Simulation of Fluid Flow in a 2-D Lid Driven Cavity

- Soham Shirodkar 19110167
- Deepak Singh 19110115
- Rudresh Rai 19110162
- Digvijay Mali 19110013
- Shridhar Pawar 19110101

Overview

1. Definition of problem & its relevance
2. Development of Mathematical model
3. Numerical analysis methods
4. Description of Algorithm
5. Results and Discussions
6. Conclusion
7. References

Definition of model & its significance

Domain fig

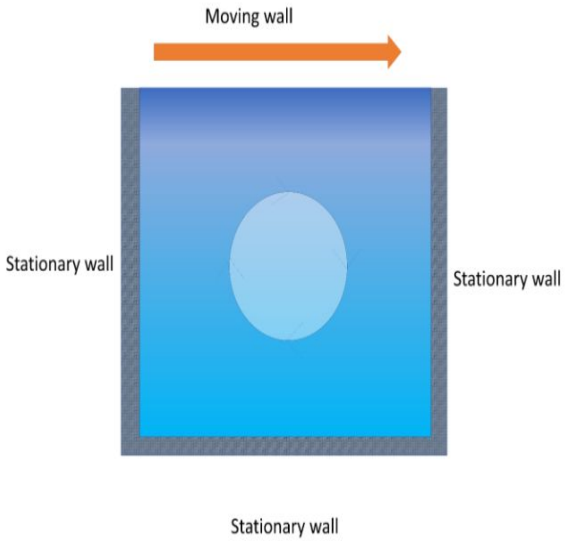


Figure 1

We will analyze the Velocity / Vorticity and Stream Function

Fluid is contained in the 2-D Domain with walls as boundaries

The topmost wall is moving with a velocity U at time t=0 and it imparts the velocity on the fluid

Due to No-slip conditions, the fluid layers starts moving

Navier Stokes Equations

Control volumes - integral part of Fluid Mechanics

Such CV analyses can help predict the performance of complex systems

This project gives a fundamental idea about CFD

Development of Mathematical Model

Navier Stokes

Continuity

Vorticity - Velocity - Stream- Function relations

$\mathring{u} = u / U$

$\dot{x} = x / L$

$\dot{t} = t\,U / L$ $\quad \dot{p} = p/\rho U^2$

For right & left $\omega = -\dfrac{\partial^2 \Psi}{\partial x^2}$

For top & bottom $\omega = -\dfrac{\partial^2 \Psi}{\partial y^2}$

Governing Eqns → **Assumptions** → **Non-Dimensionalize** → **Boundary Conditions**

Finite Dimensions
No variation in Z

Newtonian Planar
Incompressible fluid

No Body forces

Parameters (ρ,μ) are constant

$u_{top} = U_{wall}$

v_{top} & $v_{bottom} = 0$

u_{right} & $u_{left} = 0$

Governing Equations

1. Navier stokes

$$\frac{\partial v}{\partial t} + (v.\nabla)v = -\frac{\nabla p}{\rho} + v\nabla^2 v + g$$

2. Continuity

$$\nabla.v = 0$$

3. Vorticity - velocity

$$\omega = \nabla \times v$$

4. Stream Function - Vorticity relations

$$\frac{\partial^2 \Psi}{\partial^2 x} + \frac{\partial^2 \Psi}{\partial^2 y} = -w$$

Specific Boundary Conditions

Using Taylor Series expansion for Stream function $\Psi_{i,j=2} = \Psi_{i,j=1} + h\frac{\partial \Psi_{i,j=1}}{\partial y} + \frac{h^2}{2}\frac{\partial^2 \Psi_{i,j=1}}{\partial^2 y} + O(h^3)$

Using - $\omega_{wall} = -\frac{\partial^2 \Psi_{i,j=1}}{\partial y^2}$ and $U_{wall} = \frac{\partial \Psi_{i,j=1}}{\partial y}$

We get $\Psi_{i,j=2} = \Psi_{i,j=1} + U_{wall}h - \omega_{wall}\frac{h^2}{2} + O(h^3)$

To get the equation in terms of vorticity, $\omega_{wall} = (\Psi_{i,j=1} - \Psi_{i,j=2})\frac{2}{h^2} + U_{wall}\frac{2}{h} + O(h)$

These conditions are written for all the respective walls

Numerical analysis methods

These are the main equations that we will solve to ultimately obtain the desired velocity of the grid points-

$$\frac{\partial \omega}{\partial t} = -\frac{\partial \Psi}{\partial y}\frac{\partial \omega}{\partial x} + \frac{\partial \Psi}{\partial y}\frac{\partial \omega}{\partial y} + \frac{1}{Re}\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) \tag{5.1}$$

$$\frac{\partial^2 \Psi}{\partial^2 x} + \frac{\partial^2 \Psi}{\partial^2 y} = -w \tag{5.2}$$

We use discretization method:

$$\frac{\partial f(x)}{\partial x} = \frac{f(x+h) - f(x-h)}{2h} + \frac{\partial^3 f(x)}{\partial x^3}\frac{h^2}{6} + \dots$$

$$\frac{\partial^2 f(x)}{\partial x^2} = \frac{f(x+h) - 2f(x) + f(x-h)}{2h} + \frac{\partial^4 f(x)}{\partial x^4}\frac{h^2}{12} + \dots$$

$$\frac{\partial f(t)}{\partial t} = \frac{f(t+\Delta t) - f(t)}{\Delta t} + \frac{\partial^2 f(t)}{\partial t^2}\frac{\Delta t}{2} + \dots$$

to convert the equations 5.1 & 5.2 to 5.3 and 5.4

$$\omega_i, j^{n+1} = \omega_{i,j}^{\ n} - \Delta t[\frac{(\Psi i, j + 1^n - \Psi^n_{i,j-1})}{2h} \frac{(\omega^n_{i+1,j} - \omega^n_{i-1,j})}{2h} + \frac{(\Psi^n_{i+1,j} - \Psi^n_{i-1,j})}{2h} \frac{(\omega^n_{i,j+1} - \omega^n_{i,j-1})}{2h} + \frac{1}{Re}(\frac{\omega^n_{i+1,j} + \omega^n_{i,j+1} - 4\omega^n_{i,j} + \omega^n_{i+1,j} + \omega^n_{i,j-1}}{h^2})] \qquad (5.3)$$

(Where Re is the Reynolds no , 100 here)

$$\Psi_{i,j} = \frac{\omega_{i,j}h^2 + \omega_{i,j+1} + \omega_{i,j-1} + \omega_{i-1,j}}{4} \qquad (5.4)$$

Now we have Equation 5.3 & 5.4 to calculate the 2D matrix of ω and ψ which iterate to change the value until the error condition is justified using Gauss-Seidel Method.
Finally we have met the error condition, we have our matrix of ω and ψ .

Our task was to evaluate the Velocity of the fluid at each point of the described grid and is given by

$$u = \frac{\partial \Psi}{\partial y} \qquad v = -\frac{\partial \Psi}{\partial x}$$

Using Central difference method to find X and Y components in the Velocity - Stream Function Relations -

$$u = \frac{\Psi_{i,j+1} - \Psi_{i,j-1}}{2h} \quad (5.5) \qquad v = \frac{-\Psi_{i+1,j} + \Psi_{i-1,j}}{2h} \quad (5.6)$$

We implement discretization on the boundary condition to obtain:

Top wall

$$\omega(1:N,N) = \frac{-2\Psi(1:N,N-1)}{\partial y^2} - \frac{2 * U_{wall}}{\partial y}$$

Left wall

$$\omega(1,1:N) = \frac{-2\Psi(2,1:N)}{\partial x^2}$$

Bottom wall

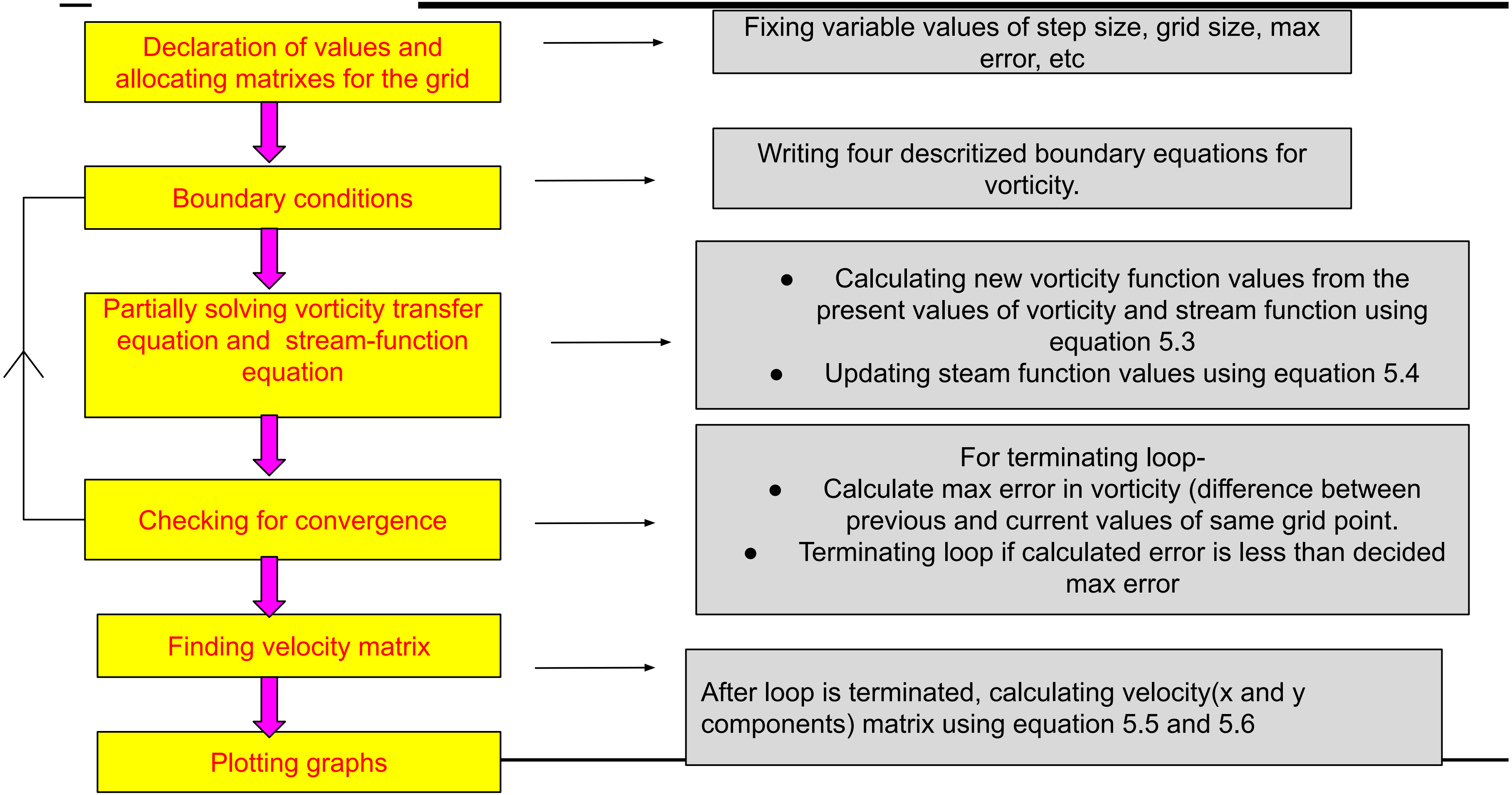
$$\omega(1:N,1) = \frac{-2\Psi(1:N,2)}{\partial y^2}$$

Right wall

$$\omega(N,1:N) = \frac{-2\Psi(N-1,1:N)}{\partial x^2}$$

Description of algorithm

The algorithm is implemented in Matlab software



Declaration of values and allocating matrixes for the grid
Fixing variable values of step size, grid size, max error, etc
Boundary conditions
Writing four descritized boundary equations for vorticity.
Partially solving vorticity transfer equation and stream-function equation
• Calculating new vorticity function values from the present values of vorticity and stream function using equation 5.3
• Updating steam function values using equation 5.4
Checking for convergence
For terminating loop-
• Calculate max error in vorticity (difference between previous and current values of same grid point.
• Terminating loop if calculated error is less than decided max error
Finding velocity matrix
After loop is terminated, calculating velocity(x and y components) matrix using equation 5.5 and 5.6
Plotting graphs

```
%%%solving vorticity/stream function and velocity function
%%for lid driven cavity
clear;
close all
%declaration of terms
N = 30;
l = 1; %length of grid
U_wall = 1; %Velocity of moving wall
rho = 1; mu = 0.01; % Density; Dynamic Viscosity;
dt = 0.001;% Time Step
maxIt = 50000; maxe = 1e-7; % Max iter; Max error
% SETUP 2D GRID
h=l/(N-1); %step size of grid
i = 2:N-1;
j = 2:N-1;
% allocating MATRIXES
w = zeros(N,N);
psi = zeros(N,N);
wp = zeros(N,N);
u = zeros(N,N);
v = zeros(N,N);
```

SOLVING LOOP SIMILAR TO GAUSS-SIEDEL METHOD

```
for iter = 1:maxIt
%BOUNDARY CONDITIONS
w(1:N,N) = -2*psi(1:N,N-1)/(h^2) - U_wall*2/h; % Top
w(1:N,1) = -2*psi(1:N,2) /(h^2); % Bottom
w(1,1:N) = -2*psi(2,1:N) /(h^2); % Left
w(N,1:N) = -2*psi(N-1,1:N)/(h^2); % Right
```

SINGLE STEP OF SOLVING VORTICITY TRANSPORT EQUATION

```
wp = w;
w(i,j) = wp(i,j) + ...
    (-1*(psi(i,j+1)-psi(i,j-1))/(2*h)...
    .* (wp(i+1,i)-wp(i-1,j))/(2*h)+(psi(i+1,j)-psi(i-1,j))/(2*h) .* (wp(i,j+1)-wp(i,j-1))/(2*h)+...
mu/rho*(wp(i+1,j)+wp(i-1,j)-4*wp(i,j)+wp(i,j+1)+wp(i,j-1))/(h^2))*dt ;
```

SINGLE STEP OF SOLVING VORTICITY - STREAM FUNCTION EQUATION

```
psi(i,j) = (w(i,j)*h^2 + psi(i+1,j) + psi(i,j+1) + psi(i,j-1) + psi(i-1,j))/4;
% CHECKING FOR CONVERGENCE

if iter > 40
    error = max(max(w - wp)) ;
    if error < maxe
        break;
    end
end
end
```

CREATE VELOCITY FROM STREAM FUNCTION

```
u(2:N-1,N) = U_wall;
u(i,j) = (psi(i,j+1)-psi(i,j-1))/(2*h); % X-COMPONENT OF VELOCITY
v(i,j) = (-psi(i+1,j)+psi(i-1,j))/(2*h); %Y-COMPONENT OF VELOCITY
```

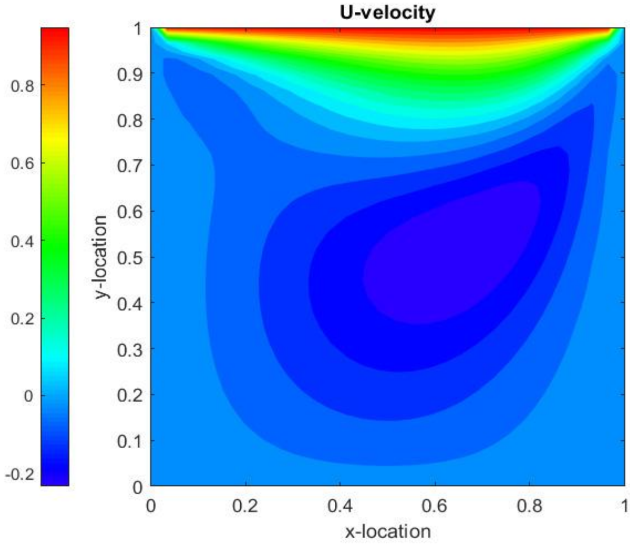
Results and Discussion

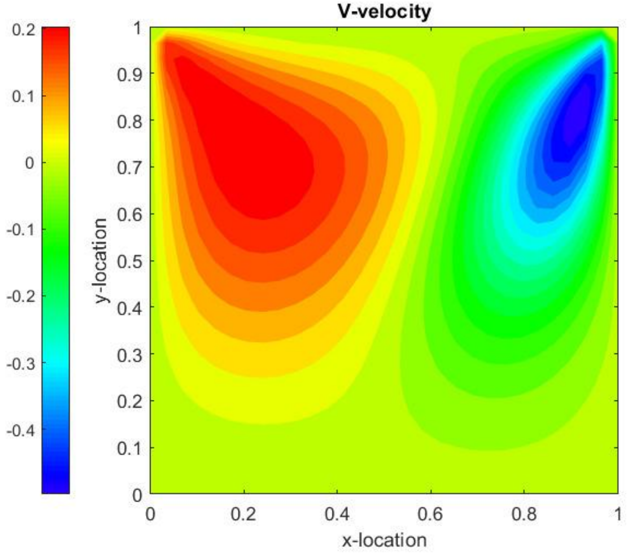
For a grid size of 30X30, we got the following results.

Figures here show the x-component (u) and y-component (v)of velocity.

Horizontal velocity is decreasing in downward direction. It means fluid nearer to the moving wall is moving faster along the same direction.

If we have moved left or right wall instead of upper, u and v will exchange the graph with 90 degree rotation accordingly.



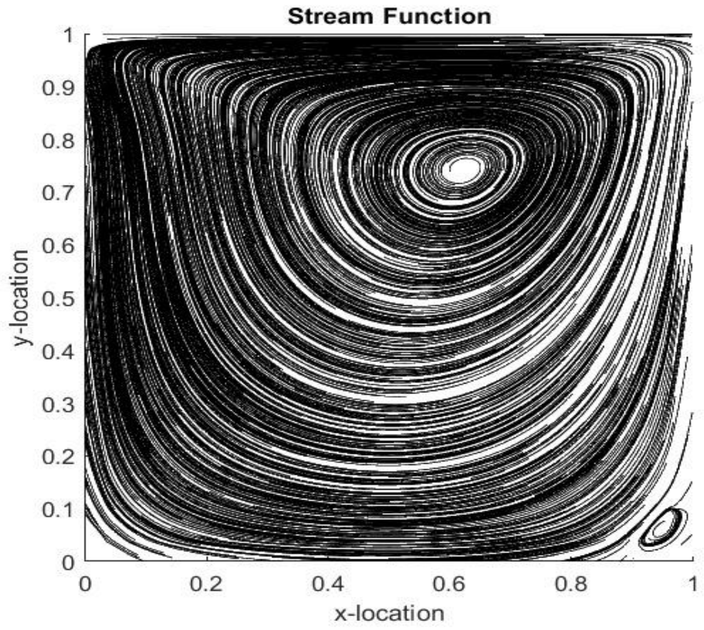


The graph in the right side represent stream function in the grid.

We can compare the magnitude and direction of fluid flow from this graph. The density of black lines denotes the magnitude of fluid flow.

We observed that stream function and vorticity are converging after some iterations. It suggests that this fluid is slowly moving toward a steady state flow.

Our numerical approach is $O(\Delta t^2, \Delta h^2)$ consistent. Also it is convergent as the solution approaches the true solution on more iterations. The error is attenuating as computation progress suggesting it is stable.



References

1. Prof. Hong G. Im, University of Michigan, Fall 2001 Course Content -Computational Fluid Dynamics-I
2. Resolution of the lid-driven cavity flow problem with an influence matrix method, Jonkman, A. S. (Author). 25 Jun 2018
3. A compact and fast Matlab code solving the incompressible
4. Navier-Stokes equations on rectangular domains, Benjamin Seibold, MIT
5. Fluid Mechanics Robert W. Fox, Alan T.McDonald
6. CFD Online https://www.cfd-online.com/Wiki/Lid-driven_cavity_problem

Thank you