Empirical Data to Determine Transonic Drag Coefficient

Oleg Mazonka[1]

2016

Abstract

This paper presents a computational model for external ballistics. The model includes physical aspects most important for precise projectile trajectory calculations, yet is limited to those which are usually known and well defined, such as projectile mass and muzzle velocity. The most difficult but still important ingredient is the drag coefficient. Its estimation is complex especially surrounding the supersonic velocity threshold. Since the drag coefficient depends on velocity, it must be represented as a function. In the present work this function is hypothesized to be formed out of two components: fluid drag and dry friction. The later component appears only for supersonic velocities. This decomposition makes parametrisation of the drag function simple. The presented model allows the drag coefficient to be deduced from empirical data of arbitrary form, amount, and consistency. The model tries to adjust the parameters of the drag function to fit any available empirical data, thus making ballistic trajectory calculations closer to the reality.

Contents

1 Introduction

Standard ballistic calculators require input data in a form of a set of parameters that are known upfront. Those include projectile mass, calibre, initial velocity and others. Sophisticated calculators may require other physical properties such as atmospheric humidity or barrel twist. The ballistic coefficient among other parameters greatly affects the result and is usually required to be known. Ballistic movement through the air is a complex phenomenon and there is no strict theoretical way to calculate this parameter from the physical factors of the projectile and air. Besides, the ballistic coefficient being a number works well on short trajectories where the velocity of the projectile does not change much or well below the speed of sound. More precise ballistic calculators use a drag function, which represents drag coefficient depending on the projectile velocity. The drag coefficient C_d is counter proportional to the ballistic coefficient. It is defined via relation between drag force F_d and the projectile velocity v as:

$$F_d = \frac{1}{2} C_d A \rho v^2 \tag{1}$$

where A is the cross section of the projectile, and ρ is the air density. This relation naturally appears from the drag coefficient definition as the ratio between the work produced by air friction E_d and the kinetic energy E_k required to accelerate the mass of displaced air to the velocity v, i.e. $E_d = C_d E_k$.

When a projectile travels distance s, the work produced by the air friction is $E_d = F_d s$. On the other hand, the mass of the displaced air m_a is the product of the air density and the displaced volume: $m_a = \rho A s$. The energy to accelerate it to the velocity v is $E_k = m_a v^2/2$. Putting together these relations results in Eq (1).

Ballistic models deducing the drag function from theoretical principles may have certain success. However, if the theoretical derivations are based on the shape of the projectile and include six degrees of freedom effects (e.g. such as Coriolis effect), the input data required from the final user may be prohibitive. The model presented in this work tries to fit a simple physics model to empirical data that often is limited and varying in format.

Contribution: The novelty of the present work includes the following methods.

- The drag function is expressed (and parametrised accordingly) as a sum of two simpler components representing fluid and dry friction, that allows simple yet efficient parametrisation of the drag function.
- The algorithms adjust the drag function parameters to fit any empirical data such as maximal range or different trajectory parameters with known or unknown elevation angle.

The ballistic model is selected so to reduce upfront knowledge about physical parameters that are difficult to find, but without artificial internal simplifications such as flat earth surface and constant air density and temperature.

2 Ballistic Model

2.1 Overview

The model uses simple integration of the Newtonian equation

$$\dot{v} = f_g + f_d + f_s + f_R \tag{2}$$

where $\dot{v}$ on the left side is the acceleration of the projectile and on the right side is the sum of specific forces (per mass unit of the projectile): f_d gravitational, f_d drag and range wind, f_s side wind, and f_R fluctuation. The

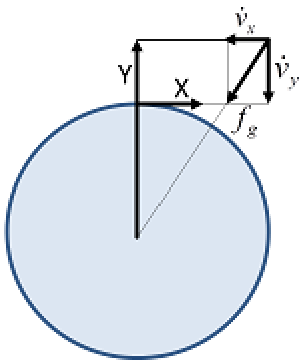

Figure 1: Schematic representation of the Earth, coordinate system, and the gravitational force.

Property	Included
Trajectory calculation, 3D	Yes
6DOF	No
Curvature of the Earth's surface	Yes
Orientation of the projectile to wind	Yes
Air density and temperature change with altitude	Yes
Range wind and cross wind	Yes
Turbulence	Yes
Non-spherical shape of the Erath	No
Orientation of the projectile	No
Precession of the projectile due to spin	No
Stabilization of the projectile	No
Drift due to gyroscopic effect	No
Drift due to Magnus effect	No
Drift due to Coriolis effect	No

Table 1: Features considered for the model.

coordinate system is Cartesian (Figure 1) with X-axis directing along the tangent to the earth's surface in the plane of initial trajectory, Y-axis pointing up, perpendicular to the Earth's surface, and Z-axis is perpendicular to the initial projectile's direction. The coordinate system's origin is the initial position of the projectile shot from zero height. Without side wind and fluctuations the trajectory remains in XY plane.

Table 1 lists the features included and not included in the model. Six degrees of freedom (6DOF) calculations used in sophisticated ballistic models [5] are not included in the calculations because of two reasons: 1) the physics (and calculations) is complex to model making the computation slower; and 2) such calculations require the knowledge of projectile's spin and other projectile's physical properties, which are rarely known. This simplification leads to the assumption that the projectile is perfectly stabilized, i.e. in the absence of wind the drag force is directed along the trajectory line and the cross section influencing the drag is constant (no wobbling). On the other hand the length of the projectile is usually known, so when the wind is present, the cross section and the corresponding forces can be adjusted.

There is no point to assume the Earth's surface flat, but including non-spherical shape of the Earth would require knowledge of geolocation of the trajectory. Hence the model assumes the Earth to be a perfect sphere. There is data describing standard change of air density and temperature with altitude, which can be incorporated into the model without any extra input from the user. Therefore this feature is also included.

In the absence of wind and air turbulences, two main forces: gravitational and drag, determine the trajectory of the projectile. Wind and air turbulence require additional input from the user. They are discussed in Section 5.

2.2 Gravitational force

The gravitational specific force is given by Newton's law:

$$f_g = GM_E\frac{1}{r^2}$$

where G is the gravitational constant, M_E is the Earth's mass, and r is the distance from the centre of the Earth to the projectile.

Ignoring Z coordinate in gravitational force is equivalent to assuming Earth's surface to be flat in Z direction – perpendicular to the trajectory. Since the deviation of trajectory from its course due to the side wind or atmospheric turbulences is small comparing to the flight range, this assumption is fair.

At a point (x, y) the acceleration in XY directions is equal to:

$$\dot{v_x} = -GM_E\frac{x}{r^3}, \qquad \dot{v_y} = -GM_E\frac{R_E + y}{r^3}$$

where R_E is the Earth's radius. In the above expressions x/r and $(R_E + y)/r$ are sine and cosine of the angle between the origin and the point seen from the centre of the Earth (Figure 1).

2.3 Drag force

The specific drag force f_d in Eq (2) is defined by Eq 1:

$$f_d = \frac{1}{2m}\rho(h)C_d(\mu(h))Av^2 \tag{3}$$

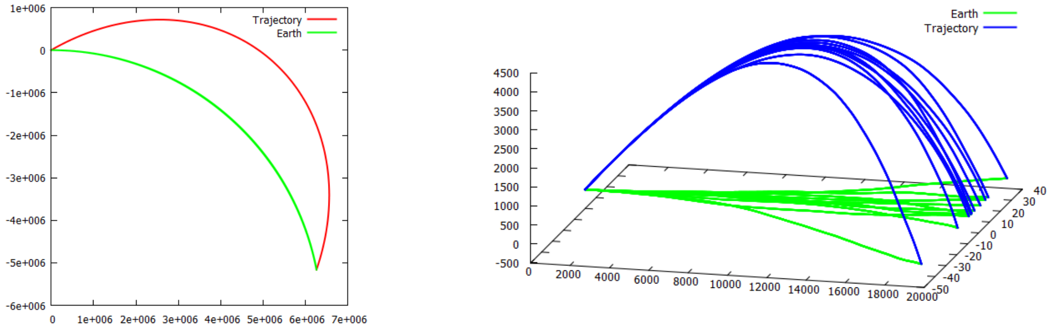

Figure 2: Example of trajectories. Left figure: A projectile shot at 30° at 7000 ms^{-1} with no drag. The green line represents the earth surface. Right figure: M107 projectile shot at 30° with air turbulence parameter on. Ten trajectories are shown.

where m is the mass of the projectile, $\rho(h)$ is the air density depending on altitude h, $C_d(\mu)$ is the drag function depending on Mach number μ, and A is the projectile's cross section. Mach number depends on speed of sound c which in turn depends on altitude: $\mu(h) = v/c(h)$. Altitude h is equal to $h = r - R_E$, where $r = \sqrt{x^2 + (y + R_E)^2}$.

According to [4, 6] air density ρ and speed of sound c can be calculated by the following formulae.

$$\rho = \frac{(T/T_0)^{\frac{gM_a}{R_g L_k}} M_a P_0}{10^3 R_g T}$$

$$c = c_0\sqrt{T}$$

$$T = T_0 - L_k \frac{10^3 R_E h}{R_E + h}$$

where T is the air temperature and the following constants are used: P_0 is sea level standard pressure in Pascals, equal to 101325 Pa; M_a is molecular weight of dry air in gram per mole, equal to 28.9644 gm/mol; T_0 is sea level standard temperature in Kelvins, equal to 288.15 K; R_g is universal gas constant in Joules per mole-Kelvin, equal to 8.31432 J/(mol K); L_k is temperature lapse rate in Kelvin per kilometre, equal to 6.5 K/km; g is free-fall acceleration, equal to $GM_E R_E^{-2}$ and assumed 9.80665 m/s^2; c_0 is speed of sound at 1K temperature, equal to 20.046 m/s; R_E is the Earth's radius in meters, equal to 6356766 m; and h is the altitude in meters.

Figure 2 shows the trajectories calculated by the model. The left graph shows the trajectory without drag simulated by setting $A = 0$ in XY plane.

The Earths surface is shown with a green line. The right picture shows 10 trajectories of M107 projectile calculated with drag and turbulences. The calculation of trajectories is fast enough – hundreds of trajectories are calculated within a second on a common PC.

3 Drag Function

When a body moves through the air it is surrounded by air molecules moving at the same speed. In other words, the body rests in its neighbourhood surrounding. The drag force is the result of molecule exchange between the surrounding and the air relatively moving at constant speed v. Let $n_U = nS_e u$ be the rate of molecule exchange due to thermal movement u of molecules, where S_e is some effective area through which exchange occurs and n is molecule density. Then $S_e u$ expresses a volume per time unit, and n_U – a number of molecules per second. Let $n_V = nA_e v$ be the rate of molecule exchange due to outside molecules flying into the surrounding with the speed v through some effective cross section A_e. Each molecule of mass $m_0 = \rho/n$ leaving the surrounding removes the momentum $m_0 u$, but each molecule entering the surrounding brings in the momentum $m_0(u+v)$. This means that the net rate of momentum received by the body would be

$$F_d = -(n_U + n_V)m_0 u + (n_U + n_V)m_0(u + v)$$

which gives

$$F_d = \rho S_e uv + \rho A_e v^2$$

The functions S_e and A_e are unknown. If they are redefined with other arbitrary functions α and β as $S_e u = Ac\beta/2$ and $A_e = A\alpha/2$, then drag force takes a form

$$F_d = \frac{1}{2}(\alpha + \beta/\mu)\rho A v^2$$

Comparing to Eq (1) drag coefficient becomes

$$C_d = \alpha + \beta/\mu$$

The first term in the formula contributes to F_d proportionally to v^2 and corresponds to fluid drag. The second term contributes proportionally to the first order of velocity v and is closer in its form to dry friction. In this work we hypothesise that the functions α and β change little below and above sonic barrier, when $\mu \ll 1$ or $\mu \gg 1$. For example, if $\alpha = 0.1$ and $\beta = 0, \mu < 1$ or $\beta = 0.3, \mu > 1$, the function qualitatively resembles a drag function obtained from experimental data [3]. Figure 3 shows its comparison

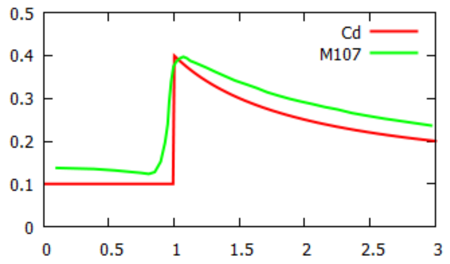

Figure 3: Simplistic C_d - red line, and experimental data for M107 155mm howitzer projectile green line.

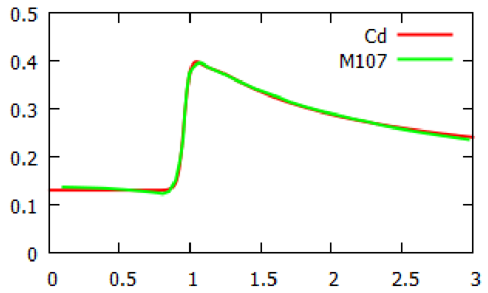

Figure 4: Function C_d fit to experimental data with smooth step transition function.

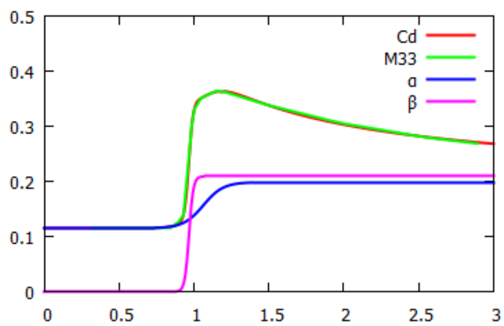

Figure 5: Function Cd fit to experimental data Ball M33 Cal .5. Functions α and β are shows as well.

to the drag function curve obtained by fitting to experimental data points for the howitzer projectile M107 155mm [3].

Passing the critical point at $\mu = 1$ can be smoothed by a symmetric function $(1 + \exp(-x))^{-1}$. So the functions α and β can be parametrized with the following 7 parameters

$$\alpha(\mu) = \alpha_0 + \frac{\alpha_1}{1 + \exp(-w_\alpha^{-1}(\mu - \alpha_\mu))}$$
$$\beta(\mu) = \frac{\beta_1}{1 + \exp(-w_\beta^{-1}(\mu - \beta_\mu))}$$

These functions are smoothed step functions with a step around points α_μ and β_μ, and the widths w_α and w_β. Roughly speaking α is equal to α_0 below and to $\alpha_0 + \alpha_1$ above sonic speed, and β is equal to 0 below and to β_1 above sonic speed. Figure 4 shows the same example as above but with smoothed functions α and β.

This parametrisation fits well another projectile type Ball M33, Calibre 12.7mm [3]. Figure 5 shows C_d along with functions α and β that change their values around sonic barrier.

4 Algorithms and Optimisation

4.1 Drag function optimisation

Once drag function is defined as described in Section 3, the Newtonian equation, Eq (2), can be integrated. The remaining unknown parameters, parametrisation of C_d denoted here as $[C_d]$, can be tuned by comparing the calculation with empirical data. First, empirical data may contain information about different trajectories with different values such as initial velocity, elevation angle, angle of fall, time of flight, range, final velocity, and apex height and apex range. Usually a subset of these values is available. If elevation angle is missing, the model must solve the problem by finding the angle giving the known value for the range. In this case there are two solutions representing high and flat trajectories, because there are two different elevation angles that give the same range. The model guesses the type of trajectory using other available data, for example time of flight, final velocity, etc. Second, empirical data may define only maximal range. In this case, the model must solve the maximal range problem given initial velocity of the projectile – to find the elevation angle which sends the projectile to the maximal distance.

Inside the model each piece of information corresponding to one particular trajectory is presented as a record. A set of all records is the total information about projectile provided to the model. The model runs with a guess $[C_d]$, solving if necessary for the best matching trajectory for each record. Once this step is done, the result can be presented in exactly the same form as the initial set of records. This time, however, the values calculated by the model may differ from the original values. When the values are close to the initial data, one can say that the model adequately describes the empirical data. If the values are not close, then there is a possibility that $[C_d]$ can be modified to get better fit. To measure how well the model matches the data, a utility function U is introduced

$$U = \sum_{i=1}^{N} (\ln x_i - \ln y_i)^2$$

This formula sums over all N values x_i and y_i in all records, where x_i are of the input data and y_i are calculated. Logarithms are used here to make relative error contribute on the same scale regardless of the magnitude of absolute values. An advanced global optimisation algorithm is run to minimise U

$$[C_d]^* = \arg\min U([C_d])$$

finding the optimal parametrisation for C_d. Advanced optimisation algorithms have to be used because evaluation of function U is slow for different $[C_d]$ – each evaluation requires solving for the best matching trajectory for each record. Moreover unavoidable computational error makes the values of U not smooth on some threshold scale of precision, i.e. two close sets of $[C_d]$ do not converge to the same U up to some level of computational error. This makes the calculation of derivatives in global optimisation problematic. These kinds of constraints require calculation of trajectories to be fast but also precise. The model uses Runge-Kutta method of integration with a validity check based on conservation of energy.

4.2 Energy conservation

The projectile moving in the gravitational field without additional forces conserves the sum of its potential and kinetic energies. Since the equations of movement are analytical derivatives from the expressions of potential and kinetic energies, they are linearly independent. Therefore, the total energy is conserved only in theory, but not during numerical integration. The integration step governs the precision of calculation. In a smooth physical system where the forces change gradually with the gradual change of the state of the system, smaller step size of integration converges to the theoretical solution with a smaller violation of energy conservation. In this case the user may define a portion to total energy that is allowed to be volatile (a typical value for this can be of the floating precision order $\approx 10^{-15}$). The model uses three levels of energy conservation error tolerance: minimal, maximal, and rejecting. If the error stays between minimal and maximal, the step size of the integration is not affected. If the error descends below minimal, the step size is increased; if above maximal, the step size is decreased. If the error rises above rejecting level, the step is rejected and recalculated with a smaller step size.

The drag, wind and turbulence forces are excluded from the energy conservation calculations.

4.3 Potential energy

Calculation of energy while integrating must be fast and precise. The kinetic energy formula $e_k = v^2/2$ is straightforward and stable, but calculation of potential energy using a direct formula does not satisfy the precision requirements.

The difference of potential energy between points $a = (x_a, y_a)$ and $b = (x_b, y_b)$ is given by the relation:

$$e_p = GM_E \left(\frac{1}{r_a} - \frac{1}{r_b} \right)$$

where e_p is the specific potential energy difference between points a and b, G is the gravitational constant, M_E is the Earths mass, and r_a, r_b are distances from the centre of the Earth to corresponding points. This equation cannot be used directly in computation because the computational error is too high due to the fact that the Earths radius R_E is too big compared to the difference in altitudes of points a and b. To overcome this problem the expression must be rearranged so the subtractions are expressed in coordinates and not in distances from the centre of the Earth:

$$e_p = GM_E \frac{2R_E(y_b - y_a) + y_b^2 - y_a^2 + x_b^2 - x_a^2}{r_a r_b (r_a + r_b)}$$

Here a relation $r^2 = (R_E + y)^2 + x^2$ has been used. This expression remains computationally stable and precise regardless of the balance between R_E and point coordinates.

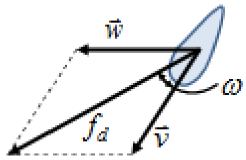

Figure 6: Law of cosine defines the angle at which the air hits the projectile.

5 Atmospheric Effects

5.1 Wind considerations

The projectiles cross section A depends on the direction of air flow around the projectiles body. Without the range wind this cross section is equal to the frontal cross section A_f, because perfect stabilization is assumed.

If a range wind exists, then the full cross section becomes a combination of the frontal A_f and the side A_s

$$A = A_f \cos\omega + A_s \sin\omega$$

with ω being an angle between the trajectory and the direction of air flow around the projectiles body. The trajectory direction is the direction of the air flow around the projectiles body in the absence of range wind. So the two velocities $\vec{v}$ and $(\vec{v}+\vec{w})$ form a triangle as shown in Figure 6, where w is the range wind. The angle ω can be calculated from the law of cosine:

$$\cos\omega = \frac{v^2 + (\vec{v}+\vec{w})^2 - w^2}{2\sqrt{(\vec{v}+\vec{w})^2 v^2}}$$

Side wind is an external parameter which plays rather phenomenological role and does not directly influence the main effect – forward propagation through the air. Since the ballistic characteristics of the projectile moving sideways are not known, the model assumes side drag coefficient to be of a unit value (close to side cylinder at small speed), so the side force acting on the projectile is equal to:

$$f_s = \frac{1}{2m}\rho A_s w_Z^2$$

where A_s is side cross section and w_Z is the side wind.

Having calculated ρ and μ for f_d, the drag coefficient C_d is obtained by querying a parametrized function described in Section 3.

5.2 Turbulence force

The specific turbulence force f_R is modelled as a Wiener process with the variance:

$$\mathrm{Var}(f_s\delta) = \widetilde{f_0}\sqrt{\frac{f_d\delta}{v}}$$

where δ is time interval of integration, and $\widetilde{f_0}$ is an external parameter – the strength of turbulences defined in meters per second intuitively corresponding to average movement of the body in the air without any external force. This equation is derived in [2]. Here $\widetilde{f_0} = \sqrt{2E}$ plays the role of energy parameter E.

6 Conclusion

In this work an external ballistic model is presented. The model can be fed with empirical data of known ballistic trajectories, resulting in better fit for drag coefficient function and therefore better precision of the model. The drag function is split into two components: one relating to fluid drag and the other to dry friction. This allowed a new successful parametrisation of drag function. The code implementing the model is open source [1].

References

[1] *Github*. URL: https://github.com/mazonka/artill (cit. on p. 13).

[2] Oleg Mazonka. "Easy as Pi: The Fluctuation-Dissipation Relation". In: *Journal of Reference* 16 (2016). URL: http://jrxv.net/x/16/fdt.pdf (cit. on p. 13).

[3] Robert L. McCoy. *Modern Exterior Ballistics*. 1999 (cit. on pp. 7, 9).

[4] Nasa. "U.S. Standard Atmosphere". In: (1976). URL: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19770009539.pdf (cit. on p. 6).

[5] *PRODAS*. URL: http://www.prodas.com/ (cit. on p. 4).

[6] Richard Shelquist. "An Introduction to Air Density and Density Altitude Calculations". In: (2016). URL: https://wahiduddin.net/calc/density_altitude.htm (cit. on p. 6).

Author address

1. **Oleg Mazonka**, Defence Science and Technology Group, Australia.
`mailto:Oleg.Mazonka@dsto.defence.gov.au`