Smoothed probabilistic data association filter

Smoothed Probabilistic Data Association Filter

Abu Sajana Rahmathullah, Lennart Svensson, Daniel Svensson, and Peter Willett

***Abstract*—This paper presents the Smoothed Probabilistic Data Association Filter (SmPDAF) that attempts to improve the Gaussian approximations used in the Probabilistic Data Association Filter (PDAF). This is achieved by using information from future measurements. Newer approximations of the densities are obtained by using a combination of expectation propagation, which provides the backward likelihood information from the future measurements, and pruning, which uses these backward likelihoods to reduce the number of components in the Gaussian mixture. Performance comparison between SmPDAF and PDAF shows us that the root mean squared error performance of SmPDAF is significantly better than PDAF under comparable track loss performance.**

***Index Terms*—PDA, filtering, smoothing, factor graph, Gaussian mixtures, message passing, expectation propagation, target tracking, pruning.**

I. Introduction

In target tracking, the sensors pick up signals not only from the target but also from unwanted sources. When tracking with radar, echoes are reflected from ground, sea, etc. which can be sources of clutter signals. Even while tracking a single target, there can be a large number of measurements observed. It can also be that, at times, the target is not observed.

Even under simple assumptions on the process and measurement model — linear with additive Gaussian noise — the problem is computationally intractable. It is easy to show that the true posterior filtered density has the Gaussian Mixture (GM) form with an exponentially increasing number of components over time, thus making approximations inevitable. There are many suboptimal algorithms that have been proposed and been in use. The two major approaches to the problem are one that uses a random-set framework and another that uses a data association framework. Probabilistic Hypothesis Density (PHD) [10] filters belong to the former category, whereas the Multiple Hypothesis Tracking (MHT) filters [13, 4] and the PDAFs [3] belong to the latter. This paper focuses on the data association setting; it should be pointed out that the solution presented in this paper is extendable to scenarios with GMs in general.

In the data association setting, multiple hypotheses are formed, one for each data-to-target association. Under each hypothesis, and under linear-Gaussian assumptions, the filtered posterior is a Gaussian density, thus resulting in a GM for the overall filtered posterior density. This mixture, when propagated through time, has exponentially increasing number of components. Thus, the number of components in the GM has to be reduced. There are several GM reduction algorithms [14, 15, 16] that can be employed to reduce the number of components in the GM. These algorithms vary in terms of complexity, the divergence measure that is minimized, the number of components retained, etc. [7].

MHT uses pruning and/or merging techniques to keep the number of terms under control. The GM reduction at a certain time instant is based on the information from future measurements. But the complexity involved is typically exponential. Based on the parameters of choice, MHT can be quite close to the optimal solution, but the closer it is to the optimal solution, the higher the computational complexity. In PDAF, the GM posterior is reduced to one single Gaussian which has the least Kullback-Leibler (KL) divergence [9] from the mixture. Thus, the PDAF has linear complexity in the number of measurements received.

In this paper, we present the SmPDAF algorithm that uses a combination of PDAF, Expectation Propagation (EP) [11] and pruning to reduce the GM to a single Gaussian density. The approximation is different from the PDAF in the way that before reducing the GM to a single Gaussian, SmPDAF tries to improve the approximation of the prediction density. Likelihood from the future measurements are used to improve the approximations made for the GMs in the past time instants. This improves the prediction, thereby improving the current posterior. This involves iterating between filtering and smoothing [12]. EP is used in obtaining the likelihood from the future measurements, and these likelihoods are used to prune components in the GM of the filtered density. The complexity of the algorithm is linear in the number of measurements and in the depth (or the lag) involved in the smoothing.

The proposed algorithm is compared with PDAF. The two performance measures that are used for comparison are the root mean squared error (RMSE) and the track loss (TL). It will be later shown in the results that although TL is almost the same for both PDAF and SmPDAF, the latter does show a significant improvement in the RMSE performance. The performance has been compared for varying measurement noise levels and probabilities of detection. Results are shown for a single target scenario. A comparison of complexities of PDAF and SmPDAF is also discussed in the results section.

The layout of the paper is as follows: Section II describes the model assumptions, the clutter measurement distribution, etc. The problem statement and a brief description of the idea behind the SmPDAF algorithm are also presented in this section. Section III gives a comprehensive background on filtering, PDAF, smoothing and EP. The SmPDAF algorithm is described in detail in Section IV. It details on how EP helps us to obtain the likelihoods from the future measurements. Also explained is how pruning exploits these backward likelihoods to improve the approximation of the GM densities. An algo-

A. S. Rahmathullah and L. Svensson are with the Department of Signals and Systems, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden, E-mail: {sajana, lennart.svensson}@chalmers.se. D. Svensson is with Electronic Defence Systems, Saab AB, Gothenburg, Sweden, Email: daniel.x.svensson@saabgroup.com. P. Willett is with University of Connecticut, Storrs, CT 06269, USA.

rithmic description of SmPDAF is also provided. Section V presents the results and a discussion on the interpretation of the performance for varying measures. Appendices A and B throw more light into the details of Gaussian divisions involved in EP and the backward predictions, respectively.

II. Problem Formulation and Idea

A. Model assumption

The state at time k is given by the process model,

$$x_k = Fx_{k-1} + v_k$$

and the target measurement at time k is described by the measurement model,

$$z_k^t = Hx_k + w_k$$

where $x_k \in \mathbb{R}^M$, $z_k^t \in \mathbb{R}^N$, $F \in \mathbb{R}^{M\times M}$, $H \in \mathbb{R}^{N\times M}$, $v_k \sim \mathcal{N}(0, Q)$ and $w_k \sim \mathcal{N}(0, R)$.

In this paper, we only consider single-target scenarios. On top of the target-generated measurement, there are also spurious measurements due to clutter. The collection of measurements at time k is represented by the set Z_k and the collection of measurement sets from time 1 to k is given by $Z_{1:k}$. The clutter measurements are assumed to be uniformly distributed over an observed region of volume V, and independent of the target state. The number of clutter measurements is also a random variable that follows a Poisson distribution with parameter λV, where λ is the clutter density. Also, it is assumed that the initial prior, $p(x_1)$, is known and Gaussian. P_D will be the probability of detection.

B. Problem statement and Idea

Under the above-mentioned assumptions, it is easy to show that the filtered posterior density $p(x_k|Z_{1:k})$ is a GM with exponentially increasing number of components. The goal of this paper is to find a better approximation for the filtered density, $p(x_k|Z_{1:k})$.

The filtered posterior density is given by

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

where $p(x_k|Z_{1:k-1})$ is the prediction density and $p(Z_k|x_k)$ is the likelihood from measurements observed at k. Any improvements on the approximation of $p(x_k|Z_{1:k-1})$ will help in improving the approximation of $p(x_k|Z_{1:k})$. This is achieved by improving the approximations of the filtered densities $p(x_l|Z_{1:l})$ at the past time instants $l < k$, using the backward likelihood messages $p(Z_{l+1:k}|x_l)$ from the future measurements. These likelihoods are obtained using EP and are passed backward in time for a fixed number of time steps. The backward likelihoods available at each of these time instants l, are used for pruning some of the components in the GM corresponding to the filtered density $p(x_l|Z_{1:l})$.

III. Background

The focus of the paper is on improving the approximation for filtering. To achieve, this we use smoothing along with EP. A brief background on filtering, smoothing, PDAF and EP is provided in this section. For a more thorough explanation on these topics, readers are referred to [3], [12] and [11].

A. Filtering and PDAF

Filtering, in short, is combining the inference about the current state variable obtained from the past measurements along with the inference made from the current measurements to obtain the filtered posterior density of the state variable. This is performed in a recursive manner so that only the filtered density of the state at past time instant can be used along with the current measurement to obtain the filtered density of the current state variable; not all the past measurements have to be retained. But these operations are not always easy; the densities are complicated and it becomes computationally intractable to use recursion. Approximations are necessary. The PDAF is a filtering algorithm that uses Gaussian approximation at each time instant. The section provides some details on how this is performed. Also presented in this section is a means of involving future measurements to present inference: that is, a smoothed estimate.

Let the state transition density be $f(x_k|x_{k-1})$ and the measurement likelihood be $p(Z_k|x_k)$. This state space model is illustrated in Fig. 1.

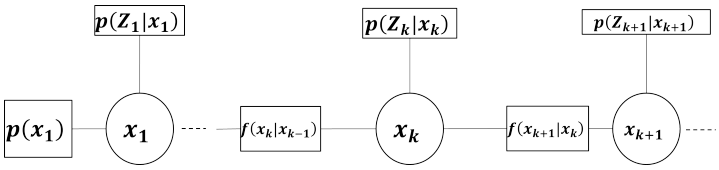

Figure 1. State space model for filtering and smoothing.

Filtering comprises two steps: prediction and update. Prediction is given by

$$p(x_k|Z_{1:k-1}) = \int p(x_{k-1}|Z_{1:k-1})\ f(x_k|x_{k-1})\ dx_{k-1} \tag{2}$$

In this equation, the filtered density at time $k-1$ is propagated through the process model to obtain the prediction of the current state variable x_k from the past measurements $Z_{1:k-1}$. The prediction step is followed by the update step — the information from the current likelihood is updated along with the prediction to obtain the filtered density.

$$p(x_k|Z_{1:k}) \propto p(x_k|Z_{1:k-1})\, p(Z_k|x_k) \tag{3}$$

The two components in equation (3) can be interpreted as:

- $p(x_k|Z_{1:k-1})$ is the information that the measurements from the past, $Z_{1:k-1}$, provide us about the current state x_k.
- $p(Z_k|x_k)$ is the information that the current measurement Z_k has about the current state x_k.

The update step puts together these pieces of information to give us the filtered posterior $p(x_k|Z_{1:k})$.

Under the model assumptions made in section II-A, the likelihood $p(Z_k|x_k)$ is a GM.

$$p(Z_k|x_k) = w_0 + \sum_{i=1}^{m_k} w_i \mathcal{N}(Hx_k; z_{k,i}, R) \tag{4}$$

where $\{z_k^i\}_{i=1,..m_k}$ is the set of all measurements at time instant k, $w_i \propto \frac{1}{m_k}$, for $i = 1, ..m_k$ is the probability of the

hypothesis that the measurement z_k^i correspond to the target, and $w_0 \propto 1 - P_D P_G$ is the probability of the null hypothesis i.e., that the target did not yield any measurement. P_G is the probability of gating.

With a likelihood as in equation (4) and a Gaussian initial prior, the filtered density will be a Gaussian mixture. As this is propagated through time, all the filtered densities will be GMs and the number of terms in the GM grows exponentially with time. Thus, approximation of the GM is necessary. The PDAF approximates the filtered posterior, $p\left(x_k|Z_{1:k}\right)$, with a single Gaussian. Assuming that the prediction $p\left(x_k|Z_{1:k-1}\right)$ is a Gaussian density $\mathcal{N}\left(x_k; \hat{x}_{k|k-1}, P_{k|k-1}\right)$ (which is typically due to the approximations made at the previous time instants), the filtered posterior will be:

$$\begin{aligned} p\left(x_k|Z_{1:k}\right) &\propto p\left(x_k|Z_{1:k-1}\right) p\left(Z_k|x_k\right) \\ &= w_0 \mathcal{N}\left(x_k; \hat{x}_{k|k-1}, P_{k|k-1}\right) \\ &\quad + \sum_{i=1}^{m_k} w_i' \mathcal{N}\left(x_k; \hat{x}_{k|k,i}, P_{k|k,i}\right) \end{aligned} \tag{5}$$

which is again a GM with the same number of components as $p\left(Z_k|x_k\right)$ but with updated means $\hat{x}_{k|k,i}$, covariances $P_{k|k,i}$ and weights w_i'. The new weights w_0 and w_i' are such that the weights shift towards the most likely component in the GM. The PDAF makes the following approximation of the GM:

$$\tilde{p}\left(x_k|Z_{1:k}\right) = \text{MM}\left(p\left(x_k|Z_{1:k}\right)\right)$$

where $\text{MM}\left(\psi(x)\right)$ stands for moment matching that would return a Gaussian with the same first two moments as $\psi(x)$. This method minimizes the KL divergence $\text{KL}\left(p\left(x_k|Z_{1:k}\right) \,||\, \tilde{p}\left(x_k|Z_{1:k}\right)\right)$.

B. Smoothing

Referring back to equation (3), if along with the information, if the information from the future measurements is also incorporated, that gives us the smoothed density. Let us consider smoothing for time instant l such that $l < k$. The filtered density at l will be

$$p\left(x_l|Z_{1:l}\right) \propto p\left(x_l|Z_{1:l-1}\right) p\left(Z_l|x_l\right).$$

The smoothed density at l will be

$$p\left(x_l|Z_{1:k}\right) \propto p\left(x_l|Z_{1:l-1}\right) p\left(Z_l|x_l\right) p(Z_{l+1:k}|x_l). \tag{6}$$

$p(Z_{l+1:k}|x_l)$ is the likelihood of the current state x_l from the future measurements $Z_{l+1:k}$. This likelihood is obtained in two steps, which are similar to the prediction and update step, but in a different order:

- Likelihood update:

$$p(Z_{l+1:k}|x_{l+1}) = p(Z_{l+1}|x_{l+1}) p(Z_{l+2:k}|x_{l+1})$$

- Backward prediction:

$$p\left(Z_{l+1:k}|x_l\right) = \int p\left(Z_{l+1:k}|x_{l+1}\right)\ f\left(x_{l+1}|x_l\right)\ dx_{l+1} \tag{7}$$

C. EP

In the current algorithm, to compute the backward likelihood $p(Z_{l+1:k}|x_{l+1})$, EP [11] is used. In section IV-A, it will be explained how and why EP is employed to obtain an approximation of the backward likelihood. In this section, the general working of EP will be explained. EP is an approximation technique that can be used to obtain a Gaussian approximation of factors of a function. If the function is a product of several non-Gaussian factors, then a Gaussian approximation of each of the factors is obtained using the information from the other factors. This can be performed in an iterative fashion among all the factors in the function. This can also be used to get a Gaussian approximation of the function.

Let $f(x)$ be a function that is made up of several factors $f_i(x)$. These factors are non-Gaussian and we would like to obtain a Gaussian approximation of it.

$$f(x) = \prod_{i=1}^{N} f_i(x)$$

Let us assume that we have an initial approximation of the factors, $\tilde{f}_i(x)$. We are interested in obtaining the approximation for the j^{th} factor $f_j(x)$. The new approximation provided by EP will be

$$\tilde{f}_j^{new}(x) = \frac{\text{MM}\left(f_j(x) \prod_{\substack{i=1 \\ i \neq j}}^{N} \tilde{f}_i(x)\right)}{\prod_{\substack{i=1 \\ i \neq j}}^{N} \tilde{f}_i(x)}$$

The j^{th} factor is replaced with this new approximation and the procedure can be iterated for the remaining factors. The advantage in this is that the approximation of $f_j(x)$ is made in the region where the product of all the factors has its mass. In the next section, there will be more explanation on the relevance of this region of interest.

IV. SmPDAF

The main goal of this paper is to obtain an accurate approximation of the filtered density. SmPDAF tries to make the approximation accurate in the region of interest. In this section we argue that the region of interest is where the smoothed density has its mass, which hence depends on all the available information. An important consequence is that future measurements influence where the density approximations should be accurate, i.e., $Z_{l+1:k}$ influence where our approximation of $p(x_l|Z_{1:l})$ should be accurate in order to yield an accurate approximation of $p(x_k|Z_{1:k})$ (which is our objective). To be noted is that if we have an exact, and Gaussian, expression for the filtered density in the first place, the information from the future is not going to change anything in the filtered density. The idea is to use the older approximations of the filtered density and the information from the future to obtain a more accurate approximation of the filtered density.

Under the model assumptions given in Section II-A, it was shown very briefly in Section III-A that the true filtered density $p(x_k|Z_{1:k})$ is a GM. A PDAF makes a Gaussian approximation (by moment matching) of the filtered density at each stage. This filtered density depends on the prediction density $p(x_k|Z_{1:k-1})$, which in turn depends on (the approximation of) the filtered densities, $p(x_l|Z_{1:l})$ at the past time instants $l < k$. Each of this filtered density is originally obtained through a PDAF. Thus before MM, each of this is a GM but with far lesser number of components than the true filtered density.

The idea is to improve on the GM in PDAF before MM. By improve, we mean that we want the GM to become more unimodal, with the mode corresponding to the true hypothesis gaining more weight. EP and pruning are used to perform this. This involves computing the backward likelihoods $p(Z_{l+1:k}|x_l)$ $(l < k)$ from the future measurements, which again for computational simplicity and tractability, has to be approximated as a Gaussian density. EP is one way to obtain these backward likelihoods. It uses the prediction $p(x_l|Z_{1:l-1})$ and the current likelihood $p(Z_l|x_l)$ to approximate the backward likelihood at the previous time step. Based on this backward likelihood, we use pruning in order to remove components in the GM before MM.

Both EP and pruning use all the available information — prediction, current likelihood and the backward likelihood — to improve on the approximation of the backward likelihood and thereby, the approximation of the filtered density, $p(x_l|Z_{1:l})$. This will later be propagated forward to improve the filtered density $p(x_k|Z_{1:k})$. Thus, it becomes quite natural to make the approximation of these densities accurate in the region which uses the information from all measurements $Z_{1:k}$. Since the smoothed density is obtained using all the information, the region of interest is then the region where the smoothed density has most of its mass. EP and pruning provide an accurate approximation of the backward likelihood and the filtered density in this region of interest. In the following sections, there will be more detailed discussions on the working of EP and pruning.

A. EP

It was argued in the above section that it is crucial to obtain an accurate Gaussian approximation of the backward likelihood $p(Z_{l+1:l}|x_l)$. This can be obtained in two steps:

1) Starting with the Gaussian likelihood $p(Z_{l+2:k}|x_{l+1})$ and the GM likelihood $p(Z_{l+1}|x_{l+1})$ at time $l+1$, EP is used to obtain a Gaussian approximation of $p(Z_{l+1:l}|x_{l+1}) = p(Z_{l+1}|x_{l+1})p(Z_{l+2:k}|x_{l+1})$,
2) The Gaussian approximation from 1) is passed backwards through the process model to obtain the necessary likelihood $p(Z_{l+1:l}|x_l)$. (This step is referred as backward prediction in this paper).

It is emphasized that starting from time $l+1$, these above two steps can be repeated for each l. The details of the backward prediction are provided in Appendix B. To be noted is that given the Gaussian density input, there are no additional approximations made in this backward prediction step.

For notational simplicity and without loss of generality, in this section we will discuss the possibilities to obtain the likelihood $p(Z_{l:k}|x_l)$, instead of $p(Z_{l+1:k}|x_{l+1})$, where

$$p(Z_{l:k}|x_l) = p(Z_l|x_l)p(Z_{l+1:k}|x_l). \tag{8}$$

The current likelihood has the form,

$$p(Z_l|x_l) \quad = \quad w_0 + \sum_{i=1}^{m_k} w_i \mathcal{N}\left(Hx_l; z_{l,i}, R\right).$$

If $p(Z_{l+1:k}|x_l)$ is a proper Gaussian density, the product will yield a GM which can be moment matched and approximated to a Gaussian density. On the other hand, if $p(Z_{l+1:k}|x_l)$ is of the form $\mathcal{N}\left(Ux;\, \mu,\, P\right)$ where the matrix U is not a square matrix, then the product in equation (8) will result in a function of the form,

$$p(Z_{l:k}|x_l) \quad = \quad w_0 \mathcal{N}\left(Ux_l;\, \mu_0,\, P_0\right) + \sum_{i=1}^{m_l} w_i' \mathcal{N}\left(U'x_l;\, \mu_i',\, P'\right).$$

This function is not a density in x_l and cannot be normalized. It is therefore difficult to find an approximation that is Gaussian-like and has the same moments. EP comes to the rescue. EP suggests that the prediction density can be used in obtaining the approximation. It serves two purposes: 1) the approximation is made accurate in the region where the smoothed density has most of its mass, and 2) it overcomes the problem of normalization and computing the moments. EP does the following:

$$p(Z_{l:k}|x_l) \approx \frac{\text{MM}\left(p(x_l|Z_{1:l-1})p(Z_l|x_l)p(Z_{l+1:k}|x_l)\right)}{p(x_l|Z_{1:l-1})} \tag{9}$$

The likelihood $p(Z_{l:k}|x_l)$ is first multiplied by the prediction $p(x_l|Z_{1:l-1})$, to get the smoothed density $p(x_l|Z_{1:k})$. This will result in a GM which can be normalized. A Gaussian approximation of this density is made by moment matching. And the resulting Gaussian density is divided by the prediction to get an approximation of the desired likelihood $p(Z_{l:k}|x_l)$. This approximation for $p(Z_{l:k}|x_l)$ is obtained so that the Kullback-Leibler divergence between the moment matched smoothed density, $p(x_l|Z_{1:l-1})p(Z_{l:k}|x_l)$ and the GM smoothed density $p(x_l|Z_{1:l-1})p(Z_l|x_l)p(Z_{l+1:k}|x_l)$ is minimized, i.e.,

$$\begin{aligned} p(Z_{l:k}|x_l) \quad \approx \quad & \underset{p(Z_{l:k}|x_l)}{\text{argmin}}\, \text{KL}\left\{p(x_l|Z_{1:l-1})p(Z_{l:k}|x_l)\right. \\ & \qquad \left. ||p(x_l|Z_{1:l-1})p(Z_{l:k}|x_l)\right\} \end{aligned}$$

is minimized. Thus the approximation is made such that the smoothed density is more accurate and thus yields an accurate approximation for the backward likelihood. This likelihood can be propagated back through the process model to obtain $p(Z_{l:k}|x_{l-1})$; the details of this are in n Appendix B.

It can be seen from equation (9) that EP involves division of Gaussian densities. This operation is carried out similar to the Gaussian multiplication operation. There will be addition of covariance matrices in Gaussian multiplication to obtain the resultant (product) covariance. In case of Gaussian division, there is subtraction of covariance matrices. The problem is that this may result in a matrix with non-positive eigenvalues. The numerator in the Gaussian division is obtained after

moment matching. Thus the covariance of the numerator will be much wider because of the spread of means term involved in moment matching a GM. If the covariance of the numerator is wider in certain dimensions compared to the covariance of the denominator, the subtraction of covariance matrices will result in a matrix with negative eigenvalues. The approach taken in overcoming this is to use only the dimensions along which the eigenvalues are positive and to ignore the remaining dimensions. The details of how this is performed is explained in Appendix A.

B. Pruning

EP provides us with a good approximation to the backward likelihood $p(Z_{l+1:k}|x_l)$. The question now is how this backward likelihood can be used to obtain an approximation of the filtered density $p(x_l|Z_{1:l})$.

One natural way of utilizing the backward likelihood would be to use EP as

$$p(x_l|Z_{1:l}) \approx \frac{\text{MM}\left(p(x_l|Z_{1:l-1})p(Z_l|x_l)p(Z_{l+1:k}|x_l)\right)}{p(Z_{l+1:k}|x_l)}.$$

The problem with this method is that it involves Gaussian division and can result in a density with infinite covariance along certain state dimensions, which is not desirable.

Another way would be to use the smoothed density as an approximation,

$$p(x_l|Z_{1:l}) \approx \text{MM}\left(p(x_l|Z_{1:l-1})p(Z_l|x_l)p(Z_{l+1:k}|x_l)\right).$$

The disadvantage with this would be that as we iterate between the approximations of the backward likelihood $p(Z_{l+1:k}|x_l)$ and the filtered density $p(x_l|Z_{1:l})$, the information in $p(Z_{l+1:k}|x_l)$ will be overused (this is sometime referred to as data "incest") and can result in an underestimated covariance.

The method that is suggested in this paper is to perform pruning on the filtered GM $p(x_l|Z_{1:l})$ based on the smoothed density which uses the backward likelihood $p(Z_{l+1:k}|x_l)$. The filtered density and the smoothed density are GMs (but we assume that the prediction and the backward likelihood are Gaussian functions). Let them be denoted as:

$$\begin{aligned} p(x_l|Z_{1:l}) &= \sum_{i=0}^{m_l} w_i \mathcal{N}\left(x_l;\, \mu_l^i,\, P_l^i\right) \\ p(x_l|Z_{1:k}) &= \sum_{i=0}^{m_l'} w_i' \mathcal{N}\left(x_l;\, \eta_l^i,\, B_l^i\right) \end{aligned}$$

The different components in the GM correspond to the hypothesis that the measurement z_l^i is due to the target, and $i = 0$ refers to a missed detection. If the weights of some of the components in the GM $p(x_l|Z_{1:k})$ are negligible compared to the weights of the filtered density GM, then it would imply that the backward likelihood indicates that the impact of those components become insignificant in the future time instants. If the impact of those components is negligible, these components may as well be pruned in the filtered GM density at time l. This helps in reducing the number of components in the GM. The lesser the number of components in the GM, the easier (or more accurate) can be the approximation of the GM to single Gaussian density. Thus, a better and an easier approximation for the filtered density $p(x_l|Z_{1:l})$ would be to prune those components and moment-match the remaining components. Thus the idea is to prune the components in the filtered density GM that have negligible weights (or insignificant) in the smoothed density. The remaining components in the filtered density are moment-matched to obtain a Gaussian density approximation.

The pruning operation will be denoted as '$\text{Prune}_{1:k}$' in the algorithmic description 1.

C. Algorithmic Description

An algorithmic description of SmPDAF is provided as Algorithm 1. It is assumed that the lag parameter L and the iteration parameter, J are defined. For each time instant, k, the procedure is explained in Algorithm 1. The filtered density at time $k-1$ is assumed to be available.

Algorithm 1 Algorithmic description of SmPDAF

1) Propagate the prior through the process model to get the prediction $p(x_k|Z_{1:k-1})$ as in equation (2), compute the filtered density at k as in equation (5) followed by moment matching and set $b = k$.
2) At b, compute the backward likelihood as in equation (9) with $l = b$ and propagate it backward through the process model as in equation (7) with $l = b-1$. Set $b = b-1$.
3) Repeat step 2) until $b = k - L$ and set $f = b$.
4) At f, compute the new prediction as in equation (2) and perform pruning on filtered density as
$$p(x_f|Z_{1:f}) = \text{Prune}_{1:k}(p(x_f|Z_{1:f-1})\, p(Z_f|x_f)),$$
perform moment matching as
$$p(x_f|Z_{1:f}) = \text{MM}(p(x_f|Z_{1:f}))$$
and set $f = f+1$.
5) Repeat step 4) until $f = k$.
6) Repeat steps 2) to 5) the specified iteration 'J' number of times.

V. Results

Performance comparisons between SmPDAF and PDAF are presented in this section. Simulation comparisons of the root mean squared error and the track loss between the PDAF and SmPDAF are presented. The asymptotic complexity of the SmPDAF algorithm is discussed later.

A. Root Mean Squared Error and Track Loss

The model considered for simulations is the constant velocity model with a two dimensional state vector containing position and velocity. The trajectory was generated with zero process noise. The initial speed was 50m/s. For the simulations, the acceleration noise had a standard deviation of around 4m/s^2. The lag parameter L, for the SmPDAF was set as 5 time

steps and for each time instant, the SmPDAF was performed once, i.e., the iteration parameter J was set as 1.

The performance measures considered are RMSE and TL. The results are averaged over 1000 Monte Carlo iterations. The RMSE performance is shown in Fig. 2 and Fig. 3. The RMSE in the position obtained with PDAF is 10% higher than that of SmPDAF for higher P_D(See Fig. 2); for lower P_D, PDAF has 5% more RMSE in position than SmPDAF. With respect to the RMSE in velocity, PDAF has around 2% more RMSE than SmPDAF.

The TL decision was made by thresholding the Mahalanobis distance between the estimated state and the true state. The TL of SmPDAF is slightly better than the TL of the PDAF for higher P_D (See Fig. 4). For lower P_D, SmPDAF has higher track loss than PDAF. A closer analysis showed that the tracks were lost when the approximation of the backward likelihood made by EP was not accurate. This mainly happened when the target measurements were missed and were not part of the backward likelihood.

B. Complexity

The complexity of SmPDAF is analyzed as a function of the number of terms in the GM filtered density. It will be argued that smoothing back until a certain lag and filtering forward have same complexity (asymptotically).

At each time instant, SmPDAF performs PDAF and smoothing back until a certain time step, followed by forward filtering forward employing pruning. Thus, the complexity of SmPDAF can be grouped into three parts: PDAF (Step 1 in Algorithm 1), smoothing back until a certain lag (Step 2), forward filtering with pruning (Step 4). PDAF involves MM of the filtered GM density at each time instant. Thus PDAF has linear complexity in the number of components in the GM, which depends on the number of clutter measurements at each time instant. On an average, PDAF has a complexity linear in the expected number of measurements at a time instant, let's say 'C'.

The smoothing step is very similar to the PDAF step. In smoothing, there is the GM backward likelihood instead of a GM filtered density; instead of direct MM, EP is performed. EP has an additional step of Gaussian division, which is independent of the number of components in GM. Thus, following a similar argument as before, smoothing for a time step has linear complexity in the expected number of measurements, C. This smoothing back is performed until certain lag, say 'L'. Thus, this results in a complexity of $L \times C$ for the smoothing step.

The filtering forward with pruning step is again very similar to PDAF, with additional pruning happening before MM. Thus, filtering forward with pruning for 'L' times steps has a complexity of $L \times C$. The smoothing and the filtering forward with pruning steps are repeated 'J' number of times (Step 6)). Putting all these together, SmPDAF has a complexity of $(2 \times L \times J + 1)C$ for each time step, which is asymptotically linear in $L \times C \times J$.

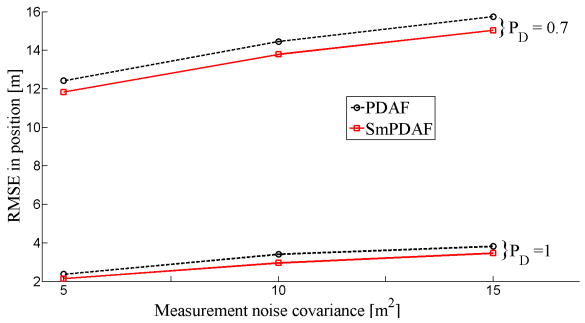

Figure 2. Comparison of the RMSE in position between SmPDAF and PDAF

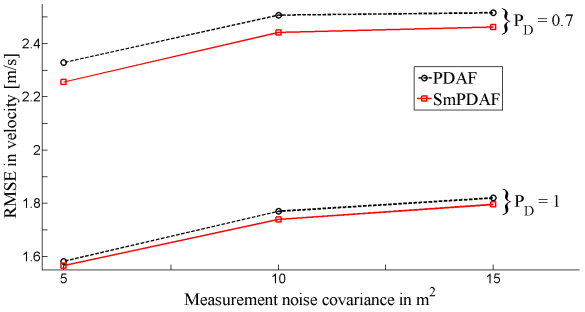

Figure 3. Comparison of the RMSE in velocity between SmPDAF and PDAF

VI. Conclusions

In this paper, the Smoothed Probabilistic Data Association Filter algorithm was presented. The core idea of the algorithm is to exploit the information from the future to improve on the Gaussian approximation of the filtered density at the current time instant. The work done in this paper presented a way to use the idea of a combination of the probabilistic data association filter (PDAF), expectation propagation (EP) and pruning. To compute the backward likelihood from future measurements, EP was employed. Pruning was a way suggested to use this backward likelihood to improve the approximation of the filtered density. Both EP and pruning work in such a way that the approximations are made accurate in the region of interest. The RMSE performance of the SmPDAF algorithm was found to be significantly better than that of PDAF in a simulated single-target scenario with clutter and non-unity probability of detection.

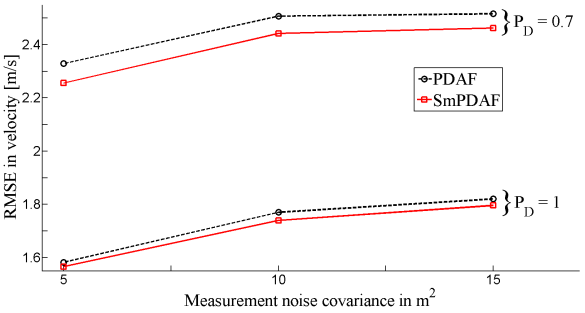

Figure 4. TL comparison between SmPDAF and PDAF

APPENDIX A
GAUSSIAN DIVISIONS

When computing the backward likelihoods from the future measurements, EP is used as in equation (9). This involves division of Gaussian densities that leads to subtracting the inverses of the two covariance matrices, which may result in covariance matrices with negative eigenvalues. One way to overcome this is to exploit the dimensions along which the eigenvalues are positive, and propagate the corresponding density.

Let us assume that the densities involved in the Gaussian division are $\mathcal{N}(x;\mu_a,P_a)$ and $\mathcal{N}(x;\mu_b,P_b)$ and the interest is in computing the function $g(x)$ such that it has a Gaussian form with positive definite covariance matrix.

$$\begin{aligned} g(x) &\propto \frac{\mathcal{N}(x;\mu_a,P_a)}{\mathcal{N}(x;\mu_b,P_b)} \\ &\propto \exp\left\{-\frac{1}{2}\left[x^T\left(P_a^{-1}-P_b^{-1}\right)x \right.\right. \\ &\quad \left.\left. -2x^T\left(P_a^{-1}\mu_a - P_b^{-1}\mu_b\right)\right]\right\} \end{aligned}$$

Let the eigendecomposition of $P_a^{-1}-P_b^{-1}$ be $U^T\Lambda U$. Λ is such that

$$\Lambda = \begin{bmatrix} \Lambda_p & \mathbf{0} \\ \mathbf{0} & \Lambda_{np} \end{bmatrix}$$

where Λ_p and Λ_{np} are diagonal matrices with positive eigenvalues and non-positive eigenvalues along their diagonals, respectively. The eigenvectors in the U matrix are sorted accordingly,

$$U = \begin{bmatrix} U_p \\ U_{np} \end{bmatrix}$$

and $U_pU_p^T = I$. The approximation made will be that $P_a^{-1}-P_b^{-1}$ is replaced with $U_p^T\Lambda_pU_p$, thereby using only the dimensions along which the eigenvalues are positive. Thus,

$$g(x) \propto \mathcal{N}(U_px;\mu_p,P_p)$$

where $\mu_p = \Lambda_p^{-1}U_p\left(P_a^{-1}\mu_a - P_b^{-1}\mu_b\right)$ and $P_p = \Lambda_p^{-1}$.

APPENDIX B
BACKWARD PREDICTION

The backward likelihood functions used in SmPDAF have to be propagated back through the process model as in equation (7), This operation is very similar to the prediction step in filtering. Equation (7) is

$$p(Z_{k+1:N}|x_k) = \int p(Z_{k+1:N}|x_{k+1})\ f(x_{k+1}|x_k)\ dx_{k+1}$$

The likelihood $p(Z_{k+1:N}|x_{k+1})$ is a function obtained by using EP that involves Gaussian division as in Appendix A. Thus, it will have the form $\mathcal{N}(U_px_{k+1};\ \mu_p,\ P_p)$. The state transition density is given by $\mathcal{N}(x_{k+1};\ Fx_k,\ Q)$.

$$\begin{aligned} p(Z_{k+1:N}|x_k) &= \int \mathcal{N}(U_px_{k+1};\ \mu_p,\ P_p)\times \\ &\quad \mathcal{N}(x_{k+1};\ Fx_k,\ Q)\,dx_{k+1} \\ &= \int \mathcal{N}(x_{k+1};\bullet,\bullet)\times \\ &\quad \mathcal{N}\left(U_pFx_k;\ \mu_p,\ U_pQU_p^T+P_p\right)dx_{k+1} \end{aligned}$$

The second term in the above integral is independent of x_{k+1} and hence can be moved out of the integral and the first term is a Gaussian density in x_{k+1} and integrates to unity. Thus,

$$p(Z_{k+1:N}|x_k) = \mathcal{N}\left(U_pFx_k;\ \mu_p,\ U_pQU_p^T+P_p\right)$$

REFERENCES

[1] Y. Bar-Shalom and A. Jaffer, "Adaptive nonlinear filtering for tracking with measurements of uncertain origin," in *Conference on Decision and Control and Symposium on Adaptive Processes, 11 th, New Orleans, La*, 1972, pp. 243–247.

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

[3] Y. Bar-Shalom and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, vol. 11, no. 5, pp. 451–460, 1975.

[4] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 19, no. 1, pp. 5–18, 2004.

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

[6] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Multi-target tracking using joint probabilistic data association," in *Decision and Control including the Symposium on Adaptive Processes, 1980 19th IEEE Conference on*, vol. 19. IEEE, 1980, pp. 807–812.

[7] J. R. Hershey and P. A. Olsen, "Approximating the Kullback Leibler divergence between Gaussian mixture models," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 4. IEEE, 2007, pp. IV–317.

[8] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *Information Theory, IEEE Transactions on*, vol. 47, no. 2, pp. 498–519, 2001.

[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] R. P. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 39, no. 4, pp. 1152–1178, 2003.

[11] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2001, pp. 362–369.

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

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

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

[15] 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, 2009.

[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.