Two-filter Gaussian mixture smoothing with posterior pruning

Abu Sajana Rahmathullah*, Lennart Svensson*, Daniel Svensson†
*Department of Signals and Systems, Chalmers University of Technology, Sweden
†Electronic Defence Systems, SAAB AB, Sweden
Emails: {sajana, lennart.svensson, daniel.svensson}@chalmers.se

***Abstract*—In this paper, we address the problem of smoothing on Gaussian mixture (GM) posterior densities using the two-filter smoothing (TFS) strategy. The structure of the likelihoods in the backward filter of the TFS is analysed in detail. These likelihoods look similar to GMs, but are not proper density functions in the state-space since they may have constant value in a subspace of the state space. We present how the traditional GM reduction techniques can be extended to this kind of GMs. We also propose a posterior-based pruning strategy, where the filtering density can be used to make further approximations of the likelihood in the backward filter. Compared to the forward–backward smoothing (FBS) method based on $N-$scan pruning approximations, the proposed algorithm is shown to perform better in terms of track loss, normalized estimation error squared (NEES), computational complexity and root mean squared error (RMSE).**

***Index Terms*—filtering, smoothing, Gaussian mixtures, two-filter smoothing, backward likelihood, data association**

I. Introduction

In many sequential estimation problems, such as data association problem [1] and glint problem [2] in radar tracking and synchronization problem in communication systems, the uncertainties involved in the system are multi-modal and can be modeled using Gaussian mixtures (GMs). In these applications, the optimal solutions to the estimation problems of prediction, filtering and smoothing of GMs have closed form expressions, which can be obtained using the optimal solutions for the Gaussian densities [8, 14].

Often the optimal solutions have a complexity that increases exponentially with time. For instance, in the data association problem in target tracking, to obtain the optimal solution, each observed measurement is either associated to a target or declared as clutter at each time resulting in sequences of data association hypotheses referred to as track hypotheses. Using these data association hypotheses during prediction, filtering and smoothing results in GM posterior densities in which the number of components grows exponentially with time.

The two main smoothing methods [6] are forward-backward smoothing (FBS) and two-filter smoothing (TFS). When the posterior densities are Gaussian, closed form solutions are available in [11] for FBS and in [7] for TFS. In this paper, we focus on the TFS method for GMs. The TFS algorithm works by running two independent filtering algorithms: the forward filtering (FF) and the backward filtering (BF). The smoothing density is obtained by multiplying the outputs of the FF and the BF. Ideally, the two filters are run independently of each other. The FF is the conventional GM filtering one (eg. a Multiple hypothesis tracking algorithm [12]). The BF, as the name suggests, is run in the opposite direction from the last time step to the start and can be obtained using a recursive procedure, similar to the FF. The likelihood returned by the BF, referred as reduced dimension GM (RDGM) in this paper, has a GM type of structure, but is not normalizable. The problem is that the traditional GM reduction (GMR) techniques, such as pruning and merging, cannot be used directly on the RDGM of the BF, as these techniques work on normalizable densities. Therefore, design of GMR techniques for the RDGM structure is necessary for a practical algorithm.

The TFS on GMs is discussed in [8], which provides the structure of the RDGM, but practical approximation techniques are not discussed in detail. In the particle filter setting of the TFS problem, the work in [4] proposes using an artificial prior in the BF, similar to the prior which we use in the FF, to make the output of the BF a proper density. The conventional techniques can then be used on the BF's output to sample particles. But extending this artificial prior concept to GM TFS has the problem that it involves division of densities, which can lead to indefinite covariances.

In this paper, we study the TFS of GM, with focus on the backward filter. Based on the analysis of the structure of the RDGM of the BF, we propose strategies where GMR techniques are used within certain groups of components of the RDGM. We also present the smoothed posterior pruning (SPP) method where we show that the filtering density from the FF can be used to approximate the RDGM. Simulations for a single target scenario show that the TFS based on proposed strategies outperforms the pruning-based FBS method. Additionally, the estimates from the TFS are consistent, with lower track loss and cheaper computational complexity.

II. Problem statement and idea

We consider a single target moving in a cluttered background. The state vector x_k of dimension N_x at time k is according to the process model,

$$x_k = Fx_{k-1} + v_k, \tag{1}$$

where v_k is Gaussian with zero mean and covariance Q and is denoted as $\sim \mathcal{N}(0, Q)$. The target is detected with probability P_D. Measurement set Z_k includes the clutter measurements and the target measurements z_k^t, when detected. The target measurement is given by

$$z_k^t = Hx_k + w_k \tag{2}$$

where $w_k \sim \mathcal{N}(0, R)$. The clutter measurements are assumed to be uniformly distributed in the observation region of volume V. The number of clutter measurements is Poisson distributed with parameter βV, where β is the clutter density. The number of measurements in Z_k is denoted by m_k. We assume that we have access to the measurements $Z_{1:K}$ from time 1 to K where $K \geq k$.

The objective is to compute the smoothing density $p(x_k|Z_{1:K})$ using the TFS method, which involves running two filters — the FF gives the filtering density $p(x_k|Z_{1:k})$ and the BF gives the backward likelihood (BL) $p(Z_{k+1:K}|x_k)$. The FF is the same as the one in FBS, which is well studied in the literature. The BL, which is the output of the BF, looks similar to a GM density function but is not normalizable. Because of this structure of the likelihood, the traditional GMR techniques cannot be applied directly. The goal of this paper is to devise strategies for GMR of the BL to reduce the complexity of the BF.

A. Idea

The BL of the BF has a mixture structure in which the components are densities in different subspaces of the state space. This structure is referred to as reduced dimension GM (RDGM) in this paper. By close investigation, one can observe that the components in the RDGM can be partitioned into groups such that the components within each group are density functions in the same subspace. Consequently, this grouping allows us to use traditional GMR techniques within each group. The details of how the intragroup approximations are performed are covered in Section IV. We also discuss the limitations of these intragroup methods.

Besides the intragroup approximations, the availability of the filtering density from the FF can be used to reduce the number of components further in the RDGM of the BF. That is, based on the components in the smoothing density, further GMR can be performed on the RDGM, as will be shown in Section V.

III. Background

In this section, we present a background to the TFS method for GM densities. The conceptual solutions to the FF and the BF in the TFS are provided. The solution to the FF and BF of GM is described using a graphical illustration.

A. Two-filter smoothing

The goal of smoothing is to compute the smoothing posterior $p(x_k|Z_{1:K})$. The TFS method computes the smoothing density, at each k, according to

$$p(x_k|Z_{1:K}) \propto p(x_k|Z_{1:k})p(Z_{k+1:K}|x_k). \tag{3}$$

The filtering density $p(x_k|Z_{1:k})$ and the BL $p(Z_{k+1:K}|x_k)$ are obtained as outputs of the FF and the BF, respectively. It should be pointed out, that the counterpart of TFS in the factor graph world is the belief propagation algorithm [10], where the filtering steps are termed as message passing.

1) Forward Filtering : The FF involves two steps, namely the prediction and the update step, which are performed recursively from time $k = 1$ to $k = K$. The prediction step computes the prediction density,

$$p(x_k|Z_{1:k-1}) = \int p(x_{k-1}|Z_{1:k-1})f(x_k|x_{k-1})dx_{k-1} \tag{4}$$

followed by the update step to compute the filtering density,

$$p(x_k|Z_{1:k}) \propto p(x_k|Z_{1:k-1})p(Z_k|x_k). \tag{5}$$

For the assumptions made in Section II, it can be shown that the filtering density is a GM and has the form

$$p(x_k|Z_{1:k}) = \sum_{i=1}^{M_k^f} w_{k,i}^f \mathcal{N}\left(x_k;\, \mu_{k,i}^f,\, P_{k,i}^f\right) \tag{6}$$

where $\mathcal{N}\left(x_k;\, \mu_{k,i}^f,\, P_{k,i}^f\right)$ represents a Gaussian density in variable x_k with mean $\mu_{k,i}^f$ and covariance $P_{k,i}^f$ and $w_{k,i}^f$ is the corresponding weight. The weight $w_{k,i}^f$ in (6) is the probability $\Pr\{\mathcal{H}_{k,i}^f|Z_{1:k}\}$ of the hypothesis $\mathcal{H}_{k,i}^f$ which corresponds to a sequence of measurements or the missed-detection associations from time 1 to k. Kalman filtering under $\mathcal{H}_{k,i}^f$ gives the mean $\mu_{k,i}^f$ and the covariance $P_{k,i}^f$.

The number of components in the filtering GM is the product of the number of measurements observed over time. So, the number of components grow exponentially with time. Therefore, the number GMs have to be reduced using GMR techniques, such as pruning and merging. GMR techniques such as pruning and merging are used for practical implementations of FF. A few pruning methods to mention are gating [3], $N-$scan pruning, M-best pruning and threshold-based pruning. For merging algorithms, one can refer to [9], [13], [16] and [5]. The merging algorithms presented in these papers preserve the moments, i.e., the mean and covariance of the GM before and after merging are the same.

2) Backward Filtering: The BF in TFS also involves recursive steps, similar to the FF, from time $k = K - 1$ to $k = 1$. The first step is the update step given by

$$p(Z_{k+1:K}|x_{k+1}) = p(Z_{k+1}|x_{k+1})p(Z_{k+2:K}|x_{k+1}), \tag{7}$$

followed by the retrodiction step to obtain the BL,

$$p(Z_{k+1:K}|x_k) \propto \int p(Z_{k+1:K}|x_{k+1})\, f(x_{k+1}|x_k)\, dx_{k+1}. \tag{8}$$

For the assumptions made in Section II, similar to the FF, the BF can also be interpreted as running filtering operations under data association hypotheses. In the BF, each hypothesis, denoted $\mathcal{H}_{k,j}^b$, corresponds to a sequence of measurement or missed-detection associations made from time K until time $k+1$, where $k < K$.

Assuming that there are only pruning approximations made during FF and BF, the TFS can be illustrated using graphical structures as shown in Fig. 1. Two hypothesis trees (to be precise, the BF needs a graph illustration, as there is not one root), one for each filter, are shown in where the nodes represent the components in the GM of $p(x_k|Z_{1:k})$. The details of the structure of the output of the BF and how the proposed algorithms can be used to make approximations in the BF are presented in the Sections IV and V.

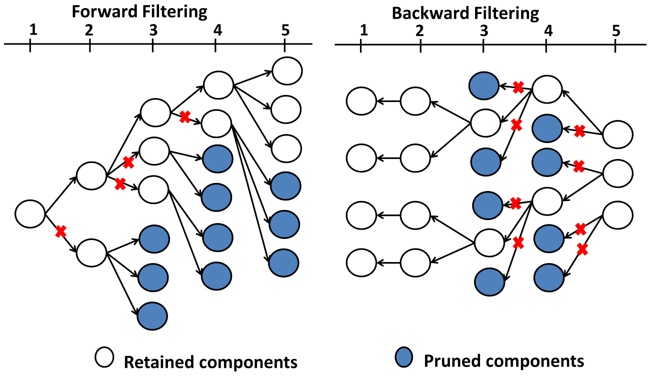

Figure 1: Illustration of TFS: The nodes correspond to the GM components in the filter outputs. The 'blue nodes' represent the pruned nodes and they are not propagated further. At each time instant, the smoothing density is obtained as the product of the GMs, corresponding to all the nodes in the two filters.

IV. Intragroup approximations of the backward likelihood

In this section, we analyze the details of the BL in the BF that has the RDGM structure. We discuss why the conventional pruning and merging strategies are not always suitable for the BL. Based on the analysis of the RDGM structure of the BL, we show that the components in the RDGM can be grouped and that the traditional GMR techniques can be used within the groups. It will also be shown that the number of groups is polynomial in the lag $K-k$, and that this can be a limitation of the intragroup GMR strategies, especially when the lag $K-k$ is large.

A. Structure of the backward likelihood

Under the assumptions made in Section II, starting with the likelihood of the form

$$p(Z_K|x_K) = \beta_{K,0} + \sum_{m=1}^{m_K} \beta_{K,m} \mathcal{N}\left(Hx_K;\, z_{K,m},\, R\right) \quad (9)$$

at time K and evaluating the update and retrodiction recursions in (8) and (7), the BL can be written in the form,

$$p(Z_{k+1:K}|x_k) = w^b_{k,0} + \sum_{j=1}^{M^b_k} w^b_{k,j} \mathcal{N}\left(H^b_{k,j}x_k; \mu^b_{k,j}, P^b_{k,j}\right) \quad (10)$$

for any $k < K$. The parameters in (9) and (10) can be obtained using the extended observation model described in Section 3 of [8] or from Section II-C of [15]. Note that the term $w^b_{k,0}$ is zero if the probability of detection is one. The expression in (10), which is the RDGM, is similar to a GM, with weights $w^b_{k,j}$, means $\mu^b_{k,j}$ and covariances $P^b_{k,j}$. However, the terms $\mathcal{N}\left(H^b_{k,j}x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right)$ are not generally densities in x_k and are not guaranteed to be normalizable since the integral $\int \mathcal{N}\left(H^b_{k,j}x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right) dx_k$ may be infinite.

B. Normalizability of the BL and intragroup approximations

Similar to the GM filtering density in the FF, the number of terms M^b_k in the RDGM in (10) of the BF grows exponentially with time. Thus, GMR approximations are necessary to reduce the complexity. The catch is that the conventional GMR techniques mentioned in Section III-A1 are developed for GM density functions and cannot be applied directly to RDGMs. For instance, the conventional pruning strategies are applied to GMs that are normalized, in which the weights of the components define the relative sizes of the Gaussian components. So, the weights of the components can be compared and the ones with insignificant weights can be removed. However, in the RDGM in (10), it is possible that a component with the smallest weight $w^b_{k,j}$ is in fact among the largest components. So, the components in the RDGM cannot be compared based only on the weights unless the components $\mathcal{N}\left(H^b_{k,j}x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right)$ are normalized. Similarly, the conventional merging strategies applied to GMs involve moment matching, and to compute the moments, the components should be normalizable densities. In the trivial case, when the components in the RDGM are normalizable, i.e., when $\text{rank}(H^b_{k,j}) = N_x$, it is possible to rewrite the components in the RDGM $\sum_j w^b_{k,j}\mathcal{N}\left(H^b_{k,j}x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right)$ into the GM form $\sum_j w'^b_{k,j}\mathcal{N}\left(x_k;\, \mu'^b_{k,j},\, P'^b_{k,j}\right)$ to which the GMR techniques can be applied.

The RDGM is in general not normalized, like in the data association problem, and the normalizability depends on the structure of the matrices $H^b_{k,j}$. AS will be shown in Section IV-B1, in the RDGM, groups of components have identical $H^b_{k,j}$ matrices. Thus, if the matrices $H^b_{k,j}$ are also of full row rank in a group, then it is possible to compare the components within the group, and so, one can apply the GMR technique to that group.

Let I_g be the index set that contains the indices j of the components in a group g that have the same matrices, i.e., $H^b_{k,j} = H_g$ for $j \in I_g$ and let $\text{rank}(H_g) = m_g$. Using this notation, the RDGM in (10) can be written as

$$p(Z_{k+1:K}|x_k) = \sum_{g=1}^{N_G} \sum_{j \in I_g} w^b_{k,j} \mathcal{N}\left(H_g x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right) \quad (11)$$

where N_G is the number of groups. The functions $\mathcal{N}\left(H_g x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right)$ are scaled Gaussian functions in the row space of the matrix H_g, and have constant values in the null space of H_g. The idea is that the GMR can be applied to approximate the group $\sum_{j\in I_g} w^b_{k,j}\mathcal{N}\left(H_g x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right)$ of components in this row space and that no approximation needs to be performed in the null space, as the values of the components are constant in the null space. An interpretation of this can be obtained using a change of basis matrix, say $A_g = \begin{bmatrix} H_g \\ H_g^\perp \end{bmatrix}$ in $\mathbb{R}^{N_x}$, where the matrix $H_g^\perp$ is such that the columns and rows of A_g span $\mathbb{R}^{N_x}$. Let us define a variable $y_g = A_g x_k$ with respect to the new basis in $\mathbb{R}^{N_x}$. Using this variable y_g, the function $w^b_{k,j}\mathcal{N}\left(H_g x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right)$ is written as $w^b_{k,j}\mathcal{N}\left(y_g\left[1:m_g\right];\, \mu^b_{k,j},\, P^b_{k,j}\right)$, where $y_g\left[1:m_g\right]$ denotes the first m_g elements in the vector y_g. Clearly, the functions $\mathcal{N}\left(y_g\left[1:m_g\right];\, \mu^b_{k,j},\, P^b_{k,j}\right)$ are Gaussian in the variable $y_g[1:m_g]$. These functions are not integrable in $y_g[m_g+1:N_x]$, but are constant (or uniform) in that variable. This observation allows us to treat the group $\sum_{j\in I_g} w^b_{k,j}\mathcal{N}\left(y_g\left[1:m_g\right];\, \mu^b_{k,j},\, P^b_{k,j}\right)$ as a GM in the variable $y_g[1:m_g]$. So, pruning and merging can be applied to approximate this mixture as a function of $y_g[1:m_g] = H_g x_k$. Overall, within each group g, the GMR can be applied as

$$\begin{aligned} &\sum_{g=1}^{N_G}\sum_{j\in I_g} w^b_{k,j}\mathcal{N}\left(y_g[1:m_g];\, \mu^b_{k,j},\, P^b_{k,j}\right) \\ \approx\ &\sum_{g=1}^{N_G}\sum_{j\in I'_g} \tilde{w}^b_{k,j}\mathcal{N}\left(y_g[1:m_g];\, \tilde{\mu}^b_{k,j},\, \tilde{P}^b_{k,j}\right) \\ =\ &\sum_{g=1}^{N_G}\sum_{j\in I'_g} \tilde{w}^b_{k,j}\mathcal{N}\left(H_g x_k;\, \tilde{\mu}^b_{k,j},\, \tilde{P}^b_{k,j}\right) \qquad (12)\end{aligned}$$

where I'_g refers to the index set corresponding to components in group g after GMR. From (12), it can be noticed that after the intragroup approximations, the number of components in the RDGM is at least the number of groups, N_G.

1) Grouping: As discussed earlier in this section, the grouping of the components in the the BL in (10) is key in being able to apply the GMR techniques to approximate the RDGM. In this section, with the data association in target tracking as an example, we discuss why the grouping is possible. For this example, we also analyze how the number of groups grows with time to point out the limitation of the intragroup GMR methods.

By investigating (9) and (10) for the DA problem, one can see that many of the components in (10) have the same $H^b_{k,j}$ matrices. It turns out that if two DA hypotheses, $\mathcal{H}^b_{k,i}$ and $\mathcal{H}^b_{k,j}$, are such that they have measurement associations and missed-detection associations at the same time instants, then $H^b_{k,i} = H^b_{k,j}$. This observation allows us to partition the components in the RDGM into groups of components that have the same $H^b_{k,j}$ matrices.

Consider the constant velocity model, in which the state contains the position and velocity components, and assume that only the position components are observed. One then needs measurements from at least two different time instants to estimate the velocity components. This implies that under the corresponding DA hypotheses $\mathcal{H}^b_{k,j}$ in the BF, which have at least two measurement associations across time, the state can be estimated with finite covariance. So, the corresponding components $\mathcal{N}\left(H^b_{k,j}x_k;\, \mu^b_{k,j},\, P^b_{k,j}\right)$ in the BL will be normalizable and therefore, the ranks of the corresponding matrices $H^b_{k,j}$ is $N_x = \dim(x_k)$. These normalizable components will form one group. Additionally, there will also be components with $H^b_{k,j}$ such that $\text{rank}(H^b_{k,j}) < N_x$ which correspond to the hypotheses sequences $\mathcal{H}^b_{k,j}$ that have less than two measurements associations across time.

We will now analyze how the number of groups grows with time. Let us assume that the matrices H and F in the motion model (1) and measurement model (2) are such that measurements from at least M different time instants (not necessarily consecutive) are needed to estimate all dimensions of the state x_k, i.e., to ensure that the corresponding component in the BL is normalizable. Using this parameter M and the time lag $K-k$, the number of groups can be analysed. As discussed for the constant velocity model, the components corresponding to the hypothesis sequences that have at least M measurement associations across time will be normalizable and form a group. Components that have hypothesis sequences with exactly l ($l < M$) measurement associations at the same time instants belong to the same group. There are $\begin{pmatrix} K-k \\ l \end{pmatrix}$ ways of assigning l measurement associations in $K-k$ time steps for each $l = 0, 1 \ldots M-1$. Thus, in total, the number of groups is given by

$$N_G(k,K) \;=\; 1 + \sum_{l=0}^{M-1}\begin{pmatrix} K-k \\ l \end{pmatrix} \qquad (13)$$

which grows in the order of $\mathcal{O}\left((K-k)^{M-1}\right)$ as a function of the lag $K-k$.

One can see that using the intragroup approximations, the number of components in the RDGM cannot be reduced to less than the number of groups, which grows as $\mathcal{O}\left((K-k)^{M-1}\right)$ with the lag $K-k$. Hence, these intragroup approximations are not sufficient to reduce the complexity due to the RDGM structure of the BL. It is therefore essential to present a sound manner to compare components across groups in order to enable us to reduce the complexity further. In the next section, we present the smoothed posterior-based pruning with which the components in the BL can be compared and pruned.

V. Smoothed posterior-based pruning

In this section, we show that it is valid to compare components across groups in the BL based on the smoothing probabilities of the corresponding hypotheses in the BF. Surprisingly enough, using these probabilities, large components or even groups of components can be pruned from the BL and propagated, without affecting the smoothing densities at any time instant. The smoothing probabilities of the hypotheses in the output of the BF can be calculated using the FF densities. We also show how this idea can be generalized for pruning components from the FF as well.

A. Posterior-based pruning

The main objective is to compute the smoothing posterior density given in (3). The two terms in the right-hand side of (3) are the filtering density and the backward likelihood, whichs are the outputs of the FF and BF, respectively. The filtering density is reduced using traditional GMR techniques. The BL can be reduced using the intragroup approximations discussed before, but that may not be enough due to its limitations discussed in the previous section. We propose (and later prove) that the filtering density of the FF in (3) can be used to prune components in the BL of the BF, in regions where the smoothing posterior density is small, and therefore is not affected by this pruning. Fig. 2 shows an illustration of this strategy, referred to as smoothed posterior pruning (SPP), for one time instant. The GMs are in 2D and the curves shown in the figure are the contour plots of the GMs involved. More importantly, we can propagate the pruned version of the BL backwards in time without affecting the smoothing density at any other time instant.

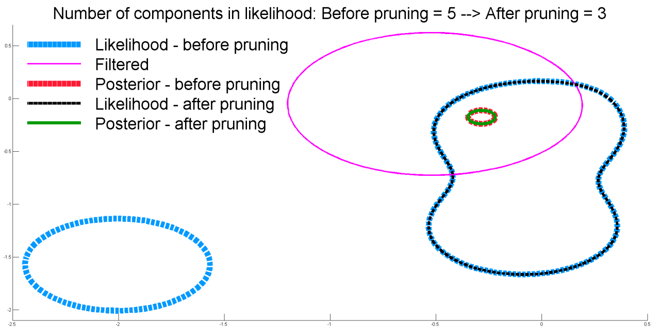

Figure 2: Illustration of SPP: The green and the red curves, which represent the posterior density before and after the approximation of the likelihood, are very close to each other. This shows that the shape of the corresponding (smoothing) posterior is unaffected by the posterior pruning of the likelihood. Also, parts of the blue curve and black curve, which represent the likelihood before and after pruning, are very close to each other.

The SPP idea can also be used the other way around. That is, it is possible to approximate the FF based on the BL. The BL of the BF, when available, can be used to prune components in the filtering density of the FF in the regions where the smoothing density is small. The bottom line is that the SPP idea presented in this section can be generalized to both the FF and the BF, depending on what functions are available to compute the posterior. In this section, we discuss the SPP idea for the FF, but the same arguments hold for SPP of BF as well.

For the pruning step in the SPP, any of the pruning strategies mentioned in Section III-A1 can be used, once we know the weights of the components to be compared. Note that the weights relate to the probability of the data association hypothesis sequence $\mathcal{H}^f_{k,i}$ (cf. Section III-A1). In SPP, as stated before, we want to perform the pruning based on the smoothing posterior. So, we do not use the filtering probability $\Pr\{\mathcal{H}^f_{k,i}|Z_{1:k}\}$ of the hypothesis. Instead, we use the smoothing probability $\Pr\{\mathcal{H}^f_{k,i}|Z_{1:K}\}$ of the hypothesis. As we will show in the following proposition, if the smoothing posterior probability $\Pr\{\mathcal{H}^f_{k,i}|Z_{1:K}\}$ is zero, then the i^{th} Gaussian component can be pruned from the filtering density, without affecting the smoothing posterior distributions. The intuitive reason is that the smoothing probabilities of the offsprings of the pruned component, if it was propagated, would also be zero, which means that they would not influence the smoothing densities at later times either. In the following proposition, we also provide the mathematical justification for the same.

Proposition 1. *Suppose that the objective is to compute the smoothing density $p(x_l|Z_{1:K})$ for $l = 1, \ldots, K$. If $\Pr\left\{\mathcal{H}^f_{k,i}|Z_{1:K}\right\} = 0$, then the i^{th} component can be pruned from $p(x_k|Z_{1:k})$ and propagated to time $k+1$ during FF without affecting the smoothing density $p(x_l|Z_{1:K})$ for $l \geq k$.*

Analogously, if $\Pr\{\mathcal{H}^b_{k,j}|Z_{1:K}\} = 0$, then the j^{th} component can be pruned from $p(Z_{k+1:K} \mid x_k)$ and propagated to time $k-1$ during BF without affecting the smoothing density $p(x_l|Z_{1:K})$ for $l \leq k$.

Proof: In the following, we sketch the proof for the first part of the proposition, i.e., for SPP on the FF. It is possible to derive an analogous proof for SPP on the BF.

Let us first consider how the suggested pruning of components in the FF affect the smoothing density $p(x_k|Z_{1:K})$ at time k. At time k, the relation between the smoothing density $p(x_k|Z_{1:K})$ and the smoothing probability $\Pr\left\{\mathcal{H}^f_{k,i}|Z_{1:K}\right\}$ is given by

$$\begin{aligned} p(x_k|Z_{1:K}) &= \sum_i p(x_k|Z_{1:K}, \mathcal{H}^f_{k,i}) \Pr\{\mathcal{H}^f_{k,i}|Z_{1:K}\} \quad (14)\\ &\propto \sum_i \Pr\{\mathcal{H}^f_{k,i}|Z_{1:k}\} p(x_k|\mathcal{H}^f_{k,i}, Z_{1:k}) \\ &\qquad \times p(Z_{k+1:K}|x_k) \quad (15) \end{aligned}$$

where the product $\Pr\{\mathcal{H}^f_{k,i}|Z_{1:k}\} p(x_k \mid \mathcal{H}^f_{k,i},\, Z_{1:k})$ is the i^{th} component in the filtering density GM $p(x_k|Z_{1:k})$. Clearly, the i^{th} component in the summation of (14), is due to the product of the i^{th} component of the filtering density $p(x_k|Z_{1:k})$ and the BL $p(Z_{k+1:K} \mid x_k)$ as in (15). We note that the $\Pr\left\{\mathcal{H}^f_{k,i}|Z_{1:K}\right\} = 0$, for some value of i, implies that the i^{th} component can be removed from the filtering density without affecting the smoothing density $p(x_k|Z_{1:K})$ at k.

We are now left to prove that the suggested pruning technique does not affect the smoothing density $p(x_l|Z_{1:K})$ for $l = k+1, \ldots K$. To this end, we use the fact that

$$\Pr\left\{\mathcal{H}^f_{k,i}|Z_{1:K}\right\} = \sum_{j \in S_{k \to l,i}} \Pr\left\{\mathcal{H}^f_{l,j}|Z_{1:K}\right\} \quad (16)$$

where the summation is over the set $S_{k \to l,i}$ that contains the indices of the components j at time l which are offsprings of the component i at time k. So, $\Pr\left\{\mathcal{H}^f_{k,i}|Z_{1:K}\right\} = 0 \iff \Pr\left\{\mathcal{H}^f_{l,j}|Z_{1:K}\right\} = 0 \ \forall j \in S_{k \to l,i}$. Arguing as we did for pruning the i^{th} component from $p(x_k|Z_{1:k})$ at time k, we can show that the components $j \in S_{k \to l,i}$ with $\Pr\left\{\mathcal{H}^f_{l,j}|Z_{1:K}\right\} = 0$ can be pruned from $p(x_l|Z_{1:l})$, without affecting the smoothing density $p(x_l|Z_{1:K})$. ■

The smoothing probability of a hypothesis $\mathcal{H}^f_{k,i}$ in the FF can be evaluated from the weights of the components in the smoothing density using (15):

$$\Pr\left\{\mathcal{H}^f_{k,i}|Z_{1:K}\right\} = \int \Pr\{\mathcal{H}^f_{k,i}|Z_{1:k}\} p(x_k|\mathcal{H}^f_{k,i}, Z_{1:k}) \\ p(Z_{k+1:K}|x_k) dx_k$$

$$
\begin{aligned}
&= \int w^f_{k,i}\mathcal{N}\left(x_k;\mu^f_{k,i},P^f_{k,i}\right)\\
&\quad \times\sum_j w^b_{k,j}\mathcal{N}\left(H^b_{k,j}x_k;\mu^b_{k,j},P^b_{k,j}\right)dx_k\\
&= \sum_j w^f_{k,i}w^b_{k,j}\int\mathcal{N}\left(x_k;\mu^f_{k,i},P^f_{k,i}\right)\\
&\quad \times\mathcal{N}\left(H^b_{k,j}x_k;\mu^b_{k,j},P^b_{k,j}\right)dx_k\\
&= \sum_j w^f_{k,i}w^b_{k,j}\\
&\quad \times\mathcal{N}\left(H^b_{k,j}\mu^f_{k,i};\mu^b_{k,j},H^b_{k,j}P^f_{k,i}H^{b^T}_{k,j}+P^b_{k,j}\right). \quad (17)
\end{aligned}
$$

The product of the three terms in the summation of (17) is indeed the weights of the components in the smoothing density $p(x_k|Z_{1:K})$, obtained as the product of the i^{th} component in the filtering density and the BL. Similarly, the smoothing probability for the hypothesis $\mathcal{H}^b_{k,j}$ during BF can be calculated from the smoothing posterior density as

$$
\Pr\left\{\mathcal{H}^b_{k,j}|Z_{1:K}\right\}=\sum_i w^f_{k,i}w^b_{k,j}\times
\mathcal{N}\left(H^b_{k,j}\mu^f_{k,i};\mu^b_{k,j},H^b_{k,j}P^f_{k,i}H^{b^T}_{k,j}+P^b_{k,j}\right). \quad (18)
$$

Note that the summation is over the index i of the filtering density. Analogously, the product in the summation gives the weights of the components in the smoothing density obtained due to product of the j^{th} component in the BL and the filtering density.

VI. Algorithm

In this section, we present the algorithmic description of the TFS algorithm performed in this paper. The computations can be divided into two parts — the FF and the BF. The FF algorithms are well studied in the literature. Therefore, only the algorithmic details of the BF are explained in this section (cf. Algorithm 1). The intragroup approximations described in Section IV and the SPP described in Section V are included in the BF algorithm. The smoothing density will be obtained as part of the SPP performed in the BF as in step 5 of Algorithm 1. We assume that at each k, the output of the FF will be the parameters of the filtering Gaussian mixture: weights $w^f_{k,i}$, means $\mu^f_{k,i}$ and covariances $P^f_{k,i}$. Let M^f_k, M^b_k and M^s_k be the number of components in the output of the FF, BF and the smoothing density respectively.

VII. Implementation and simulation results

A. Simulation scenario

As mentioned before, we consider the problem of tracking a single target moving in a cluttered environment. The model used for simulation is the constant-velocity model where the state vector contains the positions and velocities along x and y dimensions. The target is assumed to be a slowly moving target with acceleration noise standard deviation of $0.07\,m/s^2$. The trajectory is generated for $K = 40$ time steps with a sampling time of $1\,s$. The clutter data is generated in the whole volume of the track.

Algorithm 1 Backward filter of the TFS

```
Input: Likelihoods: $\beta_{k,i}$, $H$, $z_{k,i}$ and $R$ for $i = 0,\dots m_k$ and $k = 1,\dots K$.
Initialize $k \leftarrow K-1$, $w^b_{K,j} \leftarrow 1$, $H^b_{K,j} \leftarrow []$, $\mu^b_{K,j} \leftarrow []$ and $P^b_{K,j} \leftarrow []$ for $j = 0$.
Repeat
  1) Update: for every $i$, $j$, compute $\eta_{k+1,l} = \beta_{k+1,i}w^b_{k+1,j}$,
     $U_{k+1,l} = \begin{bmatrix} H \\ H^b_{k+1,j} \end{bmatrix}$, $\psi_{k+1,l} = \begin{bmatrix} z_{k+1,i} \\ \mu^b_{k+1,j} \end{bmatrix}$ and
     $G_{k+1,l} = \begin{bmatrix} R & \mathbf{0} \\ \mathbf{0} & P^b_{k+1,j} \end{bmatrix}$.
     a) If rank$(U_{k+1,l}) = N_x$, then set $\eta_{k+1,l} = \frac{\eta_{k+1,l}}{\det(U_{k+1,l})}$, $\psi_{k+1,l} = U^{-1}_{k+1,l}\psi_{k+1,l}$, $G_{k+1,l} = U^{-1}_{k+1,l}G_{k+1,l}U^{-T}_{k+1,l}$, and $U_{k+1,l} = I_{N_x}$ end
  2) Grouping: If components $l_1$ and $l_2$ are such that $U_{k+1,l_1} = U_{k+1,l_2}$, then the components belong to the same group.
  3) Intragroup approximation: Within each group, the traditional pruning and merging are performed.
  4) Retrodiction: Set $w^b_{k,l} = \eta_{k+1,l}$, $\mu^b_{k,l} = \psi_{k+1,l}$, $H^b_{k,l} = U_{k+1,l}F$ and $P^b_{k,l} = U_{k+1,l}QU^T_{k+1,l} + G_{k+1,l}$.
  5) Smoothing density: For every $i$ and $j$, compute $w^s_{k,l}$, $\mu^s_{k,l}$ and $P^s_{k,l}$ same way as in the Update step 1 with $k = k-1$, $\beta_{k+1,i} = w^f_{k,i}$, $H = I_{N_x}$, $z_{k+1,i} = \mu^f_{k,i}$ and $R = P^f_{k,i}$. Note that the rank calculated will always be $N_x$ for this case.
  6) SPP: Calculate $w^{bs}_{k,j} = \sum_l w^s_{k,l}$ according to (18), for every $j$. Remove component $j$ from the BL based on the weights, $w^{bs}_{k,j}$ and renormalize.
  7) $k \leftarrow k-1$
until $k = 1$
```

The values for the measurement noise R, the probability of detection P_D and the clutter intensity β, are varied for the simulations. The measurement noise R is set to $50 \times I$ and $150 \times I$. P_D is either 0.7 or 1. The values used for β are 0.0001 and 0.0002. Thus, there are 8 sets of parameters for which the simulation results are compared.

The TFS algorithm is compared with FBS based on an $N-$scan pruning algorithm where the FF is performed using $N-$scan pruning and Rauch-Tung-Striebel algorithm is used on each branch in the filtering hypothesis tree.

B. Implementation details

In both the FF and BF of TFS, to reduce the complexity, extra gating is performed in addition to the conventionally used ellipsoidal gating. This extra gate is rectangular, with dimensions based on the measurement noise covariance and the center at the prediction density mean. The gating probability P_G and the pruning threshold P^f_P for the FF are set as $(1-10^{-5})$ and 10^{-4} respectively.

The merging algorithm in FF is a cheaper variant of Salmond's algorithm [13]. The original Salmond's algorithm looks for the minimum merging cost across every pair of components in the GM. Therefore, it has a quadratic complexity in the number of components. So, to reduce the complexity involved, instead of looking for the minimum cost, we use a heuristic algorithm in this paper. Starting with the components that have the least weights, we compute the cost of merging pairs of components, and if the cost is lower than a threshold (0.001 × state dimension), then the components are merged and replaced in the GM. The procedure is continued with the new reduced GM until there are no pairs that have costs lower than the threshold.

In the BF of the TFS, both the intragroup approximations and the SPP are used to reduce the components in the BL. The intragroup pruning is based on the maximum weight in the group. The components that have weights lesser than $1/100^{\text{th}}$ of the maximum weight in the group are pruned. The intragroup merging is based on the variant of the Salmond's algorithm discussed above. To employ the SPP method followed by these intragroup approximations, first the smoothing density has to be computed. This involves taking the product of the filtering density GM and the RDGM BL which is an expensive operation. So, to reduce the number of operations involved in computing this smoothing density, we reduce the filtering density GM to a single Gaussian, and compute the smoothing density using this reduced filtering density and the BL. Then, using this interim smoothing density, SPP is employed to prune components from the BL.

In case of $N-$scan pruning in the FBS algorithm, the parameter N for the various settings is chosen to be the largest possible N such that the complexity (run-time) for a single run is within the threshold of $2\,s$. The rectangular gating and ellipsoidal gating are also used here.

The performance measures used for comparison are root mean squared error (RMSE), normalized estimation error squared (NEES), complexity and track loss. A track is considered lost if the true state is more than three standard deviations (obtained from the estimated covariance) away from the estimated state for five consecutive time steps. The computational complexity is calculated as the average time taken during MATLAB simulations to run each algorithm on the entire trajectory of 40 time steps. The graphs are obtained by averaging over 1000 Monte Carlo iterations.

C. Results

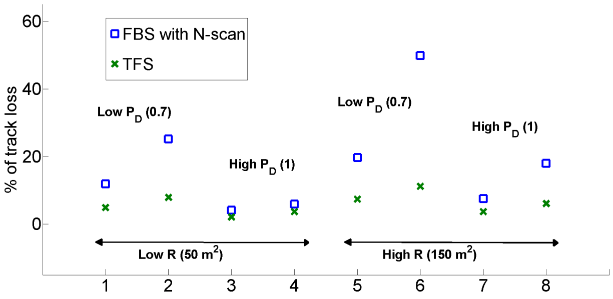

Figure 3: Track Loss performance: Every odd point on x-axis (1, 3, 5, 7) is for low clutter intensity $\beta = 0.0001$ and every even point (2,4,6,8) is for high $\beta = 0.0002$. The order of the eight scenarios is the same also for the others plots in Fig. 4, Fig. 5 and Fig. 6.

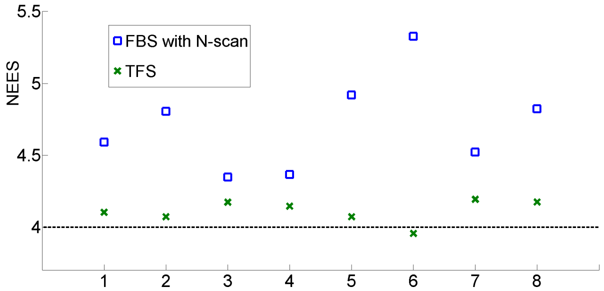

Figure 4: NEES performance: Compared to the FBS, the values of the NEES for the TFS are very close to the optimal value of 4 in all the scenarios.

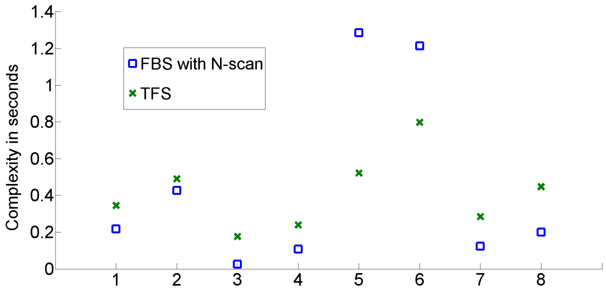

Figure 5: Computational complexity: The TFS algorithm with intra group approximations and SPP is computationally cheaper compared to the FBS with $N-$scan.

The results of the simulation are presented in Fig. 3 to 6. It can be seen that the TFS performs significantly better than the FBS with $N-$scan for most of the scenarios. From the Fig. 3 for track loss performance, one can notice that the performance gain is higher for TFS compared to FBS when P_D is low and the measurement noise R and the clutter intensity β are high (point 6 on the x-axis in Fig. 3). The reason for this is that in these scenarios, the number of components in the filtering GMs before approximations is quite large. To limit the number of components, the pruning during FBS can be quite aggressive resulting in the undesirable 'degeneracy' problem in the FBS. The impact of this degeneracy problem can also be observed in the NEES performance plot in Fig. 4 (point 6 on the x-axis). In the degeneracy case, the uncertainties are underestimated, i.e., the estimated covariances are smaller compared to the optimal, resulting in a larger value for the NEES compared to the expected value of 4. In addition to

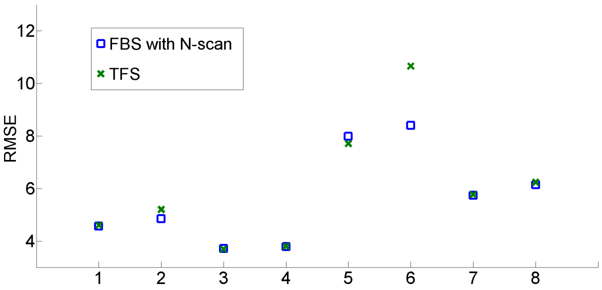

Figure 6: RMSE performance: The results are very similar for the TFS and the FBS.

the better track loss and NEES performance, TFS based on intragroup approximations and SPP offers a computationally cheaper solution compared to the FBS based on $N-$scan pruning as can be observed in Fig. 5. However, the RMSE performance of the TFS and FBS are very similar in most scenarios as seen in Fig. 6.

VIII. Conclusion

In this paper, we present how two-filter smoothing can be performed when the posterior densities are Gaussian mixtures (GMs). GM reduction (GMR) techniques such as pruning and merging are used in the forward filter of the two-filter smoothing method. The structure of the backward likelihood at the output of the backward filter is a reduced dimension GM (RDGM). GMR techniques are also applied within groups of components in the RDGM. Since this strategy has limitations in the reduction, we have proposed the smoothed posterior pruning, where components in the backward likelihood are pruned based on the smoothing posterior weights of those components. The proposed algorithm is shown to have better track loss, root mean squared error, normalized estimation error squared as well as lower computationally complexity compared to a forward-backward smoothing algorithm based on $N-$scan pruning.

References

[1] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, 2009.

[2] I. Bilik and J. Tabrikian, "Maneuvering target tracking in the presence of glint using the nonlinear gaussian mixture kalman filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 1, pp. 246–262, 2010.

[3] S. S. Blackman and R. Popoli, *Design and analysis of modern tracking systems*. Artech House Norwood, MA, 1999, vol. 685.

[4] M. Briers, A. Doucet, and S. Maskell, "Smoothing algorithms for state-space models," *Annals of the Institute of Statistical Mathematics*, vol. 62, no. 1, pp. 61–89, 2010. [Online]. Available: http://dx.doi.org/10.1007/s10463-009-0236-2

[5] D. Crouse, P. Willett, K. Pattipati, and L. Svensson, "A look at gaussian mixture reduction algorithms," in *Proceedings of the fourteenth International Conference on Information Fusion*, 2011, pp. 1–8.

[6] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of Nonlinear Filtering*, vol. 12, pp. 656–704, 2009.

[7] D. Fraser and J. Potter, "The optimum linear smoother as a combination of two optimum linear filters," *IEEE Transactions on Automatic Control*, vol. 14, no. 4, pp. 387–390, 1969.

[8] G. Kitagawa, "The two-filter formula for smoothing and an implementation of the Gaussian-sum smoother," *Annals of the Institute of Statistical Mathematics*, vol. 46, no. 4, pp. 605–623, 1994.

[9] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[10] J. Pearl, "Reverend bayes on inference engines: A distributed hierarchical approach," in *AAAI*, 1982, pp. 133–136.

[11] H. Rauch, F. Tung, and C. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, pp. 1445–1450, 1965.

[12] D. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[13] D. 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.

[14] R. A. Singer, R. G. Sea, and K. Housewright, "Derivation and evaluation of improved tracking filter for use in dense multitarget environments," *IEEE Transactions on Information Theory*, vol. 20, no. 4, pp. 423–432, 1974.

[15] B.-N. Vo, B.-T. Vo, and R. P. Mahler, "Closed-form solutions to forward–backward smoothing," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 2–17, 2012.

[16] J. L. Williams and P. S. Maybeck, "Cost-function-based Gaussian mixture reduction for target tracking," in *Proceedings of the Sixth International Conference of Information Fusion*, vol. 2, 2003, pp. 1047–1054.