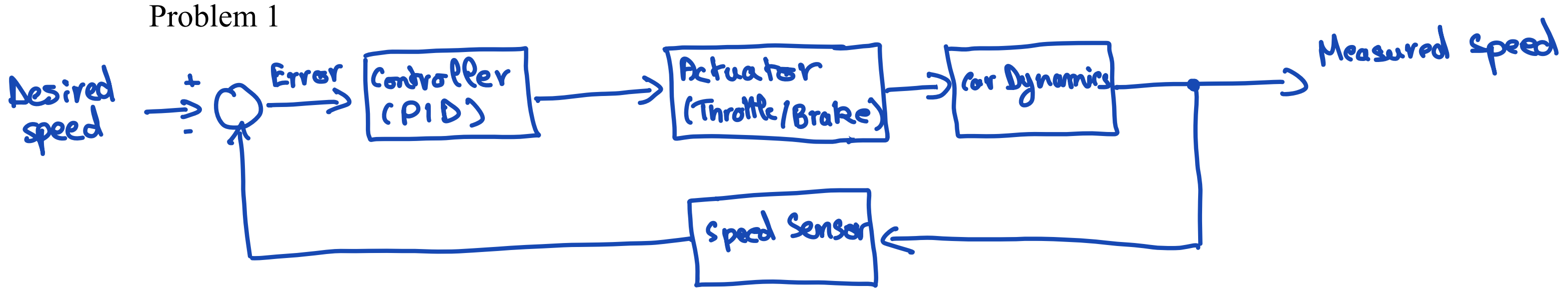
Mamadou Diao Kaba (27070179)

Problem 1



2.

To fully automate this control system in an autonomous self-driving car, additional components are integrated, such as advanced sensors (e.g., LiDAR, cameras), complex algorithms for decision making, and machine learning models that predict and adapt to changing conditions. These systems work together to maintain the desired speed while accounting for road conditions, traffic, and obstacles, enabling the car to drive autonomously.

Problem 2

Closed-loop control is preferred over open-loop control for most physical systems because it can automatically correct any deviation from the desired performance by continuously monitoring and adjusting the system based on feedback. This makes closed-loop systems more robust and accurate in handling disturbances and uncertainties, ensuring stability and reliability in dynamic environments.

Problem 3

Assuming that physical systems have a linear model is acceptable when the system operates within a range where the relationship between input and output is approximately linear. This simplification is often used because linear models are mathematically easier to analyze and design, and they provide sufficiently accurate approximations for control purposes in many practical scenarios. Linear models are especially useful during the initial design phase or for systems where nonlinearities are minimal or can be effectively compensated.

Problem 4

$$\ddot{x} = \frac{-m^2L^2 g\cos\theta\sin\theta + mL^2(mL\dot{\theta}^2\sin\theta - b\dot{x}) + mL^2 F}{mL^2(M + m(1-\cos^2\theta))}$$

$$\ddot{\theta} = \frac{(m+M)mgL\sin\theta - mL\cos\theta(mL\dot{\theta}^2\sin\theta - b\dot{x}) + mL\cos\theta F}{mL^2(M + m(1-\cos^2\theta))}$$

we express $\ddot{x}$ and $\ddot{\theta}$ in terms of state variables:

$$\dot{z}_1 = z_2$$

$$\dot{z}_2 = \frac{-m^2L^2 g\cos z_3\sin z_3 + mL^2(mL^2(mLz_4^2\sin z_3 - bz_2) + mL^2 u}{mL^2(M + m(1-\cos^2 z_3))}$$

$$\dot{z}_3 = z_4$$

$$\dot{z}_4 = \frac{(m+M)mgL\sin z_3 - mL\cos z_3(mLz_4^2\sin z_3 - bz_2) + mL\cos z_3 u}{mL^2(M + m(1-\cos^2 z_3))}$$

The output $y = x$ can be written as:

$$y = z_1$$

Thus, the nonlinear state-space representation is:

$$\dot{z}_1 = z_2$$

$$\dot{z}_2 = \frac{-m^2L^2 g\cos z_3\sin z_3 + mL^2(mLz_4^2\sin z_3 - bz_2) + mL^2 u}{mL^2(M + m(1-\cos^2 z_3))}$$

$$\dot{z}_3 = z_4$$

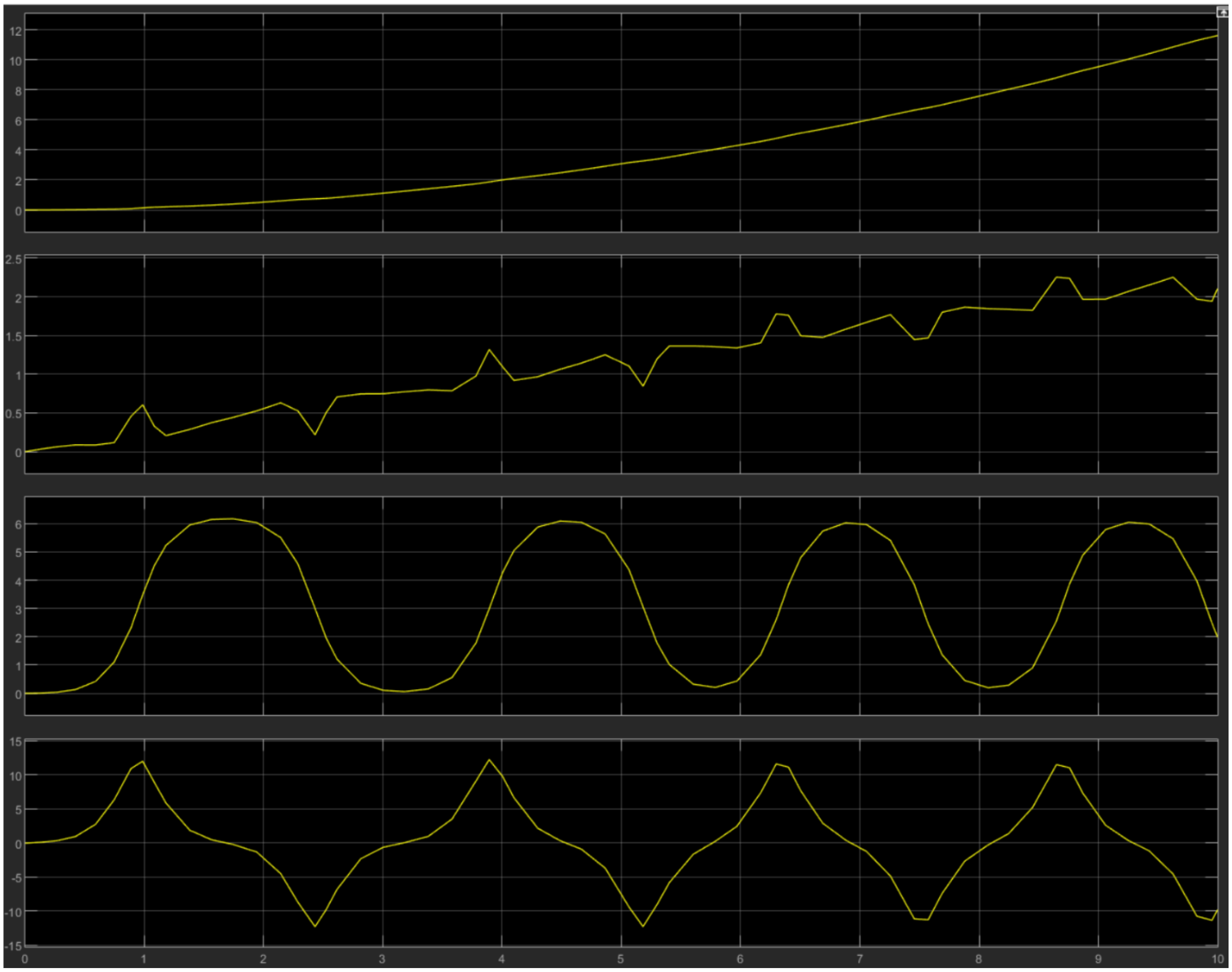
$$\dot{z}_4 = \frac{(m+M)mgL\sin z_3 - mL\cos z_3(mLz_4^2\sin z_3 - bz_2) + mL\cos z_3 u}{mL^2(M + m(1-\cos^2 z_3))}$$

and the output equation is:

$$y = z_1$$

Problem 4

b)



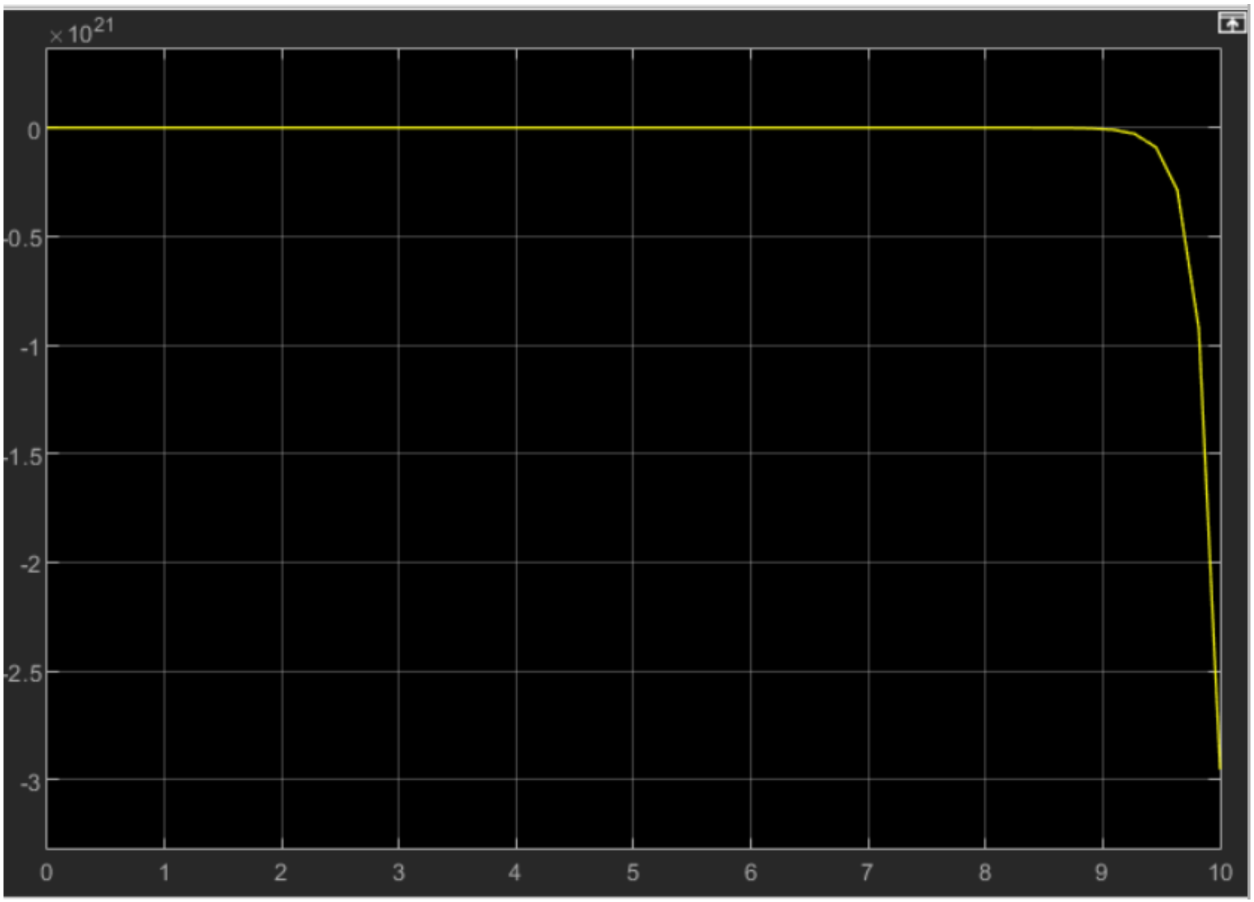
⍰ **Plot 1 (Cart Position x(t)):** This plot shows the position of the cart over time. It appears to be increasing steadily, indicating that the cart is moving to the right continuously. This might be due to a constant force being applied.

⍰ **Plot 2 (Cart Velocity $\dot{x}$(t)):** This plot shows the velocity of the cart. The velocity appears to be oscillating but generally increasing. This suggests that the cart is accelerating over time with some fluctuations.

⍰ **Plot 3 (Pendulum Angle θ(t)):** This plot shows the angle of the pendulum from the vertical. It seems to exhibit oscillatory behavior with increasing amplitude. This indicates that the pendulum is swinging more wildly over time.

□ **Plot 4 (Pendulum Angular Velocity θ˙(t)):** This plot shows the angular velocity of the pendulum. The oscillatory nature here indicates the pendulum is swinging back and forth with increasing speed.

c)



The simulation shows a drastic change towards the end of the simulation time, which is quite different from the nonlinear case.

d)

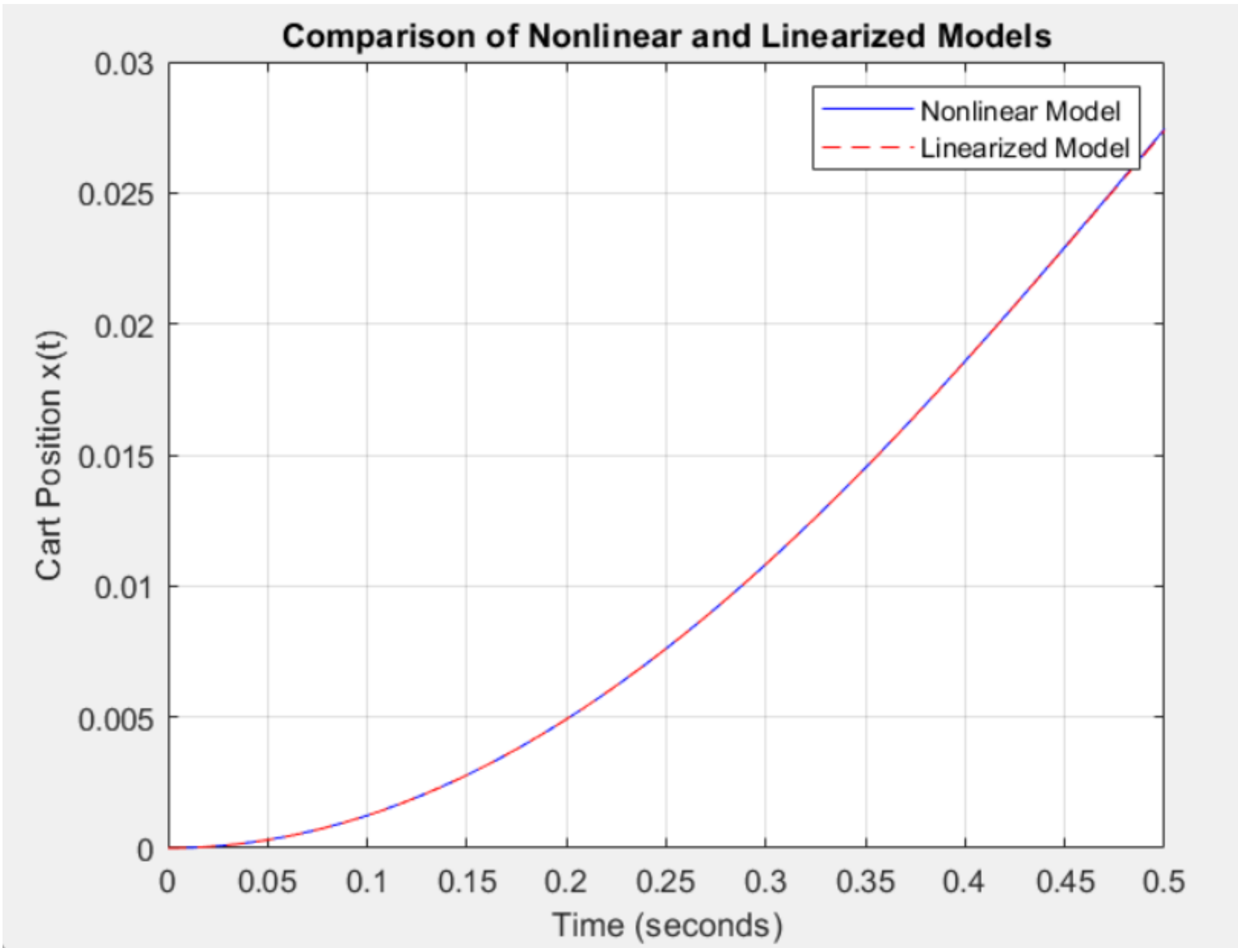
```
>> % Load the linearized model matrices
load('linearized_model.mat');

% Compute the transfer function
[num, den] = ss2tf(A, B, C, D);

% Display the transfer function
G = tf(num, den);
disp('Transfer function G(s):');
disp(G);
Transfer function G(s):
  tf with properties:

      Numerator: {[0 0 0.2500 6.6613e-16 -11.2406]}
    Denominator: {[1 0.0250 -40.8750 -0.9197 0]}
       Variable: 's'
        IODelay: 0
     InputDelay: 0
    OutputDelay: 0
      InputName: {''}
      InputUnit: {''}
     InputGroup: [1×1 struct]
     OutputName: {''}
     OutputUnit: {''}
    OutputGroup: [1×1 struct]
          Notes: [0×1 string]
       UserData: []
           Name: ''
             Ts: 0
       TimeUnit: 'seconds'
   SamplingGrid: [1×1 struct]
```

e)



The plot shows the cart position x(t) for both the nonlinear and linearized models. The responses are very similar, which is expected for a linearized model around an equilibrium point when subjected to a small input.

Problem 5

a) $G(s) = \dfrac{1.6667(s+0.1)}{(s-6.388)(s+6.398)(s+0.04)}$ $\qquad U(s) = \dfrac{1}{s}$ because $u(t)$: unit step

So $Y(s) = G(s)U(s) = \dfrac{1.6667(s+0.1)}{s(s-6.388)(s+6.398)(s+0.04)}$

$$\frac{1.6667(s+0.1)}{s(s-6.388)(s+6.398)(s+0.04)} = \frac{A}{s} + \frac{B}{s-6.388} + \frac{C}{s+6.398} + \frac{D}{s+0.04}$$

$$1.6667(s+0.1) = A(s-6.388)(s+6.398)(s+0.04) + B(s)(s+6.398)(s+0.04) + C(s)(s-6.388)(s+0.04) + D(s)(s-6.388)(s+6.398)$$

To find A, $s=0$

$$1.6667(0.1) = A(-6.388)(6.398)(0.04)$$

$A = -0.1017$

To find B, $s = 6.388$

$$1.6667(6.388+0.1) = B(6.388)(6.388+6.398)(6.388+0.04)$$

$$B = \frac{1.6667 \times 6.488}{6.388 \times 12.786 \times 6.428}$$

$B = 0.0207$

To find C, $s = -6.398$

$$1.6667(-6.398+0.1) = C(-6.398)(-6.398-6.388)(-6.398+0.04)$$

$$C = \frac{1.6667(-6.298)}{-6.398(-12.786)(-6.358)}$$

$C = -0.0201$

To find D, $s = -0.04$

$$1.6667)(-0.04+0.1) = D(-0.04)(-0.04-6.388)(-0.04+6.398)$$

$$D = \frac{1.6667(0.06)}{(-0.04)(-6.428)(6.358)}$$

$$= 0.06111$$

$$\frac{1.6667(s+0.1)}{s(s-6.388)(s+6.398)(s+0.04)} = \frac{-0.1017}{s} + \frac{0.0207}{s-6.388} + \frac{0.0201}{s+6.398} + \frac{0.0611}{s+0.04}$$

$$L^{-1}\left(\frac{-0.1017}{s}\right) = -0.1017$$

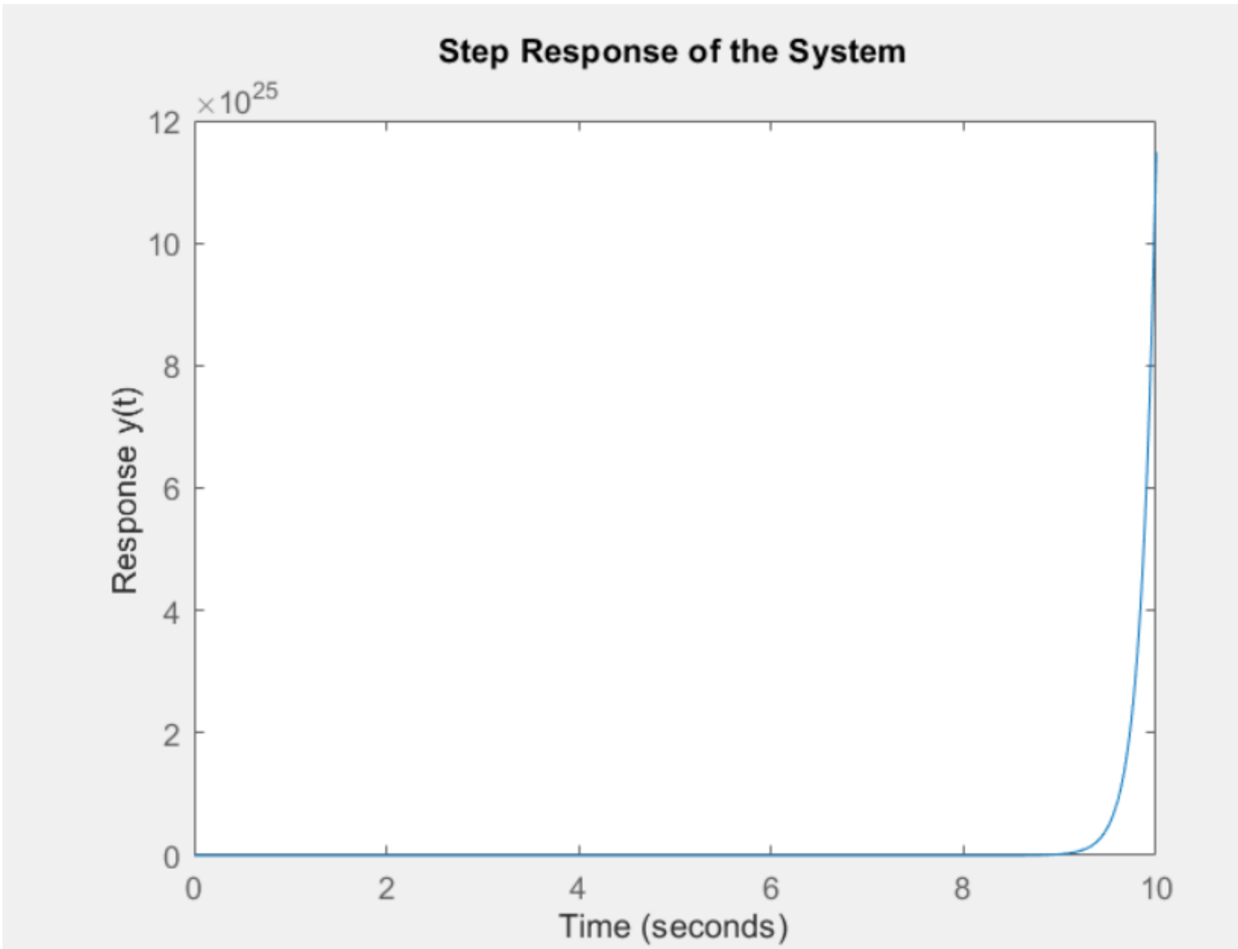
$$L^{-1}\left(\frac{0.0207}{s-6.388}\right) = 0.0207e^{6.388t}$$

$$L^{-1}\left(\frac{-0.0201}{s+6.398}\right) = -0.0201e^{-6.398t}$$

$$L^{-1}\left(\frac{0.0611}{s+0.04}\right) = 0.0611e^{-0.04t}$$

we get : $\boxed{y(t) = -0.1017 + 0.0207e^{6.388t} - 0.0201e^{-6.388t} + 0.0611e^{-0.04t}}$

Problem 5



Step Response of the System
×10^25
12
10
8
6
4
2
0
Response y(t)
0
2
4
6
8
10
Time (seconds)