Process Control - Project Task 2
Control of a Multivariable Process

Submitted by
Md. Yasin Arafat

Newell and Lee Evaporator

Steady State Result

Method 1: After mathematical modelling of the Newell and Lee Evaporator we can use ODE solvers in Matlab. This method solves (through integration) the ODE system in time domain. At first, initial condition should be chosen arbitrarily. Time span should be long enough so that the system states reach the steady state at the final time.

Evaporator Equation's Matlab Code

Matlab code for mathematical modelling including constraints:

```
function dxdt = evapmod(t, x)
% here evapmod=evaporator modelling
%'x' is matrix of the values [dX2dt dP2dt dL2dt]

global F2  P100  F200  F1  XF1  T1  F3  T200
X2 = x(1);
P2 = x(2);
L2 = x(3);

% Parameters and their chosen values by Newell and Lee
C = 4; Cp = 0.07;
lam = 38.5; lams = 36.6;  rhoA = 20;
M = 20; UA2 = 6.84;

% Algebraic equations
% Evaporator and Heater steam jacket
T2 = 0.5616*P2 + 0.3126*X2 + 48.43;
```

```
T3 = 0.507*P2 + 55;
T100 = 0.1538*P100 + 90;
Q100 = 0.16*(F1 + F3)*(T100 -T2);
F100 = Q100/lams;
F4 = (Q100 - F1*Cp*(T2 - T1))/lam;
%Condenser
Q200 = UA2*(T3-T200)/(1+UA2/(2*Cp*F200));
T201 = T200 + Q200/(F200*Cp);
F5 = Q200/lam;
%Differential equations
dX2dt = (F1*XF1 - F2*X2)/M;
dP2dt = (F4 - F5)/C;
dL2dt = (F1 - F4 - F2)/rhoA;
%Output: three states
dxdt = [dX2dt dP2dt dL2dt]';
end
```

Matlab Code for Steady State Solution

Here the declaration of the global variables have been defined. And the initial conditions as per the method 1 of the given instruction have been assumed. We have included the plotting code for steady state solution and final steady state solution for all three state have been saved.

```
close all
clear all
clc

global F2 P100 F200 F1 XF1 T1 F3 T200
% Inputs
F2 = 2;             % [kg/min]
P100 = 194.7;       % [kPa]
F200 = 208;         % [kg/min]
F1 = 10;            % [kg/min];
```

```
T1 = 40;              % [°C]
XF1 = 5;              % [%] here, XF1 = X1
                      %steady state of L2 will not be found in Fig 2 if XF1=10
F3 = 50;              % [kg/min]
T200 = 25;            % [°C]

%Initial dynamics of the solution
X0 = [3 3 8.8];       % Arbitrary Initial Condition
tspan=[ 0  70];       % Simulation Time
[t,x]=ode45(@evapmod,tspan, X0);   % calling the integrator (ODE solver)
figure(1);
plot(t,x), grid on
legend('X_2','P_2','L_2')
xlabel('Time (second)')
ylabel('State Variables')
title('Initial dynamics of the solution ')
%Steady state solution at the final time
X0 = [ 3 3 8.8];      % Arbitrary Initial Condition
tspan=[ 0  1000];     % Simulation Time
% tolerance settings  (= higher accuracy) for the Matlab solver to increase
% the accuracy of the numerical solution.
options = odeset('RelTol',1e-6,'AbsTol',[1.0e-6 1.e-06 1.e-06]);
[t,x]=ode45(@evapmod,tspan, X0, options);
X0ss= x(end,:) ;
X0ss(3) = 1;
save init_ss X0ss;  %final steady state solution for all three is saved in X0ss

figure(2);
plot(t,x), grid on
xlabel('Time(second)')
ylabel('State variable')
```

```
legend('X_2','P_2','L_2')
title('Steady state solution at the final time')
```

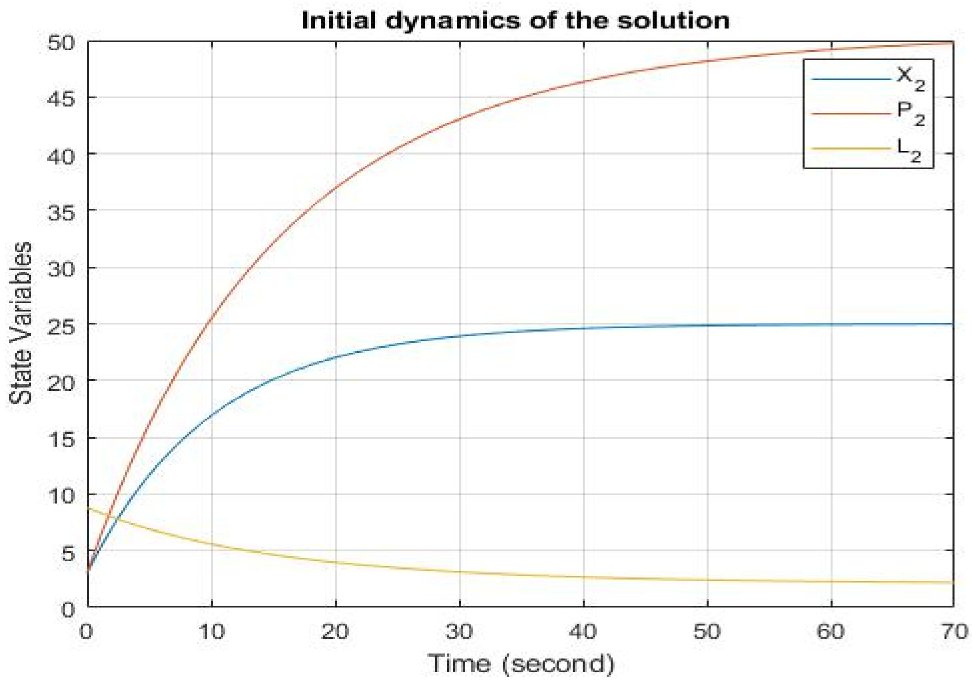

Fig 1: Initial dynamics of the outputs variables

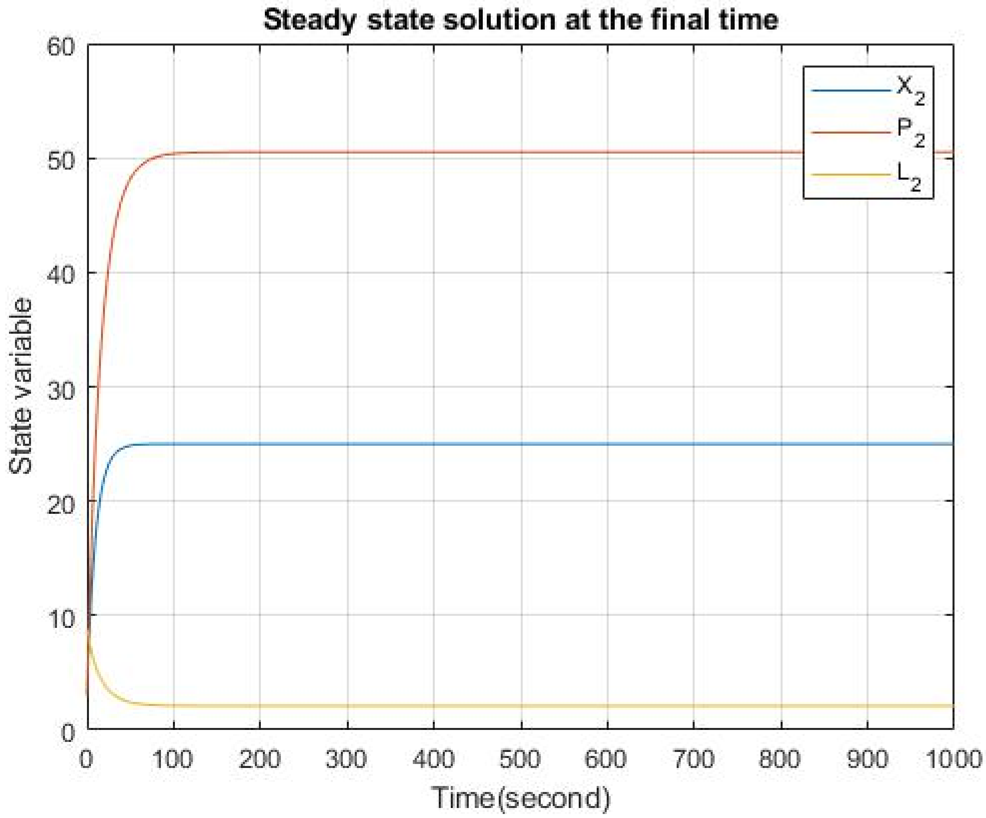

Fig 2: Steady state solution of output variables

Dynamic simulation scenarios

Given step changes in the disturbances are:

a) A step change of +10% in the feed flow rate, F1

b) A step change of -10% in the feed flow rate, F1

c) A step change of +15% in the feed composition, XF1 (we took, X1=XF1)

d) A step change of +10% in the feed composition, XF1

e) A step change of +10% in the cooling water inlet temperature, T200

f) A step change of +10% in the cooling water inlet temperature, T200

Matlab code for the above step-changes

```
clear all
clc
global F2  P100  F200  F1  XF1  T1  F3  T200
```

```
% Inputs
F2 = 2;              % [kg/min]
P100 = 194.7;        % [kPa]
F200 = 208;          % [kg/min]
F1 = 10;             % [kg/min];
T1 = 40;             % [°C]
XF1 = 5;             % [%]
F3 = 50;             % [kg/min]
T200 = 25;           % [°C]
load init_ss         % Initial 'X0ss' will be loaded  with init_ss.mat
% Disturbance scenario : 10 % step increase in F1
F1 = 10;
XF1 = 5;
F1 = F1*1.10 ;
tend = 150;
tspan=[ 0  tend];
options = odeset('RelTol',1e-6,'AbsTol',[1.0e-6 1.e-06 1.e-06]);
[t1,x1]=ode45(@evapmod,tspan, X0ss, options);
% Disturbance scenario : 10% step decrease in F1
F1 = 10;
XF1 = 5;
F1 = F1*0.90 ;
[t2,x2]=ode45(@evapmod,tspan, X0ss, options);
% Disturbance scenario : 15 % step increase in XF1
F1 = 10;
XF1 = 5;
XF1 = XF1*1.15 ;
[t3,x3]=ode45(@evapmod,tspan, X0ss, options);
% Disturbance scenario : 15 % step decrease in XF1
F1 = 10;
XF1 = 5;
```

```
XF1 = XF1*0.85 ;
[t4,x4]=ode45(@evapmod,tspan, X0ss, options);
%Disturbance scenario: 20% step increase in T200
F1 = 10;
XF1 = 5;
T200 = 25;
T200 = T200*1.20;
[t5,x5]= ode45(@evapmod,tspan, X0ss, options);
%Disturbance scenario: 20% step decrease in T200
F1 = 10;
XF1=5;
T200=25;
T200=T200*0.80;
[t6,x6]=ode45(@evapmod,tspan, X0ss, options);
% Plot step responses to F1 disurbance scenario
figure(3);
subplot(3,1,1)
grid on
plot(t1,x1(:,1),t2,x2(:,1),'--');
grid
legend('+10% F1' ,' -10% F1');
xlabel('t (second)')
ylabel('X2 ')
title(' Responses of outputs to the disurbance in F1  ')
subplot(3,1,2)
grid on
plot(t1,x1(:,2),t2,x2(:,2),'--');
grid
xlabel('t (second)')
ylabel('P2 ')
subplot(3,1,3)
```

```
grid on
plot(t1,x1(:,3),t2,x2(:,3),'--');
grid
xlabel('t (second)')
ylabel('L2 ')
% Plots step states to XF1 disturbance scenario
figure(4);
subplot(3,1,1)
grid on
plot(t3,x3(:,1),t4,x4(:,1),'--');
grid
legend('+15% XF1',' -15% XF1');
xlabel('t (second)')
ylabel('X2 ')
title(' Responses of outputs to the disturbance in XF1 ')
subplot(3,1,2)
grid on
plot(t3,x3(:,2),t4,x4(:,2),'--');
grid
xlabel('t (second)')
ylabel('P2 ')
subplot(3,1,3)
grid on
plot(t3,x3(:,3),t4,x4(:,3),'--');
grid
xlabel('t (second)')
ylabel('L2 ')
% Plots step states to T200 disturbance scenario
figure(5);
subplot(3,1,1)
grid on
```

```
plot(t5,x5(:,1),t6,x6(:,1),'--');
grid
legend('+20% T200' ,' -20% T200');
xlabel('t (second)')
ylabel('X2 ')
title(' Response of states to disurbance in T200  ')
subplot(3,1,2)
grid on
plot(t5,x5(:,2),t6,x6(:,2),'--');
grid
xlabel('t (second)')
ylabel('P2 ')
subplot(3,1,3)
grid on
plot(t5,x5(:,3),t6,x6(:,3),'--');
grid
xlabel('t (second)')
ylabel('L2 ')
```

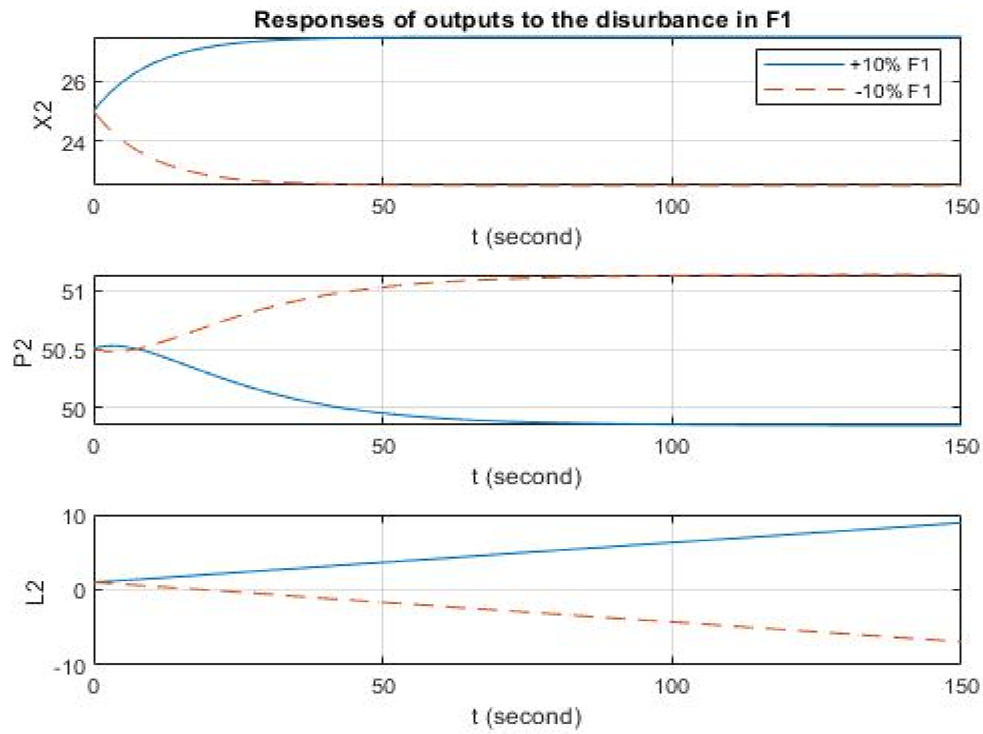

Fig 3: Outputs responses due to disturbance in F1

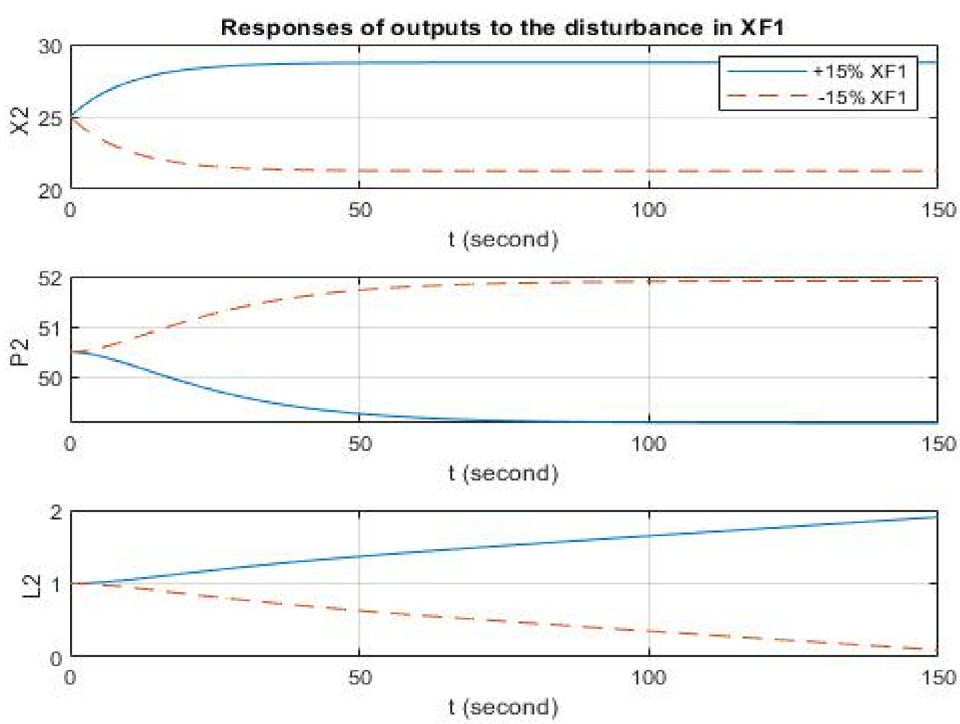

Fig 4: Output responses due to disturbance in XF1

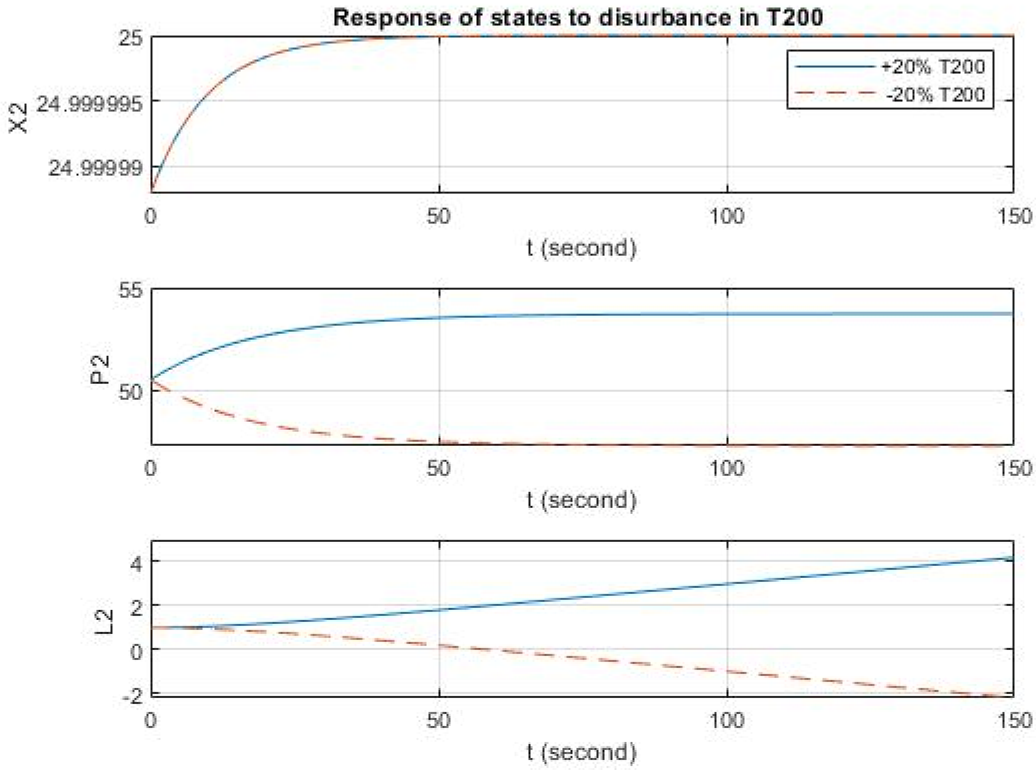

Fig 5: Output responses due to disturbance in T200

References

- Exercise of Two Tank exercise provided by Professor
- Task2_PC_project.pdf (guideline documents from Professor)