FFI Norwegian Defence Research Establishment

21/02284

FFI-RAPPORT

Algorithms for analysis of ballistic radar data

Øyvind Grandum

Algorithms for analysis of ballistic radar data

Øyvind Grandum

Norwegian Defence Research Establishment (FFI) 14 December 2021

Keywords
Radar
Ballistikk
Simulering

FFI report
21/02284

Project number
1603

Electronic ISBN
978-82-464-3364-6

Approvers
Hege Jødahl, *Research Manager*
Arne Petter Bartholsen, *Director of Research*

The document is electronically approved and therefore has no handwritten signature.

Copyright

Summary

This report describes the algorithms used for processing ballistic radar data at the Norwegian Defence Research Establishment (FFI). A ballistic radar measures the velocity history of projectiles. This is used to determine the drag coefficient (CD) of the projectile, which in turn is used in ballistic trajectory simulations.

The given radar data is primarily *radial* velocity data. These data are first smoothed and parallax corrected. A *tangential* velocity history is then calculated by assuming a ballistic trajectory, given an elevation, wind and possibly secondary aerodynamic forces.

Given the tangential velocity history, together with meteorological information, a drag coefficient (CD) is determined. After smoothing, this results in a CD versus Mach number curve.

The obtained CD function is checked by comparing a simulated radial velocity history with the corresponding measured one.

The report also describes a method used at FFI to determine the ballistics of ricochets from radar measurements and a method for processing spin measurements.

The algorithms are implemented in an in-house code called *Weibelwin* (programmed in C#).

Sammendrag

Denne rapporten dokumenterer algoritmene som brukes for behandling av ballistiske radardata ved Forsvarets forskningsinstitutt (FFI). En ballistisk radar måler hastigheten på prosjektiler. Dette brukes til å bestemme dragkoeffisienten (CD) til prosjektilet, som igjen brukes i ballistiske banesimuleringer.

De gitte radardataene er i første rekke *radiell* hastighet. Disse glattes og parallakse-korrigeres. *Tangentiell* hastighet beregnes deretter ved å anta en ballistisk bane, gitt elevasjon, vind og eventuelle sekundære aerodynamiske koeffisienter.

Gitt den tangentielle hastigheten og meteorologiske data bestemmes dragkoeffisienten (CD). Etter glatting resulterer dette i en kurve over CD som funksjon av machtall.

Den beregnede CD-funksjonen kontrolleres ved å sammenlikne simulert radiell hastighet med tilsvarende målte data.

Rapporten beskriver også en metode brukt ved FFI for å bestemme ballistikken til rikosjetter ut fra radarmålinger samt en metode for å prosessere spinnmålinger.

Algoritmene er implementert i en intern kode kalt *Weibelwin* programmert i C#.

Contents

Symbol list

$\vec{X}_0$	Position of the muzzle exit
$\vec{X}(t)$	Position of the projectile at time t
$\vec{X}_R$	Position of the radar
$\vec{g}$	Acceleration of gravity
$\vec{U}$	Tangential velocity vector of the projectile
$\vec{U}_r$	Radial velocity projectile as seen from the radar
$\vec{W}$	Wind vector (in the direction of blowing)
$\vec{V}$	“Aerodynamic velocity”, $\vec{V} = \vec{U} - \vec{W}$
$\vec{e}_r$	Unity vector in the direction from the radar to the projectile
$\vec{e}_U$	Unity vector along the tangential velocity vector
$\vec{e}_V$	Unity vector along the aerodynamic velocity vector
$\vec{e}_M, \vec{e}_L$	Unity vectors along the Magnus force and lift force, respectively
θ	Trajectory tangent angle with the horizontal plane
φ	Angle between the horizontal and the radial velocity vector
ϕ	Latitude (positive on northern hemisphere)
α	Angle between the horizontal plane and the aerodynamic velocity vector $\vec{V}$
α_R	Yaw of repose (average or equilibrium yaw angle)
β	Angle between $\vec{U}$ and $\vec{V}$
γ	Angle between the launcher and the radar as seen from the projectile
θ_0	Elevation at launch (positive up)
ψ_0	Azimuth at launch (positive to the right)
ρ	Air density
R	Distance between radar and projectile
$\vec{D}, \vec{M}, \vec{L}$	Drag force, Magnus force and lift force vectors, respectively
$C_D,\ C_{mag}, C_{La}$	Coefficients of drag force, Magnus force derivative and lift force derivative
C_{ma}	Coefficient of overturning (or “static pitch”) moment derivative
$\vec{\Lambda}$	Coriolis specific force vector
$\Delta x_L, \Delta y_L, \Delta z_L$	Position of the launcher relative to the radar
p, C_{lp}, C_{ld}	Spin rate (rad/s), Spin damping coefficient, spin driving coefficient
$m,\ d,\ I_x$	Mass, diameter and axial moment of inertia of projectile
S	Reference area, usually the cross-sectional area $S = \dfrac{\pi d^2}{4}$

1 Introduction

A *ballistic radar* measures the trajectory of projectiles by measuring the *Doppler velocity* of the projectile. That is, the radial velocity component as seen from the radar. This can be used to calculate the true velocity history, which in turn can be used to determine the *drag coefficient (CD)*.

The Norwegian armed forces uses a ballistic radar from *Weibel Scientific*. It is a *tracking radar*, that is, the direction of the radar beam is following the projectile, keeping it in the main lobe during the whole trajectory. The tracking radar from *Weibel* also measures angles (elevation and azimuth), but these data are not used in the determining of drag coefficient, since the quality of the data is varying.

This report documents the algorithms used in an in-house code used at FFI, called *Weibelwin*. The purpose of this program is to convert radar measurements into useful ballistic information, primarily the drag coefficient (CD).

2 Overview

The main procedure in the above mentioned FFI program *Weibelwin* is:

1) The input data are the radial velocity measurements versus time from launch. These data points have to been smoothed before further processing. This is done by a “least-square” fitting of the radial data.

2) The true velocity vector is calculated from the radial velocities through a point mass (or *modified* point mass) simulation, given the gravity acceleration, the assumption of ballistic trajectory (or “near-ballistic” trajectory) and the elevation. The wind is also taken into account. The result is the true (tangential) velocity vector.

3) Given the smoothed tangential velocity history, the gravity vector and the assumed/measured meteorological data during test, the drag coefficient (CD) is found from the equations of motion.

4) The CD curve is smoothed if needed, and plotted against e.g. Mach number.

5) The correctness of the resulting CD curve is checked by making a simulation of the measured shot with the resulting CD and using the same weather conditions as in the test. The simulated radial velocity should not differ significantly from the measured ones.

3 Point mass simulation from measured radial velocities – 2D method

What is measured by the radar is the radial velocity as seen from the radar. With the Weibel tracking radar, the elevation and azimuth angles are also measured, but these angles are not used in the present method, other than for control.

In the 2D method, the trajectory is assumed to lie in a vertical plane, *xy*-plane, with the launcher at the origin. Sideways drift (side wind or spin drift) is ignored.

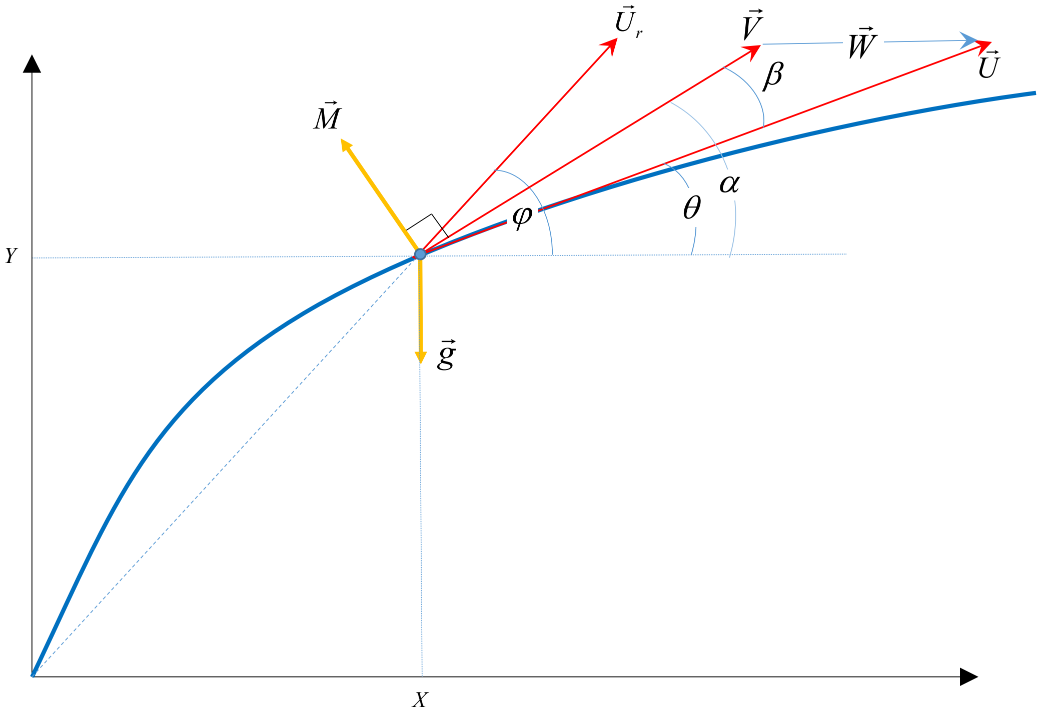


Figure 3.1 Trajectory in 2D with range wind, with forces (yellow) and velocities (red). The launcher is at the origin.

3.1 Parallax correction

In this 2D method, we assume that the measured radial velocity is “parallax corrected”, that is, corrected to the radial velocity as seen from the muzzle (“launcher”). This is done by (3.1):

$$U_r^{corrected} = \frac{U_r^{measured}}{\cos \gamma} \tag{3.1}$$

Here, γ is the angle launcher – projectile – radar. With reference to Figure 3.2, this angle can be found by

$$\sin\gamma = \frac{\left|\vec{r}_L - (\vec{r}_L \cdot \vec{e}_U)\vec{e}_U\right|}{R} , \tag{3.2}$$

where $\vec{r}_L$ is the vector from the radar to the (muzzle of) launcher, $\vec{e}_U$ is a unit vector along line of fire, R is the distance to the projectile from the radar.

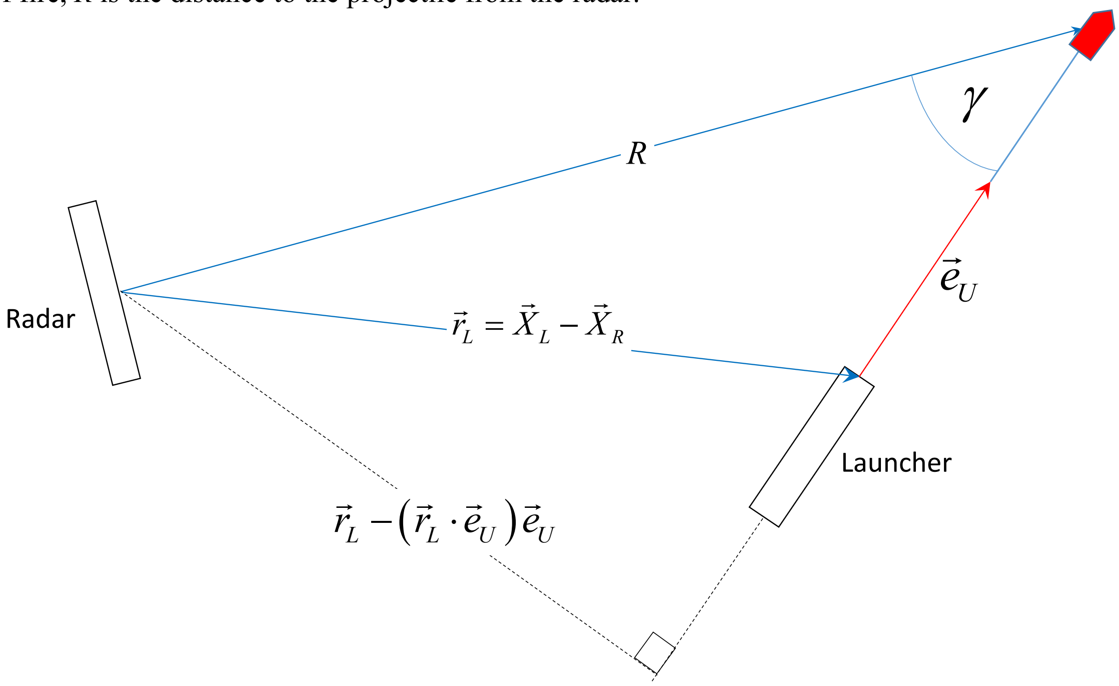

γ
R
Radar
$\vec{e}_U$
$\vec{r}_L = \vec{X}_L - \vec{X}_R$
Launcher
$\vec{r}_L - (\vec{r}_L \cdot \vec{e}_U)\vec{e}_U$

Figure 3.2   Setup for calculating parallax correction

In coordinates where the *y*-axis is the vertical, and the line of fire is given as elevation θ_0 and azimuth ψ_0, we have

$$\vec{e}_U = \begin{bmatrix} \cos\theta_0 \cos\psi_0 \\ \sin\theta_0 \\ \cos\theta_0 \sin\psi_0 \end{bmatrix} \text{ and } \vec{r}_L = \begin{bmatrix} \Delta x_L \\ \Delta y_L \\ \Delta z_L \end{bmatrix} , \tag{3.3}$$

giving

$$\boxed{\begin{aligned} &\sin\gamma = \frac{\sqrt{(\Delta x_L - k\cos\theta_0\cos\psi_0)^2 + (\Delta y_L - k\sin\theta_0)^2 + (\Delta z_L - k\cos\theta_0\sin\psi_0)^2}}{R} \\ &\text{where} \quad k = \Delta x_L \cos\theta_0 \cos\psi_0 + \Delta y_L \sin\theta_0 + \Delta z_L \cos\theta_0 \sin\psi_0 \end{aligned}} \tag{3.4}$$

If the x-axis is along the line of fire ($\psi_0 = 0$), as it usually is, this is reduced to (after some calculations):

$$\sin\gamma = \frac{\sqrt{\left(\Delta x_L \sin\theta_0 - \Delta y_L \cos\theta_0\right)^2 + \left(\Delta z_L\right)^2}}{R}. \tag{3.5}$$

Further simplifications, if the weapon is not elevated ($\theta_0 = 0$):

$$\sin\gamma = \frac{\sqrt{\left(\Delta y_L\right)^2 + \left(\Delta z_L\right)^2}}{R}. \tag{3.6}$$

The simplified situation is shown in Figure 3.3

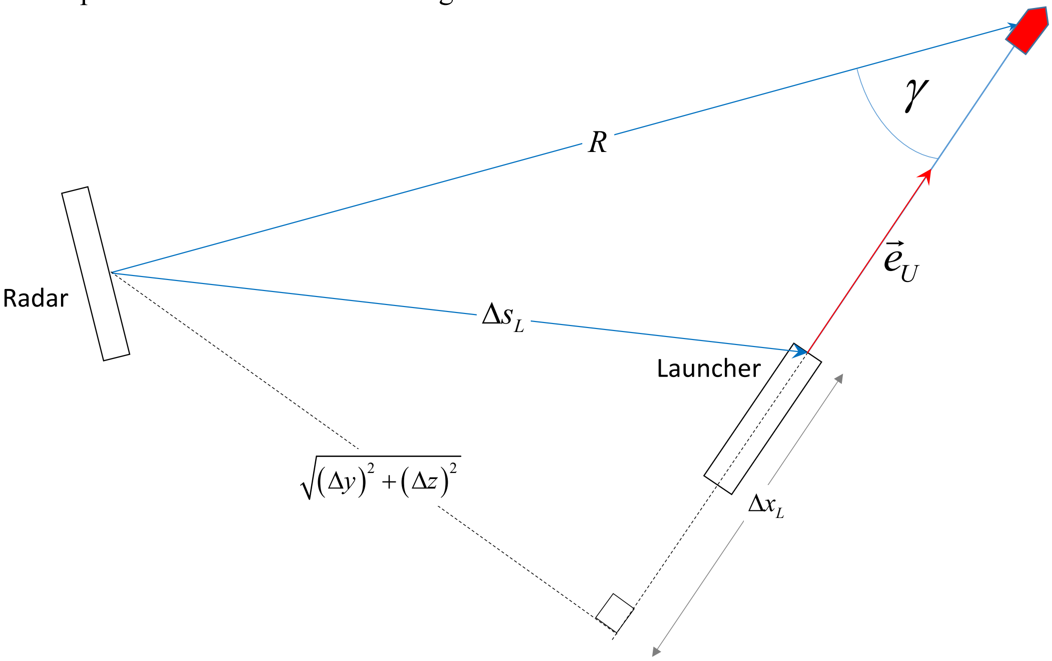


Figure 3.3 Setup for calculating parallax correction in case of horizontal firing and x-axis in line of fire

The distance (“slant range”) R to the projectile from the radar is either measured directly, or integrated by the radar software. Otherwise, it can be integrated from the measured radial velocities. The accuracy of slant range is not crucial for the purpose of parallax correction.

3.2 Equations of motion

The equations of motion are, with reference to Figure 3.1:

$$\begin{cases} \dot{X} = U\cos\theta \\ \dot{Y} = U\sin\theta \\ U = \dfrac{U_r}{\cos(\varphi-\theta)}, \quad \varphi = \arctan\left(\dfrac{Y}{X}\right) \\ \dot{\theta} = f\left(U,\theta,\varphi,\dot{\varphi},\vec{W},....\right) \end{cases} \tag{3.7}$$

with initial conditions

$$\begin{cases} X(0) = Y(0) = 0 \\ \theta(0) = \varphi(0) = \theta_0 \end{cases} \tag{3.8}$$

The main input data is the measured parallax corrected radial velocity U_r.

The challenge in the case of non-zero wind is the determination of the rate of change of elevation $\dot{\theta}$, since the drag force is not parallel with the velocity vector. The Magnus force is also not longer perpendicular to the velocity vector.

Derivation of formulas (3.9) and (3.10), valid with range wind (including vertical wind) in 2D, is shown in Appendix A.1.

$$\dot{\theta} = \frac{\left(M - g\cos(\beta+\theta)\right) + \left(\dfrac{\dot{U}_r}{\cos(\varphi-\theta)} + \dot{\varphi}U\tan(\varphi-\theta)\right)\sin\beta}{U\left(\cos\beta + \sin\beta\tan(\varphi-\theta)\right)} \tag{3.9}$$

where M is the specific Magnus force (force per unit mass) (optional), and β and $\dot{\varphi}$ are given by

$$\begin{cases} \sin\beta = \dfrac{U_y W_x - U_x W_y}{UV} \\ \dot{\varphi} = \dfrac{U\left(X\sin\theta - Y\cos\theta\right)}{X^2+Y^2} \end{cases}. \tag{3.10}$$

In the case of no range wind ($\beta = 0$), the expression (3.9) is reduced to

$$\dot{\theta} = \frac{(M - g\cos\theta)}{U} \quad \text{(no wind)}, \tag{3.11}$$

which can also be deduced directly from Figure 3.4.

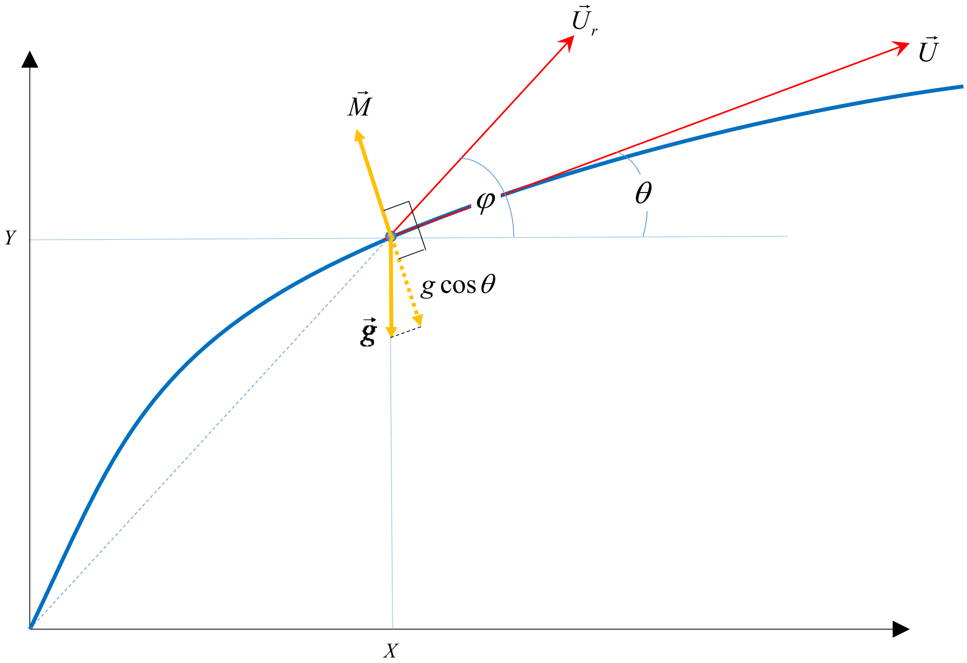


Figure 3.4 Trajectory in 2D without wind

4 Point mass simulation from measured radial velocities - 3D method

The most general method for integrating the measured radial velocities into tangential velocity and trajectory, shown below, admits both radar data measured from an off-centered radar and also side wind and spin drift (lift due to yaw of repose).

The radial velocity U_r is measured. Given is the initial elevation θ_0 and azimuth ψ_0. Also given is the position of the radar $\vec{X}_R$ from a chosen origin. In the following, a coordinate-free approach is used to calculate the true trajectory $\vec{U}$ from the measured U_r, wind vector $\vec{W}$, (specific) lift-force $\vec{L}$ and (specific) Magnus force $\vec{M}$.

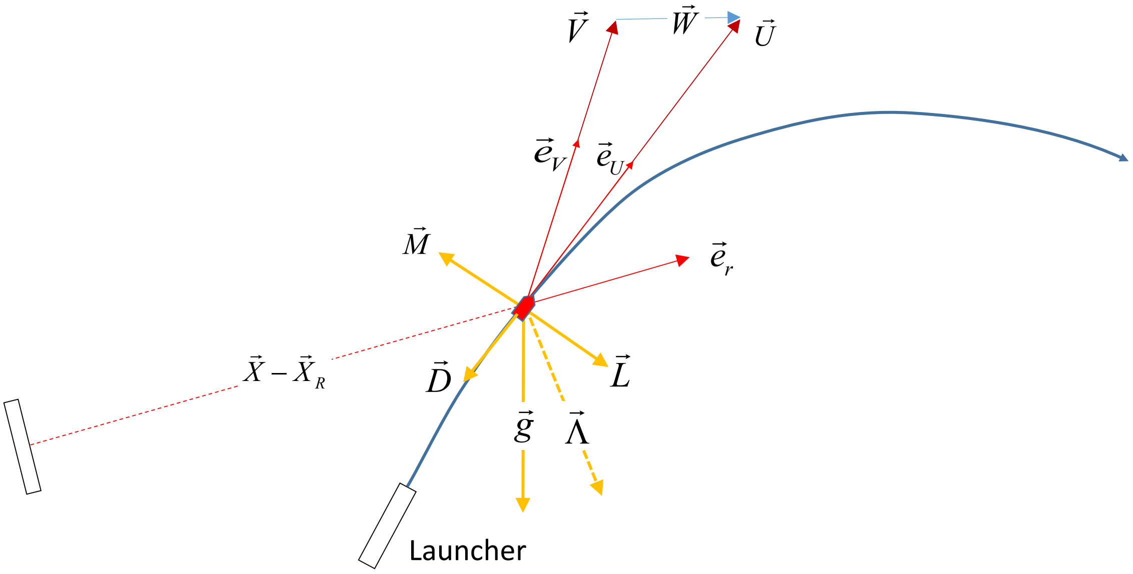


Figure 4.1 3D trajectory (blue) with forces (yellow) and position and velocity vectors (red). These vectors need not be co-planar.

4.1 Equations of motion

In this setting, the equations of motion are, considering Figure 4.1:

$$\begin{cases} \dot{\vec{X}} = U\vec{e}_U \\ U = \dfrac{U_r}{\vec{e}_U \cdot \vec{e}_r} \quad , \quad \vec{e}_r = \dfrac{\vec{X} - \vec{X}_R}{\left|\vec{X} - \vec{X}_R\right|} \\ \dot{\vec{e}}_U = F\left(\vec{e}_U, \dot{\vec{e}}_r, \dot{U}_r, U, g, \vec{W}, L, M, \vec{\Lambda}, \ldots\right) \end{cases} , \tag{4.1}$$

with initial conditions

$$\begin{cases} \vec{X}(0) = \vec{X}_0 \\ \vec{e}_U(0) = \begin{bmatrix} \cos\theta_0 \cos\psi_0 \\ \sin\theta_0 \\ \cos\theta_0 \sin\psi_0 \end{bmatrix} \end{cases} . \tag{4.2}$$

The last equation assumes that the y-axis is the vertical axis.

As in the 2D case, the challenge is the rate of change of the velocity direction, $\dot{\vec{e}}_U = \begin{bmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \end{bmatrix}$.

This vector has to be found by solving the linear system of equation (4.3), e.g. by LU-factorization. The derivation is shown in Appendix A.2.

$$\boxed{\begin{bmatrix} -r_0\beta_0 + u_0 & -r_1\beta_0 + v_2\cos\gamma + u_1 & -r_2\beta_0 - v_1\cos\gamma + u_2 \\ -r_0\beta_1 - v_2\cos\gamma + u_0 & -r_1\beta_1 + u_1 & -r_2\beta_1 + v_0\cos\gamma + u_2 \\ -r_0\beta_2 + v_1\cos\gamma + u_0 & -r_1\beta_2 - v_0\cos\gamma + u_1 & -r_2\beta_2 + u_2 \end{bmatrix} \begin{bmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix} \qquad (4.3)}$$

where

$$\begin{aligned} &\vec{e}_U = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix}, \quad \vec{e}_r = \begin{bmatrix} r_0 \\ r_1 \\ r_2 \end{bmatrix} = \frac{\vec{X} - \vec{X}_R}{\left|\vec{X} - \vec{X}_R\right|}, \quad \vec{e}_V = \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix} = \frac{U\vec{e}_U - \vec{W}}{\left|U\vec{e}_U - \vec{W}\right|}, \\ &\cos\gamma = \vec{e}_U \cdot \vec{e}_r = \frac{U}{U_r} \end{aligned} \tag{4.4}$$

and

$$\vec{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} \times \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} u_1 v_2 - u_2 v_1 \\ u_2 v_0 - u_0 v_2 \\ u_0 v_1 - u_1 v_0 \end{bmatrix}, \tag{4.5}$$

The right hand side is given by

$$\vec{f} = \left[\left(\frac{U - U_r \cos\varphi}{R} - \frac{\dot{U}_r}{U} \right) \vec{\beta} + \frac{\cos\gamma}{U} \left(\left(\vec{g} + \vec{\Lambda} \right) \times \vec{e}_V + L\vec{e}_M - M\vec{e}_L \right) \right], \tag{4.6}$$

where

$$\vec{e}_L = \frac{\vec{e}_V \times \vec{e}_y}{\left| \vec{e}_V \times \vec{e}_y \right|}, \quad \vec{e}_M = \vec{e}_L \times \vec{e}_V, \quad R = \left| \vec{X} - \vec{X}_R \right|. \tag{4.7}$$

4.2 Calculation of aerodynamic forces

If one wish to take into account the lateral forces (Magnus force and lift), coefficients for these have to be known, that is C_{mag} and C_{La}, and also spin damping coefficient C_{lp} and overturning moment coefficient C_{ma}, at different Mach numbers. The simplest model for lift and Magnus force is the Modified Point Mass Model (MPM) as described in e.g. NATO STANAG 4355 or McCoy [1].

In MPM, both lift and Magnus force depend on the spin p. This has to be numerically integrated throughout the trajectory. The equation of spin rate is

$$\dot{p} = \frac{1}{I_x} \left(\frac{\pi d^2}{4} \right) \tfrac{1}{2} \rho V^2 C_{lp} \frac{pd}{V} = \frac{\pi d^3}{8 I_x} \rho V \cdot C_{lp} p \,. \tag{4.8}$$

The *yaw of repose* α_R is given by

$$\sin\alpha_R = \frac{8 I_x g \cos\theta \cdot p}{\pi \rho d^3 V^3 C_{ma}}. \tag{4.9}$$

The lift and Magnus force are then given by

$$
\begin{aligned}
L &= \frac{\pi d^2}{8m}\rho V^2 C_{La}\sin\alpha_R \\
M &= \frac{\pi d^3}{8m}\rho V^2\, C_{mag}\frac{p}{V}\sin\alpha_R = \frac{\pi d^3}{8m}\rho V\, C_{mag}\, p\sin\alpha_R\,.
\end{aligned}
\tag{4.10}
$$

It should be noted that the air density ρ cancels out in the above equation (due to presence of ρ in α_R), and does not need to be evaluated for this purpose.

The coefficients depend on the Mach number, and therefore the speed of sound, which in turn requires the air temperature.

4.3 Calculation of Coriolis acceleration and gravity vector

The Coriolis acceleration is given by

$$
\vec{\Lambda} = 2\left(\vec{U}\times\vec{\Omega}\right) = 2\left(U\vec{e}_U\times\vec{\Omega}\right),
\tag{4.11}
$$

where $\vec{U}$ is the projectile velocity and $\vec{\Omega}$ is the rotation velocity vector of the earth. The latter is given by the latitude ϕ and the firing direction from north Az, in a coordinate system with x-axis along the line of fire and y-axis is the vertical axis at launch as:

$$
\vec{\Omega} = \Omega\begin{bmatrix}\cos\phi\cos\left(Az\right)\\ \sin\phi\\ \cos\phi\sin\left(Az\right)\end{bmatrix},\quad \Omega = 7.29\cdot10^{-5}\ \text{rad/s}\,.
\tag{4.12}
$$

The gravity vector is, taking the curvature of the earth into account, to first order,

$$
\vec{g} = \begin{bmatrix} -g_0\dfrac{X_x}{R_E}\\ -g_0\left(1-2\dfrac{X_y}{R_E}\right)\\ -g_0\dfrac{X_z}{R_E}\end{bmatrix},\qquad R_E = 6.356766\cdot10^{6}\ \text{m}\,,
\tag{4.13}
$$

where the ground value of g is given by (h = altitude above sea level):

$$
g_0 = 9.80665\ \text{m/s}^2\cdot\left(1-0.0026373\cdot\cos2\phi-\left(3.086\cdot10^{-6}\ \text{m}^{-1}\right)h\right).
\tag{4.14}
$$

4.4 Algorithm for 3D mass point simulation

The formulas above can be put together in the following algorithm.

The goal is to find the velocity vector, in particular the absolute value U which can be related to the measured radial velocity, but also the direction, given by elevation θ and azimuth ψ. Also needed is the position at all times.

Given a smoothed radial velocity history $U_r(t)$ as seen from the radar (or a *parallax corrected* radial velocity as seen from the launcher, by setting $\vec{X}_R = [0,0,0]^T$).

Step 1: Calculate initial values of $\vec{e}_U$ and $\vec{e}_r$ by

$$\vec{e}_U(0) = \begin{bmatrix} \cos\theta_0 \cos\psi_0 \\ \sin\psi_0 \\ \cos\theta_0 \sin\psi_0 \end{bmatrix} \tag{4.15}$$

$$\vec{e}_r(0) = \frac{\vec{X}(0) - \vec{X}_R}{\left|\vec{X}(0) - \vec{X}_R\right|} \quad , \quad \text{if } \left|\vec{X}(0) - \vec{X}_R\right| \approx 0 \text{, set } \vec{e}_r(0) = \vec{e}_U(0). \tag{4.16}$$

Step 2: Calculate tangential velocity U and rate of change of radial velocity $\dot{U}_r$:

$$U = \frac{U_r}{\vec{e}_U \cdot \vec{e}_r} \tag{4.17}$$

$$\dot{U}_r \approx \frac{U_r(t+\Delta t) - U_r(t)}{\Delta t} \tag{4.18}$$

$$\vec{e}_r = \frac{\vec{X} - \vec{X}_R}{R} \quad , \quad \text{where } R = \left|\vec{X} - \vec{X}_R\right| \tag{4.19}$$

Step 3 (optional): Calculate the aerodynamic forces (lift *L* and Magnus force *M*) according to section 4.2. Start with updating the spin rate *p* from (4.8) and calculate the yaw of repose from (4.9).

Step 4: With the given wind vector $\vec{W}$ calculate $\dot{\vec{e}}_U$ by (4.3) to (4.7).

<u>Step 5</u>: Update $\vec{e}_U$ from previous value based on $\dot{\vec{e}}_U$:

$$\vec{e}_U\left(t+\Delta t\right)=\vec{e}_U\left(t\right)+\dot{\vec{e}}_U\cdot\Delta t \tag{4.20}$$

<u>Step 5:</u> The position is updated by

$$\vec{X}\left(t+\Delta t\right)=\vec{X}\left(t\right)+\vec{U}\left(t\right)\Delta t\,. \tag{4.21}$$

Repeat from Step 2 until termination.

The elevation and azimuth angles are given from

$$\begin{cases}\theta=\arcsin\left(u_1\right)\\ \psi=\operatorname{arctan2}\left(u_2,u_0\right)\end{cases}\qquad\text{where } \vec{e}_U=\begin{bmatrix}u_0\\u_1\\u_2\end{bmatrix}. \tag{4.22}$$

5 Smoothing of radial velocity data

The measured radial velocity is usually very dispersed, and a smoothed version is needed. The data should be *parallax corrected* in advance.

5.1 Methods

The measured flight time is divided into uniform intervals, with a chosen time step, e.g. $\Delta t = 0.01$ s, The time interval will be used in the mass point simulation described in section 3 or 4, and therefore must be small enough for accuracy, but not too small for numerical errors.

The measured data points are transformed to a new dataset by a least square (LS) fitting:

$$\left(t_i, U_{r,i}^{raw}\right)_{i=1}^{N_{raw}} \rightarrow \left(t_j, U_{r,j}^{smoothed}\right)_{j=1}^{N} \tag{5.1}$$

At each time t_i, a given number of measurement points (typically 20 – 50) on each side of t_i is chosen for the LS-fitting (symmetric fitting). At the next time t_{i+1}, new points are chosen and a new fitting is made. The smoothed velocity at time t_i is then given by the determined linear or quadratic fitting formula.

Near the two ends, we have two options: 1) reduce the number of fitting points or 2) use an *asymmetric* fitting. The first option is tried first until a reasonable minimum number is reached, typically 10 – 20. Even closer to the muzzle or end of trajectory, asymmetric number of fitting points is used.

An example of a LS-fitting at time 35.00 seconds is shown in Figure 5.1.

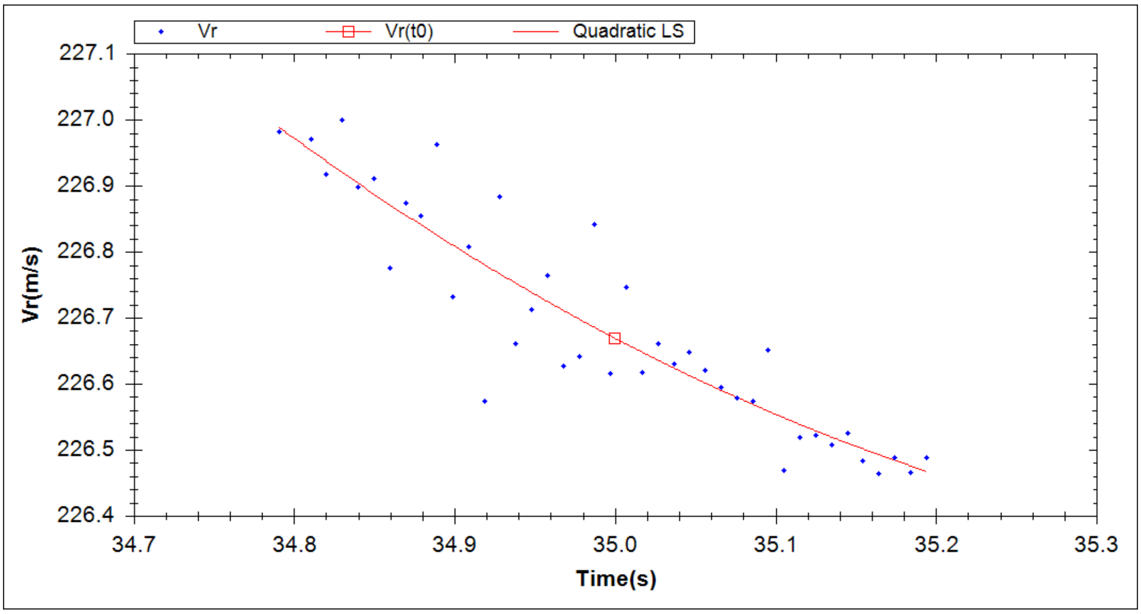


Figure 5.1 Example of a quadratic LS-fitting to the inverse velocity with ±20 points around the center time (in this case 35 seconds)

5.1.1 Method 1: "Inverse linear"

If a constant drag coefficient CD is assumed, the flat trajectory solution of velocity [2] is

$$U(t) = \frac{U_0}{1 + kU_0 t}, \tag{5.2}$$

where k is a constant proportional to CD. The inverse velocity is a linear function of time:

$$\frac{1}{U(t)} = \frac{1}{U_0} + kt. \tag{5.3}$$

Therefore, linear regression on the data set $\left(t_i, U_i^{-1}\right)_{i=1}^{N}$ should result in a good fit.

5.1.2 Method 2: "Inverse quadratic"

Method 2 is a variation of Method 1. Even the inverse velocity data set often has some curvature. Therefore, a *quadratic* least square fitting is preferred, that is, fitting a quadratic function $V^{-1} = a_0 + a_1 t + a_2 t^2$ to the inverse velocity data set. *This is generally the preferred method.*

5.1.3 Method 3: "Power law"

In the derivation of method 1 and 2, a constant CD was assumed. In some special cases, e.g. plastic ammunition with very high deceleration, it gives quite inaccurate and varying results for the muzzle velocity and initial drag coefficient, since the drag coefficient has changed a lot during the time from launch to the first measure point (due to varying Mach number). In this case, a more advanced formula for the velocity versus time function is appropriate. In the *supersonic* region, the drag function can be well modelled by

$$C_D = kU^{-\alpha}, \tag{5.4}$$

where *k* is proportional to CD. The "flat trajectory solution" (curvature ignored) with this assumption is given by (ref [2]):

$$\begin{cases} U(t) = \left(U_0^{\alpha-1} - (\alpha-1)kt\right)^{\frac{1}{\alpha-1}}, & \alpha \neq 1 \\ U(t) = U_0 e^{-kt}, & \alpha = 1 \end{cases} \tag{5.5}$$

The value $\alpha = \frac{1}{2}$ gives the *d'Antonio's formula* (ref [2]), and should be used as a primary value, while $\alpha = 0$ gives the constant CD solution (5.3).

For a given $\alpha \neq 1$ a linear LS-fit can be made on the data set $\left(t_i, U_i^{\alpha-1}\right)_{i=1}^{N}$, while for α near zero, the set $\left(t_i, \ln U_i\right)_{i=1}^{N}$ should be used.

It is also possible to vary the exponent α to find the fitting with the least root mean square error, but it does not always give consistent results.

Figure 5.2 and Figure 5.3 show the effect of the three methods on a plastic ammunition where the projectile has retarded to Mach 3.1 at the first measure point. In this case, the "power method" with $\alpha = 0.5$ seems to get most reasonable result, and also has low dispersion in predicted muzzle velocity. Figure 5.4 shows a typical radial velocity profile with the extrapolation back to time of launch. All methods give similar extrapolations.

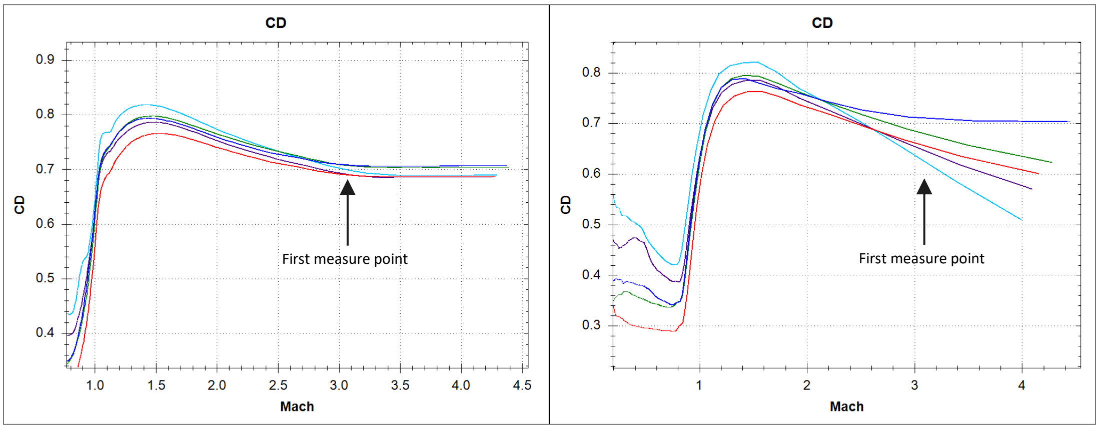


Figure 5.2 Example: 12.7 mm Plastic ammunition with the "inverse linear" method (left) and the "inverse quadratic" method (right). Each colored curve represents a projectile.

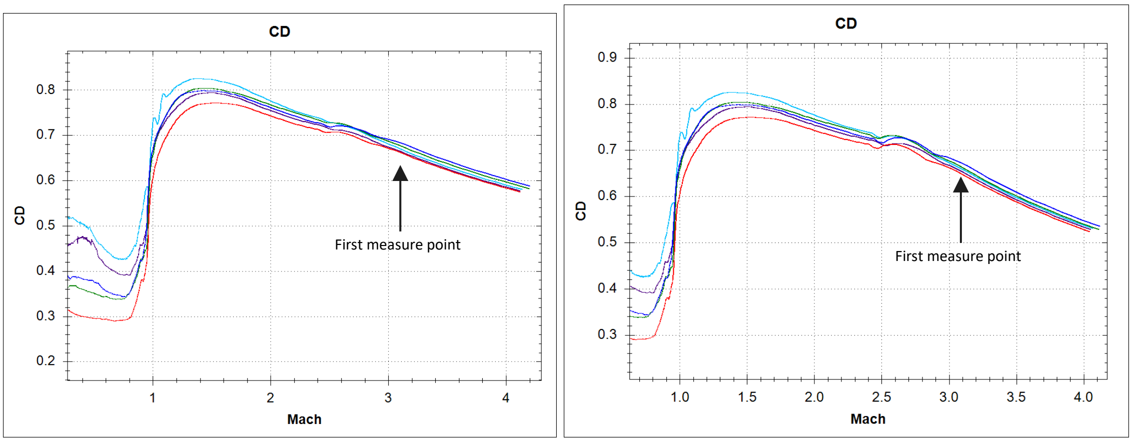


Figure 5.3 Example: 12.7 mm Plastic ammunition with the "power method" with $\alpha = 0.5$ *(left) and* $\alpha = 0.8$ *(right). Each colored curve represents a projectile.*

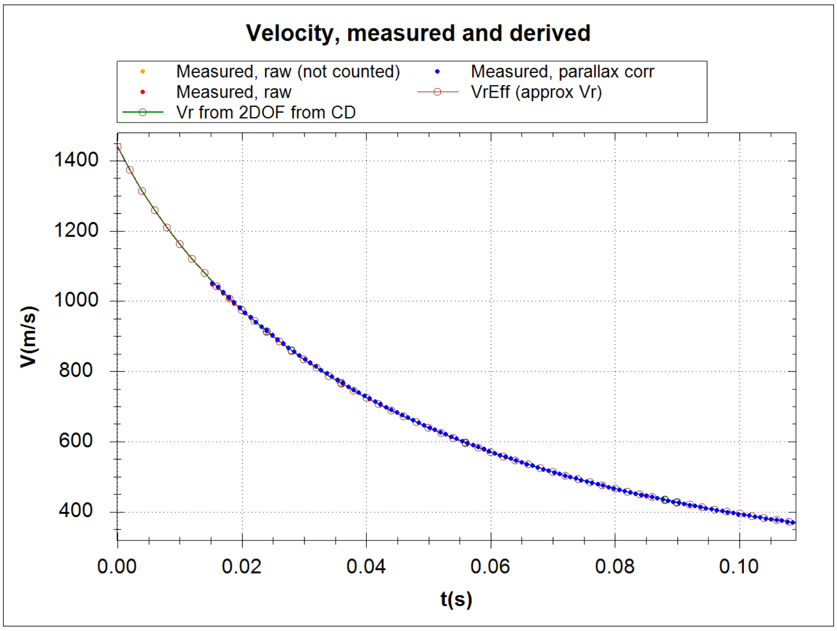


Figure 5.4 Example: 12.7 mm Plastic ammunition. Radial velocity vs time for one round.

5.2 Extrapolating back to muzzle

The smoothing methods in section 5.1 gives a smooth radial velocity history which traces back to the muzzle ($t = 0$). This also gives the muzzle velocity U_0. It is often necessary to smoothen the smoothed radial velocity further due to bad parallax correction and bad data near the muzzle.

The red points in Figure 5.4 are the smoothed radial velocity history. These points are generated by least square fittings on a symmetric interval around each red point as described in section 5.1. However, near the muzzle, an *asymmetric* fitting has to be used, which gives less accurate results. The smoothed points where an asymmetric fitting has been used, are ignored and replaced by points determined by an asymmetric quadratic or inverse quadratic LS-fitting of a number *N* of (smoothed) points. *N* could be in the range of 5 - 20.

Figure 5.5 shows the radial velocity fitting (red points on green line) and the resulting CD vs Mach curve when *not* adjusting the first points. Figure 5.6 shows the same round *with* adjustments.

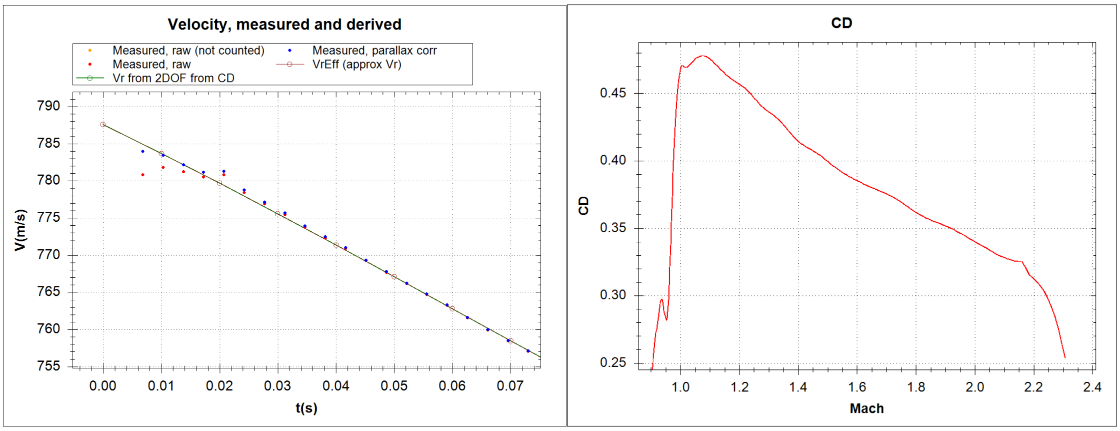


Figure 5.5 Radial velocity vs time (left) and resulting CD vs Mach number (right) when NOT adjusted near the muzzle

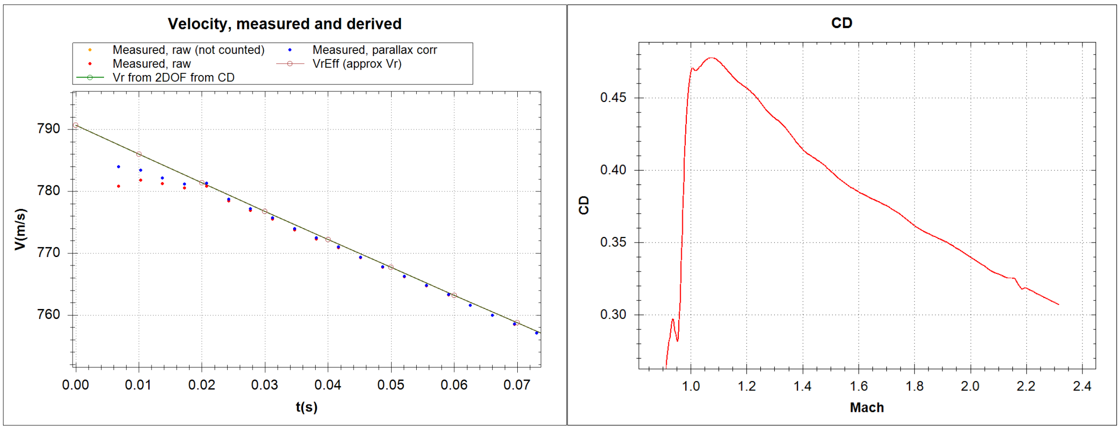


Figure 5.6 Radial velocity vs time (left) and resulting CD vs Mach number (right) after adjustment near the muzzle with 20 fitting points

5.3 Number of smoothing points

The choice of number of smoothing points in the LS-fitting is a trade-off of getting a smooth function and not capture sharp changes in velocity. Particularly, the drag coefficient varies greatly in the transonic region. In order to get a sharp drag coefficient profile, it is necessary to limit the number of smoothing points.

In order to find the exact location of the “sound barrier” (the steep drag rise around Mach 1), the smoothed radial velocity profile is examined to find the area in the transonic region with highest curvature. This is done by taking a 7-point quadratic LS-fit according to the formula $U^{-1} = a_0 + a_1 t + a_2 t^2$ at each time point in the transonic region. The time with the largest curvature (largest $|a_2|$) is chosen for the "center of sound barrier". In the example in Figure 5.7 this center is at Mach 0.95.

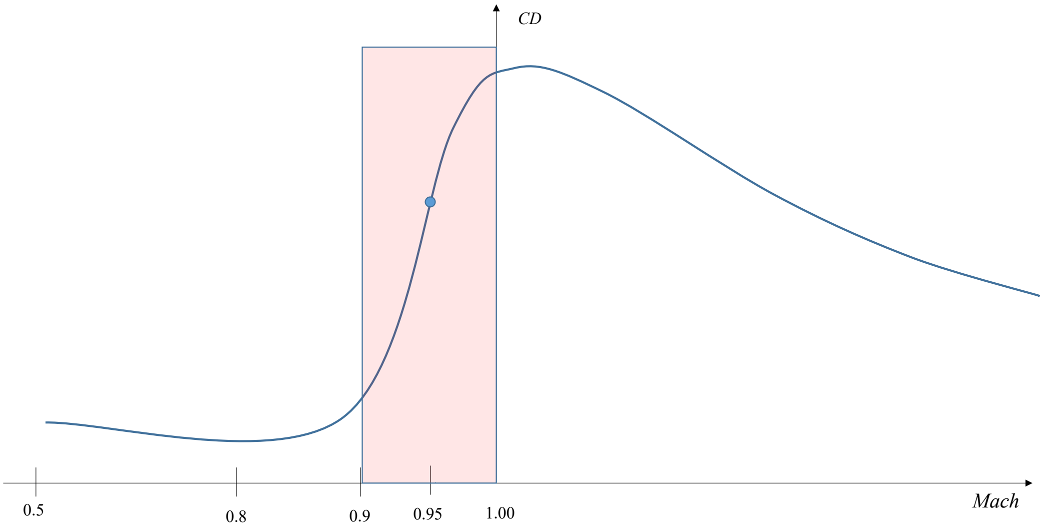


Figure 5.7 The transonic region of the CD vs Mach number curve centered at Mach 0.95

The number of smoothing points is given at some chosen Mach numbers. An example of Mach regions and number of points on each side of each fitting interval is given in Table 5.1. The numbers in each region are determined by linear interpolation.

Mach	Zone	# points
0.5	Low subsonic	100
0.8	High subsonic	20
0.9	Low transonic	20
0.95	Mid transonic	5
1.00	High transonic	20

Table 5.1 An example of number of smoothing points for the radial velocity in different Mach regions. The number of points is for each side of a given time point.

5.4 Further refinements

The smoothing of the measured radial velocities is done in three steps: First a run with constant numbers of smoothing points, then a run with the variable numbers of points according to section 5.2, and finally a LS-fit on the *smoothed* velocity data set. The effect of the last step is shown in Figure 5.8. The left figure is before the final smoothing.

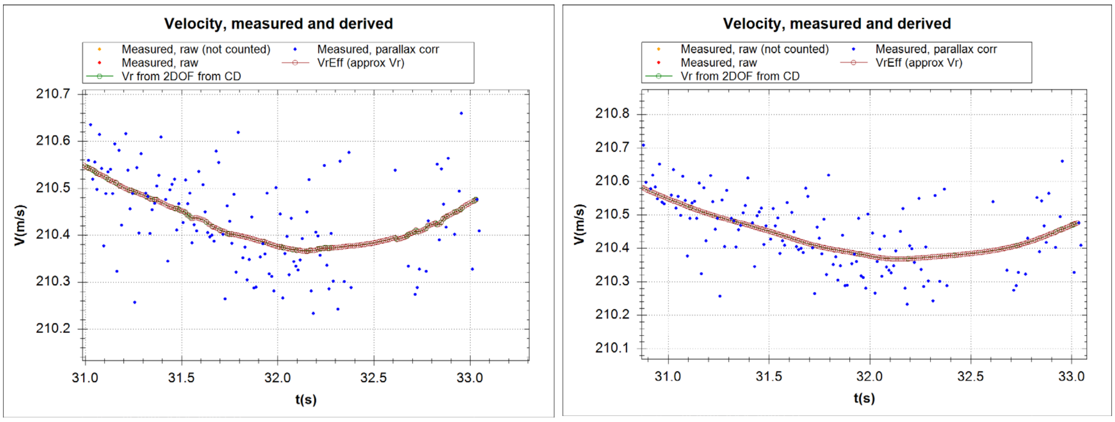


Figure 5.8 The smoothed data before (left) and after the final smoothing (right)

6 Extracting CD

After the (modified) point mass simulation from the radial velocities (section 3 or 4), *smooth* histories of tangential velocity, velocity direction and position are obtained. The drag coefficient (CD) is then calculated by "inverse point mass simulation" by solving (6.1) for C_D:

The point mass equation with wind is

$$\dot{\vec{U}} = -C_D \frac{\rho}{2m} V\vec{V} + \vec{g} \;\; , \quad \vec{V} = \vec{U} - \vec{W} \, , \tag{6.1}$$

where ρ = air density, m = projectile mass, $\vec{W}$ = wind, $\vec{g}$ = gravity acceleration vector.

From the smooth trajectory we have the data sets $\left\{ \vec{U}_i = U_i \vec{e}_i \right\}_{i=1}^{N}$ and $\left\{ \vec{X}_i \right\}_{i=1}^{N}$. From the current atmosphere model appropriate for the radar test, air density, the wind vector and gravity at time t_i is obtained:

$$\rho_i = \rho\left(Y_i\right) \quad , \quad \vec{W}_i = \vec{W}\left(\vec{X}_i\right) \; , \quad \vec{g}_i = \vec{g}\left(\vec{X}_i\right) \tag{6.2}$$

Equation (6.1) can be written as

$$\dot{\vec{U}} = -C_D \vec{b} + \vec{g} \, . \tag{6.3}$$

where $\vec{b} = \dfrac{\rho}{2m} V\vec{V}$ is known at each time step.

A discrete version of (6.3) is

$$\dot{\vec{U}}_i = -C_{D,i} \vec{b}_i + \vec{g}_i \tag{6.4}$$

where $\dot{\vec{U}}_i$ denotes a discretized version of $\dot{\vec{U}}$ at time t_i.

After taking dot product with $\dfrac{\vec{b}_i}{\vec{b}_i \cdot \vec{b}_i}$, the $C_{D,i}$ can be found:

$$\boxed{C_{D,i} = \left(\vec{g}_i - \dot{\vec{U}}_i \right) \cdot \frac{\vec{b}_i}{\vec{b}_i \cdot \vec{b}_i} \qquad , \; i = 0, \ldots, N-1} \tag{6.5}$$

The challenge is to find a good discretization of $\dot{\vec{U}}_i$. Symmetric 3-point formulas do the trick. At the ends, *asymmetric* 3-point formulas must be used.

$$\dot{\vec{U}}_i = \begin{cases} \frac{1}{2\Delta t}\left(-3\vec{U}_i + 4\vec{U}_{i+1} - \vec{U}_{i+2}\right), & i = 0 \\ \frac{1}{2\Delta t}\left(-\vec{U}_{i-1} + \vec{U}_{i+1}\right), & 0 < i < N-1 \\ \frac{1}{2\Delta t}\left(\vec{U}_{i-2} - 4\vec{U}_{i-1} + 3\vec{U}_i\right), & i = N-1 \end{cases} \tag{6.6}$$

It is also possible to use a 5-point formula, but it does not seem to be more accurate:

$$\dot{\vec{U}}_i = \begin{cases} \frac{1}{12\Delta t}\left(-25\vec{U}_i + 48\vec{U}_{i+1} - 36\vec{U}_{i+2} + 16\vec{U}_{i+3} - 3\vec{U}_{i+4}\right), & i = 0 \\ \frac{1}{12\Delta t}\left(-3\vec{U}_{i-1} - 10\vec{U}_i + 18\vec{U}_{i+1} - 6\vec{U}_{i+2} + \vec{U}_{i+3}\right), & i = 1 \\ \frac{1}{12\Delta t}\left(\vec{U}_{i-2} - 8\vec{U}_{i-1} + 8\vec{U}_{i+1} - \vec{U}_{i+2}\right), & 1 < i < N-2 \\ \frac{1}{12\Delta t}\left(-\vec{U}_{i-3} + 6\vec{U}_{i-2} - 18\vec{U}_{i-1} + 10\vec{U}_i + 3\vec{U}_{i+1}\right), & i = N-2 \\ \frac{1}{12\Delta t}\left(3\vec{U}_{i-4} - 16\vec{U}_{i-3} + 36\vec{U}_{i-2} - 48\vec{U}_{i-1} + 25\vec{U}_i\right), & i = N-1 \end{cases} \tag{6.7}$$

6.1 Smoothing of CD

If necessary, the CD vs time history may be smoothed by “least square” (LS) fittings. The number of fitting points should vary with Mach number. The subsonic region often requires more points in order to smoothen out CD variations. Dynamic instability in the subsonic region often leads to artificial (non-physical) sinusoidal variations. These variations could be dampened out by LS smoothing.

LS smoothing is done by a quadratic fitting of the set $\left\{t_i, C_{D,i}\right\}_{i=1}^{N}$ with the number of fitting points around t_i possible varying with Mach number. Usually, there is the greatest need for smoothing in the subsonic region.

7 Ricochet calculations

7.1 Setup

In ricochet measurements, the trajectory is divided into two parts: 1: From launcher to target, and 2: From target to ricochet. Of interest is the incoming velocity, the exit velocity, angle of departure in azimuth and elevation, and drag coefficient.

In ricochet measurements, the target plate (where the ricochet occurs) takes the position as the "Launcher", see Figure 7.1. The radar has to be placed well behind the launcher in order to keep the ricochets in the radar beam for as long time as possible. The 3D method from section 4 has to be used, now with the target plate as the "Launcher".

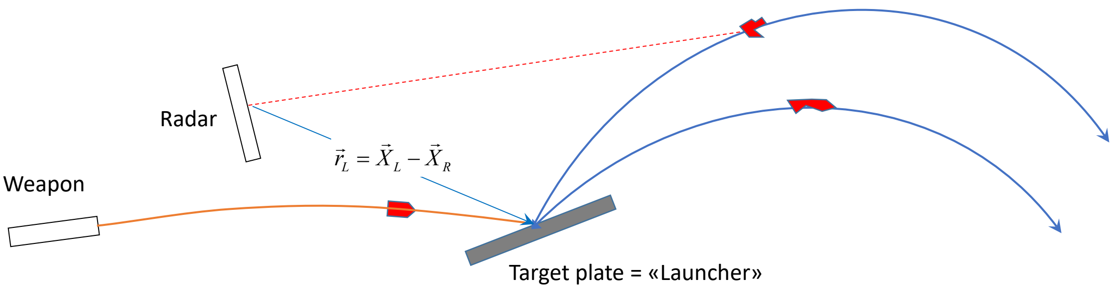

Radar
Weapon
$\vec{r}_L = \vec{X}_L - \vec{X}_R$
Target plate = «Launcher»

Figure 7.1 Ricochet setup. The new "Launcher" is the target plate

7.2 Ricochet departure angles

The 3D method assumed that the elevation and azimuth angles are known. That is not the case for ricochets, and we have to rely on the *measured* elevation and azimuth angles *as seen from the radar*. The outgoing elevation and azimuth angles (as seen from the target) could be calculated if the radial velocity and the rate of change of the measured elevation and azimuth angles are known.

However, the angular radar data from ricochets are usually quite noisy. A better method is to simulate the elevation and azimuth as seen from the radar, given a *guess* of the ricochet elevation and azimuth.

With this guessed ricochet direction, the radar centered 3D method of section 4 is applied. From the resulting positions, the elevation θ_R and azimuth ψ_R of the ricochet as *seen from the radar* are calculated:

$$\begin{cases} \theta_R(t) = \arctan\left(\dfrac{Y_R(t)}{\sqrt{X_R(t)^2 + Z_R(t)^2}} \right) \\ \psi_R(t) = \arctan\left(\dfrac{Z_R(t)}{X_R(t)} \right) \end{cases} \tag{7.1}$$

Here $\begin{bmatrix} X_R(t) \\ Y_R(t) \\ Z_R(t) \end{bmatrix} = \begin{bmatrix} X(t) \\ Y(t) \\ Z(t) \end{bmatrix} + \begin{bmatrix} \Delta X_L \\ \Delta Y_L \\ \Delta Z_L \end{bmatrix}$ is the position of the ricochet as seen from the radar, $\begin{bmatrix} X(t) \\ Y(t) \\ Z(t) \end{bmatrix}$ is the position of the ricochet as seen from the target plate and $\begin{bmatrix} \Delta X_L \\ \Delta Y_L \\ \Delta Z_L \end{bmatrix}$ is the position of the “Launcher” (target plate) relative to the radar.

An example fitting of this kind is shown in Figure 7.2. The green curve is the elevation and azimuth of the ricochet as calculated by (7.1), while the blue points are the measured values.

If the elevation/azimuth guess does not fit, a new guess is made, and a re-calculation is done.

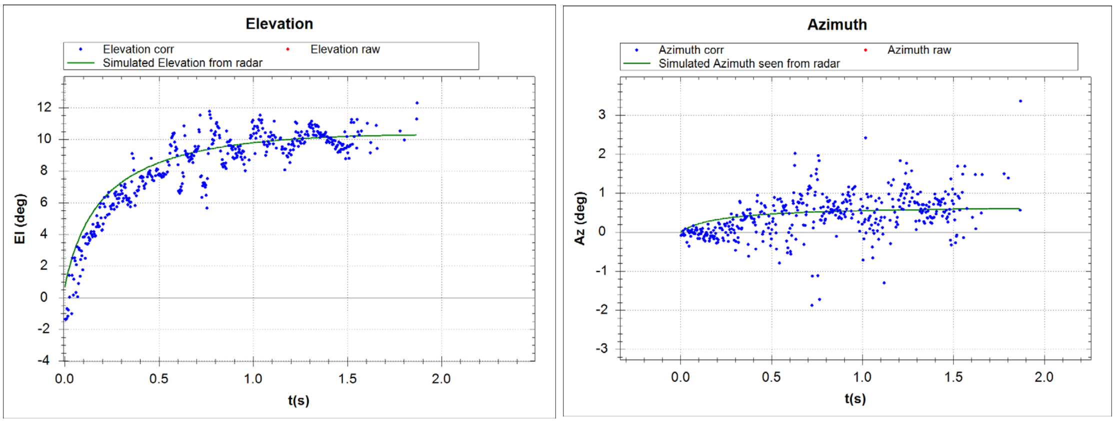


Figure 7.2 Fitting of the measured elevation (left) and azimuth (right) data

7.3 Ricochet velocities

The time of hit at the target plate is automatically detected by searching for a time where the radial velocity has a sudden change. In Figure 7.3 the measure points before impact are shown in orange, while the ricochet is shown in blue. The orange points are not used other than to determine the impact velocity. The time of impact is defined as time zero.

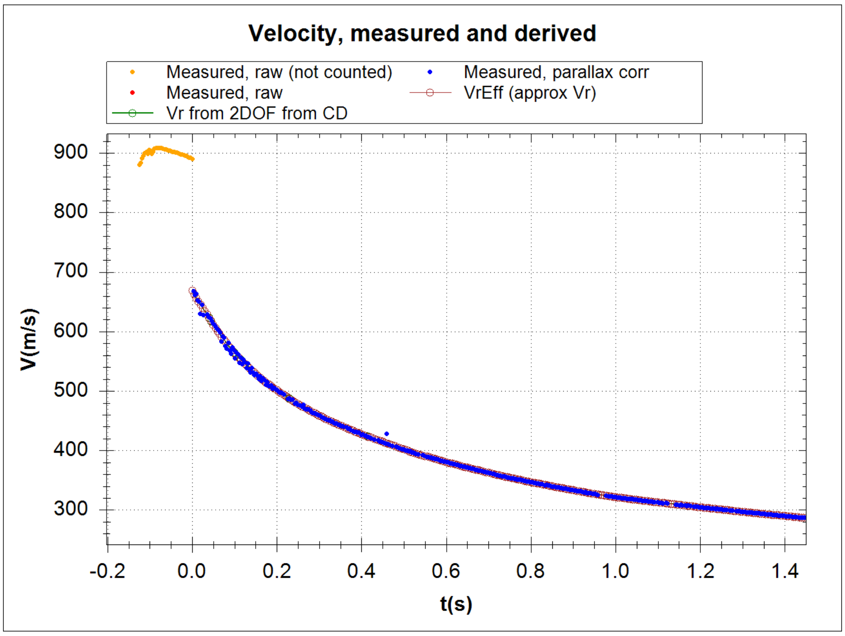


Figure 7.3 Radial velocities before (orange) and after (blue) ricochet where the time axis is translated such that the ricochet occurs at t=0 s

7.4 Extrapolation

In most cases, the ricochet disappears from the radar beam shortly after ricochet. Therefore, an extrapolation is often necessary to determine ricochet ranges. This is done by a point mass simulation. The CD value used in this simulation could be a frozen value from the last measured points. Instead, we are using a constant "form factor" *f* determined from the last measured points:

$$f = \frac{C_{D,ricochet}(M_{end})}{C_{D,projectile}(M_{end})}, \qquad (7.2)$$

where $C_{D,ricochet}(M_{end})$ is a LS-fitting of the last measured CD values at the last Mach number M_{end}, and $C_{D,projectile}(M_{end})$ is the CD of the (undeformed) projectile (given).

The CD values used thereafter, are simply the CD of the projectile scaled up:

$$C_{D,ricochet}(M) = f \cdot C_{D,projectile}(M) \tag{7.3}$$

In this way, a reasonable CD vs Mach function is used throughout the whole trajectory, even if only a few seconds (or tens of seconds) is actually measured.

An extrapolation is also used in non-ricochet cases if one wants to follow e.g. an artillery grenade all the way to the ground. In these cases, only the last seconds need to be extrapolated, and the CD is simply frozen to an average of the last CD values.

8 Spin measurements

It is sometimes possible to extract spin rate information from the radar data. This can happen if the projectile has some asymmetries which generates amplitude modulation of signal received. It can be made deliberately by making a "spin slot" on the projectile base, se examples in Figure 8.1, or accidentally by some other asymmetries. The principle is that there will be destructive interference when the spin slot is oriented in parallel with the polarization direction of the radar beam, which gives a varying amplitude of the reflected signal as the projectile rotates.

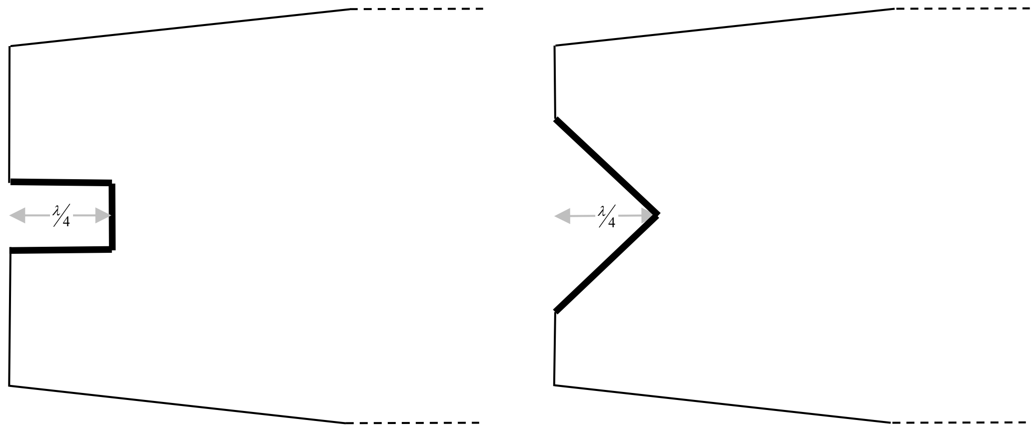


Figure 8.1 Two examples of spin slot behind a projectile, seen from the side. The depth should be ¼ wave length of the radar beam.

The amplitude modulated signal gives rise to a twin track on each side of the main track in the Doppler intensity plot from the radar software ("Wintrack" from Weibel Scientific), see Figure 8.2. The distance of the twin track from the main track is proportional to the spin rate.

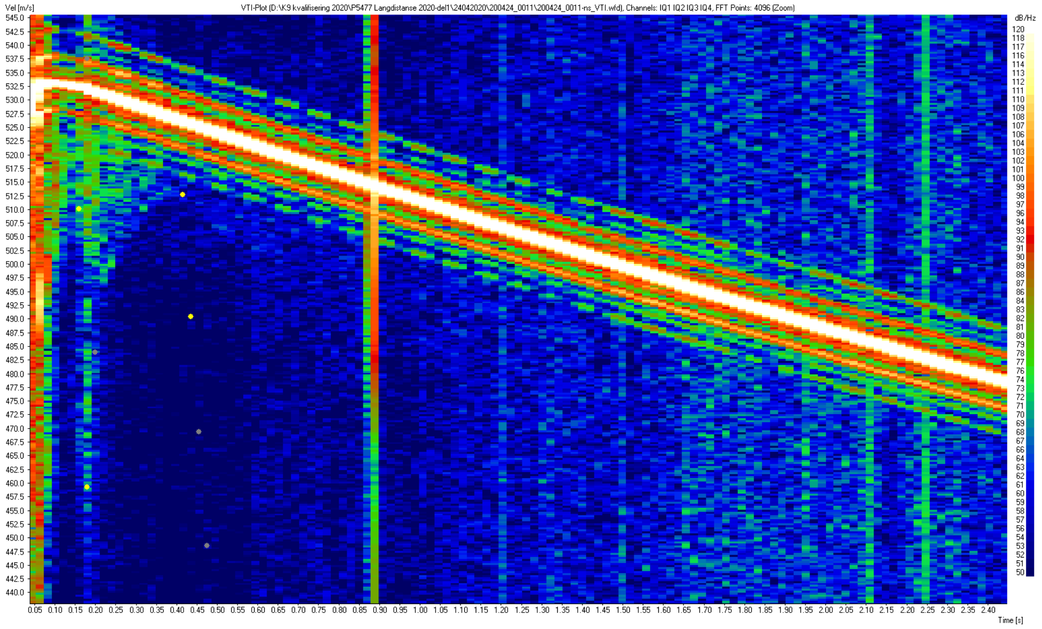


Figure 8.2 Doppler intensity plot showing the projectile track in the middle with a pair of twin tracks on each side.

The equation for the spin rate is

$$I_x \dot{p} = \frac{1}{2}\rho V^2 Sd\left[C_{ld} + \frac{pd}{V}C_{lp}\right], \tag{8.1}$$

where $S = \frac{\pi d^2}{4}$ is the reference area. C_{ld} is the spin driving moment which for axisymmetric projectiles is zero. With constant aerodynamic coefficients (C_D and C_{lp}) this equation is solvable if time derivative is replaced by distance derivative ($\frac{d}{dt} = V\frac{d}{ds}$).

It can be shown (Appendix A.3) that in this case (*s* is arc length):

$$p(s) = V_0 e^{-K_D s}\left(\eta_\delta + (\eta_0 - \eta_\delta)e^{-(K_p - K_D)s}\right) \tag{8.2}$$

where $K_D = \frac{\rho S}{2m}C_D$, $K_\delta = \frac{\rho Sd}{2I_x}C_{ld}$, $\eta_\delta = \frac{K_\delta}{K_p - K_D}$, $K_p = \frac{\rho Sd^2}{2I_x}\left(-C_{lp}\right)$

and $\eta_0 = \frac{p_0}{V_0} = \frac{p(0)}{V(0)}$.

In the ordinary case with spin stabilized projectiles, the spin driving moment is zero, and equation (8.2) is reduced to

$$p(s) = p_0 e^{-K_p s} \ , \ \text{where } p_0 = p(0) . \tag{8.3}$$

Taking logarithms:

$$\ln p(s) = \ln p_0 - K_p s \ , \ \text{where } p_0 = p(0) . \tag{8.4}$$

That means that, at any given time t_c, a data set $\left\{ s_i - s_c, \ln p_i \right\}_{i=1}^{N}$ around that time should be well fitted by a linear least square fitting with coefficients *a* and *b*:

$$y(s) = a\left(s - s_c\right) + b . \tag{8.5}$$

Then the spin value and spin damping coefficient at time t_c are simply

$$\boxed{\begin{cases} C_{lp}(t_c) = \dfrac{2I_x}{\rho S d^2} a \\ p\left(t_c\right) = e^b \end{cases} \qquad (8.6)}$$

The air density ρ is taken as the average value in the interval.

An example of such a fitting for one instant is shown in Figure 8.3, while Figure 8.4 shows the whole fitted spin history. In this example, a new fitting is made every 0.5 seconds (green points).

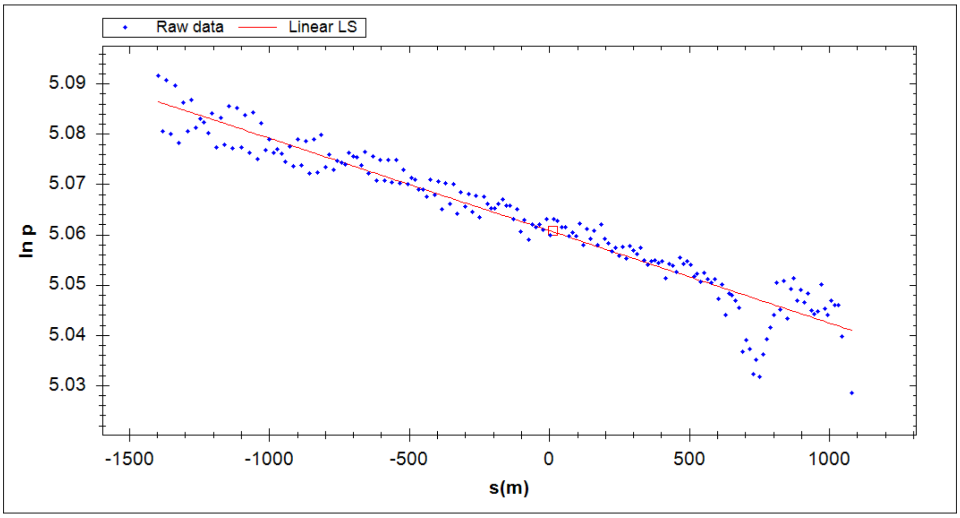


Figure 8.3 Example of one linear fitting of ln(p) vs arc length. Blue points are measure points, red line is the linear fitting.

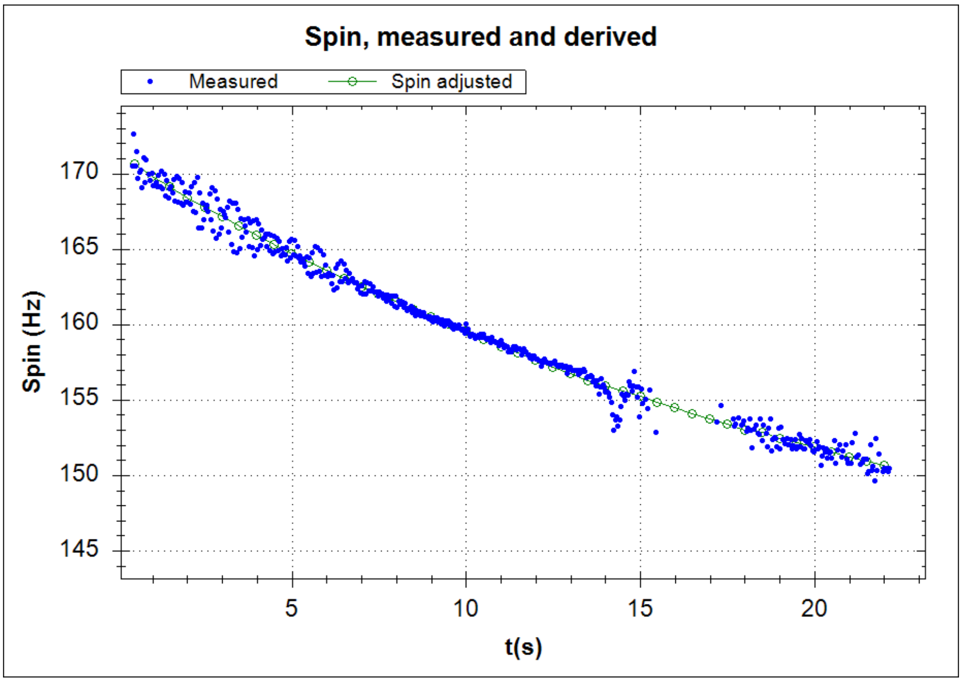


Figure 8.4 Spin data points (blue) together with fitting (green)

If both spin damping and spin driving coefficients are to be extracted (e.g. fin-stabilized projectiles), the best method is probably to fit those coefficients manually using simulation with guessed coefficients.

9 Conclusions

Methods and algorithms for analysis of ballistic radar data have been developed and implemented in an in-house code at FFI called “Weibelwin”. The main purpose of these algorithms is to extract drag coefficients, but other useful information can also be extracted, like vertical trajectory, elevation and azimuth, side drift, hit points. Methods for handling ricochet measurements and spin measurements have also been developed.

A Derivation of formulas

A.1 Derivation of rate of change of velocity direction in 2D

The following derivation is to establish expression (3.9).

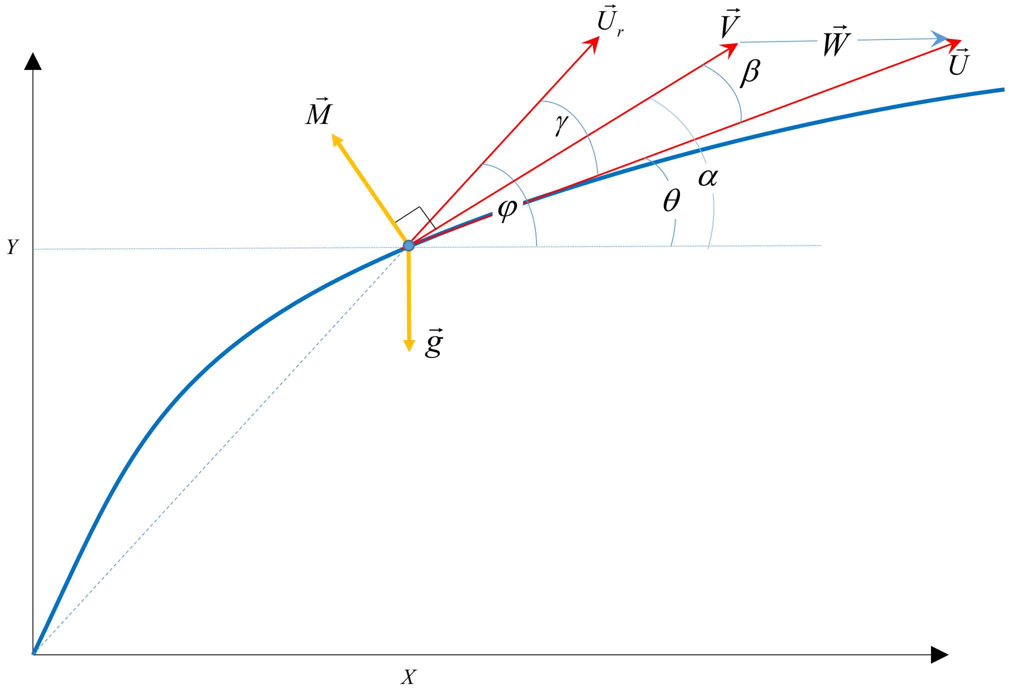


Figure A.1 Trajectory in 2D with range wind. Forces in yellow and velocities in red. The launcher is at the origin.

With reference to Figure A.1, the angle β between the true velocity vector $\vec{U}$ and the aerodynamic velocity vector $\vec{V}$ (assuming 2D plane) is given by

$$\sin\beta = \frac{\left(\vec{U}\times\vec{V}\right)_z}{UV} = \frac{\left(\vec{U}\times\left(\vec{U}-\vec{W}\right)\right)_z}{UV} = \frac{\left(-\vec{U}\times\vec{W}\right)_z}{UV} = \frac{U_yW_x - U_xW_y}{UV}. \tag{A.1}$$

Let $\alpha = \theta + \beta$ be the angle between the horizontal line and the aerodynamic velocity vector.

Let φ be the angle between the horizontal line and the direction of sight from the launcher to the object:

$$\varphi = \arctan\left(\frac{Y}{X}\right) \quad \Rightarrow \quad \dot{\varphi} = \frac{\dot{Y}X - \dot{X}Y}{X^2+Y^2} = \frac{U\left(X\sin\theta - Y\cos\theta\right)}{X^2+Y^2} \tag{A.2}$$

The overall change in the velocity vector (acceleration) is equal to the specific forces (forces per unit mass) according to Newton's second law:

$$\dot{\vec{U}} = \vec{g} + \vec{D} + \vec{M}\,, \tag{A.3}$$

where $\vec{D}$ is the (specific) drag force (which is along the aerodynamic velocity vector), $\vec{M}$ is the (specific) Magnus force (which is perpendicular to the aerodynamic velocity vector), and $\vec{g}$ is the gravity acceleration vector.

Define unity vectors $\vec{e}_D$ along the drag direction, $\vec{e}_M$ perpendicular to the drag direction, and $\vec{e}_r$ along the radial direction from launcher to projectile:

$$\begin{cases} \vec{e}_D = \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} \\ \vec{e}_M = \begin{bmatrix} -\sin\alpha \\ \cos\alpha \end{bmatrix} \\ \vec{e}_r = \begin{bmatrix} \cos\varphi \\ \sin\varphi \end{bmatrix} \end{cases} \tag{A.4}$$

The velocity vector and its derivative is in component form:

$$\begin{gathered} \vec{U} = U\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \\ \Downarrow \\ \dot{\vec{U}} = \dot{U}\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} + U\dot{\theta}\begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} \end{gathered} \tag{A.5}$$

The drag is unknown, and can be eliminated by considering the normal component to the velocity vector:

$$\dot{\vec{U}}\cdot\vec{e}_M = \vec{g}\cdot\vec{e}_M + \vec{D}\cdot\vec{e}_M + \vec{M}\cdot\vec{e}_M = \vec{g}\cdot\vec{e}_M + \vec{M}\cdot\vec{e}_M \tag{A.6}$$

Written out:

$$\left(\dot{U}\begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix}+U\dot{\theta}\begin{bmatrix}-\sin\theta\\ \cos\theta\end{bmatrix}\right)\cdot\begin{bmatrix}-\sin\alpha\\ \cos\alpha\end{bmatrix}=\begin{bmatrix}0\\ -g\end{bmatrix}\cdot\begin{bmatrix}-\sin\alpha\\ \cos\alpha\end{bmatrix}+M \tag{A.7}$$

or

$$-\dot{U}\left(\sin\alpha\cos\theta-\cos\alpha\sin\theta\right)+U\dot{\theta}\left(\sin\theta\sin\alpha+\cos\theta\cos\alpha\right)=-g\cos\alpha+M \tag{A.8}$$

That is,

$$-\dot{U}\sin\left(\alpha-\theta\right)+U\dot{\theta}\cos\left(\alpha-\theta\right)=M-g\cos\alpha\,. \tag{A.9}$$

Or, since $\alpha-\theta=\beta$

$$-\dot{U}\sin\beta+U\dot{\theta}\cos\beta=M-g\cos\alpha\,. \tag{A.10}$$

We also need to express $\dot{U}$ from $\dot{U}_r$. From Figure A.1:

$$\begin{aligned}&U_r=U\cos\left(\varphi-\theta\right)\\ &\Downarrow\\ &\dot{U}_r=\dot{U}\cos\left(\varphi-\theta\right)-U\sin\left(\varphi-\theta\right)\left(\dot{\varphi}-\dot{\theta}\right)\\ &\Downarrow\\ &\dot{U}=\frac{\dot{U}_r+U\sin\left(\varphi-\theta\right)\left(\dot{\varphi}-\dot{\theta}\right)}{\cos\left(\varphi-\theta\right)}\end{aligned} \tag{A.11}$$

Inserted into (A.10):

$$\left(-\frac{\dot{U}_r+U\sin\left(\varphi-\theta\right)\left(\dot{\varphi}-\dot{\theta}\right)}{\cos\left(\varphi-\theta\right)}\right)\sin\beta+U\dot{\theta}\cos\beta=M-g\cos\alpha \tag{A.12}$$

Solving for $\dot{\theta}$ we arrive at the conclusion:

$$\boxed{\dot{\theta}=\frac{\left(M-g\cos\left(\beta+\theta\right)\right)+\left(\dfrac{\dot{U}_r}{\cos\left(\varphi-\theta\right)}+\dot{\varphi}U\tan\left(\varphi-\theta\right)\right)\sin\beta}{U\left(\cos\beta+\sin\beta\tan\left(\varphi-\theta\right)\right)} \qquad (A.13)}$$

where α is replaced by $\alpha=\beta+\theta$.

A.2 Derivation of rate of change of velocity direction in 3D

This section develops the formulas for $\dot{\vec{e}}_U$, the rate of change of velocity direction.

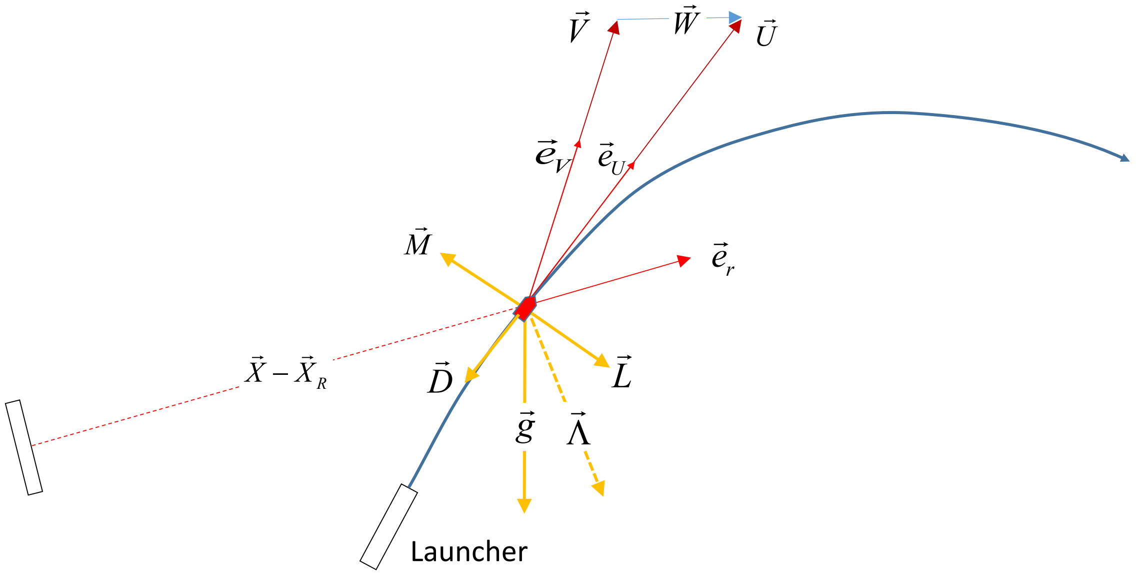


Figure A.2 Trajectory in 3D and the relevant vectors

With reference to Figure A.2, let $\vec{L}$ be the *specific* lift force (that is, lift force per unit mass) and $\vec{M}$ be the specific Magnus force. The lift force is horizontal to the right and perpendicular to the aerodynamic velocity vector $\vec{V}$. The Magnus force is in the vertical plane and perpendicular to $\vec{V}$ and $\vec{L}$. *These forces are assumed known from their given aerodynamic coefficients.* In addition, in the case of artillery, the Coriolis acceleration $\vec{\Lambda}$ may be taken into account.

The (modified) mass point model with drag $\vec{D}$, wind vector $\vec{W}$, gravity vector $\vec{g}$, the lift force L, Magnus force M and Coriolis acceleration $\vec{\Lambda}$ can be expressed by

$$\dot{\vec{U}} = -D\vec{e}_V + \vec{g} + L\vec{e}_L + M\vec{e}_M + \vec{\Lambda} = -D\vec{e}_V + \vec{F}\ , \tag{A.14}$$

where $\vec{V} = \vec{U} - \vec{W}\ ,\quad \vec{e}_V = \dfrac{\vec{V}}{V}\quad ,\ \vec{e}_y = \dfrac{-\vec{g}}{g}$

$$\vec{F} = \vec{g} + L\vec{e}_L + M\vec{e}_M + \vec{\Lambda} \tag{A.15}$$

$$\vec{e}_L = \frac{\vec{e}_V \times \vec{e}_y}{\left|\vec{e}_V \times \vec{e}_y\right|}\ ,\quad \vec{e}_M = \vec{e}_L \times \vec{e}_V \tag{A.16}$$

Here, a varying gravity vector may be used (in case of long range artillery). The formulas for the gravity vector and Coriolis vector are shown in section 4.3 (page 19).

To cancel out the unknown drag function (which is ultimately to be found), take cross products of (A.14) with $\vec{e}_V$ which is parallel to the drag vector:

$$\dot{\vec{U}} \times \vec{e}_V = \vec{F} \times \vec{e}_V = \left(\vec{g} + \vec{\Lambda}\right) \times \vec{e}_V + L\vec{e}_M - M\vec{e}_L \,. \tag{A.17}$$

Geometry gives

$$\begin{aligned}
&U_r = U\left(\vec{e}_U \cdot \vec{e}_r\right)\\
&\Downarrow\\
&\dot{U}_r = \dot{U}\left(\vec{e}_U \cdot \vec{e}_r\right) + U\left(\dot{\vec{e}}_U \cdot \vec{e}_r + \vec{e}_U \cdot \dot{\vec{e}}_r\right)\\
&\Updownarrow\\
&\dot{U}_r = \dot{U}\frac{U_r}{U} + U\left(\dot{\vec{e}}_U \cdot \vec{e}_r + \vec{e}_U \cdot \dot{\vec{e}}_r\right)\\
&\Updownarrow\\
&\dot{U} = \frac{U}{U_r}\left[\dot{U}_r - U\left(\dot{\vec{e}}_U \cdot \vec{e}_r + \vec{e}_U \cdot \dot{\vec{e}}_r\right)\right] = \frac{\dot{U}_r - U\left(\dot{\vec{e}}_U \cdot \vec{e}_r + \vec{e}_U \cdot \dot{\vec{e}}_r\right)}{\cos\gamma}
\end{aligned} \tag{A.18}$$

where $\cos\gamma = \vec{e}_U \cdot \vec{e}_r = \dfrac{U}{U_r}$

Equation (A.17) together with (A.18) gives:

$$\begin{aligned}
&\left(\dot{U}\vec{e}_U + U\dot{\vec{e}}_U\right) \times \vec{e}_V = \vec{F} \times \vec{e}_V\\
&\Updownarrow\\
&\dot{U}\left(\vec{e}_U \times \vec{e}_V\right) + U\left(\dot{\vec{e}}_U \times \vec{e}_V\right) = \vec{F} \times \vec{e}_V\\
&\Updownarrow\\
&\frac{\dot{U}_r - U\left(\dot{\vec{e}}_U \cdot \vec{e}_r + \vec{e}_U \cdot \dot{\vec{e}}_r\right)}{\cos\gamma}\left(\vec{e}_U \times \vec{e}_V\right) + U\left(\dot{\vec{e}}_U \times \vec{e}_V\right) = \vec{F} \times \vec{e}_V
\end{aligned} \tag{A.19}$$

The unknown entity to be solved for is $\dot{\vec{e}}_U$:

$$-\left(\dot{\vec{e}}_U \cdot \vec{e}_r\right)\vec{\beta} + \cos\gamma\left(\dot{\vec{e}}_U \times \vec{e}_V\right) = \vec{f}\left(\vec{e}_U, \dot{\vec{e}}_r, \dot{U}_r, U, g, \vec{W}, L, M\right) \tag{A.20}$$

where $\vec{\beta} = \left(\vec{e}_U \times \vec{e}_V\right)$, $\cos\gamma = \vec{e}_U \cdot \vec{e}_r$ and

$$\vec{f} = \left[\left(\left(\vec{e}_U \cdot \dot{\vec{e}}_r\right) - \frac{\dot{U}_r}{U}\right)\vec{\beta} + \frac{\cos\gamma}{U}\left(\vec{F} \times \vec{e}_V\right)\right] \tag{A.21}$$

To be able to solve for $\dot{\vec{e}}_U$, equation (A.20) has to be put into coordinate form.

Define coordinates

$$\vec{e}_U = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix}, \; \dot{\vec{e}}_U = \begin{bmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \end{bmatrix}, \; \vec{e}_r = \begin{bmatrix} r_0 \\ r_1 \\ r_2 \end{bmatrix}, \; \vec{e}_V = \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix}, \; \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} \times \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix} \text{ and } \vec{f}(...) = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix}$$

Then, (A.20) becomes

$$\begin{bmatrix} -\left(\dot{u}_0 r_0 + \dot{u}_1 r_1 + \dot{u}_2 r_2\right)\beta_0 + \left(\dot{u}_1 v_2 - \dot{u}_2 v_1\right)\cos\gamma \\ -\left(\dot{u}_0 r_0 + \dot{u}_1 r_1 + \dot{u}_2 r_2\right)\beta_1 + \left(\dot{u}_2 v_0 - \dot{u}_0 v_2\right)\cos\gamma \\ -\left(\dot{u}_0 r_0 + \dot{u}_1 r_1 + \dot{u}_2 r_2\right)\beta_2 + \left(\dot{u}_0 v_1 - \dot{u}_1 v_0\right)\cos\gamma \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix} \tag{A.22}$$

Or in matrix form for the solution of $\dot{\vec{e}}_U = \left(u_0, u_1, u_2\right)$:

$$\begin{bmatrix} -r_0\beta_0 & -r_1\beta_0 + v_2\cos\gamma & -r_2\beta_0 - v_1\cos\gamma \\ -r_0\beta_1 - v_2\cos\gamma & -r_1\beta_1 & -r_2\beta_1 + v_0\cos\gamma \\ -r_0\beta_2 + v_1\cos\gamma & -r_1\beta_2 - v_0\cos\gamma & -r_2\beta_2 \end{bmatrix} \begin{bmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix} \tag{A.23}$$

Equation (A.20) or (A.23) only determines the plane in which $\dot{\vec{e}}_U$ lies (the matrix in (A.23) has determinant = 0). One additional information has to be added. That is $\dot{\vec{e}}_U \cdot \vec{e}_U = 0$ (since $\vec{e}_U$ is a unity vector), written in component form:

$$\dot{u}_0 u_0 + \dot{u}_1 u_1 + \dot{u}_2 u_2 = 0 \tag{A.24}$$

or in matrix form:

$$\begin{bmatrix} u_0 & u_1 & u_2 \\ u_0 & u_1 & u_2 \\ u_0 & u_1 & u_2 \end{bmatrix} \begin{bmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \end{bmatrix} = 0 \tag{A.25}$$

which can be added to (A.23) in order to obtain a linearly independent system of equations. The result is:

$$\boxed{\left(\begin{bmatrix} -r_0\beta_0 & -r_1\beta_0+v_2\cos\gamma & -r_2\beta_0-v_1\cos\gamma \\ -r_0\beta_1-v_2\cos\gamma & -r_1\beta_1 & -r_2\beta_1+v_0\cos\gamma \\ -r_0\beta_2+v_1\cos\gamma & -r_1\beta_2-v_0\cos\gamma & -r_2\beta_2 \end{bmatrix}+\begin{bmatrix} u_0 & u_1 & u_2 \\ u_0 & u_1 & u_2 \\ u_0 & u_1 & u_2 \end{bmatrix}\right)\begin{bmatrix} \dot u_0 \\ \dot u_1 \\ \dot u_2 \end{bmatrix}=\begin{bmatrix} f_0 \\ f_1 \\ f_2 \end{bmatrix} \qquad \text{(A.26)}}$$

which, after solution of the linear system, gives the desired $\dot{\vec{e}}_U=\begin{bmatrix} \dot u_0 \\ \dot u_1 \\ \dot u_2 \end{bmatrix}$.

The right hand side is given by

$$\begin{aligned} \vec f &= \left[\left(\left(\vec e_U\cdot\dot{\vec e}_r\right)-\frac{\dot U_r}{U}\right)\vec\beta+\frac{\cos\gamma}{U}\left(\vec F\times\vec e_V\right)\right] \\ &= \left[\left(\frac{U-U_r\cos\varphi}{d}-\frac{\dot U_r}{U}\right)\vec\beta+\frac{\cos\gamma}{U}\left(\vec F\times\vec e_V\right)\right] \end{aligned} \qquad \text{(A.27)}$$

The only unknown term now is $\vec e_U\cdot\dot{\vec e}_r$. Starting with $\vec e_r=\frac{\vec X-\vec X_R}{\left|\vec X-\vec X_R\right|}$, time derivation gives

$$\dot{\vec e}_r=\frac{\left(\dot{\vec X}-\dot{\vec X}_R\right)\left|\vec X-\vec X_R\right|-\left(\vec X-\vec X_R\right)\left(\frac{d}{dt}\left|\vec X-\vec X_R\right|\right)}{\left|\vec X-\vec X_R\right|^2}. \qquad \text{(A.28)}$$

Let $R=\left|\vec X-\vec X_R\right|$ be the distance from radar to projectile. Then its time derivative is simply the measured radial velocity U_r: $\dot R=U_r$ which gives

$$\dot{\vec e}_r=\frac{\vec U R-\left(\vec X-\vec X_R\right)U_r}{R^2}=\frac{U\vec e_U-U_r\vec e_r}{R}. \qquad \text{(A.29)}$$

What we only need is $\vec e_U\cdot\dot{\vec e}_r$:

$$\vec e_U\cdot\dot{\vec e}_r=\frac{U\vec e_U\cdot\vec e_U-U_r\vec e_U\cdot\vec e_r}{R}=\frac{U-U_r\cos\varphi}{R} \qquad \text{(A.30)}$$

Therefore, the right hand side becomes

$$\vec{f} = \left[\left(\frac{U - U_r \cos\varphi}{R} - \frac{\dot{U}_r}{U}\right)\vec{\beta} + \frac{\cos\gamma}{U}\left(\left(\vec{g} + \vec{\Lambda}\right)\times\vec{e}_V + L\vec{e}_M - M\vec{e}_L\right)\right] \tag{A.31}$$

Reduction to 2D

It can be shown that the solution to (A.26) reduces to (A.13) in the case of 2D. In the following, the *xy*-plane is the vertical plane, and angles are defined in Figure A.1.

2D means that $v_2 = u_2 = r_2 = \beta_0 = \beta_1 = L = 0$, $\beta_2 = \sin\beta$,

$\vec{e}_y \times \vec{e}_V = \begin{bmatrix} 0 \\ 0 \\ -\cos\alpha \end{bmatrix}$, and $\vec{e}_L = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

It follows that (A.26) is reduced to

$$\begin{bmatrix} u_0 & u_1 & -v_1\cos\gamma \\ u_0 & u_1 & v_0\cos\gamma \\ -r_0\beta_2 + v_1\cos\gamma + u_0 & -r_1\beta_2 - v_0\cos\gamma + u_1 & 0 \end{bmatrix} \begin{bmatrix} -\dot{\theta}\sin\theta \\ \dot{\theta}\cos\theta \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ f_2 \end{bmatrix} \tag{A.32}$$

Now, since $\begin{bmatrix} u_0 \\ u_1 \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$, (A.32) is reduced to

$$\begin{bmatrix} \cos\theta & \sin\theta & -v_1\cos\gamma \\ \sin\theta & \sin\theta & v_0\cos\gamma \\ -r_0\sin\beta + v_1\cos\gamma + \cos\theta & -r_1\sin\beta - v_0\cos\gamma + \sin\theta & 0 \end{bmatrix} \begin{bmatrix} -\dot{\theta}\sin\theta \\ \dot{\theta}\cos\theta \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ f_2 \end{bmatrix} \tag{A.33}$$

The first two equations give 0 = 0.

The third equation in (A.32), using that $\begin{bmatrix} r_0 \\ r_1 \end{bmatrix} = \begin{bmatrix} \cos\varphi \\ \sin\varphi \end{bmatrix}$ and $\begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix}$, gives

$$\begin{aligned} &\left(-\cos\varphi\sin\beta + \sin\alpha\cos\gamma + \cos\theta\right)\left(-\dot{\theta}\sin\theta\right) \\ &\quad + \left(-\sin\varphi\sin\beta - \cos\alpha\cos\gamma + \sin\theta\right)\dot{\theta}\cos\theta = f_2 \end{aligned} \tag{A.34}$$

where $f_2 = \left(-\dot{\varphi}\sin\left(\varphi-\theta\right) - \frac{\dot{U}_r}{U} \right)\sin\beta - \frac{\cos\gamma}{U}\left(M - g\cos\alpha\right)$

Further reduction of (A.34) gives

$$\begin{aligned}
&\left(-\cos\varphi\sin\beta + \sin\alpha\cos\gamma + \cos\theta\right)\left(-\dot{\theta}\sin\theta\right) \\
&\qquad + \left(-\sin\varphi\sin\beta - \cos\alpha\cos\gamma + \sin\theta\right)\dot{\theta}\cos\theta = f_2 \\
&\qquad \vdots \\
&\dot{\theta}\left[\cos\gamma\cos\left(\alpha-\theta\right) + \sin\beta\sin\left(\varphi-\theta\right)\right] = -f_2
\end{aligned} \tag{A.35}$$

This gives, remembering that $\alpha - \theta = \beta$, and $\gamma = \varphi - \theta$:

$$\dot{\theta} = \frac{\left(\dot{\varphi}\sin\left(\varphi-\theta\right) + \frac{\dot{U}_r}{U}\right)\sin\beta + \frac{\cos\left(\varphi-\theta\right)}{U}\left(M - g\cos\alpha\right)}{\cos\left(\varphi-\theta\right)\cos\beta + \sin\beta\sin\left(\varphi-\theta\right)} \tag{A.36}$$

which is essentially equal to (A.13).

A.3 Derivation of the spin equation

The equation for the spin rate is

$$\dot{p} = \frac{\rho V^2 S d}{2 I_x}\left[C_{ld} + \frac{pd}{V} C_{lp} \right], \tag{A.37}$$

where $S = \frac{\pi d^2}{4}$ is the reference area. C_{ld} is the spin driving moment coefficient.

Introduce $K_D = \frac{\rho S}{2m} C_D$, $K_\delta = \frac{\rho S d}{2 I_x} C_{ld}$, $\eta_\delta = \frac{K_\delta}{K_p - K_D}$, $K_p = \frac{\rho S d^2}{2 I_x}\left(-C_{lp}\right)$.

Then equation (A.37) becomes

$$\dot{p} = V^2 \left(K_\delta - K_p \frac{p}{V} \right). \tag{A.38}$$

Further, introduce $\eta = \dfrac{p}{V}$ [rad/m], and replace time derivative by distance derivative (‘) $\dfrac{d}{dt} = \dfrac{ds}{dt}\dfrac{d}{ds} = V\dfrac{d}{ds}$. We also use that $V(s) = V_0 e^{-K_D s}$ is the solution of velocity vs distance (in linear motion) with constant drag coefficient, and thereby $V' = \dfrac{d}{ds}V = -K_D V$.

Then equation (A.38) is transformed as

$$\dot{p} = \frac{d}{dt}\left(\eta V\right) = \dot{\eta}V + \eta\dot{V} = \eta' V^2 + \eta V V' = \eta' V^2 + \eta V\left(-K_D V\right) \tag{A.39}$$

$$\Updownarrow$$

$$V^2\left(\eta' + \eta\left(-K_D\right)\right) = \dot{p} = V^2\left(K_\delta - K_p\frac{p}{V}\right) \qquad ,$$

$$\Updownarrow$$

$$\eta' + \left(K_p - K_D\right)\eta = K_\delta$$

where η' denotes derivative of η with respect to distance.

This equation has solution

$$\boxed{\eta(s) = \left(\eta_\delta + (\eta_0 - \eta_\delta)e^{-(K_p - K_D)s}\right), \qquad \text{(A.40)}}$$

where $\eta_\delta = \dfrac{K_\delta}{K_p - K_D}$ is the “steady state” value of η, and $\eta_0 = \dfrac{p_0}{V_0}$ is the initial value of η,

Therefore, it is concluded that

$$\boxed{p(s) = V_0 e^{-K_D s}\left(\eta_\delta + (\eta_0 - \eta_\delta)e^{-(K_p - K_D)s}\right). \qquad \text{(A.41)}}$$

References

[1] R. L. McCoy, *Modern Exterior Ballistics - The launch and Flight Dynamics of Symmetric Projectiles*. Schiffer Publishing Ltd, 1999.

[2] Ø. Grandum, "Analytiske metoder for baneberegning av flatbaneprosjektiler," Forsvarets forskningsinstitutt, FFI-notat-2003/01742, 2003.

About FFI

The Norwegian Defence Research Establishment (FFI) was founded 11th of April 1946. It is organised as an administrative agency subordinate to the Ministry of Defence.

FFI's mission

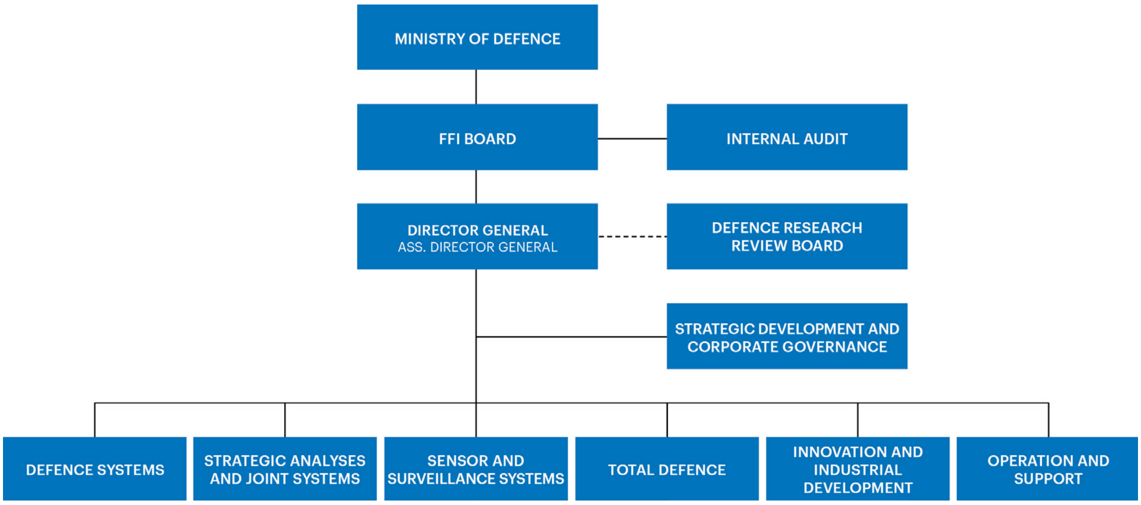
FFI is the prime institution responsible for defence related research in Norway. Its principal mission is to carry out research and development to meet the requirements of the Armed Forces. FFI has the role of chief adviser to the political and military leadership. In particular, the institute shall focus on aspects of the development in science and technology that can influence our security policy or defence planning.

FFI's vision

FFI turns knowledge and ideas into an efficient defence.

FFI's characteristics

Creative, daring, broad-minded and responsible.


MINISTRY OF DEFENCE
FFI BOARD
INTERNAL AUDIT
DIRECTOR GENERAL
ASS. DIRECTOR GENERAL
DEFENCE RESEARCH REVIEW BOARD
STRATEGIC DEVELOPMENT AND CORPORATE GOVERNANCE
DEFENCE SYSTEMS
STRATEGIC ANALYSES AND JOINT SYSTEMS
SENSOR AND SURVEILLANCE SYSTEMS
TOTAL DEFENCE
INNOVATION AND INDUSTRIAL DEVELOPMENT
OPERATION AND SUPPORT

Forsvarets forskningsinstitutt
Postboks 25
2027 Kjeller

Besøksadresse:
Instituttveien 20
2007 Kjeller

Telefon: 63 80 70 00
Telefaks: 63 80 71 15
Epost: post@ffi.no

Norwegian Defence Research Establishment (FFI)
P.O. Box 25
NO-2027 Kjeller

Office address:
Instituttveien 20
N-2007 Kjeller

Telephone: +47 63 80 70 00
Telefax: +47 63 80 71 15
Email: post@ffi.no