

Inertia Wheel Inverted Pendulum

Ashwin Krishna

Harvard University

Email: ashwin_krishna@college.harvard.edu

Nao Ouyang

Harvard University

Email: nouyang@g.harvard.edu

Abstract—We explore the classic nonlinear controls problem, inverting a pendulum, using analyses learned in this class. Specifically, we look at using a flywheel to stabilize the pendulum. We build a physical system from scratch. In simulation, we derive the equations of motion and apply LQR and region of attraction analyses for our system. We additionally perform controllability analysis to the case where the full state may not be measurable, as happened in our physical system. In hardware, we successfully implement downward stabilization and swingup controls using both a PD controller and a bang-bang controller. We discuss the design decisions, design iterations, and present work toward implementing a controller for the inverted state.

I. INTRODUCTION

Pendulum is inherently unstable around upright fixed point.

This demo file is intended to serve as a “starter file” for IEEE conference papers produced under L^AT_EX using IEEE-tran.cls version 1.8b and later. I wish you the best of success.

mds

August 26, 2015

II. LQR ANALYSIS

A. Equations of Motion

To derive the equations of motion (EOM), we use the Lagrangian method. Let L equal to the kinetic energy plus the potential energy of the system.

$$L = KE - PE \quad (1)$$

By Lagrange’s method,

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = \sum_{i=0}^n F_i \quad (2)$$

for $i = 1, 2, 3 \dots n$ forces.

Thus, we need to write out the KE, the PE, the derivative of L with respect to each state q , the derivative of L with respect to the (time) derivative of each state q , and then the time derivative of that last term.

Let us first consider the unaltered case, from the problem set, where here we will derive the equations of motion by hand but otherwise simply explain the derivation in detail. Later, we will consider a system that more closely matches our real-life system. We will not be able to compare the model with reality, since we were unable to implement the full state measurement so LQR cannot apply. Instead, we show another example as applied to a modified system where the reaction wheel pendulum is put on an (unpowered) cart.

1) **Write the KE of the system.** We can decompose this into the translational and rotational components.

First, let us consider (abstractly) the translational KE of a point mass m rotating around the origin on a massless string of length l . θ is defined as angle from the downward vertical point, increasing counterclockwise (diagram not provided). The position of the point mass is $x = l \cos \theta$ and $y = l \sin \theta$. KE is $\frac{1}{2} m \cdot \dot{q}^2$, where q is the position.

$$KE_x = 0.5m \left(l \cdot \frac{d}{dt} \sin \theta \right)^2 = \frac{1}{2} m (l \dot{\theta} \cos \theta)^2 \quad (3)$$

$$KE_y = 0.5m \left(l \cdot \frac{d}{dt} \cos \theta \right)^2 = \frac{1}{2} m (-l \dot{\theta} \sin \theta)^2 \quad (4)$$

$$KE = KE_x + KE_y = \frac{1}{2} m l^2 \dot{\theta}^2 (\cos^2 \theta + \sin^2 \theta) \quad (5)$$

$$= \frac{1}{2} m l^2 \dot{\theta}^2 \quad (6)$$

(7)

where on the last step we used the trig identity $\cos^2 + \sin^2 = 1$.

Now applying this to the stick and flywheel components of our system, we calculate 1) the stick around the origin 2) the flywheel around the origin. Note that the KE of the stick acts at l_1 , the center-of-mass of the stick, not l_2 .

$$KE_{\text{translational}} = \frac{1}{2} m_1 (l_1 \dot{\theta}_1)^2 + \frac{1}{2} m_2 (l_2 \dot{\theta}_1)^2 \quad (8)$$

(9)

Additionally we have the inertial component of KE since we have angular velocities here and our stick has mass and our previous point mass is instead a rotating flywheel. The general formula is $KE = \frac{1}{2} I_2 \dot{\theta}^2$. Noting that angular velocities “add”, and applying this to each component of our system; we calculate 1) inertial KE of the stick 2) inertial KE of the flywheel.

$$KE_{\text{inertial}} = \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 \quad (10)$$

(11)

The total KE of the system is the sum of the above.

2) **Write the PE of the system.** This is more straightforward. Gravitationally speaking, (and with a bit of geometry - note that our θ is defined from vertical and increasing counterclockwise)

$$PE = m_1 g(-l_1 \cos \theta_1) + m_2 g(-l_2 \cos \theta_1) \quad (12)$$

3) Now we have the Lagrangian $L = KE - PE$ and must take the partial of the Lagrangian with respect to each state variable, in our case θ_1 and θ_2 .

Using sympy (note: we left the sympy ordering intact, so the terms are a bit weird), we calculate

$$\frac{\partial L}{\partial q} = \begin{bmatrix} -gl_1 m_1 \sin(\theta_1) - gl_2 m_2 \sin(\theta_1) \\ 0 \end{bmatrix} \quad (13)$$

4) As an intermediate step, we calculate the

$$\frac{\partial L}{\partial \dot{q}} = \begin{bmatrix} I_1 \dot{\theta}_1 + I_2(\dot{\theta}_1 + \dot{\theta}_2) + l_1^2 m_1 \dot{\theta}_1 + l_2^2 m_2 \dot{\theta}_1 \\ I_2(\dot{\theta}_1 + \dot{\theta}_2) \end{bmatrix} \quad (14)$$

5) Finally, we calculate the time derivative of the previous term

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} = \begin{bmatrix} I_2 \ddot{\theta}_2 + \ddot{\theta}_1(I_1 + I_2 + m_1 l_1^2 + m_2 l_2^2) \\ I_2 \ddot{\theta}_1 + I_2 \ddot{\theta}_2 \end{bmatrix} \quad (15)$$

6) We set the equation equal, on the right hand side, to our input torque τ .

We may then directly ask sympy to solve for \ddot{q}

$$\ddot{\theta}_1 = -g \frac{(m_1 l_1 + m_2 l_2) \sin(\theta_1)}{I_1 + m_1 l_1^2 + m_2 l_2^2} \quad (16)$$

$$\ddot{\theta}_2 = g \frac{(m_1 l_1 + m_2 l_2) \sin(\theta_1)}{I_1 + m_1 l_1^2 + m_2 l_2^2} \quad (17)$$

More neatly, we can go directly from Eq. (15) and Eq. (13) to the "manipulator equations" as per the class textbook. Specifically, we put Eq. (15) on the left hand side, factoring out $\ddot{\theta}_1$ and $\ddot{\theta}_2$; then on the right hand side we put Eq. (13), factoring out θ_1 and θ_2 as well as adding in our input torque τ .

That is, we rewrite in form

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = \tau_g(q) + Bu \quad (18)$$

Doing so, we then get as given to us in the homework (yay it matches!)

$$\begin{bmatrix} m_1 l_1^2 + m_2 l_2^2 + I_1 + I_2 & I_2 \\ I_2 & I_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + 0 = \begin{bmatrix} -(m_1 l_1 + m_2 l_2)g \sin \theta_1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tau \quad (19)$$

B. Linearization Around Fixed Point

We can further use sympy to linearize our fixed points.

Focusing on the upright case, we can use the approximation

$$\sin \theta \approx \pi - \theta \text{ for } \theta \approx \pi \quad (20)$$

For the downward case, we can similarly use the approximation

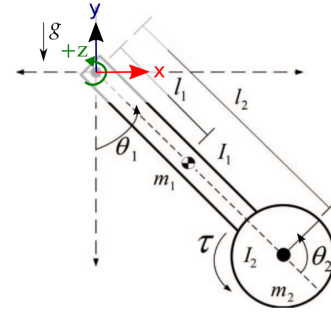


Fig. 1. Free body diagram

$$\sin \theta \approx \theta \text{ for } \theta \approx 0 \quad (21)$$

After plugging in to sympy, we get the same result as in the homework,

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{(m_1 l_1 + m_2 l_2)g}{(m_1 l_1^2 + m_2 l_2^2 + I_1)} & 0 & 0 & 0 \\ -\frac{(m_1 l_1 + m_2 l_2)g}{(m_1 l_1^2 + m_2 l_2^2 + I_1)} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 - 180 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{-1}{(m_1 l_1^2 + m_2 l_2^2 + I_1)} \\ \frac{1}{I_2} + \frac{1}{(m_1 l_1^2 + m_2 l_2^2 + I_1)} \end{bmatrix} [\tau] \quad (22)$$

1) A and B: If we plug in the measurements from our physical system as in Table I, we get the following A and B matrices (rounded).

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 312 & 0 & 0 & 0 \\ -312 & 0 & 0 & 0 \end{bmatrix} \quad (23)$$

$$B = \begin{bmatrix} 0 \\ 0 \\ -1395 \\ 1901 \end{bmatrix} \quad (24)$$

Note: We treat the stick mass as negligible compared the motor, which is modelled as a point mass at distance l_2 ; thus we set l_1 equal to l_2 , and $m_1 = m_{motor}$.

C. Constants

D.

Now we simply supply a cost function. We care a lot about the θ_1 , some about the θ_1 , a bit about $\dot{\theta}_2$, and not at all about θ_2 . We also put a cost on the input using \mathbf{R} (here we closely follow the assignment, since it turns out we will not be able to apply LQR to our physical system).

TABLE I
SYSTEM CONSTANTS

| Property | Measurement |
|-------------------|-------------|
| $m_{stick} = m_1$ | 115 g |
| $m_{flywheel}$ | 546 g |
| m_{motor} | 450 g |
| $l_{stick} = l_2$ | 21 cm |
| $r_{flywheel}$ | 8.5 cm |

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \quad (25)$$

$$R = [0.1] \quad (26)$$

Using the LQR function built into Drake, we get (rounded)

$$K = [-35 \quad 0 \quad -5 \quad -1] \quad (27)$$

$$S = \begin{bmatrix} 12 & 0 & 1 & 0. \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0. & 0. \\ 0. & 0 & 3 & 0. \end{bmatrix} \quad (28)$$

E. Region of Attraction via Lyapunov

F. Energy-based Controller for Swingup

G. Is it Underactuated?

III. LQR FOR "SYSTEM ON WHEELS"

For a detour (in order to demonstrate understanding of the problem set material) we imagine sticking the whole thing on wheels and redo the same analysis, although for sanity we run the calculations through sympy instead of by hand. [1]

A. Controllability

We begin by analyzing the controllability of the system. At an intuitive level, we see that we've added a state (the x position of the cart) without adding any control inputs (we can still only control the flywheel directly). We might expect that we can still stabilize the system, but cannot control the cart position while keeping the pendulum upright (at least with LQR feedback).

To check our intuition, we check the rank of the controllability matrix, defined as

- 1) Equations of Motion:
- 2) Linearization:

IV. CONCLUSION

The conclusion goes here.

ACKNOWLEDGMENT

The authors would like to thank...

REFERENCES

- [1] J. Mellado. (2018) js-aruco: Javascript library for augmented reality applications. [Online]. Available: <https://github.com/jcmellado/js-aruco>

APPENDIX

A. Bill of Materials

A list of the components of the sensor is found in II.

TABLE II
LIST OF COMPONENTS AND APPROXIMATE COSTS

| Part | Details | Cost |
|---------------------------|---|------|
| Camera | Mini Camera module, AmazonSIN: B07CHVYTGD | \$20 |
| LED | Golden DRAGON Plus White, 6000K. 124 lumens | \$2 |
| 4 springs | Assorted small springs set | \$5 |
| 3D printed pieces | PLA filament | \$5 |
| Heat-set Threaded Inserts | Package of 50 from McMaster-Carr (use 2) | \$1 |
| Misc. Bolts | Hex socket head | \$1 |
| Epoxy | 5 minute | \$5 |
| 3.3 V source (Arduino) | Optional | \$15 |