Finite Difference Methods for Elliptic PDEs

A. Donev, Courant

We will consider boundary value problems (BVPs), mostly linear ones:

$$\begin{cases} \mathcal{L}\, u(\vec{x}\,; \cdot) = f(\vec{x}\,; \cdot) \text{ in } \Omega \\ \mathcal{B}\, u(\vec{x} \in \partial\Omega\,; \cdot) = g(\vec{x} \in \partial\Omega, \cdot) \text{ on } \partial\Omega \end{cases}$$

where $\mathcal{L}$ and $\mathcal{B}$ are linear (differential) operators.

Note that domain Ω can be unbounded but this is usually harder

An important example in 1D are ②

Sturm - Liouville (SL) problems

$$\begin{cases} \mathcal{L}u = -\left(p(x)\,u'(x)\right)' + q(x)\,u(x) \\ p(x) > 0 \,, \quad q(x) > 0 \quad \text{on } [a,b] \end{cases}$$

with either periodic "boundary" conditions (PBCs), or BC's of the form:

$$\underbrace{\underbrace{\alpha\, u'(a)}_{\text{Neumann}} + \underbrace{\beta\, u(a)}_{\text{Dirichlet}}}_{\text{Mixed / Robin}} = \underbrace{\gamma(t)}_{\substack{\text{If non-zero} \\ \text{inhomogeneous}}}$$

On <u>paper</u>, it does not really make sense to write something like

$$u = \mathcal{L}^{-1} f$$

because $\mathcal{L}$ is an infinite dimensional operator and inverse is unclear (<u>no</u> Gaussian elimination)

Instead, we use <u>eigenfunctions and eigenvalues</u> of $\mathcal{L}$

Consider <u>homogeneous BCs</u> first:

$$\begin{cases} \mathcal{L}u = f \\ Bu = 0 \end{cases} \xrightarrow[\text{eigenproblem}]{} \begin{cases} \mathcal{L}u_k = \lambda_k u_k \\ Bu = 0 \end{cases}$$

(λ_k, u_k) are eigensolutions, $k = 0, 1, 2, 3, \ldots$

If $\mathcal{L}$ is Hermitian or self-adjoint (symmetric), $\boxed{\mathcal{L}^* = \mathcal{L}}$, as is the case for SL BVPs, then the eigenvalues are real, and eigenvectors/functions are enumerable and orthogonal and a complete basis (let's not be too technical here).

If $\mathcal{L}$ is symmetric positive definite (SPD), as it is for SL BVPs, then all $\lambda > 0$ ($\lambda \geq 0$ if SPsD)

BVP has unique solution if $\lambda \neq 0$

Since eigenspace is a complete basis for L_2, we can expand both solution & r.h.s. into a generalized Fourier Series: (5)

$$\begin{cases} u = \sum_n a_n u_n \\ f = \sum_n b_n u_n \end{cases}$$

$$\mathcal{L} u = \sum_n a_n (\mathcal{L} u_n) = \sum \lambda_n a_n u_n = f = \sum b_n u_n$$

$$\Rightarrow \boxed{a_n = \frac{b_n}{\lambda_n}}$$

If we have inhomogeneous BCs (6)

then we only need to find one

particular solution of

$$\boxed{\begin{cases} \mathcal{L}\bar{u} = 0 \\ B\bar{u} = g \end{cases}} \Rightarrow \boxed{u = \bar{u} + \underbrace{\sum_n a_n u_n}} = \underbrace{\bar{u} + \tilde{u}}_{\text{Superposition principle}}$$

where $\begin{cases} \mathcal{L}\tilde{u} = f \\ B\tilde{u} = 0 \end{cases}$

Another approach on <u>paper</u> is to use the <u>Green's function</u> for the <u>PDE</u>. $G(\vec{x};\vec{y})$

$$\begin{cases} \mathcal{L}\, G(\vec{x}) = \delta(\vec{y} \in \Omega) \\ \mathcal{B}\, G(\vec{x}) = 0 \end{cases} \Rightarrow$$

quadrature

$$\boxed{u(\vec{x} \in \Omega) = \bar{u} + \int_{y \in \partial\Omega} f(\vec{y})\, G(\vec{x}, \vec{y})\, d\vec{y}}$$

<u>The problem</u>: We cannot analytically compute eigenfunctions or Green's functions ☹

Poisson eq. n 1D is easy though :

(8)

$$\begin{cases} u''(x) = f(x) \\ \text{example: } \quad u(a) = u_1, \quad u(b) = u_2 \end{cases} \qquad \boxed{L = b - a}$$

First inhomogeneous :

$$\bar{u}''(x) = 0 \quad \Rightarrow \quad \bar{u}(x) = \alpha x + \beta \quad (\text{linear})$$

$$\bar{u}(a) = u_1, \ \bar{u}(b) = u_2 \quad \Rightarrow \quad \bar{u}(x) = \frac{u_2 - u_1}{b - a}(x-a) + u_1$$

Now homogeneous :

$$\begin{cases} u'' = f(x) \\ u(a) = u(b) = 0 \end{cases}$$

Eigenfunctions :

$$u_k'' = \lambda_k u_k$$

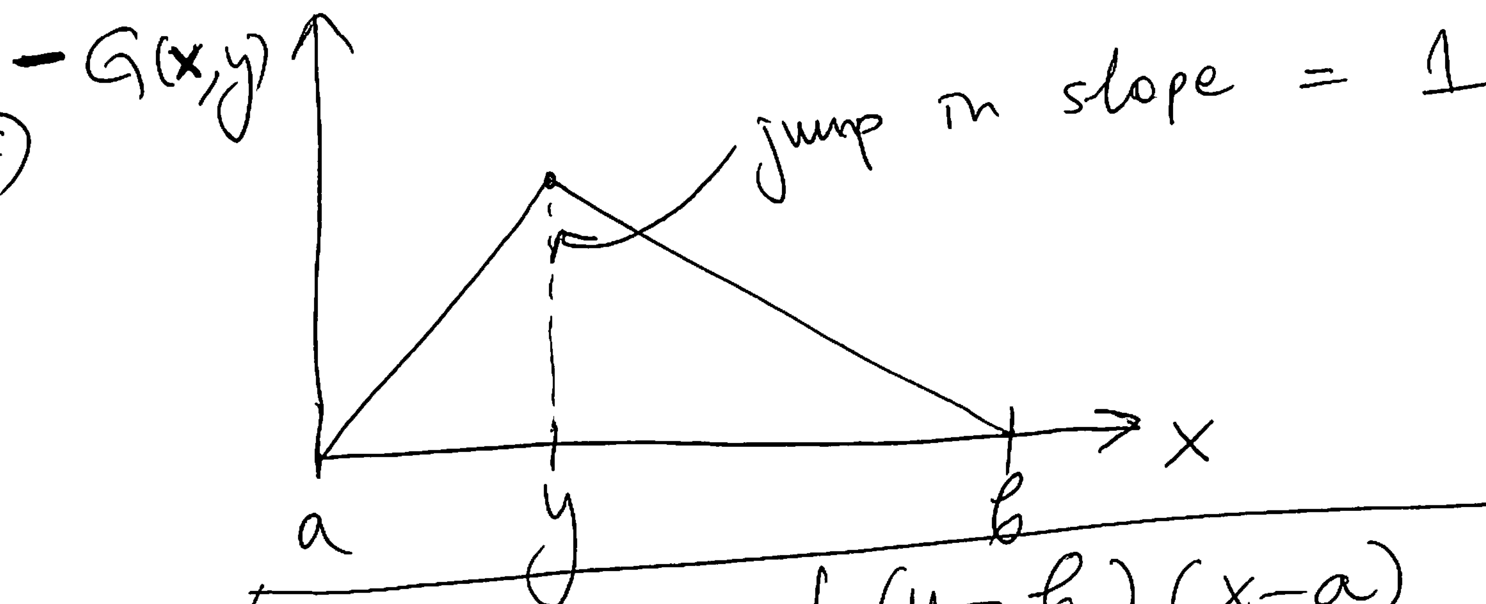
$$u_k(a) = u_k(b) = 0$$

$$\lambda_k = \left(\frac{2\pi k}{L}\right)^2 \qquad \Rightarrow \qquad \boxed{u_k = \sin\left(\frac{2\pi k}{L}(x-a)\right)}$$

Or, use Green's function

$$\begin{cases} u'' = \delta(y) \Rightarrow \\ u'(y^+) - u'(y^-) = 1 \\ u(a) = u(b) = 0 \end{cases}$$

$\Rightarrow$



$$\Rightarrow \boxed{G(x,y) = \frac{1}{b-a} \begin{cases} (y-b)(x-a), & a < x \le y \\ (y-a)(x-b), & y \le x \le b \end{cases}}$$

All of these pen-and-paper concepts will appear in our <u>analysis</u> of methods. (10)
However, numerical methods will not be based on <u>either</u> of these approaches, in general.
<u>Here is why</u>:

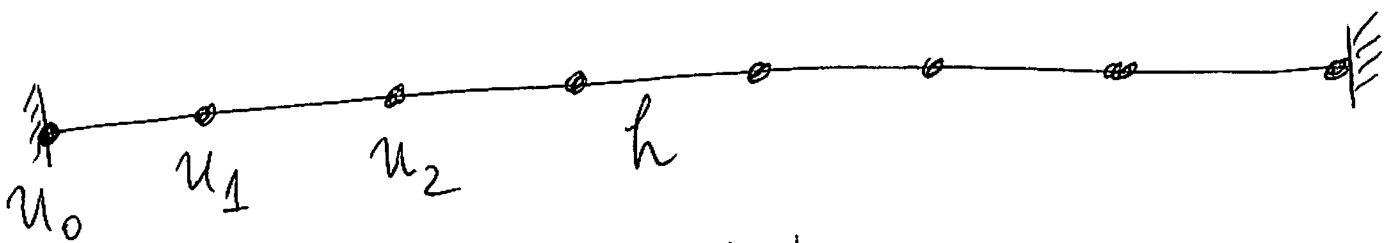
(1) Green's function is either hard to compute or is not smooth (δ function not good for numerics!)

(2) Computing eigenvalues of a matrix is more expensive than computing $u = \mathcal{L}^{-1} f$

→ $\boxed{\text{We will solve linear systems on computer}}$

Finite Difference Operators

In FD methods, we put down a grid of points with spacing h, and approximate derivatives using finite differences:



FD pointwise interpretation:

$$\boxed{u(x = kh) \approx u_k}$$

$u'(kh)$, $u''(kh)$ etc. approximated from pointwise values

$$u'(\bar{x}) \approx \begin{cases} (D_+ u)(\bar{x}) = \dfrac{u(\bar{x}+h) - u(x)}{h} + O(h) \end{cases}$$

one-sided difference OR

$$(D_- u)(\bar{x}) = \frac{u(\bar{x}) - u(\bar{x}-h)}{h} + O(h)$$

OR

two-sided centered approximation

$$(D_0 u)(\bar{x}) = \frac{u(\bar{x}+h) - u(\bar{x}-h)}{2h} + O(h^2)$$

OR

$$(D_3 u)(\bar{x}) = \frac{1}{6h}\left[2u(\bar{x}+h) + 3u(\bar{x}) - 6u(\bar{x}-h) + u(\bar{x}-2h)\right] + O(h^3)$$

OR

...

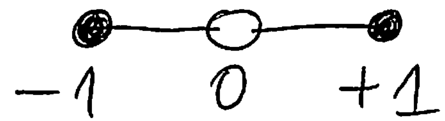
Each formula is characterized by its __stencil__, the set of neighboring points and their coefficients, e.g.

$$\left(D_0 \vec{u}\right)_j = \frac{u_{j+1} - u_{j-1}}{h}$$

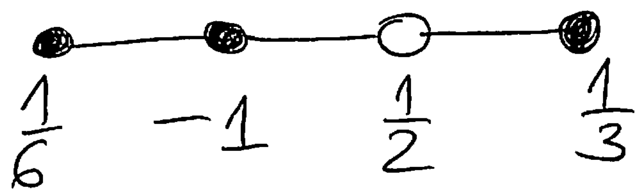
$$\Rightarrow \quad D_0 = \begin{bmatrix} \ddots & \ddots & \ddots & & & \\ & -1 & 0 & 1 & & \\ & & \ddots & \ddots & 1 & \\ & & & \ddots & 0 & \ddots \\ & & & -1 & \ddots & \ddots \end{bmatrix}$$

← __circulant__ or Toeplitz matrix far from boundaries or with PBCs

D_0 stencil is



D_3 stencil is



Truncation error can be computed easily using Taylor series

E.g.

$$(D_3 u)(\bar{x}) = u'(x) + \frac{h^3}{12} u^{(4)}(\bar{x}) + O(h^4)$$

One can most easily construct finite-difference formulas by fitting a polynomial through the points of the stencil and then differentiating that.

If the polynomial is of order p ($p+1$ points in stencil) then error $= O(h^{p-1})$

Second-order derivatives

<u>In 1D</u>, by far the simplest and most common is the <u>centered second order</u>

$$\boxed{D^2 = D_+ D_-}$$

$$D^2 u(\bar{x}) = \frac{u(\bar{x}-h) - 2u(\bar{x}) + u(\bar{x}+h)}{h^2}$$

$$= u''(\bar{x}) + \frac{h^2}{12} u^{(3)}(\bar{x}) + O(h^4)$$

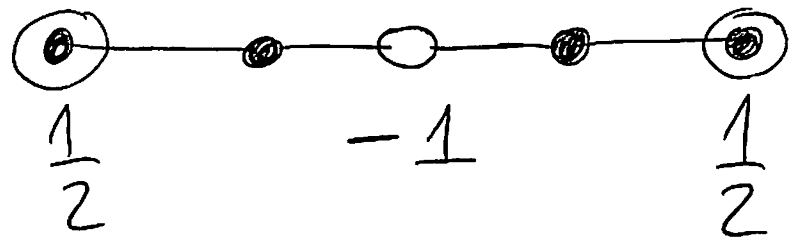
No $O(h^3)$ because of symmetry – only even powers

$$\frac{1}{h^2} \quad \bullet\!-\!\!-\!\!-\!\circ\!-\!\!-\!\!-\!\bullet \qquad \begin{matrix} 1 & -2 & 1 \end{matrix}$$

$$D^2 = \begin{bmatrix} -2 & 1 & & \\ 1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix}$$ as a matrix

Observe how this is different and **better** than

$$\tilde{D}^2 = D_0^2 \equiv \frac{1}{h^2}$$



In this wide stencil, odd and even points on the grid are uncoupled:
We can change all even values and not change the result for odd points.

$\tilde{D}^2$ has a non-trivial null space of checkerboard solutions!

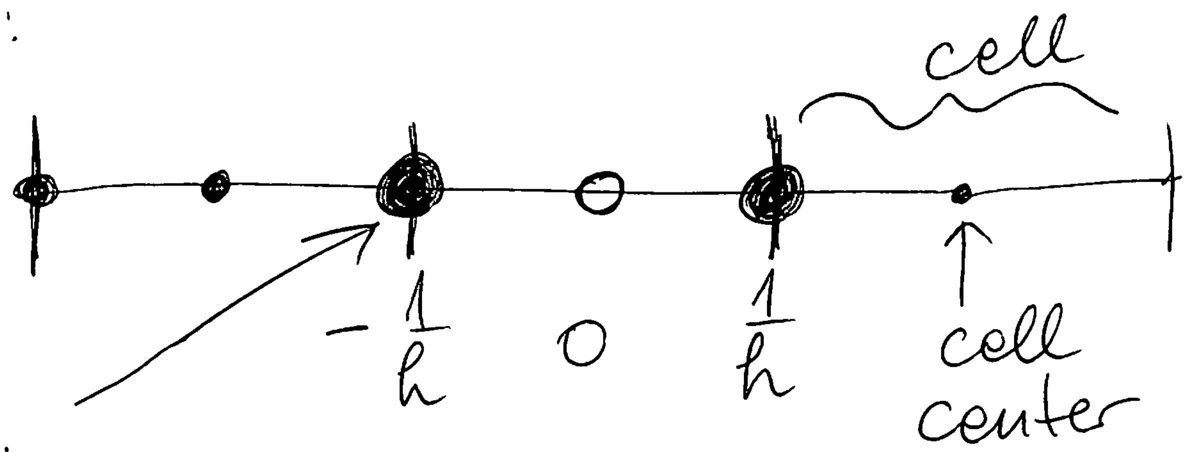
$$\boxed{\text{Accuracy is } \underline{\text{not}} \text{ everything!}}$$

An alternative view of D^2:

Define a <u>staggered centered difference</u>

$$u'(\bar{x}) \simeq (\hat{D}_0 u)(\bar{x}) = \frac{1}{h}\left[u\left(x+\frac{h}{2}\right) - u\left(x-\frac{h}{2}\right)\right]$$

stencil:



node or face center or vertex depending on context.

<u>claim</u> (check)

$$\hat{D}_0^T = -\hat{D}_0$$

on an unbounded domain (formal)

More generally define :

$\hat{D}_0$: cell centers $\to$ nodes

$u^{(C)}$ $\qquad$ $u^{(F)}$

$\hat{D}_0^*$: nodes $\longrightarrow$ cell centers

$\underbrace{\hat{D}_0^*}_{\text{adjoint}}$

$$\left(\hat{D}_0 \vec{u}^{(C)}\right)_{j+\frac{1}{2}} = \frac{u^{(C)}_{j+1} - u^{(C)}_{j}}{h}$$

$$\left(\partial_x\right)^* = - \partial_x \quad \text{(integration by parts)}$$

$$\left(\hat{D}_0^* \vec{u}^{(F)}\right)_j = - \frac{u^{(F)}_{j+1/2} - u^{(F)}_{j-1/2}}{h}$$

Claim:

$$\boxed{D^2 = -\hat{D}_0 \hat{D}_0^*}$$

which is a discrete equivalent of

$$\nabla^2 u = \nabla \cdot (\nabla u)$$

$$\Delta = \text{div} \circ \text{grad}$$

This can be generalized to higher dimensions

$$\boxed{\underset{\text{Laplacian}}{L} = \underset{\text{divergence}}{D}\ \underset{\text{gradient}}{G} = -DD^*}$$

(including boundary conditions)

$$\left(-\hat{D}_0 \hat{D}_0^* u\right)_j = \left(D^2 u\right)_j = \frac{\left(\frac{u_{j+1}-u_j}{h}\right) - \left(\frac{u_j - u_{j-1}}{h}\right)}{h}$$

Immediate corollary:

1. $\boxed{D^2 \text{ is negative semidefinite}}$

(as a matrix)

just as the $\mathcal{L}$aplacian as an operator

2. $\boxed{D^2 \text{ has only constant fields as null-vectors}}$ $\vec{u}_j = \text{const}$

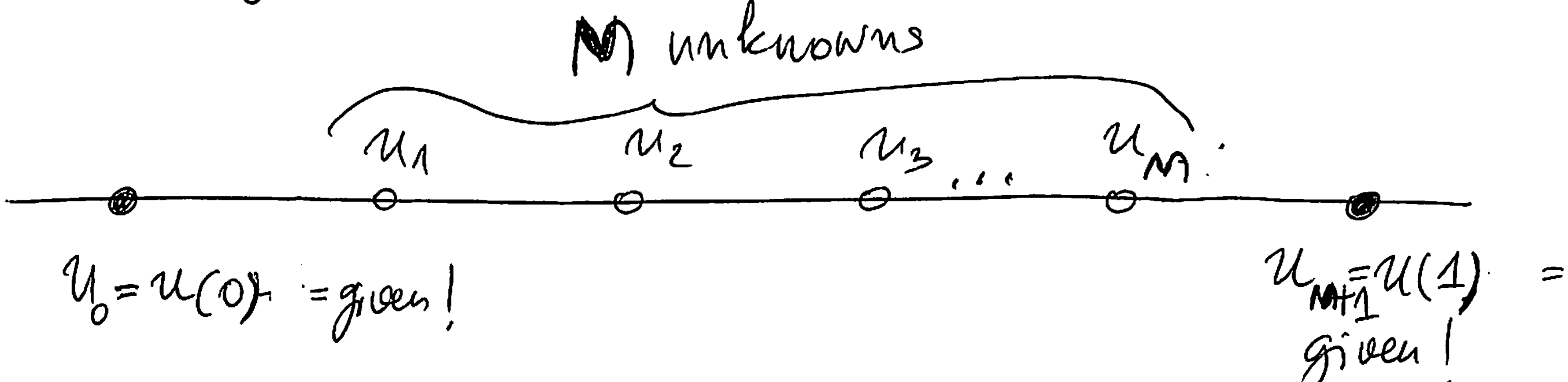
also like the continuum $\mathcal{L}$aplacian

DIY: at home

Write a finite difference discretization of $\mathcal{L} = \frac{d}{dx} c(x) \frac{d}{dx}$ where $c(x) > 0$ that you can prove is negative (semi) definite

What about boundary conditions?

The easiest is a Dirichlet BC with an FD grid <u>aligned</u> with the boundary:



u_0 and u_{M+1} are <u>not</u> real variables, they are <u>known</u> constants (or functions of time for parabolic PDEs)

E.g. $$\begin{cases} u''(x) = f(x) & x \in [0,1] \\ u(x=0) = \alpha \\ u(x=1) = \beta \end{cases}$$ (22)

Convert into:

$$\begin{cases} (D^2 u)_j = \frac{1}{h^2}(u_{j-1} - 2u_j + u_{j+1}) = f(x_j) \\ \qquad j = 1, \ldots, m \\ u_0 = \alpha \\ u_{m+1} = \beta \end{cases}$$

This has the form

$$\tilde{A}\,\underline{\underline{\vec{u}}} = \underline{\underline{\vec{f}}}$$

$$\boxed{\underline{\underline{\vec{u}}} = \begin{bmatrix} u_0 \\ \vdots \\ u_{m+1} \end{bmatrix}}$$

But here $\tilde{A}$ is not a matrix but rather an <u>affine</u> linear operator (23)
because u_0 and u_{m+1} are constants

$$\tilde{A}\,\underline{\underline{\vec{u}}} = \tilde{A}\begin{bmatrix} 0 \\ u_1 \\ \vdots \\ u_m \\ 0 \end{bmatrix} + \tilde{A}\begin{bmatrix} \alpha \\ \vdots \\ \beta \end{bmatrix}$$

$$= \tilde{A}\,\vec{u}_{int} + \tilde{A}\,\vec{u}_{BC}$$

(or ghost) <u>boundary</u>

<u>interior</u>

where now $\vec{u}_{int} = (0, u_1, \ldots, u_m, 0)$ so that $\tilde{A}\,\underline{\vec{u}}$ contains just the part from <u>homogeneous BCs</u>

We can instead consider

$$\vec{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} \quad \text{as the unknown}$$

and write

$$\boxed{A\vec{u} = \vec{f} - (\text{inhomogeneous part}) = \underset{\sim}{f}}$$

$$A = \begin{bmatrix} -2 & 1 & & \\ 1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix} \qquad \vec{\underset{\sim}{f}} = \begin{bmatrix} f_1 - \alpha/h^2 \\ f_2 \\ \vdots \\ f_{m-1} \\ f_m - \beta/h^2 \end{bmatrix}$$

<u>OR</u> $\quad \tilde{A}\,\underline{\underline{\vec{u}}} = \underline{\underline{\vec{f}}} + \begin{cases} u_0 = \alpha \\ u_{m+1} = \beta \end{cases}$ (extended)

Note that in many "real life" codes

one uses $\tilde{A}$ and not A, just so that the code can handle arbitrary BCs with little changes, and for efficiency (no *if* statements in loops).

The values u_0 and u_{m+1} are then called **ghost** or **virtual cells**.

E.g. useful for periodic BCs where ghost cells are *images* (copies) of interior values on the other side of the domain.

We have converted the BVP into a <u>linear system</u>

$$\boxed{A u = \tilde{f}}$$

the tricky part?

The size of A (dimension) grows as we add more points, Is there a clear limit as $m \to \infty$?

We will answer this next lecture.

For now let's discuss a bit other BCs

Periodic BCs:

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

← Symmetric (negative definite!) just as Dirichlet

A is a circulant matrix.

Observe A is diagonalized by the Fourier transform (why?):

$$A = F \Lambda F^*$$

F: (i) FFT

Λ: diagonal of eigenvalues.

F^*: (i) FFT

Excercise:

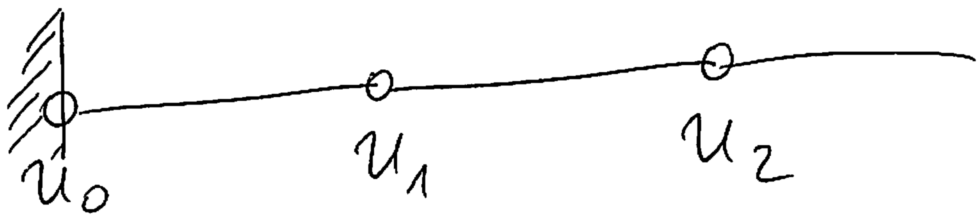
Compute Λ

(will appear in Homework X later)

How about **Neumann BCs**, e.g.

$$u'(0) = \sigma$$

Now u_0 is a **real variable**:



$$\frac{du}{dx} = \sigma$$

First idea (section 2.12 in Le Veque)

$$\frac{u_1 - u_0}{h} = \sigma$$

← use **one-sided difference** to implement BC (first order)

Instead of $u_1 = \alpha$

In implementation, we can either add u_0 as a variable and this extra equation, or, eliminate

$$u_0 = u_1 - h\sigma$$

as a ghost cell value. This really means we are extrapolating from the interior to the ghost cells using the BCs.

What error do we make?

Local truncation error (LTE)

$$\tau_0 = \frac{u(x_1) - u(x_0)}{h} - \sigma = \frac{1}{2} h u''(x_0) + O(h^2)$$

So this is only first order accurate at the boundary. Will this ruin the overall second-order <u>global</u> accuracy? We will figure that out next time.

<u>Second idea</u>

Use a <u>ghost point</u> u_{-1} and a second-order centered difference

$$\frac{1}{2h}\left(u_1 - u_{-1}\right) = \sigma \Rightarrow$$

$$\Rightarrow \quad u_{-1} = u_1 - 2h\sigma$$

Extrapolate linearly to the ghost cell!

Let's see if this works.

31

$$\frac{1}{h^2}\left(u_{-1} - 2u_0 + u_1\right) = f_0 = f(x_0)$$

$$u_{-1} = u_1 - 2h\sigma$$

Becomes

$$\boxed{\frac{1}{h}\left(u_1 - v_0\right) = \sigma + \frac{h}{2} f(x_0)}$$

Compare to Taylor series:

$$\frac{1}{h}\left(u(x_1) - u(x_0)\right) = \sigma + \frac{h}{2} u''(x_0)$$

$$u''(x_0) \equiv f(x_0) \text{ by PDE}$$

So now it is second-order!

Third idea

Use a second-order three-point one-sided difference

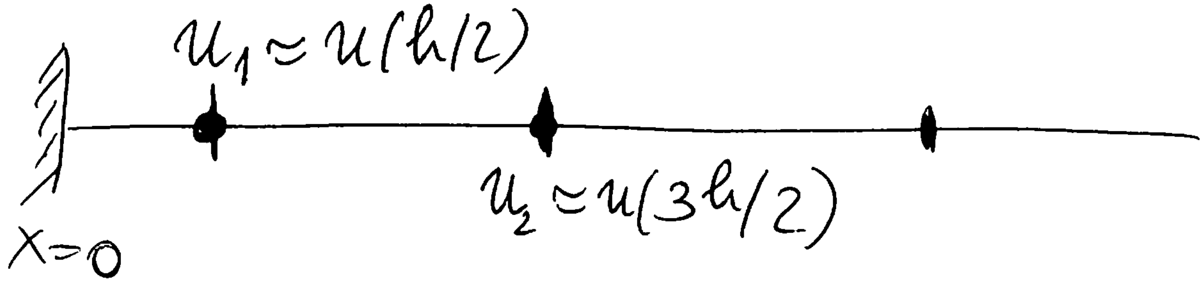
$$u'(x_0) \approx \frac{1}{h}\left(\frac{3}{2}u_0 - 2u_1 + \frac{u_2}{2}\right) = \sigma$$

Matrix becomes

$$\frac{1}{h^2}\begin{bmatrix} 3h/2 & -2h & h/2 & & \\ 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & \cdots & & \end{bmatrix} \leftarrow \text{NOT symmetric!}$$

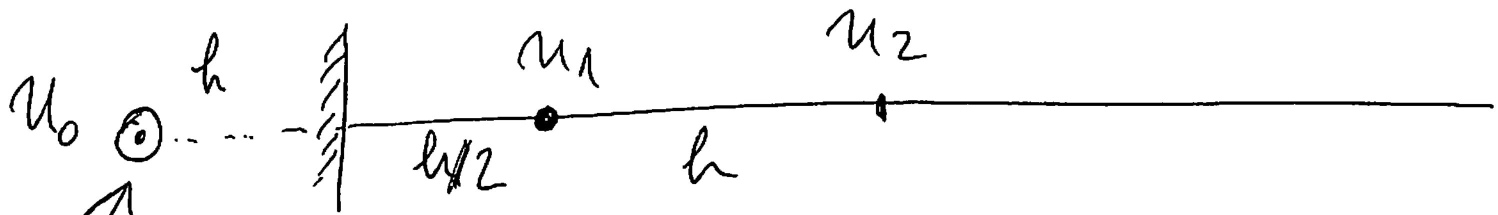
Which method is "best"? Depends...
Neither matrix symmetric!?!

Note that sometimes even for Dirichlet BCs we may need extrapolation / ghosts or one-sided, because the grid may be <u>staggered</u> by $h/2$ from the boundary



$u(x=0) = \alpha$

<u>Examples</u>: Finite volume methods for mixed parabolic-hyperbolic laws, (Navier-) Stokes with staggered pressure / velocity grids, magnetic fields and charges, etc.



$u(x=0) = \alpha$

ghost cell $\quad u_0 \approx u(x=-h/2)$ (makes no sense, but...)

B.C.: $\quad \dfrac{u_1+u_0}{2} \approx \alpha = u(x=0)$

$$u_0 = 2\alpha - u_1$$

Now

$$\frac{1}{h^2}(u_0 - 2u_1 + u_2) = f(h/2)$$

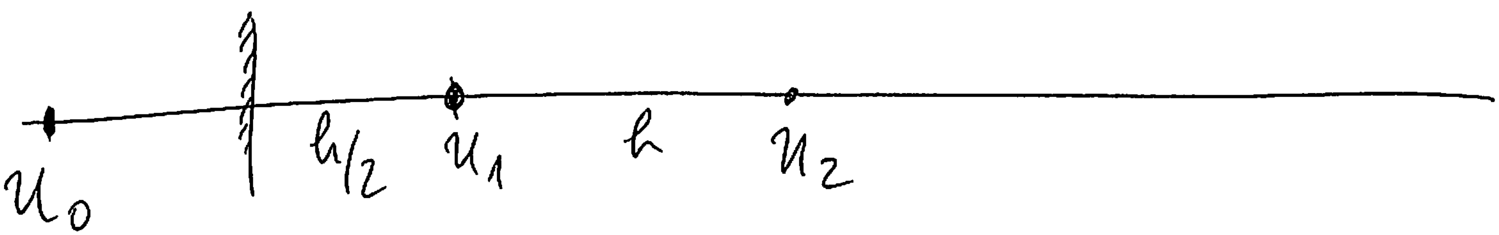
$$\boxed{\frac{1}{h^2}(-3u_1 + u_2) = f_{1/2} + \frac{2\alpha}{h^2}}$$

Now the matrix is

$$\frac{1}{h^2}\begin{bmatrix} -3 & 1 & & & \\ 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & & \ddots & \end{bmatrix}$$

so it is <u>symmetric</u> (and in fact <u>negative definite</u>!)

If we had <u>Neumann with staggered</u>



$$\frac{u_1 - u_0}{h} \approx u'(0) = \beta$$

$$\Rightarrow \quad u_0 = u_1 - \beta h$$

$$\frac{1}{h^2}(u_0 - 2u_1 + u_2) = \frac{1}{h^2}(-u_1 + u_2) - \frac{\beta}{h} = f_{1/2}$$

ALSO <u>SYMMETRIC</u>

Of course, we can use <u>higher-order</u> differences, e.g.

$$\frac{1}{12h^2}\left[-u_{j-2}+16u_{j-1}-30u_j+16u_{j+1}-u_{j+2}\right]$$
$$= u''(jh)+O(h^4) = f(jh)$$

Matrix is $1/12h^2$ *

$$\begin{bmatrix}
-30 & 16 & -1 & & & & \cdots & 1 & 16 \\
16 & -30 & 16 & -1 & & & \vdots & 0 & 1 \\
-1 & 16 & -30 & 16 & -1 & & & & \\
0 & -1 & 16 & -30 & 16 & 1 & & & \\
 & & & & & \ddots & & &
\end{bmatrix}$$

LESS SPARSE!

Periodic BCs

Is this NEGATIVE SEMIDEFINITE? TRY to prove it

Will numerical solution satisfy MAXIMUM PRINCIPLE?

Addendum to <u>FD for 1D BVPs</u>

I asked in class for you to think about how you would discretize using finite differences the BVP

$$\left(k(x)\, u'(x) \right)' = f(x)$$

which models heat conduction in a non-uniform rod.

The <u>wrong</u> thing to do (go back to lecture on <u>spectral methods</u>) is to use the chain rule

$$\cancel{k u'' + k' u' = f(x)} \quad !$$

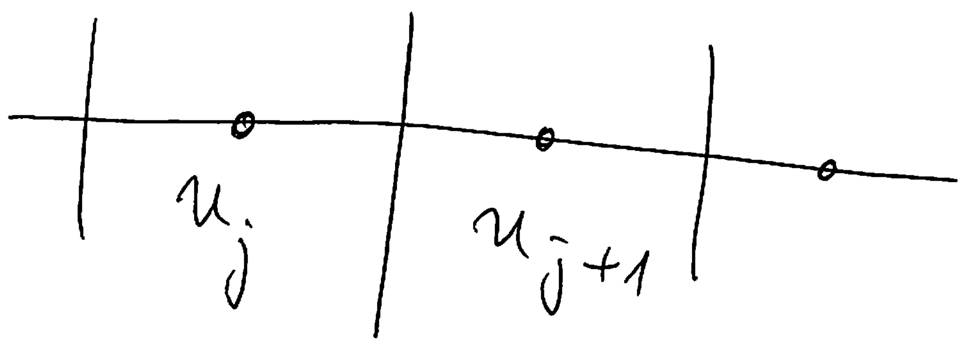
<u>wrong</u>

Instead, use physics

$$(\kappa u')' = (\varphi)'$$

where $\varphi = \kappa u' =$ flux

Put flux on a staggered grid (related to finite volume methods)



$$\varphi_{j+1/2} = \kappa_{j+1/2} \frac{u_{j+1} - u_j}{h}$$

whee $\kappa_{j+1/2} = \kappa(x_{j+1/2})$

Now

$$(\varphi')_j = \frac{\varphi_{j+1/2} - \varphi_{j-1/2}}{h}$$

$$\Rightarrow (Ku')'_i = \frac{1}{h}\left[K_{i+1/2}\left(\frac{u_{i+1}-u_i}{h}\right) - K_{i-1/2}\left(\frac{u_i - u_{i-1}}{h}\right)\right]$$

$$(Ku')' = Au \quad \text{where}$$

A is a discretization of the operator

$$\mathcal{L} = \frac{d}{dx} K(x) \frac{d}{dx} \quad \text{where} \quad K(x) > 0$$

$\mathcal{L}$ is a negative semidefinite operator with constant fields in its null space (depends on BCs)

Claim:

(1) A is a symmetric and negative definite (for Dirichlet BCs)

How to prove it? (in class on board)

(2) The numerical solution satisfies a maximum principle, i.e, the extremal values are achieved on the boundary

For simple Poisson, observe | for $f(x)=0$

$$\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} = 0 \quad \text{implies}$$

the DISCRETE mean value property $u_i = \dfrac{u_{i} + u_{i+1}}{2}$

$$u_i = \frac{K_{i+1/2}\, u_{i+1} + K_{i-1/2}\, u_{i-}\ldots}{K_{i+1/2} + K_{i-1/2}} \qquad (41)$$

$$\Rightarrow \boxed{u_i = w_i u_{i-1} + (1-w_i) u_{i+1} \quad \text{where} \quad 0 < w_i < 1}$$

so indeed u_i cannot be larger than the larger of u_{i-1} and u_{i+1} (same for smaller), and so extremum is on the boundary.

LOOK AT PHYSICS TO INFORM GOOD DISCRETIZATIONS