14th International Conference on Information Fusion
Chicago, Illinois, USA, July 5-8, 2011

Track-Before-Fuse Error Bounds for Tracking Passive Targets

Ben Shapo
Integrity Applications Incorporated
900 Victors Way, # 220, Ann Arbor MI 48108
bshapo@integrity-apps.com

Chris Kreucher
Integrity Applications Incorporated
900 Victors Way, # 220, Ann Arbor MI 48108
ckreucher@integrity-apps.com

Abstract[1] ***- In tracking applications, data fusion involves combining data from multiple sensors to produce lower error target state estimates than were possible from using data from individual sensors alone. The main focus of this paper is performance modelling and evaluation for two different fusion paradigms: the track-before-fuse (or track fusion) approach and the fuse-before-track approach. Track-before-fuse is often required in distributed, low-bandwidth environments, while fuse-before-track is an option when centralization is possible. It is anticipated that fuse-before-track will provide superior performance over track-before-fuse because it postpones hard decision making until the most data is received, analogous to track-before-detect. This paper computes approximate error bounds for a track-before-fuse approach and verifies them via simulation. It also shows that a fuse-before-track approach out-performs a standard track-before-fuse method.***

Keywords: Fusion performance modeling, nonlinear filtering, track-before-fuse, fuse-before-track, passive acoustics.

1 Introduction

Automated target tracking is an important task in many civilian and military systems. Fusing data from multiple sensors has the potential to improve tracking performance in many of these systems. The traditional approach for fusing data involves performing tracking functions locally at each sensor, followed by fusing these tracks. An alternative approach is fuse-before-track [1], in which a centralized processor combines data from each sensor prior to processing. This paper compares these two approaches, focusing on the passive acoustics tracking scenario where the only measurements available are target bearings.

Traditional fusion systems (e.g., [2]-[4]) execute their fusion steps by associating tracks that have been formed locally (at each sensor). While this approach is sub-optimal, it is sometimes required due to system limitations (such as limited communications between sensors). In the application presented here, required communication bit rates are low because the data comprises only a collection of amplitudes for a small number of bearing measurements at each update. Furthermore, applying this approach to two sensors on the same vessel requires only that the two sensors be connected on the vessel. Thus, full fuse-before-track approaches are feasible in our scenario.

The literature contains other fusion approaches that do not require explicit track association. Included are Bethel's PDF Tracker [5], Stone's Unified Data Fusion [6], Kastella's JMPD [7], and others [8]-[10].

Papers specific to track-before-fuse approaches are also common. Examples include [11]-[16]. In these papers, the authors investigate techniques for both associating tracks across different sensors, and target state estimates given the (associated) individual-sensor tracks.

This work contains two main contributions. First, we derive an analytic formulation for tracking error from a track-before-fuse approach. Second, we compare these bounds with a fuse-before-track method and illustrate the latter's benefit. Its main innovations are thus providing analytical error predictions for track-association and comparing these to a fuse-before-track method.

This paper proceeds as follows. Section 2 describes the notional passive sonar scenario and localization equations. Section 3 derives the performance models for our scenario. Section 4 compares these results to a fuse-before-track method. Finally, we conclude in Section 5.

2 The Passive Sonar Scenario

This section outlines the notional scenario for the work presented in the following sections, and provides an example simulated data surface.

2.1 Notional Scenario

Figure 1 defines the notional scenario for our experiment. Here, a ship platform points local north. The ship has access to two sensing arrays, a Hull Array and a Towed Array (both indicated by text and arrows in the figure). The ship tracks a target (also indicated by a text label in the figure). For convenience (but with no loss of generality in a 2D scenario), we place the coordinate

[1] This work was sponsored by NAVSEA contract N00024-10-C-5226

system origin at the Hull Array ("HA"). The x-axis is local east, and the y-axis is local north. In our rigid body model, the Towed Array ("TA") phase center is T meters behind the boat, thus its coordinates are (0,-T). The target is at an external point (x, y). The angles θ_{HA} and θ_{TA} are defined from the sensors to the target counter-clockwise from the x-axis.

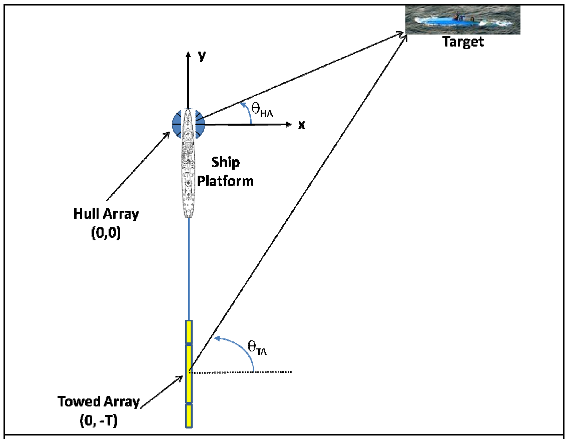

Figure 1: Geometric scenario for Fusion Algorithm

Because we model the passive sonar scenario, inputs to the tracker are bearings-only measurements, over time. Thus, one of the well-known challenges in sonar signal processing is estimating target range from passive sensors ("passive ranging").

2.2 Data Surface

Figure 2 gives an example of the data surfaces from the Hull Array (left-hand panel) and the Towed Array (right-hand), respectively. In both cases, the horizontal axis corresponds to bearing (relative to the sensor) and the vertical axis corresponds to time (most recent time at the bottom). Dark regions indicate low intensity pixels, and white pixels indicate areas with significant energy. Red arrows indicate the target position (bearing) on both surfaces. In the current figure, we see a single trace on each data surface, indicating a single target's trajectory over time.

The hydrophones on the HA are linearly spaced in angle around the array's surface, so the natural coordinate system for this array's bearing axis linearly spans the space from zero to 360°. In contrast, the hydrophones on the TA are linearly spaced in a straight line, so the natural coordinate system for this array is sin(θ_{TA}) [17].

Given these geometries, zero degrees on the HA corresponds to a target directly east of the HA sensor, 90° on the HA represents a target north of the ship, and so on. On the TA, sin(θ_{TA}) of -1 indicates a target to the south of the TA and sin(θ_{TA}) of 0 shows a target east of the TA.

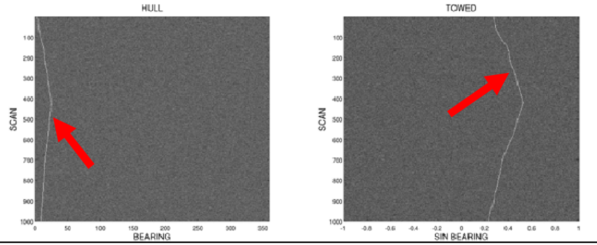

Figure 2: Input Data Surfaces for Fusion & Tracking

See Figure 3 for examples (with reduced numbers of beams for clarity). The left-hand panel shows the HA beam configuration, with beams uniformly spaced in azimuth. The right-hand panel shows the TA beam configuration, with uniform beam spacing in sin(θ), therefore non-uniform beam spacing in θ.

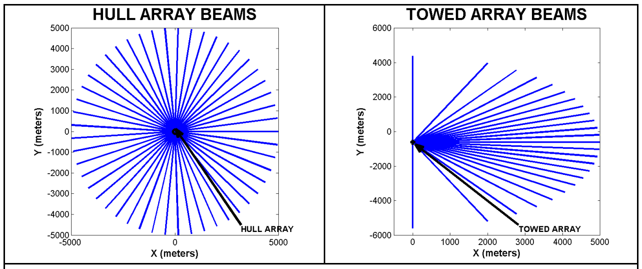

Figure 3: Uniformly distributed HA beams (LHS), Cosine spaced TA beams (RHS)

In the current problem, we assume we have estimates of target bearing from both the Hull Array ("HA") and the Towed Array ("TA") from the data. Of course, perfect bearing estimates at both sensors will yield perfect 2D localization via geometry. As mentioned above, one of the challenges in passive sonar is obtaining good estimates of target range. The work presented here helps us better understand the impact of bearing measurement error on estimated target range.

First, from the geometry (Figure 1) the 2D target position must satisfy the following relations:

$$y = y_{HA} + x\tan\theta_{HA} = x\tan\theta_{HA} \qquad (1)$$

$$y = y_{TA} + x\tan\theta_{TA} \ . \qquad (2)$$

Thus we can solve for the two unknowns x and y by using this equation:

$$x = \frac{T}{\tan(\theta_{TA}) - \tan(\theta_{HA})} \qquad (3)$$

$$y = \frac{T\tan(\theta_{HA})}{\tan(\theta_{TA}) - \tan(\theta_{HA})} \ . \qquad (4)$$

In the situation where θ_{HA} and θ_{TA} are measured, we can thus write the estimated contact position as a function of the measured angles $\hat{\theta}_{HA}$ and $\hat{\theta}_{TA}$ as:

$$\hat{x} = \frac{T}{\tan(\hat{\theta}_{TA}) - \tan(\hat{\theta}_{HA})} \qquad (5)$$

$$\hat{y} = \frac{T\tan(\theta_{HA})}{\tan(\hat{\theta}_{TA}) - \tan(\hat{\theta}_{HA})}\,, \qquad (6)$$

and the estimated range as:

$$\hat{r} = \frac{T\tan(\hat{\theta}_{HA})}{\tan(\hat{\theta}_{TA}) - \tan(\hat{\theta}_{HA})}\sec(\hat{\theta}_{HA})\,. \qquad (7)$$

3 Baseline Approach and Analytics

This section describes our analytical approach to estimating localization error from scene geometry.

3.1 Analytic Derivation of Range Error Sensitivity

With the previous equation describing target range as a function of measured target bearings, we can directly compute estimated passive ranging error as a function of the measurements $\hat{\theta}_{HA}$ and $\hat{\theta}_{TA}$. Define the error in range estimate as:

$$r_{err} = \hat{r} - r_{true}\,, \qquad (8)$$

and note

$$\frac{\partial r_{err}}{\partial \theta} = \frac{\partial \hat{r}}{\partial \theta}\,. \qquad (9)$$

We compute the expected range error $\Delta\hat{r}$ using a first order Taylor series. In the HA case, the relationship is:

$$\Delta\hat{r} \approx \frac{\partial \hat{r}}{\partial \hat{\theta}_{HA}}\Delta\hat{\theta}_{HA}\,. \qquad (10)$$

This linearized relationship means that we can estimate range error if we know the derivative of the estimated range with respect to azimuth and if we know the likely azimuthal error (i.e., uncertainty on the sensor data surface). The analytic relationship we have between estimated range and measured bearings allows us to take the continuous partial derivative. The process is straightforward and yields the rate of change of the range error (i.e., the sensitivity) with respect to the estimate $\hat{\theta}_{HA}$, under the assumption that θ_{TA} is correct:

$$\begin{aligned}\frac{\partial \hat{r}}{\partial \hat{\theta}_{HA}} &= \frac{\partial}{\partial \hat{\theta}_{HA}}\left[\frac{T\tan(\hat{\theta}_{HA})}{\tan(\theta_{TA}) - \tan(\hat{\theta}_{HA})}\sec(\hat{\theta}_{HA})\right] \\ &= \frac{T\sec(\hat{\theta}_{HA})}{\left(\tan(\theta_{TA}) - \tan(\hat{\theta}_{HA})\right)^2}\left(1 + \tan(\theta_{TA})\tan(\hat{\theta}_{HA})\right)\end{aligned} \qquad (11)$$

A similar pair of relationships holds for the TA:

$$\Delta\hat{r} \approx \frac{\partial \hat{r}}{\partial \hat{\theta}_{TA}}\Delta\hat{\theta}_{TA} \qquad (12)$$

and

$$\begin{aligned}\frac{\partial \hat{r}}{\partial \hat{\theta}_{TA}} &= \frac{\partial}{\partial \hat{\theta}_{TA}}\left[\frac{T\tan(\theta_{HA})}{\tan(\hat{\theta}_{TA}) - \tan(\theta_{HA})}\sec(\theta_{HA})\right] \\ &= \frac{-T\sec(\theta_{HA})}{\cos^2(\hat{\theta}_{TA})\left(\tan(\hat{\theta}_{TA}) - \tan(\theta_{HA})\right)^2}\,.\end{aligned} \qquad (13)$$

3.2 Range Error for the Model Problem

We begin estimating range error with a simple case. We assume all measurement bearing error derives from the HA data surface (no error from TA). We further make the assumption that the error is on the order of a beamwidth. In other words, we assume that target azimuth is known within the highest amplitude beam on the HA data surface, but that target azimuth is unknown within that beam. Recall from Figure 3 that, in the case of the HA, beam widths (in radians or degrees) are uniform. However, the analytic derivative of estimated range with target azimuth still depends on target location via the trigonometric quantities in the target range expression.

Our goal is a 2D estimate of range error as a function of target location. This computation requires selection of parameters. In Figure 4, the array phase centers are separated by 2000 feet. Assume 400 beams on the HA (spaced linearly) and 801 on the TA. From Figure 3, these beams are spaced linearly in *sin*(θ_{TA}).

The horizontal and vertical axes of the figure represent target position in real space (e.g., East and North) in meters. The legend indicates the symbols that represent the HA and TA positions (separated by 2000 feet). The color at each pixel represents error amplitude at that target location in 2D (x, y) space, and the color bar on the figure's right side shows the relationship between colors and error absolute value.

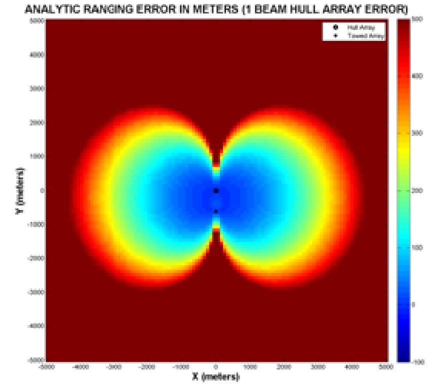

Figure 4: Range error due to Hull Array azimuthal Error

The figure is in agreement with basic intuition that localization of near-range broadside targets has the lowest error, with error smoothly increasing at greater ranges and closer to endfire. Note that endfire performance is worse for at least two reasons in this scenario. First, TA beams are widest in this direction (refer back to the RHS of Figure 3), providing the poorest azimuthal resolution (greatest $\Delta\theta_{TA}$ in terms of the symbols above). In addition, endfire has the smallest effective aperture for the two incoherently combined arrays (the triangulation), and therefore the most sensitivity.

Figure 5 shows a comparable figure for TA error. All parameters are the same, except here the azimuthal error is now $\Delta\theta_{TA}$ instead of $\Delta\theta_{HA}$, i.e., the error source is now one beamwidth on the TA. Here we see the same characteristic behavior of best performance at broadside and worst performance at endfire. Comparing to Figure 4 shows that broadside performance for 1-beam TA error is better than that for 1 beam HA error. This is as expected because the TA has more aperture and therefore better resolution at broadside (which fully utilizes the whole TA aperture) than the HA.

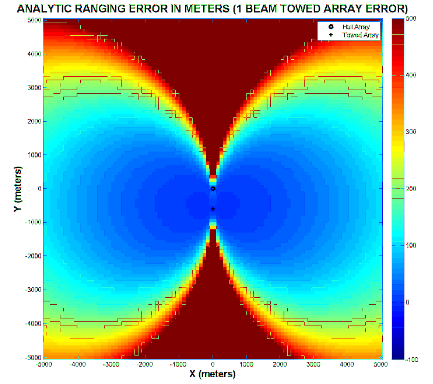

Figure 5: Range error due to Towed Array azimuthal Error

These calculations allow us to make some comments about the region of applicability of passive ranging techniques. Under the physical assumptions here of a 2000 foot displacement between array phase centers, and assumed 0.9° beamwidth on the HA, and assumed 0.3° beamwidth (at broadside) on the TA, we see that the localization error from a 1-beam HA error exceeds 500m at about 3km of target range and then grows dramatically with increasing range.

This error can be reduced by finer estimation of target bearing, e.g., by techniques that use temporal averaging and kinematic modeling. Therefore we can judge the value of a particular passive localization algorithm by measuring its improvement with respect to the analytic analysis of a measurement-only track fusion (i.e., track association algorithm), assuming the 1-beam error presented here.

3.3 Verification via discrete differencing Model

To verify the procedure above, we have computed discrete approximations to the derivatives for both the TA and the HA. We seek to determine an estimate for range error given a one beam bearing error on each sensor. Figure 6 illustrates the procedure. Here, black text indicates the two sensors. Solid blue lines show HA beam boundaries, and solid green represents TA beam boundaries. A black dot shows a hypothesized target location (for example, a point in the (x,y) space of Figure 4 or Figure 5).

Our target location falls into a beam pair for the HA and TA BTRs. The yellow, shaded region indicates all the area in (x,y) space that falls inside these beams. To approximate the error in range estimate for a one-beam error on the HA, we compute the difference in range for the two points at the center of the target's TA beam that intersect the HA beam boundaries. Red dots and red text labels indicate these two points (and the corresponding ranges from the HA origin). The range error for a one-beam HA error at this TA beam bearing is the difference between the ranges indicated by the red arrows R2 and R1.

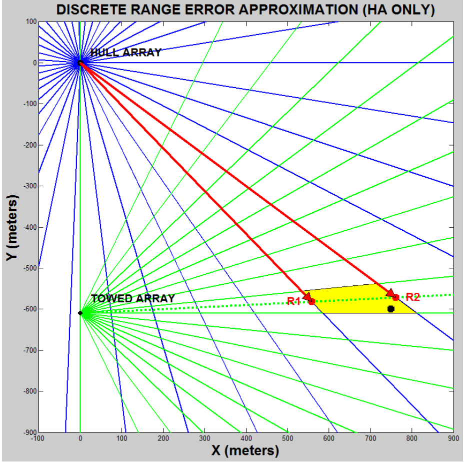

Figure 6: Approximation to error Estimate

Performing this computation over all the (*x, y*) points from Figure 4 and Figure 5 yields a discrete approximation to the errors represented in those figures. Figure 7 shows the results. Comparing these results to Figure 4 and Figure 5 shows excellent agreement, providing evidence that supports the validity of our analytic approach.

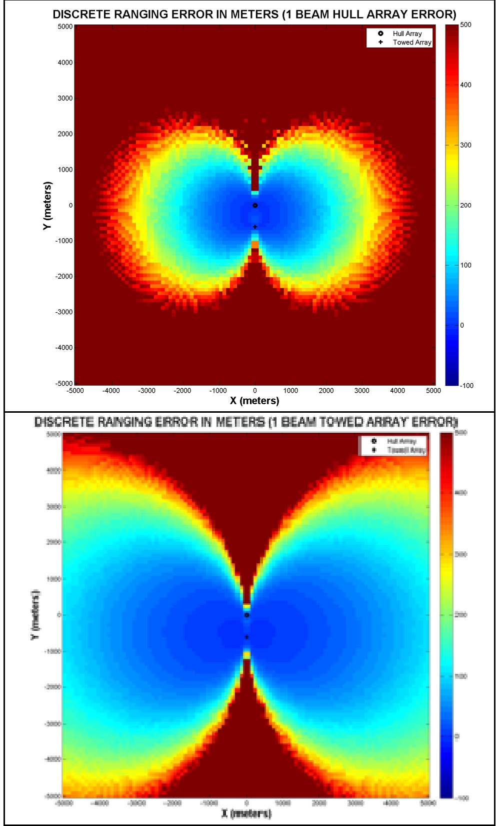

Figure 7: Range error due to HA (top) and TA (bottom) bearing Error

4 Comparison to Fuse-Before-Track

This section compares the performance of a fuse-before-track method to the track-before-fuse approach modeled earlier. We continue to use the passive acoustic scenario given in Section 2. We start by considering the performance of the two approaches against the single hypothetical target trajectory shown in Figure 8. In this example, a two-sensor platform moves north towing a rigidly attached array and a simulated target moves north-east. Refer back to Figure 2 to see the simulated data for the two sensor arrays.

4.1 Track-before-fuse Performance

Track-before-fuse methods estimate target bearing at each sensor independently using a single-sensor tracker, and then use geometry (specifically, Equation (5) and Equation (6)) to compute the target geo position. Perfect estimation of the bearing from the HA and TA, coupled with perfect association of tracks, results in perfect reconstruction of the target trajectory through geometry. However, in practice, estimates of single-sensor bearings are imperfect due to a number of factors, including closely spaced targets, false energy distracting a single-sensor tracker, and the beam size at each sensor (which is related to the element spacing).

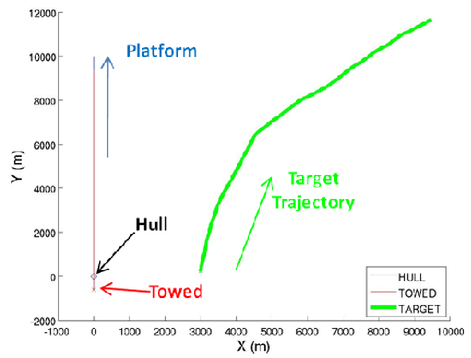

Figure 8: Simulated Ship, Towed Array, and Target Trajectory

Here we ignore the effects of both closely spaced targets and false energy distracting a single-sensor tracker and continue to focus on how the accuracy of track association fusion approaches are influenced by accuracy of the bearing estimate inside a beam. In our notional scenario, we assumed that the HA has 0.9° bearing beams and the TA has 0.3° bearing beams at broadside. A notional single-sensor that is at most 1 beam off in its estimate ("within 1 beam") can be described as having a worst case error of 1.35° in the HA (1.5 beams) and 0.45° in the TA at broadside. Likewise, consider a single-sensor tracker that perfectly estimates the correct bearing beam ("correct beam"), but is unable to determine the bearing with any more precision than within that beam. Such a tracker always estimates the target bearing as the center of the correct beam, and will have a worst-case error of 0.45° in the HA and 0.15° in the TA (at broadside). A higher-fidelity single-sensor tracker that is able to estimate with sub-beam accuracy to the half-beam ("1/2 beam accuracy") will have worst case-errors of 0.225° in the HA and .075° in the TA at broadside, and so on. A single-sensor tracker may achieve sub-beam accuracy through temporal integration or sidelobe models. However with a single-sensor tracker, models must operate in the measured data domain and so achieving significantly better than single-beam accuracy is optimistic.

Figure 9 parameterizes the RMS range estimation error of the track association algorithm in our notional scenario as a function of the accuracy of the measurements of the single-sensor bearing. In this computation, the "mean" refers to averaging over points on the target's trajectory over time. The x-axis indicates the accuracy of the single-sensor trackers in units of beams (finer accuracy to the right), and the y-axis shows RMSE. As tracker accuracy improves (estimates good to a smaller fraction of a beamwidth), RMSE decreases.

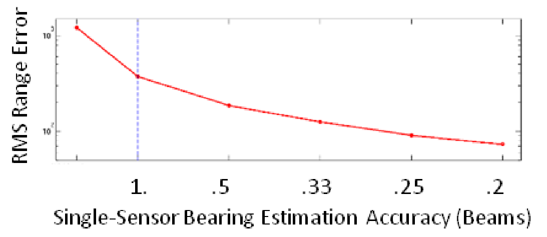

Figure 9: Range Error (Track-fusion Algorithm)

4.2 Track-before-fuse error vs. analytic Prediction

The analytic predictions of Section 3.1 are spatially varying and provide an estimate of the error if a single-sensor tracker is exactly one beam in all of its bearing estimates. Figure 10 compares the analytically predicted error estimates as a function of time in our simulation (since the position of the target in Figure 8 varies with time) with the actual range error achieved from our implementation of a track-association fusion algorithm. In this implementation, we assumed the single-sensor trackers were able to estimate the correct beam for the target. It shows that the analytic predictions are in close agreement with the actual track fusion results.

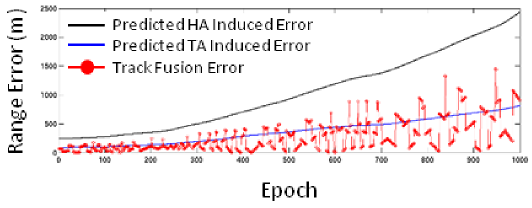

Figure 10: Track fusion error on target Postion

In the figure, the solid black line shows the predicted 2D tracker estimation error based on a one-beam single-sensor tracker error at the HA. Similarly, the solid blue line shows the predicted tracker error for a one-beam error at the TA. The red line shows the error for a track-association based fusion tracker, running on the simulated data from Figure 2 (which was derived, in turn, from the scenario in Figure 8).

The track fusion estimate error is highly oscillatory. At times where the actual target is close to the center of the HA and TA beams, the error is very low. Error increases rapidly as target bearing progresses towards the edge of the beam. This behavior is magnified as range increases, for example, near the end of the simulation. The analytic computations always assume a 1-beam error in one sensor and perfect estimation in the other sensor, which means they are approximations to the error found in practice with the tracker operating on synthetic data.

4.3 Fuse-before-track Performance

Fuse-before-track is expected to outperform track-before-fuse by delaying hard decisions until all information is received. The fuse-before-track algorithm we employed [1] on this data used the same inputs as those used in the track-before-fuse evaluation, where we again assume that the HA has 0.9° bearing beams and the TA has 0.3° bearing beams at broadside. Since a fuse-before-track approach is able to exploit a kinematic model in the coordinate system of the target (East/North) rather than being restricted to the coordinate system of the measurements, and is also able to use information from **both** sensors before making a target state estimate, it can provide higher fidelity results than the track association approach.

Some details concerning the fuse-before-track approach are as follows. We estimate the joint probability of target presence and the target's state using a two-step recursion [5]. The first step is a Bayesian measurement update, in which new measurements update the current density on target state via a likelihood ratio formulation [6]. The second is a (Markov) kinematic prediction that employs target motion models to project current densities to the next discrete time. When new data arrives at the next time, we perform the measurement update again, and this process continues recursively.

To implement these densities on target state, we need a numerical scheme to propagate spatial probabilities over time. We employ a fixed, evenly spaced, multidimensional grid [18], with discrete samples of the density at each grid point. The grid points span a 4D space, corresponding to position and velocity in the *XY* (or East/North) plane: $[x\ v_x\ y\ v_y]^T$. The Bayesian update refines the probabilities at the grid points as a function of the incoming measurements, and the kinematic prediction propagates the grid-sampled density to the next time step.

Note that inputs to the fuse-before-track approach are amplitudes from the data surfaces of Figure 2 at each time epoch. (This contrasts with the track-before-fuse method, which ingests single-sensor **tracker outputs** before fusing them via geometry.)

Figure 11 shows the track-error improvement of the fuse-before-track approach over the track-fusion method in our notional problem. Note that the range error from fuse-before-track processing (green line) is significantly less than the track-fusion error (red line) on average, and that we have left the analytic error predictions (black line and blue line) on the plot for reference.

Figure 12 shows fuse-before-track RMS range error compared to that achieved by track fusion as a function of single-sensor error (discussed in Section 4.1). The axes are the same as in Figure 9 (RMSE versus single-sensor tracker error). The green line indicates the RMSE of our fuse-before-track approach. Note this is a fixed quantity and does not depend on single-sensor tracking accuracy because the inputs to the fuse-before-track method are the energy surfaces of Figure 2, and not the single-sensor track outputs (thus, the fidelity of the single-sensor trackers does not impact track fusion performance). The red line shows RMSE for the track-fusion approach.

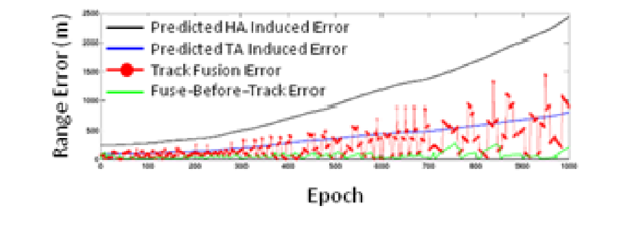

Figure 11: Fuse-before-track error vs. Time

[redacted]

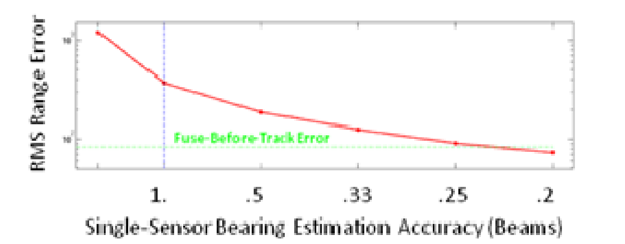

Figure 12: Fuse-before-track range Error

The figure shows where track fusion techniques that use a pair of single-sensor trackers (one for the HA and one for the TA) achieve the same performance as our fuse-before-track approach (intersection of the green line with the red line). The conclusion is that the track-fusion single-sensor trackers would have to estimate the target bearing with better than ¼ beam accuracy on each array to achieve the same performance as the proposed fuse-before-track technique. Therefore, a fuse-before-track method implicitly generates performance equivalent to single-sensor trackers with ¼ beam accuracy.

4.4 Monte Carlo Comparison

We extended the simulations above by running a series of Monte Carlo trials over target trajectories. In particular, we allowed the target trajectory to vary in East/North space over trials. With this approach to computing error statistics, we are now able to determine the expected ranging error as a function of space (like in Figure 7 above). This accumulation requires that we run a very large number of trials to guarantee that the target traverses all discrete cells in space at least once.

Figure 13 shows a Monte Carlo comparison of the track-before-fuse and fuse-before-track approaches, where we perform error averaging over target trajectories. In the top panel, we see the track-before-fuse error. We expect this to be a blend of the HA-error and the TA error (both from Figure 7). Comparing the results in Figure 13 with Figure 7 shows that Figure 13 does have the expected behavior (ranging error generally between that of the HA-only case and the TA-only case).

The bottom panel shows ranging error from our fuse-before-track approach. We see that the errors are much smaller over most cells in the surveillance region.

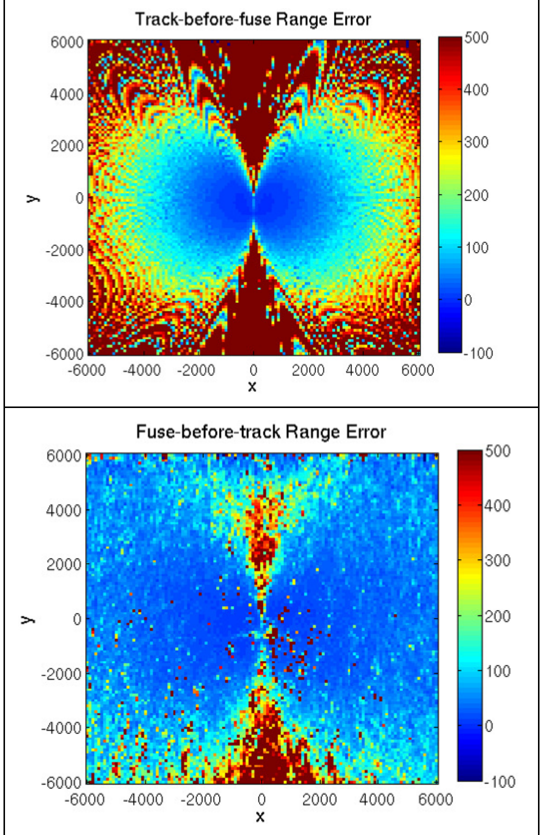

Figure 13: Monte Carlo comparison of (top) track-before-fuse and (bottom) fuse-before-track approaches

5 Conclusion

Traditional track fusion techniques operate by running trackers separately at each sensor and then fusing these single-sensor tracks for associated targets via a geometric localization step. In this paper, we have derived analytic expressions (based on scene geometry) that describe the state estimation errors that characterize such track-fusion methods. Comparing those results to outputs of a track-fusion implementation with simulated data produced strong agreement.

We also outlined a Bayesian fuse-before-track approach for tracking moving targets. The approach differs from traditional fusion methods because it fuses data from multiple sensors **prior** to tracking. Finally, we executed a fuse-before-track algorithm for comparison with the track-fusion approach. The fuse-before-track method out-performs the track-fusion method in terms of RMSE for realistic single-sensor tracker performance.

6 References

[1] C. Kreucher, B. Shapo, and R. Bethel, *Multitarget Detection and Tracking using Multi-Sensor Passive Acoustic Data*, Proc. IEEE Aerospace Conference, Big Sky, MT, April 2009.

[2] K. Chang, T. Zhi, R. Saha, *Performance Evaluation of Track Fusion with Information Matrix Filter*, IEEE Transactions on AES, vol. 38, no. 2, April 2002.

[3] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*. New York: Academic Press, 1988.

[4] S. Coraluppi and C. Carthel, *Recursive Track Fusion for Multisensor Surveillance*, Information Fusion, vol. 5, no. 1, pp. 23–33, March 2004.

[5] R. Bethel and G. Paras, *A PDF multisensor Multitarget Tracker*, IEEE Transactions on AES, vol. 34, no. 1, pp. 153–168, January 1998.

[6] L. Stone, C. Barlow, and T. Corwin, *Bayesian Multiple Target Tracking*. Boston: Artech House, 1999.

[7] K. Kastella, *Joint Multitarget Probabilities for Detection and Tracking*, in Proceedings of SPIE Acquisition, Tracking and Pointing XI, 1997.

[8] A. Srivastava, M. Miller, and U. Grenander, *Jump-diffusion Processes for Object Tracking and Direction Finding*, Proc. of 29th Allerton Conf. on Communication, Control, and Computing, 1991, pp. 563 – 570.

[9] E. Kamen, *Multiple Target Tracking Based on Symmetric Measurement Functions*, IEEE Transactions on Automatic Control, vol. 37, no. 3, pp. 371–374, 1992.

[10] R. Mahler, *A Unified Foundation for Data Fusion*, 7th Joint Service Data Fusion Symposium, 1994

[11] S. Mori, B. Barker, C. Chong, and K. C. Chang, *Track Association and Track Fusion with Non-Deterministic Target Dynamics*, Proc. IEEE Fusion 1999, pp. 231-238, July, 1999.

[12] G. Foster, *Analysis of Track Fusion using the Reduced State Estimator*, Proc. IEEE Fusion 2010, July 2010.

[13] X. Tian and Y. Bar-Shalom, *Exact algorithms for four track-to-track fusion configurations: All you wanted to know but were afraid to ask,* Proc. IEEE Fusion 2009, pp. 537-544, July, 2009.

[14] R. Canavan, C. McCullough, and W. Farrell, *Track-centric metrics for track fusion systems*, Proc. IEEE Fusion 2009, pp. 1147-1154, July, 2009.

[15] C. Yang and E. Blasch, *Track Fusion with Road Constraints*, Proc. IEEE Fusion 2007, July, 2007.

[16] H. Chen and X. R. Li, *On track fusion with communication constraints*, Proc. IEEE Fusion 2007, July 2007.

[17] H. Van Trees, *Detection, Estimation, and Modulation Theory IV: Optimum Array Processing*, 2002.

[18] K. Kastella and C. Kreucher, *Multiple model nonlinear filtering for low signal ground target applications*, IEEE Transactions on Aerospace and Electronic Systems, vol. 41, no. 2, pp. 549–564, April, 2005.