# Lab 9: Ballistic Pendulum

Jonah Spector, Alec Gray, Sarah McMahon

24 October 2024

## 1 Introduction

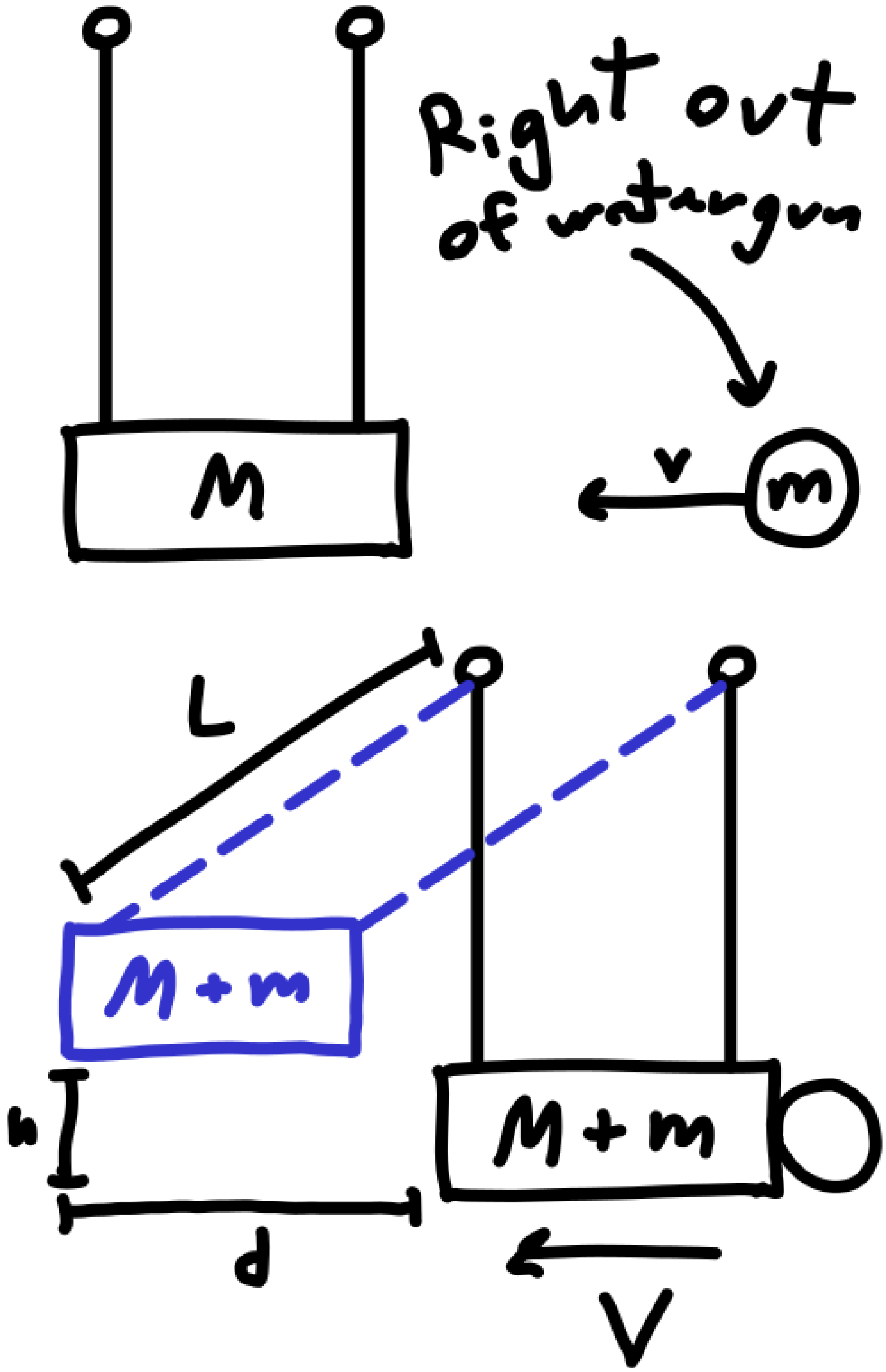
The law of conservation of energy states that energy cannot be created or destroyed, only transformed or transferred. The application of this law is the primary focus of today's lab, particularly with a focus on the conservation of momentum. Using the equations derived before the lab, we plan to predict the muzzle velocity of a water gun, which can then be used to predict the ballistic trajectory that a shot will follow. We intend to create a highly accurate prediction through the use of repeated measurements, which we will calculate the error bounds for later.

## 2 Derivations

Before moving on to our actual experiment in conservation of momentum and energy, the equations for it must be found. The experiment involves a watergun firing a waterball into a bottle, measuring the muzzle velocity thereof, and using that to calculate an expected firing distance.

| Variable | Description |
|---|---|
| $m$ | Mass of waterball |
| $M$ | Mass of waterbottle |
| $d$ | Distance waterbottle swings back |
| $L$ | Length of waterbottle pendulum |
| $h$ | Height waterbottle swings to |
| $v$ | Velocity of waterball |
| $V$ | Velocity of waterball/waterbottle |
| $D$ | Horizontal distance (test shot) |
| $H$ | Vertical distance (test shot) |

**Equation One**




First, to solve for $v$ in terms of $V$, we use conservation of momentum. At the start, the momentum of the system is:

$$p = mv$$

After the collision, the momentum is:

$$p = (m + M)V$$

Solving the for $v$ in terms of $V$:

$$mv = (m + M)V$$

$$v = V\left(\frac{m + M}{m}\right)$$

$$\boxed{v = V\left(1 + \frac{M}{m}\right)}$$

**Equation Two**

The equation for the initial kinetic energy of the waterball and waterbottle system, right after the collision, is:

$$K = \frac{1}{2}(m + M)V^2$$

The potential energy of the combined system at the maximum of its swing backwards, where the velocity is 0, is:

$$U_g = (m + M)gh$$

Solving for $V$ using conservation of energy:

$$\frac{1}{2}(m + M)V^2 = (m + M)gh$$

$$V^2 = 2gh$$

$$\boxed{V = \sqrt{2gh}}$$

**Equation Three** is given as:

$$\boxed{h \approx \frac{d^2}{2L}}$$

where the equation uses a small angle approximation (meaning a bigger value for $L$ is more accurate).

**Equation for Muzzle Velcoity**

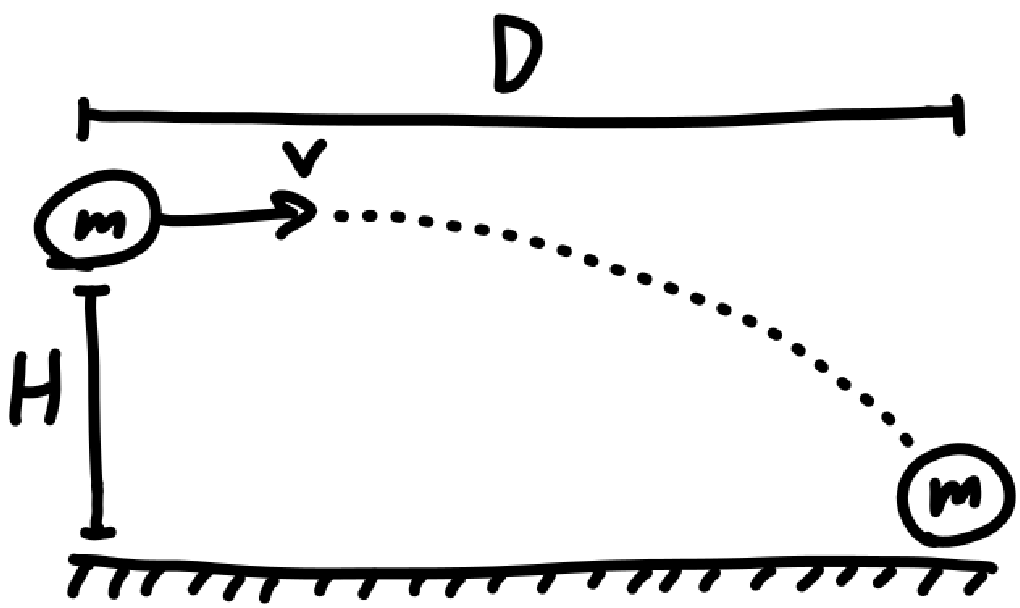
Finally, to find the muzzle velocity, simply plug in the previous equations:

$$v = \sqrt{2gh}\left(1 + \frac{M}{m}\right)$$

$$v = \sqrt{\frac{2gd^2}{2L}}\left(1 + \frac{M}{m}\right)$$

$$\boxed{v = d\sqrt{\frac{g}{L}}\left(1 + \frac{M}{m}\right)}$$

**Equation Four (For Test)**




This is a simple enough kinematics problem, since $v$ only has a horizontal component ($v_y = 0$):

$$H = \frac{1}{2}gt^2$$

$$\sqrt{\frac{2H}{g}} = t$$

$$D = vt$$

$$\boxed{D = v\sqrt{\frac{2H}{g}}}$$

**Equation Five**

Putting everything together:

$$D = d\sqrt{\frac{g}{L}}\left(1 + \frac{M}{m}\right)\sqrt{\frac{2H}{g}}$$

$$\boxed{D = d\sqrt{\frac{2H}{L}}\left(1 + \frac{M}{m}\right)}$$

# 3 Methods

Over the course of twelve trials, we would record the displacement of the water backwards, as well as the mass of the bottle before and afterwards.

1. Measure the initial mass of the bottle ($M$)
2. Stabilize the bottle of the sling
3. Shoot the bottle and record a video with a ruler in the background
4. Measure the distance from the video ($d$)
5. Measure the final mass of the bottle ($M_f$)
6. Dispose of water and repeat

To calculate the waterball mass, we weighed the waterbottle before ($M$) and after ($M_f$) it received the waterball. Thus, the waterball mass is $M_f - M$.

# 4 Error Propagation

It is assumed that all instruments are 100% accurate *but not precise*, since there is no way to know the actual, true value of the measurement in this lab. Regardless, we first found the instrument limits of error for things measured once by hand (the height $H$ for the test shot and length $L$ for the pendulum). For the swing-back distance $d$ and mass measurements $M/M_f/m$, the standard error of the dataset is used. It makes more since to use standard error over the minimum and maximum of the dataset due to the prospect of outliers and it makes more sense than the standard deviation since there *is* some value that the measurement is approximating. Standard error is calculated from the standard deviation like so:

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$$

$$\sigma = \sqrt{\frac{\sum(x-\bar{x})^2}{n-1}}$$

Error needs to be propagated from the collected data in our final equation. Intermediate errors, like the ones in the calculated muzzle velocities, are not computed separately for simplicity, since only the error of the final test shot horizontal distance is necessary for the experiment.

$$D = d\sqrt{\frac{2H}{L}}\left(1+\frac{M}{m}\right)$$

The values $d$, $L$, $H$, $M$, and $m$ have errors, $\sigma_d$, $\sigma_L$, $\sigma_H$, $\sigma_M$, $\sigma_m$, which need to be propogated.

We can do this using the following multivariable variance formula:

$$\sigma_f = \sqrt{\left(\frac{\partial f}{\partial x_1}\sigma_{x_1}\right)^2 + \left(\frac{\partial f}{\partial x_2}\sigma_{x_2}\right)^2 + ... + \left(\frac{\partial f}{\partial x_n}\sigma_{x_n}\right)^2}$$

Plugging the values we have in question in:

$$\sigma_D = \sqrt{\left(\frac{\partial D}{\partial d}\sigma_d\right)^2 + \left(\frac{\partial D}{\partial L}\sigma_L\right)^2 + \left(\frac{\partial D}{\partial H}\sigma_H\right)^2 + \left(\frac{\partial D}{\partial M}\sigma_M\right)^2 + \left(\frac{\partial D}{\partial m}\sigma_m\right)^2}$$

The partial derivatives are:

$$\frac{\partial D}{\partial d} = \sqrt{\frac{2H}{L}}\left(1+\frac{M}{m}\right) \qquad \frac{\partial D}{\partial L} = -\frac{d}{2}\sqrt{\frac{2H}{L^3}}\left(1+\frac{M}{m}\right)$$

$$\frac{\partial D}{\partial H} = \frac{d}{2}\sqrt{\frac{2}{LH}}\left(1+\frac{M}{m}\right) \qquad \frac{\partial D}{\partial M} = \frac{d}{m}\sqrt{\frac{2H}{L}} \qquad \frac{\partial D}{\partial m} = -\frac{dM}{m^2}\sqrt{\frac{2H}{L}}$$

It must also be noted that, since $m = M_f - M$ (a product of the method of experimentation), error for the water slug can be calculated as

$$\sigma_m = \sqrt{\sigma_{M_f}^2 + \sigma_M^2}$$

Sources:


- https://www.epfl.ch/labs/lben/wp-content/uploads/2018/07/Error-Propagation_2013.pdf
- https://en.wikipedia.org/wiki/Propagation_of_uncertainty#Simplification

## 5 Data

[21]:

```python
from math import *

import numpy as np

# Gravity
g = 9.81  # m/s

# e_ denotes standard error, d_ denotes derivatives


## Functions and error propagator ##
def muzzle_velocity(d, L, M, m):
    return d * sqrt(g / L) * (1 + (M / m))


def shot_distance(v, H):
    return v * sqrt(2 * H / g)


def error_propagator(d, L, H, M, m, e_d, e_L, e_H, e_M, e_m):

    # Calculate the partial derivatives
    dD_dd = sqrt(2 * H / L) * (1 + (M / m))
    dD_dL = (-d / 2) * sqrt(2 * H / (L**3)) * (1 + (M / m))
    dD_dH = (d / 2) * sqrt(2 / (L * H)) * (1 + (M / m))
    dD_dM = (d / m) * sqrt(2 * H / L)
    dD_dm = (-d * M / (m**2)) * sqrt(2 * H / L)

    # Make all the elements in the variance formula
    partials = [dD_dd * e_d, dD_dL * e_L, dD_dH * e_H, dD_dM * e_M, dD_dm * e_m]

    # Put the elements through the variance formula
    return sqrt(sum([part**2 for part in partials]))


## Measurements ##

# Masses added to the waterbottle (unused since M_f includes these)
# added_mass     = 2*0.5 # kg, the two added masses
# dry_towel_mass = _     # kg, to reduce splashback

# Pendulum length measured by hand
L = 1 + (0.73 - 0.49)  # m
e_L = 0.005  # m (ILE)
```

```python
# Test shot height measured by hand
H = 1  # m
e_H = 0.005  # m (ILE)

# Tabular measurements (12 trials)
M = np.array(
    [
        1.17242,  # Towel 1
        1.18605,
        1.19035,
        1.18793,
        1.19312,
        1.19201,
        1.15789,  # Towel 2
        1.18140,
        1.19352,
        1.19623,
        1.19903,
        1.19617,
    ]
)  # kg
M_f = np.array(
    [
        1.1952,  # Towel 1
        1.21104,
        1.20938,
        1.20638,
        1.21281,
        1.21535,
        1.18203,  # Towel 2
        1.20604,
        1.21686,
        1.22097,
        1.22092,
        1.22080,
    ]
)  # kg
d = np.array(
    [
        0.560 - 0.38,  # Towel 1
        0.55 - 0.39,
        0.64 - 0.48,
        0.62 - 0.46,
        0.56 - 0.41,
        0.53 - 0.38,
        0.57 - 0.42,  # Towel 2
        0.56 - 0.41,
```

```
        0.57 - 0.42,
        0.53 - 0.39,
        0.54 - 0.39,
        0.53 - 0.38,
    ]
)  # m

## Calculations ##

av_M = np.mean(M)
av_M_f = np.mean(M_f)
av_d = np.mean(d)

# Standard error used
e_M = np.std(M) / sqrt(len(M))  # kg
e_M_f = np.std(M_f) / sqrt(len(M_f))  # kg
e_d = np.std(d) / sqrt(len(d))  # m

# Calculating the water slugs' masses
m = M_f - M  # kg
av_m = np.mean(m)
e_m = sqrt((e_M**2) + (e_M_f**2))

# Final muzzle velocity calculations
v = np.array([muzzle_velocity(dd, L, MM, mm) for dd, MM, mm in zip(d, M, m)])
av_v = muzzle_velocity(av_d, L, av_M, av_m)

# Find test shot estimation distances
shot_d = np.array([shot_distance(vv, H) for vv in v])
av_shot_d = shot_distance(av_v, H)

# Find test shot standard errors
e_shot_d = error_propagator(av_d, L, H, av_M, av_m, e_d, e_L, e_H, e_M, e_m)

# Rounding is arbitrary - there is no way to get sub-centimeter precision anyway
print(f"Shot velocity: {round(av_v, 2)} m/s")
print(f"Shot distance: {round(av_shot_d, 2)} \u00B1 {round(e_shot_d, 2)} m (H = 
 ↪{H} m)")
```

```
Shot velocity: 23.17 m/s
Shot distance: 10.46 ± 2.08 m (H = 1 m)
```

## 6 Analysis

It is clear in the shot spread that our prediction was incredibly accurate. The middle of spread was 7 meters from the start of the shot, but the wind (which was very strong) accounts this discrepancy from our prediction of 10.46 meters. The error bounds on our spread from the center are almost exactly two meters, just like our prediction of ± 2.08 meters.

## 7 Conclusion

Our experiment on the ballistic pendulum successfully applied the law of conservation of energy and momentum. We were able to predict the muzzle velocity of the water gun using the displacement values (all around 15 cm). From this we calculated a shot distance of around 10 m. Despite some discrepancies due to external factors such as wind, our results show an incredible agreement with the predicted values, specifically the error bounds (± 2.08 m) calculated with error propagation.