

Fast multiple target tracking using particle filters

Mark R. Morelande* and Darko Mušicki*

Melbourne Systems Laboratory

Department of Electrical and Electronic Engineering

The University of Melbourne, Australia

email: m.morelande@ee.unimelb.edu.au, d.musicki@ee.unimelb.edu.au

Abstract—A fast algorithm for accurate initiation and tracking of multiple targets is developed. The basis of the approach is the linear multi-target (LM) method in which exact joint data association for multiple targets is approximated by single target data association for each target with clutter density modified to include contributions from neighbouring targets. This enables multiple target tracking to be performed with the computational expense of single target tracking. Previously the LM method has been combined with tracking using a Gaussian approximation. In this paper accurate tracking is achieved through the use of a variant of the auxiliary particle filter. Simulation results show that the use of the particle filter is particularly advantageous in demanding scenarios where targets are in close proximity for long periods.

I. INTRODUCTION

The main difficulty in multiple target tracking is posed by measurement origin uncertainty. This arises because each measurement can be due to one of several targets or to some non-target related phenomena. Measurements from non-target related phenomena are referred to as clutter. The optimal way of resolving measurement origin uncertainty is to enumerate and evaluate all possible associations between the measurements and the targets. In an environment in which the number of targets is unknown it is also necessary to enumerate hypotheses proposing the existence of new targets. Exhaustive enumeration of all possible association hypotheses is an impossible task and sophisticated hypothesis management is required [2], [15]. Algorithms which use this approach are collectively referred to as multiple hypothesis trackers (MHTs).

The difficulty of developing hypothesis management techniques which reduce computation while maintaining performance led to the development of the joint integrated probabilistic data association filter (JIPDAF) [12]. The JIPDAF uses as its basis the joint probabilistic data association filter (JPDAF) [9]. The JPDAF tracks a fixed number of targets by enumerating all measurement-target associations at each scan and then combining them into a single component. This can be regarded as an extreme form of hypothesis management. The JIPDAF extends the JPDAF through the calculation of an “existence probability” which enables target tracks to be initiated, confirmed and terminated depending on the received measurements.

The requirement for joint association of several measurements with several targets limits the scenarios in

which the MHT and JIPDAF can be applied with reasonable computational expense. In particular, the number of targets in a given area cannot be too large since the number of association hypotheses increases exponentially with the number of targets. There has therefore been much interest in developing methods for performing joint data association in such a way that this exponential increase in computational expense is avoided with a minimal loss of accuracy [8], [16], [17], [18].

More recently, a technique which permits approximate joint data association with a computational expense which is linear in the number of targets was proposed in [13]. This technique will be referred to as linear multi-target (LM) tracking. The key idea in LM tracking is that each target performs single target data association with the neighbouring targets treated as clutter. This takes into account the presence of neighbouring targets without incurring exponentially increasing computational expense. In fact, LM tracking for multiple targets is only slightly more computationally expensive than performing single target data association for each target. The algorithms of [8], [17] also have a computational expense which is linear in the number of targets but perform poorly when a large number of targets are in close proximity [16]. The robustness of LM tracking to increases in the number of targets is demonstrated in [11].

As with any data association technique, optimal target state estimation using the LM method, in the Bayesian sense, requires a computational expense which increases exponentially with time. This prevents exact computation of the posterior distribution of the target state. In [13], probabilistic data association (PDA), in which the posterior density is approximated by a Gaussian by combining the association hypotheses at each time step [1], was used. The combination of the LM method with IPDA is referred to as the LMIPDAF. PDA is a reasonable approximation but tends to break down in more demanding scenarios involving large numbers of targets and/or clutter measurements. In such cases the use of a numerical technique which permits arbitrarily accurate approximation of the posterior distribution, albeit at additional computational expense, is desirable. This approach is pursued here with a particle filter (PF) used to approximate the posterior distribution of the target state. The resulting algorithm is referred to as the LM integrated PF (LMIPF).

Essentially, PFs use Monte Carlo simulation to approximate the integrals which arise in the evaluation of the posterior distribution [6]. This results in a representation of the posterior distribution comprised of a number of weighted random samples, or particles. The advantages of PFs, in comparison to other numerical techniques [3], lie in the ease with which they can be implemented, the generality with which they can be applied and their ability, when properly designed, to reduce the effects of the curse of dimensionality [5]. Convergence results for PFs under a wide range of conditions can be found in [4].

PFs are commonly implemented using the sequential importance sampling (SIS) framework which involves particle proposal, weight update and possibly resampling at each time step. Depending on the manner in which the PF is implemented data association is performed in the weight update, as in the bootstrap filter [10], or is part of both the particle proposal and weight update, as in a measurement-directed proposal [7]. A measurement-directed proposal provides superior performance for a given sample size and will be the approach used here. In particular, a variant of the auxiliary PF [14] is proposed. LM data association, which permits particles to be proposed and updated separately for each target using single target data association, greatly reduces the computational expense of the particle proposal and weight update steps. Simulations will demonstrate that this decrease in computational expense is achieved while maintaining good performance.

The paper is organised as follows. Section II defines the notation. Section III contains a brief review of the LM method. The particle filtering algorithm is described in Section IV. Section V contains the simulation results and conclusions are drawn in Section VI.

II. NOTATION

Measurements are made at times t_1, t_2, \dots . At each time step the LM algorithm maintains a number of tentative tracks. Let r_k denote the number of tentative tracks at time t_k . The relevant information concerning the i th track is contained in the dynamic state $x_{k,i}$ and the existence state. The dynamic state contains kinematic information such as target position and velocity and manoeuvring mode. The binary existence state permits track formation and deletion. The event that the i th track is following a target, i.e., the i th track exists, is denoted as $\chi_{k,i}$ with $\bar{\chi}_{k,i}$ denoting the converse.

Assume that the i th track is initiated at time k_0 . Then the existence state of the i th track evolves as

$$\mathbf{P}(\chi_{k,i} | \chi_{k-1,i}) = \eta, \quad (1)$$

$$\mathbf{P}(\bar{\chi}_{k,i} | \bar{\chi}_{k-1,i}) = 1, \quad k = k_0 + 1, k_0 + 2, \dots, \quad (2)$$

with $\mathbf{P}(\chi_{k_0,i}) = \rho_0$. Conditional upon target existence, the dynamic state $x_{k,i}$ evolves according to

$$x_{k,i} | x_{k-1,i} \sim N(f_k(x_{k-1,i}), Q_k), \quad (3)$$

for $k = k_0 + 1, k_0 + 2, \dots$, where $N(\mu, \Sigma)$ is the Gaussian distribution with mean μ and covariance matrix

Σ . If necessary, the transition probability (1) and transition density can differ between targets. At time k_0 , $x_{k_0,i} \sim \tau_{k_0,i}$.

At time t_k , m_k measurements are received and collected into $y_k = \{y_{k,1}, \dots, y_{k,m_k}\}$. The i th target is associated with a set of $m_{k,i}$ validated measurements which are obtained by performing gating in the usual manner. Let $G_{k,i} \subseteq \{1, \dots, m_k\}$ denote the set of indices of the measurements validated by the i th target. Then, $\theta_{j,i}$, $i = 1, \dots, r_k$, $j \in G_{k,i}$ is the event that the j th measurement is due to the i th target and $\theta_{0,i}$ denotes the event that the i th target has not been detected. Considering the i th target in isolation we have, for $j, l \in G_{k,i}$,

$$y_{k,l} | x_{k,i}, \theta_{j,i} \sim \begin{cases} N(h(x_{k,i}), R_k), & l = j, \\ c_k, & \text{otherwise,} \end{cases}$$

where c_k is the assumed probability density function for clutter measurements. Under $\theta_{0,i}$, $y_{k,l} \sim c_k$ for $l \in G_{k,i}$. Note that $\theta_{j,i}$, $j \in \{0\} \cup G_{k,i}$ are the complete collection of *single* target association hypotheses for the i th target. Thus if the j th measurement is not due to the i th target it is a clutter measurement. In multi-target tracking it is also necessary to consider the possibility that the j th measurement is due to some other target. The optimal approach does this by enumerating additional association hypotheses. In order to reduce computational expense the LM method does it by modifying the clutter density according to the proximity of neighbouring targets. This is described in Section III.

Targets are detected with probability P_D at each time step. If a target is detected, the probability that the target measurement falls in the validation gate of that target is P_G . When the notation p is used to denote probability density functions the particular density under consideration will be clear from the arguments.

The goal of the tracking algorithm is to recursively compute the posterior density of the dynamic state, denoted as $\pi_{k|k,i}$ at time t_k for the i th target, and the posterior probability of track existence for each tentative track. Here the LM method is used in conjunction with a particle filter to compute these quantities. The procedures are described in the following two sections.

III. REVIEW OF LINEAR MULTI-TARGET TRACKING

The key idea in LM tracking is that when updating one track, detections of targets followed by other tracks are unwanted measurements or clutter. Clutter measurement density is modulated to account for the additional source(s) of clutter, after which single target tracking is performed.

Denote with y^k the set of measurement up to and including time t_k :

$$y^k = \bigcup_{t=1}^k y_t.$$

Let

$$\nu_{k,i}(j) = \int p(y_{k,j} | x) \pi_{k|k-1,i}(x) dx \quad (4)$$

denote the *a priori* target measurement density for the i th track evaluated at the j th measurement. Denote the clutter measurement density at the j th measurement as $\rho_k(j) = c_k(y_{k,j})$.

The first step of the LM approach is to calculate *a priori* data association probabilities for each track, under the assumption that the selected measurements may have originated from the target or from clutter only. Under this assumption, the *a priori* probability that the j th measurement is the detection of the i th target is approximated by, for $j \in G_{k,i}$,

$$P_j^i \approx P_D P_G \mathbf{P}(\chi_{k,i} | y^{k-1}) \frac{\nu_{k,i}(j)}{\rho_k(j)} \frac{1}{\sum_{l \in G_{k,i}} \frac{\nu_{k,i}(l)}{\rho_k(l)}}. \quad (5)$$

The *a priori* clutter measurement density of the j th measurement when updating the i th track is, after including the contributions of neighbouring targets, given by

$$\rho_{k,i}(j) = \rho_k(j) + \sum_{\substack{d=1 \\ d \neq i}}^{r_k} \nu_{k,d}(j) \frac{P_j^d}{1 - P_j^d}. \quad (6)$$

The LM method involves using $\rho_{k,i}(j)$ rather than $\rho_k(j)$ when computing data association probabilities for the i th track and otherwise ignoring neighbouring tracks.

Automatic track formation is performed by initialising new tracks at each time step using two-point initiation. These new tracks are given a prior target state density, based on the two measurements used for initialisation, assigned an initial existence probability and added to the set of tentative tracks. Tracking of the existing tentative tracks is performed using the algorithm of Section IV with the existence probability updated using (19). Tentative tracks are classified as confirmed, i.e., they are deemed to belong to a target, or are terminated on the basis of the existence probability. Confirmation occurs when the existence probability exceeds a threshold P_c while termination occurs if the existence probability falls below a probability P_t . This simple track management is augmented with procedures for merging tracks which are very close for long periods of time. Details can be found in [13].

IV. PARTICLE FILTERING USING THE LM METHOD

In a PF, the posterior density of the i th target state at time t_{k-1} is represented by a set of particles $x_{k-1,i}^1, \dots, x_{k-1,i}^n$ and weights $w_{k-1,i}^1, \dots, w_{k-1,i}^n$ where n is the sample size. Upon receipt of measurements at time t_k , these particles and weights are modified to produce an approximation to the posterior density. In SIS the new particles and weights are produced by importance sampling which involves drawing samples from a proposal or importance density and applying a multiplicative adjustment to the weights. To prevent sample degeneracy, in which the weights become concentrated in only a few particles, resampling should be performed regularly. Resampling

involves selecting particles according to the weights so that particles with small weights will be removed while those with large weights will be selected many times.

A. Target state posterior density approximation

The best importance density to use, in the sense that it minimises the variance of the particle weights conditional upon the measurement history to time t_k and the particle trajectory to time t_{k-1} , is the optimal importance density (OID) [7]. However, the OID can be used only if the measurement equation is linear and Gaussian. In the following we use a variant of the auxiliary PF (APF) [14] to provide a natural extension of the OID for nonlinear measurement equations. The proposed method reduces to the OID when the measurement equation is linear.

The PF approximation to the posterior density of the i th target state at time t_{k-1} can be written as

$$\hat{\pi}_{k-1|k-1,i}(x) = \sum_{t=1}^n w_{k-1,i}^t \delta(x - x_{k-1,i}^t). \quad (7)$$

Using Bayes' rule leads to the following approximation to the posterior density at time t_k :

$$\tilde{\pi}_{k|k,i}(x) \propto p(y_k | x) \sum_{t=1}^n w_{k-1,i}^t p(x | x_{k-1,i}^t). \quad (8)$$

Equivalently,

$$\tilde{\pi}_{k|k,i}(x, t) \propto p(y_k | x) p(x | x_{k-1,i}^t) w_{k-1,i}^t, \quad (9)$$

where $t \in \{1, \dots, n\}$, an index on the mixture in (8), is referred to as an auxiliary variable. It is proposed to obtain samples from (9) using importance sampling with an importance density of the form

$$q(x, t | y^k) = \gamma_t(y_k) g(x | x_{k-1,i}^t, y_k), \quad (10)$$

where $\sum_{t=1}^n \gamma_t(y_k) = 1$. A draw from (10) is made by selecting $t = l$ with probability $\gamma_l(y_k)$ and then drawing $x_{k,i}^t$ from $g(\cdot | x_{k-1,i}^t, y_k)$. The probabilities $\gamma_t(y_k)$, $t = 1, \dots, n$ will be referred to as the first-stage weights. In [14], it is suggested to use first-stage weights of the form

$$\gamma_t(y_k) \propto w_{k-1,i}^t p(y_k | \mu_k^t), \quad (11)$$

where μ_k^t is some value which characterises the transition density $p(x_{k,i} | x_{k-1,i}^t)$. Since the best approach, if possible, is to draw samples from the OID, it is of interest to examine the relationship between the use of (10) with (11) and the use of the OID. In the OID updated weights are proportional to $w_{k-1,i}^t p(y_k | x_{k-1,i}^t)$. The first-stage weights used in the APF can be interpreted as an attempt to approximate this weight update. Thus the APF with first-stage weights as in (11) approximates the predictive likelihood at measurement y as follows:

$$p(y | x_{k-1}) = \int p(y | x_k, x_{k-1}) p(x_k | x_{k-1}) dx_k \quad (12)$$

$$\approx \int p(y | x_k, x_{k-1}) \delta(x_k - \mu) d\mu = p(y | \mu). \quad (13)$$

Thus, if the APF is regarded as an approximation to the OID, implicit in the use of the weights (11) is an approximation of the transition density by a delta function centred at some characteristic value of the state. This is a crude approximation which will, in general, result in a sampling procedure substantially different from the OID.

Since the form of the likelihood is the barrier preventing exact computation of (12), the approach taken here is to replace the likelihood by an approximate quantity which allows analytic computation of the integral (12). This is a strategy used in nonlinear filtering approximations such as the extended and unscented Kalman filters. In this paper the likelihood is approximated by linearising the measurement equation about the predicted target state, as in the extended Kalman filter. This requires computation of the Jacobian, denoted as $H = (\nabla_x h(x)'|_{x=f(x_{k-1})})'$ with $'$ denoting matrix transpose. Under the linearised approximation to the measurement equation, the first-stage weights are

$$\gamma_t(y_k) = w_{k-1,i}^t l(y_k|x_{k-1,i}^t) \left/ \sum_{d=1}^n \{w_{k-1,i}^d l(y_k|x_{k-1,i}^d)\} \right., \quad (14)$$

where

$$l(y_k|x_{k-1,i}^t) = 1 - P_D P_G \left\{ 1 - \sum_{j \in G_{k,i}} \frac{l(y_{k,j}|x_{k-1,i}^t)}{\rho_{k,i}(j)} \right\},$$

with $\rho_{k,i}(j)$ given in (6) and

$$l(y_{k,j}|x_{k-1,i}^t) = N(y_{k,j}; h(f_k(x_{k-1,i}^t)), H Q_k H' + R_k).$$

In the remainder of the paper the notation l will be used in place of p to denote pdfs derived using linearisation of the measurement equation. In keeping with the desire for the sampling procedure to be as close to the OID sampling procedure as possible, the density g used to propose those particles selected by the first-stage weights is a linearised approximation to the OID:

$$\begin{aligned} g(x|x_{k-1,i}^t, y^k) &= \frac{l(y_k|x, x_{k-1,i}^t) p(x|x_{k-1,i}^t)}{l(y_k|x_{k-1,i}^t)} \\ &= \sum_{j \in \{0\} \cup G_{k,i}} \alpha_{k,i}^t(j) N(x; \mu_{k-1,i}^t(j), \Sigma_{k-1,i}^t(j)), \end{aligned} \quad (15)$$

where

$$\begin{aligned} \alpha_{k,i}^t(j) &= \begin{cases} \frac{1 - P_D P_G}{l(y_k|x_{k-1,i}^t)}, & j = 0, \\ \frac{P_D P_G l(y_{k,j}|x_{k-1,i}^t)}{\rho_{k,i}(j) l(y_k|x_{k-1,i}^t)}, & j \in G_{k,i}, \end{cases} \\ \mu_{k,i}^t(j) &= \begin{cases} f(x_{k-1,i}^t), & j = 0, \\ f(x_{k-1,i}^t) + K_{k,i}^t \{y_{k,j} - f(x_{k-1,i}^t)\}, & j \in G_{k,i}, \end{cases} \\ \Sigma_{k,i}^t(j) &= \begin{cases} Q_k, & j = 0, \\ Q_k - K_{k,i}^t H_{k,i}^t Q_k, & j \in G_{k,i}, \end{cases} \end{aligned}$$

with

$$K_{k,i}^t = Q_k H_{k,i}^{t'} (H_{k,i}^t Q_k H_{k,i}^{t'} + R_k)^{-1}. \quad (16)$$

While the Jacobian matrix $H_{k,i}^t$ can be different for each particle, for instance $H_{k,i}^t = (\nabla_x h(x)'|_{x=f(x_{k-1,i}^t)})'$, it is significantly simpler computationally to use the same matrix for all particles. Here we use, for $t = 1, \dots, n$, $H_{k,i}^t = H_{k,i} = (\nabla_x h(x)'|_{x=f(\hat{x}_{k-1|k-1,i})})'$, where

$$\hat{x}_{k-1|k-1,i} = \sum_{t=1}^n w_{k-1,i}^t x_{k-1,i}^t. \quad (17)$$

Thus the Jacobian, along with the gain matrix $K_{k,i}^t$, need be computed just once for each target at each time step.

The final step is to assign weights to the proposed particles. The weight for the t th particle of the i th track can be found as

$$w_{k,i}^t \propto \frac{\hat{\pi}_{k|k,i}(x_{k,i}^t, t)}{q(x_{k,i}^t, t|y^k)} = \frac{p(y_k|x_{k,i}^t)}{l(y_k|x_{k,i}^t, x_{k-1,i}^t)}. \quad (18)$$

The ratio (18) accounts for the approximations involved in linearising the measurement equation. A summary of the procedure for producing a particle approximation to the posterior distribution of the i th target state at time t_k from a corresponding approximation at time t_{k-1} is given in Table I.

TABLE I
A RECURSION OF THE PARTICLE FILTER

- | |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"> 1) Compute the first-stage weights $\gamma_1(y_k), \dots, \gamma_n(y_k)$ using (14). 2) Select particle indices j^1, \dots, j^n such that $j^t = s$ with probability $\gamma_s(y_k)$. 3) Draw target states $x_{k,i}^t \sim g(\cdot x_{k-1,i}^{j^t}, y_k)$, $t = 1, \dots, n$ with g given in (15). 4) Compute the updated weights w_1^k, \dots, w_n^k using (18). |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

For a linear measurement equation, the linearised and true likelihoods are identical and the weights are even, i.e., $w_{k,i}^t = 1/n$. If the measurement equation is nonlinear, the variability of the weights will depend on the accuracy of the linearisation. In the scenario of Section V the accuracy of the linearised measurement equation is sufficient to produce almost equal weights for each particle. This is desirable since highly variable weights reduce accuracy.

Although the use of linearisation to approximate the OID is not new, the procedure suggested here is subtly different from existing methods. The difference between the proposed technique and the APF as proposed in [14] has been described above. Linearisation of the measurement equation was also suggested in [7] although not in an auxiliary variable framework. An important aspect of the use of auxiliary variables is that particles are selected for resampling, using the first-stage weights, prior to drawing new samples. This ensures a diverse particle set, and improved performance, since all particles will have unique values. In [7] resampling is performed after drawing new particles. This causes duplication of particle values and a loss of diversity.

B. Computation of the existence probability

The posterior probability of target existence for the i th track is calculated as [11]

$$\mathbf{P}(\chi_{k,i}|y^k) = \frac{(1 - \delta_{k,i})\mathbf{P}(\chi_{k,i}|y^{k-1})}{1 - \delta_{k,i}\mathbf{P}(\chi_{k,i}|y^{k-1})}, \quad (19)$$

where

$$\delta_{k,i} = P_D P_G \left(1 - \sum_{j \in G_{k,i}} \frac{\nu_{k,i}(j)}{\rho_{k,i}(j)} \right). \quad (20)$$

The quantities $\rho_{k,i}(j)$ are computed as in (6) while the prior probability of target existence is computed using (1) and (2). This leaves the quantities $\nu_{k,i}(j)$, $j \in G_{k,i}$. Recall that $\nu_{k,i}(j)$ is the prior target measurement density for the i th target computed at the j th measurement. This can be expanded as

$$\nu_{k,i}(j) = \int p(y_{k,j}|x_{k,i}) \pi_{k|k-1,i}(x_{k,i}) dx_{k,i}. \quad (21)$$

An approximation to the density $\pi_{k|k-1,i}$ can be obtained by stochastic prediction of the particles $x_{k-1,i}^1, \dots, x_{k-1,i}^n$. This results in the approximation

$$\hat{\pi}_{k|k-1}(x_{k,i}) = \sum_{t=1}^n w_{k-1,i}^t \delta(x_{k,i} - \tilde{x}_{k,i}^t), \quad (22)$$

where $\tilde{x}_{k,i}^t \sim N(f(x_{k-1,i}^t), Q_k)$, $t = 1, \dots, n$. Replacing $\pi_{k|k-1,i}$ in (21) with the approximation (22) gives

$$\hat{\nu}_{k,i}(j) = \sum_{t=1}^n w_{k-1,i}^t p(y_{k,j}|\tilde{x}_{k,i}^t). \quad (23)$$

The quantities $\hat{\nu}_{k,i}(j)$, $j \in G_{k,i}$ are computed using (23) and used in place of the exact values in (20) to approximate $\delta_{k,i}$. This approximation is then substituted into (19) to obtain an approximation to the posterior probability of target existence.

V. SIMULATION RESULTS

In this section the performance of the LMIPF is examined using Monte Carlo simulations. The previously proposed LMIPDAF is taken as a benchmark. It has been demonstrated in [13] that the LMIPDAF is capable of slightly better performance than the JIPDAF, which uses exact joint association for neighbouring targets. The superiority of the LMIPDAF over the JIPDAF arises because the computational efficiency of the LMIPDAF allows it to perform approximate multiple target data association on all tentative tracks while the JIPDAF can perform multiple target data association only on confirmed tracks.

Measurements are collected at equi-spaced time instants, $t_k = t_{k-1} + T$, $k = 1, 2, \dots$ where T is the sampling period. Targets move in two dimensions with a constant velocity. The i th target state is comprised of position and velocity in each direction and evolves according to $x_{k,i} = Fx_{k-1,i}$, $k = 1, 2, \dots$ with

$$F = I_2 \otimes \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix},$$

where \otimes is the Kronecker product and I_m is the $m \times m$ identity matrix. Note that target trajectories are generated without process noise but the filters assume process noise so that target states are assumed to evolve according to $x_{k,i}|x_{k-1,i} \sim N(Fx_{i,k-1}, Q)$ where

$$Q = I_2 \otimes 1/10 \begin{pmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{pmatrix}.$$

Target position is measured in polar coordinates using a sensor located the origin of the reference frame. The measurement noise covariance matrix is

$$R_k = \text{diag}((\pi/180)^2, 16).$$

Each target is detected with probability $P_D = 0.9$ and the gating probability is set to $P_G = 0.99$ in order to minimise the number of validated measurements.

A scenario with five crossing target trajectories is considered. Targets move with a speed of 15 m/s and are tracked for 70 time steps with a sampling period of $T = 1$ s. The trajectories of all targets meet at the 40th time step. The clutter is Poisson distributed with density 0.02 pts/(scan·m·rad) throughout the surveillance region. The angle between the neighbouring trajectories is used to vary the severity of the scenario. A small angular separation increases the severity of scenario since it increases the amount of time that the targets share measurements.

Performance is measured by the mean number of confirmed true tracks for a given number of confirmed false tracks. A true track is one which has been initiated from target measurements or has a state estimate sufficiently close to the true target state. The mean number of confirmed true tracks is estimated over 1000 realisations of the scenario described above for the LMIPDAF and the LMIPF using 200, 500 and 1000 particles. The confirmation and termination probabilities are such that the false track confirmation rate is one per 50 time steps. The results are given in Figures 1 and 2 for angular separations of 10 degrees and 5 degrees, respectively. Both algorithms perform well with angular separations of 10 degrees although it is clear that a sample size of at least 500 particles is required for the LMIPF. The following comments regarding the performance of the LMIPF apply only for these larger sample sizes. Compared to the LMIPF, the LMIPDAF is quicker to confirm tracks but has a greater tendency to lose tracks when the targets are in close proximity. Evidence of this latter characteristic is the perceptibly larger decrease in the number of confirmed true tracks before and after the 40th time step for the LMIPDAF compared to the LMIPF. The results for an angular separation of 5 degrees more clearly demonstrate the superiority of the LMIPF over the LMIPDAF for tracking in dense target environments. The mean number of true tracks confirmed by the LMIPF is significantly larger across the entire observation interval and the dip in performance immediately after the target trajectories cross is less marked.

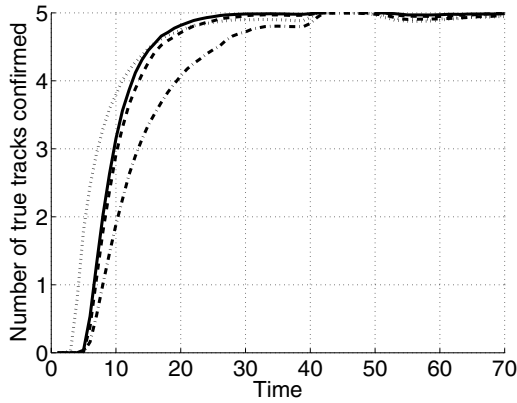


Fig. 1. Mean number of confirmed true tracks plotted against time for the LMIPDAF (dotted) and the LMIPF with 200 particles (dash-dot), 500 particles (dashed) and 1000 particles (solid). The angular separation between neighbouring target trajectories is 10 degrees.

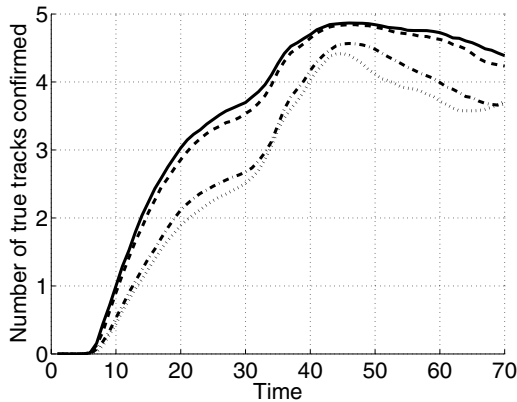


Fig. 2. Mean number of confirmed true tracks plotted against time for the LMIPDAF (dotted) and the LMIPF with 200 particles (dash-dot), 500 particles (dashed) and 1000 particles (solid). The angular separation between neighbouring target trajectories is 5 degrees.

Computational expense is an important consideration when using a numerical method such as a particle filter. The results of Figures 1 and 2 indicate that significant improvement in performance is obtained by increasing the sample size of the LMIPF from 200 to 500 particles but only marginal improvement is obtained by a further increase to 1000 particles. This implies that, for the scenario under consideration, a sample size of 500 particles provides the best trade-off between computational expense and performance. With this relatively small sample size the computational expense of the LMIPF is about four times that of the LMIPDAF. Even with this four-fold increase in computational expense over the LMIPDAF, the LMIPF is remarkably computationally efficient compared to algorithms which perform exact multi-target data association such as the MHT and JPDAF.

VI. CONCLUSIONS

An algorithm for automatic track formation and maintenance of multiple targets was proposed. The algorithm

combines the linear multi-target method, so-called because its computational expense is linear in the number of targets and measurements, with particle filtering to arrive at a fast accurate solution. The performance of the proposed algorithm was examined for a five-target scenario with positions measured in polar coordinates. Simulation results showed that, compared to an existing algorithm which employs a Gaussian approximation, the proposed algorithm performs particularly well in demanding scenarios where targets are in close proximity for a long period of time. Importantly, this performance is achieved with a reasonably small sample size. Thus, combining the linear multi-target method with particle filtering permits reliable multiple target track formation and maintenance in dense target environments with modest computational expense.

REFERENCES

- [1] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11:451–460, 1975.
- [2] S.S. Blackman. *Multiple-target Tracking with Radar Applications*. Artech House, 1986.
- [3] R.S. Bucy and K.D. Senne. Realization of optimum discrete-time nonlinear estimators. In *Proceedings of the Symposium on Nonlinear Estimation Theory and Its Applications*, pages 6–17, San Diego, USA, 1971.
- [4] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, 50(3):736–746, 2002.
- [5] F. Daum and J. Huang. Curse of dimensionality and particle filters. In *Proceedings of the Aerospace Conference*, 2003.
- [6] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, 2001.
- [7] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10:197–208, 2000.
- [8] R.J. Fitzgerald. Development of practical PDA logic for multitarget tracking by microprocessor. In Bar-Shalom Y., editor, *Multitarget-Multisensor Tracking: Advanced Applications*, chapter 1. Artech House, 1990.
- [9] T.E. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, 8(3):173–183, 1983.
- [10] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings Part F*, 140(2):107–113, 1993.
- [11] D. Mušicki. Limits of linear multi-target tracking. In *Proceedings of the International Conference on Information Fusion*, Philadelphia, USA, 2005.
- [12] D. Mušicki and R. Evans. Joint integrated probabilistic data association: JIPDA. *IEEE Transactions on Aerospace and Electronic Systems*, 40(3):1093–1099, 2004.
- [13] D. Mušicki, B.E. La Scala, and R. Evans. Multi-target tracking in clutter without clutter assignment. In *Proceedings of the IEEE International Conference on Decision and Control*, Bahamas, 2004.
- [14] M.K. Pitt and N. Shephard. Filtering via simulation: auxiliary particle filters. *Journal of the American Statistical Association*, 94:590–599, 1999.
- [15] D.B. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, 1979.
- [16] J.A. Roecker. A class of near optimal JPDA algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 30(2):504–510, 1994.
- [17] J.A. Roecker and G.L. Phillis. Suboptimal joint probabilistic data association. *IEEE Transactions on Aerospace and Electronic Systems*, 29(2):510–517, 1993.
- [18] B. Zhou and N.K. Bose. Multitarget tracking in clutter: Fast algorithms for data association. *IEEE Transactions on Aerospace and Electronic Systems*, 29(2):352–363, 1993.