The diffusion equation in 2D

$$u_t = u_{xx} + u_{yy}$$

$$u(x,y,t)$$

$$u(x,y,0) = \eta(x,y)$$

$$a \le x \le b$$
$$a \le y \le b$$

$$u(a,y,t) = g_1(y,t)$$
$$u(b,y,t) = g_2(y,t)$$
$$u(x,a,t) = g_3(x,t)$$
$$u(x,b,t) = g_4(x,t)$$

$$U'(t) = \nabla_h^2 U$$

$$\partial_{xx} = \frac{U_{i+1,j} - 2U_{ij} + U_{i-1,j}}{h^2} = (D_x^2 U)_{ij}$$

$$\partial_{yy} = \frac{U_{i,j+1} - 2U_{ij} + U_{i,j-1}}{h^2} = (D_y^2 U)_{ij}$$

$$\nabla_h^2 = D_x^2 + D_y^2$$

$$U_{ij}'(t) = (D_x^2 U)_{ij} + (D_y^2 U)_{ij}$$

Apply trapezoidal method:

$$U^{n+1} = U^n + \frac{k}{2}\left[\nabla_h^2 U^n + \nabla_h^2 U^{n+1}\right]$$

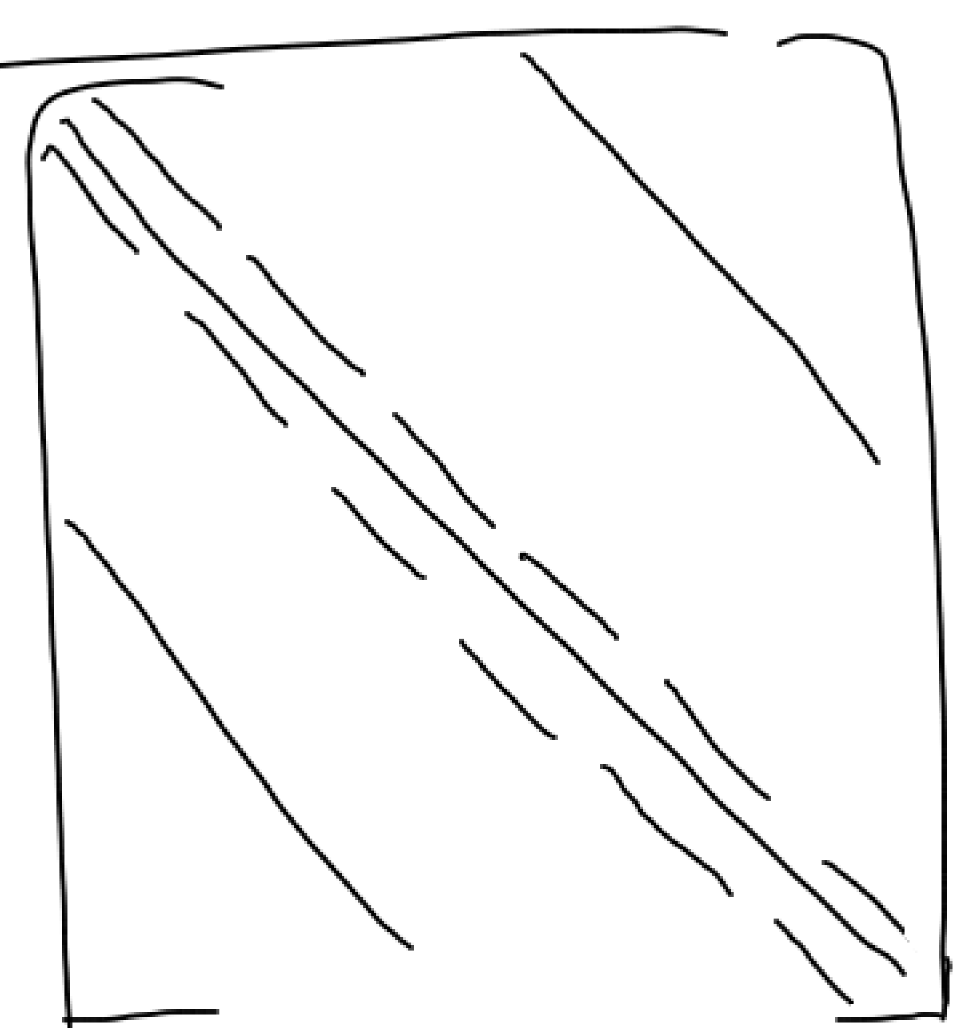
Crank-Nicolson method

$$\underbrace{\left(I-\frac{k}{2}\nabla_h^2\right)}_{M}\underbrace{U^{n+1}}=\underbrace{\left(I+\frac{k}{2}\nabla_h^2\right)U^n}$$

Must solve this system at every time step!

Direct or iterative?

Advantage of Direct: Factorize once, then solve cheaply for many RHS.

$M =$ 

If we have a $m \times m$ grid, M is $m^2 \times m^2$

Only $\sim 5m^2$ non-zeros

Direct solvers destroy the sparsity.

Iterative solvers take advantage of sparsity.

The # of iterations required depends on condition number of the matrix.

$$K(M) = \frac{\lambda_{max}}{\lambda_{min}}$$

Eigenvalues of ∇_h^2:

$$\lambda_{p,q} = \frac{2}{h^2}\left[(\cos(p\pi h)-1)+(\cos(q\pi h)-1)\right]$$

Eigenvalues of M:

$$1-\frac{k}{2}\lambda_{p,q} = O\left(\frac{k}{h^2}\right)=O\left(\frac{1}{h}\right)$$

(Since we choose $k \approx h$)

Largest eigenvalue of M: $\Theta\left(\frac{1}{h}\right)$

Smallest " " ": $\Theta(1)$

$K(M) = \Theta\left(\frac{1}{h}\right)$

This much better than $K(\nabla_h^2) = \Theta\left(\frac{1}{h^2}\right)$

Iterative solvers will converge very quickly.

- Often in 1 or 2 iterations

Note also: we have a great initial guess (U^n)

Locally one-dimensional (LOD) method
(Dimensional splitting)

$$U'(t) = (\partial_x^2 + \partial_y^2)U(t) \qquad U(0) = \eta$$

$$U(t) = e^{t(\partial_x^2+\partial_y^2)}\eta$$

$$U(t+k) = e^{k(\partial_x^2+\partial_y^2)}U(t)$$

$$\boxed{U(t+k) = U(t) + k(\partial_x^2+\partial_y^2)U(t) + \frac{k^2}{2}(\partial_x^2+\partial_y^2)^2 U(t) + O(k^3)}$$

$$= \Big(\underline{I + k(\partial_x^2+\partial_y^2)} + \frac{k^2}{2}\big(\underline{\partial_x^4} + \underline{\partial_y^4} + \partial_x^2\partial_y^2 + \partial_y^2\partial_x^2\big) + O(k^3)\Big)U(t)$$

LOD says

First solve $\underline{U'(t) = \partial_x^2 U}$ over one time step.

Then solve $\underline{U'(t) = \partial_y^2 U}$ over one time step.

Repeat.

This very cheap since we only need to do 1D solves.

How large is the error from splitting, if we ignore other errors?

$$U(t+k) = e^{k\partial_y^2} e^{k\partial_x^2} U(t)$$

$$U(t+k) = \left(\underline{\underline{I}} + \underline{k\partial_y^2} + \underline{\frac{k^2}{2}\partial_y^4} + \mathcal{O}(k^3)\right)\left(\underline{\underline{I}} + \underline{k\partial_x^2} + \underline{\frac{k^2}{2}\partial_x^4} + \mathcal{O}(k^3)\right) U(t)$$

$$= \left(\underline{I + k\left(\partial_x^2 + \partial_y^2\right)} + k^2\left(\underline{\partial_y^2\partial_x^2} + \frac{1}{2}\partial_x^4 + \frac{1}{2}\partial_y^4\right) + \mathcal{O}(k^3)\right) U(t)$$

Since $\partial_x^2\partial_y^2 = \partial_y^2\partial_x^2$, this is consistent to $\mathcal{O}(k^2)$

So if we choose time and space discretizations that are $\geq$ 2nd-order accurate, the overall method will be 2nd-order accurate.

LOD method with CD in space, trapezoidal method in time:

① $\underline{U}^* = \left(I - \frac{k}{2}D_x^2\right)^{-1}\left(I + \frac{k}{2}D_x^2\right)\underline{U}^n$

② $U^{n+1} = \left(I - \frac{k}{2}D_y^2\right)^{-1}\left(I + \frac{k}{2}D_y^2\right)U^*$

$\left(U'(t) = D_x^2 U\right)$

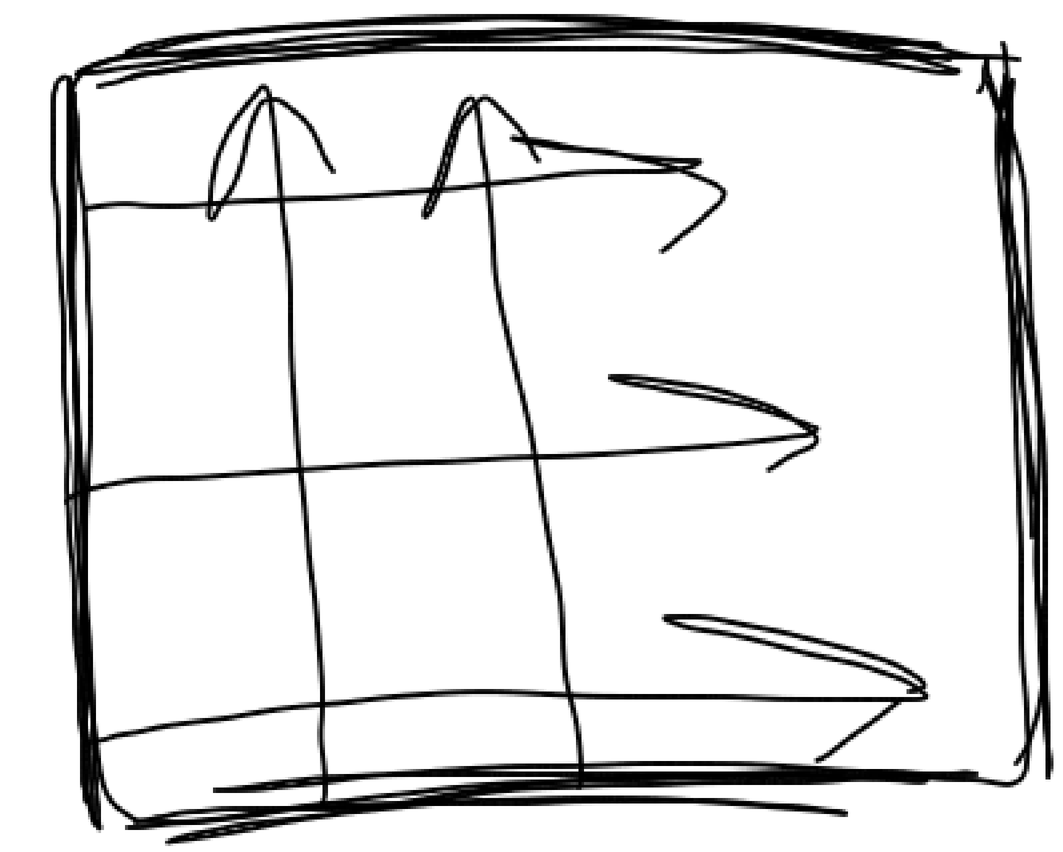
For ① we need BCs for U^n, U^* at left and right

For ② we need BCs for U^*, U^{n+1} at top and bottom.

For U^n, U^{n+1}: just evaluate BCs.

For U^*:

At top and bottom: impose BCs at t^n, and solve ① along boundaries



What about values at left and right for ①?

Impose BCs at t^{n+1}, then solve

$U^{n+1} = -\left(I - \frac{k}{2}D_y^2\right)^{-1}\left(I + \frac{k}{2}D_y^2\right)U^*$

$u_t = -u_{yy}$

Alternating Direction Implicit (ADI) method

$$U^* = U^n + \frac{K}{2}\left(D_y^2 U^n + D_x^2 U^*\right)$$

$$U^{n+1} = U^n + \frac{K}{2}\left(D_x^2 U^n + D_y^2 U^{n+1}\right)$$

Each step is implicit in one direction, explicit in the other.

Comparing this with the exact solution shows that the splitting error is $O(K^3)$, even if we replace D_x^2, D_y^2 with operators that don't commute.

Can be extended to parabolic problems with $\partial_x \partial_y$