Predetection Fusion with Doppler Measurements and Amplitude Information

Ramona Georgescu, Student Member, IEEE and Peter Willett, Fellow, IEEE

Abstract

In previous work we discussed an efficient form of predetection fusion for use as a preprocessing step before tracking on datasets with large sensor networks of low quality sensors, with particular eye to application in multistatic sonar. This 2D version (position measurements only) was compared against an optimal (slow) technique. In this work, we present the 4D (using position and Doppler measurements) and 5D versions (using position, Doppler and also aspect-dependent SNR\(^1\) measurements) of predetection fusion. We demonstrate that improved results, in the sense of root mean square error (RMSE) and number of declared targets – and consequently better tracking results – are possible when Doppler measurements and SNR information are incorporated into our algorithm.

Index Terms

Predetection Fusion, Doppler, Amplitude Information, Sensor Networks.

I. INTRODUCTION

Data fusion in large sensor networks is expected to provide better target tracking capability in terms of increased area coverage, expanded geometric diversity\(^2\), increased target hold, robustness to sensor loss and jamming, improved localization and gains in probability of detection [3]. One of the many challenges these networks bring is a possibly much increased false alarm rate after an unwary fusion step.

\(^1\)SNR stands for signal to noise ratio.

\(^2\)Sonar measurements are dependent on the geometry created by the relative placement and velocities of source, target and receiver. More sensors should bring in, among other measurements, true target detections of better quality.
An example of such a sensor network is a multistatic sonar system which consists of multiple sonar sources and receivers distributed over the surveillance area. Transmissions from one or more sources may be processed by one or more receivers to produce a large number of sonar echo contacts [3]. In recent years, interest has shifted towards employing large surveillance networks that consist of many but cheap and limited-performance sensors [2]. The realistic multistatic sonar Metron dataset is representative and has served as our motivation [4].

From an implementation perspective, a tracker can be applied in two ways to a multistatic dataset that consists of N sets of contacts from N different sensors at one scan. In the scan based approach, the N sets of contacts are fed in turn to the tracker. In the fuse-before-track approach [2], all the contacts in a scan are gathered together in a single set of contacts to be fed to the tracker. In the latter approach, it is assumed that target originated contacts are more persistent over sensors than noise or clutter returns and thus performance of the tracker would be improved [6].

Based on this assumption and following up on the static fusion work of Guerriero et al [7], we recently examined predetection fusion [5], a preprocessing step to the application of a tracker. Predetection fusion blends multisensor measurements into a much smaller set of measurements to serve as input to a tracker, considerably reducing the number of false alarms while preserving good target probability of detection. The technique is thus beneficial to algorithms such as the Cardinalized Probability Hypothesis Density (CPHD) tracker [14] which is of \(O(nm^3)\) complexity, where \(n\) is the number of targets and \(m\) is the number of measurements\(^3\).

In this paper, we incorporate Doppler measurements (resulting in what we call the 4D version of our algorithm) and amplitude information (5D version) into predetection fusion. As we will show, the 4D and 5D versions of our technique significantly improve on the 2D version; that is, Doppler and aspect-dependent SNR ought to be used and fused if available.

In what follows, we discuss each of the three versions of our predetection fusion

\[^3\text{Our technique is not tracker dependent but can be combined with a variety of trackers to help reduce computation time and obtain improved tracking performance on datasets with a large number of low quality sensors.}\]
algorithm (Section II) and compare all versions against each other in a performance study (Section III) on a dataset of increasing difficulty. We close with some concluding remarks (Section IV).

II. ALGORITHM

A. Predetection Fusion with Position Measurements (2D)

In sonar surveillance systems, measurements consist of range, bearing, possibly Doppler and maybe feature data such as amplitude. Here, range and bearing measurements are first converted into Cartesian coordinates. In this version of the algorithm, we consider networks in which Doppler information is not available. As a result, the final fused measurements output by predetection fusion are in the $xy$-plane.

1) Collection: All measurements (from all receivers) that arrived at the same time scan are gathered together in one measurement set, on which the following algorithm is run. This is the input to the predetection fusion technique.

2) Sampling: The purpose of this step is to re-create the possible locus of a target, based on the detections hypothesized to have arisen from that target, and use it as motivation for the quantization decisions to be made in the next step of the algorithm.

In large networks of low quality sensors, one expects to encounter considerable measurement errors, as a large bearing error translates into a large and elongated resolution cell at long ranges. For example, in the realistic multistatic sonar Metron dataset [12], the measurements’ Cartesian covariance ellipses are very eccentric (Figure 1), with some uncertainties as much as 10-20km (major axis of ellipse)$^4$.

In order to overcome such large measurement errors, we generate $N_{mc} = 100$ samples via Monte Carlo for each contact, according to the contact’s measurement error covariance matrix. Without this step, a large covariance measurement would still only be “seen” in the grid cell containing the measurement’s nominal value.

$^4$In this work, the measurement covariances are approximated as elliptical. Given the large bearing error, they are actually banana-shaped. For a better fit, the measurement errors (which are Gaussian in range and in bearing) could be approximated as sums of Gaussians.
Fig. 1. Sampling step motivating example (Monte Carlo sampling according to each contact’s error covariance allows all these target (black star) originated measurements to contribute to a final fused measurement estimate in the highlighted cell).

Figure 1 provides the motivation for this step. All displayed measurements are target originated and ideally, all should contribute to the final fused measurement obtained by predetection fusion. Without the sampling step, measurements such as the ones at (24760, 36670) and (18790, 32150) would be quantized to cells far from the true target location, and no cell would pass the detection test described in the thresholding step below. Generation of Monte Carlo samples for these measurements allows all the displayed measurements to be ingested into the Expectation Maximization (EM) algorithm (to be described shortly) and thus, to contribute to the final fused measurement.

A similar implementation of this step would be to calculate which cells have edges that intersect the error covariance matrix and count the corresponding contact in those cells. However, an analytic and efficient way to find all rectanguloid cells that intersect with a given covariance ellipsoid does not seem available; and Monte Carlo sampling is easy.
3) **Sifting:** We then “sift” these measurement samples according to a grid in the $xy$-plane. When a contact yields at least one sample that is quantized to a grid cell, then that contact is added to the cell’s list. Additional Monte Carlo samples from the same measurement in a given grid cell have no effect.

4) **Thresholding:** A detection is declared in a cell if and only if there are more than $\tau$ contacts added to that cell’s list. We first calculate the false alarm rate as the number of contacts divided by the number of cells in the grid and also by the number of sensors. Next, we use the false alarm rate to create the binomial probability mass function that exactly $k$ out of $n = 25$ receivers have detections. Finally, we set the threshold $\tau$ by enforcing an upper limit on the false alarm rate of the fused measurements output by predetection fusion. The details on the computation of the threshold are given in [5].

We test each grid cell’s number of hits against the calculated threshold.

5) **Fusion:** For each cell that passes the test, a detection is declared. The cell’s listed contacts are then used to refine the estimated measurement location $\hat{x}$ and to estimate the posterior covariance $\hat{R}$. Please see Appendix A for more detail on how this step is implemented based on the EM algorithm.

6) **Merging:** We merge detections that gate with each other, since often neighboring cells have used the same detections from the initial Monte Carlo step (details can be found in Appendix B). This merging step is advantageous as it helps decrease the number of fused measurements produced by the predetection fusion technique in the previous step. The fused measurements that survive the merging step are the result of the predetection fusion technique. They contain position information and its associated covariance and are ready to be directly fed to a tracker that accepts Cartesian measurements as input.

B. Predetection Fusion with Position and Doppler Measurements (4D)

The 2D version of predetection fusion is to be used when only position measurements are available (no Doppler information is available), e.g., when using a linear FM (frequency modulation) waveform. If Doppler information is available, as in the case of CW (continuous wave) waveforms, the 4D version of predetection fusion is a more attractive option because it is able to provide velocity estimates to a tracker. Note that Doppler
information describes an “ellipse-rate” and is not directly translatable to a Cartesian space in which we propose to fuse.

1) Collection: All measurements (from all receivers) that arrived at the same time scan are gathered together in one measurement set, on which the following algorithm is run.

2) Sampling: As in the 2D version, for each contact, we generate \( N_{mc} = 100 \) Monte Carlo samples according to the contact’s measurement error covariance matrix. The difference is that now sampling is in a 3D space, i.e. Cartesian space and Doppler.

3) Sifting: The true Doppler of a target originated measurement is given by:

\[
\hat{\nu} = \frac{x_t(x_t - x_s) + y_t(y_t - y_s)}{2} + \frac{x_t(x_t - x_r) + y_t(y_t - y_r)}{2} \]

where \((x_t, y_t)\) and \((\hat{x}_t, \hat{y}_t)\) are the target position and speed components, \((x_s, y_s)\) is the location of the source, \((x_r, y_r)\) is the location of the receiver.

As before, each sample contact is quantized to a grid cell in the \(xy\)-plane. We start with the assumption that the declared detection in this grid cell is at the center of the cell. Therefore, the \(x\)- and \(y\)-coordinates of the center of the cell the measurement falls in are substituted for \(x_t\) and \(y_t\) in Eq. 1. The fusion step will refine the location of the declared detection \(\hat{x}\) and estimate its posterior covariance \(\hat{R}\).

We discretize the \(\hat{xy}\)-plane according to a second grid (e.g. \(2 \times 2\))\(^6\). We will refer to it as the velocity grid. Coarse quantization is preferred for reasons given in Section III.B.2. For each cell in this plane, the coordinates of its center are substituted to Eq. 1 as \(\hat{x}_t\) and \(\hat{y}_t\). We obtain \(2 \times 2 = 4\) values for the predicted Doppler \(\hat{\nu}\) and compare these values to the observed Doppler measurement. Finally, the measurement is assigned to the cell in the 4-dimensional grid whose center coordinates gave the closest predicted Doppler to the observed Doppler measurement.

4) Thresholding: Threshold computation is done as in the 2D case. It should be noted that in the calculation of the sensor probability of false alarms \(P_{FA}\), the total number of grid cells takes into consideration the number of cells in the \(\hat{xy}\)-plane.

\(^5\)We have chosen in this manuscript to refer to the system with Doppler as the “4D” case since we do indeed make this translation, albeit coarsely. Alternatively, we admit, we might call it the “3D” case.

\(^6\)A maximum speed of 6m/sec was assumed.
5) Fusion: In the fusion step, the equations used in the 2D version must be modified to incorporate the projection of the range rate (i.e. Doppler information) into velocities in the $\hat{x}\hat{y}$-plane.

$$\begin{align*}
w_i &= \frac{\pi_1 N(z_i; H_i \hat{x}_{\text{temp}}, R_i)}{\sqrt{\frac{\pi}{V}} + \pi_1 N(z_i; H_i \hat{x}_{\text{temp}}, R_i)} \quad (2)\\
\hat{x} &= \left( \sum_i w_i H_i^T R_i^{-1} H_i \right)^{-1} \left( \sum_i w_i H_i^T R_i^{-1} z_i \right) \quad (3)\\
\hat{R} &= \left( \sum_i w_i H_i^T R_i^{-1} H_i \right)^{-1} \quad (4)
\end{align*}$$

where $H$ is the measurement matrix and $V$ is now the volume of a grid cell in four dimensions. The predicted $i^{th}$ measurement, $\hat{z}_i$, can be written as:

$$\hat{z}_i = H_i \hat{x}_{\text{temp}} \quad (5)$$

as opposed to the 2D version, in which $\hat{z}_i = \hat{x}_{\text{temp}}$. Note that $\hat{x}_{\text{temp}}$ has $[x \ y \ \hat{x} \ \hat{y}]^T$ components while $\hat{z}_i$ has $[x \ y \ \hat{r}]^T$ components.

At each scan, there is usually one active source and many active receivers. Measurements that fall in the same grid cell can come from different receivers positioned at different locations, and thus the measurement matrix $H_i$ needs to be estimated for each measurement $z_i$. We can rewrite Eq. 1 as:

$$\dot{r} = \alpha \dot{x} + \beta \dot{y} \quad (6)$$

and therefore, we can estimate:

$$H_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \alpha_i & \beta_i \end{bmatrix} \quad (7)$$

where

$$\begin{align*}
\alpha_i &= \frac{\hat{x}_{\text{temp}}(x) - x_s}{2} + \frac{\hat{x}_{\text{temp}}(x) - x_r}{2} \quad (8)\\
\beta_i &= \frac{\hat{x}_{\text{temp}}(y) - y_s}{2} + \frac{\hat{x}_{\text{temp}}(y) - y_r}{2} \quad (9)
\end{align*}$$
in which $\hat{x}_{\text{temp}}(x)$ and $\hat{x}_{\text{temp}}(y)$ are the x- and y-components of the 4-dimensional current estimate of the target location $\hat{x}_{\text{temp}}$.

We start with $\hat{x}_{\text{temp}}$ at the center of a grid cell$^7$ that passes the test. In each iteration, we calculate the weights $w_i$ as in Eq. 2. Then, we can use Eq. 3 and Eq. 4 to compute $\hat{x}$ and $\hat{R}$ for the declared target. We update $\hat{x}_{\text{temp}}$ with $\hat{x}$ and iterate $N_{\text{iter}} = 10$ times.

6) Merging: We merge any two detections that gate with each other$^8$, i.e. if the distance between them in the $xy$-plane is smaller than the diagonal of a grid cell in the $xy$-plane and also the distance between them in the $\hat{x}\hat{y}$-plane is smaller than the diagonal of a grid cell in the $\hat{x}\hat{y}$-plane. The detection with the smaller fused covariance matrix is kept and the detection with the larger fused covariance matrix is discarded.

C. Predetection Fusion with Position, Doppler and SNR Measurements (5D)

The signal reflected from a sonar target can be highly aspect dependent. In many cases, only the receivers present in a so called detection zone, determined by source-target-receiver geometry and target aspect, can detect a return signal. Moreover, a small beamwidth in the detection zone translates into unlikely detection unless the geometry is favorable – it is possible for submarines to effectively hide from traditional sonar systems [16].

A multistatic sensor network, usually employing densely deployed cheap sensors, is known to improve target detection and localization accuracy and also to reduce sensitivity to scenario geometry [13]. In distributed sensor networks, the likelihood of the return signal passing through the sensor field is high, i.e. some portion of the sensor field will reliably be illuminated by the return signal (this is the case of the green receivers in Figure 2), resulting in increased target detection and better localization [16].

The 4D version of the algorithm relies on Doppler measurements to infer velocity components. The availability of velocity components makes it possible to incorporate

---

$^7$In the absence of prior information, we have not looked at other starting positions for the EM algorithm other than the center of the cell. As there is no constraint that the fused measurement must stay within the cell it started in, the choice of initial position should not significantly affect the result.

$^8$If two neighboring cells are associated with mostly the same measurements, their fused measurements would be in close proximity and likely gate with each other; then, it is desirable that one of the fused measurements should be discarded in the merging step.
aspect information into the fusion step. If SNR measurements are provided in addition to Doppler measurements, 5D predetection fusion should be selected for use.

Including SNR information requires a rederivation of the weights for the EM algorithm in the fusion step. Eq. 2 is rewritten as:

$$w_i = \frac{\pi_1\mathcal{N}(z_i; H_i\hat{x}_{temp}, R_i)p_1^\tau(a_i)}{\pi_0 P_0^\tau(a_i) + \pi_1\mathcal{N}(z_i; H_i\hat{x}_{temp}, R_i)p_1^\tau(a_i)}$$  \hspace{1cm} (10)

where \(p_0^\tau(a)\) and \(p_1^\tau(a)\) are the probability densities for the amplitude of clutter and target returns.

The synthetic dataset has a Rayleigh target amplitude model, as in Lerro and Bar-Shalom [11]:

$$p_0^\tau(a) = a \exp \left( \frac{-a^2}{2} \right), \quad a \geq \tau$$  \hspace{1cm} (11)

$$p_1^\tau(a) = \frac{a}{1 + d} \exp \left( \frac{-a^2}{2(1 + d)} \right), \quad a \geq \tau$$  \hspace{1cm} (12)

where \(d\) is the linear predicted measurement SNR\(^9\).

The predicted SNR for target originated contacts was modeled as:

$$snr_{i,pred} = SNR_{1km} - TL_{ST} - TL_{TR} + TS$$  \hspace{1cm} (13)

where \(SNR_{1km}\) is the SNR in dB that would be observed from a specular target contact 1km away from a monostatic sonar, \(TL_{ST} = 10 \log_{10} d_{ST}\) is the transmission loss (in dB) between source and target with \(d_{ST}\) the distance between source and target, \(TL_{TR}\) is the transmission loss between target and receiver, and \(TS\) is the target strength.

TS for an aspect-dependent target was modeled as a constant -10dB and a Gaussian with standard deviation \(\sigma_{TS}\) that peaks at 0dB for broadside, either orientation, as seen in Figure 3\(^{10}\):

$$TS = 10 \log_{10} \left\{ 0.1 + 0.9 \mathcal{N}(bistatic\ angle; target\ broadside\ angle, \sigma_{TS}^2) \right\}$$  \hspace{1cm} (14)

\(^9\)In our simulations, the detection threshold \(\tau\) was set to 0dB.

\(^{10}\)In our simulations, \(\sigma_{TS}\) was set to 10°.
Fig. 2. Detection zone.

Fig. 3. TS polar pattern for $\sigma_{TS} = 10^\circ$. 

10
III. RESULTS

In [5] we showed on a synthetic dataset that the 2D predetection fusion can come close to the performance of the optimal multihypothesis generalized likelihood ratio test (GLRT) approach [4], [6] to the data fusion problem. Here, we compare the performance of the newly derived 4D and 5D versions of predetection fusion against the 2D version while varying the difficulty of the synthetic dataset.

The sensor grid in Figure 4 with sensor accuracy as in Table I was used for simulations\(^{11}\). In the snapshot in Figure 4 target originated contacts are shown in magenta, clutter originated contacts in blue, 4 sources as black squares and 25 receivers as black diamonds.

A. RMSE Performance Study

In our performance study, we looked at how the algorithm reacts to a progressive increase in difficulty of the scenario. We started with the parameters in Table I and then varied in turn the number of cells in the \(xy\)- and \(\dot{xy}\)-planes, the standard deviation of the

\(^{11}\)The Metron dataset was the motivation for developing predetection fusion. Its difficulty, however, would obscure the trends Figures 5-15 provide. Predetection fusion was applied to the Metron dataset, with results given in [4].
error in Doppler and also made changes to the clutter model for Doppler measurements and TS model.

We measure the error between the predetection fusion estimate and the true target. 100 Monte Carlo simulations were averaged for each data point. In each Monte Carlo run the true position and true velocity of the target have been randomly generated.

1) Number of Cells in $xy$-Plane: Increasing the number of cells in the $xy$-plane significantly improved performance for all versions of our algorithm. Figure 5 also displays a convergence trend. Fusion of the contacts that fell into a smaller cell for which the threshold has been met brings better resolution.

The 2D version benefitted more than the 4D version from an increased number of cells in the $xy$-plane. This is reasonable since the 4D version also relies on the number of cells in $\dot{x}\dot{y}$-plane. Incorporation of SNR measurements as in the 5D version also proved beneficial, further reducing the error with respect to the 4D version. Relying on both Doppler and SNR measurements in addition to Cartesian measurements, the 5D predetection fusion is able to perform well regardless of the fineness of the discretization of the $xy$-plane.

2) Number of Cells in $\dot{x}\dot{y}$-Plane: Increasing the number of cells in the $\dot{x}\dot{y}$-plane decreased performance for the 4D and 5D versions but did not affect the 2D version (Figure 6). The 2D version of predetection fusion does not use the $\dot{x}\dot{y}$-plane for the sifting step and therefore it was expected that its performance would not change with varying number of cells in $\dot{x}\dot{y}$-plane.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Sensors</td>
<td>25</td>
</tr>
<tr>
<td>Sensor Probability of Detection</td>
<td>0.95</td>
</tr>
<tr>
<td>False Alarms per Sensor</td>
<td>2</td>
</tr>
<tr>
<td>Delay Error (sec)</td>
<td>0.01</td>
</tr>
<tr>
<td>Bearing Error (°)</td>
<td>1</td>
</tr>
<tr>
<td>Doppler Error (m/sec)</td>
<td>0.1</td>
</tr>
<tr>
<td>Clutter Doppler Std Dev (m/sec)</td>
<td>0.25</td>
</tr>
<tr>
<td>Max Doppler (m/sec)</td>
<td>6</td>
</tr>
<tr>
<td>Number of Cells in $x$</td>
<td>20</td>
</tr>
<tr>
<td>Number of Cells in $y$</td>
<td>20</td>
</tr>
<tr>
<td>Number of Cells in $\dot{x}$</td>
<td>2</td>
</tr>
<tr>
<td>Number of Cells in $\dot{y}$</td>
<td>2</td>
</tr>
<tr>
<td>Monte Carlo Samples per Contact</td>
<td>100</td>
</tr>
<tr>
<td>Designed Fused $P_{FA}$</td>
<td>0.05</td>
</tr>
<tr>
<td>Number of EM Iterations</td>
<td>10</td>
</tr>
</tbody>
</table>

TABLE I
SIMULATION PARAMETERS.
Increasing the number of cells in the $\hat{x}\hat{y}$-plane means having more but considerably smaller cells. The velocity components cannot be directly and uniquely inferred from Doppler measurements. Couple this with measurement noise and a contact which might otherwise be assigned to its true grid cell during the sifting step could be assigned to a different $(x_t, y_t, \hat{x}_t, \hat{y}_t)$ cell. The fusion step would be performed for this erroneous, possibly neighboring, cell instead of the true cell, resulting in a larger localization error. Having the grid centers of the cells in the $\hat{x}\hat{y}$-plane further apart from each other makes it more likely to assign the contact to its true 4-dimensional cell.

Figure 6 also shows that additional SNR information given to the fusion step helps direct the EM algorithm estimate from the initialization in the erroneous cell towards the true target location.

3) **Doppler Measurements Quality:** The effect of the worsening quality of the Doppler measurements, simulated as an increasing standard deviation of the zero mean Gaussian noise added to the true measurements, on the performance of the predetection fusion algorithm can be seen in Figure 7. As expected, the 2D version proved insensitive to the quality of the Doppler data. However, for the 4D and 5D versions, performance was negatively affected by less precise Doppler measurements. The reason for this trend was given in Section III.A.2 when discussing the effect of an increasing number of cells in the $\hat{x}\hat{y}$-plane. Very noisy measurements can cause a contact which might otherwise be assigned to its true Cartesian cell-velocity cell pair during the sifting step to be assigned to an erroneous cell in $(x_t, y_t, \hat{x}_t, \hat{y}_t)$. The 5D version proved more robust than the 4D version.

4) **Clutter Model for Doppler Measurements:** The effect of the clutter model for Doppler measurements has been studied in Figure 9. Target velocity was $[3.5 \ 3.5]$ m/sec and 10 sensors, randomly selected at each Monte Carlo run, with sensor probability of detection $P_D = 0.6$ were used. Figure 8 shows a snapshot of the true Doppler distribution and the clutter Doppler distribution with $\sigma_{\text{doppler}}(\text{clutter}) = 6$ m/sec. As expected, an increasing $\sigma_{\text{doppler}}(\text{clutter})$ negatively affects performance and the 4D and 5D versions of predetection fusion do better than 2D predetection fusion.
Fig. 5. Error between true target location and target location estimate (in meters) vs. Number of cells in xy-plane.

Fig. 6. Error between true target location and target location estimate (in meters) vs. Number of cells in ̄x̄ȳ-plane.

Fig. 7. Error between true target location and target location estimate (in meters) vs. Doppler measurements quality.
Fig. 8. Histogram of Doppler measurements (clutter in blue, true measurements in red).

Fig. 9. Error between true target location and target location estimate (in meters) vs. Standard deviation for clutter Doppler.

Fig. 10. Error between true target location and target location estimate (in meters) vs. $\sigma_{TS}$. 
5) **Target Strength Model:** Target strength was modeled as a constant -10dB and a Gaussian with standard deviation $\sigma_{TS}$ that peaks at 0dB for broadside, either orientation, as described by Eq. 14.

When $\sigma_{TS}$ was progressively increased, the 2D and 4D versions of predetection fusion did not respond to its variation as they do not use SNR measurements (Figure 10). On the other hand, the 5D version of our algorithm exhibited minor improvement in RMSE. A larger $\sigma_{TS}$ value penalizes less the errors in target velocity, raised by a measurement being assigned to the wrong velocity grid cell during the sifting step, errors that propagate into the calculation of the target broadside angle.

### B. Correct Cardinality Performance Study

Besides RMS error, an important issue is the ability of our algorithm to determine the correct number of targets. We did this for a relatively difficult scenario; departures from the settings in Table I will be mentioned, in the case of a single target and also for multiple targets present in the surveillance area. We looked at how many targets were declared within a detection radius of 100m around each true target while varying scenario parameters. 100 Monte Carlo runs were performed.

1) **Single Target:** Figure 11 displays the relationship between the number of successful target detections and the finesse of the Cartesian grid. For these Monte Carlo runs, sensor probability of detection was $P_D = 0.6$ and 10 randomly selected sensors were used. The three versions of predetection fusion are very close in performance, with 2D falling a bit behind the 4D and 5D versions when using a coarse grid. Performance seems to converge towards 100%, i.e. the target is always detected, with a finer and finer grid.

In Figure 12, the sensor probability of detection was $P_D = 0.6$ and a $20 \times 20$ Cartesian grid was used. There is a noticeable gap between the performance of 2D predetection fusion and the performance of the other two versions; the latter are practically indistinguishable. As expected, an increasing number of available sensors improves target detection.

In Figure 13, the sensor probability of detection was varied while 10 sensors, randomly selected at each Monte Carlo run, and a $20 \times 20$ Cartesian grid were employed. Once again, the 2D version constantly lags behind the 4D and 5D versions of our algorithm.
The Doppler enhanced versions perform well, achieving almost full detection rate with a large sensor probability of detection, with 5D having a slight advantage over 4D. This shows Doppler enhanced predetection fusion is effective at refining the location of the target estimate from the available measurements even in situations in which 2D predetection fusion runs into difficulty. Note that better sensor probability of detection translates into better performance.

2) *Multiple Targets:* Here, we analyze how an increasing true number of targets affects the ability of predetection fusion to identify the correct target cardinality. Moreover, we investigate how the existence of multiple targets influences the average RMS error per target between true target locations and target location estimates. In these runs, sensor probability of detection was $P_D = 0.6$ and 10 sensors, randomly selected at each Monte Carlo run, were used.

Figure 14 demonstrates the superiority of 4D and 5D versions over the 2D version, the former being able to consistently declare the correct target cardinality in more than 85% cases. Additionally, the average RMS error per target (Figure 15) obtained with the Doppler enhanced versions is significantly smaller than the one obtained with 2D predetection fusion. For all versions, the error seems to stabilize, the largest jump in RMS error occurring between the case of a single target and two targets. Together, these two figures show that 5D is able to refine the location of the target estimate $\hat{x}$ slightly better than 4D (Figure 15) but not as much to meaningfully affect the detection rate (Figure 14).

All methods tend to underestimate the number of targets, leading to missed detections rather than overestimate it, which would be preferable\textsuperscript{12}.

\textsuperscript{12}Generally, there is one missed detection, e.g. in Figure 14 when the true number of targets is 2, 5D predetection fusion correctly estimates two targets in 93 out of the 100 Monte Carlo runs and one target 7 times out of 100 runs.
Fig. 11. Number of successful detections of the target (out of 100 Monte Carlo runs) vs. Number of cells in xy-plane.

Fig. 12. Number of successful detections of the target (out of 100 Monte Carlo runs) vs. Number of sensors.

Fig. 13. Number of successful detections of the target (out of 100 Monte Carlo runs) vs. Sensor probability of detection.
Fig. 14. Number of times (out of 100 Monte Carlo runs) algorithm correctly estimated the number of targets vs. True number of targets.

Fig. 15. Average RMSE per target between true target location and target location estimate (in meters) vs. True number of targets.
IV. Conclusions

In [5], we introduced predetection fusion (2D version), a contact sifting procedure followed by an Expectation Maximization step refining the location of the estimated detections.

Here we introduced the 4D and 5D versions of predetection fusion, i.e. we incorporated Doppler and SNR measurements into our algorithm. It should be mentioned that 4D and 5D predetection fusion require discretization of the velocity grid. It is not possible to incorporate Doppler information directly into the algorithm as the system is underdetermined, i.e. projection of the Doppler information on \( \hat{x} \) and \( \hat{y} \) axes does not result in a unique solution. Results show that the addition of Doppler information alone (the 4D case) significantly improves both estimation of target number and accuracy. Augmentation of this by an SNR measurement (the 5D case) offers little extra in terms of target number, but accuracy of fused measurements is improved.

Appendix

A. Fusion Step for 2D Predetection Fusion

Here we discuss the means to refine \( \hat{x} \) and \( \hat{R} \), as in [5]. One might use the cell’s center for \( \hat{x} \) and compute \( \hat{R} \) via an assumption of uniformity, but that would give very poor results. Averaging all measurements in a cell’s list is a better approach [6], but it is still far from optimal and remains problematic for the incorporation of Doppler and SNR measurements. More sophisticated approaches, such as the EM algorithm that maximizes \( p(X|Z) \) over \( X \) would seem to be promising alternatives.

The measurement model of the Probabilistic Multi-Hypothesis Tracker (PMHT) is that all the measurements in a cell’s list have independent prior probabilities of association that they originated from a target located within that cell or that they are false alarms. Data association à la the PMHT algorithm is a natural choice, as it abandons the generally accepted probabilistic structure of each target having associated at most one measurement at each time. It is a perfectly feasible event that all measurements come from the same target. The PMHT measurement model is a natural fit with EM estimation. An alternative to this step would be to use the ML estimate of the likelihood.
calculated based on the measurements in the cell’s list as the final fused estimate for the cell, similar to the approach taken in [10].

For fusion, we use the following equations, obtained as per the EM algorithm [15] with a PMHT measurement model:

\[
 w_i = \frac{\pi_1 N(z_i; \hat{x}_{\text{temp}}, R_i)}{\pi_0 + \pi_1 N(z_i; \hat{x}_{\text{temp}}, R_i)} \tag{15}
\]

\[
 \hat{x} = \left( \sum_i w_i R_i^{-1} \right)^{-1} \left( \sum_i w_i R_i^{-1} z_i \right) \tag{16}
\]

\[
 \hat{R} = \left( \sum_i w_i R_i^{-1} \right)^{-1} \tag{17}
\]

where \( w_i \) is the posterior probability that the \( i^{th} \) measurement, \( z_i \), comes from the target in that cell, \( \pi_1 \) is the prior probability that the \( i^{th} \) measurement comes from the target in that cell and \( \pi_0 \) is the prior probability that the \( i^{th} \) measurement comes from clutter (we assume equal priors \( \pi_1 = \pi_0 = 0.5 \)), \( \hat{x}_{\text{temp}} \) is the predicted measurement location that is updated in each iteration of the EM algorithm and \( V \) is the volume of a grid cell (in the 2D version, \( V \) is the grid cell area in the \( xy \)-plane) [15].

We start with \( \hat{x}_{\text{temp}} \) at the center of the grid cell that passed the test. In each iteration, we calculate the weights \( w_i \) as shown in Eq. 15. Then, we can use Eq. 16 and Eq. 17 to compute \( \hat{x} \) and \( \hat{R} \) for the declared target. We update \( \hat{x}_{\text{temp}} \) with \( \hat{x} \) and repeat for a certain number of iterations\(^\text{14}\). Note that [1] and [9] discuss the convergence properties of the EM algorithm, and showed that it is guaranteed to converge to a saddle point or a local maximum, although that may not be the desired global maximum.

B. Merging Step for 2D Predetection Fusion

We test if a fused measurement created in the previous step gates with any of the other fused measurements. Two detections gate with each other if the distance between

\(^{13}\)Another related approach [8] employs Doppler measurements to create likelihood surfaces for each of the transmitter-receiver pairs; these surfaces are combined in the state space and the result is used for position inference. Position estimates benefit from the addition of multiple receivers and/or other sources of information, e.g. bearing, and are shown to improve tracking results.

\(^{14}\)There is no constraint that the fused measurement \( \hat{x} \) must stay within the grid cell in which it started.
them is smaller than the diagonal of a grid cell. The detection with the smaller fused covariance matrix is kept and the other one discarded.

Figure 16 provides an example. In a), the fused measurements resulting from the fusion step are shown, i.e. the merging step has not yet been applied. In b), the fused measurements that survived the merging step are shown. Fused measurement F1 survived because it did not gate with any other fused measurement. Fused measurement F2 gated with F3 and since F3 had a smaller covariance, F2 was discarded. F3 gated with F4 and was discarded as F4 had the smallest covariance of all fused measurements. Same for F5 and F6, they were discarded because of gating with F4. Figure 16 demonstrates how the merging step can be a beneficial addition to predetection fusion: the output fused measurement set is small, yet includes one fused measurement that is very close to the true target location (represented by the black star) and has a small covariance. Such a fused measurement set is desirable as input for any tracker.
ACKNOWLEDGMENT

This work was supported by the Office of Naval Research under contract N00014-10-10412.

REFERENCES


23