

---

## Table of Contents

Ball Bounce Project .....	1
Initial Setup .....	1
Determine whether or not the ball will clear the obstacle .....	1
Solve for the Max Height of the Ball's Trajectory .....	2
Plot Best Path and actual .....	3
Print out a summary of the data .....	4

## Ball Bounce Project

Code By: Brandon Ching Group: Ben Bartlett, Sarah Ford, James Dornblaser

```
clc; clear; close all;
```

### Initial Setup

```
e = [0.894153; 0.809083]; % COR of ball/plate
h = 0.548; % height at which ball is dropped from above impact point
obj_x = 0.456; %x length of the object placement from impact
obj_y = 0.062; %y height of the object placement from impact
g = 9.81; % g, gravity, m/s^2
y0 = 0; % y, initial vertical position, m
x0 = 0; % x, initial horizontal position, m
alpha = 0:0.01:90; %vary the plate angle from flat to vertical
```

### Determine whether or not the ball will clear the obstacle

Solve for the exit velocity from the plate and the corresponding angle

```
v_2 = (e .* cosd(alpha) * sqrt(2*g*h)) ./
    (sind(atan2(e*cotd(alpha)))));
angle = atan2(e*cotd(alpha)) - alpha;

% Determine whether or not the ball clears the obstacle
y_clear = -((g*obj_x^2) ./ (2 * v_2.^2 .* cosd(angle).^2)) + (obj_x .*
    tand(angle));
cleared = y_clear > obj_y;

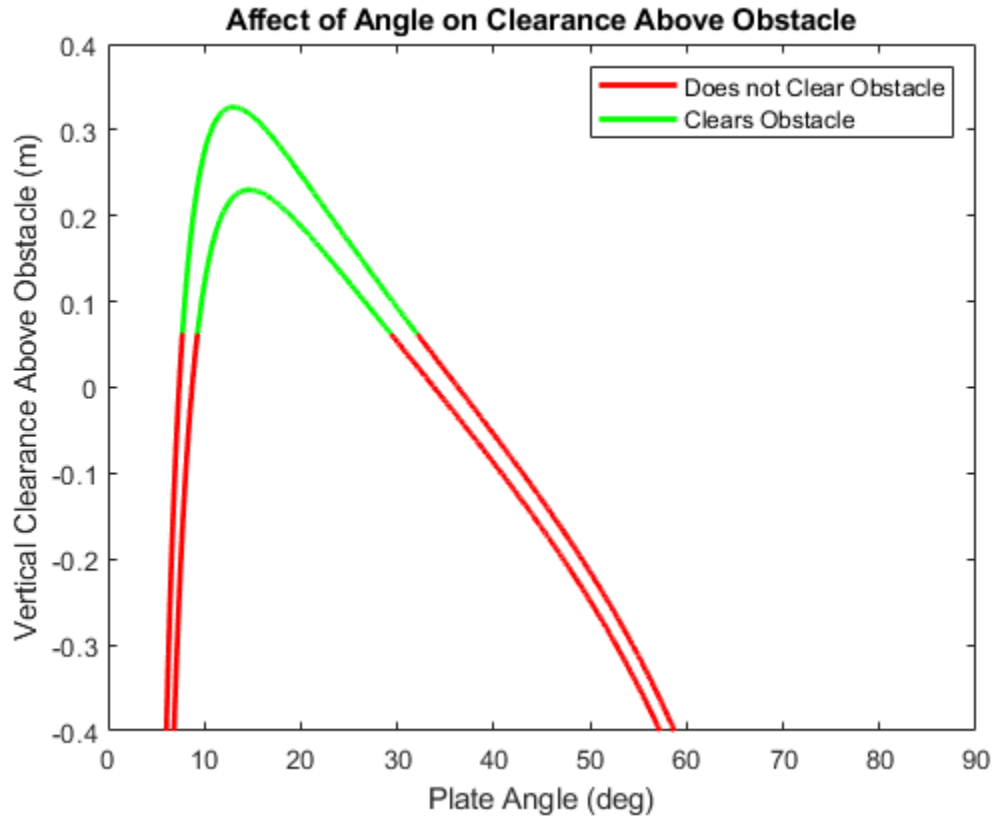
figure % New figure of angle vs how much the ball clears the obstacle
plot(alpha, y_clear, 'r', 'LineWidth', 2); hold on;
plot(alpha(cleared(1,:)), y_clear(1,cleared(1,:)), 'g', 'LineWidth', 2);
plot(alpha(cleared(2,:)), y_clear(2,cleared(2,:)), 'g', 'LineWidth', 2); hold off;
% Label Plot and Define Axes
xlabel("Plate Angle (deg)")
ylabel("Vertical Clearance Above Obstacle (m)")
title("Affect of Angle on Clearance Above Obstacle")
```

---

```

legend('Does not Clear Obstacle','','Clears Obstacle');
axis([0 90 -0.4 0.4])
text([15 15],y_clear(:,15),{'\leftarrow Aluminium/Nylon','\leftarrow
Acrylic/Delrin'})

```



## Solve for the Max Height of the Ball's Trajectory

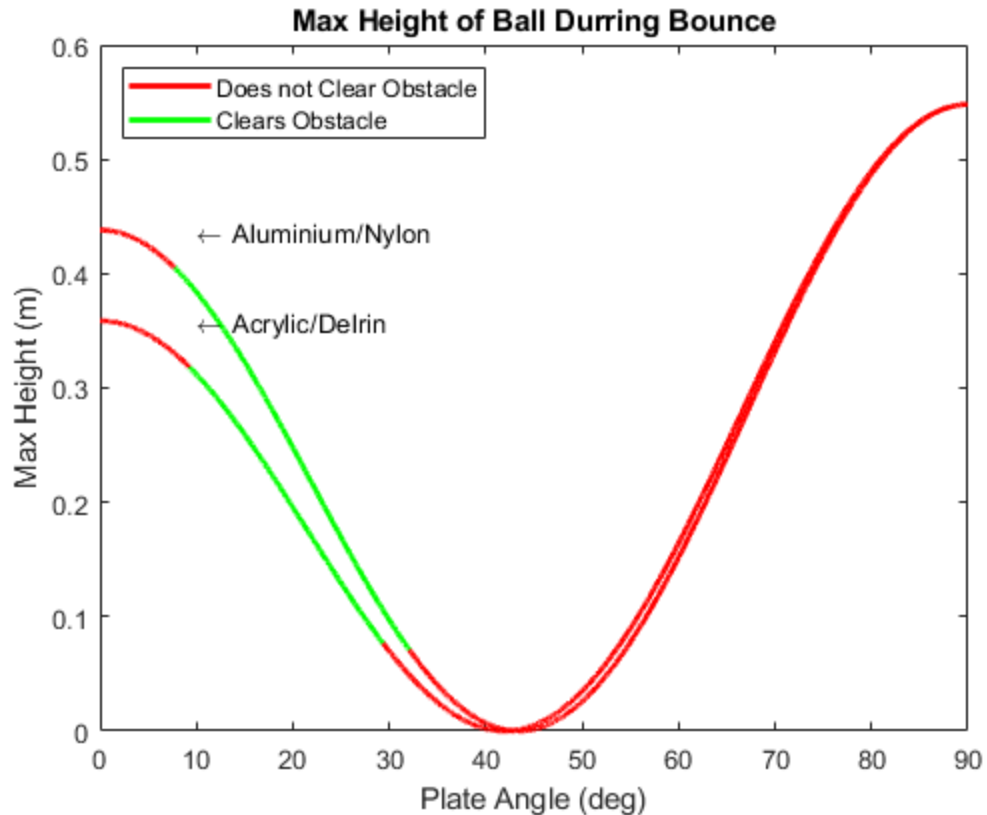
Max height of the ball's trajectory for each plate angle

```

h_max = (v_2.^2 .* sind(angle).^2) ./ (2*g);

figure % New figure comparing plate angle with max height
plot(alpha, h_max, 'r', 'LineWidth', 2); hold on; % Failed Angles
plot(alpha(cleared(1,:)),h_max(1,cleared(1,:)), 'g', 'LineWidth', 2);
plot(alpha(cleared(2,:)),h_max(2,cleared(2,:)), 'g', 'LineWidth', 2);
hold off;
% Label Plot and Define Axes
xlabel("Plate Angle (deg)")
ylabel("Max Height (m)")
title("Max Height of Ball Durring Bounce")
legend('Does not Clear Obstacle','','Clears
Obstacle','Location', 'NW');
text([10 10],h_max(:,10),{'\leftarrow Aluminium/Nylon','\leftarrow
Acrylic/Delrin'})

```



## Plot Best Path and acutal

Find the best plate angle, alpha, and the corrsponding exit velocity and angle for simulated trajectory

```
best_alpha = min(alpha(logical(cleared(1,:))));
best_v_2 = v_2(1,(find(alpha==best_alpha)));
best_angle = angle(1,(find(alpha==best_alpha)));
% Create a time array from 0 to time of impact, with 1000 data points
t_f = (2 * best_v_2 * sind(best_angle))/g;
t = linspace(0,t_f,1000);
% Generate array of X and Y position of simulated trajectory as a
function
% of time
x = best_v_2 * cosd(best_angle) * t;
y = -((1/2) * (g * t.^2)) + (best_v_2 .* sind(best_angle) .* t);

% X and Y Cord of the Actual ball's path as produced by Physlet
Tracker
x_act = [0 0.02514 0.0565 0.08277 0.113 0.142 0.180 0.203 0.233 0.262
0.294 0.322 0.352 0.379 0.402 0.43 0.45 0.472 0.490];
y_act = [0 0.07294 0.166 0.23 0.29 0.34 0.383 0.401 0.412 0.41 0.395
0.372 0.334 0.285 0.234 0.165 0.111 0.0229 -0.05723];

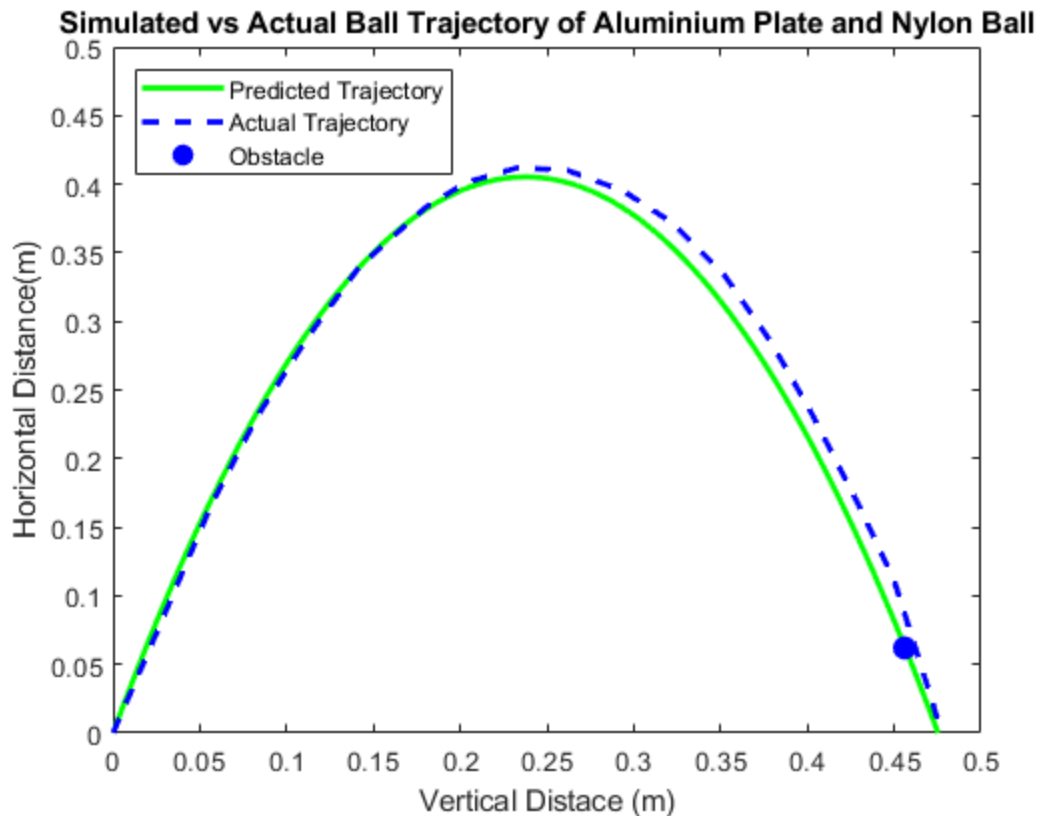
figure % New figure of simulated vs actual trajectory
plot(x, y, 'g', 'LineWidth', 2); hold on; % Simulated Trajectory
```

---

```

plot(x_act, y_act, 'b--', 'LineWidth' , 2); hold on; % Actual
trajectory
plot(obj_x,obj_y, 'b.', 'MarkerSize', 30); hold off; % Obstacle
% Label Plot and Define Axes
xlabel("Vertical Distace (m)")
ylabel("Horizontal Distance(m)")
title("Simulated vs Actual Ball Trajectory of Aluminium Plate and
Nylon Ball")
legend('Predicted Trajectory','Actual
Trajectory', 'Obstacle', 'Location', 'NW');
axis([0 0.5 0 0.5])

```



## Print out a summary of the data

```

fprintf("With a COR of e=%f, this plate/ball combination can
clear an obstacle %.3fm away and %.3fm high when the plate
angle is between %.2f-%.2f degrees.\n",e(1),obj_x,obj_y,
min(alpha(logical(cleared(1,:)))), max(alpha(logical(cleared(1,:))))))
fprintf("The best angle to clear the obstacle & maximize height is
%.2f degrees with a height of %fm.\n", best_alpha, max(h_max(1,
(logical(cleared(1,:))))))

```

*With a COR of  $e=0.894153$ , this plate/ball combination can clear an obstacle 0.456m away and 0.062m high when the plate angle is between 7.72-32.07 degrees.*

---

*The best angle to clear the obstacle & maximize height is 7.72 degrees  
with a height of 0.405275m.*

*Published with MATLAB® R2021a*