

# Finite Differences for Parabolic PDEs

A. DONEV, COURANT

Today we consider time-evolving  $u(x, t)$  for linear parabolic PDEs

$$u_t = -\underbrace{\mathcal{L} u}_\text{elliptic operator} + f(x, t) + \text{BCs} + \text{ICs} \quad \dots (*)$$

BCs are the same as for Dirichlet elliptic PDEs, e.g.  
or Neumann if  $\mathcal{L}$  is second order ( $\nabla^2$ ), on the whole boundary.

①

Parabolic PDEs are smoothing  
 (inherited from elliptic regularity)  
 and dissipative (all eigenvalues  
of  $-\mathcal{L}$  are real and non-positive,  
 since  $\mathcal{L}$  is an elliptic operator).

$$u(x, t \rightarrow \infty) \rightarrow u_{\text{ell}}(x)$$

where  $u_{\text{ell}}(x)$  solves the  
 elliptic PDE, assuming  $f = \text{const.}$

$$\mathcal{L}u_{\text{ell}} = f + \text{BCs} \dots \text{(**)}$$

i.e., the initial conditions are  
 forgotten / dissipated / erased.

②

Let a finite-difference (FD) discretization of the elliptic (\*) be

$$\stackrel{\leftrightarrow}{A} \vec{u} = \vec{f} \quad (t = \text{const})$$

Then, the corresponding **method of lines (MOL)** discretization of the parabolic (\*) is:

$$\frac{\vec{u}}{dt} \Leftrightarrow \vec{u}' = -\vec{A} \vec{u} + \vec{f}(t), \quad \vec{u}(0) = \vec{u}_0$$

which is a simple system of linear ODEs! If  $f=0$ , then

$$\vec{u}(t) = \exp(-At) \vec{u}_0$$

(3)

We are thus back to solving ODES. But, the # of variables in the ODE is very large (e.g.,  $10^6$  for 3D) and grows as the grid spacing  $h$  is reduced.

In the limit  $h \rightarrow 0$ , we get an infinite dimensional system of ODES, which is troubling! This is what makes parabolic PDEs much more challenging than (low dimensional) systems of ODES.

Let's focus for now on  
(up to) second order accuracy.  
We could try:

① Forward Euler Explicit

$$\frac{u^{n+1} - u^n}{\bar{\tau}} = -Au^n + f(t^n)$$

$$\Rightarrow u^{n+1} = (I - A\bar{\tau})u^n + \bar{\tau}f(t^n)$$

Key advantage of this is  
that method is explicit so no  
linear systems need to be solved.

⑤

② Implicit midpoint, also  
called (in physics) Crank-Nicolson

$$\frac{u^{n+1} - u^n}{\bar{\tau}} = -A \left( \frac{u^{n+1} + u^n}{2} \right) + f(t^{n+1/2})$$

$$\left( I + \frac{A\bar{\tau}}{2} \right) u^{n+1} = \left( I - \frac{A\bar{\tau}}{2} \right) u^n + f^{n+1/2}$$

which we need to solve  
every timestep

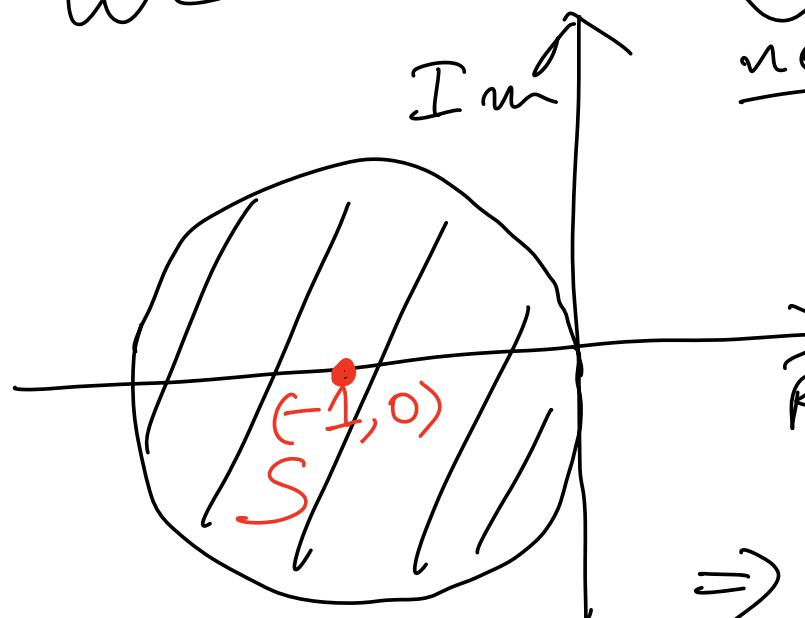
$$u^{n+1} = \left( I + \frac{A\bar{\tau}}{2} \right)^{-1} \left[ \left( I - \frac{A\bar{\tau}}{2} \right) u^n + f^{n+1/2} \right] \quad (6)$$

## Stability

The key question is, how big can  $\Delta t$  be?

## Euler

We want



$$-\lambda_p \bar{z} \in S \text{ if } \underbrace{-\lambda_p}_{\text{negative}} \Rightarrow$$

$$0 \leq \lambda_p \bar{z} < 2 \quad \text{for stability}$$

$$\Rightarrow |\lambda_{\min} \bar{z}| < 2$$

7

For example, consider on  $[0, 1]$ :

$$u_t = k u_{xx} + f(x, t) + \text{Dirichlet BCs}$$

$$A = k \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & 1 & -2 & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & 1 & -2 \end{bmatrix}$$

$$\Rightarrow \lambda_p = \frac{2k}{h^2} (\cos(p\pi h) - 1)$$

$$\Rightarrow \lambda_{\min} = \lambda_{p=1} \approx -\frac{4k}{h^2}$$

(P)

$\Rightarrow$

$$\frac{k\bar{\tau}}{h^2} = \gamma \leq \frac{1}{2}$$

(stability)  
condition

Courant - Friedrichs - Lewy (number)  
CFL condition & number

Physics: In time  $\bar{\tau}$ , diffusion  
spreads the solution over an  
average or typical distance  
 $\sqrt{k\bar{\tau}}$ , and we want  $\sqrt{k\bar{\tau}} \sim h$   
for stability

⑨

For explicit methods, stability requires

$$\Delta t \leq O(1) \frac{h^2}{k}$$

i.e.

$$v \leq O(1)$$
, where

the  $O(1)$  constant depends on the specific explicit temporal integrator used.

For forward Euler:

$$v \leq \frac{1}{2d} \quad \text{for stability}$$

↖ dimension

⑩

Aside: the physics of the CFL condition  
 $\gamma \leq \frac{1}{2d}$  where  $d = \text{dimension}$

The diffusion equation comes from random walks. Here is how. Imagine having many many "particles" / points jumping from one grid cell to one of the neighboring 2d cells (left/right, up/down in 2D). Imagine that a given cell/node has  $N$  particles at time  $t$ . Then, the number of particles

$$(N_1^+, N_1^-, N_2^+, N_2^-, \dots, N_d^+, N_d^-, N_d)$$

10.A

that jump from the given grid cell/node to one of the 2d neighboring cells/nodes, and the number  $N_0$  that stay in the same cell after a time  $\tau$  is given by the multinomial distribution with parameters  $N = N_0 + \sum_{\alpha=1}^d (N_\alpha^+ + N_\alpha^-)$ ,  $k = 2d+1$ , and probabilities

$$\left( \underbrace{\nu, \nu, \dots, \nu}_{2d \nu's}, 1 - 2d\nu \right)$$

Since probabilities are positive,  
 $1 - 2d\nu \geq 0 \Rightarrow \boxed{\nu \leq \frac{1}{2d}}$

CFL condition

(10.B)

How about accuracy? Let's examine the truncation error:

$$\begin{aligned} E(x,t) &= \frac{u(x, t+\bar{\tau}) - u(x, t)}{\bar{\tau}} \\ &- \frac{k}{h^2} [u(x-h, t) - 2u(x, t) + u(x+h, t)] \\ &= \frac{1}{2} \tau u_{tt} + O(\bar{\tau}^2) \leftarrow \text{temporal discretization error} \end{aligned}$$

$$-\frac{k h^2}{12} u_{xxxx} + O(h^4) \leftarrow \text{spatial discretization error}$$

where we assumed  $f=0$

(1)

MOL error = spatial + temporal

$$u_t = k u_{xx} \Rightarrow$$

$$u_{tt} = k (u_{xx})_t = k (u_t)_{xx} \Rightarrow$$

$$u_{tt} = k^2 u_{xxxx} \Rightarrow$$

$$\text{LTE } \epsilon = k \left( \frac{k\bar{\tau}}{2} - \frac{h^2}{12} \right) u_{xxxx} \\ + O(\bar{\tau}^2, h^4)$$

$$\epsilon = \left( \frac{\gamma^2}{2} - \frac{1}{12} \right) kh^2 u_{xxxx} \dots \text{(A)}$$

(12)

Since  $\nu \leq \frac{1}{2}$  for stability,  
 we see that for forward Euler,  
 $\nu = 1/6$  makes the spatial and  
temporal discretization/truncation  
 errors (approximately) equal  
 in magnitude

This is a good choice for  $\nu$ !

$$\tilde{\epsilon} \sim \frac{h^2}{[4-6]k} \Rightarrow \text{LTE} \sim (ku_{xxxx}) h^2$$

(13)

$$\bar{\epsilon} = O(h^2) \Rightarrow \text{LTE} = O(h^2) \text{ for Euler}$$

So the method is second-order accurate in the grid spacing  $h$ !  
Even though we used a first order temporal integrator, we expect to get 2<sup>nd</sup> order in space-time. Since  $\bar{\epsilon} \sim h^2$ , we say that the error is 2<sup>nd</sup> order under space-time refinement.  
(or 2<sup>nd</sup> order in space-time) (14)

## Crank - Nicolson (implicit)

We know that implicit midpoint method is A-stable, so the method is **unconditionally stable**.

But of course  $\bar{z}$  cannot be arbitrarily large for **accuracy**.

For CN, we have

$$\text{LTE} = \underbrace{\mathcal{O}(h^2)}_{\text{spatial}} + \underbrace{\mathcal{O}(\bar{z}^2)}_{\text{temporal}}$$

We want  $|\text{spatial}| \approx |\text{temporal}|$

$$\bar{\tau} = O(h) \Rightarrow LTE = O(h^2) \text{ for CN}$$

As  $h \rightarrow 0$ ,  $O(h^2) \ll O(h)$ ,  
so CN can take a much  
larger time step size than  
Euler for fine grids (high  
accuracy)! But, each time step  
is more expensive because we  
need to solve. Order of  
accuracy is the same, so the  
choice of method is dictated  
by computational cost!

To compare computational cost, however, we need to estimate:

- ① The constant  $C$  in  $\bar{r} = C h/k$
- ② How expensive it is to solve the linear system in  $CN$ .

Let's start with the easier question #2, solving

$$\left(I + \frac{\alpha \bar{r}}{2}\right) u^{n+1} = \left(I - \frac{\alpha \bar{r}}{2}\right) u^n + f^{n+1/2}$$

What types of solvers might we use?

17

What if we used an iterative solver like (P)CG?

Condition number:

$$K_2(I + \frac{\bar{z}A}{2}) = \left| \frac{1 + \bar{z}/2 \lambda_{\max}^{(A)}}{1 - \bar{z}/2 \lambda_{\min}^{(A)}} \right|$$

$$\approx 1 + \frac{8k\bar{z}}{2h^2} \quad \approx 1 + 42$$

$$1 + \frac{\pi^2 k \bar{z}}{2L^2} \approx 1$$

↑  
length of domain

$$k_2 \approx 1 + \frac{4k\bar{\epsilon}}{h^2} = 1 + \frac{4C}{h}$$

As  $h \rightarrow 0$ ,  $\bar{\epsilon} \rightarrow \infty$  for  $CN$ ,  
so for small  $h$  we have

$$\boxed{k_2 \approx 4\bar{\epsilon}} > 1$$

The number of iterations required to reach a certain tolerance in CG is  $\sim \sqrt{k}$

$$\# \text{ iterations} \sim 2\sqrt{2} = 2\sqrt{\frac{k_2}{h^2}} = 2\sqrt{\frac{C}{h}}$$

(19)

Each iteration of CG costs the same as a step of forward Euler (roughly), so

$$\# \text{ CN step} \Leftrightarrow \sqrt{\frac{c}{h}} \text{ Euler steps}$$

So we are back to question #1, how large can  $c$  be?

This is a little bit of a tricky question. We will answer it using physical intuition

instead of (tedious) math 😊

20

Let us denote the smallest  
relevant length scale in the  
specific physical problem of  
interest with  $\ell < L$ . This  
is problem specific and one must  
use physical understanding  
instead of trying to derive some  
sort of universal or a priori  
error estimate (this is again  
why PDEs are different / harder  
than ODEs !)

The time step size should resolve the dynamics at scale  $l$ , and the grid should resolve the scale  $l$ . So at the coarsest sensible grid,

$$h \sim \frac{l}{r}, \quad \bar{\tau} = \frac{1}{r} \frac{l^2}{k}$$

where the resolution factor  $r \sim [2-8]$  or so.

$$\bar{\tau} = \frac{ch}{k} = \frac{cl}{fk} = \frac{l^2}{fk}$$

$$\Rightarrow c \approx l \Rightarrow \boxed{\bar{\tau} \sim \frac{lh}{k}}$$

$\Rightarrow$  CN step  $\approx \sqrt{\frac{l}{h}}$  Euler steps

To reach time  $T$ , we need  
 $\frac{T}{\bar{\tau}} = \begin{cases} \frac{6Tk}{h^2} & \text{Euler steps} \\ \frac{Tk}{lh} & \text{CN steps} \end{cases}$

(23)

Total computational cost :

$$\frac{6T\kappa}{h^2} \cdot (\text{FD apply}) \text{ for Euler}$$

$$\frac{T\kappa}{\ell h} \cdot \sqrt{\frac{\ell}{h}} \cdot (\text{FD apply}) \text{ for CN}$$

$$\frac{CPU_{Euler}}{CPU_{CN}} = \frac{\frac{6T\kappa}{h^2}}{\frac{T\kappa}{\ell h}} = 6 \sqrt{\frac{\ell}{h}} > 1$$

often  $\gg 1$

So CN always beats Euler,

(24)

and really beats Euler if  
 $h \ll \ell$  (well-resolved simulation).  
Sometimes, however, our simulations  
may be under-resolved, and then  
 $h \sim \ell$  and CN is not more  
efficient but is a lot more  
work to implement (e.g., linear  
solver + BCs is hard in 3D!).  
So both methods have their  
place. But most codes use CN.

## Convergence of FD methods

For a linear PDE any one-step method will take the form

$$u^{n+1} = B(\bar{z}) u^n + f^n(\bar{z})$$

e.g.  $B = \begin{cases} \left(I + \frac{\bar{z}A}{2}\right)^{-1} \left(I - \frac{\bar{z}A}{2}\right) & CN \\ (I - \bar{z}A) & FE \\ \left(I + \bar{z}A\right)^{-1} & BE \end{cases}$

Def: A method is Lax-Richtmyer stable if

$$\forall T > 0, \exists C_T > 0 \text{ s.t.}$$

$$\boxed{\|B^n(\bar{\tau})\| \leq C_T}$$

$$\forall n \leq \frac{T}{\bar{\tau}}$$

$$\forall \bar{\tau} > 0$$

A fundamental result (obtained at Courant) is the Lax equivalence theorem:

A consistent method is convergent iff it is (Lax-Richtmyer) stable consistency + stability ( $\Rightarrow$ ) convergence

Proof sketch:

Error  $E^n = U^n - \hat{U}^n$   $\leftarrow$  true solution

$$E^{n+1} = BE^n - \bar{\epsilon} * (LTE)^n$$

$$\Rightarrow \|E^n\| \leq TC_T \max_n \|(LTE)^n\|$$

So if  $LTE = O(h^p) \Rightarrow$

$$\|E^n\| \leq TC_T \cdot O(h^p)$$

and the method is  $p$ -th  
order accurate

The weakest possible condition for L.-R. stability is

$$\|B(\bar{\tau})\| \leq 1 + \alpha \bar{\tau}$$

$$\Rightarrow \|B^n\| \leq (1 + \alpha T)^{n/\bar{\tau}} \leq e^{\alpha T} = C_T$$

But often we prefer methods that satisfy strong stability

$$\|B\| \leq 1$$

For example, absolutely stable methods are strongly stable.

But, interestingly, absolute stability is not required per se for convergence. We will see an example for hyperbolic eqs. with forward Euler soon.

Zero stability is not meaningful for PDEs, we seek instead absolute (strong) stability.

$\bar{\lambda} \sim h^q$ , eigenvalues of ODE depend on  $h$ , so  $\bar{\lambda}$   $\not\rightarrow 0$  guaranteed that  $\bar{\lambda} \lambda_{\max} \rightarrow 0$  as  $h \rightarrow 0$ !  
③〇

This is in contrast with ODEs, where  $\bar{\tau} \lambda_{\max} \rightarrow 0$  as  $\bar{\tau} \rightarrow 0$ , so one-step methods always converge.

Furthermore, there are very few cases where convergence can be proven / guaranteed for nonlinear PDEs, but this can be done for general (low-dimensional) ODEs.

## von Neumann stability analysis

As a first step when analyzing FD methods, we consider periodic BCS and use a Fourier transform, to obtain the trial ODEs:

$$\frac{d \hat{U}_k}{dt} = -\lambda_k \hat{U}_k + f_k(t)$$

symbol of FD elliptic operator

Any scheme will give ( $t=0$ ):

$$\hat{U}_k^{n+1} = g(k; \bar{\tau}) \hat{U}_k^n, \quad k \in [0, \frac{T}{h}]$$

amplification factor

L.-R. stability requires

$$|g(k; \bar{\tau})| \leq 1 + \alpha \bar{\tau}$$

which is a very general result that is not limited to MOL methods or to parabolic PDEs (e.g., it applies to space-time methods for hyperbolic PDEs)

(33)

For example, for  $u_t = K u_{xx}$   
 in 1D we get

$$\hat{L}_k = \frac{1}{h^2} (e^{ikh} - e^{-ikh})$$

$$= \frac{2}{h^2} (\cos(kh) - 1), |k| < \frac{\pi}{h}$$

$$|g_k| = |1 + \bar{\tau} \hat{L}_k| \leq 1 \quad \text{for strong stability}$$

$$\Rightarrow \left| 1 - \frac{4K\bar{\tau}}{h^2} \right| \leq 1 \Rightarrow \bar{\tau} \leq \frac{h^2}{2K}$$

At home: Find stability limit  
 on  $\bar{\tau}$  in 2D for 5 pt Laplacian

(34)