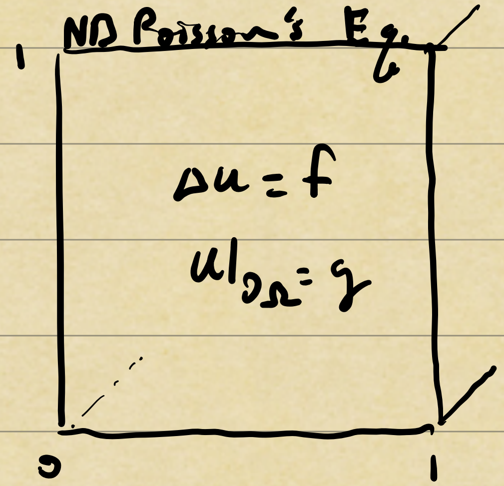
Beyond Poisson's Eq. (pt. 1)

ND Poisson's Eq.



Fast FD Solvers exploit:

=> FFT-based 1D solvers

=> Separable domain (Kronecker structure)

What about more complicated PDEs?

=> Variable Coefficients

=> Nonseparable Geometry

=> Time-dependence (parabolic, hyperbolic)

=> Nonlinearity

Key Idea 1: Many "more complicated PDE" solvers reduce problem to Boundary Value Problems, or at least to Linear Systems w/similar structure.

Key Idea 2: Many BVPs can be solved efficiently using preconditioned iterative methods.

Nonlinear Initial Value Problem

E.g. KdV Eq.

$$\underbrace{\partial_t u + \partial_x^3 u}_{\text{linear, time-dep.}} = \underbrace{-u\partial_x u}_{\text{nonlinear term}}$$

General first-order systems in time:

state

$$\mathcal{M}\partial_t X + \mathcal{L}X = \mathcal{F}(X)$$

linear diff. ops.   nonlinear diff. op.

Two useful perspectives:

$$\begin{tikzcd}
\underset{x,t}{\text{PDE}} \arrow[r] \arrow[d] \arrow[dr, bend left] & \underset{t}{\text{ODEs}} \arrow[d] \\
\underset{x}{\text{BVPs}} \arrow[r] & \text{Linear system}
\end{tikzcd}$$

1) "Propagator"

$$\mathcal{M}\left(\frac{X^{n+1} - X^n}{h}\right) + \mathcal{L}X^{n+1} = \mathcal{F}(X^n)$$

$\Rightarrow$ Solve BVPs

$$\left(\tfrac{1}{h}\mathcal{M} + \mathcal{L}\right)X^{n+1} = \tfrac{1}{h}\mathcal{M}X^n + \mathcal{F}(X^n)$$

2) "Method of Lines"

discretized ops   discretized state

$$M\partial_t X + LX = F(X)$$

$\Rightarrow$ Integrate system of coupled ODEs

* Need fast mat-vecs / solvers for $A = h^{-1}M + L$.

Iterative Methods for BVPs

1) Stationary Iterative Methods

Goal: Solve $Ax = b$ where $A = B + C$

Initial guess $x^{(0)}$
  for $n = 0, 1, 2, \ldots$
  $$Bx^{(n+1)} = Cx^{(n)} + b$$ (1 mat-vec with C, 1 linear solve w/ B)
end

<u>E.g.</u> $\mathcal{L} = -\Delta + V(x)$ on periodic $[0,1)$.

$$B = \begin{bmatrix} 2 & -1 & & & -1 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ -1 & & & -1 & 2 \end{bmatrix} \qquad C = \begin{bmatrix} \ddots & & \\ & \ddots & \\ & & \ddots \end{bmatrix}$$

variable coeff $C_{ii} = V(x_i)$

Fast Solve (FFT)   Fast multiply

<u>Convergence:</u> $e^{(n)} = x^{(n)} - x$ (error)

from iteration can write $\Rightarrow Be^{(n+1)} = Ce^{(n)}$

So, the error evolves as

$$e^{(n+1)} = \underbrace{B^{-1}C}_{D}\, e^{(n)} = D^{n+1} e^{(0)}$$

The asymptotic growth or decay of $\|D^n e^{(0)}\|$ is governed by spectral radius of D, i.e.

eigenvalues of D.

$$\rho(D) = \max_{1 \le j \le n} |\lambda_j|$$

If D has orthogonal eigenvectors, then

$$\|e^{(n)}\| = \|D^n e^{(0)}\| \le \rho(D)^n$$

and the iteration converges geometrically if

$$\rho(D) < 1$$

However, if the eigenvectors of D are far from orthonormal, the convergence rate is only asymptotic - there may be large transient growth of $\|e^{(n)}\|$ before convergence.

SPD A   Any A

2) Krylov methods (PCG, PGMRES, ...)

Iterative Approximations constructed from

$$\text{span}\{b, Ab, \ldots, A^{n-1}b\} = \text{Krylov Subspace}$$

↑ initial guess

Want (a) Fast $x \to Ax$ and (b) Rapid convergence

E.g. PCG Converges geometrically

$$\|e^{(n)}\|_A \le \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^n \|e^{(0)}\|_A \le 2e^{-2n/\sqrt{\kappa}} \|e^{(0)}\|$$

↑ error after n iterations

↖ "A-norm" $\sqrt{x^T A x}$ weighted Euclidean norm

Rate of geometric convergence slows when

$$\kappa = \frac{\lambda_{max}}{\lambda_{min}} = \text{ratio of } \frac{\text{largest}}{\text{smallest}} \text{ eigs}$$

is very large $\Rightarrow$ A is ill-conditioned.

Both Stationary & Krylov methods can be

<u>preconditioned:</u>

$$\underbrace{P_L A P_R}_{\tilde{A}} \underbrace{P_R^{-1} x}_{\tilde{x}} = \underbrace{P_L b}_{\tilde{b}}$$

Apply iterative method to preconditioned system - idea is to improve convergence rate by clever selection of P_L and P_R. (E.g. try to reduce $\rho(A)$ or $\kappa(A)$.)

<u>E.g.)</u> Schrödinger Eqn. (time-independent)

$$\Delta u + V(x)\, u = f$$

} discretize (high-order finite-diff)

$$\underbrace{(B + C)}_{A}\, x = b$$

$P_L = B^{-1}$ can be applied efficiently and $\quad P_L A = I + B^{-1}C$ has a much

smaller condition number κ than A does. Each iteration involves

(a) $\quad x \to (B+C)x \qquad$ (fast mat-vec)

(b) $\quad x \to B^{-1}x \qquad$ (fast solve)

And only need $\mathcal{O}(\log(\varepsilon))$ iterations for an $\varepsilon > 0$ accurate approximate solution.