Probabilistic Data Association Techniques for Target Tracking with Applications to Sonar, Radar and EO Sensors[1]

Y. Bar-Shalom[a], T. Kirubarajan[b] and X. Lin[a]

[a]Electrical and Computer Engineering Department
University of Connecticut
Storrs CT 06269-2157
email: {ybs,xdlin}@ee.uconn.edu

[b]Electrical and Computer Engineering Department
McMaster University
Hamilton, Ontario, Canada L8S 4K1
email: kiruba@mcmaster.ca

June 9, 2003[270]

Abstract

In this paper we present an overview of the PDA technique and its application for different target tracking scenarios, in particular for low observable (low SNR) targets. A summary of the PDA technique is presented. The use of the PDA technique for tracking low observable targets with passive sonar measurements is presented. This "target motion analysis" is an application of the PDA technique, in conjunction with the Maximum Likelihood approach, for target motion parameter estimation via a batch procedure. The use of the PDA technique for tracking highly maneuvering targets combined radar resource management is described. This illustrates the application of the PDA technique for recursive state estimation using the interacting multiple model estimator with probabilistic data association filter (IMMPDAF). Then we present a flexible (expanding and contracting) sliding-window parameter estimator using the PDA approach for tracking the state of a maneuvering target using measurements from an electro-optical sensor. This, while still a batch procedure, has the flexibility of varying the batches depending on the estimation results in order to make the estimation robust to target maneuvers as well as target appearance or disappearance.

1 Introduction

When tracking targets with less-than-unity probability of detection in the presence of false alarms or clutter[2], data association — deciding *which* of the received multiple measurements to use to update each track — is a crucial part. A number of algorithms have been developed to solve this problem [1, 2, 4, 6]. Two simple solutions are the Strongest Neighbor Filter (SNF) and the Nearest Neighbor Filter (NNF). In the SNF, the signal with the highest intensity among the validated measurements (in a gate — a region around the predicted measurement used to select the candidate measurements for association, to be discussed in more detail later) is used for track update and the others are discarded. In the NNF, the measurement closest to the predicted measurement is used. While these simple techniques work reasonably well with benign (high SNR and non-maneuvers) targets in sparse scenarios, they begin to fail as the false alarm rate increases or with low observable (low probability of target detection) maneuvering

---

[1]IEEE System Magazine, 2003.

[2]In some real world problem, one also has fixed clutter ("targets of no interest") which are, typically, ignored. The clutter we consider here is the "random" one, due to sensor or background noise.

targets [15, 19]. Instead of using only one measurement among the received ones and discarding the others, an alternative approach is to use all of the validated measurements with different weights (probabilities), known as Probabilistic Data Association (PDA) [4]. The standard PDA and its numerous improved versions have been shown to be very effective in tracking a single target in clutter [11, 19, 21, 13].

The data association problem becomes more difficult with multiple targets where the tracks compete for measurements. Here, in addition to a track validating multiple measurements as in the single target case, a measurement itself can be validated by multiple tracks, i.e., one faces contention among tracks for measurements. Many algorithms exist to handle this contention: the global nearest neighbor (GNN) approach makes "hard" assignments of measurements to tracks; the Joint Probabilistic Data Association (JPDA) algorithm is used to track multiple targets by evaluating the measurement-to-track association probabilities and combining them to find the state estimate [4]; the Multiple Hypothesis Tracking (MHT) or multiframe (S-dimensional) assignment [14, 25, 26] is a more powerful (but much more complex) algorithm that handles the multitarget tracking problem by evaluating the likelihood that there is a target given a sequence of measurements [6].

In the tracking benchmark problem [10] designed to compare the performance of different algorithms for tracking highly maneuvering targets in the presence of electronic countermeasures, the PDA-based estimator, in conjunction with the Interacting Multiple Model (IMM) estimator, yielded one of the best solutions with performance was comparable to that of the MHT algorithm [5, 19].

In this paper[3] we present an overview of the PDA technique and its application for different target tracking scenarios, in particular for LO (low SNR) targets. In Section 2 a summary of the PDA technique is presented. The use of the PDA technique for tracking low observable (LO) targets with passive sonar measurements is presented in Section 3. This "target motion analysis" (TMA) is an application of the PDA technique, in conjunction with the Maximum Likelihood (ML) approach, for target motion parameter estimation via a batch procedure. In Section 4 the use of the PDA technique for tracking highly maneuvering targets combined radar resource management is described. This illustrates the application of the PDA technique for recursive state estimation using the IMMPDAF. Section 5 presents a flexible (expanding and contracting) sliding-window parameter estimator using the PDA approach for tracking the state of a maneuvering target using measurements from an electro-optical (EO) sensor. This, while still a batch procedure, has the flexibility of varying the batches depending on the estimation results in order to make the estimation robust to target maneuvers as well as target appearance or disappearance.

2   Probabilistic Data Association

The Probabilistic Data Association (PDA) algorithm calculates the association probability that each validated measurement at the current time belongs to the target of interest. This probabilistic (Bayesian) information is used in a tracking filter, called PDA filter (PDAF), that accounts for the measurement origin uncertainty [4].

The following assumptions are made to obtain the recursive PDAF state estimator (tracker):

1. There is only one target of interest whose state evolves according to a dynamic equation driven by process noise.

---

[3]In deference to a certain (non-equationphile) person's desires, this paper rose to the challenge of being (almost) totally equation challenged. An equationful version of this paper can be found in [18].

2. The track has been initialized.

3. The past information about the target is summarized approximately by assuming the pdf (probability density function) of the current state conditioned on the past data to be Gaussian with mean given by the predicted state, with its associated prediction covariance matrix. The above basic assumption of the PDAF is similar to the GPB1 approach [3], where a single "lumped" state estimate is a quasi-sufficient statistic.

4. At each time, a measurement validation region is set up.

5. Among the possibly several validated measurements, at most one of them can be target-originated — if the target was detected and the corresponding measurement fell into the validation region.

6. The remaining measurements are assumed due to false alarms or clutter and are modeled as i.i.d. (independent identically distributed) with uniform spatial distribution.

7. The target detections occur independently over time with known probability P_D.

These assumptions make it possible to obtain a state estimation scheme that is almost as simple as the Kalman filter (KF), but much more effective in clutter. The PDAF uses a decomposition of the estimation with respect to (w.r.t.) the origin of each element of the latest set of validated measurements.

From the Gaussian assumption, the validation region (gate) is an elliptical region centered at the predicted measurement. The size of this region is determined by the gate threshold and the covariance of the innovation corresponding to the true measurement. It is critical to have "realistic" covariances to ensure that the correct measurement is validated. Since targets can change drastically their behavior, a (single model based) KF will have difficulty providing the required realistic covariances. This is discussed in more detail in Section 4 where the IMMPDAF is described.

In view of assumption 5 above, the association events: {the i-th validated measurement is the target originated} where the index i runs over all the validated measurements; and {none of the measurements is target originated, $i=0$} are mutually exclusive and exhaustive.

Using the total probability theorem [3] w.r.t. the above events, the conditional mean of the state at current time (overall estimate) can be written as the sum over all i of the updated state estimates conditioned on the event that the i-th validated measurement is correct, weighted by the conditional probability that the i-th validated measurement is correct — the association probability, obtained from the PDA procedure, discussed in the sequel.

The estimate conditioned on measurement i being correct is obtained with a standard KF or an extended KF. For $i = 0$, i.e., if none of the measurements is correct, or, if there is no validated measurement, the updated state is the predicted state.

The state update equation of the PDAF, after some manipulations, ends up as a standard KF/EKF update with a combined innovation, which is the weighted sum of the all the innovations corresponding to the validated measurements with the above association probability as weightings.

The covariance associated with the updated state is given by the standard KF/EKF covariance update with two modifications: (1) with some probability, none of the measurements is the correct one (in which case one keeps the predicted covariance); (2) an extra term is added — the so-called "spread of the innovations" (similar to the spread of the means term in a mixture [3]).

The latter accounts for the fact that one could not decide with certainty which is the correct measurement and this causes the state uncertainty to be larger.

The prediction of the state and measurement is done as in the standard filter.

To evaluate the association probabilities, a probabilistic inference is made from both the number of measurements in the validation region (from the clutter density, if known) as well as their location.

Using Bayes' formula, the probability that validated measurement i is the correct one is obtained as the joint density of the validated measurements conditioned on measurement i being the correct one multiplied the prior probability of measurement i being the correct one and divided by a normalization constant.

The joint density of the validated measurements conditioned on measurement i being the correct one is the product of

- the (assumed) Gaussian pdf of the correct (target originated) measurement, and

- the pdf of the false measurements, assumed uniform in the validation region volume (which follows from the validation ellipse or ellipsoid).

The pdf of the correct measurement is a Gaussian with mean the predicted measurement and the standard innovation covariance (the "innovation Gaussian"), restricted to the gate (divided by the gate probability).

The pdf of all the measurements in the gate given the measurement i is the correct one is then the product of the above Gaussian evaluated at measurement i, with the uniform density (inverse of the validation region volume) for the remaining measurements (deemed false).

The probabilities of the association events conditioned only on the number of validated measurements are evaluated using the probability mass function (pmf) of the number of false measurements (false alarms or random clutter) in the validation region.

Two models can be used for the pmf $\mu_F(m)$ in a volume of interest V:

(i) a Poisson model with a certain (known/assumed) spatial density

(ii) a diffuse prior model (discussed in [3]) which uses a constant that is irrelevant since it cancels out.

Finally, the association probability of measurement i turns out to be the ratio of the innovation Gaussian evaluated at this measurement over the sum of a constant and all the innovation Gaussians. The constant accounts for the target detection probability, the gate probability and the (expected) number of false measurements in the gate.

Each association probability can be shown to be given by the likelihood ratio for each measurement (the ratio of the pdf of the measurement if it is correct to its pdf if it is false) divided by the sum of all the likelihood ratios. This connects to the next approach where use will be made of the joint likelihood ratio for all the measurements.

3 Low Observable TMA Using the ML-PDA Approach with Target Features [17]

This section considers the problem of TMA (Target Motion Analysis) — estimation of the trajectory parameters of a constant velocity target — with a passive sensor, which does not provide full target position measurements. The methodology presented here applies equally to

any target motion characterized by a deterministic equation, where the initial conditions (a finite dimensional parameter vector) characterize in full the motion (see, e.g. [29]). In this case one can use the (batch) ML parameter estimation, which is more powerful than state estimation when the target motion is deterministic (it does not have to be linear). This allows the algorithm to operate successfully at 6–9 dB lower SNR than recursive state estimators[4]. Furthermore, the (batch) ML-PDA parameter estimator approach makes *no approximation*, unlike the (recursive) PDAF state estimator. However, the ML-PDA estimator requires a numerical search, to be discussed later.

3.1   The Amplitude Information Feature

The standard TMA consists of estimating the target's position and its constant velocity from bearings-only (wideband sonar) measurements corrupted by noise [3]. Narrowband passive sonar tracking, where frequency measurements are also available, has been studied in [16]. The advantages of narrowband sonar are that it does not require a maneuver of the platform for target observability (ability to estimate the parameters of the target motion) and it enhances the accuracy of the estimates. However, not all passive sonars have frequency information available. In both cases, the intensity of the signal at the output of the signal processor, which is referred to as measurement amplitude or amplitude information (AI), is used implicitly to determine whether there is a valid measurement. This is usually done by comparing it with the detection threshold, which is a design parameter; however, the amount of threshold exceedance, which does carry valuable information for data association, is not used.

The measurement amplitude carries valuable information and its use in the estimation process inproves the observability even though the amplitude information is not related to the target state directly. Also superior global convergence properties are obtained.

We model the probability density function (pdf) of the envelope detector output (the AI) when the signal is due to noise only as a unity power Rayleigh pdf (denoted as "Rayleigh(1)") and the signal originated from the target as another Rayleigh pdf. If the expected signal-to-noise ratio[5] (SNR) is d, the second Rayleigh density will have power $1+d$ and is denoted as "Rayleigh(1+d)".

A suitable threshold, denoted by τ, is used to declare a detection. The probability of detection and the probability of false alarm are denoted by P_D and P_{FA}, respectively. Both P_D and P_{FA} can be evaluated from the probability density functions of the measurements. Clearly, in order to increase P_D, the threshold τ must be lowered. However, this also increases P_{FA}. Therefore, depending on the SNR, τ must be selected to satisfy two conflicting requirements.[6]

The above density functions correspond to the signal at the envelope detector output before thresholding. Those corresponding to the output of the threshold detector are renormalized version of the above, divided by the threshold exceedance probability, P_{FA} and P_D, respectively.

Finally the amplitude likelihood ratio is the ratio of the last (renormalized) densities. This ratio indicates the relative likelihood for a measurement being "good" vs. "bad" based on its amplitude only.

---

[4]The improvement is necessarily limited because at very low SNR there is just not enough information in the data to extract a track.

[5]This is the SNR in a resolution cell, to be denoted later as SNR_C.

[6]For other probabilistic models of the detection process, different SNR values correspond to the same P_D, P_{FA} pair. Compared to the Rician model receiver operating characteristic (ROC) curve, the Rayleigh model ROC curve requires a higher SNR for the same pair (P_D, P_{FA}), i.e., the Rayleigh model considered here is pessimistic.

3.2 Target and Measurement Models

We assume that n sets of measurements made at times $t = t_1, t_2, \ldots, t_n$ are available.

For bearings-only estimation, the target motion is defined by a 4-dimensional parameter vector $\mathbf{x}$ consisting of its position at a reference time and its (assumed constant) velocity. That is, we assume deterministic target motion (no process noise [3]). Any other deterministic motion can be handled within the same framework.

The state of the sensor platform $\mathbf{x}_p$, defined by its position and velocity (which can be arbitrary), is assumed to be known.

The following additional assumptions about the statistical characteristics of the measurements are also made [16]:

1. The measurements at two different sampling instants are, conditional on the target parameter vector, independent.

2. A bearing (azimuth) measurement that originated from the target at a particular sampling instant is obtained by the sensor only once during the corresponding scan with known probability P_D and is corrupted by zero-mean additive Gaussian noise of known variance.

   Due to the presence of false measurements it is not known which is the true measurement.

3. The false bearing measurements are distributed uniformly in the surveillance region.

4. The number of false measurements at a sampling instant obey a Poisson law with a known expected number of false measurements in the surveillance region. This is determined by the false alarm probability in a resolution cell (i.e., by the detection threshold at the sensor and the cell volume).

For narrowband sonar (with frequency measurements) the target parameter vector $\mathbf{x}$ includes as its fifth component the frequency emitted by the target, assumed constant. Due to the relative motion between the target and platform at t_i this frequency will be Doppler shifted when received at the platform by the relative radial velocity between the target and the sensor platform.

The measurement vector consists of bearing, frequency and amplitude. The frequency measurements are given by the true received (Doppler shifted) frequency, also corrupted by an additive zero-mean Gaussian noise with known variance. It is also assumed that the bearing and frequency measurement noises are conditionally independent and white. The bearing and frequency measurements due to noise only are assumed to be uniformly distributed in the entire surveillance region.

3.3 Maximum Likelihood Estimator Combined With PDA — The ML-PDA

In this section we discuss the maximum likelihood estimator combined with the PDA technique for both bearings-only tracking and narrowband sonar tracking.

If there are m_i detections at time i we have the following mutually exclusive and exhaustive events [4]:

The joint pdf of all the measurements at time i assuming that measurement j is the correct one is the product of the pdfs of

- the kinematic components of measurement (bearing and frequency) j conditioned on the (yet to be estimated) target motion parameter vector x — Gaussian with mean dependent on x and (the known) x_p

- the feature (AI) components of measurement j ("target originated", i.e., Rayleigh($1+d$))

- the kinematic components of all the other measurements (assumed false) — uniform densities (constant) in the surveillance region

- the feature (AI) components of all the false measurements — Rayleigh(1)

However, since one does not know which is the correct (target originated) measurement, one has to average over all the possibilities — each measurement has prior probability of being the correct one.

Using the total probability theorem, the target's likelihood function at time i (i.e., based on the measurement from frame i) is written as the sum of

- the joint pdf of all the measurements at time i conditioned on measurement j being the correct one

- multiplied with the prior probability of measurement j being the correct one (same for all j)

- over all j

The above also includes the event that none of the measurements is correct, i.e., the "no target detection" event is accounted for. The result is a Gaussian-Rayleigh($1+d$)–Uniform-Rayleigh(1) mixture.

Instead of using the likelihood function it is convenient to use the likelihood ratio, obtained by dividing the above by the pdf of all the measurements at time i given that they are all false. This is a convenient normalization because it leads to many terms canceling and it is a dimensionless quantity. To further improve the numerical conditioning of the expression to be maximized later (to obtain the target parameter estimate) the logarithm of the likelihood ratio is taken.

The target's total likelihood ratio, accounting for all the measurement frames (all time i) is then obtained by simply adding up the single-frame likelihood ratios. The Maximum Likelihood Estimate (MLE) is obtained by finding the state $x = \hat{x}$ that maximizes the total log-likelihood ratio.

The same ML-PDA approach is also applicable to the estimation of the trajectory of an exoatmospheric ballistic missile [20, 29]. The modification of this fixed-batch ML-PDA estimator to a flexible (sliding/expanding/contracting) procedure is discussed in Section 5 and demonstrated with an actual EO data example.

3.4   Cramer-Rao Lower Bound for the Estimate

For an unbiased estimate, the Cramer-Rao lower bound (CRLB, see, i.e., [3]) states that its mean square error is bounded from below by the inverse of the Fisher information matrix (FIM). The FIM (in a "clean environment" — where P_D=1 and P_{FA}=0, i.e., with no extraneous measurements) is given by the expected value of the outer product of the gradient of the log-likelihood function with itself, evaluated at the true value of the state parameter.

As expounded in [17], the FIM is given in the present ML-PDA approach by the FIM from a "clean environment" described above, multiplied by a scalar *information reduction factor* (IRF) that accounts for the loss of information resulting from the presence of false measurements and less-than-unity probability of detection.

The IRF depends on the target P_D and the expected number of false alarms per unit volume in the measurement space. Since the IRF is between zero and 1, it leads to larger CRLB, i.e., larger parameter estimation variances — an unavoidable consequence of a "dirty environment".

3.5   Results

Both the bearings-only and narrowband (with additional frequency measurements) sonar problems with amplitude information were implemented to track a target moving at constant velocity have been implemented. The results for the narrowband case are given below, accompanied by a discussion of the advantages of using amplitude information by comparing the performances of the estimators with and without amplitude information.

In narrowband signal processing, different bands in the frequency domain are defined by an appropriate resolution cell and a center frequency about which these bands are located. The received signal is sampled and filtered in these bands before applying FFT and beamforming. Then the angle of arrival is estimated using a suitable algorithm [24]. As explained earlier, the received signal is declared as a valid measurement only if it exceeds the threshold τ. The threshold value, together with the SNR, determines the probability of detection and the probability of false alarm.

The signal processor was assumed to consist of the frequency band [500Hz, 1000Hz] with a 2048-point FFT. This results in a frequency cell $C_\gamma = 500/2048 \approx 0.25$ Hz. Regarding azimuth measurements, the sonar is assumed to have 60 equal beams, resulting in an azimuth cell C_θ = 180º/60 = 3.0º. Assuming a uniform distribution in a cell, the frequency and azimuth measurement standard deviations are given by[7] $\sigma_\gamma = 0.25/\sqrt{12} = 0.0722$ Hz and $\sigma_\theta = 3.0/\sqrt{12}$ = 0.866º.

The SNR_C in a cell[8] was taken as 6.1dB and $P_D = 0.5$.[9] The corresponding SNR in a 1Hz bandwidth (SNR_1) is 0.1dB. These values give the detection threshold as $\tau = 2.64$ and $P_{FA} = 0.0306$. Dividing P_{FA} by the volume of the resolution on cell $C_\theta C_\gamma$ yields the expected number of false alarms per unit volume as $\lambda = 0.0407/\text{deg} \cdot \text{Hz}$.

The surveillance regions for azimuth and frequency were taken as the intervals $[-20º, 20º]$ and [747 Hz, 753 Hz]. The expected number of false alarms in the entire surveillance region (in a frame) is 9.8. This means that for every true measurement that originated from the target there are about 10 false alarms which exceed the threshold.

The estimated tracks were validated using a hypothesis testing procedure described in [17]. The track acceptance test was carried out with a miss probability of 5%.

To check the performance of the estimator, simulations were carried out with clutter only (i.e., without a target) and also with a target present; measurements were generated accordingly. Simulations were done in batches of 100 runs. When there was no target, irrespective of the initial track parameter guess, the estimated track was always rejected. This corroborates the accuracy of the validation algorithm given in [17].

For the set of simulations with a target, the following scenario was selected: the target moves with velocity components of 10m/s west and 5m/s north starting from (5000m, 35000m).

---

[7]The "uniform" factor $\sqrt{12}$ corresponds to the worst case. In practice, σ_θ and σ_γ are functions of the 3dB-bandwidth and of the SNR.

[8]The commonly used SNR, designated here as SNR_1, is signal strength divided by the noise power in a 1Hz bandwidth. SNR_C is the signal power divided by the noise power in a resolution cell. The relationship between them, for $C_\gamma = 0.25$Hz is $SNR_C = SNR_1 - 6$dB. SNR_C is believed to be the more meaningful SNR because this determines the ROC curve and, thus, P_D and P_{FA}.

[9]The estimator is not very sensitive to an incorrect P_D. This is verified by running the estimator with an incorrect P_D on the data generated with a different P_D. Differences up to 0.15 are tolerated by the estimator.
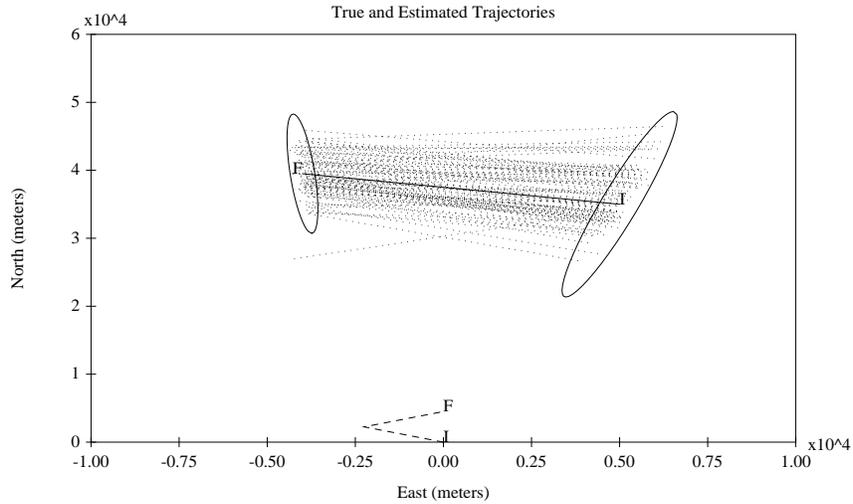
Figure 1: Scenario and estimated tracks from 100 runs for narrowband sonar with AI

The (unknown) emitted signal frequency from the target is 750Hz. The true target parameter is $\mathbf{x} = [5000\text{m}, 35000\text{m}, -10\text{m/s}, 5\text{m/s}, 750\text{Hz}]$. The motion of the platform consisted of two velocity legs in the northwest direction during the first half, and in the northeast direction during the second half of the simulation period with a constant speed of 7.1m/s. Measurements were taken at regular intervals of 30s. The observation period was 900s.

Figure 1 shows the scenario including the target true trajectory (solid line), platform trajectory (dashed line) and the 95% probability regions of the position estimates at the initial and final sampling instants based on the CRLB (with IRF). The initial and the final positions of the trajectories are marked by "I" and "F" respectively. The purpose of the probability region is to verify (visually) the validity of the CRLB as the actual parameter estimate covariance matrix from a number of Monte Carlo runs [3]. A set of bearing, frequency and amplitude measurements are shown in Figures 2, 3 and 4, respectively. Figure 1 also shows the 100 tracks formed from the estimates. Note that in all but 6 runs (i.e., 94 runs) the estimated trajectory endpoints fall in the corresponding 95% uncertainty ellipses.

Table 1 gives the numerical results from 100 runs. Here $\overline{\mathbf{x}}$ is the average of the estimates, $\hat{\sigma}$ the standard error of the estimates evaluated from 100 runs, and σ_{CRLB} the theoretical CRLB derived in Section 3.4. The range of the rough grid search used to find the initial guesses to start off the estimator are given by $\mathbf{x}_{\text{init}}$.

The efficiency of the estimator was verified using the normalized estimation error squared (NEES) [3] using the CRLB (with the IRF) as the covariance. Assuming approximately Gaussian estimation error, the NEES is chi-square distributed with n degrees of freedom where n is the number of estimated parameters. For the 94 accepted tracks the NEES was obtained as 5.46, which lies within the 95% confidence region $[4.39, 5.65]$. Also note that each component of $\overline{\mathbf{x}}$ is within $2\hat{\sigma}/\sqrt{100}$ of the corresponding component of $\mathbf{x}_{\text{true}}$. This confirms that the estimator is efficient, i.e., its covariance is indeed given by the CRLB (with the IRF). Viewing the CRLB as *the existing information* in the measurements, one can state that the ML-PDA, shown to be statistically efficient down to 6dB SNR, managed to extract all the existing information. It should be pointed out that at lower SNR the CRLB is larger (i.e., there is inherently more uncertainty) but we also lose the ability to meet this bound — the estimator is not efficient any
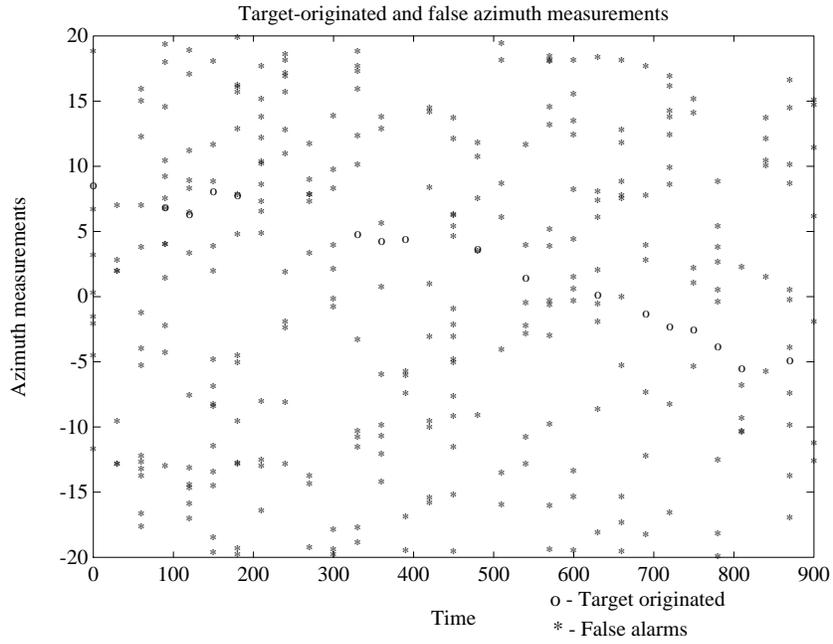
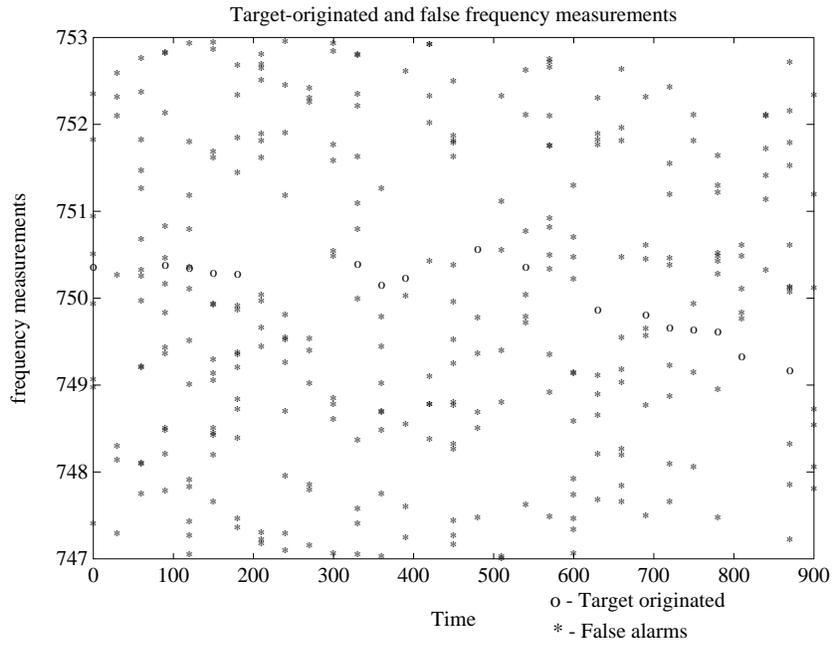Figure 2: Azimuth (bearing) measurements in a single run
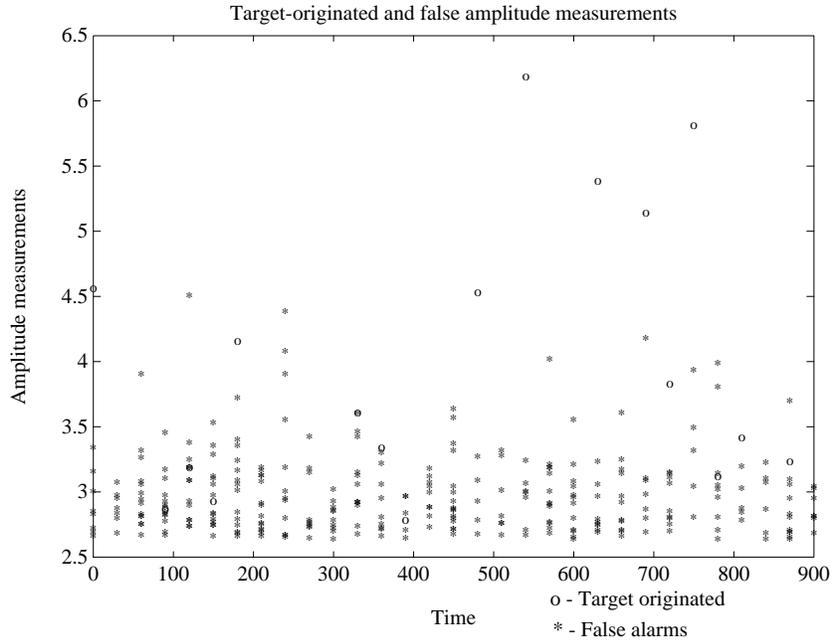

Figure 3: Frequency measurements in a single run

Figure 4: Amplitude measurements in a single run

Unit	$\mathbf{x}_{\text{true}}$	$\mathbf{x}_{\text{init}}$	$\overline{\mathbf{x}}$	σ_{CRLB}	$\hat{\sigma}$
m	5000	-12000 to 12000	4991	667	821
m	35000	49000 to 50000	35423	5576	5588
m/s	-10	-16 to 5	-9.96	0.85	0.96
m/s	5	-4 to 9	4.87	4.89	4.99
Hz	750	747 to 751	749.52	2.371	2.531

Table 1: Results of 100 Monte Carlo runs for narrowband sonar with AI ($\text{SNR}_{\text{C}} = 6.1\text{dB}$)

more. The loss of efficiency is due to the nonlinearilities of the problem.

4   The IMMPDAF for Tracking Maneuvering Targets [19]

Some of the problems of interest in single-target tracking with a single sensor are tracking maneuvering targets [3], tracking in the presence of clutter [4] and electronic countermeasures (ECM). In addition to these tracking issues, to be complete, a tracking system for a sophisticated electronically steered antenna radar has to consider radar scheduling, waveform selection and detection threshold selection. Although many researchers have worked on these issues and many algorithms are available, there had been no standard problem comparing the performances of the various algorithms. Rectifying this, the first benchmark problem focusing only on tracking a maneuvering target and pointing/scheduling a phased array radar was developed [8]. Of all the algorithms considered for this problem, the interacting multiple model (IMM) estimator yielded the best performance [7]. The second benchmark problem [9] included false alarms (FA) and ECM — specifically, a standoff jammer (SOJ) and range gate pull off (RGPO) — as well as several possible radar waveforms from which the resource allocator has to select one at every revisit time. Results for this problem showed that the IMM and multiple hypothesis tracking (MHT) algorithms were the best solutions [5, 19]. The MHT algorithm, while 1–2 orders of magnitude costlier computationally than the IMMPDAF (IMM estimator with probabilistic data association filter — PDAF — modules [4]) for the problem considered (as many as 40 hypotheses are needed[10]), yielded comparable results with the IMMPDAF. The benchmark problem of [9] was expanded in [10] to require the radar resource allocator/manager to select the operating constant false alarm rate (CFAR) and include the effects of the SOJ on the direction of arrival (DOA) measurements; also the SOJ power was increased to present a more challenging problem. While in [9] the primary performance criterion for the tracking algorithm was minimization of radar energy, the primary performance was changed in [10] to minimization of a weighted combination of radar time and energy.

This section discusses the IMMPDAF technique for automatic track formation, maintenance and termination. The coordinate selection for tracking, radar scheduling/pointing and the models used for mode-matched filtering (the modules inside the IMM estimator) are also discussed. These cover the target tracking aspects of the solution to the benchmark problem and are based on the benchmark problem tracking and sensor resource management [10, 19].

4.1   Coordinate Selection

For target tracking in track dwell mode of the radar, the number of detections at scan k (time t_k) is denoted by m_k. The m-th detection report $\overline{\zeta}_m(t_k)$ $(m = 1, 2, \ldots, m_k)$ consists of a time stamp t_k, range r_m, bearing b_m, elevation e_m, amplitude information (AI) ρ_m given by the measured SNR. In addition, one has the standard deviations of range, bearing and elevation measurements. The latter two are calculated in real time because the depend on the SNR. (For details on the different waveforms and their accuracies, see [10].)

The AI is used here only to declare detections and select the radar waveform for the next scan. Since the use of AI, for example, as in [21], can be counterproductive in discounting RGPO measurements, which generally have higher SNR than target-originated measurements, AI is not

---

[10]A more recent IMM-MHT (as opposed to Kalman filter based MHT) required 6–8 hypotheses [5].

utilized in the estimation process itself.[11]

For target tracking, the measurements and their covariance matrix are converted from spherical coordinates to Cartesian coordinates and then the IMMPDAF is used on these converted measurements. This conversion avoids the use of extended Kalman filters[12] and makes the problem linear [4]. The debiased conversion, if needed (see [3]), also guarantees the consistency of the covariances in all practical situations.

4.2   Track Formation

In the presence of false alarms, track formation is a crucial issue. Incorrect track initiation will result in target loss. In [4] an automatic track formation/deletion algorithm in the presence of clutter was presented based on the IMM algorithm. In the present benchmark problem a noisy measurement corresponding to the target of interest is given in the first scan allowing to form new tracks for each validated measurement (based on a velocity gate), as suggested in [4] and as implemented in [19], at subsequent scans. However, this is expensive in terms of both radar energy and computational load. In this implementation, track formation is simplified and handled as follows:

Scan 1 (t=0s) As defined by the benchmark problem, there is only one (target-originated, noisy) measurement. The position component of this measurement is used as the starting point for the estimated track.

Scan 2 (t=0.1s) The radar illumination beam is pointed at the location of the first measurement. This yields, possibly, more than one measurement and these measurements are gated using the maximum possible velocity of the targets to avoid the formation of impossible tracks. The maximum speed in each direction is assumed to be 500m/s.

The measurement in the first scan and the measurement with the highest SNR in the second scan are used to form a track with the two-point initialization technique [3]. The track splitting used in [4] and [19] was found unnecessary — the strongest validated measurement was adequate. This technique yields the position and velocity estimates and the associated covariance matrices in all three coordinates.

Scan 3 (t=0.2s) The pointing direction for the radar is given by the predicted position at $t = 0.2s$ using the estimates from scan 2. An IMMPDA filter with three models discussed in the sequel is initialized with the estimates and covariance matrices obtained at the second scan. The acceleration component for the third order model is assumed zero with variance $(a_{max})^2$, where $a_{max} = 70\text{m/s}^2$ is the maximum expected acceleration of the target.

From scan 3 on, the track is maintained using the IMMPDAF as described in Section 4.3. In order to maintain a high SNR for the target-originated measurement during track formation, a high energy waveform is used. Also, at the second scan, 3 dwells are used to ensure target

---

[11]Using the AI would require a separate model for the RGPO intensity, which cannot be estimated in real time due to its short duration and variability [21].

[12]This holds for linear motion models because in this case the measurements are also linear. Even range rate measurements can be treated as linear under certain conditions. However, if one uses, e.g., a coordinated turn motion model with unknown turn rate (see, e.g. [3]) then an EKF is still necessary because the turn rate enters nonlinearly.

detection. This simplified approach cannot be used if the target-originated measurement is not given at the first scan. In that case, the track formation technique in [4] can be used.

Immediate revisit (i.e., with the minimal sampling interval 0.1s) is carried out during track formation because the initial velocity of the target is not known — in the first scan only the position is measured and there is no a priori velocity. This means that in the second scan, the radar must be pointed at the first scan position, assuming zero velocity. Waiting longer to obtain the second measurement could result in the loss of the target-originated measurement due to incorrect pointing. Also, in order to make the IMM mode probabilities converge to the correct values as quickly as possible the target is revisited at a high rate.

4.3   Track Maintenance

The state vector of the target at t_k is chosen as the position, velocity and acceleration in the 3 Cartesian coordinates. The measurement vector consists of the Cartesian position components at t_k.

Assuming that the target motion is linear in the Cartesian coordinate system, the state equation for the target is a discrete white noise acceleration (WNA, 2-dimensional per coordinate) or Wiener process acceleration (WPA, 3-dimensional per coordinate), decoupled between coordinates. The predicted state and its covariance, and the predicted measurement and its associated (innovation) covariance are calculated according to the standard linear procedures [3].

4.3.1   Validation and Probabilistic Data Association

During track maintenance, each measurement at scan k is validated against the established track. This is achieved by setting up a validation region centered around the predicted measurement at t_k given by an ellipsoid determined by the innovation covariance.

Since an IMM estimator [3] is used to model the various motion modes of the target, each of its modules has a different predicted measurement and a different innovation covariance. As discussed in [4] Sec. 4.5, it is necessary to use a common validation gate so the mode likelihood functions have the same physical dimension so they can be compared.[13] This gate should be located at the weighted average of the mode-conditioned predicted measurements, with weights the predicted mode probabilities. Since, in general, the mode with the largest process noise has by far the largest innovation covariance, one can use this to determine the validation ellipsoid.

With the modules of the IMM estimator being PDAFs, the mode-conditioned estimation is done as discussed in Sec. 2. The individual mode motion models are discussed later.

4.3.2   IMM Estimator Combined with the PDA Technique

In the IMM estimator it is assumed that at any time the target trajectory evolves according to one of a finite number of models, which differ in their noise levels and/or structures [3]. By probabilistically combining the estimates of the filters, typically Kalman, matched to these modes, an overall estimate is found. In the IMMPDAF the Kalman filter is replaced with the PDA filter, which handles the data association.

---

[13]Since the likelihood functions are the joint probability density functions of the validated measurements, their physical dimension is the inverse of the physical dimension of the measurement space raised to power of the number of validated measurements. For example for m bearing frequency measurements, the physical dimension of their likelihood functions is $(\text{deg·Hz})^{-m}$. Thus, having the same measurements in each mode likelihood function makes them commensurate.

Let r be the number of mode-matched filters used, $M(t_k)$ the index of the mode in effect in the semi-open interval $(t_{k-1}, t_k]$ and $\mu_j(t_k)$ be the probability that mode j ($j = 1, 2, \ldots, r$) is in effect in the above interval given the data up to and including k. The mode transitions are modeled by a Markov chain with transition probability discussed later in more detail. The state estimates and their covariance matrix at t_k conditioned on the j-th mode are denoted by $\hat{\mathbf{x}}_j(t_k)$ and $P_j(t_k)$, respectively.

The steps of the IMMPDAF are as follows [4]:

Step 1 — Mode interaction or mixing. The mode-conditioned state estimate and the associated covariances from the previous iteration are mixed to obtain the initial condition for the mode-matched filters.

Step 2 — Mode-conditioned filtering. A PDAF is used for each mode to calculate the mode-conditioned state estimates and covariances. In addition, we evaluate the likelihood function $\Lambda_j(t_k)$ of each mode at t_k using a Gaussian-uniform mixture consisting of a product of

- the pdf of measurement j given that it is the target-originated innovation, assumed Gaussian
- the pdf of the false measurements (the remaining ones) assumed uniform in the validation region volume

summed over each measurement being the correct one, with weight given by the prior probability of this event (same for all the measurements).

This amounts to a Gaussian-uniform mixture. The above summation also accounts for the probability that the target was not detected (or its measurement was not validated). The probability of target detection P_D is given by the target's expected SNR.

Note that the likelihood function, as a pdf, has a physical dimension that depends on the number of the measurements m_k. Since ratios of these likelihood functions are to be calculated, they all must have the same dimension, i.e., the same m_k. Thus a common validation region is vital for all the models in the IMMPDAF. Typically the "largest" innovation covariance matrix corresponding to "noisiest" model covers the others and, therefore, this can be used.

Step 3 — Mode update. The mode probabilities are updated based on the likelihood of each mode using Bayes' formula.

Step 4 — State combination. The mode-conditioned estimates and covariances are combined to find the overall estimate $\hat{\mathbf{x}}(t_k)$ and its covariance matrix $P(t_k)$ are obtained according to the mixture equation [3].

- The overall estimate is the sum of the all mode-conditioned estimates weighted by the updated mode probabilities
- The overall covariance is the sum of all the mode-conditioned covariances weighted by the updated mode probabilities plus the "spread of the means terms".

Note that the above combination of the mode-conditioned estimates and covariances is similar to what the PDAF does with the measurement conditioned update. The common thread between these techniques is the fact that they account to the replacement of a Gaussian mixture pdf with a moment matched single Gaussian (see [4] Sec. 1.4.16).

4.3.3 The Models in the IMM Estimator

The selection of the model structures and their parameters is one of the critical aspects of the implementation of IMMPDAF. Designing a good set of filters requires a priori knowledge about the target motion, usually in the form of maximum accelerations and sojourn times in various motion modes [3]. The tracks considered in the benchmark problem span a wide variety of motion modes — from benign nearly constant velocity motions to maneuvers up to $7g$. To handle all possible motion modes and to handle automatic track formation and termination, the following models are used:

Benign motion model (M^1). This second order (white noise acceleration — WNA) model per coordinate with low process corresponds to the non-maneuvering intervals of the target trajectory. For this model the process noise is, typically, assumed to model air turbulence.

Maneuver model (M^2). This second order model (WNA) with high noise level corresponds to on-going maneuvers. For this model the process noise standard deviation is taken as $\sigma_{v_2} = \alpha a_{max}$, where a_{max} is the maximum acceleration in the corresponding modes and $0.5 < \alpha \leq 1$ [3].

Maneuver detection model (M^3). This is a third order (Wiener process acceleration — WPA) model with high level noise. For highly maneuvering targets, like military attack aircraft, this model is useful for detecting the onset and termination of maneuvers. For civilian air traffic surveillance [30], this model is not necessary.

For this model, the process noise standard deviation is chosen as $\sigma_{v_3} = \min\{\beta\Delta_a\delta, a_{max}\}$, where Δ_a is the maximum acceleration increment per unit time (jerk) and δ is the sampling interval and $0 < \beta \leq 1$ [3].

For the targets under consideration, $a_{\max} = 70\text{m/s}^2$ and $\Delta_a = 35\text{m/s}^3$. Using these values, the process noise standard deviations were taken as[14]

$$
\begin{aligned}
&\sigma_{v_1} = 3\text{m/s}^2 && \text{(for non-maneuvering intervals)}.\\
&\sigma_{v_2} = 35\text{m/s}^2 && \text{(for maneuvering intervals)}.\\
&\sigma_{v_3} = \min\{35\delta, 70\} && \text{(for maneuver start/termination)}.
\end{aligned}
$$

In addition to the process noise levels, the elements of the Markov chain transition matrix between the modes are also design parameters. Their selection depends on the sojourn time in each motion mode. The transition probability depends on the expected sojourn time via $\tau_i = \delta/(1 - p_{ii})$, where τ_i is the expected sojourn time of the i-th mode, p_{ii} is the probability of transition from i-th mode to the same mode and δ is the sampling interval [3]. For the above models, $p_{ii}, i = 1, 2, 3$ are calculated as $p_{ii} = \min\{u_i, \max(l_i, 1 - \delta/\tau_i)\}$, where $l_i = 0.1$ and $u_i = 0.9$ are the lower and upper limits, respectively, for the ith model transition probability. The expected sojourn times of 15, 4, and 2s, are assumed for modes M^1, M^2 and M^3, respectively.

The selection of the off-diagonal elements of the Markov transition matrix depends on the switching characteristics among the various modes and is done as follows:

$$
\begin{aligned}
&p_{12} = 0.1(1 - p_{11}) && p_{13} = 0.9(1 - p_{11})\\
&p_{21} = 0.1(1 - p_{22}) && p_{23} = 0.9(1 - p_{22})\\
&p_{31} = 0.3(1 - p_{33}) && p_{32} = 0.7(1 - p_{33})
\end{aligned}
$$

The x, y, z components of the target dynamics are (assumed) uncoupled and the same process noise is used in each coordinate.

---

[14]The benchmark requirement was to have a single estimator design for all targets.

4.4 Track Termination

According to the benchmark problem a track is declared lost if the estimation error is greater than the two-way beamwidth in angles or 1.5 radar range gates in range. In addition to this problem-specific criterion, the IMMPDAF declares (on its own) track loss if the track is not updated for 100s. Alternatively, one can include a "no target" model [4], which is useful for automatic track termination, in the IMM mode set. In a more general tracking problem, where the true target state is not known, the "no target" mode probability or the track update interval would serve as the criterion for track termination and the IMMPDAF would provide a unified framework for track formation, maintenance and termination.

4.5 Simulation Results

In this section results obtained using the algorithms discussed above are presented. The computational requirements and root-mean-square errors (RMSE) are given.

The tracking algorithm using the IMMPDAF is tested on the following six benchmark tracks[15]:

Target 1 A large military cargo aircraft with maneuvers up to $3g$.

Target 2 A Learjet or commercial aircraft which is smaller and more maneuverable than target 1 with maneuvers up to $4g$.

Target 3 A high speed medium bomber with maneuvers up to $4g$.

Target 4 Another medium bomber with good maneuverability up to $6g$.

Targets 5 & 6 Fighter or attack aircraft with very high maneuverability up to $7g$.

Figure 5 shows the trajectory of target 6: The target maintains constant speed and course for 30s before making a 7g turn at $t = 30$s. The new course is maintained for another 30s and a 6g turn is performed while the throttle is reduced and the aircraft is nosed over in order to rapidly decrease altitude. This turn-and-dive maneuver is accompanied by RGPO and the SOJ happens to be in the mainlobe of the radar at the same time to maximize the tracking difficulty. After 30s another 6g turn at full throttle is performed. Finally, constant velocity is maintained after another 7g turn at $t = 150$s [10].

In Table 2, the performance measures and their averages for the IMMPDAF (in the presence of FA, RGPO and SOJ [10, 19]) are given. During maneuvering periods the revisit intervals were shorter. The averages are obtained by adding the corresponding performance metrics of the six targets (with those of target 1 added twice due to its relative frequency in the real world) and dividing the sum by 7. In the table, the maneuver density is the percentage of the total time that the target acceleration exceeds $0.5g$. The average floating point operation (FLOP) count per second was obtained by dividing the total number of floating point operations by the target track length. This is the computational requirement for target and jammer tracking, neutralizing techniques for ECM and adaptive parameter selection for the estimator, i.e., it excludes the computational load for the radar emulation.

---

[15]The tracking algorithm does not know the type of the target under track — the parameters are selected to handle any target.
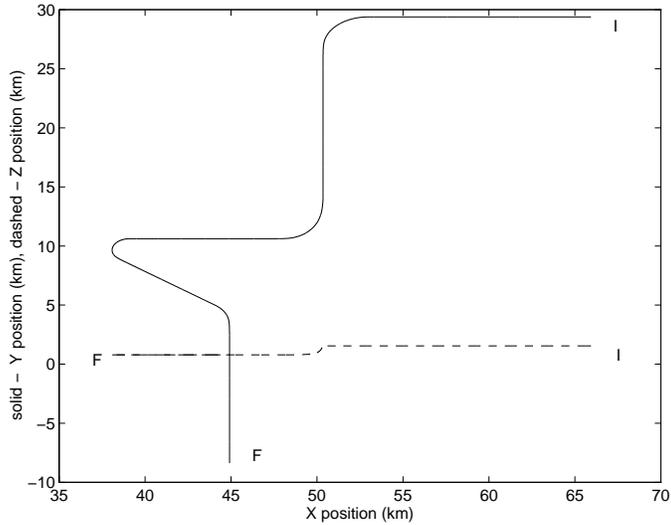
Figure 5: A Trajectory of Target 6 (I - Initial Position, F - Final Position)

Target	Time Length (s)	Max. Acc. (m/s^2)	Man. Density (%)	Sample Period (s)	Ave. Power (W)	Pos. RMSE (m)	Vel. RMSE (m/s)	Ave. Load (kFLOPS)	Lost Tracks (%)
1	165	31	25	2.65	8.9	98.1	61.3	22.2	1
2	150	39	28.5	2.39	5.0	97.2	68.5	24.3	0
3	145	42	20	2.38	10.9	142.1	101.2	24.6	1
4	184	58	20	2.34	3.0	26.5	25.9	24.3	0
5	182	68	38	2.33	18.4	148.1	110.7	27.1	2
6	188	70	35	2.52	12.4	98.6	71.4	24.6	1
Ave.	–	–	–	**2.48**	**8.3**	–	–	**24.5**	–

Table 2: Performance of IMMPDAF in the Presence of False Alarms, Range Gate Pull-Off and the Standoff Jammer

The average FLOP[16] requirement is 25 kFLOPS, which can be compared with the FLOP rate of 78 MFLOPS of a (by now obsolete) Pentium processor running at 133MHz. Thus, the real-time implementation of the complete tracking system is possible for numerous such targets simultaneously. With the average revisit interval of 2.5s, the FLOP requirement of the IMMPDAF is 62.5 kFLOP/radar cycle. With the revisit time calculations taking about the same amount of computation as a cycle of the IMMPDAF, but running at half the rate of the Kalman Filter (which runs at constant rate), the IMMPDAF with adaptive revisit time is about 10 times costlier computationally than a Kalman Filter. Due to its ability to save radar resources, which are much more expensive than computational resources, the IMMPDAF is a worthwhile and viable alternative to the Kalman filter, which is the standard "workhorse" in many current tracking systems.[17]

---

[16]The FLOP count is obtained using the built-in MATLAB function `flops`. Note that these counts, which are given in terms of thousands of floating point operations per second (kFLOPS) or millions of floating point operations per second (MFLOPS), are rather pessimistic — the actual FLOP requirement would be considerably lower. Nevertheless, this is the best estimate of the computational requirements at this point.

[17]Some systems still use the α-β filter as their "workmule". As suggested by a well-known person in tracking,

5 A Flexible-Window ML-PDA Estimator for Tracking LO Targets [12]

One difficulty with the ML-PDA approach of Section 3, which uses a fixed set of scans of measurements as a batch, is the incorporation of non-informative scans when the target is not present in the surveillance region for some consecutive scans. For example, if the target appears within the surveillance region of the sensor after the first few scans, the estimator can be misled by the pure clutter in those scans — the earlier scans contain no relevant information and the incorporation of these into the estimator not only increases the amount of processing (without adding any more information) but also results in less accurate estimates or even possible track rejection. Also, a target could disappear from the surveillance region for a while during tracking and reappear sometime later. Then, these scans contain little or no information about the target and can potentially mislead the tracker.

In addition, the standard ML-PDA estimator assumes that the target SNR, the target velocity and the density of false alarms over the entire tracking period remain constant. In practice, this may not be the case and then the standard ML-PDA estimator will not yield the desired results. For example, the average target SNR may vary significantly as the target gets closer to or moves away from the sensor. Also, the target might change its course and/or speed intermittently over time. For electro-optical (EO) sensors, depending on the time of the day and weather, the number of false alarms may vary as well.

Because of these concerns, an estimator capable of handling time-varying SNR (with online adaptation), false alarm density and slowly evolving course and speed is needed. While a recursive estimator like the IMM-PDA is a candidate, in order to operate under low SNR conditions in heavy clutter, a batch estimator is still preferred. In this section, the above problems are addressed by introducing an estimator that uses the ML-PDA with AI *adaptively* in a sliding-window fashion [12], rather than using all the measurements in a single batch as the standard ML-PDA estimator does [17]. The initial time and the length of this sliding window are adjusted adaptively based on the information content in the measurements in the window. Thus, scans with little or no information content are eliminated and the window is moved over to scans with "informative" measurements. This algorithm is also effective when the target is temporarily lost and reappears later. In contrast, recursive algorithms will diverge in this situation and may require an expensive track reinitiation. The standard batch estimator will be oblivious to the disappearance and may lose the whole track. This section demonstrates the performance of the adaptive sliding-window ML-PDA estimator on a real scenario with heavy clutter for tracking a fast-moving aircraft using an EO sensor.

5.1 The Scenario

The adaptive ML-PDA algorithm was tested on an actual scenario consisting of 78 frames of Long Wave Infrared (LWIR) IR data collected during the Laptex data collection, which occurred in July 1996 at Crete, Greece. The sequence contains a single target — a fast-moving Mirage F1 fighter jet. The 920×480 pixel frames, taken at a rate of 1Hz were registered to compensate for frame-to-frame line-of-sight (LOS) jitter. Figure 6 shows the last frame in the F1 Mirage sequence, with the target detection surrounded by an ellipse.

A sample detection list for the Mirage F1 sequence obtained at the end of preprocessing is shown in Figure 7. Each '×' in the figure represents a detection above the threshold. As it can
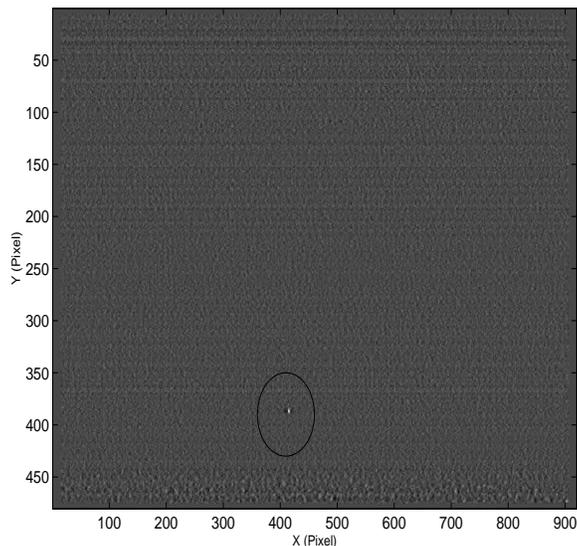
---

this filter should be outlawed.

Figure 6: The last frame in the F1 Mirage sequence

be seen, the number of detections is overwhelming and only a sophisticated algorithm can find the target.

5.2 The ML-PDA Estimator

In this section the target and measurement models used by the estimator in the tracking algorithm are discussed and the statistical assumptions made by the algorithm are presented. The ML-PDA estimator for these models is then outlined.

5.2.1 Target and Measurement Models

The ML-PDA tracking algorithm is used on the detection lists after the data pre-processing phase. It is assumed that there are n detection lists obtained at times $t = t_1, t_2, ..., t_n$. The i-th detection list, where $1 \leq i \leq n$, consists of m_i detections at pixel positions (x_{ij}, y_{ij}) along the X and Y directions. In addition to locations, the signal strength or amplitude, a_{ij}, of the j-th detection in the i-th list, where $1 \leq j \leq m_i$, is also measured. Thus, assuming constant velocity over a number of scans, the problem can be formulated as a 2-dimensional scenario in the sensor's focal plane array with the target motion defined by the 4-dimensional vector consisting of the horizontal and vertical pixel positions of the target at a reference time and the corresponding velocities along these directions, assumed constant.

A measurement can either originate from a true target or from a spurious source. In the former case, each measurement is assumed to have been received only once in each scan with a detection probability P_D and to have been corrupted by zero-mean additive white Gaussian noise of known variance, independent between the coordinates.

Thus, the joint probability density function of the position components of a target originated measurement is the product of two Gaussians with means being known functions of the target parameter vectors and variance given by the measurement noise variances. The false alarms are assumed to be distributed uniformly in the surveillance region and their number at any sampling instant obeys the Poisson probability mass function with known spatial density.
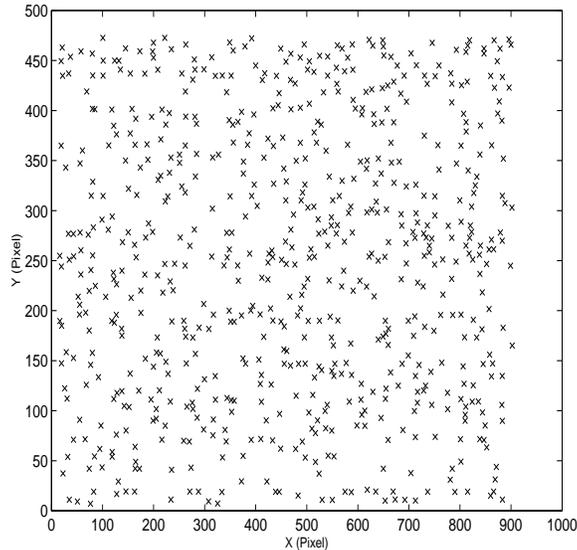
20

Figure 7: Detection list corresponding to the frame in Figure 6

It has been shown in [17] that the performance of the ML-PDA estimator can be improved by using amplitude information (AI) of the received signal in the estimation process itself, in addition to thresholding. After the signal has been passed through the matched filter, an envelope detector can be used to obtain the amplitude of the signal. The noise at the matched filter output is assumed to be narrowband Gaussian. When this is fed through the envelope detector, the output is Rayleigh distributed.

The ratio of the probability density functions at the output of the threshold detector corresponding to detected measurements from the target and false alarms is the amplitude likelihood ratio. This is a ratio of two Rayleigh densities with power $1+d$, for target+noise, and unity power for noise only, where d is the SNR. In IR applications the signal strength, while heavy-tailed, is typically not Rayleigh. However, our experience on real data showed that the overall track extraction results are not very sensitive to the intensity model — using more realistic (mixture) models yielded the same results, so the simple Rayleigh models were kept.

5.2.2 The Maximum Likelihood - Probabilistic Data Association Estimator

The maximum likelihood estimator combined with the PDA approach is obtained, as in Section 3 using the total probability theorem. The total log-likelihood ratio for the entire data set is given as in Sec. 3, by a mixture of Gaussian-Rayleigh($1+d$) with uniform-Rayleigh(1) densities.

The Maximum Likelihood Estimate (MLE) is obtained by finding the vector $\mathbf{x} = \hat{\mathbf{x}}$ that maximizes the total log-likelihood ratio. This maximization is performed using a quasi-Newton (variable metric) method. This can equivalently be done by minimizing the negative log-likelihood function. In our implementation of the MLE, the Davidon-Fletcher-Powell variant of the variable metric method was used. This method is a Conjugate Gradient technique which finds the minimum value of the function iteratively [27]. However, the negative log-likelihood function may have several local minima, i.e., it has multiple modes. Due to this property, if the search is initiated too far away from the global minimum, the line search algorithm may converge to a local minimum. To remedy this, a multi-pass approach is used as in [17].

21

5.3 The Adaptive ML-PDA

Usually, the measurement process begins before the target becomes visible, that is, the target enters the surveillance region of the sensor some time after the sensor started to record measurements. Also, the target may disappear from the surveillance region for a certain period of time before reappearing. During these periods of blackout, the received measurements are purely noise-only and the scans of data contain no information about the target under track. Incorporating these scans into a tracker reduces its accuracy and efficiency. Thus, detecting and rejecting these scans is important to ensure the fidelity of the estimator. This subsection presents a method which uses the ML-PDA algorithm in a variable sliding-window fashion. In this case, the algorithm uses only a subset of the data at a time rather than all of the frames at once, to eliminate the use of scans that have no target. The initial time and the length of the sliding window are adjusted adaptively based on the information content of the data — the smallest window, and thus the fewest number of scans, required to identify the target is determined online and adapted over time.

The key steps in the adaptive ML-PDA estimator are as follows:

1. Start with a window of minimum size.

2. Run the ML-PDA estimator within this window and carry out the validation test on the estimates.

3. If the estimate is accepted (i.e., if the test is passed), and if the window is of minimum size, accept the window. The next window is the present window advanced by one scan. Go to step 2.

4. If the estimate is accepted, and if the window is greater than minimum size, try a shorter window by removing the initial scan. Go to step 2 and accept the window only if estimates are better than those from the previous window.

5. If the test fails and if the window is of minimum size, increase the window length to include one more scan of measurements and, thus, increase the information content in the window. Go to step 2.

6. If the test fails and if the window is greater than minimum size, eliminate the first scan, which could contain pure noise only. Go to step 2.

7. Stop when all scans are used.

The algorithm is described below.
In order to specify the exact steps in the estimator, the following variables are defined:

W = Current window length

$W_{\min}$ = Minimum window length

$Z(t_i)$ = Scan (set) of measurements at time t_i

To illustrate the adaptive algorithm, consider a scenario where a sensor records 10 scans of measurements over a surveillance region. The target, however, appears in this region (i.e., its intensity exceeds the threshold) only after the second scan (i.e., from the third scan onwards). This case is illustrated in Figure 8. The first two scans are useless, because they contain only noise.
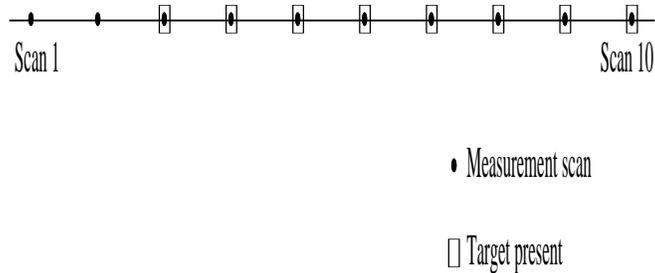
Figure 8: Scenario with a target being present for only a partial time during observation
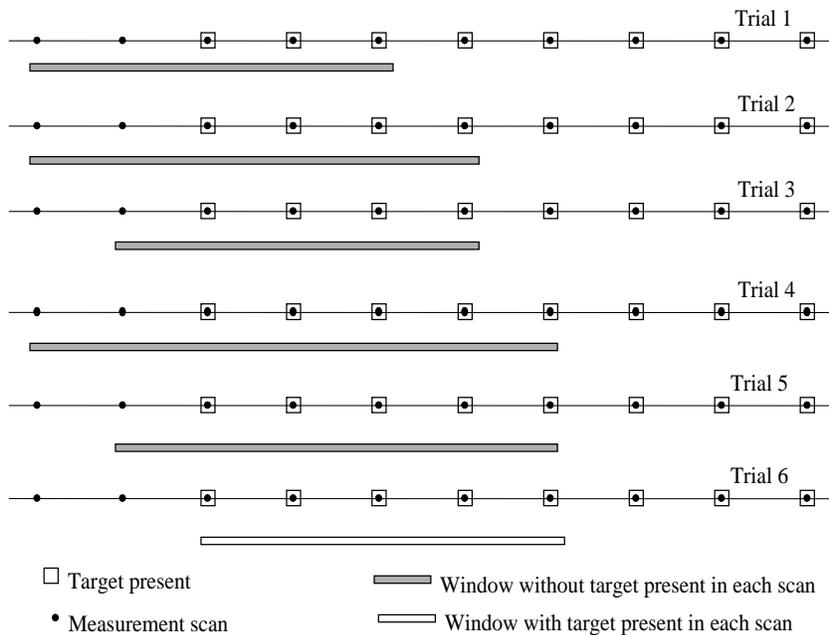

Figure 9: Adaptive ML-PDA algorithm applied to the scenario illustrated above

Consider the smallest window size required for a detection to be 5. Then the algorithm will evolve as shown in Figure 9. First, for the sake of illustration, assume that a single "noisy" scan present in the data set is sufficient to cause the MLE to fail the hypothesis test for track acceptance. The algorithm tries to expand the window to include an additional scan if a track detection is not made. This is done because an additional scan of data may bring enough additional information to detect the target track. The algorithm next tries to cut down the window size by removing the initial scan. This is done to check whether a better estimate can be obtained without this scan. If this initial scan is noise only, then it degrades the accuracy of the estimate. If a better estimate is found (i.e., a more accurate estimate) without this scan, the latter is eliminated. Thus, as in the example given above, the algorithm expands at the front (most recent scan used) and contracts at the rear end of the window to find the best window that produces the strongest detection, based on the validation test.

Since this algorithm can yield, due to data "dropouts", several track segments, if one desires to connect them, the technique of [23] can be used.

5.4 Results

5.4.1 Estimation Results

The Mirage F1 data set on which this algorithm was exercised consists of 78 scans or frames of LWIR data. The target appears late in this scenario and moves towards the sensor. There are about 600 detections per frame. In this implementation the parameters shown in Table 3 were chosen.

Parameter	Value
σ_1	1.25
σ_2	1.25
Min Window Size, W	10
Initial target SNR, d_0	9.5
P_{DC}	0.70
α	0.85
π_m	5%
$\bar{v}$	5.0
$\bar{\sigma}_v$	0.15
K	4

Table 3: Parameters used in the ML-PDA Algorithm for the F1 Mirage Jet

The choice of these parameters is explained as follows:

- σ_1 and σ_2 are the standard deviations along the horizontal and vertical axes respectively. The value of 1.25 for both variables models the results of the preprocessing.

- The minimum window size, W_{min}, was chosen to be 10. The algorithm will expand this window if a target is not detected in 10 frames. Initially a shorter window was used, but the estimates appeared to be unstable. Therefore, less than 10 scans is assumed to be ineffective at producing an accurate estimate.

- The initial target SNR, d_0, was chosen as 9.5 dB because the average SNR of all the detections over the frames is approximately 9.0 dB. However, in most frames, numerous
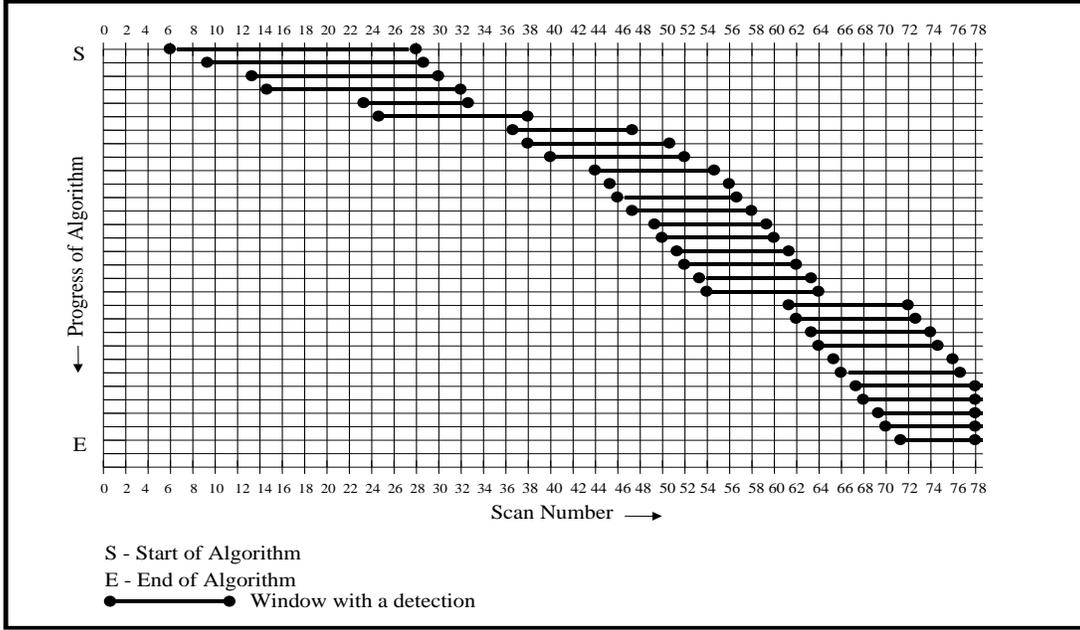
24

Figure 10: Progress of the Algorithm showing Windows with Detections.

random spikes were noted. In the first frame, where a target is unlikely to be present, a single spike of 15.0 dB is noted. These spikes, however, cannot and should not be modeled as the target SNR.

- A constant probability of detection (P_{DC}) of 0.7 was chosen. A value that is too high would bring down the detection threshold and increase P_{FA}.

- α is the parameter used to update the estimated target SNR with an α filter [3]. A high value is chosen for the purpose of detecting a distant target that approaches the sensor over time and to account for the presence of occasional spikes of noise. Thus, the estimated SNR is less dependent on a detection that could originate from a noisy source and, thus, set the bar too high for future detections.

- π_m is the miss probability (probabilities of rejecting a correct track).

- $\bar{v}$ and $\bar{\sigma}_v$ are used in the multi-pass approach of the optimization algorithm [16, 17].

- The number of passes K in the multi-pass approach of the optimization algorithm was chosen as 4 (this is not a significant burden because, as discussed later, the initialization of the search is by far the most time consuming part).

To get a better idea of the detection process, Figure 10 depicts the windows where the target has been detected.

From the above results, note the following:

- The first detection uses 22 scans and occurs at scan 28. This occurs because the initial scans have low-information content as the target appears late in the frame of surveillance. The IMM-MHT algorithm [28] required 38 scans for a detection, while the IMMPDA [22] required 39 scans. Some spurious detections were noticed at earlier scans, but these were rejected.

- The next few detection windows produce similar target estimates. This is because a large number of scans repeat themselves in these windows.

- After the initial detections, there is a 'jump' in the scan number at which a detection is made. In addition, the estimates, particularly the velocity estimates, deteriorate. This could either indicate that the target has suddenly disappeared (became less visible) from the region of surveillance or that the target made a maneuver.

- From scan 44 onward, the algorithm stabilizes for several next windows. At scan 52, however, there is another 'jump' in detection windows. This is also followed by a drop in the estimated target SNR, as explained above. This, however, indicates that the algorithm can adjust itself and restart after a target has become suddenly invisible. Recursive algorithms will diverge in this case.

- From scan 54 onward, the algorithm stabilizes, as indicated by the estimates. Also, a detection is made for every increasing window, because the target has come closer to the sensor and, thus, is more visible. This is noted by the sharp rise in the estimated target SNR after scan 54.

- The above results provide an understanding of the target's behavior. The results suggest that the Mirage F1 appears late in the area of surveillance and moves towards the sensor. However, initially it remains barely visible and possibly undergoes maneuvers. As it approaches the sensor, it becomes more and more visible and, thus, easier to detect.

5.4.2   Computational Load

The adaptive ML-PDA tracker took 442s, including the time for data input/output, on a Pentium III Processor running at 550MHz to process the 78 scans of the Mirage F1 data. This translates into about 5.67s per frame (or 5.67s running time for 1s data), including input/output time. An efficient implementation on a dedicated processor can easily make the algorithm real-time capable on a newer processor. Also, by paralellizing the initial grid search, which required more than 90% of the time, the adaptive ML-PDA estimator can be made even more efficient.

6   Summary

In this article we presented the use of the PDA technique for different tracking problems. Specifically, the PDA approach was used for parameter estimation as well as recursive state estimation. As an example of parameter estimation, track formation of a low observable target using a nonlinear Maximum Likelihood estimator in conjunction with the PDA technique with passive (sonar) measurements was presented. The use of the PDA technique in conjunction with the IMM estimator, resulting in the IMMPDAF, was presented as an example of recursive estimation on a benchmark radar tracking problem in the presence of ECM. Also presented was an adaptive variable-sliding-window PDA-based ML estimator that retains the advantages of the batch (parameter) estimator while being capable of tracking the motion of maneuvering targets. This was illustrated on an real EO surveillance problem. These applications demonstrate the usefulness of the PDA approach for a wide variety of real tracking problems.

Acknowledgment

References

[1] Bar-Shalom, Y. (Editor), *Multitarget-Multisensor Tracking: Advanced Applications*, Volume I, Artech House, Dedham, MA, 1990. Reprinted by YBS Publishing, 1998.

[2] Bar-Shalom, Y. (Editor), *Multitarget-Multisensor Tracking: Applications and Advances*, Volume II, Artech House, Dedham, MA, 1992. Reprinted by YBS Publishing, 1998.

[3] Bar-Shalom, Y., Li, X. R., and Kirubarajan, T., *Estimation with Applications to Tracking and Navigation: Algorithms and Software for Information Extraction*, J. Wiley and Sons, 2001.

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

[5] Blackman, S. S., Dempster, R. J., Busch, M. T., and Popoli, R. F., "IMM/MHT solution to radar benchmark tracking problem", *IEEE Trans. Aerospace and Electronic Systems*, Vol. 35, No. 2, pp. 730-738, April 1999.

[6] Blackman, S. S., and Popoli, R., *Design and Analysis of Modern Tracking Systems*, Dedham, MA: Artech House, 1999.

[7] Blair, W. D., and Watson, G. A., "IMM algorithm for solution to benchmark problem for tracking maneuvering targets", *Proc. SPIE Acquisition, Tracking and Pointing Conference*, Orlando, FL, April 1994.

[8] Blair, W. D., Watson, G. A., and Hoffman, S. A., "Benchmark problem for beam pointing control of phased array radar against maneuvering targets", *Proc. American Control Conference*, Baltimore, MD, June 1994.

[9] Blair, W. D., Watson, G. A., Hoffman, S. A., and Gentry, G. L., "Benchmark problem for beam pointing control of phased array radar against maneuvering targets in the presence of ECM and false alarms", *Proc. American Control Conference*, Seattle, WA, June 1995.

[10] Blair, W. D., Watson, G. A., Kirubarajan, T., and Bar-Shalom, Y., "Benchmark for radar resource allocation and tracking in the presence of ECM", *IEEE Trans. Aerospace and Electronic Systems*, Vol. 34, No. 3, pp. 1015-1022, October 1998.

[11] Chen, J., Leung, Hl, Lo, T., Litva, J., and Blanchette, M. "A Modified Probabilistic Data Association Filter in a Real Clutter Environment", *IEEE Trans. Aerospace and Electronic Systems*, Vol. 32, No. 1, pp. 300-313, January 1996.

[12] Chummun, M. R., Kirubarajan, T., and Bar-Shalom, Y., "An Adaptive Early-Detection ML/PDA Estimator for LO Targets with EO Sensors", *IEEE Trans. Aerosp. Electronic Systems*, 38(2):694-707, April 2002.

[13] Daum, F. E., "Applications of PDA", Seminar at Univ. of Connecticut, May 2002.

[14] Deb, S., Yeddanapudi, M., Pattipati, K. R., Bar-Shalom, Y., "A generalized S-dimensional assignment for multisensor-multitarget state estimation", *IEEE Trans. Aerospace and Electronic Systems*, Vol. 33, No. 2, pp. 523-538, April 1997.

[15] Feo, M., Graziano, A., Miglioli, R., and Farina, A., "IMMJPDA vs. MHT and Kalman Filter with NN Correlation: Performance Comparison", *IEE Proc. on Radar, Sonar and Navigation (Part F)*, Vol. 144, No. 2, pp. 49–56, April 1997.

[16] Jauffret, C., and Bar-Shalom, Y., "Track formation with bearing and frequency measurements in clutter", *IEEE Transactions on Aerospace and Electronic Systems,* AES-26, pp. 999-1010, November 1990.

[17] Kirubarajan, T., and Bar-Shalom, Y., "Target Motion Analysis in Clutter for Passive Sonar Using Amplitude Information", *IEEE Trans. Aerospace and Electronic Systems*, Vol. 32, No. 4, pp. 1367–1384, October 1996.

[18] Kirubarajan, T., and Bar-Shalom, Y., "Probabilistic Data Association Techniques for Target Tracking in Clutter", To appear in the Special Issue on Sequential State Estimation, *IEEE Proceedings*, November 2003.

[19] Kirubarajan, T., Bar-Shalom, Y., Blair, W. D., and Watson, G. A., "IMMPDA Solution to Benchmark for Radar Resource Allocation and Tracking in the Presence of ECM", *IEEE Trans. Aerospace and Electronic Systems*, Vol. 34, No. 3, pp. 1023-1036, October 1998.

[20] Kirubarajan, T., Wang, Y., and Bar-Shalom, Y., "Passive Ranging of a Low Observable Ballistic Missile in a Gravitational Field Using a Single Sensor", *IEEE Trans. Aerosp. Electronic Systems*, AES-37(2):481–494, April 2001.

[21] Lerro, D., and Bar-Shalom, Y., "Interacting Multiple Model Tracking with Target Amplitude Feature", *IEEE Trans. Aerospace and Electronic Systems*, AES-29, No. 2, pp. 494-509, April 1993.

[22] Lerro, D., and Bar-Shalom, Y., "IR Target Detection and Clutter Reduction Using the Interacting Multiple Model Estimator", *Proc. of SPIE Conference on Signal and Data Processing of Small Targets*, Vol. 3373, Orlando, FL, April 1998.

[23] Mucci, R., Arnold, J. and Bar-Shalom, Y., "Track Segment Association with a Distributed Field of Sensors", *J. Acoust. Soc. of America*, 78(4): 1317-1324, Oct. 1985.

[24] Nielsen, R. O., *Sonar Signal Processing*, Boston, MA: Artech House, 1991.

[25] Pattipati, K. R., Popp, R. L., and Kirubarajan, T., "Survey of Assignment Techniques for Multitarget Tracking", Chapter 2 in *Multitarget-Multisensor Tracking: Applications and Advances*, Volume III, Y. Bar-Shalom and W. D. Blair (eds.), Artech House, Dedham, MA, 2001.

[26] Poore, A. B., "Multidimensional assignment formulation of data-association problem arising from multitarget and multisensor tracking", *Computational Optimization and Applications,* Vol. 3, No. 1, pp. 27-57, March 1994.

[27] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and, Flannery, B. P., *Numerical Recipes in C*, Cambridge University Press, 1992.

[28] Roszkowski, S. H., "Common Database for Tracker Comparison", *Proc. SPIE Conf. Signal and Data Processing of Small Targets*, Vol. 3373, Orlando, FL, April 1998.

[29] Sivananthan, S., Kirubarajan, T., and Bar-Shalom, Y., "A Radar Power Multiplier Algorithm for Acquisition of LO Ballistic Missiles Using an ESA Radar", *IEEE Trans. Aerosp. Electronic Systems*, AES-37(2):401–418, April 2001.

[30] Yeddanapudi, M., Bar-Shalom, Y., and Pattipati, K. R., "IMM estimation for multitarget-multisensor air traffic surveillance", *Proc. IEEE*, Vol. 85, No. 1, pp. 80-94, January 1997.