Fluid Dynamics — Numerical Techniques

MATH5453M Numerical Exercises 1, 2024
Aly Ilyas
mmai@leeds.ac.uk

Due date: Friday October 18th

Sources: Chapter 2 of Morton and Mayers (2005, M&M), Internet.

Problem Statement

Consider the non-dimensional linear advection-diffusion equation for the variable/unknown $u = u(x, t)$, with an initial condition and boundary conditions:

$$u_t - a(t)u_x - \epsilon u_{xx} = 0, \quad x \in [L_p, L] \tag{1a}$$

$$u(x, 0) = u_0(x) \tag{1b}$$

$$u(L_p, t) = u(L, t) = 0, \tag{1c}$$

where ϵ is a small constant diffusion and $a(t)$ is a given function. The boundary conditions are classical homogeneous Dirichlet conditions. The above system arose from the research on machine learning of Choi et al. (2022). In the end, we will use $L_p = -1$ and $a(t) = 1$.

Tasks

Task 1:

Equation (1) is referred to as a linear advection-diffusion equation because it combines the effects of both advection (transport by a flow) and diffusion (spreading due to gradients), and its structure is linear in the unknown $u(x, t)$. Let's break down the three key terms:

The term **linear** means that the equation is linear in the unknown variable $u(x, t)$ and its derivatives. Specifically, the equation is of the form:

$$u_t - a(t)u_x - \epsilon u_{xx} = 0. \tag{1}$$

The unknown function u and its derivatives u_x, u_{xx}, and u_t all appear with constant coefficients (or coefficients that are independent of u), and there are no terms like u^2, $\sin(u)$, or any other nonlinear functions of u. In this case, the coefficient of u_x is $a(t)$, which is a function of time, but this does not depend on u itself, making the equation linear.

The term **advection** refers to the transport of a quantity (in this case, u) due to the movement of a medium. In the equation, this is represented by the term:

$$-a(t)u_x. \tag{2}$$

The derivative u_x describes how u changes with respect to space, and $a(t)$ represents the velocity at which u is transported in space. When $a(t) = 1$, it means the quantity u is advected or transported at a constant speed of 1 in the x-direction. The negative sign indicates that the dirrection of advection is opposite to the dirrection of incresing x (if $a(t)$ is positive, the flow is in the positive x dirrection. Thus the term (2) describes how the quantity u is carried along by the flow or shift of the solution over time.

The term **diffusion** refers to the spreading of a quantity due to gradients, which is typically modeled by a second-order spatial derivative. In the equation, this is represented by:

$$-\epsilon u_{xx}. \tag{3}$$

The second derivative u_{xx} describes how the gradient of u changes in space, and ϵ is a small constant representing the strength of the diffusion. This term causes u to "smooth out" or diffuse over time. The larger the value of ϵ, the faster the diffusion. In the case of small ϵ, diffusion is weak.

Task 2

Given the equation (1) at a specific time point (x_j, t_n), the time derivative u_t can be approximated using a central difference between t_n and t_{n+1} as:

$$\delta_t u_j^{n+\frac{1}{2}} = u_j^{n+1} - u_j^n. \tag{4}$$

Then, we expand both term u_j^{n+1} and u_j^n in Taylor series around mid point $(x_j, t_{n+\frac{1}{2}})$. We get:

$$u_j^{n+1} = \left[u + \frac{1}{2}\Delta t u_t + \frac{1}{2}\left(\frac{1}{2}\Delta t\right)^2 u_{tt} + \frac{1}{6}\left(\frac{1}{2}\Delta t\right)^3 u_{ttt} + \cdots \right]_j^{n+\frac{1}{2}}, \qquad (5)$$

and

$$u_j^{n} = \left[u - \frac{1}{2}\Delta t u_t + \frac{1}{2}\left(\frac{1}{2}\Delta t\right)^2 u_{tt} - \frac{1}{6}\left(\frac{1}{2}\Delta t\right)^3 u_{ttt} + \cdots \right]_j^{n+\frac{1}{2}}. \qquad (6)$$

Subtracting (5) to (6) and we obtain equation 2.80 in M&M:

$$\delta_t u_j^{n+\frac{1}{2}} = u_j^{n+1} - u_j^n = \left[\Delta t u_t + \frac{1}{24}(\Delta t)^3 u_{ttt} + \cdots \right]_j^{n+\frac{1}{2}}. \qquad (7)$$

Next, the spatial second derivative u_{xx} at point x_j based on equation 2.30 in M&M, for $n+1$:

$$\delta_x^2 u_j^{n+1} = u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}. \qquad (8)$$

For (8), we expand both term u_{j+1}^n and u_{j-1}^n. We get:

$$u_{j+1}^{n+1} = u_j^{n+1} + \Delta x u_x^{n+1} + \frac{(\Delta x)^2}{2} u_{xx}^{n+1} + \frac{(\Delta x)^3}{6} u_{xxx}^{n+1} + \frac{(\Delta x)^4}{24} u_{xxxx}^{n+1} + \cdots, \qquad (9)$$

and

$$u_{j-1}^{n+1} = u_j^{n+1} - \Delta x u_x^{n+1} + \frac{(\Delta x)^2}{2} u_{xx}^{n+1} - \frac{(\Delta x)^3}{6} u_{xxx}^{n+1} + \frac{(\Delta x)^4}{24} u_{xxxx}^{n+1} + \cdots. \qquad (10)$$

Subtituting (9) and (10) into (8) and we get the equation 2.81 from M&M as:

$$\delta_x^2 u_j^{n+1} = \left[(\Delta x)^2 u_{xx} + \frac{1}{12}(\Delta x)^4 u_{xxxx} + \frac{2}{6!}(\Delta x)^6 u_{xxxxxx} + \cdots \right]_j^{n+1}. \qquad (11)$$

Then we expand each term in (11) in powers of Δt, about the point $(x_j, t_{n+\frac{1}{2}})$. The Taylor series expansion for a function $u(x,t)$ is given by:

$$\begin{aligned} u(x,t) = u(&x_j, t_{n+\frac{1}{2}}) \\ &+ (x - x_j) u_x(x_j, t_{n+\frac{1}{2}}) \\ &+ (t - t_{n+\frac{1}{2}}) u_t(x_j, t_{n+\frac{1}{2}}) \\ &+ \frac{1}{2}(x - x_j)^2 u_{xx}(x_j, t_{n+\frac{1}{2}}) \\ &+ \frac{1}{2}(t - t_{n+\frac{1}{2}})^2 u_{tt}(x_j, t_{n+\frac{1}{2}}) + \cdots. \end{aligned} \qquad (12)$$

Now we apply (12) to the second derivative u_{xx} and higher derivatives, for u_{xx}:

$$u_{xx} = u_{xx}(x_j, t_{n+\frac{1}{2}}) + \Delta t\, u_{xxt}(x_j, t_{n+\frac{1}{2}}) + \frac{1}{2}(\Delta t)^2 u_{xxtt}(x_j, t_{n+\frac{1}{2}}) + \cdots\ , \tag{13}$$

for u_{xxxx}:

$$u_{xxxx} = u_{xxxx}(x_j, t_{n+\frac{1}{2}}) + \Delta t\, u_{xxxxt}(x_j, t_{n+\frac{1}{2}}) + \cdots\ , \tag{14}$$

for u_{xxxxxx}:

$$u_{xxxxxx} = u_{xxxxxx}(x_j, t_{n+\frac{1}{2}}) + \Delta t\, u_{xxxxxxt}(x_j, t_{n+\frac{1}{2}}) + \cdots\ . \tag{15}$$

Then we subtitute (13), (14), and (15) (PS. let's just drop $(x_j, t_{n+\frac{1}{2}})$ for simplicity) into (11) and get:

$$\begin{aligned}
\delta_x^2 u_j^{n+1} = &\left[(\Delta x)^2 u_{xx} + \frac{1}{12}(\Delta x)^4 u_{xxxx} + \frac{2}{6!}(\Delta x)^6 u_{xxxxxx} + \cdots\right] \\
&+ \left[\frac{1}{2}\Delta t(\Delta x)^2 u_{xxt} + \frac{1}{12}(\Delta x)^4 u_{xxxxt} + \cdots\right] \\
&+ \left[\frac{1}{2}(\frac{1}{2}\Delta t)^2(\Delta x)^2 u_{xxtt} + \cdots\right] + \cdots\ .
\end{aligned} \tag{16}$$

There is similar expansion for $\delta_x^2 u_j^n$ and combining it with (16), we get the equation 2.82 in M&M:

$$\begin{aligned}
\theta\delta_x^2 u_j^{n+1} + (1-\theta)\delta_x^2 u_j^n = &\left[(\Delta x)^2 u_{xx} + \frac{1}{12}(\Delta x)^4 u_{xxxx} + \frac{2}{6!}(\Delta x)^6 u_{xxxxxx} + \cdots\right] \\
&+ \left[\left(\theta - \frac{1}{2}\right)\Delta t(\Delta x)^2 u_{xxt} + \frac{1}{12}(\Delta x)^4 u_{xxxxt} + \cdots\right] \\
&+ \frac{1}{8}(\Delta t)^2(\Delta x)^2 u_{xxtt} + \cdots\ .
\end{aligned} \tag{17}$$

We already got (7) and (17), and the form of truncation error is given by equation 2.83 in the M&M as:

$$T_j^{n+\frac{1}{2}} := \frac{\delta_t u_j^{n+\frac{1}{2}}}{\Delta t} - \frac{\theta\delta_x^2 u_j^{n+1} + (1-\theta)\delta_x^2 u_j^n}{(\Delta x)^2}. \tag{18}$$

Finally, we get the equation 2.84 in M&M as:

$$\begin{aligned} T_j^{n+\frac{1}{2}} := [u_t - u_{xx}] &+ \left[\left(\frac{1}{2}-\theta\right)\Delta t\, u_{xxt} - \frac{1}{12}(\Delta x)^2 u_{xxxx}\right] \\ &+ \left[\frac{1}{24}(\Delta t)^2 u_{ttt} - \frac{1}{8}(\Delta t)^2 u_{xxtt}\right] \\ &+ \left[\frac{1}{12}\left(\frac{1}{2}-\theta\right)\Delta t(\Delta x)^2 u_{xxxxt} - \frac{2}{6!}(\Delta x)^4 u_{xxxxxx}\right]. \end{aligned} \tag{19}$$

Task 3

Given the non-dimensional linear advection-diffusion equation (1a) with initial and boundary conditions (1b) and (1c), we will let the domain $[L_p, L]$ be divided into J points.

$$\Delta x = \frac{L - L_p}{J - 1}. \tag{20}$$

The spatial grid points are denoted as x_j for $j = 0, 1, 2, \ldots, J-1$, where:

$$x_j = L_p + j\Delta x. \tag{21}$$

Next, we discretize the time domain into M points, with time step size Δt, such that the time grid points are t_n for $n = 0, 1, 2, \ldots, M-1$, where:

$$t_n = n\Delta t. \tag{22}$$

Then, the time derivative u_t is discretized using the θ-method:

$$u_t \approx \frac{u_j^{n+1} - u_j^n}{\Delta t}. \tag{23}$$

The θ-method approximates the equation at time level $n+1$ as:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \theta\left(-a(t)u_x^{n+1} - \epsilon u_{xx}^{n+1}\right) + (1-\theta)\left(-a(t)u_x^n - \epsilon u_{xx}^n\right). \tag{24}$$

Here, $\theta \in [0,1]$ controls the scheme: $\theta = 0$ gives a fully explicit scheme, $\theta = 1$ gives a fully implicit scheme, and $\theta = \frac{1}{2}$ gives the Crank-Nicolson scheme (midpoint).

The advection term $a(t)u_x$ is discretized using the first-order upwind scheme. The direction of the upwinding is determined by the sign of $a(t)$. Since we assume $a(t) = 1$ (positive), we use the leftward upwind stencil, thus, the advection term becomes:

for $n+1$:

$$u_x^{n+1} \approx \frac{u_j^{n+1} - u_{j-1}^{n+1}}{\Delta x}, \tag{25}$$

and for n:

$$u_x^n \approx \frac{u_j^n - u_{j-1}^n}{\Delta x}. \tag{26}$$

The diffusion term ϵu_{xx} is discretized using the central difference scheme: for $n+1$:

$$u_{xx}^{n+1} \approx \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{\Delta x^2}, \tag{27}$$

and for n:

$$u_{xx}^n \approx \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2}. \tag{28}$$

The boundary conditions are:

$$u(L_p, t) = u(L, t) = 0. \tag{29}$$

At grid points corresponding to $j = 0$ and $j = N-1$, we enforce:

$$u_0^{n+1} = u_0^n = 0, \quad u_{N-1}^{n+1} = u_{N-1}^n = 0. \tag{30}$$

For near the boundary scenario, the internal points $j = 1, 2, \ldots, N-2$, the discretization becomes:

$$\begin{aligned} \frac{u_j^{n+1} - u_j^n}{\Delta t} &= \theta \left(-a(t) \frac{u_j^{n+1} - u_{j-1}^{n+1}}{\Delta x} - \epsilon \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{\Delta x^2} \right) \\ &+ (1-\theta) \left(-a(t) \frac{u_j^n - u_{j-1}^n}{\Delta x} - \epsilon \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2} \right). \end{aligned} \tag{31}$$

The we rearrange terms to isolate u_j^{n+1} (unknown) on the left-hand side (LHS) and u_j^n (known) on the right-hand side (RHS):

$$\begin{aligned} u_j^{n+1} \left(1 + \theta \frac{a(t)}{\Delta x} + 2\theta \frac{\epsilon}{\Delta x^2} \right) - u_{j-1}^{n+1} \left(\theta \frac{a(t)}{\Delta x} + \theta \frac{\epsilon}{\Delta x^2} \right) - u_{j+1}^{n+1} \theta \frac{\epsilon}{\Delta x^2} \\ = u_j^n \left(1 - (1-\theta) \frac{a(t)}{\Delta x} - 2(1-\theta) \frac{\epsilon}{\Delta x^2} \right) \\ + u_{j-1}^n \left((1-\theta) \frac{a(t)}{\Delta x} + (1-\theta) \frac{\epsilon}{\Delta x^2} \right) + u_{j+1}^n (1-\theta) \frac{\epsilon}{\Delta x^2}. \end{aligned} \tag{32}$$

The system of (32) can also be written in matrix form:

$$Au^{n+1} = Bu^n, \tag{33}$$

where A and B are matrices containing the coefficients from the discretization of u_j^{n+1} and u_j^n, respectively.

Task 4

In order to reproduce Fig 2.2 in M&M, we need to solve a one-dimensional partial differential equation (PDE) of the form:

$$u_t = u_{xx}, \tag{34}$$

where $u(x,t)$ represents a quantity (e.g., temperature or concentration) at position x and time t. The boundary conditions are homogeneous Dirichlet conditions:

$$u(0,t) = u(1,t) = 0, \tag{35}$$

and the initial condition is given by a "hat" function:

$$u_0(x) = \begin{cases} 2x, & 0 \leq x \leq 0.5, \\ 2(1-x), & 0.5 \leq x \leq 1. \end{cases} \tag{36}$$

The explicit finite difference scheme for this PDE is written as:

$$u_j^{n+1} = u_j^n + \mu \left(u_{j+1}^n - 2u_j^n + u_{j-1}^n\right), \tag{37}$$

where u_j^n is the value of $u(x_j, t_n)$ at the j-th spatial point and the n-th time step, and $\mu = \frac{\Delta t}{\Delta x^2}$. The boundary conditions ensure that $u_0^n = u_J^n = 0$ for all n.

Using **number4.py** we simulate both case. For the stable case, the time step $\Delta t = 0.0012$ results in $\mu = 0.48$, which satisfies the stability condition:

$$\mu = \frac{\Delta t}{\Delta x^2} \leq 0.5. \tag{38}$$

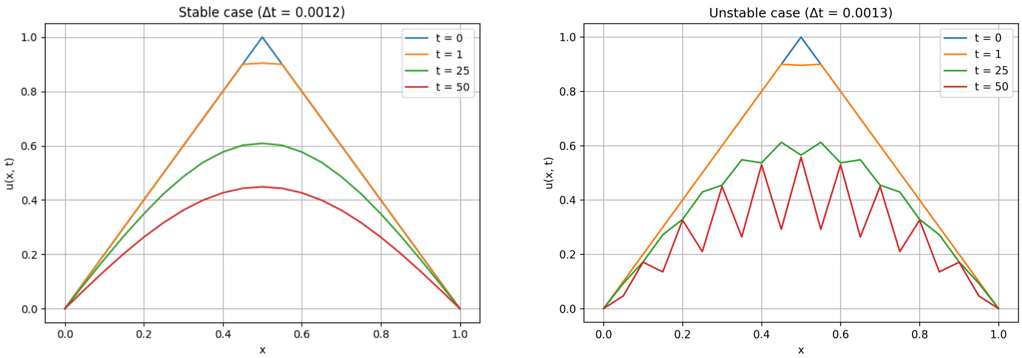
The solution remains stable without oscillations or unbounded growth, as shown in the Figure 1a. The solution evolves smoothly over time, reflecting a well-behaved diffusion process.

In the unstable case, the time step $\Delta t = 0.0013$ leads to $\mu = 0.52$, which exceeds the stability threshold of 0.5. As a result, the solution becomes unstable, and oscillations grow over time, as shown in the Figure 1b.

The amplification factor for the Fourier mode with wavenumber k in the explicit scheme is:

$$\lambda(k) = 1 - 4\mu \sin^2\left(\frac{k\Delta x}{2}\right). \tag{39}$$

Stability requires that $|\lambda(k)| \leq 1$ for all modes k. If $\mu > 0.5$, some Fourier modes grow exponentially, leading to instability. For $\mu = 0.52$, the amplification factor exceeds 1 for certain wavenumbers, causing the unbounded oscillations observed in the unstable case.



(a) (b)

Figure 1: Stability and unstability comparison (reproduce from Fig 2.2 in M&M using **number4.py**) (a) stable case where $\mu = 0.48$ using the time step $\Delta t = 0.0012$ (b) unstable case where $\mu = 0.52$ using the time step $\Delta t = 0.0013$.

The explicit scheme satisfies the maximum principle when $\mu \leq 0.5$. This ensures that the solution $u(x,t)$ remains bounded within the initial and boundary conditions. When $\mu > 0.5$, this principle is violated, leading to the unbounded growth of oscillations, as observed in the unstable case.

Next, we extend the explicit scheme to the previous statement problem (1) using **number4b.py**. In this case, we are considering the advection-diffusion equation with the following parameters: $L = 1.0$ (right boundary), $L_p = -1.0$ (left boundary), $J = 40$ (number of spatial points), $\Delta x = \frac{L-L_p}{J} = 0.05$ (spatial step size), $\epsilon = 1 \times 10^{-3}$ (diffusion constant), and $a = 1.0$ (advection speed). The initial condition is given by function:

$$u(x,0) = (1-x)^4(1+x) \tag{40}$$

The explicit scheme is subject to stability conditions for both advection and diffusion. The stability condition for diffusion is determined by:

$$\epsilon \frac{\Delta t}{\Delta x^2} < \frac{1}{2}. \tag{41}$$

Solving for Δt, we obtain:

$$\Delta t_{\text{diffusion}} < \frac{1}{2} \times \frac{\Delta x^2}{\epsilon} = 1.25. \tag{42}$$

Similarly, the stability condition for advection is given by:

$$\frac{a\Delta t}{\Delta x} < 1. \tag{43}$$

Solving for Δt, we obtain:

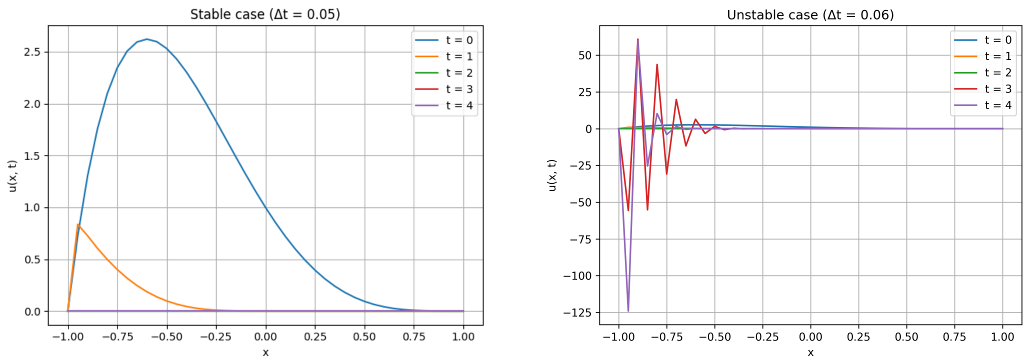
$$\Delta t_{\text{advection}} < \frac{0.05}{1.0} = 0.05. \tag{44}$$

Since both diffusion and advection must be stable, the critical time step $\Delta t_{\text{critical}}$ is determined as the smaller of the two values of (41) and (43):

$$\Delta t_{\text{critical}} = 0.05. \tag{45}$$

Thus, for stability, we use $\Delta t_1 = 0.05$ for the stable case. To illustrate an unstable case, we select $\Delta t_2 = 0.06$, which exceeds the critical value.

Figure 2 demonstrates the behavior of the system using these two time steps. Figure 2a shows the stable case with $\Delta t_1 = 0.05$, where the solution becomes stable by time step 2. Meanwhile, Figure 2b illustrates the unstable case with $\Delta t_2 = 0.06$, where oscillations begin to appear at time step 3.



(a) (b)

Figure 2: Stability and unstability comparison from problem statement (1) using **number4b.py** (a) stable case where $\Delta t = 0.05$ (b) unstable case where $\Delta t = 0.06$.

Task 5

We have already presented the non-dimensional linear advection-diffusion equation (1a) along with the initial and boundary conditions (1b) and (1c). Additionally, we have the θ-scheme to discretize the time derivative from

(31). From equations (25), (26), (27), and (28), the discretization for the advection term at time step $n+1$ is given by:

$$a(t)\frac{u_j^{n+1} - u_{j-1}^{n+1}}{\Delta x}. \tag{46}$$

Similarly, at time step n, it is:

$$a(t)\frac{u_j^{n} - u_{j-1}^{n}}{\Delta x}. \tag{47}$$

Next, we can derive the diffusion term. At time step $n+1$, the diffusion term becomes:

$$\epsilon\frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{\Delta x^2}. \tag{48}$$

Similarly, at time step n, it is:

$$\epsilon\frac{u_{j+1}^{n} - 2u_j^{n} + u_{j-1}^{n}}{\Delta x^2}. \tag{49}$$

We already got the equation in matrix form in (33) and initial condition in (40), for the matrix A, we use the following relationships for the coefficients of u_j^{n+1}:

$$A[j, j-1] = \theta\left(-a(t)\frac{1}{\Delta x} - \epsilon\frac{1}{\Delta x^2}\right), \tag{50}$$

$$A[j, j] = 1 + 2\theta\epsilon\frac{1}{\Delta x^2}, \tag{51}$$

$$A[j, j+1] = \theta\epsilon\frac{1}{\Delta x^2}. \tag{52}$$

For the matrix B, we use the following relationships for the coefficients of u_j^n:

$$B[j, j-1] = (1-\theta)\left(a(t)\frac{1}{\Delta x} + \epsilon\frac{1}{\Delta x^2}\right), \tag{53}$$

$$B[j, j] = 1 - 2(1-\theta)\epsilon\frac{1}{\Delta x^2}, \tag{54}$$

$$B[j, j+1] = (1-\theta)\epsilon\frac{1}{\Delta x^2}. \tag{55}$$

The boundary conditions are applied explicitly, ensuring $u_0 = u_{N-1} = 0$ for Dirichlet boundary conditions.

The stability of the θ-scheme is explored using both Fourier analysis and the maximum principle Given the parameters: $L = 1.0$, $L_p = -1.0$, $J = 100$,

$\Delta x = \frac{L-L_p}{J} = \frac{2.0}{100} = 0.02$, $\Delta t = 0.02$, $a = 1.0$, $\epsilon = 0.001$. To assess the stability of the scheme, we perform a Fourier analysis on the advection and diffusion parts separately. For the advection term, stability is governed by the Courant-Friedrichs-Lewy (CFL) condition:

$$\frac{a(t)\Delta t}{\Delta x} \leq 1. \tag{56}$$

For the diffusion term, the stability condition is:

$$\Delta t \leq \frac{\Delta x^2}{2\epsilon}. \tag{57}$$

Substituting the values into the CFL condition:

$$\frac{a(t)\Delta t}{\Delta x} = \frac{1.0 \times 0.02}{0.02} = 1.0 \leq 1. \tag{58}$$

This confirms that the scheme is stable regarding the advection term, as the condition is satisfied.

Now we analyze the stability condition for the diffusion term:

$$\Delta t \leq \frac{\Delta x^2}{2\epsilon}. \tag{59}$$

Calculating the right-hand side:

$$\frac{\Delta x^2}{2\epsilon} = \frac{(0.02)^2}{2 \times 0.001} = \frac{0.0004}{0.002} = 0.2. \tag{60}$$

Since $\Delta t = 0.02 \leq 0.2$, the diffusion stability condition is also satisfied.

The maximum principle guarantees that the numerical solution remains bounded under certain conditions. For the advection term, the maximum principle requires:

$$\frac{a(t)\Delta t}{\Delta x} \leq 1. \tag{61}$$

Since this condition holds true as discussed above, the numerical solution will not exhibit unbounded growth due to the advection component. For the diffusion term, the maximum principle is satisfied if:

$$\frac{\epsilon\Delta t}{\Delta x^2} \leq \frac{1}{2}. \tag{62}$$

Calculating the left-hand side:

$$\frac{\epsilon\Delta t}{\Delta x^2} = \frac{0.001 \times 0.02}{(0.02)^2} = \frac{0.00002}{0.0004} = 0.05 \leq 0.5. \tag{63}$$

This condition also holds true, confirming that the numerical solution remains bounded with respect to the diffusion component.

Both the Fourier analysis and the maximum principle indicate that the θ-scheme remains stable under the given conditions for the selected parameters. These results affirm that for appropriate choices of Δt, Δx, and the advection speed a, the numerical solution will accurately reflect the underlying physical behavior of the advection-diffusion problem without instabilities or violations of the maximum principle. We implemented the θ-scheme in **number5.py** using linear algebra routines to solve the matrix system and the result is shown in Figure 3.

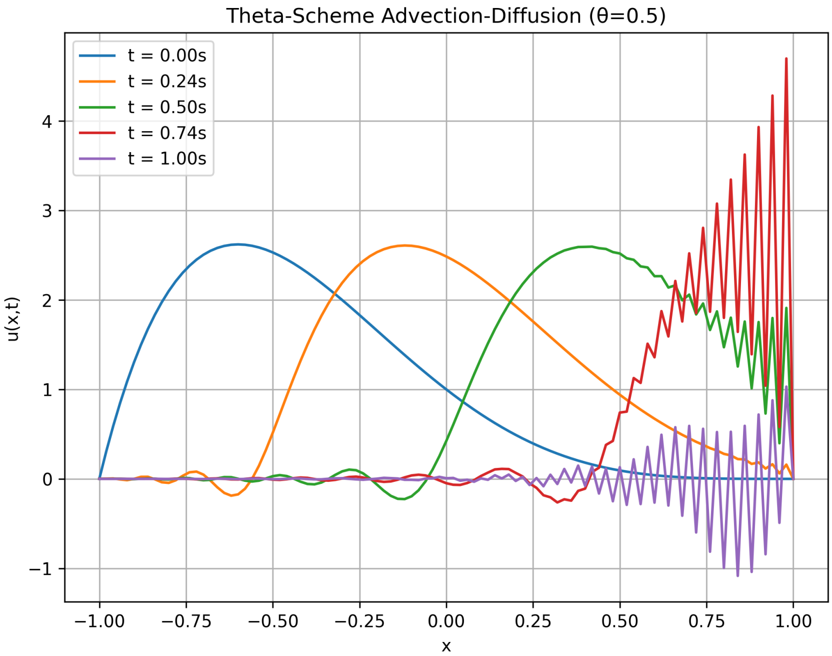


Figure 3: Numerical solution of the non-dimensional linear advection-diffusion equation obtained using the θ-scheme in **number5.py**. The solution is computed under the specified initial and boundary conditions, as outlined in equations (1b) and (1c).

Task 6

Now, we investigate the case where $a(t) = 1$ and the initial condition is given by

$$u(x,0) = (1-x)^4(1+x)\left(\sum_{k=0}^{3} b_k\phi_k(x) + C\right), \tag{64}$$

with the boundaries set at $L_p = -1$ and $L = 1$. The parameters for the simulation are chosen as $\epsilon = 10^{-3}$, $T = 1$ and $t \in [0, T]$. The Legendre polynomials used are defined as:

$$\begin{aligned}
\phi_0(x) &= 1,\\
\phi_1(x) &= x,\\
\phi_2(x) &= \frac{3}{2}x^2 - \frac{1}{2},\\
\phi_3(x) &= \frac{5}{2}x^3 - \frac{3}{2}x.
\end{aligned}$$

The coefficients b_k are randomly selected from a uniform distribution in the interval $[0, 1)$ for $k = 0, 1, 2, 3$. A constant $C \geq 0$ is determined numerically such that

$$\sum_{k=0}^{3} b_k\phi_k(x) + C \geq 0. \tag{65}$$

For the numerical implementation, we utilize a while-loop with discrete time steps rather than fixed iterations. This approach ensures that the computed time profiles reflect the time-dependent behavior of the solution.

To validate our results, we explore various values of θ and μ. The stability and potential violation of the maximum principle are examined through a μ–θ parameter plot. Specifically, we select three combinations of μ and θ corresponding to the cases of $\theta = 0$, $\frac{1}{2}$, and 1.

The function implementing the explicit scheme calculates the solution iteratively and captures the results at set time intervals. The choice of θ significantly impacts stability; for instance, when $\theta = 1$, the CFL condition is satisfied, ensuring stability.

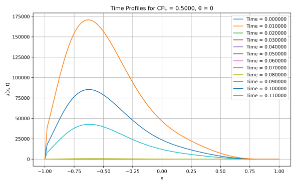
Figure 4 shows the results for different θ combinations concerning different stability criteria based on the maximum principle. The maximum principle guarantees that the numerical solution remains bounded under certain conditions. For the advection term, the maximum principle requires:

$$a\Delta t \leq \Delta x \quad \text{(advection stability condition)} \tag{66}$$

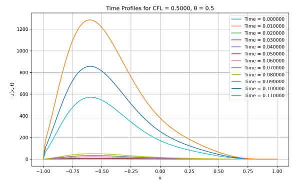
For the diffusion term, the maximum principle is satisfied if:

$$\Delta t \leq \frac{\Delta x^2}{2\epsilon} \quad \text{(diffusion stability condition).} \tag{67}$$

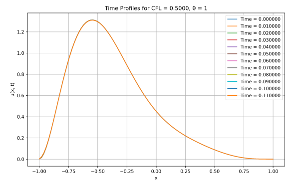
If these conditions are violated, the explicit scheme becomes unstable, and the solution may diverge or exhibit oscillations.



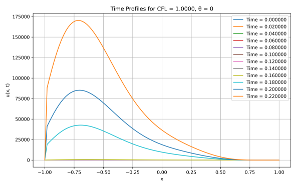
(a)
$CFL = 0.5$ and
$\theta = 0$



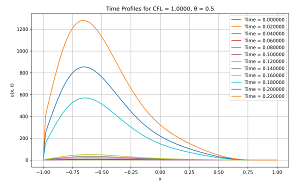
(b)
$CFL = 0.5$ and
$\theta = 0.5$



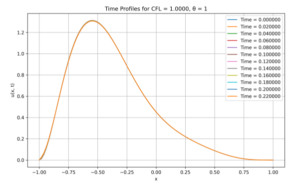
(c)
$CFL = 0.5$ and
$\theta = 1$



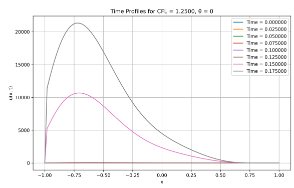
(d)
$CFL = 1$ and
$\theta = 0$



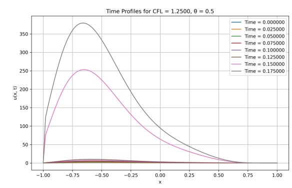
(e)
$CFL = 1$ and
$\theta = 0.5$



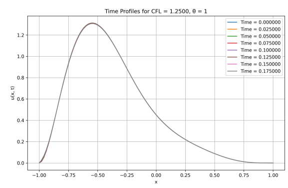
(f)
$CFL = 1$ and
$\theta = 1$



(g)
$CFL = 1.25$ and
$\theta = 0$



(h)
$CFL = 1.25$ and
$\theta = 0.5$



(i)
$CFL = 1.25$ and
$\theta = 1$

Figure 4: the results for different θ combinations concerning different stability criteria based on the maximum principle in **number6.py**.

In the code of **number6.py**, we use $\Delta t = 0.01$, $\Delta t = 0.02$, and $\Delta t = 0.025$. While all of these satisfy the diffusion stability condition, $\Delta t = 0.01$ satisfies the advection stability condition with a CFL of 0.5. From the figure, it can be seen that it stably converges to zero for both the explicit scheme and the Crank-Nicolson scheme. However, for $\theta = 1$, it remains stable but does not converge to zero.

For $\Delta t = 0.02$ (CFL $= 1$), $\theta = 0$ is stable and converges to zero, while $\theta = 0.5$ is stable and converging to zero slower than $\theta = 0$. For $\theta = 1$, it is stable but does not converge. When using $\Delta t = 0.025$, it does not satisfy the advection stability condition, as CFL = 1.25. Instead of converging to zero stably, it oscillates to a high amplitude for both $\theta = 0$ and $\theta = 0.5$, while for $\theta = 1$, it is stable but again does not converge.

In our simulations, the random coefficients were reported as:

$$b_k = [0.37454012, 0.95071431, 0.73199394, 0.59865848].$$

Through this exploration, we illustrate how various parameter choices influence the stability and accuracy of the numerical solution, providing insights into the behavior of the modeled physical phenomena. Choosing $\theta = 1$ in the $\mu - \theta$ scheme, where CFL = 1, is necessary for ensuring the stability of time-dependent solutions in the advection-diffusion equation, particularly when balancing computational efficiency with accuracy. The fully implicit method associated with $\theta = 1$ is unconditionally stable, allowing for larger time steps (Δt) without causing instability, such as oscillations or divergence. Thus, setting $\theta = 1$ with CFL = 1 is not only practical but essential for achieving stable, accurate results in advection-diffusion simulations.

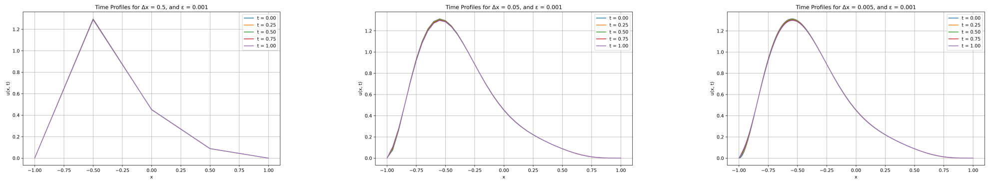
Task 7

We consider the case where the advection term $a(t) = 1$ and the initial condition is defined in (64), where $L_p = -1$, $L = 1$, $\epsilon = 10^{-3}$, and $T = 1$ with $t \in [0, T]$.

In Figure 5, the simulation resultsfrom **number7.py** for three different spatial resolutions, $\Delta x = 0.5$, $\Delta x = 0.05$, and $\Delta x = 0.005$, are shown. For $\Delta x = 0.5$, while the solution remains stable, the results appear pointy and deviate from the expected form, likely due to the relatively large spatial step size. As the grid is refined to $\Delta x = 0.05$, the results not only remain stable but also exhibit a good approximation to the original function, demonstrating that the solution has reached convergence. With further refinement to $\Delta x = 0.005$, the solution remains stable and becomes even smoother, although convergence is achieved more slowly. Figure 6 provides a detailed view by zooming in on the range $x = -1$ to $x = -0.9$, highlighting the smoother behavior of the solution at finer resolutions.

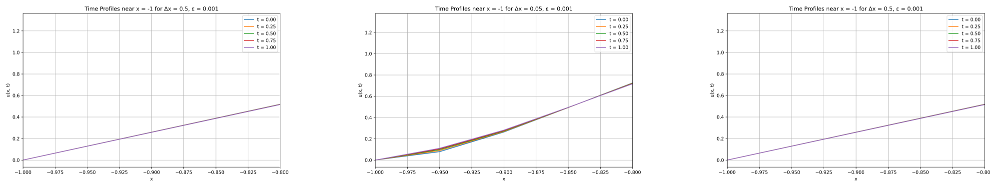
Task 8

Lastly, we explore the impact of smaller values of the diffusion coefficient ϵ in **number8.py**, specifically $\epsilon = 10^{-4}, 10^{-5}, 10^{-6}$, on the convergence behavior of the solution. As shown in Figure 7, the results reveal that as ϵ decreases,



(a) $\Delta x = 0.5$ (b) $\Delta x = 0.05$ (c) $\Delta x = 0.005$

Figure 5: Simulation results for different spatial resolutions: $\Delta x = 0.5$, $\Delta x = 0.05$, and $\Delta x = 0.005$. (a) $\Delta x = 0.5$, the result is stable but pointy, likely due to the large Δx. (b) $\Delta x = 0.05$, the solution reaches convergence and exhibits a smoother form. (c) $\Delta x = 0.005$, the solution is even smoother but convergence takes longer.



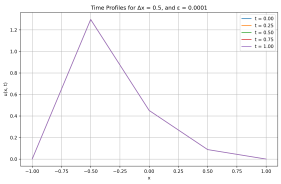
(a) $\Delta x = 0.5$ (b) $\Delta x = 0.05$ (c) $\Delta x = 0.005$

Figure 6: Zoomed-in view of Figure 5, focusing on the range $x = -1$ to $x = -0.9$. The refinement in Δx leads to progressively smoother solutions, illustrating the convergence behavior more clearly at finer spatial resolutions.

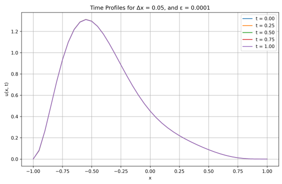
the solution exhibits increasing convergence, approaching a state that resembles a purely advection-driven system. In these cases, the diffusion term has a diminishing influence, which leads to the preservation of sharper gradients in the solution over time, as the smoothing effect of diffusion is reduced.

The zoomed-in view in Figure 8, focusing on the interval $x = -1$ to $x = -0.9$, provides a clearer depiction of how the solution evolves at finer spatial resolutions. The smaller the value of ϵ, the sharper the solution becomes, with less diffusion smoothing out the initial conditions. As ϵ approaches 10^{-6}, the solution behaves almost as though there is no diffusion, allowing distinct features to persist throughout the simulation.

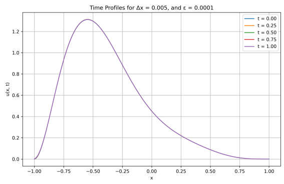
These findings highlight the influence of the diffusion coefficient on the system's behavior, with smaller ϵ values reducing the role of diffusion. This is significant for understanding systems where diffusion is minimal or nearly absent, as it shows how the absence of diffusion allows for sharper transitions and more pronounced features to be retained in the solution.



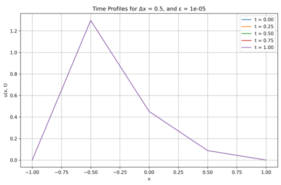
(a)
$\Delta x = 0.5$ and
$\epsilon = 10^{-4}$



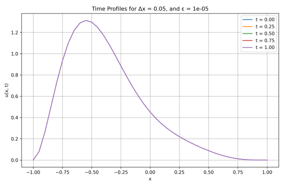
(b)
$\Delta x = 0.05$ and
$\epsilon = 10^{-4}$



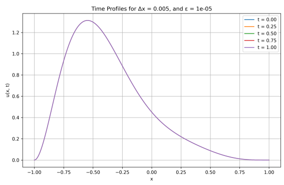
(c)
$\Delta x = 0.005$ and
$\epsilon = 10^{-4}$



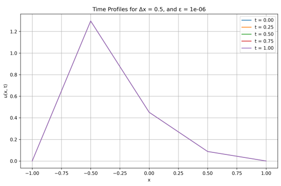
(d)
$\Delta x = 0.5$ and
$\epsilon = 10^{-5}$



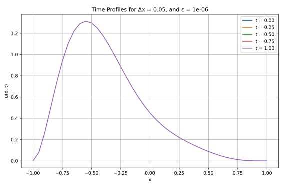
(e)
$\Delta x = 0.05$ and
$\epsilon = 10^{-5}$



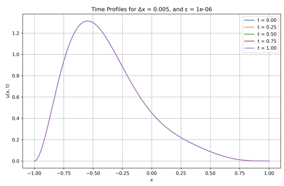
(f)
$\Delta x = 0.005$ and
$\epsilon = 10^{-5}$



(g)
$\Delta x = 0.5$ and
$\epsilon = 10^{-6}$

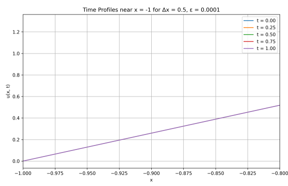


(h)
$\Delta x = 0.05$ and
$\epsilon = 10^{-6}$

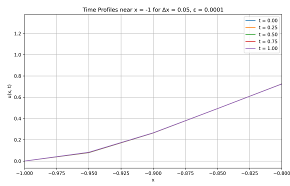


(i)
$\Delta x = 0.005$ and
$\epsilon = 10^{-6}$

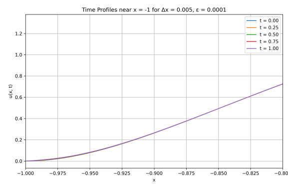
Figure 7: the result of smaller values of the diffusion coefficient ϵ in **number8.py**.



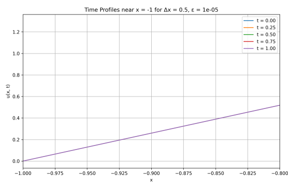
(a)
$\Delta x = 0.5$ and
$\epsilon = 10^{-4}$



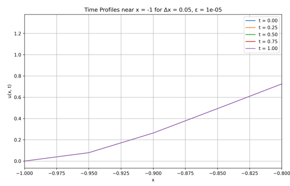
(b)
$\Delta x = 0.05$ and
$\epsilon = 10^{-4}$



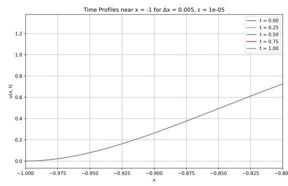
(c)
$\Delta x = 0.005$ and
$\epsilon = 10^{-4}$



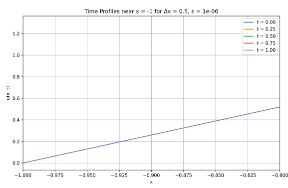
(d)
$\Delta x = 0.5$ and
$\epsilon = 10^{-5}$



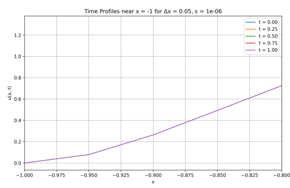
(e)
$\Delta x = 0.05$ and
$\epsilon = 10^{-5}$



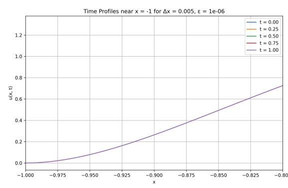
(f)
$\Delta x = 0.005$ and
$\epsilon = 10^{-5}$



(g)
$\Delta x = 0.5$ and
$\epsilon = 10^{-6}$



(h)
$\Delta x = 0.05$ and
$\epsilon = 10^{-6}$



(i)
$\Delta x = 0.005$ and
$\epsilon = 10^{-6}$

Figure 8: Zoomed-in view of Figure 7, focusing on the range $x = -1$ to $x = -0.9$ in **number8.py**.