Analysis of 2-D Lid-Driven Cavity Problem

Aakash Yadav[1]

[1]**Indian Institute of Technology, Tirupati**

April 2019

1 Introduction

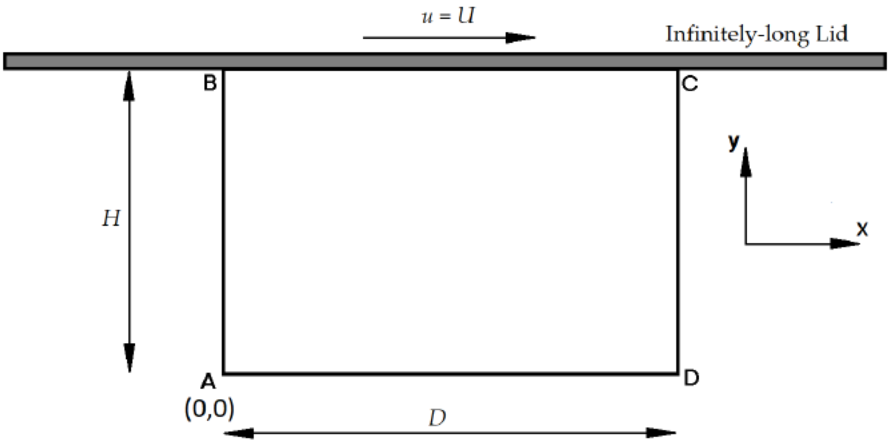

Figure 1: scaling of the WF law by $A\left(r, \mu^{*}\right)$

In this project, we will analyze the unsteady, viscous, incompressible,isothermal, two-dimensional, laminar flow of Newtonian fluid in a cavity covered with a lid. Consider a rectangular cavity ABCD of dimensions shown in Figure 1. AB, CD, and AD are rigid walls, whereas BC is open. The cavity is of length D and width H , in the x-y plane, as shown in Figure 1.1. The aspect ratio is defined as $R = H/D$. The Reynolds number is defined based on the velocity scale U and the length scale D. Acceleration due to gravity acts in the negative-z direction. The top of the cavity, BC, is covered with an infinitely-long rigid lid. Initially ($t \leq 0$), the fluid inside the cavity is at rest. At time $t > 0$, the lid is set in motion to the right with a constant velocity U. You are to analyze the fluid flow inside the cavity for various conditions using proper governing equations, boundary conditions, initial conditions, and numerical schemes.

2 Governing Equations

The Wiedemann Franz Law (1853) states that the ratio of the electronic contribution of the thermal conductivity κ to the electrical conductivity σ of a metal is proportional to the temperature T . [1]

Continuity equation for 2-D incompressible isothermal flow

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{1}$$

Stream function

$$\omega = -\left[\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2}\right] = -\nabla^2 \psi \tag{2}$$

Navier-Stokes equations

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial x} + g_x + \nu\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right] \tag{3}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial y} + g_y + \nu\left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right] \tag{4}$$

Vorticity relation

$$\frac{\partial \omega_z}{\partial t} + u\frac{\partial \omega_z}{\partial x} + v\frac{\partial \omega_z}{\partial y} = \nu\left[\frac{\partial^2 \omega_z}{\partial x^2} + \frac{\partial^2 \omega_z}{\partial y^2}\right] \tag{5}$$

For the sake of contnuity we will drop the subscript from the ω_z herafter.

3 Courant–Friedrichs–Lewy analysis

In mathematics, the Courant–Friedrichs–Lewy (CFL) condition is a necessary condition for convergence while solving certain partial differential equations (usually hyperbolic PDEs) numerically. It arises in the numerical analysis of explicit time integration schemes, when these are used for the numerical solution. As a consequence, the time step must be less than a certain time in many explicit time-marching computer simulations, otherwise the simulation produces incorrect results. The condition is named after Richard Courant, Kurt Friedrichs, and Hans Lewy who described it in their 1928 paper [3]. We have the equation as

$$\frac{\partial \omega}{\partial t} + u\frac{\partial \omega}{\partial x} + v\frac{\partial \omega}{\partial y} = \nu\left[\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right] \tag{6}$$

We shall now use the Fourier wave form to bring out the required conditions. The amplification is easily identified using the Fourier wave form, moreover the waveform remains preserved.

$$\omega = \sum_m A^n(t)e^{\hat{i}(k_m x + k_n y)} \tag{7}$$

Here we can assume the wavenumber of the wave in both x and y direction to be same i.e. $k_n = k_m$

$$\omega = A^n e^{\hat{i}(ik_m \Delta x + jk_n \Delta y)} \tag{8}$$

$$\omega^n_{i,j} = A^{n+1} e^{\hat{i}(ik_m \Delta x + jk_n \Delta y)} \tag{9a}$$

$$\omega^{n+1}_{i,j} = A^{n+1} e^{\hat{i}(ik_m \Delta x + jk_n \Delta y)} \tag{9b}$$

$$\omega^n_{i+1,j} = A^n e^{\hat{i}((i+1)k_m \Delta x + jk_n \Delta y)} \tag{9c}$$

$$\omega^n_{i-1,j} = A^n e^{\hat{i}((i-1)k_m \Delta x + jk_n \Delta y)} \tag{9d}$$

$$\omega^n_{i,j+1} = A^n e^{\hat{i}(ik_m \Delta x + (j+1)k_n \Delta y)} \tag{9e}$$

$$\omega^n_{i,j-1} = A^n e^{\hat{i}(ik_m \Delta x + (j-1)k_n \Delta y)} \tag{9f}$$

Substituting this in our main equation as described above and simplification results in

4 Initial Conditions and Boundary Conditions

At time $t = 0$ everything is at rest and hence all the values are zero. At the moment $t \geq 0$ the lid will be moving at speed $u = U$, which will try to set the fluidin motion. We have the following boundary conditions for $t \geq 0$:
No slip condition will result in zero velocity along the wall tangent

$$u(x, 0) = 0 \tag{10a}$$

$$v(D, y) = 0 \tag{10b}$$

$$u(x, H) = U \tag{10c}$$

$$u(0, y) = 0 \tag{10d}$$

The no penetration condition that the walls of the cavity are impervious results into

$$v(x, 0) = 0 \tag{11a}$$

$$u(D, y) = 0 \tag{11b}$$

$$v(x, H) = 0 \tag{11c}$$

$$u(0, y) = 0 \tag{11d}$$

5 Non-dimensionalization of the Governing Equations

It is convenient numerically to make the equations for y and w dimensionless. This means we need to introduce the appropriate scalings for the dimensionless variables.

$$\hat{x} = \frac{x}{D}, \hat{y} = \frac{y}{H}, \hat{t} = \frac{U}{D}t, \hat{u} = \frac{u}{U}, \hat{v} = \frac{v}{V_{ref}} \tag{12}$$

Non-dimensionalising Equation 1 results into

$$\frac{U}{D}\frac{\partial \hat{u}}{\partial \hat{x}} + \frac{V_{ref}}{H}\frac{\partial \hat{v}}{\partial \hat{y}} = 0 \tag{13a}$$

$$\frac{U}{D} \sim \frac{V_{ref}}{H} \tag{13b}$$

$$V_{ref} = \left(\frac{H}{D}\right) U = RU \tag{13c}$$

$$\hat{\psi} = \frac{\psi}{UD}, \hat{\omega} = \frac{\omega D}{U} \tag{14}$$

Non-dimensionalising the stream function equation (2)

$$\frac{\hat{\omega} U}{D} = -\left[\frac{\partial^2(\hat{\psi} UD)}{\partial(\hat{x}D)^2} + \frac{\partial^2(\psi \hat{U} D)}{\partial(\hat{y}H)^2}\right] \tag{15a}$$

$$\hat{\omega} = -\left[\frac{\partial^2 \hat{\psi}}{\partial \hat{x}^2} + \frac{1}{r^2}\frac{\partial^2 \hat{\psi}}{\partial \hat{y}^2}\right] \tag{15b}$$

Non-dimensionalising the vorticity function equation (5)

$$\frac{\partial\left(\frac{U}{D}\hat{\omega_z}\right)}{\partial\left(\frac{D}{U}\hat{t}\right)} + \hat{u}U\frac{\partial\left(\frac{U}{D}\hat{\omega_z}\right)}{\partial(\hat{x}D)} + \frac{UH}{D}\hat{v}\frac{\partial\left(\frac{U}{D}\hat{\omega_z}\right)}{\partial(\hat{y}H)} = \nu\left[\frac{\partial^2\left(\frac{U}{D}\hat{\omega_z}\right)}{\partial(\hat{x}D)^2} + \frac{\partial^2\left(\frac{U}{D}\hat{\omega_z}\right)}{\partial(\hat{y}H)^2}\right] \tag{16a}$$

$$\frac{\partial \hat{\omega_z}}{\partial \hat{t}} + \hat{u}\frac{\partial \hat{\omega_z}}{\partial \hat{x}} + \hat{v}\frac{\partial \hat{\omega_z}}{\partial \hat{y}} = \frac{\nu}{UD}\left[\frac{\partial^2 \hat{\omega_z}}{\partial \hat{x}^2} + \frac{1}{r^2}\frac{\partial^2 \hat{\omega_z}}{\partial \hat{y}^2}\right] \tag{16b}$$

Also we have

$$u = \frac{\partial \psi}{\partial y}, v = -\frac{\partial \psi}{\partial x} \tag{17}$$

Non-dimensionalisation of the above equation results into

$$\hat{u} = \frac{1}{r}\frac{\partial \hat{\psi}}{\partial \hat{y}}, \hat{v} = -\frac{1}{r}\frac{\partial \hat{\psi}}{\partial \hat{x}} \tag{18}$$

Substituting equation (17) in equation (16b) yields

$$\frac{\partial \hat{\omega_z}}{\partial \hat{t}} + \frac{1}{r}\left[\frac{\partial \hat{\psi}}{\partial \hat{y}}\frac{\partial \hat{\omega_z}}{\partial \hat{x}} - \frac{\partial \hat{\psi}}{\partial \hat{x}}\frac{\partial \hat{\omega_z}}{\partial \hat{y}}\right] = \frac{1}{Re}\left[\frac{\partial^2 \hat{\omega_z}}{\partial \hat{x}^2} + \frac{1}{r^2}\frac{\partial^2 \hat{\omega_z}}{\partial \hat{y}^2}\right] \tag{19}$$

where, $Re = \frac{UD}{\nu}$
Deriving the boundary conditions in the form of vorticity

$$\hat{\omega}(\hat{x}, 0) = -\frac{1}{r^2}\left(\frac{\partial^2 \hat{\psi}}{\partial \hat{y}^2}\right)_{\hat{y}=0} \tag{20a}$$

$$\hat{\omega}(\hat{x}, 1) = -\frac{1}{r^2}\left(\frac{\partial^2 \hat{\psi}}{\partial \hat{y}^2}\right)_{\hat{y}=1} \tag{20b}$$

$$\hat{\omega}(0, \hat{y}) = -\left(\frac{\partial^2 \hat{\psi}}{\partial \hat{x}^2}\right)_{\hat{x}=0} \tag{20c}$$

$$\hat{\omega}(1, \hat{y}) = -\left(\frac{\partial^2 \hat{\psi}}{\partial \hat{x}^2}\right)_{\hat{x}=1} \tag{20d}$$

Deriving the boundary conditions in the form of vorticity. For the bottom surface $\hat{u} = \hat{v} = 0$, substituiting this equation (18) yields

$$\hat{\psi}_{(\hat{y}=0)} = c_1 \tag{21}$$

Where c_1 is the integration constant. Similarly we can obtain the equations for the other walls as

$$\hat{\psi}_{(\hat{x}=0)} = c_2, \hat{\psi}_{(\hat{x}=1)} = c_3 \tag{22}$$

For the top wall

$$\hat{u} = U = \frac{1}{r}\frac{\partial \hat{\psi}}{\partial \hat{y}} \Rightarrow \hat{\psi} = rUH + c_4 = c_5 \tag{23}$$

$$\hat{\psi}_{(\hat{y}=1)} = c_5 \tag{24}$$

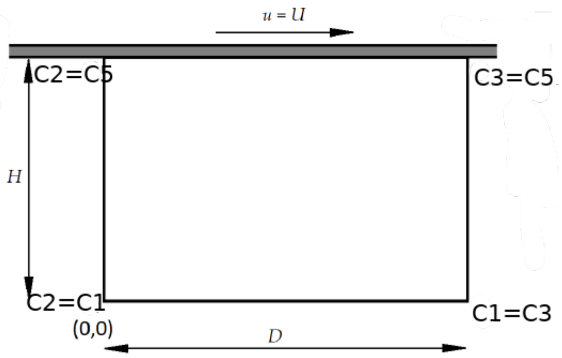

Figure 2: scaling of the WF law by $A\,(r, \mu^*)$

We have $c_1 = c_2 = c_3 = c_5 = c$ because of the continuity at the corners as shown in the Figure 2. Without any loss of generality we can set $c = 0$ as ψ is a relative term. Hence

$$\hat{\psi}_{(y=0)} = \hat{\psi}_{(y=1)} = \hat{\psi}_{(x=0)} = \hat{\psi}_{(x=1)} = 0 \tag{25}$$

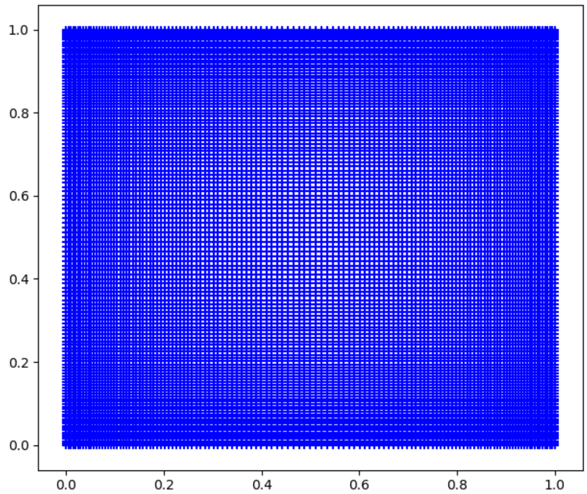

Figure 3: A 128 × 128 grid generated by the program

6 Discretization Schemes

In the analysis that follows we will drop the 'hat' notation for dimensionless variables for convienance. We will first use the usual FTCS scheme and a grid that is not equally spaced as shown in Figure 3. Derive the orders of accuracies of this scheme for both time and space. The values of Δx and Δy are not constant. You must use an appropriate stretching function so that the number of grid points vary in such a way that more grid points are concentrated closer to walls and the lid. Why is this necessary? Comment on this! Do a Von-Neumann stability analysis to find out the CFL criterion and the maximum time step to be used in your calculations. You may notice that $\Delta x, \Delta y, u, v$ are going to vary throughout the domain. So, the grid Courant number is also going to vary for every node, and every time step. So your code/program/algorithm must be adaptive; i.e, it must be able to choose the correct values of Δt so that the CFL criterion is satisfied at all times and at all locations in the domain.
Discretiztion of equation 15b using forward time and central differencing scheme. Second order accurate in space.

$$\frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{(\Delta x)^2} + \frac{1}{r^2}\frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{(\Delta y)^2} = -\omega_{zij} \tag{26}$$

Von-Newmann stability analysis for the above equation -
Discretiztion of equation 19 using forward time and central differencing scheme. First order time accurate and second order space accurate.

$$\begin{aligned}\frac{\omega^{n+1}_{i,j} - \omega^{n}_{i,j}}{\Delta t} + \frac{1}{r}\left[\left(\frac{\psi_{i,j+1} - \psi_{i,j-1}}{2\Delta y}\right)\left(\frac{\omega_{i+1,j} - \omega_{i-1,j}}{2\Delta x}\right) - \left(\frac{\psi_{i+1,j} - \psi_{i-1,j}}{2\Delta x}\right)\left(\frac{\omega_{i,j+1} - \omega_{i,j-1}}{2\Delta y}\right)\right] \\ = \frac{1}{Re}\left[\left(\frac{\omega_{i+1,j} - 2\omega_{i,j} + \omega_{i-1,j}}{(\Delta x)^2}\right) + \frac{1}{r^2}\left(\frac{\omega_{i,j+1} - 2\omega_{i,j} + \omega_{i,j-1}}{(\Delta y)^2}\right)\right]\end{aligned} \tag{27}$$

Von-Neutmann stability analysis for the above equation -

7 Algorithm

The Fermi-Dirac integrals play very improtant role in the study of semiconductors appear frequently in semiconductor problems. It is thus a topic of special interest among physicist working in this field.

No.	Steps
1	Initialize u and v values
2	Define the vorticity equation
3	Solve Poisson equation for stream function
4	Solve vorticity transport equation at a forward time step $t + \Delta t$
5	Solve the Poisson equation for stream function at $t + \Delta t$
6	Find u and v using the vorticity function equation
7	Check if the CFL criteria is satisified, if yes proced to next step
8	Else change the time step size and repeat the loop

8 Results

The Fermi-Dirac integrals play very improtant role in the study of semiconductors appear frequently in semiconductor problems. It is thus a topic of special interest among physicist working in this field.

8.1 Grid Independence Studies

The formulation for obtaining the minimum lattice thermal conductivity is given by approach developed by Cahill [2]. To minimise the Phonon thermal conductivity, the integrand has been extremised and simplified yielding solution in the form of offset log function. The plot for the integrand essentially Planck's blackbody radiation function has also been shown in Figure 2.

8.2 Streamlines for Varying R and Re

The efficiency of the thermoelectric material is directly proportional to the electrical conductivity of the material. Maximizing the same is of the utmost importance inorder to increase its efficiency. We have the following expression for the electrical . [4]

8.3 Profiles for Steady Flow

Exact Fermi Dirac Integral expressions can be very helpful in generalizing the Wiedemann Franz Law.

9 Discussion

Exact Fermi Dirac Integral expressions can be very helpful in generalizing the Wiedemann Franz Law. Exact analytic expressions of the same will greatly assist and equip the researchers in the new material design processes. Electronic thermal conductivity, κ_e and minimum lattice thermal conductivity $\kappa_{l,min}$ have exact analytic expressions and we have obtained very interesting forms of solutions while maximising the two. More recent observations on the influence of anharmonicity on $\kappa_{l,min}$ suggest that the Polylogarithms and Lambert W can have more interesting applications.

Appendix

The below code can be used for plotting the scaling factor A as shown in Figure 1.

```
"""
Reference(s):
http://caefn.com/cfd/hyperbolic-tangent-stretching-grid
"""
from sympy import *
import matplotlib.pyplot as plt
import numpy as np
from array import *
```

```python
from scipy.sparse import *

# TODO: fix the grid function for odd value of grid elements
def grid(nx,ny, gama):
    # use ty tx to change number of elements
    tx=(2)/((nx+1)+1)
    ty=(2)/((ny+1)+1)
    x=[]
    y=[]
    nx=[]
    ny=[]

    # x elements on left half
    for i in np.arange(0., 1., tx):
        nx.append(i)
    for j in nx:
        x.append(1-(np.tanh(gama*(1-(2*j)/len(nx))))/(np.tanh(gama)))

    # y elements on right half
    for i in np.arange(0., 1., ty):
        ny.append(i)
    for i in ny:
        y.append(1-(np.tanh(gama*(1-(2*i)/len(ny))))/(np.tanh(gama)))

    # mirroring x and y elements for the right half
    for i in range(len(nx)-1):
        x.append(x[len(nx)+i-1]+x[len(nx)-(i+1)]-x[len(nx)-(i+2)])
    for i in range(len(ny)-1):
        y.append(y[len(ny)+i-1]+y[len(ny)-(i+1)]-y[len(ny)-(i+2)])

    xd=[]
    yh=[]
    for i in x:
        xd.append(i/(x[len(x)-1]))
    for i in y:
        yh.append(i/(y[len(y)-1]))
    return(xd,yh)

def gridPlot(c,d):
    for i in c:
        for k in d:
            plt.scatter(i,k,color='black',marker='+')
    plt.show()

def spacing(test_list):
    res = [test_list[i + 1] - test_list[i] for i in range(len(test_list)-1)]
    return(res)

def uGrid(nx,ny):
    l1=[]
    l2=[]
    for i in range(nx+1):
        l1.append(1/(nx+1))
    for i in range(ny+1):
        l2.append(1/(ny+1))
    return(l1,l2)

## print(uGrid(10,10))
```

```
# print(grid(10,10,3))
# # NOTE: call from any program using
# # from gridGen import grid,gridPlot
# x,y=grid(10,10,3)    # accepts (nx, ny, game)
# gridPlot(grid(10,10,3)[0],grid(10,10,3)[1])
```

References

[1] N.W. Ashcroft and N.D. Mermin. *Solid State Physics*. Saunders College, Philadelphia, 1976.

[2] David G. Cahill, S. K. Watson, and R. O. Pohl. Lower limit to the thermal conductivity of disordered crystals. *Phys. Rev. B*, 46:6131–6140, Sep 1992.

[3] Friedrichs K. Courant R. and Lewy H. Math. Ann. Über die partiellen differenzengleichungen der mathematischen physik. *Springer-Verlag*, 1928.

[4] Muralikrishna Molli, Venkataramaniah Kamisetti, and Sree Ram Valluri. The polylogarithm and the lambertwfunctions in thermoelectrics. 89:1171–1178, 11 2011.