

The graph above shows the different trajectories of each golf ball.   
The blue line is with no spin and was dimpled. The angle used was 0.575 radians. The max range was 184.05m and the max height was 55.232m.

The red line is with no spin and no dimples. The angle used was 0.6 radians. The max range was 92.338m and the max height was 35.329m.

The green line is with spin and dimples. The angle used was 0.1 radians. The max range was 205.14m and the max height was 39.336m.

The magenta line is with spin and no dimples. The angle used was 0.3 radians. The max range was 101.56m and the max height was 32.411m.

The yellow line is with the doubled spin and dimples. The angle used was 0 radians. The max range was 177.97m and the max height was 77.462m.

**Code**

%\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*%

%\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Projectile Motion Lab \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*%

% Jonathan Shipley

% Scientific Modeling

% 2/14/17

% Golf ball can be dimpled and undimpled. Do projectile motion of a golf ball including

% air resistance and the Magnus Force. Assume it is hit from and lands at the same level

% Important Parameters:

% Dimpled ball: C = 1/2 for v = 14m/s and under. C = 7/v for above 14 m/s

% Undimpled ball: C = 1/2 for all speeds

% Assume ball is hit at 70 m/s and golf ball mass = 45.93 g and radius is 2.133 cm

% Input: Angle from ground that ball is hit from

% Output: Plot of y vs x

% Max height and range

%\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*%

%\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*%

% Variable Declaration and initialization

t = 0:0.01:300;

dt = 0.01;

g = 9.81;

times = 300 /dt + 2;

m = 0.04593; % kg

r = 0.02133; % m

rho = 1.293; % kg/m^3

A = pi \* r\*r; % Surface area of the ball

R = rho \* A \* (1/m);

MFF = 0.25; % Magnus force factor

% Get user input for angle and for dimpled, and spin

th = input('Input initial angle in radians: ');

spin = input('Is the ball spinning: (y/n) ', 's');

dim = input('Is the ball dimpled: (y/n) ', 's');

maxRange = 0;

maxHeight = 0;

vi = 70; % m/s

vx = vi\*cos(th);

vy = vi\*sin(th);

x = 0;

y = 0;

if dim == 'y'

if spin == 'y'

% Do dimpled with back spin

for k = 1:times

% find v

v = sqrt(vx(k)\*vx(k) + vy(k)\*vy(k));

% Check for c

if v <= 14

C = 0.5;

else

C = 7/v;

end

% update velocities

vx(k+1) = vx(k) - (C\*R\*vx(k)\*vx(k)\*dt) - MFF \* vy(k) \* dt;

vy(k+1) = vy(k) - g\*dt - (C\*R\*vy(k)\*vy(k)\*dt) + MFF \* vx(k) \* dt;

% update positions

x(k+1) = x(k) + vx(k+1)\*dt;

y(k+1) = y(k) + vy(k+1)\*dt;

if y(k+1) < 0

y(k+1) = 0;

break;

end

end

end % end for spin =='y'

if spin =='n'

% Do dimpled with no back spin

for k = 1:times

% find v

v = sqrt(vx(k)\*vx(k) + vy(k)\*vy(k));

% Check for c

if v <= 14

C = 0.5;

else

C = 7/v;

end

% update velocities

vx(k+1) = vx(k) - (C\*R\*vx(k)\*vx(k)\*dt);

vy(k+1) = vy(k) - g\*dt - (C\*R\*vy(k)\*vy(k)\*dt);

% update positions

x(k+1) = x(k) + vx(k+1)\*dt;

y(k+1) = y(k) + vy(k+1)\*dt;

if y(k+1) < 0

y(k+1) = 0;

break;

end % end for if

end % end for for

end % end for spin == 'n'

end % end for dimpled

if dim == 'n'

if spin == 'y'

% Do undimpled with back spin

for k = 1:times

% find v

v = sqrt(vx(k)\*vx(k) + vy(k)\*vy(k));

C = 0.5;

% update velocities

vx(k+1) = vx(k) - (C\*R\*vx(k)\*vx(k)\*dt) - MFF \* vy(k) \* dt;

vy(k+1) = vy(k) - g\*dt - (C\*R\*vy(k)\*vy(k)\*dt) + MFF \* vx(k) \* dt;

% update positions

x(k+1) = x(k) + vx(k+1)\*dt;

y(k+1) = y(k) + vy(k+1)\*dt;

if y(k+1) < 0

y(k+1) = 0;

break;

end

end

end

if spin == 'n'

% Do undimpled wtih no back spin

for k = 1:times

% find v

v = sqrt(vx(k)\*vx(k) + vy(k)\*vy(k));

% Check for c

C = 0.5;

% update velocities

vx(k+1) = vx(k) - (C\*R\*vx(k)\*vx(k)\*dt);

vy(k+1) = vy(k) - g\*dt - (C\*R\*vy(k)\*vy(k)\*dt);

% update positions

x(k+1) = x(k) + vx(k+1)\*dt;

y(k+1) = y(k) + vy(k+1)\*dt;

if y(k+1) < 0

y(k+1) = 0;

break;

end

end

end

end

% Find max range and max height

max(x)

max(y)