AIRSOFT ROUND AERODYNAMICS

Student Name	:	**ENES ŞAHİNER**
Student ID No	:	**030200018**
Due Date	:	**06.04.2023**
Lecture Name	:	**MAK 202E NUMERICAL METHODS**
Lecturer	:	**ALPER TOLGA ÇALIK**
Course CRN	:	**22194**

PROBLEM DEFINITION

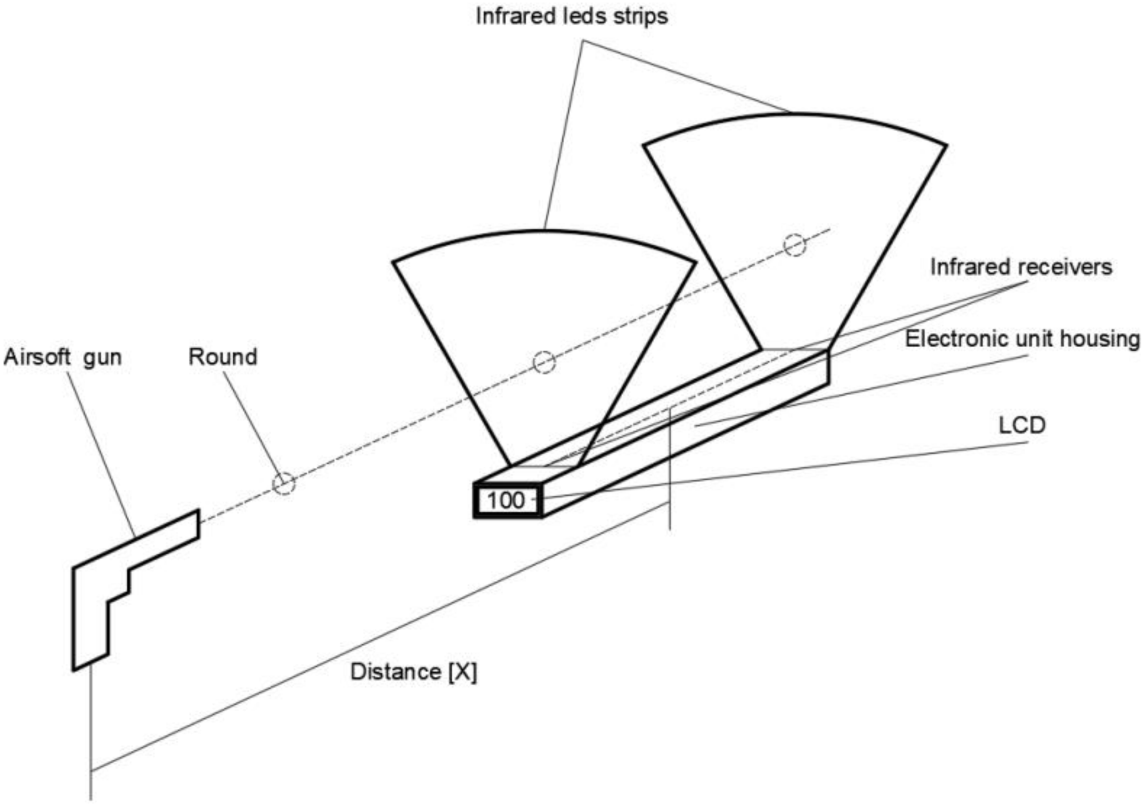


Figure 1: Infrared Radiation Based Projectile Speed Measurement System Scheme

Infrared radiation-based measurement units can be used to measure the speed of small arms rounds, as demonstrated in Figure 1. The setup involves two led strips and infrared receivers, which the round passes through, causing a decrease in the amount of light received. By measuring the time gap between the light decrement of the two strips, the average speed of the round can be calculated. However, the sensors used in this setup are susceptible to measurement errors due to their sensitivity to light conditions. Using a similar setup as Figure 1, Table 1 shows the measured speeds of an airsoft pistol's rounds at various distances from the pistol.

Meter/ Shot	1 meter	6 meters	11 meters	16 meters
1	109.5	90	76.8	65.2
2	110	88.5	73.6	64
3	109.3	88	74.6	63.6
4	112.4	88.5	76.2	
5	108	87.5	76.1	
6	112.7	88.5	81	
7	107.4	87.9	75.9	
8	106.9	89.6	74.9	
9	105.2	91	76.4	
10	106.9	88.9	78.9	
Average	108.83	88.84	76.44	64.26667
	V(x) m/s			

Table 1: Measured Round Longitudinal Speeds at Different Distances.

Airsoft guns typically use plastic spherical rounds that are subject to both aerodynamic and gravitational forces during flight. To ensure a safe and effective firing mechanism, it is important to accurately predict the drag coefficient (cd) of these rounds. Once this is done, the necessary parameters for a gas or electric mechanism can be determined to provide good range while maintaining a safe velocity that minimizes the risk of severe injury to the target. Figure 2 provides a kinematic and dynamic free body diagram (FBD) of the airsoft round.

Parameter	Definition	Value	Unit
m_{round}	Mass of the round.	$0.28*10^{-3}$;	[g]
d	Diameter of the round.	6	[mm]
T	Temperature of the test room.	16	[°C]
ϕ	Relative humidity of the test room air.	58	[%]
p	Absolute pressure of the test room.	1024	[mbar]
X_0	Initial position of the round. (wrt. Global frame)	0	[m]
Y_0	Initial position of the round. (wrt. Global frame)	1.5	[m]

Table 2: Parameters and initial condition for the solution

Answers

a.) Curve fitting

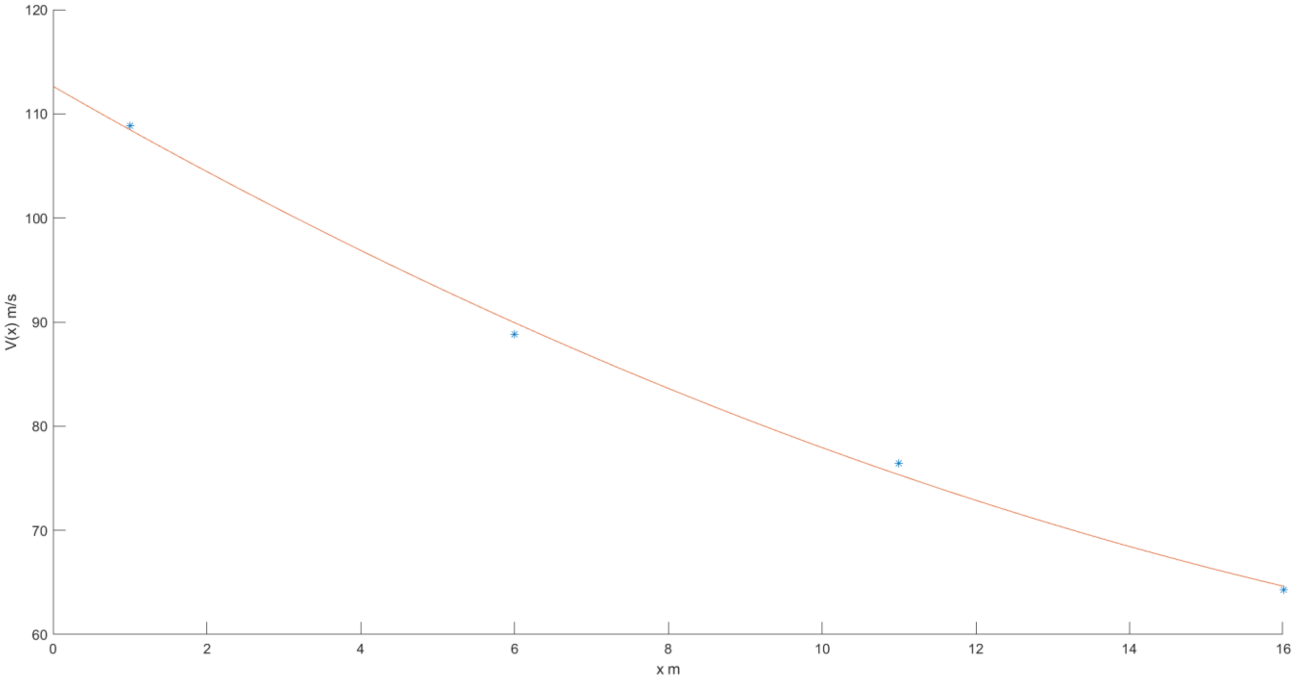
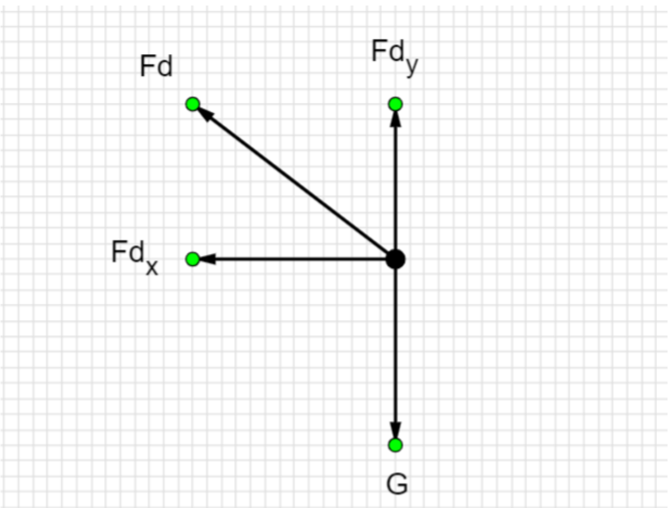


Chart 1: Curve fitting for experimental data

$$V(x) = 0.078167x^2 - 4.2506x + 112.6343$$

The value for the initial velocity obtained from this equation: $V(0) = 112.6343$

b.) Classifying the equations of planar motion of an airsoft round using FBDs



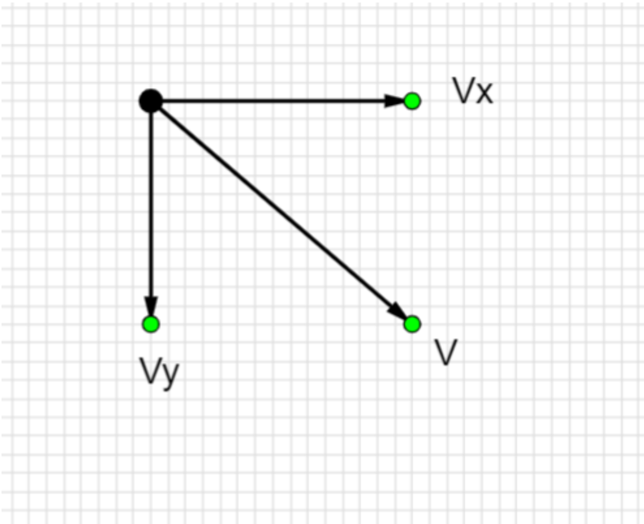
Fd : Air Drag Force

Fd_y : Vertical Air Drag Force

Fd_x : Horizontal Air Drag Force

G : Weight

Figure 2: FBD for forces



V : Vertical Velocity

Vx : Horizontal Velocity

Vy : Vertical Velocity

Figure 3: FBD for velocities

a. Calculating Air Density:

$p = \frac{P-e}{R*T}$; here, e is $e = \phi * \text{Psaturated}$.

The values of $P_{saturated}$ and R were obtained through reference to thermodynamic tables, specifically Table A-4 and Table A-2

TABLO A–2

Bilinen bazı gazların mükemmel-gaz özgül ısıları

(*a*) 300 K sıcaklıkta

Gaz	Kimyasal formülü	Gaz sabiti, *R* kJ/kg · K	c_p kJ/kg · K	c_v kJ/kg · K	*k*
Hava	—	0.2870	1.005	0.718	1.400
Argon	Ar	0.2081	0.5203	0.3122	1.667
Bütan	C_4H_{10}	0.1433	1.7164	1.5734	1.091
Karbon dioksit	CO_2	0.1889	0.846	0.657	1.289
Karbon monoksit	CO	0.2968	1.040	0.744	1.400
Etan	C_2H_6	0.2765	1.7662	1.4897	1.186
Etilen	C_2H_4	0.2964	1.5482	1.2518	1.237
Helyum	He	2.0769	5.1926	3.1156	1.667
Hidrojen	H_2	4.1240	14.307	10.183	1.405
Metan	CH_4	0.5182	2.2537	1.7354	1.299
Neon	Ne	0.4119	1.0299	0.6179	1.667
Azot	N_2	0.2968	1.039	0.743	1.400
Oktan	C_8H_{18}	0.0729	1.7113	1.6385	1.044
Oksijen	O_2	0.2598	0.918	0.658	1.395
Propan	C_3H_8	0.1885	1.6794	1.4909	1.126
Su buharı	H_2O	0.4615	1.8723	1.4108	1.327

Table 3: Thermodynamic Table A-2

TABLO A–4

Doymuş su — Sıcaklık tablosu

		Özgül hacim, m^3/kg		İç enerji, kJ/kg			Entalpi, kJ/kg			Entropi, kJ/(kg.K)		
Sıcaklık., T °C	Doymuş basıncı., P_{doy} kPa	Doymuş sıvı, v_f	Doymuş buhar, v_g	Doymuş sıvı, u_f	Buhar., u_{fg}	Doymuş buhar, u_g	Doymuş sıvı, h_f	Buhar., h_{fg}	Doymuş buhar, h_g	Doymuş sıvı, s_f	Buhar., s_{fg}	Doymuş buhar, s_g
0.01	0.6117	0.001000	206.00	0.000	2374.9	2374.9	0.001	2500.9	2500.9	0.0000	9.1556	9.1556
5	0.8725	0.001000	147.03	21.019	2360.8	2381.8	21.020	2489.1	2510.1	0.0763	8.9487	9.0249
10	1.2281	0.001000	106.32	42.020	2346.6	2388.7	42.022	2477.2	2519.2	0.1511	8.7488	8.8999
15	1.7057	0.001001	77.885	62.980	2332.5	2395.5	62.982	2465.4	2528.3	0.2245	8.5559	8.7803
20	2.3392	0.001002	57.762	83.913	2318.4	2402.3	83.915	2453.5	2537.4	0.2965	8.3696	8.6661
25	3.1698	0.001003	43.340	104.83	2304.3	2409.1	104.83	2441.7	2546.5	0.3672	8.1895	8.5567
30	4.2469	0.001004	32.879	125.73	2290.2	2415.9	125.74	2429.8	2555.6	0.4368	8.0152	8.4520
35	5.6291	0.001006	25.205	146.63	2276.0	2422.7	146.64	2417.9	2564.6	0.5051	7.8466	8.3517
40	7.3851	0.001008	19.515	167.53	2261.9	2429.4	167.53	2406.0	2573.5	0.5724	7.6832	8.2556
45	9.5953	0.001010	15.251	188.43	2247.7	2436.1	188.44	2394.0	2582.4	0.6386	7.5247	8.1633

Table 4: Thermodynamic Table A-4

$P_{saturated}$ is 1832.4 Pa

b. Equations:

Forces:

$Fd_x = ½\ p\ C_d A_r V_x^2$

$Fd_y = ½\ p\ C_d A_r V_y^2$

$G = mg$

According to the Newton's Law: $\Sigma F = m\alpha$, so $\Sigma Fx = m\ddot{x}$ ($\ddot{x}= \alpha_x$) and $\Sigma Fy = m\ddot{y}$ ($\ddot{y} = a_y$)

Horizontal Motion:

$m\ddot{x} = -½\ p\ C_d A_r V_x^2 = -½\ p\ C_d A_r \dot{x}^2$

$m\ddot{x} + ½\ p\ C_d A_r \dot{x}^2 = 0$ #nonlienar second order homogenous equation

Vertical Motion:

$m\ddot{y} = ½\ p\ C_d A_r V_y^2 – mg = ½\ p\ C_d A_r \dot{y}^2 - mg$

$m\ddot{y} + mg -½\ p\ C_d A_r \dot{y}^2 = 0$ #nonlienar second order nonhomogeneous equation

c.) Euler Forward Formula

$t = t + dt$

$V_y = V_y + [(½\ p\ C_d A_r V_y^2 – mg)/m] * dt$

$V_x = V_x + [(½\ p\ C_d A_r V_x^2 – mg)/m] * dt$

$y = y + V_y *dt$

$x = x + V_x*dt$

C_d was determined by trial and error and its value is 0.545

d.) Plotting

The first graph displays both the longitudinal velocity results from the model and the experimental data plotted against distance on a single graph. The second graph shows the positions of the airsoft round.

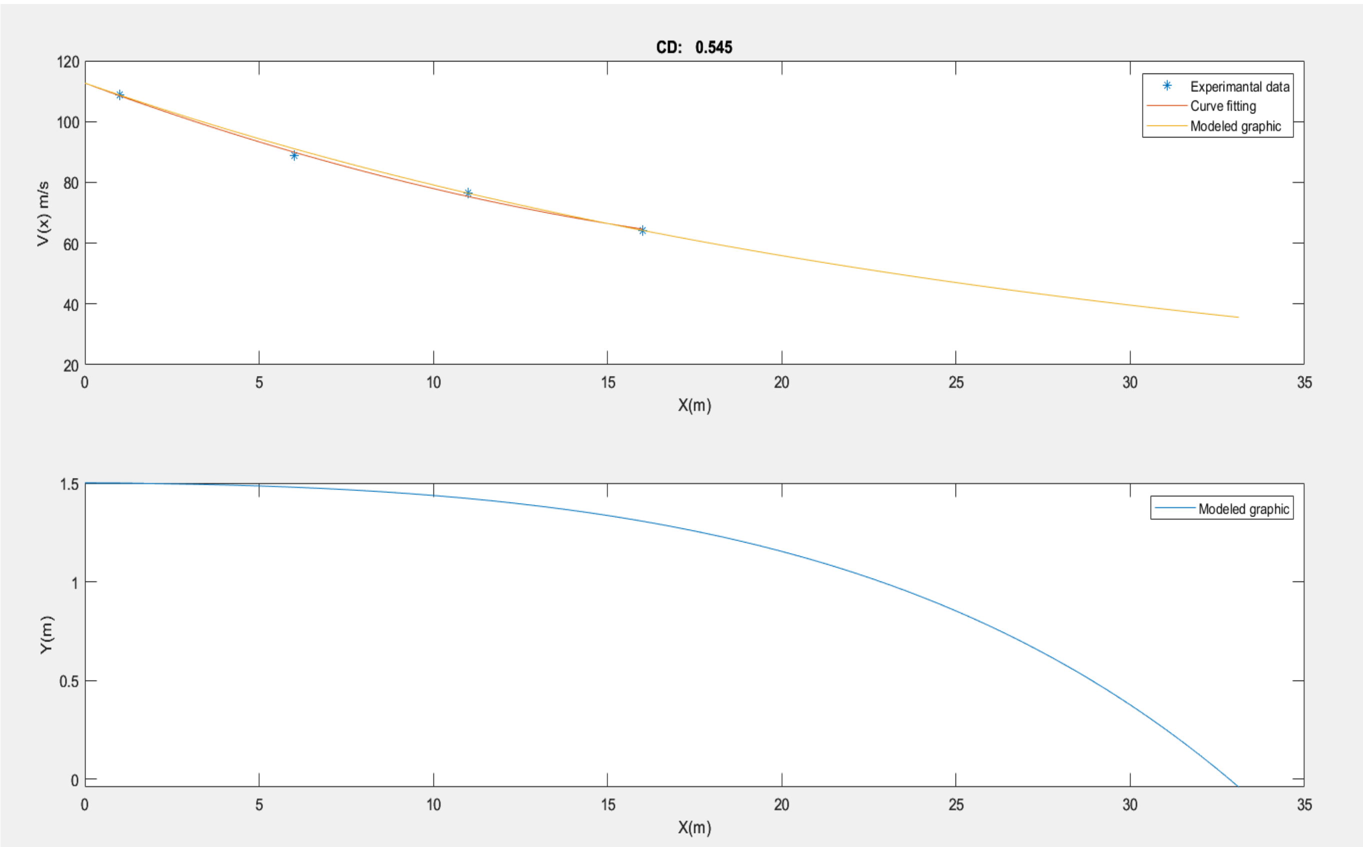


Chart 2: Modeled Velocity and Position Relationships - displaying two graphs, one showing velocity-x direction versus position-x, and the other showing position-x versus position-y

e.) State the range of the pistol and the final velocity of the round. Calculate the energy of the round when it hits to a target at a distance of 20 meters

a. According to the information taken from graphics, $V_{y,\ final}$ is 5.227 m/s and $V_{x,\ final}$ is 35.604 m/s; thus, the final velocity can be calculated by using this equation: $Vfinal = \sqrt{((Vy,final)^2 + (Vx,final)^2)}$. So, the final velocity value is 35.986 m/s.

b. The following data points relate to the round hitting a target 20 meters away: V_x = 55.89 m/s, V_y = 2.53 m/s, and Y(x) = 1.157 m. Thus, by using kinetic equation

$$\Delta E = \Delta Ek + \Delta Ep + \Delta Ee;$$

$$\Delta Ek = \left(\frac{1}{2}\right) * 0.28 * 10^{-3} * (55.89^2 + 2.53^2);$$

$$\Delta Ep = 0.28 * 10^{-3} * 9.81 * 1.157;$$

$$\Delta E = 0.4382 + 0.003178 = 0.4414 \text{ Joule}$$

f.) Elaborating results

According to the obtained data, the CD coefficient was determined as 0.545. In addition, the energy value calculated at the point where the shot reached the target was measured as 0.4414 Joules. These results provide important information about the aerodynamic properties of the shot and its impact on the target. A lower CD coefficient allows the shot to encounter less air resistance and can therefore help achieve a higher accuracy rate. Also, the high energy value means that the shot has a significant impact on the target and covers a larger area. These results are an important source of information for evaluating and improving shooting performance.

g.) The code of the program

The code of the program is written in the MATLAB programming language. The code is designed to calculate and analyze different parameters of the shot. The program calculates the final velocity and range of the shot using inputs such as target distance, projectile weight, projectile initial velocity, and weather conditions. The program can also plot the ballistic curves of the shot and visualize the results for different target distances. The code is considered a useful tool for analyzing and improving the performance of the shot.

```
clear all, close all, clc
%-------------------------------------------------
% Parameters:

m_round = 0.28*10^(-3); %kg
d = 0.006;              %m
T = 16 + 273.15;        %K
fi = 58/100;
P = 1024*100;           %pa
X_0 = 0;
Y_0 = 1.5;
dt = 0.01;
Ar = (pi*d^2)/4;        %m^2
Cd = 0.545;             % determined by trial and error
g = 9.81;               %m/s^2
%-------------------------------------------------
% Calculating air density:

%thermodynamic table-a4 for T = 16, P_saturated = 1832.4 Pa
e = fi*1832.4;
% to calculate air_density
air_density = (P-e)/(287*T); %here 287 is R, table-a1

%-------------------------------------------------
% a) Curve fitting:

x0 = 1; x1 = 6; x2 = 11; x3 = 16;
f_x0 = 108.83; f_x1 = 88.84; f_x2 = 76.44; f_x3 = 64.26667;

xs = [x0, x1, x2, x3];
f_x = [f_x0, f_x1, f_x2, f_x3];

p_0 = polyfit(xs, f_x, 2);

x_values = linspace(0,16,100);
y_values = polyval(p_0,x_values);


equation = p_0(1)+"x^2\t" + p_0(2)+"x\t" + p_0(3); %The equation presented
corresponds to the experimental data.
fprintf(equation)

```

```
%-------------------------------------------------
% c) Euler Forward Formula

%Initialize variables
v_x = 112.6343;
v_y = 0;
t = 0;
x = X_0;
y = Y_0;
times = t;
velocities_x = v_x;
velocities_y = v_y;
positions_x = x;
positions_y = y;

%Kinematic equations
while y > 0

    t = t + dt;

    v_y = v_y +(((0.5*air_density*Cd*Ar*v_y^2)-(m_round*g))/m_round)*dt;
    v_x = v_x + (-0.5*air_density*Cd*Ar*v_x^2/m_round)*dt;

    y = y + v_y*dt;
    x = x + v_x*dt;

    times = [times t];

    velocities_x = [velocities_x v_x];
    positions_x = [positions_x x];

    velocities_y = [velocities_y v_y];
    positions_y = [positions_y y];
end

%-------------------------------------------------
%Graphics:

subplot(2,1,1)
plot(xs,f_x,'*',x_values,y_values)
xlabel('X(m)')
ylabel('V(x) m/s')

hold on
plot(positions_x, velocities_x)
title("CD:    "+Cd)
legend('Experimental data','Curve fitting', 'Modeled graphic')

subplot(2,1,2)
plot(positions_x, positions_y)
xlabel('X(m)')
ylabel('Y(m)')
legend("Modeled graphic")
```