The Finite differences method for elliptic PDE (§ 12.1)

In 2D an elliptic PDE w/ constant coeff has the form

$$a_1 u_{xx} + a_2 u_{xy} + a_3 u_{yy} + a_4 u_x + a_5 u_y + a_6 u = f$$

where the coeff a_1, a_2, a_3 satisfy:

$$a_2^2 - 4a_1 a_3 < 0$$

Typical elliptic PDE: comes from steady state of heat conduction:

$$u_t = (K u_x)_x + (K u_y)_y + \Psi$$
$$= \nabla \cdot [K \nabla u] + \Psi$$

Recall:
$$\nabla u = \begin{bmatrix} \partial u/\partial x \\ \partial u/\partial y \end{bmatrix}$$
$$\nabla \cdot \underline{v} = \partial v_1/\partial x + \partial v_2/\partial y.$$

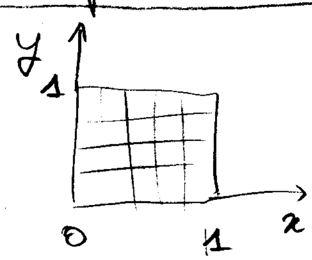
steady state: $u_t = 0$

$$\Rightarrow \quad \nabla \cdot [K \nabla u] = f = -\Psi$$

We will study Poisson's problem:

$$(*) \quad \boxed{\Delta u = f}$$

The 5-point stencil for the Laplacian



$$x_i = i\Delta x, \; i = 0 .. m+1, \quad \Delta x = \frac{1}{m+1}$$
$$y_j = j\Delta y, \; j = 0 .. n+1 \quad \Delta y = \frac{1}{n+1}$$

Let $u_{ij} \approx u(x_i, y_j)$ then we discretize the Poisson eq $(*)$ by replacing the x and y derivatives by finite differences:

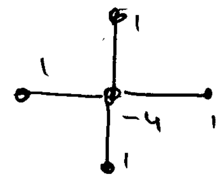
$$\frac{1}{(\Delta x)^2}(u_{i-1,j} - 2u_{ij} + u_{i+1,j}) + \frac{1}{(\Delta y)^2}(u_{i,j-1} - 2u_{ij} + u_{i,j+1}) = f_{ij}$$

when $\Delta x = \Delta y = h$ we obtain

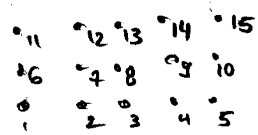
$$\boxed{\frac{1}{h^2}\left(u_{i-1,j} + u_{i+1,j} + u_{ij-1} + u_{ij+1} - 4u_{ij}\right) = f_{ij}}$$

(linear system)

This is known as a 5-point stencil:



If the nodes in grid are ordered in lexicographical (read by row) order



then the system matrix is of the form:

$$A = \frac{1}{h^2}\begin{bmatrix} T & I & & & \\ I & T & I & & \\ & I & T & I & \\ & & \ddots & \ddots & I \\ & & & I & T \end{bmatrix}, \quad \text{where} \quad T = \begin{bmatrix} -4 & 1 & & & \\ 1 & -4 & 1 & & \\ & 1 & -4 & 1 & \\ & & & \ddots & 1 \\ & & & 1 & -4 \end{bmatrix} \in \mathbb{R}^{m\times m}$$

$I = m \times m$ identity.

suitable for direct sparse methods or iterative methods.

The accuracy of this method can be analyzed as the 1D finite differences: plug true solution into the finite difference scheme we obtain:

$$\tau_{ij} = \frac{1}{h^2}\Big(u(x_{i-1},y_j) + u(x_{i+1},y_j) + u(x_i,y_{j-1}) + u(x_i,y_{j+1}) - 4u(x_i,y_j)\Big) - f(x_i,y_j)$$

= local truncation error

$$= \frac{1}{12}h^2\left(u_{xxxx} + u_{yyyy}\right) + O(h^4) \quad = \quad \text{SECOND ORDER accurate.}$$

(simply a combination of 1D analysis in x and y direction)

Here is a nifty trick to construct discretization matrix in Matlab:

```
I = eye(m);
e = ones(m,1);
T = spdiags([e -4*e e],-1:1,m,m);
S = spdiags([e e],[-1,1],m,m);
A = (kron(I,T) + kron(S,I))/h^2;
```

replicates T in pattern given by I

replicates I in pattern given by S.

These are all sparse operations, so they are relatively cheap, even for millions of variables!

Parabolic problems

Typical parabolic problem is heat equation or diffusion eq:

$$\begin{cases} u_t = K u_{xx}, & K > 0. \\ u(0,t) = g_0(t) & \text{B.C. e.g. Dirichlet} \\ u(1,t) = g_1(t) & \\ u(x,0) = \eta(x) & \text{I.C.} \end{cases}$$

Idea put together spatial discr + time discr:

$x_i = ih, \quad i = 0..m+1, \quad h = \frac{1}{m+1} = \Delta x$

$t_n = nk, \quad n = 0, \ldots, \quad k = \Delta t$ = time step.

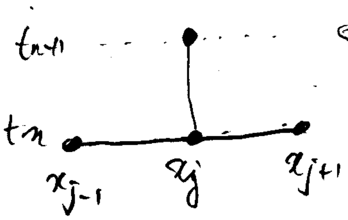
Let $U_i^n \approx u(x_i, t_n)$.

One possible discretization is:

$$\frac{U_i^{n+1} - U_i^n}{k} = \frac{1}{h^2}\left(U_{i-1}^n - 2U_i^n + U_{i+1}^n\right) \quad (\sim \text{Euler})$$

Explicit since:

$$u_i^{n+1} = u_i^n + \frac{k}{h^2}\left(u_{i-1}^n - 2u_i^n + u_{i+1}^n\right)$$



stencil

Another practical method is Crank-Nicholson:

$$\frac{U_i^{n+1} - U_i^n}{k} = \frac{1}{2}\left(D^2 U_i^{n+1} + D^2 U_i^n\right)$$

$$= \frac{1}{2h^2}\left(U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1} + U_{i-1}^n - 2U_i^n + U_{i+1}^n\right)$$

Putting all new values of U on the same side:

$$-r\,U_{i-1}^{n+1} + (1+2r)\,U_i^{n+1} - r\,U_{i+1}^{n+1} = r\,U_{i-1}^n + (1-2r)\,U_i^n + r\,U_{i+1}^n$$

where $r = \frac{k}{2h^2}$. This is an implicit method: we need to solve a system to find U_i^{n+1}. Fortunately this is a tridiag system:

$$\begin{bmatrix} 1+2r & -r & & & \\ -r & 1+2r & -r & & \\ & -r & 1+2r & -r & \\ & & \ddots & \ddots & -r \\ & & & -r & 1+2r \end{bmatrix} \begin{bmatrix} U_1^{n+1} \\ U_2^{n+1} \\ \vdots \\ U_{m-1}^{n+1} \\ U_m^{n+1} \end{bmatrix} = \begin{bmatrix} r(g_0(t_n)+g_0(t_{n+1})) + (1-2r)\,U_1^n + r\,U_2^n \\ r\,U_1^n + (1-2r)\,U_2^n + r\,U_3^n \\ r\,U_2^n + (1-2r)\,U_3^n + r\,U_4^n \\ \vdots \\ r\,U_{m-2}^n + (1-2r)\,U_{m-1}^n + r\,U_m^n \\ r\,U_{m-1}^n + (1-2r)\,U_m^n + r(g_1(t_n)+g_1(t_{n+1})) \end{bmatrix}$$

where we used B.C. $u(0,t) = g(0,t) \equiv U_0^n$

$u(1,t) = g(1,t) = U_{m+1}^n$

Local truncation error: We obtain L.T.E. in the same way: we replace approximation U_i^n by true solution and quantify defect of discrete scheme:

let $\tau_i^n = \tau(x_i, t_n)$ where

$$\tau(x,t) = \frac{u(x,t+k) - u(x,t)}{k} - \frac{1}{h^2}\left(u(x-h,t) - 2u(x,t) + u(x+h,t)\right)$$

u smooth

$$= \left(u_t + \frac{1}{2}k\,u_{tt} + \frac{1}{6}k^2 u_{ttt} + \dots\right) - \left(u_{xx} + \frac{1}{12}h^2 u_{xxxx} + \dots\right)$$

Because $u_t = u_{xx}$ (PDE!) the first term cancels out.

$u_{tt} = (u_{xx})_t = (u_t)_{xx} = u_{xxxx}$

$$\Rightarrow \tau(x,t) = \left(\frac{1}{2}k - \frac{1}{12}h^2\right)u_{xxxx} + O(k^2 + h^4)$$

Thus this is a
- second order accurate scheme in space
- first order accurate scheme in time

since $\tau(x,t) = \mathcal{O}(k + h^2)$.

It is possible to show that

$$\tau(x,t) = \mathcal{O}(k^2 + h^2) \quad \text{for } \underline{\text{Crank-Nicholson}}$$

Method of lines discretization

good for the theory, not too efficient in practice.

Idea: discretize in space first $\rightsquigarrow$ obtain a system of ODE's where each component being at a grid point. e.g.:

$$U_i'(t) = \frac{1}{h^2}\left(U_{i-1}(t) - 2U_i(t) + U_{i+1}(t)\right) \quad \text{for } i = 1, 2, \dots, m$$

$x_0 \longrightarrow U_0(t)$

$x_1 \longrightarrow U_1(t)$

$\vdots$

$x_m \longrightarrow U_m(t)$

$x_{m+1} \longrightarrow U_{m+1}(t)$

In matrix form this system of ODEs becomes:

B.C.

$$(*) \qquad U'(t) = AU(t) + g(t)$$

$$A = \frac{1}{h^2}\begin{bmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix} \qquad g(t) = \frac{1}{h^2}\begin{bmatrix} g_0(t) \\ 0 \\ \vdots \\ 0 \\ g_1(t) \end{bmatrix}$$

Could use ODE software to discretize $(*)$ in time (RK4 for systems etc...) however specially designed methods can be more efficient.

Applying ODE method we get approx. $V_i^n \approx V_i(t_n) \approx u(x_i, t_n)$ and stability analysis will depend on A which depends on spatial discretization.

Example: Euler's method:

$$U^{n+1} = U^n + k f(U^n) \qquad (f(U) = AU + g)$$

Trapezoidal rule:

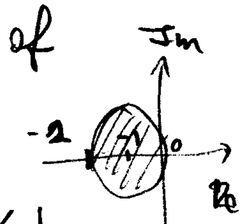
$$\frac{U^{n+1} - U^n}{k} = \frac{1}{2}\left(f(U^n) + f(U^{n+1})\right) \rightsquigarrow \text{Crank-Nicholson}$$

RK $\to$ give different method, more accurate in time but still with the second order accuracy in space we chose.
$\to$ need to change space discr. as well to obtain more accurate method.

Stability: Consider the ODE $\begin{cases} u' = \lambda u \\ u(0) = 1 \end{cases}$

Recall the region of absolute stability is region R of $\mathbb{C}$ where $\lambda k \in R \Rightarrow$ method decays to zero
(k: time step)

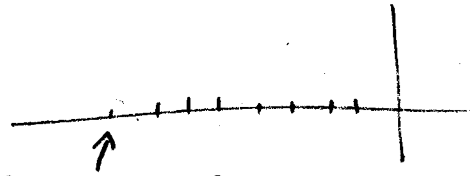
For systems of ODE's what matters are eigenvalues of linearization, in our case A.



Euler's method: stability region is $|1 + k\lambda| < 1$

eigenvalues of A are: $\lambda_p = \frac{2}{h^2}\left(\cos(p\pi h) - 1\right)$
$\left(h = \frac{1}{m+1}\right)$

So eigenvalues of A lie in negative real axis:



farthest is $\lambda_m \approx -4/h^2$

Thus requiring that all eigenvalues lie inside stability region for Euler's method gives:

$$\left|1 - \frac{k4}{h^2}\right| \leq 1 \quad =$$

$$-2 \leq -\frac{4k}{h^2} \leq 0 \quad \Rightarrow \quad \boxed{\frac{k}{h^2} \leq \frac{1}{2}}$$

very restrictive: time step has to be squared space step!

Example: Trapezoidal rule → Crank Nicholson

But trap. rule is A-stable (meaning absolute stability region is whole left hand $\mathbb{C}$ plane). So Crank-Nicholson is stable for any time step. (note: of course large time steps will give inaccurate answers)

Convergence methods we've seen are of the form:

$$(**) \quad U^{n+1} = \underbrace{B(k)}_{\in \mathbb{R}^{m \times m}} U^n + \underbrace{b^n(k)}_{\in \mathbb{R}^m} \quad , \quad h = \frac{1}{m+1}$$

we will assume $h = R(k)$, i.e. that we have fixed spatial discretization with time step according to some rule. e.g by analysis we did before we can use $k = 0.4 h^2$ which satisfies $k/h^2 \leq \frac{1}{2}$ automatically

For Euler's method:

$$B(k) = I + kA$$

For Crank-Nicholson:

$$B(k) = \left(I - \frac{k}{2}A\right)^{-1}\left(I + \frac{k}{2}A\right)$$

To show convergence we need consistency (i.e. local truncation error $\to 0$ as $k \to 0$ and $h \to 0$) and some kind of stability:

Def (Lax-Richtmyer) A linear method of the form $(**)$ is said to be Lax-Richtmyer stable if for all time T there is a constant $C_T > 0$ s.t.

$$\|B(k)^n\| \leq C_T$$

for all $k > 0$ and integers n s.t. $\underbrace{kn}_{\text{discrete time}} \leq T$

Theorem (Lax Equivalence Theorem) A consistent method $(**)$ is convergent iff it is Lax-Richtmyer stable

The idea is the same as stability for ODE:
apply numerical method to exact solution $u(x,t)$:

$$u^{n+1} = B u^n + b^n + k \underbrace{\tau^n}_{\text{local trunc. err.}}$$

where $u^n = \begin{bmatrix} u(x_1,t_n) \\ u(x_2,t_n) \\ \vdots \\ u(x_m,t_n) \end{bmatrix}$

$\tau^n = \begin{bmatrix} \tau(x_1,t_n) \\ \tau(x_2,t_n) \\ \vdots \\ \tau(x_m,t_n) \end{bmatrix}$ = local truncation error

subtracting the difference eq (our method):

$$U^{n+1} = BU^n + b^n$$

we get $E^{n+1} = BE^n - k\tau^n$, where $E^n = U^n - u^n$

hence after N time steps:

$$E^N = B^N E^0 - k \sum_{n=1}^{N} B^{N-n} \tau^{n-1}$$

$$\Rightarrow \quad \|E^N\| \le \|B^N\| \|E^0\| + k \sum_{n=1}^{N} \|B^{N-n}\| \, \|\tau^{n-1}\|$$

if method is L-R. stable then for $Nk \le T$:

$$\|E^N\| \le C_T \|E^0\| + T C_T \max_{1 \le n \le N} \|\tau^{n-1}\|$$

$\to 0$ as $k \to 0$ for $Nk \le T$.

since method is <u>consistent</u> we do have:

$\|\tau^n\| \to 0$ and we need good I.C. s.t. $\|E^0\| \to 0$ as $k \to 0$

Example: for heat eq:

$$B(k) = I + kA = \text{symm matrix.}$$

The eigenvalues of $B(k)$ are:

$$\lambda_p(B(k)) = 1 + k\,\lambda_p(A(k)) = 1 + \frac{2k}{h^2}\underbrace{(\cos(p\pi h) - 1)}_{-2 \le \cdot \le 0}$$

But we assumed that $\frac{k}{h^2} \le \frac{1}{2}$:

$$\underbrace{}_{-2 < \cdot \le 0}$$

$$\Rightarrow \quad |\lambda_p(B(k))| \le 1$$

$$\Rightarrow \quad \| B(k) \| \le 1$$

for Crank-Nicholson.

$$B(k) = (I - \tfrac{k}{2}A)^{-1}(I + \tfrac{k}{2}A)$$

$$\lambda_p(B(k)) = \frac{1 + k\lambda_p/2}{1 - k\lambda_p/2} \le 1$$

So C-N is stable for all n.