Proceedings of the 28th Conference
on Decision and Control
Tampa, Florida • December 1989

AUTOMATIC TRACK FORMATION IN CLUTTER WITH A RECURSIVE ALGORITHM

Y. Bar-Shalom
Univ. of Conn. U-157
Storrs, CT 06269-3157

K.C. Chang
Advanced Decision Systems
Mt. View, CA

H.A.P. Blom
Univ. of Conn.
(on leave from NLR,
The Netherlands)

Abstract

In this paper we present a recursive algorithm for forming tracks in a cluttered environment. The approach consists of the Interacting Multiple Model (IMM) algorithm combined with the Probabilistic Data Association Filter (PDAF), resulting in the IMMPDAF. The track formation is accomplished by considering two models: one being the "true target", with a certain probability of detection P_D, the other one being an "unobservable target" (or "no target"): same model as the former except that P_D=0. The latter represents either a true target outside the sensor coverage or an erroneously hypothesized target. Assuming the clutter measurements as uniformly distributed, the algorithm yields the "True Target Probability" (TTP) of a track, i.e., it can be called "intelligent" since it has a quantitative assessment of whether it has a target in track. This algorithm is useful for low SNR situations where the detection threshold has to be set low in order to detect the target and this leads to a high rate of false alarms.

1. INTRODUCTION

The problem of track formation in clutter consists of the association of several detections over time and a decision that accepts these detections as having originated from the same target. This problem, also known as track initiation, has received a great deal of attention. The most commonly used approach is based on an M/N detection logic, which can be evaluated using a Markov chain [13,21,15]. These evaluations assumed fixed size gates and, thus, yielded only approximate results for false tracks (due to clutter), where the gates vary and their sizes are critical. A first technique that took into account the varying gate sizes, as well as the effect of clutter on the logic-based formation of true tracks, was presented in [1,2].

For tracking in a cluttered environment there are several algorithms ranging from the Probabilistic Data Association Filter (PDAF) [3,16,14,7], which is only slightly more complex than a standard filter, to the much more complex Multiple Hypotheses Technique (MHT) [26,15,6,3]. These algorithms can be combined with approximate Bayesian techniques to account for non-Gaussian target motion motion. The combination of the PDAF with a (non-switching) multiple model estimation technique to track a maneuvering target was presented in [17]. For switching models (dynamic multiple models) the PDAF was combined with the Interacting Multiple Model (IMM) algorithm [9,10] in [8,19], with the latter considering multiple sensors. Similar combination is possible for the MHT technique, which has the capability of initiating tracks, but it is very complicated and expensive to implement [25,25]. In [23,24] the MHT approach was used for initiation with pruning to the best N hypotheses via an "efficient search". In spite of this, it is believed that the difficulties of implementation of the MHT in a dense cluttered environment (computation, memory, data base management, the need for environmental parameters [3] and multitude of outputs) warrant the interest in recursive algorithms with *fixed requirements* and a minimum of modeling parameters, such as the PDAF.

The original version of the PDAF did not have capability of track initiation or deletion. Recently, [14] augmented the PDAF to include initiation and deletion by adding in the association an event corresponding to "unobservable target", which can represent either a true target outside the sensor coverage or an erroneously hypothesized target, i.e., it is equivalent to *no target*. This work motivated the present investigation of track formation within the general context of hybrid state (dynamic multiple model) estimation [27]. To this purpose, a dynamic multiple model is first needed for the track formation problem, and then incorporated within a suitable recursive Bayesian estimation algorithm.

The procedure to put the track formation problem into the context of multiple models is as follows: in both models the measurements can originate from the target, whose detection probability is P_D, or clutter; however, in the "no target" model one has P_D=0. The resulting algorithm, which yields model probabilities, provides the *True Target Probability* (TTP) for each track under consideration. Thus, *this algorithm can assess* its own performance, i.e., it can be called "intelligent". A technique to obtain a TTP was developed in [18] based on the observed sequence of detections, but without accounting for the locations of the corresponding measurements; this technique requires the spatial density of the false detections.

The recursive track formation algorithm for a cluttered environment presented in this paper consists of the combination of the IMM (with two models: "true target" and "no target") with the PDAF to associate the measurements to the tracks that are formed. The PDAF calculates the probabilities of each measurement falling in the validation region that it originated from the target of interest. The non-parametric version of the PDAF [3], which is used here, assumes a known target detection probability while it does not need the spatial density of the false measurements. This latter feature makes it suitable for operation in an environment where the false detection rate might change drastically within the surveillance region.

In the context of hybrid systems, the algorithm of [14] is of the Generalized Pseudo-Bayesian of order 1 (GPB1) type [27]. The GPBn algorithm is a suboptimal multiple model technique with memory depth of n-1 scans: it uses, for r models, r^n filters whose ouputs are combined, at the end of each cycle, into r^{n-1} estimates (we designate the GPB algorithm's order according to the number of filters it uses rather than its memory depth). As shown in [10], the IMM, which is between GPB1 and GPB2 [11],

Research supported by ONR Contract N00014-87-K-0057.

has the higher performance of the latter for (almost) the lower cost of the former.

Section 2 presents the problem formulation. Section 3 summarizes the PDAF and Section 4 the IMMPDAF algorithm. Section 5 shows how the Automatic Track Formation algorithm operates starting from the detections from the first two scans followed by an IMMPDAF. The evaluation of the performance of the proposed algorithm is discussed in Section 6. Simulation results with clutter of various densities, including very heavy clutter, are presented in Section 7.

2. PROBLEM FORMULATION

Two models are used: one for observable ("true") target, designated as Model t=2, one for unobservable target (or false target), designated as Model t=1. In Model t a target-originated measurement is detected with probability P_D^t: for Model t=2 (observable target), $P_D^2=P_D$, the target detection probability, while for Model t=1 (unobservable target), $P_D^1=0$. The latter can also be considered as a target "too weak" to be tracked.

The target motion is modelled in Cartesian coordinates as

$$x^t(k+1) = F^t\, x^t(k) + v^t(k) \qquad k=0,1,\ldots \qquad t=1,2 \tag{2.1}$$

where $x^t(k)$ is the state of the target at time k for Model t. The state vector is taken the same for both models, as a 4-dimensional vector consisting of position and velocity (in each of the 2 Cartesian coordinates)

$$x^t = [x\ \dot{x}\ y\ \dot{y}]' \qquad t=1,2 \tag{2.2}$$

F^t is the transition matrix of Model t for the sampling period T,

$$F^t = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad t=1,2 \tag{2.3}$$

$v^t(k)$ is the zero-mean white Gaussian process noise with known variance

$$E[v^t(k)\ v^t(j)'] = Q^t\ \delta(k,j) \tag{2.4}$$

where

$$Q^t = \begin{bmatrix} u^t & 0 \\ 0 & u^t \end{bmatrix} \tag{2.5}$$

and

$$u^t = \begin{bmatrix} T^4/4 & T^3/2 \\ T^3/2 & T^2 \end{bmatrix} q_t \qquad t=1,2 \tag{2.6}$$

In the above q_t is the variance of the process noise modeling the motion uncertainty (acceleration) in Model t.

The state vectors for the two models can be different [8,19]. In the sequel the superscript t on the state vector indicating the model will be dropped for simplicity wherever this does not cause ambiguity.

The target-originated measurements, which occur with probability P_D, are modelled as

$$z(k) = H\ x(k) + w(k) \qquad k=0,1,\ldots \tag{2.7}$$

where

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.8}$$

w(k) are the measurement noises, white, zero-mean and with known variances

$$E[w(k)\ w(j)'] = R\ \delta(k,j) \tag{2.9}$$

where

$$R = \begin{bmatrix} R_{11} & 0 \\ 0 & R_{22} \end{bmatrix} \tag{2.10}$$

The locations of the false measurements will be modeled as uniformly distributed. The number of the false measurements will be assumed to have a "diffuse prior" (any number of false measurements is equiprobable) [3] - this will allow us to use a state estimation algorithm that does not require the spatial density of the false measurements (clutter), described in the next section.

A Markov chain will model the observable/unobservable situation as follows. Denoting by M(k) the model in effect during period k, the following transition (model switching) probabilities will be assumed:

$$P\{M(k+1)=\text{unobs.}|M(k)=\text{unobs.}\} = p_{11} = 1 - \epsilon_1 \tag{2.11}$$

$$P\{M(k+1)=\text{observable}|M(k)=\text{unobservable}\} = p_{12} = \epsilon_1 \tag{2.12}$$

$$P\{M(k+1)=\text{unobservable}|M(k)=\text{observable}\} = p_{21} = \epsilon_2 \tag{2.13}$$

$$P\{M(k+1)=\text{observable}|M(k)=\text{observable}\} = p_{22} = 1 - \epsilon_2 \tag{2.14}$$

i.e., with transitions between the models assumed with some low probability. Practical values for these "design parameters" will be discussed in the simulation section.

3. THE PROBABILISTIC DATA ASSOCIATION FILTER

At each scan, a validation gate, centered around the predicted measurement of the target, is set up to select the measurements to be associated probabilistically to the target. The validation region is

$$[z(k)-\hat{z}(k|k-1)]'\ [S(k)]^{-1}\ [z(k)-\hat{z}(k|k-1)] < g^2 \tag{3.1}$$

where S(k) is the covariance of the innovation correspondong to the true measurement. Denote the set of validated measurements at time k as

$$Z(k) = \{z_j(k)\}_{j=1}^{m(k)} \tag{3.2}$$

and the cumulative set of measurements as

$$Z^k = \{Z(\kappa)\}_{\kappa=1}^{k} \tag{3.3}$$

The single-model minimum variance estimate of the state is given in the PDAF approach [3] by

$$\hat{x}(k|k) = E\{x(k)|Z^k\} = \sum_{j=0}^{m(k)} E\{x(k)|\theta_j(k),Z^k\}\ P\{\theta_j(k)|Z^k\} \tag{3.4}$$

where m(k) is the number of validated measurements at time k, $\theta_j(k)$ is the event that $z_j(k)$ is the correct measurement from the target, $\theta_0(k)$ is the event that none of the validated measurements is correct and

$$P\{\theta_j(k)|Z^k\} \triangleq \beta_j(k) \tag{3.5}$$

is the probability that the measurement $z_j(k)$ is from the target.

Using the Non-parametric PDAF, the association probabilities (3.5) are [3]

$$\beta_j(k) = e_j(k) \left[b(k) + \sum_{j=1}^{m(k)} e_j(k) \right]^{-1} \qquad j=1,\ldots,m(k) \tag{3.6}$$

$$\beta_0(k) = b(k) \left[b(k) + \sum_{j=1}^{m(k)} e_j(k) \right]^{-1} \tag{3.7}$$

where

$$e_j(k) \triangleq (P_G)^{-1} \mathcal{N}[\nu_j; 0, S(k)] \tag{3.8}$$

$$b(k) \triangleq m(k) (1 - P_D P_G) [P_D P_G V(k)]^{-1} \tag{3.9}$$

P_D is the probability of detection; $\mathcal{N}[a;0,S(k)]$ is the normal pdf with argument a, mean zero and variance S(k); P_G is the probability that the target measurement falls in the 2-dimensional validation region ("g-sigma gate") whose volume is

$$V(k) = g^2 \pi |S(k)|^{1/2} \tag{3.10}$$

and

$$\nu_j(k) = z_j(k) - \hat{z}(k|k-1) \tag{3.11}$$

is the innovation for measurement j.

The estimate conditioned on measurement j being correct is

$$\hat{x}_j(k|k) = \hat{x}(k|k-1) + W(k)\ \nu_j(k) \tag{3.12}$$

with the filter gain given by

$$W(k) = P(k|k-1)\ H(k)'\ [H(k)P(k|k-1)H(k)'+R]^{-1}$$
$$\triangleq P(k|k-1)\ H(k)'\ S(k)^{-1} \tag{3.13}$$

where P(k|k-1) is the covariance of the predicted state $\hat{x}(k|k-1)$ and H(k) is given in (2.8).

The single-model estimate can be rewritten as:

$$\hat{x}(k|k) = \sum_{j=0}^{m(k)} \beta_j(k)\ \hat{x}_j(k|k) = \hat{x}(k|k-1) + W(k)\ \nu(k) \tag{3.14}$$

where the combined innovation is

$$\nu(k) = \sum_{j=1}^{m(k)} \beta_j(k)\ \nu_j(k) \tag{3.15}$$

The covariance associated with (3.12), i.e., conditioned on knowing the correct measurement, is

$$P_j(k|k) = P(k|k-1) - W(k)\ S(k)\ W(k)' \tag{3.16}$$

The covariance associated with (3.14) is

$$P(k|k) = \beta_0(k)\ P(k|k-1) + \sum_{j=1}^{m(k)} \beta_j(k)\ P_j(k|k)$$
$$+ \sum_{j=0}^{m(k)} \beta_j(k)\ \hat{x}_j(k|k)\ \hat{x}_j(k|k)' - \hat{x}(k|k)\ \hat{x}(k|k)'$$
$$= \beta_0(k)\ P(k|k-1) + [1 - \beta_0(k)]\ P_j(k|k)$$
$$+ W(k) \left[\sum_{j=1}^{m(k)} \beta_j(k)\ \nu_j(k)\ \nu_j(k)' - \nu(k)\ \nu(k)' \right] W(k)' \tag{3.17}$$

The state and measurement prediction are done according to the standard Kalman filter equations.

4. THE INTERACTING MULTIPLE MODEL PDAF

The IMM filtering algorithm [8,9,10,4] assumes that the system obeys one of a finite number of models (with known parameters) and that model switches occur according to a Markov chain with known transition probabilities. The algorithm consists of the following:

- model-matched filters are run in parallel for each model, yielding the state estimate conditioned on each model being the current one
- the current probability of each model is evaluated in a Bayesian framework using the likelihood function of each filter [3]
- the input to each filter at the beginning of the cycle is a mixing (interaction) of their outputs from the previous cycle with suitable weightings that reflect the current probability of each model and the model transition probabilities
- the combined state estimate and covariance is computed (for output only) using the current model probabilities

We shall denote by $M_t(k)$ the event that Model t is in effect during the k-th sampling period. Similarly, $M_s(k-1)$ denotes the event that Model s is in effect during period k-1.

The basic equation of the IMM (for r models), obtained by a suitable use of the total probability theorem, is

$$p[x(k)|Z^k] \approx \sum_{t=1}^{r}\Big\{\sum_{s=1}^{r} p[x(k)|z(k),M_t(k),M_s(k-1),Z^{k-1}]\cdot$$
$$\mu_{s|t}(k-1|k-1)\Big\}\mu_t(k) \tag{4.1}$$

where the probability that Model t is correct at time k is

$$P(M_t(k)|Z^k) \triangleq \mu_t(k) \tag{4.2}$$

and

$$\mu_{s|t}(k-1|k-1) \triangleq P(M_s(k-1)|M_t(k),Z^{k-1}) \tag{4.3}$$

The inner summation in (4.1) represents one filter since the weightings $\mu_{s|t}(k-1|k-1)$ are conditioned on Z^{k-1}, i.e., are available prior to k. This represents the "interaction" or "mixing" of the estimates from time k-1 at the input of filter t.

In the above it is assumed that there is only one measurement, z(k), given by the sensor. The extension to the situation with clutter is obtained as follows:

- The standard filters in the IMM configuration are replaced by PDAFs
- The calculation of the model probabilities conditioned on the measurements is done using the likelihood function of the PDAF.

The block diagram of the IMMPDAF is presented in Fig. 1. The resulting algorithm is recursive, since it consists of a set of recursive PDA filters, which interact during each cycle. Furthermore, unlike the MHT algorithm, this has fixed memory and computational requirements.

The details of the algorithm for two models are presented below. One cycle of the algorithm consists of the following:

Step 1 (The IMM step [9]). Starting with $\hat{x}^s(k-1|k-1)$, one computes the mixed initial conditon for the filter matched to Model t according to the inner summation from (4.1)

$$\hat{x}^{0t}(k-1|k-1) = \sum_{s=1}^{2} \hat{x}^s(k-1|k-1)\ \mu_{s|t}(k-1|k-1) \qquad t=1,2 \tag{4.4}$$

where

$$\mu_{s|t}(k-1|k-1) \triangleq P(M_s(k-1)|M_t(k),Z^{k-1})$$
$$= \frac{1}{\bar{c}_t} P(M_t(k)|M_s(k-1),Z^{k-1})\ P(M_s(k-1)|Z^{k-1})$$
$$= \frac{1}{\bar{c}_t} p_{st}\ \mu_s(k-1) \tag{4.5}$$

$$\bar{c}_t \triangleq \sum_{s=1}^{2} p_{st}\ \mu_s(k-1) \tag{4.6}$$

and p_{st} is the assumed Markov model-switching probability giving the jump probability from Model s at time k-1 to Model t at time k. These model transition probabilities are assumed known - they are part of the design process, similar to the choice of the model parameters.

The covariance corresponding to (4.4) is

$$P^{0t}(k-1|k-1) = \sum_{s=1}^{2} \mu_{s|t}(k-1|k-1) \Big\{P^{s}(k-1|k-1) + [\hat{x}^{s}(k-1|k-1)-\hat{x}^{0t}(k-1|k-1)] \cdot [\hat{x}^{s}(k-1|k-1)-\hat{x}^{0t}(k-1|k-1)]'\Big\} \tag{4.7}$$

The estimate (4.4) and covariance (4.7) are used as input to the filter matched to Model t to yield $\hat{x}^{t}(k|k)$ and $P^{t}(k|k)$.

Step 2 (The PDA step [3]). In the presence of clutter (for a nonparametric PDAF), the likelihood function is the joint probability density function of the innovations, written as follows [3]:

$$\Lambda_t(k)=p[Z(k)|Z^{k-1}] = p[\nu_1^t(k),...,\nu_m^t(k)|m(k),Z^{k-1}]$$
$$= V(k)^{-m}\ \gamma_0[m(k)] + V(k)^{-m+1} \sum_{j=1}^{m(k)} P_G^{-1}\ \mathcal{N}[\nu_j^t(k);0,S^t(k)]\ \gamma_j[m(k)] \tag{4.8}$$

where $\nu_j^t(k)$ is the innovation at time k corresponding to measurement j and $S^t(k)$ is the covariance matrix of the predicted measurement, both computed in filter t; V(k) is the validation region volume and γ_j is the a priori probability for the corresponding measurement being correct, given by

$$\gamma_j[m(k)] = \begin{cases} \frac{1}{m(k)}\ P_D\ P_G & j=1,...,m(k) \\ 1-P_DP_G & j=0 \end{cases} \tag{4.9}$$

Using (3.7) and (3.8), one can rewrite (4.8) as follows:

$$p[\nu_1^t(k),...,\nu_m^t(k)|m(k),Z^{k-1}] = \left[\ b(k) + \sum_{j=1}^{m(k)} e_j(k)\ \right] \frac{P_D\ P_G}{m(k)}\ V(k)^{-m(k)+1} \tag{4.10}$$

Step 3 (The multiple model PDA step [17,3]). The model probabilities are updated as follows:

$$\mu_t(k) \triangleq P\{M_t(k)|Z^k\} = \frac{1}{c}\ p[Z(k)|M_t(k),\ Z^{k-1}]\ P\{M_t(k)|Z^{k-1}\}$$
$$= \frac{1}{c}\ \Lambda_t(k) \sum_{s=1}^{2} P\{M_t(k)|M_s(k-1),Z(k-1)\}\ P\{M_s(k-1)|Z(k-1)\}$$
$$= \frac{1}{c}\ \Lambda_t(k) \sum_{s=1}^{2} p_{st}\ \mu_s(k-1) = \frac{1}{c}\ \Lambda_t(k)\ \bar{c}_t \tag{4.11}$$

where $\bar{c}_t$ is the expression from (4.6) and $\Lambda_t(k)$ is given in (4.8).

Step 4 (For output only). The combination of the model-conditioned estimates and covariances is done according to the following equations:

$$\hat{x}(k|k) = \sum_{t=1}^{2} \hat{x}^t(k|k)\ \mu_t(k) \tag{4.12}$$

$$P(k|k) = \sum_{t=1}^{2} \mu_t(k)\ \{P^t(k|k) + [\hat{x}^t(k|k) - \hat{x}(k|k)]\ [\hat{x}^t(k|k) - \hat{x}(k|k)]'\} \tag{4.13}$$

5. THE AUTOMATIC TRACK FORMATION ALGORITHM

A *tentative track* is started for every detection in scan (frame) 1. The corresponding association region in scan 2 is determined as follows. It is assumed that the velocity of the target along coordinate i is within the interval $[-v_{i_{max}}, v_{i_{max}}]$, i=1,2. With the measurement noises independent of each other, with variances R_{ii}, i=1,2, the gate is taken as a rectangle with sides determined by the maximum velocity plus two standard deviations of the measurement noise on each side. The volume (actually, area) of this association region is thus

$$V(2) = [2(v_{1_{max}}T + 2\sqrt{R_{11}})]\ [2(v_{2_{max}}T + 2\sqrt{R_{22}})] \tag{5.1}$$

Each measurement in such a gate yields an initiating pair, from which a ***preliminary track*** is started. Starting from scan 3, a 2-model PDAF as described in the previous section, is run on each preliminary track. The state estimate initialization is done using the measurements from the first two scans [3]. One model is "observable target" ("true" or "live") with the assumed target $P_D>0$, while the other one is "unobservable target" ("false" or "dead") - same as the true one, but with $P_D=0$. A Markov chain transition matrix is assumed between these two models as in (2.11)-(2.14). Each model is assumed to have initial probability 0.5. The algorithm computes the true track probability (TTP) of each track and discards the tracks with TTP below a certain threshold, e.g., 0.05.

Since several tracks can be initiated in a cluttered environment, a test of similarity (if they can be assumed as having the same origin) is carried out to avoid redundant tracks according to the track-to-track association technique presented in [3, Sec. 10.2]. The resulting test statistic has a 4 degrees of freedom chi-square distribution. Its 99% point is about 13 - if the test statistic is smaller, then the two tracks under consideration can be accepted as having the same origin.

Following a certain number of scans, an acceptance decision (confirmation) can be made and the accepted tracks can be passed to a track maintenance algorithm (precision filtering). The latest state estimate of such a track is then the initial estimate for the precision filtering. The track maintenance can also be done via an IMMPDAF with the following models [5]:

- "no target" (unobservable)
- nearly constant velocity motion
- large acceleration increment
- nearly constant acceleration motion

This allows tracking of the target if it maneuvers, recognizing if the target has been lost or disappeared (track termination), as well as eventual elimination of false tracks that were not discarded during the track formation stage. In other words, one can say that this algorithm is a (reasonably) ***intelligent tracker***. Track maintenance with several motion models has been previously implemented in [8,19] (straight line and high-g turns).

6. SIMULATION RESULTS

The track formation algorithm presented above has been implemented to operate in a fixed window mode. Since a real-time implementation would be in a sliding window mode, the technique described in Appendix A was used to obtain the sliding window performance figures.

The maximum target velocity was assumed $v_{i_{max}}=40$, i=1,2; the measurement noise variances were $R_{11}=R_{22}=25$. The sampling period was T=1 and the process noise was taken as q_t=0.1 for each of the two coordinates, for both models (different values can be taken for the two models if desired). The model switching probabilities were taken as

$$p_{11}=p_{22}=0.98 \qquad p_{12}=p_{21}=0.02$$

The area of the association region at the second scan was, from (5.1),

$$V(2) = [2(40 + 2\sqrt{25})]^2 = 10^4$$

while for the third scan the 99% gate area is [2]

$$V(3) = 9\ \pi\ 5\sqrt{R_{11}R_{22}} \approx 3.5\cdot 10^3$$

The minimum value of the true track probability (TTP) of a track, in order not to be discarded, was 0.05 .

Table 1 presents the track formation performance of the IMMPDAF vs. the GPB1PDAF (the equivalent of [14]) for two levels of clutter. These evaluations are based on 300 Monte Carlo runs for the heavy clutter and 1000 runs for the lower level (medium) clutter. In the surveillance region, which had an area of 2.4E5, there was one target moving with a velocity vector of [30 20]. The target detection probability was taken as 0.9 and 0.8.

Figure 2 illustrates the heavy clutter density in the surveillance region by showing the measurements from the first scan (frame). Figure 3 shows the track of the target as formed by the IMMPDAF track formation algorithm in a sample run with the heavy clutter and target P_D=.9 (in this run there was no false track). The ellipses represent the the 2σ uncertainty for the estimated target position. The track formation algorithm knew only the value of P_D, but used no information about the spatial density of the false alarms.

The heavy clutter corresponds to a false alarm probability per resolution cell (whose size is about $10\sqrt{R_{11}R_{22}}$) of 0.025. Such a situation occurs if, due to a ***weak target***, the detection threshold has to be set low in order to detect the target with a reasonably high probability. This high false alarm rate can be handled by the present algorithm.

For the heavy clutter case, the density of false tracks after 7 scans with the IMMPDAF was significantly lower than with the GPB1PDAF. The expected confirmation time for the GPB1PDAF was slightly shorter than for the IMMPDAF. The expected confirmation times were evaluated according to the technique of Appendix A.

Overall, it seems that the IMMPDAF is preferable to the GPB1PDAF because of the significantly lower false track probability shown here. Moreover, the IMMPDAF can be also used to obtain excellent maneuvering target tracking capability as shown earlier in [8, 19].

8. SUMMARY AND CONCLUSIONS

A new algorithm for automatic track formation in clutter has been presented. This algorithm consists of the IMM combined with the PDAF and relies on the key concept of "target observability" from [14], which has been put in the general framework of dynamic multiple model (hybrid systems) estimation. Two models, differing only in the detection probability P_D, are considered in the process of the track formation: one for a target with high P_D, and one for an "unobservable target", defined by P_D=0 . This algorithm is useful for low SNR situations where the detection threshold has to be set low in order to detect the target and this leads to a high rate of false alarms.

This technique of track formation can be naturally followed by the maneuvering target tracking technique consisting of the same algorithm with additional models for the target's maneuvering modes [8,19,5]. This technique provides a sytematic way, via a recursive algorithm, to obtain the probability that there is a target in track as well as maneuvering target tracking capability in clutter.

APPENDIX A. EVALUATION OF THE AUTOMATIC TRACK FORMATION ALGORITHM

The evaluation of the ATF algorithm operating in a sliding window mode, based on fixed window simulation results, can be obtained as follows.

Denote by D the detection event at times 1 and 2 and by A the event of acceptance of the resulting preliminary track as a confirmed track by the end of the initiation window. Define the following states of a Markov chain:

1. Initial state: $D\rightarrow 2$, $\bar{D}\rightarrow 1$.
2. First detection: $D\rightarrow 3$, $\bar{D}\rightarrow 1$.
3. Second detection: $A\rightarrow 4$, $\bar{A}\rightarrow 1$.
4. Confirmed target.

The final confirmation probability (for a fixed window) of a true target's track is, in terms of the above quantities,

$$P_{C_{FW}} = P_D^2\, P_A \tag{A.1}$$

The transition probabilities of the Markov chain are

$$\begin{array}{ll} \pi_{11}=1-P_D & \pi_{12}=P_D \\ \pi_{21}=1-P_D & \pi_{23}=P_D \\ \pi_{31}=1-P_A & \pi_{34}=P_A \\ \multicolumn{2}{c}{\pi_{44}=1} \end{array} \tag{A.2}$$

and the initial state is

$$\mu_1(0) = 1 \tag{A.3}$$

The Monte Carlo simulations of the ATF/IMMPDA for a fixed window, N_W, yield $P_{C_{FW}}$; then P_A follows from the knowledge of P_D.

Note that the last step in the Markov chain is N_W-2 sampling periods of the system. Thus the probability of the last state of the chain after k chain time units is the confirmation probability from a sliding window after k+N_W-3 system sampling periods

$$\mu_C(k+N_W-3) = \mu_4(k) \tag{A.4}$$

and the expected confirmation time (in system sampling periods) is

$$\bar{t}_C = \sum_{i=1}^{\infty}(i+N_W-3)\mu_4(i) \tag{A.5}$$

The cumulative distribution function of the sliding window confirmation time is

$$P_{C_{SW}}(k+N_W-3) = \sum_{i=1}^{k}\mu_4(i) \tag{A.6}$$

References

[1] Y. Bar-Shalom, L. Campo and P.B. Luh, "From Receiver Operating Characteristic to System Operating Characteristic: Evaluation of a Large-Scale

Surveillance System", **Proc. EASCON 87**. Washington, DC, Oct. 1987. To appear in **IEEE** Trans. Auto. Control, 1989.
[2] Y. Bar-Shalom, "Multitarget-Multisensor Tracking I: Principles and Techniques", UCLA Extension / University of Maryland Short Course, 1987-8.
[3] Y. Bar-Shalom and T.E. Fortmann, **Tracking and** Data **Association**, Academic Press, 1988.
[4] Y. Bar-Shalom, K.C. Chang and H.A.P. Blom, "Tracking a Maneuvering Target using Input Estimation vs. the Interacting Multiple Model Algorithm", **IEEE Trans. Aerosp. Electronic Systems**, Nov. 1988.
[5] Y. Bar-Shalom, MULTIDAT - Multimodel Data Association Tracker, Interactive software, 1988.
[6] S.S. Blackman, **Multiple Target Tracking with** Radar **Applications**, Artech House, 1986.
[7] S. Blake and S.C. Watts, "A Multitarget Track-While-Scan Filter", **Proc. IEE Radar-87** Conf., London, England, Oct. 1987.
[8] H.A.P. Blom, "A Sophisticated Tracking Algorithm for ATC Surveillance Data", **Proc. Intn'l Radar** Conf., pp. 393-398, Paris, France, May 1984.
[9] H.A.P. Blom, "An Efficient Filter for Abruptly Changing Systems", **Proc. 23rd IEEE Conf. Decision** and Control, pp. 656-658, Las Vegas, NV, Dec. 1984.
[10] H.A.P. Blom and Y. Bar-Shalom, "The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients", **IEEE Trans. Auto.** Control, Aug. 1988.
[11] C.B. Chang and M. Athans, "State Estimation for Discrete Systems with Switching Parameters", **IEEE** Trans. Aerosp. Electronic Systems, AES-14: 418-425, May, 1978.
[12] C.B. Chang, K.P. Dunn and L.C. Youens, "A Tracking Algorithm for Dense Target Environments", **Proc. 1984 American Control Conf.**, San Diego, CA, June 1984.
[13] F.R. Castella, "Sliding Window Detection Probabilities", **IEEE Trans. Aerosp. Electronic** Systems, AES-12: 815-819, Nov. 1976.
[14] S.B. Colegrove, A.W. Davis and J.K. Ayliffe, "Track Initiation and Nearest Neighbors Incorporated into Probabilistic Data Association", **J. of** Electrical and Electronics Engrg.-Australia, Vol. 6, No. 3, Sept. 1986.
[15] A. Farina and F.A. Studer, **Radar Data** Processing, (Vol. I: Introduction and Tracking, Vol II: Advanced Topics and Applications), Research Studies Press, Letchworth, Hertfordshire, England and J. Wiley & Sons, 1985.
[16] R.J. Fitzgerald, "Development of Practical PDA Logic for Multitarget Tracking by Microprocessor", **Proc. 1986 American Control Conf., Seattle, WA**, June 1986.
[17] M. Gauvrit, "Bayesian Adaptive Filter for Tracking with Measurements of Uncertain Origin", **Automatica**, 20: 217-224, March 1984.
[18] J.E. Holmes, "The Development of Algorithms for the Formation and Updating of Tracks", **Proc. IEEE** 1977 Intn'l Radar Conf., London, Oct. 1977.
[19] A. Houles and Y. Bar-Shalom, "Multisensor Tracking of a Maneuvering Target in Clutter", **Proc. NAECON 1987**, to appear in **IEEE Trans. Aerosp. Electronic Systems**, 1989.
[20] T. Kurien, "Multitracker" in **Multitarget-Multisensor Tracking - Advanced** Applications (Y. Bar-Shalom, Coordinator), UCLA Extension Short Course, Jan. 1988.
[21] C.E. Muehe and R.M. O'Donnell, "Automating Radars for Air Traffic Control", **Proc. ELECTRO** '78, Boston, MA, May 1978.
[22] V. Nagarajan, R.N. Sharma and M.R. Chidambara, "An Algorithm for Tracking a Maneuvering Target in Clutter", **IEEE Trans. Aerosp. Electronic Systems**, AES-20: 560-573, Sept. 1984.
[23] V. Nagarajan, M.R. Chidambara and R.N. Sharma, "New Approach to Improved Detection and Tracking Performance in Track-While-Scan Radars", **IEE** Proc., 134F: 89-112, Feb. 1987.
[23] V. Nagarajan, M.R. Chidambara and R.N. Sharma, "Combinatorial Problems in Multitarget Tracking - a Comprehensive Solution", **IEE Proc., 134F: 113-118**, Feb. 1987.
[25] K.R. Pattipati and N.R. Sandell, Jr., "A Unified View of State Estimation in Switching Environments", **Proc. 1983 American Control Conf., San Francisco**, CA, June 1983.
[26] D.B. Reid, "An Algorithm for Tracking Multiple Targets", **IEEE Trans. Auto. Control**, AC-24: 843-854, Dec. 1979.
[27] J.K. Tugnait, "Detection and Estimation for Abruptly Changing Systems", **Automatica**, 18: 607-615, Sept. 1982.

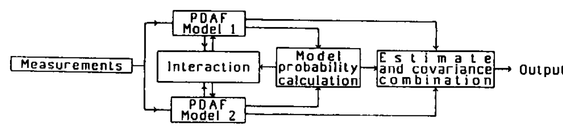

Fig. 1. The IMMPDAF (for two models).

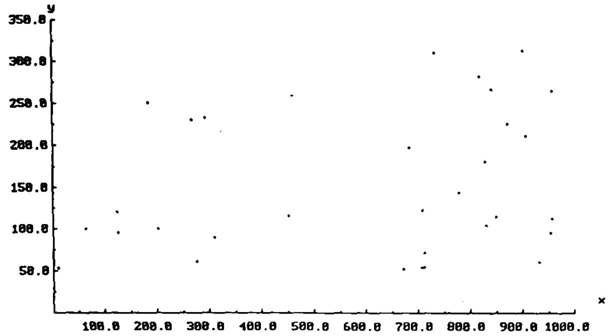

Fig. 2. One frame of the surveillance region with heavy clutter.

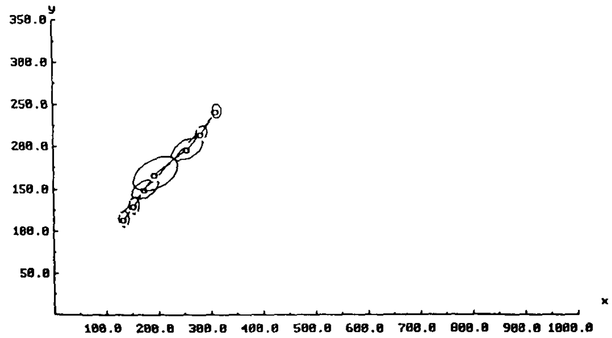

Fig. 3. True target track formed in a sample run.

Average number of false alarms per unit area		1E-4		1E-5	
Average number of false alarms in area $\sqrt{R_{11}R_{22}}$		2.5E-3		2.5E-4	
Average number of initiators in surveillance region		24		2.4	
Average number of false alarms in 99% gate					
in scan 2		1		.1	
in scan 3		.35		.035	
Target detection probability		.9	.8	.9	.8
Average number of false tracks after 7 scans					
in surveillance region	I	.61	2.6	1E-3	3E-3
	G	3.1	5.4	1E-3	4E-3
per unit area	I	2.6E-6	1.1E-5	4E-9	1.2E-8
	G	1.3E-5	2.3E-5	4E-9	1.7E-8
in area $\sqrt{R_{11}R_{22}}$	I	6.1E-5	2.7E-4	1E-7	3E-7
	G	3.1E-4	5.7E-4	1E-7	4E-7
Average TTP of false tracks after 7 scans	I	.29	.24	.95*	.13
	G	.29	.28	.50*	.27
Fraction of false tracks with TTP>.5	I	.20	.09	1*	0
	G	.15	.13	0*	.25
Fixed window (7 scans) target track confirmation probability					
	I	.68	.66	.74	.58
	G	.78	.74	.74	.58
Average TTP of target track	I				
at scan 3		.54	.50	.56	.53
at scan 4		.65	.54	.65	.58
at scan 5		.69	.59	.72	.66
at scan 6		.74	.61	.77	.70
at scan 7		.76	.63	.79	.72
Average TTP of target track	G				
at scan 3		.55	.50	.56	.53
at scan 4		.64	.56	.66	.59
at scan 5		.68	.60	.76	.68
at scan 6		.73	.62	.82	.72
at scan 7		.74	.64	.85	.76
Fraction of target tracks at scan 7 with TTP>.5					
	I	.84	.64	.87	.80
	G	.79	.69	.92	.81
Sliding window confirmation probability	I				
at scan 8		.75	.76	.81	.70
at scan 9		.82	.86	.89	.81
at scan 10		.92	.93	.95	.89
at scan 11		.94	.96	.97	.93
at scan 12		.96	.97	.99	.96
Sliding window expected confirmation time					
	I	7.98	7.78	7.90	8.21
	G	7.47	7.48	7.90	8.21

Table 1. Performance of the IMMPDAF ("I") vs. GPB1PDAF ("G") in track formation
(* statistically not significant - based on a single outcome in 1000 runs)