Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Home Affairs FDHA
**Federal Office of Meteorology and Climatology MeteoSwiss**

# dusk & dawn - Differential Operators Exercise

DSL Training Workshop

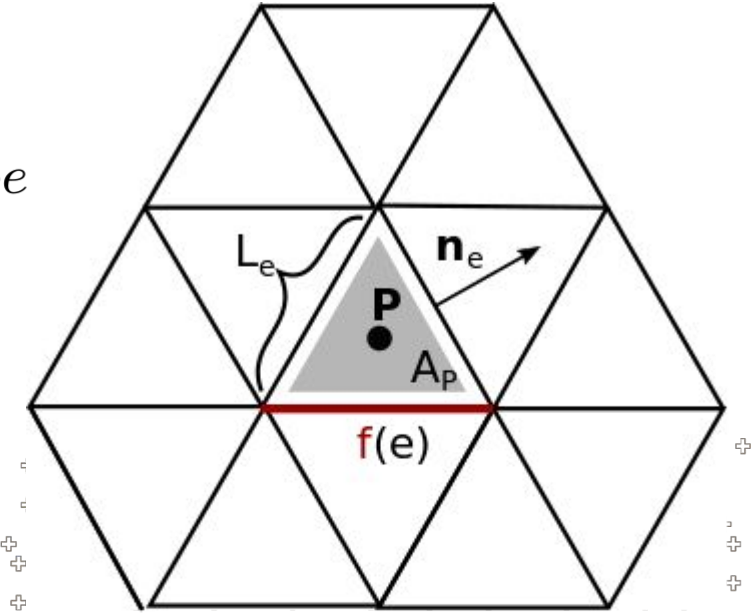
# Contents

- For this exercise, you will implement (**FVM approximations** to) the three most basic **differential operators** using dusk & dawn
- You will use basic dawn concepts, especially the **reduction** primitive

# The Gradient

- The Gradient is an operator in a scalar field f that returns a vector ∇f
- It points into the direction of the steepest slope of the scalar field, and its magnitude is a measure on how steep that slope is
- On a (triangular) FVM mesh, the Gradient can be approximated using:

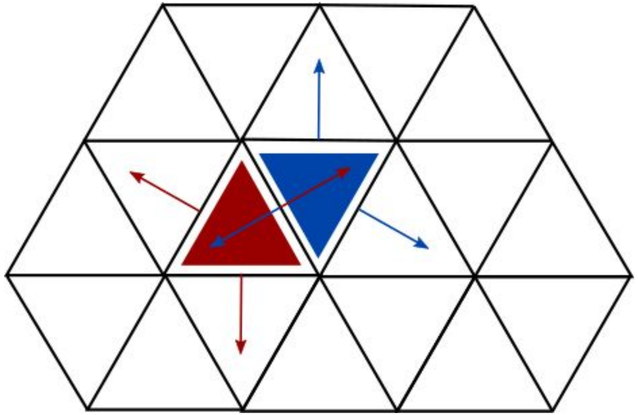
$$\langle \nabla f(\mathbf{P}) \rangle_{FVM} = \frac{1}{A_P} \sum_{e=1}^{3} f(e) L_e \mathbf{n}_e$$



- Note that this equation only holds if the normals $\mathbf{n}_f$ point outside the triangle with centroid P

# A Note on Geometrical Factors

- The equation for the gradient on the last slide holds only if all normals point outside
- However, one clearly can't arrange a mesh such that all normals point outside

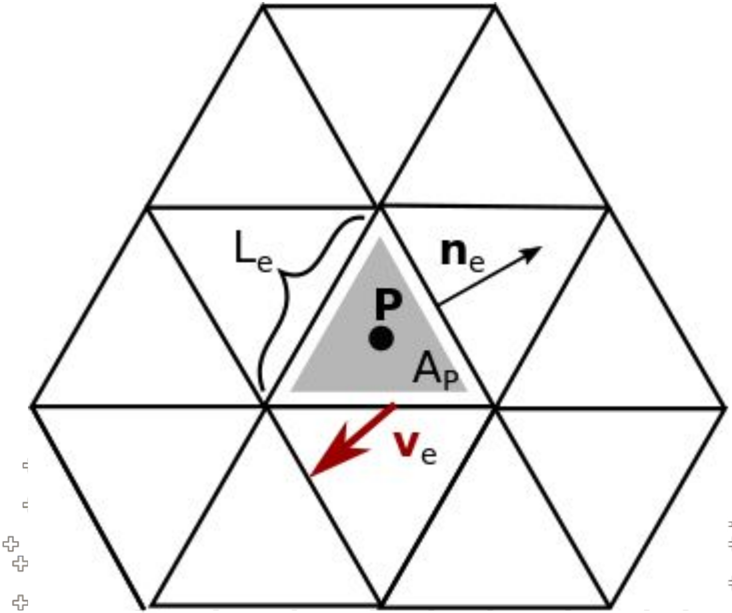


- To check, for each normal, in which side it points during the computation of the gradient is not efficient (involves the evaluation of a dot product)
- For this exercise, geometric factors (`edge orientation`) have been prepared for you to ensure just that. You do not need to wonder about their location type or how they work exactly, just be aware that you **need to multiply by the edge orientation every time you access a normal**! (e.g. nx*`edge_orientation`)

# The Divergence

- The Divergence is an operator on a vector field **v** = [u, v] that returns a scalar ∇▪**v**
- The scalar indicates the intensity of the source (extension / contraction) of a vector field (e.g. the divergence of heating, and hence expanding, air would be positive)

$$\langle \nabla \cdot \mathbf{v}(\mathbf{P}) \rangle_{FVM} = \frac{1}{A_P} \sum_{e=1}^{3} (\mathbf{v}_e \cdot \mathbf{n}_e)\, L_e$$

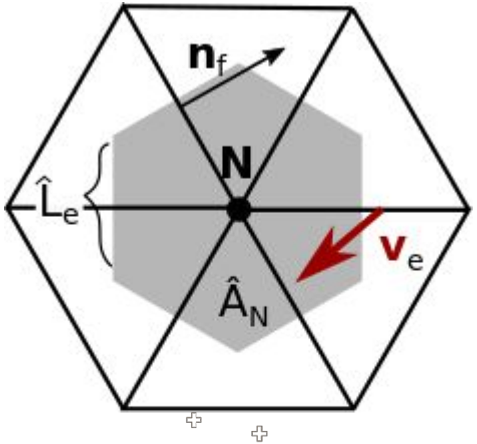


- Note that this equation only holds if the normals **n**$_f$ point outside the triangle with centroid P

# The Curl

- The Curl is an operator on a vector field **v** = [u, v, w] that returns a vector **∇×v**
- Strictly speaking, the curl is only defined in a 3d field, but emulating it in 2D using **v** = [u, v, 0] will reveal that only ever the third component of **∇×v** takes a value (other than zero).
- So **for our intents** and purposes, the **curl is scalar (!)** and is a measure for the rotation of the vector field **v**

$$\langle \nabla \times \mathbf{v}(\mathbf{N}) \rangle_{FVM} = \frac{1}{\hat{A}_N} \sum_{e=1}^{6} (\mathbf{n}_e \cdot \mathbf{v}_e)\, \hat{L}_e$$



- Note that this equation only holds if the normals **n**$_e$ form a left handed coordinate system with vectors **e**$_i$ (i.e. unit vectors along the edges) and a vector **z** pointing into the plane

# Exercise

- The exercise consists of implementing these three differential operators
- The function for the **gradient** is this wave function**:**

Function:

$$f(x, y) = sin(x)cos(x)$$

Gradient:

$$\nabla f(x, y) = \begin{bmatrix} \cos(x)\sin(y) \\ sin(x)cos(x) \end{bmatrix}$$

# Exercise

- The exercise consists of implementing these three differential operators
- The vector field for the **divergence** and **curl** are the spherical harmonics:

Field:

$$\begin{cases} u(x,y) := \dfrac{1}{4}\sqrt{\dfrac{105}{2\pi}} \cos 2x \, \cos^2 y \, \sin y \quad , \\ v(x,y) := \dfrac{1}{2}\sqrt{\dfrac{15}{2\pi}} \cos x \, \cos y \, \sin y \qquad . \end{cases}$$

Divergence:

$$\nabla \cdot \mathbf{v} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = \frac{-1}{2\sqrt{2\pi}} \left( \sqrt{105} \, \sin 2x \, \cos^2 y \, \sin y \quad , + \sqrt{15} \, \cos x \, \cos 2y \right)$$

Curl:

$$\nabla \times \mathbf{v} = \left( \frac{1}{4}\sqrt{\frac{105}{2\pi}} \right) \cos(2x)\cos(y)(\cos(y)\cos(y) - 2\sin(y)\sin(y)) - \left( \frac{1}{2}\sqrt{\frac{15}{2\pi}} \right) \cos(y)\sin(x)\sin(y)$$

# Variable Reference

An overview over all variables is given below. The ones in bold are the ones you're supposed to be writing to. The others can be treated as read only.

| | |
|---|---|
| f: Field[Edge] | wave function on edges |
| **f_x**: Field[Cell], **f_y**: Field[Cell] | gradient of wave function on cells |
| nx: Field[Edge], ny: Field[Edge] | cell normals on edges (x and y component) |
| L: Field[Edge] | edge lengths |
| A: Field[Cell] | cell areas |
| edge_orientation: Field[Cell > Edge] | sparse dimension that indicates which normals need to be flipped for gradient / div computation (+1/-1) |
| u: Field[Edge], v: Field[Edge] | vector field / spherical harmonics on edges |
| **uv_div**: Field[Cell] | divergence of vector field /spherical harmonics on cells |
| dualL: Field[Edge] | dual edge length $\hat{L}_e$ (c.f. slide 8) |
| dualA: Field[Vertex] | dual cell area $\hat{A}_N$ (c.f. slide 8) |
| **uv_curl**: Field[Vertex] | curl of vector field / spherical harmonics on vertices |
| edge_orientation: Field[Vertex > Edge] | sparse dimension that indicates which normals need to be flipped in curl computation (+1/-1) |

# Hints

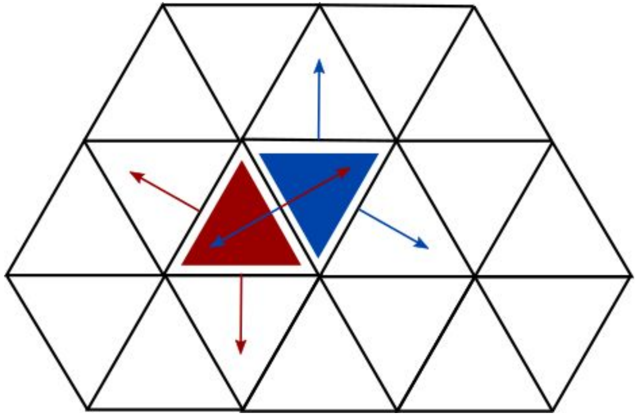
- divergence and gradient operate on the same neighbor chain
- the computations are all quite short
  - do not overthink things
  - all three operators can be expressed in one or two lines

# Appendix

- The following slides give a short preview / basic explanation on sparse dimensions
- You don't need to consider them for this exercise, and sparse dimensions will be explained again later on in more detail
- Still, if you're interested and have time to spare for this exercise session, feel free to study them at your leisure

# A Note on Geometrical Factors

- The equation for the gradient on the last slide holds only if all normals point outside
- However, one clearly can't arrange a mesh such that all normals point outside

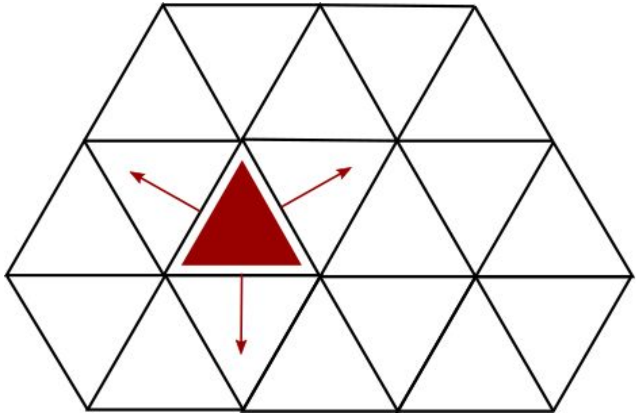


- To check, for each normal, in which side it points during the computation of the gradient is not efficient (involves the evaluation of a dot product)
- Usually, this is solved by pre-computing a geometrical factor. In this case three signs for each triangle, that indicate whether the normal is flipped during the summation
- This is a prime example of a sparse field

# A Note on Geometrical Factors

- The equation for the gradient on the last slide holds only if all normals point outside
- However, one clearly can't arrange a mesh such that all normals point outside

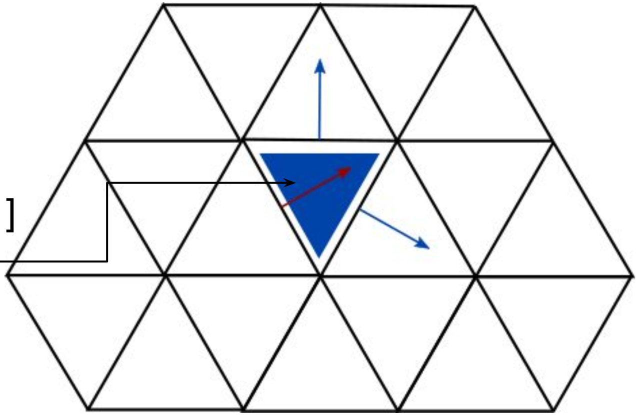
edge_orientation(**cell_a**,:) = [1, 1, 1]



- To check, for each normal, in which side it points during the computation of the gradient is not efficient (involves the evaluation of a dot product)
- Usually, this is solved by pre-computing a geometrical factor. In this case three signs for each triangle, that indicate whether the normal is flipped during the summation
- This is a prime example of a sparse dimension

# A Note on Geometrical Factors

- The equation for the gradient on the last slide holds only if all normals point outside
- However, one clearly can't arrange a mesh such that all normals point outside



edge_orientation(**cell_b**,:) = [**-1**, 1, 1]

- To check, for each normal, in which side it points during the computation of the gradient is not efficient (involves the evaluation of a dot product)
- Usually, this is solved by pre-computing a geometrical factor. In this case three signs for each triangle, that indicate whether the normal is flipped during the summation
- This is a prime example of a sparse dimension