ME16B063
Nitesh Kumar

Objective of this project is to design a controller using MPC toolbox for the Shell oil Fractionator. This is one of the standard problems in literature. Engineers at Shell oil company performed multiple step tests to develop the first order plus delay(FOPTD) models for the system.The two primary outputs of the fractionator are compositions of the top and side streams(y1,y2 respectively). Task is to design a controller to keep the y1 and y2 value at 0 using the developed FOPTD models which can handle load disturbances, model-mismatch and measurement noises.

The model structure used in an MPC controller is shown in the following
Illustration.

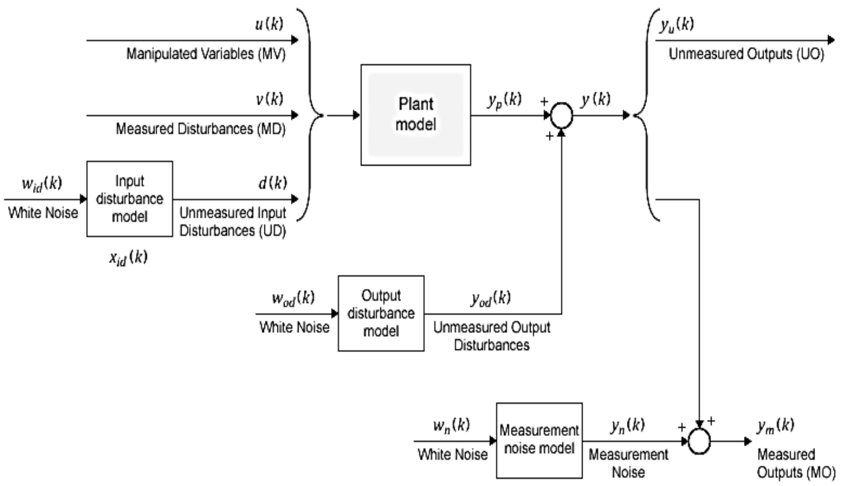

Fig 1

Case1: Measured disturbances with no model-plant mismatch (no MPM)

First of all a transfer function with 5 inputs and 3 outputs is obtained from the models given in problem statement and then it is converted to State-Space form with sample time of 2 (as shown below). 5 inputs= 3 M.V + 2 M.D

```
Gm=[Gm Gd];
Gm=c2d(ss(Gm),2);
```

Channels are defined for manipulated variables and measured disturbances using

```
Gm = setmpcsignals(Gm,'MV',[1 2 3],'MD',[4 5]);
```

After define mpc object and setting up the constraints for plots are obtained from

```
[y t u]=sim(MPCobj,Tf,r,v);    % v is measured disturbances
```

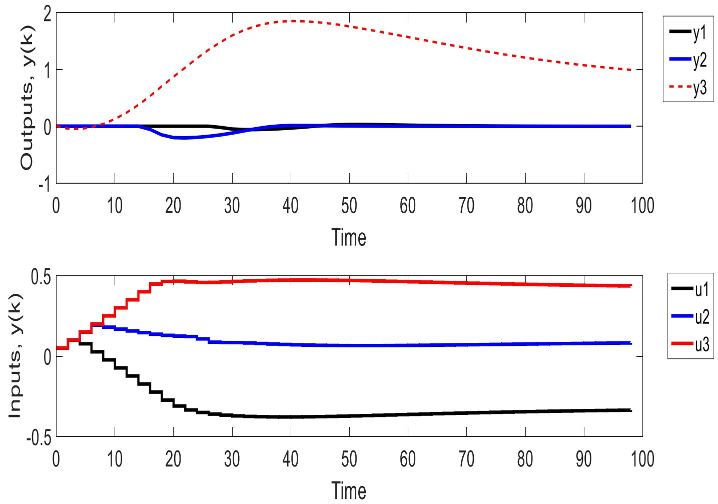

Fig 2

Fig 2 shows the results for measured disturbances. It can be interpreted from the plots the designed controller can take of measured disturbances.

Case2: Unmeasured disturbance with no model-plant mismatch (no MPM)

First of all a transfer function with 5 inputs and 3 outputs is obtained from the models given in problem statement and then it is converted to State-Space form with sample time of 2 (as shown below). 5 inputs= 3 M.V + 2 U.D

```
Gm=[Gm Gd];
Gm=c2d(ss(Gm),2);
```

Channels are defined for manipulated variables and measured disturbances using

```
Gm = setmpcsignals(Gm,'MV',[1 2 3],'UD',[4 5]);
```

MPC object was defined and input disturbance model is added to the MMPC object

```
MPCobj.Model.Disturbance = [tf(1,[1 0]); tf(1,[1 0])];

%For unmeasured input disturbances, its model is an integrator to make it
integrated white noises
```

This disturbance model will be used to change the white noise to Integrated White Noise.

```
options = mpcsimopt(MPCobj);
options.UnmeasuredDisturbance = [(randn(Tf,1)) (rand(Tf,1))] ; %white noise
```

Plots are obtained using :

```
[y t u]=sim(MPCobj,Tf,r,options);
```

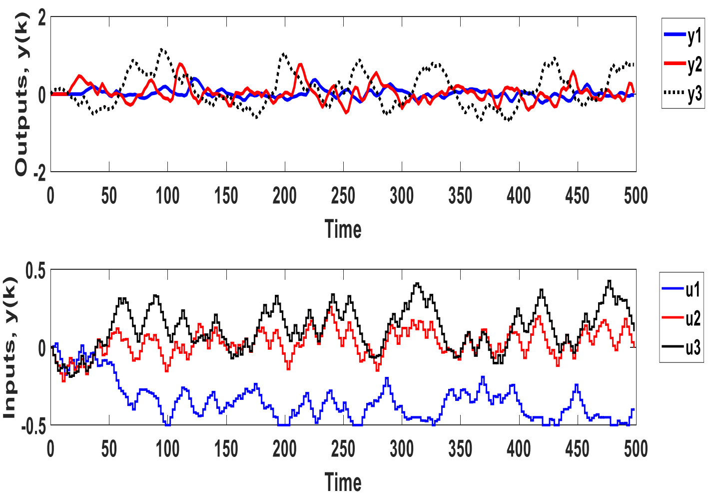

Fig 3

It can be seen from the plot that the our controller is able to handle Unmeasured Disturbances also. Value of y1 and y2 (objectives) is very close to zero at every time instant.

Case3: Model-Plant Mismatch

First of all a transfer function with 5 inputs and 3 outputs is obtained for both models and plant from the given information in the problem statement and then it is converted to State-Space form with sample time of 2 (as shown below). 5 inputs= 3 M.V + 2 U.D

```
Gm=[Gm Gd];
Gm=c2d(ss(Gm),2);

Gp=c2d(ss([Gp Gd]),2);
```

Then same steps as in Case2 were followed. For testing of Controller following steps are used.

```
options = mpcsimopt(MPCobj);
options.Unmeas = [(randn(Tf,1)) (randn(Tf,1))] ; % white noise
options.Model = Gp;

[y t u]=sim(MPCobj,Tf,r,options);
```

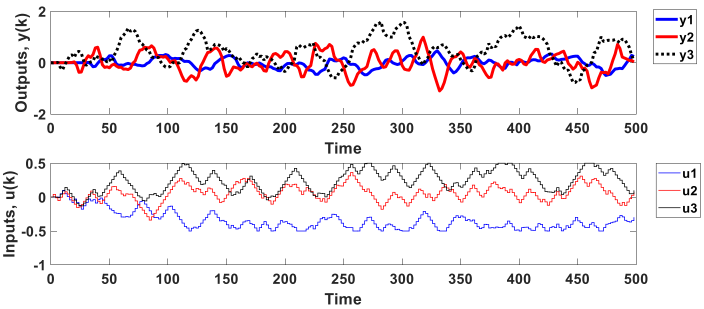

Fig 4

Here also our controller performed well but not good as compared to case2

Case4: Model-Plant Mismatch with Load Disturbance

In this case transfer function with 3 inputs 3 outputs is defined for both model and plant and converted to discrete time ss model.

```
Gp=c2d(ss(Gp),2);

Gm=c2d(ss(Gm),2);
```

As the load disturbances are present at the output we need to define output disturbance model as:

```
setoutdist(MPCobj,'model',Gd);
options = mpcsimopt(MPCobj);
options.OutputNoise=[randn(Tf,1) randn(Tf,1) randn(Tf,1)];
options.Model = Gp;
```

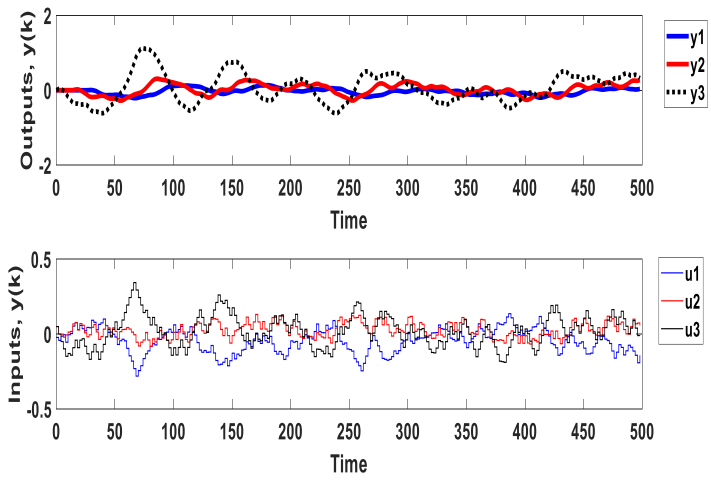

Fig5

Controller is able to reject output disturbance very effectively.

Codes:

Case1:

```
clear;
clc;
% Model

beta=0;
G11=tf(4.05+2.11*beta,[50 1],'InputDelay',27);
G12=tf(1.77-0.39*beta,[60 1],'InputDelay',28);
G13=tf(5.88+0.59*beta,[50 1],'InputDelay',27);
G21=tf(5.39+3.29*beta,[50 1],'InputDelay',18);
G22=tf(5.72-0.57*beta,[60 1],'InputDelay',14);
G23=tf(6.9+0.89*beta,[40 1],'InputDelay',15);
G31=tf(4.38+3.11*beta,[33 1],'InputDelay',20);
G32=tf(4.42-0.73*beta,[44 1],'InputDelay',22);
G33=tf(7.2+1.33*beta,[19 1]);

Gm=[G11 G12 G13; G21 G22 G23;G31 G32 G33];

clear G11 G12 G13 G21 G22 G23 G31 G32 G33;

% Model Disturbances

G11=tf(1.2+0.12*beta,[45 1],'InputDelay',27);
G12=tf(1.44+0.16*beta,[40 1],'InputDelay',27);
G21=tf(1.52+0.13*beta,[25 1],'InputDelay',15);
G22=tf(1.83+0.13*beta,[20 1],'InputDelay',15);
G31=tf(1.14+0.18*beta,[27 1]);
G32=tf(1.26+0.18*beta,[32 1]);
Gd=[G11 G12;G21 G22;G31 G32];

Gm=[Gm Gd];
Gm=c2d(ss(Gm),2);

clear G11 G12 G21 G22 G31 G32 Gd;



Gm = setmpcsignals(Gm,'MV',[1 2 3],'MD',[4 5]);
MPCobj=mpc(Gm);
MPCobj.P=40;
MPConj.C=10;
MPCobj.MV(1).Min = -0.5;
MPCobj.MV(1).Max = 0.5;
MPCobj.MV(1).RateMin = -0.05;
MPCobj.MV(1).RateMax = 0.05

MPCobj.MV(2).Min = -0.5;
MPCobj.MV(2).Max = 0.5;
MPCobj.MV(2).RateMin = -0.05;
MPCobj.MV(2).RateMax = 0.05;

MPCobj.MV(3).Min = -0.5;
MPCobj.MV(3).Max = 0.5;
MPCobj.MV(3).RateMin = -0.05;
```

```
MPCobj.MV(3).RateMax = 0.05;


MPCobj.OV(1).Min = -0.5;
MPCobj.OV(1).Max = 0.5;
MPCobj.OV(3).Min = -0.5;


MPCobj.W.OV=[1 1 0];                          % weight
MPCobj.W.MVRate=[1.5 0.15 1.5];               %manipulated rate weights


Tf=50;
r=[0 0 0];
v=[-0.5 -0.5];  %measured disturbances

[y t u]=sim(MPCobj,Tf,r,v);

subplot(2,1,1)
plot(t,y)
xlabel('Time'); ylabel('Outputs, y(k)')
subplot(2,1,2)
stairs(t,u)
xlabel('Time'); ylabel('Inputs, y(k)')
```

Case2:

```
beta=0;
G11=tf(4.05+2.11*beta,[50 1],'InputDelay',27);
G12=tf(1.77-0.39*beta,[60 1],'InputDelay',28);
G13=tf(5.88+0.59*beta,[50 1],'InputDelay',27);
G21=tf(5.39+3.29*beta,[50 1],'InputDelay',18);
G22=tf(5.72-0.57*beta,[60 1],'InputDelay',14);
G23=tf(6.9+0.89*beta,[40 1],'InputDelay',15);
G31=tf(4.38+3.11*beta,[33 1],'InputDelay',20);
G32=tf(4.42-0.73*beta,[44 1],'InputDelay',22);
G33=tf(7.2+1.33*beta,[19 1]);

Gm=[G11 G12 G13; G21 G22 G23;G31 G32 G33];

clear G11 G12 G13 G21 G22 G23 G31 G32 G33;


G11=tf(1.2+0.12*beta,[45 1],'InputDelay',27);
G12=tf(1.44+0.16*beta,[40 1],'InputDelay',27);
G21=tf(1.52+0.13*beta,[25 1],'InputDelay',15);
G22=tf(1.83+0.13*beta,[20 1],'InputDelay',15);
G31=tf(1.14+0.18*beta,[27 1]);
G32=tf(1.26+0.18*beta,[32 1]);
Gd=[G11 G12;G21 G22;G31 G32];
Gm=[Gm Gd];
Gm=c2d(ss(Gm),2);

clear G11 G12 G21 G22 G31 G32 Gd;

Gm = setmpcsignals(Gm,'MV',[1 2 3],'UD',[4 5]);
MPCobj=mpc(Gm);
MPCobj.P=40;
MPConj.C=10;
MPCobj.MV(1).Min = -0.5;
MPCobj.MV(1).Max = 0.5;
MPCobj.MV(1).RateMin = -0.05;
MPCobj.MV(1).RateMax = 0.05

MPCobj.MV(2).Min = -0.5;
MPCobj.MV(2).Max = 0.5;
MPCobj.MV(2).RateMin = -0.05;
MPCobj.MV(2).RateMax = 0.05;

MPCobj.MV(3).Min = -0.5;
MPCobj.MV(3).Max = 0.5;
MPCobj.MV(3).RateMin = -0.05;
MPCobj.MV(3).RateMax = 0.05;


MPCobj.OV(1).Min = -0.5;
MPCobj.OV(1).Max = 0.5;
MPCobj.OV(3).Min = -0.5;
```

```
MPCobj.W.OV=[1 1 0];                              % weight
MPCobj.W.MVRate=[1.5 0.15 1.5];                   %manipulated rate weights
MPCobj.Model.Disturbance = [tf(1,[1 0]); tf(1,[1 0])];  %For unmeasured
input disturbances, its model is an integrator to make it integrated white
noises

Ts=2
Tstop = 500;                                    % simulation time
Tf = round(Tstop/Ts);

options = mpcsimopt(MPCobj);
options.Model = Gm;
options.UnmeasuredDisturbance = [(randn(Tf,1)) (rand(Tf,1))] ;


r=[0 0 0];


[y t u]=sim(MPCobj,Tf,r,options);

subplot(2,1,1)
plot(t,y)
xlabel('Time'); ylabel('Outputs, y(k)')
subplot(2,1,2)
stairs(t,u)
xlabel('Time'); ylabel('Inputs, y(k)')
```

Case3:

```
clear;
clc;
G11=tf(4.05,[50 1],'InputDelay',27);
G12=tf(1.77,[60 1],'InputDelay',28);
G13=tf(5.88,[50 1],'InputDelay',27);
G21=tf(5.39,[50 1],'InputDelay',18);
G22=tf(5.72,[60 1],'InputDelay',14);
G23=tf(6.9,[40 1],'InputDelay',15);
G31=tf(4.38,[33 1],'InputDelay',20);
G32=tf(4.42,[44 1],'InputDelay',22);
G33=tf(7.2,[19 1]);

Gp=[G11 G12 G13; G21 G22 G23;G31 G32 G33];
clear G11 G12 G13 G21 G22 G23 G31 G32 G33;


G11=tf(1.2,[45 1],'InputDelay',27);
G12=tf(1.44,[40 1],'InputDelay',27);
G21=tf(1.52,[25 1],'InputDelay',15);
G22=tf(1.83,[20 1],'InputDelay',15);
G31=tf(1.14,[27 1]);
G32=tf(1.26,[32 1]);
Gd=[G11 G12;G21 G22;G31 G32];
Gp=c2d(ss([Gp Gd]),2);
clear G11 G12 G21 G22 G31 G32 Gd;



beta=1;
G11=tf(4.05+2.11*beta,[50 1],'InputDelay',27);
G12=tf(1.77-0.39*beta,[60 1],'InputDelay',28);
G13=tf(5.88+0.59*beta,[50 1],'InputDelay',27);
G21=tf(5.39+3.29*beta,[50 1],'InputDelay',18);
G22=tf(5.72-0.57*beta,[60 1],'InputDelay',14);
G23=tf(6.9+0.89*beta,[40 1],'InputDelay',15);
G31=tf(4.38+3.11*beta,[33 1],'InputDelay',20);
G32=tf(4.42-0.73*beta,[44 1],'InputDelay',22);
G33=tf(7.2+1.33*beta,[19 1]);

Gm=[G11 G12 G13; G21 G22 G23;G31 G32 G33];

clear G11 G12 G13 G21 G22 G23 G31 G32 G33;


G11=tf(1.2+0.12*beta,[45 1],'InputDelay',27);
G12=tf(1.44+0.16*beta,[40 1],'InputDelay',27);
G21=tf(1.52+0.13*beta,[25 1],'InputDelay',15);
G22=tf(1.83+0.13*beta,[20 1],'InputDelay',15);
G31=tf(1.14+0.18*beta,[27 1]);
G32=tf(1.26+0.18*beta,[32 1]);
Gd=[G11 G12;G21 G22;G31 G32];
Gm=[Gm Gd];
Gm=c2d(ss(Gm),2);
clear G11 G12 G21 G22 G31 G32 Gd;
```

```
Gp = setmpcsignals(Gp,'MV',[1 2 3],'UD',[4 5]);
Gm = setmpcsignals(Gm,'MV',[1 2 3],'UD',[4 5]);

MPCobj=mpc(Gm);
MPCobj.P=40;
MPConj.C=10;
MPCobj.MV(1).Min = -0.5;
MPCobj.MV(1).Max = 0.5;
MPCobj.MV(1).RateMin = -0.05;
MPCobj.MV(1).RateMax = 0.05

MPCobj.MV(2).Min = -0.5;
MPCobj.MV(2).Max = 0.5;
MPCobj.MV(2).RateMin = -0.05;
MPCobj.MV(2).RateMax = 0.05;

MPCobj.MV(3).Min = -0.5;
MPCobj.MV(3).Max = 0.5;
MPCobj.MV(3).RateMin = -0.05;
MPCobj.MV(3).RateMax = 0.05;


MPCobj.OV(1).Min = -0.5;
MPCobj.OV(1).Max = 0.5;
MPCobj.OV(3).Min = -0.5;


MPCobj.W.OV=[1 1 0];                              % weight
MPCobj.W.MVRate=[1.5 0.15 1.5];                   %manipulated rate weights
MPCobj.Model.Disturbance = [tf(1,[1 0]); tf(1,[1 0])];  %For unmeasured
input disturbances, its model is an integrator

Ts=2;
Tstop = 500;                                   % simulation time
Tf = round(Tstop/Ts);

options = mpcsimopt(MPCobj);
options.Unmeas = [(randn(Tf,1)) (randn(Tf,1))] ; % white noise
options.Model = Gp;


r=[0 0 0];


[y t u]=sim(MPCobj,Tf,r,options);

subplot(2,1,1)
plot(t,y)
xlabel('Time'); ylabel('Outputs, y(k)')
subplot(2,1,2)
stairs(t,u)
xlabel('Time'); ylabel('Inputs, u(k)')
```

Case4:

```
clear;
clc;
G11=tf(4.05,[50 1],'InputDelay',27);
G12=tf(1.77,[60 1],'InputDelay',28);
G13=tf(5.88,[50 1],'InputDelay',27);
G21=tf(5.39,[50 1],'InputDelay',18);
G22=tf(5.72,[60 1],'InputDelay',14);
G23=tf(6.9,[40 1],'InputDelay',15);
G31=tf(4.38,[33 1],'InputDelay',20);
G32=tf(4.42,[44 1],'InputDelay',22);
G33=tf(7.2,[19 1]);

Gp=[G11 G12 G13; G21 G22 G23;G31 G32 G33];
clear G11 G12 G13 G21 G22 G23 G31 G32 G33;


Gp=c2d(ss(Gp),2);


G11=tf(1.2,[45 1],'InputDelay',27);
G12=tf(1.44,[40 1],'InputDelay',27);
G21=tf(1.52,[25 1],'InputDelay',15);
G22=tf(1.83,[20 1],'InputDelay',15);
G31=tf(1.14,[27 1]);
G32=tf(1.26,[32 1]);
Gd=[G11 G12;G21 G22;G31 G32];

clear G11 G12 G21 G22 G31 G32;

beta=1;
G11=tf(4.05+2.11*beta,[50 1],'InputDelay',27);
G12=tf(1.77-0.39*beta,[60 1],'InputDelay',28);
G13=tf(5.88+0.59*beta,[50 1],'InputDelay',27);
G21=tf(5.39+3.29*beta,[50 1],'InputDelay',18);
G22=tf(5.72-0.57*beta,[60 1],'InputDelay',14);
G23=tf(6.9+0.89*beta,[40 1],'InputDelay',15);
G31=tf(4.38+3.11*beta,[33 1],'InputDelay',20);
G32=tf(4.42-0.73*beta,[44 1],'InputDelay',22);
G33=tf(7.2+1.33*beta,[19 1]);

Gm=[G11 G12 G13; G21 G22 G23;G31 G32 G33];

clear G11 G12 G13 G21 G22 G23 G31 G32 G33;


Gm=c2d(ss(Gm),2);

Gp = setmpcsignals(Gp,'MV',[1 2 3]);
Gm = setmpcsignals(Gm,'MV',[1 2 3]);

MPCobj=mpc(Gm);
MPCobj.P=40;
MPConj.C=10;
MPCobj.MV(1).Min = -0.5;
MPCobj.MV(1).Max = 0.5;
```

```
MPCobj.MV(1).RateMin = -0.05;
MPCobj.MV(1).RateMax = 0.05

MPCobj.MV(2).Min = -0.5;
MPCobj.MV(2).Max = 0.5;
MPCobj.MV(2).RateMin = -0.05;
MPCobj.MV(2).RateMax = 0.05;

MPCobj.MV(3).Min = -0.5;
MPCobj.MV(3).Max = 0.5;
MPCobj.MV(3).RateMin = -0.05;
MPCobj.MV(3).RateMax = 0.05;


MPCobj.OV(1).Min = -0.5;
MPCobj.OV(1).Max = 0.5;
MPCobj.OV(3).Min = -0.5;


MPCobj.W.OV=[1 1 0];                          % weight
MPCobj.W.MVRate=[1.5 0.15 1.5];               %manipulated rate weights

Ts=2
Tstop = 500;                                 % simulation time
Tf = round(Tstop/Ts);

setoutdist(MPCobj,'model',Gd);% integrator for white noise
options = mpcsimopt(MPCobj);
options.OutputNoise=[randn(Tf,1) randn(Tf,1) randn(Tf,1)];
options.Model = Gp;


r=[0 0 0];


[y t u]=sim(MPCobj,Tf,r,options);

subplot(2,1,1)
plot(t,y)
xlabel('Time'); ylabel('Outputs, y(k)')
subplot(2,1,2)
stairs(t,u)
xlabel('Time'); ylabel('Inputs, y(k)')
```