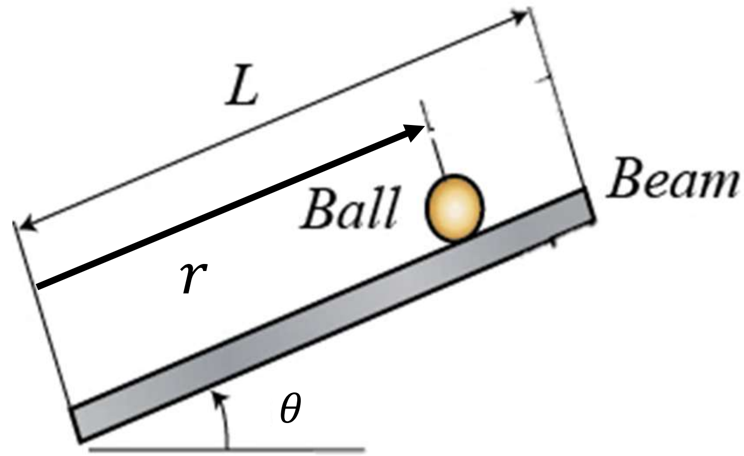
Drones and Autonomous Systems Laboratory
Ball balancing on the beam class 2

-Ball and Beam system dynamics-

DONGBIN KIM –
Ph.D. Candidate
Mechanical Engineering,
University of Nevada, Las Vegas

1. Introduction – scheme

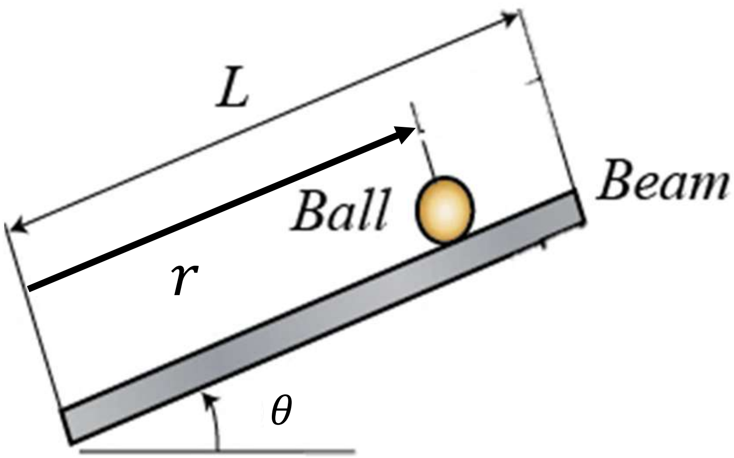
Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV



(Ball and Beam Model)

2. Equation of Motion

Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV



(Ball and Beam Model)

(Equation of Motion)

$$\left(\frac{J}{R^2}+M\right)\ddot{r}+mg\theta=0$$

(State Space Representation)

$$[\dot{x}]=\begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}[x]+\begin{bmatrix}0\\ mb*\dfrac{g}{\left(\dfrac{J_b}{r_b^2}+m_b\right)}\end{bmatrix}[u]$$

Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV

Lagrangian Method

1. Newton – Euler method is a "force balance" approach to dynamics
2. Lagrangian method is an "energy based" approach to dynamics
3. The Lagrangian L is defined as the followings

$$L = K - P \ (K = kinetic\ energy, P = Potential\ Energy)$$

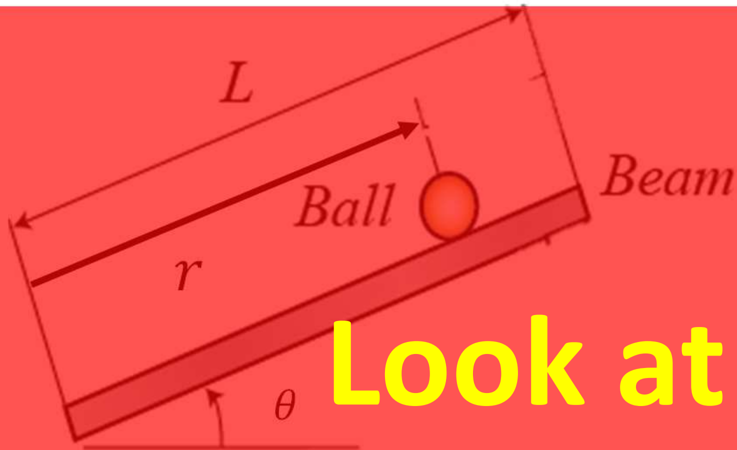
4. The dynamics equations, in terms of the coordinates used to express the kinetic and potential energy are obtained as

$$F_i = \frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i}$$

Where q_i is the coordinates in which the kinetic and potential energy are expressed, $\dot{q}_i$ is the corresponding velocity, and F_i the corresponding force or torque

2. Equation of Motion - Derivation

Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV



(Ball and Beam Model)

(Equation of Motion)

$$\left(\frac{J}{R^2} + M\right)\ddot{r} + mg\theta = 0$$

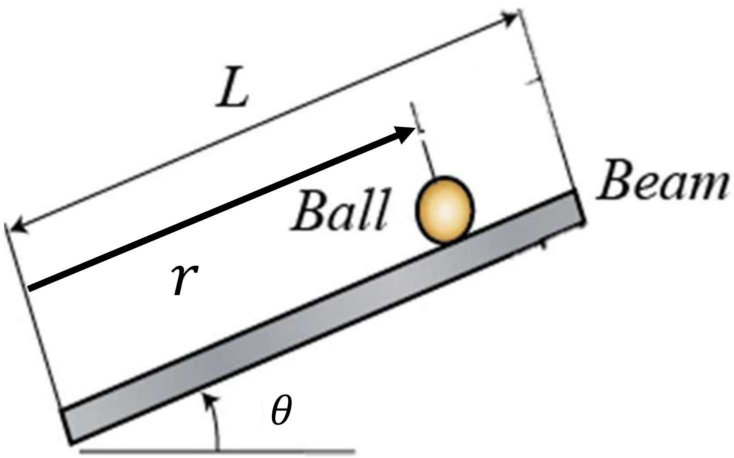
Look at the whiteboard

(State Space Representation)

$$[\dot{x}] = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}[x] + \begin{bmatrix} 0 \\ mb * \dfrac{g}{\left(\dfrac{J_b}{r_b^2} + m_b\right)} \end{bmatrix}[u]$$

3. Simulation - MATLAB

E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV



(Ball and Beam Model)

$$[\dot{x}] = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} [x] + \begin{bmatrix} 0 \\ mb * \dfrac{g}{\left(\dfrac{J_b}{r_b^2} + m_b\right)} \end{bmatrix} [u]$$

```
mb = 0.004; %mass of the ball yello : 0.002
rb = 0.024; %radius of the ball
Jb = (2*mb*rb^2)/3; %MOI of the ball
g=9.81; %Gravitional Acceleration.

MatrixA = [0 1;0 0];
MatrixB = [0;mb*g/(Jb/(rb^2)+mb)]
MatrixC = [1 0];
```

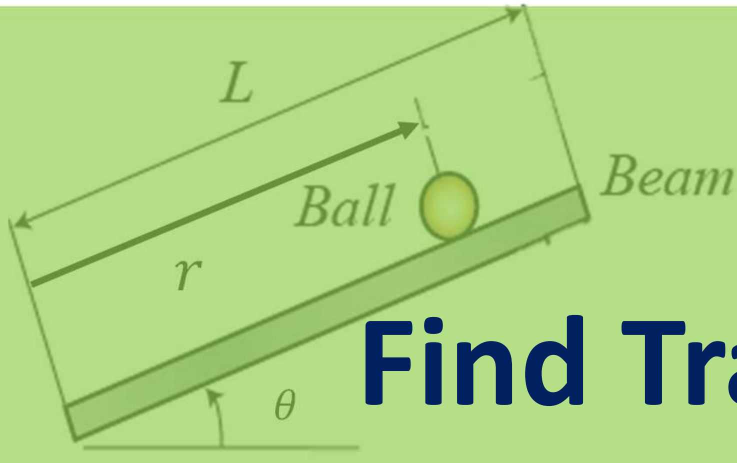
(Equation of Motion in State Space Representation)

Drones and Autonomous Systems Lab @ UNLV

3. Simulation – Transfer function

Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV

Find Transfer Function



(Ball and Beam Model)

$$[\dot{x}] = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} [x] + \begin{bmatrix} 0 \\ mb * \dfrac{g}{\left(\dfrac{J_b}{r_b^2} + m_b\right)} \end{bmatrix} [u]$$

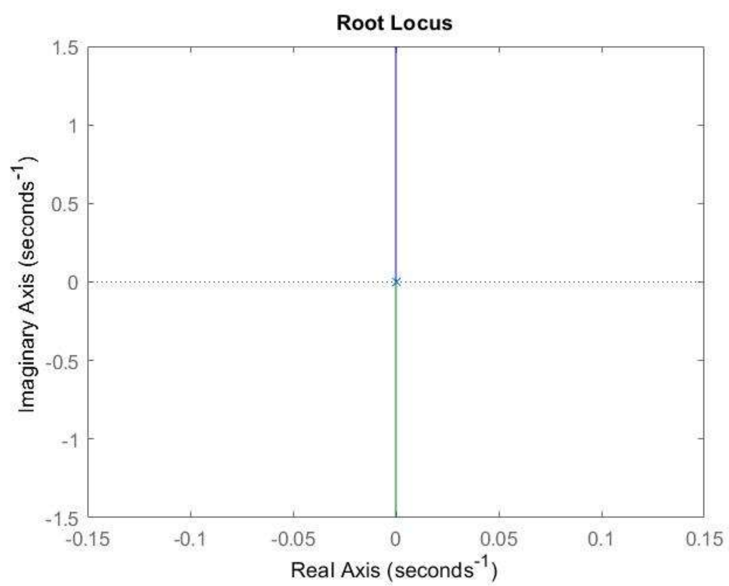
```
mb = 0.004; %mass of the ball yello : 0.002
rb = 0.024; %radius of the ball
Jb = (2*mb*rb^2)/5; %Moment of the ball
g=9.81; %Gravitional Acceleration.

MatrixA = [0 1;0 0];
MatrixB = [0;mb*g/(Jb/(rb^2)+mb)]
MatrixC = [1 0];
```

(Equation of Motion in State Space Representation)

3. Simulation – Transfer Function

Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV



Transfer Function,
Root Locus Results

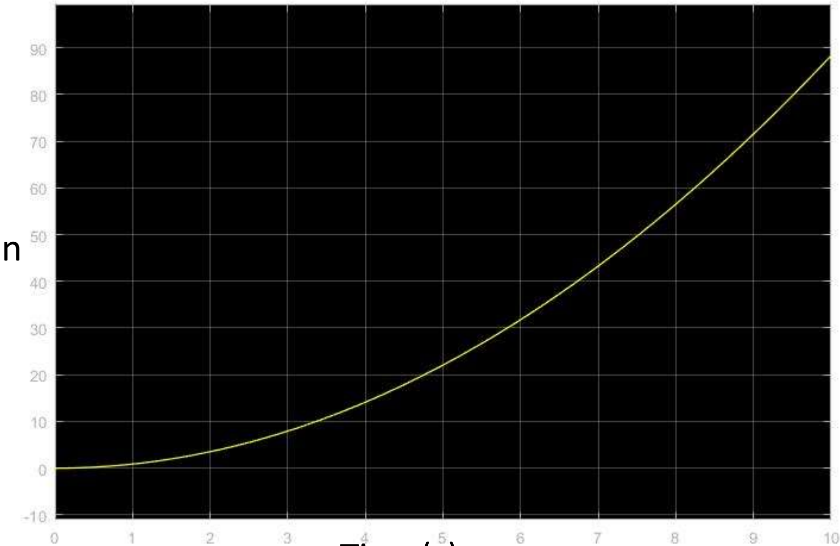
1. Divergence response results
2. Two zero Poles
3. Step Response -> Divergence
4. How we are going to make the system stable?
5. PID? Full state-feedback?

Drones and Autonomous Systems Lab @ UNLV

3. Simulation – Transfer Function

Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV

Ball Position (m)



Time(s)

(Step Response)

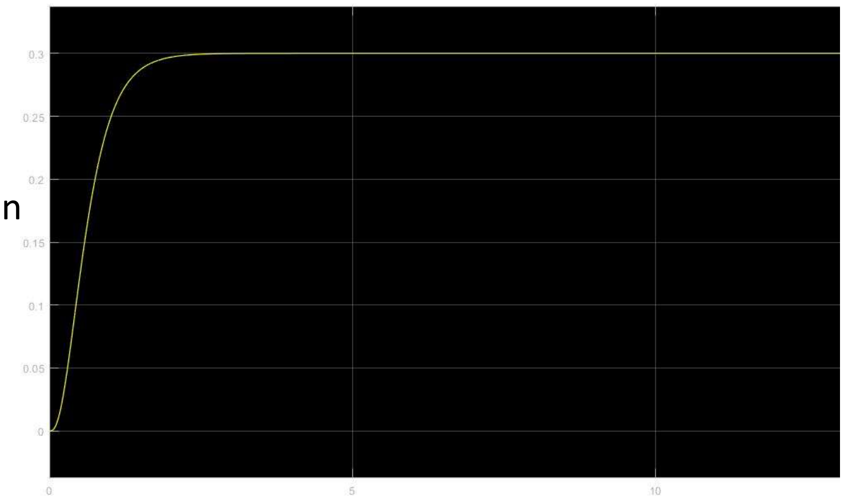
Transfer Function, Root Locus Results

1. Divergence response results
2. Two zero Poles
3. Step Response -> Divergence
4. How we are going to make the system stable?
5. PID? Full state-feedback?

Drones and Autonomous Systems Lab @ UNLV

3. Simulation – Transfer Function

E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV



(PID Compensated Results)

Transfer Function,
Root Locus Results

1. Divergence response results
2. Two zero Poles
3. Step Response -> Divergence
4. How we are going to make the system stable?
5. PID? Full state-feedback?

4. Homework

Copyright : Dongbin Kim.
E-mail : dongbin.kim@unlv.edu
Drones and Autonomous Systems Lab, UNLV

Homework for next course

1. Using Newton method to derive equation of motions
2. Create your own PID compensator to stabilize your system, show the plots, and Simulink or MATLAB code

Drones and Autonomous Systems Lab @ UNLV