Numerics Exercise 3

Anthony Tran

December 2025

1 Question 1

We consider the following Poisson system:

$$-\nabla^2 u = f \tag{1}$$

$$f(x,y) = 2\pi^2 sin(\pi x)cos(\pi y) \tag{2}$$

$$u(0,y) = u(1,y) = 0 \tag{3}$$

$$\partial_y u|_{y=0} = \partial_y u|_{y=1} = 0 \tag{4}$$

1.1 i)

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

$$u(x,y) = sin(\pi x)cos(\pi y) \tag{5}$$

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

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

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:

$$E(u) = \int_\Omega \frac{1}{2}|\nabla u|^2 - fud\Omega \tag{8}$$

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

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

We now expand our equation to simplify the expression:

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

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

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

$$\frac{dE}{d\lambda} = 0 \tag{12}$$

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

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

We can simply rearrange our equation to get:

$$\int_\Omega \nabla u\cdot\nabla\delta u\, d\Omega = \int_\Omega f\delta u d\Omega \tag{14}$$

The condition is that δu is

$$\delta u(0,y) = \delta u(1,y) = 0 \tag{15}$$

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

$$\int_\Omega \nabla u\cdot\nabla w\, d\Omega = \int_\Omega fw\, d\Omega \tag{16}$$

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:

$$(-\nabla^2 u)w = fw \tag{17}$$

We now integrate this over the domain Ω:

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

We apply the following vector identity:

$$\nabla\cdot(w\nabla u) = (\nabla^2 u)w + \nabla u\cdot\nabla w \tag{19}$$

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

Applying the divergence theorem yields:

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

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

We substitute the expression to get:

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

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

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

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,

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

Hence, we find the weak form:

$$\int_\Omega \nabla u\cdot\nabla w\, d\Omega = \int_\Omega fw\, d\Omega \tag{26}$$

1.2 ii)

given that:

$$u_\sim u_h$$

$$u_h = \sum_{j=1}^{N_h} u_j\phi_j = u_j\phi_j \tag{27}$$

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:

$$\int_\Omega \nabla\sum_{j=1}^{N_h} u_j\phi_j\cdot\nabla\phi_i\, d\Omega = \int_\Omega f\phi_i\, d\Omega \tag{28}$$

Equivalently, we may rewrite it in the form:

$$\sum_{j=1}^{N_h} u_j\int_\Omega \nabla\phi_j\cdot\nabla\phi_i\, d\Omega = \int_\Omega f\phi_i\, d\Omega \tag{29}$$

we can rewrite this in the form:

$$Au = b \tag{30}$$

where:

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

and

$$b_i = \int_\Omega f\phi_i\, d\Omega$$

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

$$E(u) = \int_\Omega \frac{1}{2}|\nabla u|^2 - fud\Omega \tag{31}$$

We use:

$$u_h = \sum_{j=1}^{N_h} b_j\Gamma_j = b_j\Gamma_j \tag{32}$$

Hence,

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

We rewrite the expression as:

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

As before, we minimize our expression by finding

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

After rearranging, we find:

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

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

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

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

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:

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

The integral for a quadrilateral element is then computed as:

$$A^K_{\alpha\beta} = \int_K (\nabla N_\alpha \cdot \nabla N_\beta) \det(J) d\bar{\eta} \tag{40}$$

This may also be written as:

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

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

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

2 Question 4

2.1 p=1 and various grid sizes.

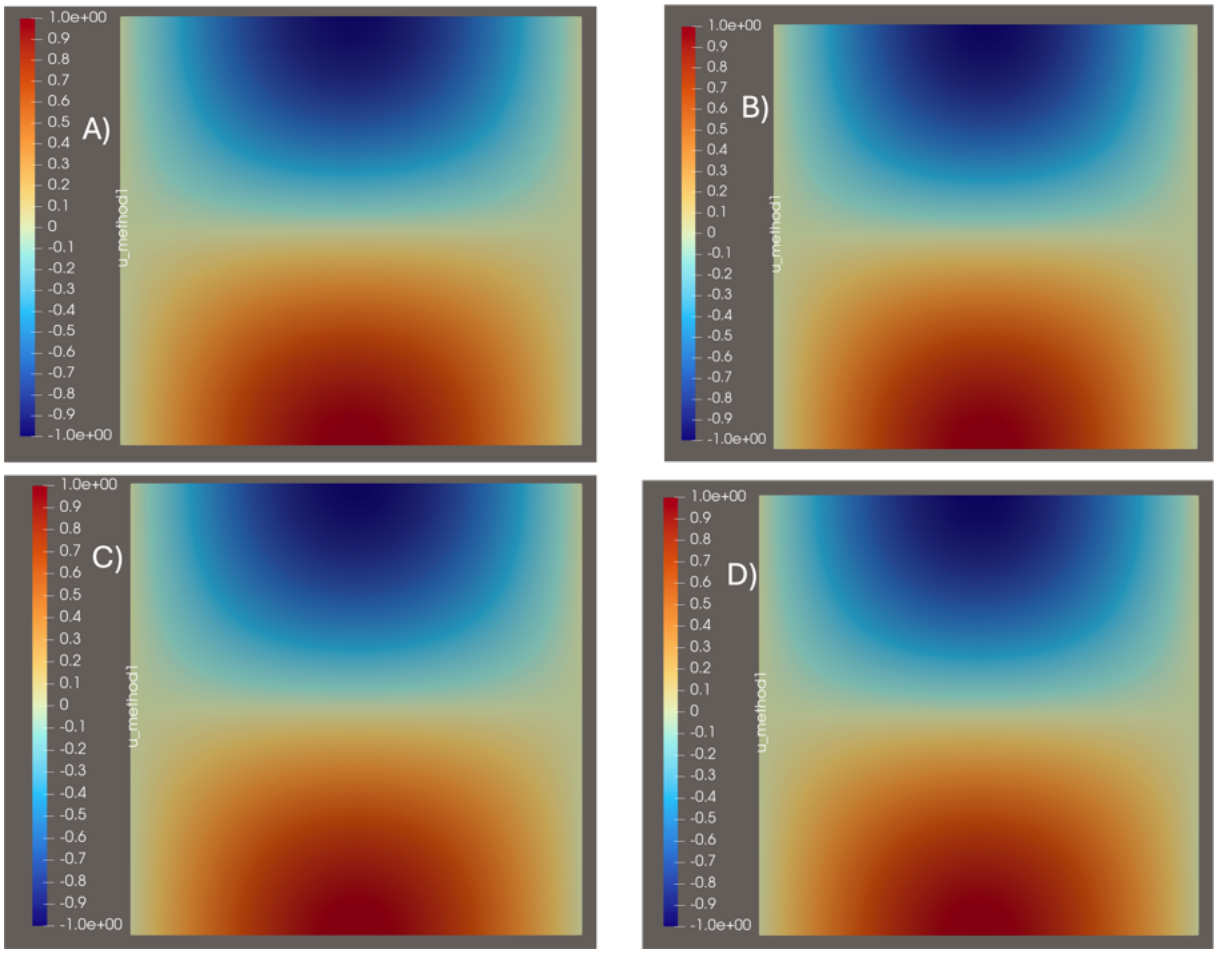

Figure 1: Method 1: Contour plots of velocity for different grid sizes and p=1. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

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.

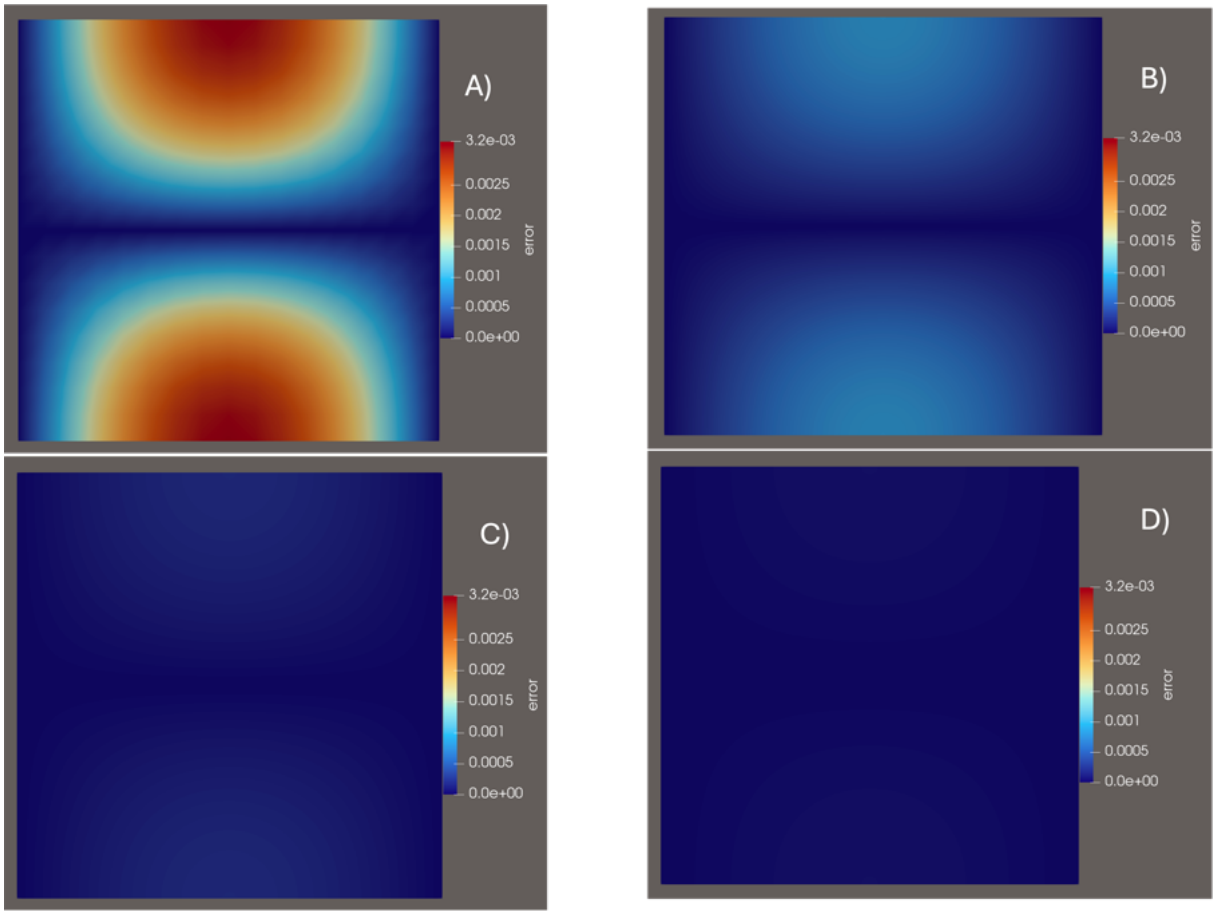

Figure 2: Method 1: Difference between numerical solution and the exact solution for p=1. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

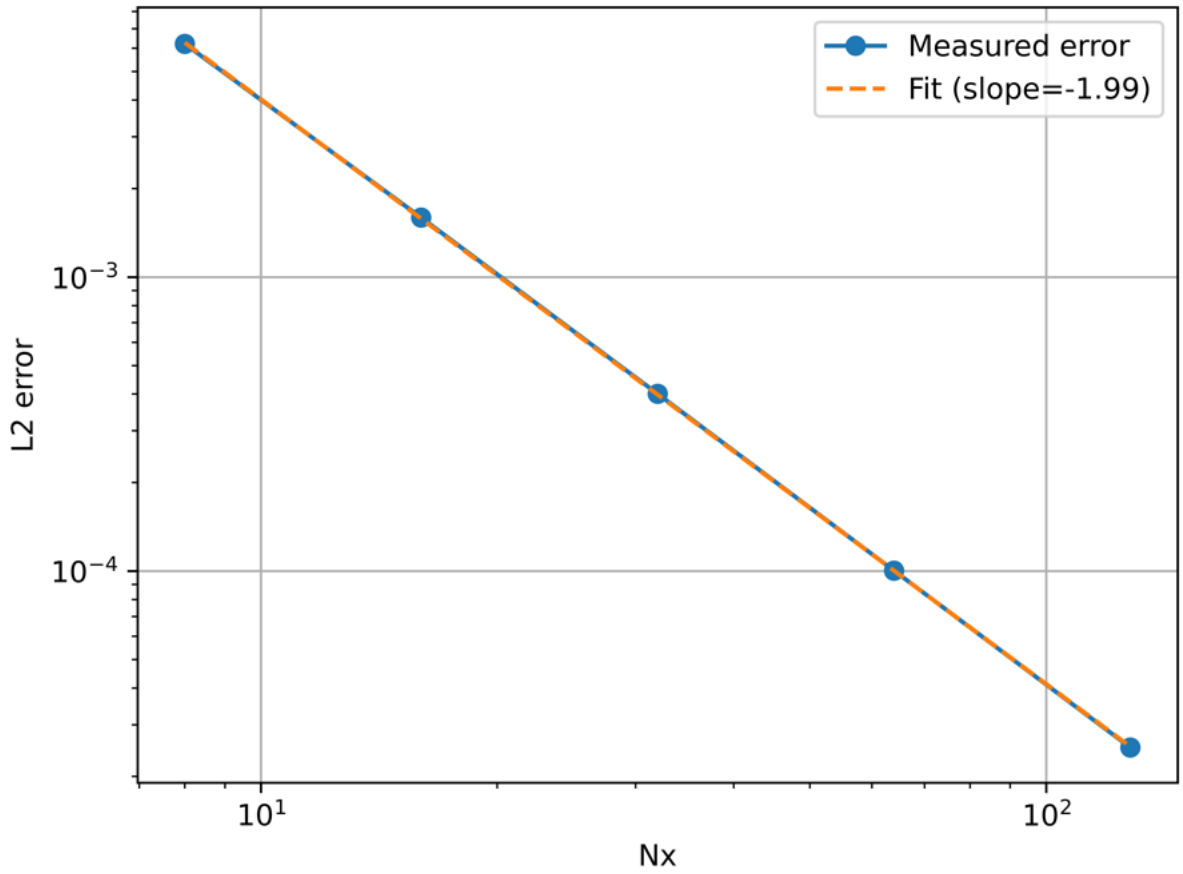

Figure 3: Method 1: L2 norm as a function of grid size for p=1.

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.

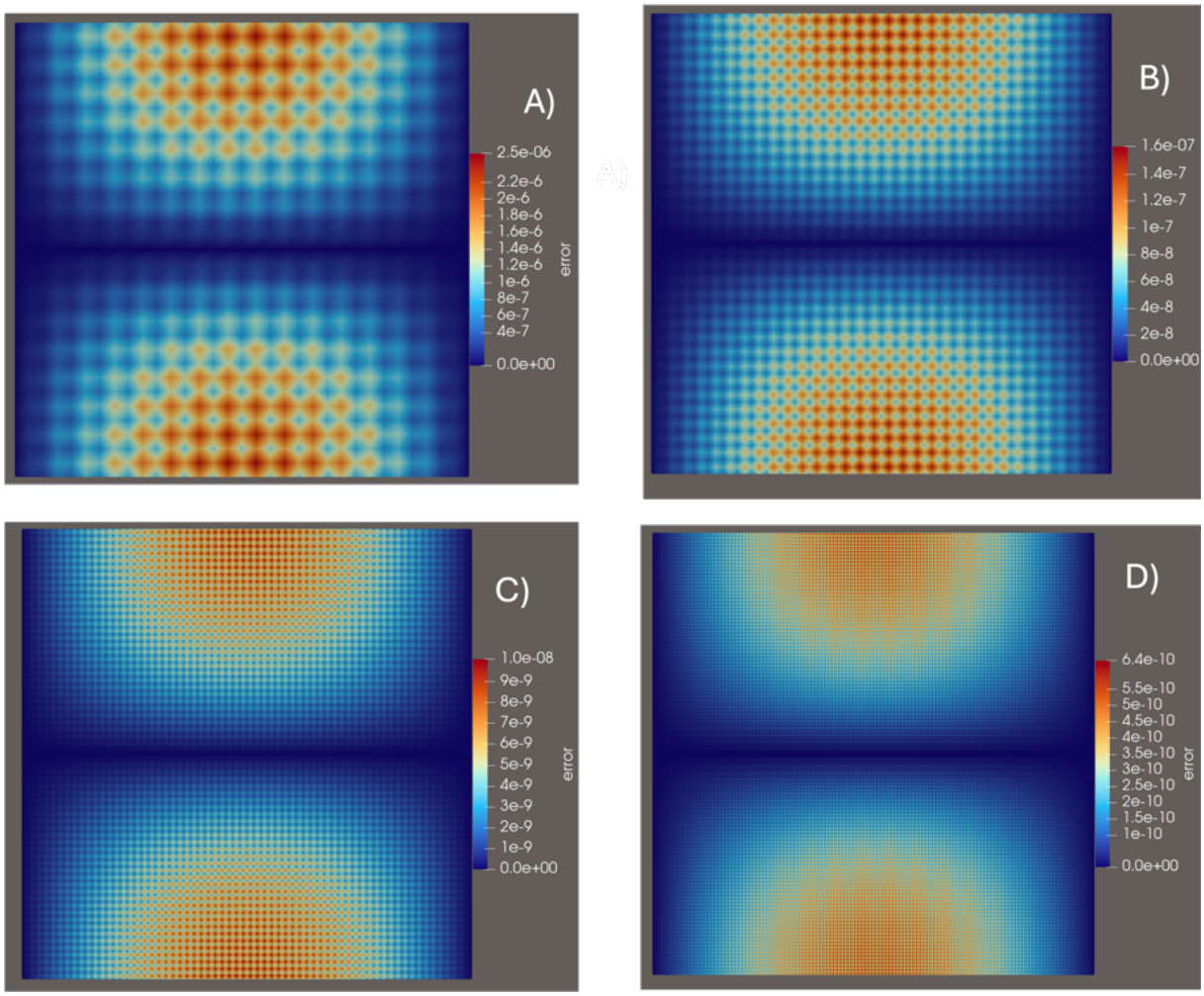

Figure 4: Method 1: Difference between numerical solution and the exact solution for p=2. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

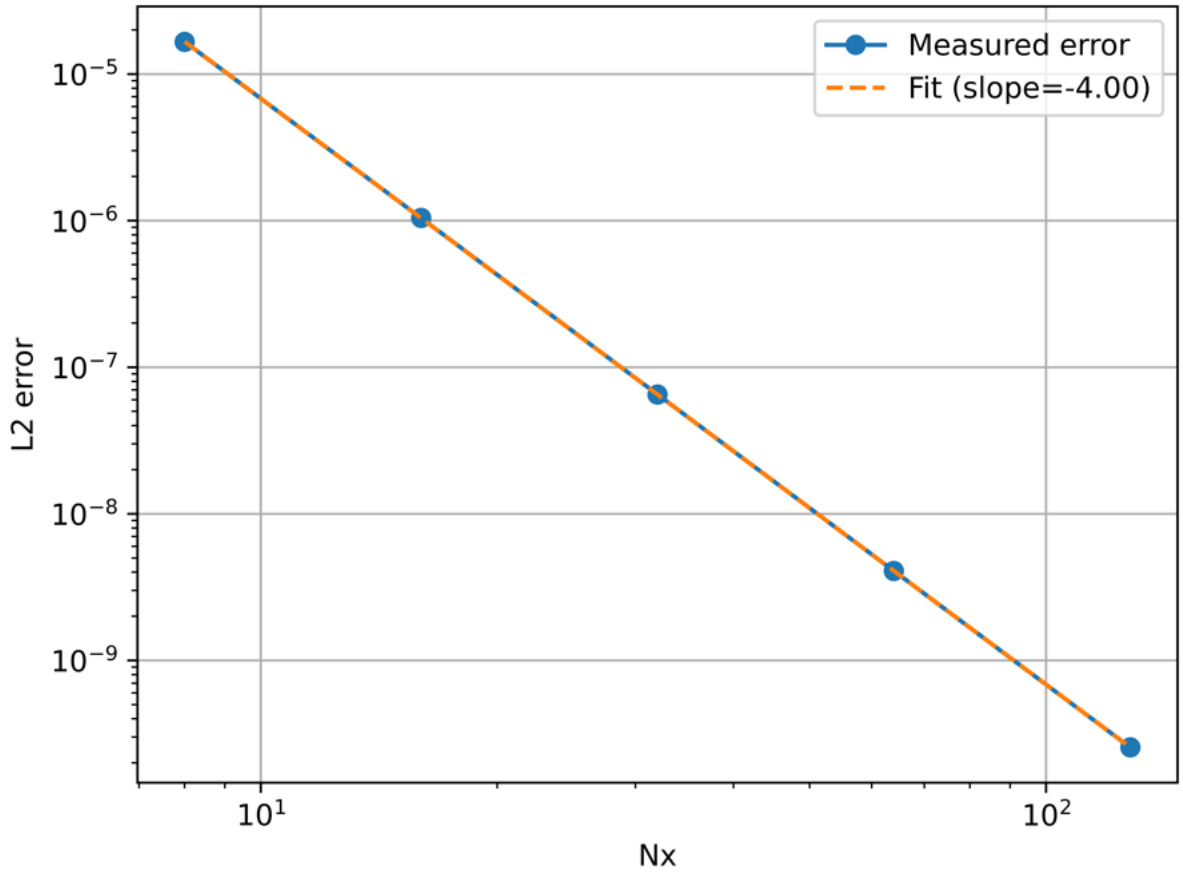

Figure 5: Method 1: L2 norm as a function of grid size for p=2.

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 16x16 grid and decreases to 1.1×10^{-11} for the 128 x 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.

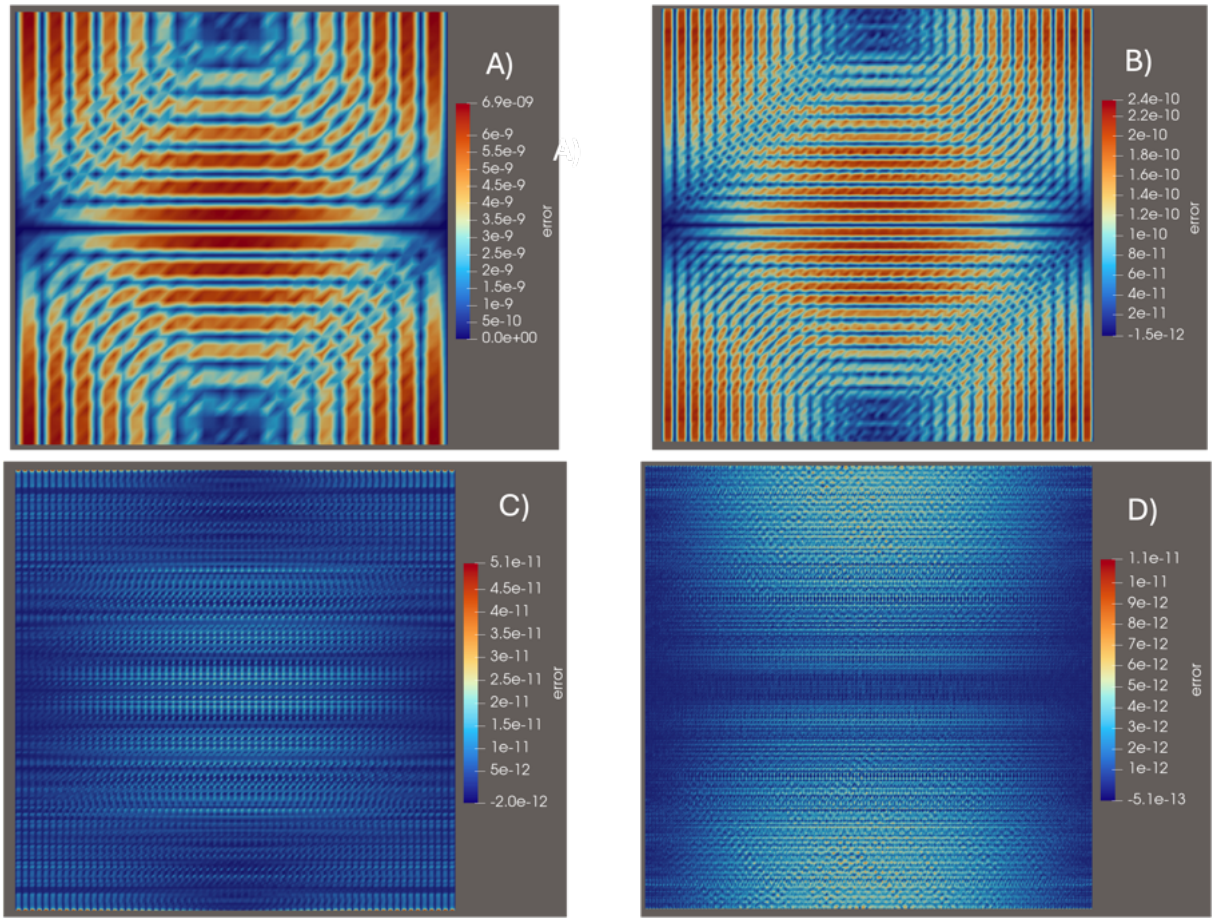

Figure 6: Method 1: Difference between numerical solution and the exact solution for p=3. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

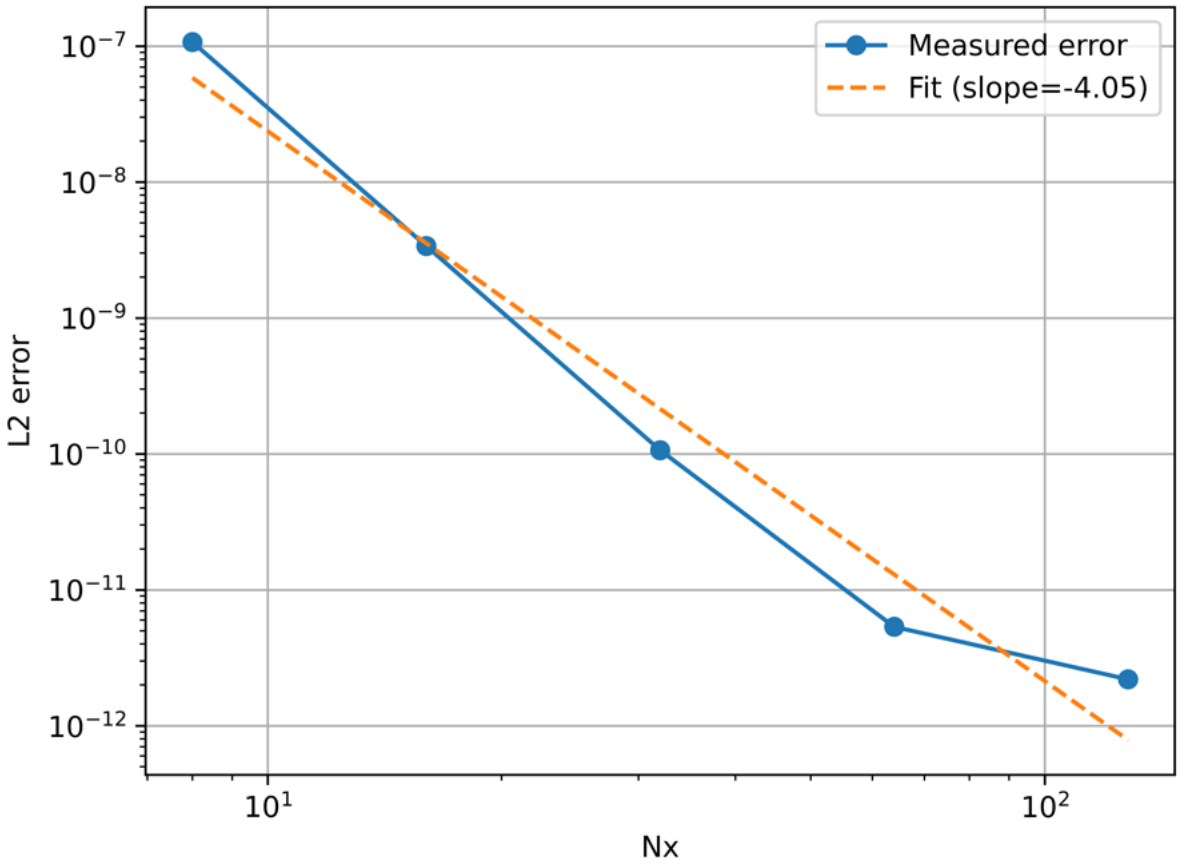

Figure 7: Method 1: L2 norm as a function of grid size for p=3.

Resolution	**p=1**	**p=2**	**p=3**
8	6.2×10^{-3}	1.6×10^{-5}	1.0×10^{-7}
16	1.5×10^{-3}	1.0×10^{-6}	3.3×10^{-9}
32	4.0×10^{-4}	6.5×10^{-8}	1.0×10^{-10}
64	1.0×10^{-4}	4.0×10^{-9}	5.3×10^{-12}
128	2.5×10^{-5}	2.5×10^{-10}	2.1×10^{-12}

Table 1: L2 norms for different grid sizes and p values

Just by looking at the errors, we can see that the pairs h=128,p=1 and $h = 8, p = 2$ are equivalent. In addition, $h = 64, p = 2$ and $h = 16, p = 3$ are equivalent. In general, we observe that the errors scale like h^p. Clearly, high order CGs allow accurate solutions with a coarser mesh. This is because they capture curvatures much better than lower orders (e.g p=1), which may just not capture it at all.

3 Question 5

:

```python
from firedrake import *
import math

def solve_poisson_firedrake(nx=16, ny=16, degree=1,CG="CG",save_data=False, quad=True, solver_params=None):

    if solver_params is None:
        solver_params = {'ksp_type': 'cg', 'pc_type': 'none'}

    # Mesh and function space
    # --- Step 1: Discretize the domain Omega ---
    # Creates the grid. Quadrilateral=True uses squares instead of triangles.

    mesh = UnitSquareMesh(nx, ny, quadrilateral=quad)

    # This represents u_h = sum(u_j * phi_j).
    # 'CG' 1 means Continuous Galerkin (linear polynomials)
    V = FunctionSpace(mesh, CG, degree)

    x, y = SpatialCoordinate(mesh)

    # Exact solution and forcing
    u_exact = sin(pi * x) * cos(pi * y)
    f = Function(V).interpolate(2 * pi**2 * u_exact)

    # Boundary conditions (x = 0 and x = 1)
    bc_x0 = DirichletBC(V, Constant(0.0), 1)
    bc_x1 = DirichletBC(V, Constant(0.0), 2)
    bcs = [bc_x0, bc_x1]

    # ------------------------------------------------------------------
    # Method 1: Manual weak form
    # ------------------------------------------------------------------
    # u is the TrialFunction (the unknown solution u_h)
    # v is the TestFunction (the weight function w or phi_i)
    u = TrialFunction(V)
    v = TestFunction(V)
    # a(u, v) = integral( grad(u) . grad(v) dx ) -> Corresponds to Matrix A_ij
    a = inner(grad(u), grad(v)) * dx
    # L(v) = integral( f * v dx ) -> Corresponds to Vector b_i

    L = f * v * dx

    u_1 = Function(V, name="u_method1")
    # Solves the linear system Au = b
    solve(a == L, u_1, bcs=bcs, solver_parameters=solver_params)

    # ------------------------------------------------------------------
    # Method 2: Variational ( R i t z Galerkin ) formulation
    # ------------------------------------------------------------------
    u_2 = Function(V, name="u_method2")

    # The Energy Functional E(u) = integral( 0.5*|grad(u)|^2 - f*u )
    J = (0.5 * inner(grad(u_2), grad(u_2)) - u_2 * f) * dx
    # F is the derivative (variation) of the energy.
    # Solving F == 0 is equivalent to finding the minimum dE/du = 0.
    F = derivative(J, u_2, v)

    # Nonlinear solver (Newton) used here to find the stationary point of the functional
```

```
    solve(F == 0, u_2, bcs=bcs)

    # ------------------------------------------------------------------
    # Error analysis
    # ------------------------------------------------------------------
    u_exact_fun = Function(V).interpolate(u_exact)

    # Calculate L2 Norm of the error: sqrt( integral( (u_approx - u_exact)^2 ) )
    L2_1 = sqrt(assemble((u_1 - u_exact_fun)**2 * dx))
    L2_2 = sqrt(assemble((u_2 - u_exact_fun)**2 * dx))
    L2_diff = sqrt(assemble((u_2 - u_1)**2 * dx))

    if save_data:
      outfile = VTKFile(f'output_{nx}_{degree}.pvd')
      outfile.write(u_1, u_2)

    return {
        "mesh_size": 1.0 / nx,
        "u_method1": u_1,
        "u_method2": u_2,
        "L2_error_method1": L2_1,
        "L2_error_method2": L2_2,
        "L2_difference": L2_diff,
    }
```

Listing 1: Firedrake Poisson Solver

4 Question 6

We use the following u for our manufactured solution:

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

with

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

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

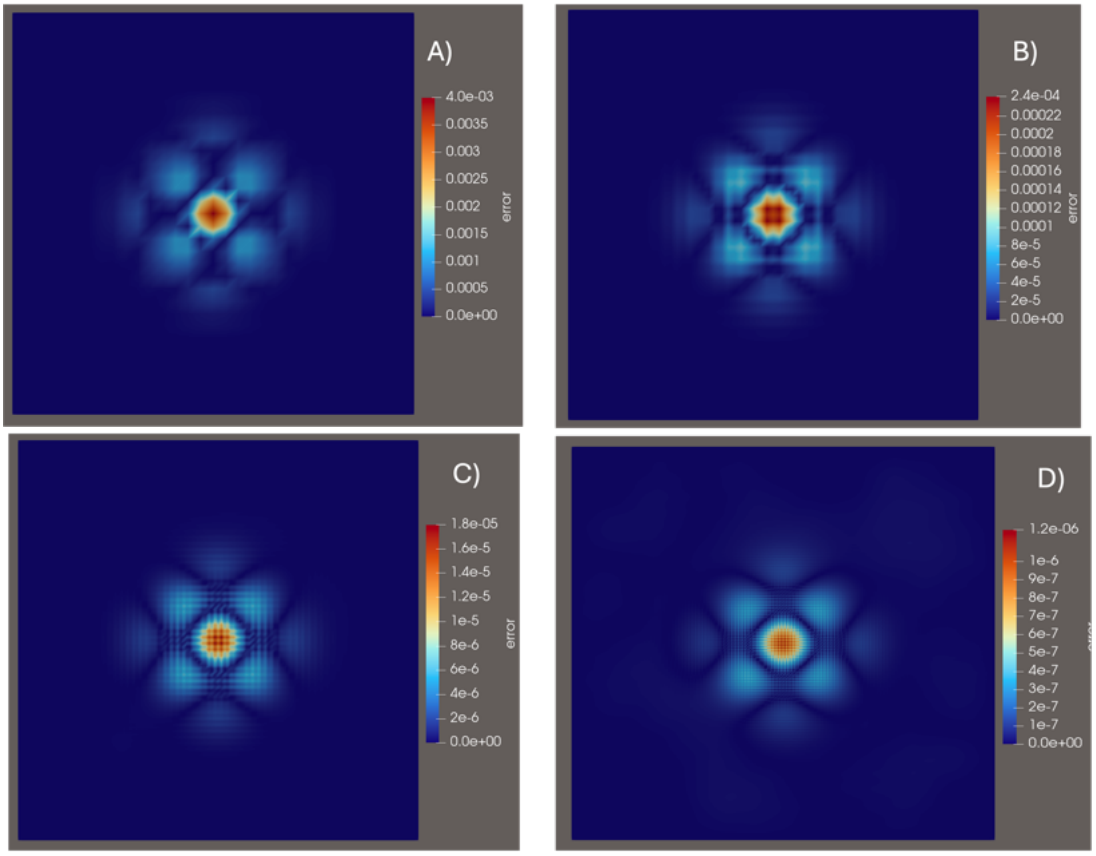

Figure 8: Method 1: Difference between numerical solution and the new exact solution for p=2. A) 16x16 grid, B) 32 x32 grid , C) 64 x64 grid and D) 128 x128 grid.

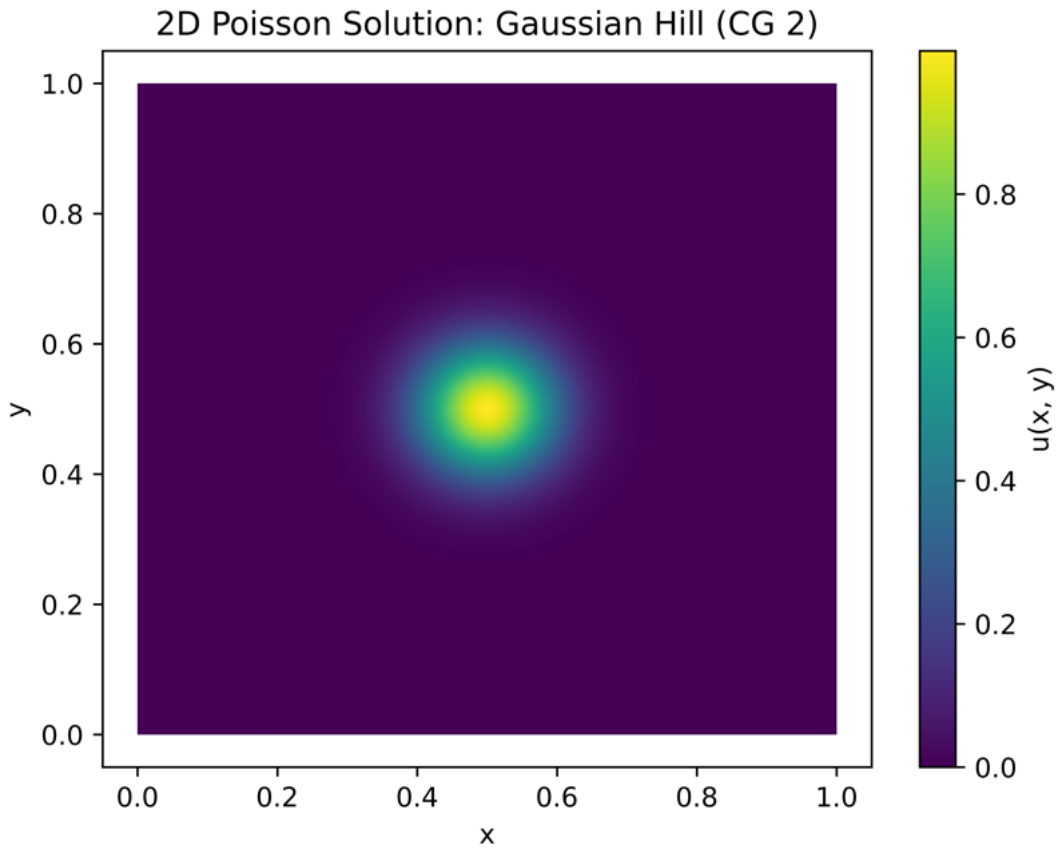

Figure 9: Constructed Analytical solution

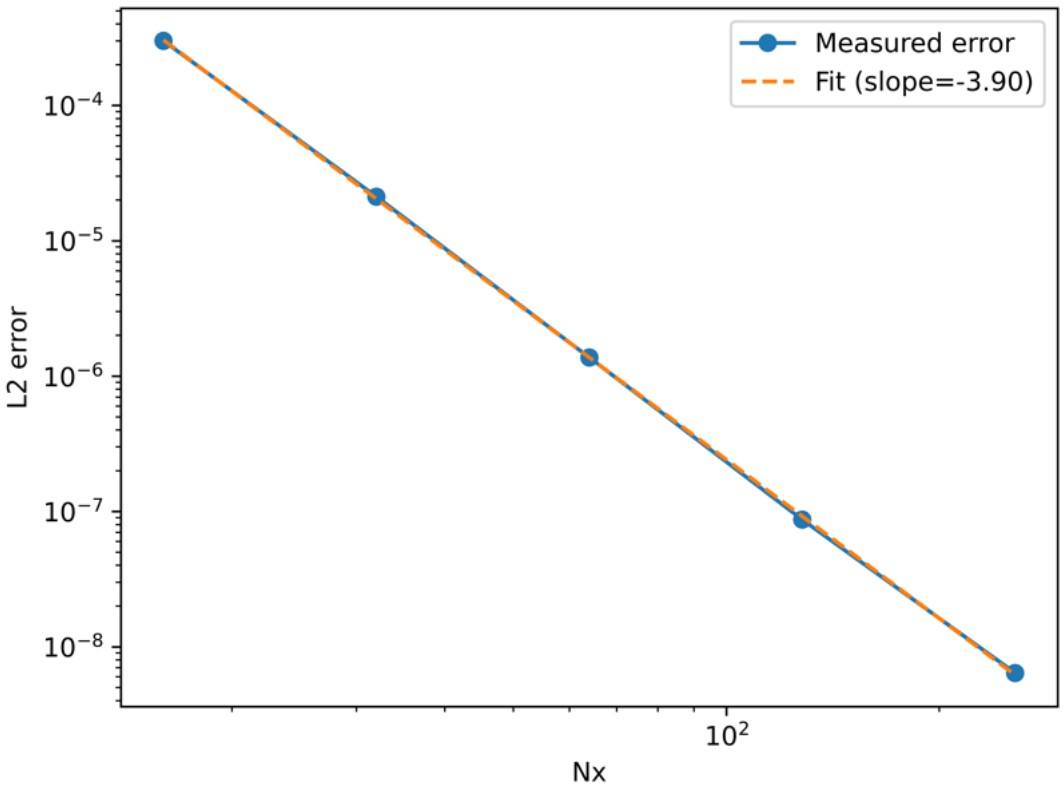

Figure 10: L2 error as a function of the grid size for the new solution.

5 Groundwater

We are given the equations:

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

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

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

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

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:

$$\int_0^{L_y} \alpha g \partial_y(w_v h_m \partial_y h_m)\phi \, dy = \alpha g w_v h_m \partial_y h_m \phi|_{y=0}^{y=L_y} - \int_0^{L_y} \alpha g w_v h_m \partial_y h_m \partial_y \phi \, dy \quad (45)$$

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 $h_m(0,t) = h_{cm}(t)$ since that is where the canal is and to ensure continuity.
We can now discretize our equation. First, we make the following approximation:

$$h_m = \sum_{j=1}^{N} h_j \phi_j \tag{51}$$

Hence, the left hand side becomes:

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

We can rewrite this as:

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

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

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

The right hand side of the equation becomes:

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

We can also substitute our expression for h_m to get:

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

The update equation is given by:

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

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

The finite differences scheme is given by:

$$\frac{h_i^{n+1} - h_i^n}{\Delta t} - \frac{\alpha g}{2} \frac{(h_{i+1}^n)^2 - 2(h_i^n)^2 + (h_{i-1}^n)^2}{\Delta y^2} = \frac{R^n}{\sigma_e m_{por}} \tag{59}$$

Rearranging gives:

$$h_i^{n+1} = h_i^n + \frac{\Delta t \alpha g}{2} \frac{(h_{i+1}^n)^2 - 2(h_i^n)^2 + (h_{i-1}^n)^2}{\Delta y^2} + \frac{\Delta t R^n}{\sigma_e m_{por}} \tag{60}$$

Note that we can also discretize the nonlinear diffusion equation using the self adjoint method, which gives the following update scheme:

$$h_i^{n+1} = h_i^n + \Delta t \alpha g \frac{h_{i+1/2}^n (h_{i+1}^n - h_i^n) - h_{i-1/2}^n (h_i^n - h_{i-1}^n)}{\Delta y^2} + \frac{\Delta t R^n}{\sigma_e m_{por}} \tag{61}$$

where

$$h_{i+1/2}^n = \frac{h_{i+1}^n - h_{i-1}^n}{2}$$

The canal equation can also be discretized using forward Euler to give:

$$h_1^{n+1} = h_1^n + \frac{\Delta t m_{por} \sigma_e \alpha g}{L_c \Delta y} \big((h_2)^2 - (h_1)^2 \big) - \sqrt{g} max(\frac{2}{3} h_1, 0)^{3/2} \tag{62}$$

where h_1 is the boundary cell and h_2 is the cell next to the boundary.
Since this is a diffusion like problem, our time step restriction is

$$\Delta t < \frac{\Delta y^2}{2 \alpha g h_{max}} \tag{63}$$

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 Nx=40, and the results indicate convergence, as shown in Figure 13.

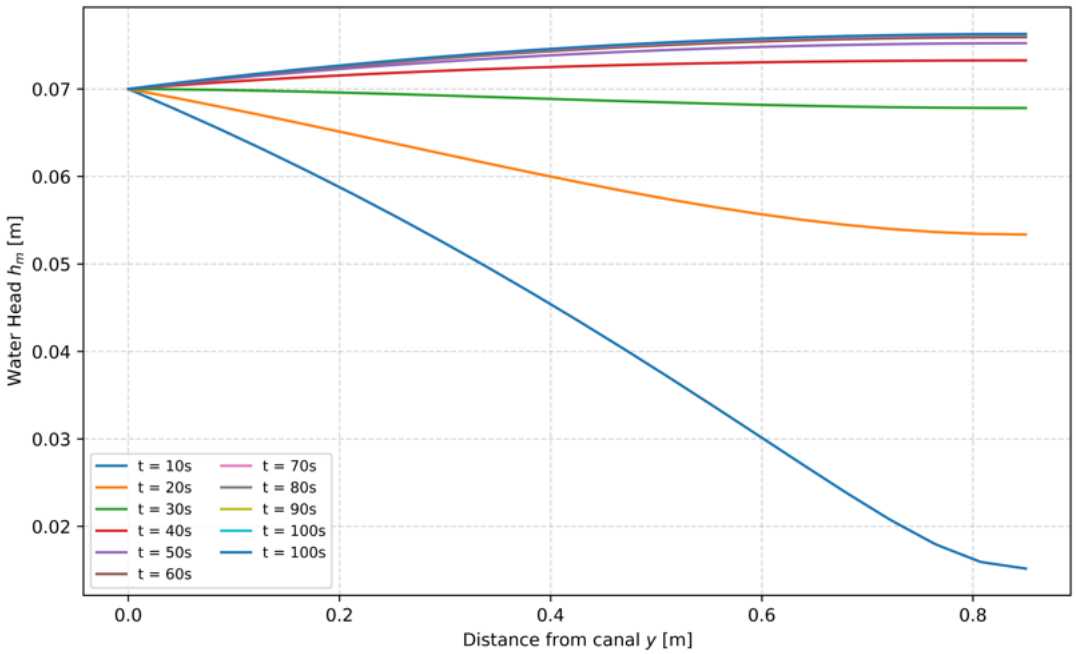

Figure 11: Water head at different times where h_{cm}=0.07 is fixed.

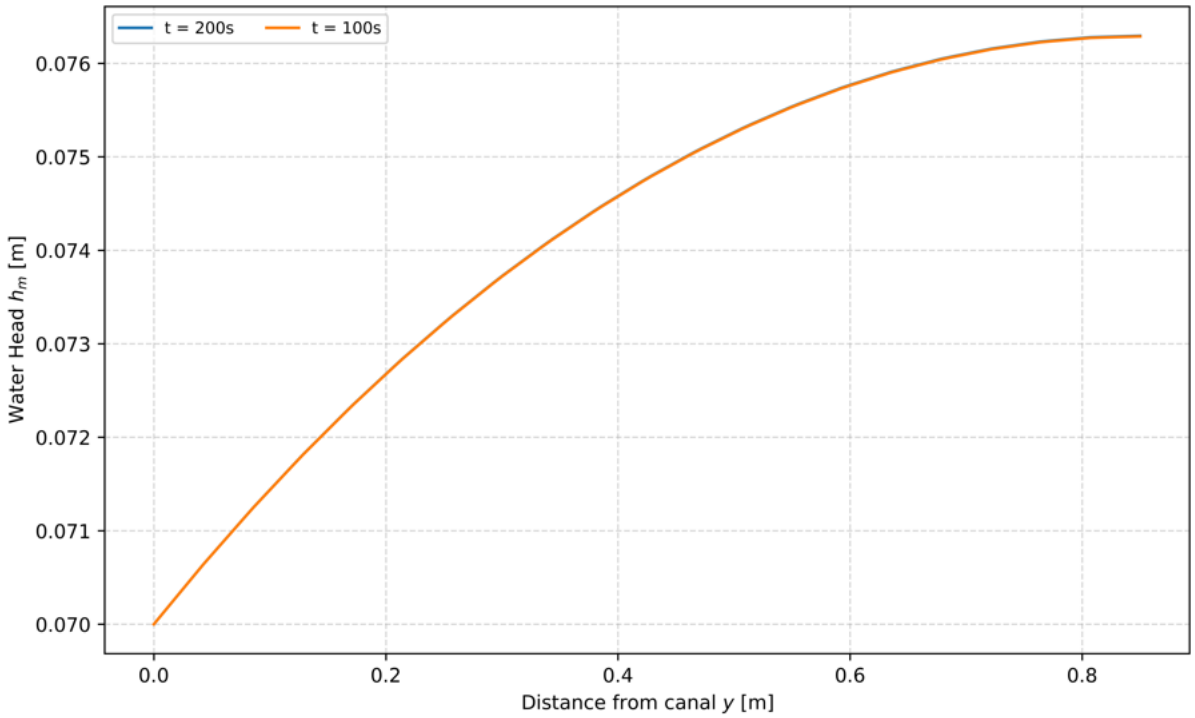

Figure 12: Water head plotted at t=100 and t=200 for $h_{cm} = 0$ Steady state is reached by 100 seconds.

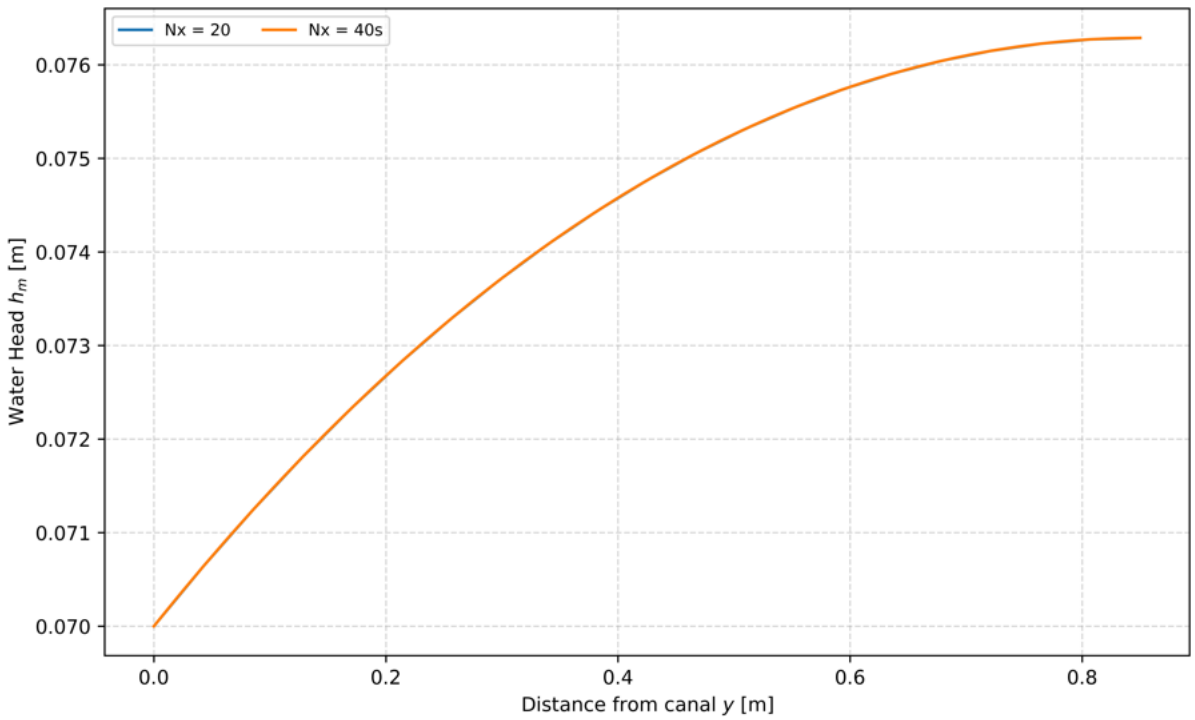

Figure 13: Convergence of the solution. The water head profile at t=100 for Nx=20 and Nx=40.

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 Nx=20, Nx=40 and Nx=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 Nx=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.

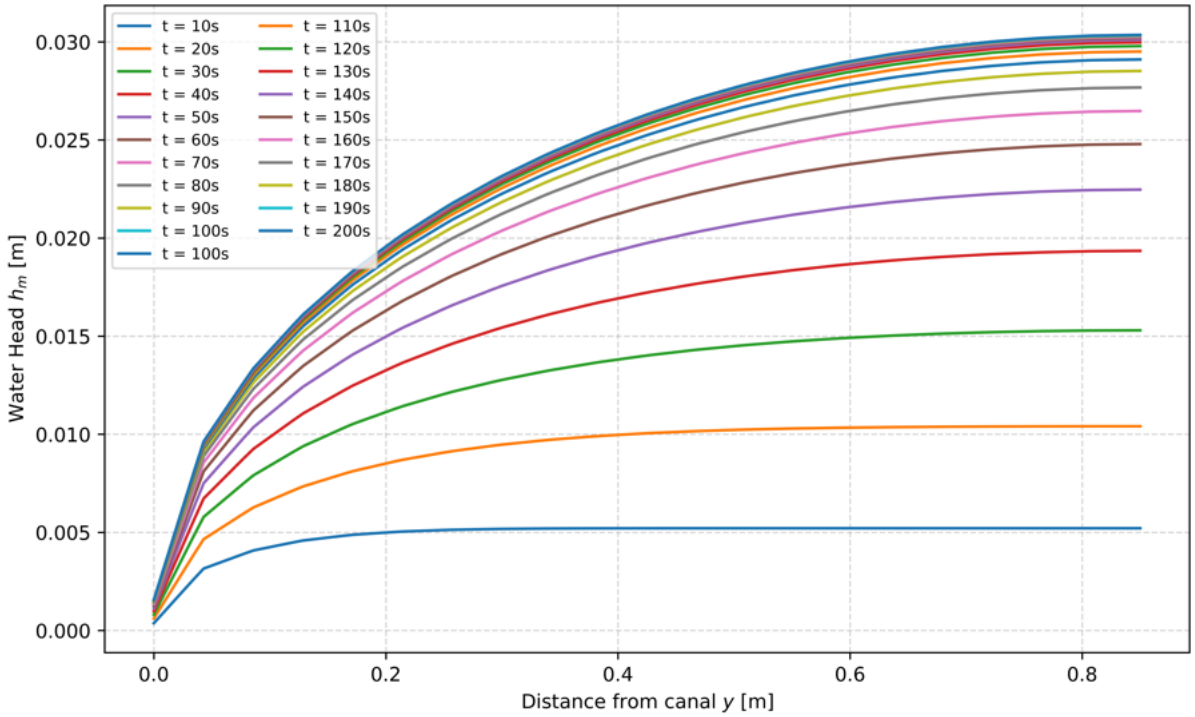

Figure 14: Water head at different times for h_{cm}=0.0 and where h_{cm} is not fixed

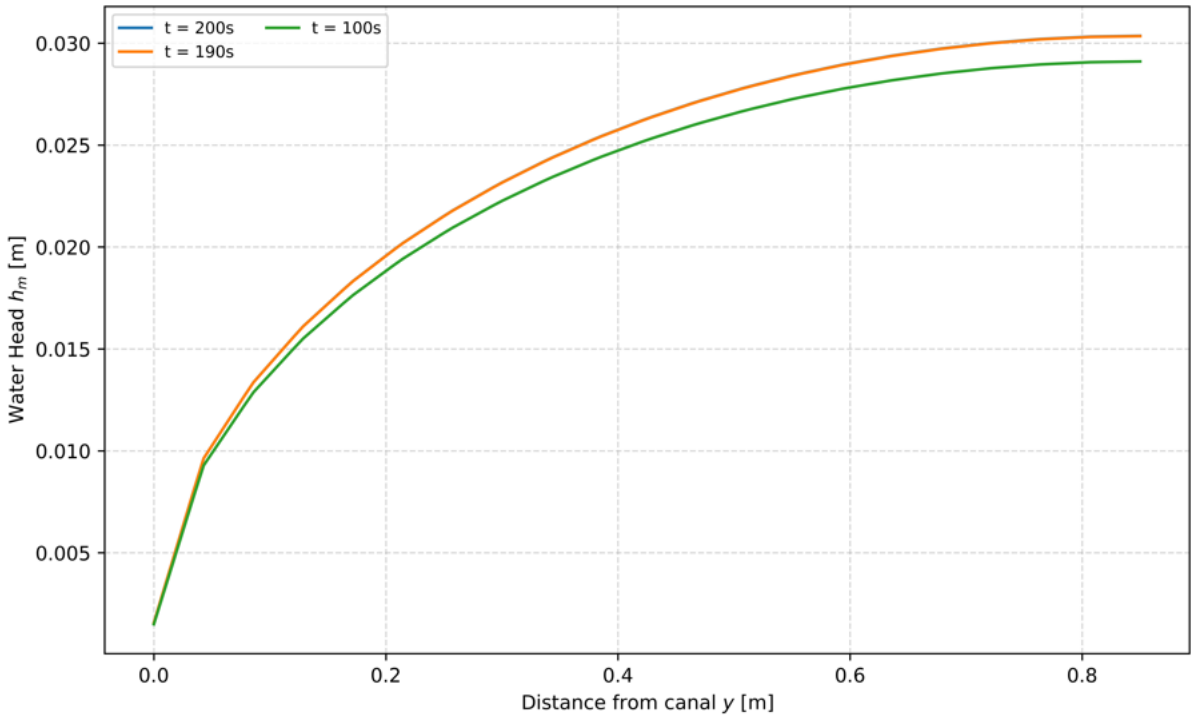

Figure 15: Water head plotted fat t=100,t=200 and t=190 s. Steady state is reached by 200s.

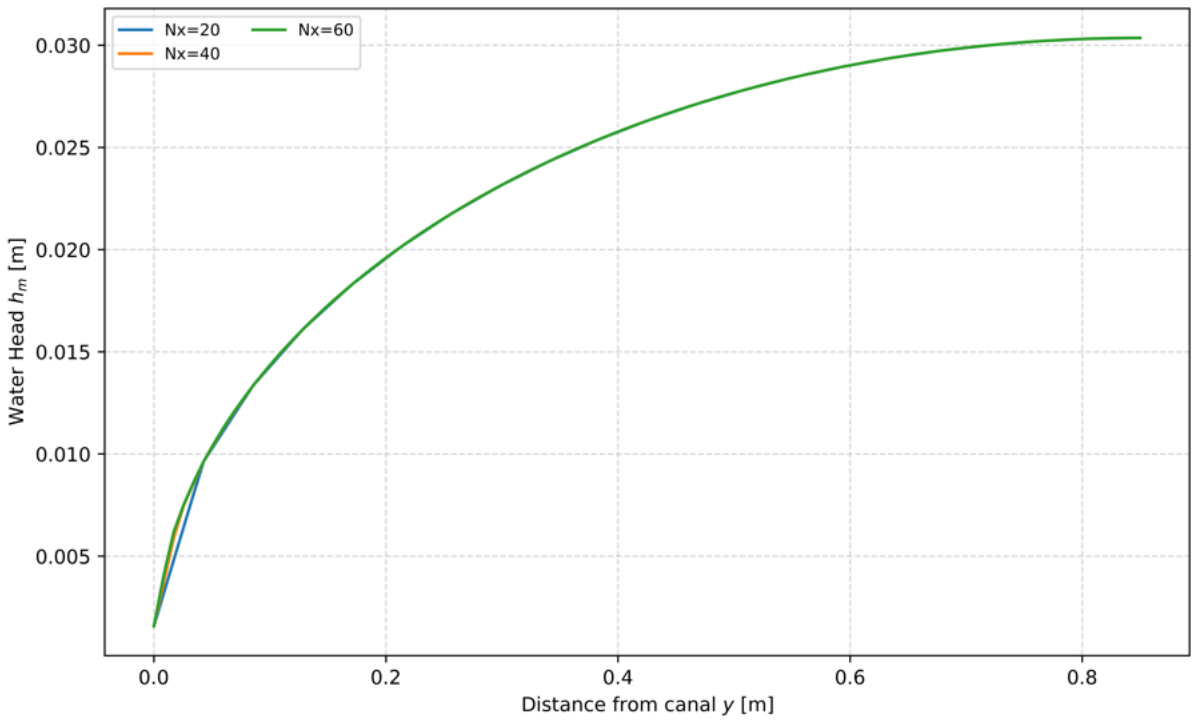

Figure 16: Convergence of the solution. The water head profile at t=200 for Nx=20, Nx=40 and Nx=60.

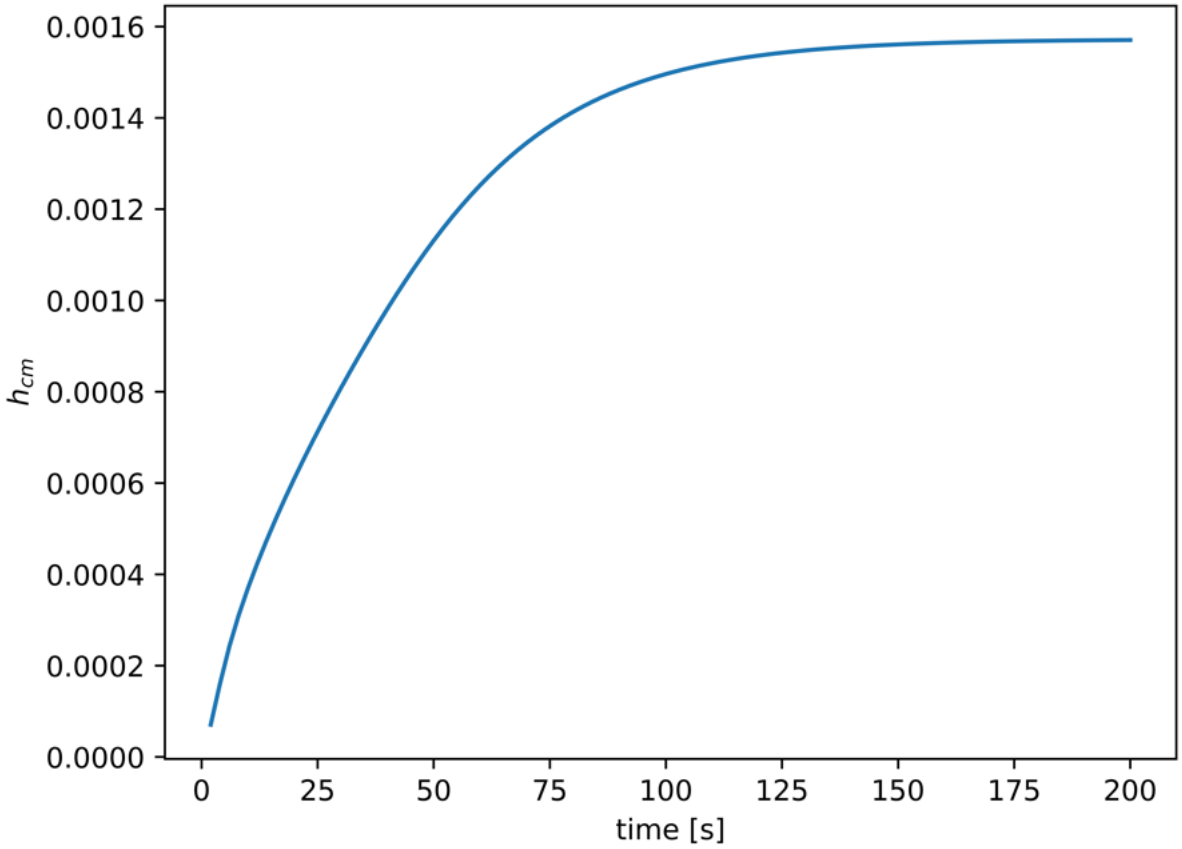

Figure 17: h_{cm} plotted over time. Steady state value is roughly 0.0016.

Variable rainfall was also tested. Every 10-second window, the rain fell only fell 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=200s 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

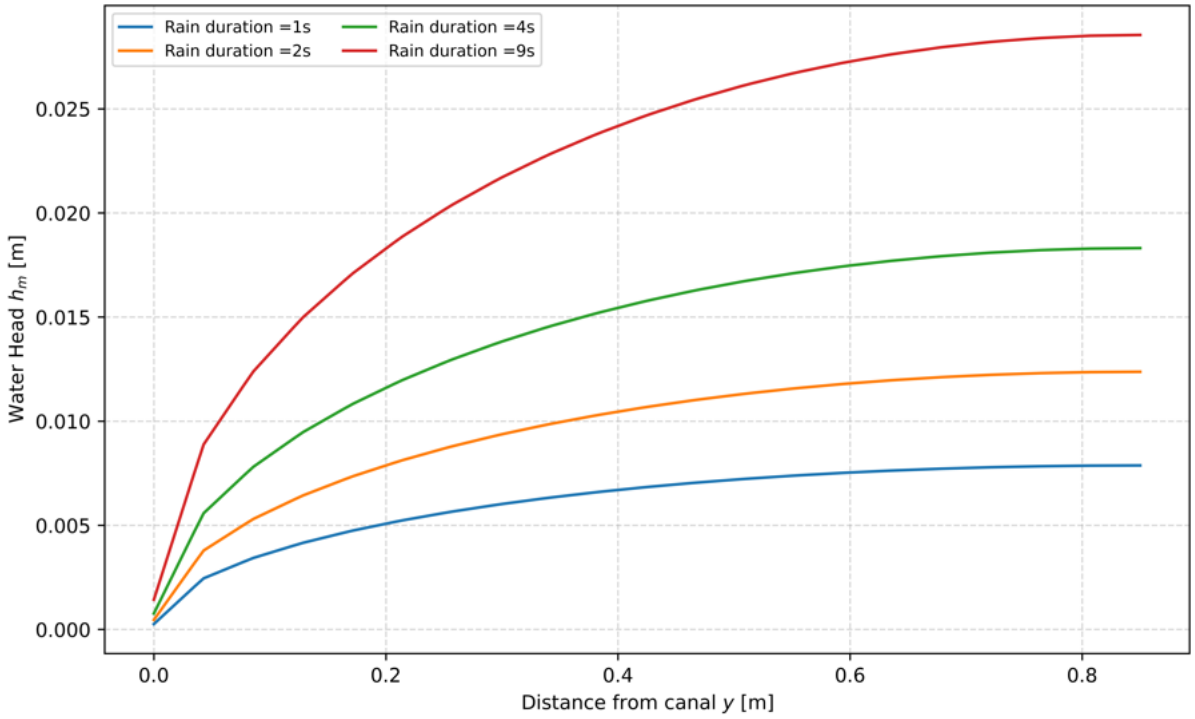

Figure 18: Water head plotted at t=200 for rain durations 1s,2s,4s and 9s.

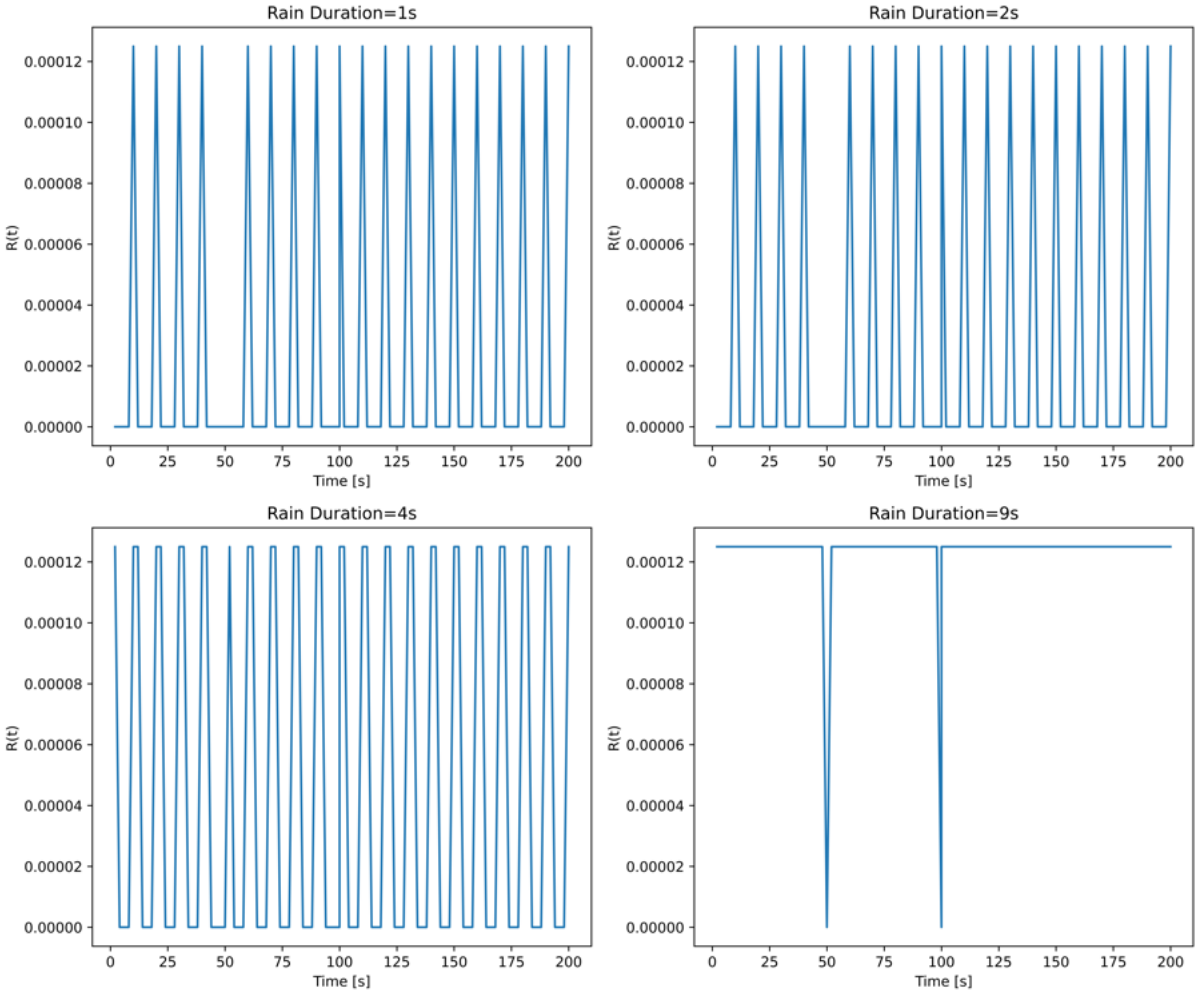

Figure 19: Rainfall plotted over time for rain durations 1s,2s,4s and 9s.

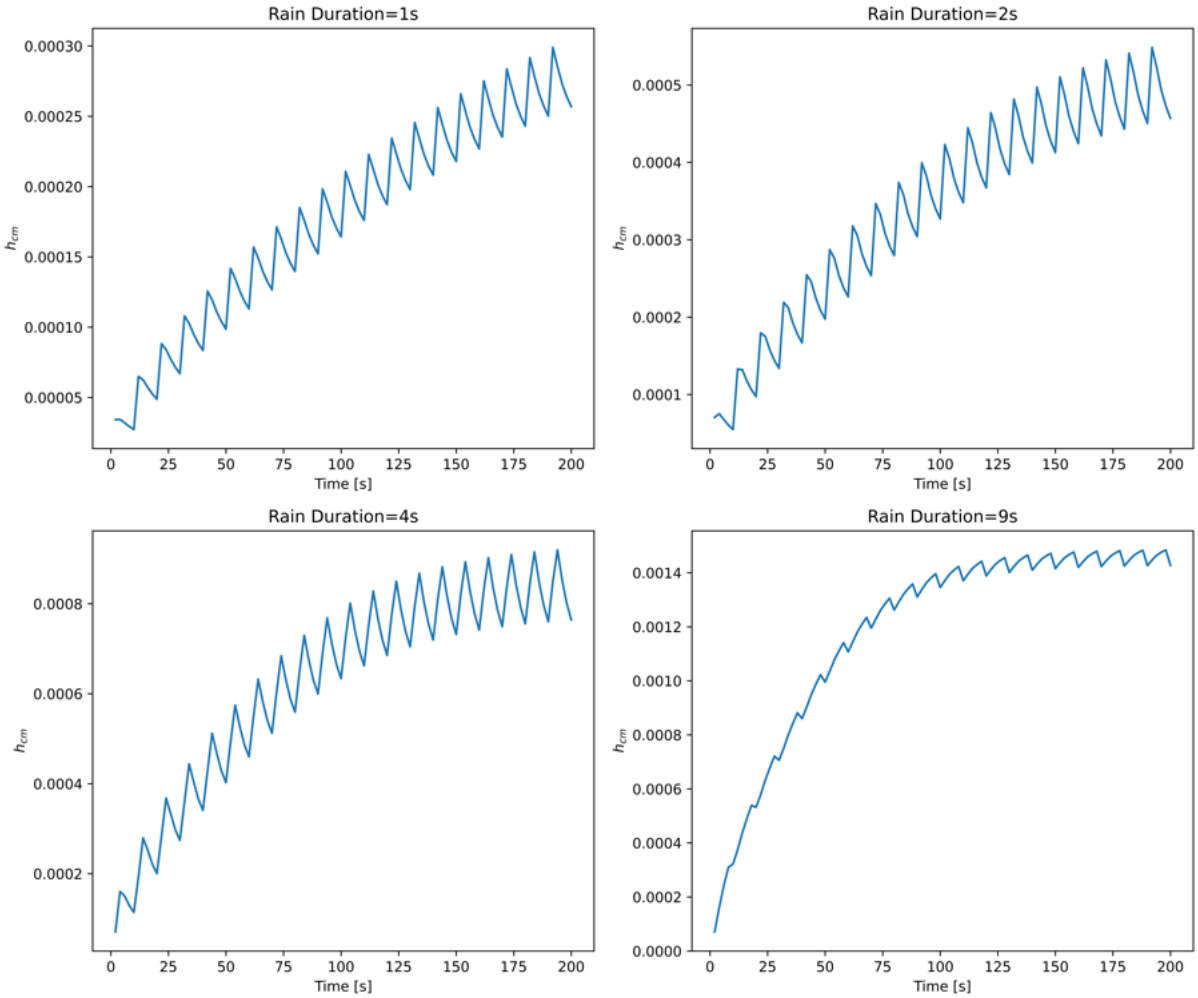

Figure 20: h_{cm} plotted over time for rain durations 1s,2s,4s and 9s.

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 valueh_{cm} over time is only half the maximum value of the constant rainfall case (see Figure 23).

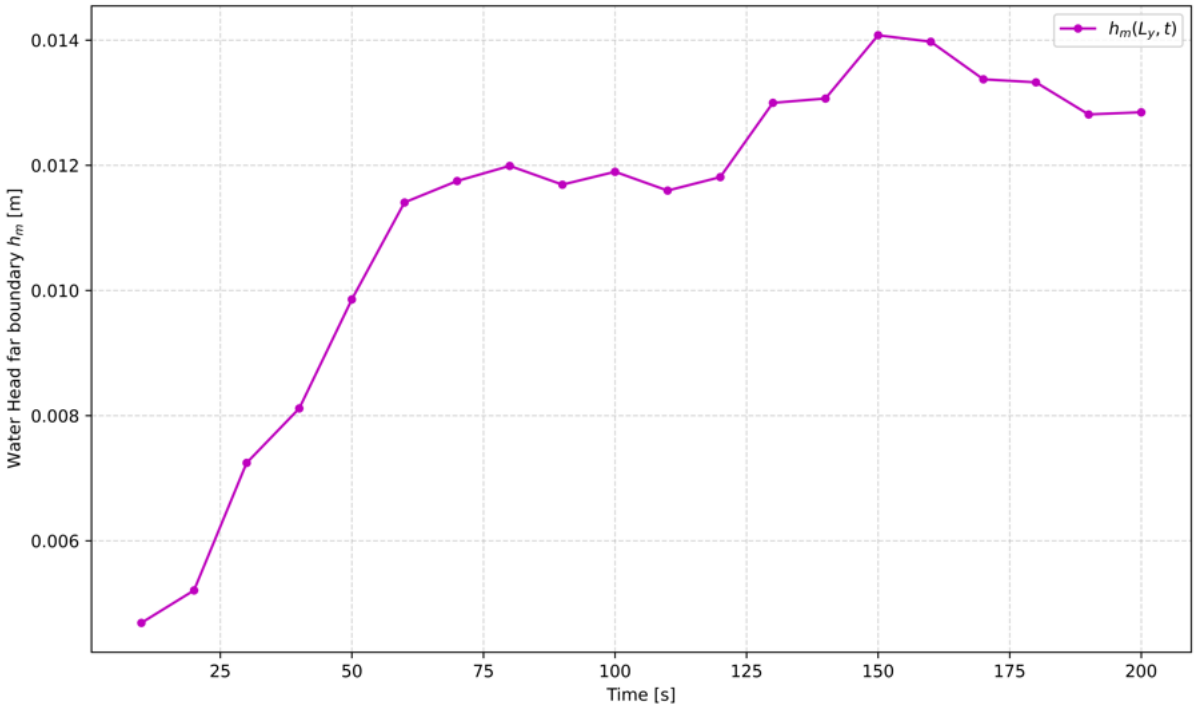

Figure 21: Water head at x=0.85 over time for rainfall duration based on probabilities.

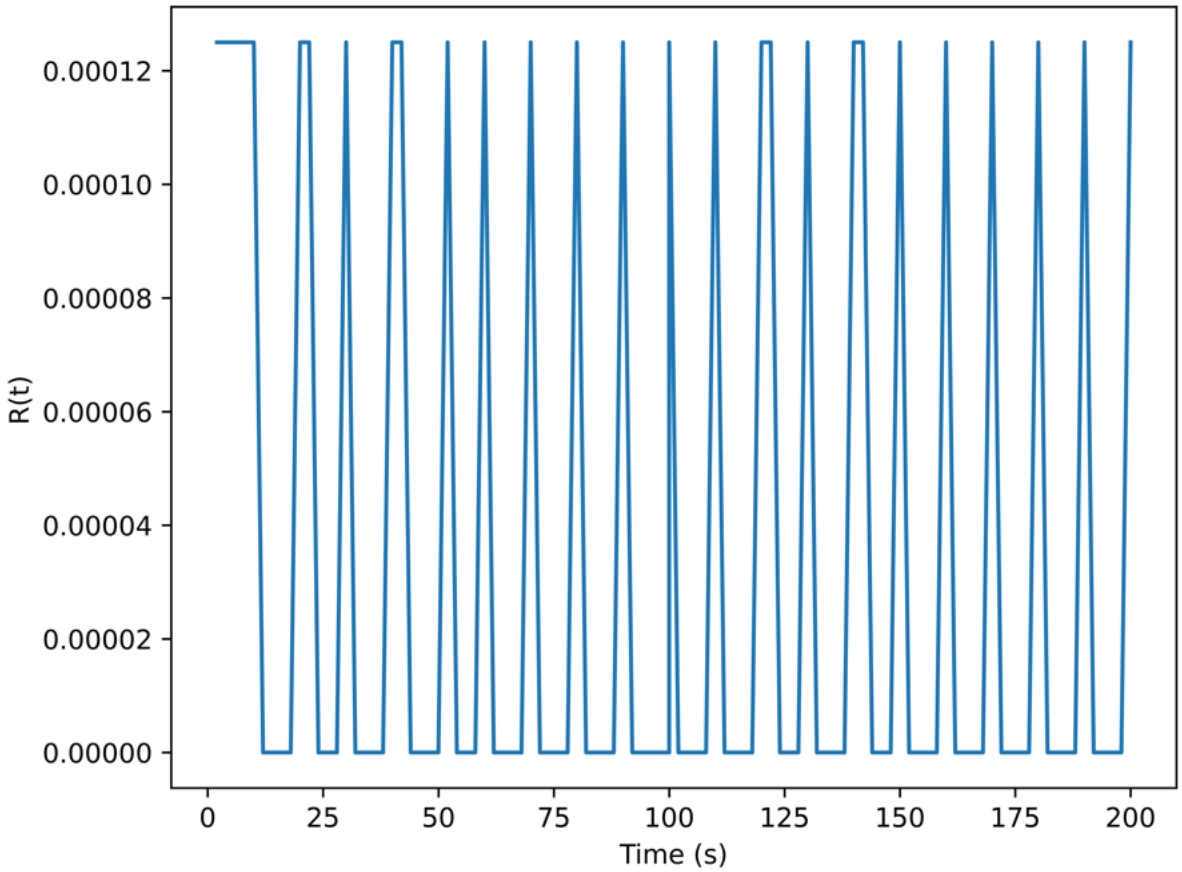

Figure 22: Rain fall over time for rainfall duration based on probabilities.

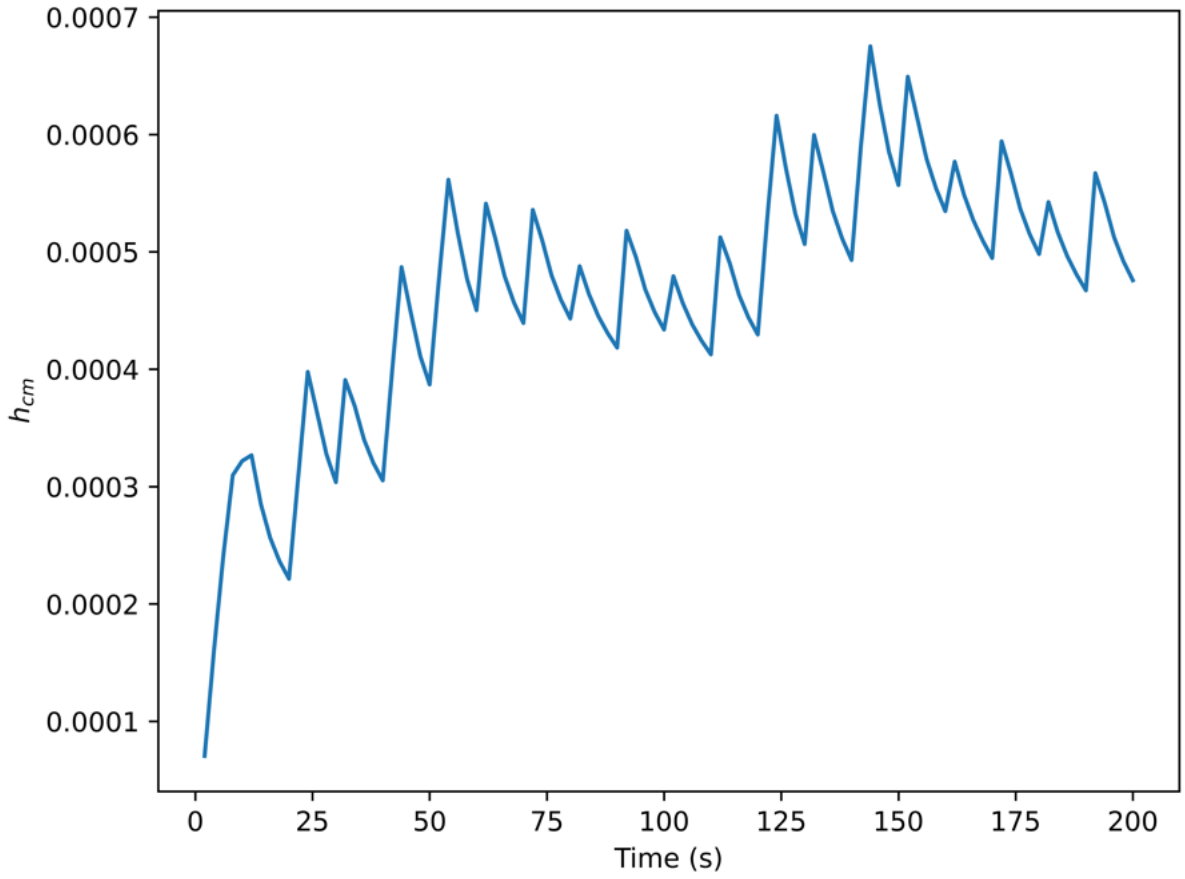

Figure 23: h_{cm} when the rainfall durations are based on probabilities.

5.2.4 Comparison with finite differences

Figure 24 shows the non self adjoint finite difference and finite element solutions at t=200 when rain is turned on all the time. Clearly, the agreement is excellent. Figure 25 shows the h_{cm} curves. The non self adjoint finite differences curve is slightly below the finite element curve, and a likely reason is that the grid resolution isn't high enough, suggesting that finite differences require finer grids. Variable rainfall was also tested and the results can be seen in Figure 26. Clearly, the agreement is excellent for all rainfall durations. The h_{cm} curves are less accurate when for short rain durations (e.g 1 second) but become more accurate as the rain duration increases, as shown in Figure 27. Figures 29 and 30 compare the finite element solution with the self adjoint solution for h_m and h_{cm}. Now, the h_m is less accurate and the h_{cm} is more accurate. The same can be observed for variable rainfall in Figures 31 and 32.

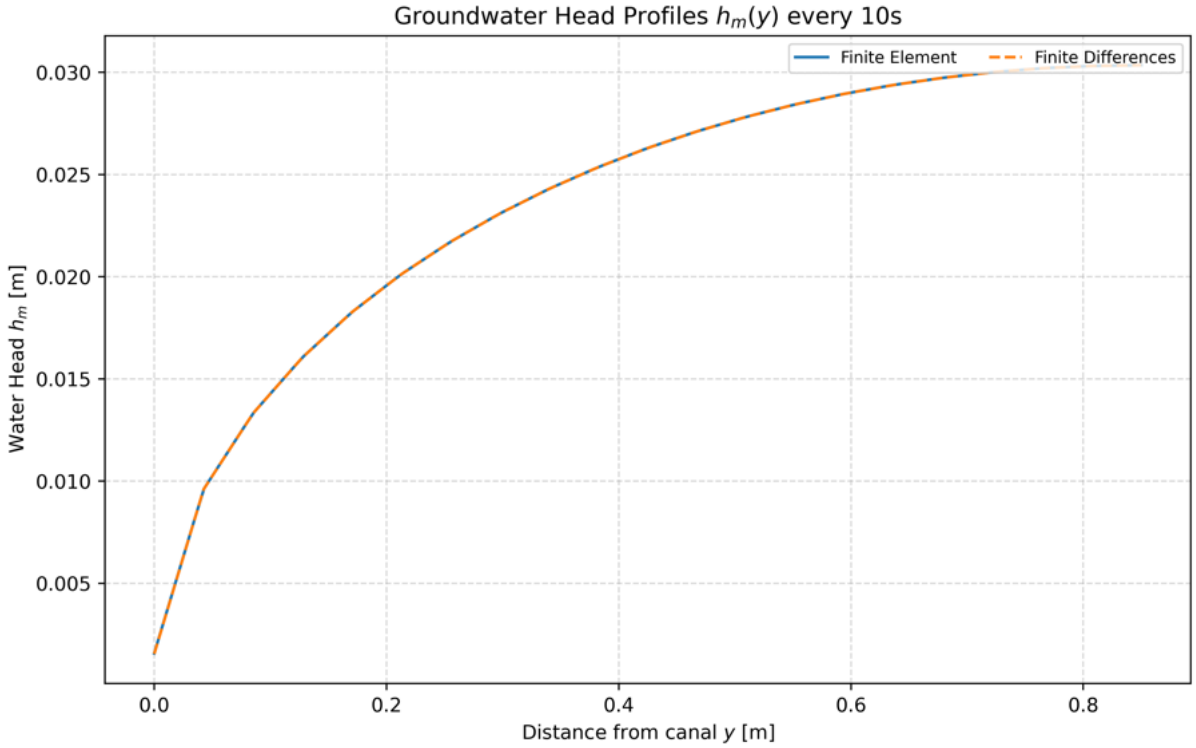

Figure 24: Non Self adjoint method:h_m at t=200. Comparison between finite element and finite differences solution. Note that rainfall is constant. Nx=20

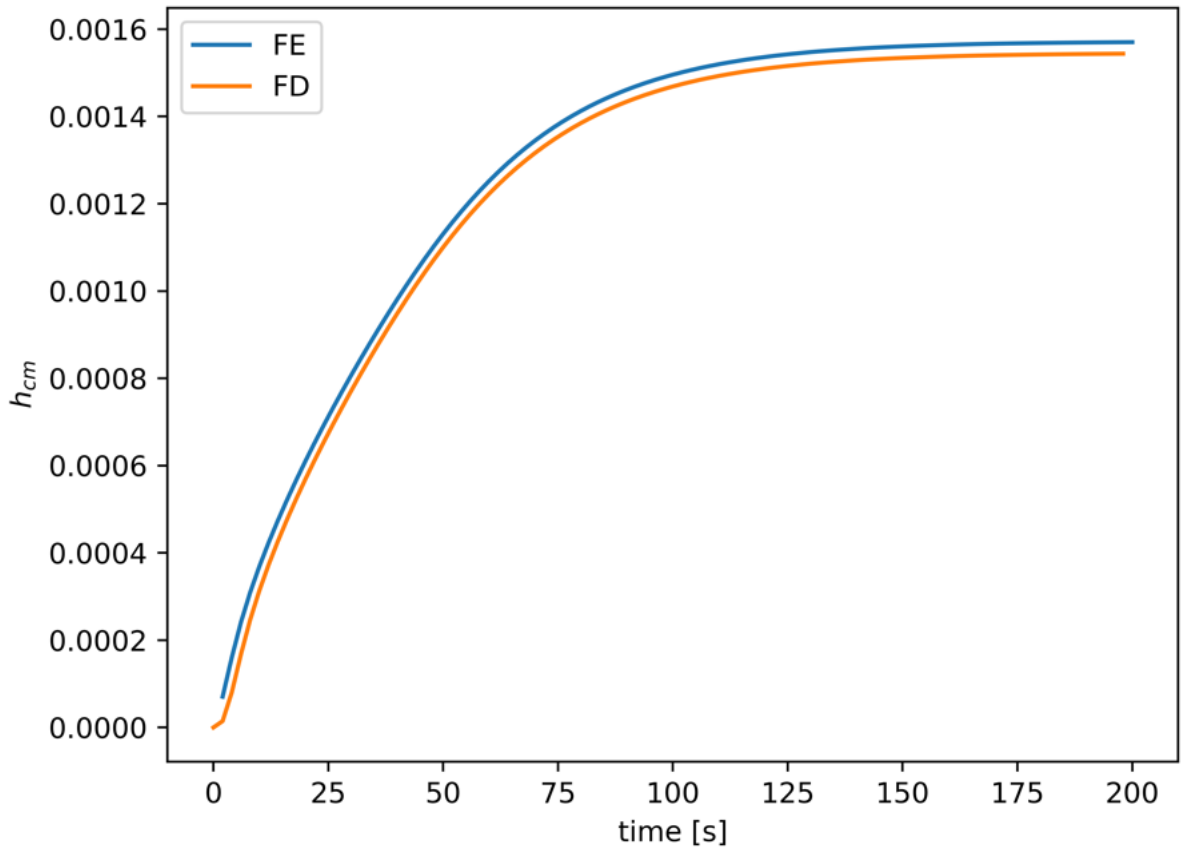

Figure 25: Non Self adjoint method: h_{cm} vs time. Finite element vs finite differences.Nx=20

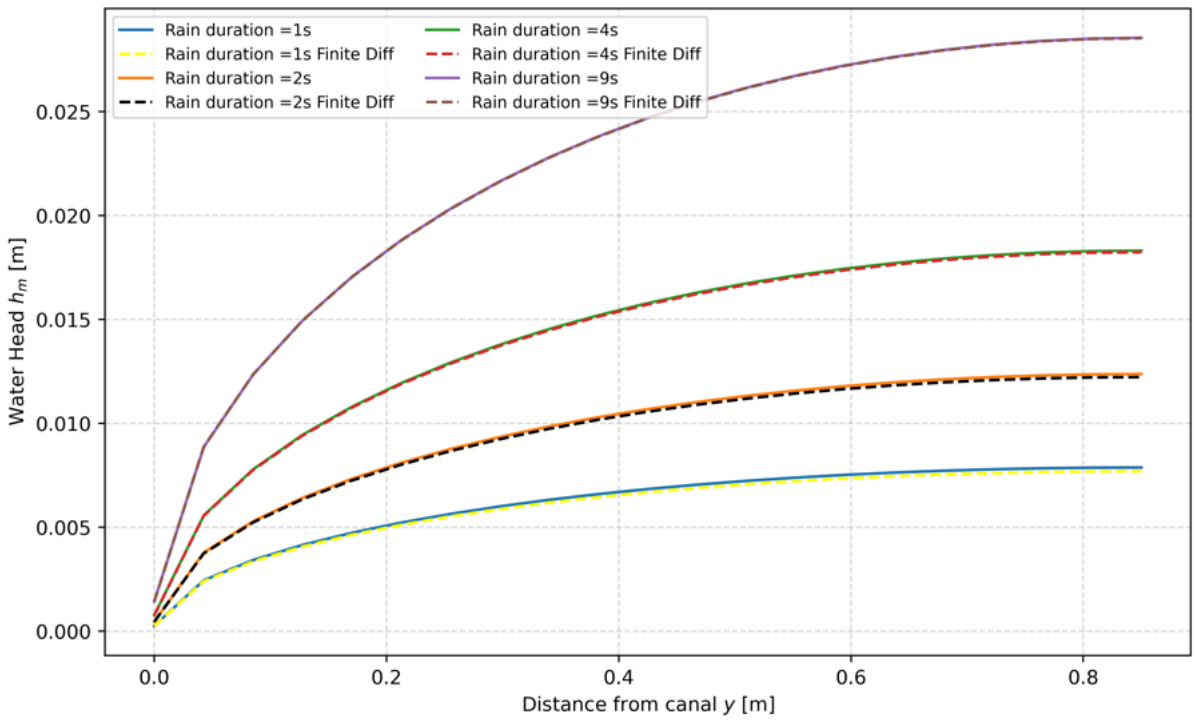

Figure 26: Non Self adjoint method: h_m for different rainfall durations. Solid lines indicate finite element solution and dotted lines indicate finite differences solution. Nx=20

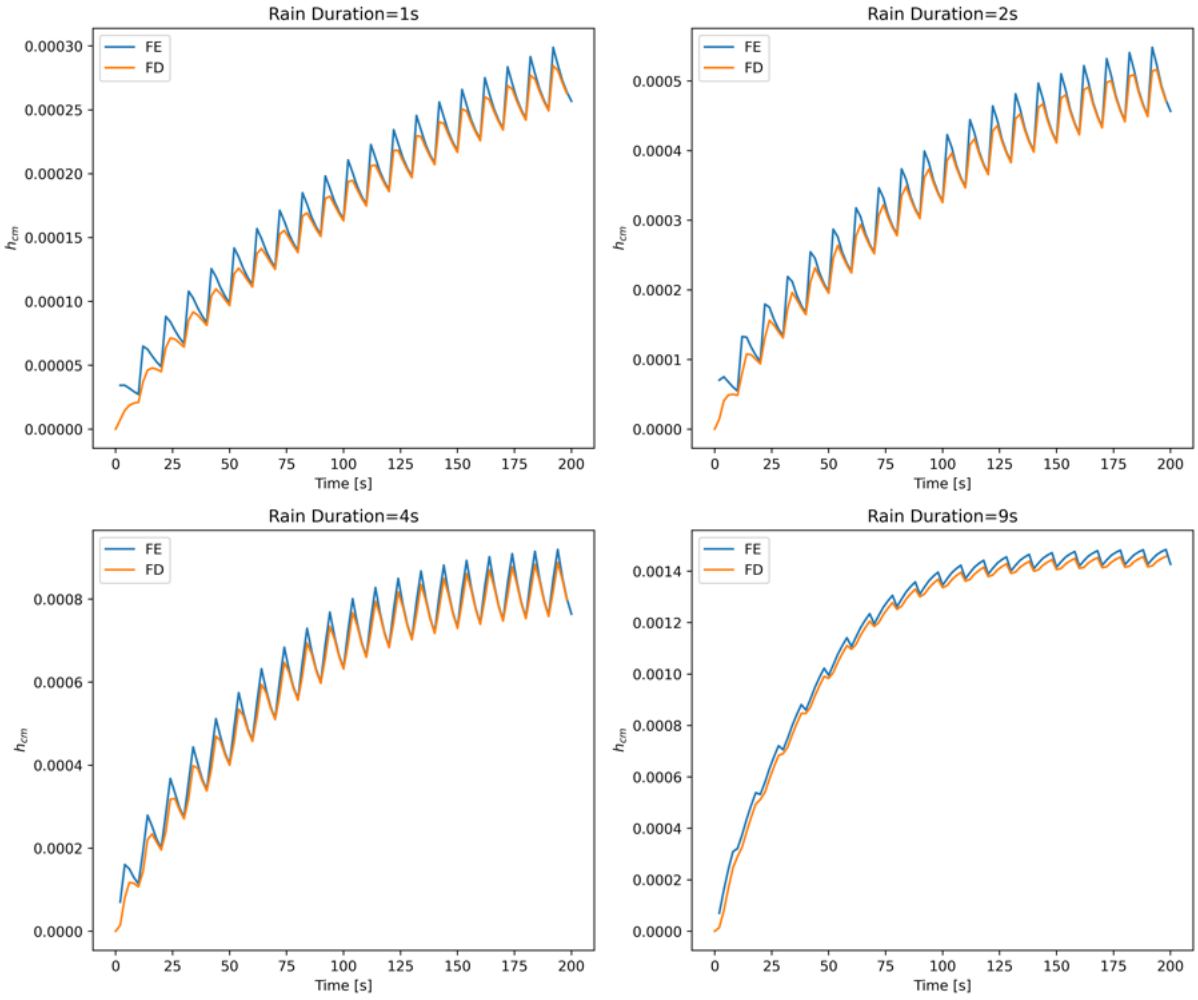

Figure 27: Non Self adjoint method: Comparison between finite element and finite differences solutions for h_{cm} when rainfall duration is varied.Nx=20

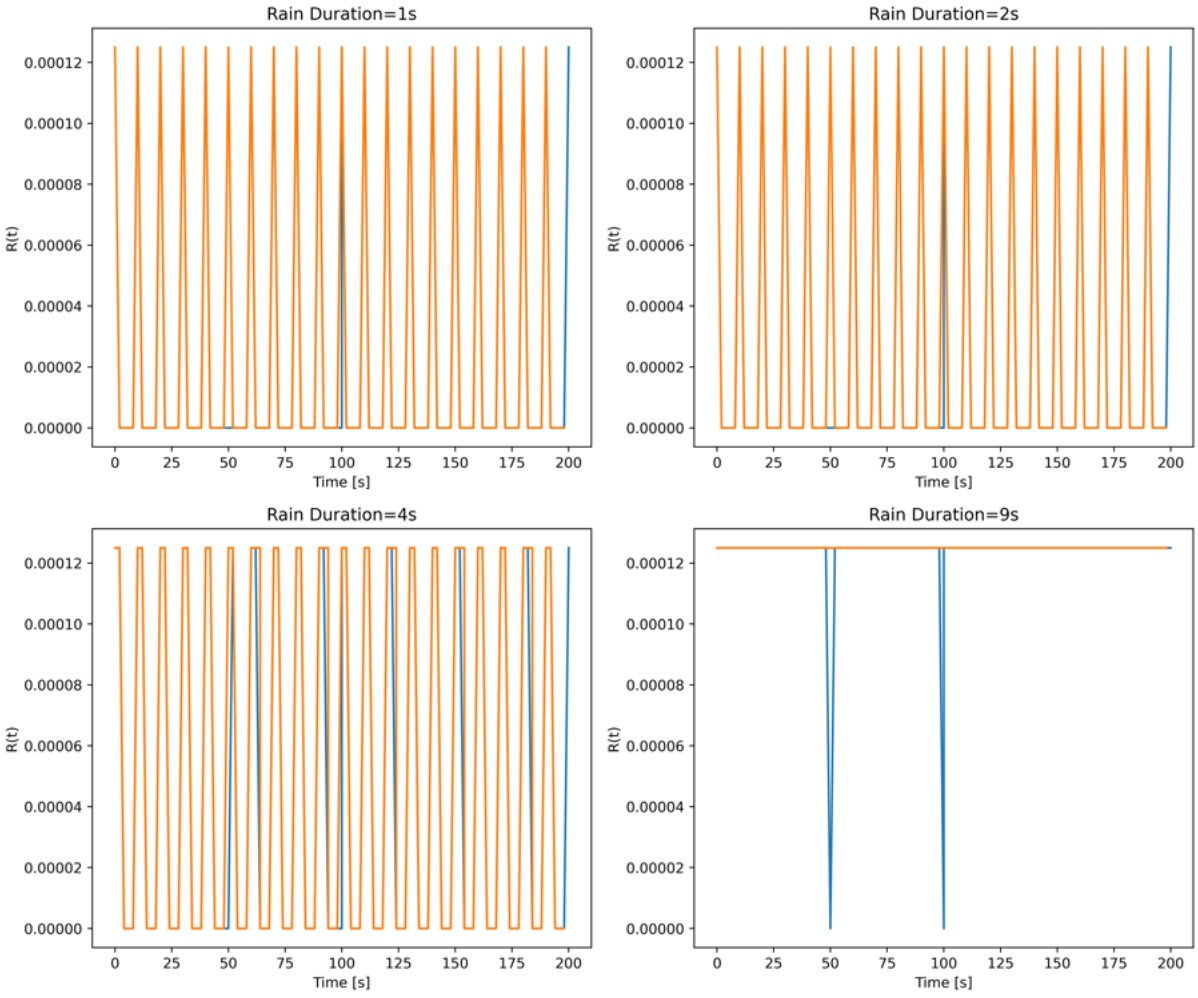

Figure 28: Non Self adjoint method: Rain over time. Orange indicates finite differences and blue lines indicate finite element.Nx=20

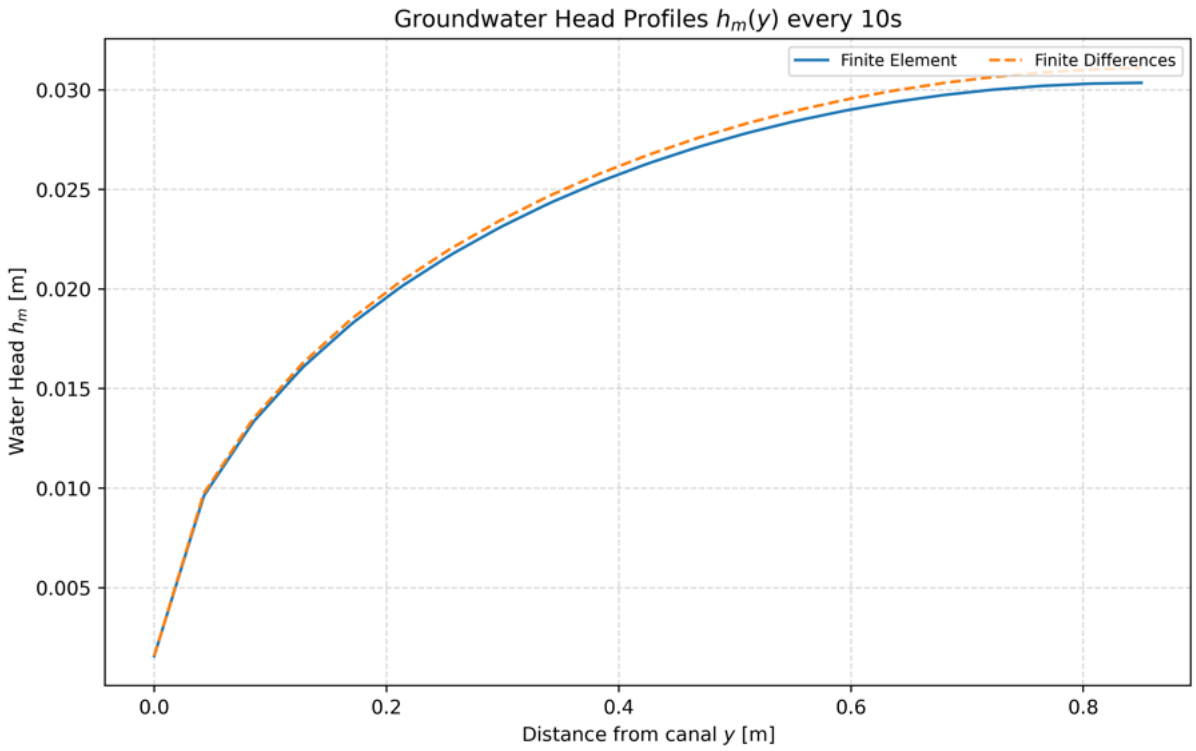

Figure 29: Self adjoint method: h_m at t=200 for self-adjoint method. Comparison between finite element and finite differences solution. Note that rainfall is constant. Nx=20.

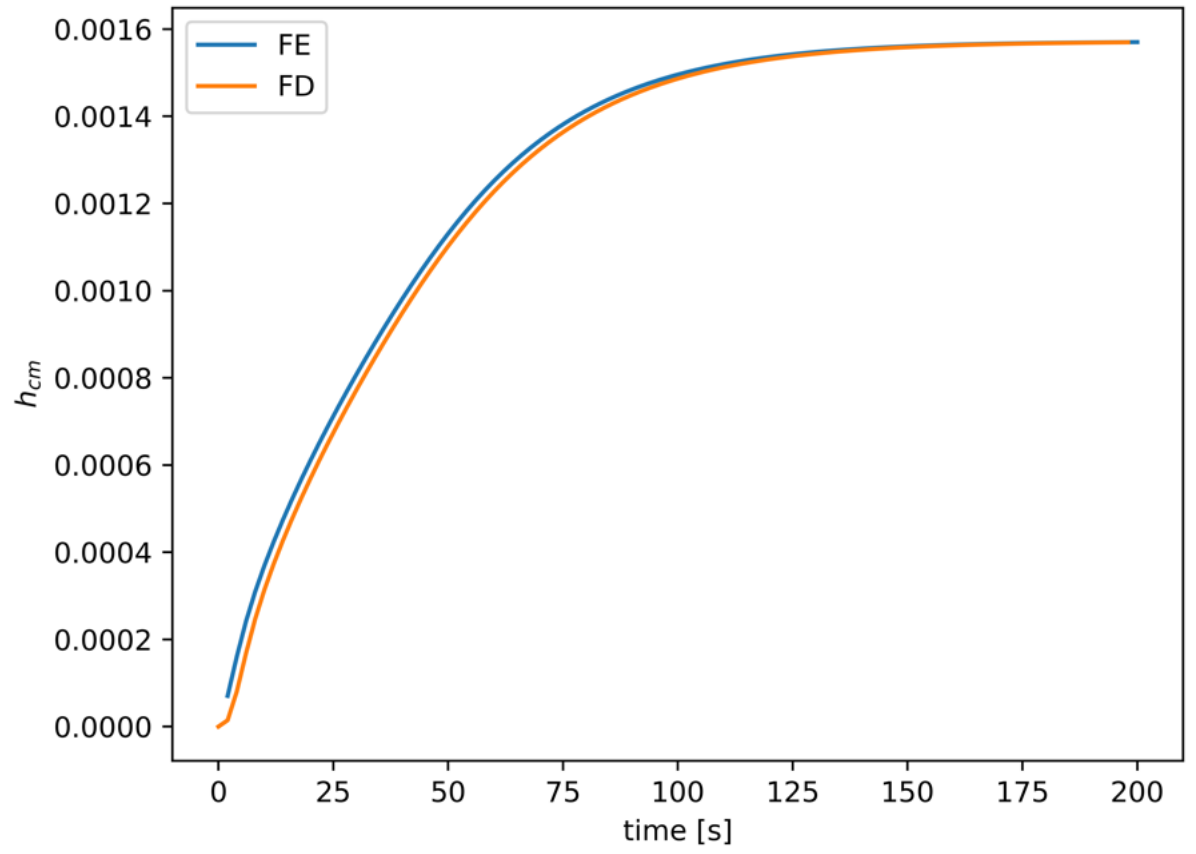

Figure 30: Self adjoint method: h_{cm} vs time. Finite element vs finite differences.Nx=20

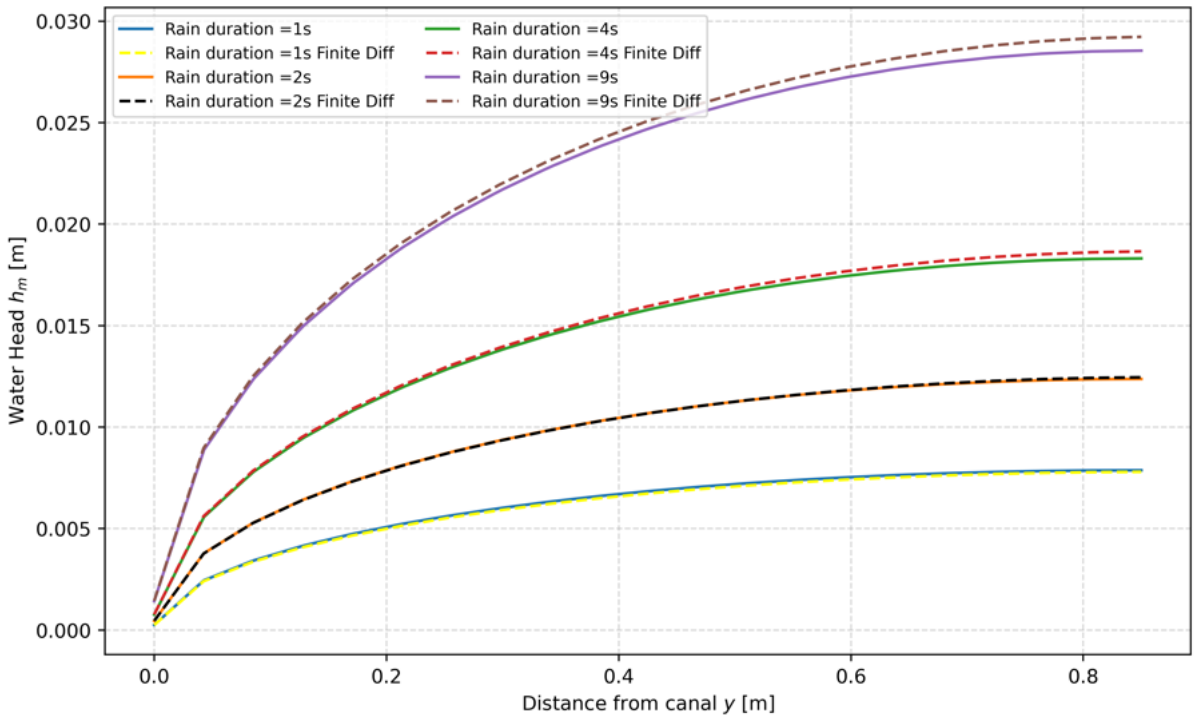

Figure 31: Self adjoint method:h_m for different rainfall durations. Solid lines indicate finite element solution and dotted lines indicate finite differences solution. Nx=20

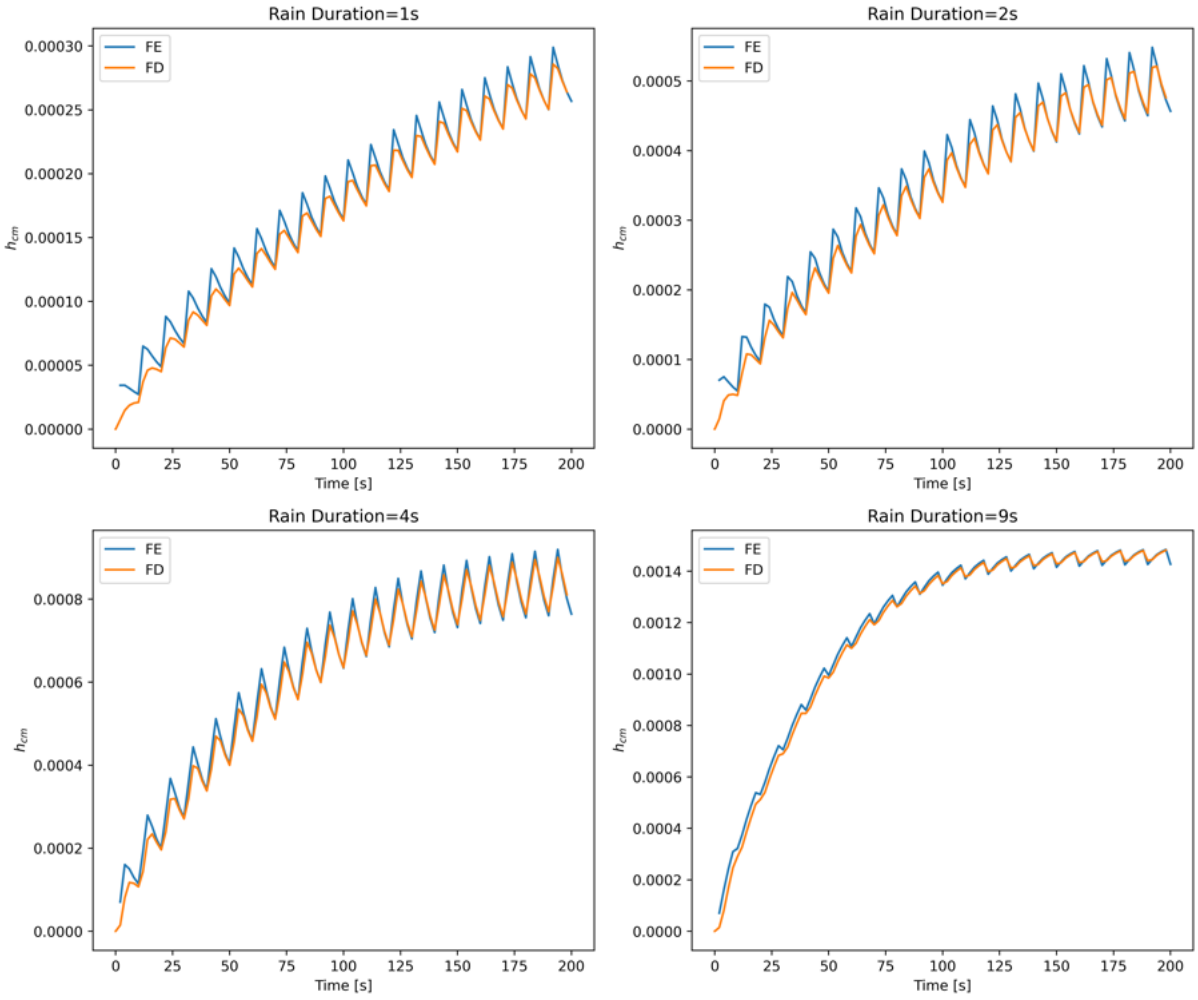

Figure 32: Self adjoint method: h_m for different rainfall durations. Solid lines indicate finite element solution and dotted lines indicate finite differences solution. Nx=20

5.3 Question 3

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

$$\frac{M_{ij}h_j^{n+1} - M_{ij}h_j^n}{\Delta t} + \frac{L_c}{\sigma_e m_{por}}\frac{h_1^{n+1} - h_1^n}{\Delta t}\delta_{i1} = \frac{1}{2}\big(F^{n+1} + F^n\big) \tag{64}$$

where

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

Rearranging the equation gives:

$$M_{ij}h_j^{n+1} + \frac{L_c}{\sigma_e m_{por}}h_1^{n+1}\delta_{i1} - \frac{1}{2}\Delta t F^{n+1} = M_{ij}h_j^n + \frac{L_c}{\sigma_e m_{por}}h_1^n\delta_{i1} + \frac{1}{2}\Delta t F^n \tag{65}$$

$$\begin{aligned} M_{ij}h_j^{n+1} + \frac{L_c}{\sigma_e m_{por}}h_1^{n+1}\delta_{i1} - \frac{1}{2}\Delta t\, b_i^{n+1} + \frac{1}{2}\Delta t\,\frac{1}{\sigma_e m_{por}}\sqrt{g}\max\left(\frac{2}{3}h_1^{n+1}(t), 0\right)^{3/2}\delta_{i1} \\ = M_{ij}h_j^n + \frac{L_c}{\sigma_e m_{por}}h_1^n\delta_{i1} + \frac{1}{2}\Delta t\, b_i^n - \frac{1}{2}\Delta t\,\frac{1}{\sigma_e m_{por}}\sqrt{g}\max\left(\frac{2}{3}h_1^n(t), 0\right)^{3/2}\delta_{i1}. \end{aligned} \tag{66}$$

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. Also note that the code sets the tolerance to 1e-14. Figure 33 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 34 shows the residuals at different iterations. Again, the figure clearly shows that the CN scheme converged after 2 iterations. Figure 35 shows the water head profile for the CN scheme, which looks identical to the one where explicit Euler was used.

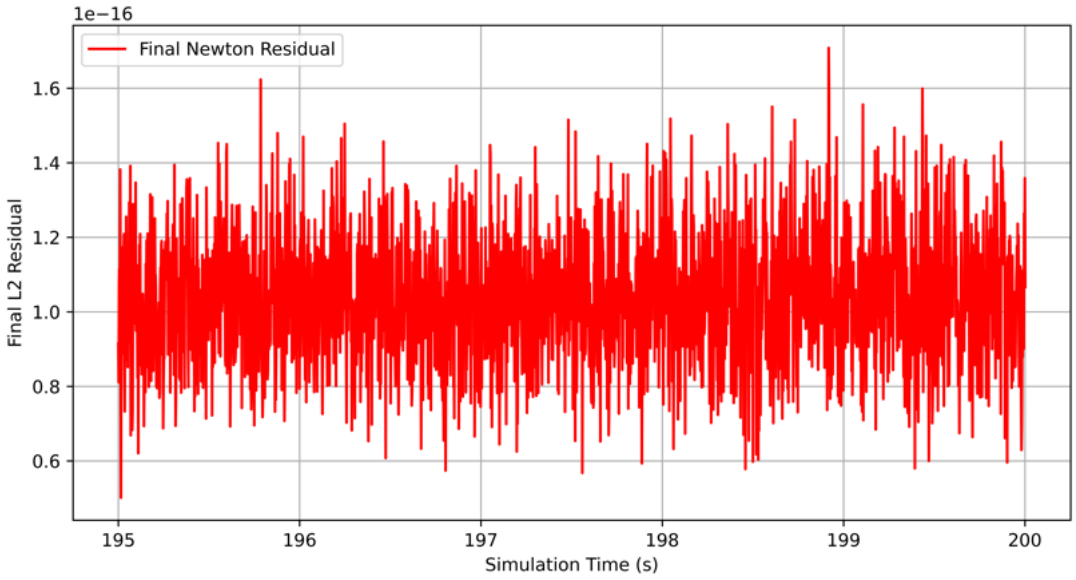

Figure 33: Residuals of the last iteration from t=195 to t=200

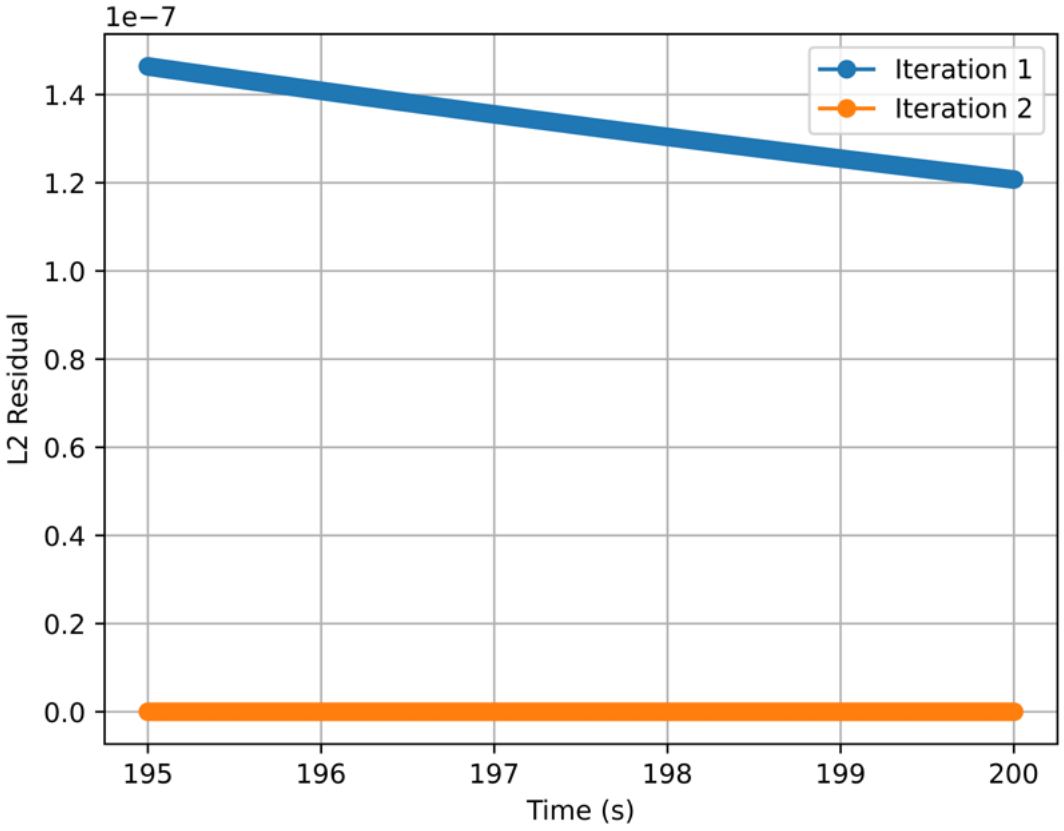

Figure 34: All iterations and their corresponding residuals from t=195 to t=200.

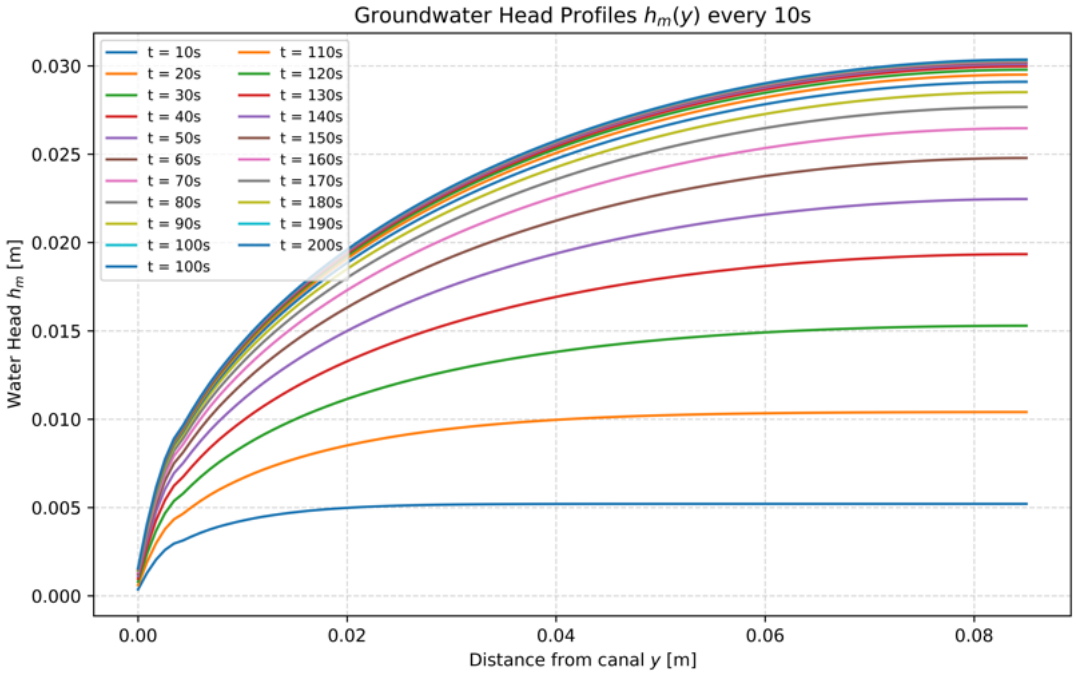

Figure 35: Water head over time when CN is used.

5.4 Question 4

Figure 36 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 37 shows the result when CFL is changed to 1.

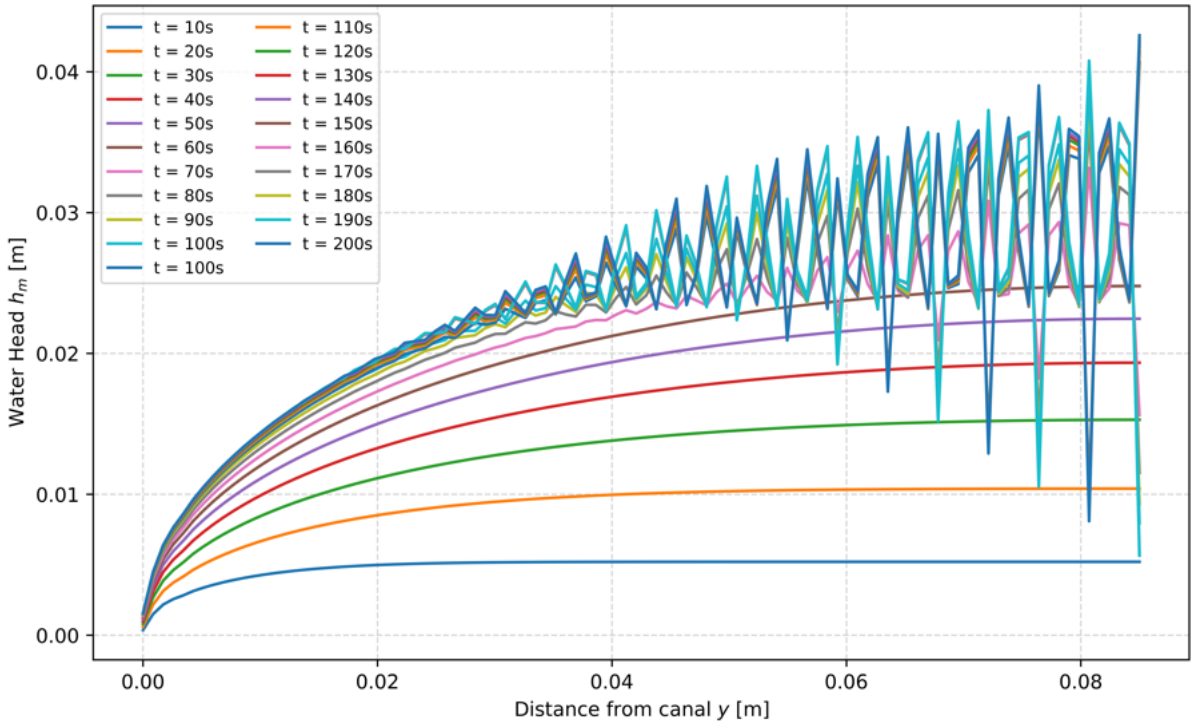

Figure 36: Water head over time with CG2 and explicit euler for CFL 2.3.

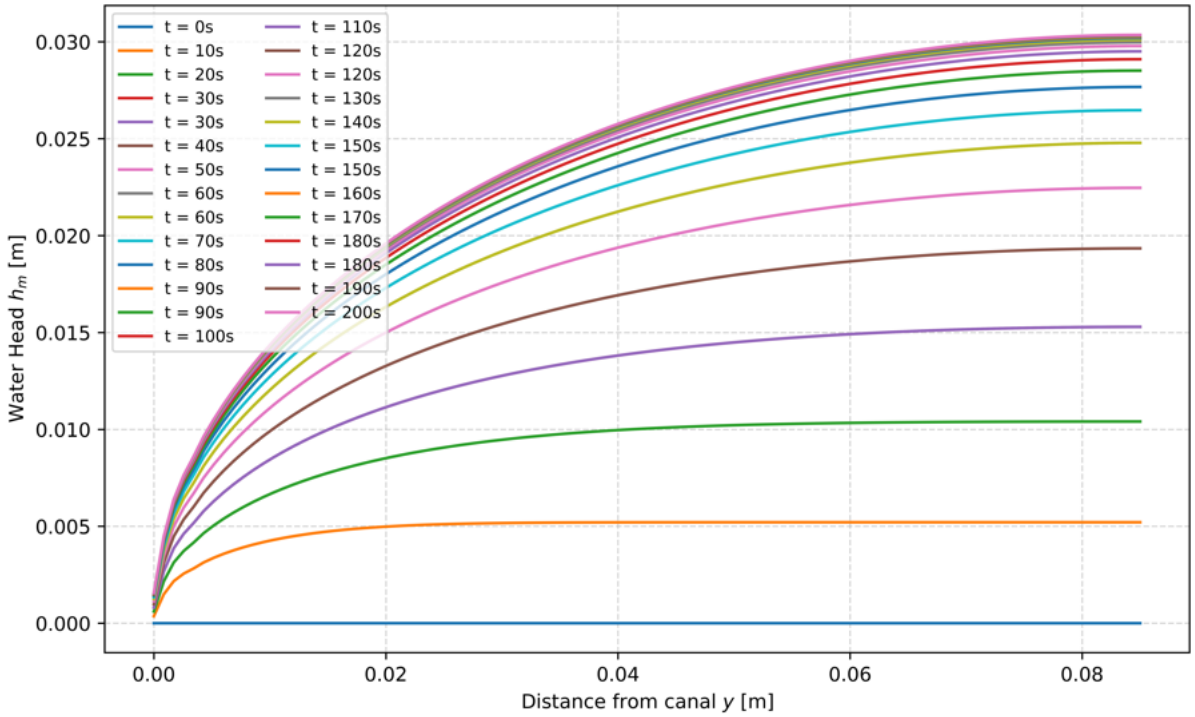

Figure 37: Water head over time with CG2 and explicit euler for CFL 1.