A General Approach for Altitude Estimation and Mitigation of Slant Range Errors on Target Tracking using 2D Radars

Edson Hiroshi Aoki
Department of Applied Mathematics
University of Twente/Embraer
Enschede, Netherlands/Sao Jose dos Campos, Brazil.
e.h.aoki@gmail.com

Abstract – *When target tracking using polar (azimuth and slant range only) measurements is performed, the most usual approach is to simply ignore slant range errors and perform target position estimation on a 2D plane. In reality, slant range errors are very significant and can seriously impair tracking. 3D target tracking can mitigate the effect of slant range errors, and, in some cases, even allow altitude to be estimated. This paper analyzes previous approaches on 3D target tracking using 2D radars and their drawbacks, and proposes efficient methods (based on EKF, UKF and AMM-EKF) that can be used on realistic scenarios, without significant increase on computational cost. A robust filter initialization technique is also proposed for these methods.*

Keywords: Tracking, slant range error, 2D radar, altitude estimation.

1 Introduction

Despite the increasing use of 3D radars, 2D radars are still widely used on surveillance and air traffic control applications. Although slant range measurements provided by 2D radars contain information about both target horizontal position and altitude, it's frequently assumed that the slant range error (the difference between slant range and ground range as represented on Fig. 1) is negligible and target tracking is performed on the horizontal (2D) plane.

As we know, however, slant range errors may be very significant - for instance, a target with an altitude of 10,000 m at a distance of 25,000 m from the radar will generate a measurement with slant range error of almost 2,000 m. The difficulty with dealing with slant range errors is such that it was even claimed [1] that the only solution to the problem was the use of 3D radars, or, at best, inflating the filter's range measurement error parameter.

However, it's quite clear that slant range errors are observable in certain conditions. For instance, if a target originates measurements from multiple synchronous 2D sensors, ground range may be calculated using geometric methods, such as triangulation or trilateration. Also, when a single 2D radar is available, and it's known that the target is performing a "stable" (constant ground velocity and altitude) trajectory, slant range errors can be calculated from observed deviations from this ideal trajectory. This "geometric-kinematic" approach was presented in [2].

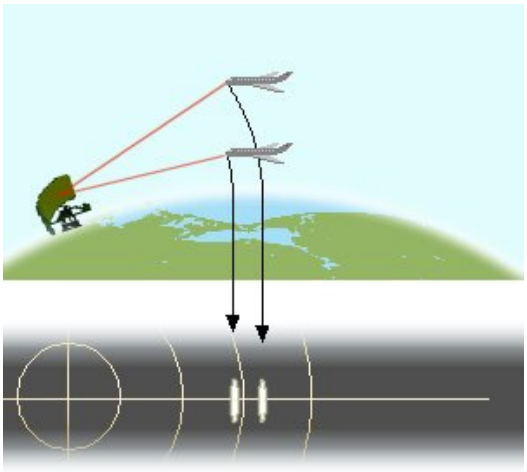

Figure 1: Slant range vs. ground range

Geometric and geometric-kinematic methods perform calculation of the slant range error and use it to compute the altitude, or do the opposite. In reality, stochastic errors, which include process noises (i.e. deviations from assumed kinematics) and measurement noises, are present and corrupt the results of these theoretically simple calculations. Another drawback of these methods is that they are only effective in the aforementioned conditions, and can't be applied to the general case.

Because of these problems, it's intuitive for us to look for a Bayesian solution which considers all *a priori* information and attempts to jointly mitigate slant range errors and estimate altitude. Since slant range measurements contain both horizontal position and altitude information, this could be achieved by performing tracking on the 3D space, rather than on the horizontal plane. This approach would not require any particular condition, but we may expect that we

will obtain better results when the conditions for applying geometric or geometric-kinematic methods apply, and we thus know that slant range errors are observable.

Unfortunately, since the relationship between polar measurements and 3D target position is nonlinear, optimal or quasi-optimal estimation would require nonlinear filtering techniques, such as particle filters. Due to the high dimensionality of 3D target state representation, the computational cost of nonlinear filtering may be prohibitive in typical online multi-target tracking applications. The Height-Parameterised Extended Kalman Filter (HPEKF) [3] was proposed to deal with such nonlinearities, and is discussed in detail in the next section.

This paper is organized as follows. Section 2 reviews the Height-Parameterised Extended Kalman Filter (HPEKF) approach for 3D target tracking using polar measurements, and summarizes its main problems. Section 3 presents a comparison between coordinate systems which may be used to represent the 3D target state, according to their suitability to the proposed problem. Section 4 proposes three filtering approaches that use measurements in polar coordinates and state vectors in geodetic coordinates, and introduces a robust filter initialization technique that can be applied to these methods. Section 5 presents simulations to compare the proposed and previous methods. Section 6 draws conclusions and suggests further work.

2 The Height-Parameterised Extended Kalman Filter

An Autonomous Multiple Model (AMM) Extended Kalman Filter (EKF) approach for 3D target tracking using 2D measurements was proposed in [3]. The method, named Height-Parameterised Extended Kalman Filter (HPEKF) consists on using paralell EKFs which estimate target states in cartesian coordinates using measurements in polar coordinates. An AMM filter is similar to the Interactive Multiple Model (IMM) filter, but without the mixing step, which means that no information exchange exists between the paralell filters.

In the HPEKF, the inital state is represented by a gaussian mixture, with each component of the mixture initialized with an assumed initial height. Each component of the mixture is then independently tracked by an EKF, and the state estimate obtained by performing a weighted sum of the components. This method is clearly based on the Range-Parameterised Extended Kalman Filter (RPEKF) proposed in [4], which is an approach to the problem of tracking 2D target position using bearing-only measurements.

The work of Ming-jiu et al. on the HPEKF showed that slant range error mitigation and altitude estimation is indeed possible when only 2D measurements from a single radar are available. Unfortunately, its implementation overlooked two important differences between this problem and the problem of tracking 2D target position using bearings-only measurements. First, like the RPEKF, the HPEKF uses a target state representation in cartesian coordinates relative to sensor position. The use of cartesian coordinates is problematic because the HPEKF requires the height of target to be nearly constant in order to provide adequate convergence. Due to the effect of Earth curvature, targets moving with constant altitude relative to the Earth surface will not have constant height in a local cartesian reference system (unless they move in circles around the sensor).

Second, in the original RPEKF, the mixture components are chosen by maintaining a constant Coefficient of Variation, C_r, given by σ_r/r, where r is the assumed initial range for the component and σ_r is its initial standard deviation. Then, σ_r is used to set the initial variances of x and y position coordinates. This approach makes sense since given a bearing-only measurement, its standard deviations on the horizontal plane increases linearly with assumed range. But Ming-jiu et al. recommends using a similar Coefficient of Variation for the HPEKF, only replacing range r with height z, despite the fact that no linear relationship exists between height and standard deviations on the horizontal plane. It's worthwhile to note that incorrect initialization may lead to considerable degradation of performance, since the HPEKF's multiple filters executing in paralell are only distinguished by their initial condition.

3 Analysis of coordinate systems for 3D tracking

3.1 Stereographic coordinates

An application of stereographic coordinates for the case of multi-sensor tracking is presented in [5]. A cartesian position (x,y,z) representation in stereographic coordinates is obtained by selecting a point (the origin of the coordinate system) and projecting the Earth surface on a plane centered on that point. Detailed information about stereographic projection can be found in [6].

The advantage of using stereographic coordinates over local cartesian coordinates is that, regardless of the origin of the coordinate system, the cartesian height of stereographic projection is equivalent to the geodetic altitude; thus, an aircraft that maintains nearly constant altitude h over the sea level will also have nearly constant cartesian height z. Besides, the transformation is conformal, which means that trajectory angles are preserved on the stereographic projection. However, the transformation is not isometric; a target following a rhumb route with a certain speed will have a different speed shown in the stereographic representation.

3.2 Geodetic coordinates

In [7], a filter that uses state representation in geodetic coordinates is proposed for target tracking using over-the-horizon radars. Modelling target trajectories directly in geodetic coordinates seems advantageous as it would theoretically allow great circle and rhumb routes to be described with precision.

Since describing motion models directly in geodetic coordinates is very difficult, a solution is to adapt motion models described in cartesian coordinates such as those listed in [8].

To accomplish that, an "hybrid" state representation may be used, with the position represented in geodetic coordinates and velocity (and accelerations, if applicable) represented in local cartesian coordinates relative to target position (not sensor position). An arbitrary vector v in local cartesian coordinates relative to target position may be converted to geodetic coordinates by using the following relations:

$$v_\phi = \frac{v_x}{M(\phi)} \tag{1}$$

$$v_\lambda = \frac{v_y}{N(\phi)} \tag{2}$$

$$v_h = v_z \tag{3}$$

where the cartesian coordinates x, y are in the NE (North, East) convention, and $M(\phi)$, $N(\phi)$ are respectively the arcradian lengths in latitude and longitude directions

$$M(\phi) = \frac{1}{ab\left(\frac{\sin^2(\phi)}{a^2} + \frac{\cos^2(\phi)}{b^2}\right)^{3/2}}$$

$$N(\phi) = \frac{a\cos(\phi)}{b\sqrt{\frac{\sin^2(\phi)}{a^2} + \frac{\cos^2(\phi)}{b^2}}}$$

where a and b are respectively the semi-major axis and semi-minor axis of the ellipsoid model.

As an example, let's convert the Constant Velocity (CV) model described in [8], given by

$$\begin{bmatrix} x_{k+1} \\ \dot{x}_{k+1} \\ y_{k+1} \\ \dot{y}_{k+1} \\ z_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ \dot{x}_k \\ y_k \\ \dot{y}_k \\ z_k \end{bmatrix} + W_k, \tag{4}$$

$$E[W_k W_k^T] = \begin{bmatrix} \frac{S_x T^3}{3} & \frac{S_x T^2}{2} & 0 & 0 & 0 \\ \frac{S_x T^2}{2} & S_x T & 0 & 0 & 0 \\ 0 & 0 & \frac{S_y T^3}{3} & \frac{S_y T^2}{2} & 0 \\ 0 & 0 & \frac{S_y T^2}{2} & S_y T & 0 \\ 0 & 0 & 0 & 0 & \frac{S_z}{T} \end{bmatrix}. \tag{5}$$

Using (1), (2) and (3) we obtain the following representation of the model in geodetic coordinates

$$\begin{bmatrix} \phi_{k+1} \\ \dot{x}_{k+1} \\ \lambda_{k+1} \\ \dot{y}_{k+1} \\ h_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \frac{T}{M(\phi_k)} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & \frac{T}{N(\phi_k)} & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \phi_k \\ \dot{x}_k \\ \lambda_k \\ \dot{y}_k \\ h_k \end{bmatrix} + \Upsilon_k, \tag{6}$$

$$E[\Upsilon_k \Upsilon_k^T] = \begin{bmatrix} \frac{S_x T^3}{3(M(\phi_k))^2} & \frac{S_x T^2}{2M(\phi_k)} & 0 & 0 & 0 \\ \frac{S_x T^2}{2M(\phi_k)} & S_x T & 0 & 0 & 0 \\ 0 & 0 & \frac{S_y T^3}{3(N(\phi_k))^2} & \frac{S_y T^2}{2N(\phi_k)} & 0 \\ 0 & 0 & \frac{S_y T^2}{2N(\phi_k)} & S_y T & 0 \\ 0 & 0 & 0 & 0 & \frac{S_z}{T} \end{bmatrix}. \tag{7}$$

Due to the dependency of arcradian lengths on ϕ, process noise Υ_k is correlated with target state $X_k = [\phi_k, \dot{x}_k, \lambda_k, \dot{y}_k, h_k]^T$, and X_{k+1} is a nonlinear function of X_k. Fortunately, for small distances the variation of arcradian lengths with latitude is almost negligible, so prediction may be performed using a standard EKF or UKF without significant performance loss.

To compare the tracking performance of stereographic and geodetic coordinate systems, we simulated a great circle trajectory of a target moving at 250 m/s to the northwest. For a radar with a scan rate of 10 s, the target moves 2500 m between two consecutive track updates. We then performed an one-step prediction using both the CV model in stereographic coordinates and our proposed modified CV model in geodetic coordinates, and compared the predicted position with the true position of the target after 10 s. The results are shown on Fig. 2, with the results for prediction using stereographic coordinates shown for varied distances (d) between target and coordinate center.

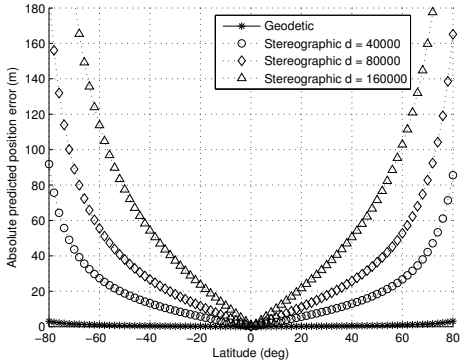

Figure 2: Absolute one-step prediction errors for stereographic and geodetic motion models

As it can be seen, prediction using geodetic coordinates is fairly accurate for all latitudes, while with stereographic coordinates, accuracy is lost as latitude and/or distance from coordinate center increases. These results suggest that despite the extra complexity, it is worthwhile to use geodetic coordinates for tracking.

4 Proposed filters for 2D measurements

4.1 Prediction step

Duan et al. [7] suggests using the Unscented Kalman Filter (UKF) on the prediction step for a tracker using geodetic coordinates. We prefer to use the EKF instead, since the computation of the jacobian of state transition in geodetic coordinates is relatively simple; we can assume it's much faster than performing the Cholesky decomposition of both track and process noise covariance required by the UKF. For the CV model described by (6), the jacobian of state transi-

tion is given by

$$F_k = \begin{bmatrix} 1+O(\phi_{k|k})T\dot{x}_{k|k} & \frac{T}{M(\phi_{k|k})} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ P(\phi_{k|k})T\dot{y}_{k|k} & 0 & 1 & \frac{T}{N(\phi_{k|k})} & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

where

$$O(\phi_{k|k}) = 3\left(\frac{b}{a} - \frac{a}{b}\right)\sin(\phi_{k|k})\cos(\phi_{k|k}) \sqrt{\frac{\sin^2(\phi_{k|k})}{a^2} + \frac{\cos^2(\phi_{k|k})}{b^2}}$$

$$P(\phi_{k|k}) = \frac{b^2\sin(\phi_{k|k})}{a^2\cos^2(\phi_{k|k})} \frac{1}{\sqrt{a^2\cos^2(\phi_{k|k}) - b^2\cos^2(\phi_{k|k}) + b^2}}.$$

4.2 Update step

In this section, three implementations of the update step of a filter that uses measuremnts in 2D polar coordinates and state vectors in 3D geodetic coordinates are proposed: UKF, EKF, and (revised) HPEKF.

4.2.1 UKF

The Unscented Kalman Filter [9] seems a natural choice for this problem, since it doesn't require the complex computation of the jacobian of the geodetic to polar transformation. Unfortunately, application of UKF is not as trivial as it seems to be, because we have chosen a hybrid state representation $X_k = [\phi_k, \dot{x}_k, \lambda_k, \dot{y}_k, h_k]^T$, with some components of the state vector expressed in radians and others in meters. The effect of this choice is a large disparity in the orders of magnitude of elements of the covariance matrix. Thus, applying the Unscented transform on such state vector generates a set of highly spaced and assymetrically distributed sigma points, which in turn may lead to the calculation of a covariance matrix with poor numeric conditioning (after the nonlinear transformation is applied to the sigma points). From our experience, using a scaled Unscented transform [10] reduces but not sufficiently alleviates the problem.

In [7], a square root UKF was used to deal with this problem. We propose an alternate solution, which consists in perform a re-scaling of state vector and covariance matrix prior to applying the Unscented transform. For instance, we may apply the inverse conversion of (1) and (2) to the horizontal position components of state vector $X_{k|k}$ and covariance matrix $P_{k|k}$, effectively turning the entire state vector and covariance into metric units. However, since the sole purpose of this conversion is re-scaling, accurate calculations are unnecessary and we may assume a spherical Earth model for simplicity. We believe that re-scaling is a better solution than square root filtering because it address the source of the problem (the poor distribution of sigma points) rather than the consequence (the poor numerical conditioning of the calculated covariance matrix), but we haven't performed meaningful experimental comparison between the two approaches.

4.2.2 EKF

Implementation of EKF requires the computation of the jacobian of state vector (in geodetic coordinates) to measurement (in polar coordinates) transformation. Let's consider this transformation as a 3-step procedure composed of 1) conversion from geodetic to Earth-Centered Earth-Fixed (ECEF) coordinates, 2) conversion from ECEF to local cartesian coordinates, and 3) conversion from local cartesian to polar coordinates. The jacobian H is then given by

$$H = H_3H_2H_1H_0 \quad (9)$$

where

$$H_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

and H_1, H_2 and H_3 are the jacobians of steps 1, 2 and 3 respectively. The jacobian of geodetic to ECEF conversion, H_1, is given by

$$\begin{aligned} H_1(1,1) &= t_1e^2\sin(\phi)\cos^2(\phi)\cos(\lambda) - (t_2+h)\sin(\phi)\cos(\lambda) \\ H_1(1,2) &= -(t_2+h)\cos(\phi)\sin(\lambda) \\ H_1(1,3) &= \cos(\phi)\cos(\lambda) \\ H_1(2,1) &= t_1e^2\sin(\phi)\cos^2(\phi)\sin(\lambda) - (t_2+h)\sin(\phi)\sin(\lambda) \\ H_1(2,2) &= (t_2+h)\cos(\phi)\cos(\lambda) \\ H_1(2,3) &= \cos(\phi)\sin(\lambda) \\ H_1(3,1) &= t_1\left(1-e^2\right)e^2\sin^2(\phi)\cos(\phi) + \left(t_2\left(1-e^2\right)+h\right)\cos(\phi) \\ H_1(3,2) &= 0 \\ H_1(3,3) &= \sin(\phi) \end{aligned} \quad (11)$$

where $[\phi, \lambda, h]^T$ is the predicted position (with the subscript $k+1|k$ omitted for convenience), e is the reference ellipsoid's eccentricity, and

$$t_1 = \frac{a}{\left(1-e^2\sin^2(\phi)\right)^{\frac{3}{2}}}$$

$$t_2 = \frac{a}{\sqrt{\left(1-e^2\sin^2(\phi)\right)}}.$$

The jacobian of ECEF to local cartesian conversion, H_2, is given by

$$H_2 = \begin{bmatrix} -\sin(\lambda_s) & \cos(\lambda_s) & 0 \\ -\cos(\lambda_s)\sin(\phi_s) & -\sin(\lambda_s)\sin(\phi_s) & \cos(\phi_s) \\ \cos(\phi_s)\cos(\lambda_s) & \cos(\phi_s)\sin(\lambda_s) & \sin(\phi_s) \end{bmatrix} \quad (12)$$

where ϕ_s and λ_s are the latitude and longitude of radar position, respectively. Here we assume that local cartesian coordinates are in EN (East, North) convention. Finally, the

jacobian of local cartesian to polar coordinates is given by

$$H_3 = \begin{bmatrix} \frac{y}{x^2+y^2} & -\frac{x}{x^2+y^2} & 0 \\ \frac{x}{\sqrt{x^2+y^2+z^2}} & \frac{y}{\sqrt{x^2+y^2+z^2}} & \frac{z}{\sqrt{x^2+y^2+z^2}} \end{bmatrix} \quad (13)$$

where $[x,y,z]^T$ is the predicted position converted to local cartesian coordinates.

4.2.3 HPEKF

We can now propose an AMM filter that uses multiple paralell filters, each using state vectors in geodetic coordinates and measurements in polar coordinates, similar to the original HPEKF reviewed in Section 2. The initial state is represented by a gaussian mixture with N components, with each component corresponding to a subinterval $[h_i, h_{i+1}]$, $i = 1,2,\ldots,N$ of the range $[h_1, h_{N+1}]$ of possible altitudes. Unlike the original HPEKF, where partitioning of subintervals and component initialization are heuristic, initialization of each component of the mixture is done by directly calculating the mean and covariance conditioned on the initial measurement and the altitude subinterval:

$$X^i_{0|0} = E\left[X|Z_0, \{h_i \le h < h_{i+1}\}\right] \quad (14)$$

$$P^i_{0|0} = E\left[\left(X - X^i_{0|0}\right)\left(X - X^i_{0|0}\right)^T |Z_0, \{h_i \le h < h_{i+1}\}\right] \quad (15)$$

where Z_0 is the initial measurement. A method to calculate an approximation of (14) and (15) is presented in the next section. The mean and covariance of the mixture are then given by

$$\begin{aligned} X_{0|0} &= E\left[X|Z_0\right] \\ &= \sum_{i=1}^{N} w^i_0 X^i_{0|0} \end{aligned} \quad (16)$$

$$\begin{aligned} P_{0|0} &= E\left[\left(X - X_{0|0}\right)\left(X - X_{0|0}\right)^T |Z_0\right] \\ &= \sum_{i=1}^{N} w^i_0 \left(P^i_{0|0} + \left(X^i_{0|0} - X_{0|0}\right)\left(X^i_{0|0} - X_{0|0}\right)^T\right) \end{aligned} \quad (17)$$

where w_0 is the initial weight of the component, given by the subinterval prior probability

$$w^i_0 = P(\{h_i \le h < h_{i+1}\}). \quad (18)$$

Note that partitioning of subintervals can be arbitrary, as long as subinterval prior probabilities $P(\{h_i \le h < h_{i+1}\})$ are correctly set. In practice, it's desirable to increase the number of subintervals on regions where more estimation accuracy is needed. After initialization, the estimates and weights are updated as per standard AMM. The filters for each component of the mixture should be identical; they differ only by their initial condition. Although we retain the name "HPEKF" for convenience, the paralell filters may be UKFs or any other filter that computes the first and second moments of the state probability density function.

4.3 Filter initialization

For the revised HPEKF described in the previous section, a Minimum Mean Square Error (MMSE) estimate of the state conditioned on the first measurement is given by (16); this obviously also applies to single model EKF or UKF, with $N = 1$. Thus, optimal initialization for the three proposed filters requires calculation of (14) and (15). Here, we present an approximate method to compute them.

Let $Z_0 = [\theta_m, r_m]^T$ be the initial measurement, where θ_m is the measured azimuth and r_m is the measured range. Let $\tilde{\theta}$ and $\tilde{r}$ be the corresponding azimuth and range measurement errors, assumed independent white noise sequences and also independent of the true target state, with variances respectively given by σ^2_θ and σ^2_r. If $\tilde{\theta}$ and $\tilde{r}$ are zero-mean and normally distributed then:

$$\begin{aligned} E[\cos(\tilde{\theta})] &= e^{-\frac{\sigma^2_\theta}{2}} \\ E[\sin(\tilde{\theta})] &= 0 \\ E[\cos^2(\tilde{\theta})] &= \frac{1}{2}\left(1 + e^{-2\sigma^2_\theta}\right) \\ E[\sin^2(\tilde{\theta})] &= \frac{1}{2}\left(1 - e^{-2\sigma^2_\theta}\right) \\ E[\sin(\tilde{\theta})\cos(\tilde{\theta})] &= 0 \\ E[\tilde{r}] &= 0. \end{aligned}$$

Let z' be the true height relative to sensor position, assumed to be contained within an interval $[z_i, z_f)$, with z_i and z_f (approximately) derived from altitude limits h_i and h_f, plus other constraints such as radar elevation beam width. If no prior information other than minimum and maximum values exists, then z' is an uniform random variable according to the principle of maximum entropy, with prior probability density function given by

$$f_{Z'}(z') = \begin{cases} \frac{1}{z_f - z_i} & z_i \le z' < z_f \\ 0, & \text{otherwise} \end{cases}. \quad (19)$$

Let $\iota = \arcsin\left(\frac{z'}{r_m - \tilde{r}}\right)$ be the true elevation. We then approximate ι by

$$\iota \approx \arcsin\left(\frac{z'}{r_m}\right). \quad (20)$$

This approximation is reasonable since usually $\tilde{r} << r_m$. Next, we calculate the probability density function of ι:

$$\begin{aligned} f_I(\iota) &= \frac{\partial P\left(\left\{\arcsin\left(\frac{z'}{r_m}\right) \le \iota\right\}\right)}{\partial \iota} \\ &= \frac{\partial P(\{z' \le r_m \sin(\iota)\})}{\partial \iota} \\ &= r_m \cos(\iota) f_{Z'}(r_m \sin(\iota)) \\ &= \begin{cases} \frac{r_m \cos(\iota)}{z_f - z_i} & \iota_f \le \iota < \iota_i \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (21)$$

where $\iota_f = \arcsin\left(\frac{z_f}{r_m}\right)$ and $\iota_i = \arcsin\left(\frac{z_i}{r_m}\right)$.

Integration of (21) yields the following expected values:

$$
\begin{aligned}
E[\cos(\iota)] &= \frac{1}{2\Delta z}\left(z_f t_{z_f} + r_m \iota_f - z_i t_{z_i} - r_m \iota_i\right) \\
E[\sin(\iota)] &= \frac{1}{2 r_m \Delta z}\left(z_f^2 - z_i^2\right) \\
E[\cos^2(\iota)] &= 1 - \frac{1}{3 r_m^2 \Delta z}\left(z_f^3 - z_i^3\right) \\
E[\sin^2(\iota)] &= \frac{1}{3 r_m^2 \Delta z}\left(z_f^3 - z_i^3\right) \\
E[\sin(\iota)\cos(\iota)] &= \frac{r_m}{3\Delta z}\left(t_{z_i}^3 - t_{z_f}^3\right)
\end{aligned}
$$

where $\Delta z = z_f - z_i$, $t_{z_f} = \sqrt{1 - \frac{z_f^2}{r_m^2}}$ and $t_{z_i} = \sqrt{1 - \frac{z_i^2}{r_m^2}}$.

The true target position in cartesian coordinates (using the ENU convention - East, North, Up) is given by:

$$
\begin{aligned}
x &= (r_m - \tilde{r})\sin(\theta_m - \tilde{\theta})\cos(\iota) \\
&= \cos(\iota)\big(r_m \sin(\theta_m)\cos(\tilde{\theta}) - r_m \cos(\theta_m)\sin(\tilde{\theta}) \\
&\quad - \sin(\theta_m)\tilde{r}\cos(\tilde{\theta}) + \cos(\theta_m)\tilde{r}\cos(\tilde{\theta})\big) \qquad (22) \\
y &= (r_m - \tilde{r})\cos(\theta_m - \tilde{\theta})\cos(\iota) \\
&= \cos(\iota)\big(r_m \cos(\theta_m)\cos(\tilde{\theta}) + r_m \sin(\theta_m)\sin(\tilde{\theta}) \\
&\quad - \cos(\theta_m)\tilde{r}\cos(\tilde{\theta}) - \sin(\theta_m)\tilde{r}\sin(\tilde{\theta})\big) \qquad (23) \\
z &= (r_m - \tilde{r})\sin(\iota). \qquad (24)
\end{aligned}
$$

Note that $z \neq z'$ due to approximation (20). The first and second moments of position in cartesian coordinates are then given by (omitting the conditioning on Z_0 and $\{h_i \leq h < h_f\}$)

$$
\begin{aligned}
E[x] &= E[\cos(\iota)]E[\cos(\tilde{\theta})] r_m \sin(\theta_m) \\
E[y] &= E[\cos(\iota)]E[\cos(\tilde{\theta})] r_m \cos(\theta_m) \\
E[z] &= E[\sin(\iota)] r_m \\
E[x^2] &= \left(r_m^2 + \sigma_r^2\right) E[\cos^2(\iota)] \left(E[\cos^2(\tilde{\theta})] - \cos^2(\theta_m) e^{-2\sigma_\theta^2}\right) \\
E[y^2] &= \left(r_m^2 + \sigma_r^2\right) E[\cos^2(\iota)] \left(E[\sin^2(\tilde{\theta})] + \cos^2(\theta_m) e^{-2\sigma_\theta^2}\right) \\
E[z^2] &= \left(r_m^2 + \sigma_r^2\right) E[\sin^2(\iota)] \\
E[xy] &= \left(r_m^2 + \sigma_r^2\right) E[\cos^2(\iota)] \sin(\theta_m)\cos(\theta_m) e^{-2\sigma_\theta^2} \\
E[xz] &= \left(r_m^2 + \sigma_r^2\right) E[\sin(\iota)\cos(\iota)] E[\cos(\tilde{\theta})] \sin(\theta_m) \\
E[yz] &= \left(r_m^2 + \sigma_r^2\right) E[\sin(\iota)\cos(\iota)] E[\cos(\tilde{\theta})] \cos(\theta_m).
\end{aligned}
$$

The initial position estimate and covariance in local cartesian coordinates can be then obtained by setting

$$
X_{0_c} = \begin{bmatrix} E[x] \\ E[y] \\ E[z] \end{bmatrix} \qquad (25)
$$

$$
P_{0_c} = \begin{bmatrix} E[x^2] & E[xy] & E[xz] \\ E[xy] & E[y^2] & E[yz] \\ E[xz] & E[yz] & E[z^2] \end{bmatrix} - X_{0_c} X_{0_c}^T. \qquad (26)
$$

The next step is to convert the initial condition to geodetic coordinates, which can be done by separately converting the initial state and covariance. The conversion of the covariance can be done by

$$
P_{0_g} = L_2 L_1 P_{0_c} L_1^T L_2^T \qquad (27)
$$

where L_1 is the jacobian of local cartesian to ECEF conversion, and L_2 is the jacobian of ECEF to geodetic conversion. Note that this conversion is biased; the converted state and covariance will not correspond to the true moments of position. L_1 is given by

$$
L_1 = \begin{bmatrix} -\sin(\lambda_s) & -\cos(\lambda_s)\sin(\phi_s) & \cos(\lambda_s)\cos(\phi_s) \\ \cos(\lambda_s) & -\sin(\lambda_s)\sin(\phi_s) & \sin(\lambda_s)\cos(\phi_s) \\ 0 & \cos(\phi_s) & \sin(\phi_s) \end{bmatrix} \qquad (28)
$$

where ϕ_s and λ_s are the latitude and longitude of radar position. To compute L_2, we performed linearization of Bowring's direct transformation [11], resulting in:

$$
\begin{aligned}
L_2(1,1) &= \left(-3t_{18}x_E/t_{14} - t_7 t_{16}/t_5\right)/t_{10} \\
L_2(1,2) &= \left(-3t_{18}y_E/t_{14} - t_7 t_{17}/t_5\right)/t_{10} \\
L_2(1,3) &= \left(\frac{t_{15}}{t_{14}} - \frac{3t_7 e^2 a^3 t_4 b^3 z_E}{t_5 t_8}\right)/t_{10} \\
L_2(2,1) &= -\frac{y_E}{x_E^2\left(1 + y_E^2/x_E^2\right)} \\
L_2(2,2) &= \frac{1}{x_E\left(1 + y_E^2/x_E^2\right)} \\
L_2(2,3) &= 0 \\
L_2(3,1) &= t_6 x_E/t_1 + t_{25}\left(t_{19}x_E - t_{26}t_{16}\right) \\
&\quad + t_{23}\left(6t_{21}t_{18}x_E + 2e^2 t_{24}t_{16} + t_{22}\left(t_{19}x_E - t_{26}t_{16}\right)\right) \\
L_2(3,2) &= t_6 y_E/t_1 + t_{25}\left(t_{19}y_E - t_{26}t_{17}\right) \\
&\quad + t_{23}\left(6t_{21}t_{18}y_E + 2e^2 t_{24}t_{17} + t_{22}\left(t_{19}y_E - t_{26}t_{17}\right)\right) \\
L_2(3,3) &= t_{25}t_{20} + t_{23}\left(-2t_{21}t_{15} + 6e^4 t_{24}a^3 t_4 b^3 z_E/t_8 + t_{22}t_{20}\right)
\end{aligned} \qquad (29)
$$

where $[x_E, y_E, z_E]^T$ is the initial state converted to ECEF coordinates, a and b are respectively the semi-major axis and semi-minor axis of the ellipsoid model, e is the eccentricity, e' is the second eccentricity, and

$$
\begin{aligned}
t_1 &= \left(x_E^2 + y_E^2\right)^{\frac{1}{2}}, \quad t_2 = \left(z_E^2 a^2 + b^2 x_E^2 + b^2 y_E^2\right)^{\frac{3}{2}} \\
t_3 &= \left(z_E + e'^2 b z_E^3 a^3/t_2\right)^2, \quad t_4 = \left(x_E^2 + y_E^2\right)^{\frac{3}{2}} \\
t_5 &= \left(t_1 - e^2 a t_4 b^3/t_2\right)^2, \quad t_6 = \left(1 + t_3/t_5\right)^{\frac{1}{2}} \\
t_7 &= z_E + e'^2 b z_E^3 a^3/t_2 \\
t_8 &= \left(z_E^2 a^2 + b^2 x_E^2 + b^2 y_E^2\right)^{\frac{5}{2}} \\
t_9 &= \left(t_1 - e^2 a t_4 b^3/t_2\right)^3, \quad t_{10} = 1 + t_3/t_5 \\
t_{11} &= \left(1 - \frac{e^2 t_3}{t_5 t_{10}}\right)^{\frac{3}{2}}, \quad t_{12} = 3e^2 a t_1 b^3/t_2 \\
t_{13} &= 3e^2 a t_4 b^5/t_8, \quad t_{14} = t_1 - e^2 a t_4 b^3/t_2 \\
t_{15} &= 1 + 3e'^2 b z_E^2 a^3/t_2 - 3e'^2 b z_E^4 a^5/t_8
\end{aligned}
$$

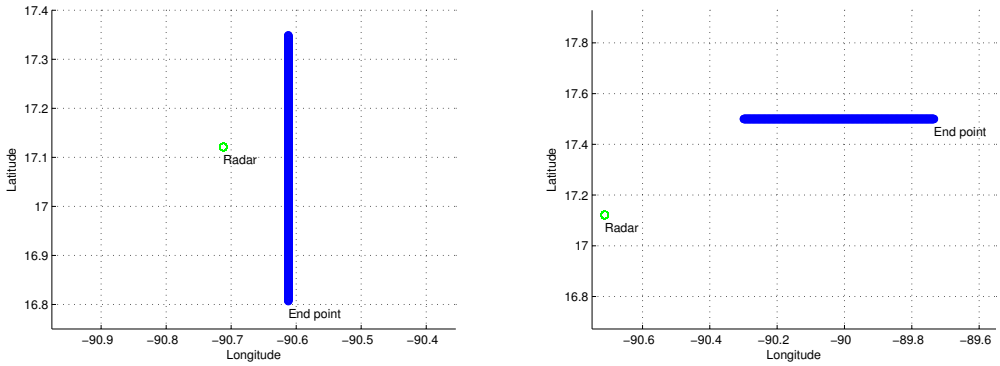

Figure 3: Simulated scenarios

$$t_{16} = x_E/t_1 - t_{12}x_E + t_{13}x_E$$
$$t_{17} = y_E/t_1 - t_{12}y_E + t_{13}y_E$$
$$t_{18} = e'^2 b^3 z_E^3 a^3/t_8, \quad t_{19} = -6t_7 t_{18}/t_5$$
$$t_{20} = \frac{2t_7 t_{15}}{t_5} - \frac{6t_3 e^2 a^3 t_4 b^3 z_E}{t_9 t_8}, \quad t_{21} = \frac{e^2 t_7}{t_5 t_{10}}$$
$$t_{22} = \frac{e^2 t_3}{t_5 t_{10}^2}, \quad t_{23} = \frac{a}{2t_{11}}, \quad t_{24} = \frac{t_3}{t_9 t_{10}}$$
$$t_{25} = \frac{t_1}{2t_6}, \quad t_{26} = 2t_3/t_9.$$

The velocity state components may be initialized using existing techniques, such as those described in [12].

5 Simulation

Consider two scenarios (Fig. 3) involving 2D radars and nonmaneuvering air targets. A simulation based on 50 Monte Carlo runs was executed for each scenario, with the three proposed implementations being compared with previous approaches (2D Kalman Filter and the original HPEKF from [3]).

The first scenario has the target flying close to the radar at 8,000 m, and the second scenario has the target flying away from the radar (also at 8,000 m altitude). The standard deviations for measurement noises are $\sigma_\theta = 0.15^\circ$ and $\sigma_r =$ 46.3 m for azimuth and range respectively.

Results are shown on Fig. 4. In the legend, 2D-KF is the 2D Kalman Filter assuming zero slant range error. O-HPEKF is the original HPEKF proposed by Ming-jiu et al., using 100 paralell EKF filters. G-EKF, G-UKF and G-HPEKF are the filters proposed on this paper (with state representation in geodetic coordinates), with the G-HPEKF using only 5 paralell EKF filters.

For the first scenario, we can see that all three filters using state representation in geodetic coordinates are able to both mitigate slant range errors and correcly estimate target altitude. Despite using 100 paralell filters, the O-HPEKF is only able to estimate altitude for a brief time, when the target is very near the sensor. The 2D KF, as expected, exhibits severe errors due to difference between slant range and ground range.

For the second scenario, while none of the implemented filters is able to correctly estimate altitude due to low observability, the "G" filters provide better slant range error mitigation than 2D-KF and O-HPEKF filters. In fact, the O-HPEKF performs significantly worse than even the 2D KF, possibly due to its biased initialization procedure.

The results also show that the G-HPEKF and the G-UKF overall performed better than the G-EKF. This suggested increasing the number of paralell filters in the G-HPEKF, or replacing the paralell EKFs with UKFs. However, in both scenarios, no significant increase in performance was obtained with these modifications.

6 Conclusions

In this paper, we presented a filtering-based approach to the problem of slant range errors caused by observation of air targets using 2D radars. We proposed three filters, which use a state representation with position in geodetic coordinates. We verified that using this representation (combined with MMSE initialization), provides better position estimation than filtering using a 2D or 3D local cartesian state representation.

The results show that for nonmaneuvering targets, slant range error mitigation is usually possible, and sometimes, altitude estimation is also possible. It is clear from the results that being able to mitigate slant range errors does not automatically imply being able to estimate altitude.

The obtained results were accomplished solely by linear filters or combination of linear filters, but suggest that further improvement may be obtained by using nonlinear filters; this investigation will be subject of further work.

References

[1] M. P. Dana, "Registration: a prerequisite for multiple sensor tracking," in *Multitarget-Multisensor Tracking: Applications and Advances*, Y. Bar-Shalom, Ed. YBS, 1996, vol. I, ch. 5.

[2] T. E. Wood, R. S. Ager, R. B. Fleury, and G. D. Heuer, "Methods and apparatus for providing target altitude

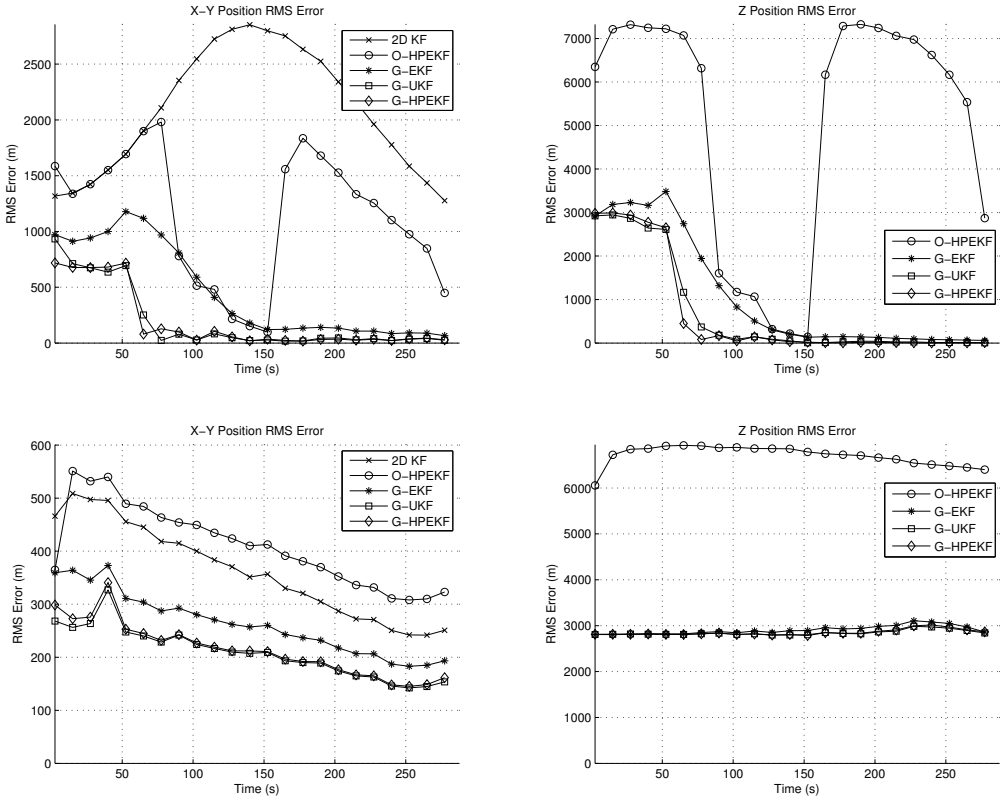

Figure 4: RMS errors

estimation in a two dimensional radar system," U.S. Patent 7 417 583, Aug. 26, 2008.

[3] G. Ming-jiu, Y. Xiao, H. You, and S. Bao, "An approach to tracking a 3D-target with 2D-radar," in *Proc. IEEE International Radar Conference*, Crystal Gateway Marriott Arlington, VA, May 9–12, 2005, pp. 763–768.

[4] N. Peach, "Bearings-only tracking using a set of range-parameterised extended kalman filters," in *Proc. IEE Control Theory Appl.*, vol. 142, no. 1, Jan. 1995, pp. 73–80.

[5] S. S. Blackman, R. J. Dempster, and T. S. Nichols, "Application of multiple hypothesis tracking to multi-radar air defense systems," in *Multi-Sensor Multi-Target Data Fusion, Tracking and Identification Techniques for Guidance and Control Applications (AGARD-AG-337)*, Oct. 1996, pp. 96–120.

[6] E. M. Shank, "A coordinate conversion algorithm for multisensor data processing," MIT Lexington Lincoln Laboratory, Cambridge, MA, Tech. Rep. DOT/FAA/PM-86/37, DTIC Acession No. ADA176368, Aug. 1986.

[7] L. Duan, X. H. Huang, B. Luo, and Q. Y. Li, "Target tracking with interactive multiple model in geodetic coordinate system for naval ships cooperative engagement," in *Proc. 11th International Conference on Information Fusion*, Cologne, Germany, Jun. 30, Jul. 1–3, 2008, pp. 2011–2018.

[8] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I: Dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[9] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Proc. SPIE Signal Processing, Sensor Fusion, and Target Recognition 'VI*, vol. 3068, Orlando, FL, Apr. 21–24, 1997, pp. 182–193.

[10] S. J. Julier, "The scaled unscented transformation," in *Proc. American Control Conference*, vol. 6, Anchorage, AK, May 8–10, 2002, pp. 4555–4559.

[11] B. R. Bowring, "Transformation from spatial to geographical co-ordinates," *Survey Review*, vol. 181, no. XXIII, pp. 323–327, Jul. 1976.

[12] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation*. New York, NY: John Wiley & Sons, 2001, ch. 5.