%% Nominal

clear

close all

clc

load nominal\_residuals %res\_nom complete fault sensitivity matrix 31x31

% To Lukas: change here when you have to do the plots for sensors 10 and 12

% instead

sensor1=14;

sensor2=30;

Omega=[res\_nom(sensor1,:);res\_nom(sensor2,:)]; %leak sensitivity matrix considering only sensor1 and sensor2 installed

figure(1)

scatter (Omega(1,:),Omega(2,:), 'filled')

title ('Nominal Residuals for the 31 different leaks')

xlabel ('Pressure residual in node 14')

ylabel ('Pressure residual in node 30')

box on

hold on

plot (0,0,'ko','MarkerSize',10) %plot origing

quiver(0,0,Omega(1,1),Omega(2,1),0) %Plot leak 1 direction residuals

quiver(0,0,Omega(1,sensor1),Omega(2,sensor1),0) %Plot leak in sensor1 node direction residuals

quiver(0,0,Omega(1,sensor2),Omega(2,sensor2),0) %Plot leak

%% Hanoi

% To Lukas: comment to select which data you want to use as the variables are named

% the same way, you need one f\_20 for sensors 14 and 30 and then another

% f\_20 for sensors 10 and 12, the same for f\_50. The difference is the size

% of the leaks, in f\_50 they are bigger.

load hanoi\_residuals\_f\_20.mat

%load hanoi\_residuals\_f\_50.mat

r1=squeeze(res\_dufu(sensor1,:,:)); %the available residuals considering sensors 1 and 2 are stored in r1 and r2

r2=squeeze(res\_dufu(sensor2,:,:));

% To Lukas: make this one pretty

figure

hold on

for leak=1:31

%Real Residuals

plot (r1(:,leak),r2(:,leak),'x')

end

plot (0,0,'ko','MarkerSize',10) %plot origin

title(['Hanoi Residuals f\_{20}'])

quiver(0,0,Omega(1,1),Omega(2,1),0) %Plot leak 1 direction residuals

quiver(0,0,Omega(1,sensor1),Omega(2,sensor1),0) %Plot leak in sensor1 node direction residuals

quiver(0,0,Omega(1,sensor2),Omega(2,sensor2),0) %Plot leak in sensor2 node direction residuals

box on

%computation correlation for all residuals leak1 with hypotesys 1, 14 and

%30 (i.e leaks in these nodes)

N\_residuals=length(r1);

Gamma=zeros(31,31); %Confuision matrix;

for leak=1:31 % All the leaks have to be studied.

for k=1:N\_residuals

V\_Ro = zeros(31,1);

for hypothesis=1:31

V\_Ro(hypothesis) = [r1(k,leak),r2(k,leak)] \* [Omega(1,hypothesis),Omega(2,hypothesis)]' / ...

(norm([r1(k,leak),r2(k,leak)]) \* norm([Omega(1,hypothesis),Omega(2,hypothesis)]));

end

[max\_phro, winner] = max(V\_Ro);

[min\_phro, loser] = min(V\_Ro);

Gamma(leak, winner) = Gamma(leak, winner) + 1;

end

end

%%Computing ATD

load matrix\_D.mat %Matrix D 31x31 contains all the possible node distances (in nodes)

ATD=0;

atd\_vec = zeros(1,31);

%leak=1; %All the leaks have to be studied.

for leak=1:31

for hypothesis=1:31,

ATD=ATD+Gamma(leak,hypothesis)\*D(leak,hypothesis);

end

atd\_vec(leak)=ATD/(sum(Gamma(leak,:)));

end

ATD=ATD/(31\*N\_residuals) %Considering Remark 3 in Activity description