Homework Assignment 12

Please submit the following files as indicated below: source code PDF file image file video file

Question 1 | **3 marks** | Today we're solving the linear advection equation

$$\frac{\partial u(t)}{\partial t} + \operatorname{div}(u(t)a(t)) = 0 \qquad \text{in } Q =]0,T[\times\Omega \tag{1a}$$

$$u(0) = u_0 \qquad \text{in } \Omega \tag{1b}$$

$$u(t) = g(t) \qquad \text{on } \Sigma_- = \{\, (t,x) \in]0,T[\times\partial\Omega \mid a(t,x)\cdot n(x) < 0 \,\}, \tag{1c}$$

where $a :]0,T[\times\Omega \to \mathbb{R}^2$ is a given vector field, thought of as the flow velocity of a carrier fluid, and $u_0 : \Omega \to [0,1]$ the initial concentration of a solute. $g :]0,T[\times\partial\Omega \to [0,1]$ prescribes the concentration on that part of the boundary $\partial\Omega$ where the flow of solvent is directed into the domain and we use n to denote the outward pointing unit normal vectors on $\partial\Omega$. For simplicity we assume in what follows that a and g do not depend on time.

Recall that upwind discontinuous Galerkin methods employ the bilinear form

$$\sum_{T\in\mathcal{T}^h} -\int_T ua\cdot\nabla v \,\mathrm{d}x + \sum_{e\in\mathcal{E}^h}\int_e [v]u_{\mathrm{up}}a\cdot n_+ \,\mathrm{d}s$$

to discretise the transport term $\operatorname{div}(ua)$.

(a) We define the positive and negative parts of a function f by

$$f^{\mathrm{pos}} = \frac{f+|f|}{2} \qquad \text{and} \qquad f^{\mathrm{neg}} = \frac{f-|f|}{2}.$$

Re-write the edge integral in the upwind DG form in terms of g, u_+, u_-, $(a\cdot n)_+^{\mathrm{pos}}$, $(a\cdot n)_+^{\mathrm{neg}}$ and $[v]$:

- If e is an interior edge:

$$\int_e [v]u_{\mathrm{up}}a\cdot n_+ \,\mathrm{d}s =$$

- If e is an exterior edge (no $\pm$ subscripts or $[\,]$ brackets needed, there is only one neighbouring triangle):

$$\int_e [v]u_{\mathrm{up}}a\cdot n_+ \,\mathrm{d}s =$$

(b) Complete the FEniCS script `hw12.py` to solve the linear advection equation with

$$T = 2\pi \qquad a(x) = \begin{pmatrix} -x_2 \\ x_1 \end{pmatrix} = r\begin{pmatrix} -\sin\phi \\ \cos\phi \end{pmatrix} \text{ in polar coordinates} \qquad g(x) \equiv 0$$

and the domain and initial data provided. Note that with these data, the advection equation rotates the initial field u_0 around the origin in anticlockwise direction, like a rigid body on a turntable.

Use a moderate degree (e.g. $r = 1$ or 2) for the DG-discretisation in space and use the method we are most familiar with, the θ-method, for time stepping. Any value of θ which results in a stable scheme is fine.

Hint: In FEniCS we use `dS` for integrals over interior edges and `ds` for integrals over exterior edges. You must add `('+')` or `('-')` to all discontinuous functions in integral expressions containing `dS`, e.g.

```
u('-')*jump(v)*dS
```

to specify on what side of the edge they should be evaluated. `jump(v)` is a shortcut for `v('+')-v('-')`.

(c) Inspect your numerical solution in ParaView, using the 'Warp by Scalar'-Filter. Recall that FEniCS generally exports solution data as continuous, piecewise linear functions, so don't be surprised when you cannot see any discontinuous jumps in your visualisation.

Describe any discrepancies between the numerical solution and the exact solution, using the appropriate terminology.

Hint: For general triangular meshes, the upwind DG(r) method discretises the advection term with an accuracy of order $r + \frac{1}{2}$, which increases to $r + 1$ on meshes with certain regularity.

Question 2 | **2 marks** | Make a copy of your script `hw12.py` and modify it to solve the advection-diffusion problem

$$\frac{\partial u(t)}{\partial t} + \operatorname{div}(u(t)a - D\nabla u(t)) = 0 \qquad \text{in } Q =]0,T[\times\Omega \tag{2a}$$

$$u(0) = u_0 \qquad \text{in } \Omega \tag{2b}$$

$$(u(t)a - D\nabla u(t)) \cdot n = 0 \qquad \text{on } \Sigma =]0,T[\times\partial\Omega \tag{2c}$$

instead, using $D = 0.01$ and all other parameters as in Question 1. The new boundary condition is a no-flux Robin condition. It admits an interpretation of a semi-permeable membrane which allows the solvent, but not the solute to pass through.

Discretise the diffusive flux with the symmetric interior penalty form

$$\sum_{T\in\mathcal{T}^h} \int_T D\nabla u \cdot \nabla v \,\mathrm{d}x \qquad \text{(bilinear form of conforming methods)}$$

$$- \sum_{e\in\mathcal{E}^h\setminus\mathcal{I}^h} \int_e vD\nabla u \cdot n \,\mathrm{d}s - \sum_{e\in\mathcal{I}^h} \int_e [v] \langle D\nabla u\rangle \cdot n_+ \,\mathrm{d}s \qquad \text{(consistency)}$$

$$- \sum_{e\in\mathcal{I}^h} \int_e [u] \langle D\nabla v\rangle \cdot n_+ \,\mathrm{d}s \qquad \text{(symmetry)}$$

$$+ \sum_{e\in\mathcal{I}^h} \frac{\sigma}{h_e} \int_e [u][v] \,\mathrm{d}s \qquad \text{(interior penalty)}$$

as derived in class, here written in a form that can be translated directly into FEniCS code (but note that the boundary condition has not been applied yet). Use the penalty parameter $\sigma = 0.1$ and $h_e = \langle h\rangle$.

Hint: The commands `h = CellSize(mesh)` and `avg(u)` may be helpful.

Your Learning Progress | What is the one most important thing that you have learnt from this assignment?

Any new discoveries or achievements towards the objectives of your course project?

What is the most substantial new insight that you have gained from this course this week? Any *aha moment*?
