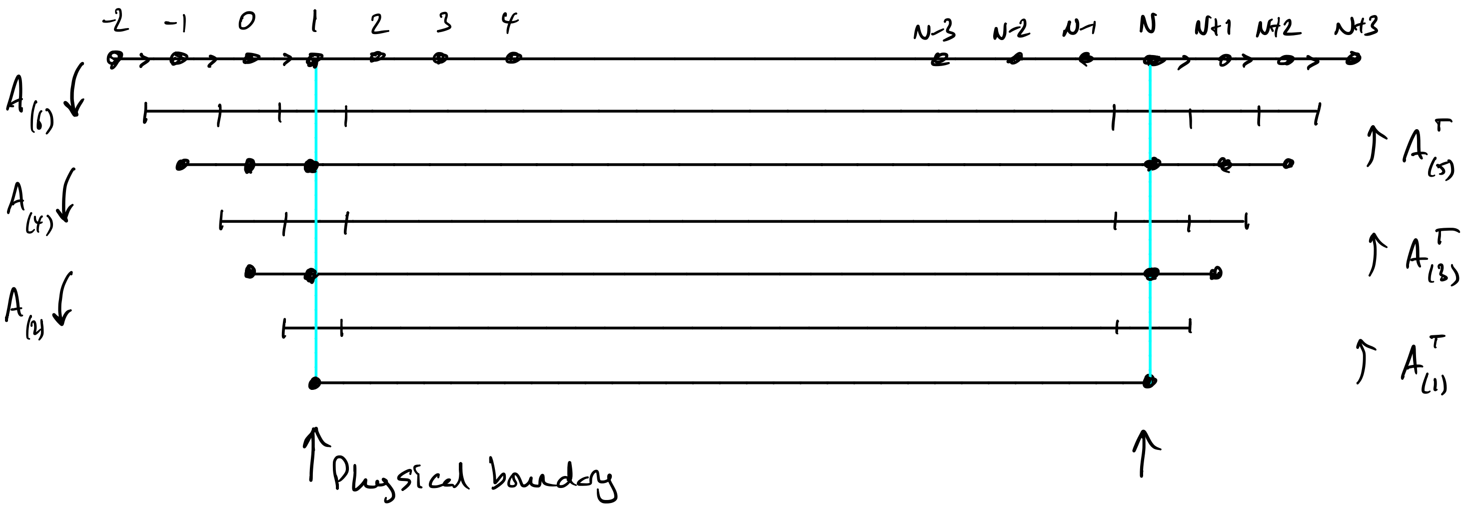
28.3.21

Updated 28.9.21

Differencing 6th order problem: 1D, uniform grid, symmetry boundary conditions.



Domain is discretised between x_1 and x_N, grid spacing Δ.
Three ghost points at either end of the domain

$$\psi_{1-j} = \psi_{1+j} \quad , \quad \psi_{N-j} = \psi_{N+j} \qquad j = 1, 2, 3$$

$\underset{\sim}{\psi}_{(6)}$ is the $N+6$ vector $(\psi_{-2}, \ldots, \psi_{N+3})^T$

$$A_{(k)} = \underbrace{\begin{pmatrix} -1 & 1 & 0 & 0 & \cdots \\ 0 & -1 & 1 & & \\ \vdots & & & & \\ \vdots & & & & \end{pmatrix}}_{N+k} \Big\updownarrow N+k-1 \qquad L_{(k)} = \frac{1}{\Delta} I_{(N+k)\times(N+k)}$$

$$I_{(N+2)\times N} = \underbrace{\begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & & \\ 0 & \cdot & \cdots & 1 & 0 \end{pmatrix}}_{N+2} \Big\updownarrow N$$

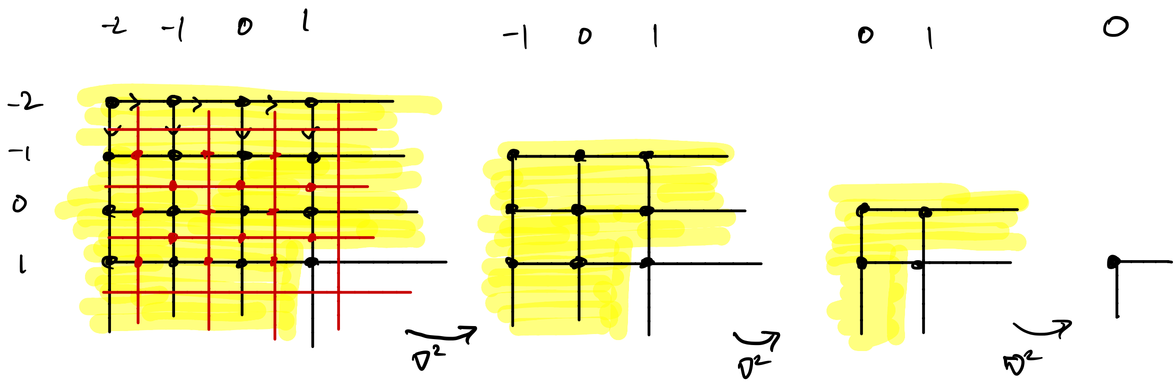
$$\underset{\sim}{L\psi}_{(4)} = \left[I_{(N+6)\times(N+4)} + L_{(4)} A^T_{(5)} L_{(5)} A_{(6)} \right] \underset{\sim}{\psi}_{(6)} \qquad L \equiv 1 + \nabla^2$$

$$\underset{\sim}{L^2\psi}_{(2)} = \left[I_{(N+4)\times(N+2)} + L_{(2)} A^T_{(3)} L_{(3)} A_{(4)} \right] \underset{\sim}{L\psi}_{(4)}$$

So
$$\frac{d}{dt} \underset{\sim}{\psi}_{(0)} = \nabla^2 \left(f(\psi) + L^2\psi \right)_{(0)} = L_{(0)} A^T_{(1)} L_{(1)} A_{(2)} \left[f(\psi)_{(2)} + \underset{\sim}{L^2\psi}_{(2)} \right]$$

In 2D

Yellow = ghost.
Start with $(N+6)\times(N+6)$
Ψ values



$L_{(5)} A_{(6)} \Psi_{(6)}$ approximates $\nabla\Psi_{(5)}$ at mid-points of primal edges (or vertices of dual network)

$L_{(4)} A_{(5)}^T L_{(5)} A_{(6)} \Psi_{(6)}$ defines $\nabla\cdot\nabla\Psi_{(4)}$ at vertices of primal network, on $(N+4)\times(N+4)$ points.

$L_{(2)} A_{(3)}^T L_{(3)} A_{(4)} [\uparrow]$ defines $\nabla^2(\nabla^2\Psi)_{(2)}$ on $(N+2)\times(N+2)$ points

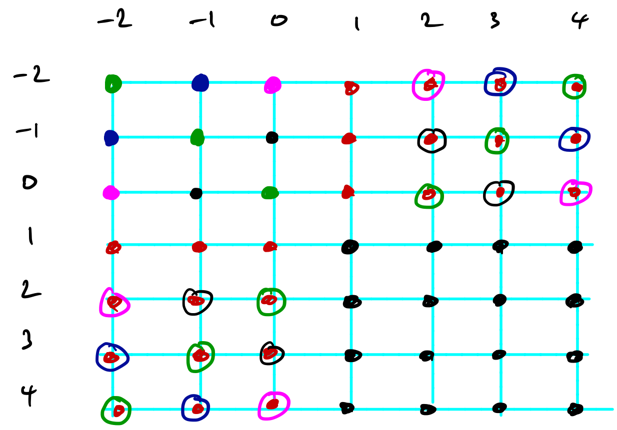
$L_{(0)} A_{(1)}^T L_{(1)} A_{(2)} [\uparrow]$ defines $\nabla^2(\nabla^2(\nabla^2\Psi))_{(0)}$ on $N\times N$ points

So original $(N+6)\times(N+6)$ grid has 3 ghost rows and 3 ghost columns around its periphery.

$$\left.\begin{array}{lll}
\Psi_{1-i,j} = \Psi_{1+i,j} & \text{for } i=1,2,3, & 1\le j\le N \\
\Psi_{N-i,j} = \Psi_{N+i,j} & \text{for } i=1,2,3 & 1\le j\le N \\
\Psi_{i,1-j} = \Psi_{i,1+j} & \text{for } j=1,2,3 & 1\le i\le N \\
\Psi_{i,N-j} = \Psi_{i,N+j} & \text{for } j=1,2,3 & 1\le i\le N
\end{array}\right\} \text{Red dots below}$$

We also need to specify the nine ghost points in each corner of the grid.

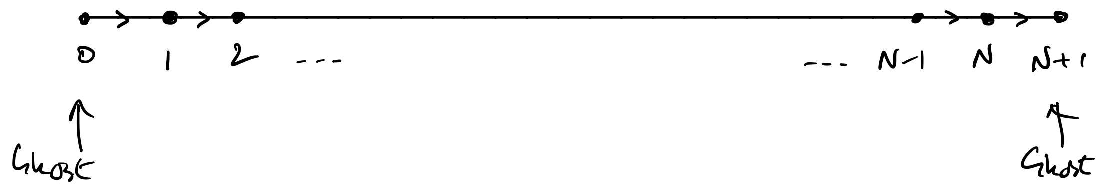
$$\Psi_{-2,-2} = \tfrac{1}{2}(\Psi_{-2,4} + \Psi_{4,-2})$$
$$\Psi_{-1,-1} = \tfrac{1}{2}(\Psi_{-1,3} + \Psi_{3,-1})$$
$$\Psi_{0,0} = \tfrac{1}{2}(\Psi_{0,2} + \Psi_{2,0})$$
$$\Psi_{-2,-1} = \theta\,\Psi_{-2,3} + (1-\theta)\,\Psi_{4,-1}$$
$$\Psi_{-1,-2} = \theta\,\Psi_{3,-2} + (1-\theta)\,\Psi_{-1,4}$$
$$\Psi_{-2,0} = \hat\theta\,\Psi_{-2,2} + (1-\hat\theta)\,\Psi_{4,0}$$
$$\Psi_{0,-2} = \hat\theta\,\Psi_{2,-2} + (1-\hat\theta)\,\Psi_{0,4}$$
$$\Psi_{-1,0} = \phi\,\Psi_{-1,2} + (1-\phi)\,\Psi_{3,0}$$
$$\Psi_{0,-1} = \phi\,\Psi_{2,-1} + (1-\phi)\,\Psi_{0,3}$$



$\theta = \hat\theta = \phi = \tfrac{1}{2}$ will work but other values might be needed to ensure second-order accuracy.

28-9-21

Example: Diffusion in 1D, no-flux boundary conditions



$$A_{(2)} = \begin{array}{c}\begin{pmatrix} -1 & 1 & 0 & \cdots \\ 0 & -1 & 1 & \cdots \\ \cdots & & & \\ 0 & 0 & -1 & 1 \end{pmatrix} \updownarrow N+1 \\ \longleftrightarrow \\ N+2 \end{array}$$

Maps from $N+2$ vertices to $N+1$ edges of primal network.

$$A_{(1)}^{(T)} = \begin{array}{c}\begin{pmatrix} -1 & 1 & 0 & \cdots \\ 0 & -1 & 1 & \cdots \\ \cdots & & & \\ 0 & 0 & \cdots & -1 & 1 \end{pmatrix} \updownarrow N \\ \longleftrightarrow \\ N+1 \end{array}$$

Maps from $N+1$ vertices (of dual network) to N edges (vertices of primal)

Then $$A_1^{(T)} A_2 = \begin{array}{c}\begin{pmatrix} -1 & 1 & 0 & \cdots \\ 0 & -1 & 1 & \cdots \\ \cdots & & & \\ 0 & 0 & \cdots & -1 & 1 \end{pmatrix} \\ \longleftrightarrow \\ N+1 \end{array} \begin{array}{c}\begin{pmatrix} -1 & 1 & 0 & \cdots \\ 0 & -1 & 1 & \cdots \\ & & & \\ 0 & 0 & \cdots & -1 & 1 \end{pmatrix} \updownarrow N+1 \\ \longleftrightarrow \\ N+2 \end{array}$$

$$= \begin{array}{c}\begin{pmatrix} 1 & -2 & 1 & 0 & \cdots \\ 0 & 1 & -2 & 1 & \cdots \\ \cdots & & & & \\ 0 & 0 & 1 & -2 & 1 \end{pmatrix} \updownarrow N \\ \longleftrightarrow \\ N+2 \end{array}$$

No-flux conditions at vertices 1 and N can be implemented with

$$\begin{pmatrix} -1 & 0 & 1 & 0 & \cdots \end{pmatrix} \begin{pmatrix} \psi_0 \\ \psi_1 \\ \psi_2 \\ \vdots \end{pmatrix} \equiv -\psi_0 + \psi_2 = 0$$

and

$$\begin{pmatrix} 0 & \cdots & & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \psi_0 \\ \vdots \\ \psi_{N+1} \end{pmatrix} \equiv -\psi_{N-1} + \psi_N = 0$$

Then $\psi_t = \psi_{xx}$ with $\psi_x = 0$ at $x = 0, L$ is represented either by

$$\begin{pmatrix} 0 \\ \psi_{1t} \\ \psi_{2t} \\ \vdots \\ \psi_{Nt} \\ 0 \end{pmatrix} = \underbrace{\begin{pmatrix} -1 & 0 & 1 & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & & \cdots & & \\ & & & 1 & -2 & 1 \\ & & & -1 & 0 & 1 \end{pmatrix}}_{N+2} \begin{pmatrix} \psi_0 \\ \psi_1 \\ \psi_2 \\ \vdots \\ \psi_N \\ \psi_{N+1} \end{pmatrix} \cdot \frac{1}{h^2} \qquad \text{where } h = \frac{L}{N}$$

or by

$$\begin{pmatrix} \psi_{1t} \\ \psi_{2t} \\ \vdots \\ \psi_{Nt} \end{pmatrix} = \underbrace{\begin{pmatrix} -2 & 2 & & & \\ 1 & -2 & 1 & & \\ & & \cdots & & \\ & & 1 & -2 & 1 \\ & & & 2 & -2 \end{pmatrix}}_{N} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_N \end{pmatrix} \cdot \frac{1}{h^2}$$

(incorporating the boundary conditions)