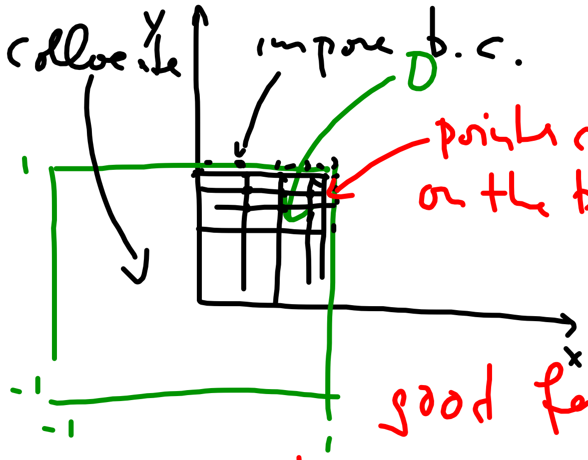
§ 6.2 2D Boundary value probl.

Ex.: $u_{xx}+u_{yy}=f(x,y)$ over $(x,y)=[-1,1]^2$

b.c. $u=0$ ∂D



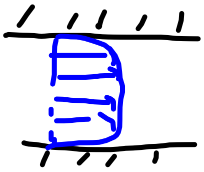
points concentrate on the boundary

$$x_j=\cos\frac{\pi(2j-1)}{2N}$$

$$y_k=\cos\frac{\pi(2k-1)}{2M}$$

$$\left.\begin{array}{l} j=1,\dots,N \\ k=1,\dots,M \end{array}\right\} NM \text{ points}$$

good for boundary layer, example: channel flow



$$u = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} a_{nm} T_n(x) T_m(y)$$

For $j = 2, \dots, N-1$; $k = 2, \dots, M-1$ collocate:

$$\sum_{n=0}^{N-1} \sum_{m=0}^{M-1} a_{nm} \left(T_n''(x_j) T_m(y_k) + T_n(x_j) T_m''(x_k) \right)$$

$= f(x_j, y_k)$: $(N-2)(M-2)$ equations

\# of variables a_{nm}: NM

b.c. $\sum \sum a_{nm} T_n(-1) T_m(y_k) = 0 \quad , k = 1, \dots, M$

$T_n(1)$

$$\sum \sum a_{nm} T_n(x_j) T_m(-1) = 0 \qquad j = 2, \dots, N-1$$

$T_m(1)$

\# of b.c. $2M + 2(N-2) + (N-2)(M-2) = NM$

\# of equations = \# of variables

Now we have to write as

$\underline{\underline{A}}\,\underline{\hat{u}} = \hat{f}$ write $a_{nm} = \hat{u}_{mN+n+1}$

length NM

$NM \times NM$ matrix

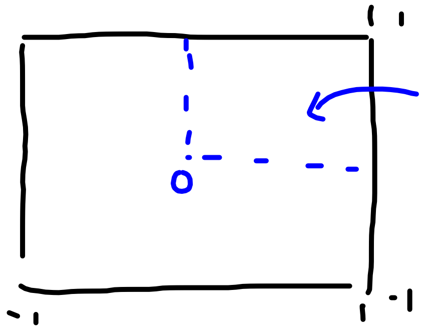
$\rightsquigarrow$ build in as much structure as possible

$$\underline{\hat{u}} = \begin{bmatrix} a_{00} \\ a_{10} \\ \vdots \\ a_{N-1\,0} \\ a_{01} \\ \vdots \\ a_{N-1\,1} \\ \vdots \\ \vdots \\ a_{N-1\,M-1} \end{bmatrix}$$

▶ build in symmetries

for example $f(-x,y) = f(x,y) = f(x,-y)$

with $u=0$ DD $\rightsquigarrow$ u has the same symm.

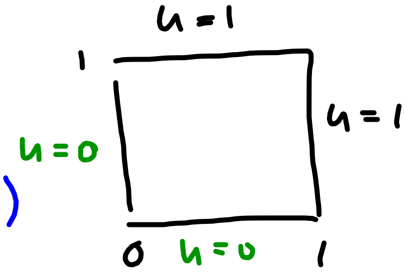


solve here

Write $u = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} a_{nm} T_{2n}(x) T_{2m}(y)$

has the right symmetries.

2nd example $u_{xx} + u_{yy} = f(x,y)$ with b.c.

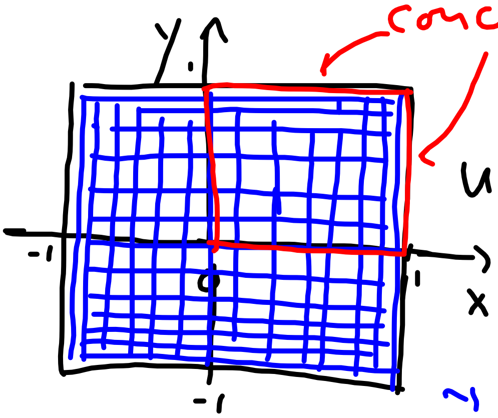


a) expansion

$$u = \sum \sum a_{nm} T_n(2x-1) T_m(2y-1)$$

and impose b.c. directly

b) expand domain and assume symmetries (odd) concentration only here!



odd symmetry:

$$u = \sum \sum b_{nm} T_{2n+1}(x)\, T_{2m+1}(y)$$

automatically $u(x,0) = 0$

$u(0,y) = 0$

$\rightsquigarrow$ explicitly impose

$$u(1,y) = u(x,1) = 1$$

§ 6.3 Chebyshev for time-dependent PDEs

⇝ spectral in space, finite difference in time. Explicit method has very severe time constraints owing to concentration of points! ⇝ implicit methods

Nevertheless consider leapfrog as described in [4], p. 80

$$\frac{\underline{u}^{n+1} - 2\underline{u}^{n} + \underline{u}^{n-1}}{(\Delta t)^2} = \underline{\underline{D}}^{(2)} \underline{u}^{n}$$

Consider the equation
$u_{tt} = u_{xx}$
$x \in [-1, 1]$

stability boundary
$\Delta t = \frac{9.2}{N^2}$ why?

Consider ODE $u_{tt} = \lambda u$

general sol. $u^n = A z_1^n + B z_2^n$

substitute $u^n = z^n$

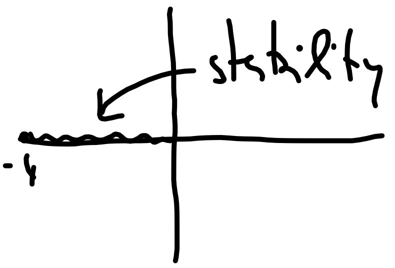
characteristic eqn : $z^2 - [2 + \lambda(\Delta t)^2]\, z + 1 = 0$

$z_1 z_2 = 1 \qquad z + \frac{1}{z} = 2 + \lambda \Delta t^2$; $z = e^{i\vartheta}$

$\vartheta = \arccos\left(1 + \frac{1}{2}\lambda(\Delta t)^2\right)$ $\rightsquigarrow$ must have

$-4 < \lambda(\Delta t)^2 < 0$

$\Delta t < \sqrt{\frac{-4}{\lambda}}$



Back to PDE: largest eigenvalue of $\underline{\underline{D}}^{(2)}$ is $-0.048\, N^4$

because smallest grid spacing $\sim \frac{1}{N^2}$.

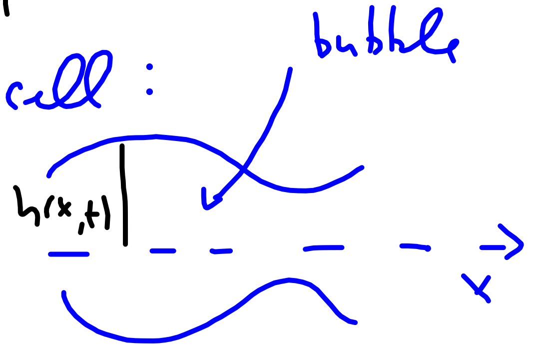
Put together

$$\Delta t < \sqrt{\frac{4}{0.048}}\,\frac{1}{N^2} \simeq \frac{9.2}{N^2}$$

bad, better use implicit scheme

§ A nonlinear equation

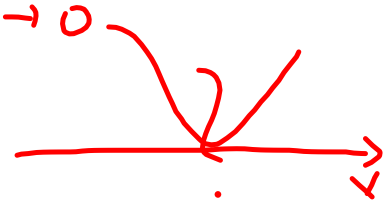
Flow in a Hele-Shaw cell:



$$h_t + (h\, h_{xxx})_x = 0$$

Interested in pinch-off

$h \to 0$

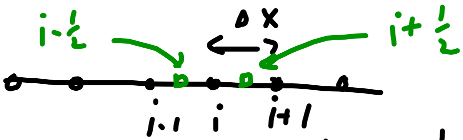


equation changes character:

$$(h\, h_{xxx})_x = \underbrace{h}_{\to 0} h_{xxxx} + h_x h_{xxx}$$

⇝ unconditionally stable!

Finite difference:



$$\frac{h_i^{n+1}-h_i^{n}}{\Delta t} = \frac{f_{i+\frac{1}{2}}^{n+1}-f_{i-\frac{1}{2}}^{n+1}}{\Delta x}$$

backwards Euler

$$u_i = h_i'' = \frac{h_{i+1}-2h_i+h_{i-1}}{\Delta x^2}$$

$$f_{i+\frac{1}{2}} = \frac{(u_{i+1}-u_i)(h_{i+1}+h_i)}{2\Delta x}$$

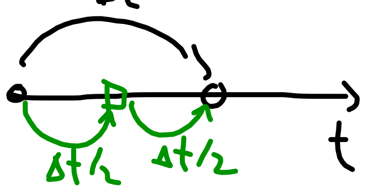
Since the equation changes character, but to solve the nonlinear set of equations.
Use Newton's method.

Stability: $\dfrac{w_{n+1}-w_n}{\Delta t} = \lambda w_{n+1}$

$w_n = A z^n \rightsquigarrow z = \dfrac{1}{1-\bar{\lambda}}$, $\bar{\lambda} = \lambda \Delta t$

But only 1st order accurate!

Use Richardson extrapolation



1st order scheme: $w_{n+1} = w_{n+1}^{exact} + c\Delta t^2 + O(\Delta t)^3$

two steps: $w_{n+1}^{(2)} = w_{n+1}^{exact} + 2c\left(\frac{\Delta t}{2}\right)^2 + \dots$

Combine the two:

$$w_{n+1}^{new} = 2\,w_{n+1}^{(2)} - w_{n+1} = u_{n+1}^{exact} + O(\Delta t^3)$$

2nd order!

Stability:

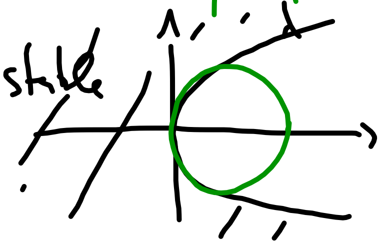
$$z_R = 2\,z^2\left(\frac{\bar\lambda}{2}\right) - \underbrace{z(\bar\lambda)}_{\text{backw. Euler}}$$

$$\boxed{z_R = 2\left(\frac{1}{1-\bar\lambda/2}\right)^2 - \frac{1}{1-\bar\lambda}}$$

stability iff $|z_R| \le 1$

unconditionally stable!

stable



- In case of stiff equations stability is everything!
- Always use step size control, because the typical time scale of a non-linear equation cannot be predicted!

Very simple option: use the difference

$\Delta_i = \left(W^{(2)}_{n+1} - W_{n+1}\right)_{(i)}$ and require

ith gridpoint

$\Delta_i < 0.01$ for all grid points

- If not reduce time step $\Delta t \rightsquigarrow \frac{\Delta t}{2}$

- if satisfied (for a while) increase slowly

$\Delta t = \Delta t \, 2^{1/6}$

For the Hele-Shaw problem, this will cause Δt to decrease by many orders of magnitude as $h \to 0$.

Also: use spatial refinement as well

adapt, adapt, adapt ...