Projectile Motion*

Muhammad Umar Hasan and Muhammad Sabieh Anwar
Center for Experimental Physics Education, Syed Babar Ali School of Science and Engineering, LUMS

V. 2017-1; May 3, 2017

The purpose of this lab is to study the motion of a projectile launched using a projectile launcher and inspect different aspects of projectile motion using high speed video processing.

Essential pre-lab reading: "*Physics for Scientists and Engineers with Modern Physics; 3rd Edition*" by Fishbane, Gasiorowicz and Thornton; (Sections 3.1-3.4)

1 Test your understanding

1. What kind of trajectory does a body follow when it is thrown in a uniform gravitational field?
2. Referring to Figure 1, write down the relations for range R, height of projectile H and time of flight t_f in terms of the angle of launch, θ_i, and the launch velocity, $\overrightarrow{v}_i$.
3. Write down an equation showing the relationship between the vertical and horizontal displacements of the projectile.
4. Write down an equation showing the relationship between the vertical displacement and the time elapsed.

2 The Experiment

Small steel balls will be fired from a Vernier projectile launcher VPL (Vernier Inc. projectile launcher) with specified angle and launch speed. See Figure 2 which shows the apparatus which will be used to throw the projectiles. The angle of launch will manually be set on the launcher using the protractor while the speed of launch will be recorded using a Vernier Lab Quest module

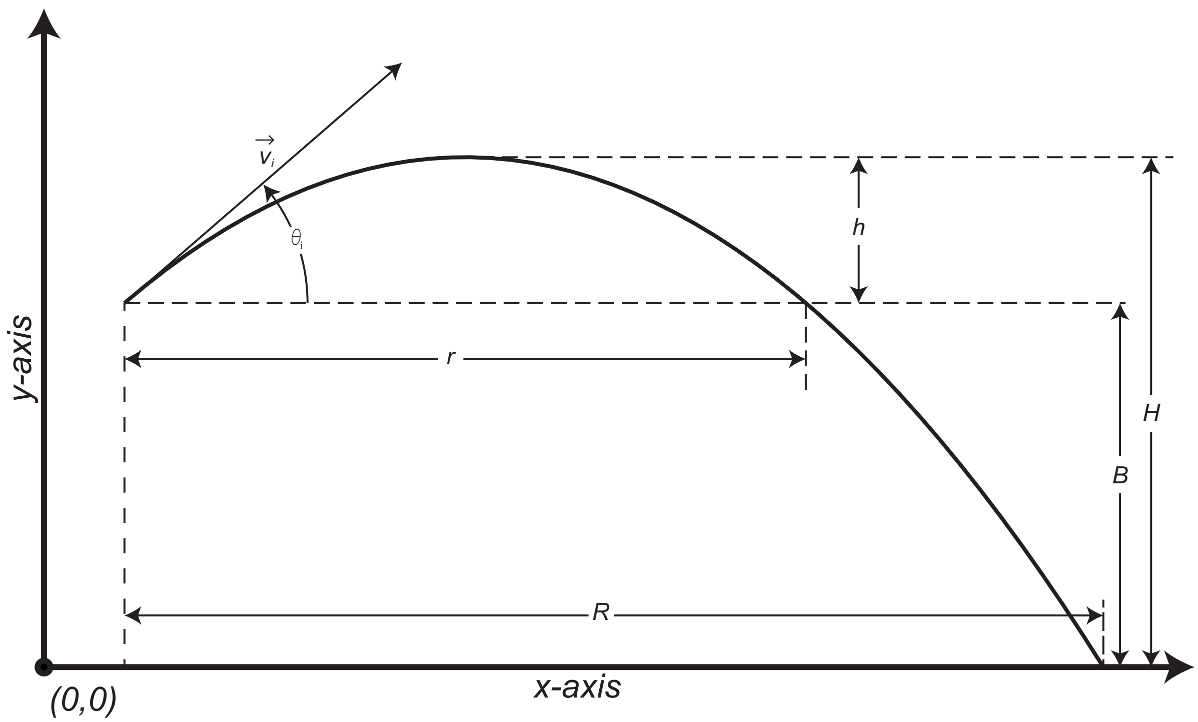

Figure 1: A diagram representing the trajectory of a projectile and different parameters associated with this kind of motion.

which will connect the VPL with a PC. Vernier's Data Logger software will be used to view these parameters on the PC. The entire flight of the projectile will simultaneously be captured on a high speed camera. This video will then be processed using PhysLab's video tracking library "PhysTrack". Subsequent position data generated will be used to further investigate various aspect of the motion.

Place the projectile launcher on a firm surface with a plain background wall. Use the spirit level attached to `the gun` to calibrate the protractor with the horizon. Once calibrated, tighten the protractor using the knobs. Connect Lab Quest module to the VPL using the digital cable and to the PC using the USB cable. Open up the data Logger software in the PC and confirm that the launcher is connected properly. Connect the hand pump with the launcher carefully and tighten the union with hands.

Setup the camera to high-speed video mode at $240fps$ and fix it on a tripod stand. Place the tripod stand 5 to 6 feet away from the apparatus. Use bull's eye level to confirm that the camera is placed horizontally. The image in the camera's view finder should look similar to the one shown in Figure 3. Adjust the height of the camera such that the view covers most of the portion where the projectile is likely to cruise.

Using a spirit level, confirm that the landing surface is horizontal. Mark a known distance on the landing surface and a reference line to serve as the plane of the projectile. See Figure 4 which illustrates the top view of the apparatus.

Figure 2: The Vernier projectile launcher (VPL) along with the air pump and the Bourdon pressure gauge.

Caution: Wear safety glasses before proceeding further.

To obtain a reading follow these steps.

1. Clean the steel ball using a tissue paper to remove any kind of dust particles or fibers from the surface.

2. Stop any on-going measurement in the Data Logger software.

3. Gently push the ball into the launcher and wait for it to settle down.

4. Adjust a desired inclination on the projectile launcher and firmly tighten the knobs to fix this orientation. Note down this angle in your notebook.

5. Use the hand pump to pressurize the launcher. Keep on pumping unless you hear a faint pressure release sound. Look at the Bourdon pressure gauge to confirm that the pressure reading has stalled. Pump the air twice more to hear the same sound again. Use the pressure threshold adjustment knob on the projectile launcher to change the pressure threshold. For the first reading, don't use pressure more than 50 psi. Note down the pressure reading in your notebook.

6. Start acquiring data the Data Logger software.

7. Get help from your partner to start the video recording right before you launch the projectile. Once the video recording has started, push `arm` and `launch` buttons on the projectile launcher simultaneously to launch the projectile.

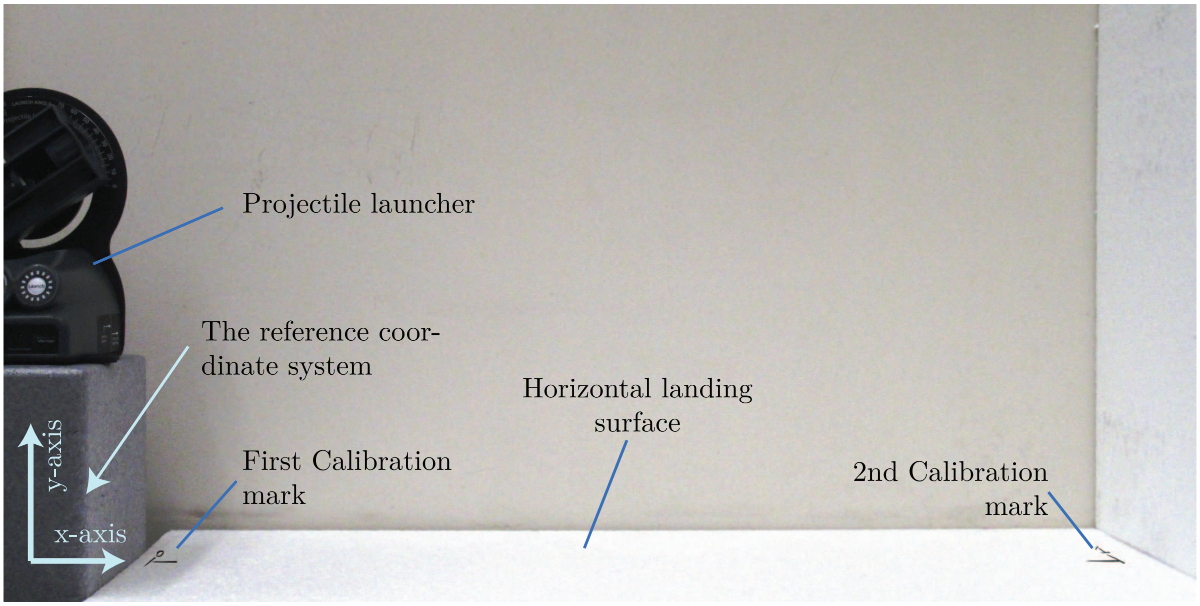

Figure 3: The experimental setup showing the projectile launcher, the landing area and the distance calibration marks.

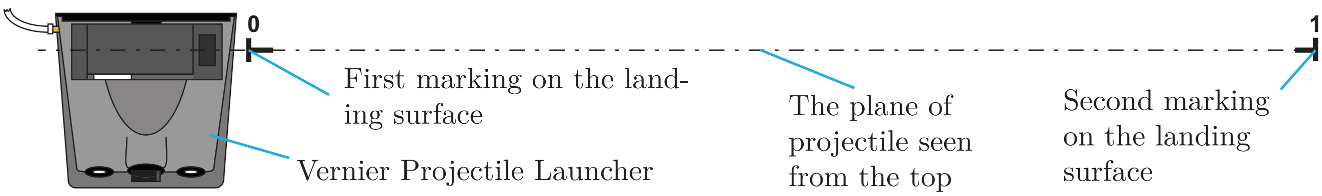

Figure 4: The top view of the VPL apparatus. See the way the first and second calibration marks have to be placed.

8. Stop the video recording once the ball has hit the soft ground. Keep an eye on the ball so that it doesn't hit any hard surface like the cemented floor. Any kind of surface indentation on the ball can deleteriously affect subsequent launches.

9. Note down the initial velocity from the data logger software in your notebook.

Take as many readings with different launch speeds and angles as necessary.

3 Analysis

3.1 Preparing Data for Quantitative Analysis

I hope the reader has already studied our "Primer on video tracking" uploaded on the website.

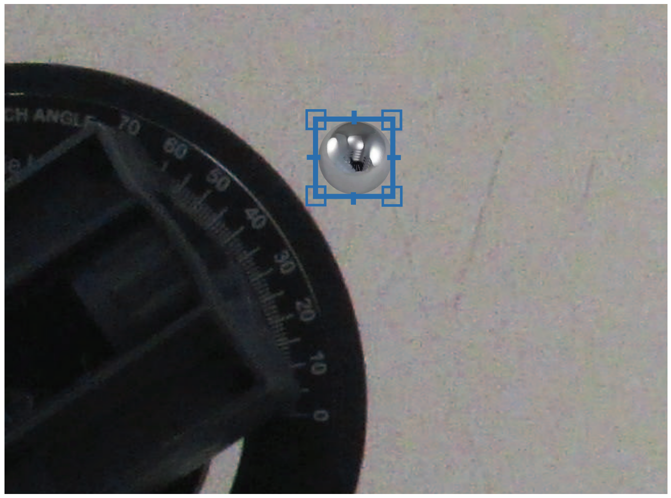

Figure 5: Identifying the projectile in object selection tool. Note that the image has been artificially enhanced for visual clarity.

Copy the videos to some appropriate location in the hard-drive. Download and extract on hard-drive the latest release of PhysTrack from our website. In Matlab, browse to the extracted PhysTrack directory where you should see the "+PhysTrack" folder and some "analyze motion scripts" in the "Current Folder" window. Run the script `analyzeProjectileMotion` which presents a series of GUI tools to setup and perform video tracking. When asked by the interface, browse and select your video file. The script will automatically show a tool to trim and crop the video. Use the `slider` and the `go-to buttons` to seek different frames of the video. Mark as `In-Frame` the moment when the ball has just left the projectile launcher and as `Out-Frame` the moment you want to stop tracking the ball. Preview the trimmed video if necessary and close the tool afterwards.

The script will now present a coordinate system tool. Follow the on-screen instructions to setup a coordinate system having the `x-axis` aligned horizontally and in the same direction as that of the projectile. While the `+y-axis` should be upwards. Use the `Reset Origin`, `Reset x-axis` and `Toggle Direction` buttons to move the origin, rotate the x-axis about the origin and toggle the y-axis direction respectively. See figure 3 which guides how to set up the coordinate system. Enter the value of the known distance marked on the horizontal surface in text-box labelled `Marked Distance` and click on the `Reset Unit Marking` button to draw a line on the two markers with known real distance marked on the landing surface, followed by the `enter` key. Repeat any of these steps as many time as needed to achieve the desired results. Once everything is final, close the tool.

The script will now open an object selection window for marking the projectile. Click `Manually Mark an Object` and drag and draw a tight square around the ball as shown in Figure 5. Marking the rectangle again on the same place replaces the previous one automatically. When you close the tool to start the tracking process, there should only be one object in the objects listbox.

When the tracking is finished the script will open up a track point filter tool to manually remove any stray points. Use the on-screen controls to seek different instances of the video and filter out the unwanted track points by drawing rectangles around them. Each step in this tool can be undone using the `Undo` button. At the end of a successful run, the script will leave some variables in the workspace. See Table 1 for details of these variables.

Physical Quantity	Variable Names
Trajectory of the projectile	*trajectory*
`xdata` of the trajectory	dx
`ydata` of the trajectory	dy
Time stamps	t

Table 1: Base workspace variables generated by the script.

Now we need to use this positional data and the respective time stamps to calculate the velocity and acceleration.

3.2 Quantitative Analysis

Plot `ydata` against `xdata` to visually observe if the projectile has followed a parabolic path. To confirm this assumption, we need to fit the `xdata` and the `ydata` into the model.

Q 1. Why do you think that Equation (1) given below is a valid model for the height of the projectile?

$$y = a_0 + a_1 x + a_2 x^2 \tag{1}$$

You will fit the data to this model using the least-squares fitting function `PhysTrack.lsqCFit`. The syntax for using this function is

```
fitResult = PhysTrack.lsqCFit(xdata, ydata, 'y', 'a0 + a1 * x + a2 * x^2', 'x');
```

The variable `fitResult` is a `cfit` object which contains separate variables for each unknown parameter. The syntax to evaluate the value of the model function at a specific value of x is

```
y_At_x = fitResult(x);
```

Q 2. Write down a model function for the height in terms of the elapsed time. Is your model same as Equation (2)?

$$y = b_0 + b_1 t + b_2 t^2, \tag{2}$$

Q 3. What do the coefficients b_0, b_1 and b_2 represent? How are they related to v_i and θ_i?

Q 4. Fit the height data, `ydata`, and the elapsed time, `t`, to Equation (2).

Q 5. Use the two model functions and the values of the constants to interpolate the data for each of the datasets.

You will now differentiate `xdata` and `ydata` with respect to time `t` to obtain the velocity using the `PhysTrack.deriv` function. Here is an example on how to use this function.

```
[td, xd] = PhysTrack.deriv(t, xdata, order);
% and for y data,
[td, yd] = PhysTrack.deriv(t, ydata, order)
```

where order is 1 for computing a first-order derivative and 2 for a second-order derivative.

Q 6. Compute the first order derivatives for each dataset. Plot your results to obtain the graphs for the vertical and horizontal velocities.

Q 7. Using the `lsqfun3` function, fit the velocity data to Equation (3) and plot the results.

$$y = c_0 + c_1 t. \tag{3}$$

Q 8. Explain your findings about the vertical and horizontal displacement, velocity and acceleration of the projectile.

Q 9. Compare the expected and real values of the range, the height of the projectile and the total time of flight. What are the factors which have played a role in altering the results?

Q 10. How can you verify the energy balance of the projectile? Using the data collected from the experiment, explain how the kinetic and potential energies inter-convert into each other.

Q 11. Using the velocity fit results, calculate the velocity of the projectile when it hits the landing surface.