ROCO218 CONTROL ENGINEERING 2018
ASSESSED COURSEWORK PART 1: SFC OF AN INVERTED PENDULUM

IMPORTANT NOTICE

Submit this report to the DLE by: 17th May 2018 (4pm)

Please only submit a single report for this coursework in PDF format. You **must** include your student number in a header on every page of the report and **also include** your student number **in the document filename**! Please use the supplied template format for your report and importantly, please use the section heading used in this coursework sheet.

This coursework should contain a few pages of explanation as required (although it can be more than this is necessary) and a set of images showing the plots requested by the individual parts of the practical exercises, as well as embedded equations and Matlab code that you developed to implement your solutions.

The report ***must*** be a single **stand-alone document**. Please embed images in-line in the report. In order to show video demonstrations if you have any, please use Internet links to videos that you have been uploaded to YouTube.

You can work in pairs for the practical sessions but submit a report **individually**!

Introduction

This assignment system involves the analysis of an inverted pendulum mounted to a motorized cart. Such a system is unstable without control. Balance in the inverted configuration can only be achieved by displacing the cart.

This first part of the coursework is to give you experience of:

- Deriving the state space model of an inverted pendulum
- Designing a direct state feedback controller to stabilize the pendulum in its inverted configuration
- Designing state feedback control using state estimated using an observer.
- Implementing a controller simulation that runs in Matlab.
- Implementing a controller solution using Euler integration that could easily be ported to run on a microcontroller.

[70 marks]

The second part of the coursework is to give you experience of implementing a state feedback controller and observer to control a real inverted pendulum:

- Augment the state space model to include a state to represent cart position and redesign the state feedback controller gains using this extra state
- Implement the augmented system using Euler integration so it can easily be ported to run on a microcontroller.
- Implement the augmented system using an Arduino Mega microcontroller to balance a real inverted pendulum

[30 marks]

Example Matlab simulation

You have been provided an example of the simulation of an uncontrolled non-linear inverted pendulum that makes use of the Matlab ode45 function.

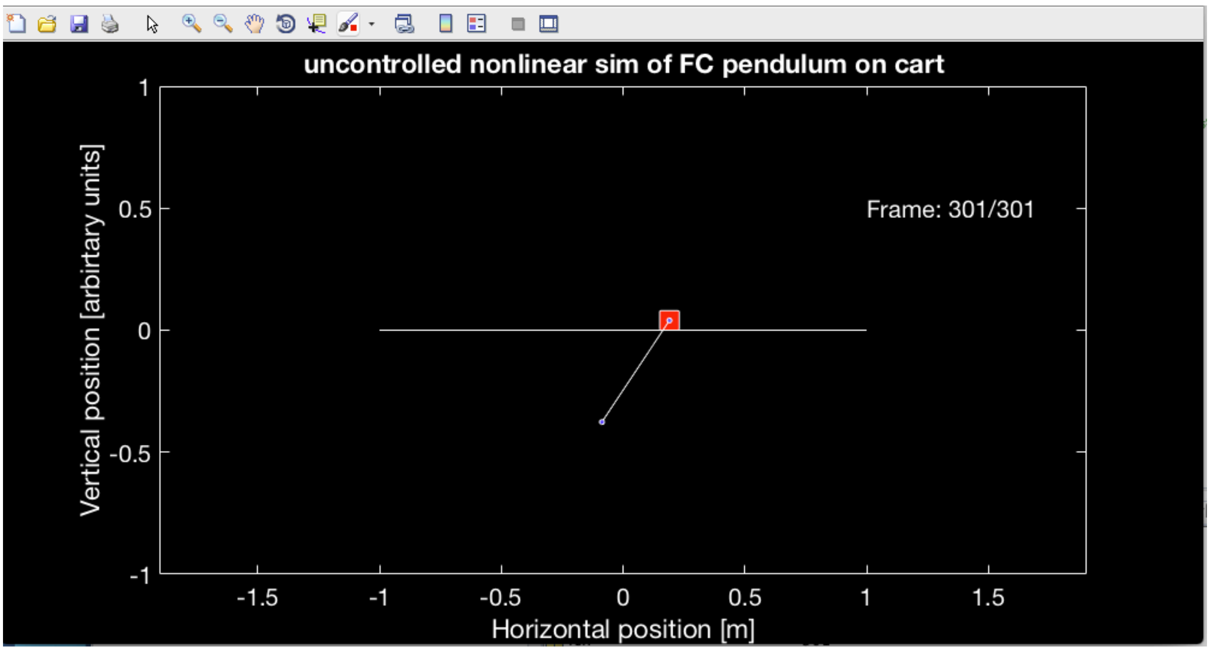
Run the script

Main_simFCOde45.m

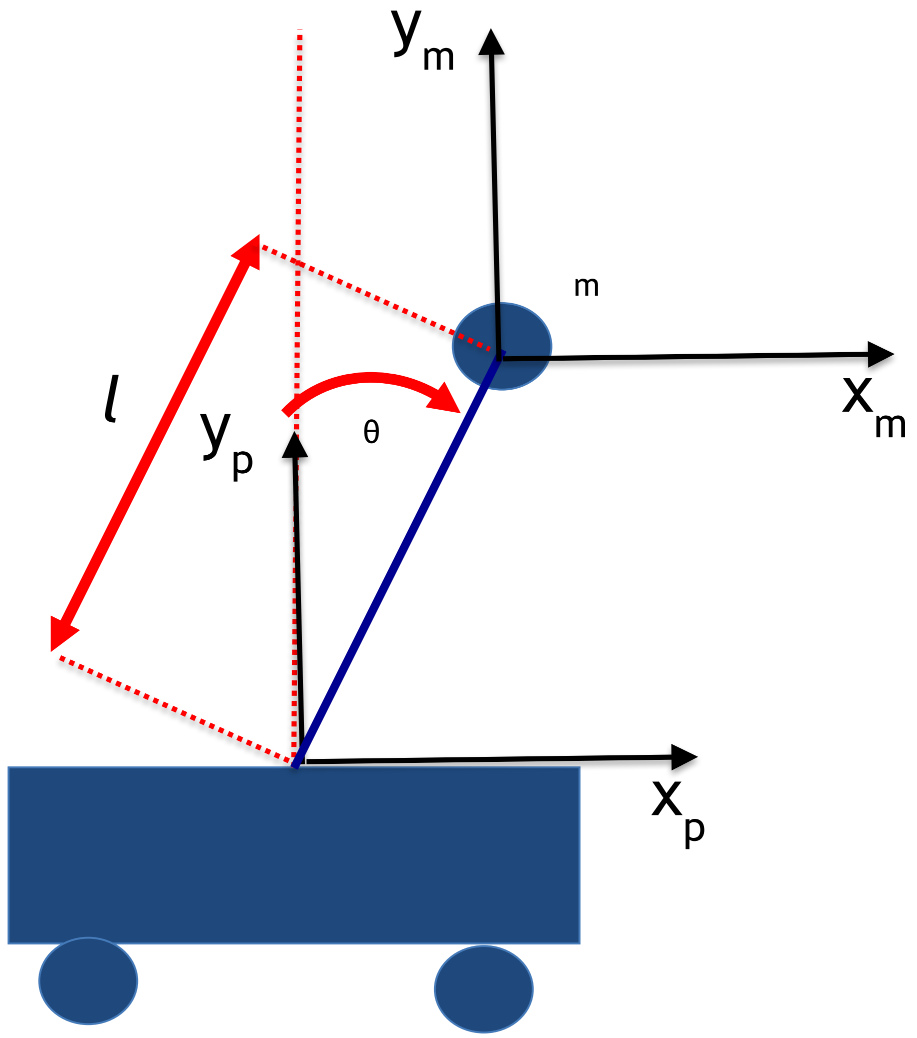
This uses ode45 to solve the non-linear differential equations that model an inverted pendulum on a cart.

The call-back function is given in the function file FCPendOnCart.m

After the simulation is over, it writes its computed output to the display function AnimatePendulumCart.m. A screenshot of his output is given below:



Differential equations to represent an inverted pendulum



Consider an inverted rod pendulum that will be stabilized by controlling cart velocity.

- The mass of the pendulum is denoted by m
- The length to the centre of mass is denoted by l
- The angle to the vertical is denoted by θ
- The displacement of the pivot is given by u_p
- The coefficient of viciously is denoted by μ
- The moment of inertial of the pendulum rod about the centre of mass is denoted by I
- Acceleration due to gravity is denoted by g

Balancing the torques around the centre of mass of the pendulum leads to the following equation of motion:

$$\left(I+ml^2\right)\frac{d^2\theta}{dt^2}+\mu\frac{d\theta}{dt}=mgl\sin(\theta)+ml\cos(\theta)\frac{d^2x}{dt^2}$$

Note that before performing numerical evaluations at critical stages in this coursework, you should substitute the following parameter values into your derivation:

- Length of pendulum rod 0.64 m
- Mass of pendulum rod 0.314 kg
- Coefficient of viscous friction of 0.05

1. Linearize the non-linear differential equation

[10 marks]

- Find the unstable equilibrium point and linearize the equation around this point.
- Do so by computing the Jacobian matrix for the system and evaluating it at the equilibrium point.
- Show that this leads to the equation

$$\left(I+ml^2\right)\frac{d^2\theta}{dt^2}+\mu\frac{d\theta}{dt}=mgl\theta+ml\frac{d^2x}{dt^2}$$

2. Write down the state space model of the system

[10 marks]

Now we move on to control!

- The pendulum control input is to be cart velocity.
- The output from the state space model is pendulum angle.
- Choose suitable states such as

$$x_1 = \theta$$
$$x_2 = \frac{d\theta}{dt} - b_0 v_c$$

where b_0 is a constant and v_c is the velocity control input

- Derive a state space model from the linearized differential equation of the system and write down the state space model in matrix form.
- For the stable equilibrium, show that this leads to the equations:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{mgl}{\left(I+ml^2\right)} & -\frac{\mu}{\left(I+ml^2\right)} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \frac{ml}{\left(I+ml^2\right)} \\ \frac{-\mu ml}{\left(I+ml^2\right)^2} \end{bmatrix} v_c$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

- Substitute in the pendulum parameter values and numerically evaluate these terms in Matlab.

3. Observability, controllability and stability

[10 marks]

Show that the uncontrolled pendulum system is

- Controllable
- Observable
- Has an unstable configuration.

- NB: You will need to examine both equilibrium points to show the latter.

4. Simulate your state space module using the Matlab ode45 function

[10 marks]

- Use the Matlab differential equation solver ode45 to solve the state space model of the uncontrolled pendulum.
- You have been provided with a Matlab function AnimatePendulumCart.m to animate the pendulum.
- NB: You will need to set the position input on this function to zero since the cart is not moving.
- In addition, plot the angle of the pendulum and the velocity of the cart to generate time waveform plots.
- Start the pendulum off in an upright position and introduce noise in the model so it loses its balance location.
- Make a YouTube video of the simulation and provide a link to it in your report.

5. Design a state feedback controller

[10 marks]

- First write down the matrix equation for state feedback control, indicating the form of the gain vector.
- Then write down the equation for state feedback control for the pendulum.
- Select suitable pole locations and using pole placement in Matlab and explain why you chose them.
- Design a state feedback controller to stabilise the inverted pendulum system.
- Use the place command in Matlab to find the control gains.
- Implement the SFC controller using the Matlab ode45 command.

6. Implement the controller system using Euler integration

[10 marks]

- Instead of using ode45, now write your own Matlab function to solve your state space model system.
- Use Euler integration to achieve this and apply it iteratively in a loop.
- Use a C-language like style for the computations so your solution will later provide an implementation that could easily be ported onto a microcontroller (such as the Arduino Mega) and used to control a real inverted pendulum.
- Show that the system performs just as well as before.

7. Add a Luenberger observer to your state feedback controller

[10 marks]

- Add an observer to the controller so it makes use of feedback from state estimates, rather than directly from the system state.
- Use the place command in Matlab to design the observer gains.
- Implement the SFC observer-controller using Euler integration.
- Show that the system behaves as it did before.