Roll: 1715006 Name: Adowan Shahriar LAB: 4

## Objectives:

- To learn about the definition of steady state error.
- To learn about steady-state error constants.
- To be able to find steade state error calculation through MATLAB coding.
- To find the steady state error constants through MATLAB coding.

## Introduction:

steady-state error is defined as the difference between the desired value and the actual value of a system output in the limit as time goes to infinity (i.e when the response of the control system has reached steady-state). steady state error is a property of the input output response for a linear system.

Steady state errors are the differences between the input and the output for a prescribed test input as $t \to \infty$.

Steady-state error can be calculated from a system's closed loop transfer function $T(s)$ or open loop transfer function $G(s)$, for unity feedback systems.

Steady-state error in terms of $T(s)$:

To find $E(s)$, the error between the input, $R(s)$ and the output $C(s)$, we write,

$$E(s) = R(s) - C(s) \quad \text{---} \quad (i)$$

$$\text{But } C(s) = R(s)\,T(s) \quad \text{---} \quad (ii)$$

Substituting eq. (ii) into eq. (i), simplifying and solving for $E(s)$ yields,

$$E(s) = R(s)\,[1 - T(s)] \quad \text{---} \quad (iii)$$

Although Eq. (iii) allows us to solve for $e(t)$ at any time, $t$, we are interested in the final value of the error, $e(\infty)$. Applying the final value theorem, which allows us to use the final value of $e(t)$ without taking the inverse

Laplace transform of E(s) and then letting 't' approach infinity, we obtain

$$e(\infty) = \lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s) \text{ ——— (iv)}$$

substituting Eq. (iii) into eq. (iv) yields

$$e(\infty) = \lim_{s \to \infty} s\, R(s) \left[ 1 - T(s) \right] \text{ ——— (v)}$$

The steady-state error performance specifications are called static error constants. Individually their names are,

Position constants, $K_p$ where, $K_p = \lim_{s \to 0} G(s)$

Velocity constant, $K_v$ where $K_v = \lim_{s \to 0} s\, G(s)$

acceleration constant, $K_a$ where

$$K_a = \lim_{s \to 0} s^2 G(s)$$

Task:

```
clc
clear all
close all
numg1=[1000 8000];deng1=poly([1 16 63]);
'G1(s)='
G1=tf(numg1,deng1)
numg2=6*poly([-9 -17]);deng2=poly([-12 -32 -68]);
'G2(s)='
G2=tf(numg2,deng2)
numh1=13;denh1=1;
'H1(s)='
H1=tf(numh1,denh1)
numh2=1;denh2=[1 7];
'H2(s)='
H2=tf(numh2,denh2)
%Close loop with H1 and form G3
'G3(s)=G2(s)/(1+G2(s)H1(s))'
G3=feedback(G2,H1)
%Form G4=G1G3
'G4(s)=G1(s)G3(s)'
G4=series(G1,G3)
%Form Ge=G4/1+G4H2
'Ge(s)=G4(s)/(1+G4(s)H2(s))'
Ge=feedback(G4,H2)
%Form T(s)=Ge(s)/(1+Ge(s)) to test stability
'T(s)=Ge(s)/(1+Ge(s))'
T=feedback(Ge,1)
'Poles of T(s)'
pole(T)
%Computer response shows that system is stable. Now find
error specs.
Kp=dcgain(Ge)
'sGe(s)='
sGe=tf([1 0],1)*Ge;
```

```
'sGe(s)'
sGe=minreal(sGe)
Kv=dcgain(sGe)
's^2Ge(s)='
s2Ge=tf([1 0],1)*sGe;
's^2Ge(s)'
s2Ge=minreal(s2Ge)
Ka=dcgain(s2Ge)
essstep=100/(1+Kp)
essramp=100/Kv
essparabola=200/Ka
```

Output:

```
ans =

    'G1(s)='


G1 =

        1000 s + 8000
  ---------------------------
  s^3 - 80 s^2 + 1087 s - 1008

Continuous-time transfer function.


ans =

    'G2(s)='
```

G2 =

```
      6 s^2 + 156 s + 918
  ---------------------------
  s^3 + 112 s^2 + 3376 s + 26112
```

Continuous-time transfer function.

ans =

'H1(s)='

H1 =

13

Static gain.

ans =

'H2(s)='

H2 =

```
    1
  -----
  s + 7

Continuous-time transfer function.


ans =

    'G3(s)=G2(s)/(1+G2(s)H1(s))'


G3 =

       6 s^2 + 156 s + 918
  ------------------------------
  s^3 + 190 s^2 + 5404 s + 38046

Continuous-time transfer function.


ans =

    'G4(s)=G1(s)G3(s)'


G4 =
```

```
          6000 s^3 + 204000 s^2 + 2.166e06 s + 7.344e06
  -----------------------------------------------------------------------------
  s^6 + 110 s^5 - 8709 s^4 - 188752 s^3 + 2.639e06 s^2 + 3.591e07 s - 3.835e07
```

Continuous-time transfer function.

ans =

```
    'Ge(s)=G4(s)/(1+G4(s)H2(s))'
```

Ge =

```
          6000 s^4 + 246000 s^3 + 3.594e06 s^2 + 2.251e07 s + 5.141e07

  -------------------------------------------------------------------------------------

  s^7 + 117 s^6 - 7939 s^5 - 249715 s^4 + 1.324e06 s^3 + 5.459e07 s^2 + 2.152e08 s

                                                  - 2.611e08
```

Continuous-time transfer function.

ans =

```
    'T(s)=Ge(s)/(1+Ge(s))'


T =

              6000 s^4 + 246000 s^3 + 3.594e06 s^2 + 2.251e07 s + 5.141e07
  ------------------------------------------------------------------------------------------
  s^7 + 117 s^6 - 7939 s^5 - 243715 s^4 + 1.57e06 s^3 + 5.818e07 s^2 + 2.377e08 s - 2.097e08

Continuous-time transfer function.


ans =

    'Poles of T(s)'


ans =

 -157.3084
   62.3717
   16.9891
  -21.6353
  -11.1624
   -6.9990
    0.7443


Kp =
```

```
   -0.1969


ans =

    'sGe(s)='


ans =

    'sGe(s)'


sGe =

             6000 s^5 + 246000 s^4 + 3.594e06 s^3 + 2.251e07 s^2 + 5.141e07 s
  ------------------------------------------------------------------------------------------
  s^7 + 117 s^6 - 7939 s^5 - 2.497e05 s^4 + 1.324e06 s^3 + 5.459e07 s^2 + 2.152e08 s
                                                                    - 2.611e08

Continuous-time transfer function.
```

Kv =

0

ans =

's^2Ge(s)='

ans =

's^2Ge(s)'

s2Ge =

```
          6000 s^6 + 246000 s^5 + 3.594e06 s^4 + 2.251e07 s^3 + 5.141e07 s^2
  ------------------------------------------------------------------------------------------
  s^7 + 117 s^6 - 7939 s^5 - 2.497e05 s^4 + 1.324e06 s^3 + 5.459e07 s^2 + 2.152e08 s
                                                         - 2.611e08
```

Continuous-time transfer function.

Ka =

0

essstep =

124.5150

essramp =

Inf

essparabola =

Inf

>>

Task:

```
clc
clear all
close all
numg1=[1 9];deng1=poly([0 -6 -12 -14]);
'G1(s)='
G1=tf(numg1,deng1)
numg2=6*poly([-9 -17]);deng2=poly([-12 -32 -68]);
'G2(s)='
```

```
G2=tf(numg2,deng2)
numh1=13;denh1=1;
'H1(s)='
H1=tf(numh1,denh1)
numh2=1;denh2=[1 7];
'H2(s)='
H2=tf(numh2,denh2)
%Close loop with H1 and form G3
'G3(s)=G2(s)/(1+G2(s)H1(s))'
G3=feedback(G2,H1)
%Form G4=G1G3
'G4(s)=G1(s)G3(s)'
G4=series(G1,G3)
%Form Ge=G4/1+G4H2
'Ge(s)=G4(s)/(1+G4(s)H2(s))'
Ge=feedback(G4,H2)
%Form T(s)=Ge(s)/(1+Ge(s)) to test stability
'T(s)=Ge(s)/(1+Ge(s))'
T=feedback(Ge,1)
'Poles of T(s)'
pole(T)
%Computer response shows that system is stable. Now find
error specs.
Kp=dcgain(Ge)
'sGe(s)='
sGe=tf([1 0],1)*Ge;
'sGe(s)'
sGe=minreal(sGe)
Kv=dcgain(sGe)
's^2Ge(s)='
s2Ge=tf([1 0],1)*sGe;
's^2Ge(s)'
s2Ge=minreal(s2Ge)
Ka=dcgain(s2Ge)
essstep=100/(1+Kp)
```

```
essramp=100/Kv
essparabola=200/Ka
```

Output:

```
ans =

    'G1(s)='


G1 =

              s + 9
  ------------------------------
  s^4 + 32 s^3 + 324 s^2 + 1008 s

Continuous-time transfer function.


ans =

    'G2(s)='


G2 =

       6 s^2 + 156 s + 918
  -----------------------------
  s^3 + 112 s^2 + 3376 s + 26112
```

```
Continuous-time transfer function.


ans =

    'H1(s)='


H1 =

  13

Static gain.


ans =

    'H2(s)='


H2 =

    1
  -----
  s + 7

Continuous-time transfer function.
```

```
ans =

    'G3(s)=G2(s)/(1+G2(s)H1(s))'


G3 =

       6 s^2 + 156 s + 918
  ------------------------------
  s^3 + 190 s^2 + 5404 s + 38046

Continuous-time transfer function.


ans =

    'G4(s)=G1(s)G3(s)'


G4 =

                        6 s^3 + 210 s^2 + 2322 s + 8262
  ---------------------------------------------------------------------------------
  s^7 + 222 s^6 + 11808 s^5 + 273542 s^4 + 3.16e06 s^3 + 1.777e07 s^2 + 3.835e07 s

Continuous-time transfer function.
```

```
ans =

    'Ge(s)=G4(s)/(1+G4(s)H2(s))'


Ge =

                    6 s^4 + 252 s^3 + 3792 s^2 + 24516 s + 57834
  ---------------------------------------------------------------------------------------------
  s^8 + 229 s^7 + 13362 s^6 + 356198 s^5 + 5.075e06 s^4 + 3.989e07 s^3 + 1.628e08 s^2
                                                       + 2.685e08 s + 8262

Continuous-time transfer function.


ans =

    'T(s)=Ge(s)/(1+Ge(s))'


T =

                    6 s^4 + 252 s^3 + 3792 s^2 + 24516 s + 57834
```

```
  ------------------------------------------------------------------------------------------

  s^8 + 229 s^7 + 13362 s^6 + 356198 s^5 + 5.075e06 s^4 + 3.989e07 s^3 + 1.628e08 s^2

                                                    + 2.685e08 s + 66096


Continuous-time transfer function.


ans =

    'Poles of T(s)'


ans =

 -157.1538
  -21.6791
  -14.0006
  -11.9987
  -11.1678
   -7.0001
   -5.9997
   -0.0002


Kp =
```

```
     7


ans =

    'sGe(s)='


ans =

    'sGe(s)'


sGe =

                6 s^5 + 252 s^4 + 3792 s^3 + 2.452e04 s^2 + 5.783e04 s
  ------------------------------------------------------------------------------------------
  s^8 + 229 s^7 + 1.336e04 s^6 + 3.562e05 s^5 + 5.075e06 s^4 + 3.989e07 s^3 + 1.628e08 s^2

                                                 + 2.685e08 s + 8262


Continuous-time transfer function.
```

```
Kv =

     0


ans =

    's^2Ge(s)='


ans =

    's^2Ge(s)'


s2Ge =

 
               6 s^6 + 252 s^5 + 3792 s^4 + 2.452e04 s^3 + 5.783e04 s^2
  ---------------------------------------------------------------------------------------------
  s^8 + 229 s^7 + 1.336e04 s^6 + 3.562e05 s^5 + 5.075e06 s^4 + 3.989e07 s^3 + 1.628e08 s^2
 
                                              + 2.685e08 s + 8262
 

Continuous-time transfer function.
```

```
Ka =

     0


essstep =

   12.5000


essramp =

   Inf


essparabola =

   Inf

>>
```

## Discussion :

In this lab we mainly focused on calculating parametors of closed loop system and according to given equations. We calculated pole locations of systems assuming system types and types of feed-back. And all by MATLAB coding. We coded to calculate pole locations of general transfer function and also closed loop system transfer functions. We calculated ~~stabily~~ stability of unity feedback system and steady state errors. We also calculated steady-state error constants.

## Conclusion :

We learned to calculate pole locations of various types of and feedback system's transfer function. We learned to evaluate steady state error and steady-state error constants. Overall through this pandemic situation is already harsh on us, but we conducted this lab with the help of our course teacher.