Predetection Fusion in Large Sensor Networks with Unknown Target Locations

RAMONA GEORGESCU
PETER WILLETT
STEFANO MARANO
VINCENZO MATTA

Fusion of multisensor data can improve target probability of detection but suffers from a potentially increased false alarm rate. The optimal sensor decision rule in the case of multiple sensor systems and known target location is of course a likelihood ratio test. This approach, however, is not applicable to many practical scenarios, such as sonar, in which the location of the target is not known and hence the alternative hypothesis becomes composite. Therefore, we propose predetection fusion and highlight its application to a variety of multitarget multisensor trackers. Additionally, the algorithm is motivated by the need for an efficient way to process the volume of data from large sensor networks that consist of low quality sensors. We thus propose predetection fusion as a contact sifting procedure followed by an Expectation Maximization step that refines the location of the estimated detections. Results are provided on a synthetic dataset and on a challenging realistic multistatic sonar dataset. The performance of predetection fusion is compared against the performance of the optimal multi-hypothesis GLRT approach.

Manuscript received October 20, 2010; revised November 28, 2011; released for publication December 11, 2011.

Refereeing of this contribution was handled by Pramod Varshney.

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

Authors' addresses: R. Georgescu and P. Willett, Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT 06269, E-mail: ({ramona, willett}@engr.uconn.edu); S. Marano and V. Matta, DIIIE, University of Salerno, via Ponte don Melillo I-84084, Fisciano (SA), Italy, E-mail: ({marano, vmatta}@unisa.it).

1557-6418/12/$17.00

1. MOTIVATION

Our motivation to devise this predetection fusion algorithm was twofold: one reason is the inapplicability of traditional *decentralized detection* to a realistic scenario in which target locations are unknown, while the other was the need for an efficient way to process the data generated in a large network of low quality sensors. As our ultimate goal is tracking, we will end this section by pointing out the benefits of predetection fusion to that.

1.1. Detection Fusion

In the 1980s and the 1990s, many papers appeared that dealt with optimal detection for multiple sensor systems, among them [5], [25], [27], [28], [29] and [30]. The problem under consideration was a binary detection problem with N sensors, and the target location was implicitly assumed known. The purpose of the detector was to optimally discriminate between the two simple hypotheses:

$$H_0: \quad \text{noise only},$$
$$H_1: \quad \text{target present} + \text{noise}.$$

In a practical case, such as sonar/radar detections, the location of the target is not known and the alternative hypothesis becomes composite. Hence, these approaches, generally based on likelihood ratio tests, cannot be directly applied. We here propose a practical implementation, which we will refer to as predetection fusion.

1.2. Large Networks of Low Quality Sensors

Data fusion in large sensor networks is expected to provide better target tracking capability in terms of increased area coverage, expanded geometric diversity, increased target hold, robustness to sensor loss and jamming, improved localization, and gains in probability of detection [10]. A possible drawback is an increased false alarm rate after an unwary fusion step.[1] Predetection fusion is a data fusion technique that attempts to maintain the target probability of detection while reducing the false alarm rate.

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 [10]. In recent years, interest has shifted towards deploying large sensor networks that consist of many but cheap, low quality sensors. The difficult Metron multistatic sonar dataset [23] is representative of such a setup.

In one scenario of the Metron dataset, there are 25 stationary sensors located as in Fig. 1. All the sensors are receivers with the exception of four which are colocated source/receiver units. The probability of detection

[1] What we mean is that with *direct* data fusion (i.e., pooling all available measurements), a target seen once may be repeatedly observed as each sensor offers its own perspective. While this may offer some advantage in terms of confidence and drill-down of localization, the pooled false alarm density is multiplied by the sensor count.

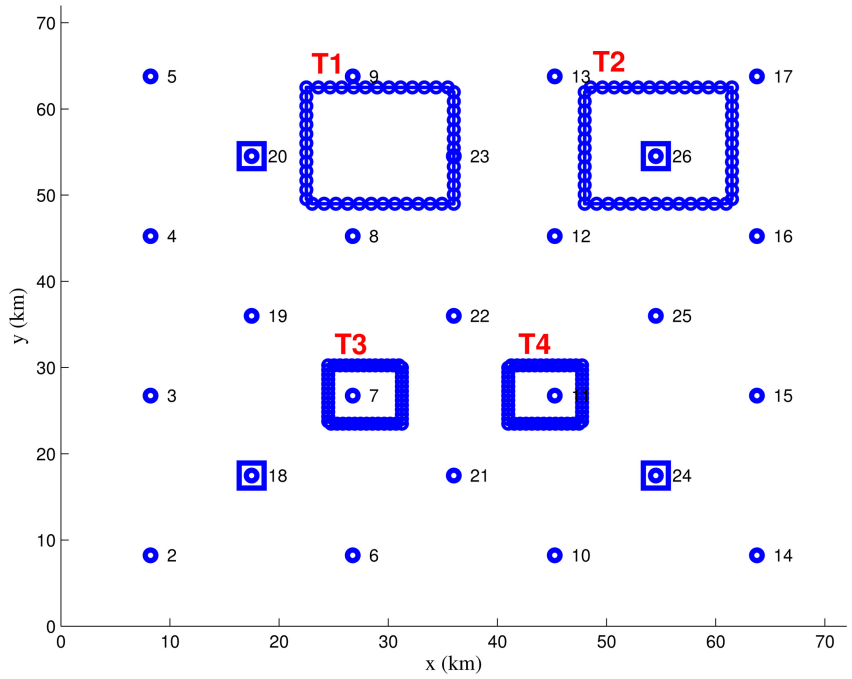

Fig. 1. Setup in the Metron dataset (4 sources S1–S4, 25 receivers RX2–RX26, 4 targets with square trajectories T1–T4).

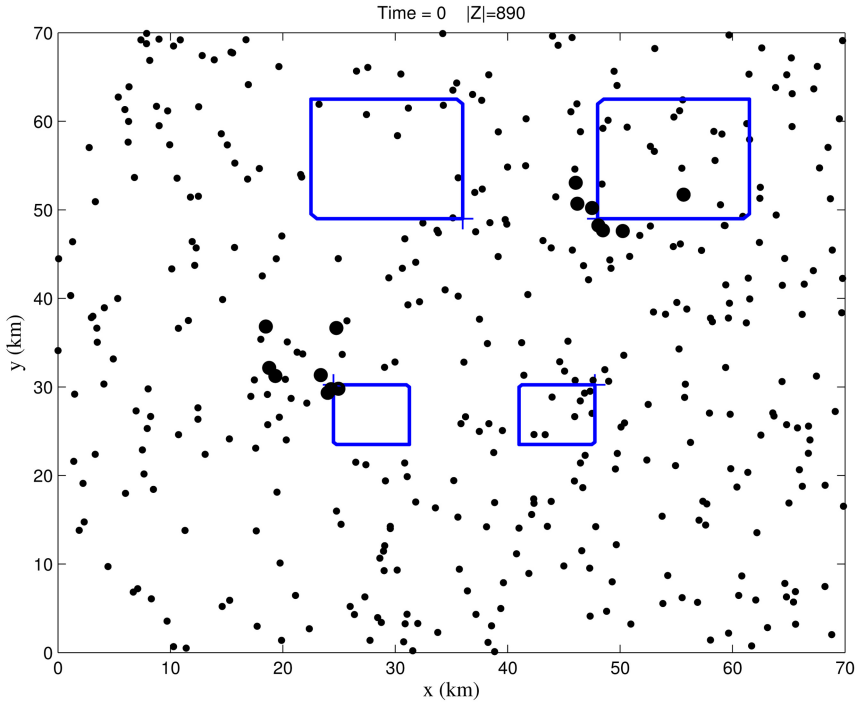

Fig. 2. Raw measurements collected from all sensors in a single scan (target originated measurements are emphasized).

is poor, on average $P_D = 0.12$ per sensor per scan. Target 1 starts in the lower right corner of its square trajectory and moves clockwise, target 2 starts in the lower left corner and moves counterclockwise, target 3 starts in the upper left corner and moves counterclockwise and target 4 starts in the upper right corner and moves clockwise. The contacts generated by targets 2 and 3 have been tagged and the ground truth is available [23]. Some further description of the Metron data is in [12], [13].

The high difficulty of the dataset is due to the extremely large number of contacts per scan and the low quality of the measurements. Fig. 2 shows the first scan of data plotted in Cartesian coordinates: there are 890 contacts out of which only 15 originate from a target, already a major challenge to any tracking paradigm. Moreover, Fig. 3 shows that the data is of very low quality: the 1-sigma covariance ellipses are very elongated (mostly due to the large uncertainty in

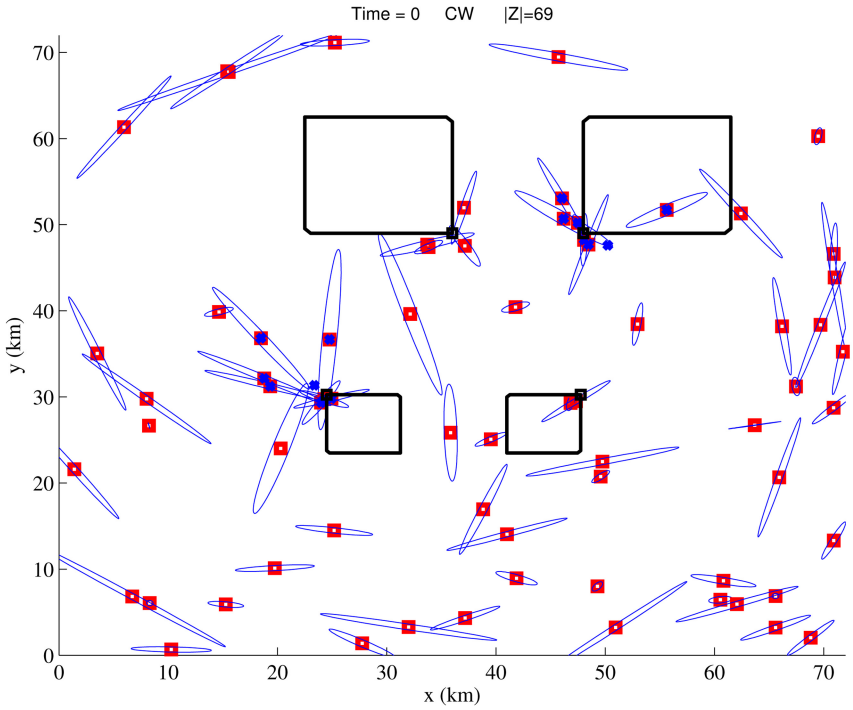

Fig. 3. Winnowed (some of the raw measurements were discarded based on their SNR and Doppler) measurements (red squares) and their covariances (blue ellipses) in the first scan (a blue dot inside a red square denotes a target originated measurement).

the bearing measurements) and can reach 20 km in their major axis. Fig. 3 displays only contacts that survived a detection test on SNR and Doppler, if available.

Predetection fusion is effective at handling both types of difficulties: on one hand, it is designed to preserve target originated measurements while at the same time removing many of the false alarms and on the other hand, its sampling step (described in Section 3) takes advantage of the large measurement uncertainties.

1.3. Improvements for Multitarget Tracking

The goal of multitarget tracking (MTT) is to estimate the states of an unknown and time-varying number of targets from a measurements series produced by a sensor network, despite data association uncertainty, sensor detection uncertainty, false alarms, and noise.

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 [16], all the contacts in a scan are fused and generate a single set of contacts to be fed to the tracker. In the latter approach (the subject of this paper), it is assumed that target originated contacts are more consistent across sensors than noise or clutter returns and thus, tracking performance would be improved.

In the case of the Metron dataset, the first approach, multi sensor scan-based tracking with the GM-CPHD, was not satisfactory [11], while predetection fusion followed by the same GM-CPHD tracker, i.e., the second approach, obtained very good results.

Predetection fusion is so flexible that it can be integrated with a variety of trackers—from well established MTT algorithms such as the Joint Probabilistic Data Association Filter (JPDAF) [1], Multiple Hypothesis Tracking (MHT) [3], and Multiple Frame Assignment (MFA) [24], to approaches recently gaining recognition such as the Maximum Likelihood-Probabilistic Data Association (ML-PDA) [4] tracker and Probability Hypothesis Density (PHD) [20] filter—to drastically reduce the number of input measurements to the tracker and therefore help reduce run time and even obtain improved multisensor tracking performance. For instance, the multisensor CPHD filter [21, 22], which is of $O(nm_1^3 \cdots m_S^3)$ complexity, where n is the number of targets and m_i is the number of measurements from sensor i out of a total of S sensors, would benefit greatly from a reduced number of input measurements.

In the following, we discuss previous attempts at data fusion in sensor networks (Section 2) and provide a detailed description of our algorithm (Section 3). We then present results on a synthetic dataset and on the realistic dataset representative of a large sensor network of low quality sensors (Section 4) and conclude in Section 5.

2. PREVIOUS APPROACHES

In the following, we discuss previous attempts at practical target location-unaware data fusion in sensor networks.

De Theije, et al. [8], [9] presented an algorithm that can be used to fuse two sets of Cartesian contacts observed by two active sonar systems, based on the calculated probability of association between nearest neighbor pairs of contacts, one from each sonar system. The performance of OR and AND fusion rules was evaluated in the presence of position errors in the observations, by means of receiver operating characteristics (ROC) curves. Simulations showed that the benefit of the fusion algorithm was not directly in terms of increased detection performance in the sense of improved ROCs, but instead in enhanced position information of the contacts. The approach is not straightforward to generalize to the case of N sensors. Moreover, it is intuitively easy to see that on datasets with many sensors with low P_{D} and high P_{FA}, such as the Metron dataset, this method would not be able to significantly help a tracker.

Krout and Hanusa [19] analyzed the Metron dataset using their PDA, PDAFAI, and PDAFAIwTS algorithms. In [18], the algorithms were extended to the JPDA and, in order to mitigate the overwhelming amount of false tracks created, a preprocessing step that utilizes a likelihood surface computed over all receivers was introduced prior to tracking. The top 30 local maxima of the final likelihood surface were extracted and sent to the JPDA tracking algorithm as measurements at a particular scan. The results for scenarios 1 and 4 of the Metron dataset showed promise but track fragmentation and track probability of detection were still in need of improvement.

The fuse-before-track architecture (FbT) constitutes an attempt to address the issue of how to best process data in large multisensor surveillance networks with a large number of cheap and limited performance sensors [6]. FbT combines measurement scans through a static fusion operation [17] that leverages more powerful batch processing techniques than can be achieved with scan-based processing. Then, scan-based processing is applied to the output of the static fusion process, enabling real-time surveillance results. Improved performance of FbT processing over centralized tracking has been demonstrated on simulated data [16].

Predetection fusion relies on the FbT approach but improves upon it in a couple of ways. First, the approach described above has been derived only for measurements in Cartesian space while predetection fusion can also easily incorporate Doppler measurements (4D version), and SNR measurements (5D version) which leads to improved accuracy in the fused measurements and helps discriminate between closely spaced targets. Second, measurement covariances are underutilized in the above approach. On the other hand, predetection fusion uses measurement covariances in its initial Monte Carlo sampling step, to alleviate the difficulty introduced by the poor quality of sensors as in the Metron dataset and in EM algorithm step, to further improve accuracy of fused measurements with respect to the true location of the target. Third, our algorithm allows for more sophisticated threshold selection methods.

The multi-hypothesis Generalized Likelihood Ratio Test (GLRT) approach developed by Guerriero, et al. [17] is the natural way to tackle the problem of data fusion in large sensor networks. For each hypothesized target, the location estimate that maximizes the likelihood function is found and the hypothesis with the largest likelihood is selected. Thus, the likelihood function is maximized with respect to both the *number* of targets and their *locations* in Cartesian coordinates.[2]

The drawback of the multi-hypothesis GLRT approach lies in the absence of a penalty mechanism for over-modeling. Therefore, we implement a modified version of this technique, in which the minimum description length (MDL) criterion is used to decide on the number of targets, and compare its performance against predetection fusion. The disadvantages of using the multi-hypothesis GLRT method with MDL penalty are twofold. First, the multi modality of the likelihood surface may induce a loss in performance through missing the global maximum during the optimization step. And second, the computational load is a serious issue.

3. PREDETECTION FUSION WITH POSITION MEASUREMENTS (2D)

In sonar surveillance systems, measurements consist of range, bearing, and possibly Doppler. Range and bearing can be converted into Cartesian measurements. In this version of the algorithm, we consider networks in which Doppler information is not available. As a result, the final fused measurements are two-dimensional (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.

2) Sampling: The purpose of this step is to recreate 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 Metron dataset, the measurements' Cartesian covariance ellipses are very eccentric (see Fig. 3), with some uncertainties as much as 10–20 km (major axis of ellipse).[3]

In order to overcome such large measurements errors, we generate $N_{mc} = 100$ samples via Monte Carlo for each contact, according to the contact's measure-

[2]Details on the multi-hypothesis GLRT can be found in Appendix B.

[3]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 by sums of Gaussians.

ment 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.

Fig. 4 illustrates the need for this step in the Metron dataset. 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 that in all likelihood would not 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 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, a graceful way to find all rectanguloid cells that intersect with a given covariance ellipsoid eludes us; 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 MC 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 τ contacts added to that cell's list. The threshold τ is a tunable parameter and can be computed as follows.

We approximate the sensor probability of false alarms as

$$P_{\mathrm{FA}} = \frac{\text{total number of contacts}}{\text{number of grid cells} \times \text{number of sensors}}. \tag{1}$$

Next, we create the binomial probability mass function that exactly k out of $n = 25$ receivers have detections

$$\Pr(K = k) = \frac{n!}{k!(n-k)!} p^k (1-p)^{n-k} \tag{2}$$

where p is the sensor P_{FA}. More generally, P_{FA} could vary from sensor to sensor. We set the threshold τ by enforcing an upper limit (at designed fused P_{FA}, e.g. 5%) on the false alarm rate of the fused measurements obtained after predetection fusion

$$\min_{\tau} \left\{ 1 - \sum_{k=0}^{\tau} \Pr(K = k) < \text{designed fused } P_{\mathrm{FA}} \right\}. \tag{3}$$

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

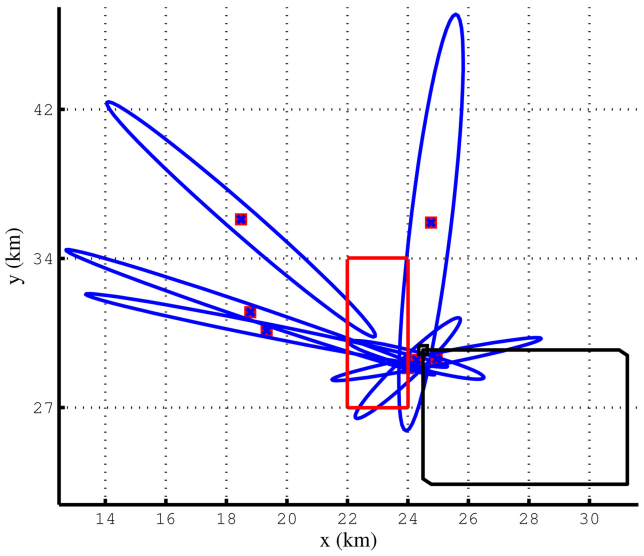

Fig. 4. Sampling step motivating example (Monte Carlo sampling according to each contact's error covariance matrix allows all of these target originated measurements to contribute to the final fused measurement estimate).

to estimate the posterior covariance $\hat{R}$. For example, 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, as demonstrated by the FbT architecture described in Section 2, 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 $p(X \mid Z)$ over X would seem to be promising alternatives.

The PMHT measurement model 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 by Krout and Hanusa [18], described in Section 2.

In this fusion step, we use the following equations, obtained as per the EM algorithm with a PMHT measurement model described in Appendix A

$$w_i = \frac{\pi_1 \mathcal{N}(z_i; \hat{x}_{\mathrm{temp}}, R_i)}{\frac{\pi_0}{V} + \pi_1 \mathcal{N}(z_i; \hat{x}_{\mathrm{temp}}, R_i)} \tag{4}$$

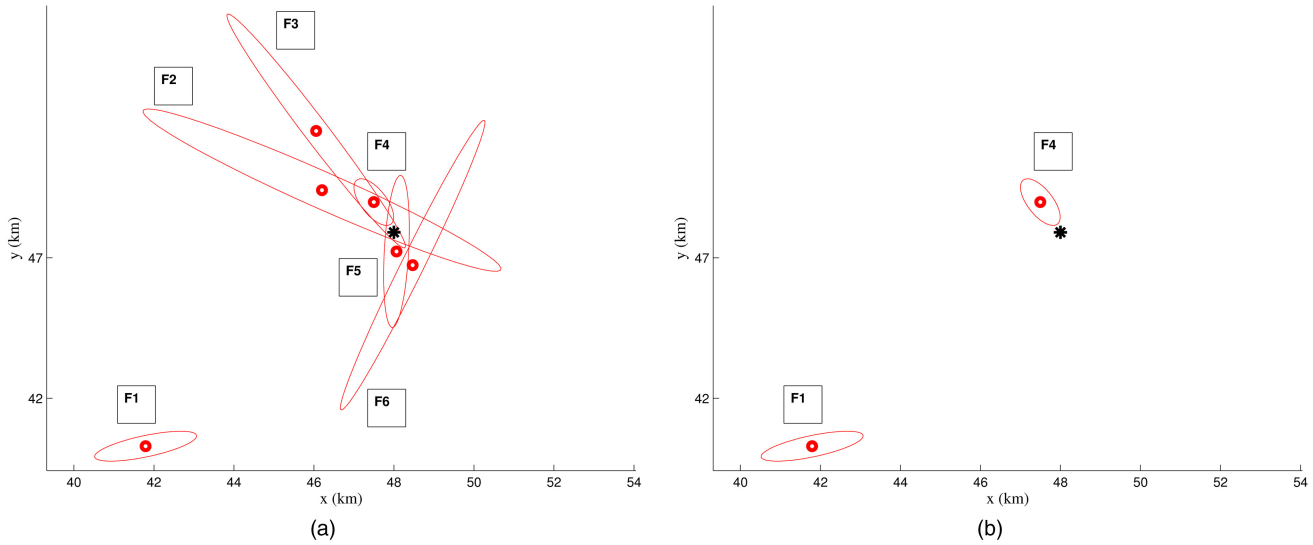

Fig. 5. Merging step notional example. Note that the fused measurement, of small covariance, close to the true location of the target survived merging while redundant fused measurements were removed. (a) Fused measurements before the merging step. (b) Fused measurements that survived the merging step.

$$\hat{x} = \left(\sum_i w_i R_i^{-1} \right)^{-1} \left(\sum_i w_i R_i^{-1} z_i \right) \tag{5}$$

$$\hat{R} = \left(\sum_i w_i R_i^{-1} \right)^{-1} \tag{6}$$

where w_i is the posterior probability that the ith measurement, z_i, comes from the target in that cell, π_1 is the prior probability that the ith measurement comes from the target in that cell, π_0 is the prior probability that the ith 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) [31].

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 (4). Then, we can use (5) and (6) to compute $\hat{x}$ and $\hat{R}$ for the declared target. We update $\hat{x}_{\text{temp}}$ with $\hat{x}$ and we repeat for a certain number of iterations.[4]

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. Merging also helps reduce the number of fused measurements that predetection fusion would feed to a tracker while preserving a good target probability of detection.

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 them (in the xy-plane) is smaller than the diagonal of a grid cell (in the xy-plane). The detection with the smaller fused covariance matrix is kept and the detection with the larger fused covariance matrix is discarded.

Fig. 5 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. Fig. 5 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.

3.1. Extensions of Predetection Fusion

Extensions to this work were discussed at length in [14]. Here, we provide an outline for them in order to emphasize that predetection fusion is a flexible, general, and powerful technique.

1) *Incorporating Doppler Measurements*: 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 FM waveform. However, multitarget tracking usually operates in a 4-dimensional state space, $[x \; y \; \dot{x} \; \dot{y}]^T$. If Doppler information is available, as in the case of CW waveforms, the 4D version of predetection fusion is a more attractive option because it is able to provide velocity estimates to a tracker, due to its use of Doppler measurements.

[4]Note that there is no constraint that the fused measurement $\hat{x}$ must stay within the grid cell in which it started.

The main differences between the 2D and 4D versions are in the sifting step and the fusion step. As described before, we quantize a sample contact to a grid cell in the xy-plane. Next, we discretize the $\dot{x}\dot{y}$-plane according to a second grid (e.g., 2×2). We calculate possible Doppler values based on the center of the grid cell in the xy-plane the contact was assigned to, all the centers of the grid cells in the $\dot{x}\dot{y}$-plane and the source and receiver location and velocity. We compare the resulting values with the observed Doppler and assign the contact to the cell in the the $\dot{x}\dot{y}$-plane that gave the closest possible Doppler to the observed Doppler.

For 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 $\dot{x}\dot{y}$-plane achieved in the modified sifting step just described. For example, (5) would become

$$\hat{x} = \left(\sum_i w_i H_i^T R_i^{-1} H_i \right)^{-1} \left(\sum_i w_i H_i^T R_i^{-1} z_i \right). \quad (7)$$

Details on the estimation of the measurement matrix, H_i, will be given in another publication [14].

2) *Improvement via SNR Information*: The 4D version of the algorithm relies on Doppler measurements to infer velocity components. The availability of velocity components makes it possible to incorporate aspect information in the fusion step. If SNR measurements are provided in addition to Doppler measurements, the 5D version of predetection fusion should be considered for use.

We quantize a contact to a grid cell the same way as in the 4D version. Then, including SNR information in the predetection fusion algorithm requires a rederivation of the weights of the EM algorithm to have them take into account the likelihood that the contact could have originated from clutter and the likelihood that the contact could have originated from a target. Calculation of these likelihoods entails the evaluation of the predicted contact SNR (based on the SNR model of the dataset) and the observed SNR of the contact.

A related data fusion technique we have subsequently developed, Random Finite Set-based Markov Chain Monte Carlo, has been analyzed and compared against 2D predetection fusion in [15].

4. RESULTS

The predetection fusion algorithm was first tested on a simple synthetic dataset and then on a realistic multistatic sonar dataset of considerably higher difficulty. Additionally, we compared the performance of predetection fusion against that of the (optimal) multihypothesis GLRT approach.

4.1. Results on Synthetic Dataset

In this dataset, we assumed 25 identical sensors, with sensor probability of detection $P_D = 50\%$ and nominal standard deviation in two coordinates $\sigma_z = 5000$. The clutter was assumed to be distributed as a homogeneous Poisson process with expected number of false alarms per sensor $\lambda V = 5$.

TABLE I
Simulation Parameters

Number of Sensors	25
Sensor Probability of Detection	0.95
False Alarms per Sensor	2
Delay Error (s)	0.01
Bearing Error (°)	1
Doppler Error (m/s)	0.1
Clutter Doppler Std Dev (m/s)	0.25
Max Doppler (m/s)	6
Number of Cells in x	20
Number of Cells in y	20
Number of Cells in $\dot{x}$	2
Number of Cells in $\dot{y}$	2
Monte Carlo Samples per Contact	100
Designed Fused P_{FA}	0.05
Number of PMHT Iterations	10

Using the same source/receiver grid as in the Metron dataset, we simulated the snapshot in Fig. 6.[5] There are four targets present in the surveillance area, represented in all the following figures by magenta stars. The ellipses represent covariance matrices in Cartesian coordinates; please note their eccentricity and density.

2D predetection fusion was applied. The state space was discretized into 20×20 grid in the xy-plane and the required fused probability of false alarm was 5%.

The results of 2D predetection fusion on the synthetic dataset of Fig. 6 can be seen in Fig. 7, while detailed views of the results can be found in Figs. 8–11. The diameter of the blue dots grows with the number of detections in each cell's list. A circle within the blue dot stands for a declared detection in that particular cell. Red $\times$s and their corresponding covariance ellipses belong to the contacts generated through predetection fusion.

The 2D version of predetection fusion declared five targets, i.e., it correctly identified the four targets present and it generated one false alarm, which is a satisfactory result given the difficulty of the data. The algorithm estimated target locations accurately, in the vicinity of the true location of the targets and with consistent covariance.

4.2. Performance Comparison

We compared the performance of 2D predetection fusion and the multi-hypothesis GLRT approach using snapshots generated with the parameters in Table I and one target present. The error (i.e., norm) between the predetection fusion estimate and the true target was used as metric of performance.

The static fusion approach [17] that motivated predetection fusion was shown to be dependent on cell size.

[5] In Figs. 6–11, the units of both axes are meters.

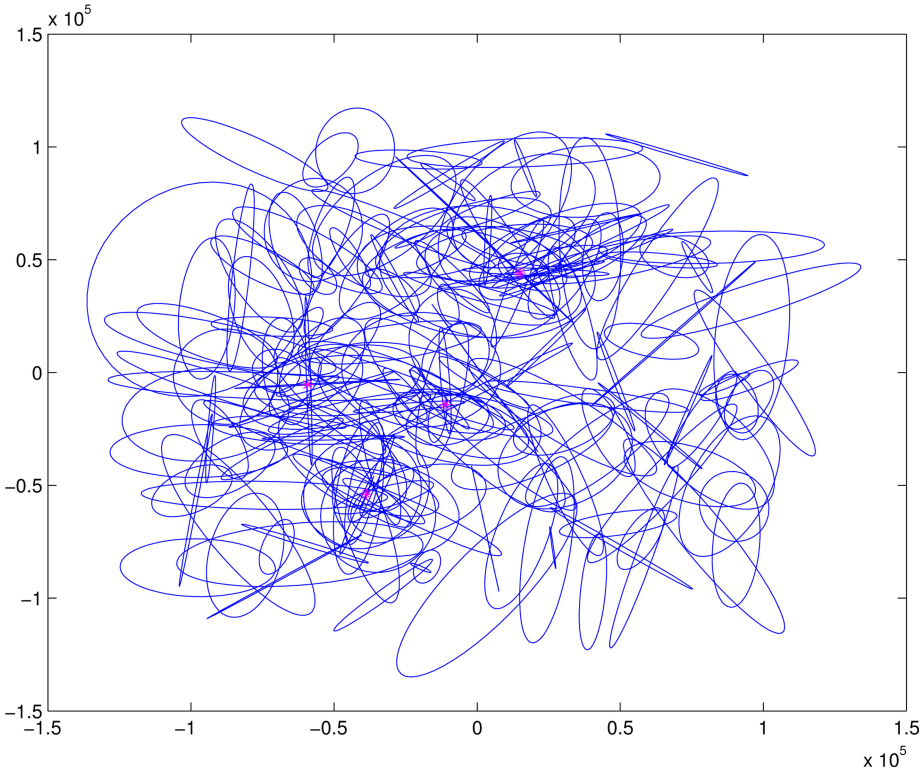

Fig. 6. Setup for synthetic dataset (true target locations in magenta, contacts from all sensors in blue). Note density of contacts and eccentricity of measurement covariances.

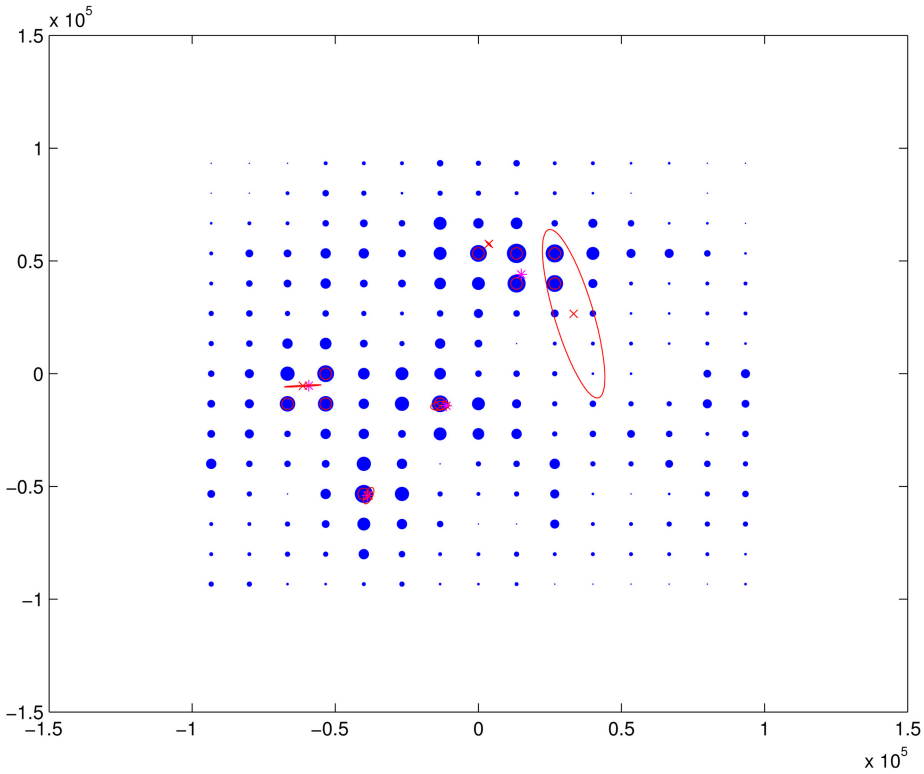

Fig. 7. 2D predetection fusion results in the xy-plane for the synthetic dataset (true target locations in magenta, fused measurements in red, size of blue dots grows with number of detections in cell's list, circle in blue dot stands for declared detection in the cell).

Therefore, we decided to investigate the effect of an increasing number of grid cells in the xy-plane on predetection fusion. We averaged 100 Monte Carlo simulations for each point in Fig. 12 and in each Monte Carlo run, the true position and true velocity of the target were randomly generated. The multi-hypothesis

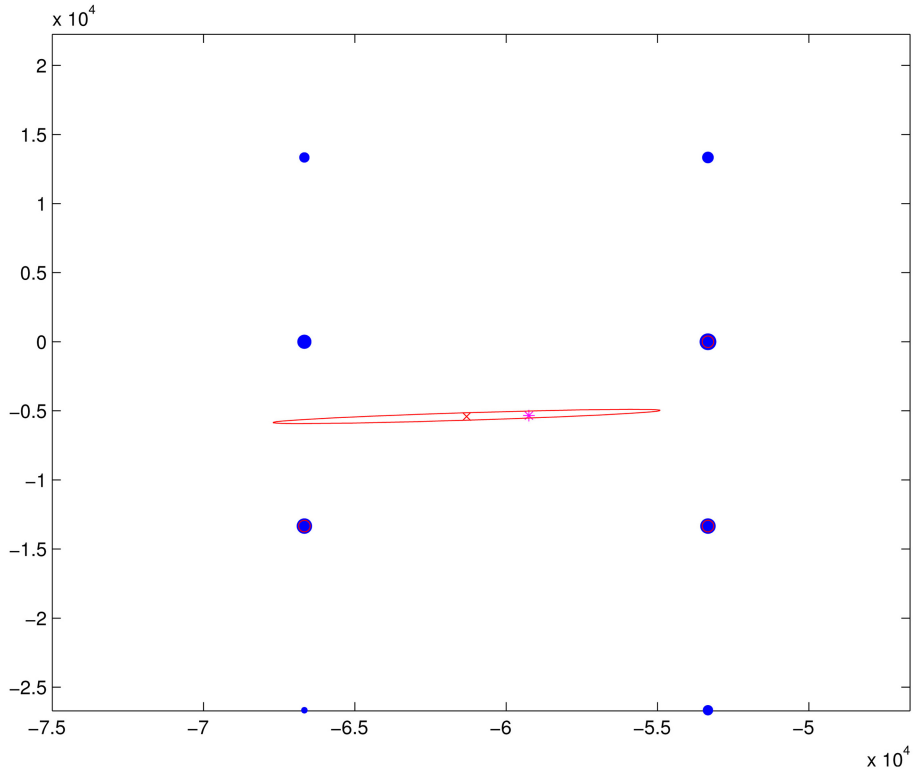

Fig. 8. 2D predetection fusion results in the xy-plane for the synthetic dataset: target 1 view.

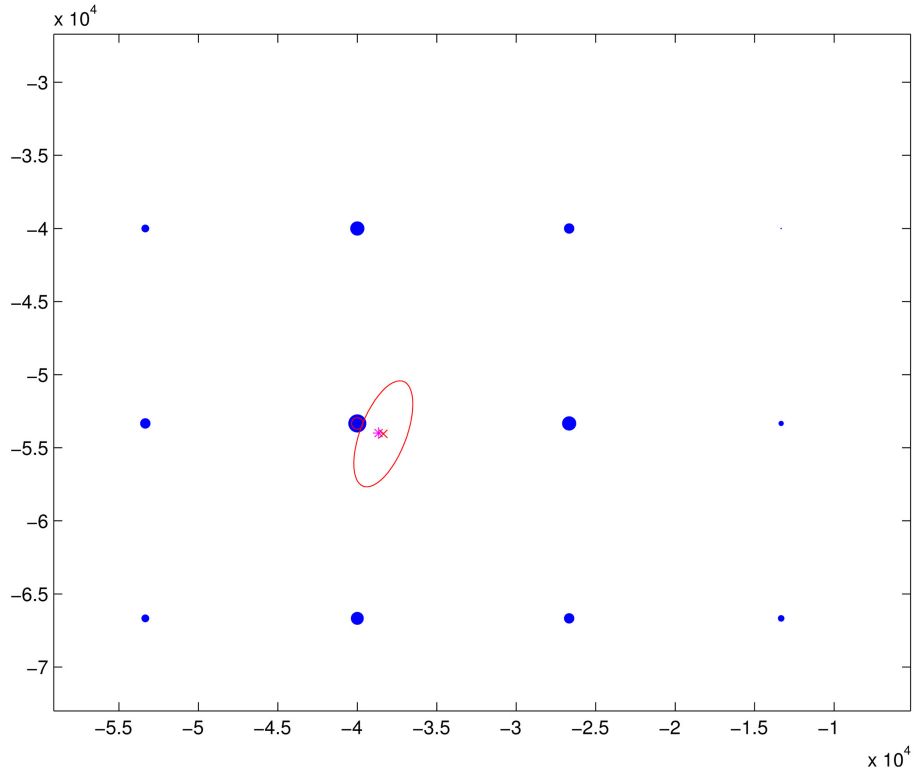

Fig. 9. Results in the xy-plane for the synthetic dataset: target 2 view.

GLRT approach is not dependent on the grid cell size.

As expected, the performance of 2D predetection fusion depended strongly on the number of grid cells in the Cartesian plane. By increasing the number of cells in the xy-plane, performance significantly improved and seemed to converge. Fusion of the contacts that fell into a smaller cell for which a detection has been declared brings better resolution.

100 Monte Carlo runs were also performed for the multi-hypothesis GLRT approach. It should be noted

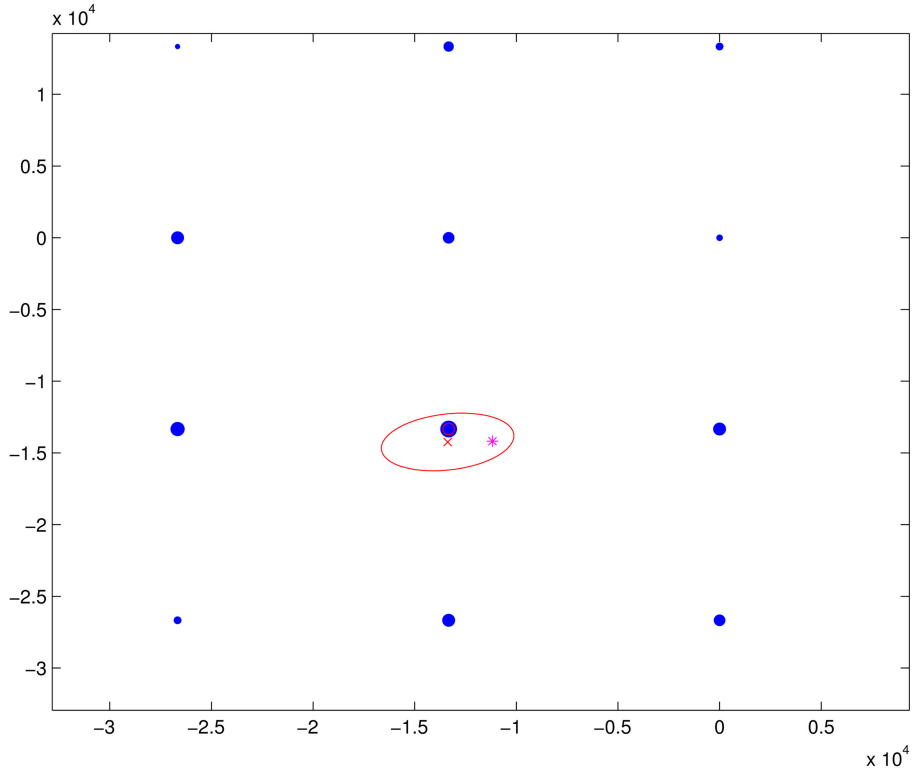

Fig. 10. Results in the xy-plane for the synthetic dataset: target 3 view.

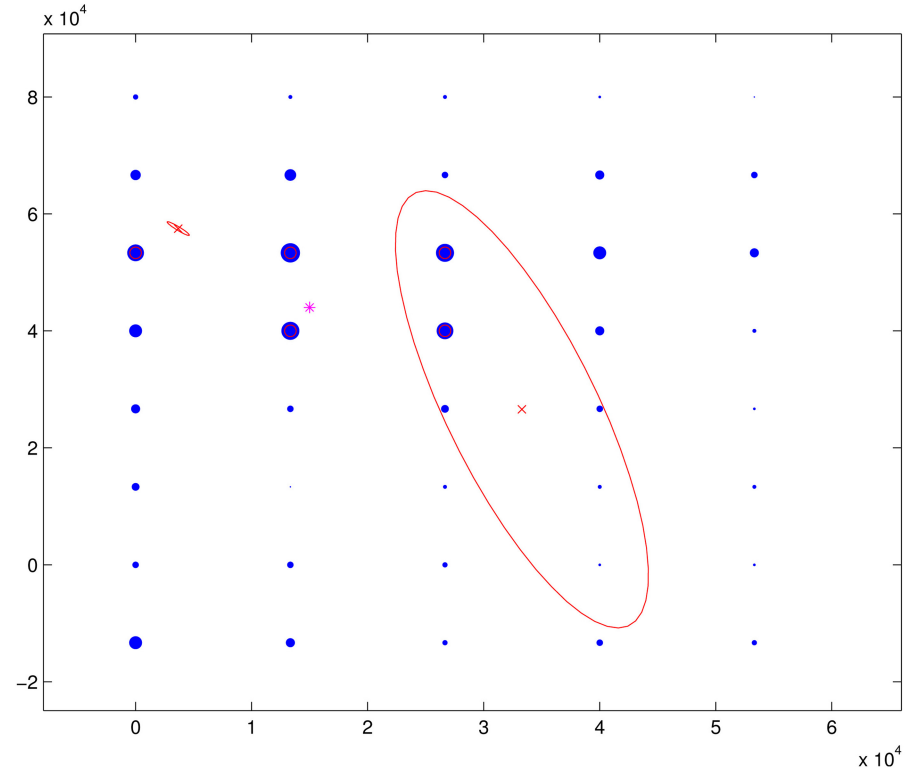

Fig. 11. Results in the xy-plane for the synthetic dataset: target 4 view.

that the multi-hypothesis GLRT approach does not depend on grid cell size and therefore its localization error appears as a flat line in Fig. 12. Predictably, the error obtained with the multi-hypothesis GLRT approach is smaller than the error achieved with predetection fusion at all grid cell sizes. The multi-hypothesis GLRT is expected to be optimal (at the expense of a sizeable run time) and it is gratifying to see that predetection fusion

TABLE II
Run Time Versus Clutter Intensity

	$\lambda V = 1$	$\lambda V = 2$	$\lambda V = 3$	$\lambda V = 4$	$\lambda V = 5$
2D	0.054	0.074	0.099	0.112	0.105 sec
GLRT	120.6	281.3	680.7	1052	2125 sec

TABLE III
Run Time Versus Number of Targets

	$N_t = 1$	$N_t = 2$	$N_t = 3$	$N_t = 4$
2D	0.065	0.091	0.128	0.155 sec
GLRT	219.1	317.8	498.8	822.7 sec

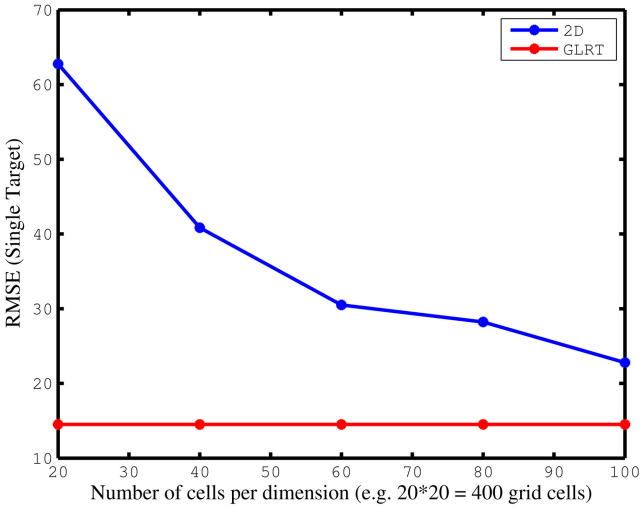

Fig. 12. Effect of number of cells on RMS error for 2D predetection fusion (Multi-hypothesis GLRT, the optimal approach, doesn't depend on cell size).

can approach its performance by using a larger number of grid cells in the Cartesian plane.

Fig. 13 shows that in the 100 Monte Carlo runs, GLRT was always able to detect the single target present at the scene while 2D predetection fusion only missed the target twice when a relatively small number of cells was used to discretize the Cartesian plane.

We also looked at how the run time is affected by increasing clutter density. Table II shows that predetection fusion executes many orders of magnitude faster than the GLRT and the gap grows wider with increasing number of false alarms per sensor per scan, λV. Moreover, Table III shows that predetection fusion executes orders of magnitude faster than the GLRT also for the case of increasing number of targets N_t present in the surveillance area.

On the other hand, the RMS error computed for the target location estimate obtained by predetection fusion that is closest to the true target location is larger than the corresponding RMS error of the GLRT and increases when more targets are added to the scene. A moderately fine grid of 60×60 cells, was used. The GLRT error also becomes larger with an increasing number of targets as seen in Fig. 14.

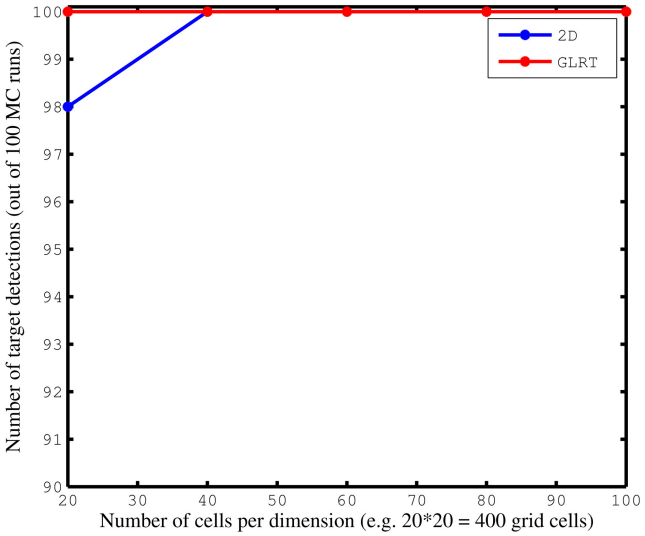

Fig. 13. Number of successful detections of the target (out of 100 MC runs) versus Number of cells in the Cartesian plane (Multi-hypothesis GLRT, the optimal approach, doesn't depend on cell size).

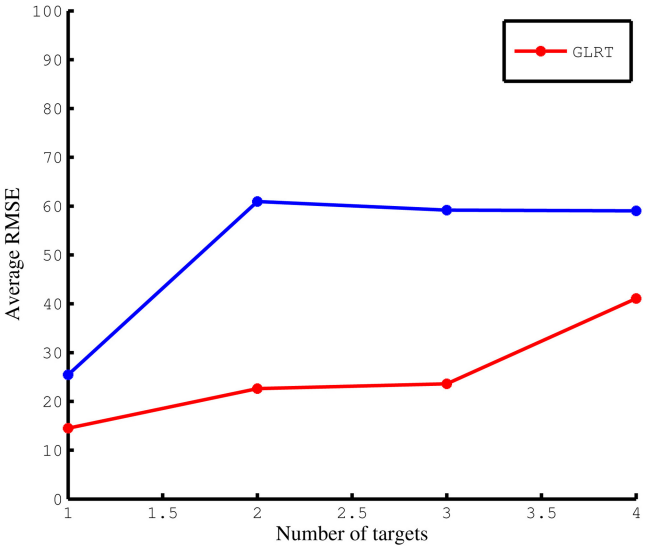

Fig. 14. RMS error for closest estimate to the true location of the target versus Number of targets.

Fig. 15 displays the behavior of the number of times (out of 100 MC runs) predetection fusion correctly estimated the number of targets present with respect to an increasing number of targets. The GLRT, the optimal method, is always able to find the correct number of targets while predetection fusion is close in performance. It should be mentioned that in the few instances in which predetection fusion does not successfully estimate the number of targets, the algorithm understimates it by one target.

4.3. Results on the Metron Dataset

Fig. 3 showed in red the location in Cartesian space of the measurements in the first scan of data of scenario 1 in the Metron dataset. This set contains measurements

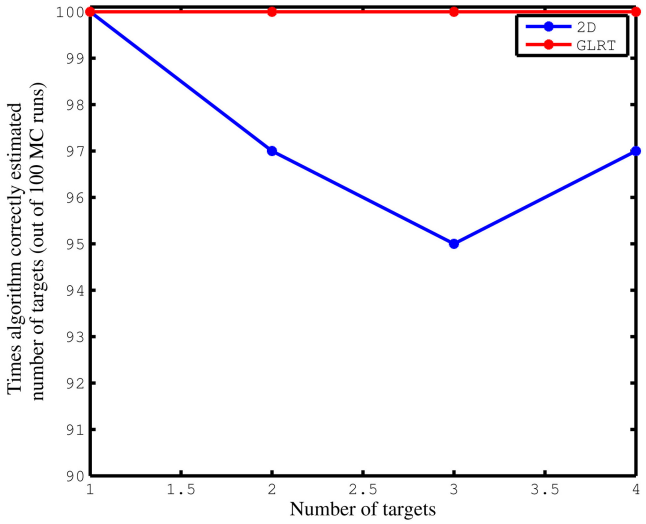

Fig. 15. Number of times (out of 100 MC runs) algorithm correctly estimated the number of targets versus Number of targets.

generated by all the 25 receivers and would serve as input to the predetection fusion algorithm. The corresponding covariances are shown in blue. Note that without the Monte Carlo step inside the predetection fusion algorithm in which 100 samples are generated for each measurement in accordance with its covariance matrix, the tagged measurements (the measurements with a blue dot inside the red square) for target 3 (lower left) would not be associated together.

Fig. 16 shows the fused measurements created through predetection fusion. The 2D version of predetection fusion was applied to the Metron dataset. The number of input contacts to be ingested by the tracker was reduced from 69 (in Fig. 3) to 53 (in Fig. 16). Although there are still many contacts left, note the single low-covariance ones at the southwest corner of target 1, southeast corner of target 2, northeast corner of target 3 and northwest corner of target 4 (the starting positions of the four targets).

In [11], it has been demonstrated that insertion of a predetection fusion step prior to tracking can considerably improve performance. Also in [11], a full set of tracking results obtained with 2D predetection fusion and the GM-CPHD tracker for all five scenarios of the Metron dataset can be found.

5. CONCLUSIONS

Optimal decentralized detection schemes have been researched since the seminal paper of Tenney and Sandell [27]. However, they seem not ready to address many real world problems for radar/sonar applications, in which the location of the target is unknown and does not precisely coincide with a resolution cell grid. Here, we proposed predetection fusion, i.e., a fusion scheme that does not need to know target location a priori.

Predetection fusion was motivated by the need for an efficient way of processing the large amount of data available from sensor networks consisting of many but low performance sensors. We evaluated our algorithm on the challenging Metron multistatic sonar dataset, which is representative of such a configuration. The large number of measurements collected at each scan

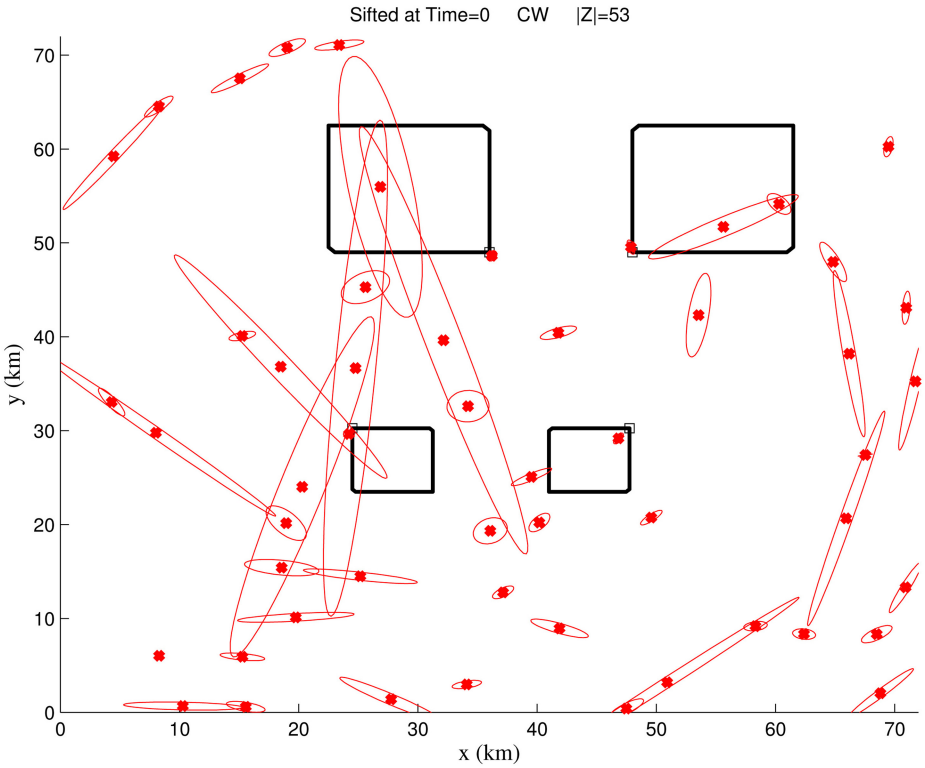

Fig. 16. Fused measurements and their covariances obtained with 2D predetection fusion for the first scan of data.

from the 25 source-receiver pairs (Fig. 2) combined with extremely low sensor quality (P_D is about 0.12 per sensor per scan) proved problematic for many well established trackers, such as MHT [7], JPDA [18].

Unsurprisingly, running the GM-CPHD tracker on the Metron data in a centralized approach, i.e., the contacts that arrived at a particular scan from all the 25 sensors were fused into one set of measurements that was sent to the tracker, resulted in considerably long run times and unacceptable performance.

Another way to go about analyzing the Metron dataset would be to apply a multisensor version of the PHD filter. However, the multisensor PHD filter has been derived in detail only for the case of two sensors. Generalization to a large number of sensors is theoretically straightforward but in terms of implementation it would seem that enumerating the partitions of the measurements is intractable when faced with a large sensor network. Running the iterated-corrector approximation, in which the updated multisensor PHD surface is the product of the individual sensor updates, would also be inefficient due to the extremely low sensor probability of detection.

In this paper, we offer an alternative approach, predetection fusion, that while mathematically not as elegant, results suggest that it can circumvent the need for a multisensor PHD. Additionally, predetection fusion is not a tracker dependent technique and 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.

Predetection fusion has been developed as a successor to the static fusion approach of Guerriero, et al. [17]. When tested against the optimal multi-hypothesis GLRT approach with MDL penalty, predetection fusion, using a very fine grid for the discretization of the Cartesian plane, performed competitively (Figs. 12 and 14), requiring many orders of magnitude lower run time.

When applied to a difficult, realistic sonar dataset containing a large sensor network composed of sensors of low quality, 2D predetection fusion enabled the GM-CPHD tracker to obtain superior tracking results [11, 18]. In such a setup, the optimal multi-hypothesis GLRT approach with MDL penalty would be impractical.

APPENDIX A. EXPECTATION MAXIMIZATION ALGORITHM

In the E-step of the EM algorithm, the expected value of the complete-data log-likelihood $\log p(X, Y \mid \Theta)$ with respect to unknown data Y given the observed data X and the current parameter estimates Θ^{n-1} is evaluated as

$$Q(\Theta, \Theta^{n-1}) = E[\log p(X, Y \mid \Theta) \mid X, \Theta^{n-1}]. \tag{8}$$

The superscript n indicates the nth EM iteration.

In the M-step, the expectation evaluated in the E-step is maximized

$$\Theta^n = \arg\max_{\Theta} Q(\Theta, \Theta^{n-1}). \tag{9}$$

The two steps are repeated as necessary. More detail on the above equations can be found in [2], which provides a good tutorial on the EM algorithm and its application to parameter estimation for Gaussian Mixture and Hidden Markov Models.

The following description borrows heavily from [31].

Let $X = \{x_s(t)\}$, where $x_s(t)$ is the state of target s at time t, $Z = \{z_r(t)\}$, where $z_r(t)$ is the rth measurement vector at time t, $K = \{k_r(t)\}$, where $k_r(t)$ is the target from which the rth measurement at time t arises. A probabilistic structure for K is needed and it can be assumed that $\Pr(k_r(t) = s) = \pi_s$ and that all are independent random variables. It should be noted that the event in which all measurements come from the same target is perfectly acceptable. Then, (8) can be written as [26]

$$Q(X^{n+1}; X^n) = \sum_K \log(p(X^{n+1}, K \mid Z)) \Pi_{t=1}^{T} \Pi_{r=1}^{n_t} w^n_{k_r(t),r}(t) \tag{10}$$

$$= \log(\Pi_{s=1}^{M} p(x_s^{n+1}(1)) \Pi_{t=2}^{T} \Pi_{s=1}^{M} \times p(x_s^{n+1}(t) \mid x_s^{n+1}(t-1)) \tag{11}$$

$$+ \sum_{K,t,r} \log(\pi_{k_r(t)} \mathcal{N}\{z_r(t); \hat{y}_{k_r(t)}, R_{k_r(t)}(t)\}) w^n_{k_r(t),r}(t) \tag{12}$$

where n_t is the number of measurements at time t, T is the number of time samples in the batch, M is the number of targets, $\mathcal{N}(x; \mu, \Sigma)$ is a Gaussian density in variable x, with mean μ and covariance Σ and $\hat{y}_{k_r(t)}(t) = H_{k_r(t)}(t) x_{k_r(t)}(t)$.

It can be shown that

$$\nabla_{X^{n+1}} Q(X^{n+1}; X^n) = \nabla_{X^{n+1}} \hat{Q}(X^{n+1}; X^n) \tag{13}$$

where

$$\hat{Q}(X^{n+1}; X^n) = \log(\Pi_{s=1}^{M} p(x_s^{n+1}(1)) \times \Pi_{t=2}^{T} \Pi_{s=1}^{M} p(x_s^{n+1}(t) \mid x_s^{n+1}(t-1)) \tag{14}$$

$$- \frac{1}{2} \sum_{s=1}^{M} \sum_{t=1}^{T} [\tilde{z}_s(t) - H_s(t) x_s^{n+1}(t)]^T \times \tilde{R}_s(t)^{-1} [\tilde{z}_s(t) - H_s(t) x_s^{n+1}(t)]. \tag{15}$$

Equation (15) is the logarithm of the joint likelihood function of M target models for which there is no data association uncertainty and for which measurements and corresponding measurement covariances are given

by their synthetic values $\tilde{z}$ and $\tilde{R}$. In predetection fusion applied to the Metron dataset, we assume one target per grid cell ($s = 1$) and in general a homothetic (or spirograph) PMHT [31], i.e., measurements at scan t can come from any Gaussian density having mean $x(t)$ and variance among $\{R_p\}_{p-1}^{P}$

$$\tilde{z}(t) = \left(\sum_{r=1}^{n_t} \sum_{p=1}^{P} w_{p,r}^{n}(t) R_p^{-1} \right)^{-1} \left(\sum_{r=1}^{n_t} \sum_{p=1}^{P} w_{p,r}^{n}(t) R_p^{-1} z_r(t) \right) \tag{16}$$

$$\tilde{R}(t) = \left(\sum_{r=1}^{n_t} \sum_{p=1}^{P} w_{p,r}^{n}(t) R_p^{-1} \right)^{-1} \tag{17}$$

$$w_{p,r}^{n}(t) = \frac{\pi_p \mathcal{N}\{z_r(t); \hat{y}_p, R_p(t)\}}{\frac{\pi_0}{V} + \sum_{l=1}^{P} \pi_l \mathcal{N}\{z_r(t); \hat{y}_l, R_l(t)\}} \tag{18}$$

meaning that $w_{l,r}^{n}(t)$ is the posterior probability (conditioned on Z and X) the rth measurement at time t is from target $k_r(t)$. In fact, for Metron data, each measurement z_i has its own covariance R_i and hence each measurement can only come from one Gaussian density. Then, $P = 1$ and the above equations simplify to those in Section 3.

APPENDIX B. MULTI-HYPOTHESIS GLRT

The mathematical model for the multi-hypothesis GLRT is described next. $\Lambda_0(Z)$ is the likelihood function given that all measurements are false alarms

$$\Lambda_0(Z) = \prod_{i=1}^{N} \frac{1}{u^{m_i}} \mu_F(m_i) \tag{19}$$

where $Z = \{Z_{ij}\}$ is the measurement set, $i = 1, 2, \ldots, N$ is the sensor number, $j = 1, 2, \ldots, m_i$ is the measurement number from sensor i, u is the search volume and $\mu_F(.)$ is the probability mass function of the number of false alarms (usually Poisson).

$\Lambda_1(Z \mid \theta_1)$ is the likelihood function given that there is one target

$$\Lambda_1(Z \mid \theta_1) = \prod_{i=1}^{N} \left[\frac{1 - P_D}{u^{m_i}} \mu_F(m_i) + \frac{P_D \mu_F(m_i - 1)}{u^{m_i - 1} m_i} \sum_{j=1}^{m_i} p(Z_{ij} \mid \theta_1) \right] \tag{20}$$

where $p(Z_{ij} \mid \theta_1) = \mathcal{N}(Z_{ij}; \theta_1, \Sigma)$ is the likelihood that measurement j from sensor i originated from target located at θ_1 with Σ as the sensor covariance matrix.

Similarly, $\Lambda_2(Z \mid \theta_1, \theta_2)$ is the likelihood function given that there are two targets

$$\Lambda_2(Z \mid \theta_1, \theta_2) = \prod_{i=1}^{N} p(Z_i \mid \theta_1, \theta_2) \tag{21}$$

where

$$p(Z_i \mid \theta_1, \theta_2) = (1 - P_D)^2 L_i^0 + P_D(1 - P_D) L_i^1 + (1 - P_D) P_D L_i^2 + P_D^2 L_i^{12} \tag{22}$$

$$L_i^0 = \frac{1}{u^{m_i}} \mu_F(m_i) \tag{23}$$

$$L_i^1 = \frac{\mu_F(m_i - 1)}{u^{m_i - 1} m_i} \sum_{j=1}^{m_i} p(Z_{ij} \mid \theta_1) \tag{24}$$

$$L_i^2 = \frac{\mu_F(m_i - 1)}{u^{m_i - 1} m_i} \sum_{j=1}^{m_i} p(Z_{ij} \mid \theta_2) \tag{25}$$

$$L_i^{12} = \frac{\mu_F(m_i - 2)}{u^{m_i - 2} m_i (m_i - 1)} \sum_{j=1}^{m_i} \sum_{l=1, l \neq j}^{m_i} p(Z_{ij} \mid \theta_1) \times p(Z_{il} \mid \theta_2). \tag{26}$$

Generalization to an arbitrary number of targets is straightforward but tedious to repeat. A constraint is imposed on the maximum number of targets, $T_{\max}$.

With a generalized likelihood ratio test (GLRT) approach, one can find the target location estimates which maximize the likelihood function for each hypothesized number of targets, and choose the largest. The algorithm starts at $\hat{\Lambda}_1(Z) = \max_{\theta_1} \Lambda_1(Z \mid \theta_1)$. The optimal target location estimate is $\hat{\theta}_1 = \arg\max_{\theta_1} \Lambda_1(Z \mid \theta_1)$. The target location estimate $\hat{\theta}_1$ is substituted back into $\hat{\Lambda}_1(Z)$.

If $\hat{\Lambda}_1(Z) < \hat{\Lambda}_0(Z)$, the algorithm stops and declares that no targets are present. Otherwise, the algorithm moves forward to computing $\hat{\Lambda}_2(Z)$ with $\hat{\theta}_1$ fixed as the location estimate for target 1. If $\hat{\Lambda}_2(Z) < \hat{\Lambda}_1(Z)$, the algorithm stops and declares that one target is present. Otherwise, the algorithm continues to $\hat{\Lambda}_3(Z)$ and beyond.

We implemented a modified GLRT, in which the minimum description length (MDL) criterion is used to decide on the number of targets. MDL is described by

$$\hat{t} = \arg\min_{t} \{ -\ln \hat{\Lambda}_t(Z) + \tfrac{1}{2} q \ln N \} \tag{27}$$

where $\hat{t}$ is the estimated number of targets, q is the number of independently adjusted parameters in the model (i.e., $q = 2t$ as target location is calculated in two Cartesian coordinates), N is the number of observations and $\ln \hat{\Lambda}_t(Z)$ is the log likelihood achieved at the Maximum Likelihood Estimates (MLEs) of the target locations for the hypothesis that there are t targets present.

REFERENCES

[1] Y. Bar-Shalom and X. R. Li
Multitarget-Multisensor Tracking: Principles and Techniques.
YBS Publishing, 1995.

[2] J. Bilmes
A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian Mixture and Hidden Markov Models.
Technical Report ICSI-TR-97-02, University of Berkeley, 1998.

[3] S. Blackman and R. Popoli
Design and Analysis of Modern Tracking Systems.
Artech House, 1999.

[4] W. Blanding, P. Willett, and Y. Bar-Shalom
ML-PDA: Advances and a new multitarget approach.
EURASIP Journal on Advances in Signal Processing, Article ID 260186, 2008.

[5] Z. Chair and P. K. Varshney
Optimal data fusion in multiple sensor detection systems.
IEEE Transactions on Aerospace and Electronic Systems, **22** (1986), 98–101.

[6] S. Coraluppi and C. Carthel
Multi-stage data fusion and the MSTWG TNO datasets.
In *Proceedings of 12th International Conference on Information Fusion*, Seattle, WA, 2009.

[7] S. Coraluppi and C. Carthel
An ML-MHT approach to tracking dim targets in large sensor networks.
In *Proceedings of 13th International Conference on Information Fusion*, Edinburgh, Scotland, 2010.

[8] P. de Theije and C. van Moll
An algorithm for the fusion of two sets of (sonar) data.
In *Oceans 2005—Europe*, vol. 1, Brest, France, 2005.

[9] P. de Theije, C. van Moll, and M. Ainslie
The dependence of fusion gain on signal-amplitude distributions and position errors.
IEEE Journal Ocean. Eng., **33**, 3 (2008), 266–277.

[10] O. Erdinc
Multistatic Sonar Target Tracking.
Ph.D. thesis, University of Connecticut, 2008.

[11] R. Georgescu, S. Schoenecker, and P. Willett
GM-CPHD and ML-PDA applied to the Metron multi-static sonar dataset.
In *Proceedings of 13th International Conference on Information Fusion*, Edinburgh, Scotland, 2010.

[12] R. Georgescu and P. Willett
The gm-cphd tracker applied to real and realistic multi-static sonar datasets.
IEEE Journal on Oceanic Engineering.

[13] R. Georgescu and P. Willett
The GM-CPHD applied to the corrected TNO-Blind, adjusted SEABAR07 and Metron multi-static sonar datasets.
In *Signal and Data Processing of Small Targets, Proc. SPIE*, vol. 7698, Orlando, FL, 2010.

[14] R. Georgescu and P. Willett
Predetection fusion with doppler measurements and amplitude information.
Accepted to *IEEE Journal of Oceanic Engineering*, 2011.

[15] R. Georgescu and P. Willett
Random finite set markov chain monte carlo predetection fusion applied to multistatic sonar data.
Accepted to *IEEE Transactions on Aerospace and Electronic Systems*, 2011.

[16] M. Guerriero
Statistical Signal Processing in Sensor Networks.
Ph.D. thesis, University of Connecticut, 2009.

[17] M. Guerriero, S. Coraluppi, and P. Willett
Analysis of scan and batch processing approaches to static fusion in sensor networks.
In *Signal and Data Processing of Small Targets, Proc. SPIE*, vol. 6969, Orlando, FL, 2008.

[18] D. Krout and E. Hanusa
Likelihood surface preprocessing with the JPDA algorithm: Metron data set.
In *Proceedings of 13th International Conference on Information Fusion*, Edinburgh, Scotland, 2010.

[19] D. Krout and D. Morrison
PDAFAI vs. PDAFAIwTS: TNO Blind dataset and SEABAR 07.
In *Proceedings of 12th International Conference on Information Fusion*, Seattle, WA, 2009.

[20] R. Mahler
Multitarget Bayes filtering via first-order multitarget moments.
IEEE Transactions on Aerospace and Electronic Systems, **39**, 4 (2003), 1152–1178.

[21] R. Mahler
The multisensor PHD filter: I. General solution via multitarget calculus.
In *Signal Processing, Sensor Fusion and Target Recognition XVIII, Proc. SPIE*, vol. 7336, 2009.

[22] R. Mahler
The multisensor PHD filter: II. Erroneous solution via Poisson magic.
In *Signal Processing, Sensor Fusion and Target Recognition XVIII, Proc. SPIE*, vol. 7336, 2009.

[23] K. Orlov
Description of the Metron simulation dataset for MSTWG.
2009.

[24] A. Poore
Multidimensional assignment formulation of data association problems arising from multitarget and multisensor tracking.
Computational Optimization and Applications, **3** (1994), 27–57.

[25] A. R. Reibman and L. W. Nolte
Optimal detection and performance of distributed sensor systems.
IEEE Transactions on Aerospace and Electronic Systems, **23** (1987), 24–30.

[26] R. Streit and T. Luginbuhl
Probabilistic multi-hypothesis tracking.
Technical Report 10428, Naval Undersea Warfare Center, 1995.

[27] R. Tenney and N. Sandell
Detection with distributed sensors.
IEEE Transactions on Aerospace and Electronic Systems, **17** (1981), 501–510.

[28] J. N. Tsitsiklis
Decentralized detection.
In *Advances in Statistical Signal Processing, Vol. 2—Signal Detection*, 1993, 297–344.

[29] J. N. Tsitsiklis
Extremal properties of likelihood ratio quantizers.
IEEE Trans. Commun., **41**, 4 (1993), 98–101.

[30] D. Warren and P. Willett
Optimum quantization for detector fusion: Some proofs, examples and pathology.
Journal of the Franklin Institute, **336**, 2 (1999), 323–359.

[31] P. Willett, Y. Ruan, and R. Streit
PMHT: Problems and some solutions.
IEEE Transactions on Aerospace and Electronic Systems, **38**, 4 (2002), 738–754.

Ramona Georgescu received her B.A.s in computer science and physics from Connecticut College in 2004 and her M.Sc. in electrical engineering from Boston University in 2007.

She is currently a Ph.D. candidate in the Electrical and Computer Engineering Department at the University of Connecticut, working under the direction of Dr. Peter Willett. Her area of interest is statistical signal processing, with an emphasis on estimation and multitarget tracking.

Peter Willett (F'03) received his B.A.Sc. (engineering science) from the University of Toronto in 1982, and his Ph.D. degree from Princeton University in 1986.

He has been a faculty member at the University of Connecticut since 1986, and since 1998 has been a professor. He has published 135 journal articles (13 more under review), 290 conference papers, and 9 book chapters. His primary areas of research have been statistical signal processing, detection, machine learning, data fusion and tracking. He has interests in and has published in the areas of change/abnormality detection, optical pattern recognition, communications and industrial/security condition monitoring.

He is editor-in-chief for *IEEE Transactions on Aerospace and Electronic Systems*, and until recently was associate editor for three active journals—*IEEE Transactions on Aerospace and Electronic Systems* (for Data Fusion and Target Tracking) and *IEEE Transactions on Systems, Man, and Cybernetics*, parts A and B. He is also associate editor for the IEEE AES Magazine, editor of the AES Magazine's periodic Tutorial issues, associate editor for ISIF's electronic *Journal of Advances in Information Fusion*, and is a member of the editorial board of IEEE's Signal Processing Magazine. He was a member of the IEEE AESS Board of Governors 2003–2009. He was general cochair (with Stefano Coraluppi) for the 2006 ISIF/IEEE Fusion Conference in Florence, Italy, Program Co-Chair (with Eugene Santos) for the 2003 IEEE Conference on Systems, Man & Cybernetics in Washington, D.C., and program cochair (with Pramod Varshney) for the 1999 Fusion Conference in Sunnyvale. He was coorganizer of the tracking subsession at the 1999 IEEE Aerospace Conference, and has been organizer of the Remote Sensing Track of that conference 2000–2003. Jointly with T. Kirubarajan he has coorganized the SPIE "System Diagnosis and Prognosis: Security and Condition Monitoring Issues" Conference in Orlando, 2001–2003. He has been a member of the IEEE Signal Processing Society's Sensor-Array & Multichannel (SAM) Technical Committee since 1997, and both serves on that TC's SAM Conferences' Program Committees and maintains the SAM website.

Stefano Marano received the Laurea degree in electronic engineering (cum laude) and the Ph.D. degree in electronic engineering and computer science both from the University of Naples, Italy, in 1993 and 1997, respectively.

Currently, he is a professor at the University of Salerno, Italy, where he was formerly assistant professor. His areas of interest include statistical signal processing with emphasis on inference, sensor networks, and information theory. He has coauthored more than 70 papers about these and related topics, including some invited, mainly on international journals/transactions and proceedings of international conferences. Professor Marano was corecipient of the S. A. Schelkunoff Transactions Prize Paper Award for the best paper published in the *IEEE Transactions on Antennas and Propagation* in 1999. Recently, he is/was in the Scientific Committee of the Remote Sensing Laboratory for Environmental Hazard Monitoring (ReSLEHM), University of Salerno, and in the Organizing Committee of the Ninth International Conference on Information Fusion (FUSION 2006).

Vincenzo Matta received the Laurea degree in electronic engineering and the Ph.D. degree in information engineering from University of Salerno, Fisciano, Italy, in 2001 and 2005, respectively.

He is currently an assistant professor with the University of Salerno. His main research interests include detection and estimation theory, signal processing, wireless communications, multiterminal inference and sensor networks.