Finite Difference Methods for Hyperbolic PDEs

A. Donev, Courant

The "right way" to solve hyperbolic conservation laws such as the advection equation or the wave eq. is to use finite volume methods.

However, for periodic domains, and up to only 2^{nd} order in space/time, there is no practical difference between FD and FV, so we proceed

We will focus on the advection eq:

$$u_t + (a(x)\,u)_x = 0,$$

written in conservation form as:

$$u_t = -\frac{\partial}{\partial x} f(u, x, t)$$

flux

where the advective flux

$$f = a u$$

gives the amount of conserved quantity transported through the point/plane at x per unit time

②

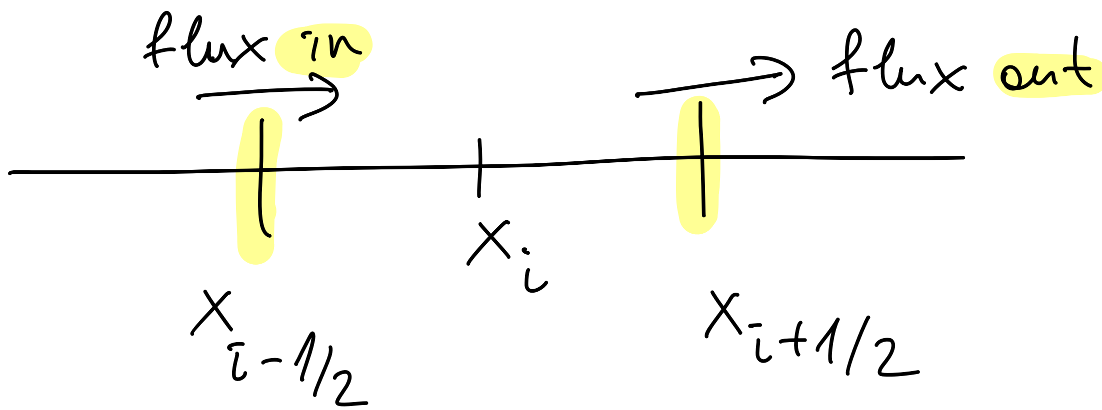
The advective velocity $a(x)$ has units of length / time.

Conservation:

$$\frac{d}{dt}\left[\int_{x-h/2}^{x+h/2} d\tilde{x}\; u(\tilde{x},t)\right] =$$

$$\int_{x-h/2}^{x+h/2} d\tilde{x}\;\partial_t u(\tilde{x},t) = -\int_{x-h/2}^{x+h/2} d\tilde{x}\;\frac{\partial f}{\partial x}(\tilde{x},t)$$

$$= -f\left(u(x+\tfrac{h}{2}), x+\tfrac{h}{2}, t\right) + f\left(u(x-h/2), x-h/2, t\right)$$



if $a > 0$

This is the basis of the FV method but we will not cover it. Nevertheless, understanding the concept of <u>advective flux</u> is crucial to understanding hyperbolic laws & solving them.
The other key concept (see PDE class) are space-time characteristics.

These notes only cover 1D periodic (ring) domains. But, important for future — see class Computational Methods for PDEs, Fall 2023, A. Donev (FV) & G. Stadler (FE) — are

- wave equation

$$u_{tt} = c^2(x)\, u_{xx}$$

- 2D/3D advection:

$$u_t = -\vec{\nabla} \cdot \left(\vec{f}(u, \vec{x}, t) \right) = -\vec{\nabla} \cdot (\vec{a}\, u)$$

where $a(\vec{x}, t)$ is a velocity field

Aside: Wave equation as 1st order system:

density of air — velocity — speed of sound

$$\begin{cases} \partial_t(\rho\vec{v}) = -\nabla(\underbrace{\rho c^2}_{\text{pressure}}) & \leftarrow \text{momentum conservation} \\ \underbrace{\partial_t \rho = -\nabla\cdot(\rho\vec{v})}_{\text{mass conservation}} & \end{cases}$$

$$\partial_{tt}\rho = -\nabla\cdot\partial_t(\rho\vec{u}) = -\nabla\cdot\nabla(\rho c^2) \Rightarrow$$

$$\partial_{tt}\rho = -\nabla^2\left[\rho\,(c(\rho,x,t))^2\right]$$

is a more general wave equation for acoustic waves/sound in air

For now focus on seemingly (!) trivial equation

$$\begin{cases} u_t + a u_x = 0 \\ u(x,0) = \eta(x) \end{cases}$$

$x \in [0, L)$ periodic domain

Aside: In higher dimensions, if $\nabla \cdot \vec{a} = 0$ (incompressible velocity field), then $u_t + \vec{a} \cdot \vec{\nabla} u = 0$

Solution $u(x,t) = \eta(x - at)$ simply translates with speed a to the right if $a > 0$, or to the left if $a < 0$.

Surprisingly, very few numerical methods can obtain the exact solution. And those that do, do not work for non-constant $\vec{a}$!

So we should not try to rely on the fact a is constant in our numerical methods at all.

Why is advection harder than diffusion for numerical methods

[class discussion of properties of heat vs. advection eq.]

Go to Fourier space:

$$\hat{u}_t = -ik\hat{u}$$

$\Rightarrow$ eigenvalues $\lambda_k = -iak$ of PDE are purely imaginary:

No dissipation (smoothing), only transport. Shocks can form for nonlinear PDEs.

$$\|u\|_2 = \|\hat{u}\|_2 = \text{const}$$

But numerical methods will have a hard time with that

9

physical constraint, especially for non-smooth solutions.

Numerical methods introduce artificial

- dissipation: $\mathrm{Re}(\lambda_k) < 0$ for most k
- dispersion: $|\lambda_k| \not\sim |k|$, i.e., different frequencies/wavelengths travel at different speeds - solution is distorted

This is covered in detail in Comp. methods for PDE class. Here we will do a demo in Matlab in class ...

$$u_t = -a u_x$$

Let's try Method-of-Lines (MOL)

Finite-Difference (FD):

$$\frac{d}{dt} u_j = \underbrace{-\frac{a}{2h}\left(u_{j+1} - u_{j-1}\right)}_{\text{centered difference}}$$

$$\frac{d\vec{u}}{dt} = \overleftrightarrow{A}\,\vec{u} \quad \text{(linear ODEs)}$$

$$\vec{u}(t) = \exp\left(\overleftrightarrow{A}\,t\right)\vec{u}(0)$$

$$A = -\frac{a}{2h}\begin{bmatrix} 0 & 1 & & & -1 \\ -1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ 1 & & & -1 & 0 \end{bmatrix}$$

In Fourier space (DFT):

$$\frac{d}{dt}\hat{u}_k = -a\left(\frac{e^{+ikh} - e^{-ikh}}{2h}\right)\hat{u}_k$$

$$\frac{d}{dt}\underrightarrow{\hat{u}} = \hat{A}\,\underrightarrow{\hat{u}}$$ where

12

$$\hat{A} = \text{Diag}\left\{ -\frac{ia}{h} \sin\left(\frac{2\pi}{L} k h\right) \right\}$$

k = wave index

$$\lambda_k = -\frac{ia}{h} \sin(kh) = -iak + O(h^2)$$

purely imaginary

second order

This means we cannot use explicit RK1 (Euler) or RK2,

need at least RK3 for centered advection (explicit) if we want <u>strong</u> stability.

If we only want Lax-Richtmyer stability, for forward Euler

$$\frac{\vec{u}^{\,n+1} - \vec{u}^{\,n}}{\tau} = \overleftrightarrow{A}\,\vec{u}^{\,n}$$

$$\| \overleftrightarrow{I} + \tau \overleftrightarrow{A} \|_2 \leq 1 + \alpha\tau$$

for all $h < h_0$

$$\Rightarrow \quad |1 + \tau \lambda_{max}|^2 \leq 1 + \tau^2 |\lambda_{max}|^2$$

$$= 1 + \alpha\tau$$

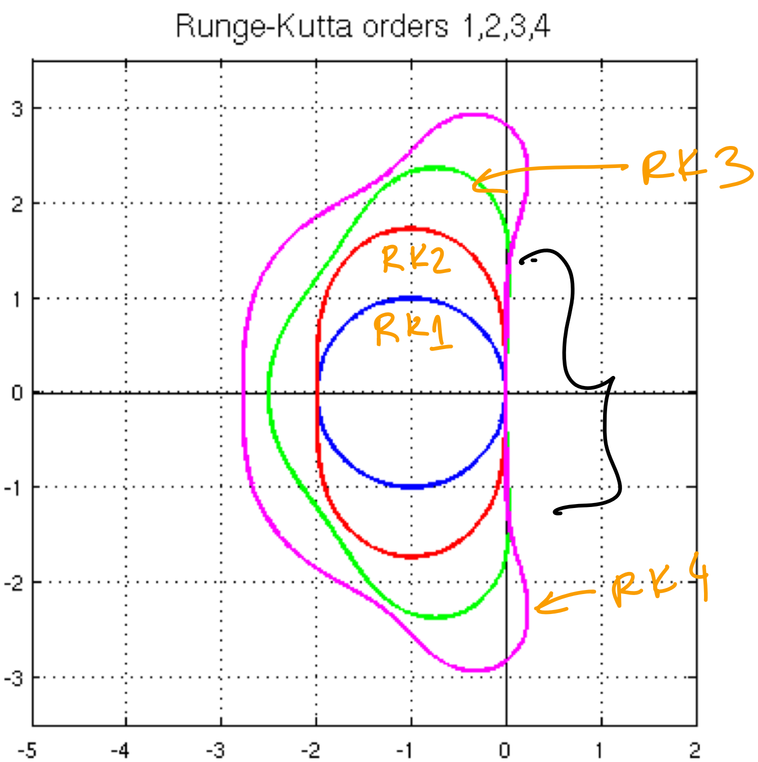
$$|\lambda_{max}| = \left|\frac{ia}{h}\right| = \frac{|a|}{h}$$

$$\Rightarrow \quad 1 + \tau^2 \frac{|a|^2}{h^2} = 1 + \alpha\tau$$

$$\tau = \alpha \frac{h^2}{|a|} \Rightarrow \tau = O(h^2) \text{ for Euler}$$

This does <u>not</u> make physical sense at <u>all</u> for advection, even by physical units $\left(\text{time} = \frac{\text{length}}{\text{speed}}\right)$

Instead, if we use RK3+



absolute stability region includes $[-ic, +ic]$

then we get <u>strong</u> stability if

$$\boxed{\tau \le c \frac{h}{|a|}} \equiv$$ Advective CFL / Courant condition

$c = O(1)$

16

$$\nu = \frac{\tau |a|}{h} \leq C \leftarrow$$ Courant <u>advective</u> number

Now this makes sense physically in terms of <u>domain</u> of dependence of PDE (see 10.7 in LeVeque),

and units make sense too

$$\text{time} = \frac{\text{length}}{\text{space}}$$

Information must not propagate by further than (about) one grid cell per time step

But, RK3 is expensive!

Temporal error $= O(\tau^3) = O(h^3)$

but spatial error $= O(h^2)$,

and recall we want

spatial error $\simeq$ temporal error $\simeq O(h^2)$

How can we accomplish that?

RK2 is not absolutely stable

for imaginary eigs. $\Rightarrow$

We MUST switch to a non-MOL scheme