

# INF5620 Project: 2D wave equation

Håkon Østerbø og Jarle Sogn  
Institutt for matematikk  
Universitetet i Oslo  
haakonoo@math.uio.no  
jarlesog@math.uio.no

October 17, 2012

## Introduction

The goal for this project is to solve the two-dimensional, standard, linear wave equation with damping by an explicit numeric scheme (see equation 1). With the Neumann boundary condition  $\frac{\partial u}{\partial n} = 0$ . The initial conditions are shown in equation 2 and 3. We consider a rectangular domain  $[0, L_x] \times [0, L_y]$ .

$$\frac{\partial^2 u}{\partial t^2} + b \frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left( q(x, y) \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( q(x, y) \frac{\partial u}{\partial y} \right) + f(x, y, t) \quad (1)$$

$$u(x, y, 0) = I(x, y) \quad (2)$$

$$u_t(x, y, 0) = V(x, y) \quad (3)$$

As our extension of the project, we solve the problem in a C++ program. All calculations are done in C++ and then the program writes the data to files. For simplicity we have made a python script that then calls the C++ program and plots the data files. After it's done it deletes the data files.

## Structure of the C++ program

### Stability criteria

$$1 \geq c \Delta t \left( \frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right)^{\frac{1}{2}} \quad (4)$$

# 1 Verification

All numeric schemes and implementation should be verified and ours is no exception. There are several different ways of verifying. What's important to have in mind is that the PDE can be simplified by changing the parameters, this is both a good and a bad thing. For example by setting  $b = 0$ , the PDE no longer have a damping effect. It's now more easy to perform verification tests. However if we forget to test for  $b \neq 0$ , we will not know if the damping is implemented correctly. We should also test for  $L_x \neq L_y$ ,  $N_x \neq N_y$  and  $f(x, y), q(x, y) \neq \text{const}$  to make a thorough verification.

## 1.1 Constant solution

By setting  $I(x, y) = 1$ ,  $V(x, y)$  and  $f(x, y, t) = 0$ . We easily see from equation 1 that  $u(x, y, t) = 1$ , hence a constant solution. When we simulated this, we ran into problem at the corners that were not equal to one. This verification test quickly revealed that our corner point  $(u_{0,0}, u_{N_x,0}, u_{N_x,N_y}$  and  $u_{0,N_y})$  were not being updated correctly.

## 1.2 Exact 1D solution

To obtain a exact 1D solution, we set  $I(x, y) = 1$  for  $x \in [0.4, 0.6]$  and zero elsewhere. We also set  $c \frac{\Delta t}{\Delta x} = 1$ . This solution is independent of  $y$  (similar thing can be done for  $y$  and then the solution is independent of  $x$ ). After doing this, we discovered some new bugs (we mixed  $N_x$  and  $N_y$  some places in the code) and debugged them.

## 1.3 Standing wave

## 1.4 Manufactured solution

In this verification method we normally chose a linear function  $I(x, y)$  that satisfy the B.C. and is a stationary solution  $u(x, y)$ . Then we calculate what  $f(x, y)$  have to be in order to obtain the stationary solution  $u(x, y) = I(x, y)$ . We don't have any time dependence in this problem, so we are actually just solving the Poisson equation. If there is an error, it must then be in the right part of equation 1. Since we are working with Neumann B.C and not Dirichlet B.C. a linear  $I(x, y)$  will not work. We have to use a cubic  $I(x, y)$  to satisfy the B.C., see equation 5.

$$I(x, y) = \left(\frac{x}{3} - \frac{L_x}{2}\right) x^2 \left(\frac{y}{3} - \frac{L_y}{2}\right) y^2 \quad (5)$$

We should check if this give us exact numeric differentiation. For the  $\frac{\partial^2 f}{\partial x^2}$  we used this scheme:

$$\frac{\partial^2 f}{\partial x^2} \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

By inserting  $f(x) = x^3$  we get:

$$\frac{x^3 + 3x^2h + 3xh^2 + h^3 - 2x^3 + x^3 - 3x^2h + 3xh^2 - h^3}{h^2} = 6x$$

which is the exact solution (we do the same for  $x$  and  $x^2$ ). We also use center difference at the boundary.

$$\frac{\partial f}{\partial x} \approx \frac{f(x+h) - f(x-h)}{2h}$$

$$\frac{x^3 + 3x^2h + 3xh^2 + h^3 - x^3 + 3x^2h - 3xh^2 + h^3}{2h} = 3x^2 + h^2$$

This is not exactly equal to the analytical. Our numeric solution should then be equal to the analytic to machine in the interior at  $t = 0$ . The center of the domain will also be exact for low values of  $t$ . Last thing to do now is to find  $f(x, y)$  that gives us the wanted solution:

$$f(x, y) = (L_x - 2x) \left( \frac{y}{3} - \frac{L_y}{2} \right) y^2 + (L_y - 2y) \left( \frac{x}{3} - \frac{L_x}{2} \right) x^2$$

As I mention above: We only verify the Poisson equation, since this is a stationary solution. If we use the same  $u(x, y, t)$  and multiply with  $0.7t + 0.2$ , we get a non-stationary manufactured solution.  $I(x, y)$  is then equation 5 multiplied with 0.2.  $V(x, y)$  is then equation 5 multiplied with 0.7.  $f(x, y) = bu_t - u_{xx} - u_{yy}$ .

$$f(x, y) = \left( \frac{x}{3} - \frac{L_x}{2} \right) x^2 \left( \frac{y}{3} - \frac{L_y}{2} \right) y^2 \left[ 0.7b - (0.7t + 0.2) \left( \frac{2x - L_x}{\frac{x}{3} - \frac{L_x}{2} x^2} + \frac{2y - L_y}{\frac{y}{3} - \frac{L_y}{2} y^2} \right) \right]$$

In the code we used backward Euler  $u_t(x, y, t) \approx \frac{u(x, y, t) - u(x, y, t - \Delta t)}{\Delta t}$ . It can easily be verified that this give a exact differentiation for our chose of  $u(x, y, t)$ .