vectorOps - non-uniform

Interpolation/extrapolation operators

myFaceAverage (Face2AdjacentFace)

face2CellCenter interpolation

The value u_n is the face normal velocity. At location t, meaning the direction into the face of which you want the average:

$$\left\{u_{ni+\frac{1}{2}} = \frac{1}{2}(u_{ni} + u_{ni+1})\right\}_t$$

Similarly, at location $t-1$,

$$\left\{u_{ni+\frac{1}{2}} = \frac{1}{2}(u_{ni} + u_{ni+1})\right\}_{t-1}$$

cellCenter2Face interpolation

Now, we finally must interpolate between these two values (which are currently at cell centers)

$$\frac{u_{ni+\frac{1}{2},t} - u_{ni+\frac{1}{2},t-1}}{h_{ct} - h_{ct-1}} = \frac{faceAve_{ni+\frac{1}{2},t-1/2} - u_{ni+\frac{1}{2},t-1}}{h_{nt-1} - h_{ct-1}}$$

Solving for this yields

$$faceAve_{ni+\frac{1}{2},t-1/2} = u_{ni+\frac{1}{2},t-1} + \frac{h_{nt-1} - h_{ct-1}}{h_{ct} - h_{ct-1}}\left(u_{ni+\frac{1}{2},t} - u_{ni+\frac{1}{2},t-1}\right)$$

Substituting we have

$$faceAve_{ni+\frac{1}{2},t-1/2} = \frac{1}{2}(u_{ni} + u_{ni+1})_{t-1} + \frac{h_{nt-1} - h_{ct-1}}{h_{ct} - h_{ct-1}}\left(\frac{1}{2}(u_{ni} + u_{ni+1})_t - \frac{1}{2}(u_{ni} + u_{ni+1})_{t-1}\right)$$

Collecting terms we have

$$\begin{aligned} faceAve_{ni+\frac{1}{2},t-1/2} &= \frac{1}{2}\Bigg\{u_{nit}\,\frac{h_{nt-1} - h_{ct-1}}{h_{ct} - h_{ct-1}} + u_{ni+1,t}\,\frac{h_{nt-1} - h_{ct-1}}{h_{ct} - h_{ct-1}} + u_{ni,t-1}\left\{1 - \frac{h_{nt-1} - h_{ct-1}}{h_{ct} - h_{ct-1}}\right\} \\ &\quad + u_{ni+1,t-1}\left\{1 - \frac{h_{nt-1} - h_{ct-1}}{h_{ct} - h_{ct-1}}\right\}\Bigg\} \end{aligned}$$

myFace2Edge

Face to edge requires a linear interpolation since the cell edges are not necessarily midway between two cell faces. The cell edges are coincident with the two cell faces that are used for the interpolation.

$$\frac{f_i - f_{i+1}}{h_{ci} - h_{ci+1}} = \frac{e_i - f_{i+1}}{h_{ni} - h_{ci+1}}$$

Solving for e_i yields

$$e_i = f_{i+1} + \alpha(f_i - f_{i+1})$$

where

$$\alpha = \frac{h_{ni} - h_{ci+1}}{h_{ci} - h_{ci+1}}$$

This routine only needs interpolation because the edges are of size $N + 1$ and lie in between cell faces.

myEdge2Node

Edge to node requires interpolation since the nodes are not necessarily midway between the edges.

$$\frac{N_i - e_{i+1}}{h_{ni} - h_{ci+1}} = \frac{e_i - e_{i+1}}{h_{ci} - h_{ci+1}}$$

Solving for N_i we have

$$N_i = e_{i+1} + \alpha(e_i - e_{i+1})$$

Where

$$\alpha = \frac{h_{ni} - h_{ci+1}}{h_{ci} - h_{ci+1}}$$

The edge values in the fictitious cells are assumed to be unknown. Therefore, this routine requires forward and backward extrapolation from the first and last interior edges to the first and last nodes.

Backward Extrapolation

$$\frac{N_i - e_{i+1}}{h_{ni} - h_{ci+1}} = \frac{e_{i+1} - N_{i+1}}{h_{ci+1} - h_{ni+1}}$$

Solving for N_i yields

$$N_i = e_{i+1} + \alpha(e_{i+1} - N_{i+1})$$

Where

$$\alpha = \frac{h_{ni} - h_{ci+1}}{h_{ci+1} - h_{ni+1}}$$

Forward Extrapolation

$$\frac{N_{i+1} - e_{i+1}}{h_{ni+1} - h_{ci+1}} = \frac{e_{i+1} - N_i}{h_{ci+1} - h_{ni}}$$

Solving for N_{i+1} yields

$$N_i = e_{i+1} + \alpha(e_{i+1} - N_i)$$

Where

$$\alpha = \frac{h_{ni+1} - h_{ci+1}}{h_{ci+1} - h_{ni}}$$

myNode2Edge

Node to edge requires an ordinary average for interior data as well as extrapolation to get the fictitious edges.

$$e_i = \frac{1}{2}(N_i + N_{i+1})$$

Backward Extrapolation

$$\frac{e_i - e_{i+1}}{h_{ci} - h_{ci+1}} = \frac{N_i - e_{i+1}}{h_{ni} - h_{ci+1}}$$

Solving for e_i yields

$$e_i = e_{i+1} + \alpha(N_i - e_{i+1})$$

Where

$$\alpha = \frac{h_{ci} - h_{ci+1}}{h_{ni} - h_{ci+1}}$$

Forward Extrapolation

$$\frac{e_{i+1} - e_i}{h_{ci+1} - h_{ci}} = \frac{N_i - e_i}{h_{ni} - h_{ci}}$$

Solving for e_{i+1} yields

$$e_{i+1} = e_i + \alpha(N_i - e_i)$$

Where

$$\alpha = \frac{h_{ci+1} - h_{ci}}{h_{ni} - h_{ci}}$$

myFace2CellCenter

Let p_i be the cell center value between f_i and f_{i+1}. The same formulas for the exterior extrapolation must be used as the ones in the cellCenter2Face to ensure that applying them consecutively results in the original input.

Face to cell center is simple average since the cell centers are mid way between the faces

$$p_i = \frac{1}{2}(f_i + f_{i+1})$$

To calculate the cell center on the exterior (fictive) cells, we must extrapolate:

Backward Extrapolation

From the cellCenter2Face, we had

$$\frac{p_i - p_{i+1}}{h_{ci} - h_{ci+1}} = \frac{f_i - p_{i+1}}{h_{ni} - h_{ci+1}}$$

Solving for p_i yields

$$p_i = p_{i+1} + \alpha(f_i - p_{i+1})$$

Where

$$\alpha = \frac{h_{ci} - h_{ci+1}}{h_{ni} - h_{ci+1}}$$

Forward Extrapolation

Version 1

From the cellCenter2Face, we had

$$\frac{p_i - p_{i+1}}{h_{ci} - h_{ci+1}} = \frac{f_i - p_{i+1}}{h_{ni} - h_{ci+1}}$$

Using the result from the CellCenter2Face

$$f_i = p_{i+1} + \alpha(p_i - p_{i+1})$$

Where

$$\alpha = \frac{h_{ni} - h_{ci+1}}{h_{ci} - h_{ci+1}}$$

We can solve for p_{i+1}

$$p_{i+1}(1 - \alpha) = f_i - \alpha p_i$$

Solving for p_{i+1} we have

$$p_{i+1} = \frac{f_i - \alpha p_i}{1 - \alpha}$$

Version 2

Similarly, we can use the same formula, but interpolate from another location so that p_{i+1} does not show up twice. i.e.

$$\frac{p_i - f_i}{h_{ci} - h_{ni}} = \frac{f_i - p_{i+1}}{h_{ni} - h_{ci+1}}$$

Solving for p_{i+1} yields

$$p_{i+1} = f_i - \alpha(p_i - f_i)$$

Where

$$\alpha = \frac{h_{ni} - h_{ci+1}}{h_{ci} - h_{ni}}$$

Version 3

Similar to above, we have

$$p_{i+1} = p_i + \alpha(f_i - p_i)$$

Where

$$\alpha = \frac{h_{ci} - h_{ci-1}}{h_{ni} - h_{ci-1}}$$

myCellCenter2Face

Let p_i be the cell center value between f_i and f_{i+1}. Cell center to face requires a linear interpolation since the cell faces are not necessarily midway between two cell centers.

$$\frac{p_i - p_{i+1}}{h_{ci} - h_{ci+1}} = \frac{f_i - p_{i+1}}{h_{ni} - h_{ci+1}}$$

Solving for the face value yields

$$f_i = p_{i+1} + \alpha(p_i - p_{i+1})$$

Where

$$\alpha = \frac{h_{ni} - h_{ci+1}}{h_{ci} - h_{ci+1}}$$

myCellCenter2Node

Cell centers to node requires linear interpolation between the cell centers in the x,y and z directions. This was accomplished by using the cell center2Face interpolation along all directions.

myFace2Node

Face to node requires linear interpolation between two faces to get edge data, then again to land on the node.

Derivative operators

Face-based operators

myFaceAdvect

myFaceCurl

myFaceDiv

myFaceLap

Node/Cell center based operators

myDiv

myCurl

myAdvect

myLap