PHY407F: Explanatory Notes for Lab 8

Idil Yaktubay and Souren Salehi

15 November 2022

1 Question 1 - by Souren Salehi and Idil Yaktubay

(a) In this question, we have calculated the electrostatic potential due to a simple 2D electronic capacitor system that has two flat metal plates of negligible thickness enclosed in a square metal box. The boundaries of the square metal box are at zero potential, whereas the two metal plates have potentials +1V and -1V, respectively. The metal box has a side length of 10 cm, and both plates, who are placed vertically and 6 cm apart, are 6 cm long. To calculate the electrostatic potential along this 2D box, we have created a grid with dimensions 100x100 that meets these conditions. Then, we have used the Gauss-Seidel method without over-relaxation to update the potential at every grid point until all of the calculated potentials reached the target accuracy of $10^{-6}\,V$. For us, the execution time of this method in this scenario was about 12.1 seconds. The pseudocode and the Python code for this calculation are included in the file named `LAB08_Q1.py`. Figure 1 depicts a contour plot of the system potential generated without over-relaxation.

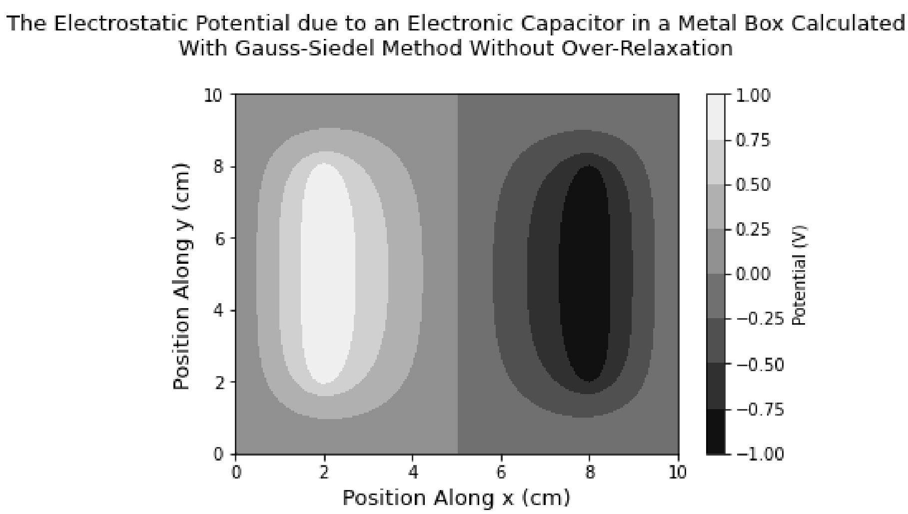

Figure 1: Contour plot of the electrostatic potential in a 2D system consisting of a two-plate capacitor enclosed in a zero potential metal box. The first plate has potential +1V and the second plate has potential -1V. The electrostatic potential was calculated using the Gauss-Seidel method without over-relaxation with a grid of 100x100 points and a target accuracy of $10^{-6}\,V$. For us, the execution time for this method was about 12.1seconds.

(b) We have repeated the above calculations for the same system, only this time using the Gauss-Seidel method with over-relaxation with over-relaxation constants of $\omega = 0.1$ and $\omega = 0.5$. Figures 2 and 3 depict the plots corresponding to $\omega = 0.1$ and $\omega = 0.5$, respectively. The execution times with $\omega = 0.1$ and $\omega = 0.5$ were about 11.8 seconds and 5.0 seconds, respectively. The former case shows slight improvement from Gauss-Seidel without over-relaxation, whereas the latter case shows excellent improvement in efficiency compared to the other two cases. Since we have not compromised accuracy for efficiency, we can conclude that using over-relaxation with $\omega = 0.5$ is the most ideal method out of the three. The pseudocode and the Python code for these plots are included in the file named `LAB08_Q1.py`.

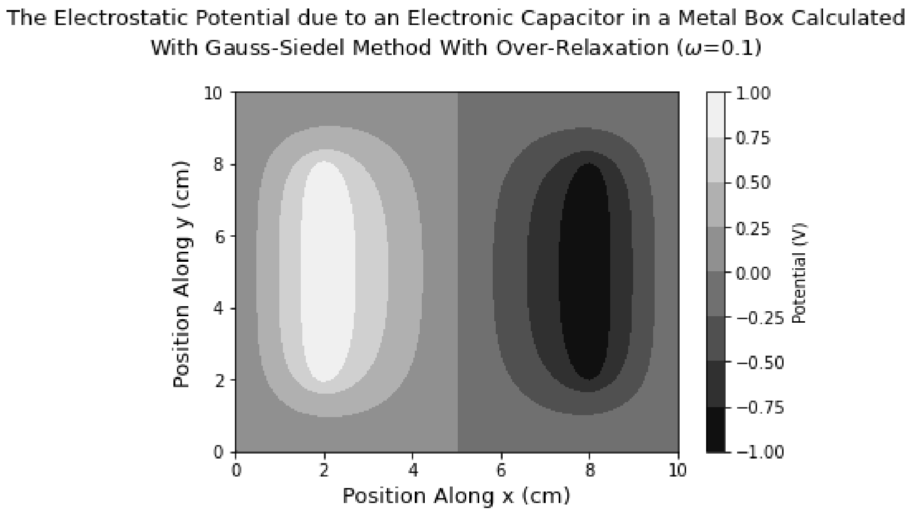

Figure 2: Contour plot of the electrostatic potential for the same system depicted in Figure 1. Here, the potential values were calculated using the Gauss-Seidel method with over-relaxation and an over-relaxation constant of $\omega = 0.1$, a grid of 100x100 points and a target accuracy of $10^{-6}\,V$. The execution time was about 11.8 seconds.

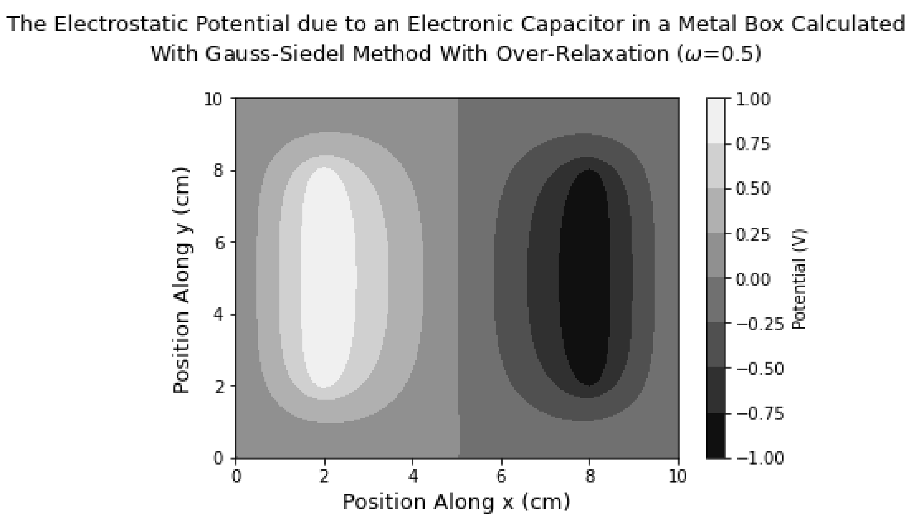

Figure 3: Contour plot of the electrostatic potential for the same system depicted in Figure 1. Here, the potential values were calculated using the Gauss-Seidel method with over-relaxation and an over-relaxation constant of $\omega = 0.5$, a grid of 100x100 points and a target accuracy of $10^{-6}\,V$. The execution time was about 5.0 seconds.

2 Question 2 - by Souren Salehi and Idil Yaktubay

(a) The 1D shallow-water equations are given by the following:

$$\begin{aligned} \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} &= -g\frac{\partial \eta}{\partial x} \\ \frac{\partial \eta}{\partial t} + \frac{\partial (uh)}{\partial x} &= 0 \end{aligned} \tag{1}$$

where u is the fluid velocity along the x direction, $h = \eta - \eta_b$ is the water column height, η is the altitude of the free water surface, η_b is the altitude of the bottom topography, and g is the gravitational acceleration. We can write these 1D shallow-water equations in the flux-conservative form given by

$$\frac{\partial \vec{u}}{\partial t} = -\frac{\partial \vec{F}(\vec{u})}{\partial x} \tag{2}$$

where $\vec{u} = (u, \eta)$ and $\vec{F}(u,\eta) = \left[\frac{1}{2}u^2 + g\eta,\ (\eta - \eta_b)u\right]$. To show this, we evaluate the right- and left-hand-sides of equation 2 to show that they are equal. First, the right-hand-side evaluates as

$$\begin{aligned} -\frac{\partial \vec{F}(\vec{u})}{\partial x} &= -\frac{\partial}{\partial x}\left[\frac{1}{2}u^2 + g\eta,\ (\eta - \eta_b)u\right] \\ &= -\left[\frac{1}{2}\frac{\partial (u^2)}{\partial x} + g\frac{\partial}{\partial x}\eta,\ \frac{\partial (uh)}{\partial x}\right] \\ &= -\left[u\frac{\partial u}{\partial x} - \frac{\partial u}{\partial t} - u\frac{\partial u}{\partial x},\ -\frac{\partial \eta}{\partial t}\right] \\ &= -\left[-\frac{\partial u}{\partial t},\ -\frac{\partial \eta}{\partial t}\right] \\ &= \left[\frac{\partial u}{\partial t},\ \frac{\partial \eta}{\partial t}\right] \end{aligned}$$

where we have used the chain rule and substituted equations 1 for $\frac{\partial (uh)}{\partial x}$ and $g\frac{\partial \eta}{\partial x}$. Further, the left-hand-side evaluates as

$$\frac{\partial \vec{u}}{\partial t} = \left[\frac{\partial}{\partial t}u,\ \frac{\partial}{\partial t}\eta\right].$$

which is identical to our result from the right-hand-side of equation 2. Therefore, equations 1 can be written in the flux-conservative form of equation 2. Now, we can use the FTCS scheme to discretize the 1D shallow-water equations as

$$\begin{aligned} u_j^{n+1} &= u_j^n - \frac{\Delta t}{2\Delta x}\left[\frac{1}{2}(u_{j+1}^n)^2 + g\eta_{j+1}^n - \frac{1}{2}(u_{j-1}^n)^2 - g\eta_{j-1}^n\right] \\ \eta_j^{n+1} &= \eta_j^n - \frac{\Delta t}{2\Delta x}\left[(\eta_{j+1}^n - \eta_{b_{j+1}})u_{j+1}^n - (\eta_{j-1}^n - \eta_{b_{j-1}})u_{j-1}^n\right] \end{aligned} \tag{3}$$

where superscripts n refer to the time step indices, superscripts j refer to the spatial step indices, and $\eta_b(x)$ is a known function.

(b) We have implemented the 1D shallow-water system with the FTCS scheme given by equation 3. In this specific case, we have used a domain width of $L = 1\,m$, a spatial step of $\Delta x = 0.02\,m$, a time step of $\Delta t = 0.01\,s$, an average water column height of $H = 0.01\,m$, and a bottom topography given by $\eta_b = 0\,m$. We have taken the gravitational acceleration to be $g = 9.81\,\frac{m}{s^2}$. We assumed the system to have rigid walls, which meant that our boundary conditions were given by $u(0,t) = u(L,t) = 0$. Further, the initial conditions of our systems were given by equation 4, where $A = 0.002\,m$, $\mu = 0.5\,m$, $\sigma = 0.05\,m$, and $\langle\rangle$ is the average operator to ensure that H remains as the free surface altitude at rest. Figure 4 depicts plots of the shallow-water function η with respect to position x at times $t = 0\,s$, $t = 1\,s$, and $t = 4\,s$. The pseudocode and the Python code that generates this graph is included in the file named `LAB08_Q2.py`.

$$u(x, 0) = 0, \;\; \eta(x, 0) = H + Ae^{-(x-\mu)^2/\sigma^2} - \langle Ae^{-(x-\mu)^2/\sigma^2} \rangle \tag{4}$$

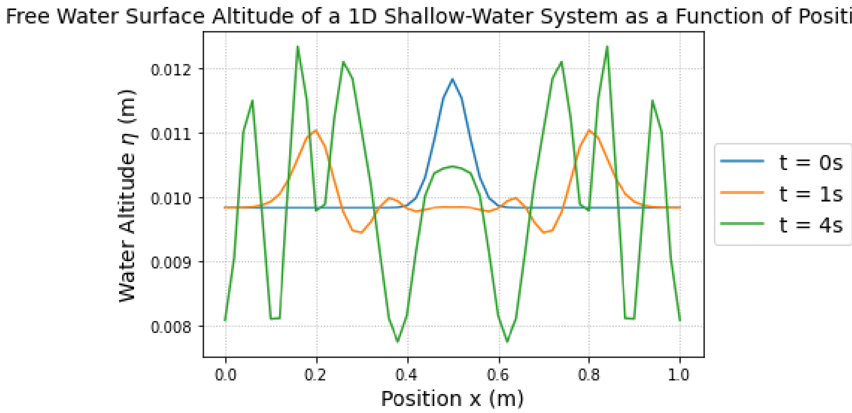

Figure 4: Altitude of the free water surface η with respect to position along x for times $t = 0s$, $1s$, $4s$ for a 1D shallow-water system. Boundary conditions and initial conditions of this system are given by $u(0, t) = u(L, t) = 0$ and equation 4, respectively. The equilibrium of the free water surface is at $H = 0.01m$, whereas the bottom topography is at $\eta_b = 0\,m$. The system is under gravitational force only. The plots were generated by applying the FTCS scheme to the flux-conservative form of equations 1 to find solutions for η at different times.

(c) Next, we have applied the Neumann stability analysis of the FTCS scheme applied to equations 1. Since the Neumann stability analysis only applies to linear equations, we can show that the linearized equations 1 about $(u, \eta) = (0, H)$ are with constant topography η_b. We evaluate equations 1 at $(u, \eta) = (0, H)$ as the following:

$$\begin{aligned}\frac{\partial \eta}{\partial t} &= -\frac{\partial}{\partial x}[u(\eta - \eta_b)] \\ &= -\frac{\partial}{\partial x}[u(H - 0)] = -H\frac{\partial u}{\partial x}\end{aligned}$$

and

$$\frac{\partial u}{\partial t} + 0 \cdot \frac{\partial u}{\partial x} = -g\frac{\partial \eta}{\partial x}.$$

Therefore, equations 1 are linear and we can now apply the Neumann analysis.

We may write,

$$\begin{aligned}u(x, t + \Delta t) &= u(x, t) - g\Delta t \cdot \frac{\eta(x + \Delta x, t) - \eta(x - \Delta x, t)}{2\Delta x} \\ \eta(x, t + \Delta t) &= \eta(x, t) - H\Delta t \cdot \frac{u(x + \Delta x, t) - u(x - \Delta x, t)}{2\Delta x}.\end{aligned} \tag{5}$$

We can then define u and η as ,

$$\begin{aligned}u(x, t) &= a_k(t)e^{ikx} \\ \eta(x, t) &= b_k(t)e^{ikx},\end{aligned} \tag{6}$$

so that equation 5 becomes

$$
\begin{aligned}
u(x,t+\Delta t) &= a_k(t)e^{ikx} - \frac{g\Delta t}{2\Delta x} b_k(t)e^{ikx}(e^{ik\Delta x} - e^{-ik\Delta x}) \\
\eta(x,t+\Delta t) &= b_k(t)e^{ikx} - \frac{H\Delta t}{2\Delta x} a_k(t)e^{ikx}(e^{ik\Delta x} - e^{-ik\Delta x}).
\end{aligned} \tag{7}
$$

We can define $\vec{u}$ as,

$$
\begin{aligned}
\vec{u}(x,t) = (u(x,t), \eta(x,t)) &= (a_k(t+\Delta t), b_k(t+\Delta t))e^{ikx} \\
&= \vec{c}(t+\Delta t)e^{ikx}.
\end{aligned} \tag{8}
$$

Therefore by equating equation 7 and 8 and eliminating the e^{ikx} we have,

$$
\vec{c}(t+\Delta t) = \begin{pmatrix} 1 & -\frac{g\Delta t}{\Delta x} isin(k\Delta x) \\ -\frac{H\Delta t}{\Delta x} isin(k\Delta x) & 1 \end{pmatrix} \vec{c}(t) \tag{9}
$$

The eigenvalue equation is thus

$$
(1-\lambda)^2 + \frac{gH\Delta t^2}{\Delta x^2} sin^2(k\Delta x) = 0, \tag{10}
$$

and thus we have,

$$
|\lambda| = \sqrt{1 + \frac{gH\Delta t^2}{\Delta x^2} sin^2(k\Delta x)}. \tag{11}
$$

We expect FTCS to be unstable as for any step in space or time we will have the magnitude of the eigenvalue be larger than unity.