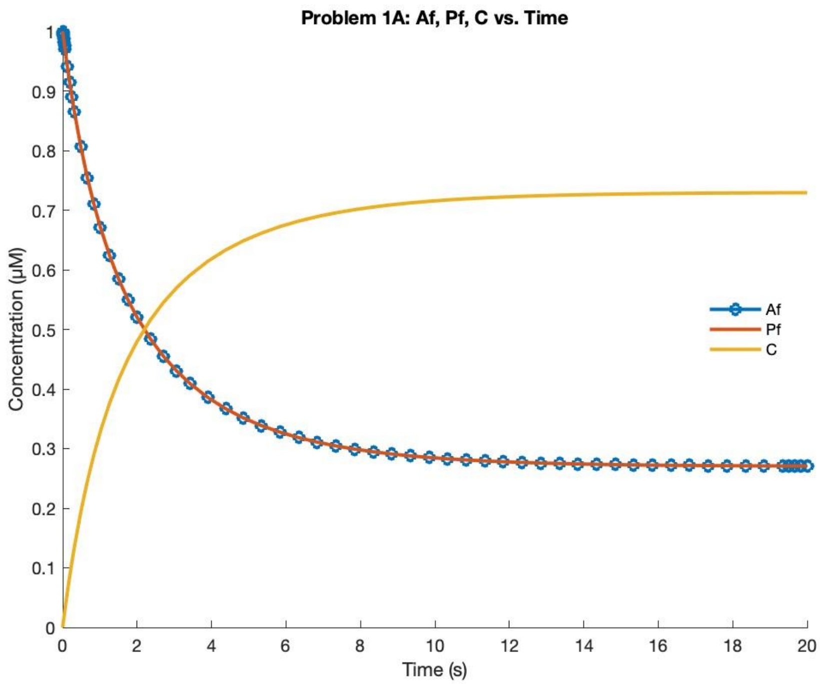
Robert Heeter
BIOE 446
22 September 2023

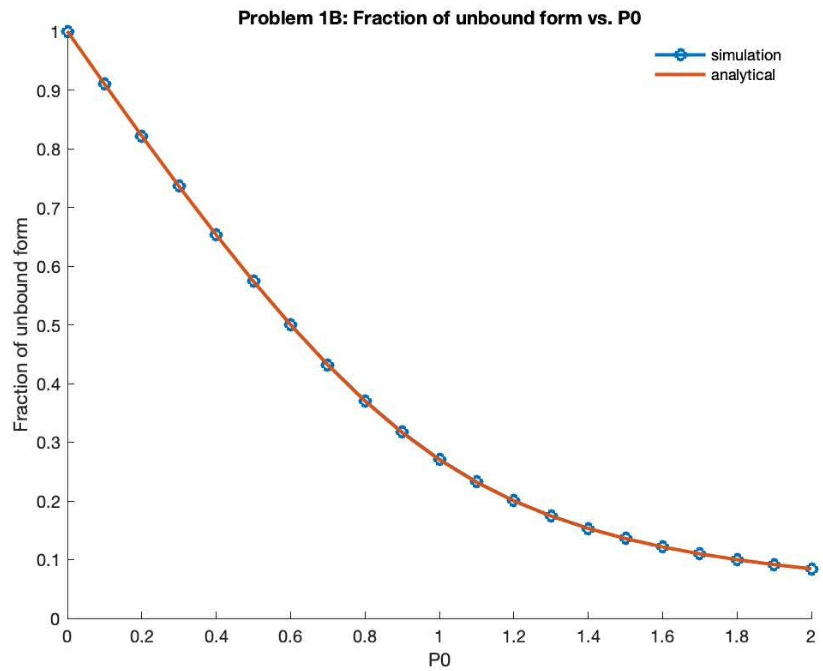
Lab 4: Simulation and Analysis of Biochemical Oscillators

ANSWERS

1. **Stoichiometric sequestration and sensitivity (DEMO)**
    a. Figure

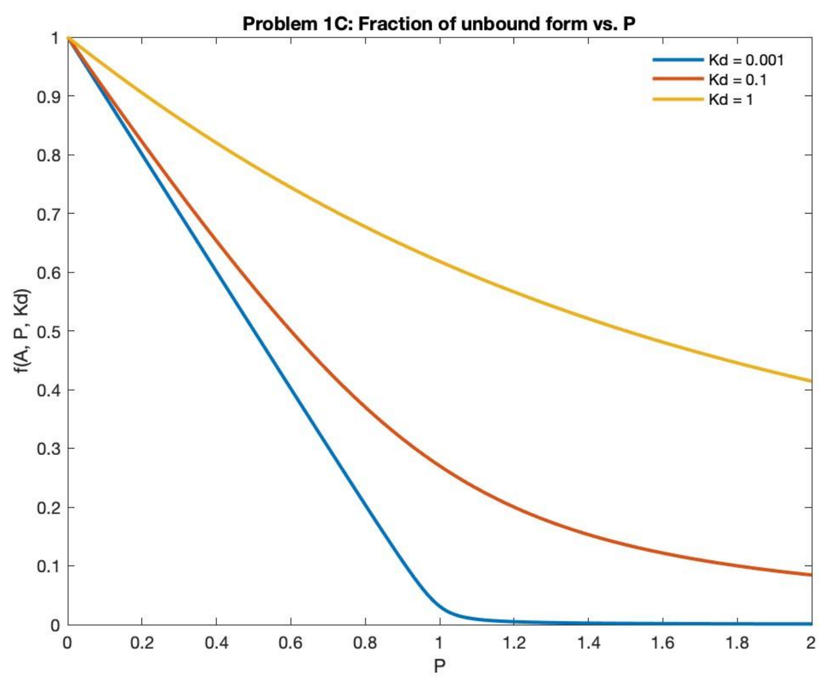


    b. Figure



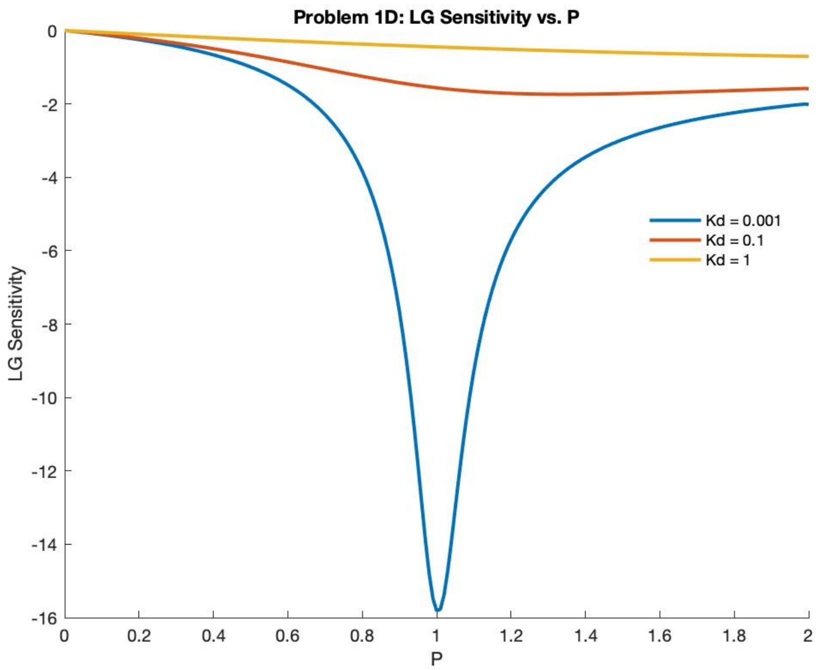
The analytical formula for the fraction of the unbound form and the simulation/numerical solution give nearly the identical result based on the figure.

c. Figure



With stronger binding (decreased Kd dissociation constant) the curve becomes steeper and approaches a linear trend as f(P, A, Kd) goes from 1 to 0 as P goes from 0 to 1. This means with strong binding, the fraction of the unbound form decreases approximately linearly with P.

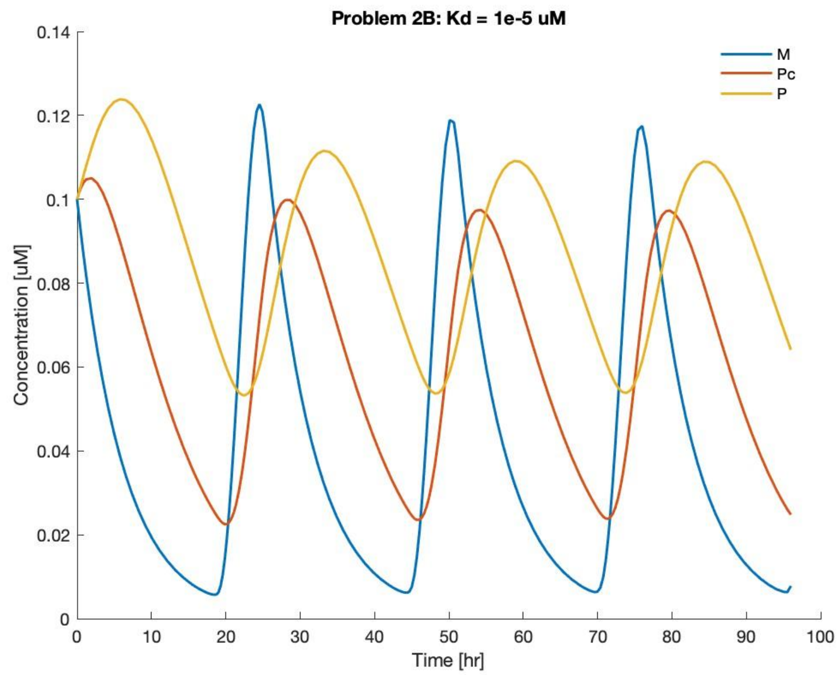
d. Figure



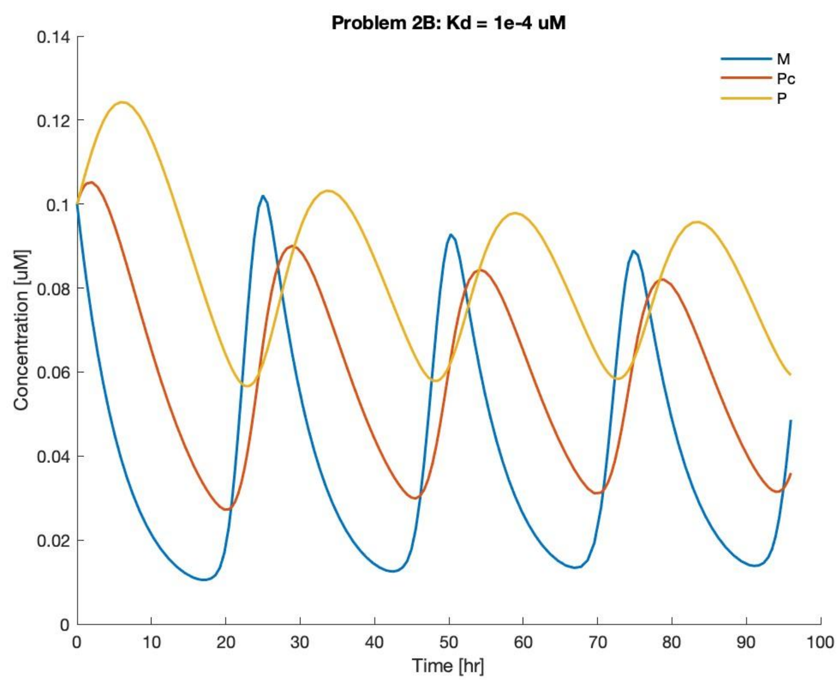
As the dissociation constant Kd decreases (stronger binding affinity), the overall LG sensitivity across the range of 0 < P < 2 increases (more negative). For an ultrasensitive response, Kd must be sufficiently small (here around 0.001) and the stoichiometric ratio of P:A must be about 1.

2. **Single negative feedback (SNF) loop**
   a. dM/dt @t=0 = -0.01643 uM/hr
      dPc/dt @t=0 = 0.00650 uM/hr
      dP/dt @t=0 = 0.00650 uM/hr

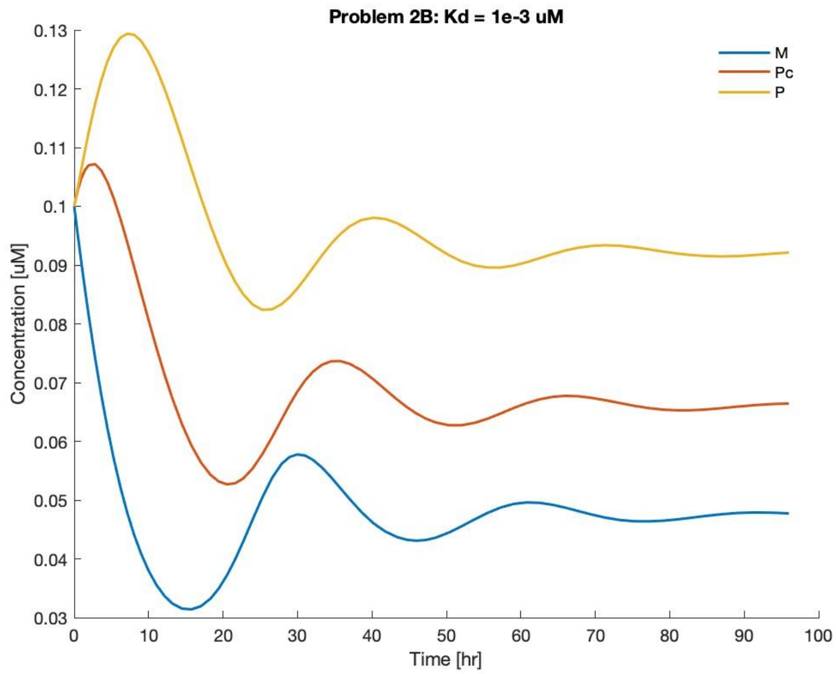
   b. Figures



Problem 2B: Kd = 1e-5 uM
M
Pc
P
Concentration [uM]
Time [hr]
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0
10
20
30
40
50
60
70
80
90
100



Problem 2B: Kd = 1e-4 uM
M
Pc
P
Concentration [uM]
Time [hr]
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0
10
20
30
40
50
60
70
80
90
100

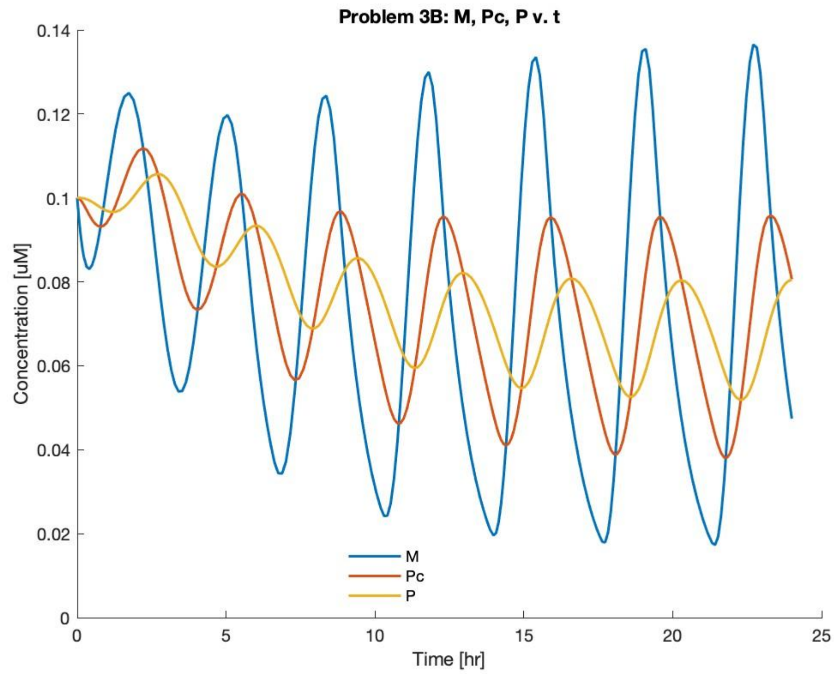


c. *As the dissociation constant Kd increases from 1e-5 to 1e-3, the binding affinity decreases which reduces the frequency and magnitude of oscillations for M. In other words, an increased Kd increases the "damping" on the system (though in the range 1e-5 to 1e-3 the system remains either undamped or underdamped).* Oscillations here are due to fluctuations in activator-repressor binding over time.

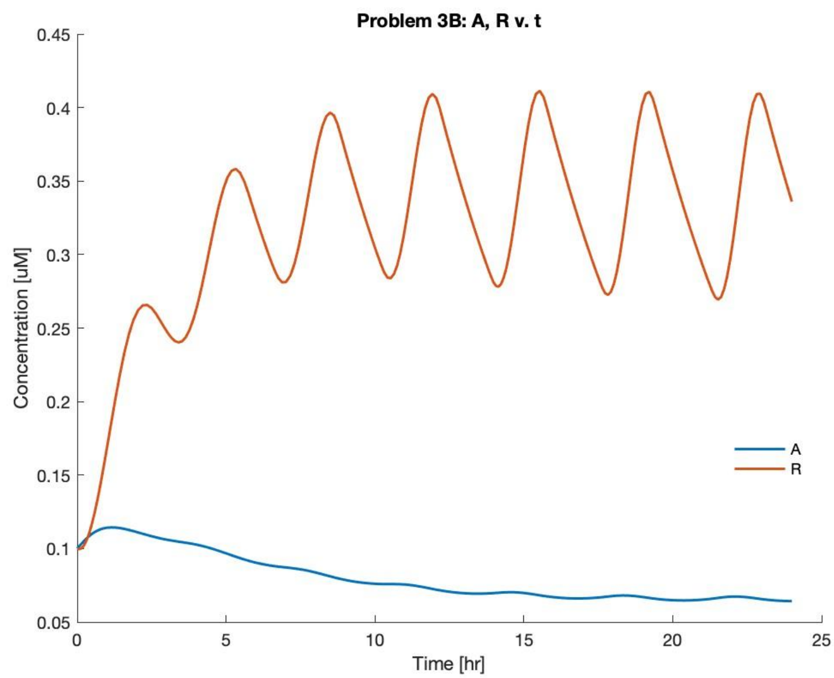
3. Adding a second feedback loop

a. dA/dt @t=0 = 0.02300 uM/hr
dR/dt @t=0 = -0.01005 uM/hr

b. Figures

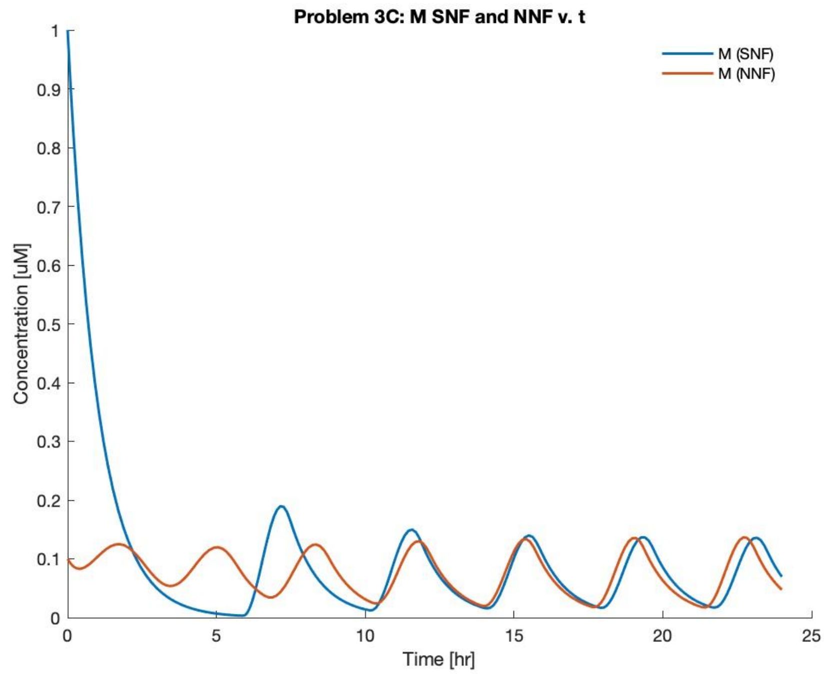


Problem 3B: M, Pc, P v. t
Concentration [uM]
Time [hr]
M
Pc
P

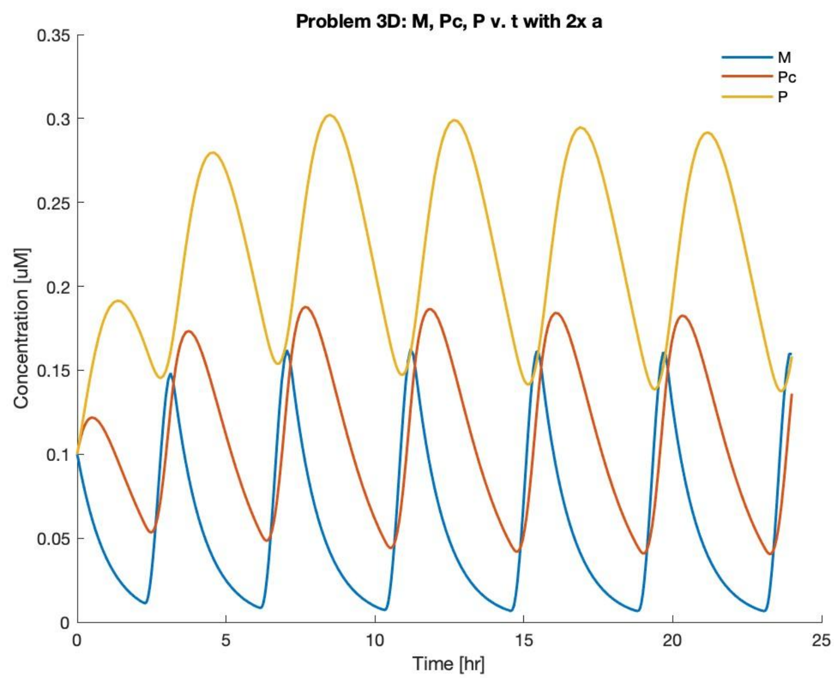


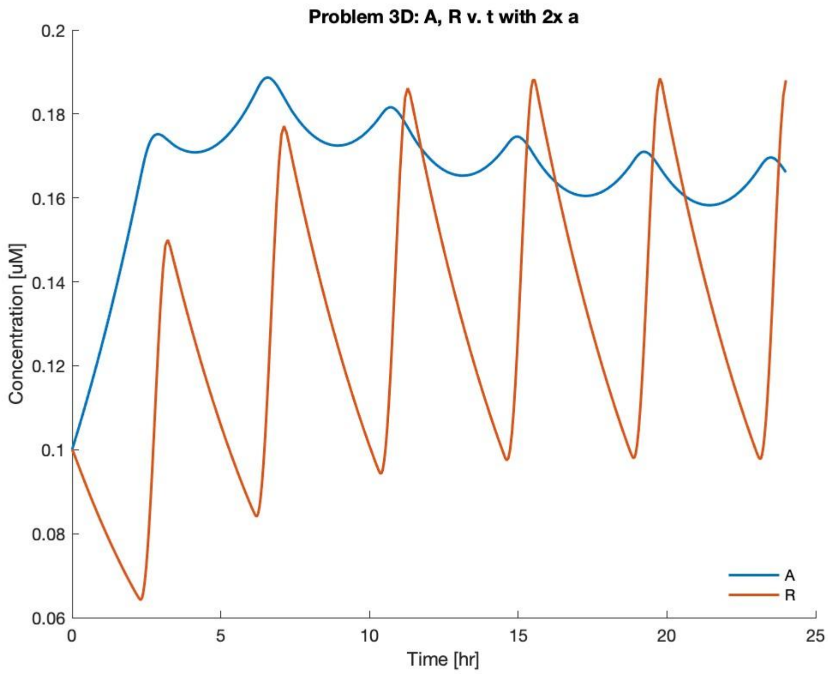
Problem 3B: A, R v. t
Concentration [uM]
Time [hr]
A
R

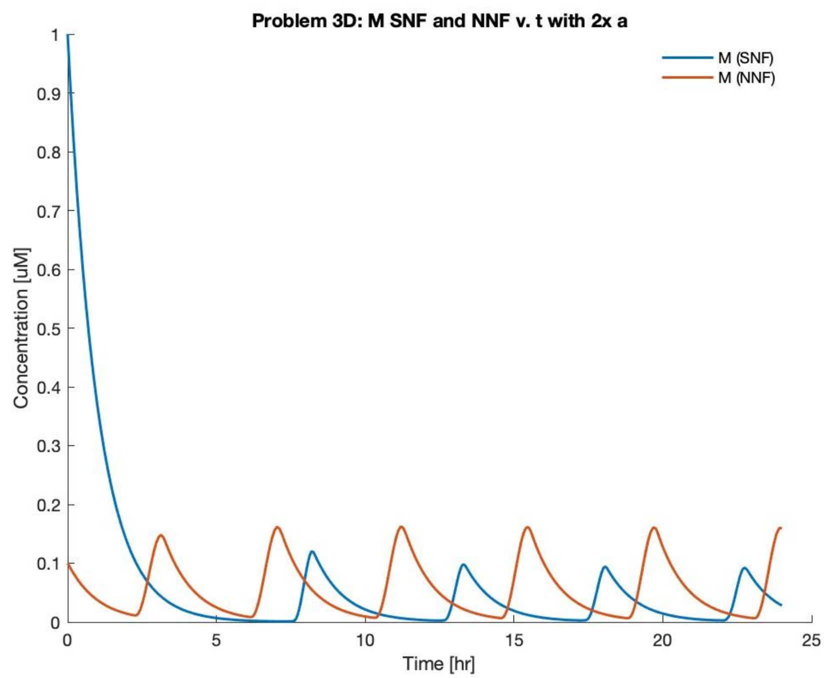
c. Figure


Problem 3C: M SNF and NNF v. t
M (SNF)
M (NNF)
Concentration [uM]
Time [hr]
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
5
10
15
20
25

d. Figures


Problem 3D: M, Pc, P v. t with 2x a
M
Pc
P
Concentration [uM]
Time [hr]
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
5
10
15
20
25

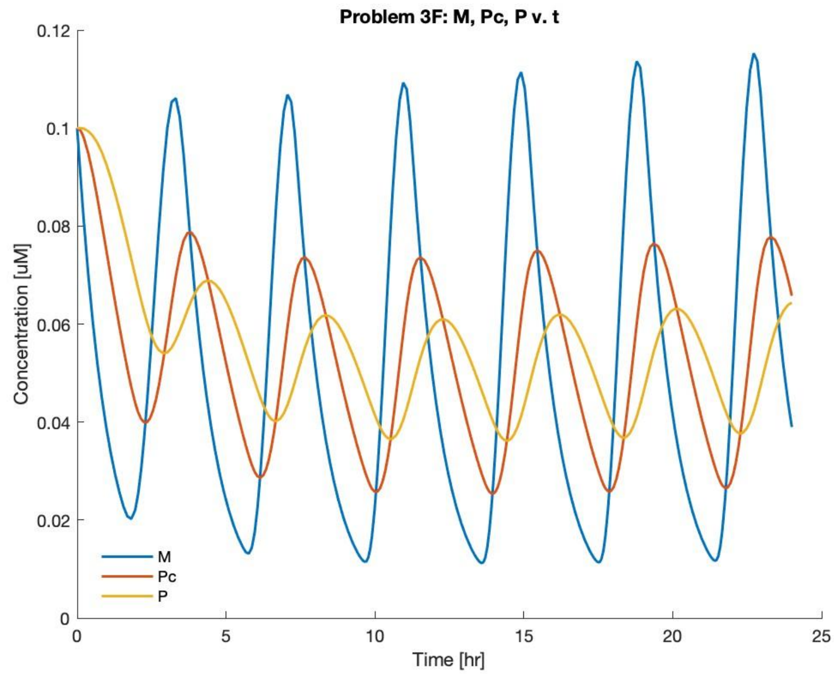


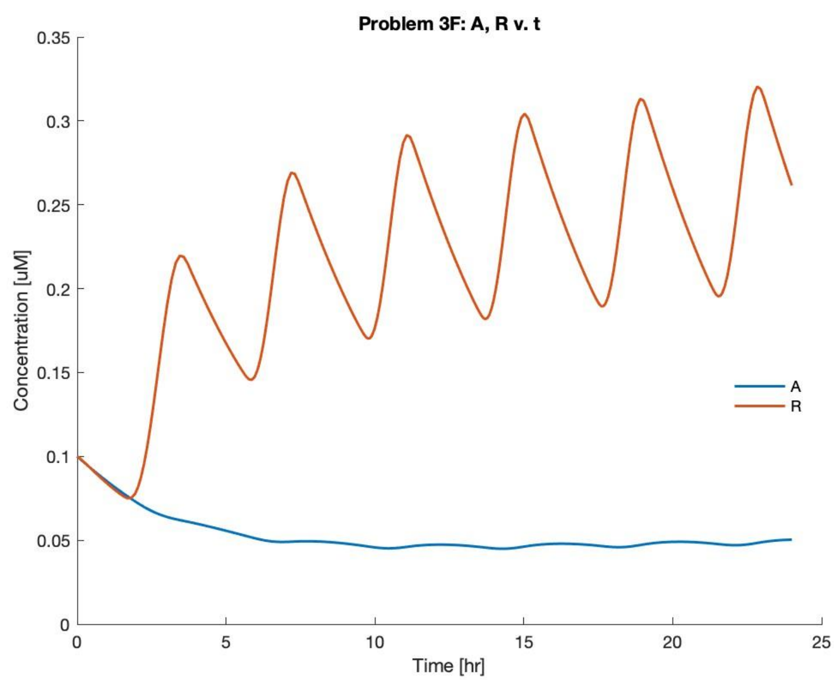


The single negative feedback (SNF) structure appears more sensitive to perturbations in alpha (a) than the negative-negative feedback (NNF) structure, comparing the figures for M concentration in 3C and 3D. When alpha is doubled, the frequency and amplitude of oscillations decreases. The frequency and amplitude of oscillations for the NNF structure are roughly the same for both alpha values.

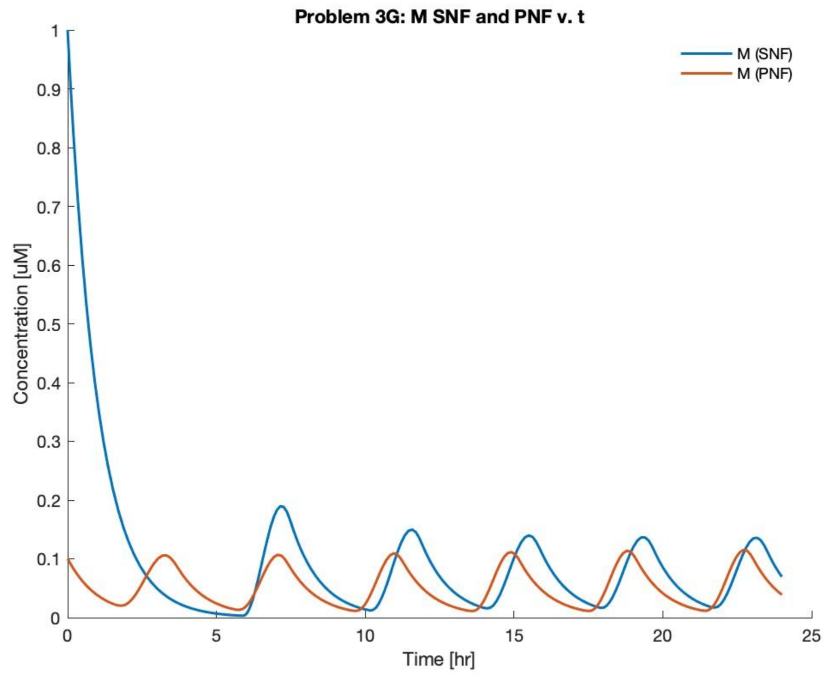
e. dA/dt @t=0 = -0.01605 uM/hr

f. Figures



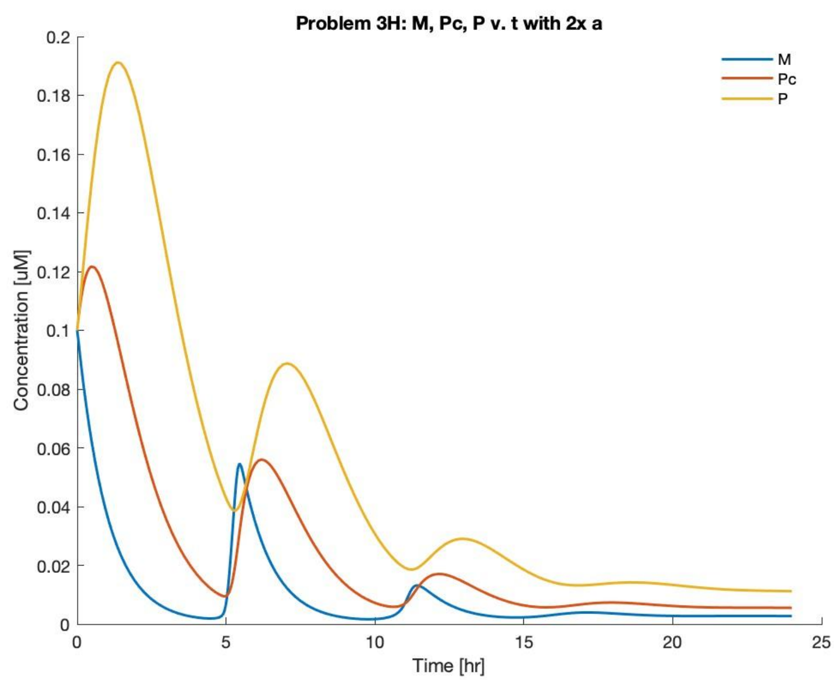


g. Figure

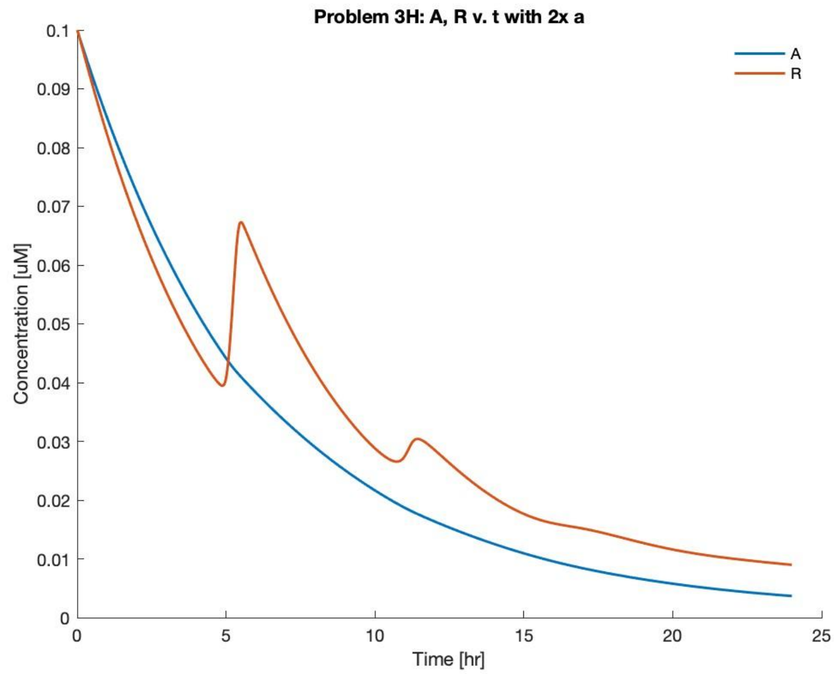


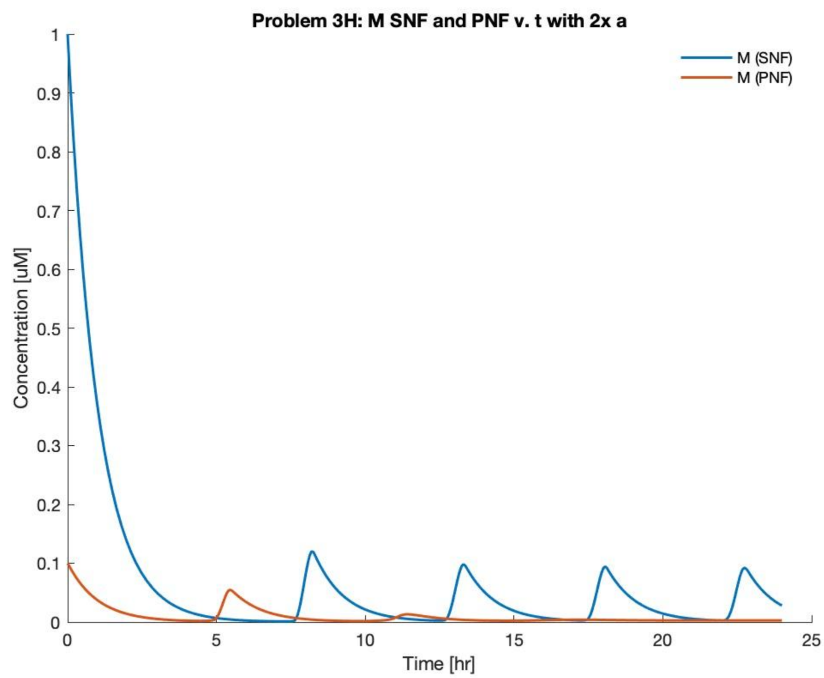
Problem 3G: M SNF and PNF v. t
M (SNF)
M (PNF)
Concentration [uM]
Time [hr]

h. Figures



Problem 3H: M, Pc, P v. t with 2x a
M
Pc
P
Concentration [uM]
Time [hr]





The positive-negative feedback (PNF) structure appears more sensitive to perturbations in alpha (a) than the single negative feedback (SNF) structure, comparing the figures for M concentration in 3G and 3H. When alpha is doubled, the frequency and amplitude of oscillations decreases much more than for the SNF structure.

CODE

```
close all
clc

%% Problem 1: Stoichiometric sequestration and sensitivity (DEMO)
disp('PROBLEM 1')
clear

%% Problem 1, Part A
kon = 0.5;
Kd = 0.1;
koff = kon*Kd;
params = [kon, koff];

% A0, P0, C0
A0 = 1;
P0 = 1;
C = 0;
state_vector = [A0, P0, C];

% simulate activator repressor dyn
[t, y] = ode45(@(t, s) activator_repressor_dyn(t, s, params), [0 20], state_vector);

% plot results
figure(1)
hold on
plot(t, y(:,1), 'o-', LineWidth=2, DisplayName='Af')
plot(t, y(:,2), LineWidth=2, DisplayName='Pf')
plot(t, y(:,3), LineWidth=2, DisplayName='C')
title('Problem 1A: Af, Pf, C vs. Time')
xlabel('Time (s)')
ylabel('Concentration (μM)')
legend(Location="best")
legend box off
hold off

%% Problem 1, Part B
P0_values = 0:0.1:2;
f_end = zeros(size(P0_values));

for i = 1:length(P0_values)
    state_vector = [A0, P0_values(i), C];
    [t, y] = ode45(@(t, s) activator_repressor_dyn(t, s, params), [0 20],
state_vector);

    Af_end = y(end, 1);
    C_end = y(end, 3);

    % calculate final fraction of free activator
    f_end(i) = Af_end / (Af_end + C_end);
```

```
end

% calculate fraction of unbound form with analytical formula
frac_unbound = (A0-P0_values-Kd+sqrt((A0-P0_values-Kd).^2+4*A0*Kd))./(2*A0);

% plot results
figure(2)
hold on
plot(P0_values, f_end, 'o-', LineWidth=2, DisplayName='simulation')
plot(P0_values, frac_unbound, LineWidth=2, DisplayName='analytical')
title('Problem 1B: Fraction of unbound form vs. P0')
xlabel('P0')
ylabel('Fraction of unbound form')
legend(location='best')
legend box off
hold off

%% Problem 1, Part C
% initialize Kd and P
Kd = [0.001, 0.1, 1];
P = 0:0.01:2;

% initialize frac_unbound
frac_unbound = zeros(length(P), length(Kd));

for i = 1:length(Kd)
    frac_unbound(:,i) = (A0-P-Kd(i)+sqrt((A0-P-Kd(i)).^2+4*A0*Kd(i)))./(2*A0);
end

% plot results
figure(3)
plot(P, frac_unbound, LineWidth=2)
title('Problem 1C: Fraction of unbound form vs. P')
xlabel('P')
ylabel('f(A, P, Kd)')
legend('Kd = 0.001', 'Kd = 0.1', 'Kd = 1', Location='best')
legend box off

%% Problem 1, Part D
% initialize results matrix
lg_sensitivity = zeros(length(P), length(Kd));

for i = 1:length(Kd)
    df = gradient(frac_unbound(:,i));
    dp = gradient(P');
    lg_sensitivity(:,i) = df./dp.*P'./frac_unbound(:,i);
end

% plot results
figure(4)
hold on
```

```
plot(P, lg_sensitivity, LineWidth=2)
title('Problem 1D: LG Sensitivity vs. P')
xlabel('P')
ylabel('LG Sensitivity')
legend('Kd = 0.001','Kd = 0.1','Kd = 1', Location='best')
legend box off


%% Problem 2: Single negative feedback (SNF) loop
disp('PROBLEM 2')
clear

%% Problem 2, Part A
a1 = 0.23;
a2 = 0.23;
a3 = 0.23;

B1 = 0.165;
B2 = 0.165;
B3 = 0.165;

A = 0.0659;
Kd = 1e-5;
params = [a1 a2 a3 B1 B2 B3 A Kd];

M0 = 0.1;
Pc = 0.1;
P = 0.1;
state_vector = [M0 Pc P];

dydt = odes2A([0, 1], state_vector, params);

dMdt_t0 = dydt(1);
dPcdt_t0 = dydt(2);
dPdt_t0 = dydt(3);

fprintf('Problem 2A:\ndM/dt @t=0 = %.5f\ndPc/dt @t=0 = %.5f\ndP/dt @t=0 = %.5f\n\n',
dMdt_t0, dPcdt_t0, dPdt_t0);

%% Problem 2, Part B
tspan = [0 96];

params(end) = 1e-5;
[t1, y1] = ode45(@(t, s) odes2A(t, s, params), tspan, state_vector);

params(end) = 1e-4;
[t2, y2] = ode45(@(t, s) odes2A(t, s, params), tspan, state_vector);

params(end) = 1e-3;
[t3, y3] = ode45(@(t, s) odes2A(t, s, params), tspan, state_vector);
```

```
figure(5)
hold on
plot(t1, y1(:,1), LineWidth=1.5, DisplayName='M')
plot(t1, y1(:,2), LineWidth=1.5, DisplayName='Pc')
plot(t1, y1(:,3), LineWidth=1.5, DisplayName='P')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 2B: Kd = 1e-5 uM")
legend(Location='best')
legend box off
hold off

figure(6)
hold on
plot(t2, y2(:,1), LineWidth=1.5, DisplayName='M')
plot(t2, y2(:,2), LineWidth=1.5, DisplayName='Pc')
plot(t2, y2(:,3), LineWidth=1.5, DisplayName='P')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 2B: Kd = 1e-4 uM")
legend(Location='best')
legend box off
hold off

figure(7)
hold on
plot(t3, y3(:,1), LineWidth=1.5, DisplayName='M')
plot(t3, y3(:,2), LineWidth=1.5, DisplayName='Pc')
plot(t3, y3(:,3), LineWidth=1.5, DisplayName='P')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 2B: Kd = 1e-3 uM")
legend(Location='best')
legend box off
hold off

%% Problem 2, Part C
% No code required.


%% Problem 3: Adding a second feedback loop
disp('PROBLEM 3')
clear

%% Problem 3, Part A
a1 = 1;
a2 = 1;
a3 = 1;

B1 = 1;
B2 = 1;
```

```
B3 = 1;

g1 = 1;
g2 = 0.0043;

d1 = 0.2;
d2 = 0.2;

Kd = 1e-5;

params = [a1 a2 a3 B1 B2 B3 g1 g2 d1 d2 Kd];

M0 = 0.1;
Pc = 0.1;
P = 0.1;
R = 0.1;
A = 0.1;

state_vector = [M0 Pc P A R];

dydt = odes3A([0, 1], state_vector, params);

dMdt_t0 = dydt(1);
dPcdt_t0 = dydt(2);
dPdt_t0 = dydt(3);
dAdt_t0 = dydt(4);
dRdt_t0 = dydt(5);

fprintf('Problem 3A:\ndA/dt @t=0 = %.5f\ndR/dt @t=0 = %.5f\n\n', dAdt_t0, dRdt_t0);

%% Problem 3, Part B
tspan = [0 24];
[t_NNF, y_NNF] = ode45 (@(t, s) odes3A(t, s, params), tspan, state_vector);

figure(8)
hold on
plot(t_NNF, y_NNF(:,1), LineWidth=1.5, DisplayName='M')
plot(t_NNF, y_NNF(:,2), LineWidth=1.5, DisplayName='Pc')
plot(t_NNF, y_NNF(:,3), LineWidth=1.5, DisplayName='P')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3B: M, Pc, P v. t")
legend(Location='best')
legend box off
hold off

figure(9)
hold on
plot(t_NNF, y_NNF(:,4), LineWidth=1.5, DisplayName='A')
plot(t_NNF, y_NNF(:,5), LineWidth=1.5, DisplayName='R')
xlabel("Time [hr]")
```

```
ylabel("Concentration [uM]")
title("Problem 3B: A, R v. t")
legend(Location='best')
legend box off
hold off

%% Problem 3, Part C
a1 = 1;
a2 = 1;
a3 = 1;

B1 = 1;
B2 = 1;
B3 = 1;

A = 0.0659;

Kd = 1e-5;

params = [a1 a2 a3 B1 B2 B3 A Kd];

M0 = 1;
Pc = 1;
P = 1;

state_vector = [M0 Pc P];

tspan = [0 24];
[t_SNF, y_SNF] = ode45 (@(t, s) odes2A(t, s, params), tspan, state_vector);

figure(10)
hold on
plot(t_SNF, y_SNF(:,1), LineWidth=1.5, DisplayName='M (SNF)')
plot(t_NNF, y_NNF(:,1), LineWidth=1.5, DisplayName='M (NNF)')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3C: M SNF and NNF v. t")
legend(Location='best')
legend box off
hold off

%% Problem 3, Part D
% Repeat Part B
a1 = 2;
a2 = 2;
a3 = 2;

B1 = 1;
B2 = 1;
B3 = 1;
```

```
g1 = 1;
g2 = 0.0043;

d1 = 0.2;
d2 = 0.2;

Kd = 1e-5;

params = [a1 a2 a3 B1 B2 B3 g1 g2 d1 d2 Kd];

M0 = 0.1;
Pc = 0.1;
P = 0.1;
R = 0.1;
A = 0.1;

state_vector = [M0 Pc P A R];

tspan = [0 24];
[t_NNF2, y_NNF2] = ode45 (@(t, s) odes3A(t, s, params), tspan, state_vector);

figure(11)
hold on
plot(t_NNF2, y_NNF2(:,1), LineWidth=1.5, DisplayName='M')
plot(t_NNF2, y_NNF2(:,2), LineWidth=1.5, DisplayName='Pc')
plot(t_NNF2, y_NNF2(:,3), LineWidth=1.5, DisplayName='P')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3D: M, Pc, P v. t with 2x a")
legend(Location='best')
legend box off
hold off

figure(12)
hold on
plot(t_NNF2, y_NNF2(:,4), LineWidth=1.5, DisplayName='A')
plot(t_NNF2, y_NNF2(:,5), LineWidth=1.5, DisplayName='R')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3D: A, R v. t with 2x a")
legend(Location='best')
legend box off
hold off

% Repeat Part C
a1 = 2;
a2 = 2;
a3 = 2;

B1 = 1;
B2 = 1;
```

```
B3 = 1;

A = 0.0659;

Kd = 1e-5;

params = [a1 a2 a3 B1 B2 B3 A Kd];

M0 = 1;
Pc = 1;
P = 1;

state_vector = [M0 Pc P];

tspan = [0 24];
[t_SNF2, y_SNF2] = ode45 (@(t, s) odes2A(t, s, params), tspan, state_vector);

figure(13)
hold on
plot(t_SNF2, y_SNF2(:,1), LineWidth=1.5, DisplayName='M (SNF)')
plot(t_NNF2, y_NNF2(:,1), LineWidth=1.5, DisplayName='M (NNF)')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3D: M SNF and NNF v. t with 2x a")
legend(Location='best')
legend box off
hold off

%% Problem 3, Part E
a1 = 1;
a2 = 1;
a3 = 1;

B1 = 1;
B2 = 1;
B3 = 1;

g1 = 1;
g2 = 0.0395;

d1 = 0.2;
d2 = 0.2;

Kd = 1e-5;

params = [a1 a2 a3 B1 B2 B3 g1 g2 d1 d2 Kd];

M0 = 0.1;
Pc = 0.1;
P = 0.1;
R = 0.1;
```

```matlab
A = 0.1;

state_vector = [M0 Pc P A R];

dydt = odes3E([0, 1], state_vector, params);

dMdt_t0 = dydt(1);
dPcdt_t0 = dydt(2);
dPdt_t0 = dydt(3);
dAdt_t0 = dydt(4);
dRdt_t0 = dydt(5);

fprintf('Problem 3E:\ndA/dt @t=0 = %.5f\n\n', dAdt_t0);

%% Problem 3, Part F
tspan = [0 24];
[t_PNF, y_PNF] = ode45 (@(t, s) odes3E(t, s, params), tspan, state_vector);

figure(14)
hold on
plot(t_PNF, y_PNF(:,1), LineWidth=1.5, DisplayName='M')
plot(t_PNF, y_PNF(:,2), LineWidth=1.5, DisplayName='Pc')
plot(t_PNF, y_PNF(:,3), LineWidth=1.5, DisplayName='P')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3F: M, Pc, P v. t")
legend(Location='best')
legend box off
hold off

figure(15)
hold on
plot(t_PNF, y_PNF(:,4), LineWidth=1.5, DisplayName='A')
plot(t_PNF, y_PNF(:,5), LineWidth=1.5, DisplayName='R')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3F: A, R v. t")
legend(Location='best')
legend box off
hold off

%% Problem 3, Part G
figure(16)
hold on
plot(t_SNF, y_SNF(:,1), LineWidth=1.5, DisplayName='M (SNF)')
plot(t_PNF, y_PNF(:,1), LineWidth=1.5, DisplayName='M (PNF)')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3G: M SNF and PNF v. t")
legend(Location='best')
legend box off
```

```
hold off

%% Problem 3, Part H
% Repeat Part F
a1 = 2;
a2 = 2;
a3 = 2;

B1 = 1;
B2 = 1;
B3 = 1;

g1 = 1;
g2 = 0.0395;

d1 = 0.2;
d2 = 0.2;

Kd = 1e-5;

params = [a1 a2 a3 B1 B2 B3 g1 g2 d1 d2 Kd];

M0 = 0.1;
Pc = 0.1;
P = 0.1;
R = 0.1;
A = 0.1;

state_vector = [M0 Pc P A R];

tspan = [0 24];
[t_PNF2, y_PNF2] = ode45 (@(t, s) odes3E(t, s, params), tspan, state_vector);

figure(17)
hold on
plot(t_PNF2, y_PNF2(:,1), LineWidth=1.5, DisplayName='M')
plot(t_PNF2, y_PNF2(:,2), LineWidth=1.5, DisplayName='Pc')
plot(t_PNF2, y_PNF2(:,3), LineWidth=1.5, DisplayName='P')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3H: M, Pc, P v. t with 2x a")
legend(Location='best')
legend box off
hold off

figure(18)
hold on
plot(t_PNF2, y_PNF2(:,4), LineWidth=1.5, DisplayName='A')
plot(t_PNF2, y_PNF2(:,5), LineWidth=1.5, DisplayName='R')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
```

```
title("Problem 3H: A, R v. t with 2x a")
legend(Location='best')
legend box off
hold off

% Repeat Part G
figure(19)
hold on
plot(t_SNF2, y_SNF2(:,1), LineWidth=1.5, DisplayName='M (SNF)')
plot(t_PNF2, y_PNF2(:,1), LineWidth=1.5, DisplayName='M (PNF)')
xlabel("Time [hr]")
ylabel("Concentration [uM]")
title("Problem 3H: M SNF and PNF v. t with 2x a")
legend(Location='best')
legend box off
hold off


%% Functions
% Problem 1
function output = activator_repressor_dyn(t, s, p)
    % Calculate dA/dt, dP/dt, dC/dt
    % Arguments
    % t (time)
    % s (initial conditions - state)
    % p (vector of parameters)

    % Return
    % column dA/dt, dP/dt, dC/dt

    % initializing parameters
    kon = p(1);
    koff = p(2);

    % current conditions (state vector)
    Af = s(1);
    Pf = s(2);
    C = s(3);

    % ODEs
    Af_dot = -1*kon*Af*Pf + koff*C;
    Pf_dot = -1*kon*Af*Pf + koff*C;
    C_dot = kon*Af*Pf - koff*C;

    output = [Af_dot; Pf_dot; C_dot];
end

% Problem 2
function dydt = odes2A(t, s, p)
    dydt = zeros(size(s));
```

```matlab
    a1 = p(1);
    a2 = p(2);
    a3 = p(3);

    B1 = p(4);
    B2 = p(5);
    B3 = p(6);

    A = p(7);
    Kd = p(8);

    M = s(1);
    Pc = s(2);
    P = s(3);

    dydt(1) = (a1.*f(P, A, Kd)) - (B1.*M);
    dydt(2) = (a2.*M) - (B2.*Pc);
    dydt(3) = (a3.*Pc) - (B3.*P);
end

function out = f(P, A, Kd)
    out = (A - P - Kd + sqrt((A - P - Kd).^2 + 4.*A.*Kd))./(2.*A);
end

% Problem 3
function dydt = odes3A(t, s, p)
    dydt = zeros(size(s));

    a1 = p(1);
    a2 = p(2);
    a3 = p(3);

    B1 = p(4);
    B2 = p(5);
    B3 = p(6);

    g1 = p(7);
    g2 = p(8);

    d1 = p(9);
    d2 = p(10);

    Kd = p(11);

    M = s(1);
    Pc = s(2);
    P = s(3);
    A = s(4);
    R = s(5);

    dydt(1) = (a1.*f(P, A, Kd)) - (B1.*M);
```

```
    dydt(2) = (a2.*M) - (B2.*Pc);
    dydt(3) = (a3.*Pc) - (B3.*P);
    dydt(4) = (g2./R) - (d2.*A);
    dydt(5) = (g1.*f(P, A, Kd)) - (d1.*R);
end

function dydt = odes3E(t, s, p)
    dydt = zeros(size(s));

    a1 = p(1);
    a2 = p(2);
    a3 = p(3);

    B1 = p(4);
    B2 = p(5);
    B3 = p(6);

    g1 = p(7);
    g2 = p(8);

    d1 = p(9);
    d2 = p(10);

    Kd = p(11);

    M = s(1);
    Pc = s(2);
    P = s(3);
    A = s(4);
    R = s(5);

    dydt(1) = (a1.*f(P, A, Kd)) - (B1.*M);
    dydt(2) = (a2.*M) - (B2.*Pc);
    dydt(3) = (a3.*Pc) - (B3.*P);
    dydt(4) = (g2.*R) - (d2.*A);
    dydt(5) = (g1.*f(P, A, Kd)) - (d1.*R);
end
```