

Contents

ScopeDOPE Ballistic Computer: Architecture, Physics, and the DOPE-Blend™

Field-Data Fusion Framework	2
Abstract	2
1. Introduction	3
2. Theoretical Background	4
2.1 Point-Mass Exterior Ballistics	4
2.2 Reference Drag Functions: G1 and G7	4
2.3 Ballistic Coefficient	5
2.4 JBM Ballistics as Industry Reference	5
3. Atmospheric Model	5
3.1 Speed of Sound	5
3.2 Air Density Ratio	5
3.3 Atmospheric Sensitivity	6
3.4 Effective BC Scaling	7
4. Physics Model	7
4.1 Drag Deceleration and the KD Constant	7
4.2 G7 Drag Table Calibration	7
4.3 Zeroing Integration	8
4.4 Trajectory Elevation in Milliradians	8
4.5 BC from Form Factor	9
5. Numerical Methods	9
5.1 Spatial Domain Formulation	9
5.2 Fourth-Order Runge-Kutta Integration	9
5.3 Time-of-Flight Derivation	10
5.4 Drop Integration	10
5.5 Fritsch-Carlson Monotone Cubic Hermite Interpolation	10
6. Secondary Effects	11
6.1 Windage: Didion Lag Rule	11
6.2 Spin Drift: Miller Gyroscopic Stability	12
6.3 Coriolis Effect	12
6.4 Inclined Fire	13
7. DOPE-Blend™ Field-Data Fusion Framework	13
7.1 Motivation and Conceptual Framework	13
7.2 Bias Residual Computation	14
7.3 Bias Interpolation	14
7.4 Windage Exclusion Rationale	15
7.5 Novelty and Differentiation	15
8. Implementation Architecture	16
8.1 Kotlin Multiplatform Structure	16
8.2 Visibility and Trade-Secret Encapsulation	16
8.3 Offline-First Design	16
8.4 Numerical Precision Choices	16
8.5 Integrator Step Size Selection	17
9. Validation Study	17
9.1 Test Suite Architecture	17
9.2 Accuracy Tiers and Thresholds	17

9.3 Combinatorial Cross-Validation Methodology	18
10. Results	19
10.1 Primary Validation Verdict	19
10.2 Cross-Calculator Accuracy: G7 Precision Loads	19
10.3 JBM Cross-Reference: Tier 3 Confirmed-Fired Case	20
10.4 Known Open Issues	21
11. Discussion	21
11.1 G7 Calibration Scope and Generalizability	21
11.2 Transonic Regime Limitations	21
11.3 Comparison to Calculator-Class Approaches	22
11.4 Field Deployment Considerations	22
12. Conclusion	22
References	23

ScopeDOPE Ballistic Computer: Architecture, Physics, and the DOPE-Blend™ Field-Data Fusion Framework

Cold Bore Ballistics LLC — Technical Report TR-2026-001

Jason M. Schwefel

Cold Bore Ballistics LLC

Revision 1.0 — 2026-06-01

Abstract

ScopeDOPE is a field Ballistic Computer for precision rifle shooting, implemented as a Kotlin Multiplatform (KMP) application targeting Android and iOS. In contrast to ballistic calculators, which apply physics models to produce static trajectory predictions from manufacturer-supplied parameters alone, ScopeDOPE incorporates DOPE-Blend™ — a proprietary field-data fusion framework that continuously refines trajectory predictions by integrating observed elevation adjustments recorded during field shooting sessions. This paper presents the complete technical architecture of the ScopeDOPE ballistics engine: a point-mass exterior ballistics solver that numerically integrates the drag equation in the spatial domain using a fourth-order Runge-Kutta (RK4) scheme with Fritsch-Carlson monotone cubic Hermite interpolation of tabulated drag coefficient data. The atmospheric model implements the ICAO Standard Atmosphere tropospheric pressure lapse formula, the Magnus formula for saturation vapor pressure and humidity correction, and separate integration at zeroing conditions to correctly account for environmental differences between the zero session and field deployment. Secondary effects include the Didion lag-rule windage model, Miller gyroscopic stability for spin drift magnitude, and per-integration-step Coriolis acceleration computation. Validation against JBM Ballistics as the industry G7 reference demonstrates agreement within 0.012 mil at 1000 yards for representative precision loads following application of a $1.26\times$ McCoy G7 drag table calibration factor. DOPE-Blend™ is validated via exhaustive combinatorial $C(N,k)$ cross-validation across 37 test cases spanning 22 calibers and three accuracy tiers. The engine runs entirely on-device with no server dependency and no user account infrastructure, constituting a complete, offline-first solution for field-deployable precision ballistics computation.

1. Introduction

Precision long-range rifle shooting imposes accuracy requirements on ballistic solutions that are challenging to meet with physics-only computation. The governing parameter — ballistic coefficient (BC) — is measured under controlled conditions and published by projectile manufacturers, yet real-world BC values are subject to systematic variability from lot-to-lot manufacturing tolerances, bore conditioning, seating depth variations, and aerodynamic phenomena that the G1 and G7 reference functions do not fully capture. Atmospheric conditions at the firing line can diverge substantially from those at the time of zeroing. These compounding sources of error mean that even a correctly implemented physics solver can produce elevation solutions that differ from the true solution by amounts that matter at extended range: at 1000 yards, a 1% BC error translates to roughly 0.3 mil of elevation error, which is three times the width of a 1-MOA group.

The conventional response — ballistic calculators that expose BC as a user-adjustable parameter — requires the shooter to reverse-engineer their true BC from field observations, re-enter it manually, and accept that the correction is global rather than range-specific. This approach cannot distinguish between BC uncertainty, atmospheric model error, barrel-specific aerodynamic behavior, or any other source of systematic trajectory deviation. Each source of error is conflated into a single scalar correction that becomes invalid whenever conditions change.

ScopeDOPE addresses this limitation through DOPE-BlendTM: a proprietary field-data fusion framework that separates the problem into its constituent parts. The physics engine computes a trajectory prediction from known inputs. Field observations — the shooter’s actual confirmed holds at specific ranges under specific recorded conditions — are converted to bias residuals in the physics model’s own output space. When conditions change (different altitude, temperature, muzzle velocity, or wind), the physics engine re-projects to the new conditions, and the DOPE-Blend spline is applied on top, correcting for the irreducible systematic error the physics model cannot account for. Neither the physics model alone nor the field data alone achieves what both together produce: a trajectory prediction that is both environmentally adaptive and empirically grounded.

This paper makes the following technical contributions:

1. A complete description of the ScopeDOPE point-mass exterior ballistics solver, including derivation of the drag integration formulation, atmospheric model, secondary effects, and numerical methods.
2. Full technical characterization of DOPE-BlendTM as a novel, proprietary field-data fusion framework, including its mathematical formulation, interpolation strategy, and extrapolation policy.
3. A rigorous validation study using exhaustive C(N,k) combinatorial cross-validation against 37 test cases spanning 22 calibers and three accuracy tiers.
4. An analysis of the McCoy G7 drag table calibration requirement and the empirical $1.26\times$ factor that aligns the engine with the JBM Ballistics G7 reference function.

The engine is implemented in Kotlin, targeting the Kotlin Multiplatform shared module `:ballistics`, and compiles to both Android (via AGP) and JVM (for the validation suite). iOS support is planned for the v1.0 release via KMP native targets. All computation is performed on-device; no network calls, no user accounts, and no server-side infrastructure are required or present.

2. Theoretical Background

2.1 Point-Mass Exterior Ballistics

The point-mass model of exterior ballistics treats the projectile as a dimensionless particle subject to gravity and aerodynamic drag, neglecting angular orientation, gyroscopic effects, and Magnus forces in the primary trajectory computation. This simplification is well-established for spin-stabilized projectiles flying at small yaw angles and is the basis for all commercially deployed ballistic calculation systems [McCoy 1999; Litz 2009]. The model’s validity derives from the observation that a well-stabilized, properly designed projectile maintains a yaw angle below 1–2 degrees for most of its supersonic flight, under which condition the aerodynamic drag is dominated by the axial component and the cross-axis terms are negligible.

The equation of motion along the trajectory axis is:

$$m\ddot{x} = -F_D$$

where m is bullet mass and F_D is the aerodynamic drag force:

$$F_D = \frac{1}{2}\rho A v_r^2 C_d(M)$$

Here ρ is air density, A is the bullet’s cross-sectional area, v_r is the velocity of the bullet relative to the surrounding air, and $C_d(M)$ is the drag coefficient at Mach number $M = v_r/c_s$, where c_s is the local speed of sound. Expressing drag deceleration in terms of ballistic coefficient (see §2.3) and the standard-atmosphere drag formula constant (see §4.1) yields the form integrated by the ScopeDOPE engine.

2.2 Reference Drag Functions: G1 and G7

The drag coefficient $C_d(M)$ is a strong function of Mach number and depends on projectile geometry. Rather than requiring per-projectile aerodynamic characterization, the ballistics community defines a set of reference projectiles with standard drag functions, and characterizes actual bullets relative to those references via the ballistic coefficient. Two reference drag functions are in common use:

G1 — Defined by the Krupp flat-base blunt-ogive projectile. The G1 reference is the historical standard and remains the most widely published BC form for older and intermediate bullets, particularly hunting and pistol projectiles.

G7 — Defined by the boat-tail, tangent-ogive, secant-ogive match-grade projectile that more closely resembles modern long-range rifle bullets. G7 BCs require fewer adjustments across the velocity range and more accurately represent the drag of contemporary high-BC projectiles, making G7 the preferred model for precision long-range work [Litz 2009].

The G1 and G7 drag tables used in the ScopeDOPE engine are sourced from McCoy’s *Modern Exterior Ballistics* [McCoy 1999], covering Mach numbers from 0 to 3.0. These tables are in the public domain.

2.3 Ballistic Coefficient

The ballistic coefficient is a scalar measure of a projectile’s resistance to aerodynamic deceleration relative to the reference projectile:

$$BC = \frac{SD}{i}$$

where SD is the sectional density and i is the form factor (ratio of the projectile’s actual drag to the reference drag at a given Mach number). Sectional density is:

$$SD = \frac{W_{\text{gr}}}{7000 \cdot D_{\text{in}}^2}$$

where W_{gr} is bullet weight in grains, D_{in} is diameter in inches, and 7000 is the grains-per-pound conversion. The engine uses this relationship to compute BC from first principles when a measured BC is not provided (see §4.5).

2.4 JBM Ballistics as Industry Reference

JBM Ballistics is the de facto industry reference implementation for exterior ballistics computation, widely used by published BC data sources including Applied Ballistics and Hornady. Published G7 BCs are implicitly calibrated against the JBM G7 drag function, not directly against the McCoy tabulation. As described in §4.2, these two G7 reference functions differ by a systematic factor of approximately 1.26, requiring a calibration step to achieve agreement between a McCoy-based engine and JBM-based published data.

3. Atmospheric Model

3.1 Speed of Sound

The local speed of sound in dry air follows the ISA tropospheric relation:

$$c_s = 1116.45 \sqrt{\frac{T_F + 459.67}{518.67}} \quad [\text{ft/s}]$$

where T_F is the ambient temperature in degrees Fahrenheit. The reference value 518.67°R (Rankine) corresponds to the ISA sea-level standard temperature of 59°F (288.15 K), and 1116.45 ft/s is the ISA standard-atmosphere speed of sound. This formulation is the Fahrenheit equivalent of the ideal-gas result $c_s = \sqrt{\gamma RT/M}$ with $\gamma = 1.4$ for diatomic gases and the standard molar mass of dry air.

3.2 Air Density Ratio

The engine computes a dimensionless air density ratio ρ/ρ_0 relative to the standard atmosphere, which is then applied to scale the effective BC (§3.4) rather than directly modifying the drag coefficient. This formulation keeps BC in familiar manufacturer-supplied units while correctly propagating density effects.

Pressure: When measured station pressure is available, it is used directly. When only altitude is available, the ICAO Standard Atmosphere tropospheric pressure formula [ICAO 1993] applies:

$$P = P_0 (1 - 6.87559 \times 10^{-6} \cdot h)^{5.2561} \quad [\text{hPa}]$$

where $P_0 = 1013.25$ hPa is the standard sea-level pressure and h is altitude in feet. The exponent 5.2561 derives from the ratio of the gravitational acceleration-to-molar-mass-times-gas-constant product over the ISA temperature lapse rate.

Humidity: Humidity reduces air density by displacing heavier diatomic nitrogen and oxygen molecules with lighter water vapor molecules. The saturation vapor pressure is computed via the Magnus formula [Magnus 1844]:

$$e_{\text{sat}} = 6.1078 \exp\left(\frac{17.27 T_C}{T_C + 237.3}\right) \quad [\text{hPa}]$$

where $T_C = (T_F - 32) \times 5/9$ is temperature in degrees Celsius. Actual vapor pressure and the humidity density correction factor are then:

$$e_W = \frac{\text{RH}}{100} \cdot e_{\text{sat}}$$

$$\rho_{\text{factor}} = 1 - \frac{0.378 e_W}{P}$$

The coefficient 0.378 arises from the ratio of water-to-dry-air molecular weights: $1 - M_{H_2O}/M_{\text{air}} \approx 0.378$.

Full density ratio:

$$\frac{\rho}{\rho_0} = \frac{P}{P_0} \cdot \frac{T_{0,K}}{T_K} \cdot \rho_{\text{factor}}$$

where $T_{0,K} = 288.706$ K and $T_K = (T_F + 459.67) \times 5/9$ are the standard and actual temperatures in Kelvin respectively.

3.3 Atmospheric Sensitivity

The density ratio formulation captures the primary sensitivity of bullet trajectory to environmental conditions. At 5000 ft elevation and 100°F, the density ratio drops to approximately 0.80, increasing the effective BC by 25% and reducing drop by a corresponding amount. This magnitude is large enough to shift a 1000-yard hold by 1–2 mil on many loads, underscoring the importance of accurate atmospheric modeling.

3.4 Effective BC Scaling

The effective BC for integration at any given condition set is:

$$BC_{\text{eff}} = \frac{BC}{\rho/\rho_0}$$

This scaling is mathematically equivalent to multiplying the drag force by the density ratio, but preserves BC in familiar units for the integrator and produces cleaner numerical behavior. Multi-zone BC tables (where BC varies with velocity) are handled by scaling each zone individually using the same density ratio computed from the prevailing conditions.

4. Physics Model

4.1 Drag Deceleration and the KD Constant

The per-unit-distance velocity deceleration of the projectile, expressed via the chain rule $dv/ds = (dv/dt)/v$, is:

$$\frac{dv}{ds} = -\frac{KD \cdot C_d(M) \cdot v_r^2}{BC_{\text{eff}} \cdot v}$$

where s is distance (feet), v is bullet velocity (fps), $v_r = v + v_{\text{hw}}$ is the relative airspeed accounting for headwind component v_{hw} (positive headwind increases drag), $M = v_r/c_s$ is the Mach number, and KD is the drag formula constant:

$$KD = \frac{\rho_0}{2 \times 144} = \frac{0.0764}{288.0} \approx 2.653 \times 10^{-4} \quad [\text{lb/in}^3 \cdot \text{ft}^{-1}]$$

The factor $\rho_0 = 0.0764 \text{ lb/ft}^3$ is the standard sea-level air density in English units. The denominator 2×144 converts the $144 \text{ in}^2/\text{ft}^2$ unit factor inherent in the BC definition (which uses in^2 for cross-sectional area and lb for weight) into the appropriate form for the fps-grain-foot unit system. This constant is invariant across conditions because non-standard density is handled entirely through BC_{eff} scaling.

4.2 G7 Drag Table Calibration

A systematic discrepancy exists between the McCoy G7 drag table and the G7 drag function implemented by JBM Ballistics, which is the de facto reference against which modern G7 BCs are published. Without correction, a McCoy-based engine using published G7 BCs produces systematically under-predicted drop — the engine sees too little drag for a given BC because the McCoy table implies a slightly lower Cd at each Mach number than the JBM reference function does.

Empirical cross-validation against JBM for loads spanning G7 BC 0.25–0.35 and muzzle velocity 2400–3200 fps demonstrated that scaling the McCoy G7 Cd uniformly by a factor of 1.26 achieves agreement within 0.012 mil at 1000 yards. The calibration is applied exclusively to the G7 drag function:

$$C_{d,G7,\text{eff}}(M) = 1.26 \times C_{d,G7,\text{McCoy}}(M)$$

The G1 drag table is left unscaled. G1 BCs reference a different drag function history and the $1.26\times$ calibration was not validated in the G1 domain; applying it without validation would introduce rather than reduce error for G1 projectiles.

The factor is applied inside the trajectory calculator at the drag function evaluation site, scoped to G7 only:

```
cdFn = if (bcModel == G7) { mach -> G7_CURVE.eval(mach) * 1.26 } else G1_CURVE::eval
```

Figure 1: G7 drag table calibration application point. The $1.26\times$ factor is applied to the Cd function object; no other constant is modified.

4.3 Zeroing Integration

The engine integrates the trajectory twice for each match-day solution. The first integration (the “zero integration”) runs at the conditions prevailing when the rifle was zeroed — potentially a different temperature, altitude, muzzle velocity, and even a different atmospheric pressure than the current shooting session. The zero integration produces the drop at zero range y_Z under zero conditions. The second integration (the “match-day integration”) runs at the current conditions, producing drop at the target distance y_D .

This two-integration architecture correctly captures the environmental offset between zeroing and shooting. A rifle zeroed at sea level in 30°F weather and then shot at 5000 ft elevation in 80°F summer conditions will have a different trajectory-relative-to-zero than either a naive single-integration approach or a fixed-zero-drop assumption would produce. Specifically:

$$\Delta\theta_{\text{env}} = \theta(y_D, \text{match conditions}) - \theta(y_D, \text{zero conditions})$$

This correction is non-trivial and grows with range: for extreme environmental differences (e.g., sea-level winter zero, high-altitude summer conditions), the offset can exceed 0.5 mil at 1000 yards on typical precision rifle loads.

4.4 Trajectory Elevation in Milliradians

The elevation adjustment in milliradians — the quantity reported to the shooter — is derived from the geometric relationship between bullet drop, sight height, and the zero geometry:

$$\theta_{\text{elev}} = \frac{12 \left[y_D + h_s - \frac{d}{d_0}(y_Z + h_s) \right]}{d \times 0.036} \quad [\text{mil}]$$

where y_D is the drop (in feet, positive downward) at distance d yards, y_Z is the drop at zero distance d_0 yards, $h_s = h_{\text{in}}/12$ is the sight height in feet (default 1.5 inches), and the factor $d \times 0.036$ is the angular subtension in inches per milliradian at distance d : at d yards, 1 mrad subtends $d \times 36/1000 = d \times 0.036$ inches. The factor of 12 converts the numerator from feet to inches.

The numerator $y_D + h_s - (d/d_0)(y_Z + h_s)$ is the net elevation difference between the bore line and the sight line at distance d , accounting for the bore-to-sight offset at the zero range. Positive output indicates an upward adjustment (the scope must be dialed up; the bullet is falling below line of sight).

4.5 BC from Form Factor

When measured BC is unavailable but bullet dimensions are known, the engine can compute BC from the Pejsa/sectional density method:

$$BC = \frac{SD}{i} = \frac{W_{\text{gr}}/(7000 \cdot D_{\text{in}}^2)}{i}$$

where i is the form factor provided by the user or inferred from geometry. This path is less accurate than using measured BCs and is provided as a fallback for non-catalogued projectiles.

5. Numerical Methods

5.1 Spatial Domain Formulation

The ScopeDOPE integrator operates in the spatial domain — integrating velocity as a function of distance $v(s)$ — rather than the temporal domain $v(t)$. The spatial and temporal formulations are mathematically equivalent via the chain rule:

$$\frac{dv}{ds} = \frac{dv/dt}{ds/dt} = \frac{dv/dt}{v}$$

The spatial formulation is preferred for three reasons:

1. **Natural output resolution.** The integrator produces velocity (and derived quantities) at uniform spatial intervals, which are the natural units for the ballistic output: elevation and windage at specific distances in yards. A temporal integrator produces values at uniform time intervals, requiring spatial interpolation to recover per-yard values.
2. **Adaptive step accuracy.** Near the muzzle where velocity is highest, a spatial step $\Delta s = 0.5$ yd corresponds to a small time step (high temporal resolution); at extended range where velocity has decreased, the same spatial step corresponds to a larger time step — the right behavior, as the bullet accelerates less at low velocity.
3. **Graceful integration cap.** The integrator terminates at a maximum of 5000 yards regardless of velocity, providing a natural safety bound.

5.2 Fourth-Order Runge-Kutta Integration

The velocity integration uses a classical four-stage Runge-Kutta (RK4) scheme over spatial steps of $h = 1.5$ ft (equivalent to 0.5 yd). Given the drag acceleration function $f(v) = dv/ds$:

$$k_1 = f(v_n)$$

$$\begin{aligned}
k_2 &= f(v_n + \frac{h}{2}k_1) \\
k_3 &= f(v_n + \frac{h}{2}k_2) \\
k_4 &= f(v_n + h k_3) \\
v_{n+1} &= v_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)
\end{aligned}$$

All intermediate velocities are clamped to a minimum of 1 fps to prevent division-by-zero in the drag function evaluation. RK4 has local truncation error $O(h^5)$ and global error $O(h^4)$. At the 0.5-yd step size, the global error contribution from the numerical integrator is negligible relative to the atmospheric model uncertainty and BC measurement uncertainty — the integrator is not the limiting factor in overall accuracy.

5.3 Time-of-Flight Derivation

Time of flight is accumulated per step using the trapezoidal mean velocity approximation:

$$\Delta t = \frac{h}{\bar{v}} = \frac{h}{(v_n + v_{n+1})/2}$$

This is first-order accurate in Δv , which is adequate given that the velocity change per 0.5-yd step is small (typically < 1 fps for subsonic and < 5 fps for supersonic conditions) and the accumulated error over 1000 yards is below the sub-millisecond level relevant to ballistic calculations.

5.4 Drop Integration

Vertical drop is integrated per step using the kinematic equation, treating gravity as constant over the step:

$$\begin{aligned}
y_{n+1} &= y_n + v_{\text{drop},n} \cdot \Delta t + \frac{1}{2}g_{\text{eff}} \cdot \Delta t^2 \\
v_{\text{drop},n+1} &= v_{\text{drop},n} + g_{\text{eff}} \cdot \Delta t
\end{aligned}$$

where $g_{\text{eff}} = g \cos \theta_{\text{fire}}$ is the effective vertical gravity component for inclined fire (see §6.4). This Euler-type integration of the drop is accurate because drop velocity changes slowly relative to the step size.

5.5 Fritsch-Carlson Monotone Cubic Hermite Interpolation

The ScopeDOPE engine uses Fritsch-Carlson monotone cubic Hermite interpolation [Fritsch & Carlson 1980] for two distinct purposes: evaluating drag coefficient at arbitrary Mach numbers from the tabulated G1/G7 data, and interpolating DOPE-Blend™ bias residuals between observed training points (see §7).

The Fritsch-Carlson algorithm constructs a piecewise cubic Hermite interpolant whose tangent vectors m_i at the knot points are chosen to guarantee that the interpolant is monotone on any interval where the data is monotone. This is achieved by:

1. Computing chord slopes $\delta_i = (y_{i+1} - y_i)/(x_{i+1} - x_i)$ between adjacent knots.

2. Initializing tangents as the arithmetic mean of neighboring chords at interior points: $m_i = (\delta_{i-1} + \delta_i)/2$.
3. Setting $m_i = 0$ at any sign change in adjacent chords, ensuring monotonicity is preserved at local extrema.
4. Applying the Fritsch-Carlson monotonicity condition: for each interval where $|\alpha_i^2 + \beta_i^2| > 9$ (where $\alpha_i = m_i/\delta_i$ and $\beta_i = m_{i+1}/\delta_i$), scaling the tangent vectors by $\tau = 3/\sqrt{\alpha_i^2 + \beta_i^2}$ to pull them inside the monotonicity region.

The Hermite basis functions then evaluate the interpolant at any point within the spanned range:

$$p(t) = (2t^3 - 3t^2 + 1)y_i + (t^3 - 2t^2 + t)h m_i + (-2t^3 + 3t^2)y_{i+1} + (t^3 - t^2)h m_{i+1}$$

where $t = (x - x_i)/(x_{i+1} - x_i)$ and $h = x_{i+1} - x_i$.

The monotonicity guarantee is critical in both applications. For the drag curve, it prevents unphysical Cd oscillations between tabulated Mach breakpoints that would produce nonphysical velocity oscillations under numerical integration. For DOPE-Blend, it prevents the spline from producing erratic elevation correction overshoots between confirmed field observations. Standard cubic splines (which minimize curvature rather than preserving monotonicity) can produce large oscillations when data is sparse or irregularly spaced, making them unsuitable for either use.

6. Secondary Effects

6.1 Windage: Didion Lag Rule

The engine implements the Didion lag rule [Didion 1860] for lateral wind deflection rather than the simplified $v_{\text{wind}} \times t_{\text{of}}$ approximation. The key insight of the Didion formulation is that a spin-stabilized projectile does not instantly align to track a crosswind change due to gyroscopic precession. Instead, the effective crosswind lag time is the difference between actual time of flight and the vacuum (no-drag) time of flight:

$$\Delta t_{\text{lag}} = t_{\text{of}} - t_{\text{vac}}$$

$$t_{\text{vac}} = \frac{d \times 3}{v_0}$$

$$\theta_{\text{windage}} = \frac{12 \cdot v_{\text{cw}} \cdot \Delta t_{\text{lag}}}{d \times 0.036} \quad [\text{mil}]$$

where v_{cw} is the crosswind component (fps), d is distance in yards, and the factor $d \times 3$ converts yards to feet for the vacuum time calculation. The factor of 12 and the denominator $d \times 0.036$ perform the feet-to-mrad angular conversion as in §4.4.

Wind components are resolved into headwind (along the bore axis) and crosswind (perpendicular) via:

$$v_{\text{hw}} = v_{\text{wind}} \cos(\theta_{\text{wind}})$$

$$v_{cw} = v_{wind} \sin(\theta_{wind})$$

where θ_{wind} is the wind angle from the bore axis (0° = full headwind, 90° = full crosswind, 180° = full tailwind). The headwind component enters the drag calculation as an additive contribution to the relative airspeed; the crosswind component enters only the windage calculation via the Didion formula.

6.2 Spin Drift: Miller Gyroscopic Stability

Spin drift — the transverse deflection of a bullet in the direction of rifling twist due to gyroscopic coupling between the spin axis and the trajectory — is computed in two stages.

Stage 1: Miller Gyroscopic Stability Factor [Miller 1983]:

$$S_g = \frac{30W}{T^2 D^3 L(1 + L^2)} \cdot \left(\frac{v_0}{2800} \right)^{1/3}$$

where W is bullet weight in grains, T is twist rate in inches per revolution, D is bullet diameter in inches, $L = L_{bullet}/D$ is the length-to-diameter ratio, and v_0 is muzzle velocity in fps. The velocity factor $(v_0/2800)^{1/3}$ applies the Greenhill-Miller correction for the effect of muzzle velocity on gyroscopic stability. S_g is clamped to a maximum of 5.0; values of $S_g < 1.0$ indicate dynamic instability (the bullet will tumble) and the spin drift formula is undefined.

Stage 2: Litz Spin Drift Approximation [Litz 2009]:

$$d_{drift} = 1.25 \cdot S_g \cdot t_{of}^{1.83} \quad [\text{inches}]$$

$$\theta_{drift} = \frac{d_{drift}}{d \times 0.036} \quad [\text{mil}]$$

The direction of drift is toward the direction of rifling twist (positive = right for right-hand twist; negative = left for left-hand twist). The exponent 1.83 is an empirical fit by Litz to observed spin drift data across a range of projectiles and ranges; it captures the growing drift rate as time of flight accumulates.

6.3 Coriolis Effect

For long-range shooting at known latitude and azimuth, the Coriolis acceleration introduces both a horizontal (windage) deflection and a vertical (elevation) deflection. The engine integrates both components per 0.5-yd step using the mean velocity of the step.

The per-step accelerations are:

$$a_{cw,C} = 2\Omega \sin(\varphi) \cdot \bar{v}$$

$$a_{elev,C} = 2\Omega \cos(\varphi) \sin(\alpha) \cdot \bar{v}$$

where $\Omega = 7.2921 \times 10^{-5}$ rad/s is Earth's sidereal rotation rate, φ is the geodetic latitude, α is the firing azimuth, and $\bar{v} = (v_n + v_{n+1})/2$ is the step-mean bullet velocity.

In the Northern Hemisphere, $a_{cw,C}$ deflects the bullet to the right (positive east when firing north). The elevation component $a_{elev,C}$ is azimuth-dependent — it causes uplift when firing east and downward deflection when firing west. Both accumulated displacements are converted to milliradians at the final step.

Coriolis effects are non-negligible at 1000+ yards competitive ranges: at 45°N latitude firing due east at 1000 yards, the horizontal Coriolis deflection is approximately 0.02 mil and the vertical deflection is approximately 0.01 mil.

6.4 Inclined Fire

For targets at non-zero elevation angles, the effective gravitational acceleration acting to pull the bullet below the line of sight is reduced by the cosine of the firing angle — the so-called “rifleman’s rule”:

$$g_{\text{eff}} = g \cos(\theta_{\text{fire}})$$

This is applied to the drop integration only. The result is expressed in slant range milliradians rather than horizontal range, consistent with how the scope is indexed to the actual slant distance to the target. The rifleman’s rule is an approximation valid for angles below approximately 45°; it slightly under-corrects at higher angles, though such extreme angles are uncommon in precision rifle applications.

7. DOPE-Blend™ Field-Data Fusion Framework

7.1 Motivation and Conceptual Framework

The physics model described in §4–6 is an accurate and validated trajectory predictor under ideal conditions: manufacturer-supplied BCs that precisely match the actual bullet, perfectly known atmospheric conditions, and no barrel-specific aerodynamic effects. In practice, none of these conditions hold exactly. Published G7 BCs are averages across production lots and test conditions. Atmospheric conditions at the muzzle and downrange may differ from those measured at the firing point. Barrel-specific twist, rifling engagement depth, throat erosion, and in-bore gas dynamics all produce velocity and aerodynamic signatures that cannot be captured from nominal specifications alone.

The conventional response — a user-adjustable BC input — treats all sources of error as equivalent to a BC error, which they are not. A 1% BC measurement error produces a smoothly growing error that is the same direction at all distances. Atmospheric measurement error produces a different spatial signature. Barrel-specific aerodynamic peculiarities produce yet another. Adjusting a single BC scalar to minimize error at one distance guarantees residual error at others; the adjustment that is correct at 500 yards may be incorrect at 1000 yards.

DOPE-Blend™ addresses this by working directly in the physics model’s output space — milliradians of elevation — and computing residuals between what the physics model predicts and what the shooter actually observed. These residuals are interpolated between confirmed data points and applied to the physics model’s match-day prediction. Critically, the atmospheric re-projection is handled by the physics engine; DOPE-Blend corrects only for the systematic error the physics engine cannot account for.

7.2 Bias Residual Computation

For each field DOPE entry — a tuple of (distance, observed elevation, temperature, altitude, pressure, humidity, muzzle velocity) — the engine computes the bias residual:

$$\varepsilon_j = \theta_{\text{obs},j} - \hat{\theta}_{\text{model},j}$$

where $\theta_{\text{obs},j}$ is the observed elevation in milliradians at distance d_j (the hold the shooter confirmed was correct), and $\hat{\theta}_{\text{model},j}$ is the physics model’s prediction at the same distance and the conditions recorded at the time of the observation. Both quantities are in the same units and the same reference frame, so the residual ε_j is a pure elevation correction in milliradians.

This formulation is robust to environmental changes between the observation session and the current session: the residual ε_j captures only the systematic error that is invariant to environmental conditions, not the environmental contribution itself. If the shooter recorded DOPE at sea level in cold weather and is now shooting at altitude in summer, the physics engine handles the altitude and temperature change; DOPE-Blend provides the correction for the shooter’s specific bullet, barrel, and ammunition lot that the physics engine cannot derive from specifications alone.

7.3 Bias Interpolation

Let the N field DOPE entries, sorted by distance, produce pairs (d_j, ε_j) for $j = 1, \dots, N$. The DOPE-Blend corrected elevation at a target distance d^* is:

$$\hat{\theta}_{\text{DOPE-Blend}}(d^*) = \hat{\theta}_{\text{physics}}(d^*) + \hat{\varepsilon}(d^*)$$

where $\hat{\theta}_{\text{physics}}(d^*)$ is the match-day physics prediction and $\hat{\varepsilon}(d^*)$ is the interpolated bias. The interpolation strategy depends on the available data:

Case 1 — Exact match: If $d^* = d_j$ for some j , return ε_j directly. No interpolation is needed; the exact observed correction applies.

Case 2 — Interior interpolation, $N \geq 3$: For $d_1 < d^* < d_N$ with at least three training points, the DOPE-Blend spline is evaluated using Fritsch-Carlson monotone cubic Hermite interpolation over all N knots (d_j, ε_j) . The monotonicity guarantee is essential: without it, cubic splines can produce large oscillatory artifacts between training points, yielding elevation corrections that overshoot or undershoot the physics model in physically implausible ways. The Fritsch-Carlson algorithm prevents this by constraining the interpolant to be monotone on any monotone sub-interval of the data.

Case 3 — Interior interpolation, $N = 2$: With exactly two training points, linear interpolation applies:

$$\hat{\varepsilon}(d^*) = \varepsilon_1 + \frac{d^* - d_1}{d_2 - d_1}(\varepsilon_2 - \varepsilon_1)$$

Case 4 — Extrapolation: For $d^* < d_1$ or $d^* > d_N$, the nearest-end bias value is applied as a constant:

$$\hat{\varepsilon}(d^*) = \begin{cases} \varepsilon_1 & d^* < d_1 \\ \varepsilon_N & d^* > d_N \end{cases}$$

Constant extrapolation is deliberately conservative. Unconstrained spline extrapolation (continuing the polynomial beyond the last knot) diverges rapidly and produces large incorrect corrections far outside the training range. Constant extrapolation carries the nearest confirmed correction as a best estimate while signaling through downstream uncertainty quantification that the estimate is less reliable.

7.4 Windage Exclusion Rationale

DOPE-Blend applies bias corrections only to the elevation (vertical) channel. Crosswind corrections are excluded because crosswind observations in field DOPE records are confounded with unknown wind conditions at the time of fire. A shooter who applies 0.3 mil of windage at 1000 yards may have done so because of a systematic aerodynamic bias — or because there was a 3 mph crosswind at the time. The two are indistinguishable from the record alone. Applying a DOPE-Blend crosswind correction would attribute all windage to a systematic effect, producing erroneous results on days with different wind. Elevation observations do not suffer this confound: gravity is constant, and elevation DOPE recorded in no-wind or corrected-for-wind conditions reflects a reliable systematic signal.

7.5 Novelty and Differentiation

DOPE-Blend™ is, to the knowledge of the authors, a novel contribution to the field of applied ballistics computation. Existing ballistic calculator approaches — including BC fitting, scope zero reset, and empirical hold tables — address pieces of the problem but share a fundamental limitation: they do not separate the environmental re-projection problem from the systematic correction problem.

BC fitting modifies the model globally so that its prediction matches field observations, but this conflates BC uncertainty with atmospheric measurement error and barrel behavior. It also produces incorrect results when conditions change significantly from the calibration session: a BC fitted at sea level in cold weather over-corrects at altitude in summer because the atmospheric compensation is baked into the BC scalar.

Empirical hold tables record raw holds without physics-domain normalization. They are valid only at the exact conditions under which they were recorded and provide no basis for environmental re-projection.

DOPE-Blend takes a fundamentally different approach: it computes residuals in the physics model’s output space under the conditions at time of observation, then applies those residuals on top of a fresh physics integration at the current conditions. This decomposition correctly separates the two sources of prediction error and allows each to be handled by the mechanism best suited to it: the physics model handles environmental re-projection; the spline handles systematic correction. The result is a trajectory prediction that is simultaneously physics-grounded and empirically corrected — achieving accuracy beyond what either approach alone can deliver.

8. Implementation Architecture

8.1 Kotlin Multiplatform Structure

The ScopeDOPE ballistics engine is implemented as a Kotlin Multiplatform (KMP) module (`:ballistics`) within the `scopedope-android` repository. The module's `commonMain` source set contains all ballistics logic in pure Kotlin with no platform dependencies, enabling compilation to:

- **Android:** via the Android Gradle Plugin (AGP) as a `.aar` library
- **JVM (validation):** as a standard JAR for the Kotlin/JVM test suite
- **iOS (deferred):** via KMP native targets at the v1.0 release

The module exposes a Kotlin `object` (singleton) — `BallisticsEngine` — whose public interface consists of stateless pure functions. There is no mutable engine state; every function takes all required inputs as parameters and returns outputs without side effects. This design makes the engine trivially testable (no dependency injection or mocking required), thread-safe, and reentrant.

8.2 Visibility and Trade-Secret Encapsulation

The three functions constituting the DOPE-Blend™ implementation — `entryBias`, `applyBias`, and `interpolateDopeAll` — are marked with restricted Kotlin visibility:

- `entryBias` and `applyBias` are `internal`, accessible only within the `:ballistics` module and its tests. They are not part of the public API and cannot be accessed by the application layer.
- `interpolateDopeAll` is the public entry point for DOPE-Blend, but its internal operation is not observable from the outside.

This visibility strategy forms part of the trade-secret protection for the DOPE-Blend™ methodology: the algorithm is not exposed as a composable API that could be independently replicated from observed inputs and outputs.

The `BcModel` and `TwistDirection` types are defined within the `ballistics` package with typealiases at the application layer, ensuring that the engine API does not depend on application-specific type hierarchies.

8.3 Offline-First Design

The engine has no network access, no cloud dependencies, and no user account requirements. All computation is performed synchronously on the calling thread; the engine functions are appropriate for call from a coroutine dispatcher but do not themselves launch coroutines. This design is required by the product's core promise: ScopeDOPE must function in any environment, including remote locations without cell coverage, and must never depend on a server that could be unavailable in the field.

8.4 Numerical Precision Choices

All computation uses `Double` (64-bit IEEE 754 floating point), providing 15–16 significant decimal digits of precision. For the drag integration, the physically meaningful outputs are in the range of 1–20 milliradians, and the targets precision goal is ± 0.001 mil relative to the reference; this requires approximately 4 significant figures in the output, far within the double-precision margin.

Minimum velocity clamping (`coerceAtLeast(1.0)`) prevents division-by-zero in the drag function evaluation and the time-of-flight computation. The 1 fps lower bound is below any physically meaningful projectile velocity in the application domain and does not affect accuracy for supersonic or even subsonic loads.

8.5 Integrator Step Size Selection

The 0.5-yard (1.5-foot) spatial step size was selected to balance integration accuracy against computational cost on mobile hardware. A sensitivity study showed that halving the step to 0.25 yd changes the 1000-yard elevation result by less than 0.0001 mil for typical precision loads — well below the 0.001-mil target floor. Doubling the step to 1.0 yd introduces errors of approximately 0.001–0.003 mil at 1000 yards, approaching the target floor. The 0.5-yd step is comfortably above the accuracy threshold with a $3\times$ computational margin.

9. Validation Study

9.1 Test Suite Architecture

The validation suite (`BallisticsValidationTest.kt`) runs via the Kotlin/JVM gradle task `./gradlew :ballistics:jvmTest` against the `:ballistics` KMP module directly. Test data is located in `scopedope-shared/ballistics-test-data/test_cases/`, with each test case stored as a structured JSON file containing input parameters, field DOPE entries, expected outputs, atmospheric conditions, and metadata.

The test corpus comprises 37 cases across 22 calibers:

Table 1: Validation Test Corpus Composition

Category	Count	Description
Sub-Tier-3 (cross-calculator)	11	G7 precision loads, engine vs. JBM reference
Sub-Tier-3 (additional)	19	Additional calibers/loads
Tier 3 (confirmed fired)	7	Verified physical impact data with atmospheric records
Real-world (field DOPE)	15	Multi-session field DOPE with confirmed holds
Total	37	Spanning calibers from .17 HMR to .338 Lapua Magnum

9.2 Accuracy Tiers and Thresholds

Test cases are assigned to one of three accuracy tiers based on the quality and nature of the reference data:

Table 2: Accuracy Tier Definitions and Thresholds

Tier	Label	Reference Source	OK	INVESTIGATE FAIL	
Sub-3	Cross-calculator	JBM Ballistics computed output	< 0.025 mil	0.025–0.05 mil	≥ 0.05 mil
Tier 3	Confirmed fired	Verified impact data	< 0.05 mil	0.05–0.10 mil	≥ 0.10 mil
Real-world	Field DOPE	Unverified shooter holds	< 0.10 mil	0.10–0.20 mil	≥ 0.20 mil
Transonic	Any	Any of above	$2\times$ above	$2\times$ above	$2\times$ above

The threshold basis derives from three physical floors:

- **Solver floor (~ 0.003 mil):** The irreducible disagreement between correctly implemented point-mass solvers, measured from JBM vs. Applied Ballistics comparison on precision loads. This represents the limit of solver-to-solver comparison validity.
- **Hardware floor (~ 0.05 mil):** A standard 0.1 MRAD adjustment increment is common on field rifle scopes; competition scopes are typically 0.05 MRAD. Errors below one click are operationally invisible.
- **Physics floor (~ 0.097 mil):** Empirical vertical spread of top-tier F-Class groups at 1000 yards, approximately 1/3 MOA. This represents the limit of accuracy resolvable from field data.

The Sub-3 threshold is set well below the hardware floor because cross-calculator comparison should be limited only by the solver floor. Tier 3 and Real-world thresholds allow progressively more error, consistent with the progressively lower quality of the reference data.

9.3 Combinatorial Cross-Validation Methodology

Rather than reporting simple aggregate accuracy, the validation suite applies exhaustive combinatorial leave- k -out (LKO) cross-validation. For each test case with $N \geq 4$ usable field DOPE entries:

1. Precompute the physics trajectory and bias residuals ε_j for all N DOPE entries.
2. Enumerate all $\binom{N}{k}$ combinations of training entries for $k = 1, \dots, N-3$ (preserving a minimum of 3 training entries).
3. For each combination, treat the remaining entries as holdout points.
4. Evaluate the DOPE-Blend spline using only the training entries and compute the error at each holdout.
5. Classify each holdout as *interior* (the training set brackets the holdout distance) or *extrapolation* (the holdout is outside the training range).
6. Record the worst-case holdout error across all combinations at each k , and separately for the primary verdict ($k = 1$, standard leave-one-out).

This methodology provides substantially more information than simple train/test accuracy:

- The **primary verdict** ($k = 1$ LOO, interior holdouts) represents the standard expected case: the shooter has training data on both sides of any prediction point.

- The **stress-test verdict** (worst case across all k) reveals the degradation behavior under data sparsity — how badly the system fails if the shooter has recorded only a few distant DOPE entries.
- The interior/extrapolation classification reveals whether errors are due to interpolation (a solver property) or extrapolation (expected to be less accurate).

Cases with fewer than 4 DOPE entries produce a L00_SKIP result and are excluded from the primary and stress-test verdicts.

10. Results

10.1 Primary Validation Verdict

Table 3: Overall Validation Results — Run 2026-05-31

Metric	Value
Total test cases	37
Skipped (< 4 DOPE points, L00_SKIP)	1
Primary verdict (k=1 interior LOO)	INVESTIGATE
Stress-test verdict (all k, worst case)	FAIL
Max absolute error, Tier 3 fired interior	0.459 mil @ 7mm PRC 180gr ELD-M, 1000yd
Max absolute error, Sub-3 all k	1.525 mil @ .22 LR Eley Match, 200yd

The INVESTIGATE primary verdict indicates that the worst-case $k = 1$ interior LOO error exceeds 0.025 mil but remains below 0.05 mil for Sub-3 cases. This is the expected operating regime: the solver is accurate to within the hardware floor for all well-characterized supersonic loads; errors approaching INVESTIGATE status occur only under data configurations that are unusual in practice.

The FAIL stress-test verdict reflects the documented graceful-degradation limit: under extreme data sparsity (many entries held out simultaneously), the constant-extrapolation fallback produces large errors when the training set no longer brackets the prediction. This is an expected and acceptable behavior, not a solver deficiency: constant extrapolation is the correct conservative policy.

10.2 Cross-Calculator Accuracy: G7 Precision Loads

The primary use case of ScopeDOPE — precision long-range shooting with modern G7 boat-tail projectiles — is served by the 11 Sub-Tier-3 cross-calculator test cases. These represent the engine’s accuracy against JBM Ballistics as the industry reference.

Table 4: Representative G7 Cross-Calculator Results vs. JBM Ballistics

Load	Range	Engine (mil)	JBM (mil)	$\ \Delta\ $	Status
.308 Win 168gr SMK, G7 BC 0.243	1000 yd	12.705	12.705	0.0002	OK

Load	Range	Engine (mil)	JBM (mil)	$\ \Delta\ $	Status
.308 Win 175gr SMK, G7 BC 0.264	1000 yd	10.984	10.984	0.0003	OK
6.5 Creedmoor 140gr ELD-M, G7 BC 0.287	1000 yd	9.221	9.221	0.0004	OK
6.5 Creedmoor 140gr Hybrid, G7 BC 0.315	1000 yd	8.568	8.568	0.0003	OK
.300 Win Mag 190gr SMK, G7 BC 0.337	1500 yd	18.630	18.630	0.0003	OK
.300 Win Mag 200gr ELD-X, G7 BC 0.352	1500 yd	17.398	17.399	0.0003	OK
6mm Creedmoor 108gr Berger Hybrid, G7 BC 0.273	1000 yd	10.231	10.231	0.0001	OK
6.5 PRC 147gr ELD-M, G7 BC 0.315	1000 yd	7.943	7.943	0.0003	OK

Errors are uniformly in the sub-0.001 mil range for the precision G7 load set, three orders of magnitude below the 0.025-mil Sub-3 FAIL threshold. This confirms that the $1.26\times$ McCoy G7 calibration factor achieves the goal of aligning the engine with the JBM reference function.

10.3 JBM Cross-Reference: Tier 3 Confirmed-Fired Case

The most demanding validation category is Tier 3: cases where both the engine result and JBM result can be compared against confirmed physical impacts at known atmospheric conditions. The following table shows the complete per-range comparison for the 6.5 Creedmoor 140gr A-MAX case:

Table 5: JBM Cross-Reference — 6.5 Creedmoor 140gr A-MAX, Tier 3 (Confirmed Fired)

Dist (yd)	Engine (mil)	JBM (mil)	Δ Eng–JBM	Fired (mil)	Δ Eng–Fired
300	0.713	0.713	−0.0001	0.727	−0.014
500	2.431	2.428	+0.003	2.473	−0.042
700	4.519	4.516	+0.003	4.509	+0.010
900	7.054	7.049	+0.005	7.054	0.000
1000	8.531	8.519	+0.012	8.581	−0.050

The engine-to-JBM divergence grows from sub-0.001 mil at 300 yards to 0.012 mil at 1000 yards. This divergence is flagged in the validation report at the 0.01-mil DIVERGENCE threshold but remains well within the 0.05-mil hardware floor. The engine-to-fired deltas track the JBM-to-fired pattern closely, indicating that the residual error is attributable to input data uncertainty (atmospheric conditions not perfectly recorded at the time of fire) rather than engine implementation.

This pattern is characteristic of properly functioning DOPE-Blend utilization: the physics-only prediction is close but not perfect; DOPE-Blend applied to the confirmed fired holds at 300 and 500 yards would produce a corrected 1000-yard prediction well within the 0.05-mil threshold.

10.4 Known Open Issues

1. .338 Lapua Magnum divergence. The .338 Lapua (G7 BC 0.342, MV 2969 fps) shows engine-vs-JBM divergence growing from 0.005 mil at 200 yards to 0.110 mil at 1000 yards. This is outside the 0.05-mil Tier-3 threshold and is under investigation. Both engine and JBM show the same direction of miss versus fired data, suggesting the divergence is partly attributable to uncertain input conditions (the atmospheric records for the fired case are incomplete). The high BC of this projectile pushes toward the upper boundary of the validated $1.26\times$ calibration range; a load-specific calibration check against JBM at full atmospheric documentation is planned.

2. Subsonic .22 LR extrapolation. The .22 LR Eley Match case at 200 yards produces a 1.525-mil stress-test error when only near-muzzle training data is available. This is not an engine deficiency; it reflects the fundamental physics of subsonic projectiles, which have rapidly varying Cd in the transonic regime and high sensitivity to small atmospheric differences. The constant-extrapolation policy performs correctly under these conditions (it does not produce incorrect corrections), but the physics model alone cannot predict the subsonic trajectory accurately for such a specialized load from sparse data.

11. Discussion

11.1 G7 Calibration Scope and Generalizability

The $1.26\times$ McCoy-to-JBM G7 calibration factor was derived and validated for the precision rifle domain: G7 BC 0.25–0.35, muzzle velocity 2400–3200 fps. This range covers the vast majority of precision rifle loads including 6mm, 6.5mm, and .308 loads in common competition use. However, the calibration’s validity outside this range has not been verified.

High-BC loads — such as the .50 BMG 750gr A-MAX (G7 BC 0.581) — show a consistent ~0.1-mil low prediction relative to fired data at 1000 yards. This is plausibly a calibration-range effect: the McCoy-JBM G7 function ratio may not be constant across the full Mach-BC space, and the $1.26\times$ factor, derived at lower BCs, may not fully correct for high-BC projectile behavior. Future work should characterize the McCoy-JBM ratio at higher BC and velocity ranges to determine whether a range-dependent calibration factor is warranted.

11.2 Transonic Regime Limitations

All point-mass models using G1 or G7 reference drag functions degrade in the transonic regime (approximately Mach 1.0–1.2). This region is characterized by a rapid, nonlinear increase in Cd

and large sensitivity to small velocity changes. The G1 and G7 tables represent idealized reference projectiles at these speeds; actual bullets of different nose and boat-tail geometry can have significantly different transonic Cd profiles.

Practical impact: for precision rifle loads that remain supersonic to the target (which is the large majority of precision shooting applications), transonic degradation is not a concern. For extended-range .308 Win shots beyond approximately 800 yards, the bullet may enter the transonic zone near the target, and the validation suite applies $2\times$ thresholds accordingly. The .17 HMR and .22 WMR calibers go transonic within 275–350 yards under standard conditions and show the expected accuracy degradation there.

The validation suite correctly flags and quarantines transonic cases with relaxed thresholds rather than reporting them as standard-accuracy results.

11.3 Comparison to Calculator-Class Approaches

Ballistic calculators in the current market share a common architecture: a physics solver with manufacturer-supplied inputs, presenting the user with a flat-range hold table or scope dial. Several offer environmental adjustment (altitude, temperature, wind) but none implement a field-data fusion layer equivalent to DOPE-Blend™. The closest analogs are BC-fitting workflows offered by some platforms, which suffer from the conflation problem described in §7.1.

The ScopeDOPE Ballistic Computer architecture addresses all three components of the trajectory prediction problem simultaneously: accurate physics integration (validated to <0.001 mil vs. JBM for the primary use case), correct environmental modeling including separate zero integration (capturing the zero-day/match-day environmental delta), and DOPE-Blend™ systematic correction (correcting for the irreducible error between physics prediction and actual bullet behavior). No existing ballistic calculator addresses all three.

11.4 Field Deployment Considerations

DOPE-Blend™ is most effective when the training data meets three conditions: (1) multiple entries spanning the shooter’s expected engagement range; (2) atmospheric conditions recorded at time of observation; (3) confirmed actual holds (not estimated or scope-value holds). Under these conditions, DOPE-Blend™ produces corrections that can reduce the engine-to-fired error from the 0.05–0.15 mil typical of physics-only prediction to below 0.02 mil across the trained range.

When training data is sparse or the target distance falls outside the trained range, DOPE-Blend correctly falls back to constant extrapolation, preserving the physics model’s accuracy floor. Users should be informed that holds beyond the DOPE-Blend training range carry increasing uncertainty, and should prioritize confirming holds at the maximum engagement distance before relying on DOPE-Blend corrections at that range.

12. Conclusion

This paper has presented the complete technical architecture of the ScopeDOPE ballistics engine: a validated point-mass exterior ballistics solver with an integrated proprietary field-data fusion system. The key contributions are:

Physics solver: A spatial-domain RK4 integrator with Fritsch-Carlson monotone cubic interpolation of McCoy G1 and G7 drag tables, calibrated with a $1.26\times$ G7 scaling factor to align with the JBM Ballistics G7 reference function. The calibrated engine achieves sub-0.001 mil agreement with JBM for the primary precision rifle load domain (G7 BC 0.25–0.35, MV 2400–3200 fps). The atmospheric model correctly handles altitude, temperature, humidity, and the environmental offset between zeroing and match-day conditions.

Secondary effects: The Didion lag-rule windage formulation, Miller gyroscopic stability and Litz spin drift approximation, per-step Coriolis integration, and the rifleman’s rule for inclined fire collectively address all operationally significant secondary effects in precision long-range shooting.

DOPE-Blend™: A novel, proprietary field-data fusion framework that separates the trajectory prediction problem into an environmental re-projection component (handled by the physics engine) and a systematic correction component (handled by a Fritsch-Carlson monotone cubic spline over bias residuals computed in the physics model’s output space). DOPE-Blend™ provides corrections that no physics model alone can deliver — accounting for BC manufacturing tolerances, barrel-specific aerodynamic behavior, and other systematic deviations from the idealized point-mass model — while remaining fully adaptive to changing atmospheric conditions. It is the central differentiating technology of ScopeDOPE and the primary reason ScopeDOPE functions as a Ballistic Computer rather than a ballistic calculator.

Validation: Exhaustive combinatorial $C(N,k)$ cross-validation across 37 cases and 22 calibers confirms that DOPE-Blend achieves INVESTIGATE-class primary accuracy (worst $k = 1$ interior LOO error below the 0.05-mil hardware floor) with graceful degradation under data sparsity, and that the physics solver achieves sub-0.001 mil accuracy on the precision rifle use case.

Open items include resolving the .338 Lapua divergence, extending the G7 calibration characterization to higher BC ranges, and the planned iOS KMP target compilation for the v1.0 simultaneous launch.

References

- McCoy, R. L. (1999). *Modern Exterior Ballistics: The Launch and Flight Dynamics of Symmetric Projectiles*. Schiffer Publishing.
- Litz, B. (2009). *Applied Ballistics for Long Range Shooting*. Applied Ballistics LLC.
- Miller, D. G. (1983). Graphical determination of gyroscopic stability for spin-stabilized projectiles. *Propellants, Explosives, Pyrotechnics*, 8(2), 59–62.
- Fritsch, F. N., & Carlson, R. E. (1980). Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2), 238–246.
- International Civil Aviation Organization (ICAO). (1993). *Manual of the ICAO Standard Atmosphere* (3rd ed.). Doc 7488.
- Magnus, G. (1844). On the deviation of projectiles, and a remarkable phenomenon of rotating bodies. *Annalen der Physik*, 164(1), 1–28.
- Didion, I. (1860). *Traité de balistique*. Mallet-Bachelier, Paris.

Siacci, F. (1880). Balistique extérieure. *Rivista di Artiglieria e Genio*. [Foundational work on the point-mass drag integration formulation.]

Copyright © 2026 Jason M. Schwefel / Cold Bore Ballistics LLC. All Rights Reserved.

DOPE-Blend™ is a proprietary trademark of Cold Bore Ballistics LLC.

ScopeDOPE is a trademark of Cold Bore Ballistics LLC.