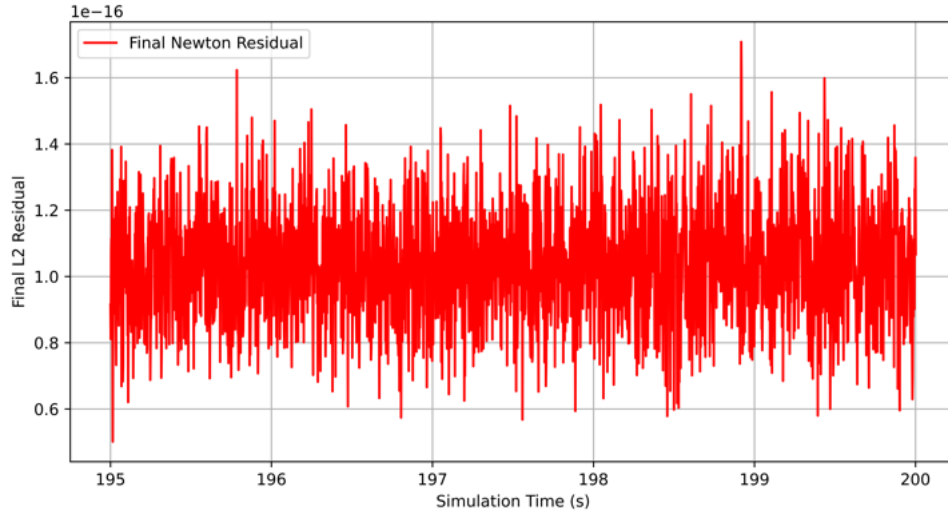


Figure 24: Residuals of the last iteration from $t=195$ to $t=200$

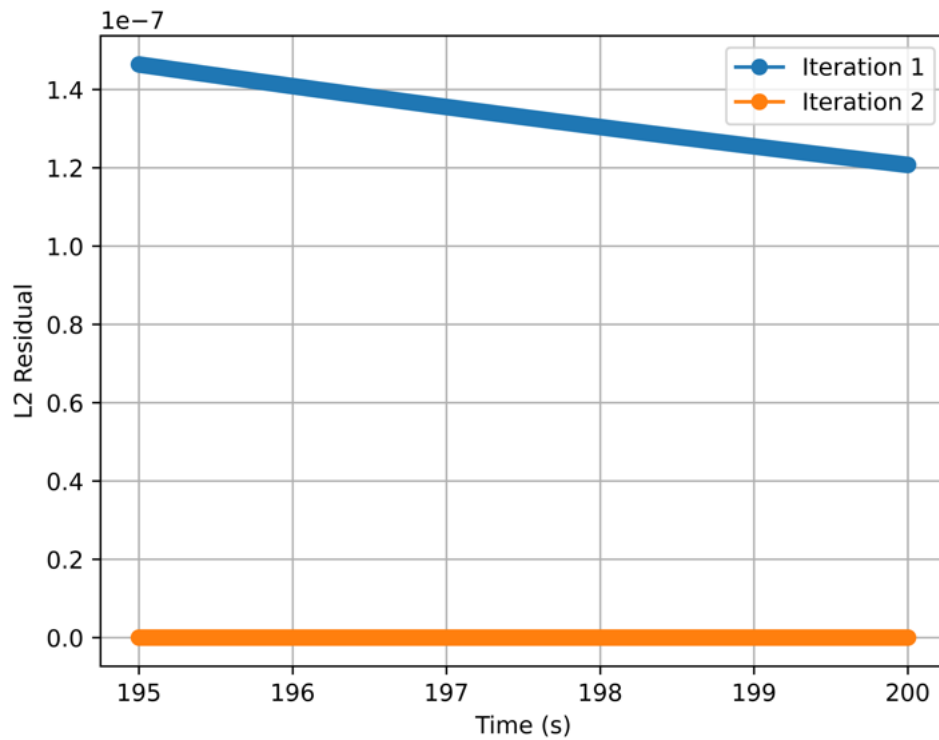


Figure 25: All iterations and their corresponding residuals from $t=195$ to $t=200$.

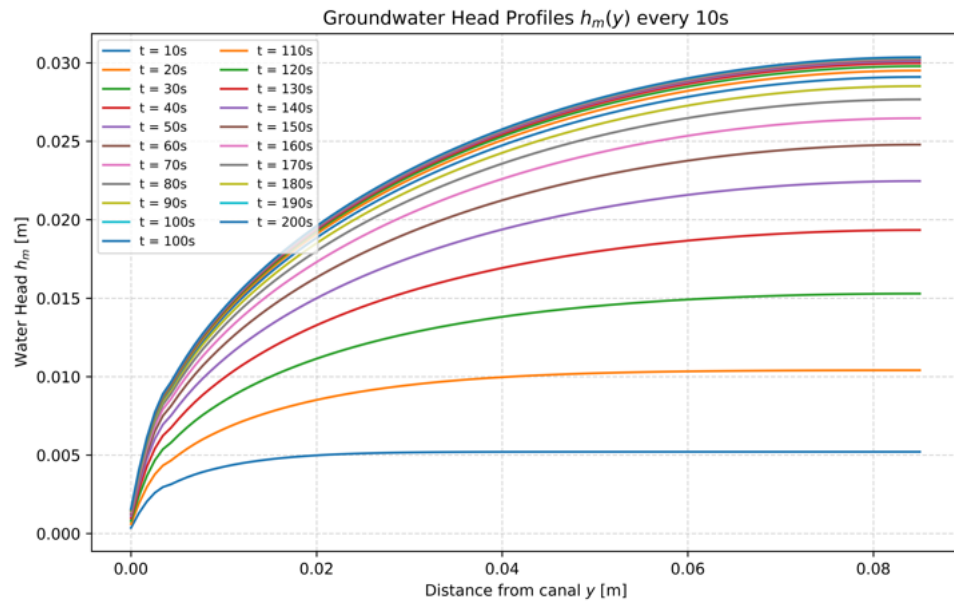


Figure 26: Water head over time when CN is used.

5.4 Question 4

Figure 27 shows the water head profile when $CG=2$ and explicit Euler is used. The CFL number is 2.3. The issue with the oscillations is caused by the time step, which violates the stability criterion. Figure 28 shows the result when CFL is changed to 1.

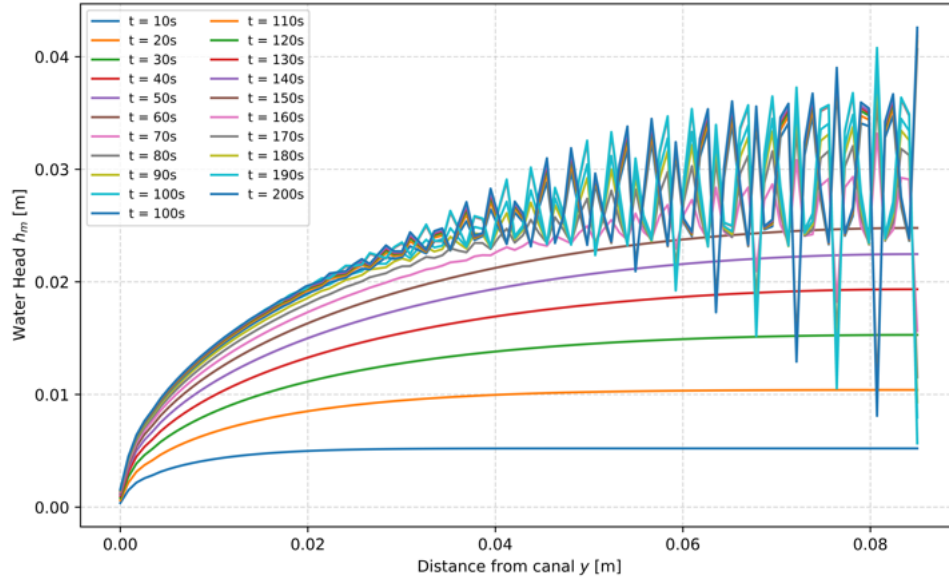


Figure 27: Water head over time with CG2 and explicit euler for CFL 2.3.

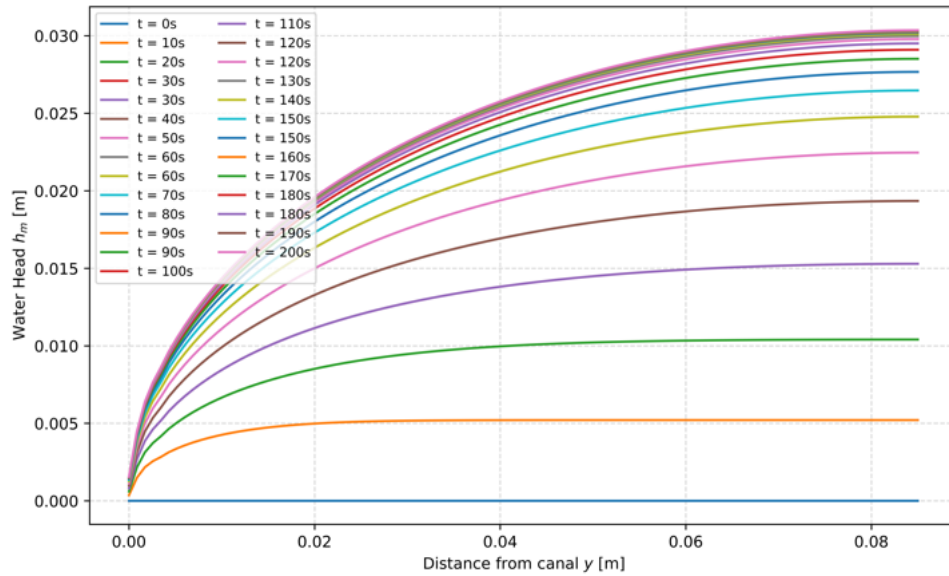


Figure 28: Water head over time with CG2 and explicit euler for CFL 1.