Process Control - Project (WiSe 2020/21)

Control of a Multivariable Process

Submitted By:
Shailesh Kumar XXX (229916),
Veeranjaneyulu Akula (229473),
Arman Ahmed Khan (230601)
Electrical Engineering and Information Technology
Otto-von-Guericke-Universitat Magdeburg

Newell and Lee Evaporator

Evaporator model in Matlab/Simulink

Implement the **Evaporator model** in Matlab/Simulink via an S-function .

Evaporator Equations Matlab Code

```
function dxdt = evapmod(t, x, u)

% Inputs
F2 = u(2);                    % [kg/min]
P100 = u(3);             % [kPa]
F200 = u(4);                % [kg/min]
F1 = u(1);                    % [kg/min];
T1 = u(5);                  % [°C]
XF1 = u(6);                   % [%]
F3 = u(7);                  % [kg/min]
T200 = u(8);                % [°C]

% Outputs
X2 = x(1);
P2 = x(2);
L2 = x(3);

% Parameters
C = 4; Cp = 0.07;
lam = 38.5; lams = 36.6;  rhoA = 20;
M = 20; UA2 = 6.84;

% Algebraic equations
% Evaporator and 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
dxdt = [dX2dt dP2dt dL2dt]';
end
```

Matlab Code for S-Function

```
function [sys,x0] = evapmods(t, x, u, flag)
%
%  Simulink interface to evapmods.m
%
% Inputs:    t    - time in [min].
%            x    - 3 states, The level of the liquid in the separator tank L2.
%                   Concentration of the liquid obtained as the product X2.
%                   The operating pressure P2 in the evaporator.
%            u(1) - F1 , inlet feed flowrate
%             F2 = u(2);              % [kg/min]
%             P100 = u(3);       % [kPa]
%             F200 = u(4);          % [kg/min]
%              T1 = u(5);             % [°C]
%             XF1 = u(6);             % [%]
%             F3 = u(7);            % [kg/min]
%            T200 = u(8);
%
% Outputs:   sys and x0 as described in the SIMULINK manual.
%            when flag is 0 sys contains sizes and x0 contains
%            initial condition.
%            when flag is 1, sys contains the derivatives,
%            and when flag is 3 sys contains outputs;
%            y(1)    - Concentration of the liquid obtained as the product X2.
%            y(2)    - The operating pressure P2 in the evaporator.
%            y(3)    - The level of the liquid in the separator tank L2.
% Initialize :  define initial conditions,  X0
%               define system in terms of number of states, inputs etc.
%             e.g. sys = [2, 0, 2, 1, 0, 0];
%            1st array :  number of continuous states
%            2nd array :  number of discrete states
%            3rd array :  number of outputs
%            4th array :  number of inputs
%            5th array :  flag for direct feedthrough
%            6th array :  the number of sample times
if abs(flag) == 1
  % Return state derivatives.
  sys = evapmod(t,x,u);
elseif abs(flag) == 3
  % Return system outputs.
  sys(1,1,1) = x(1);       %  X2
  sys(2,1,1) = x(2);       %  P2
  sys(3,1,1) = x(3);       %  L2
```

```
elseif flag == 0
  % Initialize the system
  load init_ss
  x0=X0ss ;
  sys = [3, 0, 3, 8 , 0, 0];
else
  sys = [];
end
```

Simulink Model

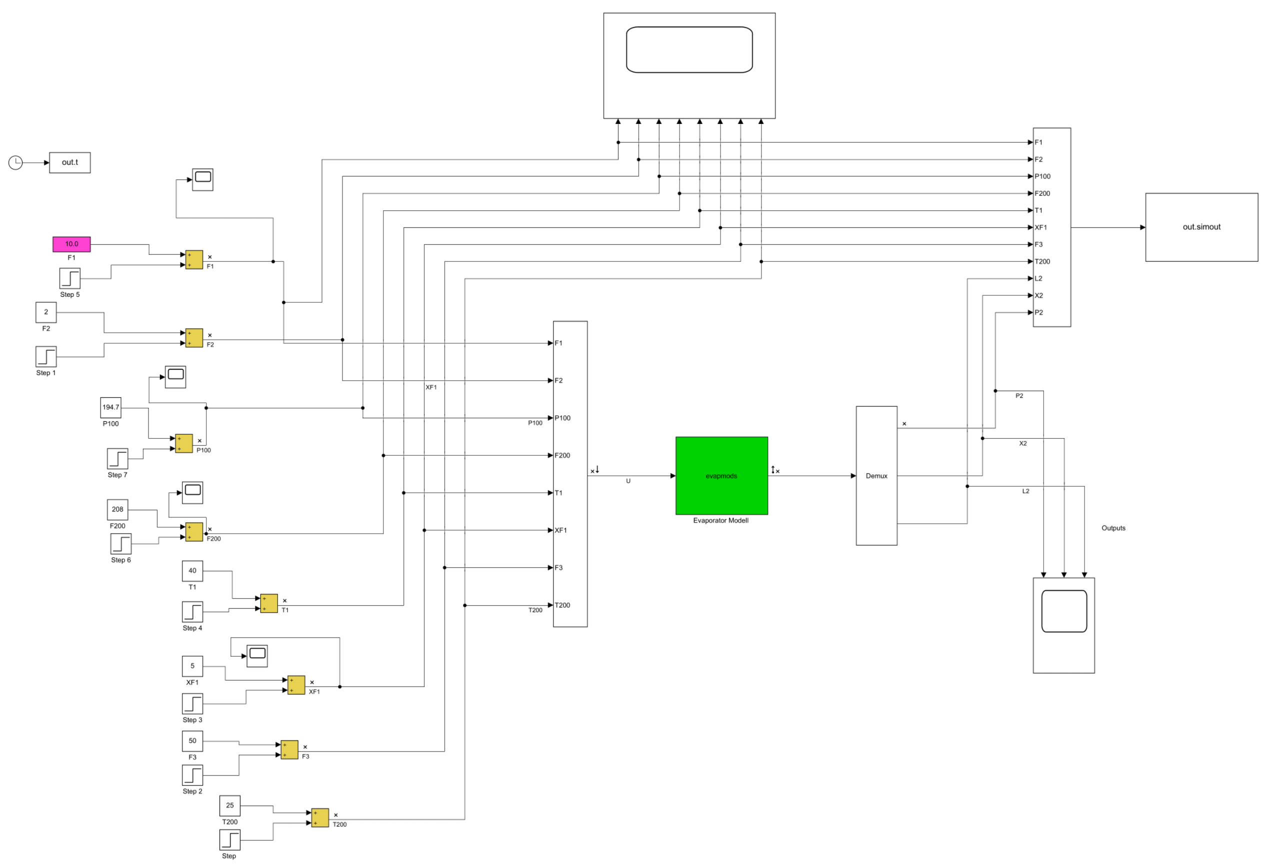

Fig 1 : Simulink Model of Evaporator

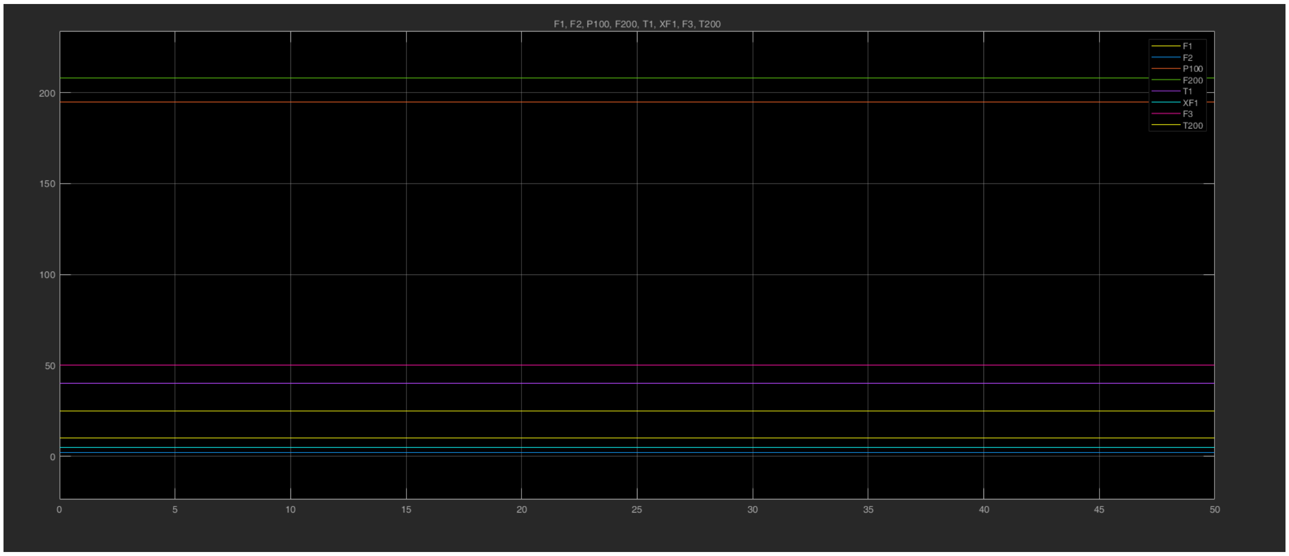

Fig 2 : Steady State Solution of the Input variables

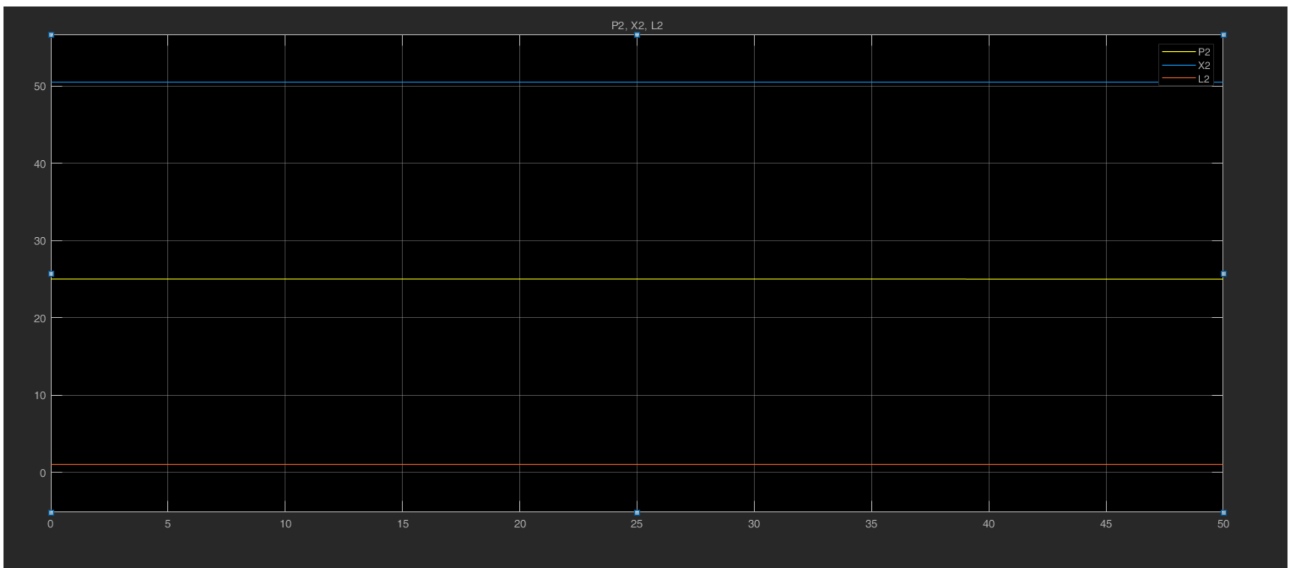

Fig 3 : Steady State Solution of the Controlled (output) variables

Dynamic simulation scenarios

Here we altered some parameters and re-evaluated the three outputs and their updated steady states.

Simulation scenarios : Step change in some disturbances .

i. A step change of + 15 % in the feed flow rate , F1
ii. A step change of - 15 % in the feed flow rate , F1
iii. A step change of + 2 % in the feed composition , x1
iv. A step change of - 2 % in the feed composition, x1

Presentation of results :

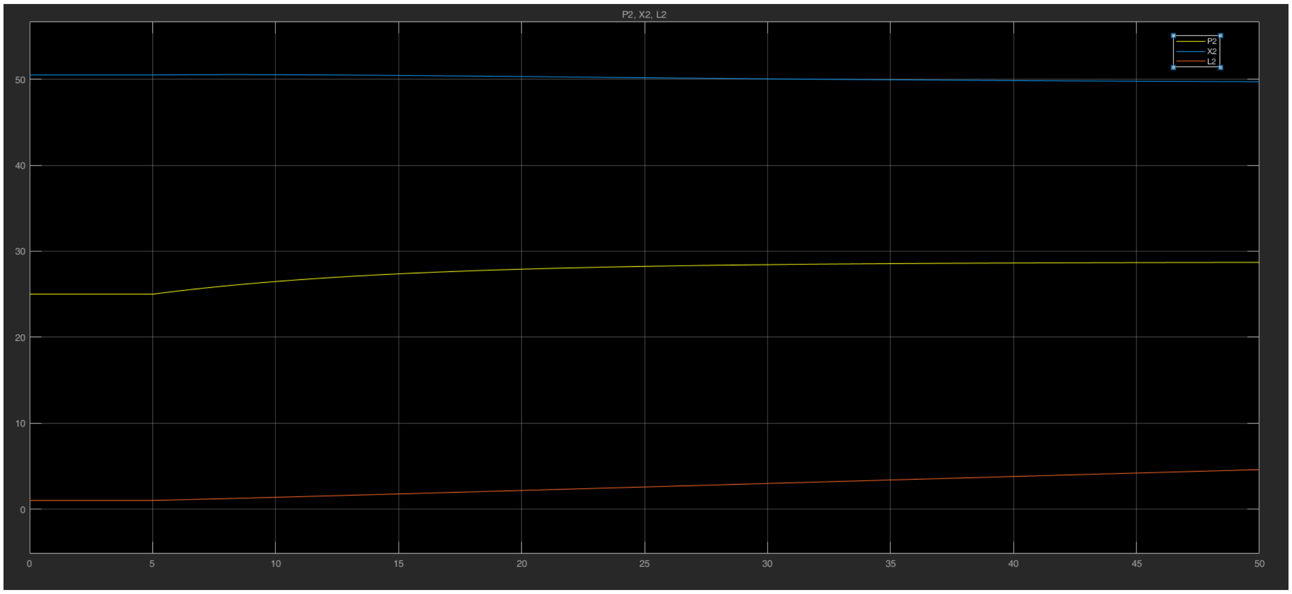

Fig 4 : A step change of + 15 % in the feed flow rate , F1

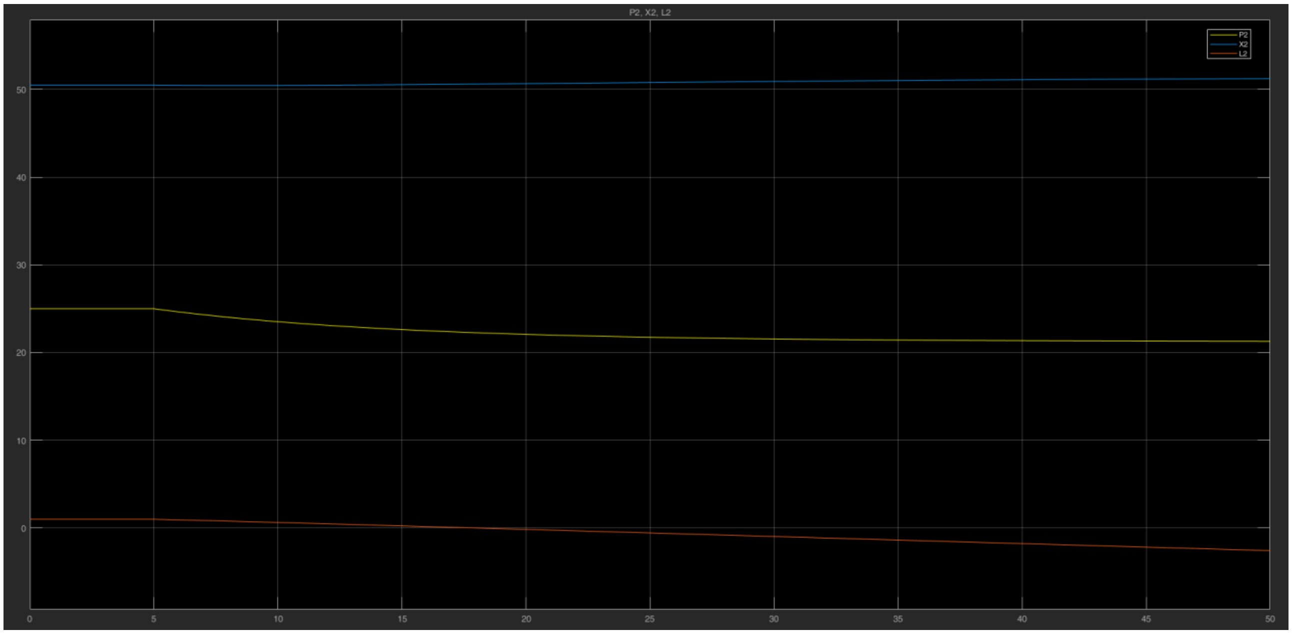

Fig 5 : A step change of - 15 % in the feed flow rate , F1

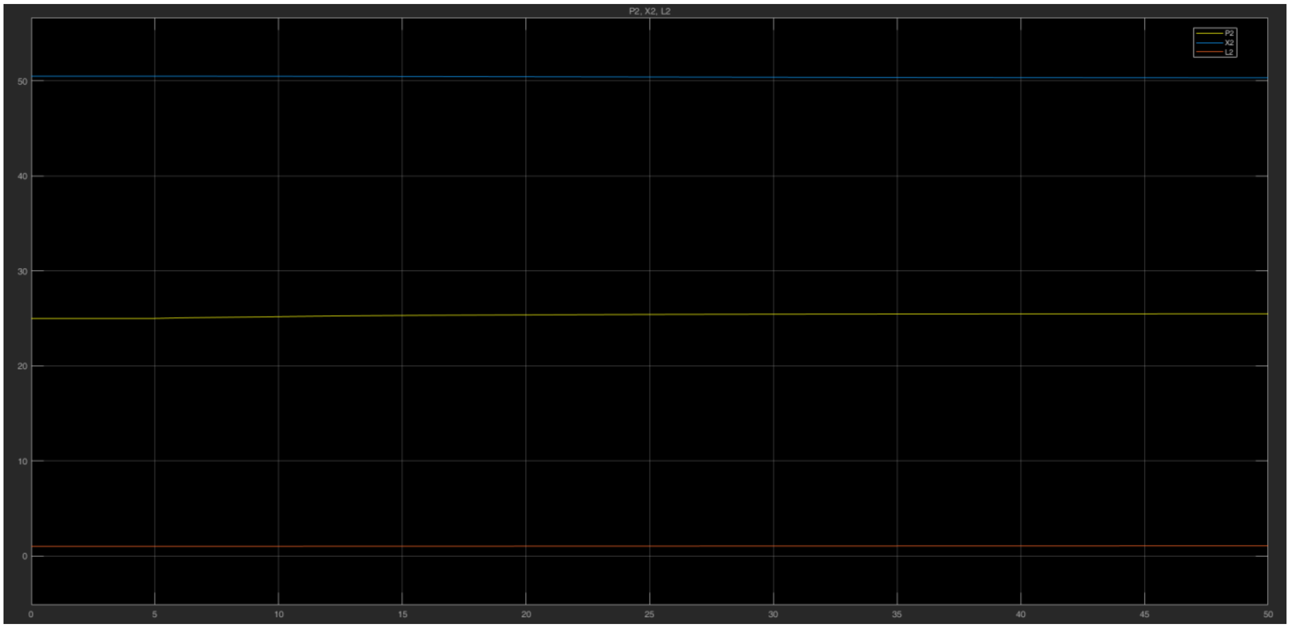

Fig 6 : A step change of + 2 % in the feed composition , x1

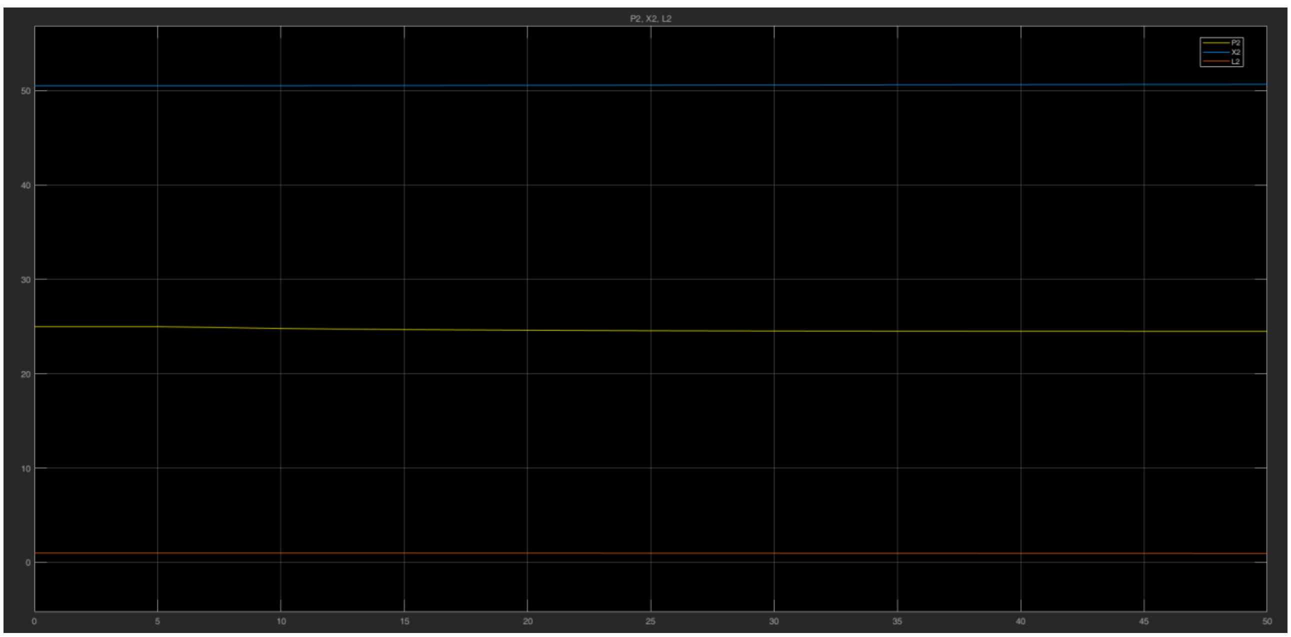

Fig 7 : A step change of - 2 % in the feed composition , x1

Additional simulation scenarios : Step change in some manipulated variables

i. A step change of + 15 % in F200
ii. A step change of - 15 % in F200
iii. A step change of + 15 % in P100
iv. A step change of - 15 % in P100

Presentation of results :

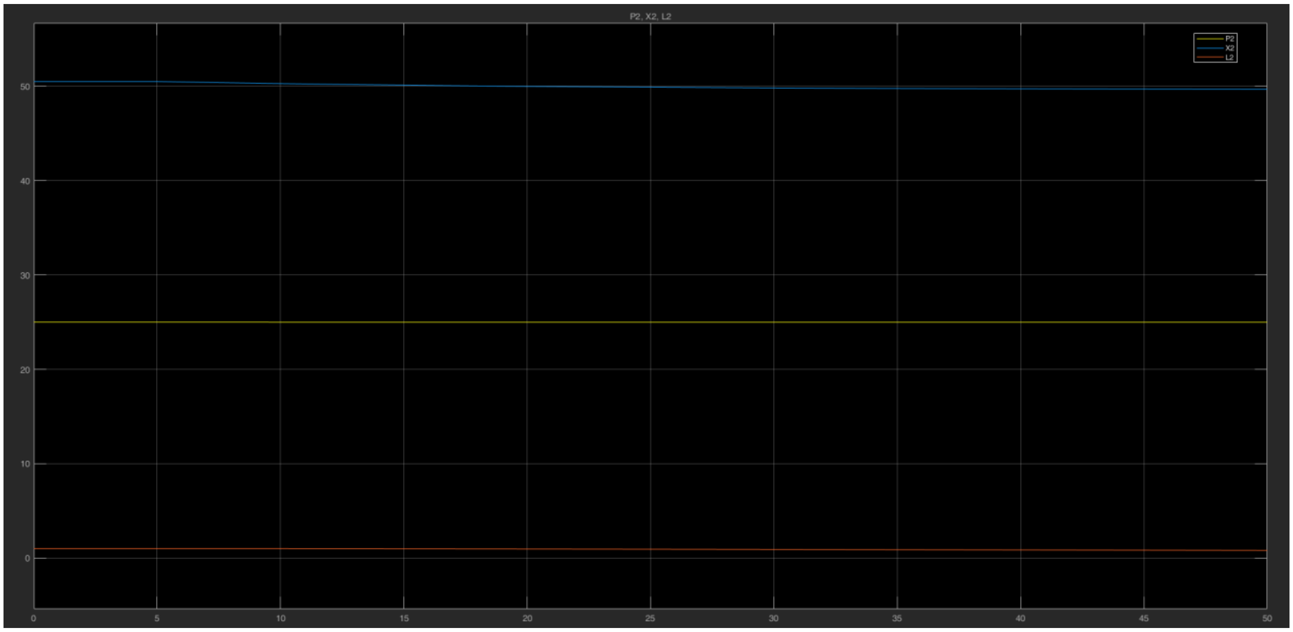

Fig 8 : A step change of + 15 % in F200

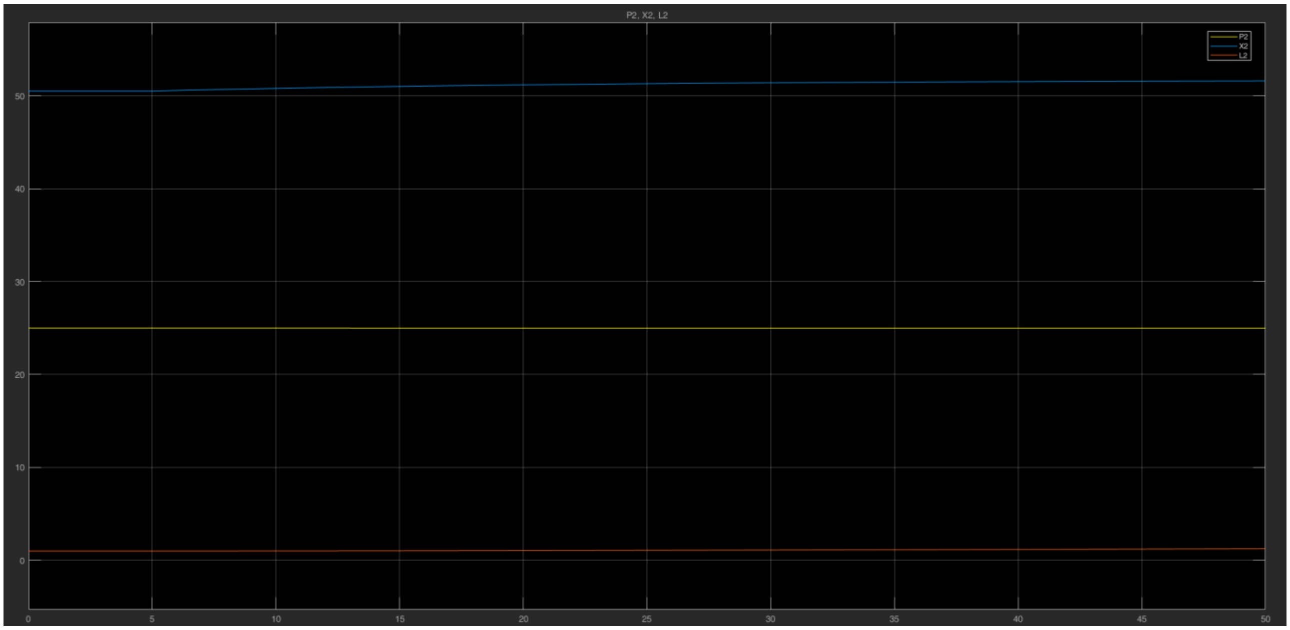

Fig 9 : A step change of - 15 % in F200

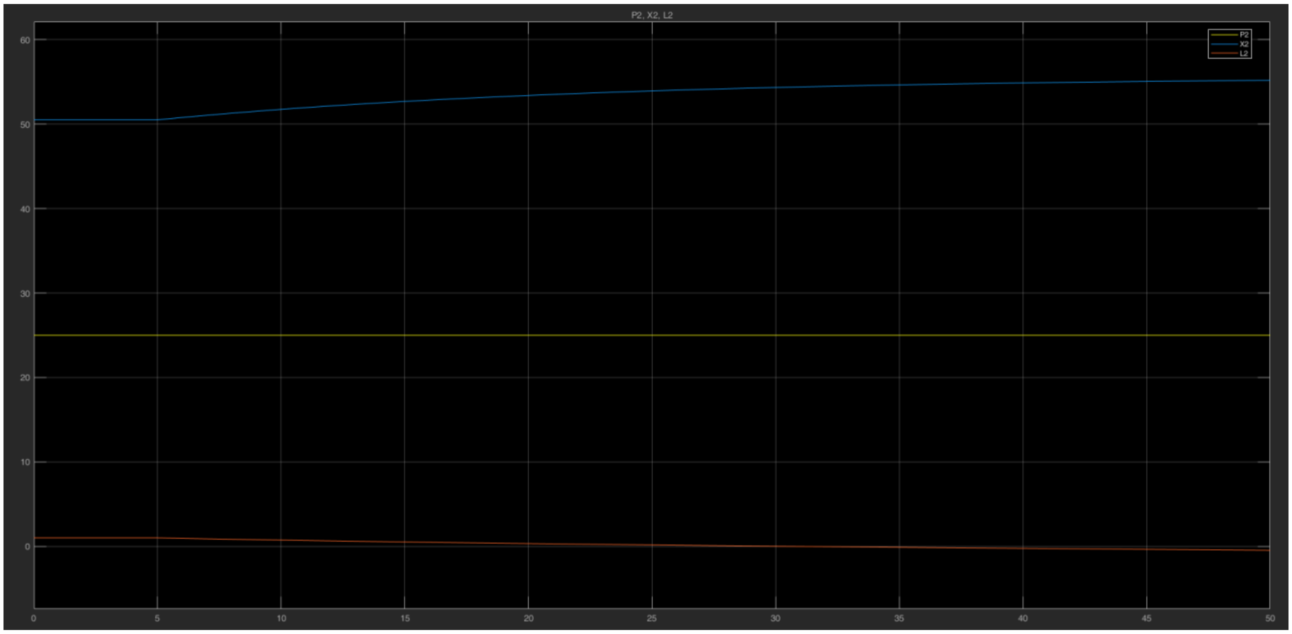

Fig 10 : A step change of + 15 % in P100

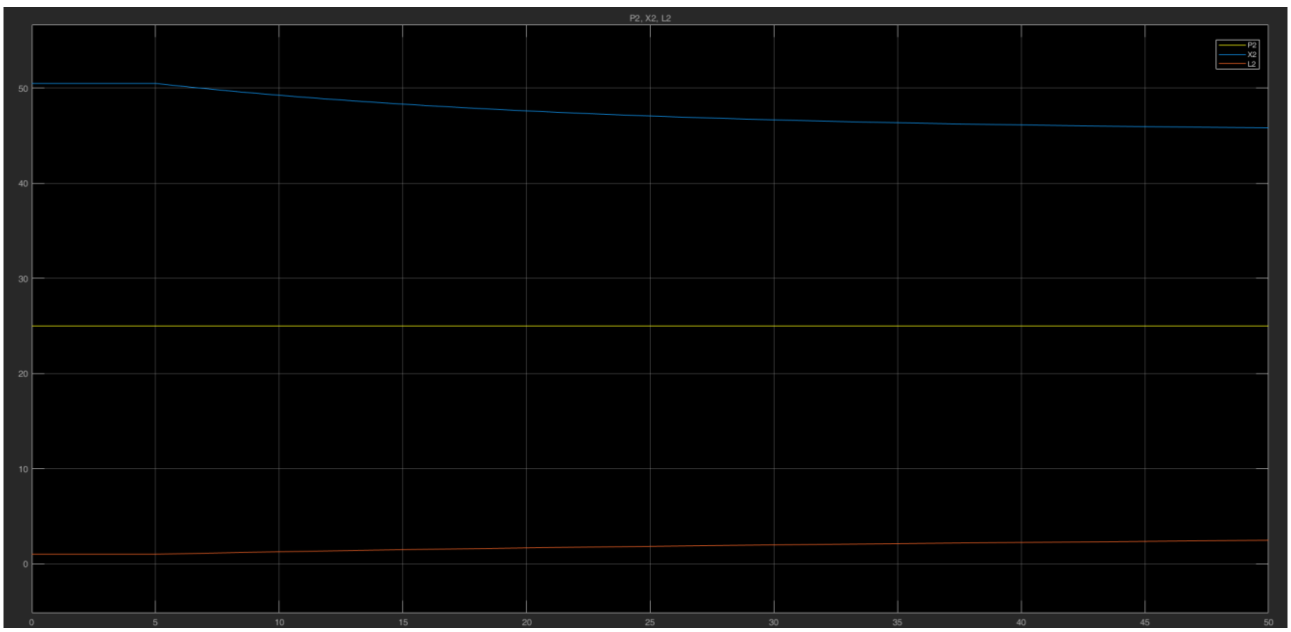

Fig 11 : A step change of - 15 % in P100

Linear Analysis of Evaporator Simulink model

Compute the linear system matrices, A,B,C,D :

```
>> [A,B,C,D] = ssdata(linsys1)

A =

   -0.1000         0         0
   -0.0209   -0.0558         0
    0.0042    0.0075         0


B =

    0.2500   -1.2500         0         0         0    0.5000         0         0
    0.0164         0    0.0096   -0.0018    0.0045         0    0.0367    0.0360
    0.0467   -0.0500   -0.0019         0   -0.0009         0   -0.0073         0


C =

    1.0000         0         0
         0    1.0000         0
         0         0    1.0000


D =

     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0
```

Compute G0 for the 2x2 MIMO system :

```
>> lambda=eig(A)

lambda =

         0
   -0.0558
   -0.1000
```

Computation of RGA0 based on G0 :

```
>> G0= -C*pinv(A)*B + D

G0 =

    2.4930  -12.4910    0.0001    0.0000    0.0000    4.9991    0.0003   -0.0006
   -0.7636    4.8395    0.1733   -0.0322    0.0822   -1.8894    0.6635    0.6334
         0         0         0         0         0         0         0         0

>> RGA0 = G0.*(pinv(G0'))
```

RGA0 =

0.2333	0.5757	0.0000	-0.0000	0.0000	0.1911	0.0001	-0.0002
-0.1598	0.2603	0.0327	0.0011	0.0073	-0.0568	0.4790	0.4362
0	0	0	0	0	0	0	0

Pole Zero Plot of Linear model of Evaporator

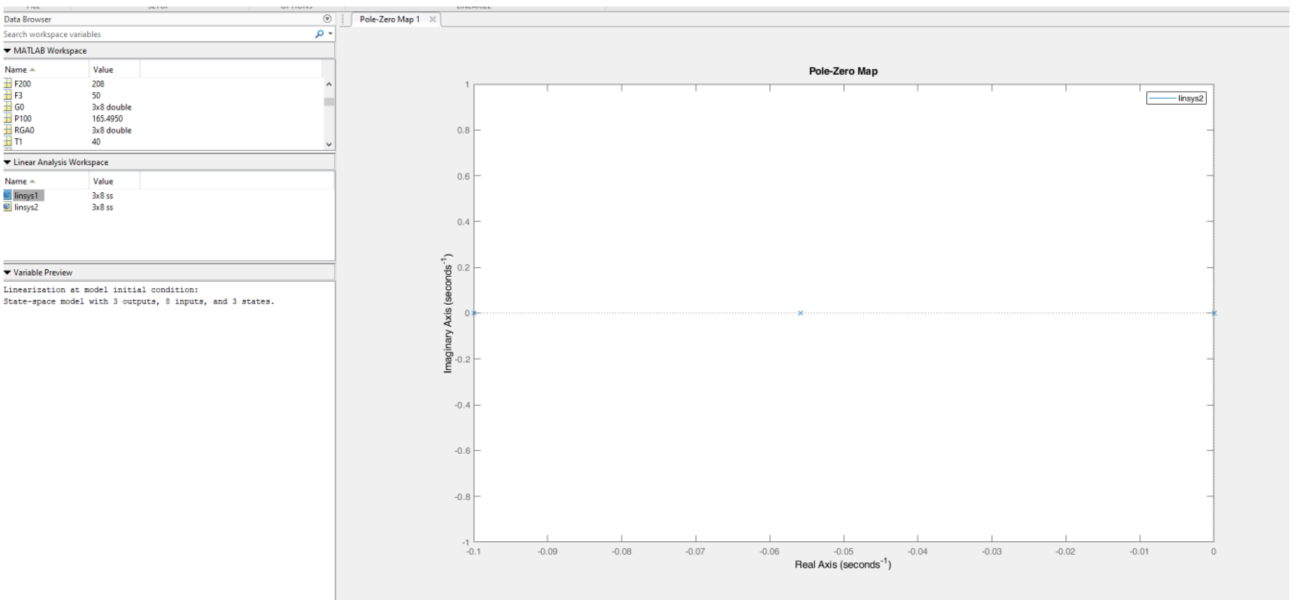

Fig 12 : Pole Zero Plot of Linear model of Evaporator

As we can see that there is in fact one zero eigen value hence we can't proceed further. Also the the general property of RGA satisfied is not satisfied. Hence we have to control L2 separately.

As we can see from the differential equation of L2, it is mostly dependent on F2 and F1. By applying P controller on F2 we can control L2.

Doing the RGA analysis after controlling L2 we have the following results.

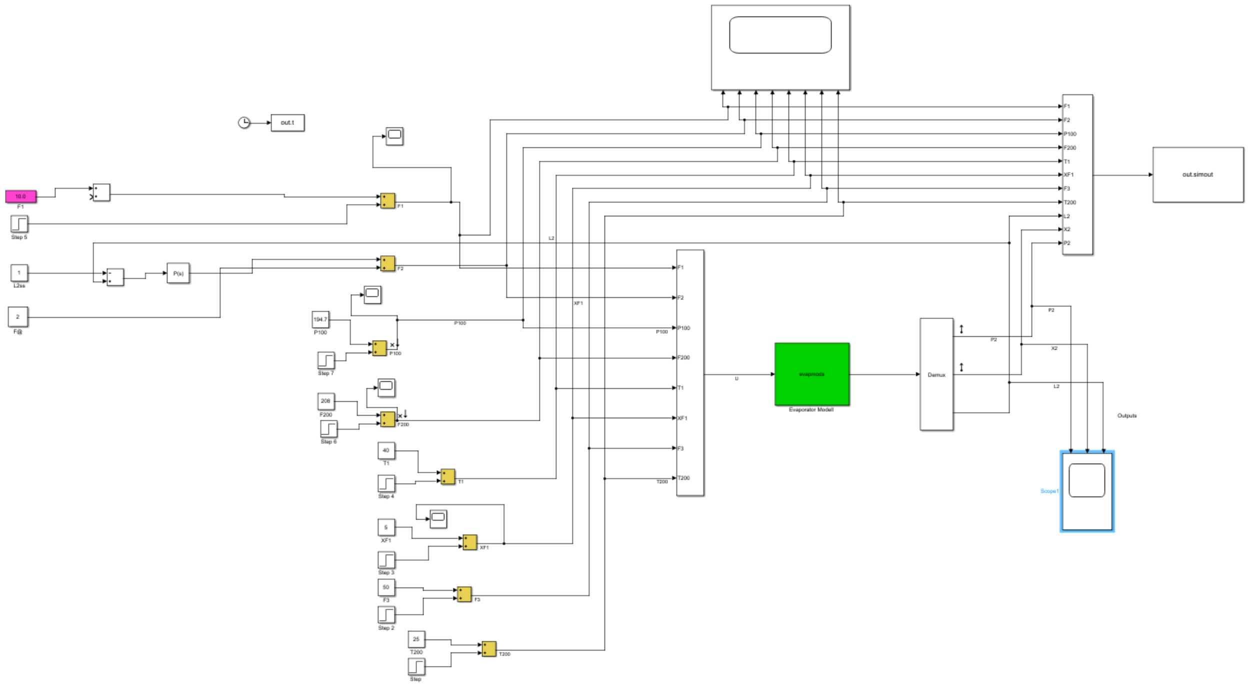

Fig 13 : Updated Simulink model of Evaporator

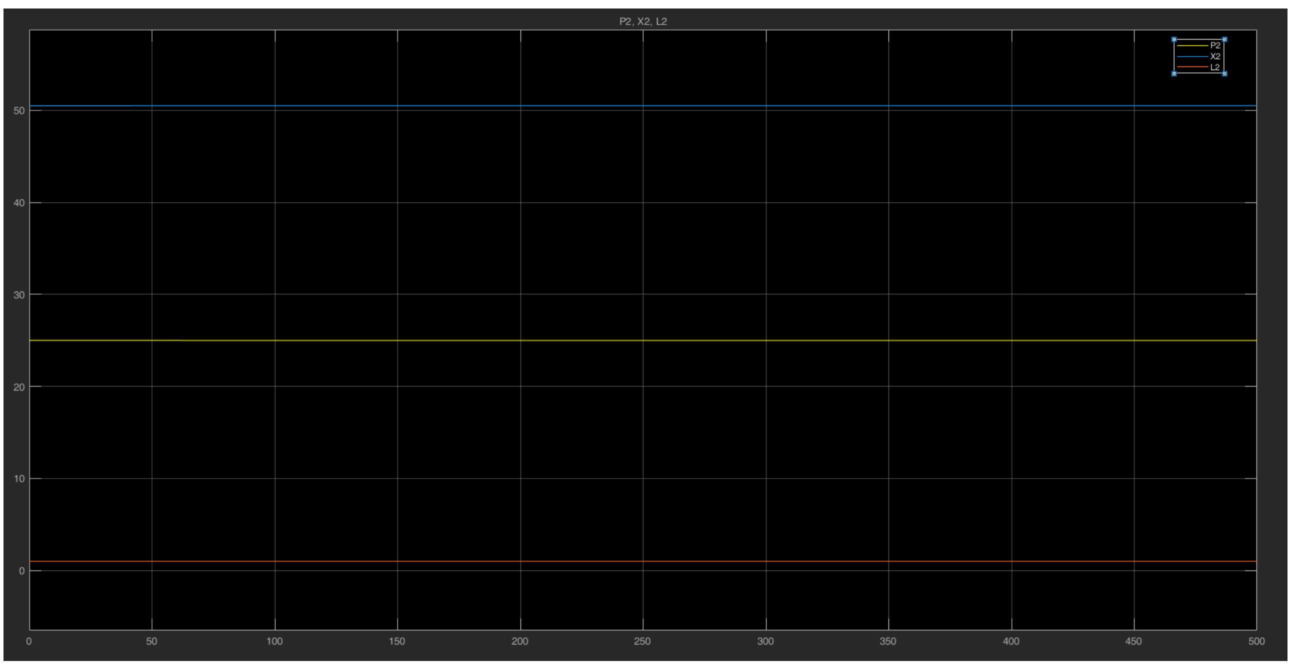

Fig 14 : Updated steady state output of Evaporator model.

Linear Analysis of Evaporator Simulink model without L2

Compute the linear system matrices, A,B,C,D :

```
>> [A,B,C,D] = ssdata(linsys1)
```

A =

```
   -0.1000         0   -1.2500
   -0.0209   -0.0558         0
    0.0042    0.0075   -0.0500


B =

         0         0
   -0.0018    0.0096
         0   -0.0019


C =

    1.0000         0         0
         0    1.0000         0


D =

     0     0
     0     0
```

Compute G0 for the 2x2 MIMO system :

```
>> ambda=eig(A)

ambda =

  -0.0893 + 0.0759i
  -0.0893 - 0.0759i
  -0.0273 + 0.0000i
```

Computation of RGA0 based on G0 :

```
>> G0= -C*pinv(A)*B + D

G0 =

    0.0459    0.1168
   -0.0500    0.1281

>> RGA0 = G0.*(pinv(G0'))

RGA0 =

    0.5018    0.4982
    0.4982    0.5018
```

Computation of Niderlinski index, NI:

```
>> G0jj =diag(G0)

G0jj =
```

```
    0.0459
    0.1281

>> NI = det(G0)/(G0jj(1)*G0jj(2))

NI =

    1.9930
```

References

- Two Tank Example Provided by Professor