From Last Tuesday

$$D^2 = \frac{4}{h^2}\sum_{j=0}^{\infty}\sum_{\ell=0}^{\infty}\frac{1}{2j+1}\binom{-1/2}{j}\frac{1}{2\ell+1}\binom{-1/2}{j}\left(\tfrac{1}{2}\Delta_0\right)^{2j+2\ell+2}$$

From last Thursday replace ∞ by 0 to obtain...

$$D^2 = \frac{4}{h^2}\left(\tfrac{1}{2}\Delta_0\right)^2 = \frac{1}{h^2}\Delta_0^2 + \mathcal{O}(h^2).$$

use same formula

$$D^2 u(x_k) \approx \frac{1}{h^2}\Delta_0^2 u_k = \frac{1}{h^2}\left(u_{k+1} - 2u_k + u_{k-1}\right)$$

where $u(x_k) \approx u_k$

Poisson Equation 8.2

$$\nabla^2 u = f \quad \text{for } (x,y) \in \Omega$$

domain:
- open set
- connected
- bounded
- piecewise smooth boundary.

or

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

Boundary condition (Dirichlet)

$$u(x,y) = \phi(x,y) \quad \text{for } (x,y) \in \partial\Omega$$

$\partial\Omega$ boundary of Ω

Ω open means the set doesn't contain its boundary.

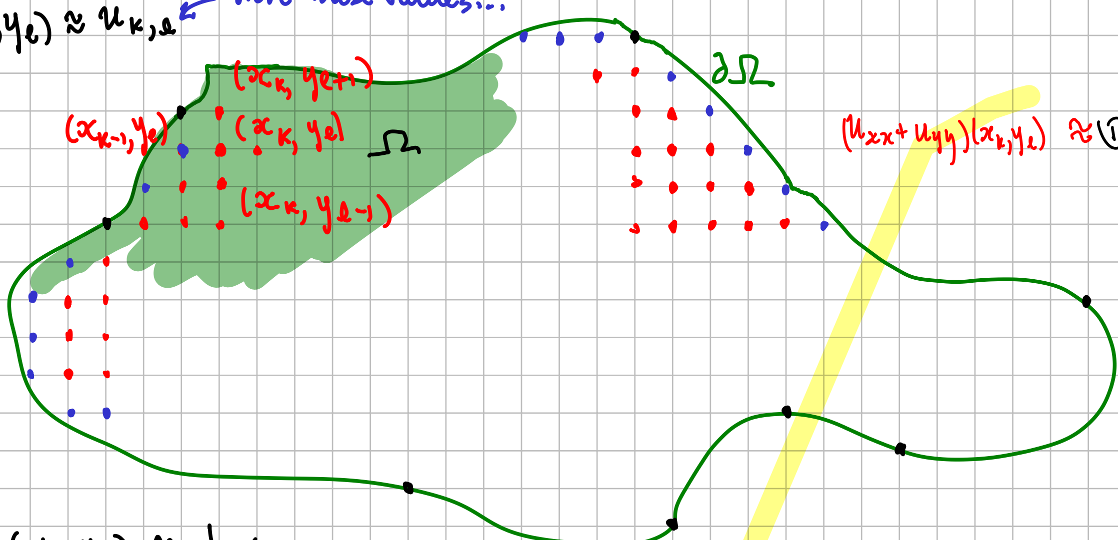
Discretize the solution on a grid...

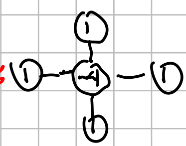
$x_k = x_0 + kh$

$y_\ell = y_0 + \ell h$

$\operatorname{cl}\Omega = \overline{\Omega} = \Omega \cup \partial\Omega$ closure of Ω

$u(x_k, y_\ell) \approx u_{k,\ell}$ ← store these values...

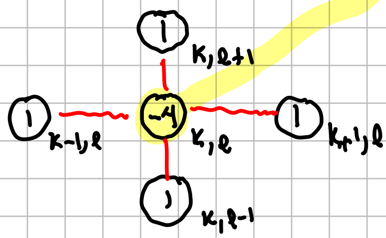


$(u_{xx} + u_{yy})(x_k, y_\ell) \approx$  $u_{k\ell}$

$$u_{xx}(x_k, y_\ell) \approx \frac{1}{h^2}\left(u_{k+1,\ell} - 2u_{k,\ell} + u_{k-1,\ell}\right)$$

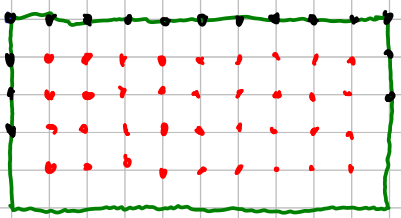
$$u_{yy}(x_k, y_\ell) \approx \frac{1}{h^2}\left(u_{k,\ell+1} - 2u_{k,\ell} + u_{k,\ell-1}\right)$$

$$(u_{xx} + u_{yy})(x_k, y_\ell) \approx \frac{1}{h^2}\left(u_{k+1,\ell} + u_{k,\ell+1} - 4u_{k,\ell} + u_{k-1,\ell} + u_{k,\ell-1}\right)$$



- On the boundary, use the boundary conditions.
- Inside use the stencil, if possible.
- What to do with the near boundary points?

Before worrying about the • near boundary points try a simpler geometry...



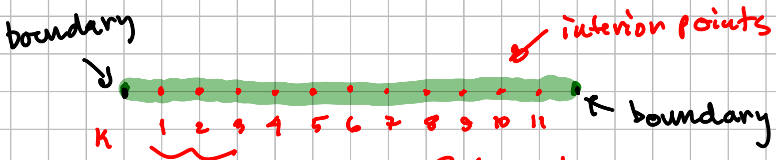
no near boundary points...

how to store the interior points as a vector to solve in a linear algebra problem?

store 36 points in a vector...

From the 1-dimensional case

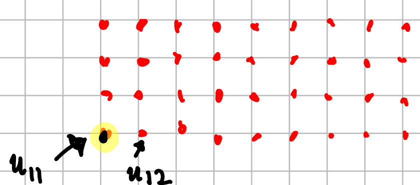
To discretize the differential equation divide the domain $[a, b]$ into $m+1$ equal pieces of size $h = (b-a)/(m+1)$. Consider the grid points $x_k = a+hk$ for $k = 1, \ldots, m$. Let y_k be approximation of the exact solution $y(x_k)$ at each grid point. Note the boundary conditions imply $y_0 = \alpha$ and $y_{m+1} = \beta$.



store the approximation in the vector y.

Solve $Ay = c$ to find y

In 2D we'll do this...

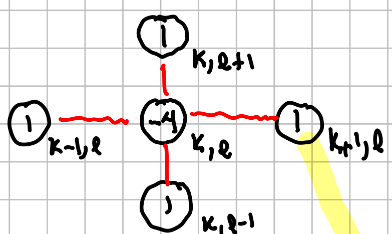


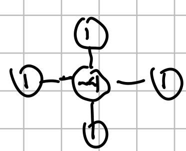
$u_{k,\ell} \to U[\ell + (k-1)*9]$

for $\ell = 1, \ldots, 9$

$k = 1, \ldots, 4$

What does A look like? $u_{xx} + u_{yy} = f$





$u_{k\ell} = h^2 f_{k,\ell}$ where $f_{k,\ell} = f(x_k, y_\ell)$

for $\ell = 1, \ldots, 9$
$k = 1, \ldots, 4$

How to efficiently solve the corresponding linear algebra problem..

$$\begin{bmatrix} -4 & 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ 1 & -4 & 1 & 0 & \cdots & 0 & 0 & 1 & \cdots \\ 0 & 0 & 1 & -4 & 1 & \cdots & 0 & 0 & 1 \\ \vdots & & \ddots & \ddots & \ddots & & & \ddots & 0 \\ 0 & & & \ddots & \ddots & \ddots & & & 1 \\ 1 & & & & \ddots & \ddots & \ddots & & \\ 0 & \ddots & & & & \ddots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 1 & -4 \end{bmatrix} \begin{array}{c} u_{k,l} \\ \begin{bmatrix} u_{11} \\ u_{12} \\ u_{13} \\ \vdots \\ u_{19} \\ u_{21} \\ u_{22} \\ \vdots \\ u_{29} \\ \vdots \\ u_{49} \end{bmatrix} \end{array} = \begin{bmatrix} h^2 f_{11} - g_{0,1} - g_{1,0} \\ h^2 f_{12} - g_{0,2} \\ h^2 f_{13} - g_{0,3} \\ \vdots \\ h^2 f_{19} - g_{0,9} - g_{1,10} \\ h^2 f_{21} - g_{2,0} \\ h^2 f_{22} \\ \vdots \\ h^2 f_{29} - g_{2,10} \\ \vdots \\ h^2 f_{49} - g_{5,9} - g_{4,10} \end{bmatrix}$$

Note: the matrix and RHS are missing some details. In particular the off-diagonal sequences of 1's have occasional 0's in them.

Questions

1. Is the matrix A in $Ay=c$ invertible... If so then there is a unique solution... (yes)

2. If $h \to 0$ does the approximation converge to the exact solution. yes How fast? $O(h^2)$

3. How best to efficiently do the numerics?
   - maybe use a sparse solver based Gaussian elimination that pays attention to the 0's.
   - Use an Iterative approximation method to approximate the solution to $Ay=c$.

Don't need to work to hard to solve $Ay=c$ because the exaction of this linear algebra problem is at best an approximation of the solution to the PDE.

Next time:

- what to do with the near boundary points.
- a computational demonstration.