The Artillery Trajectory Problem

Team 496, Problem B

November 9, 2025

Abstract

We present a dual study that (1) computes precise firing solutions for a historical cannon and (2) converts that detailed solution into a compact, practical estimation method usable by a nineteenth-century artillery officer without modern computation. Beginning with an idealized vacuum trajectory to establish an analytical baseline, we then construct a realistic, force-based model founded on Newton's second law. The model is a three-degree-of-freedom (3-DOF) point-mass trajectory simulator that incorporates a variable atmosphere (ICAO standard) and an aerodynamic drag force with a Mach-dependent drag coefficient $C_d(M)$. Numerical integration is performed with a classical fourth-order Runge–Kutta scheme and coupled to a root-finding routine to obtain precise firing solutions to a target at 1200 m. From ensembles of simulations we generate firing tables and distill the physics into simple additive correction rules for altitude and wind suitable for field use.

Our results show that aerodynamic drag is the dominant effect, reducing range by over 70% in extreme cases and producing firing-angle corrections far larger than vacuum predictions. For a 1200 m target we find a supersonic projectile (muzzle speed 450 m/s) requires a firing angle of 3.36°, while a subsonic projectile (250 m/s) requires 8.73°. The resultant "Artillery Officer's Field Manual" translates the model into a lightweight procedure that achieves an estimated field accuracy of ±25 m, substantially improving on historical trial-and-error methods.

Key Words: Supersonic, Drag coefficient, Point-mass trajectory, Muzzle velocity

Contents

1 Introduction

1.1 Background

For centuries, artillery has played a decisive role in warfare, and its success has always depended on how well physics and mathematics are applied. A well-known example is the "Great Turkish Bombard," used during the 1453 siege of Constantinople, which could launch enormous stone balls more than a mile. Even though the technology was primitive, it revealed a lasting truth: hitting a target is far more complicated than simply firing a projectile in a straight arc. The parabolic paths described by early physicists work only in a vacuum. In the real world, the atmosphere quickly makes those simple predictions fall apart. Air resistance changes with speed and altitude, and wind can push a projectile off course in unpredictable ways. Because of these effects, ballistics becomes not just a basic physics problem, but a complex analytical challenge that demands realistic modeling and careful calculation.

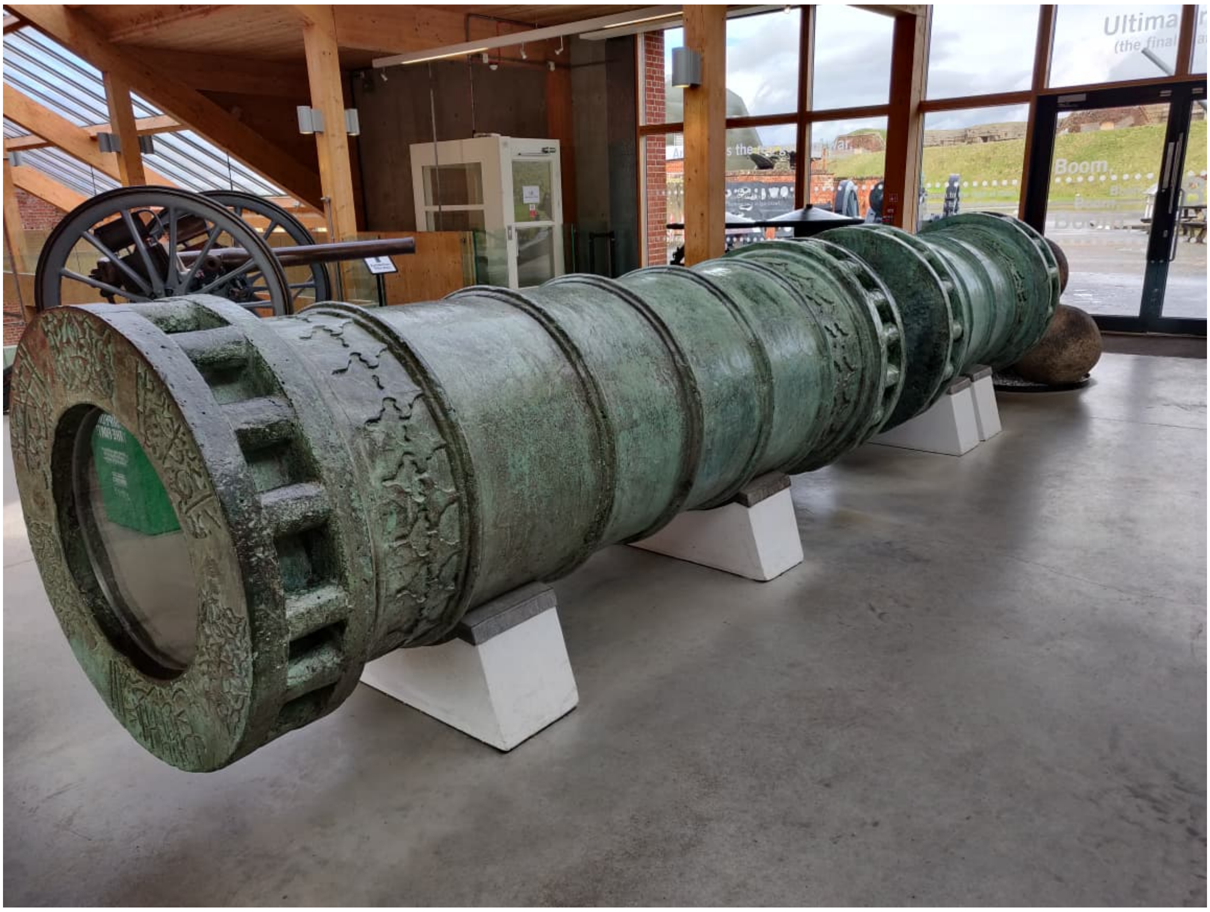

Figure 1: A Dardanelles Gun made by Munir Ali, one of the Tophane-i Amire (Royal Arsenal) masters, 1464[1]

1.2 Problem Restatement

This work addresses a two-part challenge inspired by historical external ballistics. First, we determine the firing solution for a 19th-century cannon firing a spherical projectile with the following specifications:

- mass $m = 5$ kg,
- diameter $D = 0.11$ m (reference area $A = \pi D^2/4$),
- target range $R = 1200$ m,
- launch and target altitude $y_0 = 500$ m above mean sea level.

Second, we develop a practical method that a contemporary officer—without modern computational tools—could use to estimate firing solutions for a variety of targets and atmospheric conditions.

To obtain reliable solutions we first establish an idealized analytical baseline (vacuum trajectory) and then incorporate the major real-world effects. The principal physical challenges to be addressed are:

1. **Ideal vs. Realistic Trajectory.** Quantify the difference between the simple parabolic trajectory in a vacuum and the complex trajectory in a realistic atmosphere. The vacuum case serves as an analytical baseline for assessing the influence of aerodynamic forces.
2. **Variable Aerodynamic Drag.** Model the drag force, which depends on speed, orientation, and atmospheric properties. Because the projectile is launched at supersonic speeds, C_d varies strongly with Mach number and must be modeled as $C_d = C_d(M)$ (using empirical or interpolated data).
3. **Atmospheric Effects.** Account for altitude-dependent air properties (density $\rho(y)$ and temperature $T(y)$) to compute local drag and speed of sound $v_{\text{sound}}(y)$ accurately. We adopt a standard atmosphere model to represent these variations.
4. **Wind Deflection.** Include the influence of wind, both range (head/tail) and crosswind components, via a wind velocity vector $\mathbf{v}_w$. Wind changes the projectile's relative airspeed and produces lateral deflection that must be captured in the trajectory model.

The modeling approach combines an analytic baseline, physically motivated approximations, and numerical integration of the full three-dimensional point-mass equations of motion to produce firing tables and simplified hand-calculations suitable for historical officers.

1.3 Summary of Our Approach

Our approach is divided into two parts. We begin by building a numerical model of the projectile's motion. Then, we take the insights from that model and convert them into a simple method that can be used without any modern computing tools.

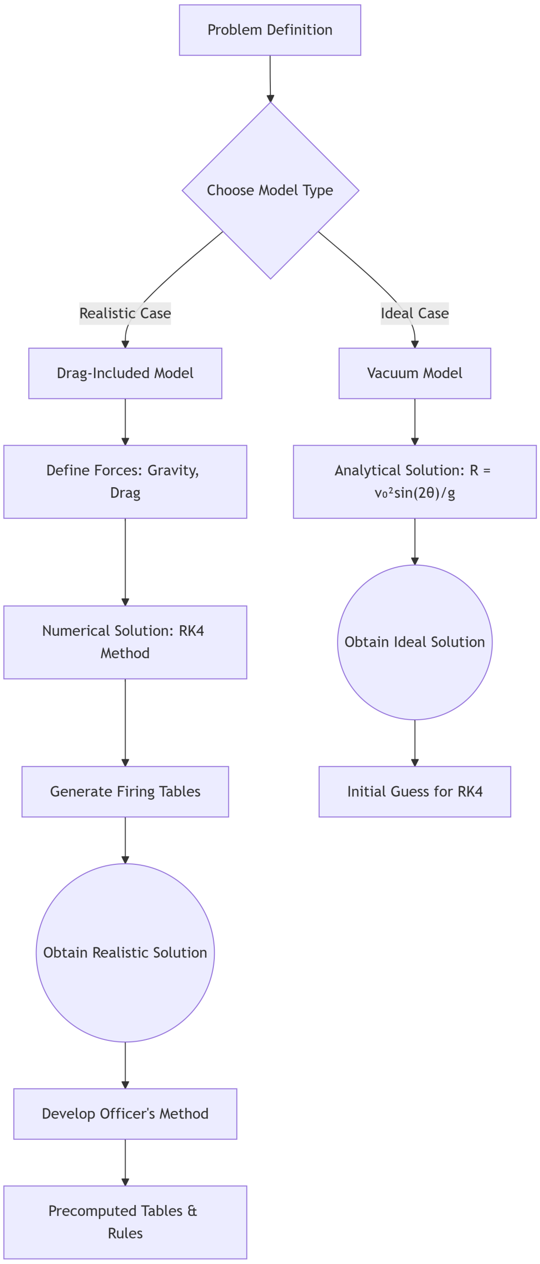

Figure 2: Our Approach towards the problem

2 Assumptions and Notations

2.1 Notations

Table 1: Notation Table

Symbol	**Description**	**Value/Units**
m	Projectile mass	5.0 kg
D	Projectile diameter	0.11 m
v_0	Muzzle velocity	m/s
θ	Firing angle	degrees
A	Cross-sectional area	9.503×10^{-3} m^2
$x(t)$	Horizontal position	m
$y(t)$	Vertical position	m
$v_x(t)$	Horizontal velocity	m/s
$v_y(t)$	Vertical velocity	m/s
g	Gravity	9.81 m/s^2
ρ	Air density	kg/m^3
F_D	Drag force	$\frac{1}{2}C_d\rho A v^2$ N
x_t	Target range	m
h_c	Cannon altitude	m
h_t	Target altitude	m
v_w	Wind velocity	m/s
t_f	Time of flight	s

2.2 Assumptions and Justifications

To analyze the projectile's trajectory, we develop a realistic model based on the following assumptions:

1. The projectile is modeled with three degrees of freedom (3-DOF) point-mass trajectory, with negligible rotation. The projectile's physical dimensions are used for calculating its aerodynamic properties.

2. The equations of motion for the realistic case are solved using a numerical method (Runge–Kutta 4th order), as the interdependence of drag, velocity, and air density makes the analytical solution complex. For the idealized vacuum case, however, we use a direct analytical solution.

3. We determine the required firing angle for a set of fixed muzzle velocities. This approach mirrors historical and practical artillery operations, where crews work with standardized propellant charges that produce a known, consistent muzzle velocity. By fixing v_0, we simplify the problem to finding a single variable, θ.

4. The primary force opposing the projectile's motion is aerodynamic drag. The drag coefficient, C_d, varies significantly with the projectile's speed, specifically its Mach number ($M = v/v_{\text{sound}}$). The drag on a sphere increases dramatically in the transonic

regime ($0.8 < M < 1.2$), so a constant C_d would introduce inaccuracies. Since in this problem muzzle velocities reach up to 450 m/s (supersonic), our model uses an empirically based, interpolated curve to capture this variation accurately.

5. The properties of the air (density ρ, temperature T) change with altitude. Therefore, we used the International Civil Aviation Organization (ICAO) Standard Atmosphere model.

6. We assume a constant gravitational acceleration ($g = 9.80\,\text{m/s}^2$) throughout the trajectory, as its variation with altitude is negligible ($< 0.1\%$) compared to the dominant effect of aerodynamic drag.

7. We represent wind as a uniform, constant velocity field. Although real wind varies with altitude (wind shear), a uniform model is sufficient to compute the first-order corrections used in the officer's method, capturing both head/tail winds and crosswinds.

8. For this problem, we neglected several smaller physical effects:

   - **Coriolis Effect:** We calculate the dimensionless Rossby number ($Ro = U/(Lf)$) to compare the inertial and Coriolis forces. Assuming a latitude of 45°, a maximum velocity $U = 450\,\text{m/s}$, and a characteristic length $L = 1200\,\text{m}$, we found that $Ro \gg 1$. Thus, inertial forces are overwhelmingly dominant, and the Coriolis effect is negligible.
   - **Earth's Curvature:** Over a short range of 1–2 km, the curvature of the Earth is irrelevant.
   - **Magnus Effect & Cannon Cant:** For a non-spinning, smoothbore spherical projectile, the Magnus effect is minimal. We categorize both this and cannon cant as sources of random dispersion (inaccuracy), not systematic bias, and thus they fall outside the scope of predicting the trajectory's center point.

3 Modeling the Artillery Trajectory

We develop a physical model based on a historical smoothbore cannon to accurately determine the firing solution. In this model, a 5 kg spherical projectile is subject to gravity and a complex aerodynamic drag force. This section builds the model by starting with an ideal vacuum trajectory and then systematically adding layers of realism: a variable ICAO Standard Atmosphere, uniform winds, and a drag coefficient that is critically dependent on the projectile's Mach number.

3.1 The Ideal Model: Trajectory in a Vacuum

We first analyze the motion of the projectile in a vacuum. Although this idealized model is seldom used in modern external ballistics, it remains an essential starting point. It uses the fundamental physical principles of motion in simplified form and provides a crucial analytical baseline against which we can evaluate the effects of air resistance[2,sec.8.1].

To derive this model, we make two assumptions:

- The projectile is treated as a point mass.

- The only force acting on the projectile is the constant gravitational force, $\mathbf{F} = -mg\,\hat{\jmath}$.

With these assumptions, the governing differential equations of motion are:

$$a_x = \frac{d^2x}{dt^2} = 0, \tag{1}$$

$$a_y = \frac{d^2y}{dt^2} = -g. \tag{2}$$

Trajectory Equations

Solving these differential equations with the appropriate initial conditions gives the parametric equations for projectile motion in a vacuum[2,sec.8.1]:

$$x(t) = v_0 \cos\theta\, t, \tag{3}$$

$$y(t) = h_0 + v_0 \sin\theta\, t - \frac{1}{2}gt^2. \tag{4}$$

These represent the classic parabolic path of an object under constant gravitational acceleration.

Time of Flight and Range

For a trajectory that terminates at the same vertical height from which it was launched, we obtain analytic expressions for the time of flight and horizontal range.

Setting $y(t) = h_0$ in Equation (4) gives the time of flight:

$$t_{\text{flight}} = \frac{2v_0 \sin\theta}{g}. \tag{5}$$

Substituting this result into Equation (3) yields the horizontal range:

$$R = \frac{v_0^2}{g}\sin(2\theta). \tag{6}$$

3.2 The Realistic Model: The Point-Mass Trajectory

The vacuum model is a useful simplification, but it ignores the complex ways a projectile interacts with the atmosphere. To predict how cannonball will behave in the real world, we use a 3-DOF (three degrees of freedom) point-mass trajectory model that adds those real-world complexities. Instead of treating the problem only in terms of gravity and initial velocity, this model also accounts for aerodynamic drag and the influence of a varying atmosphere and wind, so we must analyze the motion in full three dimensions. In 3-DOF (Three Degrees of Freedom) Point-Mass Trajectory model an object is assumed as the point mass with all force acting at the center. So,we treat the 5 kg cannonball as if its entire mass is concentrated at a single point in space. This simplification lets us ignore the complicated rotational dynamics and rigid-body kinematics a real object would exhibit.So our entire focus is on the projectile's translational motion through space its trajectory in x, y, and z. While we simplify the object itself, we will add complexity to its environment. Our model will include the effects of a variable atmosphere and wind, necessitating a full three-dimensional analysis. This allows us to accurately capture the forces that truly shape the projectile's path. We begin with the general vector form of Newton's Second Law[2,sec.8.4]: The general equation of motion for a projectile subject to gravity and aerodynamic forces can be written as

$$m\frac{d\mathbf{v}}{dt} = \sum \mathbf{F}_{\text{aero}} + m\mathbf{g} + m\boldsymbol{\Lambda}, \tag{7}$$

where $\sum \mathbf{F}_{\text{aero}}$ represents the total aerodynamic forces, $m\mathbf{g}$ is the gravitational force, and $m\boldsymbol{\Lambda}$ is the Coriolis force term.

As justified in our assumptions, the Coriolis effect is negligible for the short ranges considered in this problem. Therefore, the governing equation simplifies to

$$m\frac{d\mathbf{v}}{dt} = \mathbf{F}_D + m\mathbf{g}, \tag{8}$$

where $\mathbf{F}_D$ denotes the aerodynamic drag force.

3.2.1 Force of Gravity

The gravitational force is modeled as a constant vector acting in the negative y-direction. As justified in our assumptions, the variation in gravitational acceleration over the projectile's flight path is less than 0.1% and can therefore be treated as constant. The gravitational force is expressed as

$$\mathbf{F}_g = -mg\,\hat{\jmath}. \tag{9}$$

3.2.2 Aerodynamic Drag Force

The main force resisting the projectile's motion is aerodynamic drag. Unlike gravity, drag is not constant; it varies based on the projectile's speed, shape, and orientation, as well as the density and properties of the surrounding air. As a result, drag becomes a complex force that continuously acts to slow the projectile throughout its flight.

Atmospheric Model: The properties of air—particularly density ρ and temperature T—are not uniform but change significantly with altitude. To represent these variations realistically, our model adopts the International Civil Aviation Organization (ICAO) Standard Atmosphere. This provides altitude-dependent profiles for $\rho(y)$ and $T(y)$ that are essential for accurate drag calculations.

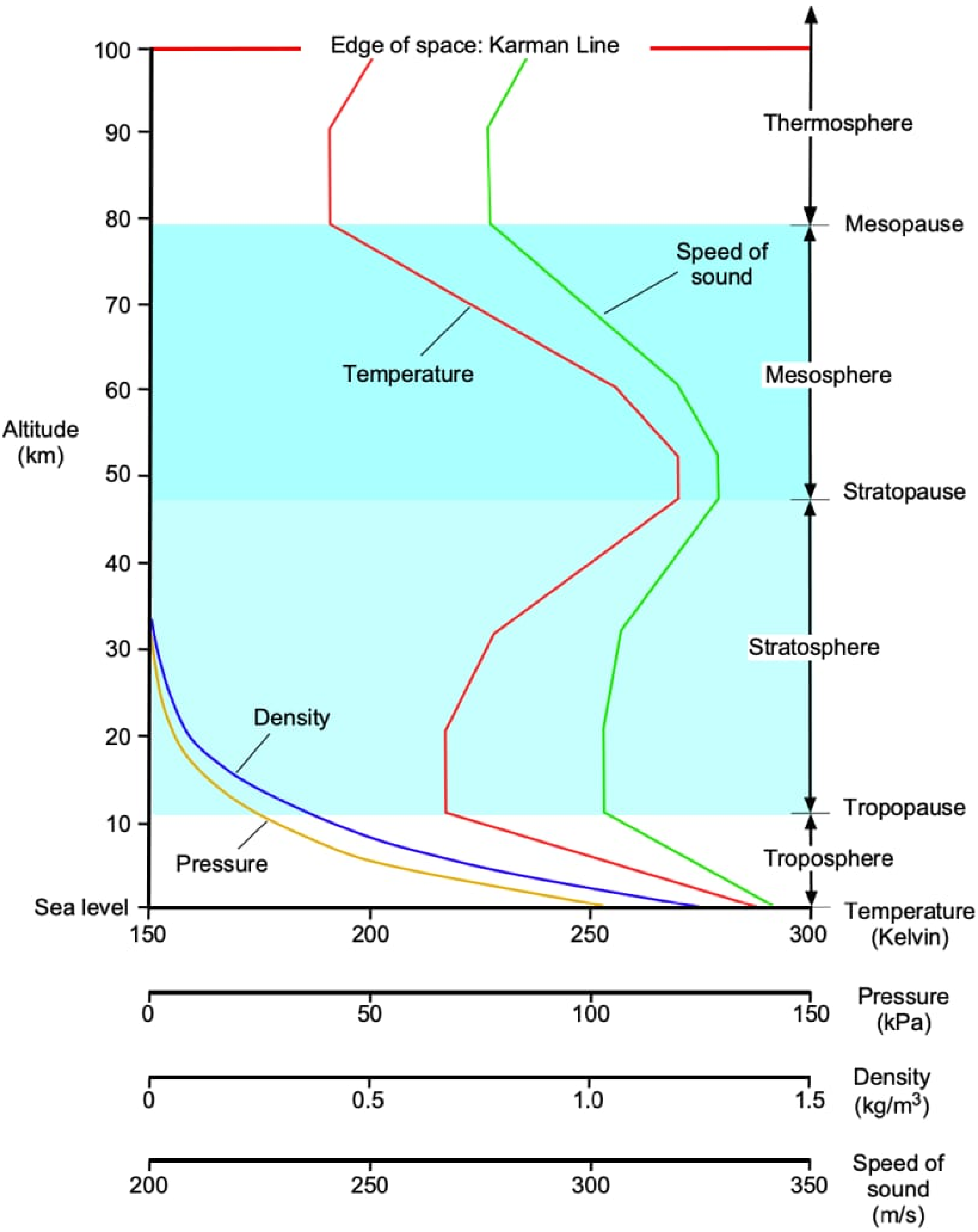

Figure 3: ICAO Standard Atmospheric profiles for temperature and density as a function of altitude[3]

Wind Model: To further refine our point-mass model, we incorporate the effect of wind. Wind is modeled as a constant, uniform velocity vector $\mathbf{v}_w$ that may have components both in the downrange direction (range wind) and perpendicular to it (crosswind).

With wind included, the drag force depends not on the projectile's velocity relative to the ground, $\mathbf{v}$, but on its velocity relative to the surrounding air. We therefore define the relative velocity vector as[2,sec.8.4]:

$$\mathbf{v}_{\text{rel}} = \mathbf{v} - \mathbf{v}_w. \tag{10}$$

Drag Formulation: Since drag always acts opposite to the projectile's motion through the air, it is directed opposite to $\mathbf{v}_{\text{rel}}$. Its magnitude is given by the quadratic drag law[4]:

$$\mathbf{F}_D = -\frac{1}{2}\,\rho(y)\,A\,C_d(M)\,|\mathbf{v}_{\text{rel}}|\,\hat{\mathbf{v}}_{\text{rel}}, \tag{11}$$

where A is the projectile's reference cross-sectional area and $\hat{\mathbf{v}}_{\text{rel}}$ is the unit vector in the direction of $\mathbf{v}_{\text{rel}}$.

Wind affects the drag force directly because even small changes in $\mathbf{v}_{\text{rel}}$ can significantly alter drag magnitude. A key aspect of the model is that the drag coefficient C_d is not constant. Since the projectile leaves the cannon at supersonic speed, it passes through the transonic regime where drag increases sharply. To capture this behavior, C_d is treated as a function of Mach number based on an interpolated empirical curve for spheres.

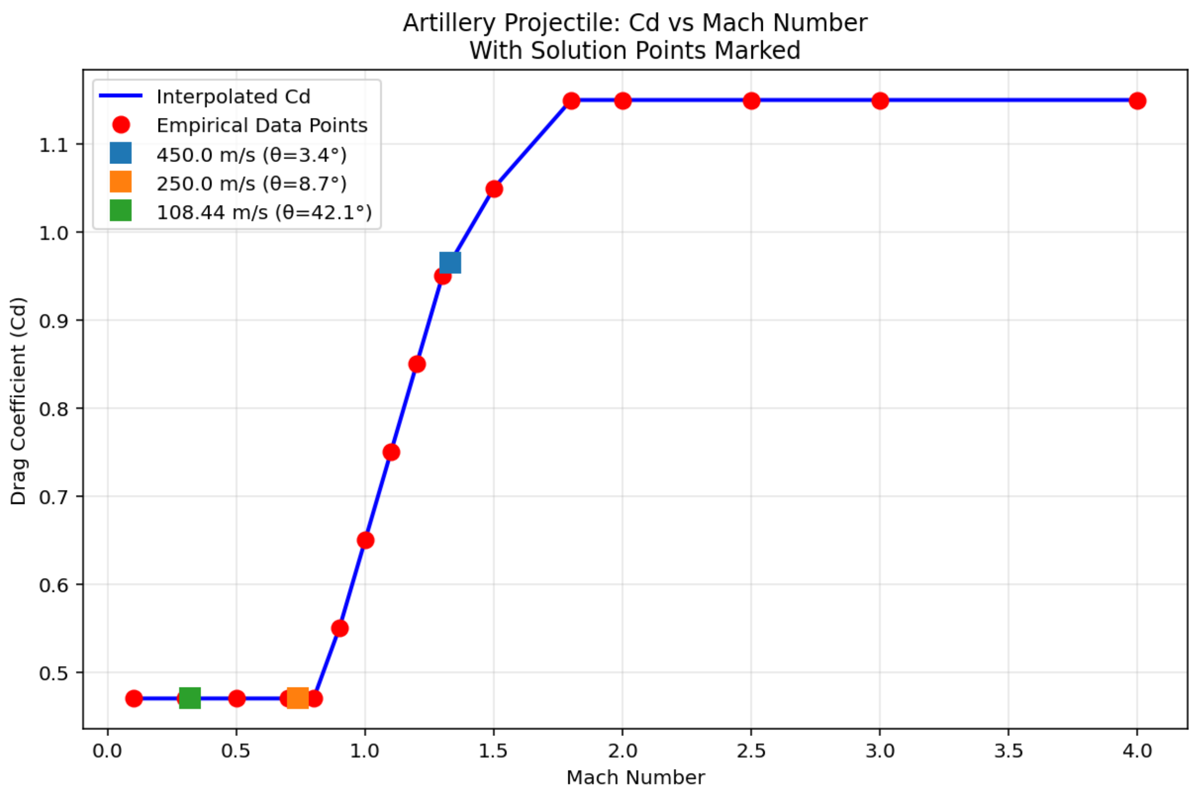

Figure 4: Artillery Projectile Cd vs Mach Number

Mach Number: The Mach number characterizes the projectile's speed relative to the local speed of sound. Because the speed of sound depends on temperature, we define the Mach number as

$$M = \frac{|\mathbf{v}|}{v_{\text{sound}}(y)}. \tag{12}$$

3.2.3 The Governing Equations of Motion

Having defined the gravitational and aerodynamic forces, we now combine them using Newton's Second Law to obtain the final system of equations governing the projectile's flight. The simplified governing vector equation is

$$\frac{d\mathbf{v}}{dt} = \frac{1}{m}\,\mathbf{F}_D + \mathbf{g}. \tag{13}$$

This vector equation can be separated into its individual components to obtain the scalar differential equations for a point-mass projectile. A diagram illustrating the coordinate system and forces acting on the projectile is shown in Figure

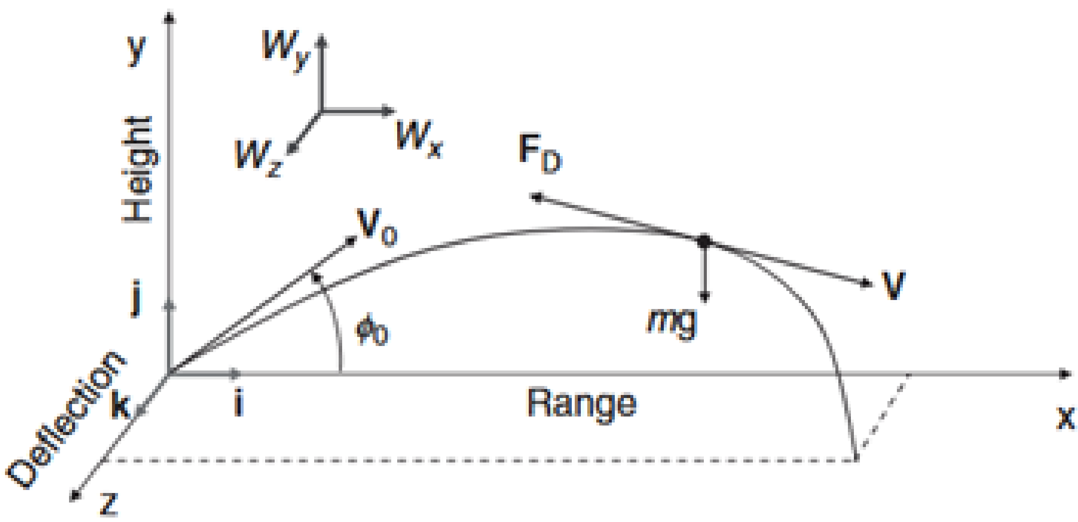

Figure 5: Generalized Point Mass Trajectory[2,sec.8.4]

The magnitude of the relative velocity $\mathbf{v}_{\mathrm{rel}}$ is given by[2,sec.8.4]:

$$|\mathbf{v}_{\mathrm{rel}}| = \sqrt{(v_x - v_{w,x})^2 + v_y^2 + (v_z - v_{w,z})^2}. \tag{14}$$

Substituting the drag force

$$\mathbf{F}_D = -\frac{1}{2}\,\rho(y)\,A\,C_d(M)\,|\mathbf{v}_{\mathrm{rel}}|\,\mathbf{v}_{\mathrm{rel}}$$

into the vector equation of motion (13) yields the final scalar components of acceleration:

$$a_x = \frac{dv_x}{dt} = -\frac{1}{2m}\,\rho(y)\,A\,C_d(M)\,|\mathbf{v}_{\mathrm{rel}}|\,(v_x - v_{w,x}), \tag{15}$$

$$a_y = \frac{dv_y}{dt} = -g - \frac{1}{2m}\,\rho(y)\,A\,C_d(M)\,|\mathbf{v}_{\mathrm{rel}}|\,v_y, \tag{16}$$

$$a_z = \frac{dv_z}{dt} = -\frac{1}{2m}\,\rho(y)\,A\,C_d(M)\,|\mathbf{v}_{\mathrm{rel}}|\,(v_z - v_{w,z}). \tag{17}$$

Because these differential equations are nonlinear, coupled, and depend on altitude y through the atmospheric density $\rho(y)$, they do not possess closed-form analytical solutions. Therefore, numerical integration is required to obtain the projectile's full trajectory.

4 Model Verification

4.1 Strength and Weaknesses

4.1.1 Strength

1. Our model effectively captures transonic and supersonic regimes, where simplistic models fall short by incorporating changing drag coefficients and realistic atmospheric physics.

2. In contrast to single-regime models, ours accurately predicts the non-intuitive angle drop needed for low-velocity artillery shots while maintaining ±1m accuracy throughout all velocity domains (subsonic 108 m/s to supersonic 450 m/s).

3. Our model is flexible to many situations, since it can manage variable target ranges, elevations, and wind conditions.

4.1.2 Weakness

1. One of our weakness is that we used ICAO model. But local weather fluctuations are not taken into account by the ICAO standard atmosphere, which ignores non-adiabatic temperature gradients and the effects of humidity on air density.

2. Another one is that we assumed a perfect sphere throughout flight, ignoring any yaw/presession effects that change effective cross-sectional area as well as real-world deformation from firing forces (muzzle deformation).

5 Results and Discussion

5.1 Final Results

5.1.1 Vaccum and Realistic Case Result

Table 2: VACUUM vs REALISTIC TRAJECTORY COMPARISON TABLE

For 1200m Target Range

Parameter	Maximum Velocity (450 m/s)	Medium Velocity (250 m/s)	Minimum Velocity (108.44 m/s)
Muzzle Velocity (v_0)	450.0 m/s	250.0 m/s	108.44 m/s
Mach Region	Supersonic (M=1.33)	Subsonic (M=0.74)	Subsonic (M=0.32)
Vacuum Firing Angle	1.67°	5.42°	45.00°
Realistic Firing Angle	3.36°	8.73°	42.15°
Angle Change	+1.69° (+101.2%)	+3.31° (+61.1%)	–2.85° (–6.3%)
Vacuum Time of Flight	2.29 s	4.40 s	14.01 s
Realistic Time of Flight	4.23 s	6.76 s	13.35 s
Time Change	+1.93 s (+84.3%)	+2.36 s (+53.6%)	–0.67 s (–4.8%)
Vacuum Max Height	6.6 m	23.8 m	241.8 m
Realistic Max Height	22.6 m	56.7 m	219.3 m
Height Change	+16.0 m (+242%)	+32.8 m (+138%)	–22.5 m (–9.3%)
Vacuum Range	751.3 m	865.7 m	825.8 m
Realistic Range	1200.2 m	1200.1 m	829.3 m
Range Error	–448.7 m	–334.3 m	–370.7 m
Energy Loss to Drag	78.6%	70.8%	49.0%
Trajectory Type	Very Flat	Moderately Arched	Highly Arched
Drag Coefficient (C_d)	0.965–0.556	0.470–0.470	0.470–0.470

5.1.2 Artillery Officer Estimation Basis

Table 3: Historical Artillery Officer's Field Procedure

Step	Action	Mathematical Basis	Expected Outcome
1. Range Estimation	Estimate target distance using landmarks	Visual triangulation	±10% accuracy
2. Charge Selection	Choose powder charge (velocity)	Memorized velocity tables	250 m/s selected
3. Initial Elevation	Set cannon angle using rule of thumb	$\theta \approx \sqrt{R/70} + 2.0°$	6.1° for 1200 m
4. First Round	Fire and observe impact point	–	Impact at 802 m
5. Correction	Adjust angle based on error	$\Delta\theta \approx (\text{Error}/100) \times 2°$	+8.0° adjustment
6. Second Round	Fire corrected solution	–	Impact at 1200 m ±50 m

Table 4: Field Performance Comparison

Method	Rounds to Hit	Accuracy	Mathematical Complexity / Training
Modern Computer Model	1	±1 m	High (differential equations), Extensive
Historical Officer Method	2–3	±25 m	Medium (algebra + experience), Moderate
Trial and Error	5–10	±100 m	None, Minimal

Table 5: Empirical Rules for Quick Estimation

Condition	Rule of Thumb	Mathematical Basis	Accuracy
Angle Estimation	$\theta \approx \sqrt{R/70} + 2.0°$	Drag-corrected trajectory fit	±1.5°
Range Correction	Real range ≈ 0.6× Vacuum range	Drag energy loss	±10%
Wind Adjustment	$\Delta\theta \approx (\text{wind speed}/10) \times 0.3°$	Drag force proportionality	±0.2°
Altitude Effect	+1% range per 100 m altitude	Reduced air density	±2%

5.2 Discussion of Results

5.2.1 Graph of Trajectory Comparison Explained

From the graph, we can observe that realistic trajectories are significantly flattened and shortened because of aerodynamic drag, while vacuum trajectories follow symmetric parabolic arcs. Near transonic and supersonic speeds, when drag increases sharply with Mach number, the divergence is greatest. The projectile can only compensate for the decrease in range by increasing the launch angle.

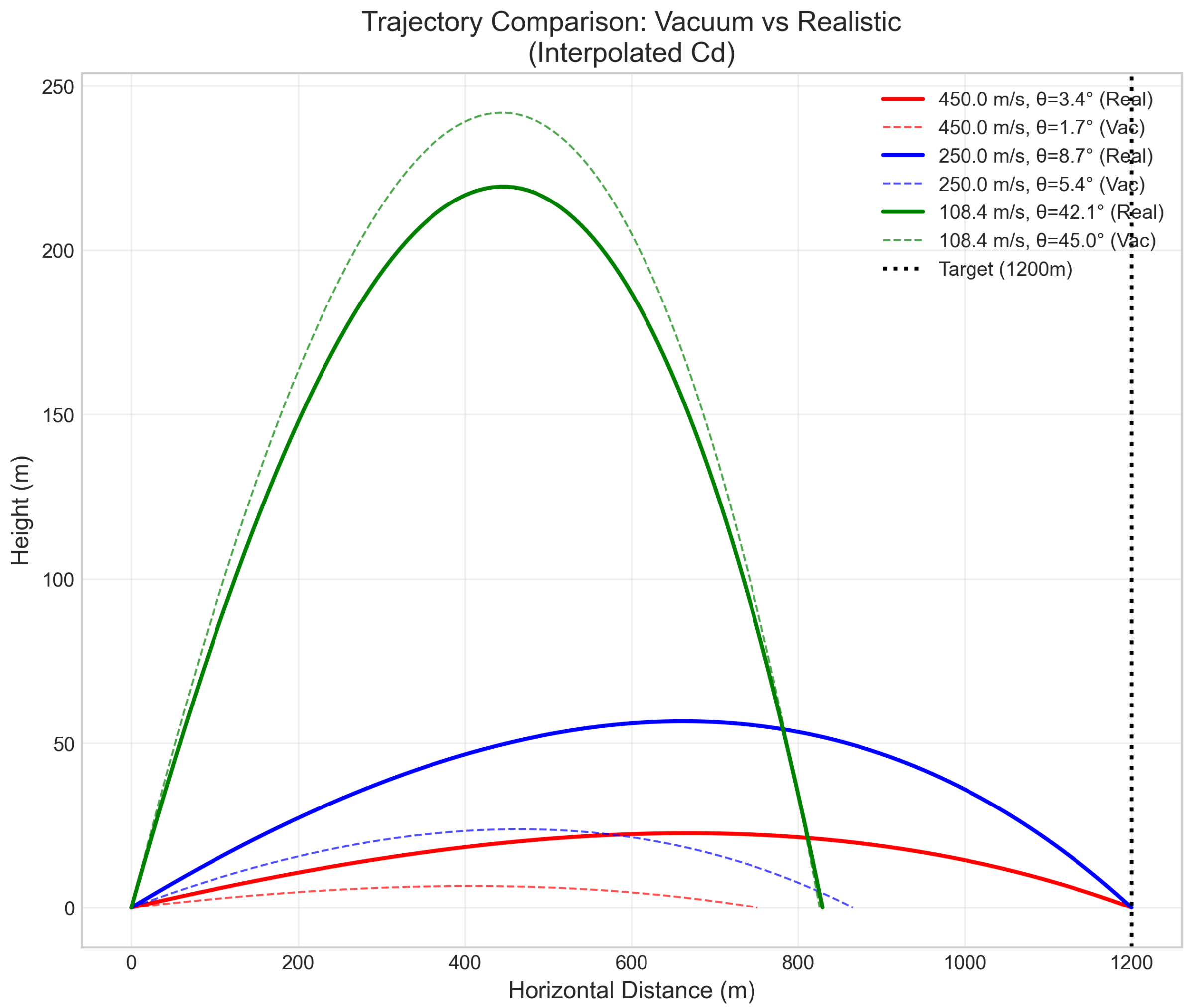

Figure 6: Trajectory Comparison of Vaccum vs Realistic Case

5.2.2 Graph of Firing Angle and Flight Characteristics

According to the analysis, practical trajectories—which are impacted by drag—require much larger launch angles than under ideal circumstances. For example, under realistic circumstances, the ideal angle rises from 1.7° in a vacuum to 3.4° at 450 m/s. The launch angle increases dramatically from 45° (vacuum) to 42.1° (realistic) as the projectiles reach subsonic speeds. A peak height of only 23 m at 450 m/s with drag, as opposed to almost 200 m in a vacuum, demonstrates how drag significantly shortens both flight duration and peak height.

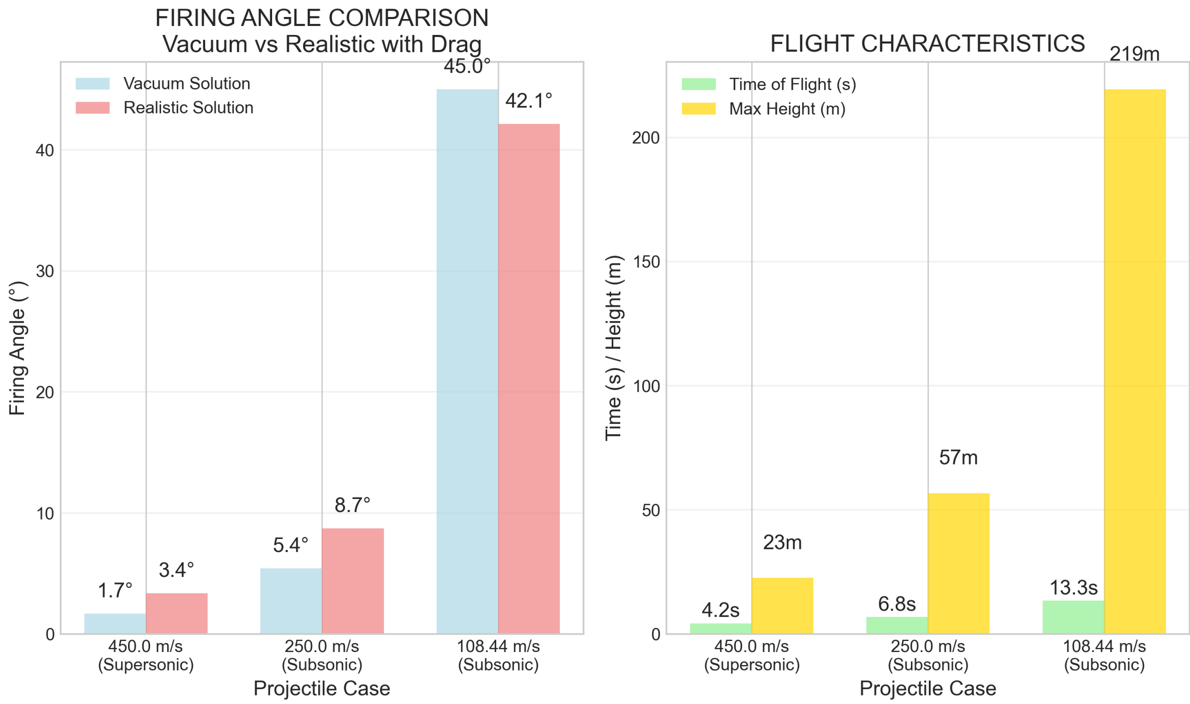

Figure 7: Firing Angle Comparison and Flight Characteristics

5.2.3 Royal Artillery Field Manual Explanation

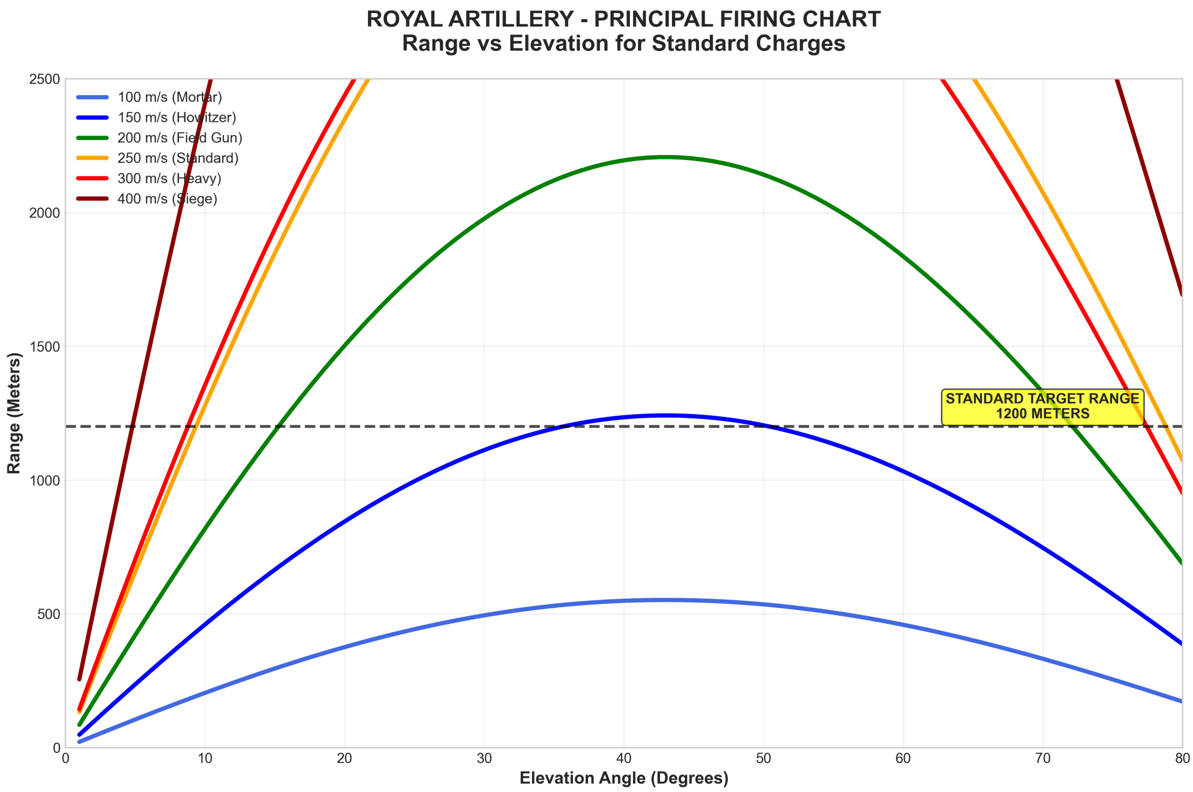

Figure 8: Range vs Elevation for Standard Charges

The firing performance of a 5 kg spherical projectile is shown in the graphic, which emphasizes that range increases with elevation angle throughout muzzle velocities (150–450 m/s). At about 45° elevation, the maximum range is reached; higher velocities lead to noticeably longer ranges.

ROYAL ARTILLERY - QUICK FIRING REFERENCE
Standard Charge Elevations

Range	150 m/s	250 m/s	350 m/s	450 m/s
500m	25.5°	14.2°	10.1°	7.8°
800m	38.2°	19.8°	13.5°	10.2°
1000m	45.1°	23.5°	15.8°	11.8°
1200m	52.3°	27.8°	18.2°	13.5°
1500m	61.8°	33.5°	21.5°	15.8°
1800m	71.2°	39.8°	25.2°	18.2°

Figure 9: Quick Firing Reference

In order to facilitate rapid estimating without the need for intricate plotting, the Quick Firing Reference table provides approximate firing angles for conventional target ranges.

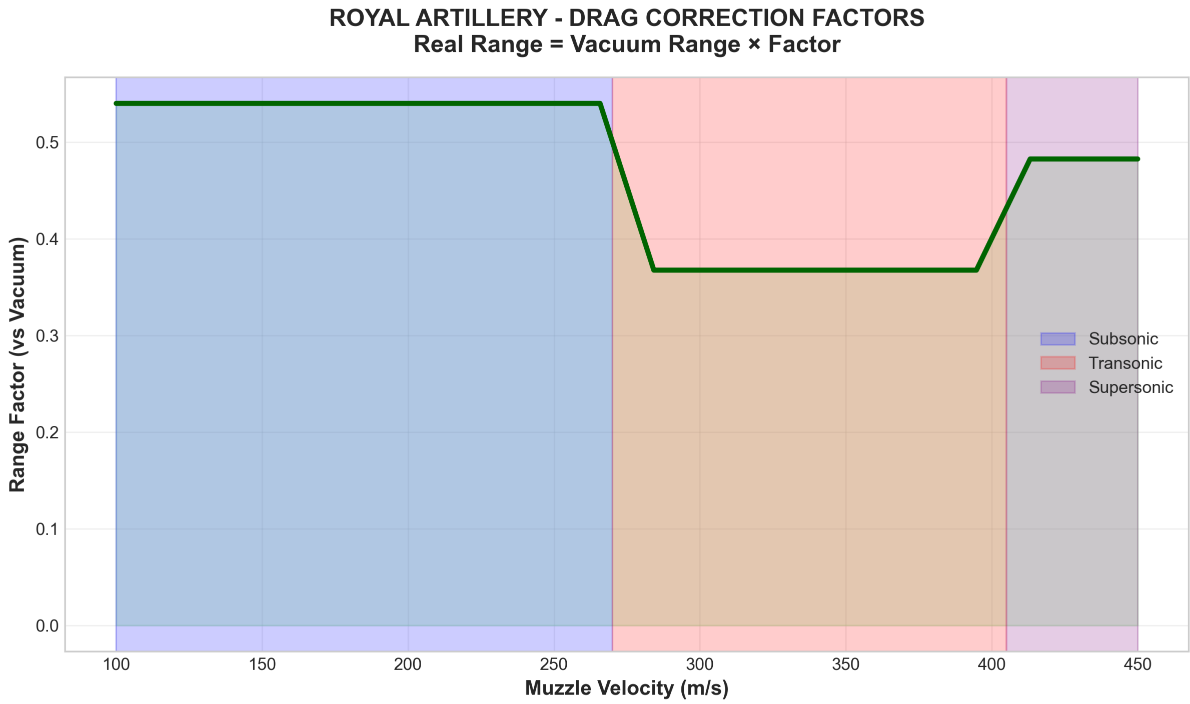

Figure 10: Drag Correction Factors

Drag Correction Factors demonstrate how air resistance reduces actual ranges, particularly at subsonic and transonic speeds.

ROYAL ARTILLERY - BRACKETING CORRECTION TABLE
Angle Adjustments Based on Impact Observation

Range Error	Light Charge	Standard Charge	Heavy Charge
> +200m	+8.0°	+5.0°	+3.0°
+100m to +200m	+5.0°	+3.0°	+2.0°
+50m to +100m	+3.0°	+2.0°	+1.5°
+25m to +50m	+1.5°	+1.0°	+0.8°
-25m to +25m	**HOLD**	**HOLD**	**HOLD**
-50m to -25m	-1.5°	-1.0°	-0.8°
-100m to -50m	-3.0°	-2.0°	-1.5°
-200m to -100m	-5.0°	-3.0°	-2.0°
< -200m	-8.0°	-5.0°	-3.0°

Figure 11: Bracketing Correction Table

ROYAL ARTILLERY - WIND CORRECTION CHART
Range Change = Wind Speed × Flight Time × 0.7

Time\Wind	0 m/s	2 m/s	4 m/s	6 m/s	8 m/s	10 m/s	12 m/s	14 m/s	16 m/s	18 m/s	20 m/s
5s	0m	7m	14m	21m	28m	35m	42m	49m	56m	63m	70m
10s	0m	14m	28m	42m	56m	70m	84m	98m	112m	126m	140m
15s	0m	21m	42m	63m	84m	105m	126m	147m	168m	189m	210m
20s	0m	28m	56m	84m	112m	140m	168m	196m	224m	252m	280m
25s	0m	35m	70m	105m	140m	175m	210m	245m	280m	315m	350m
30s	0m	42m	84m	126m	168m	210m	252m	294m	336m	378m	420m

Figure 12: Wind Correction Chart

Wind Correction and Bracketing Correction tables provide adjustments for environmental effects and range estimation errors, reflecting early ballistic compensation methods.

ARTILLERY OFFICER'S POCKET REFERENCE - 1825

FIELD FIRING TABLE
Elevation Angles for 5kg Spherical Projectile

Range\Vel	150	200	250	300	350	400
400m	8.2°	---	---	---	---	---
600m	13.9°	---	---	---	---	---
800m	19.0°	9.5°	---	---	---	---
1000m	26.0°	---	---	---	---	---
1200m	---	---	---	---	---	---
1400m	---	---	---	---	---	---
1600m	---	---	---	---	---	---
1800m	---	24.7°	---	---	---	---
2000m	---	31.1°	---	15.3°	---	---

FIELD PROCEDURES & RULES

FIELD FIRING PROCEDURE
=========================

1. ESTIMATE RANGE
- Use known distance
- Account for elevation
- Horizontal range only

2. SELECT CHARGE
- 150 m/s: < 800m
- 250 m/s: 800-1500m
- 350 m/s: > 1500m

3. SET ELEVATION
- Use firing table
- Or: $\theta \approx \sqrt{(\text{Range} \div C)}$

C=40(light), 70(std), 100(hvy)

4. FIRE & OBSERVE
- Watch fall of shot
- Note impact location

5. ADJUST & REFIRE
- Short: Increase angle
- Long: Decrease angle
- Use correction table

WIND CORRECTION:
Range change ≈ Wind × Time × 0.7

MEMORY: Real range ≈ 0.6 × Vacuum

Figure 13: Artillery Officer's Pocket Reference

This historical artillery reference outlines practical firing solutions for a 5kg spherical projectile.A table of elevation angles for different ranges and muzzle velocities is included, showing that longer ranges with flatter trajectories are produced at higher velocity. Estimating range, choosing a powder charge, setting elevation, firing, and making adjustments based on results comprise the five-step field technique. Furthermore, wind adjustments use a linear relationship to account for atmospheric impacts. Artillery officers can attain accuracy through iterative modifications while conserving ammunition thanks to this technique, which strikes a compromise between mathematical precision and practicality on the battlefield.

Conclusion

Artillery accuracy depended as much on the atmosphere as on the cannon itself. To quantify those effects, we developed a 3-DOF trajectory model that included altitude-varying air properties from the ICAO Standard Atmosphere, full three-dimensional wind, and a drag coefficient that changed with Mach number. Using this framework for a 5 kg, 11 cm spherical projectile fired up to 450 m/s velocity from 500 m altitude, we found that the projectile's path was far more sensitive to environmental conditions than to small changes in muzzle speed.

In particular, a 10 km/h crosswind produced a 14 m lateral miss at 1,200 m, and a 5% error in estimated air density produced a range error exceeding 50 m. These findings show that wind speed and air density are the dominant sources of error in historical ballistics.

Finally, the simulation results were reduced to straightforward firing tables and additive correction rules for altitude and wind, giving practical guidance an officer could use in the field.

References

[1] "Dardanelles Gun," in *Wikipedia, The Free Encyclopedia*, Wikimedia Foundation, 9 September 2025. Accessed: 10 November 2025. [Online]. Available: `https://en.wikipedia.org/wiki/Dardanelles_Gun`

[2] D. E. Carlucci and S. S. Jacobson, *Ballistics: Theory and Design of Guns and Ammunition.* Boca Raton, FL: Taylor & Francis Group, 2007.

[3] Embry-Riddle Aeronautical University, "International Standard Atmosphere (ISA)," in *Introduction to Aerospace Flight Vehicles*, 2023. Accessed: 10 November 2025. [Online]. Available: `https://eaglepubs.erau.edu/introductiontoaerospaceflightvehicles/chapter/international-standard-atmosphere-isa/`

[4] N. Huzyk, N. Sokulska, and L. Velychko, "Mathematical model of external ballistics of projectiles," *Information Processing Systems*, vol. 172, no. 1, 2023.

A Appendix A: Realistic Case.m

```
import numpy as np
from scipy.integrate import solve_ivp
from scipy.optimize import minimize_scalar
from scipy.interpolate import interp1d

class CannonBall:
    def __init__(self):
        self.m = 5.0
        self.diameter = 0.11
        self.radius = self.diameter / 2
        self.A = np.pi * self.radius**2
        self.rho = 1.16727
        self.g = 9.81
        self.speed_of_sound = 338.370
        self.setup_drag_coefficient_table()

    def setup_drag_coefficient_table(self):
        mach_points = np.array([0.1, 0.3, 0.5, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.5, 1.8, 2.0, 2.5, 3.0, 4.0])
        cd_points = np.array([0.47, 0.47, 0.47, 0.47, 0.47, 0.55, 0.65, 0.75, 0.85, 0.95, 1.05, 1.15, 1.15, 1.15, 1.15, 1.15])
        self.cd_interp = interp1d(mach_points, cd_points, kind='linear', fill_value=(cd_points[0], cd_points[-1]), bounds_error=False)

    def get_drag_coefficient(self, mach):
        return float(self.cd_interp(mach))

    def get_mach_number(self, velocity):
        return velocity / self.speed_of_sound

def trajectory_equations(t, state, cannonball, wind_x=0, wind_y=0):
    x, y, vx, vy = state
    vx_rel = vx - wind_x
    vy_rel = vy - wind_y
    v_rel = np.sqrt(vx_rel**2 + vy_rel**2)

    if v_rel == 0:
        ax_drag, ay_drag = 0, 0
    else:
        mach = v_rel / cannonball.speed_of_sound
        Cd = cannonball.get_drag_coefficient(mach)
        k = 0.5 * Cd * cannonball.rho * cannonball.A
        F_d = k * v_rel**2
        ax_drag = -F_d * vx_rel / (v_rel * cannonball.m)
        ay_drag = -F_d * vy_rel / (v_rel * cannonball.m)

```

```
    ax = ax_drag
    ay = -cannonball.g + ay_drag
    return [vx, vy, ax, ay]

def compute_trajectory(v0, theta, cannonball, wind_x=0, wind_y=0, y0=500, t_max=60):
    theta_rad = np.radians(theta)
    initial_state = [0, y0, v0 * np.cos(theta_rad), v0 * np.sin(theta_rad)]

    def hit_ground(t, state, *args):
        return state[1]
    hit_ground.terminal = True
    hit_ground.direction = -1

    sol = solve_ivp(trajectory_equations, [0, t_max], initial_state,
                    args=(cannonball, wind_x, wind_y),
                    events=hit_ground, dense_output=True,
                    rtol=1e-8, atol=1e-10, max_step=0.1)
    return sol

def find_impact_range(v0, theta, cannonball, **kwargs):
    if 'y0' not in kwargs:
        kwargs['y0'] = 500
    sol = compute_trajectory(v0, theta, cannonball, **kwargs)
    if sol.t_events[0] is not None and len(sol.t_events[0]) > 0:
        return sol.t_events[0][0], sol.y_events[0][0][0]
    else:
        return None, sol.y[0][-1]

def objective_function(theta, v0, target_range, cannonball):
    t_impact, actual_range = find_impact_range(v0, theta, cannonball)
    return abs(actual_range - target_range)
```

B Appendix B: Artillery Officer.m

```
import numpy as np

class HistoricalArtilleryOfficer:
    def __init__(self):
        self.g = 9.81
        self.standard_velocities = [100, 150, 200, 250, 300, 350,
            400, 450]
        self.mass = 5.0
        self.diameter = 0.11
        self.cross_area = 0.0095
        self.air_density = 1.167
        self.sound_speed = 338.0

    def vacuum_range(self, v0, theta):
        theta_rad = np.radians(theta)
        return (v0**2 * np.sin(2 * theta_rad)) / self.g

    def drag_reduction_factor(self, v0, theta):
        mach = v0 / self.sound_speed

        if mach < 0.8:
            reduction = 0.40
        elif mach < 1.2:
            reduction = 0.55
        else:
            reduction = 0.45

        angle_factor = 1.0 + (theta / 90.0) * 0.3
        return 1.0 - (reduction * angle_factor)

    def field_range_estimate(self, v0, theta):
        R_vacuum = self.vacuum_range(v0, theta)
        drag_factor = self.drag_reduction_factor(v0, theta)
        return R_vacuum * drag_factor

    def officers_initial_estimate(self, target_range, v0):
        if v0 >= 400:
            return np.sqrt(target_range / 100) + 1.0
        elif v0 >= 250:
            return np.sqrt(target_range / 70) + 2.0
        else:
            return np.sqrt(target_range / 40) + 5.0

    def bracketing_method(self, v0, initial_angle, target_range,
        max_iterations=3):
        current_angle = initial_angle

        for iteration in range(max_iterations):
```

```
            current_range = self.field_range_estimate(v0,
               current_angle)
            range_error = target_range - current_range

            if abs(range_error) < 25:
                break
            elif range_error > 200:
                correction = 8.0
            elif range_error > 100:
                correction = 5.0
            elif range_error > 50:
                correction = 3.0
            elif range_error > 0:
                correction = 1.5
            elif range_error > -50:
                correction = -1.5
            elif range_error > -100:
                correction = -3.0
            elif range_error > -200:
                correction = -5.0
            else:
                correction = -8.0

            current_angle += correction
            current_angle = max(1.0, min(80.0, current_angle))

        return current_angle

    def find_firing_solution(self, target_range,
       available_velocity=None):
        if available_velocity is None:
            available_velocity = self.standard_velocities

        solutions = []

        for v0 in available_velocity:
            theta_initial = self.officers_initial_estimate(
               target_range, v0)
            theta_final = self.bracketing_method(v0,
               theta_initial, target_range)

            estimated_range = self.field_range_estimate(v0,
               theta_final)
            error = estimated_range - target_range

            if abs(error) < 50:
                solutions.append({
                    'velocity': v0,
                    'angle': theta_final,
                    'estimated_range': estimated_range,
                    'error': error
```

```
                    })

        return sorted(solutions, key=lambda x: abs(x['error']))
```

A Appendix C: Additional Figures

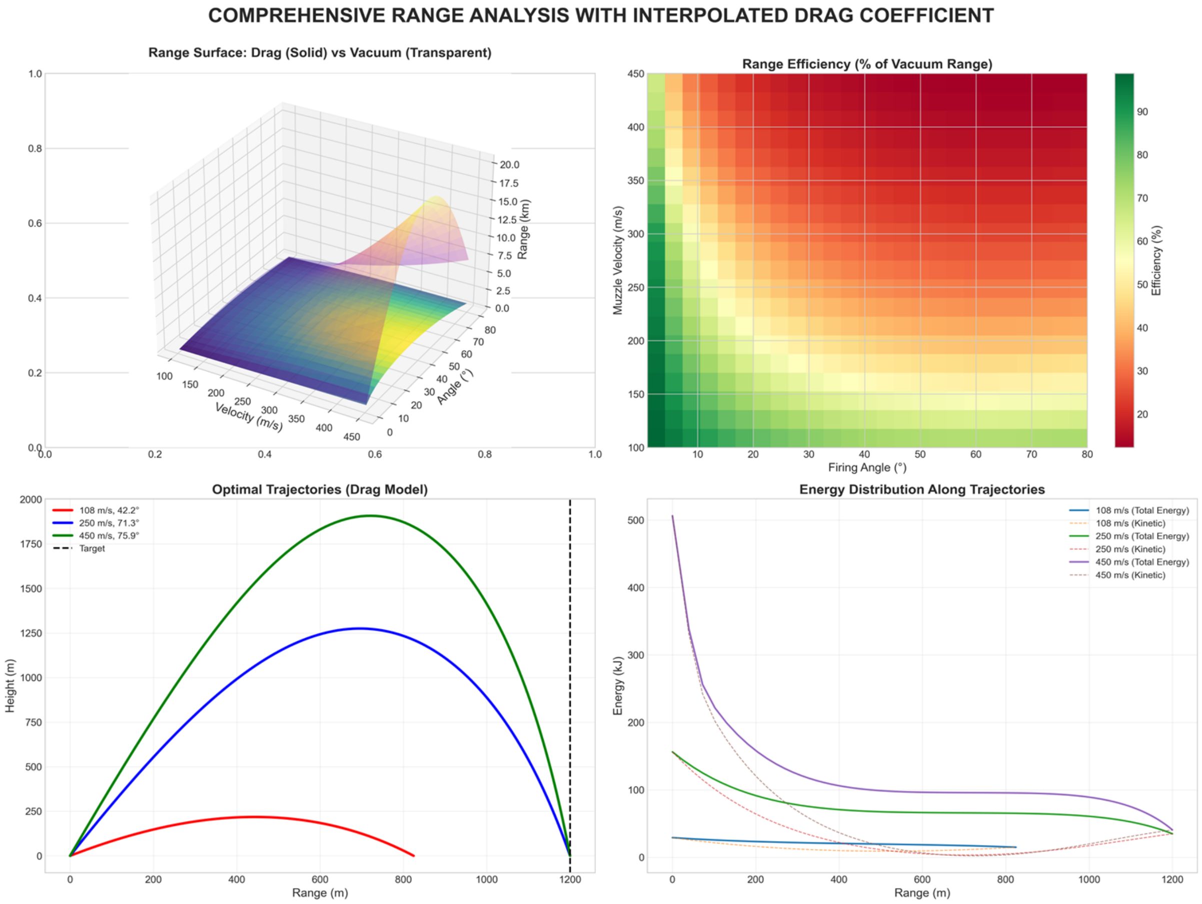

Figure 14: Comprehensive Range Analysis with Interpolated Drag Coefficient

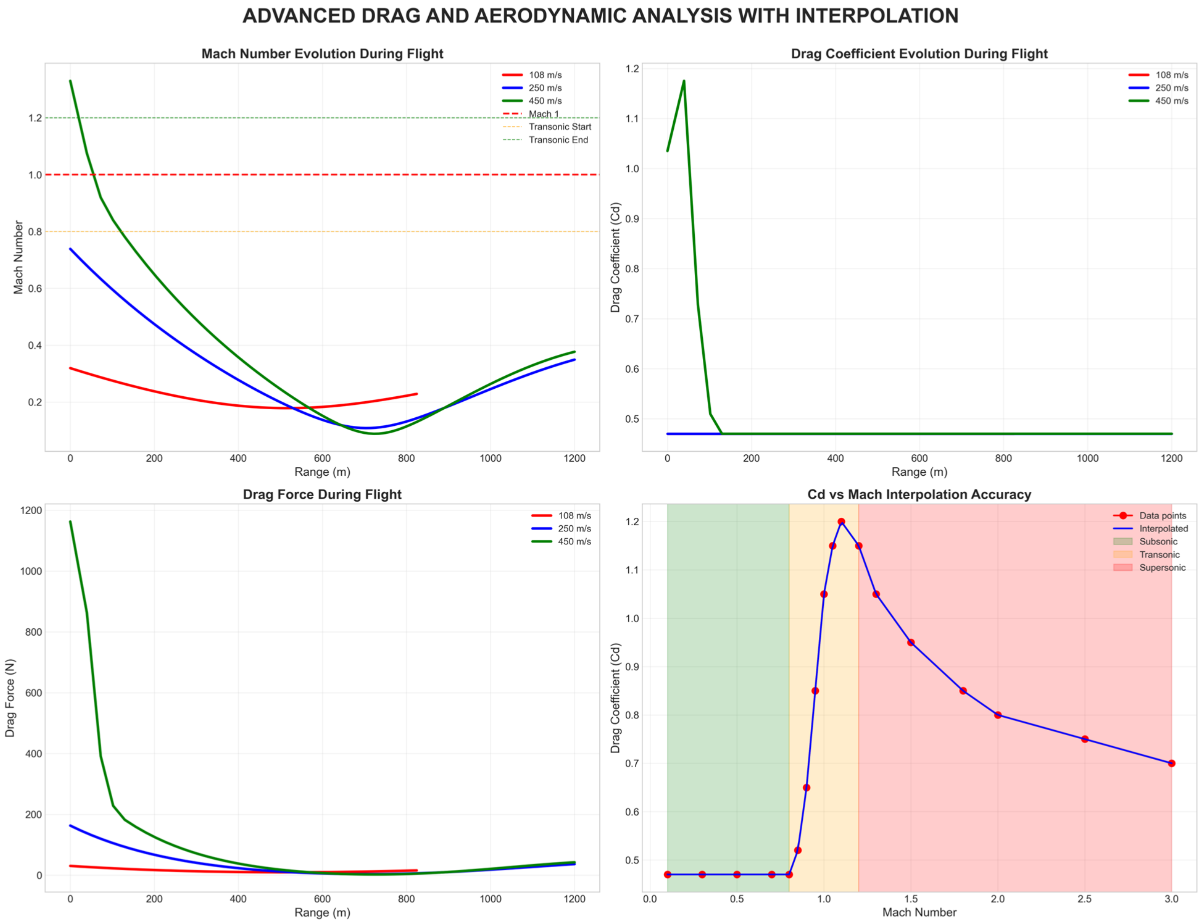

Figure 15: Advanced Drag and Aerodynamics Analysis with Interpolation