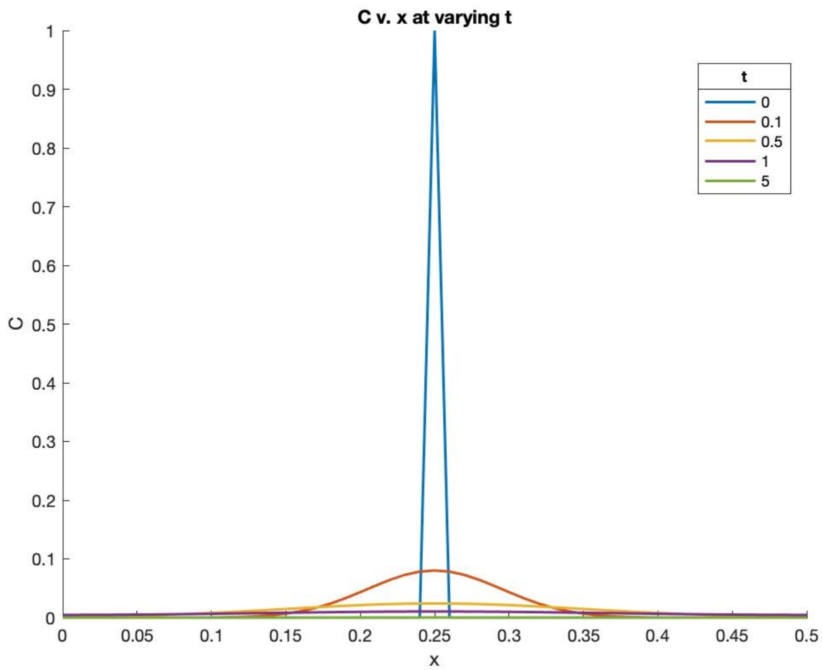
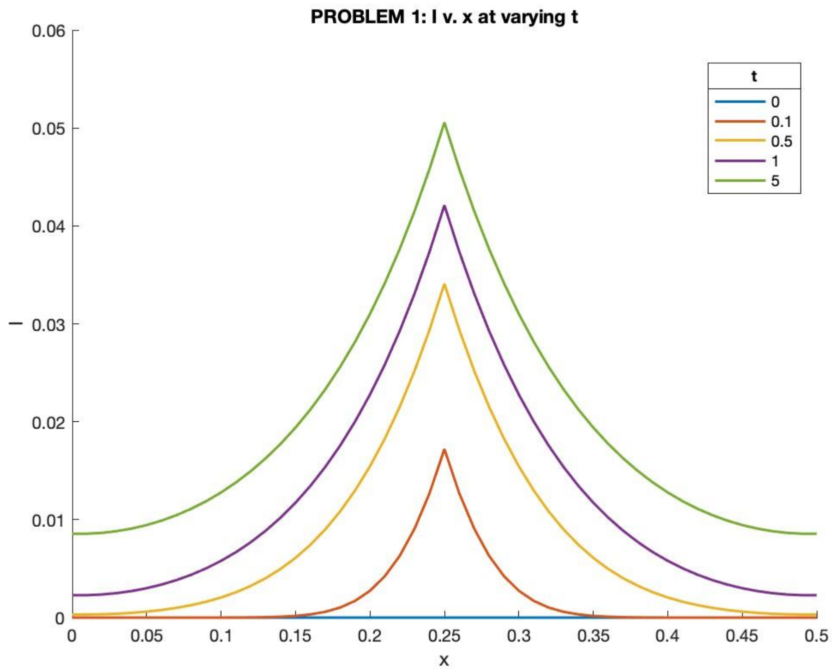
Robert Heeter
BIOE 446
10 November 2023

Lab 11: Excitatory Dynamics in Cardiac Cells and Tissue

ANSWERS

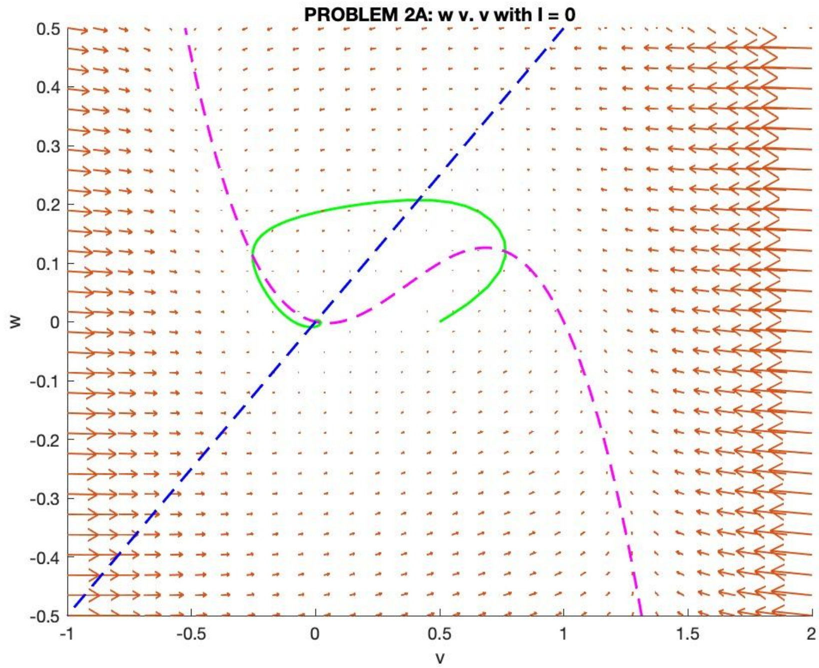
1. **Simulation of partial differential equations in MATLAB (DEMO)**
   a. Figures

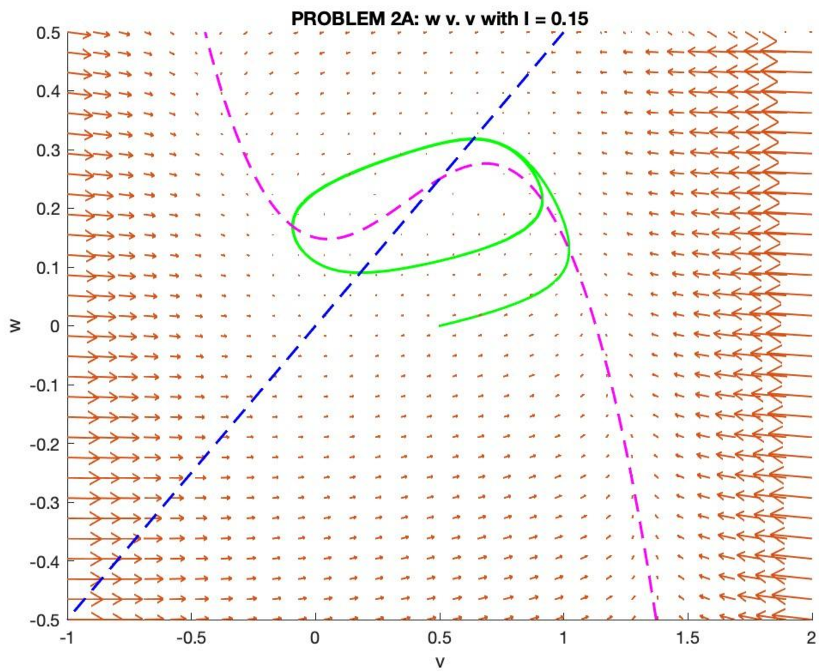


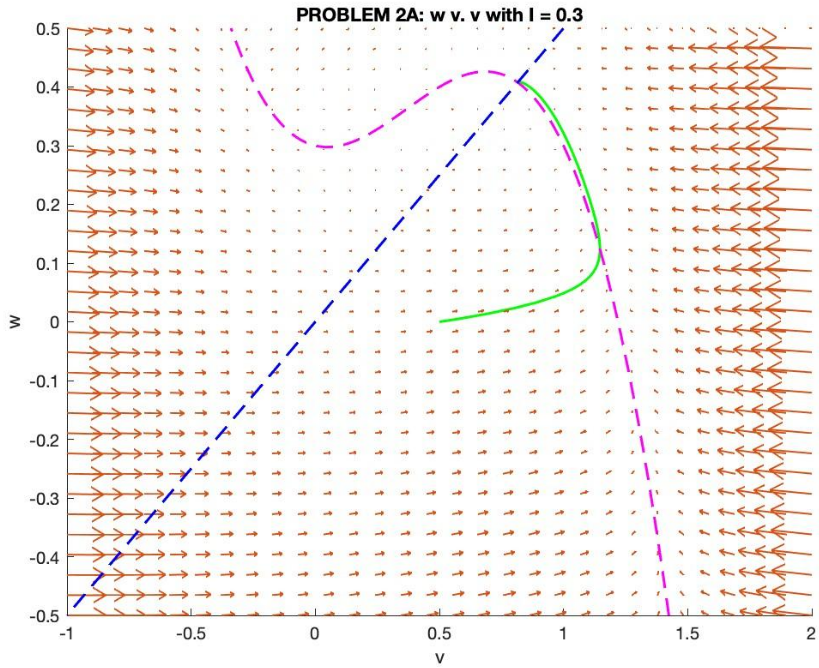


As time passes, the concentration of the species C spreads out spatially along x (i.e., C diffuses outwards), shown as the large peak of C at t = 0 quickly flattens out over time. This diffusion is symmetric about x = 0.25. The concentration of the species I increases at all space points as C reacts to form I over time, though I has a peak at x = 0.25 (corresponding to the highest concentration of C).

2. **Simulation of Fitzhugh-Nagumo model**
    a. Figures


PROBLEM 2A: w v. v with I = 0
w
v
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
-1
-0.5
0
0.5
1
1.5
2


PROBLEM 2A: w v. v with I = 0.15
w
v
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
-1
-0.5
0
0.5
1
1.5
2

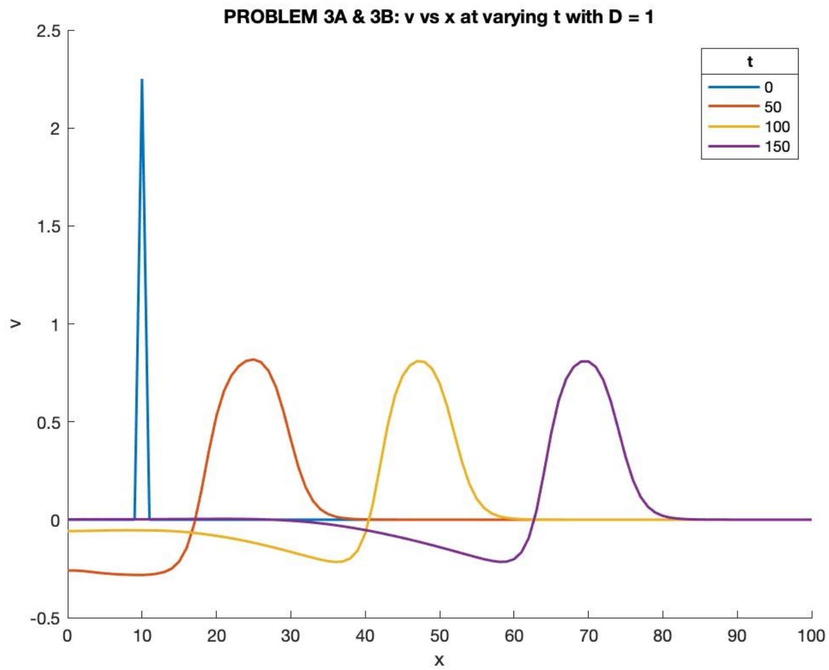


As the applied current on the system (I) increases across the 3 figures, both w and v reach or oscillate around a steady-state (intersection of the nullclines) that is larger in both dimensions. The simulations for I = 0 and I = 0.3 do not produce any oscillations; w and v reach the steady-state intersection of the nullclines. The simulation for I = 0.15 (and likely applied currents around 0.15 as well) produces oscillations in the simulated time range, seen as the green line circling the steady-state solution.

b. Critical epsilon = 16.1100 to stop oscillations.

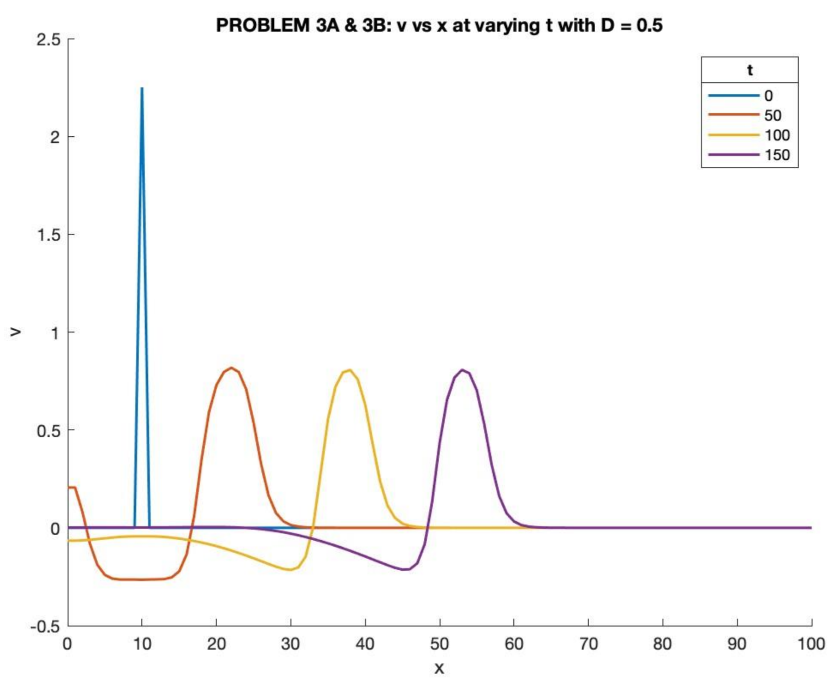
Increasing the value of epsilon increases the rate of the system response, hence producing weaker oscillations. At the critical epsilon value, the system is damped and the oscillations stop.

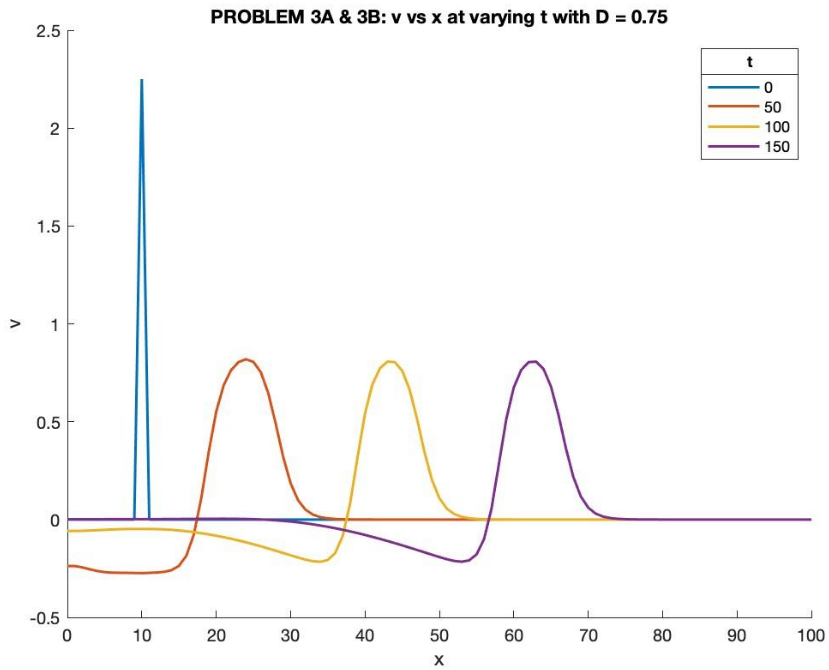
3. **Simulation of the Fitzhugh-Nagumo model in a 1D spatial system**
   a. Figure

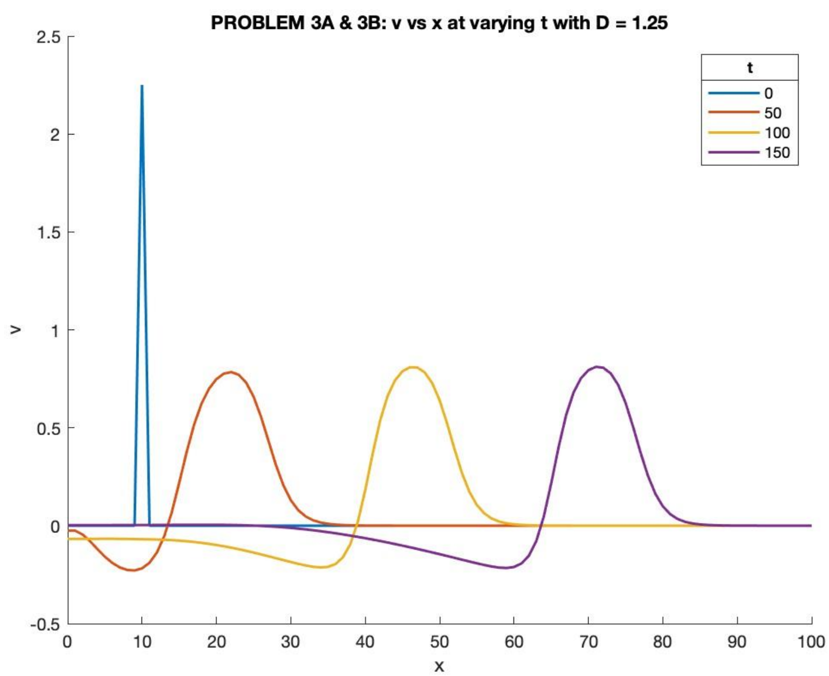


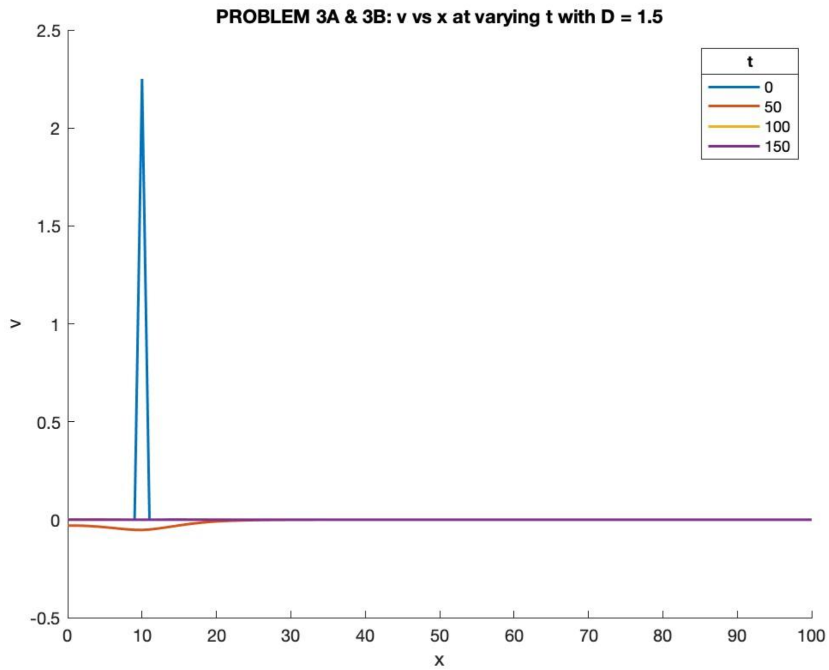
According to the results in the figure, the signal appears to propagate through the tissue as after the initial stimulus, the signal wave remains at a fixed amplitude/magnitude and propagates to larger x-values with time (i.e., the signal does not lose "strength").

b. Figures




PROBLEM 3A & 3B: v vs x at varying t with D = 0.75
t
0
50
100
150
v
x


PROBLEM 3A & 3B: v vs x at varying t with D = 1.25
t
0
50
100
150
v
x



A diffusivity of D = 1.5 prevents the signal from propagating, shown as the waveform does not translate/move from the initial stimulus at t = 0. At a large diffusivity value, the signal diffuses so much (i.e., loses so much strength) that nearby cells are not stimulated/cannot react to the signal. These cells likely require a significant strength signal to react. Overall, the signal is unable to propagate through the tissue.

CODE

```
close all
clc

%% Problem 1: Simulation of partial differential equations in MATLAB (DEMO)
disp('PROBLEM 1')
clear

%% Problem 1, Part A
% set up parameters
D = 0.01;
k = 1;

% create our concentration and time vectors
dx = 0.01;
dt = 0.0025;
x = 0:dx:0.5;
t = 0:dt:5;

% initialize C and I matrices
C = zeros(length(t), length(x));
I = C;

% set initial conditions
C(1,26) = 1;

% iterate through each point and calculate concentrations using discretized
% PDE

for ii = 2:length(t)
    for jj = 2:length(x)-1
        C(ii,jj) = (D*dt/dx^2)*(C(ii-1,jj-1) - 2*C(ii-1,jj) + C(ii-1,jj+1)) -
k*dt*C(ii-1,jj) + C(ii-1,jj);
        I(ii,jj) = C(ii-1,jj)*k*dt + I(ii-1,jj);
    end

    % setting concentrations at boundaries based on no flux boundary
    % condition
    C(ii,1) = C(ii,2);
    C(ii,end) = C(ii,end-1);
    I(ii,1) = I(ii,2);
    I(ii,end) = I(ii,end-1);
end

% set timepoints to plot
t_plot = [0 0.1 0.5 1 5];

figure(1)
hold on
for ii = 1:length(t_plot)
    plot(x, C(t==t_plot(ii), :), LineWidth=1.5)
```

```
end
title('C v. x at varying t')
xlabel('x')
ylabel('C')
l = legend(string(t_plot), Location='best');
title(l,'t')

figure(2)
hold on
for ii = 1:length(t_plot)
    plot(x, I(t==t_plot(ii), :), LineWidth=1.5)
end
title('PROBLEM 1: I v. x at varying t')
xlabel('x')
ylabel('I')
l = legend(string(t_plot), Location='best');
title(l,'t')


%% Problem 2: Simulation of Fitzhugh-Nagumo model
disp('PROBLEM 2')
clear

%% Problem 2, Part A
a = 0.1;
b = 0.05;
gamma = 0.1;
I_set = [0, 0.15, 0.3];
tspan = [0 500];

v0 = 0.5;
w0 = 0;
y0 = [v0 w0];

dvdt = @(v, w, I) (v.*(a-v).*(v-1)) - w + I; %% HH model with I as variable
dwdt = @(v, w) (b.*v) - (gamma.*w);

for i = 1:length(I_set)
    I = I_set(i);
    [t, y] = ode45(@(t, y) [dvdt(y(1), y(2), I); dwdt(y(1), y(2))], tspan, y0);

    figure(i+2)
    hold on
    plot(y(:, 1), y(:, 2), '-g', LineWidth=1.5);

    l = linspace(-1, 2, 30);
    m = linspace(-0.5, 0.5, 30);
    [L, M] = meshgrid(l, m);

    dvdt2 = @(v, w) (v.*(a-v).*(v-1)) - w + I; %% HH model with I as constant
    dwdt2 = @(v, w) (b.*v) - (gamma.*w);
```

```
    quiver(L, M, dvdt2(L, M), dwdt2(L, M), AutoScaleFactor=2, LineWidth=1)
    fimplicit(dvdt2, '--m', 'LineWidth', 1.5)
    fimplicit(dwdt2, '--b', 'LineWidth', 1.5)
    xlim([-1, 2])
    ylim([-0.5, 0.5])

    title(['PROBLEM 2A: w v. v with I = ', num2str(I)]);
    xlabel('v')
    ylabel('w')
end

%% Problem 2, Part B
I = 0.125;
eps = 1;
difference = 1;

v0 = 0.5;
w0 = 0;
y0 = [v0 w0];

while difference >= 0.01
    b = 0.01*eps;
    gamma = 0.02*eps;

    dvdt = @(v, w, I) (v*(a-v)*(v-1)) - w + I; %% HH model
    dwdt = @(v, w) (b*v) - (gamma*w);

    [t, y] = ode45(@(t, y) [dvdt(y(1), y(2), I); dwdt(y(1), y(2))], tspan, y0);
    v = y(:, 1);

    difference = max(abs((v(end-9:end) - v(end))./v(end)));

    eps = eps + 0.01;
end

disp(eps)
disp(difference)


%% Problem 3: Simulation of the Fitzhugh-Nagumo model in a 1D spatial system
disp('PROBLEM 3')
clear

%% Problem 3, Parts A and B
a = 0.1;
b = 0.01;
gamma = 0.02;
I = 0;
D_set = [0.5, 0.75, 1, 1.25, 1.5];
```

```
dx = 1;
dt = 0.025;
x = 0:dx:100;
t = 0:dt:200;

v = zeros(length(t), length(x));
w = v;

v(1, 11) = 2.25;

% simulation same as in Problem 1 but with new equations
for j = 1:length(D_set)
    for ii = 2:length(t)
        D = D_set(j);

        for jj = 2:length(x)-1
            v(ii, jj) = (D*dt/dx^2)*(v(ii-1, jj-1) - 2*v(ii-1,jj) + v(ii-1, jj+1)) +
I*dt - w(ii-1,jj)*dt + dt*v(ii-1,jj)*(a-v(ii-1,jj))*(v(ii-1,jj)-1)+v(ii-1,jj);
            w(ii, jj) = dt*b*v(ii-1,jj)-dt*gamma*w(ii-1,jj)+w(ii-1,jj);
        end

        v(ii, 1) = v(ii, 2);
        v(ii, end) = v(ii, end-1);
        w(ii, 1) = w(ii, 2);
        w(ii, end) = w(ii, end-1);
    end

    t_plot = [0 50 100 150];

    figure(j+5)
    hold on
    for ii = 1:length(t_plot)
        plot(x, v(t == t_plot(ii), :), 'LineWidth', 1.5);
    end
    title(['PROBLEM 3A & 3B: v vs x at varying t with D = ', +  num2str(D)]);
    xlabel('x');
    ylabel('v');
    l = legend(string(t_plot), 'Location', 'best');
    title(l, 't');
end


%% Functions

% None
```