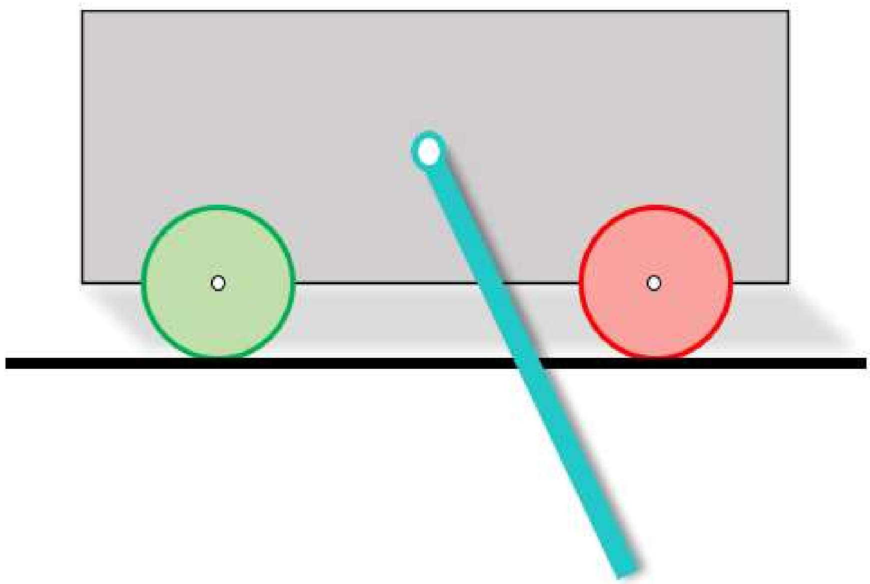
MEEM 4775 Project 1 Report

SIMULATION OF A CART AND ROD SYSTEM AND DESIGN OF A PROPORTIONAL CONTROLLER



Created & Submitted by: Vrushaketu Mali

03-Oct-2020

Table of Contents

Chapter 1. Introduction

The first step in the dynamic system simulation and control system design is to create the differential equations of the system to be simulated. These models can either be constructed from the physics equations or experimental data. Apart from writing bunch of syntax in the MATLAB code, we can also simulate the system by mathematical modeling in Simulink.

Simulink is a model-based design environment integrated with MATLAB used for modeling, simulating and analyzing multi-domain dynamic systems. It offers a way to solve the dynamic system equations numerically rather than requiring a code. Models contain blocks which are mathematical functions and signals which are lines connecting different blocks serving as carrying input and output values.

The derived differential equations of the system may be linear or non-linear. The non-linear system considers the effect of all states interactions and thus represents the true response of the system. However, for developing a control system we need to have a simplified system which can be done by linearizing the non-linear system where me make certain assumptions. The intent of doing this is to verify the approach of control system design followed based on the linear system also gives satisfactory response for non-linear system.

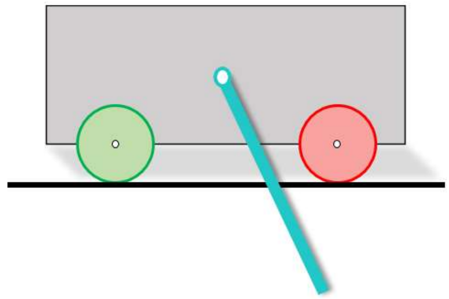
Chapter 2. Summary

This project aims at creating signal-based math model in Simulink for linearized version of original non-linear model of cart and a rod system and then developing a proportional controller design for the cart to follow the commanded position. Thus, the project involves deriving the differential equations, writing a MATLAB script and building a mathematical model for simulation and control of the system. Some level of model verification is also shown in the project.

Accomplishments:

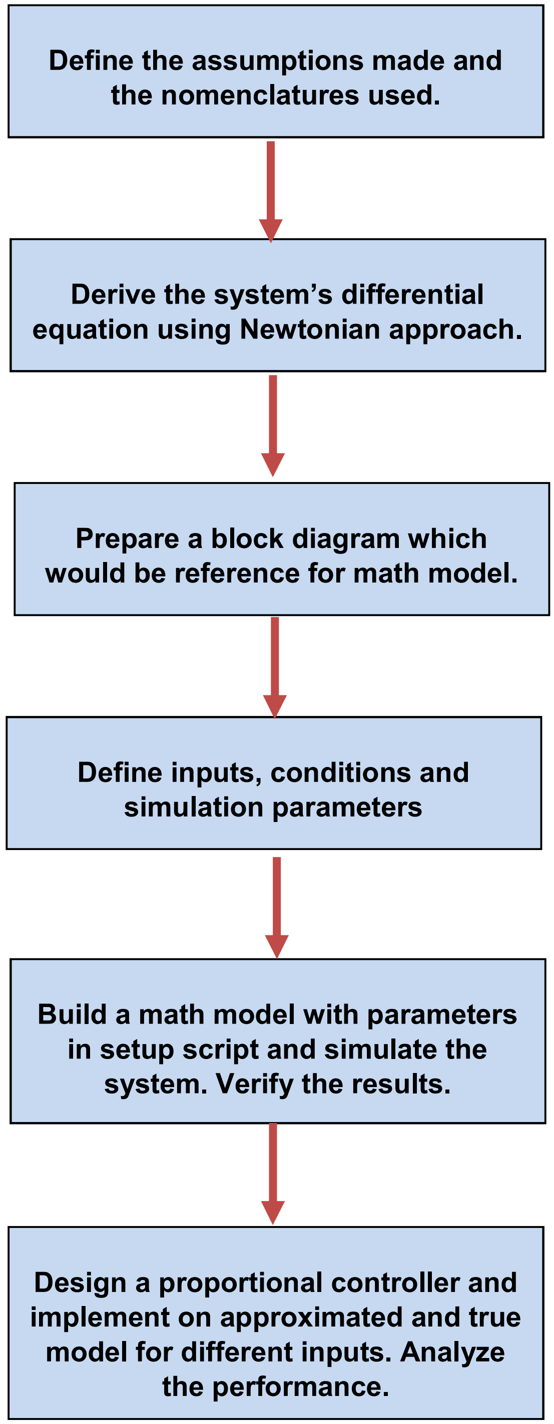
1. In this project, first of all we have derived the differential equations of system using Newtonian approach.
2. Then, the non-linear system is converted to linear system which is simulated. Further, this linear system is simplified in order to build a proportional controller on it.
3. Proportional control design approach is discussed and documented and is verified against different reference inputs provided to the system for the position of the cart.
4. Also, some aspects of this system such as comparison between approximated decoupled model and true coupled model response is discussed.

Chapter 3. Project/Problem Statement



1. Derive the system's nonlinear differential equation model where you use x for the cart position and Ø for the rod angle where Ø = 0 implies the rod is hanging down. Note: using x for the cart position requires you to be careful with the state space representation since the vector X is often used to denote the state. We'll call this the "Nonlinear Model." Document the derivation in your report.
2. Create a linearized, state variable model, called the "True Model," where you assume the rod angle and rate to be small and hanging down in steady-state. Simulate this model in Simulink where you configure its parameters using a rational setup script. In your report you'll need to show how you developed the model and some level of model verification.
3. Create a new differential equation model by decoupling the two linearized differential equations. Here you just remove the terms that cause them to be coupled. The resulting model is two differential equations, one in x and one in Ø.
4. Implement a proportional controller to both the coupled and decoupled plants. Create a switchable input so can easily toggle between a step input and sine wave input.
5. Design a proportional controller, using the decoupled model's cart position transfer function, so that its position step response has 20% overshoot. After determining the gain analytically, apply it to the decoupled model and verify that it works as expected. Document your controller design approach in your report.
6. Apply the same gain to the fully, coupled model. Log data for both a step command of 30 cm, and a sine command with amplitude 30 cm and frequency 1 rad/s.
7. Perform another comparison where the reference input is a sine wave with amplitude of 30 cm, but where the frequency is the same as that of your swinging rod. Again, compare the responses between the approximated and true model.

Chapter 4. Project Approach



Chapter 5. Assumptions and Nomenclatures

While deriving the differential equations of this system and simulating the system, following assumptions and nomenclatures are used.

1. The wheels of the cart purely roll and do not slip.
2. The cart, the rod and the motor's rotating bits are the only masses.
3. The idler wheel (green axle) is frictionless and so it acts as a support sliding on a frictionless surface.
4. The motor's inductance term is neglected because of small value.
5. The wheels are always remaining in contact with the surface. That means, the cart never rotates.
6. The value of gravitational acceleration is considered as 9.81 m/s^2.

m_w = mass of wheels, motor armature and other rotating parts

m_c = Mass of cart, motor housing and other non-rotating parts

$m_T = m_c + m_w$

J_m = Armature inertia of the motor

J = Gearbox inertia

$J_w = J_m + J$ = Effective inertia of rotating parts

$x, \dot{x}, \ddot{x}$ = Displacement, velocity and acceleration of cart in positive x direction

$\theta, \dot{\theta}, \ddot{\theta}$ = Angular displacement, velocity and acceleration of the driving wheel

k_t = The torque constant of the motor

k_e = Back e.m.f constant of the motor

B_m = Damping coefficient of motor

R_a = Armature Resistance of motor

$\theta_m, \dot{\theta}_m$ = Angular position & velocity of motor shaft

τ_m = Motor Torque

r = radius of pinion in gearbox driving the wheel

N = Gear ratio

η = Gearbox efficiency

F_t = Frictional force between wheel and the surface

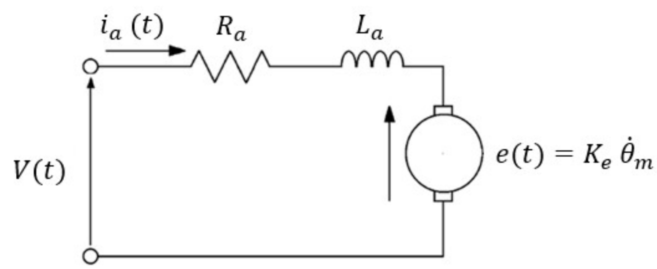
Chapter 6. Derivation of non-linear differential equations of the system

As we are going to create a signal-based model, we need to have the differential equations of the system ready so that we can enter those equations (in the matrix form) in the signal-based model and run the simulation. In this project, we are going to derive the differential equation of the system using Newtonian approach.

Approach followed for deriving the differential equation:

1. Consider only cart, motor and wheel and then derive the equations.
2. Add the rod and derive the equation of rod.

6.1 DC Motor Modeling:



According to Kirchhoff's voltage law, the summation of voltage around a loop in any circuit is equal to zero.

According to Ohm's law, the current is proportional to voltage and inversely proportional to resistance giving the relation V = I.R

Voltage drop across resistor = $R_a . i_a$

Voltage drop across inductor = $L_a . \frac{d}{dt}(i_a)$

Voltage loss due to back e.m.f = $K_e . \dot{\theta}_m$ where K_e is back emf constant

Thus, applying KVL and Ohm's law

$$V - L_a . \frac{d}{dt}(i_a) - R_a . i_a - K_e . \dot{\theta}_m = 0$$

Neglecting the effect of inductance and rearranging this equation for armature current

As per the gear dynamics, the angular position of driven gear is reduced by a factor of N which gives relationship between motor shaft position and output shaft position

$$\theta_m = N . \theta$$

Also, the translation of the cart can be given as

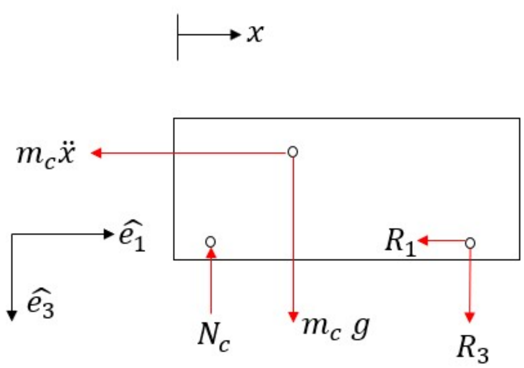
$$x_A = r.\theta$$

Using these relationships and neglecting the effect of inductance we can re-arrange this equation for armature current

$$i_a = \frac{V - K_e.\dot{\theta}_m}{R_a} = \frac{V - K_e.N.\dot{\theta}}{R_a} = \frac{V - \frac{K_e.N}{r}\dot{x}}{R_a} \ldots\ldots\ldots\ldots\ldots\ldots..\mathbf{1}$$

6.2 Cart Modeling:

Let's draw the free body diagram of the cart.



Looking at FBD above, we can write the equations as

Along $\widehat{e_1}$ $\quad -m_c\,\ddot{x} - R_1 = 0 \ldots\ldots\ldots\ldots\ldots\ldots..\mathbf{2}$

Along $\widehat{e_3}$ $\quad -N_c + m_c\,g + R_3 = 0$

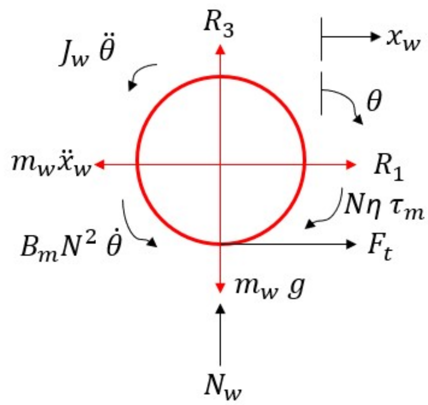
6.3 Wheel Modeling:

As per our assumption, the cart is only translating and there is no rotation. Thus, whatever is the translation of the cart is equal to the translation of the wheel.

$$x_w = x$$

We will make this change in the derived equation of the wheel.

Let's draw the free body diagram of the wheel.



Along $\widehat{e_1}$ $\quad -m_w\,\ddot{x} + R_1 + F_t = 0$ ……………….. **3**

Along $\widehat{e_3}$ $\quad m_w\,g - R_3 - N_w = 0$

Along rotation (considering positive for wheel rotation)

$$\frac{-J_w\ddot{x}}{r} - \frac{B_m N^2 \dot{x}}{r} + N\eta\tau_m - F_t\,.\,r = 0$$

The torque for motor is defined as the product of torque constant and armature current.

$$\tau_m = k_t\,.\,i_a$$

Substituting this in above equation

$$\frac{-J_w\ddot{x}}{r} - \frac{B_m N^2 \dot{x}}{r} + N\eta k_t\,.\,i_a - F_t\,.\,r = 0 \quad \text{………………..}\,\mathbf{4}$$

Using equations from 1 to 4, we can write the differential equation for cart movement.
Using equations 2 & 3, we can substitute for R_1 and can write the equation

$$-m_w\,\ddot{x} - m_c\,\ddot{x} + F_t = 0$$

$$F_t = (m_w + m_c)\;\ddot{x}$$

Putting this expression of F_t in equation 4

$$\frac{-J_w\ddot{x}}{r} - \frac{B_m N^2 \dot{x}}{r} + N\eta k_t \, . \, i_a - ((m_w + m_c) \ \ddot{x}\,)\, r = 0$$

We can put the expression for i_a from equation 1

$$\frac{-J_w\ddot{x}}{r} - \frac{B_m N^2 \dot{x}}{r} + \ N\eta k_t \left(\frac{V - \frac{K_e . N}{r} \ \dot{x}}{R_a} \right) - ((m_w + m_c) \ \ddot{x}\,)\, r = 0$$

Diving whole equation by r and re-arranging coefficients for $\ddot{x}$

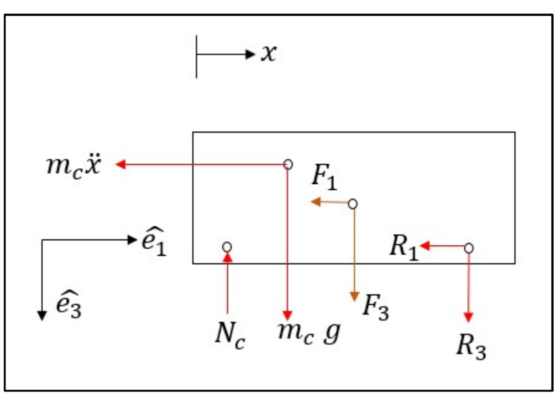
$$\left(\frac{-J_w}{r^2} - \ m_w - m_c \right) \ddot{x} - \frac{B_m N^2 \dot{x}}{r^2} + \frac{N\eta k_t \, V}{r \, R_a} - \frac{N^2 \eta k_t \, K_e \, \dot{x}}{r^2 \, R_a} = 0$$

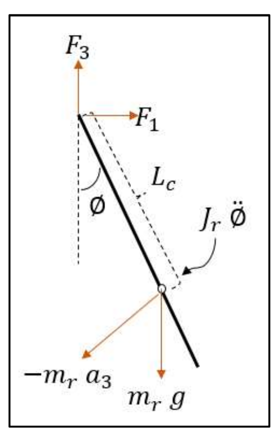
Re-arranging for the coefficients of $\ddot{x}$ and sign changes

$$\left(\frac{J_w}{r^2} + \ m_w + m_c \right) \ddot{x} + \left(\frac{N}{r} \right)^2 \left(B_m + \frac{\eta k_t \, K_e}{R_a} \right) \dot{x} = \frac{N\eta k_t \, V}{r \, R_a}$$

6.4 Rod Modeling:

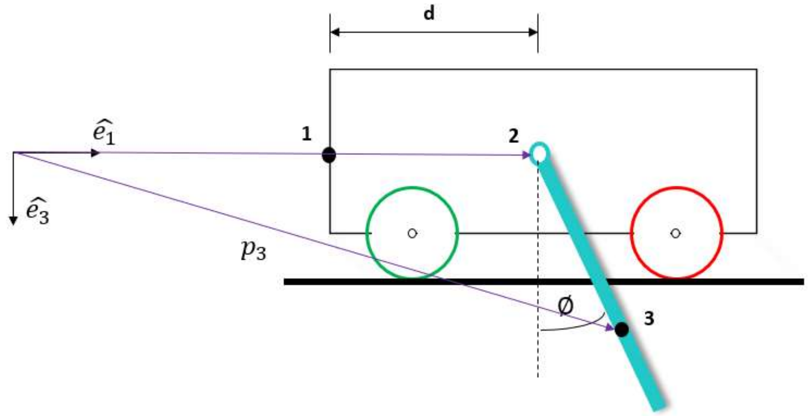
Now, when we add the rod, the drive wheel and motor equations remain unchanged while the cart equations get changed because of the reactions we consider between rod and the cart. Therefore, we will redraw the FBD for cart and also draw and FBD for rod.





Here a_3 is the absolute acceleration of the rod with respect to the global co-ordinate frame.

In order to find a_3, we have to find the position of point 3 (p_3) with respect to global coordinate as per the kinematics shown in the figure below.



$$p_3 = p_1 + p_{2/1} + p_{3/2}$$

$$p_3 = \begin{bmatrix} x \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} L_c \sin(\phi) \\ 0 \\ L_c \cos(\phi) \end{bmatrix}$$

$$\dot{p_3} = \begin{bmatrix} \dot{x} + L_c \dot{\phi} \cos(\phi) \\ 0 \\ -L_c \dot{\phi} \sin(\phi) \end{bmatrix}$$

$$\ddot{p_3} = \begin{bmatrix} \ddot{x} + L_c \ddot{\phi} \cos(\phi) - L_c \dot{\phi}^2 \sin(\phi) \\ 0 \\ -L_c \ddot{\phi} \sin(\phi) - L_c \dot{\phi}^2 \cos(\phi) \end{bmatrix} = \begin{bmatrix} a_{31} \\ a_{32} \\ a_{33} \end{bmatrix}$$

Now, rewriting the equations for cart, wheel and rod (Wheel equation remains unchanged)

Looking at FBD of cart, we can write the equation as

Along $\widehat{e_1}$ $\quad -m_c\,\ddot{x} - R_1 - F_1 = 0 \ldots\ldots\ldots\ldots\ldots\ldots..\mathbf{5}$

Looking at FBD of rod, we can write the equation as

Along $\widehat{e_1}$ $\quad F_1 - m_r\,a_{31} = 0 \ldots\ldots\ldots\ldots\ldots\ldots..\mathbf{6}$

Along $\widehat{e_2}$ $\quad -F_3 - m_r\,a_{33} + m_r g = 0 \ldots\ldots\ldots\ldots\ldots\ldots..\mathbf{7}$

$$-J_r\,\ddot{\emptyset} - F_1\,L_c\,\cos\emptyset - F_3\,L_c\,\sin\emptyset = 0 \ldots\ldots\ldots\ldots\ldots\ldots..\mathbf{8}$$

Putting the values of a_{31} & a_{33} from acceleration matrix in equation 6 & 7

$F_1 - m_r\left(\ddot{x} + L_c\ddot{\emptyset}\,\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right) = 0$ which can be written as

$$F_1 = m_r\left(\ddot{x} + L_c\ddot{\emptyset}\,\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right)$$

$F_3 + m_r\left(-L_c\ddot{\emptyset}\,\sin\emptyset + L_c\,\dot{\emptyset}^2\cos\emptyset\right) - m_r g = 0$ which can be written as

$$F_3 = -m_r\left(-L_c\ddot{\emptyset}\,\sin\emptyset + L_c\,\dot{\emptyset}^2\cos\emptyset\right) + m_r g$$

Putting the value of F_1 in equation 5

$$-m_r\left(\ddot{x} + L_c\ddot{\emptyset}\,\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right) - m_c\,\ddot{x} = R_1$$

From wheel modeling we do have the equation 3 which is

$$-m_w\,\ddot{x} + R_1 + F_t = 0$$

Putting the value of R_1 here and solving for F_t

$$F_t = m_w\,\ddot{x} + m_r\left(\ddot{x} + L_c\ddot{\emptyset}\,\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right) + m_c\,\ddot{x}$$

From wheel modeling we do have the equation 4 which is

$$\frac{-J_w\ddot{x}}{r} - \frac{B_m N^2\dot{x}}{r} + N\eta k_t\,.\,i_a - F_t\,.\,r = 0$$

Putting the entire expression of F_t now in this equation,

$$\frac{-J_w\ddot{x}}{r} - \frac{B_m N^2 \dot{x}}{r} + N\eta k_t\,.\,i_a - \left(m_w\,\ddot{x} + m_r\left(\ddot{x} + L_c\ddot{\emptyset}\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right) + m_c\,\ddot{x}\right) r = 0$$

Multiplying whole equation by r

$$\frac{-J_w\ddot{x}}{r^2} - \frac{B_m N^2 \dot{x}}{r^2} + N\eta k_t\,.\,i_a - \left(m_w\,\ddot{x} + m_r\left(\ddot{x} + L_c\ddot{\emptyset}\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right) + m_c\,\ddot{x}\right) = 0$$

We can put the expression for i_a from equation 1 and rewrite $\theta_m = N.\theta_1$

$$\frac{-J_w\ddot{x}}{r^2} - \frac{B_m N^2 \dot{x}}{r^2} + N\eta k_t\,.\left(\frac{V - \frac{K_e.N}{r}\,\dot{x}}{R_a}\right) - \left(m_w\,\ddot{x} + m_r\left(\ddot{x} + L_c\ddot{\emptyset}\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right) + m_c\,\ddot{x}\right) = 0$$

Let's write $m_w + m_c = m_T$ and combine the coefficients

$$\left(m_T + m_r + \frac{J_w}{r^2}\right)\ddot{x} + \left(m_r\,L_c\cos\emptyset\right)\ddot{\emptyset} + \left(\frac{N}{r}\right)^2\left(B_m + \frac{\eta k_t\,K_e}{R_a}\right)\dot{x} - \left(m_r\,L_c\sin\emptyset\right)\dot{\emptyset}^2 = \frac{N\eta k_t\,V}{r\,R_a} \quad \ldots\ldots\ldots\ldots\ldots\ldots\mathbf{9}$$

Now, if we combine equations, 6, 7 & 8 of the rod

That means, putting the expressions for F_1 & F_3 in equation 8, we get

$$-J_r\,\ddot{\emptyset} - \left(m_r\left(\ddot{x} + L_c\ddot{\emptyset}\cos\emptyset - L_c\,\dot{\emptyset}^2\sin\emptyset\right)\right)L_c\cos\emptyset - \left(-m_r\left(-L_c\ddot{\emptyset}\sin\emptyset + L_c\,\dot{\emptyset}^2\cos\emptyset\right) + m_r g\right)L_c\sin\emptyset = 0$$

Expanding above equation

$$-J_r\,\ddot{\emptyset} - m_r\,L_c\cos\emptyset\,\ddot{x} - m_r\,{L_c}^2\,\ddot{\emptyset}\cos^2\emptyset + \cancel{m_r\,{L_c}^2\,\dot{\emptyset}^2\sin\emptyset\cos\emptyset} - m_r\,{L_c}^2\ddot{\emptyset}\sin^2\emptyset - \cancel{m_r\,{L_c}^2\,\dot{\emptyset}^2\sin\emptyset\cos\emptyset} - m_r\,g\,L_c\sin\emptyset = 0$$

Combining the coefficients and making the sign changes

$$\left(m_r\,L_c\cos\emptyset\right)\ddot{x} + \left(J_r + m_r\,{L_c}^2\right)\ddot{\emptyset} + m_r\,g\,L_c\sin\emptyset = 0 \quad \ldots\ldots\ldots\ldots\ldots\ldots\mathbf{10}$$

These two equations 9 & 10 represent the non-linear differential equations of the model which are again stated below.

$$\left(m_T + m_r + \frac{J_w}{r^2}\right)\ddot{x} + (m_r L_c \cos\emptyset)\ddot{\emptyset} + \left(\frac{N}{r}\right)^2 \left(B_m + \frac{\eta k_t K_e}{R_a}\right)\dot{x} - (m_r L_c \sin\emptyset)\dot{\emptyset}^2 = \frac{N\eta k_t V}{r R_a}$$

$$(m_r L_c \cos\emptyset)\ddot{x} + \left(J_r + m_r {L_c}^2\right)\ddot{\emptyset} + m_r g L_c \sin\emptyset = 0$$

Chapter 7. Linearized State-Space Model

7.1 Linearization

The two equations which we derived above are definitely non-linear. On the other hand, the control system design and analysis methods are much easier for linear models. Therefore, in order to design and implement a control system, we linearize the nonlinear differential equations.

There are different methods by which one can perform the linearization. These methods include linearization by small signal analysis around equilibrium point, linearization by inverse non-linearity, linearization by assuming certain variables have small values and thus neglecting their derivatives.

In our case, we are linearizing two equations by assuming the rod angle and the angular rate of the rod being small and thus we can establish following relationships.

$$\emptyset \ll 1, \;\; \dot{\emptyset} \ll 1$$

And therefore, we can approximate functions

$$\sin\emptyset \approx \emptyset, \quad \cos\emptyset \approx 1 \quad \dot{\emptyset}^2 \approx 0$$

If we use these relations in our derived nonlinear differential equations, we can rewrite them as below

$$\boxed{\begin{aligned} &\left(m_T + m_r + \frac{J_w}{r^2}\right)\ddot{x} + m_r L_c \ddot{\emptyset} + \left(\frac{N}{r}\right)^2 \left(B_m + \frac{\eta k_t K_e}{R_a}\right)\dot{x} = \frac{N\eta k_t V}{r R_a} \\ &m_r L_c \ddot{x} + \left(J_r + m_r L_c{}^2\right)\ddot{\emptyset} + m_r g L_c \emptyset = 0 \end{aligned}}$$

So, these two equations are linearized differential equations of the system.

Let's rename those bigger coefficients as below.

$$a_1 = m_T + m_r + \frac{J_w}{r^2} \;\; , b_1 = m_r L_c \, , c_1 = \left(B_m + \frac{\eta k_t K_e}{R_a}\right)$$

$$b_2 = \left(J_r + m_r L_c{}^2\right) \;\; , c_2 = m_r g L_c, \qquad d = \frac{N\eta k_t}{r R_a}$$

So now, we are able to reduce the equation in much simplified form as below

$$a_1\, \ddot{x} + b_1\, \ddot{\emptyset} + c_1 \dot{x} = d\, V$$

$$b_1\, \ddot{x} + b_2 \ddot{\emptyset} + b_1\, g\, \emptyset = 0$$

Writing for $\ddot{x}$ & $\ddot{\emptyset}$

$$\ddot{x} = \frac{d\, V - b_1\, \ddot{\emptyset} - c_1 \dot{x}}{a_1} \text{ and } \ddot{\emptyset} = \frac{-b_1\, g\, \emptyset - b_1\, \ddot{x}}{b_2}$$

Putting up the equation for $\ddot{\emptyset}$ in $\ddot{x}$ equation

$$\ddot{x} = \frac{d\, V - b_1 \left(\frac{-b_1\, g\, \emptyset - b_1\, \ddot{x}}{b_2}\right) - c_1 \dot{x}}{a_1}$$

After expanding and solving, we get equation for $\ddot{x}$ as

$$\ddot{x} = \frac{\frac{d}{a_1} V + \frac{{b_1}^2\, g}{a_1\, b_2}\, \emptyset - \frac{c_1}{a_1} \dot{x}}{1 - \frac{{b_1}^2}{a_1\, b_2}}$$

If we put the equation of $\ddot{x}$ in $\ddot{\emptyset}$ equation

$$\ddot{\emptyset} = \frac{-b_1\, g\, \emptyset - b_1 \left(\frac{d\, V - b_1\, \ddot{\emptyset} - c_1 \dot{x}}{a_1}\right)}{b_2}$$

After expanding and solving, we get equation for $\ddot{\emptyset}$ as

$$\ddot{\emptyset} = \frac{\frac{-b_1\, g}{b_2}\, \emptyset - \frac{b_1\, d\, V}{a_1\, b_2} + \frac{b_1\, c_1}{a_1\, b_2} \dot{x}}{1 - \frac{{b_1}^2}{a_1\, b_2}}$$

Thus, we have reduced two equations in the form of terms having first derivatives only. Let’s define one more parameter as

$$e = 1 - \frac{{b_1}^2}{a_1\, b_2}$$

7.2 State Space Modeling & Simulating the linear system

Looking at the two equations above, we have two dependent variables x & $\emptyset$

Highest derivative of x = 2, Highest derivative of $\emptyset$ = 2

So, order of the system = 2 + 2 =4

Therefore, we need 4 state variables.

Let's define states input and output as below

$$x_1 = x$$
$$x_2 = \emptyset$$
$$x_3 = \dot{x}$$
$$x_4 = \dot{\emptyset}$$
$$u = V$$
$$y = x, \emptyset\, , \dot{x}\, , \dot{\emptyset}$$

So, the derivatives of the states would be

$$\dot{x}_1 = \dot{x} = \ x_3$$
$$\dot{x}_2 = \dot{\emptyset} \ = x_4$$

$$\dot{x}_3 = \ddot{x} \ = \left(\frac{d}{a_1\ e}\right) u + \left(\frac{{b_1}^2\ g}{a_1\ b_2\ e}\right) x_2 - \left(\frac{c_1}{a_1\ e}\right) x_3$$

$$\dot{x}_4 = \ddot{\emptyset} \ = \left(\frac{-b_1\ g}{b_2\ e}\right) x_2 \ - \left(\frac{b_1\ d}{a_1\ b_2\ e}\right) u + \left(\frac{b_1\ c_1}{a_1\ b_2\ e}\right) x_3$$

$$\dot{x} = f(x, u) = \begin{bmatrix} x_3 \\ x_4 \\ \left(\frac{d}{a_1\ e}\right) u + \left(\frac{{b_1}^2\ g}{a_1\ b_2\ e}\right) x_2 - \left(\frac{c_1}{a_1\ e}\right) x_3 \\ \left(\frac{-b_1\ g}{b_2\ e}\right) x_2 - \left(\frac{b_1\ d}{a_1\ b_2\ e}\right) u + \left(\frac{b_1\ c_1}{a_1\ b_2\ e}\right) x_3 \end{bmatrix}$$

$$y = g(x, u) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

For the linear equations, we can create state space matrices as below

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \frac{\partial f_1}{\partial x_4} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \frac{\partial f_2}{\partial x_4} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} & \frac{\partial f_3}{\partial x_4} \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_2} & \frac{\partial f_4}{\partial x_3} & \frac{\partial f_4}{\partial x_4} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{b_1^2 g}{a_1 b_2 e} & \frac{-c_1}{a_1 e} & 0 \\ 0 & \frac{-b_1 g}{b_2 e} & \frac{b_1 c_1}{a_1 b_2 e} & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{\partial f_1}{\partial u} \\ \frac{\partial f_2}{\partial u} \\ \frac{\partial f_3}{\partial u} \\ \frac{\partial f_4}{\partial u} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \frac{d}{a_1 e} \\ \frac{-b_1 d}{a_1 b_2 e} \end{bmatrix}$$

$$C = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \frac{\partial g_1}{\partial x_3} & \frac{\partial g_1}{\partial x_4} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \frac{\partial g_2}{\partial x_3} & \frac{\partial g_2}{\partial x_4} \\ \frac{\partial g_3}{\partial x_1} & \frac{\partial g_3}{\partial x_2} & \frac{\partial g_3}{\partial x_3} & \frac{\partial g_3}{\partial x_4} \\ \frac{\partial g_4}{\partial x_1} & \frac{\partial g_4}{\partial x_2} & \frac{\partial g_4}{\partial x_3} & \frac{\partial g_4}{\partial x_4} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} \frac{\partial g_1}{\partial u} \\ \frac{\partial g_2}{\partial u} \\ \frac{\partial g_3}{\partial u} \\ \frac{\partial g_4}{\partial u} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Now, we can write the MATLAB code to calculate state-space matrices.

```
%% Clean Up
clearvars;  % Clear the workspace
clc;  % Clear the command window
close('all');  % Close the open figures
Simulink.sdi.clear;  % Clear the simulink data inspector


%% Physical Constancts
g  = 10;   % grav acceleration in m/s^2


%% Problem Statement Parameters
% Motor
Motor.Kt =  7.68*10^-3; % Torque Constant(Nm/A)
Motor.Ke = Motor.Kt;  % Back emf constant
Motor.Ra =  2.60;   % Armature resistance(ohm)
Motor.La =  1.8*10^-4; % Inductance (H)
Motor.Jm = 3.90*10^-7;  % rotational moment of Inertia(Kg-m2)
Motor.Bm = 1.0e-6; % Viscous friction factor of motor


% Gear box parameters
Gear.R =  6.35*10^-3; % Driving gear radius (m)
Gear.N =  3.71; % Gear Ratio
Gear.eta = 0.9;  % Gear box efficiency
Gear.J = 0;  % Gear box inertia


% Cart parameters
Cart.Mc = 0.521;  % Mass of Cart in Kg
Cart.Mw = 0.0; % Mass of wheel in Kg
Cart.Mt = Cart.Mc + Cart.Mw ;  % Total mass of cart
Cart.Jw = Motor.Jm + Gear.J; % Moment of inertia of wheel
Cart.Rw = 0.01483; % Radius of encoder wheel in m


% Rod
Rod.Mr = 0.23; % Mass of rod (Kg)
Rod.Lc = 0.64; % Stiffness of suspension spring (N/m)
Rod.Jr = 7.88e-3;  % Moment of inertia of rod in Kg-m2


%% Lumped Parameters
Calc.a1 = Cart.Mt + Rod.Mr + (Cart.Jw/Gear.R^2);
Calc.b1 = Rod.Mr*Rod.Lc/2;
Calc.c1 = (Gear.N^2/Gear.R^2)*((Gear.eta*Motor.Kt*Motor.Ke/Motor.Ra)+Motor.Bm);
Calc.b2 = Rod.Jr + (Rod.Mr*(0.5*Rod.Lc)^2);
Calc.d = Gear.N*Gear.eta*Motor.Kt/(Gear.R*Motor.Ra);
Calc.e = 1 - (Calc.b1^2/(Calc.a1*Calc.b2));
Mat.A32 = g*Calc.b1^2/(Calc.a1*Calc.b2*Calc.e);
Mat.A33 = -Calc.c1/(Calc.a1*Calc.e);
```

```
Mat.A42 = -g*Calc.b1/(Calc.b2*Calc.e);
Mat.A43 = Calc.b1*Calc.c1/(Calc.a1*Calc.b2*Calc.e);
Mat.B31 = Calc.d/(Calc.a1*Calc.e);
Mat.B41 = -Calc.b1*Calc.d/(Calc.a1*Calc.b2*Calc.e);


%% State Space Matrices
A = [ 0    0        1     0
      0    0        0     1
      0 Mat.A32  Mat.A33  0
      0 Mat.A42  Mat.A43  0 ]; % A Matrix

B = [0;0;Mat.B31;Mat.B41]; % B Matrix

C = [1 0 0 0
     0 1 0 0
     0 0 1 0
     0 0 0 1]; % C Matrix


D = [0;0;0;0]; % D Matrix
```

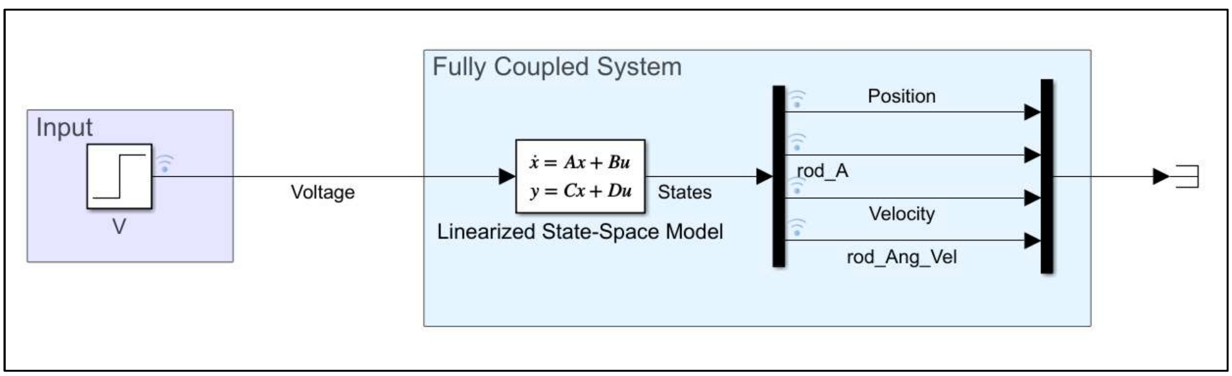
After running this MATLAB command, the numerical matrices we get as below

```
A = 4x4
           0         0    1.0000         0
           0         0         0    1.0000
           0    2.9293  -12.4261         0
           0  -30.2747   29.0965         0

B = 4x1
           0
           0
      2.6400
     -6.1818

C = 4x4
      1     0     0     0
      0     1     0     0
      0     0     1     0
      0     0     0     1

D = 4x1
      0
      0
      0
```

Now, we will simulate this linearized system in Simulink.

For simulating the system, we have used the step input command such that voltage goes from 0 to 1 at 1 second and the system is simulated for 20 seconds.
System is simulated with 0.001 seconds step size and with ode3 as integration method.

```
%% Input Parameters

V = 1;
Input.ts = 1;

%% Simulation parameters
simPrm.solTyp = 'Fixed-step';  % Solver type
simPrm.sol    = 'ode3';  % Solver type 2
simPrm.dt   = 0.001; % Integration step size
simPrm.tEnd = 20; % Simulation end time
simPrm.t = 0:simPrm.dt:simPrm.tEnd;  %simulation time goes from 0 to 20


%% Simulate the math model
set_param('simTrue','SolverType',simPrm.solTyp);  % set this solver type in
simulink parameters
set_param('simTrue','Solver',simPrm.sol);     % set this integration method in
simulink solver
SimOut = sim('simTrue','SignalLoggingName','sdata'); % Simulate the math model and
save the data
```



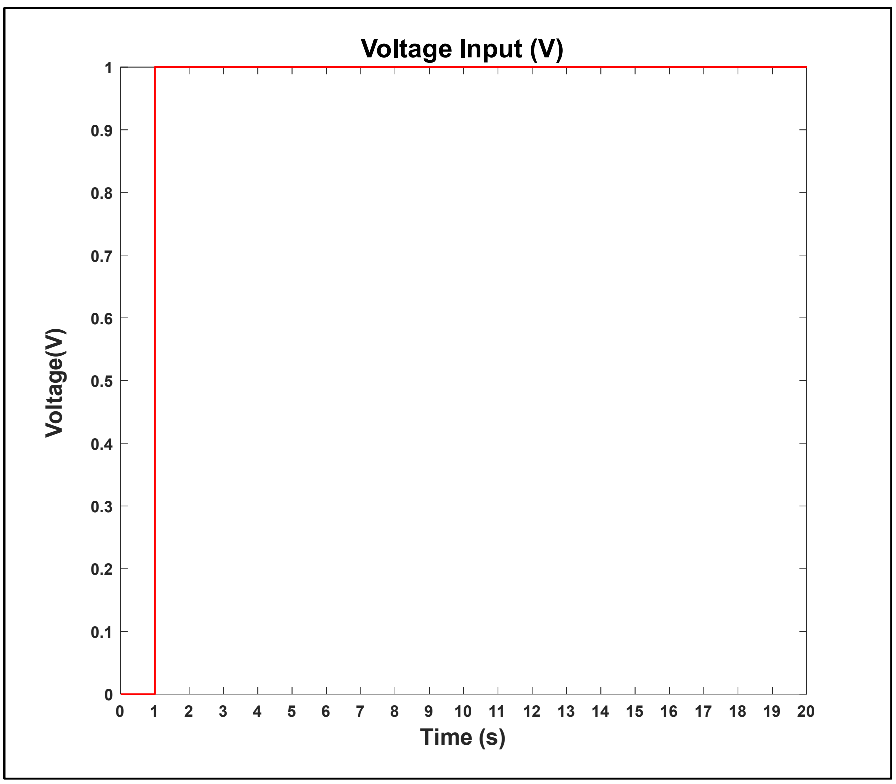
7.3 Simulation Results

After running the MATLAB script above, we get the results from the state space block as four output values we have defined. The graphs of input, position of the cart and rod angle values are plotted below. For this, the below MATLAB script is coded.

```
%% Extract Data for plotting
Position = 1;  % Variable for extracting x1 state of linear model
Velocity = 2;  % Variable for extracting x3 state of linear model
rod_A = 3;  % Variable for extracting x2 state of linear model
rod_Ang_Vel = 4; % Variable for extracting x4 state of linear model
Voltage = 5;  % Variable for extracting Voltage data


%Results
Results.X = SimOut.sdata{Position}.Values.Data(:,1); % Cart position of Lin model
Results.R = SimOut.sdata{rod_A}.Values.Data(:,1); % Rod angle of Lin model
Results.V = SimOut.sdata{Voltage}.Values.Data(:,1); % Voltage of Lin model
Results.Xd = SimOut.sdata{Velocity}.Values.Data(:,1); % Cart velocity of Lin model
Results.Rd = SimOut.sdata{rod_Ang_Vel}.Values.Data(:,1); % Rod angular velocity of
Lin model


%% Plot the results
figure(1);
plot(simPrm.t,Results.V,'r','LineWidth',1);
hold on
xticks(0:1:20);  % Give the x axis ticks
yticks(0:0.1:1); % Give the y axis ticks
title("Voltage Input (V)"); % Give the title
xlabel('Time (s)');  % Give X label
ylabel('Voltage(V)');  % Give Y label


figure(2);
plot(simPrm.t,Results.X,'b','LineWidth',1);
hold on
xticks(0:1:20);  % Give the x axis ticks
yticks(0:0.5:5); % Give the y axis ticks
title("Cart position vs Time for coupled model"); % Give the title
xlabel('Time (s)');  % Give X label
ylabel('Cart position(m)');  % Give Y label


figure(3);
plot(simPrm.t,Results.R,'b','LineWidth',1);
hold on
xticks(0:1:20);  % Give the x axis ticks
title("Rod Angle vs Time for coupled model"); % Give the title
xlabel('Time (s)');  % Give X label
ylabel('Rod Angle(rad)');  % Give Y label
```

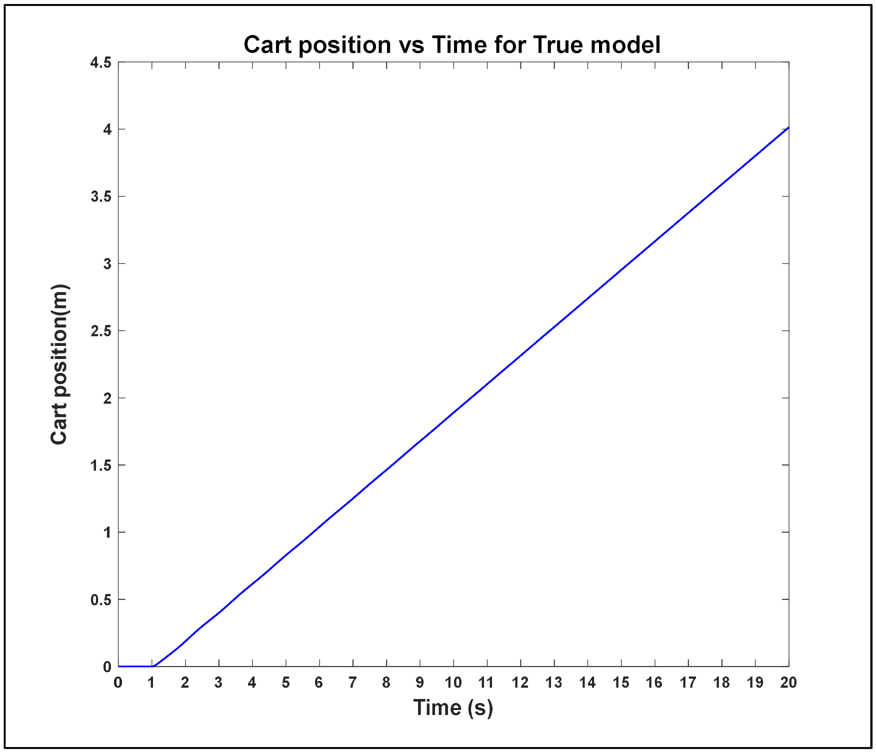
```
figure(4);
plot(simPrm.t,Results.Xd,'b','LineWidth',1);
hold on
xticks(0:1:20);  % Give the x axis ticks
title("Cart Velocity vs Time for coupled model"); % Give the title
xlabel('Time (s)');  % Give X label
ylabel('Cart velocity (m/s)');  % Give Y label


figure(5);
plot(simPrm.t,Results.Rd,'b','LineWidth',1);
hold on
xticks(0:1:20);  % Give the x axis ticks
title("Rod Angular Velocity vs Time for coupled model"); % Give the title
xlabel('Time (s)');  % Give X label
ylabel('Rod Angular Velocity (rad/s)');  % Give Y label
```

7.3.1 Input Voltage vs Time



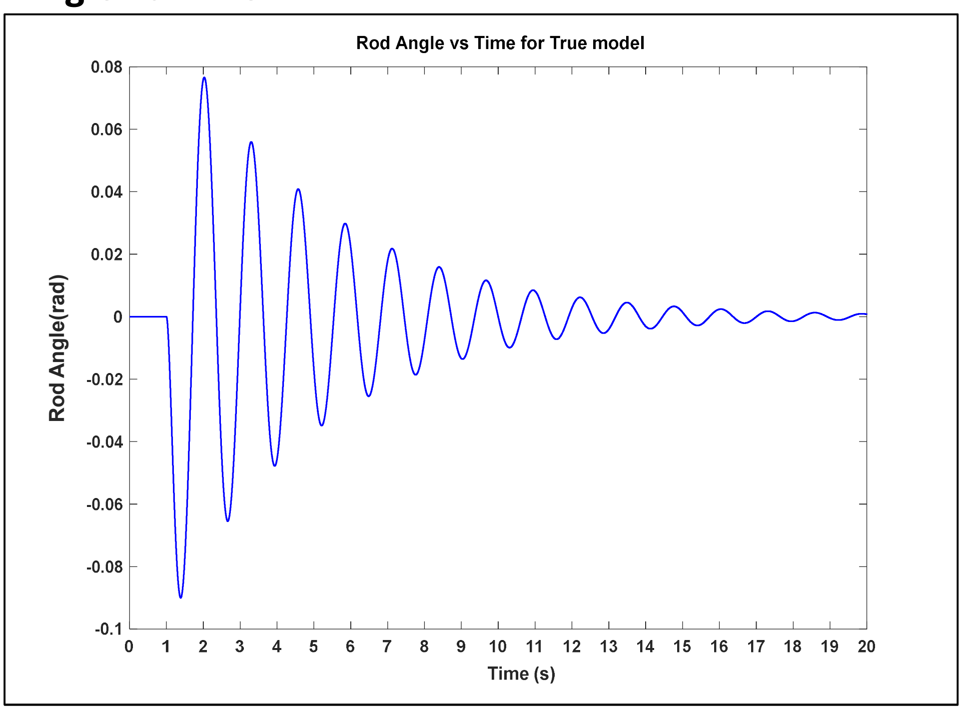
The above plot shows the step input voltage applied as a function of time. At 1 second, the voltage steps us from 0 to 1 Volts. This input is applied to true model in simTrue.slx Simulink file.

7.3.2 Cart Position vs Time



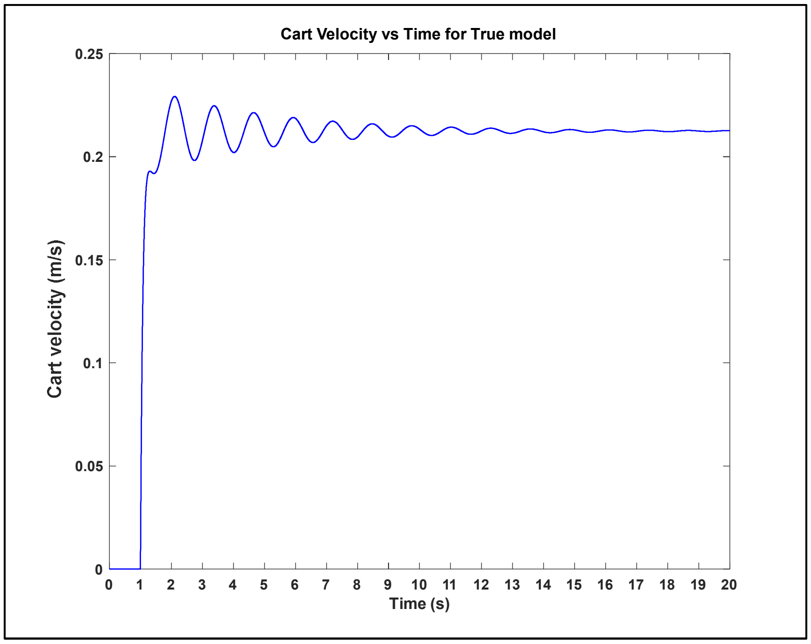
The above plot shows the cart position with respect to time for step input voltage. As long as the voltage is applied, the cart position keeps on increasing.

7.3.3 Rod Angle vs Time



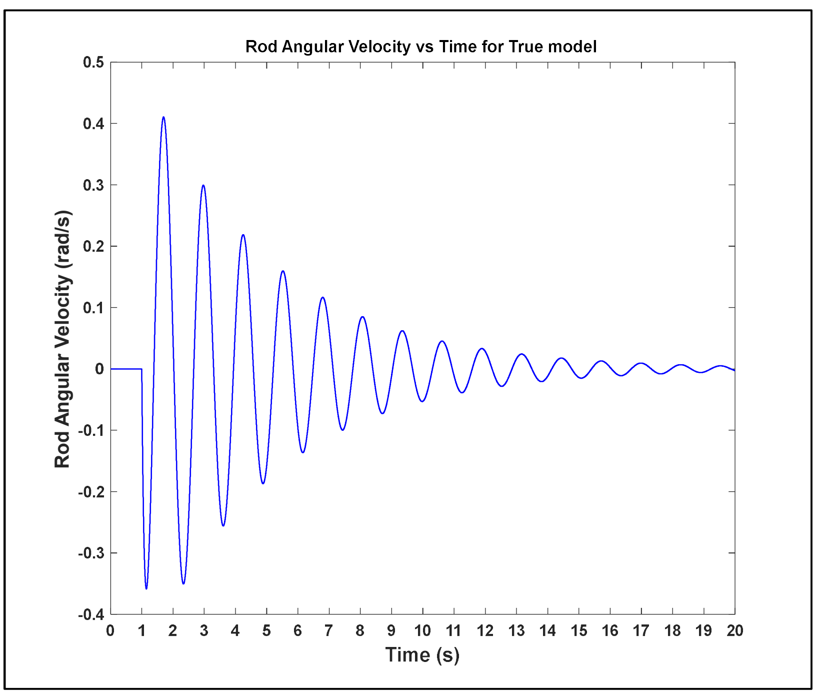
The above plot shows the rod angle with respect to time for step input voltage. Initially rod vibrates because of inertia but when voltage remains constant then vibration reduces and at certain point rod becomes straight down.

7.3.4 Cart Velocity vs Time



The above plot shows the cart velocity with respect to time for step input voltage. As long as the voltage remains constant, the cart velocity remains constant as position increases at constant rate.

7.3.5 Rod Angular Velocity vs Time

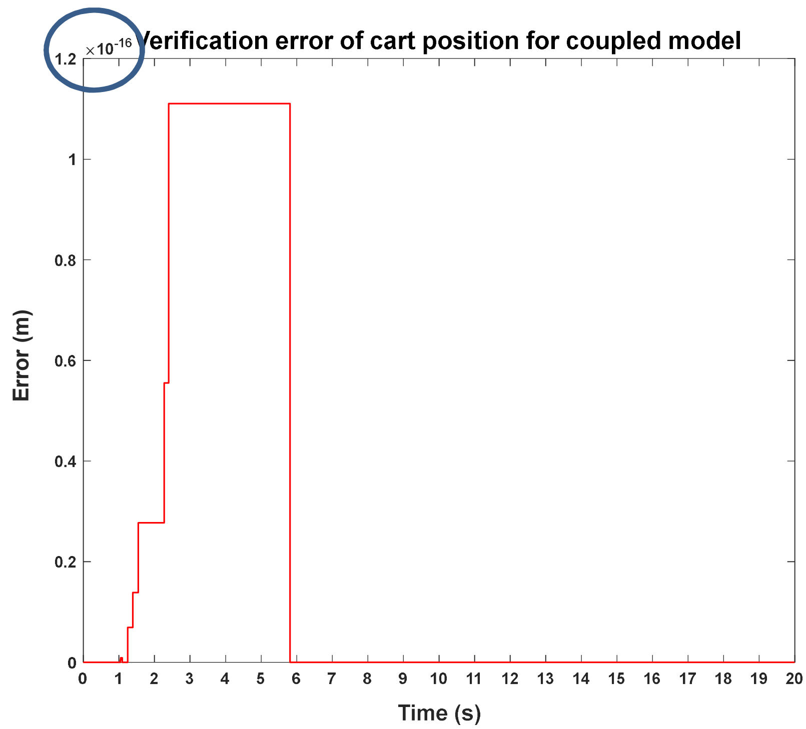


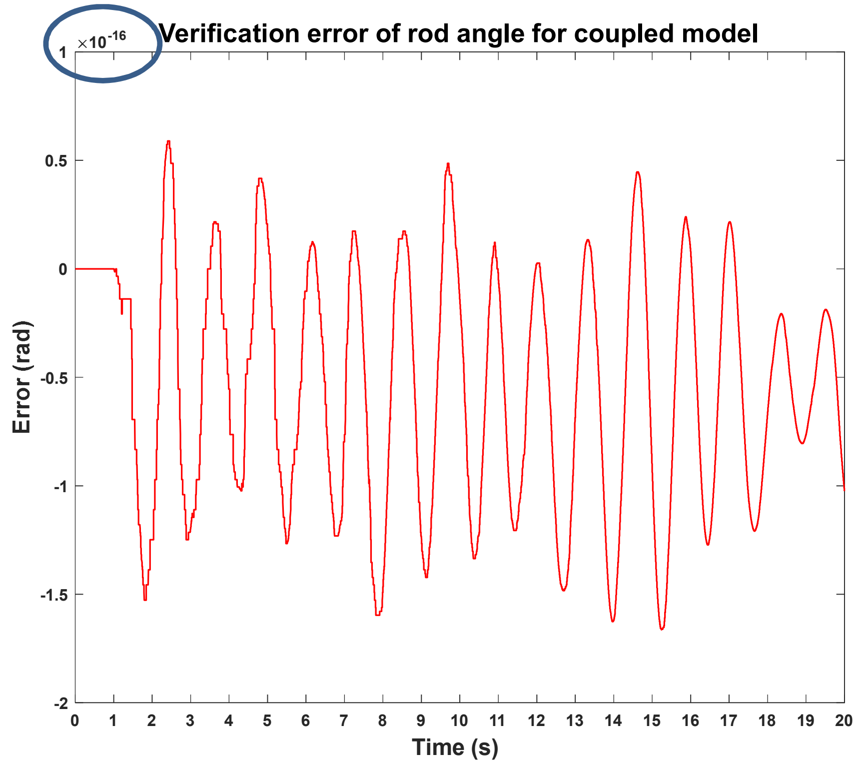
The above plot shows the rod angular velocity with respect to time for step input voltage. Similar to rod angle values, this also stabilizes after certain point.

7.4 Verification

As a part of the project task, we have compared the results with colleague who has performed same simulation in order to check the correctness of the derived differential equation.

Therefore, the data from the colleague has been imported and then it is plotted against our simulation data. However, the comparison is made only for cart position and rod angular velocity. The plots below shows the error between two is plotted against time.





Maximum Error in the position of cart (m)	1.11 e-16
Maximum error in the rod angle (rad)	5.9 e-17

Thus from the error values, it can be said that the model differential equations are derived correctly and model behavior is simulated correctly.

Chapter 8. Decoupled Model Differential Equation

In earlier chapter, we saw that the differential equations derived were coupled to each other through $\ddot{x}$ & $\ddot{\emptyset}$

In Order to design a controller for this system, we will further simplify this model by decoupling the two equations.

Decoupling means, we will remove $\ddot{\emptyset}$ term from cart equation and $\ddot{x}$ term from the rod equation.

After doing that, we get the system equations of the following form.

$$\left(m_T + m_r + \frac{J_w}{r^2}\right)\ddot{x} + \left(\frac{N}{r}\right)^2\left(B_m + \frac{\eta k_t K_e}{R_a}\right)\dot{x} = \frac{N\eta k_t V}{r R_a}$$

$$\left(J_r + m_r L_c{}^2\right)\ddot{\emptyset} + m_r g L_c \emptyset = 0$$

So, in terms of the coefficient we had defined, these equations would become

$$a_1 \ddot{x} + c_1\dot{x} = d V$$

$$b_2\ddot{\emptyset} + b_1 g \emptyset = 0$$

8.1 State Space Modeling & Simulating the Decoupled linear system

There won't be any difference in the state variables and output definitions since we are dealing with same system. Only change would be in the definition of state derivatives.

$$x_1 = x$$
$$x_2 = \emptyset$$
$$x_3 = \dot{x}$$
$$x_4 = \dot{\emptyset}$$
$$u = V$$
$$y = x, \emptyset, \dot{x}, \dot{\emptyset}$$

The derivatives of the states would be

$$\dot{x}_1 = \dot{x} = x_3$$
$$\dot{x}_2 = \dot{\emptyset} = x_4$$

$$\dot{x}_3 = \ddot{x} = \frac{d}{a_1}\, u - \left(\frac{c_1}{a_1}\right) x_3$$

$$\dot{x}_4 = \ddot{\emptyset} = \left(- \frac{b_1\, g}{b_2}\right) x_2$$

$$\dot{x} = f(x,u) = \begin{bmatrix} x_3 \\ x_4 \\ \frac{d}{a_1}\, u - \left(\frac{c_1}{a_1}\right) x_3 \\ \left(- \frac{b_1\, g}{b_2}\right) x_2 \end{bmatrix}$$

$$y = g(x,u) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Again defining, state space Matrices as per the definition in previous chapter

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\left(\frac{c_1}{a_1}\right) & 0 \\ 0 & \left(- \frac{b_1\, g}{b_2}\right) & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{d}{a_1} \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

These matrices have been entered into new script called setupBoth and corresponding Simulink model is developed which is called as simBoth.
Note: The matrices for coupled and decoupled systems are identified by adding suffixes c and d to them. Coupled system matrices won't change.

Running the MATLAB script provides the numerical matrices for decoupled system as below.

```
Ad = 4×4
          0         0    1.0000         0
          0         0         0    1.0000
          0         0   -9.6108         0
          0  -23.4156         0         0

Bd = 4×1
          0
          0
     2.0419
          0

Cd = 4×4
     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1

Dd = 4×1
     0
     0
     0
     0
```

Chapter 9. Proportional controller design on decoupled model

Proportional controller design is a type of feedback controller design where the control action is proportional to the difference between the reference input and the measured output which is called as error. The error is multiplied by a gain which is called as proportional gain K_p.

The design approach for proportional control is as below

If we have a system with transfer function G such that

$$G = \frac{c_2}{s(s + c_1)}$$

then the ratio between input and output in proportionally controlled model can be given as

$$\frac{Y}{U} = \frac{K_p\, c_2}{s(s + c_1)}$$

Comparing this equation with $\frac{\omega_n^2}{s^2+2\,\zeta\,\omega_n\,s}$

We get, $\omega_n^2 = K_p\, c_2$

And $c_1 = 2\,\zeta\,\omega_n$

Where ζ = damping ratio

$\omega_n =$ natural frequency of the system

From the two equations of decoupled model we have, we need to choose the cart equation because that is the only one where input is applied.

$$a_1\, \ddot{x} + c_1 \dot{x} = d\, V$$

Taking Laplace transform on both sides

$$(a_1\, s^2 + c_1 s)\, X = d\, V$$

Thus,

$$\frac{X}{V} = \frac{d}{a_1\, s^2 + c_1 s}$$

Which can be re-arranged as

$$\frac{X}{V} = \frac{\frac{d}{a_1}}{s\,(s + c_1)}$$

If we apply the proportional controller gain to this transfer function then this equation would be modified as

$$\frac{X}{V} = \frac{K_p \frac{d}{a_1}}{s\,(s + c_1)}$$

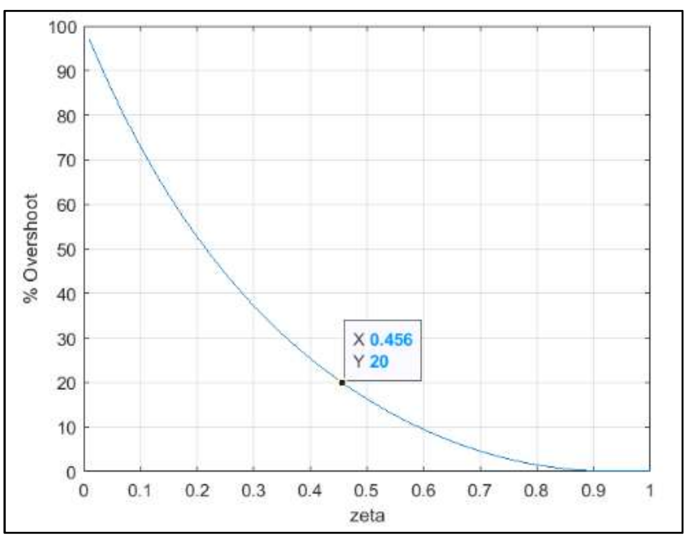
Here, a_1, d, c_1 are the coefficients which we had simplified in previous chapter.

For our system, we have to determine the proportional gain for 20% overshoot.

The relationship between % overshoot and the damping factor is given as

$$\%Mp = e^{\frac{-\pi\,\zeta}{\sqrt{(1-\zeta)^2}}}$$

If we plot the graph of % overshoot against zeta values then we get the graph as below

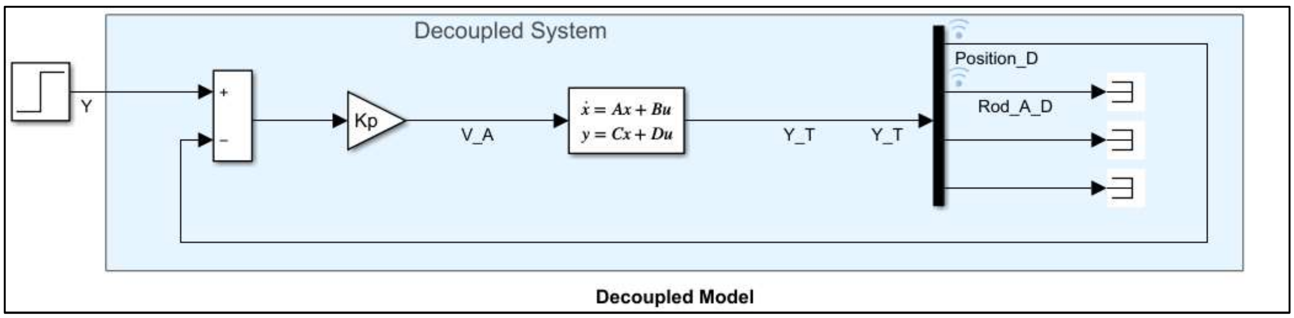


From this, zeta is 0.456.

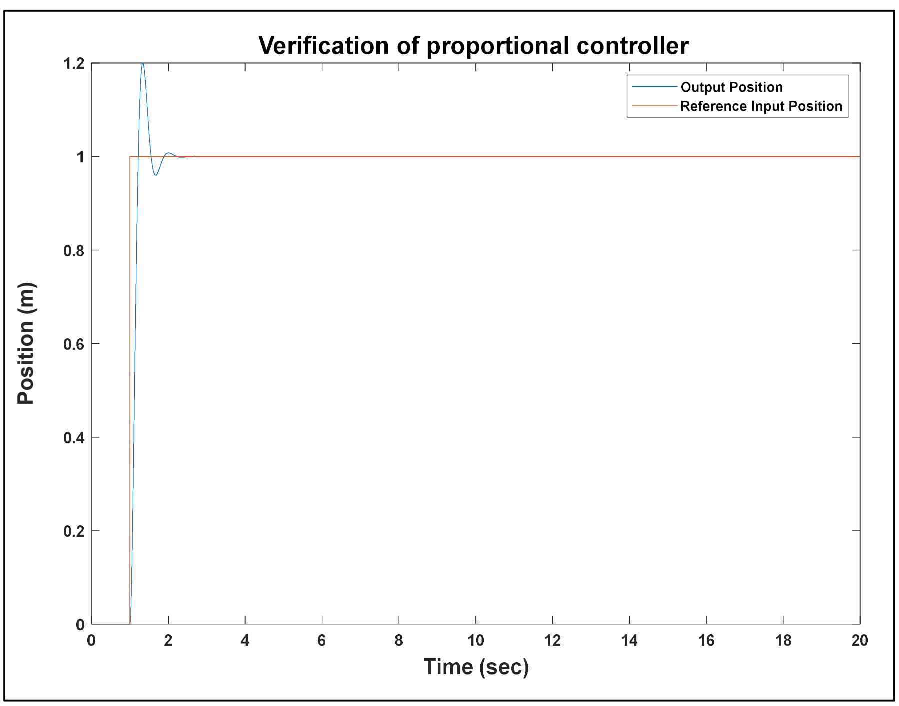
Then we can write the MATLAB command to find the Kp value as below.

```
%% Calculate Kp


Zeta = 0.456;
Omega_n = (Calc.c1/Calc.a1)/(2*Zeta);
Kp = Omega_n^2/(Calc.d/Calc.a1);
```

From this, we get the value of K_p as 54.3876
Let us implement this proportional controller for decoupled model.



After running the simulation, we got below results.



From the figure, it can be seen the position step response has reached to 1.2 m. That means the goal of the 20% overshoot has been achieved. This proportional gain can be used for different inputs now.

Chapter 10. Coupled and Decoupled system response to controlled input

Now as we have calculated and verified the proportional gain in previous chapter, we applied same gain to coupled and decoupled system and checked the performance. We compared the system response for two different inputs.

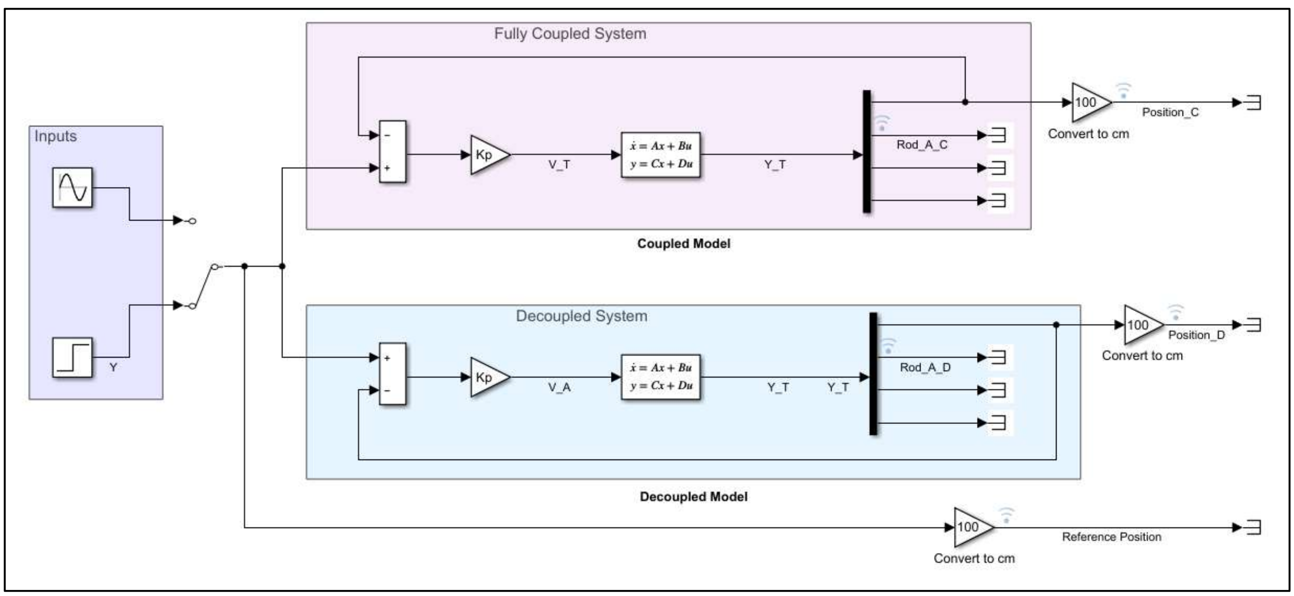
Input 1 – Step command of 30 cm
Input 2 – Sine wave input of amplitude 30 cm and frequency 1 rad/s

The same MATLAB code is continued for simulating both systems under proportional control.

```
%% Simulation parameters
simPrm.solTyp = 'Fixed-step';  % Solver type
simPrm.sol    = 'ode3';  % Solver type 2
simPrm.dt  = 0.001; % Integration step size
simPrm.tEnd = 20; % Simulation end time
simPrm.t = 0:simPrm.dt:simPrm.tEnd;  %simulation time goes from 0 to 20


%% Simulate the math model
set_param('simBoth','SolverType',simPrm.solTyp);  % set this solver type in
simulink parameters
set_param('simBoth','Solver',simPrm.sol);     % set this integration method in
simulink solver
SimOut = sim('simBoth','SignalLoggingName','sdata'); % Simulate the math model and
save the data


% Extract Data for plotting
Reference_Position = 1;  % Variable for extracting reference input
Position_D = 2;  % Variable for extracting x3 state of linear model
Position_C = 3;  % Variable for extracting x2 state of linear model
Rod_A_C = 4; % Variable for extracting x4 state of linear model
Rod_A_D = 5;  % Variable for extracting Voltage data


%Results
Results.Xd = SimOut.sdata{Position_D}.Values.Data(:,1); % Cart position of Lin
model
Results.Rd = SimOut.sdata{Rod_A_D}.Values.Data(:,1); % Rod angle of Lin model
Results.Xc = SimOut.sdata{Position_C}.Values.Data(:,1); % Voltage of Lin model
Results.Rc = SimOut.sdata{Rod_A_C}.Values.Data(:,1); % Cart velocity of Lin model
Results.Ref = SimOut.sdata{Reference_Position}.Values.Data(:,1); % Rod angular
velocity of Lin model
```

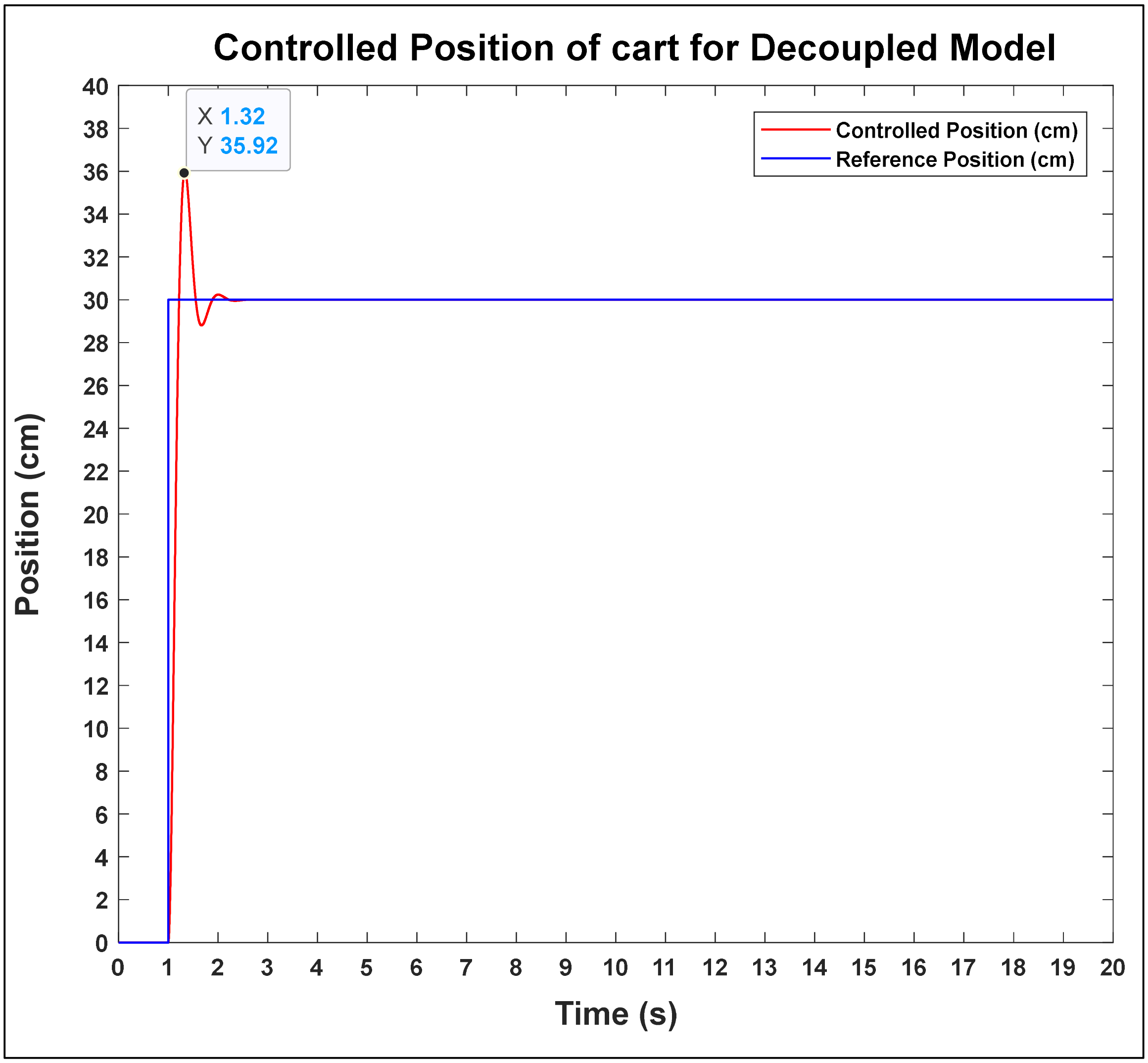
```
%% Plot the results
figure(1);
plot(simPrm.t,Results.Xd,'r','LineWidth',1);
hold on
plot(simPrm.t,Results.Ref,'b','LineWidth',1);
xticks(0:1:20);  % Give the x axis ticks
title("Controlled Position of cart for Decoupled Model"); % Give the title
xlabel('Time (s)');  % Give X label
ylabel('Position (cm)');  % Give Y label
legend('Controlled Position (cm)','Reference Position (cm)'); % legend


figure(2);
plot(simPrm.t,Results.Xc,'r','LineWidth',1);
hold on
plot(simPrm.t,Results.Ref,'b','LineWidth',1);
xticks(0:1:20);  % Give the x axis ticks
title("Controlled Position of cart for coupled Model"); % Give the title
xlabel('Time (s)');  % Give X label
ylabel('Position (cm)');  % Give Y label
legend('Controlled Position (cm)','Reference Position (cm)'); % legend
```

The Simulink model is modified which is shown as below.



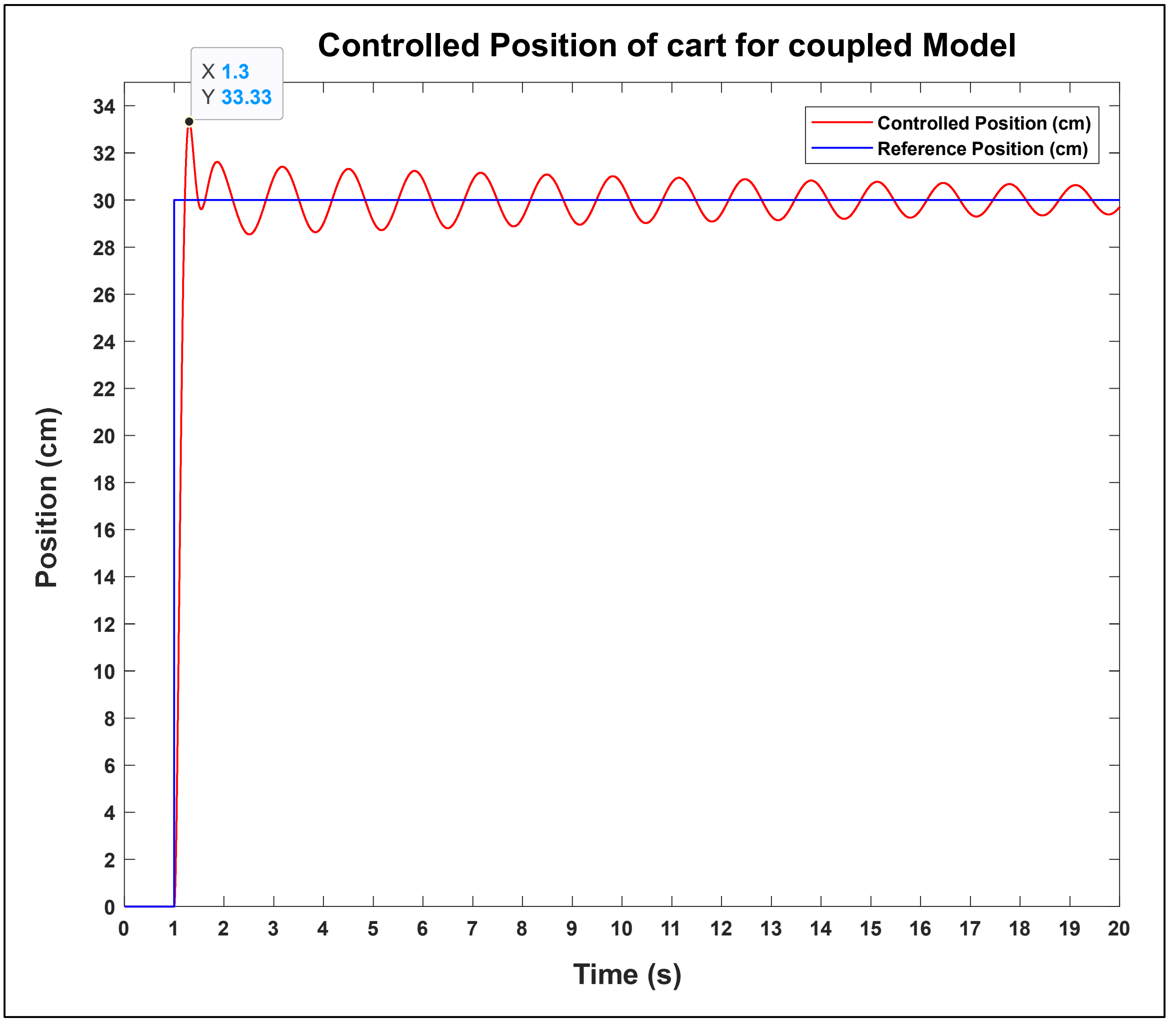
10.1 Simulation Results for Step input of 30 cm

10.1.1 Controlled position of cart for decoupled model



The above figure shows the graph of controlled position and reference position of 30 cm step for decoupled model based on the designed proportional controller.

10.1.2 Controlled position of cart for coupled model



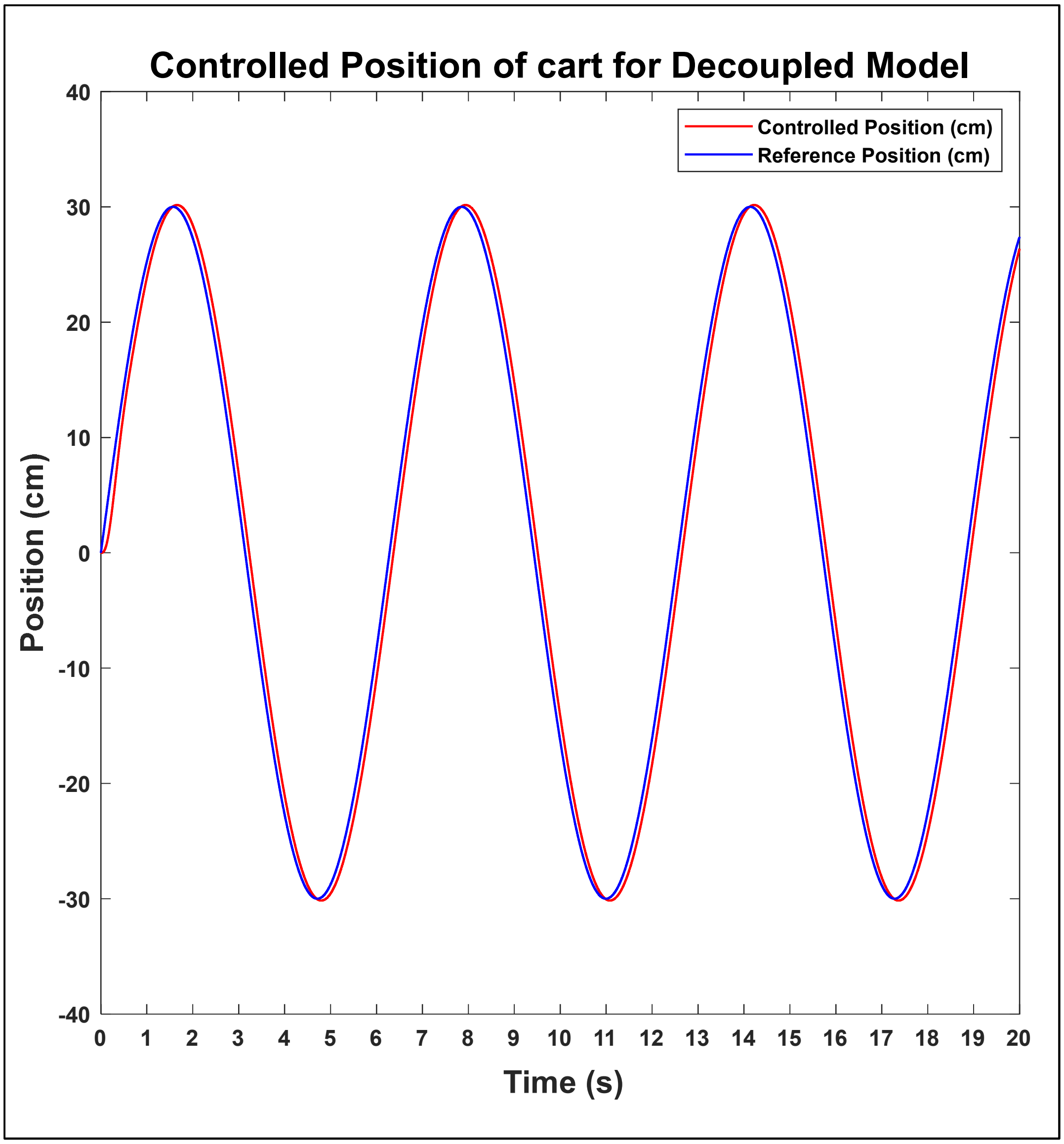
The above figure shows the graph of controlled position and reference position of 30 cm step for coupled model based on the designed proportional controller.

10.1.3 Conclusion from simulation results of step input

If we observe two figures above, then we can see that proportional controller behaves very well for the decoupled model and it should be because we have designed the controller on that model. When we apply same controller to coupled model then we can see that there are certain oscillations in the output position and the system is taking time to reach to steady state. However, we can say that system is behaving good in achieving the controlled position. The controller needs to be further tuned for reducing the settling time.

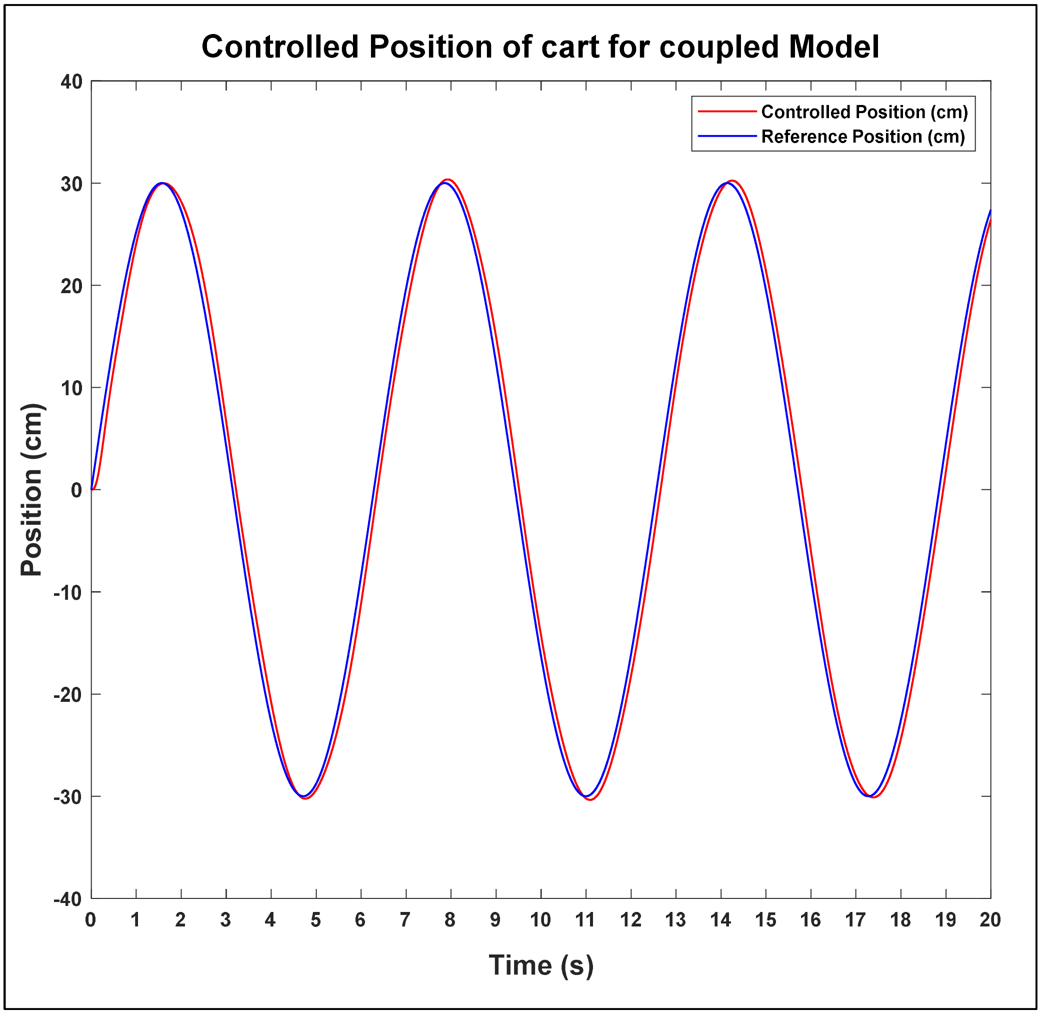
10.2 Simulation Results for Sine input of 30 cm amplitude

10.2.1 Controlled position of cart for decoupled model



The above figure shows the graph of controlled position and reference position of 30 cm sine wave with frequency of 1 rad/sec for decoupled model based on the designed proportional controller.

10.2.2 Controlled position of cart for coupled model



The above figure shows the graph of controlled position and reference position of 30 cm sine wave with frequency of 1 rad/sec for coupled model based on the designed proportional controller.

10.1.3 Conclusion from simulation results of sinewave input

If we observe two figures above, then we can see that proportional controller behaves very well for the decoupled as well as coupled model. At this point, we can say that the controller which is designed based on decupled model is behaving well for the coupled model as well.

Chapter 11. Sine wave input with frequency same as the frequency of the swinging rod

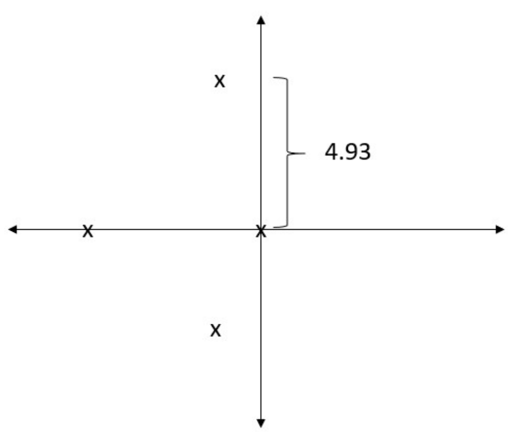
Furthermore, we changed the sinewave input frequency which would be equal to the frequency of swinging rod.

If we calculate the eigen values of A matrix in coupled system, we get following values.

0.0000 + 0.0000i
-0.2468 + 4.9319i
-0.2468 - 4.9319i
-11.9325 + 0.0000i

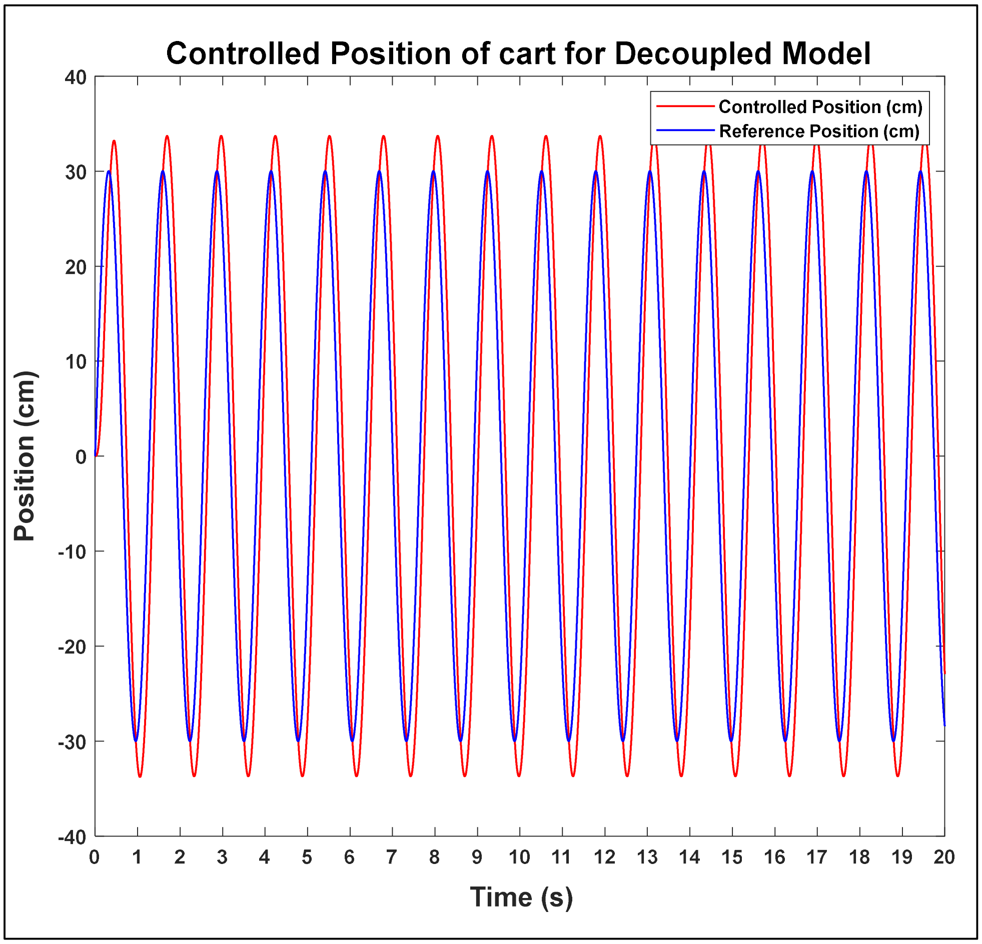
The eigen values of A matrix are nothing but the poles of transfer function.

Thus, there are two poles which are located at a distance of 4.9319 from real axis along imaginary axis. This value is nothing but the frequency of the swinging rod. We will use this frequency for the sine wave input now and check the results.



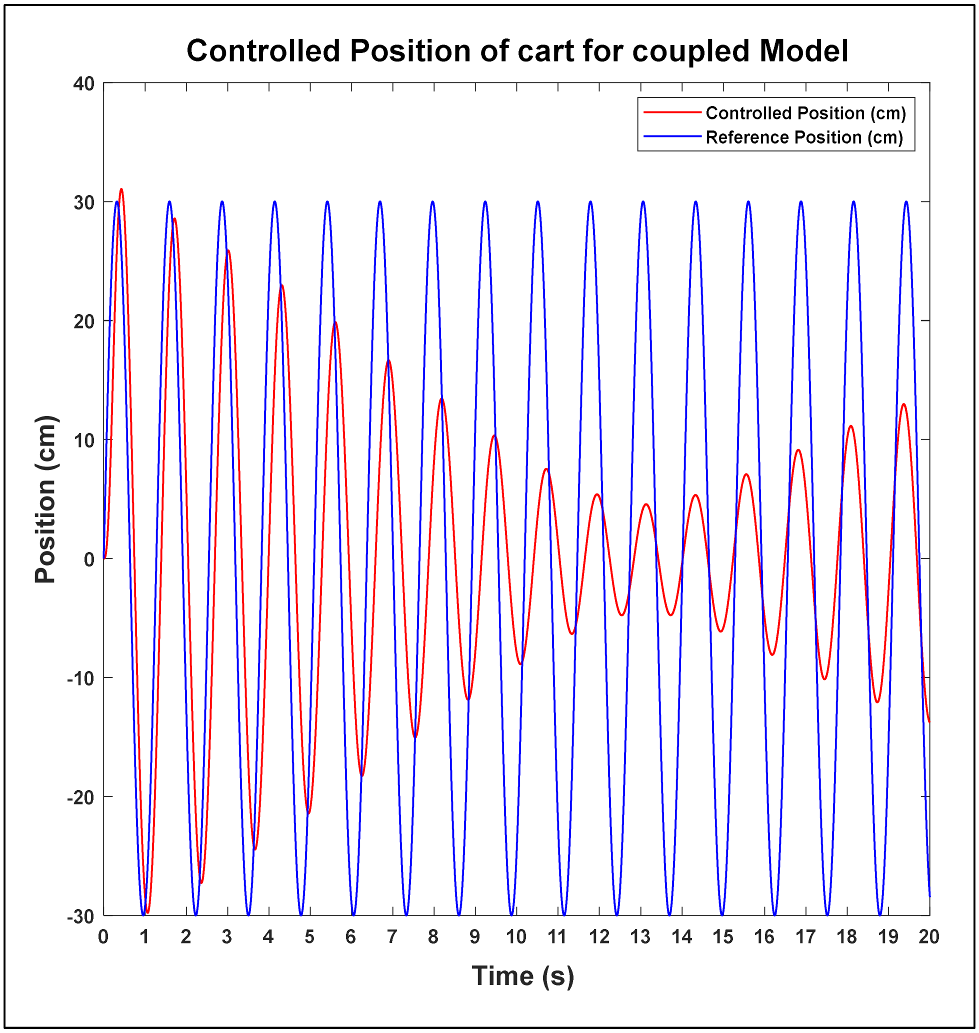
11.1 Simulation Results

11.1.1 Controlled position of cart for decoupled model



The above figure shows the graph of controlled position and reference position of 30 cm sine wave with frequency of swinging rod for decoupled model based on the designed proportional controller.

11.1.2 Controlled position of cart for coupled model



The above figure shows the graph of controlled position and reference position of 30 cm sine wave with frequency of swinging rod for coupled model based on the designed proportional controller.

11.2 Conclusion from simulation results

If we observe two figures above, then we can see that proportional controller is not behaving good for coupled model and also for decoupled model the magnitude of the controlled position is bigger than the reference position input.

References

1. Canvas Notes-MEEM 4775 – Dr.Gordon Parker, Michiagn Technological University

2. System Dynamics By Katsuhiko Ogata

3. Feedback Control of Dynamic Systems by Gene Franklin, J David-Powell, Abbas Emami-Naeini