4-24-2019

Projectile Trajectory Modeling from Motion Imaging

Mark William Richter

Projectile Trajectory Modeling from Motion Imaging

by

Mark William Richter

An Honors Capstone

submitted in partial fulfillment of the requirements

for the Honors Certificate

to

The Honors College

of

The University of Alabama in Huntsville

April 24th, 2019

Honors Capstone Director: Dr. Matthew Turner

The Systems Management and Production Center

____
Student Date

____
Director Date

____
Department Chair Date

____
Honors College Dean Date

HONORS COLLEGE
THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

Honors College
Frank Franz Hall
+1 (256) 824-6450 (voice)
+1 (256) 824-7339 (fax)
honors@uah.edu

Honors Thesis Copyright Permission

This form must be signed by the student and submitted as a bound part of the thesis.

In presenting this thesis in partial fulfillment of the requirements for Honors Diploma or Certificate from The University of Alabama in Huntsville, I agree that the Library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by my advisor or, in his/her absence, by the Chair of the Department, Director of the Program, or the Dean of the Honors College. It is also understood that due recognition shall be given to me and to The University of Alabama in Huntsville in any scholarly use which may be made of any material in this thesis.

____

Student Name (printed)

____

Student Signature

____

Date

In the development of my capstone, I aimed to assist my senior design group to improve K through 12 STEM education. As a team we worked to design and create a Ramp Up PBLE (Project Based Learning Experience) kit to be used in classrooms by teachers and students to aid with STEM topics. The primary focus of the Ramp Up PBLE kit designed and created by our team is the introduction of projectile motion. To achieve this, a kit was designed for three grade levels in which students work to build a catapult. The catapults are used to introduce appropriate STEM topics for the intended grade level. The catapults allow for students to engage hands on in STEM and gain a better understanding of projectile motion first hand.

My addition to the group's project aims to focus on the kits intended for middle and high school level students to promote STEM education through an interactive program. Working in parallel with the Ramp Up PBLE kit, the program demonstrates to students how the STEM topics learned both in the classroom and through the developed kits can be applied. By using rapid image capturing to observe the launch of a projectile from the catapult built by the student, my program walks the user through the steps of determining the launch angle and velocity. The program then uses this information to create a model of the projectile's flight considering both the forces of gravity and drag. While projectile motion has many more factors and variables which influence the projectiles path, the model is simplified to a two-dimensional model only considering the previously mentioned forces. This simplification allows students to obtain a better understanding of the process in which the model is created by allowing the user to apply what they know to determine the initial values of the projectile's flight.

The aim of this program is to use the skills students are currently learning in the classroom to collect information which the program then uses to create a model of the projectile's flight. Using high speed image capturing, users observe the launch of the projectile of their choice from the kit provided to them. From the skills they have developed in school, they are then able to determine the initial speed and angle of the projectile. This information, along with the chosen projectile type, is then placed in the program to create a model of the flight. Students can then use the program to change the different inputs and gain an understanding of how different variables impact the flight of the projectile. The program acts as a tool to aid students in the understanding of the engineering process.

The focus of this project is centered around education at a K through 12 level. The program works to both create a model of the flight of a chosen projectile and educate the user on the fundamentals of projectile motion. Working within the bounds of the Ramp Up PBLE kit developed, this program allows students using the kits to test multiple variables afforded to them in the building of the catapult. In classrooms, students use the program as a tool to analyze and understand the STEM concepts highlighted within the kits.

To ensure the program met educational standards, an objective of plus or minus thirty centimeters in the accuracy of the model was set in the creation of this program. While the model is a simplified two-dimensional model, it was important to achieve this accuracy to further demonstrate STEM topics and add to the hands-on learning experience of the Ramp Up PBLE kit. With the in-classroom application of the program, this goal was believed to be very achievable.

With the inclusion of the force of drag, the projectile motion equation cannot be solved analytically, and thus a numerical solution must be obtained. This is achieved by using the Runge-Kutta method to create an iterative solution to the position and velocity of the projectile. At each time step, the acceleration in both the X and Y directions, as determined by the forces of gravity and drag, are calculated using the velocity from the previous step. This is calculated using equations 1 and 2. The value of velocity in the X and Y directions is then updated from the acceleration value, using equations 3 and 4, and the position is updated using the updated values of velocity, equations 6 and 7. This iterative loop runs from a time of zero through one and a half of the theoretical time of flight with a time step of 0.0001 seconds. However, only steps in which the Y position of the projectile is greater than zero are considered.

$$a_x = -k * V * V_x \tag{1}$$

$$a_y = g - (k * V * V_y) \tag{2}$$

$$V_x = V_x + (a_x * \Delta t) \tag{3}$$

$$V_y = V_y + \left(a_y * \Delta t \right) \tag{4}$$

$$V = \sqrt{{V_x}^2 + {V_y}^2} \tag{5}$$

$$x = x + (V_x * \Delta t) \tag{6}$$

$$y = y + (V_y * \Delta t) \tag{7}$$

To simplify the use of the drag force within the program, a constant, k, was used to describe the air density, drag coefficient, and cross-sectional area of each given projectile thus making the force of drag a function of only velocity. This constant was found experimentally through a similar Runge-Kutta method described above. Each projectile option was dropped from a set height and the time in which it took the projectile to fall to the ground was observed and recorded. Using this value along with the mass of the selected projectile, a k value was found.

$$k = \frac{1}{2} (\rho * C_d * A_c) \tag{8}$$

This program has been used in classrooms along side the Ramp Up PBLE kit. The program works well with students and the ability to use what they know to model what they see engages students working with the kits. Problems encountered through the use of the program in classrooms were fixed and using suggestions from students the program was further developed to become more user friendly. While used in classrooms, an accuracy of plus or minus ten centimeters was observed.

This user interface of the program is as follows.

The user observes the launch of an assembled catapult in front of a standardized grid using rapid image capturing.

After using rapid image capturing to observe the launch of a select projectile from an assembled catapult, the user is prompted to provide the initial time, final time, distance, in number of blocks, traveled in the X direction, and distance, in number of blocks, traveled in the Y direction. Because the grid pattern is standard and each box is five square centimeters, a distance in meters is recorded for the X and Y direction along with a value for change in time. From these values, the user is prompted to use Pythagorean's theorem to find the total distance traveled by the projectile in the observed time period. Knowing all three sides of the triangle along with the total time elapsed, the user is then able to calculate velocity and launch angle. The program calculates the value for launch angle and velocity from the provided values in the case that the user provides an incorrect value or does not know how to calculate the value. The user is given Figure 1 for visual aid.

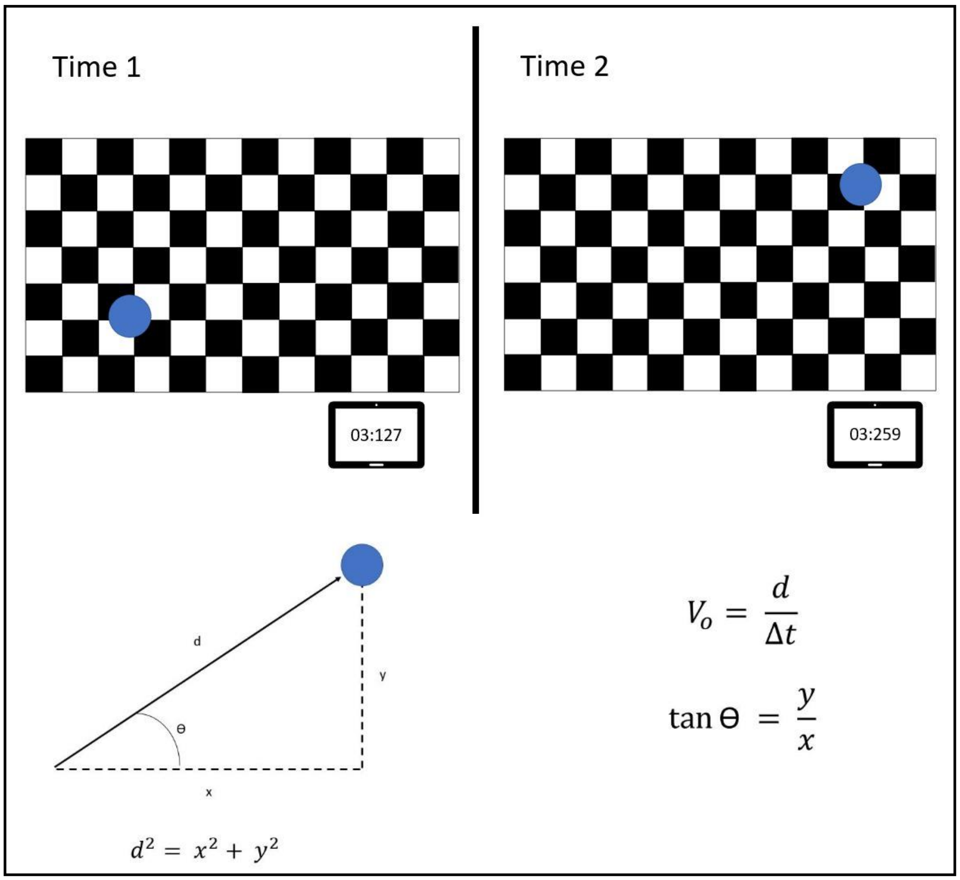

Figure 1: Visual Aid for User Calculation

From the found values of the launch angle and velocity, the program uses the Runge-Kutta method described above to solve for position, velocity, and acceleration iteratively. The values of position and velocity in both the X and Y direction at each time step are added to an individual array to create a model of the projectiles flight. The array of the Y position of the projectile is plotted versus the array of the X position of the projectile to create a 2-D graphical model of the projectile's flight as shown in figure 2. Only array positions in which the Y position array value is greater than or equal to zero are displayed. The position of the maximum height is displayed in the plot. This value is found using the position of the maximum value in the array of Y position. As well, the distance traveled is given. This value is found using the position of the first value in the Y position array in which the value is less than or equal to zero. The same position in the array of the X position array represents the total distance traveled by the projectile.

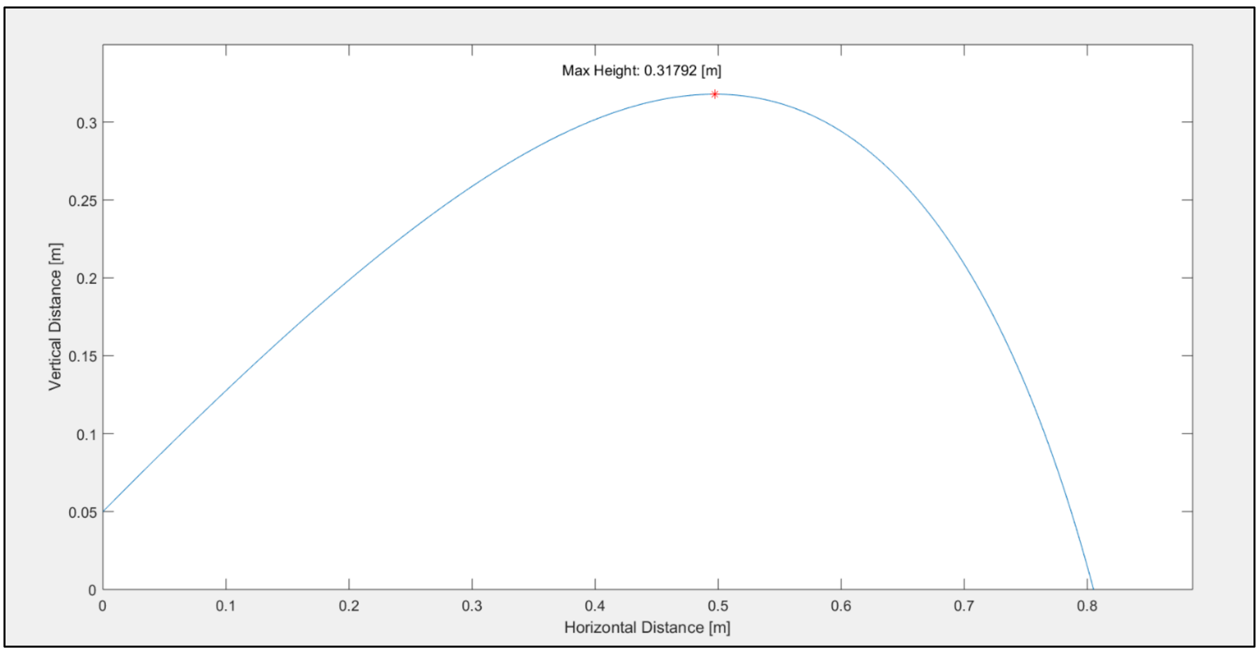

Figure 2: Sample Model Plot

With the finalized program, the objective is for the student to test different variables within the construction of the catapult provided in the Ramp Up PBLE kit to find which best meets the outcome they desire. Working with the program to test multiple variations of an assembled catapults improves the STEM skills developed in the classroom through practicing of fundamentals. Calculating the values prompted by the program with the given aid encourages improvement of students' skills in STEM. As students work through multiple models, they gain an understand of the basics of projectile motion and what variables contribute and influence different parts of a projectile's flight path. Overall, the program works to help students conceptualize projectile motion while practicing STEM topics.

```
%% Calculation

tf = ((2 * V0 * sind(T0)) / 9.81);              % Initial Guess on Total Time
x_vel = V0 * cosd(T0);                          % Initial Velocity in the X Direction
y_vel = V0 * sind(T0);                          % Initial Velocity in the Y Direction
vel = sqrt( (x_vel)^2 + (y_vel)^2 );            % Initial Velocity Magnitude
x_pos = 0;                                      % Initial X Position
y_pos = 0.05;                                   % Initial Y Position
to = 0;                                         % Initial Time
i = 0;                                          % Counting Variable

for t = 0 : .0001 : (1.5*tf)

    i = i + 1;                                  % Update Counting Variable

    x_acc = -k * vel * x_vel;                   % Calculate Acceleration in X due to Drag
    y_acc = -9.81 - ( k * vel * y_vel );        % Calculate Acceleration in Y due to Gravity and Drag
    x_vel = x_vel + ( x_acc * (t - to) );       % Update Velecity in X Direction
    y_vel = y_vel + ( y_acc * (t - to) );       % Update Velovity in Y Direction
    vel = sqrt( (x_vel)^2 + (y_vel)^2 );        % Update Velocity Magnitude
    x_pos = x_pos + ( x_vel * (t - to) );       % Find X Possition
    y_pos = y_pos + ( y_vel * (t - to) );       % Find Y Possition

    X_pos(i) = x_pos;                           % Store X Possition
    Y_pos(i) = y_pos;                           % Store Y Possition
    X_vel(i) = x_vel;                           % Store X Velocity
    Y_vel(i) = y_vel;                           % Store Y Velocity

    to = t;                                     % Update Time
end

[h , t_h] = max(Y_pos);                         % Find Max Height and Location of Max Height in the Array
xf = X_pos(find(Y_pos<=0, 1, 'first'));         % Find the FinalX Possition
```

Figure 3: Program Calculations

```
%% Calculations

error = 1;
g = -9.81;
k = 0;

while error > 0.01
    k = k + 0.001;

    y = 0.5;
    v = 0;
    to = 0;

    for t = .001 : .001 : tf

        v = v + ((1/m)*((m*g)+(k*(v^2)))*(t-to));
        y = y + (v*(t-to));

        to = t;
    end

    error = abs(y);
end
```

Figure 4: Experimental Calculation of Drag Constant K