Discretizing Stokes with variable viscosity

Governing equations:

lin. mom. : $-\nabla\cdot\left[\mu(T)\left(\nabla\underline{v} - \nabla^T\underline{v}\right)\right] + \nabla\pi = 0$

mass : $\nabla\cdot\underline{v} = 0$

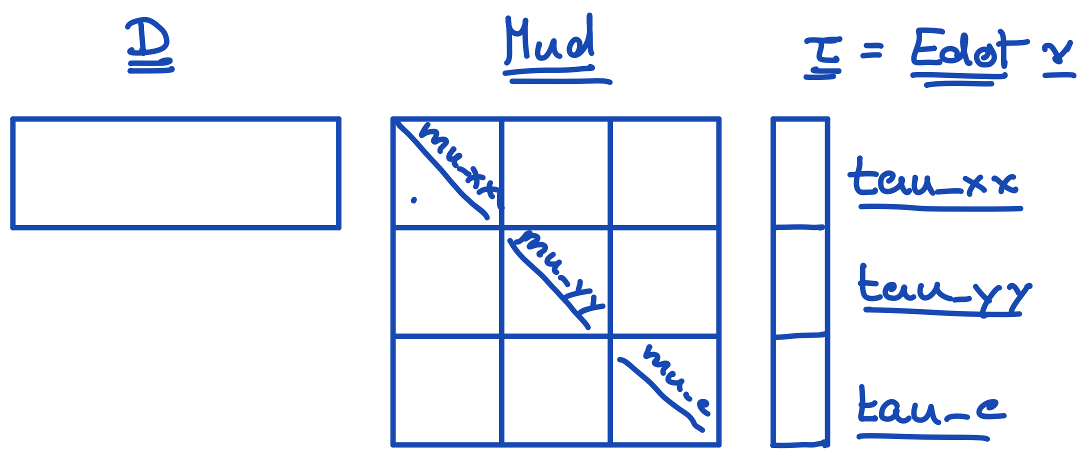
Discrete system:

$$-\underline{\underline{D}} * \underline{\underline{Mud}}\ \underline{\underline{Edot}}\ \underline{v} + \underline{\underline{Gp}} * p = 0$$

$$\underline{\underline{Dp}} * \underline{v} = 0$$

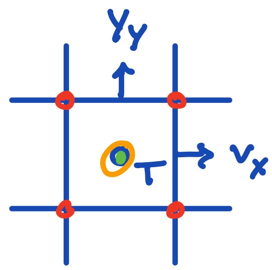
$\underline{\underline{Mud}}$ is a diagonal matrix similar to $\underline{\underline{Kd}}$ that contains the appropriate average of $\mu(T)$

General structure of $\underline{\underline{Mud}}$



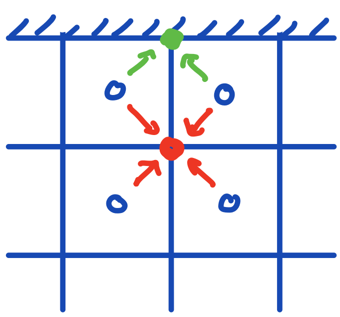
The Mud matrix has 3 diagonal blocks containing the vectors

- mu_xx
- mu_yy
- mu_c



mu_xx and mu_yy are at cell centers ⇒ no averaging but mu_c is averaged from cell center to the cell corners.

Averaging to cell corners



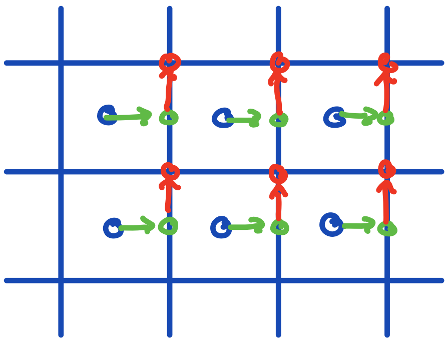
Need to average the 4 surrounding cell centers to corner in interior. On boundary only the neighboring two cell centers!

⇒ Different mean matrix Mc

$\underline{\underline{M_c}}$ is N_c by N matrix averaging from the cell centers to cell corners

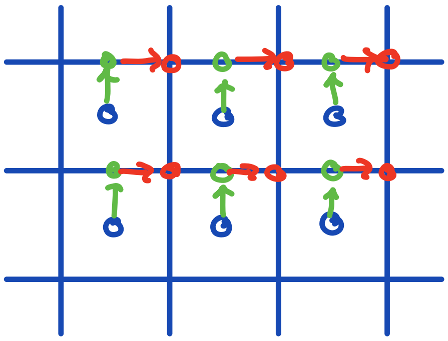
$N_c = (N_x+1)(N_y+1)$ is the number of corners

This matrix can be built by composing the mean matrices from different grids



1) Centers to x-faces

2) x-faces to corners



1) Centers to y-faces

2) y-faces to corners

$\Rightarrow$ We choose x-faces (both give same result)

In build_stokes_ops.m we have

Mp mean matrix on pressure grid

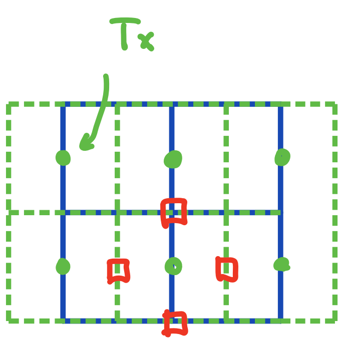
Mx mean matrix on x-velocity grid

$$\underline{\underline{Mp}} = \begin{bmatrix} \underline{\underline{Mpx}} \\ \underline{\underline{Mpy}} \end{bmatrix}$$ averages to both x & y faces.

$\underline{\underline{Mpx}} = \underline{\underline{Mp}}$ [1 : Grid.p.Nfx, :]

averages from centers to x-faces

$\underline{Tx} = \underline{\underline{Mpx}} * \underline{T}$ temperature averaged to x-faces



$$\underline{\underline{Mx}} = \begin{bmatrix} \underline{\underline{Mxx}} \\ \underline{\underline{Mxy}} \end{bmatrix}$$ is mean matrix on the x-velocity grid

Mxx averages from x-faces back to cell center

Mxy averages from x-faces to cell corners

$$\Rightarrow \underline{T_c} = \underline{\underline{M_{xy}}}\,\underline{T_x} = \underset{N_c \cdot N_{fx}}{\underline{\underline{M_{xy}}}}\ \underset{N_{fx} \cdot N}{\underline{\underline{M_{px}}}}\ \underset{N \cdot 1}{\underline{T}}$$

Hence the matrix averaging to corners is

$$\boxed{\underline{\underline{M_c}} = \underline{\underline{M_{xy}}} * \underline{\underline{M_{px}}}} \qquad N_c \text{ by } N \text{ matrix}$$

$\underline{\underline{M_c}}$ can be used in comp_mean.m like any other mean matrix

$\Rightarrow$ calculate $\underline{\text{mu\_c}}$ vector

<u>Two options:</u>

1) Evaluate first then average

- $\underline{\text{mu\_c}} = \text{mu}(\underline{T})$ evaluate viscosity in cell center

- $\underline{\underline{\text{Mud}}} = \text{comp\_mean}(\underline{\text{mu\_c}}, \underline{\underline{M_c}}, \pm 1, \ldots)$

  here we can choose arithmetic/harmonic

2) Average first then evaluate

- $\underline{T_c}$ = comp_mean(T, Mc, 1 ...

  this creates diagonal matrix with lots of zero that blow up if we eval. mu($\underline{T_c}$)

  Instead: $\underline{T_c} = \underline{\underline{M_c}}\, \underline{T}$

  This is a vector of arithmetic averages. Since we are interpolating a smooth function harmonic average does not make sense

- $\underline{\underline{Mud}}$ = spdiags(mu($\underline{T_c}$), 0, ...)

Typically we choose the <u>second</u> option!

But this only computes the means to the corner the full $\underline{\underline{Mud}}$ matrix also has cell center values.

$\Rightarrow$ need to add cell center values

Full $\underline{\underline{Mud}}$ matrix

One of the few cases where it seems simple but requires some more thought.

$$\underline{tau} = \begin{bmatrix} tau\_xx \\ tau\_yy \\ tau\_c \end{bmatrix} = \underbrace{\begin{bmatrix} \underline{\underline{Ip}} \\ \underline{\underline{Ip}} \\ \underline{\underline{Mc}} \end{bmatrix}}_{\underline{\underline{Ms}}} \begin{bmatrix} eps\_dot\_xx \\ eps\_dot\_yy \\ eps\_dot\_c \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \end{matrix}\right\} \rightarrow \text{centers !} \\ \rightarrow \text{corners} \end{matrix}$$

$\Rightarrow$ just add identities from pressure grid an use $\underline{\underline{Ms}}$ instead of $\underline{\underline{Mc}}$ in averaging so that $\underline{\underline{A}} = 2\,\underline{\underline{D}} * \underline{\underline{Mud}} * \underline{\underline{Edot}}$

$\Downarrow$

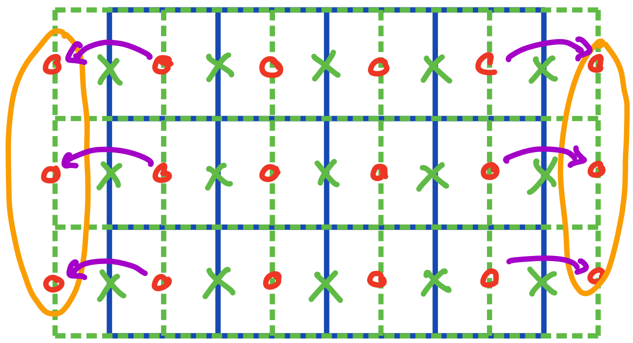
size of $\underline{\underline{Mud}}$ generated by $\underline{\underline{Ms}}$ is wrong !

Problem: $\underline{eps\_dot} = \underline{\underline{Edot}} * \underline{v}$ (Lecture 21)

$$eps\_dot = \begin{bmatrix} eps\_dot\_xx \\ eps\_dot\_yy \\ eps\_dot\_c \end{bmatrix}$$

The xx and yy vectors are evaluated at cell center but their length is <u>not</u> N !

Consider the standard grid $N_x = 4$ $N_y = 3$



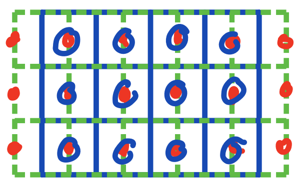
eps_dot_xx = $\underline{\underline{G_{xx}}}$ $\underline{v_x}$

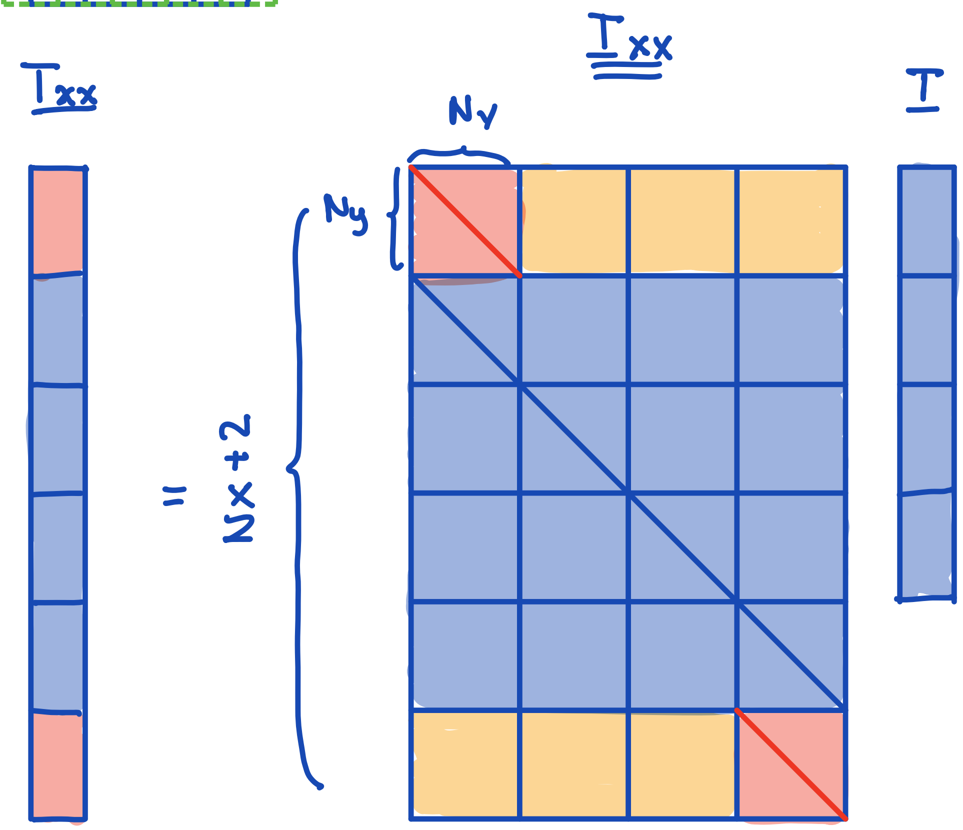
Additional entries on xmin & xmax bnd of x-velocity grid.

$\Rightarrow$ length of eps_dot_xx is $(N_x+2)\,N_y = 18$.

Additional entries are the zero derivatives from the natural BC's and their not used.

Need to come up with an "identity matrix" that copies closest T there. Value does not matter, but it should not be zero.





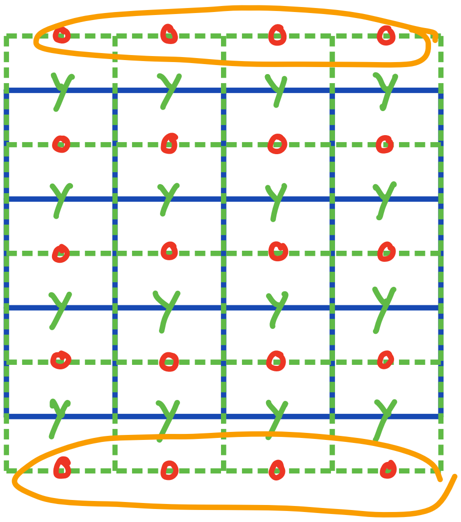
$\underline{\underline{I_p}}$ identity on primary grid

$\underline{\underline{I_{ny}}}$ is N_y by N_y identity

$\underline{\underline{Z_{ny}}}$ is N_y by $N-N_y$ zero's matrix

$$\boxed{\underline{\underline{I_{xx}}} = [\underline{\underline{I_{ny}}}\ \underline{\underline{Z_{ny}}} ; \underline{\underline{I_p}} ; \underline{\underline{Z_{ny}}}\ \underline{\underline{I_{ny}}}]}$$

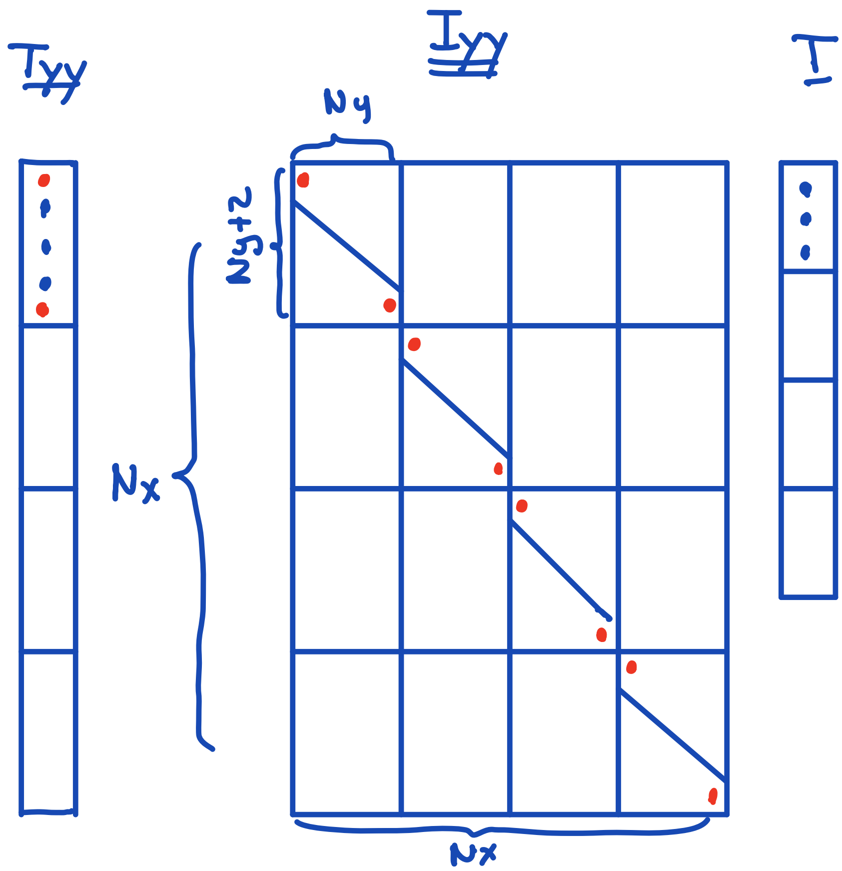
Similarly eps_dot_yy is not length N!



eps_dot_yy = Gyy * vy

Additional entries on ymin & ymax boundaries of y-velocity grid.

length of eps_dot_yy is $N_x\,(N_y+2) = 20$



This can be assembled with tensor products.

Build diagonal block (1D)

$\underline{zy} = \text{zeros}(1, Ny-1);$ $\quad \underline{\underline{I_y}} = \text{eye}(Ny);$

$\underline{\underline{I_y}} = [1 \ \underline{zy} ; \underline{\underline{I_y}} ; \underline{zy} \ 1];$ $\quad \underline{\underline{I_x}} = \text{eye}(Nx)$

$$\boxed{\underline{\underline{I_{yy}}} = \text{kron}(\underline{\underline{I_x}}, \underline{\underline{I_y}})}$$

Now we can assemble the full Stokes mean matrix:

$$\boxed{\underline{\underline{M_s}} = \begin{bmatrix} \underline{\underline{I_{xx}}} \\ \underline{\underline{I_{yy}}} \\ \underline{\underline{M_c}} \end{bmatrix}}$$