1 Introduction

In this text we will addresses the two-dimensional, standard, linear wave equation, with damping and reflecting boundaries. We will develop a scheme for solving it and apply it to the problem of a tsunami over an uneven seabed. Especial focus will be put on the question of which kinds of seabeds causes numerical instability.

2 Mathematical model

For x,y in the domain $D = [0, L_y] \times [0, L_x]$ and $t \in [0, T]$ we will study the following set of equations:

$$u_t t + bu_t = (q(x,y)u_x)_x + (q(x,y)u_y)_y + f(x,y,t), \qquad x,y \in D, t \in [0,T] \quad (1)$$

$$\frac{\delta u}{\delta n} = 0 \qquad x,y \in \delta D, t \in [0,T] \quad (2)$$

$$u(x,y,0) = I(x,y) \qquad x,y \in \delta D \quad (3)$$

$$u_t(x,y,0) = V(x,y) \qquad x,y \in \delta D \quad (4)$$

3 Numerical scheme

3.1 Numerical scheme for the inner points

In operator notation we want to use the following scheme for inner points:

$$[D_t D_t u + bD_2 tu = D_x(qD_x u) + D_y(qD_y u) + f]^n_{i\,j} \quad (5)$$

Calculating each summand we find:

$$[D_t D_t u]^n_{i\,j} = \frac{u^{n+1}_{i,j} - 2u^n_{i,j} + u^{n-1}_{i,j}}{\Delta t^2} \quad (6)$$

$$[bD_2 tu]^n_{i\,j} = b\frac{u^{n+1}_{i,j} - u^{n-1}_{i,j}}{2\Delta t} \quad (7)$$

$$[D_x qD_x u]^n_{i\,j} = \frac{q_{i+\frac{1}{2},j}(u^n_{i+1,j} - u^n_{i,j}) - q_{i-\frac{1}{2},j}(u^n_{i,j} - u^n_{i-1,j})}{\Delta x^2} \quad (8)$$

$$[D_y qD_y u]^n_{i\,j} = \frac{q_{i,j+\frac{1}{2}}(u^n_{i,j+1} - u^n_{i,j}) - q_{i,j-\frac{1}{2}}(u^n_{i,j} - u^n_{i,j-1})}{\Delta x^2} \quad (9)$$

$$[f]^n_{i\,j} = f(x_i, y_i, t_i) \quad (10)$$

Inserting these expressions in (5) and solving for $u^{n+1}_{i,j}$ we get the following scheme for inner points:

$$\begin{aligned}
u^{n+1}_{i,j} = {} & \frac{2u^n_{i,j} + (\frac{1}{2}b\Delta t - 1.0)u^{n-1}_{i,j}}{1 + \frac{1}{2}b\Delta t} \\
& + \frac{\Delta t^2}{\Delta x^2(\frac{1}{2}b\Delta t - 1.0)}(q_{i+\frac{1}{2},j}(u^n_{i+1,j} - u^n_{i,j}) - q_{i-\frac{1}{2},j}(u^n_{i,j} - u^n_{i-1,j})) \\
& + (\frac{\Delta t^2}{\Delta y^2(\frac{1}{2}b\Delta t - 1.0)}(q_{i,j+\frac{1}{2}}(u^n_{i,j+1} - u^n_{i,j}) - q_{i,j-\frac{1}{2}}(u^n_{i,j} - u^n_{i,j-1})) \\
& + \frac{\Delta t^2}{\frac{1}{2}b\Delta t - 1.0}f(x_i, y_i, t_i)
\end{aligned}$$

3.2 Numerical scheme for the first time step

For the first time step $u^{n-1}_{i,j}$ isn't a part of the grid. To circumvent this problem we have to apply the intial condition $u_t(x, y, 0) = V(x, y)$ for $x, y \in \delta D$. We discretize it and get:

$$[D_{2t}u = V]^0_{i\,j} \implies \frac{u^1_{i\,j} - u^{-1}_{i\,j}}{2\Delta t} = V(x_i, y_j) \implies u^{-1}_{i\,j} = u^1_{i\,j} - 2\Delta t V(x_i, y_j)$$

Inserting this in equation (6) and (12) we get:

$$[D_tD_tu]^n_{i\,j} = 2(\frac{u^{n+1}_{i,j} - u^n_{i,j}}{\Delta t^2} - \frac{V(x_i, y_i)}{\Delta t}) \tag{11}$$

$$[bD_2tu]^n_{i\,j} = b\frac{V(x_i, y_i)}{\Delta t} \tag{12}$$

Inserting these expressions in (5) and solving for $u^{n+1}_{i,j}$ we get the following scheme for inner points in the first time step:

$$\begin{aligned}
u^{n+1}_{i,j} = {} & u^n_{i,j} + (1 - \frac{1}{2}b\Delta t) * \Delta t V^{n-1}_{i,j} \\
& + \frac{\Delta t^2}{2\Delta x^2}(q_{i+\frac{1}{2},j}(u^n_{i+1,j} - u^n_{i,j}) - q_{i-\frac{1}{2},j}(u^n_{i,j} - u^n_{i-1,j})) \\
& + (\frac{\Delta t^2}{\Delta 2y^2}(q_{i,j+\frac{1}{2}}(u^n_{i,j+1} - u^n_{i,j}) - q_{i,j-\frac{1}{2}}(u^n_{i,j} - u^n_{i,j-1})) \\
& + \frac{\Delta t^2}{2}f(x_i, y_i, t_i)
\end{aligned}$$

3.3 Numerical scheme for the boundary

On the boundary this scheme won't work as one, when adding or subtracting one from one of the indices, risks leaving grid. We discretize the Neumann condition to solve this problem:

$$[D_{2x}u]^n_{0\,j} = [D_{2x}u]^n_{Nx\,j} = 0 \quad \text{for j = 0,1..}N_y\text{, n = 0,1,..}N_t \tag{13}$$

$$[D_{2y}u]^n_{i\,0} = [D_{2y}u]^n_{0,Ny} = 0 \quad \text{for i = 0,1..}N_y\text{, n = 0,1,..}N_t \tag{14}$$

Which leads to the equations:

$$u^n_{1,j} = u^n_{-1,j}, u^n_{N_x-1,j} = u^n_{Nx+1,j} \quad \text{for j = 0,1..}N_y\text{, n = 0,1,..}N_t \tag{15}$$

$$u^n_{i,1} = u^n_{i,-1}, u^n_{i,Ny+1} = u^n_{i,Nx+1} \quad \text{for i = 0,1..}N_y\text{, n = 0,1,..}N_t \tag{16}$$

A scheme for the borders can now be found by applying each equality to the inner point scheme for its boundary. The corners are found by applying two equalities, one for each of the adjecent border.

3.4 Approximating q(x,y) outside the grid

To approximate the q function when evaluated outside the grid we will apply the arithmetic and harmonic mean:

$$q_{i+\frac{1}{2},j} = \frac{q_{i,j} + q_{i+1,j}}{2} \tag{17}$$

$$q_{i+\frac{1}{2},j} = 2(\frac{1}{q_{i,j}} + \frac{1}{q_{i+1,j}})^{-1} \tag{18}$$

4 Implementation

We implemented the code in pure python. See the wave2D_du0.py for the full code.

TODO: Show the core of the program in minted enviroment

5 Numerical experiments