Contents


```
%Tyler Matthews
%System Simluation Midterm P2
clc; close all;
```

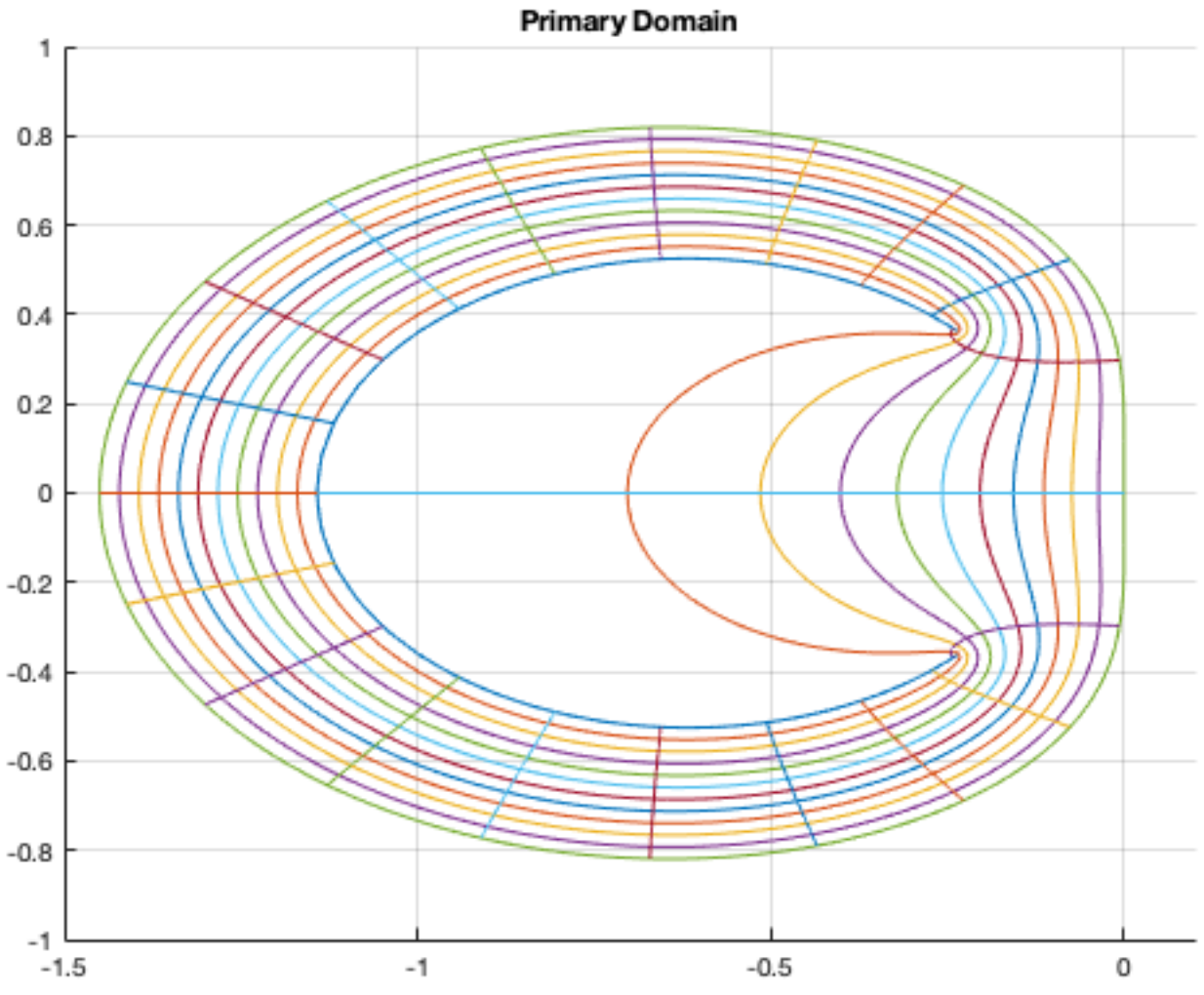
PARTC

```
num = [0 1.27 -0.73];
den = [1 -1.45 0.45];


Hp = tf(num, den)
zeros = roots(num)
poles = roots(den)

Phi = tf(den, num) %(sigma / roe) : (row - l*sigma)
newNum = [12700 -14600 4870] %Numerator of derivative of Phi

badPoints = roots(newNum)
magnitude = abs(badPoints)

Nt=21;
Nr=12;

theta=linspace(0,2*pi,1001);
rho=linspace(0.6192,1,1001);
tvec=linspace(0,2*pi,Nt);
rvec=linspace(0.6192,1,Nr);

temp = (roots(den - num*0.5748));
mag = abs(temp)
ang = angle(temp)

for k=1:length(rvec)
 z=rvec(k)*exp(i*theta);
 w=(z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);

 hold on
 plot(real(w), imag(w))
 hold off
end

for k=1:length(tvec)-1
 z=rho*exp(i*tvec(k));
 w=(z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);
 hold on
 plot(real(w), imag(w))
 hold off
```

```
end

grid on
axis([-1.5 0.1 -1 1])
title('Primary Domain')
% TESTING TO FIND INTERSECTION POINT -> Intersection at 0.5748

% z77 = 0.7742*exp(i*theta);
% w77=(z77.^2-z77.*1.45 + 0.45)./(z77.*1.27-0.73);
% figure(1)
% clf
% plot(real(w77),imag(w77))
%
% z49 = 0.4936*exp(i*theta);
% w49=(z49.^2-z49.*1.45 + 0.45)./(z49.*1.27-0.73);
%
% figure(2)
% clf
% plot(real(w49),imag(w49))


% for N=1:10
%     temp =  0.5748 + N*0.00001
%     val = sprintf('N = %0.5f',temp);
%     z = (temp) * exp(i*theta);
%     w = (z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);
%     plot(real(w), imag(w));
%     title(val);
%     disp(val);
%     disp(w(1));
%     disp(w(2));
%     pause;
% end
```

```
Hp =

     1.27 s - 0.73
  -------------------
  s^2 - 1.45 s + 0.45

Continuous-time transfer function.


zeros =

    0.5748


poles =

    1.0000
    0.4500


Phi =
```

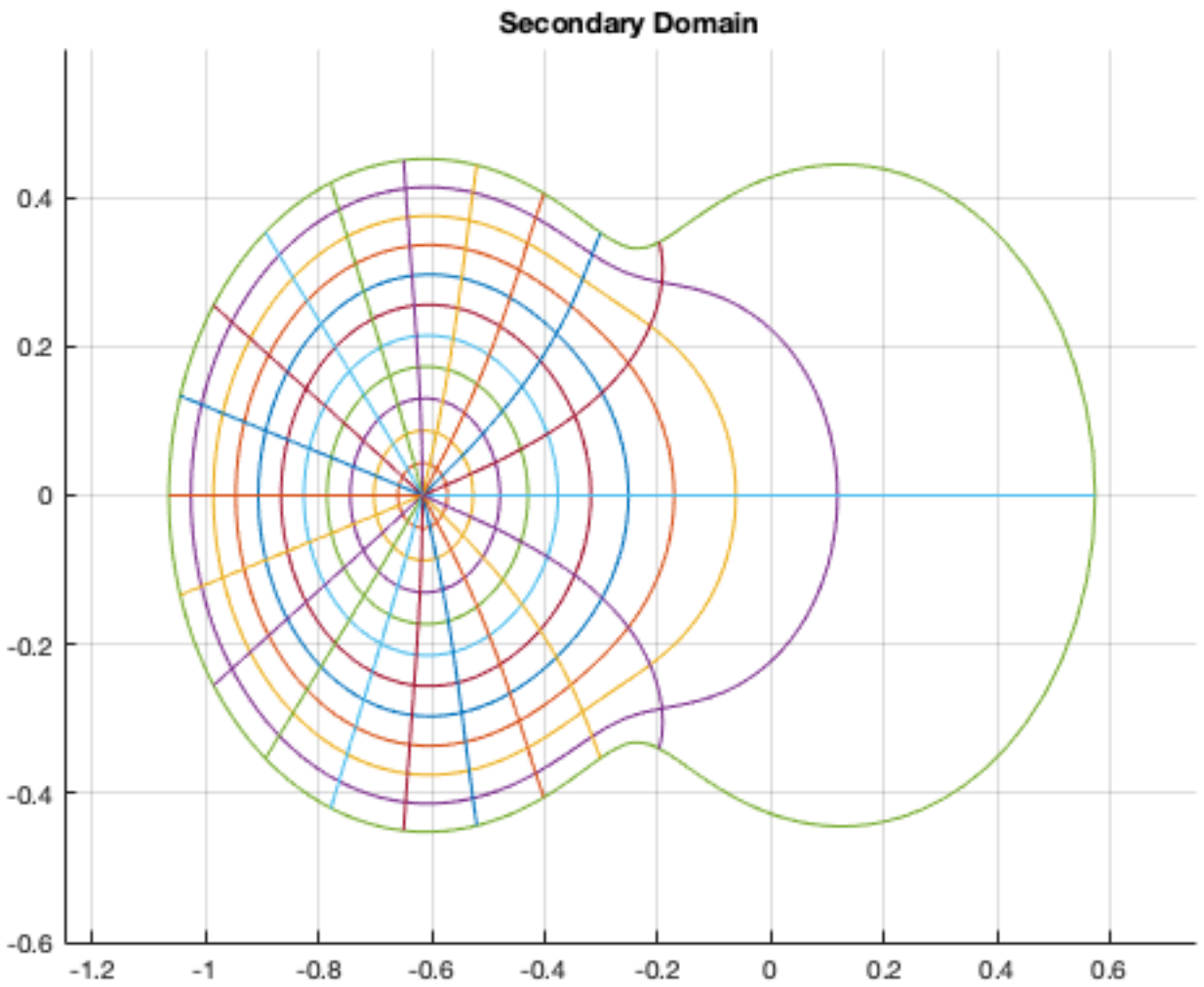
```
  s^2 - 1.45 s + 0.45
  -------------------
     1.27 s - 0.73

Continuous-time transfer function.


newNum =

       12700      -14600        4870


badPoints =

   0.5748 + 0.2304i
   0.5748 - 0.2304i


magnitude =

    0.6192
    0.6192


mag =

    1.6543
    0.5256


ang =

         0
         0
```



PART D

```
figure;

Nt=21;
Nr=12;

theta=linspace(0,2*pi,1001);
rho=linspace(0,0.5256,1001);
tvec=linspace(0,2*pi,Nt);
rvec=linspace(0,0.5256,Nr);

for k=1:length(rvec)
 z=rvec(k)*exp(i*theta);
 w=(z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);

 hold on
 plot(real(w), imag(w))
 hold off
end

for k=1:length(tvec)-1
 z=rho*exp(i*tvec(k));
 w=(z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);
 hold on
 plot(real(w), imag(w))
 hold off
end
```

```
grid on
axis([-1.25 0.75 -0.6 0.6])
title('Secondary Domain')

% TESTING TO FIND INTERSECTION POINT -> Intersection at 0.6192

% for N=1:10
%     temp =  0.619 + N*0.0001
%     val = sprintf('N = %0.5f',temp);
%     z = (temp) * exp(i*theta);
%     w = (z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);
%     plot(real(w), imag(w));
%     title(val);
%     disp(val);
%     disp(w(1));
%     disp(w(2));
%     pause;
% end
```

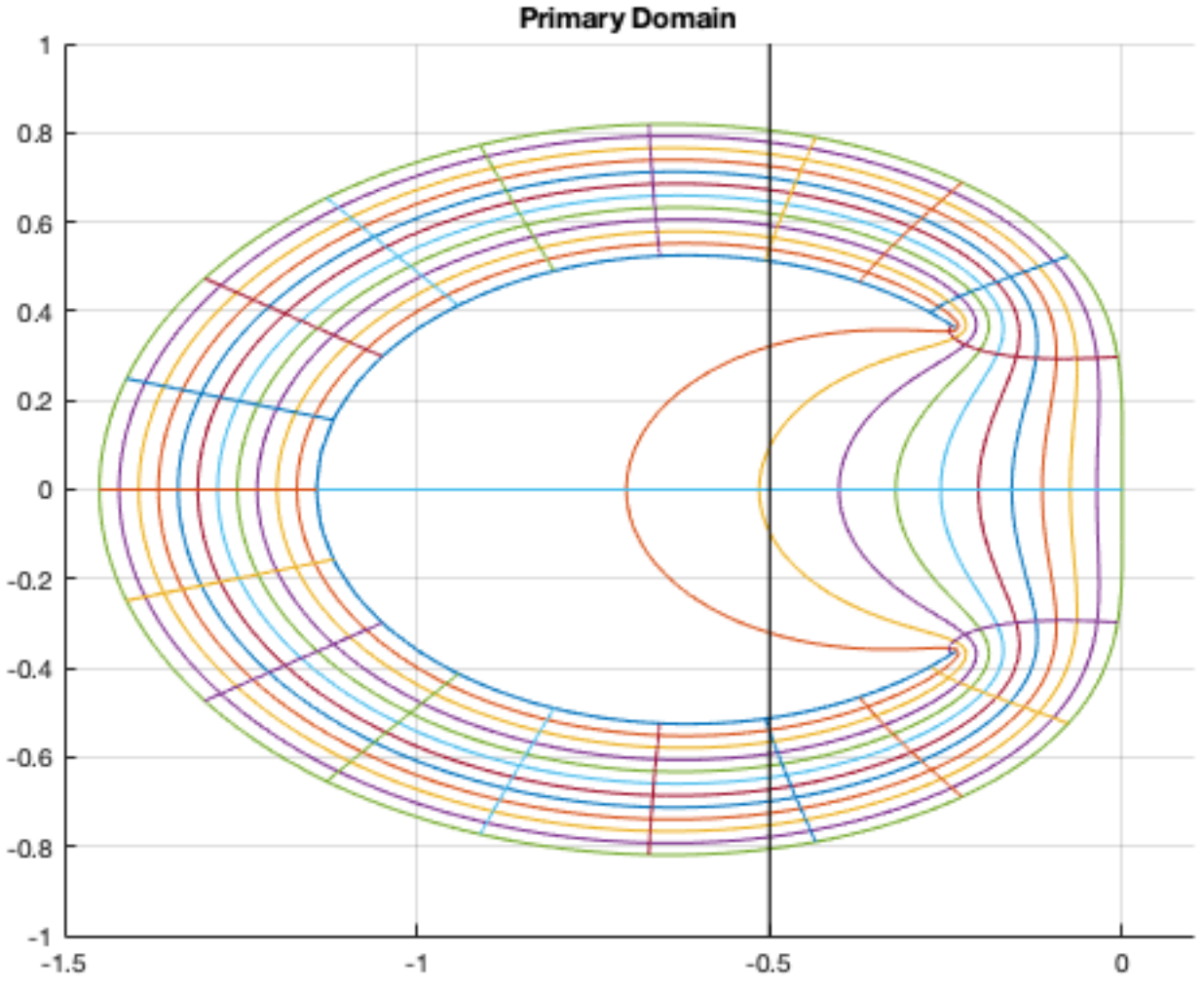


PART E -- Stability Region

```
disp('Stable and Accurate Region is inside of the green outlining edge, to the right of the bl
ack line in Figure 3')
disp('Stable and Inaccurate Region is inside of the green outlining edge, to the left of the b
lack line in Figure 3')
disp('Unstable and Inaccurate Region is outside of the green outlining edge on the primary reg
ion plot')
```

```
figure;

Nt=21;
Nr=12;

theta=linspace(0,2*pi,1001);
rho=linspace(0.6192,1,1001);
tvec=linspace(0,2*pi,Nt);
rvec=linspace(0.6192,1,Nr);

for k=1:length(rvec)
 z=rvec(k)*exp(i*theta);
 w=(z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);

 hold on
 plot(real(w), imag(w))
 hold off
end

for k=1:length(tvec)-1
 z=rho*exp(i*tvec(k));
 w=(z.^2-z.*1.45 + 0.45)./(z.*1.27-0.73);
 hold on
 plot(real(w), imag(w))
 hold off
end

hold on
    plot([-0.5 -0.5], [-1 1], 'black')
hold off


grid on
axis([-1.5 0.1 -1 1])
title('Primary Domain')
```

```
Stable and Accurate Region is inside of the green outlining edge, to the right of the black li
ne in Figure 3
Stable and Inaccurate Region is inside of the green outlining edge, to the left of the black l
ine in Figure 3
Unstable and Inaccurate Region is outside of the green outlining edge on the primary region pl
ot
```



Published with MATLAB® R2018a