Numerical Methods in ESP

Numerical Methods II

Graziano Giuliani

International Centre for Theoretical Physics

Second Semester 2023-24

/afs/ictp.it/public/g/ggiulian/WORLD/num2_lesson3.pdf

The linear advection equation

Consider the partial linear differential equation:

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} = 0 \tag{1}$$

where u is a constant velocity in the domain $0 \leq x \leq 1$ with periodic boundary conditions $\phi(0, t) = \phi(1, t)$ and initial condition given as $\phi(x, 0) = F(x)$, with F being a given function.
The analytical solution of the problem is:

$$\phi(x, t) = F(x - ut) \tag{2}$$

We call this equation the Linear Advection Equation as it describes the shape preserving movement of the initial profile $F(x)$ advected with speed u over time t.

The leapfog scheme: CTCS

- approximate the spatial and temporal derivative with centered difference formula. i.e:

$$\frac{\phi_j^{n+1} - \phi_j^{n-1}}{2\Delta t} = -u\frac{\phi_{j+1}^{n} - \phi_{j-1}^{n}}{2\Delta x} \tag{3}$$

- This is a three time level formula, because it contains the values ϕ_j at the three times t^{n-1}, t^n, t^{n+1}.
- Rearranging for the unknown we have:

$$\begin{aligned} \phi_j^{n+1} &= \phi_j^{n-1} - c(\phi_{j+1}^{n} - \phi_{j-1}^{n}) \\ c &= u\frac{\Delta t}{\Delta x} \end{aligned} \tag{4}$$

Stability analysis for CTCS scheme 1

- Substitute $\phi_j^n = A^n e^{ikj\Delta x}$ into equation [3] to obtain:

$$\frac{A^{n+1}e^{ikj\Delta x} - A^{n-1}e^{ikj\Delta x}}{2\Delta t} = -u\left(\frac{A^n e^{ik(j+1)\Delta x} - A^n e^{ik(j-1)\Delta x}}{2\Delta x}\right) \tag{5}$$

- If now we set $c = u\Delta t/\Delta x$, we can write [5] as:

$$A * A^n e^{ikj\Delta x} - A^n/Ae^{ikj\Delta x} = -cA^n e^{ikj\Delta x}\left(e^{ik\Delta x} - e^{-ik\Delta x}\right)$$

$$A^2 + (2ic\ sin(k\Delta x))A - 1 = 0$$

$$A_+ = -ic\ sin(k\Delta x) + \sqrt{1 - c^2 sin^2(k\Delta x)} \tag{6}$$

$$A_- = -ic\ sin(k\Delta x) - \sqrt{1 - c^2 sin^2(k\Delta x)} \tag{7}$$

Stability analysis for CTCS scheme 2

- There are two cases to consider:
  - $|c| > 1$. In this case for some k, $(c\ sin(k\Delta x))^2 > 1$. Therefore, for some k, at least one of the roots:

$$|A_\pm|^2 = \left[-c\ sin(k\Delta x) \pm \sqrt{(c^2 sin^2(k\Delta x) - 1)}\right]^2 \tag{8}$$

    has modulus greater than one.
  - $|c| \leq 1$, therefore $c\ sin(k\Delta x) \leq 1$. In this case the modulus square of the two roots of the amplification formula are:

$$\begin{aligned} |A_+|^2 &= c^2 sin^2(k\Delta x) + 1 - c^2 sin^2(k\Delta x) = 1 \rightarrow |A_+| = 1 \\ |A_-|^2 &= c^2 sin^2(k\Delta x) + 1 - c^2 sin^2(k\Delta x) = 1 \rightarrow |A_-| = 1 \end{aligned} \tag{9}$$

- *The CTCS scheme is thus conditionally stable for the numerical solution of a linear advection problem with CFL condition* $|c| = |u|\Delta t/\Delta x \leq 1$.

Physical and Computational Modes in CTCS 1

- In the stable case, we have obtained two values for the amplification factor:

$$A_p = -ic\ sin(k\Delta x) + \sqrt{1 - c^2 sin^2(k\Delta x)}$$
$$A_c = -ic\ sin(k\Delta x) - \sqrt{1 - c^2 sin^2(k\Delta x)}$$

- The general form of the numerical solution is thus:

$$\phi_j^n = \left(PA_p^n + CA_c^n\right) e^{ikj\Delta x} \tag{10}$$

with P, C complex constants determined by the initial conditions at $t = 0$:

$$\phi_j^0 = e^{ikj\Delta x} = (P + C)e^{ikj\Delta x} \Rightarrow P + C = 1 \tag{11}$$

- Conveniently choosing $c = 1$, and assuming $\alpha = k\Delta x = uk\Delta t$, we can write $A_{(p,c)}$ in the form:

$$A_p = e^{-i\alpha}; A_c = -e^{i\alpha} \tag{12}$$

Physical and Computational Modes in CTCS 2

- Using [12] in [10] we obtain:

$$\phi_j^n = Pe^{ik(j\Delta x - un\Delta t)} + (-1)^n Ce^{ik(j\Delta x + un\Delta t)} \tag{13}$$

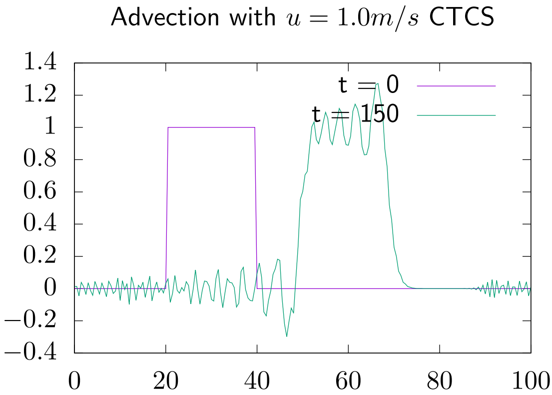
- Using the initial condition in [11]:

$$\phi_j^n = \underbrace{(1 - C)e^{ik(j\Delta x - un\Delta t)}}_{\text{Physical Mode}} + \underbrace{(-1)^n Ce^{ik(j\Delta x + un\Delta t)}}_{\text{Computational Mode}} \tag{14}$$

  - The Physical mode is proportional to the exact solution $e^{ik(x-ut)}$.
  - The Computational mode does not correspond to any solution of the original differential equation.

Computational Mode

- The Computational Mode is an artifact of the numerical method
- The Computational Mode is:
  - oscillating in time from one step to the next step because of the $(-1)^n$.
  - propagating in the opposite direction to the true solution because of the $+un\Delta t$ in the exponential instead of the $-un\Delta t$ in the true solution.



The Robert-Asselin Filter 1

Just looking at the formula in the CTCS scheme in [4] we can note that:

- The solution at the new timestep ϕ_j^{n+1} does not depend from the value of the function at the actual timestep in the same point ϕ_j^n but only from the value in the past at the same point j, ϕ_j^{n-1}.
- To keep the amplitude of the computational mode small it is necessary to couple the solutions on the two sets of alternating grid points between which the solution oscillates.

The most common way to do this commonly used in an atmospheric model is by the use of the Robert-Asselin (RA) filter.

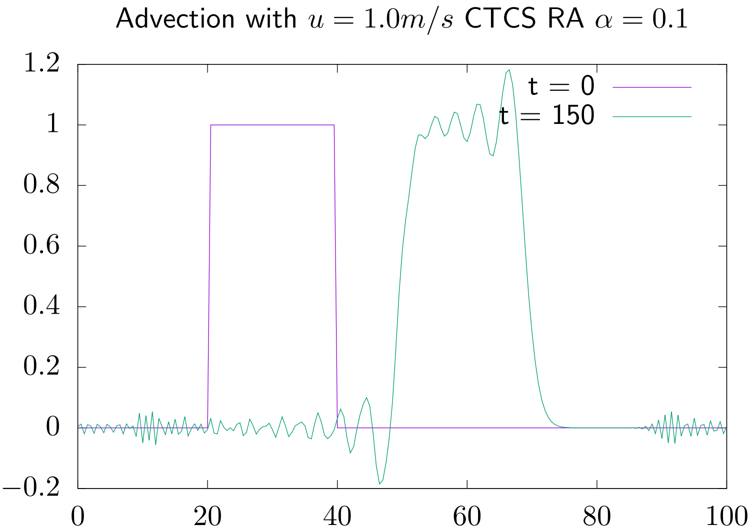
- Compute the filter displacement:

$$d = \alpha(\phi^{n-1} - 2\phi^n + \phi^{n+1}) \tag{15}$$

- Apply the filtering in the update:

$$\begin{aligned}\hat{\phi}^n &= \phi^n + d = (1-2\alpha)\phi^n + \alpha(\phi^{n+1} + \phi^{n-1})\\ \phi^{n-1} &= \hat{\phi}^n\\ \phi^n &= \phi^{n+1}\end{aligned}$$

The Robert-Asselin filter 2



The Robert-Asselin-William Filter 1

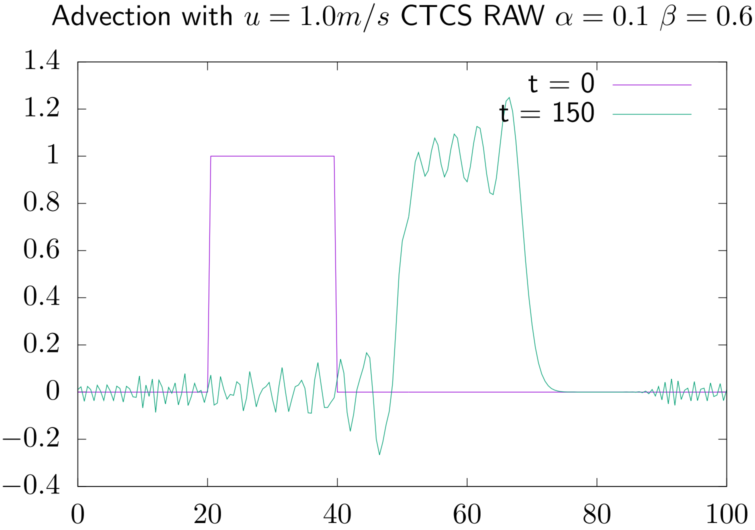
- In the RAW filter we apply the filtering not only to ϕ^n, but also to ϕ^{n+1}.
- As in the RA filter, we compute the filter displacement:

$$d = \alpha(\phi^{n-1} - 2\phi^n + \phi^{n+1}) \tag{16}$$

- Apply the filtering in the update:

$$\begin{aligned}
\hat{\phi}^n &= \phi^n + \beta d \\
\hat{\phi}^{n+1} &= \phi^{n+1} + (1-\beta) d \\
\phi^{n-1} &= \hat{\phi}^n \\
\phi^n &= \hat{\phi}^{n+1}
\end{aligned}$$

The Robert-Asselin-William filter 2



Exercise on CTCS RAW scheme

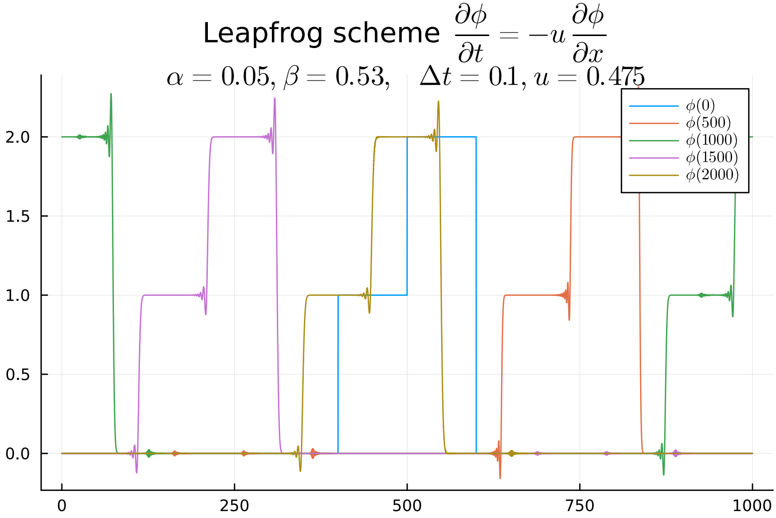
- Write a program to integrate the linear advection equation in [1] using the CTCS scheme in the domain $0 \leq x \leq 1000m$ with advection velocity $u = 0.475m/s$. Let $\Delta x = 0.125m$ and assume periodic boundary conditions. Assume the initial shape to be:

$$\phi(x,0) = \begin{cases} 0.0 & for \quad x < 400 \\ 1.0 & for \quad 400 \leq x < 500 \\ 2.0 & for \quad 500 \leq x \leq 600 \\ 0.0 & for \quad x > 600 \end{cases} \tag{17}$$

- Integrate forward and show solutions from $t = 0s$ up to $t = 2000s$ every $500s$ and explain the characteristics of the solution.
- Repeat the exercise implementing the RA filter with $\alpha = 0.1$.
- Repeat the exercise implementing the RAW filter with $\alpha = 0.05$, $\beta = 0.53$.

Hint: Remember the CFL condition!

Expected result



Julia Code

```
function ftfs(phi_now)
  (1+c) * phi_now - c * circshift(phi_now,-1)
end;

function ftbs(phi_now)
  (1-c) * phi_now + c * circshift(phi_now,1)
end;

function ctcs(phi_old,phi_now)
  phi_old - c * (circshift(phi_now,-1) - circshift(phi_now,1))
end;

phi_old = map(phi0,x);
phi_now = Array(phi_old);
d = Array(phi_old);

if u > 0
  global phi_now = ftbs(phi_old);
else
  global phi_now = ftfs(phi_old);
end;
t = t0+dt;
while (t < t1)
  phi_new = ctcs(phi_old,phi_now);
  global d = alpha * (phi_old+phi_new-2.0*phi_now);
  global phi_old[:] = phi_now + beta*d;
  global phi_now[:] = phi_new + (1-beta)*d;
  global t = t + dt;
  if mod(t,tp) < dt
    plot!(...)
  end;
end;
```