Mini-Project 3 Day 2: Including the sling launch in your simulation

Now that you have a working 2D simulation of the ball trajectory, it is now time to simulate the effect of the sling (the bungee strap) when launching the ball. This will involve two types of modifications to your existing code. First you will need to set new initial conditions for x and y, and second you will have to include the force of the sling in your equations of motion.

On the day of the contest, you will need to specify the launch angle, θ, and the initial sling deflection, d, you want for your shot. The modifications you make to your code today will make your code use these initial parameters.

We will use the notation shown in Figure 1 below to describe the geometry of our launcher.

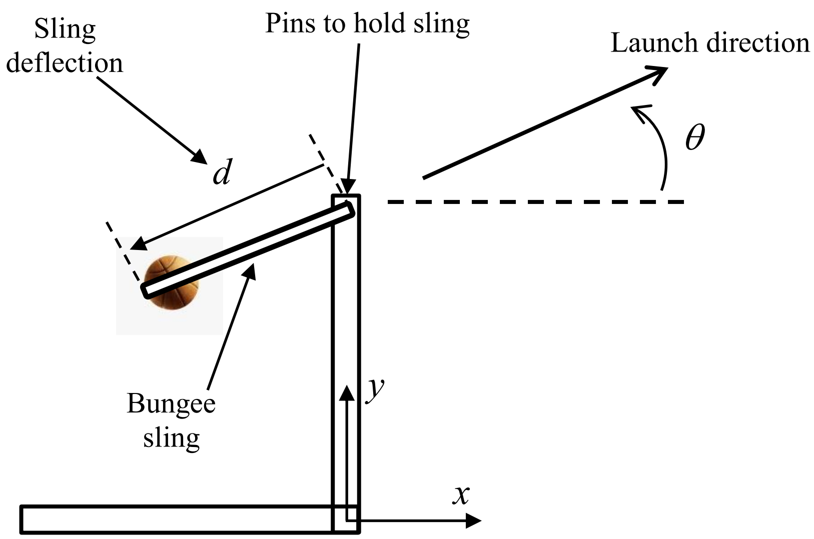

Figure 1 Schematic of basketball sling shot launcher. The basketball is shown in the sling stretched back a distance d from the pins.

The force of the sling, F_{sling}, is somewhat like our bungee cord project. It is dependent on the amount of stretch, δ, you give to the sling as shown in Figure 2.

The stretch in the sling can be found by subtracting the zero-force deflection of the sling from the deflection d as shown in equation (1) below.

The data in Figure 2 can be found on the course web site in the file named “slingforce.xls.”

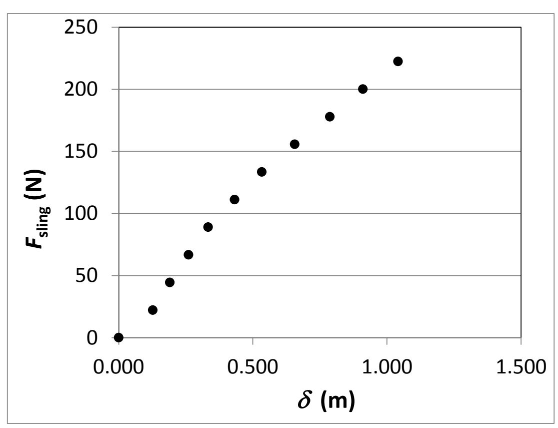

Figure 2 Amount of stretch, δ, in the sling as a function of sling force.

The equations of motion with air drag and with sling force are:

$$\delta = d - d_0 \tag{1}$$

$$\frac{dV_x}{dt} = -kC_d V V_x + \frac{1}{m} F_{sling} \cos\theta \tag{2}$$

$$\frac{dV_y}{dt} = -g - kC_d V V_y + \frac{1}{m} F_{sling} \sin\theta \tag{3}$$

$$\frac{dx}{dt} = V_x \tag{4}$$

$$\frac{dy}{dt} = V_y \tag{5}$$

where

$$k = \frac{\rho \pi D^2}{8m} \tag{6}$$

$$V = \sqrt{V_x^2 + V_y^2} \tag{7}$$

The initial conditions for this problem now account for the fact that we stretch the sling back from the position of the pins on the launch structure. From this position the ball starts with a velocity of zero.

$$x(0) = x_{pin} - d\cos\theta \tag{8}$$

$$y(0) = y_{pin} - d\sin\theta \tag{9}$$

$$V_x(0) = 0 \tag{10}$$

$$V_y(0) = 0 \tag{11}$$

In these expressions,

x	=	x position of the projectile (m)
y	=	y position of the projectile (m)
x_{pin}	=	x position of the pin that holds the sling (m) = 0 m for our apparatus.
y_{pin}	=	y position of the pin that holds the sling (m) = 1.75 m for our apparatus.
δ	=	Stretch in the sling (m)

d_0	=	Zero-force sling deflection (m). Our sling has a value of 0.406m.
d	=	Sling deflection (m)
V_x	=	Horizontal velocity of projectile (m/s)
V_y	=	Vertical velocity of projectile (m/s)
θ	=	Angle of shot (degrees or radians)
C_d	=	Drag coefficient
g	=	Acceleration due to gravity (9.81 m/s^2)

Today's Task

Your task this second day is to update your trajectory.m function to use the sling force. Use the same small time step as last time, $\Delta t = 0.001$ sec. To help everyone's initial attempts look similar, let us use $\theta = 45^o$, x_{max} = 30m, and an initial sling deflection, *d*, of 114cm (45 in.). Like last time, use a ball diameter of 0.228m and a ball mass of 0.45kg. Plot the trajectory of the projectile—use `plot(x,y)` —to help evaluate whether you answers look reasonable.

Here are some words of wisdom:

- Make sure you do not use an `xlsread` command anywhere that it will be run repeatedly. Remember that command is very slow. You should read in the Excel data *once* and then pass it to your functions as a variable.
- At any given instant in time after the ball is released, you can compute the sling deflection, *d*, with the equation below. This way if *d* is positive and greater than d_0, you know that the sling is giving a positive force on the ball. If *d* is less than d_0 or negative, you know the sling force is zero.

$$d = \frac{y_{\text{pin}} - y}{\sin\theta}$$

- One more issue with computing sling force: You should consider the current time as well. If the current time is greater than the time to launch the ball, then your F_{sling} should be zero—even if you later in the simulation you compute a *d* greater than 0.406m!
- Consider making the sling force calculation part of a function. That will help keep your trajectory.m code less cluttered.