Foundations of Fluids Numerical Exercises 2

Billy Hollis

October 2024

We consider the linearised shallow water equations

$$\frac{\partial \eta}{\partial t} + \frac{\partial (Hu)}{\partial x} = 0 \quad \text{and} \quad \frac{\partial u}{\partial t} + \frac{\partial (g\eta)}{\partial x} = 0 \tag{1}$$

where $u = u(x,t)$ is the velocity, $\eta = \eta(x,t)$ is the free surface deviation, $H(x)$ is the rest depth and g is the acceleration due to gravity. We introduce the scales

$$u = U_0 u' \qquad x = L_s x' \qquad t = \left(\frac{L_s}{U_0}\right) t' \qquad \eta = H_{0s}\eta' \qquad H = H_{0s}H' \tag{2}$$

After having dropped the primes for convenience, the first equation remains unchanged. The second equation reads

$$\frac{\partial u}{\partial t} + \frac{H_{0s}}{U_0^2}\frac{\partial (g\eta)}{\partial x} = 0 \tag{3}$$

where we have once again dropped primes for convenience. If we introduce the scale

$$g = \frac{U_0^2}{H_{0s}} g'$$

then we obtain identical equations to (1) which are now dimensionless. Our Riemann problem will consist of equation (1) along with the piecewise-constant initial conditions

$$u(x,0) = \begin{cases} u_l & \text{for } x < 0 \\ u_r & \text{for } x \geq 0 \end{cases} \quad \text{and} \quad \eta(x,0) = \begin{cases} \eta_l & \text{for } x < 0 \\ \eta_r & \text{for } x \geq 0 \end{cases} \tag{4}$$

We now assume that $H(x) = H_0$ is constant and write our system in the form

$$\frac{\partial}{\partial t}\begin{pmatrix} \eta \\ H_0 u \end{pmatrix} + A\frac{\partial}{\partial x}\begin{pmatrix} \eta \\ H_0 u \end{pmatrix} = 0 \tag{5}$$

where

$$A = \begin{pmatrix} 0 & 1 \\ c_0^2 & 0 \end{pmatrix}$$

and $c_0^2 = gH_0$. It is easy to verify that equations (1) and (5) are identical. We now calculate the eigenvalues of A by noting that

$$\det(A - \lambda I) = \begin{vmatrix} -\lambda & 1 \\ c_0^2 & -\lambda \end{vmatrix} = \lambda^2 - c_0^2$$

Setting this equal to zero yields $\lambda_1 = c_0$ and $\lambda_2 = -c_0$. Having found the eigenvalues, we now calculate the corresponding eigenvectors. For λ_1 we have the eigenvector $\mathbf{v} = (v_1, v_2)^T$ where $v_1 = \frac{v_2}{c_0}$. For λ_2, we have $v_1 = -\frac{v_2}{c_0}$. We are free to choose v_2 as $\frac{1}{2}$ in which case we have the matrix of right eigenvalues

$$\mathbf{B} = \frac{1}{2c_0}\begin{pmatrix} 1 & -1 \\ c_0 & c_0 \end{pmatrix} \tag{6}$$

By performing some elementary row operations we obtain the inverse

$$\mathbf{B^{-1}} = \begin{pmatrix} c_0 & 1 \\ -c_0 & 1 \end{pmatrix} \tag{7}$$

From linear algebra principles, we would expect that $\mathbf{B^{-1}AB} = \Lambda$ where

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \tag{8}$$

which is indeed the case. We now define a vector $\mathbf{r} = \mathbf{B^{-1}} \begin{pmatrix} \eta \\ H_0 u \end{pmatrix}$ where we note that the components of $\mathbf{r}$ read $r_1 = c_0\eta + H_0 u$ and $r_2 = H_0 u - c_0\eta$. Making use of the identity $\mathbf{B^{-1}B} = \mathbf{I}$, we write equation (5) as

$$\mathbf{B^{-1}} \frac{\partial}{\partial t} \begin{pmatrix} \eta \\ H_0 u \end{pmatrix} + \mathbf{B^{-1}ABB^{-1}} \frac{\partial}{\partial x} \begin{pmatrix} \eta \\ H_0 u \end{pmatrix} = 0 \tag{9}$$

This simplifies to give

$$\frac{\partial}{\partial t}\mathbf{r} + \Lambda \frac{\partial}{\partial x}\mathbf{r} = 0 \tag{10}$$

Since Λ is diagonal, we have a pair of linear advection equations, the first of which reads

$$\frac{\partial r_1}{\partial t} + \lambda_1 \frac{\partial r_1}{\partial x} = 0 \tag{11}$$

which we solve subject to the initial condition

$$r_1(x,0) = \begin{cases} r_{1l} & \text{for } x < 0 \\ r_{1r} & \text{for } x \geq 0 \end{cases} \tag{12}$$

Geometrically, equation (11) asserts that, in the (x,t) plane, the directional derivative of r_1 is zero in the direction of the vector $(1, \lambda_1)^T$. In other words, r_1 is constant along the characteristic lines satisfying $\frac{dx}{dt} = \lambda_1$. That is, r_1 is constant along the lines $x - \lambda_1 t = C$ where C represents a constant. Hence, advection equations such as (11) simply translate the initial condition and we have

$$r_1(x,t) = r_1(x - \lambda_1 t, 0) \tag{13}$$

In our case, we have

$$r_1(x,t) = \begin{cases} r_{1l} & \text{for } x < c_0 t \\ r_{1r} & \text{for } x \geq c_0 t \end{cases} \tag{14}$$

Analogous working holds for our other linear advection equation which comes from (10) and we obtain

$$r_2(x,t) = \begin{cases} r_{2l} & \text{for } x < -c_0 t \\ r_{2r} & \text{for } x \geq -c_0 t \end{cases} \tag{15}$$

We now use these expressions to solve our original problem. By noting that $u = \frac{1}{2}(r_1 + r_2)/H_0$ and $\eta = \frac{1}{2}(r_1 - r_2)/c_0$, we have

$$H_0 u(x,t) = \frac{1}{2}(r_1(x,t) + r_2(x,t)) = \begin{cases} u_l & \text{for } x < -c_0 t \\ \frac{1}{2}\left[H_0(u_l + u_r) + c_0(\eta_l - \eta_r)\right] & \text{for } -c_0 t \leq x \leq c_0 t \\ u_r & \text{for } x > c_0 t \end{cases} \tag{16}$$

$$\eta(x,t) = \frac{1}{2c_0}(r_1(x,t) - r_2(x,t)) = \begin{cases} \eta_l & \text{for } x < -c_0 t \\ \frac{1}{2}\left[\frac{H_0}{c_0}(u_l - u_r) + (\eta_l + \eta_r)\right] & \text{for } -c_0 t \leq x \leq c_0 t \\ \eta_r & \text{for } x > c_0 t \end{cases} \tag{17}$$

1 Godunov's Method

We will now describe the Godunov method which applies to the linear shallow water equations (1). We can write these equations in the form

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) = 0 \tag{18}$$

where we have $\mathbf{u} = \begin{pmatrix} \eta \\ u \end{pmatrix}$ and $\mathbf{f}(\mathbf{u}) = \begin{pmatrix} Hu \\ g\eta \end{pmatrix}$ where $H = H(x)$ is no longer necessarily constant. We integrate (18) over $x_{j-\frac{1}{2}} < x < x_{j+\frac{1}{2}}$ and $t_n < t < t_{n+1}$. We obtain

$$\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \mathbf{u}(x, t_{n+1})dx - \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \mathbf{u}(x, t_n)dx + \int_{t_n}^{t_{n+1}} \mathbf{u}(x_{j-\frac{1}{2}}, t)dt - \int_{t_n}^{t_{n+1}} \mathbf{u}(x_{j+\frac{1}{2}}, t)dt = 0 \tag{19}$$

We define the cell average

$$\mathbf{U}_j^n = \frac{1}{h_j} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \mathbf{u}(x, t_n)dx$$

Dividing equation (19) by $h_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$ and rearranging gives us

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n + \frac{1}{h_j}\left[\int_{t_n}^{t_{n+1}} \mathbf{f}(\mathbf{u}(x_{j-\frac{1}{2}}, t))dt - \int_{t_n}^{t_{n+1}} \mathbf{f}(\mathbf{u}(x_{j+\frac{1}{2}}, t))qdt\right] \tag{20}$$

If we define the approximate numerical flux as

$$F(\mathbf{U}_j^n, \mathbf{U}_{j+1}^n) = \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} \mathbf{u}(x_{j+\frac{1}{2}}, t)dt \tag{21}$$

We obtain

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n + \frac{\Delta t}{h_j}\left[F(\mathbf{U}_j^n, \mathbf{U}_{j+1}^n) - F(\mathbf{U}_{j-1}^n, \mathbf{U}_j^n)\right] \tag{22}$$

Of course, this represents a system of two equations which will evolve η and u respectively. Care must be taken when defining the numerical fluxes. At each cell edge, we can use a locally approximate constant $H(x)$ since we are solving the local Riemann problems and we choose a small enough time step that the solutions of such problems have not begun to interact. Namely, we have the time step estimate :

$$\Delta t \leq \frac{\text{CFL}\Delta x}{c_0}$$

2 Implementation

Using the given codes as 'inspiration' we attempt to implement the Godunov method using Firedrake. In particular, we consider the setup of a standing wave between two solid walls. The main consideration here is the implementation of the fluxes, both on the interior and at the solid walls. We will use extrapolating boundary conditions which can be seen in lines 105, 106, 115 and 116 of Thismaywork.py. For the velocity field, we ensure equal and opposite flux at the boundaries and for the surface elevation, we choose a very small value as setting this equal to 0 interferes with the weak form calculations. For the interior of the domain, we use a numerical flux derived in equations (16) and (17).

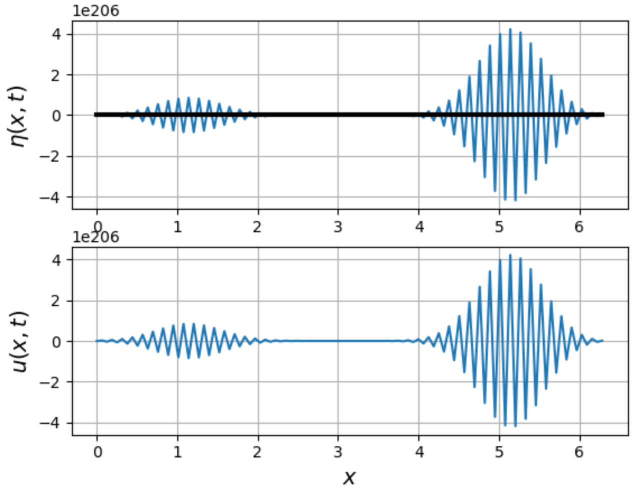

Figure 1: Sample output from Thismaywork.py. I was unable to get the black line to appear in the correct location, namely at $y = -1$ although the required code is in the program. The initial surface deviation and initial velocity can be found in the code.

In this code, we chose a very conservative estimate to ensure compliance with our time step bound that we obtained previously. With more care we may have been able to compare the effect of changing the CFL number. With question 5, I am not actually sure what the question is asking, it does not seem like there is anything to do here.

For question 6, we alter the flux boundary conditions for the Nbc = 3 case where we obtain a figure such as the following.

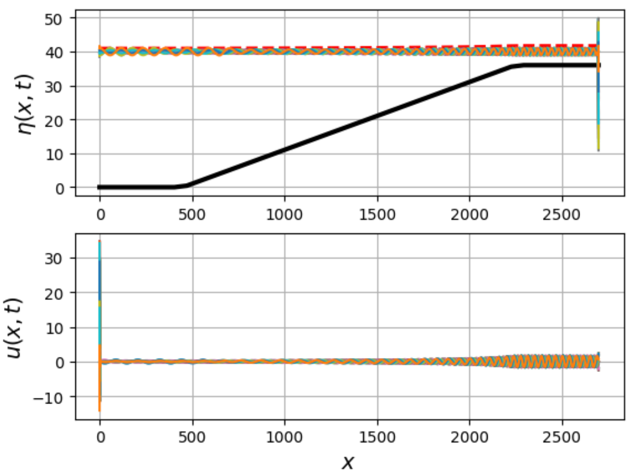

Figure 2: Using the code SolidWallsQ6.py we obtain this figure, clearly showing the presence of solid walls at either end of our computational domain.

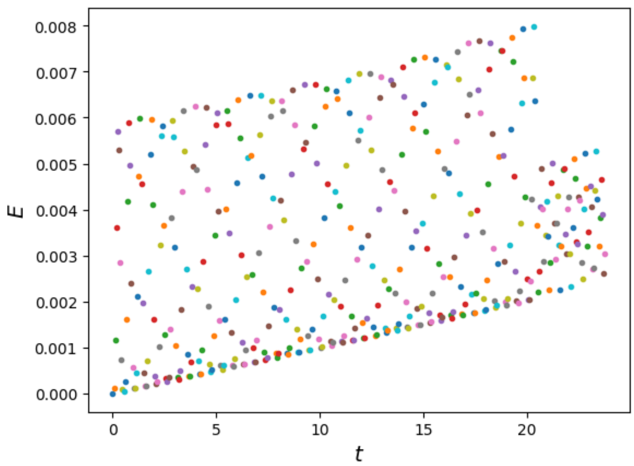

Figure 3: Energy evolution from using SolidwallsQ6.py where we have simulated solid walls and used the already coded alternating flux.

3 Final Question

Although the code I have written is not particularly convincing, we now use the given firedrake code to investigate the effects of changing the topography. With the conditions given in question 8, we have the following figure.

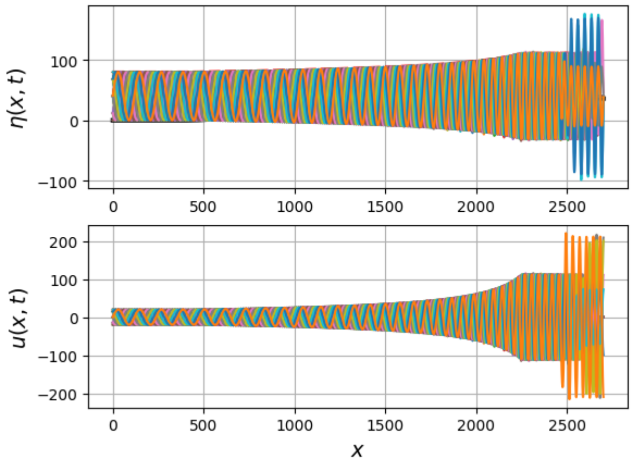

Figure 4: Simulation using the given parameters in Q8, using Q8.py

Although firedrake has kindly opted to not plot the topography, it can be easily observed that the wave amplitude is greater at a shallower depth, namely towards the right side of the domain. These figures show some nice behaviour but if we set $W = 5$ and $H_b = 0.5$, we suddenly have more 'interesting' behaviour.

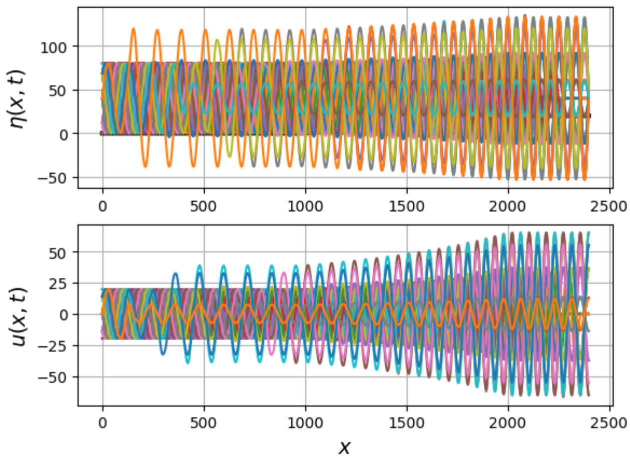

Figure 5: Effect of changing the topography using Q8.py.