Modelling and Simulation

MPMEE02

Case Study

Modelling of a Yo-Yo System

Rohan Deswal

2020UMP1071

	Index

1 Mathematical Modelling of a Yo-Yo

This document aims to collect and information about and analyse the motion of a yo-yo and hopes to achieve a Simulated model to predict its state at varying inputs

2 Introduction – What is a Yo-Yo?

A yo-yo is a toy consisting of an axle connected to two disks, and a string looped around the axle, similar to a spool. It is played by holding the free end of the string known as the handle (by inserting one finger—usually the middle or ring finger—into a slip knot), allowing gravity (or the force of a throw and gravity) to spin the yo-yo and unwind the string. The player then allows the yo-yo to wind itself back to the player's hand, exploiting its spin (and the associated rotational energy). There are friction pads between the two disks that allow the string to attach or fix itself to the surface of the disk and wind itself back onto the yo-yo.

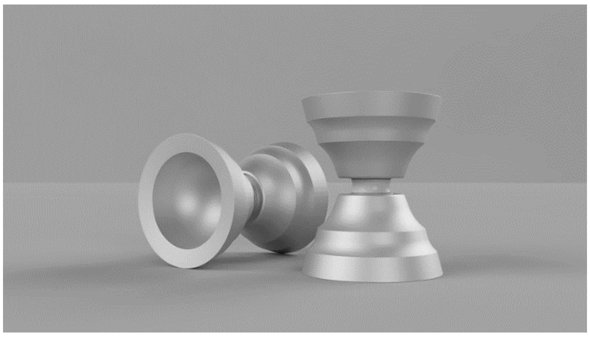


Figure 2 Image of a common Yo-Yo

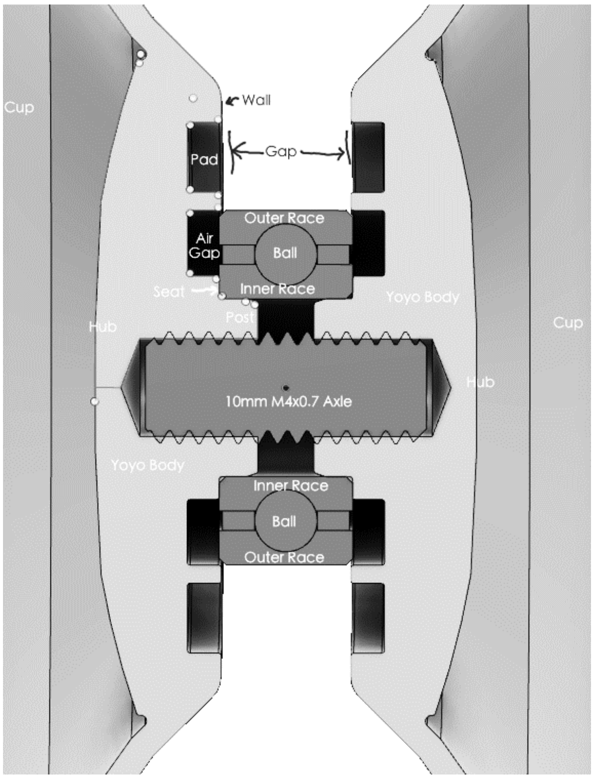


Figure 1 Cross-Section of a Yo-Yo showcasing, disks, axle, bearings and friction pads

3 How to model?

The modelling of the toy is going to be done by taking consideration of the energy balance that exists while playing with it. Follow this, when the toy is going up it tends to convert the kinetic energy into potential energy by winding the string and when it is going down it converts the potential energy to kinetic energy by the action of unwinding the string as the loop is opening. The toy dissipates energy due to the friction between the string and the toy/friction pads. Therefore, the toy will come to a halt if the player stops playing or stops moving the string. Hence, to keep the toy running the player has to supply energy.

Here, the description is of a Discrete Time Model, which describes the top height of a Yo-Yo. First a nominal input motion is decided to simulate the motion of a hand. Next, we compute the toy’s motion by integrating the equations of motion that we will derive. Here, only the recurrence formula for the top of the toy’s height is going to be derived and used since we are going to take that as out control parameter. Also remember that top-height multiplied by the mass of the toy and the gravitational acceleration is the potential energy of the toy therefore the top height equation is also important for the Energy Balancing.

4 Derivation of Equations of Motion of a Yo-Yo

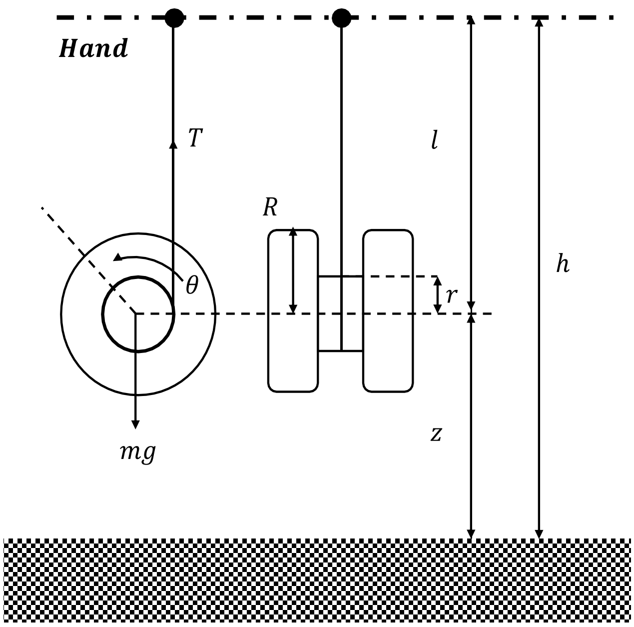


Figure 3 Schematic Diagram of a Yo-Yo

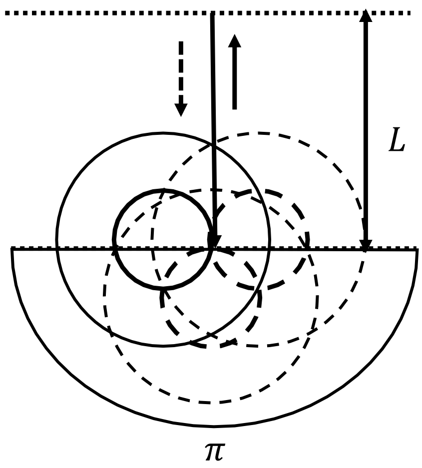


Figure 4 Rotation of π at the bottom

In order the make the analysis much simpler we are making the following assumptions:

Assumption 1: The diameter and mass of the string are neglected. That is it is a mass-less and unidimensional string.

Assumption 2: The string is always perpendicular the axis of the axle which means the centre of mass moves only in vertical direction and we have a constant and fixed rotational axis.

Assumption 3: All frictions are viscous and thus proportional to the rotational velocity of the Yo-Yo.

Assumption 4: The rotational velocity of the Yo-Yo does not change at the bottom. The rotation of π at the bottom is neglected.

Now according, the given schematic in Figure 3., we have the following variables and constants.

1. r, is the radius of the axle
2. θ, is the rotation angle of the Yo-Yo
3. l, is the current or the unwind length of the string
4. z, is the height of the Yo-Yo
5. h, is the height of the hand

Let us start with $l = 0$, when $\theta = 0$, therefore we will have,

$$l = r\theta = h - z$$

Let the friction coefficient be ϵ, the tension of the string be T, the mass and inertia of the Yo-Yo be m and I, respectively. Therefore, when the Yo-Yo goes down, the equations of translation of mass centre and rotation about the mass centre will be given as,

$$m\ddot{z} = -mg + T,\ I\ddot{\theta} = rT - r\epsilon\dot{\theta}$$

On the other hand, after the Yo-Yo turns its direction, we have to following relation,

$$l = L - r\theta$$

Where L is the total length of the string. According to assumption 4, the energy is not dissipated at the bottom. Thus, equation of motion when the Yo-Yo goes up becomes,

$$I\ddot{\theta} = -rT - r\epsilon\dot{\theta}$$

From the above equations we get the equation of motion as,

$$(I + mr^2)\ddot{\theta} = \pm mr(\ddot{h} + g) - r\epsilon\dot{\theta},$$

Where + denotes downward motion and – denoted upward motion.

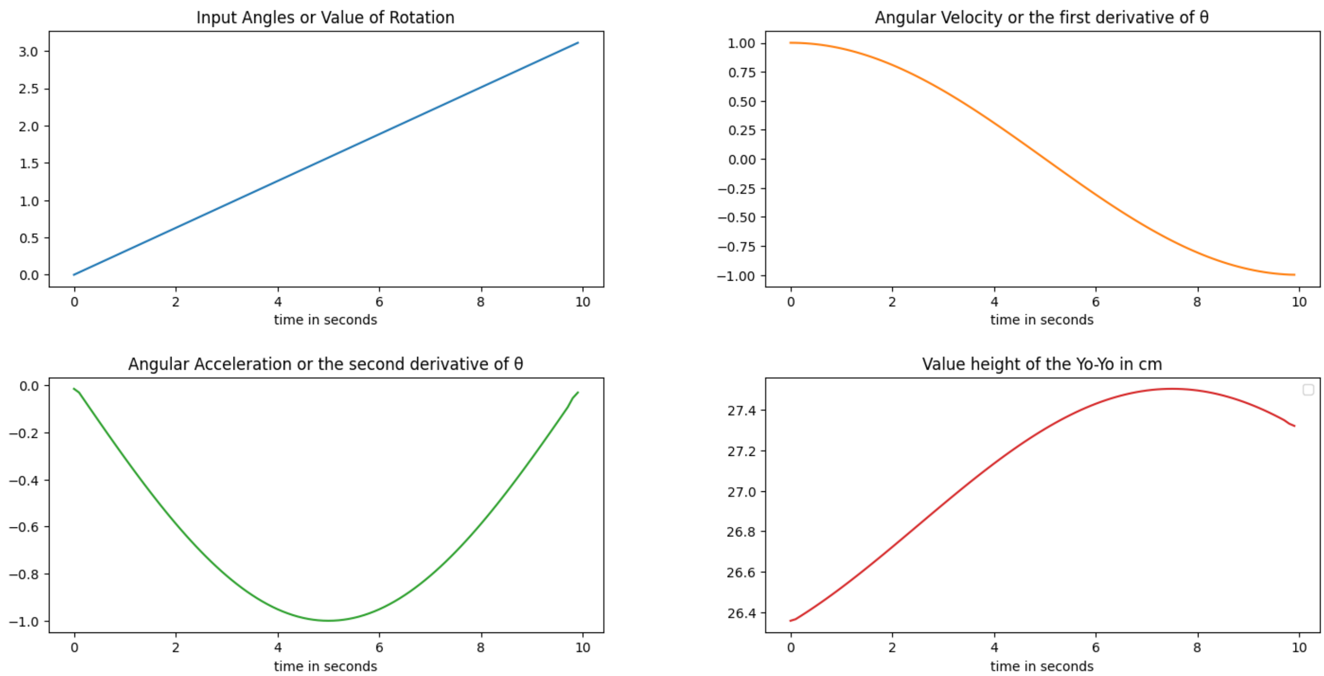
5 Simulation of the Generated Model

Now let us simulate the trajectories of θ and h in python and plot them using matplotlib.

First, we generate a linear-space of possible values of theta for $\theta \in [0, \pi)$, then we calculate a list of first derivatives of θ, then second derivatives of θ that is $\dot{\theta}$ and $\ddot{\theta}$ respectively to represent corresponding angular velocity and angular acceleration for the given values of rotations.

We are going to calculate the derivatives using the NumPy. Gradient () function for a sinusoidal input of rotations.

We are going to use these values as inputs for the equation of motion and get the corresponding values of $\ddot{h}$, then integrate twice to get the values of first linear velocity of motion and then the heights at various rotations.



Here are the output trajectories and their plots generated by the code,

6 Conclusion

We have used the analytically derived model and generated a hypothetical trajectory and numerically calculated its trajectory velocity, acceleration and hence the resultant height of the Yo-Yo based on our model.

7 Future Work

The predicted trajectory needs to be verified and compared against an actual Yo-Yo attached to a robotic arm or an actuated string.

8 References

Figure 1: forums.yoyoexpert/u/RyoCanCan

Figure 2: forums.yoyoexpert/u/MarkDferences

K. Hashimoto and T. Noritsugu, "Modeling and control of robotic yo-yo with visual feedback," Proceedings of IEEE International Conference on Robotics and Automation, 1996, pp. 2650-2655 vol.3, doi: 10.1109/ROBOT.1996.506562.

H. Kim, Y. Yamakawa, T. Senoo and M. Ishikawa, "Robotic manipulation of rotating object via twisted thread using high-speed visual sensing and feedback," 2015 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), 2015, pp. 265-270, doi: 10.1109/MFI.2015.7295819.

9 Here is the code used for the above simulation.

```python
'''
This is the code for the Yo-Yo
Simulated Model written by
Rohan Deswal, 2020UMP1071, MPAE, NSUT Delhi
based on the work of Koichi Hashimoto and Toshiro Noritsugu
of Okayama University.
'''

import numpy as np
import scipy as sp
import matplotlib.pyplot as plt

m = 1 # mass of the Yo-Yo
r = 0.1 # radius of the Yo-Yo
l = 1 # lenght of the complete string
e_friction = 0.15 # approx coefficent of friction between common plastics and cotton strings
I = 0.5*m*(r**2) # inertia fo the Yo-Yo
g = 9.801 # gravitational acceleration

dt = np.pi/100 # Time Interval
time_axis = np.arange(0,10,0.1)
theta = np.arange(0,np.pi,dt) # Input of Various Angular Positions
theta_d1 = np.gradient(np.sin(theta),dt) # Velocity profile for a sinusoidal input rotation
theta_d2 = np.gradient(theta_d1,dt) # corresponding angular acceleration

def linear_acceleration(theta_d1_val, theta_d2_val):
    '''
    This function is used to calculate the instantaneous
    linear acceleration of the Yo-Yo given the angular
    acceleration and angular velocity of the Yo-Yo based on
    the analysis given in the document.
    '''
    return (((I+m*(r**2))*theta_d2_val + r*e_friction*theta_d1_val)/(m*r)) - g

h_d2 = [-linear_acceleration(theta_d1[i],theta_d2[i]) for i in range(len(theta_d1))]
h_d1 = [0]
h = [0]

# Calculation of Integral to find Height of the Yo-Yo
for a in h_d2:
    h_d1.append(h_d2[-1] + a*dt)
    h.append(100*(h_d1[-1] + 0.5*a*dt**2) % 1000)

fig, axs = plt.subplots(2, 2)
axs[0, 0].plot(time_axis, theta)
axs[0, 0].set_title('Input Angles or Value of Rotation')
axs[0, 1].plot(time_axis, theta_d1, 'tab:orange')
axs[0, 1].set_title('Angular Velocity or the first derivative of Î¸')
axs[1, 0].plot(time_axis, theta_d2, 'tab:green')
axs[1, 0].set_title('Angular Acceleration or the second derivative of Î¸')
axs[1, 1].plot(time_axis, h[1:], 'tab:red')
axs[1, 1].set_title('Value height of the Yo-Yo in cm')

fig.tight_layout()

for ax in axs.flat:
    ax.set(xlabel='time in seconds')

plt.legend(loc="best")
plt.show()
```