MAE3456 – MEC3456 LAB 03

Due: 11:00PM (Sharp), Wednesday 29th April 2020 (Mid Week 6)

This lab should be completed INDIVIDUALLY. Plagiarism will result in a mark of zero. Plagiarism includes letting others copy your work and using code you did not write yourself without citing the source. Collaborating with others to discuss algorithms and details of MATLAB syntax and structures is acceptable (indeed encouraged), however you *MUST* write your own MATLAB code. All assignments will be checked using plagiarism-detecting software and similarities in submitted code will result in a human making a decision on whether the similarity constitutes plagiarism.

INSTRUCTIONS

Download **`template.zip`** from Moodle and update the M-Files named **Lab_03_**`Q1.m`**,** **Lab_03_**`Q2.m`**,** etc… with your Lab code. DO NOT rename the M-Files in the template and do not modify **`run_all.m`**. Once you have coded, check your solutions to the questions by running **`run_all.m`** and ensuring all questions are answered as required.

SUBMITTING YOUR ASSIGNMENT

Submit your assignment online using Moodle. You must include the following attachments:

A **ZIP** file (**NOT .rar or any other format**) named in the following way:

Surname_StudentID_Lab_1.zip (e.g. Rudman_23456789_Lab_1.zip)

The zip file should contain the following:

a. All MATLAB m-files for lab tasks: **run_all.m**, **LabNN.m, `Q1a.m`**, **`Q1b.m`**, etc…

b. Any additional MATLAB function files required by your code

c. All data files needed to run the code, **including any input data provided to you**

d. Any hand calculations or written responses asked for - scanned in as a ***SINGLE*** PDF file

YOUR ZIP FILE WILL BE DOWNLOADED FROM MOODLE AND ONLY THOSE FILES INCLUDED IN YOUR SUBMISSION WILL BE MARKED

We will extract (unzip) your ZIP file and mark your lab based on the output of **run_all.m**. and any hand calculations or written responses. It is your responsibility to ensure that everything needed to run your solution is included in your ZIP file. It is also your responsibility to ensure that everything runs seamlessly. Code that does not run will get a mark of zero.

I RECOMMEND AFTER UPLOADINGTO MOODLE, YOU DOWNLOAD YOUR CODE TO A NEW FOLDER AND RUN IT TO CHECK YOU HAVE DONE EVERYTHING CORRECTLY

MARKING SCHEME

This lab is marked out of 80 and full marks is worth 8% of your total Unit mark for the semester. Code will be graded using the following criteria:

1) **`run_all.m`** produces results **automatically** (no additional user interaction needed except where asked explicitly – NOTE, I have included pause commands in run_all so that intermediate answers can easily be viewed by the demonstrators – please don't remove them)

2) Your code produces correct results (printed values, plots, etc…) and is well written.

3) Programming style, efficiency of algorithm and quality of output (figures, tables, written text ...)

ASSIGNMENT HELP

1) You can ask questions in the Discussion Forum on Moodle

2) Hints and additional instructions are provided as comments in the assignment template M-Files

3) Hints may also be provided during lectures

4) The questions have been split into sub-questions. It is important to understand how each sub-question contributes to the whole, but each sub-question is effectively a stand-alone task that does part of the problem. Each can be tackled individually.

5) I recommend you break down each sub-question into smaller parts too, and figure out what needs to be done step-by-step. Then you can begin to put things together again to complete the whole.

6) To make it clear what must be provided as part of the solution, I have used bold italics and a statement that (usually) starts with a verb (e.g. ***Write a function ..., Print the value...***, etc.)

QUESTION 1 **[15 MARKS TOTAL]**

Q1 – The answer to this question should be written as a word or text document or hand written and then scanned, and placed in a SINGLE PDF document that is submitted with your code. NO JPEGs, GIFs, etc.

Background

In this question, you will use Romberg integration by hand, to calculate the volumetric flow in a channel. We considered Romberg integration Lecture 09.

Consider the flow of a fluid between two infinite parallel plates as shown schematically in Figure 1.

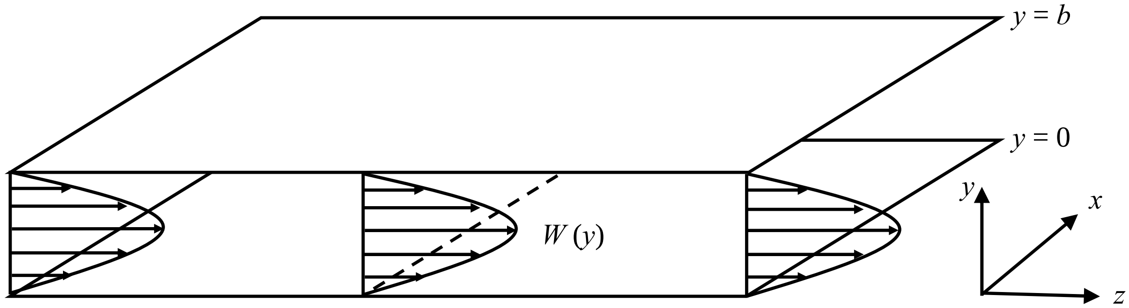

Figure 1 Schematic of channel flow between two infinite flat plates that are a distance b apart. The velocity in the z-direction is $W(y)$.

The solution for $W(y)$ depends on the distance between the plates b, the pressure gradient in the z-direction G and the fluid kinematic viscosity ν. It is given by equation 1:

$$W(y) = \frac{G}{2\mu} y\left(y - b\right) \qquad \text{Equation 1}$$

NOTE: A negative pressure gradient drives the flow in a positive direction.

By integrating equation 1 in the y-direction, the volumetric flow rate (per unit x-width of the channel) can easily be shown to be

$$Q = -\frac{1}{12}\frac{Gb^3}{\mu} \qquad \text{Equation 2}$$

Q1a

Let b=1 AND G/μ= -1. By hand, ***calculate the Trapezoidal rule estimate*** of Q using values of $W(y)$ calculated from equation 1. ***Use*** 1, 2 and 4 intervals and also write the error in each approximation in a table like that below. ***SHOW all your working*** and work to 8 decimal places.

Number of intervals	Trap. Estimate	Error
1		
2		
4		

Q1b

Use Romberg integration (by hand) to calculate the most accurate estimate of the integral using ONLY the 3 estimates from the Trapezoidal rule calculated in Q1a. ***SHOW all your working***, keep 8 decimal places and present the answers from Q1 and all intermediate answers in a table like that below:

Level	Number of intervals	Trap. Estimate	1^{st} Richardson extrapolation	Romberg integral (2^{nd} level R.E.)
1	1			
2	2			
3	4			

Write a statement about what you notice about the error once you have done the 1^{st} Richardson extrapolation (and the final answer). Is this expected, or not? (**HINT**: What kind of function is $W(y)$ and what is the 1^{st} level of Richardson extrapolation equivalent to?)

Q1c

Use 2-point Gauss-Legendre quadrature to estimate the same integral. What result do you get? Is this expected or unexpected?

QUESTION 2 [15 MARKS TOTAL]

Background

Power-law fluids are fluids whose dynamic viscosity is a (power-law) function of the fluid strain rate. Such fluids occur in many practical applications such as fine particle suspensions, certain polymers and food stuffs. For a plane shear flow like that shown in Figure 1, the dynamic viscosity of a power-law fluid is written

$\mu = K\left|\frac{\partial W}{\partial y}\right|^{n-1}$, where K is a parameter called the consistency and n is the flow index which determines the type of fluid. When $0<n<1$ the fluid is called a shear thinning fluid, n=1 is a Newtonian fluid and n>1 is a shear thickening fluid. It can be shown that the velocity profile for such a fluid in a channel driven by a pressure gradient of $-G$ is given by

$$W(y) = \frac{n}{n+1}\left(\frac{G}{K}\right)^{\frac{1}{n}}\left\{\left(\frac{b}{2}\right)^{\frac{n+1}{n}} - \left(\left|y - \frac{b}{2}\right|\right)^{\frac{n+1}{n}}\right\} \qquad \text{Equation 3}$$

(By setting n=1 in Eqn 3 you will see we recover eqn 1 where $K=\mu$.)

Q2a

i) Write a MATLAB function that calculates an integral of a specified function using the recursive trapezoidal rule ***STARTING*** with the estimate for just one segment. The function should keep halving the interval until such time as the difference between 2 estimates is less than a tolerance specified in the variable `tol`.

The function header MUST be

```
function [S_T,nlev] = R_Trap(f,xrange,tol)
```

Here, the input parameters are

- **f** – Function handle of the function to be integrated.
- **xrange -** a 2-vector of x-values that specifies the limits of integration (x(1) is LHS, x(2) is RHS)
- **tol** – Error tolerance

The output parameters are

- **S_T -** the Trapezoidal rule estimate of the integral
- **nlev –** of times the initial interval (xrange) has been halved.

Q2b

Modify the template **Lab_03_Q2.m** to include code that does the next part of the question.

Calculate the integral of equation 3 over the range [0,b] using R_Trap – i.e., calculate the volumetric flow rate of a power law fluid in a channel. Let G=1, K=1 and b=1 for n=0.5.

Choose error tolerances of 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}.

Plot the number of segments as a function of the error tolerance in a log-log plot.

Write a few sentences to the command window describing what you observe AND why you think this is the case. (**HINT:** Based on your knowledge of the global error in the trapezoidal rule, how does error scale with step size, and how is step size related to n?)

Q2c

Continue to modify **Lab_03_Q2.m** to include code that does the next part of the question.

Choose a tolerance of 10^{-6} and the same values of G, K and b as in part **Q2b**, and ***plot the volumetric flow rate*** as a function of the flow index n given by (0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6).

(***WARNING:*** If you change the value of a constant that is part of an anonymous function definition, you ***MUST*** redefine the anonymous function.)

Write a few sentences to the command window describing what you observe. Is there an obvious relationship between flow rate and n? If so try and describe it.

QUESTION 3 [20 MARKS TOTAL]

Q3a

Write a MATLAB function Romberg that performs Romberg integration as discussed in Lecture 9. The function header should look like

```
function [S_R,nlev] = Romberg(f, xrange, tol)
```

Here, the input parameters are

- **f** – The function to be integrated.
- **xrange -** a 2-vector of x-values that specifies the limits of integration (x(1) is LHS, x(2) is RHS)
- **tol** – Error tolerance (i.e. difference between 2 successive Romberg estimates).

The output parameters are

- **S_R -** the Romberg estimate of the integral
- **nlev –** number of levels needed to attain error tolerance

Q3b

Modify the template **Lab_03_Q3.m** to include code that does the next part of the question.

For the same power-law flow integral as in **Q2b** (G=1, K=1, b=1 and n=0.5), ***determine the number of segments*** required to obtain an error less than 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}, 10^{-9}, 10^{-10}, 10^{-11}, 10^{-12} and ***print this*** to the command window for each tolerance.

Then ***Plot the*** number of segments vs error tolerance in a log-log plot.

The exact flow rate for n=0.5 and these parameters is Q=0.03125. ***Calculate the actual error*** (not just the estimate based on subsequent Romberg estimates) and ***plot the*** actual error versus error tolerance in a log-log plot. To help understand these results, ALSO include the line that is "actual error = error tolerance" (i.e. y=x).

You will notice some quite odd behavior in the two plots in this question. ***Write*** a few sentences to the command window describing what you see, including a suggested explanation for this behavior. (**HINT:** Is the actual error bigger or less than the error tolerance in the previous figure?)

Q3c

Compare the number of segments required to obtain a given accuracy when using Romberg integration and the Trapezoidal rule. ***Write*** a few sentences to the command window outlining your findings and include a statement that indicates which method is more efficient (i.e., has a smaller operation count). Refer back to your results from Q2 if you like.

QUESTION 4 [15 MARKS TOTAL]

Q4a

Write a function that performs n-point Gauss-Legendre quadrature over an interval [a,b]. The function header MUST be

```
function [Integral] = GL_Quad(f,n,a,b)
```

where

- **f** is a function handle of the function
- **n** is the number of points required in the quadrature.
- **a, b** is the interval over which the integration occurs

The Gauss-Legendre weights and quadrature points can be calculated in the MATLAB function `GaussLegendre.m`. that is supplied as part of the template. You will need to use this function *inside* GL_Quad. (To use GaussLegendre, the command [x A] = GaussLegendre(n) will return the quadrature points in x and the weights in A.

Q4b

Modify the script **Lab_04_Q2.m** so that it does the following:

1. ***Prints a short statement*** to the command window indicating HOW you could verify that your GL_Quad function is working correctly
2. ***Prints evidence*** to the command window that it does work using a suitable test case of your choosing
3. ***Calculates the same flow integral*** as that in Questions 2 and 3 using n=2,3, … 20-point Gaussian quadrature, a flow index n=0.5 and the same values for G, K and b.

Q4c

The analytic solution for the flow rate when n=0.5 is Q=0.03125. Define the error for this case as error = abs(Calculated integral – exact value).

1. ***Plot the error*** as a function of n in loglog coordinates.
2. ***Write a short statement*** to the command window that states what you observe and what you think the relationship between error and n is.

QUESTION 5 [15 MARKS TOTAL]

In this question you will be doing a two-dimensional integral.

Background

Consider the flow of a fluid in a rectangular micro-fluidic channel as shown in isometric view in Figure 2 with contours of axial velocity $W(x,y)$.shown in an end view (looking in the z-direction) in Figure 3.

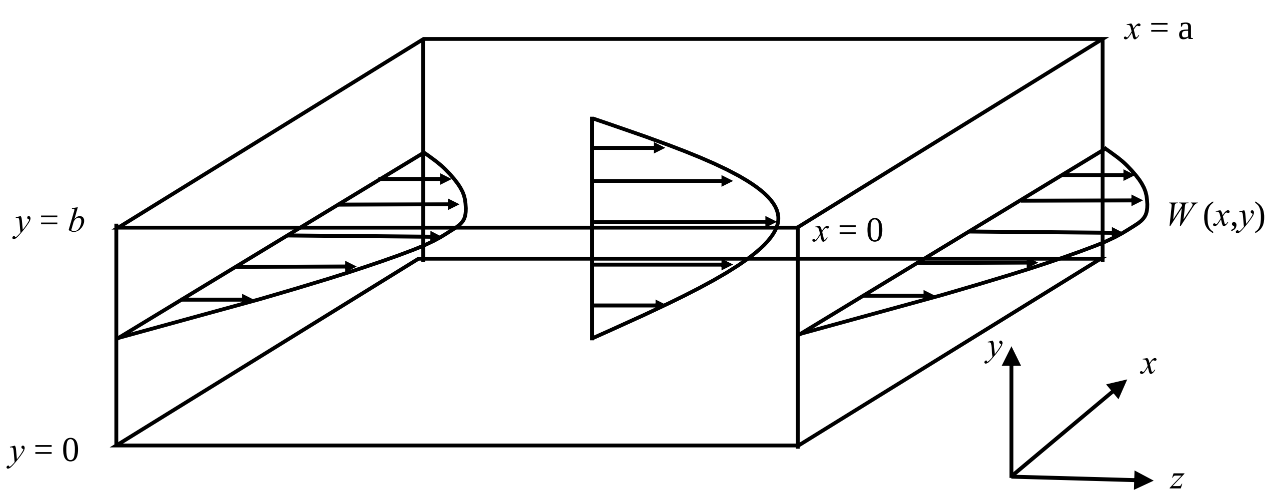

Figure 2 Schematic of 2D microfluidic micro-channel The velocity in the z-direction is $W\,(x,y)$.

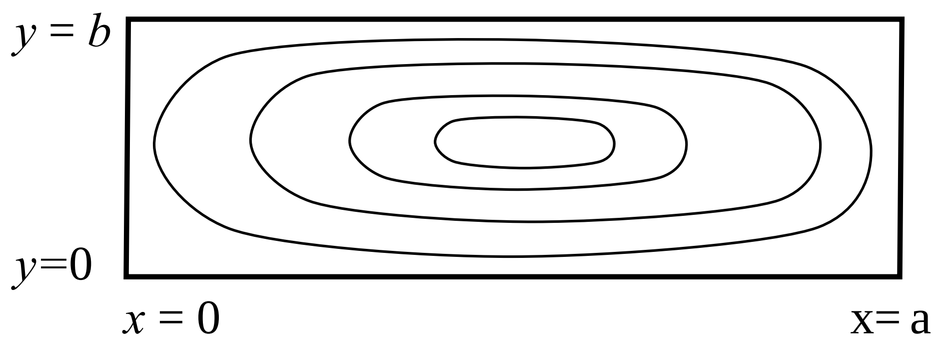

Figure 3 Schematic of contours of axial velocity $W\,(x,y)$ in a microfluidic channel. The maximum velocity is in the centre of the channel, and the velocity at the walls is zero.

The solution for $W\,(x,y)$ depends on the size of the channel (a, b), the pressure gradient in the z-direction G and the fluid dynamic viscosity μ. It is given by equation 4:

$$W(x,y) = \frac{a^2 G}{\mu}\frac{16}{\pi^4}\sum_{n=1,3,5\ldots}\left(\sum_{m=1,3,5\ldots}\left(\frac{\sin\left(\frac{n\pi x}{a}\right)\sin\left(\frac{m\pi y}{b}\right)}{nm\left(n^2+m^2/b^2\right)}\right)\right) \qquad \text{Equation 4}$$

The aspect ratio of the channel is written as $\beta = b\,/\,a$. You are provided with a function that calculates $W\,(x,y,a,b)$ in the file Wxy.m. (It assumes G=1, μ=1). **NOTE:** this function will ONLY calculate the value at a single point. If you feed it a vector of points you will get an error.

Q5a

Write a function that performs 2D Gauss-Legendre quadrature over x $\in$ [0,a] and y $\in$ [0,b]. The function header MUST be

```
function [Integral] = Integrate2(f,n,xr,yr)
```

where the input parameters are

- **f** - the function handle for the function you want to integrate
- **n** – the order if quadrature (use the same in both directions)
- **xr -** a vector for the x-range of integration [xmin, xmax]
- **yr -** a vector for the y-range of integration [ymin,ymax][
-

Q5b

You will use your function Integrate2 to ***integrate*** Equation 5 in x and y to calculate the volumetric flow rate in the channel , $Q = \int_{y=0}^{b}\int_{x=0}^{a} W\left(x,y\right)dx\,dy$.

Choose a single value for n, and ***print*** a sentence to the command window justifying why your chosen value is acceptable.

Use the following parameters: G=1, μ=1 and aspect ratios of β = 1, 0.5, 0.25, 0.125 and 0.0625. We want to keep the cross sectional area of the channel the same for different aspect ratios (we will choose an area of unity), which means that our choice of a and b is $a = \sqrt{1/\beta}$ $\quad b = \sqrt{\beta}$.

(**NOTE:** Again, you will need to define the function handle for each new beta individually. For example, you can define W = @(x,y) Wxy(x,y,a,b), where Wxy is the function provided to you in the previous example.)

Print the volumetric flow rate for each aspect ratio to the command window.

Plot the volumetric flow rate as a function of aspect ratio.

Print a statement to the command window that answers the following question

For a given pressure gradient, what aspect ratio channel will provide the highest flow rate.

Poor Programming Practices [-10 Marks]

(Includes, but is not limited to, poor coding style or insufficient comments or unlabeled figures, etc.)

(END OF LAB)