Problem 1)

a1 = [-0.1950 -0.9750 -1.5955 -1.9114];

a2 = ones(1,4)\*0.95;

R = [];

eigvals = [];

figure

colors = ['r', 'b', 'g', 'm'];

% Calculates the four 2x2 correlation matrices and stores them in

% a 2x8 matrix. Also computes the eigenvalues/eigenvectors and plots

% them

vectPlot = figure

valPlot = figure

for n = 1:4

tmp\_R = [1 (-a1(n)/(1+a2(n))\*1) ; (-a1(n)/(1+a2(n))\*1) 1]

[tmp\_eigvecs tmp\_eigvals] = eig(tmp\_R)

R = [R tmp\_R]

eigvals = [eigvals [tmp\_eigvals(1,1); tmp\_eigvals(2,2)]]

figure(vectPlot)

plot(tmp\_eigvecs, colors(n));

hold on

figure(valPlot)

plot([tmp\_eigvals(1,1); tmp\_eigvals(2,2)])

hold on

end

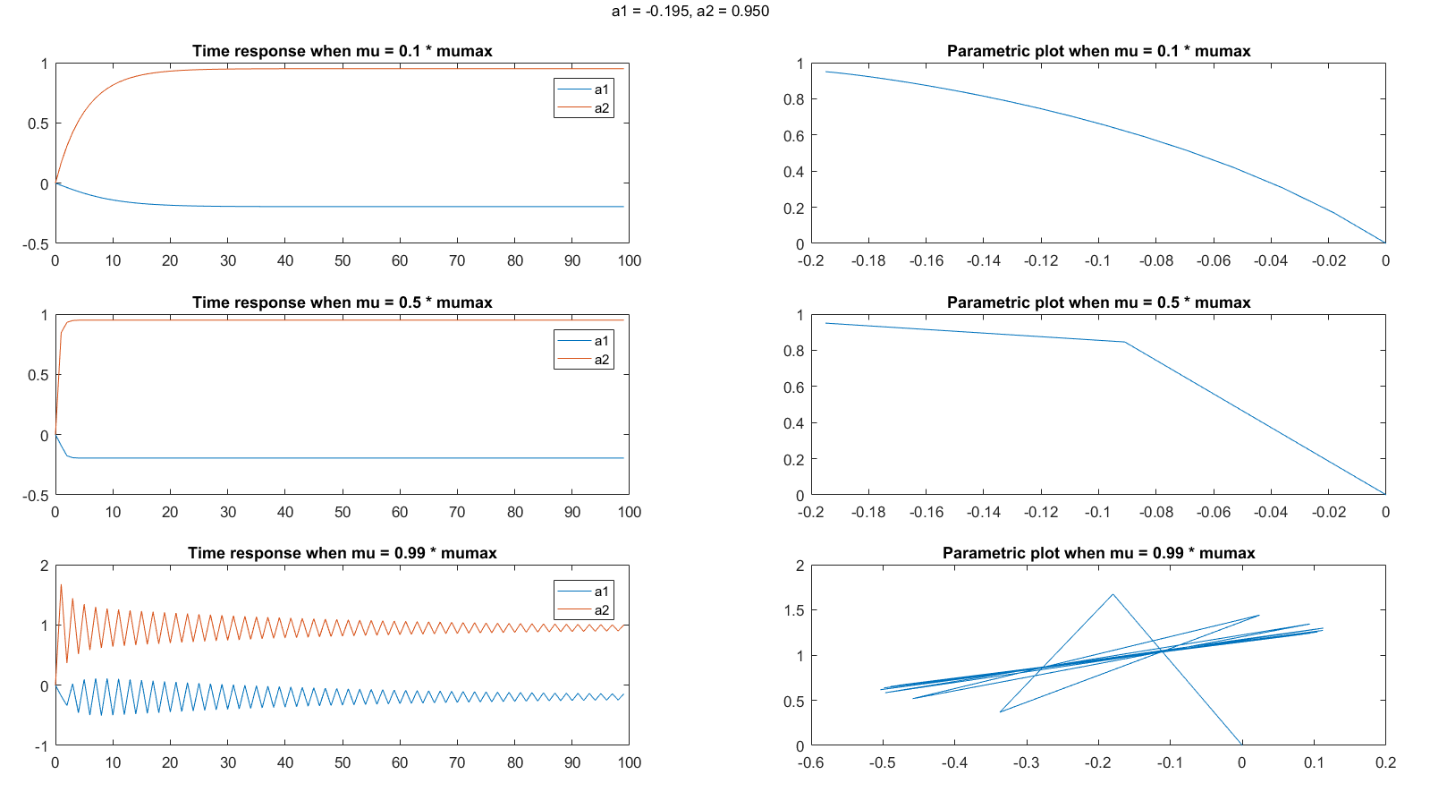
figure(vectPlot)

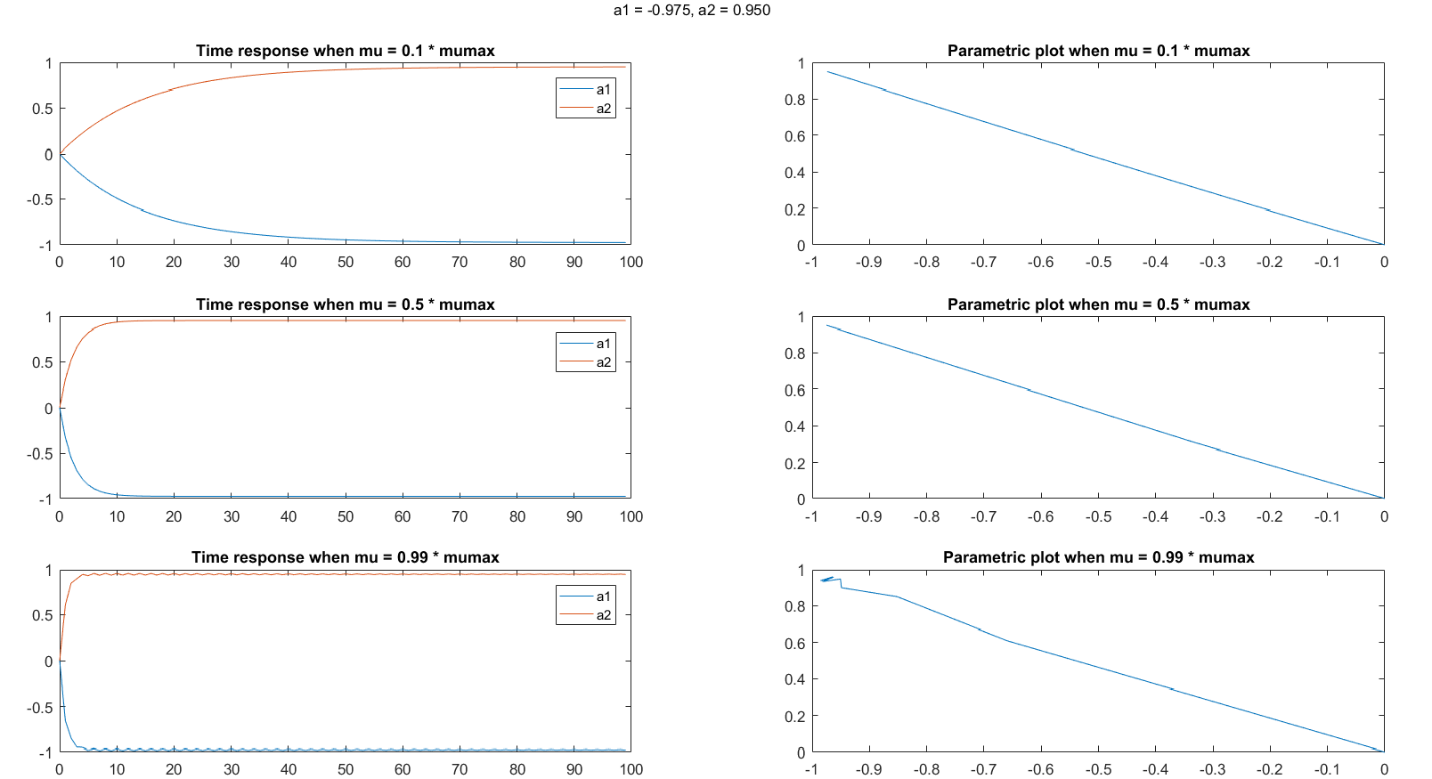
title('Eigenvectors of all sets of coeffecients')

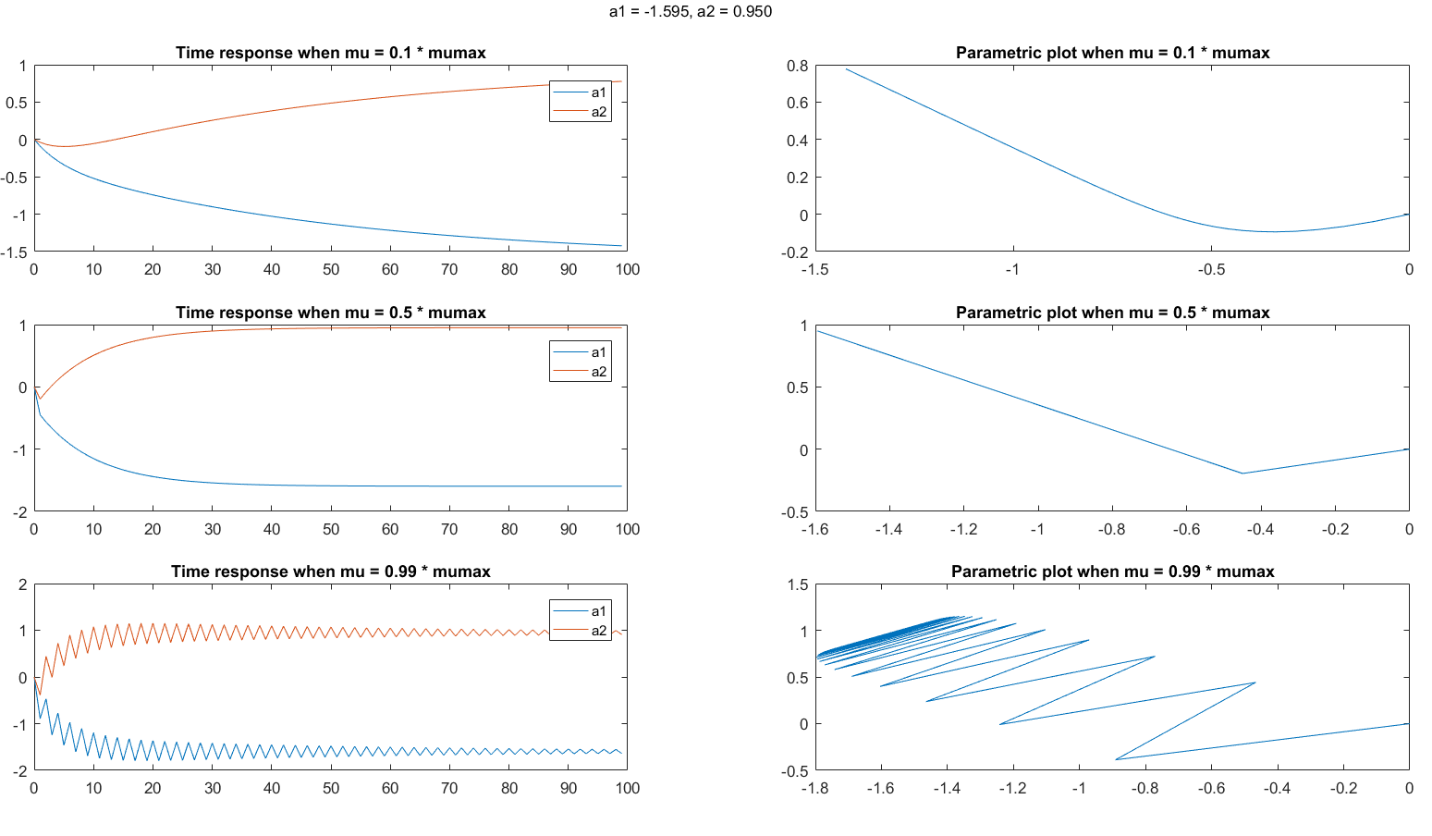
figure(valPlot)

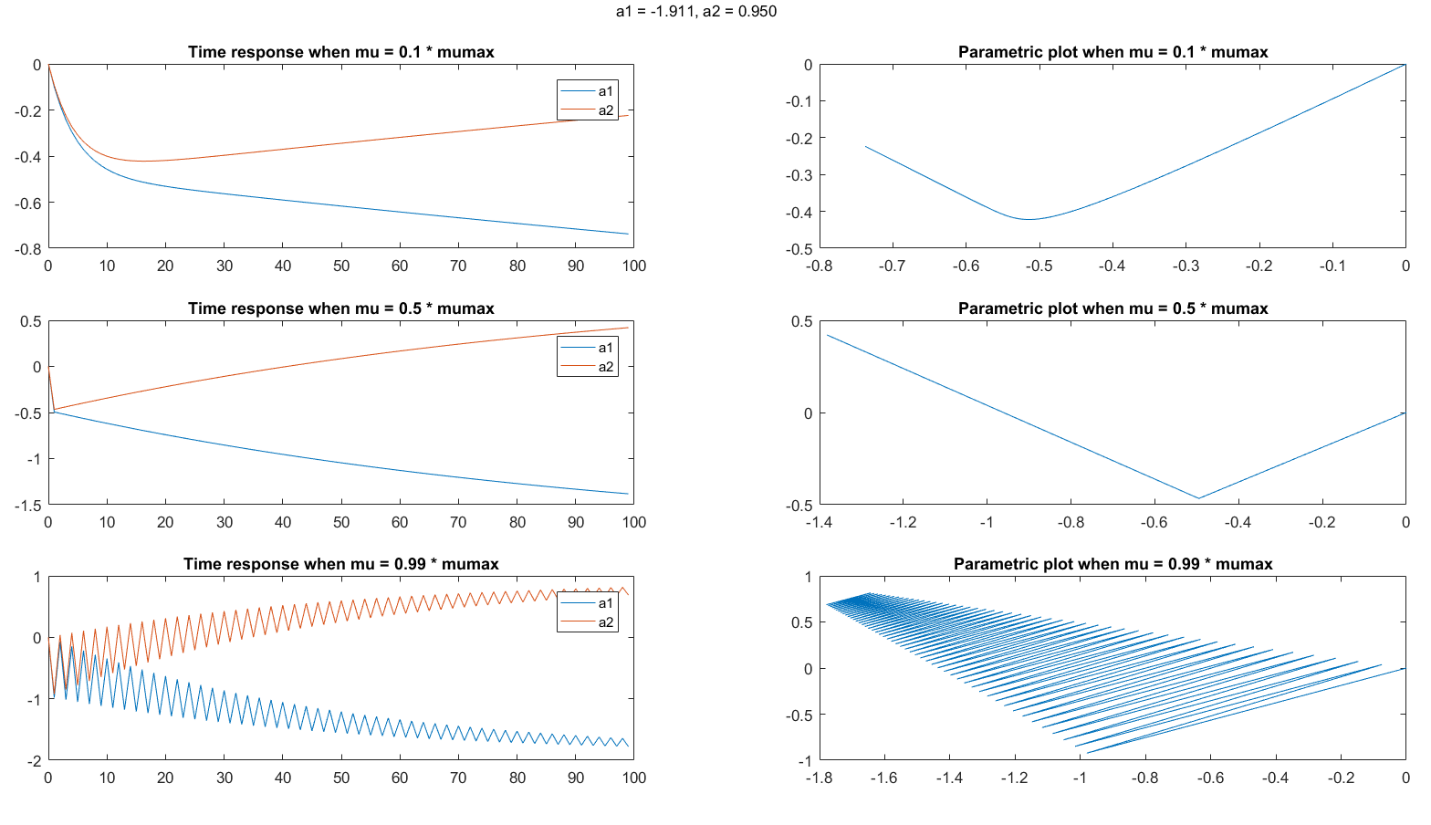
title('Eigenvalues of all sets of coeffecients')

Problem 2)









N = 100;

ww\_sv1 = zeros(8, N);

ww\_sv2 = zeros(8, N);

ww\_sv3 = zeros(8, N);

% Applies the gradient descent algorithm for each correlation matrix. Result

% is stored in a 8x100 matrix, where rows 1,2 are the coeffecients for

% the first correlation matrix, rows 3,4 are for the second correlation

% matrix, and so on.

for k = 1:4

m = 2\*k -1;

Rtmp = R(:, m:m+1);

p = Rtmp \* [a1(k); a2(k)];

mumax = 2/max(eigvals(:,k));

ww1=zeros(1,2)';

ww2=zeros(1,2)';

ww3=zeros(1,2)';

for nn=1:N

% mu = 0.1\*mumax

ww\_sv1(m:(m+1),nn)=ww1;

mu = 0.1 \* mumax;

ww1=ww1+mu\*(p-Rtmp(1:2, 1:2)\*ww1);

% mu = 0.5 \* mumax

ww\_sv2(m:(m+1),nn)=ww2;

mu = 0.5 \* mumax;

ww2=ww2+mu\*(p-Rtmp(1:2, 1:2)\*ww2);

% mu = 0.99 \* mumax

ww\_sv3(m:(m+1),nn)=ww3;

mu = 0.99 \* mumax;

ww3=ww3+mu\*(p-Rtmp(1:2, 1:2)\*ww3);

end

end

for n = 1:4

figure

m = 2 \* n -1;

% Plots the data for mu = 0.1\*mumax

subplot(3,2,1)

plot(0:N-1, ww\_sv1(m:m+1, :))

legend('a1', 'a2')

title('Time response when mu = 0.1 \* mumax')

subplot(3,2,2)

plot(ww\_sv1(m,1:N), ww\_sv1(m+1,1:N))

title('Parametric plot when mu = 0.1 \* mumax')

% Plots the data for mu = 0.5\*mumax

subplot(3,2,3)

plot(0:N-1, ww\_sv2(m:m+1, :))

legend('a1', 'a2')

title('Time response when mu = 0.5 \* mumax')

subplot(3,2,4)

plot(ww\_sv2(m,1:N), ww\_sv2(m+1,1:N))

title('Parametric plot when mu = 0.5 \* mumax')

% Plots the data for mu = 0.99\*mumax

subplot(3,2,5)

plot(0:N-1, ww\_sv3(m:m+1, :))

legend('a1', 'a2')

title('Time response when mu = 0.99 \* mumax')

subplot(3,2,6)

plot(ww\_sv3(m,1:N), ww\_sv3(m+1,1:N))

title('Parametric plot when mu = 0.99 \* mumax')

ha = axes('Position',[0 0 1 1],'Xlim',[0 1],'Ylim',[0

1],'Box','off','Visible','off','Units','normalized', 'clipping' , 'off');

text(0.5, 1,sprintf('a1 = %0.3f, a2 = %0.3f', a1(n), a2(n)),'HorizontalAlignment','center','VerticalAlignment', 'top')

end