Sequential Monte Carlo Algorithms for Joint Target Tracking and Classification Using Kinematic Radar Information

Donka Angelova
Central Laboratory for Parallel Processing
Bulgarian Academy of Sciences, 25A Acad.
G. Bonchev St, 1113 Sofia, Bulgaria
donka@bas.bg

Lyudmila Mihaylova
Department of Electrical and Electronic
Engineering, University of Bristol
Woodland Road, Bristol BS8 1UB, UK
mila.mihaylova@bristol.ac.uk

Abstract – *This paper considers the problem of joint maneuvering target tracking and classification. Based on recently proposed Monte Carlo techniques, a multiple model (MM) particle filter and a mixture Kalman filter (MKF) are designed for two-class identification of air targets: commercial and military aircraft. The classification task is carried out by processing radar measurements only, no class (feature) measurements are used. A speed likelihood function for each class is defined using a priori information about speed constraints. Class-dependent speed likelihoods are calculated through the state estimates of each class-dependent tracker. They are combined with the kinematic measurement likelihoods in order to improve the process of classification. The two designed estimators are compared and evaluated over a rather complex target trajectory. The results are demonstrating the usefulness of the proposed scheme for the incorporation of an additional speed information. Both filters illustrate the opportunity of the particle filtering and MKF to incorporate constraints in a natural way, providing reliable tracking and correct classification.*

Keywords: Joint tracking and classification, particle filter, multiple model, maneuvering target tracking, mixture Kalman filtering

1 Introduction

Recently there has been a great interest in the problem of joint target tracking and classification. Actually, the simultaneous implementation of these two important tasks in the surveillance systems facilitates the situation assessment, resource allocation and decision-making [1, 2]. *Classification* (or *identification*) usually includes target allegiance determination and/or target profile assessment such as *vehicle*, *ship* or *aircraft* type. Target class information could be obtained from an *electronic support measure* (*ESM*) sensor, friend-and-foe identification system, high resolution radar or other identity sensors. It could be inferred from a tracker, using kinematic measurements only or in a combination with identity sensors. Target type knowledge applied to the tracker can improve tracking performance by the possibility of selecting appropriate target models. Classification information can assist in correct data association and false tracks elimination in multiple target tracking systems.

Two basic approaches to classification exist based on *Bayesian* and *Dempster-Shafer theories* [3, 2, 1]. Challa and Pulford [4] reveal the feedback loop between tracking and identification and introduce the notion of *joint tracking and classification (JTC)*. They suggest a Bayesian algorithm for JTC using ESM and radar data. The numerical implementation of their algorithm utilizes a grid-based approach. It is well known that the computational efficiency of the *grid-based algorithms* depends on the state vector dimension. In contrast to the grid-based algorithms, the *Monte Carlo algorithms* are more easily implementable for highly dimensional systems. Feasible implementations of Bayesian JTC via particle filtering are reported in [5, 6, 2, 7]. We have to mention that [5] is one of the first papers devoted to the application of the particle filtering technique to tracking and identification of two closely spaced objects in clutter. Particle filter for tracking and classifying multiple targets is proposed in [6] as well. Automatic target recognition is realized by the inclusion of radar cross section measurements into the measurement vector.

The Monte Carlo approach allows for an accurate representation of joint state and class probability distributions. This is guaranteed by calculating all integrals as accurately as possible [2] and is achieved at the expense of increased computational costs. The highly non-linear relationships between state and class measurements and non-Gaussian noise processes can be easily processed by the particle filtering technique. In addition, flight envelope constraints, particularly useful for this task, can be incorporated into the filtering algorithm in a natural and consistent way [8].

The authors of [2] suggest a unified algorithm for joint tracking and identification in the framework of the Bayesian theory. A bank of filters, covering the state and feature space are run in parallel with each filter matched to a different target class. An example of the successful application of this particle filter to littoral tracking with classification is presented in [7]. The authors assign a target class to each land or water region and use a reflecting boundary condition to enforce the region constraints. A special feature of this algorithm is that the number of particles for each class remains constant during the tracking process. The class-conditioned independent filters stay in position “alert” and the filtering system can “change its mind” regarding the class identification if changes in the target behavior occur. This feature makes the filter versatile, but in some cases (e.g. for maneuvering target tracking) it could lead to misclassification or may increase the computational load due to a delayed stopping time of the unlikely filters.

In the present paper, motivated by the results reported in [7], we develop two *sequential* Monte Carlo algorithms: a particle filter and a mixture Kalman filter (MKF) for solving the problem of tracking and classifying *a maneuvering target* using kinematic measurements only. Two air target classes are considered: *commercial* aircraft (slowly maneuverable, mainly straight line) and *military* aircraft (highly maneuverable turns are possible). We should be able to understand which type of aircraft we are observing. In view the fact that both types of aircraft can perform slow maneuvers, the recognition can only be achieved during the aircraft's maneuvers with high speed and acceleration. For this purpose, a bank of two *multiple model* (MM) class-dependent particle filters is designed and implemented. The novelty of the paper relies also on accounting for two kinds of constraints : both on the *acceleration* and on the *speed*. We show that "hard constraints" can be naturally incorporated into the Monte Carlo framework. Two speed likelihood functions are defined based on a prior information about speed constraints of each class. Such kind of constraints are incorporated in other approaches for decision making [9]. At each filtering step, the estimated speed from each class-dependent filter is used to calculate a class-dependent speed likelihood and together with kinematic likelihood both are improving the classification process.

The remaining part of the paper is organized as follows. Section 2 summarizes the Bayesian formulation of the JTC problem according to [2, 7, 10]. Section 3 presents a developed MM particle filter and MKF using both speed and acceleration constraints. Simulation results are given in Section 4, and conclusions generalized in Section 5.

2 Bayesian joint target tracking and classification

Consider the following model of a discrete-time jump Markov system, describing the target dynamics and sensor measurements

$$x_k = F(m_{k-1})\, x_{k-1} + G(m_{k-1})\, u_{k-1} + B(m_{k-1})\, w_{k-1}, \quad (1)$$

$$z_k = h(m_k, x_k) + D(m_k)\, v_k, \qquad k = 1, 2, \ldots, \quad (2)$$

where $x_k \in \mathbb{R}^{n_x}$ is the *base (continuous) state* vector with transition matrix F, $z_k \in \mathbb{R}^{n_z}$ is the measurement vector with measurement function h, and $u_k \in \mathbb{U}$ is a known control input. The noise processes w_k and v_k are *independent identically distributed* (*i.i.d.*) Gaussian having characteristics $w_k \sim N(0,Q)$ and $v_k \sim N(0,R)$, respectively. w_k is the random input vector, and v_k is the random measurement error vector. All vectors and matrices are assumed of appropriate dimensions. The *modal (discrete) state* $m_k \in \mathbb{S} \triangleq \{1, 2, \ldots, s\}$ is a time-homogeneous first-order Markov chain with transition probabilities

$$\pi_{ij} \triangleq Pr\{m_k = j \mid m_{k-1} = i\},\ (i, j \in \mathbb{S}) \quad (3)$$

and initial probability distribution $P_1(i) \triangleq Pr\{m_1 = i\}$ for $i \in \mathbb{S}$, such that $P_1(i) \geq 0$, and $\sum_{i=1}^{s} P_1(i) = 1$. We assume that the target belongs to one of the M classes $c \in C$ where $C = \{c_1, c_2, \ldots, c_M\}$ represents the set of the target classes. Generally, the number of the discrete states $s = s(c)$, the initial probability distribution $P_1^c(i)$ and the transition probability matrix $\pi = [\pi_{ij}]^c$, $i, j \in \mathbb{S}$ are different for each target class.

The joint state and class is time varying with respect to the state and time invariant with respect to the class [2]. Let

$$\left\{Z^k, Y^k\right\} = \{z_i, y_i\} : i = 1, \ldots, k \quad (4)$$

be the cumulative set of *kinematic* (Z^k) and *class (feature)* measurements (Y^k) up to time k.

The *goal* of the joint tracking and classification task is to estimate the *state* x_k and the *posterior classification probabilities* $P\left(c \mid \left\{Z^k, Y^k\right\}\right), c \in C$ based on all available measurement information $\left\{Z^k, Y^k\right\}$.

If we can construct the *posterior joint state-class probability density function* (pdf) $p\left(x_k, c \mid \left\{Z^k, Y^k\right\}\right)$, then the posterior classification probabilities can be obtained by marginalization over x_k:

$$P\left(c \mid \left\{Z^k, Y^k\right\}\right) = \int_{x_k} p\left(x_k, c \mid \left\{Z^k, Y^k\right\}\right) dx_k. \quad (5)$$

Suppose that we know the posterior joint state-class pdf $p\left(x_{k-1}, c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right)$ at time instant $k-1$. According to the Bayesian framework, $p\left(x_k, c \mid \left\{Z^k, Y^k\right\}\right)$ can be computed recursively from $p\left(x_{k-1}, c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right)$ in two steps – *prediction* and *measurement update* [2, 7].

The predicted state-class pdf $p\left(x_k, c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right)$ at time k is given by the equation

$$p\left(x_k, c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right) = \quad (6)$$
$$\int_{x_{k-1}} p\left(x_k \mid x_{k-1}, c\right) p\left(x_{k-1}, c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right) dx_{k-1},$$

where the class- and state-conditioned state prediction pdf $p\left(x_k \mid x_{k-1}, c, \left\{Z^{k-1}, Y^{k-1}\right\}\right)$ is obtained from the state transition equation (1)

$$p\left(x_k \mid x_{k-1}, c, \left\{Z^{k-1}, Y^{k-1}\right\}\right) = \quad (7)$$
$$\sum_{j=1}^{s(c)} p\left(x_k \mid x_{k-1}, m_k = j, \left\{Z^{k-1}, Y^{k-1}\right\}\right)$$
$$\times P\left(m_k = j \mid x_{k-1}, c, \left\{Z^{k-1}, Y^{k-1}\right\}\right) =$$
$$\sum_{j=1}^{s(c)} p\left(x_k \mid x_{k-1}, m_k = j, \left\{Z^{k-1}, Y^{k-1}\right\}\right)$$
$$\times \sum_{l=1}^{s(c)} \pi_{lj} P\left(m_{k-1} = l \mid c, \left\{Z^{k-1}, Y^{k-1}\right\}\right).$$

The form of the conditional pdf of the measurements

$$p\left(\{z_k, y_k\} \mid x_k, c\right) = \lambda_{\{x_k, c\}}\left(\{z_k, y_k\}\right) \quad (8)$$

is usually known. This is the likelihood of the joint state and feature and has a key role in the classification algorithm.

It should be noted that because in our case we don't have feature measurements, the set $\{Y^k\}$ is replaced in the MM particle filter and in the MKF by the speed estimates from the M classes. Together with a speed envelope which form is given in subsection 3.3, they form a *virtual* "feature measurement".

When the measurements $\{z_k, y_k\}$ arrive, the update step can be completed

$$p\left(x_k, c \mid \left\{Z^k, Y^k\right\}\right) = \frac{\lambda_{\{x_k,c\}}\left(\{z_k, y_k\}\right) p\left(x_k, c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right)}{p\left(\{z_k, y_k\} \mid \{Z^{k-1}, Y^{k-1}\}\right)}, \tag{9}$$

where

$$p\left(\{z_k, y_k\} \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right) = \sum_{c \in C} \int_{x_k} p\left(\{z_k, y_k\} \mid x_k, c\right) p\left(x_k, c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right) dx_k. \tag{10}$$

The recursion (6)-(9) begins with the prior density $P\left\{x_1, c\right\}, x_1 \in \mathbb{R}^{n_x}, c \in C$, which is assumed known.

Using Bayes' theorem, the posterior probability of the discrete state m_k for class c is expressed by

$$P\left(m_k = j \mid c, \left\{Z^k, Y^k\right\}\right) = \frac{1}{l_k} p\left(\{z_k, y_k\} \mid m_k = j, c, \left\{Z^{k-1}, Y^{k-1}\right\}\right) \times \sum_{l=1}^{s(c)} \pi_{lj} P\left(m_{k-1} = l \mid c, \left\{Z^{k-1}, Y^{k-1}\right\}\right), \tag{11}$$

where l_k is a normalizing constant. Eq. (11) is substituted in (7) in order to predict the state pdf at time $k+1$.

Then the target classification probability is calculated by the equation

$$P\left(c \mid \left\{Z^k, Y^k\right\}\right) = \frac{p\left(\{z_k, y_k\} \mid \ c, \left\{Z^{k-1}, Y^{k-1}\right\}\right) P\left(c \mid \left\{Z^{k-1}, Y^{k-1}\right\}\right)}{\sum_{c \in C} p\left(\{z_k, y_k\} \mid \ c, \{Z^{k-1}, Y^{k-1}\}\right) P\left(c \mid \{Z^{k-1}, Y^{k-1}\}\right)} \tag{12}$$

with an initial prior target classification probability $P_1(c)$,

$\sum_{c \in C} P_1(c) = 1$.

The state estimate $\hat{x}_k^c$ for each class c

$$\hat{x}_k^c = \int_{x_k} x_k p\left(x_k, c \mid \left\{Z^k, Y^k\right\}\right) dx_k, \ c \in C \tag{13}$$

takes part in the calculation of the *combined* state estimate

$$\hat{x}_k = \sum_{c \in C} \hat{x}_k^c P\left(c \mid \left\{Z^k, Y^k\right\}\right). \tag{14}$$

It is obvious from (6)-(14) that the estimates, needed for each class, can be calculated independently from the other classes. Therefore, the JTC task can be accomplished by the simultaneous work of M independent filters [11]. The scheme of the particle filter bank, implemented in the present paper is described in Section 3.

3 Maneuvering target tracking and classification

3.1 Maneuvering target model

The two-dimensional target dynamics is given by

$$x_k = F x_{k-1} + G\left(u_{k-1} + w_{k-1}\right), \ k = 1, 2, \ldots, \tag{15}$$

where the state vector $x = (x, \dot{x}, y, \dot{y})'$ contains target positions and velocities in the horizontal (Oxy) Cartesian coordinate frame. The control input vector $u = (a_x, a_y)'$ includes target accelerations along x and y coordinates. The process noise $w = (w_x, w_y)'$ models perturbations in the accelerations. The transition matrices F and G are [12]

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad G = \begin{bmatrix} \frac{T^2}{2} & 0 \\ T & 0 \\ 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix}, \tag{16}$$

where T is the sampling interval and $B = G$. The target is assumed to belong to one of two classes ($M = 2$), representing either a lower speed *commercial aircraft* with limited maneuvering capability (c_1) or a highly maneuvering *military aircraft* (c_2) [4]. The flight envelope information comprises speed and acceleration constrains, characterizing each class. The speed $\mathrm{v} = \sqrt{\dot{x}^2 + \dot{y}^2}$ of each class is limited respectively to the interval:

$\{c_1 : \mathrm{v} \in (100, 300)\}$ $[m/s]$ and
$\{c_2 : \mathrm{v} \in (150, 650)\}$ $[m/s]$.

The range of the speed overlap section is $[150, 300]$ $[m/s]$. The control inputs are restricted to the following sets of accelerations:

$\{c_1 : u \in (0, +2g, -2g)\}$ $[m/s^2]$ and
$\{c_2 : u \in (0, +5g, -5g)\}$ $[m/s^2]$,

where $g = 9.81$ $[m/s^2]$ is the acceleration due to gravity. The acceleration process u_k is a Markov chain with five states (modes) $s(c_1) = s(c_2) = 5$ [13]:

1. $a_x = 0, \quad a_y = 0$ 2. $a_x = A, \quad a_y = A$
3. $a_x = A, \quad a_y = -A$ 4. $a_x = -A, \quad a_y = A$
5. $a_x = -A, \quad a_y = -A$,

where $A = 2g$ stands for class c_1 target and $A = 5g$ refers to the class c_2. The initial probabilities of the Markov chain are selected as follows: $P_1(1) = 0.6$, $P_1(2) = P_1(3) = P_1(4) = P_1(5) = 0.1$. The matrix π of transition probabilities $\pi_{ij}, i, j \in \mathbb{S}$ is assumed of the same form for both types of targets:

$$\pi = \begin{bmatrix} 0.70 & 0.10 & 0.05 & 0.10 & 0.05 \\ 0.15 & 0.70 & 0.05 & 0.05 & 0.05 \\ 0.15 & 0.05 & 0.70 & 0.05 & 0.05 \\ 0.15 & 0.05 & 0.05 & 0.7 & 0.05 \\ 0.15 & 0.05 & 0.05 & 0.05 & 0.70 \end{bmatrix} \tag{17}$$

The standard deviations of the process noise $w \sim N(0, diag(\sigma_{wx}^2, \sigma_{wy}^2))$ are different for each mode and class:

$$\{c_1 : \sigma_w^j = 5.5\ [m/s^2], j = 1, \dots, 5\} \text{ and}$$
$$\{c_2 : \sigma_w^1 = 7.5,\ \sigma_w^j = 17.5\ [m/s^2], j = 2, \dots, 5\},$$

where $(\sigma_{wx} = \sigma_{wy} = \sigma_w)$.

3.2 Measurement model

The measurement model at time k is described by

$$z_k = h(x_k) + v_k, \tag{18}$$

where

$$h(x) = \left(\sqrt{x^2 + y^2}, \arctan \frac{x}{y}\right)'. \tag{19}$$

The measurement vector $z = (D, \beta)'$ contains the distance to the target D and bearing β, measured by the radar. The parameters of the measurement error vector $v \sim N(0, R)$, $R = diag(\sigma_D^2, \sigma_\beta^2)$ are as follows: $\sigma_D = 100.0\ [m]$; $\sigma_\beta = 0.15$ [deg].

3.3 Speed constraints

Acceleration constraints are imposed on the filter operation by the use of an appropriate control input in the target model. The speed constraints are enforced through the speed likelihood functions. They are constructed based on the speed envelope information (3.1). If we assume that

$$g_1\left(\mathrm{v}_k^{c_1}\right) = \begin{cases} 0.8 & \text{if } \mathrm{v}_k^{c_1} \le 100 \quad [\mathrm{m/s}] \\ 0.8 - \kappa_1\left(\mathrm{v}_k^{c_1} - 100\right) & \text{if } (100 < \mathrm{v}_k^{c_1} \le 300) \\ 0.1 & \text{if } \mathrm{v}_k^{c_1} > 300 \quad [\mathrm{m/s}] \end{cases}$$

and

$$g_2\left(\mathrm{v}_k^{c_2}\right) = \begin{cases} 0.1 & \text{if } \mathrm{v}_k^{c_2} \le 150 \quad [\mathrm{m/s}] \\ 0.1 + \kappa_2\left(\mathrm{v}_k^{c_2} - 150\right) & \text{if } (150 < \mathrm{v}_k^{c_2} \le 650) \\ 0.95 & \text{if } \mathrm{v}_k^{c_2} > 650 \quad [\mathrm{m/s}] \end{cases}$$

for $\kappa_1 = 0.7/200$ and $\kappa_2 = 0.85/500$, then the class-conditioned speed likelihood functions will have the form, depicted in Fig. 1.

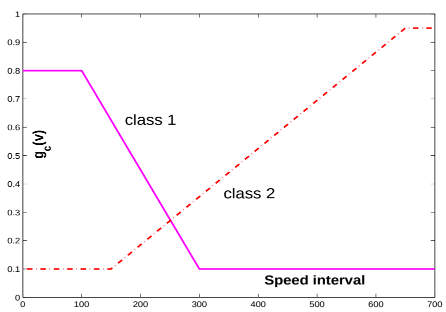

Fig. 1: Speed likelihood functions

According to the problem formulation, presented in Section 2, two class-dependent filters work in parallel with N_c number of particles for each class. At time step k, each filter gives a state estimate $\{\hat{x}_k^c,\ c = 1, 2\}$. Let us assume, that the estimated speed from the previous time step, $\{\hat{\mathrm{v}}_{k-1}^c,\ c = 1, 2\}$, is a kind of "feature measurement".

The likelihood $\lambda_{\{x_k,c\}}(\{z_k, y_k\})$ is factorized [2]

$$\lambda_{\{x_k,c\}}(\{z_k, y_k\}) = f_{x_k}(z_k)\, g_c\left(y_k^c\right), \tag{20}$$

where $y_k^c = \hat{\mathrm{v}}_{k-1}^c$. Practically, the normalized speed likelihoods represent estimated by the filters speed-based class probabilities. The posterior class probabilities are modified by this additional speed information at each time step k. The inclusion of the speed likelihoods is done after some "warming-up" interval, comprising filter initialization

3.4 Multiple model particle filter algorithm

Assuming two classes of targets (commercial and non commercial), we design a bank of two *independent* particle filters for each class. Every particle filter is based of multiple models for the unknown target acceleration u_k. The hybrid particle $\mathbf{x} = \{x, m, c\}$ contains then all the necessary information about the target state, mode and class.

The scheme of each particle filter incorporates the steps:

1. Initialization, $k = 1$.
   For class $c = 1, 2, \dots, M$ set $P(c) = P_1(c)$
   * For $j = 1, \dots, N_c$, sample
   $\left\{x_1^{(j)} \sim p_1(x_1, c),\ m_1^{(j)} \sim \{P_1^c(m)\}_{m=1}^{s(c)},\ c^{(j)} = c\right\}$
   and set $k = 2$.
   End for c

2. For $c = 1, \dots, M$ (possibly in parallel) execute
   * Prediction step
   For $j = 1, \dots, N_c$ generate samples
   $m_{k-1}^{(j)} \sim \{\pi_{lm}^c\}_{m=1}^{s(c)}$ for $l = m_{k-2}^{(j)}$ and $c^{(j)} = c$
   $x_k^{(j)} = F x_{k-1}^{(j)} + G u_{k-1}(m_{k-1}^{(j)}, c) + G w_{k-1}$,
   $w_{k-1} \sim N(0, Q(m_{k-1}^{(j)}, c))$,
   * Measurement processing step
   on receipt of a new measurement $\{z_k, y_k\}$:
   For $j = 1, \dots, N_c$ evaluate the weights
   $W_k^{(j)} = f(z_k \mid x_k^{(j)}) g_c\left(y_k^c\right)$,
   where $f(z_k \mid x_k^{(j)}) = N(z_k; h(x_k^{(j)}), R)$
   and $g_c\left(y_k^c\right) = g_c\left(\hat{\mathrm{v}}_{k-1}^c\right)$;
   calculate
   $p\left(\{z_k, y_k\} \mid c, \{Z^{k-1}, Y^{k-1}\}\right) = \sum_{j=1}^{N_c} W_k^{(j)}$
   set $L(c) = \sum_{j=1}^{N_c} W_k^{(j)}$
   * Selection step
   normalize the weights $W_k^{(j)} = W_k^{(j)} / \sum_{j=1}^{N_c} W_k^{(j)}$
   resample with replacement N_c particles
   $(x_k^{(j)}; j = 1, \dots, N_c)$ from the set
   $(x_k^{(l)}; l = 1, \dots, N_c)$ according to the weights
   * Compute updated state estimate and posterior mode probabilities
   $\hat{x}_k^c = \frac{1}{N_c} \sum_{j=1}^{N_c} x_k^{(j)}$,
   $P(m_k = l) = \frac{\sum (m_k^{(j)} = l, j \in \{1, \dots, N_c\})}{\sum_{j=1}^{N_c} m_k^{(j)}}$, $l = 1, \dots, s(c)$
   End for c

3. Output: Compute posterior class probabilities and combined output estimate

   $P\left(c \mid \left\{Z^k, Y^k\right\}\right) = \frac{L(c)P\left(c|\left\{Z^{k-1},Y^{k-1}\right\}\right)}{\sum_{c=1}^{M} L(c)P(c|\{Z^{k-1},Y^{k-1}\})},$

   $\hat{x}_k = \sum_{c=1}^{M} P\left(c \mid \left\{Z^k, Y^k\right\}\right) \hat{x}_k^c,$

4. Set $k \longleftarrow k+1$ and go to step 2.

Actually, there is fusion at two levels: *(i)* of state estimates and their pdfs with respect to the classes; and *(ii)* regarding the acceleration grid within each particle filter.

4 The mixture Kalman filter algorithm

The mixture Kalman filter (MKF) [14, 15] is another sequential Monte Carlo estimation technique which has been successfully applied to different problems in target tracking and digital communications (See e.g. [16]). The MKF is essentially a bank of Kalman filters (KFs) or extended KFs run with Monte Carlo sampling approach. The MKF is derived for state-space models in special form, namely *conditional dynamic linear model*, *conditional linear Gaussian model*, or *partially linear Gaussian model*:

$$\begin{cases} x_k = F_{\lambda_{k-1}} x_{k-1} + G_{\lambda_{k-1}}(u_{k-1} + w_{k-1}), \\ z_k = H_{\lambda_k} x_k + V_{\lambda_k} v_k, \end{cases} \tag{21}$$

where $w_k \sim \mathcal{N}(0, \Sigma_w)$, $v_k \sim \mathcal{N}(0, \Sigma_v)$ are Gaussian distributed processes. The term *conditional* justifies the characteristic of these models: they are linear and their formulation depends on extra random variables, called *latent*, denoted as λ. Then, the Monte Carlo sampling is working in the space of *latent variables* instead of in the space of the state variables. The matrices F_λ and H_λ are known, assuming that λ is known. For simplicity, in the sequel we are omitting the subscript $_\lambda$ from the matrices of (21).

Given the indicator variable, the KF provides a sufficient statistical characterization of the system dynamics. The MKF relies on the conditional Gaussian property and uses a marginalization operation in order to improve the efficiency of the sequential Monte Carlo estimation technique.

In our JTC problem the indicator variable λ (corresponding to m from the previous sections) takes values from a finite discrete set $\mathbb{S} \triangleq \{1,\ 2,\ \ldots,\ s(c)\}$ and evolves according to a Markov chain with transition probabilities (3).

Let $KF_k^{(j)} = \{\mu_k(\lambda_{1:k}^{(j)}), \Sigma(\lambda_{1:k}^{(j)})\}$ denote the sufficient statistics that characterize the posterior mean and covariance matrix of the state x_k, conditional on the observations $z_{1:k}$ accumulated up to the time instant k, and indicator variable $\lambda_{1:k}^{(j)}$.

The MKF algorithm [15] which we developed for JTC has the following form:

1. Initialization, $k = 1$
   For class $c = 1, 2, \ldots, M$ set $P(c) = P_1(c)$
   * For $j = 1, \ldots, N_c$,
   sample $\lambda_1^{(j)} \sim \{P_1^c(\lambda)\}_{\lambda=1}^{s(c)}$
   and set $KF_1^{(j)} = \{\mu_1(\lambda_1^{(j)}), \Sigma(\lambda_1^{(j)})\}$,
   where $\mu_1(\lambda_1^{(j)}) = \hat{\mu}_1$ and $\Sigma(\lambda_1^{(j)}) = \Sigma_1$ are the mean and covariance of the initial state
   $x_1 \sim N(\hat{\mu}_1, \Sigma_1)$. Set $k = 2$.
   End for c

2. For class $c = 1, 2, \ldots, M$ execute
   For $j = 1, \ldots, N_c$,
   * For λ_{k-1}^i, $i = 1, \ldots, s(c)$ $(\lambda_{k-1}^i \triangleq \lambda_{k-1} = i)$

   - run a KF time update step
     $(\mu_{k|k-1}^{(j)})^i = F\mu_{k-1|k-1}^{(j)} + Gu_{k-1}(\lambda_{k-1}^i, c),$
     $(\Sigma_{k|k-1}^{(j)})^i = F\Sigma_{k-1|k-1}^{(j)}F^T + G\Sigma_w(\lambda_{k-1}^i, c)G^T,$
     $(z_{k|k-1}^{(j)})^i = h((\mu_{k|k-1}^{(j)})^i),$
     $(S_k^{(j)})^i = (H_k^{(j)})^i(\Sigma_{k|k-1}^{(j)})^i(H_k^{(j)})^{iT} + V\Sigma_v V^T.$
   - on receipt of a measurement z_k calculate $vv_i^{(j)} = f(z_k|\lambda_k^i, KF_{k-1}^{(j)})p(\lambda_k^i|\lambda_{k-1}^{(j)})$, where $f(z_k|\lambda_k^i, KF_{k-1}^{(j)}) = N(z_k; (z_{k|k-1}^{(j)})^i, (S_k^{(j)})^i)$, and
     $p(\lambda_k^i|\lambda_{k-1}^{(j)})$ is the prior transition probability of the indicator.
   - end for λ_{k-1}^i

   * Sample $\lambda_k^{(j)}$ from a set $\mathbb{S}$ with probability, proportional to $vv_i^{(j)}, i = 1, \ldots, s(c)$.

   Let $KF_k^{(j)}$ be the one with $\lambda_k^{(j)} = l$.
   * complete the KF iteration

   $K_{k|k}^{(j)} = (\Sigma_{k|k-1}^{(j)})^l(H_k^{(j)})^{lT}[(S_k^{(j)})^l]^{-1},$
   $\mu_{k|k}^{(j)} = (\mu_{k|k-1}^{(j)})^l + K_{k|k}^{(j)}[z_k - (z_{k|k-1}^{(j)})^l],$
   $\Sigma_{k|k}^{(j)} = (\Sigma_{k|k-1}^{(j)})^l - K_{k|k}^{(j)}(S_k^{(j)})^l K_{k|k}^{(j)T},$

   * update the importance weights
   $W_k^{(j)} = W_{k-1}^{(j)} g_c\left(\hat{\mathrm{v}}_{k-1}^c\right) \sum_{i=1}^{s(c)} vv_i^{(j)}$

   end for j

   * Resampling in the same way as in the particle filter: generate a new set of samples with associated weights

   * Compute the updated state estimate and posterior class probabilities (as in the particle filter)

   End for c
   Set $k \longleftarrow k+1$ and go to step 2.

A reasonable choice of the proposal distribution $q(\lambda_{k+1}^{(j)}|\lambda_{1:k}^{(j)}, KF_k^{(j)})$ for the indicator variable is its predictive distribution $q(\lambda_{k+1}^{(j)}|\lambda_{1:k}^{(j)}, KF_k^{(j)}, z_{k+1})$ [14].

The designed here MKF is based on Extended KFs, obtained after linearizing the measurement equation (2).

5 Simulation results

The performance of the implemented filters for JTC is evaluated by simulations over a representative test trajectory given in Figure 2, together with the radar location, indicated by a triangle. The target motion is generated without process noise. The MM particle filter and the MKF accounting for speed and acceleration constraints are compared to filters without speed constraints, i.e. which likelihood is computed not such as in (20), but is equal to $\lambda_{\{x_k,c\}} = f_{x_k}(z_k)$.

Measures of performance. *Root-Mean Squared Errors* (RMSEs) [17]: on position (both coordinates combined) and speed (magnitude of the velocity vector), *average probability of correct class identification* and *average time per update* are used to evaluate the filters performance. The results presented below are based on 100 Monte Carlo runs. The cloud of the particles for each class is with size $N_c = 3000$ for the MM particle filter (PF) and $N_c = 300$ for the MKF, whereas the sampling period is $T = 5\ [s]$. The prior class probabilities are chosen as follows: $P_1(1) = P_1(2) = 0.5$. The parameters of the base state vector initial distribution $x_1 \sim N(x_1; m_1, P_1)$ in the particle filter algorithm are selected as follows: $P_1 = diag\{150^2\ [m],\ 20.0^2\ [m/s],\ 150^2\ [m],\ 20.0^2\ [m/s]\}$; m_1 contains the exact initial target parameters. The MKF initial parameters are: $\hat{\mu}_1$ the mean and the covariance Σ_1 of the initial state $x_1 \sim N(\hat{\mu}_1, \Sigma_1)$ are obtained by a two-point differencing technique [12] (p 253). Notice that the noise covariance matrices of the MKF coincide with those of the particle filter, namely $\Sigma_v = R$, $V = I$, $\Sigma_w = diag\{\sigma_{wx}^2, \sigma_{wy}^2\}$ with $\sigma_{wx} = \sigma_{wy}$, given in Sec. 3.1

Test trajectory. The target performs four turn maneuvers with intensity $1g$, $2g$, $5g$, $2g$. The speed is constant, equal to 260 $[m/s]$. After the $5g$ maneuver, the MM particle filter without speed constraints correctly identifies the real second class, but after the last maneuver of $2g$, a tendency for misclassification is present (Figure 5). The MM particle filter with speed constraints correctly determines the class (Figure 6). According to the results from the RMSEs (Figures 3, 4) the developed MM particle filter with acceleration and speed constraints can reliably track maneuvering targets.

Nevertheless, as evident from Figures 7 and 12, the filters clearly distinguish different motion segments and provide good estimates of the model probabilities.

It should be mentioned that the selected target model (15) in combination with the particle filtering technique or MKF provides an easy way of imposing acceleration constraints on the target dynamics. Air targets usually perform turn maneuvers with varying accelerations along x and y coordinates. These varying accelerations consecutively make active different models from the designed multiple model configuration, since the models have fixed x- and y- acceleration inputs. During maneuvering different models may have similar probabilities which makes difficult to infer which is the most probable between them.

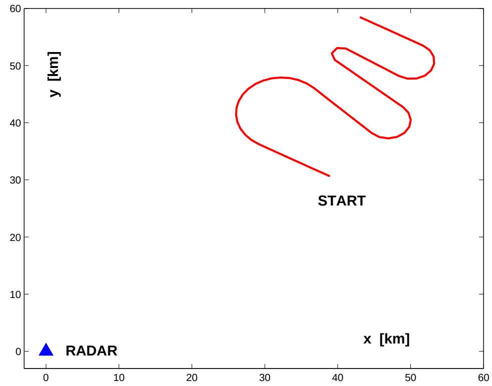

Fig. 2: Test scenario

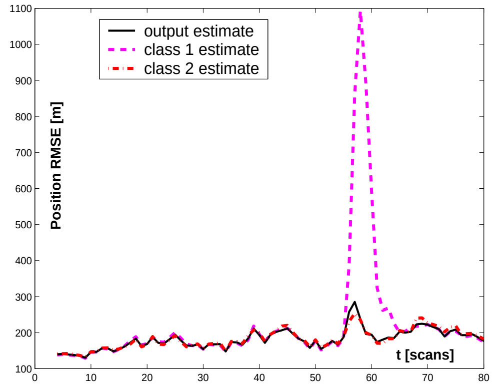

Fig. 3: PF position RMSE [m]

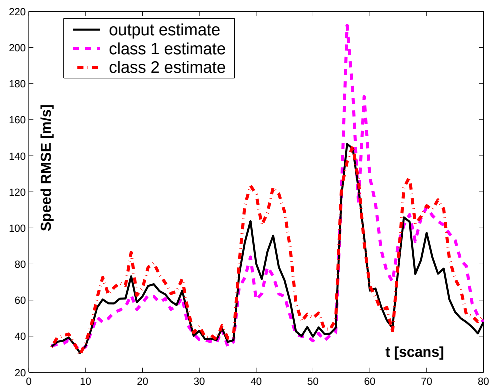

Fig. 4: PF speed RMSE [m/s]

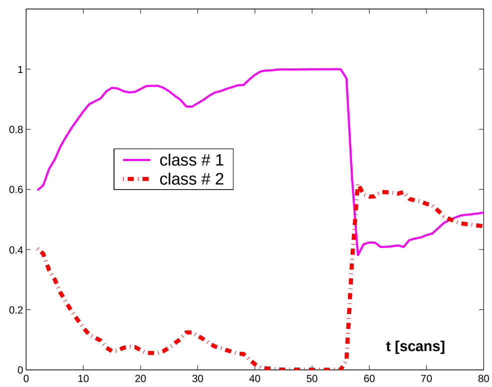

Fig. 5: PF class probabilities (without speed constraints)

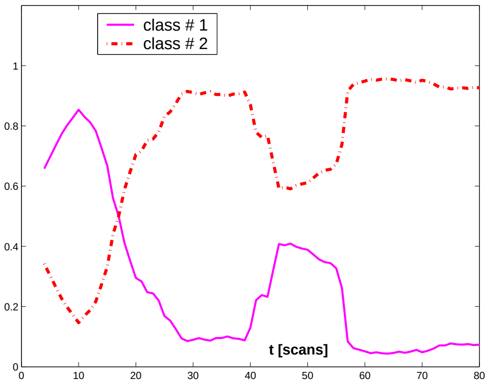

Fig. 6: PF class probabilities (speed constraints)

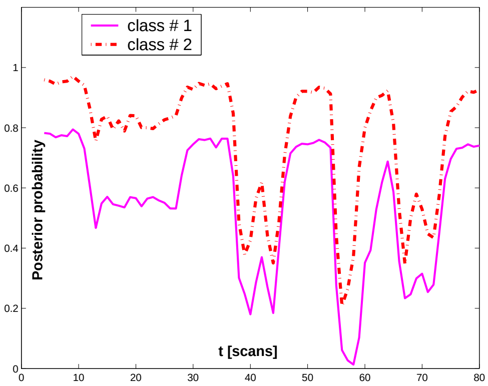

Fig. 7: PF posterior mode (m=1) probabilities

Figures 8-11 illustrate the performance for the MKF. An important advantage of the MKF compared to the MM particle filter is the smaller peak-dynamic errors during intensive maneuvers (with an acceleration $5\ g$ in the test).

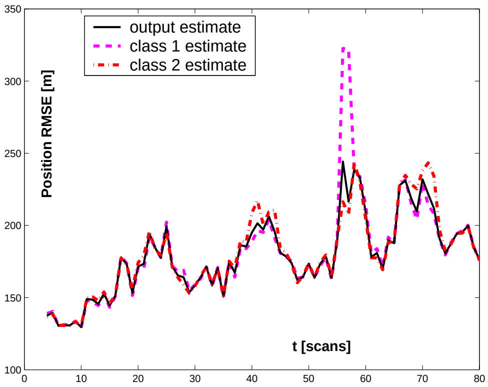

Fig. 8: MKF position RMSE [m]

The speed-based class probabilities $g_c(\hat{v}^c)$, $c = 1, 2$, obtained by the MM particle filter and MKF are quite similar. For these reasons we present only the estimated by the MKF functions in Figure 13. The target speed of $260[m/s]$ provides a slight superiority of the probability, that the target belongs to class 2, according to the speed constraints. The estimated speed probabilities assist in the proper class identification, as we can seen in Figure 11.

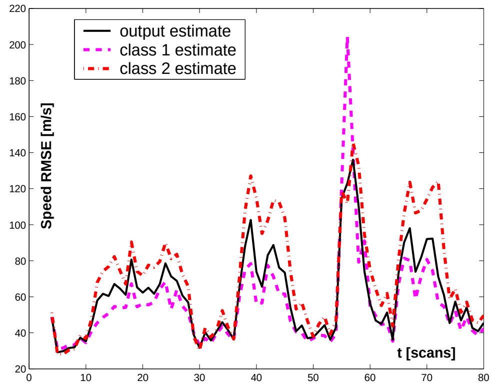

Fig. 9: MKF speed RMSE [m/s]

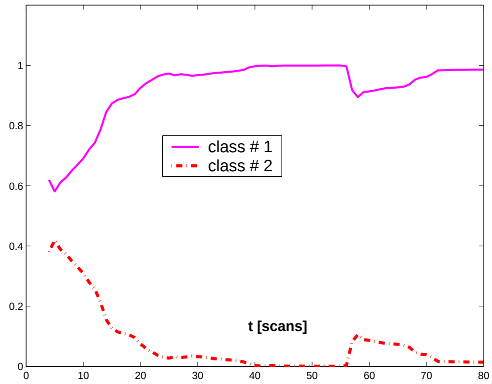

Fig. 10: MKF class probabilities calculated without taking into account speed constraints

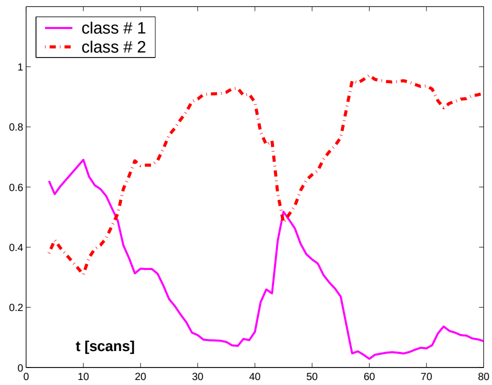

Fig. 11: MKF class probabilities calculated using both speed and acceleration constraints

We have to notice that the MM particle filter and MKF computational complexity allow for an on-line implementation. An advantage of the MKF is its reduced complexity compared to the MM particle filter. The computational time of the PF (with $N_c = 3000$ samples) versus the respective one of the MKF (with $N_c = 300$) is 1.73. We obtained very good results for the MKF with $N_c = 200$ as well. In this case the ratio PF computation time/ MKF computation time becomes 2.7. All experiments were performed on PC computer with AMD Athlon processor 2 GHz. Both algorithms permit parallelization at least of some parts: the MM filters corresponding to each class can be definitely run in parallel.

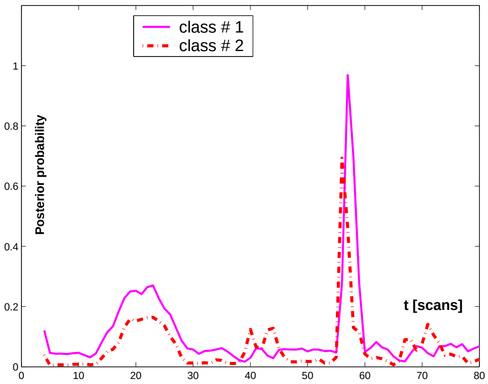

Fig. 12: MKF posterior mode (m=3) probabilities

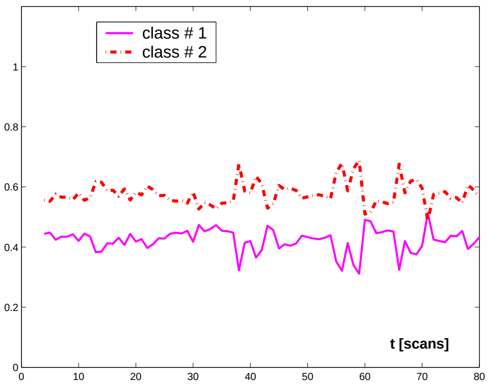

Fig. 13: MKF speed-based class probabilities

6 Conclusions

A Bayesian joint tracking and classification technique has been recently proposed in [2]. It offers the designer a possibility of selecting different state spaces and different filtering procedures, suitable for each target type. Motivated by this approach, we have designed a multiple model particle filter and a mixture Kalman filter for the purposes of joint *maneuvering* target tracking and classification and evaluated their performance. We have shown that distinct constraints, enforced by the changeable target behavior can be easily incorporated into the Monte Carlo framework. Two air target classes are considered: *commercial* and *military* aircraft. The classification task is accomplished by processing kinematic information only, no class (feature) measurements are used. For that purpose a bank of two *multiple model* class-dependent particle filters is designed and implemented in the presence of speed and acceleration constraints. The acceleration constraints for each class are imposed by using different control inputs in the target model. The speed constraints are enforced by constructing class-dependent speed likelihood functions. Speed likelihoods are calculated at each filtering step and assist in the process of classification. It was shown that speed and acceleration constraints can be accounted for in a similar way in a MKF.

The filters performance is analyzed by simulation over typical target trajectory in a plane. The results show a reliable tracking and correct target type classification. A generalization of the algorithms' application to the three-dimensional case is straightforward.

Acknowledgements. This research is supported by the Bulgarian Foundation for Scientific Investigations under grants I-1202/02 and I-1205/02 and in part by the UK MOD Data and Information Fusion Defence Technology Center.

References

[1] Henry Leung and Jiangfeng Wu. Bayesian and Dempster-Shafer target identification for radar surveillance. *IEEE Trans. Aerospace and Electr. Systems*, 36(2):432–447, 2000.

[2] Neil Gordon, Simon Maskell, and Thia Kirubarajan. Efficient particle filters for joint tracking and classification. In *Proc. SPIE Signal and Data Processing of Small Targets*, volume 4728, Orlando, USA, April 1–5 2002. SPIE.

[3] Alfonso Farina, Pierfrancesco Lombardo, and M. Marsella. Joint tracking and identification algorithms for multisensor data. *IEE Proc.-Radar Sonar, Navig.*, 149(6):271–280, 2002.

[4] Subhash Challa and Graham Pulford. Joint target tracking and classification using radar and ESM sensors. *IEEE Trans. on Aerosp. and Electr. Systems*, 37(3):1039–1055, 2001.

[5] David Salmond, David Fisher, and Neil Gordon. Tracking and identification for closely spaced objects in clutter. In *Proc. of the Europ. Contr. Conf.*, Brussels, Belgium, 1997.

[6] Shawn Herman and Pierre Moulin. A particle filtering appropach to FM-band passive radar tracking and automatic target recognition. In IEEE, editor, *Proc. of the IEEE Aerospace Conf.*, Big Sky, Montana, March 2002. IEEE.

[7] Mahendra Malick, Simon Maskell, Thia Kirubarajan, and Neil Gordon. Littoral tracking using particle filter. In *Proc. of the Fifth Int. Conf. Information Fusion*, USA, July 2002.

[8] Subhash Challa and Niclas Bergman. Target tracking incorporating flight envelope information. In *Proc. Third Int. Conf. Information Fusion*, Paris, France, July 2000. ISIF.

[9] Albena Tchamova, Tzvetan Semerdjiev, and Jean Dezert. Estimation of target behaviour tendencies using Dezert-Smarandache theory. In *Proc. of the Sixth Intl. Conf. Inform. Fusion*, pages 1349–1356, Australia, July 2003. ISIF.

[10] Arnaud Doucet, Neil Gordon, and Vikram Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Trans. on Signal Proc.*, 49(3):613–624, 2001.

[11] Arnaud Doucet, Nando de Freitas, and Neil Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, USA, 2001.

[12] Yaakov Bar-Shalom and Xiao-Rong Li. *Estimation and Tracking: Principles, Techniques, and Software*. Artech House, Norwood, MA, 1993.

[13] Amir Averbach, Samuel Itzikowitz, and Tal Kapon. Radar target tracking – Viterbi versus IMM. *IEEE Trans. on Aerospace and Electronic Systems*, 27(3):550–563, 1991.

[14] Jun S. Liu and Rong Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

[15] Rong Chen and Jun S. Liu. Mixture Kalman filters. *Journal of the Royal Statistical Society B*, (62):493–508, 2000.

[16] Zhe Chen. Bayesian filtering: From Kalman filters to particle filters, and beyond. *Adaptive Syst. Lab., McMaster Univ., Hamilton, ON, Canada. [Online], http://soma.crl.mcmaster.ca/˜zhechen/homepage.htm*, 2003.

[17] Yaakov Bar-Shalom and Xiao-Rong Li. *Multitarget-Multisensor Tracking: Principles and Techniques*. YBS Publishing, Storrs, CT, 1995.