ELL225 (Control Engineering)

RIDERLESS BICYCLE CONTROL

Submitted by :

Amitesh Gupta(2021EE30724)

Advait Rajesh Ninawe(2021EE30714)

Ayush Gupta(2021EE30036)

Under the guidance of

Dr. Subashish Datta

Department of Electrical Engineering
Indian Institute of Technology Delhi

Introduction:

A riderless bicycle is a type of autonomous vehicle that can balance and maneuver itself without the need for a human rider. This technology has many potential applications, including in transportation, logistics, and robotics. Control engineering plays a crucial role in the design and development of riderless bicycles, as it involves the use of control systems to regulate the bike's stability, movement, and speed. By harnessing the power of control engineering, riderless bicycles can provide an efficient and innovative solution to various challenges in today's world. In this project report, we will explore the principles of control engineering and how they are applied to the design and implementation of a riderless bicycle.

Modelling:

We considered a bicycle consisting of four rigid bodies, namely, the rear frame, the front fork with the handlebar, and the rear and front knife-edge wheels. The four bodies are interconnected by revolute hinges, and, in the reference configuration, they are all symmetric relative to the bicycle longitudinal axis. The contact between the stiff nonskipping wheels and the flat level surface is modeled by holonomic constraints in the normal direction as well as by nonholonomic constraints in the longitudinal and lateral direction.

This model adequately describes the main dynamics of the bicycle as it considers the three degree of freedom namely the roll angle, Φ(t),the steering angle δ(t), and the speed v(t). The linearized equation of motions are two coupled second order ordinary differential equations written in matrix form as

$$M\ddot{q}(t) + v(t)C_1\dot{q}(t) + (K_0 + v(t)^2K_2)q(t) = f(t),$$

with

$$q(t) = \begin{bmatrix} \phi(t) \\ \delta(t) \end{bmatrix}, \ f(t) = \begin{bmatrix} T_\phi(t) \\ T_\delta(t) \end{bmatrix},$$

where Tϕ(t) is an exogenous roll-torque disturbance and Tδ(t) is the steering torque provided by the actuator on the handlebar axis. The remaining quantities are the symmetric mass matrix M, the speed-dependent damping matrix v(t)C1, and the stiffness matrix, which is the sum of a constant symmetric part K0 and a quadratically speed-dependent part v2K2.

The linearized ordinary differential equation is rewritten in state-space form choosing the roll angle ϕ(t), the steering angle δ(t), and their derivatives ϕ˙(t) and δ˙(t), respectively, as state variables. The control input u(t) is the torque Tδ(t) applied to the handlebar axis. The measured output y(t) includes all of the state variables. On the basis of the above considerations, the state-space equations are

$$\dot{x}(t) = A(v(t))x(t) + Bu(t), \qquad (2)$$

$$y(t) = Cx(t) + Du(t), \qquad (3)$$

where

$$x(t) = [\phi(t)\delta(t)\phi(t)\delta(t)]^{\mathrm{T}}, \;\; u(t) = T_\delta(t), y(t) = x(t).$$

The entries of the matrices A(v(t)), B, C, and D depend on the geometry as well as the physical parameters of the bicycle. In particular, the numerical values of the matrices A(v(t)), B, C, and D for the prototype are:

$$A(v) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 13.67 & 0.225 - 1.319v^2(t) & -0.164v(t) & -0.552v(t) \\ 4.857 & 10.81 - 1.125v^2(t) & 3.621v(t) & -2.388v(t) \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 0 \\ -0.339 \\ 7.457 \end{bmatrix}, \;\; C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \;\; D = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Mathematical Analysis:

The MATLAB code used to analyses the above describe model and finding the output parameters:

```
syms s t
syms v;
% creating matrix depending on velocity
A = @(v)[0 0 1 0; 0 0 0 1; 13.67 0.225-1.319*(v^2) -0.164*v -0.552*v; 4.857 10.81-
1.125*(v^2) 3.621*v -2.388*v];
B = [0; 0; -0.339; 7.457];
C = [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 0 1];
D = [0; 0; 0; 0];

% Using the control system tool bar
% for v = 0 mps
system1 = ss(A(0),B,C,D);
transferfunction1 = tf(system1);
% the transferfunction wrt to roll angle is transferfunction1(1)
roll_tranfer1 = transferfunction1(1);


% for v = 0.5 mps
system2 = ss(A(0.5),B,C,D);
transferfunction2 = tf(system2);
% the transferfunction wrt to roll angle is transferfunction2(1)
roll_tranfer2 = transferfunction2(1);


% for v = 5 mps
system3 = ss(A(5),B,C,D);
transferfunction3 = tf(system3);
% the transferfunction wrt to roll angle is transferfunction2(1)
roll_tranfer3 = transferfunction3(1);



% Finding the zeroes
% --
% corresponding to v1 = 0 mps
z1 = zero(transferfunction1(1));
% corresponding to v1 = 0.5 mps
z2 = zero(transferfunction2(1));
% corresponding to v1 = 5 mps
z3 = zero(transferfunction3(1));

% Finding the poles
% --
% corresponding to v1 = 0 mps
p1 = pole(transferfunction1(1));
% corresponding to v1 = 0.5 mps
p2 = pole(transferfunction2(1));
% corresponding to v1 = 5 mps
p3 = pole(transferfunction3(1));


% Finding eigen value of matrix A
%--
```

```
% For v1 = 0 mps
e1 = eig(A(0));
% For v1 = 0.5 mps
e2 = eig(A(0.5));
% For v1 = 5 mps
e3 = eig(A(5));

% Finding the response

% 1. Step response
% For v1 = 0 mps
step(roll_tranfer1);
% For v1 = 0.5 mps
step(roll_tranfer2);
% For v1 = 5 mps
step(roll_tranfer3);

% 2. Zero input response with x(0) = [1;1;1;1]

x_in = [1; 1; 1; 1];
% u(t) = 0 gives
% x(t) = e^(A*t)*x_in
% y = C*x(t) = C*e^(A*t)*x_in

% for v = 0 mps
r1=t*A(0);
y1 = C*exp(r1)*x_in;

% plot y1

fplot(y1);

% for v = 0.5 mps
r2=t*A(0.5);
y2 = C*exp(r2)*x_in;

% plot y2

fplot(y2);

% for v = 5 mps
r3=t*A(5);
y3 = C*exp(r3)*x_in;

% plot y

fplot(y3);
```

Outputs:

Part(a): After the above analysis we got the following outputs for the above system

1. **At v = 0 mps**.

roll_tranfer1:

```
                   -0.339 s^2 + 5.342
  -----------------------------------------------------
  s^4 + 3.553e-15 s^3 - 24.48 s^2 - 2.842e-14 s + 146.7
```

Zeroes:

```
z1 =

    3.9698
   -3.9698
```

Poles:

```
p1 =

   -3.7432
   -3.2355
    3.7432
    3.2355
```

Eigen values of matrix A:

```
e1 =

 -3.7432
 -3.2355
  3.7432
  3.2355
```

2. A v = 0.5 mps

roll_tranfer2:

```
          -0.339 s^2 - 2.463 s + 2.788
  --------------------------------------------
 s^4 + 1.276 s^3 - 23.6 s^2 - 15.66 s + 144.4
```

Zeroes:

```
z2 =
 -8.2608
  0.9956
```

Poles:

```
p2 =
-3.7997 + 0.1830i
-3.7997 - 0.1830i
 3.2844 + 0.0000i
 3.0390 + 0.0000i
```

Eigen Values for matrix A:

```
e2 =
  3.2844 + 0.0000i
  3.0390 + 0.0000i
 -3.7997 + 0.1830i
 -3.7997 - 0.1830i
```

3. At v = 5 mps

roll_tranfer3:

```
      -0.339 s^2 - 24.63 s - 250.1
  -----------------------------------------
  s^4 + 12.76 s^3 + 63.41 s^2 + 457.3 s - 77.63
```

Zeroes:

```
z3 =
 -60.4476
 -12.2043
```

Poles:

```
p3 =

 -10.8598 + 0.0000i
 -1.0330 + 6.4842i
 -1.0330 - 6.4842i
  0.1658 + 0.0000i
```

Eigen Values for matrix A:

```
e3 =
 -10.8598 + 0.0000i
 -1.0330 + 6.4842i
 -1.0330 - 6.4842i
  0.1658 + 0.0000i
```

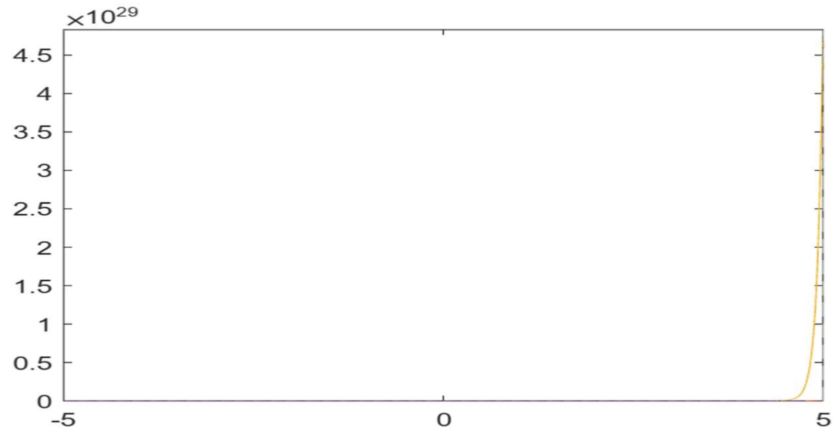
Part(b): The time response as follows for the corresponding inputs:

1. **Zero input with non-zero initial condition**
   Let non zero input be x (0) = [1;1;1;1]

For v = 0 mps

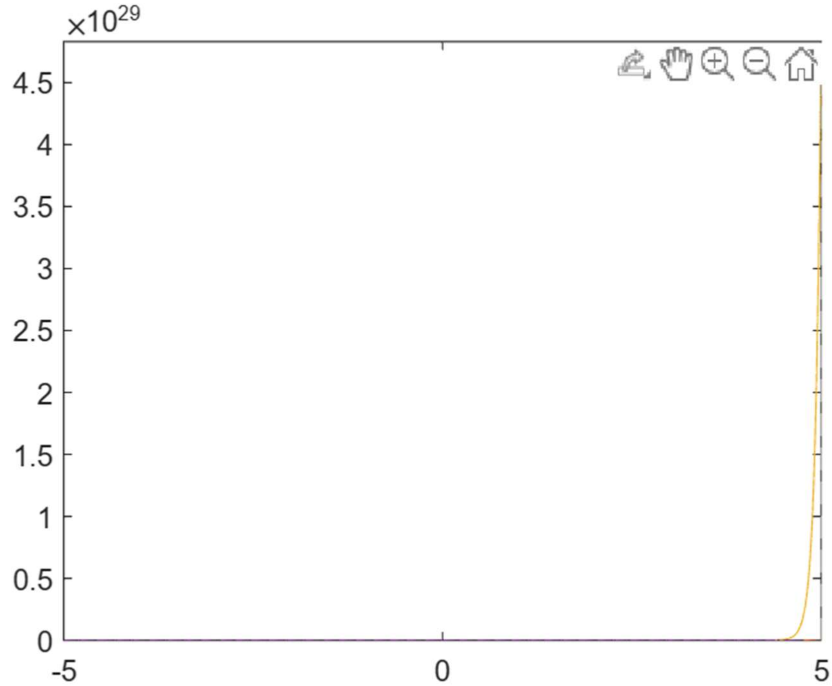
```
y1 =

                                  exp(t) + 3
                                  exp(t) + 3
      exp((9*t)/40) + exp((1367*t)/100) + 2
exp((1081*t)/100) + exp((4857*t)/1000) + 2
```

```
The plot is:
```



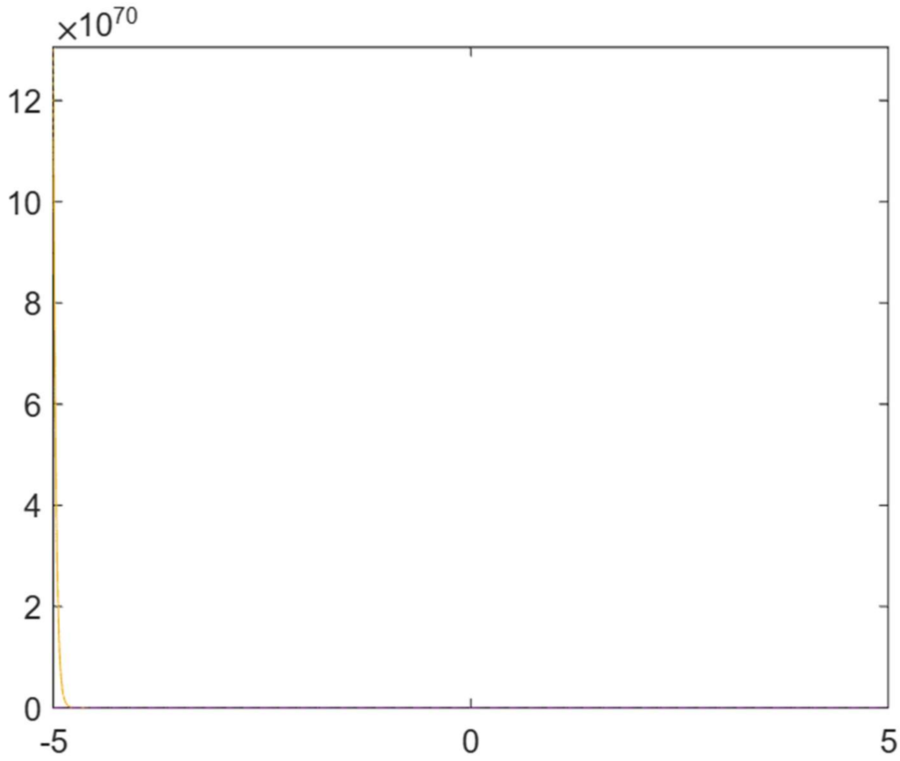
For v = 0.5 mps

```
y2 =


exp(t) + 3

exp(t) + 3
   exp(-(69*t)/250) + exp(-(41*t)/500) + exp((1367*t)/100) + exp(-
(419*t)/4000)
exp(-(597*t)/500) + exp((3621*t)/2000) + exp((4857*t)/1000) +
exp((8423*t)/800)


The plot is:
```



For v = 5 mps

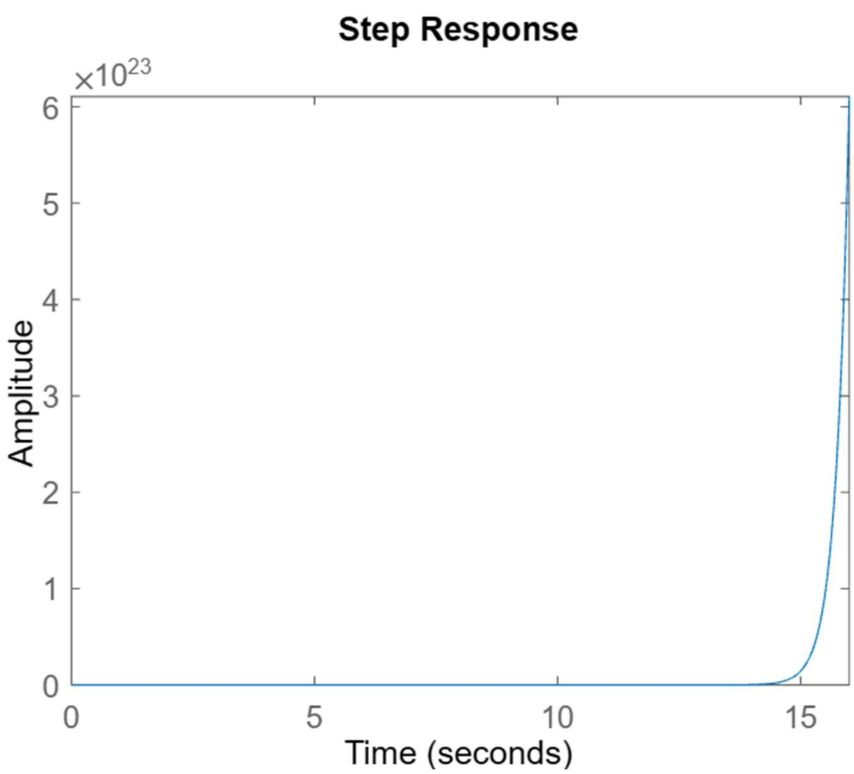
```
y3 =


exp(t) + 3

exp(t) + 3
       exp(-(41*t)/50) + exp(-(69*t)/25) + exp(-(131*t)/4) +
exp((1367*t)/100)
exp(-(597*t)/50) + exp(-(3463*t)/200) + exp((3621*t)/200) +
exp((4857*t)/1000)

The plot is:
```



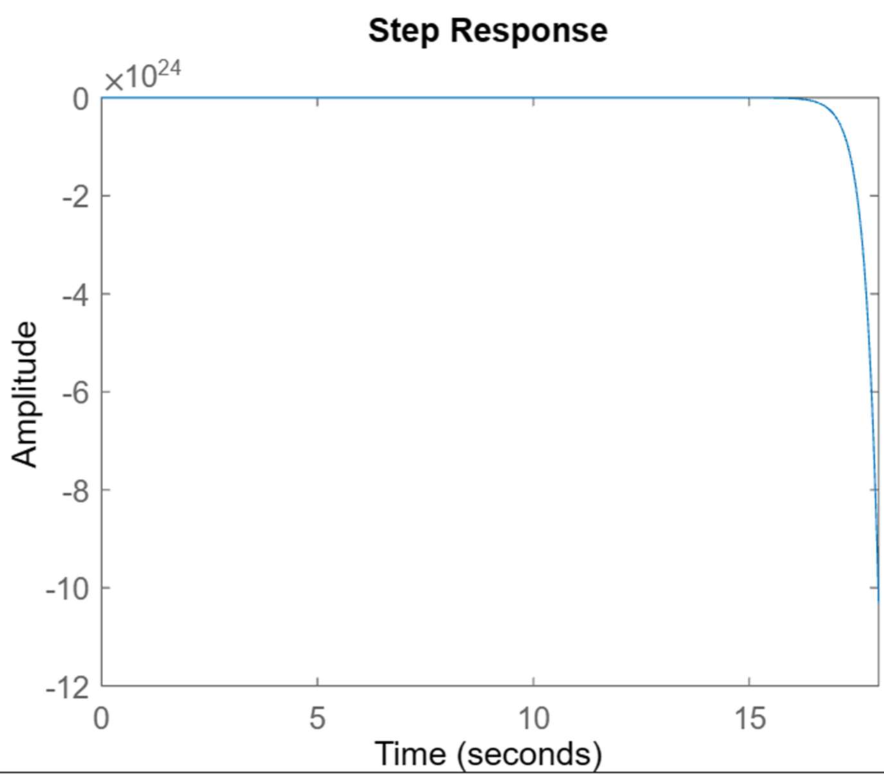
is:

2. Unit step response

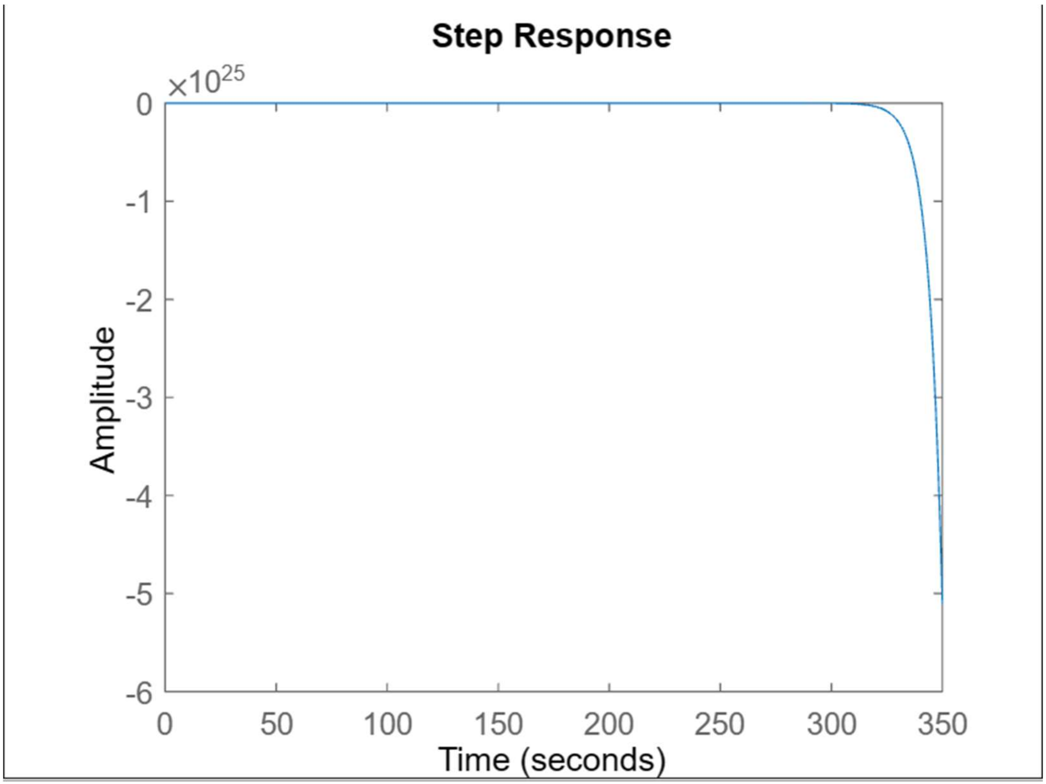
For v = 0 mps


Step Response
×10^23
6
5
4
3
2
1
0
Amplitude
0
5
10
15
Time (seconds)

For v = 0.5 mps


Step Response
×10^24
0
-2
-4
-6
-8
-10
-12
Amplitude
0
5
10
15
Time (seconds)

For v = 5 mps



Conclusion:

We have successfully modeled the riderless bicycle in control system and have analyzed its dynamic behavior. Our findings related to the transfer function, eigenvalues, and step response can provide valuable insights into the stability and performance of the system. These results only provides the limited understanding of the behaviors.