

# Numerical PDEs

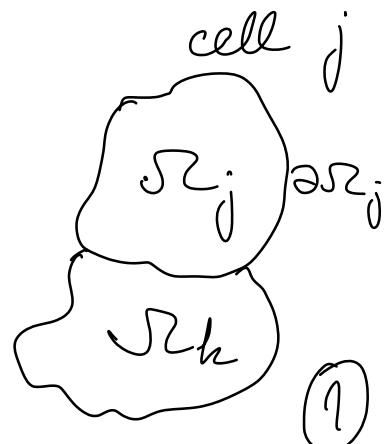
A. DONEV, Fall 2021

## Finite Volume Methods

The bible on this topic is  
book "FVM for hyperbolic problems"  
R. LeVeque  
— freely available as PDF to you

Key idea: Break up domain into  
a grid of cells, and use  
as variables the average  
of  $n$  over each cell

$$\bar{u}_j = \frac{1}{|\Sigma_j|} \int_{\Sigma_j} u \, d\sigma$$



Conservation law

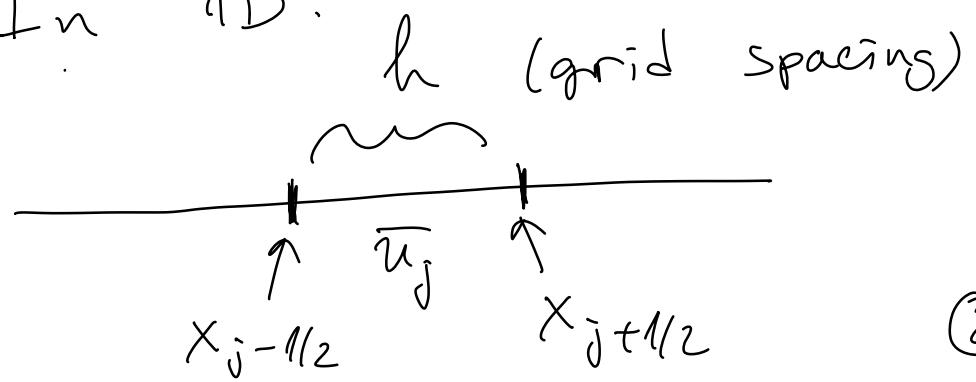
$$\int_{\Sigma_j} \frac{\partial u}{\partial t} dr = - \int (\nabla \cdot \vec{f}) dr$$

$$|\Sigma_j| \frac{d \bar{u}_j}{dt} = - \int_{\partial \Sigma_j} \vec{f} \cdot \vec{n} dA$$

$$\frac{d \bar{u}_j}{dt} = - \frac{1}{|\Sigma_j|} \int_{\partial \Sigma_j} \vec{f} \cdot \vec{n} dA$$

which is a system of ODEs

In 1D:



②

In 1D advection:

$$h \cdot \frac{d}{dt} \bar{u}_j = - \left( f_{j+1/2} - f_{j-1/2} \right) =$$

$$- \left[ a(x_{j+1/2}) u(x_{j+1/2}) - a(x_{j-1/2}) u(x_{j-1/2}) \right] \\ + \left[ d(x_{j+1/2}) u_x(x_{j+1/2}) - d(x_{j-1/2}) u_x(x_{j-1/2}) \right]$$

This is a **weak form of PDE**  
and not (yet) a discretization,  
i.e., it is exact.

To make it into a scheme we  
need to figure out the  
fluxes in terms of the  
cell averages.

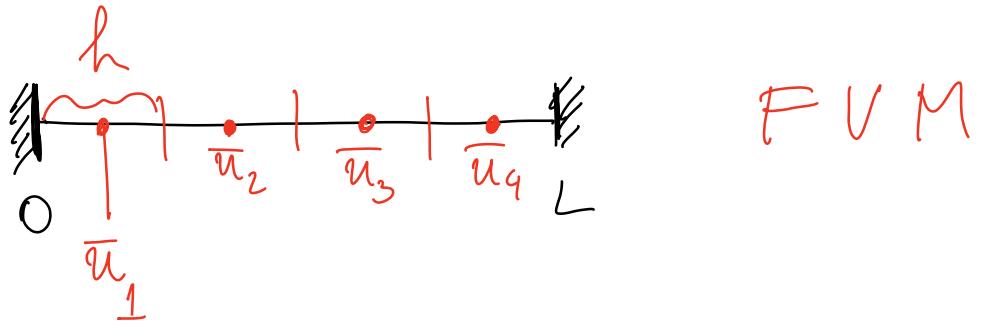
(3)

Note that for low-order (1<sup>st</sup> or 2<sup>nd</sup>) schemes there is really not that much difference between finite difference & finite volume - it is more of a matter of mental picture.

Key to FVM is to write fluxes not divergence of fluxes.

Another difference is with boundary conditions:

Physical boundaries should overlap with cell boundaries for FVM schemes.



We know the cell averages,  
not a function  $u(x)$ .

Constructing an approximation  $u(x)$  from  $\bar{u}$ 's is called reconstruction in FVM.

As with finite difference (FD) methods, there are two main approaches:

- MOL (method of lines): write ODEs for  $\bar{u}_j$  and solve
- Space time schemes: write  $(\bar{u}_j^{n+1} - \bar{u}_j^n)/\Delta t$

⑤

For space time:

$$\frac{\bar{u}_j^{n+1} - \bar{u}_j^n}{\Delta t} = \frac{1}{h} \int_{j-\frac{1}{2}}^{j+\frac{1}{2}} \int_{n\Delta t}^{(n+1)\Delta t} (f(t) - f(\tau)) dt$$

So we need a way to approximate the total or average flux over a time step.

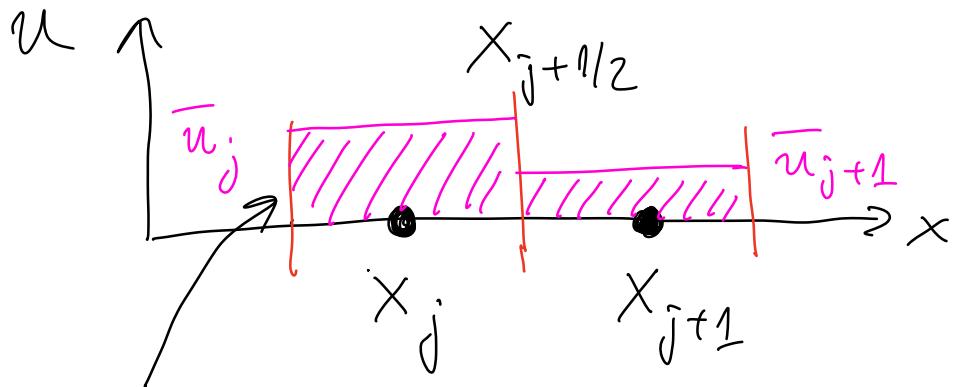
For MOL, we need to estimate:

a) Advection flux

$$f_j^{(a)} = a(x_{j+\frac{1}{2}}) u(x_{j+\frac{1}{2}})$$

↑  
"easy" (evaluate  $a(x)$ )

But how do we get  $u(x_{j+\frac{1}{2}})$  from  $\bar{u}$ 's? This is the key complexity for advection:  
 extrapolate from cell centers to faces. ⑥



Piecewise constant reconstruction

$$u(x_{j-1/2} < x < x_{j+1/2}) = \bar{u}_j + O(h^2)$$

This is discontinuous at

$x_{j+1/2}$ : Do we set?

$$f_{j+1/2} = \begin{cases} a_{j+1/2} \bar{u}_j & \text{or} \\ a_{j+1/2} \bar{u}_{j+1} \end{cases}$$

If  $a_{j+1/2} > 0$  then we know solution moves to the right, i.e., information comes from the left: upwind! (7)

$$f_{j+1/2} = \begin{cases} a_{j+1/2} \bar{u}_j & \text{if } a_{j+1/2} > 0 \\ a_{j+1/2} \bar{u}_{j+1} & \text{otherwise} \end{cases}$$

Upwind flux

For diffusive flux,

"obviously" :

$$f_j^d = d(x_{j+1/2}) u_x(x_{j+1/2})$$

$$\approx d(x_{j+1/2}) \left( \frac{\bar{u}_{j+1} - \bar{u}_j}{h} \right)$$

to  $O(h^2)$ , and this is what is most often used in CFD codes.

(8)

The real challenge in CFD is handling advection, i.e., handling hyperbolic conservation laws more generally. The physical reason for this is:

Advection is non-dissipative,  
& simple advection is "non-dispersive".  
Dissipation stabilizes numerical methods, but we don't want it.

For dispersion, read appendix E.3.9 in FD book of LeVeque (it is excellent!)

Basic idea: In Fourier space

$$\hat{u}(k, t) = e^{-i\omega(k)t} \hat{u}(0, t)$$

where  $k$  is the wave number

(9)

The equation relating  $\omega$  to  $k$   
is called the **dispersion relation**.

It can be found by putting

$$u(x, t) = e^{-i\omega t} e^{ikx}$$

into the PDE for simple  
constant-coefficient equations.

$$c_p(k) = \frac{\omega(k)}{k} \quad \text{is phase velocity}$$

$$c_g(k) = \frac{d\omega(k)}{dk} \quad \text{is group velocity}$$

If  $\omega(k)$  is real then the  
PDE is non-dissipative.

Special case is simple advection:

$$c_p = c_g = a = \text{const.}$$

"non-dispersive"

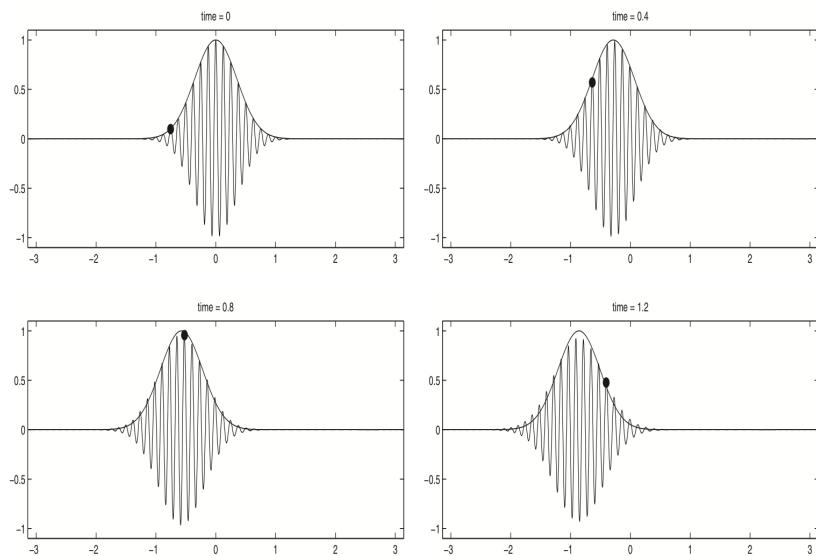
10



Subject to SIAM license or copyright; see <http://www.siam.org/journals/ojsa.php>

328

## Appendix E. Partial Differential Equations



**Figure E.2.** The oscillatory wave packet satisfies the dispersive equation  $u_t + au_x + bu_{xxx} = 0$ . Also shown is a black dot attached to one wave crest, translating at the phase velocity  $c_p(\xi_0)$ , and a Gaussian that is translating at the group velocity  $c_g(\xi_0)$ . Shown for a case in which  $c_g(\xi_0) < 0 < c_p(\xi_0)$ .

$c_g$  determines the speed of propagation of the envelope of the wave packet, while  $c_p$  of an individual peak / crest.

$$\text{E.g. } u_t + a_1 u_x + a_2 u_{xx} + a_3 u_{xxx} +$$

$$a_4 u_{xxxx} = 0 \Rightarrow$$

$$\omega(k) = a_1 k + i a_2 k^2 - a_3 k^3 - i a_4 k^4$$

(11)

$$\hat{u}(k, t) = e^{(a_2 k^2 - a_4 k^4)t} e^{i(a_1 k - a_3 k^3)t} \hat{u}(k, 0)$$

Dissipative  
for  $a_2 < 0, a_4 > 0$

For  $a_2 = a_4 = 0$ ,

$$c_p(k) = a_1 - a_3 k^2$$

$$c_g(k) = a_1 - 3a_3 k^2$$

$$u(x, t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{u}(k, 0) e^{ik(x - c_p(k)t)} dk$$

For a numerical method, find

**numerical dispersion relation**

by plugging into method wave:

$$\bar{u}_j^n = e^{-iwnst} e^{ikjh}$$

(12)

An alternative, which gives us intuition quickly but is not rigorous, is to look at modified equations: find a PDE that the method solves to higher order than it does the PDE we want to solve.

We can do this to a MOL scheme separately for the spatial discretization (the rhs of the system of ODEs)