數學流體力學之理論與計算

Homework Assignment 1

Due date: Prob.1-5 on Oct. 12.
Prob.6 on Oct. 26.

Part I: Theoretical assignments

Problem 1. Complete the following.

1. Let $\delta_{\cdot\cdot}$'s are the Kronecker deltas. Show that

$$\sum_{i=1}^{3} \varepsilon_{ijk}\varepsilon_{irs} = \delta_{jr}\delta_{ks} - \delta_{js}\delta_{kr}\,, \tag{0.1}$$

2. Use (0.1) to show the following identities:

   (a) $u \times (v \times w) = (u \cdot w)v - (u \cdot v)w$ if u, v, w are three 3-vectors.

   (b) $\text{curlcurl}u = -\Delta u + \nabla \text{div} u$ if $u : \Omega \to \mathbb{R}^3$ is smooth.

   (c) $u \times \text{curl}u = \dfrac{1}{2}\nabla(|u|^2) - (u \cdot \nabla)u$ if $u : \Omega \to \mathbb{R}^3$ is smooth.

Problem 2. Let $\Omega \subseteq \mathbb{R}^{\rm n}$ be a bounded smooth domain. Suppose that the map $\psi(\cdot, t) : \begin{array}{ccc} \Omega & \to & \Omega(t) \\ y & \mapsto & x \end{array}$ is a diffeomorphism, and define $\mathrm{J} = \det(\nabla\psi)$ as well as $\mathrm{A} = (\nabla\psi)^{-1}$. Complete the proof of the following identities

$$\sum_{j=1}^{\rm n} \frac{\partial}{\partial y_j}(\mathrm{J}\mathrm{A}_i^j) = 0 \tag{0.2a}$$

$$n \circ \psi = \frac{\mathrm{A}^{\rm T}\mathrm{N}}{|\mathrm{A}^{\rm T}\mathrm{N}|} \tag{0.2b}$$

by the following arguments:

1. Let $u(\cdot, t) : \Omega(t) \to \mathbb{R}^{\rm n}$ be a smooth vector field. Show that

$$\int_{\Omega(t)} \text{div} u \, dx = \sum_{i,j=1}^{\rm n} \int_{\Omega} \mathrm{J}\mathrm{A}_i^j \frac{\partial (u \circ \psi)^i}{\partial y_j}\, dy\,;$$

   thus by the divergence theorem,

$$\begin{aligned} &\int_{\partial\Omega(t)} u \cdot n \, dS_x \\ &= \sum_{i,j=1}^{\rm n} \int_{\partial\Omega} \mathrm{J}\mathrm{A}_i^j (u \circ \psi)^i \mathrm{N}_j \, dS_y - \sum_{i,j=1}^{\rm n} \int_{\Omega} \frac{\partial (\mathrm{J}\mathrm{A}_i^j)}{\partial y_j}(u \circ \psi)^i dy\,. \end{aligned} \tag{0.3}$$

As a consequence, for all $u(\cdot,t):\Omega(t)\to\mathbb{R}^{\rm n}$ vanishing on $\partial\Omega(t)$,

$$\sum_{i,j=1}^{\rm n}\int_\Omega \frac{\partial(\mathrm{J}\mathrm{A}_i^j)}{\partial y_j}(u\circ\psi)^i dy = 0$$

which in turn implies that (0.2a) is valid.

2. By the Piola identity (0.2a), (0.3) implies that

$$\int_{\partial\Omega(t)} u\cdot n\, dS_x = \sum_{i,j=1}^{\rm n}\int_{\partial\Omega} \mathrm{J}\mathrm{A}_i^j (u\circ\psi)^i \mathrm{N}_j\, dS_y \quad \forall\, u(\cdot,t):\Omega(t)\to\mathbb{R}^{\rm n} \text{ smooth.}$$

Therefore, $n\circ\psi /\!/ \mathrm{A}^{\mathrm{T}}\mathrm{N}$ which implies (0.2b).

Part II: Numerical assignments

Problem 3. Write a matlab® program to find all the prime numbers in between 1 and 10000, and print a list of these prime numbers.
Hint: Find a way to justify whether a number can be divisible by another number or not.

Problem 4. In this problem, we discretize the unit square. Let mesh_size be a given positive number which is less than 1, and G be a system of grid lines on the x-y plane in which the distance between two closest parallel lines is mesh_size. Write a matlab® function "genmesh_fdm_square" with input "mesh_size" and outputs "position", "int_ext", "adjacent_pts"; that is, in the M-file one starts with

"function [position, int_ext, adjacent_pts] = genmesh_fdm_square(mesh_size)",

where meanings of the variables are as follows:

1. Let n be the number of intersection points of grid lines of G inside the square $D = [0,1]\times[0,1]$, and m be the number of points where the grid lines of G intersect with ∂D. The first output "position" is an $(n+m)\times 2$ matrix in which each row vector is the coordinates of one of the $(n+m)$ intersection points. Later on we call the point with coordinate "position(j,:)" **the j-th intersection point**, and j is called **the label** of that point.

2. The second output "int_ext" is an $(n+m) \times 1$ column vector whose entries have values 0 or 1. The value of "int_ext(j)", the j-th component of the column vector, is 1 if the j-th intersection point belongs to the interior of the unit disk, and is 0 if the j-th intersection point is on ∂D.

3. Two points are called adjacent if they can be connected by **one segment** of the grid lines with length not larger than mesh_size. The j-th row of the third output "adjacent_pts" is the label of the adjacent intersection points in counter-clockwise order, starting from the right adjacent point. To be more precise, the third output "adjacent_pts" is an $(n+m) \times 4$ matrix in which:

   (a) The value of "adjacent_pts(j,1)", or equivalently the first component of the j-th row of matrix "adjacent_pts", is the label of the adjacent intersection point right to the j-th intersection point. If there is no such point, the value is set to 0.

   (b) The value of "adjacent_pts(j,2)", or equivalently the second component of the j-th row of matrix "adjacent_pts", is the label of the adjacent intersection point above the j-th intersection point. If there is no such point, the value is set to 0.

   (c) The value of "adjacent_pts(j,3)", or equivalently the third component of the j-th row of matrix "adjacent_pts", is the label of the adjacent intersection point left to the j-th intersection point. If there is no such point, the value is set to 0.

   (d) The value of "adjacent_pts(j,4)", or equivalently the fourth component of the j-th row of matrix "adjacent_pts", is the label of the adjacent intersection point below the j-th intersection point. If there is no such point, the value is set to 0.

For example, the following figure is for the case that mesh_size = 0.3. Then n = 9 and m = 16. Suppose that we label the intersection points as shown in the figure above, then

(a) Each row of the position matrix "position" is the coordinates of one of these 25 intersection points.

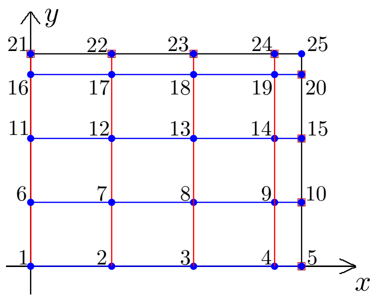

Figure 1: A discretization of the unit square

(b) The matrix "int_ext(1:10,:)", or equivalently the first 10 components of the matrix, is [0 0 0 0 0 0 1 1 1 0]′. By looking at this matrix, we know that 7 of these 10 intersection points locate on the boundary, while the other 3 intersection points locate in the interior of the square.

(c) According to the definition of the matrix "adjacent_pts", we must have

$$\text{adjacent\_pts}(7{:}9,:) = \begin{bmatrix} 8 & 12 & 6 & 2 \\ 9 & 13 & 7 & 3 \\ 10 & 14 & 8 & 4 \end{bmatrix}$$

and

$$\text{adjacent\_pts}(1{:}3,:) = \begin{bmatrix} 2 & 6 & 0 & 0 \\ 3 & 7 & 1 & 0 \\ 4 & 8 & 2 & 0 \end{bmatrix}$$

and etc.

Problem 5. In this problem, we discretize the periodic domain. The discretization of the periodic domain is the same as the one in Problem 4 with one additional information which identifies points. This is done by constructing an additional $(n+m) \times 1$ matrix named "identify_pts" whose value indicates to which intersection point this point is identical. For example, the matrix

$$\begin{aligned} &\text{identify\_pts} \\ &\quad = [1\ 2\ 3\ 4\ 1\ 6\ 7\ 8\ 9\ 6\ 11\ 12\ 13\ 14\ 11\ 16\ 17\ 18\ 19\ 16\ 1\ 2\ 3\ 4\ 1]' \end{aligned}$$

indicates that the 5th, the 21th and the 25th intersection points are identical to the 1st intersection point, while the 10th intersection point is identical to the 6th intersection point, and etc. Write a matlab® program starting with

"function identify_pts = identity_points(periodic_type)"

which does the following:

1. If periodic_type is 0, then identify_pts = [1 2 ⋯ n+m] (which implies that no boundary intersection points are identical).

2. If periodic_type is 1, then identify the right boundary of D to the left boundary of D.

3. If periodic_type is 2, then identify identify the top boundary of D to the bottom boundary of D.

If several intersection points are identical (as in the example above), the corresponding component are set to be the smallest label of these points.

Problem 6. In this problem we discretize the unit disk. Let mesh_size be a given positive number which is less than 1, and G be a system of grid lines on the x-y plane in which the distance between two closest parallel lines is mesh_size. Write a matlab® function "genmesh_fdm_disk" with input "mesh_size" and outputs "position", "int_ext", "adjacent_pts"in which the meanings of the variables are as follows:

1. Let n be the number of intersection points of grid lines of G inside the unit disk, and m be the number of points where the grid lines of G intersect with the unit circle. The first output "position" is an $(n+m) \times 2$ matrix in which each row vector is the coordinates of one of the $(n+m)$ intersection points.

2. The second output "int_ext" is an $(n+m) \times 1$ column vector whose entries have values 0 or 1. The value of "int_ext(j)", the j-th component of the column vector, is 1 if the j-th intersection point belongs to the interior of the unit disk, and is 0 if the j-th intersection point is on the unit circle.

3. The j-th row of the matrix “adjacent_pts” is the label of the adjacent intersection points (defined in Problem 4) in counter-clockwise order, starting from the right adjacent point.

For example, the following figure is for the case that mesh_size = 0.3. Then

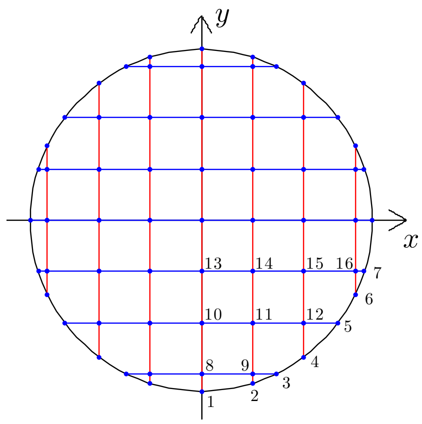

Figure 2: A discretization of the unit square

n = 37 and m = 28. Suppose that one labels part of the intersection points as shown in the figure above, then

(a) The matrix “position(1:16,:)”, or equivalently the first 16 rows of the position matrix, is the coordinates of these 16 intersection points.

(b) The matrix “int_ext(1:16,:)”, or equivalently the first 16 components of the matrix, is [0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1]′.

(c) According to the definition (of the matrix “adjacent_pts”), we must have

adjacent_pts(9,:) = [3 11 8 2];

adjacent_pts(11,:) = [12 14 10 9];

adjacent_pts(12,:) = [5 15 11 4];

and

$$\text{adjacent\_pts(2:6,:)} = \begin{bmatrix} 0 & 9 & 0 & 0 \\ 0 & 0 & 9 & 0 \\ 0 & 12 & 0 & 0 \\ 0 & 0 & 12 & 0 \\ 0 & 16 & 0 & 0 \end{bmatrix}.$$