On Mixture Reduction for Multiple Target Tracking

Tohid Ardeshiri*, Umut Orguner†, Christian Lundquist*, Thomas B. Schön*
* Division of Automatic Control, Department of Electrical Engineering, Linköping University, 581 83 Linköping, Sweden
e-mail: {tohid, lundquist, schon}@isy.liu.se
† Department of Electrical and Electronics Engineering, Middle East Technical University, 06531 Ankara Turkey
email: umut@eee.metu.edu.tr

Abstract—**In multiple hypothesis or probability hypothesis based multiple target tracking the resulting mixtures with ever growing components should be approximated by a reduced mixture. Although there are cost based and more rigorous mixture reduction algorithms, which are computationally expensive to apply in practical situations especially in high dimensional state spaces, the mixture reduction is generally done based on ad hoc criteria and procedures. In this paper we propose a sequentially pairwise mixture reduction criterion and algorithm based on statistical decision theory. For this purpose, we choose the merging criterion for the mixture components based on a likelihood ratio test. The advantages and disadvantages of some of the previous reduction schemes and the newly proposed algorithm are discussed in detail. The results are evaluated on a Gaussian mixture implementation of the PHD filter where two different pruning and merging schemes are designed: one for computational feasibility, the other for the state extraction.**

Keywords: Mixture Reduction, Tracking, Gaussian Mixture, PHD Filter, Pruning, Merging.

I. Introduction

Multi-target tracking (MTT) using tracking algorithms such as Gaussian mixture probability hypothesis density filter (GM-PHD), multi-hypothesis tracking (MHT) and Gaussian sum filter results in an increasing number of components in the mixture representations of the targets over time. To be able to implement these algorithms for real time applications a merging and pruning step, where the Gaussian mixture is reduced to a computationally tractable mixture whenever needed, is necessary. The aim of the reduction algorithm is typically both to reduce the computational complexity within a predefined budget and let the components be well separated in the state space in order to easily be able to extract state estimated from them. The reduction is performed under the constraint that the information loss in the reduction step should be minimized.

The problem of reducing a mixture to another mixture with less components is addressed in several papers such as [1]–[11]. In [9] and [1] an ad hoc similarity measure is used for merging components of a mixture. In [2] and [11] covariance union and generalized covariance union are described, respectively. In [5] a Kullback-Leibler approach to Gaussian mixture reduction is proposed and is evaluated against other reduction methods in [3]. In [4] a Gaussian mixture reduction algorithm using homotopy to avoid local minima is suggested. A good summary of the Gaussian mixture reduction algorithms is given in [3]. A greedy approach to mixture reduction is to reduce the mixture to a mixture composed of less components via one to one comparison of the components. Such reduction approaches have three components; a metric for forming clusters, a threshold which is used in the metric and a merging algorithm. The metric and threshold are for first discarding some of the components and forming clusters of the remaining components to be merged in the merging algorithm. The merging algorithm merges two or more components into a single new component.

The reduction problem can be formulated as a nonlinear optimization problem where cost functions such as Kullback-Leibler divergence or integral squared error [6] are selected. The reduction problem using clustering techniques are also compared to optimization approaches in e.g., [3]. In these approaches the number of components in the reduced Gaussian mixture can be known in advance or not. These approaches can be quite expensive and not suitable for real time implementation.

In this paper, we investigate the merging and pruning algorithms in multiple target tracking and propose improvements on two different levels: high level and low level.

The current mixture reduction convention in MTT is to use exactly the same algorithm for reducing the computational load to a feasible level as for extracting the state estimates. In general, the mixture reduction for the state extraction should be much more aggressive than that for computational feasibility. For this reason, the number of components in the mixtures have to be reduced much more than what the computational resources actually allow for. This can result in coarser approximations than what is actually necessary. In this work, as our high level improvement, we propose to split the merging and pruning operations into two separate procedures according to:

- *Reduction in the loop* is a merge and pruning step which must be performed at each point in time for computational feasibility of the overall target tracking framework. The objective for this algorithm is to reduce the number of components and to minimize the information loss.
- *Reduction for extraction* aims at reducing the number of components so that the remaining components can be considered as state estimates in the target tracking framework.

This separation makes it possible to tailor these two algorithms to fulfill their individual objectives, which reduces the unnecessary approximations in the overall algorithm.

On a low level, a further contribution in this work is to cast the problem of mixture reduction in the framework of statistical decision theory providing a rigorous basis for evaluation of the reduction algorithms. Although there are cost based and more rigorous mixture reduction algorithms, which are computationally expensive to apply in practical situations especially in high dimensional state spaces, the mixture reduction is generally done based on ad hoc criteria and procedures. Although many greedy reduction algorithms can be proposed by combining the different merging algorithms and different merging statistics and thresholds, these choices should not be made independently. We suggest that not only the parameters of the mixture components, but also the parameters of the merged component should affect the merging decision as in [5]. Statistical decision theory and especially the Neyman-Pearson's theorem [12] provides a tool for evaluating and comparing these reduction algorithms and guides us towards the choice of optimal reduction algorithm based on the specifications of the application such as tolerable type I and type II error.

The rest of this paper is organized as follows. We will describe our first contribution regarding splitting the reduction algorithm in Section II and our second contribution regarding application of hypothesis testing to mixture reduction in Section III. In Section IV we adapt our contributions to reduction algorithms for Gaussian mixture PHD filter and perform numerical simulations. Concluding remarks can be found in Section V.

II. Splitting the Reduction Algorithm

A block diagram of the conventional mixture reduction method on a high level is shown in Figure 1. The proposed implementation of the reduction algorithm is split into two subroutines each of which is tailored for its own purpose, see Figure 2. The first reduction algorithm, denoted *reduction in the loop*, is designed to reduce the computational cost of the algorithm to the computational budget between the updates. In this reduction step the number of components should be reduced to a number that is tractable by the available computational budget and minimal loss of information is in focus. The second reduction algorithm, denoted *reduction for extraction*, is designed to reduce the mixture to as many components as there are targets. In this part of the algorithm application dependent specifications and heuristics can enter into the picture. If the purpose of state extraction is only visualization, the second reduction does not have to be performed at the same frequency as the measurements are received and can be made less frequent. The advantages of the proposed algorithm are that the unnecessary loss of information in the reduction in the loop step will only be due to the finite computational budget rather than closeness of the components. Furthermore, some computational cost can be discounted if the state extraction does not have to be performed for every measurement update step.

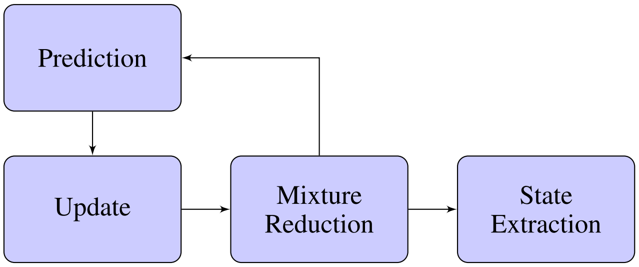


Figure 1. The standard flowchart of the MTT algorithms has only one mixture reduction block.

Another important advantage of the proposed algorithm is that the number of final components in both of the reduction algorithms is known since the computational budget is predefined in the reduction in the loop algorithm. Furthermore, the number of target states can be predetermined by summarizing the weights in e.g., a GM-PHD filter and utilized in the reduction for extraction algorithm. The clustering or optimization method selected for reduction can be executed more efficiently compared to a scenario where the number of components is left to be decided by the algorithm itself. In section IV-D we will show the impact of our proposition in a simulation.

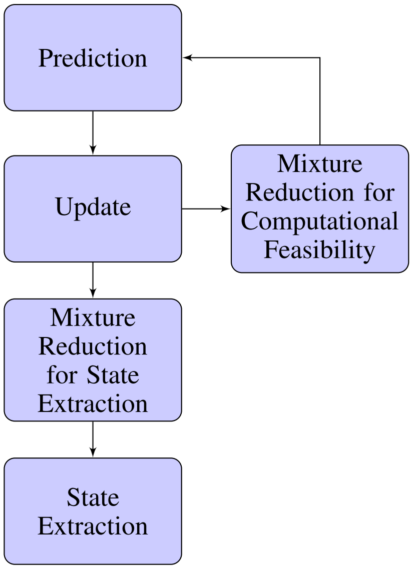

Prediction
Update
Mixture Reduction for Computational Feasibility
Mixture Reduction for State Extraction
State Extraction

Figure 2. The proposed block diagram of the MTT algorithm with two mixture reduction blocks; one tailored to keep the computational complexity within the computational budget and one tailored for state extraction.

III. Hypotheses Testing and its Application to Mixture Reduction

The basic problem in statistical decision theory is to make optimal choices from a set of alternatives, i.e., hypotheses, based on noisy observations. The intention is here to cast the problem of greedy mixture reduction in the framework of statistical decision theory so that the reduction method can move towards the optimal choice. The alternative decisions to be made in the reduction problem are whether to merge two components or not. However, there are no observation to base the decision on and there is only a small freedom in the formulation of the hypotheses. The remedy in other reduction algorithms has been to use all or some parameters of the components as observation and the present algorithm is not an exception.

In this section a short description of the generalized likelihood test (GLRT) is given in Section III-A, and in Section III-B it is shown how it can be applied to mixture reduction.

A. Generalized Likelihood Ratio Test

Assume that the data x has the Probability Density Function (PDF) $p(x;\theta_0,\mathcal{H}_0)$ under the null hypothesis $\mathcal{H}_0$ and the PDF $p(x;\theta_1,\mathcal{H}_1)$ under the alternative hypothesis $\mathcal{H}_1$. The statistics of the distributions are comprised in the parameters θ_0 and θ_1. The generalized likelihood ratio test [13] is given as follows. Decide $\mathcal{H}_1$ if

$$L_G(x) = \frac{p(x;\widehat{\theta}_1,\mathcal{H}_1)}{p(x;\widehat{\theta}_0,\mathcal{H}_0)} > \gamma, \tag{1}$$

where $\widehat{\theta}_i$ is the maximum likelihood estimate (MLE) of θ_i and γ is some suitably chosen threshold.

Two type of errors are important to keep in mind;

- type I error $p(\mathcal{H}_1|\mathcal{H}_0)$, i.e., decide $\mathcal{H}_1$ when $\mathcal{H}_0$ is true and
- type II error $p(\mathcal{H}_0|\mathcal{H}_1)$ i.e., decide $\mathcal{H}_0$ when $\mathcal{H}_1$ is true.

The errors can be calculated for each value of the threshold γ. It is possible to reduce one error by changing the threshold, but not both types of errors at the same time. The choice of value for the threshold γ depends on what probability of type I error that is tolerable i.e.

$$Pr(L_G(x) > \gamma; \mathcal{H}_0) = \alpha. \tag{2}$$

The tolerable probability of type I error α should be given as an input and the threshold γ can be calculated using equation (2). For further reading on this subject refer to [13].

B. Reduction Algorithm Based on GLRT

An initial mixture consisting of N components is given by

$$p(x) = \sum_{I=1}^{N} w^I q(x; \theta^I) \tag{3}$$

where I is an indicator variable with $p(I) = w^I$. Therefore the conditional density of x is according to $p(x|I) = q(x; \theta^I)$. In order to use the hypothesis testing we need observations and we will assume that the modes of the distribution $\{x^i\}_{i=1}^N$, where

$$\{x^i = \arg\max_x q(x; \theta^i)\}_{i=1}^N, \tag{4}$$

are given as observations. For a given merging algorithm $f(\cdot)$ and for a subset $L \subset \{i\}_{i=1}^N$ the two hypotheses are formulated as

$$\mathcal{H}_0 : p(x|I = i) = q(x; \theta^i) \qquad \forall i \in L \tag{5a}$$

$$\mathcal{H}_1 : p(x|I = i) = q(x; \theta^L) \qquad \forall i \in L \tag{5b}$$

where $\theta^L = f(\{w^i, \theta^i\}_{i \in L})$ and L is the subset of the components that we want to decide whether to merge or not. Here the null hypothesis corresponds to the case where the hypothetical random observations x^i are distributed independently with the densities $q(x; \theta^i)$, where θ^i are the parameters of the individual components. The alternative hypothesis on the other hand, corresponds to the case where all x^i are distributed (again independently) with the same density $q(x; \theta^L)$, where the merged parameter θ^L is a function of the individual component weights $\{w^i\}_{i \in L}$ and individual parameters $\{\theta^i\}_{i \in L}$. The function $f(\cdot)$ represents a merging function to obtain θ^L. Here, the fact that the alternative hypothesis is dependent on the specific merging algorithm $f(\cdot)$ is an attractive property that makes the merging statistics depend on the specific merging algorithm used to combine the components.

The merge statistics $L_G(x)$ and the GLRT are

$$L_G(\{x^i\}_{i \in L}) = \frac{\prod_{i \in L} p(x^i; \theta^L)}{\prod_{i \in L} p(x^i; \theta^i)} > \gamma \tag{6}$$

and the threshold γ is given by the tolerable type I error according to (2). It should be noted that by taking the expectation over the logarithm of the likelihood ratio, $\mathbb{E} \log L_G$, we can achieve the merging statistics given in [5] which is given as an upper bound on the Kullback-Leibler divergence of the original mixture and the resulting merged Gaussian component.

IV. Case Study on the Gaussian Mixture PHD Filter

The proposed ideas presented in the previous sections are applied to the Gaussian mixture PHD (GM-PHD) filter in this section. The conventional methods for reduction of the Gaussian mixtures in the GM-PHD filter is described in Section IV-A, some shortcomings with the standard method are shown and simple improvements are discussed in Section IV-B. The proposed hypothesis test based reduction algorithm is more robust compared to the conventional method and can be tuned considering the trade off between type I and type II error instead of ad hoc choice of threshold as shown in Section IV-C. Finally, in Section IV-D we will illustrate the impact of the proposed framework in a single target and a two targets scenario and show how the state estimate is improved using our propositions in the GM-PHD filter.

A. Conventional PHD Merging and Pruning Algorithms

The Gaussian mixture PHD filter is described in [14] and its merging and pruning algorithm follows the implementation in Figure 1. The pruning and merging block and the state extraction block presented in [14] are repeated here for convenience in Algorithm 1 and Algorithm 2, respectively. The two algorithms are widely used in implementations of the GM-PHD filter but are also used in other MTT filters where Gaussian mixture reduction and state estimate extraction is important such as interactive multiple models (IMM), see e.g., [15].

In the introduction it was mentioned that a greedy approach has three components; a metric, a threshold and a merging algorithm. The threshold is a design variable in the algorithms.

The expression in (9) compares the Mahalanobis distance metric of the "observation" m_k^j to the Gaussian density $\mathcal{N}(m_k^i, P_k^i)$. There are other measures of similarity between densities such as Bhattacharyya distance [16] or symmetrized Kullback-Leibler divergence which can be used to construct a metric. For a survey on divergences see [17].

The equations (10)-(12) represents the merging algorithm and follow from the minimization of the Kullback-Leibler divergence between the Gaussian mixture representation of the normalized density and a single Gaussian approximation of it. There are other statistical distances that can be minimized between the Gaussian mixture representation and Gaussian approximation of the mixture, which would result in a different merging formula. Jensen-Shannon divergence [18] and integral squared error [6] are examples of such statistical distances.

B. Shortcomings and Simple Improvements of Conventional Algorithms

Consider the simple problem of merging two Gaussian components $\{w^1, m^1, P^1\}$ and $\{w^2, m^2, P^2\}$, where we assume without loss of generality that $w^2 \geq w^1$. According to (9) if $(m^1 - m^2)^T (P^1)^{-1} (m^1 - m^2) \leq U$, then the two Gaussian components should be merged. This merging criterion has the following shortcomings

1) P^2 is not affecting the merging decision of (9).
2) If P^1 is relatively large the condition will be satisfied. That is, uncertain targets are sniffed into target groups with larger weight.
3) The uncertainty of target estimates does not affect the mean of the merged targets in (11). That is, an uncertain

Algorithm 1 Reducing Gaussian Mixtures

```
given $\{w_k^i, m_k^i, P_k^i\}_{i=1}^{J_k}$ a truncation threshold $TT$ a merging treshold $U$, and a maximum allowable number of Gaussian terms $J_{\max}$. Set $l = 0$ and $I = \{i = 1, ..., J_k | w_k^i > TT\}$.
repeat
    $l = l + 1.$    (7)
    $j = \text{argmax}\,\{w_k^i\}_{i=1}^{J_k}.$    (8)
    $L = \{i \in I | (m_k^i - m_k^j)^T (P_k^i)^{-1} (m_k^i - m_k^j) \leq U\}.$    (9)
    $\widetilde{w}_k^l = \sum_{i \in L} w_k^i.$    (10)
    $\widetilde{m}_k^l = \frac{1}{\widetilde{w}_k^l} \sum_{i \in L} w_k^i m_k^i.$    (11)
    $\widetilde{P}_k^l = \frac{1}{\widetilde{w}_k^l} \sum_{i \in L} w_k^i \left(P_k^i + (m_k^i - \widetilde{m}_k^l)(m_k^i - \widetilde{m}_k^l)^T\right).$    (12)
    $I = I \backslash L.$    (13)
until $I = \emptyset$.
if $l > J_{\max}$ then replace $\{\widetilde{w}_k^i, \widetilde{m}_k^i, \widetilde{P}_k^i\}_{i=1}^{l}$ by those of the $J_{\max}$ Gaussians with largest weights.
output $\{\widetilde{w}_k^i, \widetilde{m}_k^i, \widetilde{P}_k^i\}_{i=1}^{l}$ as pruned Gaussian components.
```

Algorithm 2 State Extraction

```
given $\{w_k^i, m_k^i, P_k^i\}_{i=1}^{J_k}$
Set $\widehat{X}_k = \emptyset$.
  for $i = 1, ..., J_k$ do
    if $w_k^i > 0.5$ then
      for $j = 1, ..., \text{round}\,(w_k^i)$ do
        $\widehat{X}_k := [\widehat{X}_k, m_k^i]$
      end for
    end if
  end for
output $\widehat{X}_k$ as the multi-target state estimate
```

target (i.e. a target with very large covariance) will satisfy the merging criterion and affect the mean of the merged component according to its relative weight $\frac{w^1}{w^1+w^2}$.

The third shortcoming of the algorithm can be mitigated by incorporating the covariances of the Gaussian components in (11) and forming a weighted sum of the means as in

$$\widetilde{m}_k^l = \left(\sum_{i \in L} w_k^i (P_k^i)^{-1}\right)^{-1} \sum_{i \in L} w_k^i (P_k^i)^{-1} m_k^i, \tag{14}$$

to reduce the impact of Gaussian components with large covariances.

In order to illustrate how the merging statistics behaves when the component covariance changes the merging statistics of a slightly modified version of (9) (i.e. $(m^1 - m^2)^T (P^1 + P^2)^{-1} (m^1 - m^2) \leq U$) is plotted with respect to the component covariance in Figure 3.

A popular remedy suggested in [15] for the aforementioned disadvantages is to consider the mean m^1 as a measurement (with measurement covariance P^1) for the random variable

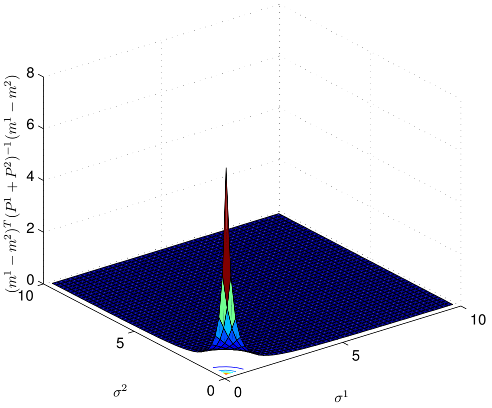


Figure 3. A slightly modified version of the merging statistics of (9); $(m^1 - m^2)^T (P^1 + P^2)^{-1} (m^1 - m^2) \leq U$ is plotted. Here $m^1 = -m^2 = \frac{1}{2}$. The merging criterion will be satisfied if either component has a large covariance compared to the distance of the means of the components.

$x \sim \mathcal{N}(x; m^2, p^2)$ which yields the test

$$\mathcal{N}(m^1; m^2, P^1 + P^2) \geq \gamma, \tag{15}$$

which implies that if

$$(m^1 - m^2)^T (P^1 + P^2)^{-1} (m^1 - m^2) \leq \widetilde{U} - \log\det(P^1 + P^2), \tag{16}$$

then the two Gaussian components, where $\gamma = (2\pi)^{-d/2} \exp\left(-\frac{1}{2}\widetilde{U}\right)$ and d is the dimension of m^1, are merged. This criterion is plotted with respect to the covariance change in Figure 4. Advantages of this approach are:

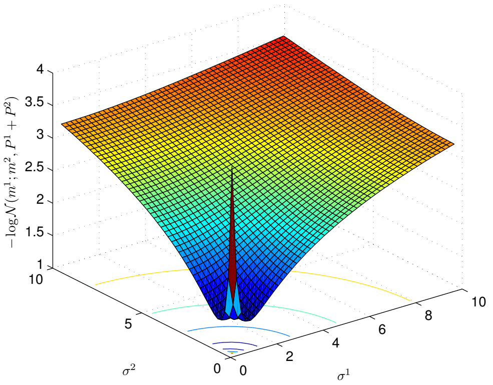


Figure 4. The merging statistics of $-\log\mathcal{N}(m^1; m^2, P^1 + P^2)$ is plotted. Here $m^1 = -m^2 = \frac{1}{2}$. The threshold that merges two components with large covariance will merge two components that are close in distance and have smaller covariance relative to their distance.

1) Both P^1 and P^2 are affecting the merging decision.
2) The merging condition is less sensitive to large covariances due to the second term on the right hand side of (16).

However, this approach has also two disadvantages;

1) Two identical Gaussian components are not always merged. If $\widetilde{U}$ is chosen small, even if two Gaussian components are identical, they are not merged if $\log\det(2P) > \widetilde{U}$

2) If the threshold $\widetilde{U}$ is selected sufficiently large so that the first disadvantage is overcome, then even two well-separated components that have small covariances might be merged.

C. Gaussian Mixture Reduction using GLRT

One advantage of casting the mixture reduction algorithm as a statistical decision theory problem is that two different reduction algorithms can be compared in terms of type I errors and type II errors.

For example, Algorithm 1 can be cast as a decision problem as follows. We want to decide whether or not to merge the two Gaussian components $\{w_k^i, m_k^i, P_k^i\}$ and $\{w_k^j, m_k^j, P_k^j\}$ from the Gaussian mixture at time k

$$p(x) = \sum_{r=1}^{N} w^r \mathcal{N}(x; m_k^r, P_k^r). \tag{17}$$

In Algorithm 1 the merging statistics is defined in (9). In order to cast the problem as a statistical decision theory problem we need an observation. We will take m_k^j as observation and in order to be consistent with the notation used in Section III-B we will refer to it as x^j and we rewrite the test statistics of equation (9). According to (9) the merging statistics $T(\cdot)$ is defined as

$$T(x^j) = (m_k^i - x^j)^T (P_k^i)^{-1} (m_k^i - x^j) \tag{18}$$

Now we formulate the merging statistics and rewrite the test statistics $T(x^j)$ for the observation x^j as a likelihood ratio test.

Decide $\mathcal{H}_1$ (decide to merge) if for the observation $x^j = m_k^j$

$$T(x^j) = (m_k^i - x^j)^T (P_k^i)^{-1} (m_k^i - x^j). \tag{19a}$$

$$= -\log \frac{\mathcal{N}(x^j; m_k^i, P_k^i)}{\mathcal{N}(x^j; m_k^j, P_k^i)} < U, \tag{19b}$$

or equivalently

$$\log \frac{\mathcal{N}(x^j; m_k^i, P_k^i)}{\mathcal{N}(x^j; m_k^j, P_k^i)} > -U. \tag{19c}$$

This likelihood ratio test can be a standard likelihood ratio test for two synthesized simple hypotheses

$$\mathcal{H}_0 : x^j = m_k^j + v[0], \tag{20a}$$

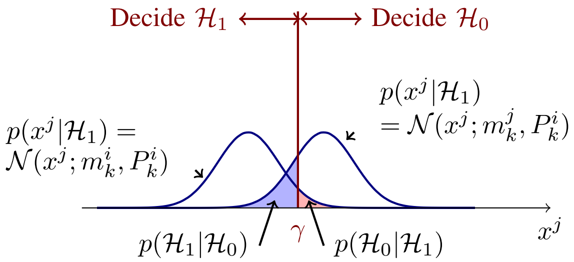
$$\mathcal{H}_1 : x^j = m_k^i + v[0], \tag{20b}$$

where $v[0] \sim \mathcal{N}(0, P_k^i)$. In other words, the hypothesis test $T(\cdot)$ is the likelihood ratio test, suggested by Neyman-Pearson lemma [13], for the hypotheses formulated in (20).

Type I error $p(\mathcal{H}_1|\mathcal{H}_0)$ and type II error $p(\mathcal{H}_0|\mathcal{H}_1)$ for a value of the threshold $\gamma = \exp(-U)$ are illustrated in Figure 5. The tolerable type I or type II error can be given as input and the threshold γ can then be calculated.

By taking a closer look at the formulation of the hypotheses and observations it can be realized that the hypotheses can better represent the merging decision than what is stated in (20). Especially the $\mathcal{H}_1$ in (20) is not presenting the merging hypothesis properly. Here we will formulate two hypotheses and a set of observations which are better than (20) both based on intuition and evidence which is provided in Section IV-D.

First, we will use the notation and formulation given in section III-A to formulate the merging decision as a GLRT.



Decide $\mathcal{H}_1$
Decide $\mathcal{H}_0$
$p(x^j|\mathcal{H}_1)$ $= \mathcal{N}(x^j; m_k^j, P_k^i)$
$p(x^j|\mathcal{H}_1) =$ $\mathcal{N}(x^j; m_k^i, P_k^i)$
$p(\mathcal{H}_1|\mathcal{H}_0)$
γ
$p(\mathcal{H}_0|\mathcal{H}_1)$
x^j

Figure 5. Illustration of hypothesis testing errors $p(\mathcal{H}_1|\mathcal{H}_0)$ and $p(\mathcal{H}_0|\mathcal{H}_1)$ together with their probabilities for the hypotheses formulated in (20).

Assume that a realization of a random variable x is available. In the following section we will refer to it as $(x^i, x^j)^T$ and assign the value $(m_k^i, m_k^j)^T$ i.e., the means of the two Gaussian components to it.

We need a merging function $f(\cdot)$ for merging the two Gaussian components which can be the same as (10)-(12) in Algorithm 1. We will define the notation

$$[w_k^L, m_k^L, P_k^L] = f(\{w_k^l, m_k^l, P_k^l\}_{l \in \{i,j\}}) \tag{21}$$

to be used the in formulation of the hypotheses.

The PDF of the observed random variable under the null hypothesis (not merging) and the alternative hypothesis (merging) are

$$\mathcal{H}_0 : p(x|\mathcal{H}_0) = \mathcal{N}\left(x; \begin{pmatrix}\theta_0^1 \\ \theta_0^2\end{pmatrix}, \begin{pmatrix}P_k^i & 0 \\ 0 & P_k^j\end{pmatrix}\right), \tag{22a}$$

$$\mathcal{H}_1 : p(x|\mathcal{H}_1) = \mathcal{N}\left(x; \begin{pmatrix}\theta_1 \\ \theta_1\end{pmatrix}, \begin{pmatrix}P_k^L & 0 \\ 0 & P_k^L\end{pmatrix}\right), \tag{22b}$$

or equivalently

$$\mathcal{H}_0 : (x^i, x^j)^T = (\theta_0^1, \theta_0^2)^T + e[0], \tag{23a}$$

$$\mathcal{H}_1 : (x^i, x^j)^T = (\theta_1, \theta_1)^T + e[0], \tag{23b}$$

where

$$p(e|\mathcal{H}_0) = \mathcal{N}\left(\begin{pmatrix}0 \\ 0\end{pmatrix}; \begin{pmatrix}P_k^i & 0 \\ 0 & P_k^j\end{pmatrix}\right), \tag{24a}$$

$$p(e|\mathcal{H}_1) = \mathcal{N}\left(\begin{pmatrix}0 \\ 0\end{pmatrix}; \begin{pmatrix}P_k^L & 0 \\ 0 & P_k^L\end{pmatrix}\right), \tag{24b}$$

and $e[0]$ is a realization of e. To determine the most likely hypothesis based on the observation $(x^i, x^j)^T = (m_k^i, m_k^j)^T$, i.e., the mean value of the non-merged components, the generalized likelihood ratio test [13] is applied as follows. Decide $\mathcal{H}_1$ if

$$L_G(x^i, x^j) = \frac{p((x^i, x^j)^T; \widehat{\theta}_1, \mathcal{H}_1)}{p((x^i, x^j)^T; \widehat{\theta}_0, \mathcal{H}_0)} \tag{25a}$$

$$= \frac{p\left(\begin{pmatrix}m_k^i \\ m_k^j\end{pmatrix}; \begin{pmatrix}m_k^L \\ m_k^L\end{pmatrix}, \mathcal{H}_1\right)}{p\left(\begin{pmatrix}m_k^i \\ m_k^j\end{pmatrix}; \begin{pmatrix}m_k^i \\ m_k^j\end{pmatrix}, \mathcal{H}_0\right)} \tag{25b}$$

$$= \frac{\mathcal{N}(m_k^i; m_k^L, P_k^L)\mathcal{N}(m_k^j; m_k^L, P_k^L)}{\mathcal{N}(m_k^i; m_k^i, P_k^i)\mathcal{N}(m_k^j; m_k^j, P_k^j)} > \gamma, \tag{25c}$$

where $\widehat{\theta}_i$ is the MLE of θ_i (maximizes $p(x;\theta_i,\mathcal{H}_i)$). In this problem the MLE of the mean of the Gaussian distributions of each hypothesis are $\widehat{\theta}_0^1 = m_k^i$, $\widehat{\theta}_0^2 = m_k^j$ and $\widehat{\theta}_1 = m_k^L$. The expression of (25c) is the product of the ratios between the likelihood of each mean given the merged density and given the unmerged density.

In order to illustrate how the merging statistics for the merging algorithm of (10)-(12) behaves when the component covariance and the relative weight changes, the merging statistics of (25) is plotted with respect to the components' covariance in Figure 6 for two values of relative weight and with respect to relative weight and relative covariance in Figure 7. It should be noted that the merging statistics would have different behavior if we had chosen a different merging algorithm. The most notable strength of the proposed Gaussian mixture merging statistics, as can be seen in Figure 6 and Figure 7, is that the merging statistics decides not to merge components with large covariance with components with small covariance, i.e., targets with good estimates are not corrupted by targets with bad estimates, since the covariance of the good target, given the merging algorithm we have used (equations (10)-(12)), will be corrupted. Furthermore, the merging statistics is sensitive to the difference in the weight of the components and does not merge components with large weight and covariance, with targets with small weight and covariance. This is a desired property, since if we merge the components using the given merging algorithm a potentially good estimate will be corrupted by an uncertain component.

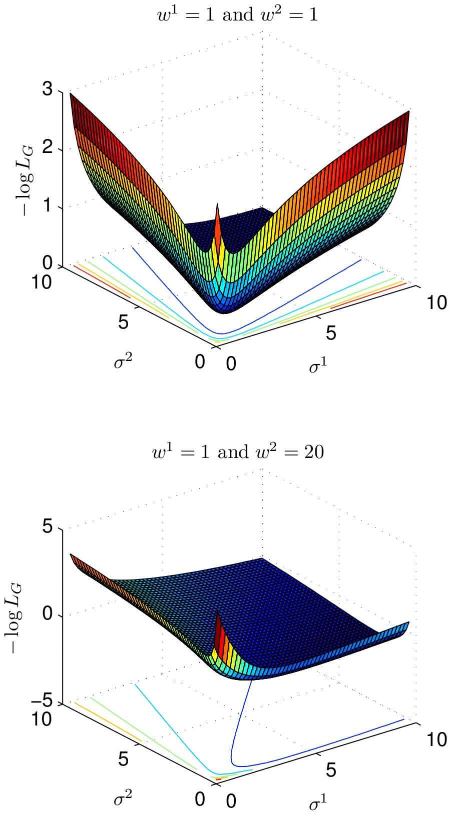


Figure 6. The performance of the proposed merging criterion using hypothesis testing is plotted for two scenarios, in both scenarios $m^1 = -m^2 = \frac{1}{2}$. In the top plot the weights of the Gaussian components are $w^1 = w^2 = 1$ while in the bottom plot $w^2 = 20 \times w^1$. The merging criterion decides not to merge Gaussian components with large weight and uncertainty, but merges Gaussian components with small weight and large covariance. Gaussian components with small covariance are not merged regardless of their weight.

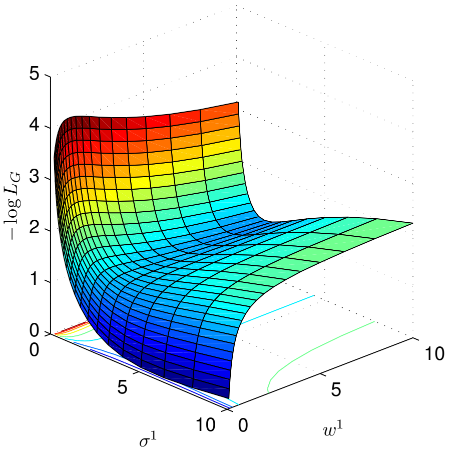


Figure 7. The robustness of the proposed merging criterion using hypothesis testing is shown. $m^1 = -m^2 = \frac{1}{2}$, $w^2 = 1$ and $\sigma^2 = 1$. The merging criterion decides not to merge Gaussian components with large weight and uncertainty, but merges Gaussian components with small weight and large covariance. Gaussian components with small covariance are not merged regardless of their weight.

A standard method of comparing statistical tests is via Receiver Operating Characteristic (ROC) curve. In Figure 8 the ROC curve for both of the studied merging statistics, GLRT and Mahalanobis Distance Test (MDT), for three Gaussian pairs are plotted. As it is can be seen the GLRT gives lower type II error (higher $p(\mathcal{H}_1|\mathcal{H}_1)$) for all values of type I error compared to the MDT. This result is not a surprise and follows the Neyman-Pearson lemma.

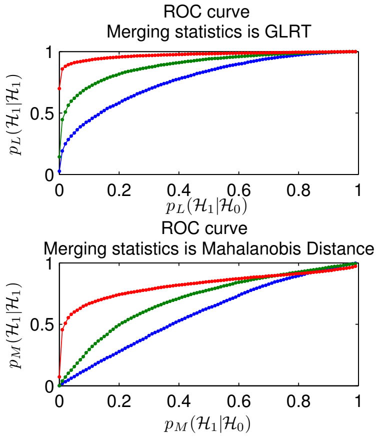


Figure 8. The ROC curve for three pairs of Gaussian components are plotted for two types of tests. The parameters of the Gaussian components are $w^1 = 1, m^1 = 0, P^1 = 4$ and $w^2 = 2, m^2 \in \{2,4,9\}, P^2 = 1$.

D. Numerical Simulation

The GM-PHD filter gives rise to a mixture with an increasing number of Gaussian components even in a "no clutter single target" scenario. The resulting mixture has to be reduced to maintain the computational feasibility. In such a scenario two Gaussian components one of which originating from the Kalman filter update by a measurement, and another one originating from prediction of the same component without association of measurement will usually be very close. These two components will potentially be merged in the conventional GM-PHD filter which will result in a suboptimal estimate of the target state and associated covariance. This phenomenon can be alleviated if the reduction is performed according to Figure 2.

In order to illustrate the impact of Gaussian mixture reduction in the conventional algorithm and highlight possibilities in performing the reduction according to Figure 2, a Monte Carlo simulation is performed for 30 realizations of a scenario where a single target is moving along a line. The standard GM-PHD filter according to [14] is implemented for three values of the merging threshold $U = 4$, 0.4, 0.04 and the covariance of the extracted target state is compared with the covariance obtained from the Kalman filter in Figure 9. When a small merging threshold is chosen, Gaussian components are more likely to remain unmerged with other components until their weight turns so small that they are pruned away. Therefore, the components of the mixture which are close to measurements have a covariance which is closer to the estimate covariance obtained from the Kalman filter (the optimal estimate). The simple conclusion from this experiment is that if the proposed split reduction algorithm is adopted with two separate reduction algorithms and the merging is done only in response to computational feasibility rather than proximity of components, the state estimate can be improved.

To emphasize on importance of the choice of merge statistics and its sensitivity we show an example which is frequently encountered in GM-PHD filtering. Here we want to compare the proposed GLRT which will be denoted by $L_G(\cdot)$ with the conventional MDT which will be denoted by $T(\cdot)$, but first we need to find equivalent thresholds for both tests. This can be done using the ROC curve or by an intuitive method suggested below. Consider a Gaussian component $\{w^i = 1, m^i = 0, P^i = 1\}$ which corresponds to a target in the GM-PHD framework. Now consider two other Gaussian components with parameters $\{w^j = 0.2, m^j = 10, P^j = 100\}$ and $\{w^k = 0.2, m^k = 100, P^k = 10000\}$ which if merged with the first component will result in $\{w^{ij} = 1.2, m^{ij} = 1.667, P^{ij} = 31.39\}$ and $\{w^{ik} = 1.2, m^{ik} = 16.67, P^{ik} = 3056\}$ respectively. The j^{th} component can be produced from the i^{th} target in the GM-PHD filter's recursion, but the k^{th} component can be from another target rather far away from the i^{th} target.

Now let us look at the merge statistics given by the GLRT and the MDT. For the pair $\{i,j\}$, $T(\{i,j\}) = 1$. Let us assume that the merging of the pair $\{i,j\}$ defines the threshold for a tracking application i.e. the threshold for the MDT is set to 1. In order to draw equivalence between the thresholds we calculate $L_G(\{i,j\}) = 0.1008$, i.e. we use 0.1008 as the threshold in the GLRT merging statistics. Now, let us look at the pair $\{i,k\}$. Although from the parameters of the Gaussian components it is obvious that merging the pair $\{i,k\}$ is undesired, because of large displacement of the estimate of a target, $T(\{i,k\}) = 1$, i.e., the statistics based on the Mahalanobis distance will merge the pair $\{i,k\}$, while the statistics based on the GLRT will not: $L_G^{ik} = 0.0100 < 0.1008$. In this simple example we illustrated the importance of a properly designed merge statistics and how the implicit feedback from the resulting merged Gaussian in the merge statistics strengthen the test with respect to outliers. The type of problem that is highlighted here can not be solved by simply reducing the merging threshold since regardless of the threshold the same type of realistic scenario can be produced. Inspired by this simple example we simulated a tracking scenario composed of

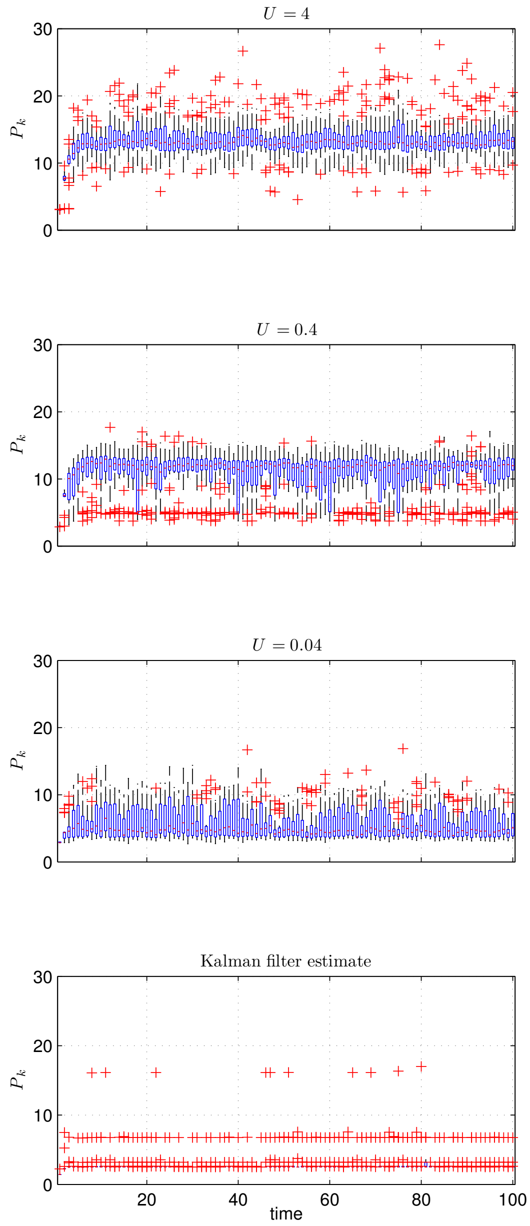


Figure 9. A single target moving in one dimensional space with nearly constant velocity model. The covariance of the estimate of target's position P_k is plotted versus time in a box plot. The impact of merging threshold U on the estimate covariance is shown for three scenarios of GM-PHD filtering with different merging thresholds in the top three plots. In the bottom plot the estimate covariance given by the Kalman filtering is presented. Simulation parameters are $P_D = 0.85$, Number of Monte Carlo simulations =30, measurement covariance=4 and process noise covariance=$4 * I_2$. The estimate covariance decreases as the threshold decreases and more components are pruned rather than merged with other components.

two targets moving along the same line with some longitudinal distance in between and a tracker using a GM-PHD filter which is illustrated in Figure 10. The difference between the two position estimates is due to the different merge statistics in the two GM-PHD filters implemented on the same set of measurements. There are two occasions where the MDT merges Gaussian components from the two distinct targets and creates erroneous state estimates. See time indices 80 and 95 for the undesired merging instances.

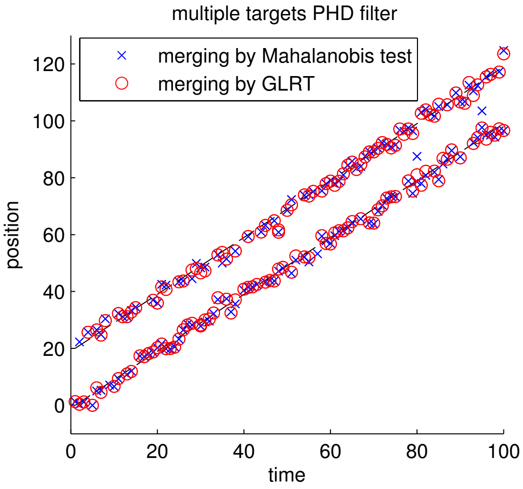


Figure 10. Tracks of two targets moving along the same line give by GM-PHD filter. The difference between the two position estimates is only due to the different merge statistics in two GM-PHD filters implemented on the same set of measurements. There are two occasions where the MDT merges Gaussian components from the two distinct targets and creates wrong state estimates. See time indices 80 and 95 for the undesired merging instances.

In multiple target tracking scenarios, the Gaussian components with large weight and covariance are created when the means of several components are spread close to each other or when a clutter measurement happens to be close to the prediction of a target or two targets get too close to each other. Large covariance can also be created due to target maneuvers and large process noise in the model which was the case for our simulation.

V. Conclusion

We have proposed a sequentially pairwise mixture reduction criterion and algorithm based on statistical decision theory and generalized likelihood ratio test. The advantages and disadvantages of some of the previous reduction schemes and the newly proposed algorithm are discussed in detail. The results are evaluated on a Gaussian mixture implementation of the PHD filter in a two target scenario and it is illustrated how robust the proposed merging statistics is with regard to uncertain Gaussian components. Using the proposed framework the merging statistics is selected in a way that is tightly coupled with the merging algorithm and the choice of the threshold is connected to type I and type II errors in decision theory. Using the hypotheses test framework a window towards using other measures defined over distributions and other statistical test will be opened which will allow a more rigorous treatment of reduction algorithms. The second contribution in this paper has been splitting of the reduction algorithm into two subroutines, one for computational feasibility, the other for the state extraction each of which tailored for its purpose. The future work on this subject can be comparison of the proposed method with other existing algorithms such as the algorithm proposed by Runnalls in [5]. Further evaluation of the proposed reduction algorithm for aggressive reductions such as in the reduction for state extraction and less aggressive reductions focused on maintaining the computational feasibility can be mentioned as a future work as well.

Acknowledgment

The authors would like to thank the project Collaborative Unmanned Aircraft Systems (CUAS), funded by the Swedish Foundation for Strategic Research (SSF), as well as the Linnaeus research environment CADICS and the frame project grant Extended Target Tracking (621-2010-4301), both funded by the Swedish Research Council (VR) for financial support.

References

[1] D. Salmond, "Mixture reduction algorithms for target tracking," in *State Estimation in Aerospace and Tracking Applications, IEE Colloquium on*, London, UK, dec 1989, pp. 7/1–7/4.

[2] O. Bochardt, R. Calhoun, J. Uhlmann, and S. Julier, "Generalized information representation and compression using covariance union," in *Information Fusion, 2006 9th International Conference on*, Florence, Italy, july 2006, pp. 1–7.

[3] D. Crouse, P. Willett, K. Pattipati, and L. Svensson, "A look at Gaussian mixture reduction algorithms," in *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, july 2011, pp. 1–8.

[4] M. Huber and U. Hanebeck, "Progressive Gaussian mixture reduction," in *Information Fusion, 2008 11th International Conference on*, Cologne, Germany, july 2008, pp. 1–8.

[5] A. Runnalls, "Kullback-Leibler approach to gaussian mixture reduction," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 43, no. 3, pp. 989–999, july 2007.

[6] J. L. Williams and P. S. Maybeck, "Cost-function-based hypothesis control techniques for multiple hypothesis tracking," *Mathematical and Computer Modelling*, vol. 43, no. 9-10, pp. 976–989, May 2006. [Online]. Available: http://dx.doi.org/10.1016/j.mcm.2005.05.022

[7] D. Schieferdecker and M. Huber, "Gaussian mixture reduction via clustering," in *Information Fusion, 2009. FUSION '09. 12th International Conference on*, Seattle, WA, USA, july 2009, pp. 1536–1543.

[8] J. E. Harmse, "Reduction of Gaussian mixture models by maximum similarity," *Journal of Nonparametric Statistics*, vol. 22, no. 6, pp. 703–709, 2010. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/10485250903377293

[9] D. Salmond, "Mixture reduction algorithms for point and extended object tracking in clutter," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 45, no. 2, pp. 667–686, april 2009.

[10] P. Bruneau, M. Gelgon, and F. Picarougne, "Parsimonious reduction of Gaussian mixture models with a variational-bayes approach," *Pattern Recogn.*, vol. 43, pp. 850–858, March 2010. [Online]. Available: http://dl.acm.org/citation.cfm?id=1660180.1660681

[11] S. Reece and S. Roberts, "Generalised covariance union: A unified approach to hypothesis merging in tracking," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 46, no. 1, pp. 207–221, jan. 2010.

[12] J. Neyman and E. S. Pearson, "On the problem of the most efficient tests of statistical hypotheses," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 231, pp. 289–337, 1933.

[13] S. Kay, *Fundamentals of Statistical Signal Processing: Detection theory*, ser. Prentice Hall signal processing series. Prentice-Hall PTR, 1998. [Online]. Available: http://books.google.se/books?id=vA9LAQAAIAAJ

[14] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *Signal Processing, IEEE Transactions on*, vol. 54, no. 11, pp. 4091–4104, nov. 2006.

[15] S. Blackman and R. Popoli, *Design and analysis of modern tracking systems*, ser. Artech House radar library. Artech House, 1999.

[16] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.

[17] M. D. Reid and R. C. Williamson, "Information, Divergence and Risk for Binary Experiments," *Journal of Machine Learning Research*, vol. 12, pp. 731–817, Mar. 2011.

[18] J. Lin, "Divergence measures based on the Shannon entropy," *Information Theory, IEEE Transactions on*, vol. 37, no. 1, pp. 145–151, jan 1991.