Two-stage Tracking Algorithm for Passive Radar

Mateusz Malanowski
Institute of Electronic Systems
Warsaw University of Technology
Warsaw, Poland
m.malanowski@elka.pw.edu.pl

Krzysztof Kulpa
Institute of Electronic Systems
Warsaw University of Technology
Warsaw, Poland
k.kulpa@elka.pw.edu.pl

Radoslaw Suchozebrski
Institute of Electronic Systems
Warsaw University of Technology
Warsaw, Poland
adros5@interia.pl

Abstract – Passive Coherent Location (PCL) radar measures the bistatic parameters of a target: the time difference of arrival and the Doppler shift. In order to localize a target in the Cartesian coordinates, the data from multiple transmitter-receiver pairs can be used. This task is, however, challenging due to the ambiguities in the measurements assignment. In the paper, a tracking algorithm is presented, which decomposes the complicated task of target localization into two stages: tracking in the bistatic domain and tracking in the Cartesian domain. The bistatic tracker is used only for plot-to-plot association. The Cartesian tracker, based on the extended Kalman filter, uses the raw plots associated by the bistatic tracker to calculate the Cartesian parameters of the target.

Keywords: passive coherent location, passive bistatic radar, target tracking, extended Kalman filter.

1 Introduction

In the paper, a passive coherent location (PCL) radar based on FM-radio transmitters is considered. The PCL radar measures the target parameters in the bistatic coordinates: the time difference of arrival (proportional to the bistatic range) and the Doppler shift (proportional to the bistatic velocity). One can determine the target position in the Cartesian coordinates by calculating the intersection of the bistatic ellipse with the direction of arrival (DOA), as in [1]. In this paper, we assume that DOA is not available. An alternative solution is to use the data from multiple transmitter-receiver pairs (here one receiver and multiple transmitters are assumed) to find intersection of the bistatic ellipses (or ellipsoids in 3D). The tracking algorithms for passive radar based on the extended Kalman filter (EKF) and particle filter were presented in [2], and based on probability hypothesis density (PHD) in [3].

In this paper, we present a two-stage tracking algorithm. First, the plots corresponding to different transmitters are used by separate bistatic trackers. In this way, the false measurements are separated from true target detections. The second stage is based on the EKF. The filter uses the raw plots associated by the bistatic tracker. In this way, the problem of false measurements is easily eliminated, and the EKF works only with the true measurements. We present two structures of the EKF, with parallel and sequential updating.

A similar approach of multi-stage algorithm was presented in [4] in the context of target tracking using a single frequency network of DAB/DVB-T transmitters. The algorithm presented in this paper is different since we calculate intersections of bistatic ellipsoids in 3D – in this way, the number of ghost targets is reduced.

Section 2 describes the relationship between the Cartesian and bistatic parameters as well as the tracking in the bistatic coordinates. In section 3, the Cartesian tracker based on EKF is presented. Section 4 shows how the two stages of the tracking are combined. Some numerical results are presented in section 5. The paper ends with conclusions.

2 Bistatic tracking

The instantaneous bistatic range (proportional to the time difference of arrival) can be calculated from:

\[
r(t) = \sqrt{(x-x_t)^2 + (y-y_t)^2 + (z-z_t)^2} + \sqrt{(x-x_r)^2 + (y-y_r)^2 + (z-z_r)^2} - r_b,
\]

where \((x, y, z)\) is the target position, \((x_t, y_t, z_t)\) is the transmitter position, \((x_r, y_r, z_r)\) is the receiver position, and \(r_b\) is the base line length (distance between transmitter and receiver).
The instantaneous bistatic velocity (proportional to the Doppler shift) calculated as the first derivative of the bistatic range has the following form:

\[
v(t) = \frac{(x - x_t)w_x + (y - y_t)w_y + (z - z_t)w_z}{\sqrt{(x - x_t)^2 + (y - y_t)^2 + (z - z_t)^2}}, \quad (2)
\]

where \((w_x, w_y, w_z)\) is the vector of target velocities. In the sequel, the equations (1) and (2) will be used for constructing the EKF.

For the purpose of the tracking in the bistatic coordinates, the bistatic range can be approximated by a second-degree polynomial:

\[
r(t) = R + Vt + \frac{At^2}{2}, \quad (3)
\]

where \(R\) is the bistatic range, \(V\) is the bistatic velocity and \(A\) is the bistatic acceleration.

One can construct a tracker operating in bistatic coordinates based on (3) using an almost-constant-acceleration model [5, 6]:

\[
x_b(k) = F_b x_b(k-1) + v_b(k), \quad (4)
\]

where \(x_b(k) = [R(k), V(k), A(k)]\),

\[
F_b = \begin{bmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}, \quad (5)
\]

and \(v_b(k)\) is uncorrelated Gaussian process noise with covariance:

\[
Q_b = E[v_b(k)v_b'(k)] = \begin{bmatrix} \frac{T^4}{4} & \frac{T^3}{2} & \frac{T^2}{2} \\ \frac{T^3}{2} & T^2 & T \\ \frac{T^2}{2} & T & 1 \end{bmatrix}, \quad (6)
\]

The measurement is modeled as:

\[
z_b(k) = H_b x_b(k) + w_b(k), \quad (7)
\]

where \(H_b = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}\), \(w_b(k)\) is uncorrelated Gaussian measurement noise with covariance:

\[
R_b = E[w_b(k)w_b'(k)] = \begin{bmatrix} \sigma_k^2 & 0 \\ 0 & \sigma_v^2 \end{bmatrix}, \quad (9)
\]

Based on (3) and (4), standard linear Kalman filtering equations can be used [5, 6].

3 Cartesian tracking

As can be seen from (1) and (2), the relationship between bistatic and Cartesian parameters is nonlinear. One of the algorithms that can be applied in this situation is EKF, which uses linear approximations of the nonlinear equations.

Let us assume that state vector in Cartesian coordinates is:

\[
x_c(k) = [x(k), v_x(k), v_y(k), v_z(k), y(k), v_y(k), z(k), v_z(k)]'. \quad (8)
\]

The state evolution can be described by:

\[
x_c(k) = F_c x_c(k-1) + v_c(k), \quad (10)
\]

where \(F_c = diag(F,F,F)\), with

\[
F = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad (11)
\]

and \(v_c(k)\) is uncorrelated Gaussian process noise with covariance \(Q_c = diag(\sigma_w^2 Q, \sigma_w^2 Q, \sigma_w^2 Q)\) where

\[
Q = \begin{bmatrix} T^2 / 2 & T \\ T & 1 \end{bmatrix}. \quad (12)
\]

The measurement corresponding to the \(m\)-th transmitter is modeled as:

\[
z_c^m(k) = h^m(x_c(k)) + w_c^m(k), \quad (13)
\]

where \(z_c^m(k) = \begin{bmatrix} \tilde{r}^m(k), \dot{V}^m(k) \end{bmatrix}\) is the measurement vector, and \(h^m(\cdot)\) is a nonlinear function transforming the Cartesian parameters to bistatic parameters according to equations (1) and (2). Note, that the measurement noise is modeled here directly in the bistatic domain. The covariance of the measurement noise is \(R_b\) from equation (9).

In the EKF, standard Kalman equations are used, but the state-to-measurement matrix is calculated as a
linearization of the function \( h^m(\cdot) \) at the predicted position \( \hat{x}(k | k - 1) \) with respect to state vector elements:
\[
H^m_{[i,j]}(k) = \frac{\partial h^m(\hat{x}(k | k - 1))}{\partial x_{[i]}}.
\] (14)

The prediction in the considered case has the following form:
\[
x_c(k | k - 1) = F_c x_c(1 | k - 1),
\] (15)
\[
P_c(k | k - 1) = F_c P_c(k | k - 1) F_c' + Q_c,
\] (16)
where \( P_c(k | k - 1) \) is the a priori state covariance matrix and \( P_c(k | k - 1) \) is the a posteriori covariance matrix.

There are two methods for updating the estimate in a multi-sensor scenario: parallel and sequential [6]. In the parallel method, the measurements \( z_b^m(k) \), and the matrices \( H^m(k) \) are stacked and the update is performed in one step. In the sequential method, the state vector and covariance are updated with measurements corresponding to each transmitter separately. When the system is linear, the both methods are equivalent, however, the first is more computationally expensive (it requires operations on larger vectors and matrices). If the system is nonlinear, as in the considered case, one should start with the most accurate measurement to minimize the linearization errors. Here, we assume that the measurements corresponding to all transmitters have similar accuracies, and the order of sequential updating is chosen arbitrarily.

The two methods will be shortly described below. We focus on the use of three transmitters.

### 3.1 Parallel updating

The measurement vector as well as the state-to-measurement matrix is stacked to form the quantities corresponding to all three transmitters:
\[
z(k) = \begin{bmatrix} z_1^m(k) \\ z_2^m(k) \\ z_3^m(k) \end{bmatrix},
\] (17)
\[
H(k) = \begin{bmatrix} H^1(k) \\ H^2(k) \\ H^3(k) \end{bmatrix}.
\] (18)

The covariance matrix of the measurement error for all transmitters is also created from matrices for separate transmitters:
\[
R = \text{diag}(R_b, R_a, R_b).
\] (19)

The update equations are as follows:
\[
S(k) = H(k)P_c(k | k - 1)H'(k) + R,
\] (20)
\[
K(k) = P_c(k | k - 1)H'(k)S^{-1}(k),
\] (21)
\[
\hat{x}(k | k) = \hat{x}(k | k - 1) + K(k)(z(k) - h(\hat{x}(k | k - 1))),
\] (22)
\[
P_c(k | k) = (I - K(k)H(k))P_c(k | k - 1).
\] (23)

In the parallel updating, we assume a fixed structure of the filter: data from three transmitters are expected. In the case of lack of target detection corresponding to one of the transmitters, we use the predicted value from the bistatic tracker, however, the measurement covariance matrix \( R_b \) corresponding to the appropriate transmitter is set to a large value. In this way we inform the EKF that this measurement is not reliable.

### 3.2 Sequential updating

In the sequential method, the state estimate and covariance are updated by each measurement separately. First, temporary variables are created:
\[
\hat{x}^0(k | k - 1) = \hat{x}(k | k - 1),
\] (24)
\[
P_c^0(k | k - 1) = P_c(k | k - 1).
\] (25)

Next, the sequential updating is performed for each of the \( M \) measurements:

for \( m = 1, \ldots, M \)
\[
S^m(k) = H^m(k)P_c^{m-1}(k | k - 1)H'^m(k) + R_b,
\] (26)
\[
K^m(k) = P_c^{m-1}(k | k - 1)H'^m(k)S^{-1}(k)H^m(k),
\] (27)
\[
\hat{x}^m(k | k - 1) = \hat{x}^{m-1}(k | k - 1) + K^m(k)(z^m(k) - h^m(\hat{x}^{m-1}(k | k - 1))) + \left( I - K^m(k)H^m(k) \right)P_c^{m-1}(k | k - 1).
\] (28)
The updated state estimate and covariance matrix are:

\[
\hat{x}(k | k) = \hat{x}^M(k | k-1),
\]

\[
P_k(k | k) = P_c^M(k | k-1).
\]

(30)
(31)

If the detection corresponding to one of the transmitters did not occur, the updating is performed only for existing measurements. In this way, a universal algorithm is obtained, which can be easily extended to a larger number of transmitters.

An additional advantage of the sequential approach is that the updating is performed separately for each of the transmitters, which decreases the number of possible associations. If \( N_m \) denotes the number of bistatic measurements for \( m \)-th transmitter, the sequential updating requires \( \sum_m N_m \) association tests. In the case of parallel updating, \( \prod_m N_m \) tests are required, since all possibilities of combined measurement vector \( z(k) \) from (17) have to be checked.

### 4 Tracking algorithm

We propose the following tracking procedure based on the presented algorithms. For each transmitter, the plots from the extractor blocks are fed to the separate bistatic trackers. The bistatic tracks are initialized using simple \( M/N \) cascaded logic [7]. When a bistatic track is confirmed, it is used by a Cartesian track initialization procedure. The procedure calculates intersection of the bistatic ellipsoids – if three ellipsoids (corresponding to three transmitters) intersect, they are regarded as originating from the same new target. Having position estimate, the velocity vector can be calculated from the bistatic velocity measurements. The algorithm for calculating Cartesian position and velocity vector is presented in [8] and it is similar to the one from [9], adopted to the passive radar scenario. Since we are using full 3D estimation, the number of ghost targets (intersections of ellipsoids not corresponding to the same target) is much lower than in the 2D case. At this stage, the bistatic tracks are associated and a Cartesian track is created.

In the following steps, the bistatic tracks from 3 trackers are used to update the Cartesian track (using either parallel or sequential updating). However, we use the raw plots from each of the (confirmed) bistatic tracks and not the state estimates. The reason for this is the following. Usually, it can be assumed that the error of raw measurement is uncorrelated – therefore, the assumption made in the Kalman filter is fulfilled. After the bistatic tracker, the error of the state estimate is highly correlated. Therefore, using the bistatic state estimate in the EKF would lead to suboptimal performance. For this reason, the bistatic tracker serves only as a tool for data association and false alarms rejection. The block diagram of the tracking algorithm is shown in Fig. 1.

![Block diagram of the tracking algorithm](image)

**5 Simulations**

In the first experiment, three targets were simulated. The target trajectories were generated using the assumed model (10) with \( \sigma_{w_x} = \sigma_{w_y} = 1 \text{ m/s} \) and \( \sigma_{w_z} = 0.1 \text{ m/s} \).

The receiver \( R_x \) was located at \((0, 0, 0) \text{ km}\). The transmitters \( T_x1, T_x2 \) and \( T_x3 \) were located at \((42, 40, 0) \text{ km}\), \((-20, 0, 0) \text{ km}\) and \((10, -40, 0) \text{ km}\), respectively. The refresh interval \( T \) was equal to 1 s. The target trajectories, the transmitters and the receiver are shown in Fig. 2.

![Target trajectories in Cartesian coordinates](image)
Fig. 3. True, measured and tracked bistatic parameters versus time for transmitter Tx1 and target 1.

Fig. 4. True, measured and tracked bistatic parameters versus time for transmitter Tx2 and target 1.

Fig. 5. True, measured and tracked bistatic parameters versus time for transmitter Tx3 and target 1.

Fig. 6. True and tracked Cartesian coordinates versus time for target 1 (parallel updating).

Fig. 7. Real and filter-calculated $x$, $y$, and $z$ position accuracies versus time for target 1 (parallel updating).

Fig. 8. Real and filter-calculated $x$, $y$, and $z$ position accuracies versus time for target 1 (sequential updating).
Fig. 9. Real and filter-calculated \( x, y, \) and \( z \) velocity accuracies versus time for target 1 (parallel updating).

Figs. 3, 4 and 5 show true, measured and tracked bistatic parameters corresponding to one of the targets (target 1). The error of bistatic measurements can be clearly seen, especially for the bistatic range. This is the result of low bandwidth of the FM signal (approx. 100 kHz), which leads to low range resolution (app. 3 km resolution cell).

As can be seen, the Kalman filters working in the bistatic coordinates increase the accuracy of bistatic range measurement significantly. However, as was indicated earlier, the bistatic trackers are used only for data association – the EKF uses the raw plots and not the state estimate of the bistatic trackers.

Fig. 6 shows the true and tracked Cartesian coordinates of target 1 versus time using parallel updating. In can be seen that parameters in \( x \) and \( y \) coordinates are tracked with high accuracy. The estimate of the \( z \) coordinate is characterized by a large error, which decreases slowly in time. This is a well known phenomenon – due to the geometrical relationships, the accuracy of the height estimation is worse than the accuracy of \( x \) and \( y \) coordinates.

In Fig. 7, a comparison of the theoretical (filter-calculated) and the actual position errors \( \Delta \_x, \Delta \_y \) and \( \Delta \_z \) for parallel updating is presented. The real error was calculated as a standard deviation of the difference between true and tracked position coordinate. Filter-calculated accuracy was the averaged value of the appropriate element of the \( P(k|k) \) matrix. As could be inferred from the previous figure, the accuracy of the height estimation is worse than the accuracy of \( x \) and \( y \) coordinates. In the case of \( x \) and \( y \) coordinates accuracy of hundreds of meters can be expected, which decreases to tens of meters during the course of tracking. The initial height accuracy is over 1 km and decreases to hundreds of meters. One can also observe that the actual and theoretical accuracies are very similar. This would not be true if the state estimates from the bistatic tracker were used in EKF – the filter would calculate the covariance too optimistically because of the correlation of the input measurements.

Fig. 8 shows the same position errors obtained using the sequential updating approach. As can be seen, the results are very similar to the structure with parallel updating. It is worth mentioning that using only bistatic measurements, without DOA information, it is possible to achieve 3D tracking with acceptable accuracy, not worse than in classical active radars.

The actual and theoretical errors corresponding to the three velocity components \( \Delta \_Vx, \Delta \_Vy \) and \( \Delta \_Vz \) are shown in Fig. 9. In this case, inferior accuracy in \( z \) dimension is also visible.

In order to compare the proposed two-stage algorithm with a one-stage tracker using the EKF only, the following simulation was carried out. Three targets were simulated and their bistatic parameters were calculated. The probability of detection was equal to 1. Apart from the target-originated detections, false alarms were added to the bistatic measurements. The false detections were generated uniformly on the bistatic range – bistatic velocity surface. It was assumed that the crossambiguity of the reference and surveillance signals is calculated for bistatic ranges (0, 300) km and bistatic velocities (–400, +400) m/s. The bistatic range resolution cell was 3 km and the bistatic velocity resolution cell was 2 m/s. As a result, 100 range resolution cells and 400 velocity resolution cells were obtained – 40000 cells in total. The number of false detections was generated according to the Poisson distribution based on the number of the resolution cells and the probability of false alarm \( P \_fa \). In the one-stage tracker, the raw bistatic plots were used for Cartesian track initialization. In the two-stage algorithm, the bistatic trackers used 3/3 cascaded initialization logic. Only the bistatic plots corresponding to the initialized bistatic tracks were used by the Cartesian track initialization procedure. In both cases, one- and two-stage algorithm, a Cartesian track was created for each output of the ellipsoid intersection algorithm, which tests every possible combination of the bistatic measurements in search for simultaneous ellipsoid intersection. The simulation was repeated 1000 times.

The results of the comparison are listed in tables I and II for different values of \( P \_fa \). The first table shows the average number of combinations of bistatic measurements which have to be checked by the Cartesian track initialization procedure. The number of the possibilities which the algorithm has to test is \( \prod \_m N \_m \), where \( N \_m \) is the number of bistatic measurements corresponding to the \( m \)-th transmitter. As can be easily seen, the number of possibilities to test grows very fast with \( N \_m \). In the one-stage tracker, \( N \_m \) is equal to the number target-originated measurements plus the number of false detections. In the
two-stage tracker, \( N_m \) is very close to the number target-originated measurements – random false detections cause bistatic track initialization very rarely.

The values in table II indicate the average number of “ghost Cartesian tracks” – Cartesian tracks initialized by ghost targets. The results show that the number of ghost tracks in the case of two-stage tracker is reduced, especially in the case of high \( P_{fa} \). It results from the fact that bistatic trackers in the two-stage algorithm eliminate the false detections almost completely, and the probability of random intersection of the bistatic ellipsoids is lower. The remaining ghost targets result from the random intersections of the bistatic ellipsoids corresponding to the true targets.

\[
\begin{array}{|c|c|c|}
\hline
P_{fa} & \text{One-stage tracker} & \text{Two-stage tracker} \\
\hline
10^{-3} & 74975.38 & 161.92 \\
10^{-4} & 321.31 & 27.05 \\
10^{-5} & 41.28 & 27.00 \\
\hline
\end{array}
\]

Tab. I. A comparison of the average number of possible combinations of bistatic measurements for one- and two-stage algorithm.

\[
\begin{array}{|c|c|c|}
\hline
P_{fa} & \text{One-stage tracker} & \text{Two-stage tracker} \\
\hline
10^{-3} & 212.90 & 0.92 \\
10^{-4} & 1.40 & 0.32 \\
10^{-5} & 0.43 & 0.31 \\
\hline
\end{array}
\]

Tab. II. A comparison of the average number of ghost tracks for one- and two-stage algorithm.

The disadvantage of the two-stage approach is the time delay associated with the process of bistatic track initialization. The Cartesian track is created only after the bistatic tracks corresponding to all transmitters are initialized. In the case of \( M/N \) initialization logic, at least \( M \) observations are required to confirm a track. If the probability of detection is low, the average number of observations needed for track confirmation may be much larger.

The computational complexity of additional bistatic trackers in the two-stage approach is compensated by the reduced number of ellipsoid intersection tests due to the false detections elimination.

6 Conclusions

In the paper, a tracking algorithm for passive radar was presented. The approach proposed here is to use a two-stage tracking: one in the bistatic coordinates and second in the Cartesian coordinates. The first stage is used only for data association – the second stage is fed with the raw plots. In this way, the false measurements are (almost completely) eliminated by the bistatic trackers. This leads to reduced number of ellipsoid intersection tests and mitigates the ghost target phenomenon. As a result, the association problem for the Cartesian tracker using extended Kalman filter is greatly reduced.

We compared two updating schemes for the extended Kalman filter: parallel and sequential. The latter requires less computations and is more universal – number of used transmitters can be easily extended without changing the structure of the filter. The results obtained with two structures are very similar.

7 Acknowledgements

This work has been partially supported by the Polish Ministry of Science and Higher Education from the science budget for the years 2007-2010 under Commissioned Research Project PBZ-MNiSW-DBO-04/I/2007 and by the European Union in the framework of European Social Fund through the Warsaw University of Technology Development Program.

References