16-642: Manipulation Estimation and Control
Problem Set 1

Abhinandan Vellanki

1 Mass-Spring-Damper System

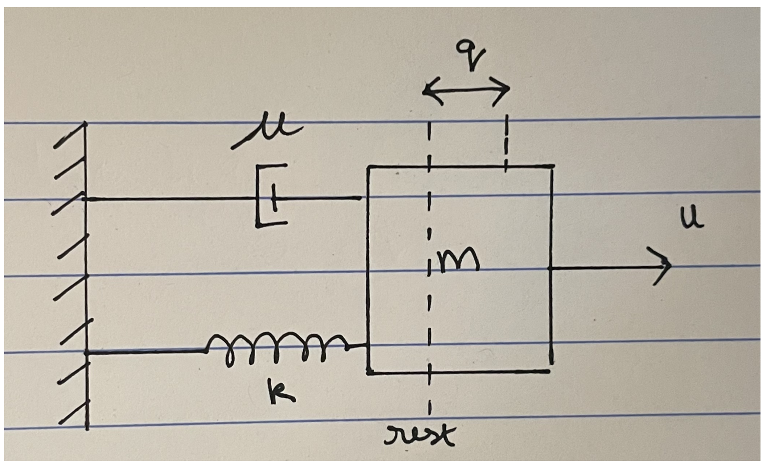

Figure 1: Mass-Spring-Damper System

1.a

Input force $(F) = u$
According to Hooke's Law, spring force $(F_s) = -kq$
Damper force $(F_d) = -\mu\dot{q}$

According to Newton's Second Law,

$$F_{total} = ma$$
$$\implies u - kq - \mu\dot{q} = m\ddot{q}$$

Rearranging terms, the second order ODE that describes the system is:

$$u = kq + \mu\dot{q} + m\ddot{q} \tag{1}$$

1.b

Defining the state as

$$x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \tag{2}$$

The state equation is given by:

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} \tag{3}$$

Solving for $\ddot{q}$ from the ODE derived in (1),

$$\ddot{q} = \frac{u}{m} - \frac{kq}{m} - \frac{\mu\dot{q}}{m}$$

Substituting back into the state equation (3),

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \frac{u}{m} - \frac{kq}{m} - \frac{\mu\dot{q}}{m} \end{bmatrix}$$

This can be rewritten as follows:

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\mu}{m} \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u$$

Substituting (2) into the equation, it can be written as:

$$\implies \dot{x} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\mu}{m} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u \tag{4}$$

For the output equation, assuming y = q,

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix}$$

$$\implies y = \begin{bmatrix} 1 & 0 \end{bmatrix} x \tag{5}$$

The state equation (4) and output equation (5) together comprise the linear state space model for this system, where

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\mu}{m} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

1.c

To discuss stability of this system, it must be examined without any input force i.e., u = 0. The state equation for the system hence becomes:

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\mu}{m} \end{bmatrix} x$$

Since A is invertible, x_e = 0 is the only equilibrium point.
Computing the eigenvalues λ_i of the dynamics matrix A by setting $det(A - \lambda I) = 0$:

$$\begin{aligned} (0-\lambda)\left(\frac{-\mu}{m} - \lambda\right) - \left(-\frac{k}{m}\right) &= 0 \\ \implies \frac{\lambda\mu}{m} - \lambda^2 + \frac{k}{m} &= 0 \end{aligned}$$

From quadratic formula, the eigenvalues are:

$$\lambda_i = -\frac{\mu}{2m} \pm \sqrt{\left(\frac{\mu}{2m}\right)^2 - \left(\frac{k}{m}\right)}$$

Applying the following physical constraints:

- m > 0 (mass must be positive)
- k > 0 (spring constant must be positive)
- $\mu > 0$ (damping constant must be positive)

This makes $-\frac{\mu}{2m}$ always negative.

Stability:
Since A has no repeated eigenvalues with zero real part, the system is **stable** only if the real parts of the eigenvalues are < 0. Which means,

$$\begin{aligned} \left(\frac{\mu}{2m}\right)^2 - \left(\frac{k}{m}\right) &\leq 0 \\ \implies \mu^2 - 4mk &\leq 0 \\ \\ \implies 2\sqrt{mk} &\geq \mu \end{aligned}$$

Alternatively, the system is **unstable** if $2\sqrt{mk} < \mu$.

1.d

Controllability From the dynamics matrix A and control matrix B for the mass-spring-damper system, we can write the controllability matrix Q:

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\mu}{m} \end{bmatrix}; B = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}$$

$$Q = \begin{bmatrix} 0 & \frac{1}{m} \\ \frac{1}{m} & \frac{\mu}{m^2} \end{bmatrix}$$

If Q needs to be full rank, its determinant must be non zero:

$$det(Q) \neq 0$$
$$\implies -\frac{1}{m^2} \neq 0$$

This essentially means that as long as the mass is non-zero, the system is controllable, regardless of the values of the spring constant k and damping coefficient μ.

1.e

The output of the unforced system sampled every 0.1s is:

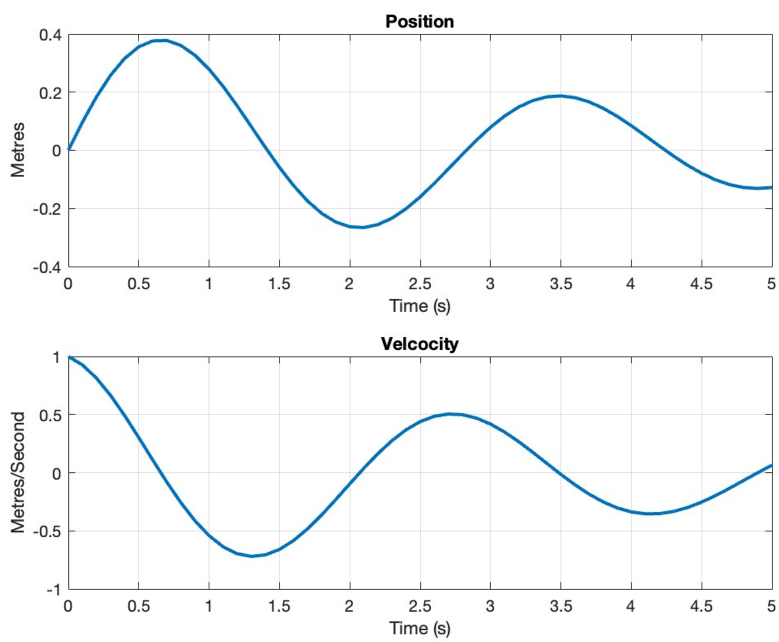

Figure 2: Open Loop Unforced Mass Spring Damper System

1.f

The K matrix obtained using eigenvalue placement in MATLAB is:

$$K = \begin{bmatrix} -3.0 & 1.5 \end{bmatrix}$$

Once we introduce feedback $-Kx$ into the system, the new dynamic matrix is $(A - BK)$. Since we want this new closed-loop system to be asymptotically stable, we choose eigenvalues λ_i such that $Re(\lambda(i) < 0$.

1.g

The output of the closed-loop unforced system sampled every 0.1s is:

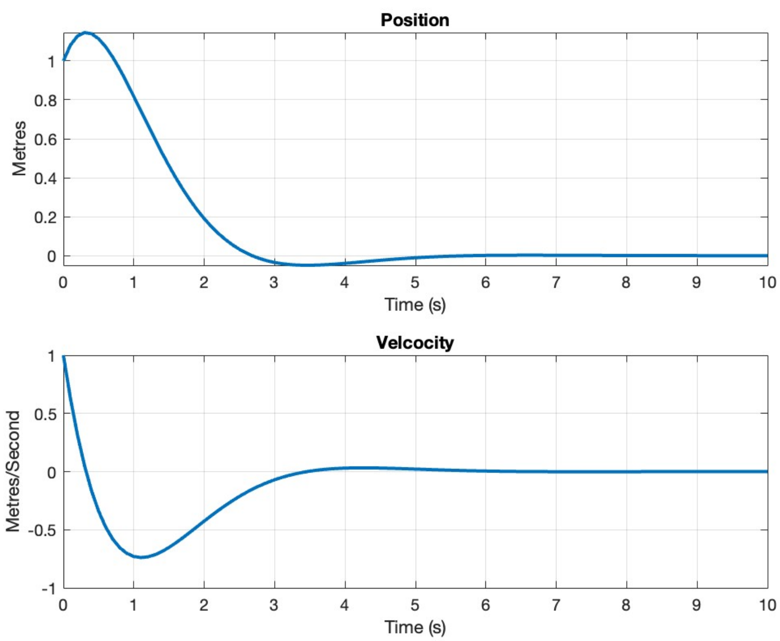

Figure 3: Closed-Loop Unforced Mass Spring Damper System

2 Pendulum On A Cart System

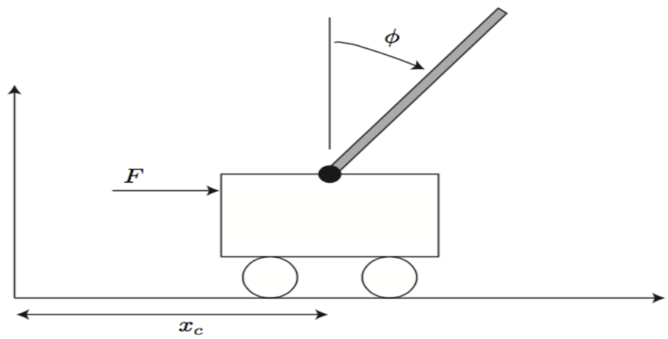

Figure 4: Pendulum on a cart System

$$\gamma \ddot{x}_c - \beta \ddot{\phi} \cos\phi + \beta \dot{\phi}^2 \sin\phi + \mu \dot{x}_c = F$$

$$\alpha \ddot{\phi} - \beta \ddot{x}_c \cos\phi - D \sin\phi = 0,$$

Figure 5: Equations of motion

2.a

a) given state $x = \begin{bmatrix} x_c \\ \phi \\ \dot{x}_c \\ \dot{\phi} \end{bmatrix}$ and $u = F$ (input)

1) $M = \begin{bmatrix} \gamma & -\beta\cos\phi \\ -\beta\cos\phi & \alpha \end{bmatrix}$

given $M \begin{bmatrix} \ddot{x}_c \\ \ddot{\phi} \end{bmatrix} = N$

$\Rightarrow N = \begin{bmatrix} \gamma \ddot{x}_c - \ddot{\phi}\beta\cos\phi \\ -\ddot{x}_c \beta\cos\phi + \ddot{\phi}\alpha \end{bmatrix}$

From the equations of motion,

$N = \begin{bmatrix} F - \beta\dot{\phi}^2 \sin\phi - \mu \dot{x}_c \\ D\sin\phi \end{bmatrix}$

Substitute $F = u$

$N = \begin{bmatrix} u - \beta\dot{\phi}^2 \sin\phi - \mu \dot{x}_c \\ D\sin\phi \end{bmatrix}$

2) $$M^{-1} = \frac{1}{\gamma\alpha - (\beta\cos\phi)^2}\begin{bmatrix}\alpha & \beta\cos\phi \\ \beta\cos\phi & \gamma\end{bmatrix}$$

The non linear state space equation:

$$\begin{bmatrix}\ddot{x}_c \\ \ddot{\phi}\end{bmatrix} = M^{-1}N$$

$$= \frac{1}{\gamma\alpha - (\beta\cos\phi)^2}\begin{bmatrix}\alpha & \beta\cos\phi \\ \beta\cos\phi & \gamma\end{bmatrix}\begin{bmatrix}u - \beta\dot{\phi}^2\sin\phi - \mu\dot{x}_c \\ D\sin\phi\end{bmatrix}$$

$$= \frac{1}{\gamma\alpha - (\beta\cos\phi)^2}\begin{bmatrix}\alpha u - \alpha\beta\dot{\phi}^2\sin\phi - \alpha\mu\dot{x}_c + \beta\cos\phi D\sin\phi \\ \beta\cos\phi u - \frac{1}{2}\beta^2\dot{\phi}^2\sin 2\theta - \mu\dot{x}_c\beta\cos\theta + \gamma D\sin\phi\end{bmatrix}$$

The non-linear state-space model:

$$\dot{x} = \begin{bmatrix}\dot{x}_c \\ \dot{\phi}_c \\ \ddot{x}_c \\ \ddot{\phi}_c\end{bmatrix} = \frac{1}{\gamma\alpha - (\beta\cos\phi)^2}\begin{bmatrix}[\gamma\alpha - (\beta\cos\phi)^2]\,\dot{x}_c \\ [\gamma\alpha - (\beta\cos\phi)^2]\,\dot{\phi}_c \\ \alpha u - \alpha\mu\dot{x}_c - \alpha\beta\dot{\phi}^2\sin\phi + \frac{1}{2}\beta D\sin 2\phi \\ \beta\cos\phi u - \frac{1}{2}\beta^2\dot{\phi}^2\sin 2\phi - \mu\dot{x}_c\beta\cos\phi + \gamma D\sin\phi\end{bmatrix}$$

2.b

MATLAB output:

```
>> symbolic_solve_2
The accelerations are:
        (- alpha*sin(phi)*beta*phi_dot^2 + alpha*u - alpha*mu*xc_dot + cos(phi)*sin(phi)*D*beta)/(alpha*gamma - beta^2*cos(phi)^2)
(- cos(phi)*sin(phi)*beta^2*phi_dot^2 + u*cos(phi)*beta + sin(phi)*D*gamma - mu*xc_dot*cos(phi)*beta)/(alpha*gamma - beta^2*cos(phi)^2)
```

This solution matches the one derived by hand. In my derivation by hand, I factored a term out of the 'matrix' to increase readability. It is evident that upon simplifying, my solution's last two rows match the solution given by MATLAB

2.c

To find the equilibrium points x_e for this system, we can set $\dot{x} = 0$:

$$\dot{x_c} = 0$$
$$\dot{\phi} = 0$$
$$\ddot{x_c} = 0$$
$$\ddot{\phi} = 0$$

Also considering the system in its unforced state, we can set u = 0.
Hence, the state space equation becomes:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \frac{1}{\gamma\alpha} \begin{bmatrix} 0 \\ 0 \\ 0.5 * \beta D \sin 2\phi \\ \gamma D \sin\phi \end{bmatrix}$$

Equating these we get,

$$\frac{\beta D \sin\phi \cos\phi}{\gamma\alpha} = 0$$
$$\frac{D \sin\phi}{\alpha} = 0$$

Since β, D, γ, α are non-zero constants, we are left with,

$$\sin\phi = 0$$

The equilibrium points are:

$$x_e = \begin{bmatrix} x_c \\ n\pi \\ 0 \\ 0 \end{bmatrix}, \forall n \in Z, \forall x_c \in R.$$

This means that in the absence of external forces, the system will come to equilibrium when the cart is stationary ($\dot{x_c} = 0$) at any position on the same horizontal surface and when the pendulum is at rest ($\dot{\phi} = 0$) either in the upright ($\phi = (0, 2\pi, ...)$) or inverted ($\phi = (\pi, 3\pi, ...)$) position.

2.d

The system has been linearized into the form $\dot{x}(t) = Ax(t) + Bu(t)$ using provided values for the constants :

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & -3 & 0 \\ 0 & 2 & -3 & 0 \end{bmatrix} ; B = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

The eigenvalues of A, computed using MATLAB are: {0, -3.3301, 1.1284, -0.7984}.

Since there exists at least one eigenvalue with a real part greater than 0, the linearized system is **unstable** about the equilibrium point x = 0.

The same reasoning applies for the original non linear system, which is also **unstable** about the equilibrium point x = 0.

2.e

The following plots were obtained for the state variables of the **Linear System**:

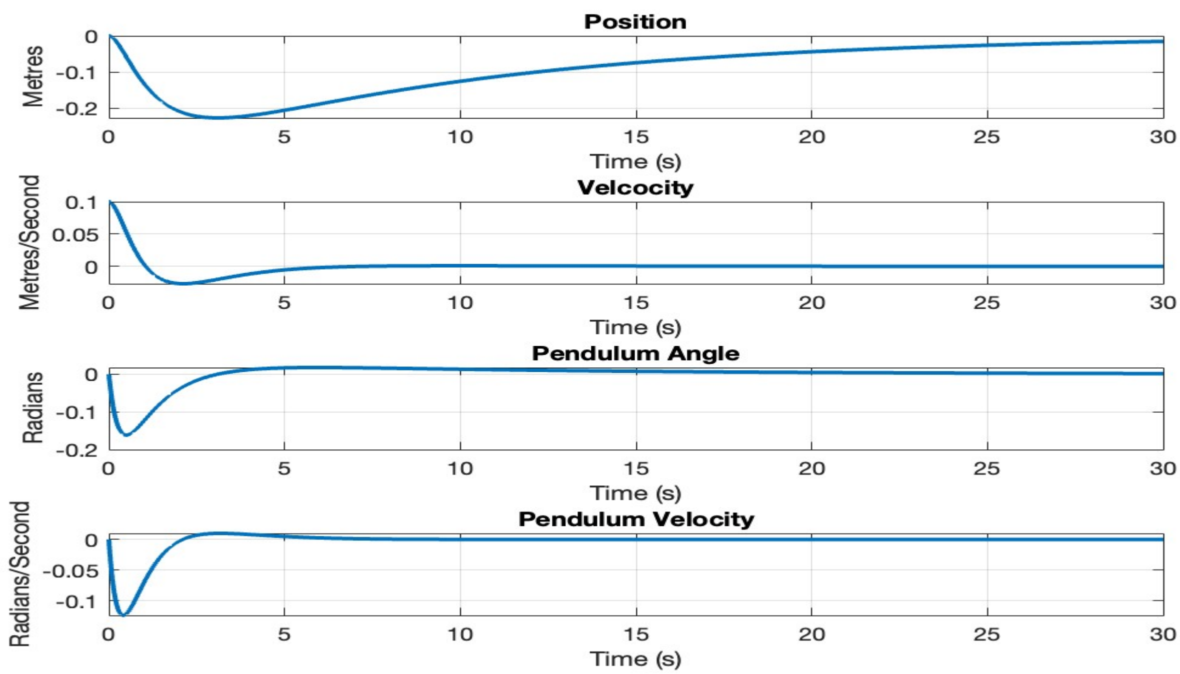

Figure 6: With initial state as [0, 0.1, 0, 0]

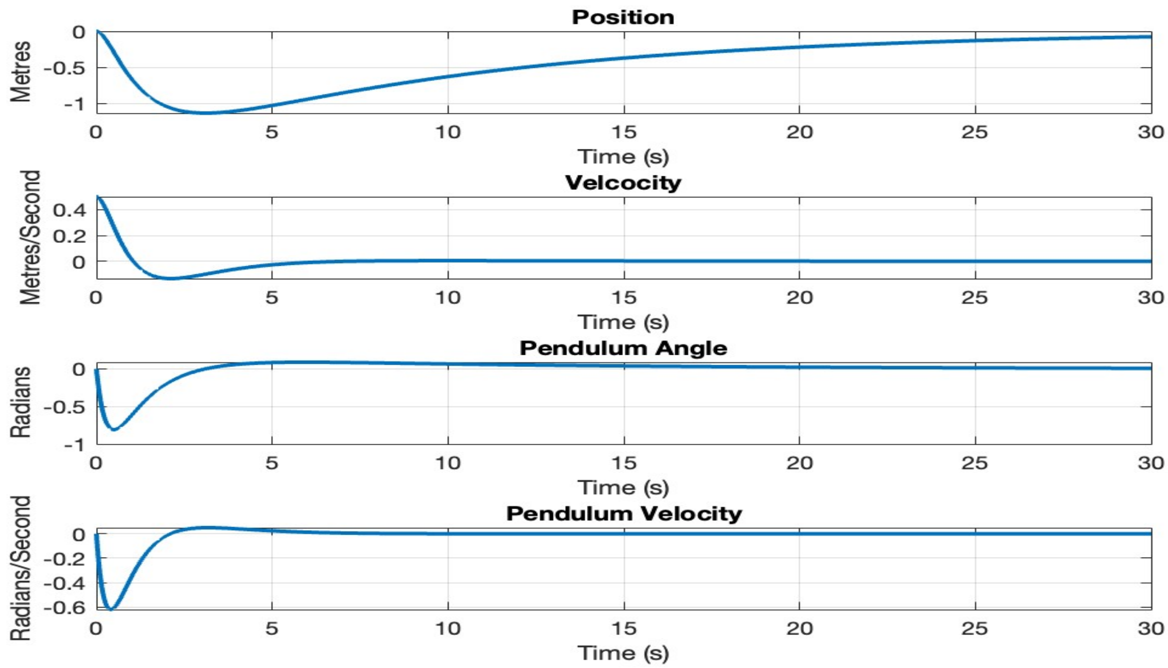

Figure 7: With initial state as [0, 0.5, 0, 0]

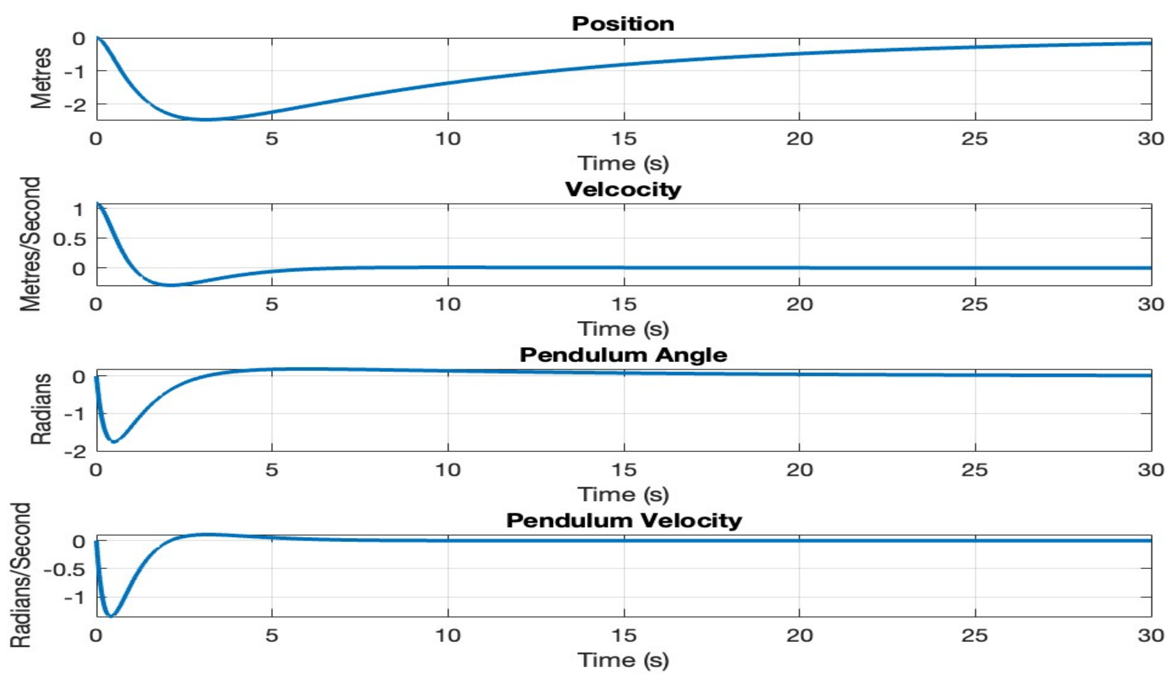

Figure 8: With initial state as [0, 1.0886, 0, 0]

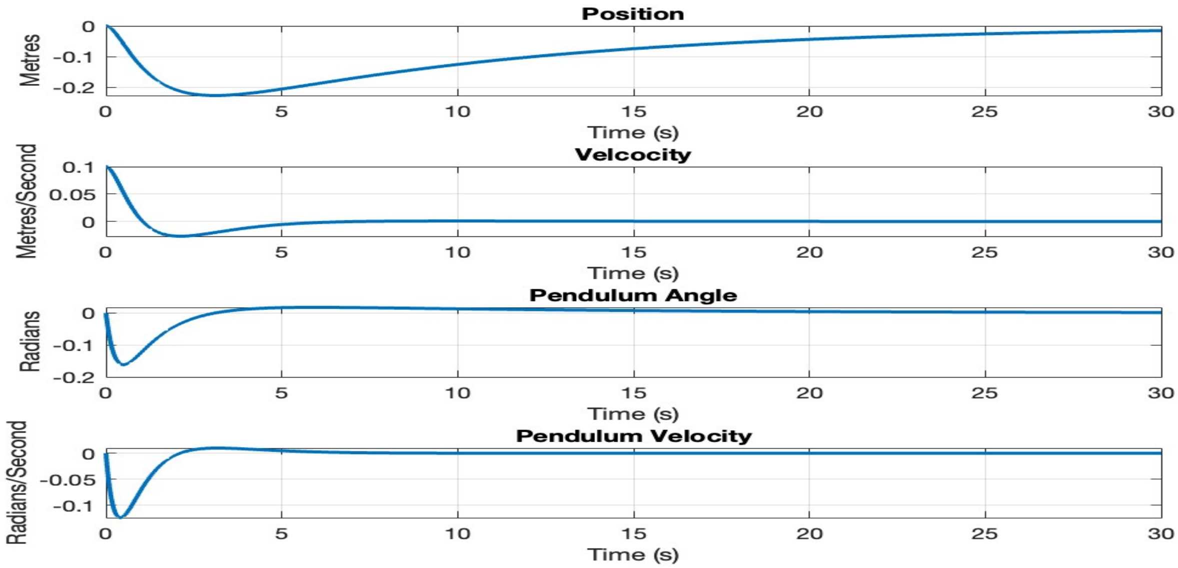

Figure 9: With initial state as [0, 1.1, 0, 0]

2.f

The following plots were obtained for the state variables of the **Non Linear System**

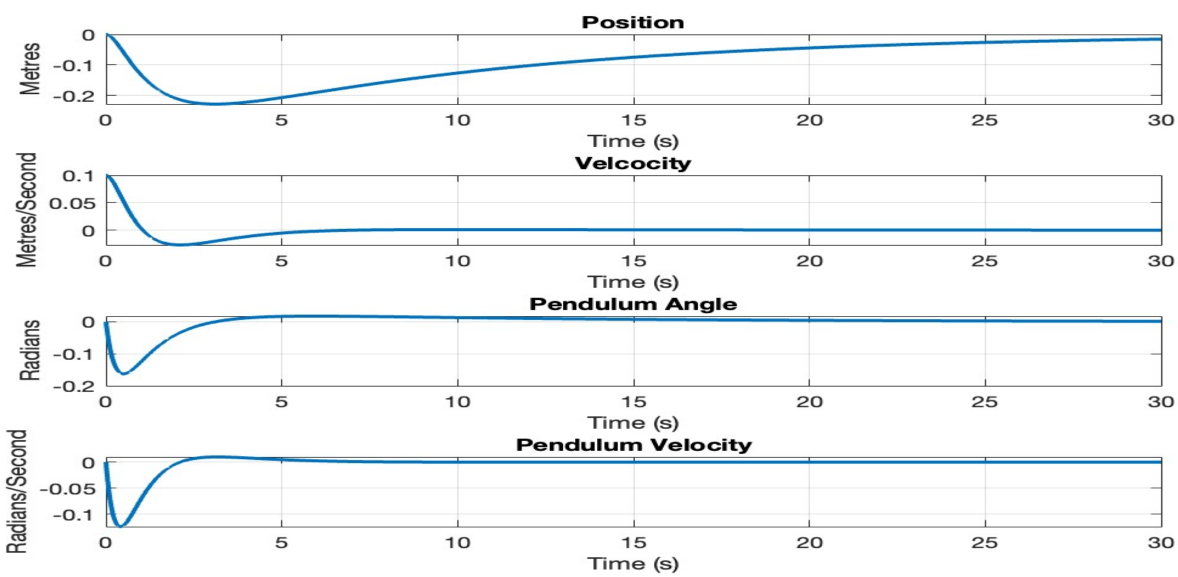

Figure 10: With initial state as [0, 0.1, 0, 0]

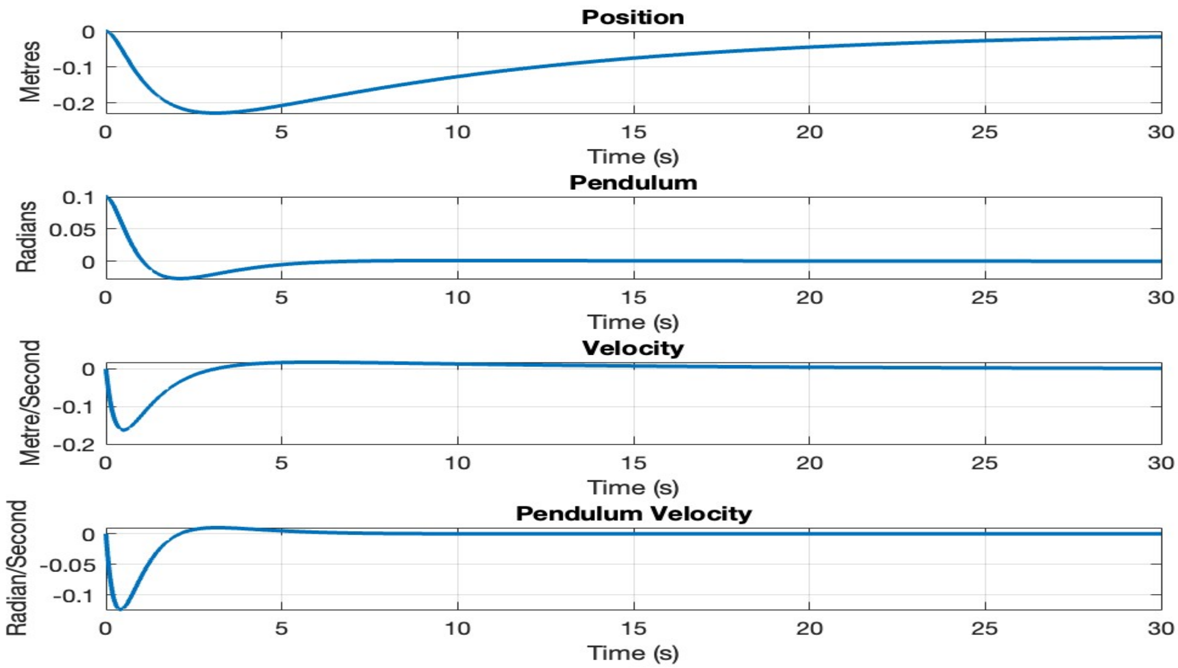

Figure 11: With initial state as [0, 0.5, 0, 0]

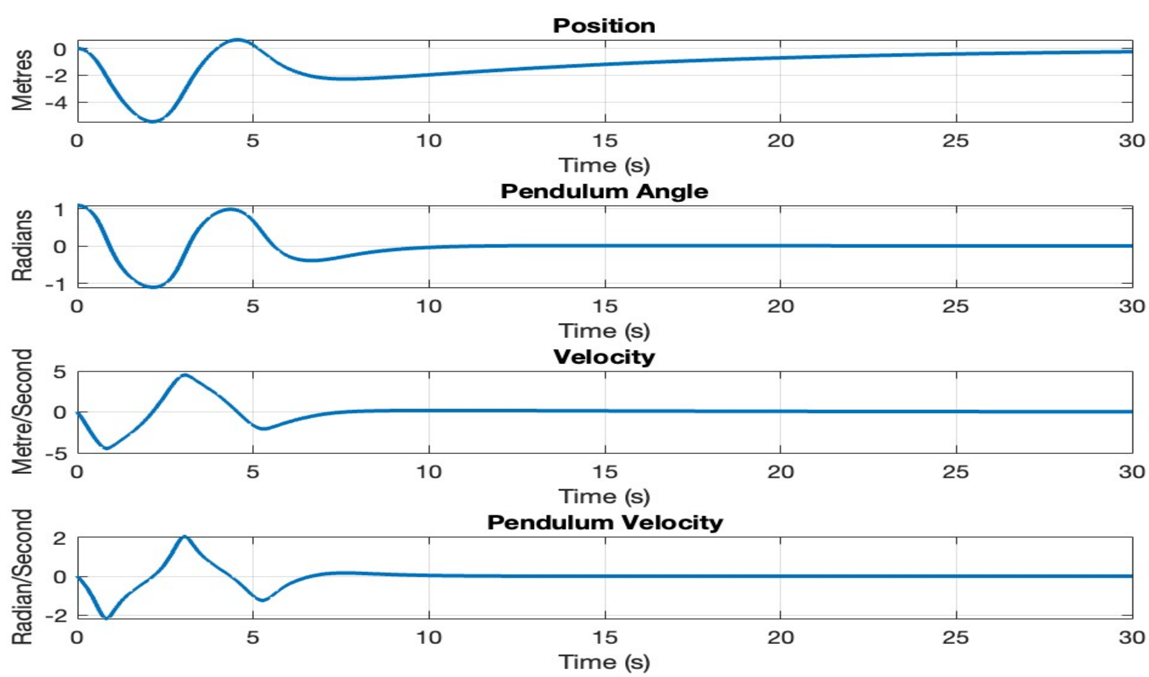

Figure 12: With initial state as [0, 1.0886, 0, 0]

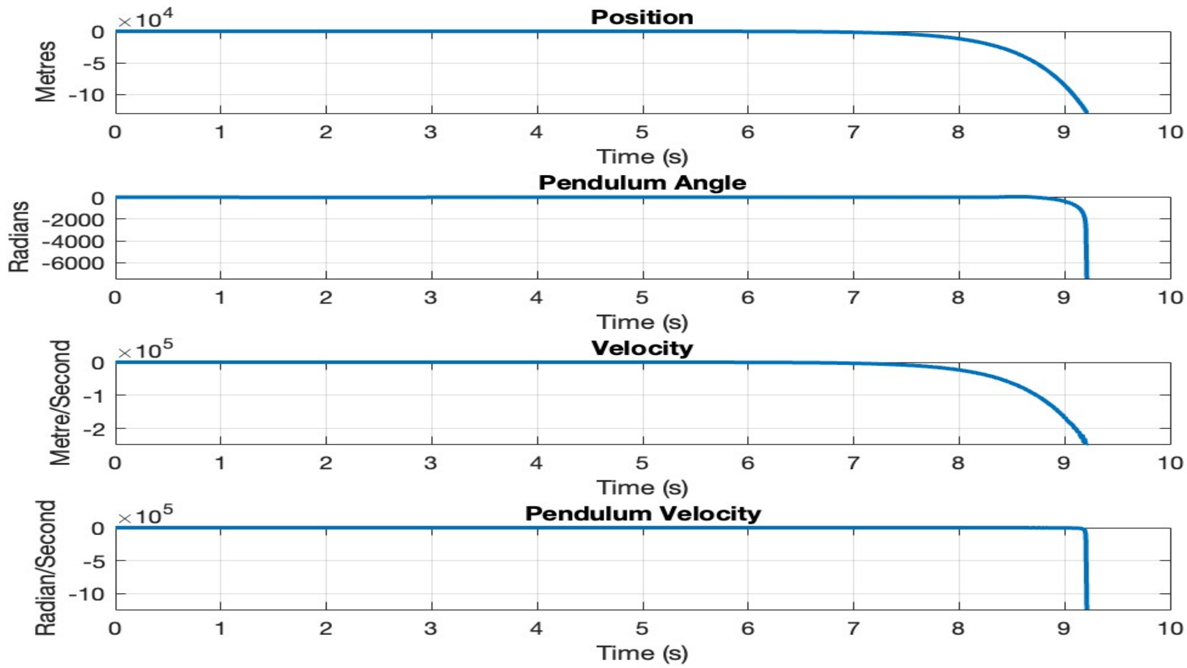

Figure 13: With initial state as [0, 1.1, 0, 0]

MATLAB throws this error:

```
Warning: Failure at t=9.213377e+00.  Unable to meet integration tolerances without reducing the step size below
the smallest value allowed (2.842171e-14) at time t.
> In ode45 (line 350)
In Pendulum_on_a_Cart_Non_Linear (line 86)
```

It can hence be observed that the Linear System is stable, while the Non Linear System is stable 'near' the equilibrium point x=0. However, the Non Linear System becomes unstable after about 2 seconds. When the initial state is $[0, 1.1, 0, 0]$, MATLAB is unable to solve the ODE.

2.g

The matrix C was derived to be:

$$C = \begin{bmatrix} 39.37 & 0 & 0 & 0 \end{bmatrix}$$

2.h

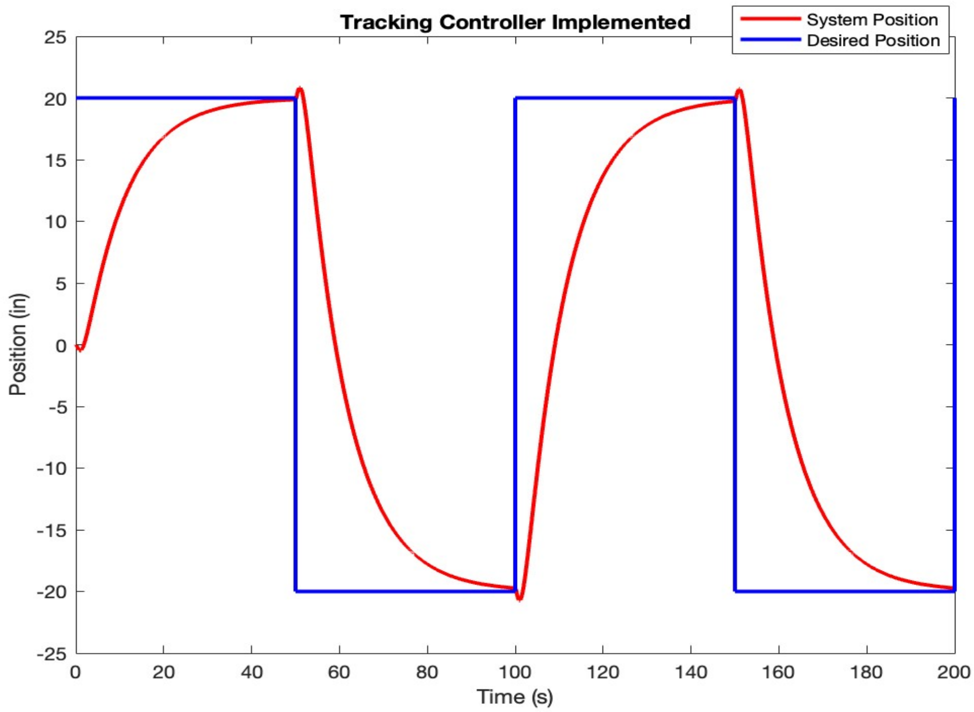

Figure 14: Tracking Controller

2.i

The approach I took to make the controller "better" was to try to get the actual path as close to the desired path as possible for the greatest number of time steps. However, I had to ensure that at sharp position changes, the system does not overshoot too much.
The best balance I got was using these values:

$$Q_u = 1$$

$$Q_x = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The intuition behind this was to penalise the position and velocity deviations of the cart by a large amount while ignoring the pendulum states. The penalty on using input energy was also removed

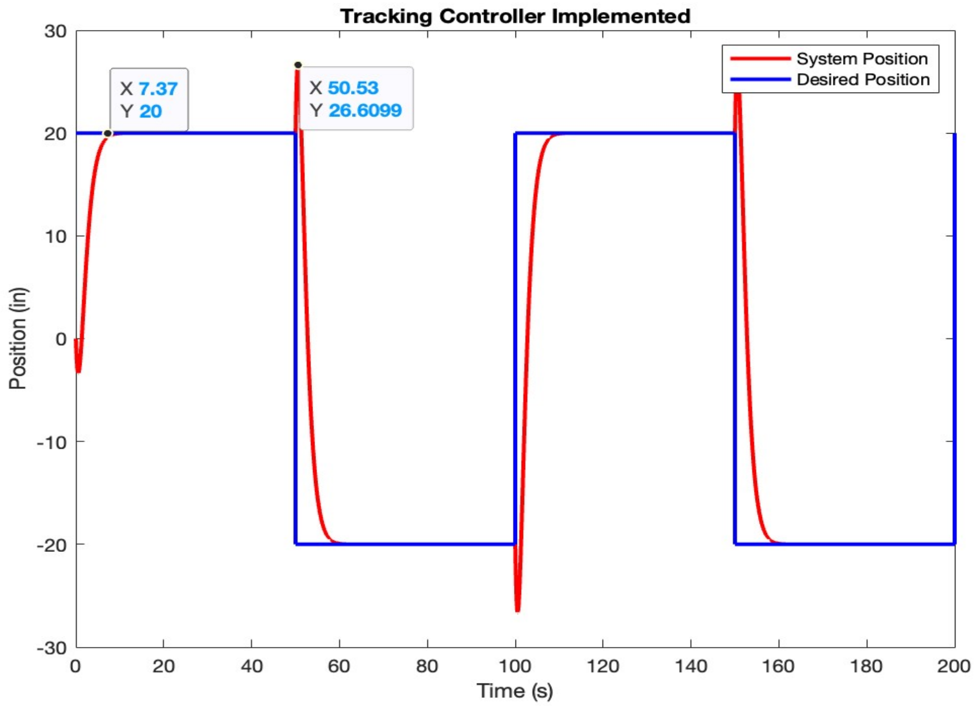

Figure 15: Tracking Controller After Tuning

— THE END —