

Mini-Project 3: Shooting a Basketball**Problem Description:**

We will be launching basketballs with a large sling shot. Your task will be to write a MATLAB program and a Working Model simulation that will predict how far back the sling shot must be pulled and what angle the shot must take to make a basket.

On the day of our contest, we will go to the SRC. You will be given the following parameters: ball mass, ball diameter and distance to basket. You will need to find a sling deflection value and a shooting angle.

One of the main issues with a problem like this is that you cannot “solve for” the answers directly. Rather you will write a program that will run your simulation many, many times with lots of different sling deflections and shooting angles. The program will have to evaluate the results and tell you which values of sling deflection and shooting angle would be the best to use.

Suggested Schedule:

Day 1 (Tuesday, Feb. 7):

Create a function that will compute your 2D trajectory including the effect of drag. This function will use only an initial angle and initial velocity. Create a main program that will test this function.

Day 2 (Thursday, Feb. 9):

Modify your function to account for the force of the sling (the bungee cord). Instead of an initial velocity, you will use an initial sling deflection.

Day 3 (Friday, Feb. 10):

Modify your main program to enable searching through many values of launch angle and bungee deflection to find combinations that hit the target.

Day 4 (Monday, Feb. 13):

Simulate the ball launching with Working Model to check your MATLAB results.

Day 5 (Tuesday, Feb. 14):

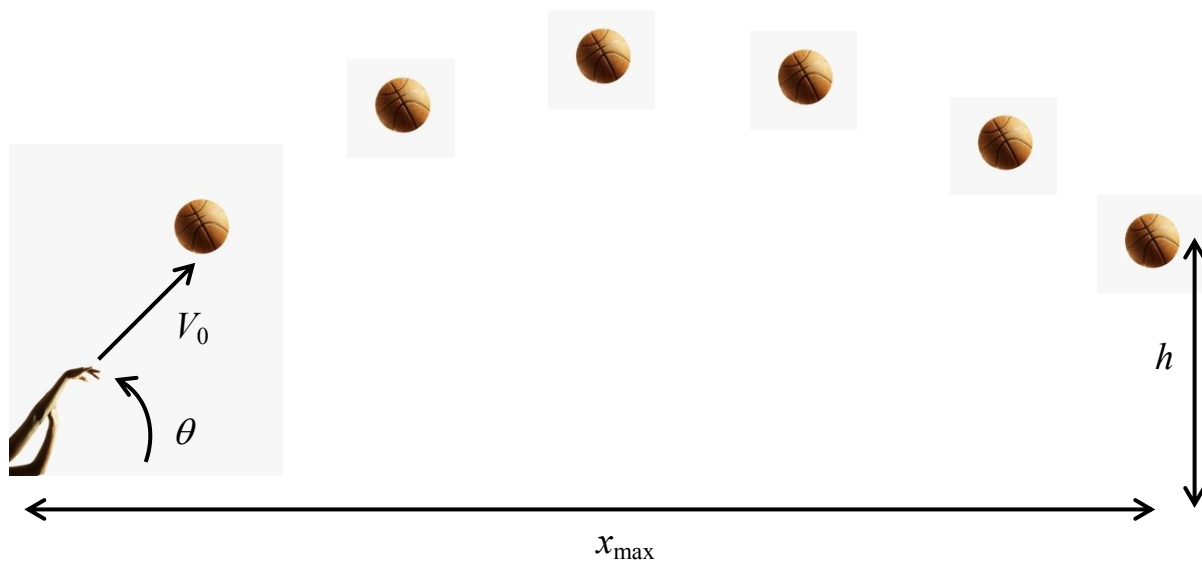
In class work time

Day 6 (Wednesday, Feb. 15):

Contest day at SRC

Mini-Project 3 Day 1: Computing 2D trajectories with drag

We've learned a lot about using Euler's method to model simple equations. Now let us model the 2D flight of a ball after it is released. In class today we will work on the equations of flight with air drag. You will write a function to compute the trajectory and a main program to call that function. Remember, we want the trajectory computations in a function because we need to compute trajectories for many different guesses of initial velocity and angle.



The equations of motion with air drag are:

$$\frac{dV_x}{dt} = -kC_dVV_x \quad (1)$$

$$\frac{dV_y}{dt} = -g - kC_dVV_y \quad (2)$$

$$\frac{dx}{dt} = V_x \quad (3)$$

$$\frac{dy}{dt} = V_y \quad (4)$$

where

$$k = \frac{\rho\pi D^2}{8m} \quad (5)$$

$$V = \sqrt{V_x^2 + V_y^2} \quad (6)$$

The initial conditions to use for this problem are:

$$x(0) = 0 \quad (7)$$

$$y(0) = 0 \quad (8)$$

$$V_x(0) = V_0 \cos \theta \quad (9)$$

$$V_y(0) = V_0 \sin \theta \quad (10)$$

In these expressions,

- x = x position of the projectile (m)
- y = y position of the projectile (m)
- V_0 = Magnitude of velocity of ball as it leaves the shooter's hands (m/s)
- 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/s²)

Your task this first day is to use Euler's method to solve these equations. For today, use a small time step, say $\Delta t = 0.001$ s. To help everyone's plot look similar, let us use $\theta = 50^\circ$, $x_{max} = 30$ m, and $V_0 = 20$ m/s for these plots. Use a ball diameter of 0.228 m and a ball mass of 0.45 kg. Plot the trajectory of the projectile—use `plot(x,y)`—to help evaluate whether your answers look reasonable.

Here are some words of wisdom:

- Put clear variables, close all at the start of the calling program.
- Use a function to do the trajectory calculation. Within the function, use *one* “while” loop for all four differential equations (equations (1) to (4) above). Keep looping until the projectile reaches x_{max} or until the ball hits the ground.
- Let the inputs to the function be the things that might change. Do not put any constants in the function that aren't declared as data near the beginning of your function, for example g . If you do not do this now, we assure you that you will regret it for at least the next two weeks as you try to debug your code.
- *Don't forget to use good commenting!*
- Here is a suggestion for the start of your function:

```
function [x,y,t]=trajectory(angle,V0,D,m,drag,dt,xmax)
%
% Compute the trajectory of the projectile fired from the cannon.
% This version does not include the effects of air drag.
%
% Inputs:
%   angle = angle of cannon (degrees)
%   V0    = velocity of projectile when it leaves the cannon (m/s)
%   m     = mass of ball (kg)
%   D     = diameter of ball (m)
%   dt    = time step size (seconds)
%   xmax  = distance to the target
%   drag  = matrix of drag data - 1st column is Re, 2nd column is Cd
%
% Outputs: x = array of x coordinates of the projectile
%          y = array of y coordinates of the projectile
%          t = array of times to go with x and y
%
```