Optimal Control Project Proposal

Ball and Beam

Baraghini Nicholas
Curto Fabio
Iadarola Federico

Group 21

University of Bologna
Automation Engineering Master Degree course

Contents

1 Introduction

The ball and beam system is one of the most enduringly popular and important laboratory models for teaching control systems engineering. The ball-and-beam is widely used because it is very simple to understand as a system, and yet the control techniques that can be studied cover many important classical and modern design methods. It has a very important property; it is open-loop unstable.

The system is very simple, a steel ball rolling on the top of a long beam. The beam is mounted on the output shaft of an electric motor and so the beam can be tilted about its center axis by applying an electrical control signal to the motor amplifier. The control job is to automatically regulate the position of the ball on the beam by changing the angle of the beam. This is a difficult control task because the ball does not stay in one place on the beam but moves with acceleration that is approximately proportional to the tilt of the beam. In control terminology, the system is open-loop unstable because the system output (the ball position) increases without limit for a fixed input (beam angle). Feedback control must be used to stabilize the system and to keep the ball in a desired position on the beam.

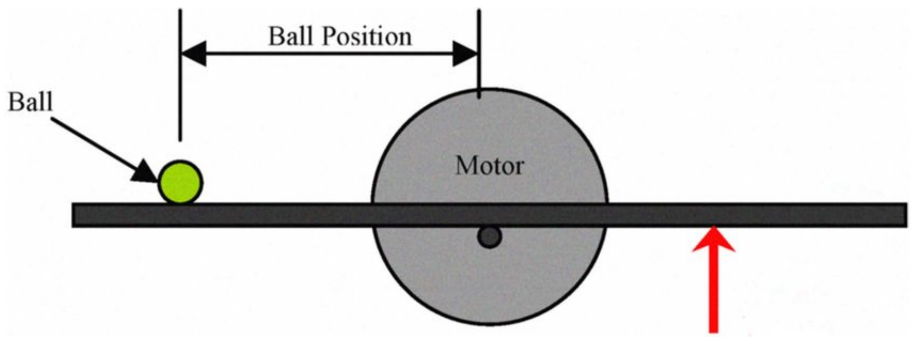
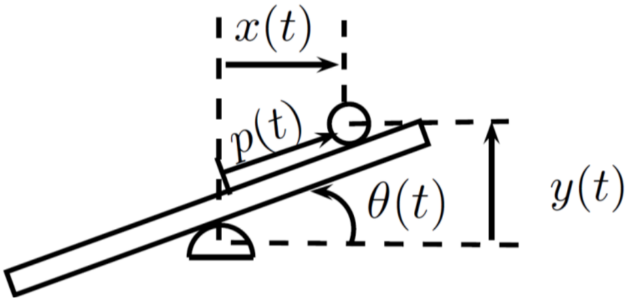


Figure 1: Ball and Beam Diagram

2 Mathematical Modeling

2.1 Lagrangian Method

The control objective is to control the torque τ applied at the pivot of the beam, such that the ball can roll on the beam and track a desired trajectory. The torque causes thus a change of the beam angle and a movement in the position of the ball. The ball rolls on the beam without slipping under the action of the force of gravity.



Defining the generalized coordinate:

$$q(t) = \begin{bmatrix} p(t) \\ \theta(t) \end{bmatrix}$$

Containing the 2 D.o.F. of the system: the position of the ball and the angle of the beam, respectively.

Considering:

- K_1: Kinetic energy of the beam;
- K_2: Kinetic energy of the ball;
- U: total potential energy of the system;
- K: total kinetic energy of the system;

The Lagrangian of a system is defined as:

$$L = K - U$$

The computation of the **Kinetic Energy** of the two system components:

$$K_1 = \frac{1}{2} J \dot{\theta}^2$$

$$K_2 = \frac{1}{2} J_b {\dot{\theta}_b}^2 + \frac{1}{2} m {v_b}^2$$

Where:

- J: moment of inertia of the beam;
- J_b: moment of inertia of the ball;
- $\cdot\theta_b$: angular speed of the ball;

- ν_b: linear speed of the ball;

Actually both the linear and the angular speed of the ball can be expressed in terms of the generalized coordinates considering r the ball radius:

$$\dot{\theta}_b = \frac{p}{r}$$

$$\nu_b{}^2 = \dot{p}^2 + p^2\dot{\theta}^2$$

Therefore the kinetic energy of the ball in function of the generalized coordinates can be written as:

$$K_2 = \frac{1}{2}\left(\frac{J_b}{r^2} + m\right)\dot{p}^2 + \frac{1}{2}mp^2\dot{\theta}^2$$

The **Potential Energy** of the system:

$$U = mgpsin\theta$$

The **Lagrangian of the system**:

$$L = \frac{1}{2}\left(\frac{J_b}{r^2} + m\right)\dot{p}^2 + \frac{1}{2}(mp^2 + J)\dot{\theta}^2 - mgpsin\theta$$

The two lagrange equations that provide the dynamics of the system will result to be:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{p}}\right) - \frac{\partial L}{\partial p} = 0$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = \tau$$

explaining the terms of the equations:

$$\frac{\partial L}{\partial \dot{p}} = \left(\frac{J_b}{r^2} + m\right)\dot{p}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{p}}\right) = \left(\frac{J_b}{r^2} + m\right)\ddot{p}$$

$$\frac{\partial L}{\partial p} = mp\dot{\theta}^2 - mgsin\theta$$

$$\frac{\partial L}{\partial \dot{\theta}} = (mp^2 + J)\dot{\theta}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) = 2mp\dot{p}\dot{\theta} + (mp^2 + J)\ddot{\theta}$$

$$\frac{\partial L}{\partial \theta} = -mgpcos\theta$$

Therefore the two dynamic equations would result to be:

$$\left(\frac{J_b}{r^2} + m\right)\ddot{p} + mgsin\theta - mp\dot{\theta}^2 = 0$$

$$(mp^2 + J)\ddot{\theta} + 2mp\dot{p}\dot{\theta} + mgpcos\theta = \tau$$

2.2 Non-Linear and Linear State-Variable Representation

The equations of motion we derived for the Ball and Beam system can be written in state-variable representation.

Defining the **state vector**:

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} = \begin{bmatrix} p(t) \\ \dot{p}(t) \\ \theta(t) \\ \dot{\theta}(t) \end{bmatrix}$$

The equations of motion can then be written in terms of the state variables as:

$$\dot{x} = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} x_2 \\ \dfrac{m(x_1 {x_4}^2 - gsinx_3)}{\dfrac{J_b}{r^2} + m} \\ x_4 \\ \dfrac{-2mx_1x_2x_4 - mgx_1cosx_3 + \tau}{(m{x_1}^2 + J)} \end{bmatrix} = f(x, \tau)$$

This state vector is composed of the minimum set of variables required to determine the future response of the system given the input and the current state.
The Jacobian of the right hand side of the above equation, with respect to the state vector x, yields

$$\frac{\partial f}{\partial x}(x,\tau) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \dfrac{mx_4^2}{\frac{J_b}{r^2}+m} & 0 & \dfrac{-mgcosx_3}{\frac{J_b}{r^2}+m} & \dfrac{2mx_1x_4}{\frac{J_b}{r^2}+m} \\ 0 & 0 & 0 & 1 \\ \dfrac{\partial f_4}{\partial x_1} & \dfrac{-2mx_1x_4}{mx_1^2+J} & \dfrac{mgx_1sinx_3}{mx_1^2+J} & \dfrac{-2mx_1x_2}{mx_1^2+J} \end{bmatrix}$$

with:

$$\frac{\partial f_4}{\partial x_1} = \frac{(-2mx_2x_4 - mgcosx_3)(mx_1^2+J) - (-2mx_1x_2x_4 - mgx_1cosx_3 + \tau)}{(mx_1^2+J)^2}$$

The Jacobian with respect to the input yields:

$$\frac{\partial f}{\partial \tau}(x,\tau) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{1}{mx_1^2+J} \end{bmatrix}$$

Now by defining an operating point corresponding to a constant ball position with null velocity. Being null also the corresponding angle and angular velocity, the operating state will be then given by:

$$x_0 = \begin{bmatrix} p_0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

In addition it can also been defined the operating input required to maintain the operating point, above defined, as:

$$u_\tau = mgp_0$$

Then we can evaluate the matrix A,B,C of the state-variable representation at the operating point:

$$A = \frac{\partial f}{\partial x}(x_0,\tau_0) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{-mg}{\frac{J_b}{r^2}+m} & 0 \\ 0 & 0 & 0 & 1 \\ \dfrac{-mg}{mp_0^2+J} & 0 & 0 & 0 \end{bmatrix}$$

$$B = \frac{\partial f}{\partial \tau}(x_0, \tau_0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{mp_0^2 + J} \end{bmatrix}$$

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

Using these matrices, the system can now be written in a state-space representation of the form:

$$\dot{x}(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t)$$

From this state space representation, the system can be analyzed and a controller may be designed.

3 Proposed tasks

3.1 Task-1: Trajectory Exploration

Choose two equilibria and define a step between these two configurations. Compute the optimal transition for the robot to move from one equilibrium to another exploiting the DDP algorithm.

3.2 Task-2: Trajectory Optimization

Define the reference (quasi) trajectory which the ball needs to follow exploiting the DDP algorithm to compute the optimal trajectory.

3.3 Task-3: Trajectory Tracking

Linearizing the system dynamics about the (optimal) trajectory (x, u) computed in Task 2, exploit the LQR algorithm to define the optimal feedback controller to track this reference trajectory.

3.4 Task-4: Physical Implementation / Animation

Presentation of a simple prototype using Raspberry Pi board programmed in python or by making a simple animation of the robot executing Task 3.