

Linearized Model of Blood Flow in Vessels

Project 2 for Modeling and Simulation in Science, Engineering and Economics

Murphy Pan

Courant Institute of Mathematical Sciences - New York University

Abstract. Some fundamental concepts of modeling and simulation in the field of biology involve an understanding of the circulation model. In our case, we will take the non-linear Navier-Stokes equation for conservation of mass and linearize them. We will implement a two step Lax-Wendroff method. Our results will tell us how pressure affects area and flow of a blood vessel.

1 Introduction & Motivation

1.1 Introduction

Blood flows through vessels commonly known as arteries and veins. To begin, blood first flows through the arteries, these blood vessels carry blood away from the heart. It begins with a systolic contraction to increase pressure, the heart contracts and forces blood to pass through the arteries. This pressure then decreases during a diastolic relaxation, the heart relaxes after the contraction as the heart fills up with blood. This is where the 120/80 comes from; 120 refers to the systolic pressure in mmHg and 80 refers to diastolic pressure which is 80 mmHg. Because of this, arteries typically are high pressured vessels. Veins on the other hand are typically low pressure vessels. Because of this, some veins have valves that promote unidirectional flow. To increase pressure and cause blood to flow back to the heart, most veins are surrounded by skeletal muscles that help drive blood back to the heart. For example, as leg muscles contract, which usually occur during walking or running, they exert pressure on nearby veins with their numerous one-way valves. This increased pressure causes blood to flow upward and back to the heart.

1.2 Motivation

One of the topics covered in Modeling and Simulations was the heart. We wanted to extend this by examining precisely how blood flow is related to pressure and area. Additionally, we are motivated to see how pressure affects area and flow. Moreover, we want to see how the stiffness of the vessel affects flow and area. Finally, we want to see how density affects flow and area.

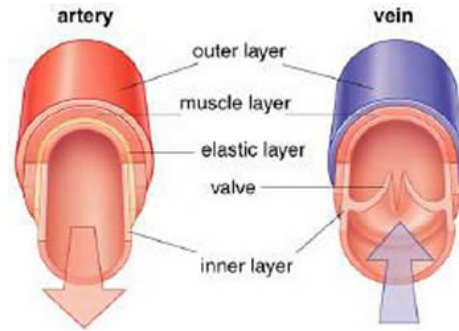


Fig. 1. An example of blood vessels that we will model

2 Governing Equations

2.1 Navier-Stokes Equation

Before the actual construction and modeling of a blood vessel, it is necessary to build the equations to describe blood flow. To do this, we have to look at the Navier-Stokes equations. These equations describe the flow of viscous fluids. Viscosity refers to the informal concept of "thickness". For example, syrup has a higher viscosity compared to water. Since we are primarily concerned about area and flow the equation will be in terms of area and flow. Area and flow both being function of space and time. Thus, the equation can then be described as follows:

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0 \quad (1)$$

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q^2}{A} \right) + \frac{A}{\partial x} \frac{\partial p}{\partial x} = 2 \frac{Q}{1 A} \quad (2)$$

$$p = (A^{\frac{1}{2}} - A_0^{\frac{1}{2}}) + p_0 \quad (3)$$

The equation (1) expresses the idea that the fluid is incompressible. The equation (2) is our non-linear Navier-Stokes equation. Equation (3) is our assumption that pressure is dependent on some rest area A_0 , the current area A , and a rest pressure p_0 .

Equation (2) is very difficult to model, thus we will make some assumptions, first, we will assume $\frac{\partial Q}{\partial t}$ to be 0 and $\frac{\partial}{\partial x} \left(\frac{Q^2}{A} \right)$ to be 1. Next, we insert p into equation (2) and group all the spatial derivatives together. We want an equation that is of the form:

$$\frac{\partial(\mathbf{u})}{\partial t} + \frac{\partial F(\mathbf{u})}{\partial x} = 0 \quad (4)$$

where \mathbf{u} consists of $[A, Q]$, such that A and Q are functions of x and t . We do this because it is easier to model an advection equation. By doing this, we see that $F(\mathbf{u})$ is

$$\begin{bmatrix} Q \\ \frac{Q^2}{A} + \frac{1}{3}A^{\frac{3}{2}} \end{bmatrix} \quad (5)$$

We will linearize this equation about an (A, Q) . Namely, we will linearize about $(A_0, 0)$. This means that at some rest area, there is no flow.

We can rewrite equation (4) to be

$$\frac{\partial \mathbf{u}}{\partial t} + r_{\mathbf{u}} F(\mathbf{u}) \frac{\partial \mathbf{u}}{\partial X} = 0 \quad (6)$$

where

$$r_{\mathbf{u}} F(\mathbf{u}) = \begin{bmatrix} 0 & 1 \\ \frac{Q^2}{A^2} + \frac{1}{2}A^{\frac{1}{2}} & \frac{2Q}{A} \end{bmatrix} \quad (7)$$

and serves as the jacobian matrix of $F(\mathbf{u})$. To linearize $F(\mathbf{u})$ about $[A_0, 0]$, we need to take a Taylor expansion of $F(\mathbf{u})$. This means that

$$F(\mathbf{u}) = F(\mathbf{u}_0) + r_{\mathbf{u}} F(\mathbf{u}_0)(\mathbf{u} - \mathbf{u}_0) + \dots = 0 + r_{\mathbf{u}} F(\mathbf{u}_0) \frac{\partial \mathbf{u}}{\partial X} + \dots \quad (8)$$

where our primary interest will lie within the first two terms. Substituting the results of (7) into equation (4), we get our desired linearized equation of

$$\frac{\partial \mathbf{u}}{\partial t} + r_{\mathbf{u}} F(\mathbf{u}_0) \frac{\partial \mathbf{u}}{\partial X} = 0 \quad (9)$$

and

$$r_{\mathbf{u}} F(\mathbf{u}_0) = \begin{bmatrix} 0 & 1 \\ \frac{1}{2}A_0^{\frac{1}{2}} & 0 \end{bmatrix} \quad (10)$$

we can add the columns together to get functions of A and Q . Thus we can get

$$r_{\mathbf{u}} F(\mathbf{u}_0) = \begin{bmatrix} F'(A) \\ F'(Q) \end{bmatrix} = \begin{bmatrix} 1 \\ \frac{1}{2}A_0^{\frac{1}{2}} \end{bmatrix} \quad (11)$$

From this, we can get our linearized $F(\mathbf{u})$ to be:

$$r_{\mathbf{u}} F(\mathbf{u}_0) = \begin{bmatrix} F(A) \\ F(Q) \end{bmatrix} = \begin{bmatrix} Q \\ \frac{1}{2}A_0^{\frac{1}{2}}A \end{bmatrix} \quad (12)$$

2.2 Riemann Invariant

The equation above satisfies all the interior points of our domain x . However, we have to consider how to model our boundary points. Riemann Invariants are applicable throughout the interval that we are considering. Most importantly, we can derive our boundary conditions based on these equations. Since there are two boundaries on our blood vessel, where blood flows in and where blood flows out, there are two equations are of the form:

$$\frac{\partial w_1}{\partial t} + \lambda_1 \frac{\partial w_1}{\partial x} = 0 \quad (13)$$

$$\frac{\partial w_2}{\partial t} + \lambda_2 \frac{\partial w_2}{\partial x} = 0 \quad (14)$$

where w_1 corresponds to where the blood flows in, and where w_2 corresponds to where blood flows out. In both of these cases, λ_1 and λ_2 are function of A and Q . In general, we need these equations to determine how to calculate boundary conditions that are consistent with values that are determined in equation (8). This means that we want to trace the points along the boundary back along their characteristic curves to determine their solution.

We now need to calculate our eigenvalues, which is done as follows:

$$(r_{\mathbf{u}} F(\mathbf{u}_0) - \mathbf{I}) = \begin{bmatrix} 0 & 1 \\ \frac{1}{2} A_0^{\frac{1}{2}} & 0 \end{bmatrix} \quad (15)$$

By calculating the determinant, we get the following eigenvalues.

$$\lambda_1 = C(A_0) = \left(\frac{1}{2}\right)^{\frac{1}{2}} A_0^{\frac{1}{4}} \quad (16)$$

$$\lambda_2 = -C(A_0) = -\left(\frac{1}{2}\right)^{\frac{1}{2}} A_0^{\frac{1}{4}} \quad (17)$$

From these eigenvalues, we get the the following eigenvectors:

$$l_1 = \left[\frac{C(A_0)}{A_0}, \frac{1}{A_0} \right] \quad (18)$$

$$l_2 = \left[-\frac{C(A_0)}{A_0}, \frac{1}{A_0} \right] \quad (19)$$

Now we will show that these are the eigenvalues and functions for l_1 . It will follow for l_2 . We have to test these functions against equation (10).

Proof:

$$l_1 \begin{bmatrix} 0 & 1 \\ \frac{1}{2} A_0^{\frac{1}{2}} & 0 \end{bmatrix} = \left[\frac{C(A_0)}{A_0}, \frac{1}{A_0} \right] \begin{bmatrix} 0 & 1 \\ \frac{1}{2} A_0^{\frac{1}{2}} & 0 \end{bmatrix} = \left[\frac{C(A_0)^2}{A_0}, \frac{C(A_0)}{A_0} \right] = C(A_0) l_1 \quad (20)$$

Thus, we see that $C(A_0)$ is an eigenvalue and l_1 is the corresponding function. Moreover, from equations (18) and (19), we can deduce

$$w_1 = \frac{Q}{A_0} + \frac{C(A_0)}{A_0} A \quad (21)$$

$$w_2 = \frac{Q}{A_0} - \frac{C(A_0)}{A_0} A \quad (22)$$

These equations are our characteristic equations for the advection partial differential equation that we are trying to solve in equation (9). These will help us calculate boundary conditions of our model. This will be useful in calculating the boundary conditions for our partial differential equation.

3 Numerical Scheme

3.1 Model

We are modeling the blood vessel as a tube with a cross sectional area throughout the vessel. Moreover, we apply the Lax-Wendroff Method to solve for flow and area within the vessel.

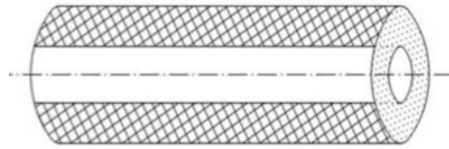


Fig. 2. This is a drawing of our theoretical vessel

3.2 Two-Step Lax-Wendroff Scheme

We are using this scheme primarily because it provides greater accuracy and stability compared to the forward Euler method. This is because Lax-Wendroff is 2nd order stable which means errors are on the order of Δx^2 and Δt^2 .

To implement this scheme, the first step is to calculate both:

$$u_{j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{2}(u_{j+1}^n + u_j^n) - \frac{t}{x}(F(u_{j+1}^n) - F(u_j^n)) \quad (23)$$

$$u_j^{n+\frac{1}{2}} = \frac{1}{2}(u_j^n + u_{j-\frac{1}{2}}^n) - \frac{t}{x}(F(u_j^n) - F(u_{j-\frac{1}{2}}^n)) \quad (24)$$

The second step is to calculate the actual step:

$$u_j^{n+1} = u_j^n - \frac{t}{x} (F(u_{j+\frac{1}{2}}^{n+\frac{1}{2}}) - F(u_{j-\frac{1}{2}}^{n+\frac{1}{2}})) \quad (25)$$

where n indicates time steps and j indicates grid points. Note that we first calculate half grid points before we calculate the actual grid points. This is how we maintain a high level of accuracy within our results.

3.3 Riemann Invariants

To calculate the boundary conditions, we have to calculate the flow (Q) on both sides.

$$Q_{left}^{n+1} = A_0(w_2) + A_{in}^{n+1} \sqrt{\frac{w_2}{2}} A_0^{\frac{1}{4}} \quad (26)$$

$$Q_{right}^{n+1} = A_0(w_1) - A_{out}^{n+1} \sqrt{\frac{w_1}{2}} A_0^{\frac{1}{4}} \quad (27)$$

These equations come from w_1 and w_2 that we "derived". Notice how they are applied. We need to use w_1 for the flow that is on the right because the eigenvalue shows us that w_1 have a positive slope. All solutions are the same along each individual w_1 and w_2 since they are characteristic lines. This means for some point on the right boundary and w_1 , we can get the solution by tracing back along the characteristic line. The same applies to w_2 . Since this eigenvalue is negative, we know that it has to solve the left boundary.

Thus to formally get a solution for these boundary points, we have to interpolate where the characteristic is within our grid of space. Moreover, to ensure that these characteristics actually intersect within our boundary, we have to ensure that the CFL condition is satisfied. For our case, we set our Δx and Δt so that this condition is met. Most importantly, by increasing Δx we must automatically shrink Δt so that we maintain a consistent number that is below our CFL condition.

4 Results

4.1 Singular Pulse

We first must test that this model works properly. To begin, we want to send a singular sinusoidal pulse and see what happens. Thus, we impose the following conditions:

$$A_{in} = \left(\frac{\sin(t \cdot t)}{2} + A_0^{\frac{1}{2}} \right)^2$$

$$A_{out} = A_0$$

such that we only send a singular sinusoidal pulse from A_{in} . Moreover, we apply a to be 10, b to be 1, and p_0 to be 0.

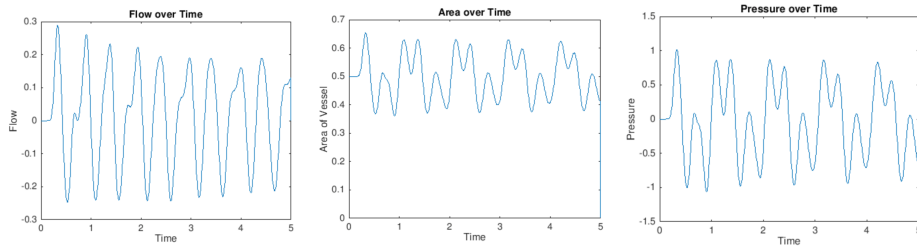


Fig. 3. For time $t=5$, 30 points

From figure (3), we see that area and pressure seem to be constant. However, flow seems to be decaying. This might be from not using enough points. In this figure, we are using 30 points. Thus, we will now test for the same time $t=5$, but with more points, namely 100. The results are as follows:

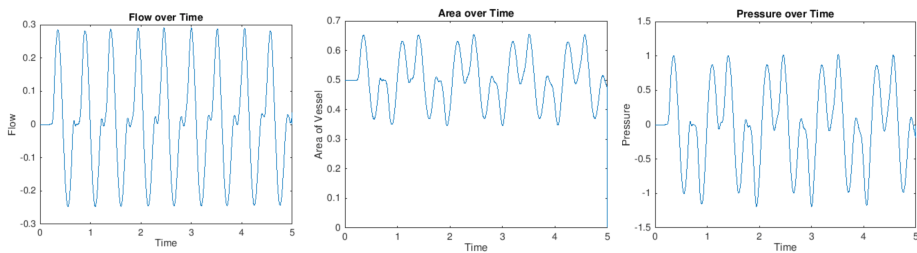


Fig. 4. For time $t=5$, 100 points

From figure (4), it seems that flow is constant within this test. Now, we will increase the time to 20 and maintain the number of points. We will now see if the results still hold. They do not hold as flow seems to still decrease over time as seen in figure (5).

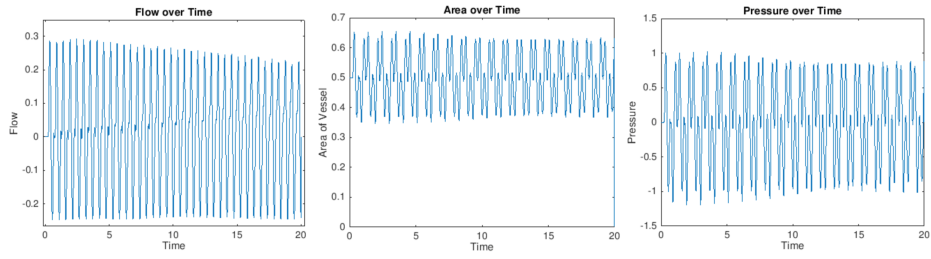


Fig. 5. For time $t=20$, 100 points

4.2 Womersley Pressure

We will now apply a sinusoidal pressure gradient across this boundary. Thus we use the same area conditions as seen in section 4.1, but we do not restrict A_{in} to a singular wave. To begin, we will test this for 30 points.

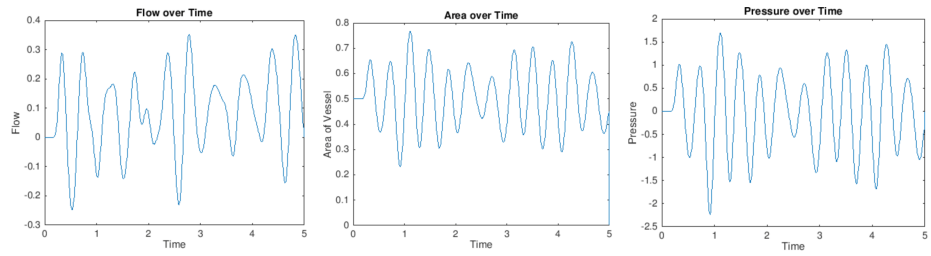


Fig. 6. For time $t=5$, 30 points

By applying a Womersley pressure gradient, it seems that flow is constant over time along with area and pressure, according to figure (6). We will now apply this condition for 100 points and for a time $t=20$.

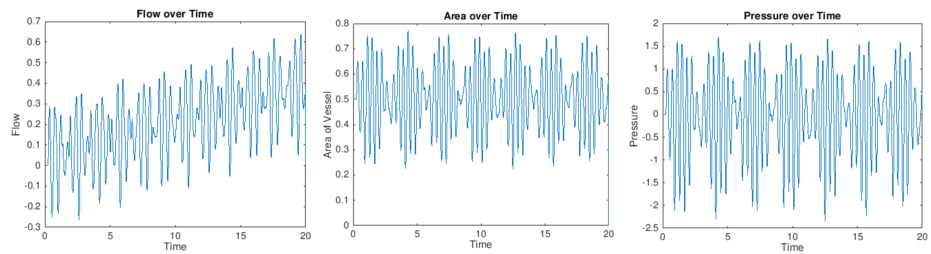


Fig. 7. For time $t=20$, 100 points

From figure (7), it now becomes evident that flow is not stable. It seems that flow increases over time, while area and pressure both oscillate around a steady state. We will now see what happens when we change beta. We now increase beta to 100, and maintain the same time period and number of points, namely 20 and 100 respectively.

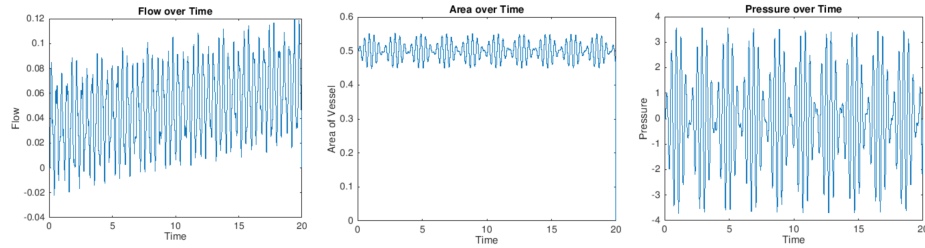


Fig. 8. For time $t=20$, 100 points

From figure (8), it appears that as we increase beta, we increase flow variability, but we decrease area variability. This means that beta can represent the stiffness of our vessel. Now, we look at what happens to when we change ρ . We will now set ρ to be 15.

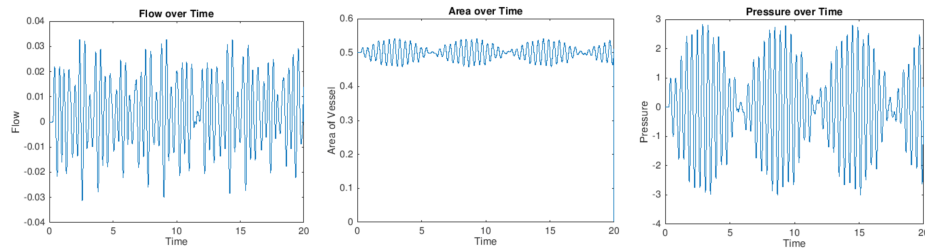


Fig. 9. For time $t=20$, 100 points

According to figure (9), it seems that as we increase ρ , we "stretch" out the pulses in area and pressure, and make them appear more standard. This could be equivalent to "getting a slower heart beat", since the heart is working harder to pump fluid. Thus ρ could represent density of blood. Moreover, flow looks as though it is constant, however, with further tests using $t=100$, flow is still increasing.

5 Comments

5.1 Possible Sources of Error

It seems that we might have made a bad assumption when we assumed α to be 0 in equation (2); that term might have been the term that "restrained" flow from increasing over time.

The error we observe in flow could be a numerical instability, since in figures (4),(6), and (9) the flow appeared to be stable. Given the fact that area fluctuates, it could mean that the decreases in flow come from moments when we violate the CFL condition. Moreover, the decreases in flow could mean that we simply haven't chosen a time step small enough for the length of time we prescribed.

5.2 Further Study

For further study, we might construct a system of vessels such that they form a complete system, just like the circulatory system. We could track the flow, pressure, and area of each vessel.

For further study, we might assume that some sort of disease such as hypertension. Also, we could also study possible shocks that can happen to blood vessels. A possible shock is distributive shock, which is a decrease in blood flow. This would require us to prescribe a flow instead of area. This might mean that area is not stable. Moreover, we could look diseases such as heart arrhythmia. This would require us to prescribe a pressure that consists of other sine and cosine terms.

Since these sine waves are interacting with one another in unknown ways, we could change the area on the right side of the boundary so that it can "absorb" some of the flow.

Another study that we could do is to make this flow multidimensional. Currently, we are only treating this flow in one dimension. We could do two dimensions or three dimensional flow.

5.3 Acknowledgements

We are grateful for the guidance and help provided by Charles Edward Puelz, much thanks for his generous support and advice throughout this project. His work may be found here: <https://cpuelz.github.io/index.html>.

6 Matlab Code

```
close all; clc;

%initial parameters
Nx = 100;
xmi n = 0;
xmax = 1;
tmi n = 0;
tmax = 100;
period = 0.4;
omega = 2*pi /period;
safety = 0.45; % have this less than 0.5

%conditions
A0 = .5;
beta = 10;
rho = 1;
P0=0;

c = sqrt(beta/(2*rho))*A0^(1/4);
FA = @(A, Q) Q;
FQ = @(A, Q) (beta/(2*rho))*sqrt(A0)*A;

%mesh in time and space
dx = (xmax-xmi n)/(Nx+2);
dt = safety*dx/c;
Nt = floor(tmax/dt);

%Riemann invariant
l1 = c;
l2 = -c;
W1 = @(A, Q) ((Q/A0) + (c/A0)*A);
W2 = @(A, Q) ((Q/A0) - (c/A0)*A);

xvec = linspace(xmi n, xmax, Nx+2);
tvec = linspace(tmi n, Nt*dt, Nt);

Avec = zeros(Nx+2, 1);
Qvec = zeros(Nx+2, 1);
Avec_ol d = zeros(Nx+2, 1);
Qvec_ol d = zeros(Nx+2, 1);

Avec_poshal f = zeros(Nx+2, 1);
Qvec_poshal f = zeros(Nx+2, 1);
Avec_neghal f = zeros(Nx+2, 1);
```

```

Qvec_neghalf = zeros(Nx+2, 1);

%initial condition
for x = 1:length(xvec)
    Avec(x, 1) = A0;
    Qvec(x, 1) = 0.0;
end

Qmat=zeros(1, length(tvec));
Amat=zeros(1, length(tvec));
Pmat=zeros(1, length(tvec));

%Lax-Wendroff Scheme 2 step
for t = 1:length(tvec)-1

    disp(tvec(t))

    % compute extrapolated Riemann invariants
    xleft = xmin-l2*dt; % computing the tails of the characteristic curves.
    xright = xmax-l1*dt;
    Aleft = interp1(xvec', Avec(:, 1), xleft);
    Qleft = interp1(xvec', Qvec(:, 1), xleft);
    Aright = interp1(xvec', Avec(:, 1), xright);
    Qright = interp1(xvec', Qvec(:, 1), xright);
    W2left = W2(Aleft, Qleft);
    W1right = W1(Aright, Qright);

    % populate boundary conditions
    Qvec(1, 1) = A0*W2left + Avec(1, 1)*c;
    Qvec(end, 1) = A0*W1right - Avec(end, 1)*c;

    % store previous timestep
    Avec_old = Avec;
    Qvec_old = Qvec;

    %interior points
    for x = 2:length(xvec)-1
        Avec_poshalf(x, 1) = .5*(Avec_old(x+1, 1)+Avec_old(x, 1))-(dt/(2*dx))*
            (FA(Avec_old(x+1, 1), Qvec_old(x+1, 1))-FA(Avec_old(x, 1), Qvec_old(x, 1)));
        Avec_neghalf(x, 1) = .5*(Avec_old(x, 1)+Avec_old(x-1, 1))-(dt/(2*dx))*
            (FA(Avec_old(x, 1), Qvec_old(x, 1))-FA(Avec_old(x-1, 1), Qvec_old(x-1, 1)));

        Qvec_poshalf(x, 1) = .5*(Qvec_old(x+1, 1)+Qvec_old(x, 1))-(dt/(2*dx))*
            (FQ(Avec_old(x+1, 1), Qvec_old(x+1, 1))-FQ(Avec_old(x, 1), Qvec_old(x, 1)));
    }
}

```

```

        Qvec_neghalf(x, 1) = .5*(Qvec_ol d(x, 1)+Qvec_ol d(x-1, 1))-(dt/(2*dx))*
        (FQ(Avec_ol d(x, 1), Qvec_ol d(x, 1))-FQ(Avec_ol d(x-1, 1), Qvec_ol d(x-1, 1)));
    end
    for x = 2:length(xvec)-1
        Avec(x, 1) = Avec_ol d(x, 1)-(dt/dx)*(FA(Avec_poshalf(x, 1), Qvec_poshalf(x, 1))
        -FA(Avec_neghalf(x, 1), Qvec_neghalf(x, 1)));
        Qvec(x, 1) = Qvec_ol d(x, 1)-(dt/dx)*(FQ(Avec_poshalf(x, 1), Qvec_poshalf(x, 1))
        -FQ(Avec_neghalf(x, 1), Qvec_neghalf(x, 1)));
    end

% set boundary conditions for A
if tvec(t)<period
    Avec(1, 1) = ((sin(omega*tvec(t+1))/beta-P0/beta) + sqrt(A0))^2;
    Avec(end, 1) = A0;
else
    Avec(1, 1)=A0;
    Avec(end, 1)=A0;
end

%Avec(1, 1) = ((sin(omega*tvec(t+1))/beta-P0/beta) + sqrt(A0))^2;
%Avec(end, 1) =A0;

%Calculate flow in the future
Qvec(1, 1) = A0*W2left + Avec(1, 1)*c;
Qvec(end, 1) = A0*W1right - Avec(end, 1)*c;

figure(1)
subplot(2, 1, 1), plot(xvec, Qvec(:, 1), 'linewidth', 4);
ylim([-1 3])
title("Flow v. Position");
xlabel ("Position");
ylabel ("Flow");
subplot(2, 1, 2), plot(xvec, Avec(:, 1), 'linewidth', 4);
ylim([0 2])
title("Area v. Position");
xlabel ("Position");
ylabel ("Area of Vessel");
pause(0.001)

Qmat(1, t)=Qvec(floor(Nx/2), 1);
Amat(1, t)=Avec(floor(Nx/2), 1);
Pmat(1, t)=P0 + beta.*(sqrt(Avec(floor(Nx/2), 1)) - sqrt(A0));
end

```

```
figure(2)
plot(tvec,Qmat(1,:)); hold on
title("Flow over Time");
xlabel("Time");
ylabel("Flow");
```

```
figure(3)
plot(tvec,Amat(1,:)); hold on
title("Area over Time");
xlabel("Time");
ylabel("Area of Vessel");
```

```
figure(4)
plot(tvec,Pmat(1,:)); hold on
title("Pressure over Time");
xlabel("Time");
ylabel("Pressure");
```