Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2008, Article ID 260186, 13 pages
doi:10.1155/2008/260186

Research Article

ML-PDA: Advances and a New Multitarget Approach

Wayne Blanding,[1] Peter Willett,[2] and Yaakov Bar-Shalom[2]

[1] *Physical Sciences Department, York College of Pennsylvania, York, PA 17405, USA*
[2] *Department of Electrical and Computer Engineering, University of Connecticut, 371 Fairfield Road, Storrs, CT 06269-2157, USA*

Correspondence should be addressed to Wayne Blanding, wblandin@ycp.edu

Received 30 March 2007; Accepted 23 September 2007

Recommended by Roy L. Streit

Developed over 15 years ago, the maximum-likelihood-probabilistic data association target tracking algorithm has been demonstrated to be effective in tracking very low observable (VLO) targets where target signal-to-noise ratios (SNRs) require very low detection processing thresholds to reliably give target detections. However, this algorithm has had limitations, which in many cases would preclude use in real-time tracking applications. In this paper, we describe three recent advances in the ML-PDA algorithm which make it suitable for real-time tracking. First we look at two recently reported techniques for finding the ML-PDA track estimate which improves computational efficiency by one order of magnitude. Next we review a method for validating ML-PDA track estimates based on the Neyman-Pearson lemma which gives improved reliability in track validation over previous methods. As our main contribution, we extend ML-PDA from a single-target tracker to a multitarget tracker and compare its performance to that of the probabilistic multihypothesis tracker (PMHT).

1. INTRODUCTION

The problem of tracking very low observable (VLO) targets in clutter has been an active area of research for a number of years. The term VLO refers to targets with low signal-to-noise ratio (SNR), either because the target is stealthy or because of elevated background noise which masks the target. A key difficulty lies in the relationship between target detection probability (P_d) and false alarm probability (P_{fa}). In order to achieve a value of P_d sufficient to reliably track, one must lower the detection threshold which has the undesirable consequence of increasing P_{fa}. As the false alarm rate increases (increasing clutter), conventional Kalman filter-based tracking algorithms, such as multihypothesis trackers (MHTs) which explicitly form track hypotheses based on hard measurement-to-target associations, rapidly lose efficiency and effectiveness. The number of hypotheses in MHT grows exponentially as the number of measurements increases.

Therefore, new techniques have been developed to track VLO targets. One major class consists of track-before-declare (TBD)—also called track-before-detect—techniques (see, e.g., [1–3]). TBD refers to the fact that these techniques simultaneously perform track estimation and track acceptance (validation or detection) functions. These techniques share common traits. They typically use either unthresholded sensor data or thresholded data with significantly lower thresholds than used with conventional trackers, thereby increasing the measurement data set by one or more orders of magnitude. They usually operate on measurement data over several scans or frames in a batch algorithm to obtain a track estimate. Note that single-frame Bayesian TBD techniques, including those based on particle filters, also exist, for example, [4, 5].

As a consequence of the very low or no detection level thresholding, the computational complexity of TBD algorithms is generally much higher than that of conventional (i.e., Kalman-filter based) trackers. TBD algorithms are therefore better suited to those VLO problems where conventional trackers are unable to initiate or sustain a track. Additionally, as the computational cost is already high, these trackers are also better focused on problems in which contact density is relatively low (i.e., the number of interacting contacts is limited). An example of such an application is in very long range sonar tracking.

One algorithm within this class is the maximum likelihood-probabilistic data association (ML-PDA) tracker. ML-PDA uses low-thresholded measurement data over a batch of measurement frames and computes track estimates using a sliding window. It is a parameter estimation technique which assumes deterministic target motion (no process noise). Originally developed in 1990 [14], it was later enhanced by incorporating measurement amplitude as a feature into the ML-PDA likelihood function [16]. It has been used to track sonar targets using bearings-only and bearing/frequency information [14, 16] as well as tracking an aircraft in an optical data set (infrared) [9]. More recently it has been used on active sonar data sets, including multistatic tracking [5, 30].

Despite its ability to effectively track VLO targets, ML-PDA has suffered from some limitations. As with most TBD algorithms, it has high-computational complexity and as the clutter level increases beyond a certain (problem-specific) point it can no longer perform real-time tracking without resorting to parallel processing. Second, because ML-PDA always provides a track estimate, some form of track validation must take place to determine if the estimate is the result of an actual target or from noise-due measurements. The challenge for track validation lies in obtaining the appropriate statistical distributions from which to perform the correct hypothesis test. And finally ML-PDA, in its original formulations, is restricted to single-target tracking. In this paper, we review recent advances that alleviate the first two limitations which brings context to the major contribution of this paper—extending ML-PDA to a multitarget tracking algorithm.

First, we briefly describe two recently reported techniques for obtaining the ML-PDA track estimate which have been shown to be significantly more efficient than the previous method used. These techniques are later used in the multitarget version of ML-PDA.

Second, we describe work recently reported on an ML-PDA track validation procedure based on application of the Neyman-Pearson lemma. We show using extreme value theory that the statistics of the LLR global maximum under the "no-target" hypothesis is more closely approximated by the Gumbel distribution as opposed to the Gaussian distribution used by earlier researchers.

As our main contribution, we extend the ML-PDA algorithm to jointly estimate the parameters of multiple targets in a joint ML-PDA (JMLPDA) algorithm. By use of measurement validation gating techniques, we incorporate ML-PDA and JMLPDA into a multitarget ML-PDA (MLPDA (MT)) tracking system. Comparisons are made between MLPDA (MT) and the probabilistic multihypothesis tracker (PMHT). PMHT is a multitarget tracking algorithm which has good computational efficiency characteristics [26, 30].

The remainder of this paper is organized as follows. Section 2 defines the terminology and gives a summary of the ML-PDA algorithm. Section 3 describes the computational efficiency improvements in ML-PDA by use of the genetic search and the directed subspace search techniques. Section 4 summarizes the ML-PDA track validation procedure. Section 5 derives the JMLPDA for multitarget tracking. Section 6 outlines the ML-PDA (MT) procedure to track multiple targets and presents the comparison between MLPDA (MT) and PMHT in a 2-target scenario. Section 7 summarizes.

2. ML-PDA PROBLEM FORMULATION

The ML-PDA algorithm was originally developed for use in passive narrowband target motion analysis for LO targets [14] and was later extended to incorporate amplitude information to handle VLO targets [16]. In the window-based ML-PDA algorithm, designed for use in real-time applications, a subset of the N_w most recent data frames is used to compute the track estimate. When a new frame of data is received, the ML-PDA algorithm is repeated after adding the new frame and deleting one or more of the oldest frames from the data set, in effect creating a variable sliding window for track detection and update.

2.1. ML-PDA derivation

A detailed derivation of the ML-PDA algorithm incorporating amplitude information in a 2D measurement space can be found in [16]. A summary of the ML-PDA algorithm incorporating amplitude information is presented in this section, generalized to arbitrary sized measurement and parameter spaces. The ML-PDA algorithm uses the following assumptions.

(1) A single target is present in each data frame with a given detection probability (P_d) and detections are independent across frames.

(2) At most one measurement per frame corresponds to the target.

(3) The target operates according to deterministic kinematics (i.e., no process noise).

(4) False detections are distributed uniformly in the search volume (U).

(5) The number of false detections is Poisson distributed according to probability mass function $\mu_f(m)$, with parameter λ (spatial density), a function of the detector P_{fa} in a resolution cell, independent across frames.

(6) The amplitudes of target originated and false detections are distributed according to pdf $p_1(a)$ and $p_0(a)$, respectively. The target SNR, which affects $p_1(a)$, is either known or estimated in real time.

(7) Target originated measurements are corrupted with additive zero-mean white Gaussian noise.

(8) Measurements obtained at different times are, conditioned on the target state, independent.

The target parameter, $\mathbf{x}_r$, consists of the target kinematic state at a given reference time and is related to the target state at any time using the (possibly nonlinear) relation

$$\mathbf{x}(i) = \mathbf{F}(\mathbf{x}_r, i). \tag{1}$$

The measurement set is given by

$$\begin{aligned}(\mathbf{Z},\mathbf{a}) = \{(\mathbf{Z}_i,\mathbf{a}_i)\} &= \{(\mathbf{z}_{ij},a_{ij})\}.\\ i &= 1,2,\dots,N_w \text{ frame number},\\ j &= 1,2,\dots,m_i \text{ measurement number},\end{aligned} \tag{2}$$

where $\mathbf{z}_{ij}$ consists of the kinematic measurement and a_{ij} the measurement amplitude.[1] Amplitude refers to the envelope output of the detector in a single resolution cell [18, 28]. Measurements with a single subscript refer to all measurements in a single data frame. Measurements with two subscripts identify a specific measurement.

There are some cases where assumption (2) above breaks down and the target may appear in more than one measurement cell in a single frame of data. This may occur in an active sonar or radar sensor when the target extent exceeds one resolution cell, or for either passive or active sensors when the target signal strength is high enough such that detectable energy above the detector threshold is received in adjacent cells or beams. In such cases, one can use redundancy elimination logic [3] such as centroiding detections to eliminate or consolidate the multiple target-originated measurements. Such logic however may have the undesirable effect in a multiple interacting target scenario of masking the weaker target when its detections are adjacent to a stronger target—the detections would be combined into a single centroided detection.

A measurement, assuming it is target originated, is related to the parameter $\mathbf{x}_r$ using the (possibly nonlinear) relation

$$\mathbf{z} = \mathbf{H}(\mathbf{x}_r,\mathbf{x}_s(i),i) + \mathbf{w}_i, \tag{3}$$

where $\mathbf{w}_i$ is a zero-mean white Gaussian noise with known covariance matrix $\mathbf{R}$. The sensor kinematic state, $\mathbf{x}_s(i)$, is included to account for (known) sensor motion. From this we obtain for a target originated measurement

$$p(\mathbf{z}_{ij} \mid \mathbf{x}_r) = \mathcal{N}(\mathbf{z}_{ij};\mathbf{H}(\mathbf{x}_r,\mathbf{x}_s(i),i),\mathbf{R}). \tag{4}$$

The maximum likelihood approach finds the target parameter that maximizes the likelihood function, $p(\mathbf{Z},\mathbf{a} \mid \mathbf{x}_r)$. When incorporating amplitude into the likelihood function, it is convenient to define an amplitude likelihood ratio as

$$\rho_{ij} = \frac{p_1(a_{ij} \mid \tau)}{p_0(a_{ij} \mid \tau)}, \tag{5}$$

where τ is the detector threshold (in each resolution cell) and the conditioning is on the amplitude exceeding the threshold, $a_{ij} > \tau$. For many applications (including those used in this paper), the Rayleigh distribution is used which corresponds to a Swerling-I target fluctuation model.

From these assumptions and definitions, the likelihood function becomes

$$\begin{aligned} &p(\mathbf{Z},\mathbf{a} \mid \mathbf{x}_r)\\ &\quad= \prod_{i=1}^{N_w} p(\mathbf{Z}_i,\mathbf{a}_i \mid \mathbf{x}_r)\\ &\quad= \prod_{i=1}^{N_w}\Bigg[(1-P_d)\prod_{j=1}^{m_i} p(\mathbf{z}_{ij},a_{ij} \mid \text{“clutter”})\\ &\qquad\quad + \frac{P_d}{m_i}\sum_{j=1}^{m_i} p(\mathbf{z}_{ij},a_{ij} \mid \mathbf{x}_r)\prod_{l\neq j} p(\mathbf{z}_{il},a_{il} \mid \text{“clutter”})\Bigg]\\ &\quad= \prod_{i=1}^{N_w}\Bigg[\frac{1-P_d}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i} p_0(a_{ij} \mid \tau)\\ &\qquad\quad + \frac{P_d\mu_f(m_i-1)}{U^{m_i-1}m_i}\prod_{j=1}^{m_i} p_0(a_{ij} \mid \tau)\sum_{j=1}^{m_i} p(\mathbf{z}_{ij} \mid \mathbf{x}_r)\rho_{ij}\Bigg]. \end{aligned} \tag{6}$$

The above equation represents the weighted sum of *all* the likelihoods of associating a specific measurement (or no measurement) to the target with all other measurements false detections. This is obtained using the total probability theorem and is the essence of the PDA approach [1].

Dividing (6) by the likelihood function given that all measurements are false detections, namely,

$$\prod_{i=1}^{N_w}\left[\frac{1}{U^{m_i}}\mu_f(m_i)\prod_{j=1}^{m_i} p_0(a_{ij} \mid \tau)\right], \tag{7}$$

and taking the logarithm of the resulting function, a more compact form (the log-likelihood ratio (LLR)) is obtained and is given by

$$\Lambda(\mathbf{Z},\mathbf{a} \mid \mathbf{x}_r) = \sum_{i=1}^{N_w} \ln\left[(1-P_d) + \frac{P_d}{\lambda}\sum_{j=1}^{m_i}\rho_{ij}p(\mathbf{z}_{ij} \mid \mathbf{x}_r)\right]. \tag{8}$$

The ML-PDA track estimate, $\hat{\mathbf{x}}(k)$, is given by the LLR global maximum in the parameter space

$$\hat{\mathbf{x}}(t) = \arg\max_{\mathbf{x}_r} \Lambda(\mathbf{Z},\mathbf{a} \mid \mathbf{x}_r), \tag{9}$$

where t is the reference time corresponding to the target parameter $\mathbf{x}_r$.

The reference time for parameter $\mathbf{x}_r$ can be selected arbitrarily. Referencing the parameter to the middle of the ML-PDA batch yields a track estimate which minimizes track errors, but which also induces a time latency in the track estimate. Referencing the parameter to the end of the ML-PDA batch (i.e., the most recent time) yields larger estimation errors, but without the time latency.

As with any tracking algorithm, in order for a track estimate to have a finite covariance matrix the system must have the property of observability; see, for example, [19].

3. ML-PDA EFFICIENCY IMPROVEMENTS

Because the ML-PDA algorithm is essentially a maximum-likelihood method, the track estimate is the parameter value

[1] Any other feature with a probabilistic model can also be used.

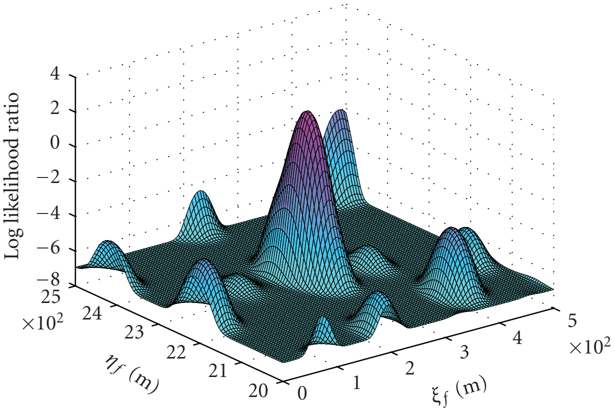

Figure 1: Representative LLR surface at a velocity maximizing the center peak. This figure is repeated from [6].

which maximizes the ML-PDA LLR. The LLR is a highly nonconvex function which contains many (from several hundred to over a thousand) local maxima. Additionally the LLR surface contains large regions where the LLR is near its minimum value and a near-zero gradient. Figure 1 illustrates the complexity of the LLR surface by showing a small representative region of the parameter space over position with velocity fixed, for a 4-dimensional parameter (two position dimensions, two velocity dimensions). The measurement space consists of range, bearing, and range rate simulating an active sonar problem.

Three methods to compute the LLR global maximum have been reported in the literature. Prior researchers used a multipass grid (MPG) search to find the LLR global maximum [14, 16]. The genetic search (GS) and directed subspace search (DSS) have been shown to perform better than the MPG in terms of reduced computational complexity (averaging an order of magnitude) and increased ability to distinguish the LLR global maximum (3 dB improved performance) [6]. As these two methods are used in later simulations, they are summarized below.

3.1. Genetic search

The GS is a stochastic search technique which is motivated from evolutionary biology and "survival of the fittest." While this technique has seen little use in the tracking community, it has been used effectively to solve many complex optimization problems.

More fully described in [12], the genetic search mimics biological evolution in that the ML-PDA parameter (described by a binary bit string) is analogous to biological DNA. Starting with a (randomized) population of parameter values, each population member is assigned a fitness value based on the LLR evaluated at each population member's parameter. Population members, the parents of the current generation, are selected for reproduction based on their fitness value. The "fitter" population members (those with higher LLR values) are more likely to reproduce with the best members reproducing multiple times (i.e., bearing more children).

Population members selected for reproduction are randomly paired with other population members selected for reproduction. With a given probability, the two parents will produce two children which each share characteristics (pieces of the parameter bit string) of each parent; otherwise the children will be clones of the parents. Parents selected to reproduce multiple times will be paired with different mates.

The children then become the parents of the next generation. As this process is propagated over generations, the population becomes more fit and will eventually converge to a single parameter which is then taken as the LLR global maximum.

The GS implemented in this paper[2] is an "off the shelf" implementation using the techniques from [12]. While there are few theoretical results which guarantee the convergence of the GS to the global maximum of an arbitrary function, by tuning our algorithm for both speed of execution as well as effectiveness at finding the global maximum of the ML-PDA LLR, results, described more fully in [6], show improvement in both speed of execution as well as effectiveness compared to the MPG search.

3.2. Directed subspace search

The second method for maximizing the ML-PDA LLR is a recently developed technique called the directed subspace search. The DSS is motivated from the desire to use information from measurement data to help guide the search for the LLR global maximum. Grid searches and GSs are general optimization tools that do not take advantage of the structure of the objective function to guide the search process. By using the structure of the objective function to identify areas in the parameter space that are more likely to contain local or global maxima, a more efficient search is possible. Considering that in the active sonar application presented here, about 70% of the LLR surface is at the floor value (meaning $\sum_{i=1}^{N_w} \log(1 - P_d)$ in (8)), bypassing these areas becomes desirable.

First we observe that in many tracking applications the measurement space is a subspace of the parameter space. In the 3D measurement space described in this section (bearing, range, range rate), one can map (bearing, range) to the Cartesian parameter positions. Range rate is equivalent to radial velocity (referenced to the sensor), leaving tangential velocity as a "free" parameter.

Next we observe that LLR maxima can only result from the aggregation of one or more measurements in the N_w-frame data set that closely fit a parameter $\mathbf{x}_r$. In regions of the parameter space where no measurements influence the LLR (i.e., where $p(\mathbf{z}_{ij} \mid \mathbf{x}_r) \approx 0 \; \forall i, j$ from (8)), the LLR will be at the floor value. In regions of the parameter space where only a single measurement influences the LLR, the LLR will

[2] A more detailed description of the specific GS algorithm used can be found in [6].

Table 1: Directed subspace search algorithm.

Step	Action
1	Set grid density for free parameter(s)
2	Map one measurement to parameter space
3	Using the measurement, compute LLR values over grid of free parameter(s)
4	Repeat steps 2, 3 for all measurements in data set
5	Pass best result to local optimization routine

be at a local maximum. That is for the bearing, range, range rate measurement, the LLR will be at a constant (local maximum) value for all values of tangential velocity. In areas of the parameter space where two or more measurements influence the LLR, each measurement (bearing, range, range rate) will lie in the vicinity of the local maximum produced by the measurements.

Therefore, by mapping measurements to the parameter space and searching only those regions of parameter space where measurements exist and can contribute to an increase in the LLR, one narrows down the parameter space of interest and bypasses those regions where it is known (from the lack of supporting measurements) that the LLR is near the floor value. This effect produces the first advantage of the DSS search over the MPG search or the GS—a reduced search volume containing a subset of the full parameter space and consequently improved computational efficiency.

The DSS search algorithm, outlined in Table 1, is designed to search this reduced parameter space. In the DSS search, we take each measurement in the N_w-frame data set, map it to parameter space, and compute the LLR. Since this mapping leaves one or more free parameters which can take on any value for a given measurement, the LLR is computed over a set of values defined by the measurement and a grid of values of the free parameter(s). For example, using a measurement space of bearing, range, range rate, one computes the LLR at the bearing, range, range rate given by the measurement over a grid of tangential velocities. This process is repeated for every measurement in the measurement set. Figure 2 illustrates the DSS search. Three measurements are shown plotted in position subspace along with their corresponding radial velocity vectors. The grid of tangential velocity points is overlaid on each measurement. The LLR is evaluated at each tangential velocity and for each measurement.

Once the LLR is computed over the set of grid points of the free parameter(s) for each measurement in the N_w-frame data set, the parameter that gives the maximum LLR value is taken and used to initialize a local optimization algorithm (we use the Davidon-Fletcher-Powell algorithm [21]). The reason a local optimization algorithm is needed is that while the DSS grid search will return the local maximum from any single-measurement maximum, it will only return parameter values in the vicinity of local maxima caused by two or more measurements, not the maximum itself. The final, converged parameter from the local optimization algorithm is the DSS estimate of the LLR global maximum.

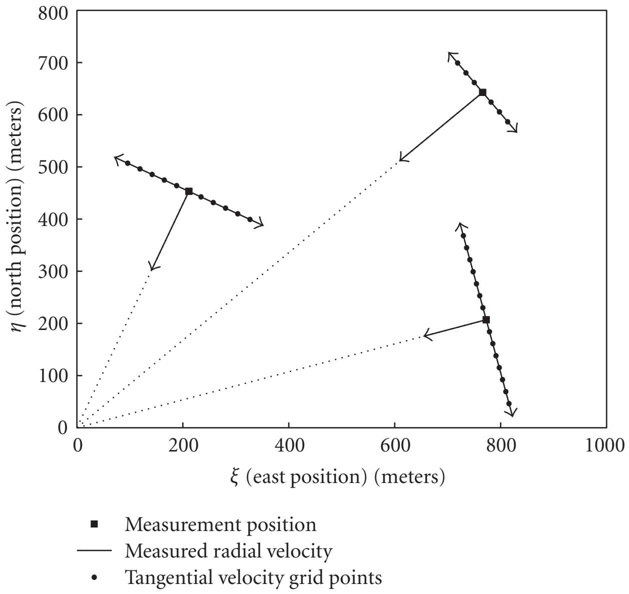

Figure 2: Three measurements (position, radial velocity) overlaid with their respective DSS search grid points of tangential velocity. Vectors represent velocities. Sensor is at origin.

4. ML-PDA TRACK VALIDATION

Since the ML-PDA track estimate is the location of the ML-PDA LLR global maximum in the parameter space, ML-PDA will always return a track estimate even when a target is not present. Therefore, a reliable means of validating the track estimate as target-originated is required. This becomes a hypothesis testing problem—given the value of the LLR global maximum, is this value more consistent with a "target present" (H_1) or a "target absent" (H_0) hypothesis?

According to the Neyman-Pearson lemma [17], the most powerful test of H_0 versus H_1 is given by comparing the likelihood ratio (or log-likelihood ratio) to a threshold. If a valid track estimate exists, then by using ML principles, it is given by the location of the LLR global maximum. Therefore, the test becomes determining if the LLR global maximum is more likely to have been formed from only noise-originated measurements (H_0) or target-plus noise-originated measurements (H_1). The threshold for this test is selected to maximize the power of the test, P_{DT} (true track detection probability), at a given size or level of significance, P_{FT} (false track acceptance probability). Thus the threshold value, γ, is chosen based on the statistics of the global LLR maximum under H_0,

$$\gamma = F_w^{-1}(1 - P_{\mathrm{FT}}), \tag{10}$$

where $F_w(w)$ is the cumulative distribution function of the LLR global maximum. If $F_w(w)$ is known exactly, this hypothesis test becomes the optimal test as it obeys all conditions required by the Neyman-Pearson lemma. However, since we do not a priori know this distribution, optimality of this test is not guaranteed.

4.1. The LLR global maximum under H_0

Previous researchers assumed that the distribution of the LLR global maximum under H_0 was Gaussian based on the central limit theorem (CLT) [14, 16]. More recently using extreme value theory, the Gumbel distribution has been shown to be both a better theoretical model and a closer match to the empirical distribution obtained from Monte Carlo simulations [7]. The theoretical analysis is summarized next.

The LLR global maximum can be viewed as the maximum from the set of all LLR local maxima. Define the random variable y with cumulative distribution function (cdf) $F_y(y)$ to be the value of an LLR local maximum and the random variable w with cdf $F_w(w)$ to be the value of the LLR global maximum. Then using the formula for the distribution of the maximum order statistic from a set of M LLR local maxima (samples) [20],

$$F_w(w) = \left[F_y(w)\right]^M. \tag{11}$$

Implicit is the assumption that the LLR local maxima are i.i.d. The independence assumption is not strictly valid in that LLR local maxima which share measurements will be correlated to some extent. However, it can be considered a good approximation because the maxima are generally well separated in the parameter space (see Figure 1) and noise-related maxima will principally result from groupings of a small number (one or two) of measurements. The small number of measurements contributing to an LLR maximum limits the correlation between maxima for a given measurement data set. These assumptions remain valid over a wide range of problem formulations and typical values of P_{fa}.

Further one can consider the pdf of the LLR local maximum to be a mixture distribution. Let the random variable y represent the value of an LLR local maximum with pdf $f_y(y)$. Each component of the mixture is distributed according to $f_y^i(y)$ with the superscript indicating the number of measurements that associate to form the LLR local maximum. The probability of an LLR local maximum consisting of i associated measurements is denoted p_i. Theoretically i can take on values from 1 to the total number of measurements in the data set. The distribution $f_y(y)$ can therefore be expressed by

$$f_y(y) = \sum_i p_i f_y^i(y). \tag{12}$$

Absent conditioning on the number of measurements associated with an LLR local maxima, the LLR local maxima can be considered to be identically distributed according to the mixture distribution described in (12).

EVT describes the asymptotic (large sample size) behavior of the largest value from an i.i.d. sample of size M from a distribution with a cdf $F_y(y)$, and is well developed in the statistical literature [10, 13]. Let

$$w = \max\{y_1, y_2, \dots, y_M\}. \tag{13}$$

EVT states that if a limiting cdf of w exists as $M \to \infty$, then that distribution must belong to one of three forms (Gumbel, Weibull, or Frechet). The distribution appropriate to a specific application is based on the support of the underlying distribution of $F_y(y)$. The Gumbel cdf is the appropriate distribution in our application because the support of the distribution of the LLR local maximum is restricted to $0 < y < \infty$, and is of the form

$$F_w(w) = \exp\{-\exp[-a_n(w - u_n)]\}, \tag{14}$$

where a_n and u_n are the scale and location parameters for the distribution and which depend on the number of samples used in (13).

The level of accuracy to which the Gumbel distribution approximates the distribution of the LLR global maximum is affected by two important issues.

(1) There is no guarantee that an asymptotic distribution exists for the given $f_y(y)$. Specifically the structure of $f_y(y)$ as a mixture distribution described by (12) may preclude the existence of an asymptotic distribution.

(2) While the number of LLR local maxima is large, $F_w(w)$ may not have reached its asymptotic distribution. It has been noted for example that while the maximum from samples of an exponential distribution attains the asymptotic distribution with a relatively small number of samples (fast convergence), for a Gaussian distribution a much larger sample size is required to attain the asymptotic distribution (slow convergence) [13].

4.2. Methods to estimate $F_w(w)$

Two methods were used in [7] to estimate the Gumbel distribution parameters, and to thereby estimate the distribution of the LLR global maximum—an offline method and a real-time method. In the offline method used in the simulations described later, the tracking problem is repeatedly simulated under H_0 to obtain a set of LLR global maxima. Then maximum likelihood techniques [13] are used to estimate the Gumbel parameters.

This method has the advantage of yielding an optimal (in the ML sense) estimate of the Gumbel distribution parameters, although as has been previously stated the Gumbel distribution is only an approximation to the true distribution of $F_w(w)$. This approach may be impractical due to the extensive offline simulations required. For a general-purpose tracking system using this methodology, separate sets of Gumbel distribution parameters must be estimated for the full range of possible P_{fa}, target SNR, and N_w as well as for variations in the boundaries and volumes of the measurement and parameter spaces since each of these factors affect either the number of local LLR maxima or the distribution $f_y(y)$ or both. If the system were designed for a single special purpose use, this method may be advantageous.

5. JOINT ML-PDA

In this section, we derive the multitarget version of ML-PDA, called joint ML-PDA (JMLPDA). The derivation of the JMLPDA algorithm is similar to that of ML-PDA. In this section, the JMLPDA formulation for obtaining the joint track estimate of $K = 2$ targets is presented. JMLPDA can be further extended to jointly estimate any number of targets by extending the JLLR framework in this section to K targets.

5.1. JMLPDA derivation

The assumptions from Section 2 used in ML-PDA are also used in JMLPDA and are supplemented by the following additional assumptions.

(1) K previously confirmed targets exist.
(2) At most one measurement per frame corresponds to each target.
(3) A measurement cannot be associated to more than one target.
(4) Measurements originating from different targets are independent.
(5) Target originated measurement errors have the same distribution for each target (i.e., are a function of the sensor, not the target).

The parameter to be estimated is the kinematic state of all targets at a given reference time

$$\mathbf{x}_r = \begin{bmatrix} \mathbf{x}_r^{1T} & \cdots & \mathbf{x}_r^{KT} \end{bmatrix}^T, \tag{15}$$

where $\mathbf{x}_r^k$ is the kinematic state of the kth target whose target motion model is described by

$$\mathbf{x}^k(i) = \mathbf{F}^k(\mathbf{x}_r^k, i). \tag{16}$$

The measurement set is given by

$$\begin{aligned}(\mathbf{Z}, \mathbf{a}) = \{(\mathbf{Z}_i, \mathbf{a}_i)\} &= \{(\mathbf{z}_{ij}, a_{ij})\}, \\ i = 1, 2, \dots, N_w &\quad \text{frame number}, \\ j = 1, 2, \dots, m_i &\quad \text{measurement number},\end{aligned} \tag{17}$$

where $\mathbf{z}_{ij}$ consists of the kinematic measurement and a_{ij} the measurement amplitude. Amplitude refers to the envelope output of the detector in a single resolution cell [18, 28]. Measurements with a single subscript refer to all measurements in a single data frame. Measurements with two subscripts identify a specific measurement.

Because the SNR of each target can be different, the measurement amplitude likelihood ratio must be defined for each target. The amplitude likelihood ratio for the kth target is now given by

$$\rho_{ij}^k = \frac{p_1(a_{ij} \mid \tau, H_k)}{p_0(a_{ij} \mid \tau)}, \tag{18}$$

where τ is the detector threshold (in each resolution cell) and the pdf is conditioned on the amplitude exceeding τ and that the measurement originates from the kth target (hypothesis H_k).

A measurement, assuming it is target originated, is related to the kth target $\mathbf{x}_r^k$ using the (possibly nonlinear) relation

$$\mathbf{z} = \mathbf{H}^k(\mathbf{x}_r^k, \mathbf{x}_s(i), i) + \mathbf{w}_i, \tag{19}$$

where $\mathbf{w}_i$ is a zero-mean white Gaussian noise with covariance matrix $\mathbf{R}$ and $\mathbf{x}_s(i)$ is the sensor kinematic state such that for a target originated measurement

$$p(\mathbf{z}_{ij} \mid \mathbf{x}_r^k) = \mathcal{N}(\mathbf{z}_{ij}; \mathbf{H}^k(\mathbf{x}_r^k, \mathbf{x}_s(i), i), \mathbf{R}). \tag{20}$$

In the case where $K = 2$ targets, the joint likelihood function, $p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r)$, for a single frame of data is formed as the weighted sum of four terms corresponding to the four possible target detection events. Let

$$\begin{aligned} L_i^0 &= p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r, \text{no target detections}), \\ L_i^1 &= p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r, \text{only target 1 detected}), \\ L_i^2 &= p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r, \text{only target 2 detected}), \\ L_i^{12} &= p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r, \text{both targets detected}). \end{aligned} \tag{21}$$

Then $p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r)$ is given by

$$\begin{aligned} p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r) = {} & (1 - P_d^1)(1 - P_d^2)L_i^0 + P_d^1(1 - P_d^2)L_i^1 \\ & + (1 - P_d^1)P_d^2 L_i^2 + P_d^1 P_d^2 L_i^{12}, \end{aligned} \tag{22}$$

where P_d^k is the single frame detection probability of the kth target.

The individual terms, L_i^k, are formed by associating a specific measurement to each detected target with all other measurements considered as false detections and are given by

$$\begin{aligned} L_i^0 &= \frac{\mu_f(m_i)}{U^{m_i}} \prod_{j=1}^{m_i} p_0(a_{ij} \mid \tau), \\ L_i^1 &= \frac{\mu_f(m_i - 1)}{U^{m_i - 1} m_i} \prod_{j=1}^{m_i} p_0(a_{ij} \mid \tau) \sum_{j=1}^{m_i} p(\mathbf{z}_{ij} \mid \mathbf{x}_r^1) \rho_{ij}^1, \\ L_i^2 &= \frac{\mu_f(m_i - 1)}{U^{m_i - 1} m_i} \prod_{j=1}^{m_i} p_0(a_{ij} \mid \tau) \sum_{j=1}^{m_i} p(\mathbf{z}_{ij} \mid \mathbf{x}_r^2) \rho_{ij}^2, \\ L_i^{12} &= \frac{\mu_f(m_i - 2)}{U^{m_i - 2} m_i (m_1 - 1)} \prod_{j=1}^{m_i} p_0(a_{ij} \mid \tau) \\ & \quad \times \sum_{j=1}^{m_i} \sum_{\substack{l=1 \\ l \neq j}}^{m_i} p(\mathbf{z}_{ij} \mid \mathbf{x}_r^1) p(\mathbf{z}_{il} \mid \mathbf{x}_r^2) \rho_{ij}^1 \rho_{il}^2. \end{aligned} \tag{23}$$

The joint likelihood function considering all N_w frames of data is the product of the single frame joint likelihood functions

$$p(\mathbf{Z}, \mathbf{a} \mid \mathbf{x}_r) = \prod_{i=1}^{N_w} p(\mathbf{Z}_i, \mathbf{a}_i \mid \mathbf{x}_r). \tag{24}$$

The joint log-likelihood ratio (JLLR) is obtained by dividing (24) by the likelihood that all measurements are noise originated, $\prod_{i=1}^{N_w}(1-P_d^1)(1-P_d^2)L_i^0$, and taking the logarithm of the result yielding

$$\begin{aligned}\Lambda(\mathbf{Z},\mathbf{a}\mid\mathbf{x}_r)=\sum_{i=1}^{N_w}\ln\Bigg[1&+\frac{P_d^1}{\lambda(1-P_d^1)}\sum_{j=1}^{m_i}p(\mathbf{z}_{ij}\mid\mathbf{x}_r^1)\rho_{ij}^1\\&+\frac{P_d^2}{\lambda(1-P_d^2)}\sum_{j=1}^{m_i}p(\mathbf{z}_{ij}\mid\mathbf{x}_r^2)\rho_{ij}^2\\&+\frac{P_d^1P_d^2}{\lambda^2(1-P_d^1)(1-P_d^2)}\\&\times\sum_{j=1}^{m_i}\sum_{\substack{l=1\\l\neq j}}^{m_i}p(\mathbf{z}_{ij}\mid\mathbf{x}_r^1)p(\mathbf{z}_{il}\mid\mathbf{x}_r^2)\rho_{ij}^1\rho_{il}^2\Bigg].\end{aligned}\tag{25}$$

The global maximum of the JLLR defines the parameter estimate, $\hat{\mathbf{x}}_r$, and gives the track estimate of the two targets. A separate test must be performed to determine if the track estimates are the result of noise or target originated measurements (a test for target existence).

The extension of JMLPDA to an arbitrary number of targets is straightforward in that one must extend (22) to become the weighted sum of all possible target detection events for the given number of targets. The number of terms in this function, however, increases exponentially with the number of targets, K, according to 2^K. An additional consideration is that one is maximizing the JLLR over $4K$ dimensions (assuming 4-dimensional state vector for each target). The computational cost of maximizing this function therefore grows with the number of targets. Some level of separation of the $4K$-dimensional problem could be exploited in that for a given frame of data in the batch, not all targets may be interacting with each other. This would lead to a reduced computational complexity ranging from that of K four-dimensional problems to one $4K$-dimensional problem depending upon the level of separation achieved. Based on these considerations, application of JMLPDA to more than 3 targets may not be practical. In this paper, we consider only the 2-target JMLPDA case.

The JMLPDA algorithm also assumes that the number of targets is known. In the context of a multitarget application, this knowledge comes from the prior target state estimates. The presence of a new (previously undetected) or spawned target in the measurement set will cause JMLPDA to behave in an unpredictable way but will generally not give accurate track estimates. If the number of targets is fewer than that assumed in JMLPDA (i.e., a target death event occurs), the target validation procedure will correctly not validate the track estimate for the nonexistent target(s).

5.2. *JMLPDA track validation*

A procedure for JMLPDA track validation along the same lines as ML-PDA track validation appears feasible. However, the computational complexity associated with JMLPDA track estimates could make implementation difficult. Further, one must account for all possible combinations of track validation results for each target (e.g., target 1 valid/target 2 invalid). Therefore, for simplicity, we apply directly the ML-PDA track validation technique to the JMLPDA track estimates using an adjusted measurement data set described next.

Table 2: JMLPDA track validation procedure.

Step	Action
1	Find global JLLR maximum
2	Identify measurement-to-target associations for all targets
3	Select a target
4	Edit out measurements (using the complete measurement set) associated with all other targets
5	Compute single-target LLR at selected target's parameter estimate
6	Validate estimate using the off-line track validation threshold
7	Repeat steps 3–6 for all targets

The procedure for obtaining JMLPDA track estimates is summarized in Table 2. To perform track validation, one first obtains the joint track estimates for each target using the JMLPDA algorithm. Then based on the track estimates at each frame in the batch, one obtains the posterior likelihood that each measurement in the data set is associated with each target, similar to the PDA approach [1]. From these association probabilities, the measurement with the highest association probability is associated with each target. If multiple targets share the same "most likely" measurement, the measurement is associated to the target with the largest association probability between the two targets (a greedy approach) and the remaining targets associate with their next most likely measurement. As the posterior association probabilities account for the possibility of associating none of the measurements to a target, a measurement is associated to a target only if the posterior association probability for that measurement exceeds the posterior probability of associating none of the measurements to the target.

Once these hard measurement-to-target associations are made, to validate the track estimate for a single target the measurement data set is modified by editing out those measurements that are associated to all other targets. Then the LLR is computed at the track estimate of the target under test using the ML-PDA LLR of (8) and compared to the ML-PDA track validation threshold. The ML-PDA track validation threshold is obtained using the procedure outlined in Section 4. Thus the JMLPDA track validation problem is reformulated into an ML-PDA track validation problem.

6. MULTITARGET ML-PDA

Multitarget ML-PDA is a tracking system which incorporates all phases of the tracking problem: track initiation, track maintenance/update, and track termination functions and uses the ML-PDA and JMLPDA algorithms for track update. Figure 3 shows a flowchart of the actions taken by the tracking system upon receipt of a new frame of data. The following subsections describe in more detail how the measurement gating is carried out, the track validation for ML-PDA and

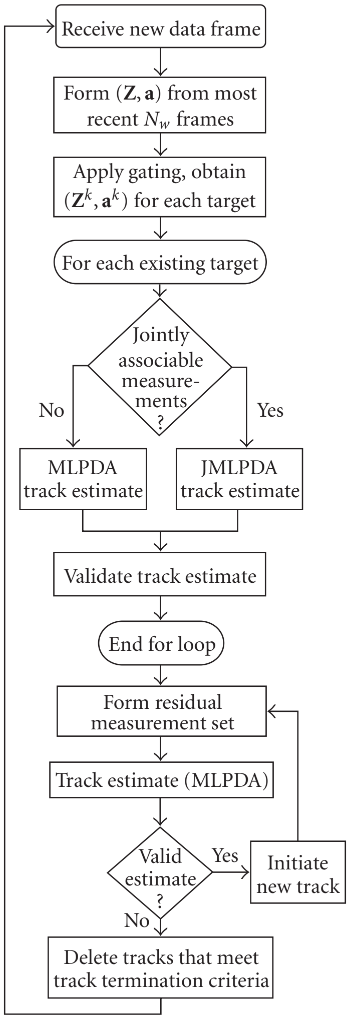
Receive new data frame
Form (Z, a) from most recent N_w frames
Apply gating, obtain (Z^k, a^k) for each target
For each existing target
Jointly associable measurements ?
No
Yes
MLPDA track estimate
JMLPDA track estimate
Validate track estimate
End for loop
Form residual measurement set
Track estimate (MLPDA)
Valid estimate ?
Yes
No
Initiate new track
Delete tracks that meet track termination criteria

FIGURE 3: Flowchart for one iteration of the MLPDA(MT) tracking system.

JMLPDA track estimates, the formation of the residual measurement set, and the track termination criteria.

6.1. Measurement gating

Measurement gating, or using a subset of the full measurement set to obtain a track estimate for a single target, is a well-known technique in multitarget tracking (see, e.g., [1, 4]). Since the JMLPDA algorithm is computationally more complex than the ML-PDA algorithm (particularly when the joint estimates of more than two targets is being performed), use of measurement gating becomes vital. The advantage lies in using JMLPDA only for those cases where targets share gated measurements. Further, by reducing the size of the measurement set computation time for the ML-PDA algorithm is reduced as well.

In this application, a measurement gate is set up based on the prior track estimate and its associated covariance using the Mahalanobis distance whereby measurements are included which satisfy the relation

$$(\mathbf{z}_{ij} - \hat{\mathbf{z}}_i)^T \mathbf{S}_i^{-1} (\mathbf{z}_{ij} - \hat{\mathbf{z}}_i) \le \gamma, \quad (26)$$

where $\hat{\mathbf{z}}_i$ is the predicted measurement at frame i of the current batch based on the last validated track estimate from the ML-PDA or JMLPDA algorithm. Track estimates are propagated forward in time as necessary according to the constant velocity model. The $\mathbf{S}_i$ term is the innovation covariance at frame i and is given by

$$\mathbf{S}_i = \mathbf{H}\mathbf{P}_i\mathbf{H}^T + \mathbf{R}, \quad (27)$$

where $\mathbf{P}_i$ is the covariance of the track estimate on the ith frame of the current batch based on the last validated track estimate and $\mathbf{H}$ is the measurement matrix where $\mathbf{z} = \mathbf{Hx}$, assuming a linear measurement model. The limiting threshold γ is set based on a desired probability of containing the target-originated measurement within the gate volume (P_G).

6.2. Track validation

Because ML-PDA and JMLPDA will always return a track estimate (the global maximum of the LLR or JLLR), one must test the track estimate for validity. The ML-PDA track validation procedures are described in Section 4. JMLPDA track validation procedures are described in Section 5.2.

6.3. Residual measurement set

Once track estimates are obtained for all targets currently in track, a search for new targets must take place. JMLPDA is unsuitable for this task since it requires knowledge of the number of new targets. Therefore, ML-PDA is used.

In order to eliminate the effects of known targets on the ML-PDA LLR, one must account for those measurements that can be associated with known targets. To do this, one associates at most one measurement in each data frame to each target using the same PDA approach described in Section 5.2 for track validation. The residual data set is then the original data set with those measurements associated to known targets edited out.

Using the residual data set the ML-PDA algorithm is applied and the resulting track estimate validated. If a new target is validated, its associated measurements are also edited out to form a new residual data set and the process repeated until the ML-PDA algorithm returns a track estimate that fails the validation test. This technique assumes that new targets are well separated in which target-originated measurements from one new target do not affect the LLR at the track estimate for any other new target.

6.4. Track termination

If taken in isolation, the failure of ML-PDA to validate a track estimate is sufficient to declare there is no track present. However, this does not account for any prior knowledge that a track had previously existed based on measurement data outside of the window of the current ML-PDA batch. When tracking VLO contacts, in order to limit the false track acceptance rate, the true target acceptance rate (based on the track validation threshold value) can be relatively low (in the 50–70% region) even when target detections are present in the batch. Therefore, in the MLPDA (MT), we have elected to incorporate an additional higher level track termination test beyond the ML-PDA track validation.

Existing targets are tested for termination using an *M/N* rule. A track is terminated if fewer than *M* validated track estimates were obtained from the most recent *N* applications of MLPDA (MT) on that target. Based on the operating characteristic of the ML-PDA algorithm, one can obtain the probability of detecting a true track (P_{DT}) using ML-PDA. Then for given values of *M* and *N*, one can obtain the track termination statistics (probability of correctly terminating a track that is lost and the probability of incorrectly terminating a track that is still held). In making this calculation, one must also account for the fact that P_{DT} relates to independent track estimates. When using a sliding window ML-PDA implementation in which a new track estimate is obtained from the most recent N_w frames of data, track estimates separated by fewer than N_w frames are correlated in that they use common data frame(s).

6.5. ML-PDA (MT) performance

A 2-target crossing scenario was developed to test the performance of the MLPDA (MT) tracking system. The surveillance region consists of a 12 km-by-12 km square region (the origin is located at the southwest corner) in which two targets are placed. Target 1 is initially located near the southwest corner of the region moving northeast and target 2 is initially located near the northwest corner of the region moving southeast. Target motion is constant velocity in an x-y plane. Measurements consist of x-y position. The parameters for this scenario are listed in Table 3. Figure 4 shows the target trajectories and a representative single frame of clutter over the surveillance region.

The simulations are intended to test the ability of the MLPDA (MT) algorithm to maintain track when multiple targets are present in the surveillance region. The ability of the MLPDA (MT) to initiate new tracks and delete lost tracks will be explored in future work. Monte Carlo simulations were performed over a range of P_d and P_{fa} values, with 100 simulations conducted at each operating point. To establish a performance comparison, each simulation was run using the MLPDA (MT) algorithm and the probabilistic multiple hypothesis tracker (PMHT) algorithm [24–26].

PMHT is a capable multiple target tracking algorithm that uses data in a batch of N_w frames and computes the joint track estimates at each frame of data using the expectation-maximization (EM) algorithm [11] which returns the maximum a posteriori (MAP) estimate. The EM algorithm is a general estimation technique for incomplete or missing data problems and is guaranteed to converge to at least a local maximum of the objective function. For PMHT the missing data are the specific measurement-to-target associations. The specific version of PMHT used in the simulations is the homothetic PMHT [29].

Table 3: Scenario parameters.

Parameter	Value
Initial $\mathbf{x}_r^1$ (in m and m/s)	$[2000\ \ 2000\ \ 5\ \ 5]^T$
Initial $\mathbf{x}_r^2$ (in m and m/s)	$[2000\ \ 10000\ \ -5\ \ 5]^T$
Target P_d	0.7/0.9
Target SNR	5–15 dB
False Alarm Density (λ)	4–20 × 10^{-7}
False Alarm Rate (P_{fa})	0.05–0.25
Avg. Number of False Alarms (Surveillance Region)	57–285
Avg. Number of False Alarms (Validation Gate)	0.33–1.64
Batch Size (N_w)	7 frames
$\mathbf{R}$	$\mathrm{diag}[101^2\ \ 101^2]\ \mathrm{m}^2$
Sample Period (T)	20 sec
Scenario Duration	80 frames

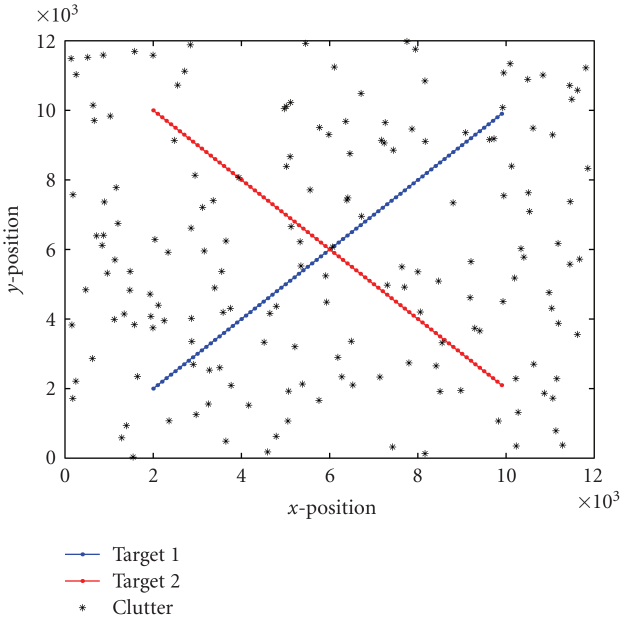

Figure 4: Simulation scenario showing target trajectories and a representative frame of noise measurements, $P_{fa} = .15$.

In our implementation, the PMHT uses a continuous white noise acceleration (CWNA) target motion model [2] with the process noise spectral density set at values of 0.0125 and 0.05 $\mathrm{m}^2/\mathrm{s}^3$. The process noise spectral density was set so that the tracker could accommodate small target velocity changes on the order of 0.5 and 1.0 m/s over a sample interval.

As we are comparing two distinctly different trackers, it is worthwhile to highlight some of the key distinctions or advantages one tracker inherently has over the other and which make direct comparisons more difficult.

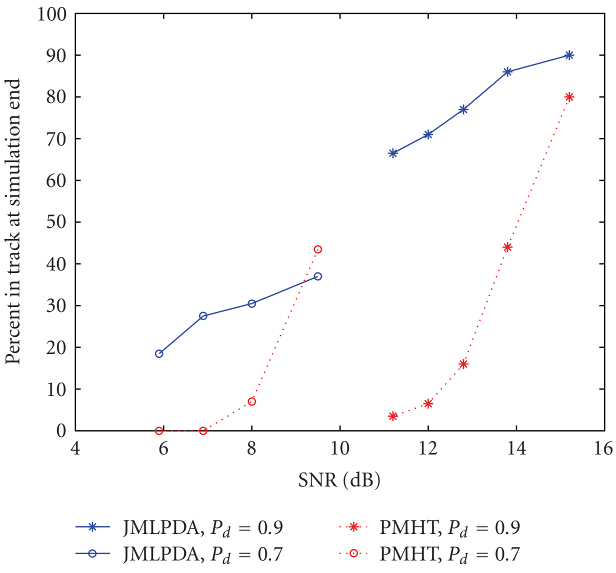

FIGURE 5: Percent of simulations, each tracker remained in track to the end of the run.

(1) The simulations do not test the track initiation ability of each tracker. This is a key advantage to ML-PDA in that it simultaneously performs track initiation and track update functions. PMHT requires a separate track initiation module. In a heavy clutter environment such as the ones examined in this paper, track initiation becomes problematic for MHT-based track initiation modules (including PMHT) due to the explosion of hypotheses resulting in the testing of tentative (unconfirmed) tracks which are mostly clutter.

(2) Since PMHT computes the MAP estimate (a Bayesian tracker), it incorporates prior information on the target state into the current track estimate. Being a maximum-likelihood tracker, ML-PDA and JMLPDA do not take advantage of any prior information in computing the current track estimate. Consequently, for the same batch length (N_w) for each tracker, one would expect PMHT to provide more accurate track estimates than ML-PDA/JMLPDA.

(3) PMHT gets the best state estimate, $\hat{\mathbf{x}}$, at frame N_w subject to its assumptions. Due to ML-PDA/JMLPDA assumptions, it fits the best constant velocity solution to the entire batch. Consequently, ML-PDA/JMLPDA position errors for the target are smallest in the middle of the batch and largest at frame N_w.

(4) Because it incorporates process noise into its target motion model, PMHT should perform better in situations where the target is maneuvering. ML-PDA has a reduced capability to track maneuvering targets because its target motion model is deterministic. Since the true target motion is constant-velocity, one would expect (examining this factor in isolation) ML-PDA to have reduced rms tracking errors and possibly a longer mean track life.

Examining these factors in advance of the simulations, it is unclear which tracker is expected to perform better—ML-PDA or PMHT. Factors (2) and (3) above give PMHT an advantage. Factor (4) gives ML-PDA an advantage. Factor (1) was not exercised in the simulations.

These simulations analyze the ability of the MLPDA(MT) and PMHT tracking systems to maintain track on the targets as they cross. In addition to monitoring tracking errors over time, statistics on the track life (time until the track diverges from the target) are evaluated for each tracker. Each tracker was provided with randomized initial target states with accuracy similar to that of the steady-state tracker errors. The initial covariance provided to the PMHT was artificially large (making the initial track estimate uninformative). However, once PMHT processed its first set of data, it used its own track estimate covariance for subsequent batches thereby including the effects of an informative prior for PMHT.

MLPDA (MT) does not use prior information except to establish a measurement validation gate. However, MLPDA (MT) does have the ability to accept or reject track estimates based on track validation criteria. In the case where the MLPDA (MT) track estimate is not validated, the current track estimate and covariance reported by the algorithm is the previous track estimate and covariance propagated forward in time by applying the motion model equations.

The simulations yield the performance of the PMHT and MLPDA (MT) algorithms in terms of their ability to keep track of the targets without track divergence. A track estimate was considered to have diverged from the target if the combined rms position errors exceeded 800 m, equating to a position error in each dimension exceeding 4 standard deviations. Additional statistics are provided to illustrate rms errors over the course of the scenario.

Figure 5 shows the percent of simulations where a target remained in track for the full 80 frames of the scenario for a variety of P_d and SNR values for each tracker. Since the in-track percentage for each target and a given tracker was approximately the same, the results for the two targets are combined such that the in-track percentage is determined over 200 opportunities (100 simulations and 2 targets).

Figures 6–8 present results for the $P_d = 0.9$ and $P_{fa} = 0.1$ set of simulations plotted as a function of time so that performance of each tracker can be observed. Figure 6 shows the percent of simulations each tracker is in track of a target over time. Figures 7 and 8 show the evolution of rms position and velocity errors of each tracker over time. As before, the results for the two targets are combined to yield 200 samples (100 simulations and 2 targets) for each tracker.

In these simulations, MLPDA (MT) consistently outperforms PMHT in terms of in-track percentage. This can be attributed to the ability of MLPDA (MT) to reject “bad” track estimated through its validation criteria. An increased track loss rate is observed for MLPDA (MT) when the targets are interacting and JMLPDA is used to jointly estimate the target tracks.

It can also be observed that the performance of MLPDA (MT) is relatively independent of P_d in that performance decreases approximately linearly with SNR along the same line for the two P_d values shown in Figure 5. In contrast,

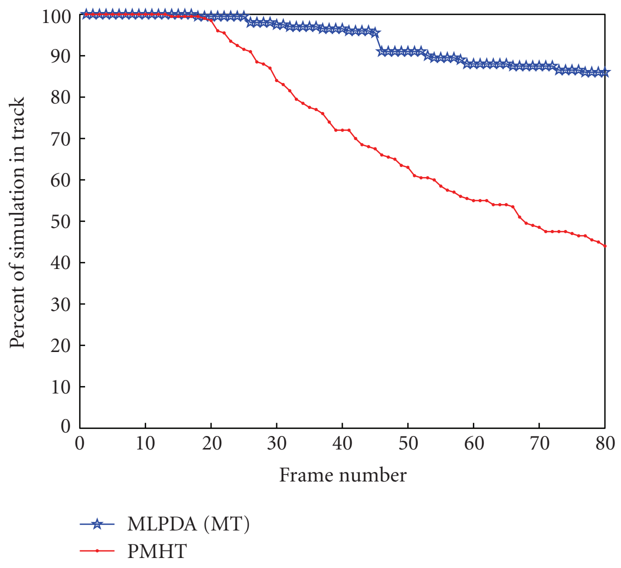

Figure 6: Percent of simulations, each tracker is in track of each target.

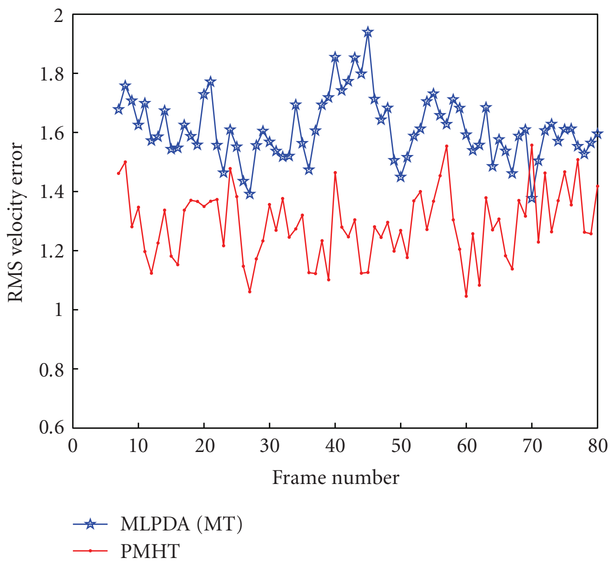

Figure 8: Evolution of rms velocity errors over time for each tracker and target.

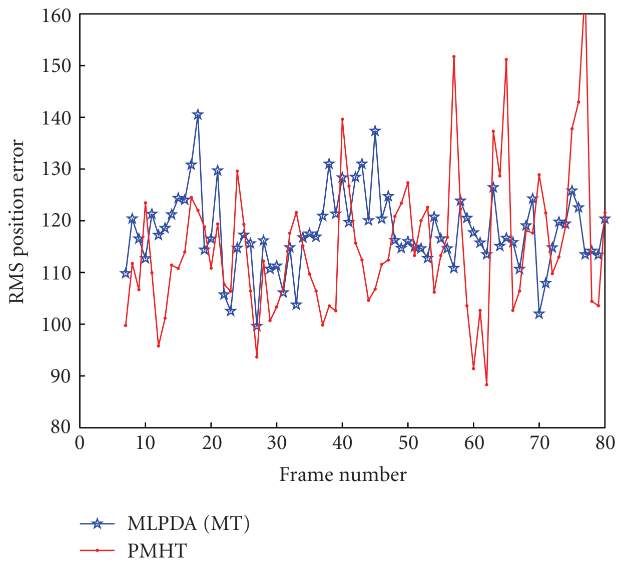

Figure 7: Evolution of rms position errors over time for each tracker and target.

PMHT performance improves significantly when the detector threshold is raised, simultaneously reducing the number of clutter points and reducing target P_d. Further analysis is required to fully quantify this effect.

When considering tracking over the full scenario, changes in the PMHT process noise power spectral density have a significant effect on PMHT's ability to remain in track. Simulations were conducted where the PMHT modeled process noise was reduced from 0.05 to 0.0125 m^2/s^3. With the reduced process noise, PMHT performance improved to the point that for each P_d considered and at low clutter levels PMHT in-track performance was superior to that of MLPDA (MT). As clutter increased, PMHT performance degraded faster than that of MLPDA (MT) such that there was a clutter lever beyond which MLPDA (MT) outperformed PMHT.

As a final comment, it is reemphasized that the strength of ML-PDA and MLPDA (MT) lies in its ability to successfully track VLO targets. A consequence of this performance is a higher computational complexity than most other tracking algorithms. When extended to the multitarget arena, MLPDA (MT) will be limited in its ability to track large numbers of interacting targets. However, its computational complexity in tracking noninterfering targets does remain linear in target number.

Another situation where MLPDA(MT) may fail is in a relatively common passive sonar problem wherein a high SNR target lies in the same (or adjacent) resolution cell as the VLO target of interest. In this case, since ML-PDA assumes at most one measurement corresponds to the target, ML-PDA will preferentially assign the measurement to the stronger target, which has a higher P_d. In order to successfully track the weaker target, one would need to increase the size of the ML-PDA batch to include frames where the interaction is less. Analysis of this type of situation, including guidelines on the adaptive sizing of the ML-PDA batch is a subject of future research.

7. CONCLUSIONS

While ML-PDA has been demonstrated to be an effective tracking algorithm when tracking VLO targets, no multitarget version of ML-PDA has been reported in the literature. In this paper, we first described several recent advances in the ML-PDA target tracking algorithm which make this tracker feasible for implementation in real-time tracking systems.

Next, we extended the ML-PDA framework to jointly track multiple targets using JMLPDA. Incorporating both

ML-PDA and JMLPDA into a multitarget tracking system yields the ML-PDA (MT) target tracking system. Comparisons were made with the PMHT using a 2-target crossing scenario which showed that as clutter density increased, ML-PDA performed better at maintaining track through the problem than did PMHT.

More work remains in extending the comparison of MLPDA (MT) to PMHT (or other multitarget algorithms) to situations commonly found in passive sonar tracking as well as to maneuvering targets.

ACKNOWLEDGMENT

This research was supported by the Office of Naval Research.

REFERENCES

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

[2] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, New York, NY, USA, 2001.

[3] S. S. Blackman, *Multiple-Target Tracking with Radar Applications*, Artech House, Boston, Mass, USA, 1986.

[4] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, Boston, Mass, USA, 1999.

[5] W. R. Blanding, P. K. Willett, Y. Bar-Shalom, and R. S. Lynch, "Covert sonar tracking," in *Proceedings of IEEE Aerospace Conference*, pp. 2053–2062, Big Sky, Mont, USA, March 2005.

[6] W. R. Blanding, P. K. Willett, Y. Bar-Shalom, and R. S. Lynch, "Directed subspace search ML-PDA with application to active sonar tracking," to appear in *IEEE Transactions on Aerospace and Electronic Systems*.

[7] W. R. Blanding, P. K. Willett, and Y. Bar-Shalom, "Offline and real-time methods for ML-PDA track validation," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, part 2, pp. 1994–2006, 2007.

[8] S. Buzzi, M. Lops, and L. Venturino, "Track-before-detect procedures for early detection of moving target from airborne radars," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 3, pp. 937–954, 2005.

[9] M. R. Chummun, Y. Bar-Shalom, and T. Kirubarajan, "Adaptive early-detection ML-PDA estimator for LO targets with EO sensors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 2, pp. 694–707, 2002.

[10] H. A. David, *Order Statistics*, John Wiley & Sons, New York, NY, USA, 1970.

[11] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society B*, vol. 39, no. 1, pp. 1–38, 1977.

[12] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Boston, Mass, USA, 1989.

[13] E. J. Gumbel, *Statistics of Extremes*, Columbia University Press, New York, NY, USA, 1958.

[14] C. Jauffret and Y. Bar-Shalom, "Track formation with bearing and frequency measurements in clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 6, pp. 999–1010, 1990.

[15] L. A. Johnston and V. Krishnamurthy, "Performance analysis of a dynamic programming track before detect algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 1, pp. 228–242, 2002.

[16] T. Kirubarajan and Y. Bar-Shalom, "Low observable target motion analysis using amplitude information," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1367–1384, 1996.

[17] E. L. Lehmann and J. P. Romano, *Testing Statistical Hypotheses*, Springer, New York, NY, USA, 3rd edition, 2005.

[18] D. Lerro and Y. Bar-Shalom, "Interacting multiple model tracking with target amplitude feature," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 2, pp. 494–509, 1993.

[19] S. C. Nardone and V. J. Aidala, " Observability criteria for bearings-only target motion analysis," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 17, no. 2, pp. 162–166, 1981.

[20] A. Papoulis and S. Pillai, *Probability, Random Variables and Stochastic Processes*, McGraw-Hill, New York, NY, USA, 2002.

[21] W. Press, S. Teukolosky, W. Vetterling, and B. Flannery, *Numerical Recipes in C*, Cambridge University Press, New York, NY, USA, 1992.

[22] M. G. Rutten, B. Ristic, and N. J. Gordon, "A comparison of particle filters for recursive track-before-detect," in *Proceedings of the 8th International Conference on Information Fusion*, vol. 1, pp. 169–175, Philadelphia, Pa, USA, July 2005.

[23] L. D. Stone, C. A. Barlow, and T. L. Corwin, *Bayesian Multiple Target Tracking*, Artech House, Boston, Mass, USA, 1999.

[24] R. L. Streit and T. E. Luginbuhl, "A probabilistic multihypothesis tracking algorithm without enumeration and pruning," in *Proceedings of the 6th Joint Data Fusion Symposium*, Laurel, Md, USA, 1993.

[25] R. J. Streit and T. E. Luginbuhl, "Maximum likelihood method for probabilistic multihypothesis tracking," in *Proceedings of Signal and Data Processing of Small Targets*, vol. 2235 of *Proceedings of SPIE*, pp. 394–405, Orlando, Fla, USA, April 1994.

[26] R. L. Streit and T. E. Luginbuhl, "Probabilistic multihypothesis tracking," Tech. Rep. NUWC-NPT Technical Report 10,428, Naval Undersea Warfare Center, Newport, RI, USA, February 1995.

[27] S. M. Tonbsen and Y. Bar-Shalom, "Maximum likelihood track-before-detect with fluctuating target amplitude," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 3, pp. 796–809, 1998.

[28] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part III*, John Wiley & Sons, New York, NY, USA, 2001.

[29] P. Willett, Y. Ruan, and R. Streit, "PMHT: problems and some solutions," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp. 738–754, 2002.

[30] P. Willett, S. Coraluppi, and W. Blanding, "Comparison of soft and hard assignment ML trackers on multistatic data," in *Proceedings of IEEE Aerospace Conference*, p. 12, Big Sky, Mont, USA, March 2006.