The heat equation in 2D

$u(x,y,t)$ $\quad (x,y)\in[0,1]^2$

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

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

$$u(0,y,t)=g_0(y,t)$$
$$u(1,y,t)=g_1(y,t)$$
$$u(x,0,t)=f_0(x,t)$$
$$u(x,1,t)=f_1(x,t)$$

$x_i = ih$ $\quad i,j=0,1,\dots,m+1$

$y_j = jh$ $\quad h=\frac{1}{m+1}$

$$U_{ij}^n \approx u(x_i,y_j,t_n)$$

$$u_{xx}(x_i,y_j,t_n)\approx (D_x^2 U^n)_{ij} = \frac{U_{i+1,j}^n - 2U_{ij}^n + U_{i-1,j}^n}{h^2}$$

$$u_{yy}(x_i,y_j,t_n)\approx (D_y^2 U^n)_{ij} = \frac{U_{i,j+1}^n - 2U_{ij}^n + U_{i,j-1}^n}{h^2}$$

Semi-discrete: $U'(t) = D_x^2 U(t) + D_y^2 U(t)$

$$= \nabla_h^2 U(t)$$

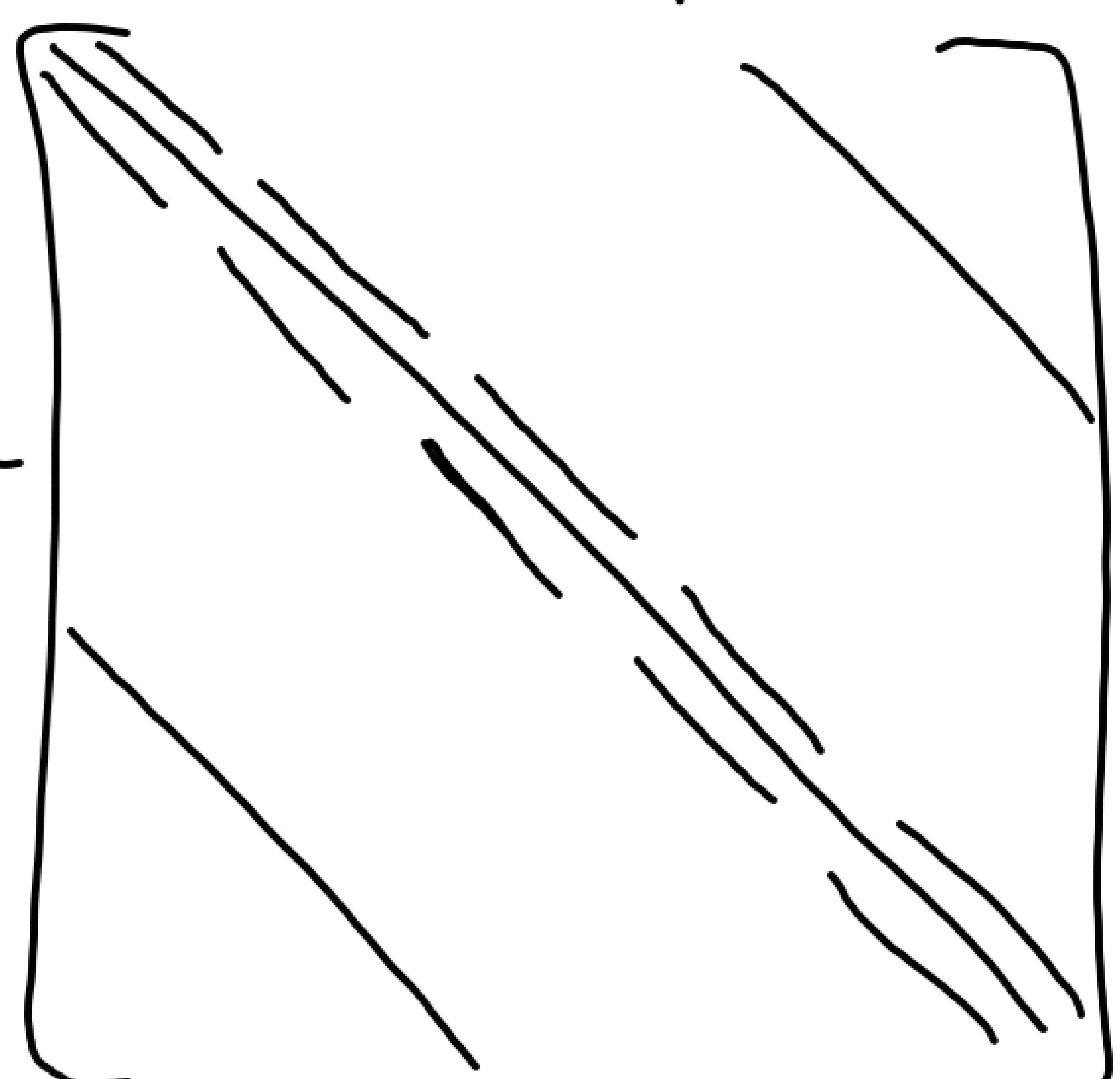
Implicit Trap. Method in time:

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

$$\left(I - \frac{K}{2}\nabla_h^2\right)U^{n+1} = \left(I + \frac{K}{2}\nabla_h^2\right)U^n$$

Must solve a sparse linear system at each time step.

$$M = I - \frac{K}{2}\nabla_h^2 =$$



$m^2 \times m^2$

$\mathcal{O}(m^4)$

Eigenvalues of M:

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

$$\max_{p,q} |\lambda_{p,q}| \approx 4\frac{K}{h^2} = \mathcal{O}\left(\frac{1}{h}\right)$$

$$\min_{p,q} |\lambda_{p,q}| \approx |1 + K\pi^2| \approx 1$$

$$\mathrm{Cond}(M) = \frac{\max|\lambda|}{\min|\lambda|} = \mathcal{O}\left(\frac{1}{h}\right)$$

We have a very good initial guess: U^n

(or we can use an explicit method as predictor)

So it's efficient to use an iterative solver for this algebraic system.

Dimensional Splitting (Locally one-dimensional)

Instead of solving

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

alternate between solving

$$u_t = u_{xx}$$

and $\quad u_t = u_{yy} \quad (*)$

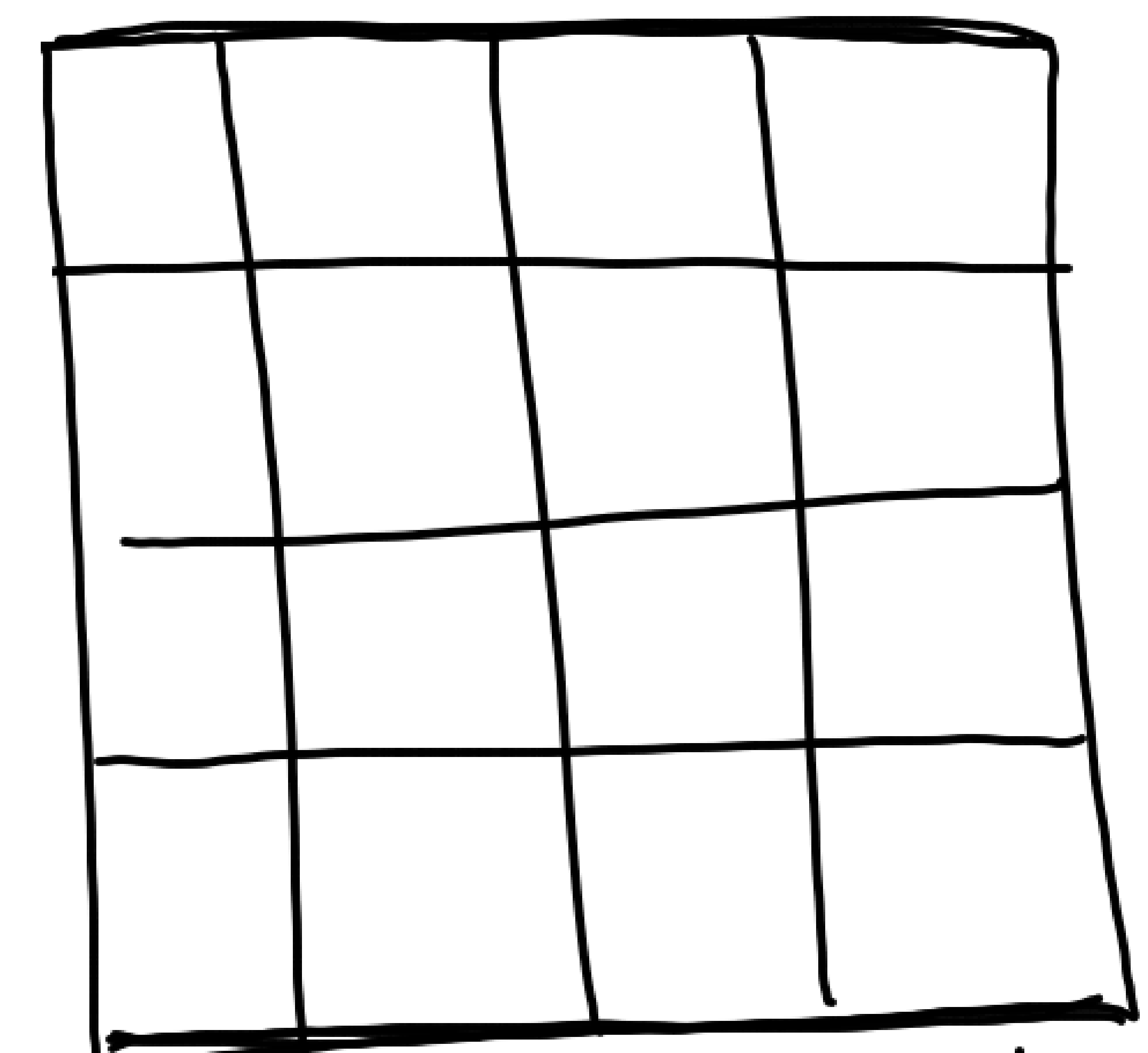
2nd-order accurate

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

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

Solve $2m$ systems of size $m \times m$ (tridiagonal) $\longrightarrow \mathcal{O}(m^2)$ work (optimal)

Boundary conditions



To get BCs for U^* in (2), take boundary at top/bottom at t_n and solve $u_t = u_{xx}$

To get BCs for U^* in (1), take BCs at left/right at t_{n+1}, and solve $(*)$ backward

Alternating Direction Implicit (ADI)

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

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

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

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

Again we only need to solve $2m$ tridiagonal systems of size $m \times m$.

Boundary conditions:
Just evalute at $t_n + \frac{K}{2}$.

$\Rightarrow$ 2nd-order accuracy.

Stable for $K = O(h)$