AE332 – Modeling and Analysis Lab

Name: Shingala Vaidik ID : SC21B054

- **Comparison of the paths taken by the cannonball in atmosphere and in vacuum.**

1. **Initial velocity and angle of elevation (with respect to horizontal) to send the cannonball about 20 Km away (initial energy as low as possible) in vaccum:**

$$V = 443\frac{m}{s}; \qquad \theta = pi/4;$$

Matlab Code:

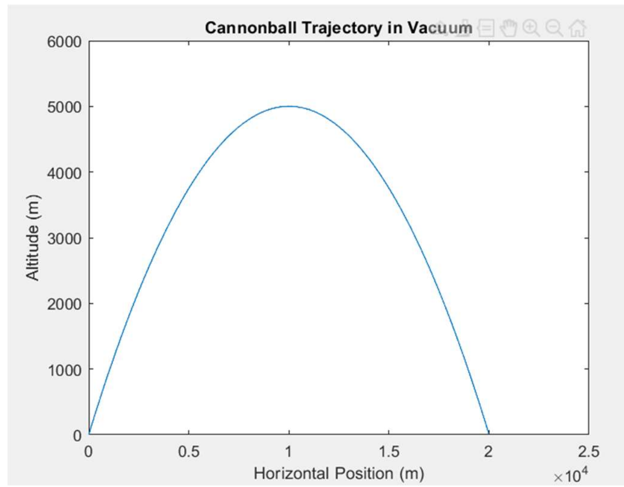
```
% Constants
g = 9.81;   % Acceleration due to gravity (m/s²)

% Define the system of differential equations
f = @(t, z) [z(3,1); z(4,1); 0; -g];

% Initial conditions
hori0 = 0;      % Initial horizontal position (m)
vy0 = 443 * sin(pi/4); % Initial vertical velocity (m/s)
vert0 = 0;      % Initial vertical position (m)
vx0 = 443 * cos(pi/4); % Initial horizontal velocity (m/s)

% State vector
z0 = [hori0; vert0; vx0; vy0];

% Time span for simulation
t = 0:0.01:81.8;
tol1=odeset('RelTol',1e-12,'AbsTol',1e-12);
[t, z] = ode45(f, t, z0, tol1);


i=1;
z1=z(:,1);
z2=z(:,2);

horizontal_position=zeros;
altitude=zeros;

while(z2(i)>=0)
    horizontal_position(i)=z1(i);
    altitude(i)=z2(i);
    i=i+1;
end

Range = max(horizontal_position);
disp(['Range in vaccum: ' num2str(Range) ' meters']);

% calculating error in Energy for accuracy of solution
TotalE1=0.5*40*(z(:,3).^2+z(:,4).^2) + 40*g*z(:,2);
```

```
error1=max(TotalE1)-min(TotalE1);
disp(['error in Energy: ' num2str(error1)]);

% Plot the results
clf();
plot(horizontal_position, altitude);
xlabel('Horizontal Position (m)');
ylabel('Altitude (m)');
title('Cannonball Trajectory in Vacuum');
```

Output:

```
>> abc
Range in vaccum: 20004.0367 meters
error in Energy:: 3.2596e-09
```

Graph:



Here we can see that error between max. and min. energy is very less so our solution is accurate.

2. **Now considering Atmosphere: For that we have to consider Drag**

Matlab code:

```
% Initial conditions
hori0 = 0;      % Initial horizontal position (m)
vy0 = 443 * sin(pi/4); % Initial vertical velocity (m/s)
vert0 = 0;      % Initial vertical position (m)
vx0 = 443 * cos(pi/4); % Initial horizontal velocity (m/s)
whos("atmsphr1.m","atmsphr","height","Temp","P","density");
height=atmsphr(:,1);
Temp=atmsphr(:,2);
density=atmsphr(:,4);

[t,z]=ode45(@(t,z) f(t,z,height,Temp, density),(0:0.1:122),[0 0 vx0
vy0 0]);
i=1;
z1=z(:,1);
z2=z(:,2);
horizontal_position=zeros;
altitude=zeros;
```

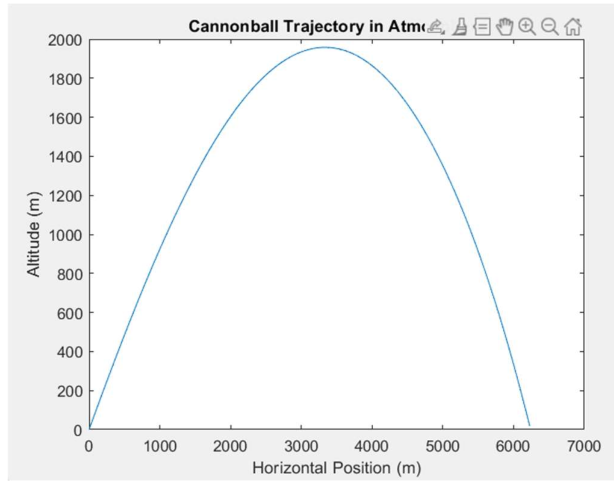
```
while(z2(i)>=0)
    horizontal_position(i)=z1(i);
    altitude(i)=z2(i);
    i=i+1;
end

Range = max(horizontal_position);
disp(['Range in Atmosphere: ' num2str(Range) ' meters']);

% calculating Energy loss
Energyloss= 0.5*40*((z(273,3)^2+z(273,4)^2)-(z(1,4)^2+z(1,4)^2));
disp(['Energy Loss: ' num2str(Energyloss) ' meters']);

% calculating error in Energy for accuracy of solution
TotalE2=0.5*40*(z(:,3).^2+z(:,4).^2) + 40*9.81*z(:,2) + z(:,5);
error2=max(TotalE2)-min(TotalE2);
disp(['error in Energy: ' num2str(error2)]);

% Plot the results
 clf();
 plot(horizontal_position, altitude);
 xlabel('Horizontal Position (m)');
 ylabel('Altitude (m)');
 title('Cannonball Trajectory in Atmosphere');


function CD = calculate_CD(M, R)
    % Coefficients for CD calculation
    c = [0.0000641, -0.0006166, -0.0075524, 0.1175224, -0.0009032,
0.0136069, 0.0086353, -1.0093621, 0.0041238, -0.0858483,0.4289429,
1.5431515, -0.0064570, 0.1778755, -1.5559375, 4.0394577];
    CD = c(1)*M^3*R^3 + c(2)*M^3*R^2 + c(3)*M^3*R + c(4)*M^3 +
c(5)*M^2*R^3 + c(6)*M^2*R^2 + c(7)*M^2*R + c(8)*M^2 + c(9)*M*R^3 +
c(10)*M*R^2 + c(11)*M*R + c(12)*M + c(13)*R^3 + c(14)*R^2 + c(15)*R
+c(16);
end

function dzdt=f(~,z,height,Temp, density)
    m=40;
    d=0.22;
    A=(pi*d^2)/4;
    g=9.81

    temp=interp1(height,Temp,z(2,1)/1000);
    density=interp1(height,density,z(2,1)/1000);

    mu=(1.458e-06*temp^1.5)/(temp+110.4);
    Re=density*(sqrt(z(3,1)^2+z(4,1)^2))*d/mu;

    R=log(Re);
    M=sqrt(z(3,1)^2+z(4,1)^2)/(sqrt(1.4*287*temp));

    % Coefficients for CD calculation
     CD = abs(calculate_CD(M, R));
   dzdt(1,1)=z(3,1);
   dzdt(2,1)=z(4,1);
   dzdt(3,1)=-0.5*(density/m)*A*CD*(sqrt(z(3,1)^2+z(4,1)^2))*z(3,1);
```

```
 dzdt(4,1)=-g-0.5*(density/m)*A*CD*(sqrt(z(3,1)^2+z(4,1)^2))*z(4,1);
   dzdt(5,1)=-0.5*(density/m)*A*CD*(sqrt(z(3,1)^2+z(4,1)^2))^3;
end
```

Output:

```
>> se02que2

Range in Atmosphere: 6228.2372 meters
Energy Loss: -3365194.6742 meters
error in Energy: 3.87399e-06
```

Graph:



- **Energy Loss by the ball by the end of trajectory**

$$\boldsymbol{Energyloss = 0.5mV_{end}{}^2 - 0.5mV_{start}{}^2}$$

```
% calculating Energy loss
Energyloss= 0.5*40*((z(273,3)^2+z(273,4)^2)-(z(1,4)^2+z(1,4)^2));
disp(['Energy Loss: ' num2str(Energyloss) ' meters']);
```

```
Energy Loss: -3365194.6742 meters
```

- **Accuracy of Solution**

$$\boldsymbol{Total\ Energy = 0.5mV_{end}{}^2 + mgH + E_{dissipation}}$$

1. For vaccum : No dissipation in Energy

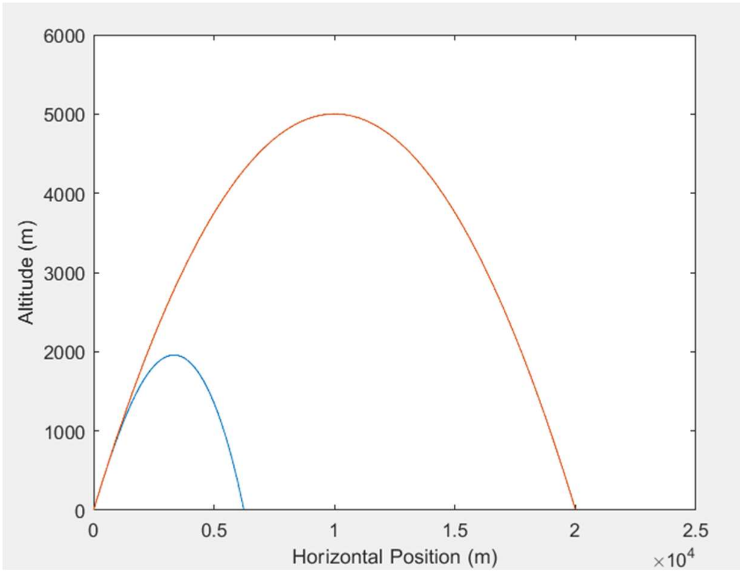
```
% calculating Energy loss more accuracy of solution
TotalE1=0.5*40*(z(:,3).^2+z(:,4).^2) + 40*g*z(:,2);
error1=max(TotalE1)-min(TotalE1);
disp(['error in Energy: ' num2str(error1)]);
```

```
error in Energy:: 3.2596e-09
```

2. For Atmosphere:

```
    % calculating error in Energy for accuracy of solution
TotalE2=0.5*40*(z(:,3).^2+z(:,4).^2) + 40*9.81*z(:,2) + z(:,5);
    error2=max(TotalE2)-min(TotalE2);
    disp(['error in Energy: ' num2str(error2)]);
```

```
error in Energy: 3.87399e-06
```



Range in vaccum:
20004.0367 meters

Range in Atmosphere:
6228.2372 meters

- **Find the Mach number and Reynolds number variations along the trajectory, and also the drag and CD variation.**

Now we have height variation of ball with us, so for graph of Mach number, Reynolds number, drag and CD variations along the trajectory, we have to just call pervious code height result in this code.

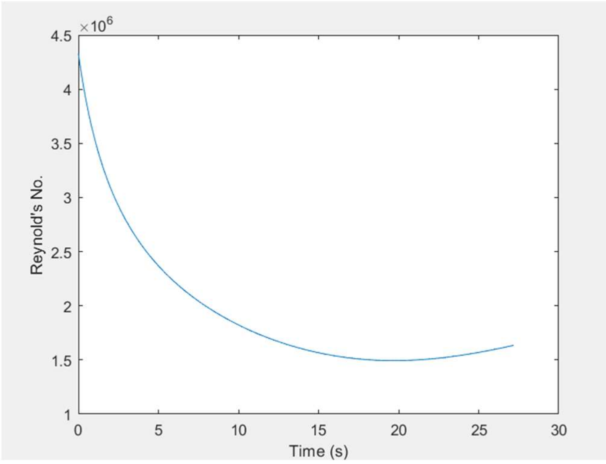
Matlab code:

```
whos("atmsphr1.m","atmsphr","height","Temp","P","density");

g=9.8;
m=40;%kg
beta=1.458e-06;
d=0.155;%diameter in m
temp=interp1(height,Temp,z(2,1)/1000);
density=interp1(height,density,z(2,1)/1000);
a=sqrt(1.4*287.*temp);
mu=beta*temp.^1.5./(temp+110.4);
M=sqrt(z(:,3).^2+z(:,4).^2)./a;
Re=density.*sqrt(z(:,3).^2+z(:,4).^2)*d./mu;

plot(t,Re)
xlabel("Time (s)");
ylabel("Reynold's No.");
R=log(Re);

c = [0.0000641, -0.0006166, -0.0075524, 0.1175224, -0.0009032,
0.0136069, 0.0086353, -1.0093621, 0.0041238, -0.0858483,0.4289429,
1.5431515, -0.0064570, 0.1778755, -1.5559375, 4.0394577];
CD = c(1).*M.^3.*R.^3 + c(2).*M.^3.*R.^2 + c(3).*M.^3.*R +
c(4).*M.^3 + c(5).*M.^2.*R.^3 + c(6).*M.^2.*R.^2 + c(7).*M.^2.*R +
```

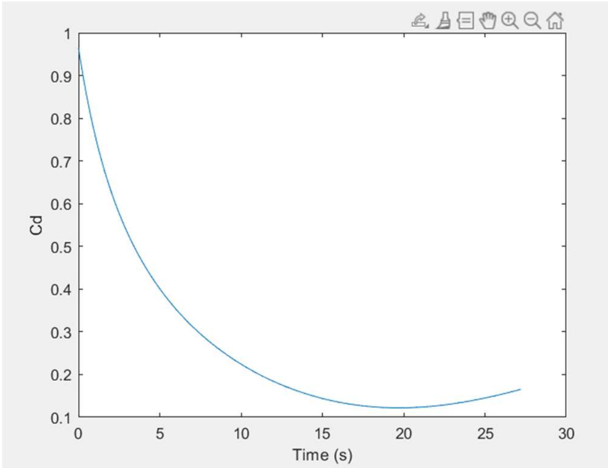
```
c(8).*M.^2 + c(9).*M.*R.^3 + c(10).*M.*R.^2 + c(11).*M.*R + c(12).*M
+ c(13).*R.^3 + c(14).*R.^2 + c(15).*R +c(16);
plot(t,CD)
xlabel("Time (s)");
ylabel("Cd");

s=(pi*d^2)/4;
D=0.5.*CD.*density.*(z(:,3).^2+z(:,4).^2).^2.*s;
plot(t,D)
xlabel("Time (s)");
ylabel("Drag");

plot(t,M)
xlabel("Time (s)");
ylabel("Mach Number");
```

output:

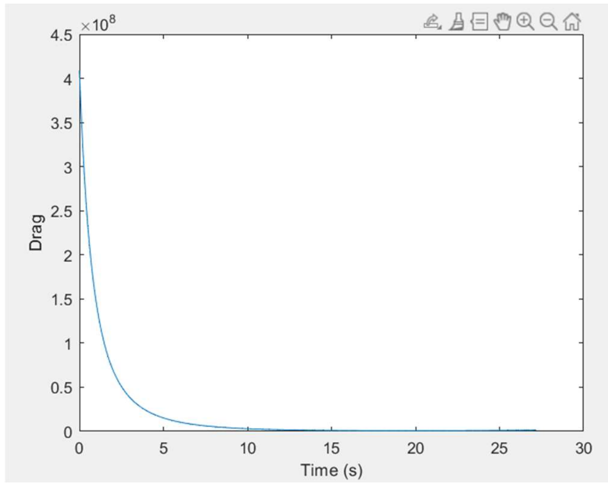
- **Reynolds number variations along the trajectory**



- **Coefficient of Drag (Cd) variations along the trajectory**



- **Drag variations along the trajectory**



- **Mach No. variations along the trajectory**