Math 228B - Project 1: Due date is February 13 (uploaded on bCourses)

1. Consider the problem of finding the flow rate of water flowing through a cross-section of a channel of trapezoidal shape. The cross-section, shown in Fig. 1, has a constant perimeter $L = B + 2D$. Therefore, the shape can be described in terms of two parameters: the size of the base in the trapezoid B, and the height of the cross-section H.

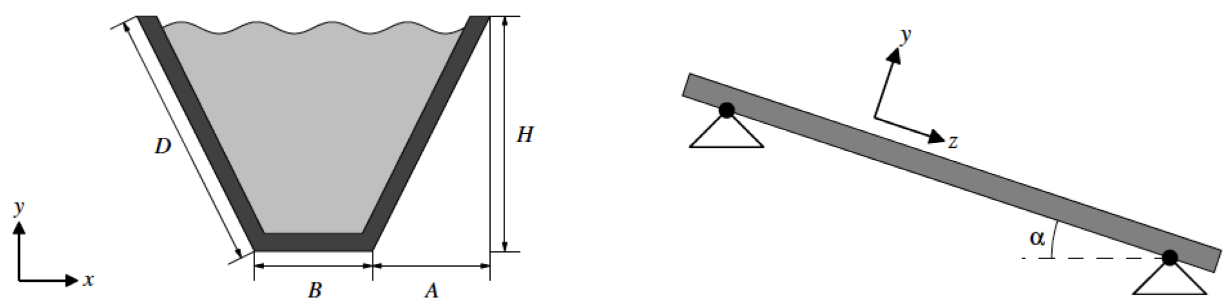

Figure 1: Trapezoidal cross-section and flow channel.

Given B and H, the size of the geometric parameters A and D from Fig. 1 can be calculated as follows:

$$D = \frac{1}{2}(L - B) \quad \text{and} \quad A = \sqrt{\frac{1}{4}(L - B) - H^2}.$$

We are interested in the flow rate Q, defined as the integral of the fluid velocity u (normal to the cross-section) over the cross-section Φ:

$$Q = \int_\Phi u \, dx \, dy.$$

The channel is at the slope α, as shown in Fig. 1, so that the horizontal components of the gravitational force create the pressure gradient for the downward flow. The governing equations for the motion of an incompressible, viscous fluid such as water, are the Navier-Stokes equations:

$$\begin{aligned} \mathbf{u}_t + (\mathbf{u} \cdot \mathbf{u})\mathbf{u} + \nabla p &= \mathbf{f} + \nu \nabla^2 \mathbf{u}, \\ \nabla \cdot \mathbf{u} &= 0. \end{aligned}$$

With the assumption of the fully developed flow, these can be reduced to the Poisson equation for the velocity component u normal to the cross-section of the channel:

$$\nabla^2 u = \frac{g \sin \alpha}{\nu},$$

where g is the gravitational constant, and ν the kinematic viscosity of the fluid. For simplicity we assume that

$$\frac{g \sin \alpha}{\nu} = 1.$$

On the channel walls we assume the classical no-slip condition describing the velocity of the fluid to be equal to zero, and on the free surface we prescribe the zero stress condition $\partial u / \partial \mathbf{n} = 0$, where $\mathbf{n}$ is the outward unit normal to the fluid domain.

The goal of this problem is to calculate the flow, namely the flow rate Q defined above, of the fluid flow in the channel.

Solution guide: Use a finite difference method to solve the problem. Due to the symmetry of the problem, consider only one half of the fluid domain, i.e., consider the right half of the channel, as shown in Fig. 2, denoted by Ω. Denote by $P_i, i = 1, 2, 3, 4$ the corners of the fluid domain boundary, as shown in Fig. 2. The boundary conditions on the boundary of Ω are:

$$u = 0, \quad \text{on } P_1P_2 \text{ and } P_2P_3, \tag{1}$$

$$\frac{\partial u}{\partial \mathbf{n}} = 0, \quad \text{on } P_3P_4, \tag{2}$$

and the symmetry boundary condition:

$$\frac{\partial u}{\partial \mathbf{n}} = 0, \quad \text{on } P_1P_4.$$

Map the half domain Ω onto a rectangular (reference) computational domain $\hat{\Omega}$. Use a rectangular grid $(n+1) \times (n+1)$ on $\hat{\Omega}$, giving square cells of size $\Delta\xi = \Delta\eta = 1/n$. To complete the problem:

- In LaTeX, compose a file in which you explain the main steps of your solution, which include the following:

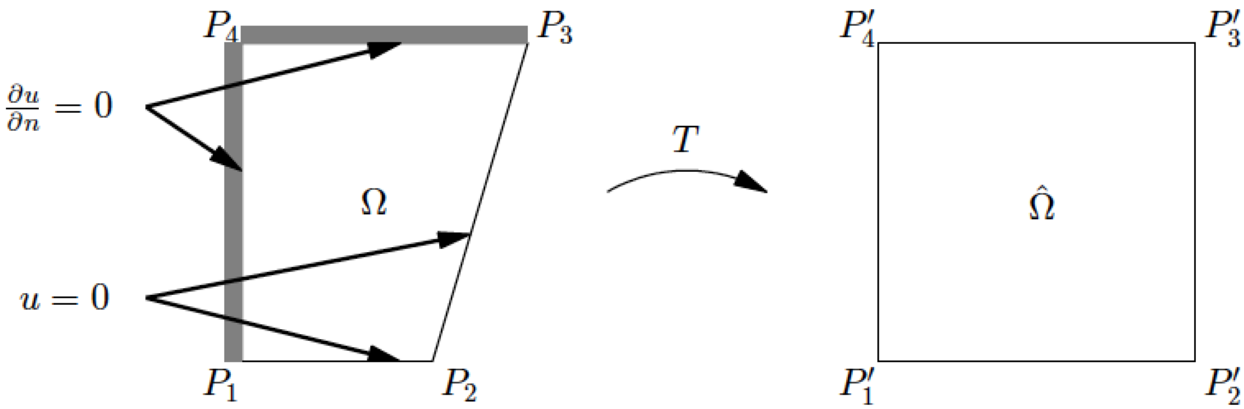

Figure 2: Geometric transformation from the physical space to the computational domain.

- Define the mapping that transforms the original problem defined on Ω, with physical coordinates denoted by (x, y), onto a rectangular domain $\hat{\Omega}$, with coordinates denoted by (ξ, η).

- In transformed coordinates write down second-order accurate finite difference schemes for the discretization of all the derivative terms in the interior of the domain, as well as for the boundary points. You do not have to derive special schemes for the corner points: use the left boundary scheme at the corner P_4, and $u = 0$ at the other three corners.

- Show how to evaluate the integral in the approximate flow rate $\hat{Q}$ numerically in the reference (computational) domain.

- Write a Matlab function that solves the problem for the given values of L, B and H, and grid size n.

- To debug/test the code, calculate the solution and $\hat{Q}$ using the grid size $n = 20$, for $L = 3.0, B = 0.5$ and $H = 1.0$, and compare your results with the sample solution in Fig. 3.

- Make the (log-log) convergence plots and calculate the convergence rates for the error in the output for $\hat{Q}$. Do the calculation for $L = 3.0$, $H = 1.0$, and $B = 0.0, 0.5, 1.0$ (three convergence plots). Verify that you obtain second-order convergence. Use the grid sizes $n = 10, 20, 40, 80$. Since we do not know the exact solution, use the solution for $n = 80$ as a reference.

- Plot in 3D the fluid velocity profile (that is, the profile of the normal component of the fluid velocity) versus the channel cross section for the solution obtained with $L = 3.0$, $B = 0.5$, $H = 1.0$ and $n = 40$.

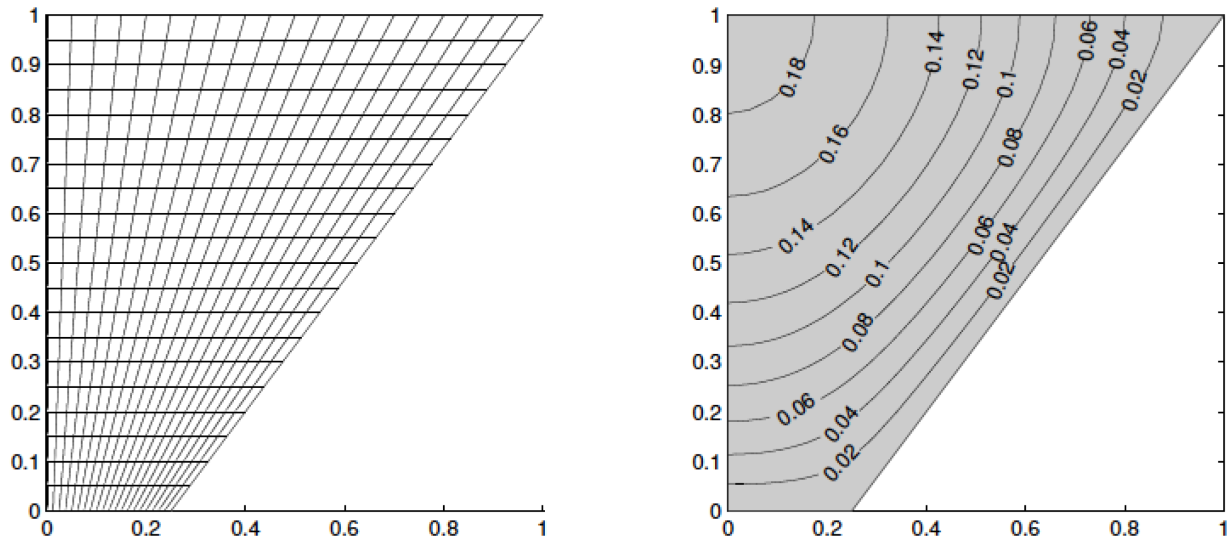

Figure 3: Sample solution for $L = 3.0, B = 0.5, H = 1.0$ and $n = 20$. The plot shows the grid (left) contour lines of the solution (right).

2. Consider the PDE

$$u_t = \kappa u_{xx} - \gamma u,$$

which models diffusion with decay provided $\kappa > 0$ and $\gamma > 0$.

Consider finite difference methods of the form

$$\begin{aligned} U_j^{n+1} &= U_j^n + \frac{k}{2h^2}\left[U_{j-1}^n - 2U_j^n + U_{j+1}^n + U_{j-1}^{n+1} - 2U_j^{n+1} + U_{j+1}^{n+1}\right] \\ &\quad -k\gamma\left[(1-\theta)U_j^n + \theta U_j^{n+1}\right], \end{aligned}$$

where θ is a parameter. In particular, when $\theta = 1/2$ the decay term is modeled with the same centered-in-time approach, as the diffusion term and the method can be obtained by applying the Trapezoidal method to the MOL formulation of the PDE. If $\theta = 0$ then the decay term is handled explicitly. For more general reaction-diffusion equations, it may be advantageous to handle the reaction term explicitly, since these terms are generally nonlinear, so making them implicit would require solving nonlinear systems in each time step.

Compute the local truncation error and show that the method is $\mathcal{O}(k^p + h^2)$ accurate, where $p = 2$ if $\theta = 1/2$, and $p = 1$ otherwise.

Project Submission: Submit the answers in a pdf file, together with the Matlab code, all in one zip-file, and upload them on bCourses.