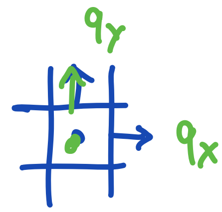
Lecture 12: Discretization in 2D

Logistics: - HW 5 is due Th 5/12

seems to be more trouble than expected

Last time: Started 2D discretization

- Matlab y-first ordering
- Staggered grid



- $\boxed{\underline{q} = \begin{bmatrix} q_x \\ q_y \end{bmatrix}}$ $\Rightarrow$ $\underline{\underline{G}} = \begin{bmatrix} \underline{\underline{G_x}} \\ \underline{\underline{G_y}} \end{bmatrix}$ $\underline{\underline{D}} = [\underline{\underline{D_x}} \; \underline{\underline{D_y}}]$

$$\underline{\underline{D_y}} = \begin{bmatrix} \underline{\underline{D_y}} & & & \\ & \underline{\underline{D_y}} & & \\ & & \underline{\underline{D_y}} & \\ & & & \underline{\underline{D_y}} \end{bmatrix} = \underline{\underline{I_x}} \otimes \underline{\underline{D_y}}$$

Kronecker

$$\underline{\underline{D_x}} = \frac{1}{\Delta x} \begin{bmatrix} \underline{\underline{I_y}} & -\underline{\underline{I_y}} & & \\ & \underline{\underline{I_y}} & -\underline{\underline{I_y}} & \\ & & \underline{\underline{I_y}} & -\underline{\underline{I_y}} \end{bmatrix} = \underline{\underline{D_x}} \otimes \underline{\underline{I_y}}$$

Matlab $\rightarrow$ kron()

Today: - Gradient 2D, Mean $\underline{\underline{M}}$

- Testing & convergence
- Code transition from 1D to 2D

Discrete gradient matrix in 2D

$$\underline{\underline{G}} = \begin{bmatrix} \underline{\underline{G_x}} \\ \underline{\underline{G_y}} \end{bmatrix}$$

Two options: 1) Adjoint relation between D & G

2) Tensor products

1

1) $\boxed{\underline{\underline{G}} = -\underline{\underline{D}}^T}$ in the interior

Impose natural BC's $\Rightarrow$ set $\underline{\underline{G}} = 0$ on all

~~bou~~ rows corresponding to bnd faces

2 dof_f_bnd = [dof_f_xmin; dof_f_xmax;
dof_f_ymin; dof_f_ymax];

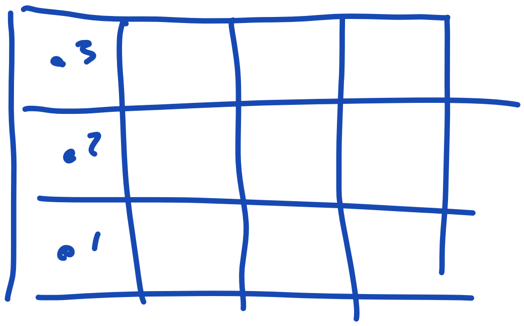
Zero them out

3 G(dof_f_bnd, :) = 0;

2) Kronecker product

$$\underline{\underline{G_y}} = \underline{\underline{I_x}} \otimes \underline{\underline{G_y}} \qquad \underline{\underline{G_x}} = \underline{\underline{G_x}} \otimes \underline{\underline{I_y}}$$

$\underline{\underline{I_x}}$ = speye(Nx, Nx) $\qquad$ $\underline{\underline{I_y}}$ = speye(Ny, Ny)



$N_y = 3 \qquad N_x = 4$

$\uparrow \qquad \uparrow$

$\underline{\underline{D_y}}^{1D} \qquad \underline{\underline{D_x}}^{1D}$

Two 1D ops

$D_y = \text{spdiags}\left(\frac{1}{\Delta y}[\underline{e}, -\underline{e}], [0,1], N_y, N_y+1\right)$

$D_x = \text{spdiags}\left(\frac{1}{\Delta x}[\underline{e}, -\underline{e}], [0,1], N_x, N_x+1\right)$

Mean operator in 2D

structure is same as $\underline{\underline{G}}$

$$\Rightarrow \quad \underline{\underline{M}} = \begin{bmatrix} \underline{\underline{M_x}} \\ \underline{\underline{M_y}} \end{bmatrix}$$

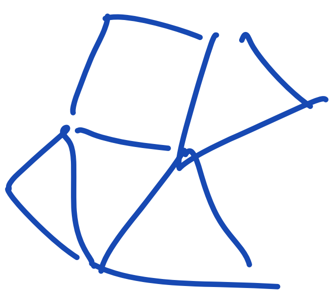
use Kronecker product to go from 1D $\rightarrow$ 2D

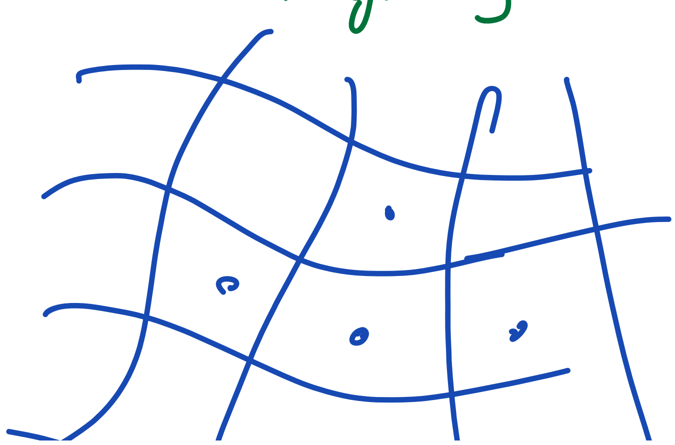
$$\boxed{\begin{aligned} \underline{\underline{M_x}} &= \underline{\underline{M_x}}^{1D} \otimes \underline{\underline{I_y}} \\ \underline{\underline{M_y}} &= \underline{\underline{I_x}} \otimes \underline{\underline{M_y}}^{1D} \end{aligned}} \quad \Rightarrow M = \begin{bmatrix} M_x \\ M_y \end{bmatrix}$$

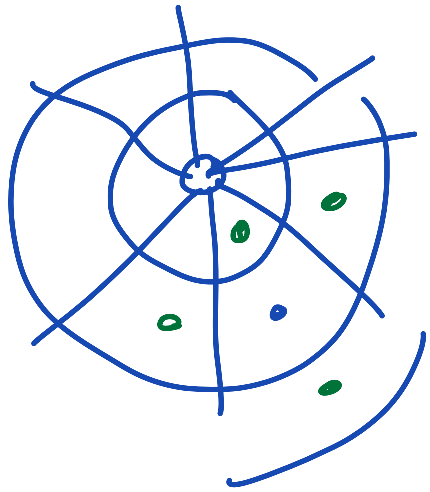
$\uparrow$

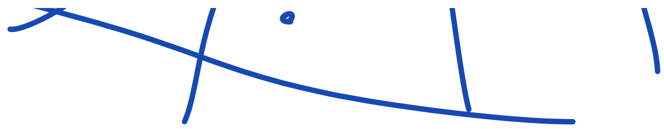
2D

Logically cartesian

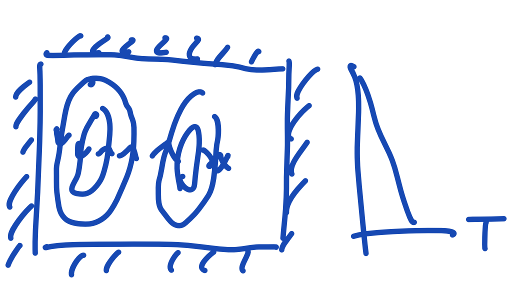








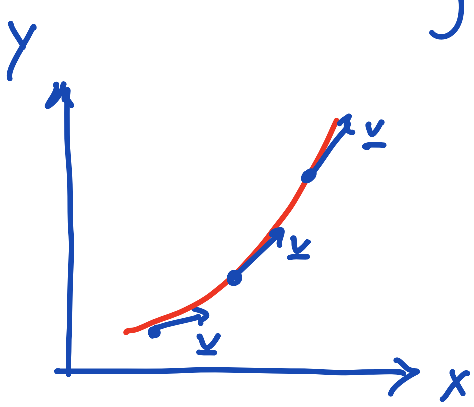
Streamlines & Streamfunction



Streamlines provide one of the best ways to illustrate flow fields.

Definition:

Streamlines are the family of curves that are instantaneously tangent to the velocity field.



$$\underline{v} = \frac{q}{\phi}$$

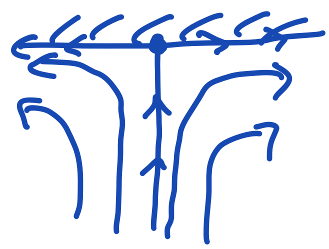
In a <u>steady</u> flow field streamlines are particle trajectories.

Definition of velocity field provides an system of ODE's to compute streamlines

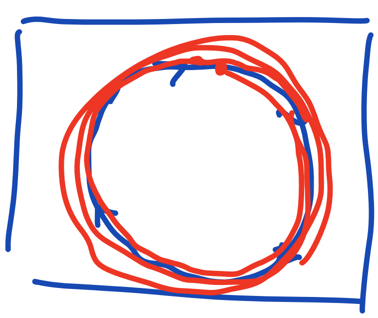
$$
\left.
\begin{array}{ll}
1) & \dfrac{dx}{dt} = v_x(\underline{x}) \\[2ex]
2) & \dfrac{dy}{dt} = v_y(\underline{x})
\end{array}
\right\}
\qquad \frac{dy}{dx} = \frac{v_y}{v_x}
\qquad \underline{v} = \begin{pmatrix} v_x \\ v_y \end{pmatrix}
$$

Notes:

- generally safer to solve system because $\frac{dy}{dx}$ may not be finite, and $y(x)$ may be multivalued
- ODE system has trouble near stagnation points



- only find stagnation points by trial and error.



Next time: Stream function.