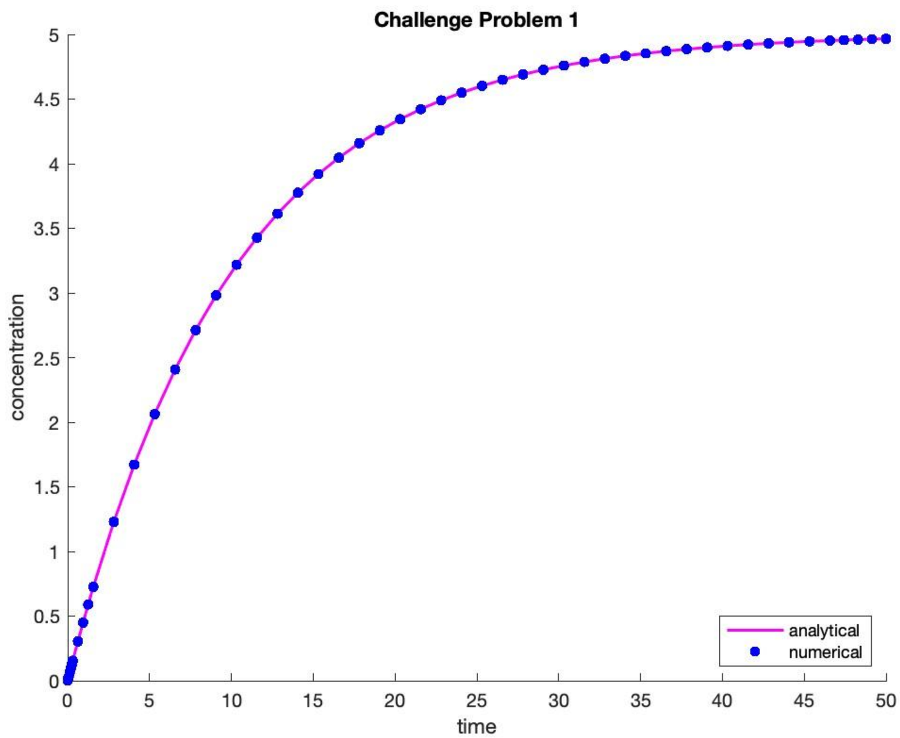
Robert Heeter
BIOE 446
15 September 2023

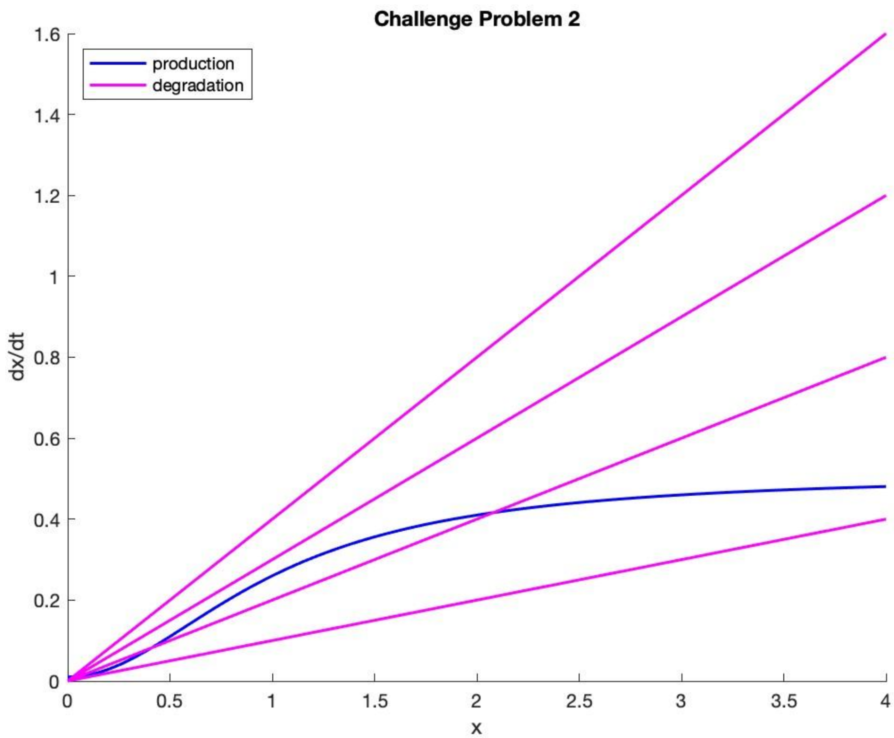
Lab 3: Biochemical Circuits: Bistability

CHALLENGE PROBLEMS

1. Figure



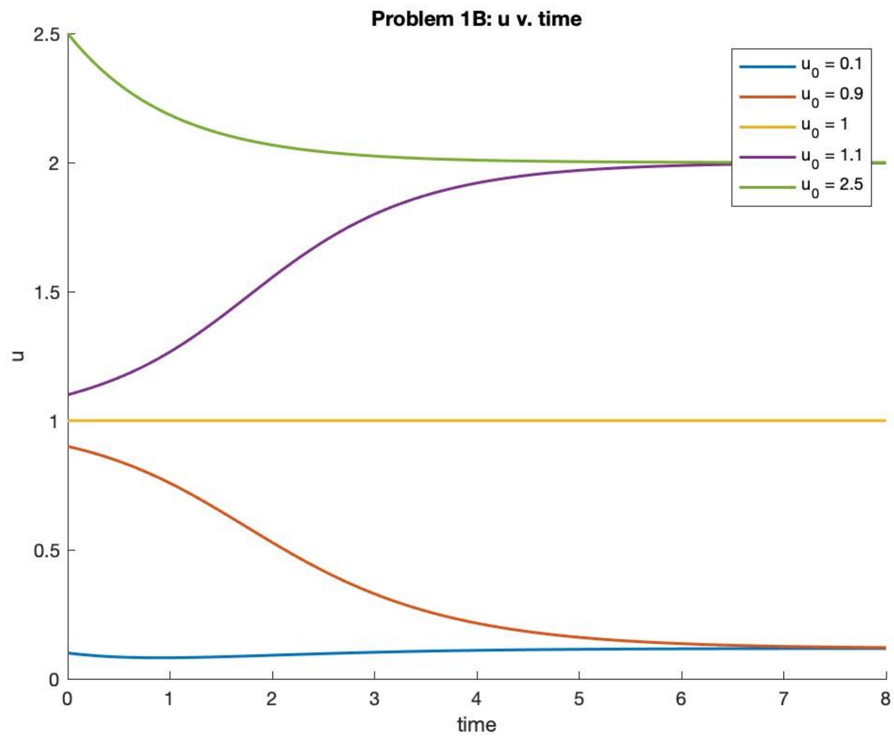
2. Figure

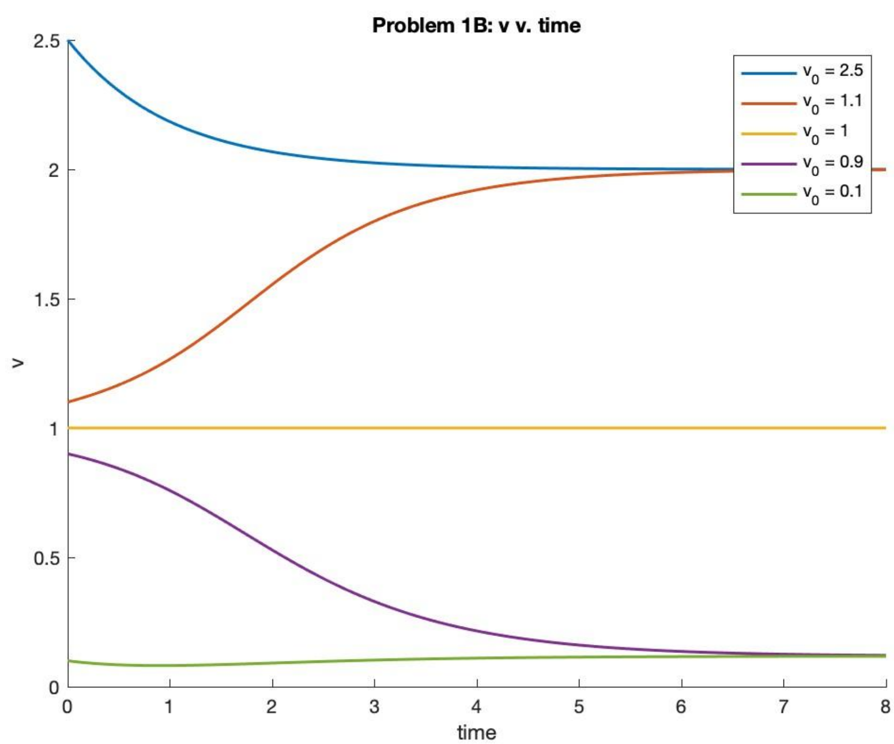


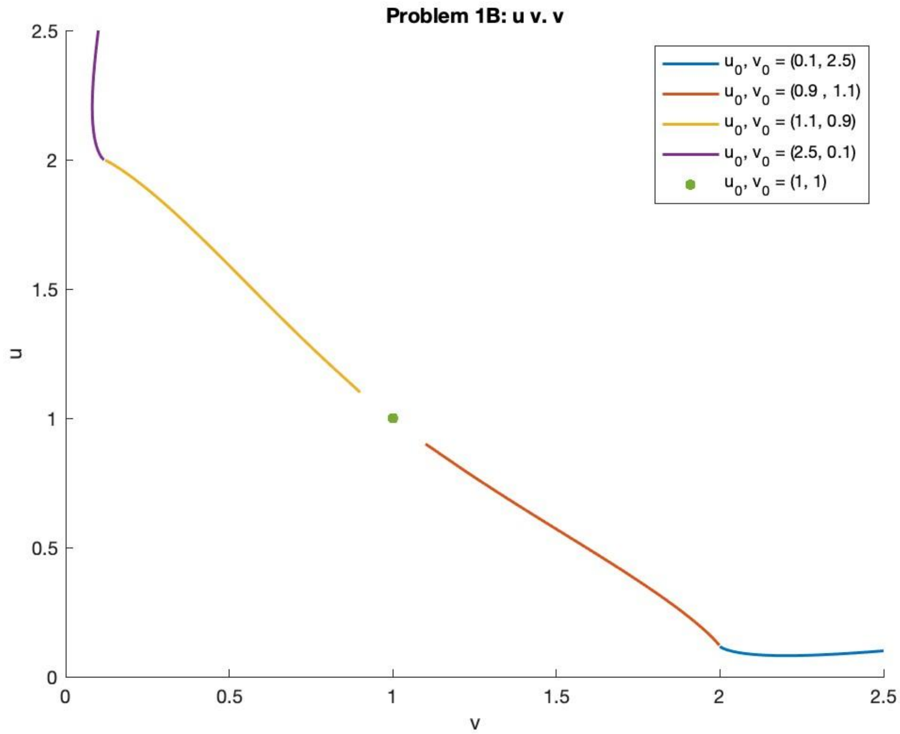
ANSWERS

1. **Bistability of the toggle swtich network (DEMO)**
   a. du/dt = 0.02
      dv/dt = -0.00

   b. Figures

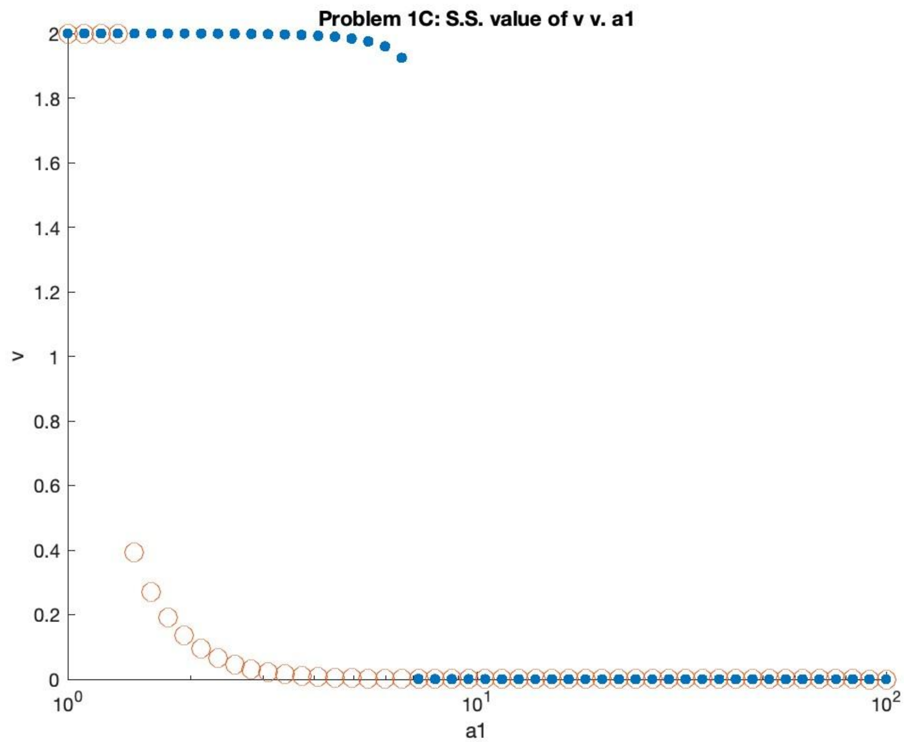

Problem 1B: u v. time
u
time
u_0 = 0.1
u_0 = 0.9
u_0 = 1
u_0 = 1.1
u_0 = 2.5
0
0.5
1
1.5
2
2.5
0
1
2
3
4
5
6
7
8


Problem 1B: v v. time
v
time
v_0 = 2.5
v_0 = 1.1
v_0 = 1
v_0 = 0.9
v_0 = 0.1
0
0.5
1
1.5
2
2.5
0
1
2
3
4
5
6
7
8



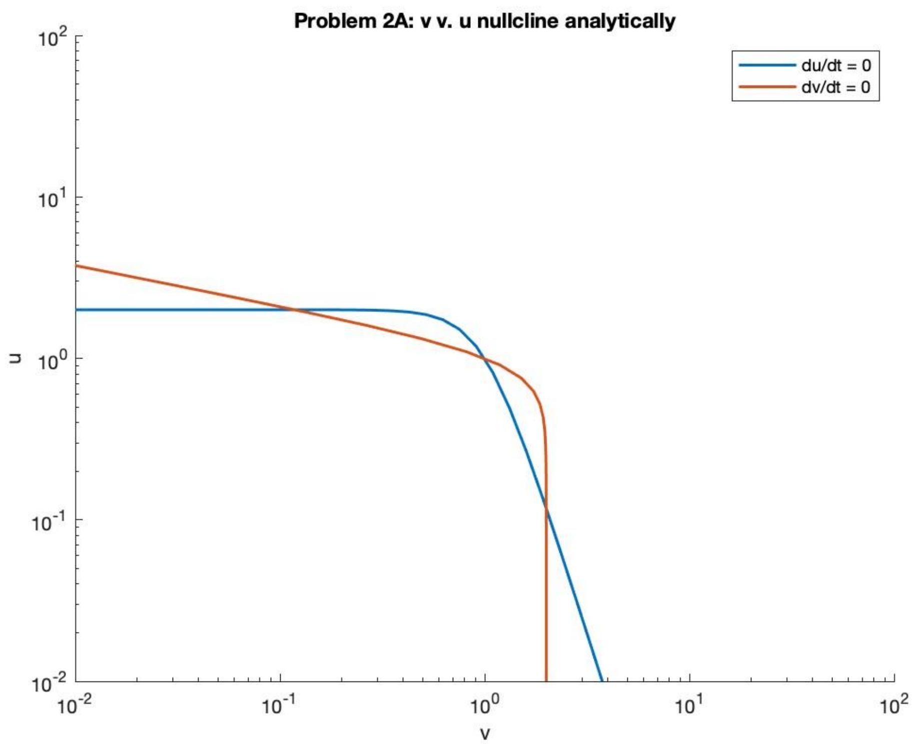
The initial conditions affect which of the three steady states the system converges to. u < 1 approaches a steady state of u ~ 0.1, u > 1 approaches a steady state of u ~ 2. v < 1 approaches a steady state of v ~ 0.1, v > 1 approaches a steady state of v ~ 2. When u = 1 or v = 1, the system remains in an unsteady equilibrium. The system has three steady states, though the steady states where (u, v) = (2, 0.1) or (u, v) = (0.1, 2) are stable equilibria, whereas (u, v) = (1, 1) is an unsteady equilibrium since any small perturbation away from this point causes the system to drift to one of the other two steady equilibria.

c. Figure

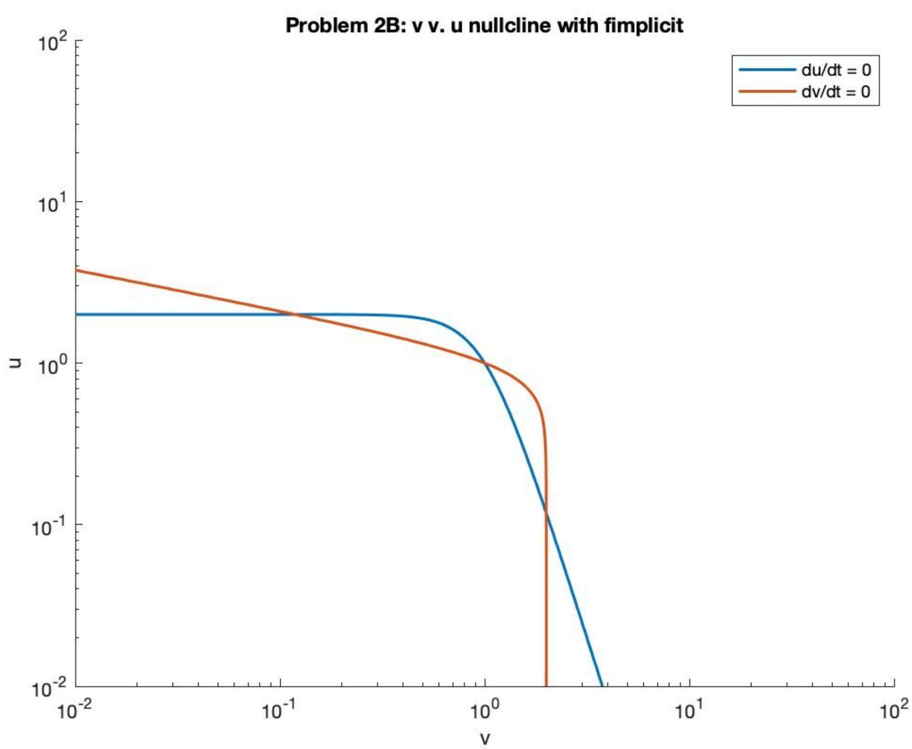


The system appears to be bistable between values of 1.4 < a1 < 6.6, where the plot shows that there are two steady states depending on which direction (forward or backward) v is changing. At a1 ~ 1.4 or a1 ~ 6.6, the system switches abruptly to the other steady state as seen on the graph.

2. **Generation and analysis of nullcline plots**
   a. Figure


Problem 2A: v v. u nullcline analytically
du/dt = 0
dv/dt = 0
u
v
10^{2}
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}

b. Figure

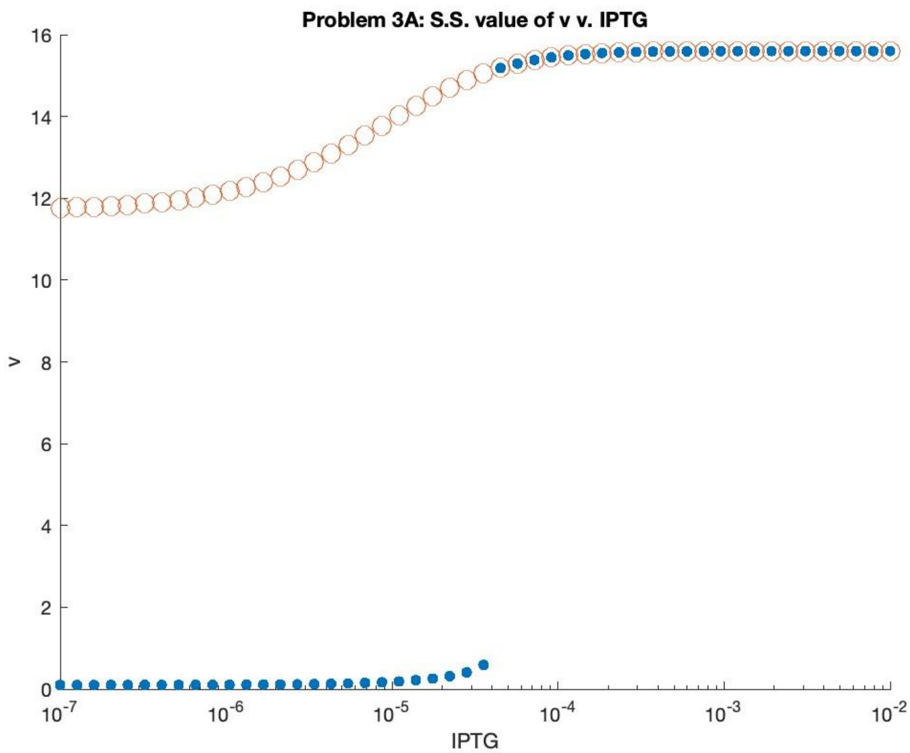

Problem 2B: v v. u nullcline with fimplicit
du/dt = 0
dv/dt = 0
u
v
10^{2}
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}

There are 3 intersections of the nullclines, which represent the 3 equilibria/steady states of the system.

c. Intersection 1 (v, u) = (0.1177, 1.9996)
   Intersection 2 (v, u) = (1.9996, 0.1177)
   Intersection 3 (v, u) = (1.0000, 1.0000)

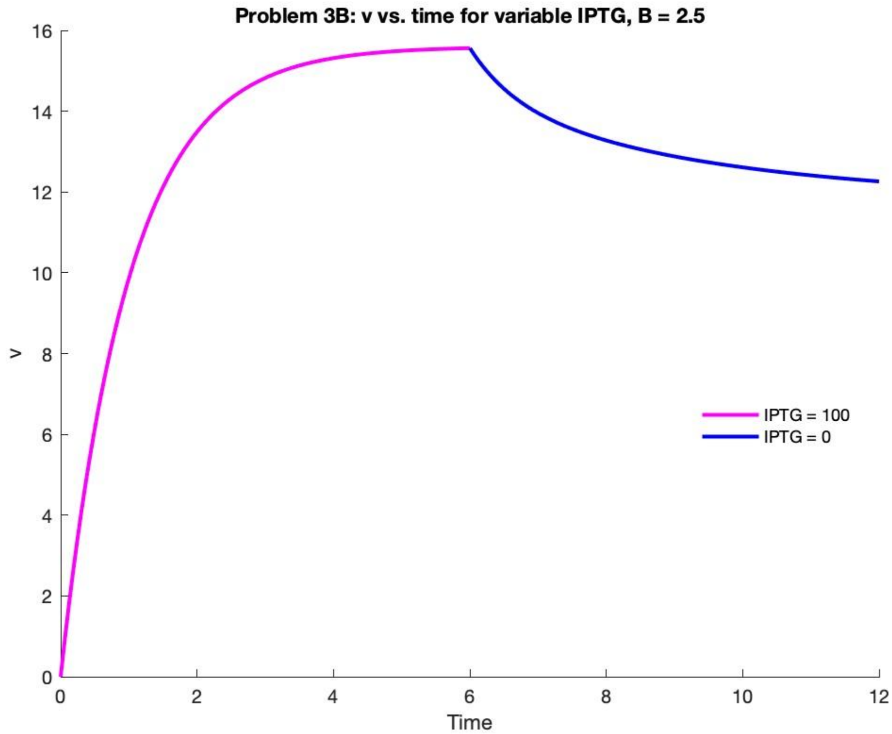
3. Analysis of IPTG inducible toggle switch system

a. Figure



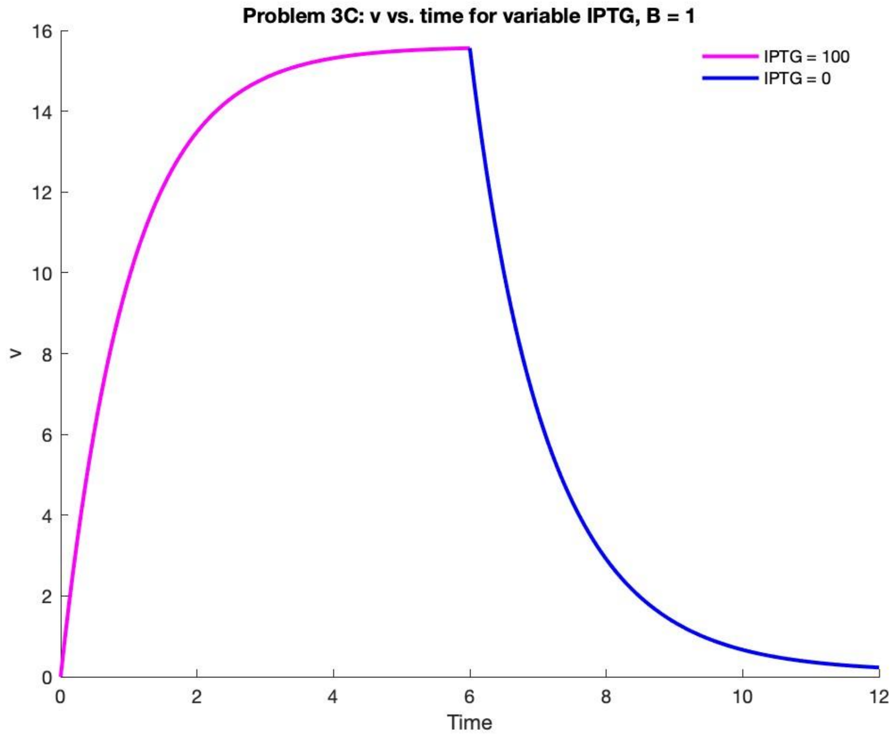
The system appears to be bistable at an IPTG concentrations less than approximately 4e-5.

b. Figure



When IPTG is removed from the system, the value of v decays nonlinearly approximately 15.5 to a new steady state of approximately 12.

c. Figure



When IPTG is removed from the system with the decreased B value of 1, the value of v decays more rapidly and nonlinearly from approximately 15.5 to a steady state of about 0.

CODE

```
close all
clc

%% Challenge Problem 1
disp('CHALLENGE PROBLEM 1')
Clear

[t, Y] = ode45(@(t, y) 0.5-0.1*y, [0, 50], 0);
figure(1)
hold on
plot(t, 0.5./0.1.*(1-exp(-0.1.*t)), '-m', 'LineWidth', 1.5) % analytical solution
plot(t, Y, '.b', 'MarkerSize', 15) % numerical solution
legend('analytical', 'numerical', 'Location', 'southeast')
xlabel('time')
ylabel('concentration')
title('Challenge Problem 1')
hold off

%% Challenge Problem 2
disp('CHALLENGE PROBLEM 2')
Clear

x = linspace(0, 4, 100);
a = [0.1; 0.2; 0.3; 0.4];
deg = a.*x;
prod = 0.01 + ((0.5.*x.^2)./(1.^2+x.^2));

figure(2)
hold on
plot(x, prod, '-b', 'LineWidth', 1.5)
plot(x, a.*x, '-m', 'LineWidth', 1.5)
legend('production', 'degradation', 'Location', 'northwest')
xlabel('x')
ylabel('dx/dt')
title('Challenge Problem 2')


%% Problem 1: Bistability of the toggle swtich network
disp('PROBLEM 1')
Clear

%% Problem 1, Part A
% setting parameters
a1 = 2;
a2 = 2;
B = 4;
g = 4;
p = [a1 a2 B g];

% setting time and concentrations
```

```
t = 0;
y = [0.1 2];

dydt = toggle_switch_rates(t, y, p);

dudt = dydt(1);
dvdt = dydt(2);

fprintf('Problem 1A:\ndu/dt = %.2f\ndv/dt = %.2f\n\n', dudt, dvdt);

%% Problem 1, Part B
% create lists of all initial conditions
u0 = [0.1 0.9 1 1.1 2.5];
v0 = [2.5 1.1 1 0.9 0.1];

% creating timespan
tspan = linspace(0, 8);

% iterating through each initial condition and simulating
u_sim = zeros(length(tspan), length(u0));
v_sim = zeros(length(tspan), length(u0));

for ii = 1:length(u0)
      y0 = [u0(ii) v0(ii)];
      y_sim = simulate_toggle_switch(tspan, y0, p);
      u_sim(:, ii) = y_sim(:, 1);
      v_sim(:, ii) = y_sim(:, 2);
end

figure(3)
hold on
plot(tspan, u_sim, 'LineWidth', 1.5)
title('Problem 1B: u v. time')
legend('u_0 = 0.1', 'u_0 = 0.9', 'u_0 = 1', 'u_0 = 1.1', 'u_0 = 2.5')
xlabel('time')
ylabel('u')
hold off

figure(4)
hold on
plot(tspan, v_sim, 'LineWidth', 1.5)
title('Problem 1B: v v. time')
legend('v_0 = 2.5', 'v_0 = 1.1', 'v_0 = 1', 'v_0 = 0.9', 'v_0 = 0.1')
xlabel('time')
ylabel('v')
hold off

figure(5)
hold on
plot(v_sim(:, [1 2 4 5]), u_sim(:, [1 2 4 5]), 'LineWidth', 1.5)
plot(v_sim(:, 3), u_sim(:, 3), '.', 'MarkerSize', 15)
```

```
title('Problem 1B: u v. v')
legend('u_0, v_0 = (0.1, 2.5)', 'u_0, v_0 = (0.9 , 1.1)', 'u_0, v_0 = (1.1, 0.9)',
'u_0, v_0 = (2.5, 0.1)','u_0, v_0 = (1, 1)')
xlabel('v')
ylabel('u')
hold off

%{
The initial conditions affect which of the three steady states the system
converges to. u < 1 approaches a steady state of u ~ 0.1, u > 1 approaches
a steady state of u ~ 2. v < 1 approaches a steady state of v ~ 0.1, v > 1
approaches a steady state of v ~ 2. When u = 1 or v = 1, the system
remains in an unsteady equilibrium. The system has three steady states,
though the steady states where (u, v) = (2, 0.1) or (u, v) = (0.1, 2) are
stable equilibria, whereas (u, v) = (1, 1) is an unsteady equilibrium since
any small perturbation away from this point causes the system to drift to
one of the other two steady equilibria.
%}

%% Problem 1, Part C
% set range of a1 values
a1_range = logspace(0, 2);

% simulating with first of a1 to obtain initial guess
a2 = 2;
B = 4;
g = 4;
p = [a1_range(1), a2, B, g];

% simulate
tspan = linspace(0, 8);
y0 = [0 0];
y = simulate_toggle_switch(tspan, y0, p);
v0 = y(end, 2);

% iterating through in forward direection
v_ss_f = zeros(1, length(a1_range));
for ii = 1:length(a1_range)
      % setting function for fitting
      fit_func = @(v) combined_toggle_switch(v, a1_range(ii));

      % using fzero to find ss value of v
      v_ss = fzero(fit_func, v0);
      v_ss_f(ii) = v_ss;
      v0 = v_ss;
end

% now iterating through in reverse direction
v_ss_r = zeros(1, length(a1_range));
for ii = 1:length(a1_range)
      % setting function for fitting
```

```
    fit_func = @(v) combined_toggle_switch(v, a1_range(length(a1_range) -
    (ii-1)));
    % using fzero to find ss value of v
    v_ss = fzero(fit_func, v0);
    v_ss_r(length(v_ss_r) - (ii-1)) = v_ss;
    v_0 = v_ss;
end

figure(6)
hold on
plot(a1_range, v_ss_f, '.', 'MarkerSize', 15)
plot(a1_range, v_ss_r, 'o', 'MarkerSize', 10)
title('Problem 1C: S.S. value of v v. a1')
xlabel('a1')
ylabel('v')
set(gca, 'Xscale', 'log')

%{
The system appears to be bistable between values of 1.4 < a1 < 6.6, where
the plot shows that there are two steady states depending on which
direction (forward or backward) v is changing. At a1 ~ 1.4 or a1 ~ 6.6, the
system switches abruptly to the other steady state as seen on the graph.
%}

%% Problem 2: Generation and analysis of nullcline plots
disp('PROBLEM 2')
clear

%% Problem 2, Part A
% setting parameter values
a1 = 2;
a2 = 2;
B = 4;
g = 4;

% setting range of u and v values
u_span = logspace(-2, 2);
v_span = logspace(-2, 2);

% solving for dudt = 0 and dvdt = 0 analytically
u = (a1./(1+v_span.^B));
v = (a2./(1+u_span.^g));
figure(7)
hold on
plot(v_span, u, '-', 'LineWidth', 1.5)
plot(v, u_span, '-', 'LineWidth', 1.5)
title('Problem 2A: v v. u nullcline analytically')
legend('du/dt = 0', 'dv/dt = 0')
xlabel('v')
ylabel('u')
xlim([1e-2, 1e2])
```

```
ylim([1e-2, 1e2])
set(gca, 'Xscale', 'log')
set(gca, 'Yscale', 'log')
hold off

%% Problem 2, Part B
figure(8)
hold on
f1 = @(v, u) (a1./(1+v.^B)) - u;
f2 = @(v, u) (a2./(1+u.^g)) - v;
fimplicit(f1, [1e-2 1e2 1e-2 1e2], 'LineWidth', 1.5)
fimplicit(f2, [1e-2 1e2 1e-2 1e2], 'LineWidth', 1.5)
title('Problem 2B: v v. u nullcline with fimplicit')
legend('du/dt = 0', 'dv/dt = 0')
xlabel('v')
ylabel('u')
set(gca, 'Xscale', 'log')
set(gca, 'Yscale', 'log')
hold off

%{
There are 3 intersections of the nullclines, which represent the 3
equilibria/steady states of the system.
%}

%% Problem 2, Part C
p = [a1 a2 B g];

% find intersections
int1 = fsolve(@(y) toggle_switch_rates(0, y, p), [0.12, 1.98]);
int2 = fsolve(@(y) toggle_switch_rates(0, y, p), [1.98, 0.12]);
int3 = fsolve(@(y) toggle_switch_rates(0, y, p), [1, 1]);

v_int1 = int1(1);
u_int1 = int1(2);

v_int2 = int2(1);
u_int2 = int2(2);

v_int3 = int3(1);
u_int3 = int3(2);

fprintf('Problem 2C:\nIntersection 1 (v, u) = (%.4f, %.4f)\nIntersection 2 (v, u) =
(%.4f, %.4f)\nIntersection 3 (v, u) = (%.4f, %.4f)\n\n', v_int1, u_int1, v_int2,
u_int2, v_int3, u_int3);

%% Problem 3: Analysis of IPTG inducible toggle switch system
disp('PROBLEM 3')
clear

%% Problem 3, Part A
```

```
a1 = 156.25;
a2 = 15.6;
B = 2.5;
g = 1;
K = 2.9618e-5;
n = 2.0015;

p = [a1 a2 B g K n];

IPTG_range = logspace(-7,-2);

% simulate
tspan = linspace(0, 8);

y0 = [0 0];

y = simulate_inducible_toggle_switch(tspan, y0, IPTG_range(1), p);
v0 = y(end, 2);

% iterating through in forward direction
v_ss_f = zeros(1, length(IPTG_range));

for ii = 1:length(IPTG_range)
      % setting function for fitting
      fit_func = @(v) combined_inducible_toggle_switch(v, IPTG_range(ii), p);

      % using fzero to find ss value of v
      v_ss = abs(fzero(fit_func, v0));
      v_ss_f(ii) = v_ss;
      v0 = v_ss;
end

% now iterating through in reverse direction
v_ss_r = zeros(1, length(IPTG_range));

for ii = 1:length(IPTG_range)
      % setting function for fitting
      fit_func = @(v) combined_inducible_toggle_switch(v,
      IPTG_range(length(IPTG_range) - (ii-1)), p);

      % using fzero to find ss value of v
      v_ss = abs(fzero(fit_func, v0));
      v_ss_r(length(v_ss_r) - (ii-1)) = v_ss;
      v_0 = v_ss;
end

figure(9)
hold on
plot(IPTG_range, v_ss_f, '.', 'MarkerSize', 15)
plot(IPTG_range, v_ss_r, 'o', 'MarkerSize', 10)
title('Problem 3A: S.S. value of v v. IPTG')
```

```matlab
xlabel('IPTG')
ylabel('v')
set(gca, 'Xscale', 'log')

%{
The system appears to be bistable at an IPTG concentrations less than
approximately 4e-5.
%}

%% Problem 3, Part B
a1 = 156.25;
a2 = 15.6;
B = 2.5;
g = 1;
K = 2.9618e-5;
n = 2.0015;

p = [a1 a2 B g K n];

IPTG1 = 100;
IPTG2 = 0;

tspan1 = linspace(0, 6);
tspan2 = linspace(6, 12);

y0 = [0 0];
y_t1 = simulate_inducible_toggle_switch(tspan1, y0, IPTG1, p);

y0 = y_t1(end, :);
y_t2 = simulate_inducible_toggle_switch(tspan2, y0, IPTG2, p);

v1 = y_t1(:, 2);
v2 = y_t2(:, 2);

figure(10)
hold on
plot(tspan1, v1, '-m', LineWidth=2, DisplayName='IPTG = 100')
plot(tspan2, v2, '-b', LineWidth=2, DisplayName='IPTG = 0')
title('Problem 3B: v vs. time for variable IPTG, B = 2.5')
xlabel('Time')
ylabel('v')
legend(location='best')
legend box off
hold off

%{
When IPTG is removed from the system, the value of v decays nonlinearly approximately
15.5 to a new steady state of approximately 12.
%}

%% Problem 3, Part C
```

```
a1 = 156.25;
a2 = 15.6;
B = 1;
g = 1;
K = 2.9618e-5;
n = 2.0015;

p = [a1 a2 B g K n];

IPTG1 = 100;
IPTG2 = 0;

tspan1 = linspace(0, 6);
tspan2 = linspace(6, 12);

y0 = [0 0];
y_t1 = simulate_inducible_toggle_switch(tspan1, y0, IPTG1, p);

y0 = y_t1(end, :);
y_t2 = simulate_inducible_toggle_switch(tspan2, y0, IPTG2, p);

v1 = y_t1(:, 2);
v2 = y_t2(:, 2);

figure(11)
hold on
plot(tspan1, v1, '-m', LineWidth=2, DisplayName='IPTG = 100')
plot(tspan2, v2, '-b', LineWidth=2, DisplayName='IPTG = 0')
title('Problem 3C: v vs. time for variable IPTG, B = 1')
xlabel('Time')
ylabel('v')
legend(location='best')
legend box off
hold off

%{
When IPTG is removed from the system with the decreased B value of 1, the value of v
decays more rapidly and nonlinearly from approximately 15.5 to a new steady state of
0.
%}

%% Functions
% Problem 1
function dydt = toggle_switch_rates(t, y, p)
% Function for calculating rates of change in toggle switch model
      % initializing parameters
      a1 = p(1);
      a2 = p(2);
      B = p(3);
      g = p(4);
```

```
        % initializing species
        u = y(1);
        v = y(2);

        % calculating rates of change
        dydt = zeros(size(y));

        % calculating dudt
        dydt(1) = a1/(1+v^B) - u;

        % calculate dvdt
        dydt(2) = a2/(1+u^g) - v;
end

function y = simulate_toggle_switch(tspan, y0, p)
% Function for simulating toggle switch rate laws with ode45
        % initializing rate laws
        dydt = @(t, y) toggle_switch_rates(t, y, p);

        % simulate with ode45
        [~, y] = ode45(dydt, tspan, y0);
end

function dvdt = combined_toggle_switch(v, a1)
% Function for calculating dvdt given v and a1
        % setting parameters
        a2 = 2;
        B = 4;
        g = 4;

        % calculating u
        u = a1/(1+v^B);

        % calculating dvdt
        dvdt = a2/(1+u^g) - v;
end

% Problem 2

% Problem 3
function dydt = inducible_toggle_switch_rates(t, y, IPTG, p)
        a1 = p(1);
        a2 = p(2);
        B = p(3);
        g = p(4);
        K = p(5);
        n = p(6);

        u = y(1);
        v = y(2);
```

```
      dydt = zeros(size(y));

      dydt(1) = (a1./(1+v.^B)) - u;
      dydt(2) = (a2./(1+(u./((1+(IPTG./K)).^n)).^g)) - v;
end

function y = simulate_inducible_toggle_switch(tspan, y0, IPTG, p)
      % initializing rate laws
      dydt = @(t, y) inducible_toggle_switch_rates(t, y, IPTG, p);

      % simulate with ode45
      [~, y] = ode45(dydt, tspan, y0);
end

function dvdt = combined_inducible_toggle_switch(v, IPTG, p)
% Function for calculating dvdt given v and a1
      a1 = p(1);
      a2 = p(2);
      B = p(3);
      g = p(4);
      K = p(5);
      n = p(6);
      v = abs(v);

      % calculating u
      u = (a1./(1+v.^B));

      % calculating dvdt
      dvdt = (a2./(1+(u./((1+(IPTG./K)).^n)).^g)) - v;
end
```