Flow around a Bluff-Body in a Channel Flow

Gökhan Sarı 69645

Koç University

CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

u	velocity in x direction
u	velocity in y direction
T	temperature
F	force
p	pressure
$f_{i,j}^{n}$	a property of the fluid at x = i and y = j at time t = n
Nu	Nusselt number
C_D	Drag coefficient
h	element size
ρ	density
μ	viscosity
V	volume
A	area
L	characteristic length

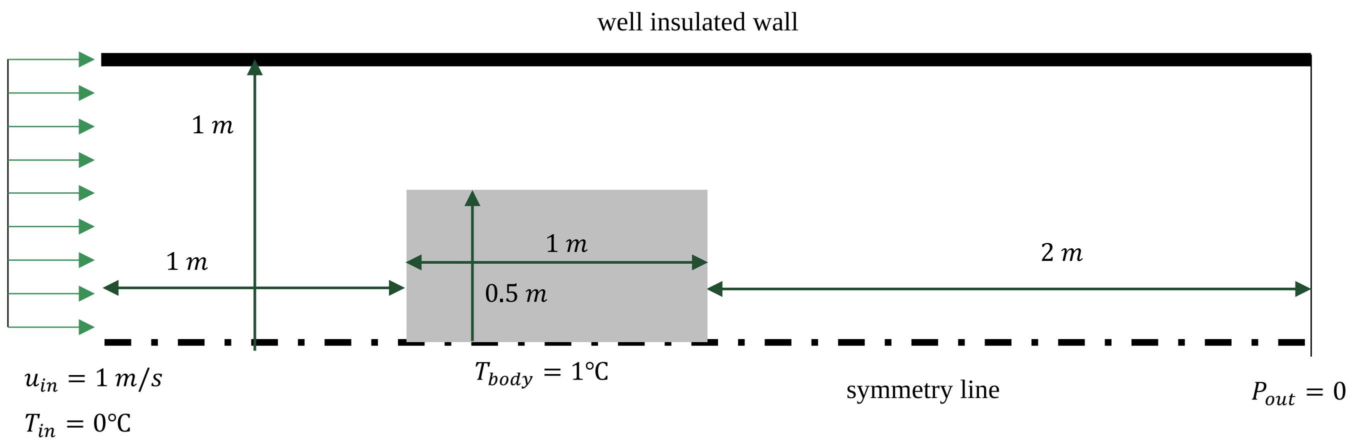


Figure 1: Sketch of the channel and bluff-body

1. INTRODUCTION

The aim of this project is to simulate the flow in a planar channel around a square shaped bluff-body with low Reynolds Number using numerical methods. Velocity vectors, pressure contours, temperature distribution are studied. Drag force acting on the object as well as Nusselt number are calculated. Navier-Stokes equations and energy equation are discretized using finite-volume method. Staggered grid is used. The results are calculated and plotted on MATLAB.

1.1 Project Description

The channel has a length of 4 meters and a height of 2 meters. The square bluff-body is 1 meter long and is between x=1 m and x=2 m. It has a temperature of 1°C. Inlet temperature is 0°C and inlet velocity is 1 m/s. Top wall is well insulated and no-slip boundary condition is applied. Normal gradient of temperature at outlet, symmetry line and top wall is zero. Pressure at the outlet is zero. Fluid density ρ is equal to 1 kg/m^3, viscosity is 0.1 kg/m.s. Reynolds number is 20. The simulation runs until the solution converges to a steady state.

2. EQUATIONS AND DISCRETIZATION

Navier-Stokes equations and energy equations are both discretized using finite volume method. Navier-Stokes equation is shown in Eq. (1).

$$\frac{\partial}{\partial t}\int_V udV = -\oint_S u\boldsymbol{u}\cdot\boldsymbol{n}dS + \nu\oint_S \nabla u\cdot\boldsymbol{n}dS - \frac{1}{\rho}\oint_S pn_x dS \qquad (1)$$

The discretized form of Eq. (1) is given by the following equations.

$$\frac{\partial}{\partial t}\int_V udV = \frac{u^{n+1}_{i+1/2,j} - u^{n}_{i+1/2,j}}{\Delta t} \qquad (2)$$

$$-\oint_S u\boldsymbol{u}\cdot\boldsymbol{n}dS = \frac{-1}{h}\left((u^2)^n_{i+1,j} - (u^2)^n_{i,j} + (uv)^n_{i+1,2,j+\frac{1}{2}} - (uv)_{i+\frac{1}{2},j-\frac{1}{2}}\right) \qquad (3)$$

$$\nu\oint_S \nabla u\cdot\boldsymbol{n}dS = \frac{\nu}{h^2}\left(u^n_{i+\frac{3}{2},j} + u^n_{i-\frac{1}{2},j} + u^n_{i+\frac{1}{2},j+1} + u^n_{i+\frac{1}{2},j-1} - 4u^n_{i+\frac{1}{2},j}\right) \qquad (4)$$

$$-\frac{1}{\rho}\oint_S pn_x dS = -\frac{1}{h}\left(P_{i+1,j} - P_{i,j}\right) \qquad (5)$$

The energy equation and the discretized version are given by the equations Eq. (6) and Eq. (7), respectively.

$$\frac{\partial T}{\partial t} + \boldsymbol{u}\cdot\nabla T = \alpha\nabla^2 T \qquad (6)$$

$$T^{n+1} = T^n + \Delta t\cdot\left[\frac{1}{2h}\left(-u_E(T+T_E) + u_w(T+T_w) - v_N(T+T_N) + v_S(T+T_S)\right) + \cdots\frac{\alpha}{h^2}(T_E + T_N + T_S + T_W - 4T)\right] \qquad (7)$$

2.1 Computational Domain

Three different grids with element sizes of h = 0.0625 m, h = 0.03125 m and h = 0.15625 m are used. A simplified version of the grid is shown in Fig. (2), which has 10 x 34 = 340 cells. The numbers on the figure correspond to inlet, top wall (no-slip boundary condition), outlet, symmetry line (full-slip boundary condition) and bluff-body (T = 1°C), respectively.

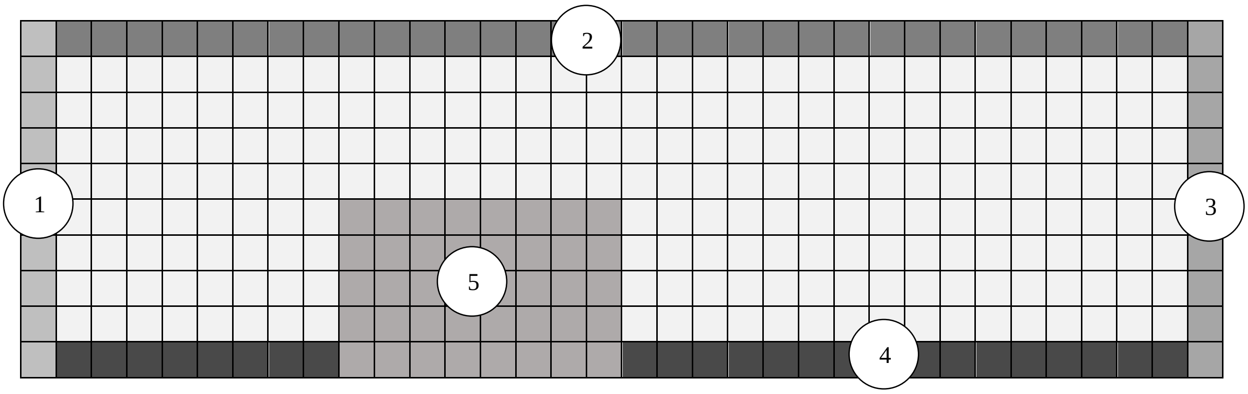


Figure 2: A simpified version of the grid

2.2 Implementation of Boundary Conditions

No-slip boundary condition is applied on both the bluff body and the top wall. Velocities in the ghost cells are defined such that the velocity on the wall is zero. A part of the grid is shown in Fig. (3).

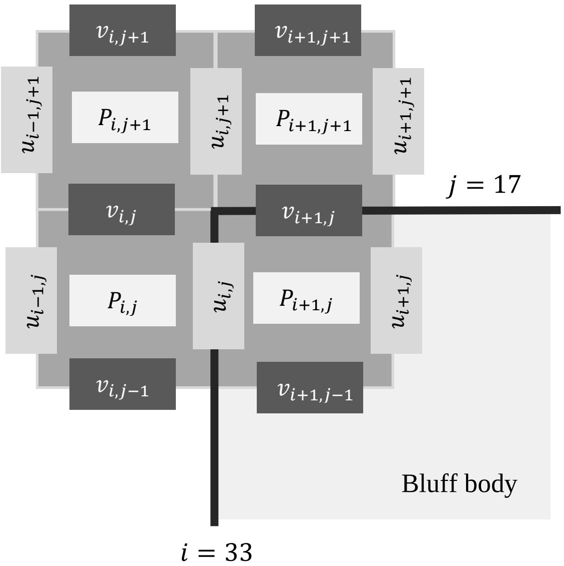


Figure 3: A part of grid at the corner of the bluff body

The following equations are used on the boundaries of the bluff body.
Left side:

$$u(i = 33, j = 1{:}17) = 0 \tag{8}$$
$$v(i = 34, j = 1{:}16) = -v(i = 33, j = 1{:}16) \tag{9}$$

Top:

$$u(i = 34{:}64, j = 17) = -u(i = 34{:}64, j = 18) \tag{10}$$
$$v(i = 34{:}65, j = 17) \tag{11}$$

Full slip boundary condition is applied on the symmetry line. There is no crossflow along the symmetry line, therefore, v velocity along the line is set to zero. u velocity, on the other hand, is set to be equal to the u velocity of the cell above.

Pressure at the outlet is zero, therefore, pressure value in the ghost cells at the outlet is set to be the negative of the pressure value of non-ghost neighbor cell, thus their sum is equal to zero. u and v velocities at the outlet are set to have the same velocities of the cell on their left.

Inlet velocity is constant and uniform. u velocities at the inlet are 1 m/s and v velocities are 0 m/s.

Normal gradient of temperature at outlet, top wall and symmetry line is zero. Therefore, temperature values at these locations are set to be equal to the temperature value of the neighbor cell.

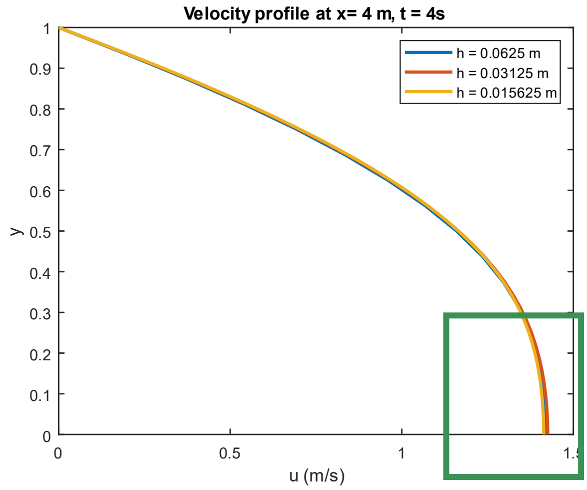
VERIFICATION

Mass flow rate at the inlet and outlet is compared for three different mesh sizes. The differences between outlet mass flow rate and inlet mass flow rate are shown in Table 1. The difference between mass flow rates decreases as finer grids are used as expected.

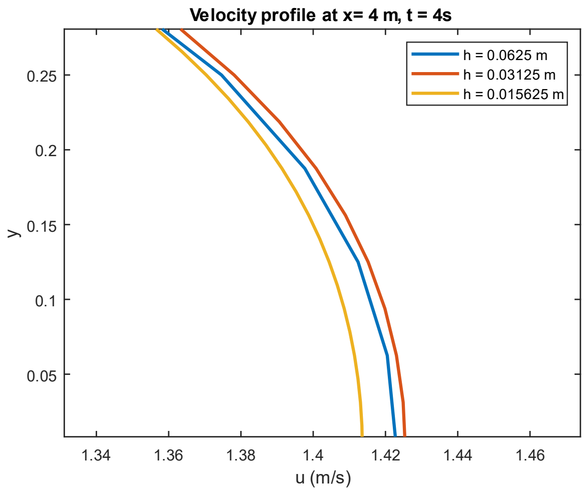
Table 1: Element size vs difference between mass flow

Element Size	**Differrence between $\dot{m}_{in}$ and $\dot{m}_{out}$ (Percentage)**
h = 0.0625 m	4.45 %
h = 0.03125 m	2.72 %
h = 0.015625 m	1.39 %

Velocity profiles at x = 4 are also compared. In Fig. (3a) u velocity profile is shown, and in Fig. (3b), a closer look to of the velocity profile in the green region is shown where difference between grids is largest.



(a)



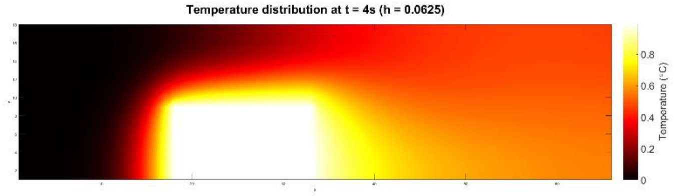
(b)

Figure 4: Velocity profiles at x = 4 for different grids

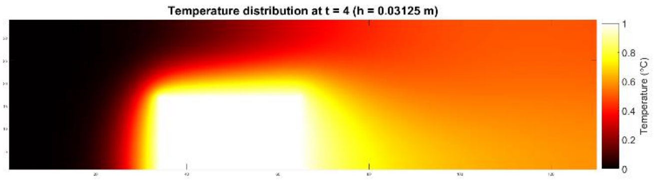
The results are sufficiently close to conclude that grid convergence is achieved.

RESULTS AND DISCUSSION

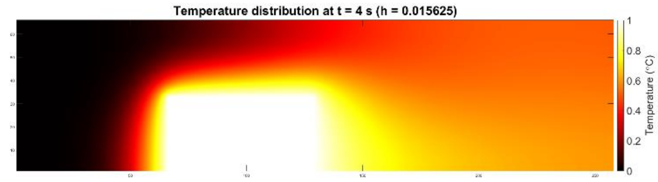
Temperature and pressure contours for three different mesh sizes are shown in Fig. (5) and Fig.(6), respectively. There is no significant difference between different meshes for temperature contours and overall pressure distribution is similar among meshes, but maximum pressure increases as finer mesh is used.



(a)

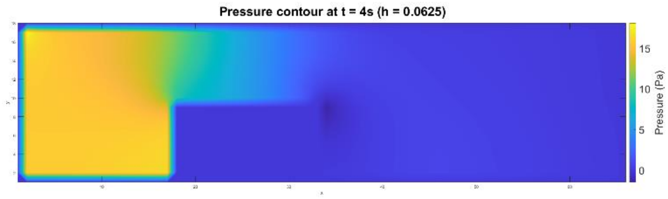


(b)

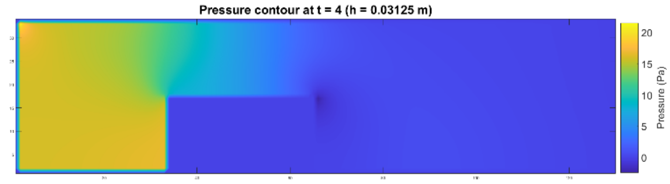


(c)

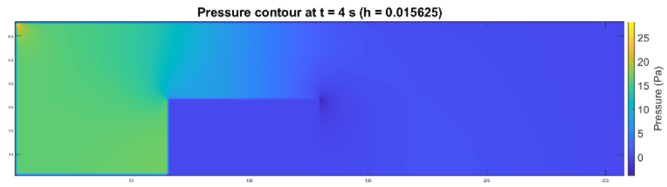
Figure 5: Temperature contours **(a)** h = 0.0625 m **(b)** h = 0.03125 m **(c)** h = 0.015625 m



(a)



(b)



(c)

Figure 6: Pressure contours **(a)** h = 0.0625 m **(b)** h = 0.03125 m **(c)** h = 0.015625 m

RESULTS AND DISCUSSION

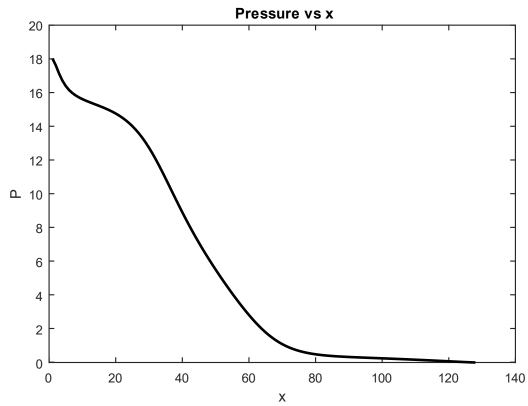
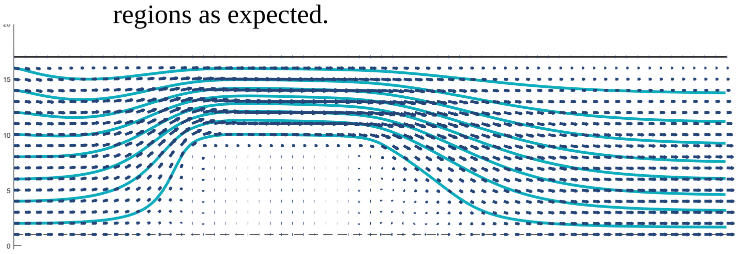
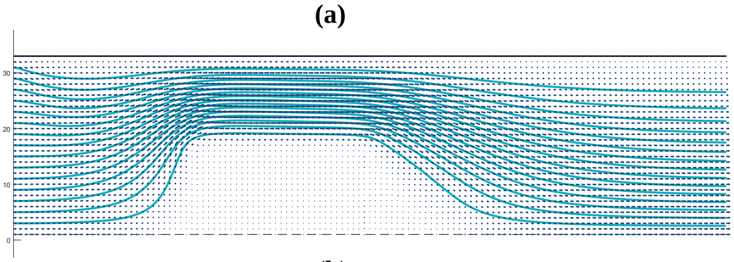


Figure 7: Pressure versus x (for h = 0.03125 m)

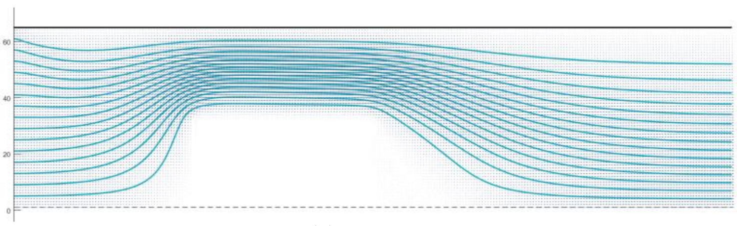
In Fig. (7), plot of the pressure along x close to top wall is shown. Pressure decreases faster in narrow regions as expected.



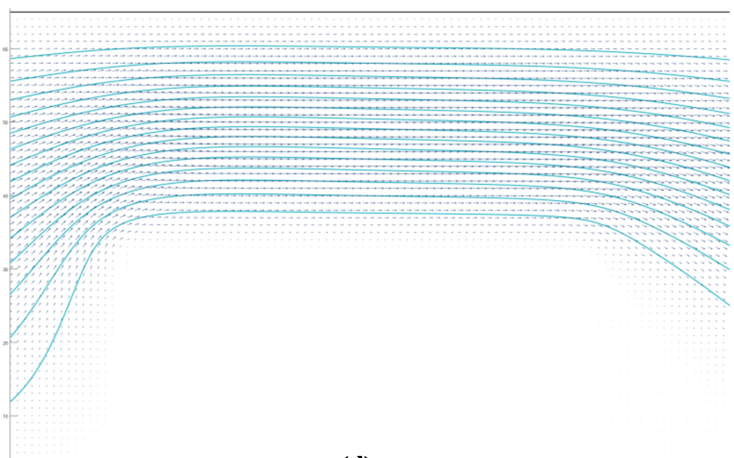
(a)



(b)



(c)



(d)

Figure 4: Velocity vectors **(a)** h = 0.0625 m **(b)** h = 0.03125 m **(c,d)** h = 0.015625 m (d is closer look to c)

In Fig. (8), velocity vectors are shown. Velocity increases in narrow regions and decreases in wider regions, thus, mass conservation is maintained along the channel.

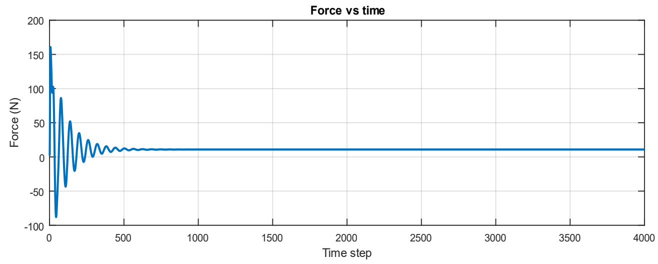


Figure 9: Force acting on the upper half of block body vs time (for h = 0.03125 m)

In Fig. (9), force vs time graph is shown, however an unexpected behavior is observed. Force oscillates over time and converges to 11.27 N. This could be due to error in pressure since pressure is solved iteratively with a maximum number of iteration 100.

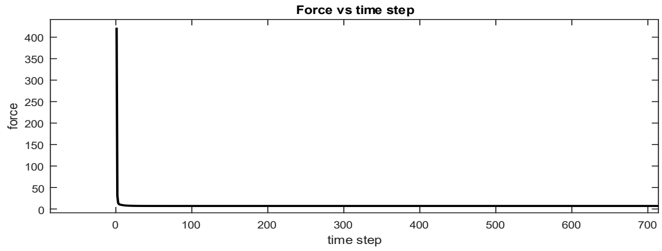


Figure 10: Force acting on the upper half of block body vs time (for h = 0.0625 m)

In Fig. (10), force vs time is shown for the coarse grid. Pressure is solved iteratively based on error instead of maximum iteration number, and no oscillation is observed. Force converges to 7 N.

Drag coefficient C_D is calculated to be 43.5. Drag force is taken as 11.27 x 2 N instead of 7 x 2 N, since finer grid is used. Nusselt number Nu is calculated as 16.4. Details of calculations of both Nusselt number and drag coefficient are available in Appendix A.

In Fig. (11), temperature flux from the object over time is shown. Pressure flux is highest at the beginning and decreases sharply in a short period of time as expected since temperature difference at the beginning is the highest.

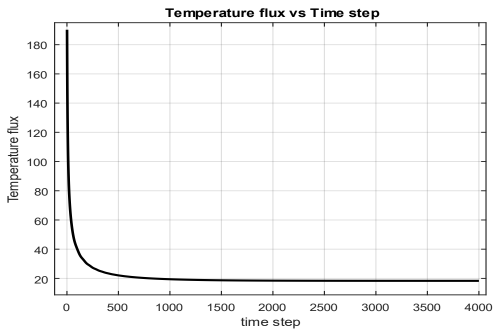
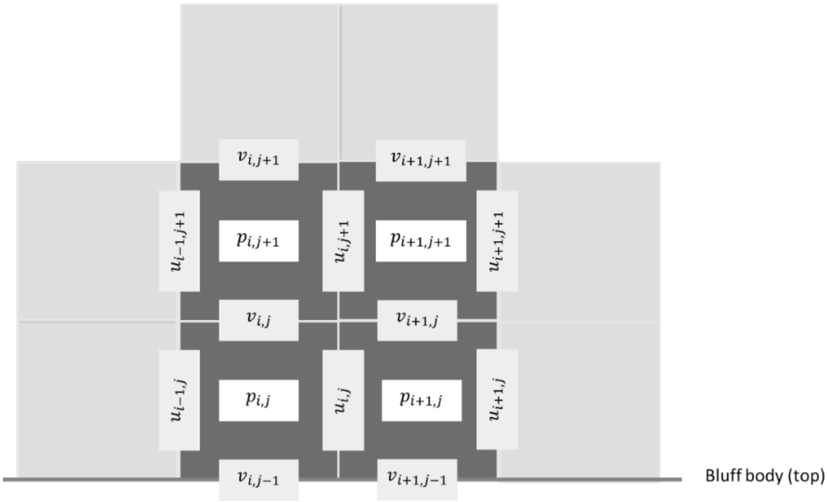


Figure 11: Temperature flux vs time

APPENDIX A: CALCULATIONS

Calculation of Drag Force



$\tau = \mu \frac{du}{dy} = \mu \frac{u_{i,j+1} - u_{i,j}}{h}$

$dF_{drag} = \tau dA = \mu \frac{u_{i,j+1} - u_{i,j}}{h} h = \mu \left(u_{i,j+1} - u_{i,j} \right)$

dF's are calculated on the top surface of the bluff body and then summed up.

Calculation of Nussel Number

$$Nu = \frac{L}{\Delta T} \frac{1}{A} \iint_A \left(\frac{\partial T}{\partial n} |_{wall} \right) dA$$

$$Nu = 2 \cdot \sum_{all\ surfaces} (T_{n+1} - T_n) \cdot \frac{1}{A}$$

Characteristic length L is taken as the height of the channel, which is 2.

APPENDIX B: MATLAB CODE

```
clear; clc; close all;
nx=128;ny=32;
dt=0.001;nstep=4000;
mu=0.1;
maxit=5000;
beta=1.8;
h=1/ny;
u=zeros(nx+1,ny+2);v=zeros(nx+2,ny+1);p=zeros(nx+2,ny+2);
ut=zeros(nx+1,ny+2);vt=zeros(nx+2,ny+1);c=zeros(nx+2,ny+2)+0.25;
uu=zeros(nx+1,ny+1);vv=zeros(nx+1,ny+1);w=zeros(nx+1,ny+1);
force = zeros(1,nstep);
T_flux = zeros(1,nstep);
nu = zeros(1,nstep);
% Initial boundary conditions
u(1,:) = 1; % inlet velocity
T = zeros(nx+2,ny+2);
T(34:65, 1:17) = 1; % Temperature of the body

c(2,3:ny)=1/3;
c(nx+1,3:ny)=1/4;
c(3:nx,2)=1/3;
c(3:nx,ny+1)=1/3;
c(2,2)=1/2;c(2,ny+1)=1/2;
c(nx+1,2)=1/3;c(nx+1,ny+1)=1/3;

% c values for the body
c(33,2:17) = 1/3;
c(66,1:17) = 1/3;
c(34:65,18) = 1/3;
c(33,2) = 1/2;
c(66,2) = 1/2;
c(34:65,2:17) = 0;


time = 0;
for is = 1:nstep
    % Symmetry line (Full-slip boundary condition)
    u(1:nx+1,1)=u(1:nx+1,2);
    v(:,1) = 0;
    T(:,1) = T(:,2);

    % Top wall
    u(1:nx+1,ny+2) = -u(1:nx+1,ny+1);
    v(:,end) = 0;
    T(:,end) = T(:,end-1); % well insulated wall

    % Outlet
    u(end,:)=u(end-1,:);
    v(end,:)=v(end-1,:);
    T(end,:) = T(end-1,:);

    % Body (Top)
    u(34:64,17) = -u(34:64,18);
    v(34:65,17) = 0;

    % Body (Left)
    u(33,1:17) = 0;
    v(34,1:16) = -v(33,1:16);

    % Body (Right)
    u(65,1:17) = 0;
    v(64,1:16) = -v(65,1:16);

    ut = u;
    vt = v;
```

```
    for i=2:nx
        for j=2:ny+1 % temporary u-velocity
            ut(i,j)=u(i,j)+dt*(-(0.25/h)*((u(i+1,j)+u(i,j))^2-(u(i,j)+...
                u(i-1,j))^2+(u(i,j+1)+u(i,j))*(v(i+1,j)+...
                v(i,j))-(u(i,j)+u(i,j-1))*(v(i+1,j-1)+v(i,j-1)))+...
                (mu/h^2)*(u(i+1,j)+u(i-1,j)+u(i,j+1)+u(i,j-1)-4*u(i,j)));
        end
    end
    for i=2:nx+1
        for j=2:ny % temporary v-velocity
            vt(i,j)=v(i,j)+dt*(-(0.25/h)*((u(i,j+1)+u(i,j))*(v(i+1,j)+...
                v(i,j))-(u(i-1,j+1)+u(i-1,j))*(v(i,j)+v(i-1,j))+...
                (v(i,j+1)+v(i,j))^2-(v(i,j)+v(i,j-1))^2)+...
                (mu/h^2)*(v(i+1,j)+v(i-1,j)+v(i,j+1)+v(i,j-1)-4*v(i,j)));
        end
    end
    for it=1:maxit % solve for pressure
        p_old = p;
        for i=2:nx+1
            for j=2:ny+1
                p(i,j)=beta*c(i,j)*(p(i+1,j)+p(i-1,j)+p(i,j+1)+p(i,j-1)-...
                    (h/dt)*(ut(i,j)-ut(i-1,j)+vt(i,j)-vt(i,j-1)))+(1-beta)*p(i,j);
                p(34:65,1:17) = 0;
                p(end,:) = -p(end-1,:);
            end
        end
        if norm(p-p_old) < 0.0001
            break
        end
    end

    % correct the velocity
    u(2:nx,2:ny+1)=...
        ut(2:nx,2:ny+1)-(dt/h)*(p(3:nx+1,2:ny+1)-p(2:nx,2:ny+1));
    v(2:nx+1,2:ny)=...
        vt(2:nx+1,2:ny)-(dt/h)*(p(2:nx+1,3:ny+1)-p(2:nx+1,2:ny));

    time=time+dt;

    % Body (Top)
    u(34:64,17) = -u(34:64,18);
    v(34:65,17) = 0;

    % Body (Left)
    u(33,1:17) = 0;
    v(34,1:16) = -v(33,1:16);

    % Body (Right)
    u(65,1:17) = 0;
    v(64,1:16) = -v(65,1:16);

    % Body (in)
    u(34:64,1:17) = 0;
    v(35:64,1:16) = 0; % CHANGED 34 -> 35

    % Temperature
    T2 = T;
    for i=2:nx+1
        for j=2:ny+1
            T2(i,j) = T(i,j) + dt*( -(1/h)*(-(T(i-1,j)+T(i,j))*u(i-1,j)/2 + ...
                (T(i+1,j)+T(i,j))*u(i,j)/2 + (T(i,j+1)+T(i,j))*v(i,j)/2 + ...
                -(T(i,j-1)+T(i,j))*v(i,j-1)/2) + ...
                (0.1/h^2)*(T(i+1,j)+T(i-1,j)+T(i,j-1)+T(i,j+1)-4*T(i,j)));
        end
    end
    T = T2;
    T(34:65,1:17) = 1;
```

```
    % plot results
    uu(1:nx+1,1:ny+1)=0.5*(u(1:nx+1,2:ny+2)+u(1:nx+1,1:ny+1));
    vv(1:nx+1,1:ny+1)=0.5*(v(2:nx+2,1:ny+1)+v(1:nx+1,1:ny+1));
    w(1:nx+1,1:ny+1)=(u(1:nx+1,2:ny+2)-u(1:nx+1,1:ny+1)-...
        v(2:nx+2,1:ny+1)+v(1:nx+1,1:ny+1))/(2*h);

    hold off;
    contourf(T', 'LineColor','none');
    colormap jet
    hold on;
    quiver(uu',vv')
    %contour(w',20);
    axis equal;
    title(num2str(is*dt))
    pause(0.001)

    % Force
    f_drag = sum(mu*(uu(33:65,18)-u(33:65,17)));
    f_pres = sum(p(33,2:17)*h - p(66,2:17)*h);
    force(is) = f_drag + f_pres;

    % Temperature flux
    T_flux(is) = (0.1/h^2)*(sum((T(66,1:17)-T(65,1:17))) + ...
        sum((T(33,1:17)-T(34,1:17))) +...
        sum((T(34:65,18)-T(34:65,17))));

    % Nusselt number
    nu(is) = 2*(sum(T(34,1:17)-T(33,1:17))/0.5+...
    sum(T(34:65,17)-T(34:65,18))/1+...
    sum(T(65,1:17)-T(66,1:17))/0.5);


end


figure

for i = 1:2:33
    strm = streamline(uu',vv',1,i);
    set(strm, 'Color', [14 173 190]/255)
    set(strm, 'LineWidth', 4)
    hold on
end
plot([1,nx+1],[ny+1,ny+1],'k-', 'LineWidth',2)
plot([1,nx+1],[1,1],'k--')
axis equal
```