

Project Report for TMA4212 - Spring 2021

10070^a, 10049^a

^aDepartment of Mathematical Sciences, NTNU - Norwegian University of Science and Technology, NO-7491 Trondheim, Norway.

Abstract

In this project we have solved various partial differential equations using mainly the finite difference method but also the finite element method. For all the partial differential equations we have considered, our numerical solution converges to the analytical solution.

Contents

1.1 Problem 1 - The Poisson Equation with the Finite Difference Method	2
1.1.1 Mathematical Formulation	3
1.1.1.1 Analytical Solution	3
1.1.1.2 Norms and Errors	3
1.1.1.3 Difference Schemes	3
1.1.1.4 Computational Aspects	4
1.1.2 Numerical Results	5
1.2 Problem 2 - Heat Equation and Inviscid Burgers Equation	11
1.2.1 Mathematical Formulation	11
1.2.1.1 Difference Schemes	11
1.2.1.2 Manufactured Solution	12
1.2.1.3 Breakdown Time of the Inviscid Burgers Equation	12
1.2.1.4 Computational Aspects	12
1.2.2 Numerical Results	13
1.2.2.1 Heat Equation	13
1.2.2.2 Inviscid Burgers Equation	23
1.3 Problem 3 - The Laplace Equation	25
1.3.1 Mathematical Formulation	25
1.3.1.1 Analytical Solution using Separation of Variables	25
1.3.1.2 Difference Scheme	26
1.3.2 Numerical Results	26
1.4 Problem 4 - Korteweg-deVries Equation	29
1.4.1 Mathematical Formulation	29
1.4.1.1 Analytical Solution	29
1.4.1.2 Difference Schemes	29
1.4.1.3 Von Neumann Stability Analysis	29
1.4.1.4 Conservation of the L_2 Norm	31
1.4.1.5 Expected Convergence Rates	31
1.4.2 Numerical Results	32
1.4.2.1 KdV Equation	32

1.4.2.2 Conservation of the ℓ_2 -norm	35
1.5 Problem 5 - The Poisson Equation by Finite Element Methods	37
1.5.1 Mathematical Formulation	37
1.5.1.1 Assembly	37
1.5.1.2 Analytical Solutions	38
1.5.1.3 Numerical Implementation	39
1.5.2 Numerical Results	39
2.1 1D Wave Equation by FDM	44
2.1.1 Mathematical Formulation	44
2.1.1.1 Difference Scheme	44
2.1.1.2 Von Neumann Stability Analysis	44
2.1.1.3 Analytical Solution using Separation of Variables	45
2.1.2 Numerical Results	46
2.2 Wave Equation in Two Dimensions	48
2.2.1 Mathematical Formulation	48
2.2.1.1 Von Neumann Stability Analysis	48
2.2.1.2 Analytical Solution by Separation of Variables	49
2.2.2 Numerical results	50
2.2.2.1 2D Wave Equation	50
2.2.2.2 Computational time	52

Introduction

This report is the main assignment in the course TMA4212 - Numerical solution of differential equations by difference methods, and is based on [1] and [2]. The main topic is solving different partial differential equations (PDEs) by applying the finite difference method (FDM) as well as solving one PDE applying a finite element method (FEM) with the Galerkin approach.

The equations we will be considering in part 1 are the Poisson equation, the heat equation, the inviscid Burgers equation, the Korteweg-deVries and the Laplace equation, which will be solved by FDM. We will also be solving the Poisson equation by applying FEM. In part 2, we will be solving the wave equation in both one and two dimensions using FDM. The equations will be solved for different initial conditions and boundary conditions including homogenous and inhomogenous Dirichlet boundary conditions and Neumann boundary conditions.

We will mainly be studying how the numerical solutions converge to the analytical or manufactured solutions, attempting to verify the theoretical converge rates. Our main method of convergence verification will be convergence plots based on the discrete ℓ_2 - and the continuous L_2 norm, and combined plots of the numerical and analytical solutions to see how they behave. Both uniform and adaptive mesh refinement will be applied in the attempt to improve numerical accuracy.

Project Part 1

1.1. Problem 1 - The Poisson Equation with the Finite Difference Method

In problem 1 we consider the Poisson equation

$$u_{xx} = f(x), \quad (1.1.1)$$

with the following sets of Dirichlet and Neumann boundary conditions,

$$u(0) = \alpha, \quad u(1) = \sigma, \quad (1.1.2)$$

$$u(0) = 1, \quad u(1) = 1, \quad (1.1.3)$$

$$u_x(0) = 0, \quad u_x(1) = \frac{1}{2}, \quad \text{and} \quad (1.1.4)$$

$$u(x) = e^{\frac{1}{\epsilon}(x-\frac{1}{2})^2} \quad \text{in} \quad \Omega = (0, 1). \quad (1.1.5)$$

1.1.1. Mathematical Formulation

1.1.1.1. Analytical Solution

To begin with, the analytical solution for the Poisson equation with Dirichlet boundary conditions and

$$f(x) = \cos(2\pi x) + x, \quad \alpha = \sigma = 0 \quad (1.1.6)$$

was found to be

$$u(x) = -\frac{1}{4\pi^2} \cos(2\pi x) + \frac{1}{6}x^3 - \frac{1}{2}x + \frac{1}{4\pi^2}. \quad (1.1.7)$$

This will be utilized to see how the numerical solution calculated by FDM converges to the analytical solution. This will be done by considering the relative errors in terms of increasing number of grid points in space, M .

1.1.1.2. Norms and Errors

The discrete ℓ_2 -norm and the continuous L_2 -norm are defined for a vector $\mathbf{V} \in \mathbb{R}^N$ and a function $v(x) \in L_2(\Omega)$ as

$$\|\mathbf{V}\|_2 = \sqrt{\frac{1}{N} \sum_{i=1}^N (V_i)^2} \quad (1.1.8)$$

for the ℓ_2 -norm, and

$$\|v(x)\|_2 = \sqrt{\int_{\Omega} v^2(x) d\Omega} \quad (1.1.9)$$

for the L_2 -norm. The relative errors $e_{\ell_2}^r$ and $e_{L_2}^r$ are defined as

$$e_{\ell_2}^r = \frac{\|\mathbf{u} - \mathbf{U}\|_2}{\|\mathbf{u}\|_2}, \quad (1.1.10)$$

$$e_{L_2}^r = \frac{\|u(x) - U(x)\|_2}{\|u(x)\|_2}. \quad (1.1.11)$$

1.1.1.3. Difference Schemes

For the Poisson equation the central difference scheme

$$U_m'' = \frac{1}{h^2}(U_{m-1} - 2U_m + U_{m+1}) + \mathcal{O}(h^2), \quad (1.1.12)$$

was applied together with the second order backward difference scheme for Neumann boundary conditions,

$$u'_{M+1} = \sigma = -\frac{1}{2h}U_{M-1} + \frac{2}{h}U_M - \frac{3}{2h}U_{M+1} + \mathcal{O}(h^2). \quad (1.1.13)$$

From (1.1.12) and (1.1.13) we can expect the convergence to be of order $\mathcal{O}(h^2)$ [3].

Further, the finite difference method was applied for different boundary conditions, first a Dirichlet boundary condition and then a Neumann boundary condition.

At last we apply the condition (1.1.5) with $\epsilon = 0.01$ on the Poisson equation. For a first order method, a forward difference scheme,

$$U_m'' = \frac{1}{h^2}(U_m - 2U_{m+1} + U_{m+2}) + \mathcal{O}(h), \quad (1.1.14)$$

was applied. For second order methods, the central difference scheme as in (1.1.12) was applied.

Moreover, adaptive mesh refinement with Dirichlet boundary conditions was utilized by applying a difference scheme

$$U_m'' = aU_{m-2} + bU_{m-1} - (a+b+c)U_m + cU_{m+1}. \quad (1.1.15)$$

By inserting

$$\begin{aligned} a &= \frac{2(h_m - h_{m-1})}{(h_{m-2} + h_{m-1})(h_{m-2} + h_{m-1} + h_m)h_{m-2}}, \\ b &= \frac{2(h_{m-2} + h_{m-1} + h_m)}{h_{m-2}h_{m-1}(h_{m-1} + h_m)}, \quad \text{and} \\ c &= \frac{2(h_{m-2} + 2h_{m-1})}{h_m(h_{m-1} + h_m)(h_{m-2} + h_{m-1} + h_m)} \end{aligned} \quad (1.1.16)$$

for $h_m = x_{m+1} - x_m$, we obtain the difference scheme

$$U_m'' = \frac{2}{h_{m-1} \cdot h_m} \left(\frac{h_m}{h_{m-1} + h_m} U_{m-1} - U_m + \frac{h_{m-1}}{h_{m-1} + h_m} U_{m+1} \right) \quad (1.1.17)$$

which we will apply for the adaptive refinement process.

1.1.1.4. Computational Aspects

The finite difference method was applied on the Poisson equation on an equidistributed grid where

$$x_0 = 0, x_1 = \frac{1}{M+1}, \dots, x_M = \frac{M}{M+1}, x_{M+1} = 1.$$

A second order central difference in space was applied to approximate the second order derivative, resulting in the linear system $A_h \mathbf{U} = \mathbf{F}$ where

$$A_h = \frac{1}{h^2} \begin{bmatrix} -2 & 1 & 0 & \dots & 0 \\ 1 & -2 & 1 & \ddots & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & \ddots & 1 & -2 & 1 \\ 0 & \dots & \frac{-h}{2} & 2h & \frac{-3h}{2} \end{bmatrix}, \mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_M \\ U_{M+1} \end{bmatrix}, \mathbf{F} = \begin{bmatrix} f(x_1) - \alpha_h^2 \\ f(x_2) \\ \vdots \\ f(x_M) \\ \sigma \end{bmatrix}.$$

With Neumann boundary conditions on both sides, we encounter a problem as the A_h matrix becomes singular and the system does not turn out to have a unique solution. We approximate

$$u_x(0) \approx \frac{U_1 - U_{-1}}{2h} = \sigma_0 = 0, \quad \text{and} \quad u_x(1) \approx \frac{U_{M+2} - U_M}{2h} = \frac{1}{2}.$$

Combining this with the second order central difference in space gives the linear system $A_h \mathbf{U} = \mathbf{F}$, where

$$A_h = \frac{1}{h^2} \begin{bmatrix} -h & h & 0 & 0 & 0 \\ 1 & -2 & 1 & \ddots & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & \ddots & 1 & -2 & 1 \\ 0 & \dots & 0 & h & -h \end{bmatrix}, \mathbf{U} = \begin{bmatrix} U_0 \\ \vdots \\ U_{M+1} \end{bmatrix}, \mathbf{F} = \begin{bmatrix} \frac{h}{2}f_0 \\ f_1 \\ \vdots \\ f_M \\ f_{M+1} \frac{h}{2} - \frac{1}{2} \end{bmatrix}.$$

It can be proven that this system admits a solution if and only if $\mathbf{F} \in Range(A_h)$. Further, one can prove that $\mathbf{F} \in Range(A_h)$ if and only if $\mathbf{F} \perp Ker(A_h^T)$ [4]. One solution to this is to utilize a Neumann boundary condition only on one side and thereby avoid the issue of A_h being a singular matrix. Another approach is to add an additional condition to the solution vector, for instance fixing the solution at a single node to some constant value. This will again make the system admit a unique solution.

In the last item of problem 1, the scheme (1.1.15) was applied on the Poisson equation. Both first and second order methods were applied, and both the discrete ℓ_2 - and continuous L_2 -error were utilized to evaluate the error. In order to be able to utilize the continuous L_2 -norm, cubic interpolation was used, as the numerical solution originally obtained is discrete.

1.1.2. Numerical Results

From figure 1 it is evident that the analytical solution is well approximated numerically even for small M . For $M = 11$, the numerical solution has nearly overlapped the analytical solution, and for $M = 23$ they are observed to be practically the same. From the convergence plot in figure 2, it can be observed that the relative errors coincide when the solution has converged. In addition the convergence is of order less than $\mathcal{O}(h^2)$, as expected from (1.1.12).

We see from figure 3 that the numerical solution tends to the manufactured solution as the number of gridpoints M gets larger than approximately 23. From the convergence plot in 4, we see that we have convergence of order close to $\mathcal{O}(h^2)$, as expected from (1.1.13).

From figure 5 we see that the numerical solution tends to the analytical solution for $u(x)$ in the domain in (1.1.5) for $M = 95$ and larger. For low values of M , the shape of the numerical solution looks strange, but it seems to converge for higher M . From the convergence plot in figure 6 we see that the convergence is parallel to $\mathcal{O}(h^2)$ for gridsize greater than 10. This fits well with our expectation.

From figure 7 we see that the numerical solution tends to the analytical solution of $u(x)$ for values of M from 47 and larger, i.e. values of M smaller than 47 when using forward difference discretization for the same problem. The convergence plot in figure 8 is however rather similar to the convergence plot we get when using forward difference discretization, with convergence parallel to $\mathcal{O}(h^2)$ as expected.

From the plots of the numerical and analytical solutions in figure 9 we see that the numerical solution tends to the manufactured solution (1.1.5) for values of M larger than approximately 39. The plot of convergence from figure 10 is however not linear, but this is expected since the refinement is not uniform over the grid.

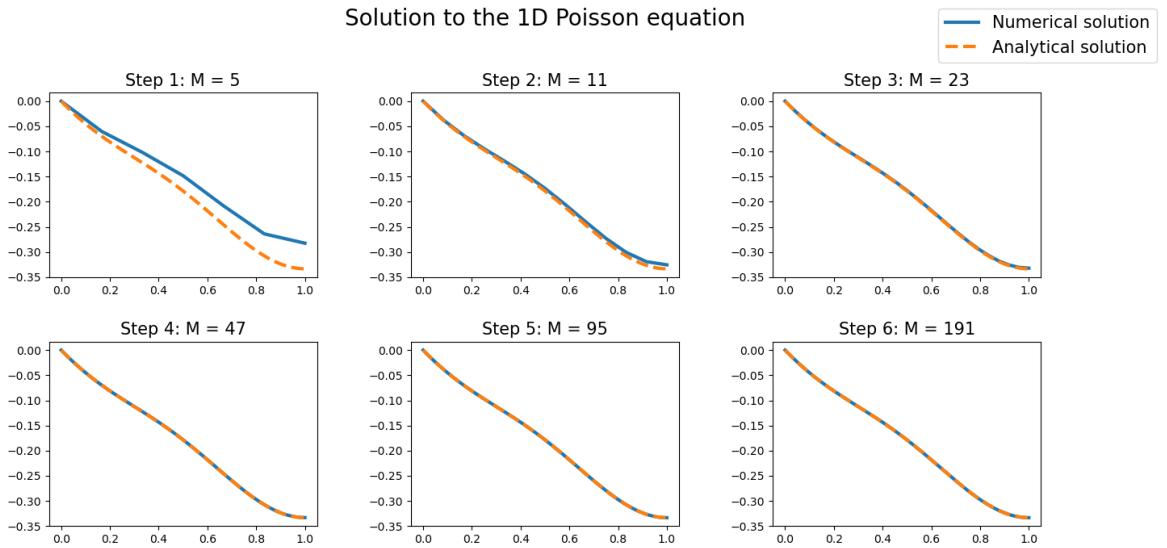


Figure 1: Plots of solutions to the Poisson equation with Dirichlet boundary conditions (1.1.2) for different stepsizes.

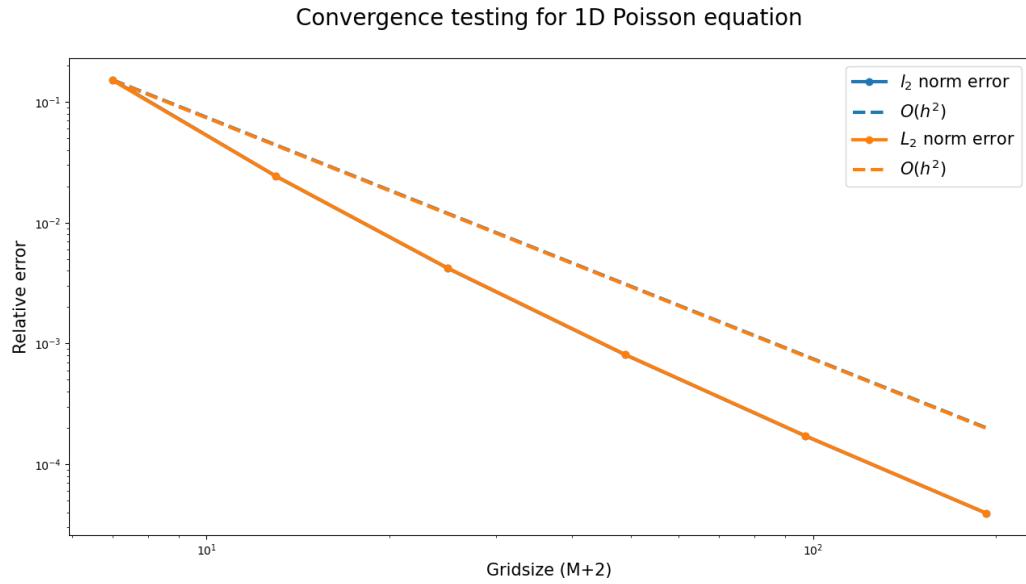


Figure 2: Convergence plot for the Poisson equation with Dirichlet boundary conditions (1.1.2).

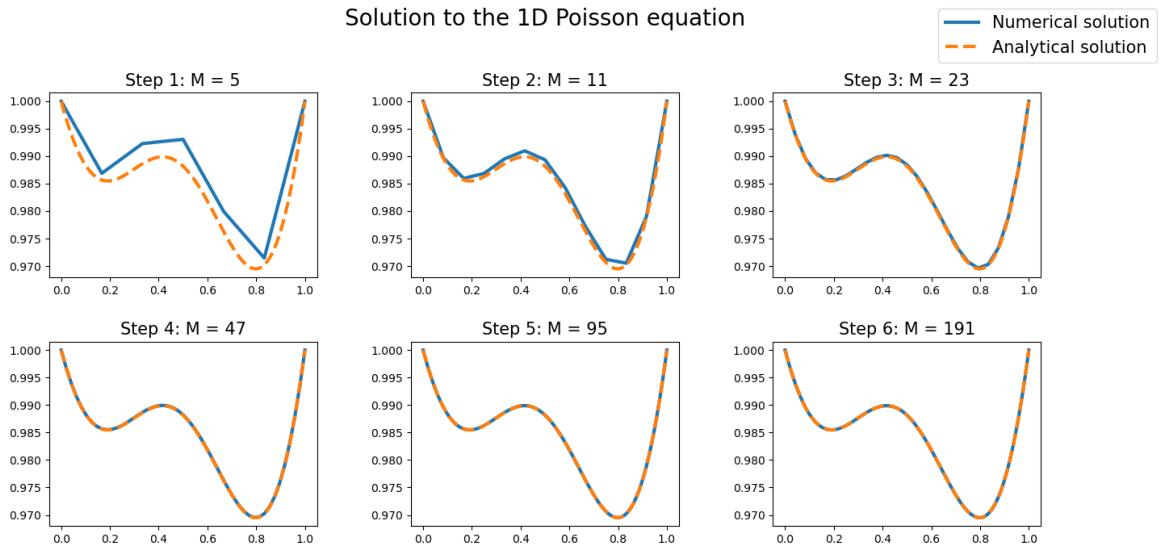


Figure 3: Plots of solutions to the Poisson equation with Dirichlet boundary conditions (1.1.3) for different stepsizes.

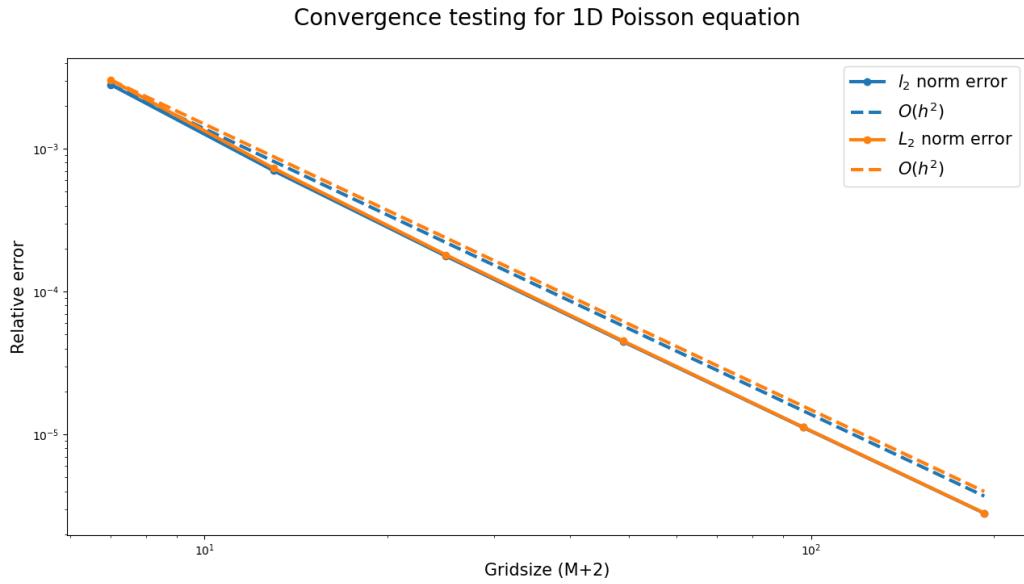


Figure 4: Convergence plot for the Poisson equation with Dirichlet boundary conditions (1.1.3).

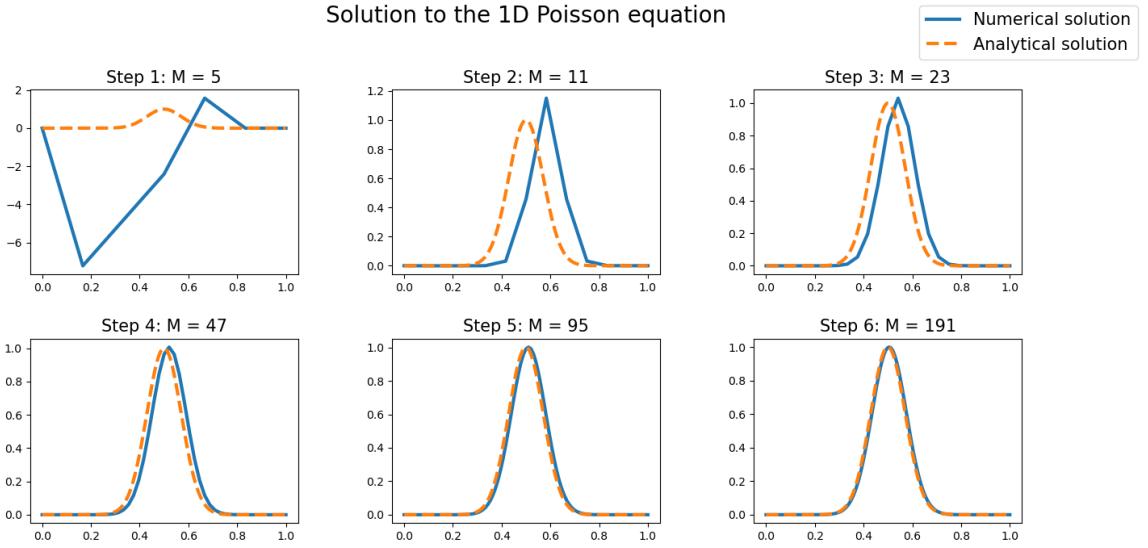


Figure 5: Plot of solution to the Poisson equation with manufactured solution (1.1.5) using uniform mesh refinement (UMR) with a forward difference scheme.

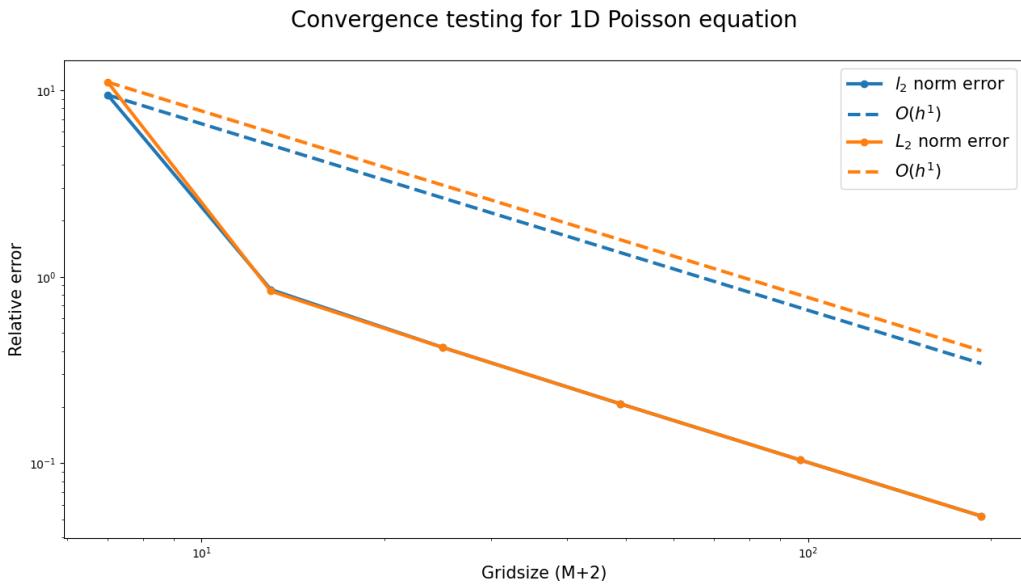


Figure 6: Convergence plot for the Poisson equation with manufactured solution (1.1.5) using UMR with a forward difference scheme.

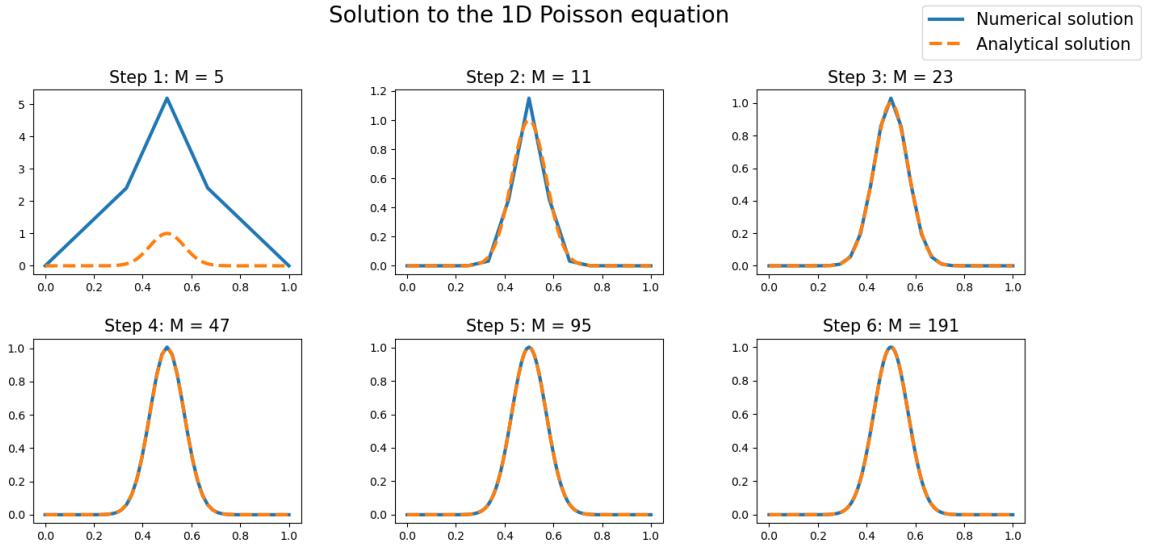


Figure 7: Plot of solution for the Poisson equation with manufactured solution (1.1.5) using UMR with a central difference scheme.

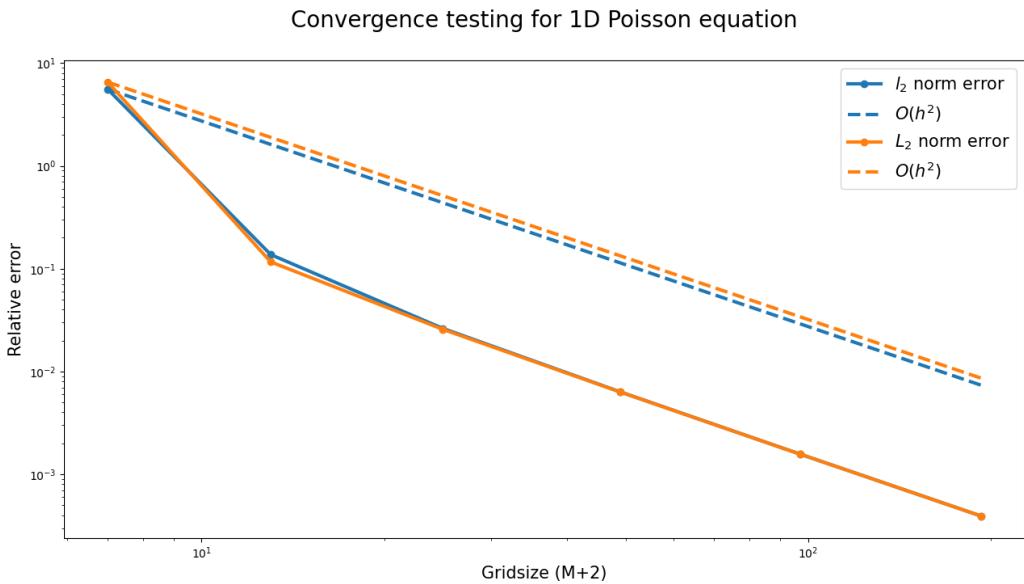


Figure 8: Convergence plot for the Poisson equation with manufactured solution (1.1.5) using UMR with a central difference scheme.

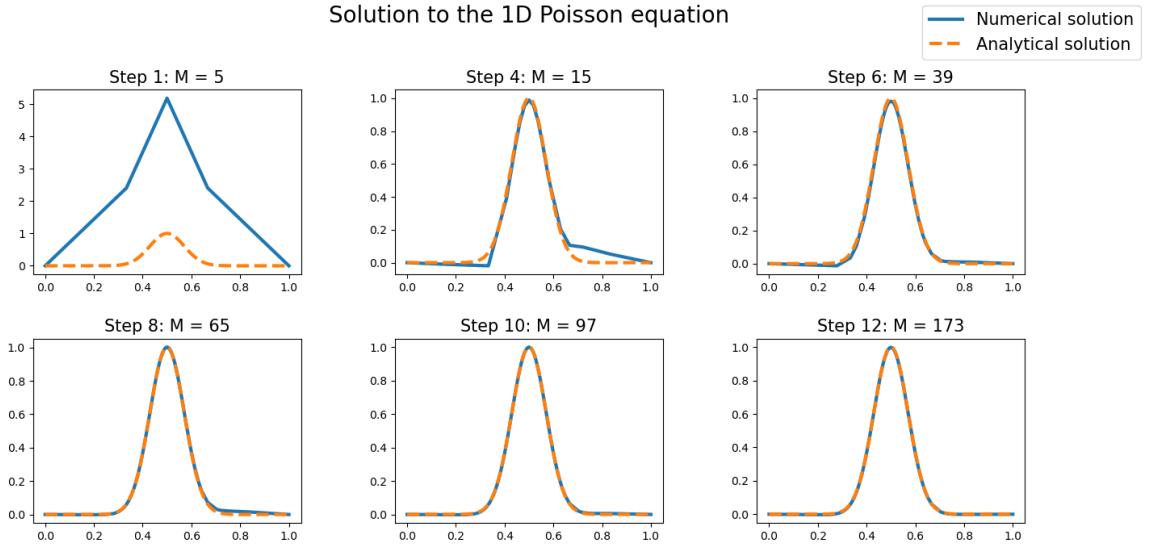


Figure 9: Plots for solution of the Poisson equation with manufactured solution (1.1.5) using adaptive mesh refinement (AMR).

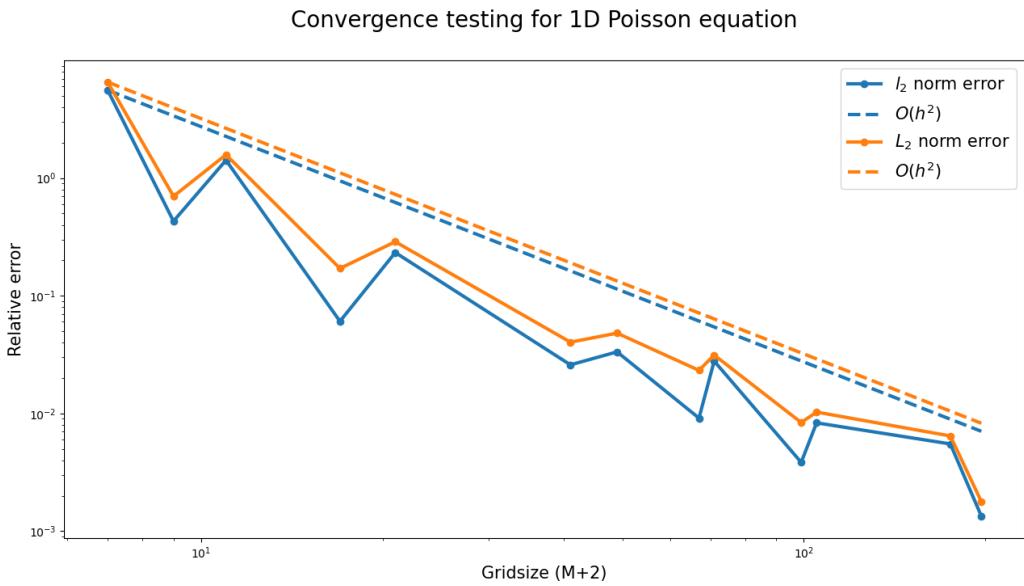


Figure 10: Convergence plot for the Poisson equation with manufactured solution (1.1.5) using AMR.

1.2. Problem 2 - Heat Equation and Inviscid Burgers Equation

In this problem we will consider the heat equation

$$u_t = u_{xx}, \quad (1.2.1)$$

first in the domain

$$x \in [0, 1] , \quad t > 0, \quad (1.2.2)$$

with Neumann boundary conditions and an initial condition given by

$$u_x(0, t) = u_x(1, t) = 0, \quad u(x, 0) = 2\pi x - \sin(2\pi x), \quad (1.2.3)$$

and then on the domain

$$\Omega(x, t) ; \quad x \in [0, 1] , \quad t \in [0, T] \quad (1.2.4)$$

with Dirichlet boundary conditions.

At last we will consider the inviscid Burgers equation

$$u_t = -uu_x \quad (1.2.5)$$

on

$$x \in [0, 1] , \quad t > 0, \quad (1.2.6)$$

with homogenous Dirichlet boundary conditions and initial condition

$$u(0, t) = u(1, t) = 0 , \quad u(x, 0) = e^{(-400(x-1/2)^2)}. \quad (1.2.7)$$

1.2.1. Mathematical Formulation

1.2.1.1. Difference Schemes

Firstly, we applied the backward Euler method for the heat equation which gives the difference scheme

$$U_m^{n+1} - r(U_{m-1}^{n+1} - 2U_m^{n+1} + U_{m+1}^{n+1}) = U_m^n + \mathcal{O}(kh^2 + k^2). \quad (1.2.8)$$

which is first order in time and second order in space [4]. In order to incorporate the Neumann boundary conditions (1.2.3) into our scheme to find the solution nodes at the boundary, we introduce fictitious nodes which need to be eliminated,

$$\sigma_0(t) = \frac{\partial u}{\partial x}(0, t) \approx \frac{U(h, t) - U(-h, t)}{2h}, \quad (1.2.9)$$

leading to

$$U_0^{n+1} - r(U_{-1}^{n+1} - 2U_0^{n+1} + U_1^{n+1}) = U_0^n, \quad (1.2.10)$$

implying

$$(1 + 2r)U_0^{n+1} - r(2U_1^{n+1} + 2h\sigma_0^{n+1}) = U_0^n. \quad (1.2.11)$$

By substitution, similar reasoning for $U(1, t)$ gives

$$(1 + 2r)U_{M+1}^{n+1} - r(2U_M^{n+1} + 2h\sigma_{M+1}^{n+1}) = U_{M+1}^n , \quad \sigma_{M+1}(t) = \frac{\partial u}{\partial x}(1, t). \quad (1.2.12)$$

Then we applied the Crank-Nicolson method, a second method in both space and time, on the heat equation, which gives

$$U_m^{n+1} - \frac{r}{2}(U_{m-1}^{n+1} - 2U_m^{n+1} + U_{m+1}^{n+1}) = U_m^n + \frac{r}{2}(U_{m-1}^n - 2U_m^n + 2U_{m+1}^n) + O(k^3 + kh^2). \quad (1.2.13)$$

[4] Again, we utilize the Neumann boundary conditions (1.2.3) and $\sigma_0(t)$ and $\sigma_{M+1}(t)$ defined respectively in (1.2.9) and (1.2.12) to solve for the boundary nodes in a similar manner as for the backward Euler scheme,

$$(1 + r)U_0^{n+1} - rU_1^{n+1} + rh\sigma_0^{n+1} = (1 - r)U_0^n + rU_1^n - rh\sigma_0^n, \quad (1.2.14)$$

and

$$(1 + r)U_{M+1}^{n+1} - rU_M^{n+1} + rh\sigma_{M+1}^{n+1} = (1 - r)U_{M+1}^n + rU_M^n - rh\sigma_{M+1}^n. \quad (1.2.15)$$

This gives a matrix system $AU^{n+1} = DU^n + \mathbf{b}$, where the right hand side is solved first, and then the equation $AU^{n+1} = \mathbf{c}$.

1.2.1.2. Manufactured Solution

The manufactured solution of the heat equation on the domain (1.2.4) we used was

$$u(x, t) = \frac{1}{\sqrt{4\pi(t+1)}} \exp\left(-\frac{(x-\frac{1}{2})^2}{4(t+1)}\right), \quad (1.2.16)$$

with boundary conditions

$$u(0, t) = u(1, t) = \frac{1}{\sqrt{4\pi(t+1)}} \exp\left(-\frac{1}{16(t+1)}\right) \quad (1.2.17)$$

and initial condition

$$u(x, 0) = \frac{1}{\sqrt{4\pi}} \exp\left(-\frac{(x-\frac{1}{2})^2}{4}\right). \quad (1.2.18)$$

1.2.1.3. Breakdown Time of the Inviscid Burgers Equation

For the inviscid Burgers equation, the method of characteristics was applied to show where the solution breaks down, with the characteristic curves defined by

$$\frac{d\alpha}{dt} = u(\alpha, t) = u(\xi, 0) = F(\xi), \quad (1.2.19)$$

where ξ denotes the α -intercept of the characteristic curve. Direct integration gives the expression

$$\alpha = F(\xi)t + \xi \quad (1.2.20)$$

which has been plotted for an array of ξ values. We begin to see the characteristic curves crossing each other at time t^* . This is found from [5] to be

$$t^* = -\frac{1}{\min\{F'(\alpha)\}}. \quad (1.2.21)$$

1.2.1.4. Computational Aspects

For the refinement process for the heat equation, we did refinement in space and time, increasing the number of grid points M in space and N in time, decreasing the grid spacings $h = 1/M$ and $k = 1/N$. We also did refinement where $k = h$, doubling the number of grid points in both space and time for each iteration, and refinement where $r = kh^2$ was held constant, i.e $N = M^2$ where M is doubled each iteration.

For the inviscid Burgers equation, $u_t = uu_\alpha$ is semi-discretized by setting $\dot{v} \approx \dot{u}$ and writing

$$\dot{v} = F(t, v) \approx -uu_\alpha = -\frac{1}{2}\partial_\alpha u^2. \quad (1.2.22)$$

On vector form with indices m we then find the right hand side

$$F_m = -\frac{1}{4h}(v_{m+1}^2 - v_{m-1}^2), \quad (1.2.23)$$

where central difference is applied in the α -space. From here a tridiagonal matrix $A = \text{tridiag}(1, 0, 1)$ was constructed

$$A = \begin{bmatrix} 0 & 1 & 0 & & 0 \\ 1 & 0 & 1 & \ddots & 0 \\ \ddots & \ddots & \ddots & \ddots & \\ 0 & \ddots & 1 & 0 & 1 \\ 0 & \dots & 0 & 1 & 0 \end{bmatrix}.$$

With Dirichlet-conditions (1.2.7) at indices $m = 0$ and $m = M + 1$, the equation (1.2.24) was then solved for $m = 1, \dots, M$ using `scipy.integrate.solve_ivp()` [6] evaluating at steps $t = k \cdot m$

$$\dot{v}_m = F_m = -\frac{1}{4h} A v_m^2. \quad (1.2.24)$$

1.2.2. Numerical Results

1.2.2.1. Heat Equation

For the refinement process, we expect first order convergence in time and second order convergence in space for the backward Euler scheme (1.2.8), and second order convergence in both space and time for the Crank Nicolson scheme (1.2.13).

Looking at figure 11, we see that the numerical solution does not match perfectly with the manufactured solution even for $N = 703$ gridpoints. From the convergence plot in figure 12 we see that the relative error is approximately parallel to convergence order $\mathcal{O}(h^2)$ for approximately $M > 10^2$, as expected.

We see from figure 13 that the numerical solution tends to the manufactured solution when N reaches approximately 175. From the convergence plot 14, we see that it converges parallel to $\mathcal{O}(h^2)$ for $M \geq 10^2$, as expected.

In the figures 15 and 20 we see that the numerical solution seems to coincide well with the manufactured solution for relatively low gridsizes N , both when applying the backward Euler and the Crank-Nicolson methods.

From the figures 17-19 we see that for refinements $h, k, k = h$ and kh^2 respectively, the numerical solution found by backward Euler discretization converges with ratios $\mathcal{O}(h^2)$, $\mathcal{O}(h)$, $\mathcal{O}(\sqrt{h})$ and $\mathcal{O}(h^{\frac{2}{3}})$ respectively. [7].

From the figures 22-24 we see that for refinements $h, k, k = h$ and kh^2 respectively, the numerical solution found by Crank-Nicolson discretization converges with ratios $\mathcal{O}(h^2)$, $\mathcal{O}(h^2)$, $\mathcal{O}(h)$ and $\mathcal{O}(h)$ respectively. [7].

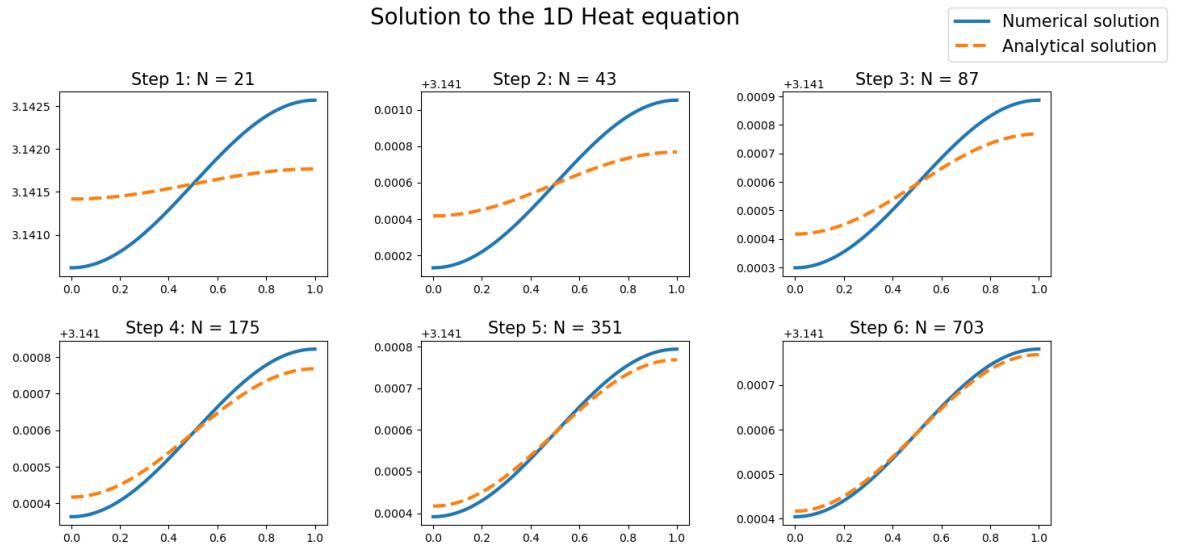


Figure 11: Solution to the heat equation with initial conditions and Neumann boundary conditions (1.2.3) using backward Euler difference scheme (1.2.8).

Convergence testing for the 1D Heat equation

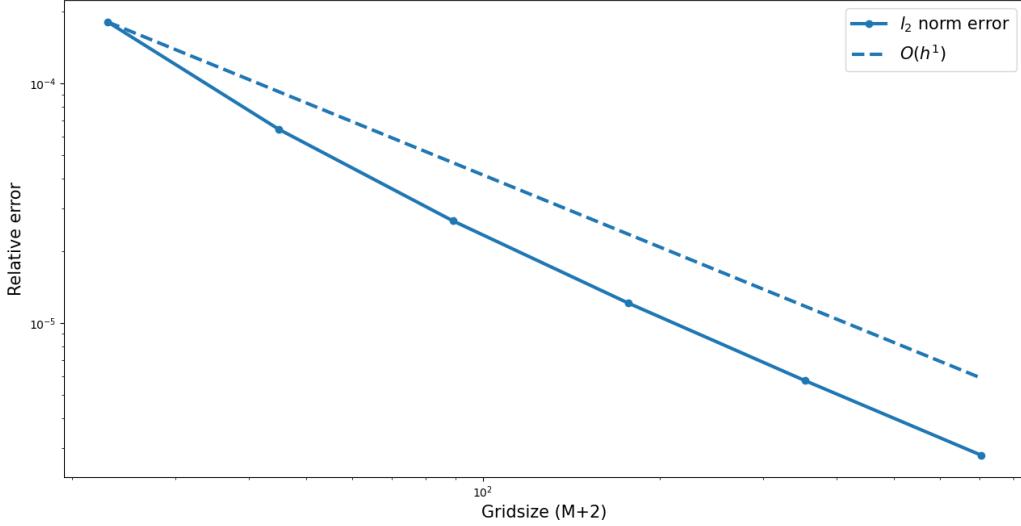


Figure 12: Convergence plot to the heat equation with initial conditions and Neumann boundary conditions (1.2.3) using backward Euler difference scheme (1.2.8).

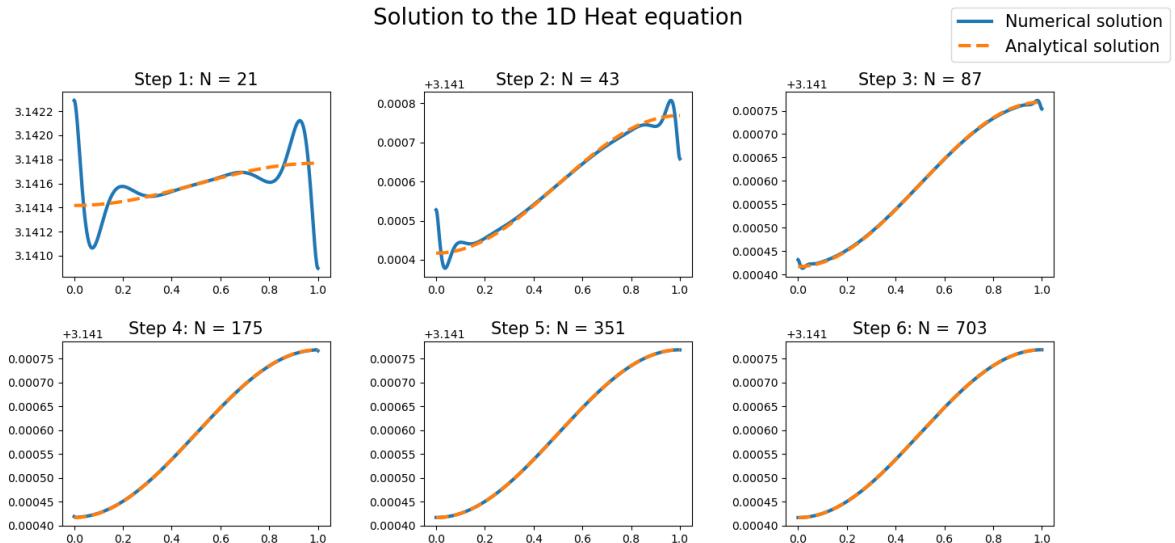


Figure 13: Solution to the heat equation with initial conditions and Neumann boundary conditions (1.2.3) using Crank-Nicolson difference scheme (1.2.13).

Convergence testing for the 1D Heat equation

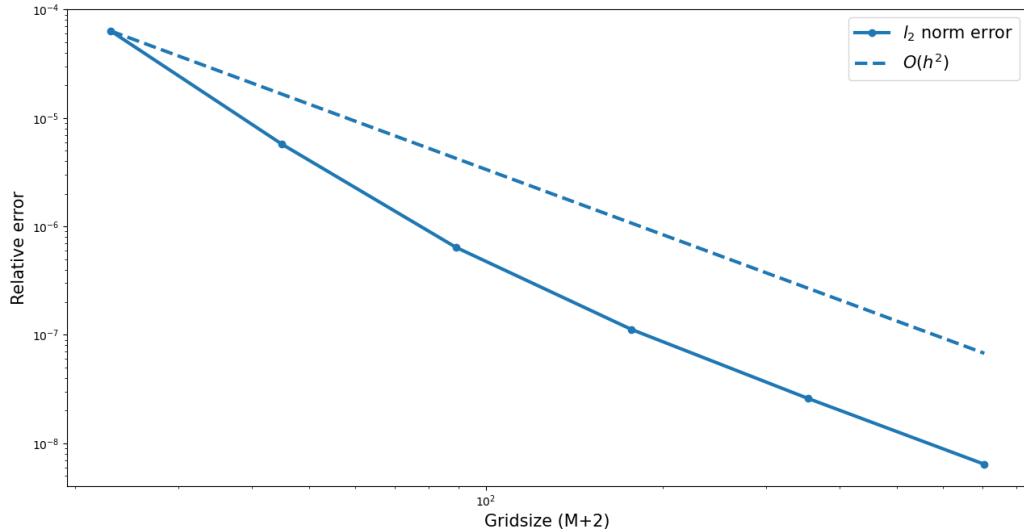


Figure 14: Convergence plot to the heat equation with initial conditions and Neumann boundary conditions (1.2.3) using the Crank-Nicolson difference scheme (1.2.13).

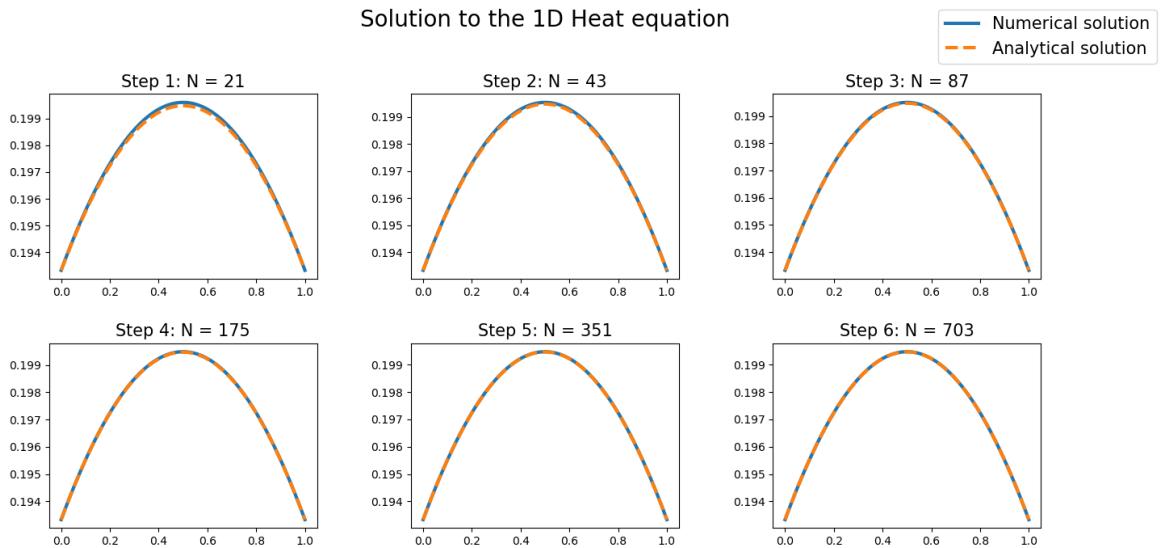


Figure 15: Solution to the heat equation in the domain (1.2.2) with Dirichlet boundary conditions using backward Euler discretization.

Convergence testing for the 1D Heat equation

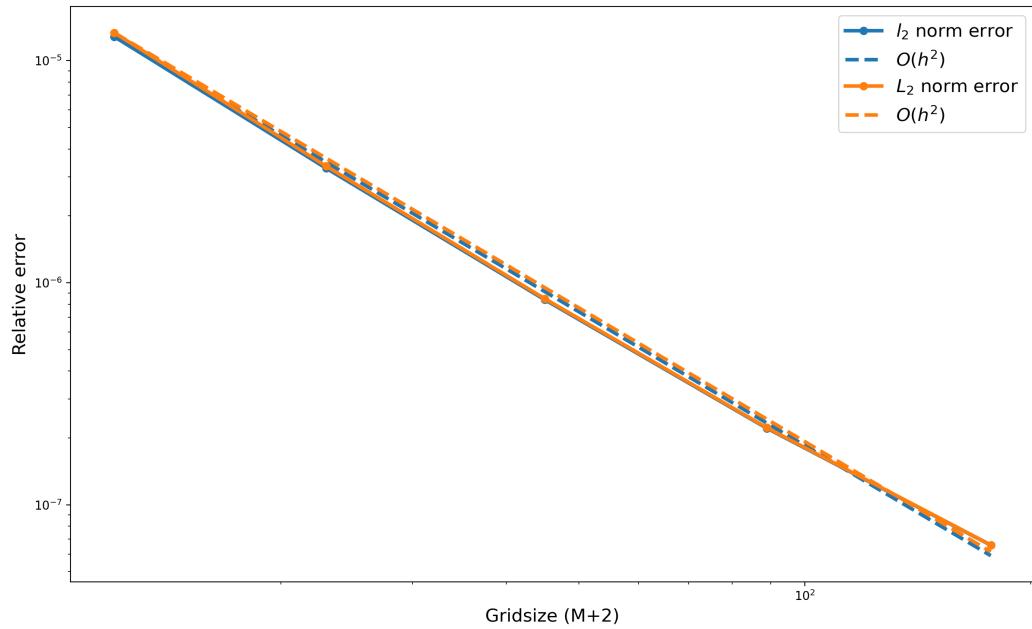


Figure 16: Convergence plot for the heat equation in the domain (1.2.2) with Dirichlet boundary conditions using backward Euler discretization with h -refinement.

Convergence testing for the 1D Heat equation

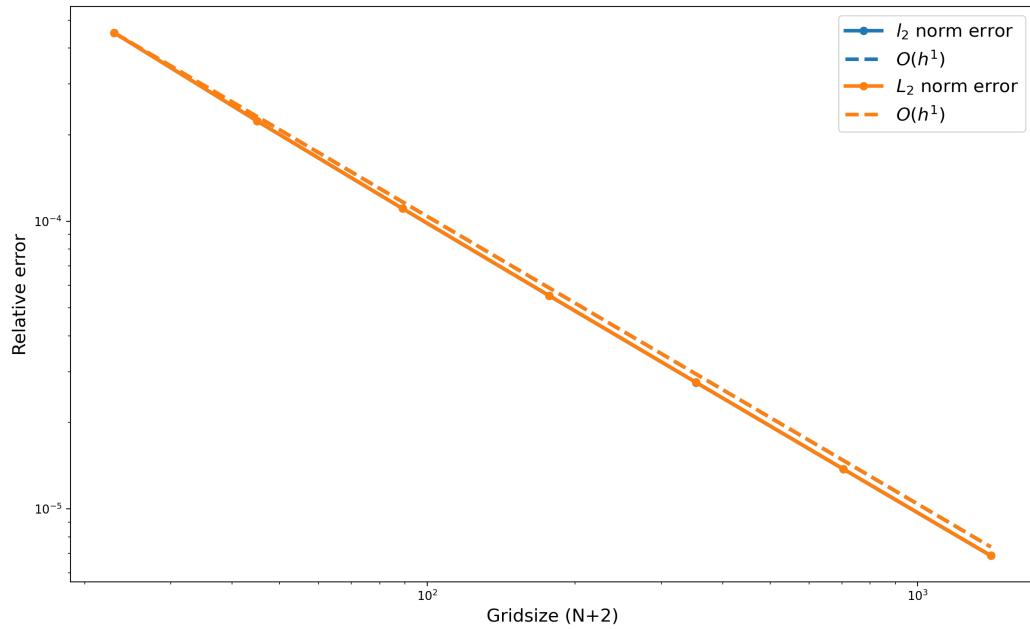


Figure 17: Convergence plot for the heat equation with Dirichlet boundary conditions using backward Euler discretization with k -refinement.

Convergence testing for the 1D Heat equation

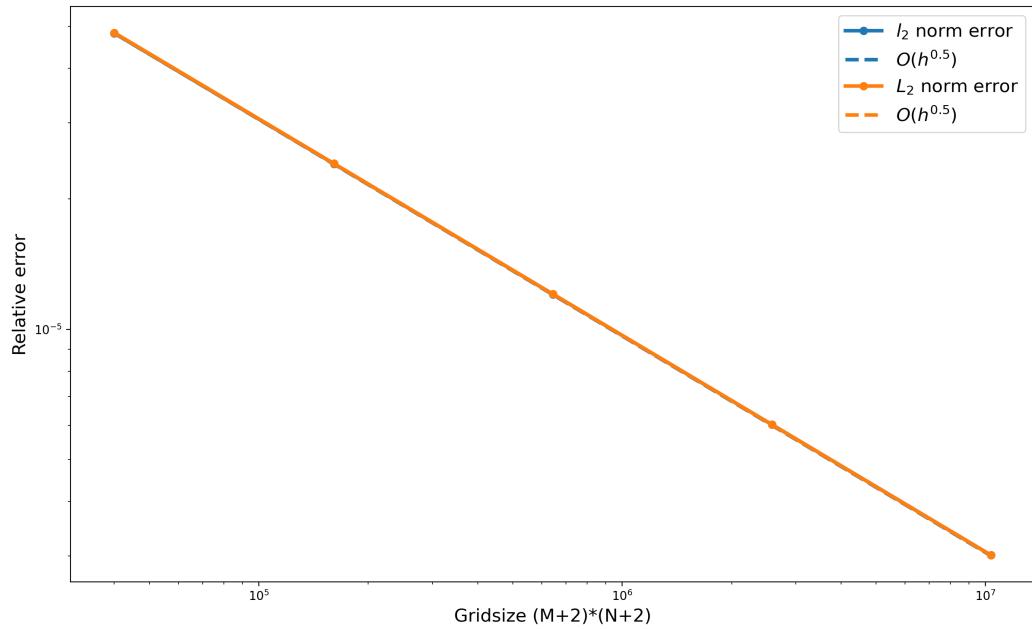


Figure 18: Convergence plot for the heat equation with Dirichlet boundary conditions using backward Euler discretization with $k = h$ -refinement.

Convergence testing for the 1D Heat equation

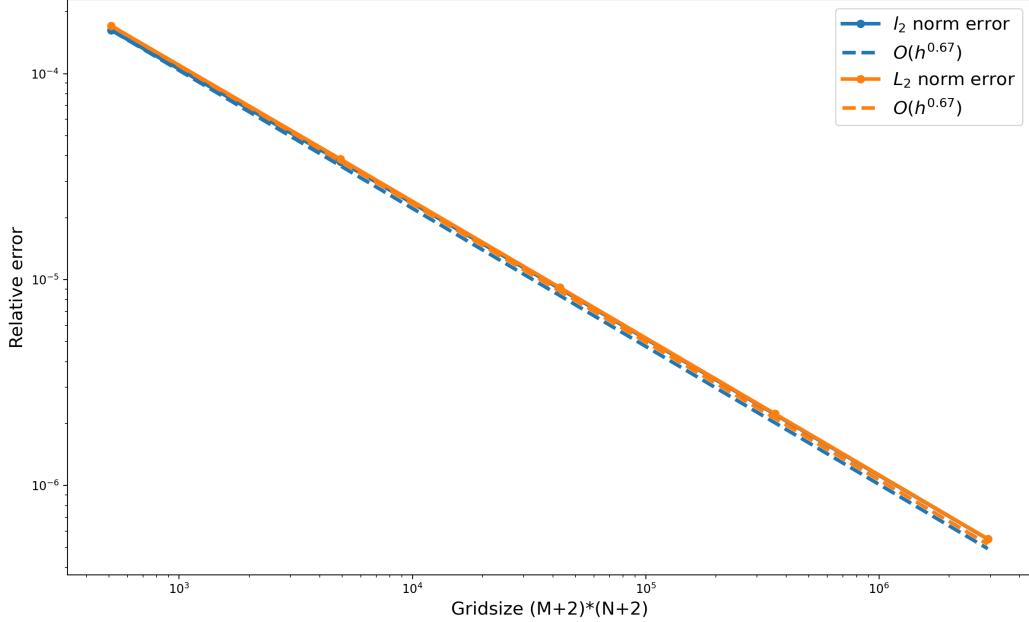


Figure 19: Convergence plot for the heat equation with Dirichlet boundary conditions using backward Euler discretization and refinement where $r = kh^2$ is constant.

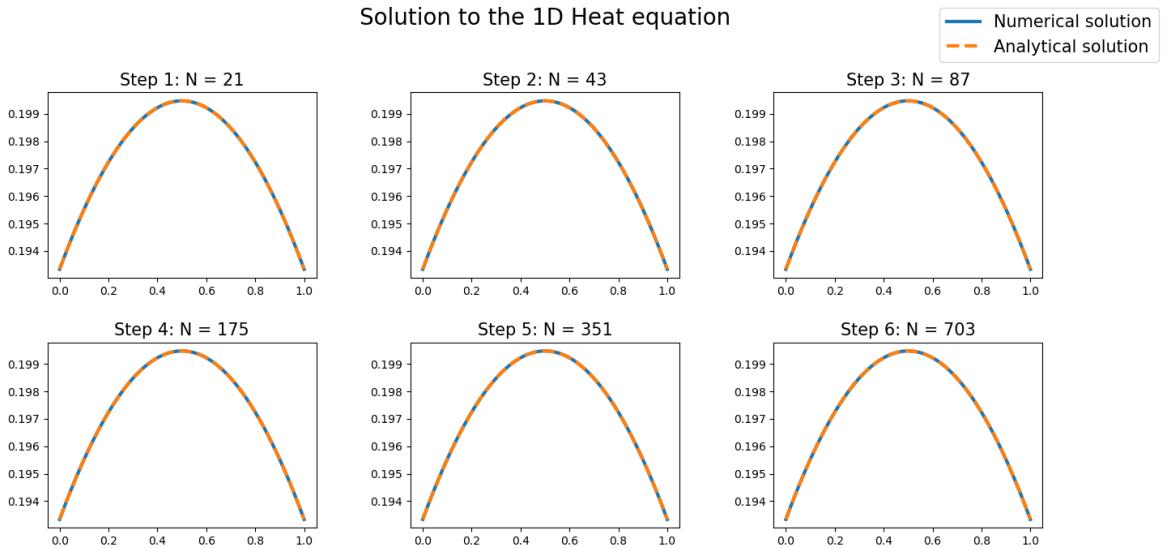


Figure 20: Solution to the heat equation in the domain (1.2.2) with Dirichlet boundary conditions using Crank-Nicolson discretization.

Convergence testing for the 1D Heat equation

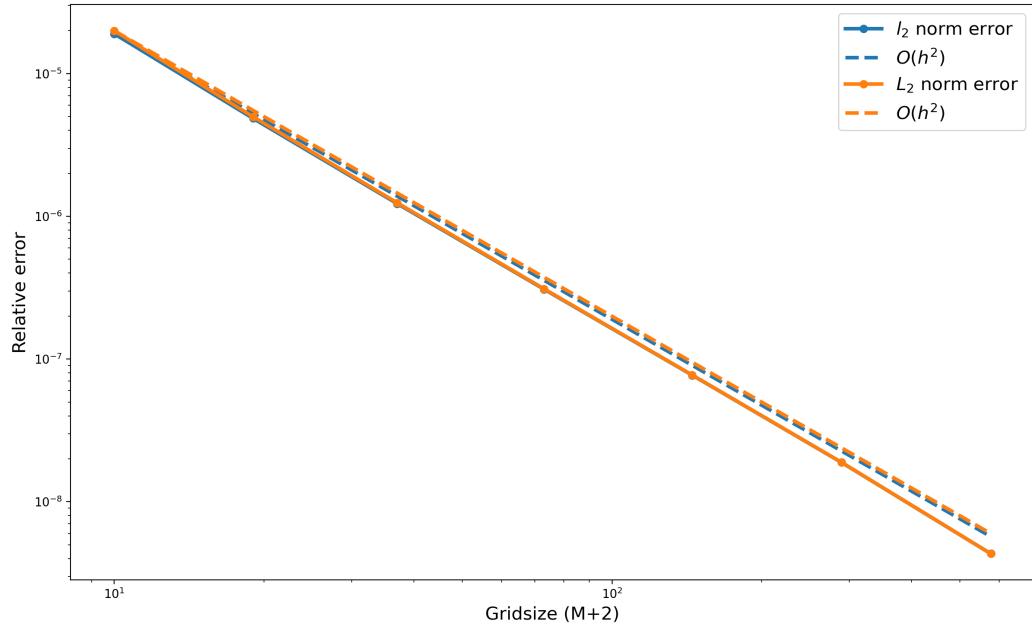


Figure 21: Convergence plot for the heat equation in the domain (1.2.2) with Dirichlet boundary conditions using Crank-Nicolson discretization with h -refinement.

Convergence testing for the 1D Heat equation

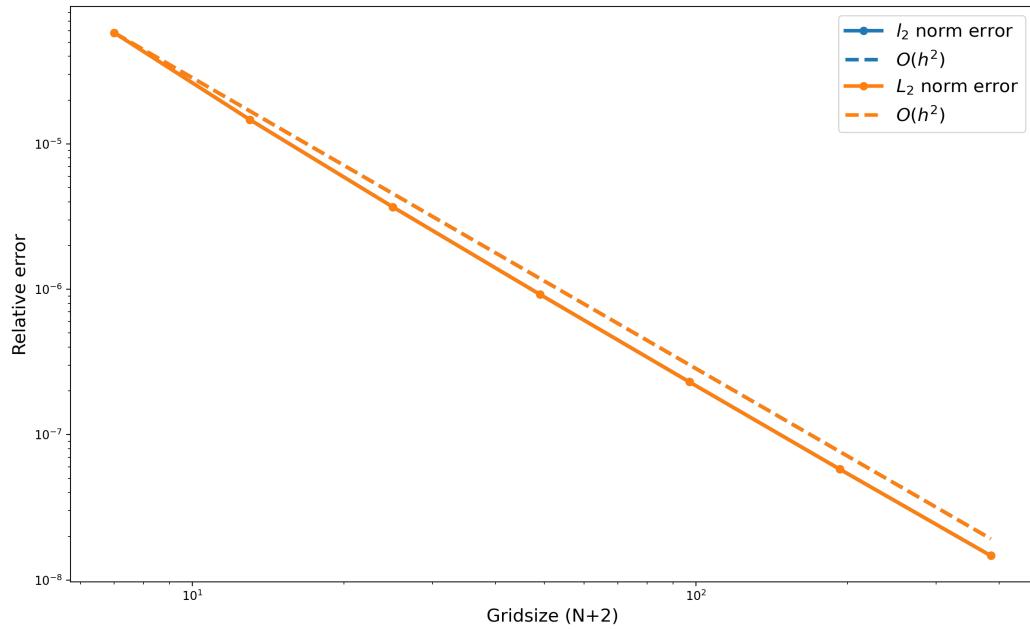


Figure 22: Convergence plot for the heat equation with Dirichlet boundary conditions using Crank-Nicolson discretization with k -refinement.

Convergence testing for the 1D Heat equation

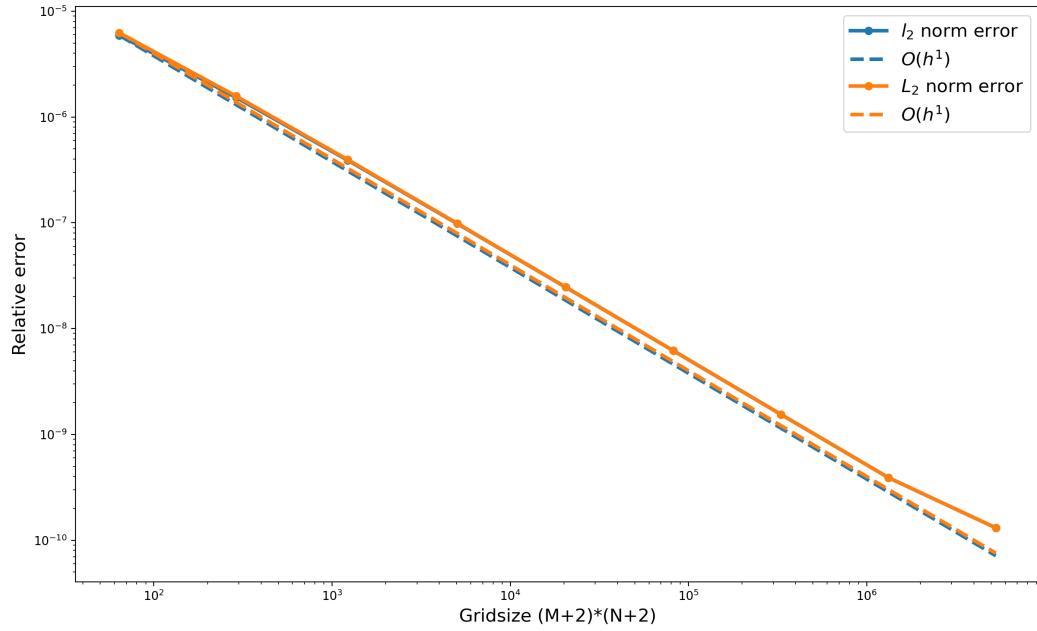


Figure 23: Convergence plot for the heat equation with Dirichlet boundary conditions using Crank-Nicolson discretization with $k = h$ -refinement.

Convergence testing for the 1D Heat equation

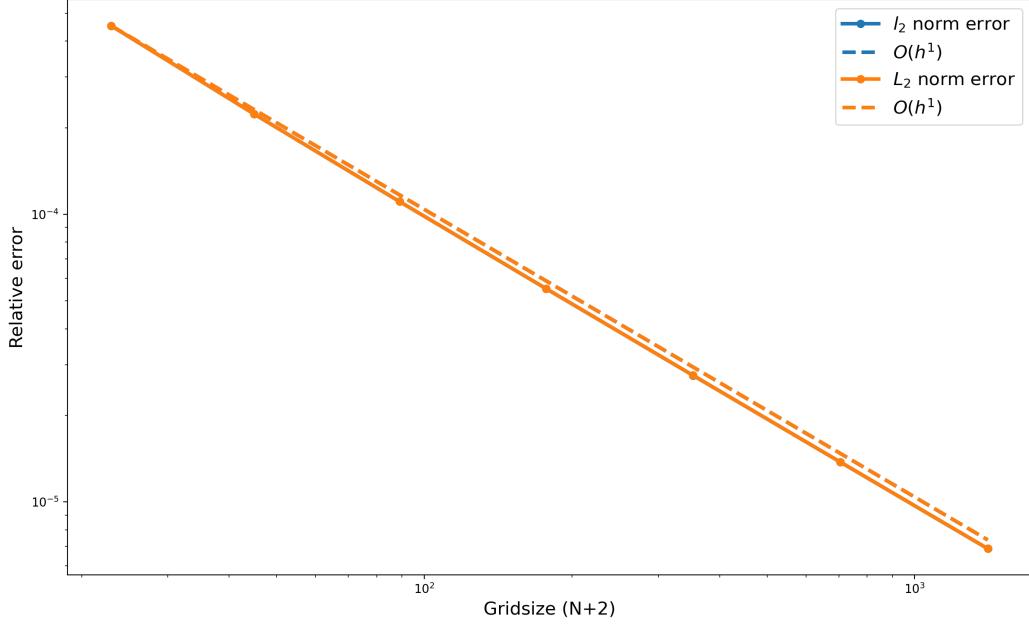


Figure 24: Convergence plot for the heat equation with Dirichlet boundary conditions using Crank-Nicolson discretization and refinement where $r = kh^2$ is constant.

1.2.2.2. Inviscid Burgers Equation

In the figures 25, 26 and 27 we see respectively the inviscid Burgers equation on the domain (1.2.6) with initial conditions and homogenous Dirichlet boundary conditions (1.2.7) at its breaking time $t = 0.01$ seconds (1.2.21), its contour plot until the breaking point as well as the initial profile and its characteristic curves.

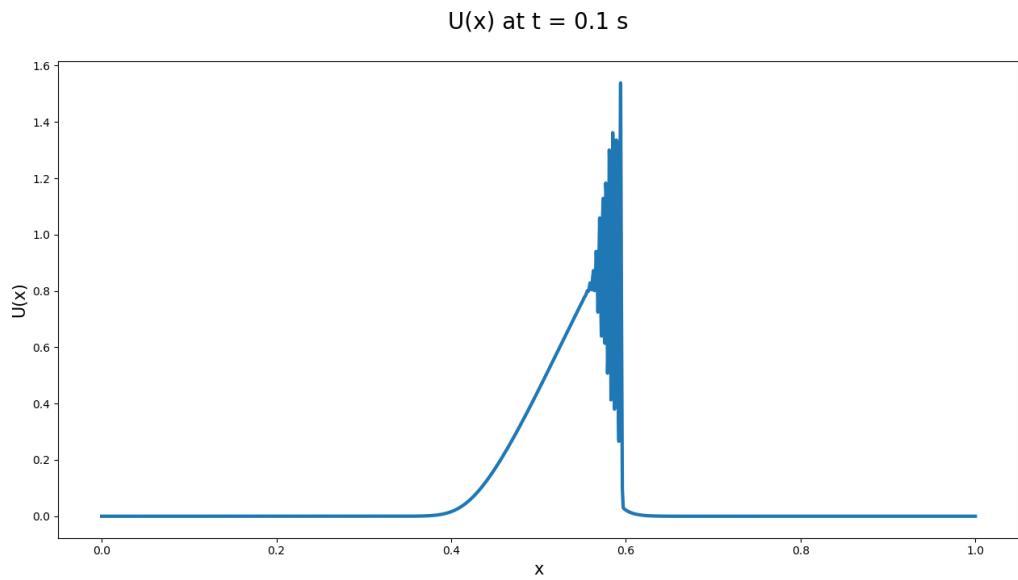


Figure 25: Plot for the inviscid Burgers equation on the domain (1.2.6) with initial conditions and homogenous Dirichlet boundary conditions (1.2.7) at its breaking point at time $t = 0.1$ seconds.

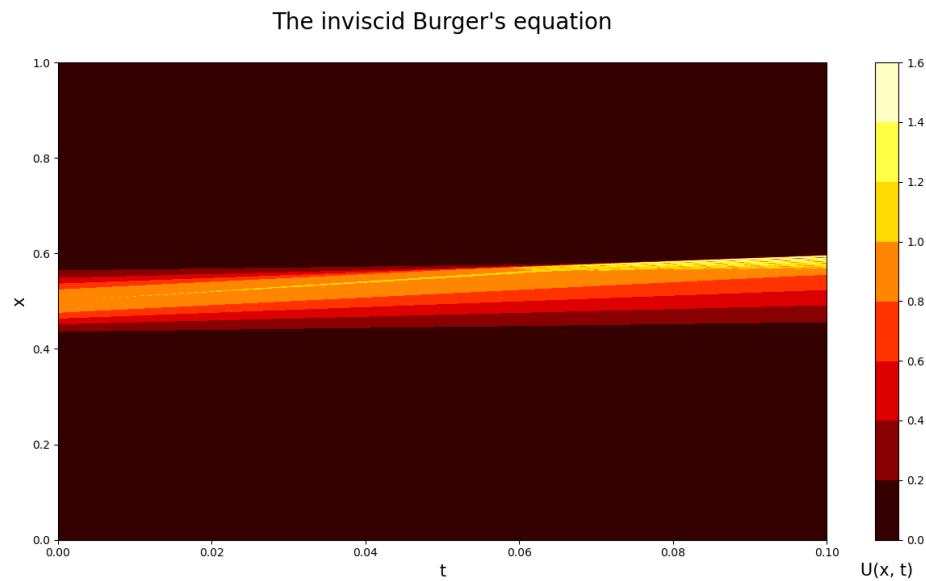


Figure 26: Contour plot for the inviscid Burgers equation on the domain (1.2.6) with initial conditions and homogenous Dirichlet boundary conditions (1.2.7).

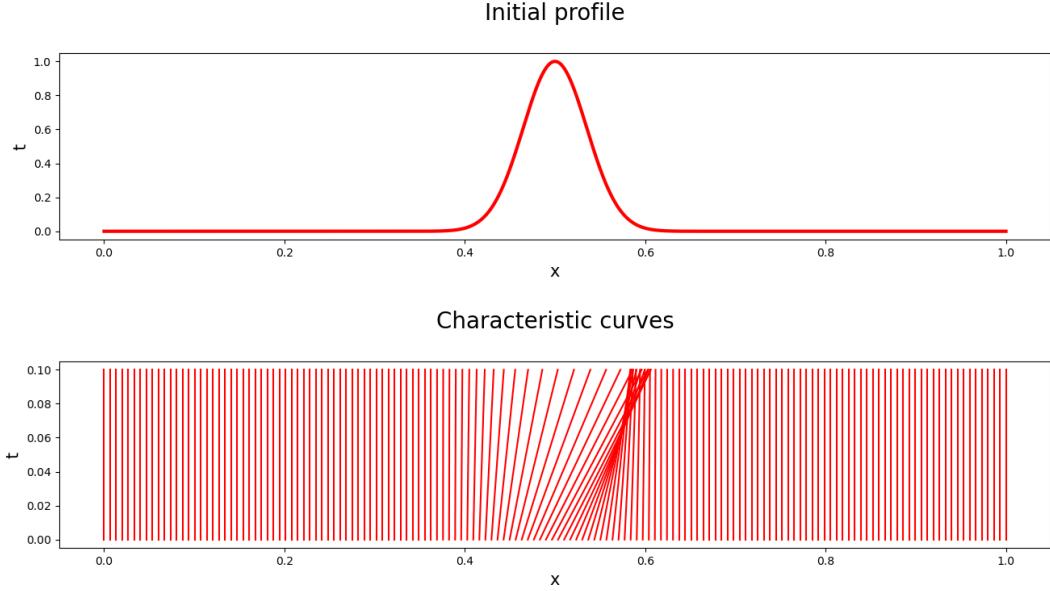


Figure 27: Initial profile and characteristic curves for the inviscid Burgers equation on the domain (1.2.6) with initial conditions and homogenous Dirichlet boundary conditions (1.2.7).

1.3. Problem 3 - The Laplace Equation

Problem 3 considers the two-dimensional Laplace equation on the unit square $(x, y) \in \Omega := [0, 1]^2$.

$$u_{xx} + u_{yy} = 0, \quad (1.3.1)$$

where we define

$$u(x, y) = g(x, y), \quad (1.3.2)$$

and have the boundary conditions

$$g(0, y) = 0, \quad 0 \leq y \leq 1, \quad (1.3.3)$$

$$g(x, 0) = 0, \quad 0 \leq x \leq 1, \quad (1.3.4)$$

$$g(1, y) = 0, \quad 0 \leq y \leq 1, \quad (1.3.5)$$

$$g(x, 1) = \sin(2\pi x), \quad 0 \leq x \leq 1. \quad (1.3.6)$$

1.3.1. Mathematical Formulation

1.3.1.1. Analytical Solution using Separation of Variables

We shall first derive the analytical solution to this problem. To begin with, we apply separation of variables by writing the solution as $u(x, y) = X(x) \cdot Y(y)$, leading to

$$\frac{X''(x)}{X(x)} = -\frac{Y''(y)}{Y(y)} = -\lambda$$

such that

$$X''(x) + \lambda X(x) = 0, \quad Y''(y) - \lambda Y(y) = 0$$

with boundary conditions $X(0) = X(1) = 0$ as in (1.3.3) and (1.3.5), $Y(0) = 0$ as in (1.3.4) and $Y(1) = \sin(2\pi x)$ as in (1.3.6). Further, the first boundary value problem gives the eigenvalues $\lambda_n = (n\pi)^2$ and eigenfunctions $X_n(x) = \sin(n\pi x)$ for $n = 1, 2, 3, \dots$. Applying this to the differential equation including $Y(y)$ gives $Y_n(y) = c_1 \cosh(n\pi y) + c_2 \sinh(n\pi y)$, which for the boundary condition $Y(0) = 0$ gives $Y_n(y) = c_2 \sinh(n\pi y)$ and the solution

$$u(x, y) = \sum_{n=1}^{\infty} B_n \sinh(n\pi y) \sin(n\pi x).$$

By applying the last boundary condition $u(x, 0) = \sin(2\pi x) = \sum_{n=1}^{\infty} B_n \sinh(n\pi) \sin(n\pi x)$, we observe that we have a Fourier sine series, such that B_n can be decided from

$$B_n = \frac{2}{\sinh(\pi n)} \int_0^1 \sin(2\pi x) \sin(n\pi x) dx$$

which is zero for all other values than $n = 2$, where $B_n = 1/\sinh(2\pi)$. The analytical solution of the boundary value problem is thereby

$$u(x, y) = \frac{\sinh(2\pi y) \cdot \sin(2\pi x)}{\sinh(2\pi)}. \quad (1.3.7)$$

1.3.1.2. Difference Scheme

To approximate the derivatives, we used as in [3] second order central finite difference approximations in both spatial dimensions such that

$$u_{xx} + u_{yy} = \frac{1}{k^2} (U_{i,j+1} - 2U_{i,j} + U_{i,j-1}) + \frac{1}{h^2} (U_{i+1,j} - 2U_{i,j} + U_{i-1,j}) + \mathcal{O}(k^2) + \mathcal{O}(h^2). \quad (1.3.8)$$

Based on our difference schemes we expect second order convergence in both x - and y -dimensions, which is also observed in the convergence plots below.

1.3.2. Numerical Results

We see from figure 28 that the numerical solution to the Laplace equation with Dirichlet boundary conditions tends to the analytical solution (1.3.7) for values of M_x from around 15 and larger. From the convergence plot in figure 29 we see that the convergence rate approximately is of order $\mathcal{O}(h^2)$, as expected.

By looking at figure 30 we see that the numerical solution to the Laplace equation with Dirichlet boundary conditions seem to coincide with the analytical solution for values of M_y from between 7 and 15 and larger. From figure 31 the convergence seems to be of order $\mathcal{O}(k^2)$, as expected.

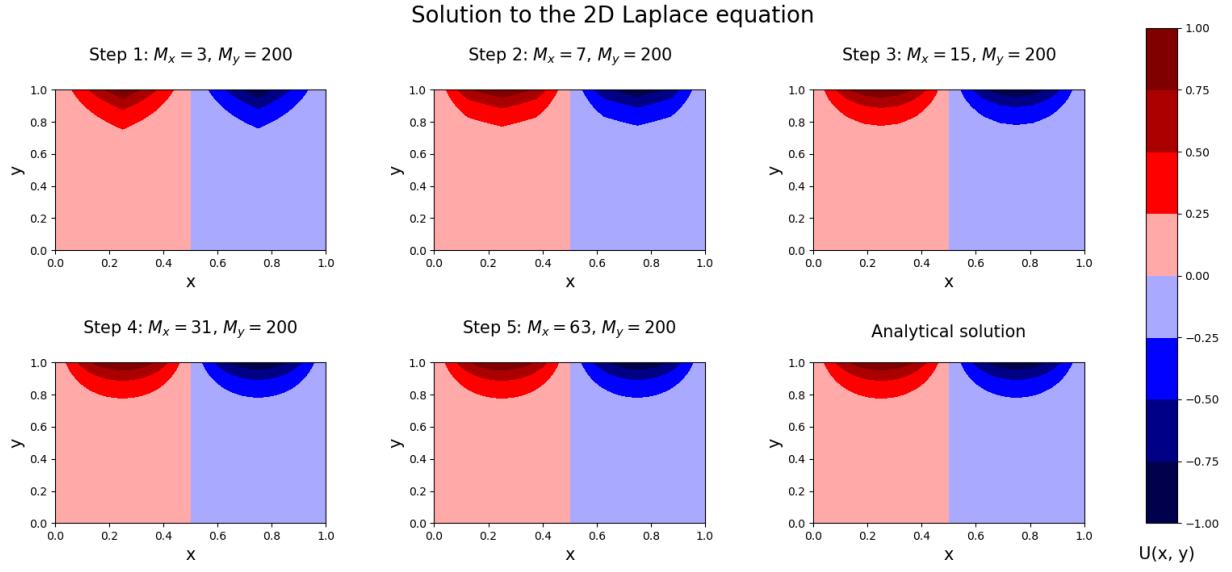


Figure 28: Numerical solution to the Laplace equation with Dirichlet boundary conditions for different gridpoints M_x along the x -axis together with the analytical solution.

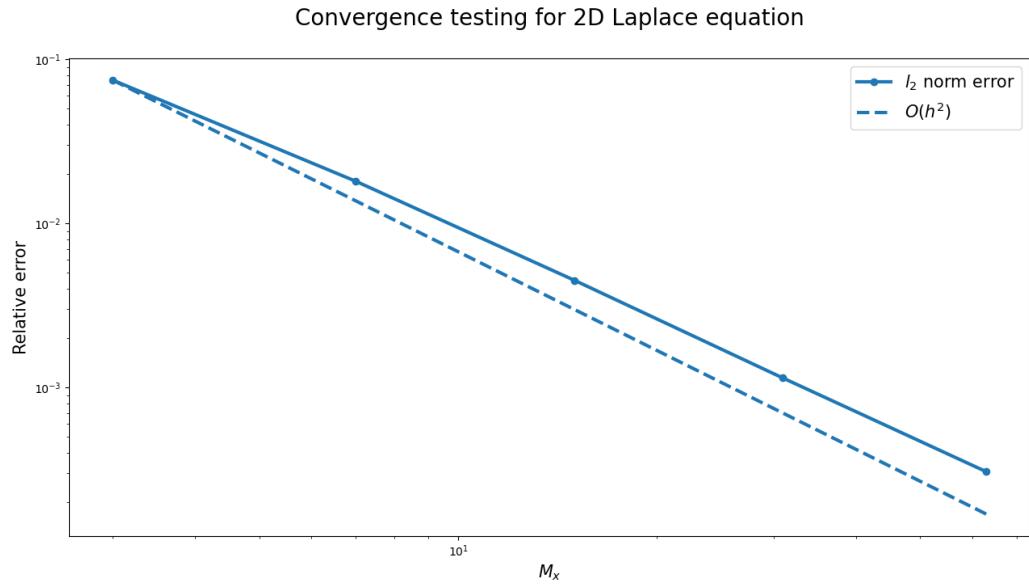


Figure 29: Convergence plot for the Laplace equation with Dirichlet boundary conditions for different gridpoints M_x along the x -axis.

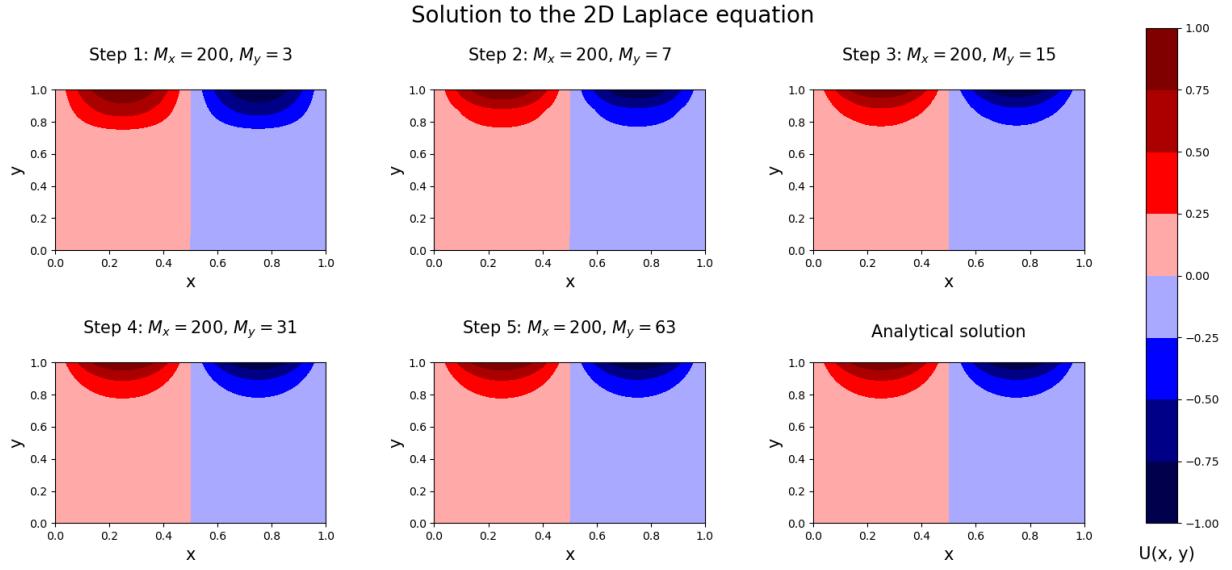


Figure 30: Numerical solution to the Laplace equation with boundary conditions for different gridpoints M_y along the y -axis together with the analytical solution.

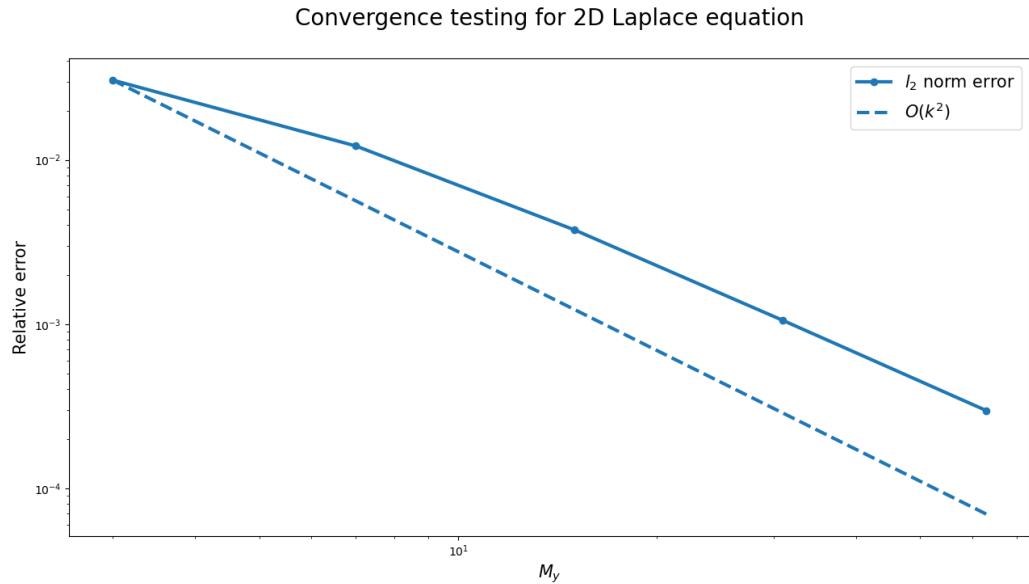


Figure 31: Convergence plot for the Laplace equation with Dirichlet boundary conditions for different gridpoints M_y along the y -axis.

1.4. Problem 4 - Korteweg-deVries Equation

Problem 4 considers a linearized Korteweg-deVries (KdV) equation given as

$$u_t + (1 + \pi^2)u_x + u_{xxx} = 0 \quad (1.4.1)$$

with initial condition

$$u(x, 0) = \sin(\pi x), \quad (1.4.2)$$

and a periodic boundary condition

$$u(x + 2, t) = u(x, t). \quad (1.4.3)$$

1.4.1. Mathematical Formulation

1.4.1.1. Analytical Solution

The analytical solution to the problem is given from [1] to be

$$u(x, t) = \sin(\pi(x - t)). \quad (1.4.4)$$

1.4.1.2. Difference Schemes

For the discretization of the KdV equation, we apply certain conditions given in [1]. In space, we apply central difference discretization, in time we apply both forward Euler and the trapezoidal rule for discretization, while the discretization of the u_{xxx} -term will be

$$u_{xxx}|_{x_m} = \frac{U_{m+3}^n - 3U_{m+1}^n + 3U_{m-1}^n - U_{m-3}^n}{8h^3} + \mathcal{O}(h^2). \quad (1.4.5)$$

Applying the forward Euler method for temporal discretization, our difference scheme becomes

$$\frac{U_m^{n+1} - U_m^n}{k} + (1 + \pi^2) \frac{U_{m+1}^n - U_{m-1}^n}{2h} + \frac{U_{m+3}^n - 3U_{m+1}^n + 3U_{m-1}^n - U_{m-3}^n}{8h^3} + \mathcal{O}(k) + \mathcal{O}(h^2). \quad (1.4.6)$$

For the Crank-Nicolson difference scheme we approximate the partial derivatives of u in (1.4.1) such that the difference scheme becomes

$$\begin{aligned} & \frac{U_j^{n+1} - U_j^n}{k} + \frac{1}{4h} \left((U_{j+1}^n - U_{j-1}^n) + (U_{j+1}^{n+1} - U_{j-1}^{n+1}) \right) + \\ & \frac{1}{16h^3} \left((U_{j+3}^n - 3U_{j+1}^n + 3U_{j-1}^n - U_{j-3}^n) + (U_{j+3}^{n+1} - 3U_{j+1}^{n+1} + 3U_{j-1}^{n+1} - U_{j-3}^{n+1}) \right) + \mathcal{O}(h^2) + \mathcal{O}(k). \end{aligned} \quad (1.4.7)$$

1.4.1.3. Von Neumann Stability Analysis

To begin with, we want to calculate the von Neumann stability conditions for the KdV equation. Substituting $U_m^n = \xi^n e^{i\beta x_m}$ in the forward Euler difference scheme, we get

$$\xi = 1 - (1 + \pi^2) \frac{k}{2h} \cdot 2i \sin(\beta h) - \frac{k}{8h^3} (2i \sin(3\beta h) - 6i \sin(\beta h))$$

such that

$$\xi = 1 - \frac{ki}{4h^3} (\sin(3\beta h) - 3 \sin(\beta h) + 4h^2(1 + \pi^2) \sin(\beta h)).$$

Knowing that

$$|\xi|^2 = \operatorname{Re}(\xi)^2 + \operatorname{Im}(\xi)^2, \quad (1.4.8)$$

we get

$$|\xi|^2 = 1 - \frac{k^2}{h^2} \left((1 + \pi^2) \sin(\beta h) - \frac{1}{4h^2} \sin(3\beta h) + \frac{3}{4h^2} \sin(\beta h) \right)^2,$$

such that

$$|\xi| = \sqrt{\left(1 + \frac{k^2}{h^2} \left((1 + \pi^2) \sin(\beta h) - \frac{1}{4h^2} \sin(3\beta h) + \frac{3}{4h^2} \sin(\beta h)\right)\right)} \\ \cdot \sqrt{\left(1 - \frac{k^2}{h^2} \left((1 + \pi^2) \sin(\beta h) - \frac{1}{4h^2} \sin(3\beta h) + \frac{3}{4h^2} \sin(\beta h)\right)\right)}.$$

This can not be written on the form

$$|\xi| \leq 1 + \mu k \quad (1.4.9)$$

which we want to prove stability implying that (1.4.1) is unstable using forward Euler discretization.

Appling the same substitution as above, we get

$$\frac{\xi - 1}{k} + (1 + \pi^2) \frac{1}{4h} \left(2i \sin(\beta h)\right)(1 + \xi) + \frac{1}{16h^3} \left(2i(\sin(3\beta h) - 3 \sin(\beta h))\right)(1 + \xi) = 0$$

implying

$$\xi = \frac{1 - 2ki \left((1 + \pi^2) \frac{1}{4h} \sin(\beta h) + \frac{1}{16h^3} (\sin(3\beta h) - 3 \sin(\beta h))\right)}{1 + 2ki \left((1 + \pi^2) \frac{1}{4h} \sin(\beta h) + \frac{1}{16h^3} (\sin(3\beta h) - 3 \sin(\beta h))\right)}$$

such that

$$\xi = \frac{1 - 4k^2 \left((1 + \pi^2) \frac{1}{4h} \sin(\beta h) + \frac{1}{16h^3} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2}{1 + 4k^2 \left((1 + \pi^2) \frac{1}{4h} \sin(\beta h) + \frac{1}{16h^3} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2} \\ - \frac{2ki \left((1 + \pi^2) \frac{1}{4h} \sin(\beta h) + \frac{1}{16h^3} (\sin(3\beta h) - 3 \sin(\beta h))\right)}{1 + 4k^2 \left((1 + \pi^2) \frac{1}{4h} \sin(\beta h) + \frac{1}{16h^3} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2}$$

Taking (1.4.8) into account, and introducing $r = k/h^2$ we get that

$$|\xi|^2 = \left(\frac{1 - 4rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2}{1 + 4rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2} \right)^2 \\ + \left(\frac{2i\sqrt{rk} \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)}{1 + 4rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2} \right)^2 \\ = \left(1 - \frac{8rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2}{1 + 4rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2} \right)^2 \\ - \left(\frac{2\sqrt{rk} \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)}{1 + 4rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2} \right)^2.$$

In order to obtain (1.4.9), let

$$\left| 1 - \frac{8rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2}{1 + 4rk \left((1 + \pi^2) \frac{1}{4} \sin(\beta h) + \frac{1}{16h^2} (\sin(3\beta h) - 3 \sin(\beta h))\right)^2} \right| \leq 1 + \tilde{\mu}k.$$

This is satisfied for

$$|\xi|^2 \leq 1 + \frac{1}{4}r \left((1 + \pi^2) + \frac{1}{h^2}\right)k, \quad (1.4.10)$$

such that the von Neumann stability criterion is obtained for $\mu = \frac{1}{4}r \left((1 + \pi^2) + \frac{1}{h^2}\right)$.

1.4.1.4. Conservation of the L_2 Norm

In this part of the problem we shall prove that

$$\|u(x, t)\| := \sqrt{\frac{1}{2} \int_{-1}^1 |u(x, t)|^2 dx} = \sqrt{\frac{1}{2} \int_{-1}^1 |u(x, 0)|^2 dx}. \quad (1.4.11)$$

We are given that

$$\begin{aligned} u(x, t) &= \sum_{k \in \mathbb{Z}} \hat{u}(k, t) e^{i\pi kx}, \\ \hat{u}(k, t) &= \frac{1}{2} \int_{-1}^1 u(x, t) e^{-i\pi kx} dx, \end{aligned} \quad (1.4.12)$$

and

$$u(x, t) = \sum_{k \in \mathbb{Z}} \hat{u}(k, 0) e^{-i\pi k(1+\pi^2)t + i\pi^3 k^3 t} e^{i\pi kx}. \quad (1.4.13)$$

By applying the following identities

$$\begin{aligned} |u(x, t)|^2 &= u(x, t) \overline{u(x, t)}, \\ e^{ia} \cdot \overline{e^{ia}} &= e^{ia} \cdot e^{-ia} = e^{ia - ia} = e^0 = 1, \end{aligned}$$

on (1.4.13) with (1.4.12) inserted, we obtain

$$\begin{aligned} \|u(x, t)\| &:= \sqrt{\frac{1}{2} \int_{-1}^1 |u(x, t)|^2 dx} = \sqrt{\frac{1}{2} \int_{-1}^1 u(x, t) \overline{u(x, t)} dx} \\ &= \sqrt{\frac{1}{2} \int_{-1}^1 \sum_{k \in \mathbb{Z}} \left(\frac{1}{2} \int_{-1}^1 u(x, 0) e^{-i\pi kx} dx \right) \cdot \sum_{k \in \mathbb{Z}} \left(\frac{1}{2} \int_{-1}^1 \overline{u(x, 0)} e^{i\pi kx} dx \right) dx} \\ &= \sqrt{\frac{1}{2} \int_{-1}^1 \sum_{k \in \mathbb{Z}} \hat{u}(k, 0) \cdot \sum_{k \in \mathbb{Z}} \overline{\hat{u}(k, 0)} dx} \\ &= \sqrt{\frac{1}{2} \int_{-1}^1 u(x, 0) e^{-i\pi kx} \cdot \overline{u(x, 0)} e^{i\pi kx} dx} \\ &= \sqrt{\frac{1}{2} \int_{-1}^1 u(x, 0) \cdot \overline{u(x, 0)} dx} \\ &= \sqrt{\frac{1}{2} \int_{-1}^1 |u(x, 0)|^2 dx}, \end{aligned}$$

which is what we wanted to prove.

1.4.1.5. Expected Convergence Rates

From the convergence plots, we expect to see second order convergence in space for both the Crank-Nicolson and forward Euler methods due to the truncation errors found in (1.4.5) and (1.4.6). Based on our stability analysis, we should also expect to observe the instability of the Euler method in addition to the stability of the Crank-Nicolson method. Indeed, we see that the Crank-Nicolson method converges nicely, while the relative error for the Euler method eventually grows exponentially large.

1.4.2. Numerical Results

1.4.2.1. KdV Equation

From figure 32 we observe that the numerical solution coincides with the analytical solution for $M \geq 80$ when the amount of timesteps is $N = 1000$. Furthermore, from the convergence plot in 33 we see that the solution for the Crank-Nicolson discretization method converges of order $\mathcal{O}(h^2)$ as expected from (1.4.7), which is reasonable as we found the method to be stable from (1.4.9).

From the figures 34 and 36 we see that for both $N = 50 \cdot 10^6$ and $N = 10^7$ number of timesteps the numerical solution to the KdV equation tends to the analytical solution (1.4.4) for values of number of gridpoints $M \approx 40$. However, as M increases, the solution becomes not clearly defined, which we also can see from the convergence plots in the figures 35 and 37, where they seem to diverge. This is reasonable as we from the von Neumann stability analysis found that the Euler method is unstable for this problem.

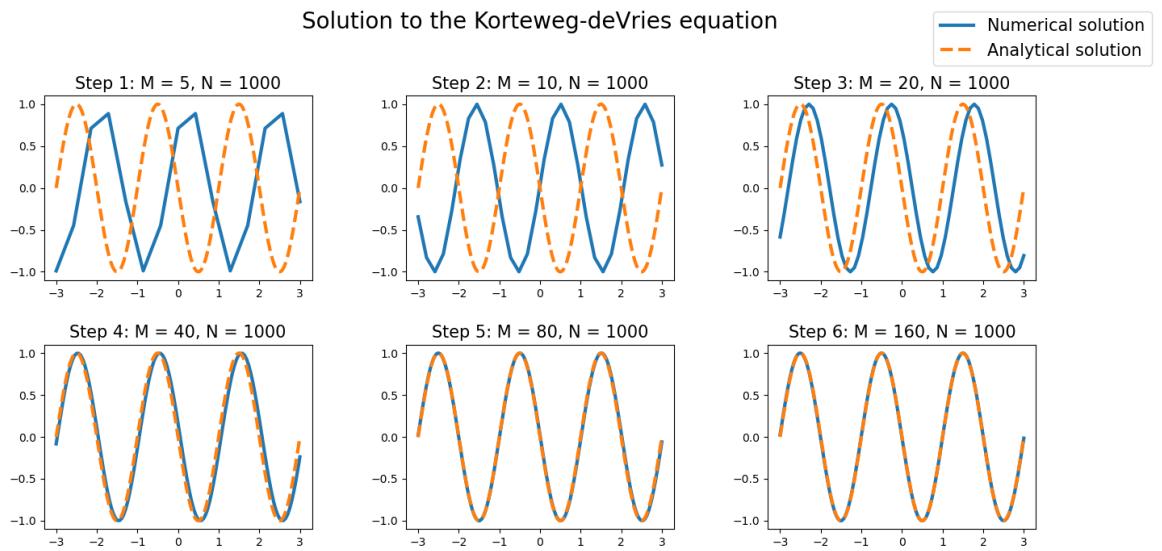


Figure 32: Numerical solution to the KdV equation using the Crank-Nicolson discretization method (1.4.7) compared to the analytical solution (1.4.4).

Convergence testing for the Korteweg-deVries equation

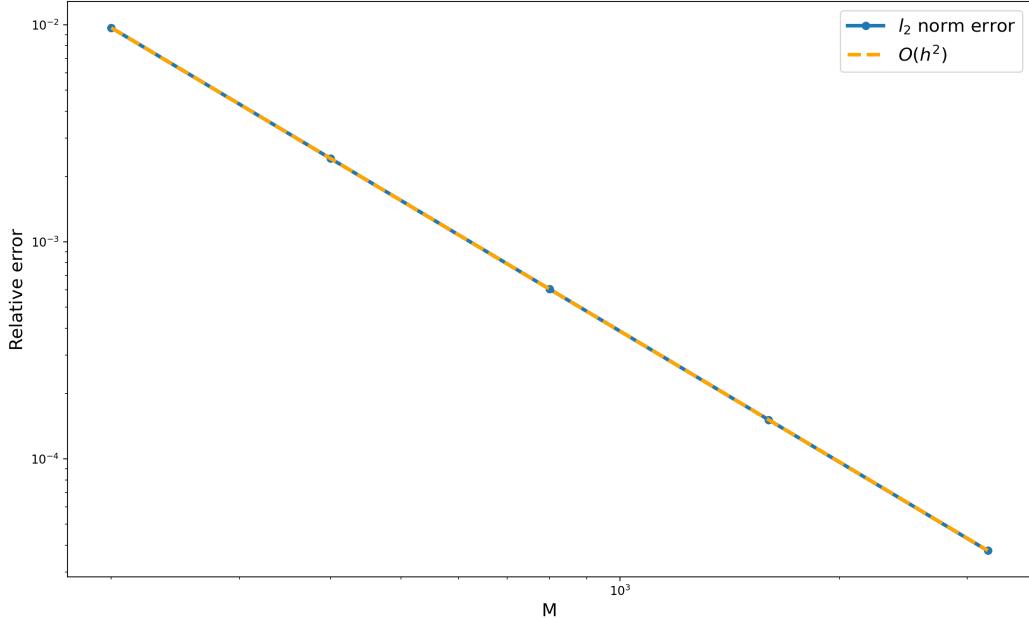


Figure 33: Convergence plot of the KdV equation using the Crank-Nicolson discretization method.

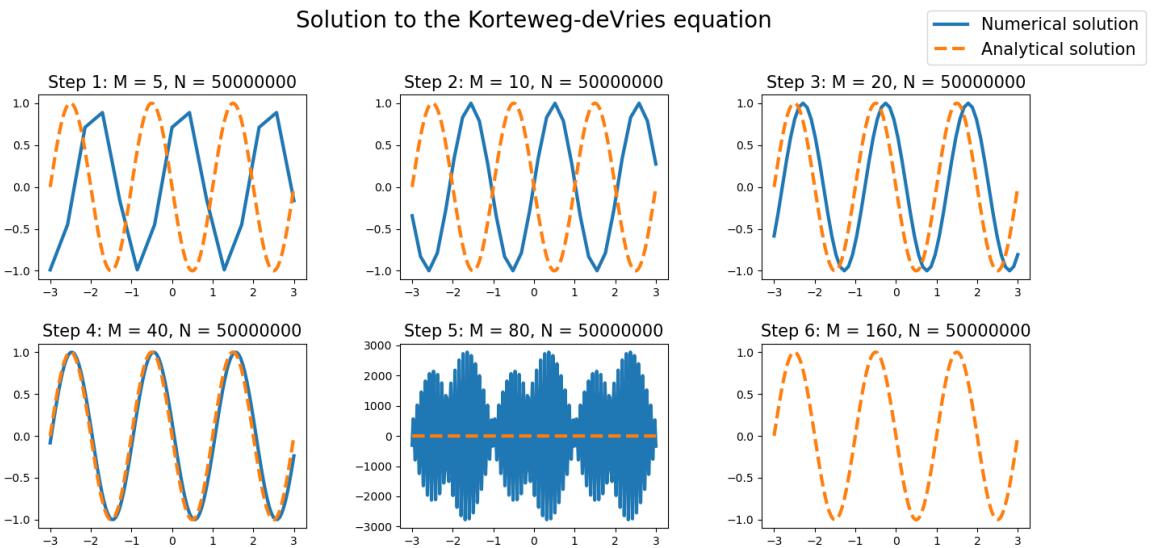


Figure 34: Numerical solution to the KdV equation using forward Euler discretization method (1.4.6) with $N = 50 \cdot 10^6$ timesteps compared to the analytical solution.

Convergence testing for the Korteweg-deVries equation

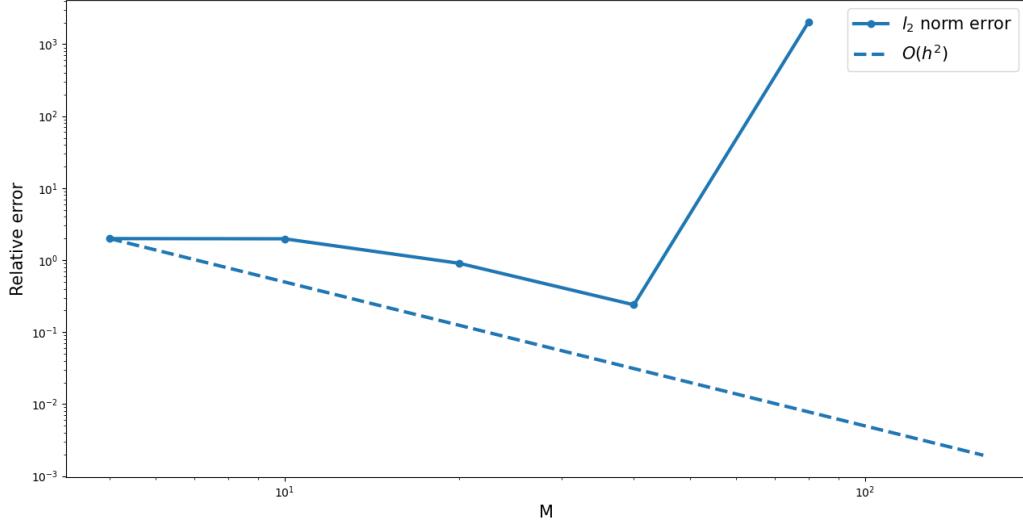


Figure 35: Convergence plot of the KdV equation using the forward Euler discretization method with $N = 50 \cdot 10^6$ timesteps.

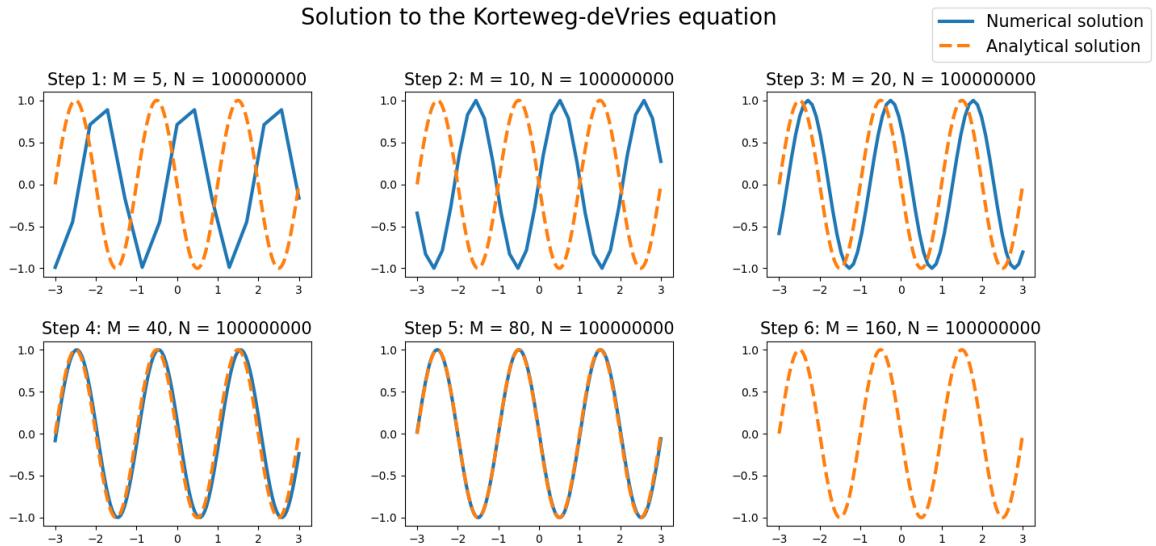


Figure 36: Numerical solution to the KdV equation using forward Euler time discretization with $N = 10^7$ timesteps compared to the analytical solution.

Convergence testing for the Korteweg-deVries equation

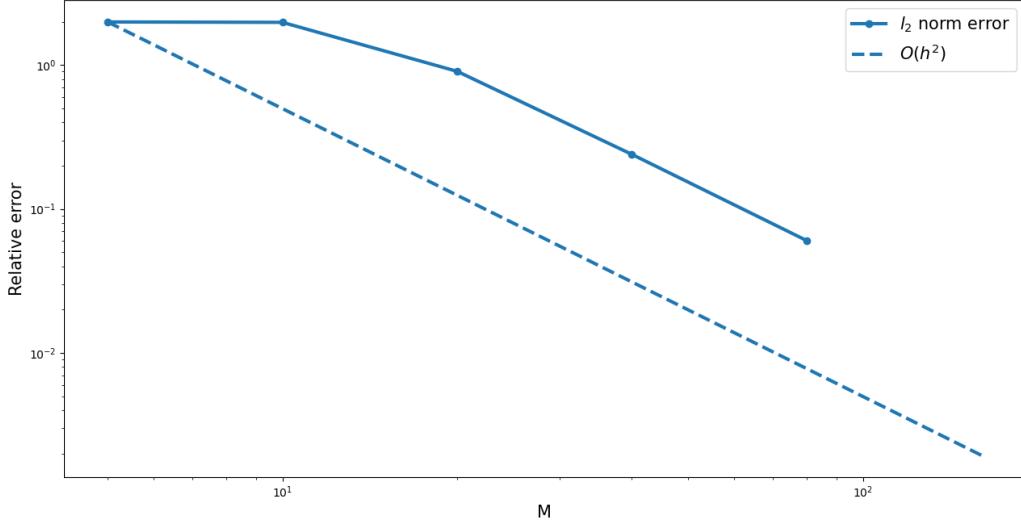


Figure 37: Convergence plot of the KdV equation using forward Euler time discretization with $N = 10^7$ timesteps.

1.4.2.2. Conservation of the ℓ_2 -norm

For the numerical verification of (1.4.11) we chose the initial condition $u(x, 0) = \cos(4\pi x)$ which gives the analytical solution $\cos(\pi x - t)$. The discrete ℓ_2 -norm is plotted against time for both the forward Euler and Crank-Nicolson methods. We see that the ℓ_2 -norm is indeed conserved when using the Crank-Nicolson scheme. However, when using the Euler method we again run into stability issues, making it impossible to verify the numerical conservation of the L_2 -norm for this method.

Discrete ℓ^2 Norm over time for the Korteweg-deVries equation

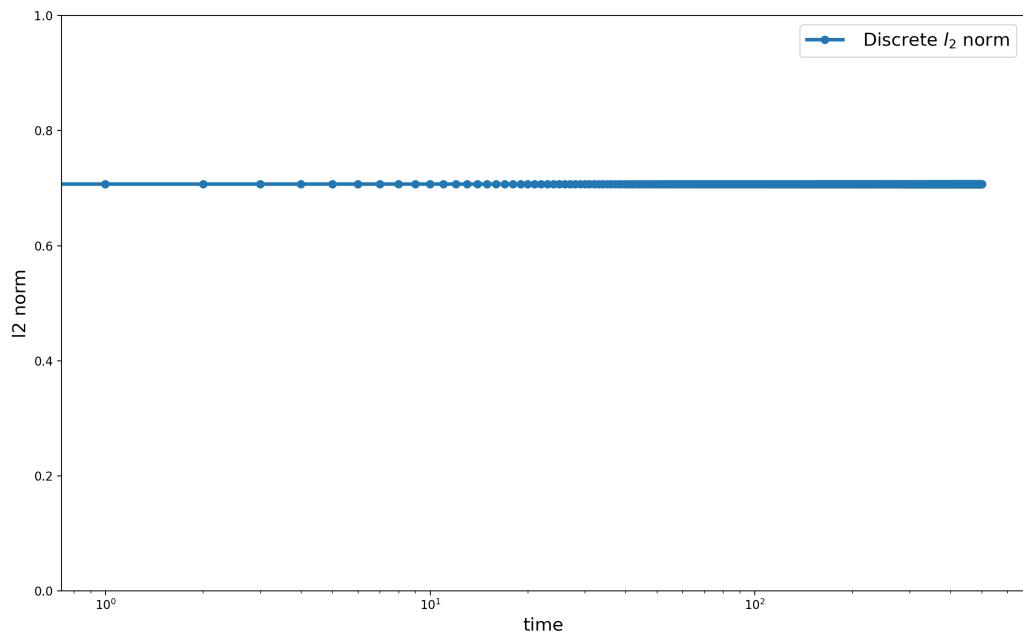


Figure 38: Discrete ℓ_2 norm plotted against timesteps for the Crank-Nicolson difference scheme (1.4.7).

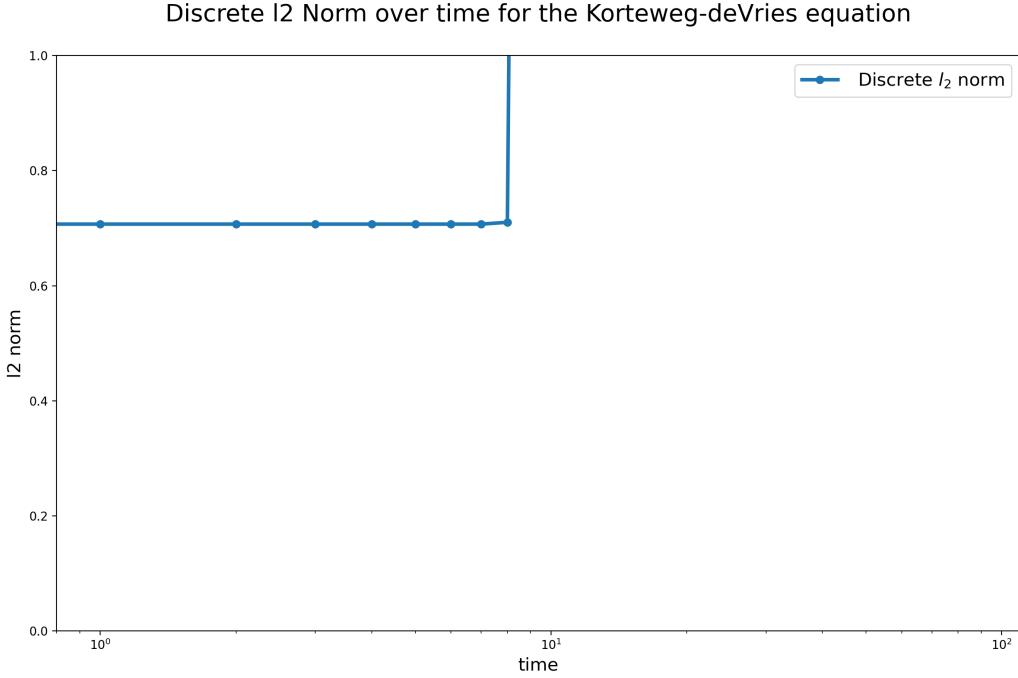


Figure 39: Discrete ℓ_2 norm plotted against timesteps for the forward Euler difference scheme (1.4.6).

1.5. Problem 5 - The Poisson Equation by Finite Element Methods

In the last problem of part 1 we will consider the Poisson equation

$$-u_{xx} = f(x), \quad (1.5.1)$$

with inhomogenous Dirichlet boundary conditions

$$u(a) = d_1, \quad u(b) = d_2. \quad (1.5.2)$$

1.5.1. Mathematical Formulation

1.5.1.1. Assembly

We begin by discretizing the equation into a linear system $\mathbf{Au} = \mathbf{f}$. This is done by first setting up a variational form of (1.5.1) such that

$$\int_0^1 u'(x)v'(x)dx = \int_0^1 f(x)v(x)dx. \quad (1.5.3)$$

In addition, we use basis functions $\phi_i(x)$ given by

$$\phi_i(x) = \begin{cases} \frac{x-x_{i-1}}{x_i-x_{i-1}} & x_{i-1} < x < x_i \\ \frac{x_{i+1}-x}{x_{i+1}-x_i} & x_i < x < x_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (1.5.4)$$

for $i \in [0, N]$. Assembling this to a linear system gives the following expressions

$$a_{ij} = \int_0^1 \phi'_i(x) \phi'_j(x) dx , \quad f_j = \int_0^1 f(x) \phi_j(x) dx \quad (1.5.5)$$

where a_{ij} is element number ij in our \mathbf{A} matrix and f_j is the j -th element in our vector \mathbf{f} . We observe from the expressions for ϕ_i that the derivatives become

$$\phi'_i(x) = -\frac{1}{x_{i+1} - x_i} , \quad \phi'_{i+1}(x) = \frac{1}{x_{i+1} - x_i}. \quad (1.5.6)$$

Taking this, and the integrals above into account we then get for each element

$$A_h = \frac{1}{x_{k+1} - x_k} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} , \quad \mathbf{f}_h = \begin{bmatrix} \int_{x_k}^{x_{k+1}} (x_{k+1} - x) f(x) dx \\ \int_{x_k}^{x_{k+1}} (x - x_k) f(x) dx \end{bmatrix} \quad (1.5.7)$$

contributing to the matrix A and vector \mathbf{f} such that

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 & \dots & 0 \\ -1 & 2 & -1 & 0 & \dots & 0 \\ 0 & -1 & 2 & -1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & -1 & 2 & -1 \\ 0 & \dots & 0 & 0 & -1 & 1 \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} \int_0^1 (x_1 - x) f(x) dx \\ \int_0^1 (x - x_0) f(x) dx + \int_1^2 (x_2 - x) f(x) dx \\ \int_1^2 (x - x_1) f(x) dx + \int_2^3 (x_3 - x) f(x) dx \\ \dots \\ \int_{N-2}^{N-1} (x - x_{N-2}) f(x) dx + \int_{N-1}^N (x_N - x) f(x) dx \\ \int_{N-1}^N (x - x_{N-1}) f(x) dx \end{bmatrix}. \quad (1.5.8)$$

The numerical approximation for \mathbf{f} can then be found from Gaussian quadrature where numerical approximations of the integrals can be found by

$$\int_a^b f(x) dx \approx \frac{b-a}{2} \sum_{q=1}^{N_q} \rho_q f\left(\frac{b-a}{2} z_q + \frac{b+a}{2}\right), \quad (1.5.9)$$

where ρ_q are the weights of the Legendre polynomials and z_q are the roots.

Since we have inhomogenous Dirichlet boundary conditions, we can rewrite our solution as

$$u(x) = \hat{u}(x) + R_h(x), \quad (1.5.10)$$

where $\hat{u}(x)$ is the solution to the homogenous problem and $R_h(x)$ is the lifting function used to handle the boundary conditions. Applying (1.5.10) to the linear system $\mathbf{A}\mathbf{u} = \mathbf{f}$ we get a new linear system to solve

$$\mathbf{A}\hat{u}(x) = \mathbf{f} - \mathbf{A} \cdot R_h(x) \quad (1.5.11)$$

which then allows us to solve (1.5.10). Lastly, the solution to the Poisson equation is given by

$$u_h(x) = \sum_i u_i \phi_i(x) \quad (1.5.12)$$

1.5.1.2. Analytical Solutions

We will be solving (1.5.1) numerically for four different boundary value problems.

First, we will consider $f(x) = -2$, $d_1 = 0$ and $d_2 = 1$ on the domain $0 \leq x \leq 1$. The analytical solution to this problem is

$$u(x) = x^2. \quad (1.5.13)$$

Then, we will consider (1.5.1) for $f(x) = -(40000x^2 - 200)e^{-100x^2}$, $d_1 = e^{-100}$ and $d_2 = e^{-100}$ on the domain $-1 \leq x \leq 1$. The analytical solution to this problem is

$$u(x) = e^{-100x^2}. \quad (1.5.14)$$

Next, we will consider $f(x) = -(4000000x^2 - 2000)e^{-1000x^2}$, $d_1 = e^{-1000}$ and $d_2 = e^{-1000}$ on the domain $-1 \leq x \leq 1$. The analytical solution to this problem is

$$u(x) = e^{-1000x^2}. \quad (1.5.15)$$

At last, we will consider (1.5.1) for $f(x) = \frac{2}{9}x^{-\frac{4}{3}}$, $d_1 = 0$ and $d_2 = 1$ on the domain $0 \leq x \leq 1$. The analytical solution to this was problem is

$$u(x) = x^{\frac{2}{3}}. \quad (1.5.16)$$

1.5.1.3. Numerical Implementation

After finding the discrete vector \mathbf{u} from (1.5.11) and (1.5.10) we approximate the solution (1.5.12) using linear interpolation. Furthermore, the convergence plots were made starting with $N = 8$ grid points and doubling each iteration, ending with $N = 2048$.

1.5.2. Numerical Results

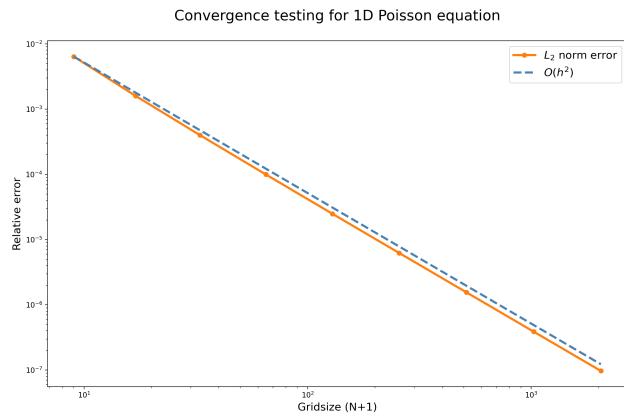
Given that the analytical solution is $u_{ex} \in H^{k+1}$, the convergence rate of our method is $\mathcal{O}(h^{k+1})$ [8]. We observe second order convergence for the first three problems, but for the problem with solution (1.5.16) we observe only first order convergence when using uniform refinement, which is because the analytical solution for this problem is not as regular as for the first three. Moreover, for both of the problems with solutions (1.5.14) and (1.5.15), the convergence rate has an initial bend before it becomes linear. This issue can be resolved by choosing a higher initial value for N when doing the iterations. In that case, the convergence plot will closely resemble the one for problem (1.5.13) as the convergence rate will be of order 2 exactly.

From figure 40 we see that the numerical solution to the Poisson equation with $f(x) = -2$ converges to the analytical solution (1.5.13) with a second order convergence rate, both for uniform and adaptive refinement.

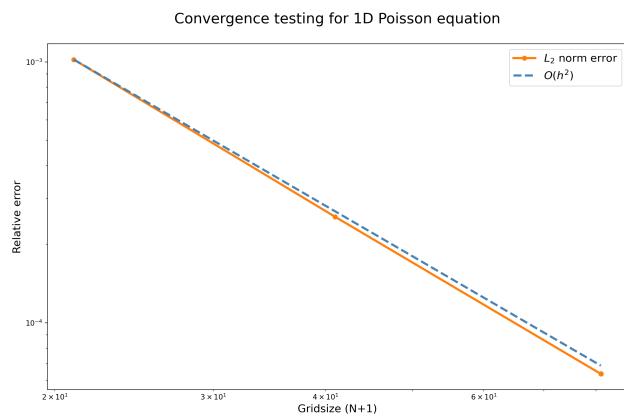
From figure 41 we see that the numerical solution to the Poisson equation converges approximately of order 2. However, we see from the plots that the solution differs from the $\mathcal{O}(h^2)$ line for low M , but seem parallel for higher M .

From figure 42 we see the same tendencies as in figure 41, i. e. that the solution converges of order 2 for sufficiently large M , which makes sense knowing that the expressions in the problems resulting in the figures 41 and 42 has the same shape but coefficients of different magnitude.

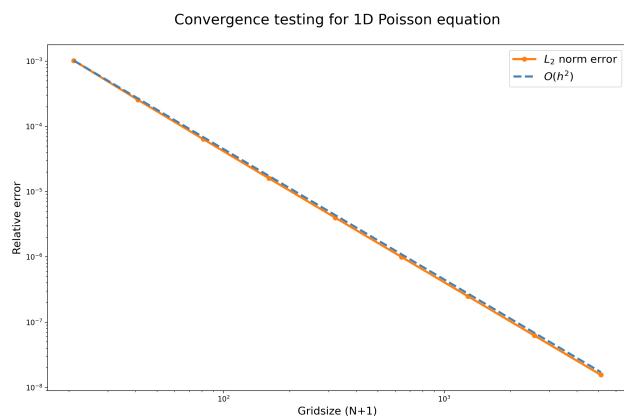
Figure 43 shows that the solution to the Poisson equation applied on the conditions leading to the analytical solution (1.5.16) results in approximately second order convergence for adaptive grid refinement but only first order for uniform refinement. The reason is likely the the analytical solution to this problem is not as regular as for the others [8].



(a) Uniform Refinement

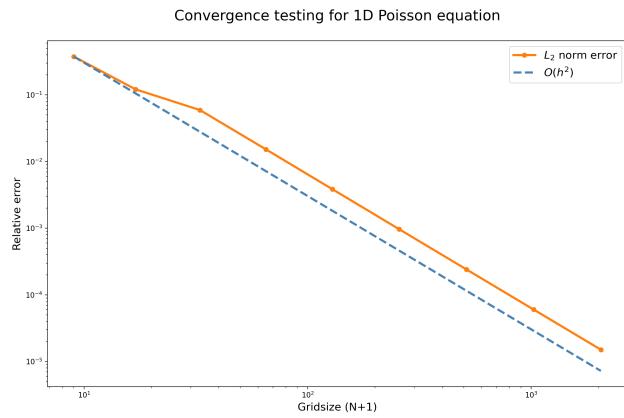


(b) Adaptive Refinement Average Strategy, $\alpha = 1$

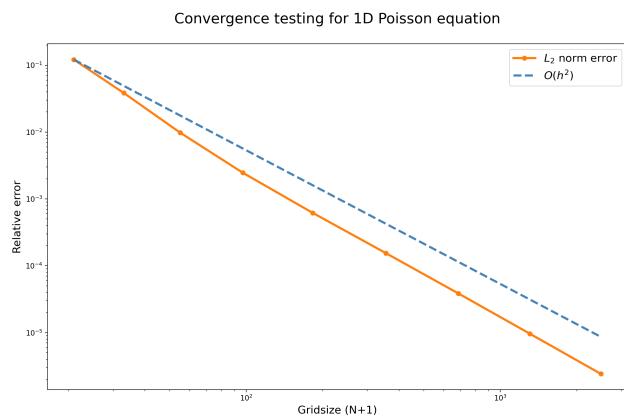


(c) Adaptive Refinement Maximum Strategy, $\alpha = 0.7$

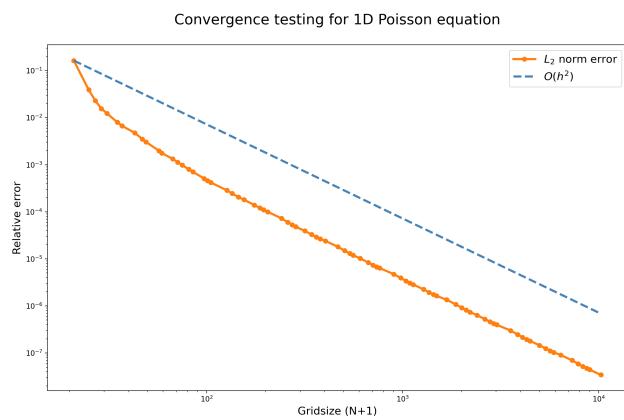
Figure 40: Convergence plots for the Poisson equation with conditions regarding the analytical solution from (1.5.13).



(a) Uniform Refinement

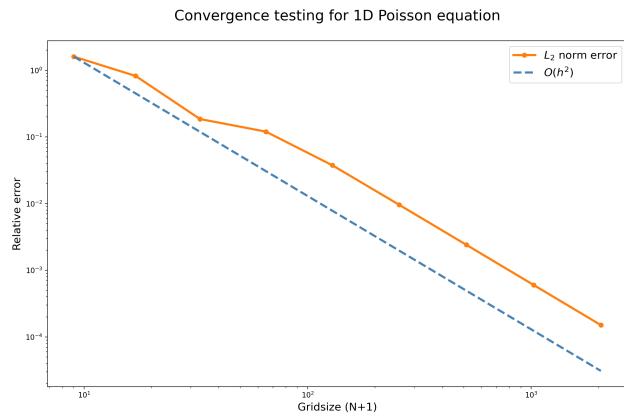


(b) Adaptive Refinement Average Strategy, $\alpha = 1$

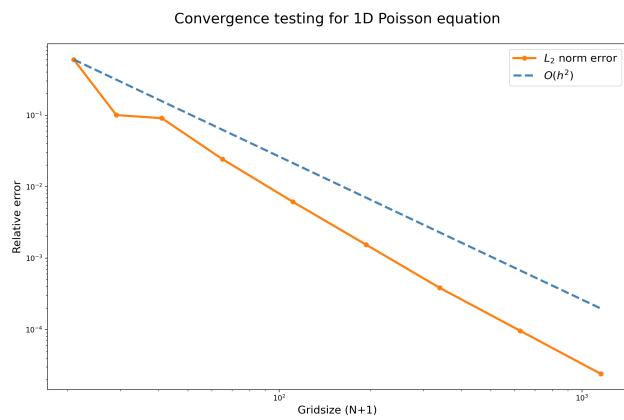


(c) Adaptive Refinement Maximum Strategy, $\alpha = 0.7$

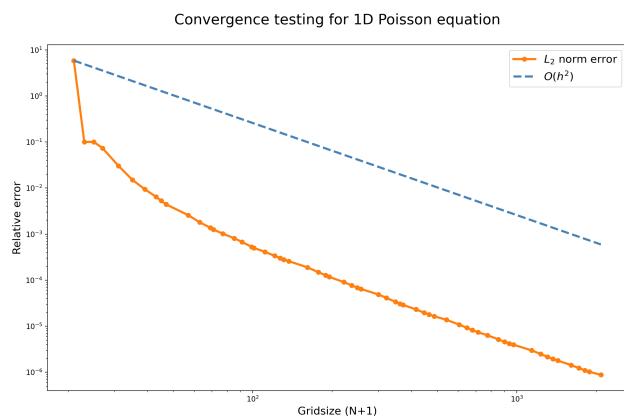
Figure 41: Convergence plots for the Poisson equation (1.5.1) with conditions regarding the analytical solution from (1.5.14).



(a) Uniform Refinement

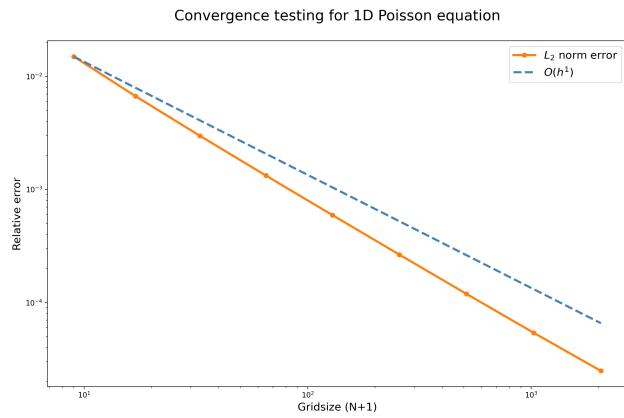


(b) Adaptive Refinement Average Strategy, $\alpha = 1$

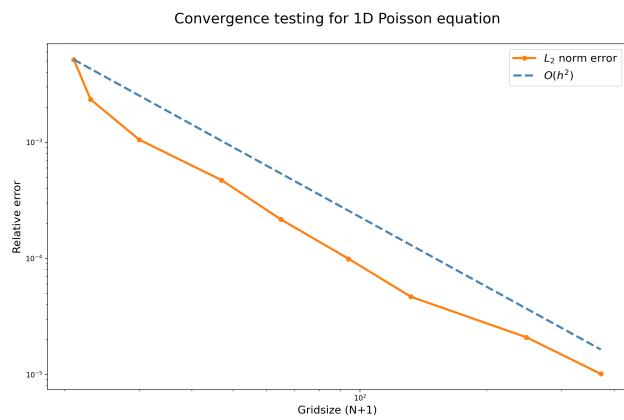


(c) Adaptive Refinement Maximum Strategy, $\alpha = 0.7$

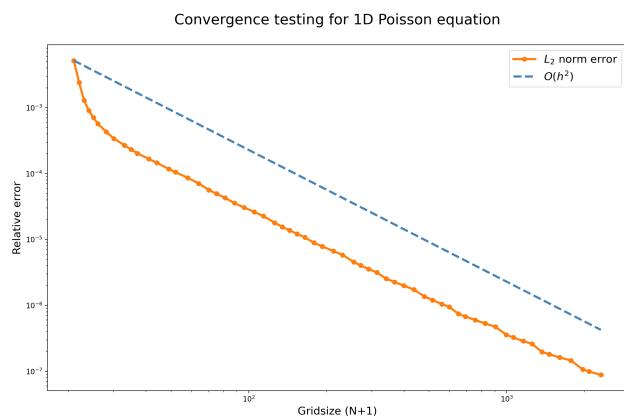
Figure 42: Convergence plots for the Poisson equation with conditions regarding the analytical solution from (1.5.15).



(a) Uniform Refinement



(b) Adaptive Refinement Average Strategy, $\alpha = 1$



(c) Adaptive Refinement Maximum Strategy, $\alpha = 0.7$

Figure 43: Convergence plots for the Poisson equation with conditions regarding the analytical solution from (1.5.16).

subfig

Project Part 2 - Wave Equation

In this part we will consider the wave equation in one and two dimensions, and for each problem solve it analytically and numerically for given initial and boundary conditions as well as finding its conditions for stability.

2.1. 1D Wave Equation by FDM

The one-dimensional wave equation we will be solving is given by

$$u_{tt}(x, t) = c^2 u_{xx}(x, t), \quad u(x, 0) = g(x), \quad u_t(x, 0) = h(x), \quad u(x + 1, t) = u(x, t), \quad (2.1.1)$$

for $x \in [0, 1]$ and $t > 0$. We will consider two different sets of initial conditions:

$$g(x) = \cos(4\pi x), \quad h(x) = 0 \quad (2.1.2)$$

$$g(x) = \exp(-100(x - 1/2)^2), \quad h(x) = 0 \quad (2.1.3)$$

2.1.1. Mathematical Formulation

2.1.1.1. Difference Scheme

In the first part of the problem, we will take the difference scheme of the wave equation

$$\frac{U_j^{n+1} - 2U_j^n + U_j^{n-1}}{k^2} = c^2 \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{h^2} \quad (2.1.4)$$

into account and rewrite it when $n = 0$ with U_j^{-1} in terms of the initial condition $u_t(x, 0) = h(x)$. This is done by first expressing the central difference scheme for $u_t(x, 0)$ when $n = 0$ as

$$u_t(x, 0) = \frac{U_j^1 - U_j^{-1}}{2k} = h(x)$$

and then solve it for U_j^{-1} such that

$$U_j^{-1} = U_j^1 - 2k \cdot h(x). \quad (2.1.5)$$

Inserting (2.1.5) into (2.1.4) for $n = 0$, we obtain

$$\frac{2U_j^1 - 2U_j^0 - 2k \cdot h(x)}{k^2} = c^2 \frac{U_{j+1}^0 - 2U_j^0 + U_{j-1}^0}{h^2}. \quad (2.1.6)$$

2.1.1.2. Von Neumann Stability Analysis

We now want to find the stability criteria for the method, i.e. the condition for $r = ck/h$ for which $|\xi| \leq 1$. By substituting $U_j^n = \xi^n \exp(\frac{2\pi i \beta j}{M+1})$ and inserting it into (2.1.4) we get

$$\xi + \xi^{-1} - 2 = r^2 \left[\exp\left(\frac{2\pi i \beta}{M+1}\right) + \exp\left(-\frac{2\pi i \beta}{M+1} - 1\right) \right] = 2r^2 \left[\cos\left(\frac{2\pi \beta}{M+1}\right) - 1 \right].$$

Further we let

$$\sigma = r^2 \left[\cos\left(\frac{2\pi \beta}{M+1}\right) - 1 \right] = -2r^2 \sin^2\left(\frac{\pi \beta}{M+1}\right) \quad (2.1.7)$$

such that

$$\xi^2 - 2(1 + \sigma)\xi + 1 = 0,$$

implying that ξ has roots given by

$$\xi = 1 + \sigma \pm \sqrt{\sigma^2 + 2\sigma}.$$

This means that we have real roots for $\sigma^2 + 2\sigma > 0$, i.e. for $\sigma < -2$. We observe that in our case $1 + \sigma - \sqrt{\sigma^2 + 2\sigma} > -1$ which regarding to the stability criteria $|\xi| \leq 1$ implies that the method is unstable for real roots, i.e. when $\sigma < -2$. By taking complex roots into account we obtain the following expression for ξ :

$$\xi = 1 + \sigma \pm i\sqrt{-\sigma^2 - 2\sigma}$$

such that

$$\begin{aligned} |\xi|^2 &= (1 + \sigma)^2 + (\sqrt{-\sigma^2 - 2\sigma})^2 \\ &= (1 + \sigma)^2 + (-\sigma^2 - 2\sigma) \\ &= 1 + 2\sigma + \sigma^2 - \sigma^2 - 2\sigma \\ &= 1. \end{aligned}$$

For $\sigma = 2$ we observe that $\xi = -1$ meaning that stability is satisfied for $\sigma \geq -2$. Applying (2.1.7) combined with this we get

$$-2r^2 \sin^2 \left(\frac{\pi\beta}{M+1} \right) \geq -2.$$

Solving this inequality for r we get

$$|r| \leq \frac{1}{\left| \sin \left(\frac{\pi\beta}{M+1} \right) \right|}, \quad (2.1.8)$$

and since $|\sin(x)| \in [0, 1] \forall x$, the condition

$$|r| \leq 1, \quad (2.1.9)$$

satisfies (2.1.8), which is the condition for r such that $|\xi| \leq 1$ for any $\beta \in \mathbb{Z}$, which we wanted to find.

2.1.1.3. Analytical Solution using Separation of Variables

We will now derive the analytical solution to the wave equation with initial conditions (2.1.2) by separating the equation such that

$$u(x, t) = X(x) \cdot T(t)$$

leading to

$$\begin{aligned} u_{tt}(x, t) &= T''(t) \cdot X(x) = c^2 X''(x) \cdot T(t) = c^2 u_{xx}(x, t), \\ \frac{X''(x)}{X(x)} &= \frac{1}{c^2} \frac{T''(t)}{T(t)} = \lambda^2, \\ X''(x) - \lambda^2 X(x) &= 0, \quad T''(t) - c^2 \lambda^2 T(t) = 0. \end{aligned}$$

λ can now be decided as $\lambda = 0$ for a linear solution, $\lambda = \omega^2 > 0$ for an exponential solution, and $\lambda = -\omega^2$ for a periodic solution, where the last choice of lambda will fit the best with our periodic boundary condition $u(x+1, t) = u(x)$. We have then so far got the following expressions for $X(x)$ and $T(t)$:

$$T(t) = d_1 \cos(\lambda ct) + d_2 \sin(\lambda ct), \quad X(x) = d_3 \cos(\lambda x) + d_4 \sin(\lambda x).$$

By further applying the boundary condition $u_t(x, 0) = 0$, we can see that $d_2 = 0$. We set $\lambda = n\pi$ and $c = 1$, such that

$$u_n(x, t) = (d_3 \cos(n\pi x) + d_4 \sin(n\pi x)) d_1 \cos(n\pi t).$$

From here we can apply the boundary condition $u(x, 0) = \cos(4\pi x)$ and get

$$d_1 d_3 \cos(n\pi x) + d_1 d_4 \sin(n\pi x) = \cos(4\pi x).$$

From here we can see that setting $d_1 d_4 \sin(n\pi x) = 0$, $n = 4$ and $d_1 d_3 = 0$ such that our analytical solution becomes

$$u(x, t) = \cos(4\pi x) \cdot \cos(4\pi t),$$

which satisfies our initial value conditions and is a suitable solution for the equation. This can be rewritten using trigonometric identities to

$$u(x, t) = \frac{1}{2} (\cos(4\pi(x + t)) + \cos(4\pi(x - t))). \quad (2.1.10)$$

2.1.2. Numerical Results

As we see from figure 44, we initially start out with one wave with its wavetop at $x = 0.5$. It then begins to sink together as timestep t approaches 700, and at $t = 1100$ it has split into two waves propagating away from each other. Furthermore, after reaching its boundaries at $x = 0$ and $x = 1$, at $t = 4300$, the propagation changes direction, and the wave movement is a reflected motion of what we observed until it reached its boundaries, which is a consequence of the periodic boundary condition.

Applying the initial conditions (2.1.2) and the analytical solution, convergence testing was done using the relative ℓ_2 -error, and we expect the convergence rate to be of second order since the truncation error of the second order central finite difference scheme is of second order [4].

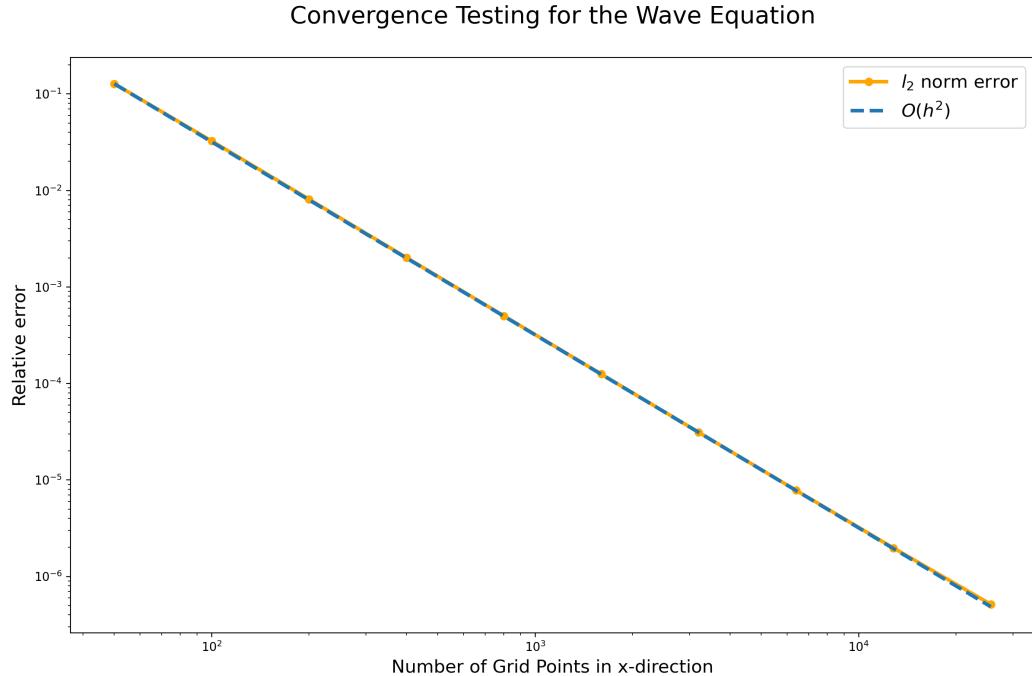


Figure 45: Convergence plot for the 1D Wave equation.

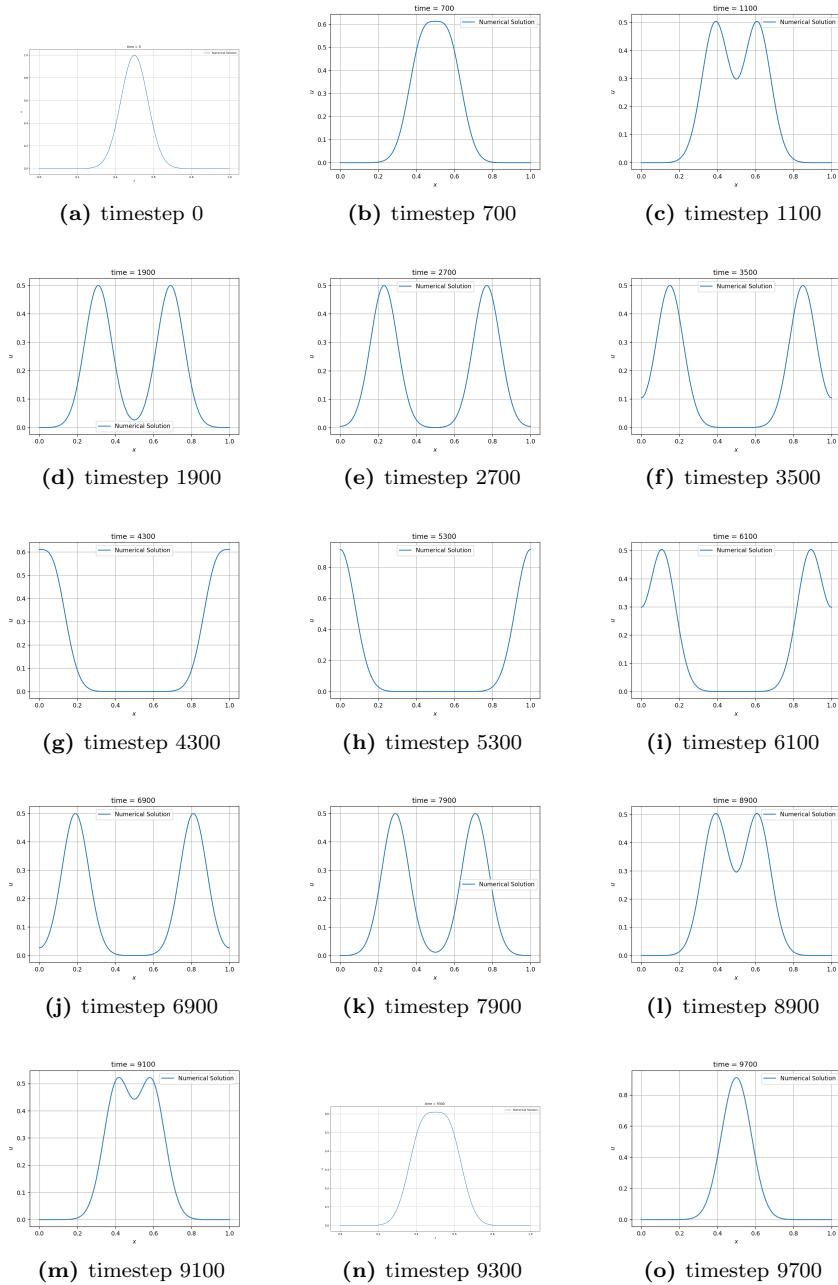


Figure 44: Numerical solution to the 1D Wave equation for different timesteps.

2.2. Wave Equation in Two Dimensions

We will in this section consider the two-dimensional wave equation given by

$$u_{tt}(x, y, t) = c^2(u_{xx}(x, y, t)) + u_{yy}(x, y, t)) , \quad (2.2.1)$$

with initial conditions

$$u(x, y, 0) = \cos(4\pi x) \sin(4\pi y) , \quad u_t(x, y, 0) = 0 \quad (2.2.2)$$

and boundary conditions

$$u(x+1, y, t) = u(x, y, t) , \quad u(x, y+1, t) = u(x, y, t) \quad (2.2.3)$$

2.2.1. Mathematical Formulation

2.2.1.1. Von Neumann Stability Analysis

Using the difference scheme

$$\frac{U_{j,l}^{n+1} - 2U_{j,l}^n + U_{j,l}^{n-1}}{k^2} = c^2 \left(\frac{U_{j+1,l}^n - 2U_{j,l}^n + U_{j-1,l}^n}{h_x^2} + \frac{U_{j,l+1}^n - 2U_{j,l}^n + U_{j,l-1}^n}{h_y^2} \right), \quad (2.2.4)$$

the identity

$$U_{j,l}^n = \xi^n \exp \left(2\pi i \beta_1 \frac{j}{M+1} \right) \exp \left(2\pi i \beta_2 \frac{l}{N+1} \right), \quad (2.2.5)$$

as well as $r_x = ck/h_x$ and $r_y = ck/h_y$, we obtain, using the same calculations as in 1D, the equation

$$\xi - 2 + \xi^{-1} = -4 \left[r_x^2 \sin^2 \left(\frac{\pi \beta_1}{M+1} \right) + r_y^2 \sin^2 \left(\frac{\pi \beta_2}{N+1} \right) \right].$$

doing the same analysis as in 1D we obtain the inequality

$$r_x^2 \sin^2 \left(\frac{\pi \beta_1}{M+1} \right) + r_y^2 \sin^2 \left(\frac{\pi \beta_2}{N+1} \right) \leq 1, \quad (2.2.6)$$

or

$$\frac{r_x^2}{\left[\sin^2 \left(\frac{\pi \beta_1}{M+1} \right) \right]^{-1}} + \frac{r_y^2}{\left[\sin^2 \left(\frac{\pi \beta_2}{N+1} \right) \right]^{-1}} \leq 1 \quad (2.2.7)$$

This represents all the values within an ellipse in the r_x, r_y -plane with the center in the origin and half axes from $-\left[\sin \left(\frac{\pi \beta_1}{M+1} \right) \right]^{-1}$ to $\left[\sin \left(\frac{\pi \beta_1}{M+1} \right) \right]^{-1}$ along the r_x -axis and from $-\left[\sin \left(\frac{\pi \beta_2}{N+1} \right) \right]^{-1}$ to $\left[\sin \left(\frac{\pi \beta_2}{N+1} \right) \right]^{-1}$ along the r_y -axis, and are the values of r_x and r_y for which $|\xi| \leq 1$ and the method is stable. However, since $|\sin(x)| \leq 1 \forall x$, the condition

$$r_x^2 + r_y^2 \leq 1 \quad (2.2.8)$$

satisfies (2.2.6), which is the unit circle in the r_x, r_y -plane, where the method is stable for values of r_x and r_y within it.

2.2.1.2. Analytical Solution by Separation of Variables

We now wish to derive the analytical solution for the two-dimensional problem. Similar to the problem in 1D, we separate $u(x, y, t)$ as

$$u(x, y, t) = X(x) \cdot Y(y) \cdot T(t)$$

such that

$$\begin{aligned} T''(t) \cdot X(x) \cdot Y(y) &= c^2(T(t) \cdot X''(x) \cdot Y(y) + T(t) \cdot X(x) \cdot Y''(y)) \\ \frac{1}{c^2} \frac{T''(t)}{T(t)} &= \frac{X''(x)}{X(x)} + \frac{Y''(y)}{Y(y)} = \lambda^2 \\ T''(t) - c^2 \lambda^2 T(t) &= 0 ; \quad \frac{X''(x)}{X(x)} = -\frac{Y''(y)}{Y(y)} + \lambda^2. \end{aligned}$$

Introducing μ and ν such that $\nu^2 = \lambda^2 + \mu^2$ we get

$$X''(x) - \mu^2 X(x) = 0 , \quad Y''(y) - \nu^2 Y(y) = 0 , \quad T''(t) - c^2 \lambda^2 T(t) = 0. \quad (2.2.9)$$

By applying the periodic boundary conditions (2.2.3), we obtain the following solutions to (2.2.9)

$$X(x) = X_m(x) = \sin(\mu_m x) + \cos(\mu_m x), \quad (2.2.10)$$

$$Y(y) = Y_n(y) = \sin(\nu_n y) + \cos(\nu_n y), \quad (2.2.11)$$

$$T(t) = T_{mn}(t) = B_{mn}^* \sin(\lambda_{mn} t) + B_{mn} \cos(\lambda_{mn} t), \quad (2.2.12)$$

where

$$\mu_m = m\pi/L_x , \quad \nu_n = n\pi/L_y , \quad \lambda_{mn} = c\sqrt{\mu_m^2 + \nu_n^2} = c\pi\sqrt{m^2 + n^2}.$$

Knowing that $c = 1$, and $L_x = L_y = 1$, we get by inserting (2.2.10), (2.2.11) and (2.2.12) into (2.2.9), we get

$$u_{mn}(x, y, t) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} (\sin(m\pi x) + \cos(m\pi x)) \cdot (\sin(n\pi y) + \cos(n\pi y)) \cdot (B_{mn}^* \sin(\lambda_{mn} t) + B_{mn} \cos(\lambda_{mn} t)).$$

The initial conditions (2.2.2) gives that

$$\sum_{n=1}^{\infty} \sum_{m=1}^{\infty} B_{mn} (\sin(m\pi x) + \cos(m\pi x)) (\sin(n\pi y) + \cos(n\pi y)) = \cos(4\pi x) \sin(4\pi y)$$

and

$$\sum_{n=1}^{\infty} \sum_{m=1}^{\infty} B_{mn}^* (\sin(m\pi x) + \cos(m\pi x)) (\sin(n\pi y) + \cos(n\pi y)) = 0. \quad (2.2.13)$$

A plausible solution to (??) is $B_{mn}^* = 0$ for all $m, n \in \mathbb{N}$. Furthermore, by looking at (2.2.13), a plausible solution is setting the terms $\sin(m\pi x) = \cos(n\pi y) = 0$, as well as $B_{mn} = 1$, and $m = n = 4$, such that our analytical solution becomes

$$u(x, y, t) = \cos(4\pi x) \cdot \sin(4\pi y) \cdot \cos(\sqrt{32}\pi t). \quad (2.2.14)$$

This satisfies our equation with its boundary conditions and is therefore a satisfactory analytical solution to our problem.

2.2.2. Numerical results

2.2.2.1. 2D Wave Equation

The 3D plots from figure 46 clearly display the periodicity of the numerical solution.

Using the initial condition (2.2.2) we again plot the relative l_2 -rror against the number of grid points in space, M . The expected convergence rate is of order 2 for the same reason as in the 1D case. In reality however, we observe that the convergence rate is slightly below the theoretical second order. The reason is most likely that the grid spacings we used are not fine enough to achieve the asymptotic convergence rates. Since the problem is two-dimensional and the numerical solver relies on an iterative method, using a high number of grid points in space is infeasible as it leads to a very high running time for the numerical solver.

The plot in figure 48 is also linear and is similar to the plot 47 which is expected since the number of degrees of freedom is proportional to the number of grid points in space.

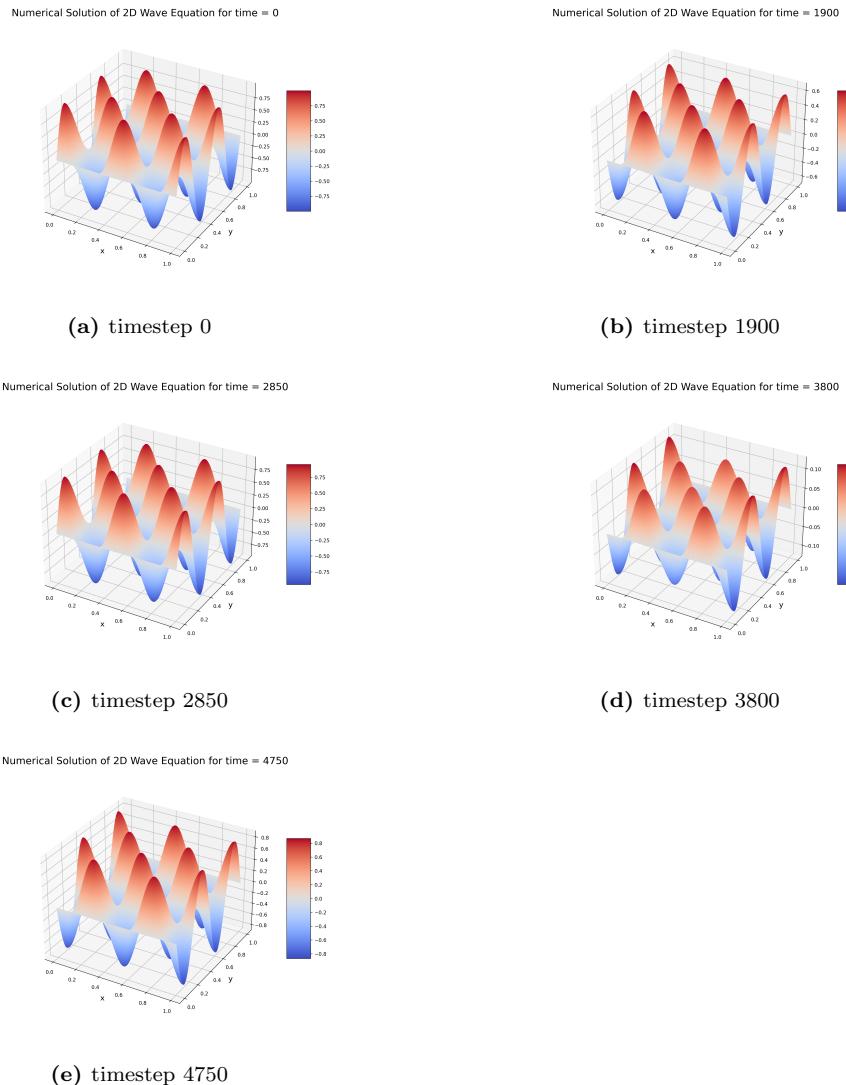


Figure 46: 3D Plots of the numerical solution of the 2D Wave equation for different timesteps.

Convergence testing for 2D Wave Equation

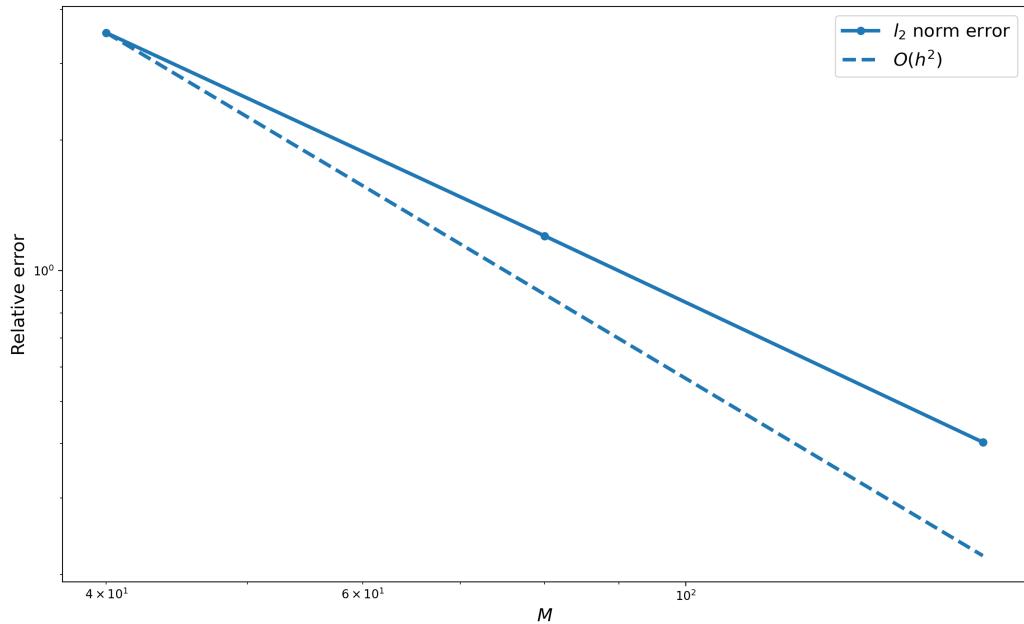


Figure 47: Convergence plot for the 2D wave equation.

Convergence testing for 2D Wave Equation

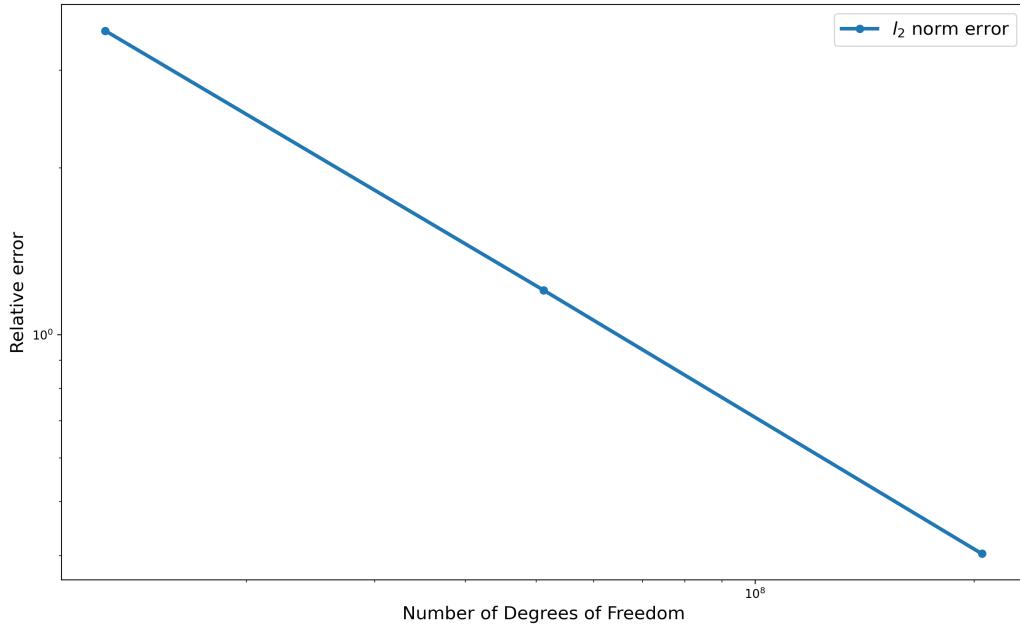


Figure 48: Relative ℓ_2 -error plotted against number of degrees of freedom.

2.2.2.2. Computational time

For each iteration, we doubled the number of grid points in x - and y -direction, which means the computational time increases massively for each iteration. This explains the steep increase in computational time as the number of degrees of freedom increases.

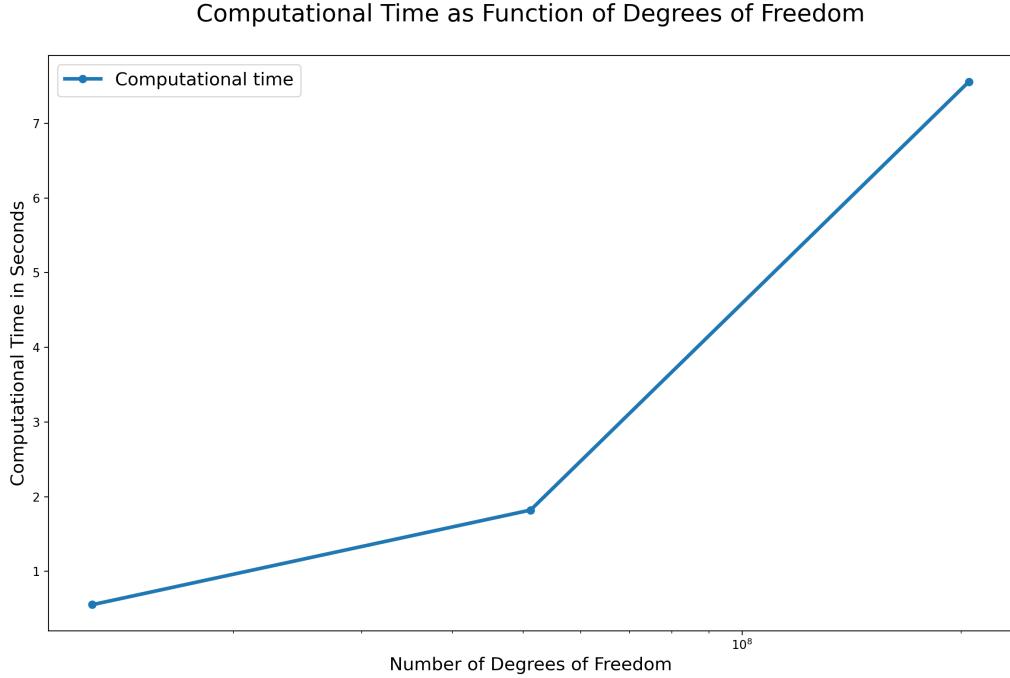


Figure 49: Computational time plotted against number of degrees of freedom.

Conclusion

We have now solved various PDEs applying the finite difference method and one PDE applying the finite element method. In all cases, our numeric solvers have worked well, and for the most part we have been able to achieve the expected theoretical convergence rates.

One issue we ran into was the high running time for the numerical solver of the 2-dimensional wave equation. Due to this, we were not able to do as much grid refinement as we would have liked for this problem since the computational processing power we had available was not sufficient. The high running time for the solver was expected as it uses an explicit iterative approach, but still made the convergence analysis difficult. We made sure to utilize vectorization and avoid loops as much as possible in our code to decrease the running time.

All in all, we are satisfied with our results, and we are ready to apply what we have learnt during the project for future challenges.

References

- [1] T. Kvamsdal, Y. Suzuki, and A. Abdulhaque, “Tma4212 finite difference method spring 2021 semester project part 1,” no. 1, pp. 1–6.
- [2] T. Kvamsdal, Y. Suzuki, and A. Abdulhaque, “Tma4212 finite difference method spring 2021 semester project part 2,” no. 2, pp. 7–10.
- [3] J. W. Thomas, *Numerical Partial Differential Equations: Finite Difference Methods*. No. 22.
- [4] B. Owren, “Tma4212 numerical solution of partial differential equations with finite difference methods,” pp. i–iv + 1–98.

- [5] A. Salih, “Inviscid burgers’ equation,” pp. 1–19.
- [6] T. S. community, “scipy.integrate.solve_ivp,”
- [7] T. Kvamsdal and A. Abdulhaque, “Tma4212 bvp,” pp. 1–21.
- [8] O. Ghattas and U. Villa, “Computational and variational methods for inverse problems,”