Numerical Analysis Final Report

Constructing Model for Model Velocity by Navier-Stokes Equations

NTUAS Freshman Aaron Hsieh

Contents

01 Problem Description

- How to use some known parameters to construct a model?
- How to use numerical methods to find out the flow of blood, and the steady state

02 Derivation

Momentum conservation

$$\overbrace{\rho\left(\underbrace{\frac{\partial \mathbf{v}}{\partial t}}_{\text{Unsteady acceleration}} + \underbrace{(\mathbf{v}\cdot\nabla)\mathbf{v}}_{\text{Convective acceleration}}\right)}^{\text{Inertia}} = \underbrace{-\nabla p}_{\text{Pressure gradient}} + \underbrace{\mu\nabla^2\mathbf{v}}_{\text{Viscosity}} + \underbrace{\mathbf{f}}_{\text{Other forces}}$$

Mass conservation (Incompressible)

$$\nabla\cdot\mathbf{v} = 0$$

Assume density is constant

$$\rho\left(\frac{\partial v_x}{\partial t} + v_x\frac{\partial v_x}{\partial x} + v_y\frac{\partial v_x}{\partial y} + v_z\frac{\partial v_x}{\partial z}\right) = \mu\left[\frac{\partial^2 v_x}{\partial x^2} + \frac{\partial^2 v_x}{\partial y^2} + \frac{\partial^2 v_x}{\partial z^2}\right] - \frac{\partial p}{\partial x} + \rho g_x$$

$$\rho\left(\frac{\partial v_y}{\partial t} + v_x\frac{\partial v_y}{\partial x} + v_y\frac{\partial v_y}{\partial y} + v_z\frac{\partial v_y}{\partial z}\right) = \mu\left[\frac{\partial^2 v_y}{\partial x^2} + \frac{\partial^2 v_y}{\partial y^2} + \frac{\partial^2 v_y}{\partial z^2}\right] - \frac{\partial p}{\partial y} + \rho g_y$$

$$\rho\left(\frac{\partial v_z}{\partial t} + v_x\frac{\partial v_z}{\partial x} + v_y\frac{\partial v_z}{\partial y} + v_z\frac{\partial v_z}{\partial z}\right) = \mu\left[\frac{\partial^2 v_z}{\partial x^2} + \frac{\partial^2 v_z}{\partial y^2} + \frac{\partial^2 v_z}{\partial z^2}\right] - \frac{\partial p}{\partial z} + \rho g_z$$

02 Derivation

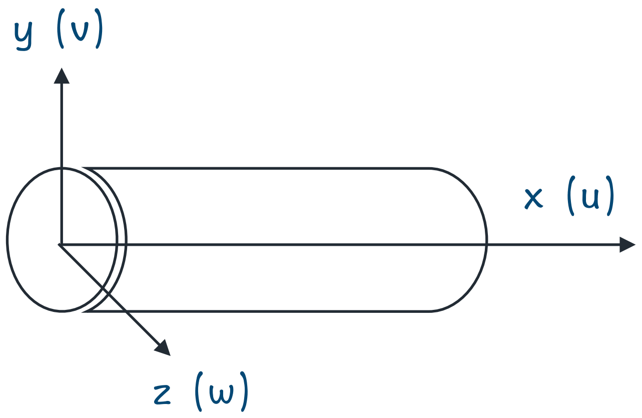
Assume $v = w = 0$ $\quad P = P(x)$ $\quad u = u(y, z)$ $\quad \nabla \cdot \vec{v} = 0$

Analyze x-direction

$$\rho\left(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}\right) = \mu\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right] - \frac{\partial p}{\partial x} + \rho g_x$$

$$\longrightarrow \quad \rho\frac{\partial u}{\partial t} = -\frac{\partial P}{\partial x} + \mu\left(\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)$$

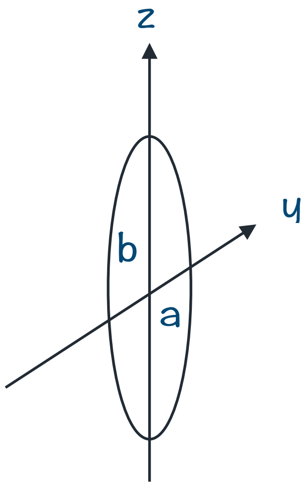
$$\longrightarrow \quad \partial u = \left(\frac{-1}{\rho}\frac{\partial P}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)\right)\partial t$$



02 Derivation

Consider an elliptic cross section

$$u(y,z) = \frac{1}{2\mu}\frac{dP}{dx}\frac{a^2b^2}{a^2+b^2}(\frac{y^2}{a^2}+\frac{z^2}{b^2}-1)$$



For circular cross section

$$u(y,z) = \frac{1}{4\mu}\frac{dP}{dx}(r^2-R^2)$$

02 Derivation — Code

```python
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.cm as cm


def FD2(ur, delta_x, ax):
    du_dx = (np.roll(ur, -1, axis=ax) - ur) / delta_x
    return du_dx


def SD3(ur, delta_x, ax):
    du_dx_2 = (np.roll(ur, -1, axis=ax) - 2 * ur + np.roll(ur, 1, axis=ax)) / delta_x ** 2
    return du_dx_2


def BC(u):
    r = np.zeros(np.shape(u[0, :, :]))
    ny = np.squeeze(np.shape(u[0, :, 0]))
    nz = np.squeeze(np.shape(u[0, 0, :]))

    yc = dy * (ny - 1) / 2  # Center of y (R)
    zc = dz * (nz - 1) / 2  # Center of z (R)
    for i in range(ny):
        for j in range(nz):
            y = i * dy
            z = j * dz
            r[i, j] = ((y - yc) ** 2 + (z - zc) ** 2) ** 0.5  # Calculate the distance from center
            if r[i, j] >= dy * (ny - 1) / 2:  # If distance >R, set u to zero
                u[:, i, j] = 0
    return u
```

```python
# initial value
mu = 3 * 10 ** (-3)
rho = 1.06 * 10 ** 3
nu = mu / rho

n = 101  # 切幾等分
R = 0.0125  # 血管半徑
y = np.linspace(-R, R, n)
dy = y[1] - y[0]
z = np.linspace(-R, R, n)
dz = z[1] - z[0]
r = np.zeros([101, 101])

for i in range(101):
    for j in range(101):
        r[i, j] = np.sqrt(y[i] ** 2 + z[j] ** 2)

r = np.where(r <= R, r, 0)

dt = 0.001

x = 1  # 血管長度
P1 = 15998.6842  # 初始壓力
P = np.zeros((n, n, n))
dx = x / n
for i in range(n):
    P[i, :, :] = P1 - 666.611842 * dx * i

dP_dx = FD2(P, dx, 0)

u = np.zeros([n, n, n])

d2u_dy2 = SD3(u, dy, 1)
d2u_dz2 = SD3(u, dz, 2)
```

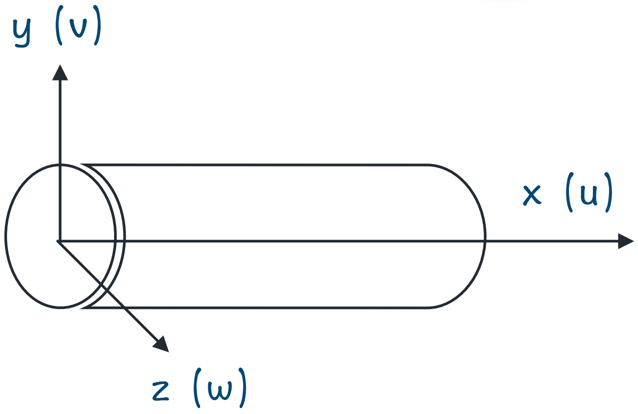
02 Derivation — Code

```
for i in range(20000):
    u = BC(u)
    u += (-1 / rho * dP_dx + nu * (d2u_dy2 + d2u_dz2)) * dt
    d2u_dy2 = SD3(u, dy, 1)
    d2u_dz2 = SD3(u, dz, 2)
    t = round(i * dt, 3)

    if i % 100 == 0:
        plt.figure(figsize=(8, 6))
        plt.title("The velocity of the blood, t = %f" % t)
        a = plt.contourf(u[50, :, :], cmap=cm.jet, levels=np.arange(0, 0.65, 0.01))
        plt.colorbar(a)
        plt.savefig("%d.png" % i)
        plt.close()
```

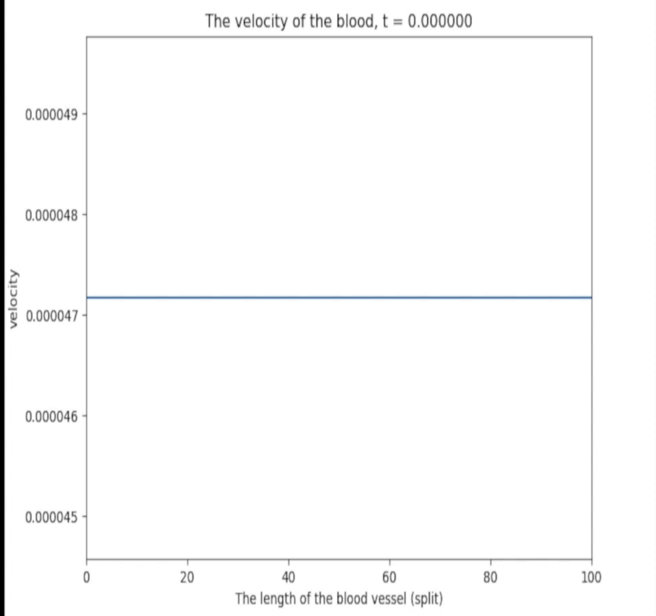
03 Results——Numerical solution （Case 1）

- Vessel radius : 0.0125 [m]
- Vessel length : 0.5 [m]
- Dynamic Viscosity : $\mu = 3 \times 10^{-3} \quad [\frac{kg}{s \times m}]$
- Kinetic Viscosity : $\nu = 2.8 \times 10^{-6} \quad [\frac{m^2}{s}]$
- Pressure : $120 \quad [mm - Hg] = 15998.6842 \quad [\frac{N}{m^2}]$
- Pressure Difference : $50 \quad [Pa]$

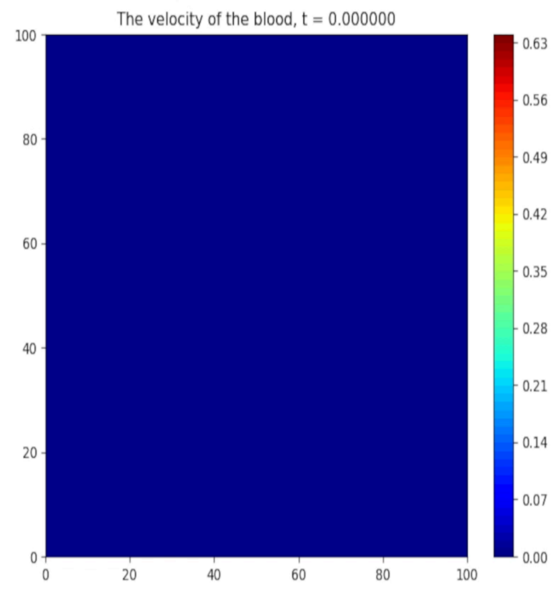


03 Results —— Numerical Solution (Case 1)

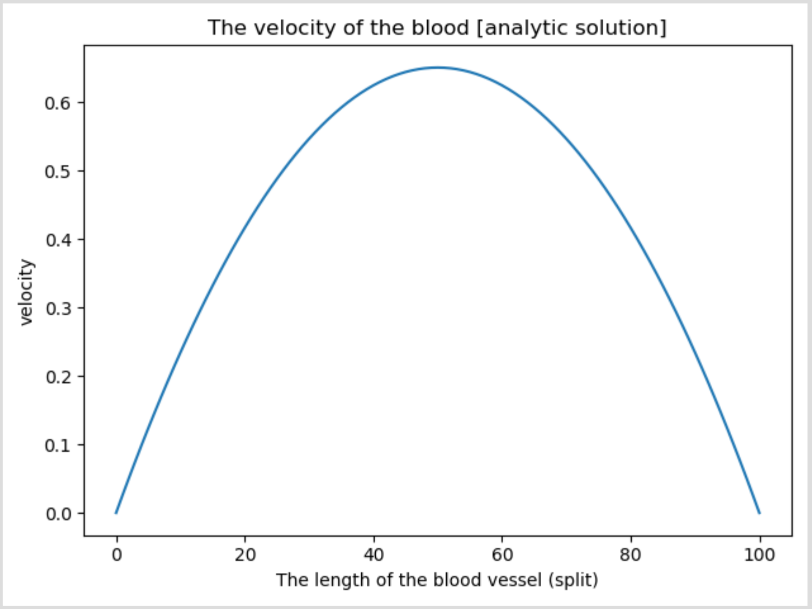
Link (Click Me)

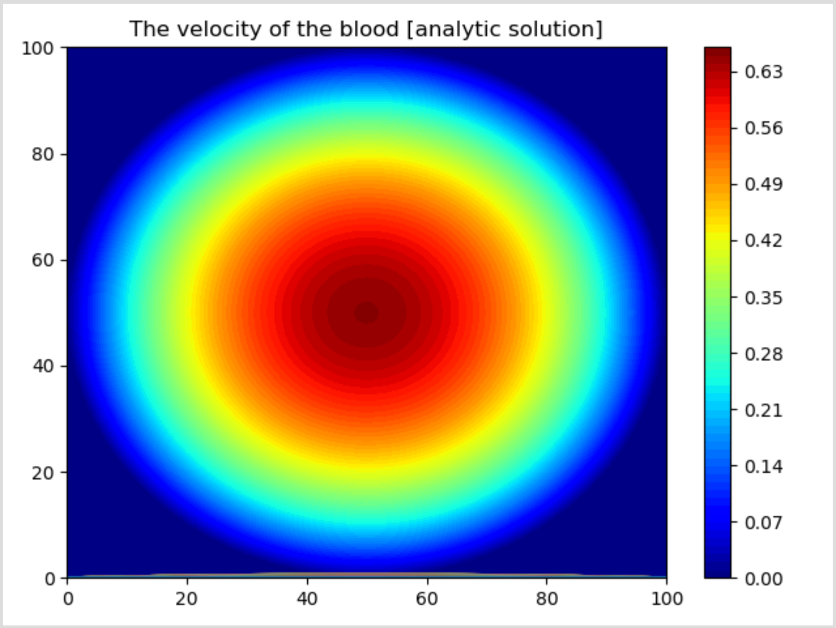


Link (Click Me)



03 Results —— Analytical Solution (Case 1)





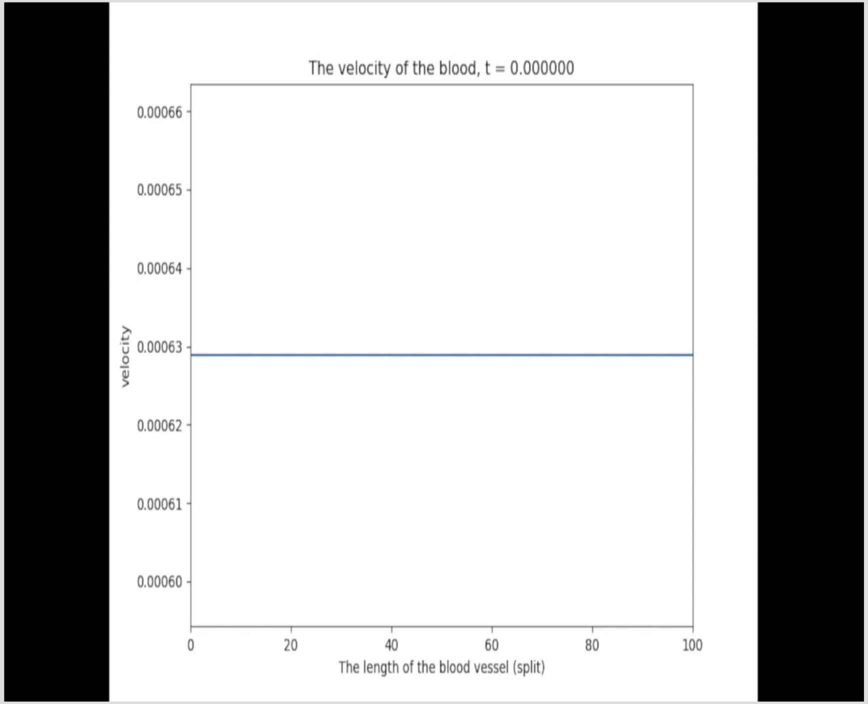
03 Results——Numerical solution（Case 2）

- Vessel radius : 0.0125 [m]
- Vessel length : 1 [m]
- Dynamic Viscosity : $\mu = 3 \times 10^{-3} \quad [\frac{kg}{s \times m}]$
- Kinetic Viscosity : $\nu = 2.8 \times 10^{-6} \quad [\frac{m^2}{s}]$
- Pressure : $120 \quad [mm - Hg] = 15998.6842 \quad [\frac{N}{m^2}]$
- Pressure Difference : $666.611842 \quad [Pa]$

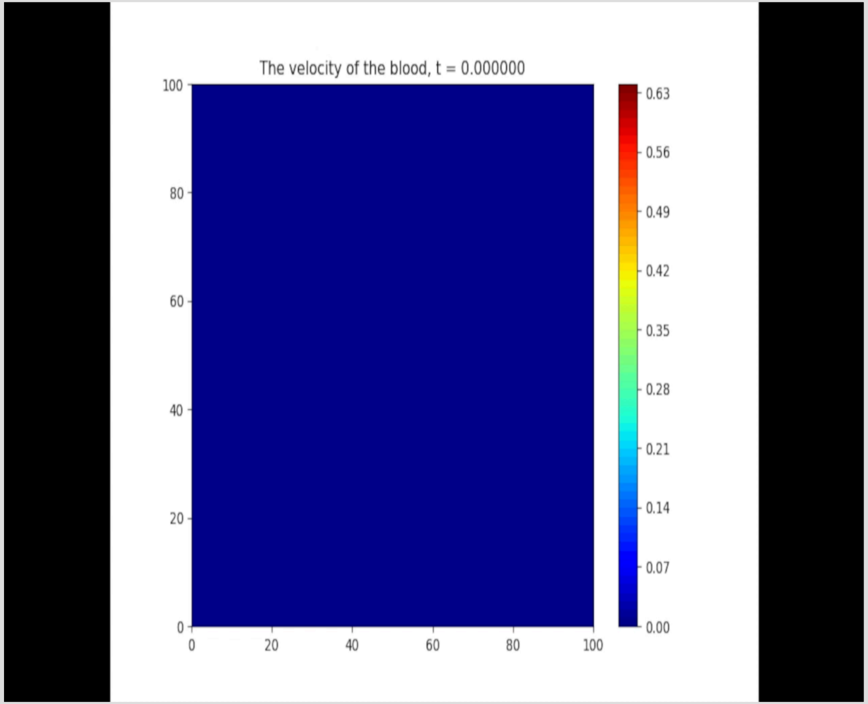
03 Results —— Numerical Solution (Case 2)

The real blood velocity in arteries is 0.5 m/s

Link (Click Me)

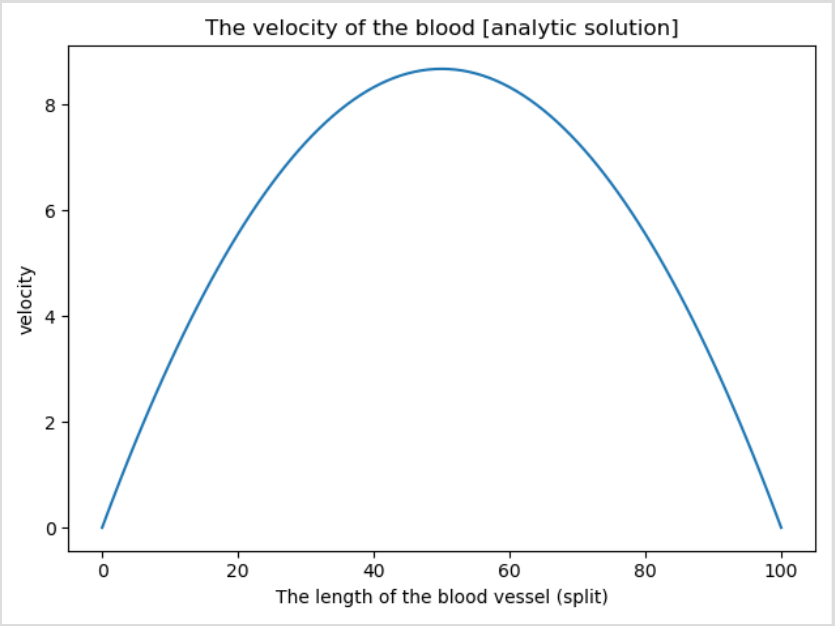


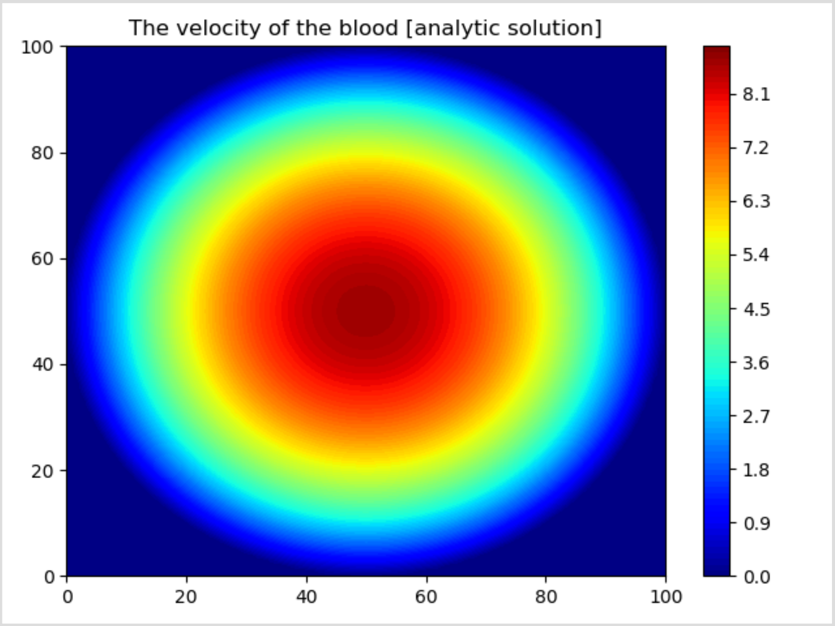
Link (Click Me)



03 Results——Analytical solution（Case 2）

Steady State





04 Conclusion and Discussion

- The blood velocity didn't reach steady state in the simulation

- The results from the model are consistent with the theoretical values. The blood velocity at the wall is slow and that at the center is high.

04 Conclusion and Discussion

How to improve

- v and w should also be considered
- The thick and thin part of blood vessels can be considered
- The time evolution of blood pressure should also be considered