

## Research Article

# Iterative Mixture Component Pruning Algorithm for Gaussian Mixture PHD Filter

**Xiaoxi Yan**

*School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, China*

Correspondence should be addressed to Xiaoxi Yan; [yanxiaoxi1981@gmail.com](mailto:yanxiaoxi1981@gmail.com)

Received 23 April 2014; Accepted 30 June 2014; Published 15 July 2014

Academic Editor: Suiyang Khoo

Copyright © 2014 Xiaoxi Yan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As far as the increasing number of mixture components in the Gaussian mixture PHD filter is concerned, an iterative mixture component pruning algorithm is proposed. The pruning algorithm is based on maximizing the posterior probability density of the mixture weights. The entropy distribution of the mixture weights is adopted as the prior distribution of mixture component parameters. The iterative update formulations of the mixture weights are derived by Lagrange multiplier and Lambert  $W$  function. Mixture components, whose weights become negative during iterative procedure, are pruned by setting corresponding mixture weights to zeros. In addition, multiple mixture components with similar parameters describing the same PHD peak can be merged into one mixture component in the algorithm. Simulation results show that the proposed iterative mixture component pruning algorithm is superior to the typical pruning algorithm based on thresholds.

## 1. Introduction

The objective of multitarget tracking is to estimate target number and target states from a sequence of noisy and cluttered measurement sets. The tracked target is generally simplified as a point [1–3]. Most of the existing point target tracking algorithms are based on data association where the correspondence of measurements to targets has to be set up. The simplest data association algorithm is the nearest-neighbour algorithm in which the measurement closest in statistical distance to predicted state is used to update target state estimate. Probabilistic data association is another typical algorithm in which all the measurements close to the predicted state are used to update target state estimate [4]. Joint probabilistic data association is a generalization of probabilistic data association for multiple target tracking in which association probabilities of all the targets and measurements are described by confirmed matrices [5, 6]. Multitarget tracking algorithms based on data association are in individual view, where the problem of multitarget tracking is converted into the multiple problems of single target tracking. In the multitarget tracking, both the measurements and the estimations are gained in the set form. Thus, multitarget tracking is naturally a class of set-valued

estimation problems. The probability hypothesis density (PHD) filter derived by Mahler based on random finite sets statistics theory is an elegant and tractable approximate solution to the multitarget tracking problem [7, 8]. Another interpretation of the PHD in bin-occupancy view is presented in [9]. By now, there have been two implementations of PHD filter, Gaussian mixture implementation [10, 11] and sequential Monte Carlo implementation [12–16], which are suitable for linear Gaussian dynamics and nonlinear non-Gaussian dynamics. The convergence of Gaussian mixture implementation is discussed in [17] and the convergence of sequential Monte Carlo implementation in [15, 18, 19]. The cardinalized PHD (CPHD) filter propagating both the PHD and the distribution of target number is developed to improve the performance of the PHD filter [20]. Generally, the CPHD filter is computationally less tractable compared to the PHD filter. There have been the Gaussian mixture implementation of CPHD filter under multitarget linear Gaussian assumptions [21] and the sequential Monte Carlo implementation [22]. As promising and unified methodologies, the PHD and CPHD filters have been widely applied in many fields, such as maneuvering target tracking [23, 24], sonar tracking [25, 26], and visual tracking [27–29]. As the sensor resolution is greatly improved, target tracking should

be formulated as extended object tracking [30]. Extended object PHD filter is also derived by Mahler in [31]. There have been some implementations of extended object probability hypothesis density filter by now [32–36]. The convergence of the Gaussian mixture implementation of extended object probability hypothesis density filter is discussed in [37]. When the Gaussian mixture model is applied in set-valued multitarget tracking, the Gaussian mixture reduction is an important topic [10, 38]. The earlier work in Gaussian mixture reduction for target tracking has been done in [39, 40]. As the Gaussian mixture reduction is implemented, there are several criterions such as maximum similarity [41], Euclidean distance [42–44], and Kullback-Leibler divergence measure [45]. The concentrations of this paper are on the Gaussian mixture reduction of the Gaussian mixture implementation of PHD filter.

As far as the Gaussian mixture implementation of the PHD filter is concerned, it approximates the PHD by the summation of weighted Gaussian components under the multitarget linear Gaussian assumptions [10]. In the Gaussian mixture PHD filter, the PHD is presented by a large number of weighted Gaussian components that are propagated over time. The sum of the weights of Gaussian components is the expected target number since the integral of the PHD over the state space is the expected target number. The output of Gaussian mixture PHD filter is weighted Gaussian components. However, the Gaussian mixture PHD filter suffers from computation problems associated with the increasing number of Gaussian components as time progresses, since mixture component number increases both at prediction step and at update step. In fact, component number increases without bound. Thus, the Gaussian mixture PHD filter is infeasible without component pruning operation. The goal of this paper is to prune the Gaussian components to make the Gaussian mixture PHD filter feasible. An iterative mixture component pruning algorithm is proposed for the Gaussian mixture PHD filter. The pruning operation of mixture components is done by setting mixture weights to zeros during the iteration procedure.

The remaining parts of this paper are organized as follows. Section 2 describes the component increasing problem in Gaussian mixture PHD filter. The iterative mixture component pruning algorithm is derived in Section 3. Section 4 is devoted to the simulation study. Conclusion is provided in Section 5.

## 2. Problem Description

The predictor and connector of PHD filter [7, 8] are

$$v_{k|k-1}(x) = \int p_{S,k}(\zeta) f_{k|k-1}(x|\zeta) v_{k-1}(\zeta) d\zeta + \int \beta_{k|k-1}(x|\zeta) v_{k-1}(\zeta) d\zeta + \gamma_k(x), \quad (1)$$

$$v_k(x) = [1 - p_{D,k}(x)] v_{k|k-1}(x) + \sum_{z \in Z_k} \frac{\varphi_{z,k}(x) v_{k|k-1}(x)}{\kappa_k(z) + \int \varphi_{z,k}(\xi) v_{k|k-1}(\xi) d\xi}, \quad (2)$$

respectively, where  $v(\cdot)$  is the PHD,  $\gamma_k(x)$  is the birth PHD at time step  $k$ ,  $\beta_{k|k-1}(\cdot|\zeta)$  is the spawned PHD from  $\zeta$  at time step  $k-1$ ,  $\kappa_k(z)$  is the clutter PHD,  $p_{S,k}(\zeta)$  is the survival probability,  $p_{D,k}(x)$  is the detection probability,  $\varphi_{z,k}(x) = p_{D,k}(x) g_k(z|x)$ ,  $g_k(z|x)$  is the single target likelihood, and  $Z_k$  is the measurements at time step  $k$ .

Under the linear Gaussian assumptions, the Gaussian mixture PHD filter is derived in [10]. The main steps of the Gaussian mixture PHD filter are summarized as follows. If the PHD at time step  $k-1$  is in the form of Gaussian mixture

$$v_{k-1}(x) = \sum_{i=1}^{J_{k-1}} w_{k-1}^{(i)} \mathcal{N}(x; m_{k-1}^{(i)}, P_{k-1}^{(i)}), \quad (3)$$

where  $w$  is the mixture weight,  $\mathcal{N}(\cdot)$  is the Gaussian distribution,  $m$  is the mean,  $P$  is the covariance, and  $J$  is the component number, then the predicted PHD for time step  $k$  is given by

$$v_{k|k-1}(x) = v_{S,k|k-1}(x) + v_{\beta,k|k-1}(x) + \gamma_k(x), \quad (4)$$

where  $\gamma_k$  is the birth PHD

$$\gamma_k(x) = \sum_{i=1}^{J_{y,k}} w_{y,k}^{(i)} \mathcal{N}(x; m_{y,k}^{(i)}, P_{y,k}^{(i)}), \quad (5)$$

$v_{S,k|k-1}$  is the survival PHD

$$v_{S,k|k-1}(x) = p_{S,k} \sum_{j=1}^{J_{k-1}} w_{k-1}^{(j)} \mathcal{N}(x; m_{S,k|k-1}^{(j)}, P_{S,k|k-1}^{(j)}), \quad (6)$$

$m_{S,k|k-1}^{(j)}$  is the predicted mean of the Gaussian component

$$m_{S,k|k-1}^{(j)} = F_{k-1} m_{k-1}^{(j)}, \quad (7)$$

$P_{S,k|k-1}^{(j)}$  is the predicted covariance of the Gaussian component

$$P_{S,k|k-1}^{(j)} = Q_{k-1} + F_{k-1} P_{k-1}^{(j)} F_{k-1}^T, \quad (8)$$

$v_{\beta,k|k-1}$  is the spawned PHD

$$v_{\beta,k|k-1}(x) = \sum_{j=1}^{J_{k-1}} \sum_{l=1}^{J_{\beta,k}} w_{k-1}^{(j)} w_{\beta,k}^{(l)} \mathcal{N}(x; m_{\beta}^{(j,l)}, P_{\beta}^{(j,l)}), \quad (9)$$

$m_{\beta}^{(j,l)}$  is the spawned mean of the Gaussian component

$$m_{\beta}^{(j,l)} = F_{\beta,k-1}^{(l)} m_{k-1}^{(j)} + d_{\beta,k-1}^{(l)}, \quad (10)$$

and  $P_{\beta}^{(j,l)}$  is the spawned covariance of the Gaussian component

$$P_{\beta}^{(j,l)} = Q_{\beta,k-1}^{(l)} + F_{\beta,k-1}^{(l)} P_{\beta,k-1}^{(j)} (F_{\beta,k-1}^{(l)})^T. \quad (11)$$

If formula (4) is rewritten in the simple form of the Gaussian mixture

$$v_{k|k-1}(x) = \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^{(i)} \mathcal{N}(x; m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}), \quad (12)$$

then the posterior PHD at time step  $k$  is

$$v_k(x) = (1 - p_{D,k}) v_{k|k-1}(x) + \sum_{z \in Z_k} v_{D,k}(x; z), \quad (13)$$

where  $v_{D,k}$  is the detected PHD

$$v_{D,k}(x; z) = \sum_{j=1}^{J_{k|k-1}} w_k^{(j)}(z) \mathcal{N}(x; m_{k|k}^{(j)}(z), P_{k|k}^{(j)}), \quad (14)$$

$w_k^{(j)}$  is the updated weight

$$w_k^{(j)}(z) = \frac{p_{D,k} w_{k|k-1}^{(j)} q_k^{(j)}(z)}{\kappa_k(z) + p_{D,k} \sum_{l=1}^{J_{k|k-1}} w_{k|k-1}^{(l)} q_k^{(l)}(z)}, \quad (15)$$

$m_{k|k}^{(j)}(z)$  is the updated mean

$$m_{k|k}^{(j)}(z) = m_{k|k-1}^{(j)} + K_k^{(j)}(z - H_k m_{k|k-1}^{(j)}), \quad (16)$$

$P_{k|k}^{(j)}$  is the updated covariance

$$P_{k|k}^{(j)} = [I - K_k^{(j)} H_k] P_{k|k-1}^{(j)}, \quad (17)$$

and  $K_k^{(j)}$  is the gain

$$K_k^{(j)} = P_{k|k-1}^{(j)} H_k^T (H_k P_{k|k-1}^{(j)} H_k^T + R_k)^{-1}. \quad (18)$$

It can be seen from formula (4) that component number increases from  $J_{k-1}$  to  $J_{k|k-1}$  by  $J_{k-1} \cdot J_{\beta,k} + J_{\gamma,k}$  at the prediction step. It is obvious in formula (13) that component number increases from  $J_{k|k-1}$  to  $J_k$  by  $J_{k|k-1} \cdot |Z_k|$  at the update step. Hence, the number of Gaussian components  $J_k$  representing PHD  $v_k$  at time step  $k$  in Gaussian mixture PHD filter is

$$J_k = (J_{k-1} (1 + J_{\beta,k-1}) + J_{\gamma,k}) (1 + |Z_k|), \quad (19)$$

where  $J_{k-1}$  is the number of components of the PHD  $v_{k-1}$  at time step  $k-1$ . In formula (19), the component number increases in  $\mathcal{O}(J_{k-1} |Z_k|)$ . In particular, the component number mostly increases in  $(J_{k-1} (1 + J_{\beta,k}) + J_{\gamma,k}) |Z_k|$  at the update step. Indeed, the number of Gaussian components increases without bound so that the computation of the Gaussian mixture PHD filter is intractable after several time steps. Therefore, it is necessary to reduce the number of components to make the Gaussian mixture PHD filter feasible. The goal of this paper is to prune the Gaussian mixture components to reduce component number in Gaussian mixture PHD filter.

### 3. Iterative Pruning Algorithm

For simplicity, the time index  $k$  is neglected and let  $M = J_{k|k}$  represent component number.  $w_S$  is the sum of the weights of the Gaussian components:

$$w_S = \sum_{j=1}^M w_j. \quad (20)$$

In the iterative pruning algorithm, the weights of Gaussian components are normalized by  $\{w_1/w_S, \dots, w_M/w_S\}$  at first so that

$$\sum_{j=1}^M w_j = 1. \quad (21)$$

Let  $\theta_j = \{m^{(j)}, P^{(j)}\}$  represent the parameters of the  $j$ th Gaussian component, where  $m^{(j)}$  and  $P^{(j)}$  are the mean and covariance, respectively. Then, the whole parameter set of  $M$  Gaussian components is  $\theta = \{w_1, \dots, w_M, \theta_1, \dots, \theta_M\}$ .

The entropy distribution of the mixture weights is adopted as the prior of  $\theta$ :

$$p(\theta) \propto \exp(-H(w_1, \dots, w_M)), \quad (22)$$

where  $H(w_1, \dots, w_M) = -\sum_{j=1}^M w_j \log w_j$  is the entropy measure [46, 47]. The goal of this choice of prior distribution, which depends only on the mixture weights, is to reduce mixture components by the adjustment of mixture weights during the iteration procedure. If we define the log-likelihood of the measurements  $Z = \{z_1, \dots, z_n\}$  given the mixture parameters as

$$\log p(Z | \theta) = \sum_{i=1}^n \log \sum_{j=1}^M w_j g(z_i | \theta_j), \quad (23)$$

where  $g(z | \theta_j)$  is the single target likelihood in  $j$ th component, then the MAP estimate of  $\theta$  is

$$\hat{\theta} = \arg \max_{\theta} \{\log p(Z | \theta) + \log p(\theta)\}. \quad (24)$$

For the mixture weight  $w_j$ , the MAP estimate can be computed by setting the derivative of the log-posterior to zero:

$$\frac{\partial}{\partial w_j} (\log p(Z | \theta) + \log p(\theta)) = 0. \quad (25)$$

The MAP estimate of  $w_j$  is computed by maximizing  $\log p(Z | \theta) + \log p(\theta)$  under the constraint (21):

$$\frac{\partial}{\partial w_j} \left( \log p(Z | \theta) + \log p(\theta) + \lambda \left( \sum_{j=1}^M w_j - 1 \right) \right) = 0, \quad (26)$$

where  $\lambda$  is Lagrange multiplier. Substituting formulas (22) and (23) into formula (26) gives

$$\frac{\sum_{i=1}^n w_j(z_i)}{w_j} + \log w_j + \lambda + 1 = 0, \quad (27)$$

where  $\omega_j(z)$  represents the membership that  $z$  is from the  $j$ th mixture component:

$$\omega_j(z) = \frac{w_j g(z | \theta_j)}{\sum_{l=1}^M w_l g(z | \theta_l)}. \quad (28)$$

Formula (27) is a simultaneous transcendental equation. We solve it for the  $w_j$  using the Lambert  $W$  function [48], an inverse mapping satisfying  $W(y)e^{W(y)} = y$ , and therefore  $\log W(y) + W(y) = \log y$ . The Lambert  $W$  function of complex  $y$  is defined as  $W(y)$ , which is a set of functions. The complex  $y$  can be computed by the equation  $W(y)e^{W(y)} = y$ , where  $e^{W(y)}$  is the exponential function. Lambert  $W$  function  $W(y)$  is a multivalued function defined in general for  $y$  complex and assumed  $W(y)$  complex. If  $y$  is real and  $y < -1/e$ , then  $W(y)$  is multivalued complex. If  $y$  is real and  $-1/e \leq y < 0$ ,  $W(y)$  has two possible real values. If  $y$  is real and  $y > 0$ ,  $W(y)$  has one real value. Then, for the Lambert  $W$  function  $W(y)$ ,

$$-W(y) - \log W(y) + \log y = 0. \quad (29)$$

Setting  $y = e^x$ , formula (29) can be rewritten as

$$-W(e^x) - \log W(e^x) + x = 0. \quad (30)$$

In formula (27), it is assumed that

$$\omega_j = \sum_{i=1}^n \omega_j(z_i). \quad (31)$$

Consequently, formula (30) is

$$\frac{\omega_j}{-\omega_j/W(e^x)} + \log\left(\frac{-\omega_j}{W(e^x)}\right) + x - \log(-\omega_j) = 0. \quad (32)$$

Comparing the Lambert  $W$  function (32) to formula (27), (32) can be reduced to (27) by setting  $x = 1 + \lambda + \log(-\omega_j)$ :

$$\frac{\omega_j}{-\omega_j/W(e^x)} + \log\left(\frac{-\omega_j}{W(e^x)}\right) + 1 + \lambda = 0. \quad (33)$$

Consequently,

$$w_j = \frac{-\omega_j}{W(e^{1+\lambda+\log(-\omega_j)})} = \frac{-\omega_j}{W(-\omega_j e^{1+\lambda})}. \quad (34)$$

Formula (27) and formula (34) constitute an iterative procedure for the MAP estimates of  $\{w_1, \dots, w_M\}$ : (1) given  $\lambda$ ,  $\{w_1, \dots, w_M\}$  are calculated by formula (34); (2)  $\{w_1, \dots, w_M\}$  are normalized; (3) given normalized  $\{w_1, \dots, w_M\}$ ,  $\lambda$  is computed by formula (27). The iteration procedure stops when the difference rate of log-posterior is smaller than the given threshold.

At the normalization step of the iteration procedure, if a mixture weight becomes negative, the corresponding component is removed from the mixture components by setting its weight to zero. The removed mixture component will not be considered when the log-posterior is computed

in the following iterations. The mixing weights of survival mixture components are normalized at the end of this step.

The effect of entropy distribution of mixing weights is taken during the iterative procedure. The mixture weights of components negligible to the PHD become smaller and smaller iteration by iteration, since the parameter estimates are driven into low-entropy direction by entropy distribution. The low-entropy tendency can also promote competition among the mixture components with similar parameters which can then be merged into one mixture component with larger weight.

For the mean  $m^{(j)}$  and covariance  $P^{(j)}$  of mixture component with nonzero weight  $w_j$ , they are updated by

$$m^{(j)} = (\omega_j)^{-1} \sum_{i=1}^n z_i \omega_j(z_i), \quad (35)$$

$$P^{(j)} = (\omega_j)^{-1} \sum_{i=1}^n (z_i - m^{(j)})(z_i - m^{(j)})^T \omega_j(z_i). \quad (36)$$

The main steps of iterative mixture component pruning algorithm are summarized in Algorithm 1.

#### 4. Simulation Study

A two-dimensional scenario with unknown and time-varying target number is considered to test the proposed iterative mixture component pruning algorithm. The surveillance region is  $[-1000, 1000] \times [-1000, 1000]$  (in meter). The target state consists of position and velocity, while target measurement is the position. Each target moves according to the following dynamics:

$$x_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix}, \quad (37)$$

where  $x_k = [x_{1,k}, x_{2,k}, x_{3,k}, x_{4,k}]^T$  is the target state,  $[x_{1,k}, x_{2,k}]^T$  is the target position, and  $[x_{3,k}, x_{4,k}]^T$  is the target velocity at time step  $k$ . The process noises are a zero-mean Gaussian white noise with standard deviations  $\sigma_{v_1} = \sigma_{v_2} = 5$  (m/s<sup>2</sup>). The survival probability is  $p_{S,k} = 0.99$ . The number of targets is unknown and variable over all scans. New targets appear spontaneously according to a Poisson point process with PHD function  $\gamma_k = 0.2\mathcal{N}(\cdot; \bar{x}, Q)$ , where

$$\bar{x} = \begin{bmatrix} -400 \\ -400 \\ 0 \\ 0 \end{bmatrix}, \quad Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 25 & 0 \\ 0 & 0 & 0 & 25 \end{bmatrix}. \quad (38)$$

$\mathcal{N}(\cdot; \bar{x}, Q)$  is the Gaussian component with mean  $\bar{x}$  and covariance  $Q$ . The spawned PHD is  $\beta_{k|k-1}(x | \zeta) = 0.05\mathcal{N}(x; \zeta, Q_\beta)$ , where  $Q_\beta = \text{diag}([100, 100, 400, 400])^T$ .

```

(1) normalize  $w_1, \dots, w_M$  by formula (20).
(2)  $t = 0$ .
(3)  $t = t + 1$ .
(4) for  $i = 1, \dots, n$  do
(5)   for  $j = 1, \dots, M$  do
(6)     compute  $\omega_j(z_i)$  by formula (28).
(7)   end for
(8) end for
(9) for  $j = 1, \dots, M$  do
(10)  compute  $\omega_j$  by formula (31).
(11) end for
(12) for  $j = 1, \dots, M$  do
(13)  compute  $w_j$  by formula (34).
(14) end for
(15) for  $j = 1, \dots, M$  do
(16)  if  $w_j < 0$  do
(17)    for  $l = j, \dots, M - 1$  do
(18)       $w_l = w_{l+1}$ .
(19)       $m^{(l)} = m^{(l+1)}$ .
(20)       $P^{(l)} = P^{(l+1)}$ .
(21)    end for
(22)     $j = j - 1$ .
(23)     $M = M - 1$ .
(24)  else do
(25)    compute  $m^{(j)}$  by formula (35).
(26)    compute  $P^{(j)}$  by formula (36).
(27)  end if
(28) end for
(29) normalize  $w_1, \dots, w_M$ ;
(30) compute  $\lambda$  by formula (27);
(31) if  $\log p(\theta(t) | Z) - \log p(\theta(t-1) | Z) > \varepsilon \cdot \log p(\theta(t-1) | Z)$  do
(32)  goto step 3;
(33) end if.

```

ALGORITHM 1: Iterative pruning algorithm.

Each target is detected with probability  $p_{D,k} = 0.98$ . The target-originated measurement model is

$$y_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x_k + \begin{bmatrix} w_{1,k} \\ w_{2,k} \end{bmatrix}, \quad (39)$$

where the measurement noise is a zero-mean Gaussian white noise with standard deviation  $\sigma_{w_1} = \sigma_{w_2} = 10$  (m). Clutter is modelled as a Poisson random finite set with intensity

$$\kappa_k(z_k) = \lambda_c \cdot c_k(z_k), \quad (40)$$

where  $\lambda_c$  is the average number of clutter measurements per scan and  $c(z)$  is the probability distribution over surveillance region. Here  $c(z)$  is a uniform distribution and  $\lambda_c$  is assumed to be 50.

The means of the Gaussian mixture components with mixing weights greater than 0.5 are chosen as the estimates of multitarget states after the mixture reduction.

The tracking results in one Monte Carlo trial are presented in Figures 1 and 2. It can be seen from Figures 1 and 2 that the Gaussian mixture PHD filter with the proposed iterative mixture component pruning algorithm is able to

detect the spontaneous and spawned targets and estimate the multiple target states.

The mixture components with weights larger than 0.0005 at the 86th time step before pruning operation in the above Monte Carlo simulation trial are presented in Figure 3. The mixture components with weights larger than 0.01 after pruning operation are presented in Figure 4. It is obvious that the mixture components with similar parameters describing the same PHD peak can be merged into one mixture component.

The typical mixture component pruning algorithm based on thresholds in [10] is adopted as the comparison algorithm. The thresholds in typical mixture component pruning algorithm are weight pruning threshold  $10^{-5}$ , mixture component merging threshold 4, and maximum allowable mixture component number 100. We evaluate the tracking performance of proposed algorithm against the typical algorithm by Wasserstein distance [49]. The Wasserstein distance is defined as

$$d_p(\widehat{X}, X) = \min_C \sqrt[p]{\sum_{i=1}^{|\widehat{X}|} \sum_{j=1}^{|X|} C^{ij} \|\widehat{x}^i - x^j\|^p}, \quad (41)$$

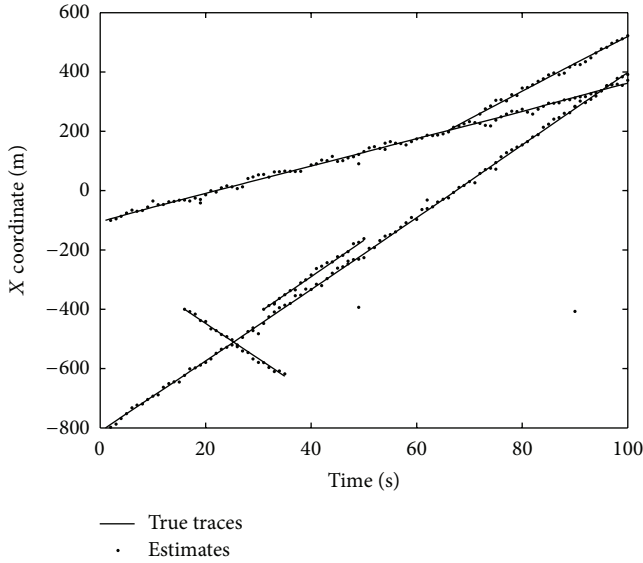


FIGURE 1: True traces and estimates of X coordinates.

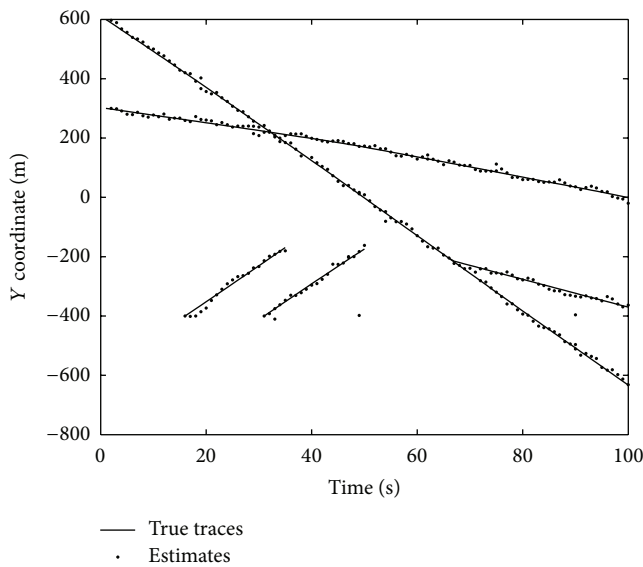


FIGURE 2: True traces and estimates of Y coordinate.

where  $\widehat{X}$  is the estimate of multitarget state set and  $X$  is the true multitarget state set. The minimum is taken over the set of all transportation matrices  $C$  (a transportation matrix  $C$  is one whose entries  $C^{ij}$  satisfy  $C^{ij} \geq 0$ ,  $\sum_{j=1}^{|X|} C^{ij} = 1/|\widehat{X}|$ , and  $\sum_{i=1}^{|\widehat{X}|} C^{ij} = 1/|X|$ ). This distance is not defined if either  $X$  or  $\widehat{X}$  is not defined. Figure 5 shows the mean Wasserstein distances of two algorithms over 100 simulation trials. Process noise, measurement noise, and clutter are independently generated at each trial. It can be seen from Figure 5 that the proposed iterative mixture component pruning algorithm is superior to the typical algorithm at most time steps. The proposed iterative mixture component pruning algorithm is worse than

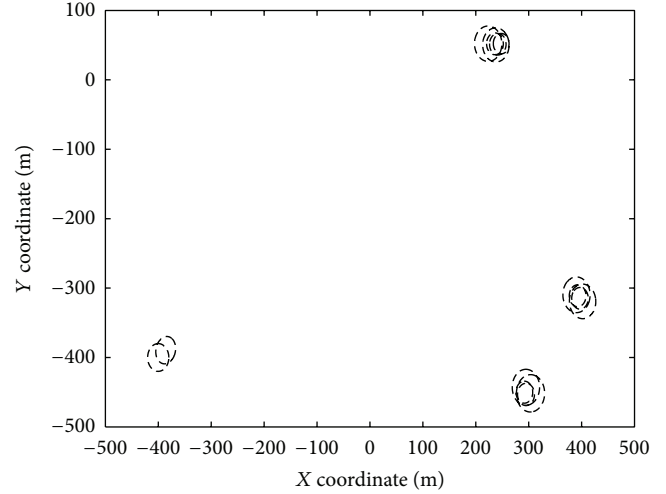


FIGURE 3: Components before pruning operation.

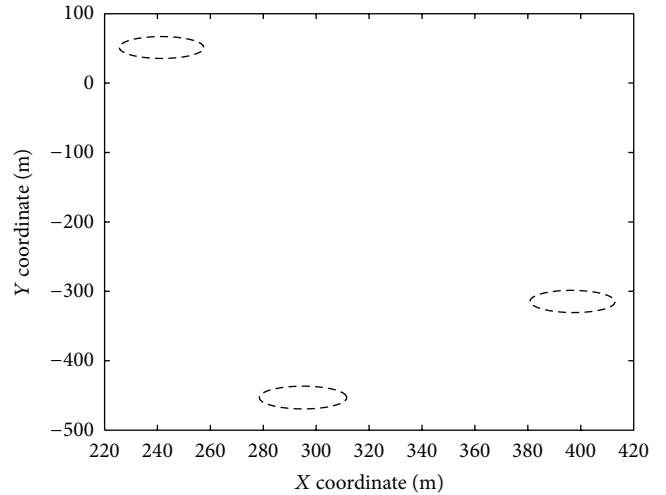


FIGURE 4: Components after pruning operation.

typical algorithm when spawned target is generated and two or more targets are close to each other. Two PHD peaks of two close targets may be regarded as one PHD peak in the proposed algorithm as a result of low-entropy tendency of entropy distribution. Then, some targets are not detected.

Figure 6 shows the estimates of target numbers of two algorithms. It is obvious that the estimates of target number of proposed algorithm are closer to the ground truth than typical algorithm at most time steps.

Figure 7 shows the mean component numbers of two algorithms after component pruning operations over 100 simulation trials. The component numbers of proposed algorithm are smaller than typical algorithm.

The case of low signal-to-noise rate (SNR) is yet considered for the further comparison of two algorithms.  $\lambda_c$  is assumed 80 in this low SNR case. The corresponding

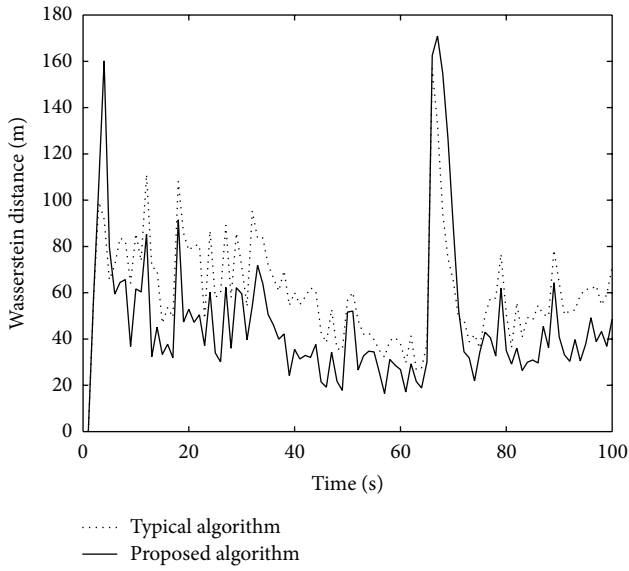


FIGURE 5: The averaged Wasserstein distances.

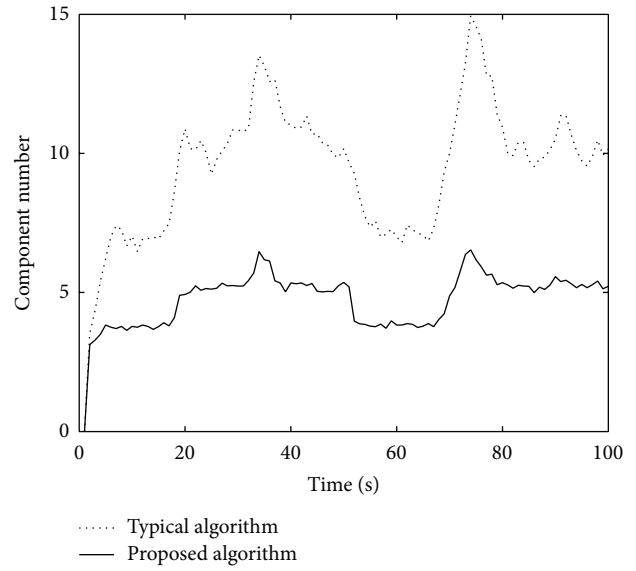


FIGURE 7: The averaged component numbers.

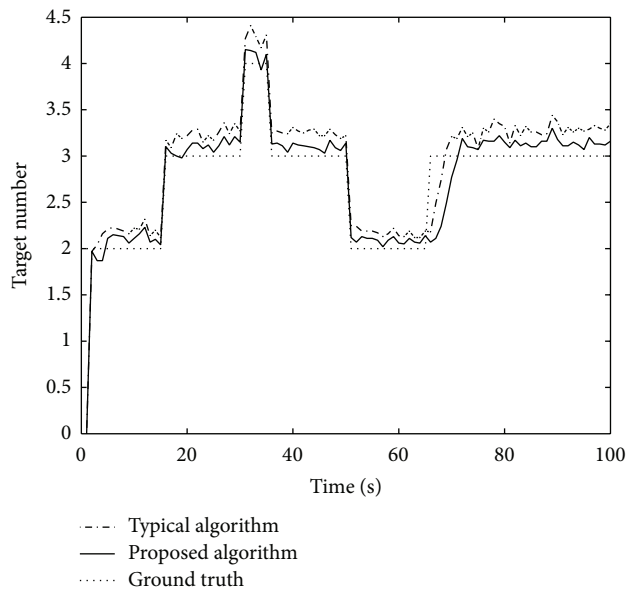


FIGURE 6: Estimates of target numbers.

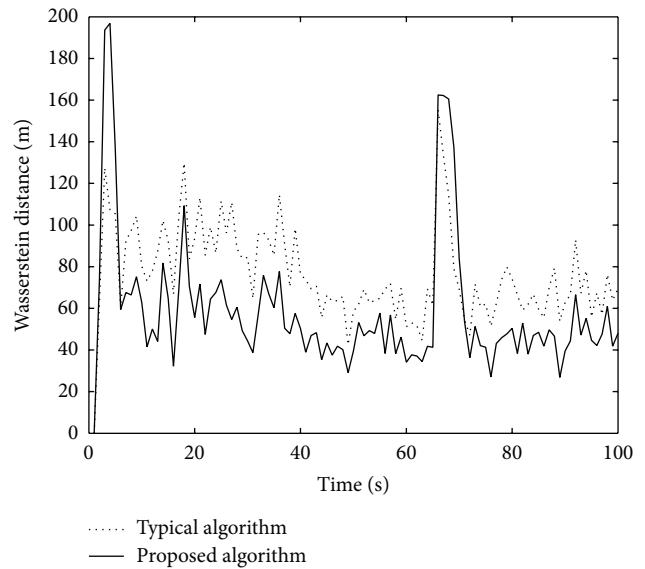


FIGURE 8: The averaged Wasserstein distances under low SNR.

Wasserstein distances, target number estimates, and component numbers are presented in Figures 8, 9, and 10. It can be seen that the proposed iterative mixture component pruning algorithm is also superior to the typical mixture component pruning algorithm based on thresholds in low SNR case.

### 5. Conclusion

An iterative mixture component pruning algorithm is proposed for the Gaussian mixture PHD filter. The entropy distribution of the mixture weights is used as the prior

distribution of mixture parameters. The update formula of the mixture weight is derived by Lagrange multiplier and Lambert  $W$  function. When the mixture weight becomes negative during the iteration procedure, the corresponding mixture component is pruned by setting the weight to zero. Simulation results show that the proposed iterative mixture component pruning algorithm is superior to the typical mixture component pruning algorithm based on thresholds at most time steps.

### Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

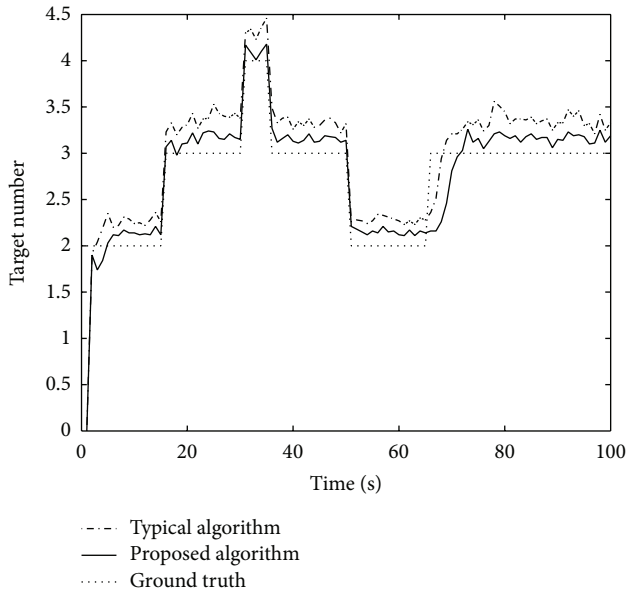


FIGURE 9: Estimates of target numbers under low SNR.

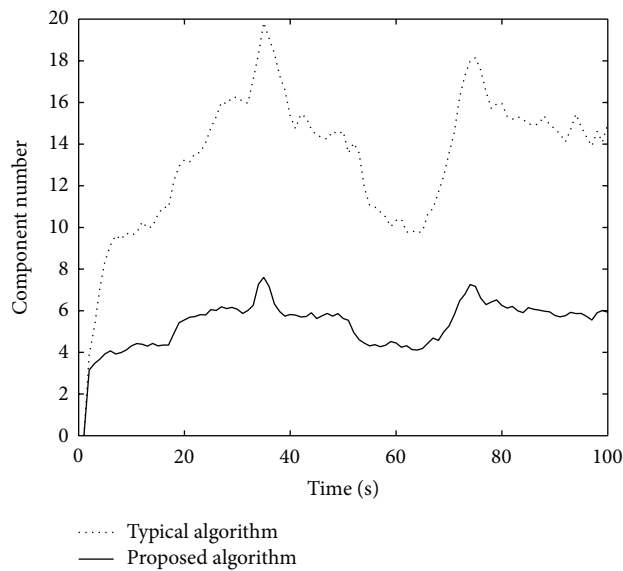


FIGURE 10: The averaged component numbers under low SNR.

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant no. 61304261), the Senior Professionals Scientific Research Foundation of Jiangsu University (Grant no. 12JDG076), and a Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

## References

[1] G. W. Pulford, "Taxonomy of multiple target tracking methods," *IEE Proceedings—Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 291–304, 2005.

- [2] S. Blackman and R. F. Popoli, *Design and Analysis of Modern Tracking System*, Artech House, Boston, Mass, USA, 1999.
- [3] Y. Barshalom and X. R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*, Yaakov Bar-Shalom, Storrs, Conn, USA, 1995.
- [4] Y. Barshalom, "Tracking in a cluttered environment probabilistic data association," *Automatica*, vol. 11, no. 5, pp. 451–460, 1975.
- [5] Y. Barshalom, "Tracking methods in a multitarget environment," *IEEE Transactions on Automatic Control*, vol. 23, no. 4, pp. 618–626, 1978.
- [6] K. C. Chang and Y. Barshalom, "Joint probabilistic data association for multitarget tracking with possibly unresolved measurements and maneuvers," *IEEE Transactions on Automatic Control*, vol. 29, no. 7, pp. 585–594, 1984.
- [7] R. P. S. Mahler, "Multi-target Bayes filtering via first-order multi-target moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.
- [8] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion. 1em Plus 0.5em Minus 0.4em*, Artech House, Norwood, Mass, USA, 2007.
- [9] O. Erdinc, P. Willett, and Y. Bar-Shalom, "The bin-occupancy filter and its connection to the PHD filters," *IEEE Transactions on Signal Processing*, vol. 57, no. 11, pp. 4232–4246, 2009.
- [10] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, 2006.
- [11] S.-A. Pasha, B.-N. Vo, and H.-D. Tuan, "A Gaussian mixture PHD filter for jump Markov system models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 3, pp. 919–936, 2009.
- [12] H. Sidenbladh, "Multi-target particle filtering for the probability hypothesis density," in *Proceedings of the 6th International Conference of Information Fusion*, vol. 2, pp. 800–806, Cairns, Australia, July 2003.
- [13] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo implementation of the PHD filter for multi-target tracking," in *Proceedings of the International Conference on Information Fusion*, pp. 792–799, Cairns, Australia, 2003.
- [14] T. Zajic and R. P. S. Mahler, "A particle-systems implementation of the PHD multi-target tracking filter," in *Signal Processing, Sensor Fusion and Target Recognition XI*, I. Kadar, Ed., vol. 5096 of *Proceedings of SPIE*, pp. 291–299, 2003.
- [15] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multi-target filtering with random finite sets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, 2005.
- [16] N. Whiteley, S. Singh, and S. Godsill, "Auxiliary particle implementation of probability hypothesis density filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 7, pp. 1437–1454, 2010.
- [17] D.-E. Clark and B.-N. Vo, "Convergence analysis of the Gaussian mixture PHD filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 4, pp. 1204–1212, 2007.
- [18] D.-E. Clark and J. Bell, "Convergence results for the particle PHD filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 7, pp. 2252–2261, 2006.
- [19] A. M. Johansen, S. S. Singh, A. Doucet, and B.-N. Vo, "Convergence of the SMC implementation of the PHD filter," *Methodology and Computing in Applied Probability*, vol. 8, no. 2, pp. 265–291, 2006.



- [20] R. P. S. Mahler, "PHD filters of higher order in target number," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1523–1543, 2007.
- [21] B. Vo, B. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, part 2, pp. 3553–3567, 2007.
- [22] B.-T. Vo, B.-N. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-Bernoulli filter and its implementations," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 409–423, 2009.
- [23] K. Punithakumar, T. Kirubarajan, and A. Sinha, "Multiple-model probability hypothesis density filter for tracking maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 87–98, 2008.
- [24] D. Dunne and T. Kirubarajan, "Multiple model multi-Bernoulli filters for manoeuvring targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2627–2692, 2013.
- [25] D.-E. Clark and J. Bell, "Bayesian multiple target tracking in forward scan sonar images using the PHD filter," *IEE Proceedings-Radar Sonar and Navigation*, vol. 152, no. 5, pp. 327–334, 2005.
- [26] D.-E. Clark, I.-T. Ruiz, Y. Petillot et al., "Particle PHD filter multiple target tracking in sonar image," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 1, pp. 409–416, 2007.
- [27] E. Maggio, M. Taj, and A. Cavallaro, "Efficient multitarget visual tracking using random finite sets," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1016–1027, 2008.
- [28] Y. D. Wang, J. K. Wu, A. A. Kassim et al., "Data-driven probability hypothesis density filter for visual tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1085–1095, 2008.
- [29] E. Maggio and A. Cavallaro, "Learning scene context for multiple object tracking," *IEEE Transactions on Image Processing*, vol. 18, no. 8, pp. 1873–1884, 2009.
- [30] K. Gilholm and D. Salmond, "Spatial distribution model for tracking extended objects," *IEE Proceedings Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 364–371, 2005.
- [31] R. P. S. Mahler, "PHD filters for nonstandard targets, I: extended targets," in *Proceedings of the 12th International Conference on Information Fusion, IEEE*, vol. 1–4, pp. 915–921, 2009.
- [32] K. Granstrom, C. Lundquist, and U. Orguner, "A Gaussian mixture PHD filter for extended target," in *Proceedings of the International Conference on Information Fusion*, Edinburgh, UK, 2010.
- [33] K. Granstrom, C. Lundquist, and U. Orguner, "Extended target tracking using a Gaussian-mixture PHD filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3268–3286, 2012.
- [34] K. Granstrom and U. Orguner, "A PHD filter for tracking multiple extended targets using random matrices," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5657–5671, 2012.
- [35] C. Lundquist, K. Granstrom, and U. Orguner, "An extended target CPHD filter and a gamma gaussian inverse wishart implementation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 472–483, 2013.
- [36] K. Granstrom and U. Orguner, "On spawning and combination of extended/group targets modeled with random matrices," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 678–692, 2013.
- [37] F. Lian, C. Han, J. Liu, and H. Chen, "Convergence results for the Gaussian mixture implementation of the extended-target PHD filter and its extended Kalman filtering approximation," *Journal of Applied Mathematics*, vol. 2012, Article ID 141727, 20 pages, 2012.
- [38] B.-N. Vo, B.-T. Vo, and R. P. S. Mahler, "Closed-form solutions to forward-backward smoothing," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 2–17, 2012.
- [39] D. J. Salmond, "Mixture reduction algorithms for target tracking in clutter," in *Signal and Data Processing of Small Targets Conference*, vol. 1305 of *Proceedings of SPIE*, pp. 434–445, 1990.
- [40] D. J. Salmond, "Mixture reduction algorithms for point and extended object tracking in clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 2, pp. 667–686, 2009.
- [41] J. E. Harmse, "Reduction of Gaussian mixture models by maximum similarity," *Journal of Nonparametric Statistics*, vol. 22, no. 5–6, pp. 703–709, 2010.
- [42] M. F. Huber and U. D. Hanebeck, "Progressive Gaussian mixture reduction," in *Proceedings of the 11th International Conference on information Fusion*, pp. 1–8, Cologne, Germany, 2008.
- [43] O. C. Schrempf, O. Feiermann, and U. D. Hanebeck, "Optimal mixture approximation of the product of mixtures," in *Proceedings of the 8th International Conference on Information Fusion*, pp. 85–92, Philadelphia, Pa, USA, 2005.
- [44] D. Schieferdecker and M. F. Huber, "Gaussian mixture reduction via clustering," in *Proceedings of the 12th International Conference on Information Fusion (FUSION '09)*, pp. 1536–1543, Seattle, Wash, USA, July 2009.
- [45] A. R. Runnalls, "Kullback-Leibler approach to Gaussian mixture reduction," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 989–999, 2007.
- [46] A. Caticha and R. Preuss, "Maximum entropy and Bayesian data analysis: entropic prior distributions," *Physical Review E*, vol. 70, no. 4, Article ID 046127, 12 pages, 2004.
- [47] M. Brand, "Structure learning in conditional probability models via an entropic prior and parameter extinction," *Neural Computation*, vol. 11, no. 5, pp. 1155–1182, 1999.
- [48] R. M. Corless, G. H. Gonnet, D. E. G. Hare, and D. E. Knuth, "On the Lambert W function," *Advances in Computational Mathematics*, vol. 5, no. 1, pp. 329–359, 1996.
- [49] J. Hoffman and R. P. S. Mahler, "Multitarget miss distance via optimal assignment," *IEEE Transactions on Systems, Man, and Cybernetics-Part A*, vol. 34, no. 3, pp. 327–336, 2004.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

